

UNIVERZITA PARDUBICE
Fakulta elektrotechniky a informatiky

Sledovací systém dostupnosti routeru
se vzdáleným HW resetem

Tomáš Hotovec

Bakalářská práce
2010

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Tomáš HOTOVEC**
Osobní číslo: **I07891**
Studijní program: **B2646 Informační technologie**
Studijní obor: **Informační technologie**
Název tématu: **Sledovací systém dostupnosti routeru se vzdáleným HW
resetem**
Zadávající katedra: **Katedra informačních technologií**

Z á s a d y p r o v y p r a c o v á n í :

Teoretická část:

- * Vytvořte přehled existujících řešení watchdog.
- * Diskutujte jejich možnosti zejména se zaměřením na možnost studeného restartu počítače.

Praktická část:

- * Navrhněte jednoduchý a velmi levný watchdog, který lze snadno připojit k běžnému (i staršímu) PC (např. pomocí konektoru klávesnice nebo paralelního portu).
- * Ve spolupráci s FEI watchdog realizujte a vyzkoušejte ho v praxi.

Rozsah grafických prací:

Rozsah pracovní zprávy:

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

Seznam odborné literatury:

***Kolektiv autorů: LINUX dokumentační projekt, Computer Press 1998 (3. aktualizované vydání), ISBN 80-7226-761-2**

***Shah, S.: Administrace systému Linux ? podrobný průvodce začínajícího administrátora, Grada Praha 2002, ISBN 80-7169-586-6**

***dokumentace PC, interface klávesnice, paralelní port**

***programování v Unixu: UNIX programming Tutorials**
<http://users.actcom.co.il/~choo/lupg/tutorials/index.html>

Vedoucí bakalářské práce:

Mgr. Tomáš Hudec

Katedra informačních technologií

Datum zadání bakalářské práce: **15. ledna 2010**

Termín odevzdání bakalářské práce: **14. května 2010**



prof. Ing. Simeon Karamazov, Dr.

děkan



L.S.



Ing. Lukáš Čegan, Ph.D.

vedoucí katedry

V Pardubicích dne 31. března 2010

Prohlášení autora

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 25. 7. 2010

Tomáš Hotovec

Poděkování

Tímto bych rád poděkoval vedoucímu bakalářské práce Mgr. Tomáši Hudcovi za vedení práce a rady, které mi v průběhu řešení mé práce poskytl a také Ing. Martinu Hájkovi za pomoc při návrhu desek plošných spojů.

Anotace

Cílem této práce je navržení a realizace monitorovacího zařízení (watchdogu), které bude sledovat činnost systému případně nějaké služby v systému a v případě, že nedostane do určité doby signál, restartuje router.

Klíčová slova

watchdog, router, Linux

Title

Monitoring the availability of the router with a hardware reset.

Annotation

The bachelor thesis deals with the monitoring device (watchdog), which will monitor the availability of the router. If it does not receive at predetermined intervals a signal restarts the router.

Keywords

watchdog, router, Linux

Obsah

Seznam zkratk	8
Seznam obrázků	9
Seznam tabulek	10
Úvod	11
1 Přehled možných řešení watchdog	12
1.1 Továrně vyráběná zařízení.....	12
1.2 Zařízení vlastní konstrukce.....	13
1.2.1 Watchdog bez mikroprocesoru.....	13
1.2.2 Watchdog s mikrokontrolérem.....	14
2 Komunikace s počítačem	15
2.1 Paralelní port (LPT).....	15
2.2 Port klávesnice.....	17
2.3 Sériový port.....	17
2.4 Sběrnice USB.....	18
2.5 Komunikace s počítačem – Shrnutí.....	18
3 Sériový port RS-232	19
3.1 Základní parametry RS-232.....	19
3.1.1 Délka vedení.....	19
3.1.2 Napěťové úrovně.....	20
3.1.3 Parametry datového přenosu.....	20
3.1.4 Přenosové rychlosti.....	21
3.1.5 Řízení toku dat.....	22
4 Sběrnice USB	23
4.1 Přenosové rychlosti a verze USB.....	23
4.2 USB kabely a konektory.....	24
4.3 Enumerace – rozpoznávání zařízení.....	26
5 Vlastní návrh a konstrukce watchdogu	27
5.1 Řídící mikrokontrolér (MCU) ATTiny 2313.....	27
5.2 Komunikace sériovým portem, obvod MAX232.....	27
5.3 Komunikace pomocí rozhraní USB, obvod FT-232RL.....	28
5.3.1 Zapojení a popis jednotlivých vývodů.....	29

5.3.2 Signály CBUS.....	31
5.4 Popis zapojení.....	32
5.4.1 RS-232 verze.....	32
5.4.2 Verze pro sběrnici USB.....	34
5.5 Mechanická konstrukce.....	35
5.6 Rozpis součástí.....	38
5.6.1 Verze RS-232.....	38
5.6.2 Verze USB.....	39
5.7 Popis funkce a stavová LED.....	40
5.8 Firmware.....	40
5.8.1 Ukázky zdrojového kódu.....	40
5.8.2 Naprogramování firmware do mikrokontroléru.....	43
5.9 Konfigurace watchdogu, popis komunikace.....	43
6 Obslužný software pro OS Linux.....	47
6.1 Přehled knihovních funkcí thwdtlib.h.....	47
6.2 Popis programu WDTD.....	49
Závěr.....	52
Literatura.....	53

Seznam zkratk

ASCII	American Standard Code for Information Interchange
MCU	Micro Controller Unit
OS	Operační Systém
UART	Universal Asynchronous Receiver and Transmitter
USART	Universal Synchronous Asynchronous Receiver and Transmitter
USB	Universal Serial Bus
RLE	Run Length Encoding

Seznam obrázků

Obrázek 1: Jednoduchý watchdog s obvodem 555 (Převzato z [3]).....	13
Obrázek 2: Zapojení konektoru paralelního portu.....	15
Obrázek 3: Konektor DSub 25 na straně PC.....	15
Obrázek 4: Konektor RS-232 DB-9 [7].....	19
Obrázek 5: Synchronizace RS-232	20
Obrázek 6: Přenos 8 datových a 1 paritního bitu [5].....	21
Obrázek 7: Zařízení Full Speed / Low Speed [11].....	24
Obrázek 8: Struktura kabelu USB.....	25
Obrázek 9: Konektory USB.....	25
Obrázek 10: Zapojení pinů portu USB na základní desce [12].....	25
Obrázek 11: Integrovaný obvod MAX-231 [18].....	28
Obrázek 12: Blokové schéma obvodu FT-232R [19].....	29
Obrázek 13: Rozmístění vývodů obvodu FT232RL (pouzdro SSOP 28) [19].....	30
Obrázek 14: Watchdog – zapojení konektoru SV1.....	32
Obrázek 15: Watchdog – schéma verze RS-232.....	33
Obrázek 16: Watchdog – schéma verze USB.....	34
Obrázek 17: Watchdog RS-232 bottom.....	36
Obrázek 18: Watchdog RS-232 top.....	36
Obrázek 19: Watchdog RS-232 – rozmístění součástek bottom.....	36
Obrázek 20: Watchdog RS-232 – rozmístění součástek top.....	36
Obrázek 21: Watchdog USB bottom.....	37
Obrázek 22: Watchdog USB top.....	37
Obrázek 23: Watchdog USB rozmístění součástek bottom.....	37
Obrázek 24: Watchdog USB rozmístění součástek top.....	37
Obrázek 25: Konfigurační bity MCU.....	43

Seznam tabulek

Tabulka 1: Konektor klávesnice PS/2 [16].....	17
Tabulka 2: Konektor klávesnice DIN5 [15].....	17
Tabulka 3: Napěťové úrovně RS-232.....	20
Tabulka 4: Přenosové rychlosti sériového portu.....	21
Tabulka 5: Standardy USB a přenosové rychlosti.....	23
Tabulka 6: Vývody FT232RL.....	30
Tabulka 7: Sběrnice CBUS [19].....	31
Tabulka 8: Výchozí konfigurace pinů CBUS.....	31
Tabulka 9: Stavová LED – popis signalizace.....	40

Úvod

Tato bakalářská práce se zabývá možnostmi realizace a vlastním návrhem jednoduchého zařízení watchdog pro monitorování routeru. První kapitola pojednává o tom, jakým směrem se lze ubírat při vlastním návrhu watchdog zařízení, a také se zmíním o továrních řešeních, která jsou v dnešní době k dispozici.

V druhé kapitole představím některá z komunikačních rozhraní počítače, která by bylo možné při realizaci zařízení použít. Například paralelní port, sériový port, USB, port klávesnice a podobně. Uvedu jejich základní parametry a v závěru kapitoly zhodnotím jejich použití v navrhovaném zařízení a prodiskuji jejich výhody či nevýhody.

V dalších dvou kapitolách již podrobně popíši parametry dvou vybraných rozhraní. Proberu parametry datového přenosu, podporované přenosové rychlosti, zapojení konektorů a další parametry. Od 5. kapitoly se už budu zabývat konkrétní realizací watchdogu, kde uvedu použité součástky, popis zapojení, mechanické konstrukce a obslužného softwaru.

1 Přehled možných řešení watchdog

Tato kapitola se zabývá možnostmi realizace monitorovacího zařízení (dále jen watchdog), které v případě, že nedostane od monitorovaného počítače do určité doby signál, zajistí jeho restart.

1.1 Továrně vyráběná zařízení

Při hledání továrně vyráběných zařízení watchdog jsem narazil na dvě zahraniční firmy, které se výrobou takovýchto zařízení profesionálně zabývají. Je to americká firma Berkshire Products, Inc. [1] a německá QUANCOM Informationssysteme GmbH [2].

Firma Berkshire Products má ve své nabídce několik variant lišících se funkcemi, způsobem připojení a samozřejmě i cenou. Nejlevnějším zařízením v nabídce je interní USB watchdog za 100 USD. Klasická funkce watchdog je zde doplněna ještě o monitorování teploty vestavěným senzorem, např. zda nedošlo k poruše ventilátorů. O něco dražší, asi za 125 USD, je verze na sériový port, která obsahuje průchozí sériovou linku, umožňuje tak i monitorování komunikace mezi dvěma zařízeními. Jak sériová, tak USB verze je nabízena mimo zmiňovaných interních ještě v externím provedení. Dále jsou v nabídce provedení do ISA a PCI-express slotů nebo s rozhraním Ethernet + USB v cenách kolem 150 USD. Většina z nabízených produktů funguje jak pod MS Windows tak pod Linuxem mimo PCIe a Ethernet + USB verze, které podporují zatím jen MS Windows.

QUANCOM Informationssysteme vyrábí jen zařízení v podobě ISA nebo PCI rozšiřujících karet. V nabídce jsou tři verze do ISA slotu a dvě do PCI. Nejlevnější dva ISA watchdogy jsou na první pohled prakticky stejné konstrukce. Jednodušší verze je nabízena za 64 USD. Druhá verze za 88 USD je navíc rozšířena o možnost použít druhé relé ke spínání dalšího externího zařízení. Cena PCI verzí se pohybuje přibližně od 107 USD za PWDOG1 do 138 USD za PWDOG2. Obě PCI verze podporují standard Plug&Play. Druhá ze zmiňovaných verzí PWDOG2 je rozšířena o podporu ATX desek bez vyvedeného resetu. Posledním z nabídky je ISA Watchdog3 za 230 USD, který má čtyři opticky oddělené vstupy a teplotní senzor. Všechny watchdogy firmy QUANCOM Informationssysteme podporují jak operační systémy MS Windows tak Linux.

Bližší info o všech produktech lze získat na internetových stránkách výrobců [1], [2]. Většina těchto zařízení je určena pro profesionální použití v systémech, kde by mohl mít jakýkoliv výpadek velmi vážné následky, tomu odpovídá také cena těchto zařízení.

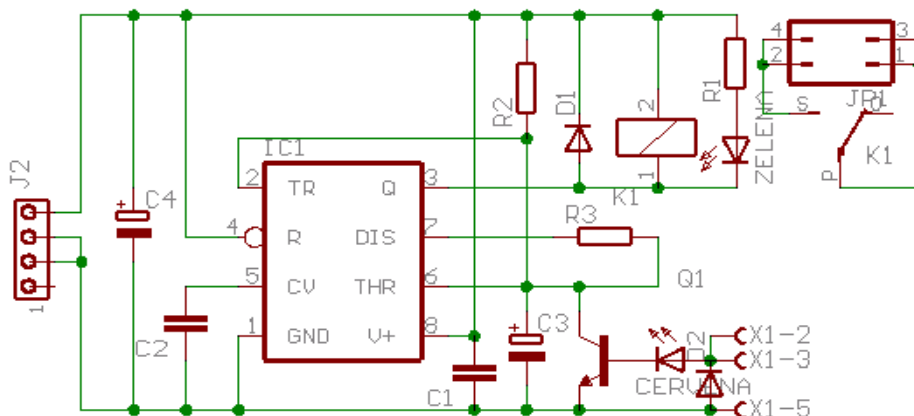
1.2 Zařízení vlastní konstrukce

1.2.1 Watchdog bez mikroprocesoru

Watchdog lze realizovat i bez použití řídicího mikroprocesoru, například použitím obvodu 555 nebo podobných integrovaných časovačů. V určitém časovém intervalu musí být časovač resetován, jinak krátkodobě (řádově na několik sekund) sepne výstupní relé, a tím restartuje monitorovaný počítač. Tento časový interval musí být zároveň dostatečně dlouhý, aby umožnil bezpečný start OS monitorovaného počítače a nedošlo k cyklickým restartům. Po startu OS je automaticky spuštěn software, který při správné činnosti systému pravidelně generuje resetovací impuls pro časovač připojený sériovým případně paralelním portem.

Takové řešení je sice jednoduché, ale má několik omezení. Většinou se používá pevně nastavený timeout. Obvod reaguje na jakýkoliv příchozí signál, není schopen rozlišit jestli tento signál byl určen právě jemu, nebo šlo o náhodně odeslaná data. Problémem může také nastat, pokud chceme z nějakého důvodu watchdog dočasně zakázat, v tomto případě to softwarově nelze, takže jedinou možností je odpojit obvod od počítače.

Příkladem této koncepce je velmi jednoduchý watchdog s obvodem 555 [3] na obrázku 1.



Obrázek 1: Jednoduchý watchdog s obvodem 555 (Převzato z [3])

1.2.2 Watchdog s mikrokontrolérem

Mikrokontroléry se v dnešní době rozšířily do všech oblastí vývoje elektrotechnických zařízení. Použití mikrokontroléru má hned několik výhod. MCU lze snadno propojit s počítačem pomocí sériového portu nebo USB a konkrétními příkazy je možné watchdog dočasně nebo i trvale zakázat nebo povolit, nulovat časovač watchdogu, případně i restartovat monitorovaný počítač.

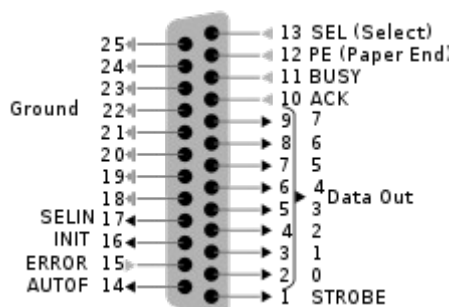
Stejnou cestou lze také provádět konfigurace jednotlivých parametrů, jako časový interval během kterého musí dorazit příkaz k vynulování časovače, jinak dojde k restartu (timeout), čas za jak dlouho po zapnutí se aktivuje funkce watchdogu atd. Výslednou konfiguraci je možné uložit do paměti EEPROM, (např. vnitřní EEPROM mikrokontroléru, pokud je použit MCU s integrovanou EEPROM pamětí), aby veškeré nastavení zůstalo zachováno i po odpojení watchdogu od napájení.

2 Komunikace s počítačem

Aby mohl watchdog správně plnit svou funkci, musí nějakým způsobem komunikovat s monitorovaným počítačem. K tomu je možné použít některé z běžných rozhraní počítače jako např. paralelní port, konektor klávesnice, sériový port nebo USB.

2.1 Paralelní port (LPT)

Paralelní port byl původně vytvořen pro komunikaci PC s tiskárnou, proto umožňoval pouze jednosměrný přenos dat. Později byl dalšími módy rozšířen o obousměrný přenos dat rychlostí až jednotek MB/s. Zapojení konektoru viz následující obrázek.



Obrázek 2: Zapojení konektoru paralelního portu.



Obrázek 3: Konektor DSub 25 na straně PC

Obrázky převzaty z [13].

Základní parametry:

- Paralelní komunikace.
- Konektor CANNON 25.
- Obsahuje 17 digitálních linek z toho 8 datových a 9 pro řízení komunikace.
- Používá napěťové úrovně TTL.
- Délka kabelu 2 m, při správném stínění datových vodičů lze prodloužit na max. 5 m.
- Výstupní proud 4 až 20 mA v závislosti na fyzické realizaci portu. Výstupy obvykle nejsou chráněny proti přetížení, proto je vhodné použít oddělovací buffer, aby nedošlo k přetížení portu a následně jeho zničení [14].

Módy paralelního portu :

SPP Mode

- definuje jen přenos typu forward (od PC k periferiím),
- přenosová rychlost do 150 kB/s.

Nibble Mode

- obousměrná komunikace na SPP paralelním portu využitím 5 stavových linek, pomocí kterých periferie odešle byte směrem do PC jako posloupnost dvou půlbytů (4 bitů),
- přenosová rychlost 100 kB/s.

Byte Mode

- umožňuje obousměrnou 8 bitovou komunikaci po datových linkách,
- přenosová rychlost 100 až 200 kB/s.

EPP (Enhanced Parallel Port)

- umožňuje poloduplexní obousměrný přenos dat,
- přenosová rychlost 500 kB/s až 2 MB/s.

ECP (Extended Capability port) [4]

- poloduplexní obousměrný přenos dat,
- využívá FIFO buffery a DMA,
- možnost využít RLE komprese při přenosu dat,
- přenosová rychlost bez použití DMA 800 kB/s a 2 MB/s při použití DMA při implementaci na ISA sběrnici. Při implementaci na PCI lze dosáhnout až 5 MB/s.

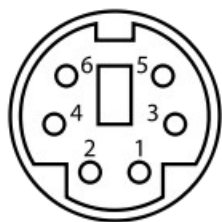
2.2 Port klávesnice

Můžeme se setkat s dvojitým provedením klávesnicového portu. Ve starším provedení, známém z počítačů standardu PC/AT, byl použit 5pinový DIN konektor. Ten byl s příchodem standardu ATX nahrazen konektorem PS/2 (Mini-DIN 6). Obě verze jsou ale elektricky shodné, klávesnici s konektorem PS/2 tedy lze přes vhodnou redukci připojit na starší konektor DIN5 a naopak.

Základní parametry

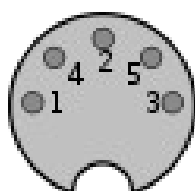
- synchronní sériové rozhraní s napěťovými úrovněmi TTL,
- sériový přenos o frekvenci 10 až 16 kHz s 1 startovacím bitem, 8 datovými bity, 1 stop bitem a 1 paritním bitem [16].

Zapojení konektorů:



Pin 1	DATA	Data
Pin 2	NC	Nezapojeno
Pin 3	GND	Zem
Pin 4	Vcc	+5 V DC 275 mA
Pin 5	CLK	Hodinový signál
Pin 6	NC	Nezapojeno

Tabulka 1: Konektor klávesnice PS/2 [16]



Pin 1	CLK	Hodinový signál
Pin 2	DATA	Data
Pin 3	NC	Nepřipojen.
Pin 4	GND	Zem
Pin 5	Vcc	+5V DC

Tabulka 2: Konektor klávesnice DIN5 [15]

2.3 Sériový port

Základní parametry

- Asynchronní obousměrné sériové rozhraní.
- Přenosová rychlost až 115 200 baudů.
- Délka vedení 15 m nebo délka vodiče o kapacitě 2 500 pF. Při použití kvalitních vodičů nebo nižších přenosových rychlostí lze ještě prodloužit.

- Napěťové úrovně Log. 1 v rozsahu -5 V až -15 V , Log. 0 v rozsahu $+5\text{ V}$ až $+15\text{ V}$ pro vysílač a Log. 1 v rozsahu -3 V až -25 V , Log. 0 v rozsahu $+3\text{ V}$ až $+25\text{ V}$ pro přijímač [5].

2.4 Sběrnice USB

USB je univerzální obousměrné sériové rozhraní s plnou podporou standardu Plug&Play.

Základní parametry:

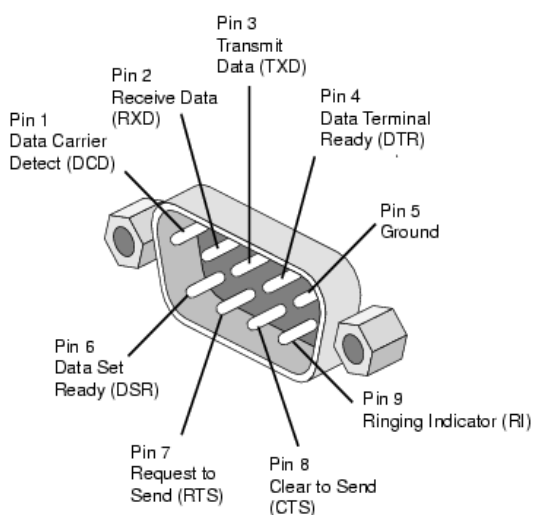
- Sériové rozhraní.
- Přenosové rychlosti od $1,5\text{ Mb/s}$ do 480 Mb/s .
- Možnost připojit různá zařízení až na vzdálenost 5 m .
- Při použití hubů (rozbočovačů) lze připojit až 127 zařízení.
- Napájení zařízení z konektoru USB.
- Běžně lze odebírat 100 mA , po enumeraci max. 500 mA .

2.5 Komunikace s počítačem – Shrnutí

Alespoň jeden paralelní port lze nalézt především na většině starších základních desek PC. Na moderních základních deskách PC již nemusí být vyveden. Pokud budeme chtít watchdog připojit k speciálním routerboardům, nastane podobný problém. Také velký konektor paralelního portu by zvětšoval velikost výsledného zařízení, ale výhodou může být to, že používá napěťové úrovně TTL, což usnadňuje propojení s řídicím mikrokontrolerem watchdogu. Nemusí se používat žádný převodník stejně jako u portu klávesnice, který pracuje též s 5voltovou logikou. Výhodou klávesnicového portu jistě je to, že ho lze nalézt ať už ve starší či novější verzi PS/2 prakticky na všech základních deskách PC, ale nemusí být obsažen na deskách určených přímo pro routery („routerboardech“). Asi bude spíše vhodný pro jednodušší watchdogy, kde by se přes něj jen předávala informace k nulování časovače např. blikáním stavových LED klávesnice. Jako nejvhodnější se jeví použití sériového portu nebo rozhraní USB. Alespoň jedno z těchto dvou rozhraní u routeru pravděpodobně bude k dispozici. Další výhodou je softwarová kompatibilita při použití převodníku USB / UART. K tomu lze s výhodou použít například integrované obvody FTDI FT-232, jejichž ovladače v systému vytvoří virtuální sériový port, se kterým potom software watchdogu pracuje jako s klasickým sériovým portem. Lze tedy jednoduše použít stejný software jak pro sériovou, tak pro USB verzi zařízení. V následujících kapitolách tato rozhraní popíší trochu podrobněji.

3 Sériový port RS-232

Standard RS-232C vznikl roku 1969. Původně byl koncipován pro připojení textových terminálů k jednoduchému modemu nebo blízkému serveru, postupem času se však jeho působnost dosti rozšířila, používal se k připojení počítačové myši, perových i řezacích plotterů, některých tiskáren a telefonních modemů [6]. Dodnes se používá jako komunikační rozhraní počítačů a další elektroniky. V některých oblastech elektroniky je stále rozšířen pro své specifické rysy. Například se pomocí něho mohou programovat regulátory motorů v průmyslových strojích, frekvenční měniče, mikroprocesory, paměti a další.



Vývod	Zkratka	Směr	Název
1	CD	vstup	Carrier Detect
2	RxD	vstup	Receive Data
3	TxD	výstup	Transmit Data
4	DTR	výstup	Data Terminal Ready
5	GND	—	Signal Ground
6	DSR	vstup	Data Set Ready
7	RTS	výstup	Request To Send
8	CTS	vstup	Clear To Send
9	RI	vstup	Ring Indicator

Obrázek 4: Konektor RS-232 DB-9 [7]

3.1 Základní parametry RS-232

RS-322 je komunikační rozhraní pro obousměrnou komunikaci dvou zařízení na vzdálenost do 20 m. Pro větší odolnost proti rušení jsou data přenášena větším napětím než standardních 5 V. Datový přenos probíhá asynchronně, pevně nastavenou přenosovou rychlostí. Synchronizace je zajištěna sestupnou hranou startovacího impulsu (startovacího bitu).

3.1.1 Délka vedení

Standard RS-232 udává jako maximální délku vedení při přenosové rychlosti 19 200 Bd 15 m nebo délku vodiče o kapacitě 2 500 pF [5]. To znamená, že při použití kvalitních vodičů a při zachování jmenovité kapacity lze dodržet standard a prodloužit

vzdálenost až na cca 50 m. Kabel lze prodlužovat také při snížení přenosové rychlosti, protože potom bude přenos odolnější vůči větší kapacitě vedení.

3.1.2 Napěťové úrovně

RS-232 definuje napěťové úrovně Log. 1 v rozsahu -5 V až -15 V , Log. 0 v rozsahu $+5\text{ V}$ až $+15\text{ V}$ pro vysílač a Log. 1 v rozsahu -3 V až -25 V , Log. 0 v rozsahu $+3\text{ V}$ až $+25\text{ V}$ pro přijímač [5].

Úroveň	Vysílač	Přijímač
Log. 0 (L)	$+5\text{ V}$ až $+15\text{ V}$	$+3\text{ V}$ až $+25\text{ V}$
Log. 1 (H)	-5 V až -15 V	-3 V až -25 V
Zakázané pásmo	-5 V až $+5\text{ V}$	-3 V až $+3\text{ V}$

Tabulka 3: Napěťové úrovně RS-232

3.1.3 Parametry datového přenosu

Sériový port RS-232 používá, jako každé jiné zařízení UART (*Universal Asynchronous Receiver and Transmitter*) – asynchronní přenos dat. Data jsou přenášena konstantní rychlostí jako sekvence 5, 6, 7, nejčastěji 8 bitů (1 byte). Aby bylo možné zahájit komunikaci, obě zařízení, jak vysílač tak přijímač, musejí být nejdříve vhodným způsobem nakonfigurována. Přijímač musí vědět v jakém formátu má data očekávat a jakou rychlostí vzorkovat datovou linku (jaká přenosová rychlost je použita) a stejně tak vysílač musí vědět, v jakém formátu má data vysílat a jakou rychlostí. Na obou zařízeních se proto musí nastavit kolik datových bitů je přenášeno v jednom celku, přenosová rychlost, délka stop bitu, zda a jakým způsobem je přenášen paritní bit.

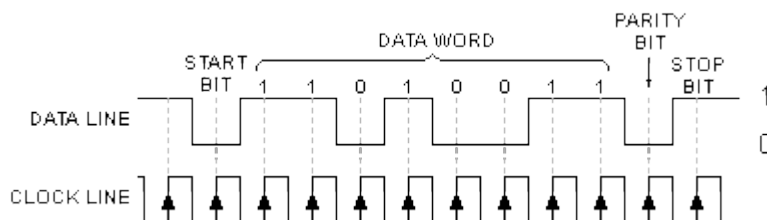
V klidovém stavu je datový vodič v Log. 1, což znamená záporné napětí. Tato úroveň může trvat libovolně dlouhou dobu. Komunikace je zahájena startovacím bitem, kterým je vždy Log. 0, tím je zaručeno, že se klidový stav linky vždy změní, aby přijímač správně rozeznal začátek komunikace – viz obrázek 5 (převzato z [5]).



Obrázek 5: Synchronizace RS-232

Za startovacím bitem následují již vlastní data (5 až 8 bitů). Na konci datové sekvence může být, v závislosti na konfiguraci, paritní bit. Ihned za ním následuje stopovací bit. Ten má vždy hodnotu Log. 1. Délka stop bitu může být delší než délky

předešlých datových bitů, aby měl přijímač dostatek času na zpracování přijatých dat. Lze použít délku rovnou například 1,5- či dvojnásobku délky běžného datového bitu.



Obrázek 6: Přenos 8 datových a 1 paritního bitu [5]

3.1.4 Přenosové rychlosti

Nejčastěji se používají přenosové rychlosti odvozené od násobků 300 baud, ovšem některá starší zařízení používala i nižší rychlosti, například 50, 75, 110, či 150 baud. Přehled všech dostupných přenosových rychlostí sériového portu je uveden v následující tabulce [8].

Přenosové rychlosti sériového portu [Bd]
50 baud
75 baud
110 baud
134,5 baud
150 baud
200 baud
300 baud
600 baud
1200 baud
1800 baud
2400 baud
4800 baud
9600 baud
19200 baud
38400 baud
57600 baud
76800 baud
115200 baud

Tabulka 4: Přenosové rychlosti sériového portu

3.1.5 Řízení toku dat

Některá zařízení zpracovávají data rozdílnou rychlostí než jiná, aby spolu tato zařízení (například počítač a telefonní modem) mohla komunikovat beze ztráty informací, je nutné tok dat nějakým způsobem řídit.

Řízení toku dat na sériovém portu lze provádět dvěma způsoby, buď softwarově nebo hardwarově. K softwarovému řízení se používají znaky ASCII 17 (X-ON) a 19 (X-OFF). Příkladem může být komunikace počítače s telefonním modemem, který komunikuje s počítačem vyšší rychlostí než je rychlost odesílání a příjmu dat po telefonní lince. Pokud modem nestíhá zpracovávat data posílaná počítačem, odešle po sériové lince ASCII znak X-OFF. Počítač po přijetí tohoto znaku přestane posílat další data. Jakmile je modem opět připraven přijímat data, vyšle znak ASCII X-ON.

Hardwarové řízení toku dat namísto odesílání znaků ASCII používá k předávání signálu, že zařízení již nemůže přijímat další data, linky RTS/CTS nebo DTR/DSR. Oba způsoby jsou prakticky totožné, pouze se k přenášení signálů používá jiná dvojice pinů.

4 Sběrnice USB

USB (Universal Serial Bus) je univerzální obousměrné sériové rozhraní s plnou podporou standardu Plug&Play, zařízení lze tedy připojovat a odpojovat za provozu.

Základní vlastnosti:

- Sériové rozhraní.
- Přenosové rychlosti od 1,5 Mb/s do 480 Mb/s.
- Možnost připojit různá zařízení až na vzdálenost 5 m.
- Při použití hubů lze připojit až 127 zařízení (USB používá 7bitovou adresaci).
- USB automaticky zajišťuje správné přidělení prostředků (IRQ, DMA, ...).
- Možnost napájení zařízení z USB konektoru. Běžně lze odebírat 100 mA, při speciálním přihlášení (enumeraci) si zařízení může vyžádat až 500 mA.
- Podpora ve všech běžně používaných operačních systémech (MS Windows, Linux, Mac OS).

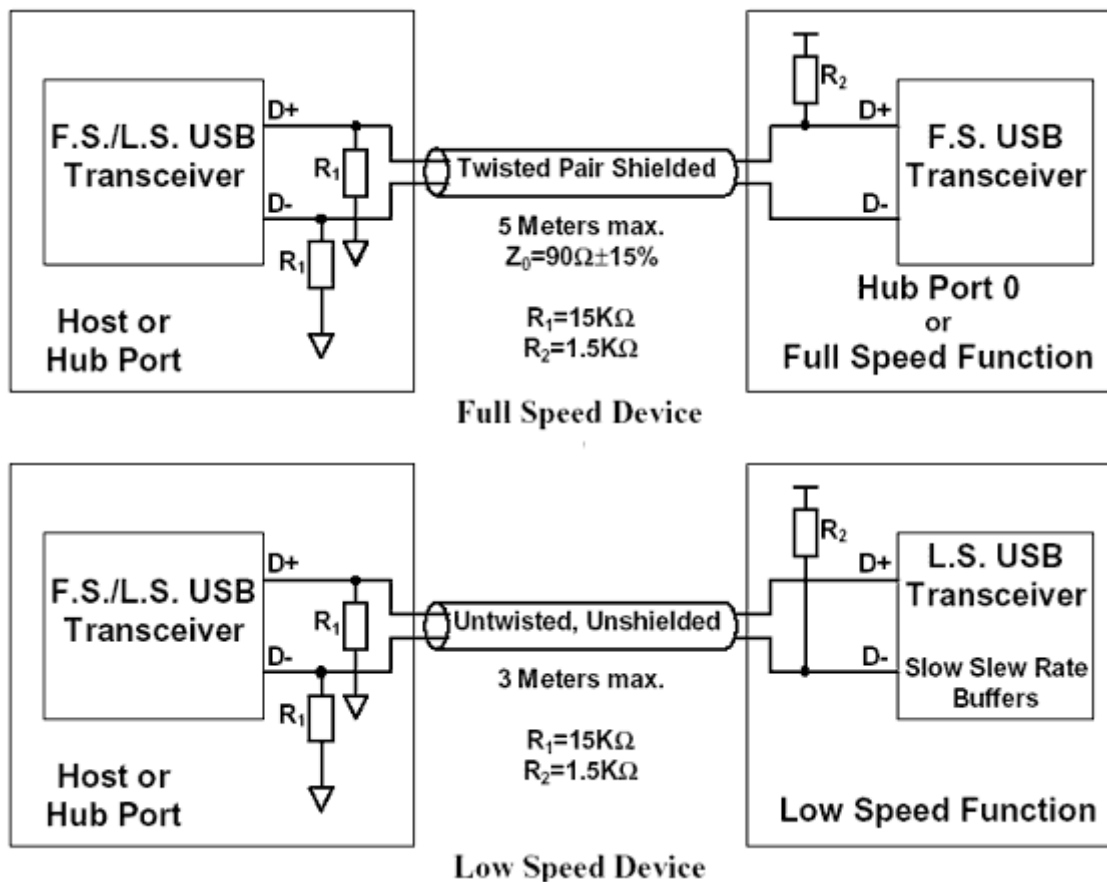
4.1 Přenosové rychlosti a verze USB

Standard USB 1.1 umožňoval přenosovou rychlost do 12 Mb/s, to bylo postupem času nedostačující, proto v roce 2000 vznikla nová verze USB 2.0. K rychlostem 1,5 Mb/s a 12 Mb/s, které umožňovala verze 1.1, přidává specifikace USB 2.0 rychlost přenosu dat 480 Mb/s (viz tabulka 5). Nová verze USB zároveň zachovává zpětnou kompatibilitu s předchozí verzí, takže všechna zařízení vyhovující standardu USB1.1, lze bez problémů připojit i na USB 2.0.

Označení	Přenosová rychlost	Standard
Low Speed	1,5 Mb/s	USB 1.1 / 2.0
Full Speed	12 Mb/s	USB 1.1 / 2.0
High Speed	480 Mb/s	USB 2.0

Tabulka 5: Standardy USB a přenosové rychlosti

Použitou přenosovou rychlost definují sama zařízení. Zařízení **Low Speed** připojuje pull-up rezistor 1,5 k Ω mezi D⁻ a 3,3 V, zařízení **Full Speed** připojuje zvyšovací (pull-up) rezistor 1,5 k Ω mezi D⁺ a 3,3 V (viz obrázek 7).



Obrázek 7: Zařízení Full Speed / Low Speed [11]

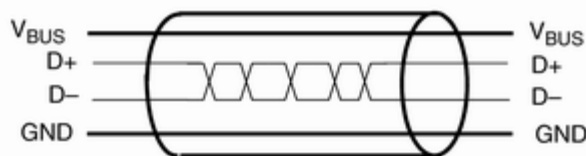
Připojením rezistorů pull-up na D+ nebo D- zároveň hubu sděluje, že je připojeno zařízení, protože jinak jsou linky taženy směrem k nule snižovacími (pull-down) rezistory 15 k Ω (viz obrázek 7) [9].

Zařízení High Speed se detekuje stejně jako zařízení Full Speed s tím, že změna rychlosti je řešena programově. Při připojení vysokorychlostního zařízení High Speed s rozhraním USB 2.0 a hubu či počítačového adaptéru USB 1.1, bude zařízení USB 2.0 pracovat v pomalém módu 1.1 na 12 Mb/s.

4.2 USB kabely a konektory

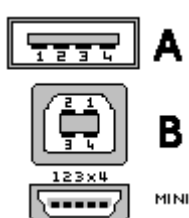
K propojení zařízení se používá čtyřžilový kabel se strukturou dle obrázku 8 [10]. Vodič označený jako GND propojuje země obou zařízení, vodič VBUS vede napájecí napětí +5 V, pro napájení připojených periférií. Zařízení s malým odběrem (do 500 mA) tak nemusejí mít vlastní napájecí zdroj, což zjednodušuje jejich konstrukci. Zbylé dva vodiče, D+ a D-, slouží pro sériový poloduplexní přenos dat s využitím diferenčního kódování, čímž se dosáhne snížení vlivu rušení. Pro přenosové rychlosti 12 Mb/s a 480 Mb/s je vyžadována kroucená dvojlinka o charakteristické impedanci 90 Ω (odchylka

od této hodnoty by neměla být větší jak 15 %), u přenosové rychlosti 1,5 Mb/s se nemusí nutně použít, ale je doporučena.



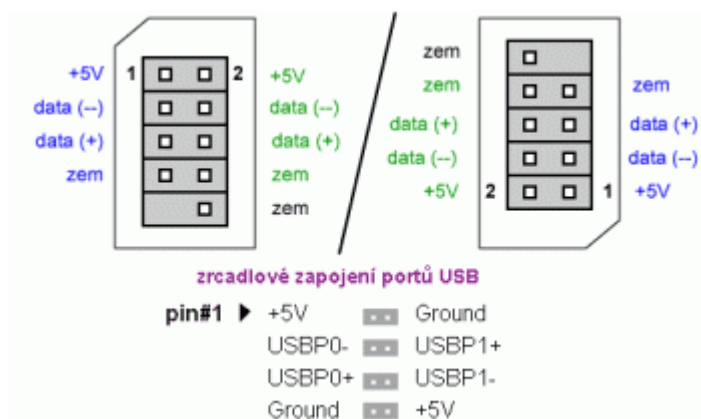
Obrázek 8: Struktura kabelu USB

Počítač je vybaven standardně dvěma nebo čtyřmi, v případě nových základních desek i více, konektory typu A (viz obrázek 9).



Pin	Název	Barva	Popis
1	VBus	Červená	+5 V DC
2	D-	Bílá	Data-
3	D+	Zelená	Data+
4	GND	Černá	Zem

Obrázek 9: Konektory USB



Obrázek 10: Zapojení pinů portu USB na základní desce [12]

Další porty USB bývají zpravidla vyvedené pomocí pinů na základní desce. Lze je využít například k připojení čtečky paměťových karet nebo mohou být vyvedeny na panel počítače jako konektory typu A. Zapojení pinů se může u různých výrobců mírně lišit. Na

nových základních deskách se nejčastěji používá uspořádání 2×5 nebo 4+5, jejichž zapojení je na obrázku 10.

Menší zařízení, jako například myši, používají pevně připojený kabel zakončený konektorem typu A. Zařízení s odnímatelným kabelem jsou obvykle vybavené konektorem USB typu B (viz obrázek 9). Pro zařízení, kde z rozměrových důvodů nelze použít standardní konektor typu B, existuje ještě miniaturní provedení Mini USB, ten lze nalézt například v mobilních telefonech nebo menších čtečkách paměťových karet.

4.3 Enumerace – rozpoznávání zařízení

USB podporuje Plug&Play, takže každé zařízení, které připojíme, musí být automaticky rozpoznáno operačním systémem.

Enumerace zařízení spočívá v tom, že se operační systém dotazuje nově připojeného zařízení na určité parametry ve formě tzv. descriptorů [9]. Při tomto procesu si také zařízení může vyžádat povolení vyššího odběru proudu než je standardních 100 mA.

5 Vlastní návrh a konstrukce watchdogu

V této kapitole se budu zabývat konkrétním návrhem zařízení watchdog. Jak už jsem se zmínil výše, mým cílem bylo zkonstruovat watchdog, který bude možno připojit k většině počítačů. Proto nakonec vznikly dvě verze. Jedna pro připojení na sériový port a druhá se připojuje prostřednictvím sběrnice USB. Obě verze přitom zůstávají softwarově i firmwarově kompatibilní, díky použití převodníku USB FT-232RL firmy FTDI [19]. Jeho ovladače v systému vytvoří virtuální sériový port, takže není nutný žádný zásah do obslužných programů.

5.1 Řídící mikrokontrolér (MCU) ATTiny 2313

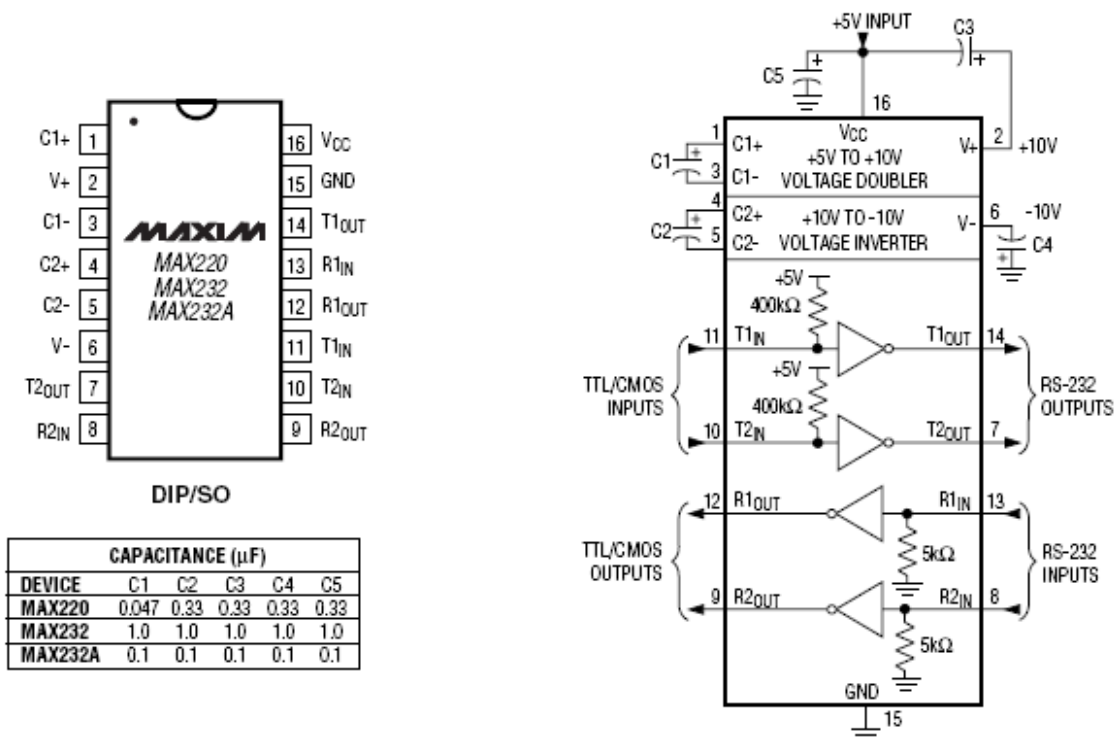
Srdcem celého zařízení je 8bitový mikrokontrolér Atmel ATTiny 2313 [17], postavený na architektuře AVR RISC. Disponuje 120 instrukcemi, z toho velké množství je jednocyklových, proto může dosáhnout propustnosti až 20 MIPS při frekvenci hodin 20 MHz. Tento mikrokontrolér jsem si vybral proto, že jeho architektura dovoluje použití vyšších programovacích jazyků. Dále tento typ, na rozdíl od ostatních malých mikrokontrolérů z rodiny ATTiny, v sobě obsahuje hardwarový USART (Universal Synchronous Asynchronous Receiver Transmitter). Sám zajišťuje serializaci a odvysílání dat zapsaných do registru sériového kanálu. Zefektivňuje tak programování celé sériové komunikace.

Základní parametry MCU ATTiny2313

Procesor obsahuje 2 kB paměti Flash programu a 128 B datové paměti EEPROM, (obě je možno programovat přímo v zařízení pomocí rozhraní ISP), 128 B datové paměti SRAM, jeden 16bitový a jeden 8bitový čítač/časovač, interní kalibrovaný oscilátor, watchdog timer, programovatelný monitor napájení Brown-out, full duplex USART.

5.2 Komunikace sériovým portem, obvod MAX232

Obvod mikrokontroléru USART podporuje stejný způsob komunikace jako sériový port PC (tj. asynchronní přenos dat), ale používá k tomu jiné napěťové úrovně. MCU pracuje s úrovněmi TLL, kdežto sériový port používá napěťové úrovně RS-232. Chceme-li tato dvě zařízení propojit, je nutné vřadit do cesty odpovídající převodník. K tomu byl použit snadno dostupný obvod MAX232 zapojený dle doporučení výrobce [18]. Obvod pracuje na principu nábojové pumpy. Ke své činnosti si vystačí pouze s napájením +5 V, odpadá tak potřeba dalšího napájecího napětí.



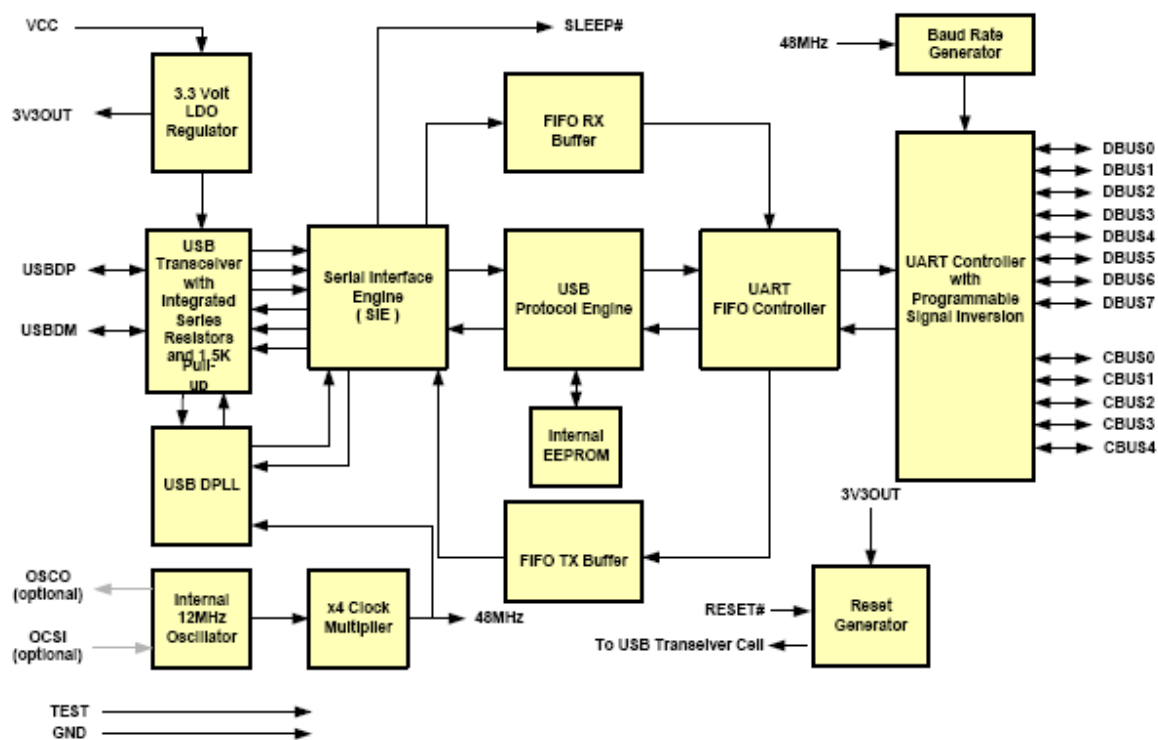
Obrázek 11: Integrovaný obvod MAX-231 [18]

5.3 Komunikace pomocí rozhraní USB, obvod FT-232RL

Obvod FT-232R je čtvrtou generací převodníků USB od firmy FTDI Chip. Oproti svému předchůdci FT-232BL přináší verze RL několik výrazných modernizací, především redukuje počet vnějších součástek. Tím se sníží náklady na výrobu a vývoj zařízení a v neposlední řadě lze také uspořit místo na desce výsledného zařízení. V pouzdře IO jsou již integrovány rezistory USB, konfigurační paměť EEPROM, zdroj hodinového signálu (není tak potřeba vnější krystal nebo zdroj hodin). Další novinkou jsou konfigurovatelné řídicí signály CBUS0 až CBUS4 [19].

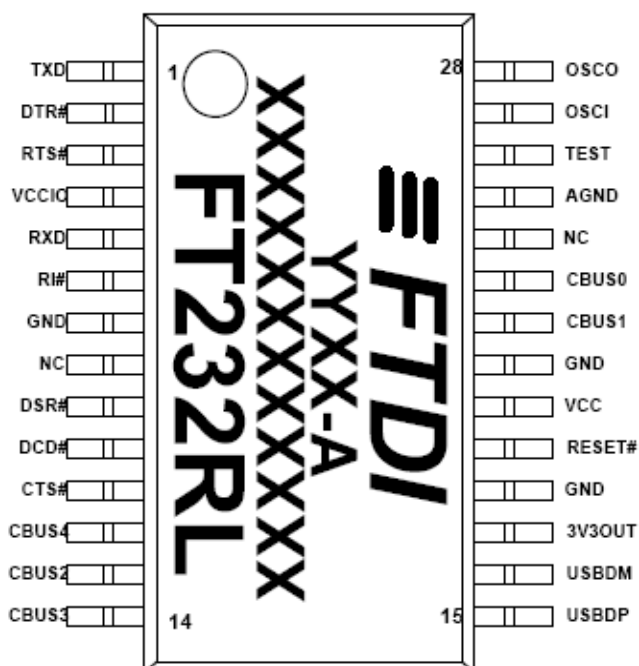
5.3.1 Zapojení a popis jednotlivých vývodů

Na obrázku 12 je blokové schéma obvodu FT-232R [19]. Integrovaný obvod je vyráběn ve dvou pouzdech – 32pinovém QFN pod označením FT232RQ a 28pinovém SSOP označeným FT232RL. Dále budu popisovat jen obvod v pouzdře SSOP, které je použitelné i pro ruční pájení.



Obrázek 12: Blokové schéma obvodu FT-232R [19]

Rozmístění vývodů na pouzdru SSOP znázorňuje obrázek 13.



Obrázek 13: Rozmístění vývodů obvodu FT232RL (pouzdro SSOP 28) [19]

Název	Popis
RXD, TXD	Vstup a výstup dat UART
DTR#, RTS#, RI#, DSR#, DCD#, CTS#	Linky modemu UART
VCC	Napájecí napětí 3,3 až 5,25 V
GND	Zem
AGND	Analogová zem pro interní oscilátor
VCCIO	1,8 V až 5,25 V napájecí napětí pro V/V obvody
3V3OUT	3,3 V výstup z vestavěného LDO stabilizátoru
USBDP	USB Data Plus
USBDM	USB Data Minus
RESET#	Resetovací vstup aktivní v nule. Pokud není třeba provádět vnější reset, může zůstat nepřipojen, nebo lze připojit na VCC.
TEST	Určen pro testování funkce výrobcem. Pro normální funkci obvodu musí být připojen na GND.
CBUS0 až CBUS4	Konfigurovatelné V/V vývody řídicí sběrnice

Tabulka 6: Vývody FT232RL

5.3.2 Signály CBUS

Funkce jednotlivých pinů CBUS0, 1, 2, 3, 4 je konfigurovatelná pomocí vnitřní paměti EEPROM. Přehled všech dostupných signálů udává tabulka 7.

Signál	Dostupný na pinu	Popis
TXDEN	CBUS0 až 4	Povolení vysílání dat pro RS485
PWREN#	CBUS0 až 4	Přechází do log. 0 po nakonfigurování obvodu přes USB. V USB suspend log. 1. Lze použít pro řízení P-kanálového tranzistoru MOSFET.
TXLED#	CBUS0 až 4	Výstup pro indikační LED vysílání dat přes USB
RXLED#	CBUS0 až 4	Výstup pro indikační LED příjmu dat přes USB
TX&RXLED#	CBUS0 až 4	Indikační LED příjmu nebo vysílání – sdružená funkce
SLEEP#	CBUS0 až 4	Při USB suspend v log. 0
CLK48	CBUS0 až 4	48MHz CLK výstup
CLK24	CBUS0 až 4	24MHz CLK výstup
CLK12	CBUS0 až 4	12MHz CLK výstup
CLK6	CBUS0 až 4	6MHz CLK výstup
CbitBangI/O	CBUS0 až 3	V režimu BitBang lze 4 vývody používat jako V/V (nutno nakonfigurovat v EEPROM)
CBitBangWRn	CBUS0 až 3	Signál zápisu WR# pro synchronní a asynchronní režim BitBang
CBitBangRDn	CBUS0 až 3	Signál zápisu RD# pro synchronní a asynchronní režim BitBang

Tabulka 7: Sběrnice CBUS [19]

Výchozí nastavení pinů CBUS

Pin	Výchozí nastavení od výrobce
CBUS0	TXLED#
CBUS1	RXLED#
CBUS2	TXDEN
CBUS3	PWREN#
CBUS4	SLEEP#

Tabulka 8: Výchozí konfigurace pinů CBUS

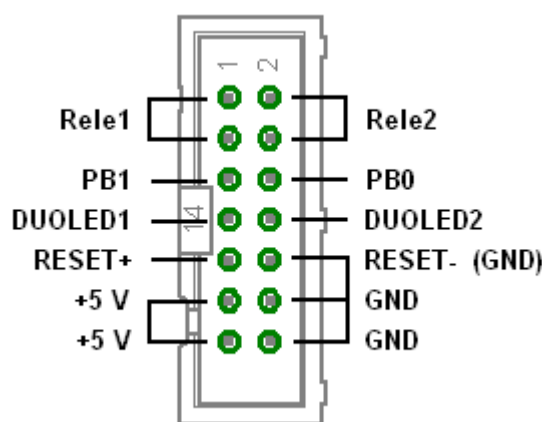
5.4 Popis zapojení

Zařízení lze rozdělit na dva funkční celky. Prvním je mikroprocesorová část s obvodem ATTiny2313, který řídí veškerou činnost watchdogu. Ten je u obou verzí prakticky totožný. Druhým funkčním celkem jsou obvody komunikačního rozhraní.

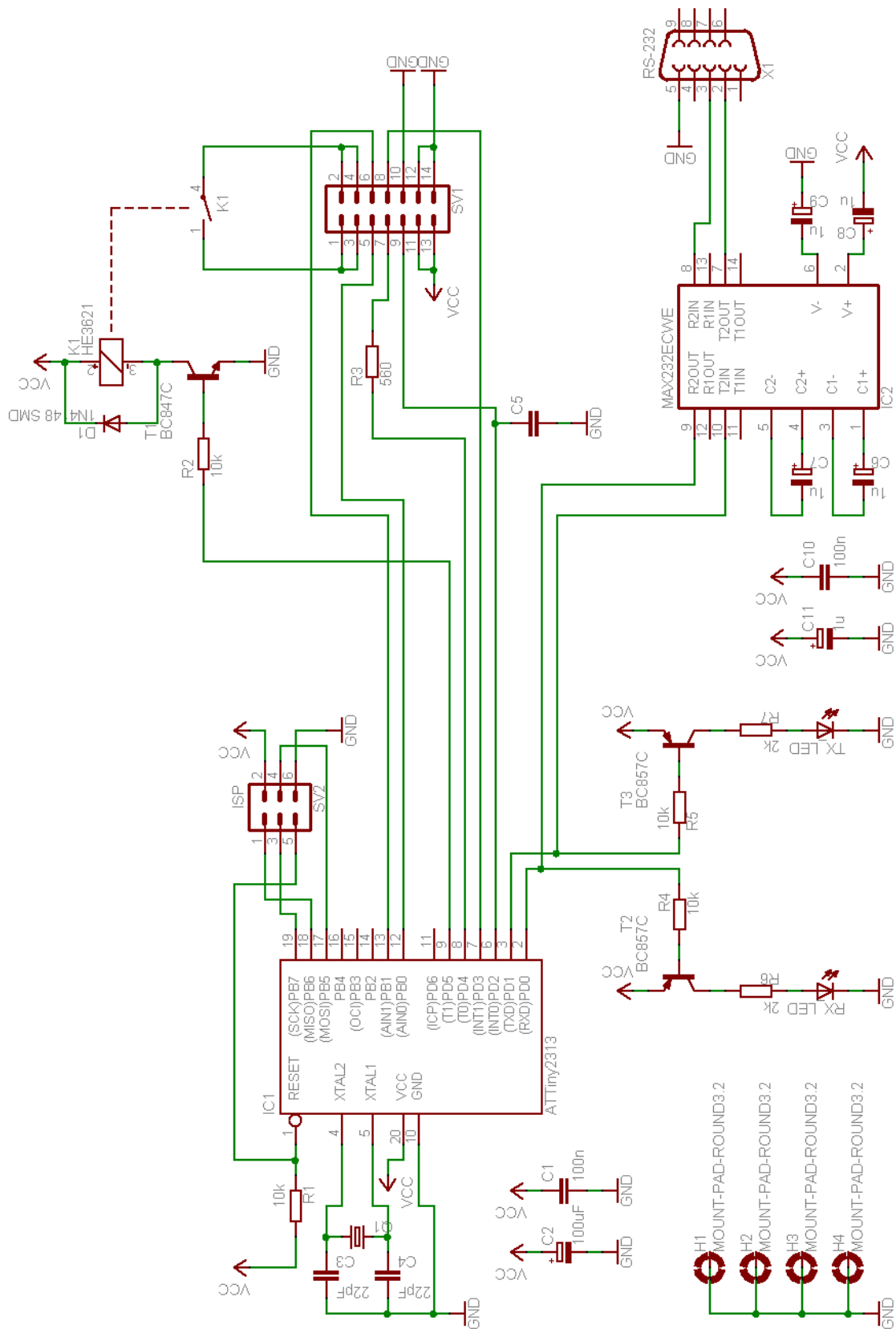
5.4.1 RS-232 verze

Základem celého zapojení je mikrokontrolér IC1. Kmitočet hlavního oscilátoru mikroprocesoru je nastaven krystalem Q1 na 7,372 8 MHz. Tato hodnota byla volena s ohledem na to, aby se celočíselným vydělením dalo dosáhnout některé ze standardních přenosových rychlostí sériového portu. Integrovaný obvod IC2 zajišťuje přizpůsobení napěťových úrovní sériového kanálu MCU napěťovým úrovním RS-232. Diody RX_LED a TX_LED společně s rezistory R4 až R7 a PNP-tranzistory T2, T3 indikují aktivitu na sériových linkách. V klidovém stavu je na linkách stále log. 1 a diody jsou zhasnuté. V případě, že přejde některá z linek do log. 0, proud tekoucí rezistorem R4 nebo R5 otevře odpovídající tranzistor a rozsvítí tak příslušnou LED. Jako indikační diody je vhodné použít nízkopříkonové typy, aby se zbytečně nezvyšoval odběr celého zařízení. Proud diodami je nastaven rezistory R6, R7 přibližně na 2 mA.

Ke spínání resetu monitorovaného počítače je watchdog vybaven miniaturním relé RE1, které je řízeno mikrokontrolérem prostřednictvím tranzistoru T1. Dioda D1 chrání tranzistor před napěťovými špičkami, které vznikají při odpojování indukční zátěže (cívky relé). Spínací kontakty relé jsou vyvedené na piny 1, 2 a 3, 4 hlavního konektoru SV1. Tento konektor v sobě sdružuje také přívod napájecího napětí, vstup resetovacího tlačítka počítače, výstup na stavovou dvoubarevnou antiparalelní diodu LED, a dva rozšiřující V/V piny, které jsou připojeny přímo na vývody PB0 a PB1 mikrokontroléru. Zapojení jednotlivých pinů konektoru SV1 viz obrázek 14.



Obrázek 14: Watchdog – zapojení konektoru SV1

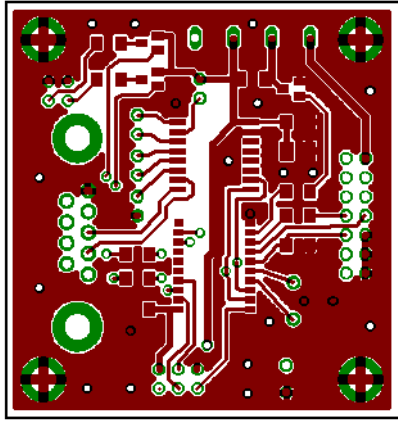


Obrázek 15: Watchdog – schéma verze RS-232

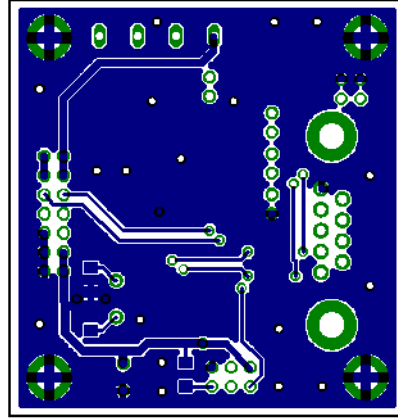
K připojení na sběrnici USB lze použít jednak klasický konektor USB typu B nebo interní čtyřpinový konektor ve schématu označený jako X2. Napájecí napětí z USB je vedeno přes feritovou perličku L1 a připojeno k napájecím vstupům obvodu FT-232RL. Kondenzátory C6 až C8 slouží k blokování a filtraci napájecího napětí převodníku. K indikaci datového přenosu je v tomto případě s výhodou použita sběrnice CBUS, která již obsahuje odpovídající výstupy. Dalším rozdílem oproti verzi na sériový port je to, že v USB verzi si lze zvolit, jak bude watchdog napájen. Máme na výběr ze dvou možností, buď z napájecího vstupu konektoru SV1 nebo z USB. Volbu provedeme zapojením nebo vyjmutím jumperu JP1. Je-li jumper zapojen, watchdog bude napájen z USB. V takovém případě nesmí být přivedeno napájecí napětí na konektor SV1, hrozilo by poškození portu USB. Tranzistor T3 automaticky připojuje napětí z USB na JP1 po úspěšné enumeraci zařízení počítačem. Společně s rezistorem R7 a kondenzátorem C10 také funguje jako obvod „soft start“.

5.5 Mechanická konstrukce

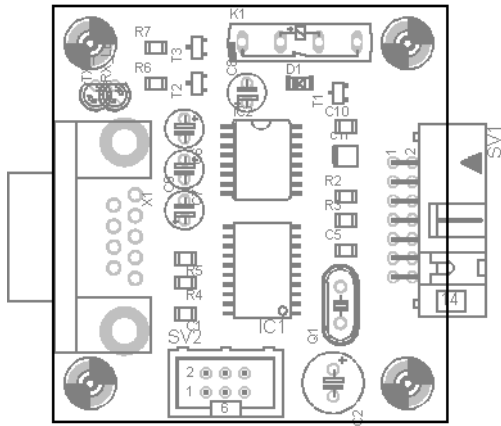
Obě verze watchdogu jsou navrženy na oboustrannou desku s prokovenými otvory a nepájivou maskou. Převážná většina použitých součástek je v provedení pro povrchovou montáž SMD. To umožnilo dosáhnout relativně malých rozměrů. Desky jsou vybaveny čtyřmi montážními otvory pro šrouby M3, kterými lze watchdog připevnit do skříně počítače například pomocí distančních sloupků. Další možností je použít plechovou záslepku z nějaké vyřazené karty PCI nebo ISA a pomocí ní upevnit desku watchdogu na zadní panel běžného ATX case do prostoru pro rozšiřující karty. Tomuto způsobu montáže bylo také přizpůsobeno rozmístění konektorů. Konektory USB, sériového portu a indikační LED datového přenosu jsou umístěny na straně plechové záslepky. Na protější stranu je umístěn hlavní konektor pro připojení napájecího napětí +5 V, resetovací tlačítka, výstupu relé a dvoubarevné stavové diody LED. Stavová LED tak může být umístěna na předním panelu počítače, nebo kdekoliv jinde dle uvážení uživatele, a s deskou watchdogu je propojena dvoužilovým kablíkem.



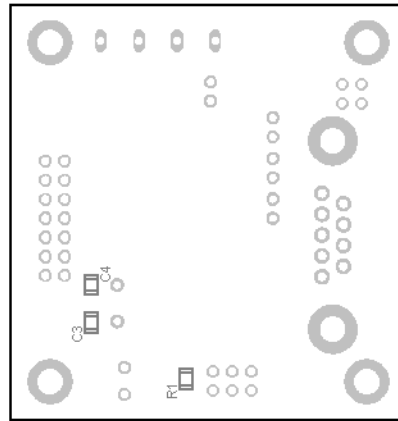
Obrázek 18: Watchdog RS-232 top



Obrázek 17: Watchdog RS-232 bottom



Obrázek 20: Watchdog RS-232
– rozmístění součástek top



Obrázek 19: Watchdog RS-232
– rozmístění součástek bottom

5.6 Rozpis součástek

Následující tabulky obsahují seznam všech použitých součástek, včetně orientační ceny za kus dle aktuálního ceníku GM Electronic [22]. Do uvedené celkové ceny nebyly započítány ceny propojovacích kabelů a montážního příslušenství.

5.6.1 Verze RS-232

Označení ve schématu	Hodnota	Počet kusů	Cena za kus
C1, C10	100 nF X7R, 1206	2 ks	2,50 Kč
C3, C4	22 pF NPO, 1206	2 ks	2,50 Kč
C11	1 μ F/35 V, SMD B	1 ks	4,00 Kč
C6, C7, C8, C9	1 μ F/35 V	4 ks	3,00 Kč
C2	100 μ F/16 V	1 ks	1,50 Kč
D1	1N4148 SMD	1 ks	1,00 Kč
Q1	Krystal 7,372 MHz HC49U-S	1 ks	10,60 Kč
R6, R7	2 k Ω , 1206	2 ks	2,00 Kč
R1, R2, R4, R5	10 k Ω , 1206	4 ks	2,00 Kč
R3	560 Ω , 1206	1 ks	2,00 Kč
IC1	ATTiny2313-20SU	1 ks	48,20 Kč
IC2	MAX232CWE	1 ks	26,80 Kč
Rel	RELSIA05-500 (RELSIA05D500)	1 ks	23,50 Kč
RX_LED	Červená LED 3 mm, 2 mA	1 ks	4,00 Kč
TX_LED	Zelená LED 3 mm, 2 mA	1 ks	4,00 Kč
–	L-937EGW červeno-zelená LED	1 ks	8,00 Kč
T1	BC847	1 ks	1,10 Kč
T2, T3	BC857	2 ks	1,10 Kč
X1	CAN 9 Z 90	1 ks	10,00 Kč
SV1	MLW14A 2 \times 7 pin	1 ks	8,50 Kč
SV2	MLW06G 2 \times 3 pin	1 ks	23,00 Kč
–	KONPC 2 pin	4 ks	3,00 Kč
–	Deska plošných spojů Watchdog RS-232 DPS	1 ks	150 Kč *
			Celkem 365,40 Kč

* Přibližná cena jedné desky při výrobě technologií POOL SERVIS ve firmě PragoBoard [23].

5.6.2 Verze USB

Označení ve schématu	Hodnota	Počet kusů	Cena za kus
C1, C7, C9, C10	100 nF X7R, 1206	4 ks	2,50 Kč
C3, C4	22 pF NPO, 1206	2 ks	2,50 Kč
C6	10 nF X7R, 1206	1 ks	2,50 Kč
C8	4,7 μ F/20 V, SMD B	1 ks	3,00 Kč
C2	100 μ F/16 V	1 ks	1,50 Kč
D1	1N4148 SMD	1 ks	1,00 Kč
Q1	Krystal 7,372 MHz HC49U-S	1 ks	10,60 Kč
R1, R2, R6	10 k Ω , 1206	3 ks	2,00 Kč
R3	560 Ω , 1206	1 ks	2,00 Kč
R4, R5	2 k Ω , 1206	2 ks	2,00 Kč
R7	1 k Ω , 1206	1 ks	2,00 Kč
IC1	ATTiny2313-20SU	1 ks	48,20 Kč
IC2	FT232RL	1 ks	150,00 Kč
JP1	Jumper propojka	1 ks	2,00 Kč
JP1	Jumper kolíky	1 ks	5,00 Kč
Re1	RELSIA05-500 (RELSIA05D500)	1 ks	23,50 Kč
L1	Feritová perlička	1 ks	4,00 Kč
T1	BC847	1 ks	1,10 Kč
T2	IRLML6402	1 ks	4,30 Kč
RX_LED	Červená LED 3 mm, 2 mA	1 ks	4,00 Kč
TX_LED	Zelená LED 3 mm, 2 mA	1 ks	4,00 Kč
–	L-937EGW červeno-zelená LED	1 ks	8,00 Kč
X1	USB1X90B PCB	1 ks	6,50 Kč
X2	1 \times 4 pin	1 ks	
SV1	MLW14A 2 \times 7 pin	1 ks	8,50 Kč
SV2	MLW06G 2 \times 3 pin	1 ks	23,00 Kč
–	KONPC 2 pin	4 ks	3,00 Kč
–	Deska plošných spojů Watchdog USB DPS	1 ks	120 Kč *
			Celkem 471,7 Kč

* Přibližná cena jedné desky při výrobě technologií POOL SERVIS ve firmě PragoBoard [23].

5.7 Popis funkce a stavová LED

Po připojení watchdogu k napájecímu napětí se nejprve inicializují veškeré použité periferie mikrokontroléru. Následně je načtena konfigurace z paměti EEPROM. Jestliže je watchdog povolen, rozsvítí se stavová LED zeleně, odpočet je nastaven na BootTimeout a watchdog čeká než nastartuje operační systém, maximálně však do doby než vyprší boot timeout. Příchodem prvního nulovacího signálu je odpočet nastaven na standardní timeout. Stavová LED bliká zeleně a signalizuje tak aktivní watchdog.

V případě, že do vypršení timeoutu nepříjde žádný signál nebo je stisknuto resetovací tlačítko, watchdog restartuje počítač sepnutím miniaturního relé. Po restartu je opět nastaven timeout na BootTimeout a čeká se na příchod prvního signálu.

Stavová LED	Popis
Svíí zeleně	Watchdog čeká než naběhne OS, maximálně však dobu \leq BootTimeout
Bliká zeleně	Watchdog aktivní, odpočet nastaven na standardní timeout
Svíí červeně	Signalizace sepnutého výstupního relé
Nesvíí	Watchdog je deaktivován

Tabulka 9: Stavová LED – popis signalizace

5.8 Firmware

Vzhledem k tomu, že použitý mikrokontrolér umožňuje programování ve vyšších programovacích jazycích, zvolil jsem si k programování jazyk C. Zdrojový kód byl přeložen překladačem AVR-GCC 4.1.2 s použitím knihovny avr-libc verze 1.4.6.

5.8.1 Ukázky zdrojového kódu

Po resetu jsou všechny piny mikrokontroléru nastavené jako vstupní a všechna přerušení zakázána. Proto je nejdříve nutné provést inicializaci portů a dalších potřebných periférií mikrokontroléru. Poté povolit příslušná přerušení nastavením registru TIMSK a povolit globální přerušení.

Inicializace periférií mikrokontroléru:

```
int main(void){
  /****** >> Inicializace portu << *****/
  // DDRx: log0 input, log 1 output
  // PORTx: input: 1 pull-up rezistor activated
  //           1 pin driven high, 0 pin driven low
  DDRD = (1<<PD1) | (1<<PD3) | (1<<PD4) | (1<<PD5); //PD1, 3, 4, 5 output
  PORTD = (1<<PD2); //PD2 pull-up ON

  /* Set baud rate 4800 baud, F_CLK = 7,3728MHz => UBRR = 0x005F */
  UBRRH = 0x00;
  UBRRL = 0x5F;
  /* Enable receiver and transmitter, RX interrupt enable */
  UCSRB = (1<<RXEN) | (1<<TXEN) | (1<<RXCIE);
  /* Set frame format: 8 data, 1 stop bit*/
  UCSRC = (3<<UCSZ0);

  /* Nastaveni 16bit T/C1*/
  TCCR1B = (1<<WGM12) | (1<<CS12); /*CLK 1/256, CTC (auto reload) */
  /* Nastaveni output compare match*/
  OCR1A = 0x7080; // 256 * 28800 * 0,13563368 uS = 1s (0,999999995904)
  TIMSK = (1<<OCIE1A); //Output Compare Match A interrupt enable

  read_conf(); //Nacteni konfigurace z EEPROM
  time_out = wd_pon_tout;
  ucBoot = 1;
  //SREG|=(1<<SREG_I); //Globalni povoleni preruseni, stejne jako sei();
  sei();

  if (wd_enabled==1) LED_color_Green();
  else LEDs_OFF();

  for(;;){
    ;
  }
}
```

Obsluha jednotlivých událostí, jako komunikace po sériové lince, měření času, sledování resetovacího tlačítka, obsluha stavové LED a výstupního relé atd. je řešena prostřednictvím přerušení.

Ukázka obsluhy sériové komunikace:

```
SIGNAL(SIG_USART0_RECV){
    usart_buffer[id_buff] = UDR;
    if (usart_buffer[0]==0xAB) if (++id_buff>10) id_buff = 0;

    if((id_buff!=0) && (usart_buffer[id_buff-1]=='#'))
    {
        switch (usart_buffer[1]) {
            /*Obsluha prijateho prikazu*/
        }
    }
}
```

K měření času je použit 16bitový čítač/časovač v režimu CTC (Clear Timer on Compare) společně s přerušením od Output Compare Match 1A. Čítač čítá směrem nahoru, ve chvíli, kdy je hodnota registru TCNT1 rovna Registru Output Compare OCR1A, je v následujícím cyklu hodin nastaven příznak OCF1A, čítač restartován od nuly a vygenerováno přerušení. Hodnota OCR1A je nastavena tak, aby přerušení přicházelo každou sekundu.

Ukázka měření času:

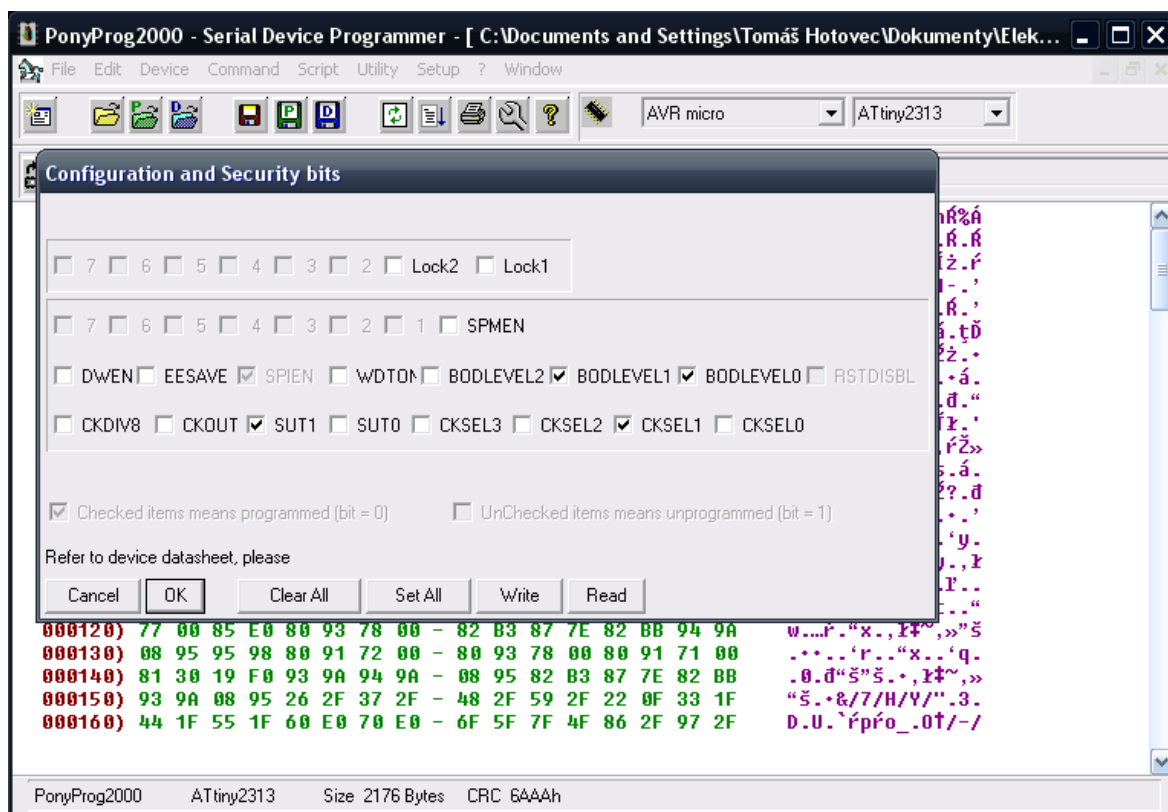
```
// ----- >> Output Compare match A Interrupt << -----
SIGNAL(SIG_OUTPUT_COMPARE1A){
    if (++sec > 59) {sec=0; if(time_out>0){time_out--;} }

    if (switch_t>0) {
        if (--switch_t==0) {
            reset_OFF();
            ucBoot = 1;
        }
    } //po uplynuti switch_t s vypnout rele

    if ( ( ( wd_enabled==1) && (time_out==0) ) || !(PIND & (1<<PD2)) ||
    ( ucRestart == 1) ) {
        LED_blik_OFF();
        reset_ON();
    }
}
```

5.8.2 Naprogramování firmware do mikrokontroléru

Programování firmware do mikrokontroléru se provádí přes standardní šestipinový konektor ISP zapojený dle dokumentace [20] firmy Atmel. K naprogramování proto lze použít jakýkoliv programátor kompatibilní s AVR ISP. Já jsem k tomu použil program PonyProg 2000 s jednoduchým hardware podle [21]. Při prvním programování je též nutné naprogramovat takzvané „fuse bits“, někdy uváděné jako propojky nebo konfigurační bity – viz obrázek 25. Zaškrtnuté políčko znamená naprogramováno (bit = 0), nezaškrtnuté znamená nenaprogramováno (bit = 1).



Obrázek 25: Konfigurační bity MCU

5.9 Konfigurace watchdogu, popis komunikace

Zařízení neobsahuje žádné konfigurační jumpéry ani přepínače. Jedinou výjimkou je jumper pro volbu napájení ve verzi USB, kterou je možné napájet buď z USB nebo z konektoru SV1. Veškerá konfigurace se provádí přes sériový port, ať už fyzický, či virtuální ve verzi USB. Aktuální firmware umožňuje konfigurovat dobu potřebnou pro start operačního systému monitorovaného počítače, timeout (čas, během které musí přijít příkaz pro nulování časovače), dobu sepnutí relé. Všechny zmiňované parametry jsou při konfiguraci automaticky uloženy do vnitřní paměti EEPROM mikrokontroléru. Dále umožňuje watchdog trvale nebo dočasně povolit/zakázat, načíst informace o verzi firmware, restartovat monitorovaný počítač a samozřejmě nulovat časovač watchdogu.

Komunikace s počítačem probíhá rychlostí 4 800 bitů za sekundu s následujícími parametry: 1 startovací bit, 8 datových bitů, 1 stopovací bit, bez parity a žádné řízení toku dat.

Konfigurace a řízení watchdogu je řešeno pomocí 3- až 5bytových instrukcí. Každá instrukce musí obsahovat počáteční a koncový byte, mezi kterými je odeslán samotný kód instrukce a případná data. Počáteční byte informuje watchdog o příchozí instrukci, jinak jsou příchozí data ignorována. Koncový byte sděluje zařízení, že příchozí instrukce byla kompletně přijata a může ji začít zpracovávat.

Počáteční byte	Kód instrukce (+ Data)			Koncový byte
0xAB	0x??	0x??	0x??	'#'

Nyní krátce popíši jednotlivé instrukce.

Instrukce SET

Skupina instrukcí sloužící ke konfiguraci parametrů watchdogu. Nastavení se ukládá do interní EEPROM a ihned po provedení instrukce se automaticky načte celá konfigurace z EEPROM. Tím se zároveň přepíše dočasné povolení/zakázání watchdogu hodnotou z EEPROM.

Aktivace / Deaktivace watchdogu

Povolí nebo zakáže funkci watchdogu, nastavení zapíše do EEPROM.

DATA = 0x00 – deaktivuje watchdog,

DATA = 0x01 – aktivuje watchdog.

0xAB	0x53 (ASCII 'S')	0x01	[DATA]	'#'
------	---------------------	------	--------	-----

Nastavení času pro bootování systému

Nastavuje délku prodlevy po zapnutí či restartu počítače, aby mohl bezpečně nastartovat operační systém, nastavení zapíše do EEPROM. Byte DATA udává čas v minutách v rozsahu 1 až 255 min. Pokud je zadána 0 mikrokontrolér použije výchozí hodnotu.

DATA = 0 – bude použita výchozí hodnota definovaná ve firmware.

DATA = 1 až 255 – nastaví požadovaný čas.

0xAB	0x53	0x02	[DATA]	'#'
------	------	------	--------	-----

Nastavení timeoutu

Nastaví timeout watchdogu, neboli interval, během kterého je nutné nulovat časovač, aby nedošlo k restartu.

DATA = 0 Bude použita výchozí hodnota definovaná ve firmware.

DATA = 1 až 255 Nastaví požadovaný čas v minutách.

0xAB	0x53	0x03	[DATA]	'#'
------	------	------	--------	-----

Nastavení délky sepnutí relé

Délku sepnutí relé lze nastavit dle potřeby, v rozsahu 1 až 255 sekund.

DATA = 0 Bude použita výchozí hodnota definovaná ve firmware.

DATA = 1 až 255 Nastaví požadovanou délku sepnutí v sekundách.

0xAB	0x53	0x04	[DATA]	'#'
------	------	------	--------	-----

Všechny výše uvedené instrukce jsou po úspěšném vykonání potvrzeny odesláním zprávy zpět do počítače. Formát potvrzující zprávy je shodný s formátem instrukce, jen na místě datového bytu je odeslán znak 'A'.

0xAB	[Kód instrukce]	'A'	'#'
------	-------------------	-----	-----

Řídící instrukce:

Nulování časovače watchdogu

Odesláním instrukce nulování časovače dává počítač signál watchdogu, že vše pracuje správně.

0xAB	'C'	'#'
------	-----	-----

O úspěšném provedení watchdog informuje zprávou:

0xAB	'C'	'A'	'#'
------	-----	-----	-----

Restart počítače

Tato instrukce provede okamžitý restart počítače a to i v případě, že watchdog je deaktivován.

0xAB	'R'	'#'
------	-----	-----

Dočasné aktivování / deaktivování

Umožní dočasně aktivovat nebo deaktivovat watchdog beze změny hodnoty v konfigurační EEPROM.

DATA = 'T' – aktivace watchdogu,

DATA = 'F' – deaktivace watchdogu.

0xAB	'E'	[DATA]	'#'
------	-----	--------	-----

Úspěšné vykonání je potvrzeno zprávou:

0xAB	'E'	[DATA]	'A'	'#'
------	-----	--------	-----	-----

Ostatní instrukce:

Informace o firmware

0xAB	'V'	'#'
------	-----	-----

Odpovědí je 16bytová informace, identifikující verzi firmware.

6 Obslužný software pro OS Linux

Pro komunikaci s watchdogem jsem si vytvořil vlastní knihovnu funkcí *thwdtlib.h*. Knihovna obsahuje veškeré funkce pro komunikaci se zařízením. Usnadňuje uživateli začlenění watchdogu do vlastní aplikace. Nemusí se již zabývat tím, v jakém formátu jsou odesílány jednotlivé příkazy do zařízení a odpovědi na ně. Tuto problematiku řeší už jednotlivé funkce.

6.1 Přehled knihovných funkcí thwdtlib.h

int wdt_clr_timer (int _fd)

Slouží k nulování časovače watchdogu. Pokud je watchdog aktivní, obslužný software musí v pravidelných intervalech volat tuto funkci a informovat tak watchdog o správné funkci systému či služby.

int wdt_tenable (int _fd, unsigned char _en)

Funkce dočasně aktivuje nebo deaktivuje watchdog. Prvním parametrem je file descriptor zařízení. Druhý parametr určuje, zda se jedná o aktivaci *_en = 1* nebo deaktivaci *_en = 0*. Nastavení provedené funkcí se obnoví na původní hodnotu uloženou v paměti EEPROM po odpojení watchdogu od napájení nebo odesláním některé z instrukcí SET, (například pomocí funkcí *wdt_set_**).

int wdt_set_enabled (int _fd, unsigned char _en)

Funguje obdobně jako předchozí funkce s tím rozdílem, že provedené nastavení je uloženo do EEPROM mikrokontroléru. Prvním parametrem funkce je file descriptor zařízení, druhý parametr *_en = 0* znamená deaktivaci, *_en = 1* aktivaci.

*int wdt_set_boot_tout (int _fd, unsigned char *_t)*

Nastavuje délku časového intervalu, po zapnutí nebo restartu počítače, kterou watchdog čeká, než nastartuje operační systém a dorazí první příkaz k nulování časovače. Hodnota je uložena do interní EEPROM mikrokontroléru. Prvním parametrem je file descriptor zařízení a druhý parametr udává čas v minutách. Vzhledem k použití 1B proměnné v mikrokontroléru, může být tento čas v rozsahu 0 až 255 minut. Pokud jako čas zadáme hodnotu 0 min, mikrokontrolér použije defaultní hodnotu 5 min definovanou ve firmware.

int wdt_set_tout (int _fd, unsigned char *_t)

Funkce odešle do watchdogu nové nastavení timeoutu. Hodnota bude uložena v interní EEPROM. Prvním parametrem funkce je opět file descriptor zařízení a druhým čas v rozsahu 0 až 255 min, předávaný hodnotou ukazatele jako v předchozím případě. Při zadání 0, mikrokontrolér použije defaultní hodnotu 2 min.

int wdt_set_relayswt (int _fd, unsigned char *_t)

Funkcí `wdt_set_relayswt()` můžeme změnit délku sepnutí resetovacího relé watchdogu, nastavená hodnota se ukládá do EEPROM mikrokontroléru. Čas lze v tomto případě volit v rozsahu 0 až 255 vteřin. Při zadání 0 mikrokontrolér použije defaultní hodnotu 1 s.

int wdt_version (int _fd, char *_buf, size_t _buflen)

Načte verzi firmware do bufferu `_buf` v podobě nulou zakončeného řetězce znaků. Pro načtení kompletní informace je zapotřebí buffer o velikosti alespoň 17 B (16 B zabírají informace o verzi + 1 B zakončení řetězce). Jestliže je zadán buffer s kratší délkou, nebude informace načtena celá, ale pouze část o velikosti délka_bufferu - 1. Poslední byte je použit na zakončení řetězce.

Návratová hodnota všech výše uvedených funkcí je rovna 0 při úspěšném provedení. V případě chyby, když watchdog neodpoví během určeného časového intervalu, je návratová hodnota rovna 1.

void wdt_restart (int _fd)

Ihned provede studený restart monitorovaného počítače, jako by bylo stisknuto tlačítko reset. Jediným parametrem funkce je file descriptor odkazující na zařízení. Funkce nevrací žádnou návratovou hodnotu.

6.2 Popis programu WDTD

Funkce z knihovny `thwdtlib.h` jsem následně využil a otestoval při programování obslužného programu WDTD k ověření funkčnosti celého zařízení. Tento program umožňuje jak konfiguraci zařízení, tak kompletní obsluhu včetně nulování časovače v pravidelných intervalech. Také jako server naslouchá na zvoleném portu TCP/IP. Přes něj komunikuje s okolím a přijímá povely, které dále interpretuje watchdogu. Ke konfiguraci programu WDTD byl použit jednoduchý textový konfigurační soubor s následující strukturou, kterou je nutné přesně dodržet.

```
/dev/ttyS0
2
5060

# wtd.conf structure
#1 path to serial port
#2 clear interval (v minutach), pokud != 0 program bude automaticky nulovat
watchdog
#3 TCP/IP port number
```

První řádek udává cestu k sériovému portu, na kterém je watchdog připojen. Druhý řádek představuje délku časového intervalu, do jehož vypršení je třeba alespoň jednou odeslat povel k nulování časovače watchdogu. Jeho hodnota by měla být menší nebo rovna hodnotě `timeoutu`. Jestliže na tomto místě bude 0, program nebude automaticky provádět nulování časovače. Časovač bude nulován jen na povel přijatý pomocí socketu TCP/IP. Třetí řádek udává číslo portu TCP/IP, na kterém bude program naslouchat. Zbylé řádky konfiguračního souboru jsou programem ignorovány. Uživatel si na ně může například napsat poznámky atd.

Sockety umožňují programu komunikovat s vlastní klientskou aplikací nebo můžeme využít například program Netcat. Nyní uvedu popis všech příkazů programu WDTD včetně příkladu konfigurace watchdogu právě pomocí programu Netcat.

wdt-clr_timer

- nuluje časovač watchdogu,
- po úspěšném provedení odešle zpět ke klientovi řetězec: *wdt-clr_timer OK*

wdt-restart

- odešle zpět klientovi informaci o plánovaném restartu,
- po uplynutí 5 s je watchdogu poslán příkaz k restartování počítače.

wdt-set_enabled <TRUE/FALSE>

- S parametrem TRUE povolí watchdog, s parametrem FALSE ho naopak zakáže.
- Příkaz bude poslán watchdogu za použití instrukce set, nastavení bude uloženo do EEPROM.
- Po úspěšném provedení zašle zpět klientovi: *wdt-set_enabled TRUE/FALSE OK*

wdt-set_boot_timeout <čas v min>

- WDTD pošle watchdogu novou hodnotu časového intervalu určeného pro start systému.
- Čas lze zadat v rozsahu 0 až 255 min.
- Nová hodnota bude zapsána do interní EEPROM mikrokontroléru.
- Po úspěšném provedení zašle zpět klientovi: *wdt-set_boot_timeout <čas> OK*

wdt-set_timeout <čas v min>

- WDTD pošle watchdogu novou hodnotu timeoutu.
- Čas lze zadat v rozsahu 0 až 255 min.
- Nová hodnota bude zapsána do interní EEPROM mikrokontroléru.
- Po úspěšném provedení zašle zpět klientovi: *wdt-set_timeout <čas> OK*

wdt-set_relayswt <čas v sec>

- Provede nastavení délky sepnutí resetovacího relé.
- Čas lze zadat v rozsahu 0 až 255 vteřin.
- Po úspěšném provedení zašle zpět klientovi: *wdt-set_relayswt <čas> OK*

wdt-tenabled <TRUE / FALSE>

- Dočasně povolí (TRUE) nebo zakáže (FALSE) watchdog, nastavení nebude uloženo do EEPROM.
- Po úspěšném provedení zašle zpět klientovi: *wdt-tenabled TRUE/FALSE OK*

wdt-version

- Načte a odešle zpět klientovi informace o verzi firmware watchdogu.

V případě jakýchkoliv chyb, např. watchdog ve stanoveném čase neodpověděl, byl přijat neznámý příkaz nebo chybný parametr a podobně, je klientovi odeslána chybová hláška s popisem chyby.

K testování obslužného programu a zároveň také watchdogu byl použit program Netcat.

Připojení k WDTD

(Předpokládejme, že WDTD je spuštěn na localhostu a naslouchá na portu 5050. Chceme nastavit boot timeout 8 min, timeout 4 min, délku sepnutí relé 2 s a povolit watchdog.)

```
$ nc localhost 5050
wdt-set_boot_timeout 8
wdt-set_boot_timeout 8 OK
wdt-set_timeout 4
wdt-set_timeout 4 OK
wdt-set_relayswt 2
wdt-set_relayswt 2 OK
wdt-set_enabled TRUE
wdt-set_enabled TRUE OK
```

Závěr

Cílem této bakalářské práce bylo navrhnout a realizovat jednoduchý watchdog, který lze snadno připojit k běžnému i staršímu PC, což se mi dle mého názoru podařilo.

V průběhu návrhu zařízení jsem se setkal s několika problémy, které se ale nakonec podařilo vyřešit. V původní koncepci jsem uvažoval nad smíšenou montáží z klasických a SMD součástek v kombinaci s jednostrannou deskou plošných spojů. Ale tato volba nebyla zrovna nejtěšnější. Desky vycházely poměrně dost velké. Další problém nastal s integrovaným obvodem FT-232, který je v pouzdře SSOP 28. Jeho vývody mají malou rozteč, navíc k nim bylo nutné vést i úzké spoje, které by při ruční výrobě dělaly nemalé problémy. Z těchto důvodů bylo nakonec rozhodnuto o použití dvouvrstvého plošného spoje s prokovenými otvory a nepájivou maskou.

Výsledkem jsou dvě verze watchdogu pro připojení na sériový port a USB. Obě verze přitom zůstaly softwarově kompatibilní díky integrovanému jednočipovému převodníku od firmy FTDI.

Pro jednodušší začlenění watchdogu do uživatelské aplikace byla vytvořena knihovna funkcí pro komunikaci se zařízením. Funkce z knihovny byly poté využity při programování obslužného programu.

Díky této bakalářské práci jsem se dozvěděl spoustu nového. Vyzkoušel jsem si kompletní návrh elektronického zařízení včetně návrhu plošných spojů, programování firmware mikrokontroléru a obslužné aplikace. Celkově bych hodnotil tuto práci jako pro mě velice zajímavou.

Literatura

- [1] *Berkshire Products, Inc.* [online]. c2009 [cit. 2010-04-28]. Dostupné z WWW: <<http://www.berkprod.com/Default.aspx>>.
- [2] *QUANCOM Informationssysteme GmbH* [online]. c2001 [cit. 2010-04-28]. Dostupné z WWW: <http://www.pc-watchdog.de/quancom/html.nsf/pages/pc_watchdog.html>.
- [3] LADMAN, Jakub. *MCUserver* [online]. 16. 07. 2003 [cit. 2010-04-28]. Jednoduchý watchdog pro routery, servery a podobně. Dostupné z WWW: <<http://mcu.cz/comment.php?comment.news.363>>.
- [4] PEACOCK, Craig. *Beyond Logic* [online]. 15. 6. 2005. [cit. 2010-04-28]. Interfacing the Extended Capabilities Port. Dostupné z WWW: <<http://www.beyondlogic.org/ecp/ecp.htm>>.
- [5] OLMR, Vít. *HW.cz* [online]. 12. prosinec 2005 [cit. 2010-04-28]. Sériová linka RS-232. Dostupné z WWW: <<http://hw.cz/rs-232>>.
- [6] TIŠNOVSKÝ, Pavel. *ROOT.CZ* [online]. 27. 11. 2008 [cit. 2010-04-28]. Sériový port RS-232C. Dostupné z WWW: <<http://www.root.cz/clanky/seriovy-port-rs-232c/>>.
- [7] TIŠNOVSKÝ, Pavel. *Root.cz* [online]. 11. 12. 2008 [cit. 2010-04-28]. Komunikace pomocí sériového portu RS-232C podruhé. Dostupné z WWW: <<http://www.root.cz/clanky/komunikace-pomoci-serioveho-portu-rs-232c-podruhe>>.
- [8] SWEET, Michael. *Serial Programming Guide* [online]. 2005 [cit. 2010-05-23]. Dostupné z WWW: <<http://www.easysw.com/~mike/serial/serial.html>>.
- [9] MATOUŠEK, David. *USB prakticky: s obvody FTDI*. Praha: BEN, 2003. 272 s. ISBN 80-7300-103-9.
- [10] TIŠNOVSKÝ, Pavel. *ROOT.CZ* [online]. 22. leden 2009 [cit. 2010-05-23]. Komunikační protokol universální sériové sběrnice. Dostupné z WWW: <<http://www.root.cz/clanky/komunikacni-protokol-universalni-seriove-sbernice/>>.
- [11] *Hw.cz* [online]. HW server, s. r. o., 7. květen 2002 [cit. 2010-05-23]. USB – Universal Serial Bus – Popis rozhraní. Dostupné z WWW: <<http://hw.cz/Teorie-a-praxe/Dokumentace/ART327-USB---Universal-Serial-Bus---Popis-rozhrani.html>>.
- [12] ŘEZÁČ, Petr. *PR-Software* [online]. c2010 [cit. 2010-05-23]. Zapojení USB motherboard. Dostupné z WWW: <<http://www.pr-software.net/document/usb-motherboard/>>.

- [13] Paralelní port. In *Wikipedia: the free encyclopedia* [online]. St. Petersburg (Florida): Wikipedia Foundation, 12. 1. 2010 [cit. 2010-07-25]. Dostupné z WWW: <http://cs.wikipedia.org/wiki/Paralelní_port>.
- [14] OLMR, Vit. *Hw.cz* [online]. 12. prosinec 2005 [cit. 2010-07-25]. Paralelní port – LPT (IEEE 1284). Dostupné z WWW: <<http://hw.cz/lpt>>.
- [15] AT keyboard. In *Wikipedia: the free encyclopedia* [online]. St. Petersburg (Florida): Wikipedia Foundation, 6. 4. 2010 [cit. 2010-07-25]. Dostupné z WWW: <http://en.wikipedia.org/wiki/AT_keyboard>.
- [16] PS/2_connector. In *Wikipedia: the free encyclopedia* [online]. St. Petersburg (Florida): Wikipedia Foundation, 22. 7. 2010 [cit. 2010-07-25]. Dostupné z WWW: <http://en.wikipedia.org/wiki/PS/2_connector>.
- [17] *ATMEL* [online]. Rev. 2543K. Atmel Corporation, 2010 [cit. 2010-07-02]. ATtiny2313/V. Dostupné z WWW: <http://www.atmel.com/dyn/resources/prod_documents/doc2543.PDF>.
- [18] *MAXIM* [online]. Rev 15. Sunnyvale: Maxim Integrated Products, 1/2006 [cit. 2010-07-02]. MAX220-MAX245. Dostupné z WWW: <<http://datasheets.maxim-ic.com/en/ds/MAX220-MAX249.pdf>>.
- [19] *FTDI Chip* [online]. Version 2.06. 2010 [cit. 2010-07-02]. FT232R USB UART IC Datasheet. Dostupné z WWW: <http://www.ftdichip.com/Documents/DataSheets/DS_FT232R.pdf>.
- [20] *Equinox Technologies* [online]. Atmel Corporation, 2001 [cit. 2010-07-02]. AVR ISP Users Guide. Dostupné z WWW: <http://www.equinox-tech.com/downloads/atmel/avr%20tools/avrisp/AVRISP_User_Guide.pdf>.
- [21] LANCONELLI, Claudio. *LancOS* [online]. 2008-01-05 [cit. 2010-07-02]. SI Prog schematics. Dostupné z WWW: <<http://www.lancos.com/siprogsch.html>>.
- [22] *GM Electronic* [online]. GM Electronic, spol. s r. o., c2010 [cit. 2010-07-25]. Dostupné z WWW: <<http://www.gme.cz>>.
- [23] *PragoBoard* [online]. PragoBoard, s. r. o., c2009 [cit. 2010-07-25]. Dostupné z WWW: <<http://www.pragoboard.cz>>.