



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV MIKROELEKTRONIKY

DEPARTMENT OF MICROELECTRONICS

NÁVRH VÝVOJOVÉHO KITU S OBVODEM FPGA

DESIGN OF FPGA DEVELOPMENT KIT

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Petr Zach

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Lukáš Fucik, Ph.D.

BRNO 2021



Bakalářská práce

bakalářský studijní program **Mikroelektronika a technologie**

Ústav mikroelektroniky

Student: Petr Zach

ID: 211249

Ročník: 3

Akademický rok: 2020/21

NÁZEV TÉMATU:

Návrh vývojového kitu s obvodem FPGA

POKYNY PRO VYPRACOVÁNÍ:

Prozkoumejte možnosti návrhu vývojového kitu s potřebnými periferiemi, jako jsou klasické IO konektory, vysokorychlostní konektory, stereo MEMs mikrofony, DDR3 a další. Zaměřte se na návrh desky plošných spojů vývojového kitu. V rámci bakalářské práce otestujte funkčnost kitu.

DOPORUČENÁ LITERATURA:

Podle pokynů vedoucího práce.

Termín zadání: 8.2.2021

Termín odevzdání: 3.6.2021

Vedoucí práce: doc. Ing. Lukáš Fucík, Ph.D.

doc. Ing. Jiří Háze, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato bakalářská práce se zabývá návrhem vývojového kitu s FPGA obvodem pro vývoj a implementaci logických obvodů. První kapitola popisuje různé typy programovatelných logických obvodů a srovnává jejich výhody a nevýhody. Také popisuje nejznámější způsoby konfigurace programovatelných logických obvodů a porovnává je mezi sebou. Druhá kapitola podrobně rozebírá architekturu Xilinx FPGA obvodu 7. řady. Třetí kapitola popisuje návrh vývojového kitu a programovacího zařízení. Také jsou zde shrnuty dostupné prostředky použitého FPGA obvodu, podrobně rozebrány periferie, které se na vývojovém kitu nachází a porovnání s vývojovými kity dostupnými na trhu. Poslední kapitola nastiňuje problematiku výroby prototypu vývojového kitu a jeho testování.

KLÍČOVÁ SLOVA

Xilinx FPGA obvod, vývojový kit, PLD, architektura FPGA obvodu, programovací technologie, digitální obvody

ABSTRACT

This bachelor thesis focuses on the design of a FPGA development kit for the development and implementation of logic circuits. The first chapter describes the various types of programmable logic devices and compares their advantages and disadvantages. Furthermore, the first chapter also describes the most common configurations of programmable logic devices and compares them to each other. The second chapter provides an in-depth description of the 7 series Xilinx FPGA architecture. The third chapter describes the design of the development kit and the programming device. Furthermore, the third chapter summarizes the available resources of the used FPGA as well as the devices and ports that are part of the development kit, and compares the development kit to other development kits available on the market. The final chapter outlines the process of producing and testing a development kit prototype.

KEYWORDS

Xilinx FPGA, development kit, PLD, FPGA architecture, programmable technologies, digital circuits

ZACH, Petr. *Návrh vývojového kitu s obvodem FPGA*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav mikroelektroniky, 2021, 87 s. Bakalářská práce. Vedoucí práce: doc. Ing. Lukáš Fucik, Ph.D.

Prohlášení autora o původnosti díla

Jméno a příjmení autora: Petr Zach
VUT ID autora: 211249
Typ práce: Bakalářská práce
Akademický rok: 2020/21
Téma závěrečné práce: Návrh vývojového kitu s obvodem FPGA

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu doc. Ing. Lukášovi Fujcikovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci. Poděkování také patří panu Ing. Alexandru Otáhalovi, Ph.D. za pomoc při konstrukci prototypu zařízení a Ing. Jaroslavu Dohnalovi za pomoc při návrhu zařízení a předání cenných rad a zkušeností. V neposlední řadě chci poděkovat také rodině za podporu během celé doby studia.

Obsah

Úvod	11
1 Programovatelné logické obvody	12
1.1 SPLD obvody	12
1.2 CPLD obvody	16
1.3 FPGA obvody	17
1.4 Programovací technologie	19
1.4.1 SRAM technologie	19
1.4.2 Flash/EEPROM technologie	20
1.4.3 Anti-fuse technologie	21
2 Architektura FPGA	23
2.1 Programovatelná propojovací struktura (PI)	23
2.1.1 Globální dvourozměrná propojovací architektura	24
2.1.2 Lokální dvourozměrná propojovací architektura	24
2.2 Distribuce hodinového signálu	26
2.3 Vstupně-výstupní buňky (IOBs)	27
2.4 Logické buňky (CLBs)	30
2.4.1 Náhledová tabulka (LUT)	32
2.4.2 Klopný obvod	34
2.4.3 Carry logika	35
2.5 Blokovaná RAM	36
2.6 DSP bloky	36
3 Návrh vývojového kitu	38
3.1 Vývojový kit	38
3.1.1 Napájení	38
3.1.2 Propojovací komponenty	41
3.1.3 Paměťové a výpočetní komponenty	42
3.1.4 Audio a video komponenty	43
3.1.5 Ostatní komponenty	44
3.1.6 Parametry DPS a cena vývojového kitu	45
3.2 Programovací zařízení	46
4 Osazení a oživení vývojového kitu	48
4.1 Volba pájecího procesu	48
4.1.1 Nanesení pájecí pasty	48
4.1.2 Osazení komponent	50

4.1.3 Pájení	50
4.2 Oživení a testování vývojového kitu	51
Závěr	53
Literatura	54
Seznam symbolů a zkratk	56
Seznam příloh	59
A Soupis komponent	60
B Vývojový kit	65
C Programovací zařízení	82

Seznam obrázků

1.1	Struktura PROM [3]	13
1.2	Struktura PAL [3]	14
1.3	Struktura PLA [3]	14
1.4	Implementace součinných termů funkce [6]	15
1.5	Porovnání základní buňky FPGA a CPLD obvodů [4]	16
1.6	Implementace logické funkce pomocí multiplexorů [6]	17
1.7	Implementace logické funkce pomocí LUT [6]	18
1.8	Způsoby využití SRAM buněk [5]	20
2.1	Základní bloky FPGA obvodu [5]	23
2.2	Lokální dvourozměrná architektura [5]	25
2.3	Příklad rozložení různě dlouhých segmentů [5]	26
2.4	Zjednodušené schéma vstupně-výstupní buňky FPGA obvodu [4]	28
2.5	Impedanční přizpůsobení cest pomocí DCI [11]	29
2.6	Schéma řezu SLICEM [8]	31
2.7	ASMBL architektura [8]	32
2.8	Dvě možné konfigurace KO v řezu [8]	35
2.9	Struktura bloku DSP48E1 [13]	37
3.1	Blokový diagram vývojového kitu	39
3.2	Příklad zvolených vstupně-výstupních konektorů	42
3.3	Příklad zapojení R2R DA převodníku	44
3.4	3D model vývojového kitu	46
3.5	3D model programovacího zařízení	47
4.1	Šablona a pájecí pasta	49
4.2	Pájecí proces: a) natištění pájecí pasty, b) osazení komponent, c) přetavení pájecí pasty	50
4.3	Fotografie testovacích přípravků: a) napájecí zdroje, b) audio DA převodník	51
4.4	Fotografie funkčního prototypu vývojového kitu	52
B.1	Pohled na všechny vrstvy vývojového kitu	66
B.2	Pohled na vrstvu TOP vývojového kitu	66
B.3	Pohled na vrstvu IN1 vývojového kitu	67
B.4	Pohled na vrstvu IN2 vývojového kitu	67
B.5	Pohled na vrstvu IN3 vývojového kitu	68
B.6	Pohled na vrstvu IN4 vývojového kitu	68
B.7	Pohled na vrstvu BOT vývojového kitu	69
C.1	Pohled na všechny vrstvy plošného spoje	83
C.2	Pohled na vrstvu TOP plošného spoje	83

C.3	Pohled na vrstvu IN1 plošného spoje	84
C.4	Pohled na vrstvu IN2 plošného spoje	84
C.5	Pohled na vrstvu BOTTOM plošného spoje	85

Úvod

Vývoj a implementace digitálních obvodů se v současnosti prosazuje stále více. Právě digitální systémy jsou upřednostňovány před analogovými systémy, a to z několika důvodů. Návrh digitálních obvodů oproti analogovým je zpravidla jednodušší a časově méně náročný. Významnou výhodou je také určitá flexibilita digitálních obvodů a jejich opakovatelná rekonfigurace.

V minulosti se pro implementaci logických obvodů na trhu nacházely pouze jednoduché PLD (Programmable Logic Devices) obvody a nebo ASIC (Application Specific Integrated Circuit) obvody. Jednoduché programovatelné logické obvody jsou pro své parametry mnohdy nedostačující a naopak výroba ASIC obvodů je z časových i finančních hledisek velmi náročná. Mezeru mezi těmito dvěma krajními stavy vyplnily až FPGA obvody. Ty poskytují potřebné parametry, jako je flexibilita, pracovní rychlost nebo například opakovatelná rekonfigurovatelnost přímo v systému, a to za příznivou cenu.

Tato bakalářská práce se zabývá návrhem vývojového kitu s FPGA obvodem. Tento kit umožňuje právě vývoj, implementaci a testování digitálních obvodů, ale také například zpracování vysokorychlostních dat. Mimo to nabízí flexibilitu, velký výpočetní výkon, značné paměťové úložiště a jiné. Díky této univerzální platformě je možné otestovat funkcionalitu navrženého digitálního obvodu před zahájením výroby finálního zařízení, a tak celý proces optimalizovat a předejít finančním a časovým ztrátám. Vývojový kit se mimo to také používá na různé specifické aplikace a vývoj budoucích ASIC obvodů.

Mimo samotný vývojový kit je v této práci taktéž rozebrána problematika programovacích technologií logických obvodů a návrh programovacího zařízení pro konfiguraci Xilinx FPGA obvodů.

1 Programovatelné logické obvody

Programovatelné logické obvody, taktéž PLD (Programmable Logic Devices), je obecný název označující jakýkoli typ integrovaného obvodu používaného pro implementaci digitálního hardwaru, kde samotný čip může být nakonfigurován koncovým uživatelem. Programování takového zařízení může zahrnovat umístění čipu do speciálního programovacího zařízení, nicméně všechny moderní čipy mohou být nakonfigurovány přímo v systému.

Moderní integrované obvody spadající do této kategorie se nazývají SPLD (Simple PLD), CPLD (Complex PLD) a FPGA (Field-Programmable Gate Array). [1]

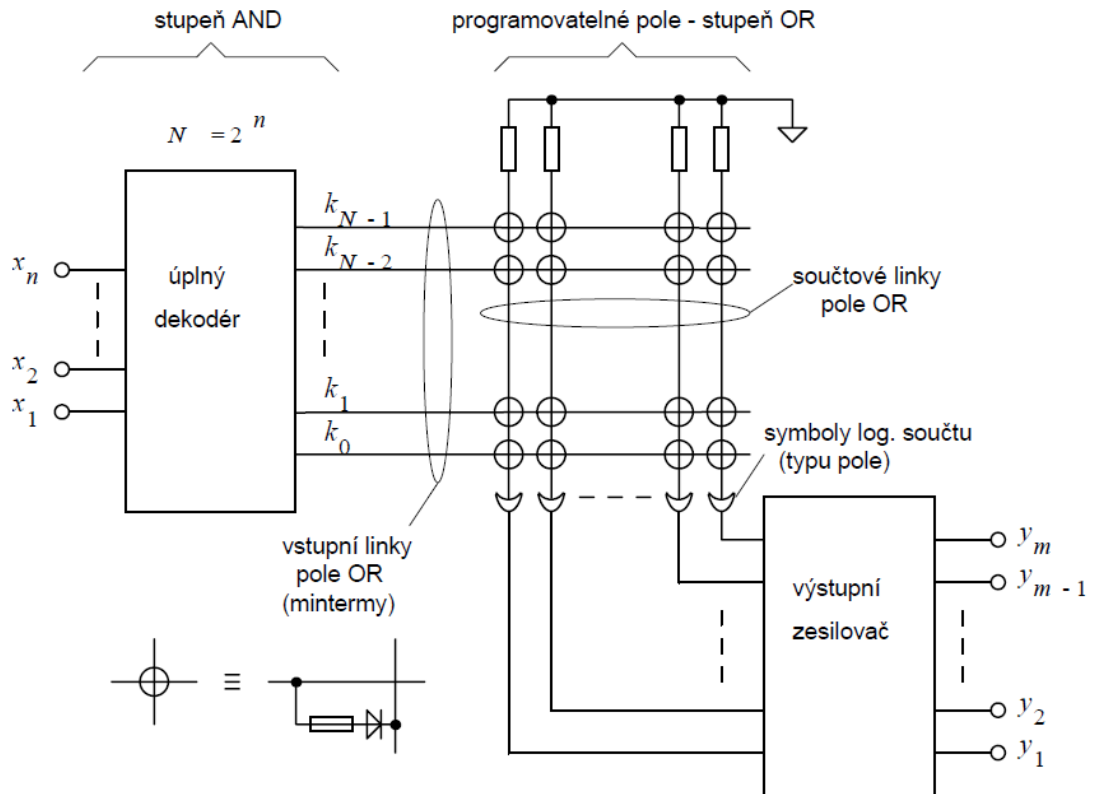
1.1 SPLD obvody

Jednoduché programovatelné obvody, taktéž SPLD (Simple Programmable Logic Devices), mají název odvozený od faktu, že jejich interní struktura je jednodušší než například u CPLD nebo FPGA obvodů. SPLD obvody jsou založeny na dvoustupeňové struktuře realizující kombinační logické funkce typu součet součinů. První stupeň je tvořen polem AND hradel a druhý stupeň je tvořen polem OR hradel. Propojení jednoho nebo obou polí může být nakonfigurováno takovým způsobem, aby reprezentovalo logickou funkci vyjádřenou ve formátu součet součinů. Vnitřní strukturu lze realizovat ve třech variacích:

1. PROM (Programmable Read Only Memory) – kde stupeň AND je fixní a stupeň OR programovatelný
2. PAL (Programmable Array Logic) – kde stupeň AND je programovatelný a stupeň OR fixní
3. PLA (Programmable Logic Array) – kde jsou oba stupně AND i OR programovatelné

Na rozdíl od FPGA a CPLD obvodů, SPLD obvody nemají obecnou propojovací strukturu, a proto je jejich funkčnost výrazně omezena. Z tohoto důvodu jsou SPLD postupně vyřazovány. PROM, PAL a PLA jsou novodobě používány spíše jako interní bloky CPLD obvodů, popřípadě ASIC (Application Specific Integrated Circuit) obvodů, než jako samostatný čip. [2, 3]

Paměti PROM (viz obr. 1.1) jsou univerzální v tom smyslu, že v nich lze realizovat jakoukoli funkci jejich vstupních proměnných. To je umožněno díky tomu, že stupeň AND představuje úplný dekodér vstupních signálů vytvářející všechny možné

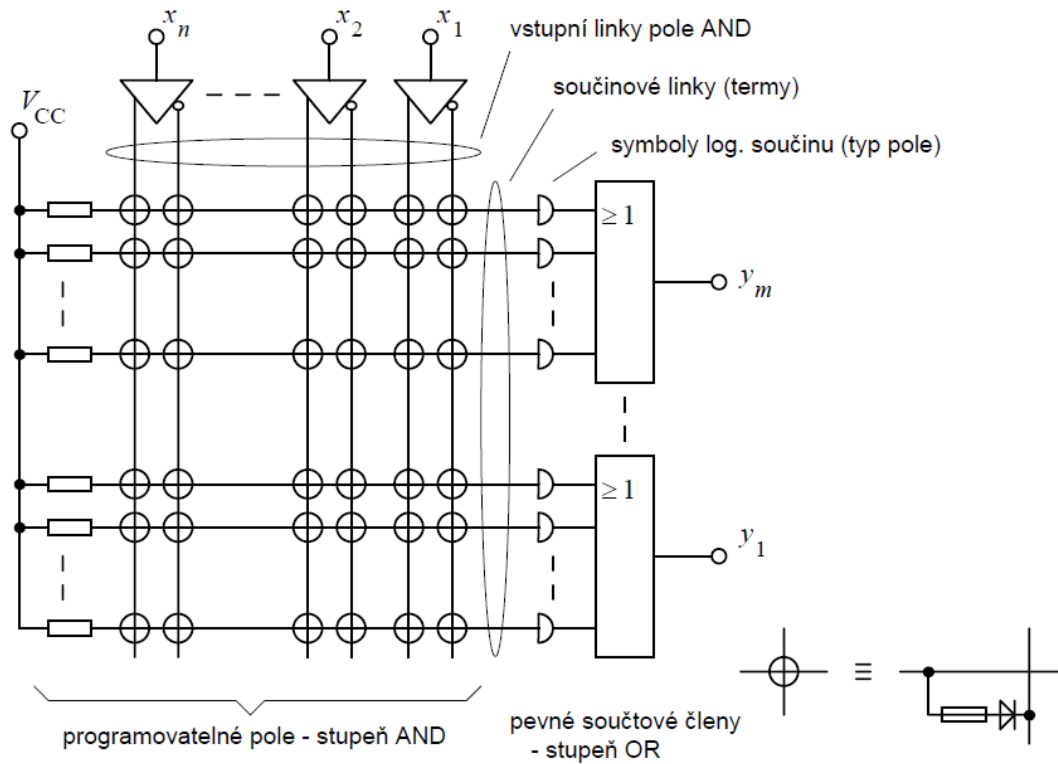


Obr. 1.1: Struktura PROM [3]

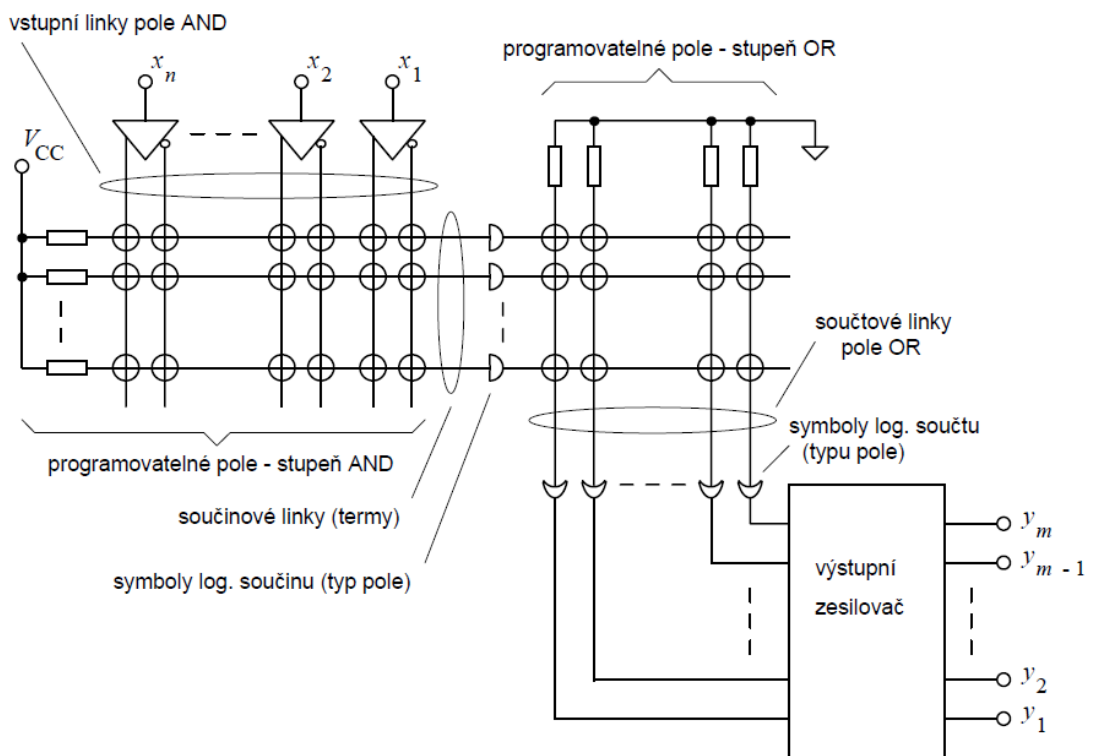
mintermy. Jejich hlavní nevýhodou pro účely realizace logických funkcí je, vedle neúspornosti struktury, především jejich (ne)dostupný sortiment. Paměti PROM se totiž vyrábějí především jako prvky pro uchování většího objemu dat, proto jsou pro obvyklé funkce většinou příliš velké, takže se zřídka využijí z více než několika procent svých možností. Velikost celé struktury, především programovatelného pole, má za následek značné parazitní kapacity, takže běžné paměti PROM jsou zhruba o řád pomalejší než jiné kombinační logické obvody. [3]

Funkce ve struktuře PAL (viz obr. 1.2) jsou realizovány na základě minimálního tvaru zápisu, takže pro běžné funkce bývá velikost programovatelného pole značně menší než u struktury PROM. Tato výhoda je však vykoupena ztrátou univerzality, protože ve struktuře PAL lze (s jedním průchodem) vytvořit jen funkce, u nichž počet termů nepřesahuje počet součinných linek příslušejících ve struktuře jedné buňce. V současné době se obvody PLD se strukturou PAL používají nejčastěji. [3]

Obvody se strukturou PLA (viz obr. 1.3) mohou představovat nejúspornější řešení. Součinné termy mohou být současně využity pro více vytvářených funkcí a mohou jim být přiřazovány podle skutečné potřeby. Tato struktura je charakteristická vel-



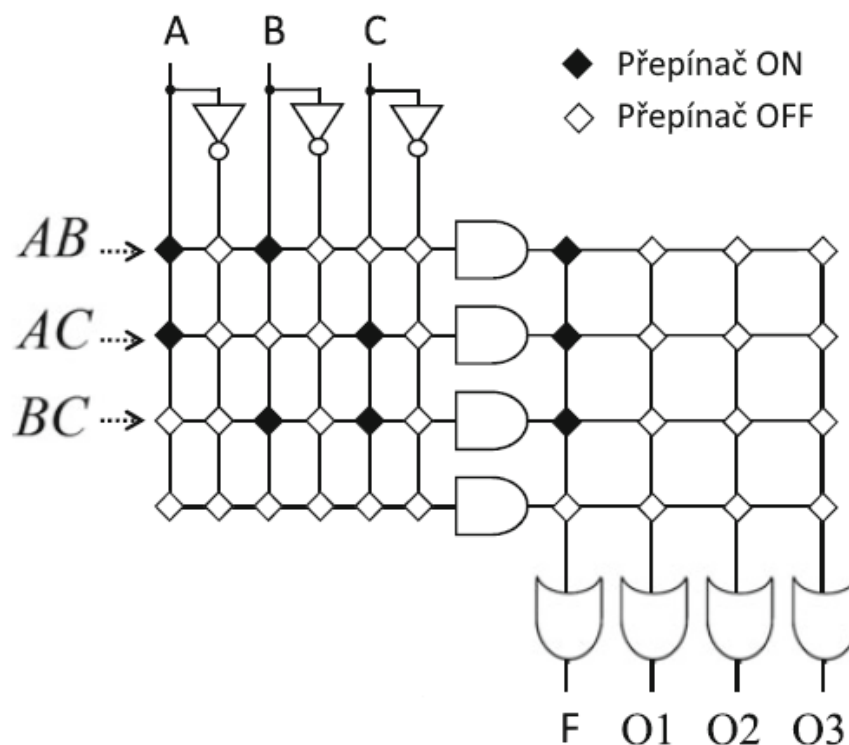
Obr. 1.2: Struktura PAL [3]



Obr. 1.3: Struktura PLA [3]

kým počtem programovatelných propojení, a proto v době vývoje techniky obvodů PLD nebyla příliš úspěšná. Dnes již však jsou technologické předpoklady k využití jejich výhod lepší a struktura PLA se znovu objevuje v některých nových typech obvodů CPLD. [3]

Pro názornost je zde uveden příklad realizace jednoduché funkce $F = AB + AC + BC$ implementované do struktury PLA. Ve vstupním (AND) poli je každá proměnná (zde A , B a C) spojena se vstupy všech AND hradel pomocí programovatelných přepínačů. V následujícím (OR) poli jsou všechny vstupy OR hradel spojeny se všemi výstupy AND hradel taktéž za pomoci programovatelných přepínačů. Obecně lze tedy v AND poli naprogramovat k součinných termů s až n vstupními proměnnými. Kromě toho je k výstupů přivedeno do pole OR hradel, kde je možné naprogramovat až m součtů k vstupů. Na příkladu znázorněném na obr. 1.4 je tedy možné implementovat až 4 logické funkce 3 vstupních proměnných vyjádřených jako součet součinů. [6]



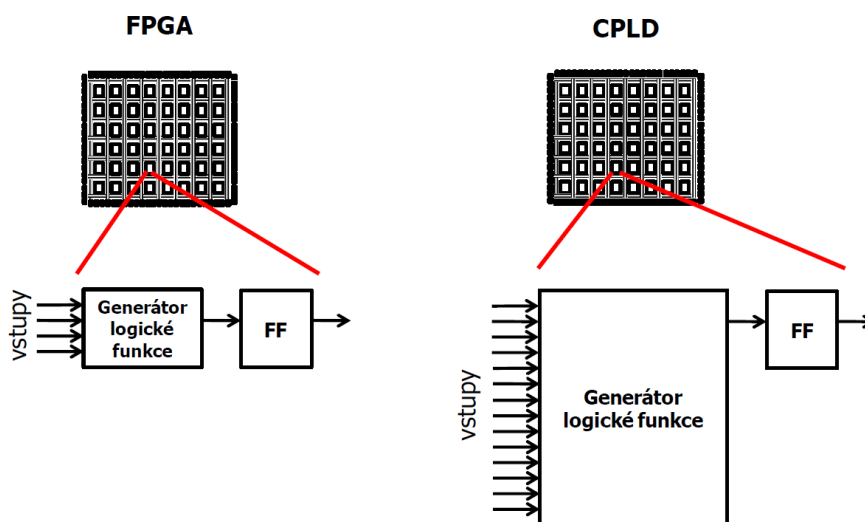
Obr. 1.4: Implementace součinných termů funkce [6]

1.2 CPLD obvody

Komplexní programovatelné obvody, zkratkou CPLD (Complex Programmable Logic Devices), obsahují v porovnání s SPLD obvody sofistikovanější logické buňky. Jejich logická buňka se typicky skládá z klopného obvodu typu D a makrobuněk se strukturou PAL, popřípadě PLA. Vnitřní programovatelná propojovací struktura obvodů CPLD bývá více centralizovaná s několika skupinami soustředěných propojovacích linek.

Naproti tomu logická buňka FPGA obvodu je obvykle menší. Typicky se sestává z klopného obvodu typu D a malé náhledové tabulky LUT (Look-Up Table), nebo skupiny multiplexorů. Vnitřní propojovací struktura mezi logickými buňkami bývá distribuovaná a flexibilnější. Pro svou distribuční povahu jsou FPGA obvody vhodnější pro velké a vysoko kapacitní programovatelné obvody.

Z hlediska architektury jde o koncept velmi podobný obvodům FPGA – pole programovatelných buněk doplněné o programovatelnou propojovací strukturu. Rozdíl je ve velikosti elementární buňky, a tím pádem ve složitosti kombinační logické funkce, kterou buňka dokáže realizovat (obr. 1.4). FPGA obvod je založen na náhledových tabulkách (LUT), které mají typicky 4-6 vstupů, kdežto CPLD obvod má vstupů výrazně více. Problematika volby vhodné velikosti elementární buňky bude nastíněna v podkapitole 1.3 FPGA obvody. [2, 3, 4]



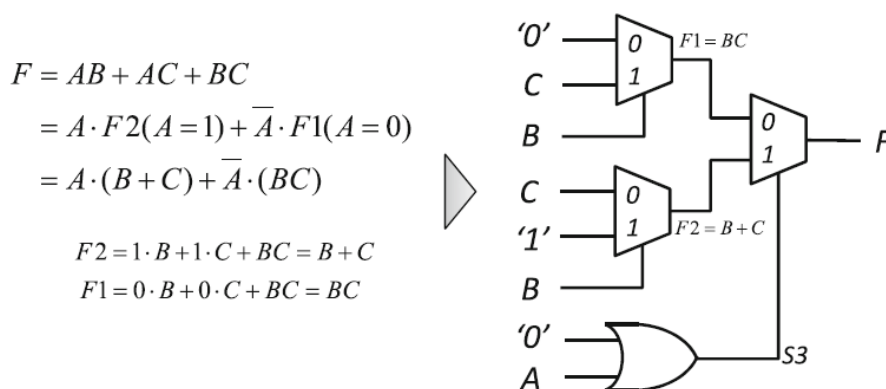
Obr. 1.5: Porovnání základní buňky FPGA a CPLD obvodů [4]

1.3 FPGA obvody

Programovatelné hradlové pole, taktéž FPGA (Field Programmable Gate Array), je typ integrovaného obvodu, který může být použit pro implementaci jakéhokoliv digitálního obvodu. Architektura FPGA obvodu je tvořena základními bloky, jako jsou programovatelné logické buňky, vstupně-výstupní buňky a programovatelná propojovací struktura. Pro zvýšení výkonnosti čipu se do jeho struktury integrují některé další elementy, jako jsou například blokové RAM, DSP bloky nebo mikroprocesory. [4, 6]

K realizaci logické funkce v FPGA obvodech je využito buď multiplexorů (MUX), nebo náhledových tabulek (LUT).

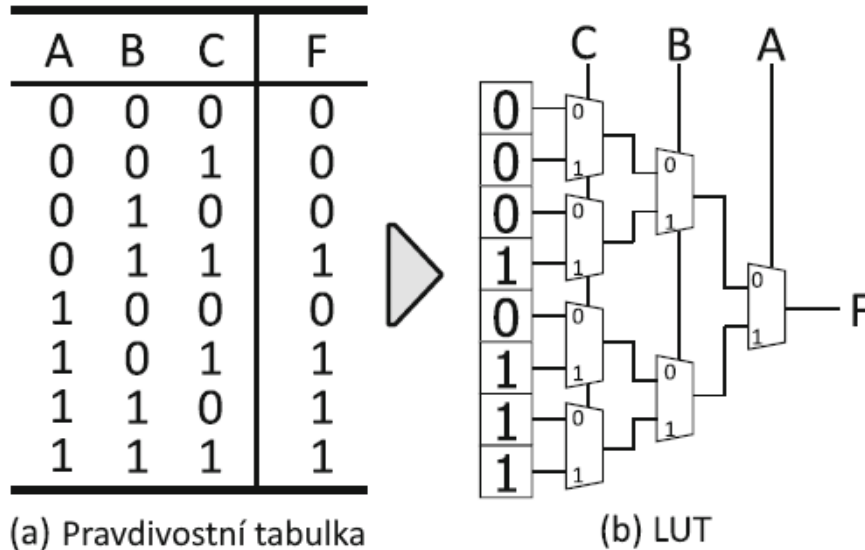
V případě MUX struktury je zde jako příklad uvedena logická buňka realizovaná třemi 2-1 multiplexory (2 vstupy, 1 výstup) a jedním OR hradlem (viz obr. 1.6). Na rozdíl od zbylých dvou způsobů realizace logické funkce, zde není možné implementovat všechny funkce vstupních proměnných. Jestliže 2-1 MUX uvažujeme jako logickou buňku se třemi vstupy a jedním výstupem, můžeme ji využít k realizaci maximálně 10 různých logických obvodů (v případě LUT o stejném počtu vstupů i výstupů by se jednalo o 256 různých obvodů). Je možné implementovat klopné obvody typu „latch“ i „flip flop,“ hradla NAND, AND, OR a NOR až na čtyři vstupy a také vstupy invertovat a vytvářet složená hradla (např. AND-OR a OR-AND). [6]



Obr. 1.6: Implementace logické funkce pomocí multiplexorů [6]

Při samotné realizaci funkce je nejprve provedeno Shannonovo rozšíření logického výrazu o proměnnou A, čímž vzniknou dvě parciální funkce F1 a F2. Vzniklé funkce v tomto případě odpovídají funkcím AND a OR, tudíž obě mohou být realizovány pomocí jednoho multiplexoru. V současnosti se tento způsob realizace logických funkcí v komerčně dostupných obvodech nevyužívá. [6]

Druhý způsob, v současnosti nejvyužívanější, je založený na LUT. Jde o malé paměťové pole (typicky SRAM), kde adresní vstupy paměti jsou reprezentovány vstupy LUT a výstup paměti je analogický k výstupu LUT. Obecně platí, že LUT o N vstupech je složena z 2^N SRAM buněk (bitů) a z multiplexoru o $2 \times N$ vstupech a může realizovat 2^{2^N} logických funkcí. Implementace logické funkce $F = AB + AC + BC$ je zde znázorněna na příkladu 3-vstupé LUT (viz obr. 1.7). [6]



Obr. 1.7: Implementace logické funkce pomocí LUT [6]

Při realizaci logické funkce pomocí LUT dojde k vytvoření pravdivostní tabulky na základě počtu jejich vstupů. Hodnota funkce (sloupec F) je zapsána do konfigurační paměti, kde každému řádku odpovídá jedna paměťová buňka. Na základě kombinace vstupních proměnných je poté adresním dekodérem spojena příslušná paměťová buňka s výstupem. [6]

Volba počtu vstupů LUT je komplexní a jedná se o poměr mezi flexibilitou, plochou na čipu, propagačním zpožděním LUT aj. Jak již bylo zmíněno, potřebný počet paměťových buněk je 2^N , čili velikost paměti pro realizaci náhledové tabulky závisí na počtu jejich vstupů exponenciálně. V případě, že má LUT 6 vstupů, je počet potřebných paměťových buněk roven 64. Když by ovšem náhledová tabulka měla například 16 vstupů, byl by potřebný počet buněk již 65 536. Kromě paměťového pole samotného se také výrazně zvětšuje vstupní část paměti, kterou představuje adresní dekodér, což je fixní kombinační obvod, který pro každou kombinaci vstupních signálů aktivuje právě jednu paměťovou buňku.

V případě příliš velkého počtu vstupů by tedy hrozilo, že při implementaci jednodušších funkcí zůstane velká plocha čipu nevyužitá. Také by byla nutná implementace velkého adresního dekodéru, který by měl za následek poměrně velké distribuční

zpoždění a docházelo by k limitaci maximálního pracovního kmitočtu. V opačném případě, a to potřeby implementovat složitější logickou funkci, je nutné využít více LUT spojených propojovací strukturou, což má také za následek vzrůst zpoždění. [4, 9]

1.4 Programovací technologie

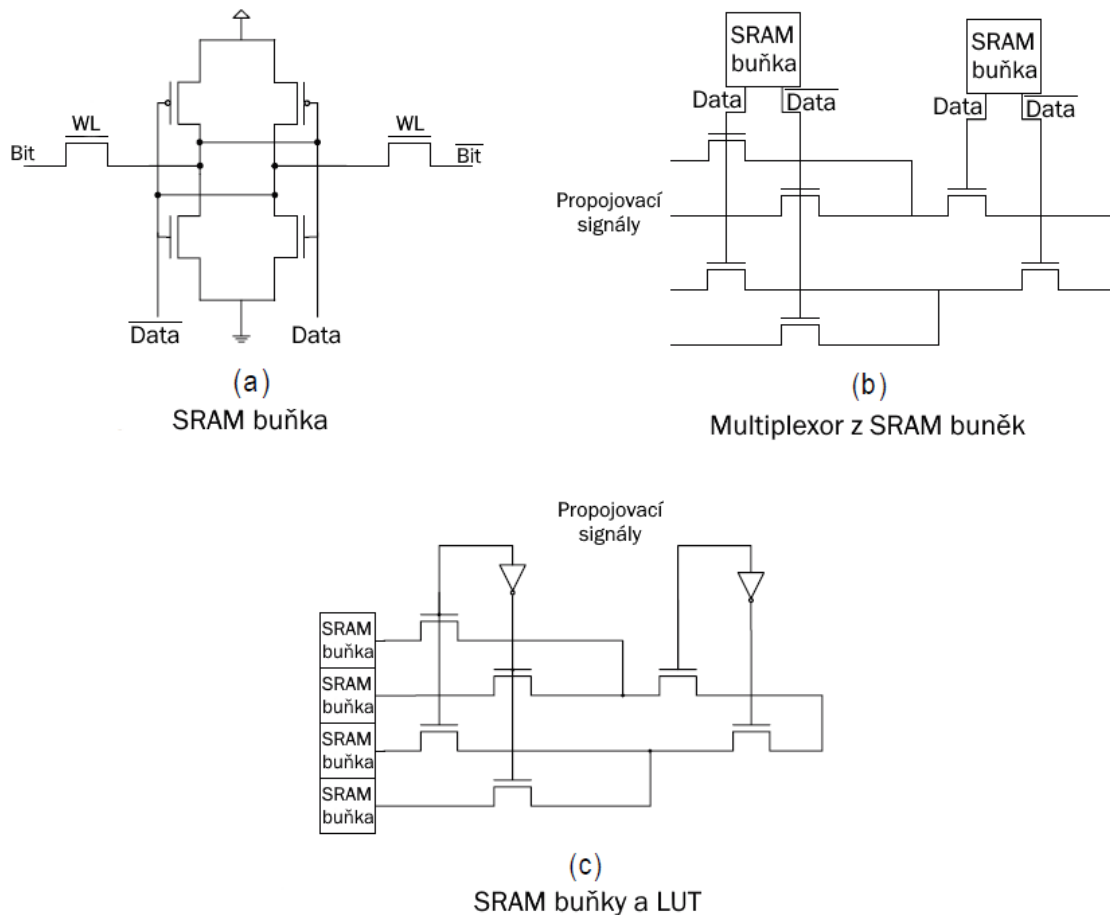
Programovací technologie je zásadní součástí každého programovatelného logického zařízení. Slouží k řízení přepínačů, které umožňují programovatelnost PLD. Existuje celá řada programovacích technologií a jejich rozdíly mají významný vliv na výslednou architekturu konfiguračních obvodů. Historicky se k těmto účelům používaly EPROM, EEPROM, FLASH, SRAM a ANTI-FUSE technologie. V současných FPGA obvodech zůstaly zachovány pouze technologie FLASH, SRAM a ANTI-FUSE, které budou stručně popsány. [5]

1.4.1 SRAM technologie

Jak už bylo naznačeno, jedná se v současnosti o nejpoužívanější programovací technologii pro FPGA obvody. Základem SRAM technologie jsou statické paměťové buňky (viz obr. 1.8 a), které jsou rozloženy napříč celým FPGA zajišťující konfigurovatelnost. Pro SRAM buňky existují dvě základní využití. Většina z nich je využívána na určení signálů vstupujících do multiplexorů, které následně řídí propojovací signály. Pro FPGA obvody založené na SRAM technologii je typické využití náhledových tabulek ve své architektuře, a právě v nich je využita většina zbývajících SRAM buněk za účelem uchování dat. Na obrázcích obr. 1.8 b) a obr. 1.8 c) jsou znázorněny zmíněné dva způsoby využití.

SRAM programovací technologie se stala dominantní v oblasti FPGA obvodů díky svým dvěma zásadním výhodám: přeprogramovatelnost a využití standardní CMOS technologie. Z praktického hlediska může být SRAM buňka nakonfigurována nespočetněkrát. Příslušné obvody na FPGA inicializují všechny SRAM bity po připojení k napájení a následně je naprogramují dle konfiguračního souboru (bitstreamu) dodaného uživatelem. Oproti ostatním programovacím technologiím nejsou vyžadovány žádné další speciální výrobní metody nad rámec standardních CMOS. Navíc FPGA obvody založené na SRAM technologii mohou využívat nejnovější dostupnou CMOS technologii, a tak těžit ze zvyšujícího se stupně integrace, zvyšující se rychlosti a snižování nároků na dynamický odběr energie, které nabízí nové technologické procesy s menšími minimálními rozměry.

Nicméně existuje zde i řada nevýhod. První z nich je velikost, neboť SRAM buňky vyžadují 5 nebo 6 tranzistorů a programovatelný element k propojení signálů



Obr. 1.8: Způsoby využití SRAM buněk [5]

vyžaduje alespoň jeden samostatný tranzistor. Druhou nevýhodou může být volatilita této technologie. Vzhledem k tomu, že SRAM buňky potřebují pro uchování informace trvalou přítomnost napájecího napětí, je tedy vyžadováno použití externích zařízení k trvalému uložení konfiguračních dat. Tyto externí flash, popřípadě EEPROM paměti zvyšují cenu FPGA obvodů založených na SRAM technologii. Za další nevýhodu lze označit možný problém se zabezpečením. Po připojení obvodu k napájení dochází k načtení konfiguračního souboru a během tohoto procesu může být soubor teoreticky zachycen a odcizen například konkurenční společností. Nicméně moderní FPGA produkty často poskytují pro konfigurační soubory šifrovací metody, které toto riziko efektivně eliminují. [5]

1.4.2 Flash/EEPROM technologie

Jedna z alternativ, která řeší nedostatky SRAM technologie, je použití programovací technologie založené na plovoucích hradlech. Principem takové technologie je dodání náboje na hradlo, které „plave“ nad tranzistorem. Toto řešení se vyskytuje

právě v pamětech typu flash nebo EEPROM. Samotné buňky těchto pamětí jsou nevolatilní, tudíž nedojde ke ztrátě informace po odpojení napájecího napětí.

Programovací technologie založené na flash paměťových buňkách nabízí několik podstatných výhod. Díky tomu, že jsou nevolatilní, dochází k eliminaci potřeby použití externího zařízení pro uložení konfiguračních dat. Dále dokáže flash technologie fungovat okamžitě po připojení k napájení, namísto čekání na načtení konfiguračních dat. A také je efektivnější z hlediska velikosti plochy na čipu. V porovnání s anti-fuse technologií, což je taktéž nevolatilní programovací technologie, FPGA obvody založené na flash technologii jsou rekonfigurovatelné a mohou být programovány přímo na desce plošných spojů. Použití technologie plovoucích hradel k ovládní spínacího tranzistoru zvyšuje složitost designu FPGA, protože je třeba dbát na to, aby napětí na tranzistoru mezi elektrodami source-drain zůstalo dostatečně nízké a nedošlo tak k dodání náboje do plovoucího hradla. Ale vzhledem k tomu, že novější procesy vyžadují stále nižší napěťové úrovně, může se tento problém stát v budoucnu méně důležitým.

Naopak fakt, že počet jejich nakonfigurování je omezený, patří k jedním z nevýhod. Nahromadění náboje v oxidu nakonec zabrání tomu, aby bylo zařízení dále správně vymazáno a naprogramováno. Dalším významným nedostatkem této technologie je potřeba použití nestandardního CMOS procesu. Obdobně jako SRAM technologie, i flash technologie trpí poměrně vysokou rezistivitou a kapacitou kvůli použití tranzistorových spínačů.

Trend, který se nedávno objevil, je použití flash paměti v kombinaci s SRAM programovací technologií. V takových zařízeních je potom flash paměť spolu s FPGA čipem v jednom pouzdře, propojeny bondovací technologií, a využita jako nevolatilní uložení, zatímco SRAM buňky jsou stále používány pro řízení programovatelných elementů. Tato kombinace sice vyřeší většinu podstatných nevýhod čisté SRAM technologie, nicméně negativně se podepíše na výsledné ploše integrovaného obvodu a také na jeho ceně. [5]

1.4.3 Anti-fuse technologie

Tato technologie je založena na strukturách, které za normálních okolností vykazují velmi vysokou rezistivitu, ale lze je programovatelně „prorazit“ (ve skutečnosti spojit) a vytvořit tak elektrické spojení s nízkou rezistivitou. Obvod vytvořený tímto způsobem je permanentní, na rozdíl od předchozích technologií. Programovatelný prvek je v tomto případě přímo použitý pro přenos signálů. K implementaci této technologie se přistupovalo dvojím způsobem. Dielektrické anti-pojistky se skládají z nevodivého dielektrika oxid – nitrid – oxid umístěného mezi vodivou N+ difúzní vrstvou a křemík. Přiložení dostatečně vysokého napětí způsobí proražení dielek-

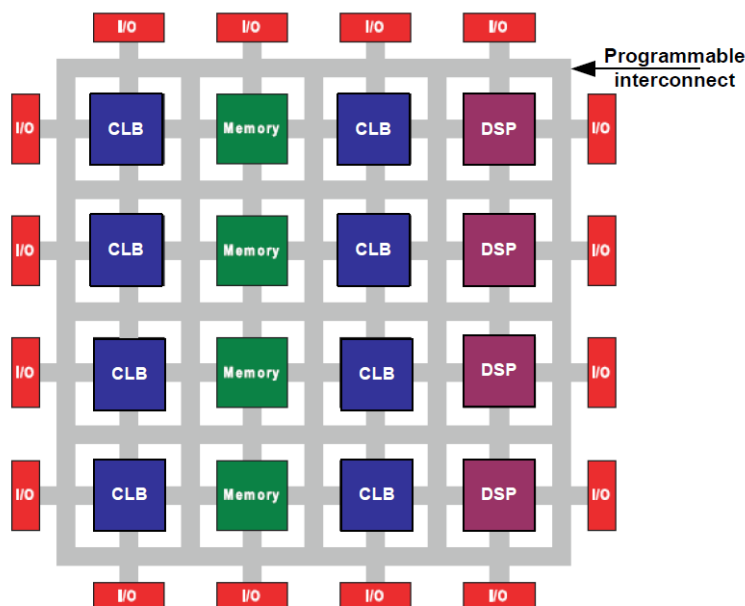
trika a vytvoření vodivého spojení s rezistivitou typicky mezi 100 až 600 Ω . Později byl tento způsob využívání dielektrika nahrazen technologií na bázi kov-kov. Takové anti-pojistky jsou vytvořeny vložením izolačního materiálu (např. amorfni křemík nebo oxid křemičitý) mezi dvě kovové vrstvy. Dostatečně vysokým napětím je opět možné prorazit anti-pojistku a dosáhnout vodivého spojení. Výhodou kov-kov technologie je snížení výsledné rezistivity ve vodivém stavu, a to na hodnotu 20 - 100 Ω .

Hlavní výhodou anti-fuse technologie je malá plocha na čipu. Avšak přítomnost relativně velkého tranzistoru, který je potřebný pro dodání dostatečně velkého proudu za účelem konfigurace, tuto výhodu neguje. Další, neméně podstatnou, výhodou je menší rezistivita a parazitní kapacita v naprogramovaném stavu, než jakou vykazují ostatní technologie. Výhody této technologie dohromady umožňují implementovat více přepínačů do obvodu, než je praktické u ostatních technologií. Z principu funkce anti-fuse technologie je zřejmé, že se jedná o nevolatilní, jednou programovatelnou technologii, což může být žádoucí v aplikacích, kde je kladen důraz na zabezpečení zařízení.

Anti-fuse technologie má ovšem významné nevýhody, kvůli kterým není využívána. Zejména se jedná o fakt, že vyžaduje nestandardní CMOS proces, a to ji činí, ve srovnání s SRAM technologií, zastaralou. Navíc dle dostupných informací nejpokročilejší zařízení používají 0,15 μm technologii, která je několik generací pozadu oproti technologii používané pro nové CMOS standardy, a tudíž anti-pojistky již v současnosti nejsou využitelné. [4, 5]

2 Architektura FPGA

Následující kapitola podrobně popisuje architekturu FPGA obvodu. Popis se vztahuje výhradně k Xilinx FPGA obvodům rodiny Artix-7, a to jak ve smyslu struktury jednotlivých bloků, tak i ve smyslu názvosloví. Rozbor architektury a názvosloví nelze zobecnit, neboť nejsou standardizované a každý výrobce si je určuje samostatně. Typické rozpořádání bloků uvnitř Xilinx FPGA a jejich pojmenování je zobrazeno na obr. 2.1



Obr. 2.1: Základní bloky FPGA obvodu [5]

2.1 Programovatelná propojovací struktura (PI)

Programovatelná propojovací struktura v FPGA obvodech poskytuje elektrické propojení mezi logickými bloky a vstupně-výstupními bloky. Sestává se z vodičů a programovatelných přepínačů (k jejichž konfiguraci je využito programovacích technologií, viz podkapitola 1.4), které vytvářejí propojení za účelem vytvořit uživatelem navržený obvod.

Aby bylo možné implementovat širokou škálu obvodů, je nutné, aby byla propojovací struktura dostatečně flexibilní. Vyžadujeme od ní, aby umožňovala propojit jednotlivé elementy na krátké i dlouhé vzdálenosti, spolu s ohledem na spotřebu energie a na rychlost výsledného designu.

Lokální propojovací architektura definuje uspořádání přepínačů mezi logickými bloky a vnitřní propojovací strukturou a délku segmentu vodiče.

Globální propojovací architektura definuje relativní polohu propojovacích kanálů ve vztahu k umístění logických bloků, propojení kanálů mezi sebou a počet vodičů v každém kanálu. Může být charakterizována jako hierarchická nebo jako ostrůvková (island-style), nicméně všechny Xilinx FPGA obvody jsou založeny na ostrůvkové architektuře. Někdy se také označuje jako dvourozměrná architektura. [5, 6]

2.1.1 Globální dvourozměrná propojovací architektura

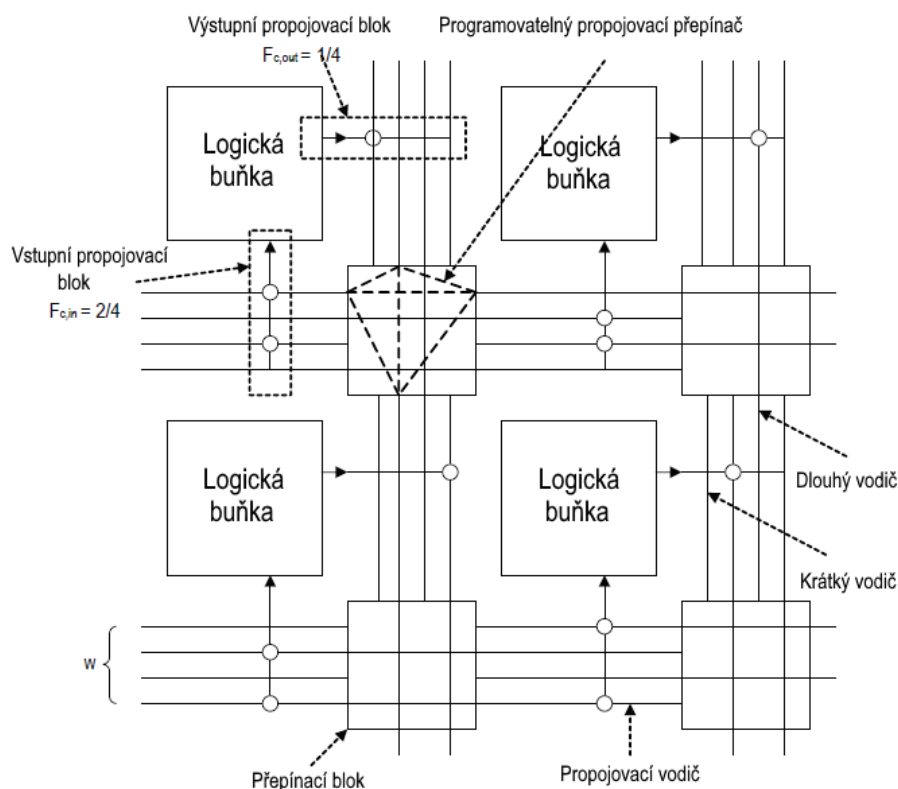
Globální dvourozměrná (ostrůvková) architektura je charakteristická tím, že logické bloky jsou spolu s propojovacími prvky rovnoměrně uspořádány ve dvourozměrné mřížce (viz obr. 2.1).

Typicky se propojovací kanály nacházejí podél všech čtyř stran logických bloků. Počet vodičů (značený W) obsažených v jednom kanálu, je přednastaven během výroby a jde o jeden z klíčových parametrů během návrhu samotného FPGA čipu. V této architektuře se obecně v každém kanálu používají segmenty vodičů různých délek ve snaze poskytnout pro každé dané propojení vodič s nejvhodnější délkou. Také se typicky liší umístění počátečního bodu jednotlivých segmentů tak, aby každá logická buňka měla možnost připojit se na vodič nejvhodnější délky. [5]

2.1.2 Lokální dvourozměrná propojovací architektura

Lokální dvourozměrná (ostrůvková) propojovací architektura definuje strukturu propojení mezi vodičovými segmenty a logickými a vstupně-výstupními bloky a také definuje samotné segmenty vodičů. Piny logických buněk, které jsou napojeny na propojovací strukturu, se dělí na vstupní piny (přivádí data do buňky) a výstupní piny. Jak je možné vidět na obr. 2.2, vstupní piny logického bloku jsou spojeny se segmenty vodičů v propojovací struktuře pomocí přepínačů soustředěných v tzv. vstupním propojovacím bloku. Výstupní piny jsou obdobně připojeny prostřednictvím přepínačů ve výstupním propojovacím bloku. Zlomek vodičových segmentů v propojovacím kanálu, jež se mohou připojit na vstupní piny logické buňky, se nazývá flexibilita vstupního propojovacího bloku a značí se $F_{c,in}$. Obdobně pro připojení výstupních pinů logického bloku je zaveden pojem flexibilita výstupního propojovacího bloku a značí se $F_{c,out}$. [5]

Přepínací blok realizuje propojení mezi jednotlivými segmenty vodičů v každém průsečíku horizontálních a vertikálních kanálů. Každý přepínací blok obsahuje řadu přepínačů, které umožňují vstupním segmentům napojení na segmenty v přilehlých kanálech. I v tomto případě se dá hovořit o flexibilitě přepínacího bloku označované jako F_s . Flexibilita přepínacího bloku udává počet možných propojení v daném bloku, které může jeden segment vytvořit s ostatními segmenty. Na obr. 2.2

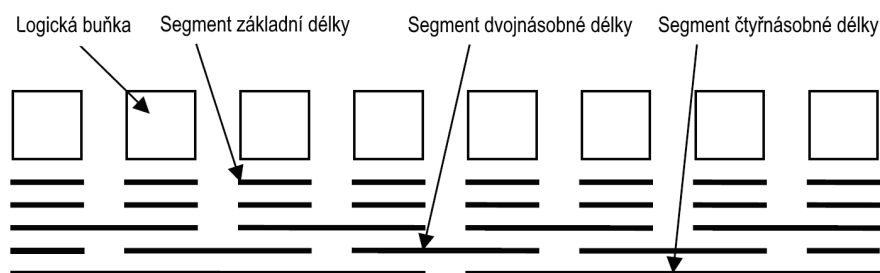


Obr. 2.2: Lokální dvourozměrná architektura [5]

je počet propojení uvnitř přepínacího bloku znázorněn čárkovanými čarami a pro tento případ je F_s rovno 3. [5]

Z důvodu komplikovanosti a velkého počtu jednotlivých elementů propojovací struktury je obtížné vytvořit ideální, tedy nejkratší, možné propojení pro všechny obvody. Proto ve struktuře existují různě dlouhé segmenty (viz obr. 2.3). Délka vodičového segmentu je předurčena počtem logických bloků. Na tomto obrázku jsou kromě základních délek segmentů také dvojnásobné a čtyřnásobné. Základní délky se zde vyskytují ve 40 %, dvojnásobné délky obdobně ve 40 % a čtyřnásobné délky ve 20 %. Také jsou do struktury přidány dlouhé vodiče, které protínají celé zařízení.

Propojovací elementy, ať už vstupní a výstupní propojovací bloky, nebo přepínací bloky, mají velký dopad na výslednou plochu FPGA obvodu a také na jeho zpoždění. Na výsledné ploše obvodu se výraznou měrou podílejí z toho důvodu, že je potřeba velké množství těchto elementů pro zajištění univerzality FPGA obvodu. Co se týče zpoždění, v současné době zpoždění vzniklé na propojovací struktuře převládá nad zpožděním jednotlivých logických hradel. Je tedy zřejmé, že je potřeba volit kompromis mezi flexibilitou propojovací struktury a výkonem. [5, 6]



Obr. 2.3: Příklad rozložení různě dlouhých segmentů [5]

2.2 Distribuce hodinového signálu

Strukturu distribuční sítě hodinového signálu je možné rozdělit na globální síť, regionální síť a hodinový signál pro I/O buňky. Kromě prostředků pro distribuci hodinového signálu se v FPGA nachází také bloky CMT (Clock Management Tiles) pro úpravu jeho vlastností.

V každém Xilinx FPGA obvodu 7. řady se nachází 32 globálních hodinových linek s nejvyšším fanoutem, na které mohou být napojeny veškeré příslušné piny klopných obvodů (CLK, CE, SR) a piny dalších logických obvodů. K přístupu ke globální hodinové síti je využito globálních bufferů. V každém regionu je až 12 globálních hodinových linek, které mohou být řízeny jakoukoli kombinací 32 globálních bufferů. Globální hodinové signály jsou nejčastěji řízeny pomocí CMT, což může zcela eliminovat základní zpoždění při distribuci hodinového signálu.

Regionální hodinové signály mohou řídit všechny elementy využívající hodinový signál v daném regionu. Region je definovaný jako oblast o velikosti 50 I/O a 50 CLB buněk. Každý regionální buffer může být řízen z jakéhokoliv ze 4 vstupní pinů podporujících hodinový signál (CC piny) a frekvence může být dělitelná v rozsahu 1 až $8\times$.

Hodinové signály pro vstupně-výstupní buňky jsou obzvláště rychlé a slouží pouze pro I/O logiku a serializéry/deserializéry. Mají přímé propojení s MMCM (Mixed-Mode Clock Manager) blokem pro nízkošumová a vysoce výkonná rozhraní.

Kromě stromové struktury distribuce hodinového signálu a velkého počtu dedikovaných bufferů patří mezi důležité části také CMT bloky. Každý CMT blok se skládá z elementů MMCM a PLL (Phase-Locked Loop). MMCM i PLL slouží jako frekvenční syntezátory pro širokou škálu frekvencí, jako jitter filtr pro interní i externí hodinové signály a pro rovnoměrnou distribuci hodinového signálu. Hlavní schopností PLL a MMCM elementů je nekonečně jemný fázový posun v obou směrech a také použití v režimu dynamického fázového posunu. [14, 15]

2.3 Vstupně-výstupní buňky (IOBs)

Jednotlivé vstupně-výstupní piny se typicky sdružují do větších seskupení nazývaných vstupně-výstupní banky. V rodině Artix-7 se nachází dva typy bank, a to vysoko výkonné (HP; High-Performance) a široko rozsahové (HR; High-Range). Banky HP jsou navrženy tak, aby splňovaly výkonnostní požadavky vysokorychlostních pamětí a dalších mezi čipových rozhraní s napětím až 1,8 V. Banky HR jsou navrženy tak, aby podporovaly širší škálu I/O standardů s napětím až 3,3 V.

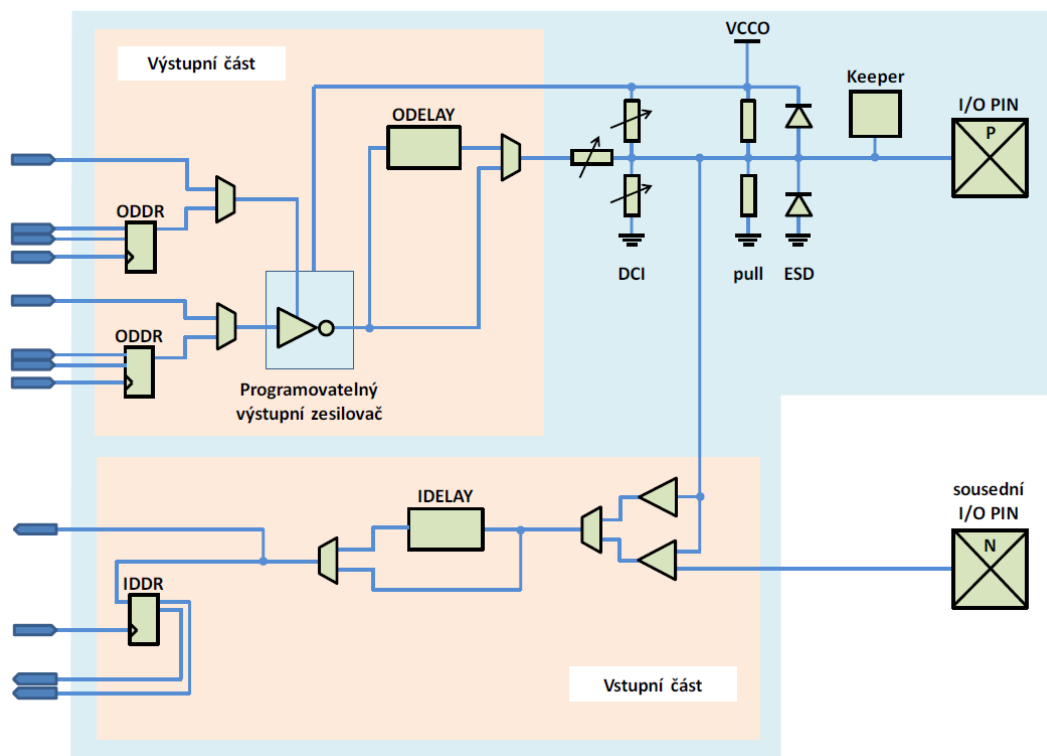
Každá I/O banka obvykle obsahuje 50 I/O pinů. Dva piny na samém kraji banky lze použít pouze s jedno-vodičovými I/O standardy. Zbylých 48 pinů může být použito buďto s jedno-vodičovými, nebo diferenciálními standardy. Při potřebě zpracování diferenciálního signálu dojde ke spojení dvou sousedních pinů do páru, kde jeden představuje P (positive) větev a druhý N (negative) větev. Každý I/O pin obsahuje 3-stavový řadič, který umožňuje jeho nastavení jako vstupní pin, výstupní pin, nebo nastavení do stavu vysoké impedance. Standardy, jež je možné implementovat pomocí I/O pinů, jsou následující:

- Jedno-vodičové I/O standardy: LVCMOS, LVTTL, HSTL, PCI a SSTL
- Diferenciální I/O standardy: LVDS, Mini_LVDS, RSDS, PPDS, BLVDS a diferenciální HSTL a SSTL

Xilinx FPGA obvody 7. řady poskytují rozsáhlou sadu funkcí, jež zahrnuje programovatelné řízení výstupního výkonu a doby přeběhu (SR; Slew Rate), terminaci na čipu pomocí digitálně řízené impedance (DCI; Digitally-Controlled Impedance) a schopnost vnitřně vytvořit referenční napětí (INTERNAL_VREF). Banky HR nicméně DCI neobsahují, proto veškerý popis DCI bude vztahován k HP bankám. Zjednodušená struktura I/O buňky je zobrazena na obr. 2.4. [4, 11]

V levé části jsou vstupy a výstupy buňky, které zajišťují její vazbu na programovatelnou strukturu pole. Výstupní část obsahuje blok ODDR, což je obvod podporující výstup dat technologií DDR (Dual Data Rate).

Následuje výstupní programovatelný zesilovač (angl. output buffer), do kterého mohou data vstupovat buďto z ODDR bloku, nebo přímo z programovatelného hradlového pole. Ten převádí signál, který má logické úrovně odpovídající jádru FPGA na signál, který má logické úrovně požadované obvody mimo FPGA. Logické úrovně výstupního signálu jsou dány napájecím napětím příslušné banky a konfigurací FPGA. Programovatelný výstupní zesilovač je také možné nastavit do stavu vysoké impedance, čímž je umožněno I/O buňku použít jako vstupní. Další výhodou je programovatelný slew rate, což umožňuje snížit strmost hran (a tím elektromagnetické rušení) u méně kritických signálů. Výstupní zesilovače také umožňují řídit



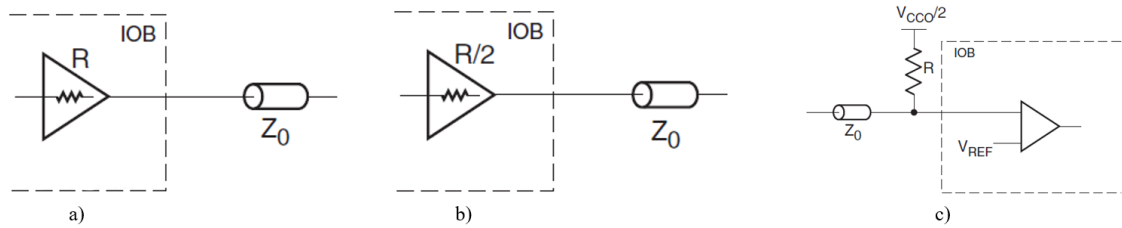
Obr. 2.4: Zjednodušené schéma vstupně-výstupní buňky FPGA obvodu [4]

velikost výstupního proudu, a to v několika diskretních rozsazích v rozmezí 2 až 24 mA (dle typu banky a implementovaného standardu).

V HP bankách je za výstupním zesilovačem zařazen blok ODELAY. Je to blok o velikosti 31 tapů s kalibrovatelným rozlišením. ODELAY umožňuje zpožďovat výstupní signály na individuální bázi.

Následuje blok DCI, který umožňuje přizpůsobit impedanční zakončení signálních vedení přímo na čipu. V závislosti na I/O standardu může DCI řídit výstupní impedanci řadiče, nebo přidat paralelní terminaci na straně řadiče, popřípadě přijímače, s cílem přesného přizpůsobení charakteristické impedance přenosového vedení. DCI aktivně reguluje tyto impedance uvnitř I/O, aby byly zkalibrovány na externí přesné referenční rezistory umístěné na pinech VRN a VRP. To kompenzuje změny impedance v I/O buňce způsobené procesními změnami a také změny impedance způsobené kolísáním teploty a napájecího napětí. Výhodou DCI je, že eliminuje potřebu terminačních rezistorů na DPS, tím pádem ulehčuje návrh DPS jako takový, snižuje počet použitých komponent a zejména zlepšuje integritu signálů eliminací vzniku odrazu signálů. I/O standardy, které podporují řízení impedance řadičů pomocí DCI, mohou nakonfigurovat řadiče jako „řadič s řízenou impedancí“ (obr. 2.5a) a nebo jako „řadič s řízenou impedancí na polovinu“ (obr. 2.5b). Pro I/O standardy, které podporují paralelní terminaci, DCI vytvoří rozdělenou terminaci,

také označovanou jako Theveninovo zakončení (obr. 2.5c). [4, 11]



Obr. 2.5: Impedanční přizpůsobení cest pomocí DCI [11]

Vstup i výstup FPGA obvodu je možné upnout k napájecímu napětí, nebo zemi pomocí upínacích rezistorů (angl. pull-up, pull-down). Tím je zajištěno, že i při přechodných stavech (např. při připojení napájecího napětí) je přesně definovaný stav I/O pinu. V tomto FPGA jsou k danému účelu použity velké hodnoty rezistorů, a tak tvoří slabé upnutí (angl. weak pull-up/down), nicméně ve většině případů je to dostačující. Dalším blokem je tzv. keeper, který upíná linku do stavu logické 1 nebo 0 podle toho, v jakém stavu se naposledy nacházela. Tyto obvody dokáží zajistit, že i při ponechání vstupu ve stavu vysoké impedance je vstupní komparátor schopen jednoznačně určit logickou hodnotu signálu a nehrozí tak vznik metastability na jeho výstupu.

Každý pin je také vybaven základní ESD (Electrostatic Discharge) ochranou v podobě dvojice diod. Ochranné diody jsou dimenzovány pouze na malé proudy a krátkodobé zatížení.

Vstupní část I/O buňky obsahuje obdobné bloky, jako výstupní část. Samotný vstup je tvořen komparátorem, který může pracovat buď s interní referencí, externí referencí, nebo jako diferenční komparátor, který využívá také sousední I/O buňku pro realizaci vstupu diferenčního páru. Za komparátorem se nachází blok IDELAY, který má funkcionalitu obdobnou jako ODELAY. Signál pak může vstupovat přímo do programovatelného hradlového pole, nebo může být zpracován dedikovaným blokem IDDR.

Mimo již zmíněné, relativně základní, bloky se v I/O buňkách nachází i další prokročilé elementy. Do této skupiny spadají například serializéry/deserializéry (SERDES). V FPGA obvodu konkrétně nalezneme blok ISERDESE2, který realizuje vstupní sériově-paralelní převody a blok OSERDESE2, který realizuje výstupní paralelně-sériové převody. Podporují vysokorychlostní přenos dat a umožňují vnitřní logice fungovat až na 8× menší rychlosti než samotná I/O buňka.

Výhody, které ISERDESE2 přináší, zahrnují vysokorychlostní datový přenos bez nutnosti synchronizace frekvence vnitřní logiky FPGA obvodu se vstupními daty, podporu jak SDR (Single Data Rate) tak DDR (Double Data Rate), sub-modul

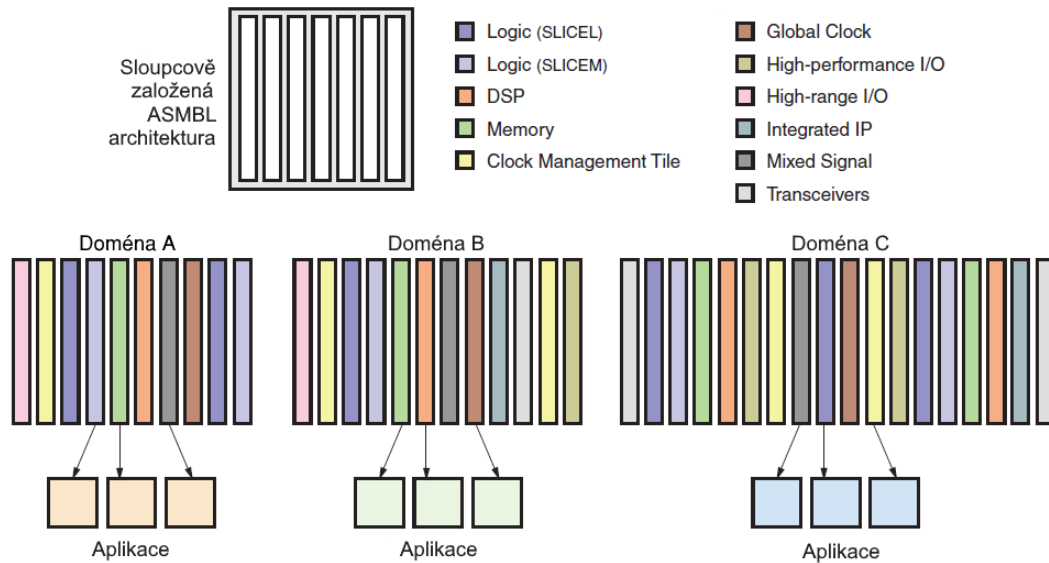
starající se o ztrátu bitové synchronizace a podporu pro paměťové rozhraní DDR3. V SDR módu může konvertor vytvořit 2 až 8 bitové slovo, v DDR módu může jeden ISERDESE2 blok vytvořit 4, 6, nebo 8 bitové slovo, dva kaskádově zapojené bloky dokážou vytvořit 10 nebo 14 bitové slovo.

OSERDESE2 obsahuje dedikovaný serializér pro data a pro řízení stavu 3-stavových elementů. Oba serializéry mohou být nastaveny v SDR i DDR módu. Datová serializace může být až 8:1 (10:1 a 14:1 jestliže je využita šířková expanze), 3-stavová serializace až 14:1. Také je zde dedikovaný DDR3 mód na podporu vysokorychlostních pamětí. [4, 11]

2.4 Logické buňky (CLBs)

Logické buňky, taktéž CLB (Configurable Logic Blocks), jsou hlavními elementy pro implementaci jak sekvenčních, tak i kombinačních obvodů. Poskytují pokročilou a vysoce výkonnou logiku – 6-vstupou LUT, nebo duální 5-vstupou LUT, distribuovanou paměť a posuvný registr, dedikovanou vysoko rychlostní carry logiku a multiplexory. Každý CLB blok se dále člení na dva řezy (angl. slices) a je připojen k propojovací matici, která zprostředkovává přístup ke globální propojovací struktuře. Jednotlivé řezy se spojují ve sloupcích, v rámci jedné buňky mezi sebou propojeny nejsou.

Existují tři možné typy řezů: SLICEM, SLICEL a SLICEX, nicméně SLICEX v Xilinx FPGA obvodech 7. řady není použitý. Řez typu SLICEX je svou strukturou podobný řezu SLICEL, ovšem nedisponuje rychlou carry logikou. Naposled byl tento typ použit v FPGA obvodech rodiny Spartan-6. Oba typy, SLICEM i SLICEL, obsahují čtyři generátory logických funkcí, neboli LUT, osm klopných obvodů typu D, multifunkční multiplexory a tzv. carry logiku, což je speciální struktura umožňující konstrukci velmi rychlých aritmetických obvodů. Rozdíl mezi jednotlivými řezy je ten, že řez typu SLICEM (viz obr. 2.6) podporuje dvě funkce navíc – ukládání dat využíváním distribuované RAM (256×1b) a řazení dat pomocí posuvných registrů (128 bitů). Každá logická buňka může obsahovat dva řezy typu SLICEL, nebo jeden řez typu SLICEL a jeden řez typu SLICEM, nicméně ve výsledném součtu se v FPGA obvodu nachází dvakrát více řezů typu SLICEL. [7, 8, 9]



Obr. 2.7: ASMBL architektura [8]

2.4.1 Náhledová tabulka (LUT)

Náhledová tabulka se prioritně používá jako prostředek pro implementaci logických funkcí, proto se jí někdy také říká generátor funkcí. V rodině Artix-7 je každá LUT implementována se šesti nezávislými vstupy a dvěma nezávislými výstupy. Do náhledové tabulky může být implementováno:

- Libovolná Booleovská funkce šesti vstupních proměnných
- Dvě libovolné Booleovské funkce o pěti vstupních proměnných, za předpokladu, že obě funkce mají vstupní proměnné totožné
- Dvě libovolné Booleovské funkce o třech, dvou a nebo méně vstupních proměnných

Při implementaci funkce šesti proměnných jsou potom využity vstupy $A_1 - A_6$ a výstup O_6 . Při implementaci funkce pěti proměnných a méně je vstup A_6 nastaven do logické úrovně 1 a využity mohou být vstupy $A_1 - A_5$ spolu s oběma výstupy O_5 a O_6 .

Zpoždění signálu způsobené průchodem LUT je nezávislé na implementované funkci. Signály vystupující z náhledové tabulky mohou:

- Opustit řez - přes A, B, C, D výstup, jde-li o signál z výstupu O_6 , nebo přes AMUX, BMUX, CMUX a DMUX výstup, jde-li o signál z výstupu O_5
- Sloužit jako vstup do dedikovaného XOR hradla, jde-li o signál z výstupu O_6

- Sloužit jako vstup do řetězce carry logiky, jde-li o signál z výstupu O_5
- Sloužit jako vstup do výběrového multiplexoru, jde-li o signál z výstupu O_6
- Sloužit jako data pro vstup D klopného obvodu
- Sloužit jako vstup do multiplexorů F7AMUX nebo F7BMUX, jde-li o signál z výstupu O_6

Základní náhledové tabulky se v řezech nacházejí spolu se třemi multiplexory (F7AMUX, F7BMUX a F8MUX), pomocí kterých je možné zkombinovat až čtyři náhledové tabulky tak, aby bylo možné vytvořit funkci o sedmi, nebo dokonce osmi vstupních proměnných přímo uvnitř jednoho řezu.

- F7AMUX - slouží k realizaci sedmivstupých funkcí z LUT A a LUT B
- F7BMUX - slouží k realizaci sedmivstupých funkcí z LUT C a LUT D
- F78MUX - slouží k realizaci osmivstupých funkcí zkombinováním všech LUT

Funkce o více než osmi vstupních proměnných mohou být vytvořeny jedině využitím více řezů. [8]

Náhledovou tabulku v řezu SLICEM je navíc možné nakonfigurovat jako RAM, nebo posuvný registr.

Distribuovaná RAM

Paměť implementovaná do LUT se nazývá distribuovaná RAM. Je možné kombinovat LUT v řezu různými způsoby a dosáhnout tak paměti o vyšší kapacitě. Tyto paměti jsou v rámci jednoho řezu konfigurovatelné následujícími způsoby:

- Jedno-portová RAM o velikosti $32 \times 1b$, $64 \times 1b$, $128 \times 1b$ nebo $256 \times 1b$
- Dvou-portová RAM o velikosti $32 \times 1b$, $64 \times 1b$ nebo $128 \times 1b$
- Čtyř-portová RAM o velikosti $32 \times 2b$ nebo $64 \times 1b$
- Jednoduchá dvou-portová RAM o velikosti $32 \times 6b$ nebo $64 \times 3b$

Zápis do distribuované RAM je synchronní. Čtení je v základu asynchronní, nicméně může být přenastaveno na synchronní za pomoci flip-flopu v tentýž řezu. Přidáním flip-flopu je zároveň zvýšen výkon paměti, neboť dojde ke snížení zpoždění na úroveň zpoždění danou samotným flip-flopem. Distribuovaná RAM též sdílí hodinový vstup s ostatními elementy v řezu, proto je pro čtení nutné zavést signál WE (Write Enable). [8]

Posuvný registr

LUT může být taktéž nakonfigurována jako 32 bitový posuvný registr, bez nutnosti použít flip-flopy dostupné v řezu. Jestliže je LUT využita tímto způsobem, může

zpozdít sériová data v rozsahu 1 až 32 hodinových cyklů. V rámci řezu je opět možné LUT kaskádovat a dosáhnout tak až 128 bitového posuvného registru, čili zpozdít sériová data až o 128 hodinových cyklů. Také je možné tvořit větší posuvné registry napříč více řezy SLICEM. [8]

2.4.2 Klopný obvod

Klopný obvod je nejjednodušší element sekvenční logiky. Přejít mezi logickými stavy 1 (high) a 0 (low) probíhá skokově a je řízen hodinovým signálem. Dle způsobu reakce na řídicí hodinový signál můžeme rozdělit klopné obvody na obvody řízené úrovní signálu (latch) a na obvody řízené hranou signálu (FF; Flip-Flop). [10]

Ze schématu řezu SLICEM (obr. 2.6) vidíme, že se v jednom řezu nachází osm klopných obvodů typu D. Čtyři z nich mohou být nakonfigurovány jako hranou řízený flip-flop, nebo jako úrovní řízený latch. Datový vstup D těchto KO může být nastaven výstupem náhledové tabulky přes příslušný multiplexor (AFFMUX, BFFMUX, CFFMUX nebo DFFMUX), nebo samostatným vstupem řezu (AX, BX, CX nebo DX), který při své cestě k datovému vstupu náhledovou tabulkou neprochází. KO nakonfigurované jako latch jsou aktivní při logické úrovni signálu CLK v 0.

Zbylé čtyři KO mohou být nastaveny pouze jako hranou řízené flip-flopy. Datový vstup je v tomto případě nastavován výstupem náhledové tabulky O_5 , nebo opět samostatným vstupem řezu (AX, BX, CX nebo DX), který neprochází přes náhledovou tabulku. Tyto čtyři přídavné KO lze použít pouze v případě, kdy je první čtveřice KO nakonfigurovaná jako flip-flop. Na obr. 2.8 jsou vyobrazeny oba zmíněné typy KO v řezu se všemi možnými řídicími a datovými signály. [8]

Řídicí signály, mezi které patří clock (CLK), clock enable (CE) a set/reset (SR), jsou v rámci jednoho řezu společné pro všechny klopné obvody. Jestliže má jeden KO povolené signály SR a CE, ostatní KO mají taktéž povolené signály SR a CE. Signál CLK má jako jediný z uvedených signálů navíc programovatelnou polaritu. Signály CE a SR jsou aktivní v logické 1. [8]

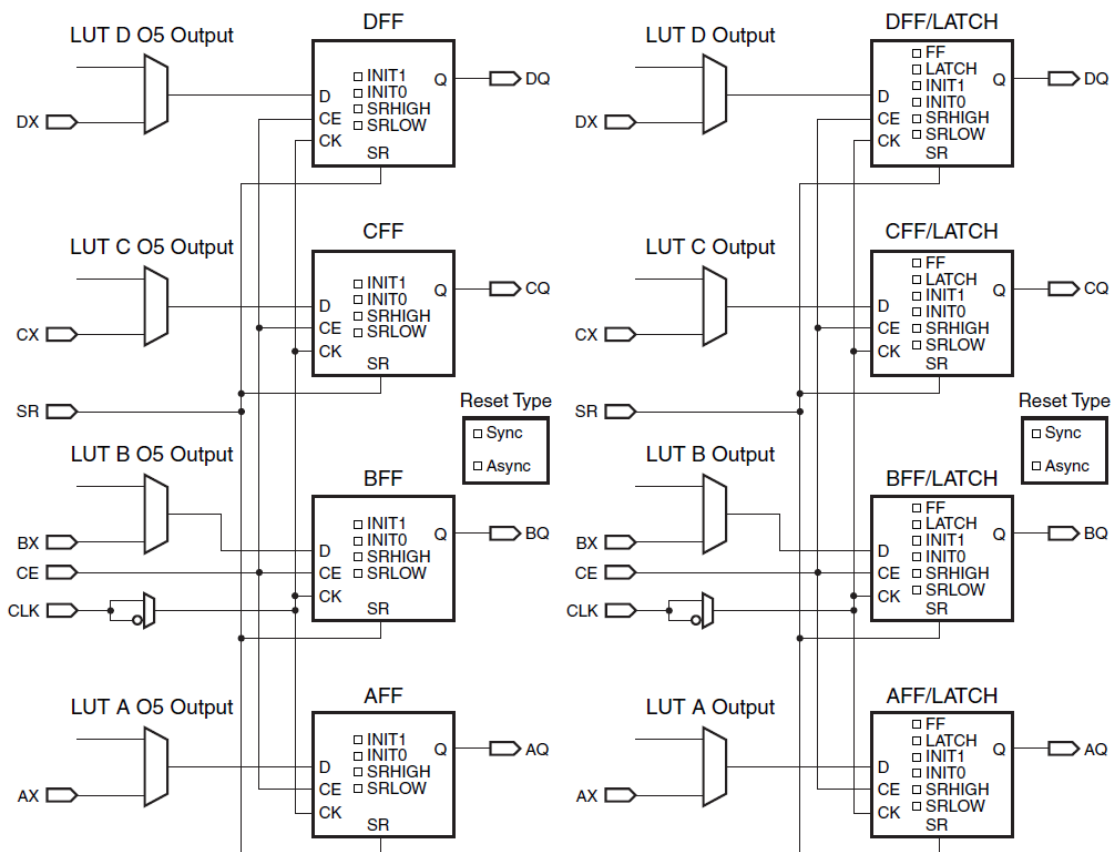
Inicializační možnosti pro klopné obvody jsou následující:

- SRLOW - synchronní nebo asynchronní RESET, jestliže je SR signál uplatněn
- SRHIGH - synchronní nebo asynchronní SET, jestliže je SR signál uplatněn
- INIT0 - asynchronní RESET při zapnutí, nebo globální SET/RESET
- INIT1 - asynchronní SET při zapnutí, nebo globální SET/RESET

Atributy SRHIGH a SRLOW mohou být nastaveny individuálně pro každý klopný obvod v řezu, nicméně zda bude daný set/reset synchronní (SYNC), nebo

asynchronní (ASYNC) individuálně nastavit možné není. Inicializační stav po konfiguraci, nebo globální inicializační stav je definován samostatnými atributy INIT0 a INIT1. Konfigurační možnosti pro SET a RESET klopných obvodů tedy jsou: [8]

- Ani SET ani RESET
- Synchronní SET
- Synchronní RESET
- Asynchronní SET (pre-set)
- Asynchronní RESET (clear)



Obr. 2.8: Dvě možné konfigurace KO v řezu [8]

2.4.3 Carry logika

Carry logika je logický obvod, jehož účelem je zlepšení výkonu provádění aritmetických operací. Ve skutečnosti lze aritmetické operace implementovat pouze pomocí LUT, nicméně použití carry logiky je efektivnější jak co se týče stupně integrace, tak i provozní rychlosti. Mimo to umožňují propojit přímo sousední LUT tak, aby nebylo nutné použít výrazně pomalejší běžnou lokální propojovací strukturu. [6, 9]

Logická buňka Xilinx FPGA obvodů 7. řady obsahuje dva samostatné obvody pro přenos bitů, které jsou navíc kaskádovatelné tak, aby bylo možné vytvořit více bitovou sčítačku/odčítačku. Pro každý bit je v řezu carry multiplexor (MUXCY) a dedikované XOR hradlo, které zajišťuje sčítání/odčítání operandů s vybranými přenosovými bity.

Zpoždění signálu ve sčítačce roste lineárně s počtem bitů. Obvody carry logiky mohou být implementovány spolu klopným obvodem ve stejném řezu. Kaskádování carry logiky je limitováno pouze výškou sloupců v FPGA. [8]

2.5 Bloková RAM

Kromě distribuované RAM a vysokorychlostních I/O pamětových rozhraní nabízí FPGA rodiny Artix-7 také velký počet blokových RAM (BRAM) o velikosti 36 kb. Každá bloková RAM obsahuje dvě nezávisle ovládané 18 kb RAM. Stejně jako ostatní elementy, i BRAM jsou na čipu soustředěné do sloupců. Blokové RAM jsou kaskádovatelné, což umožňuje implementaci paměti s širší datovou sběrnici a vyšším rozsahem adres s minimální časovou penalizací. Klíčové výhody blokové RAM jsou:

- Dvou-portová 36 kb RAM s šířkou portu až 72 bitů
- Programovatelná FIFO (First In, First Out) paměť
- Vestavěný obvod pro korekci chyb (detekuje jednobitovou a dvoubitovou chybu při čtení dat z RAM) [12]

2.6 DSP bloky

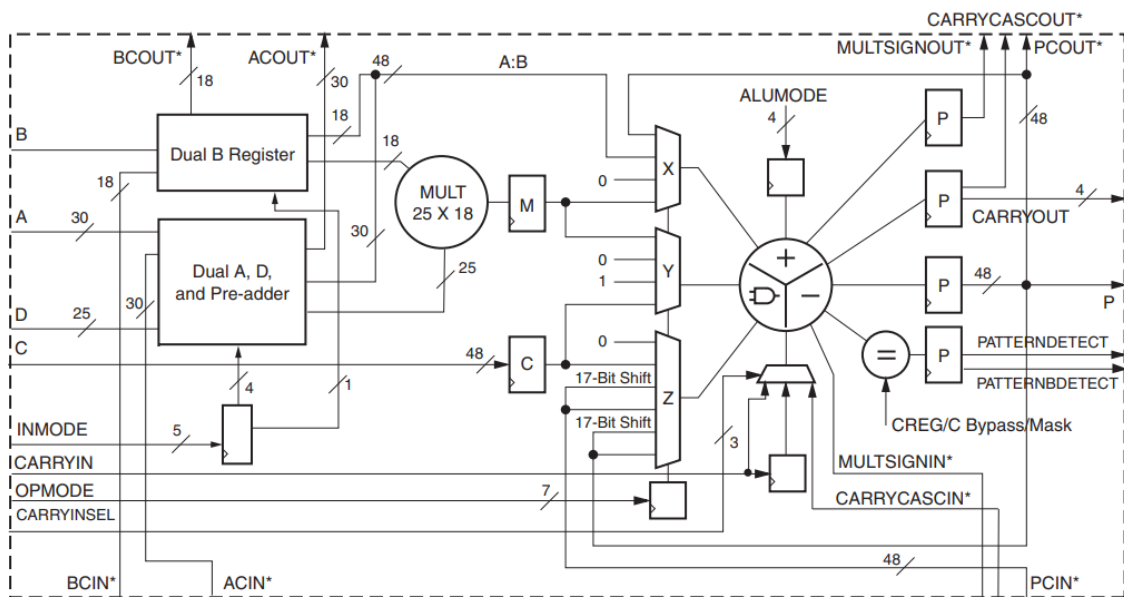
FPGA čipy jsou efektivní v aplikacích digitálního zpracování signálu (DSP; Digital Signal Processing), a to zejména proto, že umožňují implementovat vlastní a plně paralelní algoritmy. Aplikace pro digitální zpracování signálu vyžadují velké množství násobiček a akumulátorů, které je v FPGA nejlépe možné implementovat pomocí dedikovaných DSP bloků (obr. 2.9). K největším výhodám DSP bloků patří:

- Násobička $25 \times 18b$
- Akumulátor 48-bit
- Před-sčítačka
- Aritmetická jednotka SIMD (Single Instruction, Multiple Data)
- Generátor logických funkcí
- Volitelný pipelining
- Dedikované sběrnice pro kaskádování bloků

Každý DSP blok, v rodině Artix-7 označovaný DSP48E1, má dvou-vstupovou násobičku následovanou multiplexorem a tří-vstupovou sčítačkou/odsčítačkou/akumulátorem. Násobička má asymetrický vstup a akceptuje 18 bitový a 25 bitový operand ve dvojkovém doplňku. Následuje sčítačka/odčítačka, která pracuje se třemi 48 bitovými operandy také ve dvojkovém doplňku. V SIMD módu sčítačka/odčítačka podporuje navíc aritmetické operace s 2×24 bity a dvěma samostatnými CARRYOUT signály nebo se 4×12 bity a čtyřmi samostatnými CARRYOUT signály. SIMD mód umožňuje plně využít kapacitu ALU (Aritmeticko Logická Jednotka) při zpracování dat s menší přesností, než je bitová šířka ALU.

Pokročilejší funkce digitálního zpracování signálů jsou umožněny kaskádním zapojením jednotlivých DSP bloků. Skládání bloků je umožněno díky interním signálům (na obr. 2.9 označené hvězdičkou).

DSP bloky také obsahují prostředky pro konfigurovatelný pipelining. Pipelining je možné využít na vstupní operandy, mezivýsledky a výstupy akumulátorů. Někdy se také označuje jako zřetěžené zpracování a jeho cílem je rozdělit funkci do menších logických bloků. Výsledkem této operace je vyšší pracovní kmitočet, respektive větší propustnost dat. [13]



Obr. 2.9: Struktura bloku DSP48E1 [13]

3 Návrh vývojového kitu

Tato kapitola popisuje návrh samotného vývojového kitu spolu s programovacím zařízením. Jsou zde podrobně rozebrány zvolené periferie vývojového kitu a jejich základní vlastnosti. Také je zde nastíněna problematika návrhu desek plošných spojů s ohledem na výrobní technologická omezení a výsledná cena materiálu potřebného pro výrobu vývojového kitu.

3.1 Vývojový kit

Na místo hlavní komponenty tohoto kitu byl zvolen Xilinx FPGA obvod s označením XC7A200T-2FBG484I. Byl volen s ohledem na počet prostředků, kterými disponuje, dostupnou dokumentaci, technickou podporu a cenu. Jedná se o obvod vyrobený 28 nm technologií, jež poskytuje vysoký výkon v aplikacích, které jsou citlivé na cenu a spotřebu elektrické energie. Porovnání dalších možných alternativ FPGA obvodů a jejich základních prostředků je uvedeno v tab. 3.1. Všechny uvedené obvody jsou pinově kompatibilní, výsledný vývojový kit je tedy možné osadit kterýmkoli z nich. Mimo prostředky uvedené v tabulce, FPGA obvody také nabízí rozhraní pro PCIe komunikaci, 4 GTP (Gigabit Transceiver with Low Power) a duální 12 bitový A/D převodník se vzorkovacím kmitočtem 1 MHz a teplotním senzorem přímo na čipu [15].

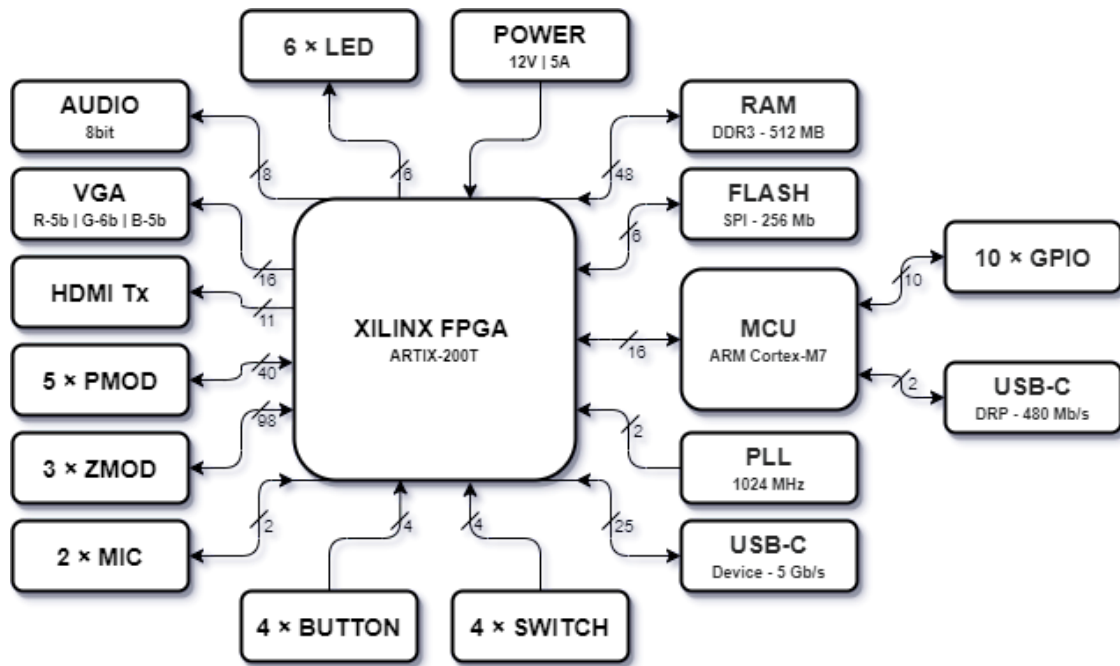
Blokový diagram vývojového kitu je znázorněn na obr. 3.1.

Tab. 3.1: Přehled základních prostředků zvoleného FPGA obvodu [15]

FPGA	Logické buňky	Konfigurovatelné logické bloky		DSP48E1 bloky	BRAM bloky		I/O piny	Cena [USD]
		Řezy	Max. distrib. RAM [kb]		36 kb	Max [kb]		
XC7A75T	75 520	11 800	892	180	105	3 780	285	37,5
XC7A100T	101 440	15 850	1 188	240	135	4 860	285	51
XC7A200T	215 360	33 650	2 888	740	365	13 140	285	90,2

3.1.1 Napájení

Jak je již zřejmé z kapitoly 2, FPGA je komplexní integrovaný obvod, jehož struktura se skládá z několika dílčích bloků (např. logické buňky, jádro čipu, pomocné



Obr. 3.1: Blokový diagram vývojového kitu

obvody, AD převodník, atd.). Každá tato část obvodu vyžaduje pro správnou funkčnost specifické napájení (velikost napětí, dodávaný výkon, dynamickou odezvu, zvlnění napětí, apod.). Navíc je také vyžadována určitá sekvence náběhu jednotlivých napájecích větví FPGA obvodu tak, aby se při zapnutí napájení vývojového kitu neobjevovaly nadměrné špičkové proudové odběry, a také aby nedošlo k poškození samotného FPGA čipu. Právě z těchto důvodů se na kitu nachází hned několik samostatných napájecích zdrojů, jejichž souhrn je v následující tab. 3.2. Napájení celého kitu je zajištěno barelovým napájecím konektorem o rozměrech 5,5/2,5 mm s nominálním napětím 12 V přivedeným na střední pin o maximálním výkonu 60 W.

Jednotlivé napěťové zdroje jsou v tabulce seřazeny seshora podle požadavků na počáteční napájecí sekvenci. Tato sekvence počíná napájením pro jádro čipu, pokračuje přes terminační a pomocné obvody a končí u napájení vstupně-výstupních bank. Na zbylá napájení, jako je například napájení AD převodníku, se požadavky na napájecí sekvenci nevztahují.

Výkony napájecích zdrojů, zejména výkon zdroje pro napájení jádra FPGA obvodu, byly dimenzovány podle největšího ze tří FPGA, které je možné na vývojový kit osadit. Přesné určení maximálního požadovaného výkonu je však v tomto případě téměř nemožné, neboť spotřeba FPGA obvodu je silně závislá na velikosti implementovaného designu a na operacích, které vykonává. Proto byl proveden kla-

Tab. 3.2: Souhrn napájecích zdrojů FPGA obvodu

Napájecí zdroje FPGA obvodu			
Komponenta	U_{NOM} [V]	I_{MAX} [A]	P_{MAX} [W]
Jádro, blok RAM	1,00	25	25,0
Transceiver, PLL	1,00	1,2	1,20
Terminální obvody	1,20	0,2	0,24
Pomocné obvody	1,80	4,0	7,20
IO banka 34, DDR3	1,35	1,2	1,62
IO banky 0, 13, 14, 15, 16, 35	3,30	4,0	13,2
AD převodník napájení	1,80	0,3	0,54
AD převodník reference	1,20	< 0,1	0,10

sifikovaný odhad maximálního požadovaného výkonu a na tomto základě následně zvolen vhodný napájecí zdroj pro jádro, který výkonové požadavky splňuje s rezervou větší než 20 %. Pro potřeby chlazení FPGA obvodu je možné na vývojový kit namontovat pasivní chladič i ventilátor s napájecím napětím 12 V a možností řízení otáček pomocí PWM (Pulse Width Modulation) regulace.

Napájecí zdroj pro jádro s označením *ADP1853* se nachází na DPS v blízkosti FPGA čipu pro co nejlepší transientní odezvu. Na rozdíl od ostatních zdrojů napětí pro FPGA je umístěný v samostatném pouzdře a spolu s přílehlými MOS tranzistory pro spínání vyšších proudů je od ostatních komponent odstíněn pomyslnou zdí prokovů. Toto odrušení slouží zejména pro spínací frekvence zdroje, které jsou v jednotkách MHz, ale také pro omezení vlivů elektromagnetických polí způsobených velkými a rychlými změnami odběru proudu. Dále se na DPS nachází integrovaný obvod s označením *ADP5052*, který vytváří požadované napájecí napětí pro zbylé části FPGA s výjimkou AD převodníku. Jedná se o 5kanálový napájecí zdroj se čtyřmi integrovanými DC-DC spínanými měniči a jedním lineárním regulátorem. Výhodou tohoto řešení je značná úspora místa i ceny, ale také jednodušší návrh zapojení. Pro zmíněné napájecí větve byl navíc implementován rezistor pro snímání proudu. Úbytek napětí na těchto rezistorech snímá zesilovač *INA199A2DCKR* a zesiluje jej se zesílením 100. Proudový odběr jednotlivých napájecích větví je tedy možné měřit přímo na vývojovém kitu pomocí multimetru. Velikost proudu tekoucího do jádra FPGA je také navíc možné zjistit integrovaným AD převodníkem na čipu. Tento AD převodník je napájen samostatným zdrojem s označením *ADP223* pro zajištění dostatečně čistého napětí. Referenční napětí převodníku je potom zajištěno precizní napětovou referencí s označením *LT6654*.

Dále se na kitu nachází pomocné 5 V napájení *MP2315*, regulátory s ultra nízkým

výstupním šumem pro napájení DA převodníku a externí PLL s označením *LT3045* a dva *LP5907* regulátory pro napájení mikrofonů a pomocné napájení PLL. Vybraná napětí jsou také vyvedena do konektoru a přístupná pro napájení externích zařízení.

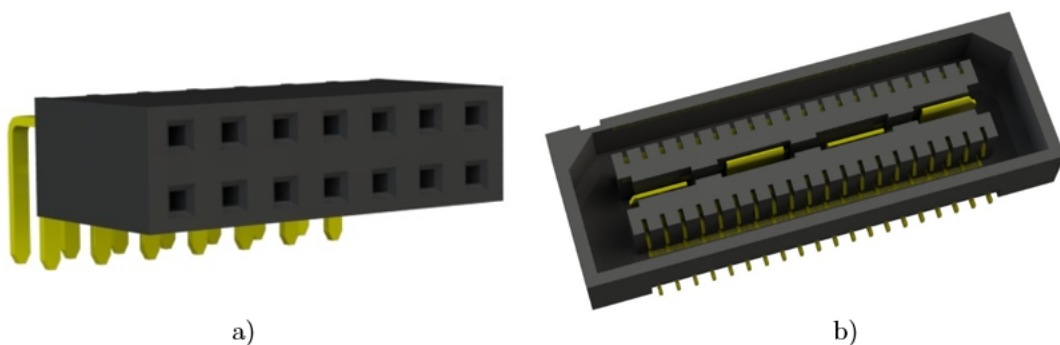
3.1.2 Propojovací komponenty

Vývojový kit je vybaven hned několika typy konektorů.

Prvním a nejpočetnějším typem jsou PMOD konektory. Jedná se o, pro vývojové kity, typické vstupně-výstupní konektory, které umožňují připojení libovolných externích zařízení. Jsou využitelné pro přenos signálů o frekvencích v řádu jednotek až desítek MHz, mohou poskytovat externím zařízením napájení 3,3 V a jejich pinout je kompatibilní s komerčně dostupnými vývojovými kity a zařízeními k nim již navržených. Na kitu jsou implementovány ve dvou variantách, a to jak pro přenos signálů vyžadující jednostranně zakončené linky, tak pro přenos signálů vyžadujících diferenciální linky. Ke konektorům s implementovanými jednostranně zakončenými linkami jsou přidány sériově zapojené rezistory jako ochrana vstupně-výstupní buňky FPGA čipu při zkratu na externím zařízení. Navíc jeden konektor typu PMOD byl vyhrazen pro diferenciální linky vyvedené z AD převodníku integrovaném na čipu FPGA, tudíž je možné připojit i externí zařízení s analogovým výstupem. Podoba tohoto typu konektoru je vyobrazena na obr. 3.2 a).

Druhým typem vstupně-výstupních konektorů jsou konektory označovány jako SYZYGYY, někdy Mezzanine, nebo také ZMOD. V tomto případě se jedná o konektor výrobce Samtec s označením *QSE-020-01-F-D-A*. Tyto konektory jsou na kitu umístěny za účelem přenosu vysokorychlostních signálů, a to diferenciálními linkami až 14 GHz, jednostranně zakončenými linkami až 9 GHz. Na pinech těchto konektorů jsou vyvedeny vstupně-výstupní piny z FPGA čipu schopné akceptovat hodinový signál, klasické vstupně-výstupní piny, ale také hodinový signál z externí PLL a napájení 5 V a 12 V. Jeden ze tří těchto konektorů umístěných na vývojovém kitu je napojen na GTP banku FPGA obvodu. Jedná se o signály připojené ke gigabitovým transceiverům v FPGA, což umožňuje připojení externích zařízení využívající komunikační standardy jako jsou PCI-e, SATA, apod. Datová propustnost těchto konektorů je omezena maximální frekvencí, se kterou zvládne FPGA obvod operovat a počtem diferenciálních linek. První dva konektory obsahují maximálně 16 diferenciálních linek a frekvence přenosu dat je limitována FPGA obvodem na 628 MHz, z čehož plyne maximální datová propustnost zhruba 10 Gb/s. Zbývající vysokorychlostní konektor napojený na GTP banku obsahuje 4 diferenciální linky pro vysílání dat, 4 diferenciální linky pro příjem dat a v tomto případě je frekvence omezena gigabitovými transceivery na 3,125 GHz, z čehož plyne datová propustnost duplexně až 50 Gb/s. Podoba tohoto typu konektoru je vyobrazena na obr. 3.2 b).

Dále se na kitu nacházejí dva konektory typu USB-C (Universal Serial Bus). První z nich je implementován v konfiguraci DRP (Dual Role Port), což znamená, že může fungovat jako USB device i jako USB host. Komunikaci mezi konektorem a FPGA obvodem zajišťuje procesor *MIMXRT1011* a maximální přenosová rychlost tohoto USB konektoru je 480 Mb/s. Druhý USB-C konektor je možné použít pouze v konfiguraci USB device. Komunikaci mezi tímto konektorem a FPGA obvodem zajišťuje čip *FT600Q* a maximální přenosová rychlost je 5 Gb/s. Motivací pro implementaci obou těchto konektorů byla v prvním případě flexibilita připojení externích zařízení využívajících USB standard, jako je například myš, klávesnice, nebo flash disk, v druhém případě byl kladen důraz na vysokorychlostní přenosy dat z počítačů a jiných zařízení využívajících USB standard.



Obr. 3.2: Příklad zvolených vstupně-výstupních konektorů

3.1.3 Paměťové a výpočetní komponenty

Spolu s FPGA obvodem jsou na kitu umístěny také externí paměti (Flash paměť a RAM) a procesor.

V případě flash paměti se jedná o sériovou NOR paměť o velikosti 256 Mb a maximální rychlostí 133 MHz. Slouží zde primárně pro načtení a uchování programovacího bitstreamu a následné konfiguraci FPGA obvodu. Velikost paměti byla určena s ohledem na maximální velikost bitstreamu, což je v případě voleného FPGA obvodu 128 Mb.

Další paměť je RAM typu DDR3 s označením *AS4C256M16D3LC-10BCN*. Tato paměť je konfigurována jako 256 Mb \times 16 b, z čehož plyne celková kapacita 4 Gb. Maximální frekvence hodinového signálu paměti je 933 MHz, nicméně FPGA obvod podporuje paměťové rozhraní o maximální frekvenci 400 MHz. Z uvedených údajů poté vyplývá maximální možná datová propustnost, a to 12,8 Gb/s.

Procesor s označením *MIMXRT1011DAE5A* je založený na 32 bitové ARM architektuře. Jádrem procesoru je Cortex-M7 s pracovní rychlostí až 500 MHz. Mezi

výhody tohoto procesoru patří zejména RAM paměť o velikosti 128 KB implementovaná přímo na čipu, integrované napájecí zdroje, které redukuje plochu na DPS a zjednodušují návrh a v neposlední řadě množství komunikačních protokolů, jako je UART (Universal asynchronous receiver-transmitter), SPI (Serial Peripheral Interface), I²C (Inter-Integrated Circuit), USB a další. Procesor je s FPGA obvodem propojený 16bitovou sběrnicí. Vzhledem k flexibilitě vstupně-výstupních pinů procesoru (FlexIO piny) je tato sběrnice překonfigurovatelná na několik odlišných komunikačních standardů. Z procesoru je také vyvedeno několik vstupně-výstupních pinů do hřebínkového konektoru na vývojovém kitu. Jeho výpočetní výkon je tak lehce přístupný uživateli a rozšiřuje možnosti a flexibilitu vývojového kitu jako takového.

3.1.4 Audio a video komponenty

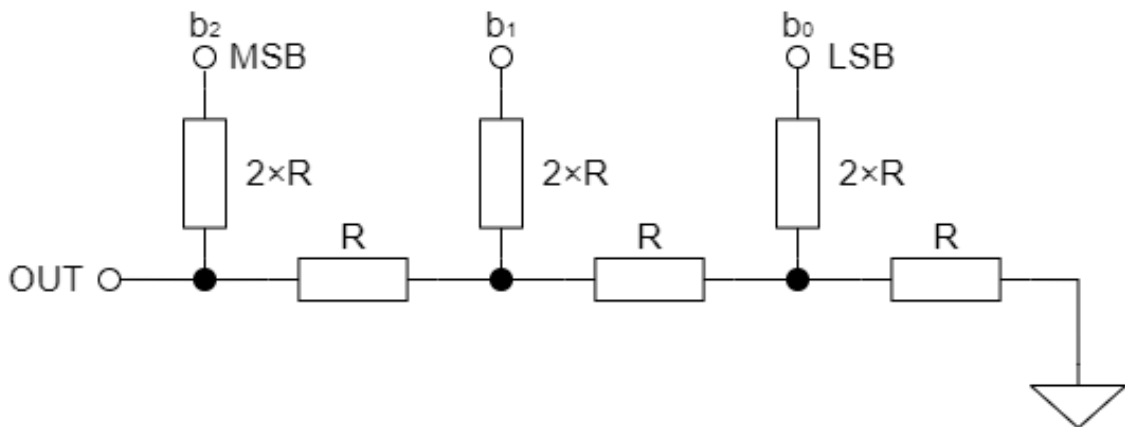
Na kitu jsou také k dispozici komponenty pro audio vstup i výstup a video výstup.

Za účelem zaznamenávání zvuku jsou na kitu dva mikrofony s označením *IMP34D T05TR*. Tyto mikrofony jsou vyrobeny technologií MEMS (Micro Electro Mechanical Systems). MEMS systémy jsou takové systémy, jež sdružují mikroelektronické komponenty spolu s mikromechanickými komponenty o velmi malých rozměrech (řádově jednotky až desítky μm). Předností těchto mikrofonů jsou velmi malé rozměry, dynamický rozsah výstupního signálu až 95dB a také nízké zkreslení výstupního signálu. Další výhodou zvolených mikrofonů je výstup v podobě PDM (Pulse Density Modulation), což redukuje počet obsazených pinů v FPGA obvodu.

Pro audio výstup je na kitu zařazena klasická 3,5mm zásuvka pro jack konektor. DA převodník pro transformaci digitálního signálu z FPGA na analogový signál byl navržen diskrétně. Jedná se o DAC pole tvořené jednoprvkovými elementy. Každý tento element je tvořen klopným obvodem typu D, přesnými rezistory a filtračními kondenzátory. Jednotlivé klopné obvody jsou řízeny výstupem modulátoru navrženém v FPGA a synchronizovány hodinovým signálem generovaným v precizní PLL. Tyto klopné obvody transformují napěťovou úroveň signálu z FPGA, která je 3,3 V, na 5 V. Napájení těchto klopných obvodů je vytvořeno z precizní reference, která poskytuje stabilní napájení bez rušení. Rezistory sériově zapojené na výstup klopných obvodů slouží jako převodník napětí na proud. Jednotlivé výstupy jsou spojeny paralelně a přivedeny na vstup operačních zesilovačů, které slouží jako převodník proudu na napětí. Vzhledem k tomu, že operační zesilovač udržuje na svých vstupech napěťovou nulu, tak úbytek napětí na precizních rezistorech v případě logické úrovně '1' na výstupu klopného obvodu, bude vždy stejný. Tímto dvojnásobným převodem a zajištěním konstantního úbytku napětí na rezistorech dojde k eliminaci nežádoucího vlivu napěťového koeficientu rezistorů na přenos signálu. Paralelně s rezistory jsou

filtrační kondenzátory, které slouží ke svodu vysokofrekvenčního rušení do země. Na výstupu DA převodníku je zařazen filtr typu dolní propust 2. řádu s vícenásobnou zpětnou vazbou, jehož mezní kmitočet byl volen jako kompromis mezi filtrací vysokofrekvenčních signálů a neovlivňováním užitečných signálů v rámci audio pásma. Diskrétní návrh audio DA převodníku byl převzat z [21].

Video výstup může být zajištěn dvojím způsobem. Na kitu je za tímto účelem implementovaný VGA konektor a HDMI konektor. HDMI je oproti VGA modernější digitální standard, který kromě přenosu obrazu umožňuje i přenos zvuku, nicméně jeho implementace do FPGA je značně náročnější. VGA je naopak analogový standard, a proto je nutné spolu s ním implementovat také DA převodník. Lépe řečeno jsou potřeba hned tři DA převodníky, pro každou z barev RGB jeden. Jejich bitové rozlišení je dáno VGA standardním rozložením barev jako R-5, G-6, B-5, kde číslice vyjadřuje právě počet bitů na barvu. Tyto převodníky byly také navrženy diskrétně a jejich topologie je označována jako R2R. Jedná se o rezistorový žebřík, kde napěťový příspěvek každého bitu je určen jeho pozicí vůči výstupu. Bit, který je nejbližší výstupu převodníku má největší váhu, tzv MSBit (Most Significant Bit). Příklad 3bitového R2R DA převodníku je znázorněn na obr. 3.3.



Obr. 3.3: Příklad zapojení R2R DA převodníku

3.1.5 Ostatní komponenty

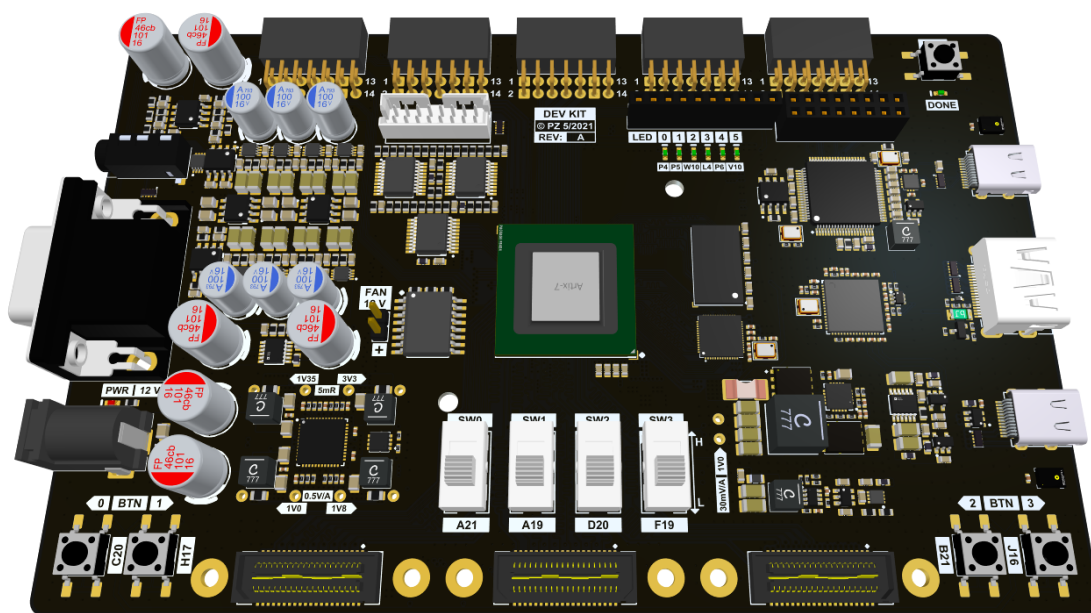
Mimo již zmíněné komponenty je kit také doplněn o precizní PLL s označením *Si5341A* pro generování čistých hodinových signálů o kmitočtech až 1028 MHz, čtyři klasické přepínače, čtyři tlačítka a šest signalizačních LED. Tyto prvky, s výjimkou PLL, slouží zejména pro testování nebo základní řízení implementovaných logických obvodů.

3.1.6 Parametry DPS a cena vývojového kitu

Návrh desky plošných spojů byl zvolen na 6 vrstev. Zvolit 6vrstvou DPS bylo nutné zejména kvůli velkému počtu vývodů FPGA čipu na malé ploše (BGA pouzdro se 484 vývody a roztečí 1 mm) a také RAM čipu (BGA pouzdro s 96 vývody a roztečí 0,8 mm). Mimo složitost zmíněných čipů hrála roli také komplexnost zapojení a rychlost přenášených signálů. Z těchto 6 vrstev jsou 3 vrstvy signálové, popřípadě napájecí, a 3 vrstvy referenční. Minimální rozměry dovolené výrobní technologií jsou 0,1 mm pro šířku vodivé cesty a poměr 0,2/0,4 mm pro vytvoření vodivého prokovu. Materiály DPS byly zvoleny tak, aby umožňovaly vytvořit vodivé cesty s kontrolovanou impedancí a tak dosažení optimálního přizpůsobení přenosu signálů. Většina jednostranně zakončených signálů na DPS má charakteristickou impedanci 50 Ω a většina diferenciálních linek má charakteristickou impedanci 100 Ω , nicméně nachází se zde výjimky jako např. diferenciální signály USB konektoru, jejichž charakteristická impedance je 90 Ω . Při návrhu byla požadovaná impedance určena šířkami vodivých cest, v případě diferenciálních párů i jejich vzdáleností mezi sebou, a také vzdáleností referenční plochy.

Další kritickou oblastí při návrhu kitu bylo délkové přizpůsobení vodičů přenášejících signály na sobě závislé (tzv. length matching). Tuto operaci bylo nutno provést u všech vysokorychlostních periférií, jako je RAM, USB-C 3.0 a vysokorychlostní vstupně-výstupní konektory. Délkové přizpůsobení vodičů se provádí za účelem dosažení stejného propagačního zpoždění při průchodu signálů, které jsou vysílány a čteny synchronně. Například u RAM se jedná o skupinu 8 bitů, která tvoří jeden datový byte, a hodinový signál k nim přidružený. Všechny tyto signály musí být navrženy s velmi podobným časovým zpožděním, aby nedošlo k chybnému čtení dat při vyšších frekvencích. V případě tohoto kitu byly jednotlivé signály přizpůsobeny s časovým rozdílem v jednotkách ps [piko sekund], čehož bylo dosaženo délkovým přizpůsobením vodičů v desetinách, maximálně jednotkách mm [mili metrů]. Výsledný návrh DPS vývojového kitu je na obr. 3.4 a rozložení jednotlivých vrstev DPS v příloze B

Soupis jednotlivých komponent včetně kusů použitých na DPS a jejich cena je v příloze A. Je potřeba zmínit, že důsledkem celosvětové pandemie a tím zapříčiněným omezením výrobních kapacit se cena a dostupnost elektronických komponent časem zásadně mění, z toho důvodu se může stát, že uváděné ceny nejsou aktuální. Výsledná cena jednoho vývojového kitu vychází z ceny a počtu kusů součástek potřebných na deset vývojových kitů a je rovna 8 057 CZK. Jestliže chceme porovnat tento vývojový kit s kity již dostupnými na trhu od výrobců jako je Diligent nebo Xilinx, je potřeba si jich nejprve několik představit. Nabízí se porovnání se



Obr. 3.4: 3D model vývojového kitu

dvěma kity, a to s kitem označeným *USB104 A7* [19] od výrobce Digilent a kitem *AC701 Evaluation Kit* [20] od výrobce Xilinx. První jmenovaný se skládá z FPGA XC7A100T-1CSG324I, které je v porovnání s FPGA na našem vývojovém kitu méně výkonné, jednoho vysokorychlostního konektoru, třech PMOD konektorů a DDR3 paměti. Tento kit nabízí značně menší flexibilitu i výpočetní výkon a jeho cena je v přepočtu 7 315 CZK. Druhý jmenovaný se skládá z FPGA XC7A200T-2FBG676C, které je v porovnání s FPGA na našem vývojovém kitu stejně výkonné, ale disponuje více vstupně-výstupními piny. Dále obsahuje několik vysokorychlostních konektorů a paměťových uložišť (např. FMC konektor, gigabitový ethernet port, SFP konektor, konektor pro SD kartu, DDR3 paměť, apod.). Využití tohoto kitu je patrně cílené na výpočetně náročné aplikace a vzhledem k absenci základních vstupně-výstupních konektorů je nevhodný na běžné použití. Cena tohoto kitu je v přepočtu 27 143 CZK. Z tohoto porovnání vyplývá, že náš vývojový kit na trhu potenciálně zaujímá prázdné místo a nabízí velkou flexibilitu a dostatečný výpočetní výkon, při zachování kompaktních rozměrů, za příznivou cenu.

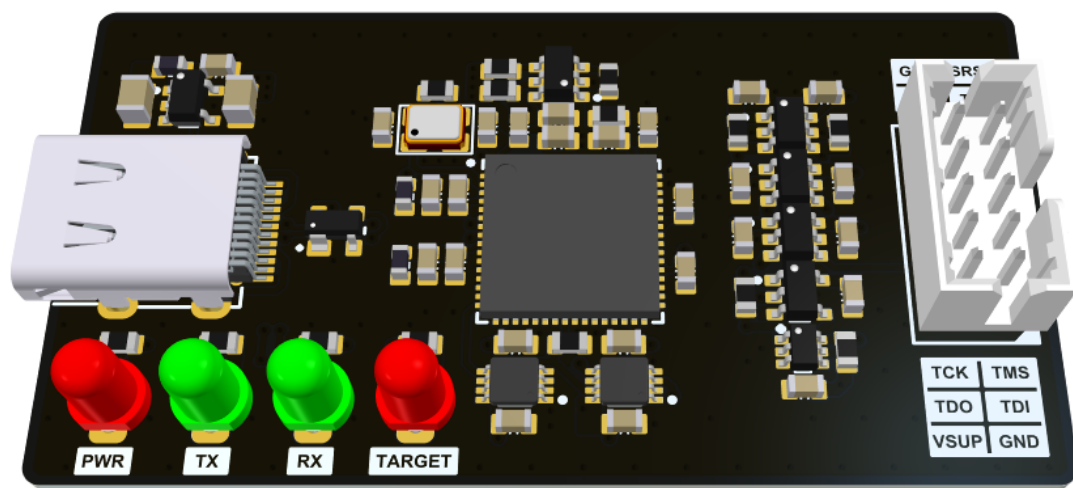
3.2 Programovací zařízení

Programovací zařízení slouží ke zprostředkování přenosu programovacího souboru (angl. bitstreamu) z počítače do FPGA obvodu. Pravděpodobně nejpoužívanějším

rozhraním pro přenos těchto dat je standard JTAG (Joint Test Action Group) a v tomto případě tomu není jinak.

Návrh programovacího zařízení není obsahem bakalářské práce, nicméně dostupné programovací moduly na trhu, ať už přímo od výrobce Xilinx nebo Digilent, svou cenou neodpovídají složitosti zařízení. Z tohoto důvodu bylo rozhodnuto o návrhu vlastního a separátního programovacího modulu. Návrh vychází z práce akademických pracovníků univerzity Českého Vysokého Učení Technického v Praze, kteří úspěšně reverz inženýrovali modul od výrobce Digilent s označením JTAG-SMT3-NC [17].

Základem programovacího zařízení je integrovaný obvod FT2232H. Tento čip je nakonfigurovaný tak, aby přijímal data z počítače pomocí USB rozhraní a konvertoval je na JTAG a UART standardy. Dále se na zařízení nachází EEPROM, jejíž správná konfigurace je nutná pro komunikaci programovacího zařízení s návrhovým prostředím Xilinx Vivado nebo Xilinx ISE. Problematika modifikace těchto konfiguračních dat je více rozebrána v literatuře [17]. V neposlední řadě je zařízení doplněno o překladače napětových úrovní pro všechny signály JTAG i UART standardu. Referenčním napětím je v tomto případě napětí z FPGA kitu o velikosti 3,3 V. Komponenty, hlavně tedy typ výstupního konektoru a THT LED, byly voleny s ohledem na možné umístění zařízení do krabičky zhotovené technologií 3D tisku. Na obr. 3.5 je 3D model navrženého programovacího zařízení.



Obr. 3.5: 3D model programovacího zařízení

4 Osazení a oživení vývojového kitu

V této kapitole je ukázáno a popsáno, jakým způsobem probíhalo osazení a zapájení vývojového kitu a jaký byl zvolený technologický postup. Také je zde popsán proces oživení a chyby, se kterými bylo potřeba se vypořádat.

4.1 Volba pájecího procesu

Volba správného pájecího procesu byla kritická zejména z důvodu dosažení spolehlivého pájeného spoje u BGA pouzder komponent FPGA a RAM. Spolu s deskami plošných spojů byla vyrobena i šablona pro tisk pasty (angl. stencil), pomocí které je možné nanést pájecí pastu přesně na jednotlivé pájecí pady na DPS.

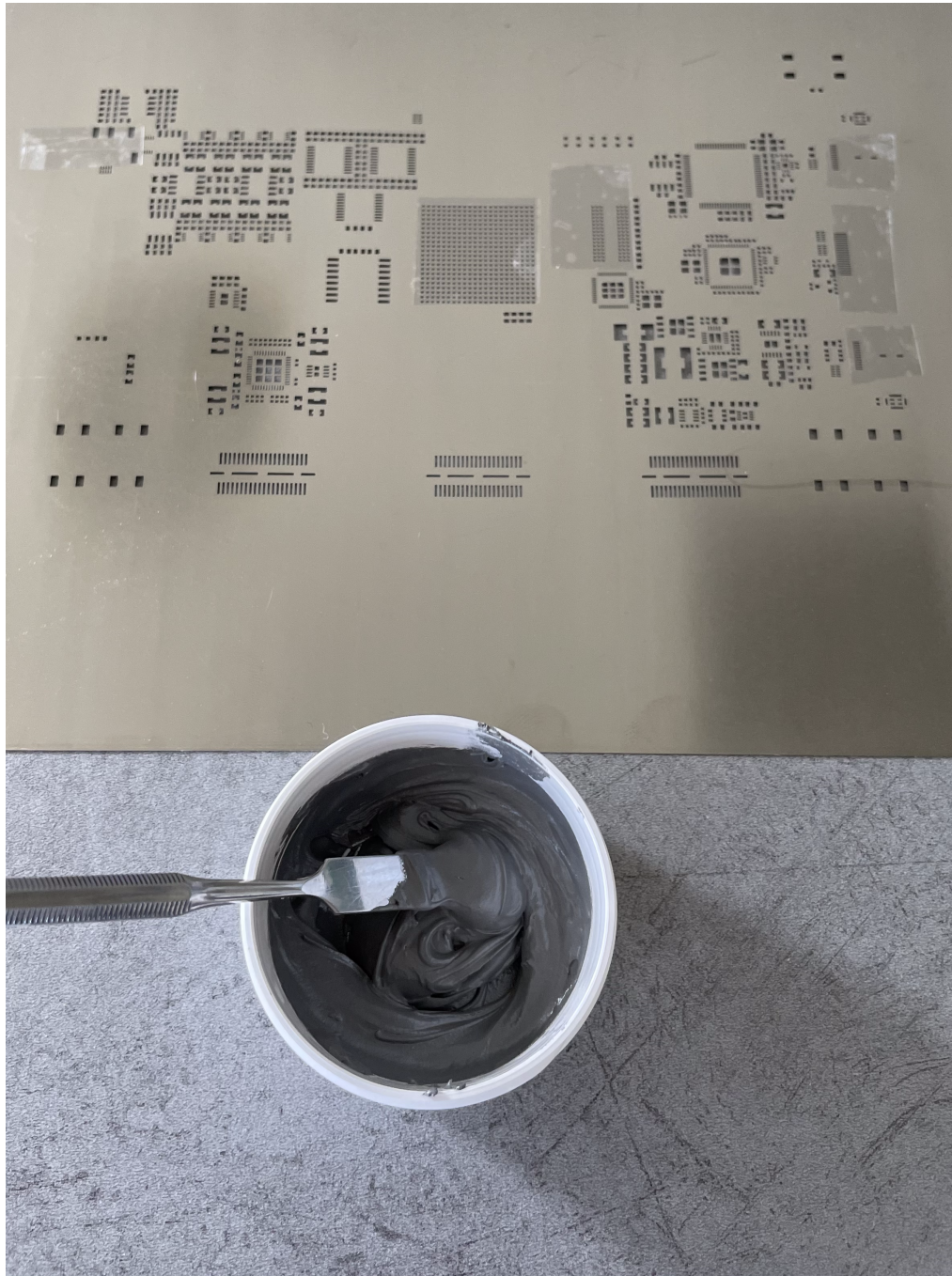
Na základě uvedených informací přicházely do úvahy tři způsoby pájení. Prvním z nich bylo přetavení pájecí pasty pomocí tzv. plotýnky (hot-plate technologie), druhým způsobem bylo přetavení pájecí pasty v přetavovací peci a posledním způsobem braným do úvahy bylo pájení v parách. U první jmenované technologie by prohřátí celé DPS a spolehlivé zapájení vyžadovalo příliš dlouhou dobu nebo příliš vysokou teplotu, čímž by došlo k nadměrnému mechanickému i tepelnému namáhání DPS, případně až k delaminaci DPS, a tedy nespolehlivému zapájení, nebo k poškození DPS. Naopak poslední jmenovaný způsob, pájení v parách, by bylo vhodnou volbou, neboť dochází k přenosu tepla rovnoměrně na celé DPS bez závislosti na barvě komponent a DPS samotné. Nicméně kapaliny, které se za tímto účelem využívají, jsou nákladné. Právě z těchto důvodů bylo zvoleno pájení přetavením v přetavovací peci. Nevýhodou tohoto způsobu při kusové výrobě je nemožnost kontrolovat usazení součástek a korigovat jejich případné vychýlení ze svých pozic. Výhodou je možnost nastavení přesně definovaného teplotního profilu a spolehlivost pájeného spoje.

4.1.1 Nanesení pájecí pasty

Pájecí slitina použitá pro zapájení vývojového kitu byla volena eutektická Sn63/Pb37 od výrobce Chip Quik s označením TS391AX50. Olovnatá slitina byla vybrána s ohledem na spolehlivost pájeného spoje a nižší teplotu přetavení, než kterých dosahují bezolovnaté slitiny. V tomto případě je použití olovnaté pájky možné z důvodu výroby prototypu a využití zařízení pouze ke studijním a výzkumným účelům.

Samotné nanesení pájecí pasty bylo provedeno přes šablonu vyrobenou na míru DPS vývojového kitu. Tato šablona pro tisk pasty, jak je možné vidět na obr. 4.1, má určitá místa přelepena izolační páskou. Jedná se zejména o BGA komponenty, které před pájením byly osazeny pouze do tavidla, neboť na těchto pouzdrech se již nacházejí pájkové kuličky a není zcela nutné přidávat další pájecí slitinu. Druhým

případem jsou komponenty, které jsou citlivé na vyšší teploty, a proto by mohlo dojít v přetavovací peci k jejich poškození. Následně došlo k pevnému uchycení DPS do přípravku, na kterou byla poté zarovnána šablona pro tisk pasty tak, aby otvory v ní vytvořené byly sesouhlaseny s pájecími pady na DPS. Na zarovnanou a rovněž uchycenou šablonu byla nanесena pájecí pasta a následně stěrkou vtlačena do jednotlivých otvorů. Detailní pohled na DPS s nanесenou pájecí pastou je zobrazen na obr. 4.2 a).



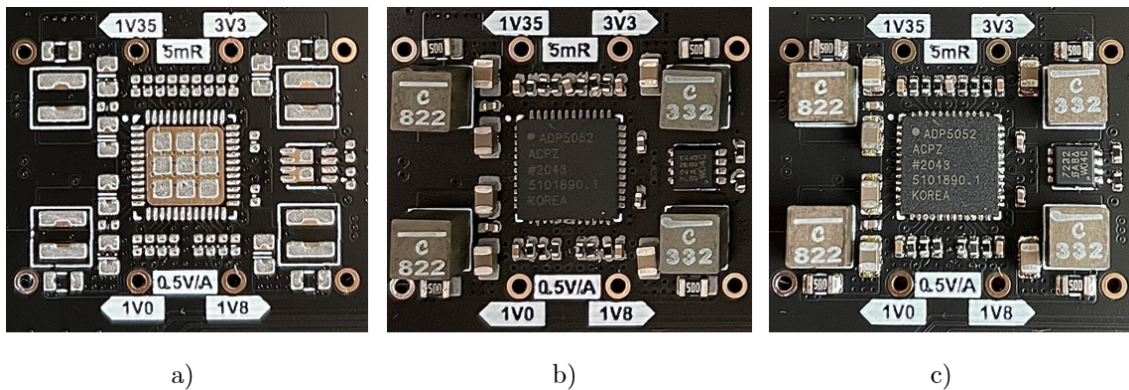
Obr. 4.1: Šablona a pájecí pasta

4.1.2 Osazení komponent

Po správném nanesení pájecí pasty na pájecí pady DPS následovalo osazení jednotlivých komponent. Většina součástek, od pasivních komponent v pouzdech 0402 (v palcích) a 0603 (rovněž v palcích), přes SOT pouzdra až po QFN pouzdra byla osazena ručně za pomoci pinzety, případně mikroskopu. Umístění komponent s BGA pouzdem bylo provedeno na zařízení Fritsch Mikroplacer. Toto zařízením nám umožnilo uchopit komponentu (díky nástroji, který po přiložení na komponentu vytvoří podtlak) a také ji přesně sesouhlasit s pájecími pady na DPS pomocí kamery, která je taktéž součástí zařízení. Po sesouhlasení byly součástky položeny na DPS, kde již bylo nanášeno speciální bezoplachové tavíadlo pro pájení BGA komponent s označením SMF-08 od výrobce Shenmao Technology. Detailní pohled na DPS s osazenými komponentami je na obr. 4.2 b).

4.1.3 Pájení

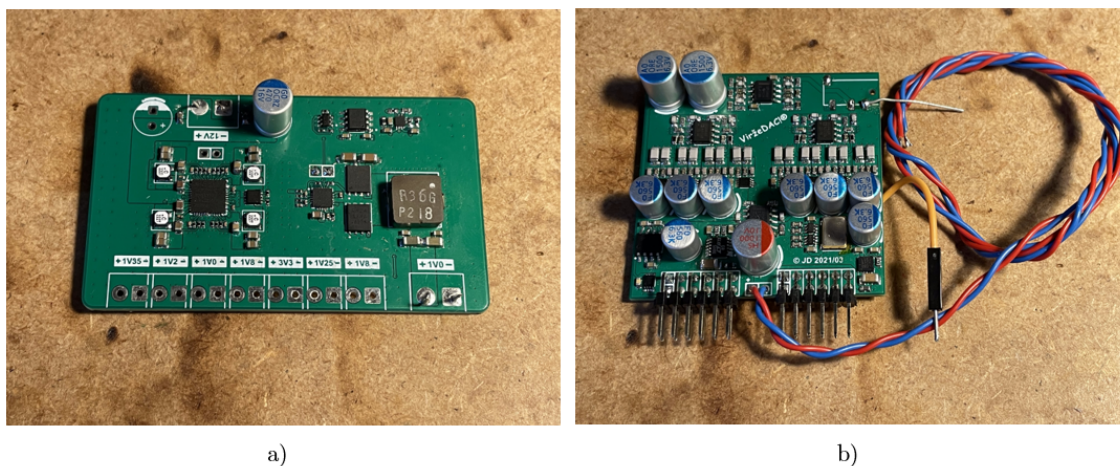
Po osazení komponent na DPS následovalo přetavení pájecí pasty v přetavovací peci. Za tímto účelem byla využita přetavovací pec na ústavu Mikroelektroniky s označením RO300FC výrobce essemtec. Přestože byla použita olovnatá pájecí slitina, tak pájecí profil v peci byl nastaven na typické hodnoty pro bezolovnatou pájecí slitinu, a to právě kvůli vývodům BGA pouzder, které jsou vyráběny z bezolovnatých slitin. Zbylé komponenty a spodní strana DPS byly zapájeny ručně. Detailní pohled na zapájenou DPS je na obr. 4.2 c).



Obr. 4.2: Pájecí proces: a) natištění pájecí pasty, b) osazení komponent, c) přetavení pájecí pasty

4.2 Oživení a testování vývojového kitu

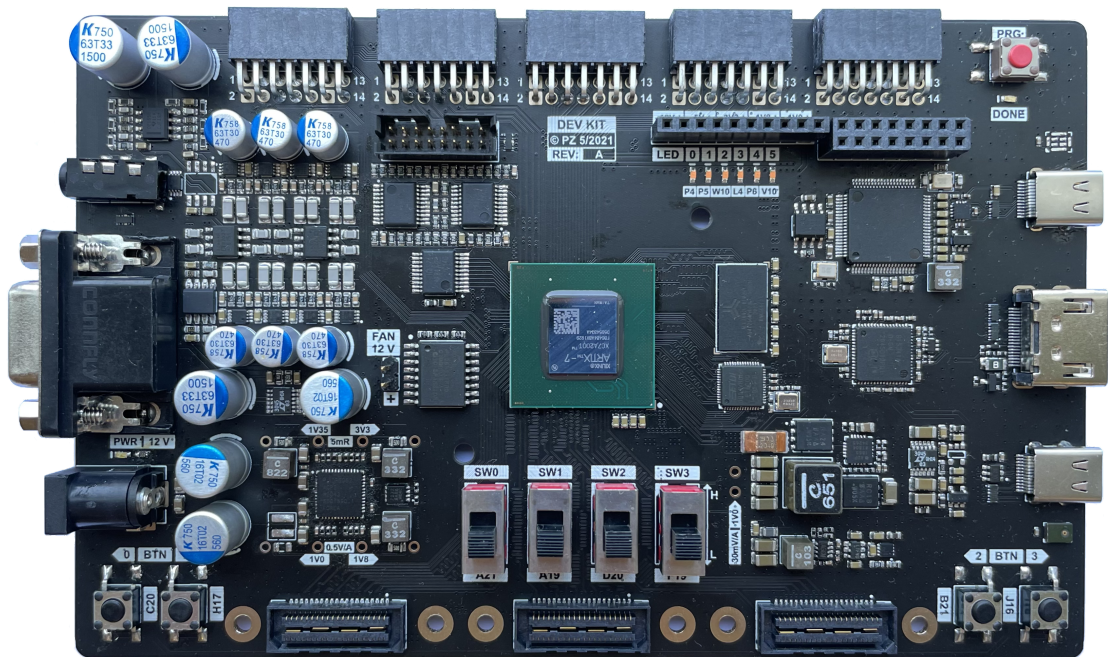
Během návrhu vývojového kitu byly separátně vytvořeny desky plošných spojů za účelem ověření funkcionality kritických nebo komplikovaných částí vývojového kitu. Testování proběhlo v průběhu návrhu vývojového kitu, aby bylo možné výsledný návrh upravit a odstranit tak případné chyby. K testovaným částem patřily napájecí zdroje pro FPGA obvod, tedy integrované obvody s označením *ADP5052*, *ADP1853* a *ADP223* a také diskretní audio DA převodník. Na základě výsledků testů byla poté v návrhu vývojového kitu upravena například velikost napájení části čipu *ADP5052* a napájecí sekvence jednotlivých zdrojů integrovaných na tentýž čipu, ale také hodnoty kompenzačních kondenzátorů a rezistoru u čipu *ADP1853*. K DA převodníku byly přidány klopné obvody, čímž došlo ke zmenšení zatížení výstupních buněk FPGA obvodu a také k odstranění jitteru způsobeného FPGA. Přípravky, na nichž byla daná zapojení testována, jsou vyfoceny na obr. 4.3.



Obr. 4.3: Fotografie testovacích přípravků: a) napájecí zdroje, b) audio DA převodník

Samotné oživení proběhlo standardně, kdy nejprve byly proměřeny jednotlivé napájecí větve, zda-li se někde po pájení nevyskytuje zkrat. Následně bylo připojeno napájecí napětí ze zdroje s proudovým omezením a měřeny napěťové hodnoty na výstupech napájecích zdrojů. Prvním problémem, který se objevil, byla nefunkčnost žádného ze zdrojů integrovaných na čipu *ADP5052*. Tento problém byl způsobený nesprávně zapájeným rezistorem R_{13} , který slouží k nastavení spínací frekvence tranzistorů uvnitř zdrojů (viz příloha B). Z pohledu integrovaného čipu tento odpor slouží k nastavení velikosti referenčního proudu proudového zrcadla, jehož výstup je použit k nabíjení kapacit uvnitř čipu. Jedná se o dvě protichůdně nabíjené kapacity, jejichž nabíjení a vybíjení způsobuje překlopení komparátoru. Oscilující výstup komparátoru poté představuje spínací frekvenci tranzistorů, a proto se tato chyba

projevila v nefunkčnosti zdrojů. Další chybou v tentýž napájecím zdroji byla zacyklená napěťová sekvence, což mělo za následek nefunkčnost většiny zdrojů, kromě zdroje s výstupem 1 V. Dále byly objeveny chyby způsobené při osazování, jako otočená LED, ale také chyby při návrhu, jako špatně zapojená tlačítka. Všechny zmíněné chyby bylo možné eliminovat přímo na DPS za použití mikrodrátku, přepájením komponent, nebo uštípnutím vývodových nožiček. Fotografie výsledného funkčního prototypu vývojového kitu je na obr. 4.4.



Obr. 4.4: Fotografie funkčního prototypu vývojového kitu

Závěr

V bakalářské práci byly rozčleněny programovatelné logické obvody do tří kategorií – SPLD, CPLD a FPGA. Každá z těchto kategorií byla popsána a byly nastíněny výhody a nevýhody jednotlivých zařízení. Způsoby implementace logických funkcí do zmíněných zařízení byly reprezentovány na praktické ukázce při implementaci konkrétní logické funkce. Odlišné způsoby implementace logické funkce se odvíjejí od strukturálního uspořádání obvodu a od programovací technologie, pro kterou byly obvody navrženy. Různé typy programovacích technologií byly rovněž popsány a byly diskutovány možné případy jejich využití, výhody a nevýhody.

Dále byla rozebrána architektura Xilinx FPGA obvodu a jeho nejdůležitější bloky. Tyto obvody poskytují velice rozsáhlé pole možností při implementaci logických funkcí. Pochopení architektury a funkcionality FPGA obvodu je nezbytné pro správný popis a optimalizaci logických funkcí v některém z návrhových prostředí (Xilinx Vivado nebo Xilinx ISE) a pro využití prostředků, které vývojový kit nabízí.

Pro realizaci vývojového kitu byl zvolen FPGA obvod s označením XC7A200T-2FBG484I, a to s ohledem na počet prostředků, kterými disponuje, dostupnou dokumentací, technickou podporu a cenu. Periferie kitu byly voleny s důrazem na univerzálnost kitu a na možnost použití v odlišných aplikacích. K vývojovému kitu bylo navíc separátně navrženo programovací zařízení, které je možné použít pro jakékoliv moderní Xilinx FPGA obvody. Návrhem tohoto programátoru byly značně sníženy výsledné náklady na výrobu a používání vývojového kitu. Výsledkem bakalářské práce je funkční prototyp vývojového kitu s FPGA obvodem.

Navazující práce na vývojovém kitu budou v podobě důkladného otestování veškerých periférií a vytvoření základních VHDL modulů pro každou periférii tak, aby došlo ke zjednodušení jakékoliv budoucí implementace. Podnětem pro vylepšení vývojového kitu je oprava chyb v návrhu, které byly objeveny a případné vytvoření druhé revize vývojového kitu.

Literatura

- [1] Brown, Stephen & Rose, Julie (2002). *Architecture of FPGAs and CPLDs: A tutorial*. IEEE Design and Test of Computers. 13.
- [2] CHU, Pong P. *RTL hardware design using VHDL: coding for efficiency, portability, and scalability*. Hoboken, N.J.: Wiley-Interscience, c2006. ISBN 9780471720928.
- [3] KOLOUCH, Jaromír. *Programovatelné logické obvody: přednášky*. Vyd. 3., dopl. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky, 2006. ISBN 80-214-3270-5.
- [4] KUBÍČEK, Michal. *Úvod do problematiky obvodů FPGA pro integrovanou výuku VUT a VŠB-TUO*. [Skriptum FEKT VUT v Brně.] Brno, 120 s. 2014.
- [5] KUON, Ian, Russell TESSIER a Jonathan ROSE. *FPGA Architecture: Survey and Challenges*. Foundations and Trends® in Electronic Design Automation [online]. 2007, 2(2), 135-253 [cit. 2020-11-18]. ISSN 1551-3939. Dostupné z: doi:10.1561/1000000005.
- [6] AMANO, Hideharu, ed. *Principles and Structures of FPGAs [online]*. Singapore: Springer Singapore, 2018 [cit. 2020-11-18]. ISBN 978-981-13-0823-9. Dostupné z: doi:10.1007/978-981-13-0824-6.
- [7] CSCE 436 - Advanced Embedded Systems [online]. *Xilinx Artix-7 architecture*. Nebraska: University of Lincoln [cit. 2020-11-18]. Dostupné z: https://cse.unl.edu/~jfalkinburg/cse_courses/2020/436/lecture/lecture32.html
- [8] Xilinx [online]. *7 Series FPGAs Configurable Logic Block: User Guide*. United States: Inc. Xilinx, 2016, September 27 [cit. 2020-11-18]. Dostupné z: <https://www.xilinx.com/support.html#documentation>
- [9] KRÁL, Jan a Michal KUBÍČEK. *Programovatelné logické obvody: Přednáška 8*. Vysoké Učení Technické v Brně [online]. [cit. 2020-11-18]. Dostupné z: <https://www.vutbr.cz/studenti/predmety/detail/149854?apid=149854>
- [10] BDIO - Digitální obvody: Úloha č. 6. Vysoké Učení Technické v Brně [online]. [cit. 2020-11-18]. Dostupné z: <https://moodle.vutbr.cz/mod/resource/view.php?id=99629>
- [11] Xilinx [online]. *7 Series FPGAs SelectIO Resources: User Guide*. United States: Inc. Xilinx, 2018, May 8 [cit. 2020-11-18]. Dostupné z: <https://www.xilinx.com/support.html#documentation>

- [12] Xilinx [online]. *7 Series FPGAs Memory Resources: User Guide*. United States: Inc. Xilinx, 2019, July 3 [cit. 2020-11-18]. Dostupné z: <https://www.xilinx.com/support.html#documentation>
- [13] Xilinx [online]. *7 Series DSP48E1 Slice: User Guide*. United States: Inc. Xilinx, 2018, March 27 [cit. 2020-11-18]. Dostupné z: <https://www.xilinx.com/support.html#documentation>
- [14] Xilinx [online]. *7 Series FPGAs Clocking Resource: User Guide*. United States: Inc. Xilinx, 2018, July 30 [cit. 2020-11-18]. Dostupné z: <https://www.xilinx.com/support.html#documentation>
- [15] Xilinx [online]. *7 Series FPGAs Data Sheet: Overview*. United States: Inc. Xilinx, 2020, September 8 [cit. 2020-11-18]. Dostupné z: <https://www.xilinx.com/support.html#documentation>
- [16] SYZYGY *Next generation FPGA connectivity [online]*. Opal Kelly, 2020 [cit. 2020-12-14]. Dostupné z: <https://syzygyfpga.io/>
- [17] BARTÍK, Matěj, Tomáš BENEŠ a Karel HYNEK. *An Example of PCB Reverse Engineering - Reconstruction of Digilent JTAG SMT3 Schematic*. In: IEEE Xplore [online]. c2020 [cit. 2020-12-14]. Dostupné z: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8977100&tag=1>
- [18] Mouser Electronics [online]. Texas: Mouser Electronics, c2020 [cit. 2020-12-15]. Dostupné z: <https://cz.mouser.com/>
- [19] USB104 A7: Artix-7 FPGA Development Board in PC/104 Form Factor. Digilent: A National Instrument Company [online]. United States of America: Digilent, 2021 [cit. 2021-5-16]. Dostupné z: <https://store.digilentinc.com/usb104-a7-artix-7-fpga-development-board-in-pc-104-form-factor/>
- [20] Xilinx Artix-7 FPGA AC701 Evaluation Kit. Xilinx [online]. United States: Xilinx, 2021 [cit. 2021-5-16]. Dostupné z: <https://www.xilinx.com/products/boards-and-kits/ek-a7-ac701-g.html>
- [21] DOHNAL, Jaroslav. Design of low order high OSR discrete time delta-sigma modulator for audio applications. Brno, 2020. Diplomová práce. Vysoké Učení Technické v Brně. Vedoucí práce doc. Ing. Jiří Háze, Ph.D.

Seznam symbolů a zkratek

ALU	Arithmetic Logic Unit
ASMBL	Advanced Silicon Modular Block
BLVDS	Bus LVDS
BRAM	Block RAM
CE	Clock Enable
CLB	Configurable Logic Block
CMOS	Complementary Metal–Oxide–Semiconductor
CMT	Clock Management Tiles
CPLD	Complex Programmable Logic Devices
DCI	Digitally Controlled Impedance
DDR	Dual Data Rate
DPS	Deska Plošných Spojů
DRP	Dual Role Port
DSP	Digital Signal Processing
DA	Digital-Analog
EEPROM	Electrically Erasable Programmable Read Only Memory
EPROM	Erasable Programmable Read Only Memory
ESD	Electrostatic Discharge
FF	Flip-Flop
FIFO	First In First Out
FPGA	Field-Programmable Gate Array
HP	High Performance
HR	High Range
HSTL	High Speed Transceiver Logic

I²C	Inter-Integrated Circuit
I/O	Input-Output
JTAG	Joint Test Action Group
KO	Klopný Obvod
LED	Light Emitting Diode
LUT	Look-Up Table
LVC MOS	Low Voltage CMOS
LVDS	Low voltage differential signaling
LVTTTL	Low Voltage Transistor Transistor Logic
MEMS	Micro Electro Mechanical Systems
MMCM	Mixed-Mode Clock Manager
MSB	Most Significant Bit
PAL	Programmable Array Logic
PCI	Peripheral Component Interconnect
PDM	Pulse Density Modulation
PLA	Programmable Logic Array
PLD	Programmable Logic Devices
PLL	Phase-Locked Loop
PPDS	Point-to-Point Differential Signaling
PROM	Programmable Read Only Memory
PWM	Pulse Width Modulation
RAM	Random Access Memory
RGB	Red, Green, Blue
RSDS	Reduced Swing Differential Signaling
SDR	Single Data Rate

SIMD	Single Instruction, Multiple Data
SPI	Serial Peripheral Interface
SPLD	Simple Programmable Logic Devices
SR	Set/Reset
SRAM	Static Random Access Memory
SSTL	Stub Series Terminated Logic
THT	Through Hole Technology
UART	Universal asynchronous receiver-transmitter
USB	Universal Serial Bus
VHDL	Very High Speed Integrated Circuit Hardware Description Language
WE	Write Enable

Seznam příloh

A	Soupis komponent	60
B	Vývojový kit	65
C	Programovací zařízení	82

A Soupis komponent

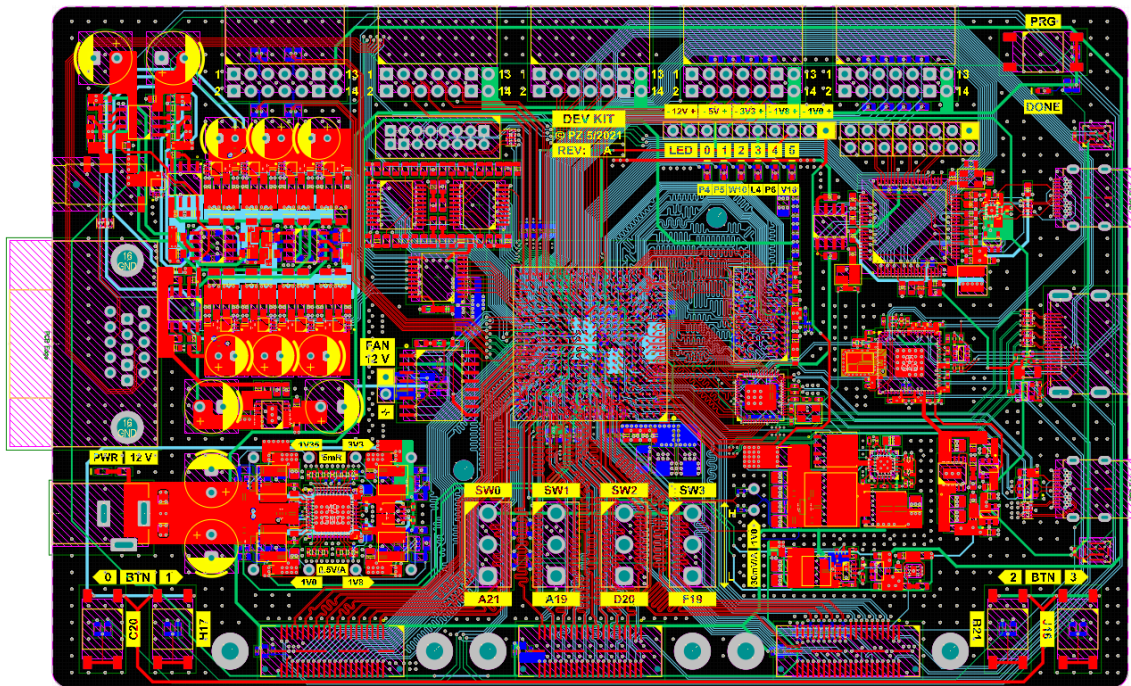
Součástka	Označení	Hodnota	Množství	Cena/ks
Integrované obvody				
FPGA	XC7A200T-2FBG484I	-	1	2 096,40 Kč
RAM DDR3L	AS4C256M16D3LC-10BCN	4Gb	1	225,32 Kč
Procesor	MIMXRT1011DAE5A	-	1	89,34 Kč
PLL	SI5341A-D-GMR	-	1	377,80 Kč
FTDI	FT600Q	-	1	186,28 Kč
FLASH paměť	AT25SF321B-SSHB-B	32Mb	1	9,46 Kč
USB spínač	AP2191DWG	-	1	8,86 Kč
USB kontroler	FUSB302BMPX	-	1	23,08 Kč
ESD ochrana	USBLC6-2SC6	-	1	8,44 Kč
ESD ochrana	RCLAMP3324P	-	6	16,51 Kč
FLASH paměť	S25FL256SAGMFIR01	256Mb	1	80,17 Kč
Zesilovač	INA199A2DCKR	100V/V	5	11,82 Kč
Mikrofon	IMP34DT05TR	-	2	38,78 Kč
Klopný obvod	SN74AHC574DBR	8	3	9,33 Kč
Invertor	NC7WZ04	-	1	6,90 Kč
Klopný obvod	NC7SZ74K8X	1	9	8,17 Kč
Operační zesilovač	AD8066	-	2	132,83 Kč
Operační zesilovač	OPA1656	-	2	63,40 Kč
PMIC	ADP5052	-	1	181,83 Kč
Napěťový regulátor	MP2315	-	1	56,59 Kč
DCDC měnič	ADP1853	-	1	67,33 Kč
Lineární regulátor	ADP223A	-	1	34,06 Kč
Napěťová reference	LT6654	-	1	131,26 Kč
Lineární regulátor	LP5907MFX-3.3	-	1	10,87 Kč
Lineární regulátor	LP5907MFX-1.8	-	1	10,45 Kč
Napěťový regulátor	LT3045EMSE	-	2	134,67 Kč
Nábojová pumpa	SP6661	-	1	31,96 Kč
Konektory, přepínače a tlačítka				
Programovací konektor	87831-1620	16p	1	44,54 Kč
Napájecí konektor	DC-10B	-	1	11,82 Kč
Hřebínkový konektor	SSW-110-01-T-S	10p	1	22,09 Kč
Hřebínkový konektor	SSW-108-01-T-D	16p	1	24,63 Kč
Vysokorychlostní konektor	QSE-020-01-F-D-A	40p	3	142,00 Kč
PMOD konektor	SSW-107-02-T-D-RA	14p	5	28,56 Kč
HDMI	DS1114-BN0	-	1	28,38 Kč
VGA	DS1038-15FBNSIA74	-	1	22,08 Kč
USB-C	217B-AG01	-	2	45,62 Kč
Jack zásuvka 3,5mm	SJ2-3574A-SMT-TR	-	1	18,50 Kč
Přepínač	5MS1S102AM2QE	-	4	22,50 Kč
Tlačítka	1301.9314	-	5	6,37 Kč

Krystaly a oscilátory				
Oscilátor	ECS-3225MVQ-1000-CN-TR	100MHz	1	32,49 Kč
Krystal	CX3225SB54000D0FPJC1	54MHz	1	21,43 Kč
Krystal	ECS-300-18-33-AEN-TR	30MHz	1	9,43 Kč
Krystal	ECS-240-12-33-AGM-TR	24MHz	1	9,35 Kč
Tranzistory a diody				
MOSFET	SIRA24DP	-	1	18,21 Kč
MOSFET	SIRC18DP	-	1	28,30 Kč
MOSFET	SI7232DN	-	1	18,52 Kč
MOSFET	FDV301N	-	1	0,71 Kč
Spínací dioda	BAS16XV2T1G	-	1	3,12 Kč
Usměrňovací dioda	MBR0520LT3G	-	1	6,63 Kč
LED zelená	FYLS-0603UGC	-	2	1,41 Kč
LED růžová	OSK40603C1E	-	6	1,69 Kč
Cívky				
Cívka	XAL4030-332	3,3 μ H	2	38,19 Kč
Cívka	XAL4030-472	4,7 μ H	1	38,19 Kč
Cívka	XAL4030-822	8,2 μ H	2	40,74 Kč
Cívka	XAL7070-651	650nH	1	64,67 Kč
Feritová perlička	MMZ2012S121AT000	-	1	1,10 Kč
Feritová perlička	MPZ1608S101ATAH0	-	20	0,71 Kč
Kondenzátory				
Kondenzátor 0402	04026D104KAT2A	0.1 μ	27	0,21 Kč
Kondenzátor 0402	JMK105BJ225KV-F	2 μ 2	111	0,66 Kč
Kondenzátor 0402	EMK105BJ105KV-F	1 μ /16V	4	0,87 Kč
Kondenzátor 0402	C0402C272K5RACTU	2n7	9	0,66 Kč
Kondenzátor 0402	CGA2B2C0G1H060D050BA	6p	1	0,71 Kč
Kondenzátor 0402	04025A100JAT2A	10p	1	0,37 Kč
Kondenzátor 0402	C1005C0G1H120J050BA	12p	2	0,68 Kč
Kondenzátor 0402	GCM155R71E223KA55D	22n	1	0,63 Kč
Kondenzátor 0402	GCM155R71H221KA37D	200p	1	0,60 Kč
Kondenzátor 0402	UMK105B7331KV-F	330p	2	0,37 Kč
Kondenzátor 0603	06033C104KAT4A	0.1 μ	1	0,52 Kč
Kondenzátor 0603	GRM1885C1H102FA01D	1n C0G 1%	6	1,47 Kč
Kondenzátor 0603	CL10A225KP8NUNC	2 μ 2	9	0,68 Kč
Kondenzátor 0603	CL10A225KO8NUNC	2 μ 2/16V	2	0,92 Kč
Kondenzátor 0603	C0603C100J3HACTU	10p	2	0,50 Kč
Kondenzátor 0603	GRM188R60J226MEA0D	22 μ	100	1,32 Kč
Kondenzátor 0603	0603B331K500CT	330p	19	0,28 Kč
Kondenzátor 0805	GRM2165C1H332FA01D	3n3 C0G 1%	4	4,09 Kč
Kondenzátor 0805	GRM2165C1H472FA01D	4n7 C0G 1%	4	4,14 Kč
Kondenzátor 0805	CL21A226MOQNNNE	22 μ /16V	8	2,88 Kč
Kondenzátor 0805	CL21A476MQYNNNG	47 μ /6V3	12	5,15 Kč

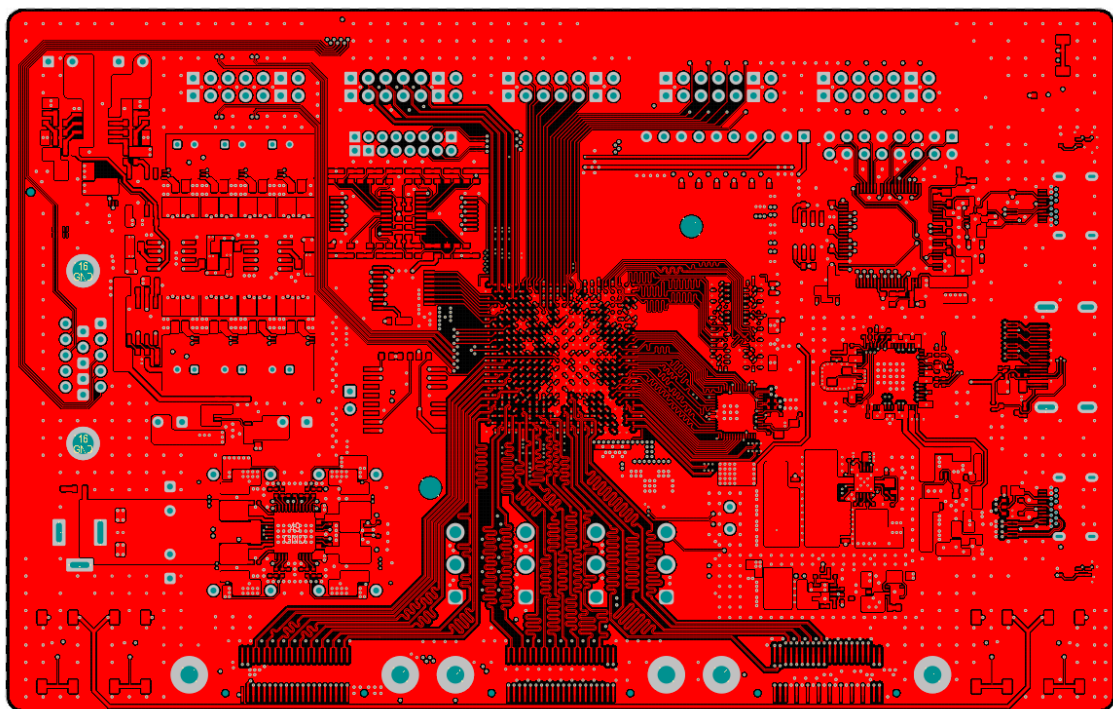
Kondenzátor 1206	EMK316BBJ476ML-T	47 μ /16V	6	13,31 Kč
Kondenzátor 1206	GRM31M5C1H473JA01L	47n C0G 1%	16	7,46 Kč
Kondenzátor 1206	CL31A107MQHNNWE	100 μ /6V3	13	7,73 Kč
Kondenzátor case C	TAJC337K004RNJ	330 μ	2	22,85 Kč
Kondenzátor	A758EK477M0JAAE015	470 μ /6V3	6	5,77 Kč
Kondenzátor	A750KS567M1CAAE013	560 μ /16V	3	16,04 Kč
Kondenzátor	A750KS158M0JAAE014	1500 μ /6V3	3	18,66 Kč
Rezistory				
Rezistor 0402	RC0402FR-071KL	1k	8	0,11 Kč
Rezistor 0402	ERJ-2RHD1601X	1k6	1	1,10 Kč
Rezistor 0402	ERJ-2RKF1741X	1k74	2	0,50 Kč
Rezistor 0402	CRCW04023K90FKEDC	3k9	1	0,71 Kč
Rezistor 0402	RC0402FR-074K7L	4k7	5	0,26 Kč
Rezistor 0402	ERJ-2RKF6801X	6k8	3	0,50 Kč
Rezistor 0402	CRCW04027K50FKEDC	7k5	2	0,71 Kč
Rezistor 0402	RC0402FR-0710KL	10k	9	0,31 Kč
Rezistor 0402	RC0402FR-0714KL	14k	1	0,26 Kč
Rezistor 0402	RC0402FR-1015KL	15k	1	0,26 Kč
Rezistor 0402	RC0402FR-0716K5L	16k5	1	0,31 Kč
Rezistor 0402	RC0402JR-1320RL	20R	1	0,21 Kč
Rezistor 0402	RC0402FR-7D20KL	20k	2	0,26 Kč
Rezistor 0402	AC0402FR-7W22KL	22k	2	0,47 Kč
Rezistor 0402	RC0402JR-0724KL	24k	3	0,21 Kč
Rezistor 0402	RC0402FR-0724K9L	24k9	1	0,26 Kč
Rezistor 0402	CRCW040227K0FKEDC	27k	1	0,71 Kč
Rezistor 0402	RC0402FR-0727RL	27R	6	0,31 Kč
Rezistor 0402	RC0402FR-1330KL	30k	2	0,26 Kč
Rezistor 0402	CR0402-FX-4022GLF	40k2	1	0,37 Kč
Rezistor 0402	AC0402FR-1347KL	47k	1	0,37 Kč
Rezistor 0402	CR0402-FX-4992GLF	50k	1	0,21 Kč
Rezistor 0402	RC0402FR-0775KL	75k	2	0,31 Kč
Rezistor 0402	RC0402FR-0780R6L	80.6R	1	0,26 Kč
Rezistor 0402	RC0402FR-07100RL	100R 1%	2	0,31 Kč
Rezistor 0402	ERJ-2RKF2003X	200k	1	0,50 Kč
Rezistor 0402	RC0402FR-07240RL	240R	1	0,31 Kč
Rezistor 0402	RC0402FR-07267RL	267R	1	0,26 Kč
Rezistor 0402	RC0402FR-07845RL	845R	1	0,26 Kč
Rezistor 0603	CR0603-FX-2001ELF	2k	1	0,26 Kč
Rezistor 0603	0603SAJ0225T5E	2M2	1	0,31 Kč
Rezistor 0603	RT0603FRE072RL	2R 0.1%	2	1,39 Kč
Rezistor 0603	RT0603FRD073K01L	3k01 0.1%	16	0,68 Kč
Rezistor 0603	AC0603FR-104K7L	4k7	5	0,50 Kč
Rezistor 0603	RC0603FR-0710KL	10k	13	0,39 Kč

Rezistor 0603	0603SAF1502T5E	15k	1	0,32 Kč
Rezistor 0603	0603SAF2102T5E	21k	5	0,35 Kč
Rezistor 0603	AR03BTCX0220	22R 0.1%	14	2,80 Kč
Rezistor 0603	0603SAF2402T5E	24k	1	0,32 Kč
Rezistor 0603	CRCW060333K2FKEAC	33k2	1	0,63 Kč
Rezistor 0603	CR0603-FX-3602ELF	36k	1	0,42 Kč
Rezistor 0603	RC0603FR-1339KL	39k	1	0,39 Kč
Rezistor 0603	AC0603FR-1047KL	47k	1	0,50 Kč
Rezistor 0603	KDV06FR050ET	50mR	2	1,44 Kč
Rezistor 0603	0603SAF680JT5E	68R	24	0,32 Kč
Rezistor 0603	ERA3AEB750V	75R	13	2,56 Kč
Rezistor 0603	AC0603FR-10100RL	100R	7	0,50 Kč
Rezistor 0603	RC0603FR-10150RL	150R	19	0,13 Kč
Rezistor 0603	CR0603-FX-2000ELF	200R	1	0,24 Kč
Rezistor 0603	AC0603FR-10330RL	330R	5	0,18 Kč
Rezistor 0603	ERA-3AEB4640V	464R 0.1%	4	5,69 Kč
Rezistor 0603	AR03BTCX5000N	500R 0.1%	4	2,79 Kč
Rezistor 0603	ERA-3AEB6340V	634R 0.1%	12	6,65 Kč
Rezistor 0805	CFN0805-FX-R005ELF	5mR	4	5,16 Kč
Rezistor 2512	CSS2H-2512R-L300F	300 μ R	1	14,28 Kč
Ostatní				
DPS	-	-	1	895,00 Kč
Celková cena vývojového kitu				8 056,97 Kč

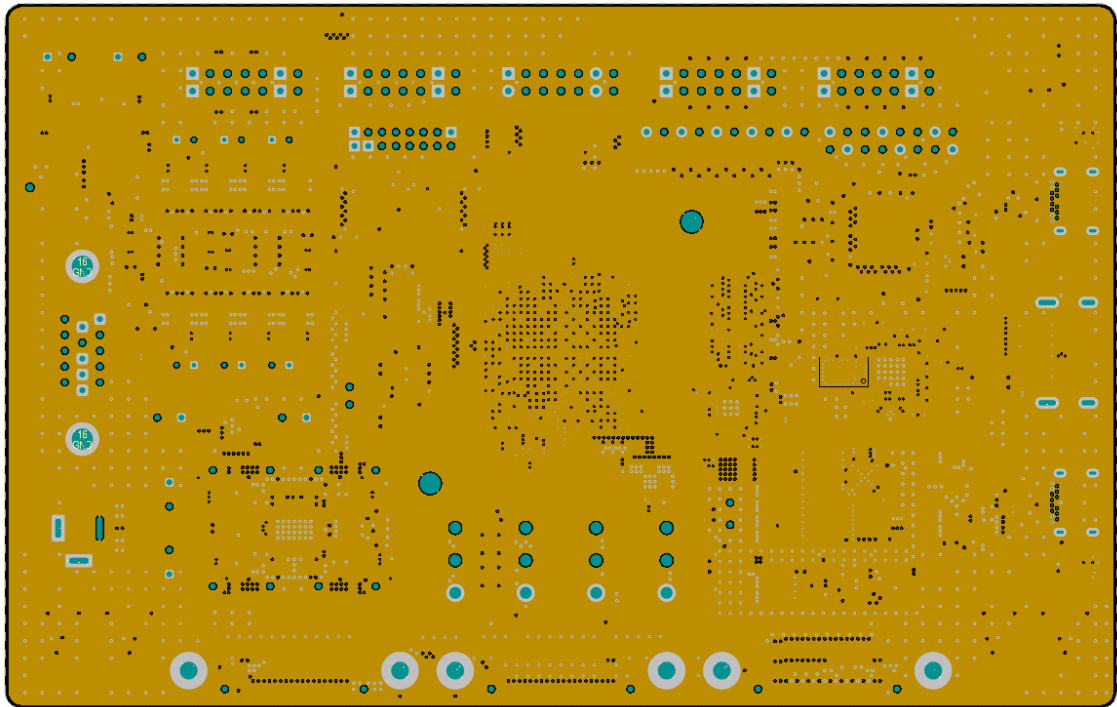
B Vývojový kit



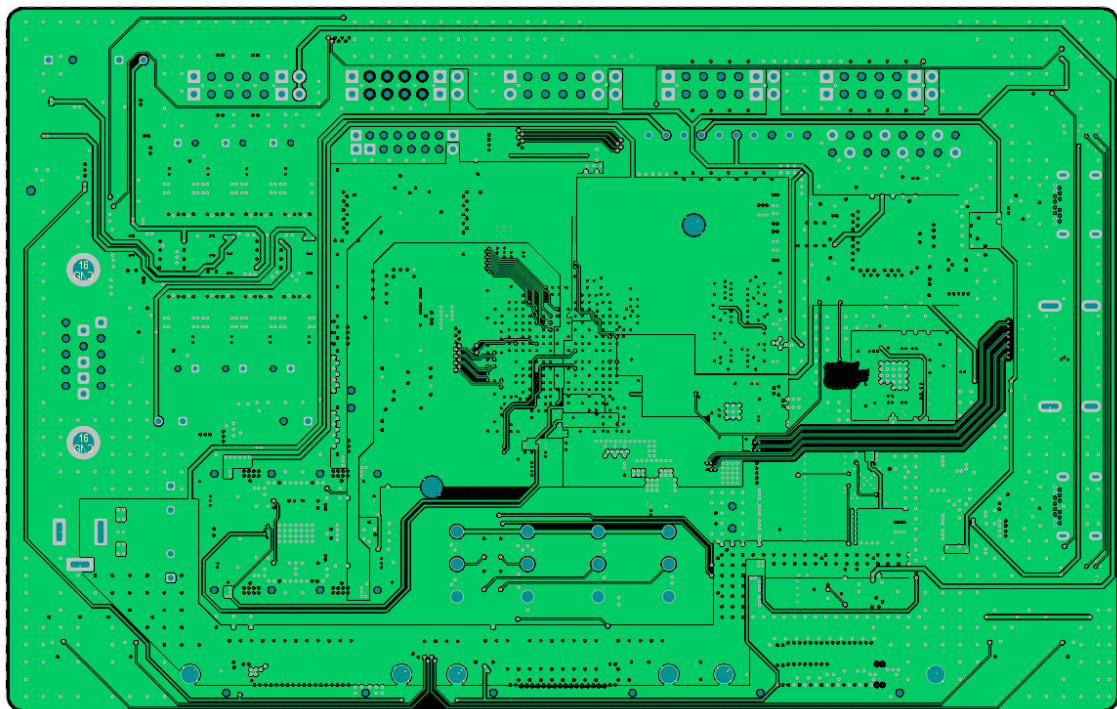
Obr. B.1: Pohled na všechny vrstvy vývojového kitu



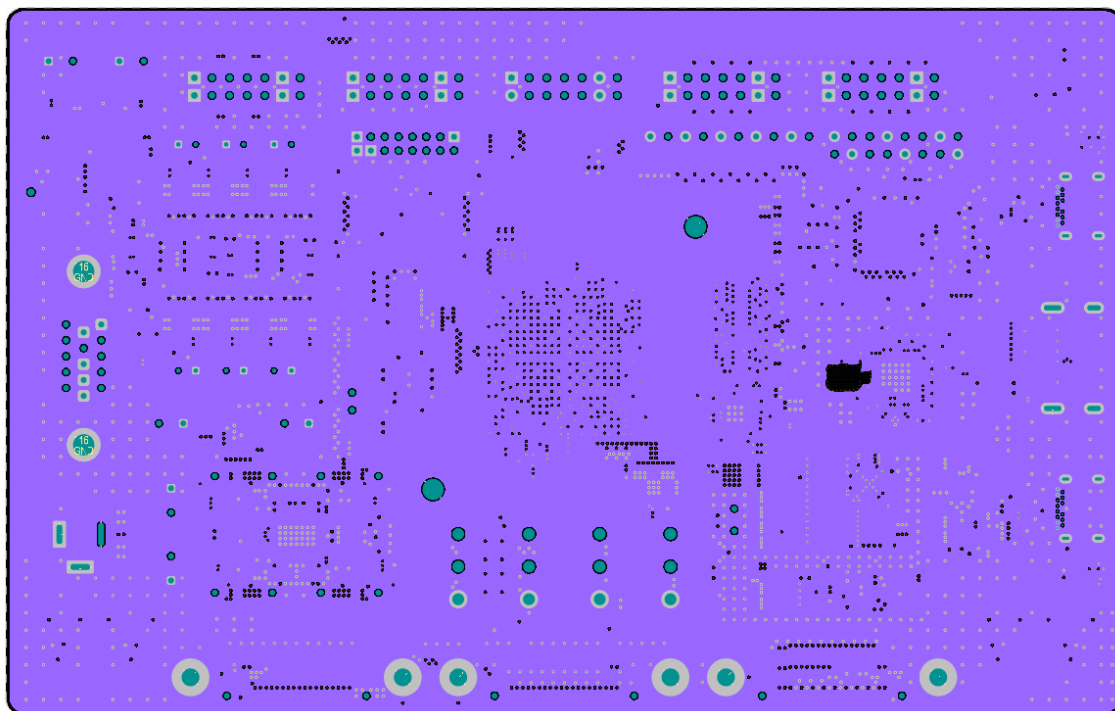
Obr. B.2: Pohled na vrstvu TOP vývojového kitu



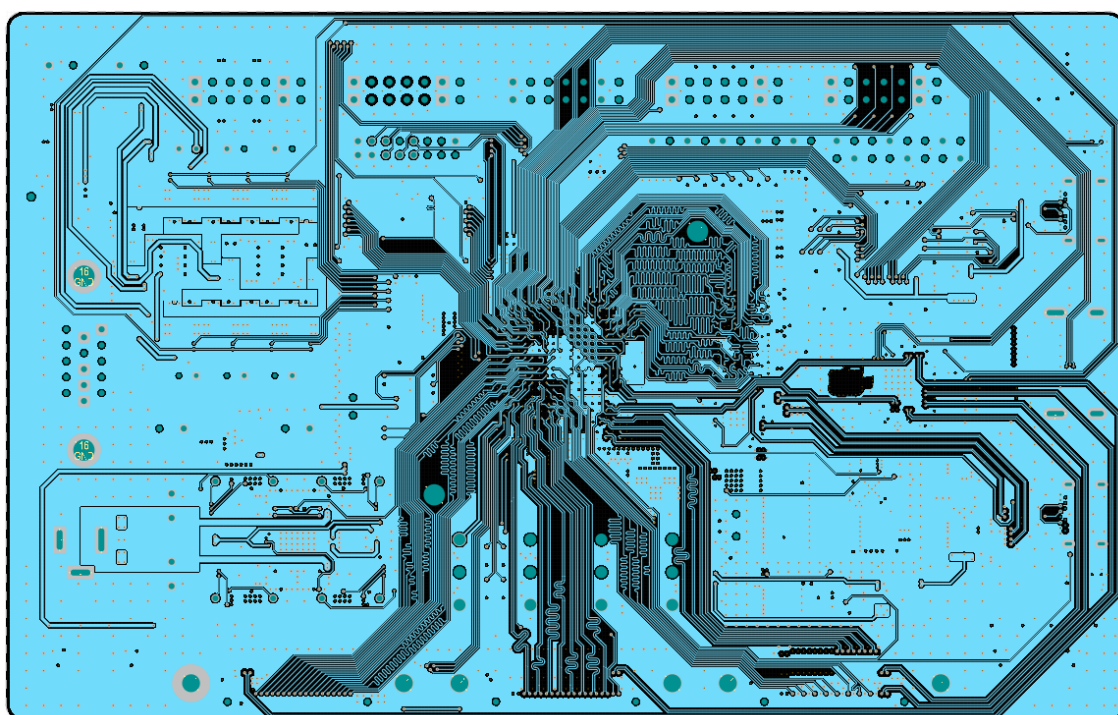
Obr. B.3: Pohled na vrstvu IN1 vývojového kitu



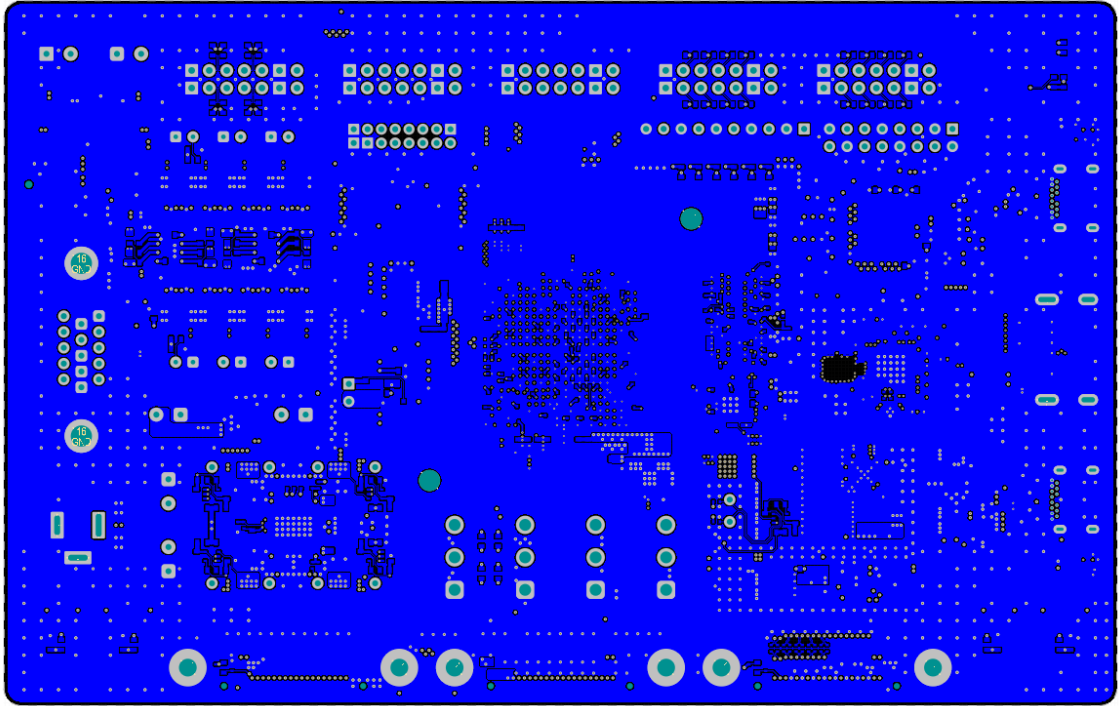
Obr. B.4: Pohled na vrstvu IN2 vývojového kitu



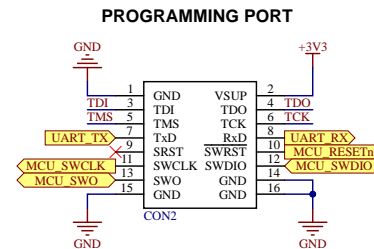
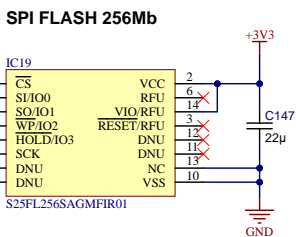
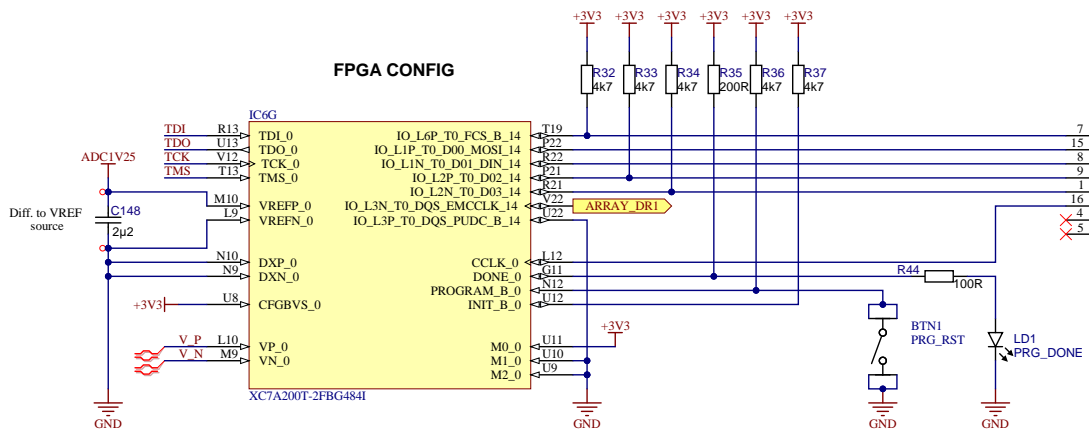
Obr. B.5: Pohled na vrstvu IN3 vývojového kitu



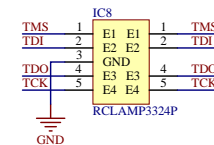
Obr. B.6: Pohled na vrstvu IN4 vývojového kitu



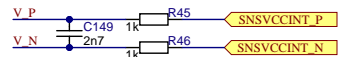
Obr. B.7: Pohled na vrstvu BOT vývojového kitu



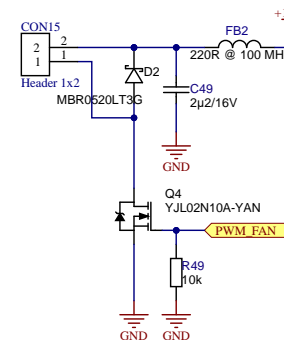
ESD PROTECTION



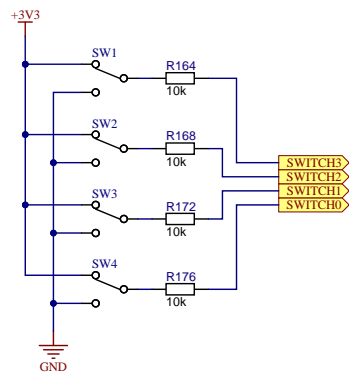
FPGA CORE CURRENT SENSE



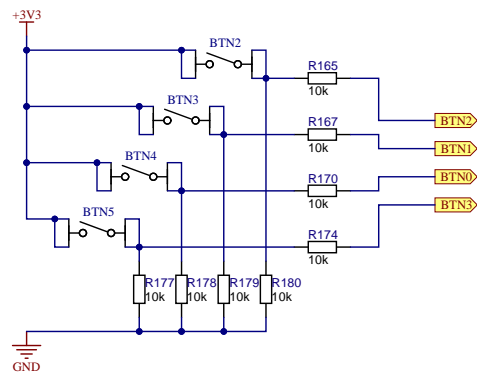
FAN



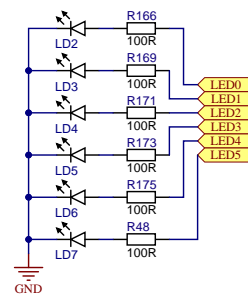
SWITCHES



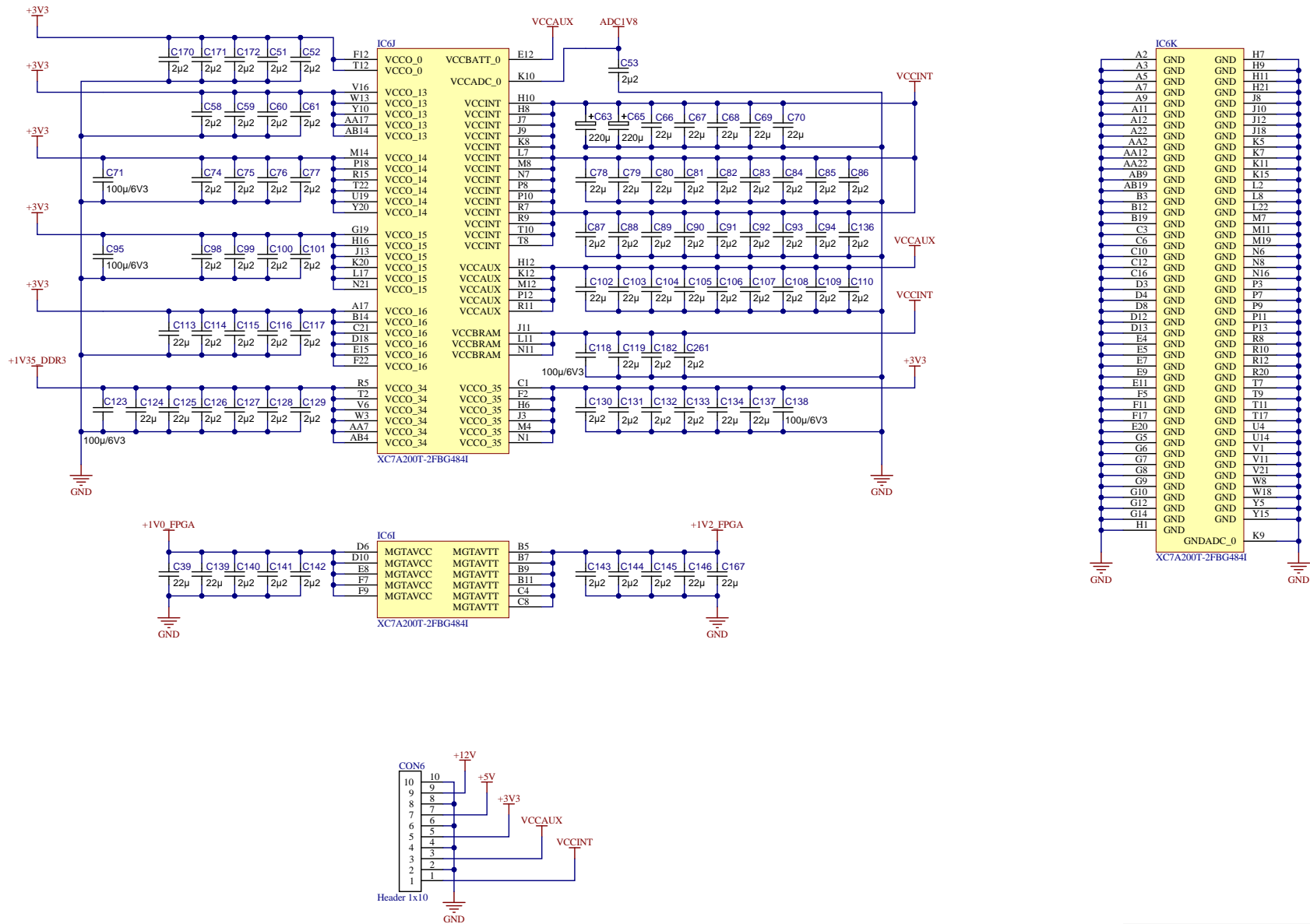
BUTTONS



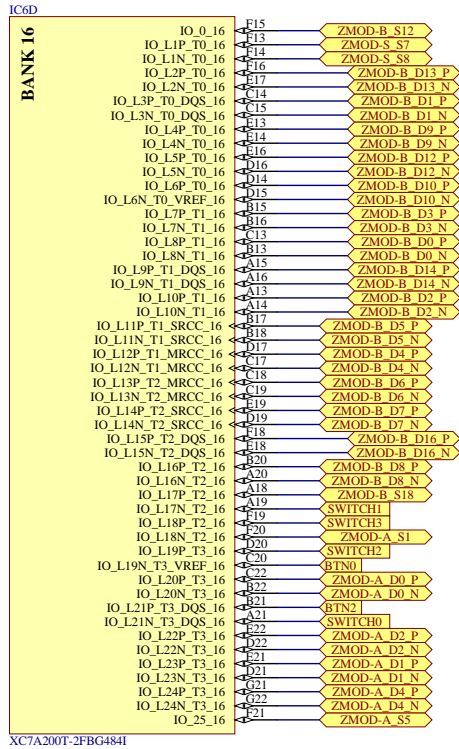
LEDs



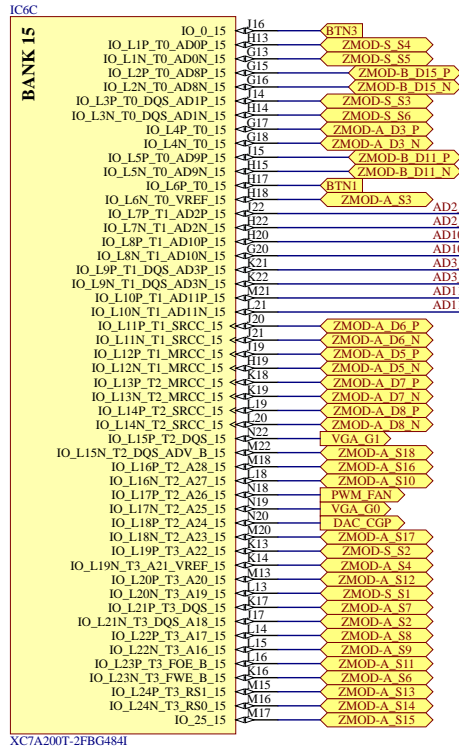
Title		
FPGA CONFIG		
Size	Number	Revision
A3		A01
Date:	5.16.2021	Sheet 1 of 12
File:	C:\Users\...\FPGA_config.SchDoc	Drawn By: Humans



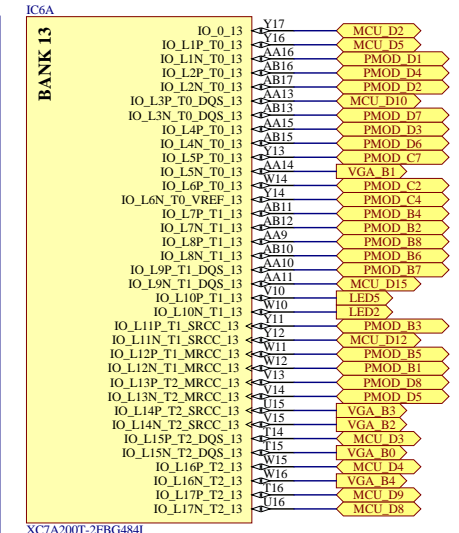
Title FPGA POWER		
Size A3	Number	Revision A01
Date: 5.16.2021	Sheet 2 of 12	
File: C:\Users\...\FPGA_power.SchDoc	Drawn By: Humans	



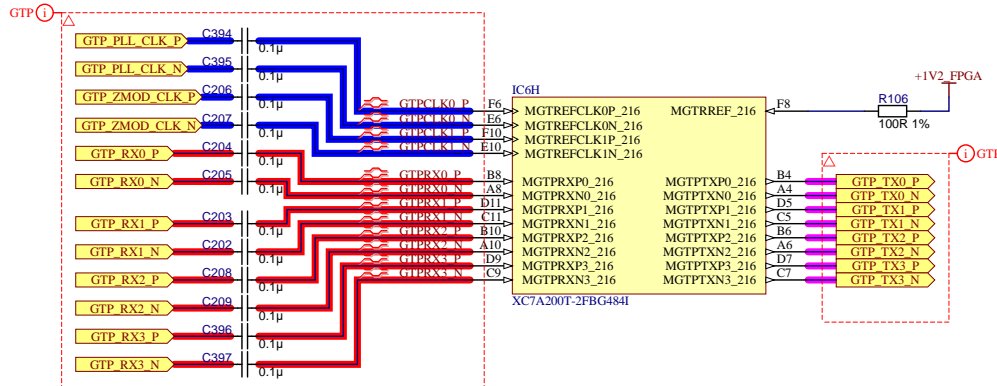
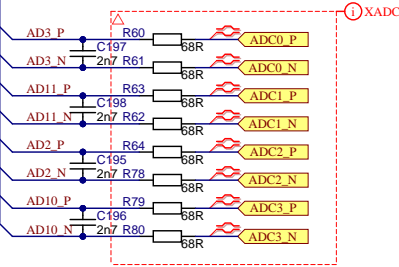
XC7A200T-2FBG484I



XC7A200T-2FBG484I

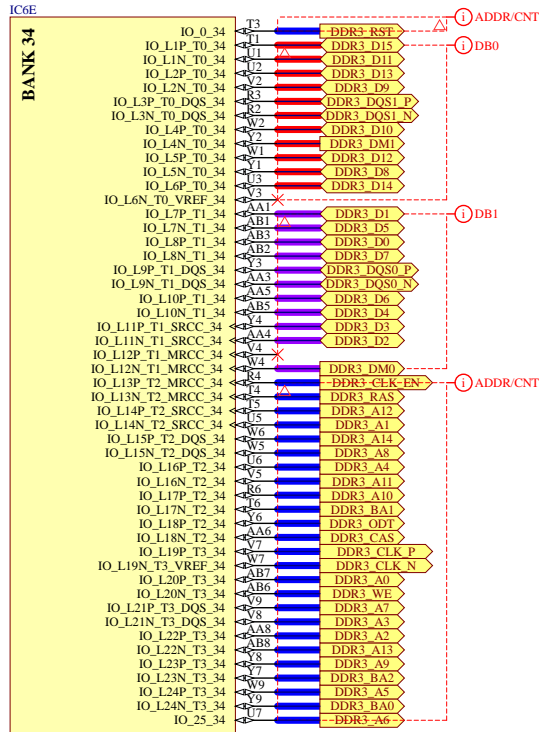


XC7A200T-2FBG484I

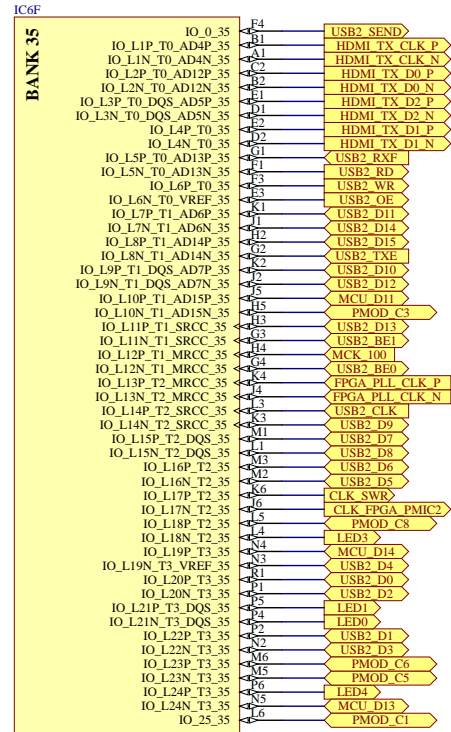


Title		
FPGA I/Os		
Size	Number	Revision
A3		A01
Date:	5.16.2021	Sheet 3 of 12
File:	C:\Users\...\FPGA_I/Os.SchDoc	Drawn By: Humans

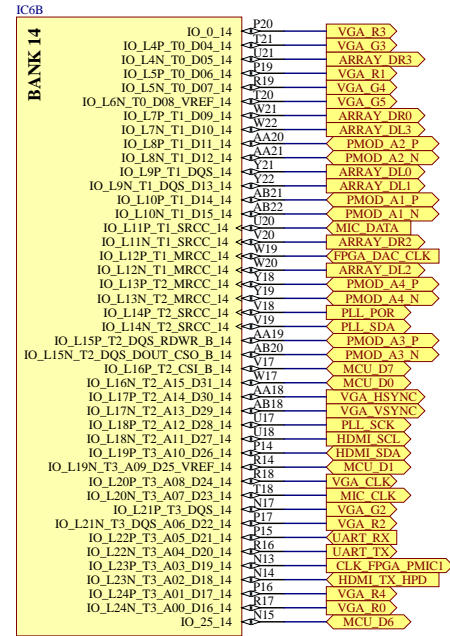
FPGA DDR3



XC7A200T-2FBG484I

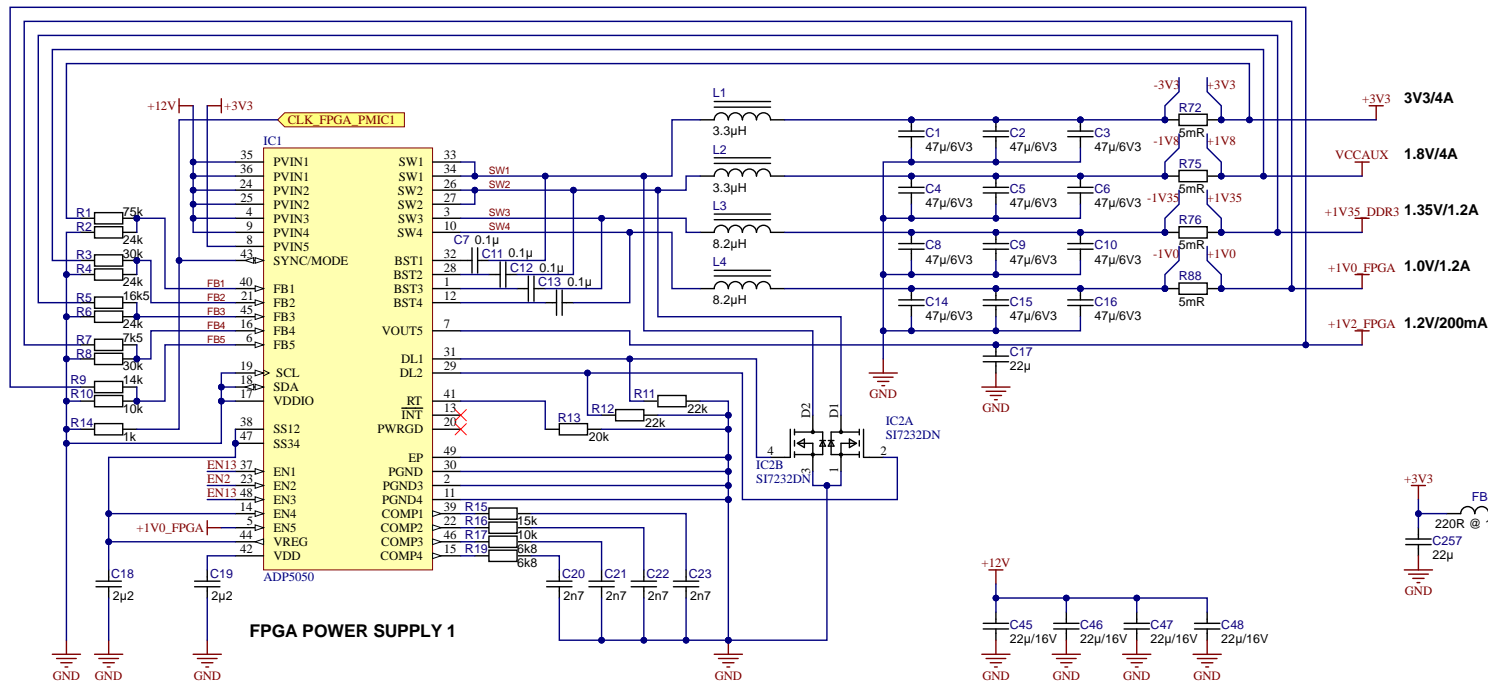


XC7A200T-2FBG484I

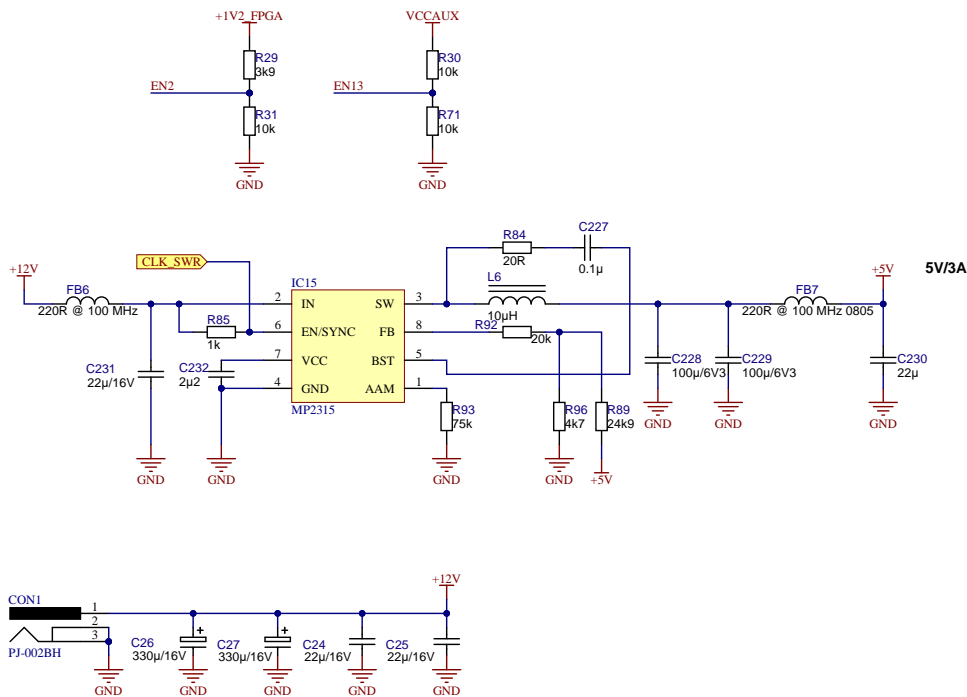
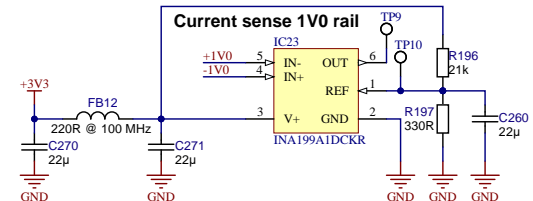
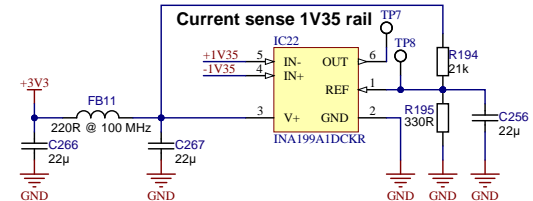
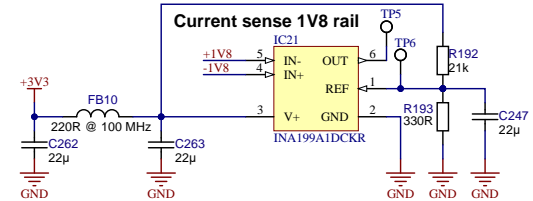
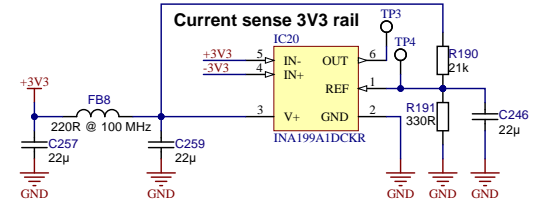
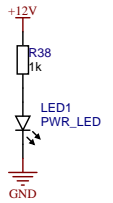


XC7A200T-2FBG484I

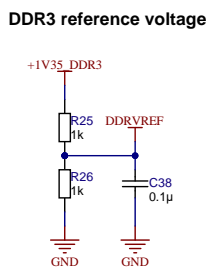
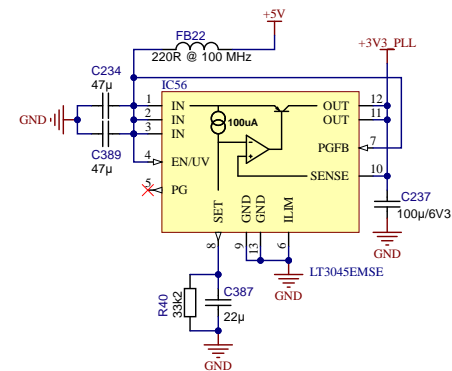
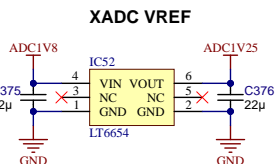
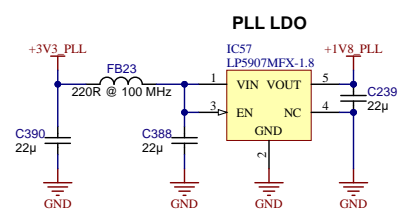
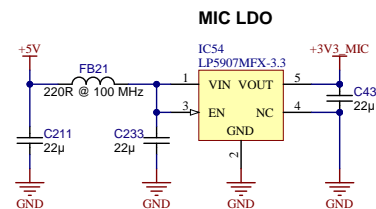
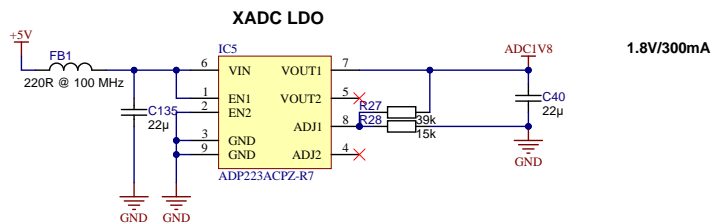
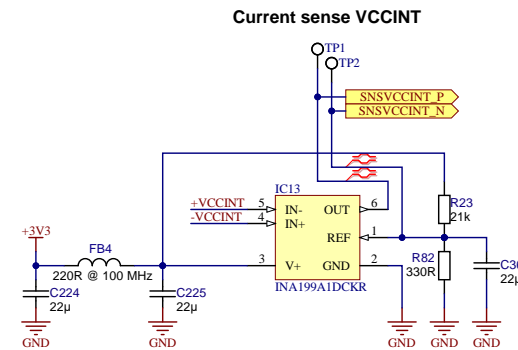
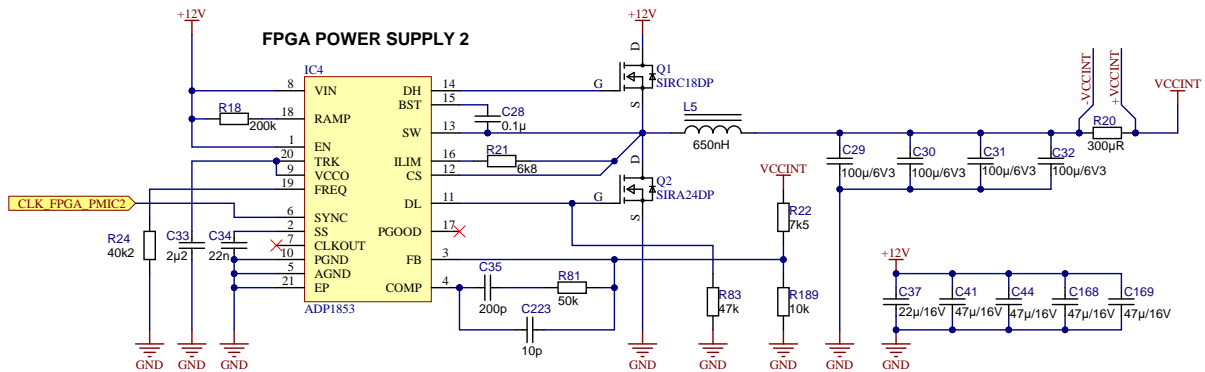
Title			FPGA BANKS		
Size	Number	Revision		A01	
A3					
Date:	5.16.2021	Sheet 4 of 12			
File:	C:\Users\... \FPGA_banks.SchDoc	Drawn By: Humans			



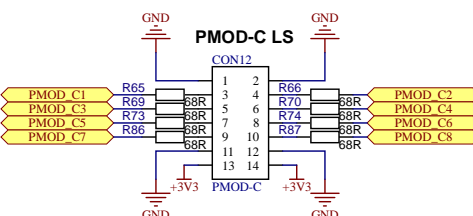
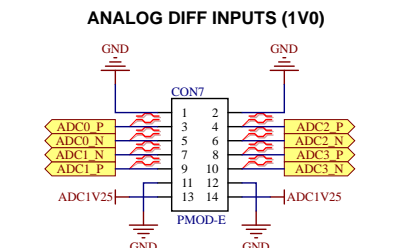
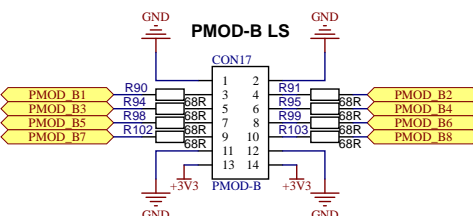
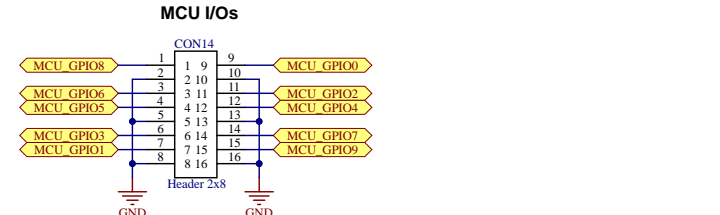
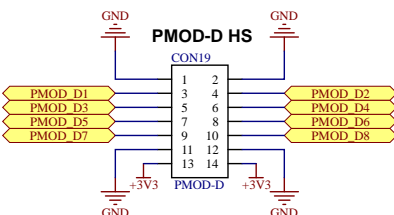
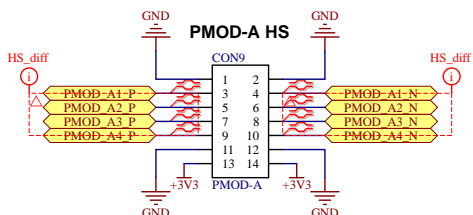
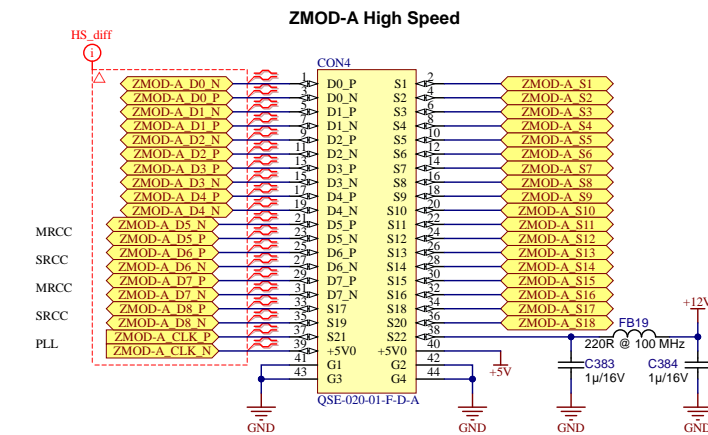
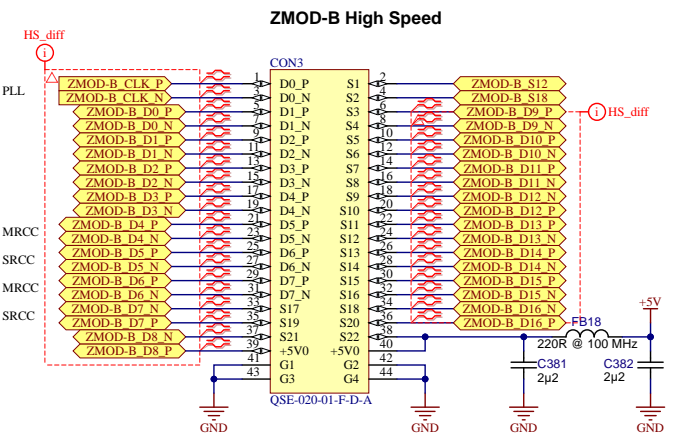
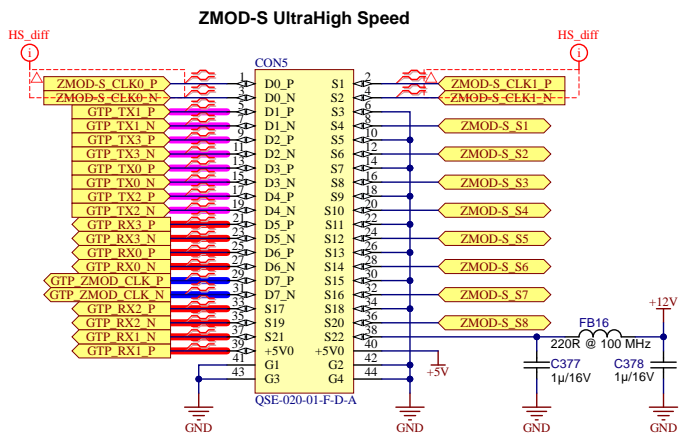
- 3V3 - VCCO (I/O banks, flash memory, μ C...)
- 1V8 - VCCAUX - various auxiliary circuits, CLMs, config. pins
- 1V35 - DDR3 RAM
- 1V0 - Vmgtavcc - PLLs, transmitters and receivers
- 1V2 - Vmgtavtt termination circuits
- 1V0 - VCCINT (internal logic, CLBs, BRAMs and DSP blocks)



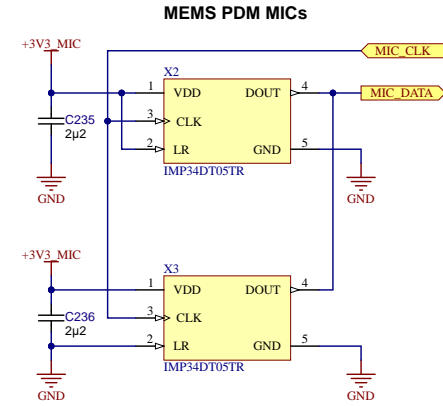
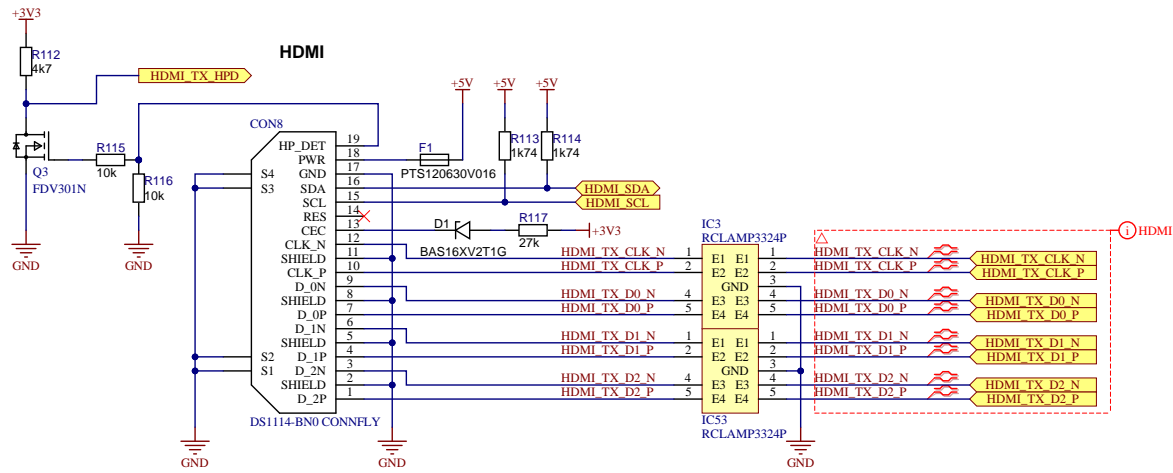
Title		
POWER REGULATION 1		
Size	Number	Revision
A3		A01
Date:	5.16.2021	Sheet 5 of 12
File:	C:\Users\... \PWR_REG_1.SchDoc	Drawn By: Humans



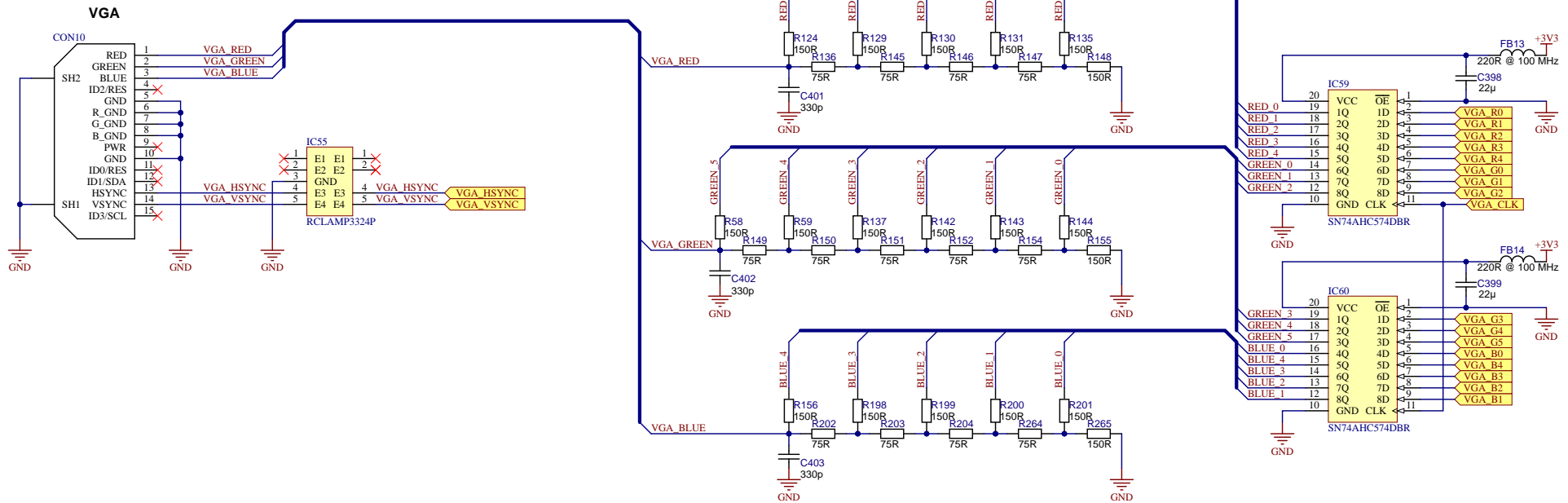
Title		
POWER REGULATION 2		
Size	Number	Revision
A3		A01
Date:	5.16.2021	Sheet 6 of 12
File:	C:\Users\...\PWR_REG_2.SchDoc	Drawn By: Humans



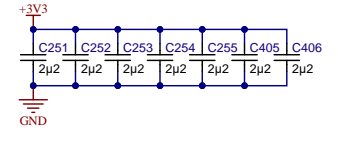
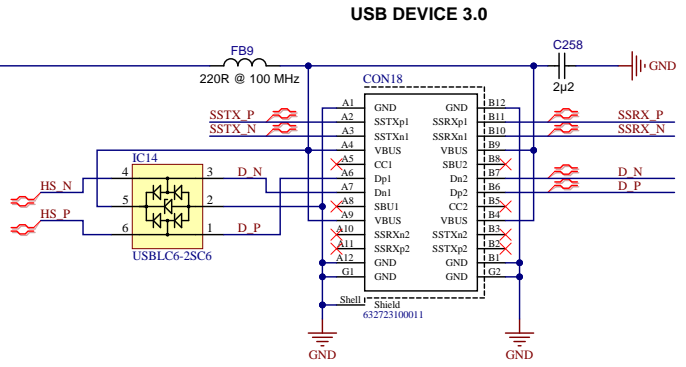
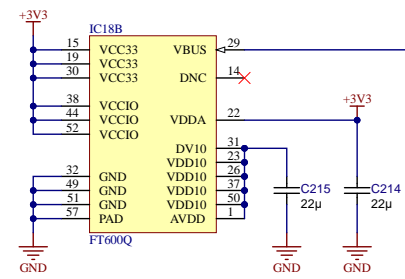
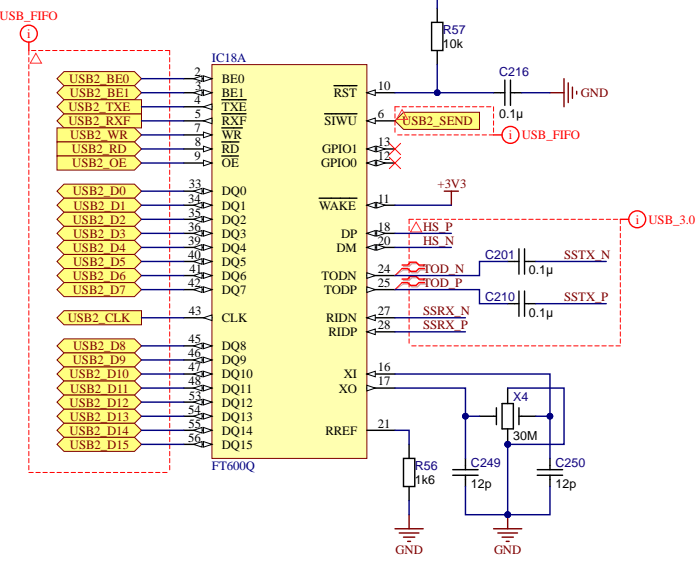
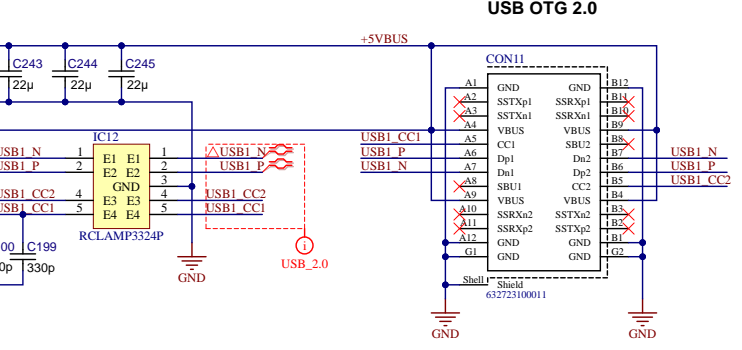
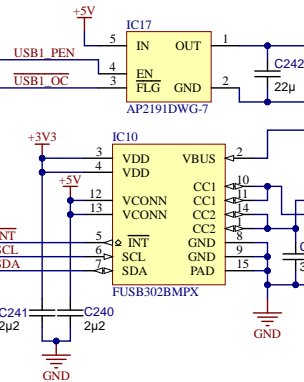
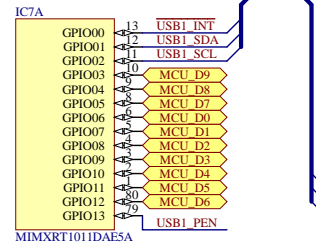
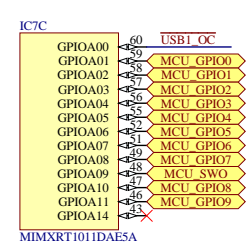
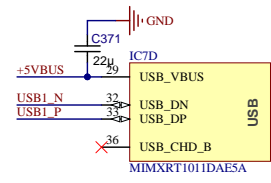
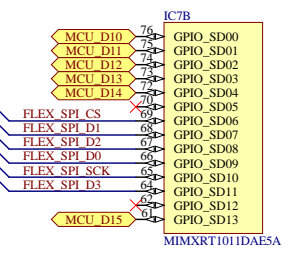
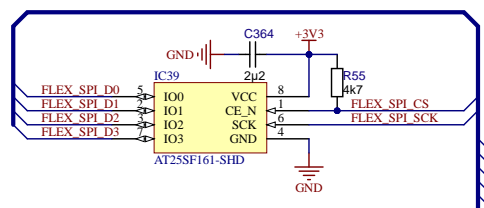
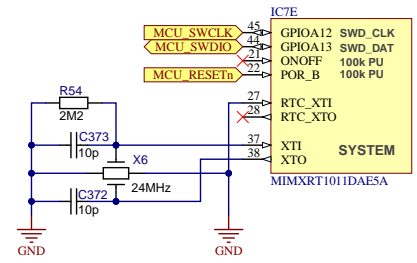
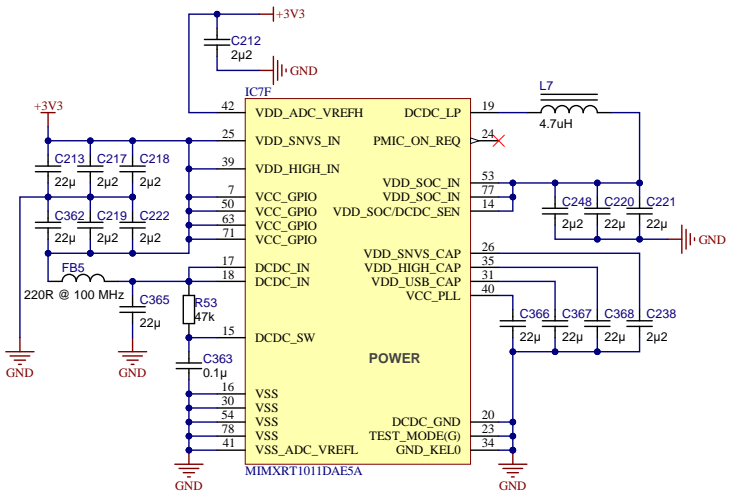
Title		
I/O connectors		
Size	Number	Revision
A3		A01
Date:	5.16.2021	Sheet 8 of 12
File:	C:\Users\...I/O_connectors.SchDoc	Drawn By: Humans



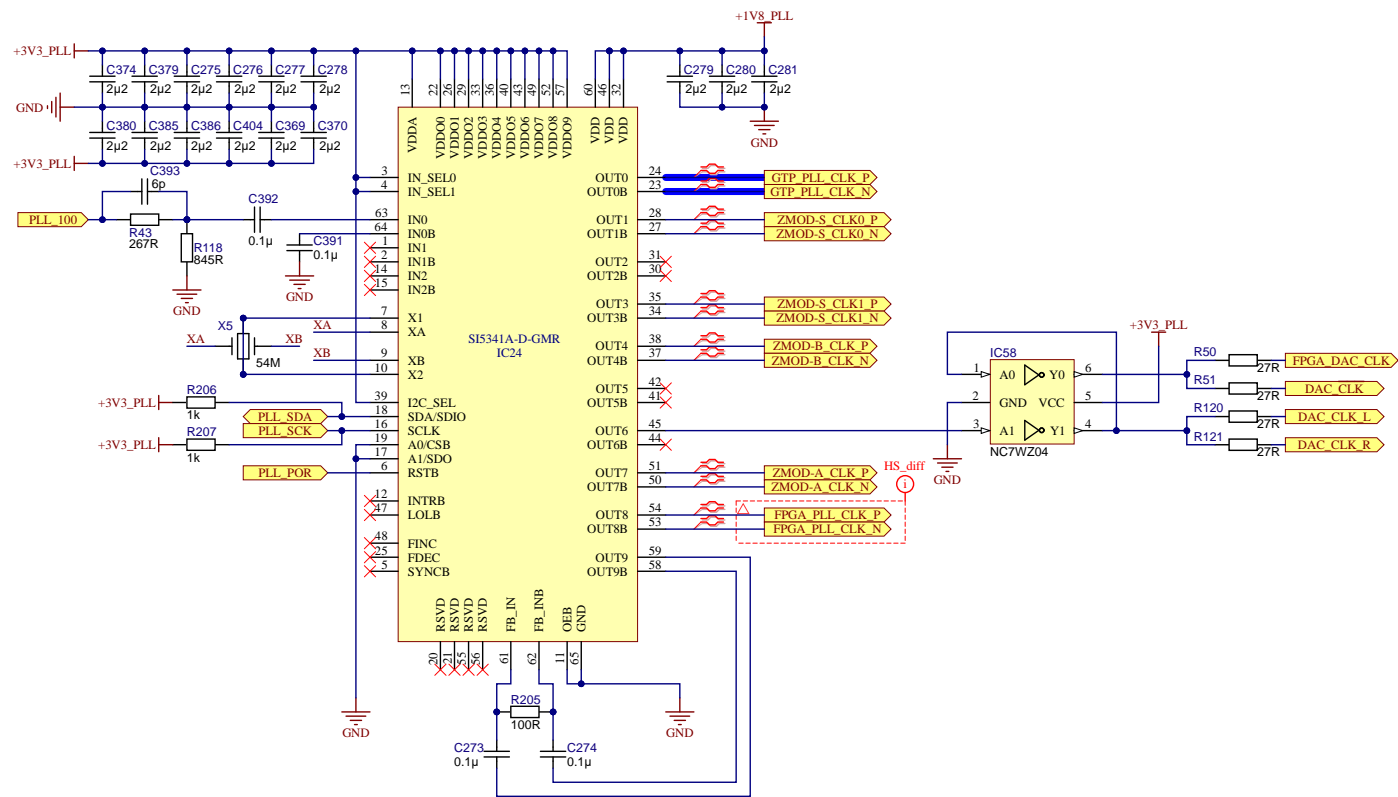
VGA R2R Dac



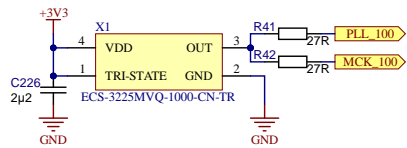
Title		
Auido & Video		
Size	Number	Revision
A3		A01
Date:	5.16.2021	Sheet 9 of 12
File:	C:\Users\... \Audio&Video.SchDoc	Drawn By: Humans



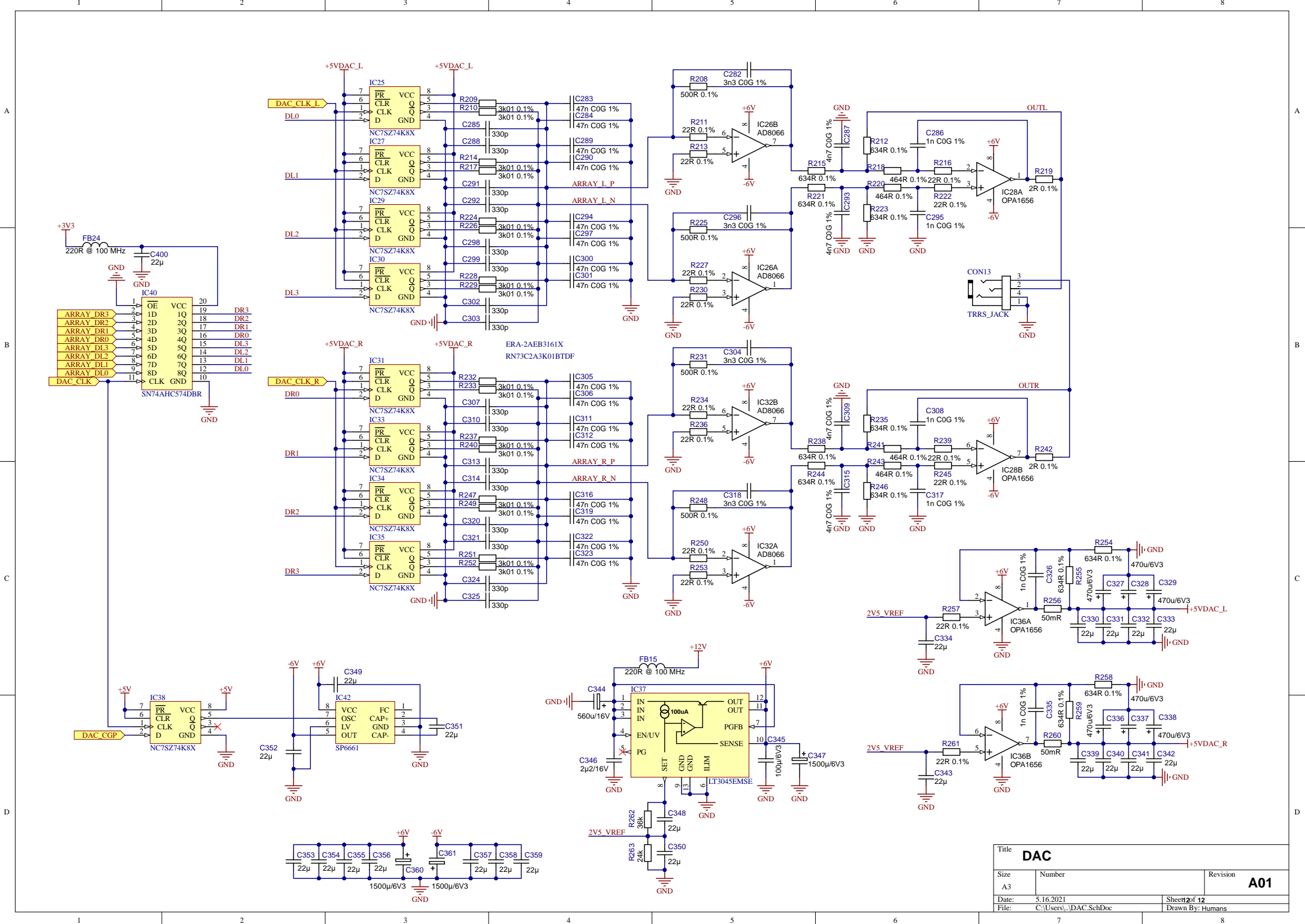
Title		
MCU & USB		
Size	Number	Revision
A3		A01
Date:	5.16.2021	Sheet# of 12
File:	C:\Users\... \MCU_USB_SchDoc	Drawn By: Humans



OSCILLATOR 100MHz

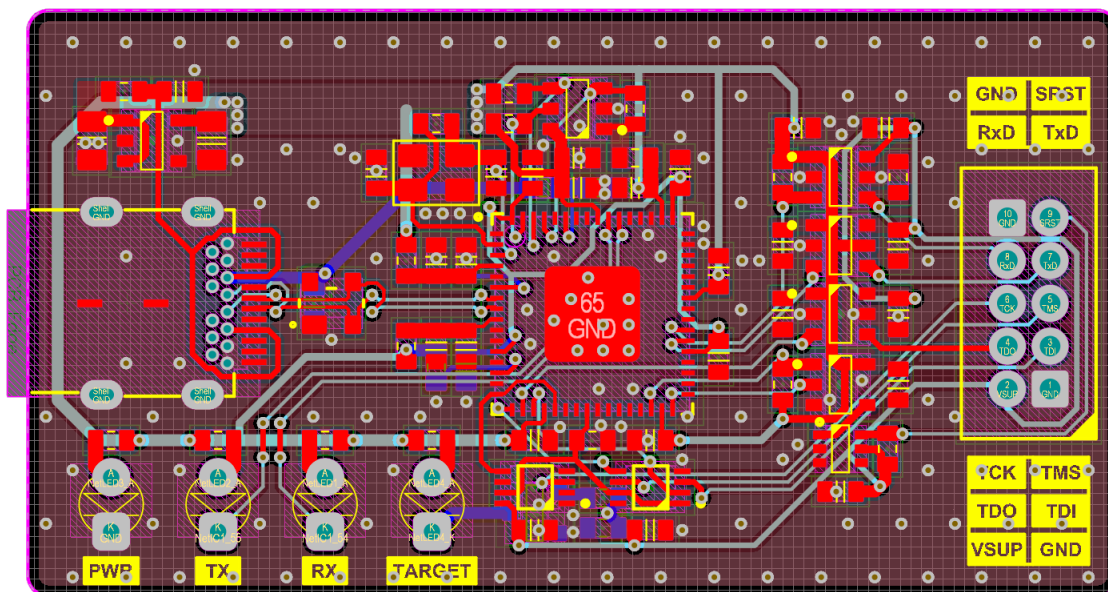


Title		CLOCK & PLL	
Size	Number	Revision	A01
A3			
Date:	5.16.2021	Sheet 1 of 12	
File:	C:\Users\... \Clock_PLL.SchDoc	Drawn By: Humans	

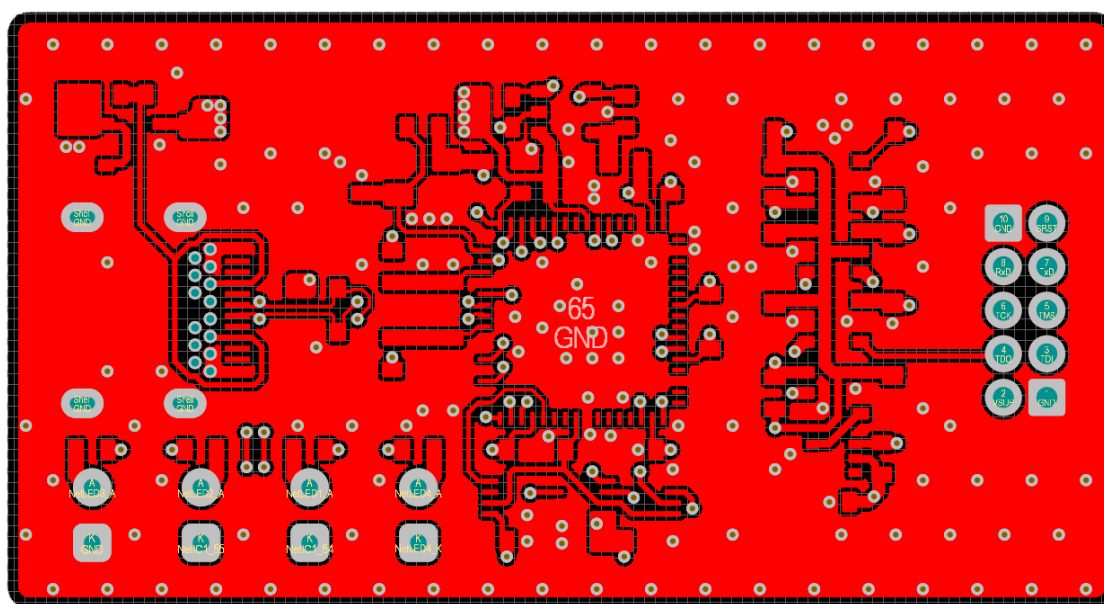


Title		
DAC		
Size	Number	Revision
A3		A01
Date:	5.16.2021	Sheet# of 12
File:	C:\Users\... \DAC.SchDoc	Drawn By: Humans

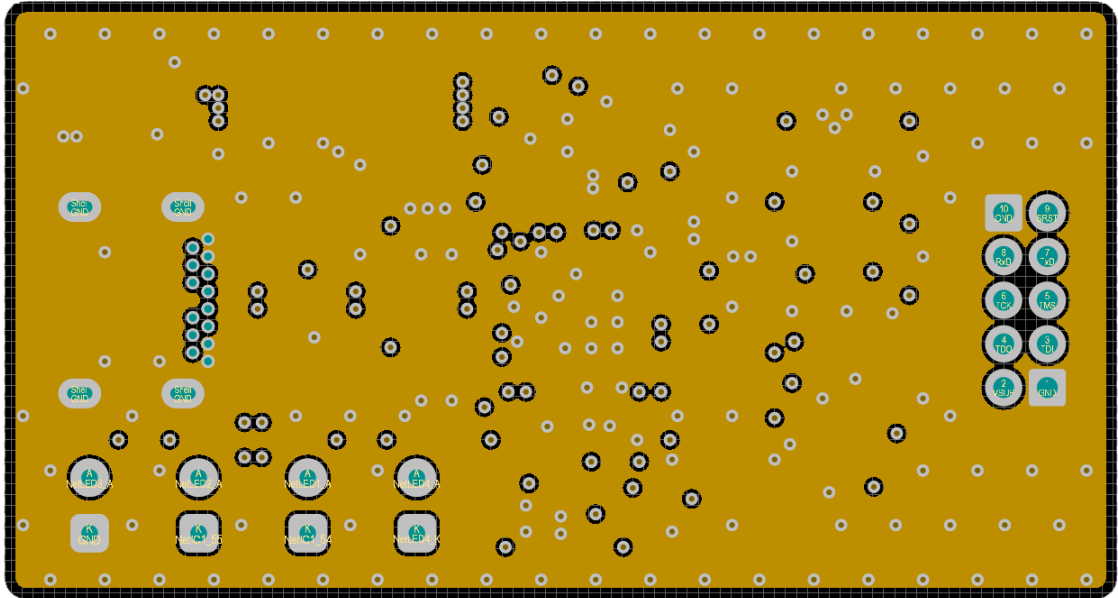
C Programovací zařízení



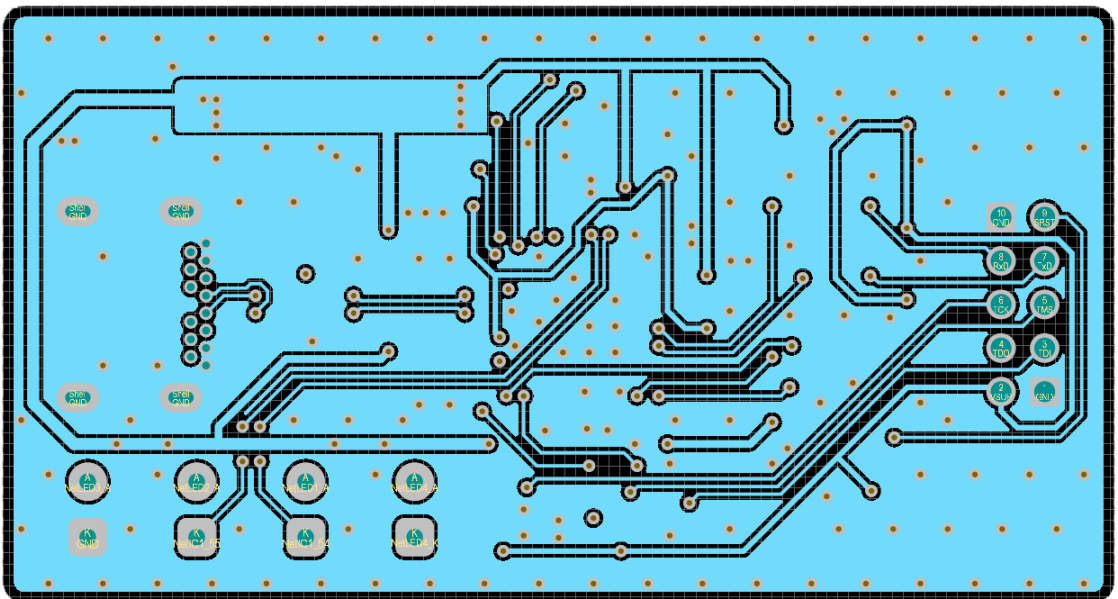
Obr. C.1: Pohled na všechny vrstvy plošného spoje



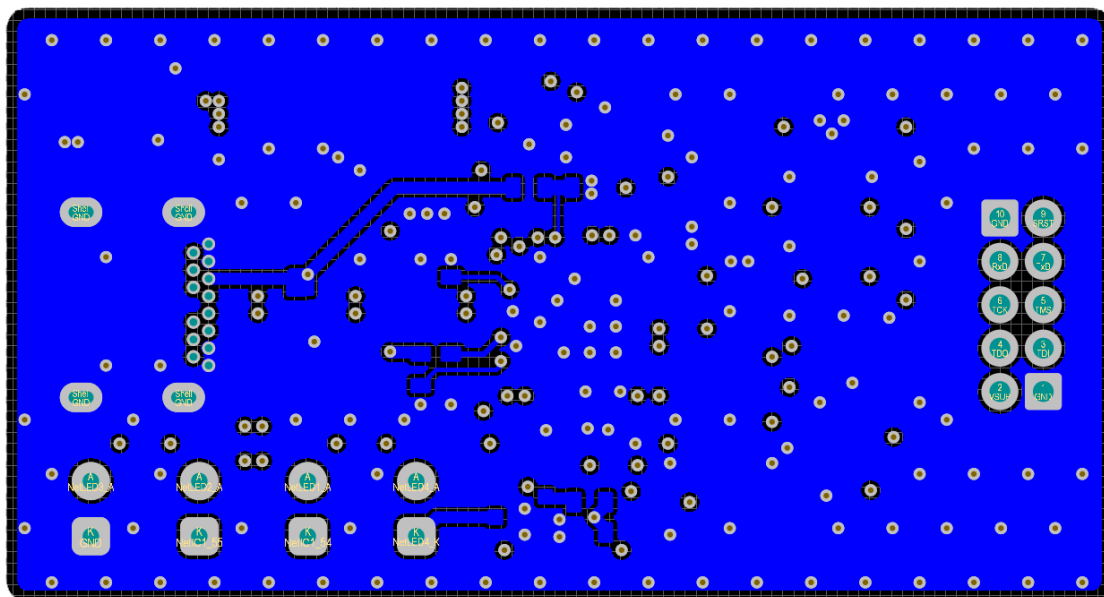
Obr. C.2: Pohled na vrstvu TOP plošného spoje



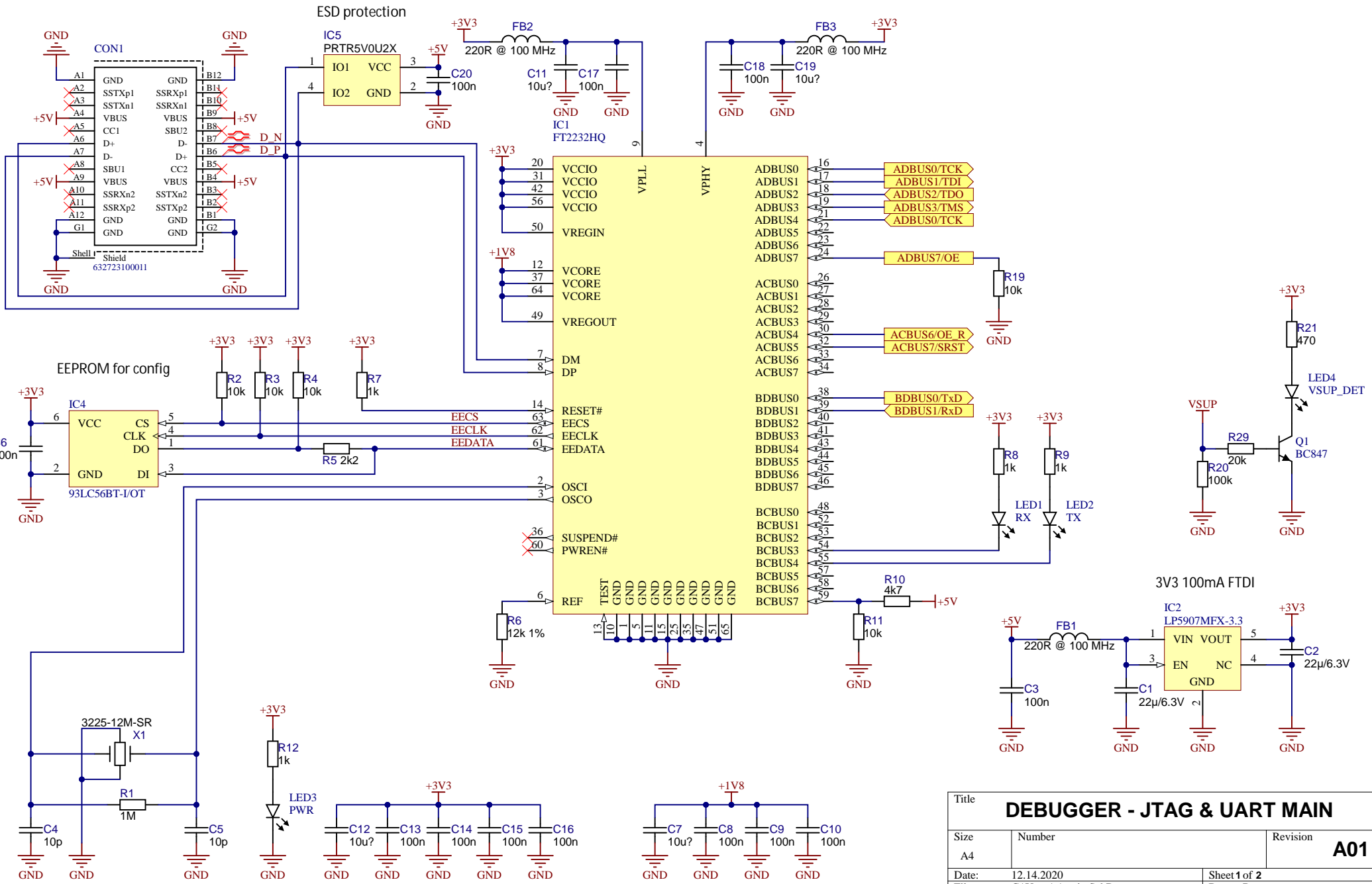
Obr. C.3: Pohled na vrstvu IN1 plošného spoje



Obr. C.4: Pohled na vrstvu IN2 plošného spoje

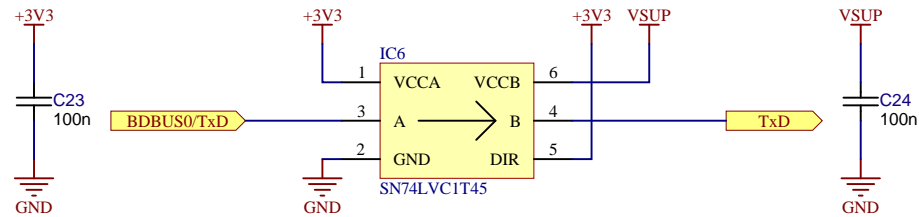
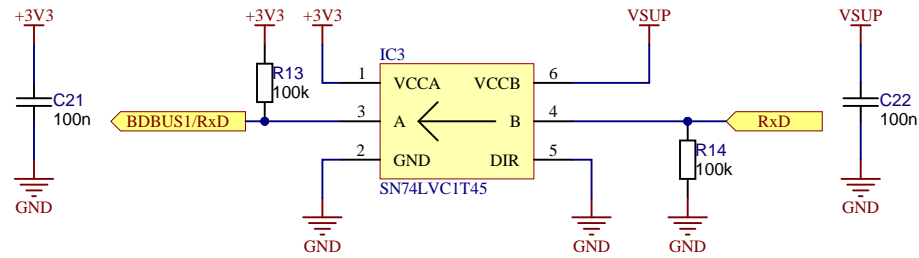


Obr. C.5: Pohled na vrstvu BOTTOM plošného spoje

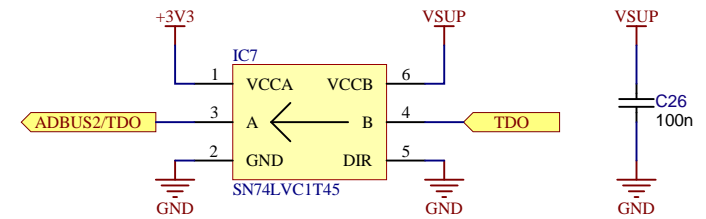
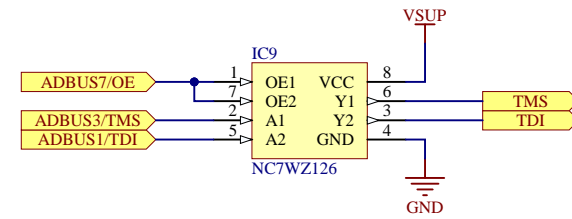
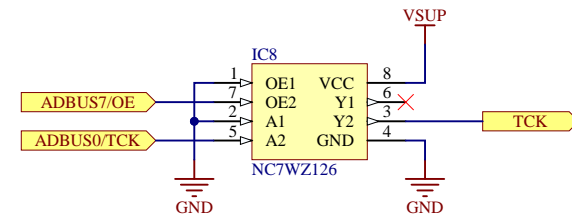


Title		
DEBUGGER - JTAG & UART MAIN		
Size	Number	Revision
A4		A01
Date:	12.14.2020	Sheet 1 of 2
File:	C:\Users\...\main.SchDoc	Drawn By: Humans

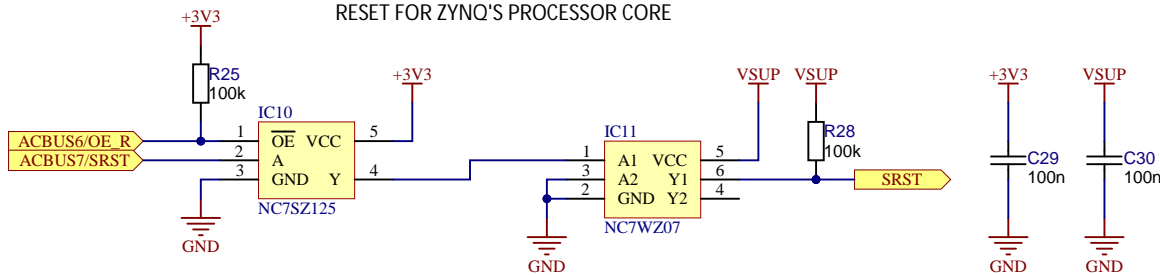
UART



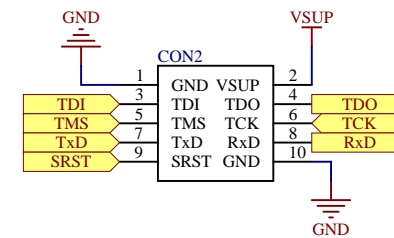
JTAG



RESET FOR ZYNQ'S PROCESSOR CORE



OUTPUT CON



Title			DEBUGGER - JTAG & UART OUTPUTS
Size	Number	Revision	
A4		A01	
Date:	12.14.2020	Sheet 2 of 2	
File:	C:\Users\...\outputs.SchDoc	Drawn By: Humans	