

**ZÁPADOČESKÁ UNIVERZITA V PLZNI  
FAKULTA ELEKTROTECHNICKÁ**

**KATEDRA ELEKTROENERGETIKY A EKOLOGIE**

# **DIPLOMOVÁ PRÁCE**

**Návrh univerzální mikroprocesorové desky s FPGA a  
procesorem ARM**

**vedoucí práce: Ing. Jan Brož  
autor: Bc. Petr Lokajíček**

**2013**



## **Anotace**

Předkládaná diplomová práce obsahuje návrh univerzální mikroprocesorové desky s FPGA a procesorem ARM. Zběžně popisuje dnešní mikroprocesory ARM a programovatelné logické obvody typu FPGA, jejich architekturu a popis funkce. Dále pak konkrétní mikroprocesor společnosti ST Microelektronics STM32F107VCT6 a konkrétní obvod FPGA společnosti Altera EP1C3T100N jež byly použity při návrhu desky. K návrhu DPS je využito programu Altium Designer verze 10.

## **Klíčová slova**

Univerzální DPS, mikrokontroléry ARM, Cortex-M3, MCU, návrh DPS, Altium Designer, programovatelné logické obvody FPGA.

## **Abstract**

Submitted diploma thesis contains a proposal for universal microprocessor board with FPGA and ARM microprocessor. briefly describes the current ARM microprocessors and programmable logic circuits called FPGA, their architecture and functional description. Furthermore, the specific microprocessor company ST Microelektronics STM32F107VCT6 and specific FPGA company Altera EP1C3T100C7N, that were used on the board, are described. Altium Designer version 10 is used to design this PCB.

## **Key words**

Universal PCB, microcontroller ARM, Cortex - M3 MCU, PCB design, Altium Designer, FPGA programmable logic circuits.

## **Prohlášení**

Předkládám tímto k posouzení a obhajobě diplomovou práci, zpracovanou na závěr studia na Fakultě elektrotechnické Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této diplomové práce.

Dále prohlašuji, že veškerý software, použitý při řešení této diplomové práce, je legální.

V Plzni dne 10.5.2013

Bc. Petr Lokajíček

.....

## **Poděkování**

Rád bych poděkoval svému vedoucímu panu Ing. Janu Brožovi za odborné vedení a nasměrování při zpracování mé diplomové práce. Děkuji své rodině, především mojí mamince, přátelům a kolegům za podporu během mého studia. Nakonec bych rád vyjádřil své děkuji všem pedagogům a pracovníkům Západočeské univerzity v Plzni jež mi umožnily získání odborných znalostí.

V Plzni dne 6.5.2013

Bc. Petr Lokajíček

.....

## Obsah

Seznam použitých zkratk	8
Úvod	9
1. Procesory ARM	10
1.1 Architektura ARM	10
1.1.1 Architektura RISC	10
1.1.1.1 Základní charakteristika procesorů ARM:	11
1.1.2 Instrukční sady ARM	12
1.2 Dostupné mikroprocesory s jádrem ARM	13
1.2.1 Cortex-M3	14
1.2.2 Použitý mikrokontrolér ARM – STM32F107VCT6	15
1.2.2.1 Periferie mikrokontroléru:	15
1.2.2.2 Možnosti lazení:	16
1.2.3 Využití pinů MCU - MicroXplorer	16
2 Integrované obvody FPGA	18
2.1 Architektura FPGA	19
2.2 Dostupné FPGA obvody	21
2.2.1 FPGA ALTERA	21
2.2.1.1 Altera Stratix	22
2.2.1.2 Altera Arria	22
2.2.1.3 Altera Cyclone	22
2.2.2 FPGA XILINX	23
2.2.2.1 Rodiny FPGA obvodů Xilinx	24
2.3 Zvolené FPGA	25
2.3.1 Specifikace EP1C3T100C7N:	25
2.3.2 Multivolt I/O support	25
3 Návrh univerzální DPS	27
3.1 Možnosti využití DPS	28
4 Vlastní návrh mikroprocesorové desky	28
4.1 Altium Designer	29
5.1 Napájení DPS	31
5.2 Časování	33
5.3 Periferie	34
5.3.1 8-bitová sběrnice	34
5.3.2 Rozhraní JTAG	34
5.3.3 Altera EPCS4SI8N	34
5.3.4 I2C sběrnice	34
5.3.5 USB OTG	35
5.3.6 RS-232 a SD karta	35
5.4 Schema zapojení MCU ARM	37
5.5 FPGA	37
6 PCB layout	38
Závěr	41
Použitá literatura	42
Technické manuály	42
Příloha 1 A – přehled procesorů ARM	43
Příloha 1 B – přehled procesorů ARM pokračování	44

## Seznam použitých zkratk

ALU	Arithmetic Logic Unit	Aritmeticko logická jednotka
ARM	Advanced RISC Machines	dříve Acron RISC Machines
CISC	Complex Instruction Set Computing	Architektura procesorů využívajících sady komplexních instrukcí
CRC	Cyclic redundancy check	Cyklický redundantní součet
DMA	Direct Memory Access	Přímý přístup k paměti
ETH	Ethernet	Ethernet
FPGA	field Programmable Gate Array	Programovatelné hradlové pole
GPIO	General purpose input output	Univerzální vstup / výstup
I/O	Input Output	Vstup výstup
IOB	Input Output Bank	Banka vstupů a výstupů
ISP	In System Programing	Programování/konfigurace za provozu systému
JTAG	Joint Test Action Group	rozhraní pro testování/ladění programů
LB	Logic Blok	Logický blok
LUT	LookUp Table	Vyhledávací tabulka
MCU	Microcontroller	Mikrokontrolér
OTG	On The Go	USB Host
PCB	Printed Circuit Board	Deska plošného spoje - DPS
PLD	Programmable Logic Device	Programovatelné logické obvody
PLL	Phase Locked Loop	Fázový závěs
RAM	Random Access Memory	Paměť s náhodným (libovolným) přístupem
RISC	Reduced Instruction Set Computing	architektura mikroprocesorů využívajících redukované sady instrukcí
SIMD	Singe Instruction Multiple Data	Architektura procesorů specializovaných na velké přenosy dat / výpočtů
SRAM	Static Random Access Memory	Statická paměť
SWD	Serial Wire Debug	Lezení přes seriovou linku
USB	Universal Serial Bus	Univerzální sériová sběrnice
VHDL	VHSIC Hardware Description Language	Programovací jazyk pro popis hardware



## Úvod

Cílem mé diplomové práce je seznámit čtenáře s dostupnými integrovanými obvody FPGA a procesory ARM. Tyto obvody, zejména FPGA se v posledních letech těší čím dál tím větší oblibě u návrhářů a dostávají se tak téměř do každého elektronického zařízení s nímž přicházíme denně do styku.

Důvodem rostoucí popularity je nepochybně klesající cena, nižší spotřeba a zároveň rostoucí výkon těchto obvodů. Samozřejmě stále nově vyvíjené a zlepšující se návrhové prostředí, což velmi usnadňuje jak návrh, tak i samotné programování a ladění programů zmíněných obvodů.

Dále se v práci zabývám návrhem univerzální mikroprocesorové desky s již zmíněnými obvody FPGA a ARM. Pro návrh jsem využil program Altium Designer, který přináší celou škálu nových a zajímavých funkcí, možností a nastavení do návrhů jak klasických obvodů, tak i programovatelných logických obvodů.

Na závěr je popsána realizace celého návrhu a představen výsledný návrh DPS. Následují chemata zapojení DPS a také obrázky PCB layoutu všech 4 vrstev. Pro zajímavost jsem také vygeneroval 3D zobrazení navržené DPS a to jak z pohledu vrchní strany, tak i z pohledu strany spodní.

## **1. Procesory ARM**

Původní znění zkratky ARM bylo Advanced RISC Machines, poté se nějakou dobu používalo označení Acorn RISC Machines. Dnes se používá už jen samotné ARM. První procesor vyvinula společnost ARM Limited. Dnes je ARM jedním z nejznámějších pojmů na poli integrovaných obvodů.

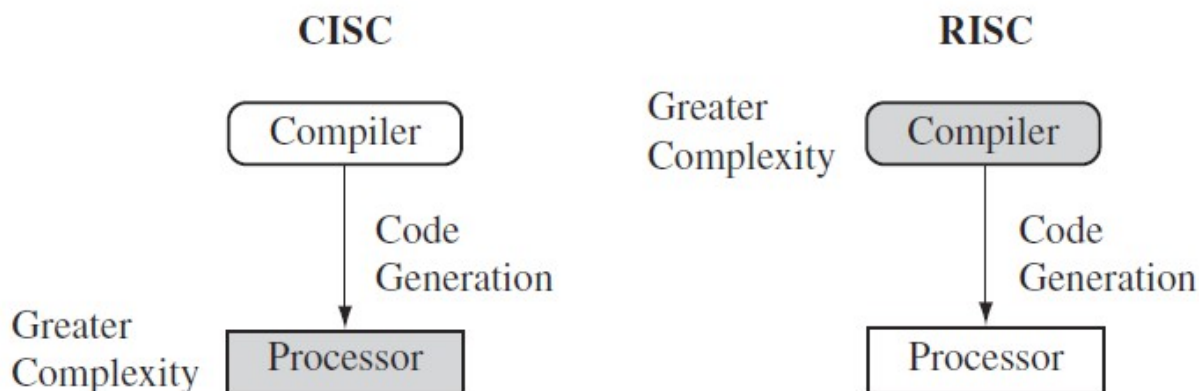
### **1.1 Architektura ARM**

#### **1.1.1 Architektura RISC**

Procesory ARM jsou založeny na architektuře RISC (Reduced Instruction Set Computing). V podstatě to znamená, že procesory architektury RISC využívají oproti procesorům využívajících CISC (Complex Instruction Set Computing) menší počet instrukcí, které mají pevnou délku 32-bitů. U procesorů architektury CISC jsou využívány instrukce proměnné délky, což jim sice umožňuje zahrnout v instrukční sadě komplexnější funkce, avšak je to na úkor vyššího zatížení procesoru. U architektury RISC je sice potřeba více instrukcí ke zvládnutí například operace dělení, zato se ale díky konstantní délce mohou načíst následující instrukce, což celý proces značně zrychluje a to bez zbytečného zatížení procesoru. Jedna instrukce se provede vždy během jednoho hodinového taktu. Existují čtyři hlavní rysy RISC architektury. [2]

1. Instrukce – Menší počet instrukcí v instrukční sadě, instrukce je pevné délky 32-bitů a je provedena vždy během jednoho hodinového cyklu.
2. Pipelines (řetězení instrukcí) – Zpracování instrukcí je rozděleno na menší části, které jsou zpracovány paralelně pomocí pipelines.
3. Registry – Velké množství univerzálních registrů, které mohou obsahovat data nebo adresu. Slouží tím jako rychlá místní paměť pro uložení dat pro všechny operace využívající data.

4. load-store architektura – procesor pracuje s daty uloženými v registrech, nemusí tak přistupovat mnohonásobně k vnější paměti. Přenos dat mezi registry a vnější paměti je proveden pomocí oddělených instrukcí pro nahrání a uložení.



Obr. 1.1: Porovnání hardwarové náročnosti architektur RISC a CISC [2]

#### 1.1.1.1 Základní charakteristika procesorů ARM:

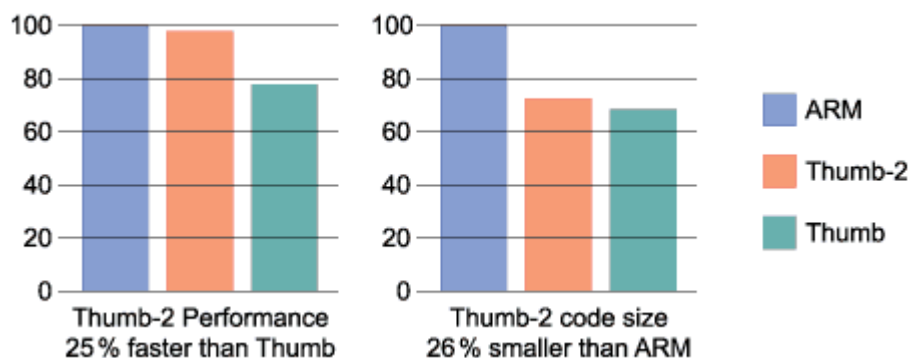
- 32-bitová architektura
- 32-bitová datová sběrnice
- 26-bitová adresová sběrnice
- 27 32-bitových registrů
- nejdelší doba reakce na přerušení 3 milisekundy
- možnost podmíněného vykonání instrukcí
- jednoduchý a výkonný instrukční soubor, jednoduše využitelné kompilátory vyšších programovacích jazyků

## 1.1.2 Instrukční sady ARM

V dnešní době představuje architektura ARM efektivní způsob využití jednočipových RISC mikrokontrolérů v celé řadě elektronických zařízeních. Podíl architektury ARM na trhu se pohybuje okolo 75%, což je mimo jiné způsobeno dosažením velmi dobrých výsledků v oblasti výkonu, spotřeby a výrobní ceny jednotlivých čipů.

Procesory ARM využívají základní instrukční sadu RISC. Ta však vyžaduje 32-bitovou šířku operačního kódu instrukce, a tak se postupně vyvíjejí další instrukční sady, resp. rozšíření původní sady instrukcí RISC. Instrukční sada Thumb využívá pouze 16-bitovou šířku operačního kódu, resp. se 32-bitová instrukce zabalí do 16-bitové, která se pak při využití rozbalí zpět na 32-bitovou. Operace rozbalení instrukce Thumb na instrukci ARM nepotřebuje žádný cyklus navíc. Výhodou je využití poloviční šířky 32bitové sběrnice, což sníží spotřebu a úspora paměti programu. Následně byla vyvinuta také vylepšená sada instrukcí Thumb, instrukční sada Thumb-2. Srovnání těchto sad je znázorněno na obrázku Obr.1. Další instrukční sada ARM Jazelle (technologie pro podporu JAVA) a SIMD (Single Instruction Multiple Data). Ta byla vyvinuta pro efektivní využití velkých datových polí, kterých je využíváno v multimediální sféře (audio video kodeky). [3][1]

Podrobné informace nejen o dalších instrukčních sadách se lze dozvědět z webových stránek ARM information center [5] nebo také v literatuře [4].



Obr. 1.2: Srovnání instrukčních sad ARM, Thumb a Thumb-2 [3]

## 1.2 Dostupné mikroprocesory s jádrem ARM

Společnost ARM se zabývá jen vývojem samotných procesorů a ne jejich výrobou. Politika společnosti je taková, že prodává licence dalším společnostem, které pak přidávají různé periferie k jádrům procesorů ARM do svých integrovaných obvodů. Procesory, resp. mikrokontroléry se pak liší zejména ve velikosti integrované paměti. Lze se tak setkat s řadou společností jež prodávají integrované obvody (mikroprocesory či mikrokontroléry) obsahující jádro ARM. Nejznámější z nich jsou například: ST Microelectronics, ATMEL, FREESCALE, NPX, Texas Instruments a neustále rostoucí řada dalších společností. Dostupné mikroprocesory ARM jsou rozděleny do takzvaných rodin (ARM family). Jednotlivé rodiny pak mohou obsahovat více jak jednu architekturu a také více typů jader mikroprocesorů. Přehled všech rodin, architektur a jader mikroprocesorů ARM spolu s přehledem funkcí, informací o paměti a výkonu je přehledně znázorněno v tabulkách v přílohách 1 A a 1 B. Poněkud zmatečné dřívější značení starších rodin a architektur typu ARM1, ARM2, až ARM11 vystřídalo přehlednější značení rodin typu Cortex-X, kde X zastupuje písmenné označení série A, R nebo M, kde

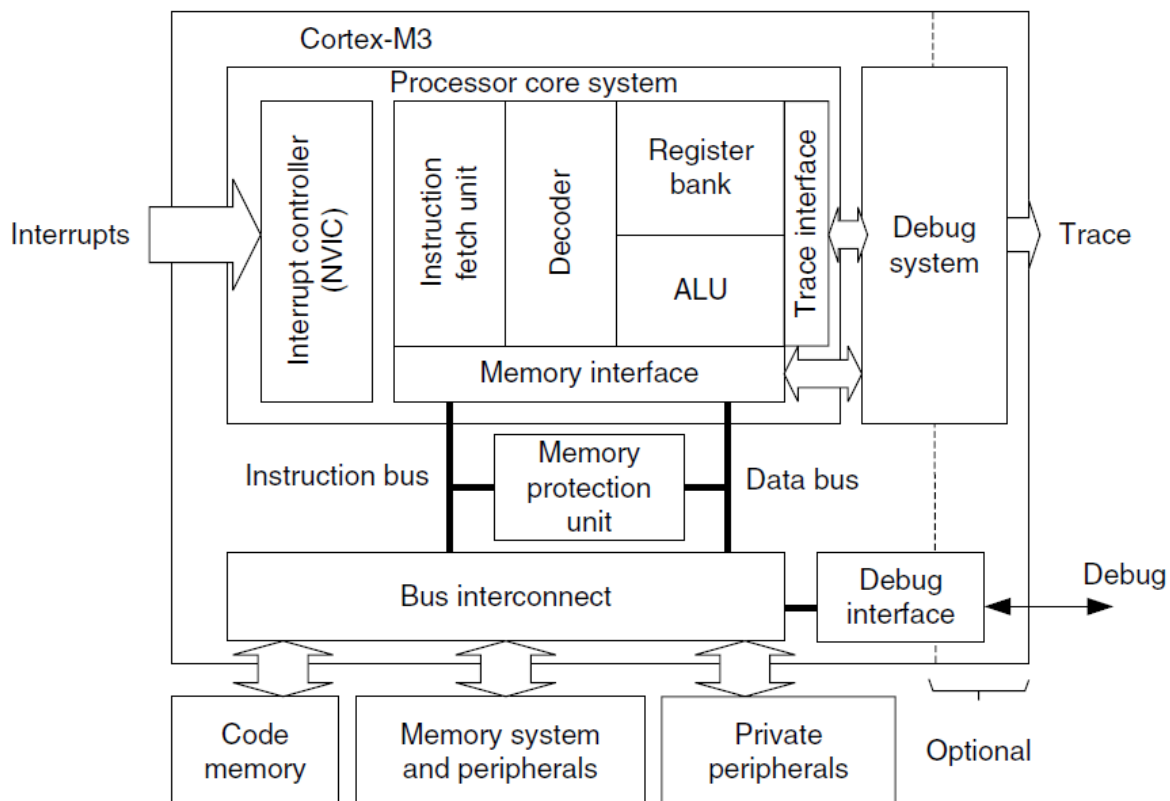
- **Cortex-A Serie** – řada aplikačních procesorů pro komplexní operační systém a uživatelské aplikace. Podporované instrukční sady jsou ARM, Thumb a Thumb-2. Rodina obsahuje procesory Cortex-A15 MPCore, ARM Cortex-A9 MPCore, ARM Cortex-A9 single core, ARM Cortex-A8, ARM Cortex-A7 MPCore, a ARM Cortex-A5.
- **Cortex-R Serie** – řada embedded procesorů optimalizovaná pro real-time systémy. Podporované instrukční sady ARM a Thumb. Rodina obsahuje procesory ARM Cortex-R5(F), ARM Cortex-R4(F) a ARM Cortex-R7 MPCore
- **Cortex-M Serie** – řada ještě více embedded procesorů než předchozí s ohledem na nízkou cenu. Podporována je pouze instrukční sada Thumb a Thumb-2. Do rodiny patří pouze ARM Cortex-M0, ARM Cortex-M0+, ARM Cortex-M1, ARM Cortex-M3, and ARM Cortex-M4.
- **Secure Core** – Speciální řada 32-bitových procesorů obohacená o bezpečnostní prvky.

### 1.2.1 Cortex-M3

Jak jsem zmínil v předchozí kapitole, jedná se o rodinu 32-bitových procesorů vyvinutou pro levná embeded řešení. mikrokontrolér této konkrétní série si našel uplatnění v řadě aplikací jako jsou automobilové řídicí systémy, systémy pro řízení motorů, průmyslové řídicí prvky, inteligentní měřicí systémy, pokročilé senzory či řízení domácích spotřebičů jako jsou ledničky, pračky, myčky a mnohé další. [1][6]

Na Obr.1.3 je znázorněno zjednodušené blokové schéma procesoru. Z obrázku lze vyčíst, že se jedná o procesory Harwardské architektury, tedy mají oddělenou sběrnici pro instrukce a data. To umožňuje procesoru přistupovat zároveň k datům i instrukcím.

Využívá instrukční sadu Thumb a Thumb-2. Výhody použití této sady jsou popsány v kapitole 1.1.1.



Obr. 1.3: Zjednodušené blokové schéma procesorů Cortex-M3 [1]

## 1.2.2 Použitý mikrokontrolér ARM – STM32F107VCT6

Na tomto místě bych rád představil použitý mikroprocesor pro navrhovanou univerzální mikroprocesorovou desku. Zvolil jsem velmi dobře dostupný 32-bitový mikrokontrolér společnosti ST Microelectronics s označením STM32F107VCT6. Tento mikroprocesor patří do rodiny Cortex-M3. Je dodáván v pouzdře LQFP100 nebo LQFP64, která má snížený počet pinů. Vzhledem k univerzálnosti navrhované desky a minimálnímu cenovému rozdílu jsem zvolil variantu s větším počtem I/O pinů, tedy variantu LQFP100. Specifikace mikrokontroléru STM32F107VCT6 je přehledně znázorněna v následující tabulce 1.

Typ integrovaného obvodu	mikrokontrolér ARM
Kapacita paměti Flash	256kB
Kmitočet taktování	72MHz
Pouzdro	LQFP100
Montáž	SMD
Počet vstupů/výstupů	80
Kapacita paměti SRAM	64kB
Napájecí napětí	2...3.6V DC
Druh architektury	Cortex-M3

Tabulka 1: Základní specifikace mikrokontroléru STM32F107VCT6 [6]

### 1.2.2.1 Periferie mikrokontroléru:

- 2x12-bitový A/D převodník (16 kanálů)
- 2x12-bitový D/A převodník
- 3xUSART
- 2xUART
- 3xSPI (pro 2 podpora I2S)
- 2xI2C
- 2xCAN (512 B SRAM)
- 10/100 Ethernet MAC s vlastní 4kB SRAM
- až 80 I/O pinů
- až 10xčasovač/čítač (až 4x16-bit časovače/čítače, 1x16-bit PWM časovač pro motory)

- 2xWatchdog
- USB 2.0 full-speed device/host/OTG
- CRC jednotka
- DMA – 12ti kanálový univerzální kontroler

### **1.2.2.2 Možnosti lazení:**

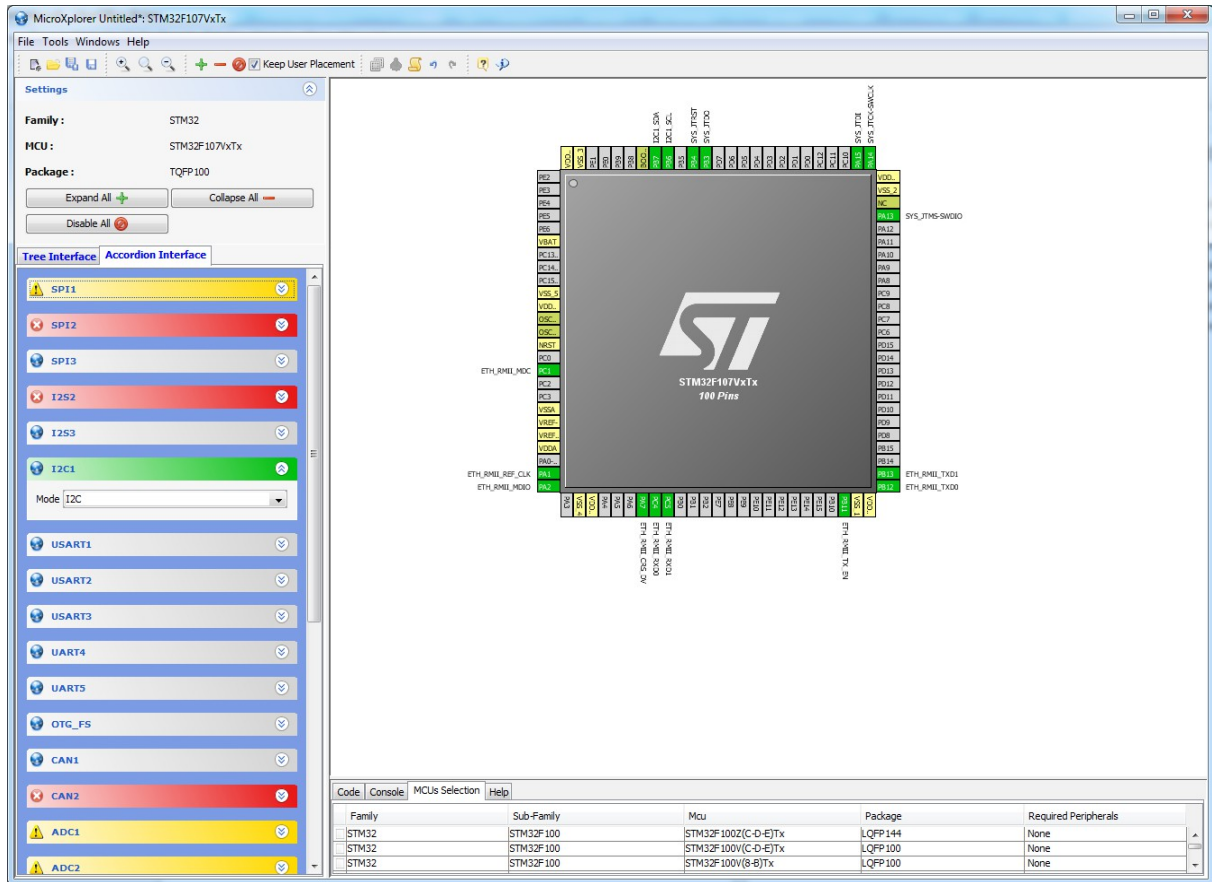
- JTAG
- SWD (Serial Wire Debug)

### **1.2.3 Využití pinů MCU - MicroXplorer**

Některé výše zmíněné periferie využívají stejné piny mikrokontroléru a nelze tedy použít všechny současně. Podrobné informace lze najít v datasheetu příslušného mikrokontroléru [20]. Ten je však značně obsáhlý a není jednoduché pamatovat si, která periferie využívá který pin.

Při návrhu se mi velmi osvědčil software vyvinutý společností ST Microelectronics s názvem MicroXplorer, který je dostupný zdarma na webu společnosti ST Microelectronics [11]. V něm je možné si zvolit konkrétní mikrokontrolér a zadávat požadované periferie. MicroXplorer poté vyznačí na kterých pinech se nachází jaké signály příslušných periférií a zároveň vyznačí, které periferie již není možno využít, případně nabídne jen jejich omezenou funkci. V případě, že máme k dispozici důvěryhodné knihovny periférií, a značení signálů se shoduje s označením uváděným MicroXplorerem minimalizuje se tím doba strávená listováním datasheety. Pracovní prostředí zmíněného softwaru je znázorněno na následujícím obrázku Obr. 1.4. Červeně vyznačené periferie již není možno využít vůbec a žlutě vyznačené jen některý režim.





Obr. 1.4: Pracovní prostředí programu MicroXplorer.

## 2 Integrované obvody FPGA

Obvody FPGA patří do skupiny programovatelných obvodů PLD (Programmable Logic Device). Ty se dají rozdělit na 3 základní skupiny. Klasické jednoduché PLD (SPDL), komplexní PLD (CPLD) a poslední skupinou jsou zmíněné obvody FPGA (field Programmable Gate Array), které se od ostatních liší především možnostmi programovatelnosti, dalo by se říci, „na místě“ (field – v poli). Jsou proto označovány jako ISP (In System Programming) obvody. Některé typy těchto obvodů lze dokonce programovat částečně i za běhu systému, kdy se část pole překonfiguruje, zatímco zbylá část pokračuje ve vykonávání své funkce. Díky této výhodné vlastnosti jsou obvody FPGA velmi flexibilní a lze je nakonfigurovat potřebám zákazníka přesně na míru. Díky těmto svým vlastnostem si již obvody FPGA našly cestu do celé řady zařízení zejména pak v průmyslovém odvětví. [9]

Jak už jsme u integrovaných obvodů zvyklí, i zde dochází postupně ke stále většímu zhušťování základního čipu i periférií, což má za následek snížení spotřeby a ceny a naopak umožňuje zvyšování výkonu. Velkou výhodou těchto obvodů je možnost programování na úrovni hardware. Znamená to, že lze přímo navrhnout funkce a propojení jednotlivých funkčních bloků do jakéhokoliv funkčního celku.

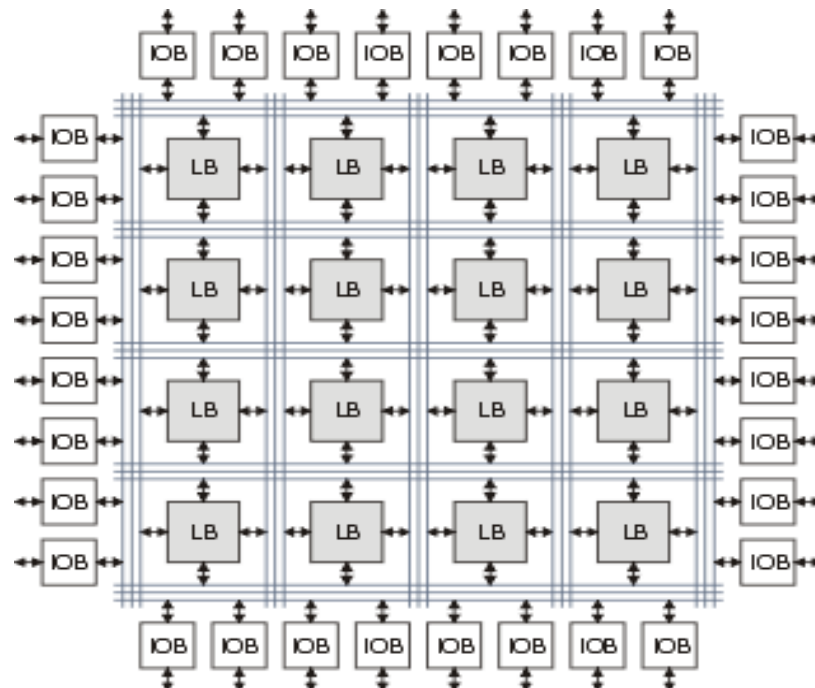
Možnost nakonfigurovat FPGA na jakoukoliv funkci, za předpokladu, že má dostatek logických obvodů, tedy výkonu, hraje velice důležitou roli v rozhodování výrobců jaký obvod mají pro svůj produkt použít. Zda zůstat u zaběhlého a zatím ještě pořád levnějšího konceptu mikroprocesor s perifériemi, nebo zda tyto mikroprocesory zcela nahradit modernější technologií. Stávající FPGA dokáží při dobře navržené konfiguraci napodobit chování mikroprocesorů a zcela je tak nahradit. Proč tedy výrobci zatím nenahradí mikroprocesory ve veškerých svých aplikacích za obvody FPGA když mají větší flexibilitu a vyšší výkon? Přestože v posledních letech ceny obvodů FPGA vytrvale klesají, pořád jsou dostupnější řešení s mikrokontroléry a to nejen díky koncové ceně. Je třeba si uvědomit další náklady spojené s přechodem k FPGA. Přepracovat osvědčená řešení zřejmě nemá velký smysl a tak zůstávají mikrokontroléry stále na svých místech. Samozřejmě v některých situacích má využití obvodů FPGA své opodstatnění. Zejména při malé, kusové sérii se FPGA vyplatí díky své flexibilitě. Naopak u větších sérií, kde se navíc hledí na každou korunu, je výhodnější řešení s levnějším MCU, který však zastane stejnou funkci. V některých případech je již nyní

finančně a časově výhodnější použití univerzálnějšího FPGA než MCU a FPGA obvody tak pomalu začínají v některých aplikacích vytlačovat mikrokontroléry.

## 2.1 Architektura FPGA

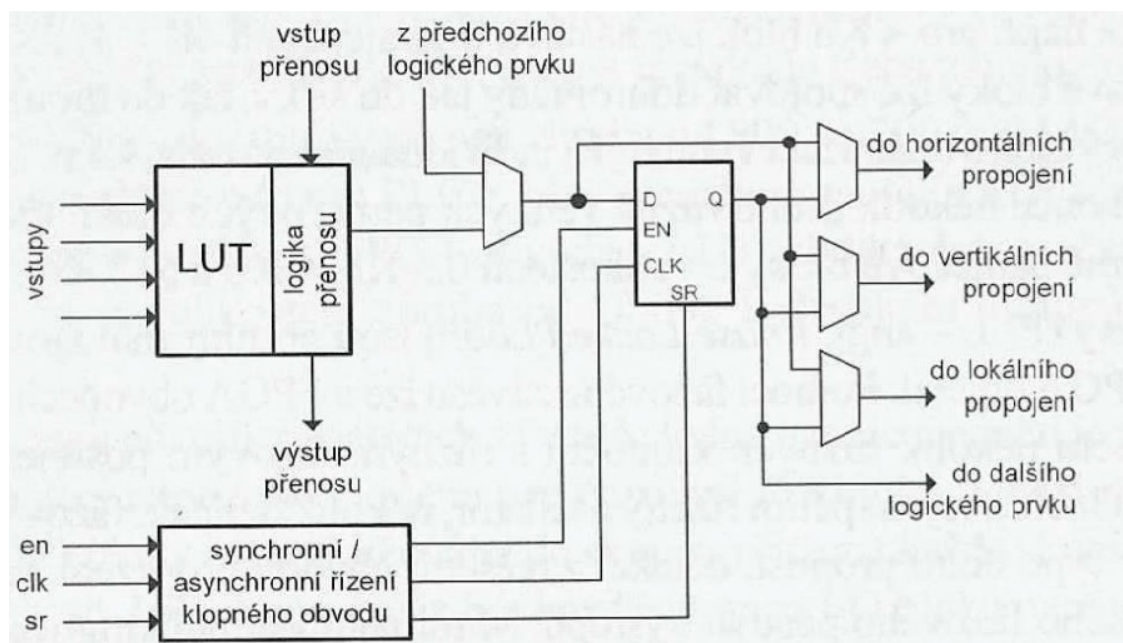
FPGA mají pravidelnou strukturu logických bloků LB (Logic Blok, Obr.5), které lze nakonfigurovat tak, aby plnili jednoduché logické funkce. Z těchto funkcí se pak dají vzájemným horizontálním či vertikálním propojováním vytvářet složité funkční celky. Propojení je možné také lokální nebo do dalšího bloku.

Obrázek 2.1 obecně znázorňuje strukturu FPGA. IOB jsou programovatelné vstupně výstupní obvody jednotlivých pinů FPGA a zastávají funkci buď pouze klasickou, jako programovatelné I/O, nebo mohou být nakonfigurovány pro potřeby periférií. Ty jsou pro přehlednost uspořádány do takzvaných bank, které mají společný zdroj napětí nebo stejnou referenční hodnotu výstupního proudu. Menší FPGA (100-200 pinů) mívá takové banky 4, jednu na každé straně obvodu. Detailnější pohled na LB nabízí Obr. 2.2. Mluvíme o takzvaných LUT tabulkách (LookUp Table), které generují požadovanou logickou funkci. [8] [9]



Obr. 2.1: Typická struktura obvodu FPGA [8]

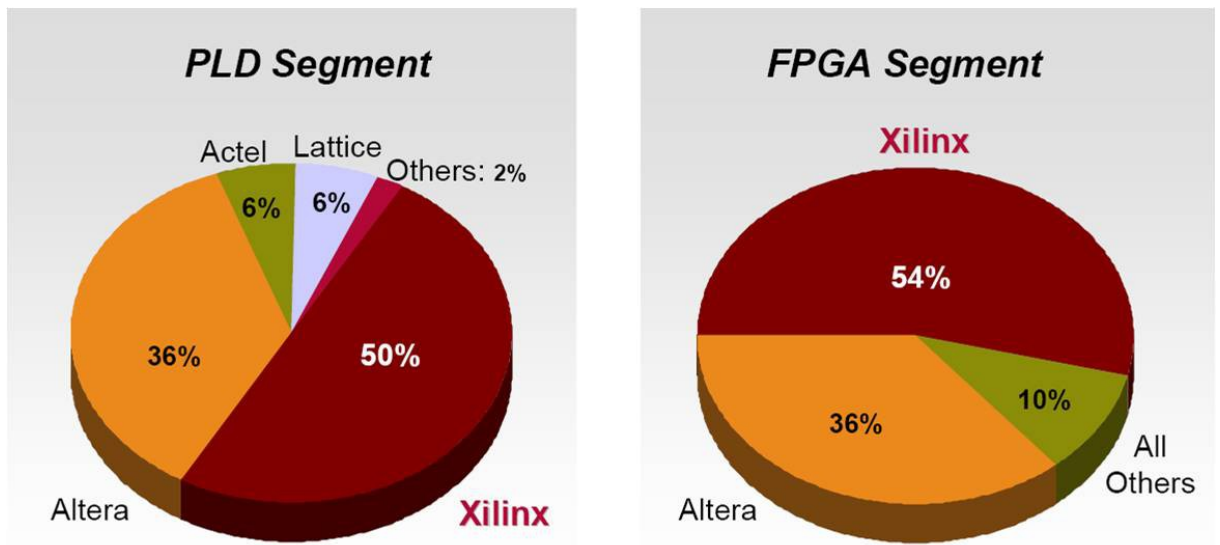
Kromě zmíněných základních stavebních prvků obsahují dnešní FPGA obvody další speciální bloky jako jsou paměti, násobičky, bloky sloužící k úpravě hodinových signálů či dokonce celé procesory. [9]



Obr. 2.2: Zjednodušené blokové schéma logického prvku [9]

## 2.2 Dostupné FPGA obvody

Při výběru FPGA obvodu se setkáme s celou řadou výrobců. Jsou to společnosti Altera, Xilinx, Lattice, Actel, Matrix Multimedi a další. Na následujícím obrázku 2.3 je vidět podíl trhu zmíněných společností. V této kapitole bych se krátce zmínil o dvou nejvýznamnějších.



Obr. 2.3: Podíl trhu z oblasti PLD a FPGA z roku 2009 [10]

### 2.2.1 FPGA ALTERA

Firma Altera je jednou z prvních společností, které začly s vývojem programovatelných integrovaných obvodů. Výrobou obvodů FPGA zabývá již řadu let a na trhu zaujímá stále silné postavení. Její obvody jsou v dnešní době k dostání ve třech sériích, kde se každá ze sérií zaměřuje na jiné parametry obvodů. Vyráběné série jsou **Stratix**, **Arria** a **Cyclone**. Pro konfiguraci svých FPGA obvodů nabízí vývojový software s názvem **Quartus II Design Software**, který je dostupný zdarma ve 30ti denní zkušební době. Více informací je možné získat na webových stránkách výrobce. Z následujících tabulek (2,3 a 4) je dobře patrné zmenšování výrobního procesu v průběhu minulých let.[7]

### 2.2.1.1 Altera Stratix

Jedná se o finančně náročnou high-end sérii zaměřenou především na vysoký výkon. Obvody jsou využívány pro nejnáročnější operace a byly použity například pro armádní letecké aplikace. Vyráběné řady této série jsou uvedeny v tabulce 2.1.

Device Family	Stratix	Stratix GX	Stratix II	Stratix II GX	Stratix III	Stratix IV	Stratix V
Year of introduction	2002	2003	2004	2005	2006	2008	2010
Process technology	130 nm	130 nm	90 nm	90 nm	65 nm	40 nm	28 nm
Recommended for new designs?	No	No	Yes	Yes	Yes	Yes	Yes

Tabulka 2.1: Vyráběné řady série Altera Stratix[7]

### 2.2.1.2 Altera Arria

Tato série střední třídy je vyvíjena s ohledem na spotřebu a nízké pořizovací náklady. Využívána je předně pro komunikační a embedded aplikace. Obvody dosahují propustnosti dat až 12,5Gb/s. Jsou podporovány protokoly PCI Express, Gigabit Ethernet či Serial RapidIO. Vyráběné řady této série jsou uvedeny v tabulce 2.2.

Device Family	Arria GX FPGA	Arria II GX FPGA	Arria II GZ FPGA	Arria V GX, GT, SX, ST FPGA	Arria V GZ FPGA
Year of introduction	2007	2009	2010	2011	2012
Process technology	90 nm	40 nm	40 nm	28 nm	28 nm
Recommended for new designs	Yes	Yes	Yes	Yes	Yes

Tabulka 2.2: Vyráběné řady série Altera Arria[7]

### 2.2.1.3 Altera Cyclone

Tato řada je navržena tak, aby splnila nároky na co možná nejnižší cenu a spotřebu. Je jasné, že nepatří mezi nejvýkonější čipy, ale i tak obvody této řady disponují výkonem, který si hravě poradí s řadou aplikací. Přehled dostupných řad této série je uveden v tabulce 2.3.

Device Family	Cyclone FPGA	Cyclone II FPGA	Cyclone III FPGA	Cyclone IV FPGA	Cyclone V FPGA
Year introduced	2002	2004	2007	2009	2011
Process technology	130 nm	90 nm	65 nm	60 nm	28 nm
Recommended for new designs	Yes	Yes	Yes	Yes	Yes

Tabulka 2.3: Vyráběné řady série Altera Cyclon[7]

## 2.2.2 FPGA XILINX

Tato společnost zaujímá dominantní postavení na trhu v oblasti programovatelných obvodů typu FPGA (Obr. 2.3). V dnešní době vyrábí pět řad obvodů FPGA, které jsou přehledně znázorněny v zastoupení svých nejvýkonějších čipů v následující tabulce 2.4. [10]

Features	Artix-7	Kintex-7	Virtex-7	Spartan-6	Virtex-6
Logic Cells	215	480	2,000,000	150	760
BlockRAM	13Mb	34Mb	68Mb	4.8Mb	38Mb
DSP Slices	740	1,92	3,6	180	2,016
DSP Performance (symmetric FIR)	930GMACs	2,845GMACs	5,335GMACs	140GMACs	2,419GMACs
Transceiver Count	16	32	96	8	72
Transceiver Speed	6.6Gb/s	12.5Gb/s	28.05Gb/s	3.2Gb/s	11.18Gb/s
Total Transceiver Bandwidth (full duplex)	211Gb/s	800Gb/s	2,784Gb/s	50Gb/s	536Gb/s
Memory Interface (DDR3)	1,066Mb/s	1,866Mb/s	1,866Mb/s	800Mb/s	1,066Mb/s
PCI Express® Interface	x4 Gen2	x8 Gen2	x8 Gen3	x1 Gen1	x8 Gen2
Analog Mixed Signal (AMS)/XADC	Yes	Yes	Yes		Yes
Configuration AES	Yes	Yes	Yes	Yes	Yes
I/O Pins	500	500	1,2	576	1,2
I/O Voltage	1.2V, 1.35V, 1.5V, 1.8V, 2.5V, 3.3V	1.2V, 1.35V, 1.5V, 1.8V, 2.5V, 3.3V	1.2V, 1.35V, 1.5V, 1.8V, 2.5V, 3.3V	1.2V, 1.5V, 1.8V, 2.5V, 3.3V	1.2V, 1.5V, 1.8V, 2.5V
EasyPath Cost Reduction Solution	-	Yes	Yes	-	Yes

Tabulka 2.4: Přehled vyráběných řad obvodů FPGA firmou Xilinx [10]

### 2.2.2.1 Rodiny FPGA obvodů Xilinx

Xilinx používá také rozdělení svých FPGA do rodin, do kterých se řadí obvody z různých řad zmíněných v předchozí tabulce 5. Zdroj [10]

- **7 Series FPGA Families** – obvody sedmé série  
Artix™-7 FPGAs, Kintex™-7 FPGAs, Virtex®-7 FPGAs, Virtex®-7 FPGAs,  
EasyPath™-7 FPGAs
- **6 Series FPGA Families** – obvody šesté série  
Virtex-6 FPGAs, Spartan®-6 FPGAs, EasyPath-6 FPGAs
- **Automotive Device Families** – zařízení pro automobilový průmysl  
XA Spartan-6 FPGAs, XA Spartan-3A FPGAs, XA Spartan-3A DSP FPGAs,  
XA Spartan-3E FPGAs
- **Defense-grade Device Families** – obvody speciální pro obranné technologie  
Defense-grade Artix-7Q FPGAs, Defense-grade Kintex-7Q FPGAs, Defense-grade  
Virtex-7Q FPGAs, Defense-grade Virtex-6Q FPGAs, Defense-grade Spartan-6Q  
FPGAs Defense-grade Virtex-5Q FPGAs, Defense-grade Virtex-4Q FPGAs
- **Space-grade Device Families** – speciální obvody pro vesmírnou techniku  
Space-grade Virtex-5QV FPGAs, Space-grade Virtex-4QV FPGAs
- **Additional Device Families** – do této rodiny se řadí všechna další FPGA  
Virtex-5 FPGA Family, Virtex-4 FPGA Family, Spartan-3A FPGA Family,  
Spartan-3AN FPGA Family, Spartan-3A DSP Family, Spartan-3E FPGA Family
- **EasyPath FPGA Families** – levná alternativa k ASICs obvodům  
EasyPath FPGAs, EasyPath-6 FPGAs, EasyPath-7 FPGAs



## **2.3 Zvolené FPGA**

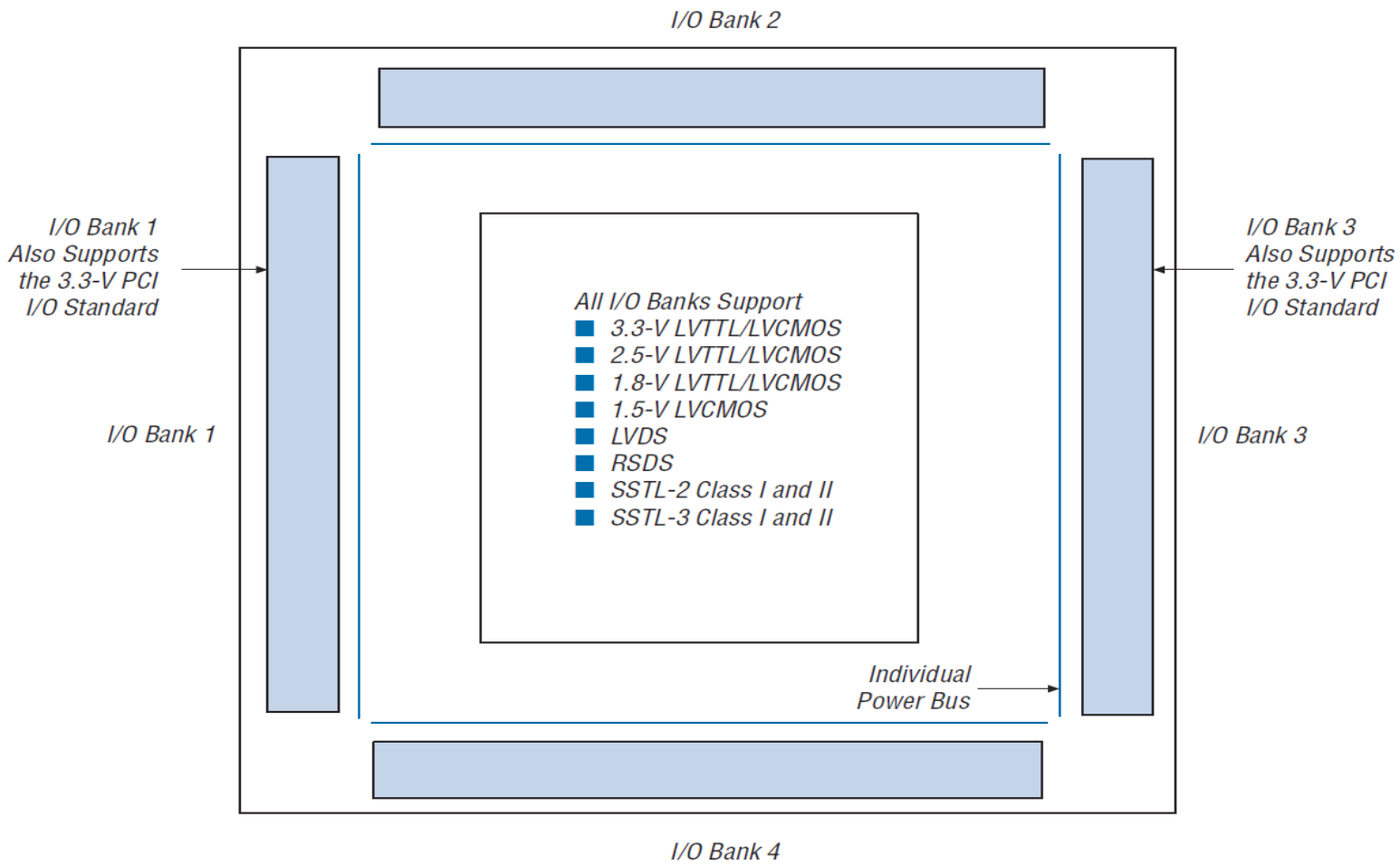
Ke spolupráci s již zmíněným mikrokontrolérem STM32F107 jsem zvolil FPGA značky Altera, konkrétně EP1C3T100C7N. Důvodem pro výběr obvodu právě od této firmy, je velmi dobrá dostupnost jejich výrobků u nás a také příznivá cena. Jedná se o obvod z nejnižší řady Cyclone, který pro potřeby pokusů a seznámení s FPGA obvody bohatě postačí.

### **2.3.1 Specifikace EP1C3T100C7N:**

- 2910 makrobuněk
- 13xM4K RAM Bloků
- 59 Kbits vlastní paměti
- 1xPLL
- max 65 I/O
- 34 diferenčních kanálů
- pracovní teploty -40 až +125 °C
- velikost pozdra TQFP 16x16 mm

### **2.3.2 Multivolt I/O support**

I/O banky mají své vlastní oddělené napájení. Jednotlivé banky tak mohou podporovat různé napěťové standardy s napěťovými hladinami 3,3V, 2,5V, 1,8V a 1,5V (Obr. 20). Každá z bank také disponuje dvojicí pinů pro připojení referenčního napětí výše zmíněných napěťových hladin. Obr. 2.4



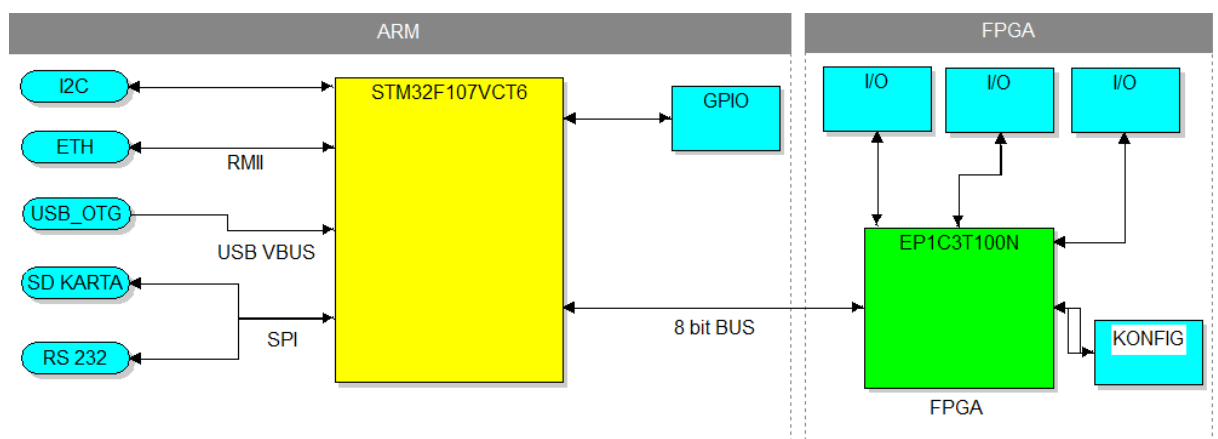
Obr. 2.4: I/O banky série Cyclone [20]

### 3 Návrh univerzální DPS

Desku jsem navrhoval s úmyslem vytvořit co možná největší flexibilitu použitých periférií, tedy pro FPGA obvod. V případě pevně navržených periférií by nebylo možné je jednoduše vyměnit za jiné a tak je většina I/O a funkčních pinů z FPGA vyvedena na příslušné konektory s respektováním bran.

Neměnné spojení je realizováno jen v nejnútnejších případech a sice pro hodinový signál a komunikaci FPGA s příslušnou SRAM pamětí. Pro zajištění možnosti komunikace FPGA obvodu s mikrokontrolérem ARM je mezi nimi vyvedena 8-bitová sběrnice z I/O pinů jak MCU tak FPGA. Piny byly vybrány s ohledem na GPIO, tedy aby bylo pro tuto funkci použito co nejvíce I/O pinů. Aby byly signály co nejbliže u sebe, sběrnici tvoří 6 I/O a 2 GPIO obvodu FPGA.

Zamýšlená funkce mikrokontroléru ARM by měla spočívat v řízení obvodu FPGA a zajištění komunikace s okolím. Proto jsou také využity všechny dostupné podporované periferie zmíněného mikrokontroléru. Na následujícím obrázku Obr.3.1 je znázorněno blokové schéma návrhu DPS.



Obr. 3.1: Blokové schéma zamýšlené návrhu DPS

### **3.1 Možnosti využití DPS**

Navrženou DPS je možno využít především jako cvičnou či vývojovou desku. Díky vyvedeným GPIO FPGA obvodu je možné sestavit si jakoukoliv periférii a neomezovat se tak pouze na periférie DPS. Tím bude umožněno vyzkoušet si a osvojit základy návrhu FPGA.

Další možností jak takovou desku s kombinací ARM a FPGA využít, je pro monitorování a případnou regulaci. Kde k obvodu FPGA může být díky vysokému počtu GPIO připojeno velké množství dalších digitálních zařízení, například senzorů. MCU pak může zajistit zpracování výsledných dat z FPGA a díky navrženým perifériím je dále zprostředkovat svému okolí.

Ideálním využitím je pak v aplikacích, kde je potřeba zpracovávat velké množství digitálních signálů, příkladem takových aplikací je zpracování digitálního audia a videa.

## **4 Vlastní návrh mikroprocesorové desky**

Návrh DPS jsem realizoval za pomoci softwarového nástroje Altium Designer představen bude v následující kapitole. Deska se skládá ze dvou hlavních obvodů ARM FPGA specifikovaných v předchozích kapitolách.

## 4.1 Altium Designer

Na myšlenku využít k návrhu DPS program Altium Designer mne přivedl vedoucí mé práce Ing. Jan Brož. Využil jsem školní licence, přesněji akademické licence ZCU-RICE a začal s prvními pokusy různých návrhů. Ukázka pracovního prostředí programu je na obrázku číslo

Zmíněný software je velmi komplexním návrhovým nástrojem, který obsahuje nástroje jak pro tvorbu elektrických schémat, DPS layout a zobrazení DPS ve 3D, tak i složité a novátorské nástroje pro návrh a programování FPGA obvodů. Jedná se hlavně o nový způsob konfigurování FPGA pomocí openbus, který je ale zároveň provázán s klasickým jazykem pro hardwarové programování VHDL či Verilog. Lze tak jedním kliknutím vygenerovat potřebné příkazy pro základní nastavení jednotlivých funkčních bloků. Dokáže toho však ještě mnohem víc. Umí simulovat navrhované systémy, takže je možné okamžitě otestovat správnost návrhu. Obsahuje i nástroje pro programování jazyků C, C++.

Po několika neúspěšných pokusech vytvořit a provázat FPGA projekt s veškerými náležitostmi jsem z časových důvodů přistoupil k návrhu klasickým způsobem, tedy klasické schéma zapojení. Pro nováčka v oblasti návrhu FPGA se zdá být kombinace všech těchto možností a celé řady možných nastavení a pravidel skoro až na škodu. Díky instruktážním videím dostupných na webových stránkách společnosti Altium [12] jsem byl schopen začít se v systému pomalu lépe orientovat a realizovat tak výsledný návrh DPS.

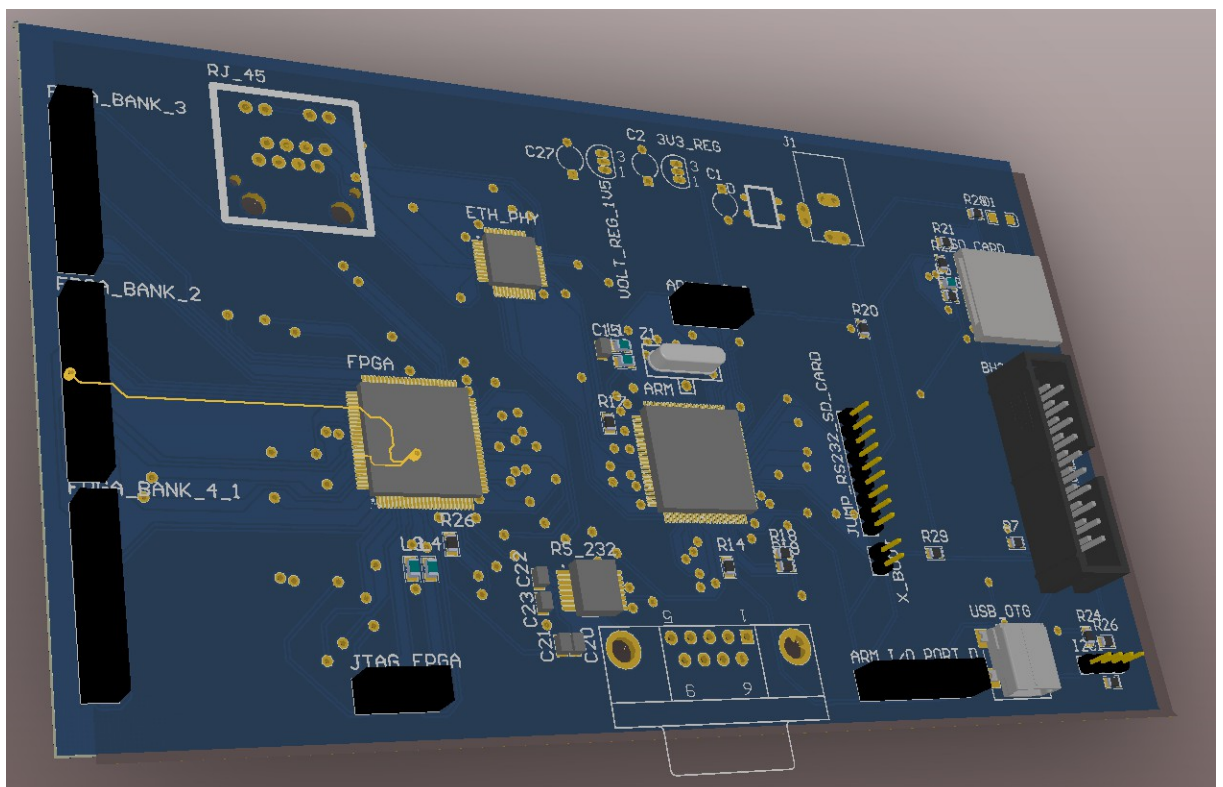
### Altium - obvodové schéma

Při sledování prvních ukázkových videích [12] se zdálo, že bude nakreslení obvodového schématu hračkou. Opak byl pravdou. Ne snad proto, že by byl systém v něčem špatný nebo snad nepřehledný, ale proto, že aktualizace a nové knihovny součástek jsou dostupné pouze po přihlášení k účtu altium live, za který si společnost právem účtuje nemalou sumu pohybující se okolo 700 dolarů ročně. Občas se dá na internetu najít nějakou tu knihovnu, ale většinou nebývají moc užitečné. Jsou to knihovny vytvořené ostatními uživateli a tedy

knihovny, které si takřkajíc ušili na míru. Je tedy jasné, že se při návrhu nevyhnu nutnosti vytvořit si vlastní knihovny a příslušné součástky.

Začít je potřeba s vytvořením takzvaného footprintu, což není nic jiného, než přesné zakreslení pinů či pájecích plošek podle výkresu v datasheetu. Následně je třeba přidat ohraničení součástky a pokud se spokojíme s 2D zobrazením je součástka hotova. Ještě zkontrolovat piny součástky zda souhlasí umístění pinů s umístěním pinů v datasheetu.

Jakmile je součástka hotova, následuje vytvoření schematické značky. Tu je možné si navrhnou podle potřeby. Jedná se jen o obdélník ke kterému vedou kontakty. Ještě jednou pro jistotu zkontrolovat správnost pinů a součástka je hotova. Takto jsem pro svou DPS musel vytvořit konektor RJ\_45, SD kartu a diodový můstek.



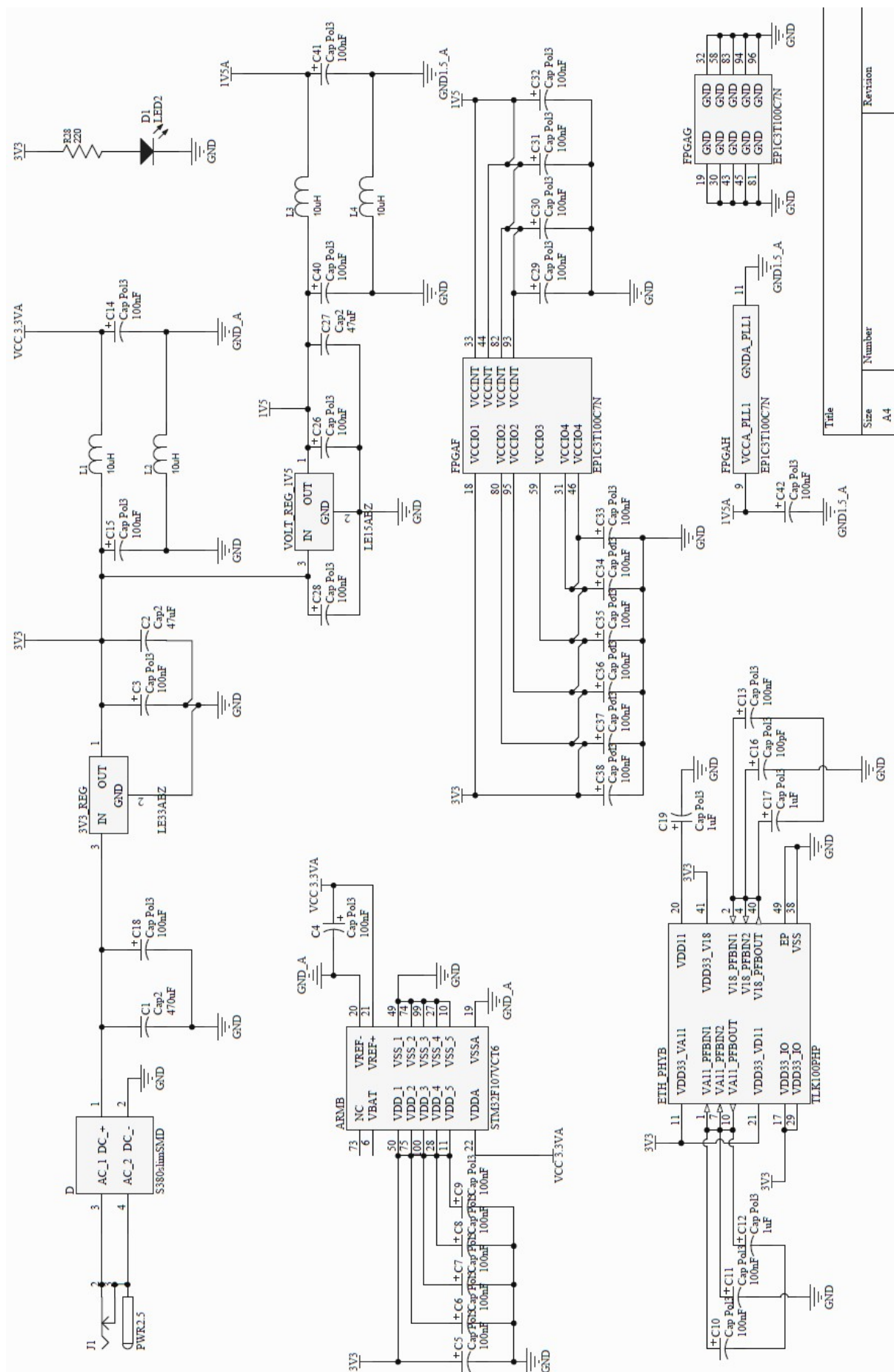
Obr. 4.1 3D zobrazení výsledné DPS:

## **5.1 Napájení DPS**

Navržená deska využívá dvou napájecích hladin 1,5V a 3,3V. Všechny obvody na desce kromě obvodu FPGA (zmíněno v kapitole) podporují 3,3V. K napájení vnitřních obvodů FPGA je využito napětí 1,5V. K dosažení napětí 3,3V byl použit napěťový regulátor LE33ABZ. Z tohoto napětí pak dalším napěťovým měničem LE15ABZ získáno napětí 1,5V.

Do napájecího konektoru navržené desky může být přivedeno napětí z externího zdroje o maximální hodnotě 18V (typicky 12V). Díky použití usměrňovacího můstku S380slimSMD je možné desku napájet jak stejnosměrným, tak i střídavým zdrojem napětí. Zároveň můstek slouží jako ochrana proti přepólování. Výstupy regulátorů i můstku jsou kvůli stabilizaci napětí opatřeny blokovacími kondenzátory. K indikaci napájení desky slouží LED dioda D1 připojená ke 3,3V.

Dále bylo nutné z obou napěťových hladin odfiltrovat napájení pro analogovou část obvodů ARM a FPGA. To je provedeno standartně filtrem. Schéma zmíněného napájení je znázorněno na následujícím obrázku 5.1.

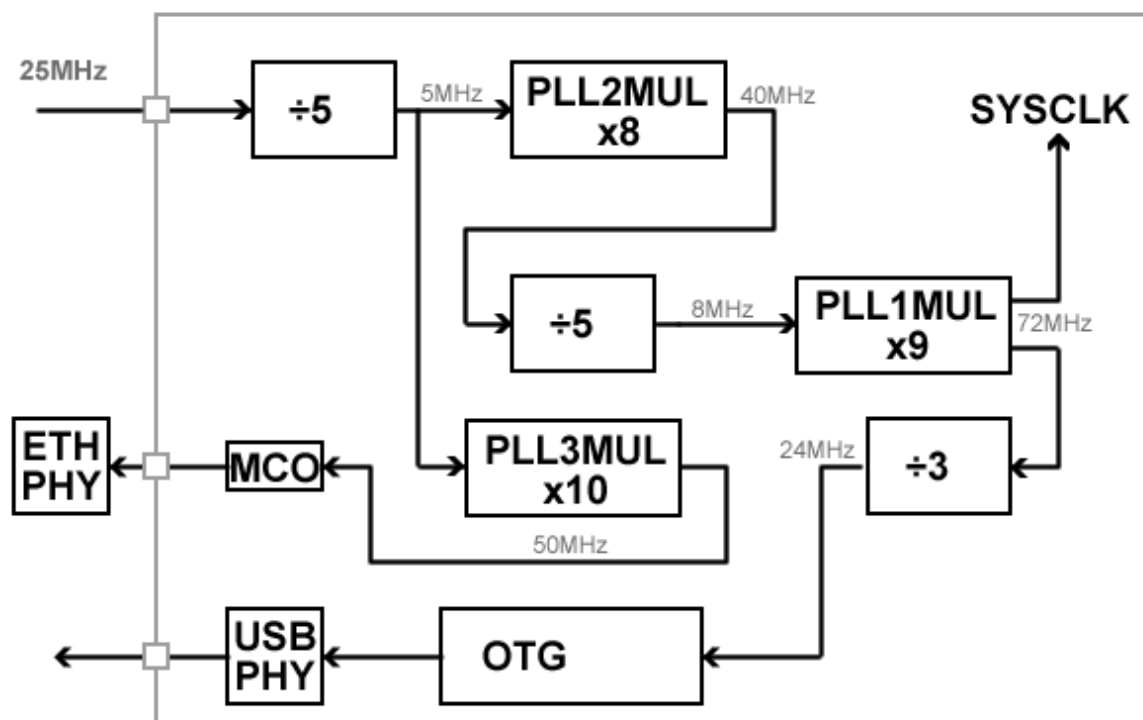


Obr. 5.1: Schéma napájení DPS



## 5.2 Časování

Základem pro hodinový signál je 25MHz krystal, ze kterého jsou pomocí PLL integrovaných v mikrokontroléru řady STM32F107 odvozeny všechny další frekvence potřebné pro obvod FPGA a další periferie. Jak je znázorněno na obrázku XY, je signál nejprve vydělen 5 a v zápětí pomocí PLL3MUL vynásoben 10x na požadovanou hodnotu 50MHz a je vyveden na pin mikrokontroléru MCO. Obvodu FPGA může být na vstup hodinového signálu CLK0 a CLK2 přiveden signál v rozmezí 20-400MHz. Bude tedy použit stejný 50MHz signál, který je použit pro časování phyteru PHY. Obdobným způsobem je možné získat zbývající hodinové signály 72MHz pro mikrokontrolér a 48MHz pro USB OTG.



Obr. 5.2: Generování hodinových signálů mikrokontrolérem

## **5.3 Periferie**

### **5.3.1 8-bitová sběrnice**

Propojení mikrokontroléru s FPGA obvodem je realizováno pomocí 8-bitové sběrnice nazvané ARM\_FPGA\_BUS, pro kterou byla vyhrazena na straně mikrokontroléru polovina portu E a to konkrétně na pinech PE8 až PE15. Na straně FPGA jsem se snažil o co nejmenší zásah do GPIO a tak sběrnice zaujímá 6 obyčejných IO portů a 2 GPIO porty obvodu FPGA. Je to z toho důvodu, aby byly signály vedené po této sběrnici co nejbližší u sebe. Zapojení je na obrázku .

### **5.3.2 Rozhraní JTAG**

Jtag se na desce vyskytuje hned dvakrát. Jeden pro mikrokontrolér a jeden pro FPGA. Tím je zajištěna, v případě potřeby, možnost ladění obou obvodů zároveň.

### **5.3.3 Altera EPCS4SI8N**

Integrovaný obvod určený pro konfiguraci FPGA. Obsahuje paměť typu FLASH 4Mbit. Zapojení pinů je na obrázku XY

### **5.3.4 I2C sběrnice**

Klasická varianta dvoudrátové sběrnice pro komunikaci s jedním či více zařízeními. Sběrnice je vyvedena na 4pinový header na okraji DPS.

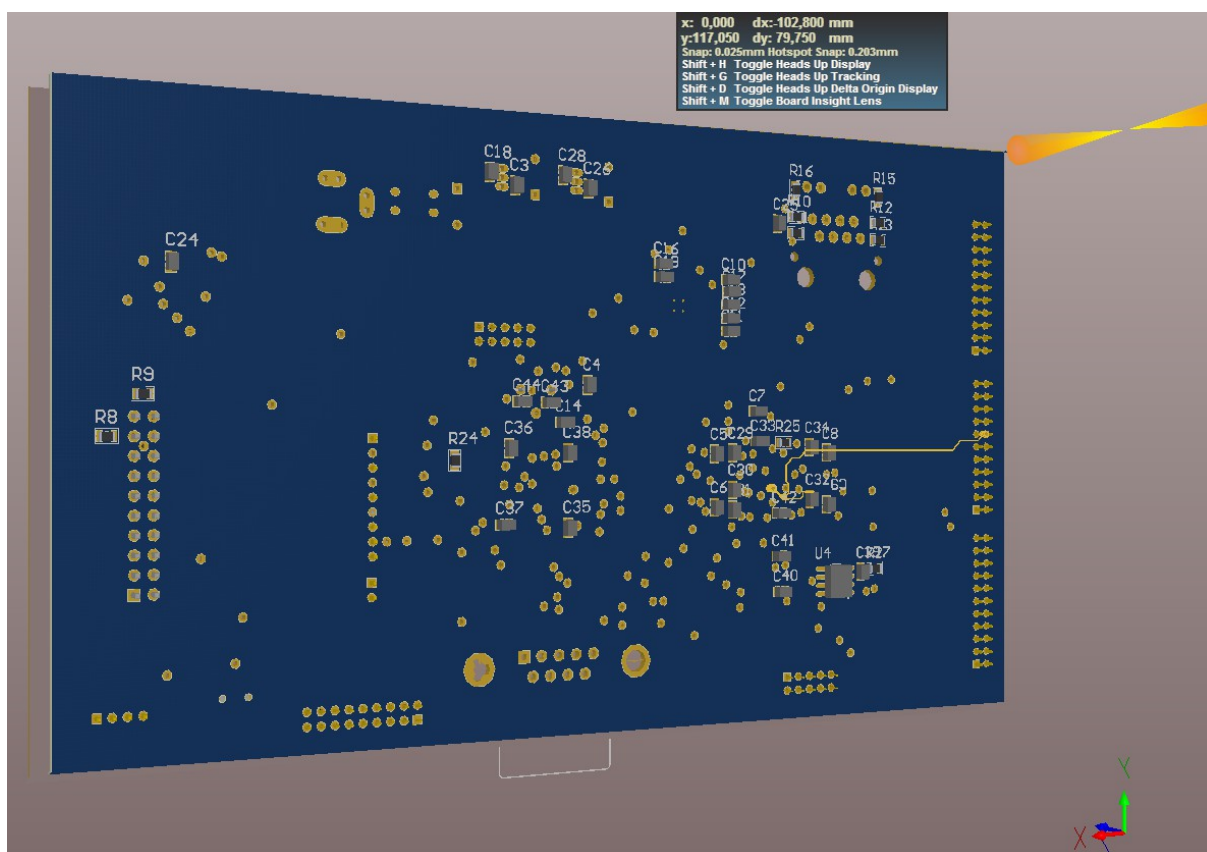
### 5.3.5 USB OTG

Konektor micro USB typu Host. Možnost ppřímého přenosu dat point to point. K napájení je využioto napájecí větve 3,3V.

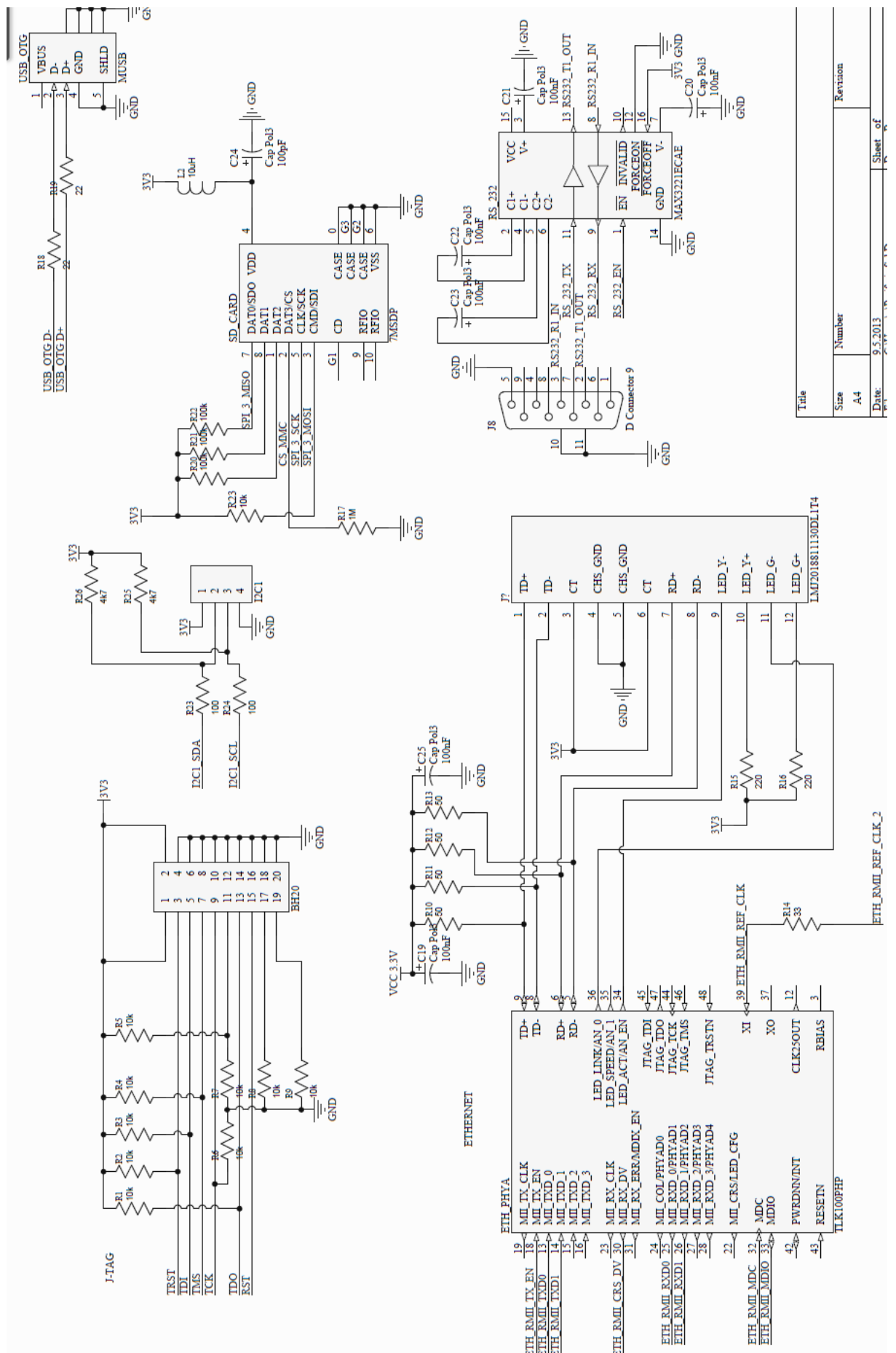
### 5.3.6 RS-232 a SD karta

Protože již na MCU nezbyla vhodná volná periferie, rozhodl jsem se pro tyto dvě periferie využít jedinou volnou sběrnici SPI3. Volba periferie je zajištěna přepojením klasického jumperu.

Na obrázku 5.3 následuje obvodové schéma zapojení jednotlivých periférií.



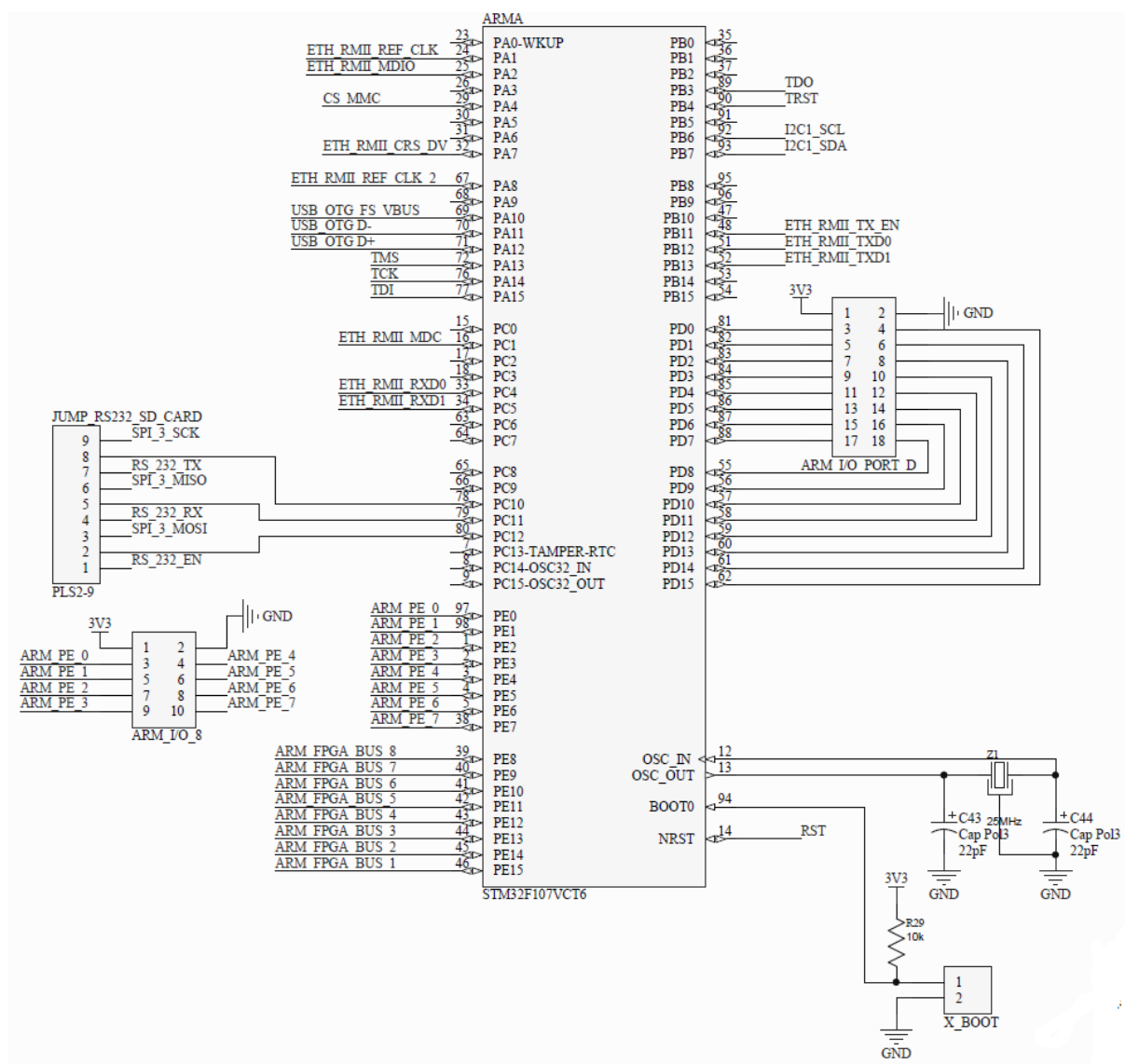
Obr. 5.3: Pohled na zadní stranu DPS ve 3D



Obr. 5.4: Chema zapojení navržených periférií

## 5.4 Schema zapojení MCU ARM

Schéma propojení mikrokontroléru s navrženými periferiemi je na následujícím obrázku 5.3.



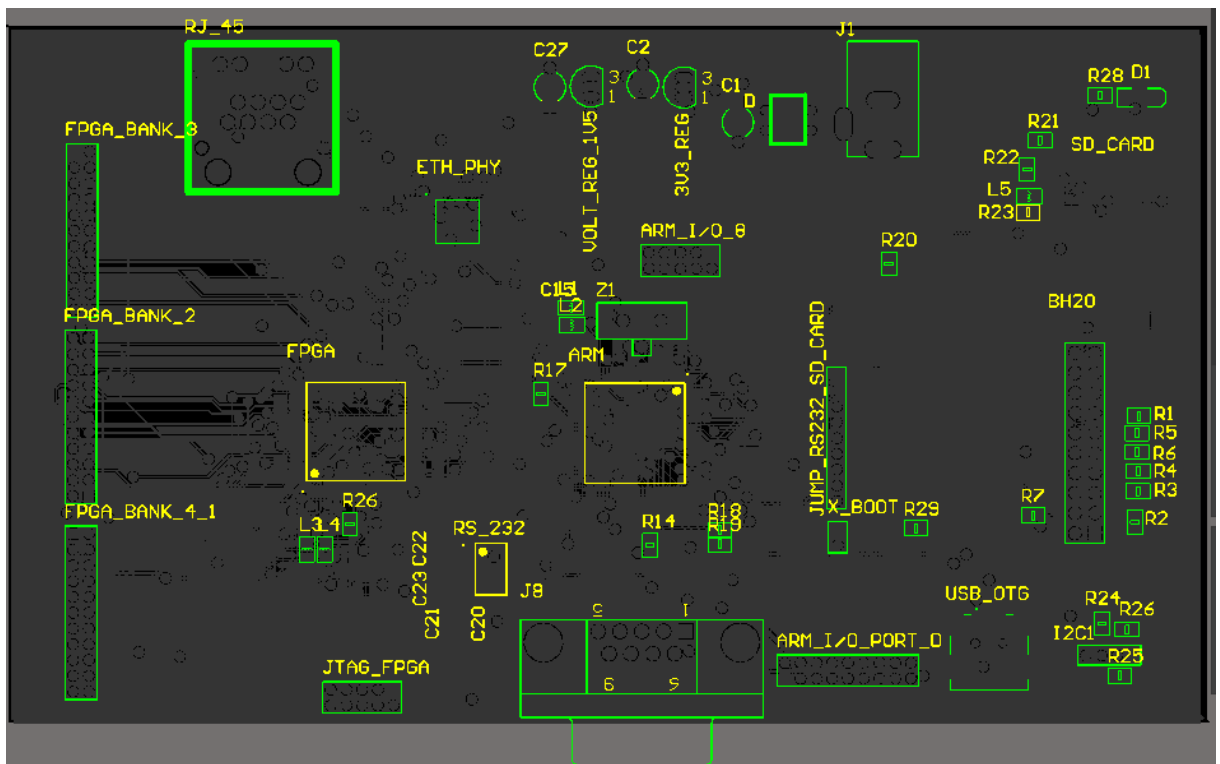
Obr. 5.5:

## 5.5 FPGA

Schéma zapojení FPGA je kvůli velikosti v Příloze 2.

## 6 PCB layout

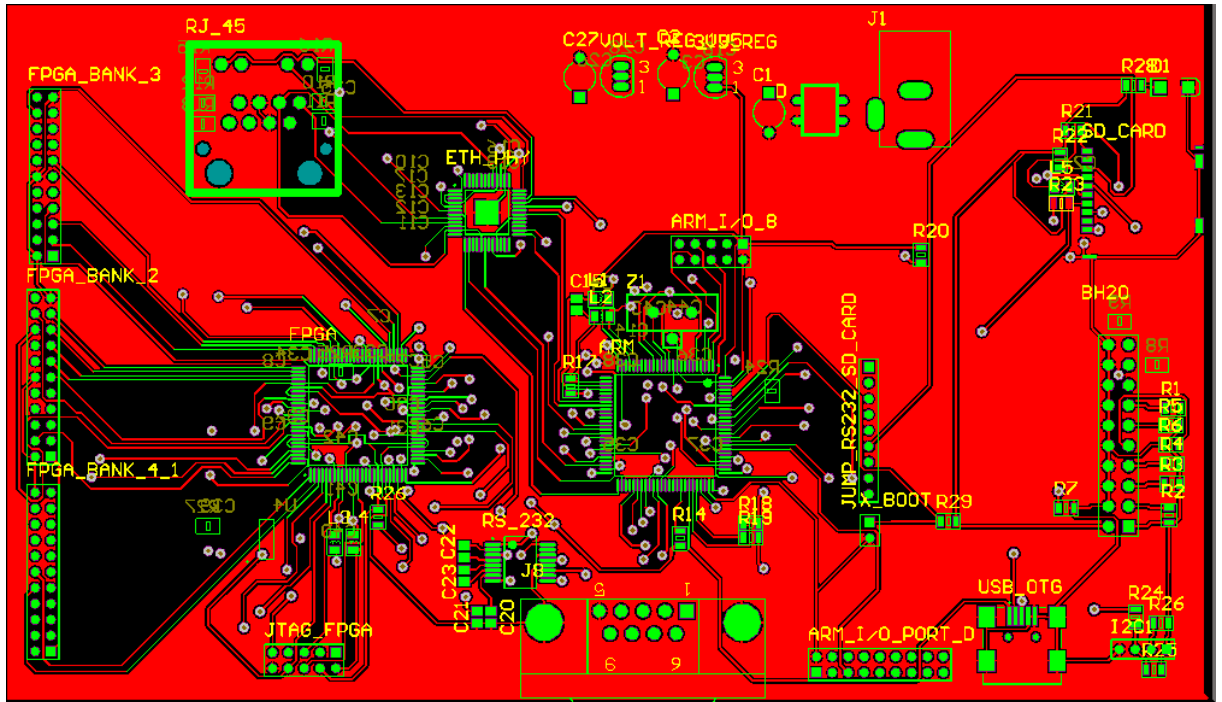
Původně měla být deska navržena jako dvouvrstvá, což se však při vytváření vodivých cest ukázalo jako nemožné kvůli velké hustotě IO pinů jak mikrokontroléru, tak FPGA. Návrh tedy počítá se čtyřvrstvou deskou. Vrchní a spodní deska je pro vylepšení odolnosti rušení vylita mědí. Na následujícím obrázku je znázorněno fyzické rozložení součástek.



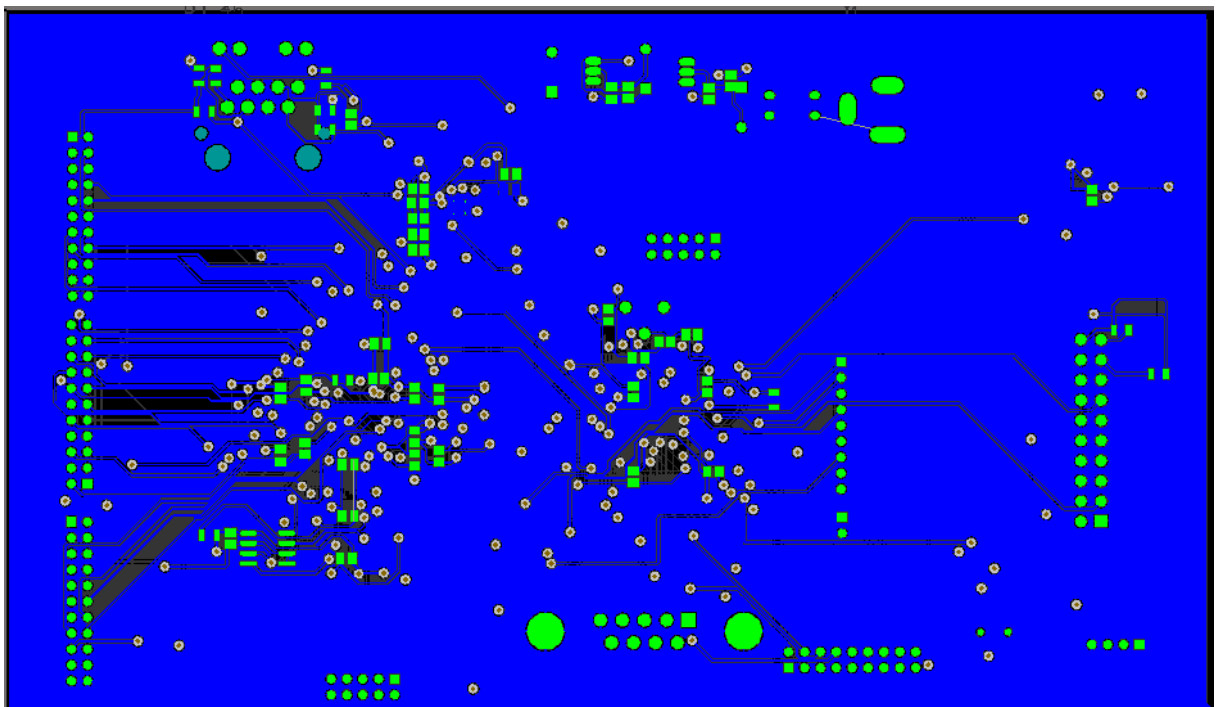
Obr. 6.1: Rozmístění součástek navrhnuté DSP

Altium Designer nabízí při návrhu PCB velmi rozsáhlé možnosti všelijakých nastavení pravidel, možností nastavení. Nechá se v něm například nastavit minimální a maximální vzdálenosti mezi všemi prvky na desce a to včetně pájecí pasty a dalších.

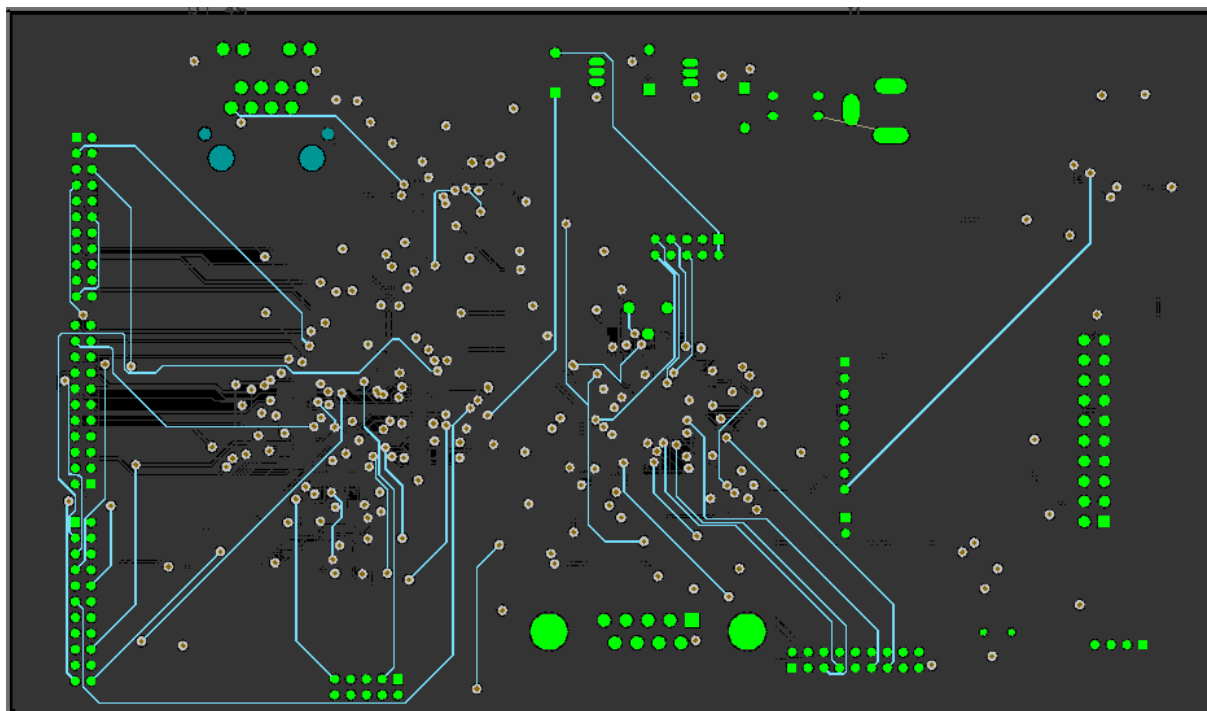
Je možné také nechat vygenerovat velmi podrobnou zprávu s informacemi o navržené DPS, bohužel má mnoho stran a není v žádném formátu, který by se do této práce vešel.



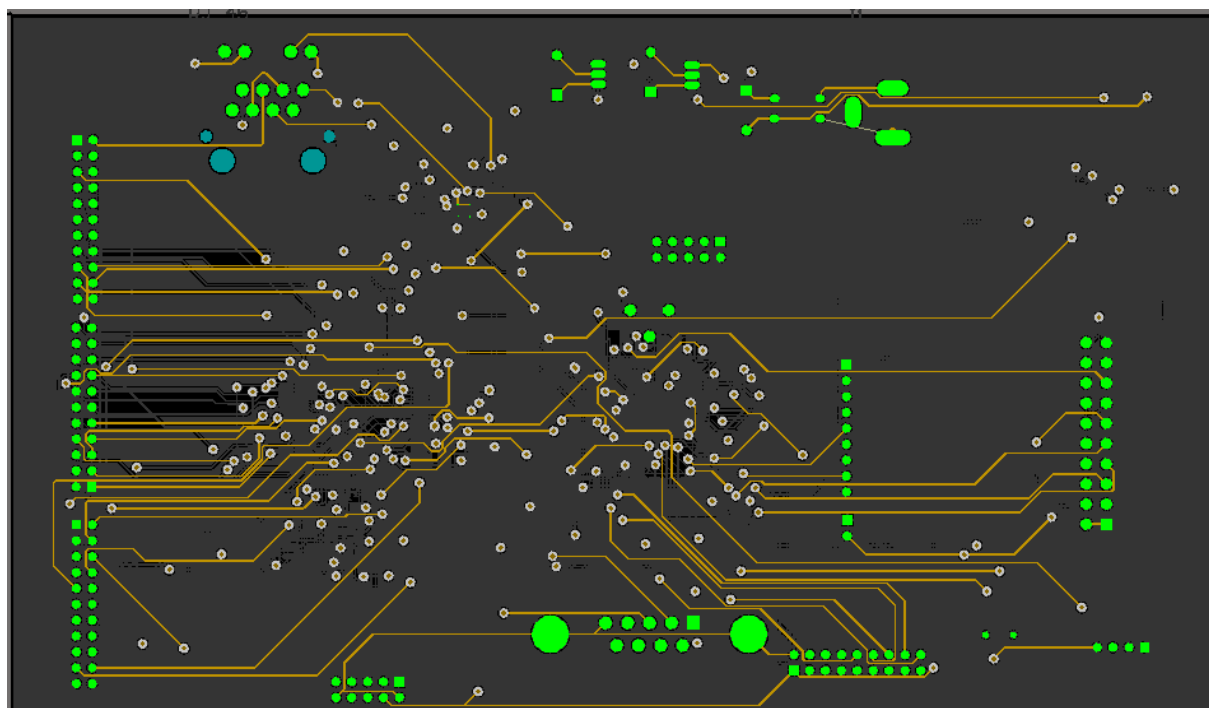
Obr. 6.2: Vrchní strana DPS



Obr. 6.3: spodní strana DPS



Obr. 6.4: Třetí vrstva DPS



Obr. 6.5: Druhá vrstva DPS



## **Závěr**

Záměrem práce bylo přiblížit část moderních integrovaných obvodů, zejména obvodů s architekturou FPGA a snad i ve čtenáři vzbudit o tyto vyjímečné obvody zájem. Obvody FPGA mají před sebou dle mého názoru ohromnou budoucnost. Postupem doby se jak doufám podaří práci s nimi ještě více zjednodušit a zpřístupnit je tak širší veřejnosti.

Mikrokontroléry stále drží s FPGA krok a výrobci stále vyvíjejí nové varianty. Výrobci stále přidávají do čipů stále větší počet periférií, ale také samotných jader procesoru ARM.

Návrhový software Altium Designer je velmi povedený a sofistikovaný software pro navrhování, programování a také ladění obvodů nejrůznějších typů i funkcí.

Musím vyzdvihnout prostředí pro kreslení vodivých cest při návrhu PCB. To je opravdu povedené, ikdyž často se objeví nějaké chybové hlášení a při malé praxi s tímto dosti složitým softwarem trvá velmi dlouho, než se najde, jestli vůbec, řešení.

3D výstup je příjemná změna v návrzích plošných spojů a překvapivě se v něm i dobře pracuje, ikdyž samozřejmě pro nakreslení vodivých cestiček je nutné být v režimu 2D.

Do budoucna bych chtěl návrh DPS v mnohém vylepšit. Například s odstupem času se mi zdá řešení přepínání periférií zvláť za pomoci tří jumperů poněkud nešťastné, využití mikropřepínače by bylo jistě vhodnější. Také čistota návrhu by mohla být lepší, což přisuzuji nezkušenosti s daným softwarem.

## Použitá literatura

- [1] YIU, Joseph. *The definitive guide to the ARM Cortex-M3: second edition*. 2nd ed. Amsterdam: Newnes, 2010, xxii, 457 s. ISBN 978-1-85617-963-8. Dostupné z: <http://hasebsohail.wordpress.com/>
- [2] SLOSS, Andrew N, Dominic SYMES a Chris WRIGHT. *ARM system developer's guide: designing and optimizing system software*. Amsterdam: Elsevier, 2004, 689 s. ISBN 15-586-0874-5.
- [3] Úvod do architektury Cortex-M3. 2010. Dostupné z: [www.Pandatron.cz](http://www.Pandatron.cz)
- [4] *ARM architecture reference manual*. 2005. Dostupné z: [https://www.scss.tcd.ie/~waldroj/3d1/arm\\_arm.pdf](https://www.scss.tcd.ie/~waldroj/3d1/arm_arm.pdf)
- [5] *ARM information center* [online]. 2012 [cit. 2013-05-07]. Dostupné z: <http://infocenter.arm.com/>
- [6] *Cortex M3 Processor - ARM* [online]. 2013 [cit. 2013-05-07]. Dostupné z: <http://www.arm.com/products/processors/cortex-m/cortex-m3.php>
- [7] FPGA, CPLD and ASIC from Altera. [online]. [cit. 2013-05-08]. Dostupné z: <http://www.altera.com/>
- [8] PECH, Jan. Nebojte se FPGA, HW CZ. In: *HW CZ* [online]. [cit. 2013-05-09]. Dostupné z: <http://www.hw.cz/teorie-a-praxe/dokumentace/nebojte-se-fpga.html>
- [9] PINKER, Jiří a Martin POUPA. *Číslicové systémy a jazyk VHDL*. 1. vyd. Praha: BEN - technická literatura, 2006, 349 s. ISBN 80-730-0198-5.
- [10] All Programmable Technologies from Xilinx Inc. [online]. [cit. 2013-05-09]. Dostupné z: <http://www.xilinx.com/>
- [11] [online]. [cit. 2013-05-09]. Dostupné z: <http://www.st.com/web/en/catalog/tools/PF251717>
- [12] ALTIUM VIDEO
- [13] ŠŤASTNÝ, Jakub. *FPGA prakticky: realizace číslicových systémů pro programovatelná hradlová pole*. 1. vyd. Praha: BEN - technická literatura, 2010, 199 s. ISBN 978-80-7300-261-9.
- [14] VÁŇA, Vladimír. *ARM pro začátečníky*. Praha: BEN - technická literatura, 2009, 195 s. ISBN 978-80-7300-246-6.
- [15] SLOSS, Andrew, Dominik SYMES a Chris WRIGHT. *ARM System Developer's Guide: Designing and Optimizing System Software*.

## Technické manuály

- [20] ST MICROELECTRONICS. *STM32F107xx: STM32F105xx*. 2011. Rev 6. Dostupné z: [www.st.com](http://www.st.com)
- [21] *Serial Configuration (EPCS) Devices Datasheet*. Dostupné z: [http://www.altera.com/literature/hb/cfg/cyc\\_c51014.pdf](http://www.altera.com/literature/hb/cfg/cyc_c51014.pdf)
- [22] *Industrial Temp, Single Port 10/100 Mb/s Ethernet Physical Layer Transceiver*. Dostupné z: <http://www.ti.com/lit/ds/slls931b/slls931b.pdf>

## Příloha 1 A – přehled procesorů ARM

ARM Family	ARM Architecture	ARM Core	Feature	Cache (I/D), MMU	Typical MIPS @ MHz
ARM1	ARMv1	ARM1	First implementation	None	
ARM2	ARMv2	ARM2	ARMv2 added the MUL (multiply) instruction	None	4 MIPS @ 8 MHz 0.33 DMIPS/MHz
	ARMv2a	ARM250	Integrated MEMC (MMU), Graphics and IO processor. ARMv2a added the SWP and SWPB (swap) instructions.	None, MEMC1a	7 MIPS @ 12 MHz
ARM3	ARMv2a	ARM3	First integrated memory cache.	4 KB unified	12 MIPS @ 25 MHz 0.50 DMIPS/MHz
ARM6	ARMv3	ARM60	ARMv3 first to support 32-bit memory address space (previously 26-bit)	None	10 MIPS @ 12 MHz
		ARM600	As ARM60, cache and coprocessor bus (for FPA10 floating-point unit).	4 KB unified	28 MIPS @ 33 MHz
		ARM610	As ARM60, cache, no coprocessor bus.	4 KB unified	17 MIPS @ 20 MHz 0.65 DMIPS/MHz
ARM7	ARMv3	ARM700		8 KB unified	40 MHz
		ARM710	As ARM700, no coprocessor bus.	8 KB unified	40 MHz
		ARM710a	As ARM710	8 KB unified	40 MHz 0.68 DMIPS/MHz
ARM7TDMI	ARMv4T	ARM7TDMI(-S)	3-stage pipeline, Thumb	none	15 MIPS @ 16.8 MHz 63 DMIPS @ 70 MHz
		ARM710T	As ARM7TDMI, cache	8 KB unified, MMU	36 MIPS @ 40 MHz
		ARM720T	As ARM7TDMI, cache	8 KB unified, MMU with Fast Context Switch Extension	60 MIPS @ 59.8 MHz
		ARM740T	As ARM7TDMI, cache	MPU	
ARM7EJ	ARMv5TEJ	ARM7EJ-S	5-stage pipeline, Thumb, Jazelle DBX, Enhanced DSP instructions	none	
ARM8	ARMv4	ARM810[4][5]	5-stage pipeline, static branch prediction, double-bandwidth memory	8 KB unified, MMU	84 MIPS @ 72 MHz 1.16 DMIPS/MHz
ARM9TDMI	ARMv4T	ARM9TDMI	5-stage pipeline, Thumb	none	
		ARM920T	As ARM9TDMI, cache	16 KB/16 KB, MMU with FCSE (Fast Context Switch Extension)[6]	200 MIPS @ 180 MHz
		ARM922T	As ARM9TDMI, caches	8 KB/8 KB, MMU	
		ARM940T	As ARM9TDMI, caches	4 KB/4 KB, MPU	
ARM9E	ARMv5TE	ARM946E-S	Thumb, Enhanced DSP instructions, caches	variable, tightly coupled memories, MPU	
		ARM966E-S	Thumb, Enhanced DSP instructions	no cache, TCMS	
		ARM968E-S	As ARM966E-S	no cache, TCMS	
	ARMv5TEJ	ARM926EJ-S	Thumb, Jazelle DBX, Enhanced DSP instructions	variable, TCMS, MMU	220 MIPS @ 200 MHz
	ARMv5TE	ARM996HS	Clockless processor, as ARM966E-S	no caches, TCMS, MPU	
ARM10E	ARMv5TE	ARM1020E	6-stage pipeline, Thumb, Enhanced DSP instructions, (VFP)	32 KB/32 KB, MMU	
		ARM1022E	As ARM1020E	16 KB/16 KB, MMU	
	ARMv5TEJ	ARM1026EJ-S	Thumb, Jazelle DBX, Enhanced DSP instructions, (VFP)	variable, MMU or MPU	
ARM11	ARMv6	ARM1136J(F)-S[7]	8-stage pipeline, SIMD, Thumb, Jazelle DBX, (VFP), Enhanced DSP instructions	variable, MMU	740 @ 532–665 MHz (i.MX31 SoC), 400–528 MHz
	ARMv6T2	ARM1156T2(F)-S	8-stage pipeline, SIMD, Thumb-2, (VFP), Enhanced DSP instructions	variable, MPU	
	ARMv6Z	ARM1176JZ(F)-S	As ARM1136EJ(F)-S	variable, MMU + TrustZone	965 DMIPS @ 772 MHz, up to 2 600 DMIPS with four processors[8]
	ARMv6K	ARM11 MPCore	As ARM1136EJ(F)-S, 1–4 core SMP	variable, MMU	
SecureCore	ARMv6-M	SC000			0.9 DMIPS/MHz
	ARMv4T	SC100			
	ARMv7-M	SC300			1.25 DMIPS/MHz

## Příloha 1 B – přehled procesorů ARM pokračování

Cortex-M	ARMv6-M	Cortex-M0 [9]	Microcontroller profile, Thumb + Thumb-2 subset (BL, MRS, MSR, ISB, DSB, DMB),[10] hardware multiply instruction (optional small), optional system timer, optional bit-banding memory	No cache, No TCM, No MPU	0.84 DMIPS/MHz
		Cortex-M0+ [11]	Microcontroller profile, Thumb + Thumb-2 subset (BL, MRS, MSR, ISB, DSB, DMB),[10] hardware multiply instruction (optional small), optional system timer, optional bit-banding memory	No cache, No TCM, optional MPU with 8 regions	0.93 DMIPS/MHz
		Cortex-M1 [12]	Microcontroller profile, Thumb + Thumb-2 subset (BL, MRS, MSR, ISB, DSB, DMB),[10] hardware multiply instruction (optional small), OS option adds SVC / banked stack pointer, optional system timer, no bit-banding memory	No cache, 0-1024 KB I-TCM, 0-1024 KB D-TCM, No MPU	136 DMIPS @ 170 MHz,[13] (0.8 DMIPS/MHz FPGA-dependent)[14]
	ARMv7-M	Cortex-M3 [15]	Microcontroller profile, Thumb / Thumb-2, hardware multiply and divide instructions, optional bit-banding memory	No cache, No TCM, optional MPU with 8 regions	1.25 DMIPS/MHz
	ARMv7E-M	Cortex-M4 [16]	Microcontroller profile, Thumb / Thumb-2 / DSP / optional FPU single-precision FPU, hardware multiply and divide instructions, optional bit-banding memory	No cache, No TCM, optional MPU with 8 regions	1.25 DMIPS/MHz
Cortex-R	ARMv7-R	Cortex-R4 [17]	Real-time profile, Thumb / Thumb-2 / DSP / optional VFPv3 FPU, hardware multiply and optional divide instructions, optional parity & ECC for internal buses / cache / TCM, 8-stage pipeline dual-core running lockstep with fault logic	0-64 KB / 0-64 KB, 0-2 of 0-8 MB TCM, opt MPU with 8/12 regions	
		Cortex-R5 (MPCore) [18]	Real-time profile, Thumb / Thumb-2 / DSP / optional VFPv3 FPU and precision, hardware multiply and optional divide instructions, optional parity & ECC for internal buses / cache / TCM, 8-stage pipeline dual-core running lock-step with fault logic / optional as 2 independent cores, low-latency peripheral port (LLPP), accelerator coherency port (ACP) [19]	0-64 KB / 0-64 KB, 0-2 of 0-8 MB TCM, opt MPU with 12/16 regions	
		Cortex-R7 (MPCore) [20]	Real-time profile, Thumb / Thumb-2 / DSP / optional VFPv3 FPU and precision, hardware multiply and optional divide instructions, optional parity & ECC for internal buses / cache / TCM, 11-stage pipeline dual-core running lock-step with fault logic / out-of-order execution / dynamic register renaming / optional as 2 independent cores, low-latency peripheral port (LLPP), ACP [19]	0-64 KB / 0-64 KB, ? of 0-128 KB TCM, opt MPU with 16 regions	
Cortex-A	ARMv7-A	Cortex-A5 [21]	Application profile, ARM / Thumb / Thumb-2 / DSP / SIMD / Optional VFPv4-D16 FPU / Optional NEON / Jazelle RCT and DBX, 1-4 cores / optional MPCore, snoop control unit (SCU), generic interrupt controller (GIC), accelerator coherence port (ACP)	4-64 KB / 4-64 KB L1, MMU + TrustZone	1.57 DMIPS / MHz per core
		Cortex-A7 MPCore [22]	Application profile, ARM / Thumb / Thumb-2 / DSP / VFPv4-D16 FPU / NEON / Jazelle RCT and DBX / Hardware virtualization, in-order execution, superscalar, 1-4 SMP cores, Large Physical Address Extensions (LPAE), snoop control unit (SCU), generic interrupt controller (GIC), ACP, architecture and feature set are identical to A15, 8-10 stage pipeline, low-power design[23]	32 KB / 32 KB L1, 0-4 MB L2, L1 & L2 have Parity & ECC, MMU + TrustZone	1.9 DMIPS / MHz per core
		Cortex-A8 [24]	Application profile, ARM / Thumb / Thumb-2 / VFPv3 FPU / Optional NEON / Jazelle RCT and DAC, 13-stage superscalar pipeline	16-32 KB / 16-32 KB L1, 0-1 MB L2 opt ECC, MMU + TrustZone	up to 2000 (2.0 DMIPS/MHz in speed from 600 MHz to greater than 1 GHz)
		Cortex-A9 MPCore [25]	Application profile, ARM / Thumb / Thumb-2 / DSP / Optional VFPv3 FPU / Optional NEON / Jazelle RCT and DBX, out-of-order speculative issue superscalar, 1-4 SMP cores, snoop control unit (SCU), generic interrupt controller (GIC), accelerator coherence port (ACP)	16-64 KB / 16-64 KB L1, 0-8 MB L2 opt Parity, MMU + TrustZone	2.5 DMIPS/MHz per core, 10,000 DMIPS @ 2 GHz on Performance Optimized TSMC 40G (dual core)
		Cortex-A15 MPCore [26]	Application profile, ARM / Thumb / Thumb-2 / DSP / VFPv4 FPU / NEON / Jazelle RCT / Hardware virtualization, out-of-order speculative issue superscalar, 1-4 SMP cores, Large Physical Address Extensions (LPAE), snoop control unit (SCU), generic interrupt controller (GIC), ACP, 15-24 stage pipeline[23]	32 KB / 32 KB L1, 0-4 MB L2, L1 & L2 have Parity & ECC, MMU + TrustZone	At least 3.5 DMIPS/MHz per core (Up to 4.01 DMIPS/MHz depending on implementation).[27]
Cortex-A50	ARMv8-A	Cortex-A53[28]	Application profile, AArch32 and AArch64, 1-4 SMP cores, Trustzone, NEON advanced SIMD, VFPv4, hardware virtualization, dual issue, in-order pipeline	8-64 KB/8-64 KB L1 per core, 128 KB-2 MB L2 shared, 40-bit physical addresses	2.3 DMIPS/MHz
		Cortex-A57[29]	Application profile, AArch32 and AArch64, 1-4 SMP cores, Trustzone, NEON advanced SIMD, VFPv4, hardware virtualization, multi-issue, deeply out-of-order pipeline	48 KB/32 KB L1 per core, 512 KB-2 MB L2 shared, 44-bit physical addresses	At least 4.1 DMIPS/MHz per core (Up to 4.76 DMIPS/MHz depending on implementation).
ARM Family	ARM Architecture	ARM Core	Feature	Cache (I/D), MMU	Typical MIPS @ MHz

