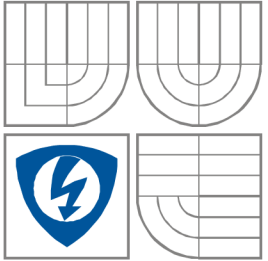




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A
KOMUNIKAČNÍCH
TECHNOLÓGIÍ
ÚSTAV RADIOELEKTRONIKY**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF RADIO ELECTRONICS

USB PŘEPÍNAČ

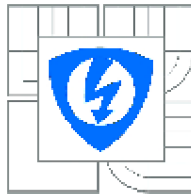
USB SWITCH

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR
VEDOUCÍ PRÁCE
SUPERVISOR

Martin Mrník
Ing. Zbyněk Fedra, Ph.D.

BRNO, 2011



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav radioelektroniky

Bakalářská práce

bakalářský studijní obor
Elektronika a sdělovací technika

Student: Martin Mrník
Ročník: 3

ID: 111106
Akademický rok: 2010/2011

NÁZEV TÉMATU:

USB přepínač

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s protokolem USB. Navrhněte zařízení pro připojení několika počítačů k více zařízením (např. tiskárnám) pomocí rozhraní USB jako alternativu síťového tisku. Zařízení bude moci přepínat USB komunikaci. Dále promyslete a zhodnoťte možnost sdíleného přístupu k paměťovým mediím (USB disky apod.). Vyberte vhodný mikrokontrolér a navrhněte zapojení obvodu. Realizujte navržené zařízení, oživte je a otestujte. Zhodnoťte možnosti či případná omezení Vaší koncepce.

DOPORUČENÁ LITERATURA:

[1] KAINKA, B. USB. Měření, řízení a regulace pomocí sběrnice USB. Praha: BEN, 2002. 1. vyd. 247 s. ISBN 80-7300-073-3.

[2] AXELSON, J. USB complete: everything you need to develop custom USB peripherals. New York: Lakeview Research, 2005. 3rd ed., 572 s. ISBN 978-1-931448-02-4.

Termín zadání: 7.2.2011

Termín odevzdání: 27.5.2011

Vedoucí práce: Ing. Zbyněk Fedra, Ph.D.

prof. Dr. Ing. Zbyněk Raida
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

Práce sa zaoberá opisom štandardu USB a princípom jeho fungovania, výhodami, nevýhodami a rôznymi možnosťami použitia. Čitateľovi ponúka bližší pohľad na procesy neviditeľné bežnému užívateľovi v priebehu inicializácie, prevádzky a správy zariadenia, ich význam, štruktúru a časovanie. Práce rozoberá možnosti pripojenia viac počítačov k USB zariadeniam prostredníctvom prepínača USB komunikácie a venuje sa výberu vhodných komponentov pre takýto prepínač, jeho vývoju, skonštruovaniu, programovaniu jeho riadiacich obvodov a obslužného programu. Práca taktiež posudzuje možnosti zdieľania pamäťových médií prostredníctvom USB prepínača.

KLÍČOVÁ SLOVA

USB prepínač

ABSTRACT

This thesis deals with USB bus specification, its advantages, disadvantages, and different possibilities of use. To the readers it offers a closer look on the processes invisible to user during startup, operation and management of the device, their meaning, structure and timing. The thesis discusses the possibility of connecting multiple computers to USB devices via USB switch and is dedicated to the selection of suitable components for such a switch, development, hardware construction, programming the control circuits and utility. The thesis also considers the possibility of sharing storage media via USB switch.

KEYWORDS

USB switch

MRNÍK, M. *USB přepínač*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky, 2011. 32 s., 3 s. příloh. Bakalářská práce. Vedoucí práce: Ing. Zbyněk Fedra, Ph.D

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma USB přepínač 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

.....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce Ing. Zbyňkovi Fedrovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne

.....

(podpis autora)

Obsah

Zoznam obrázkov	6
Zoznam obrázkov	6
1 Úvod.....	7
2 Úvod do USB.....	8
2.1 Vezie USB.....	8
2.2 Topológia.....	9
2.3 Konektory	10
2.4 Vlastnosti kábla.....	11
2.5 Povinnosti hostiteľa	12
2.6 Povinnosti zariadenia	13
3 Komunikácia.....	15
3.1 Prenosy.....	15
3.2 Endpointy - zdroje a ciele dát	16
3.3 Dátové trubice – spojenie endpointov a hostiteľa.....	17
3.4 Proces enumerácie.....	17
3.5 Komunikácia používaná pri vlastnej funkcii zariadenia.....	20
4 Možnosti vyhotovenia prepínača	21
4.1 USB prepínač s Atmega 8u2.....	21
4.2 USB prepínač s FTDI 232 RL.....	22
5 Realizácia hardware.....	23
6 Realizácia software	25
6.1 Obslužný program	25
6.2 Užívateľské rozhranie.....	28
6.3 Firmware prepínačov	29
6.4 Nároky na softwarové vybavenie počítača.....	29
7 Testovanie	30
7.1 Zdieľanie pamäťových médií.....	30
8 Záver	30
Zoznam použitých zdrojov	32
Obsah priloženého CD	33
Zoznam príloh	34

Zoznam obrázkov

Obrázok č. 1: Topológia USB	10
Obrázok č. 2: Štruktúra kábla USB.....	11
Obrázok č. 3: USB 3.0 konektory typu A a B	11
Obrázok č. 4: Schéma prenosových rámcov	15
Obrázok č. 5: Bloková schéma zapojenia.....	23
Obrázok č. 6: Schéma zapojenia prevodníka	24
Obrázok č. 7: Schéma zapojenia vstupného multiplexera.....	24
Obrázok č. 8: Vývojový diagram inicializácie	25
Obrázok č. 9: Vývojový diagram zisťovania obsadenosti prepínača a jednotlivých portov.....	27
Obrázok č. 10: Vývojový diagram nadväzovania spojenia	27
Obrázok č. 11: Snímka ovládacieho programu – je aktívne spojenie portu 1 a 2	28
Obrázok č. 12: Program MProg 3.5 a jeho nastavenie pre prevodník č.1	29
Obrázok č. 13: Výsledky testu rýchlosti prenosu dát. Na obrázku vľavo bol USB disk pripojený priamo na koreňový hub v PC. Na obrázku vpravo bol USB disk pripojený cez prepínač.	30

Zoznam obrázkov

Tabuľka č. 1: Typy transakcií	16
Tabuľka č. 2: Maximálne prenosové rýchlosti pre jednotlivé typy prenosov a rýchlosti zbernice	20

1 Úvod

Cieľom tejto práce je zoznámiť sa so štandardom USB a navrhnúť zariadenie pre pripojenie niekoľkých počítačov k viacerým zariadeniam pomocou rozhrania USB, hlavne ako alternatívu sieťovej tlače. Práca približuje základy štandardu USB, jeho výhody, nevýhody a systém fungovania. Ďalej sa venuje porovnaniu viacerých možností realizácie prepínača, výberu najvhodnejšieho riešenia a opisuje priebeh práce na hardwarovej a softwarovej časti. V závere sa venuje testovaniu prepínača, jeho vplyvu na prenos dát a zhodnoteniu dosiahnutých výsledkov a možnosti zdieľania pamäťových médií.

2 Úvod do USB

Univerzálna sériová zbernica USB je špecifikácia pre nadviazanie spojenia medzi zariadeniami a hostiteľským radičom (obvykle osobný počítač) vyvinutá Ajayom Bhattom, indickým vývojovým pracovníkom Intelu. [1] USB efektívne nahrádza širokú škálu rozhraní, medzi inými hlavne sériový a paralelný port. USB sa v súčasnosti využíva hlavne na pripojenie počítačových periférií ako myši, klávesnice, herné ovládače, web kamery, prenosné úložné zariadenia, tlačiarne ale aj pre spojenie so smartphonami, PDA, širokou škálou externého hardwaru a spotrebnej elektroniky pre ktorý sa USB stal štandardom spojenia s počítačom.

USB bolo oficiálne predstavené konzorciom siedmich počítačových a telekomunikačných lídrov trhu v roku 1995 a to Compaq, Digital Equipment Corporation (DEC), IBM, Intel, Microsoft, NEC a Northern Telecom. [2] Jeho dizajn je štandardizovaný nekomerčnou spoločnosťou USB Implementers Forum (USB-IF) ktorá má v súčasnosti viac ako osemsto členov a jej predstavenstvo sa skladá zo spoločností Hewlett-Packard, Intel, LSI, Microsoft, Renesas Electronics a ST-Ericsson. [2] Bola založená na podporu, rozvoj a prijatie technológie USB. Úlohou USB-IF je aj vydávanie Vendor ID čo je identifikátor výrobcu zariadenia.

Hlavným zámerom vyvinutia USB bolo zbaviť sa všetkých predtým používaných sériových a paralelných portov, pretože neboli dostatočne štandardizované a vyžadovali množstvo ovládačov.

USB bolo navrhnuté tak, aby periférie mohli byť pripojené Plug&Play čo je technológia umožňujúca jednoduchšie rozpoznanie hardwaru, umožňuje operačným systémom zistiť, aký hardware bol pripojený a na základe toho môže automaticky k nainštalovať potrebné ovládače. To sa deje bez potreby reštartovania či vypnutia počítača pri pripájaní či odpájaní. Veľkou výhodou USB je aj to, že dokáže pripojeným zariadeniam poskytnúť napájanie na +5V a prúd až do 500 mA čím v porovnaní s ostatnými zbernicami odpadá nutnosť mať ku každému zariadeniu externý zdroj napájania.[3]

2.1 Vezie USB

USB 1.0 a 1.1

Špecifikácia USB 1.0 bola zavedená v roku 1996 s dvoma rýchlosťami prenosu dát, Low-Speed (1,5 Mbps) a Full-Speed (12 Mbps). Špecifikácia 1.1 opravujúca niektoré nedostatky bola vydaná v roku 1998 a bola to prvá ktorá bola široko prijatá. [4]

USB 2.0

Niekoľko spoločností viedlo iniciatívu vyvinúť vyššiu rýchlosť prenosu dát, až 480 Mbps, asi 40 krát rýchlejšie ako predchádzajúca verzia. Vznikol Hi-Speed USB 2.0 a rozšíril okruh externých zariadení, ktoré mohli byť použité k počítaču. A čo bolo veľmi veľkou výhodou, ponúkol spätnú kompatibilitu s predchádzajúcimi generáciami. Špecifikácia USB 2.0 bola vydaná v roku 2000 [4].

USB On-The-Go

Mnohé zariadenia USB sú prenosné a vzniká potreba zariadení komunikovať priamo medzi sebou bez použitia počítača. Dodatok On-The-Go umožňuje periférnym zariadeniam komunikovať priamo medzi sebou. On-The-Go funkcie zahŕňajú:

- Obmedzená schopnosť hostiteľa pre komunikáciu s vybranými periférnymi zariadeniami USB

- Malý konektor vhodný pre mobilné riešenia
- Nízke nároky na spotrebu
- Schopnosť byť buď hositeľ alebo periférne zariadenie a dynamicky prepínať medzi týmito dvoma módmami.

Bezdrôtové USB

Bezdrôtové USB je vysokofrekvenčná bezdrôtová rádiová komunikačná technológia s krátkym dosahom. Pôvodne bola vyvinutá Wireless USB Promoter Group, následne prešla pod správu USB-IF. [5] Navrhnutá od základov až k riešeniu problémov bezdrôtovej komunikácie, táto norma spája rýchlosť a bezpečnosť káblových technológií a jednoduchosť používania bezdrôtovej technológie. Môže byť použitá v rôznych zariadeniach vrátane herných ovládačov, tlačiarň, skenerov, digitálnych fotoaparátov, digitálnych hudobných prehrávačov, pevných diskov či flash diskov. Pomocou bezdrôtového USB sa dá aj prenášať súbežne viac video streamov.

Pomocou bezdrôtového USB je možné posielat dáta rýchlosťou až 480 Mbps na vzdialenosť 3 m a až 110 Mbps na vzdialenosť 10 metrov. Bolo navrhnuté pre frekvenčné pásmo 3,1 až 10,6 GHz. Architektúra umožňuje bezdrôtovo pripojiť až 127 zariadení priamo k hositeľovi. S tým odpadá nutnosť potužitia rozbočovača. [5]

USB 3.0

V novembri 2008 USB 3.0 Promoter Group oznámila dokončenie špecifikácie 3.0. [4] Poskytuje až 10 krát vyššiu rýchlosť prenosu ako prostredníctvom USB 2.0 využitím až 5,0 Gbps rýchlosti dát. Navyše, má optimalizovanú energetickú účinnosť, sync-n-go technológiu, ktorá minimalizuje čas ktorý musí užívateľ čakať na konfiguráciu zariadenia a je spätne kompatibilná s USB 2.0. SuperSpeed USB zariadenia spolupracujú s 2.0 hositeľmi a SuperSpeed USB hositelia podporujú staršie Hi-Speed zariadenia. [6] Napriek tomu, že od dokončenia špecifikácie ubehlo už viac ako dva a pol roka, verzia USB 3.0 začína byť komerčne dostupná až od jari roku 2011. [6]

2.2 Topológia

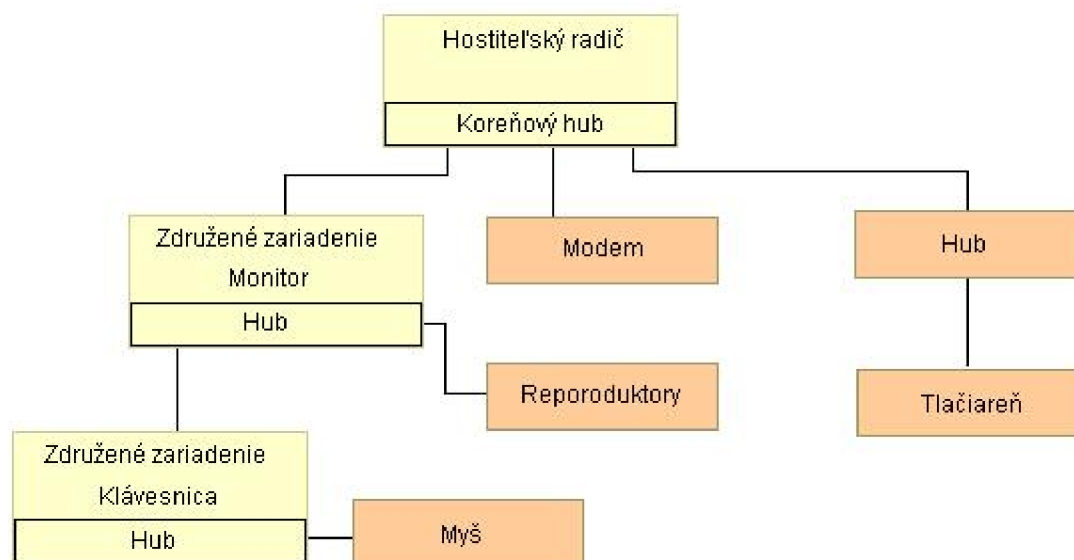
Zariadenia na zbernici USB sú fyzicky pripojené k hositeľovi cez stupňovitú hviezdicovú topológiu, ktorá vyzerá ako strom. Fyzické umiestnenie prvkov v topológii nemá vplyv na logické spojenie. Pripojovacie body (porty USB) pre periférne zariadenia sú poskytované špeciálnou triedou USB zariadení s názvom hub (rozbočovač). Na komunikáciu, hositeľ ani zariadenie nepotrebujú vedieť cez koľko hubov signály prechádzajú.

Hub komunikuje s hositeľom rovnakým spôsobom ako ktorékoľvek zariadenie pripojené do USB, ale hrá osobitnú úlohu. Koreňový hub poskytuje jeden alebo viac počiatočných bodov - portov pripojenia na hositeľa. Ďalšie porty sú poskytované externými hubmi.

Koreňový hub je integrovaný v rámci hositeľského radiča USB. Pár hositeľský radič + koreňový hub je vždy v strede hviezdry a je považovaný za prvý stupeň stromu USB. Akýkoľvek ďalší stupeň stromu je vytvorený externým hubom, ktorý je sprostredkovateľom medzi periférnymi zariadeniami a hositeľským počítačom.

Externý hub má jeden port na vyššiu úroveň v hierarchii topológie USB (upstream) a viacero (typicky 4 alebo 7) portov pre pripojenie periférnych zariadení a ďalších hubov (downstream) , ktorá umožňuje pomocou kaskádovania pripojiť až 127 zariadení.[3]

Periférne zariadenia sú vždy pomyselné listy v strome USB a sú pripojené k hostiteľovi cez reťaz USB hubov. Moderné systémy majú viacero hostiteľských radičov, takže je pravdepodobné že v jednom počítači je viac ako jeden USB strom.



Obrázok č. 1: Topológia USB

Teoreticky, protokol USB umožňuje až 127 periférnych zariadení (vrátane hubov), ktoré je možné pripojiť na jeden hostiteľský radič. Pole adresy v USB pakete, špecifikujúcej pre ktoré zariadenie je paket určený, má dĺžku 7 bitov. Z toho vyplýva rozsah hodnôt v rozmedzí 0 až 127. Adresa 0 je rezervovaná pre zariadenie, ktoré ešte nemá priradenú svoju vlastnú adresu. Avšak v praxi je počet zariadení ďaleko menší ako teoretické maximum. Je vysoko pravdepodobné, že zbernica USB vyčerpá buď šírku pásma, alebo prekročí možnosti napájania oveľa skôr ako sa počet pripojených zariadení priblíži k číslu 127. [7]

2.3 Konektory

Štandardný typ A

Štandardný typ A USB konektora má tvar splošteného obdĺžnika, ktorý sa vkladá do downstream portov na koreňovom alebo externom hube. Tento typ konektora sa často nachádza na kábli, ktorý je trvalo pripojený k zariadeniu, napríklad klávesnice alebo myši.

Opakovaným pripájaním a odpájaním sa spojenie časom opotrebuje a uvoľní. Životnosť konektora typu A je približne 1 500 pripájacích/odpájacích cyklov. [8]

Štandardný typ B

Štandardný typ B USB konektora má štvorcový tvar so skosenými vonkajšími rohmi a typicky sa zapája do upstream konektora na zariadení, ktoré používa odnímateľný kábel, napríklad tlačiarne. Použitie dvoch odlišných typov konektora bráni užívateľovi vo vytvorení elektrického skratu. [9]

USB mini a mikro konektory

Rôzne konektory boli použité pre menšie zariadenia, ako sú PDA, mobilné telefóny alebo digitálne fotoaparáty. Patrí medzi ne v súčasnosti zastaraný [10] (ale štandardizovaný) Mini-A a v súčasnosti štandardný Mini-B, [11] Mikro-A a Mikro-B konektor.

USB Micro-AB Socket On-The-Go

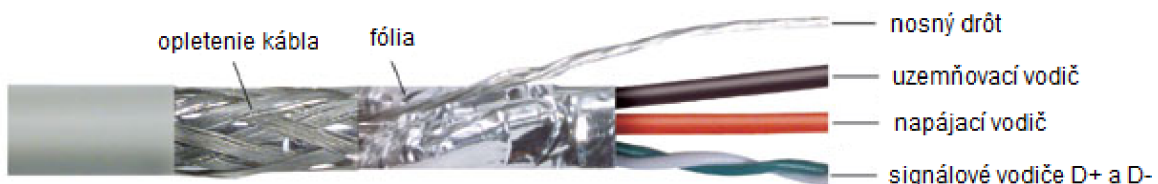
Zariadenie On-The-Go musí mať iba jeden USB konektor: Micro-AB zásuvku. Táto zásuvka je schopná prijímať buď Micro-A konektor alebo Micro-B konektor.

Prístroj OTG s pripojenou zásuvkou A sa nazýva zariadenie A a je zodpovedné za napájanie USB rozhrania v prípade potreby a v predvolenom nastavení má úlohu hostiteľa. Prístroj OTG s pripojenou B zásuvkou sa nazýva B zariadenie a v predvolenom nastavení prevezme úlohu pripojeného zariadenia. Ak prístroj OTG nemá pripojenej zásuvky, správa sa ako B zariadenie. Ak aplikácia v B zariadení vyžaduje rolu hostiteľa, protokolom HNP sa dočasnému prevedie rola hostiteľa na zariadenie B.

OTG zariadenie pripojené štandardnému zariadeniu alebo štandardnému hostiteľovi bude mať svoju úlohu určenú káblom, pretože v týchto prípadoch je možné pripojiť kábel len jedným spôsobom.

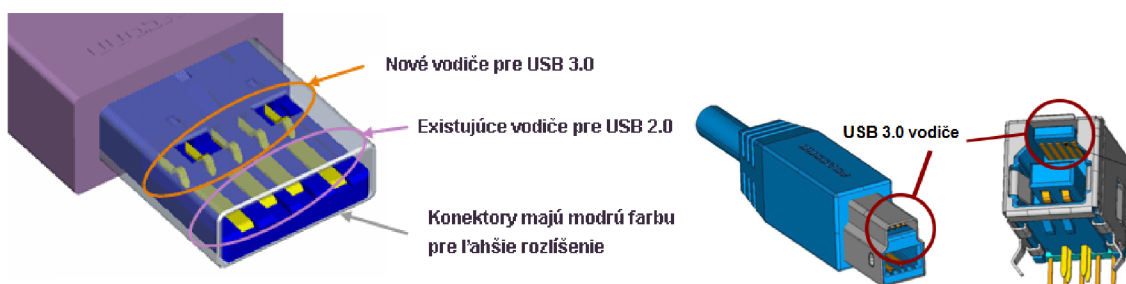
2.4 Vlastnosti kábla

Dátové káble pre USB 1.x USB a 2.x používajú krútený pár pre zníženie šumu a presluchovej. Usporiadanie znázorňuje obrázok :



Obrázok č. 2: Štruktúra kábla USB

USB 3.0 káble sú zložitejšie a obsahujú tienenie pre pridané dátové linky (2 páry) a jeden zemiaci vodič.



Obrázok č. 3: USB 3.0 konektory typu A a B

Maximálna dĺžka kábla

Pre USB 2.0 alebo staršiu verziu je maximálna dĺžka kábla 5 m. [12] Hlavným dôvodom pre toto obmedzenie je maximálne povolené round-trip oneskorenie asi 1,500 ns. Ak zariadenie neodpovedá na príkaz hostiteľa v stanovenej lehote, považuje hostiteľ príkaz za stratený. Ak pridáme čas odozvy zariadenia USB, oneskorenie z maximálneho počtu hubov a z prepojovacích káblov, maximálne prijateľné oneskorenie na jeden kábel činí 26 ns.[12]

USB 2.0 vyžaduje aby oneskorenie kábla bolo menšie ako 5,2 ns na meter (192000 km / s, čo sa blíži k maximálne dosiahnuteľnej prenosovej rýchlosti pre štandardný medený kábel). [13] To vo výsledku umožňuje dĺžku kábla 5 metrov. Štandard USB 3.0 priamo nešpecifikuje maximálnu dĺžku kábla, vyžaduje len, aby všetky káble spĺňali elektrické špecifikácie.

2.5 Povinnosti hostiteľa

Na komunikáciu s USB zariadeniami, počítač potrebuje hardware a software ktorý mu umožní zastávať funkciu USB hostiteľa. Hardware pozostáva z kontroléra a koreňového hubu s jedným alebo viac portami. Softwarom je operačný systém ktorý poskytuje mechanizmus na komunikáciu ovládačov s USB hardwarom. Hostiteľ má na starosti zbernicu. Musí vedieť aké zariadenia sú pripojené a aké sú schopnosti každého z nich. Musí takisto zaistiť, že všetky zariadenia na zbernici môžu prijímať a odosielať dáta podľa potreby v rámci možností a obsadenia zbernice.

Detekcia zariadení

Pri zapínaní huby informujú hostiteľa o všetkých pripojených USB zariadeniach. Kedykoľvek je pripojené nové zariadenie hostiteľ ho enumeruje. V tomto procese sa priradí zariadeniu adresa a získajú sa doplnujúce údaje od každého zariadenia. Odpojené zariadenie odstráni zo správcu zariadení dostupných aplikáciám.

Správa toku dát

Hostiteľ spravuje tok dát na zbernici. Viaceré pripojené zariadenia môžu požadovať prenos dát v rovnaký čas, preto kontrolér rozdelí dostupný čas do segmentov ktoré sa nazývajú rámce a mikro rámce a každému prenosu pridelí určitú časť rámca.

Prenosy, ktoré musia mať zaistenú určitú prenosovú rýchlosť majú garantovaný čas v rámci. V priebehu enumerácie ovládač zariadenia požaduje určitú šírku pásma ktorú bude potrebovať na všetky prenosy ktoré potrebujú garantovanú prenosovú rýchlosť. Ak nie je táto šírka pásma dostupná, hostiteľ nepovolí začatie komunikácie. Ovládač musí buď požiadať o menšiu šírku pásma alebo čakať kým nie je požadované prenosové pásmo dostupné. Prenosy, ktoré nemajú garantovaný čas v rámci používajú zostávajúcu časť a ak je celá zbernica obsadená, musia čakať.

Overovanie chýb

Pri prenose dát hostiteľ pridá overovacie bity. Prijímacie zariadenie vykoná kontrolný súčet prijatých dát a porovná výsledok s prijatými overovacími bitmi. Ak sa výsledok nezhoduje, zariadenie nepotvrdí príjem dát a hostiteľ vie, že ich má odoslať znova. USB podporuje aj typ prenosu ktorý nepovoľuje opätovné odosielanie dát v záujme zachovania konštantnej prenosovej rýchlosti. Hostiteľ takisto overuje dáta prijaté od pripojených zariadení.

V prípade iných alebo zložitejších chýb hostiteľ môže informovať ovládač zariadenia o probléme, ten môže upozorniť aplikáciu a tá môže vykonať zodpovedajúcu akciu.

Poskytovanie napájania

Spolu s dvoma signálovými vodičmi má USB kábel vodič s napájaním +5V a uzemňovací vodič. Podľa špecifikácie poskytuje najviac 5,25 V a nie menej ako 4,75 V ($5\text{ V} \pm 5\%$) medzi pozitívnym a negatívnym vodičom zbernice. [14] Jednotka odberu prúdu je definovaná ako 100 mA v režime USB 2.0 a maximálny odber jedného zariadenia je 5 jednotiek (500 mA). Pre USB 3.0 bola jednotka odberu zvýšená na 150 mA a maximálny odber bol zvýšený na 6 jednotiek (900 mA). Existujú dva typy zariadení: s nízkym odberom (low-power) a s vysokým odberom (high-power). Low-power zariadenia môžu odoberať maximálne 1 jednotku, s minimálnym prevádzkovým napätím 4,4 V (USB 2.0) a 4 V (USB 3.0). High-power zariadenie môže odoberať maximálny počet jednotiek stanovených štandardom. Všetky zariadenia sú na začiatku automaticky low-power, ale software zariadenia môže požiadať o zvýšenie na high-power ak je k dispozícii dostatok prúdu. [14] Zariadenia ktoré požadujú vyšší odber musia mať vlastný zdroj napájania.

Výmena dát s pripojenými zariadeniami

Všetky vyššie popísané činnosti podporujú hlavnú úlohu hostiteľa a to výmenu dát s periférnymi zariadeniami. V niektorých prípadoch ovládač zariadenia požaduje výmenu dát v špecifikovaných intervaloch, kým iné zariadenia komunikujú len keď to aplikácia alebo ovládač požaduje.

2.6 Povinnosti zariadenia

Z mnohých hľadísk, povinnosti zariadenia sú opačné k povinnostiam hostiteľa. Keď hostiteľ iniciuje komunikáciu, zariadenie musí odpovedať. Zariadenia majú aj určité unikátne povinnosti. Zariadenie nemôže začať komunikáciu samé od seba, musí čakať a iba odpovedať na požiadavky hostiteľa. USB kontrolér v zariadení vykonáva mnohé z povinností na úrovni hardwaru, miera softwarovej podpory závisí od použitého kontroléra. Zariadenie musí vykonávať všetky nasledujúce úlohy.

Detekcia komunikácie

Každé zariadenie sleduje adresu zariadenia obsiahnutú v každej komunikácii na zbernici. Ak adresa nezodpovedá adrese zariadeniu pridelenej, zariadenie komunikáciu ignoruje. Ak sa zhoduje, zariadenie uloží dáta do prijímacieho buffera a spustí prerušenie aby indikovalo, že dáta dorazili. Takmer vo všetkých čipoch sú tieto funkcie zabudované v hardwari a nepotrebujú softwarovú podporu.

Odpovedanie na štandardné požiadavky

Pri zapnutí alebo pri pripojení zariadenia k zapnutému počítaču, zariadenie musí odpovedať na štandardné požiadavky vysielané hostiteľom počas enumerácie. Hostiteľ ale môže posilať požiadavky aj kedykoľvek po skončení enumerácie.

Všetky zariadenia musia odpovedať na tieto požiadavky, zahŕňajúc dotazy na schopnosti a stav, alebo požiadavky na vykonanie určitej akcie. Po prijatí a spracovaní požiadavky, zariadenie umiestni dáta alebo informácie o stave do prenosového buffera na odoslanie k hostiteľovi. Pri niektorých požiadavkách ako napríklad výber konfigurácie, zariadenie vykoná aj inú akciu okrem odpovedania.

USB špecifikácia definuje jedenásť požiadaviek, ale trieda zariadenia alebo samotný predajca môže špecifikovať ďalšie. Zariadenie nemusí vykonať každú požiadavku, musí však na ňu odpovedať v zrozumiteľnej forme. Napríklad, keď hostiteľ požaduje konfiguráciu, ktorú zariadenie nepodporuje, musí odpovedať kódom ktorý znamená že daná konfigurácia nie je podporovaná.

Kontrola chýb

Podobne ako hostiteľ, zariadenie vkladá overovacie bity do vysielaných dát. Pri prijímaní dát vykonáva kontrolný súčet. Podľa toho či zariadenie potvrdí prenos alebo nepotvrdí hostiteľ vie či poslať dáta znova. Tieto funkcie sú obvykle zabudované v hardwari kontroléru a nie je ich treba programovať.

Správa napájania

Zariadenie môže byť napájané zo zbernice alebo mať vlastné napájanie. Tie ktoré sú napájané zo zbernice keď nevykonávajú žiadnu aktivitu musia obmedziť svoju spotrebu prúdu na minimum. Keď hostiteľ vstúpi do stavu nízkeho odberu, všetka komunikácia na zbernici končí vrátane periodických časových značiek ktoré hostiteľ posiela v normálnom režime. Pri detekovaní chýbajúcej aktivity na zbernici dlhšie ako tri milisekundy, zariadenie musí vstúpiť do stavu nízkeho odberu a obmedziť svoj odber prúdu. Zároveň ale stále musí sledovať zbernicu a vrátiť sa do normálneho režimu pri obnovení aktivity na zbernici.

Zariadenia ktoré nepodporujú funkciu vzdialeného zobudenia by nemali v stave nízkeho odberu spotrebovať viac než 500 μ A. Tie ktoré funkciu vzdialeného zobudenia podporujú a zároveň hostiteľ povolil túto funkciu môžu odoberať maximálne 2,5 mA. [3] Tieto hodnoty sú priemerné za jednu sekundu, špičkové hodnoty môžu byť vyššie.

Výmena dát s hostiteľom

Všetky vyššie spomenuté povinnosti podporujú hlavnú úlohu USB portu zariadenia a to výmenu dát s hostiteľom. Po nakonfigurovaní, zariadenie musí odpovedať na komunikáciu, ktorá môže obsahovať dáta a zároveň požadovať dáta alebo iné informácie. Schopnosti zariadenia, ovládač hostiteľa a aplikácia ktorá zariadenie používa určuje typ komunikácie a aj to kedy sa vyskytuje.

Pri väčšine prenosov, kedy hostiteľ posiela dáta do zariadenia, zariadenie musí odpovedať na každý pokus o prenos poslaním kódu ktorý vyjadruje či zariadenie dáta prijalo alebo bolo zaneprázdnené. Pri väčšine prenosov kedy zariadenie posiela dáta hostiteľovi, zariadenie musí na požiadavku o prenos odpovedať zaslaním príslušných dát alebo informáciou, že žiadne dáta na odoslanie nie sú alebo zariadenie je zaneprázdnené. Obvykle hardware odpovedá automaticky, podľa nastavenia vo firmware. Niektoré prenosy nepoužívajú potvrdenie prijímu a odosielateľ predpokladá že príjemca prijal všetky prenášané dáta. Hardware kontroléru sa takisto stará o formátovanie dát pre zbernicu. To zahŕňa pridávanie overovacích bitov, vykonávanie kontrolného súčtu a posielanie a prijímanie jednotlivých bitov po zbernici.

Okrem toho zariadenie musí vykonávať svoju funkciu pre ktorú si ju zákazník zakúpil.

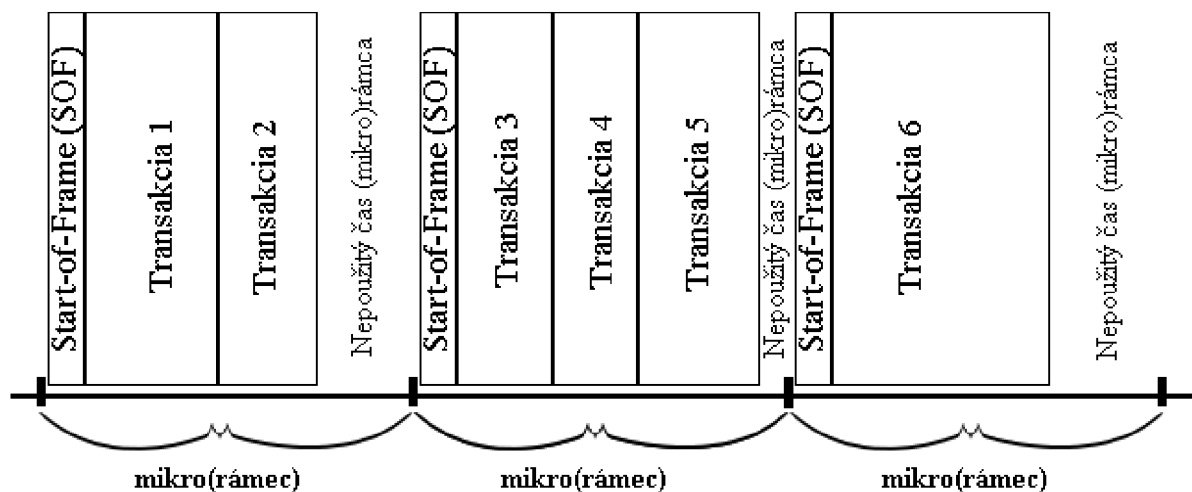
3 Komunikácia

3.1 Prenosy

Signálové linky prenášajú dáta z a do všetkých zariadení na zbernici a tvoria jednu prenosovú cestu ktorú musia všetky zariadenia zdieľať. Na rozdiel od RS-232 ktoré má TX linku na prenos dát jedným smerom a RX na prenos druhým smerom, dvojica USB liniek prenáša jeden diferenčný signál ktorého smer sa strieda.

Hostiteľ sa snaží zaistiť aby sa všetky prenosy uskutočnili čo možno najrýchlejšie. Spravuje tok dát rozdelením času na intervaly nazývané rámce (pri použití low a full speed) a mikrorámce (pri použití high speed). Hostiteľ priradí časť každého rámca alebo mikrorámca každému prenosu. Rámce majú periódu jednu milisekundu. Pre high-speed prenosy hostiteľ rozdelí každý rámec do ôsmich 125 mikrosekundových mikrorámcov. Každý rámec alebo mikrorámec začína so Start-of-frame časovou referenciou, čo je referencia ktorú hostiteľ posiela v milisekundových intervaloch pri full-speed a 125 mikrosekundových intervaloch pri high-speed. Táto Start-of-frame referencia udržuje zariadenia od vstúpenia do pozastaveného stavu keď na zbernici nie je iná aktivita. Low-speed zariadenia nevidia Start-of-frame. Namiesto toho, hub ku ktorému je zariadenie pripojené používa jednoduchšie End-of-packet signály nazývané aj low-speed udržiavajúce signály, posielané jeden krát v rámci. Tak isto ako Start-of-frame aj End-of-packet udržuje low-speed zariadenia od vstúpenia do pozastaveného stavu.

Každý prenos pozostáva z jednej alebo viacerých transakcií. Kontrolné prenosy majú vždy viacero transakcií, pretože majú niekoľko etáp z ktorých každá sa skladá z jednej alebo viacerých transakcií. Ostatné prenosy majú viac etáp iba ak je potrebné preniesť viac dát ako sa zmestí do jednej transakcie. Podľa toho ako hostiteľ naplánuje transakcie a podľa rýchlosti odpovedi zariadenia, môžu byť prenosové transakcie v jednom rámci či mikrorámci alebo môžu byť rozdelené medzi viaceré (mikro)rámce.



Obrázok č. 4: Schéma prenosových rámcov

USB komunikáciu môžeme rozdeliť do dvoch kategórií. Komunikácia používaná na enumeráciu zariadenia a komunikácia používaná na vlastnú funkciu zariadenia.

Najskôr treba vysvetliť dva dôležité pojmy.

3.2 Endpointy - zdroje a ciele dát

Pretože všetka výmena dát prebieha prostredníctvom jedného kanálu, každý prenos musí obsahovať adresu zariadenia ktorá jednoznačne určuje cieľ dát. Každé zariadenie má jedinečnú adresu priradenú hostiteľom a všetky dáta idú smerom z alebo ku nemu. Každý prenos začína tým, že hostiteľ vyšle blok dát obsahujúci adresu prijímajúceho zariadenia a špecifické umiestnenie v rámci zariadenia ktoré sa nazýva endpoint.

Všetky prenosi na zbernici idú z alebo do endpointu. Je to zásobník ktorý uchováva viacero bytov. Obvykle je to blok pamäte alebo register na čipe kontroléra. Dáta uložené v endpointe môžu byť prijaté alebo čakajúce na odoslanie. Hostiteľ má tiež zásobník dát na prijaté a odosielané dáta ale nie je to endpoint. Miesto toho hostiteľ slúži ako štart a cieľ pre komunikáciu s endpointami zariadení.

Adresa endpointu pozostáva z jeho čísla a smeru. Číslo je hodnota od 0 do 15. Smer je definovaný z perspektívy hostiteľa. IN-vstupný endpoint poskytuje dáta na odoslanie hostiteľovi a OUT-výstupný endpoint ukladá dáta prijaté. Endpoint konfigurovaný pre kontrolné prenosi musí prenášať dáta v oboch smeroch, takže v skutočnosti pozostáva z páru vstupnej a výstupnej adresy ktorá zdieľa jedno číslo endpointu. Každé zariadenie musí mať endpoint 0 nastavený ako kontrolný.

Ostatné typy prenosov posielajú dáta len jedným smerom ale stavové a kontrolné informácie môžu ísť opačným smerom. Jedno číslo endpointu môže mať vstupnú aj výstupnú adresu. Napríklad zariadenie môže mať endpoint 1 IN adresu pre posielanie dát hostiteľovi a endpoint 1 OUT adresu pre príjem dát.

Okrem endpointu 0, full a high speed zariadenie môže mať spolu až 30 endpointov (od 1 do 15, každý podporujú dva smery IN a OUT). Low speed zariadenie je obmedzené na dva endpointy v akejkoľvek kombinácii smerov (napríklad ep 1 IN a ep 1 OUT alebo ep 1 IN a ep 2 IN). Každý prenos na zbernici začína paketom obsahujúcim adresu zariadenia, číslo endpointu, kód ktorý indikuje smer toku dát a informáciu či je alebo nie je inicializovaný kontrolný prenos. Kódy sú IN, OUT a Setup.

Typ transakcie	Zdroj dát	Typy prenosov, ktoré používajú tento typ	Obsah
IN	zariadenie	všetky	dáta alebo informácie o stave
OUT	hostiteľ	všetky	
Setup	hostiteľ	kontrolné prenosi	požiadavka

Tabuľka č. 1: Typy transakcií

Tak isto ako pri smere endpointov, pomenovanie pre IN a OUT transakcie vychádza z perspektívy hostiteľa. Pri IN transakcii dáta idú smerom zo zariadenia k hostiteľovi a pri OUT transakcii je to naopak. Transakcia Setup je ako OUT, pretože dáta idú od hostiteľa k zariadeniu, ale Setup transakcia je špeciálna lebo zahajuje kontrolný prenos. Zariadenia musia vedieť identifikovať Setup transakcie, lebo tieto sú jediným typom, ktorý musia vždy prijať a odpovedať na ne. Akýkoľvek iný prenos môže použiť IN alebo OUT transakcie.

Každá transakcia obsahuje adresu zariadenia a adresu endpointu. Keď zariadenie prijme OUT alebo Setup paket obsahujúci jeho adresu, endpoint uloží dáta ktoré nasledujú a hardware obvykle spustí prerušenie. Následne obslužná rutina pre prerušenie môže spracovať prijaté dáta a vykonať požadovanú akciu. Keď zariadenie prijme IN paket obsahujúci jeho adresu, tak ak má dáta čakajúce

na odoslanie, hardware ich pošle zo špecifikovaného endpointu na zbernicu a obvykle spustí prerušenie. Obslužná rutina pre prerušenie potom urobí všetko čo je potrebné aby bolo zariadenie pripravené na ďalšiu IN transakciu.

3.3 Dátové trubice – spojenie endpointov a hostiteľa

Predtým, než sa môže uskutočniť prenos, medzi hostiteľom a zariadením sa musí vytvoriť dátová trubica. Je to pomenovanie pre spojenie medzi endpointom zariadenia a softwarom hostiteľského kontroléra. Hostiteľ zriadi trubicu počas enumerácie. Ak odoberieme zariadenie zo zbernice, hostiteľ nepotrebné trubice odstráni. Môže takisto vytvoriť nové alebo odstrániť nepotrebné aj inokedy pri požadovaní inej konfigurácie alebo iného rozhrania zariadenia. Každé zariadenie má štandardnú kontrolnú trubicu ktorá používa endpoint 0.

Konfiguračné informácie, ktoré hostiteľ získa zahŕňajú deskriptor pre každý endpoint ktorý chce zariadenie používať. Tento deskriptor je blok informácií z ktorých hostiteľ vie všetko čo potrebuje o endpointe aby s ním mohol komunikovať. Zahŕňa adresu endpointu, typ používaného prenosu, maximálnu veľkosť dátových paketov a v prípade potreby aj želaný interval pre prenosy.

Pri podrobnejšom klasifikovaní trubíc ich môžeme rozdeliť podľa toho či informácia ide jedným alebo oboma smermi na stream a message, čiže s tokom dát alebo určitými správami.

Kontrolné prenosy používajú obojsmerné trubice so správami (message pipes).

Každý začína nastavovacou Setup transakciou obsahujúcou požiadavku. Na dokončenie prenosu si hostiteľ a zariadenie vymenia dáta a stavové informácie, alebo len stavové informácie. Každý kontrolný prenos má minimálne jednu transakciu ktorá posiela informácie v každom smere.

Všetky ostatné prenosy používajú trubice stokom dát. V takejto trubici dáta nemajú definovanú štruktúru. Prijímacia strana (hostiteľ alebo zariadenie) prijme všetko čo príde. Firmware zariadenia alebo software hostiteľa potom môže dáta spracovať akýmkoľvek vhodným spôsobom. Samozrejme aj pri toku dát sa obe strany musia dohodnúť na nejakom formáte. Napríklad, aplikácia môže požadovať zaslanie série bytov indikujúcich teplotu a čas. Mohol byť použitý aj kontrolný prenos s výrobcom definovanou požiadavkou Get_temperature, ale prerušovaný (interrupt) prenos môže byť vhodnejší kôli jeho garantovanej šírke pásma.

Predtým ako obslužná aplikácia môže komunikovať s pripojeným zariadením, hostiteľ potrebuje získať od zariadenia informácie, nastaviť potrebné parametre a priradiť ovládač. Enumerácia je proces ktorý zahŕňa všetky tieto úlohy. Patrí medzi ne priradenie adresy zariadeniu, prečítanie deskriptorov, priradenie a načítanie ovládača, výber konfigurácie ktorý zodpovedá energetickým nárokom zariadenia, endpointom a ostatným funkciami. Potom je zariadenie pripravené prenášať dáta cez ktorýkoľvek endpoint v jeho konfigurácii.

3.4 Proces enumerácie

Jednou z povinností hubu je detekovanie pripojenia a odpojenia zariadení. Každý hub má prerušenie IN endpoint pre hlásenie takejto udalosti hostiteľovi. Pri štarte systému hostiteľ zisťuje od svojho koreňového hubu či sú pripojené nejaké zariadenia, vrátane prídavných hubov a zariadení pripojených na ne. Po štarte sa hostiteľ periodicky dotazuje na akékoľvek nové pripojené alebo odpojené zariadenia.

Pri zistení nového zariadenia hostiteľ pošle hubu na ktorý je zariadenie pripojené sériu požiadaviek ktorými sa vytvorí komunikačná cesta medzi hostiteľom a zariadením. Nasleduje pokus o enumeráciu zariadenia posielaním kontrolných prenosov obsahujúcich štandardné USB požiadavky na endpoint 0 daného zariadenia. Všetky USB zariadenia musia podporovať kontrolné prenosy, štandardné požiadavky a endpoint 0. Pre úspešnú enumeráciu zariadenie musí odpovedať na každú požiadavku zaslaním žiadaných informácií alebo vykonaním požadovanej akcie.

Z používateľského hľadiska je enumerácia neviditeľná a automatická okrem oznamu hlásiaceho že bolo pripojené nové zariadenie a či bola jeho inštalácia úspešná. Pri prvom použití zariadenia pre ktoré nemá operačný systém ovládače je potrebný zásah užívateľa a ich poskytnutie. Keď je enumerácia kompletná, operačný systém pridá zariadenie do správcu zariadení a takisto ho odtiaľ odoberie ak užívateľ zariadenie odpojí.

Kroky enumerácie

Špecifikácia USB definuje šesť stavov zariadenia. Počas enumerácie sa zariadenie dostáva cez nasledovné: Napájané, predvolené, adresované a konfigurované. Ďalšie dva sú pripojené a pozastavené. V každom stave má zariadenie má definované schopnosti a správanie.

Nasledovné kroky sú v typickom poradí v akom prebieha enumerácia pod Windows. Firmware zariadenia však nesmie predpokladať že požiadavky a stavy budú nasledovať v špecifickom poradí. Zariadenie musí detekovať a odpovedať na všetky kontrolné požiadavky alebo iné deje na zbernici kedykoľvek.

1. Detekcia zariadenia. Zariadenia ktoré majú zabudovanú podporu high-speed sa pripájajú ako full-speed.
2. Hostiteľ sa dozvie o novom zariadení. Každý hub používa svoje prerušenie endpoint na hlásenie nových udalostí.
3. Hub detekuje či je zariadenie low alebo high speed. Hub 1.x detekuje rýchlosť zariadenia po resete zbernice. Hub 2.0 detekuje pred resetom aby vedel, či má overiť kompatibilitu zariadenia s high-speed počas resetu ako bude popísané nižšie.
4. Hub resetuje zariadenie. Reset je špeciálny stav kedy sú obe signálové linky D+ aj D- na nízkej logickej úrovni. Hub drží tento stav minimálne 10 milisekúnd. Hub resetuje iba nové zariadenie, ostatné huby a zariadenie na zbernici rest neovplyvní.
5. Hostiteľ zisťuje či zariadenie full-speed podporuje high-speed. Na toto sa používa metóda cvrlikania (chirp). Sú definované dva stavy, chirp J a chirp K ako diferenčné jednosmerné napätie. V stave chirp J je linka D+ kladnejšia ako D- a v stave chirp K je D- kladnejšia ako D+. Počas resetu, zariadenie podporujúce high-speed vyšle chirp K. Hub schopný rozpoznať high-speed zariadenie odpovie sériou meniacich sa chirp K a chirp J. Ak zariadenie rozpozna vzorec KJKJKJ, prejde z full-speed komunikácie na high-speed. Ak hub neodpovie na vyslaný chirp K, zariadenie vie, že musí pokračovať v komunikácii rýchlosťou full-speed, preto všetky high-speed zariadenia musia byť schopné odpovedať na enumeráciu vo full-speed. Pri pripojení cez USB 3.0 vzhľadom k tomu, že používa ďalšie samostatné vedenie k tomu ktoré používa USB 2.0 a USB 1.x, nie je takáto signalizácia o šírke pásma potrebná.
6. Hub vytvorí signálovú cestu medzi zariadením a zbernicou a hostiteľ overuje, či sa zariadenie už resetovalo. Ak je potrebné, hostiteľ opakuje požiadavku kým zariadenie reset skončí.

7. Hostiteľ vyšle na adresu 0 endpoint 0 požiadavku na zistenie maximálnej veľkosti paketu pre štandardnú dátovú cestu. Pretože hostiteľ v rovnaký čas enumeruje iba jedno zariadenie, iba jedno zariadenie odpovie na komunikáciu adresovanú na adresu 0 aj keby bolo pripojených viacero nových zariadení súčasne.
8. Hostiteľ priradí zariadeniu unikátnu adresu. Od tohto bodu sa všetka komunikácia smeruje na novú adresu. Tá je platná kým je zariadenie pripojené, kým nie je vykonaný reset portu alebo sa nereštuje celý systém.
9. Hostiteľ zistí schopnosti zariadenia. Vyšle požiadavku Get_Descriptor na novú adresu. Deskriptor je dátová štruktúra obsahujúca maximálnu veľkosť paketu pre endpoint 0, počet konfigurácií ktoré zariadenie podporuje a ostatné základné informácie.
10. Hostiteľ priradí a načíta ovládač zariadenia (okrem združených zariadení). Po zistení informácií o zariadení z jeho deskriptorov, hľadá najlepší ovládač pre zariadenie. Pri výbere sa Windows snaží o najlepšiu zhodu výrobcovho ID, ID produktu alebo čísla vydania so svojimi INF súborami. Ak sa v ničom nezhoduje, pokračuje v hľadaní zhody s akoukoľvek triedou, podtriedou a protokolom zariadenia. Ak bolo zariadenie niekedy predtým úspešne enumerované, Windows použije informácie z registrov namiesto hľadania v INF súboroch. Výnimkou z tohto postupu sú združené zariadenia, ktoré môžu mať rozdielne ovládače priradené rozdielnym rozhraniam v konfigurácií. Hostiteľ môže priradiť tieto ovládače iba ak sú rozhrania povolené, čo vyžaduje aby bolo zariadenie nakonfigurované.
11. Ovládač zvolí konfiguráciu. Po zistení potrebných informácií z deskriptorov zariadenia si ovládač vyžiada konfiguráciu poslaním požiadavky Set_Configuration s želaným číslom konfigurácie. Niektoré zariadenia podporujú iba jednu konfiguráciu. Ak má zariadenie viacero konfigurácií, ovládač sa môže rozhodnúť ktorú si vyžiada na základe informácií ktoré má o tom ako bude zariadenie využívané, alebo sa môže pýtať užívateľa, alebo zvoliť prvú. Pre združené zariadenia hostiteľ v tomto bode priradí ovládače. Tak ako s inými, podľa informácií získaných od zariadenia hostiteľ nájde príslušný ovládač pre každé rozhranie v konfigurácii.

Dva zostávajúce stavy sú pripojené a pozastavené.

Pripojený stav – Ak hub na linke Vbus neposkytuje energiu pre zariadenie, je v pripojenom stave. Môže to nastať ak hub zistil príliš veľký odber prúdu alebo ak hostiteľ požiadala o odpojenie energie od portu. Bez energie na Vbus linke zariadenie nemôže s hostiteľom komunikovať, čiže ako keby nebolo vôbec pripojené.

Pozastavený stav – Zariadenie vstupuje do pozastaveného stavu potom ako nie je na zbernici žiadna aktivita dlhšie ako 3 milisekundy. V tomto stave musí obmedziť svoj odber energie. Pozastavený stav musia podporovať nakonfigurované aj nenakonfigurované zariadenia.

Enumerácia hubu

Huby sú tiež USB zariadenia a hostiteľ enumeruje pripojený hub tak isto ako iné zariadenia. Ak má hub pripojené ďalšie zariadenia, hostiteľ enumeruje každé z nich po tom ako ho hub informuje o ich prítomnosti.

Odpojenie zariadenia

Keď používateľ odstráni zariadenie zo zbernice, hub zakáže port zariadenia. Hostiteľ zistí že zariadenie bolo odstránené keď sa dotazuje hub na stav, zistí že sa stala nejaká udalosť a vyslaním požiadavky `Get_Port_Status` zisťuje aká udalosť to bola. Operačný systém odstráni zariadenie zo Správcu zariadení a jeho adresa je použiteľná pre novo pripojené zariadenie.

3.5 Komunikácia používaná pri vlastnej funkcii zariadenia

USB je navrhované pre zvládnutie rôznych typov periférií s odlišnými nárokmi na prenosovú rýchlosť, čas odozvy a korekciu chýb. Existujú štyri typy dátových prenosov, každý má iné parametre a každý slúži pre rozdielne použitie a účel zariadenia.

Kontrolné (control) prenosy sú jediným typom ktorého funkciu definuje USB špecifikácia. Kontrolné prenosy dovoľujú hostiteľovi zistiť informácie o zariadení, nastavenie jeho adresy, výber jeho konfigurácie a iné nastavenia.

Hromadné (bulk) prenosy sú zamerané na situáciu kedy prenosová rýchlosť nie je kritická, ako napríklad posielanie dát pre tlačiareň, príjem dát od skenera alebo prístup k súborom na disku. Pre tieto aplikácie sú rýchle prenosy vítané, ale dáta môžu v prípade potreby počkať. Ak je zbernica obsadená, hromadné prenosy sú odložené ale v prípade že nie je sú veľmi rýchle. Iba high-speed a full-speed zariadenia podporujú hromadné prenosy. V globále sa od nich podpora hromadných prenosov nevyžaduje, ale špecifická trieda zariadení ho môže potrebovať.

Prerušované (interrupt) prenosy sú pre zariadenia, ktoré potrebujú periodickú pozornosť hostiteľa. Na rozdiel od kontrolných prenosov, prerušovaný prenos je jediný, ktorý môžu využiť low-speed zariadenia. Klávesnice a myši používajú tento prenos na periodické odosielanie dát o stlačených klávesách alebo o polohe ukazovateľa. Prerušované prenosy môžu využívať akúkoľvek rýchlosť. Od zariadení sa nevyžaduje podpora prerušovaných prenosov, ale špecifická trieda zariadení ich môže podporovať.

Izochronne prenosy majú garantovaný čas doručenia, ale nemajú opravu chýb. Dáta ktoré využívajú tieto prenosy sú väčšinou audio a video prehrávané v reálnom čase. Izochronne prenosy sú jediným typom ktoré nepodporujú automatický opätovný prenos chybne prijatých dát, takže príležitostné chyby musia byť akceptovateľné. Používajú ich iba high-speed a full-speed zariadenia. Takisto sa nevyžaduje ich podpora, ale špecifické triedy zariadení môžu izochronne prenosy vyžadovať.

Typ prenosu	Maximálna prenosová rýchlosť na endpoint (v kilobytoch za sekundu)		
	Low Speed	Full Speed	High Speed
kontrolný	24	832	15 872
prerušovaný	0,8	64	24 456
hromadný	nie je to povolené	1216	53 248
izochrónny		1023	24 546

Tabuľka č. 2: Maximálne prenosové rýchlosti pre jednotlivé typy prenosov a rýchlosti zbernice

4 Možnosti vyhotovenia prepínača

Primárne určenie prepínača je zdieľanie viacerých tlačiarňí medzi viacerými počítačmi. Keďže USB bolo navrhované ako štruktúra jedného hlavného hostiteľského kontroléra v pozícii správcu zbernice ku ktorému sa pripájajú všetky ostatné zariadenia ako podriadené, nedá sa pre takúto funkciu použiť klasický hub. Úlohou preto bolo nájsť riešenie umožňujúce súčasné pripojenie viacerých hostiteľských kontrolérov z ktorých každý by mohol ovládať ku ktorému počítaču bude tlačiareň alebo iné USB zariadenie pripojené. Pre pripojenie budú použité dva USB káble, jeden k ovládaniu prepínača a druhý pre výmenu dát s koncovým zariadením.

Možnosti vyhotovenia

Hlavným faktorom od ktorého sa odvíjajú vlastnosti prepínača je dostupnosť USB multiplexerov a počtom portov ktoré dokážu prepínať. Aby prepínač mal čo najväčšie možnosti použitia, rozhodol som sa zamerať na multiplexery s pomerom 1:4 a ten ktorý som použil som vyberal z nasledovných:

FSUSB74 – High Speed USB 2.0 Switch/Multiplexer od spoločnosti Fairchild Semiconductor

PI3USB14 – 3.3V, 2-Channel, 4:1 Mux USB 2.0 Switch od spoločnosti Pericom

MAX4899AE – USB 2.0 High-Speed, Fault-Tolerant 4:1 Multiplexer od spoločnosti Maxim

Obvod PI3USB14 nebol v dobe konštrukcie k dispozícii a zo zostávajúcich možností som vybral multiplexer MAX4899AE hlavne kôli vlastnostiam ako veľmi nízky prevádzkový prúd (iba 200 uA), širšiemu prenosovému pásmu (425 Mhz oproti 300 Mhz), väčšej ochrane proti elektrostatickému výboju a širším možnostiam napájania. [15]

Ďalším faktorom ovplyvňujúcim funkčnosť prepínača je spôsob akým bude multiplexer ovládaný. Tu je tiež na výber viacero možností.

4.1 USB prepínač s Atmega 8u2

Riešenie vychádza s použitím mikrokontroléra Atmega 8u2 s hardwarovou podporou USB štandardu. Každá PC stanica by bola pripojená prostredníctvom jedného USB portu k mikrokontroléru a druhým USB portom k jednému zo štyroch vstupov multiplexera. Výsledná funkcia celého systému USB prepínača by bola nasledovná:

- Každý mikrokontrolér by obsluhoval iba jednu PC stanicu.
- Topologicky by bol jeden mikrokontrolér MASTER a zvyšné SLAVE (na zbernici SPI).
- MASTER by v pravidelných časových intervaloch odosielať požiadavky na zistenie stavu jednotlivých SLAVE-ov
- Pri zistení požiadavky na zmenu stavu by rozposlal príkazy pre jednotlivé SLAVE tak aby nedošlo k predčasnemu odpojeniu aktuálne pripojeného zariadenia (čo by mohlo viesť k strate dát).
- MASTER by následne nastavil vstupný a výstupný port oboch USB multiplexerov

Výsledné výhody a nevýhody tejto varianty by boli:

- + Nízka cena
- Nutnosť vytvorenia dvoch zdrojových kódov, samostatne pre MASTER a samostatne pre SLAVE
- Zbytočne veľké množstvo mikrokontrolérov s nevyužitou výkonovou kapacitou
- Nutný samostatný programátor
- Veľké množstvo súčiastok
- Zložitý návrh DPS

4.2 USB prepínač s FTDI 232 RL

Riešenie vychádza s použitím USB prevodníka FT 232 RL. Každá PC stanica by bola pripojená prostredníctvom jedného USB portu k samostatnému USB prevodníku. Každý by obsahoval naprogramovaný riadiaci algoritmus. Rozdielom oproti predchádzajúcej variante je skutočnosť, že každý prevodník by mal možnosť ovládať prepínanie vstupov a výstupov USB multiplexerov. Výsledná funkcia celého systému USB prepínača by bola nasledovná:

- Každý USB prevodník by obsluhoval iba jednu PC stanicu.
- Topologicky by bol každý MASTER
- Neprebíhala by komunikácia po zbernici, iba by sa riadiacim algoritmom vyhodnocovalo obsadenie vstupu iným USB prevodníkom pomocou jedného spoločného príznakového vodiča
- Ak by bol príznak obsadenia v aktívnom stave, musí daný USB prevodník počkať na uvoľnenie príznaku a možnosť riadiť multiplexery. Ak by bol príznak v neaktívnom stave, príslušný prevodník prevezme kontrolu nad multiplexermi a nastaví príznak obsadenia do aktívnej úrovne.

Výsledné výhody a nevýhody tejto varianty by boli:

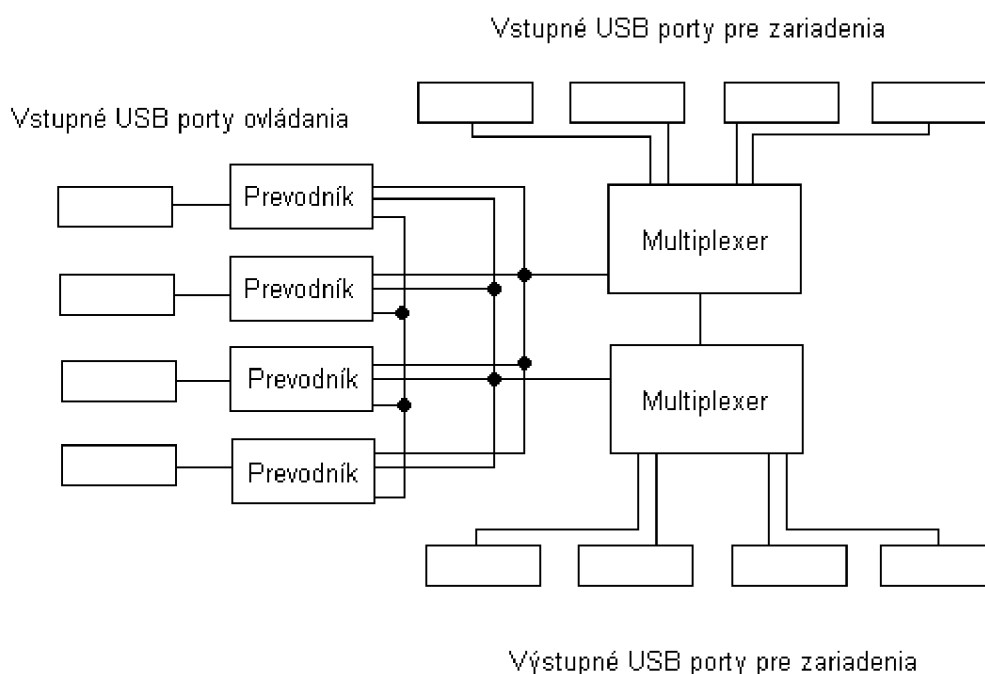
- + Vytvorenie iba jediného riadiaceho programu pre všetky FTDI prevodníky.
- + Programovanie priamo cez USB.
- + Menší počet súčiastok a obvodov.
- + Jednoduchší návrh DPS
- O niečo zložitejší spôsob programovania
- Vyššia cena

Po zhodnotení kladov a záporov jednotlivých variant, som sa rozhodol realizovať prepínač variantou v ktorej sú použité prevodníky FTDI 232RL.

5 Realizácia hardware

Popis zapojenia prepínača

Prepínač môžeme rozdeliť na dve časti. Prvá časť má na starosti komunikáciu s počítačom riadiacim prepínanie a druhá časť samotné prepínanie vykonáva.

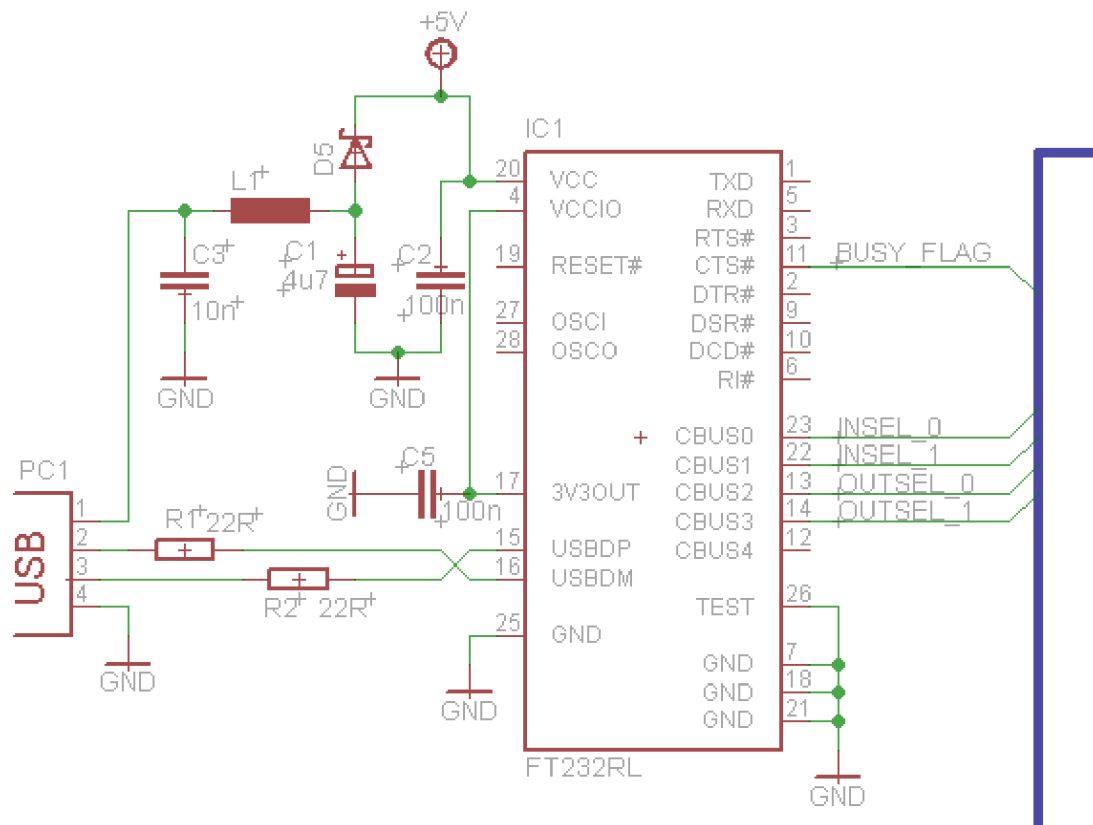


Obrázok č. 5: Blokova schéma zapojenia

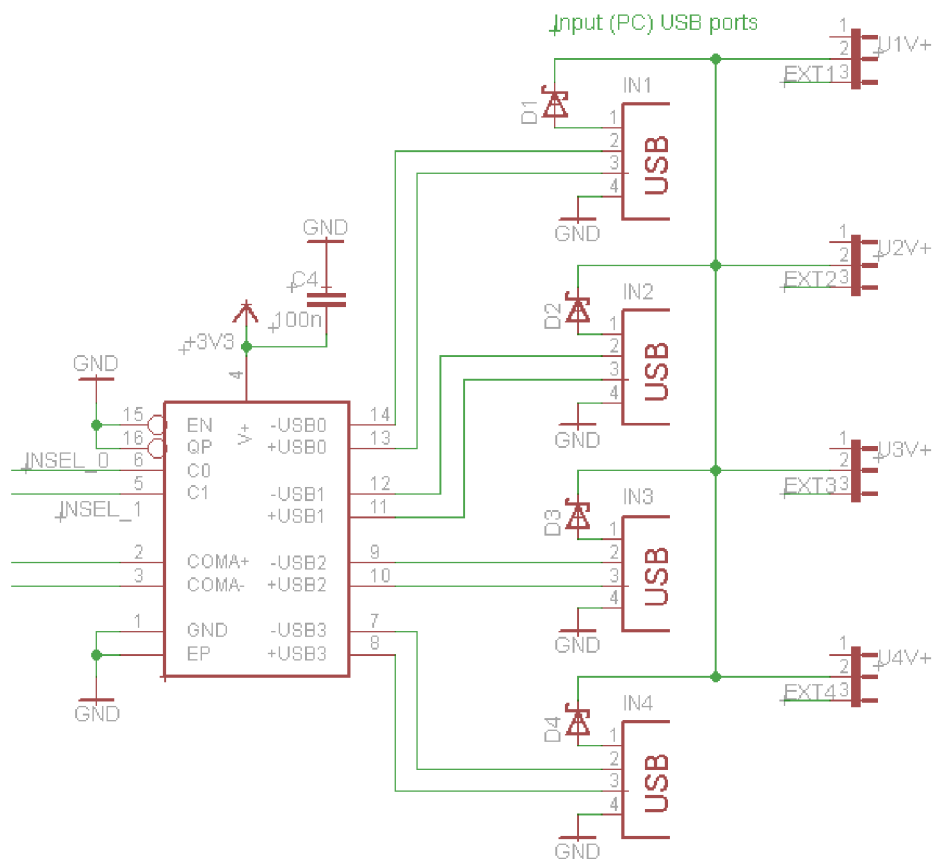
Časť komunikácie s riadiacim počítačom obsahuje štyri identické zapojenia. Každé začína USB portom typu B, z ktorého sú signálové linky D+ a D- cez 22 ohmové odpory pripojené k prevodníku FT232RL. Napájací vodič z USB portu je cez odrušovací LC filter tvorený kondenzátorom 10nF a indukčnosťou 100uH privedený na filtračný 4,7 uF a blokovací 100 nF kondenzátor a napája prevodník. Pomocou diódy sú jednotlivé napájania poskytované zbernicou USB od seba oddelené a privedené na vstup stabilizátora LF33 ktorý znižuje 5V napätie zo zbernice na 3,3V ktorými sú napájané multiplexery.

Pin VCCIO je na prevodníku spojený s pinom 3V3OUT kôli tomu aby na vstupno-výstupných pinoch prevodníka bolo napätie s logickými úrovňami na 3,3V čo presne vyhovuje potrebám pre riadenie multiplexerov.

Časť ktorá vykonáva samotné prepínanie je tvorená štyrmi vstupnými USB portami typu A, ktorých signálové linky D+ a D- sú privedené na jednotlivé vstupy prvého multiplexera, ktorý je dvoma signálovými linkami spojený s druhým multiplexerom. K nemu sú na štyroch výstupoch pripojené USB porty typu B. Pri každom vstupnom USB porte je jumper ktorý umožňuje zvoliť či bude pripojené zariadenie napájané alebo nie. Ak je napájanie zapnuté, indikuje to aj LED dióda pri každom porte.



Obrázok č. 6: Schéma zapojenia prevodníka



Obrázok č. 7: Schéma zapojenia vstupného multiplexera

Multiplexery sa ovládajú pomocou štyroch vodičov (dva pre každý multiplexer – na obrázku INSEL_0 a INSEL_1) ktoré sú pripojené ku všetkým prevodníkom. Na týchto vodičoch sú pull up odpory a sú pripojené na 3,3V pre zaručenie stabilnej logickej úrovne. Pomocou jej zmeny prevodníky ovládajú nastavenie vstupného a výstupného portu. Vodiče sú pripojené na piny prevodníka CBUS0 až 3. Tieto piny sú programovateľné a dá sa ovládať či budú ako vstupné alebo výstupné. Aby nedochádzalo k vzniku rušivých signálov sú piny CBUS nastavené ako výstupné iba ak prevodník ovláda prepínanie. Ak nie, sú nastavené ako vstupné.

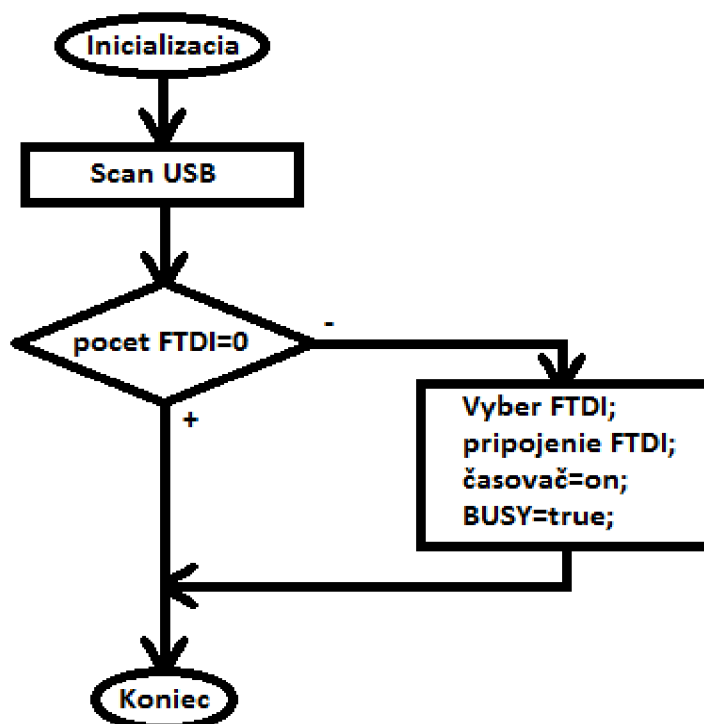
Na vodič BUSY_FLAG sú pripojené všetky prevodníky, signalizujú si pomocou neho obsadenie ovládania prepínača. Pull up odpor je aj na tomto vodiči. Ak je obsadené, svieti príslušná LED dióda.

V návrhu je použitých viacero smd odporov s hodnotou 0 ohmov, pomocou nich som bol schopný „preskočiť“ vodivé spojenie na plošnom spoji a vyhnúť sa tak prekoveniam. Vhodným umiestnením súčiastok a použitím desiatich spojok vyrobených z bežného vodiča sa podarilo navrhnuť iba jednostrannú dosku plošného spoja. Schéma zapojenia bola vytvorená programom EAGLE 5.7.0 ktorá aj vo verzii Light ktorá je dostupná zdarma plne postačovala potrebám pre túto prácu.[16] Keďže multiplexer MAX 4899 AE nebol obsiahnutý v knižniciach súčiastok programu EAGLE, bolo treba preň vytvoriť novú knižnicu so schematicou značkou a púzdom.

6 Realizácia software

6.1 Obslužný program

Program pre prepínač bol navrhovaný a programovaný pomocou Microsoft Visual Studio 2010. Popis jeho fungovania je nasledovný:



Obrázok č. 8: Vývojový diagram inicializácie

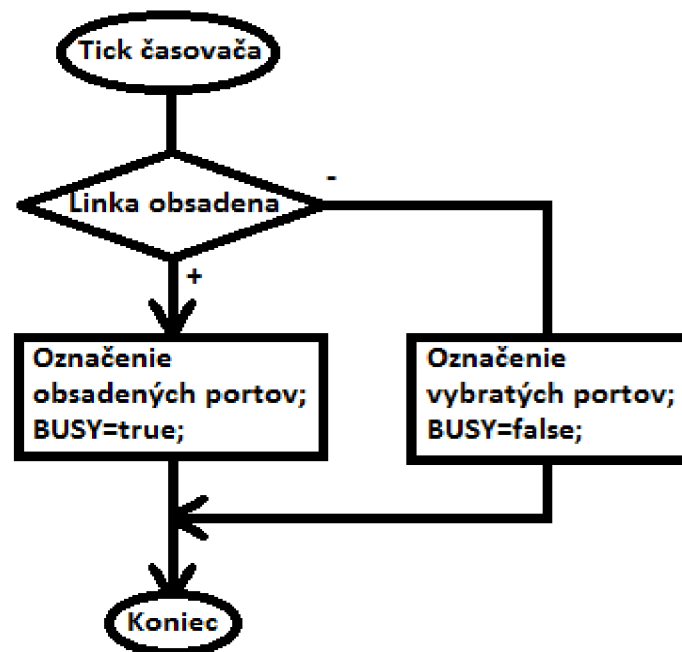
Po spustení ovládacieho programu sa vykoná funkcia ktorá preskenuje USB porty. Nájde dostupné prevodníky FTDI, zistí si ich parametre a zoradí ich do výberového zoznamu. Pri zistenom pripojení k prevodníku sa nastaví všetky grafické indikátory v programe na základné hodnoty, zmaže sa príznak obsadenia (busy), zakáže sa niektoré ovládacie prvky, vypne sa časovač a korektne sa uzavrie spojenie s FTDI. Tým je prevodník a ovládací program nastavený do východzej pozície.

Následne sa prevodník inicializuje, nastaví sa do bit-bang asynchrónneho módu. Všetky piny sa definujú ako vstup, povolí sa časovač, tlačidlá v ovládacom programe, drop-down list svýberom pripojených prevodníkov.

Bit-bang mód je špeciálny mód prevodníka, kedy je osem vstupno-výstupných pinov zmenených na 8-bitovú obojsmernú dátovú zbernicu. V asynchrónnom móde sa všetky dáta odoslané do prevodníka automaticky prenesú na piny ktoré boli definované ako výstupné. Každý pin môže byť nezávisle nastavený ako vstup alebo výstup. Frekvenciu ktorou sa dáta menia na výstupných pinoch riadi generátor prenosovej rýchlosti. Aby sa dáta na výstupe zmenili, musia byť do prevodníka odoslané nové a generátor musí prejsť jeden takt. Ak nie sú do prevodníka odoslané ďalšie dáta, na výstupné piny budú držať poslednú hodnotu ktorú mali.

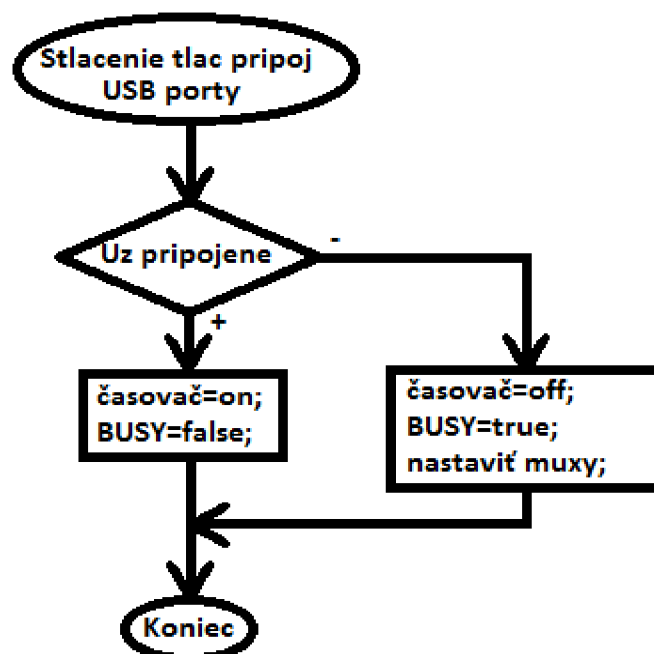
Potom začne cyklus kedy pri každom tiku časovača, ktorý je nastavený na 100 ms sa zistí stav linky indikujúcej obsadenie (busy). FTDI sa nastaví do Asynchrónneho bit-bang módu so všetkými pinmi nastavenými aby fungovali ako vstupy. Funkciou GetPinStates() sa najskôr načíta stav celého portu do premennej pinstat. Získaná hodnota sa omaskuje maskou 0x08 hexa čo odpovedá polohe pinu pripojeného na busy linku (#CTS na ftdi puzdre). Omaskovaná hodnota sa porovná s hodnotou 0. V prípade zhody je jasné že je príznak busy nastavený iným prevodníkom FTDI a USB linka je obsadená. V tomto prípade sa v ovládacom programe busy príznak nastaví na true a zároveň sa zakáže tlačidlo buttonUSBconnect_Click, ktorým sa pripájajú jednotlivé vybraté USB porty lubovoľne medzi sebou podľa toho ako sa zvolia v príslušných drop-down listoch. (list comboBoxINPUT pre zvolenie vstupného portu (PC) a list comboBoxOUTPUT pre zvolenie portu pripojeného USB zariadenia)

Ďalej sa zisťuje, ktorý USB port (vstupný a výstupný) je práve obsadený. Toto je uskutočnené prepnutím prevodníka do CBUS bit-bang módu, čo je mód podobný ako asynchrónny bit bang, len v tomto móde sa používajú iba štyri CBUS piny (zbernica je 4 bitová) a načítaním hodnôt z pinov CBUS0 az 4. Podľa zistenej kombinácie signálov na pinoch sa určí, ktorý USB vstup (CBUS0,CBUS1) a USB výstup (CBUS2,CBUS3) sú práve používané.



Obrázok č. 9: Vývojový diagram zisťovania obsadenosti prepínača a jednotlivých portov

Pripojenie USB zariadenia (konektory OUT 1 až 4) k požadovanému USB PC portu (konektory IN 1 až 4) prebieha nasledovne. V prípade že je linka obsadená a multiplexery obsluhuje iné FTDI (PC stanica), tlačítko na pripojenie je zašednuté (nedá sa stlačiť) a nie je možné nadviazať spojenie medzi požadovanými portami. V prípade že nikto nepoužíva multiplexery, užívateľ vyberie USB porty ktoré chce spojiť z drop-down listu a tlačidlo je možné stlačiť. Po jeho stlačení sa zisťuje či sa má spojenie nadviazať alebo uzatvoriť. Ak bude mať premenná connected hodnotu true, spojenie sa bude uzatvárať, v opačnom prípade sa spojenie nadviaže.



Obrázok č. 10: Vývojový diagram nadväzovania spojenia

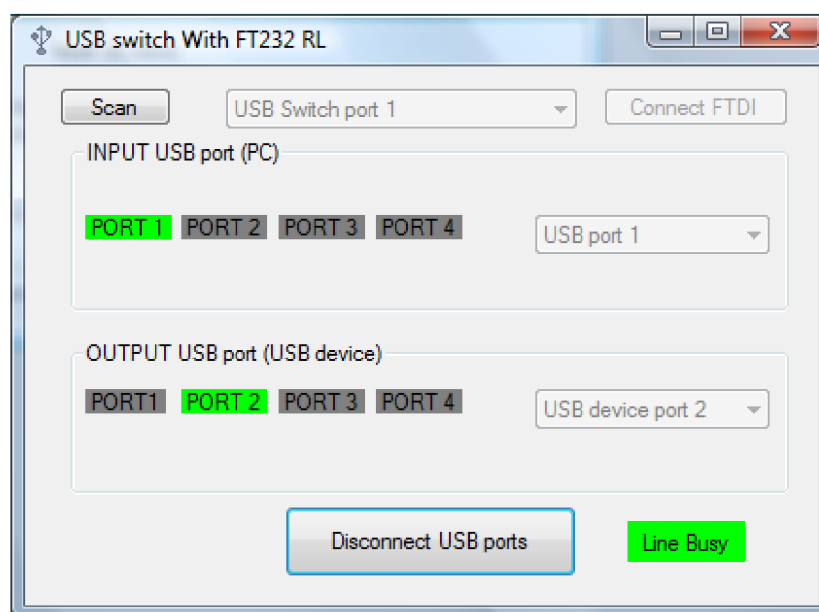
Nadviazanie spojenia prebieha v niekoľkých krokoch. V prvom kroku je potrebné zastaviť časovač ktorý v pravidelných intervaloch skenoval busy príznak (teda či je USB linka používaná iným užívateľom). Zároveň sa nastaví aj premenná `connected` na `true` (aby sa pri ďalšom stlačení spojenie ukončilo). V ďalšom kroku sa zostaví riadiace slovo pre nastavenie vstupného a výstupného USB multiplexera. Toto slovo sa zloží súčtom konštanty 240 (zaistí nastavenie pinu CBUS0 až CBUS3 ako výstupne), ďalej indexom USB portu vybraného vo výstupnom drop-down liste (`comboBoxOUTPUT.SelectedIndex`) vynásobený hodnotou 4 (na posun o dve bitové pozície doľava) a indexom USB portu vybraného vo vstupnom drop-down liste (`comboBoxINPUT.SelectedIndex`). Kde index 0 odpovedá binárnej kombinácii 00 (port 1), index 1 odpovedá binárnej kombinácii 01 (port 2) atď... Výsledné riadiace slovo sa uloží do premennej `CBUS_IO_VAL`.

Na záver sa už len nastaví prevodník FTDI do módu CBUS bit-bang a pošle sa mu riadiace slovo. Potom sa ešte nastaví busy príznak, aby bolo jasné ostatným prevodníkom ktoré by sa chceli pokúsiť pripojiť na svoje porty, že je linka obsadená.

Ukončenie spojenia prebieha jednoduchšie. Do premennej `connected` sa vloží hodnota `false` (pripojenie nie je nadviazané), zmaže sa príznak busy a spustí sa časovač ktorý bude opäť v časových intervaloch kontrolovať stav príznakovej linky (`true`, `false`).

6.2 Užívateľské rozhranie

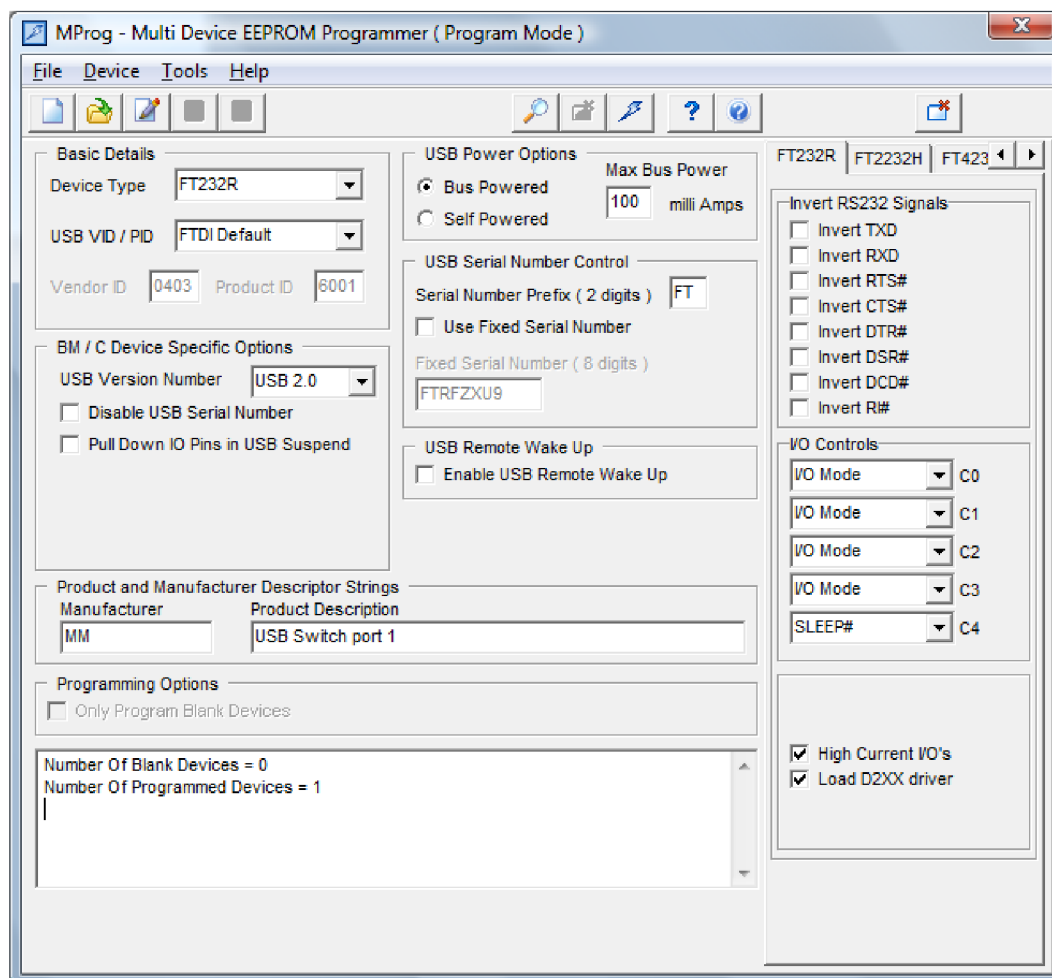
Užívateľské rozhranie je veľmi intuitívne. Po spustení ovládacieho programu najskôr zisťuje, či sú nainštalované ovládače k prevodníku FTDI. Ak nie, zobrazí varovnú hlášku. Ak chce užívateľ spojiť napríklad vstupný port 1 a výstupný port 2, stlačí najskôr tlačidlo `Scan` a obslužný program rozpozná, či a ktorý prevodník je k počítaču pripojený. Potom ho užívateľ vyberie zo zoznamu dostupných a stlačí tlačidlo `Connect FTDI`. Ak je ovládanie prepínania voľné, môže užívateľ zvoliť požadovaný vstupný a výstupný port a po stlačení tlačidla `Connect USB ports` sa spojenie nadviaže. Ak ovládanie nie je voľné, program zobrazí ktoré dva porty sú práve spojené a tlačidlo `Connect USB ports` zostane zašednuté. Ak by užívateľ označil iba jeden zo vstupných alebo výstupných portov, program opäť zobrazí varovnú hlášku.



Obrázok č. 11: Snímka ovládacieho programu – je aktívne spojenie portu 1 a 2

6.3 Firmware prepínačov

Firmware prepínačov bol programovaný pomocou nástroja MProg verzie 3.5 od výrobcu prevodníkov FTDI. [17] Pomocou neho sa dajú nastaviť vlastnosti prevodníka ako napríklad napájanie z USB zbernice, pod akým Vendor ID a Product ID sa bude hlásiť (použil som dodávané výrobcom), rýchlosť komunikácie (USB 2.0), pod akým názvom sa zobrazí prevodník v správcovi zariadení (USB Switch port 1 – pre prevodník č.1) v operačnom systéme a čo je najdôležitejšie pre správne fungovanie prepínača, akú funkciu budú mať programovateľné piny CBUS0 až 4.



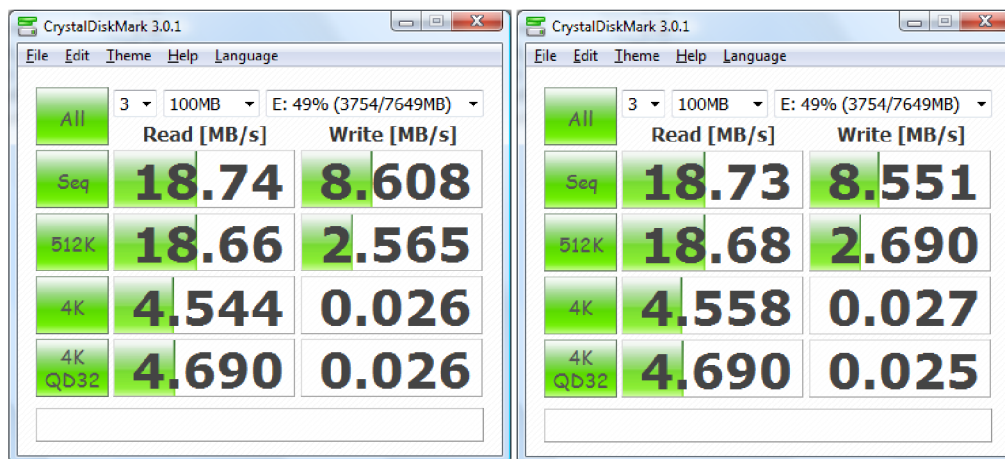
Obrázok č. 12: Program MProg 3.5 a jeho nastavenie pre prevodník č.1

6.4 Nároky na softwarové vybavenie počítača

Pre správne fungovanie obslužného programu treba mať v počítači nainštalovaný balík dotNetFx verzie 4.0 ktorý už býva súčasťou novších operačných systémov a po pripojení prevodníka k PC a jeho rozpoznaní treba inštalátoru poskytnúť ovládače D2XX.

7 Testovanie

Na testovanie funkčnosti a vlastností prepínača som použil flash disk DT Mini Slim od firmy Kingston o veľkosti 8GB a program CrystalDiskMark vo verzii 3.0.1 ktorý sa dá zdarma stiahnuť z internetu [18] a použiť na testovanie rýchlosti pevných a flash diskov v počítači. Program testuje rýchlosť sekvenčného čítania a zápisu a pre súbory o veľkosti 512KB a 4KB. Dá sa nastaviť počet opakovaní testu pre vyššiu presnosť (v tomto teste nastavená na tri) a veľkosť testovaného súboru (nastavené na 100 MB).



Obrázok č. 13: Výsledky testu rýchlosti prenosu dát. Na obrázku vľavo bol USB disk pripojený priamo na koreňový hub v PC. Na obrázku vpravo bol USB disk pripojený cez prepínač.

Zpriložených snímok je vidieť, že prepínač má na rýchlosť prenosu dát len minimálny vplyv. Testy boli realizované na notebooku s rokom výroby 2010 ktorý mal pre účely tohto testu dostatočnú výkonovú rezervu a namerané hodnoty sú maximálne prenosové rýchlosti dosiahnuteľné pre flash disk DT Mini Slim 8GB.

7.1 Zdieľanie pamäťových médií

Prepínač je na zdieľanie pamäťových médií vhodný z dvoch hlavných dôvodov. Prvý je ten, že k prepnutiu iným počítačom nedôjde skôr ako to povolí užívateľ. Pri zdieľanom prístupe k pamäťovým médiám je táto vlastnosť veľmi vítaná, lebo nemôže dôjsť k neželanému ukončeniu spojenia. A druhý je, že zdieľanie prostredníctvom prepínača znižuje mechanické opotrebenie konektorov.

Vzhľadom na to, že prepínač a pripojené zariadenia sú napájané zo zbernice USB, bez problémov sa dajú zdieľať iba usb flash disky. Externé pevné disky musia mať zabezpečený vlastný zdroj napájania.

8 Záver

Zadaním tejto bakalárskej práce bolo zoznámiť sa s protokolom USB a následne navrhnúť a skonštruovať USB prepínač na zdieľanie viacerých USB zariadení medzi viacerými počítačmi. V prvej časti sa práca venuje opisu štandardu USB a princípom jeho fungovania. V druhej časti riešení samotného prepínača. Po preštudovaní možných variant riešenia som zvolil ako hlavné stavebné prvky návrhu USB prevodník FT232RL od firmy FTDI a multiplexer MAX 4899AE od firmy Maxim. Typ použitého multiplexeru ovplyvnil maximálny počet pripojených zariadení a počítačov s nimi komunikujúcich na štyri. Spôsob akým zbernica USB pracuje ovplyvnil aj možnosti prepínača a preto

môže v jeden moment aktívne využívať jeden počítač iba jedno pripojené zariadenie. Ostatné počítače musia počkať na uvoľnenie prenosovej cesty aby nedošlo k predčasnému ukončeniu spojenia a prípadnej strate dát. Toto je ošetrené v návrhu a v ovládacom programe tak, že počítač ktorý práve ovláda prepínanie signalizuje pomocou vodiča na to určeného ostatným počítačom, že prepínač je obsadený.

Prepínač je funkčný a dá sa reálne použiť. Ideálne nasadenie je najvhodnejšie do malej kancelárie na zdieľanie tlačiarní, skenerov, usb flash diskov, hardwarových kľúčov k programom, prípadne inej kancelárskej techniky.

Zoznam použitých zdrojov

- [1] - Ajay Bhatt. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia [1] - Ajay Bhatt. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, 12 Jun 2009, last modified on 19 Maj 2011 [cit. 2011-05-24]. Dostupné z WWW: <http://en.wikipedia.org/wiki/Ajay_Bhatt>.
- [2] - KOON, John. The USB Vision: 10 Years Later. Everything USB [online]. 2005, 1, [cit. 2011-05-23]. Dostupný z WWW: <<http://www.everythingusb.com/timeline.html>> .
- [3] - AXELSON, J. USB complete: everything you need to develop custom USB peripherals. New York: Lakeview Research, 2005. 3rd ed., 572 s. ISBN 978-1-931448-02-4.
- [4] - USBlyzer [online]. 2011 [cit. 2011-05-23]. Brief USB Overview and USB History. Dostupné z WWW: <<http://www.usblyzer.com/brief-usb-overview-and-history.htm>>.
- [5] - Wireless USB. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, 3 Február 2005, last modified on 23 Apríl 2011 [cit. 2011-05-25]. Dostupné z WWW: <http://en.wikipedia.org/wiki/Wireless_USB>.
- [6] - Everything USB [online]. 2011 [cit. 2011-05-23]. SuperSpeed USB 3.0 FAQ. Dostupné z WWW: <<http://www.everythingusb.com/superspeed-usb.html#6>>.
- [7] - USBlyzer [online]. 2011 [cit. 2011-05-23]. USB Topology - A connection model between the host and USB peripherals. Dostupné z WWW: <<http://www.usblyzer.com/usb-topology.htm>>.
- [8] - BLANCHARD, Richard. What is the Life Cycle of a USB Flash Drive?. Getusb.info [online]. 2007, 1, [cit. 2011-05-23]. Dostupný z WWW: <<http://www.getusb.info/what-is-the-life-cycle-of-a-usb-flash-drive/>>.
- [9] - QUINNELL, Richard. USB: a neat package with a few loose ends. EDN Magazine [online]. 1996, 1, [cit. 2011-05-23]. Dostupný z WWW: <http://www.edn.com/archives/1996/102496/df_01.htm#USB%20fundamentals>.
- [10] - Usb.org [online]. 2007 [cit. 2011-05-23]. Deprecation Announcement 052507.pdf. Dostupné z WWW: <http://www.usb.org/developers/Deprecation_Announcement_052507.pdf>.
- [11] - Usb.org [online]. 2007 [cit. 2011-05-23]. USB-IF Compliance Updates. Dostupné z WWW: <<http://compliance.usb.org/index.asp?UpdateFile=Cables%20and%20Connectors&Format=Standard#63>>.
- [12] - Usb.org [online]. 2011 [cit. 2011-05-23]. USB Frequently Asked Questions. Dostupné z WWW: <<http://www.usb.org/developers/usbfaq>>.
- [13] - Firewall.cx [online]. 2011 [cit. 2011-05-23]. Propagation delay & its relationship to max. cable length. Dostupné z WWW: <<http://www.firewall.cx/ethernet-prop-delay.php>>.
- [14] - Usb.org [online]. 2011 [cit. 2011-05-23]. USB 3.0 Specification. Dostupné z WWW: <<http://www.usb.org/developers/docs/>>.

[15] - USB 2.0 High-Speed, Fault-Tolerant 3:1, 4:1 Multiplexers [online]. Sunnyvale, CA, USA : Maxim Integrated Products, 2006 [cit. 2011-05-23]. Dostupné z WWW: <<http://datasheets.maxim-ic.com/en/ds/MAX4899AE-MAX4899E.pdf>>.

[16] - Cadsoft.de [online]. 2011 [cit. 2011-05-23]. CadSoft Online:Freeware. Dostupné z WWW: <<http://www.cadsoft.de/freeware.htm>>.

[17] - Ftdichip.com [online]. 2011 [cit. 2011-05-23]. FTDI Utilities. Dostupné z WWW: <<http://www.ftdichip.com/Support/Utilities.htm>>.

[18] - Crystalmark.info [online]. 2011 [cit. 2011-05-23]. CrystalDiskMark - Software - Crystal Dew World. Dostupné z WWW: <<http://crystalmark.info/software/CrystalDiskMark/index-e.html>>.

[19] - KAINKA, B. USB. Měření, řízení a regulace pomocí sběrnice USB. Praha: BEN, 2002. 1. vyd. 247 s. ISBN 80-7300-073-3.

[20] - MRNÍK, Martin. Tiskový USB přepínač. Brno, 2010. 26 s. Semestrální práce. FEKT VUT v Brně.

Obsah priloženého CD

Na kompaktnom disku ktorý je prílohou tejto práce sú súbory:

\program – zložka obsahujúca ovládací program

\FTDI sw – zložka obsahujúca zdrojové kódy obslužného programu

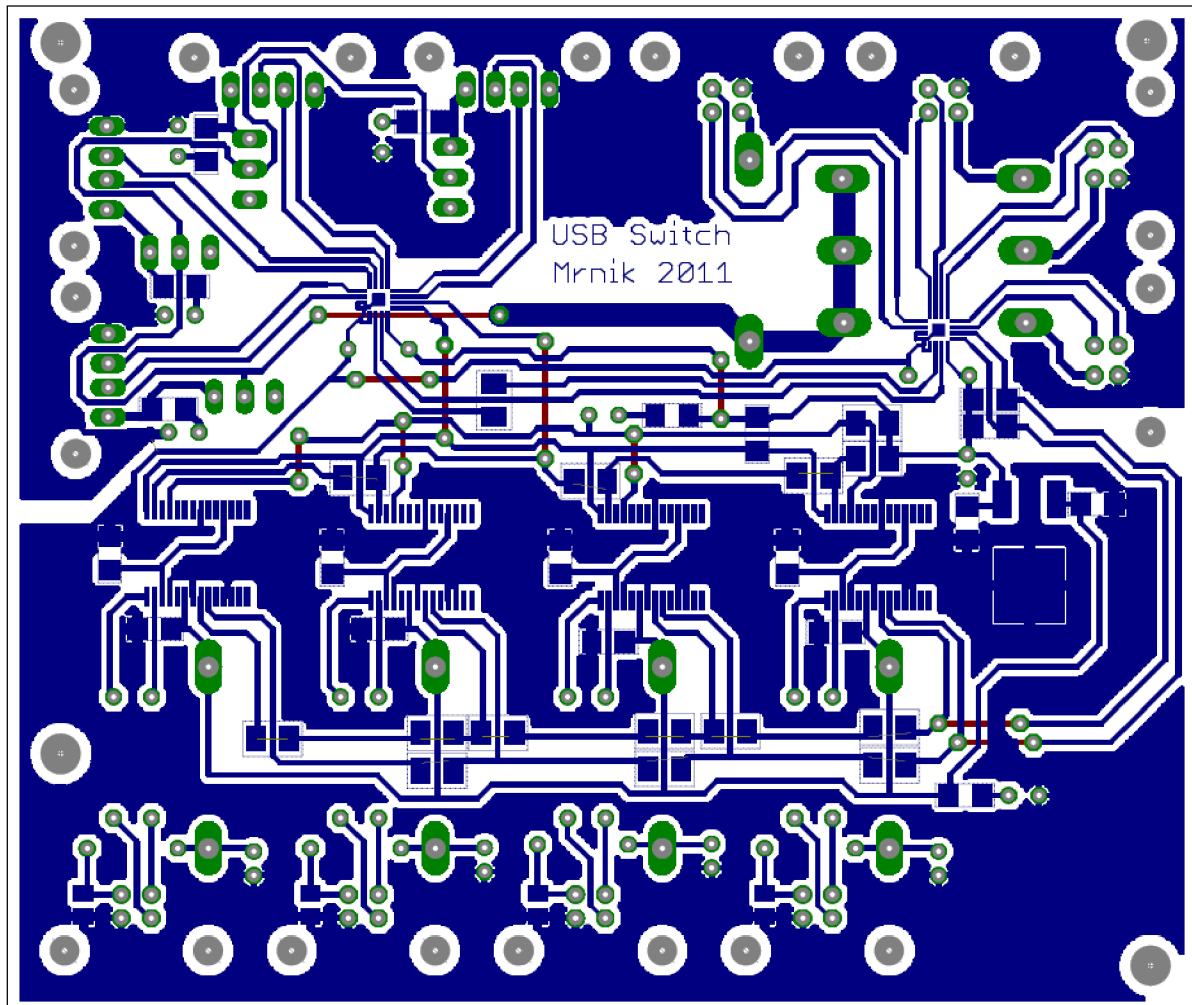
\eagle – zložka obsahujúca elektronickú verziu schémy zapojenia, dosky plošného spoja a vlastnú knižnicu pre obvod MAX4899AE

\ovladace – zložka obsahujúca ovládače k prevodníku pre windows

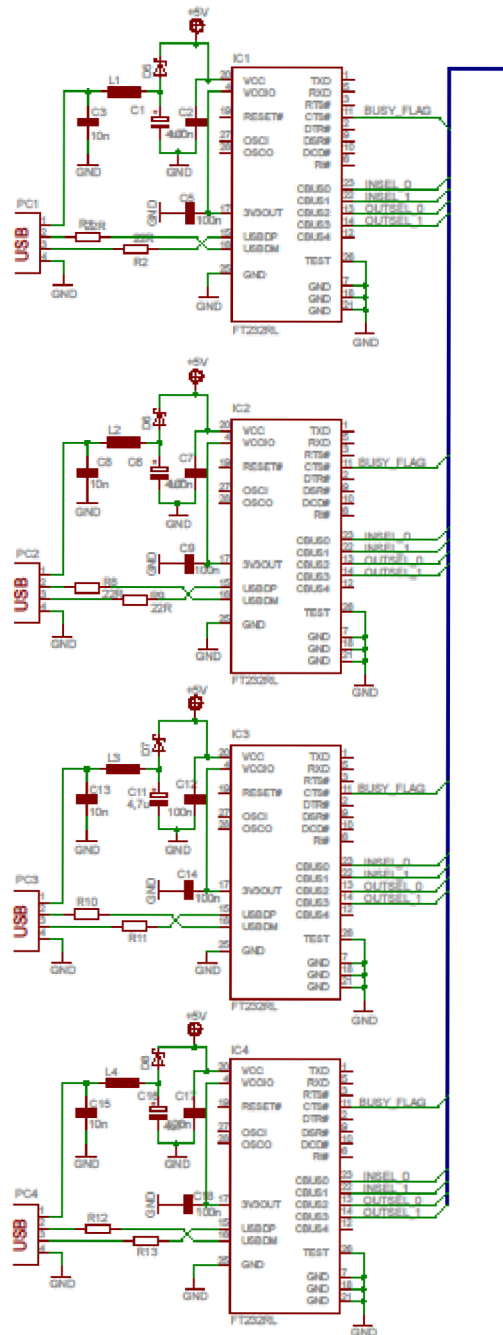
\doc – zložka obsahujúca pdf súbory dosky plošného spoja, schémy zapojenia a túto bakalársku prácu

Zoznam príloh

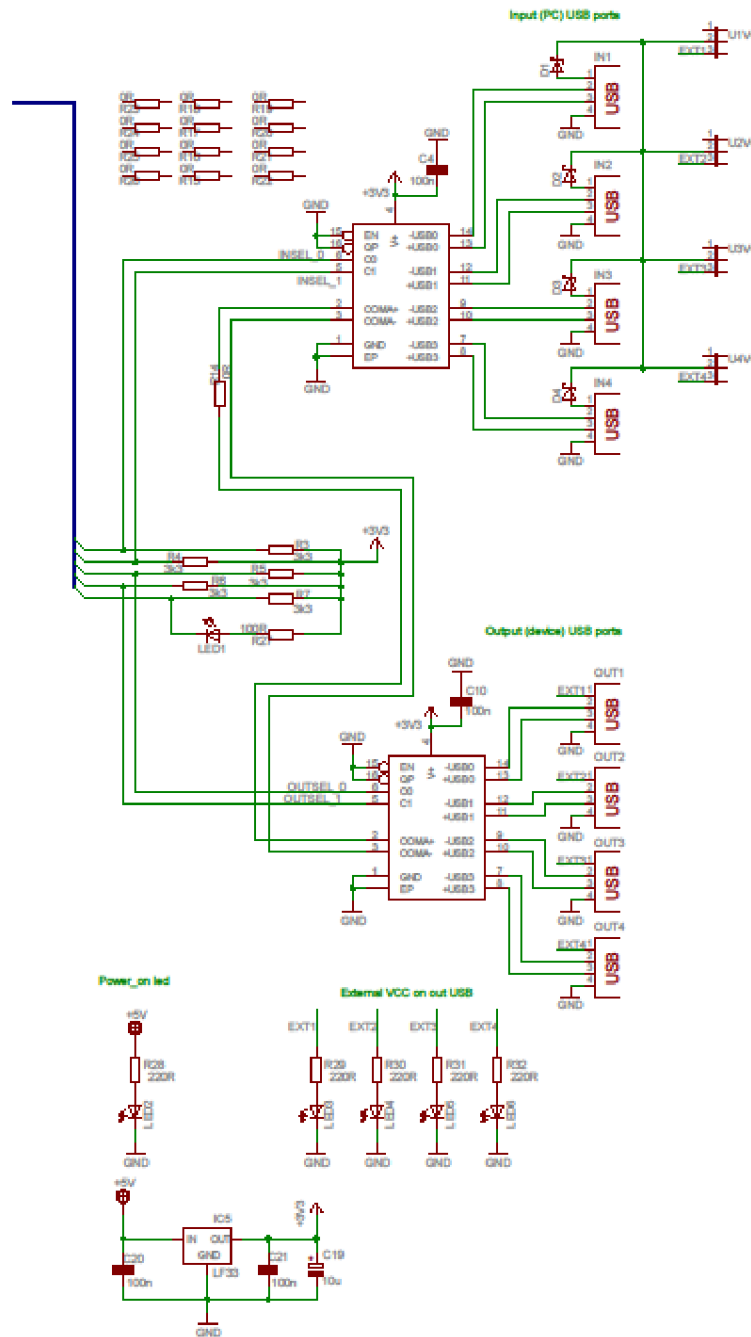
A Návrh prepínača – plošný spoj – strana spojov



B Návrh prepínača – schéma zapojenia – časť 1 : prevodníky



B Návrh prepínača – schéma zapojenia – časť 2 : multiplexery



C Zoznam súčiastok

Ozn.	Hodn.	Súčiastka	Púzdro	Ozn.	Hodn.	Súčiastka	Púzdro
C1	4u7	Keramický kond.	E2-5	R1	22R	Rezistor	0207/10
C2	100n	Elektrolitický kond.	C1206	R2	22R	Rezistor	0207/10
C3	10n	Elektrolitický kond.	C0805K C050-	R3	3k3	Rezistor	R1206
C4	100n	Elektrolitický kond.	024X044	R4	3k3	Rezistor	R1206
C5	100n	Elektrolitický kond.	C1206	R5	3k3	Rezistor	M1206
C6	4u7	Keramický kond.	E2-5	R6	3k3	Rezistor	R1206
C7	100n	Elektrolitický kond.	C1206	R7	3k3	Rezistor	M1206
C8	10n	Elektrolitický kond.	C0805K	R8	22R	Rezistor	0207/10
C9	100n	Elektrolitický kond.	C1206 C050-	R9	22R	Rezistor	0207/10
C10	100n	Elektrolitický kond.	024X044	R10	22R	Rezistor	0207/10
C11	4,7u	Keramický kond.	E2-5	R11	22R	Rezistor	0207/10
C12	100n	Elektrolitický kond.	C1206	R12	22R	Rezistor	0207/10
C13	10n	Elektrolitický kond.	C0805K	R13	22R	Rezistor	0207/10
C14	100n	Elektrolitický kond.	C1206	R14	0R	Rezistor	M1206
C15	10n	Elektrolitický kond.	C0805K	R15	0R	Rezistor	M1206
C16	4u7	Keramický kond.	E2-5	R16	0R	Rezistor	M1206
C17	100n	Elektrolitický kond.	C1206	R17	0R	Rezistor	M1206
C18	100n	Elektrolitický kond.	C1206	R18	0R	Rezistor	M1206
C19	10u	Keramický kond.	E2-5	R19	0R	Rezistor	M1206
C20	100n	Elektrolitický kond.	C1206	R20	0R	Rezistor	M1206
C21	100n	Elektrolitický kond.	C1206	R21	0R	Rezistor	M1206
D1		Schottkyho dióda	O201T15	R22	0R	Rezistor	M1206
D2		Schottkyho dióda	O201T15	R23	0R	Rezistor	M1206
D3		Schottkyho dióda	O201T15	R24	0R	Rezistor	M1206
D4		Schottkyho dióda	O201T15	R25	0R	Rezistor	M1206
D5		Schottkyho dióda	O201T15	R26	0R	Rezistor	M1206
D6		Schottkyho dióda	O201T15	R27	100R	Rezistor	R1206
D7		Schottkyho dióda	O201T15	R28	220R	Rezistor	R1206
D8		Schottkyho dióda	O201T15	R29	220R	Rezistor	R1206
IC1		prevodník FT232RL	SSOP28	R30	220R	Rezistor	R1206
IC2		prevodník FT232RL	SSOP28	R31	220R	Rezistor	R1206
IC3		prevodník FT232RL	SSOP28	R32	220R	Rezistor	R1206
IC4		prevodník FT232RL	SSOP28	U1V+		jumper	JP2
IC5		stabilizátor LF33	DPACK	U2V+		jumper	JP2
IN1		konektor USB typ A	PN61729	U3V+		jumper	JP2
IN2		konektor USB typ A	PN61729	U4V+		jumper	JP2
IN3		konektor USB typ A	PN61729	OUT1		USB typ B kon.	PN8752 0
IN4		konektor USB typ A	PN61729	OUT2		USB typ B kon.	PN8752 0
L1		indukčnosť	0204/7	OUT3		USB typ B kon.	PN8752 0

L2	indukčnost	0204/7	OUT4	USB typ B kon.	PN8752 0
L3	indukčnost	0204/7	PC1	USB typ B kon.	PN6172 9
L4	indukčnost	0204/7	PC2	USB typ B kon.	PN6172 9
LED 1	led dióda	LED3M M	PC3	USB typ B kon.	PN6172 9
LED 2	led dióda	LED3M M	PC4	USB typ B kon.	PN6172 9
LED 3	led dióda	LED3M M			
LED 4	led dióda	LED3M M			
LED 5	led dióda	LED3M M			
LED 6	led dióda	LED3M M			
MULIN	mux MAX4899AE	TQFN16			
MULOUT	mux MAX4899AE	TQFN16			