

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

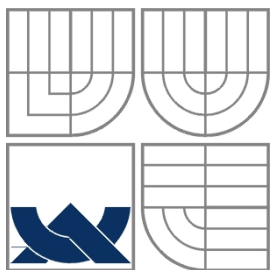
MĚŘENÍ TEPLoty VODY V MASÁŽNÍM ZAŘÍZENÍ

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

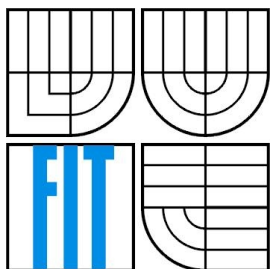
AUTOR PRÁCE
AUTHOR

MICHAL GÁBRIŠ

BRNO 2011



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

MĚŘENÍ TEPLoty VODY V MASÁŽNÍM ZAŘÍZENÍ

TEMPERATURE MEASURING IN WELLNESS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MICHAL GÁBRIŠ

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PAVEL BARTOŠ

Abstrakt

Tato bakalářská práce se zabývá tvorbou operačního programu pro digitálně kalibrovaný teploměr se dvěma měřenými teplotami, který se používá v masážních zařízeních. Teploměr pracuje ve dvou režimech, a to v režimu měření a zobrazování teploty vody a v režimu lineární digitální kalibrace, ve kterém se nastavují teplotní čidla. Práce obsahuje jak popis použitého hardware, tak i softwarového řešení, které bylo implementováno v jazyce assembler. Toto zařízení je již komerčně využíváno.

Abstract

This Bachelor thesis deals with creation of operation program for digitally calibrated thermometer measuring two temperatures, which can be used in wellness centers. The thermometer operates in two modes, the mode of measuring and displaying of water temperature, and the mode of linear digital calibration, in which the temperature sensors are set. The Bachelor thesis contains both the description of used hardware components, and software solution implemented in assembly language. This device is already in commercial use.

Klíčová slova

vestavěný systém, mikroprocesor, Microchip, PIC, měření, teplota, masážní zařízení, voda, assembler

Keywords

embedded system, microprocessor, Microchip, PIC, measurement, temperature, wellness, water, assembler

Citace

Michal Gábriš: Měření teploty vody v masážním zařízení, bakalářská práce, Brno, FIT VUT v Brně, 2011

Měření teploty vody v masážním zařízení

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Pavla Bartoše. Další informace mi poskytl zadavatel práce, Ing. Jozef Chamraz, CSc. ze společnosti INOVATICA, s.r.o.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Michal Gábriš
18.5.2011

Poděkování

Tímto bych chtěl poděkovat především dvěma lidem, a to vedoucímu práce, panu Ing. Pavlovi Bartošovi, za starostlivý a odborný přístup k práci, a zadavateli práce, panu Ing. Jozefu Chamrazovi, Csc., za přípravu zadání, které je přímo z praxe a za pomocnou ruku počas celé tvorby bakalářské práce. Dále bych chtěl poděkovat všem, kteří mě podporovali a přáli mi štěstí – své rodině a přátelům.

© Michal Gábriš, 2011

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod.....	3
1.1 Popis jednotlivých kapitol.....	3
2 Požiadavky trhu.....	4
2.1 Situácia na trhu	4
2.2 Úžitkové vlastnosti.....	4
2.3 Spoľahlivosť.....	5
3 Technické riešenie.....	7
3.1 Mikroprocesor PIC12F675.....	7
3.1.1 Zapojenie PINov mikroprocesora.....	7
3.1.2 Organizácia pamäte.....	8
3.1.3 Pamäť EEPROM.....	10
3.1.4 Analógovo-digitálny prevodník.....	11
3.2 Posuvný register 74HC164 SMD.....	12
3.3 Digitálny číslicový displej DA03-11EWA.....	13
3.4 Teplotný senzor KTY81-120.....	14
3.5 Tranzistory BC807-40SMD a BC817-40SMD.....	14
3.6 Technické obmedzenia.....	14
3.7 Adaptér pre digitálnu kalibráciu.....	16
3.7.1 Sériová komunikácia.....	16
3.8 Programátor ELNEC SmartProg2.....	17
4 Riešenie programu.....	20
4.1 Voľba vývojového prostredia	20
4.2 Výber programovacieho jazyka.....	20
4.3 Počiatočná konfigurácia mikrokontroléra.....	20
4.3.1 Konfigurovateľné obvody.....	20
4.3.2 Výber oscilátora.....	21
4.3.3 Power-up Timer.....	21
4.3.4 Watchdog Timer.....	21
4.3.5 Brown-out Detect.....	22
4.4 Inicializácia.....	22
4.4.1 Počiatočné nastavenia.....	22

4.4.2	Asynchrónny sériový prenos.....	23
4.4.3	Podprogram receive.....	23
4.4.4	Podprogram transmit.....	24
4.4.5	Odosielanie dát na displej.....	24
4.5	Meranie a zobrazovanie 2 teplôt.....	24
4.5.1	Knižnice pre výpočty s pohyblivou rádovou čiarkou.....	26
4.5.2	Normalizácia čísla.....	26
4.5.3	Nastavenie A/D prevodníka.....	26
4.5.4	Merací filter.....	27
4.5.5	Zosilnenie a ofset.....	27
4.5.6	Kontrola výsledku.....	27
4.5.7	Prevod hodnoty teploty na desiatky a jednotky.....	27
4.5.8	Získanie kódu číslice.....	28
4.6	Digitálna kalibrácia.....	28
4.6.1	Čítanie.....	29
4.6.2	Zápis kalibračných hodnôt do pamäte EEPROM.....	30
4.6.3	CRC.....	31
4.6.4	Generovanie CRC.....	32
4.6.5	Implementácia digitálnej kalibrácie.....	32
5	Finálny výrobok.....	33
6	Záver.....	35

1 Úvod

Táto bakalárska práca vznikla v prostredí Fakulty Informačných Technológií na Vysokém Učení Technickém v Brne. Práca mi bola zadaná firmou Inovatica, s.r.o., so sídlom v Malackách, Slovenská Republika, ktorá sa špecializuje na moderné riešenia elektroniky a to hlavne zapuzdrených systémov. (Embedded Systems). Hlavnou požiadavkou zadávateľa bolo vytvorenie riadiaceho programu s dvomi módmí činnosti:

- Mód1 – meranie dvoch teplôt z dvoch teplotných senzorov
- Mód2 – digitálna kalibrácia dvoch teplôt

Vzhľadom k tomu, že hlavnými určujúcimi parametrami výsledného produktu boli výsledky marketingu, požiadavky vyplývajúce z bezpečnostných predpisov a dizajnu pre cieľovú používateľskú oblasť, bolo nutné aj vývoj softwaru týmto skutočnostiam podriaďiť. Zadávateľ kladol veľký dôraz na výslednú kvalitu a spoľahlivosť produktu. Softwarové prostredie musí v maximálnej miere zaistiť ochranu kalibračných údajov uložených v pamäti EEPROM riadiaceho mikroprocesora, pretože export zariadenia sa predpokladá do takých destinácií v ktorých re-kalibrácia zariadenia neprichádza do úvahy.

1.1 Popis jednotlivých kapitol

Uvedieme si stručný popis o čom pojednávajú jednotlivé kapitoly práce.

Kapitola 2: Požiadavky trhu – v 2. kapitole budeme písať o situácii na trhu, na ktorý chceme s našim výrobkom vstúpiť, o požiadavkách na výrobok, aby mohol byť komerčne úspešný, a obmedzeniach, ktoré sme dostali od zákazníka.

Kapitola 3: Technické riešenie – v tejto kapitole sa podrobnejšie pozrieme na už vyrobený plošný spoj s komponentmi, z ktorých hlavné sú: mikroprocesor, posuvný register a 7-segmentový displej.

Kapitola 4: Riešenie programu – táto kapitola je jadrom celej práce a pojednáva o kompletnom návrhu programu pre mikroprocesor od výberu vývojového prostredia a programovacieho jazyka, cez riešenie 3 hlavných častí programu: 1. rozhodovanie, do akého stavu sa prepnúť, 2. meranie a zobrazovanie teplôt, 3. digitálna kalibrácia.

Kapitola 5: Finálny výrobok – v tejto kapitole nájdeme fotografie finálneho výrobku a stručne ho popíšeme

Záver – v poslednej kapitole zhrnieme význam práce a zamyslíme sa nad možnými rozšíreniami do budúcnosti.


Prílohy – V priložených dokumentoch sa nachádzajú informácie o hardwarovom riešení produktu a tiež výsledky z vývoja softwaru.

2 Požiadavky trhu

Obsahom tejto kapitoly je situácia na trhu, na ktorý chceme s našim výrobkom vstúpiť, požiadavky na výrobok, aby mohol byť komerčne úspešný, a obmedzenia, ktoré sme dostali od zákazníka.

2.1 Situácia na trhu

Podľa zistenia zákazníka, ktorý vyrába zariadenia pre hydroterapiu a wellness, je na európskom trhu vedúca firma produkujúca mechaniku a elektroniku pre tento trhový segment firma Koller. S našim zariadením musíme vstúpiť do jeho teritória. Táto firma vyrába aj príslušný výrobok s podobnými vlastnosťami (viď. obr. 2.1). Zákazník nás limitoval pri zadaní tvarom, rozmermi, IP krytím a finálnou cenou 40 € vrátane dvoch sensorov. (Všimnite si prosím cenový rozdiel.) Požadoval dve meracie miesta (fy Koller má len jedno) a meranie teploty v °C. Z týchto dôvodov a hlavne z dôvodov bezpečnosti zaistovanej aj IP krytím sa zadávateľ rozhodol nakupovať a používať puzdro citovanej firmy. Rozmer plošného spoja výrobku je potom limitovaný na priemer 29 mm. Požadovaná cena a rozmery výrobku výrazne ovplyvnili technické riešenie.

Description	Order No.	Price/Unit EURO
 <p>electronic temperature control device button ring: plastic body: plastic with 2 m cable mounted, with tin-plated ends of the line weight: 104 g</p> <p>functions: display of the water resp. cabin temperature temperature control of adjustable set point</p> <p>Attention! Temperature sensor necessary, supply 9-12V DC, through all Koller control panels possible. recommended control panels: 230 8022A-S15, 230 8014A-(S26,S27,...), 230 5151A-S04 It is possible to indicate the temperature in °C or °F.</p>	P-W-3321C-S01	198,20
	P-C-3321C-S01	201,27
	P-G-3321C-S01	204,79
	P-R-3321C-S01	199,73

Obr. 2.1: Popis konkurenčného výrobku firmy Koller

2.2 Úžitkové vlastnosti

Úžitkové vlastnosti nášho zariadenia musia byť lepšie. Dizajn, vzhľad, obsluha a montáž musia byť rovnaké, pretože náš výrobok nahradí výrobok konkurenčnej firmy. V porovnaní s ním musí byť veľmi atraktívna cena, pretože nahradzujeme už overený a zavedený komponent. Pre docielenie relatívne nízkej ceny je potrebné uplatniť netradičné riešenia hardwaru a netradičnú podporu tohto riešenia softwarovými prostriedkami.

Firma Koller používa bodový displej SLR2016 so špeciálnymi ASCII znakmi, ale s výškou znaku iba 4,57 mm a šírkou 2,54 mm, 5x7 bodov. Podľa nás takéto riešenie nezohľadňuje nasledovné:

1. Teplomer využíva jedine číslice.
2. Bodový displej je menej čitateľný.
3. Takýto typ displeja má komplikované ovládanie (zbernica pripomínajúca pamäť RAM).
4. Nakoľko sa jedná o špeciálny obvod, ktorý nemá vysokú používateľnosť, tento obvod má aj vysokú cenu.
5. Na riadenie používa firma Koller mikroprocesor ST62T55B s 28 vývodmi, ktorý tiež nepredstavuje výrobok masívne používaný a je skôr špeciálny.

2.3 Spol'ahlivosť

Už z návrhu musí vyplynúť vysoká spol'ahlivosť. Problém spol'ahlivosti je riešený technickými prostriedkami a návazne programovými prostriedkami (napr. údaje o kalibrácii v pamäti EEPROM musia byť odolné voči náhodnému prepísaniu, ktoré by mohlo vzniknúť v kritických situáciách).

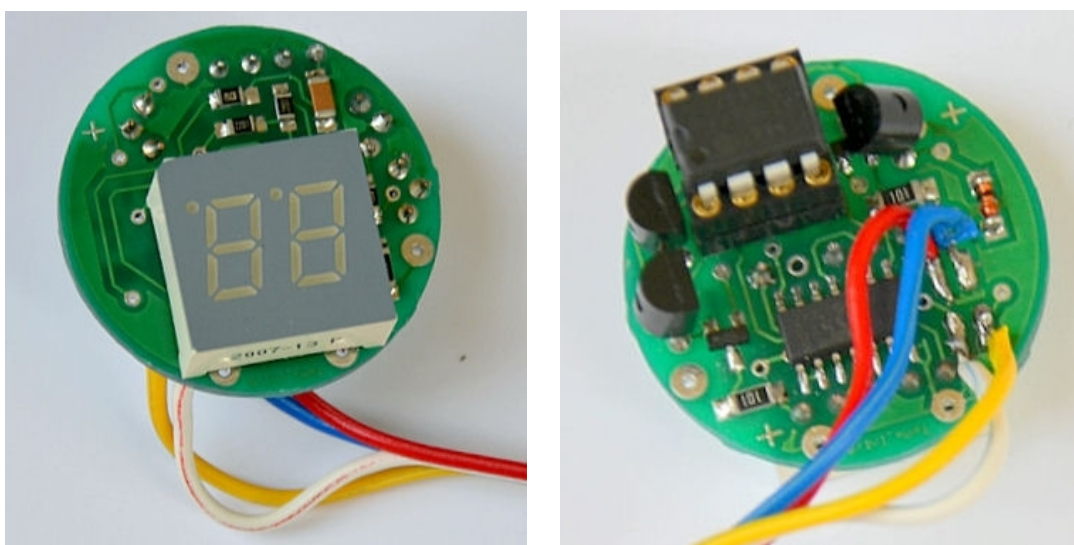
Celková spol'ahlivosť je daná spol'ahlivosťou hardwaru a programového vybavenia. V práci sa sústreďujeme na spol'ahlivosť programového vybavenia. Programové vybavenie musí zaistiť dve hlavné úlohy:

1. V prípade, že nie je pripojený počítač PC a neprebíha kalibračný proces, program musí spol'ahľivo pracovať v slučke meranie a zobrazovanie teplôt. Kalibračné údaje, to znamená údaje charakterizujúce rovnice priamky, uložené v pamäti EEPROM, musia byť dokonale chránené proti náhodnému prepísaniu. K tomuto prepísaniu nesmie prísť ani pri extrémnych situáciách (napr. búrka, hrubé zaobchádzanie zo strany obsluhy, poruchová činnosť napájacieho zdroja). Zariadenia sú exportované prakticky do celého sveta a akékoľvek reklamačné riešenie neprichádza do úvahy. Takisto neprichádza do úvahy re-kalibrácia zariadenia, prípadné zmeny parametrov musia byť zahrnuté v technických údajoch zariadenia.
2. Riešenie hardwaru minimalizuje počet aktívnych a pasívnych prvkov, ktoré by mohli ovplyvniť presnosť zariadenia. Táto minimalizácia si však vyžiadala dodatočnú softwarovú podporu, ktorú nazývame digitálna kalibrácia. Z konštrukčných dôvodov sa táto kalibrácia vykonáva pomocou prepojovacieho adaptéra RS232, ktorý je riešený neštandardným spôsobom. (Tento spôsob je popísaný v kapitole 3.5 Technické obmedzenia.) Programové vybavenie digitálnej kalibrácie musí implementovať 2 povely – zápis a čítanie z protokolu MODBUS vrátane kontroly pomocou CRC skúšobnej sumy. Pri riešení programového

vybavenia je treba ešte zaistiť ochranu proti náhodnému vyvolaniu zapisovacej procedúry.

3 Technické riešenie

V tejto kapitole budeme hovoriť o technických prostriedkoch, ktoré budeme využívať v práci. Pôjde hlavne o popis použitého hardwaru, ktorý bol navrhnutý a vyrobený firmou Inovatica, s.r.o. a spomenieme aj technické obmedzenia, ktoré vyplývajú z požiadaviek zákazníka, najmä na cenu a rozmery výrobku. Budú tu popísané komponenty plošného spoja (ktorého schéma je v prílohe 1.), adaptér pre digitálnu kalibráciu (schéma je v prílohe 2) a programátor ELNEC SmartProg2.



Obr. 3.1: Pohľad na hardwarové riešenie spredu a zozadu

3.1 Mikroprocesor PIC12F675

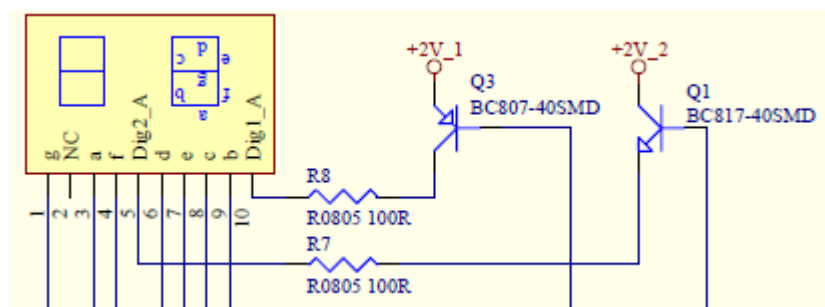
Jednoduchý mikroprocesor Microchip PIC12F675 (Obrázok 3.2) slúži k riadeniu teplomera. Popis mikroprocesora a jeho obvodov je prevzatý z [1]. Tento mikroprocesor sa vyznačuje tým, že je typu RISC (Reduced Instruction Set Computer), má iba 35 inštrukcií a všetky, okrem inštrukcií skoku, trvajú jeden inštrukčný cyklus. Mikroprocesor má 8 vývodov, je 8-bitový a založený na technológiách Flash a CMOS. Čo sa týka prevádzkových rýchlostí, má 20 MHz oscilátor a 200 ns inštrukčný cyklus. Ďalej umožňuje prerušenie, má 8 úrovňový hardwarový zásobník a 3 adresovacie módy – priame, nepriame a relatívne adresovanie.

3.1.1 Zapojenie PINov mikroprocesora

Zapojenie jednotlivých vývodov mikroprocesora je znázornené na obr. 3.2, kompletná schéma teplomera je v prílohe 1.

Zapojenie:

- vývod 1 VDD – zdroj napätia
- vývod 8 VSS – uzemnenie
- vývod 2 GP5 – je použitý ako výstupný a pomocou neho prepíname zobrazovanie a nastavovanie jednej alebo druhej číslice displeja. Toto je umožnené 2 tranzistormi Q1 a Q3 (obr. 3.2) tak, že či už je výstupom z vývodu nízka alebo vysoká hladina napätia, vždy bude výstupom z jedného tranzistoru vysoká hladina a z druhého nízka hladina napätia, a tak bude súčasne rozsvietená len jedna číslica displeja.
- vývod 3 GP4 – je použitý ako vstupný, a jeho vstupom je nameraná teplota č.1 z termocitlivého rezistora RT1
- vývod 4 GP3 – je použitý ako vstupný, a jeho vstupom je signál z tlačidla J1 a zároveň v režime digitálnej kalibrácie slúži ako vstup sériového prenosu
- vývod 5 GP2 – je použitý ako výstupný, a posieľa dáta do posuvného registra U2, odkiaľ sa potom zobrazia na displeji Disp1 a zároveň v režime digitálnej kalibrácie slúži ako výstup sériového prenosu
- vývod 6 GP1 – je použitý ako výstupný, a je to zdroj hodinového signálu pre posuvný register U2
- vývod 7 GP0 – je použitý ako vstupný, a jeho vstupom je nameraná teplota č.2 z termocitlivého rezistora RT2



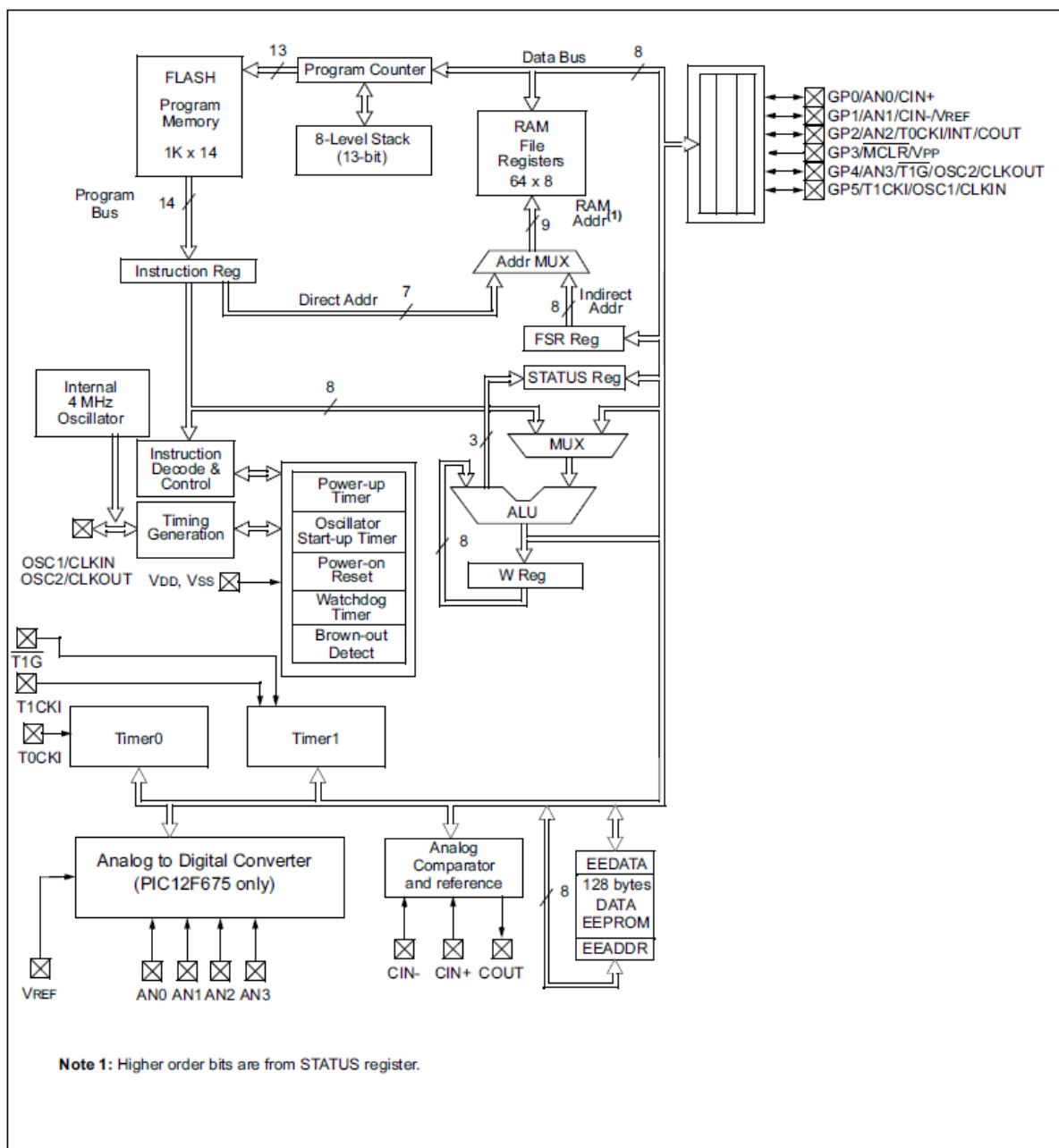
Obr. 3.2: Zapojenie tranzistorov Q1 a Q3

3.1.2 Organizácia pamäte

Mikroprocesor obsahuje celkovo 3 druhy pamätí, a to pamäť FLASH pre program, pamäť RAM pre uloženie dát a pamäť EEPROM, o ktorej budem hovoriť v kapitole 3.1.3.

Programová pamäť

Zariadenia PIC12F675 majú 13-bitový programový čítač, ktorý je schopný adresovať 8K x 14 miesta programovej pamäte. Fyzicky implementovaných je však iba prvých 1K x 14 (adresy 0000h – 03FFh). Sprístupnenie miesta nad týmito hranicami spôsobí adresovanie do prvých 1K x 14. Vektor RESET je na adrese 0000h a vektor prerušenia je na 0004h.



Obr. 3.3: Blokový diagram procesora PIC12F675 [1]

Dátová pamäť

Pamäť pre dáta je rozdelená do 2 bánk, ktoré obsahujú registre pre všeobecné použitie a špeciálne funkčné registre. Špeciálne funkčné registre sú umiestnené v prvých 32 umiestneniach každej banky. Na adresách 20h – 5Fh sú registre pre všeobecné použitie implementované ako statická RAM a sú mapované naprieč oboch bánk. Všetky ostatné RAM sú neimplementované a pri čítaní vracajú nulu.

Bitom RP0 v registry STATUS sa vyberá banka tak, že ak je RP0 = 0, zvolená je banka 0, ak sa RP0 = 1, aktívna je banka 1.

3.1.3 Pamäť EEPROM

(Electrically Erasable Programmable Read-Only Memory) je elektricky zmazateľná pamäť ROM. Je to dátová pamäť. Je možné z nej čítať aj do nej zapisovať v normálnom stave. Táto pamäť nie je priamo mapovaná do adresového priestoru registrov. Namiesto toho sa adresuje nepriamo pomocou 4 špeciálnych funkčných registrov: EECON1, EECON2, EEDATA a EEADR. V registry EEDATA sú uložené 8-bitová dáta určené pre čítanie/zápis a EEADR určuje adresu miesta v EEPROM pamäti, ktoré je sprístupnené. Mikroprocesor PIC12F675 má k dispozícii 128 bytov dátovej pamäte EEPROM s rozsahom adries od 0h po 7Fh.

Čítanie z pamäti EEPROM

Na prečítanie dát z umiestnenia v pamäti EEPROM musí užívateľ napísať adresu do registra EEADR a potom nastaviť riadiaci bit RD v registry EECON1. Dáta sú prístupné hneď v nasledujúcom cykle v registry EEDATA. To znamená, že môžu byť prečítané nasledujúcou inštrukciou. Dáta zostávajú v registry EEDATA až do ďalšieho čítania, alebo kým sú prepísané pri operácii zápisu.

Zápis do pamäti EEPROM

Pre zápis do umiestnenia v pamäti EEPROM musí najskôr užívateľ zapísať adresu do registru EEADR a dáta do registru EEDATA. Potom musí nasledovať špecifická postupnosť inštrukcií pre inicializáciu zápisu do EEPROM.

Zápis sa nezačne ak sekvencia na obr. 3.4 nie je dodržaná. Je doporučené, aby bolo v tomto segmente kódu vypnuté prerušenie. Dôvod je ten, že vo vyžadovanej sekvencii sa musí vykonať určitý presný počet cyklov procesora, ak tento počet neseďí, zápis sa nevykoná.

Navyše bit WREN v registry EECON1 musí byť nastavený, aby bol povolený zápis. Tento mechanizmus zabraňuje náhodným zápisom do dátovej EEPROM kvôli nečakanému vykonávaniu

```
bsf    STATUS,RP0    ;Bank 1
bsf    EECON1,WREN   ;Enable write
bcf    INTCON,GIE    ;Disable INTs
movlw  55h           ;Unlock write
movwf  EECON2        ;
movlw  AAh           ;
movwf  EECON2        ;
bsf    EECON1,WR     ;Start the write
bsf    INTCON,GIE    ;Enable INTs
```

Obr. 3.4: Zápis do pamäte EEPROM [1]

kódu (napr. pri stratených programoch). Užívateľ by mal udržiavať bit WREN nulovaný vždy, okrem zápisu do EEPROM. Tento bit nie je hardwarovo nulovaný.

Ochrana proti nežiadanejmu zápisu

Sú situácie, keď nechceme, aby sa zapísali dáta do pamäte EEPROM. Na ochranu proti nežiadajúcim zápisom sú zabudované viaceré mechanizmy. Pri štarte napájania je nulovaný bit WREN. Obvod Power-Up Timer (kapitola 4.3.3) bráni tiež nežiadajúcemu zápisu.

Inicializačná sekvencia a bit WREN spolu bránia náhodnému zápisu počas:

- brown out (bude popísaný v kapitole 4.3.5)
- výkyvu napätia
- nefunkčnosti softwaru

Táto ochrana je doplnená ešte softwarovou ochranou, ktorá spočíva v tom, že tesne pred vyvolaním hore uvedenej zapisovacej sekvencie skontrolujeme, či prebehla bezchybne komunikácia s PC.

3.1.4 Analógovo-digitálny prevodník

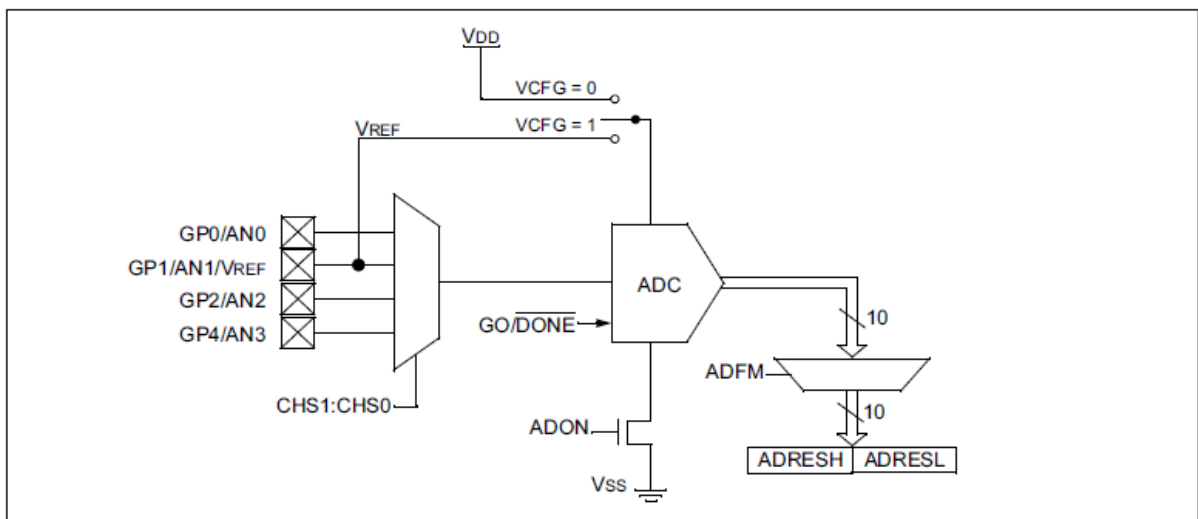
Analógovo-digitálny prevodník umožňuje prevod analógového vstupného signálu na 10-bitovú digitálnu reprezentáciu signálu. Mikrokontrolér PIC12F675 disponuje 4 analógovými vstupmi, z ktorých sa pomocou multiplexora vyberie jeden. Výstup vzorky je pripojený na vstup prevodníka. Prevodník generuje binárny výsledok metódou postupnej aproximácie a výsledok ukladá do 10-bitového registra. Referencia napätia použitá pri konverzii sa dá softwarovo vybrať. Buď je to V_{DD} alebo napätie privedené na pin VREF.

Konfigurácia A/D prevodníka

K dispozícii máme 2 registre, ktoré umožňujú nastaviť funkcionality A/D modulu, a to: 1. ADCON0 a 2. ANSEL. Bitmi ANS3:ANS0 (z registra ANSEL) nastavíme príslušné piny mikrokontroléra ako analógový vstup (takisto aj odpovedajúci TRISIO pin musí byť nastavený ako vstup). Máme k dispozícii celkovo 4 analógové kanály AN0-AN3. Bity CHS1:CHS0 ovládajú, ktorý kanál je pripojený k vzorku a obvodu.

Sú tu 2 možnosti referenčného napätia A/D prevodníka. Buď je použité štandardné referenčné napätie VDD alebo analógový zdroj napätia privedeného na pin VREF. Bit VCFG (z registra ADCON0) určuje referenciu napätia. Ak je nastavený, tak referencia je z pinu VREF, ak nie, referencia je z VDD.

Konverzný cyklus A/D prevodníka trvá 11 periód oscilátora T_{AD} . Zdroj hodinového signálu sa dá vybrať pomocou ADCS bitov (register ANSEL0). Môžeme si vybrať celkovo zo 6 možností použitia externého oscilátora, ale použiť môžeme aj interný oscilátor (pre korektnú konverziu sa musíme uistiť, aby minimálna doba periódy T_{AD} bola 1,6 μ s).



Obr. 3.5: Blokový diagram A/D prevodníka [1]

Konverzia

A/D konverzia je inicializovaná nastavením GO/DONE bitu (register ADCON0). Keď je konverzia ukončená, A/D modul nuluje GO/DONE bit, nastaví príznak ADIF (register PIR1) a generuje prerušenie (ak je povolené). Ak musí byť konverzia prerušená, GO/DONE bit sa môže softwarovo nulovať. Do registrov ADRESH:ADRESL sa nenahrajú čiastočne dokončené A/D vzorky, ale zostanú tam vzorky z minulej konverzie. Po prerušenej konverzii je požadované $2 T_{AD}$ zdržanie predtým ako je ďalšia akcia inicializovaná.

Výsledok A/D konverzie sa ukladá do dvojice registrov ADRESH:ADRESL, a to dvomi možnými spôsobmi: tzv. left-shifted alebo right-shifted. Bit ADFM (register ADCON0) určuje výstupný formát. V prípade že je ADFM nastavený, 2 najvýznamnejšie bity výsledku sú uložené na pozíciách 1 a 0 registra ADRESH a ostatné bity sú uložené v registry ADRESL. V opačnom prípade ($ADFM = 0$), 8 najvýznamnejších bitov výsledku sa uložia do registra ADRESH a v registry ADRESL na pozíciách 8 a 7 sa uložia 2 najmenej významné bity výsledku.

3.2 Posuvný register 74HC164 SMD

Ide o 8 bitový posuvný register so sériovým vstupom a paralelným výstupom dát. Informácie v tejto kapitole sú prevzaté z [3].

Zapojenie vývodov registra:

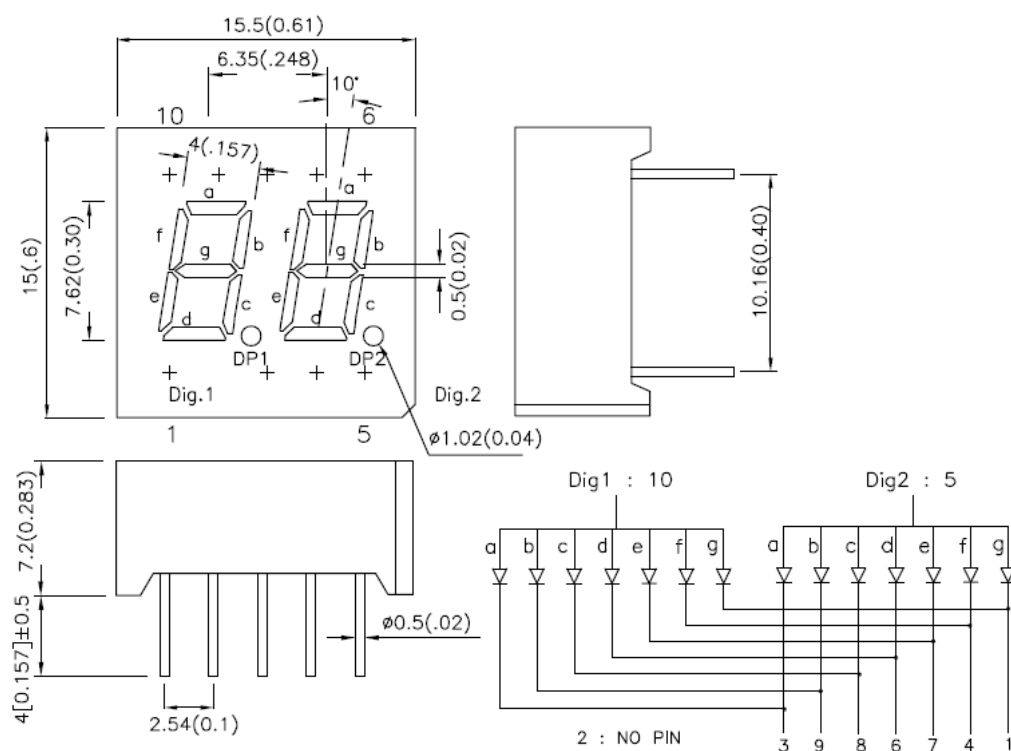
- vývody 1 a 2 - A a B – sériový vstup (vývody sú prepojené)
- vývody 3-6, 9-13 – QA-QH – paralelný výstup
- vývod 9 CLR – reset – nastaví všetky výstupy na nízku hladinu napätia, v našom prípade permanentne vypnutý
- vývod 7 GND – uzemnenie

- vývod 14 VCC – zdroj napätia

Princíp: Na sériový vstup, do ktorého privedieme výstup z vývodu číslo 5 mikroprocesora U1 pošleme dáta, čiže podľa toho aké číslo chceme zobrazit' na displeji, takú postupnosť 8 bitov pošleme. Zároveň pošleme z vývodu číslo 6 mikroprocesora U1 hodinový signál a v každom takte hodinového signálu, tj. s každou nástupnou hranou, pošleme 1 bit. Takto sa naplní celý posuvný register dátami, ktoré pošleme na displej Disp1.

3.3 Digitálny číslicový displej DA03-11EWA

Ide o jednoduchý dvoj číslicový digitálny displej s výškou číslice 0.3 palca. Každá číslica je zložená zo siedmich segmentov.[4]



Obr. 3.6: Rozmery displeja a zapojenie [4]

Všetky vývody displeja sú vstupné:

- vývod 2 NC – nezapojené (not connected)
- vývody 1,3,4,6-9 a-f – jednotlivé segmenty číslice na displeji, rozsvietia sa pri privedení nízkej hladiny napätia
- vývody 5 a 10, Dig2_a a Dig1_a – prepínanie medzi rozsvietením jednej, resp. druhej číslice displeja. Používajú sa k tomu 2 tranzistory Q1 a Q3.

3.4 Teplotný senzor KTY81-120

Teplotný senzor (alebo termistor) je typ rezistora, ktorého odpor je závislý na teplote. V našom výrobku sú použité 2 teplotné senzory KTY81-120, jeden na meranie každej teploty.

Tieto senzory majú pozitívny teplotný koeficient odporu a sú vhodné pre použitie v meraní a riadiacich systémoch. [7]

Technické parametre:

- $R(T_x) - 1 \text{ k}\Omega$ (odpor pri teplote T_x)
- $T_x - 25 \text{ }^\circ\text{C}$ (teplota T_x)
- $T_{\min} - -55 \text{ }^\circ\text{C}$ (minimálna teplota)
- $T_{\max} - 150 \text{ }^\circ\text{C}$ (maximálna teplota)
- $R(T_1)/R(T_x) - 0.49$ (odpor pri teplote T_1 / odpor pri teplote T_x)
- $T_1 - -55 \text{ }^\circ\text{C}$ (teplota T_1)
- $R(T_2)/R(T_x) - 1.696$ (odpor pri teplote T_2 / odpor pri teplote T_x)
- $T_2 - 100 \text{ }^\circ\text{C}$ (teplota T_2)

3.5 Tranzistory BC807-40SMD a BC817-40SMD

V hardwarovom riešení teplomera sú použité 2 typy tranzistorov, BC807-40SMD a BC817-40SMD. Sú to voči sebe komplementárne tranzistory. Tranzistor BC807-40SMD je typu PNP a tranzistor BC817-40SMD je typu NPN. Oba tranzistory sú tranzistory pre všeobecné použitie v prevedení pre povrchovú montáž. Medzi ich hlavné vlastnosti patrí vysoký vstupný prúd (max. 500 mA) a nízke napätie (max. 45V). Oba typy môžu byť použité na všeobecné prepínanie alebo zosilňovanie. V našom výrobku sú použité ako zosilňovače.[5][6]

3.6 Technické obmedzenia

Obmedzenia:

- Z dizajnu a použitého puzdra, ktoré zaručuje požadovanú úroveň IP krytia, vyplýva maximálny priemer plošného spoja 29 mm. Pre zaistenie nízkej ceny je potrebné použiť len 2-vrstvový plošný spoj. Z týchto dôvodov (cena a umiestnenie na malom kruhovom plošnom spoji) sú pri technickom riešení použité tieto hlavné komponenty: 2-miestny 7 segmentový displej (čitateľnosť), mikroprocesor RISC 8-vývodový v puzdre DIP (Dual in-line Package - dvojradové puzdro). Puzdro DIP používame z dôvodov ľahkej manipulácie pri programovaní. Pre zredukovanie počtu pasívnych súčiastok je použité riešenie také, že každý segment je napájaný zvlášť impulzne z kvázi napäťového zdroja (definícia: Napäťový zdroj je zdroj ideálne s nulovým vnútorným odporom. [2]). Veľkosť impulzného prúdu je volená v saturačnej oblasti LED segmentu displeja tak, aby podľa údajov výrobcu Kingbright [4]

nedochádzalo k nedovolenému preťaženiu displeja. Toto riešenie minimalizuje rozdiel v svietivosti napr. číslice 1 a 8.

- Nie je možné použiť rezistorové trimre na kalibráciu zariadenia z dôvodu rozmerov. Z týchto dôvodov je východiskom digitálna kalibrácia, kedy počítač PC automaticky vypočíta z dvoch meraných bodov koeficienty rovnice priamky GAIN a OFFSET a zapíše ich do pamäte mikroprocesora. Pri komunikácii bude použitý zjednodušený protokol MODBUS. V etape merania mikroprocesor vypočítava pomocou týchto koeficientov nameranú hodnotu.
- Pripojenie LED displeja. Korektné technické riešenie by si vyžiadalo použiť rezistory pre každú katódu a nie jeden spoločný do anódy. Toto riešenie sa však priestorovo nedá realizovať. Z tohto dôvodu bola zvolená iná tiež korektná metóda pri ktorej je spoločný impulzný budiaci prúd tak, aby budenie malého množstva segmentov bolo už v oblasti saturácie t.j. bez badateľného vplyvu na jas segmentov. Navyše každý segment je napájaný zo zvláštneho zdroja cca 2 V, takže hodnota odporu spoločného anódového rezistora je porovnateľná s dynamickým odporom LED segmentu.
- Použitie posuvného registra. Tento bol použitý preto, že umiestenie dvoch relatívne malých komponentov poskytovalo väčšie možnosti pre úspešný routing plošného spoja na malej kruhovej ploche. Bude treba avšak zabrániť blikaniu pri posuve dát.
- Použitie stabilizátorov. Použitie viacerých stabilizátorov je motivované rozložením výkonového zaťaženia, tepla a zmenšením vnútorného odporu pri nastavovaní jasu anódovým rezistorom. Prístroj môže byť napájaný jednosmerným napätím od 8 V do 15 V.
- Nie je možné realizovať kompletný interfejs RS232, ale je potrebné použiť kalibračný adaptér. Program mikroprocesora musí automaticky identifikovať či je požadovaný režim kalibrácie alebo prebieha štandardné meranie dvoch teplôt. Je potrebné nie len vyriešiť problém obojsmernej komunikácie s PC, ale navyše automatickú identifikáciu požadovaného režimu činnosti. Zjednodušený adaptér RS232 funguje bezchybne na relatívne nízkych rýchlostiach (9600 baud). Protokol je kontrolovaný CRC sumou. Je pravda, že norma predpisuje rozkmit od $\pm 5V$ do $\pm 25V$, ale komponenty ktoré obvodovo podporujú tento interfejs majú nasledovné parametre:

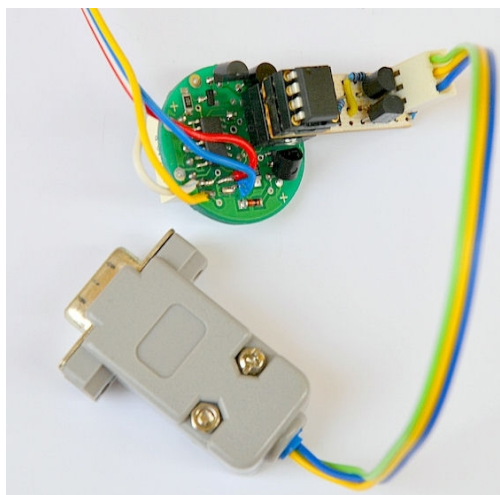
VIT+ Receiver positive-going input threshold voltage	VCC = 5 V, TA = 25 °C	1.7 - 2.4 V
VIT- Receiver negative-going input threshold voltage	VCC = 5 V, TA = 25 °C	0.8 – 1,2 V

3.7 Adaptér pre digitálnu kalibráciu

Pre potreby digitálnej kalibrácie teplotných senzorov je použitý adaptér, ktorého schéma a zapojenie je v prílohe 2. Schéma kalibračného adaptéra.

Pri digitálnej kalibrácii sa používajú 2 vývody mikroprocesora:

- vývod 4 GP3 – je použitý ako vstup sériového prenosu
- vývod 5 GP2 – je použitý ako výstup sériového prenosu



Obr. 3.7: Adaptér pre digitálnu kalibráciu

3.7.1 Sériová komunikácia

Sériový port je komunikačné rozhranie osobných počítačov a inej elektroniky. Sériový port je určený na vzájomnú sériovú komunikáciu dvoch zariadení, čiže dáta sa prenášajú po jednotlivých bitoch postupne za sebou (v sérii) po jednom vodiči.

Prenos údajov

Sériový prenos je jeden z najrozšírenejších spôsobov prenosu. Jednotlivé prvky sú v časovej postupnosti vysielané po jednej prenosovej linke. Pri prijímaní dát je potrebné, aby prijímač bol synchronizovaný s vysielateľom. Prijímač teda musí poznať začiatok a koniec kedy dochádza k zmene signálového stavu – začiatky a konce blokov dát. Na základe týchto údajov prijímač stanoví rozhodujúci okamih pre vyhodnotenie signálového stavu jednotlivého prvku.

Vysielacia a prijímacia strana sa musia na začiatku komunikácie dohodnúť na počte dátových bitov, na paritnom bite a na rýchlosti. Jednotka rýchlosti je Baud, čo je číslo reprezentujúce počet zmien stavu vodiča (linky) za sekundu. Táto hodnota nemusí byť vždy nevyhnutne taká istá ako počet prenesených bitov za sekundu (BPS - Bits per second), ale pri štandardnom prepojení dvoch zariadení

sériovou linkou sú obvykle údaje totožné. Hodnoty bit/s a počtu zmeny stavov linky sa môžu líšiť pri spojeniach prostredníctvom telefónnej linky. Dáta je možné prenášať synchronne a asynchronne.

Synchronný prenos

Pri synchronnom prenose sa na vodičoch nastaví úroveň a informácia sa potvrdí impulzom, alebo zmenou úrovne synchronizačného signálu. Synchronizácia umožňuje rýchlejší prenos dát ako asynchronný, keďže nevyžaduje vysielanie dodatočných znakov vyznačujúcich začiatok a koniec prenášaného rámca. Vysielacia a prijímacia strana musia mať časové základne (generátory taktovacích impulzov), ktoré musia byť synchronizované. Značky sú vysielané ako nepretržitý reťazec bitov, v medzerách sa automaticky vkladajú stavové bity. Začiatky a konce všetkých bitov musia korešpondovať s časovou základňou. Nevýhodou je zložité synchronizovanie, veľkou výhodou však je efektívne využitie kanála a možnosť zabezpečenia prenosu proti chybovosti. Dátové prvky majú rovnakú dĺžku trvania, prenos sa používa pre veľké objemy dát a tam, kde je potrebné počas prenosu zabezpečiť šírku pásma.

Asynchronný prenos

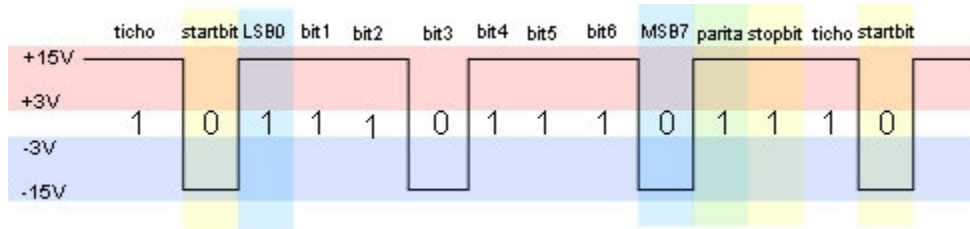
Pri asynchronnom prenose sa dáta prenášajú v sekvenciách (rámcoch) danou rýchlosťou s úvodnou štartovacou sekvenciou vyslaním štart bitu. Akonáhle bol štart bit vyslaný, vysielateľ posiela dátové bity, ktorých môže byť 5, 6, 7, alebo 8, v závislosti od konfiguračnej voľby. Nasleduje paritný bit (nie je povinný). Prenášaný rámec je ukončený stop bitom. Dĺžky trvania jednotlivých bajtov sú rôzne, rôzne sú i medzery medzi dvoma blokmi. Prijímač nerozozná dva nasledujúce dátové bloky. Objem prenášaných dát sa pri tomto type prenosu rozdelí na menšie časti, každá časť sa posiela nezávisle na ostatných, časové intervaly medzi odosielaním jednotlivých bajtov sa môžu líšiť.

RS232

Štandard definuje asynchronnú sériovú komunikáciu pre prenos dát. Poradie prenosu dátových bitov je od najmenej významného bitu (LSB) po bit najvýznamnejší (MSB). Počet dátových bitov je voliteľný, obvykle sa používa 8 bitov. Logický stav 0/1 prenášaných dát je reprezentovaný pomocou dvoch možných úrovní napätia, ktoré sú bipolárne a podľa druhu zariadenia môžu mať hodnoty ± 5 V, ± 10 V, ± 12 V alebo ± 15 V. Najčastejšie sa používa varianta pri ktorej logickej hodnote 1 odpovedá napätie -12 V a logickej hodnote 0 potom $+12$ V. Základné tri vodiče rozhrania (príjem RxD, vysielanie TxD a spoločná zem GND) sú doplnené ešte ďalšími vodičmi slúžiacimi k riadeniu prenosu (vstupy DCD, DSR, CTS, RI, výstupy DTR, RTS). Tie môžu a nemusia byť používané (zapojené).

3.8 Programátor ELNEC SmartProg2

Pre účely programovania mikroprocesora Microchip PIC12F675 a testovania jednotlivých úsekov kódu na reálnom hardware som používal programátor ELNEC SmartProg2. [8]



Obr. 3.8: Prenos jedného slova (bajtu) pomocou RS232

SmartProg2 je univerzálny programátor ELNEC kompatibilný s MS Windows (Windows 98 až Window 7 64-bit). Programátor bol vyvinutý, aby spĺňal požiadavky vývojových pracovísk a servisných centier na univerzálny a ľahko prenosný programátor.

Programátor pracuje bez použitia prídavných modulov s obvody v DIL puzdre do 40 pinov. Programátor je vybavený aj konektorom pre in-circuit sériové programovanie (ISP). SmartProg2 je nielen programátor, ale aj tester RAM pamäti.

SmartProg2 je pripojiteľný k IBM PC kompatibilným typom, od PC desktop po palmtop cez USB port, čo je dôležité pre nové typy počítačov bez vstavaného LPT-portu (napr. notebook).

K dispozícii je WIN 95/98/NT/2000/XP/Vista kompatibilný ovládací program s prehľadným ovládaním a výkonnými funkciami. Voľba programovaného obvodu je podľa výrobcov, podľa tried obvodov, alebo zadaním fragmentu označenia, uvedeného na obvode.



Obr. 3.9: Programátor ELNEC SmartProg2 [8]

Špecifikácia hardware programátora:

- dva D/A prevodníky pre VCCP a VPP s riadenou strmou nábežnej a zostupnej hrany

- rozsah VCCP: 2V až 7V/350mA
- rozsah VPP: 2V až 25V/200mA
- port USB 2.0/1.1 kompatibilný
- autokalibrácia
- schopnosť selftestu

4 Riešenie programu

V tejto kapitole budeme písať o riešení programu pre mikroprocesor, od výberu programovacieho jazyka a vývojového prostredia, cez počiatočnú konfiguráciu mikroprocesora po 3 hlavné časti programu – 1. Test, či sa máme prejsť do stavu digitálnej kalibrácie, alebo do stavu merania, 2. Meranie a zobrazovanie 2 teplôt, 3. Digitálna kalibrácia teplotných senzorov.

4.1 Voľba vývojového prostredia

Zvolili sme prostredie Microchip MPLAB IDE – toto vývojové prostredie je k dispozícii bezplatne, je veľmi jednoduché a prehľadné a pracuje pod operačným systémom Microsoft Windows. Zahŕňa voľne dostupné softwarové komponenty pre vývoj softwaru, emuláciu hardwaru a ladenie. Toto vývojové prostredie podporuje jazyky C a Assembler.

4.2 Výber programovacieho jazyka

Zvolili sme programovací jazyk assembler, pretože umožní efektívnejšie využiť malý pamäťový priestor mikroprocesora a využívať bezplatný simulátor činnosti programu.

4.3 Počiatočná konfigurácia mikrokontroléra

Súčasťou mikrokontroléra sú určité špeciálne obvody, ktoré riešia potreby aplikácii pracujúcich v reálnom čase. Týmto sa odlišujú mikrokontroléry od bežných procesorov. Rodina mikroprocesorov PIC12F675 má prostriedky pre zabezpečenie čo najvyššej spoľahlivosti systému, šetrenie energie v špeciálnych módoch s nízkou spotrebou energie a ochranu kódu. Informácie v tejto kapitole sú čerpané z [1].

4.3.1 Konfigurovateľné obvody

- výber oscilátora
- RESET
- prerušenia
- obvod Watchdog Timer
- režim spánku
- ochrana kódu
- ID kódy
- sériové programovanie v obvode

V riešenej aplikácii potrebujeme nakonfigurovať tieto bity:

- `_INTRC_OSC_NOCLKOUT` – použitie interného oscilátora
- `_WDT_OFF` – vypnutie obvodu Watchdog Timer
- `_PWRTE_ON` – zapnutie obvodu Power-up Timer
- `_CP_ON` – nastavenie ochrany pamäte programu
- `_BODEN_ON` – povolenie detekcie poklesu napájacieho napätia „Brown-out detect“
- `_MCLRE_OFF` – nastavenie funkcie pinu GP3 na pin pre všeobecné použitie (teda nie RESET)

Konfiguračné bity sú v registry CONFIG a programujú sa pomocou príkazu „`__CONFIG`“.

4.3.2 Výber oscilátora

Mikroprocesory rodiny PIC12F675 majú široký výber oscilátorov, na výber je 8 rôznych oscilačných módov. V prípade nášho zadania môžeme vybrať interný oscilátor ako zdroj hodinového signálu. Je to z toho dôvodu, že toto riešenie nevyžaduje žiaden pin mikroprocesora. Interný oscilátor, keď je nakalibrovaný, pracuje na frekvencii 4 MHz. Táto frekvencia vyhovuje pre potreby našej aplikácie.

4.3.3 Power-up Timer

Obvod Power-up Timer zabezpečuje fixné oneskorenie dĺžky 72 ms po resete zariadenia alebo po Brown-out Detect (kapitola 4.3.5). Pracuje s interným RC oscilátorom. Toto oneskorenie umožňuje stabilizáciu napájacieho napätia VDD pred štartom programu. Power-up Timer by mal byť vždy povolený keď je povolený Brown-out Detect (kapitola 4.3.5).

4.3.4 Watchdog Timer

Watchdog timer je voľne bežiaci oscilátor integrovaný na čipe mikroprocesora, ktorý nepotrebuje k svojmu chodu žiadne externé komponenty. Hlavnou úlohou tohto obvodu je zabrániť chybovým stavom chodu mikrokontroléra (napr. zacyklenie) tým, že sa musí pravidelne nulovať. Ak sa toto nestane, Watchdog generuje RESET zariadenia. Tento oscilátor je odlišný od externého oscilátora z pinu CLKIN. To znamená, že Watchdog bude bežať aj keď hodinový signál na pinoch OSC1 a OSC2 zariadenia sa zastaví (napríklad pri vykonávaní inštrukcie SLEEP). V normálnom stave generuje time-out Watchdogu RESET zariadenia. Ak je zariadenie v SLEEP móde, time-out Watchdogu spôsobí, že zariadenie sa zobudí z režimu SLEEP a pokračuje ďalšími operáciami. Watchdog môže byť permanentne zakázaný nulovaním konfiguračného bitu WDTE.

4.3.5 Brown-out Detect

Členovia rodiny PIC12F675 majú zabudovaný obvod Brown-out Detect. Môže byť povolený, alebo zakázaný konfiguračným bitom BODEN. Ak napájacie napätie V_{CC} spadne pod hodnotu V_{BOD} po dobu dlhšiu ako T_{BOD} , Brown-out zresetuje zariadenie. RESET nie je garantovaný, ak V_{CC} klesne pod V_{BOD} na dobu menšiu ako T_{BOD} . Toto zariadenie je dôležité hlavne v miestach s výkyvmi elektrického napätia.

4.4 Inicializácia

Je stav po privedení elektrickej energie do zariadenia, t.j. po resete zariadenia. V tomto stave zisťujeme, či prebieha nejaká komunikácia zo sériového portu počítača PC, v tom prípade zariadenie prejde do stavu digitálnej kalibrácie. V prípade, že po nejakom čase nenastane žiadna komunikácia z PC, zariadenie prejde do režimu merania a zobrazovania 2 teplôt.

Počas tohto stavu zobrazujeme na displeji symbol podržítka striedavo na oboch čísliciach 7 segmentového displeja.

4.4.1 Počiatočné nastavenia

Doladenie oscilátora

V kapitole 4.3.2 bolo uvedené, že pre účely nášho zariadenia použijeme interný oscilátor. Tento je potreba pri štarte zariadenia kalibrovať zápisom do registru OSCCAL. Pre každý mikrokontrolér je táto hodnota iná, a určuje sa pri výrobe. Táto hodnota je zapísaná na konci programovej pamäte (adresa 3FFh). Preto ju musíme odčítať a zapísať ju do registra OSCCAL.

Vstupy a výstupy

V trojstavovom registri TRISIO je nutné nastaviť, ktoré vývody mikroprocesora pre všeobecné použitie (GPIO) budú použité ako vstupné a ktoré ako výstupné. Zápisom hodnoty jedna na príslušný bit určíme pin ako vstup, zápisom nuly bude pin výstupný. Pre naše účely si zdefinujeme symboly, ktoré budeme používať namiesto poradového čísla pinu (obr. 4.1). Popis vstupných a výstupných vývodov je v kapitole 3.1.1.

Weak Pull-up

Každý pin okrem pinu GP3 má k dispozícii tzv. Weak Pull-up, čo je obvod, ktorý pri nestabilných hodnotách napätia na vývodoch, zvýši toto napätia na hodnotu logickej jednotky. Táto funkcia sa nastavuje v registry WPU. V programe sme nastavili všetky dostupné Weak Pull-up bity ako aktívne.

```

; Inputs

TEPL1      equ      GPIO4      ;
TEPL2      equ      GPIO0      ;
PUSH_BUTTON equ      GPIO3      ;
SERIN      equ      GPIO3      ;

; Outputs

SCAN_NUMB  equ      GPIO5      ;
Disp_CLK   equ      GPIO1      ;
Disp_DATA  equ      GPIO2      ;
SEROUT     equ      GPIO2      ;

```

Obr. 4.1: Pomenovanie PINov v programe

Komparátor

Tento obvod nebudeme používať, preto ho kvôli šetreniu elektrickej energie vypneme zápisom „111“ do troch najmenej významných bitov registra CMCON.

4.4.2 Asynchrónny sériový prenos

V stave inicializácie musíme zistiť, či prebieha z počítača PC nejaká komunikácia. V našom prípade ide o sériový asynchrónny prenos pomocou zjednodušeného protokolu RS232.

Použitý mikroprocesor je najjednoduchší typ z celej rodiny mikroprocesorov Microchip, preto nemá špeciálny hardware na zabezpečenie asynchrónneho sériového prenosu. Tento prenos zabezpečíme tým, že vytvoríme 2 podprogramy „recieve“ a „transmit“, ktorých časovanie nastavíme tak, aby splňovalo rýchlosť prenosu 9600 Baud. Presné nastavenie prenosovej rýchlosti urobíme pomocou simulátora nachádzajúceho sa vo vývojovom prostredí a pomocou funkcie stopwatch. Použijeme formát prenosu 9600, N, 2 (9600 Baud, no parity, 2 stop bity). Tento formát prenosu používa kalibračný program na PC.

4.4.3 Podprogram receive

Tento podprogram používame na testovanie, či prebehla sériová komunikácia, a taktiež v režime digitálnej kalibrácie.

Na začiatku testujeme v cykle náš pomenovaný pin SERIN (bit SERIN v registry GPIO), či na ňom bol prítomný štart bit prenosu (nízka hladina napätia). V prípade, že sa nízka hladina neobjaví po nejaký čas, vrátime sa z funkcie a nastavíme príznak ZERO (bit Z v registry STATUS) na jednotku,

čo znamená, že prenos nebol zahájený a zariadenie prejde do stavu merania a zobrazovania 2 teplôt. V prípade, že sa štart bit objaví, postupne prijímame 8 dátových bitov. Dôležité je dodržať prenosovú rýchlosť 9600 Baud. Dátové bity ukladáme do premennej *RcvReg*. Na konci skontrolujeme, či prišiel validný stop bit (napätie v logickej 1), a v prípade, že áno, príznak ZERO nastavíme na nulu. To nám bude v hlavnom programe signalizovať, že prenos prebehol a zariadenie prejde do stavu digitálnej kalibrácie.

4.4.4 Podprogram transmit

Tento podprogram začína nastavením štart bitu – vynulovaním pinu SEROUT (bit SEROUT v registry GPIO). Opäť nemôžeme zabudnúť na rýchlosť prenosu 9600 Baud a musíme vkladať medzi posielanie jednotlivých bitov oneskorenia. Následne odošleme postupne 8 dátových bitov, ktoré sme z akumulátora (W register), kam sa uložia pred zavolaním funkcie transmit, presunuli do premennej *XmtRe*. Nakoniec odošleme 2 stop bity nastavením pinu SEROUT na jednotku a zavolaním funkcie s oneskorením.

Podprogram transmit sa využíva v režime digitálnej kalibrácie.

4.4.5 Odosielanie dát na displej

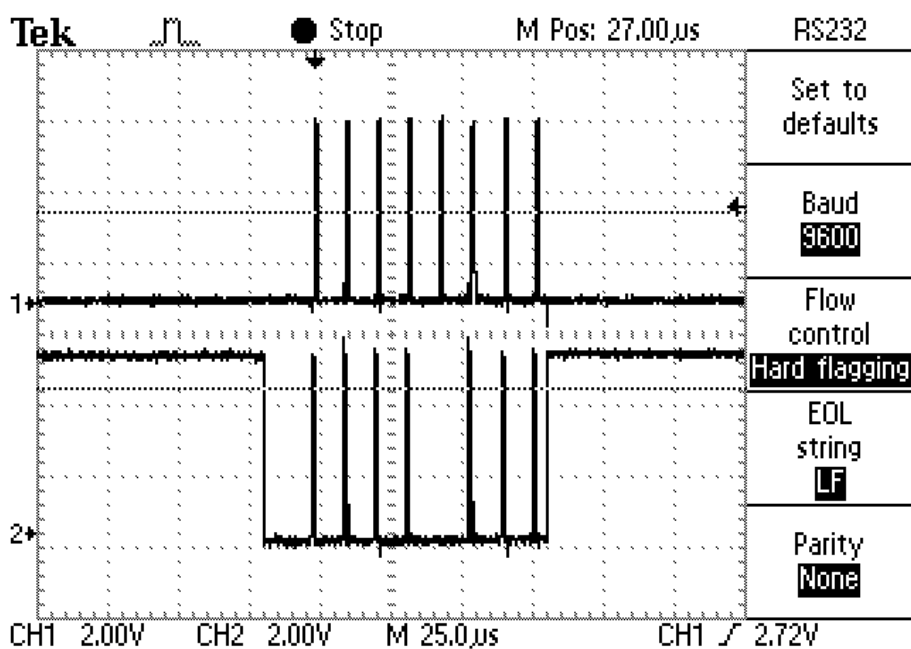
V tejto časti je nutné nejako signalizovať užívateľovi, že sme v stave inicializácie. Toto je vyriešené tak, že na displeji sa bude striedavo na oboch čísliciach zobrazovať znak podtržítka.

Nato sme vytvorili podprogram *Sh7Seg*, ktorý odošle dáta uložené v akumulátore (W register) do posuvného registra (a následne na displej). Odoslanie jedného bitu prebieha nasledovne: Vynulujeme hodinový signál (pin *Disp_CLK*), následne vystavíme na pin *Disp_DATA* dáta (jednotka alebo nula) a napokon povolíme hodinový signál (nastavíme *Disp_CLK* na jednotku). Takto odošleme s využitím operácie rotácie celý bajt dát, ktorý reprezentuje znak, ktorý sa zobrazí na displeji.

Dôležité je, aby sme jednotlivé bity posielali veľmi rýchlo tak, aby na displeji nebolo viditeľné posúvanie dát v posuvnom registri. Toto je zabezpečené tak, že medzi vystavenie platných dát a povolenie hodinového signálu vložíme len jednu inštrukciu *nop* (no operation). Toto zdržanie postačí nato, aby platné dáta boli k dispozícii v okamihu spustenia hodinového signálu. Priebiehy signálov *Disp_CLK* a *Disp_DATA* namerané osciloskopom Techtronix pri odosielaní dát na displej sú na obr. 4.2.

4.5 Meranie a zobrazovanie 2 teplôt

Do tohto stavu sa prístroj dostane po stave inicializácie v prípade, že nedošlo ku komunikácii zo strany počítača PC.



Obr. 4.2: Odosielanie dát, hore – Disp_CLK, dole – Disp_DATA

Na začiatku je potrebné otestovať, či bolo stlačené tlačítko a podľa toho sa rozhodneme, či budeme merať teplotu 1 alebo teplotu 2. Podľa toho, ktorú teplotu meriame, sa určí príslušná adresa kalibračných hodnôt offset a zosilnenie v pamäti EEPROM a A/D prevodník sa prepne na príslušný merací senzor. Následne nastavíme A/D prevodník a odčítame hodnotu výsledku prevodu. Nameranú hodnotu, ktorá je celočíselná, prevedieme na hodnotu v plávajúcej rádovej čiarky. Výslednú hodnotu teploty vypočítame pomocou vzťahu (1), kde hodnoty k (zosilnenie) a q (offset) prečítame z príslušnej adresy v pamäti EEPROM a hodnota x bude výstup A/D prevodníku.

$$y = kx + q \quad (1)$$

Vypočítanú hodnotu je nutné previesť naspäť na celočíselnú hodnotu, keďže displej je schopný zobrazit iba 2 číslice. Výsledné dvojciferné číslo musíme rozdeliť na 2 samostatné číslice a každú z číslic zakódovať tak, aby ju bolo možné zobrazit na 7 segmentovom displeji. Potom zakódovanú číslicu pošleme sériovo do posuvného registra a odtiaľ paralelne na displej. Musíme pritom určiť, či zobrazujeme číslicu 1 alebo 2.

4.5.1 Knižnice pre výpočty s pohyblivou rádovou čiarkou

Pretože výpočty s pohyblivou rádovou čiarkou nie sú pre tento mikroprocesor celkom triviálne, rozhodli sme sa použiť už pripravené knižnice, ktoré obsahujú funkcie pre prácu s číslami v pohyblivej rádovej čiarkke.

math16.inc

V tomto súbore sa nachádzajú definície a deklarácie symbolov a premenných použitých v nasledujúcich knižniciach.

flo2432.inc

Je knižnica pre konverziu čísla z 24 bitového celočíselného formátu na 32 bitové číslo v pohyblivej rádovej čiarkke.

INT3224+.a16

Je knižnica pre konverziu čísla z 32 bitového čísla v plávajúcej rádovej čiarkke na 24 bitové celé číslo.

fp32r.a16

Súčasťou tohto súboru sú funkcie pre prácu s číslami v pohyblivej rádovej čiarkke. Sú to:

- normalizácia čísla (viď. kapitola 4.5.2)
- násobenie
- sčítanie

Všetky použité knižnice sú voľne k dispozícii na stránkach spoločnosti Microchip.

4.5.2 Normalizácia čísla

Formát čísla v pohyblivej rádovej čiarkke s ktorými pracujú mikroprocesory danej rodiny Microchip je špecifický formát pre tieto mikroprocesory a odlišuje sa od tradičnej reprezentácie čísel v pohyblivej rádovej čiarkke IEEE 754 s ktorým pracujú procesory firmy Intel. Preto pri prenose čísel v pohyblivej rádovej čiarkke bude nutná konverzia z jedného formátu na druhý. Táto normalizačná funkcia je súčasťou použitých knižníc pre výpočty s pohyblivou rádovou čiarkkou.

4.5.3 Nastavenie A/D prevodníka

Pred zahájením A/D prevodu je nutné nastaviť A/D prevodník na požadované parametre. (podrobnosti v kapitole 3.1.4)

Najskôr nastavíme register ANSEL, zapísaním do neho vyberieme konverznú rýchlosť FOSC/32, teda frekvencia interného oscilátora 32x zmenšená, a podľa schémy teplomera nastavíme analógový vstup prevodníka na vstup AN3.

Nastavenia v registry ADCON0: Výstupné dáta zarovnané vpravo, referencia napätia V_{DD} , analógový kanál AN3 a uvedieme prevodník do činnosti.

Pred zahájením prevodu musíme vymazať výstupné registre ADRESL a ADRESH.

4.5.4 Merací filter

Pre čo najpresnejšie výsledky meranej teploty sme použili tzv. merací filter. Ide o to, že A/D prevod spustíme v cykle 10 krát po sebe, a všetkých 10 hodnôt sčítame do súčtu. Výsledná hodnota potom bude 10-násobkom výsledku prevodu a týmto sa vyrovnajú rôzne dočasné výkyvy pri A/D prevode.

4.5.5 Zosilnenie a ofset

Po A/D prevode skonvertujeme výslednú celočíselnú hodnotu na hodnotu v plávajúcej rádovej čiarky príslušnou funkciou z knižnice. Následne vyčítame z pamäte EEPROM hodnotu zosilnenia (čítanie a zápis do pamäte EEPROM je popísaný v kapitole 3.1.3) a ofsetu. Dôležité je prečítať hodnoty pre správny teplotný senzor (podľa toho, či tlačítko je stlačené alebo nie). Ofset pre senzor 1 začína v EEPROM na adrese 00h a zosilnenie pre tento senzor na adrese 04h. Ofset pre senzor 2 začína na adrese 20h v EEPROM a zosilnenie na adrese 24h.

Podľa vzťahu (1) a s použitím funkcií pre prácu s pohyblivou rádovou čiarkou zohľadníme ofset a zosilnenie a výsledné číslo v pohyblivej rádovej čiarky prevedieme naspäť na celé číslo.

4.5.6 Kontrola výsledku

Po vypočítaní výsledku teploty skontrolujeme, či je teplota v požadovanom rozsahu. Ak vyšla hodnota vyššia ako 99, alebo menšia ako 0, toto číslo nie je možné na displeji zobrazit', a preto namiesto neho vypíšeme chybové hlásenie v tvare „__“.

4.5.7 Prevod hodnoty teploty na desiatky a jednotky

Ak nám vyšla hodnota teploty v požadovanom rozsahu, je nutné previesť číslo na cifry – desiatky a jednotky, aby sme mohli každú cifru zobrazit' na prislúchajúcej číslici displeja.

Na prevod slúži funkcia *Conv_DESJEDN*. Pracuje tak, že postupne odčítava z výsledku hodnotu 10, pričom počítame koľkokrát sa odčítalo. V prípade, že výsledok je už menší ako 0 (príznak Carry je aktívny), ukončíme cyklus. Počet desiatok sa bude rovnať počtu odčítaní,

zníženému o 1. A počet jednotiek získame tak, že k výsledku pripočítame 10 (predtým sme odpočítali a dostali záporný výsledok).

4.5.8 Získanie kódu číslice

Aby sme mohli zobrazit' číslicu na displej, musíme odoslať kódové označenie, kde je dané, ktoré segmenty displeja sa majú vysvietiť a ktoré naopak nie. Keďže číslica displeja má 7 segmentov, pre každý segment odosieme jeden bit.

Získanie kódu číslice je implementované vo funkcii *GET_NUMBER*. Vstupom funkcie je číslica, ktorú chceme poslať na displej, a výstupom je 8 bitové kódové označenie číslice (popis displeja a jeho segmentov je v kapitole 3.3).

Toto som riešil alternatívou k príkazu *switch* v jazyku C. Programový čítač PCL zvýšime o také číslo, akú číslicu máme zobrazit'. Takto program skočí na jeden z príkazov *retlw*, a podľa toho o koľko skočí programový čítač, kód takej číslice sa vráti ako výstup funkcie.

4.6 Digitálna kalibrácia

Nakoľko pre požadovanú presnosť teplomera môžeme uvažovať použitý senzor za lineárny, digitálna kalibrácia spočíva v popise meracej sústavy (môstik tvorený senzorom, rezistorom a A/D prevodníkom) rovnicou priamky v dvojrozmernom priestore, ktorá je popísaná lineárnou rovnicou v tvare:

$$y = kx + q \tag{2}$$

Kde hodnota q je ofset a hodnota k je zosilnenie

Digitálna kalibrácia sa vykonáva pomocou prepojenia s počítačom PC, na ktorom pracuje program, ktorý pomocou dvoch párov nameraných hodnôt x_1, y_1, x_2, y_2 vypočíta koeficienty k a q riešením sústavy dvoch lineárnych rovníc o dvoch neznámych:

$$y_1 = kx_1 + q \tag{3}$$

$$y_2 = kx_2 + q \tag{4}$$

x_1 – hodnota prevodníka v kalibračnom bode 1

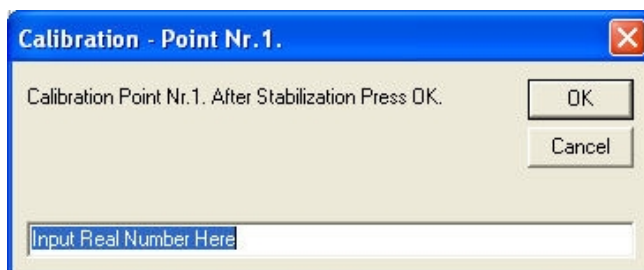
y_1 – hodnota teploty nameraná etalónovým teplomerom v bode 1

x_2 – hodnota prevodníka v kalibračnom bode 2

y_2 – hodnota teploty nameraná etalónovým teplomerom v bode 2

Z uvedených hodnôt vypočíta program na PC koeficienty k a q riešením sústavy dvoch lineárnych rovníc o dvoch neznámych.

Namerané hodnoty etalonovým kalibračným teplomerom y_1, y_2 zapíše operátor, hodnoty A/D prevodníka x_1, x_2 sú prenesené do PC pomocou zjednodušeného rozhrania RS232.



Obr. 4.3: Kalibračný program – zadanie teploty

Po vypočítaní koeficientov k, q sa tieto prenesú pomocou zjednodušeného rozhrania RS232 do riadiaceho mikroprocesora a zapíšu do pamäti EEPROM. Nameraná kalibrovaná hodnota sa vypočíta v mikroprocesore pomocou vzťahu (2).

Prenos dát medzi mikroprocesorom a počítačom PC sa vykonáva s použitím redukovaného protokolu MODBUS. Komunikácia funguje na princípe predávania dátových správ medzi klientom PC a serverom - teplomer (master a slave). Nakoľko kalibračný program vytvorený na PC je univerzálny a používaný aj na iné účely, je potreba pri digitálnej kalibrácii teplomera dodržať dole uvedený formát príkazov pre čítanie a zápis.

Z uvedeného protokolu používame príkaz pre vyčítanie údajov a príkaz pre zápis údajov. Príkazy inicializuje klient PC. Server (mikroprocesor) odpovedá.

4.6.1 Čítanie

Táto operácia je potrebná kvôli tomu, aby sme mohli v kalibračnom programe na počítači PC zobrazovať hodnoty teplôt a aby program získal informácie pre výpočet nových kalibračných hodnôt. Klient – počítač PC posielá požiadavku serveru – mikroprocesoru. V požiadavku sa nachádza, či chce zistiť hodnoty pre teplotný senzor 1 alebo 2. Všetky požiadavky aj odpovede sú chránené súčtom CRC. Mikroprocesor prepne vstup A/D prevodníka na príslušný senzor podľa adresy ADR a následne pošle počítaču odpoveď s kalibračnými údajmi žiadaného teplotného senzora a takisto hodnotu A/D prevodníka, aby bol program schopný vypočítať výslednú teplotu.

- **požiadavok od klienta PC**

ADR	03H	00H	03H	00H	03H	CRC
-----	-----	-----	-----	-----	-----	-----

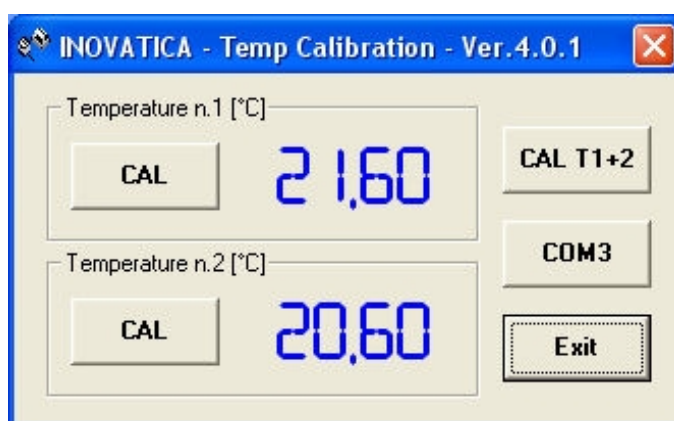
– 03H – číslo funkcie čítanie

- ADR = 0 -> teplota 1
- ADR = 1 -> teplota 2
- CRC – vypočítaná suma CRC-16

● **odpoveď s údajmi pre výpočet teploty**

ADR	03H	0AH	NH1	NH2	OFS1	OFS2	OFS3	OFS4	ZOS1	ZOS2	ZOS3	ZOS4	CRC
-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	-----

- 03H – číslo funkcie čítanie
- ADR = 0 -> teplota 1
- ADR = 1 -> teplota 2
- NH1, NH2 – hodnota AD prevodníka mikroprocesora (koeficienty x1, x2)
- OFSi – ofset teplomeru reálne číslo
- ZOSi – zosilnenie teplomeru reálne číslo
- CRC – vypočítaná suma CRC-16



Obr. 4.4: Kalibračný program – zobrazenie teplôt

4.6.2 Zápis kalibračných hodnôt do pamäte EEPROM

Pomocou tejto operácie zapíše program na PC vypočítané nové kalibračné hodnoty pre teplotné senzory. Klient – PC odošle požiadavok s rozlíšením, či ide o senzor 1 alebo 2 a pošle kalibračné údaje zosilnenie a ofset pre teplotný senzor. Mikroprocesor zapíše hodnoty ofsetu a zosilnenia do pamäte EEPROM na adresu podľa kalibrovanej teploty a odošle potvrdzujúcu odpoveď. Všetky požiadavky a odpovede sú chránené CRC sumou.

● požiadavok od klienta PC

ADR	10	00	04	00	02	08	OF1	OF2	OF3	OF4	ZO1	ZO2	ZO3	ZO4	CRC
-----	----	----	----	----	----	----	-----	-----	-----	-----	-----	-----	-----	-----	-----

- 10 – číslo funkcie zápis
- ADR = 0 -> teplota 1
- ADR = 1 -> teplota 2
- OFi – ofset
- ZOi – zosilnenie
- DATA – zapisované údaje
- CRC – vypočítaná suma CRC-16

● odpoveď po zapísaní

ADR	10	00	04	00	02	CRC
-----	----	----	----	----	----	-----

- 10 – číslo funkcie zápis
- ADR = 0 -> teplota 1
- ADR = 1 -> teplota 2
- CRC – vypočítaná suma CRC-16

4.6.3 CRC

Kontrola cyklickým kódom alebo cyklická kontrola (ang. cyclic redundancy check, skrátene CRC) je druh kontrolného súčtu používaného na kontrolu správnosti prenášaných údajov v telekomunikačnej technike a počítačových sieťach, ako aj uložených údajov na pamäťových médiách ako je napríklad pevný disk.

Na základe jednotlivých bitov sa vypočítava zabezpečovací údaj. Ten sa na konci celého bloku porovná so zabezpečovacím údajom, ktorý podľa rovnakých pravidiel vypočítal odosielateľ a pripojil k prenášanému bloku dát. Ak sa tieto dva údaje zhodujú, dá sa prenesený blok s vysokou pravdepodobnosťou predpokladať za správny. K výpočtu zabezpečovacieho údaju nám postačí jednoduchý posuvný register, umožňujúci operáciu EX-OR s pevne danou maskou. Hodnota tejto masky je jednoznačne určená tzv. generujúcim polynómom (generating polynomial), na ktorom musia byť príjemca i odosielateľ vopred dohodnutí. Použiteľných polynómov týchto tvarov je viacej. V sieťovej komunikácii sa najčastejšie používa polynóm $x^{16} + x^{12} + x^5 + 1$, doporučený organizáciou CCITT.

Kontrola cyklickým kódom, ako každý kontrolný súčet, mierne zväčšuje redundanciu správy, ale zvyšuje jej spoľahlivosť.

4.6.4 Generovanie CRC

Pre generovanie CRC slúži podprogram *GenCRC*. Tento má vstup v premennej *Messag*, kde pred volaním uložíme byte, z ktorého sa má CRC vypočítať. Výstup je do premenných *CRCaLo* a *CRCaHi*. Generovanie CRC prebieha tak, že pri prvom prijatom byte dát spustíme funkciu *GenCRC*. Keď prijmem ďalší byte, znova zavoláme funkciu *GenCRC* s prijatým bytom v premennej *Messag*. CRC suma sa bude počítat' už aj s predchádzajúcim výsledkom, takže po prijatí všetkých bytov bude vo výstupných premenných výsledná suma CRC vypočítaná zo všetkých prijatých dát.

4.6.5 Implementácia digitálnej kalibrácie

Keďže už máme prijatý jeden byte, ktorý sme prijali pri testovaní, či prebieha prenos medzi počítačom PC a mikroprocesorom, tento byte uložíme do premennej *Frame1*. Ostatné byty budeme ukladať do premenných *FrameX*, kde X bude číslo bytu. Postupne voláme podprogram *receive* a prijímame a ukladáme ďalšie byty. Prijímanie končí vtedy, keď už nepríde žiaden validný byte. Na adresovanie premenných *FrameX* používame nepriame adresovanie pomocou registrov INDF a FSR. Do registra FSR (File Select Register) vložíme adresu *Frame2* a následne prijatý byte ukladáme do registra INDF, ktorý spôsobí nepriame adresovanie a prijatý byte sa uloží na adresu, ktorá je vo FSR.

Po prijatí všetkých dát skontrolujeme, či sedí suma CRC. Ak nie, skočíme na začiatok programu. V opačnom prípade najskôr z *Frame2* zistíme, či ide o požiadavku na čítanie alebo zápis a následne z *Frame1* zistíme, či sa jedná o teplotný senzor 1 alebo 2. Následne odošleme odpoveď.

V rámci čítania odošleme (podprogramom *transmit*) prvé 2 byty, ktoré budú rovnaké, ako prvé 2 prijaté byty. Následne nastavíme vstup A/D prevodníka na teplotný senzor podľa prijatého bytu *Frame1* a zahájime prevod. Výsledok prevodu odošleme ako byty 3 a 4. Nasledujúcich 8 bytov sú hodnoty offset a zosilnenie, ktoré prečítame z pamäte EEPROM a odošleme. Nakoniec vypočítame CRC sumu a odošleme. Po úspešnom odoslaní skočíme na začiatok programu.

V rámci zápisu najskôr zapíšeme do pamäti EEPROM údaje o zosilnení a offsete, ktoré vypočítal a odoslal kalibračný program na počítači PC. Pri čítaní údajov z pamäte EEPROM opäť využijeme nepriame adresovanie s dvojicou registrov INDF a FSR. Ako odpoveď pošleme prvé 2 byty zhodné s prijatými, a následne konštanty, a nakoniec sumu CRC. Po úspešnom odoslaní skočíme na začiatok programu.

5 Finálny výrobok

V tejto krátkej kapitole si popíšeme s obrázkami finálny výrobok, a povieme si o jeho zavedení do praxe.



Obr. 5.1: Finálny teplomer

Na obrázku 5.1 je zobrazený finálny teplomer, ktorý je využívaný v praxi. Pomocou neho sa dajú merať 2 teploty – vody vo vani, a vody pritekajúcej. Zákazník medzi týmito teplotami prepína pomocou tlačítka – keď je stlačené, meria sa teplota pritekajúcej vody a keď je pustené, teplota vody vo vani. Na obrázku 5.2 je zobrazený teplomer s kabelážou a dvoma teplotnými senzormi.

Testovanie a kontrola implementovaných funkcií prebehla aj vo firme aj na zákazníkoch, ktorý výrobok hodnotia veľmi kladne a sú s ním spokojný.



Obr. 5.2: Teplomer s kabelážou a senzormi

6 Záver

Mojou úlohou bolo vytvoriť program pre mikroprocesor typu PIC, ktorý bude striedavo zobrazovať na displeji dve teploty vody v masážnom zariadení. Vychádzalo sa z konkurenčného riešenia firmy Koller, ktorá je na trhu už dlhší čas s týmto výrobkom. Preto náš výrobok musel byť lacnejší a disponovať rovnakými funkciami ako konkurencia.

Hardware – plošný spoj spolu s kalibračným adaptérom a aj obslužný software vyrobila a dala mi k dispozícii firma Inovatica, s.r.o. Ja som pracoval na vývoji softwaru. Najprv som si musel podrobne naštudovať použitý hardware – navrhnutý plošný spoj a zoznámiť sa s použitým mikroprocesorom firmy Microchip. Veľmi mi v tomto pomohol aj predmet Mikroprocesorové a vestavěné systémy, kde som sa naučil základné aj pokročilé veci z tejto oblasti. Veľmi kladne môžem hodnotiť aj spoluprácu s firmou Inovatica, s.r.o., kde mi vždy poradili, keď som mal nejaký problém, keďže s niektorými hardwarovými prostriedkami sa nestretávam príliš často.

Vývoj softwaru potom prebiehal bez väčších problémov, keď som si na začiatku spísal požiadavky, ktoré sa od softwaru vyžadujú a pripravil si kostru programu. Túto kostru som postupne doplňoval a po jednom som implementoval jednotlivé časti kódu. Kontrolu funkčnosti som robil na simulátore a aj s použitým hardware.

Za môj hlavný prínos považujem niektoré netradičné softwarové riešenia, zhodnotenie mojej práce od firmy Inovatica, s.r.o. môžete nájsť v prílohe.

Myslím si, že tento výrobok si na trhu našiel svoje miesto a používa ho veľa spokojných zákazníkov. Do budúca si nemyslím, že by sa mal nejako vylepšovať, alebo meniť. Tento výrobok splnil svoj účel a do budúca by bolo vhodné sa zamerať na niečo iné, na niečo, čo na trhu chýba, alebo by sa dalo spraviť lepšie.

Literatura

- [1] Microchip Technology: PIC12F629/675, Data Sheet [online].
<http://ww1.microchip.com/downloads/en/DeviceDoc/41190G.pdf>, 2010-04-30 [cit. 2011-05-14].
- [2] Pánek, D.: Teoretické elektrotechnika pro FAV [online].
<http://home.zcu.cz/~panek50/TEA/Teoretick%c3%9f%20elektrotechnika%20pro%20FAV.ppt>, 2005-09-23 [cit. 2011-05-14].
- [3] Koninklijke Philips Electronics N.V.: 74HC164; 74HCT164, Data Sheet [online].
<http://pdf1.alldatasheet.com/datasheet-pdf/view/15548/PHILIPS/74HC164.html>, 2005-04-04 [cit. 2011-05-14].
- [4] Kingbright Corporation: DA03-11EWA, Data Sheet [online].
<http://pdf1.alldatasheet.com/datasheet-pdf/view/111577/KINGBRIGHT/DA03-11EWA.html>, 2005-04-23 [cit. 2011-05-14].
- [5] Phillips Semiconductors: BC807, Data Sheet [online].
http://www.gme.sk/_dokumentace/dokumenty/912/912-003/dsh.912-003.1.pdf, 1999-04-08 [cit. 2011-05-14].
- [6] Phillips Semiconductors: BC817, Data Sheet [online].
http://www.gme.cz/_dokumentace/dokumenty/912/912-032/dsh.912-032.1.pdf, 1999-06-01 [cit. 2011-05-14].
- [7] Phillips Semiconductors: KTY81-1 series, Data Sheet [online].
http://www.gme.sk/_dokumentace/dokumenty/530/530-015/dsh.530-015.1.pdf, 2000-08-25 [cit. 2011-05-15].
- [8] Elnec s.r.o.: SmartProg2 [online].
<http://www.elnec.com/sk/products/universal-programmers/smartprog2/#feat>, 2011-04-06 [cit. 2011-05-14].

Seznam příloh

Příloha 1. Schéma navrhnutého templomera

Příloha 2. Schéma kalibračního adaptéra

Příloha 3. Stanovisko k řešení bakalářské práce od firmy Inovatica s.r.o.

Příloha 4. CD so zdrojovými textami