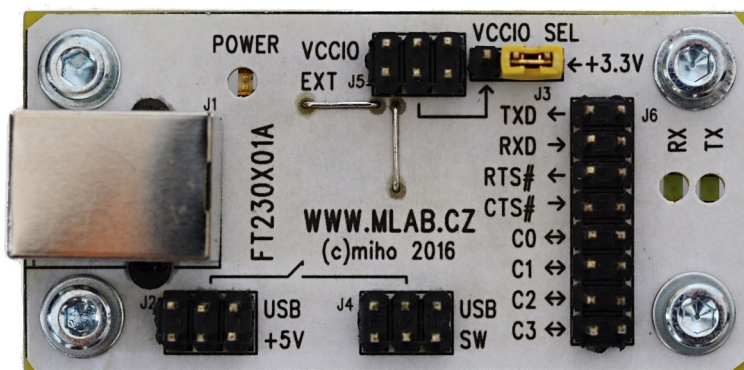


Prevodník USB na RS232

Milan Horkel

Modul slouží jako univerzální prevodník z USB na RS232 s výstupy na strane RS232 v úrovních TTL. Prevodník používá obvod FT230X od firmy FTDI. Tyto obvody jsou podporované ve všech rozšířených operačních systémech.



1. Technické parametry

Parametr	Hodnota	Poznámka
Rozhraní Konektor	USB 2.0 USB B	High Speed. Standardní velký USB konektor.
Spotřeba	8mA / 5V	Bez připojeného dalšího zařízení.
Signály sériové	RXD TXD RTS# CTS#	Základní signály sériového rozhraní s řízením toku. Volitelná polarita.
Signály pomocné	CBUS0 až CBUS3	Konfigurovatelné funkce, v továrním nastavení indikace RX a TX a spínání napájení pro externí využití.
Signálové úrovně	TTL/CMOS	VCCIO 1.62V až 3.63V, 5V tolerantní vstupy.
Řízení toku	Nastavitelné	Možno nastavit HW i SW řízení toku dat.
Režim spánku	Ano	Spínač napájení pro externí použití.
Rozměry	61x31x18mm	Výška nad základnou

2. Popis konstrukce

2.1. Úvodem

Standardní sériový port RS232 se stal vzácnou relikvií minulých času. V současné době většina zařízení (mikroprocesoru, obvodu SoC, vestavených počítačů a podobně) používá sériovou komunikaci stejně jako v minulosti, ale vyvedené signály obvykle pracují s TTL či CMOS úrovnemi napětí.

Použitý převodník USB pro sériovou komunikaci podporuje signály s napetovými úrovnemi pro CMOS/TTL logiku s úrovnemi 1.8V až 3.3V s tím, že vstupy jsou 5V tolerantní. Obvod generuje pomocné napětí 3.3V, které lze přímo použít pro napájení IO.

Důležité upozornění - podivné chování obvodu

Když se obvod FT230X připojí k USB, ale nemá přivedeno napájení IO vývodu (VCCIO), nedojde ke správné inicializaci IO vývodu a obvod pak nekomunikuje i když ze strany USB funguje normálně. Dodatečné připojení VCCIO nepomůže.

Když inicializace proběhne správně (byly obe napájení přítomné, lze obvod odpojit od USB nebo od VCCIO a po opětovném připojení chybejícího napětí funguje normálně.

Toto chování není podchyceno v Errata dokumentaci obvodu, ani v katalogovém listu. VCCIO není dostatečně nezávislé na hlavním napájení obvodu. Dle chování obvodu se lze domnívat, že se neprovede inicializace IO bunek obvodu.

Dalším překvapením je to, že výstup TXLED# se aktivuje při příjmu sériových dat a RXLED# při odesílání sériových dat (je to opakem, než je logické a než to mají starší obvody FTDI). Náprava je snadná, reprogramováním konfigurace v EEPROM obvodu.

2.2. Obvod FT230X

Jedná se o novější obvod, nástupce populárního obvodu FT232R. Je menší a podstatně levnější. Existuje několik členů v řadě X s různými rozhraními (sériové RS232 s RTS#/CTS#, sériové RS232 s plným počtem řídicích signálů, s rozhraním SPI, s rozhraním I2C slave).

Hlavní vlastnosti obvodu lze shrnout v těchto bodech:

- USB 2.0 s přenosem 300Bd až 3MBd s obvyklými nastaveními (7 nebo 8 bitů, parita, ...)
- FIFO pro příjem i vysílání, 2x512 bajtů
- Podpora řízení přenosu SW i HW (jen RTS#/CTS# u tohoto obvodu)
- Podpora ve všech běžných OS drivery od výrobce
- Programovatelné polarity signálů, pomocné signály
- Podpora úsporných režimů
- Nízká cena
- Minimum externích součástek, nepotřebuje krystal
- Konfigurační paměť EEPROM je integrovaná uvnitř obvodu

2.3. Zapojení modulu

Zapojení vychází z katalogu součástky a příslušných aplikacních poznámek.

Obvod FT230X má vestavený vnitřní stabilizátor 3.3V pro buzení USB signálu (jádro má další vnitřní stabilizátor, který ale není vyveden). Napětí 3.3V je vyvedeno na konektor J3 a lze jej využít pro napájení IO vývodu (VCCIO). Dále jej lze využít pro napájení dalších obvodů, zatížitelnost je až 50mA.

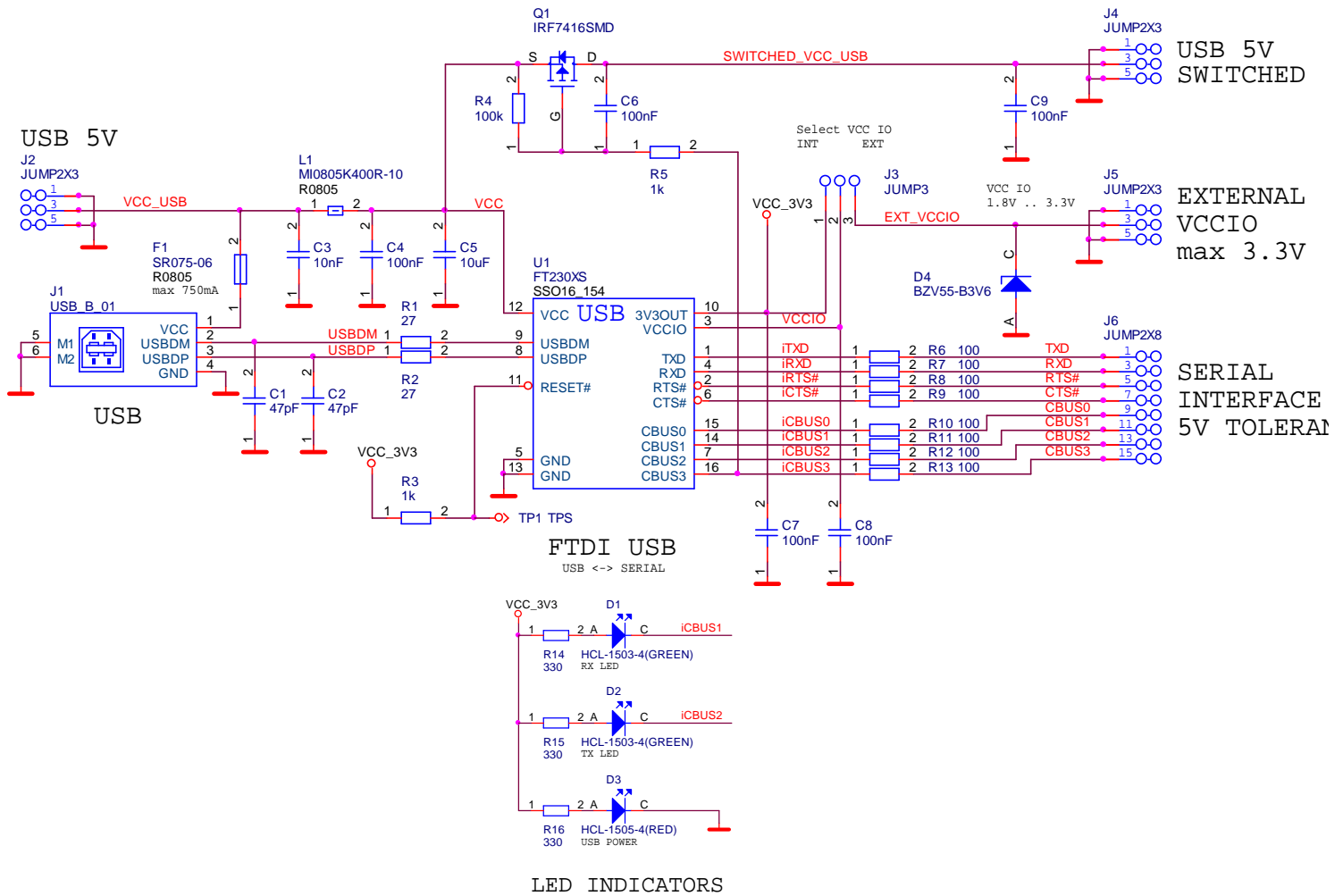
Propojkou na konektoru J3 lze zvolit VCCIO (napájení vstupně-výstupních budiců) z vnitřního stabilizátoru 3.3V, nebo z vnějšího zdroje z konektoru J5. Rozsah podporovaných napětí je 1.62V až 3.63V, vstupy jsou 5V tolerantní. Při napětí VCCIO cca 1V obvod odpojí budice.

Chování obvodu vzhledem k VCCIO je zmíněno v poznámce na začátku dokumentu, protože je to věc důležitá (zejména pro návrh nového hardwaru). Nespolehejte na nezávislost VCCIO a vnitřního napájení obvodu.

Tranzistor Q1 spíná napětí USB 5V na konektor J4. Toto napětí se automaticky vypne, když počítač přejde do režimu spánku. Pro tuto funkci se musí nakonfigurovat v paměti EEPROM obvodu na vývod C3 (CBUS3) funkce PWREN#.

LED diody indikují RX a TX aktivitu. Funkci vývodu je možno změnit modifikací obsahu EEPROM. Na vývod C1 (CBUS1) je defaultně nastavena funkce RXLED# a na vývod C2 (CBUS2) je defaultně nastavena funkce TXLED#.

Kupodivu funkce RXLED# bliká při vysílání dat na sériovou linku a funkce TXLED# bliká při příjmu dat ze sériové linky. Je to nelogické a opacné než u obvodu FT232R, ale dá se to snadno napravit naprogramováním EEPROM součástky. Viz konfigurace obvodu v sekci konfigurace.



2.4. Mechanická konstrukce

Prevodník je navržen v podobe standardního modulu stavebnice MLAB. V rozích je opatřen upevňovacími šrouby se sloupky.



Reference	Název	Pouzdro
<i>Odpory</i>		
R1, R2	27	R0805
R6, R7, R8, R9, R10, R11, R12, R13	100	R0805
R14, R15, R16	330	R0805
R3, R5	1k	R0805
R4	100k	R0805
<i>Keramické kondenzátory</i>		
C1, C2	47pF	C0805
C3	10nF	C0805
C4, C6, C7, C8, C9	100nF	C0805
C5	10uF	C0805
<i>Indukčnosti</i>		
L1	MI0805K400R-10	R0805
<i>Diody</i>		
D1, D2	HCL-1503-4 (GREEN)	LED1206REV
D3	HCL-1505-4 (RED)	LED1206REV
D4	BZV55-B3V6	MINIMELF

Reference	Název	Pouzdro
<i>Tranzistory</i>		
Q1	IRF7416SMD	SO8_FET
<i>Integrované obvody</i>		
U1	FT230XS	SSO16_154
<i>Pojistka</i>		
F1	SR075-06	R0805
<i>Mechanické součástky</i>		
J1	USB_B_01	USB_B_01
J3	JUMP3	JUMP3
J2, J4, J5	JUMP2X3	JUMP2X3
J6	JUMP2X8	JUMP2X8
<i>Konstrucní součásti</i>		
1ks	JUMPER	Propojka
4ks	Šroub M3x12	Šroub pozinkovaný
4ks	Podložka M3	Podložka pozinkovaná
4ks	Sloupek M3x5	Sloupek
1ks	PCB	Plošný spoj

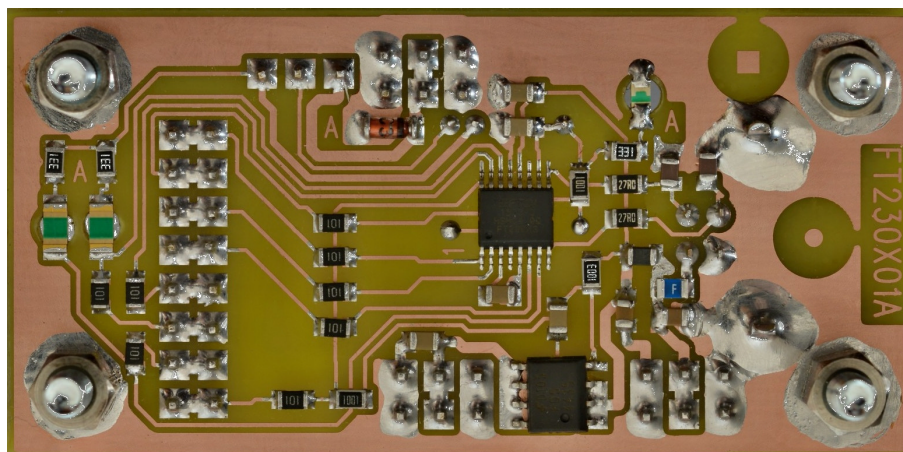
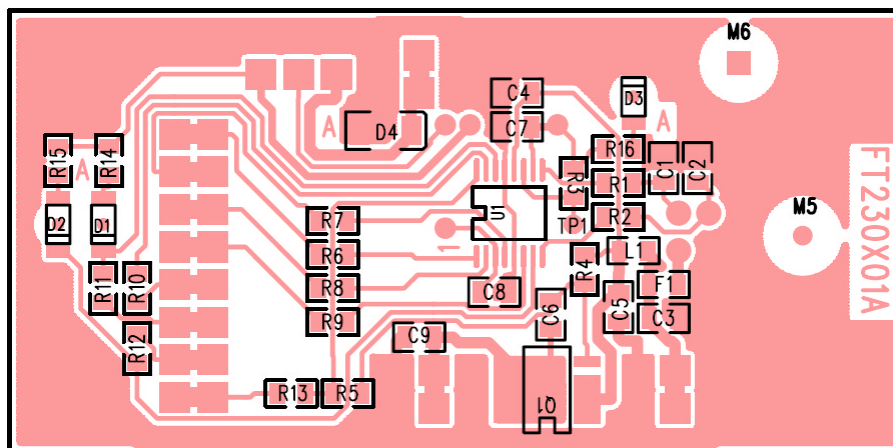
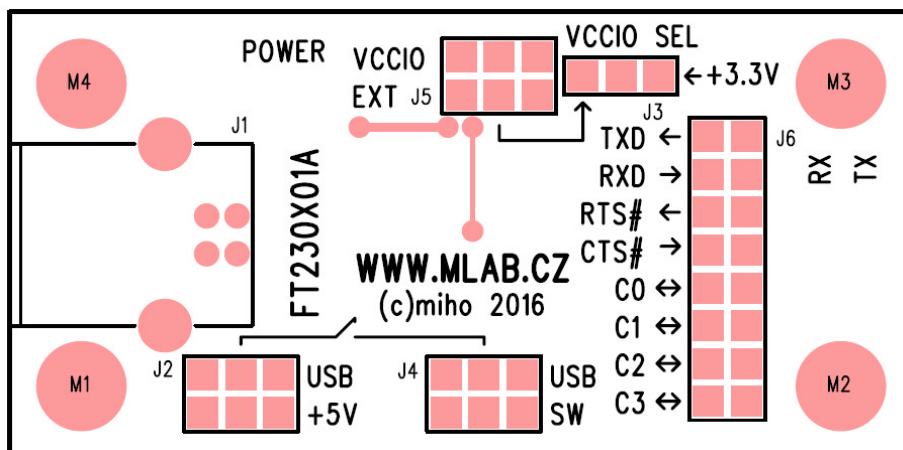
3. Osazení a oživení

3.1. Osazení

Nejprve osazujeme SMD součástky. Zčneme obvodem U1 aby nám neprekázely ostatní součástky. Je třeba použít pastového tavidla a minimum pájky. Mikropájčka je nezbytností.

Na strane součástí nezapomente osadit 2 drátové propojky. Polarita LED diod je označena (A jako anoda v motivu plošného spoje).

Diody LED jsou osazovány reverzně (koukají do díry v plošném spoji)!



3.2. Oživení

Pečlivě zkontrolujeme připojení součástek (prerušení a zkratky). Připojíme zdroj +5V na konektor J2 (spotřeba bude cca 5mA) a zkontrolujeme napětí +3.3V na konektoru J3 na vývodu 1. Indikační dioda D3 by měla svítit. Spojíme J3.2 a J3.1 (interní napájení VCCIO připojíme na interní napětí +3.3V).

Pripojíme na USB a počkáme, až se spustí ovladace (nové systémy je už mají připravené automaticky, starší budou chtít instalační soubory). Spustíme terminálový program (například PuTTY) a vyzkoušíme funkčnost. Když propojíme RX a TX propojkou budou blikat obe indikační LED a odesílané znaky se budou na terminálu opakovat.

3.3. Konfigurace

3.3.1. Provoz bez konfigurace

Obvod funguje i bez jakékoli konfigurace. Díky chybě v obvodu budou prohozené ledky indikující RX a TX aktivitu. Dále nebude správně fungovat spínání napájení tranzistorem Q1 při uspání počítače.

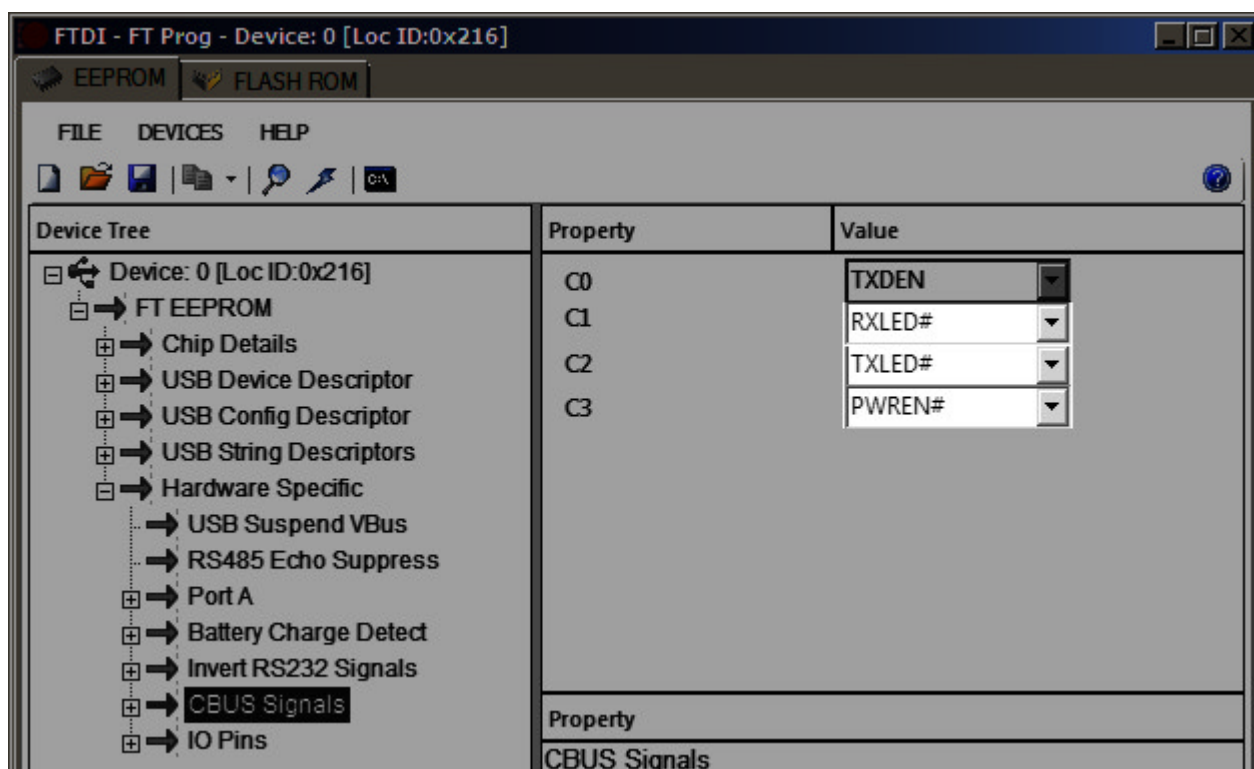
3.3.2. Konfigurace

Konfigurace obvodu se zapisuje do interní paměti EEPROM. K nastavování konfigurace je k dispozici program FT Prog, který je určen pro Windows. Nalezneme jej na webu FTDI tady:

http://www.ftdichip.com/Support/Utilities.htm#FT_PROG

Pro správnou funkci spínacího tranzistoru Q1 je třeba vývodu C3 (CBUS3) přiřadit funkci PWREN#. Uvedené nastavení funguje tak, že v případě uspání počítače (s aktivním USB portem) vypne tranzistor Q1 výstupní napětí 5V (z USB) na konektoru J4 a tím vypne spotřebu cílové aplikace.

Dále je třeba prohodit funkce TXLED# a RXLED# na vývodech C1 a C2 aby funkce odpovídala potisku. Modul má popisky správně, jen obvod FTDI funguje nelogicky opakne.



Dále lze nastavit jméno a identifikaci zařízení, případně jeho sériové číslo. Při změně položek VendorID/DeviceID je nezbytné změnit tyto parametry i v INF souborech driveru (aby se správně přiřadily) a případně drivery nechat znovu podepsat (novější systémy Windows).

Dále se konfiguruje provozní parametry součástky (režim napájení a spotřeba, polarita signálu, síla a rychlost budicu).

Konfiguraci je možné uložit do souboru a ten pak opakovaně použít. Vzor je uložen v adresáři EEPROM.

4. Programové vybavení – drivery

Drivery, programy a originální dokumentace na stránkách výrobce cipu <http://www.ftdichip.com>. Aplikacní poznámky si zaslouží alespoň zbežné prostudování.

4.1. Instalace driveru

Při prvním připojení převodníku k USB portu si systém vyžádá adresář s drivery, u novějších systémů si je najde sám. Základní chování je takové, že se pro převodník spustí drivery pro obsluhu zařízení se dvěma aplikacními rozhraními.

Zařízení je podporováno všemi novými operačními systémy (Windows, Linux, Android, tablety, mobily, RaspberryPI a podobně). Pro programátory jsou připravené knihovny pro vlastní použití (jak pro Windows, tak i pro Linux, statické i dynamické) s přehlednou dokumentací.

4.1.1. Rozhraní Virtual COM Port

Jedná se o režim emulovaného sériového portu. Zpřístupnění tohoto rozhraní je možné zakázat v konfiguraci součástky (v paměti EEPROM) nebo v konfiguraci driveru (v příslušném INI souboru).

Ke komunikaci je možné použít vhodný terminálový program, oblíbená volba je program PuTTY. Spojení se při odpojení USB kabelu bohužel přeruší.

V systému Windows se po připojení k počítači zařízení najde a spustí se pro něj příslušné drivery a v systému se zjeví nový COM port (jeho číslo se dozvíme ve správci zařízení v sekci porty).

Aplikace mohou se zařízením komunikovat prostřednictvím rozhraní Win32 COM API jako se standardním sériovým portem.

V systému Linux se po připojení v systému zjeví nové zařízení /dev/ttyUSB0 (nebo tak nějak dle konkrétní distribuce) a funguje jako nový sériový port. Jméno zařízení lze snadno zjistit příkazem dmesg.

4.1.2. Rozhraní D2XX API

Toto rozhraní je proprietární rozhraní firmy FTDI a slouží pro komunikaci s obvody FTDI včetně ovládání jejich speciálních funkcí (programování paměti EEPROM, přímé ovládání vstupu a výstupu). K přístupu k tomuto rozhraní je k dispozici knihovna (jako dynamická, tak i statická) opět pro všechny systémy.

4.2. Základy USB (pro verzi USB 2.0)

Podrobnosti na <http://www.usb.org>. Zde je k dispozici specifikace USB.

4.2.1. Úplné základy

Rychlost na USB se často plete:

- Full Speed – 480Mbit/s – pouze USB 2.0, používají například USB disky, nutné lepší kabely
- High Speed – 12MBit/s – běžná zařízení (*i náš převodník*)
- Low Speed – 1.5Mbit/s – zařízení typu HID (klávesnice, myši)

Napájení:

- Napájecí napětí 5V, k dispozici zaručene minimálně 100mA, maximálně 500mA pokud to zdroj v pocitaci nebo HUBu umožňuje
- Zařízení po zastrčení do USB nesmí mít spotřebu větší než 100mA
- Signály jsou 3.3V, diferenciální pár
- Teoreticky má být napájení jištěno *ale u některých zařízení není*

Prenos dat:

- Isochronní – zabraná šířka pásma – například pro prenos zvuku
- Interrupt prenosy – pro interaktivní prenosy – například klávesnice, signalizace a podobne
- Bulk prenosy – prenosy velkého množství dat – disky a podobne (*i náš převodník*)
- Řídící prenosy – pro řízení USB zařízení

Komunikace po USB probíhá v rámci, které se přenáší vždy po 1 milisekundu (u zařízení Low Speed nebo High Speed). Do každého rámce se snaží obslužné programy (drivery USB) nacpat co nejvíce prenosu. Přednost mají isochronní prenosy (například pro reproduktory), pak data související s interaktivním prenosem (ocuchávání klávesnice, prenos prerušení) a teprve zbytek kapacity je vyplněn hromadnými prenosy. Převodník využívá hromadné (Bulk) prenosy.

V žádném případě nelze spoléhat na časování prenosu. Data se přenáší po paketech a na strane driveru i v obvodu FTDI jsou vyrovnávací paměti pro odesílaná a pro přijímaná data. Jediná správná cesta bezchybného prenosu spočívá ve využití řízení prenosu buď HW signály (RTS#/CTS#), nebo znaky XON/XOFF (SW řízení prenosu).

4.2.2. Prenos dat u obvodu FTDI

Data se přenáší po paketech o délce až 64B. 2 bajty se použijí pro prenos stavu řídicích signálů a zbytek jsou sériová data. Prenos dat probíhá pouze pokud se zaplní vyrovnávací paměť o délce 64B nebo pokud od posledního prenosu ubehla dostatečně dlouhá doba (vyprší Latency Timer s přednastavenou dobou 16ms).

Pokud se změní stav některého z řídicích signálů dojde k prenosu v nejbližším milisekundovém rámci. Totéž platí i pro případ prenosu některého znaku pro řízení prenosu (obvykle znaky XON a XOFF).