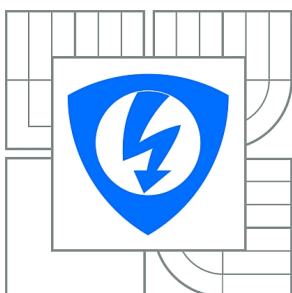




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ

ÚSTAV MIKROELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF MICROELECTRONICS

## VELKOKAPACITNÍ PAMĚŤOVÉ ZAŘÍZENÍ PRO OBVODY FPGA

MASS STORAGE DEVICE FOR FPGAS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MIROSLAV HORYNA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. MARIÁN PRISTACH

BRNO 2013



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav mikroelektroniky

# Bakalářská práce

bakalářský studijní obor  
**Mikroelektronika a technologie**

**Student:** Miroslav Horyna

**ID:** 140229

**Ročník:** 3

**Akademický rok:** 2012/2013

## NÁZEV TÉMATU:

**Velkokapacitní paměťové zařízení pro obvody FPGA**

## POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s možnostmi záznamu dat na externí paměťové zařízení. Na základě zadaných parametrů vyberte vhodný typ paměťové karty pro komunikaci s obvodem FPGA. Navrhněte rozhraní pro komunikaci s kartou. Součástí rozhraní bude vyrovnávací paměť pro urychlení komunikace s kartou. Návrh popište v jazyce VHDL a implementujte do FPGA Spartan3 XC3S200. Pro připojení karty navrhněte rozšiřující modul k vývojové desce. Navrhněte aplikaci demonstrující funkci zařízení, která bude provádět ukládání a načítání dat z karty.

## DOPORUČENÁ LITERATURA:

Podle pokynů vedoucího práce

**Termín zadání:** 11.2.2013

**Termín odevzdání:** 6.6.2013

**Vedoucí práce:** Ing. Marián Pristach

**Konzultanti bakalářské práce:**

**doc. Ing. Jiří Háze, Ph.D.**  
*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **Abstrakt**

Tato bakalářská práce se zabývá návrhem velkokapacitního paměťového zařízení pro obvod FPGA Spartan-3. Je zde popsán princip komunikace s paměťovou kartou, pomocí komunikačních protokolů. Dále je zde popsán návrh rozšiřujícího modulu pro vývojovou desku Xilinx Spartan-3 Starter Kit board a blokový popis programu pro obvod FPGA.

## **Klíčová slova**

SD karta, USB, UART, SPI, Spartan-3, FPGA

## **Abstract**

This bachelor thesis deals with mass storage devices for FPGAs Spartan-3. There is described the principle of communication with a memory card using communication protocols. The proposal for extension module for development board Xilinx Spartan-3 Starter Kit board and block program description for the FPGA device are presented.

## **Keywords**

SD card, USB, UART, SPI, Spartan-3, FPGA

### **Bibliografická citace:**

HORYNA, M. Velkokapacitní paměťové zařízení pro obvody FPGA. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav mikroelektroniky, 2013. 59 s. Vedoucí bakalářské práce Ing. Marián Pristach.

## Prohlášení

Prohlašuji, že svou bakalářskou práci na téma „**Velkokapacitní paměťové zařízení pro obvody FPGA**“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne 4. června 2013

.....  
podpis autora

## Poděkování

Děkuji vedoucímu bakalářské práce Ing. Mariánu Pristachovi za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne 4. června 2013

.....  
podpis autora

# Obsah

Seznam obrázků.....	8
Seznam tabulek.....	9
Úvod.....	10
1 Paměťové karty.....	11
1.1 SD karty.....	11
1.2 Komunikace s SD kartami.....	12
1.2.1 Režim SPI.....	12
1.2.2 Režim SD.....	14
1.3 Inicializace SD karty.....	15
1.4 Ochrana přenášených dat.....	15
1.5 Čtení dat.....	15
1.6 Zápis dat.....	16
1.7 Registr CID.....	16
1.8 Registr CSD.....	17
1.9 Příkazový rámec.....	19
1.10 Odpovědní rámce (tokeny).....	20
1.10.1 Odpovědní rámec R1.....	20
1.10.2 Odpovědní rámec R2.....	20
1.10.3 Odpovědní rámec R3.....	21
1.10.4 Odpovědní rámec R7.....	22
1.10.5 Datová odezva.....	22
1.10.6 Start Bajt.....	23
2 Velkokapacitní zařízení.....	24
2.1 Obvod FPGA Spartan-3.....	24
2.2 Vývojová deska Xilinx Spartan-3 Starter Kit board.....	24
2.3 Komunikace zařízení.....	25
2.3.1 SPI komunikace.....	25
2.3.2 UART komunikace.....	27

3	Návrh obvodového zapojení modulu.....	29
3.1	Obvod FTDI FT232RL.....	30
4	Návrh vnitřního zapojení.....	31
4.1	Blok SPI rozhraní.....	32
4.1.1	Vnitřní návrh SPI rozhraní.....	33
4.2	Blok SD kontrolér.....	35
4.2.1	Vnitřní návrh bloku SD kontrolér a popis klidového stavu.....	36
4.2.2	Inicializace karty.....	37
4.2.3	Zápis dat na kartu.....	40
4.2.4	Čtení dat z karty.....	42
4.3	Blok Buffer.....	43
4.4	Blok UART.....	44
4.4.1	Modul generátoru přenosové rychlosti.....	45
4.4.2	Modul UART RX.....	45
4.4.3	Modul UART TX.....	46
4.4.4	Modul FIFO paměti.....	47
5	Návrh demonstrační aplikace.....	49
6	Závěr.....	50
	Seznam literatury.....	51
	Seznam symbolů a zkratk.....	52
	Seznam příloh.....	54

## Seznam obrázků

Obr. 1 - Zapojení SPI sběrnice .....	13
Obr. 2 - Rozmístění pinů na SD kartě [1].....	13
Obr. 3 - zapojení SD sběrnice.....	14
Obr. 4 - Čtení bloku dat [1] .....	16
Obr. 5 - Zápis bloku dat [1] .....	16
Obr. 6 - Formát příkazového rámce [1].....	19
Obr. 7 - Odpovědní rámec R1[1].....	20
Obr. 8 - Odpovědní rámec R2 [1].....	20
Obr. 9 - Odpovědní rámec R3 [1].....	21
Obr. 10 - Odpovědní rámec R7 [1].....	22
Obr. 11 - Rámec datové odezvy [1].....	22
Obr. 12 - Definice Start bajtu [1].....	23
Obr. 13 - Vývojová deska Xilinx Spartan-3 Starter Kit board [6] .....	25
Obr. 14 - Základní koncepce SPI rozhraní [8].....	26
Obr. 15 - Časový diagram SPI [8] .....	27
Obr. 16 - Zobrazení dvou zařízení pro komunikaci pomocí UART rozhraní [10].....	27
Obr. 17 - Datový rámec UART komunikace s jedním start bitem 8 datovými bity paritním bitem a jedním stop bitem [10].....	28
Obr. 18 - Jednotlivé vývody obvodu FT232RL [7].....	30
Obr. 19 - Blokové schéma programové části pro komunikaci s počítačem .....	31
Obr. 20 - Blokové schéma programové části pro komunikaci se zařízením pro sběr dat .....	31
Obr. 21 - Blok SPI rozhraní.....	32
Obr. 22 - Stavový automat bloku SPI rozhraní .....	33
Obr. 23 - Přenos příkazu CMD8 pomocí SPI rozhraní. Výstup z logického analyzátoru.....	34
Obr. 24 - Inicializační proces [1].....	35
Obr. 25 - Blok SD kontrolér .....	36
Obr. 26 - Základní rozdělení SD kontroléru.....	37
Obr. 27 - Stavový diagram inicializačního procesu .....	38
Obr. 28 - Stavový automat pro zápis dat na kartu .....	41
Obr. 29 - Stavový automat pro čtení z karty .....	42
Obr. 30 - Blok Buffer .....	43
Obr. 31 - Blok UART.....	44
Obr. 32 - Vnitřní zapojení Bloku UART.....	45
Obr. 33 - Stavový automat UART RX .....	46
Obr. 34 - Stavový automat UART TX .....	47
Obr. 35 - FIFO paměť.....	48



## Seznam tabulek

Tab. 1 - Rozdělení SD karet [4].....	12
Tab. 2 - Pojmenování jednotlivých pinů SD karty v SPI režimu [1] .....	13
Tab. 3 – Pojmenování jednotlivých pinů SD karty v SD režimu [1].....	14
Tab. 4 – Obsah CID registru [1] .....	17
Tab. 5 – základní obsah CSD registru [1] .....	17
Tab. 6 - Přehled nejčastěji používaných příkazů [1] .....	19
Tab. 7 - Hodnoty akceptovaného napětí pro odpovědní rámec R7 [1].....	22
Tab. 8 - Definice status bitů [1].....	23
Tab. 9 - Rozmístění a přiřazení pinů na konektoru .....	29

# Úvod

Velkokapacitní paměťová zařízení se v dnešní době používají ve všech odvětvích lidské činnosti, kde se setkáme s výpočetní technikou. Požívají se pro ukládání a zálohování datových informací. Velkokapacitní paměťová média můžeme rozdělit na tři základní druhy: magnetická média, optická média a flash paměť. Typickým zástupcem magnetických médií je pevný disk, dalším typickým zástupcem pro optická média je CD (Compact Disc). U flash paměti máme zástupce například USB Flash disk nebo paměťovou kartu.

Cílem této práce je navrhnout zařízení komunikující s paměťovou kartou. Zařízení má sloužit jako komunikační rozhraní mezi počítačem a kartou, kam se data ukládají. Úkolem je návrh obvodového řešení takového rozhraní a návrh vnitřního zapojení.

Rozhraní bude pomocí zvolených komunikačních protokolů komunikovat s paměťovými kartami. Hlavní částí tohoto rozhraní bude obvod FPGA, ve kterém bude implementován vnitřní návrh pro komunikaci mezi počítačem a paměťovou kartou. Načítání a zápis dat do karty bude provádět aplikace demonstrující funkčnost rozhraní.

Hlavní použití takového rozhraní by mělo být rozšíření zařízení s FPGA obvody o možnost ukládání nebo načítání dat z paměťové karty. V případě ASIC obvodů by to mohl být dlouhodobý sběr velkého množství dat, které by se následně vyhodnocovalo, např.: monitorování prostředí, staveb, strojů apod.

# 1 Paměťové karty

Paměťových karet je několik druhů např.: Compact flash, Multi media card (MMC), SD karty (Secure digital), Memory Stick. Karty Compact flash se používají především v poloprofesionálních a profesionálních digitálních přístrojích, nejčastěji v digitálních fotoaparátech. Jejich rozměry jsou 42,8 x 36,5 x 3,3 mm nebo 42,8 x 36,5 x 5 mm a kapacita je až do 128 GB. Multi media card je předchůdce SD karet. Mají stejné plošné rozměry jako SD karty 24 × 32 mm. Oproti SD kartám mají tloušťku 1,4 mm a jejich kapacita je od 16 MB do 32 GB. Karty Memory Stick vyvinula společnost Sony pro svá zařízení. Karta je obdélníkového tvaru a její rozměry činí 50 × 21,5 × 2,8 mm a jejich kapacita je od 4 MB do 32 GB. SD karty jsou nejrozšířenější formát dneška, respektive používaný ve většině komerčních přístrojů. Některé karty umožňují zamknout obsah proti přepsání/vymazání mechanickým přepínačem. Vyrábí se ve velikostech standart SD 24 × 32 x 2,1 mm, miniSD 20 x 21,5 x 1,4 mm, mikroSD 15 × 11 × 0,7 mm a s kapacitami začínajícími na 1 MB a končí u 256 GB.

Pro svou práci jsem si vybral SD karty z důvodu přesné a volně dostupné dokumentace, ve které je popsáno, jak přesně s kartou komunikovat.

## 1.1 SD karty

SD karta (Secure Digital Card) je paměťová karta typu flash, která je specificky navržena tak, aby splnila požadavky na zabezpečení, kapacitu, výkon a environmentální požadavky spočívající v nově vznikajících zařízeních spotřební elektroniky. SD karta vychází z karet typu MMC (Multi Media Card). První karta vznikla v roce 1999 za spolupráce společností Panasonic, Toshiba a SanDisk.

Klasické SD karty s rozměry 32 x 24 x 2,1 mm byly dále zmenšovány. V roce 2003 se na trhu objevila miniSD karta mající rozměry 21,5 x 20 x 1,4 mm. V roce 2005 byla vyrobena microSD karta s rozměry 15 x 11 x 1 mm. Tyto klasické karty měli kapacitu od 1 MB až do 2 GB.

Jejich nástupcem jsou karty typu SDHC (Secure Digital High-Capacity), které disponují větší pamětí. Velikost jejich paměti začíná na hodnotě 4 GB a končí s hodnotou 32 GB. Aby bylo možné zvětšit kapacitu paměťových karet SD, bylo nutné upravit adresování paměti. Zatímco SD karty používají 32-bitové adresové pole, udávané v bajtech, karty SDHC používají 32-bitové blokové adresy. Z tohoto důvodu jsou karty SDHC nekompatibilní se staršími čtečkami. I tyto karty se vyrábí ve všech třech velikostech jako SD karty.

Nejnovějším druhem SD karet jsou karty typu SDXC (Secure Digital eXtended Capacity), které mají velikost paměti od 32 GB do 256 GB. Karty typu SDXC se vyrábí pouze ve velikostech klasické SD karty a microSD karty. [3]

Existují také karty typu SDIO (Secure Digital Input Output). Jsou to karty, které místo paměti obsahují jiné funkční obvody, zabudované do formátu karty SD např.: přijímače GPS, adaptéry Bluetooth, IrDA, LAN nebo Wi-Fi, čtečky RFID, tunery FM ad. Karty jsou plně kompatibilní s řadičem pro paměťové karty SD. Při vložení karty SDIO do zařízení, které je určeno jen pro paměti, se nic nepoškodí, ale nebude ani fungovat. Naopak ve slotu pro SDIO karty lze používat i běžné paměťové karty SD. [4]

**Tab. 1 - Rozdělení SD karet [4]**

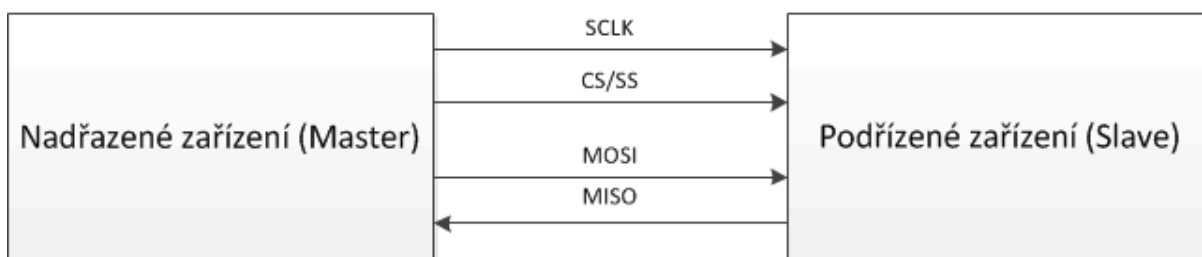
		SD	miniSD	microSD
Kapacita	SD	1 MB – 2 GB		
	SDHC	4 GB – 32 GB		
	SDXC	32 GB – 256 GB		
Rozměry		32 x 24 x 2,1 mm	21,5 x 20 x 1,4 mm	15 x 11 x 1 mm
Přenos SPI		25 Mb/s	25 Mb/s	25 Mb/s
Hodinový kmitočet		0 – 50 MHz	0 – 25 MHz	0 – 25 MHz

## 1.2 Komunikace s SD kartami

Ke komunikaci s kartami založenými na standartu SD se nejčastěji využívá dvou komunikačních režimů. A to režimu SPI a jednobitový SD režim. Třetím komunikačním režimem je čtyřbitový SD režim. [1] [2]

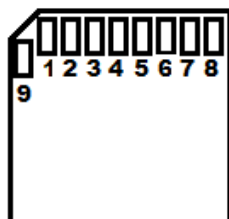
### 1.2.1 Režim SPI

Při komunikaci v SPI režimu se využívá klasického protokolu pro komunikaci po sběrnici SPI (Serial Peripheral Interface). Využívá se celkem čtyř vodičů pro komunikaci, znázorněných na obr. 1.



Obr. 1 - Zapojení SPI sběrnice

Dva vodiče, jsou označeny MISO (Master in, Slave out) a MOSI (Master out, Slave in), přes ně se přenáší data. Vodič s označením SS (Slave select) nebo CS (Chip select) určuje, které podřízené zařízení (slave) má právě v daný okamžik komunikovat s nadřazeným zařízením (master). Takto vybrané zařízení má na svůj pin SS/CS přivedenu logickou 0, čímž se aktivuje pro komunikaci. Poslední vodič je označen jako SCLK, na který je z nadřazeného zařízení přiveden hodinový synchronizační signál, který se v něm generuje. V této práci je jako nadřazené zařízení použita vývojová deska Spartan-3 Starter kit a jako slave je zde SD karta. Rozmístění pinů SD karty je znázorněno na obr. 2, pojmenování jednotlivých pinů SD karty v režimu SPI se nalézá v tab. 2. [1] [2]



Obr. 2 - Rozmístění pinů na SD kartě [1]

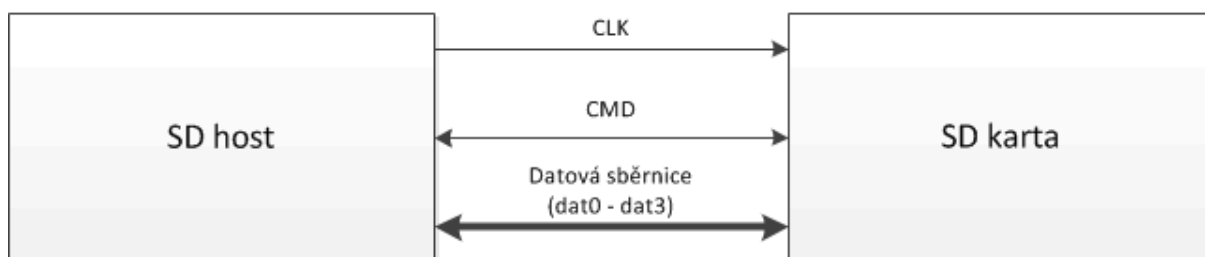
Tab. 2 - Pojmenování jednotlivých pinů SD karty v SPI režimu [5][1]

Pin #	Název	Typ	SPI popis
1	CS/SS	vstup	Výběr zařízení (aktivní na log. 0)
2	DataIn	vstup	Vstupní data ze zařízení (MOSI)
3	VSS1	napájení	Uzemnění
4	VDD	napájení	Napájecí napětí
5	CLK	vstup	Vstup hodinového signálu
6	VSS2	napájení	Uzemnění
7	DataOut	výstup	Výstupní data z karty (MISO)
8	RSV	n/a	Rezervováno
9	RSV	n/a	Rezervováno

## 1.2.2 Režim SD

Komunikace v SD režimu je založena na řídicích a datových tocích, které se spouští start bitem a ukončují stop bitem. SD režim umožňuje dynamickou konfiguraci počtu datových linek od 1 do 4 obousměrných datových signálů. Po zapnutí je karta ve výchozím nastavení, což znamená, že je připravena komunikovat v jednobitovém SD režimu. Příkazy jsou posílány uživatelem po lince CMD a karta na ně odpovídá pomocí odpovědních rámců. Na příkazy karta odpovídá rámci R1, R2, R3 po CMD lince nebo datovými rámci po datové lince. Příkazy v SD režimu mají tvar „CMDxx“ a „ACMDxx“, přičemž „CMD“ a „ACMD“ značí příkaz a „xx“ číslo konkrétního příkazu, který je posílán.

Příkazy jsou posílány pomocí příkazového rámce délky 6 bajtů. Rámec má vždy na prvních dvou bitech hodnotu „01“, kterou následuje šestice bitů označující příkaz. Další čtyři bajty jsou argument příkazu. Posledních 7 bitů je vyhrazeno pro kontrolní součet CRC a stop bit logická „1“. Příkazy se posílají na port CMD karty, popis pinů na kartě je v tab. 3, pozice jednotlivých pinů je na obr. 2. Komunikace probíhá pomocí tří až šesti vodičů a je znázorněna na obr. 3.



Obr. 3 - zapojení SD sběrnice

Tab. 3 – Pojmenování jednotlivých pinů SD karty v SD režimu [1]

Pin #	Název	Typ	SD popis
1	DAT3	Vstup/výstup	Datová linka (bit 3)
2	CMD	Vstup/výstup	Příkazy/Odpovědi
3	VSS1	napájení	Uzemnění
4	VDD	napájení	Napájecí napětí
5	CLK	vstup	Vstup hodinového signálu
6	VSS2	napájení	Uzemnění
7	DAT0	Vstup/výstup	Datová linka (bit 0)
8	DAT1	Vstup/výstup	Datová linka (bit 1)
9	DAT2	Vstup/výstup	Datová linka (bit 2)

Při každém cyklu tohoto signálu se provádí převod jednoho bitu na příkazové a datové lince. Příkazy jsou přenášeny sériově na CMD lince. Příkaz je rámec, kterým začíná operace

z hostitelského zařízení na kartu. Odpovědi jsou také přenášeny sériově na CMD lince. Karta odpovídá na předchozí příkaz odpovědním rámcem. Data mohou být přenášena z karty do hostitelského zařízení nebo naopak. Data jsou přenášena přes datové linky. [1] [2]

### 1.3 Inicializace SD karty

Po připojení napájecího napětí na kartu je tato karta standardně v režimu SD. Pro přepnutí do režimu SPI je na CS pin karty přivedena logická úroveň „0“ a musí se odeslat resetovací příkaz CMD0. Pro uvedení karty zpět do SD režimu musíme provést „tvrdý reset“, což je odpojení karty od napájení po dobu 1 ms. Po příkazu CMD0 posíláme příkaz CMD8, kterým ověřujeme funkčnost karty pro připojené napájecí napětí. SD karty verze 1 tento příkaz nepodporují, proto lze očekávat odpověď, že je tento příkaz chybný. Obdržená odpověď také může značit, že vložená karta není typu SD. Při obdržení správné odpovědi víme, že vložená karta je verze 2 či novější. V odpovědi karty sledujeme bit VCA. Jestliže je tento bit nastaven na nulu, tak karta podporuje dané napájecí napětí.

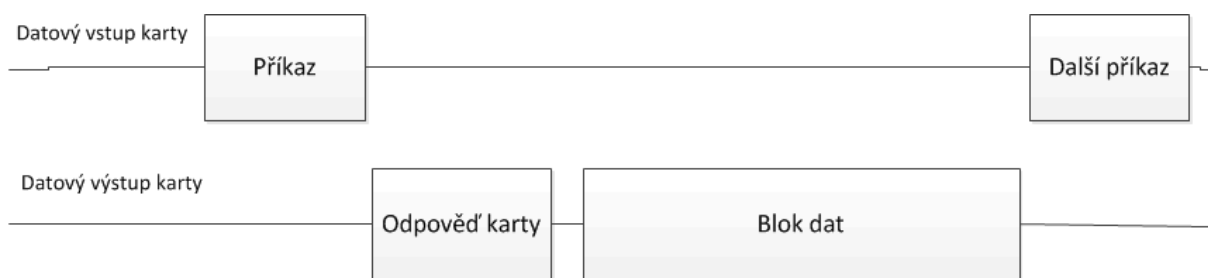
Při obdržení správné odpovědi můžeme zaslat příkaz CMD58 pro získání OCR. Příkaz CMD58 v této chvíli není povinný. Následně odešleme příkaz ACMD41, ten se opakuje, dokud nedostaneme odpověď „0x00“. Nyní je nutné zaslat CMD58 pro získání CCS bitu. Jestliže má bit CSS hodnotu „0“ je vložená karta typu Standard Capacity, pokud má hodnotu „1“ je vložená karta typu High Capacity. Tímto je dokončena inicializace SD karty. [1] [2]

### 1.4 Ochrana přenášených dat

Veškeré odesílané příkazy jsou chráněny kontrolním součtem CRC. V režimu SPI je standardně vypnuta kontrola CRC součtu. Při přepínání do SPI režimu musíme pamatovat na to, že karta je standardně v režimu SD, kde je kontrolní CRC součet vyžadován. I v SPI režimu je kontrolní součet posílán u vyžadovaných rámců, ale je po svém přijetí ignorován. Kontrolní součet může být zapnut posláním příkazu CMD59 (CRC\_ON\_OFF). [1] [2]

### 1.5 Čtení dat

Chceme-li číst data z karty, vyšleme příkaz pro čtení CMD17 nebo CMD18. Přijetím příkazu pro čtení dat karta odešle odpovědní rámec, který je následován blokem požadovaných dat. Čtení dat je zobrazeno na obr. 4. Velikost bloku dat se definuje příkazem CMD16 (SET\_BLOCKLEN). Standardně je velikost bloku nastavena na 512 B.

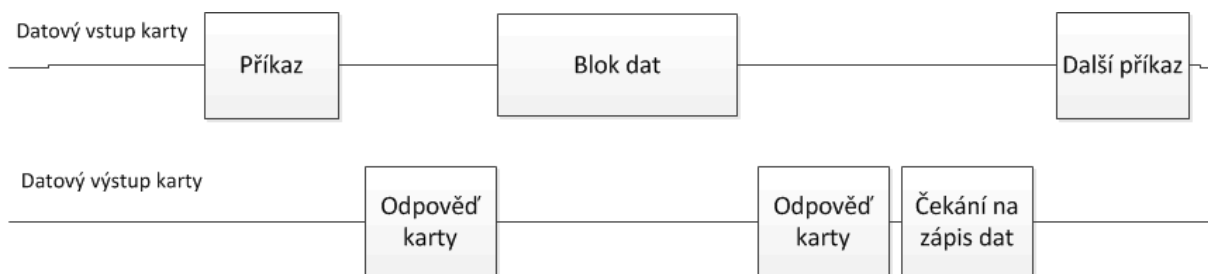


**Obr. 4 - Čtení bloku dat [1]**

Adresa, kde jsou na kartě data uložena, se posílá v příkazovém rámci. Musí být vždy nastavena na začátek čteného bloku. Není možné číst blok 512 B odkudkoliv. Pokud se čtení dat nepovede, vyšle karta místo datového bloku chybové hlášení. [1]

## 1.6 Zápis dat

Chceme-li zapsat data do karty, vyšleme příkaz pro zápis CMD24 nebo CMD25. Po přijetí příkazu pro zápis odešle karta odpovědní rámec a vyčkává na příchozí datový blok. Po přijetí dat odešle karta další odpovědní rámec o příjmu dat. Pokud jsou přijatá data bez chyb, začnou se zapisovat na kartu. Celou dobu, kdy karta zapisuje data do paměti, posílá informaci, že je zaneprázdněná. Po dokončení zápisu můžeme data zkontrolovat příkazem CMD13 viz obr. 5. [1]



**Obr. 5 - Zápis bloku dat [1]**

## 1.7 Registr CID

CID (The Card IDentification) registr je 128 bitů široký. Obsahuje identifikační údaje karty. Každá karta má své jedinečné identifikační číslo. V tab. 4 je popsána struktura CID registru.



Tab. 4 – Obsah CID registru [1]

Jméno	Pole	Velikost [b]	CID pozice
Výrobní ID	MID	8	[127:120]
OEM	OID	16	[119:104]
název	PNM	40	[103:64]
verze	PRV	8	[63:56]
sériové číslo	PSN	32	[55:24]
rezervováno	-	4	[23:20]
datum výroby	MDT	12	[19:8]
kontrolní součet CRC7	CRC	7	[7:1]
nevyužito	-	1	[0:0]

**Popis jednotlivých prvků:**

- **MID** – 8-bitové binární číslo, které identifikuje výrobce karty
- **OID** – 2 znaky z ASCII tabulky, které definují OEM nebo obsah karty (je-li používána jako ROM nebo FLASH)
- **PNM** – název produktu, řetězec 5 znaků ASCII tabulky
- **PRV** – verze karty obsažená v 2 číslech zakódovaných pomocí BCD
- **PSN** – 32-bitové sériové číslo
- **MDT** – Datum výroby se skládá ze dvou hexadecimálních číslic. Jedno je 8 bitů široké a reprezentuje rok (y). Druhé je 4 bity široké, to představuje měsíc (m). "M" pole [11:08] je měsíc, kód 1 = leden. "Y" pole [19:12] je rok, kód 0 = 2000.
- **CRC** – CRC7 kontrolní součet (7 bitů) [1]

## 1.8 Registr CSD

CSD (Card-specific data) registr obsahuje informace o přístupu na kartu. CSD definuje datový formát. Programovatelná část registru je přístupná příkazem CMD27. Struktura dat v CSD registru je odlišná pro různé verze fyzických specifikací a kapacit karet.

Tab. 5 – základní obsah CSD registru [1]

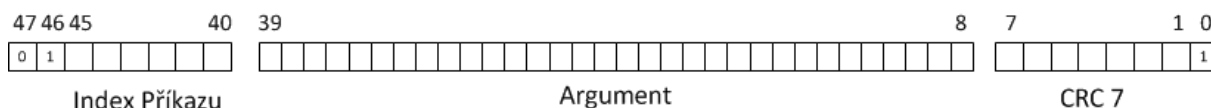
Jméno	Pole	Velikost [b]	CSD pozice
CSD struktura	CSD_STRUCTURE	2	[127:126]
Max. délka bloku dat při čtení	READ_BL_LEN	4	[83:80]
Povolení čtení po blocích	READ_BL_PARTIAL	1	[79:79]
Rozdělení na více bloků při zápisu	WRITE_BLK_MISALIGN	1	[78:78]
Rozdělení na více bloků při čtení	READ_BLK_MISALIGN	1	[77:77]
Velikost zařízení	C_SIZE	12	[73:62]
Násobek velikosti zařízení	C_SIZE_MULT	3	[49:47]
Max. délka bloku dat při zápisu	WRITE_BL_LEN	4	[25:22]
Povolení zápisu po blocích	WRITE_BL_PARTIAL	1	[21:21]
Formát skupiny souborů	FILE_FORMAT_GRP	1	[15:15]
Formát souborů	FILE_FORMAT	2	[11:10]

### Popis jednotlivých prvků:

- **READ\_BL\_LEN** – Maximální délka bloku dat při čtení se rovná  $2^{\text{READ\_BL\_LEN}}$ . Velikost bloku je v rozmezí od 512 B do 2048 B. READ\_BL\_LEN se standardně rovná WRITE\_BL\_LEN.
- **READ\_BL\_PARTIAL** – Částečné čtení bloku u SD karty je povoleno. Data mohou být čteny i s délkou, která je menší než délka bloku, minimální velikost pro čtení dat je jeden byte.
- **WRITE\_BLK\_MISALIGN** – Určuje, jestli lze zapisovaný blok dat, definovaný jediným příkazem, rozdělit na více bloků v paměti SD karty. Velikost bloku se definuje pomocí WRITE\_BL\_LEN.
- **READ\_BLK\_MISALIGN** - Určuje, jestli lze čtený blok dat, definovaný jediným příkazem, rozdělit na více bloků v paměti SD karty. Velikost bloku se definuje pomocí READ\_BL\_LEN.
- **C\_SIZE** – Používá se k výpočtu velikosti kapacity SD karty. Kapacita se určuje z C\_SIZE, C\_SIZE\_MULT, READ\_BL\_LEN.  
Kapacita paměti = BLOCKNR \* BLOCK\_LEN,  
kde:  
$$\text{BLOCKNR} = (\text{C\_SIZE} + 1) * \text{MULT}$$
$$\text{MULT} = 2^{\text{C\_SIZE\_MULT} + 2}$$
$$\text{BLOCK\_LEN} = 2^{\text{READ\_BL\_LEN}}$$
- **C\_SIZE\_MULT** – Parametr určující činitel MULT pro výpočet celkové kapacity paměti SD karty, který je definován jako  $\text{MULT} = 2^{\text{C\_SIZE\_MULT} + 2}$ .
- **WRITE\_BL\_LEN** - Maximální délka bloku dat při zápisu se rovná  $2^{\text{WRITE\_BL\_LEN}}$ . Velikost bloku je v rozmezí od 512 B do 2048 B. WRITE\_BL\_LEN se standardně rovná READ\_BL\_LEN.
- **WRITE\_BL\_PARTIAL** – Určuje, zda mohou být pro zápis použity dílčí bloky.
- **FILE\_FORMAT\_GRP** – Indikuje zvolenou skupinu formátu souborů.
- **FILE\_FORMAT** – Slouží k identifikaci souborového systému na SD kartě. [1]

## 1.9 Příkazový rámec

Všechny příkazy posílané do SD karty mají 6 bajtů. SD standart definuje dva druhy příkazů. Prvním je příkaz typu CDM (command) a druhým typem příkazu je ACMD (applications command). Příkazy jsou odlišeny dvouciferným indexem za názvem typu příkazu, např. CMD58. Obecný formát příkazu je na obr. 6. [1]



Obr. 6 - Formát příkazového rámce [1]

Příkazy posílané kartě začínají vždy start bitem s hodnotou „0“, druhý následuje přenosový bit s hodnotou „1“. V prvním bajtu se nachází také index příkazu s binární hodnotou odpovídající tomuto indexu. Dále následují 4 bajty s argumentem pro určitý příkaz. V posledním bajtu se nachází kontrolní součet CRC 7 a posledním je stop bit s hodnotou „1“. Při použití SPI režimu, není kontrolní součet CRC aktivován. Je vyžadován pouze u příkazu CMD0, kde má se stop bitem hodnotu 95h a u příkazu CMD8. Přehled nejčastěji používaných příkazů v tab. 6. [1]

Před zasláním jakéhokoli příkazu typu ACMD je zapotřebí odeslat příkaz CMD55. Příklad zaslání ACMD41:

- Prvním příkazem je CMD55 s argumentem 00000000h => 77 00 00 00 00 01h =  
= **01** 110111 00000000 00000000 00000000 00000000 00000001b
- Druhým příkazem je ACMD41 s argumentem 40000000h => 69 40 00 00 00 01h =  
= **01** 101001 01000000 00000000 00000000 00000000 00000001b [1]

Tab. 6 - Přehled nejčastěji používaných příkazů [1]

Příkaz	Argument	Odp.	Zkratka	Popis příkazu
CMD0	-	R1	GO_IDLE_STATE	Reset karty
CMD8	[11:8] napájecí napětí (VHS) [7:0] kontrolní vzorek	R7	SEND_IF_COND	Zjištění rozsahu napájení Kontrolní vzorek slouží ke kontrole správné odpovědi karty
CMD16	[31:0] délka bloku	R1	SET_BLOCKLEN	Nastavení velikosti bloku dat
CMD17	[31:0] adresa dat	R1	READ_SINGLE_BLOCK	Přečtení bloku dat
CMD24	[31:0] adresa dat	R1	WRITE_BLOCK	Zápis bloku dat
CMD55	-	R1	APP_CMD	Následuje příkaz ACMD
CMD58	-	R3	READ_OCR	Načti OCR registr
ACMD41	[30] HCS	R1	SD_SEND_OP_COND	Inicializační proces karty

## 1.10 Odpovědní rámce (tokeny)

Existuje několik typů odpovědních rámců. V režimu SD i v režimu SPI je přenášen nejdříve bit MSB. Jednotlivé bity upozorňují na určitý stav popsany v následujících podkapitolách.

### 1.10.1 Odpovědní rámec R1

Odpovědní rámec R1 karta posílá vždy po každém příkazu, s výjimkou příkazu SEND\_STATUS. Rámec R1 má délku 1 bajt a MSB je vždy nastaven na nulu. Bity indikují chybu, jsou nastaveny na logickou 1.



Obr. 7 - Odpovědní rámec R1[1]

#### Význam bitů:

7 – start bit: vždy nastaven na logickou 0

6 – parametr error: argument příkazu byl mimo povolený rozsah karty

5 – address error: v příkazu byla použita adresa, která neodpovídá délce bloku

4 – erase sequence error: došlo k chybě v sekvenci příkazů pro mazání

3 – communication CRC error: kontrolní součet CRC posledního příkazu byl chybný

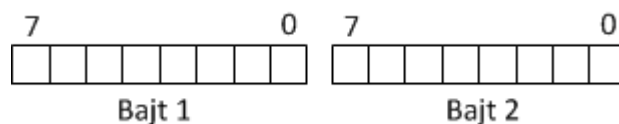
2 – illegal command: byl detekován neplatný příkaz

1 – erase reset: mazací sekvence byla zrušena před dokončením, protože byl přijat příkaz out of erase

0 – in idle state: nečinný [1]

### 1.10.2 Odpovědní rámec R2

Odpovědní rámec R2 má délku 2 bajty a je odesílán jako odpověď na příkaz SEND\_STATUS (CMD13).



Obr. 8 - Odpovědní rámec R2 [1]

## Význam bitů:

### Bajt 2

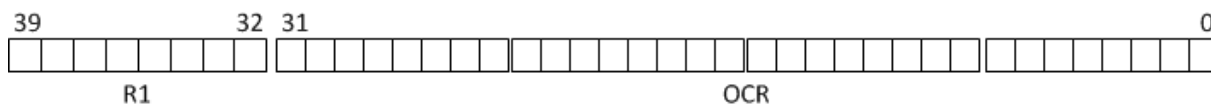
- 7 – out of range / csd overwrite
- 6 – erase param: neplatný sektor nebo skupina pro vymazání
- 5 – wp violation: příkaz zkoušel zapsat do bloku chráněného proti zápisu
- 4 – Card ECC failed: vnitřní ECC bylo použito, ale nepodařilo se opravit data
- 3 – CC error: chyba ve vnitřním řadiči karty
- 2 – Error: nastala obecná nebo neznámá chyba v průběhu operace
- 1 – Write protection erase skip: je nastaven, když se uživatel pokouší vymazat sektor chráněný proti zápisu
- 0 – Card is locked: má hodnotu logická 1, pokud je karta zamčena

### Bajt 1

Identický s R1. [1]

## 1.10.3 Odpovědní rámeček R3

Karta posílá odpovědní rámeček R3, když byl přijat příkaz READ\_OCR (CMD58). Jeho délka je 5 bajtů. První bajt je shodný s R1. Zbylé čtyři bajty obsahují registr OCR.



Obr. 9 - Odpovědní rámeček R3 [1]

### Význam bitů v OCR registru:

- 31 – Stav napájení karty
- 30 – Kapacitní status karty (CCS) určuje, zda jde o kartu typu SDSC nebo SDHC/SDXC
- 29 - 25 – Rezervováno
- 24 – Akceptováno přepnutí na napájení 1,8 V
- 23 – 3,6 V – 3,5 V
- 22 – 3,5 V – 3,4 V
- 21 – 3,4 V – 3,3 V
- 20 – 3,3 V – 3,2 V
- 19 – 3,2 V – 3,1 V
- 18 – 3,1 V – 3,0 V
- 17 – 3,0 V – 2,9 V
- 16 – 2,9 V – 2,8 V

15 – 2,8 V – 2,7 V

14 - 8 – Rezervováno

7 – Rezervováno pro rozsah nízkého napájení

6 - 0 – Rezervováno [1]

### 1.10.4 Odpovědní rámec R7

Karta posílá odpovědní rámec R7, když byl přijat příkaz SEND\_IF\_COND (CMD8). Jeho délka je 5 bajtů. První bajt je shodný s R1. Další čtyři bajty obsahují verzi příkazu, informace o akceptovaném napětí a kontrolní vzorek posílaný společně s CMD8. [1]



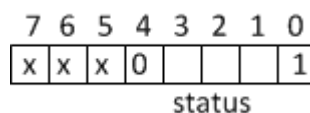
Obr. 10 - Odpovědní rámec R7 [1]

Tab. 7 - Hodnoty akceptovaného napětí pro odpovědní rámec R7 [1]

Bitové hodnoty akceptovaného napětí	Definice hodnoty
0000b	Nedefinováno
0001b	2,7 V – 3,6 V
0010b	Rezervováno pro nízkonapěťový rozsah
0100b	Rezervováno
1000b	Rezervováno
ostatní	Nedefinováno

### 1.10.5 Datová odezva

Všechny bloky dat zapsané na kartu jsou potvrzeny rámcem datové odezvy (data response token), který je dlouhý jeden bajt. Určuje, zda jsou data přijata, data odmítnuta z důvodu chybného kontrolního součtu CRC nebo data odmítnuta z důvodu chyby při zápisu. [1]



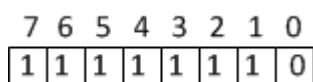
Obr. 11 - Rámec datové odezvy [1]

**Tab. 8 - Definice status bitů [1]**

Možné stavy status bitů	Význam status bitů
010b	Data přijata
101b	Data odmítnuta kvůli chybě CRC
110b	Data odmítnuta kvůli chybě při zápisu

### **1.10.6 Start Bajt**

Každý přenášený blok dat se skládá z 515 bajtů. Prvním bajtem je start bajt, který je pro čtení i zápis jednoho bloku dat pevně definován, viz: obr. 12. Poté následuje 512 bajtů uživatelských dat. Poslední dva bajty obsahují 16-bitové CRC16.



**Obr. 12 - Definice Start bajtu [1]**

## 2 Velkokapacitní zařízení

Základní částí celého velkokapacitního zařízení je obvod FPGA Spartan-3 XC3S200. Tento obvod bude zajišťovat komunikaci mezi úložnou kapacitou a ostatními perifériemi. Komunikace bude probíhat pomocí UART a SPI komunikačních protokolů. Data se budou ukládat na SD kartu. Pro funkčnost samotného obvodu FPGA a pro zajištění další komunikace jsou potřebné další obvodové prvky, které jsou obsaženy na vývojové desce Xilinx Spartan-3 Starter Kit board, jehož popis je v kapitole 2.2.

### 2.1 Obvod FPGA Spartan-3

Obvody FPGA jsou programovatelná hradlová pole. Obvod řady Spartan-3 obsahuje ekvivalentní hradla v počtu od 50000 do pěti milionů.

#### Hlavními rysy architektury Spartan-3:

- 90 nm výrobní technologie
- až pět milionů systémových hradel nebo 74880 logických buněk
- 8 globálních hodinových linek
- plná podpora vývojového prostředí Xilinx ISE

Obvod FPGA Spartan-3 XC3S200 konkrétně obsahuje 200000 ekvivalentních hradel a 4320 logických buněk. [5]

### 2.2 Vývojová deska Xilinx Spartan-3 Starter Kit board

Vývojová deska Xilinx Spartan-3 Starter Kit board, zobrazená na obr. 13, poskytuje výkonnou vývojovou platformu pro experimentování s jakoukoli novou konstrukcí, od jednoduchých logických obvodů až po celé procesorové jádro. Je vybavena obvodem FPGA Spartan-3 XC3S200 od firmy Xilinx.

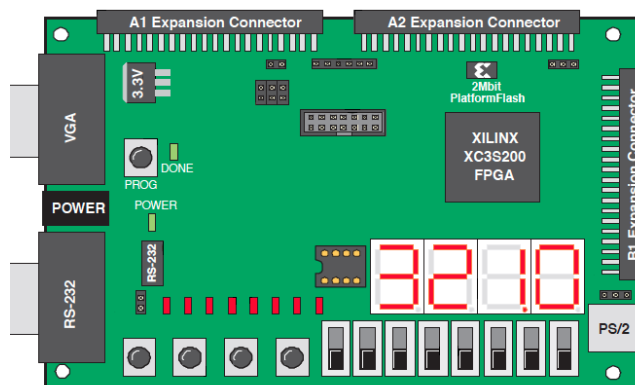
Deska také obsahuje Flash JTAG, programovatelnou konfigurační PROM paměť o velikosti 2 Mbit, do níž lze návrhy nahrávat. Spartan-3 Starter Board je plně kompatibilní se všemi verzemi vývojového nástroje Xilinx ISE.

#### Na desce se nachází:

- 8 přepínačů
- 4 tlačítka
- 8 individuálních LED diodových výstupů



- čtyřmístný sedmi-segmentový displej
- 9-pinový sériový port
- VGA port
- port PS/2 pro připojení klávesnice nebo myši
- tři 40-pinové rozšiřující konektory (A1, A2, B1)
- tři regulátory napětí (3,3V, 2,5V, 1,2V). [6]



Obr. 13 - Vývojová deska Xilinx Spartan-3 Starter Kit board [6]

## 2.3 Komunikace zařízení

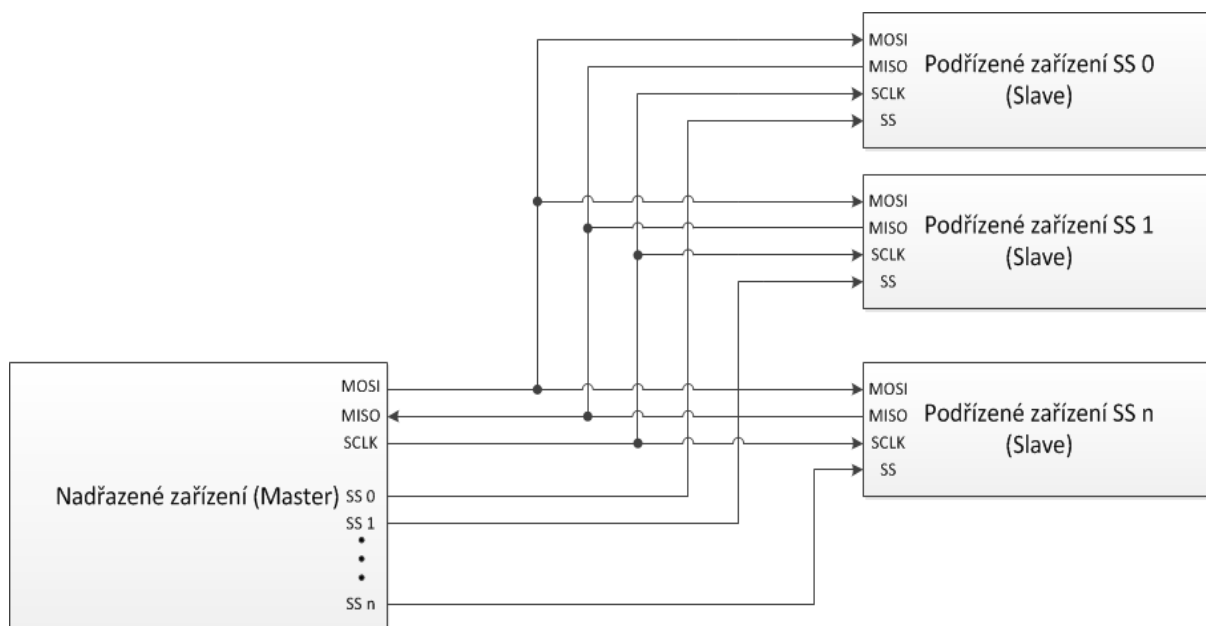
Vývojová deska Xilinx Spartan-3 Starter Kit board bude komunikovat s SD kartou pomocí protokolu pro komunikaci v SPI režimu nebo v SD režimu. Tato vývojová deska neobsahuje slot na SD kartu. Tento slot k ní bude připojen pomocí navrženého rozšiřujícího modulu, který se bude připojovat na rozšiřující port vývojové desky A2, popis navrženého modulu je v kapitole 3.

Vývojová deska bude také komunikovat s počítačem pomocí komunikačního protokolu UART. Komunikaci bude probíhat přes USB port počítače, jelikož vývojová deska neobsahuje USB port, bude navrhovaný modul obsahovat také tento port. Pro převedení USB na UART rozhraní bude použit obvod FTDI FT232RL, který je podrobněji popsán v kapitole 3.1.

### 2.3.1 SPI komunikace

SPI (Serial Peripheral Interface) je synchronní sériové rozhraní, využívající ke své funkci čtyř vodičů. Pomocí SPI rozhraní spolu komunikují dvě nebo více zařízení. Jedno ze zařízení, obvykle procesor, zde Spartan-3, má funkci nadřazeného zařízení (Master), ostatní zařízení jsou jemu podřízená (Slave). Jednotlivá zařízení mají přiveden datový vodič MOSI (Master Out, Slave In), který je přiveden z výstupu zařízení Master na všechna zařízení Slave.

Datový vodič MISO (Master In, Slave Out) je přiveden ze všech výstupů zařízení Slave na vstup Master zařízení. Vodič hodinového signálu SCLK, generovaného zařízením Master, je přiveden na všechna Slave zařízení. Každé zařízení Slave má vstup SS (Slave Select) sloužící k výběru zařízení. Z každého Slave zařízení je veden samostatný vodič SS do zařízení Master. Díky tomuto samostatnému vodiči lze snadno vybírat zařízení, se kterým má být vedena komunikace. Základní koncept SPI rozhraní je zobrazen na obr. 14.

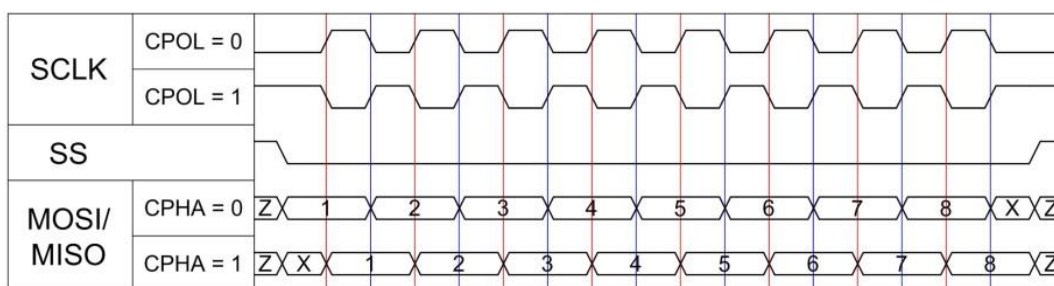


**Obr. 14 - Základní koncepce SPI rozhraní [8]**

Pro zahájení komunikace nastaví Master logickou hodnotu „0“ na SS pinu zařízení. Při více připojených zařízeních nastavuje logickou hodnotu „0“ pouze u zařízení, se kterým chce komunikovat. Začne generovat hodinový signál SCLK. Hodinový signál slouží k synchronizaci vysílání i příjmu dat.

Jakmile jsou data vyslána, může komunikace dále pokračovat nebo může být komunikace ukončena.

Vztah mezi hodinovým signálem a daty se určuje dvěma konfiguračními bity, které se označují jako CPOL a CPHA. Jedná se o bity určující polaritu hodinového signálu (CPOL) a hrana hodinového signálu, při které mají být data čtena (CPHA). Zobrazeno na obr. 15. [8]

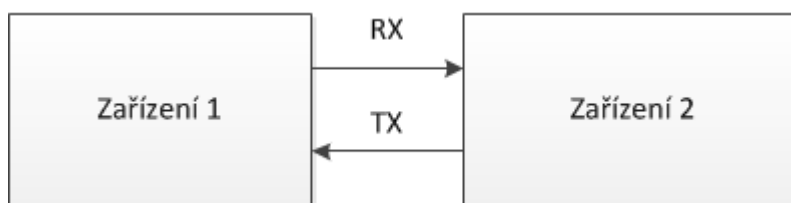


Obr. 15 - Časový diagram SPI [8]

- CPOL = 0; hodinový signál začíná nástupnou hranou
- CPOL = 1; hodinový signál začíná sestupnou hranou
- CPHA = 0; hodnota je čtena při nástupné hraně hodinového signálu (červená svislá čára: data se čtou, modrá svislá čára: data se mění)
- CPHA = 1; hodnota je čtena při sestupné hraně hodinového signálu (modrá svislá čára: data se čtou, červená svislá čára: data se mění) [8]

### 2.3.2 UART komunikace

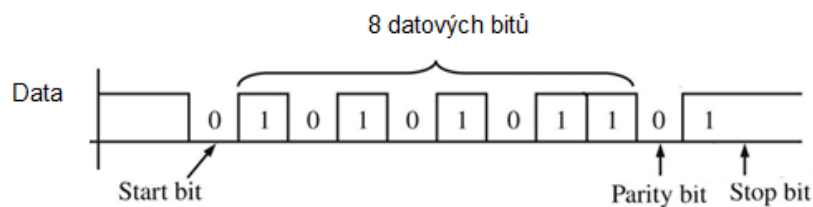
UART (Universal Asynchronous Receiver Transmitter) je rozhraní, které vysílá data přes sériovou linku. UART obsahuje přijímač, vysílač a generátor přenosové rychlosti. Vysílač je v podstatě speciální posuvný registr, který načte paralelně data a po té je bit po bitu přeneseno určitou přenosovou rychlostí. Odesílaná data přenáší sériově na TX pinu. Přijímač načítá data bit po bitu a složí je v přijatá data. Přijímá na RX pinu, která odešle do zařízení paralelně. Zobrazení základního zapojení dvou zařízení je na obr. 16.



Obr. 16 - Zobrazení dvou zařízení pro komunikaci pomocí UART rozhraní [10]

Pokud se neposílají data, mají datové vodiče úroveň logická „1“. Pro spuštění komunikace vyšleme start bit, což je jeden bit s hodnotou logická „0“. Poté jsou zaslány jednotlivé bity dat, každý bit je přenášen stejnou dobu jako všechny ostatní bity. Po odeslání posledního bitu se odešle kontrolní paritní bit a poté stop bit, což je charakterizováno přepnutím datového vodiče na logickou úroveň „1“. Volitelný paritní bit se používá k detekci chyb. Hodnota paritního bitu závisí na obsahu dat. Jestliže je celkový počet jedniček sudý, je hodnota „1“. Je-li lichý, má hodnotu „0“. Uvedený popis paritního bitu platí pro lichou paritu,

pro sudou paritu by byl obráceně. Počet stop bitů může být 1 bit, 1,5 bitu, 2 bity. Posílaný datový rámeček je na obr. 17.



**Obr. 17 - Datový rámeček UART komunikace s jedním start bitem 8 datovými bity paritním bitem a jedním stop bitem [10]**

Protože se neposílá pomocí sériové linky žádná informace o hodinovém signálu, musí se před zahájením přenosu dohodnout nastavení parametrů vysílače a přijímače. Tyto parametry jsou přenosová rychlost (tj. počet bitů za sekundu), počet datových bitů, délka stop bitu (16, 24 nebo 32 hodinových cyklů) a použití paritního bitu. Běžně používané rychlosti jsou 2400, 4800, 9600 19200, 38400, 57600, 115200 baud (1 baud[Bd] = 1 bit/s). [10]

### 3 Návrh obvodového zapojení modulu

Návrh obvodového zapojení rozšiřujícího modulu se slotem pro SD kartu a s USB konektorem, který slouží pro připojení k počítači. Schéma tohoto návrhu bylo vytvořeno v programu Eagle 6.1.0. Schéma je přiloženo v příloze č. 1. Propojení s vývojovou deskou je zajištěno 18-pinovým konektorem, který má vývody zahnuté v 90° úhlu. Pro připojení USB je použit konektor USB typu B. Dále byl použit obvod FT232RL jako převodník z USB na UART rozhraní, který má pouzdro SSOP 28, což je SMD pouzdro s 28 piny. Slot pro SD kartu je taktéž vyveden v SMD provedení. Použité pasivní součástky mají použita SMD pouzdra 1206. Rozmístění vodičů na konektoru je zaznačeno v tab. 9.

Navrhovaný obvod obsahuje jumper, kterým se dá zvolit, zda bude obvod FT232RL napájen přímo z USB počítače nebo z vývojové desky. K tomuto obvodu jsou připojeny dvě signalizační LED diody, které indikují komunikaci na sériové lince. LED1 indikuje komunikaci na pinu TXD a LED2 indikuje komunikaci na pinu RXD. Pro stabilizaci napájení jsou použity tři kondenzátory C1, C2, C3.

Slot pro SD kartu je připojen podle tab. 9 na konektor. Protože SD karta využívá napětí 3,3 V, je její napájecí pin přiveden na pin konektoru, kde vývojová deska nabízí možnost napájení 3,3 V. Blokovací kondenzátor C4 má hodnotu 4,7  $\mu$ F. Na datové vodiče a vodiče CD (Card detect) a WP (Write protekt) bylo přes pull-up rezistory přivedeno napájecí napětí 3,3 V. Vodič CD (Card detect) slouží k fyzické detekci vložení karty a vodič WP (Write protekt) slouží ke kontrole, jestli není karta fyzicky přepnuta do stavu, kdy na ni nelze zapisovat.

**Tab. 9 - Rozmístění a přiřazení pinů na konektoru**

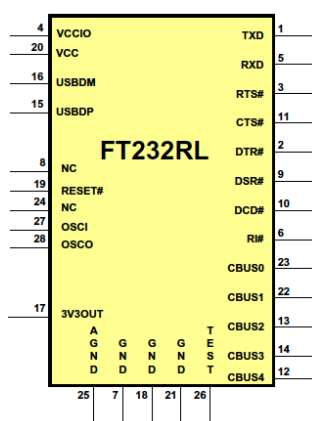
Odpovídající pin na portu A2	Název pinu	Číslo pinu na konektoru		Název pinu	Odpovídající pin na portu A2
VU (+5V)	SYS5V	18	17	GND	GND
PA-IO1 (E6)	n/a	16	15	SYS3,3V	Vcco (+3,3V)
PA-IO3 (C5)	TXD	14	13	RST	PA-IO2 (D5)
PA-IO5 (C6)	RXD	12	11	CST	PA-IO4 (D6)
PA-IO7 (C7)	PWREN	10	9	CLKOUT	PA-IO6 (E7)
PA-IO9 (C8)	DAT2	8	7	CHIP_SEL	PA-IO8 (D7)
PA-IO11 (C9)	MOSI	6	5	MISO	PA-IO10 (D8)
PA-IO13 (A3)	SCLK	4	3	DAT1	PA-IO12 (D10)
PA-IO15 (A4)	WRITE_PRO	2	1	CARD_DET	PA-IO14 (B4)

### 3.1 Obvod FTDI FT232RL

Obvod FT232RL je převodník z USB na UART rozhraní. Komunikace je zajištěna třemi hlavními vodiči RXD (příjem), TXD (vysílání) a společnou zemí GND. Obvod může pracovat jak s 5 V, 3,3 V, 2,8 V tak i 1,8 V logikou. Napájecí napětí je 3,3 - 5,25 V a je možné napájet přímo z USB.

Největší výhodou obvodu FT232RL je, že ke svému fungování potřebuje minimum externích součástek. Obvod má integrovány hodiny (6, 12, 24 a 48 MHz) a také 1024 bitů interní EEPROM obsahující konfigurační data.

Zapojení jednotlivých vývodů je zobrazeno na obr. 18. Pro připojení obvodu k vývojové desce je použito nejčastěji používané zapojení nacházející se v katalogovém listu k tomuto obvodu. [7]

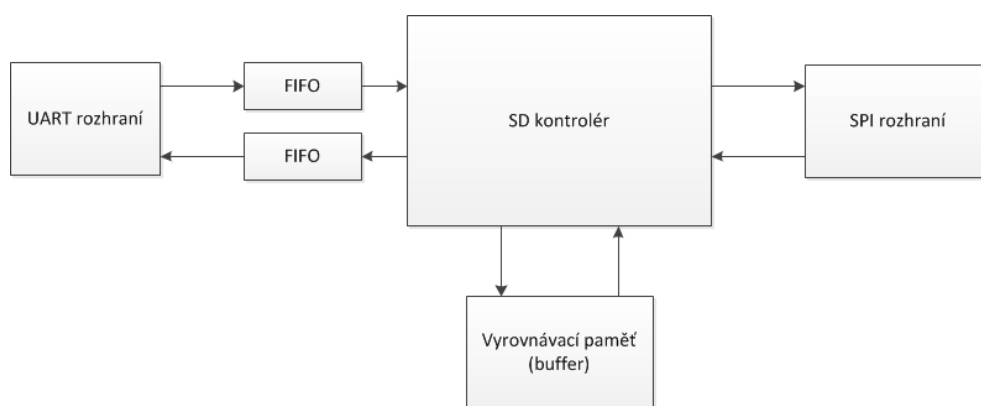


Obr. 18 - Jednotlivé vývody obvodu FT232RL [7]

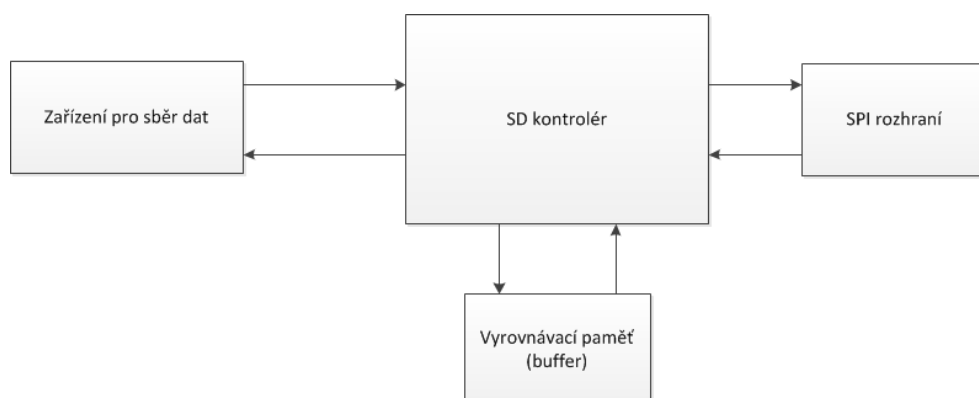
## 4 Návrh vnitřního zapojení

Návrh vnitřního zapojení určeného pro obvod FPGA XC3S200 bude mít na starost zajištění zápisu a čtení dat na paměťové zařízení. Tato komunikace bude zajišťována pomocí SPI rozhraní. Pro urychlení práce s paměťovým zařízením bude hlavní kontrolér obsahovat vyrovnávací paměť (buffer), do kterého se budou zapisovat načítaná data z paměťové karty, nebo se z něj budou data postupně posílat na paměťovou kartu. Kontrolér zajišťuje také inicializaci karty. Inicializace probíhá na frekvenci od 100 kHz – 400 kHz. Po inicializaci můžeme frekvenci pro zápis i čtení zvýšit na maximum.

Komunikace s počítačem bude v návrhu zahrnuta v bloku UART rozhraní. Pomocí tohoto rozhraní budou data posílána do vyrovnávací paměti kontroléru a po té kontrolér odešle data do paměťové karty, případně se budou data z paměťové karty číst. Následně odešle kontrolér data do počítače. Blokové zapojení programové části je na obr. 19. Návrh by měl komunikovat i se zařízeními pro sběr dat. Tento návrh je na obr. 20.



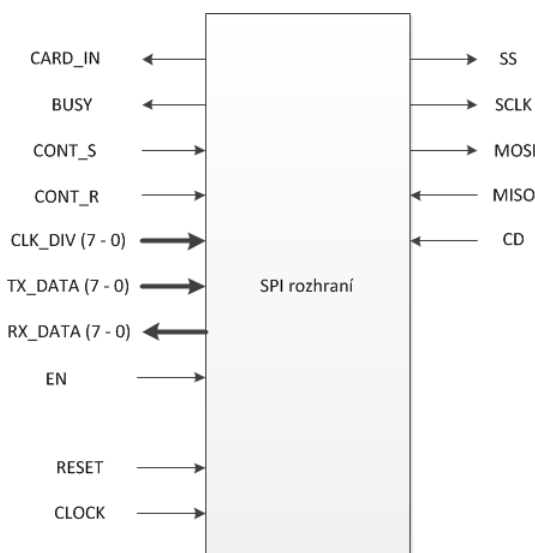
Obr. 19 - Blokové schéma programové části pro komunikaci s počítačem



Obr. 20 - Blokové schéma programové části pro komunikaci se zařízením pro sběr dat

## 4.1 Blok SPI rozhraní

SPI rozhraní zajišťuje komunikaci s paměťovou kartou. Je spojen s blokem SD kontroléru, který posílá data, jež se mají zapsat na kartu. Nebo načítá data, která by měla být přečtena z karty. Dále přijímá odpovědní rámce z karty a ty posílá kontroléru na zpracování a vysílá příkazy, které určují, co se má provést. Kontrolér posílá data, která mají být poslána kartě, po jednom bajtu. Tato data rozhraní dále odesílá bit po bitu na kartu přes signálový vodič MOSI. Data od karty přijímaná rozhraním přes signál MISO se po přijetí osmého bitu odešlou do kontroléru paralelně jako jeden bajt. Rozhraní zajišťuje nastavení hodnoty na SS pinu pro aktivování komunikace s kartou. Zajišťuje také generování hodinového signálu SCLK. Nákres bloku rozhraní je na obr. 21.



Obr. 21 - Blok SPI rozhraní

Signálové vodiče mezi kontrolérem a SPI rozhraním. Signál „card\_in“ slouží k detekování vložené karty, „busy“ slouží k určení, kdy je rozhraní zaneprázdněno a kdy je připraveno odeslat nebo přijmout další bajt dat, „cont\_s“ a „cont\_r“ nastavuje příznak pokračování v odesílání/přijímání dalšího bajtu. Sběrnice „clk\_div“ určuje hodnotu děličky systémového hodinového signálu, pomocí sběrnic „rx\_data“ a „tx\_data“ se načítají a odesílají data k dalšímu zpracování v bloku kontroléru, „en“ signál spouští komunikaci přes rozhraní SPI. Do bloku je přiveden asynchronní reset signál a signál „clock“, což je systémový hodinový signál.

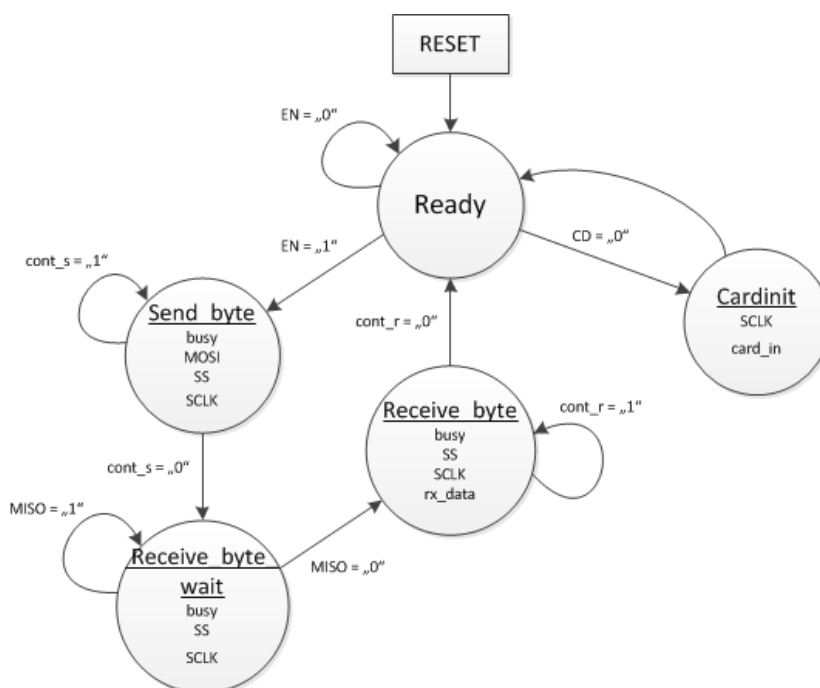
Mezi SPI rozhraním a kartou jsou vedeny signálové vodiče SS pro výběr a aktivaci karty pro komunikaci, SCLK hodinový signál pro synchronizaci přenosu dat, MOSI a MISO



signály pro odesílání a příjem jednotlivých bitů a signál CD, který detekuje fyzické vložení karty do slotu.

#### 4.1.1 Vnitřní návrh SPI rozhraní

Vnitřní návrh SPI rozhraní je tvořen stavovým automatem, který je znázorněn na obr. 22. Tento stavový automat obsahuje 6 stavů, které budou dále podrobně popsány. Jedná se o upravené rozhraní pro komunikaci s SD kartou.



Obr. 22 - Stavový automat bloku SPI rozhraní

Při resetování rozhraní se nastaví výchozí hodnoty. To znamená nastavení výstupního MOSI signálu do vysoké impedance, vynulování čítačů hodinového signálu, uvolnění všech podřízených zařízení a vymaže se přijímací port. Následuje přesunutí do stavu „ready“. Stav „send\_byte“, „receive\_byte\_wait“, „receive\_byte“ a „cardinit“ obsahují děličku systémového hodinového signálu pro generaci hodinového signálu SCLK.

Stav „ready“ signalizuje kontroléru, že je připraveno na komunikaci. Následně čeká na signál „EN“, který zahájí odesílání dat. Příjmutím signálu „EN“ se načte bajt dat posílaný kontrolérem, načte se hodnota děličky hodinového signálu a přesune se do stavu „send\_byte“. Ve stavu „ready“ se kontroluje i signál „CD“. Vložení karty a jejím detekováním se přesune stavový automat na stav „cardinit“. Nastaví se parametry děličky hodinového signálu na hodnotu 125, což odpovídá výstupní frekvenci SCLK = 200 kHz, hodnota je zvolena pro celou inicializaci a vypočtena ze vztahu (1).

Stav „send\_byte“ funguje na principu jednobitového posuvného registru. Odesílá vždy nejvyšší bit po signálovém vodiči MOSI na kartu. Vybere se podřízené zařízení nastavením SS signálu na hodnotu „0“. Pokud je přijímán znak pro pokračování přenosu „cont\_s“, tak se po odeslání posledního bitu dat načte nový bajt z kontroléru. Pokud není přijímán znak pro pokračování, ukončí se po odeslání posledního bitu přenos, MOSI se nastaví do vysoké impedance a přesune se na stav „receive\_byte\_wait“.

Stav „receive\_byte\_wait“ udržuje signál SS na hodnotě „0“. Vysílá hodinový signál SCLK. Přijímá hodnotu MISO signálového vodiče. Pokud přijme na signálovém vodiči MISO hodnotu „0“, přesune se do stavu „receive\_byte“.

Stav „receive\_byte“ funguje podobně jako stav „send\_byte“. Přijímá bity ze signálového vodiče MISO. Pokud je přijímán znak pro pokračování přenosu „cont\_r“, odešle po přijmutí posledního bitu data do kontroléru a pokračuje v přijímání následujícího bajtu dat. Není-li znak pro pokračování přijímán, ukončí se po odeslání načtených dat do kontroléru přenos. Nastavením signálu SS na hodnotu „1“ přestane se odesílat hodinový signál SCLK. Pokračuje návratem do stavu „ready“.

Stav „cardinit“ slouží k tak zvanému vyrovnání a stabilizaci napájecích a datových úrovní. Toho je dosaženo pomocí odeslání 80 hodinových cyklů na SCLK o frekvenci, kterou se bude realizovat následná inicializace. Bez vybrání podřízeného zařízení.

Výpočet hodnoty děličky systémového hodinového signálu je definován následujícím vztahem (1)

$$f_{SCLK} = \frac{f_{clock}}{2 \times clk\_div} \quad (1)$$

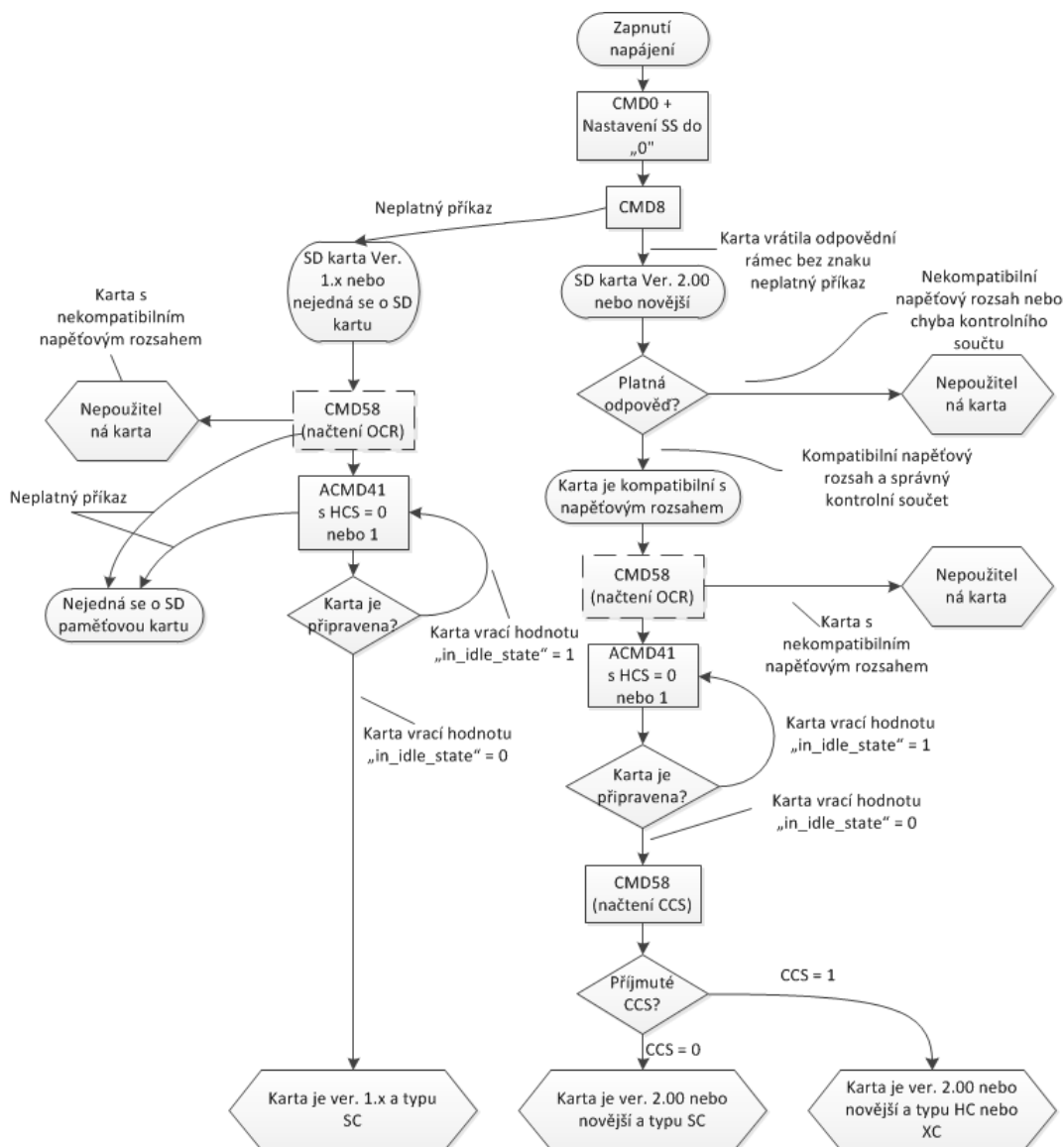
kde  $f_{SCLK}$  představuje výstupní frekvenci pro kartu,  $f_{clock}$  frekvenci systémového hodinového signálu známe, pro desku Spartan-3 má hodnotu 50 MHz,  $clk\_div$  hodnota děličky.



Obr. 23 - Přenos příkazu CMD8 pomocí SPI rozhraní. Výstup z logického analyzátoru

## 4.2 Blok SD kontrolér

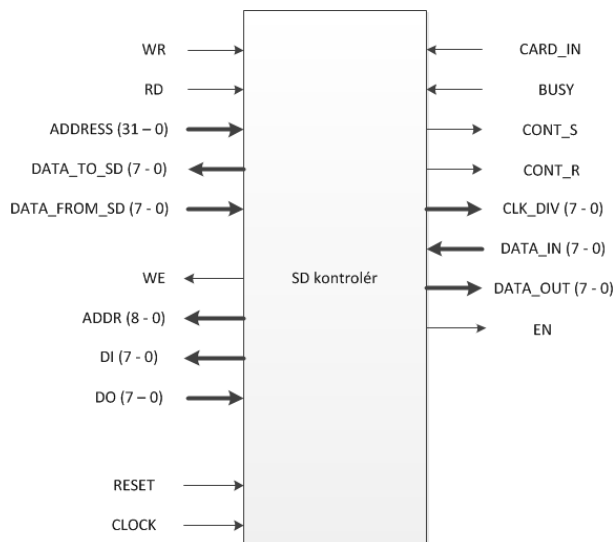
SD kontrolér obstarává odesílání a přijímání bajtů dat z SPI rozhraní. Odesílá příkazové rámce a vyhodnocuje odpovědní rámce. Zajišťuje inicializaci SD karty po jejím vložení, odesláním příkazů k inicializaci a vyhodnocením odpovědí od karty. Inicializace vychází z vývojového diagramu vyobrazeného na obr. 24.



Obr. 24 - Inicializační proces [1]

Zajišťuje také zápis a čtení bloku dat z karty. Odešle příkaz pro zápis bloku dat (CMD24) a následně odešle jeden blok dat na kartu. Data určená k zapsání se odesílají z bloku „buffer“ po jednotlivých bajtech. Nebo odešle příkaz pro čtení bloku dat (CMD17) a

následně načte přijímaná data. Přijímaná data se po jednom bajtu zapisují do bloku „buffer“. Pomocí bloku SD kontroléru se přijímají a odesílají data, která mají být zapsána nebo vyčtena z karty z dalších bloků, např. z bloku UART. Nákres bloku SD kontrolér je na obr. 25.

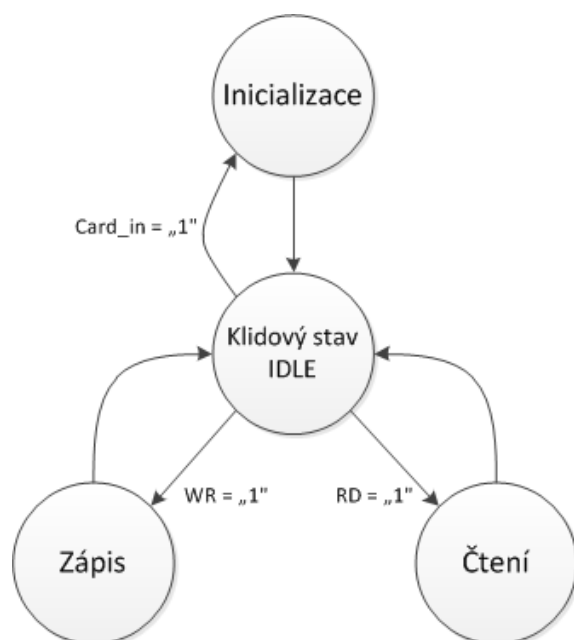


**Obr. 25 - Blok SD kontrolér**

SD kontrolér obsahuje signálové vodiče pro komunikaci s blokem SPI rozhraní, také obsahuje signálové vodiče pro komunikaci s blokem „buffer“. Signálový vodič povolující zápis dat do „bufferu“ (WE), adresní sběrnice (ADDR), určující adresu zapisovaného nebo čteného bajtu dat a dvě datové sběrnice zajišťující zápis a příjem dat „bufferu“ (DI, DO). Dále obsahuje sběrnice pro data, která mají být zapsána na kartu nebo mají být z karty přečtena (DATA\_TO\_SD, DATA\_FROM\_SD). Adresovou sběrnici pro adresování dat na kartě (ADDRESS). Dva signálové vodiče rozhodující, zda má být proveden zápis nebo čtení dat (WR, RD). Do bloku je přiveden systémový hodinový signál (clock) a asynchronní reset.

#### **4.2.1 Vnitřní návrh bloku SD kontrolér a popis klidového stavu**

Návrh vnitřního zapojení bloku SD kontrolér je rozdělen do čtyř částí. Skládá se z částí klidového stavu (IDLE), inicializace, zápis dat a čtení dat. Vyobrazení těchto částí je na obr. 26. Celý návrh sestává ze stavového automatu, který bude postupně po jednotlivých částech popsán.

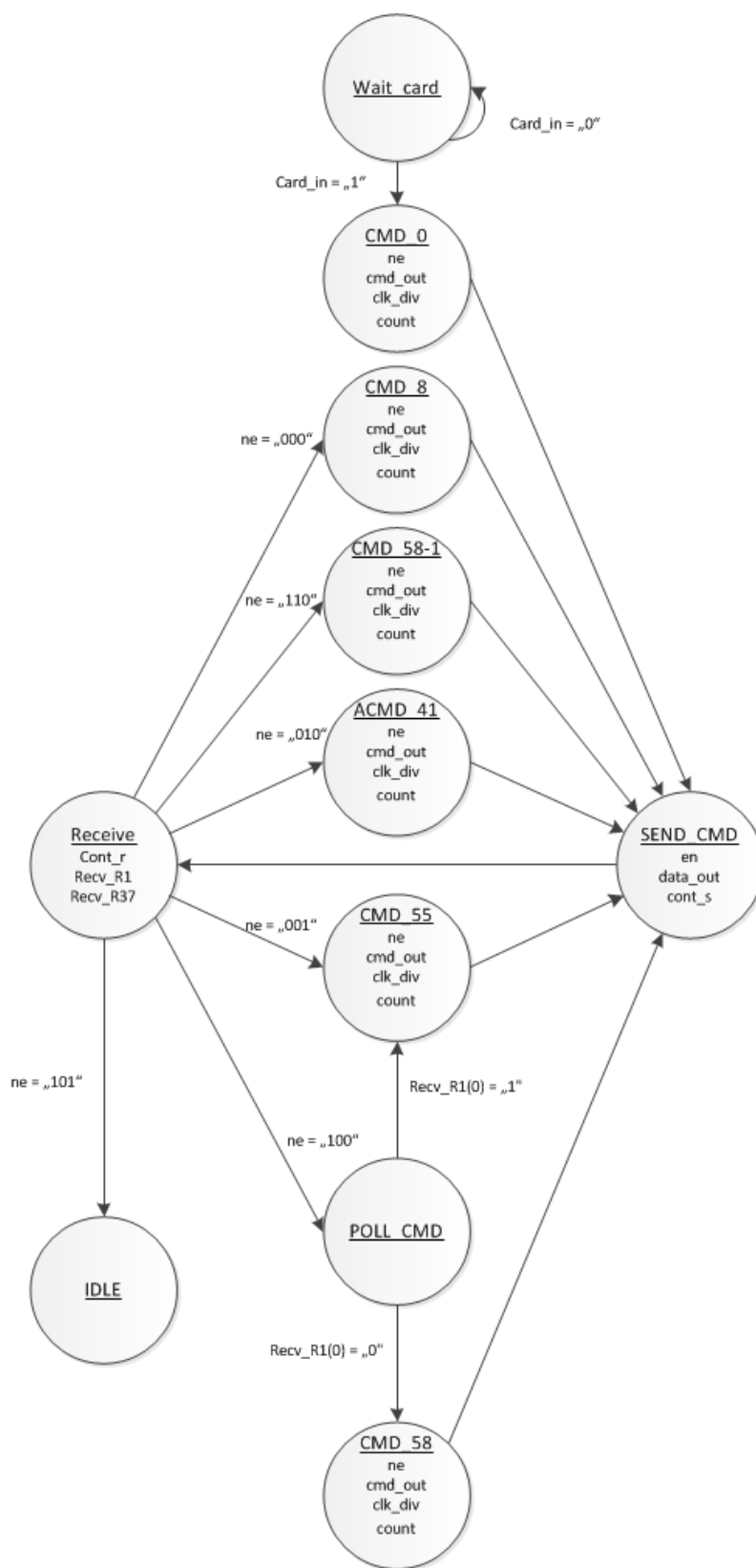


**Obr. 26 - Základní rozdělení SD kontroléru**

V klidovém stavu (IDLE) čeká stavový automat na příchozí signál, podle kterého provede následující operace. Při přijmutí signálu z bloku SPI rozhraní, karta byla vložena („card\_in“ = „1“), provede inicializaci a navrátí se zpět. Poté začíná přijímat signály i na signálových vodičích pro zápis a čtení dat (WR, RD). Přijmutím signálu WR = „1“ se přesune do části, pro zápis dat na kartu. Přijmutím signálu RD = „1“ se přesune do části, pro čtení dat z karty.

#### **4.2.2 Inicializace karty**

Všechny SD karty jsou po připojení k napájení standardně v SD režimu, proto je třeba přepnout kartu do SPI režimu. Přepnutí se provádí pomocí resetovacího příkazu (CMD0), společně s nastavením výběrového signálového vodiče (SS) na hodnotu 0. Po přepnutí do SPI režimu je třeba kartu inicializovat pomocí příkazu pro inicializaci v SPI režimu z důvodu nekompatibility příkazů používaných a inicializaci v SD režimu. Inicializací se nastavuje napájecí rozsah karty a zjišťuje se verze a typ karty. Stavový automat navrhované inicializace je zobrazen na obr. 27. Stavový automat byl navržen podle vývojového diagramu zobrazeného na obr. 24. Inicializace karty probíhá při frekvenci  $f_{SCLK} = 200 \text{ kHz}$ , což odpovídá nastavení děličky „clk\_div“ na hodnotu 125. Tuto hodnotu dostaneme výpočtem výrazu (1).



Obr. 27 - Stavový diagram inicializačního procesu

Stav „wait\_card“ ověřuje vložení karty a její základní inicializování. Neboli odeslání 80 hodinových cyklů, kvůli vyrovnání a stabilizaci napájecích a datových úrovní. Je-li karta vložena, přepne se do stavu „CMD\_0“. Není-li karta vložena, setrvá ve stavu „wait\_card“.

Stav „CMD\_0“ slouží k resetování karty a k jejímu přepnutí do SPI režimu. Nastaví se hodnota 6 bajtů příkazu (CMD0) do pomocného signálu „cmd\_out“, hodnota pro děličku SPI rozhraní „clk\_div“, počítadlo délky odpovědi na příjem pouze jednoho bajtu odpovědního rámce R1 a návratový status („ne“ = „000“) odpovídá stavu „CMD8“. Následuje přesun do stavu „SEND\_CMD“.

Stav „CMD\_8“ zjišťuje jaké verze je vložená karta a její napájecí rozsah. Zjištění verze karty zajišťuje tím, že karty verze 1.xx tento příkaz neznají a vyhodnotí ho jako chybný. Na rozdíl od karet verze 2.0 a novějších, které ho již znají. Nastaví se hodnota příkazu (CMD8 + platný CRC7, který je u toho příkazu vyžadován i v SPI režimu) do „cmd\_out“, počítadlo délky odpovědi je zde 5 bajtů, přijímaný odpovědní rámec je R7, hodnota „clk\_div“ zůstane nezměněna a návratový status („ne“ = „110“) odpovídá stavu „CMD\_58-1“. Následuje přesun do stavu „SEND\_CMD“.

Stav „CMD\_58-1“. Tento stav, ani příkaz, není v tuto chvíli povinný. Příkaz „CMD58“, který je odeslán, zajišťuje vyčtení OCR registru. V tento moment je v OCR registru známa hodnota napětového rozsahu karty, nikoli její typ. Typ karty zjistíme příkazem „CMD58“ až po úspěšně dokončené inicializaci. Nastaví se hodnota příkazu (CMD58) do „cmd\_out“, počítadlo délky odpovědi je zde 5 bajtů, přijímaný odpovědní rámec je R3, hodnota „clk\_div“ zůstane nezměněna a návratový status („ne“ = „001“) odpovídá stavu „CMD\_55“. Následuje přesun do stavu „SEND\_CMD“.

Stav „CMD\_55“. Pomocí příkazu „CMD55“ se přepíná typ dalšího příkazu na aplikační příkaz. Je nutné před každým odesílaným aplikačním příkazem odeslat příkaz „CMD55“. V tomto stavu se nastaví hodnota příkazu (CMD55) do „cmd\_out“, počítadlo délky odpovědi je zde jeden bajt, přijímaný odpovědní rámec je typu R1. Hodnota „clk\_div“ zůstane nezměněna a návratový status („ne“ = „010“) odpovídá stavu „ACMD\_41“. Následuje přesun do stavu „SEND\_CMD“.

Stav „ACMD41“. Jedná se o aplikační příkaz sloužící k dokončení inicializačního procesu. Nastavovaná hodnota příkazu „ACMD41“ je stejná jako u příkazu „CMD41“. Karta nyní očekává aplikační příkaz, z toho důvodu se nemusí nijak měnit příkazový rámec. Počítadlo délky odpovědi je jeden bajt, odpovědní rámec je typu R1. Hodnota „clk\_div“ je stále stejná. Návratový status („ne“ = „100“) odpovídá stavu „POLL\_CMD“. Následuje přesun do stavu „SEND\_CMD“.

Stav „POLL\_CMD“ porovnává poslední bit odpovědního rámce R1. Pokud je poslední bit R1 roven hodnotě „1“, znamená to, že karta není ještě zcela inicializována, následuje přesun do stavu „CMD\_55“. Dojde k zopakování odeslání příkazů „CMD55“ a „ACMD41“. Obdrží-li na posledním bitu R1 hodnotu „0“, značí to, že je karta úspěšně inicializována a můžeme pokračovat načtením OCR registru pomocí příkazu „CMD58“. Následuje tedy přesun do stavu „CMD\_58“.

Stav „CMD\_58“ slouží k načtení OCR registru, kde nás nyní zajímá bit CCS (Card Capacity Status), který určuje, zda se jedná o kartu typu SDSC (CCS = „0“), nebo typu SDHC, popřípadě SDXC (CCS = „1“). Nastaví se hodnota příkazu (CMD58) do „cmd\_out“, počítadlo délky odpovědi je zde 5 bajtů, přijímaný odpovědní rámec je R3, hodnota „clk\_div“ zůstane nezměněna a návratový status („ne“ = „101“) odpovídá stavu „IDLE“. Následuje přesun do stavu „SEND\_CMD“. Přijmutí odpovědi je inicializace úspěšně dokončena.

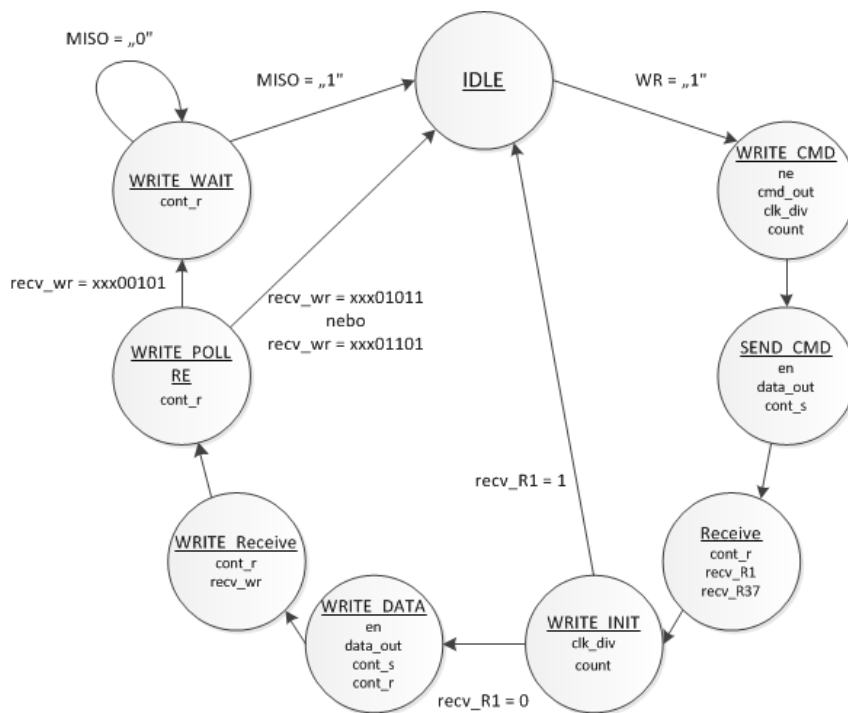
Stav „SEND\_CMD“ slouží k postupnému odesílání jednotlivých bajtů příkazů do bloku SPI rozhraní. Aktivuje přenos vysláním signálu „EN“ s hodnotou „1“. Nastaví znak pro pokračování odesílání, který vynuluje s odesláním posledního bajtu příkazu. Po odeslání celého příkazového rámce následuje přesun do stavu „Receive“.

Stav „Receive“ slouží k přijmutí odpovědního rámce typu R1, R3 a R7. Podle počtu bajtů odpovědního rámce odesílá znak pro pokračování v příjmu. Po přijmutí celého odpovědního rámce se přesuneme do dalšího stavu pomocí předem nastaveného návratového statusu.

### **4.2.3 Zápis dat na kartu**

Do této části navrhovaného popisu se dostaneme, přijde-li znak pro zápis „WR“ = „1“. Načte se adresa (ADDRESS), kam se na kartu bude zapisovat. Postupně se načítají i jednotlivé bajty dat zapisované na kartu. Tyto data se načítají přímo z vnějšího zařízení pro sběr dat nebo z UART rozhraní, nebo jsou načítána z „bufferu“, kam byla předem zapsána. Navržený stavový automat je zobrazen na obr. 28.





**Obr. 28 - Stavový automat pro zápis dat na kartu**

Stav „WRITE\_CMD“ slouží k odeslání příkazu (CMD24). Příkaz „CMD24“ se užívá k zápisu jednoho bloku dat, defaultně nastaveného na velikost 512 B. Argument příkazu je adresa, kam se na kartu bude tento blok zapisovat, délka adresy je 4 bajty. Hodnota příkazu s argumentem se zapíše do „cmd\_out“. Hodnotu děličky „clk\_div“ můžeme nastavit na maximální frekvenci. Při použití 50 MHz vstupního hodinového signálu je maximální hodnota hodinového signálu karty (SCLK) 25 MHz. Odpovědní rámec karty je typu R1, počítadlo délky odpovědi je jeden bajt. Nastaví se návratový status („ne“ = „111“) odpovídající stavu „WRITE\_INIT“. Následuje přesun do stavu „SEND\_CMD“, který byl společně se stavem „Receive“ popsán v kapitole 4.2.2.

Stav „WRITE\_INIT“ vyhodnocuje odpovědní rámec R1 karty. Je-li odpověď karty „recv\_R1“ = „00“h, nastaví se čítač odesílaných dat na hodnotu 515, nastaví se znak pro pokračování a hodnota děličky stejná jako při odesílání příkazu pro zápis. Čítač má hodnotu 515 z důvodu tvaru odesílaného bloku dat. Na pozici 515 se nachází start bajt, poté je 512 bajtů uživatelských dat a jako poslední dva bajty je odesílán CRC16 kontrolní součet, který v režimu SPI má hodnotu FFh. Následuje přesun do stavu „WRITE\_DATA“.

Stav „WRITE\_DATA“ odesílá po jednom bajtu datový rámec obsahující jeden start bajt, 512 bajtů uživatelských dat a dva bajty CRC16 kontrolního součtu do SPI rozhraní. Aktivuje přenos vysláním signálu „EN“ s hodnotou „1“, po celou dobu odesílá také znak pro

pokračování odesílání, který vynuluje s odesláním posledního bajtu. Následuje přesun do stavu „WRITE\_Receive“.

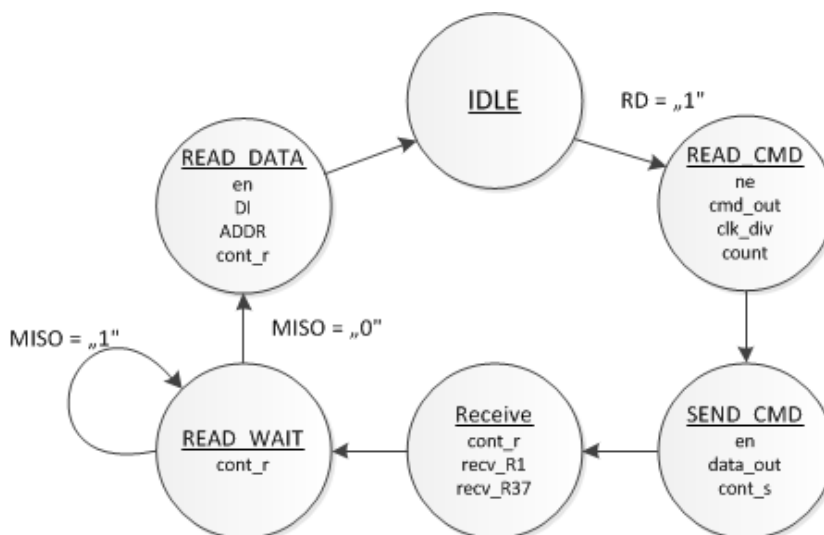
Stav „WRITE\_Receive“ přijme odpovědní rámec datové odezvy. Neustále odesílá znak pro pokračování v přijímání odpovědi. Následuje přesun do stavu „WRITE\_POLL\_RE“.

Stav WRITE\_POLL\_RE“ vyhodnocuje přijatou datovou odezvu. Při příjmu „recv\_wr“= „xxx00101“ pokračuje do stavu „WRITE\_WAIT“. Při přijmutí „recv\_wr“= „xxx01101“ nebo „recv\_wr“= „xxx01011“ odešle chybu a ukončí přenos, následně přejde do stavu „IDLE“. Odpovědní rámec „recv\_wr“= „xxx00101“ znamená data přijata, „recv\_wr“= „xxx01101“ data odmítnuta kvůli chybě při zápisu a „recv\_wr“= „xxx00101“ data odmítnuta kvůli chybě CRC.

Stav „WRITE\_WAIT“ pokračuje v odesílání znaku o pokračování příjmu, čímž se i nadále generuje hodinový signál „SCLK“ a je vybrána karta signálem „SS“. Kontroluje se signálový vodič MISO, který se nachází v hodnotě „0“. Jakmile jsou data zapsána na kartu, přepne karta signál MISO na hodnotu „1“. Tím je ukončen zápis dat a stav přestane odesílat znak pro pokračování příjmu. Následuje přesun do stavu „IDLE“

#### 4.2.4 Čtení dat z karty

Do této části se dostaneme, přijde-li znak pro čtení „RD“ = „1“. Načte se adresa (ADDRESS), odkud se z karty bude číst. Čtená data se postupně zapisují do „bufferu“, odkud budou následně odeslána k dalšímu zpracování do nějakého zařízení nebo přes UART rozhraní do počítače. Navržený stavový automat je zobrazen na obr. 29.



Obr. 29 - Stavový automat pro čtení z karty

Stav „READ\_CMD“ slouží k odeslání příkazu (CMD17). Příkaz „CMD17“ se užívá ke čtení jednoho bloku dat, defaultně nastaveného na velikost 512 B. Argument příkazu je adresa, ze které se bude tento blok dat číst, délka adresy je 4 bajty. Hodnota příkazu s argumentem se zapiší do „cmd\_out“. Hodnotu děličky „clk\_div“ nastavíme na maximální frekvenci. Odpovědní rámec karty je typu R1, počítadlo délky odpovědi je jeden bajt. Nastaví se návratový status („ne“ = „011“) odpovídající stavu „READ\_WAIT“. Následuje přesun do stavu „SEND\_CMD“, který byl společně se stavem „Receive“ popsán v kapitole 4.2.2.

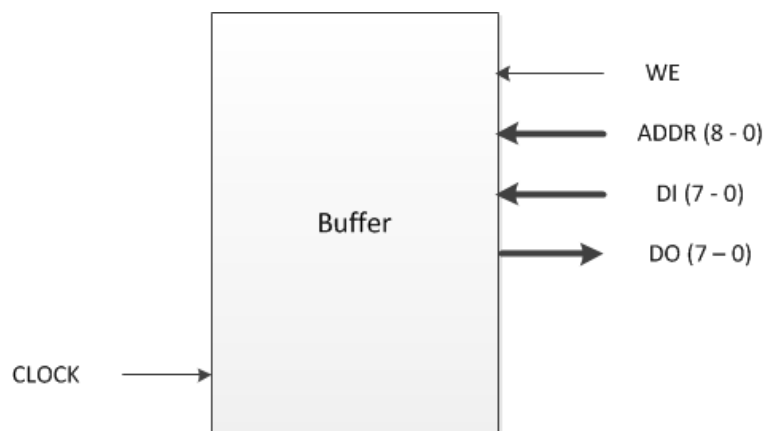
Stav READ\_WAIT čeká na příchozí start bajt od karty, který má tvar FEh. Neustále odesílá znak pokračování příjmu, aby nebyla přerušena komunikace s kartou. Po přijetí start bajtu se nastaví počítadlo přijatých dat na hodnotu 514, ve které je zahrnuto 512 bajtů uživatelských dat a 2 bajty CRC16 kontrolního součtu, který po přijmutí zahodíme. Přesuneme se na stav „READ\_DATA“.

Stav „READ\_DATA“ přijímá z SPI rozhraní jednotlivé bajty, které postupně zapisuje do „bufferu“. Po přijetí 512 bajtů přijme poslední dva bajty CRC16, které zahodí. Po skončení přenosu nastaví znak pro pokračování příjmu na hodnotu „0“. Následuje přesun do stavu „IDLE“.

### 4.3 Blok Buffer

Tento blok slouží pro urychlení komunikace s paměťovou kartou. Nahrávají se do něj data, která mají být zapsána na kartu, nebo se do něj data z karty načítají, aby mohla být odeslána k dalšímu zpracování.

Blok „buffer“ je navržen jako bloková RAM paměť, s velikostí 512 bajtů. Vnitřní popis byl převzat z knihovny příkladů kódů, kterou obsahuje vývojové prostředí Xilinx ISE. Blok je zobrazen na obr. 30.



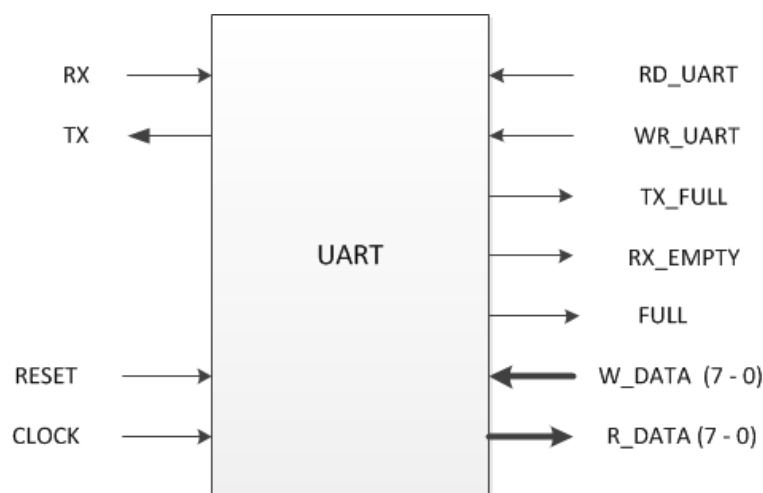
Obr. 30 - Blok Buffer

Do bloku „buffer“ je přiveden systémový hodinový signál (clock), signál pro povolení zápisu (WE), adresní sběrnice (ADDR), datová sběrnice s daty, co se mají zapsat (DI) a datová sběrnice pro čtení dat (DO).

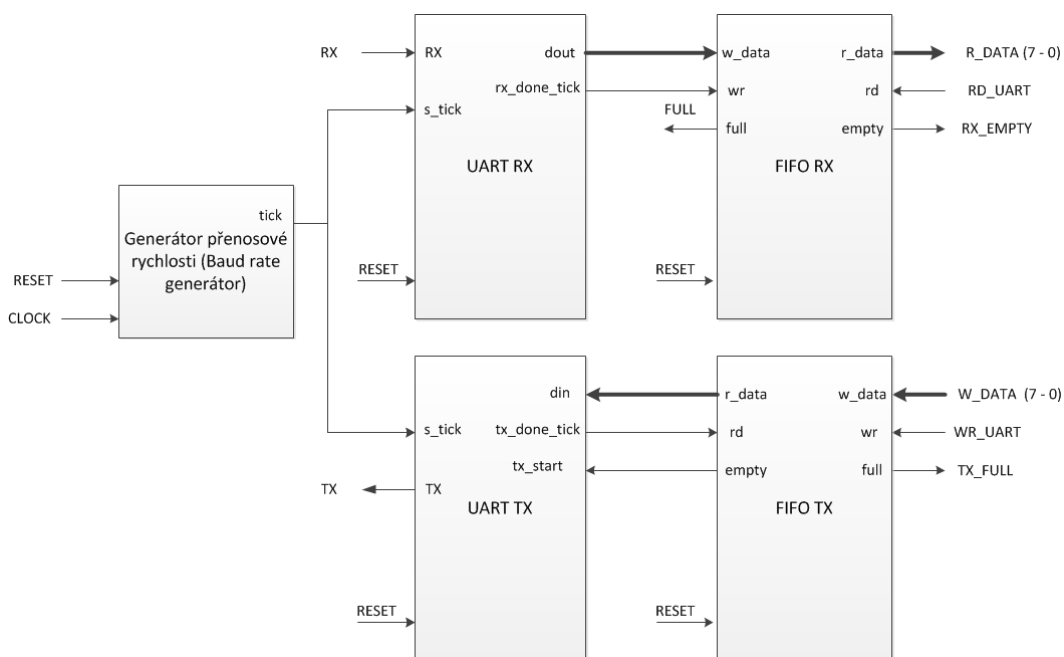
## 4.4 Blok UART

Blok UART zprostředkovává komunikaci s počítačem. Jsou v něm implementovány moduly pro příjem a odesílání dat a také generátor přenosové rychlosti zároveň s moduly FIFO pamětí. S hlavním blokem kontroléru je propojen přes moduly FIFO pamětí. Komunikace s PC je obstarávána přes výstupní signálový vodič TXD a přes vstupní signálový vodič RXD.

FIFO paměti jsou použity pro zaručení bezchybné komunikace mezi počítačem a kontrolérem. Když je kontrolér zaneprázdněn a nemůže se věnovat obsluze UART bloku, jsou data uložena právě do FIFO paměti. Až je možné provést obsluhu tohoto bloku, načtou se data z FIFO paměti, čímž nedojde k jejich ztrátě. Vnitřní návrh bloku UART byl převzat z knihy „FPGA Prototyping by VHDL Examples“ [10]. Celý blok UART je zobrazen na obr. 31. vnitřní blokové schéma je na obr. 32.



Obr. 31 - Blok UART



Obr. 32 - Vnitřní zapojení Bloku UART

#### 4.4.1 Modul generátoru přenosové rychlosti

Generátor přenosové rychlosti generuje vzorkovací signál, jehož frekvence je přesně 16 násobek UART přenosové rychlosti. Pro přenosovou rychlost 9600 Bd musí být vzorkovací frekvence 153600 (9600\*16) hodinových cyklů za sekundu. Hodnota frekvence systémového hodinového signálu je 50 MHz. Výslednou hodnotu děličky systémového hodinového signálu dostaneme dosazením do výrazu (2)

$$div = \frac{f_{CLOCK}}{baudrate * 16} \quad (2)$$

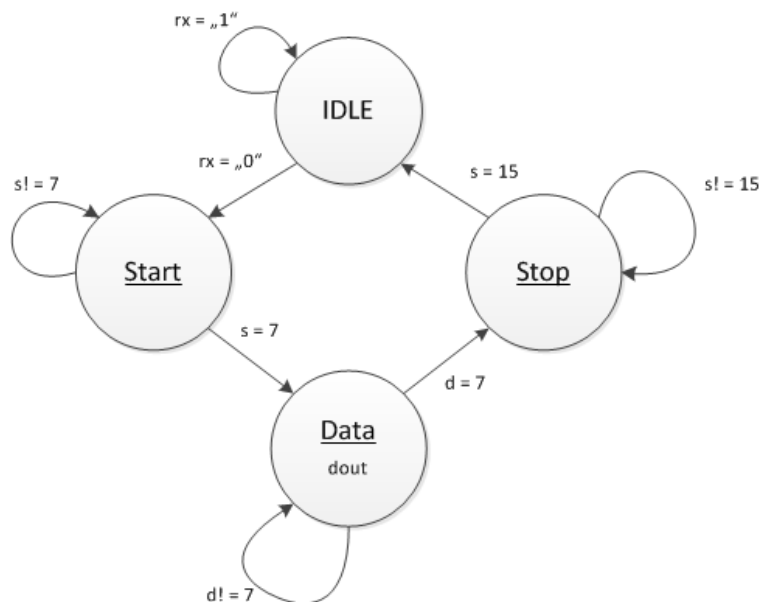
kde div je výsledná hodnota děličky,  $f_{CLOCK}$  je hodnota systémového hodinového signálu a baudrate je požadovaná přenosová rychlost. Hodnota děličky pro 9600 Bd je  $div = 325$ .

#### 4.4.2 Modul UART RX

Modul UART RX přijímá sériově data ze signálového vodiče „RX“ a po přijmutí 8 bitů tato data odešle paralelně po sběrnici „dout“. Vnitřní zapojení modulu UART RX je stavový automat zobrazený na obr. 33.

Stavový automat setrvává ve stavu „IDLE“ do doby, než se objeví na signálovém vodiči „RX“ hodnota „0“. Po té se přesune do stavu „start“, kde je po osmi hodinových cyklech, navzorkován start bit. Pokračuje se na stav „data“. Zde se každých 16 hodinových

cyklů navzorkuje jeden bit dat, celkem je přijato 8 bitů dat, které se odešlou paralelně po sběrnici. Po načtení bajtu dat se přesuneme na stav „stop“, zde se po 16 hodinových cyklech navzorkuje jeden stop bit. Podle počtu hodinových cyklů ve stavu stop rozlišujeme 1, 1,5 a 2 stop bity (16, 24 a 36 hodinových cyklů). Přijmutí je zakončeno návratem do stavu „IDLE“.

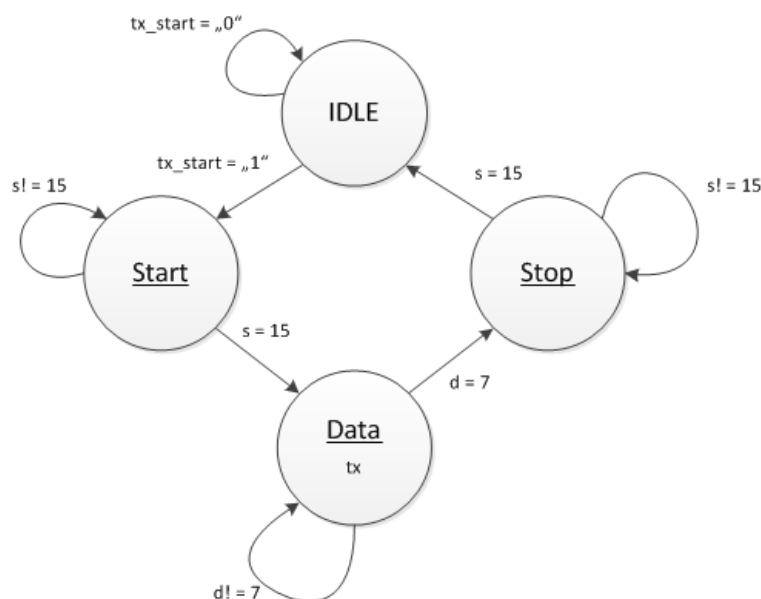


Obr. 33 - Stavový automat UART RX

#### 4.4.3 Modul UART TX

Modul UART TX načte paralelně data ze sběrnice din a odešle je sériově pomocí signálového vodiče „TX“. Nejdříve odešle start bit, po něm následuje 8 bitů dat a následně stop bit. Vnitřní zapojení modulu UART TX je stavový automat zobrazený na obr. 34.

Stavový automat setrvává ve stavu „IDLE“ do doby, než se objeví na signálovém vodiči „tx\_star“ hodnota „1“. Po té se přesune do stavu „start“, kde je po 16 hodinových cyklů, odeslán start bit. Pokračuje se na stav „data“. Zde se každých 16 hodinových cyklů odesílá jeden bit dat, celkem je odesláno 8 bitů dat, které byly paralelně přijaty po sběrnici. Po odeslání bajtu dat se přesuneme na stav „stop“, zde se po 16 hodinových cyklů odesílá stop bit. Podle počtu hodinových cyklů ve stavu stop rozlišujeme 1, 1,5 a 2 stop bity (16, 24 a 36 hodinových cyklů). Odesílání je zakončeno návratem do stavu „IDLE“.

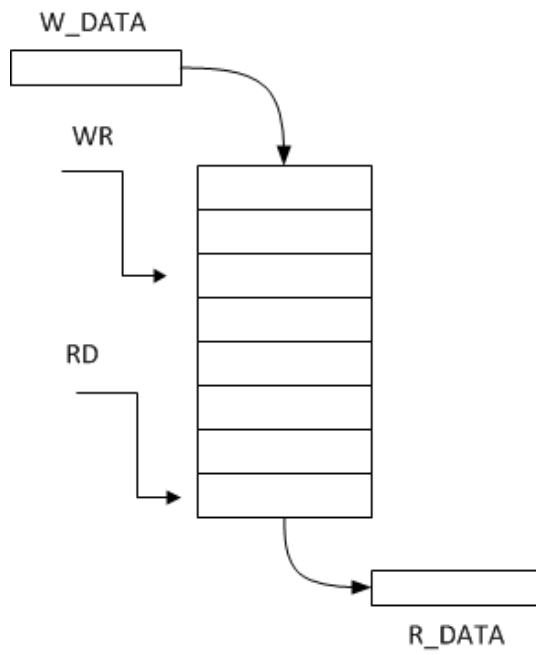


Obr. 34 - Stavový automat UART TX

#### 4.4.4 Modul FIFO paměti

Modul FIFO paměti napomáhá k bezproblémovému přenosu dat mezi komponenty UART a SD kontrolér. Do modulu FIFO paměti se zapisují data pomocí sběrnice „w\_data“, která se zapíše při přijmutí hodnoty „1“ na signálovém vodiči „wr“. Z FIFO paměti se data čtou pomocí signálové sběrnice „r\_data“, data se na ní přenáší při obdržení hodnoty „1“ na signálovém vodiči „rd“. Modul obsahuje také signálové vodiče, které sdělují, jestli je FIFO paměť zaplněná „full“ nebo prázdná „empty“.

Paměť typu FIFO znamená, že data zapsaná jako první, budou také první přečtena. Toho je docíleno pomocí dvou ukazatelů. První obsahuje adresu další volné buňky paměti, do které se bude zapisovat. Druhý obsahuje adresu buňky paměti, ze které se bude číst. Pokud ukazují oba dva na stejnou buňku a zároveň nebylo do paměti zapisováno nebo byla vyčtena všechna data, ukazuje signálový vodič „empty“, že je paměť prázdná. Pokud neukazují na stejnou buňku, sdělují signálové vodiče, že v FIFO paměti se nachází nepřečtená data. Ukazují-li oba dva na stejnou buňku a do paměti byla zapsána data, ale nebylo z ní čteno, sděluje signálový vodič „full“, že je FIFO paměť zaplněná. Znárodnění FIFO paměti je na obr. 35.



Obr. 35 - FIFO paměť



## 5 Návrh demonstrační aplikace

Pro zapisování nebo čtení obsahu paměťové karty je třeba vytvořit demonstrační aplikaci, která bude komunikovat s navrhovaným zařízením. Aplikace bude komunikovat pomocí virtuálního sériového portu, který vytváří ovladač pro integrovaný obvod FT232RL. Samotná fyzická komunikace bude probíhat pomocí USB. V rozšiřujícím modulu se tato USB komunikace převede zpět na UART pomocí obvodu FT232RL. Demonstrační aplikace bude mít za úkol čtení dat z připojené paměťové karty. Načtená data budou dále ukládána do textového souboru, který se uloží na pevný disk osobního počítače. Aplikace bude také moci zapisovat data na kartu. Aplikace se bude ovládat pomocí jednoduchého uživatelského rozhraní.

Program bude mít dvě hlavní funkce. První funkcí bude odeslání příkazu pro čtení dat. Následně vyčká na přečtení dat z karty. Přečtená data zobrazí v dialogovém okně a uloží je na pevný disk ve formě datového souboru. Druhou funkcí bude odeslání příkazu pro zápis dat. Poté odešle data do zařízení.

Při vytvořeném virtuálním sériovém portu lze program sestavit stejným způsobem, jako při běžně fyzicky dostupném sériovém portu. K vytvoření virtuálního sériového portu slouží ovladač VCP, který je volně dostupný na stránkách výrobce obvodu FT232RL. Po připojení zařízení k osobnímu počítači je pomocí tohoto ovladače presentován USB port jako port sériový. Lze využít také ovladač D2XX, který je taktéž volně dostupný na stránkách výrobce obvodu FT232RL. Nespornou výhodou použití ovladače D2XX společně s knihovnou FTD2XX.dll je zjednodušení práce při programování dané aplikace. Použitím funkcí z této knihovny se nemusí tolik brát ohled na koncové komunikační rozhraní, které je přímo převáděno na USB rozhraní.

## 6 Závěr

V bakalářské práci jsem se seznámil s komunikací s paměťovou kartou typu SD. V textu jsou popsány komunikační protokoly paměťové karty, čtení a zápis dat, důležité registry paměťové karty a její odpovědní rámce. Také je zde zmíněn obvod FPGA Spartan-3, který se nachází na vývojové desce Xilinx Spartan-3 Starter Kit board. Pro komunikaci s paměťovou kartou byl vybrán komunikační protokol SPI.

Vývojová deska Spartan-3 Starter Kit board neobsahuje slot pro SD kartu ani USB port pro připojení k počítači. Z tohoto důvodu byla navržena rozšiřující deska, která slot pro SD kartu i USB port obsahuje. Pro převod z USB na UART byl použit integrovaný obvod FT232R. Bylo vytvořeno elektronické schéma zapojení rozšiřující desky. Následně bylo navrženo rozložení součástek a vodivých cest pro dvouvrstvou desku plošného spoje. Deska byla vyrobena ve společnosti PragoBoard. Vyrobena deska byla osazena součástkami a otestována. Byl popsán návrh vnitřního zapojení pro obvod FPGA. Podařilo se popsat a zprovoznit komunikační SPI rozhraní, inicializaci, zápis na kartu, čtení dat z karty s následným zapsáním načtených dat do vyrovnávací paměti a jejich odeslání přes UART rozhraní. Návrh umožňuje zápis a čtení na karty typu SDSC i typu SDHC. Dalším možným rozšířením této práce by mohlo být např.: přidání zápisu a čtení dat pomocí více datových bloků, rozšířením o zařízení pro záznam dat, použití dvouportové vyrovnávací paměti pro přímý přístup k datům.

## Seznam literatury

- [1] *SD Specifications Physical Layer Simplified Specification*. SD Card Association [online]. May 18. 2010 [cit 2013-05-01]. Dostupné na WWW: <[https://www.sdcard.org/downloads/pls/simplified\\_specs/Part\\_1\\_Physical\\_Layer\\_Simplified\\_Specification\\_Ver\\_3.01\\_Final\\_100518.pdf](https://www.sdcard.org/downloads/pls/simplified_specs/Part_1_Physical_Layer_Simplified_Specification_Ver_3.01_Final_100518.pdf)>
- [2] *SanDisk SD Card Product Family*. OEM product manual version 2.2[online]. June 2007 [cit. 2013-05-01]. Dostupné na WWW: <<http://media.digikey.com/pdf/Data%20Sheets/M-Systems%20Inc%20PDFs/SD%20Card%20Prod%20Family%20OEM%20Manual.pdf>>
- [3] *Není karta, jako karta*. Jakub Vrbacký[online]. 15.1.2012 [cit. 2013-05-03]. Dostupné na WWW: <<http://mobilizujeme.cz/clanky/neni-karta-jako-karta-pametove-karty-od-minulosti-po-soucasnost-vedecke-okenko/>>
- [4] MYSLÍK, Alek. Paměťové karty SD. *Praktická elektronika - Amatérské radio*. Praha: Amaro, 2008, XIII, č. 3, s. 37-38. [cit. 2013-05-03] ISSN 1211-328x
- [5] *Spartan-3A FPGA Family: Data Sheet*. Xilinx[online]. August 19, 2010 [cit. 2013-05-09]. Dostupné na WWW: <[http://www.xilinx.com/support/documentation/data\\_sheets/ds529.pdf](http://www.xilinx.com/support/documentation/data_sheets/ds529.pdf)>
- [6] *Spartan-3FPGA Starter Kit Board User Guide ug130(v1.2)*[online]. June 20. 2008 [cit. 2013-05-09]. Dostupné na WWW: <[http://www.xilinx.com/support/documentation/boards\\_and\\_kits/ug130.pdf](http://www.xilinx.com/support/documentation/boards_and_kits/ug130.pdf)>
- [7] *FT232R USB UART IC* Future Technology Devices International LTD. [online]. March 2012 [cit. 2013-05-10]. Dostupné na WWW: <[http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS\\_FT232R.pdf](http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT232R.pdf)>
- [8] *SPI Block User Guide* Motorola, Inc. (v02.06) [online]. Jan 21, 2000 rev: Mar 06, 2002 [cit. 2013-05-27]. Dostupné na WWW: <<http://www.phas.ubc.ca/~scho/9s12/9s12dp256/S12SPIV2.pdf>>
- [9] *Universal Asynchronous Receiver Transmitter* Hariom Meena Rohan Sharma. [online]. 2009 [cit. 2013-05-27]. Dostupné na WWW: <<http://www.slideshare.net/cs1090211/uart-7741743>>
- [10] CHU, Pong P. FPGA prototyping by VHDL examples: Xilinx Spartan-3 version. Hoboken: Wiley-Interscience, c2008, xxv, 440 s. ISBN 978-0-470-18531-5.

## Seznam symbolů a zkratek

ACMD	Application Command - aplikační příkaz
ASIC aplikaci	Application-Specific Integrated Circuit – integrovaný obvod pro specifickou aplikaci
CD	Compact Disc
CID	Card Identifier - identifikační údaje karty
CLK	Clock signal - hodinový signál
CMD	Command – příkaz
CRC	Cyclic Redundancy Check - cyklický redundantní součet
CS	Chip select - výběr podřízeného zařízení
CSD	Card Specific Data - specifikační údaje karty
DAT	Data line - datová linka
EEPROM	elektricky smazatelná programovatelná paměť pouze pro čtení
eXFAT	Extended File Allocation Table - rozšířená alokační tabulka souborů
FAT	File Allocation Table - alokační tabulka souborů
FIFO	First In, First Out – paměť typu fronta (první dovnitř první ven)
FM	frekvenční modulace
FPGA	Field Programmable Gate Array - programovatelné hradlové pole
GPS	Global Positioning System – globální poziční systém
I/O	Input/Output - vstup/výstup
IrDA	infračervený port
JTAG	Joint Test Action Group
LAN	Local Area Network – lokální síť
LED	Light Emitting Diode - dioda emitující světlo
MISO	Master In Slave Out – vstup řídicího zařízení, výstup podřízeného zařízení
MMC	Multi Media Card
MOSI	Master Out Slave In - výstup řídicího zařízení, vstup podřízeného zařízení
OEM	Original Equipment Manufacturer – označení výrobce zařízení

PROM	programovatelná paměť pouze pro čtení
RFID	Radio Frequency Identification - identifikace na rádiové frekvenci
ROM	paměť pouze pro čtení
RSV	rezervováno
SCLK	Slave Clock - hodinový signál pro podřízené zařízení
SD	Secure Digital
SDHC	Secure Digital High Capacity
SDIO	Secure Digital Input Output
SDXC	Secure Digital eXtended Capacity
SMD	surface mount device – pouzdro pro povrchovou montáž
SPI	Serial Peripheral Interface - sériové periferní rozhraní
SS	Slave Select - výběr podřízeného zařízení
SSOP	Shrink small-outline package – pouzdro
UART	Universal Asynchronous Receiver Transmitter - univerzální asynchronní
přijímač vysílač	
UHS	Ultra-High Speed - ultra vysokorychlostní
USB	Universal Serial Bus - univerzální sériová sběrnice
VDD	Positive supply voltage - kladné napájecí napětí
VGA	Video Graphics Array – výstup pro připojení monitoru
VSS	Negative supply voltage – uzemnění
Wi-Fi	Wireless LAN - bezdrátová komunikace v počítačové síti
WP	Write Protect – ochrana proti zápisu

## **Seznam příloh**

Příloha č. 1 – Schéma zapojení rozšiřujícího modulu

Příloha č. 2 – Návrh zapojení desky v horní vrstvě a osazovací plán horní vrstvy

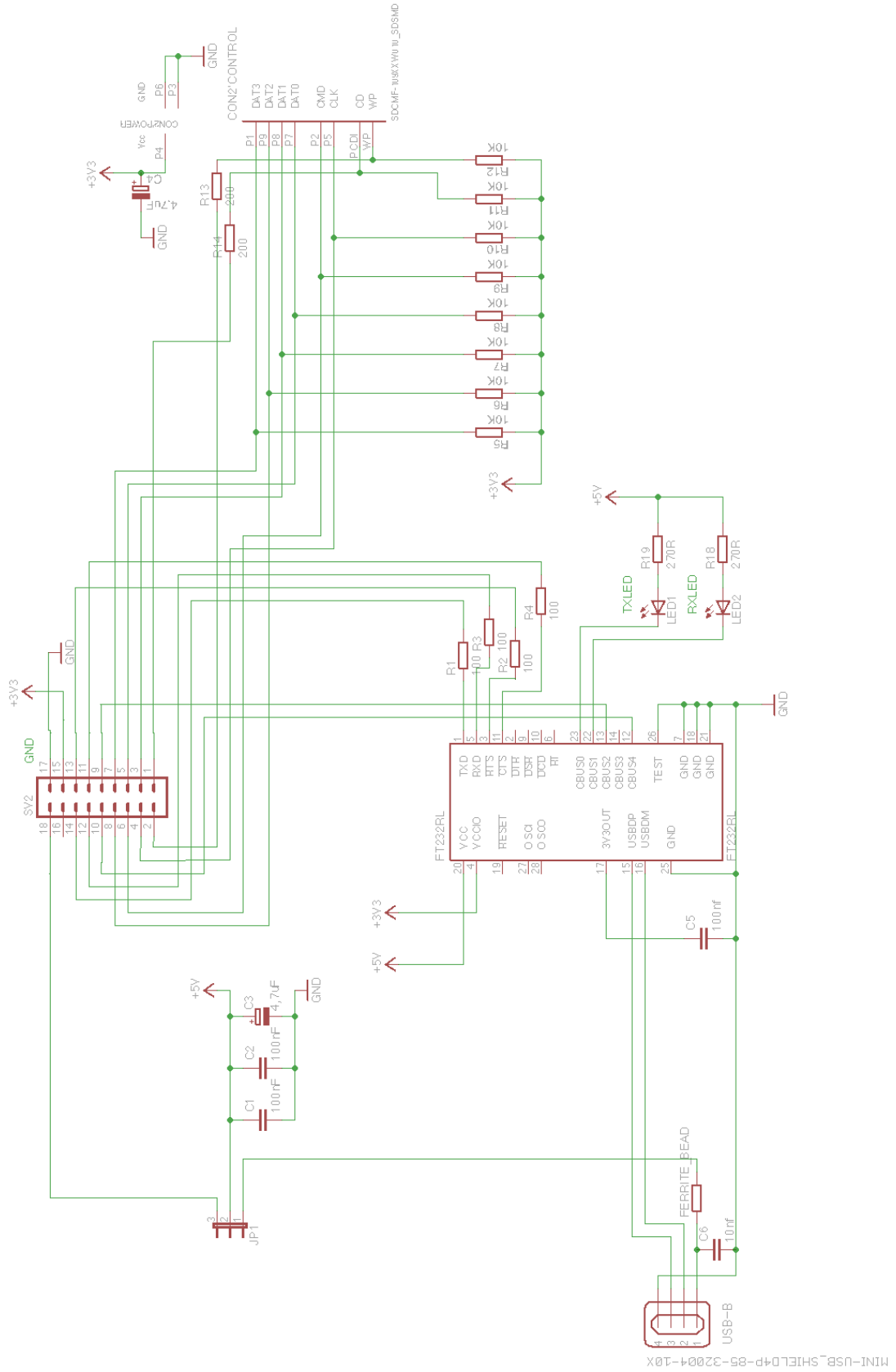
Příloha č. 3 – Návrh zapojení desky ve spodní vrstvě a osazovací plán spodní vrstvy

Příloha č. 4 – Seznam součástek

Příloha č. 5 – Fotografie hotové rozšiřující desky

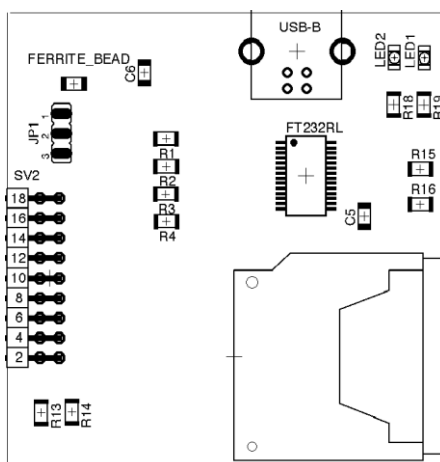
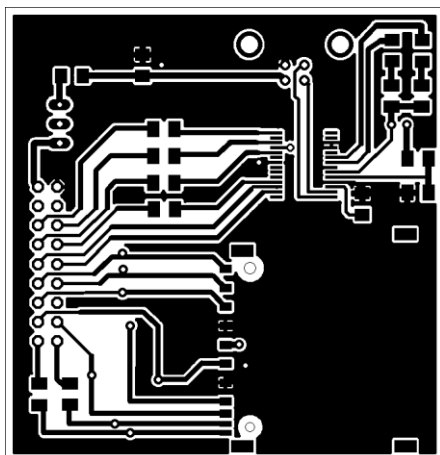
Příloha č. 6 – Na přiloženém CD se nachází veškeré zdrojové kódy ve VHDL, návrh desky v programu EAGLE a elektronická verze této práce

**Příloha č. 1 - Schéma zapojení rozšiřujícího modulu**



MINI-USB\_SHIELD+P-85-32004-10X

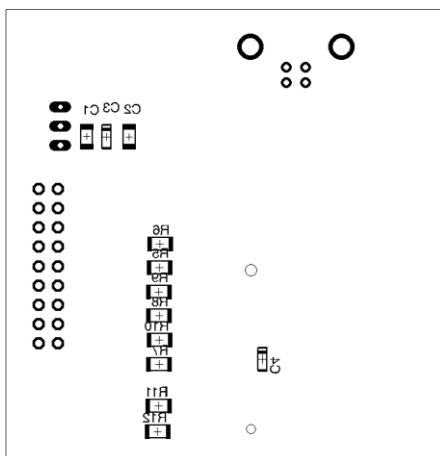
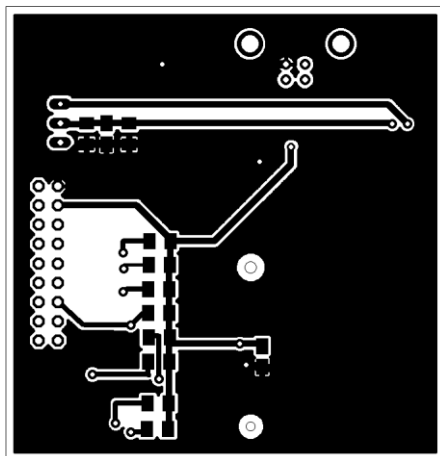
*Příloha č. 2 – Návrh zapojení desky v horní vrstvě a osazovací plán horní vrstvy*



Rozměr desky 58 x 60 [mm], měřítko M1:1



**Příloha č. 3 – Návrh zapojení desky ve spodní vrstvě a osazovací plán spodní vrstvy**



Rozměr desky 58 x 60 [mm], měřítko M1:1

**Příloha č. 4 – Seznam součástek**

<b>Označení</b>	<b>Hodnota</b>	<b>Pouzdro</b>	<b>Popis</b>
C1	100nF	C1206	Keramický kondenzátor
C2	100nF	C1206	Keramický kondenzátor
C3	4,7uF	A/3216-18R	Tantalový kondenzátor
C4	4,7uF	A/3216-18R	Tantalový kondenzátor
C5	100nF	C1206	Keramický kondenzátor
C6	10nf	C1206	Keramický kondenzátor
CON2	-	SDCMF-109XXW010	Slot paměťové karty
FERRITE_BEAD	-	R1206	Feritová perla
FT232RL	FT232RL	SSOP28	Převodník USB <> UART
JP1	-	1X03	Propojka
LED1	-	1206	LED dioda
LED2	-	1206	LED dioda
R1 - R4	100	R1206	SMD rezistor
R5 – R12	10K	R1206	SMD rezistor
R13, R14	200	R1206	SMD rezistor
R15	4K7	R1206	SMD rezistor
R16	10K	R1206	SMD rezistor
R18, R19	270	R1206	SMD rezistor
SV2	-	MA09-2W	Výstupní port
USB-B	Mini-USB	85-32004-10X	USB port

Příloha č. 5 – Fotografie hotové rozšiřující desky

