

Výukový modul pro předmět mikropočítače: jednoduchá tepelná soustava

Educational module for microcontrollers course: simple heating
plant

Zdeněk Babík



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav automatizace a řídicí techniky
akademický rok: 2006/2007

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Zdeněk BABÍK
Studijní program: B 3902 Inženýrská informatika
Studijní obor: Automatické řízení a informatika

Téma práce: Výukový modul pro předmět mikropočítače:
jednoduchá tepelná soustava

Zásady pro vypracování:

1. Seznamte se s vývojovým kitem M68EVB908GB60, který se používá pro výuku programování mikropočítačů na naší fakultě.
2. Navrhněte strukturu a zapojení modulu představujícího jednoduchou tepelnou soustavu a připojitelného na tento kit.
3. Realizujte funkční prototyp navrženého modulu a ověřte jeho funkci na vývojovém kitu.
4. Vytvořte podpůrné programové vybavení pro obsluhu modulu v jazyce symbolických adres.

Rozsah práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. MC9S08GB/GT Data Sheet, Rev.2.3. Freescale Semiconductor, 2004
2. HCS08 Family Reference Manual, Rev.1. Freescale Semiconductor, 2003
3. CPU08 Central Processor Unit Reference manual. Motorola, 2001
4. VAŠEK V., VAŠEK L.: Programování mikropočítačů. FT VUT v Brně, Zlín 1989
5. VÁŇA V.: Začínáme s mikrokontroléry Motorola HC08 Nitron. BEN — technická literatura, Praha 2003
6. LADMAN J.: Elektronické konstrukce pro začátečníky. BEN – technická literatura 2002.

Vedoucí bakalářské práce:

Ing. Jan Dolinay

Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce:

13. února 2007

Termín odevzdání bakalářské práce:

25. května 2007

Ve Zlíně dne 13. února 2007


prof. Ing. Vladimír Vašek, CSc.
děkan




prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Cílem této bakalářské práce je vytvořit funkční model tepelné soustavy s dvoupolohovým ovládáním topení a měřením teploty pomocí mikropočítače, který bude sloužit jako výuková pomůcka při programování mikropočítačů HCS08. V první části je obecně popsána problematika mikropočítačů HCS08, základů regulace a způsobů měření teploty. Ve druhé části je popsána konstrukce vlastního modelu tepelné soustavy a obslužné programové vybavení. V závěrečné části jsou provedena některá základní měření na modelu soustavy včetně identifikace soustavy a určení $G(s)$ přenosu.

Klíčová slova: Mikropočítač, HCS08, tepelná soustava, regulace, SMT 160-30

ABSTRACT

The main aim of this bachelor thesis is to create a functional model of a heat system with two-state heater control and with temperature measured by the microcontroller, which will be used as an educational instrument at programming of microcomputers HCS08. The first part describes the programming of HCS08 microcontrollers and basics of the temperature control and measurement. In the second part is described the module itself and the program library for its control. The last part presents some results of measurements made on the model including identification and transfer function evaluation.

Keywords: Microcontroller, HCS08, heat system, regulation, SMT 160-30

Na tomto místě bych rád poděkoval vedoucímu své bakalářské práce panu Ing. Janu Dolinayovi za rady, podněty a odborné vedení, které mi poskytoval během období vzniku této práce. Rád bych také poděkoval panu Ing. Petru Dostálkovi za množství odborných rad, které mi umožnily realizovat praktickou část tohoto projektu.

Nakonec bych chtěl poděkovat také rodinným příslušníkům za podporu, kterou mi poskytl, a dále za trpělivost, se kterou byli ochotni mě poslouchat.

Bez jejich přispění by tato práce nikdy nemohla vzniknout.

Motto:

„Je lepší rozsvítit, byť jen malou svíčkou, než proklínat temnotu.“

Konfucius (?551 př. n. l. - ?479 př. n. l.)

Prohlašuji, že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků, je-li to uvolněno na základě licenční smlouvy, budu uveden jako spoluautor.

Ve Zlíně, 25.5.2007

.....

Podpis diplomanta

OBSAH

ÚVOD	8
I TEORETICKÁ ČÁST	10
1 MIKROPOČÍTAČ MOTOROLA HC(S)08	11
1.1 VÝVOJ MIKROPROCESORŮ A MIKROPOČÍTAČŮ.....	11
1.2 MONOLITICKÉ MIKROPOČÍTAČE	12
1.2.1 Struktura monolitického mikropočítače.....	12
1.2.2 Rozdělení monolitických mikropočítačů	13
1.3 VÝVOJOVÝ KIT M68EVB908GB60	14
1.3.1 Seznam základních parametrů vývojového kitu M68EVB908GB60	15
1.3.2 Specifikace	16
1.3.3 Napájení	16
1.3.4 Svorkovnice TB1	17
1.4 POPIS MIKROPOČÍTAČŮ M9S08GB60.....	17
1.4.1 Struktura mikropočítačů řady MC9S08GB/GT	17
1.4.2 Mikroprocesor	18
1.4.3 Sběrnice.....	20
1.4.4 Paměť	20
1.5 PERIFERIE MIKROPROCESORU HC(S)08.....	23
1.5.1 Časovač	23
1.5.2 A/D převodník.....	24
1.6 KOMUNIKAČNÍ ROZHRANÍ	26
1.6.1 Paralelní komunikace	26
1.6.2 Sériová komunikace	27
1.7 PŘERUŠENÍ	32
1.8 REŽIMY PROVOZU SE SNÍŽENÝM PŘÍKONEM.....	33
1.8.1 Režim WAIT	33
1.8.2 Režim STOP.....	33
1.9 JAZYKY SYMBOLICKÝCH ADRES	34
1.9.1 Programování v assembleru	35
2 REGULACE A REGULAČNÍ TECHNIKA	36
2.1 ZÁKLADNÍ POJMY A DEFINICE Z REGULAČNÍ TECHNIKY.....	36
2.2 AUTOMATICKÁ REGULACE	37
2.2.1 Rozdělení a druhy regulátorů	37
2.3 NESPOJITÁ REGULACE	38
2.3.1 Dvoupolohový regulátor	38
2.4 SPOJITÁ REGULACE	39
2.4.1 Proporcionální regulátor	40
2.4.2 Integrační regulátor	40
2.4.3 Derivační regulátor.....	41
2.4.4 Sdružené regulátory	41

3	MĚŘENÍ TEPLoty	43
3.1	ROZDĚLENÍ SENZORŮ TEPLoty	43
3.2	ROZDĚLENÍ ELEKTRICKÝCH DOTYKOVÝCH SENZORŮ TEPLoty	44
3.2.1	Odporové kovové senzory teploty	44
3.2.2	Polovodičové odporové senzory teploty	45
3.2.3	Monokrystalické senzory	46
3.2.4	Teplotní čidla s digitálním výstupem	47
II	PRAKTICKÁ ČÁST	50
4	MODEL TEPELNÉ SOUSTAVY ŘÍZENÉ MIKROPOČÍTAČEM	51
4.1	PRINCIP ČINNOSTI MODELU TEPELNÉ SOUSTAVY	52
4.2	POPIS HARDWARE VYTVOŘENÉHO MODULU	52
4.2.1	Napájení	53
4.2.2	Výkonová část modelu	53
4.2.3	Teplotní čidlo SMT 160	60
5	PROGRAMOVÉ VYBAVENÍ	61
5.1	VYHODNOCOVÁNÍ SIGNÁLU POMOCÍ ČASOVAČE TPM2	63
5.2	VYHODNOCOVÁNÍ SIGNÁLU POMOCÍ VZORKOVÁNÍ	67
5.3	KNIHOVNA FUNKCÍ TOP_GB60.....	68
6	PRAKTICKÁ MĚŘENÍ NA MODELU TEPELNÉ SOUSTAVY	73
6.1	MAXIMÁLNÍ TEPLOTNÍ ROZSAH	73
6.2	JEDNODUCHÝ REGULAČNÍ POCHOD	74
6.3	IDENTIFIKACE SOUSTAVY	75
6.3.1	Statická charakteristika	75
6.3.2	Identifikace soustavy - Strejcova metoda	77
6.4	PROTOTYP MODELU TEPELNÉ SOUSTAVY ŘÍZENÉ MIKROPOČÍTAČEM.....	79
	ZÁVĚR	81
	ZÁVĚR V ANGLIČTINĚ	82
	SEZNAM POUŽITÉ LITERATURY	83
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	85
	SEZNAM OBRÁZKŮ	87
	SEZNAM TABULEK	89
	SEZNAM PŘÍLOH	90

ÚVOD

Rychlý rozvoj v oblasti informatiky a polovodičových technologií ve druhé polovině 20. století a stále rostoucí požadavky ve všech oblastech automatizace a regulační techniky zapříčinily, že se dva původně zcela samostatné obory, informatika a automatizace, stále více integrují. Dnes je prakticky pravidlem, že průmyslový podnik, banka, státní úřad či domácnost disponují systémy, v nichž se spojují prvky informatiky a automatizace. Tyto systémy jsou velmi rozmanité. Může se jednat o naváděcí systém v balistické střele, robotické rameno v automobilce, inteligentní adaptivní regulátor v elektrárně, obráběcí stroj CNC ve výrobním závodě, systém individuální regulace teploty (IRC) v kanceláři, výrobní hale či obytné místnosti, nebo „jen“ o automatickou pračku nebo mikrovlnnou troubu, které jsou neodmyslitelnou součástí většiny domácností. Společným rysem většiny těchto zařízení je, že jejich mozkiem jsou součástky označované jako mikropočítače nebo mikrokontrolery. Důvodem jejich použití je snaha zvýšit možnosti těchto zařízení, minimalizovat vliv člověka na jejich činnost a dát jim alespoň základní prvky inteligence.

Mezi často používané mikropočítače patří výrobky firmy Freescale, dříve Motorola, které si získaly proslulost především díky vysoké kvalitě, mimořádné spolehlivosti, příznivé ceně a značně rozsáhlému perifernímu vybavení, jako jsou např. standardně obsažený vícekanálový A/D převodník či široce koncipovaný návrh časovače.

Těmito vlastnostmi se chlubí i poměrně nová řada osmibitových mikropočítačů HC(S)08, která představuje nástupce známého mikropočítače M68HC11. Svým návrhem navazuje na starší řadu HC05, se kterou je plně kompatibilní (směrem vzhůru).

Oblastmi, kde jsou výrobky Freescale používány nejčastěji, jsou automobilový průmysl, telekomunikace a spotřební elektronika. Stále více se však uplatňují také v řídicích systémech malého a středního rozsahu, kde mohou využít svých širokých možností.

Smyslem této bakalářské práce je navrhnout a realizovat funkční model tepelné soustavy s dvoupolohovým ovládním topení a měřením teploty pomocí mikropočítače a dále vytvořit fyzické a programové propojení mezi mikropočítačem HCS08 a tímto modelem. Mikropočítač zde bude sloužit jako řídicí prvek, který bude umožňovat měření teploty soustavy a dále, podle uživatelem definovaného algoritmu, ovládat přívod

elektrické energie pro topný prvek. V ukázkové aplikaci bude realizován algoritmus pro dvoupolohový regulátor bez hystereze.

Dá se bez nadsázky říci, že měření a regulace teploty je jednou z nejčastěji realizovaných úloh automatizace, bez které by byla nemyslitelná řada průmyslových, lékařských nebo vědeckých procesů a technologií. Snahou je zajistit, aby se tato činnost obešla bez zásahu člověka často po podobu týdnů, měsíců, či let. Tato práce si však klade podstatně skromnější cíle. Hlavním úkolem je vytvořit učební pomůcku, která by při výuce programování mikropočítačů HCS08 studentům alespoň nastínila jednu z mnoha oblastí, ve které je možné toto zařízení s výhodou využít. Dalším cílem, kterého by mělo být dosaženo, je vytvořit materiál, který by srozumitelnou formou přiblížil problematiku mikropočítačů a regulační techniky s přihlédnutím k měření a regulaci teploty.

I. TEORETICKÁ ČÁST

1 MIKROPOČÍTAČ MOTOROLA HC(S)08

1.1 Vývoj mikroprocesorů a mikropočítačů

Vývoj mikroprocesorů, potažmo jednočipových mikropočítačů se začíná datovat od konce roku 1947, kdy v Bellových laboratořích v New Jersey doktor Shockley společně se svými kolegy, doktory Bardeenem a Brattainem, prováděl pokusy s kouskem germania opatřeného třemi vývody. Vznikl tak první hrotový tranzistor na světě. Tento tranzistor se však těm, které používáme dnes, příliš nepodobal. Jeho konstrukce byla velmi primitivní a vlastnosti zdaleka ne ideální. Především zesílení bylo malé, maximálně dvacetinásobné. Navíc, protože použité germanium nemělo dostatečnou čistotu, tento tranzistor trpěl značným šumem a měl velmi nízký mezní kmitočet. Přesto však bylo jasné, že tato součástka dokáže to samé co nesrovnatelně větší, složitější a dražší vakuová elektronika.

Byl tak zahájen proces, který ve svém důsledku vedl až k tomu, že výpočetní výkon jednoduchého mikropočítače, který bývá umístěn například v mikrovlnné troubě, je vyšší, než byl výkon prvních elektronkových počítačů, které zabíraly plochu výrobní haly.

Vývoj mikroprocesorů a mikropočítačů tak, jak je známe dnes, začal roku 1971, kdy pracovník firmy Intel E. Hoff při práci na vývoji obvodů pro stolní kalkulátor, navrhl obvod I4004. Tento obvod byl nazván mikroprocesor. Měl šířku datové sběrnice 4 bity, instrukční soubor orientovaný hlavně na práci s čísly BCD a na logické funkce a obsahoval 2300 tranzistorů. Začátkem roku 1974 pak firma Intel přišla s typem 8080, který měl šířku datové sběrnice 8 bitů a stal se na delší dobu jakýmsi světovým standardem. Byl mimo jiné vyráběn také v tehdejší Československu pod označením MHB8080A.

Asi rok po zavedení typu 8080 uvedla firma Motorola na trh mikroprocesor MC6800 a v roce 1976 firma ZILOG mikroprocesor Z80-CPU. Toto byli hlavní představitelé osmibitových mikroprocesorů. Od roku 1978 jsou středem pozornosti 16bitové mikroprocesory. Mezi ně patří mikroprocesor 8086 firmy INTEL, uvedený na trh v roce 1978, který byl vyroben technologií HMOS. Obsahoval 29 000 tranzistorů. S ročním zpožděním nabídla firma ZILOG mikroprocesor Z8000 a firma Motorola mikroprocesor MC68000, který se navenek jeví jako 16bitový, ale vnitřní strukturu má z velké části 32bitovou.

Koncem 70. let nevznikaly jen 16bitové mikroprocesory, ale objevují se i první monolitické mikropočítače, které představují samostatnou větev vývoje. Mezi první patřil

například osmibitový obvod 8048 firmy Intel, nebo mikro počítač 6801 firmy Motorola.

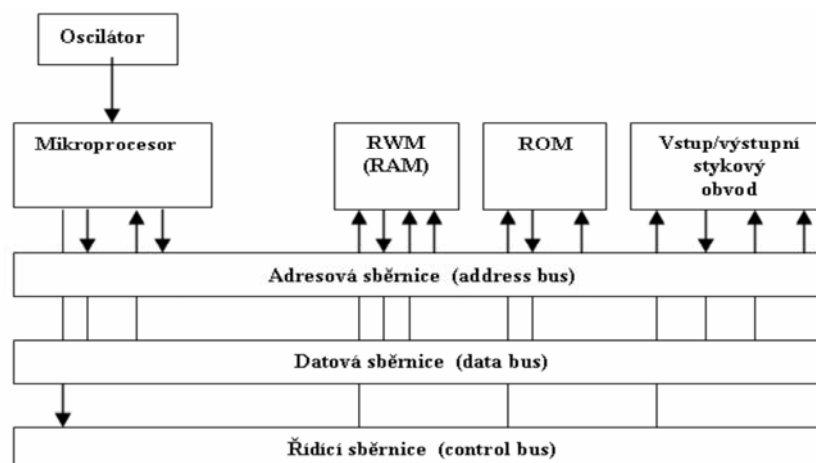
Pokud porovnáme vývoj v oblasti procesorů pro osobní počítače a vývoj v oblasti monolitických mikro počítačů, všimneme si jednoho základního rozdílu. Rozvoj u osobních počítačů je velmi rychlý. Každým rokem tak dochází k výraznému nárůstu výkonu a složitosti jednotlivých obvodů.

V oblasti monolitických mikro počítačů není tento trend vývoje tak výrazný. Mnoho mikro počítačů je jen zdokonalenými verzemi obvodů, které byly navrženy často před více než 20 lety. Vývoj se zaměřuje především na periférie umístěné na čipu, vylepšení výrobních technologií a zvýšení spolehlivosti. Výrazným krokem vpřed je také používání vnitřních pamětí typu FLASH místo dříve používaných ROM nebo EEPROM.

1.2 Monolitické mikro počítače

1.2.1 Struktura monolitického mikro počítače

Na (Obr. 1) je uvedeno zjednodušené schéma mikro počítačového systému. Obecně se každý počítačový systém skládá z jednoho nebo více mikroprocesorů, adresové, datové a řídicí sběrnice, paměti a vstupně/výstupních obvodů pro komunikaci mezi mikroprocesorem a periferními jednotkami.



Obr. 1 – Struktura mikro počítačového systému

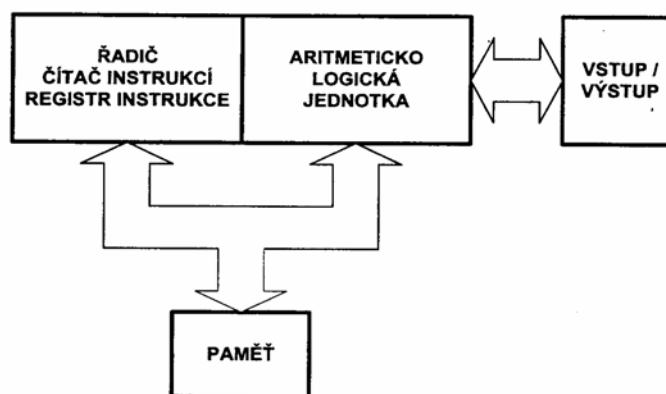
Hlavní rozdíl mezi mikroprocesorem a monolitickým mikro počítačem je ten, že mikro počítač sdružuje na jediném čipu přinejmenším mikroprocesor, generátor hodinového signálu, paměť a také vstupně/výstupní brány v rozsahu, který umožňuje alespoň v malé míře samostatnou činnost.

1.2.2 Rozdělení monolitických mikropočítačů

Monolitické mikropočítače můžeme dělit podle různých kritérií stejně jako výkonnější typy počítačů. Mezi základní dělení patří dělení podle koncepce procesorů:

a) koncepce podle von Neumanna

Tato koncepce (Obr. 2) předpokládá použití jednoho univerzálního paměťového prostoru jak pro data, tak pro umístění programu. U této koncepce není jednoznačně určeno, kde v paměťovém prostoru počítače jsou uložena data a kde je program. Používá se především u univerzálních výpočetních systémů, u kterých není dán pevný poměr mezi rozsahem programu a dat. Procesor má k dispozici pouze jednu množinu adres a programátor může vytvořit program, který se sám modifikuje v průběhu výpočetního procesu. Na základě von Neumannovy koncepce pracují všechny osobní počítače a vyšší systémy a také některé monolitické mikropočítače – mezi představitele patří například mikropočítače firmy Motorola, jako jsou typové řady HC05, HC08 a HC11. [1]

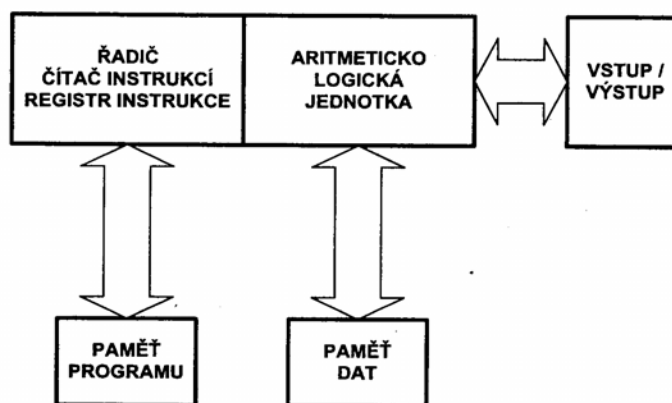


Obr. 2 – von Neumannova koncepce počítače

b) harvardská koncepce

U monolitických mikropočítačů se často uplatňuje tzv. harvardská koncepce (Obr. 3). Je to dáno především oblastí použití monolitických mikropočítačů jako jednoúčelových řídicích prvků ve spotřební a průmyslové elektrotechnice. U těchto aplikací lze při návrhu celého zařízení odhadnout velikost paměti, kterou budou zabírat data a program. Harvardská koncepce předpokládá dva oddělené paměťové prostory, které představují pro procesor dvě různé množiny adres. Z jedné lze pouze číst instrukce programu, druhou je možné používat pro ukládání zpracovávaných dat. Separovaná paměť programu dovoluje použít rozdílnou délku slova pro zpracovávaná data a pro program. [1]

Při vhodně navrženém instrukčním souboru je možné umístit každou instrukci na jednu adresu paměti a zrychlit tím vykonávání programu. Řadič procesoru nemusí vybírat informaci o instrukci z více buněk a zkracuje se doba výběru instrukce. Mezi představitele patří například velmi oblíbený mikropočítač 8051 firmy Intel, apod.



Obr. 3 – Harvardská koncepce počítače

Dalším kritériem, podle kterého lze dělit mikropočítače, je šířka datové sběrnice. Ta může být 4, 8, 16 a 32 bitů. Mezi nejčastěji používané patří 8 a 16-ti bitové mikropočítače, které jsou výhodné díky svému výkonu a poměrně nízké ceně.

Stále širší uplatnění nacházejí také 32bitové systémy a to především v oblastech, kde potřebujeme značný výkon a nevadí nám vyšší cena.

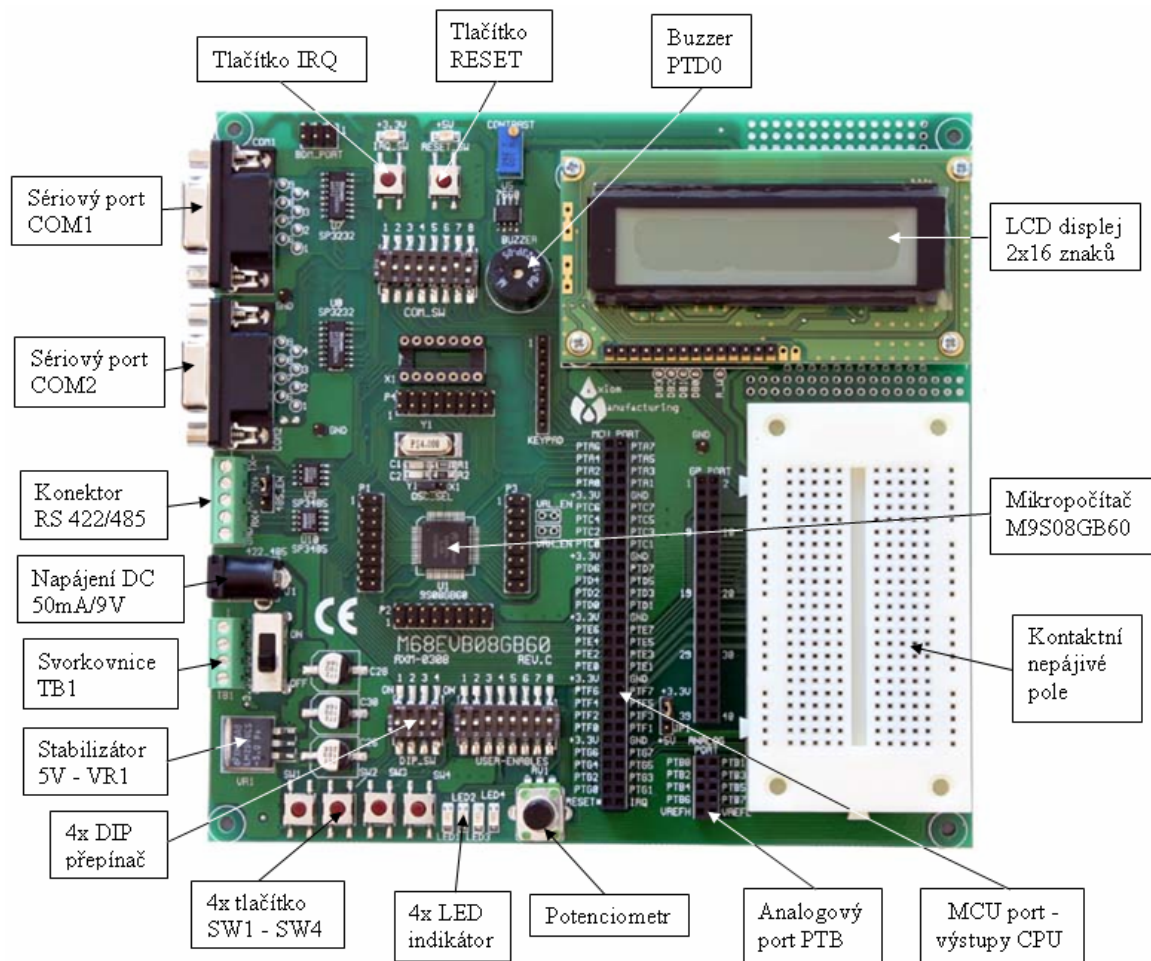
1.3 Vývojový kit M68EVB908GB60

Tento moderní vývojový prostředek je určen pro vývoj a ladění aplikací určených pro mikroprocesor Freescale MC9S08GB60. Je vybaven kontaktním nepájivým polem, které umožňuje využít výstupy mikroprocesoru podle potřeb programátora, dvouřádkovým LCD displejem, potenciometrem, který je určen pro simulaci A/D převodu a jednoduchými perifériemi jako jsou diody LED, tlačítkové spínače, přepínače DIP a piezoelektrický akustický měnič - buzzer. Kit M68EVB908GB60 umožňuje sériovou komunikaci s nadřazeným systémem pomocí rozhraní RS-232C, sběrnice I2C a průmyslového standardu RS422 & 485.

Pro komunikaci mezi perifériemi umístěnými na vývojové desce je vybaven synchronním sériovým rozhraním SPI. [9]

1.3.1 Seznam základních parametrů vývojového kitu M68EVB908GB60

- mikroprocesor M9S08GB60
 - 60K Bajtů paměti FLASH
 - 4K Bajtů paměti RAM
 - 56 I/O linek (64 vývodové pouzdro QFP 14x14mm)
 - 5 kanálový TPM 2 časovač
 - 3 kanálový TPM 1 časovač
 - 8 kanálový 10bit A/D převodník
 - SPI a I2C sériové rozhraní
 - 2 x SCI sériové rozhraní
 - výstupy CPU vyvedeny jako porty P1, P2, P3, MCU, GP a Analog
 - BDM Port
 - generátor hodin w/PLL
 - pracovní frekvence až 40Mhz
- oscilátor 32kHz až 4MHz
- nastavitelný generátor hodinového signálu
- napájecí zdroj +3.3V a +5V
- COM1 sériový port RS232 s konektorem DB9-S
 - SCIO sériové rozhraní
- COM2 CAN sériový port RS232 s konektorem DB9-S nebo konektor RS422/485.
 - SCII sériové rozhraní
- vypínač ON/OFF, signalizace napájení
- uživatelské periferie
 - čtyři LED diody (PTF0-3)
 - přepínač DIP 4x (PTB4-7)
 - čtyři tlačítkové spínače (PTA4-7)
 - modul LCD displeje 2x16 znaků (PTG3-7, PTE6-7)
 - bzučák (PTD0)
 - potenciometr pro simulaci A/D převodu
- velké kontaktní nepájivé pole o rozměrech 5,08 cm x 12,70 cm
- MCU Port s digitálními vstupy a výstupy



Obr. 4 – Vývojový kit M68EVB908GB60

1.3.2 Specifikace

rozměry desky: 15,24 cm x 15,75 cm

napájecí napětí: 6 - 20V DC, typicky 9V DC

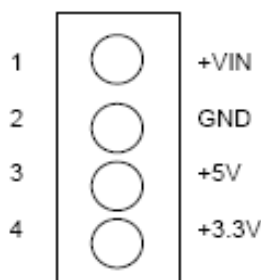
odběr: 50mA / 9V DC

1.3.3 Napájení

Napájení je realizováno externím zdrojem, který může být připojen ke konektoru J1 nebo ke svorkovnici TB1. Napájecí napětí je stabilizováno obvodem VR1, který je chráněn proti připojení opačného napětí diodou D3. Maximální proud je omezen tepelnou pojistkou FZ1. Vstupní napětí +5V je dodáváno stabilizátorem VR1. Toto napětí je dále regulováno obvodem VR2 na hodnotu +3,3V. Přítomnost napětí +5V a +3,3V je indikována LED diodami, umístěnými nad tlačítky IRQ_SW a RESET_SW. [9]

1.3.4 Svorkovnice TB1

Svorkovnice TB1 poskytuje přístup k napájecímu napětí +VIN, zemní svorce GND a stabilizovaným napětím +5V a +3,3V. Přípojka +VIN není chráněna vypínačem ani pojistkou. Pozice +5V a +3,3V mohou sloužit jako 50mA zdroj. Napájení desky napětím +5V nebo +3,3V z vnějšího zdroje přes svorkovnici TB1 není vhodné. [9]



Obr. 5 – Svorkovnice TB1

1.4 Popis mikropočítačů M9S08GB60

Mikropočítač M9S08GB60 patří do typové řady HCS08, která navazuje na starší řadu HC05. Patří mezi procesory typu CISC (procesory s úplnou instrukční sadou) a jak bylo uvedeno v kapitole 1.2.2, jeho struktura odpovídá von Neumannově koncepci, kdy data i program jsou umístěny ve stejném paměťovém prostoru.

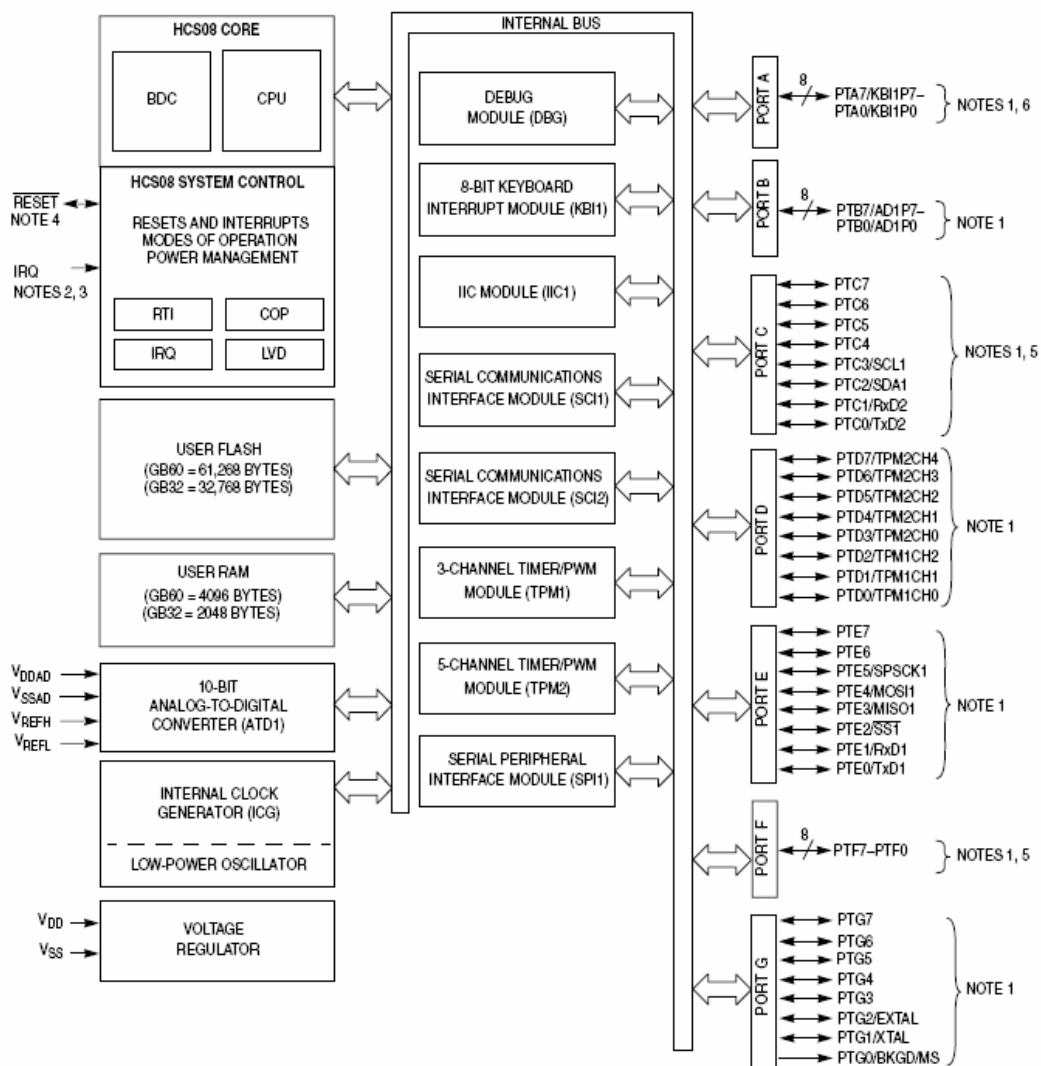
Struktura jádra HC(S)08 vychází z historického mikroprocesoru 6800 a jednočipového mikropočítače 6801. Jádro řadiče HC(S)08 bylo proti mikropočítači 6801 zjednodušeno. Charakteristikou je minimum pracovních registrů. Použití jednoho střadače je kompenzováno novými instrukcemi pro efektivnější práci s bity v paměti RAM a v registrech periférií. [2]

Tento mikropočítač poskytuje 56 vstupně/výstupních linek, které jsou organizovány do sedmi osmibitových portů (A-G). Některé z těchto linek mají pevně stanovenou funkci a směr přenosu, jiné lze použít zcela podle uvážení programátora.

1.4.1 Struktura mikropočítačů řady MC9S08GB/GT

Mikropočítač řady MC9S08GB/GT patří mezi nejvybavenější zástupce rodiny HCS08. Je vybaven 60kB paměti FLASH, 4kB paměti RAM, budičem LCD displeje, osmi kanálovým 10-bit A/D převodníkem, sériovým asynchronním rozhraním SCI (RS-232C,

RS-422 & 485), sériovým synchronním rozhraním SPI a rozhraním I2C. Představuje značně výkonný prostředek určený i pro náročné aplikace.



Obr. 6 – Struktura mikropočítače MC9S08GB/GT

1.4.2 Mikroprocesor

Mikroprocesor nebo jinak také CPU (Central Processing/Procesor Unit) představuje hlavní výkonnou část mikropočítače. Zpracovává instrukce od programů, kterými je řízen a plní tak zadané úkoly. Skládá se ze dvou částí - řadiče a aritmeticko-logické jednotky.

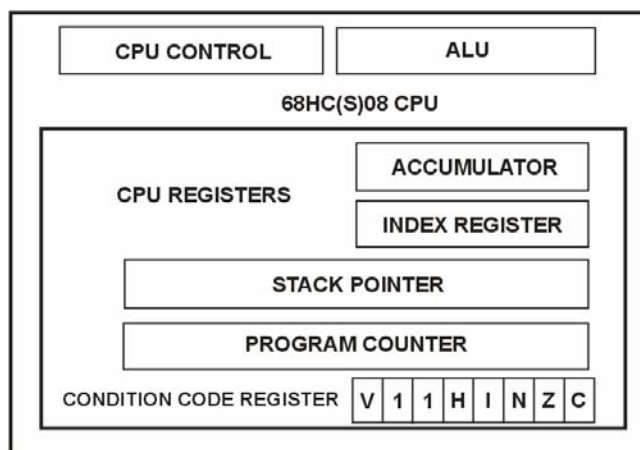
Řadič řídí všechny funkce mikropočítače. Generuje potřebné signály pro spolupráci jednotlivých částí mikropočítače. Při provádění instrukcí plní řadič tyto tři úkoly:

- řídí pořadí, v němž jsou instrukce prováděny
- dekóduje instrukce a v případě potřeby je modifikuje (skoky v programu)

- vysílá do všech ostatních částí mikropočítače řídicí signály potřebné pro provádění instrukcí

Aritmetickologická jednotka provádí aritmetické i logické operace, porovnávání a také posuny dat. Operace zde probíhají na nejnižší matematické úrovni (sčítání a odčítání).

Jednotka CPU HC(S)08 (Obr. 7) je plně kompatibilní s rodinou 68HC05. Obsahuje šestnáctibitový ukazatel zásobníku (Stack Pointer), šestnáctibitový index registr (*registr* - kapitola 1.4.4). Umožňuje adresaci až 64kB paměti, 16 adresovacích módů, přesuny z paměti do paměti bez použití střadače, rychlé násobení (8x8 bitů) a dělení (16/8 bitů). Interní sběrnice má až 8MHz. Umožňuje funkci v režimech HALT a STOP. [2]



Obr. 7 – Struktura CPU 68HC(S)08

Programovací model (Obr. 8) mikropočítače HC(S)08 je tvořen osmibitovým střadačem A, který je využíván při aritmetických a binárních operacích, při instrukcích pro porovnávání, apod.. Model dále obsahuje šestnáctibitový indexový registr H:X. Tento registr je některými instrukcemi využíván jako dva samostatné registry H a X. Mezi tyto instrukce patří např. instrukce pro celočíselné dělení DIV, která vydělí číslo v registrech H:A číslem v registru X. Výsledek je uložen do registru A, zbytek po dělení je uložen do registru H.

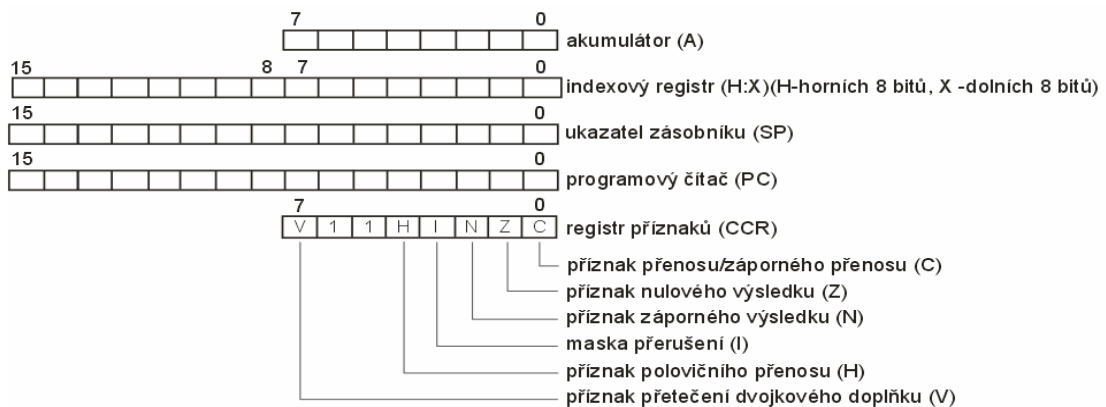
$$\text{DIV: } A \leftarrow (H:A) / (X)$$

$$H \leftarrow \text{zbytek}$$

U mikropočítače HC(S)08 neexistuje instrukce, která by ukládala registr H:X na zásobník přímo. Proto je třeba uložit, popř. vyjmout registry H a X samostatně pomocí instrukcí pro práci se zásobníkem (uložení na zásobník PSHH, PSHX, vyjmutí ze zásobníku PULH, PULX). Stejně tak neexistuje instrukce, která by umožňovala přímo

adresovat registr H. Pokud tedy potřebujeme ovlivnit nebo načíst obsah registru X, musíme použít instrukce pro práci se zásobníkem.

Dalšími registry programovacího modelu jsou šestnáctibitový ukazatel zásobníku SP a šestnáctibitový programový čítač. Posledním registrem je osmibitový stavový registr CCR.



Obr. 8 – Pracovní registry HC(S)08

1.4.3 Sběrnice

Systémová sběrnice (SYSTEM BUS) - paralelní soubor vodičů, který propojuje mikroprocesor s podpůrnými obvody. Skládá se ze tří částí:

Datová sběrnice (DATA BUS) - část vodičů určených pro přenos datových informací. Má obousměrný charakter.

Adresová sběrnice (ADDRESS BUS) - lokalizuje adresové místo, z kterého jsou data čerpána, nebo kam jsou data ukládána. Je jednosměrná, výstupní.

Řídící sběrnice (CONTROL BUS) - specifikuje směr přenosu, tj. řízení paměti a podpůrných obvodů.

1.4.4 Paměť

Paměť rozumíme jakékoliv zařízení, do kterého je možné uložit a kdykoliv poté opět vybavit informace. Paměti posuzujeme podle několika kritérií, jako jsou:

- Kapacita - množství informace, která se udává v bitech, popř. v bytech
- Vybavovací doba - čas potřebný k zápisu nebo k přečtení informace, měřený od okamžiku, kdy je známa adresa

- Spolehlivost - udává možnost vzniku chyb při čtení nebo zápisu
- Trvanlivost zápisu - posuzuje se pro kolísání napájecích napětí nebo jejich přerušování a podle doby, po kterou informace v paměti zůstává bez jejího porušení.
- Cena za 1 bit – podíl ceny a kapacity

Velmi důležité je dělení paměti podle možnosti změny dat:

- a) paměti RWM pro čtení i zápis dat za běžného provozu v počítači
- b) pevné paměti ROM, ze kterých lze za běžného provozu data jen číst. Data z nich nezmizí ani při vypnutí napájení.
 - (1) ROM - informace se do nich ukládají při jejich výrobě
 - (2) PROM - vyrábějí se informačně prázdné a data do nich může uživatel zavést sám (pouze jednou)
 - (3) EPROM - data lze vymazat ultrafialovým zářením a znovu naprogramovat
 - (4) EEPROM - jsou určeny převážně ke čtení. Počítač má sice možnost data přepisovat, ale jen ve speciálním režimu a výrazně pomaleji, než když data čte.
 - (5) FLASH - paměti mohou být zcela vymazány jediným elektrickým impulsem (flash-blesk) a potom přeprogramovány bit po bitu

Adresový prostor HC(S)08 lze podle účelu, rychlosti a přístupu k datům rozdělit na:

- Registry a zápisník
- Zásobník
- Obecnou datovou paměť

Registry a zápisník

Registry a zápisník slouží pro dočasné uchovávání zpracovávaných dat. Pomocí registrů mikropočítač přijímá a předává informace důležité pro chod programu, operandy aritmetických a logických operací, adresy a kódy právě vykonávaných instrukcí, paměťových buněk s kterými pracuje, aktuální nastavení a stav periférií, žádosti o přerušování, příznaky přetečení, atd. Některé registry (instrukční registr, adresový registr, ...) jsou určeny pouze pro čtení a snaha o zápis dat nemá žádný význam. [2]

Zápisník je konstruován jako paměť typu RAM. Kapacita zápisníku je zpravidla velmi omezená a je určen především k ukládání mezivýsledků nebo důležitých proměnných.

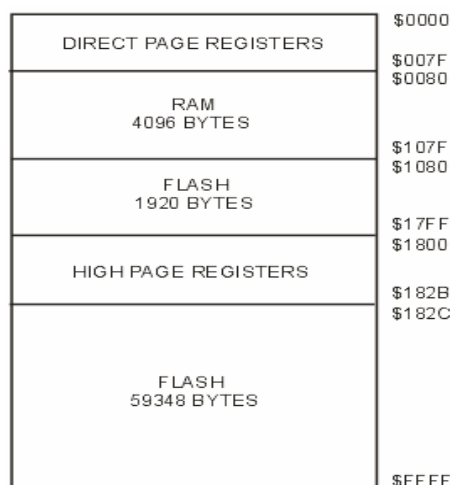
Zásobník

Zásobník u mikro počítače HC(S)08 je koncipován jako paměť typu LIFO (Last In First Out). Tento název vyjadřuje, že přístup k uloženým datům je sekvenční. To znamená, že informace uložená na vrchol zásobníku bude přečtena jako první. Vrcholem zásobníku rozumíme aktuální paměťovou buňku, do které ukládáme, nebo ze které načítáme data. Adresu vrcholu uchovává ukazatel zásobníku (SP - Stack Pointer), který je při zápisu dat inkrementován a při čtení dat dekrementován. Mikro počítače HC(S)08 inkrementují ukazatel zásobníku směrem k nižším adresám, tj. dno zásobníku má vyšší adresu než vrchol zásobníku. [10]

Zásobník nemusí být využíván jen uživatelským programem. Využívá jej také jednotka CPU pro uložení návratové adresy při volání podprogramu nebo při přerušení. Během přerušení se na zásobník automaticky ukládá obsah všech pracovních registrů, s výjimkou registru H.

Obecná datová paměť

Obecná datová paměť slouží k ukládání dat programu a větších datových struktur. Jedná se o kapacitně nejrozsáhlejší paměť, kterou mikro počítače disponují. V řadě HC(S)08 je realizována jako paměť typu ROM, EEPROM a FLASH. [10]



Obr. 9 – Paměťová mapa mikro-
počítače MC9S08GB/GT

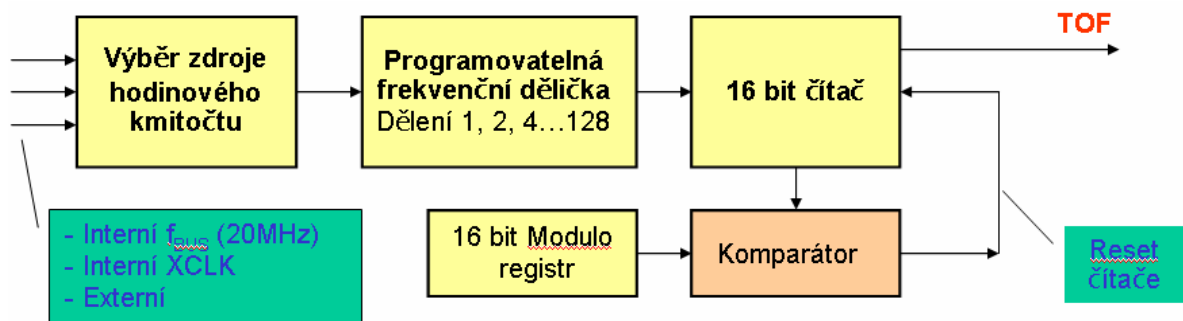
1.5 Periferie mikroprocesoru HC(S)08

Názvy registrů uvedené v následujících odstavcích jsou pouze informativní a nelze je použít při programování. Přesné názvy těchto registrů pro mikropočítač MC9S08GB60 jsou obsaženy např. v souboru MC9S08GB60.inc, který je dostupný v prostředí CodeWarrior ve složce Includes.

1.5.1 Časovač

Časovač je nejstrukturovanější periferií mikropočítačů HC(S)08. Slouží pro generování a měření přesných časových intervalů, generování maskovatelných přerušení v přesně stanoveném okamžiku (např. přerušeni každých 500ms pro generování pravoúhlého signálu s frekvencí 1Hz), dále také slouží jako zdroj hodinového signálu pro obvod COP (Watchdog).

U mikropočítače MC9S08GB60 jsou realizovány dva samostatné časovače TPM1 a TPM2. Tyto dva moduly se od sebe liší počtem kanálů (TPM1 má 3 kanály a TPM2 má 5 kanálů). Každý kanál může nezávisle pracovat v režimech Input Capture, Output Compare nebo PWM.



Obr. 10 – Schéma časovače mikropočítače HC(S)08

Input Capture

V této funkci je do jednoho z osmi šestnáctibitových záchytných registrů (TIC1 – TIC8) na základě vnější události (náběžná hrana signálu, sestupná hrana signálu, popř. obojí) registrované na jednom z vývodů portu D, přenesen stav hlavního, volně běžícího čítače. Tato událost je také zaregistrována nastavením příslušného bitu ve stavovém registru TFLGx. Může být rovněž aktivováno přerušeni.

Tuto funkci lze využít pro měření periody, fázového posuvu, PWM, apod.

Output Compare

Tato funkce srovnává stav volně běžícího čítače s hodnotou přednastavenou v jednom z osmi šestnáctibitových registrů (TOC1 – TOC8). Při dosažení shody je nastaven příslušný bit a tím dojde k registraci. Může být rovněž aktivováno přerušení. Kromě toho dojde k nastavení výstupního vývodu spojeného s touto funkcí.

PWM

PWM - zkratka z anglického Pulse Width Modulation - pulsně šířková modulace signálu. Tato metoda je založena na signálu s konstantní periodou T , u kterého se mění střída napětí (tj. poměr doby, kdy má signál hodnotu $\log 1$ a doby, kdy má signál hodnotu $\log 0$).

Do úrovně $\log 1$ nebo $\log 0$ (lze nastavit v řídicím registru) přechází signál v momentě přetečení hlavního čítače, změna úrovně je zajištěna jednotkou výstupního komparátoru.

COP (Computer Operating Properly)

Obecně známý spíše pod názvem Watchdog, je nezávislý časovač s nastavitelnou časovou konstantou (2^{18} nebo 2^{13} cyklů sběrnice), vybavený signálem přetečení. Úkolem tohoto modulu je hlídat korektní chod programu. Pokud program funguje tak jak má, je Watchdog pravidelně softwarově resetován, a proto nikdy nedojde k přetečení časovače. Pokud se však program začne chovat nestandardně (např. dostane se do nekonečné smyčky), dojde k přetečení a následně k vyvolání restartu mikropočítače. Funkci Watchdog lze softwarově vyřadit z provozu.

1.5.2 A/D převodník

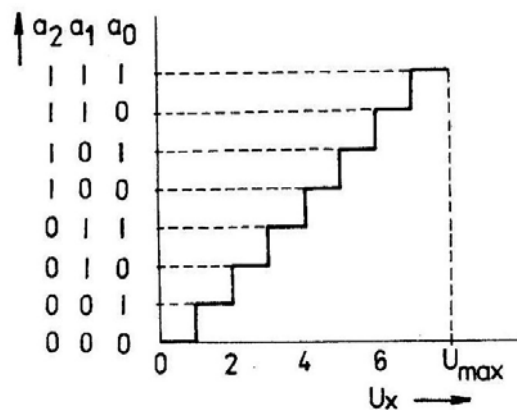
Převodník A/D mění spojitě se měnící (analogový) vstupní signál na odpovídající výstupní číslicový signál v určitém kódu. Slouží ke kvantitativnímu vyjádření velikosti vstupní veličiny pomocí čísel. Funkce takového převodníku je zřejmá ze statické převodní charakteristiky (Obr. 11). Při převodu vznikne číslicový signál diskrétní v čase i amplitudě.

Diskretizace signálu v čase se realizuje pravidelným časovým vyhodnocováním okamžité hodnoty analogového signálu, což nazýváme vzorkování. V určitém časovém okamžiku se zjistí hodnota analogové veličiny a až do dalšího zjištění se považuje její hodnota za konstantní. Tímto postupem vzniká chyba, která se však zvyšováním frekvence

vzorkování zmenšuje. Diskretizace signálu v amplitudě se nazývá kvantování. Každá z hodnot kvantovaného signálu může být vyjádřena číslcovým kódem. Číslcový údaj na výstupu převodníku A/D má nejčastěji formu "počet impulsů úměrný analogové veličině", nebo přímo formu dvouhodnotového vícebitového signálu, kódového slova složeného jen z hodnot 0 a 1.

Podstata činnosti převodníků A/D je různá. Rozdělujeme je do dvou skupin:

- nepřímé převodníky s mezipřevodem na čas nebo frekvenci
- přímé převodníky (kompenzační a komparační)



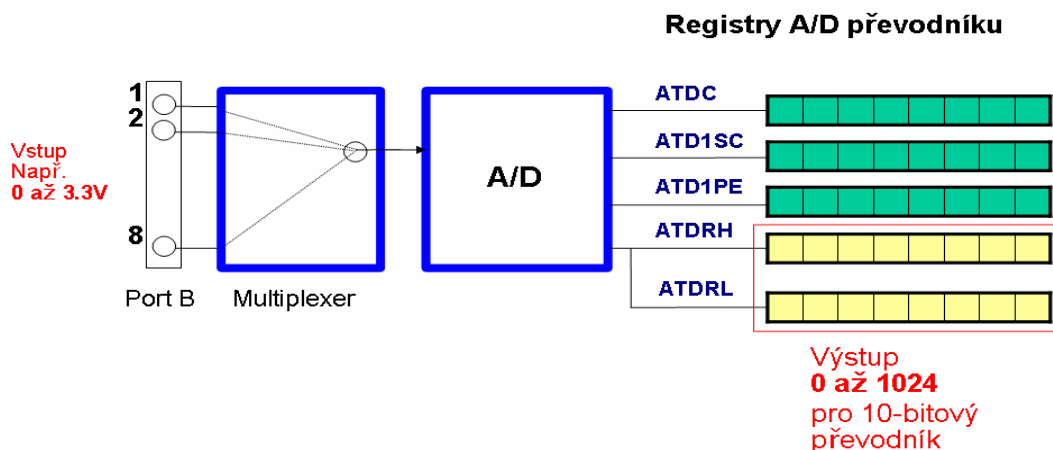
Obr. 11 – Statická charakteristika
tříbitového A/D převodníku

Mikropočítače HC(S)08 obsahují osmi nebo desetibitový A/D převodník s postupnou aproximací. U mikropočítače M9S08GB60 je převodník připojen přes osmikanálový analogový multiplexer k vývodům portu B a umožňuje podle nastavení řídicího registru ATDC osmi nebo desetibitový převod.

Doba trvání jednoho A/D převodu je přibližně 14 μ s. Výsledek převodu je uložen do dvojice registrů ATDRH (vyšší bajt) a ATDRL (nižší bajt). Dokončení převodu je signalizováno buď nastavením příznaku (bit CFF v řídicím registru ATD1SC) nebo vyvoláním přerušení (vektor přerušení na adrese \$FFD0).

A/D převodník může pracovat ve dvou režimech:

- režim jednoho převodu (single)
- kontinuální převod (continuous)



Obr. 12 – Schéma A/D převodníku mikropočítače HCS08

1.6 Komunikační rozhraní

Aby mohl mikropočítač komunikovat s okolními zařízeními, jako jsou například moduly LCD, vnější paměti nebo nadřazený systém, je třeba jej vybavit komunikačním rozhraním. Existují dvě základní metody jak komunikační rozhraní realizovat.

- a) Paralelní komunikace
- b) Sériová komunikace

1.6.1 Paralelní komunikace

Paralelní rozhraní realizuje přenos dat mezi vnitřní sběrnici řídicího systému a vnějším prostředím po znacích - bajtech. Pro komunikaci je tak zapotřebí minimálně osmi vývodů pro současné vysílání jednotlivých bitů tvořících jeden bajt. Původně bylo paralelní rozhraní určeno pro komunikaci mezi počítačem a tiskárnou. U mikropočítačů je někdy využíváno pro komunikaci mezi procesorem a vnější velkokapacitní pamětí. [1]

Paralelní porty mohou být jednosměrné nebo obousměrné, u starších typů bylo možné se setkat například s osmibitovým vysíláním a čtyřbitovým příjmem dat.

Vzhledem k tomu, že přenášíme více bitů současně, mohlo by se zdát, že paralelní komunikace je mnohonásobně rychlejší než sériový přenos. Ve skutečnosti však tento rozdíl není až tak výrazný, protože z technického hlediska je mnohem jednodušší zvýšit rychlost sériového přenosu, např. vhodnou modulací nebo zvýšením nosné frekvence, než dosáhnout stejného zvýšení komunikační rychlosti u paralelního rozhraní. Dalším problémem paralelní komunikace je velmi omezený dosah spojení (řádově metry).

U vývojového kitu M68EVB908GB60 není paralelní rozhraní realizováno, proto mu nebude věnována další pozornost.

1.6.2 Sériová komunikace

Při sériovém způsobu předávání zpráv se jednotlivé bity znaků přenášejí v čase po sobě po jednom vodiči. Původně tento způsob vznikl v telekomunikační technice, kde především šetřil spojovací vedení. Z obdobných důvodů nachází uplatnění i ve výpočetní technice, a to i u místně soustředěných systémů. Další možnosti před ním otevírá perspektiva optických spojů po skleněných vláknech. [1]

Zjednodušeně lze říci, že základem sériového přenosu jsou dva posuvné registry. Do registru vysílače mikroprocesor paralelně zavede vysílaný znak a ten se pak sériově vysouvá na vedení. Do registru přijímače znak naopak vstoupí z vedení sériově a přebírá se z jeho paralelních výstupů.

Při sériové komunikaci se nepřenáší pouze čistá původní informace, ale i redundantní bity, které přidá vysílač a které přijímač po kontrole odstraní.

Rychlost komunikace se nejčastěji udává v jednotkách Bd (Baud). Rychlost 1 Bd znamená jednu změnu od 0 k 1 nebo opačně za sekundu. Přitom přenos bitů je úrovněvý, tzn. má-li za sebou dva nebo více bitů např. hodnotu 1, pak signál mezi nimi neklesá k 0.

Rozlišujeme dva základní principy sériové komunikace:

- a) Asynchronní přenos
- b) Synchronní přenos

Asynchronní přenos (SCI)

Asynchronní způsob přenosu je založen na tom, že přenos každého znaku může být po skončení znaku předchozího zahájen kdykoliv. Vysílání se zahajuje startovacím bitem, tj. poklesem na L, podle kterého přijímač pozná začátek přenosu znaku. Pak se v pravidelných okamžicích vysílají, popř. přijímají bity přenášeného znaku, kterých může být (v závislosti na kódu) 5 až 8. Potom může, ale nemusí být vyslán paritní bit a pak následuje 1, 1,5 nebo 2 ukončovací stop bity. Všechny takto vyslané bity tvoří tzv. rámeček (frame). [1]

Princip asynchronního přenosu je znázorněn na (Obr. 13).



Obr. 13 – Asynchronní přenos

Rozhraní RS-232C

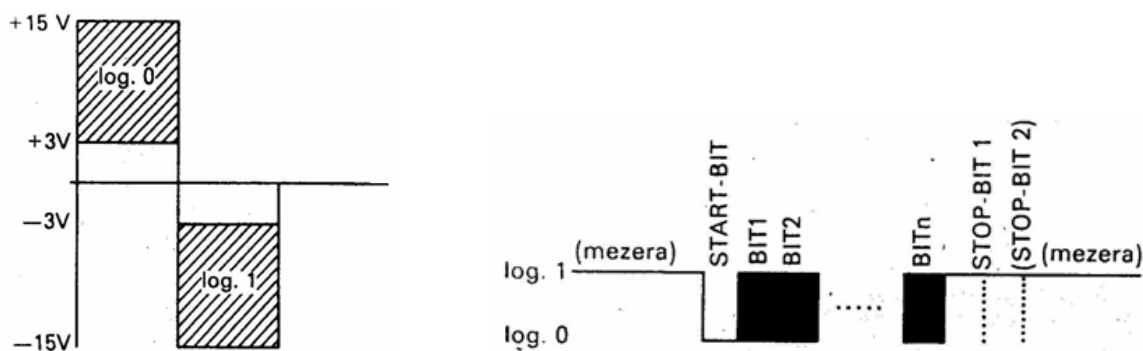
V roce 1962 bylo u americké asociace pro elektronický průmysl EIA normováno sériové rozhraní, dnes známé pod názvem normy EIA RS-232. Poslední úprava této normy s příponou "C" je platná dodnes. Toto rozhraní bylo později též normováno doporučením CCITT (mezinárodní komise pro telekomunikaci) pod článkem V.24 (odtud také "rozhraní V.24"). Z hlediska druhu a způsobu přenosu dat se u rozhraní RS-232C používá asynchronní přenos dat. Plná definice rozhraní RS-232C (resp. V.24) obsahuje kromě vlastních datových signálů též signály řídicí pro řízení přenosu informací mezi dvěma koncovými zařízeními pro přenos dat. Řízení přenosu je především záležitostí technických prostředků. [1]

Z elektrického hlediska se u rozhraní RS-232C používají tzv. nesymetrické vazební obvody podle doporučení CCITT V.28. Doporučení RS-232C tedy definuje funkci obvodů rozhraní a doporučení V.28 způsob jejich elektrické realizace. Přiřazení minimálních úrovní signálu jeho logickým hodnotám je uvedeno na (Obr. 14). Napájecí napětí obvodů je maximálně ± 15 V. Pro účely realizace rozhraní podle doporučení RS-232C byla navržena řada obvodů pro převod úrovní TTL na V.28 a naopak (např. obvody odvozené od MAX232 aj.).

Výhodou rozhraní RS-232C je jeho relativně jednoduchá technická realizace a dále jeho široké rozšíření (jedná se prakticky o celosvětový standard).

Nevýhodou tohoto rozhraní je jeho poměrně malý dosah (15m), výjimečně pro pomalé přenosy až 50m, a nízká přenosová rychlost 20000 Bd.

I přes tyto nedostatky představuje rozhraní RS-232C standardní propojení používané u PC a je prakticky pravidlem, že moderní řídicí systémy (např. PLC, PCC) jsou vybaveny buď přímo rozhraním RS-232, nebo rozhraním, které je s ním kompatibilní.



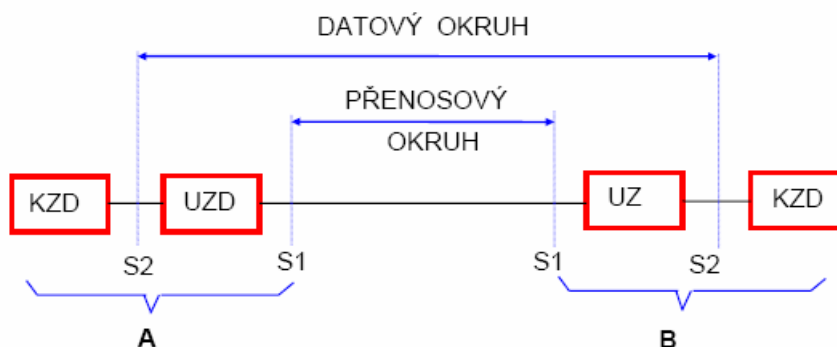
Obr. 14 – Úrovně RS 232C a formát dat RS 232C

Sériový datový okruh

Sériový přenos se děje mezi dvěma koncovými datovými zařízeními (KZD) prostřednictvím datového okruhu, který je tvořen dvěma ukončovacími datovými zařízeními (UZD), vzájemně propojenými prostřednictvím přenosového okruhu (Obr. 15). Úkolem UZD je upravit digitální signál z KZD tak, aby byl při přenosu na větší vzdálenosti odolnější proti rušení. Nejčastějším způsobem je vhodná modulace digitálního signálu na signál analogový.

Velmi účinným způsobem je také použití kroucené dvojlinky, která je mimořádně odolná proti rušení způsobenému indukovaným napětím.

Pokud se přenos děje na velmi krátkou vzdálenost (<15m), lze datový okruh vynechat a KZD propojit přímo. Propojení je nejčastěji realizováno prostřednictvím konektoru CANON9, resp. CANON25.



Obr. 15 – Sériový datový okruh

Standard RS-232 má v úplném zapojení až 25 vodičů a je tak zajištěna plná komunikace, včetně kontroly a řízení periferie (např. telefonní modem). Pro některé periferie však postačí pouze tři základní vodiče (TD, RD a SG). [3]

Rozhraní RS 485&422

Rozhraní RS 485&422 se používá v průmyslovém prostředí a v systémech pro řízení a přenos malých objemů dat. Schéma propojení RS 485 je na (Obr. 16).

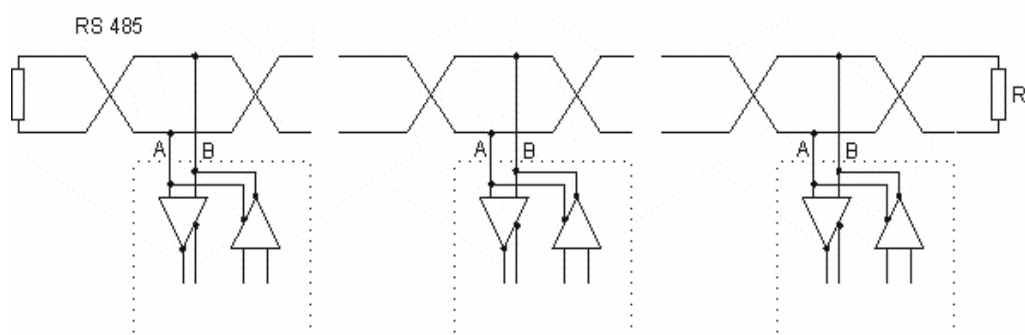
Pro každý signál je použito jedno vedení, realizované pomocí kroucené dvojlinky, tj. dvou vodičů smotaných kolem sebe. Z elektrického hlediska mluvíme o tzv. Balanced Data Transmission (BDT), nebo také Differential Voltage Transmission (DVT). Zjednodušeně řečeno, označíme-li si jeden vodič jako A a druhý jako B, potom platí: je-li signál neaktivní, je napětí na A záporné a na B kladné. V opačném případě, kdy je signál aktivní, je napětí na A kladné a na B záporné. Jedná se tedy o rozdíl (diferenci) mezi vodiči A a B. [14]

Pro RS 422 & 485 může délka vedení dosahovat přibližně 1200m a běžně vyráběné obvody dosahují přenosových rychlostí 2.5 MB/s.

U rozhraní RS 485 je komunikace mezi jednotkami typu master-slave (tj. jeden vysílá, jeden přijímá). U některých protokolů je vysílacích jednotek více. Jednotky se proto přepínají a vysílají postupně. Komunikace se označuje jako multimaster.

Rozhraní RS 422 je duplexní zapojení RS 485. Je to zdvojené zapojení pro současný přenos dat v obou směrech. [3]

Je důležité si uvědomit, že standard RS 422 & 485 popisuje pouze požadavky na technické prostředky a elektrické zapojení. Vlastní způsob přenosu dat je dán použitým přenosovým protokolem (např. Profibus, CAN, MessBus apod.).



Obr. 16 – Schéma propojení RS 485

Rozhraní I2C

V roce 1980 vyvinuli u firmy Philips Semiconductors jednoduchou obousměrnou dvoudrátovou sběrnici pro výkonové vnitřní řízení kolem obvodů s mikropočítači (MP).

Sběrnice obsahuje dva aktivní vodiče (SDA - Serial Data Line, SCL- Serial Clock Line) a vodič GND. Každé zařízení na sběrnici má vlastní unikátní adresu. Sběrnice I2C je tzv. multi-master sběrnice. To znamená, že více jak jedno zařízení se může připojit a uskutečnit svůj požadavek na přenos. Toto zařízení označujeme jako BUS Master, ostatní zařízení jsou v tomto čase BUS Slaves.

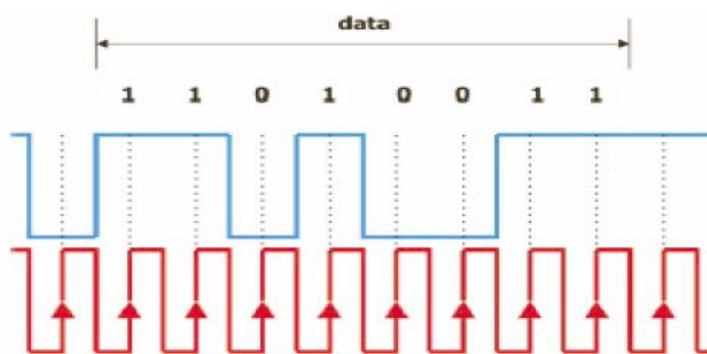
Hardwarová struktura sběrnice má dva vodiče obousměrné, tj. mohou být ovládané od MP nebo od externího zařízení. Pro zajištění těchto funkcí musí používat signály zapojení s otevřeným kolektorem nebo otevřeným výstupem. [3]

Synchronní přenos (SPI)

Synchronního přenosu se v mikropočítačích dosud používá méně a uplatňuje se hlavně pro rychlé přenosy velkého množství informací. V telekomunikacích je to přenos přes modemy a ve výpočetní technice např. výměna bloků informací mezi procesorem a velkokapacitní vnější pamětí, budičem LCD displeje, A/D převodníkem apod.

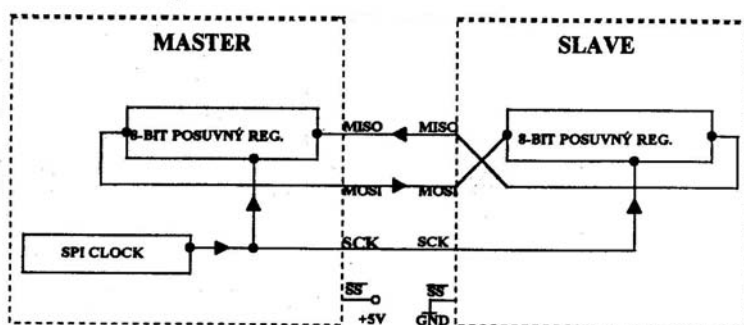
Značná délka souvislých zpráv a požadavky na rychlost přenosu i úsporu času na vedení vyžadují, aby přenášené znaky obsahovaly co možná nejméně redundantních bitů a aby následovaly za sebou v pravidelném těsném sledu. Odpadají tu tedy start bity a stop bity a přijímač i vysílač musí po celou dobu zprávy pracovat přísně synchronně. Toho lze dosáhnout buď souběžným přenosem hodinového signálu, což je dosti nepraktické, nebo vhodným způsobem modulace, kdy signál nese nejen data, ale i synchronizační informaci. Nepravidelnost příchodu dat od mikropočítače k vysílači se vyrovnává tím, že se do mezer automaticky vkládají speciální znaky SYNC, které přijímač využije při synchronizaci, ale které ze zprávy odstraní. [1]

Při synchronním přenosu vysílač i přijímač generují nebo přijímají různé pomocné signály.



Obr. 17 – Synchronní přenos

Principem tohoto přenosu je spojení dvou posuvných registrů do kruhu. Společnými taktovacími hodinami dochází k postupné vzájemné záměně obsahů těchto registrů. Tohoto mechanismu lze využít např. k tomu, že při jednom přenosu jsou z registru periferie převzata data a současně může být do něho umístěno řídicí slovo. Díky tomu, že přenos probíhá synchronně, dosahuje se podstatně vyšších přenosových rychlostí. Synchronizace přenosu je zajištěna hodinovým signálem, který je veden zvláštním vodičem. [1]



Obr. 18 – Komunikační propojení pomocí SPI

1.7 Přerušeni

Pokud potřebujeme, aby mikropočítač obsluhoval současně např. modul LCD displeje, A/D převodník a modul klávesnice, existují dva způsoby řešení.

První lze charakterizovat jako řízení v čekací smyčce. Spočívá v tom, že se testuje stav zařízení, například zda bylo stisknuto tlačítko klávesnice; pokud není podmínka splněna, test se neustále opakuje a program čeká ve smyčce na splnění podmínky. Po splnění podmínky se přechází k další části programu, která provede vlastní akci, v tomto případě sejmutí kódu tlačítka z klávesnice a vykonání příslušné funkce, např. provedení A/D převodu a zobrazení výsledku na displeji. Po provedení všech činností se program opět vrací k testu klávesnice a čeká, zda bylo stisknuto další tlačítko. Tato metoda není vhodná pro časově náročné programy, protože při čekání ve smyčce nemůže mikroprocesor obvykle vykonávat jinou činnost, takže je jeho činnost přidavnými zařízeními zpomalována.

Toto nevýhodu odstraňuje metoda založená na přerušeni programu od přidavného zařízení. Při tomto způsobu obsluhy přidavného zařízení vykonává procesor program, který nemusí přímo souviset s činností přidavného zařízení. V případě splnění určité podmínky vyšlou pomocné obvody zařízení do procesoru žádost o přerušeni. Pokud je

přerušeni povoleno a jsou splněny dané podmínky, procesor přeruší právě vykonávaný program, uloží obsah pracovních registrů na zásobník (u mikropočítače HC(S)08 se kvůli kompatibilitě s řadou HC05 na zásobník neukládá obsah registru H) a poté se začne věnovat obsluze přídatného zařízení. Na začátku obsluhy přerušeni mikropočítač provede skok na pevně danou adresu v paměti (tzv. vektor přerušeni). Tato adresa je dána při návrhu mikropočítače a každý typ přerušeni má svou unikátní adresu. Na ní se zpravidla nachází instrukce skoku na další adresu v paměti, na které je uložen začátek podprogramu pro obsluhu zařízení, které přerušeni vyvolalo. V tomto programu je třeba nejprve potvrdit příjem přerušeni (aby nedošlo k zacyklení volání obsluhy přerušeni) a potom provést vlastní úkony pro obsluhu. Po ukončení obsluhy přerušeni mikropočítač obnoví původní obsah pracovních registrů a vrátí se k vykonávání přerušeniho programu.

Pokud dojde k více žádostem o přerušeni současně, rozhoduje o tom, které bude vykonáno jako první, tzv. priorita. Zde vzniká problém s obsluhou přerušeni, která mají nízkou prioritu a mohou tak být po dlouhou dobu ignorována. Tento problém se řeší např. určením maximální doby, po kterou může zůstat žádost o přerušeni nevyřízena.

Bližší informace o obsluze přerušeni u mikropočítačů HCS08 a jednotlivých druhích přerušeni včetně vektorů těchto přerušeni lze nalézt např. v [8].

1.8 Režimy provozu se sníženým příkonem

Mikropočítače HC(S)08 jsou obvodově řešeny pro optimální poměr mezi výkonem čipu a jeho spotřebou. Jádro systému navíc umožňuje programové převedení do dvou úsporných režimů: WAIT a STOP.

1.8.1 Režim WAIT

Tento režim se využívá v situacích, kdy není vyžadována činnost vlastního procesoru, ale jedna nebo více periférií musí pokračovat ve své činnosti. CPU při přechodu do tohoto režimu sníží svůj odběr na méně než 10mA při napětí 5V a čeká na přerušeni od vnitřní periférie. Pro další snížení spotřeby lze programově vyřadit z činnosti jednotlivé periférie a příkon mikropočítače tak dále snížit.

1.8.2 Režim STOP

V tomto režimu se CPU a vnitřní periférie zastaví a jejich činnost může být obnovena pouze vnějším přerušením nebo restartem mikropočítače. Obsah paměti RWM a

všech pracovních registrů se zachovává. Odběr v režimu stop nepřesahuje 100 μ A při napětí 5V.

1.9 Jazyky symbolických adres

Jazyky symbolických adres patří mezi nejstarší programovací jazyky, se kterými se setkáváme již u počítačů první generace. Dříve než se u počítačů objevily, programovalo se přímo ve strojovém kódu, tj. většinou v binární, oktálové, nebo hexadecimální reprezentaci instrukcí počítače. Tento způsob programování byl velmi obtížný a nepřehledný a vyhovoval pouze v úplných začátcích, kdy programy byly poměrně krátké a jednoduché.

S rozvojem techniky však bylo třeba najít způsob, který by programování zjednodušil a zrychlil. Vznikly tak první jazyky symbolických adres. U nich zavádíme symbolické označování objektů, s tím, že vazbu symbolů na jejich číselné vyjádření nebude provádět programátor, ale speciální program - překladač (assembler). Symboly jsou nejčastěji tvořeny mnemotechnickými zkratkami slov z běžného jazyka, nejčastěji angličtiny. Program se tak stává čitelnější a přehlednější.

Možnosti jazyků symbolických adres byly dále rozšířeny doplněním nových prostředků (např. makrojazyk, makra). Makrojazyk je založen na možnosti pojmenovat celé posloupnosti instrukcí. Bez makrojazyka bylo možné pojmenovat pouze objekty v instrukcích.

Dalším krokem k tomu, aby se zjednodušila práce programátora, je zbavit jej závislosti na instrukční sadě mikropočítače, se kterou je úzce spjat při programování v jazyce symbolických adres. Při programování mikropočítačů se tak objevují programovací jazyky používané při práci s velkými počítači, jako jsou jazyk C, C++, Pascal, Delphi, apod. Přesto však jazyky symbolických adres nezanikly. Je to způsobeno tím, že některé úlohy nelze ve vyšších programovacích jazycích řešit. Při programování v jazyce symbolických adres se totiž dostáváme do úzkého styku se systémem a s hardwarem. Dalším důvodem je také to, že pokud je kód programu napsán efektivně, může být program v jazyce symbolických adres rychlejší, než stejný program napsaný ve vyšším programovacím jazyce.

Při programování v jazyce symbolických adres vzniká jeden zásadní problém a to, že existuje velké množství různých assemblerů. Jednak se liší instrukční soubory

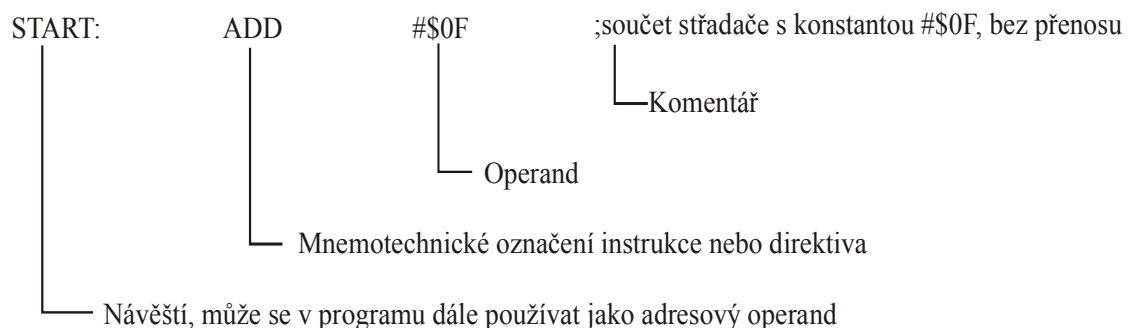
mikropočítačů různých výrobců, ale velmi často také existují rozdíly i mezi assemblyy určenými pro stejný typ mikropočítače. Také je třeba mít na paměti, že programy napsané pro určitý typ mikroprocesoru jsou většinou nepřenositelné. To znamená, že bez úprav je nelze použít pro jiný typ mikropočítače, a to ani tehdy, jedná-li se o mikropočítač pocházející ze stejné typové řady od stejného výrobce. U HC(S)08 se například jedná o názvy jednotlivých registrů periférií, rozsah paměti a periférie samotné. [2]

1.9.1 Programování v assembleru

Základní funkce, které assembler vykonává:

- a) překládá (transformuje) data nebo adresy z různých číselných soustav (např. z desítkové, osmičkové, šestnáctkové) do dvojkové a překládá i znaky do odpovídajících kódů ASCII
- b) umožňuje programátorovi, aby rezervoval a přiděloval paměť pro dočasné uchování dat
- c) umožňuje programátorovi uložit konstanty na určeném místě v paměti
- d) umožňuje programátorovi přiřadit symbolická jména paměťovým místům a vstupně-výstupním registrům
- e) umí vykonat některé aritmetické operace jako součást procesu překladač
- f) informuje programátora o formálních chybách v programu

Zdrojový program, který vytváříme textovým editorem, se skládá z posloupnosti zdrojových řádků. Zdrojový řádek lze zpravidla rozdělit do čtyř polí, např. v následujícím uspořádání:



2 REGULACE A REGULAČNÍ TECHNIKA

2.1 Základní pojmy a definice z regulační techniky

Jedním ze základních pojmů regulační techniky je *regulace*. Regulace je definována jako udržování hodnot regulované veličiny podle daných podmínek a hodnot této veličiny zjištěných měření.

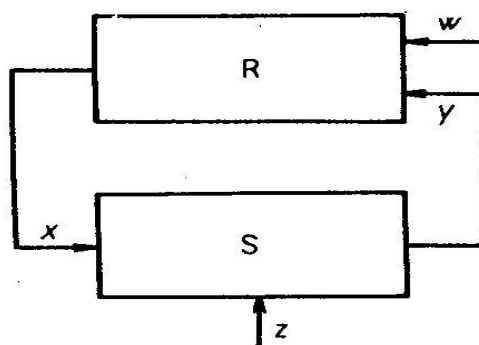
Zařízení, na kterém se regulace provádí, se nazývá *regulovaná soustava*. Regulovanou soustavou může být například nádrž, u které regulujeme výšku hladiny, turbína elektrárny při regulaci otáček, nebo pec s regulací teploty.

Zařízení, které provádí regulaci, je *regulátor*. Regulovaná soustava a regulátor tvoří dohromady *regulační obvod* (Obr. 19).

Veličina, jejíž hodnota je regulací upravována podle stanovených podmínek, se nazývá *regulovaná veličina*. Touto veličinou mohou být teplota, tlak, otáčky, apod. Skutečnou hodnotu regulované veličiny zjišťujeme měření a porovnáváme ji s *žádanou hodnotou*, kterou nastavujeme podle předem stanovených podmínek prostřednictvím *řídící veličiny*. Jestliže se skutečná hodnota a žádaná hodnota regulované veličiny shodují, regulaci nepotřebujeme.

V praxi se však bude skutečná a žádaná hodnota zpravidla lišit. Je to způsobeno tím, že na regulovanou soustavu působí tzv. *poruchová veličina*.

Rozdíl mezi žádanou a skutečnou hodnotou se nazývá *regulační odchylka*. Účelem regulace je, aby tato odchylka dosahovala minimální hodnoty, v ideálním případě aby byla nulová. K tomu potřebujeme ještě jednu veličinu, prostřednictvím které můžeme snadno ovlivňovat regulovanou veličinu. Takovou veličinu nazýváme *akční veličina*. [1]

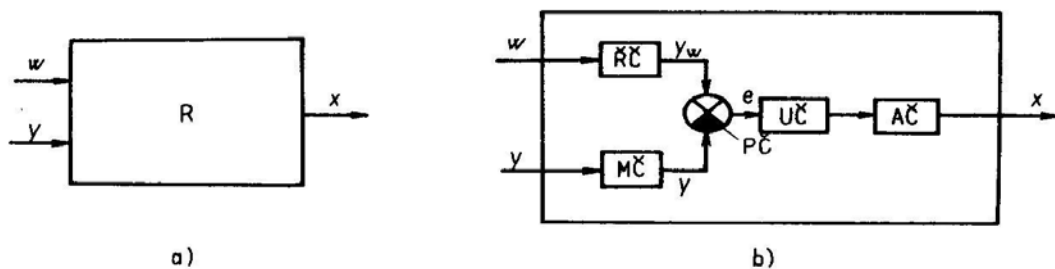


Obr. 19 – Schéma regulačního obvodu

R – regulátor	S – regulovaná soustava
y – regulovaná veličina	w – žádaná hodnota
x – akční veličina	z – poruchová veličina

2.2 Automatická regulace

Samočinné udržování hodnot regulované veličiny podle daných podmínek a hodnot této veličiny zjištěných měření, nazýváme *automatická regulace*. Je to pochod, který probíhá v uzavřeném regulačním obvodu bez zásahu člověka. Na rozdíl od ruční regulace musí tedy člověka, který byl zapojen v regulačním obvodu, nahradit přístroj – *regulátor*. Zpravidla je tvořen *měřicím členem* pro určení skutečné hodnoty regulované veličiny, dále *řídícím členem* pro nastavení žádané hodnoty, *porovnávacím členem*, který porovnává skutečnou a žádanou hodnotu regulované veličiny, *ústředním členem*, který zpracovává regulační odchylku a *akčním členem*. [1]



Obr. 20 – Regulátor a) zjednodušené blokové schéma, b) podrobné blokové schéma

2.2.1 Rozdělení a druhy regulátorů

Regulátory mohou být různého druhu a různého konstrukčního provedení. Jedním z kritérií pro rozdělení regulátorů je, zda se jejich výstupní (akční) veličina mění spojitě nebo nespojitě. Základní rozdělení regulátorů je tedy rozdělení na spojitě a nespojitě regulátory.

Spojitě regulátory jsou charakteristické tím, že všechny jejich členy pracují spojitě. Jejich akční veličina je spojitou funkcí regulační odchylky.

Nespojitě regulátory jsou charakteristické tím, že alespoň jeden člen regulátoru pracuje nespojitě. Existuje nespojitý vztah mezi regulační odchylkou (vstupní veličinou regulátoru) a akční veličinou (výstupní veličinou regulátoru).

Podle jiného hlediska můžeme regulátory rozdělit na přímé a nepřímé.

Přímé regulátory jsou charakteristické tím, že energie odebíraná z jejich snímače stačí na přímé ovládání akčního členu. Tyto regulátory tedy pro svou činnost nepotřebují pomocnou energii.

Nepřímé regulátory se používají v těch případech, kdy energie odebíraná z jejich snímače by nestačila k přímému ovládání akčního členu. Proto je nutné mezi porovnávací a akční člen zařadit zesilovač. Tyto regulátory tedy pro svou činnost potřebují pomocnou energii.

Podle druhu použité pomocné energie můžeme nepřímé regulátory rozdělit na:

- pneumatické regulátory
- hydraulické regulátory
- elektrické regulátory

Podle použití dělíme regulátory na:

- regulátory teploty
- regulátory tlaku
- regulátory množství, atd.

2.3 Nespojité regulace

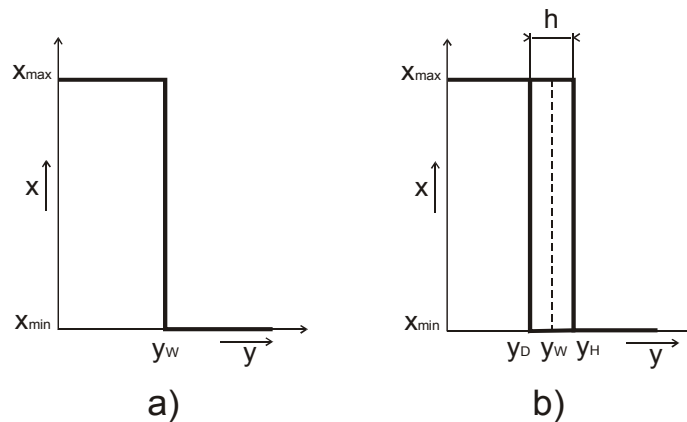
Nespojitý regulátor je charakteristický tím, že jeho výstupní signál (akční veličina) nezávisí spojitě na vstupním signálu (regulované veličině). Akční veličina se tedy nemění spojitě, ale nabývá pouze omezeného počtu hodnot, přičemž změna z jedné hodnoty na druhou probíhá skokem.

Nepojité regulátory patří mezi nejrozšířenější především pro svou jednoduchost a cenovou dostupnost.

2.3.1 Dvoupolohový regulátor

Nejjednodušším nespojitém regulátorem je dvoupolohový regulátor. Poklesne-li skutečná hodnota regulované veličiny y pod žádanou hodnotu y_w , nabude akční veličina určité pevné hodnoty x_{\max} (rozsah akční veličiny). Překročí-li skutečná hodnota regulované veličiny y žádanou hodnotu y_w , nabude akční veličina jiné pevné hodnoty x_{\min} , zpravidla nulové (Obr. 21a).

Při tomto nastavení regulátoru dochází ke spínání velmi často. To způsobuje rychlé opotřebení mechanických částí regulátoru, které vede ke zkrácení jeho životnosti.



Obr. 21 – Regulace a) bez hystereze, b) s hysterezi

Abychom tento problém vyřešili, zavádí se tzv. hystereze (Obr. 21b). To znamená, že regulátor nebude spínat už při dosažení žádané hodnoty y_w , ale zavedeme horní (y_h) a dolní (y_d) hranici hystereze. Ke spínání proto nebude docházet tak často a životnost regulátoru se mnohonásobně zvýší. Nevýhodou hystereze je snížení přesnosti regulace. Pro mnoho aplikací je však tento způsob regulace zcela dostačující. Mezi klasické aplikace patří například udržování teploty v místnostech, kdy kolísání teploty v určitém rozmezí nevadí.

2.4 Spojitá regulace

Nespojité regulátory mají většinou jednoduchou konstrukci a jsou levné, ale jsou nevhodné tím, že neudržují regulovanou veličinu přesně na žádané hodnotě, neboť regulovaná veličina i v bezporuchovém stavu neustále kmitá kolem žádané hodnoty. Je to způsobeno tím, že jejich akční veličina může nabývat pouze několika pevně stanovených hodnot. Tím je dáno, že do soustavy přivádíme střídavě více nebo méně energie nebo látky, než kolik by bylo třeba pro udržení regulované veličiny na žádané hodnotě. Chceme-li odstranit trvalé periodické kmitání regulované veličiny, musíme do soustavy přivést vždy takové množství látky nebo energie, které je potřebné pro udržení regulované veličiny na žádané hodnotě. Musíme tedy mít k dispozici regulátor, u kterého lze měnit hodnotu jeho akční veličiny plynule - *spojitý regulátor*. Výstupní veličina spojitého regulátoru (akční veličina) je spojitou funkcí jeho vstupní veličiny (regulační odchylky). Z toho vyplývá, že regulovaná veličina neustále ovlivňuje akční veličinu, která může nabývat libovolné hodnoty od $x = 0$ až po $x = x_{\max}$.

Regulátory jsou obvykle konstruovány tak, aby bylo možné jejich vlastnosti volit a tím je co nejlépe přizpůsobit dané regulované soustavě. Tato volba vlastností regulátoru spočívá v tom, že můžeme volit závislost mezi vstupní a výstupní veličinou regulátoru. [1]

Rozlišujeme tři základní typy spojitých regulátorů:

2.4.1 Proporcionální regulátor

Nejjednodušší závislost mezi vstupní a výstupní veličinou regulátoru je přímá úměrnost. Regulátor, který v rovnovážném stavu uvedenou závislost splňuje, se nazývá proporcionální regulátor. Jeho rovnice má tvar:

$$x = K_R \cdot e \quad (1)$$

kde K_R je součinitel přenosu regulátoru - zesílení.

Protože platí $e = y_w - y$, znamená to, že regulátor pracuje tak, že roste-li hodnota regulované veličiny, klesá hodnota akční veličiny, a naopak. Pro danou regulovanou soustavu je součinitel přenosu konstantní a nelze jej měnit. U regulátoru (na rozdíl od regulované soustavy) máme možnost součinitel přenosu měnit (můžeme jej nastavovat). Tím je dána i možnost ovlivňovat vlastnosti regulátoru.

Proporcionální regulátor je velmi jednoduchý, levný a stabilní. Je však nevýhodný tím, že pracuje s trvalou regulační odchylkou. K jejímu odstranění musíme použít jiný typ regulátoru.

2.4.2 Integrační regulátor

U integračního regulátoru každé hodnotě vstupní veličiny odpovídá úměrná změna rychlosti výstupní veličiny. Rovnice integračního regulátoru má tvar:

$$x = \frac{K_R}{T_i} \int e dt \quad (2)$$

kde T_i je integrační časová konstanta.

Nejvýznamnější vlastností integračního regulátoru je skutečnost, že pracuje bez trvalé regulační odchylky. Vzhledem ke svému astaticismu není vhodný pro regulaci astatických regulovaných soustav.

2.4.3 Derivační regulátor

U derivačního regulátoru každé změně rychlosti vstupní veličiny odpovídá přímo úměrná hodnota veličiny výstupní. Rovnice takto získaného regulátoru má tvar:

$$x = T_d \cdot K_R \cdot e' \quad (3)$$

kde T_d je derivační časová konstanta.

Derivační regulátor se používá pro zrychlení regulačního pochodu. Vzhledem k tomu, že tento regulátor nereaguje na ustálenou hodnotu regulační odchylky, ale pouze na změnu její rychlosti, neplní hlavní úkol regulátoru, tj. neodstraňuje regulační odchylku, a proto jej nelze použít samostatně. Používá se pouze ve spojení s předcházejícími typy regulátorů.

2.4.4 Sdružené regulátory

Vlastnosti sdružených regulátorů jsou dány součtem vlastností jednoduchých regulátorů. Teoreticky bychom se mohli setkat se čtyřmi druhy těchto regulátorů, v praxi se však používají pouze tři z nich, a to regulátory PI, regulátory PD a regulátory PID. [1]

Sdružené regulátory používáme především v těch případech, kdy regulace pomocí jednoduchých regulátorů nevyhovuje (např. potřebujeme regulovat bez trvalé regulační odchylky astatickou soustavu), nebo v případech, kdy potřebujeme zlepšit vlastnosti regulačního pochodu (např. potřebujeme regulační pochod s nulovým překmitem, regulujeme soustavu s dopravním zpožděním, apod.).

Proporcionálně integrační regulátor (regulátor PI)

Jeho vlastnosti jsou dány součtem vlastností jednoduchých regulátorů P a I. Tomu odpovídá i jeho rovnice:

$$x = K_R e + \frac{K_R}{T_i} \int e dt \quad (4)$$

Do regulačního pochodu zasáhne nejprve proporcionální složka regulátoru a teprve potom složka integrační. Základní vlastností tohoto typu regulátoru je, že pracuje bez trvalé regulační odchylky.

Proporcionálně derivační regulátor (regulátor PD)

Vlastnosti tohoto sdruženého regulátoru jsou dány součtem vlastností jednoduchých regulátorů P a D. Jeho rovnice má tvar:

$$x = K_R e + K_R \cdot T_d \cdot e' \quad (5)$$

Do regulačního pochodu nejprve zasáhne derivační složka regulátoru, která celý regulační pochod urychlí a teprve později se projeví proporcionální složka, která regulační pochod stabilizuje. Tento regulátor však pracuje s trvalou regulační odchylkou.

Proporcionálně integračně derivační regulátor (regulátor PID)

Jeho vlastnosti jsou dány součtem vlastností jednoduchých regulátorů (P, I, D) a tomu odpovídá jeho rovnice:

$$x = K_R e + \frac{K_R}{T_i} \int e dt + K_R T_d e' \quad (6)$$

Do regulačního pochodu nejprve zasáhne derivační složka, později proporcionální složka a teprve na závěr integrační složka. Uvedený sdružený regulátor PID pracuje bez trvalé regulační odchylky.

V dnešní době se stále více uplatňují také nespojitě (číslicové) regulátory P, PD, PS, PSD (S v označení těchto regulátorů znamená sumační). Tyto regulátory nevyhodnocují regulační odchylku trvale, ale periodicky. Mimo okamžiky vzorkování předpokládáme, že se regulovaná veličina nemění a regulační odchylka je konstantní.

Vlastnosti regulačního pochodu velmi ovlivňuje délka periody vzorkování. Její volba je poměrně obtížná a závisí na vlastnostech a typu regulované soustavy.

Nespojitě regulátory jsou vhodné, pokud k regulaci používáme číslicovou techniku. Regulace zpravidla probíhá tak, že se regulovaná veličina převede v A/D převodníku na číselnou hodnotu, která je porovnána se žádanou hodnotou (ta je nejčastěji zadána přímo v číselné podobě). Vypočítanou regulační odchylku dále zpracuje procesorová jednotka podle příslušného algoritmu, který je uložen v paměti mikropočítače. Vypočítaný akční zásah je poté převeden v D/A převodníku na analogovou veličinu, která řídí akční prvek.

3 MĚŘENÍ TEPLOTY

Měření teploty je jednou z nejčastěji prováděných operací v oblasti automatizace a regulační techniky. V případě této bakalářské práce je základním úkonem, bez kterého by ovládání tepelné soustavy nebylo vůbec možné.

Základní jednotkou termodynamické teploty je Kelvin [K], který je definován jako $1/273,16$ část termodynamické teploty trojného bodu vody. V praxi však častěji používáme stupeň Celsia [$^{\circ}\text{C}$]. Převod je velice jednoduchý:

$$\vartheta[^{\circ}\text{C}] = T[\text{K}] - T_0[\text{K}] \quad (7)$$

kde $T_0 = 273,15 \text{ K}$ je teplota tání vody. Ke stanovení teploty využíváme teplotní závislosti fyzikálních veličin na teplotě.

3.1 Rozdělení senzorů teploty

Měření teploty se provádí pomocí zařízení založených na fyzikálním principu závislosti fyzikální veličiny na teplotě. Tato zařízení nazýváme senzory a můžeme je rozdělit na:

- dotykové
 - elektrické
 - dilatační
 - speciální
- bezdotykové
 - tepelné
 - kvantové

Dilatační senzory jsou založeny na změně objemu látek vlivem teploty. Měření teploty se pak převádí na měření délky, objemu nebo tlaku.

Mezi **speciální senzory** patří např. senzory využívající teplotní změny rychlosti zvuku v ideálním plynu, teplotní závislosti dielektrika, magnetické susceptibility (vhodné pro extrémně nízké teploty – od jednotek mK až po cca 20K), dále senzory založené na generování šumového napětí, apod.

Mezi nejpoužívanější patří **elektrické dotykové senzory**.

3.2 Rozdělení elektrických dotykových senzorů teploty

Elektrické dotykové senzory můžeme podle principu jejich funkce rozdělit na :

- kovové odporové senzory teploty
- polovodičové odporové senzory teploty
 - termistory
 - NTC
 - PTC
 - monokrystalické odporové senzory
- monokrystalické PN senzory teploty
- termoelektrické senzory
- alternativní senzory teploty a převodníky teplota => střída nebo digitální slovo

Teplotní odporové senzory využívají fyzikální princip závislosti elektrického odporu kovů nebo polovodičových materiálů na teplotě. Vstupním podnětem je teplota prostředí, které je v přímém dotyku se senzorem. Senzor je tak součástí procesu přenosu tepla vedením, prouděním nebo sáláním z vnějšího prostředí do materiálu senzoru.

Materiálem pro výrobu teplotních senzorů je vybraný kovový nebo polovodičový materiál. Pro správnou funkci musí mít tento materiál správné složení, především musí být fyzikálně a chemicky stálý. [3]

3.2.1 Odporové kovové senzory teploty

Atomy krystalové mřížky kovu s rostoucí teplotou zvyšují amplitudu svých kmitů a kladou tak větší odpor průchodu elektronů. To je zjednodušená představa závislosti odporu kovů na teplotě, která je principem těchto senzorů. Závislost lze v rozsahu teplot 0 °C až 100 °C vyjádřit přibližným vztahem:

$$R = R_0 \cdot (1 + \alpha \cdot \vartheta) \quad (8)$$

kde α [K⁻¹] je teplotní součinitel odporu a R_0 odpor při teplotě 0 °C.

Pro kovové senzory teploty se většinou využívá platina pro její chemickou netečnost, časovou stálost, vysokou teplotu tavení a také pro možnost dosažení vysoké

čistoty (pohybuje se v rozmezí 99,9 až 99,999%). Kromě platiny se používá také nikl, měď, molybden nebo některé slitiny.

Bližší informace k odporovým kovovým sensorům teploty lze nalézt např. v [3],[12]

3.2.2 Polovodičové odporové senzory teploty

U polovodičových sensorů teploty se podobně jako u kovových využívá teplotní závislosti odporu na teplotě. Při teplotě absolutní nuly jsou všechny elektrony pevně vázány ke svým jádrům a materiál nemůže vést proud. Elektronům je třeba dodat určitou energii k tomu, aby „přeskočily“ přes tzv. zakázaný pás do pásu vodivostního a mohly se účastnit vedení proudu. Touto energií může být např. energie elektromagnetického pole nebo energie tepelná. S rostoucí teplotou tak bude koncentrace nosičů náboje růst a elektrický odpor materiálu se bude snižovat. Zatímco se tento jev snažíme u klasických polovodičových součástek potlačit, u termistorů se ho naopak snažíme vhodnou technologií a složením zvýraznit. [13]

Termistory NTC (negastory)

Termistory NTC (Negative Temperature Coefficient) mají záporný teplotní součinitel odporu. S rostoucí teplotou se zvyšuje koncentrace nosičů náboje a elektrický odpor klesá. Polykrystalické NTC termistory se vyrábí práškovou technologií spékáním oxidů Fe_2O_3 , TiO_2 , CuO , MnO , NiO , CoO , BaO , apod. Kromě termistorů pro běžné teplotní rozsahy -50°C až 150°C se vyrábí i speciální termistory pro nízké (od cca -269°C) a vysoké teploty (zhruba do 1000°C). Závislost odporu termistoru na teplotě je silně nelineární a lze ji vyjádřit přibližným vztahem:

$$R = A \cdot e^{\frac{B}{T}} \quad (9)$$

kde R [Ω] je odpor termistoru, T [K] je teplota, B [K] je teplotní konstanta, která závisí na materiálu a je úměrná aktivační energii, A [Ω] je konstanta závislá na geometrickém tvaru a materiálu senzoru.

Charakteristické vlastnosti termistorů NTC

- Teplotní součinitel odporu je záporný a zhruba o řád vyšší než u kovů (obvykle nabývá hodnot $-0,03 \text{ K}^{-1}$ až $-0,06 \text{ K}^{-1}$)

- Běžný rozsah teplot je $-50\text{ }^{\circ}\text{C}$ až $150\text{ }^{\circ}\text{C}$ (méně často $-100\text{ }^{\circ}\text{C}$ až $300\text{ }^{\circ}\text{C}$, výjimečně do $400\text{ }^{\circ}\text{C}$)
- Široké rozmezí hodnot odporu (od $0,1\ \Omega$ až po jednotky $\text{M}\Omega$). Oproti platinovým sensorům mají obvykle mnohem větší odpor a větší citlivost, což přináší méně problémů v souvislosti s teplotně závislými přírůdky
- Velká citlivost, malá hmotnost a rozměry umožňují měřit i velmi rychlé teplotní změny
- V porovnání s platinovými senzory jsou NTC termistory méně stabilní a časově nestálé
- Zásadní nevýhodou je značná nelinearita

Termistory PTC (pozistory)

Termistory PTC (Positive Temperature Coefficient) mají na rozdíl od NTC termistorů kladný teplotní koeficient a dále vykazují silnou nelineární závislost odporu na teplotě. Jejich odpor s rostoucí teplotou nejprve mírně klesá a po překročení tzv. Curieovy teploty strmě roste. Po nárůstu zhruba o tři řády hodnota odporu opět mírně klesá (s touto částí charakteristiky se však již obvykle neparuje).

Termistory PTC se pro výrobu odporových teplotních sensorů prakticky nepoužívají. Využívají se především v oblastech, ve kterých můžeme využít jejich V-A charakteristiky, jako jsou např. vratné tepelné pojistky, dvoupolohová regulace, termostaty, měření výšky hladiny, apod.

3.2.3 Monokrystalické senzory

Polovodičové monokrystalické senzory teploty lze vyrobit z křemíku, germania nebo india, v praxi se však setkáváme pouze se senzory křemíkovými.

Monokrystalické Si senzory

Pro výrobu křemíkových sensorů se používá nevlastního polovodiče typu N, tedy s dominantní elektronovou vodivostí. Pohyblivost volných nosičů náboje v krystalové mřížce křemíku závisí na teplotě a na počtu příměsí v jednotce objemu. S rostoucí teplotou dochází vlivem rozptylu nosičů náboje na mřížce polovodiče ke zmenšování pohyblivosti těchto nosičů, v důsledku čehož narůstá rezistivita, podobně jako je tomu u kovů. Monokrystalické Si senzory teploty tedy mají kladný teplotní součinitel odporu stejně jako

PTC termistory, princip jejich vodivosti je však odlišný. Křemíkové senzory se obvykle používají pro rozsah teplot -50 až 150 °C. [13]

Charakteristické vlastnosti monokrystalických Si senzorů

- Teplotní součinitel odporu je téměř konstantní v celém rozsahu teplot a jeho střední hodnota se pohybuje kolem $0,01$ K⁻¹ (platinové senzory: $0,004$ K⁻¹, NTC: v rozsahu cca $-0,03$ K⁻¹ až $-0,06$ K⁻¹)
- Dlouhodobá stabilita. Teplotní drift kolem $0,2$ K po 10000 hodinách nepřetržitého provozu při maximální provozní teplotě
- Linearita je lepší než u NTC termistorů, ale horší než u platinových senzorů, nelinearitu lze však vhodnými metodami úspěšně korigovat
- Teplotní rozsah je obvykle -55 až 150 °C, k dostání jsou však běžně i senzory s horní teplotní hranicí 300 °C
- Referenční hodnota odporu při teplotě 25 °C je obvykle 1000 nebo 2000 Ω

3.2.4 Teplotní čidla s digitálním výstupem

Poměrně novým směrem v oblasti teplotních čidel jsou čidla s digitálním výstupem (teplota=>střída, teplota=>frekvence).

Zde je velmi důležité si uvědomit, že se nejedná o žádná speciální čidla, ale o součástky, obsahující teplotně závislý prvek, který pracuje na některém z výše uvedených fyzikálních principů, doplněný dalšími elektronickými obvody. Tyto obvody slouží k vyhodnocování teplotní závislosti senzoru a jejímu převodu na digitální signál PWM, kódové slovo nebo na frekvenční závislost. Základní výhodou je skutečnost, že teplotní čidlo a převodník jsou umístěny ve společném pouzdře. Odpadají tak problémy s návrhem zapojení a s následnou kalibrací.

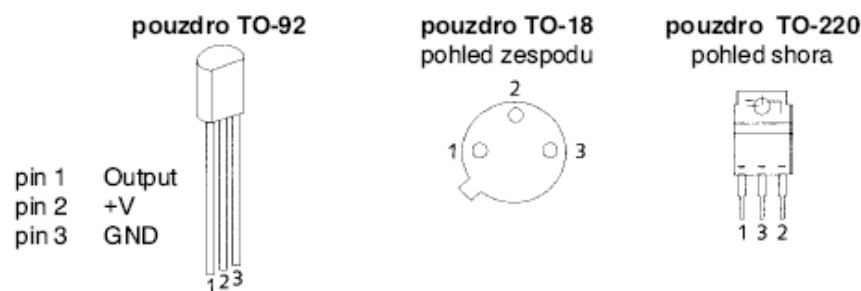
Mezi tato moderní teplotní čidla patří také teplotní senzor SMT 160-30 firmy SMARTEC. Hlavní předností senzoru SMT 160-30 je šířkově modulovaný výstupní signál, což umožňuje přímé připojení senzoru k mikroprocesoru a jednoduché vyhodnocení teploty bez potřeby použití A/D převodníku.

K jednomu mikroprocesoru lze multiplexováním připojit několik těchto čidel. Použitím jednoduchého algoritmu vyhodnocování signálu je možno získat přímo absolutní hodnotu měřené teploty.

Senzor nevyžaduje žádnou kalibraci, neboť je kalibrován přímo na čipu během výroby. Výběr mezi čtyřmi základními pouzdry běžných typů poskytuje široké možnosti při konstrukčním řešení umístění senzoru. Rovněž třídrátové připojení téměř libovolným vodičem na vzdálenost až 20 m neklade žádné konstrukční omezení při aplikaci. [11]

Charakteristické vlastnosti

- Měření teploty bez nutnosti použití A/D převodníku, výstupní signál s pulsně šířkovou modulací (PWM)
- Přímé připojení k mikroprocesoru, kompatibilní s TTL a CMOS obvody
- Výstupní signál je přímo zpracovatelný buď analogově nebo digitálně
- Široký teplotní rozsah od -45°C do $+130^{\circ}\text{C}$
- Absolutní přesnost $\pm 0,7^{\circ}\text{C}$, nelinearita maximálně $0,2^{\circ}\text{C}$
- Bez potřeby kalibrace, senzor je kalibrován při výrobě na čipu
- Napájecí napětí $+5\text{V}$, velmi nízká spotřeba $<1\text{mW}$
- K dispozici pouzdra TO-92, TO-18, TO-220, SOIC-8L (SMD), hybridní pouzdro



Obr. 22 – Teplotní čidlo SMT 160-30

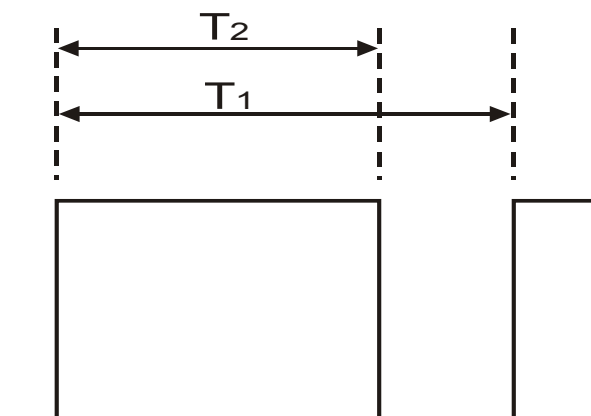
Výstupní signál

Výstupní signál má pravoúhlý průběh s přesně definovanou závislostí střídý impulsního signálu (DC) na teplotě. Střída signálu ($DC = T_2/T_1$, viz. Obr. 23) je lineárně závislá na teplotě podle vztahu:

$$DC = 0,320 + 0,0047 \cdot t \quad (10)$$

kde: t ... teplota ve $[\text{°C}]$

DC ... Duty Cycle - poměr doby, kdy je signál v úrovni log 1 (T_2 - šířka impulsu) k celkové délce periody (T_1) výstupního signálu



Obr. 23 – Výstupní signál čidla SMT 160-30

Úpravou vztahu (10) získáme vzorec pro výpočet teploty ve [°C].

$$t = 212,7660 \cdot DC - 68,0851 \quad (11)$$

Popis čidla SMT 160-30

Obvod SMT 160-30 je bipolární senzor teploty (základem je polovodičový přechod PN polarizovaný v propustném směru), doplněný elektronickými obvody pro převod signálu senzoru na signál PWM (tj. modulaci střídý impulsního signálu). Základní elektrické vlastnosti čidla SMT 160-30 jsou popsány v tab. 1. Další informace lze nalézt např. v [11].

Tab. 1 - Všeobecné technické parametry čidla SMT 160-30:

Parametr:	min.	typ.	max.	jednotky
Napájecí napětí	4,75	5	7	V
Vstupní proud	0,16		0,2	mA
Teplotní rozsah	-45		+130	° C
Dlouhodobý drift			0,1	° C
Citlivost na změnu napětí			0,1	° C/V
Výstupní frekvence	1		4	kHz
Impedance			200	Ohm
Šum			0,005	° C

II. PRAKTICKÁ ČÁST

4 MODEL TEPELNÉ SOUSTAVY ŘÍZENÉ MIKROPOČÍTAČEM

Cílem této bakalářské práce je navrhnout a sestavit funkční model tepelné soustavy řízené mikropočítačem, který bude sloužit jako výuková pomůcka pro programování mikropočítačů HCS08.

Tento model je jedním ze série projektů, které budou v rámci bakalářských prací realizovány. Tuto skutečnost je třeba brát v úvahu především při návrhu plošného spoje, protože rozměry modelu a způsob jeho propojení s vývojovým kitem jsou dány předem.

Mikropočítač bude umožňovat nastavení žádané teploty, vyhodnocovat výstupní signál teplotního čidla a převádět jej na okamžitou teplotu povrchu topného rezistoru. Dále bude porovnáváním naměřené a nastavené teploty určovat regulační odchylku a podle její hodnoty pak generovat příslušný akční zásah - v případě tohoto projektu bude akční zásah spočívat v zapnutí nebo vypnutí topení.

Řešení je možné rozdělit na několik dílčích kroků:

- volba vhodného tepelného zdroje, který nebude příliš energeticky náročný a jehož praktická realizace bude co nejjednodušší.
Jako nejvhodnější se jeví drátový rezistor v keramickém pouzdře určený pro jmenovitý výkon 5W. Výhodou tohoto rezistoru je to, že teploty povrchu přibližně 120°C dosáhneme už výkonem nižším než 3W.
- volba vyhovujícího teplotního čidla s dostatečnou přesností (chyba v rozmezí $\pm 1-2\%$), pro teplotní rozsah přibližně 0°C až 120°C.
Jako vyhovující se jeví polovodičové teplotní čidlo SMT 160 s digitálním výstupem, určené pro teplotní rozsah -40°C až 130°C. Výhodou tohoto čidla je, že nepotřebujeme A/D převodník, čidlo není třeba kalibrovat a počet pasivních a aktivních součástek pro jeho zapojení je minimální. Nevýhodou jsou vyšší pořizovací náklady.
- volba způsobu ovládnání tepelného zdroje.
Jako nejvhodnější se jeví spínací tranzistor NPN v zapojení se společným emitorem. Další možnosti jako mechanické relé nebo tyristor nejsou pro účely tohoto projektu příliš vhodné. Tyristor je určen spíše pro střídavé napájení a vyšší výkony a mechanické relé má omezenou životnost a spolehlivost kontaktů.

- návrh zapojení obvodů pro ovládání topení, měření teploty a signalizaci, návrh desky plošných spojů.
- vytvoření obslužného programového vybavení v jazyce symbolických adres.
- osazení desky plošných spojů a testování prototypu

4.1 Princip činnosti modelu tepelné soustavy

Princip činnosti je poměrně jednoduchý. Teplotní čidlo, umístěné v blízkosti topného rezistoru, vyhodnocuje teplotu jeho povrchu. Pro lepší tepelný kontakt je použita teplovodná pasta s oxidem zinečnatým. Výstupní signál tohoto čidla je vyhodnocován pomocí mikropočítače HCS08. K jeho zpracování jsou použity dvě metody:

- a) pomocí časovače TPM2 mikropočítače HCS08, jehož kanály 0 a 2 pracují v režimu Input Capture
- b) pomocí vzorkování výstupního signálu čidla SMT 160

Z naměřených hodnot T_1 (délka periody) a T_2 (doba, po kterou je výstupní signál ve stavu log 1) se následně pomocí přepočtového vzorce (11) vypočítá skutečná teplota povrchu rezistoru. V dalším kroku je třeba porovnáním naměřené a nastavené teploty určit regulační odchylku, která je následně zpracována pomocí příslušného algoritmu. Tímto způsobem se určí, zda je třeba topení zapnout, nebo vypnout.

Algoritmus pro zpracování regulační odchylky je záležitostí programátora. V ukázkovém programu pro obsluhu tohoto modelu je realizován algoritmus pro simulaci dvoupolohového regulátoru bez hystereze. Jak bude dokázáno dále, přestože regulátor nemá nastavenou žádnou hysterezi, teplota soustavy bude kmitat okolo nastavené (žádané) teploty. Je to způsobeno jednak vlastní kapacitou soustavy, vlastnostmi pouzdra teplotního čidla SMT 160 a také nedokonalým kontaktem mezi čidlem a topným rezistorem..

4.2 Popis hardware vytvořeného modulu

Model tepelné soustavy se skládá ze tří hlavních částí:

- napájení
- výkonová část
- měřící část

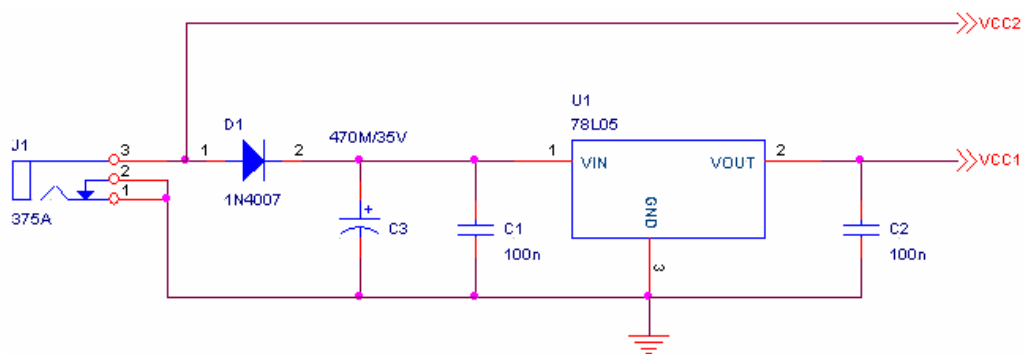
4.2.1 Napájení

Napájecí obvod poskytuje stabilizované napětí +5V pro teplotní čidlo SMT 160 a dále nestabilizované napětí +9V pro napájení výkonové části tepelné soustavy.

Vstup je tvořen souosým konektorem J1 a je chráněn proti přepólování běžnou usměrňovací diodou D1 (např. typ 1N4007).

Stabilizace napětí je provedena obvodem 78L05 ve standardním zapojení. Obvod 78L05 poskytuje napětí v rozmezí 4,6 - 5,4V a jeho výstup lze zatížit trvalým proudem maximálně 150mA. Napájecí napětí může být maximálně 30V, a aby byly zachovány stabilizační vlastnosti obvodu 78L05, nesmí napětí poklesnout pod 5V.

U nestabilizovaného napětí +9V je nutné, aby byl jeho pokles minimální. Proto jej odebíráme už před diodou D1. Vlastní ochrana proti přepólování a přetížení je provedena až ve výkonové části.



Obr. 24 – Napájecí část obvodu

Napájení celého modulu tepelné soustavy je realizováno vnějším zdrojem DC +9V/1100mA, který je primárně určen pro napájení celého vývojového kitu a je součástí originálního balení. Napájecí napětí je odebíráno z výstupů +VIN a GND svorkovnice TB1 (kapitola 1.3.4). Pokud to bude uživatel považovat za nutné, lze použít jakýkoliv vnější zdroj DC s výstupním napětím nižším než 11V.

4.2.2 Výkonová část modelu

Výkonová část obvodu je tvořena darlingtonovým tranzistorem Q2 (typ BD679) v zapojení se společným emitorem, který slouží jako bezkontaktní spínač. Do kolektorového obvodu je zapojen drátový rezistor R4 v keramickém pouzdře, který slouží jako tepelný zdroj.

Tranzistor je spínán výstupem mikropočítače přes omezující rezistor R3. Z tohoto výstupu je současně také napájena červená signalizační LED dioda D3. Přesto je, jak bude dokázáno dále, z výstupu mikropočítače při plném provozu odebírán proud menší než 2 mA. Je tak splněna podmínka, aby výstup mikropočítače nebyl namáhán proudem vyšším než 10 mA.

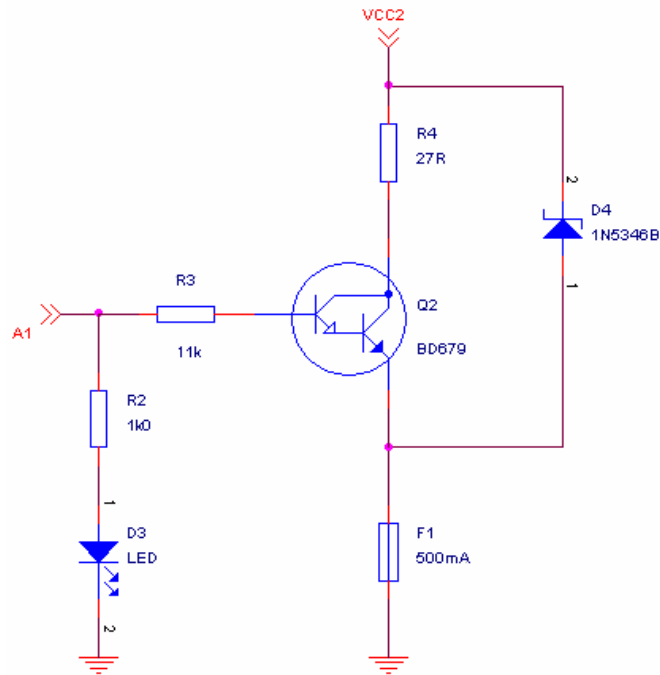
Tranzistor v darlingtonově zapojení představuje dvojici vhodně zapojených NPN nebo PNP tranzistorů, které jsou umístěny ve společném pouzdře. První tranzistor slouží jako budící obvod pro druhý, výkonový tranzistor. Tento typ tranzistoru se nejčastěji uplatňuje v koncových stupních výkonových zesilovačů, kde využíváme především jeho vysokou výkonovou zatížitelnost a malé zkreslení výstupního signálu.

V této aplikaci však využíváme další vlastnosti těchto součástek, a to vysokého proudového zesílení v zapojení se společným emitorem. Tranzistor tak můžeme maximálně otevřít už poměrně malým bázevým proudem a nemusíme zbytečně zatěžovat výstup mikropočítače.

Při běžném použití tohoto typu tranzistorů je určitou nevýhodou to, že tranzistor potřebuje poměrně vysoké vstupní napětí U_{BE} , které dosahuje přibližně dvojnásobku prahového napětí křemíkové diody, tedy asi 1,28V. V této aplikaci však tato vlastnost nevádí a představuje spíše výhodu, protože činí tranzistor odolnějším proti případnému kolísání napětí na výstupu mikropočítače ve stavu log 0.

Další nevýhodou těchto tranzistorů je značná citlivost na rušivá napětí na vstupu. Je to způsobeno vysokou hodnotou zesilovacího činitele. Protože však v tomto zapojení pracuje tranzistor pouze ve spínacím režimu, nepředstavuje tato citlivost téměř žádný problém. Pokud bychom však chtěli řídit proud topným prvkem, např. pomocí analogového výstupu, bylo by nutné použít jiný spínací prvek. Pro vyšší výkony představuje vhodný způsob např. použití tyristoru. Topný prvek v takovém případě budeme napájet střídavým napětím a k ovládní výkonu bude využita např. metoda spínání v nule.

Tranzistor a topný rezistor jsou chráněny proti přepólování nebo připojení napětí vyššího než 11V Zenerovou diodou D4 (typ 1N5346B, $U_Z=11V$) a tavnou pojistkou F1. Při překročení napájecího napětí nebo při přepólování se dioda plně otevře a dojde tak k přepálení pojistky.

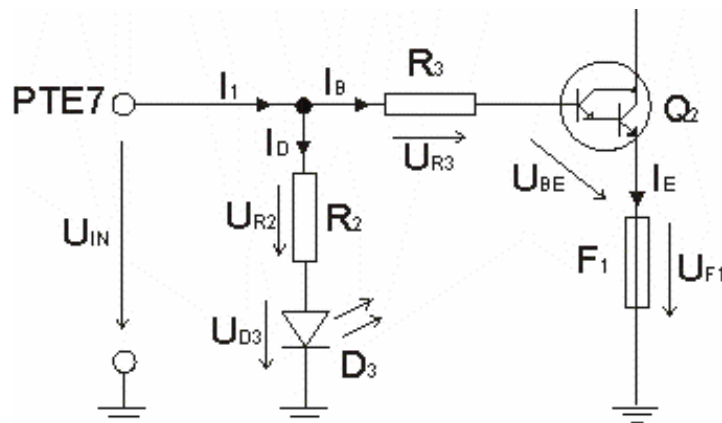


Obr. 25 – Výkonová část obvodu

Výpočetní část

Vstupní obvod tranzistoru

Na (Obr. 26) a (Obr. 27) jsou znázorněny proudy a napětí, které se vyskytují ve výkonové části tohoto modelu. Účelem je, aby vznikla představa o reálných napěťových a proudových poměrech v tomto obvodu a byl srozumitelný postup při výpočtu.



Obr. 26 – Vstupní obvod tranzistoru

Jako signalizace je použita červená LED dioda o průměru 3mm. Maximální proud, který může procházet diodou tohoto typu je 2mA. Při tomto proudu je v optimálním poměru životnost a svítivost LED diody. Pokud bychom volili proud vyšší, svítivost diody by sice vzrostla, ale její životnost by se neúměrně zkrátila.

Naměřené a zvolené hodnoty

$$U_{IN} = 3,3V \quad U_{BE} \approx 1,28V$$

$$U_D \approx 1,69V \quad I_{Dmax} = 2mA$$

$$U_{F1} = 0,5V$$

Ze známého napětí U_{IN} na výstupu PTF7 mikropočítače HCS08 a změřeného úbytku napětí na diodě U_{D3} spočítáme napěťovou ztrátu U_{R2} na předřadném rezistoru R_2 .

$$U_{IN} = U_{R2} + U_{D3} \quad \Rightarrow \quad U_{R2} = U_{IN} - U_{D3} = 3,3 - 1,69 = 1,61V \quad (1,57V)$$

Z vypočítaného napětí U_{R2} a maximálního proudu diodou I_{Dmax} spočítáme minimální hodnotu předřadného odporu R_2 .

$$U_{R2} = R_2 \cdot I_D \quad \Rightarrow \quad R_2 = \frac{U_{R2}}{I_D} = \frac{1,61}{2 \cdot 10^{-3}} = 805\Omega$$

Abychom mírně snížili zatížení výstupu mikropočítače a prodloužili životnost LED diody, můžeme zvolit vyšší hodnotu odporu rezistoru R_2 . Volíme proto $R_2 = 1k\Omega$.

Abychom vypočítali nový proud diodou $D3$, budeme pro jednoduchost předpokládat, že napětí na rezistoru R_2 zůstane při poklesu proudu diodou konstantní. Tento předpoklad vychází z toho, že prahové napětí LED diody je dáno materiálem, ze kterého je dioda vyrobena, a že změna napětí na plně otevřené diodě je při poklesu proudu velmi malá, prakticky zanedbatelná. Pro ilustraci, rozdíl mezi prahovým napětím diody a napětím na diodě, kterou protéká proud 2mA, je nižší než 0,1V.

$$I_D = \frac{U_{R2}}{R_2} = \frac{1,61}{1 \cdot 10^3} = 1,61mA$$

Při tomto proudu je svítivost červené LED diody dostatečná a pro účely signalizace zcela vyhovující.

Z napětí na výstupu mikropočítače U_{IN} , napětí U_{BE} na vstupu spínacího tranzistoru a úbytku napětí na pojistce F_1 spočítáme napěťovou ztrátu na bázevém rezistoru R_3 .

$$U_{IN} = U_{R_3} + U_{BE} + U_{F_1} \quad \Rightarrow$$

$$U_{R_3} = U_{IN} - U_{BE} - U_{F_1} = 3,3 - 1,28 - 0,5 = 1,52V \quad (1,50V)$$

Rezistor R_3 volíme tak, aby nebyl příliš zatížen výstup mikropočítače a současně aby byl spínací tranzistor plně otevřen.

$$R_3 = 11k\Omega$$

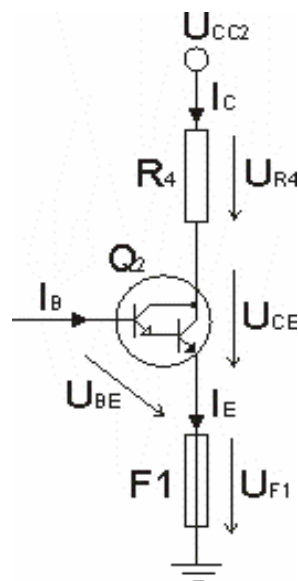
Z vypočítaného napětí na rezistoru R_3 a zvolené hodnoty rezistoru R_3 spočítáme proud báze I_B .

$$U_{IN} = I_B \cdot R_3 + U_{BE} + U_{F_1} \quad \Rightarrow \quad I_B = \frac{U_{IN} - U_{BE} - U_{F_1}}{R_3} = \frac{U_{R_3}}{R_3} = \frac{1,52}{11 \cdot 10^3} = 0,138mA$$

Z hodnoty proudu diodou I_D a bázevého proudu I_B spočítáme celkový proud I_1 , kterým je zatěžován výstup mikropočítače.

$$I_1 = I_D + I_B = 1,57 + 0,138 = 1,708mA$$

Výstupní obvod tranzistoru



Obr. 27 – Výstupní obvod tranzistoru

Naměřené a zvolené hodnoty

$$U_{CC2} = 9V \quad U_{CE} = 0,75V$$

$$R_4 = 27\Omega \quad R_{F1} = 1,7\Omega$$

Úbytek napětí na topném rezistoru R4 a pojistce F1 spočítáme z napětí zatíženého napájecího zdroje U_{CC2} a úbytku napětí U_{CE} na tranzistoru Q2. Protože by se tyto dvě hodnoty odvozovaly poněkud obtížně, byly pro jednoduchost naměřeny během provozu.

$$U_{CC2} = U_{CE} + U_{R4} + U_{F1} \quad \Rightarrow \quad U_{R4} + U_{F1} = U_{CC2} - U_{CE} = 9 - 0,75 = 8,25V \quad (8,31V)$$

Kolektorový proud I_C spočítáme z úbytků napětí U_{R4} a U_{F1} na rezistoru R4 a pojistce F1.

$$I_C = \frac{U_{R4} + U_{F1}}{R_4 + R_{F1}} = \frac{8,25}{27 + 1,7} = 0,2874A \quad (0,28A)$$

Aby byl výpočet proudu I_C co nejjednodušší, zavádíme zjednodušující předpoklad, že úbytek napětí na pojistce F1 je vyvolán pouze průchodem kolektorového proudu I_C . Ve skutečnosti je však úbytek napětí vyvolán emitorovým proudem I_E . Protože však platí, že $I_E = (h_{21E} + 1) \cdot I_B$, dá se pro hodnotu $h_{21E} > 100$ předpokládat, že $I_C \approx I_E$. Oprávněnost tohoto předpokladu je dokázána dále.

Nyní spočítáme proud emitorem I_E .

$$I_E = I_C + I_B = 0,2874 + 0,000138 \approx 0,2875A$$

Protože odpor rezistoru R4 během provozu mírně narůstá, emitorový proud by při bezporuchovém provozu neměl nikdy vzrůst nad hodnotu $I_E = 0,3A$. Proto volíme pojistku 500mA.

Z hodnoty proudu I_B a I_C můžeme spočítat proudový zesilovací činitel h_{21E} .

$$h_{21E} = \frac{I_C}{I_B} = \frac{0,2874}{0,138 \cdot 10^{-3}} \approx 2083$$

Takto vysoký zesilovací činitel je dán použitým typem tranzistoru. Darlingtonovy tranzistory mohou mít proudový zesilovací činitel až 10000.

Pokud bychom použili běžný tranzistor, museli bychom čelit značnému poklesu proudového zesilovacího činitele v oblasti mezní přímky. Tento pokles bychom pak museli kompenzovat zvýšením bázevého proudu a došlo by tak ke zbytečnému zatížení výstupu mikropočítače.

Z hodnoty odporu rezistoru R_4 a proudu I_C spočítáme výkon P_{R4} na topném rezistoru R_4 .

$$P_{R4} = R_4 \cdot I_C^2 = 27 \cdot 0,2874^2 = 2,23W$$

Z napětí U_{CE} a proudu I_C spočítáme výkonovou ztrátu P_{Q2} na spínacím tranzistoru.

$$P_{Q2} = U_{CE} \cdot I_C = 0,75 \cdot 0,2874 = 0,216W$$

Z hodnoty odporu pojistky $F1$ a proudu I_E spočítáme ztrátový výkon P_{F1} .

$$P_{F1} = R_{F1} \cdot I_E^2 = 27 \cdot 0,2876^2 = 0,141W$$

Účinnost topení.

$$\eta = \frac{P_{R4}}{Q} = \frac{P_{R4}}{P_{R4} + P_{Q2} + P_{F1}} = \frac{2,53}{2,53 + 0,23 + 0,141} = 0,872 = 87,2\%$$

Všechny vypočítané hodnoty napětí a impedancí byly ověřeny pomocí digitálního multimetru RANGE RE890C+ se vstupní impedancí $10M\Omega/50pF$ a přesností měření $\pm 1\%$.

Naměřené hodnoty (viz. hodnoty uvedené v závorce vedle hodnoty vypočítané) se od vypočítaných hodnot lišily pouze minimálně a rozdíl dosahoval maximálně $\pm 2,5\%$.

I přes velmi uspokojivé výsledky musím zdůraznit, že provedené výpočty jsou pouze orientační a postup při některých výpočtech je maximálně zjednodušený.

Topné těleso

Jako topné těleso je v tomto modelu použit drátový rezistor určený pro jmenovité výkonové zatížení $5W$ při okolní teplotě $40^\circ C$. Je umístěn v obdélníkovém keramickém pouzdře o rozměrech $9,5 \times 9 \times 22$ mm (WxHxL).

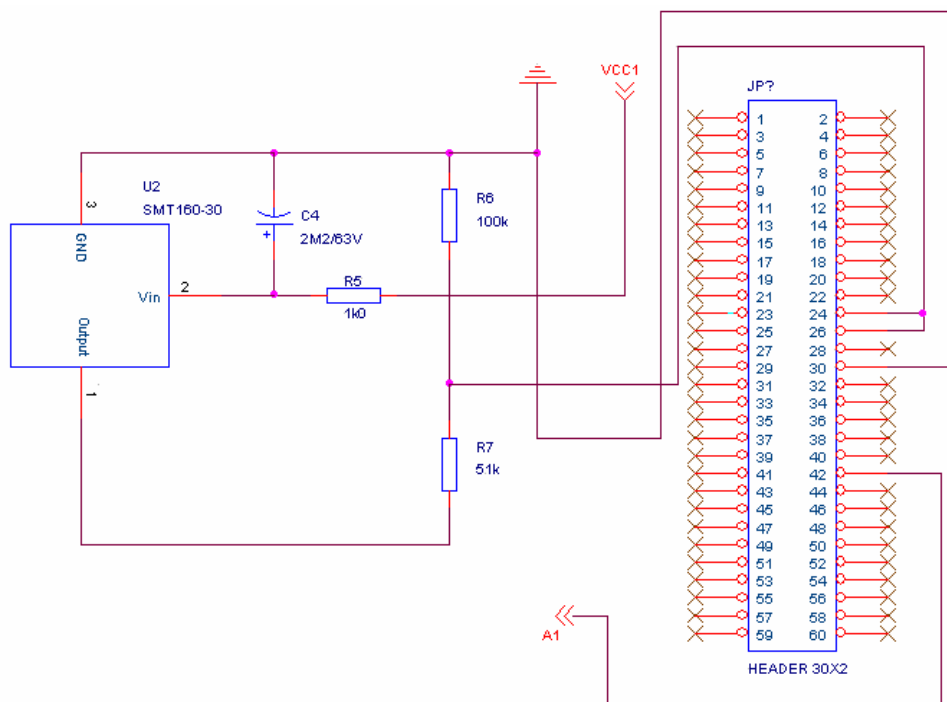
Tyto rezistory se vyrábějí v řadě E12 v rozsahu hodnot $0,1\Omega$ až $15k\Omega$ a jsou určeny pro maximální trvalé napětí 350V.

Výhodou použití toho rezistoru jako topení je značná jednoduchost konstrukce, relativně nízký teplotní koeficient ($\pm 300\text{ppm/K}$) a nízké pořizovací náklady.

4.2.3 Teplotní čidlo SMT 160

Teplotní čidlo SMT 160 má na vstup připojen RC člunek (součástky C4 a R5), který jej chrání proti poškození při náhodném přepólování. V běžném provozu by se však opačné napětí nemělo na vstup čidla nikdy dostat, protože napájení je chráněno diodou D1 (kapitola 4.2.1). RC člunek je zde tedy použit pouze pro jistotu a bylo by možné jej vynechat.

Při použití teplotního čidla SMT 160 musíme mít na paměti, že se jedná o složitou polovodičovou strukturu, která je méně odolná proti mechanickému a elektrickému namáhání než běžná kovová nebo polovodičová odporová teplotní čidla.

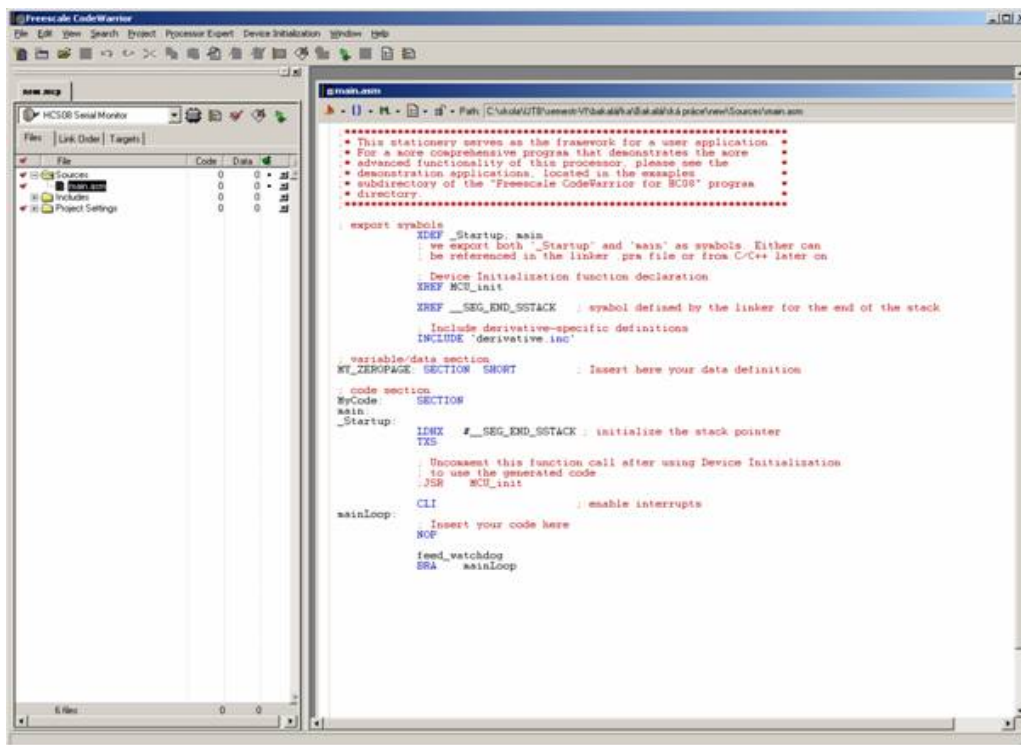


Obr. 28 – Teplotní čidlo

Protože výstupní impulsy čidla SMT 160 mají amplitudu 5V a vstup mikropočítače je určen pouze pro napětí 3,3V, je výstup čidla připojen přes napěťový dělič tvořený dvojicí rezistorů R6 a R7. Toto řešení je velmi jednoduché a zkreslení výstupního signálu je pouze minimální, proto není třeba používat žádné speciální tvarovací obvody.

5 PROGRAMOVÉ VYBAVENÍ

Software určený pro ovládání modelu tepelné soustavy řízené mikro počítačem byl vytvořen v jazyce symbolických adres pomocí vývojového prostředí Freescale Code Warrior for HC(S)08 (Obr. 29). Při jeho vývoji byl brán ohled na primární určení modelu jako výukové pomůcky pro programování mikro počítačů HCS08. Byly proto vytvořeny tři různé verze programového vybavení.



Obr. 29 – Vývojové prostředí Freescale CodeWarrior for HC(S)08

První verze obslužného programu je určena pouze pro vyzkoušení a základní provoz modelu. Uživateli by nemělo být umožněno jakkoliv zasahovat do zdrojového kódu programu.

Program je určen pro simulaci dvoupolohového regulátoru bez hystereze. Prvním krokem při vykonávání tohoto programu je nastavení žádané teploty v rozsahu 20-100°C pomocí potenciometru, který je přímo součástí vývojového kitu, a naměření teploty povrchu topného rezistoru. Následuje porovnání naměřené a nastavené teploty a podle výsledku tohoto porovnání je ovládáno topení. Hodnota naměřené a nastavené teploty se dále zobrazuje na dvouřádkovém LCD displeji. Přídavnou funkcí je posílání naměřené teploty soustavy na sériovou linku. Vyhodnocování signálu teplotního čidla se provádí pomocí časovače TPM2 s využitím funkce Input Capture.

U druhé verze zůstávají hlavní možnosti obslužného programu prakticky stejné jako u první, je zde ale použit jiný postup pro vyhodnocování výstupního signálu čidla SMT 160. Tento způsob je založen na vzorkování signálu a určování jeho logické úrovně v jednotlivých okamžicích vzorkování. Bližší popis je uveden v kapitole 5.2.

Třetí verze je pojata jako knihovna funkcí (top_gb60). Uživateli je dána možnost přímého ovládání modelu tepelné soustavy. Funkce knihovny umožňují realizaci např. některé z výše uvedených aplikací, nebo tvorbu obslužného programu zcela podle uvážení programátora (pro tento účel jsou zde vytvořeny funkce pro měření teploty, pro zapnutí nebo vypnutí topení, atd.). Algoritmus pro vyhodnocování regulační odchylky a následné ovládání topení je tak zcela v rukou uživatele.

Jednotlivé funkce knihovny top_gb60 umožňují:

- inicializaci A/D převodníku, časovače TPM2 a dále nastavení všech konstant pro správnou funkci kanálů 0 a 2 časovače TPM2 v režimu zachycení stavu (Input Capture) (kapitola 5.1)
- inicializaci A/D převodníku a dále nastavení všech konstant pro správnou funkci mikropočítače v režimu vzorkování vstupního signálu čidla SMT 160 (kapitola 5.2)
- nastavení požadované teploty pomocí potenciometru v rozsahu 20-100°C
- předání požadované teploty pomocí registru H:X jako celočíselné hodnoty v rozsahu Dolni_hranice - Horni_hranice. Tyto hodnoty je možné nastavit v knihovně funkcí, kde jsou uvedeny jako konstanty.
- vyhodnocení signálu teplotního čidla SMT 160 pomocí časovače TPM2 a následný výpočet teploty povrchu topného rezistoru
- vyhodnocení výstupního signálu teplotního čidla SMT 160 metodou vzorkování vstupu a následný výpočet teploty povrchu topného rezistoru
- porovnání naměřené a nastavené teploty a následné ovládání topení, tj. simulace dvoupolohového regulátoru bez hystereze
- dvě samostatné funkce pro zapnutí nebo vypnutí topení + signalizace pomocí LED1 umístěné na kitu.

Hodnoty nastavené a naměřené teploty jsou uživateli předávány prostřednictvím indexového registru H:X ve tvaru čtyřmístného čísla ($T_{pred} = T_x \cdot 100$). Záleží tak pouze na uživateli, k čemu a jak předané parametry využije. Jednou z možností je například konstrukce poměrně jednoduchého teploměru s měřicím rozsahem cca 0-145°C, který můžeme realizovat prostřednictvím pouhých dvou funkcí z této knihovny.

Hodnoty nastavené a naměřené teploty jsou také uchovávány v globálních proměnných *nast_teplota* a *skut_teplota*, prostřednictvím kterých je lze načítat, popř. i ovlivňovat. Tento postup ale není příliš vhodný, protože pokud dojde k předání nevhodných parametrů nebo k jejich přepisu, není zaručena funkčnost prototypu.

Bližší popis jednotlivých funkcí této knihovny je uveden v kapitole 5.3.

Při psaní všech obslužných programů byly využity již vytvořené funkce, určené pro ovládání LCD displeje, sériové linky a dále také matematická knihovna pro práci s čísly v plovoucí řádové čárce.

5.1 Vyhodnocování signálu pomocí časovače TPM2

Vyhodnocování střídy signálu teplotního čidla SMT 160 se děje prostřednictvím kanálů 0 a 2 časovače TPM2, které pracují v režimu zachycení vstupu (Input Capture).

Kanál 0 (vektor přerušení je na adrese \$FFEC/FFED) je nastaven tak, aby generoval přerušení od náběžné a sestupné hrany signálu. V obsluze přerušení se určí, která hrana přerušení generovala a podle toho se uloží obsah záchytného registru TPM2C0V do proměnné *time1* (počátek stavu log 1) nebo *time2* (konec stavu log 1). Z hodnot *time1* a *time2* poté určíme hodnotu T2 - tj. dobu trvání signálu v úrovni H (log 1).

Výpočet T2 se provádí dvěma způsoby:

- během měření nedošlo k přetečení časovače TPM2: $T2 = time2 - time1$
- během měření došlo k přetečení časovače TPM2: $T2 = TPM2MOD - time1 + time2$

Kanál 2 (vektor přerušení je na adrese \$FFE8/FFE9) je nastaven tak, aby generoval přerušení pouze při náběžné hraně signálu. V obsluze přerušení se určí, zda se jedná o

počátek nebo konec impulsu a podle toho se uloží obsah záchytného registru TPM2C2V do proměnných *time3* (začátek periody) nebo *time4* (konec periody). Z hodnot *time3* a *time4* poté určíme hodnotu T1 - tj. periodu signálu.

Výpočet T1 se provádí dvěma způsoby:

- během měření nedošlo k přetečení časovače TPM2: $T1 = time4 - time3$
- během měření došlo k přetečení časovače TPM2: $T1 = TPM2MOD - time4 + time3$

Měření periody signálu je nutné, protože kmitočet výstupního signálu se liší čidlo od čidla a může kolísat v rozmezí 1kHz až 4kHz.

V následujícím odstavci je spočítán počet period signálu připadající na jeden cyklus časovače TPM2 a dále počet period signálu během 1s. Důvodem je jednak snaha určit, kolik impulsů by bylo třeba zpracovat, pokud bychom chtěli vyhodnocovat signál přesně, a dále také určit pravděpodobnost přetečení časovače během měření jednoho impulsu.

Změřená frekvence výstupního signálu čidla SMT 160.

$$f_v = 3kHz$$

Doba trvání jedné periody signálu.

$$T_v = \frac{1}{f_v} = \frac{1}{3 \cdot 10^3} = 0,00033s$$

Počet period signálu za jednu sekundu.

$$n_v = f_v \cdot t = 3000Hz \cdot 1s = 3000$$

Rychlost časovače

Dobu, po které bude docházet k přetečení časovače TPM2, lze spočítat ze vzorce:

$$t_{TOF} = \frac{Modulo \cdot Prescaler}{f_{source}} \quad (12)$$

Kde: *Modulo* – údaj, který zapíšeme do modulo registru TPM2MOD

t_{TOF} – čas do přetečení časovače [s]

f_{source} – frekvence zdroje hodinového kmitočtu [Hz]

Prescaler – nastavení vstupní frekvenční děličky

Hodnoty nastavené v časovači TPM2:

Modulo = #ffffh = 65535D

f_{source} = 20MHz

Prescaler = 8D

Z nastavených konstant časovače spočítáme přibližnou dobu trvání jednoho cyklu.

$$t_{\text{TOF}} = \frac{\text{Modulo} \cdot \text{Prescaler}}{f_{\text{source}}} = \frac{65535 \cdot 8}{20 \cdot 10^6} = 0,026214\text{s}$$

Pokud vydělíme hodnotu t_{TOF} délkou periody signálu T_V , dostaneme počet period signálu připadajících na jeden cyklus časovače TPM2.

$$n = \frac{t_{\text{TOF}}}{T_V} = \frac{0,026214}{0,00033} \approx 79$$

Na jeden cyklus časovače tak připadá přibližně 79 impulsů výstupního signálu. Z předcházejících výpočtů je patrné, že pravděpodobnost přetečení časovače během měření jednoho impulsu teplotního čidla se pohybuje pod hranicí 2%.

Abychom dosáhli maximální přesnosti při měření teploty, bylo by vhodné, aby mikropočítač zpracovával všechny impulsy výstupního signálu a následně ze získaných hodnot vypočítal průměrnou hodnotu. To však vzhledem ke značné náročnosti výpočtu není možné a počet impulsů, které mikropočítač skutečně vyhodnotí, je podstatně menší (zde je nutné zdůraznit, že mikropočítač reaguje na všechny impulsy čidla, ale jen z některých vypočítává teplotu povrchu soustavy). Podle velmi hrubého odhadu bude počet takto vyhodnocených impulsů nižší přibližně o dva až tři řády.

Tento předpoklad je založen na tom, že hlavní program obsahuje čekací smyčky, které se vykonávají po dobu cca 0,4s. Tyto prodlevy jsou v programu použity především

proto, aby byly výsledné hodnoty na displeji zobrazeny čitelně. Dá se tedy odhadnout, že se skutečná teplota vypočítává přibližně 2krát za sekundu.

Pokud bychom čekací smyčky vynechali, bylo by možné teplotu vyhodnocovat přibližně 10krát za sekundu. I tento počet je však vzhledem k celkovému počtu impulsů (cca 3000 impulsů čidla za sekundu) zanedbatelný a pro přesné měření nedostačující.

Výpočet hodnot T1 a T2 provádí mikropočítač mimo obsluhu přerušení v hlavní smyčce programu až ve chvíli, kdy má k dispozici aktuální hodnoty proměnných *time1* až *time4*. Během tohoto výpočtu je v obsluze přerušení zakázáno přepisovat hodnoty těchto proměnných. Mohlo by se totiž stát, že než by mikropočítač vypočítal hodnotu T1 nebo T2, došlo by k přepsání některé z proměnných a výsledky by tak byly chybné.

Tento způsob získání hodnot T1 a T2 je pro reálné účely zbytečně složitý a paradoxně dosahujeme nižší přesnosti, než kdybychom použili metodu, kdy mikropočítač po dobu několika desítek milisekund testuje, zda je na vstupu, ke kterému je čidlo připojeno, stav log 1 nebo log 0. Pokud je na vstupu stav log 1, inkrementujeme proměnnou *stav1*. Proměnnou *stav2* inkrementujeme, je-li na vstupu stav log 0 nebo log 1. Podíl proměnných *stav1* a *stav2* je pak ekvivalentní podílu hodnot T2 a T1. Tento postup pak stačí periodicky opakovat, což zajistíme například pomocí časovače. Nepřesnost je zde dána pouze dobou, během které mikropočítač vykonává příslušné instrukce programu (kapitola 5.2).

Metoda zachycení vstupu (Input Capture) je však výhodnější pro studijní účely, neboť lépe ukazuje možnosti časovače, kterým je vybaven mikropočítač HCS08.

V dalším kroku z hodnot T1 a T2 mikropočítač vypočítá pomocí přepočtového vzorce $t = 212,7660 \cdot \frac{T2}{T1} - 68,0851$ teplotu povrchu topného rezistoru. Výpočet je proveden s využitím funkcí knihovny pro práci s čísly v plovoucí řádové čárce.

V další části ukázkového programu mikropočítač porovná hodnotu naměřené a nastavené teploty. Pokud je skutečná teplota nižší než teplota nastavená, je výstupem PTF7 sepnut tranzistor Q₂ a dojde tak k zapnutí topení. Pokud je skutečná teplota větší nebo rovna teplotě nastavené, je topení vypnuto. Tímto algoritmem je tedy jednoduše realizována dvoupolohová regulace teploty.

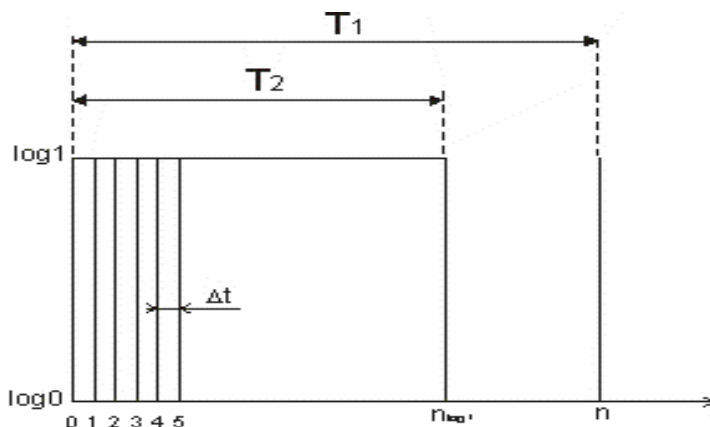
Nakonec jsou hodnoty naměřené a nastavené teploty zobrazeny na LCD displeji.

5.2 Vyhodnocování signálu pomocí vzorkování

Obecně se dá říci, že metodu vyhodnocování signálu pomocí vzorkování využívá jak metoda zachycení vstupu (Input Capture), tak také metoda popsaná v této kapitole. Hlavní rozdíl je především v délce periody vzorkování a v počtu vyhodnocených period signálu teplotního čidla.

U metody zachycení vstupu je délka vzorkovací periody krátká a je dána rychlostí časovače TPM2. Proměnná T_2 tak představuje počet impulsů časovače během doby, kdy je signál v úrovni log 1 a proměnná T_1 představuje počet impulsů časovače, které připadají na jednu periodu signálu. Hodnotami T_1 a T_2 tedy přesně popisujeme jednu periodu signálu. Princip této metody je znázorněn na (Obr. 30) a byl blíže popsán v kapitole 5.1.

Tím, že vyhodnocujeme pouze jednu periodu signálu, vzniká problém s reprodukovatelností výsledných hodnot. Je to způsobeno tím, že se délky jednotlivých impulsů poněkud liší a mikropočítač tyto odchylky vyhodnocuje. Důsledkem toho je, že jednotlivé výsledky kolísají okolo přesné hodnoty, a to i tehdy, když je měřená teplota konstantní.



Obr. 30 – Vzorkování výstupního signálu čidla SMT 160

Princip mnou navržené metody spočívá v tom, že v časových okamžicích Δt zjišťujeme logickou úroveň výstupního signálu teplotního čidla. Pokud je signál v úrovni log 1, inkrementujeme proměnnou T_2 . V případě, že je v úrovni log 0, inkrementujeme proměnnou T_1 . Perioda vzorkování Δt je zde delší než u metody zachycení vstupu a navíc její délka není konstantní. Důvodem je, že není dána přesným časovým okamžikem jako je tomu při použití časovače, ale představuje dobu, během které mikropočítač vykoná instrukce pro testování vstupu a inkrementaci proměnných T_1 a T_2 .

Pokud bychom touto metodou popisovali pouze jedinou periodu signálu, byly by výsledky nepřesné. Tento problém je zde vyřešen tak, že odebíráme maximum vzorků z velkého množství period. Tím se minimalizují nepřesnosti způsobené vzorkováním a také rozdíly v délce jednotlivých period výstupního signálu čidla SMT 160.

Abychom dokázali vypočítat teplotu povrchu topného rezistoru, musíme znát počet impulsů, kdy byl signál v úrovni log 1 a dále počet impulsů, kdy byl signál v úrovni log 1 nebo log 0. Tuto hodnotu získáme jednoduše tak, že sečteme proměnné T1 a T2 a výsledek uložíme zpět do proměnné T1. Získáme tak dvě hodnoty, které jsou ekvivalentní hodnotám T1 a T2, získaným pomocí metody zachycení vstupu (Input Capture). Díky tomu můžeme část programu pro výpočet teploty povrchu topného rezistoru použít přímo bez jakýchkoliv úprav.

Protože při vyhodnocování signálu nesmí dojít k přetečení dvoubajtových proměnných T1 a T2, je maximální možný počet odebíraných vzorků 65535 (2^{16}). Při tomto počtu vzorků je dosahovaná přesnost dostatečná, protože tato metoda jednoduchým způsobem řeší také zprůměrnování více naměřených hodnot.

Výhodou této metody je, že se obejdeme bez použití časovače a obsluhy přerušení.

Nevýhodou je velký počet instrukcí, které musí mikropočítač vykonat, abychom získali hodnoty T1 a T2 nutné pro výpočet skutečné teploty. Tento počet dosahuje hodnoty přibližně $5 \cdot 10^6$ instrukcí. Metoda je tedy vhodná pro ty případy, kdy nevyžadujeme velké množství měření a mikropočítač nemusí vykonávat jinou časově náročnou činnost.

Další postup při vykonávání ukázkového programu je stejný jako u verze programu, která byla popsána v kapitole 5.1.

Princip vykonávání programů je znázorněn pomocí vývojových diagramů, které jsou uvedeny v elektronické verzi jako příloha na CD.

5.3 Knihovna funkcí top_gb60

V úvodu kapitoly 5 byly uvedeny základní informace o knihovně funkcí top_gb60. V této části jsou popsány jednotlivé funkce této knihovny, jejich vstupy a výstupy. Použití jednotlivých funkcí je ilustrováno na jednoduchém příkladě. Podrobnější manuál je k dispozici v elektronické verzi na přiloženém CD.

Aby byla knihovna funkcí použitelná, je nutné ji nejprve umístit do projektu a inicializovat. Dále je nutné také inicializovat knihovnu pro práci s čísly v plovoucí řádové čárce (postup je zjednodušeně popsán v manuálu, který je umístěný na příloženém CD).

Seznam a popis funkcí knihovny `top_gb60`

INIT_TOP

Tuto funkci je nutné použít vždy, pokud chceme vyhodnocovat signál teplotního čidla pomocí časovače TPM2 (popis funkce - kapitola 5.1). Provádí inicializaci A/D převodníku, časovače a dále proměnných, nutných pro správnou funkci programu.

INIT_TOP_VZORKOVANI

Tuto funkci je nutné použít vždy, pokud chceme vyhodnocovat signál teplotního čidla metodou vzorkování vstupu (popis funkce - kapitola 5.2). Provádí inicializaci A/D převodníku a dále proměnných, nutných pro správnou funkci programu.

NAST_TEP

Tato funkce umožňuje nastavit žádanou teplotu pomocí potenciometru, který je součástí vývojového kitu M68EVB908GB60. Hodnota nastavené teploty je vrácena v registru H:X v rozsahu 2000-10000 (20°C-100°C). Tento tvar umožňuje vyjádřit číslo se dvěma desetinnými místy jako celočíselnou hodnotu (obsah H:X = 2536, nastavená teplota = 25,36°C). Hodnota je uložena do globální proměnné *nast_teplota* a následně předána v registru H:X do hlavního programu.

výstup: H:X - hodnota nastavené teploty, globální proměnná *nast_teplota*

NAST_TEP_HX

Tato funkce umožňuje předat žádanou teplotu z hlavního programu v registru H:X do podprogramu. Jejím úkolem je kontrola, zda zadaná hodnota teploty náleží do intervalu Dolni_hranice-Horni_hranice (přednastavený rozsah je 2000-10000 - tj. 20°C-100°C). Rozsah lze změnit přímo v knihovně `top_gb60`.

vstup: H:X - hodnota nastavené teploty

Tuto funkci lze nahradit zápisem nastavené teploty přímo do proměnné *nast_teplota*, není tak ale zaručena správná funkce programu a modelu.

TEP_TMP_GET_C

Tato funkce vypočítává teplotu povrchu topného rezistoru. Hodnoty nutné pro výpočet se získávají pomocí kanálů 0 a 2 časovače TPM2, které pracují v Input Capture (kapitola 5.1).

Aby tato funkce pracovala správně, musíme nejprve zavolat funkci INIT_TOP.

výstup: H:X - hodnota teploty povrchu topného rezistoru, globální proměnná

skut_teplota

TEP_VZORKOVANI_GET_C

Tato funkce vypočítává teplotu povrchu topného rezistoru. Hodnoty nutné pro výpočet se určují metodou vzorkování vstupu (kapitola 5.2).

Aby tato funkce pracovala správně, musíme nejprve časovač inicializovat pomocí funkce INIT_TOP_VZORKOVANI.

výstup: H:X - hodnota teploty povrchu topného rezistoru, globální proměnná

skut_teplota

TOP_ON

Tato funkce zapne topení a signalizuje je pomocí LED1, která je umístěna na vývojovém kitu.

výstup: zapnutí topení výstupem PTF7

TOP_OFF

Tato funkce vypne topení a signalizační LED1, která je umístěna na vývojovém kitu.

výstup: vypnutí topení výstupem PTF7

TOP_DVOUPOLOHOVY_REGULATOR

Tato funkce simuluje dvoupolohový regulátor bez hystereze (kapitola 2.3.1)

vstupy: hodnota žádané a naměřené teploty se předává pomocí dvojice globálních proměnných *nast_teplota* a *skut_teplota*

výstup: ovládání topení výstupem PTF7

Příklad

V pravidelných časových intervalech cca 1s změřte teplotu soustavy a tuto hodnotu zobrazte na displeji. Střídavě zapínejte nebo vypínejte topení.

definice proměnných

```
PREVOD ds.b 14
```

```
ZAPNOUT ds.b 1
```

definice konstant

```
TEXT dc.b 'teplota S:'
```

```
dc.b 0
```

realizace algoritmu

```
JSR INIT_TOP ; inicializuje knihovnu pro práci s modelem tep.
; soustavy
```

```
JSR INITMATH ; inicializuje matematickou knihovnu
```

```
MOV #%00000001, ZAPNOUT
```

OPAKUJ:

```
JSR TEP_TMP_GET_C ; načte teplotu soustavy
```

```
JSR UINT2FLT ; v následující části převedeme celočíselnou hodnotu
```

```
JSR FLTMOV1TO2 ; teploty na reálné číslo se dvěma desetinnými místy
```

```
LDHX #100
```

```
JSR UINT2FLT
```

```
JSR CH1AND2
```

```
JSR FLTDIV
```

```
LDHX # PREVOD ; adresa bufferu pro výsledek převodu
```

```
JSR FLT2ASC ; ulož do bufferu výsledek jako text
```

```
LDA ZAPNOUT ; v této části programu budeme střídavě zapínat a
```

```
CMP #%00000001 ; vypínat topení
```

```
BEQ zap
```

```
MOV #%00000001, ZAPNOUT
```

```
JSR TOP_OFF
```

```
JMP SKOK
```

ZAP:

```
MOV #%00000000, ZAPNOUT
```

JSR TOP_ON

SKOK:

```

JSR  dcls                ; vymažeme displej
LDX  #1                  ; 1. řádek
LDA  #1                  ; 1. sloupec
JSR  setcursor           ; nastavíme kurzor na první řádek a první sloupec
LDHX #TEXT
JSR  dtext
LDX  #1                  ; 1. řádek
LDA  #11                 ; 11. sloupec
JSR  setcursor           ; nastavíme kurzor na 1. řádek a první sloupec
LDHX #PREVOD             ; načteme adresu bufferu, ve které je uložena
                          ; vypočítaná teplota
JSR  dtext               ; vypíšeme teplotu na displej
JSR  CEKEJ               ; zavoláme podprogram, který se bude vykonávat cca
                          ; 1s, realizujeme jako čekací smyčku

JMP  OPAKUJ             ; opakování celého algoritmu

```

;časová prodleva

CEKEJ:

```

PSHH                ; uložíme používané registry na zásobník
PSHX
PSHA
LDA  #18
n2: LDHX #$FFFF
n1:  AIX  #-1         ; H:X = H:X-1 (pozor, nenastavuje příznaky)
    feed_watchdog    ; reset watchdogu
    CPHX #0          ; je H:X = 0?
    BNE  n1          ; pokud ne, skoč na n1
    DECA              ; A = A - 1
    BNE  n2          ; je A=0? Pokud ne, skoč na n2
    PULA
    PULX
    PULH              ; obnov původní obsah registru ze zásobníku
    RTS              ; návrat z podprogramu

```

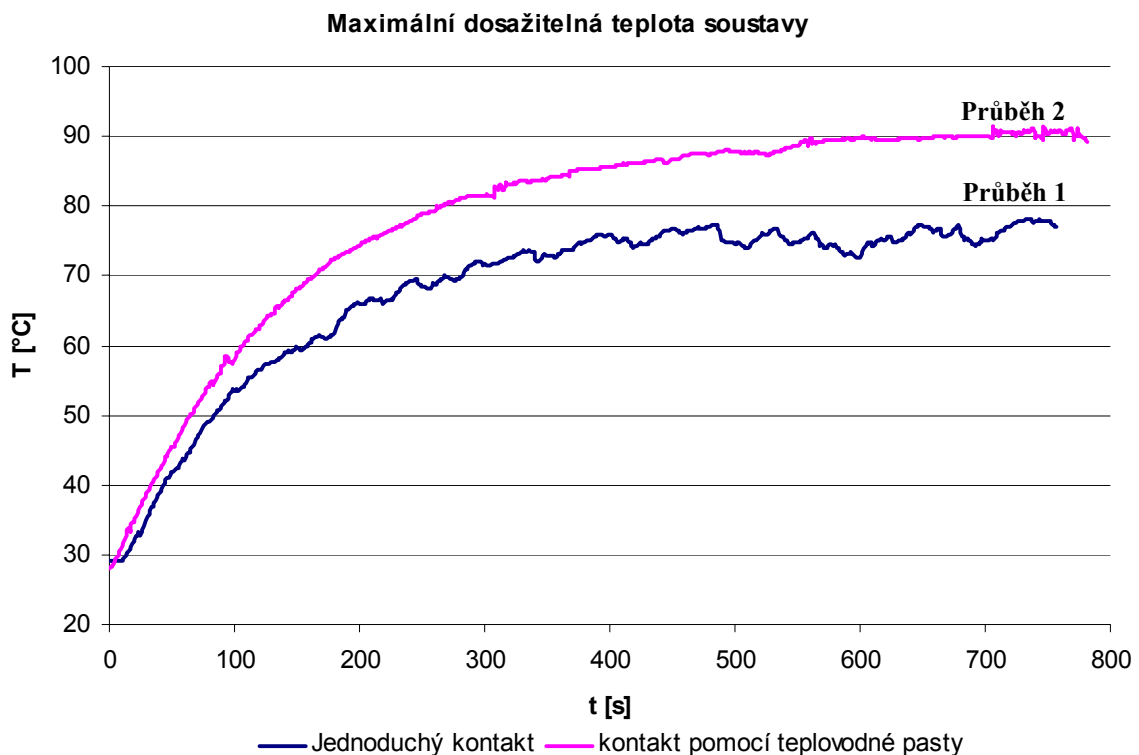

6 PRAKTICKÁ MĚŘENÍ NA MODELU TEPELNÉ SOUSTAVY

Na funkčním prototypu modelu tepelné soustavy byla provedena některá základní měření. Jejich účelem bylo především určit maximální teplotní rozsah, kterého lze u této soustavy dosáhnout, vyzkoušet některé základní průběhy regulace teploty a provést identifikaci soustavy pomocí Strejcovy metody.

6.1 Maximální teplotní rozsah

Teplotní rozsah soustavy byl naměřen při pokojové teplotě cca 25°C. Měření bylo provedeno pro dva různé způsoby kontaktu mezi teplotním čidlem a povrchem topného rezistoru:

1. kontakt je realizován pouhým přiložením čidla k topnému rezistoru
2. mezi čidlo a topný rezistor je nanesena vrstva teplovodné pasty z oxidu zinečnatého a topný rezistor je částečně zaizolován.



Obr. 31 – Maximální dosažitelná teplota soustavy

Z časového průběhu 1 je vidět, že maximální dosažitelná teplota povrchu topného rezistoru je při jednoduchém kontaktu čidla přibližně 78°C. Při této teplotě však naměřené hodnoty značně kolísají a získané výsledky jsou tak nespolehlivé.

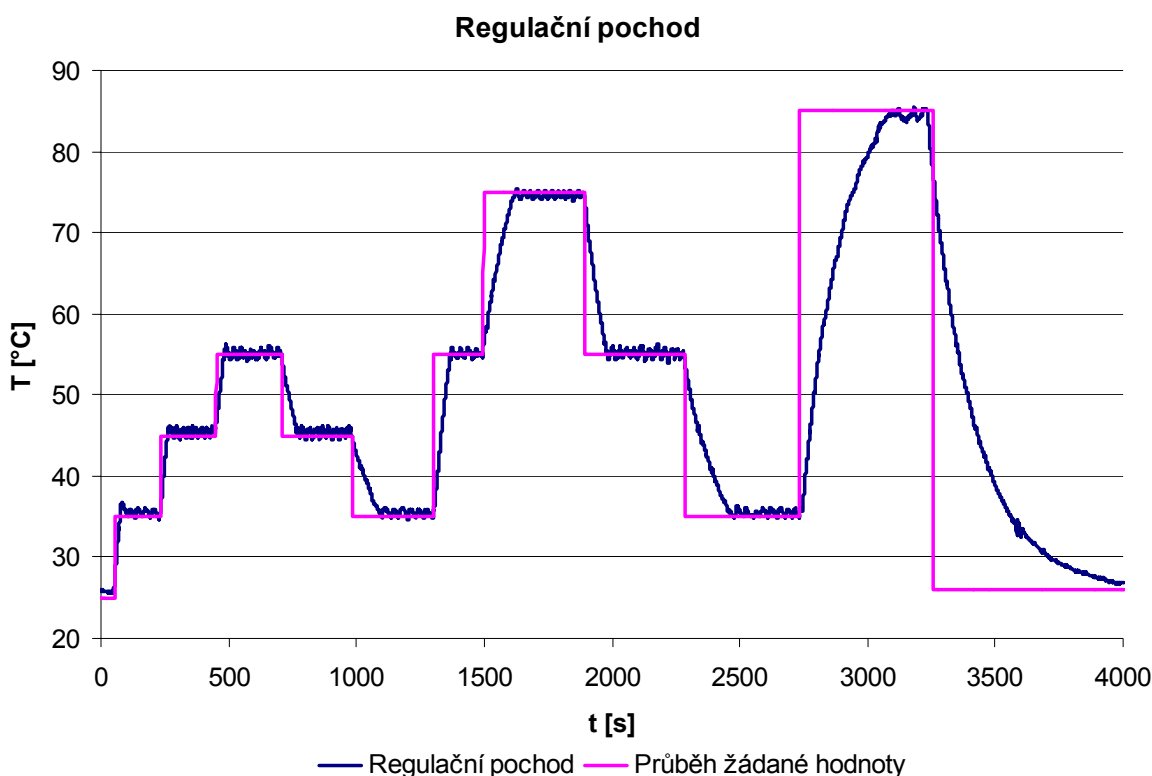
Relativně přesné výsledky tak lze získat přibližně do teploty 70°C, kdy je přesnost měření plně srovnatelná s běžnými digitálními teploměry.

Z časového průběhu 2 je jasné vidět, že teplovodná pasta a částečná izolace topného rezistoru a čidla výrazným způsobem vylepšily vlastnosti této soustavy. Maximální dosažitelná teplota vzrostla na přibližně 90°C a také se výrazně zmenšilo kolísání teploty.

6.2 Jednoduchý regulační pochod

Na (Obr. 32) je znázorněn jednoduchý regulační pochod, který byl realizován na modelu tepelné soustavy. Z naměřených hodnot je patrné, že se jedná o soustavu prvního nebo druhého řádu bez dopravního zpoždění. Přesné určení soustavy bude provedeno v kapitole 6.3.

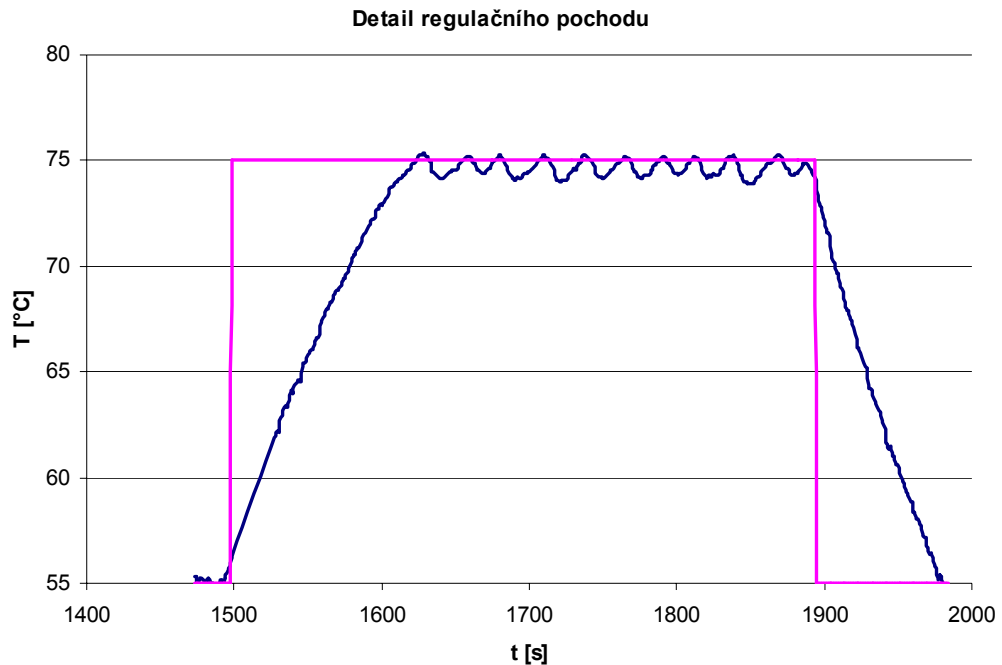
Pro jednoduchost měření se žádaná teplota měnila skokem.



Obr. 32 – Jednoduchý regulační pochod

Z (Obr. 33) je patrné, že se skutečná teplota nikdy přesně neustálí na žádané hodnotě, ale bude okolo ní trvale kmitat. Vzniká tak trvalá regulační odchylka, která se pohybuje v rozsahu cca $\pm 0,5-1^\circ\text{C}$. Toto kolísání je dáno především vlastnostmi

regulované soustavy, protože hystereze regulátoru je nulová. Pokud bychom tuto odchylku chtěli odstranit, museli bychom použít některý ze spojitých regulátorů, resp. museli bychom realizovat algoritmus, který by takový regulátor simuloval.



Obr. 33 – Detail regulačního pochodu

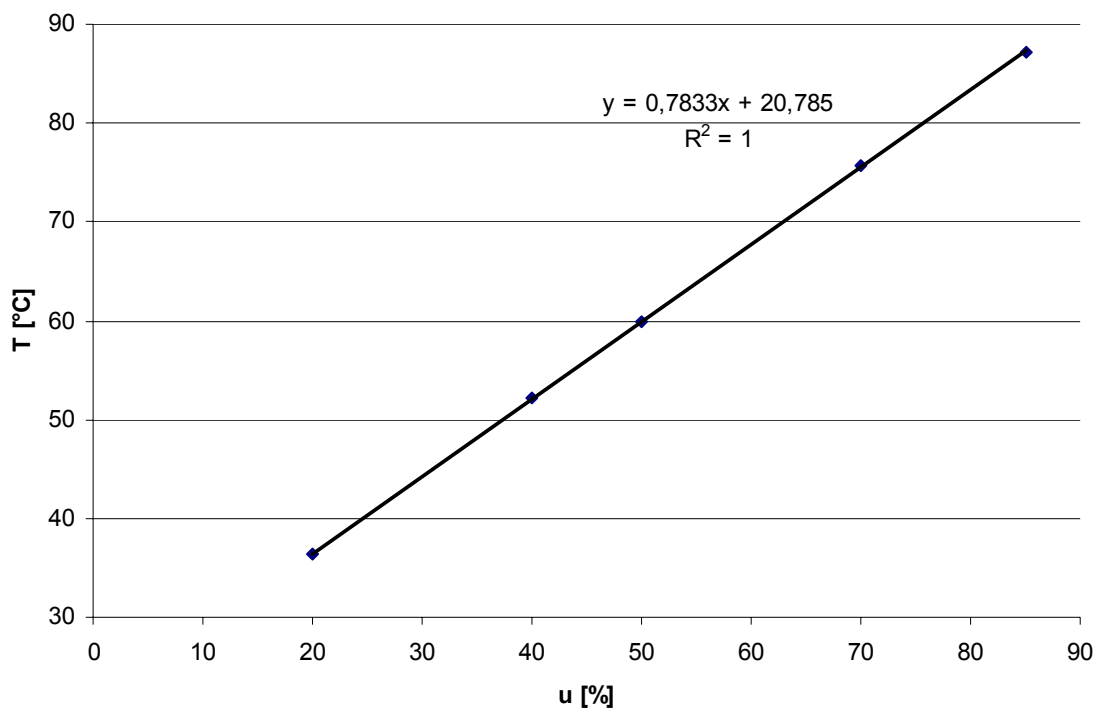
Při vyšších teplotách se navíc nepříznivě projevuje nedokonalost izolace soustavy a vlastnosti kontaktu mezi čidlem a povrchem topného rezistoru. Odchylka skutečné teploty od teploty žádané tak narůstá. Dá se ale říci, že vlastnosti soustavy zůstávají až do teploty přibližně 85 - 90°C pro simulaci přijatelné.

6.3 Identifikace soustavy

6.3.1 Statická charakteristika

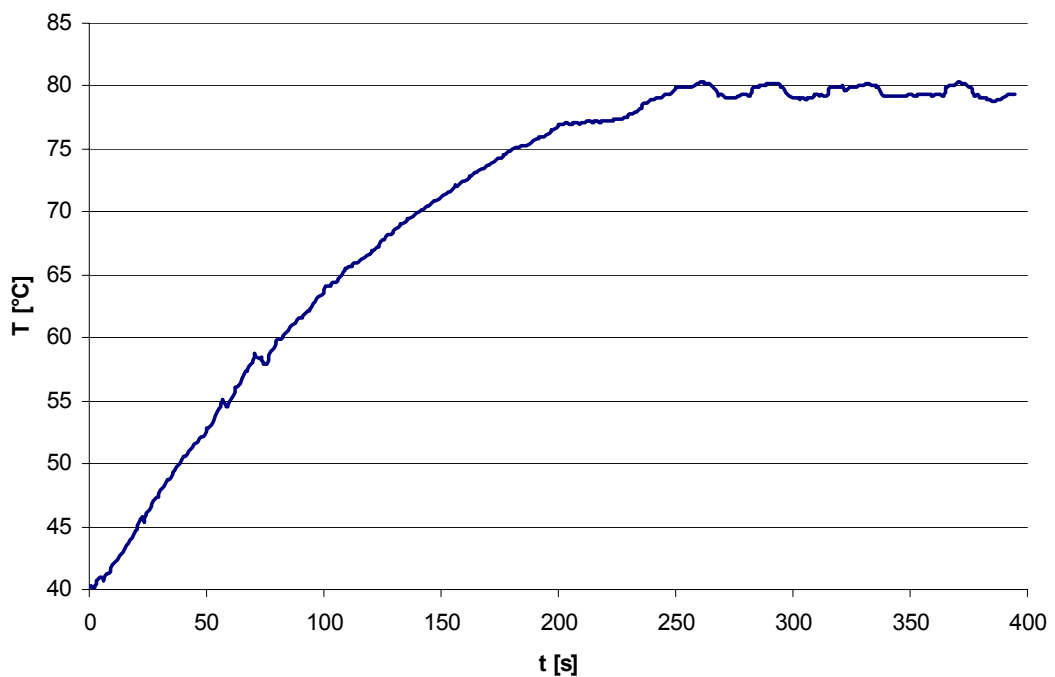
Tab. 2 - Hodnoty statické charakteristiky:

u[%]	T[°C]
20	36,33
40	52,25
50	59,98
70	75,69
85	87,26



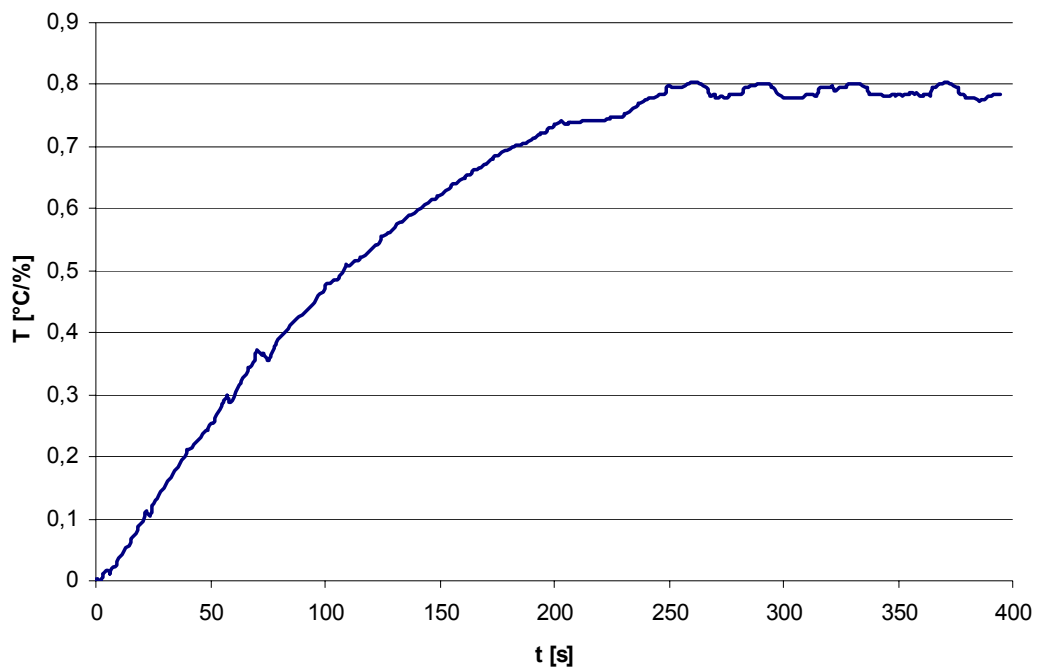
Obr. 34 – Statická charakteristika

Rovnice statické charakteristiky je $T=0,7833u+20,785$, lineární průběh je od 20% do 80% akční veličiny, proto byl pro měření zvolen skok žádané hodnoty z 25% na 75%.



Obr. 35 – Naměřená data

Na (Obr. 35) je naměřený regulační pochod, pro účely identifikace byla naměřená charakteristika posunuta do počátku a normována (Obr. 36).



Obr. 36 – Přechodová charakteristika

6.3.2 Identifikace soustavy - Strejcova metoda

V (Obr. 36) byl numericky určen inflexní bod $y(t_{inf})$ pomocí vztahu:

$$\frac{y(t^i) - y(t^{i-1})}{t^i - t^{i-1}} = \max \quad (13)$$

Deset hodnot z levého a pravého okolí inflexního bodu bylo aproximováno přímkou. Z její rovnice ve tvaru $y = a + b \cdot t = -0,0319 + 0,0074 \cdot t$ byly pomocí vztahů $T_u = -\frac{b}{a}$, $T_n = \frac{1}{a}$ získány hodnoty doby náběhu T_n a průtahu T_u .

$$T_u = -\frac{a}{b} = -\frac{-0,0319}{0,0074} = 4,311 \quad T_n = \frac{1}{b} = \frac{1}{0,0074} = 135,135$$

$$\tau_u = -\frac{T_u}{T_n} = \frac{4,311}{135,135} = 0,0319 \leq 0,104 \quad \Rightarrow \quad \text{nenásobné kořeny}$$

Z (Obr. 36) odečteme čas t_1 pro $y(t_1) = K \cdot 0,72 = 0,8038 \cdot 0,72 \approx 0,58$

$$t_1 = 133\text{s}$$

$$T_1 + T_2 = \frac{t_1}{1,2564} \quad \Rightarrow \quad T_1 + T_2 = \frac{133}{1,2564} = 105,858s$$

$$t_2 = 0,3574 \cdot (T_1 + T_2) = 0,3574 \cdot 136,1 = 37,834s$$

$$y(t_2) = 0,242937 \cong 0,243$$

$$y(t_2) = f(\tau)$$

Z grafu pro závislost $y(t_2)$ na τ (viz. [15], str. 69) byla odečtena pro $y(t_2) \cong 0,243$ hodnota $\tau \cong 0,1236$

$$\tau = \frac{T_2}{T_1} \quad \Rightarrow \quad \tau \cdot T_1 = T_2$$

$$T_1 + T_2 = \frac{t_1}{1,2564}$$

$$T_1 + \tau \cdot T_1 = 105,858$$

$$1,1236 \cdot T_1 = 105,858$$

$$T_1 = \underline{94,21s} \quad \Rightarrow \quad T_2 = 105,858 - T_1 = 105,858 - 94,21 = \underline{11,65s}$$

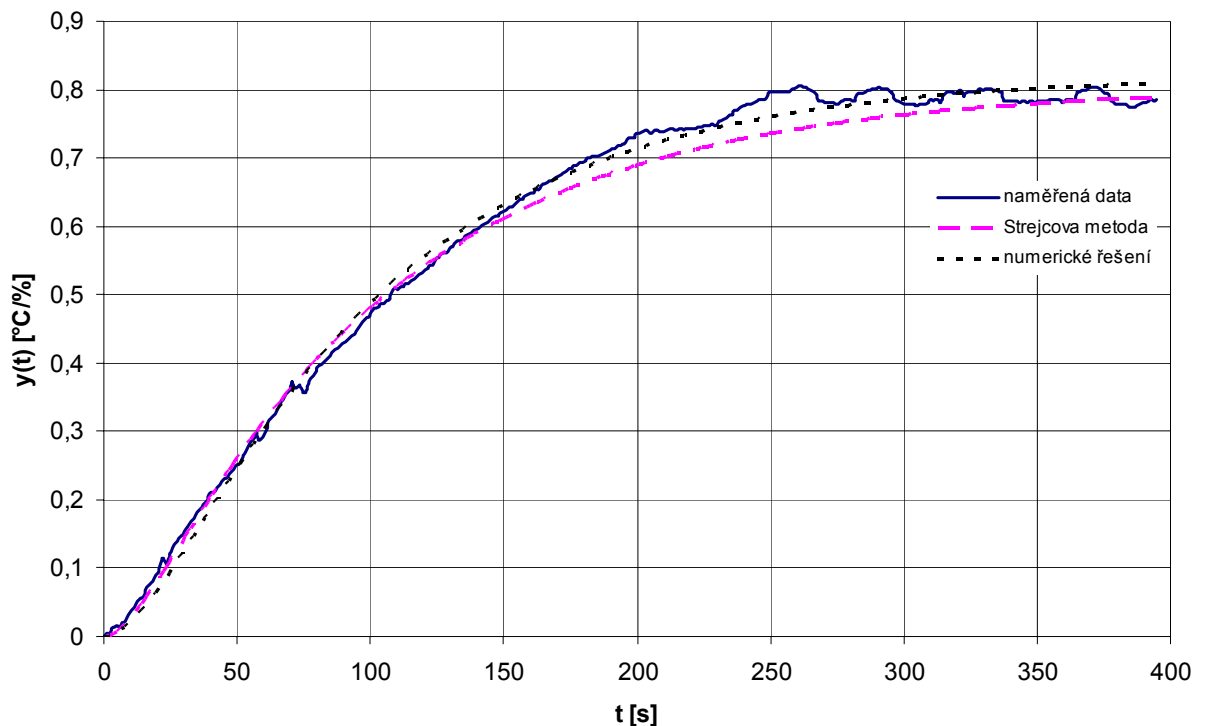
$$K = \frac{y(\infty) - y(0)}{\Delta u(t)} = \frac{40,1908}{50} = 0,8038^\circ C / \%$$

Pomocí Strejcovy metody byl získán tvar přenosu

$$G(s) = \frac{K}{(T_1 \cdot s + 1) \cdot (T_2 \cdot s + 1)} = \frac{0,8038}{(94,21 \cdot s + 1) \cdot (11,65 \cdot s + 1)}$$

Pomocí modulu řešitel aplikace Microsoft Excel byl získán zpřesněný tvar přenosu

$$G(s) = \frac{K}{(T_1 \cdot s + 1) \cdot (T_2 \cdot s + 1)} = \frac{0,8214}{(86,582 \cdot s + 1) \cdot (18,163 \cdot s + 1)}$$



Obr. 37 – Identifikace soustavy - porovnání tří přechodových charakteristik

Z provedené identifikace soustavy (Obr. 37) je patrné, že se jedná o soustavu druhého řádu bez dopravního zpoždění.

Přesnost identifikace je nepříznivě ovlivněna jednak vlastnostmi použitého regulátoru (realizovaného mikropočítačem) a dále také jednoduchou konstrukcí celé soustavy. Pokud bychom chtěli provést přesnější identifikaci, bylo by třeba minimalizovat kmitání měřené teploty kolem žádané hodnoty při ustalování. To lze provést použitím spojitého regulátoru a dále také lepší izolací soustavy, aby byla méně náchylná na rušení způsobované změnami teploty okolního prostředí.

I přes výše uvedené nedostatky je ale přesnost identifikace pro naše účely dostatečná a lze ji dále využít, např. při návrhu složitějších typů regulátorů.

6.4 Prototyp modelu tepelné soustavy řízené mikropočítačem

Na (Obr. 38) je zobrazen funkční prototyp modelu tepelné soustavy, jehož konstrukce byla popsána v předcházejícím textu. Model byl testován ve zkušebním provozu, během kterého se neprojevily žádné závažné konstrukční nebo funkční nedostatky, které by bránily jeho použití jako učební pomůcky.

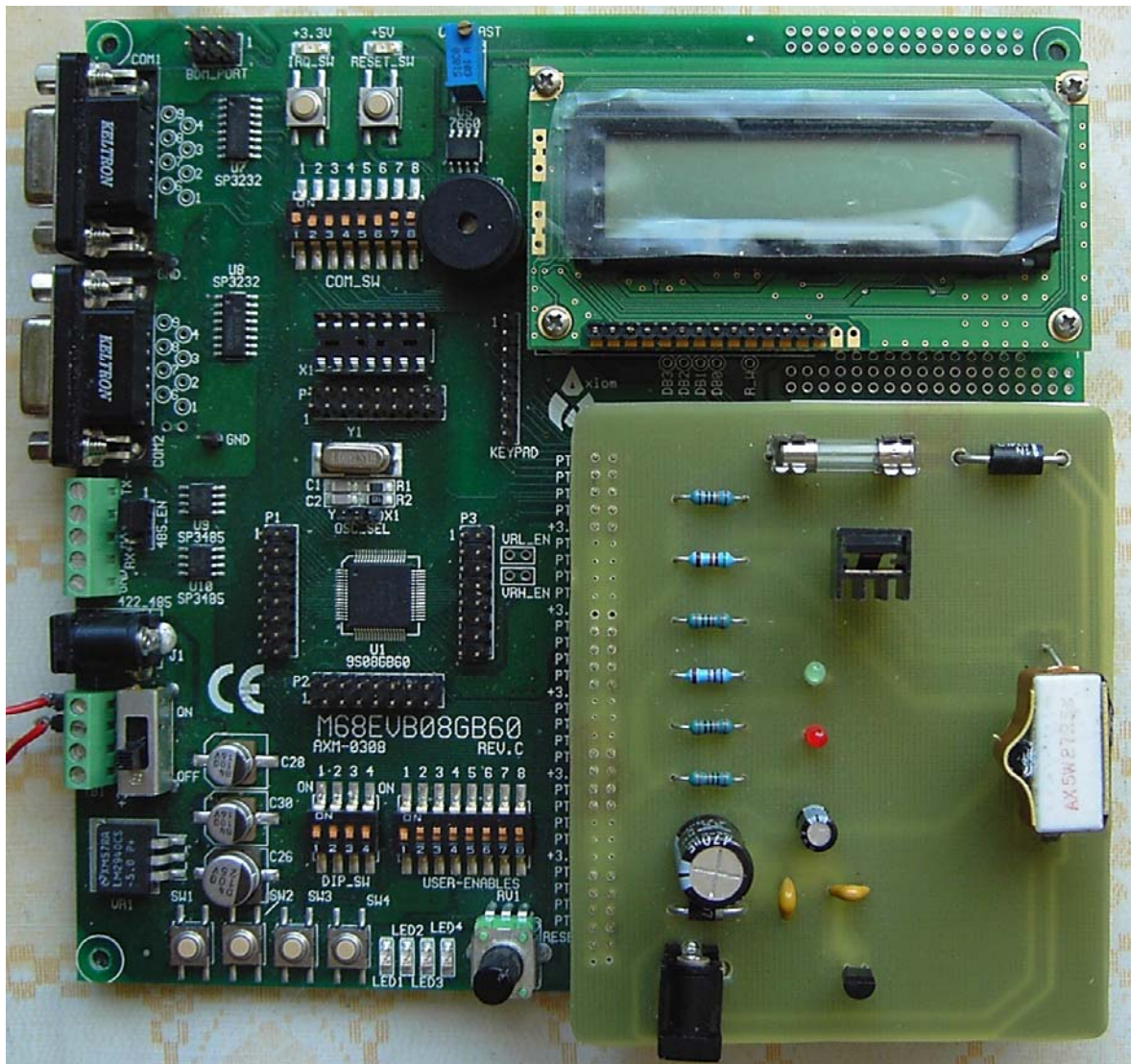
Technické parametry:

Rozměry desky: 95mm x 80mm

Napájecí napětí: 9 - 12V DC, typicky 9V DC

Odběr (pro 9 V): cca 300mA DC

Teplotní rozsah: $>0^{\circ}\text{C} - 90^{\circ}\text{C}$ (spodní hranice teplotního rozsahu je dána teplotou okolí)



Obr. 38 – Model tepelné soustavy řízené mikropočítačem

ZÁVĚR

Hlavním úkolem této bakalářské práce bylo vytvořit učební pomůcku, která by sloužila studentům při výuce programování mikropočítačů Freescale HCS08. Dalším cílem bylo také vytvořit průvodce, který by jednoduchou a srozumitelnou formou shrnul problematiku mikropočítačů a regulační techniky obecně.

Realizovaný model tepelné soustavy se vyznačuje především značnou jednoduchostí konstrukce s minimem použitých součástí. Tím jsou dány jak jeho kladné, tak také záporné vlastnosti.

Mezi výhody můžeme zařadit především poměrně nízké pořizovací náklady, minimální problémy s kalibrací jednotlivých částí modelu a také možnost dalšího rozvoje.

Mezi největší nevýhody naopak patří nepřesnosti při měření teplot vyšších než cca 85°C. Důvodem je jednak to, že topný prvek a teplotní čidlo jsou izolovány pouze částečně a dále také, že kontakt mezi čidlem a povrchem topného rezistoru je realizován pouze tenkou vrstvou teplovodné pasty. Pro teploty nižší než 85°C je však dosahovaná přesnost plně srovnatelná s běžně používanými digitálními teploměry.

V rámci tohoto projektu slouží mikropočítač v ukázkové aplikaci pouze jako zjednodušený dvoupolohový regulátor bez hystereze. Jeho možnosti však umožňují simulaci i jiných typů regulátorů (P, PD, PS a PSD). K jejich realizaci by teoreticky stačilo, abychom podle vhodného algoritmu zpracovávali přímo regulační odchylku a nejen porovnávali skutečnou a žádanou hodnotu teploty. K ovládní topného rezistoru by pak bylo možné s výhodou použít některý z kanálů časovače TPM1 nebo TPM2, který by pracoval v režimu PWM.

Z tohoto důvodu byla také provedena základní identifikace soustavy. Byl tak získán G(s) přenos soustavy, který lze případně využít při návrhu složitějších typů regulátorů.

ZÁVĚR V ANGLIČTINĚ

The main aim of this bachelor work was to create a teaching aid that would be used by students during lessons of programming Freescale HCS08 microcomputers. Further purpose, that I wanted to achieve, was a guide that would simply and intelligibly involve the readers into the problems of microcomputers and regulation technology generally.

Realization of the model of a heat system is characterized by simplicity of construction with minimum used parts. That gives both its positive and also negative points.

We can count the advantages of relatively low acquisition costs, minimum problems with calibration at single part of model and also the chances of further development.

On the other hand we can mention problems with measuring a temperature higher than c. 85°C. Reasons are: first, the heating element and temperature-sensitive element seclusion are made only in part and second the contact between the temperature sensor and the surface heating resistor are realized only with thin strata of warm-water paste. For temperatures lower than 85°C the accuracy is fully comparable with common digital thermometers.

In this project the microcontroller acts only as a simplified two-state controller without hysteresis. However, it is possible to realize other types of regulators (P, PD, PI and PID). For their realization it would suffice to process the regulation difference and not only compare the real and the set point of temperature. For operating the heating resistor we can use some of the timer TPM channel, which would work in PWM mode.

On this account the fundamental identification of the system was also performed. So the transfer function $G(s)$ of the system was obtained, which could be used for implementing more complicated controllers.

SEZNAM POUŽITÉ LITERATURY

- [1] Kovář, Josef. *Učební texty pro předměty automatizace a informatika*, Zlín: Střední průmyslová škola, 2002,2003
- [2] Váňa, Vladimír. *Začínáme pracovat s mikrokontroléry HC08 Nitron*. 1. vydání, Praha: BEN - technická literatura, 2003, 96 s, ISBN 80-7300-124-1
- [3] Hruška, František. *Technické prostředky informatiky a automatizace*. 1. vydání, Zlín: Univerzita Tomáše Bati, Fakulta aplikované informatiky, 2007, 192s, ISBN 978-80-7318-535-0
- [4] Vašek V., Vašek L. *Programování mikropočítačů*, Zlín: FT VUT v Brně, 1989
- [5] Ladman J., *Elektronické konstrukce pro začátečníky*, Praha: BEN - technická literatura, 2003
- [6] CPU08 Central Processor Unit Reference Manual. Motorola, 2001
- [7] HCS08 Family Reference Manual, Rev.1. Freescale Semiconductor, 2003
- [8] MC9S08GB60 Data sheet. Freescale Semiconductor, 2004
- [9] M68EVB908GB60 Development Board for Freescale MC9S08GB60. Freescale Semiconductor, 2006
- [10] Reich, Jakub. Bakalářská práce *Archivační teploměr s využitím mikropočítače řady HC08*. Zlín: Univerzita Tomáše Bati, Fakulta aplikované informatiky, 2006
- [11] Katalogový list SMT 160-30. Smartec, 2002, dostupný z WWW:
<http://www.hamradio.cz/aprs/uismt/SMT_160-30_CZ.pdf
- [12] Špringl, Vít. *Měření teploty - kovové odporové senzory teploty*, dostupný z WWW:
<<http://www.hw.cz/Dokumentace/Teorie-a-praxe/ART1095-Měření-teploty---kovove-odporove-senzory-teploty.html>
- [13] Špringl, Vít. *Měření teploty - polovodičové odporové senzory teploty*, dostupný z WWW:
<<http://www.hw.cz/Dokumentace/Teorie-a-praxe/ART1141-Měření-teploty---polovodicove-odporove-senzory-teploty.html>

[14] Staněk, Jan. *RS 485&422* , dostupný z WWW:

<<http://www.hw.cz/Dokumentace/Teorie-a-praxe/ART821-RS-485-26amp%3B-422.html>

[15] Navrátil, P., Gerlich, V. *Podpora seminářů předmětu "Teorie automatického řízení II"*. Zlín: Univerzita Tomáše Bati, Fakulta aplikované informatiky, 2006

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

A/D	Analog/Digital
BCD	Binary Coded Decimal
BDT	Balanced Data Transmission
CISC	Complex Instruction Set Computer
CMOS	Complementary Metal Oxide Semiconductor
CPU	Computer Processor/Processing Unit
DC	Duty Cycle
DC	Direct Current
DIP	Dual In-Line Package
DVT	Differential Voltage Transmission
EPROM	Electrically Programmable Read Only Memory
EEPROM	Electrically Erasable Programmable Read Only Memory
EMI	Electromagnetic Interference
HMOS	High Speed Metal Oxide Semiconductor
IRC	Individual Room Control
LED	Light Emitting Diode
LCD	Liquid Crystal Display
LIFO	Last In First Out
NTC	Negative Temperature Coefficient
PC	Personal Computer
PCC	Programmable Computing Controller
PLC	Programmable Logical Controller
PROM	Programmable Read Only Memory
PTC	Positive Temperature Coefficient

PWM	Pulse Width Modulation
RAM	Random Access Memory
RD	Receive Data
RISC	Reduced Instruction Set Computer
ROM	Read Only Memory
RWM	Read Write Memory
SCI	Serial Communication Interface
SCL	Serial Clock Line
SDA	Serial Data Line
SG	Signal Ground
SPI	Serial Peripheral Interface
SP	Stack Pointer
TD	Transmit Data
TTL	Transistor Transistor Logic

SEZNAM OBRÁZKŮ

Obr. 1 – Struktura mikropočítačového systému	12
Obr. 2 – von Neumannova koncepce počítače	13
Obr. 3 – Harvardská koncepce počítače	14
Obr. 4 – Vývojový kit M68EVB908GB60	16
Obr. 5 – Svorkovnice TB1	17
Obr. 6 – Struktura mikropočítače MC9S08GB/GT	18
Obr. 7 – Struktura CPU 68HC(S)08	19
Obr. 8 – Pracovní registry HC(S)08	20
Obr. 9 – Paměťová mapa mikro-	22
Obr. 10 – Schéma časovače mikropočítače HC(S)08	23
Obr. 11 – Statická charakteristika	25
Obr. 12 – Schéma A/D převodníku mikropočítače HCS08	26
Obr. 13 – Asynchronní přenos	28
Obr. 14 – Úrovně RS 232C a formát dat RS 232C	29
Obr. 15 – Sériový datový okruh	29
Obr. 16 – Schéma propojení RS 485	30
Obr. 17 – Synchronní přenos	31
Obr. 18 – Komunikační propojení pomocí SPI	32
Obr. 19 – Schéma regulačního obvodu	36
Obr. 20 – Regulátor a) zjednodušené blokové schéma, b) podrobné blokové schéma	37
Obr. 21 – Regulace a) bez hystereze, b) s hysterezi	39
Obr. 22 – Teplotní čidlo SMT 160-30	48
Obr. 23 – Výstupní signál čidla SMT 160-30	49
Obr. 24 – Napájecí část obvodu	53
Obr. 25 – Výkonová část obvodu	55
Obr. 26 – Vstupní obvod tranzistoru	55
Obr. 27 – Výstupní obvod tranzistoru	57
Obr. 28 – Teplotní čidlo	60
Obr. 29 – Vývojové prostředí Freescale CodeWarrior for HC(S)08	61
Obr. 30 – Vzorkování výstupního signálu čidla SMT 160	67
Obr. 31 – Maximální dosažitelná teplota soustavy	73

Obr. 32 – Jednoduchý regulační pochod	74
Obr. 33 – Detail regulačního pochodu.....	75
Obr. 34 – Statická charakteristika.....	76
Obr. 35 – Naměřená data	76
Obr. 36 – Přechodová charakteristika.....	77
Obr. 37 – Identifikace soustavy - porovnání tří přechodových charakteristik	79
Obr. 38 – Model tepelné soustavy řízené mikropočítačem	80

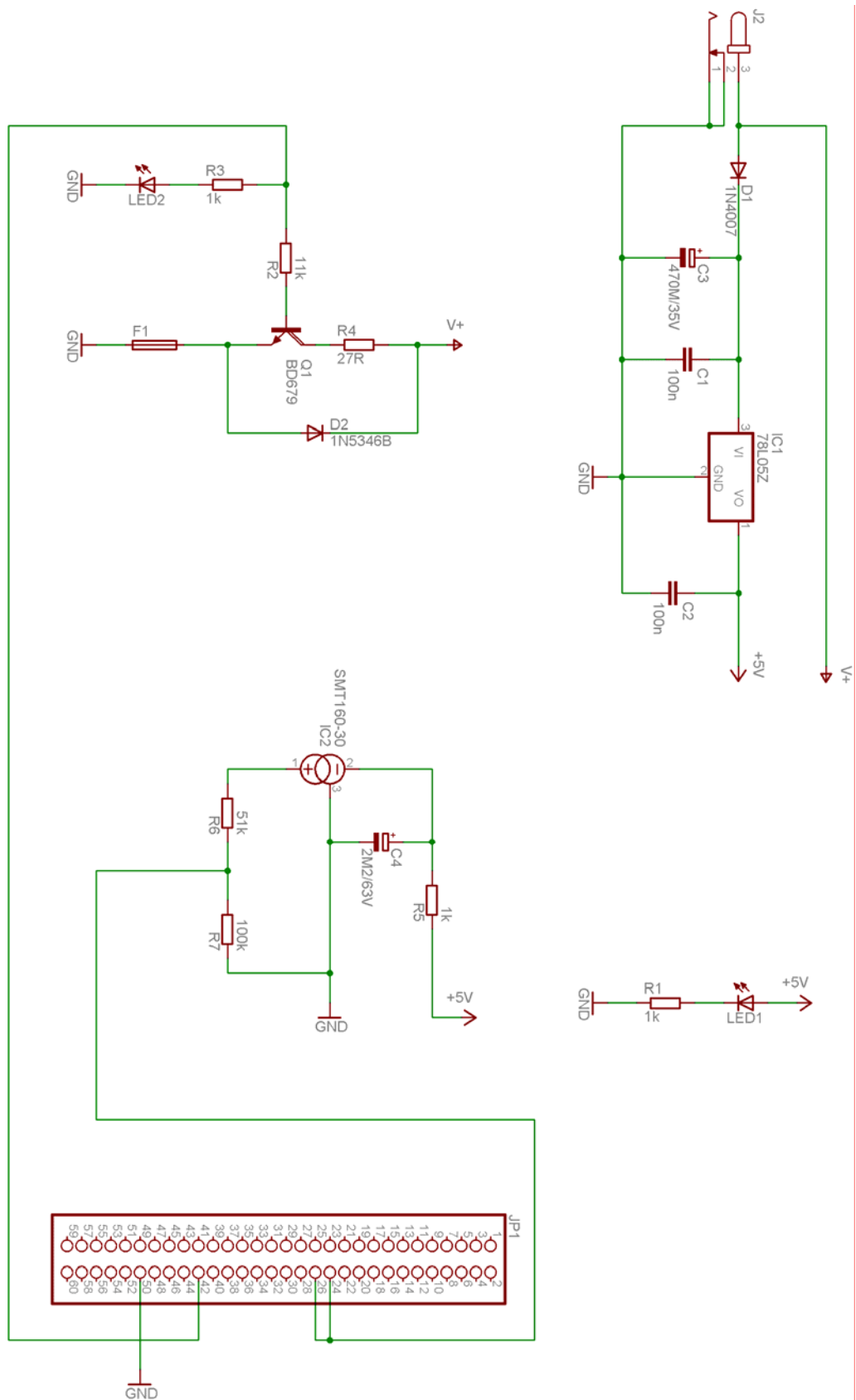
SEZNAM TABULEK

Tab. 1 - Všeobecné technické parametry čidla SMT 160-30:	49
Tab. 2 - Hodnoty statické charakteristiky:.....	75

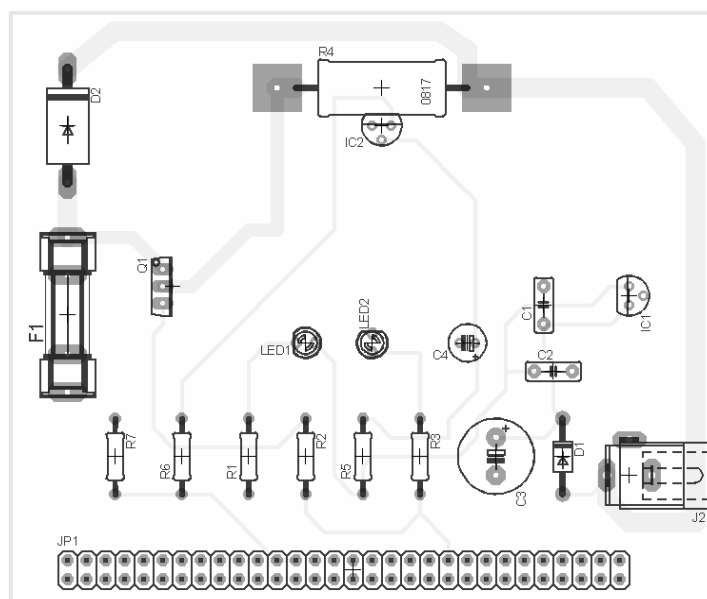
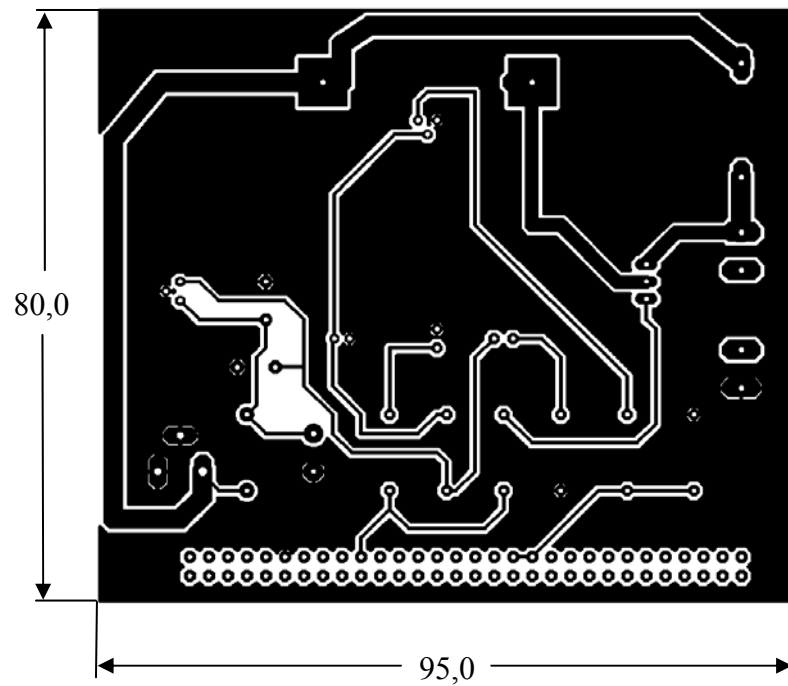
SEZNAM PŘÍLOH

- P I Schéma zapojení modelu tepelné soustavy
- P II Schéma plošného spoje modelu tepelné soustavy
- P III Seznam součástí
- P IV Elektronické přílohy a verze bakalářské práce na CD

PŘÍLOHA P I: SCHÉMA ZAPOJENÍ MODELU TEPELNÉ SOUSTAVY



PŘÍLOHA P II: SCHÉMA PLOŠNÉHO SPOJE MODELU TEPELNÉ SOUSTAVY



PŘÍLOHA P III: SEZNAM SOUČÁSTEK

	ks	typ	reference
1	2	100nF	C1, C2
2	1	470 μ /35V	C3
3	1	2 μ 2/63V	C4
4	1	1N4007	D1
5	1	1N5346B	D2
6	1	LED zelena 3mm	LED1
7	1	LED červená 3mm	LED2
8	1	Souosý nap. kon.	J1
9	1	Kolíková lišta 2x30pin	JP1
10	1	BD679	Q1
11	1	pojistka 500mA	F1
12	3	1k Ω	R1,R3,R5
13	1	11k Ω	R2
14	1	51k Ω	R6
15	1	100k Ω	R7
16	1	27 Ω /5W	R4
17	1	78L05	IC1
18	1	SMT 160-30	IC2