

Application Note



DA Core - Funkční vzorek řadiče D/A převodníku se sběrnicí SPI

Jiří Kadlec, Leoš Kafka, Jiří Svozil

kadlec@utia.cas.cz

Obsah

1. Úvod.....	2
2. Popis systému.....	2
3. Parametry modulu DA_CORE	5
4. Použité/Potřebné vybavení a nástroje	7
5. Funkční vzorek D/A převodníku.....	7
6. Poděkování	10
7. DA_CORE demo.....	11
8. Licensing and availability (anglicky)	11
9. Disclaimer (anglicky).....	11
10. Reference.....	12

Revize

Revize	Datum	Autor	Popis změn v dokumentu
0	20.4.2008	Jiří Svozil	Vytvoření dokumentu
1	15.8.2008	Jiří Svozil	Revize dokumentu
2	12.12:2009	Jiří Kadlec	Verze pro ISE 11.3 s popisem příkladu
3	31.12.2010	Jiří Kadlec	Verze pro ISE 12.4 s popisem příkladu
4	02.08.2011	Jiří Kadlec	Verze pro ISE 13.2 s popisem příkladu

1. Úvod

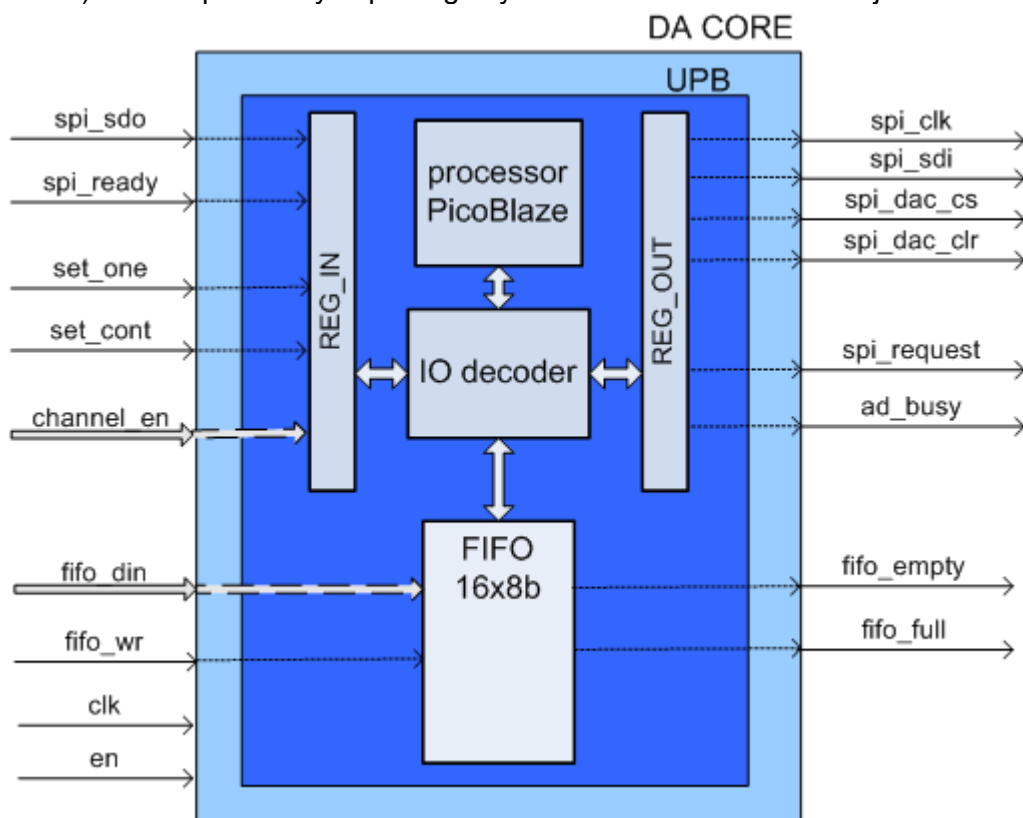
Tento dokument popisuje funkční vzorek HW modulu řadiče D/A převodníků se sběrnicí SPI. Jádro řadiče je založeno na UPB (Universal PicoBlaze Wrapper). UPB je konfigurovatelný wrapper procesoru PicoBlaze [1].

Modul řadiče je vytvořen tak, aby uživateli umožnil práci s převodníky na vyšší úrovni, tzn. i bez rozsáhlých znalostí použitého hardwaru. Uživatel může tento modul zařazovat do složitějších systémů, kde tento modul pouze připojí a nemusí se zabývat samotným procesem řízení D/A převodníku. Řadič je nakonfigurován pro čtyřkanálový D/A převodník LTC2624 [4], které je součástí přípravku Xilinx S3E1600 [2]. Veškeré řízení a obsluha této komponenty je prováděna pomocí SPI sběrnice.

Testovací příklad vychází z příkladu [6d], jehož autorem je Ken Chapman.

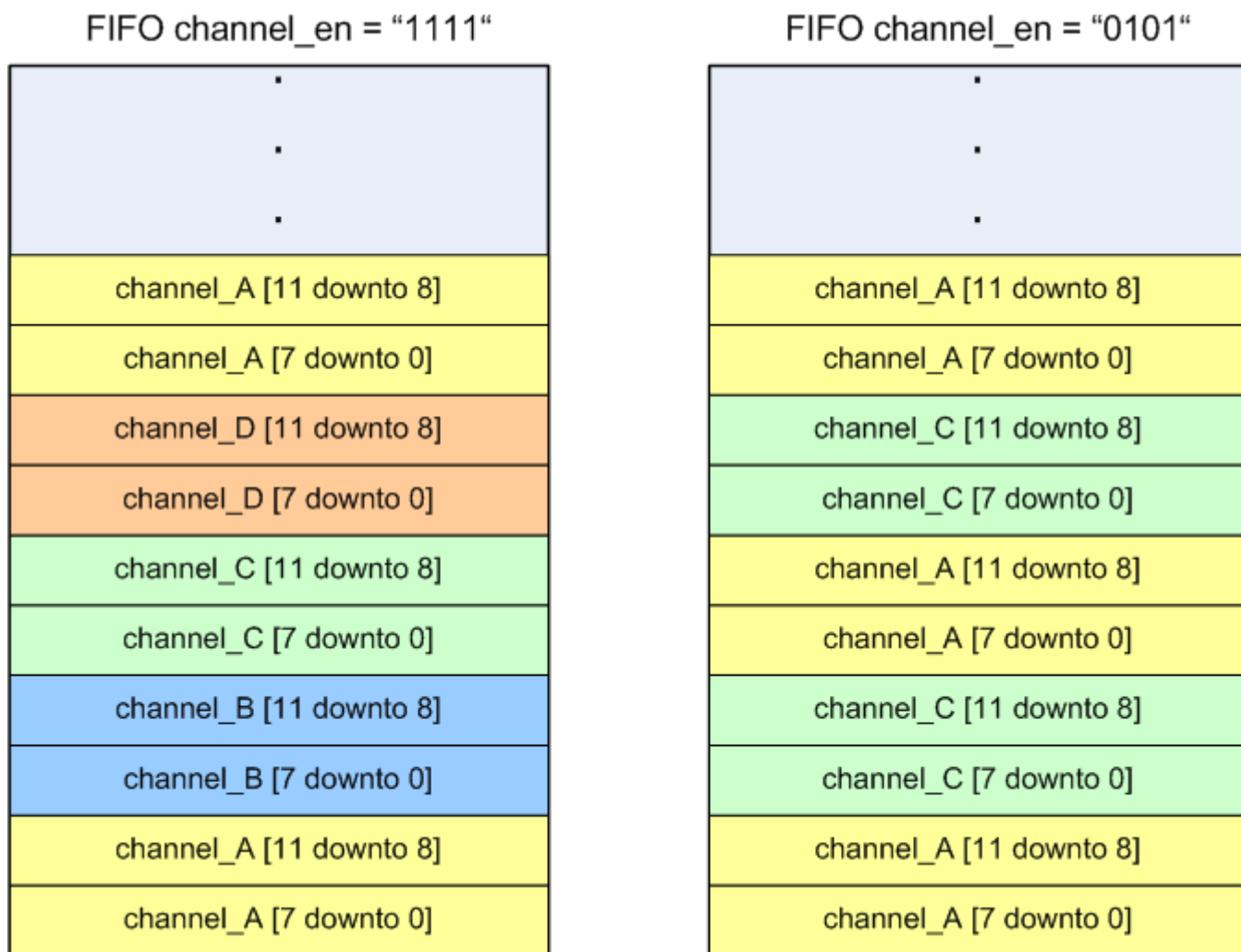
2. Popis systému

Univerzální řadič D/A převodníku je založen na konfigurovatelném wrapperu UPB, jehož autorem je Ing. Leoš Kafka. Ve funkčním vzorku DA CORE wrapper obsahuje vstupní FIFO o volitelné délce (standardně 16B) a 3 vstupní a 3 výstupní registry. Blokové schéma znázorňuje obr. 1.



Obrázek 1 Blokové schéma modulu D/A převodníku

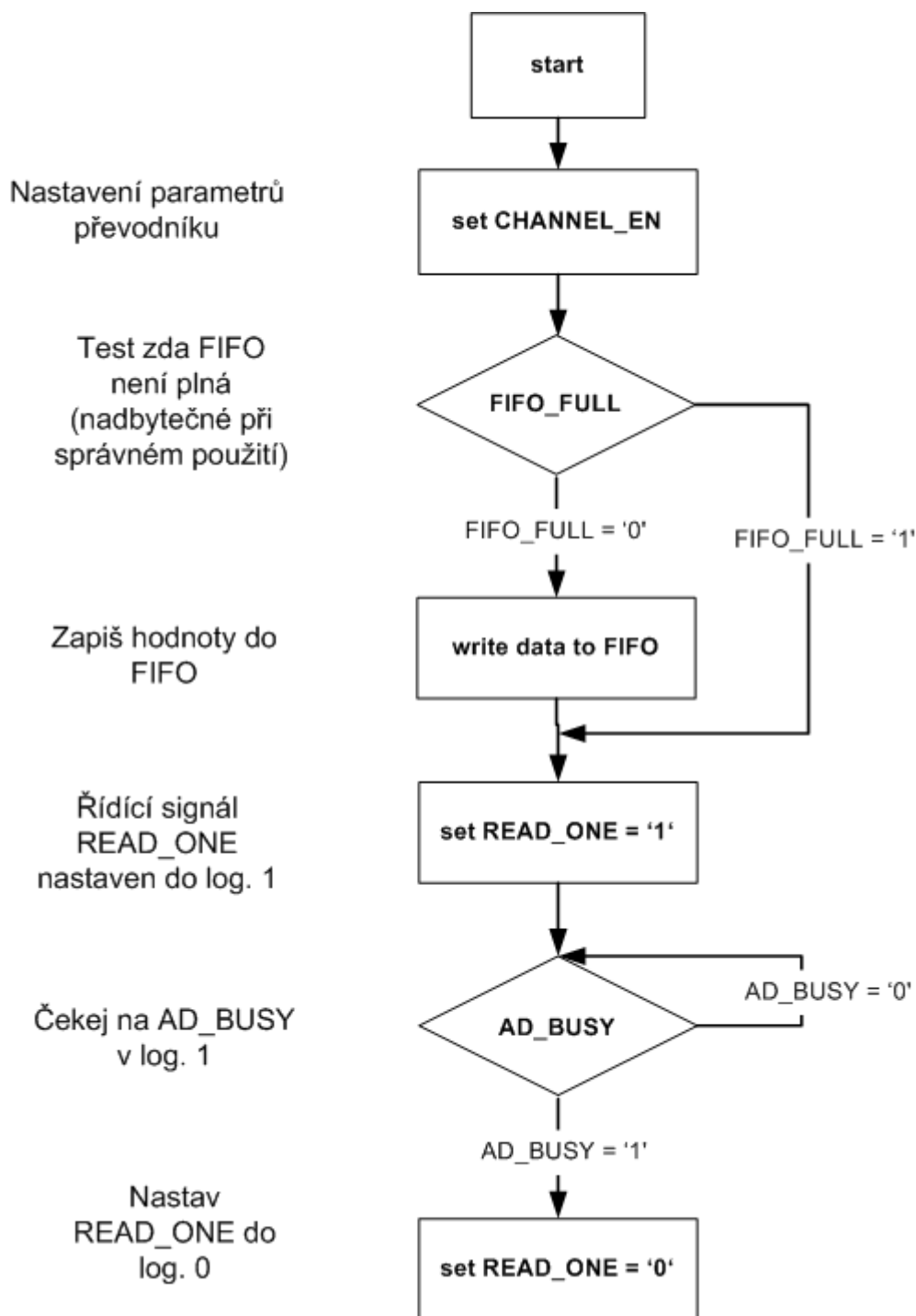
Do paměti FIFO jsou ukládána data, která slouží jako vstupní hodnoty D/A převodníků. Protože převodníky jsou 12bitové, data musí být do FIFO ukládána po bytech. Nejprve je do paměti uloženo 8 nižších bitů a poté 4 vyšší. Data musí být do paměti takto ukládána vždy pouze pro aktivní kanály (*channel_en*). Jsou-li aktivní všechny kanály, data jsou ukládána v pořadí *Channel A*, *Channel B*, *Channel C* a *Channel D*. Tento systém znázorňuje obr. 2.



Obrázek 2 Systém ukládání dat do FIFO vlevo - aktivní všechny kanály, vpravo – aktivní kanály A,C

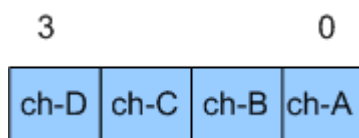
Zápis hodnot do převodníků přes SPI sběrnici je řízen signály *set_one* a *set_cont*, přičemž aktivní může být vždy pouze jeden. Signál *set_one* slouží k zapsání jednoho vzorku do D/A převodníku pro každý aktivní kanál. Signál *set_cont* pak k opakovanému zápisu v intervalech zadaných frekvencí SAMPLE_FREQ (frekvence zadána v Hz).

Způsob použití signálu *set_one* znázorňuje Obrázek 3. Signál musí zůstat aktivní, dokud není nastaven výstupní signál modulu *da_busy*. Ten je tedy v tomto případě použit jako potvrzení přijetí požadavku.



Obrázek 3 Příklad použití řídicího signálu SET_ONE

K určení aktivních převodníků slouží řídicí signál *channel_en*. Jeho velikost je volitelná (standardně 4bitová). Kanál je aktivován nastavením log. 1 na příslušné pozici signálu. Pozice převodníků odpovídající signálu *channel_en* jsou následující: nultý bit odpovídá kanálu A, první bit kanálu B, druhý kanálu C a třetí bit kanálu D (viz Obrázek 4).



Obrázek 4 Uspořádání signálu CHANNEL_EN

Signály *spi_request* a *spi_ready* slouží k řízení sběrnice SPI v případě připojení více zařízení na tuto sběrnici. Signál *spi_request* vysílá požadavek na uvolnění sběrnice SPI a teprve po potvrzení pomocí *spi_ready* je zahájena datová komunikace.

Příznak *ad_busy* signalizuje zápis hodnot do převodníku. Během doby, kdy je tento signál v log. 1, nesmějí být měněny vstupy nastavující parametry modulu, a to především signál *channel_en*.

Dalšími vstupními porty jsou hodinový vstup *clk* a vstup pro aktivování modulu *en*. Seznam portů i s popisem zobrazuje Tabulka 1.

Tabulka 1 - Přehled portů modulu DA Core

Port	Vstup/Výstup	Šířka [bit]	Význam
clk	Vstup	1	Hodinový signál
en	Vstup	1	Povolení funkce modulu (aktivní v log.1)
spi_sdo	Vstup	1	Vstup SPI dat
spi_ready	Vstup	1	Potvrzení požadavku <i>spi_request</i>
set_one	Vstup	1	Nastavení aktivních D/A převodníků
set_cont	Vstup	1	Kontinuální nastavování D/A
channel_en	Vstup	4	Povolení kanálu
fifo_din	Vstup	8	Vstupní data FIFO
fifo_wr	Vstup	1	Zápis dat do FIFO
spi_clk	Výstup	1	Hodinový signál SPI
spi_sdi	Výstup	1	Datový výstup SPI
spi_dac_cs	Výstup	1	Chip-select (aktivní v log. 0)
spi_dac_clr	Výstup	1	Reset SPI (aktivní v log. 0)
spi_request	Výstup	1	Žádost o uvolnění SPI sběrnice
ad_busy	Výstup	1	Při log. 1 nelze měnit vstupní parametry
fifo_empty	Výstup	1	Příznak prázdné FIFO
fifo_full	Výstup	1	Příznak plné FIFO

3. Parametry modulu DA_CORE

Hodiny systému 75 MHz jsou generovány v FPGA z hodinového signálu desky 50 MHz pomocí modulu *dcm_mhz*. Maximální vzorkovací frekvenci úměrnou počtu použitých převodníků uvádí tabulka 2.

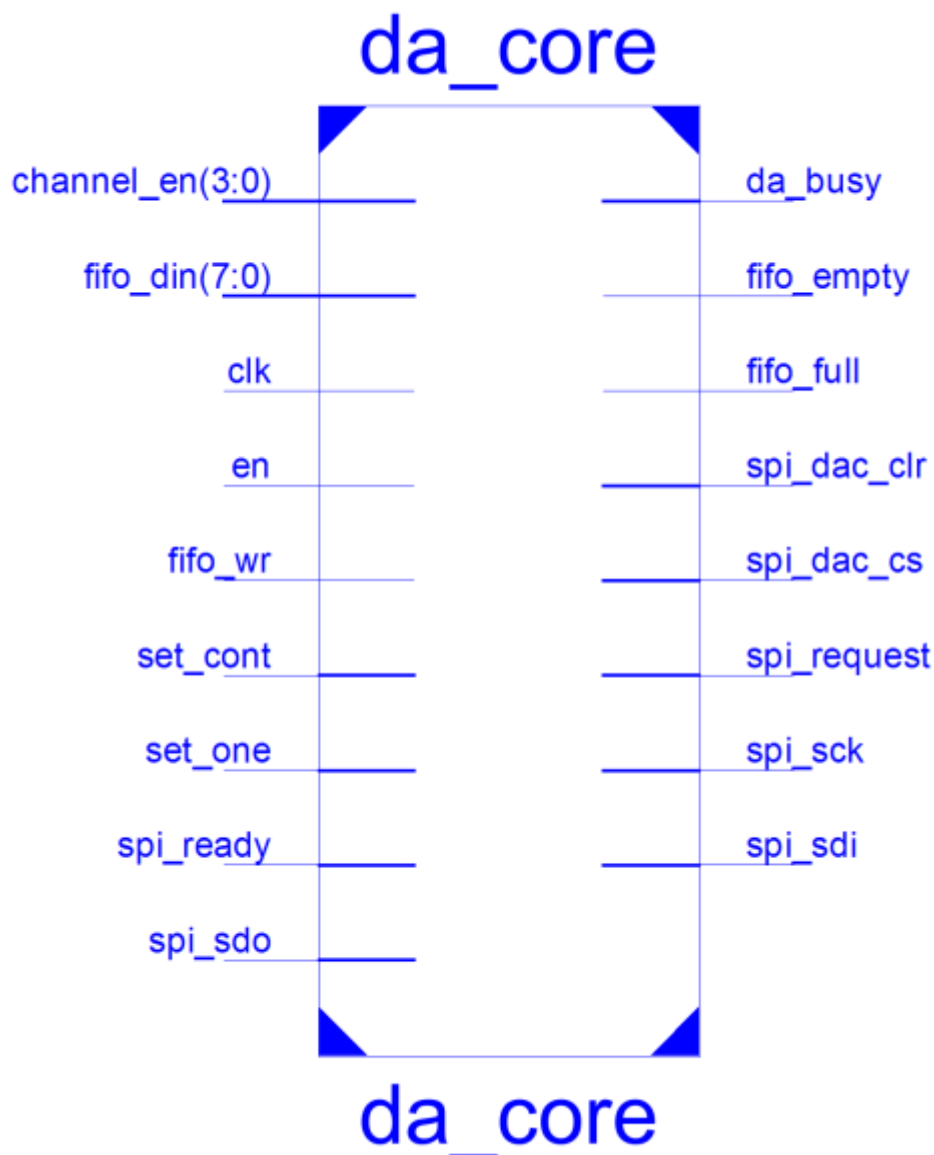
Tabulka 2 - Maximální vzorkovací frekvence modulu DA_CORE

Počet kanálů	max. frekvence
4 kanály	15 kHz
3 kanály	30 kHz
2 kanály	45 kHz
1 kanál	60 kHz

Velikost modulu DA_CORE v ISE 13.2 uvádí tabulka 3.

Tabulka 3 - HW nároky modulu DA_CORE

Number of Slices	149
Number of Slice Flip Flops	144
Number of 4 input LUTs	281
Number of BRAMs	1



Obrázek 5 `da_core` v ISE 13.2

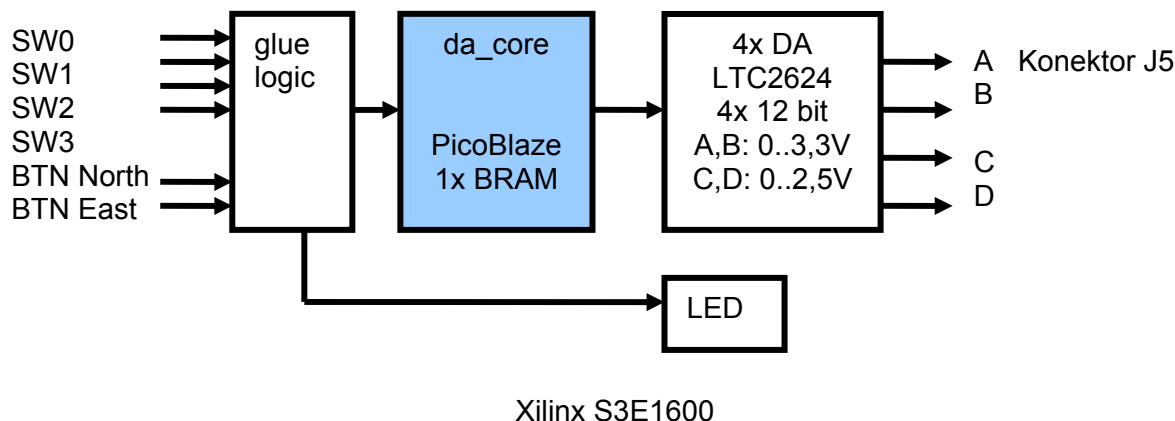
4. Použité/Potřebné vybavení a nástroje

Modul D/A převodníku byl napsán v programovacím jazyce VHDL ve vývojovém prostředí Xilinx ISE verze 13.2 [3]. Program procesoru PicoBlaze byl napsán v assembleru a k překladu byl použit assembler *KCPSM3.exe*, který je součástí balíčku procesoru PicoBlaze [1]. Testovací příklad byl vytvořen na vývojovém kitu Xilinx S3E1600 [2], viz příložené CD.

5. Funkční vzorek D/A převodníku.

Funkční vzorek HW modulu řadiče dvoukanalového D/A převodníku a programovatelného zesilovače je realizován pro kartu Xilinx S3E1600. Funkční vzorek demonstruje funkčnost modulu a

umožňuje nastavení libovolného převodníku na (programem definovanou) hodnotu. Ovládání je prováděno pomocí tlačítek a přepínačů na desce Xilinx S3E1600. Blokové schéma funkčního vzorku zachycuje Obr. 6.



Obrázek 6: Blokové schéma funkčního vzorku s AD_CORE

Výběr (povolení) převodníku se provádí pomocí přepínačů SW0 až SW3 (toto odpovídá signálu channel_en).

Zápis hodnot:

READ_ONE: Zapiše hodnotu na všechny aktuálně povolené převodníky. Ovládání: BTN North

READ_CONT: s genericky nastavenou frekvencí (10 kHz) zapisuje hodnoty na aktuálně povolené převodníky. Ovládání: BTN East

signál BUSY (da_core) je přiveden na LED - LD0 (pozorovatelný rozdíl svítivosti při aktivním 1 kanálu nebo 4 kanálech, dále pak při kontinuálním či jednotlivém čtení)

Hodiny systému 75 MHz jsou generovány v FPGA z hodinového signálu desky 50 MHz pomocí modulu dcm_mhz. Velikost implementovaného demonstrátoru funkčního vzorku v ISE 13.2 uvádí tabulka 4.

Tabulka 4 – Celkové HW nároky příkladu použití modulu DA_CORE

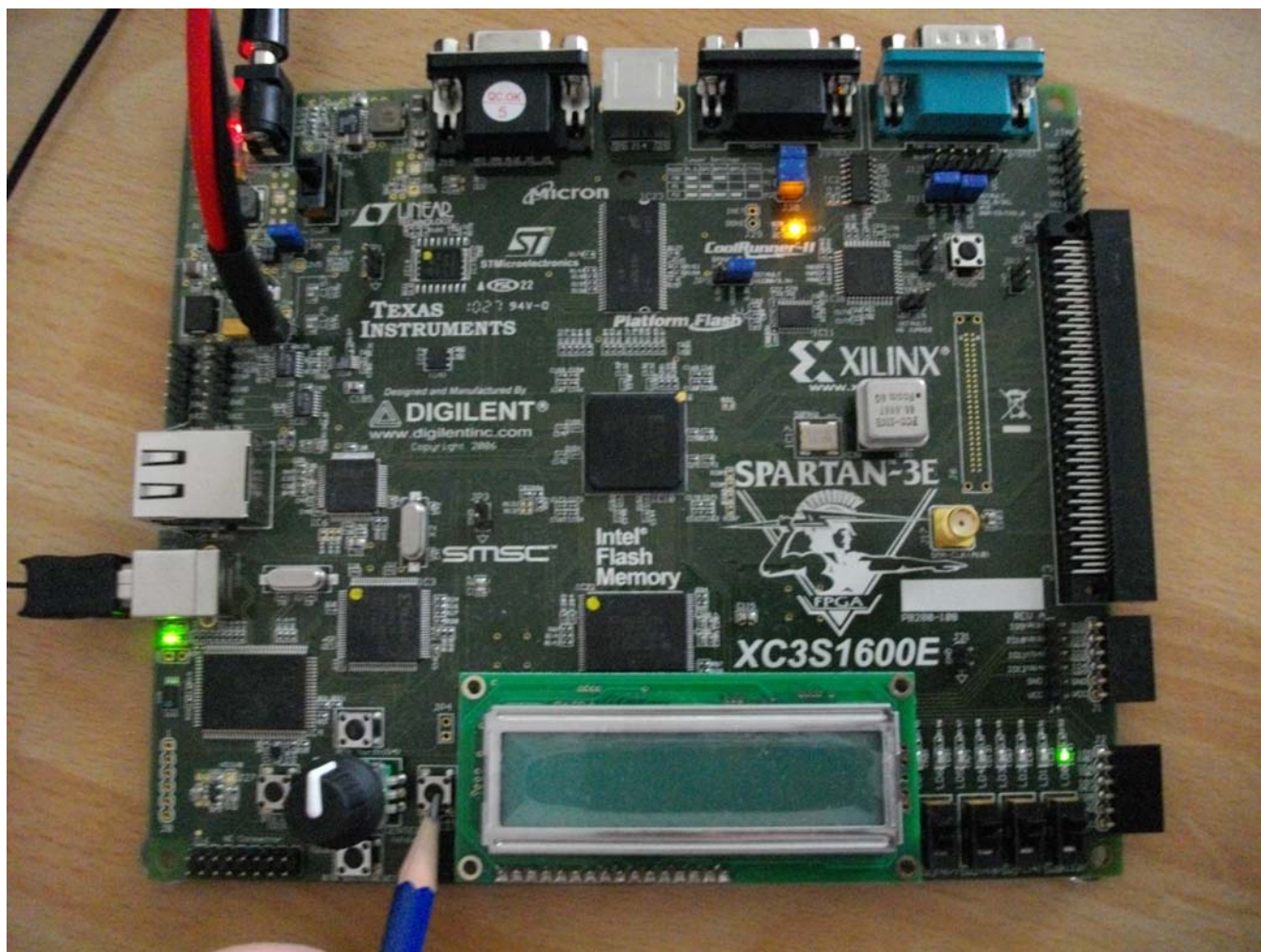
	XC3S1600E-4
	top_3s1600e
Number of Slices	162
Number of Slice Flip Flop	143
Number of 4 input LUTs	246
Number of BRAMs	1
VCCINT 1,2V (FPGA)	40mA
Odběr na 5,0V zdroji	500mA

Vstupy a výstupy funkčního vzorku modulu řadiče čtyřkanálového D/A převodníku na desce S3E1600 uvádí tabulka 5.

Tabulka 5 – Vstupy a výstupy funkčního vzorku DA_CORE na desce S3E1600

	<p>NET "clk" PERIOD = 20.0ns HIGH 50%; NET clk LOC = C9 IOSTANDARD = LVCMOS33;</p> <p>NET spi_sck LOC = U16 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 8; NET spi_sdi LOC = T4 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 8; NET spi_sdo LOC = N10 IOSTANDARD = LVCMOS33; NET spi_rom_cs LOC = U3 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 8; NET spi_amp_cs LOC = N7 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 8; NET spi_adc_conv LOC = P11 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 8; NET spi_dac_cs LOC = N8 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 8; NET spi_amp_shdn LOC = P7 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 8; NET spi_dac_clr LOC = P8 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 8;</p> <p>NET strataflash_oe LOC = C18 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 2; NET strataflash_ce LOC = D16 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 2; NET strataflash_we LOC = D17 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 2; NET platformflash_oe LOC = T3 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 2;</p> <p>NET switch_hw<0> LOC = L13 IOSTANDARD = LVCMOS33 PULLUP; NET switch_hw<1> LOC = L14 IOSTANDARD = LVCMOS33 PULLUP; NET switch_hw<2> LOC = H18 IOSTANDARD = LVCMOS33 PULLUP; NET switch_hw<3> LOC = N17 IOSTANDARD = LVCMOS33 PULLUP;</p> <p>NET button_hw<0> LOC = V4 IOSTANDARD = LVCMOS33 PULLDOWN; NET button_hw<1> LOC = H13 IOSTANDARD = LVCMOS33 PULLDOWN;</p> <p>NET led LOC = D4 IOSTANDARD = LVCMOS33 SLEW = SLOW DRIVE = 8;</p>
--	---

Funkční vzorek modulu řadiče čtyřkanálového D/A převodníku na desce S3E1600 je yobrayen na obr. 5



Obrázek 7: Funkční vzorek modulu řadiče čtyřkanalového DA převodníku a programovatelného zesilovače na desce S3E1600

6. Poděkování

Návrh a realizace tohoto funkčního vzorku byla podpořena projektem ministerstva školství a tělovýchovy číslo 2C06008, <http://www.vlam.cz>.

7. DA_CORE demo

S3E1600\ impl\ da_core\ da_core.ngc	Modul převodníku
dcm_mhz\ dcm_mhz.ngc	Modul generující hodinový signál 75 MHz
top_3s1600e\ ...	Projekt pro překlad funkčního vzorku pomocí Xilinx ISE 13.2
top_3s1600e_bit\ top_3s1600e_mcf\ src\ ucf\ top_3s1600e.ucf	Konfigurační data funkčního vzorku Konfigurační data pro platform flash Definice vstupů a výstupů pro 3s1600e
vhdl\ top_3s1600e.vhd	Zdrojový kód ve VHDL (top level)

8. Licensing and availability (anglicky)

This functional sample is provided in form of netlist compiled by Xilinx ISE 13.2 [3] for the Xilinx xc3s1600e-4fg320 [2]. If you plan to use this functional sample netlist for education purposes, you can contact UTIA AV CR, v.v.i. for support. The contact person in UTIA is Jiri Kadlec kadlec@utia.cas.cz tel. +420 2 6605 2216 fax. +420 2 6605 2511.

If you consider commercial use of this functional sample in form of netlist compiled by Xilinx ISE 13.2 [3] for the Xilinx xc3s1600e-4fg320 FPGAs or if you need the source code of this functional sample, please contact UTIA AV CR v.v.i. Commercial End User License Agreement (Commercial EULA) between you and UTIA AV CR, v.v.i. needs to be signed. The contact person in UTIA is Jiri Kadlec kadlec@utia.cas.cz tel. +420 2 6605 2216 fax. +420 2 6605 2511.

9. Disclaimer (anglicky)

This disclaimer is not a license and does not grant any rights to the materials distributed herewith. Except as otherwise provided in a valid license issued to you by UTIA AV CR v.v.i., and to the maximum extent permitted by applicable law:

- (1) THIS APPLICATION NOTE AND RELATED MATERIALS LISTED IN THE PACKAGE CONTENT ARE MADE AVAILABLE "AS IS" AND WITH ALL FAULTS, AND UTIA AV CR V.V.I. HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and
- (2) UTIA AV CR v.v.i. shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under or in connection with these materials, including for any direct, or any indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or UTIA AV CR v.v.i. had been advised of the possibility of the same.

UTIA AV CR v.v.i. products are not designed or intended to be fail-safe, or for use in any application requiring fail-safe performance, such as life-support or safety devices or systems, Class III medical devices, nuclear facilities, applications related to the deployment of airbags, or any other applications that could lead to death, personal injury, or severe property or environmental damage (individually and collectively, "Critical Applications"). Customer assumes the sole risk and liability of any use of UTIA AV CR v.v.i. products in Critical Applications, subject only to applicable laws and regulations governing limitations on product liability.

10. Reference

- [1] PicoBlaze 8-bit Embedded Processor UG129 June 22, 2011
http://www.xilinx.com/support/documentation/ip_documentation/ug129.pdf
- [2] MicroBlaze Development Kit Spartan-3E 1600E Edition User Guide UG257 (v1.1) December 5, 2007
http://www.xilinx.com/support/documentation/boards_and_kits/ug257.pdf
- [3] Xilinx ISE 13.2, <http://www.xilinx.com/>
- [4]] řadič LTC2624
<http://www.linear.com/pc/downloadDocument.do?navId=H0,C1,C1155,C1005,C1156,P2048,D2170>
- [5] Spartan3 datasheet http://www.xilinx.com/support/documentation/data_sheets/ds099.pdf
- [6] Xilinx Spartan-3E FPGA Starter Kit Board Design Examples
http://www.xilinx.com/products/boards/s3estarter/reference_designs.htm
- [6a] Ken Chapman, Initial Design for Spartan-3E Starter Kit (LCD Display Control), ISE 8.1i, Xilinx Ltd 16th, February 2006
http://www.xilinx.com/products/boards/s3estarter/files/s3esk_startup.pdf
- [6b] Ken Chapman, Rotary Encoder Interface for Spartan-3E Starter Kit, ISE 8.1i, Xilinx Ltd, 20th February 2006
http://www.xilinx.com/products/boards/s3estarter/files/s3esk_rotary_encoder_interface.pdf
- [6c] Ken Chapman, Amplifier and A/D Converter Control for Spartan-3E Starter Kit, ISE 8.1i, Xilinx Ltd, 23rd February 2006
www.xilinx.com/products/boards/s3estarter/files/s3esk_picoblaze_amplifier_and_adc_control.pdf
- [6d] Ken Chapman, D/A Converter Control for Spartan-3E Starter Kit, ISE 8.1i, Xilinx Ltd, 21st February 2006
http://www.xilinx.com/products/boards/s3estarter/files/s3esk_picoblaze_dac_control.pdf
- [6e] Ken Chapman, NOR FLASH Programmer for Spartan-3E Starter Kit, ISE 8.1i, Xilinx Ltd, March 2006
http://www.xilinx.com/products/boards/s3estarter/files/s3esk_picoblaze_nor_flash_programmer.pdf
- [6f] Ken Chapman, SPI FLASH Programmer for Spartan-3E Starter Kit ISE 7.1i, Xilinx Ltd, November 2004
http://www.xilinx.com/products/boards/s3estarter/files/s3esk_picoblaze_spi_flash_programmer.pdf
- [6g] Ken Chapman, Frequency Generator for Spartan-3E Starter Kit, ISE 8.2i, Xilinx Ltd, 18th July 2006
http://www.xilinx.com/products/boards/s3estarter/files/s3esk_frequency_generator.pdf
- [6h] Ken Chapman, Frequency Counter for Spartan-3E Starter Kit (with test oscillators), ISE 8.1i, Xilinx Ltd, 7th March 2006
http://www.xilinx.com/products/boards/s3estarter/files/s3esk_frequency_counter.pdf
- [6i] Ken Chapman, Software Implementation of Pulse Width Modulation (PWM). A reference design using the Spartan-3E Starter Kit, ISE 8.1i, Xilinx Ltd, 24th May 2006
http://www.xilinx.com/products/boards/s3estarter/files/s3esk_picoblaze_pwm_control.pdf