VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

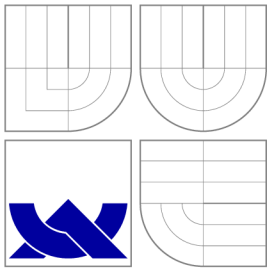
ČTEČKA PAMĚŤOVÝCH KARET

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

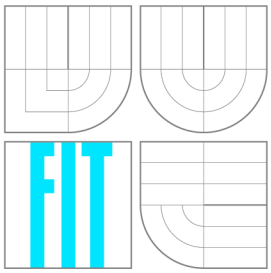
AUTOR PRÁCE
AUTHOR

MICHAL RUMPLÍK

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

ČTEČKA PAMĚŤOVÝCH KARET

MEMORY-CARD READER

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — MICHAL RUMPLÍK
AUTHOR

VEDOUCÍ PRÁCE — Ing. JOSEF STRNADEL, Ph.D.
SUPERVISOR

BRNO 2008

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačových systémů Akademický rok 2007/2008

Zadání bakalářské práce

Řešitel: **Rumplík Michal**
Obor: Informační technologie
Téma: **Čtečka paměťových karet**
Kategorie: Vestavěné systémy

Pokyny:

1. Seznamte se s existujícími typy paměťových karet (SD, SM, MMC, CF, NAND flash, atd.), s rozhraními těchto karet a s principy komunikace s těmito kartami. Získané znalosti pečlivě zdokumentujte.
2. Pro podmnožinu typů karet zvolenou po dohodě s vedoucím, navrhněte blokové schéma čtečky.
3. Čtečku realizujte a prakticky ověřte její funkčnost.
4. Shrňte dosažené výsledky a navrhněte možná rozšíření Vašeho projektu.

Literatura:

- *Konektory-BENEL, s.r.o.* [online]. c2006. http://www.konektor.cz/.
- *MMC Card Specification*. [online]. c2004. http://www.kingmaxdigi.com/product/MMC.pdf.
- *SanDisk Multimedia Card*. [online]. c2004. http://www.sandisk.com/Assets/File/OEM/Manuals/manual-rs-mmcv1.0.pdf
- *SanDisk Secure Digital Card*. [online]. c2003. http://www.cs.ucr.edu/~amitra/sdcard/ProdManualSDCardv1.9.pdf
- *SD Card Association*. [on-line]. c2007. http://www.sdcard.org/.
- *SD Card Module Evaluation Kit*. [online]. 2007. http://www.cypress.com/portal/server.pt?space=CommunityPage&control=SetCommunity&CommunityID=285&PageID=552&shortlink=

Při obhajobě semestrální části projektu je požadováno:

- Body 1 a 2.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

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

Vedoucí: **Strnadel Josef, Ing., Ph.D.**, UPSY FIT VUT
Datum zadání: 1. listopadu 2007
Datum odevzdání: 14. května 2008

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačových systémů a sítí
612 66 Brno, Božetěchova 2

doc. Ing. Zdeněk Kotásek, CSc.
vedoucí ústavu

Licenční smlouva

Licenční smlouva je uložena v archivu Fakulty informačních technologií Vysokého učení technického v Brně.

Abstrakt

Práce se zabývá problematikou paměťových karet SD a jejich principem činnosti. Cílem práce bylo navrhnout a sestrojit čtečku paměťových karet. Toto zařízení umí ukládat a číst ASCII znaky. Hardware byl navrhnut pro komunikaci přes sériové rozhraní a je řízen mikrokontrolérem MC9S08QG8.

Klíčová slova

SecureDigital, HCS08, paměťová karta, RS232, hardware, SPI

Abstract

This work deals with problems of SD memory card reader and its working principles. I have focused in my bachelor's thesis on proposition and construction of memory-card reader. This memory-card reader is able to store and read ASCII signs. The hardware was proposed for serial interface communication and it is geared by microcontroller MC9S08QG8.

Keywords

SecureDigital, HCS08, memory card, RS232, hardware, SPI

Citace

Michal Rumplík: Čtečka paměťových karet, bakalářská práce, Brno, FIT VUT v Brně, 2008

Čtečka paměťových karet

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Josefa Strnadela, Ph.D.
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Michal Rumplík
12. května 2008

Poděkování

Rád bych poděkoval vedoucímu mé bakalářské práce Ing. Josefu Strnadelovi, Ph.D., za konzultace a odborné rady.

© Michal Rumplík, 2008.
Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod **3**

2 Paměťové karty **4**
2.1 SD karty . . . 4
2.1.1 Základní vlastnosti . . . 4
2.1.2 Specifikace SD . . . 5
2.1.3 Popis rozhraní . . . 6
2.1.4 SD card režim . . . 8
2.1.5 SPI režim . . . 9
2.2 MMC . . . 12
2.3 CompactFlash . . . 12

3 HCS08 **13**
3.1 Základní vlastnosti . . . 13
3.2 Přiřazení pinů . . . 13
3.3 Paměťový model . . . 14
3.4 ICS - Vnitřní zdroj hodin . . . 15
3.5 SCI - Sériové komunikační rozhraní . . . 15
3.5.1 Registry pro SCI modul . . . 15
3.5.2 Rámec . . . 15
3.5.3 Rozhraní RS232 . . . 16
3.5.4 Převodník úrovní MAX3232 . . . 16
3.6 SPI - Synchronní sériové rozhraní . . . 16
3.6.1 Vodiče ve SPI modulu . . . 16
3.6.2 Registry pro SPI modul . . . 17

4 Návrh a implementace **19**
4.1 Návrh hardwaru . . . 19
4.1.1 Schéma . . . 19
4.1.2 Popis součástek . . . 19
4.1.3 DPS . . . 21
4.2 Firmware . . . 21
4.2.1 Popis FW . . . 21
4.2.2 Programování mikrokontroléru . . . 24

5 Testování **25**

6 Závěr **27**

Seznam příloh 28

Příloha A - Obsah CD 30

Příloha B - DPS 31

Příloha C - Fotodokumentace 32

Příloha D - Práce v hyperterminálu 34

Kapitola 1

Úvod

S rozvojem výpočetní techniky se ruku v ruce zvyšuje i náročnost na ukládání čím dál větších objemů dat. Z tohoto důvodu se s rostoucí kapacitou pevných disků vyvíjí i paměťové karty, které mají v dnešní době velké uplatnění. Aby se data uložená na tomto typu paměťového média mohla zobrazovat na různých zařízeních, je potřeba na tyto účely čtečka paměťových karet.

Tato práce se právě takovou jednoduchou implementací čtečky zabývá. Seznámí čtenáře s problematikou SD karet a jejich komunikace s hostitelským zařízením.

V druhé kapitole se čtenář dozví informace o různých typech paměťových karet, zvláště pak o SecureDigital (SD) a principu jejich komunikace přes rozhraní SPI.

Třetí kapitola popisuje typ použitého mikrokontroléru FreeScale (dříve Motorola) řady HCS08 a jeho funkce, potřebné pro činnost čtečky, jako jsou komunikační rozhraní, jejich nastavení a oscilátor hodin.

Čtvrtá a pátá kapitola popisuje výrobu desky plošných spojů (DPS), její osazení a zvláště pak vývoj firmwaru, který zajišťuje veškerou komunikaci mezi uživatelem a čtečkou paměťových karet.

Na závěr jsou uvedená možná rozšíření zařízení a celkové zhodnocení práce.

Kapitola 2

Paměťové karty

Paměťová karta je přenosné elektronické zařízení na ukládání dat. Používá se v digitálních fotoaparátech, PDA, laptopech, mobilních telefonech, přehrávačích a jiných elektronických zařízeních. Obvykle je založena na paměti typu flash. Flash je nevolatilní paměť typu RAM (s náhodným přístupem), elektricky programovatelná. Je vnitřně organizována po blocích a na rozdíl od pamětí typu EEPROM lze programovat každý blok samostatně (obsah ostatních bloků je zachován) [6].

2.1 SD karty

SD paměťové karty jsou v dnešní době jedny z nejpoužívanějších. Jejich vývoj jde s dobou a kapacita se neustále zvyšuje. Udávaná životnost je 100 000 zápisových/mazacích cyklů a spolehlivost: MTBF[1] > 1 000 000 hodin.

Existují tyto verze SD karet:

- SD
- miniSD
- microSD
- SDHC

2.1.1 Základní vlastnosti

Z produktového manuálu [5] lze vyčíst tyto obecné vlastnosti:

- nízké napájecí napětí
- frekvence 0 – 25 MHz
- podpora SPI módu
- korekce chyb
- mechanické zabezpečení proti zápisu
- Copyright ochrana zvukových nahrávek podle standardu SDMI

[1] MTBF = Mean Time Between Failures

2.1.1.1 Režim spánku

Automatický režim spánku je unikátní vlastnost těchto karet, do kterého přechází karta samostatně do několika milisekund, pokud jsou vykonány všechny příkazy. Pokud hostitelské zařízení (dále jen hostitel) pošle další příkaz, karta přejde automaticky opět do pohotovostního režimu.

2.1.1.2 Vkládání "za chodu"

Konektor na SD kartě podporuje vkládání za chodu, tzv. hot insertion. Avšak tato vlastnost musí být podporována hostitelským zařízením. Kvůli tomu jsou piny pro napájení delší než všechny ostatní.

2.1.2 Specifikace SD

2.1.2.1 Napájecí napětí

Rozsah napětí, ve kterém karta pracuje, je 2,7 – 3,6 V. Toto napětí je potřeba pro většinu příkazů a pro přístup do paměti. Avšak pro inicializaci karty po připojení napájení stačí už pouze 2 V (příkazy CMD0, CMD15, CMD55, ACMD41).

	hodnota
Čtení	max. 65mA
Zápis	max. 75mA
Režim spánku	max. 250 μA

Tabulka 2.1: Spotřeba při 25°C

2.1.2.2 Pracovní teplota a vlhkost

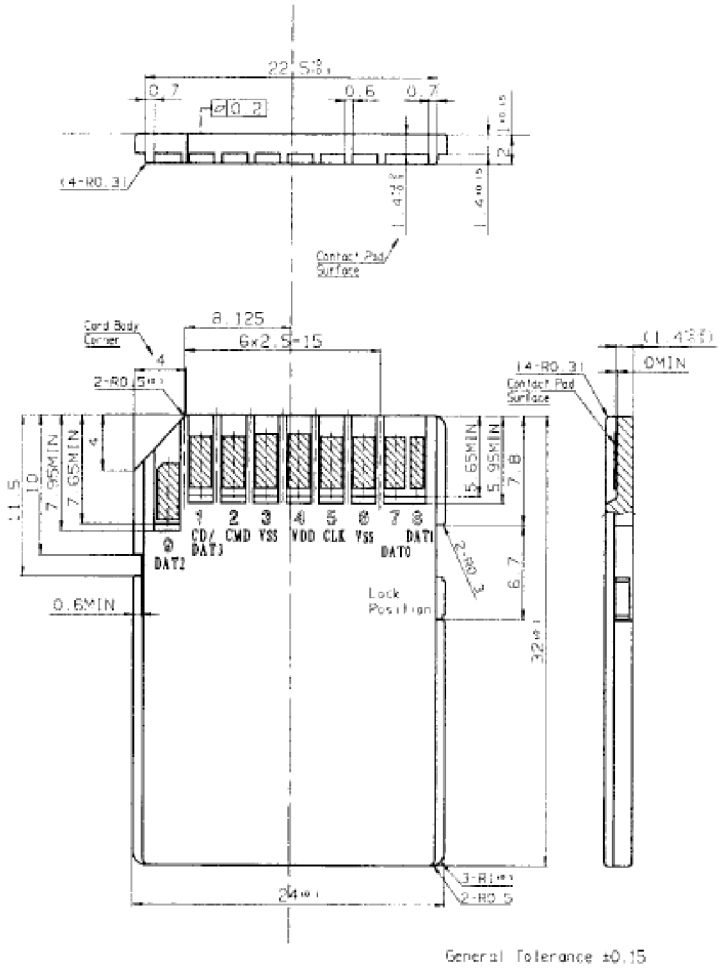
Teplota	operační klidová	-25° – 85° C -40° – 85° C
Vlhkost	operační i klidová	25 % – 85 %

Tabulka 2.2: Pracovní teplota a vlhkost

2.1.2.3 Fyzické parametry

Váha	2,0g max.
Výška	32mm ± 0,1mm
Šířka	24mm ± 0,1mm
Tloušťka	2,1mm ± 0,15mm

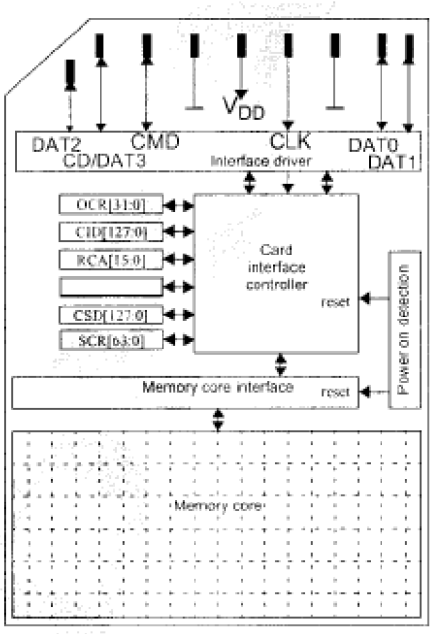
Tabulka 2.3: Fyzické vlastnosti



Obrázek 2.1: Rozměry SD karet

2.1.3 Popis rozhraní

SD paměťové karty mají 9 pinů na jedné straně, viz. 2.2. Jejich zapojení se liší podle režimu, ve kterém SD karta pracuje.



Obrázek 2.2: Architektura SD karty

2.1.3.1 SD card protokol

V SD card protokolu je 6 komunikačních a 3 napájecí vodiče. Vodiče Dat0 – Dat3 a CMD komunikují obousměrně. Vodič CLK pouze ve směru hostitel – SD karta. Zapojení všech pinů je vidět v tabulce 2.4.

Č. pinu	Jméno	Typ[2]	Popis
1	CD/Dat3	I/O	detekce karty / data - bit3
2	CMD	I/O	příkaz/odpověď
3	Vss	S	zem napájení
4	Vdd	S	napájení
5	CLK	I	hodinový signál
6	Vss2	S	zem napájení
7	Dat0	I/O	data - bit0
8	Dat1	I/O	data - bit1
9	Dat2	I/O	data - bit2

Tabulka 2.4: Přiřazení pinů v SD card režimu

2.1.3.2 SPI protokol

Oproti SD card protokolu využívá SPI protokol jen 7 pinů. Přenášení dat, příkazů a odpovědí zde probíhá pouze po 2 vodičích, a to DataIn a DataOut. Přesné zapojení lze vidět v tabulce 2.5.

[2] S = napájení; I = input; O = output

Č. pinu	Jméno	Typ	Popis
1	CS	I	výběr karty
2	DataIn	I	příkazy/data hostitel → SD karta
3	Vss	S	zem napájení
4	Vdd	S	napájení
5	CLK	I	hodinový signál
6	Vss2	S	zem napájení
7	DataOut	O	status/data SD karta → hostitel
8	RSV	I	Reserved
9	RSV	I	Reserved

Tabulka 2.5: Přiřazení pinů ve SPI režimu

2.1.3.3 Registry

Každá SD karta obsahuje 7 registrů. Jsou to:

- OCR
- CID
- CSD
- SCR
- RCA
- card status
- SD status

OCR (Operation Condition Register) je 32-bitový registr operačních podmínek, ve kterém jsou uloženy napěťové úrovně.

CID (Card Identification Register) je 16B dlouhý registr. Uchovává unikátní identifikační číslo karty, které je naprogramováno během výroby a je neměnné.

CSD (Card Specific Register) obsahuje konfigurační informace pro přístup k datům na kartě.

Bližší popis všech registrů lze najít v datasheetu [5]

2.1.4 SD card režim

Tento režim je pro SD karty základní. Po připojení napájení se karta nachází právě v tomto režimu. Pro komunikaci využívá všech 6-ti datových vodičů a je založena na příkazových a datových tocích bitů, které jsou ohraničeny start-bitem na začátku a stop-bitem na konci.

Dále jsem se tímto protokolem nezabýval, protože ke komunikaci využívam SPI rozhraní. Více se lze dočíst v datasheetu [5].

2.1.5 SPI režim

SPI režim je druhý komunikační protokol pro tyto karty. Tento mód je podmnožina protokolu SD card, upravená ke komunikaci přes rozhraní SPI (Serial Peripheral Interface), které je v dnešní době obsaženo v mnoha mikrokontrolérech. Narozdíl od SD card protokolu je komunikace u SPI byte-ově orientovaná – zarovnaná na 1B. Skládá se z příkazů (commands), odpovědí (response) na každý příkaz a z bloků dat (data blocks). Veškerá komunikace mezi kartou a hostitelským zařízením (čtečka) je řízeno hostitelem (master).

2.1.5.1 Příkazy protokolu SPI

Příkaz	Argument	Odpověď	Popis
CMD0	–	R1	reset SD karty
CMD1	–	R1	aktivace inicializačního procesu
CMD16	[31:0] délka bloku	R1	nastavení délky bloku dat v bytech
CMD17	[31:0] adresa dat	R1	čtení bloku dat nastavené délky
CMD24	[31:0] adresa dat	R1	zápis bloku dat nastavené délky

Tabulka 2.6: Použité příkazy protokolu SPI

2.1.5.2 Formát příkazů

Všechny příkazy protokolu SPI mají délku 6B a přenášen je jako první vždy MSB[3]. Popis jednotlivých bytů příkazu je popsán na obrázku 2.3.

Byte 1				Bytes 2—5		Byte 6	
7	6	5	0	31	0	7	0
0	1	Command		Command Argument		CRC	1

Obrázek 2.3: Struktura příkazů ve SPI režimu

Kontrolní součet CRC je ve SPI režimu standardně vypnut. Toto nastavení se dá změnit příkazem CMD59. Platný CRC součet je vyžadován pouze při provádění příkazu CMD0 při inicializaci karty. Z tohoto důvodu je však uvedený přímo ve specifikaci karty.

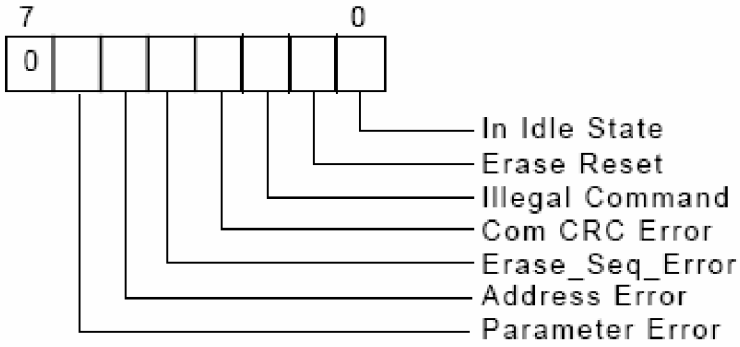
2.1.5.3 Odpovědi

Na každý příkaz odpovídá karta jedním ze 3 typů odpovědi:

- R1 response
- R2 response
- R3 response

[3]MSB – Most Significant Bit

Odpověď typu R1:
Tento typ odpovědi je posílán na většinu příkazů. Je jeden byte dlouhý a význam jednotlivých bitů v něm je uveden na obrázku 2.4. Chybu indikuje stav log. „1“ u příslušného bitu.



Obrázek 2.4: Byte odpovědi typu R1

Odpověď typu R2:
Odpověď typu R2 vznikne přidáním dalšího bytu k odpovědi R1, ve kterém jednotlivé bity označují další možné chyby. Tato odpověď se posílá pouze po zaslání příkazu CMD13.

Odpověď typu R3:
Vznikne přidáním 4 bytů k odpovědi typu R1. V těchto přidaných bytech je přenesen obsah registru OCR.

Bližší popis odpovědí R2, R3 lze najít v datasheetu SD karty.

2.1.5.4 Inicializace karty

Pro práci s daty na kartě je nutné nejdřív provést inicializaci karty. Ta se skládá z přesně daných kroků, které se musí dodržet:

- provést alespoň 74 hodinových cyklů
- přívést aktivní úroveň - log. „0“, na signál CS
- zaslat příkaz CMD0 kartě
- karta musí odpovědět response tokenem typu R1 s hodnotou 0x01
- zaslat příkaz CMD1
- karta nyní musí odpovědět hodnotou 0x00, pokud neodpoví, opakuje se zaslání příkazu CMD1 do té doby, než karta odpoví požadovanou hodnotou

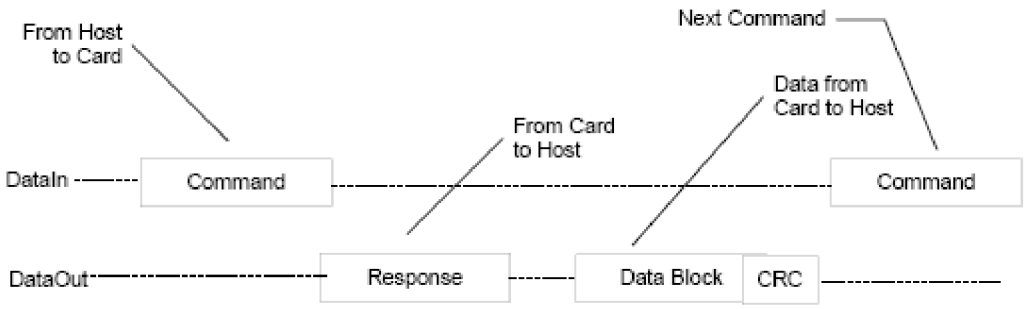
Po provedení těchto kroků lze začít přenášet data.

2.1.5.5 Čtení dat

Ve SPI módu existují dva typy čtení dat:

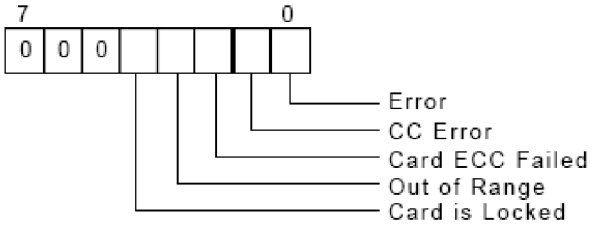
- single block read (čtení jednoho bloku)
- multiple block read (čtení více bloků)

Pro tyto operace slouží příkazy CMD17, resp. CMD18. Po odeslání jednoho z těchto příkazů hostitel musí čekat na response token typu R1, který je následován data tokenem, obsahující start byte, data dané délky a 2 byty s CRC součtem. Tato posloupnost příkazů/odpovědí je vidět na obrázku 2.5.



Obrázek 2.5: Single block read

Pokud při čtení dat dojde k nějaké chybě, je zaslán Data error token, který přesně specifikuje k jaké chybě došlo, viz 2.6.



Obrázek 2.6: Formát data error tokenu

2.1.5.6 Zápis dat

Stejně jako pro čtení, tak i pro zápis existují dva druhy:

- single block write (zápis jednoho bloku)
- multiple block write (zápis více bloků)

Celý proces zápisu dat je obdobný jako proces čtení. Nejdříve je zaslán příkaz CMD24 (CMD25). Po obdržení odpovědi typu R1 čeká karta na data token, který je potvrzen odpovědí (data response) délky 1B, kde podle hodnoty bitů lze poznat, jestli byla data přijata

nebo nastala chyba.

7							0
x	x	x	0	A	B	C	1

Tabulka 2.7: Formát data response

Pokud v bitech ABC přijmeme:

- 010 – data byla zapsána v pořádku
- 101 – data odmítnuta kvůli CRC chybě
- 110 – data odmítnuta kvůli chybě zápisu

2.2 MMC

MultiMedia Card (MMC) jsou velmi podobné kartám SD a oba typy by měly být vzájemně kompatibilní. Ve fyzických parametrech se liší pouze tloušťkou a počtem pinů, kterých je u MMC karet pouze 7. Jediný rozdíl v komunikaci nastává při inicializaci karty, kdy frekvence hodinového signálu nesmí překročit 400kHz. Po dokončení inicializace lze tuto rychlost opět zvýšit na vyšší hodnotu, kterou paměťová karta zvládne. Další informace lze najít v datasheetu [4] těchto karet.

2.3 CompactFlash

CompactFlash (CF) karty jsou od předešlých dvou typů naprosto odlišné. Liší se hlavně v principu komunikace, ke které využívají 50-ti pinový konektor. Je to kvůli rozhrní ATA, které tyto karty mají. Přenos dat tedy probíhá paralelně.

Kapitola 3

HCS08

HCS08 je rodina 8-mi bitových mikrokontrolérů od firmy Freescale s jádrem HCS08. Tato rodina se vyznačuje hlavně nízkým napájením a spotřebou, širokým výběrem modulů, velikostí pamětí a typy pouzder.

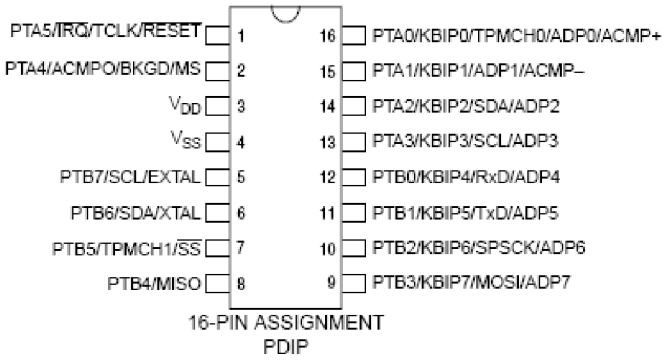
Dále budu popisovat mikrokontrolér MC9S08QG8 z této rodiny, který jsem si k implementaci čtečky vybral.

3.1 Základní vlastnosti

- CPU 20MHz
- FLASH paměť 512B, RAM 8KByte
- interní oscilátor
- komunikační moduly SCI, SPI, IIC
- časovač, A/D převodník
- analogový komparátor
- watchdog
- napájení 1,8 – 3,6 V

3.2 Přiřazení pinů

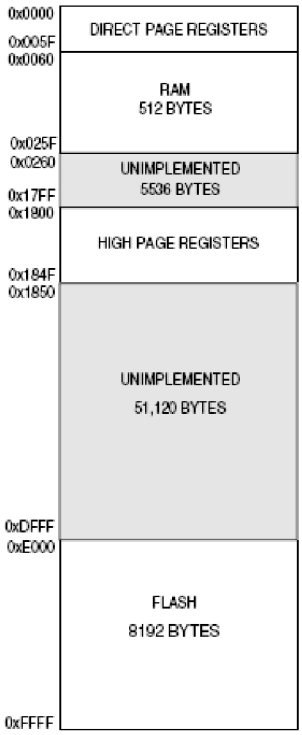
Každý pin má přiřazeno více funkcí, viz. obrázek 3.1. Jakou funkci bude pin zastávat, záleží na použití daného modulu.



Obrázek 3.1: Pouzdro mikrokontroléru a přiřazení pinů

3.3 Paměťový model

Paměť mikrokontroléru je mapována do 64kB velkého adresového prostoru. Skládá se z RAM, FLASH paměti pro program, I/O a kontrolních registrů. Její organizace je vidět na obrázku 3.2.



Obrázek 3.2: Organizace paměti mikrokontroléru

3.4 ICS - Vnitřní zdroj hodin

Vybraný mikrokontrolér obsahuje ICS modul, který vytváří referenční hodinový signál o frekvenci 16MHz. Mezi hodinový signál bus clk, který je potřeba pro komunikaci, a referenční hodinový signál je postavena dělička dvěma. Proto je nejvyšší možná výsledná frekvence bus clk 8MHz.

3.5 SCI - Sériové komunikační rozhraní

Toto rozhraní je využito ke komunikaci s PC přes rozhraní RS232. Jedná se o asynchronní přenos, u kterého nám stačí pro přenos dat pouze dva vodiče – Rx a Tx.

3.5.1 Registry pro SCI modul

Komunikační rozhraní SCI obsahuje osm 8-mi bitových registrů k nastavení baud rate, výběru voleb, čtení statusů a posílání/příjem dat. Nejdůležitější jsou:

- SCIBD
- SCIC1
- SCIC2
- SCIS1
- SCID

Registr SCIBD
Tento párový registr (SCIBDH a SCIBDL) slouží jako předdělička k nastavení znakové rychlosti – Baud rate. Výsledná baud rate se vypočítá podle vztahu: $baudrate = \frac{BUSCLK}{16*BR}$, kde: BUSCLK = hodinový signál, BR = nastavení registru SCIBD.

Registry SCIC1 a SCIC2
Tyto registry slouží k nastavení vlastností SCI rozhraní, jako jsou počet datových bitů, parita, povolení příjmu/vysílání dat a dalších, které jsou detailně popsány v datasheetu [2].

Registr SCIS1
Registr určený pouze ke čtení. Jednotlivé bity se nastavují na úroveň log. „1“, pokud dojde k určité události. Na základě testování těchto bitů lze pak na tyto události reagovat.

Registr SCID
Tento datový registr ve skutečnosti obsahuje 2 samostatné registry. Jeden slouží pouze pro čtení přijatých dat a druhý k zápisu dat pro odeslání.

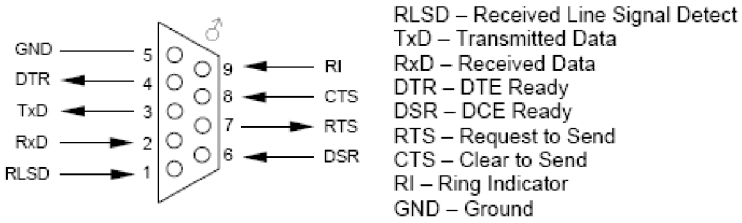
3.5.2 Rámec

Takto se nazývá celá soustava, obsahující start-bit, datové bity, příp. paritní bit a stop-bit, ve které se přenáší jedno datové slovo.

3.5.3 Rozhraní RS232

RS232 – nazývané také jako sériový port – je rozhraní pro komunikaci mezi dvěma zařízeními. V dnešní době je však na ústupu a je postupně nahrazováno portem USB (Universal Serial Bus).

Při přenosu dat si zde vystačíme pouze se dvěma vodiči. Jsou to Tx – pro vysílání a Rx – pro příjem dat. Dále je nutné propojení signálových zemí obou zařízení. Zapojení konektoru je znázorněno na obrázku 3.3 (obrázek převzat z [9]).



Obrázek 3.3: Konektor RS232 (Cannon 9)

3.5.4 Převodník úrovní MAX3232

Protože rozhraní RS232 používá napěťové úrovně ±12 V a mikrokonrolér MC9S08QG8 pouze 3 V, je potřeba převodník napětových úrovní, který tento problém vyřeší. Integrovaný obvod MAX3232 převádí napěťovou úroveň z rozhraní RS232 na úroveň podle dodaného napájecího napětí v rozmezí 3,0 – 5,5 V. Pro převod opačným směrem obsahuje IO tzv. nábojovou pumpu, která je tvořena 4-mi kondenzátory.

Přesný popis a zapojení uvedeného integrovaného obvodu lze najít v [3].

3.6 SPI - Synchronní sériové rozhraní

Synchronní sériové rozhraní SPI slouží k připojení periferií. V jednom okamžiku můžou komunikovat vždy pouze 2 komponenty. Přenos dat je plně duplexní (full-duplex), tzn. probíhá zároveň oběma směry. Každé zařízení připojené na toto rozhraní musí být v jednom ze dvou režimů – Master nebo Slave. V režimu Master by však mělo být pouze jen jedno z nich, které generuje hodinový signál a vybírá druhé zařízení v režimu Slave ke komunikaci.

3.6.1 Vodiče ve SPI modulu

Tento modul obsahuje 4 vodiče – 2 pro přenos dat, 1 pro hodinový signál a 1 pro výběr zařízení ke komunikaci. U všech je směr komunikace závislý na režimu, ve kterém se pracuje. Princip činnosti je vidět na obrázku 3.4 (obrázek převzat z [9]).

3.6.1.1 Vodiče MISO, MOSI

Vodiče MISO (Master In, Slave Out) a MOSI (Master Out, Slave In) slouží pro přenos dat. Jak už název napovídá, MISO posílá data od Slave zařízení do Master a MOSI směrem od Master do Slave.

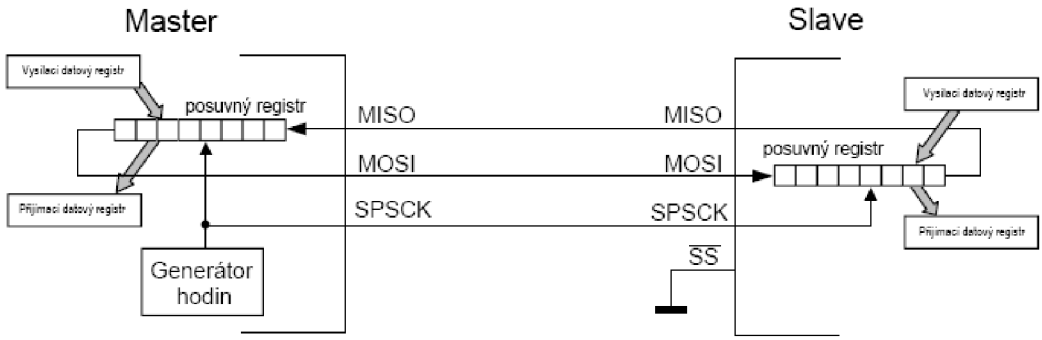
3.6.1.2 Vodič SPSCK

Tento vodič šíří hodinový signál, který je v režimu Master nastaven jako výstupní. Všechna zařízení typu Slave naopak tento signál přijímají a pin SPSCK je tedy u nich nastaven jako vstupní.

Výběr určité frekvence se provádí správným nastavením hodnot registru mikrokontroléru. Nejvyšší možná dosažitelná frekvence je však polovina BUS CLK, tj. 4Mhz.

3.6.1.3 Vodič !SS

Tento pin může nastavovat pouze Master zařízení, které přivedením úrovně log. „0“ na tento pin vybírá práve jedno zařízení typu Slave ke komunikaci.



Obrázek 3.4: Komunikace přes SPI rozhraní

3.6.2 Registry pro SPI modul

V rozhraní SPI lze pracovat s 5-ti registry pro různá nastavení, čtení příznaků a dat:

- SPIC1
- SPIC2
- SPIBR
- SPIS
- SPID

Registry SPIC1 a SPIC2
Registry slouží k povolení a nastavení SPI rozhraní. Nejdůležitější bity SPIC1 jsou 6, 4 a 1, kterými se aktivuje SPI, vybere Master nebo Slave mód a posledním se aktivuje vybrané Slave zařízení. Druhý z registrů využívá pouze 4 bity, ale pro účely čtečky paměťových karet už nejsou tak důležité.

Detailnější popis lze najít v datasheetu [2]

Registr SPIBR
SPIBR registr slouží k nastavení rychlosti komunikace rozhraní SPI. Různými hodnotami lze nastavit děličku a předděličku, které se tímto registrem nastavují a určují tak výslednou

rychlost.

Registry SPIS a SPID
Čtení a zápis do datového registru SPID je podmíněno stavovým registrem SPIS, který je pouze pro čtení. V tomto registru se nastavují příznaky po dokončení přenosu dat a až tehdy lze do datového registru zapisovat/číst.

Kapitola 4

Návrh a implementace

Tato kapitola popisuje navržené schéma čtečky a návrh desky plošných spojů (DPS), která byla vytvářena v programu Eagle verze 4.16r2. Tento program je volně ke stažení např. z [7]. Dále kapitola obsahuje vývoj firmwaru mikrokonroléru a jím ovládaných periferií.

4.1 Návrh hardwaru

4.1.1 Schéma

Na obrázku 4.1 je znázorněno navržené schéma čtečky paměťových karet, které bylo po vytvoření DPS odzkoušeno a je funkční. Při návrhu jsem vycházel z doporučených zapojení jednotlivých součástek, které jsou uvedeny v datasheetech [5, 2, 3].

4.1.2 Popis součástek

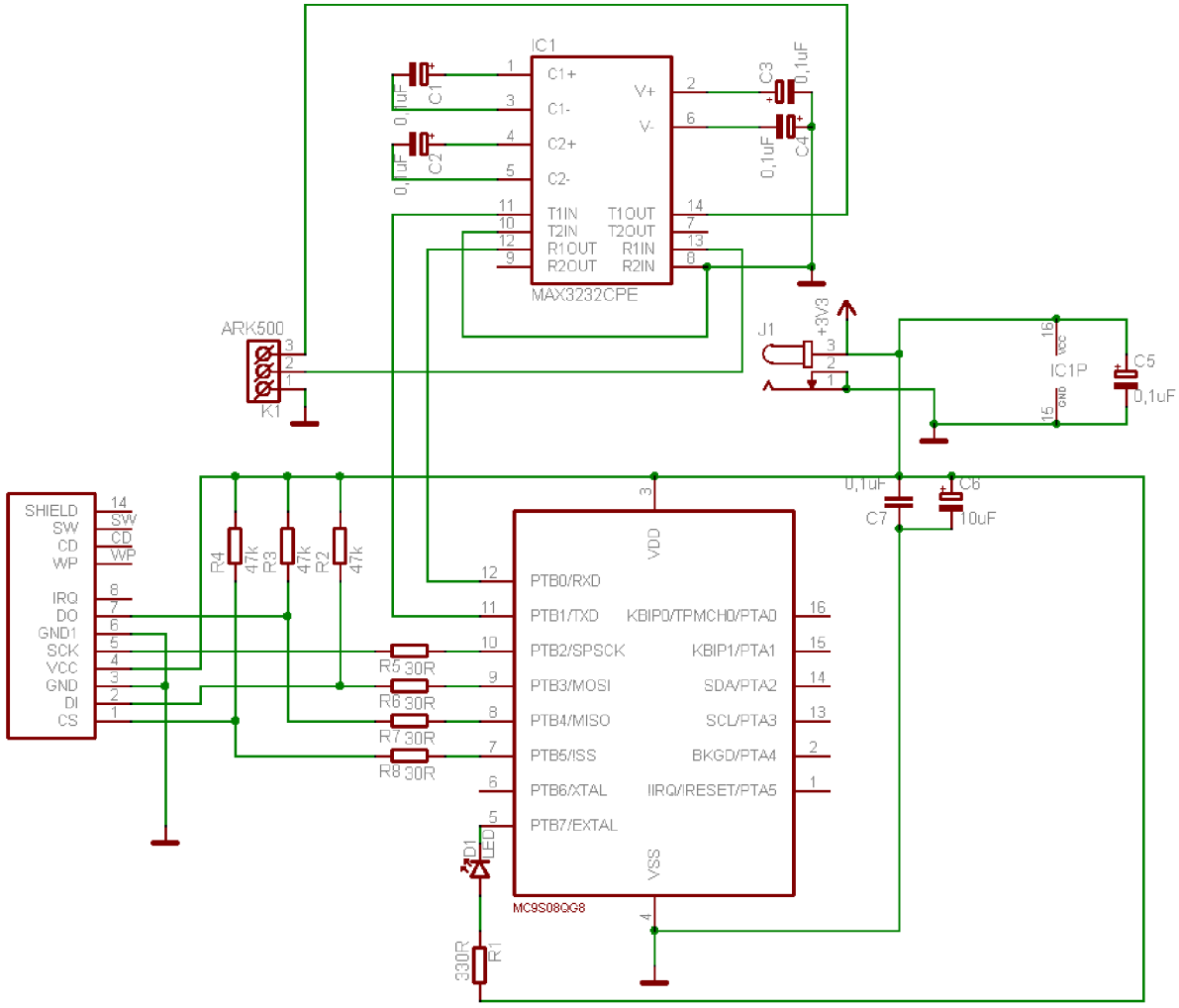
Součástka	R1	R2 – R4	R5 – R8	C1 – C5	C6	C7	D1
Hodnota	330Ω	$47k\Omega$	30Ω	$0,1\mu F$	$10\mu F$	$0,1\mu F$	LED dioda

Tabulka 4.1: Hodnoty součástek

Součástka	**Popis**
MC9S08QG8	mikrokontrolér
MAX3232	převodník úrovní
SD socket	patice na SD kartu
K1 – ARK500	svorkovnice pro 3 vodiče
J1 – K375	napájecí konektor

Tabulka 4.2: Popis ostatních součástek

Rezistor R1 zde má funkci ochranného prvku LED diody, aby nedošlo k jejímu spálení z důvodu vysokého proudu. Další rezistory R2 – R4 slouží k odstranění špiček napájecího napětí na vodičích SD karty a R5 – R8 jsou zde z důvodu lepších přenosových podmínek. Čtveřice kondenzátorů C1 – C4 tvoří tzv. nábojovou pumpu, která je využita při převodu z napěťové úrovně 3 V na úroveň rozhraní RS232. Ostatní kondenzátory zde plní ochrannou



Obrázek 4.1: Kompletní schéma čtečky

funkci napájení integrovaných obvodů. LED dioda červené barvy je na DPS umístěna pouze z vlastního důvodu při ladění firmwaru. V konečném důsledku může sloužit např. jako informační prvek, který bude signalizovat jestli čtečka pracuje nebo je v pohotovostním režimu.

4.1.3 DPS

Desku plošných spojů a rozložení součástek na ní lze najít v příloze B.

4.2 Firmware

Pro vývoj firmwaru jsem použil software přímo od výrobce mikrokontroléru, firmy Freescale, CodeWarrior v5.1. Tento software poskytuje velké možnosti nejen při programování, ale i při ladění a nahrání FW přímo do chipu. Lze jej bezplatně stáhnout po zaregistrování ze stránek výrobce [1].

Firmware je obsažen pouze v jednom souboru `Reader.c`.

4.2.1 Popis FW

Soubor `Reader.c` obsahuje několik funkcí pro nastavení mikrokontroléru, komunikaci s kartou a PC. Hlavní část funkce `Main` tvoří nekonečná smyčka pro komunikaci s uživatelem, který si může vybírat jednu ze dvou možností menu:

- Zápis na SD kartu
- Čtení z SD karty

Jako první musí FW provést správné nastavení rozhraní SCI, SPI a ICS mikrokontroléru. Toto zajištují funkce `void InitSCI(void); void InitSPI(void); void SetICS(void);` správným nastavením registrů. Detailnější popis byl uveden v kapitole 3.6.2

Pro rozhraní SCI je nejdůležitější nastavení přenosové rychosti baud rate. To se provede zápisem hodnoty 0x32 do registru SCIBD. Výsledná rychlost tedy bude 9600 baudů. Dále nastavením registru SCIC1 na hodnotu 0x00 nastavíme parametry přenosu na 8 datových bitů bez kontroly parity a registru SCIC2 na hodnotu 0x2C povolíme rozhraní SCI. Toto nastavení jsem si sám zvolil. V případě jiné konfigurace by bylo nastavení registrů odlišné.

Další funkce, InitSPI, provede povolení rozhraní SPI, nastavení módu na master, výběr polarity a hrany hodinového signálu a nastavení baud rate rychlosti tohoto rozhraní. Provede se tak nastavením registrů `SPIC1, SPIC2, SPIBR`.

Funkce InitSPI

```
void InitSPI(void){
  SPIC1 = 0x50;         /* nastavení registru SPIC1 */
  SPIC2 = 0x10;         /* nastavení registru SPIC2 */
  SPIBR = 0x04;         /* nastavení registru SPIBR */
}
```

Pokud se nepoužívá externí krystal pro řízení referenčního hodinového signálu, použije se modul ICS, který nastavením registrů `ICSC1=0x06; ICSC2=0x00; ICSSC=0x01; ICSTRM=0xA1;` povolí a nastaví referenční signál na hodnotu 16 MHz. Toto provádí funkce `void SetICS(void)`.

Po provedení těchto nastavení může začít komunikace mikrokontroléru s periferiemi.

Jako další musíme provést inicializaci paměťové karty. Přesný postup je popsán v kapitole 2.1.5.4. Funkce `void InitCard(void);` tento postup přesně kopíruje a navíc nastaví délku bloku na 512B zasláním příkazu CMD16 s parametrem 512. Po celou dobu inicializace karty je snížena frekvence hodinového signálu na 250kHz z důvodu kompatibility s paměťovými kartami MMC.

Aby se dalo s kartou komunikovat, vytvořil jsem funkce `void sendByte(byte s)` a `byte recByte(void)`.

Funkce sendByte

```
void sendByte(byte s){
  char temp;            /* pomocná proměnná */
  while(!SPIS_SPTEF){}  /* čekání, pokud se něco odesílá */
  SPID = s;             /* odeslání bajtu */
  while (!SPIS_SPRF){}  /* čekání na dokončení přenosu */
  temp = SPID;          /* přečtení přijatého bajtu */
}
```

Funkce `recByte()` je obdobná. Pro přijetí požadovaného bytu se odesílá tzv. „dummy byte“ s jakoukoli hodnotou.

Další funkce pro práci s kartou jsou `void writeSD(void)` a `void readSD(void)`.

Funkce `writeSD()` jako první uloží řetězec zadaný uživatelem do pole typu char velikosti 256 bytů. Dále funkce pošle příkaz CMD24 pro zápis na kartu, který následuje blok dat s řetězcem.

Funkce pro čtení, `readSD()`, posílá příkaz CMD17 pro čtení jednoho bloku dat. Karta na tento příkaz odpovídá zasláním posloupnosti 512 bytů s uloženými daty. Protože uživatel mohl uložit maximálně 256 bytů, jendotlivé byty se ukládájí do pole pouze po ukončovací znak řetězce `\0`. Ostatní přijaté byty se „zahazují“. Nakonec se řetězec vypíše pomocí funkce `SendMsg(char msg[])` uživateli na obrazovku.

Část kódu pro ukládání řetězce z karty:

```
for(x=0;x<512;x++){                 /* přijímá se 512B */
  if((x < 256) && (response != '\0')){  /* pokud je volno v paměti a
                                           není konec */
    response = recByte();           /* uložení bytu do pomocné proměnné */
    buffer[x]=response;             /* skládání řetězce */
  }else
    recByte();                      /* zahození nepotřebných bytů */
}
```

Pro interakci s uživatelem obsahuje firmware funkce `char RecChar(void)`, `void SendChar(char s)` a `void SendMsg(char msg[])`, které umí přijmout zadaný znak z klávesnice, vypsat znak a celý řetězec.

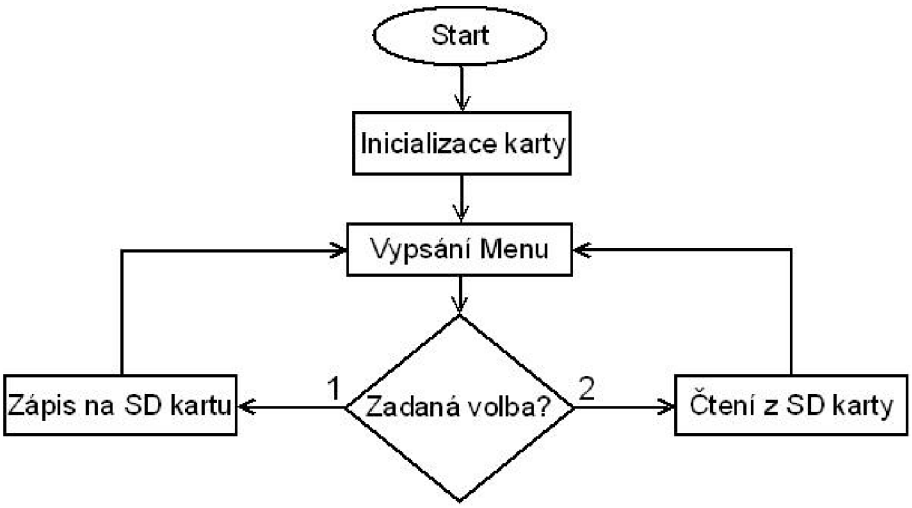
Funkce RecChar

```
char RecChar() {
  byte rec_char;

  if (SCIS1_RDRF)        /* testování příznaku čtení */
    rec_char = SCID;     /* čtením datového registru vymažu příznak */
  SCIC2_RE = 1;          /* povolení příjmu dat */
  while(!SCIS1_RDRF){}; /* dokud nenastaven příznak pro čtení, čekám */
  rec_char = SCID;       /* přečtu přijatý byte */
  SendChar(rec_char);    /* vrácení znaku, aby uživatel věděl co zadal */
  return SCID;           /* vrácení obsahu datového registru */
}
```

Nekonečná smyčka firmwaru, kterou vykonává následující kód programu, je názorně zobrazena na obrázku 4.2.

```
for(;;) {
  menu();                                  /* výpis menu */
  switch (RecChar()) {
    case 0x31  : writeSD();  break;        /* volba 1, volání fce writeSD() */
    case 0x32  : readSD();  break;         /* volba 2, volání fce readSD()*/
    default: SendMsg("\nNeplatna volba\r\n"); /* špatně zadaná volba */
  }
}
```



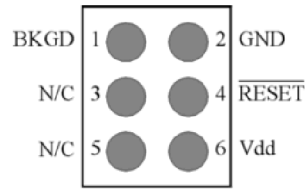
Obrázek 4.2: Vývojový diagram funkce firmwaru

4.2.2 Programování mikrokontroléru

K programování mikrokontrolérů rodiny HCS08 je potřeba programátor USB HCS08 Multilink. Toto zařízení komunikuje na straně PC přes rozhraní USB a na straně mikrokontroléru přes 6-ti pinový konektor BDM (Background Debug Mode) 4.3.

K propojení programátoru a mikrokontroléru stačí pouze 4 piny, a to !reset, BKGD a napájení (Vdd, Vss). Zbylé 2 piny konektoru programátoru jsou nevyužité.

Z důvodu zjednodušení návrhu desky plošných spojů a neumístění konektoru BDM na ní, jsem mikrokontrolér programoval přes kontaktní nepájivé pole. Toto řešení se ovšem neosvědčilo, protože neustálé „přehazování“ mikrokontroléru je zdlouhavé a časem došlo k ohnutí jeho pinů.



Obrázek 4.3: 6-ti pinový konektor BDM

Kapitola 5

Testování

Testování čtečky paměťových karet jsem prováděl v programu HyperTerminál, který je standardně obsažen v operačních systémech Windows a umožňuje komunikovat se zařízeními přes sériový port.

Pro správnou komunikaci programu se čtečkou je potřeba tento program správně nakonfigurovat podle nastaveného přenosu rozhraní SCI mikrokontroléru. Proto je potřeba v HyperTerminálu nastavit:

- Připojení pomocí portu COM
- Přenosovou rychlost 9600 bitů za sekundu
- 8 datových bitů
- Žádná parita
- 1 stop-bit
- Žádné řízení toku

Po správné konfiguraci a připojení napájení se v okně HyperTerminálu zobrazí základní údaje o čtečce jako jsou verze FW, autor a poté se vypíše menu s nabízenými možnostmi – čtení a zápis na kartu SD.

Při volbě „zápis na kartu SD“ se uživateli zobrazí výzva k zadání textu k uložení na paměťovou kartu. Poté program ukládá zadané znaky z klávesnice až do maximální délky 255 znaků. Další zadané znaky se zahazují a na další průběh nemají žádný vliv. Ukončení zadávání a uložení na SD kartu se provede až po zmáčknutí klávesy Esc (Escape). Zápis se provádí na mnou náhodně zvolenou adresu v paměťovém prostoru SD karty. Poté se opět do okna hyperterminálu vypíše menu s nabídkou.

Pokud si uživatel vybere volbu „čtení z karty SD“, na obrazovku se vypíše text přečtený z paměťové karty. I po této operaci se opět vypíše menu pro uživatele.

Při zadání špatné volby, což jsou všechny znaky kromě „1“ a „2“, se zobrazí varovná hláška, která upozorní na špatně zadanou volbu.

Během vývoje firmwaru jsem pro snazší ladění používal ladící výpisy do okna hyperterminálu a červenou LED diodu. Verze firmwaru, který obsahuje veškeré ladící výpisy, je součástí přiloženého CD.

Obrázky z testování v HyperTerminálu lze najít v příloze D.

Kapitola 6

Závěr

Čtečky paměťových karet jsou v dnešní době nepostradatelnou součástí. Jejich výroba však není zrovna jednoduchá, protože hardware, který zajišťuje přenos dat, se musí přizpůsobit paměťové kartě, která má svůj komunikační protokol. Dodržování tohoto protokolu je nutné, protože i sebemenší chyba, jak jsem se při programování několikrát přesvědčil, může způsobit nefunkčnost celého zařízení.

V této bakalářské práci jsem se zaměřil na paměťové karty SecureDigital, zkráceně SD. Díky možnosti komunikace těchto karet protokolem SPI, byl celý návrh zařízení přizpůsoben mikrokontroléru firmy Freescale – MC9S08QG8 – který jako jeden z mála lze sehnat v pouzdře PDIP a současně obsahuje komunikační rozhraní SPI a SCI.

Při vývoji firmwaru nastaly největší problémy při provádění inicializace karty. Jak už bylo řečeno, zasílání příkazů a odpovědí má přesně daný časový diagram, podle kterého se musí firmware řídit. Další obtíže nastaly při nastavování rozhraní SCI pro komunikaci s PC, kdy bylo potřeba nastavit rychlost přenosu na 9600 baudů (bitů za sekundu), při použití vnitřního oscilátoru hodinového signálu. Tento problém byl již vyřešen na stránkách firmy Freescale [8] v diskuzích, do kterých se zapojilo několik uživatelů.

Jako možné rozšíření této čtečky by mohlo být přepracování firmwaru, který by umožnil ukládat větší objemy dat, protože aktuální verze je omezena pamětí vybraného mikrokontroléru. To by mohlo být řešeno tak, že přečtený byte z klávesnice by se ihned posílal jako součást datového bloku přímo do paměťové karty.

Další možná rozšíření by mohla být podpora zápisu a čtení souborů nebo podpora více druhů paměťových karet. Tato čtečka byla testována pouze s kartami SD, avšak měla by být kompatibilní i s kartami MMC, které používají téměř shodný protokol. Tato skutečnost však nebyla prakticky ověřena. Obě tato možná rozšíření nejsou jednoduchá a vyžadovaly by použití jiného mikrokontroléru a tím i nový návrh desky plošných spojů. V případě podpory zápisu a čtení souborů by bylo nutné implementovat základní souborový systém FAT.

Literatura

[1] FreeScale Semiconductor. [on-line], 2008.
URL http://www.freescale.com/

[2] FreeScale Semiconductor: MicroControllers MC9S08QG8 DataSheet. [on-line], [cit. 2008-05-10].
URL http://www.freescale.com/files/microcontrollers/doc/data_sheet/MC9S08QG8.pdf

[3] Maxim Integrated Products: MAX3222-MAX3241 DataSheet. [on-line], [cit. 2008-05-10].
URL http://datasheets.maxim-ic.com/en/ds/MAX3222-MAX3241.pdf

[4] SanDisk Corporation: DataSheet SanDisk MultiMedia Card. [on-line], [cit. 2008-05-10].
URL
http://www.sandisk.com/Assets/File/OEM/Manuals/manual-rs-mmcv1.0.pdf

[5] SanDisk Corporation: DataSheet SanDisk Secure Digital Card. [on-line], [cit. 2008-05-10].
URL http://www.cs.ucr.edu/~amitra/sdcard/ProdManualSDCardv1.9.pdf

[6] Wikipedia. [on-line], [cit. 2008-05-10].
URL http://cs.wikipedia.org/wiki/Pam%C4%9B%C5%A5ov%C3%A1_karta

[7] CadSoft: Eagle V4.16r2. [on-line], 2006.
URL http://www.cadsoft.de/cgi-bin/download.pl?page=/home/cadsoft/html_public/download.htm.en&dir=eagle/program/4.1

[8] FreeScale: Forum. [on-line], 2008.
URL http://forums.freescale.com/

[9] Schwarz, J.; Růžička, R.; Strnadel, J.: Studijní opora k předmětu IMP. FIT VUT. Brno, 2006.

Seznam příloh

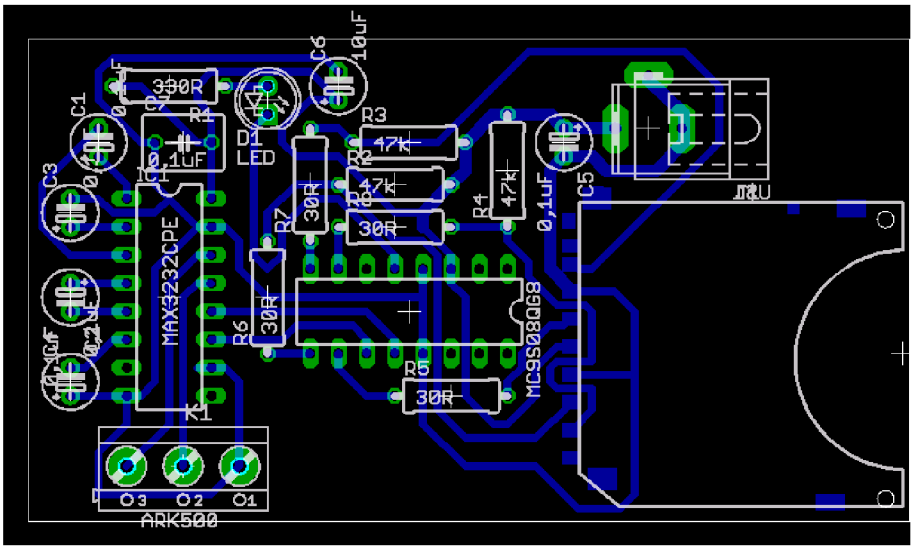
Příloha A - Obsah CD
Příloha B - DPS
Příloha C - Fotodokumentace
Příloha D - Práce v hyperterminálu
Příloha E - CD

Příloha A - Obsah CD

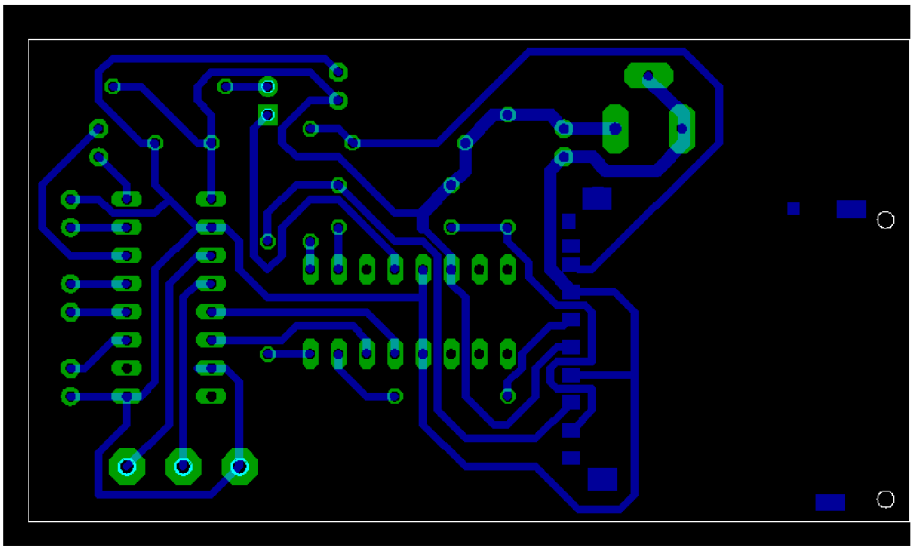
Na přiloženém CD se nachází tyto soubory a adresáře:

- Adresář Schema - obsahující schéma a desku plošných spojů vytvořenou v programu Eagle v4.16r2
- Adresář Firmware - obsahující projekty z CodeWarrioru 5.1 se 2-ma verzemi firmwaru
- Adresář Doc - obsahující datasheety mikrokontroléru a paměťových karet
- Adresář Thesis - obsahující zdrojové kódy technické zprávy v LaTeXu
- Adresář Foto - obsahující fotografie v plném rozlišení
- Soubor Thesis.pdf - technická zpráva ve formátu pdf

Příloha B - DPS

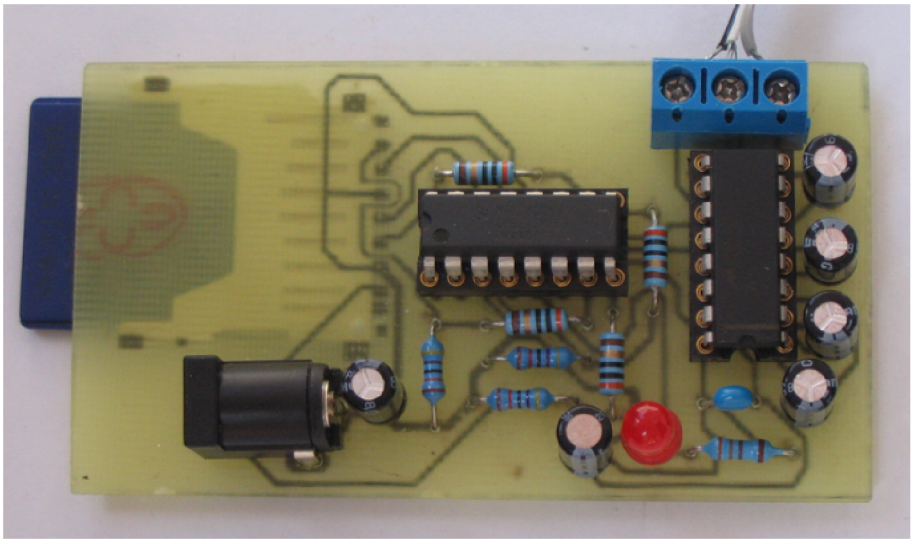


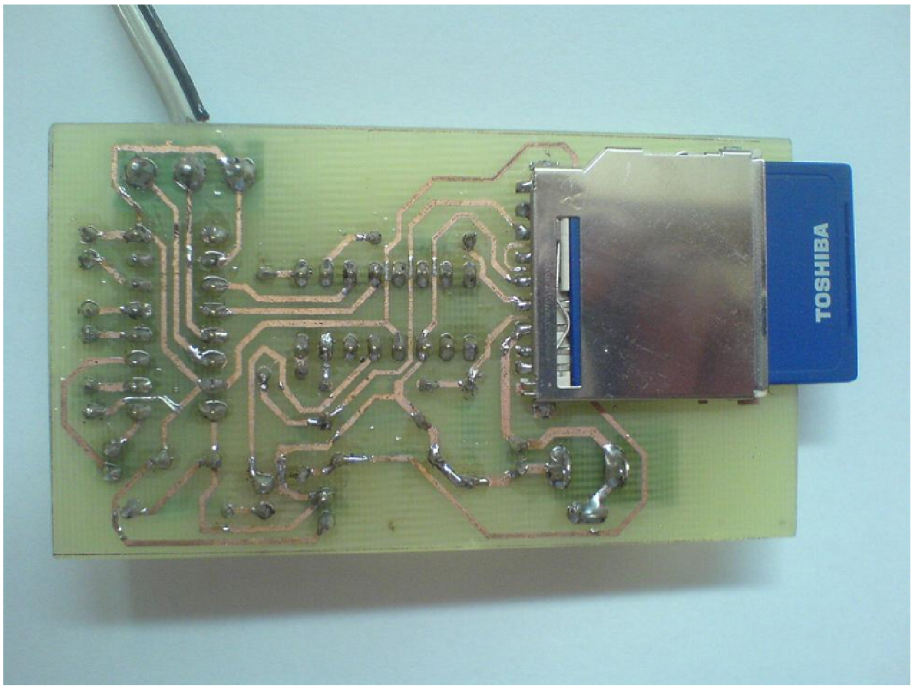
Obrázek 1: Rozmístění součástek na DPS

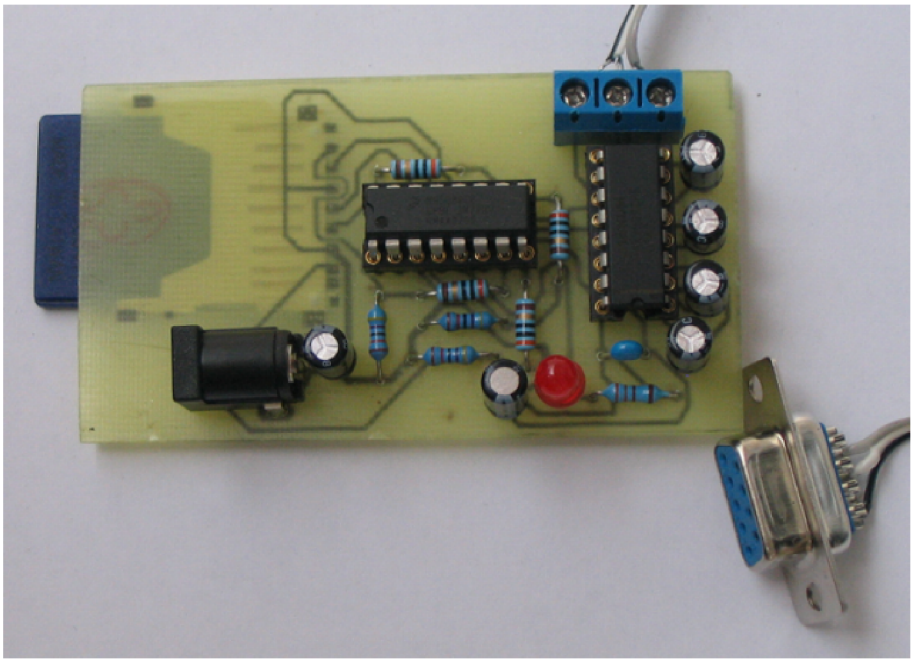


Obrázek 2: Spoje

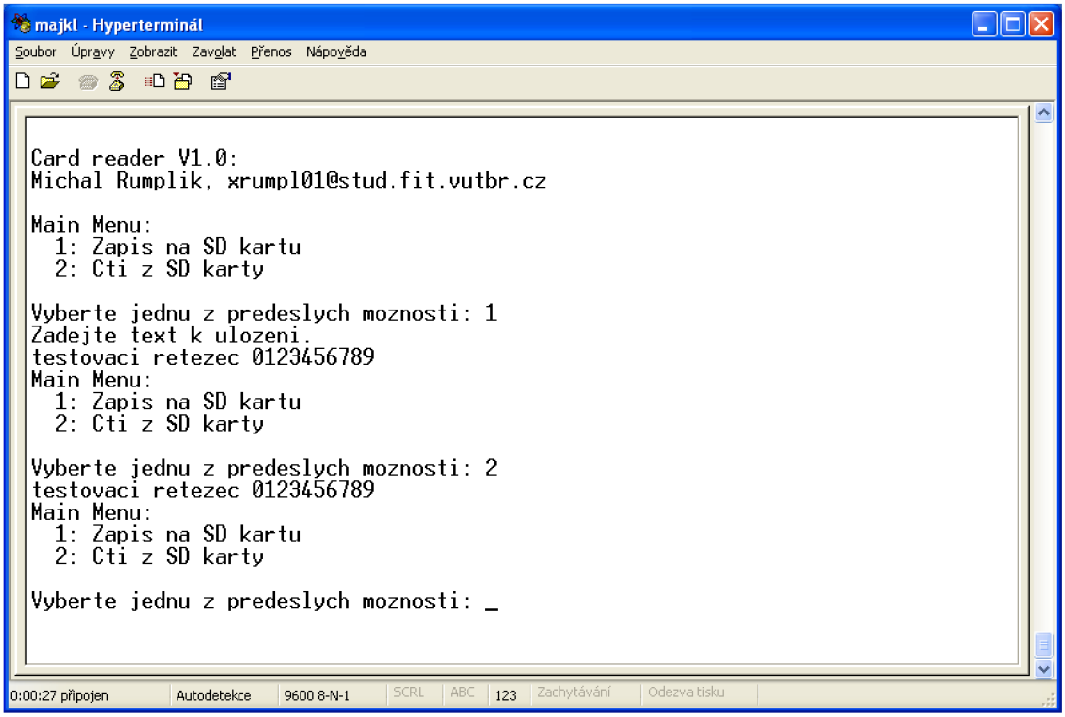
Příloha C - Fotodokumentace

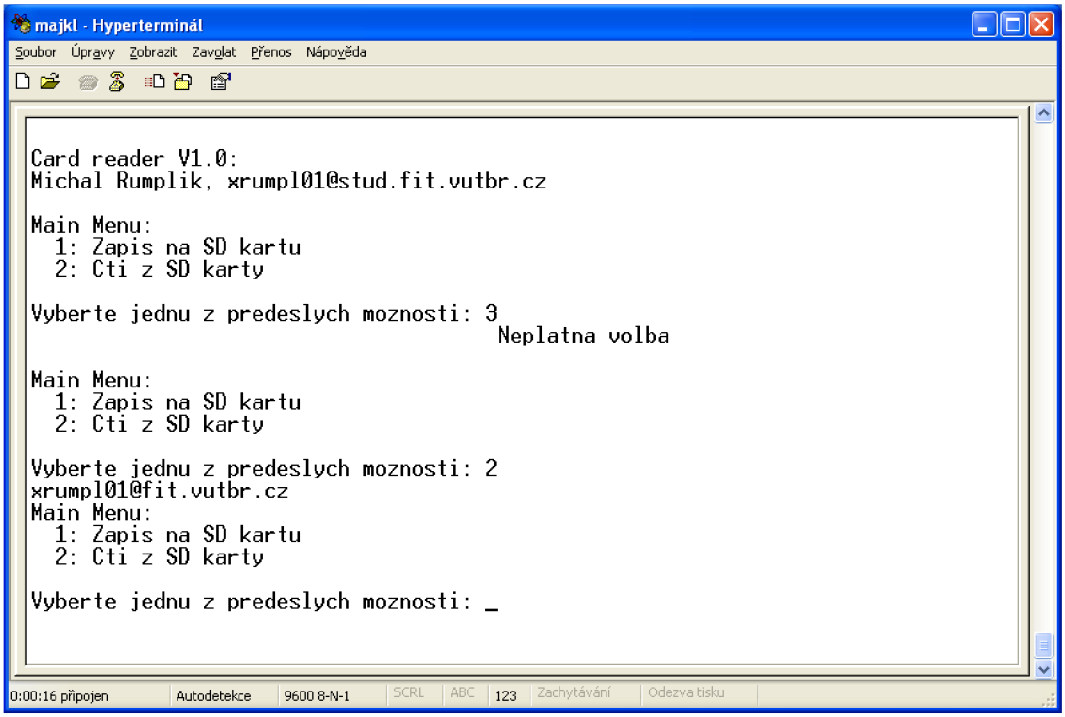


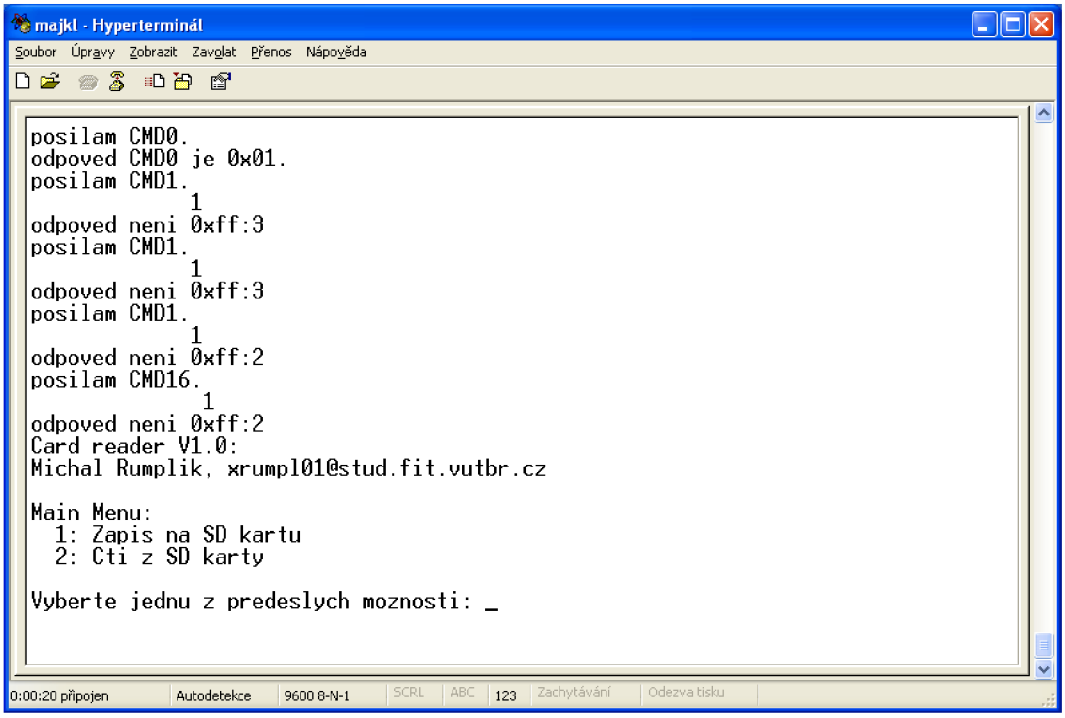




Příloha D - Práce v hyperterminálu




majkl - Hyperterminál
Soubor Úpravy Zobrazit Zavolat Přenos Nápověda
Card reader V1.0:
Michal Rumplik, xrumpl01@stud.fit.vutbr.cz
Main Menu:
1: Zapis na SD kartu
2: Cti z SD karty
Vyberte jednu z predeslych moznosti: 3
Neplatna volba
Main Menu:
1: Zapis na SD kartu
2: Cti z SD karty
Vyberte jednu z predeslych moznosti: 2
xrumpl01@fit.vutbr.cz
Main Menu:
1: Zapis na SD kartu
2: Cti z SD karty
Vyberte jednu z predeslych moznosti: _
0:00:16 připojen
Autodetekce
9600 8-N-1
SCRL
ABC
123
Zachytávání
Odezva tisku


majkl - Hyperterminál
Soubor Úpravy Zobrazit Zavolat Přenos Nápověda
posilam CMD0.
odpoved CMD0 je 0x01.
posilam CMD1.
1
odpoved neni 0xff:3
posilam CMD1.
1
odpoved neni 0xff:3
posilam CMD1.
1
odpoved neni 0xff:2
posilam CMD16.
1
odpoved neni 0xff:2
Card reader V1.0:
Michal Rumplik, xrumpl01@stud.fit.vutbr.cz
Main Menu:
1: Zapis na SD kartu
2: Cti z SD karty
Vyberte jednu z predeslych moznosti: _
0:00:20 připojen
Autodetekce
9600 8-N-1
SCRL
ABC
123
Zachytávání
Odezva tisku