

# FPGA-BASED DDR SDRAM DRIVER

Viktor PUŠ, Bachelor Degree Programme (3)  
Dept. of Computer Systems, FIT, BUT  
E-mail: xpusvi00@stud.fit.vutbr.cz

Supervised by: Ing. Tomáš Pečenka

## ABSTRACT

In the paper the architecture of the developed FPGA based DDR SDRAM driver is presented. In the beginning of paper the characteristic of DDR SDRAM memory is briefly introduced. Then the architecture of developed hardware driver is presented. At the end of the paper the implementation issues and performance optimizations are briefly discussed.

## 1 ÚVOD

Dnešní moderní síťové aplikace (jako jsou například směrovače, monitorovací adaptéry, systémy detekce útoku, atd.) často vyžadují použití specializovaných hardwarových platform. Tyto platformy pak potřebují pro svou činnost dostatečnou paměťovou kapacitu. Vhodným typem paměti, které nabízí dostatečnou paměťovou kapacitu (128 MB – 2 GB) a datovou propusnost (až 3.2 GB/s) jsou paměti typu DRAM. Cílem této práce je navrhnout univerzální řadič paměti DDR SDRAM pro síťové aplikace na bázi technologie FPGA

## 2 VLASTNOSTI POUŽITÉ PAMĚTI DDR SDRAM

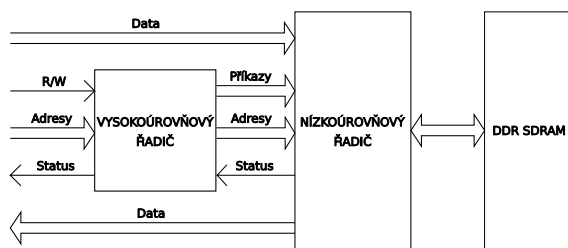
Paměť SDRAM (Synchronous Dynamic Random Access Memory) je synchronní dynamická paměť. Zkratka DDR znamená, že se data přenášejí z/do paměti jak na náběžnou, tak sestupnou hranu hodinového signálu. Implementace řadiče bude realizována na síťové platformě COMBO6X [2]. Tato platforma používá paměť DDR SDRAM DIMM 266MHz. Modul paměti má datovou sběrnici o šířce 64 nebo 72 bitů.

Při každém přístupu (čtení nebo zápisu) do paměti je nutné, aby byl otevřený příslušný řádek, teprve potom se paměti předá adresa sloupce. Paměť je organizována do čtyř nezávislých banků. V každém banku paměti může být otevřen maximálně jeden řádek. Zde se jako výhoda projevuje možnost současné práce s více banky, protože např. během čtení z banku 1 může začít zavírání řádku v banku 2.

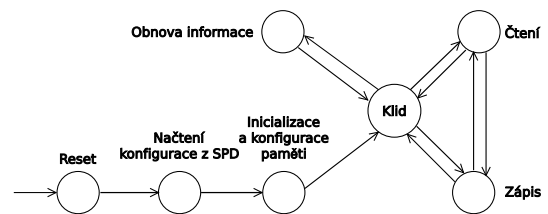
Vzhledem k technologii paměti je nutné s jistým maximálním časovým intervalem provádět obnovu informace. Obnovu realizuje řadič zasláním speciálního příkazu. Parametry pamětí od jednotlivých výrobců se mohou lišit. Řadič může informace o připojené paměti získat pomocí jednoduchého sériového protokolu SPD. SPD se typicky používá pro konfiguraci řadiče, který si před zahájením práce s pamětí může zjistit její parametry (velikost, organizace, podporované frekvence a časování) a upravit podle nich svou konfiguraci

### 3 ARCHITEKTURA ŘADIČE

Architektura je modulární, skládá se ze dvou částí. Zjednodušeně je zachycena na obrázku 1.



Obrázek 1: Architektura řadiče



Obrázek 2: Zjednodušený stavový automat

**Nízkoúrovňový řadič** je připojen k rozhraní k paměti. Úkolem řadiče je generovat řídicí signály paměti a vytvářet určitou abstrakci nad rozhraním paměti pro *vysokoúrovňový řadič*. Vstupní data a příkazy jsou ukládány do oddělených front typu FIFO. Stavový automat pak analyzuje příkazy z jednotlivých front a generuje na jejich základě řídicí signály paměti. Automat také dodržuje správné časování řídicích signálů. Navržená jednotka podporuje paralelní práci s více banky, pro každý příkaz tedy vyhodnocuje, do jakého banku míří a kdy nejdříve je možné jej vyslat do paměti.

**Vysokoúrovňový řadič** provádí dekodování adres (na adresy čipu, banku, řádku a sloupce), dále překládá požadavky na čtení a zápis na příkazy pro nízkoúrovňový řadič a zajišťuje časování požadavků na obnovu paměti. Vysokoúrovňový řadič poskytuje uživateli rozhraní pro komunikaci s pamětí.

#### Operace řadiče

- Inicializace – Po resetu je před zahájením běžné práce vyčtena konfigurace z SPD, podle ní je nastaveno chování řadiče a do paměti je vyslána inicializační sekvence v souladu se standardem [1].
- Zápis – automat spouští obvody vystavující data na datovou sběrnici.
- Čtení – automat spouští obvody záchytu dat. Vyčtená data jsou ukládána do výstupní fronty.
- Obnova informace – řadič posílá tento příkaz do paměti periodicky s intervalem 7 ns.

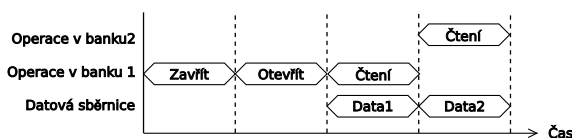
## 4 OPTIMALIZACE

V případě nevhodné nebo žádné optimalizace přístupu může efektivní přenosová rychlost klesnout i pod 10 % maximální hodnoty. Proto je nutné příkazy vhodně optimalizovat a měnit jejich pořadí tak, aby byla maximálně využita možnost paralelní práce s banky. Toto je úkol pro vysokoúrovňový řadič.

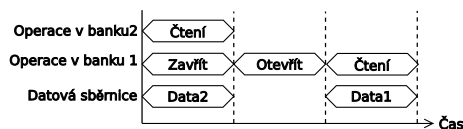
V rámci navrženého řadiče bylo realizováno několik druhů optimalizací. Základním vylepšením je neuzavírat po každém přístupu příslušný řádek a v případě, že další přístup do daného banku je i do stejného řádku, není již nutné řádek znovu otevírat.

Dalším krokem je zpracovávání požadavků mimo pořadí. HW řešení tohoto problému je možné realizovat pomocí čtyřech front typu FIFO. Požadavky jsou do front vkládány podle cílového banku. Konečný automat pak požadavky z jednotlivých front řadí ve výhodnějším pořadí, přičemž však bere ohled na možné datové konflikty v paměti.

*Příklad:* Dva požadavky na čtení, první do banku 1, kde je třeba uzavřít starý a otevřít nový řádek. Druhý požadavek je do banku 2, ale v tomto banku je již potřebný řádek otevřený. Je výhodnější nejprve provést čtení z banku 2 a v průběhu této operace otevřít potřebný řádek v banku 1. První požadavek na čtení bude proveden jako druhý, ale celková doba provádění obou požadavků bude v tomto pořadí menší.



Obrázek 3: Původní pořadí operací



Obrázek 4: Pořadí operací po optimalizaci

Během řazení požadavků se ukládá dočasná informace o změnách v pořadí a vyčtená data jsou opět složena do původního pořadí v dalších čtyřech frontách.

Výhodou navržených optimalizací je vyšší propustnost, avšak nevýhodou je vyšší latence čtení. Celý řadič totiž funguje jako zřetěžená linka, která se prodlužuje s každou frontou, autematem a podobně.

## 5 ZÁVĚR

Výsledkem práce je návrh a implementace řadiče paměti DDR SDRAM s využitím technologie FPGA. Řadič umožňuje realizovat přenos rychlostí až 1,5 GB/s. V současné době je řadič naimplementován a probíhá fáze ladění průběhů čtecích a zápisových operací na cílové platformě.

## REFERENCE

- [1] JEDEC Standard No. 79D: DOUBLE DATA RATE (DDR) SDRAM SPECIFICATION, dokument dostupný na URL <http://www.jedec.org/DOWNLOAD/search/JESD79D.pdf>
- [2] Stránky projektu Liberouter, dokument dostupný na URL <http://www.liberouter.org>