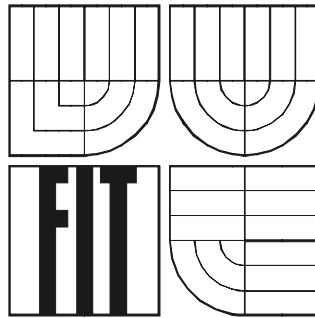


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



Monitoring teploty

Diplomová práce

Monitoring teploty

Odevzdáno na Fakultě informačních technologií Vysokého učení technického v Brně, dne 22. května 2007.

© Bc. Aleš Kotačka, 2007.

Autor díla převádí svá práva na reprodukci, distribuci a kopii celého díla i jeho části na Vysoké učení technické v Brně, Fakultu informačních technologií.

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Jána Kubeka.

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

.....
Bc. Aleš Kotačka
3.1.2007

Zadání diplomové práce

Řešitel: **Kotačka Aleš, Bc.**

Obor: Počítačové systémy a sítě

Téma: **Monitoring teploty**

Kategorie: Počítačová architektura

Pokyny:

1. Seznamte se výukovým kitem FIT a rozhraním USB.
2. Navrhněte konfiguraci pro FPGA na kitu umožňující připojení libovolného teplotního čidla (případně více čidel), případně takové teplotní čidlo navrhněte.
3. Navrhněte software pro kit umožňující zaznamenávání teplot s možností hromadného stahování zaznamenaných dat přes USB.
4. Navrhněte jednoúčelový software pro stahování těchto dat.

Literatura:

- <http://www.fit.vutbr.cz/kit>

Při obhajobě semestrální části diplomového projektu je požadováno:

- Splnění prvních 2 bodů zadání.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci ročníkového a semestrálního projektu (30 až 40% celkového rozsahu technické zprávy).

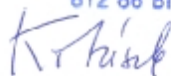
Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním paměťovém médiu (disketa, CD-ROM), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Kubek Ján, Ing., UPSY FIT VUT**

Datum zadání: 28. února 2006

Datum odevzdání: 22. května 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačových systémů a sítí
602 005 Brno, Božetěchova 2



doc. Ing. Zdeněk Kotásek, CSc.
vedoucí ústavu

Abstrakt

Práce se zabývá popisem FITkitu, jeho jednotlivých částí jako je MCU, FPGA, atd. Následuje popis USB rozhraní. Dále jsou uvedeny různé typy teplotních čidel a jejich vlastnosti a možnosti připojení čidel k FITkitu. V závěru je stručný princip činnosti celého systému.

Klíčová slova

USB rozhraní, sériový port, FITkit, teplotní čidla, měření teploty, teplotní stupnice, MCU, MSP430F168, FPGA, XC3S50-4PQ208C, USB-UART převodník, FT2232C, KTY81-210, SMT 160-30, DS18B20

Abstract

This Work deals with describe FITkit, his parts as MCU, FPGA, etc. Follow describe USB-interface. Onward is quoted different type's heat-sensitive elements and theirs properties and possibility connection to FITkit. In conclusion is short principle functioning system.

Keywords

USB interface, serial port, FITkit, heat-sensitive element, temperature detection, temperature scale, MCU, MSP430F168, FPGA, XC3S50-4PQ208C, USB-UART converter, FT2232C, KTY81-210, SMT 160-30, DS18B20

Obsah

Obsah.....	6
1 Úvod.....	7
2 FITkit.....	8
2.1 Popis FITkitu.....	8
2.1.1 MCU.....	9
2.1.2 FPGA.....	9
2.1.3 USB-UART převodník.....	10
2.1.4 Klávesnice a LCD displej.....	11
2.2 Rozhraní USB.....	11
2.2.1 Rychlost USB.....	12
2.2.2 Konektory.....	12
2.2.3 Napájení USB.....	13
2.2.4 Komunikace.....	14
2.3 Blokové schéma FITkitu.....	14
2.3.1 Princip programování MCU a FPGA.....	15
3 Měření teploty.....	16
3.1 Teplota.....	16
3.2 Teplotní čidla.....	17
3.2.1 Odporová teplotní čidla.....	17
3.2.2 Termistory.....	18
3.2.3 Převodník teplota/střída.....	19
3.2.4 Digitální teplotní čidlo.....	19
4 Analýza a návrh.....	21
4.1 Analýza.....	21
4.1.1 Uložení čísel.....	21
4.1.2 Komprimace hodnot.....	21
4.2 Princip činnosti.....	22
4.3 Komunikační protokol.....	22
5 Závěr.....	23
Literatura.....	24

1 Úvod

Tématem diplomové práce je návrh aplikace pro měření a záznam teploty z několika teplotních čidel. Systém by měl umožňovat zaznamenávání teploty na platformě FITkit bez nutnosti připojení k počítači a to po dobu několika desítek dní. Nedílnou součástí je též návrh obslužné aplikace, která bude naměřené data hromadně stahovat do počítače a umožní jejich další zpracování. Při výběru teplotních čidel byl brán důraz na cenu, jednoduchost a přesnost čidla.

Práce spadá jak do oblasti programování vestavěných systémů, tak i do oblasti běžného programování aplikací pro PC. Popisem školního vývojového kitu se zabývá kapitola druhá. V ní jsou postupně probrány jednotlivé části kitu, jako je MCU, FPGA a sběrnice USB. Právě přes USB sběrnici komunikuje kit s počítačem. V závěru kapitoly je uveden princip programování a nahrávání programu a konfigurace FPGA do FITkitu. Měření teploty se věnuje kapitola třetí, kde jsou rozepsány převody teplot mezi stupnicemi Kelvinovou, Celsiovou a Fahrenheitovou. Dále jsou uvedeny typy teplotních čidel, jejich vlastnosti a možnosti připojení ke kitu. Předposlední kapitola zvažuje způsoby uložení teploty a její kódování na menší velikost. Dále pak popisuje stručný princip funkce celého systému a komunikace kitu s počítačem. Počet kapitol zatím není konečný a v konečné verzi se určitě objeví kapitola věnovaná obslužnému programu pro PC. Závěrečná kapitola shrnuje dosavadní zkušenosti a poznatky celé práce a taktéž možnosti dalšího rozšíření projektu.

Systém bude možné využít pro např. pro statistické účely nebo v oblasti automatizace pro řízení a regulaci výrobních procesů.

2 FITkit

FITkit je vývojový kit, obsahující řadu HW komponent, určený pro praktickou výuku zejména hardwarových předmětů na Fakultě informačních technologií. Částečnou představu získáme pohledem na obrázek 2.1 nebo obrázek 2.2 a také v [1]. Jedná se tedy o vestavěný systém (anglicky *Embedded System*), jehož význam roste v poslední době. Řada aplikací, běžící na stolních počítačích, se v poslední době přesouvá do oblasti vestavěných systémů.



Obrázek 2.1



Obrázek 2.2

2.1 Popis FITkitu

Jak již bylo řečeno FITkit obsahuje řadu periférií a rozhraní:

- MCU MSP430F168 (Texas Instruments)
- FPGA Spartan 3 XC3S50-4PQ208C (Xilinx)
- USB-UART převodník FT2232C
- DRAM 8x8mbit
- Klávesnice 4x4
- Znakový LCD displej 1 x 16 znaků
- Rozhraní USB (konektor typ B)

- Sériové rozhraní RS-232 (konektor CANNON 9)
- 2x konektor PS2
- Rozhraní VGA (konektor CANNON 15)
- Audio vstup a výstup (konektor JACK 3,5 mm stereo)

2.1.1 MCU

Na kitu je osazen nízkopříkonový signálový mikrokontrolér **MSP430F168** od firmy Texas Instruments. Tento typ je určen speciálně pro provoz na baterie.

Přehled jeho vlastností:

- Nízký příkon (5 režimů příkonu)
 - 330 μ A v aktivním režimu při 1 MHz a 2,2 V
 - 1,1 μ A ve stand-by režimu
 - 0,2 μ A v režimu vypnuto
- 16-bitová RISC architektura
- Instrukční cyklus 125 ns
- Paměť
 - Flash 48 kB + 256 B
 - RAM 10 kB (podle [7] jen 2 kB)
- 2 x 16-bitový časovač
- 12-bitový 8-kanálový A/D převodník
- 12-bitový 2-kanálový D/A převodník
- 2 x asynchronní sériové rozhraní UART

Programování mikrokontroléru se provádí klasicky v jazyce C a zdrojové kódy se překládají překladačem MSPGCC [8] od Texas Instruments. Pouzdro čipu má 64 pinů. Pro úspěšné naprogramování MCU musíme mít nainstalované ovladače pro USB-UART převodník a správně nastavené parametry a čísla sériových portů převodníku v souboru ...\`base\Settings.inc`.

2.1.2 FPGA

Programovatelné hradlové pole FPGA (anglicky *Field-Programmable Gate Array*) **XC3S50-4PQ208C** řady **Spartan 3** od firmy Xilinx tvoří spolu s MCU srdce FITkitu. FPGA obsahuje funkční bloky a propojovací síť, s jejichž pomocí můžeme vytvořit požadovanou funkci. Konfigurace FPGA (nebo jen její část) se může měnit i za běhu aplikace.

- Rozhraní
 - Až 124 uživatelských vstupů/výstupů

- Podpora až 23 různých I/O standardů
- 192 konfigurovatelných logických bloků (CLBs) uspořádaných do matice o 16 řádcích a 12 sloupcích
- 1728 logických buněk
- 50 000 logických hradel
- Paměť RAM
 - 12 kb distribuovaných
 - 72 kb v jednom bloku
- 2 jednotky pro správu hodin (DCMs)
- 4 násobičky 18x18 bitů

2.1.2.1 VHDL

FPGA se programuje ve jazyce VHDL (*V*ery *H*igh *S*peed *I*ntegrated *C*ircuit *H*ardware *D*escription *L*anguage), což je jazyk pro popis hardwaru. Tento jazyk je standardizován organizací IEEE a poskytuje tak širokou kompatibilitu.

Strukturu HW ve VHDL můžeme popsat třemi způsoby:

- Popis chování (behavioral)
- Popis struktury (architecture)
- Popis datový toků (dataflow)

Všechny komponenty se pak propojí přes jejich rozhraní, které definuje vstupy a výstupy komponent.

2.1.3 USB-UART převodník

Převodník **FT2232C** (viz. obrázek 2.3) od firmy FTDI (Future Technology Devices International Ltd.) zprostředkovává komunikaci mezi MCU a PC přes sběrnici USB. Jak bylo již dříve uvedeno MCU je vybaven dvěma asynchronními sériovými porty a USB rozhraní nemá. V mnoha případech se USB rozhraní převede na asynchronní sérové rozhraní RS-232, které je na starších i novějších mikrokontrolérech dostupné a také SW podporované. V počítači se pak připojené zařízení přes USB detekuje jako dva USB sériové porty COM *x* a COM *y*. Podmínkou je nainstalování příslušných ovladačů [9]. Návod na instalaci ovladačů naleznete v [1] v sekci návody.



Obrázek 2.3

Obslužná aplikace na PC nebude rozlišovat, zda komunikuje se skutečným sériovým portem nebo s virtuálním sériovým portem přes USB sběrnici, protože v obou případech se využívá volání funkcí z jádra operačního systému. Často používané funkce jsou např. `CreateFile(...)`, `CloseHandle(...)`, `ReadFile(...)`, `WriteFile(...)` deklarované v souboru `windows.h`.

2.1.4 Klávesnice a LCD displej

Klávesnici 4x4 klávesy lze použít pro ovládání a výběr požadovaných funkcí k jednotlivým čidlům, které se budou zobrazovat na 1-řádkovém LCD displeji:

- maximální/minimální teplota
- průměrná teplota
- název/umístění čidla
- zobrazení volné/obsazené paměti
- nastavení intervalu měření

K ovládání displeje a získání stavu klávesnice jsou již vytvořeny obslužné rutiny, takže odpadá práce s programováním těchto periférií. Pouze pro speciální zobrazení informací, jako je průběh teploty, bude potřeba vytvořit speciální sadu znaků.

2.2 Rozhraní USB

USB – Universal Serial Bus, je v současné době nejrozšířenější sběrnice pro připojení externích zařízení k počítači.

Hlavní výhody:

- Jednoduchá sériová obousměrná komunikace
- Napájení zařízení ze sběrnice
- Vysoká rychlost komunikace až 480Mb/s (USB 2.0)
- Nízká cena
- Podpora více zařízení

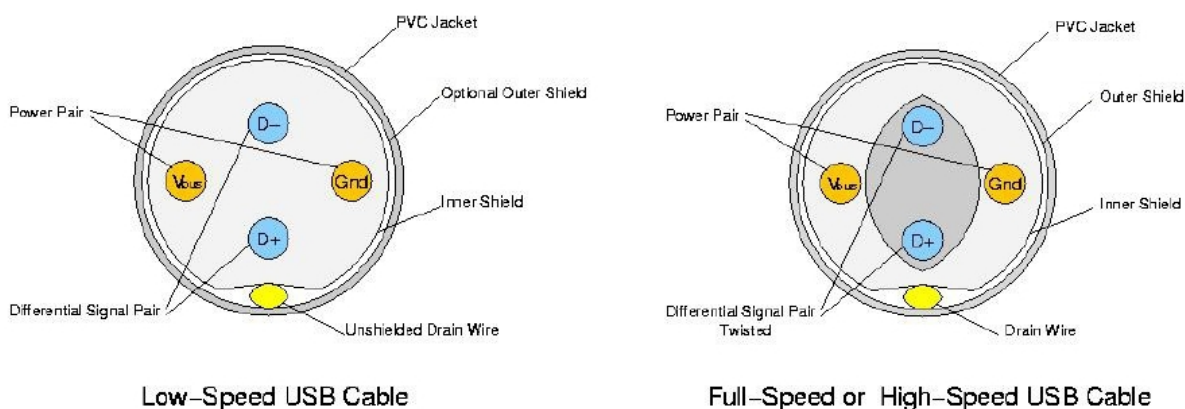
- Podpora Hot-Plug
- Automatická detekce a konfigurace zařízení
- Možnost rozšíření sběrnice

2.2.1 Rychlost USB

Standard USB definuje tři rychlosti přenosu:

- Low Speed – 1,5 Mb/s (USB 1.x)
- Full Speed – 12 Mb/s (USB 1.x)
- High Speed – 480Mb/s (USB 2.0)

Pro komunikaci FS (Full Speed) nebo HS (High Speed) by zařízení měla být propojena patřičným kabelem viz. obrázek 2.4. vpravo (diferenciální datové vodiče jsou kroucené). Maximální délka tohoto typu kabelu může být 5 m (propagační zpoždění 26 ns), pro LS kabel je maximální délka 3 m (zpoždění 18 ns).



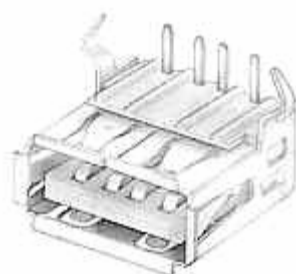
Obrázek 2.4

2.2.2 Konektory

Pro připojení zařízení USB se v praxi používají dva základní typy konektorů a jeden malý USB-mini (Mini-B) pro zařízení, kam by se klasický konektor z důvodu místa nevlezl (mobilní telefony, digitální fotoaparáty, ...) viz. obrázek 2.6.

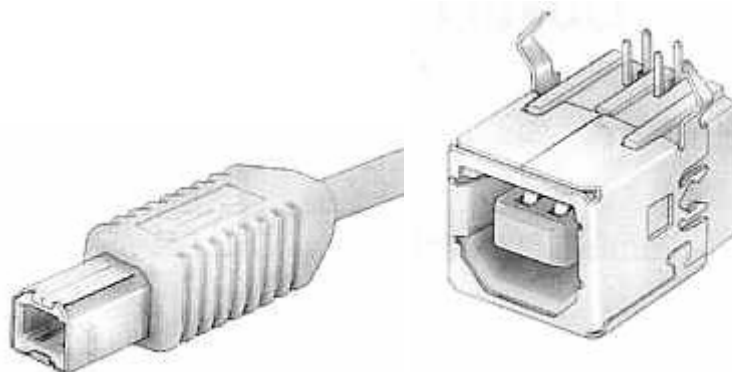


Obrázek 2.5



Obrázek 2.6

Na obrázku 2.5 je zobrazen konektor USB-A, ten bývá v naprosté většině na straně PC (samice). Protože je USB sběrnice natolik rozšířená, nalezneme na u současných stolních PC 6 a více těchto konektorů a toto množství je opravdu využitelné: myš, klávesnice, tiskárna, scanner, web-kamera, fotoaparát, mobilní telefon, ... a hned je 7 portů obsazených.



Obrázek 2.7

Na obrázku 2.7 je vyobrazen konektor USB-B, jenž je často vidět na straně periferního zařízení (samice) např. tiskárny, scanneru nebo vývojových kitů – včetně FITkitu.

2.2.3 Napájení USB

USB standard definuje napájení zařízení přímo ze sběrnice stejnosměrným napětím **5V**. Připojené zařízení může okamžitě odebírat proud až **100 mA**. Na požádání (přes ovladač) může odebírat maximální proud **500 mA**. Hostitelský počítač tedy musí zajistit napájení pro každý port 5V / 500 mA (pro 8 portů je to až 4A!).

Připojená zařízení dělíme na dva typy:

- Napájené ze sběrnice (**Bus-Powered**)
- S vlastním napájením (**Self-Powered**)

Pro nízkopříkonové zařízení jako je myš nebo klávesnice stačí napájení ze sběrnice, protože jejich proudová spotřeba nedosahuje ani 100 mA. Naopak některé náročnější externí disky potřebují externí napájecí zdroj, jedná se tedy o zařízení Self-Powered.

FITkit je kombinací obojí, jak Bus-Powered tak i Self-Powered – v případě vyššího příkonu (náročná výpočetní aplikace) je možné jej napájet z externího zdroje +5V. Externí napájení lze také použít v případě, kdy nepotřebujeme komunikovat s kitem.

2.2.4 Komunikace

Zařízení jsou s hostitelským PC propojena čtyřmi vodiči:

- Napájení +5V
- D+
- D-
- GND

Právě vodiče D+ a D- jsou datové komunikační vodiče, jež tvoří diferenciální pár. Diferenciální vedení redukuje šum (projeví se na obou vodičích stejně), ale zároveň se tím omezí komunikace pouze na half-duplex (komunikace pouze jedním směrem v jednom časovém okamžiku).

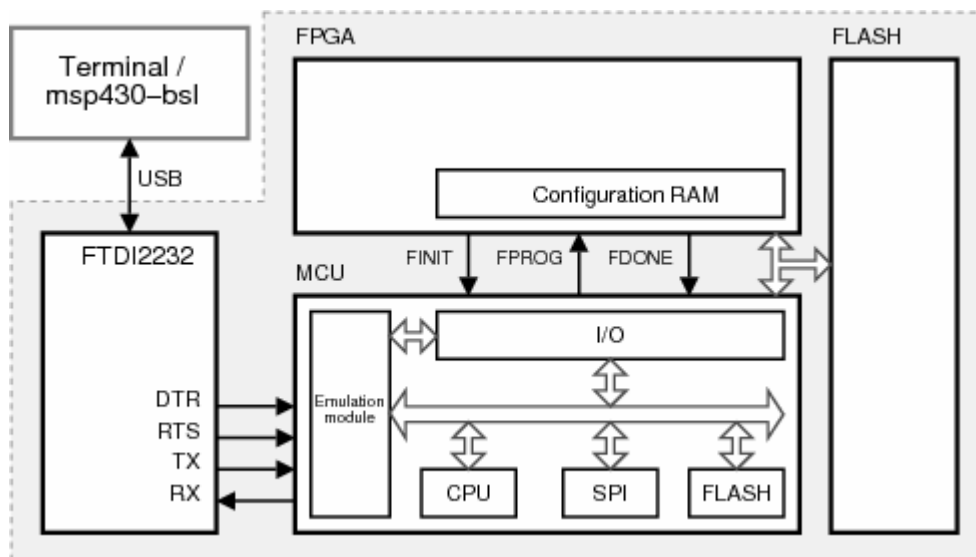
Ovladač komunikuje se zařízením prostřednictvím rour – komunikačních kanálů (pipes), přes které jdou čtyři druhy přenosů:

- Izochronní (*Isochronous*)
- Přerušovací (*Interrupt*)
- Řídící (*Control*)
- Hromadné (*Bulk*)

Jelikož ovladače čipu FT2232C převádí USB komunikaci na RS-232 (viz. kapitola 2.1.3) nemá smysl se podrobněji zabývat, více informací čtenář nalezne v [2] a [10].

2.3 Blokové schéma FITkitu

Základní blokové schéma FITkitu je vyobrazeno na obrázku 2.8.



Obrázek 2.8

2.3.1 Princip programování MCU a FPGA

Pro nahrání programu do MCU zadáme příkaz **make load** v adresáři se zdrojovým kódem v jazyce C (pro SVN je to .../sw) . Přeložený kód se zapíše do FLASH paměti v bloku MCU (viz. obrázek 2.8).

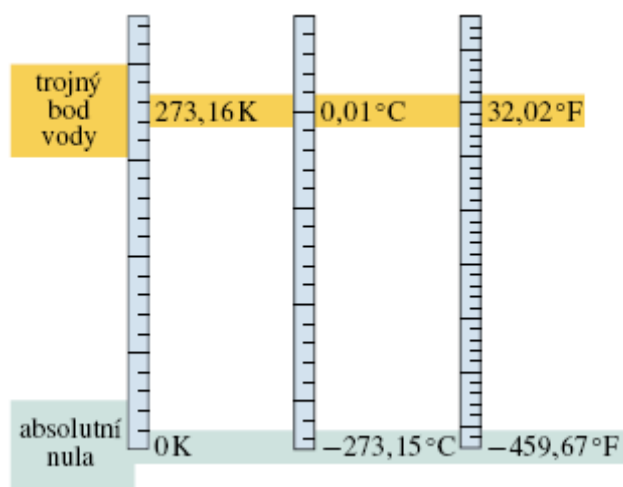
Pro nahrání konfigurace FPGA musíme ze zdrojového souboru *.vhd vytvořit (syntetizovat) binární soubor (output.bin) příkazem **make** v adresáři .../top. Poté spustíme komunikační terminál (HyperTerminal nebo PComm) pro komunikaci po sériové lince. V komunikačním programu nastavíme parametry sériových portů na: 460800 Bd, 8 bitů, 1 stop bit, bez parity. Otevřeme sériový port s vyšším číslem – do terminálu by se měla vypsát informace o FITkitu. V terminálu zadáme příkaz **prog fpga** a následně odešleme binární soubor (output.bin) protokolem Xmodem (1KCRC). A pokud nedošlo k chybě tak by mělo být FPGA naprogramované a požadovaný program běžet. Má to však jednu zásadní chybu – po odpojení napájení se konfigurace FPGA ztratí. Aby program pracoval i po opětovném připojení napájení musíme konfiguraci FPGA zapsat do FLASH paměti příkazem v terminálu **flash w fpga** a odeslat binární soubor. Pak ještě příkazem **prog fpga flash** nebo restartováním FITkitu zapíšeme konfiguraci z FLASH paměti do FPGA (to se také děje automaticky při startu FITkitu).

3 Měření teploty

Měření teploty je proces relativně pomalý, pokud bereme v úvahu běžné prostředí jako je např. teplota v místnosti nebo venkovní teplota vzduchu či vody. Na pomalé měření nám stačí menší výpočetní výkon (a tím také spotřeba zařízení). Výpočetní výkon FITkitu je v tomto ohledu zbytečně velký.

3.1 Teplota

Trojný bod vody – kapalná voda, pevný led a vodní pára mohou být spolu v tepelné rovnováze pouze při jediné teplotě a tlaku. Podle mezinárodní dohody je teplota trojného bodu stanovena na **273,16 K** [Kelvin]. Tato teplota se používá pro kalibraci teploměrů. Na obrázku 3.1 je uvedeno srovnání teplotních stupnic Kelvinovy, Celsiovy a Fahrenheitovy.



Obrázek 3.1

V našich zeměpisných šířkách se teplota udává ve stupních Celsia [°C]. Přepočet z Kelvinovy stupnice na stupnici Celsiovy udává následující vzorec:

$$T_C = T - 273,15^\circ\text{C},$$

kde T je teplota v Kelvinech

V Americe se pro změnu užívá stupnice ve Fahrenheitech [°F], přepočet na Fahrenheity ze °C určuje následující vztah:

$$T_F = \frac{9}{5}T_C + 32,$$

kde T_C je teplota ve °C

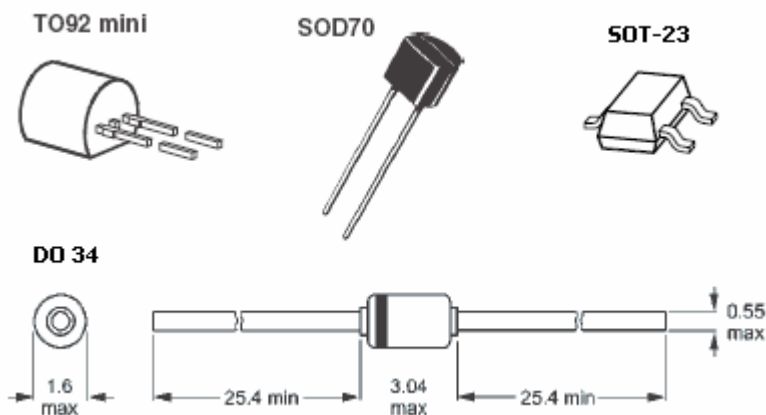
3.2 Teplotní čidla

Teplotní čidla teplotu převádějí obvykle na jiné fyzikální veličiny: odpor, frekvenci, střídu.

Veškeré ceny součástek zde uvedené jsou orientační a platné v roce 2006-2007 z nabídky firmy GM Electronics [4], která má dostatečný sortiment a cenově vychází lépe než konkurence. Teplotní rozsah polovodičových čidel se pohybuje v rozmezí $-50^{\circ}\text{C} \dots +150^{\circ}\text{C}$ a přesnost bývá od $\pm 0,5^{\circ}\text{C}$ do -5°C .

3.2.1 Odporová teplotní čidla

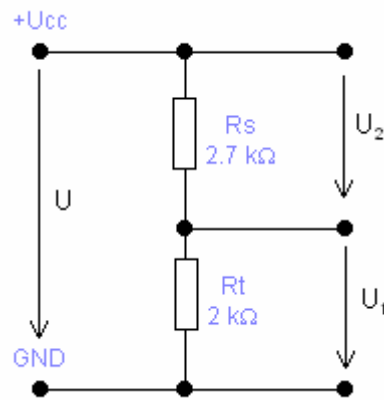
Jsou teplotně závislé součástky, jejichž odpor roste téměř lineárně se stoupající teplotou. Jsou poměrně levná, cena okolo 20Kč a přesnost obvykle $\pm 1^{\circ}\text{C} \dots \pm 2^{\circ}\text{C}$, což je z hlediska přesnosti vyhovující. Teplotní rozsah: $-55^{\circ}\text{C} \dots +150^{\circ}\text{C}$ ($+175^{\circ}\text{C}$). Jsou vyráběna v různých pouzdrech (viz. obrázek 3.2): SOD70, TO92 mini, DO 34 a SOT-23. Typicky mají odpor $1\text{ k}\Omega$ nebo $2\text{ k}\Omega$ při 25°C .



Obrázek 3.2

Čidlo lze nejjednodušším způsobem k FITkitu připojit jako odporový dělič viz obrázek 3.3., kde by se napětí U_1 přivedlo na vstup jednoho kanálu AD převodníku. R_s je sériový odpor, R_t je odporové teplotní čidlo, např. KTY81-210. Takovéto zapojení vyžaduje o něco náročnější výpočet teploty, než kdyby se čidlo připojilo do série se zdrojem konstantního proudu, protože musíme uvažovat i velikost proudu protékající čidlem, který se s teplotou mění.

K FITkitu je možné připojit až 8 takovýchto čidel (MCU má 8-kanálový 12-bitový A/D převodník). Vzhledem k tomu, že referenční napětí AD převodníku může být pouze 3,3V, nebyl by rozsah naměřených hodnot optimální (využilo by se jen asi 65% hodnot). A při použití dlouhých přívodních vodičů by mohlo případné rušení značně ovlivnit naměřenou hodnotu.



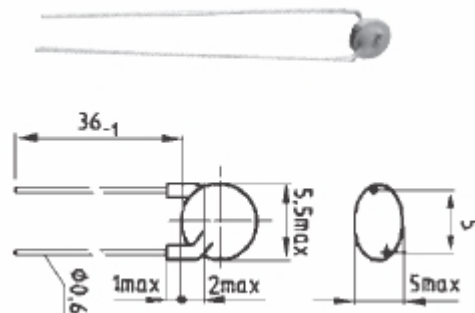
Obrázek 3.3

3.2.2 Termistory

Jsou teplotně závislé součástky – s teplotou se mění jejich odpor, ať již pozitivně nebo negativně. Teplotní charakteristika není lineární. Teplotní rozsah termistorů: $-55^{\circ}\text{C} \dots +125^{\circ}\text{C}$.

3.2.2.1 Termistory s negativní teplotní závislostí (NTC)

Hodnota odporu klesá se vzrůstající teplotou – proto negativní teplotní závislost. V nabídce jsou termistory s hodnotou od 100Ω do $100\text{k}\Omega$ a cena se pohybuje do 20Kč. Hodnota nominálního odporu se udává při 25°C a může se lišit až o 20%, což vyžaduje kalibraci. Na obrázku 3.4. je ukázka termistoru.



Obrázek 3.4

Termistory jsou sice levné, ale zase vyžadují kalibraci kvůli velkým odchylkám od nominální hodnoty. Připojení k FITkitu je možné pouze přes jednoduchý odporový dělič – převod teploty na napětí (měření AD převodníkem) nebo vytvořit převodník na frekvenci či střídě, což by znamenalo zvýšení ceny teplotní čidla a také by vnášelo nepřesnost do výsledné hodnoty.

3.2.2.2 Termistory s pozitivní teplotní závislostí (PTC)

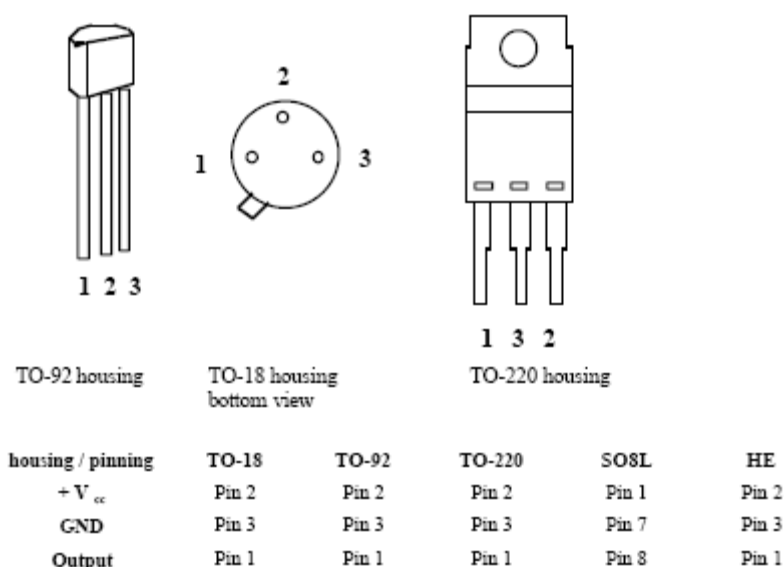
Hodnota odporu roste nelineárně se vzrůstající teplotou. Jelikož nejsou k dostání, tak se dále jimi zabývat nebudu.

3.2.3 Převodník teplota/střída

Toto čidlo, **SMT 160-30**, na výstupu generuje obdélníkový signál o frekvenci 1-4 kHz, jehož střída se mění lineárně s teplotou. Pracuje na napětí +5V a má příkon menší jak 0,1 mW. Úroveň generovaného signálu je kompatibilní s logickými úrovněmi TTL a CMOS, takže jej můžeme přímo připojit na digitální vstup mikroprocesoru nebo na vstup FPGA. Střída (*DC – Duty Cycle*) se mění podle následujícího vztahu:

$$DC = 0,32 + 0,0047t,$$

kde t je teplota v °C



Obrázek 3.5

Teplotní rozsah: -45°C...+130°C, přesnost: ±0,7°C, nelinearita převodu: 0,2°C. Čidlo je kalibrováno při výrobě, takže odpadá problém s nastavením správné teploty. Za tento čip zaplatíme asi 83 Kč. Více informací je uvedeno v pramenech [5] a [6].

Počet těchto čidel připojitelných k FITkitu je omezen především počtem vstupních pinů. V této aplikaci mohou být čidla přepínána multiplexorem implementovaným v FPGA – tím bude stačit jen jeden blok pro měření střídly.

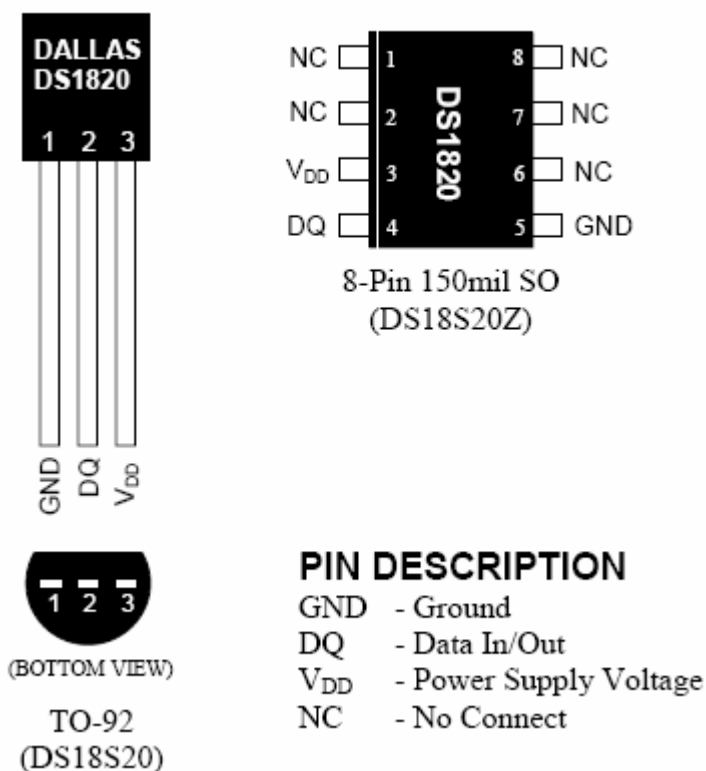
3.2.4 Digitální teplotní čidlo

Firma DALLAS Semiconductors vyrábí teplotní čidla s digitálním přenosem teploty do cílového zařízení přes sběrnici **1-Wire**. Máme k dispozici dva typy: **DS18B20** a **DS18S20**.

DS18S20 – 9-bitové rozlišení teploty (cena 71 Kč)

DS18B20 – programovatelné rozlišení 9–12 bitů (cena 60 Kč)

PIN ASSIGNMENT



Obrázek 3.6

Několik takovýchto čidel můžeme napojit paralelně na 3-vodičové vedení (+U_{CC}, data, GND), čímž ušetříme na kabeláži. Rozsah měřené teploty: -55°C...+85°C (+125°C), napájecí napětí: +3,0V...+5,0V, přesnost ±0,5°C v rozmezí teplot -10°C...+85°C. Připojení k FITkitu připadá na vstupy FPGA, ve kterém budeme muset implementovat komunikační protokol 1-Wire. Převod teploty v 12-bitovém rozlišení trvá 750ms, což ale v našem případě příliš nevádí.

3.2.4.1 Sběrnice 1-Wire

Sběrnice **1-Wire** je unikátní v tom, že po jediném datovém vodiči probíhá komunikace oběma směry. Na sběrnici musí být jeden prvek *BUS-Master*, který sběrnici řídí, a jeden nebo více prvků *slave*. Čidla se mezi sebou rozlišují pomocí svého unikátního 64-bitového čísla zapsaného v ROM (viz obrázek 3.7).

64-BIT LASERED ROM CODE Figure 6



Obrázek 3.7

Pro komunikaci je velmi důležité časování a dodržování ochranných intervalů.

4 Analýza a návrh

4.1 Analýza

Pokud vezmeme v úvahu rozsah teplot, jenž teplotní čidla dokážou změřit, tedy $-50^{\circ}\text{C}\dots+150^{\circ}\text{C}$, dostaneme celkový měřený rozsah 200°C . Rozlišení zvolíme $0,1^{\circ}\text{C}$, což by mělo dostačovat. Pro uložení jedné naměřené teploty tedy potřebujeme 2000 hodnot (200°C s rozlišením $0,1^{\circ}\text{C}$). V binární soustavě na to bude stačit 11 bitů ($2^{11} = 2048$ hodnot). $2048 > 2000$ takže nám ještě zbude místo pro hodnoty se speciálním významem, např. nepřipojeno čidlo apod.

4.1.1 Uložení čísel

Pro 16-bitovou architekturu MCU se uložení 11-bitového čísla jeví jako neefektivní, zvláště v tomto případě, kdy potřebné paměti nebude nikdy dost.

Řešení nabízí FPGA, ve kterém si můžeme nadefinovat 11-bitový registr (případně pole registrů) pro uložení hodnot a registry budou plně využité. Sadu osmi 11-bitových čísel je možné uložit do paměti MCU těsně za sebou (bitově) na 11 bajtech. Kdyby byly hodnoty uloženy bez kódování, zabíralo by 8 hodnot teploty 16 bajtů, tím dostaneme úsporu 5 bajtů.

4.1.2 Komprimace hodnot

Protože se dá předpokládat, že naměřené vzorky budou dosti podobné, někdy dokonce i stejné, bude možné je slušným komprimačním poměrem zakódovat.

Podle získaných statistik se rozhodnu pro některou z následujících kódovacích principů:

- Kódování délek sledů
- Huffmanovo kódování
- Diferenciální kódování

Podle mého názoru by Diferenciální kódování mohlo přinést největší úsporu potřebné paměti. Huffmanovo kódování je implementačně poměrně složité a výsledek by nemusel být znatelný. Kódování délek sledů by přineslo dobrou úsporu místa, pokud by se teplota zůstala po dlouho dobu stabilní.

Jelikož lze předpokládat, že naměřený vzorek bude zatížen nějakou chybou, bude vhodné odečíst např. 10 vzorků za sebou a z nich vypočítat průměr. Z 10 vzorků odstraníme vzorek s nejvyšší a nejmenší hodnotou a zbylých 8 vzorků podělíme 8. Tak by měla být naměřená teplota co nejpřesnější.

4.2 Princip činnosti

Po připojení napájecího napětí se FITkit inicializuje – zjistí počet připojených teplotních čidel. Systém bude řízen několika hodinovými signály, které se budou značně lišit v periodě. Ty nejpomalejší budou sloužit k odečítání teplot z jednotlivých čidel – po zadaném intervalu se odečtou hodnoty všech teplotních čidel a uloží se do paměti. Podle indexu uložené hodnoty lze zpětně spočítat doba ve které byla teplota změřena (pokud známe dobu periody).

Při přenosu dat z kitu se musí někde poznačit reálný čas a datum zahájení záznamu hodnot a odeslání všech hodnot do PC. Implementovat systém reálného času přímo ve FITkitu se mi jeví jako příliš složité, když tuto úlohu může převzít počítač. Řešení kalendáře, počet dní v měsíci, přestupné roky a posun letního a zimního času taktéž bude zajišťovat aplikace na straně PC.

Jakmile bude dostatek hodnot ke komprimaci, provede se komprimace nasbíraných dat, aby se ušetřilo místo v paměti pro další vzorky a byla tak maximalizována doba provozu bez připojení PC. Na displeji se také bude průběžně zobrazovat procento volné a obsazené paměti a predikce na kolik dní ještě paměť vydrží.

4.3 Komunikační protokol

Stahování naměřených dat do PC bude řízeno vlastním protokolem, založeným na příkazech. Tyto příkazy budou ve formě smysluplných ASCII řetězců posílaných po sériové lince. Příkazy také mohou sloužit k řízení a nastavení intervalu měření. MCU bude monitorovat přijímané znaky ze sériového portu a pokud nějaký řetězec bude odpovídat příkazu patřičně na něj zareaguje.

5 Závěr

Práce je teprve v počátku, takže na závěrečné pojednání je trochu brzo. Nicméně oblast vestavěných systémů představuje zajímavou a perspektivní oblast z hlediska budoucnosti, kdy celá řada aplikací přechází do vestavěných systémů. V tématu se vyskytuje spousta věcí které byly na FIT vyučovány a dají se prakticky využít.

Literatura

- [1] FITkit, url: <http://merlin.fit.vutbr.cz/FITkit/>
- [2] Přednášky předmětu NAV, url: <https://www.fit.vutbr.cz/study/courses/NAV/private/>
- [3] Halliday D., Resnick R., Walker J.: FYZIKA. ISBN 80-214-1869-9
- [4] Katalog elektronických součástek GME, url: <http://www.gme.cz>
- [5] Odkaz na katalogový list SMT 160-30 http://www.smartec.cz/pdf/SMT_160-30_CZ.pdf
- [6] Informace o SMT 160-30, url: <http://www.hw.cz/Produkty/Nove-soucastky/ART391-Prevodnik-teplota-strida-SMT160-30-92.html>
- [7] Katalogový list MSP430F168, url: <http://www.ti.com/lit/gpn/msp430f168>
- [8] MSPGCC kompilátor, url: http://sourceforge.net/project/showfiles.php?group_id=42303
- [9] Ovladače pro čip FT2232C, url: <http://www.ftdichip.com/Drivers/VCP.htm>
- [10] HW server USB 2.0, url: odkaz na <http://www.hw.cz/Rozhrani/ART1232-USB-2.0---dil-1.html>