

CLOCK DOMAIN CROSSING INTERFACES

Jakub Cabal

Bachelor Degree Programme (3), FEEC BUT

E-mail: xcabal05@stud.feec.vutbr.cz

Supervised by: Marek Bohrn

E-mail: bohrn@feec.vutbr.cz

Abstract: This work presents an easy-to-use library of clock domain crossing modules and a methodology for its use. These crossings are inevitable in moderately complex firmware designs. Incorrectly implemented clock domain crossing modules can lead to data corruption or data loss. For correct functionality of these crossings it is necessary to apply correct constraints. Automatic application of constraints is a part of the created library. Its easy use is also supported by the methodology for selection of correct clock domain crossing module in the form of a decision tree.

Keywords: FPGA, CDC, metastability, Vivado

1. ÚVOD

Pro obvody FPGA vznikají různé aplikace, u kterých je často nutné použít více hodinových domén. Při propojení dvou hodinových domén signálem nebo skupinou signálů vzniká asynchronní přechod. Špatně realizovaný asynchronní přechod může vést k poškození či ztrátě přenášené informace.

Oblast bezpečného řešení asynchronních přechodů mezi hodinovými doménami není triviální a nepatří mezi standardní znalost u návrhářů digitálních obvodů.

Cílem tohoto příspěvku je stručně představit problematiku asynchronních přechodů, popsat implementovanou knihovnu asynchronních přechodů a její použití v praxi a dále představit vytvořenou metodiku použití asynchronních přechodů z této knihovny.

2. PROBLEMATIKA ASYNCHRONNÍCH PŘECHODŮ

Při přenosu informací mezi hodinovými doménami může dojít ke čtyřem různým chybám. Při přechodu z rychlé do pomalé hodinové domény se mohou ztratit data, při přechodu z pomalé do rychlé hodinové domény se mohou některá data zachytit vícekrát, v synchronizačním obvodu se může objevit metastabilní stav anebo může dojít k poškození dat vlivem malého časového posunu [1]. Správné řešení asynchronních přechodů se liší pro jednobitové a vícebitové přechody.

2.1. JEDNOBITOVÉ SYNCHRONIZAČNÍ OBVODY

Nejjednodušším řešením jednobitového asynchronního přechodu je základní synchronizační obvod složený ze dvou nebo i více klopných obvodů typu D [1].

Dalším možným řešením jednobitového asynchronního přechodu je synchronizační obvod se zpětnou vazbou [1]. Tento obvod je složen zejména ze dvou základních synchronizačních obvodů. Jeho nevýhodou je větší zpoždění.

2.2. VÍCEBITOVÉ SYNCHRONIZAČNÍ OBVODY

Častým řešením vícebitového asynchronního přechodu je synchronizační obvod využívající asynchronní paměti FIFO, která přímo oddělí zdrojovou a cílovou hodinovou doménu [1]. Asynchronní

FIFO lze v současných obvodech FPGA naimplementovat poměrně jednoduše za pomoci dvouportových blokových pamětí doplněných o řídicí logiku s generátory příznaků plné a prázdné paměti.

Druhou možností, jak vyřešit vícebitový asynchronní přechod, je použít synchronizační obvod využívající metody handshake [1]. Přenos dat mezi hodinovými doménami je v tomto řešení řízen pomocí dvou stavových automatů. Hlavní nevýhodou tohoto řešení je jeho nižší rychlost.

3. KNIHOVNA ASYNCHRONNÍCH PŘECHODŮ

V rámci této práce byla vytvořena knihovna asynchronních přechodů, která je již v současné době nasazena v platformě NetCOPE. Tato platforma slouží pro rychlý vývoj hardwarově akcelerovalých síťových aplikací, například pro akcelerační kartu se 100 Gb/s síťovým rozhraním.

Vytvořená knihovna obsahuje pro řešení jednobitových asynchronních přechodů základní synchronizační obvod (ASYNC_OPEN_LOOP), synchronizační obvod se zpětnou vazbou (ASYNC_GENERAL) a synchronizační obvod pro reset (ASYNC_RESET), který je obdobou základního synchronizačního obvodu. Dále knihovna obsahuje variantu základního synchronizačního obvodu (ASYNC_OPEN_LOOP_SMD) určenou pro synchronizaci jednotlivých bitů Grayova čítače.

Pro řešení vícebitových asynchronních přechodů knihovna obsahuje synchronizační obvod využívající handshake metody (ASYNC_BUS_HANDSHAKE) a tři synchronizační obvody využívající asynchronní FIFO, kde první varianta (ASFIFO) používá distribuovanou paměť, druhá varianta (ASFIFO_BRAM) používá blokovou paměť a třetí varianta (ASFIFO_BRAM_7SERIES), určená pro FPGA řady Virtex-7 od společnosti Xilinx, umožňuje využívat blokové paměti přímo v režimu asynchronní FIFO [2]. Z těchto obvodů jsou dále odvozeny synchronizační obvody pro proprietární datové sběrnice MI32, FLU, FL a DMA.

3.1. OMEZUJÍCÍ PODMÍNKY

Na každý naimplementovaný synchronizační obvod, který je součástí vytvořené knihovny asynchronních přechodů, jsou aplikovány vhodné omezující podmínky (constraints). Výběr a aplikace správných omezujících podmínek představují klíčové kroky při vývoji korektních asynchronních přechodů. Omezující podmínky jsou pro každý synchronizační obvod specifické. V prostředí Vivado Design Suite se nejčastěji používají tyto: *async_reg* – zajišťuje umístění synchronizačních klopných obvodů blízko sebe, *set_false_path* – zajišťuje výjimku z časové analýzy na konkrétní cestu, *set_max_delay* – nastavuje maximální zpoždění signálu na konkrétní cestě [3].

3.2. POUŽITÍ KNIHOVNY ASYNCHRONNÍCH PŘECHODŮ

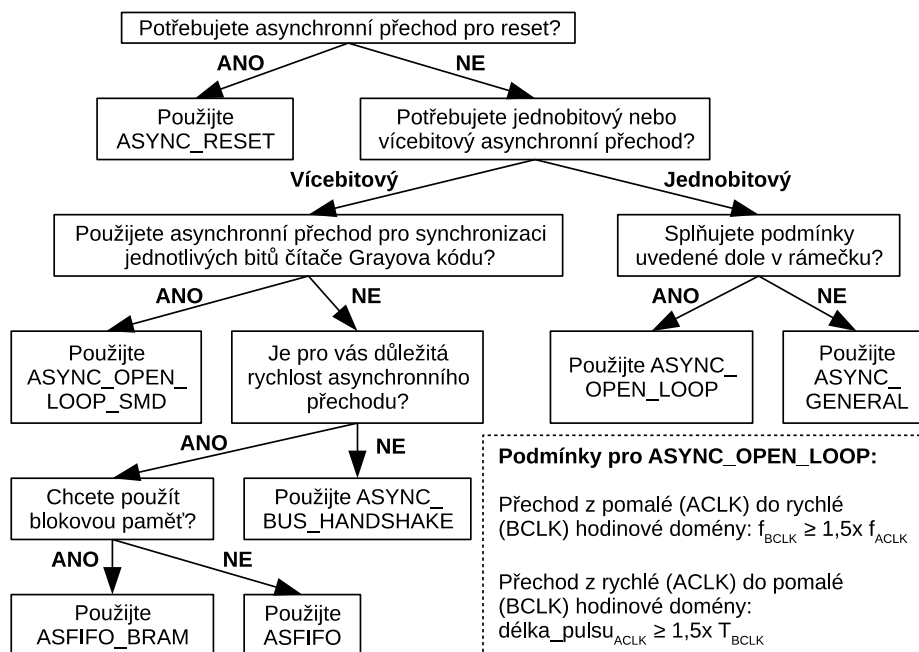
Největším přínosem vytvořené knihovny je její velmi snadné použití v praxi. Stačí pouze vybrat vhodný asynchronní přechod a následně ho použít na požadované místo v obvodu. Omezující podmínky budou na použitý asynchronní přechod aplikovány zcela automaticky. Toho je docíleno použitím příkazu *SCOPED_TO_REF*, který lze použít ve vývojovém prostředí Vivado Design Suite. Tento příkaz zajistí to, aby se na konkrétní entitu automaticky aplikovaly omezující podmínky připravené v konkrétním XDC souboru přiloženém k použité komponentě [4].

4. METODIKA POUŽITÍ ASYNCHRONNÍCH PŘECHODŮ

V rámci práce byla také vypracována metodika použití vytvořených asynchronních přechodů. Tato metodika je ve formě rozhodovacího stromu a výrazně usnadňuje výběr správného synchronizačního obvodu pro konkrétní asynchronní přechod, který je potřeba správně vyřešit. Zjednodušená verze rozhodovacího stromu je zobrazena na obrázku 1.

Rozhodovací strom provede vývojáře digitálních obvodů přes několik důležitých otázek, zdali jde o vícebitový asynchronní přechod, jestli je vyžadováno použít blokovou paměť, zda bude použito FPGA řady Virtex-7 od společnosti Xilinx a další otázky. Po zodpovězení všech otázek získá návr-

hář digitálních obvodů informaci, které řešení z knihovny asynchronních přechodů použít. Rozhodovací strom byl vytvořen podle znalostí získaných z použité literatury, ale také podle zkušeností s řešením asynchronních přechodů v návrhu digitálních obvodů v praxi.



Obrázek 1: Zjednodušená varianta vytvořeného rozhodovacího stromu.

5. ZÁVĚR

V rámci tohoto příspěvku byl proveden rozbor problematiky asynchronních přechodů mezi hodinovými doménami. Dále byla představena a popsána vytvořená knihovna asynchronních přechodů, která usnadňuje řešení asynchronních přechodů v praxi a již v současnosti je úspěšně nasazena v platformě NetCOPE používané sdružením CESNET. Popsána byla metodika použití asynchronních přechodů a způsob automatické aplikace připravených omezujících podmínek.

PODĚKOVÁNÍ

Rád bych poděkoval odbornému konzultantovi panu Ing. Jiřímu Matouškovi za odbornou pomoc.

REFERENCE

- [1] CUMMINGS, Clifford. SUNBURST DESIGN. Clock Domain Crossing (CDC) Design & Verification Techniques Using SystemVerilog. Boston, USA, 2008. Dostupné z: http://www.sunburst-design.com/papers/CummingsSNUG2008Boston_CDC.pdf
- [2] XILINX INC. 7 Series FPGAs Memory Resources: User Guide - UG473. San Jose, USA, November 2014. Dostupné z: http://www.xilinx.com/support/documentation/user_guides/ug473_7Series_Memory_Resources.pdf
- [3] XILINX INC. Vivado design suite advanced XDC and STA for ISE software users. San Jose, USA, 2012.
- [4] XILINX INC. Vivado Design Suite User Guide: Using Constraints - UG903. San Jose, USA, October 2014. Dostupné z: http://www.xilinx.com/support/documentation/sw_manuals/xilinx2014_4/ug903-vivado-using-constraints.pdf