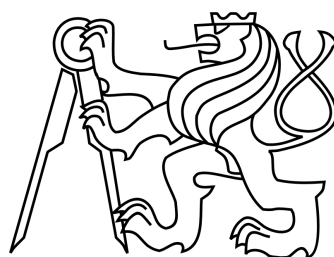




ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
FAKULTA ELEKTROTECHNICKÁ  
KATEDRA ŘÍDÍCÍ TECHNIKY



## BAKALÁŘSKÁ PRÁCE

Monitoring teploty

**Autor:** Martin Zmrhal

**Vedoucí práce:** Ing. František Vaněk

Praha, 2010

**Název práce:** Monitoring teploty

**Autor:** Martin Zmrhal

**Katedra:** Katedra řídicí techniky

**Vedoucí bakalářské práce:** Ing. František Vaněk

**e-mail vedoucího:** vanek@fel.cvut.cz

**Anotace** V předložené práci je pojednáváno o možnostech měření teploty v serverovně a následném sběru teplotních dat do PC. Bylo vybráno komerčně dostupné řešení pomocí sběrnice 1-Wire od firmy Maxim Integrated Products a sestaveno zařízení pro sběr dat z teplotních senzorů. Toto zařízení umožňuje přímé připojení k PC, pro které byla napsána aplikace zpracovávající data a umožňující tvorbu grafů teploty pomocí aplikace RRDtool. Následně byl celý měřicí systém zapojen v serverovně budovy FEL na Karlově náměstí, kde bude sloužit pro dlouhodobý monitoring teploty.

**Klíčová slova:** monitoring teploty, teploměr, 1-Wire, DS18B20

---

**Title:** Temperature monitoring

**Author:** Martin Zmrhal

**Department:** Department of Control Engineering

**Supervisor:** Ing. František Vaněk

**Supervisor's e-mail address:** vanek@fel.cvut.cz

**Abstract** In the present work we study possibilities of temperature measuring in a server room and subsequent temperature data collecting and passing the data into a PC. A commercially available solution using 1-Wire bus from Maxim Integrated Products was chosen and a device for temperature sensors data collecting was constructed. This device allows a direct connection to a PC for which a data processing application was written. This application uses the RRDtool utility for creating temperature plots. The whole measuring system was then set up in the server room in the FEE building at Charles Square in Prague, where it will serve for long-term temperature monitoring.

**Keywords:** temperature monitoring, thermometer, 1-Wire, DS18B20

# ZADÁNÍ

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW, atd.) uvedené v příloženém seznamu.

V Praze dne 27. května 2010

Martin Zmrhal

# OBSAH

<b>Anotace</b>	<b>ii</b>
<b>Zadání práce</b>	<b>iii</b>
<b>1. Úvod</b>	<b>1</b>
<b>2. Výběr vhodného způsobu měření teploty</b>	<b>2</b>
2.1. Kontaktní teploměry . . . . .	2
2.1.1. Termočlánky . . . . .	2
2.1.2. Odporové snímače teploty . . . . .	3
2.1.3. Integrované obvody pro měření teploty . . . . .	4
2.2. Výběr vhodného čidla . . . . .	5
<b>3. Čidlo DS18B20</b>	<b>8</b>
3.1. Popis vlastností čidla DS18B20 . . . . .	8
3.2. Blokové schéma . . . . .	9
3.3. Vnitřní paměť . . . . .	9
3.4. Měření teploty . . . . .	10
3.5. Komunikace s čidlem DS18B20 . . . . .	11
3.5.1. Inicializační sekvence . . . . .	12
3.5.2. ROM příkazy . . . . .	13
3.5.3. Funkční příkazy . . . . .	14
3.6. Shrnutí . . . . .	15
<b>4. Zařízení pro řízení sběrnice 1-Wire</b>	<b>16</b>
4.1. Sběrnice 1-Wire . . . . .	16
4.2. Řídící členy sběrnice . . . . .	17
4.3. Pasivní adaptér DS9097 . . . . .	18
4.4. Aktivní adaptér DS9097U . . . . .	19
4.5. Stavba vlastního 1-Wire adaptéru . . . . .	20
4.5.1. Popis a schéma adaptéru . . . . .	20
4.5.2. Konstrukce adaptéru . . . . .	22

---

<b>5. Software pro komunikaci po sběrnici 1-Wire</b>	<b>24</b>
5.1. Seznam 1-Wire API . . . . .	24
5.2. Tvorba softwaru . . . . .	25
5.3. Popis vlastní aplikace . . . . .	27
<b>6. Realizace monitorovacího systému</b>	<b>32</b>
6.1. Instalace softwaru a zapojení 1-Wire sítě . . . . .	32
6.2. Reálné měření teploty . . . . .	33
<b>7. Závěr</b>	<b>36</b>
<b>Literatura</b>	<b>39</b>
<b>Přílohy</b>	<b>I</b>
<b>A. Fotografie 1-Wire adaptéru</b>	<b>II</b>

## ÚVOD

Teplota je bezesporu jednou z nejčastěji měřených neelektrických veličin. Příčinou je závislost mnoha mechanických, elektromagnetických i chemických jevů na teplotě a především problémy vzniklé s vysokými, případně nízkými teplotami. Například na vývoji počasí nás teplota odjakživa zajímá nejvíce – lidské tělo není stavěno do extrémních teplot a nepřipravenost na klimatické podmínky nám může způsobit vážné zdravotní komplikace. Stejně jako vliv teploty okolí na lidské tělo nás zajímá vliv teploty na různé druhy elektronických zařízení, právě protože mohou být teplotou také nenávratně zničena. Teplota je tak jedním z nejdůležitějších parametrů pro řízení většiny technologických a výrobních procesů, přičemž její přesné a optimální monitorování umožňuje zajistit bezpečnost a kvalitu výroby. S velmi vysokou teplotou také úzce souvisí možnost vzniku požáru a proto jsou často součástí moderních budov teplotní senzory sledující teplotu za účelem prevence nebezpečí.

Stejně tak bylo mým úkolem při zpracovávání této práce navrhnout systém monitorující teplotu na několika místech v serverovně v budově FEL na Karlově náměstí. Požadavkem bylo komplexní řešení zahrnující výběr vhodných senzorů, sestavení zařízení zajišťujícího přenos dat do PC, zprovoznění systému v serverovně a následné sbírání a zobrazení dat v grafech vytvořených pomocí programu RRDtool.



## VÝBĚR VHODNÉHO ZPŮSOBU MĚŘENÍ TEPLoty

Prvním krokem návrhu měřicího systému byl výběr vhodného čidla pro snímání teploty vzduchu v serverovně. Čidla pro měření teploty lze v zásadě rozdělit na dva základní typy: kontaktní a bezkontaktní. Bezkontaktní teploměry (tzv. pyrometry) fungují na principu měření teplotního záření těles a nejsou proto vhodné pro měření teploty vzduchu.

### 2.1. Kontaktní teploměry

Kontaktní teploměry fungují na principu měření jejich vlastní teploty. Měření teploty vyplývá z předpokladu, že senzor je v kontaktu s měřeným objektem a probíhá vzájemná tepelná výměna. Po vytvoření tepelné rovnováhy je teplota senzoru rovna teplotě objektu a je z něj možno odečíst údaj o teplotě. Tepelná výměna probíhá i mezi senzorem a vzduchem a proto jsou pro měření teploty vzduchu v serverovně ideálním řešením.

#### 2.1.1. Termočlánky

Termočlánky využívají principu termoelektrického (Seebeckova) jevu. Při tomto jevu vzniká napětí při teplotním rozdílu mezi dvěma odlišnými kovy<sup>1</sup>. Toto tzv. termoelektrické napětí se pohybuje v řádech mikrovoltů (u nejběžnějšího typu termočlánků je citlivost přibližně  $40 \mu\text{V}/^\circ\text{C}$ ) a proto je nutné jej dále zesilovat. Výhodou je, že termoelektrické sondy nepotřebují samostatné napájení. V praxi se pro měření teploty používá zapojení se srovnávacím koncem. Termočlánek se pak skládá z měřicího konce, který přímo měří teplotu a ze srovnávacího, tzv. „studeného“ konce, který je umístěn v prostředí se známou teplotou. Vzniklý teplotní rozdíl je nutné kompenzovat, nejlépe dalším teplotním čidlem (nejčastěji se používají jednoduché polovodičové senzory – diody, tranzistory – viz. sekce 2.1.3 Integrované obvody pro měření teploty). Pro zpraco-

---

<sup>1</sup>Popsán v roce 1821 německým fyzikem Thomasem Johannem Seebeckem

vání výstupního napětí termočlánu je možné použít řadu na trhu běžně dostupných zesilujících obvodů s vlastní kompenzací studeného spoje s analogovým výstupem (např. AD595 – lineární výstup 10 mV/°C), případně obvody s digitálním výstupem (MAX6674).

Termočlány jsou vhodné tam, kde je požadováno měření velkého rozsahu teplot. Jsou rozděleny do skupin podle použitých kovů a měří v rozsazích od –270 °C do 2300 °C s přesností kolem 1% měřicího rozsahu. Jsou proto nejčastěji používány v průmyslových aplikacích a jejich ceny se pohybují v rozmezí od několika set do několika tisíc korun.

### 2.1.2. Odporové snímače teploty

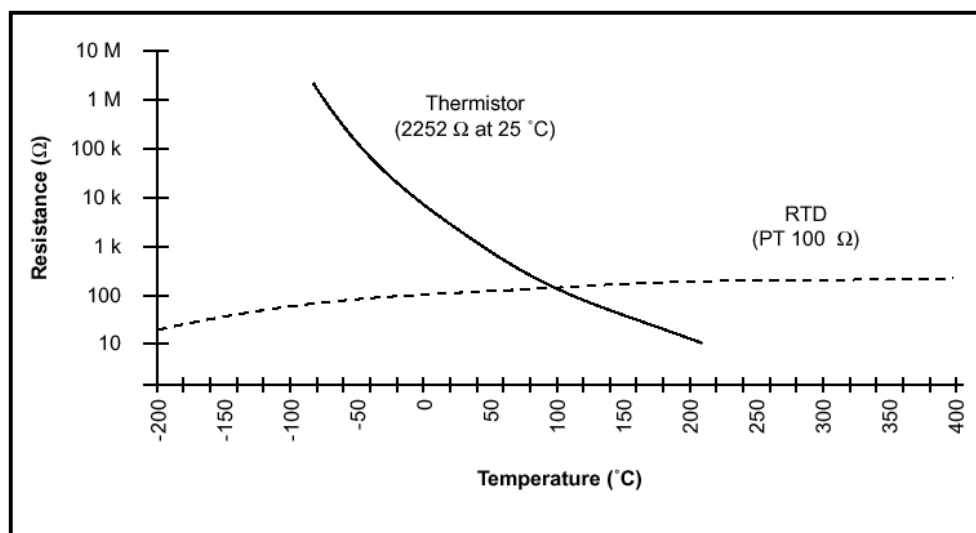
Odporové snímače teploty mění svůj elektrický odpor podle teploty. Hodnotu odporu je potom nutné vyhodnocovat například mikroprocesorem.

#### Odporové snímače teploty typu Pt100

Někdy jsou také označovány jako RTD (z angl. resistance temperature detector). Jejich typický odpor je 100Ω, případně 1000Ω při teplotě 0 °C a jsou nejčastěji vyrobeny z platinové vrstvy napařené podkladové destičce (proto označení Pt). Fungují na principu změny elektrického odporu kovu na základě změny jeho teploty. Zvýšení teploty totiž způsobuje zrychlení kmitů nečistot v krystalové mřížce kovů, tímto zpomaluje průchod elektronů a snižuje vodivost. V měřicím rozsahu lze tento jev přibližně linearizovat. Čidla musí být připojena na přesný zdroj napětí nebo proudu, aby jimi mohl protékat proud a odpor byl přesně měřitelný. Problémem může být odpor dlouhých přívodních vodičů, proto se často používá vícevodičové zapojení. Velkou výhodou těchto čidel je z běžně dostupných řešení zřejmě nejvyšší přesnost napříč celým měřicím rozsahem (0,5% měřené hodnoty), velký rozsah měřených teplot (–200 °C až 800 °C), přibližná linearita a absence nutnosti kalibrace [1][2]. Cenou za tyto vlastnosti jsou vyšší pořizovací náklady na jeden senzor (cca 800 Kč za standardní čidlo Pt100) [3].

#### Termistory

Termistory jsou na rozdíl od RTD snímačů vyrobeny z oxidů různých kovů. Mají buď pozitivní teplotní koeficient, kdy odpor roste úměrně s teplotou (označovány jako PTC (positive temperature coefficient)), nebo negativní teplotní koeficient, kdy odpor s teplotou klesá (NTC – negative temperature coefficient). Nevýhodami termistorových čidel je menší rozsah měřitelných teplot (–90 °C až 130 °C), značná nelinearita a nutnost kalibrace. Porovnání linearity průběhu odporu v závislosti na teplotě lze vidět na obrázku 2.1 na následující straně. Na druhou stranu mají velký vnitřní odpor, takže mají malý odběr proudu a nejsou téměř ovlivněny odporem přívodních vodičů. Navíc vykazují přesnost dostačující pro běžné aplikace (na koncích rozsahu cca ±5–10%, uprostřed rozsahu přibližně 1%) a jsou ze všech teplotních čidel nejlevnější [1].



Obrázek 2.1.: Porovnání závislosti odporu na teplotě u termistorů a čidla Pt100

### 2.1.3. Integrované obvody pro měření teploty

Základním principem těchto obvodů je fakt, že úbytek napětí na přechodu báze-emitor tranzistoru NPN je v určitém rozsahu závislý na teplotě. Pokud se použije zapojení s dvěma tranzistory s různou plochou emitoru jako na obrázku 2.2 na následující straně a na jejich kolektory jsou přivedeny různé proudy, odpovídá rozdíl jejich napětí báze-emitor vztahu 2.1<sup>2</sup>:

$$\Delta V_{BE} = V_{BE1} - V_{BE2} = \frac{kT}{q} \ln \frac{J_{E1}}{J_{E2}}, \quad (2.1)$$

kde

$k$  je Boltzmannova konstanta

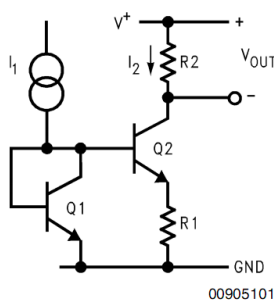
$T$  je termodynamická teplota

$q$  je náboj elektronu

$J_E$  je proudová hustota na emitoru tranzistoru

Pokud je tedy poměr proudů  $I_1$  a  $I_2$  konstantní, výstupní napětí je přímo úměrné teplotě. Tento průběh je na rozdíl od termistorů téměř lineární (v celém rozsahu  $-55\text{ °C}$  až  $150\text{ °C}$  je odchylka od linearit  $3\text{ °C}$  až  $4\text{ °C}$ ) a proto většina integrovaných čidel teploty využívá právě tohoto principu. Linearita bývá ještě dále kompenzována a čidla jsou často kalibrována přímo z výroby, což zvyšuje jejich přesnost a zjednodušuje použití [4].

<sup>2</sup>Tohoto zapojení se používá k teplotní kompenzaci zdrojů referenčního napětí ve většině integrovaných obvodů a je známo jako Brokawovo zapojení (bandgap reference).



Obrázek 2.2.: Schéma zapojení dvou tranzistorů jako zdroje napětí úměrného teplotě

Výstupem těchto čidel je buď napětí odpovídající teplotě, nebo je toto napětí dále zpracováváno: čidla obsahují převodníky napětí/frekvence, generují pulzní šířkovou modulaci, kde teplotu reprezentuje střída, případně obsahují integrovaný A/D převodník a výstupem je pak digitální informace o teplotě. Signál z čidel je potom možné zpracovávat v mikroprocesorech bez použití samostatného A/D převodníku. Mikroprocesor jednoduše změří šířku pulzu, frekvenci, nebo přečte datovou posloupnost. Velkou výhodou těchto čidel je tedy snadné zpracování signálu, poměrně vysoká přesnost, z výroby korigovaná nelinearita a především velmi příznivá cena plynoucí z principu výroby polovodičových součástek. Většina čidel se proto pohybuje v cenové hladině do 200 Kč za kus.

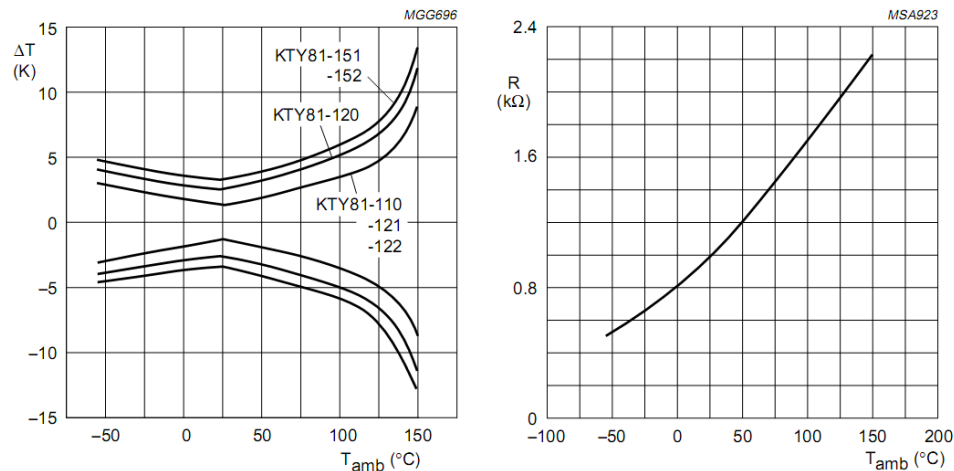
## 2.2. Výběr vhodného čidla

Z důvodu jednoduché dostupnosti jsem vhodné čidlo vybíral pouze z nabídky pražských firem GES-ELECTRONICS a.s. a GM Electronic s.r.o.. Bylo nutné brát v úvahu, že při monitoringu bude použito několik čidel, takže cena za jeden kus byla důležitým rozhodovacím faktorem. Jedním z požadavků na zpracování byl měřicí rozsah  $-20\text{ }^{\circ}\text{C}$  až  $100\text{ }^{\circ}\text{C}$  a tudíž jsem vybíral čidla s co největší garantovanou přesností v tomto rozsahu. Neméně důležitým faktorem byla možnost sběru dat ze všech čidel najednou a způsob následného přenosu těchto dat do PC.

### Odporové snímače teploty

Termočlánky ani odporové snímače teploty Pt100 žádná z těchto firem nenabízí a navíc by pro vysokou cenu tato čidla nebyla vhodná. Jiná situace je u termistorů, kterých je na trhu velké množství. NTC termistory stojí většinou do 20 Kč, ale jejich linearita a nepřesnost měření jsou příliš velké na přesné sledování teploty. Obě firmy nabízejí PTC termistory typu KTY81 a KTY83 (liší se provedením pouzdra) od firmy Philips Semiconductors, jejichž jednotlivé verze se od sebe liší nominální hodnotou odporu při teplotě  $25\text{ }^{\circ}\text{C}$  a proudu 1 mA. Ceny se pohybují okolo 20 Kč. Na obrázku 2.3 na následující straně je vlevo zobrazena chyba měření teploty u čidla

typu KTY81 a vpravo jejich typická linearita. Při teplotě 25 °C je chyba měření  $\pm 1,27$  °C a tato chyba roste s pohybem teploty oběma směry [5]. Odpor čidla by bylo nutné vyhodnocovat mikroprocesorem a potom přes odesílat tyto hodnoty do PC. Při zvážení chyby měření a nelinearity jsem tyto čidla z výběru vyloučil.



Obrázek 2.3.: Teplotní charakteristiky čidel KTY81

### Integrované obvody

Další možností bylo použít integrovaný obvod s analogovým napěťovým výstupem úměrným teplotě. Typickým příkladem je obvod LM335 v pouzdru TO92 od firmy National Semiconductor, jehož cena je kolem 15 Kč. Měřicí rozsah tohoto čidla je od -40 °C do 100 °C. Čidlo je kalibrováno v Kelvinech a jeho převodní konstanta napětí/teplota je 10 mV/K, přičemž při absolutní nule (0 K) je výstupní napětí 0 V. Tyto senzory mají korigovanou nelinearitu, s typickou/maximální odchylkou 0.3/1.5 °C a je možné jednoduše dodatečně kalibrovat případnou multiplikativní chybu. Po kalibraci na 25 °C je typická/maximální chyba tohoto čidla 1/2 °C [6]. Velmi podobné je čidlo LM35DZ od stejného výrobce, které je kalibrováno ve stupních Celsia a je u něj zlepšena linearita (rozsah je 0 °C až 100 °C a přesnost je podobná čidlu LM335) [7]. Jeho cena je však ve srovnání s LM335 téměř třikrát vyšší (kolem 45 Kč). Signál z tohoto typu čidel by bylo nutné zpracovávat A/D nebo U/f převodníkem a posléze mikroprocesorem posílat data do PC.

Dalším z integrovaných obvodů je převodník teplota/střída SMT160-30 v pouzdru TO92, jehož cena je cca 90 Kč. Rozsah měření je od -45 °C do 130 °C při absolutní přesnosti  $\pm 2$  °C v celém rozsahu ( $\pm 1,2$  °C od -30 do 100 °C). Vykazuje rovněž velmi dobrou linearitu (maximální odchylka je 0,4 °C) a není zapotřebí žádné další kalibrace, protože je kalibrován přímo z výroby. Výhodou je rovněž výstup zpracovatelný jak analogově, tak digitálně mikroprocesorem.

rem bez potřeby A/D převodníku, pouze měřením délky pulzu. Střída výstupního signálu se vypočte dle vzorce 2.2 [8]:

$$DC = 0,32 + 0,0047 \cdot t, \quad (2.2)$$

kde

$t$  je teplota ve stupních Celsia

$DC$  je střída výstupního signálu o frekvenci 1–4 kHz

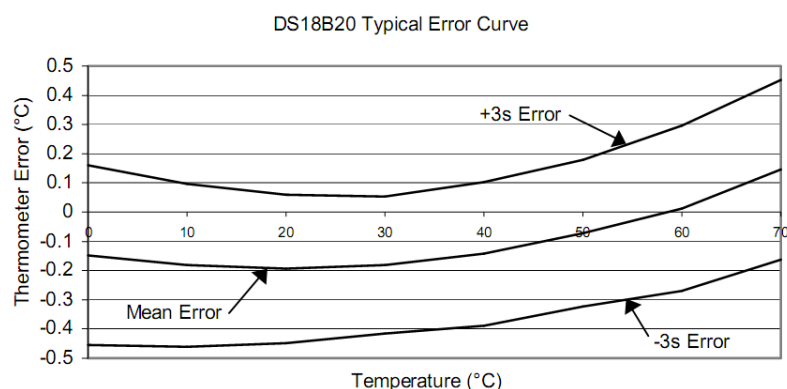
Přesnost tohoto senzoru a jeho linearita by již byly dostačující, nicméně cena je poněkud vysoká a pro připojení více čidel by bylo potřeba obsadit příslušný počet vstupů mikroprocesoru, případně použít multiplexor, což by se ovšem znamenalo vyšší složitost zapojení. Proto ani senzor SMT106-30 nebyl vhodným řešením.

Dalším běžně dostupným čidlem je obvod DS18B20 od firmy Dallas Semiconductor (nyní součást Maxim Integrated Products). Tento senzor má velmi vysokou přesnost měření, výbornou linearitu, je plně digitální a má nastavitelné rozlišení teploty. Cena kolem 50 Kč je vzhledem k parametrům a nabízeným funkcím přiměřená. Z těchto důvodů jsem pro monitoring teploty nakonec zvolil právě tento senzor a věnuji mu samostatnou kapitulu.

## ČIDLO DS18B20

### 3.1. Popis vlastností čidla DS18B20

Jedná se o čidlo teploty obsahující integrovaný A/D převodník převádějící teplotu na devíti- až dvanáctibitové slovo. Je kalibrováno ve stupních Celsia a jeho přesnost je  $\pm 0,5^\circ\text{C}$  v rozsahu  $-10^\circ\text{C}$  až  $85^\circ\text{C}$  a  $\pm 2^\circ\text{C}$  v maximálním rozsahu od  $-55^\circ\text{C}$  do  $125^\circ\text{C}$ , viz. obrázek 3.1. Vyrábí se v pouzdrech TO92 (třívývodové pouzdro pro klasickou montáž), SOIC a SOP (osmivývodové pouzdro pro SMD montáž). U pouzdra TO92 je první vývod napájecí ( $V_{DD}$ ), druhý vývod je datový vstup/výstup (DQ) a třetí je zem (GND).



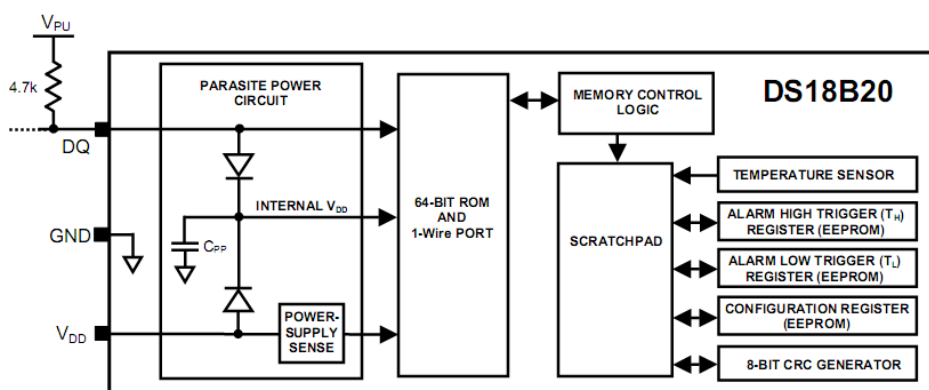
Obrázek 3.1.: Přesnost měření čidla DS18B20

Napájecí napětí se může pohybovat v rozsahu 3–5 V, proudový odběr je maximálně 1,5 mA. Komunikace s čidlem probíhá po sběrnici 1-Wire (bude popsána dále) a je možné jej napájet přímo ze sběrnice uskladněním energie na vnitřním kondenzátoru (tzv. parazitní napájení), takže je čidlo připojeno pouze na 2 vodiče (DQ a GND, vývod  $V_{DD}$  je připojen na zem). Když je signál sběrnice na vysoké úrovni, kondenzátor se nabíjí a když je signál na nízké úrovni, odebírá

čidlo energii z tohoto kondenzátoru. Výhodou tohoto zapojení je možnost absence lokálního napájení ve vzdálených aplikacích, sběrnice ovšem nemůže po dobu vykonávání konverze teploty vykonávat jinou činnost.<sup>1</sup> Parazitní napájení také není doporučeno pro měření teplot nad 100 °C z důvodu možného výskytu unikajících svodových proudů. Každé čidlo má vlastní unikátní 64bitové sériové číslo, díky kterému může být na sběrnici připojen teoreticky neomezený počet zařízení a sběrnici proto může řídit pouze jeden mikroprocesor s využitím jedné vstupně-výstupní brány. Čidlo též umožňuje nastavení spodního a horního teplotního limitu, při jehož překročení je použita funkce alarmu [9].

### 3.2. Blokové schéma

Blokové schéma čidla na obrázku 3.2 znázorňuje hlavní obvodové části čidla. Skládá se z těchto částí: obvod pro parazitní napájení, 64bitová ROM paměť (zde je uloženo unikátní identifikační číslo), snímač teploty, ovládací obvod, CRC generátor, EEPROM paměť (obsahující registry pro nastavení teplotních alarmů a konfigurační registr) a hlavní část, tzv. scratchpad paměť.



Obrázek 3.2.: Blokové schéma čidla DS18B20

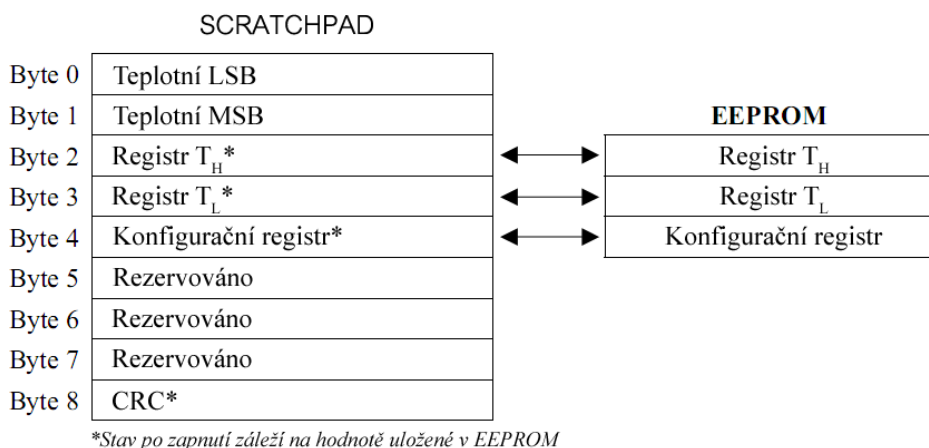
### 3.3. Vnitřní paměť

Vnitřní paměť se skládá z devítibytové scratchpad paměti typu SRAM a trvalé, energeticky nezávislé tříbytové EEPROM paměti. V EEPROM je uloženo nastavení teplotního alarmu pro vysoké a nízké teploty (dva byty – registry  $T_H$  a  $T_L$ ) a konfigurační registr (jeden byte). V tomto registru je uložena informace o nastavení rozlišení teploty při digitální konverzi.

<sup>1</sup> Po dobu konverze je sběrnice neaktivní (v log. 1) a zaručuje tak stabilní napětí pro vykonávání proudově náročné konverze.



Blokové schéma scratchpadu je na obrázku 3.3. Nultý a první byte obsahují LSB a MSB změřené teploty. Druhý a třetí jsou přechodné kopie  $T_H$  a  $T_L$  z EEPROM, které jsou obnoveny s každým resetem napájení. Čtvrtý byte je kopie konfiguračního registru, která je také obnovována s každým resetem napájení. Byty 5 až 7 jsou rezervovány pro vnitřní výpočty senzoru a nemohou být přepsány. Osmý byte obsahuje CRC kód, který je počítán ze všech osmi předchozích bytů scratchpadu.



Obrázek 3.3.: Scratchpad paměť

### 3.4. Měření teploty

Hlavní funkcí obvodu DS18B20 je přímý převod teploty do digitální podoby. Uživatel přitom může nastavit rozlišení měření na 9, 10, 11 nebo 12 bitů odpovídajícím krokům 0,5°C, 0,25°C, 0,125°C, a 0,0625°C. Výchozí rozlišení po inicializaci je 12 bitů. Nastavení rozlišení také rozhoduje o tom, jak dlouho bude konverze teploty trvat. Rozlišení a doby konverze jsou znázorněny v tabulce 3.1.

Rozlišení teploměru	Rozlišení teploměru	Čas převodu teploty
9 bitů	0,5 °C	93,75 ms
10 bitů	0,25 °C	187,5 ms
11 bitů	0,125 °C	375 ms
12 bitů	0,0625 °C	750 ms

Tabulka 3.1.: Závislost doby konverze teploty na nastaveném rozlišení

Po přijetí příkazu „Convert T“ [44h] <sup>2</sup> je proveden odměr teploty a následuje A/D konverze. Data odpovídající teplotě ve stupních Celsia jsou po převodu uložena jako dva byty do scratch-

<sup>2</sup>Viz. sekce 3.5.3 Funkční příkazy.

pad paměti a čidlo se vrátí do klidového stavu. Teplota je uložena jako šestnáctibitové binární číslo zakódované pomocí dvojkového doplňku – viz. tabulka 3.2. Znak „S“ je znaménkový bit: S=0 pro kladná čísla a S=1 pro negativní čísla. Pokud je obvod konfigurován na nižší rozlišení, nepoužité bity obsahují nuly.

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	$2^3$	$2^2$	$2^1$	$2^0$	$2^{-1}$	$2^{-2}$	$2^{-3}$	$2^{-4}$
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	$2^6$	$2^5$	$2^4$

Tabulka 3.2.: Teplotní registr

Po provedení teplotní konverze se naměřené hodnoty teploty porovnají s registrem alarmu a teprve pak se uloží do teplotních registrů. Pokud je výsledek měření teploty vyšší než  $T_H$  nebo nižší než  $T_L$ , nastaví se příznak alarmu uvnitř obvodu. Tento příznak je aktualizován s každým měřením teploty. Pokud na kterémkoliv čidlu teplota překračuje nastavené hranice, alarmující čidlo může být pomocí příkazu „Alarm Search“ [ECh] ihned identifikovatelné, aniž bychom museli zvlášť přistupovat k ostatním čidlům, nevykazujícím žádný příznak alarmu.

### 3.5. Komunikace s čidlem DS18B20

Obvod DS18B20 používá přesně definovaný komunikační protokol 1-Wire sběrnice. Komunikace na sběrnici je asynchronní a striktně dodržuje master/slave schéma. Zařízení typu slave může být připojeno více. Sběrnice je neaktivní v log. 1, což je zabezpečeno pull-up rezistorem<sup>3</sup>. Protokol definuje několik příkazů: reset pulz, prezenční pulz, zápis log. 0/1, čtení log. 0/1. Každý přenos dat pak probíhá v následujících krocích:

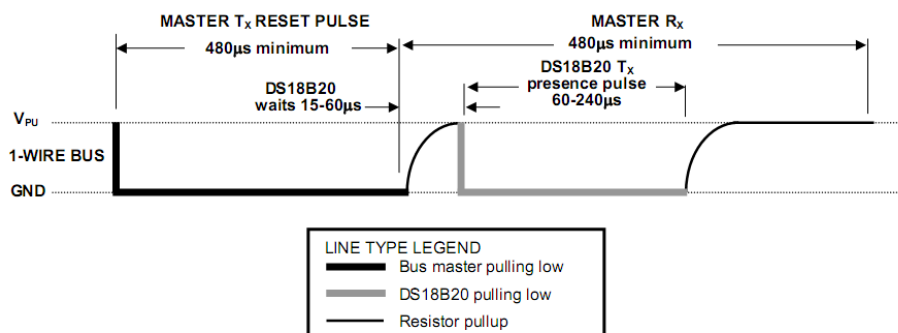
1. Inicializace
2. ROM příkazy
3. Funkční příkazy čidla DS18B20

Při každém přístupu k čidlu je nutné dodržet tento postup, vynechání kteréhokoliv kroku způsobí, že čidlo přestane odpovídat.

<sup>3</sup>Pull-up rezistor zajišťuje návrat datové linky na vysokou napěťovou úroveň. Používá se u logických obvodů, aby po odpojení externích zařízení (nebo jejich stavu vysoké impedance) byla na vstupech přesně definovaná úroveň napětí. Kdyby nebyl použit, napětí na vstupu by se vlivem rušení mohlo pohybovat a při překročení rozhodovací meze by mohlo docházet k přepínání log. 0 a log. 1, takže by na vstupu byla nedefinovaná hodnota.

### 3.5.1. Inicializační sekvence

Komunikace začíná inicializační sekvencí sestávající z reset pulzu vyslaného řídícím obvodem sběrnice (master), po kterém následuje prezenční pulz od obvodu DS18B20 (slave). Odesláním prezenčního pulzu čidlo signalizuje svoji přítomnost na sběrnici a je připraveno k další komunikaci. Sekvence začíná tím, že master stáhne sběrnici k log. 0 a vyčká 480–960  $\mu\text{s}$  ( $T_x$ ). Po této době je úroveň sběrnice vrácena na log. 1 pomocí pull-up rezistoru. Vyčkáme 1–60  $\mu\text{s}$  na přijetí prezenčního pulzu (log. 0) od slave zařízení, který trvá 60–240  $\mu\text{s}$  ( $R_x$ ). Časový průběh je znázorněn na obrázku 3.4. Tím je inicializace ukončena a master následně zapisuje a čte data ze zařízení v tzv. časových slotech. Jeden bit je přenesen za jeden časový slot.



Obrázek 3.4.: Časový průběh inicializační sekvence komunikace po 1-Wire sběrnici

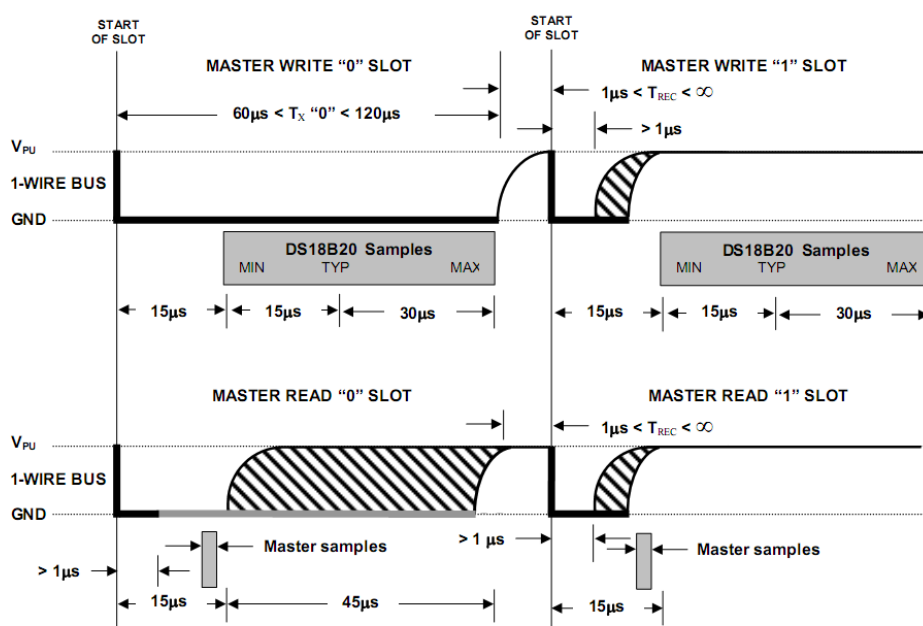
### Zapisovací slot

Zapisovací slot je zahájen sestupnou hranou datové linky a můžeme jej rozdělit na dva typy: zapsat log. 1 nebo log. 0. Všechny zapisovací sloty musí být minimálně 60  $\mu\text{s}$  dlouhé a musí mezi nimi být minimálně 1  $\mu\text{s}$  na obnovu linky. Po této regenerační době opět můžeme vyslat další zapisovací slot. Čidlo DS18B20 začne vzorkovat datovou linku po sestupné hraně v okně od 15 do 60  $\mu\text{s}$ . Pro zapsání log. 1, musí master zařízení zvednout úroveň ještě před tímto vzorkovacím oknem. Pro zapsání log. 0, musí podržet sběrnici na nízké úrovni po dobu zbývající do uplynutí zapisovacího slotu (nejméně 60  $\mu\text{s}$ ). Jakmile odezní vzorkování čidla, může sběrnici opět zvednout na vysokou úroveň a po regenerační době 1  $\mu\text{s}$  psát další slot. Časový průběh zapisovacího slotu je zobrazen na obrázku 3.5.

### Čtecí slot

Čtecí slot je opět zahájen sestupnou hranou na datové lince, která musí v nízké úrovni setrvat po dobu minimálně 1  $\mu\text{s}$ . Potom je díky pull-up rezistoru vytažena zpět na vysokou úroveň. Po inicializaci čtecího slotu může slave zařízení vysílat na sběrnici data. Pokud chce obvod

DS18B20 poslat log. 1, nechá datovou linku na vysoké úrovni a pokud chce poslat log. 0, musí stáhnout datovou linku na nízkou úroveň (po ukončení zápisu log. 0 je linka vrácena pull-up rezistorem do pohotovostního stavu na vysokou úroveň). Výstupní data z DS18B20 jsou platná 15  $\mu$ s po inicializační sestupné hraně, proto musí master uvolnit linku a zahájit její vzorkování nejpozději do 15  $\mu$ s. Celý časový úsek přitom trvá minimálně 60  $\mu$ s a za ním opět následuje úsek dlouhý minimálně 1  $\mu$ s na obnovu linky. Časový průběh čtecího slotu je zobrazen na obrázku na této straně.



Obrázek 3.5.: Časový průběh zápisu a čtení na sběrnici 1-Wire

### 3.5.2. ROM příkazy

Jakmile řídicí zařízení sběrnice zaregistruje přítomnost zařízení po přijetí prezenčního pulsu, může vyslat jeden z pěti ROM příkazů. Tyto příkazy umožňují master zařízení identifikovat zařízení na sběrnici pomocí jejich 64bitového ROM sériového čísla. Všechny ROM příkazy jsou osmibitové [v hranatých závorkách je uvedena hexadecimální reprezentace příkazu]:

- Search ROM (Vyhledej ROM) [F0h] – pokud už je komunikace inicializována, master nezná přesný počet zařízení na sběrnici, jejich typ a ani jejich 64bitové ROM kódy. Příkaz Search ROM zjistí 64bitové ROM kódy všech zařízení, která jsou připojena ke sběrnici. Pokud je připojeno jen jedno zařízení, je možné použít jednodušší příkaz Read ROM.
- Read ROM (Číst ROM) [33h] – příkaz je možné použít pouze tam, kde je na sběrnici přítomné pouze jedno zařízení. Umožňuje řídicímu zařízení přečíst ROM kód slave obvodu

bez použití příkazu Search ROM. Pokud je připojeno více zařízení na sběrnici, dojde ke kolizi, jelikož každé zařízení bude vysílat ve stejný okamžik.

- Match ROM (Porovnej ROM) [55h] – tento příkaz, následovaný 64bitovou ROM sekvencí, dovolí master zařízení adresovat konkrétní obvod na sběrnici. Jen čidlo, jehož ROM kód se přesně shoduje s 64bitovou ROM sekvencí, bude reagovat na následující příkazy. Všechna ostatní zařízení budou čekat na další inicializační sekvenci.
- Skip ROM (Přeskoč ROM) [CCh] – dovoluje master zařízení adresovat všechna zařízení bez znalosti jejich ROM kódu. Pokud je zapojeno více než jedno zařízení na sběrnici, následující příkazy se budou týkat všech zařízení. Master může například všem zařízením poslat příkaz pro konverzi teploty najednou. U příkazů pro čtení však bude docházet ke kolizím, protože se budou všechna zařízení snažit odpovědět současně.
- Alarm Search (Hledej alarm) [ECh] – příkaz je podobný příkazu Search ROM. Odpoví na něj ovšem jen čidla, která vykazují překročení teplotních limitů uložených v EEPROM. Master tak má okamžitý přístup k čidlům, která hlásí alarm.

### 3.5.3. Funkční příkazy

Po adresování ROM příkazem používá master pro komunikaci s čidlem funkční příkazy, které slouží ke čtení a zápisu do scratchpad paměti, ke spuštění teplotní konverze a zjištění způsobu napájení. Jde o tyto osmibitové příkazy [v hranatých závorkách je uvedena hexadecimální reprezentace příkazu]:

- Convert T (Převed' teplotu) [44h] – tento příkaz spustí teplotní konverzi. Čidlo DS18B20 vykoná konverzi, výsledek uloží do dvoubytového teplotního registru scratchpadu a poté zůstane nečinné s minimálním odběrem. Pokud je použito parazitní napájení, musí master do 10  $\mu$ s vrátit datovou linku na vysokou úroveň, protože čidlo potřebuje dostatek proudu (1,5 mA) k vykonání konverze a je potřeba zajistit, aby se nevybil interní kondenzátor  $C_{pp}$ . Jestliže je použito externí napájení čidla, řídicí systém může spustit čtecí slot a číst data přichází po tomto příkazu. Na sběrnici bude log. 0 po dobu, kdy čidlo provádí konverzi teploty a log. 1, jakmile je převod dokončen.
- Write Scratchpad (Zapiš scratchpad) [4Eh] – zapíše tříbytovou informaci do scratchpadu. První byte je zapsán do registru  $T_H$ , druhý do registru  $T_L$  (teplotní limity pro spuštění alarmu) a třetí do konfiguračního registru.
- Read Scratchpad (Čti scratchpad) [BEh] – tento příkaz čte postupně obsah scratchpadu. Čtení je zahájeno nultým bytem (LSB teplotního registru) a pokračuje až k osmému bytu (CRC). Master zařízení může kdykoliv ukončit čtení, pokud nechce číst celý scratchpad.

- Copy Scratchpad (Kopíruj scratchpad) [48h] – příkaz zkopíruje nastavení teplotních alarmů a konfiguračního registru (byty 2, 3 a 4) do EEPROM paměti. Pokud je použito parazitní napájení, musí master zařízení do 10  $\mu$ s vrátit datovou linku na vysokou úroveň na nejméně 10 ms, aby proběhlo zapsání do EEPROM správně (podobně jako při konverzi teploty).
- Recall E<sup>2</sup> (Čti z EEPROM) [B8h] – tento příkaz zkopíruje byty T<sub>H</sub>, T<sub>L</sub> (teplotní limity pro spuštění alarmu) a konfiguračního byte z EEPROM paměti do scratchpadu. Master může zahájit čtení a na sběrnici bude log. 0 po dobu kopírování a log. 1, jakmile je operace dokončena. Kopírování nastane automaticky po resetu napájení.

### 3.6. Shrnutí

Čidlo DS18B20 představuje moderní řešení pro snímání teploty s plně digitálním výstupem. Ve spojení s výbornými parametry měření, možností připojit několik čidel k jedné sběrnici a jednoduchostí zpracování dat se jedná o ideální řešení pro úlohu monitorování teploty s následným přenosem dat do PC. Dalším krokem byl návrh vhodného řídicího zařízení pro komunikaci s čidlem po sběrnici 1-Wire.

## ZAŘÍZENÍ PRO ŘÍZENÍ SBĚRNICE 1-WIRE

### 4.1. Sběrnice 1-Wire

Rozhraní 1-Wire bylo vyvinuto v devadesátých letech firmou Dallas Semiconductor (od roku 2001 součást koncernu Maxim Integrated Products). Jde o sběrnici využívající jeden vodič pro datový signál i napájení, druhý vodič je společná zem. Hlavní vlastnosti rozhraní 1-Wire jsou:

- jednoduché a originální řešení adresovatelnosti účastníků
- jednoduchý protokol
- jednoduchá struktura spojení
- malé množství potřebných komponentů
- snadná změna konfigurace sítě
- značný dosah sítě
- nízká cena technologie

Přenos dat v 1-Wire síti je asynchronní a poloduplexní. Veškeré informace obíhající v síti jsou účastníky přijímány jako příkazy nebo jako data. Příkazy jsou generovány řídicím členem sítě (master), který rovněž provádí různé varianty hledání a adresace známých zařízení, určuje aktivitu na síti, řídí přenos dat v síti apod. Standardní rychlost práce 1-Wire sítě 16 kbit/s byla vybrána jednak pro zajištění maximální spolehlivosti přenosu dat na velké vzdálenosti a také pro přizpůsobení rychlosti nejrozšířenějších typů mikropočítačů. Rychlost přenosu může být libovolně zmenšována díky zavedení vynuceného zpoždění mezi vysíláním jednotlivých bitů dat a prodlužováním časových slotů protokolu. Může být také zvýšena přechodem na zvláštní

zrychlený režim přenosu tzv. overdrive (125 kbit/s), který je možno provozovat na nevelké vzdálenosti s využitím kvalitní, jinými komponenty nepřetěžované spojovací linky. V síti 1-Wire se používají standardní TTL úrovně signálů a slave zařízení mohou být napájena z vnějšího zdroje s napětím od 2,8 do 6 V. Alternativou vnějšího zdroje je už zmíněné parazitní napájení[10]. Zásady komunikace po sběrnici byly nastíněny v předešlé kapitole.

Ke sběrnici 1-Wire je možné připojit velké množství slave komponent. Prvním typem jsou tzv. iButton zařízení v pouzdrech velmi podobných diskovým knoflíkovým bateriím. Mohou být připojeny ke sběrnici bez nutnosti pájení, výhodou je tedy jejich snadná přenositelnost spojená s vysokou odolností proti vnějším vlivům. Jednotlivé typy se používají především pro identifikaci pomocí jejich 64bitového sériového čísla, jako paměti, obvody reálného času, ale vyrábí se i senzory teploty a vlhkosti. Druhým typem s rozsáhlejšími možnostmi aplikace jsou obvody v plastových pouzdrech pro povrchovou montáž. Nejjednodušším obvodem je DS2401 sloužící k jednoznačné identifikaci pomocí 64bitového sériového čísla. Dalším zástupcem je čtyřkanálový A/D převodník DS2450, který umožňuje zpracovávat analogový signál například z různých běžně dostupných snímačů fyzikálních veličin. Na sběrnici 1-Wire lze také připojit množství různých pamětí (EPROM, EEPROM), vzdálených adresovatelných přepínačů, už zmíněné obvody pro měření teploty, či například obvod pro měření reálného času. Tato paleta zařízení předurčuje sběrnici 1-Wire k velmi jednoduchému řešení automatizace a řízení.

## 4.2. Řídící členy sběrnice

Pro řízení komunikace po 1-Wire sběrnici máme několik možností. Je nutné podotknout, že maximální dosah při zachování spolehlivosti vedení je závislý především na přesném časování datových průběhů na sběrnici, na přesnosti hran a rychlosti obnovení datové linky na vysokou úroveň (tzv. aktivní pull-up).

První z možností řízení je použití běžného mikroprocesoru a připojením sběrnice na jednu z jeho vstupně/výstupních bran. Nevýhodou tohoto řešení je nutnost generovat časové průběhy softwarově a nutnost obvodově řešit aktivní a přesné přidržování datové linky na vysoké úrovni. Výrobce dodává vývojovou sadu 1-Wire Public Domain Kit (open-source API psané v jazyku C), která programování mikroprocesorů značně zjednodušuje.

Další možností je použití mikroprocesoru se zabudovaným 1-Wire master řídicím obvodem (např. DS80C400). Časování sběrnice si tyto obvody řeší sami a k vývoji opět můžeme použít sadu 1-Wire Public Domain Kit. Opět je však nutno řešit přidržování datové linky na vysoké úrovni obvodově. Cena těchto mikroprocesorů je však poměrně vysoká (cca. 27 \$ za jeden obvod DS80C400).

Poslední možností je použití UART<sup>1</sup> obvodu libovolného mikroprocesoru, případně osob-

<sup>1</sup> Universal asynchronous receiver/transmitter. Obvod, který zajišťuje přeměnu sériové posloupnosti bitů do paralelního znaku a naopak.

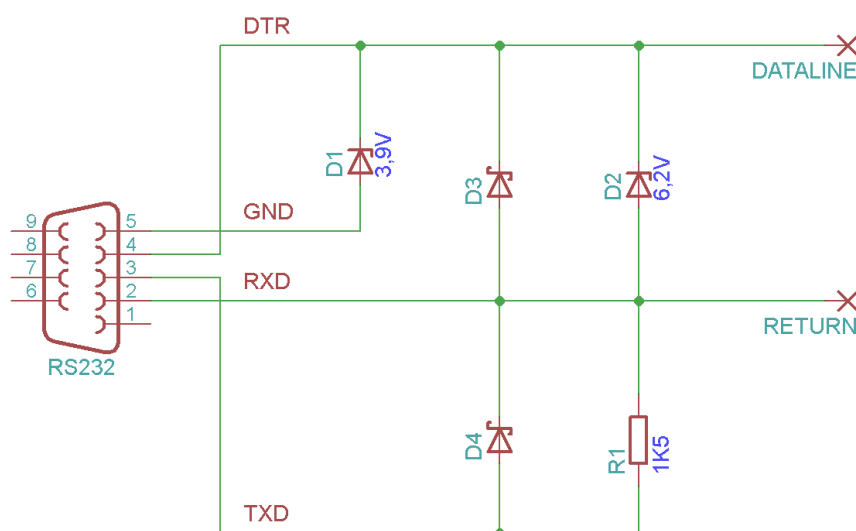


ního počítače. Výhodou je přenechání časování sběrnice obvodu UART a možnost programovat software pro komunikaci ve vyšším jazyku. Přidržování datové linky může být řešeno pull-up rezistorem, vhodnější je však použít obvod DS2480B. Jde o 1-Wire budící obvod s UART vstupním rozhraním, pracující v pěti voltové TTL logice, takže jej lze připojit přímo k mikroprocesoru. Zajišťuje aktivní, rychlé a přesné přidržování vysoké úrovně a umožňuje nastavování rychlosti komunikace po 1-Wire síti až do overdrive rychlosti (125 kbit/s)[11].

Výrobce rovněž nabízí hotové 1-Wire adaptéry typu DS9097 fungující na základě obvodu DS2480B, které umožňují připojení 1-Wire sběrnice přímo k PC pomocí sériového COM portu<sup>2</sup>. Uživatel pak může programovat programy přímo pro PC a komunikovat se sběrnicí přes sériový port. Lze použít jedno z API pro nejčastější programovací jazyky. Vzhledem k požadavku na zpracování teplotních dat v PC jsem zvolil právě toto řešení.

### 4.3. Pasivní adaptér DS9097

Nejjednodušší možností, jak připojit 1-Wire síť k PC je použitím tzv. pasivního adaptéru DS9097. Jedná se o zapojení vyžadující minimum součástek, připojitelné k sériovému COM portu. Zná-zorněno je na obrázku 4.1.



Obrázek 4.1.: Schéma pasivního adaptéru DS9097

Jde v podstatě o převodník úrovní mezi standardem RS-232 ( $\pm 12$  V) a pěti voltovou TTL logikou sběrnice 1-Wire. Napájení sběrnice je tedy řešeno přímo z COM portu (parazitní napájení).

<sup>2</sup>Výrobce má v nabídce také obvod DS2490, který představuje rozhraní mezi USB portem počítače a 1-Wire sběrnicí. Nedoporučuje však tento obvod používat v nových návrzích a je doporučeno místo něj použít obvod DS2480B.

Tvarování datového signálu je zcela v režii UART obvodu a převod úrovní je zajištěn Schottkyho a Zenerovými diodami. Datová rychlost UARTu je 115,2 kbit/s a přenáší se osmibitový znak, takže rychlost 1-Wire sběrnice je 14,4 kbit/s (tzv. regular speed). Přidržování datové linky ve vysoké úrovni je řešeno pull-up rezistorem 1,5 k $\Omega$ . Ze schématu také vyplývá, že signál RETURN neodpovídá skutečné zemi [12].

Toto jednoduché řešení jsem si vyzkoušel na nepájivém propojovacím poli a pomocí konzolové aplikace DigiTemp<sup>3</sup> získával teplotní informace z čidel. Pro krátkou vzdálenost je bez problému dostačující a na internetu lze nalézt mnoho řešení pro jednoduché měření teploty<sup>4</sup>. Z osobní zkušenosti mám vyzkoušeno čtení z jednoho čidla na 15 m dlouhém krouceném kabelu. Údajně by však mělo být možné připojit až 30 m vedení [13]. Pro delší vedení sběrnice je však toto zapojení nevhodné, protože neumožňuje aktivní pull-up ani samostatné napájení čidel. Adaptér využívající tohoto zapojení již výrobce nenabízí a doporučuje použít novější DS9097U.

#### 4.4. Aktivní adaptér DS9097U

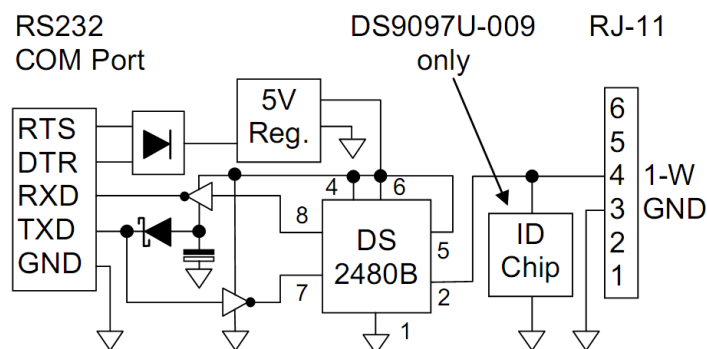
Jedná se opět o převodník úrovní mezi standardem RS-232, který však na rozdíl od předchozího pasivního adaptéru pracuje na základě obvodu DS2480B a zajišťuje aktivní tvarování hran datového signálu (aktivní pull-up). Pracuje v bytovém režimu, tzn. že každý příchozí byte z RS-232 rozhraní převádí na osm časových slotů podle definice protokolu 1-Wire. Takto je dosaženo vyšší rychlosti přenosu oproti staršímu adaptéru DS9097<sup>5</sup>. Výrobce doporučuje tento adaptér používat všude tam, kde je potřeba připojit síť 1-Wire k PC. Adaptér je připojitelný na standardní COM port počítače a 1-Wire síť se připojuje pomocí konektoru RJ11. Zjednodušené schéma je zobrazeno na obrázku 4.2 na následující straně.

Funkce obvodu je podobná zapojení adaptéru DS9097 s tím rozdílem, že je použit obvod DS2480B, který je napájen přes 5 V stabilizátor z datové linky COM portu. Vstupy obvodu DS2480B pracují v pětivoltové TTL logice, takže je nutné opět převést  $\pm 12$  V úroveň RS-232 standardu. Doporučená cena tohoto adaptéru je cca 21 \$ (u českého dodavatele cca 700 Kč). Ve spojení s cenou čidel by však tato investice do monitoringu teploty byla poněkud vysoká a proto jsem se rozhodl vyrobit alternativu tohoto adaptéru svépomocí.

<sup>3</sup>Open-source GNU program pro čtení informací z 1-Wire sběrnice. Je napsán v programovacím jazyku C a podporuje čtení dat z 1-Wire teploměrů, vlhkoměrů, čítačů, měřičů baterií, atd. Dostupný z WWW: <http://www.digitemp.com/index.shtml>.

<sup>4</sup>Za zmínku stojí například webové stránky <http://www.root.cz/clanky/nejjednodussi-online-teplomer-2/> nebo <http://trifid2.sweb.cz/teplomer/>.

<sup>5</sup>Maximální rychlost přenosu na 1-Wire sběrnici při použití DS9097 odpovídá rychlosti UARTu (UART – 115,2 kbit/s, 1-Wire – 14,4 kbit/s). Adaptér DS9097U je díky bytovému přenosu rychlejší už při rychlosti UARTu 19,2 kbit/s (každý byte příchozí z RS-232 je rozdělen na kratší časové sloty odpovídající standardu 1-Wire protokolu).



Obrázek 4.2.: Blokové schéma adaptéru DS9097U

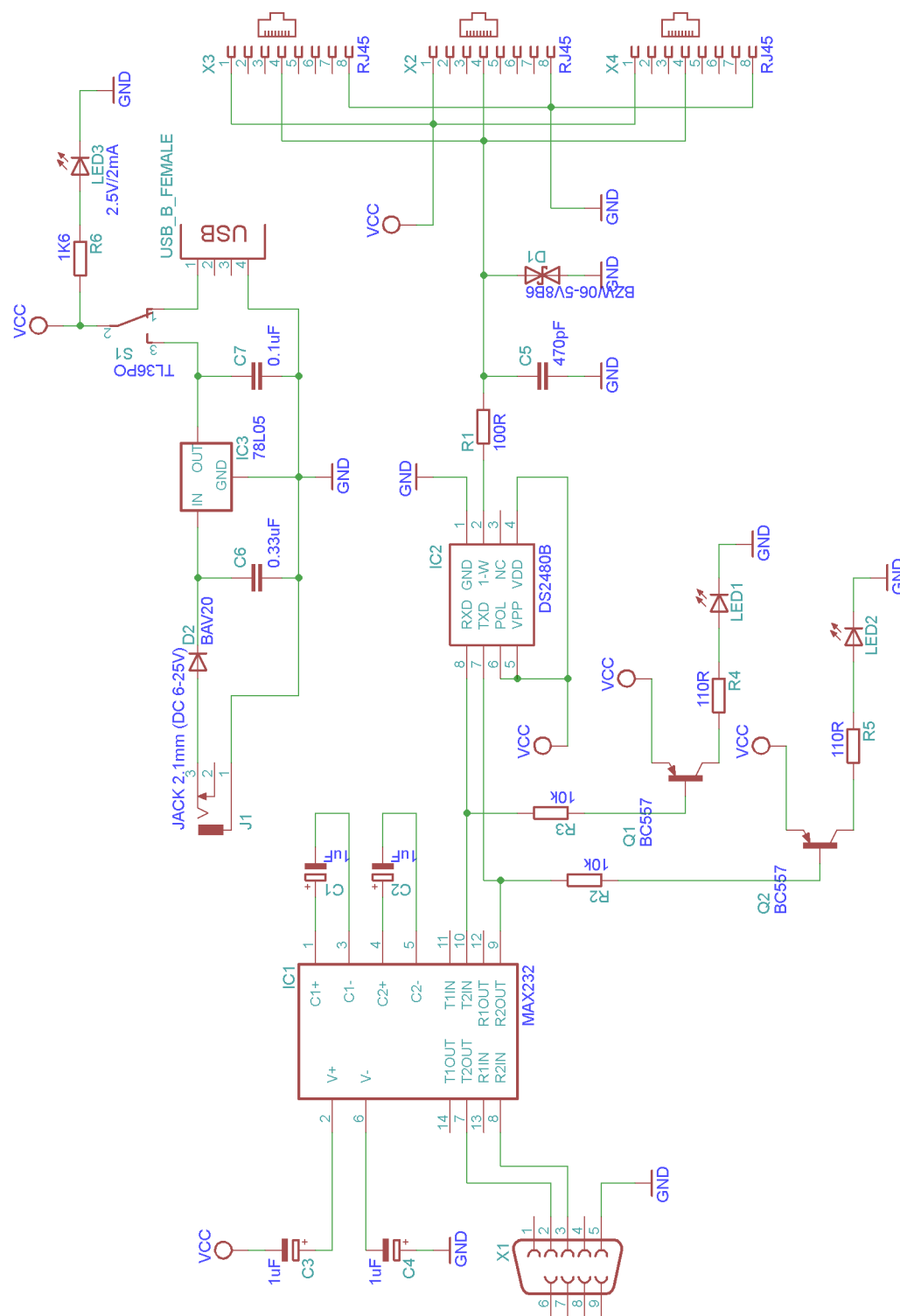
## 4.5. Stavba vlastního 1-Wire adaptéru

Jak jsem již uvedl, pasivní 1-Wire adaptér DS9097 by nebyl spolehlivým řešením pro ovládání rozlehlější sítě a aktivní adaptér DS9097U nevyhovoval svojí cenou. Nebylo mi dopředu známo, jak rozlehlou síť bude v serverovně nutné navrhnout (zadán byl pouze orientační počet čidel) a proto jsem se snažil vybrat co nejspolehlivější řešení. Spolehlivost sítě bylo nejjednodušší zajistit použitím budiče sběrnice DS2480B a rovněž jsem se rozhodl nepoužívat parazitní napájení a každé čidlo DS18B20 napájet z 5 V zdroje. Tyto důvody vedly ke stavbě vlastního 1-Wire adaptéru velmi podobnému adaptéru DS9097U.

### 4.5.1. Popis a schéma adaptéru

Na internetu se mi podařilo nalézt vcelku elegantní a jednoduché řešení připojení budiče 1-Wire sběrnice DS2480B ke COM portu [14][15]. Schéma zapojení je na obrázku 4.3 na následující straně.

Převod napěťových úrovní rozhraní RS-232 na 5 V TTL logiku (a naopak), ve které pracují vstupy (a výstupy) obvodu DS2480B je řešen pomocí obvodu MAX232CPE. Jedná se o dvojité budič/přijímač protokolu RS-232 od firmy Maxim Integrated Products. Kondenzátory kolem obvodu jsou využívány nábojovými pumpami obvodu, které při 5 V napájení zajišťují převod z TTL logických úrovní na  $\pm 12$  V. Není tedy nutné používat Zenerovy a Schottkyho diody, tak jako u pasivního adaptéru. Vstup a výstup obvodu jsou připojeny k obvodu DS2480B a k indikaci přenosu po těchto datových linkách slouží LED diody (žlutá – vysílání a zelená – příjem dat), které jsou spínány přes tranzistory BC557. K 1-Wire vstupně/výstupní bráně obvodu DS2480B je podle doporučení výrobce připojen  $100\ \Omega$  rezistor, 470 pF kondenzátor (RC filtr) a oboustranný transil (k ochraně před napěťovými špičkami) [16]. Sběrnice 1-Wire se k adaptéru připojuje pomocí RJ45 konektorů k jednomu ze tří portů, které jsou paralelně spojeny. Vodič 1

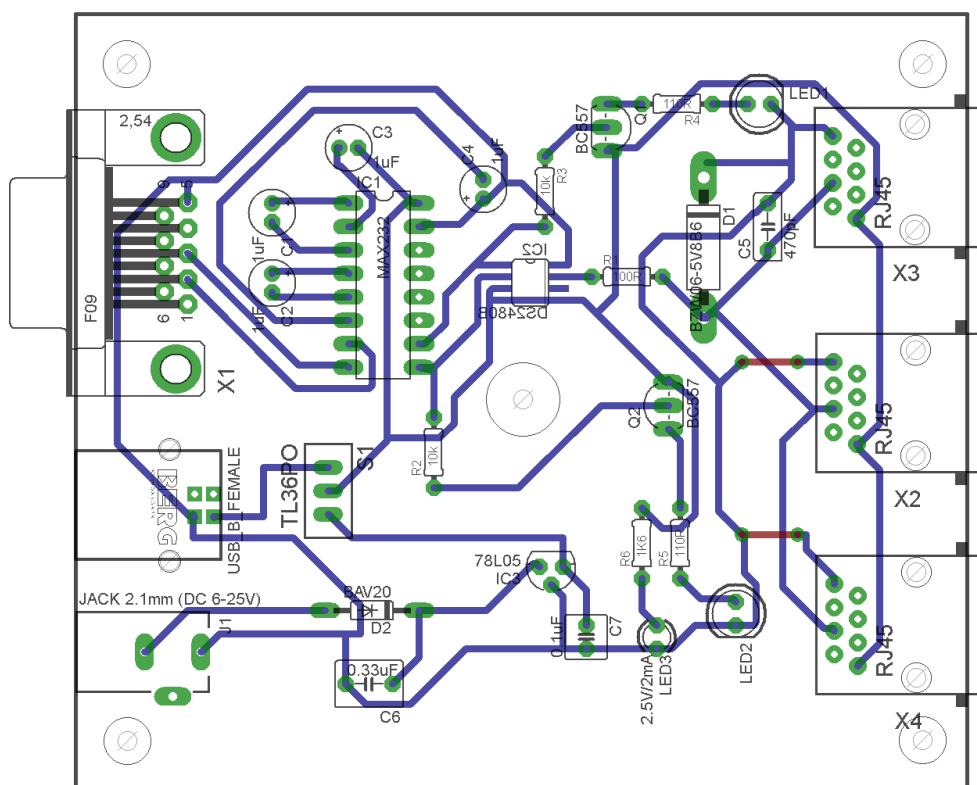


Obrázek 4.3.: Schéma zapojení pro výrobu 1-Wire adaptéru

slouží k napájení sběrnice, vodič 4 vede data a vodič 8 je zem.

Originální adaptér DS9097U je napájen přímo z COM portu počítače. Ten by ale v mém případě nebyl schopen dostatečně proudově zásobit jak obvody MAX232 a DS2480B, tak několik čidel DS18B20 na sběrnici. Maximální proudový odběr ze sériového portu je totiž většinou do 10 mA (závisí na použitých obvodech) [17]. Rozhodl jsem se proto použít samostatné 5 V napájení, adaptér je tedy možné napájet z USB portu pomocí zásuvky USB-B<sup>6</sup> nebo síťovým napájecím adaptérem. Lze použít libovolný síťový adaptér se stejnosměrným napětím 6–25 V s kladným pólem na středu 2,1 mm JACK konektoru. Napětí ze síťového adaptéru je vedeno přes diodovou ochranu proti přepólování a 5 V stabilizátor. Mezi zdroji napájení lze přepínat přepínačem a přítomnost napětí je indikována červenou LED diodou.

#### 4.5.2. Konstrukce adaptéru



Obrázek 4.4.: Deska plošného spoje pro výrobu 1-Wire adaptéru

Adaptér jsem nejdříve sestavil na nepájivém propojovacím poli, kde jsem vyzkoušel jeho funkčnost. Potom jsem navrhl desku plošného spoje (DPS) v programu Eagle, návrh jsem při-

<sup>6</sup>USB port má stabilizované napětí 5 V při maximálním proudovém odběru 500 mA.

způsobil plánovaným rozměrům plastové krabičky KP 02 od firmy A&A<sup>7</sup>. Deska plošného spoje a rozmístění součástek je zobrazeno na obrázku 4.4 na předchozí straně. DPS jsem si vyrobil sám osvitom fotocitlivého cuprexitu UV výbojkou a následným vyleptáním v chloridu železitém. V krabičce jsou vyřezány otvory pro konektory a DPS je připevněna šrouby. Jednotlivé díly krabičky jsou spojeny jedním vrutem uprostřed krytu. V příloze A jsou přiloženy fotografie hotového adaptéru (A.1, A.2, A.3).

---

<sup>7</sup>Jedná se o univerzální krabičky z tvrzeného polystyrenu. Jsou běžně dostupné např. v obchodech GES-ELECTRONICS a.s. a GM Electronic s.r.o.. Webové stránky výrobce krabiček: <http://www.krabicky.cz>.

## SOFTWARE PRO KOMUNIKACI PO SBĚRNICI 1-WIRE

### 5.1. Seznam 1-Wire API

Dalším krokem návrhu monitorovacího systému byla tvorba vhodného softwaru pro PC. Pro připojení 1-Wire sítě k počítači jsem použil výše popsany vyrobený adaptér. Rozhodl jsem se nepoužít už dostupné aplikace a pokusit se vytvořit vlastní program pro komunikaci po 1-Wire sběrnici, který by lépe odpovídal požadavkům zadání. Firma Maxim Integrated Products poskytuje velmi dobrou podporu pro tvorbu programů pro komunikaci s 1-Wire zařízeními. K dispozici je pět API<sup>1</sup>, které pracují na rozličných platformách, používají různé programovací jazyky a mají různé možnosti. Každé API je nabízeno formou balíčku, jehož součástí je rozsáhlá dokumentace a množství vzorových aplikací a příkladů, jak jednotlivá API používat. Jde o tyto API [18]:

- 1-Wire Public Domain API – kompletně open-source API, napsané v jazyku C. Je přenositelné na různé počítačové operační systémy a mikroprocesory. V operačním systému Microsoft Windows podporuje všechny dostupné 1-Wire master zařízení pomocí nativních ovladačů pro sériový port, není tedy nutno instalovat žádné další ovladače. Ostatní operační systémy mohou použít multiplatformní knihovny.
- 1-Wire API for Java – open-source API, psané v jazyku Java. Bylo navrženo jako robustní, objektově orientované API pro psaní 1-Wire aplikací. Některé platformy používají nativní ovladače pro sériový port a na dalších je možné použít RXTX knihovny<sup>2</sup>.
- 1-Wire API for .NET – jde v podstatě o 1-Wire API for Java, pouze zkompileované v jazyku J#. Aplikace pro Microsoft Windows je pak možné programovat v jazycích podporujících

<sup>1</sup>Application programming interface – rozhraní pro programování aplikací. Sbírá procedury, funkce a třídy nějaké knihovny, které může programátor využívat. API určuje, jakým způsobem se funkce knihovny volají ze zdrojového kódu programu.

<sup>2</sup>Multiplatformní API pro sériový COM port. Dostupné na <http://www.rxtx.org/>.

.NET Framework (C#, J#, VB.NET, atd.), za použití knihovny `OneWire.NET.dll`. Pro běh aplikací a komunikaci s 1-Wire zařízeními je nutné mít nainstalovány speciální 1-Wire ovladače, Microsoft .NET 2.0 Framework a Visual J# .NET 2.0 Redistributable.

- 1-Wire API for .NET Compact – zjednodušené API, napsáno v jazyku C#, je určeno primárně pro Windows CE/Mobile a Pocket PC.
- TMEX API – soubor jazykově nezávislých DLL knihoven. Podporuje všechny 1-Wire master zařízení na platformě Microsoft Windows. Je voláno ostatními API (1-Wire API for Java, .NET) pro přístup k 1-Wire adaptérům. Jde v podstatě o ovladače pro komunikaci s 1-Wire adaptéry, které jsou instalovány jako součást balíku 1-Wire Drivers.

## 5.2. Tvorba softwaru

Po domluvě s vedoucím práce byl určen jako cílový operační systém Microsoft Windows XP. Rozhodl jsem se proto použít 1-Wire API for .NET v programovacím jazyku C#. Hlavním důvodem byla možnost práce ve vyšším, objektově orientovaném jazyku a poměrně jednoduchá práce s 1-Wire zařízeními. Aplikaci jsem vyvíjel ve vývojovém prostředí Microsoft Visual Studio 2008. Výstupem měla být spustitelná aplikace pro operační systém Microsoft Windows s grafickým uživatelským rozhraním. Aplikace psané pomocí 1-Wire API for .NET dostupného ze stránek výrobce, potřebují ke svému běhu tyto balíky:

- ovladače pro komunikaci s 1-Wire adaptérem a zařízeními <sup>3</sup>
- Microsoft .NET 2.0 Framework<sup>4</sup>
- Visual J# .NET 2.0 Redistributable<sup>5</sup>

Na webu dokumentace k 1-Wire API for .NET je uveden odkaz na alternativní open-source projekt třetí strany OW.NET, který se zaměřuje na migraci 1-Wire .NET API nativně do jazyka C#, bez nutnosti použití jazyka J#<sup>6</sup>. Tímto odpadá nutnost instalovat balík Visual J# .NET 2.0 Redistributable na počítač, kde poběží 1-Wire aplikace. Projekt nemá vlastní dokumentaci, ale vzhledem k tomu, že se od původního 1-Wire API for .NET liší jen velmi málo, lze bez problému použít oficiální dokumentaci. Pro vývoj aplikace jsem použil právě tento projekt.

Základem algoritmu pro práci s 1-Wire sběrnici je třída zastupující fyzický adaptér `DSPortAdapter`. Tato třída umožňuje jak vysílání jednotlivých bitů a bytů po 1-Wire síti (např.

<sup>3</sup>Dostupné z webu výrobce: [http://www.maxim-ic.com/products/ibutton/software/tmex/download\\_drivers.cfm](http://www.maxim-ic.com/products/ibutton/software/tmex/download_drivers.cfm).

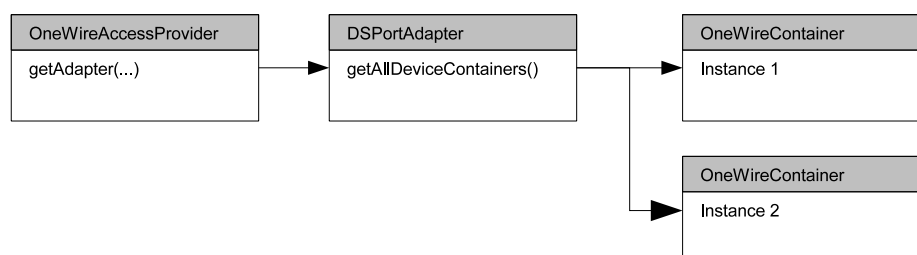
<sup>4</sup>Je součástí MS Windows SP3, Vista a Windows 7, jinak je volně stažitelný z webu společnosti Microsoft.

<sup>5</sup>Volně stažitelné z webu společnosti Microsoft.

<sup>6</sup>Domovské stránky projektu jsou dostupné na WWW: <http://sourceforge.net/projects/owdotnet/>.



reset pulz – viz. sekce 3.5 Komunikace s čidlem DS18B20), tak řízení rychlosti sítě a vyhledávání 1-Wire zařízení. Instanci této třídy získáme zavoláním metody `OneWireAccessProvider.getAdapter(String jmenoAdapteru, String jmenoPortu)`. Třída `OneWireAccessProvider` získává informace o 1-Wire adaptérech pomocí TMEX ovladačů. Relaci komunikace s 1-Wire sběrnici zahájíme metodou `DSPortAdapter.beginExclusive(true)`. Veškerá slave zařízení pro 1-Wire jsou reprezentována nadřazenou metodou `OneWireContainer`. Seznam zařízení připojených ke sběrnici získáme zavoláním metody `DSPortAdapter.getAllDeviceContainers()`. Vývojový diagram tohoto postupu je zobrazen na obrázku 5.1.



Obrázek 5.1.: Vývojový diagram přístupu k 1-Wire zařízením

Senzor DS18B20 a jemu podobná zařízení jsou reprezentována třídou `OneWireContainer28`, která obsahuje metody pro komunikaci s teplotním čidlem. Pro představu o jednoduchosti použití 1-Wire .NET API uvedu některé z těchto metod:

- `double[] getTemperatureResolutions()` – získá všechna možná nastavitelná rozlišení senzoru a uloží je do pole.
- `double getMaxTemperature()` – získá maximální měřitelnou teplotu na základě parametrů čidla (125 °C v případě čidla DS18B20).
- `double getMinTemperature()` – získá minimální měřitelnou teplotu na základě parametrů čidla (–55 °C v případě čidla DS18B20).
- `byte[] readDevice()` – přečte stav devíti bytů scratchpad paměti čidla a uloží je do referenčního stavového pole bytů `state`. S tímto polem potom pracují `get` a `set` metody.
- `void writeDevice(byte[] state)` – zapíše devět bytů `state` pole do scratchpad paměti čidla a finalizuje tím změny provedené `set` metodami.
- `void doTemperatureConvert(byte[] state)` – spustí konverzi teploty uložené ve `state` poli.

- `double getTemperature(byte[] state)` – pokud je dokončena konverze, touto metodou přečteme teplotu ve °C.
- `double getTemperatureResolution(byte[] state)` – přečte nastavené rozlišení čidla.
- `void setTemperatureAlarm(int alarmType, double alarmValue, byte[] state)` – nastaví horní (`alarmType = 1`) nebo dolní (`alarmType = 0`) hranici alarmu čidla.
- `void setTemperatureResolution(double resolution, byte[] state)` – nastaví rozlišení čidla.
- `void copyScratchpad()` – zkopíruje registry  $T_H$ ,  $T_L$  (teplotní limity pro spuštění alarmu) a konfigurační registr ze scratchpadu do EEPROM paměti.
- `byte[] recallE2()` – obnoví registry  $T_H$ ,  $T_L$  a konfigurační registr z EEPROM paměti do scratchpadu.
- `boolean isExternalPowerSupplied()` – zjistí, jestli čidlo používá parazitní napájení (`return true`), či nikoliv (`return false`).
- `float convertToFahrenheit(float celsiusTemperature)` – převede již zkonvertovanou teplotu ze stupňů Celsia na stupně Fahrenheita.

Nastavení rozlišení čidla poté probíhá například následovně:

```
byte[] state = owc28.ReadDevice();
double[] resolutions = owc28.GetTemperatureResolutions();
owc28.SetTemperatureResolution(resolutions[2], state);
owc28.WriteDevice(state);
```

Čtení teploty z čidla probíhá následujícím algoritmem:

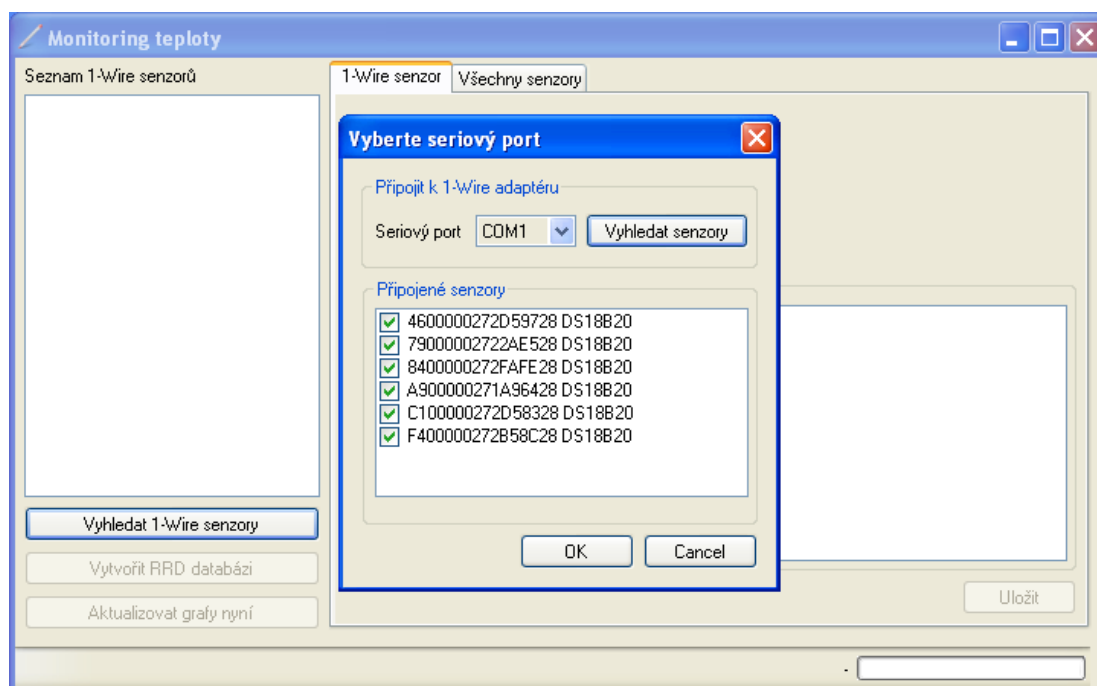
```
byte[] state = owc28.ReadDevice();
owc28.DoTemperatureConvert(state);
double temperature = owc28.GetTemperature(state);
```

### 5.3. Popis vlastní aplikace

Pro komunikaci s čidly jsem naprogramoval aplikaci s grafickým rozhraním. Tato aplikace čte pravidelně teplotu ze všech čidel a ukládá teplotní data do RRD databáze. Zároveň ve stanovených intervalech generuje grafy zobrazující průběhy teplot jednotlivých čidel. Databáze a grafy jsou vytvářeny automatickým spouštěním programu RRDtool s příslušnými parametry.

Ke spuštění aplikace je zapotřebí mít nainstalovány oficiální 1-Wire ovladače z balíku 1-Wire Drivers a rozhraní Microsoft .NET Framework. Kořenový adresář aplikace obsahuje tyto soubory: spustitelný soubor `MonitoringTeploty.exe`, DLL knihovnu `OW.Net.dll` (OW.NET 1-Wire API), spustitelný soubor `rrdtool.exe` (vytažený z balíku RRDtool) a konfigurační soubory `config.xml` a `devices.xml`.

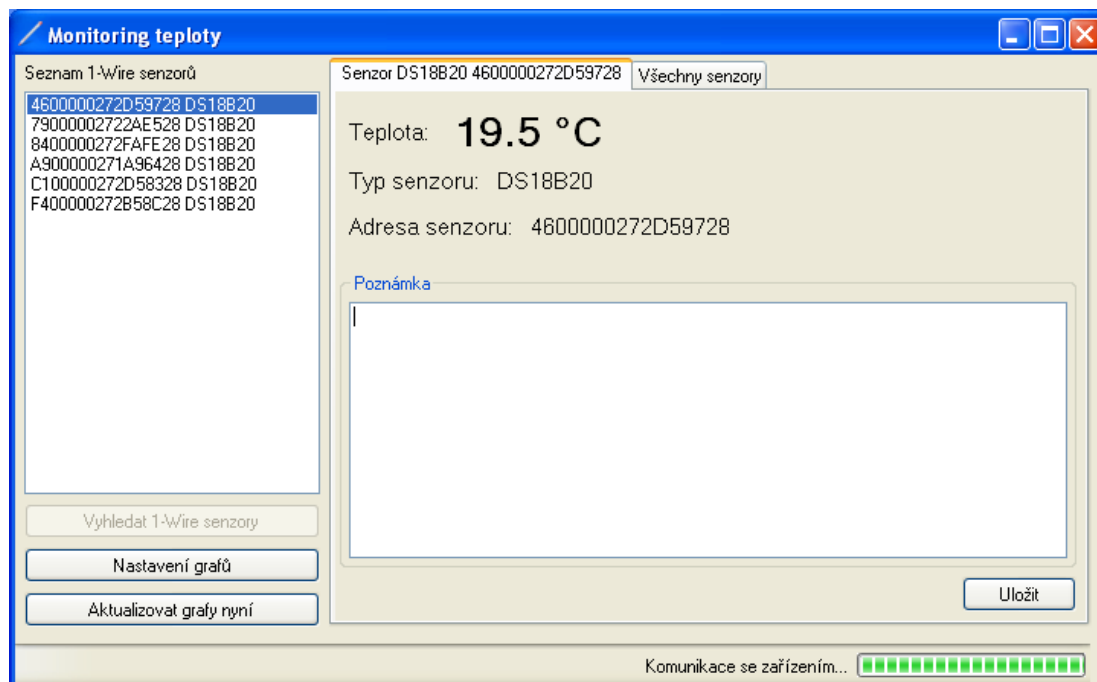
Po spuštění aplikace je nutné kliknout na tlačítko „Vyhledat 1-Wire senzory“, které vyvolá dialogové okno (viz. obrázek 5.2) s výběrem COM portu, ke kterému je připojen 1-Wire adaptér. Po výběru portu se kliknutím na tlačítko „Vyhledat senzory“ provede pomocí metod třídy `DSPortAdapter` vyhledání všem teplotních čidel připojených k adaptéru.



Obrázek 5.2.: Dialogové okno pro vyhledání čidel připojených k 1-Wire adaptéru

Čidla jsou identifikována pomocí jejich 64bitových sériových čísel a jejich seznam je po zavření dialogového okna zobrazen v levém panelu hlavního okna aplikace. Zároveň je spuštěn časovač, který čte pravidelně teplotu ze všech čidel (interval čtení se rovná počtu čidel vynásobenému dobou nutnou pro konverzi teploty jednoho čidla – počet čidel  $\times$  800 ms). Vybráním jednoho čidla v seznamu se v pravé části okna na první záložce zobrazí informace o teplotě přečtené z tohoto čidla. Hlavní okno s první záložkou je zobrazeno na obrázku 5.3 na následující straně. Na první záložce je také formulář pro uložení poznámky k čidlu (poznámka může sloužit například k uložení informace o umístění čidla v serverovně). Poznámky se ukládají do souboru `devices.xml`, kde jsou podle XML tagů přiřazeny ke konkrétnímu čidlu pomocí jeho sériového čísla. Přepnutím na druhou záložku se zobrazí přehled všech připojených čidel společně

s jejich poznámkami. Je zde také zobrazen celkový čas čtení ze všech čidel. Druhá záložka je zobrazena na obrázku 5.4 na následující straně.



Obrázek 5.3.: První záložka hlavního okna

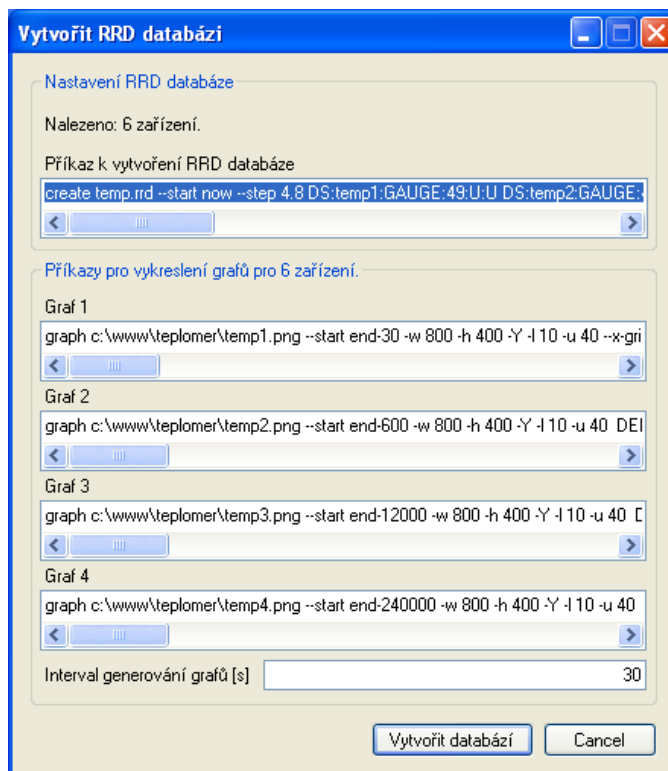
Kliknutím na tlačítko „Vytvořit RRD databázi“ je vyvoláno dialogové okno pro tvorbu RRD databáze a nastavení grafů. Tento dialog lze vidět na obrázku 5.5 na straně 31. Každé textové pole obsahuje parametry, se kterými bude spuštěn program RRDtool. V prvním textovém poli je příkaz pro vytvoření RRD databáze do souboru `temp.rrd`. Vzor pro tento příkaz je při otevření dialogu čten ze souboru `config.xml` (tento vzor je možné ručně změnit editací souboru `config.xml`) a je dynamicky doplněn pro vytvoření databáze pro tolik signálů, kolik bylo nalezeno čidel. Vzorkovací interval databáze je rovněž nastaven podle počtu nalezených čidel. Pokud již databáze existuje, je nabídnuta možnost přepsání nebo ponechání existující databáze.

Další čtyři textová pole dialogu slouží k nastavení parametrů pro tvorbu výsledných grafů a jejich vzor je opět čten ze souboru `config.xml`. Jsou vytvořeny čtyři grafy ve formátu PNG, každý s různým časovým rozpětím. Počet signálů a jejich barva je opět generován automaticky na základě počtu nalezených čidel, přičemž se do každého grafu kreslí všechny signály. V posledním textovém poli je interval, ve kterém budou pravidelně vytvořeny grafy.

Po potvrzení dialogu se vytvoří databáze v souboru `temp.rrd`. Při každém pravidelném čtení teplot se spustí program `rrdtool.exe` s parametrem `update` doplněn o hodnoty teplot z jednotlivých čidel a aktualizuje databázi<sup>7</sup>. Zároveň je také spuštěn další časovač, generující grafy

<sup>7</sup>RRDtool je konzolová aplikace, proto je při jejím spuštění potlačeno zobrazování konzolového okna.





Obrázek 5.5.: Dialogové okno pro tvorbu RRD databáze a nastavení grafů

## REALIZACE MONITOROVACÍHO SYSTÉMU

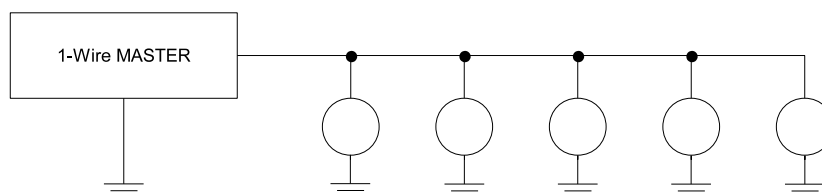
Zbývalo zapojit a zprovoznit celý monitorovací systém v serverovně. Zadání bylo specifikováno na snímání teploty ze šesti čidel. Jak jsem již uvedl, senzory DS18B20 jsou k 1-Wire adaptéru připojeny pomocí konektoru RJ45. Důvodem byla snadná dostupnost UTP kabelu kategorie 5, na který lze konektory RJ45 velmi rychle připevnit krimpovacími kleštěmi. V serverovně lze také využít existujících propojených racků pro vedení rozvodů 1-Wire sítě. Z kabelu jsou použity pouze tři vodiče ( $V_{DD}$ , DQ a GND), ke kterým je na konci připájen senzor DS18B20.

### 6.1. Instalace softwaru a zapojení 1-Wire sítě

Aplikace pro komunikaci s 1-Wire zařízeními byla společně s ovladači pro 1-Wire adaptér nainstalována na virtuální stroj běžící na jednom z počítačů v serverovně. Adaptér byl připojen na sériový port počítače a je napájen z 12 V síťového napájecího adaptéru. Rozvod 1-Wire sítě byl vyřešen pomocí existujícího spojení mezi jednotlivými racky serverovny.

První zkouškou bylo připojení jednoho čidla DS18B20 ke vzdálenému racku, délka vedení cca 10m. Aplikace pro monitoring obvod bez problémů našla. Větvení sítě mělo být řešeno zakoupenými RJ45 rozdvojkami (jeden konec male konektor RJ45, druhý konec dva male konektory RJ45, připojeny paralelně). Po připojení tří čidel pomocí rozdvojek k jednomu vzdálenému racku (propojen s adaptérem jedním kabelem) se ovšem vyskytl problém. Aplikaci se opakovaně nedařilo nalézt všechna čidla, bylo nalezeno pouze jedno či dvě. Z oficiální dokumentace vyplývá, že sběrnice 1-Wire je zřejmě velmi náchylná na topologii sítě. Je doporučována lineární topologie, kdy jsou jednotlivá zařízení připojena k hlavnímu průběžnému kabelu sběrnice pomocí krátkých odboček nejlépe stejné délky, tak jak je znázorněno na obrázku 6.1 na následující straně. Výslovně není doporučována hvězdicová topologie, kde může na uzlech docházet ke kolizím a odrazům signálu způsobeným různou impedancí připojených odboček. Tento důvod podle mě způsobil i výše zmíněný problém, protože pokud se RJ45 rozdvojky připojí těsně

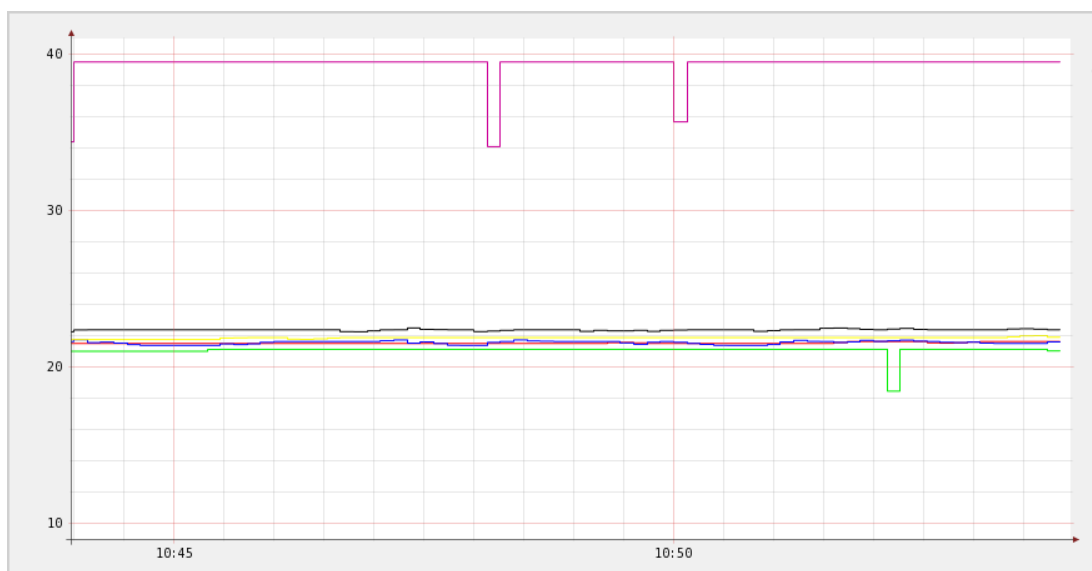
za adaptér a ke každému čidlu je přes racky veden zvláštní kabel, lze bez problému připojit libovolný počet zařízení. Takto je řešeno i finální zapojení 1-Wire sběrnice pro všech šest čidel, která jsou rozmístěna na různých místech serverovny. Budící obvod sběrnice DS2480B by měl při použití jednoduchého RC-filtru zabezpečit spolehlivou obsluhu až 200 m sítě [19].



Obrázek 6.1.: Doporučená topologie 1-Wire sítě

## 6.2. Reálné měření teploty

Po spuštění monitorovacího systému lze z výsledných grafů vidět, že občas dojde k výpadku měření u konkrétního čidla. Tento jev se mi stával už při testování na nepájivém propojovacím poli s jedním čidlem připojeným 15 m kabelem. Při konverzi teploty čidlo občas vyhodí chybu kontroly CRC, což je v mém algoritmu interpretováno jako neznámá hodnota. Program RRDtool ovšem tuto neznámou hodnotu průměruje s další platnou a proto lze na grafu 6.2 vidět u fialového průběhu malý pokles teploty.

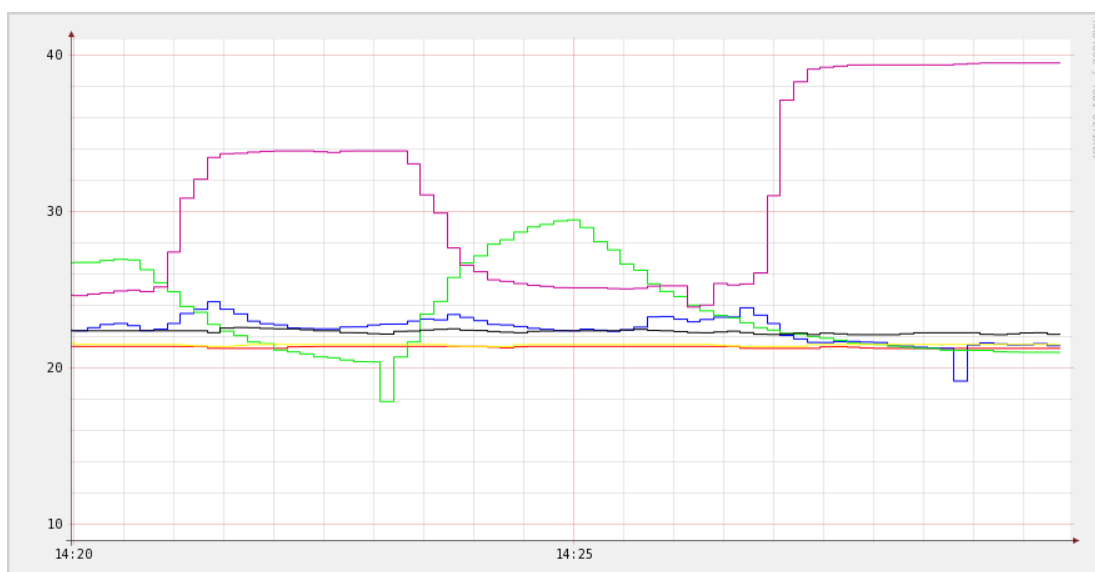


Obrázek 6.2.: Průběhy teplot v serverovně za posledních 10 min



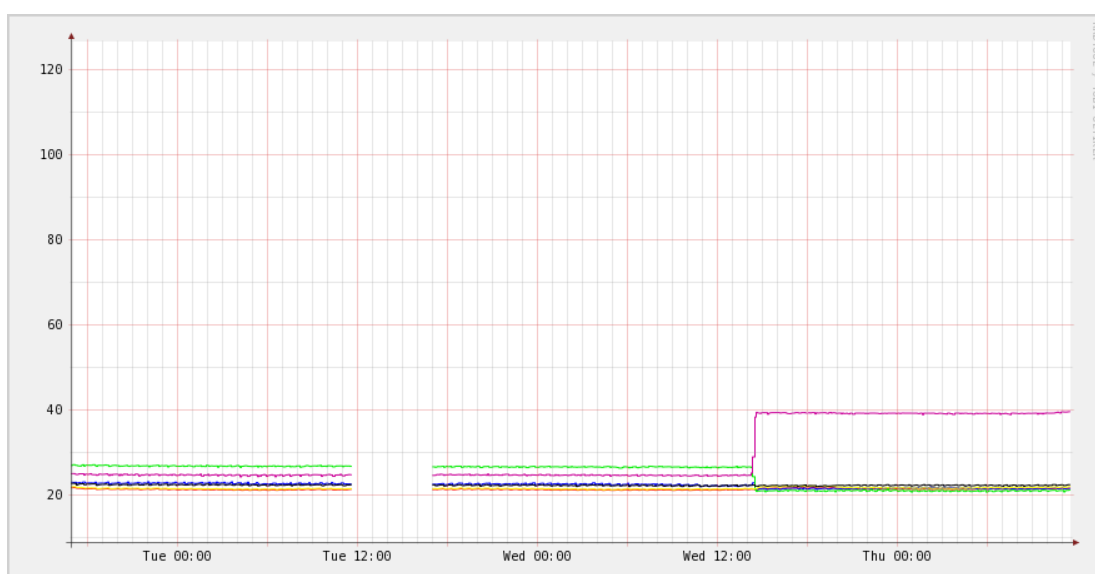
Tento jev se do určité míry podařilo odstranit filtrovacími kondenzátory 200 nF připojenými na vývody  $V_{DD}$  a GND. V delším vedení zřejmě vzniká nežádoucí indukčnost a kondenzátory pomáhají tento jev potlačit.

Systém sbírá teplotu za šesti čidel, každému čidlu odpovídá jedna barva grafu (na některých grafech jsou průběhy špatně rozlišitelné, protože čidla měřila velmi podobnou teplotu). Časový rozsah grafů odpovídá nastavení aplikace, testovací provoz běžel s rozsahy posledních 30 vteřin, 10 minut, 200 minut a 66 hodin. Při výpadku (odpojení) jednoho z čidel se jeho průběh přestane generovat. Grafy jsou ukládány na pevný disk počítače, kde běží monitorovací aplikace a přes webový server jsou přístupné na internetu, takže je možné je vzdáleně odkudkoliv sledovat.



Obrázek 6.3.: Graf reakcí na skokové změny teploty

Na grafu 6.3 lze vidět reakce čidel na skokové změny teploty. Zelený průběh reprezentuje simulovanou pomalou skokovou změnu – čidlo bylo přesunuto na bok racku v serverovně (skok z cca 20 °C na cca 30 °C), teploměr se za této situace chová jako systém prvního řádu s časovou konstantou cca 40 s. Po ohřátí bylo čidlo vráceno na původní místo, kde chladlo – opět lze namodelovat systémem prvního řádu s časovou konstantou cca 75 s. Fialový průběh odpovídá čidlu, které bylo přemístěno za větrák počítačové skříně, takže změna teploty byla rychlejší. Reakce na skok (z cca 25 °C na cca 40 °C) v tomto případě odpovídá opět systému prvního řádu s časovou konstantou cca 20 s. Na obrázku 6.4 na následující straně lze vidět průběhy teplot za posledních 66 hodin (prázdné místo je způsobeno výpadkem systému, na kterém běžela aplikace pro monitoring).



Obrázek 6.4.: Průběhy teplot v serverovně za posledních 66 hodin

## ZÁVĚR

Zpracováním návrhu teplotního monitorovacího systému jsem si vyzkoušel komplexní řešení problému sledování vzdálených veličin. Zásadním krokem byla volba senzoru, který by nejlépe vyhovoval úkolu měření teploty v serverovně. Bylo vybráno digitální čidlo DS18B20, které je připojitelné ke sběrnici 1-Wire. Naučil jsem se s touto sběrnicí pracovat a navrhovat software pro její ovládání. Zároveň jsem ověřil použitelnost čidel a sběrnice v praxi. Využití sběrnice 1-Wire by bylo vhodné například v inteligentním domě pro regulaci vytápění. Snadná rozšiřitelnost o další zařízení při zachování poměrné jednoduchosti 1-Wire sítě ji k tomu přímo předurčuje.

K datu psaní této práce běží monitorovací systém v serverovně spolehlivě již několik týdnů. Snadný přístup ke grafům je umožněn přístupem na webový server běžící na počítači, kde běží aplikace pro komunikaci se senzory. 1-Wire síť je vedena pomocí existujících síťových spojů mezi racky v serverovně a je tudíž možná její jednoduchá modifikace. Systém čidel je tak možné kdykoliv rozšířit připojením dalšího čidla ke sběrnici. Serverovna budovy FEL na Karlově náměstí tak má nyní přesný teplotní monitorovací systém.

## LITERATURA

- [1] National Instruments. *Working with Thermistors and RTDs* [online]. 2009-11-13 [cit. 2010-05-20]. Dostupné z WWW: <http://zone.ni.com/devzone/cda/tut/p/id/2948>.
- [2] Enercorp Instruments ltd. *Comparision Of Thermistors, Thermocouples, and RTD's* [online]. Poslední úpravy 2007-01-04 [cit. 2010-05-20]. Dostupné z WWW: [http://www.process-controls.com/enercorp/temp\\_comparision\\_thermistors\\_thermocouples\\_rtds.htm](http://www.process-controls.com/enercorp/temp_comparision_thermistors_thermocouples_rtds.htm).
- [3] MATULÍK, Radomír, Ing. *Měření a regulace teploty* [online]. Poslední úpravy 2009-12-21 [cit. 2010-05-20]. Dostupné z WWW: <http://www.aterm.cz/Teperm.htm>.
- [4] National Semiconductor. *LM34/LM35 Precision Monolithic Temperature Sensors* [online]. Application Note 460. October 1986 [cit. 2010-05-22]. Dostupné z WWW: <http://www.national.com/an/AN/AN-460.pdf>.
- [5] Philips Semiconductors. *KTY81-1 series Silicon temperature sensors* [online]. 2000 Aug 25 [cit. 2010-05-22]. Dostupné z WWW: [http://www.gme.cz/\\_dokumentace/dokumenty/530/530-015/dsh.530-015.1.pdf](http://www.gme.cz/_dokumentace/dokumenty/530/530-015/dsh.530-015.1.pdf).
- [6] National Semiconductor. *LM135/LM235/LM335, LM135A/LM235A/LM335A Precision Temperature Sensors* [online]. November 2000 [cit. 2010-05-22]. Dostupné z WWW: [http://www.gme.cz/\\_dokumentace/dokumenty/530/530-003/dsh.530-003.1.pdf](http://www.gme.cz/_dokumentace/dokumenty/530/530-003/dsh.530-003.1.pdf).
- [7] National Semiconductor. *LM35 Precision Centigrade Temperature Sensors* [online]. November 2000 [cit. 2010-05-22]. Dostupné z WWW: [http://www.gme.cz/\\_dokumentace/dokumenty/313/313-091/dsh.313-091.1.pdf](http://www.gme.cz/_dokumentace/dokumenty/313/313-091/dsh.313-091.1.pdf).
- [8] Smartec. *SMT16030 Digital Temperature Sensor* [online]. Version 7/2005 [cit. 2010-05-22]. Dostupné z WWW: <http://www.smartec.nl/pdf/DSSMT16030.PDF>.

- [9] Maxim Integrated Products. *DS18B20 Programmable Resolution 1-Wire Digital Thermometer* [online]. 2008 [cit. 2010-05-23]. Dostupné z WWW: <http://datasheets.maxim-ic.com/en/ds/DS18B20.pdf>.
- [10] Redakce HW serveru. *MicroLan - A jde to i s jedním vodičem!* [online]. 7. Březen 2005 [cit. 2010-05-23]. Dostupné z WWW: <http://hw.cz/Rozhrani/ART1240-MicroLan---A-jde-to-i-s-jednim-vodicem.html>.
- [11] Bernhard Linke. *Choosing the Right 1-Wire® Master for Embedded Applications* [online]. Application Note 4206. Mar 27, 2008 [cit. 2010-05-24]. Dostupné z WWW: <http://pdfserv.maxim-ic.com/en/an/AN4206.pdf>.
- [12] Dan Awtrey. *1-Wire Net Design Guide* [online]. Dallas Semiconductor Corp Tech Brief 1. 2001 [cit. 2010-05-25]. Dostupné z WWW: <ftp://ftp.elin.ru/pdf/1-Wire/Dguide2.pdf>.
- [13] Uživatel „yuppee“. *Nejjednodušší teploměr - sebrané poznatky*. [online]. 200? [cit. 2010-05-25]. Dostupné z WWW: <http://sklenik.net/info/teplomer/navod/>.
- [14] Uživatel „quattro“. *Domací automatizace III* [online]. 25 Prosinec 2008 04:42, Aktualizováno (Pondělí, 23 Únor 2009 10:41) [cit. 2010-05-25]. Dostupné z WWW: [http://quattro.haje.cz/index.php?option=com\\_content&view=article&id=52&catid=16&Itemid=26](http://quattro.haje.cz/index.php?option=com_content&view=article&id=52&catid=16&Itemid=26).
- [15] Natasza Biecek. *Sprzętowy konwerter RS232/1-Wire - aplikacja układu DS2480B* [online]. 10-02-2007, Last Updated : 2007-02-11 15:12:28 [cit. 2010-05-25]. Dostupné z WWW: <http://www.elportal.pl/index.php?module=ContentExpress&func=display&ceid=118>.
- [16] Maxim Integrated Products. *DS2480B Serial to 1-Wire Line Driver* [online]. 2010 [cit. 2010-05-25]. Dostupné z WWW: <http://datasheets.maxim-ic.com/en/ds/DS2480B.pdf>.
- [17] Tomi Engdahl. *Get power out of PC RS-232 port* [online]. 2000-04-13 [cit. 2010-05-25]. Dostupné z WWW: <http://www.tkk.fi/Misc/Electronics/circuits/rspower.html>.
- [18] Maxim Integrated Products. *1-Wire® Software Resource Guide Device Description* [online]. Application Note 155. Jul 08, 2008 [cit. 2010-05-26]. Dostupné z WWW: <http://pdfserv.maxim-ic.com/en/an/AN155.pdf>.

- [19] Maxim Integrated Products. *Guidelines for Reliable Long Line I-Wire® Networks* [online]. Application Note 148. Sep 22, 2008 [cit. 2010-05-26]. Dostupné z WWW: <http://pdfserv.maxim-ic.com/en/an/AN148.pdf>.

# **Přílohy**

---

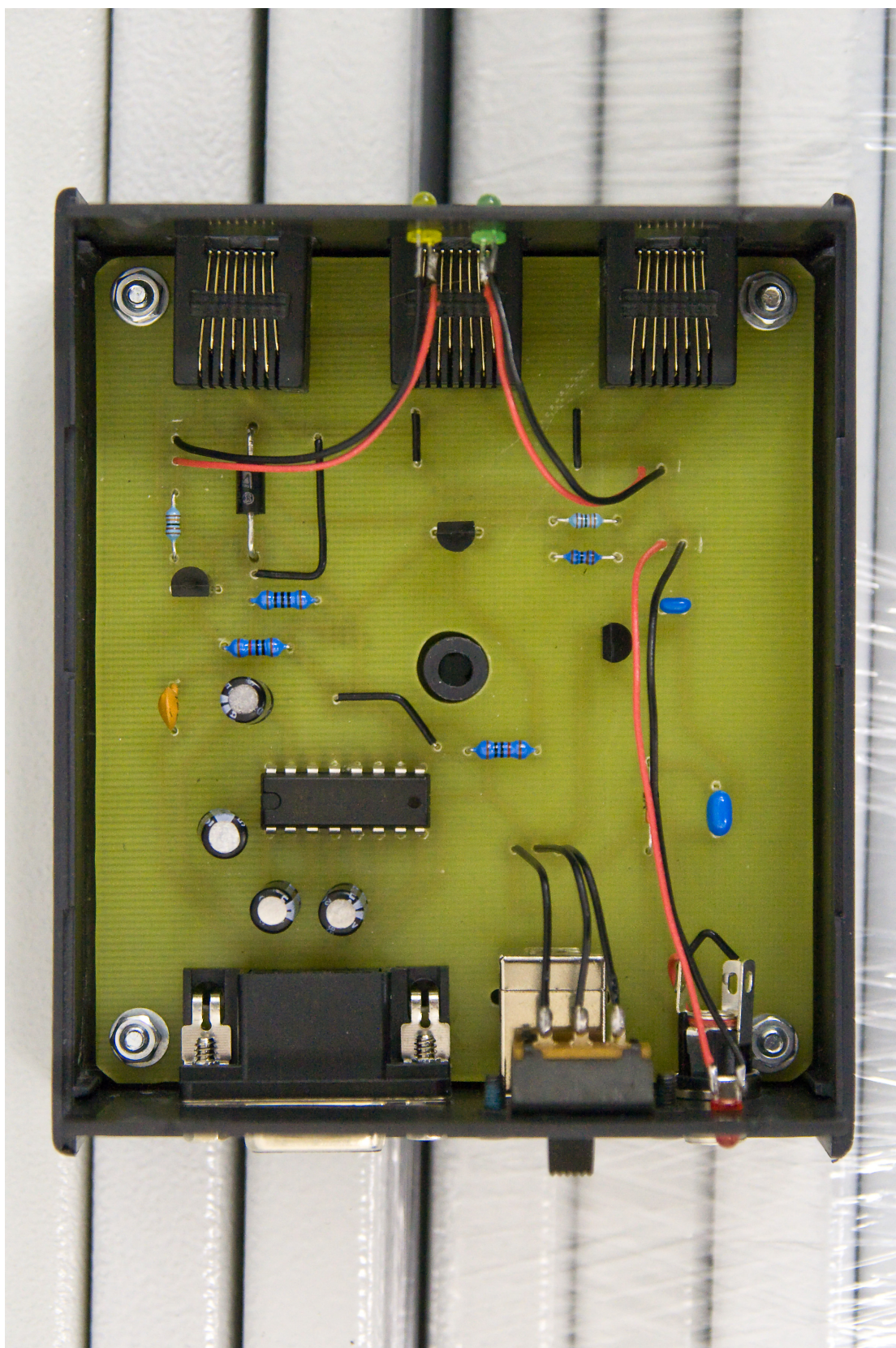
PŘILOHA

**A**

---

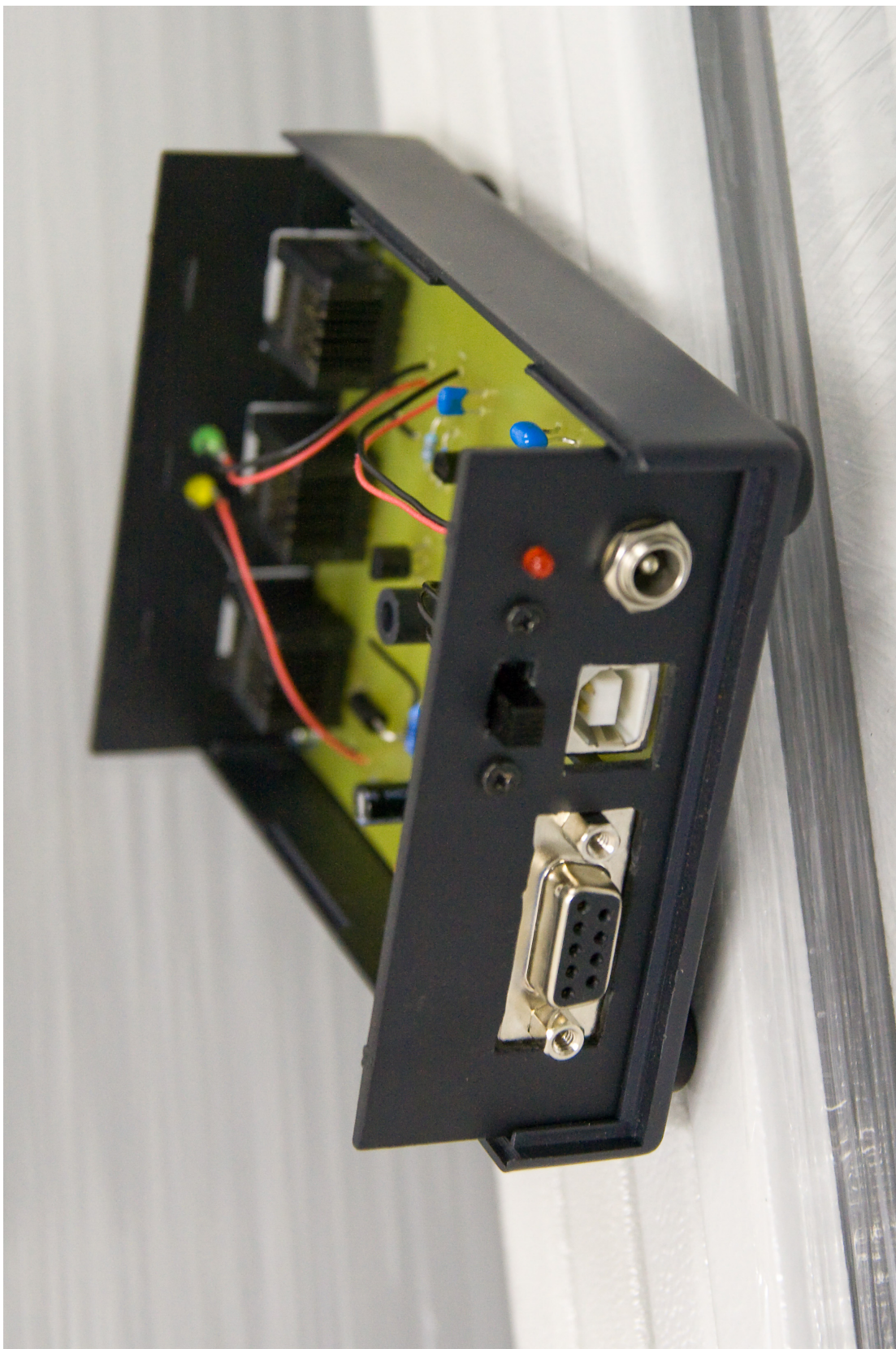
## FOTOGRAFIE 1-WIRE ADAPTÉRU





Obrázek A.1.: 1-Wire adaptér – pohled seshora





Obrázek A.2.: 1-Wire adaptér – pohled ze strany pro připojení COM konektoru





Obrázek A.3.: 1-Wire adaptér – pohled ze strany zásuvek RJ45 pro připojení sběrnice