

USB komunikacija između FPGA i personalnog računara

Vladimir Kovačević, Žarko Perić

Sadržaj - Uredaji nove generacije sve češće zahtevaju komunikaciju sa personalnim računarom. Potreba za interfejsom jednostavnim za implementaciju i korišćenje je sve učestalija. Rešenje bazirano na USB protokolu postaje logičan izbor za konekciju između personalnog računara i FPGA čipa. U radu je dat prikaz jednog takvog sistema.

Ključne reči — USB, interfejs, FTDI, FPGA

I. UVOD

Poslednjih nekoliko godina USB je sve više zastupljen kao periferni interfejs. FPGA (*Field Programmable Gate Array*) preuzima primat u implementaciji funkcija tih periferija (generisanje i obrada signala, *embedded softcore* sistemi, itd). USB je po svojoj strukturi veoma kompleksan. Umanjenje vremena razvoja uređaja je moguće dobiti upotrebom jednostavnog interfejsa. Korišćenjem FTDI čipa, implementiratiran je sistem pomenutih, na prvi pogled kontradiktornih, zahteva.

II. ISTORIJSKI RAZVOJ I OSOBINE USB INTERFEJSA

Povećanjem spektra upotrebe računarskih periferija javila se potreba za njihovom standardizacijom, pojednostavljenjem priključivanja i softvera, kao i povećanjem brzine transfera. U tu svrhu, Novembra 1995. godine nekoliko vodećih kompanija sa zajedničkom interesnom sferom (Intel, Microsoft, Philips i US Robotics) sačinilo je specifikacioni model USB (*Universal Serial Bus*) 1.0 serijske magistrale.

Pojavom prvih komercijalnih računara sa USB priključkom (Maj 1998) standardizovana je i verzija 1.1, koja podržava dva tipa protoka: *Low speed* (1.5 Mbit/s) i *Full speed* (12 Mbit/s). USB 2.0 [1] (April 2000) uvodi podršku veće brzine protoka, *High speed* (480 Mbit/s). Sve verzije protokola su kompatibilne unazad, što znači da periferije dizajnirane u skladu sa USB 1.0 specifikacijom normalno mogu da funkcionišu i sa današnjim računarima i uređajima. U pripremi je USB 3.0 specifikacija (90% je završena 13. Avgusta 2008. godine) koja je namenjena da podrži protok do 4.8 Gbit/s. Zbog potpunije slike, u nastavku su navedene neke od najvažnijih osobina USB interfejsa.

USB sistem je asimetričnog dizajna, sastoji se od host-a (sa kontrolerom), jednog ili više portova (*downstream*

tokova) i periferijalnih uređaja povezanih u topologiju stabla (*tiered-star*). Jedan fizički USB uređaj može se sastojati od više logičkih poduređaja (*sub-device*), koji se odnose na funkcionalnost, na primer, kamera (video funkcija) i mikrofon (audio funkcija). Tu osobinu omogućava struktura zasnovana na krajnjim tačkama na uređaju (*endpoint*). Uređaj može imati maksimalno 16 izlaznih (ka kontroleru) i 16 ulaznih *endpoint*-a. Svaki od njih se može konfigurisati za različite tipove komunikacije, osim nultog koji host koristi samo za inicijalnu konfiguraciju uređaja pri povezivanju na magistralu (proces enumeracije). Enumeracija (proces nabranja, prozivke magistrale hubova koju vrši host da bi detektovao priključenje ili odvajanje periferija) počinje slanjem reset signala svim USB uređajima, zatim se utvrđuju maksimalne podržane brzine, čitanje podešavanja sa samih uređaja i dodela jedinstvene 7-bitne adrese. Ako host podržava uređaj vrši se učitavanje odgovarajućih drajvera u operativnu memoriju. USB drajver na hostu upravlja interakcijom između USB periferije i korisničkog softvera. Sami transferi na magistrali se obavljaju pod kompletном kontrolom host-a, koji vrši poliranje svih priključenih uređaja po round robin principu. USB uređaj ne može započeti transfer podataka bez eksplicitne dozvole host kontrolera. Fizički interfejs se sastoji od 4 žice: Napajanje, masa i diferencijalni par za podatke, D+ i D-. Maksimalna struja kroz linije je 100 mA pri inicijalizaciji, odnosno 500 mA pri normalnom radu, sa naponom od 5V. To je još jedna od zanimljivih osobina vezanih za ovaj interfejs, koja omogućava da se periferni uređaji koji su manji potrošači (do 2W) napajaju preko USB magistrale.

III. ARHITEKTURA SISTEMA

Za komunikaciju (kontrolu i čitanje statusa) između PC računara i FPGA (u kome je implementiran periferni uređaj) često je potrebno uspostaviti jednostavan i pouzdan interfejs, sa malim vremenom razvoja, kome brzina transfera nije prioritet. Upravo takvo rešenje pruža FT245R [2] firme FTDI Chip. Implementacija USB protokola je enkapsulirana u sam čip, što pruža projektantu mogućnost implementacije samog interfejsa bez detaljnog znanja funkcionisanja USB transfera. Zahvaljujući paralelnom, asinhronom interfejsu i baferima za prijem i slanje, FPGA vidi USB kao FIFO strukturu preko koje dobija samo korisne podatke sa host-a. Sa strane računara, javno dostupan je FTDI-USB drajver, koji postavlja interfejs serijskog (COM) porta

Vladimir Kovačević, Elektrotehnički fakultet u Beogradu, Srbija (telefon: 381-63-7114791; e-mail: vladakova@gmail.com).

Žarko Perić, Elektrotehnički fakultet u Beogradu, Srbija; (telefon: 381-638524150; e-mail: periczarko@gmail.com).

ispred USB toka podataka. Na taj način je korisnička aplikacija, koja šalje podatke i obrađuje primljene sa USB interfejsa, veoma jednostavna za implementaciju.



Sl. 1: Arhitektura sistema sa FTDI USB interfejsom

Samo rešenje USB interfejsa je projektovano da bude opšte i nezavisno od tipa FPGA čipa. U konkretnoj realizaciji korišćen je Xilinx, Virtex5 [3][4], XC5VLX30T, zbog ugrađenih RocketIO gigabitnih transivera i DCM (Digital Clock Management) komponenti, koji su korišteni pri implementaciji funkcija uređaja. Iako se nalazi u sredini komunikacije u sistemu, detaljniji opis arhitekture (Sl. 1) je najlogičnije započeti opisom funkcionisanja samog FTDI čipa, zbog toga što se njemu prilagođava programabilni FPGA.

IV. OPIS FTDI ČIPA

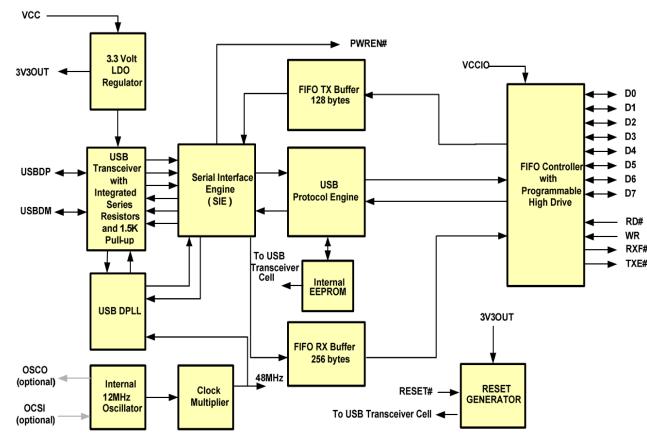
FT245R predstavlja konvertor USB protokola u paralelni FIFO interfejs. U njegovoj specifikaciji podržane su sledeće funkcionalnosti:

- Bidirekcioni USB transfer preko paralelnog FIFO interfejsa
- Enkapsulaciju USB protokola i njegovo korišćenje bez prethodnog programiranja *firmware-a*
- Integriran 1024-bitni EEPROM u kome su smešteni deskriptori uređaja i FIFO konfiguracija
- Integriran generator takta
- Maksimalan transfer podataka – 1Mbit/s
- 256 bajtova u prijemnom i 128 bajtova u predajnom baferu, čime su omogućene više bitske brzine
- Paralelni 8-bitni asinhroni interfejs, sa signalizacijom popunjenošći bafera
- Postojanje besplatnog drajvera sa interfejsom virtuelnog COM porta
- Kompatibilnost sa USB 2.0 i podržanost svih tipova host kontrolera

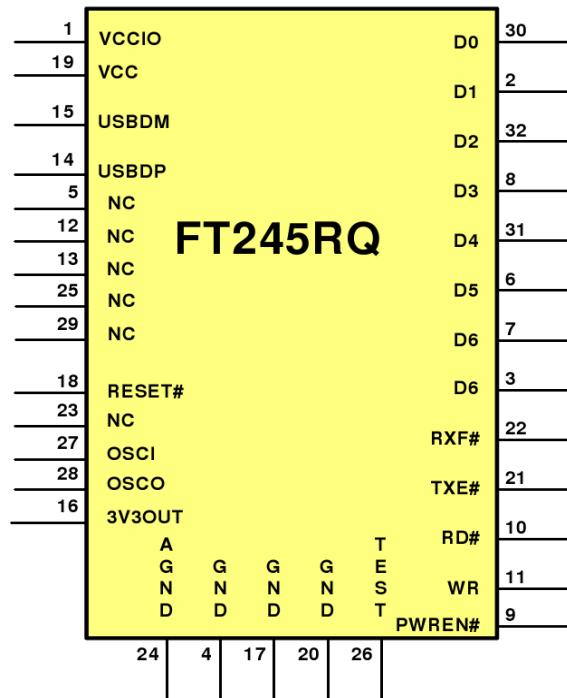
Na Sl. 2 prikazane su najvažnije komponente samog čipa. LDO regulator generiše 3.3 V napon za transivere i ostatak čipa. USB transiveri implementiraju fizički sloj USB protokola, do samog kabla. DLL se koristi za lokovanje takta sa dolazećih USB NRZI (*No Return to Zero*) podataka i generisanje obnovljenog takta serijskoj jedinici SIE (Serial Interface Engine). 12 MHz oscilator generiše takt, koji se množi 4 puta i koristi u FIFO baferima, u SIE i u samom protokolu (*USB Protocol Engine*). Serijska jedinica (SIE) vrši serijalizaciju i deserijalizaciju podataka, kao i računanje i proveru CRC5/CRC19 na tokovima podataka. FIFO RX bafer (128 bajtova) čuva podatke koje je poslao USB kontroler preko OUT *endpoint-a*. Podaci se uklanjaju iz njega čitanjem, koristeći RD# pin (detaljan opis dat je u narednom poglavljju). FIFO TX bafer (256 bajtova) čuva

podatke koji su primljeni paralelno (koristeći WR# pin) i trebaju biti učitani od strane host kontrolera preko IN *endpoint-a*. Na desnoj strani Sl. 2 nalazi se programabilni FIFO kontroler, koji obavlja paralelni prijem i slanje podataka preko D0-D7 linija. Generator reseta (*Reset Generator Cell*) omogućava pouzdan start samog uređaja pri dovođenju napona napajanja. RESET# pin omogućava da spoljni uređaj (FPGA) resetuje čip.

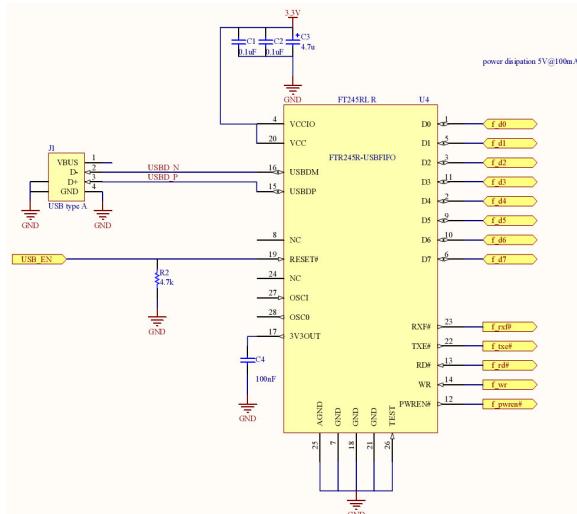
32 pina, koliko uređaj ima, mogu se podeliti na FPGA pinove (sa desne strane, Sl. 3) i ostale, koji služe za napajanje i USB diferencijalnu liniju za podatke (USBDM i USBDP). Detaljan opis pinova postoji u dokumentaciji samog čipa [2]. Istaknućemo indikator priključenja na magistralu – PWREN# i negacije pojave punog, odnosno praznog bafera – RXF# i TXE#. Na Sl. 4 data je električna šema FT245R u sistemu sa Virtex5 čipom.



Sl. 2: Blok dijagram FT245R čipa



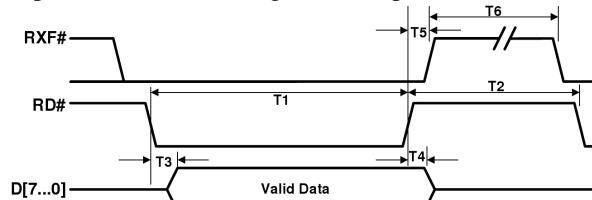
Sl. 3: FT245R - logička i fizička struktura pinova



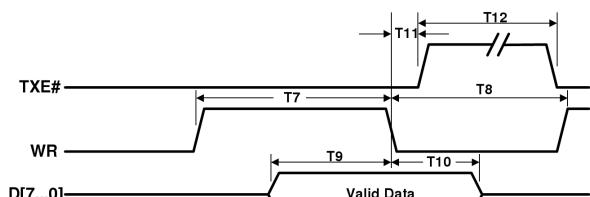
Sl. 4: Električna šema FT245R pri vezi sa FPGA

V. FPGA CORE

U samom FPGA potrebno je obezbediti modul koji će komunicirati sa FTDI čipom. Vremenski dijagrami po kojima se obavlja čitanje (Sl. 5), upis (Sl. 6) podataka, i trajanje intervala T, dati su u dokumentaciji [2], moraju se poštovati radi korektnog transfera podataka.



Sl. 5: Vremenski dijagram čitanja podataka iz USB FIFO



Sl. 6: Vremenski dijagram upisa podataka u USB FIFO

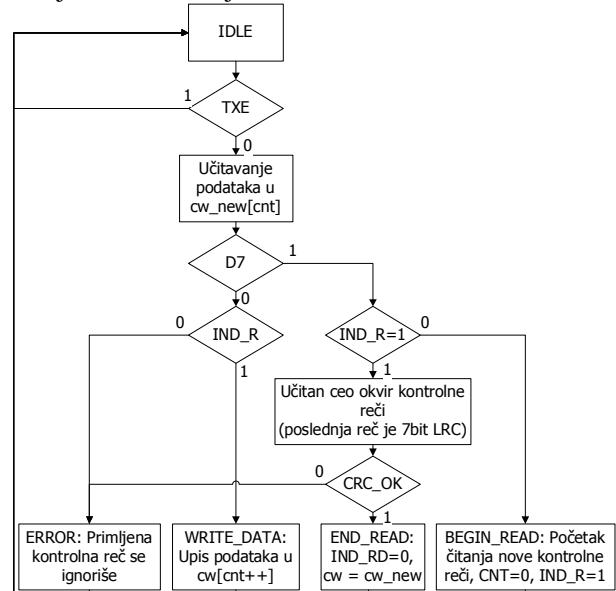
FPGA core pisan je u VHDL-u, sinteza i implementacija vršena je u Xilinx ISE 10.0 razvojnog alata, a simulacija u ModelSim 6.2. Njegova osnovna uloga je da primljenu kontrolnu reč (od PC računara, preko USB FIFO) prosledi ostatku sistema, koji će promeniti ponašanje u zavisnosti od njene vrednosti. Ostavljena je mogućnost definisanja kontrolne reči koja se sastoji od više bajtova - kontrolni okvir (frame). Zbog toga je potrebno uspostaviti protokol primanja kontrolne reči i definisati njenu strukturu:

- Prvi bajt sadrži dužinu kontrolnog okvira
- Bit najveće težine svakog bajta se koristi za signalizaciju, i to, 1 se nalazi u prvom i poslednjem bajtu, a 0 na svim ostalim
- Poslednji bajt sadrži 7-bitni LRC (*Longitudinal redundancy check*) koji se računa XOR-ovanjem svih ostalih bajtova

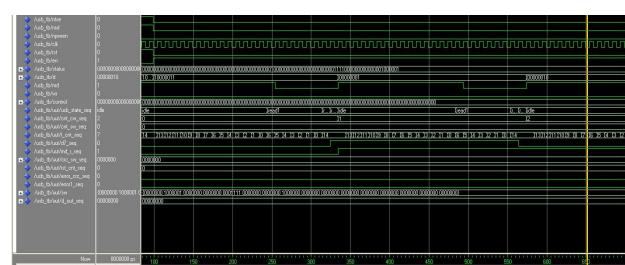
Algoritam čitanja kontrolnog frejma dat je na Sl. 7. Mašina stanja za čitanje ima 3 stanja (READ1, READ2 i

READ3). Njeni prelazi se vrše po stizanju novog bajta (RXF# postaje aktivran), testira se bit najveće težine (D7) i postavi indikator IND_R, koji je aktivran u toku primanja kontrolne reči. Da bi se obezbedilo trajanje vremena T, kao što je definisano na Sl. 4 konstruisan je brojač (t_cnt) koji broji tačno definisani broj otkucaja takta, čime je trajanje D i RD# signala tačno po specifikaciji. Za brojanje pristiglih bajtova kontrolnog okvira koristi se brojač cw_cnt, čija vrednost mora biti jednaka vrednosti prvog bajta kontrolnog okvira, po stizanju poslednjeg bajta. Na Sl. 8 i Sl. 9 dati su vremenski dijagrami čitanja kontrolne reči dobijen simulacijom, na kojima se vidi rezultat funkcionisanja opisanog algoritma.

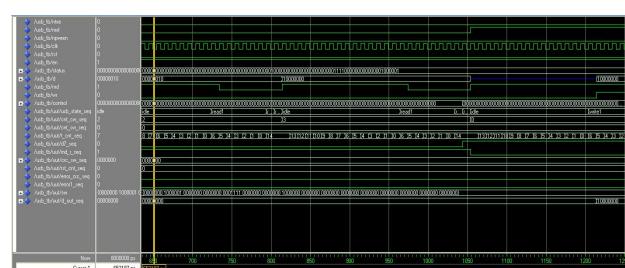
FPGA core, vrši i slanje statusne reči (dobija je od ostatka dizajna) USB FIFO interfejsu, koji je proslećuje hostu. Algoritam slanja je krajnje jednostavan, i sastoji se od generisanja sekvence signala sa Sl. 5 onoliko puta koliko ima bajtova u statusnoj reči



Sl. 7: Algoritam čitanja okvira kontrolne reči



Sl. 8: Simulacija čitanja bajta kontrolne reči



Sl. 9: Završetak čitanja okvira kontrolne reči

Pored navedenih funkcionalnosti *core* vrši ulogu kontrolne jedinice celog dizajna time što kontroliše PLL (*Phase Lock Loop*) i tako obezbeđuje takt ostatku sistema. Generisanje reseta je moguće preko kontrolne reči, posle čega će reset signal biti prosledjen ostatku čipa. Za ovako isprojektovan modul je potpuno transparentno koja je funkcija samog uređaja, što mu omogućava prenosivost i ponovnu upotrebu na sistemima slične strukture (jedina promena odnosila bi se na definisanje drukčije dužine kontrolne i statusne reči, kao i značenja njihovih bitova).

VI. KORISNIČKA APLIKACIJA I DRAJVER

Za uspostavljanje USB komunikacije sa uređajem neophodno je izvršiti učitavanje odgovarajućeg drajvera u operativnu memoriju. Za FT245R postoji javno dostupan drajver – *Virtual COM port Driver* [5] koji postavlja interfejs COM [6] (serijskog) porta ispred USB toka podataka.

Grafički interfejs preko koga bi korisnik komunicirao sa uređajem je moguće implementirati u nekoliko programskih jezika (Java, C++, C#), od kojih je C# najpogodniji zbog same podržanosti funkcija za rad sa COM portom i jednostavne konstrukcije korisničkih kontrola.

VII. ZAKLJUČAK

Pored navedene funkcionalnosti, veoma bitna osobina opisanog sistema je niska cena komponenti, besplatni razvojni alati (u osnovnoj verziji) i drajver. Nezavisnost od platforme FPGA uređaja pruža mu mogućnost za veoma široku primenu i ponovnu upotrebu. Dalji razvoj sistema podrazumevao bi generalizaciju funkcionisanja

FPGA dizajna, njegovo detaljno dokumentovanje i konstruisanje intelektualne svojine.

LITERATURA

- [1] Specifikacija USB interfejsa, zvanični sajt, <http://www.usb.org/developers/docs>
- [2] FT245R USB FIFO IC Datasheet Version 2.02
- [3] Virtex-5 User-Guide Lite, Peter Alfke, Xilinx
- [4] Virtex-5 FPGA User Guide, UG190 (v4.2) May 9, 2008, Xilinx
- [5] Drajver za FT245R, Virtuelni COM port <http://www.ftdichip.com/Drivers/VCP.htm>
- [6] Virtuelni COM port, COM port redirector, http://en.wikipedia.org/wiki/Virtual_serial_port

ABSTRACT

New generation devices often require communication with personal computer. Need for a simple interface and easy implementation becomes demanding. USB based solution is logical choice for connection between PC and FPGA. This paper shows one of the possible implementations.

USB COMMUNICATION BETWEEN FPGA AND PC

Vladimir Kovačević, Žarko Perić