

**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
MASARYKŮV ÚSTAV VYŠŠÍCH STUDIÍ**

**Katedra inženýrské pedagogiky**

**Návrh a realizace virtuální laboratoře pro výuku  
programovatelných hradlových polí**

**Design and Implementation of a Virtual Laboratory  
for Teaching Programmable Gate Arrays**

**BAKALÁŘSKÁ PRÁCE**

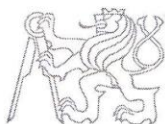
Autor: Ing. Tomáš Musil, Ph.D.

Studijní program: Specializace v pedagogice

Studijní obor: Učitelství odborných předmětů

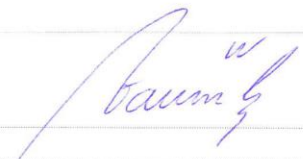
Vedoucí práce: doc. Ing. David Vaněček, Ph.D.

**Praha 2016**



## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

<b>Studijní program:</b>	Specializace v pedagogice
<b>Studijní obor:</b>	Učitelství odborných předmětů
<b>Akademický rok:</b>	2014/2015
<b>Jméno a příjmení, titul/y studenta/studentky:</b>	Tomáš Musil, Ing.
<b>Zadávací katedra:</b>	Katedra inženýrské pedagogiky
<b>Téma bakalářské práce v českém jazyce:</b>	NÁVRH A REALIZACE VIRTUÁLNÍ LABORATOŘE PRO VÝUKU PROGRAMOVATELNÝCH HRADLOVÝCH POLÍ
<b>Téma bakalářské práce v anglickém jazyce:</b>	Design and implementation of a virtual laboratory for teaching programmable gate arrays
<b>Cíl bakalářské práce: (1 – 2 věty)</b>	Cílem práce je navrhnout sadu programů pro vzdálený přístup k vývojovým prostředkům pro práci s programovatelnými hradlovými poli a realizovat ukázkové pracoviště.
<b>Stručný obsah, popis tématu, zvolených metod:</b>	<p>V teoretické části bakalářské práce bude shrnuta problematika vzdálených laboratoří a její didaktické aspekty.</p> <p>Programovatelná hradlová pole (FPGA) představují moderní a perspektivní trend v návrhu číslicových obvodů. Z tohoto důvodu začínají být zařazována do výuky předmětů s vazbou na číslicovou techniku.</p> <p>Tato technologie dovoluje studentům procvičovat si návrhové techniky z oblasti číslicových obvodů, řešit komplexní úlohy a projekty, aniž by byli nuceni trávit nezanedbatelný čas realizací obvodů pomocí klasických číslicových integrovaných obvodů (například pomocí stavebnice Dominoputer).</p> <p>Vývojové prostředky pro práci s FPGA jsou však finančně náročné - pro školy není možné je například zapůjčit studentům pro domácí a samostatné práce a pro studenty pořídit si vlastní.</p> <p>Tato práce přichází s řešením v podobě virtuální laboratoře, do které mohou studenti přistupovat pomocí veřejné sítě internet a pracovat tak vzdáleně s vývojovými</p>

	<p>prostředky, jak softwarovými, tak hardwarovými.</p> <p>V rámci práce bude nutné vyřešit:</p> <ul style="list-style-type: none"> <li>• didaktickou transformaci vybraného učiva</li> <li>• softwarové a hardwarové řešení (vzdálený přístup k PC, přihlašování uživatelů, volbu hardwarových vývojových prostředků, interface mezi hardwarovými vývojovými prostředky a PC, uživatelský interface k hardwarovým vývojovým prostředkům, kamerový systém)</li> <li>• návrh úloh</li> <li>• metodický postup práce učitele a studenta</li> </ul>
<b>Jméno, příjmení, tituly vedoucího/vedoucí bakalářské práce:</b>	David Vaněček, Doc., Ing., Ph.D.
<b>Souhlas vedoucího/vedoucí bakalářské práce:</b>	2.12.2014  (datum, podpis)
<b>Termín zadání práce:</b>	5. prosince 2014
<b>Termín odevzdání práce:</b>	6. května 2015

10.12.2014  (datum, podpis)

**Ing. Pavel Andres, Ph.D.**  
vedoucí katedry inženýrské pedagogiky  
MÚVS ČVUT



(datum, podpis)

**prof. Ing. Vladimír Kučera, DrSc. Dr.h.c.**  
ředitel MÚVS ČVUT

Potvrzení studenta/studentky o seznámení se s podmínkami pro zpracování bakalářské práce.

Datum, podpis: 2.12.2014 

### ***Prohlášení:***

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu. Nemám závažný důvod proti zpřístupnění této závěrečné práce v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) v platném znění.

V Praze dne 4.1.2016

podpis: .....

## ***Poděkování:***

Chtěl bych poděkovat doc. Ing. Davidu Vaněčkovi, Ph.D. za vedení této bakalářské práce.

Dále bych chtěl poděkovat RNDr. Bedřichu Polovi za materiální podporu poskytnutou za účelem této práce.

V neposlední řadě poděkování náleží mé rodině, především přítelkyni Petře, za jejich podporu a trpělivost.

## ***Abstrakt:***

Tato bakalářská práce je zaměřena na návrh vzdálené laboratoře pro práci s programovatelnými hradlovými poli, do které mohou žáci a studenti přistupovat pomocí veřejné sítě internet a pracovat tak vzdáleně s vývojovými prostředky jak softwarovými, tak hardwarovými. Vývojové prostředky pro práci s programovatelnými hradlovými poli jsou často finančně náročné a vzdálená laboratoř tak může být zajímavým prostředkem jak pro práci na domácích úlohách, tak pro vlastní projekty žáků a studentů.

Během řešení práce byly zjištěny a popsány specifika návrhu programovatelných hradlových polí oproti návrhu klasických číslicových logických obvodů a technické požadavky na návrh laboratoře.

Jako základní prvek laboratoře bylo zvoleno FPGA z rodiny Spartan-3A firmy Xilinx Inc. Dále byla zvolena platforma s pomocným programovatelným hradlovým polem jako rozhraním k připojenému osobnímu počítači, příslušný modul byl navržen a vyroben. Ze softwarového hlediska byla vybrána platforma pracující na bázi protokolu RFB.

Přínosem vzdálené laboratoře, kromě vzdělávacího aspektu výuky moderních technologií v oboru číslicové techniky, je možnost řešit komplexní úlohy a projekty, aniž by žáci a studenti byli nuceni trávit nezanedbatelný čas realizací obvodů pomocí klasických číslicových integrovaných obvodů.

## ***Klíčová slova:***

FPGA, vzdálená laboratoř, RFB, virtualizace

***Abstract:***

This bachelor thesis is focused on a design of a Field Programmable Gate Array remote laboratory. Students can access to this laboratory via the Internet and work with hardware and software design tools. Design tools are usually rather expensive; remote laboratory could be considered as an instrument for dealing with homework or for students' own projects.

During solving the problem, FPGA design specifics in comparison to classical digital circuits design techniques were discovered and described as well as technical requirements for the design of the laboratory.

Spartan-3A Xilinx Inc. FPGA was chosen as a main component of the remote laboratory. The platform further consists of auxiliary FPGA creating an interface to a personal computer. An appropriate electronic module board was developed and manufactured. From the software point of view the RFB protocol was chosen.

The benefit of the remote laboratory, together with the educational aspect of teaching modern digital technology, is that it provides a possibility for student to solve their own projects and tasks without spending non-negligible amount of their time with realization of the circuit with discrete integrated circuits.

***Key words:***

FPGA, remote laboratory, RFB, virtualization

## Obsah:

Abecední seznam použitých zkratk	9
Seznam obrázků	10
Seznam tabulek	10
1 Úvod	11
1.1 Úvod do problematiky	11
1.2 Fenomén vzdálených laboratoří	12
1.3 Cíle bakalářské práce	15
1.4 Členění textu	15
2 Existující vzdálené laboratoře	16
2.1 Softwarové technologie použité ve vzdálených laboratořích	16
2.2 Obecné vzdálené laboratoře	18
2.3 Vzdálené laboratoře pro práci s FPGA	20
2.4 Vzdálené laboratoře - shrnutí	21
3 Technologie programovatelných hradlových polí	22
3.1 Historie a úvod	22
3.2 Návrh programovatelných hradlových polí typu FPGA	23
4 Návrh vzdálené laboratoře	26
4.1 Vývojový systém pro návrh FPGA	26
4.2 Koncept vzdálené laboratoře	27
4.3 Systém virtualizace	29
5 Realizace vzdálené laboratoře	31
5.1 Vývojová deska s FPGA Spartan-3A	31
5.2 Virtuální vývojový modul	32
5.3 Xilinx ISE	38
5.4 Xilinx Platform Cable II	39
5.5 Logický analyzátor OMEGA	39
5.6 Osciloskop	40
5.7 Kamera Canyon CNE-CWC2	41
5.8 Editor pro psaní kódu	42
5.9 Sestava vzdálené laboratoře	42
6 Závěr	43
Seznam použité literatury	44
Přílohy	45
A Schéma zapojení a plošný spoj VLBRD-1	45
B Přiřazení vstupně výstupních pinů k FPGA Spartan-3A	48



## Abecední seznam použitých zkratk

AJAX	Asynchronous JavaScript and XML
DOM	Document Object Model
CSS	Cascading Style Sheets
CPLD	Complex Programmable logic device - typ programovatelného logického obvodu
FET	Field Effect Transistor - tranzistor řízený elektrickým polem
Flash EEPROM	Flash Electrically Erasable Programmable Read-Only Memory - blokově elektricky mazatelná a programovatelná nonvolatilní paměť pouze pro čtení
FPGA	Field-Programmable Gate Array- zákaznický programovatelné hradlové pole, integrovaný obvod obsahující matici univerzálních programovatelných buněk
GbE	Gigabite Ethernet
GPL	General Public Licence - druh licence svobodného software, která vyžaduje, aby odvozený software byl dále šířen pod touto licencí
HDL	Hardware Description Language- jazyk určený pro popis hardwarové struktury
I2C	Inter-Integrated Circuit - dvou vodičová komunikační sběrnice
IOB	Input/Output Block
LFSR	Linear Feedback Shift Register - posuvný registr se zavedenými zpětnými vazbami logické funkce XOR
LUT	Look-Up Table
ITU	International Telegraph Union
JTAG	Joint Test Action Group, standard IEEE 1149.1, původně standard pro testování osazených plošných spojů, dnes často programovací interface
LAN	Local Area Network - místní počítačová síť
LED	Light-Emitting Diode - světlo emitující dioda, svítivka
MOS	Metal Oxide Semiconductor- technologie výroby tranzistorů řízených elektrickým polem
PAL	Programmable Array Logic
PCB	Printed Circuit Board - deska tištěných spojů
RDP	Remote Desktop Protocol
RFB	Remote FrameBuffer
SPI	Serial Peripheral Interface - rozhraní pro sériový přenos dat
SRAM	Static Random Access Memory - statická paměť pro náhodný přístup
VNC	Virtual Network Computing
WAN	Wide Area Network - rozsáhlá počítačová síť, například internet
XHTML	eXtensible Hypertext Markup Language

## **Seznam obrázků:**

- Obr. 1: Typická struktura programovatelného hradlového pole
- Obr. 2: Koncepce vzdálené laboratoře
- Obr. 3: Blokové schéma vzdálené laboratoře
- Obr. 4: Vývojová deska Xilinx Spartan-3A/3AN Starter Kit Board
- Obr. 5: Aplikace Virtuální deska FPGA
- Obr. 6: Univerzální deska XC6SLX Board
- Obr. 7: Deska VLBRD-1
- Obr. 8: Sestava desek XC6SLX Board a VLBRD-1
- Obr. 9: Protokol vysílání dat
- Obr. 10: Protokol příjmu dat
- Obr. 11: Blokové schéma generátoru stisku/uvolnění tlačítka
- Obr. 12: Xilinx Platform Cable II
- Obr. 13: Logický analyzátor OMEGA
- Obr. 14: Softwarová aplikace logického analyzátoru OMEGA
- Obr. 15: Aplikace TrueRTA
- Obr. 16: Kamera Canyon CNE-CWC2
- Obr. 17: Vzdálená laboratoř

## **Seznam tabulek:**

- Tab. 1: Seznam vstupů a výstupů
- Tab. 2: Protokol vysílání dat
- Tab. 3: Protokol příjmu dat

# Kapitola 1

## Úvod

### 1.1 Úvod do problematiky

Jedním ze základních prvků kurikula střední odborné školy je osvojování si praktických zkušeností z konkrétního oboru. Elektronika jako obor zažívá již delší dobu velmi rychlý rozvoj, co se týče technologií a používaných elektronických součástek. Pro žáky a studenty jak středních odborných škol, vyšších odborných škol tak vysokých škol není tato situace jednoduchá. Osvojit si základy, a byť jen rámcové znalosti průřezem tohoto oboru je, řečeno zjednodušeně, běh na dlouho trať. Tento problém se však netýká pouze žáků a studentů, ale také pracovníků, kteří připravují studijní plány a vyučujících. Zaměříme-li se na střední odborné školy, je tento problém ještě markantnější, pokud nebudeme studium na střední odborné škole považovat pouze za přípravný kurz pro studium na vysoké škole. Žáci středních odborných škol by pak měli být seznamováni s technologiemi, které se v dané průmyslové oblasti běžně používají.

Při výuce číslicové techniky zaměřené na použití programovatelných hradlových polí se můžeme setkat s následujícími obtížemi:

- poměrně drahé vývojové přípravky v případě realizace laboratoře sestávající se z více pracovišť
- dostupné vývojové přípravky nejsou dostatečně robustní na zacházení ve školním prostředí, například z důvodu vlivu ESD (Electrostatic Discharge)
- vývojové prostředky není možné žákům a studentům zapůjčit, aby na nich mohli vytvářet vlastní projekty
- komplikovaný návrhový software, který je zároveň náročný na instalaci a hardwarové prostředky

Všechny tyto problémy lze řešit pomocí vzdáleného přístupu k laboratoři.

Na tomto místě bych chtěl upozornit na terminologickou odlišnost, ke které dochází v rámci této práce. Během zkoumání existujících laboratoří a detailnějším seznámení se s tímto oborem jsem dospěl k následujícímu názoru. Abych se ve své práci nelišil od zažité terminologie v oboru, budu v tomto dokumentu pro návrh virtuální laboratoře v rámci zadání práce používat termín vzdálená laboratoř (remote laboratory), tento termín vychází například z práce Vaněčka [1][2], shodnou terminologii uvádí ve své práci Látal[3].

## 1.2 Fenomén vzdálených laboratoří

Popularita vzdálených laboratoří roste spolu s rozšířením internetu a uvědoměním si jeho možností ve výuce. Vznikají laboratoře, které na mezinárodní úrovni spojují a zaštiťují jednotlivé samostatné vzdálené laboratoře, v tomto kontextu můžeme zmínit například vzdálenou laboratoř iLabs provozovanou týmem MIT (Massachusetts Institute of Technology) nebo LabShare provozovanou několika univerzitami v Austrálii. V Evropě můžeme zmínit například projekt LiLa (Library of Labs) podporovaný Evropskou komisí v rámci iniciativy Evropa 2020.

Poměrně kvalitní rozbor a porovnání jednotlivých typů laboratoří (reálná, virtuální, vzdálená) uvádí Nedic a kol. [4]. Výhody a nevýhody, které plynou z jednotlivých typů laboratoří a které uvádí Nedic a kol. [4] a Schauer a kol. [5] jsou shrnuty v následujícím srovnání. Záleží však na konkrétní implementaci jednotlivých typů laboratoří a v některých případech dochází k potlačení některé z nevýhod případně výhod. Schauer a kol. [5] dále uvádí důvody, proč klesá počet klasických laboratoří.

reálná(skutečná) laboratoř

výhody:

- ovládání, používání a práce s reálnými zařízeními
- učitel je přítomný během práce a může poskytnout konzultaci, radu
- možná spolupráce s ostatními žáky/studenty
- reálné prostředí a data

nevýhody:

- nutnost zajistit pro výuku v daném termínu v rámci rozvrhu laboratoř a učitele
- náklady na laboratoř a to jak investiční, tak provozní

simulovaná(virtuální) laboratoř

výhody:

- výhodné pro účely demonstrací jevů, přístupů, konceptů
- žádné časové omezení a blokování prostor
- interaktivita
- nízké náklady

nevýhody:

- data nejsou reálná
- chybí práce s reálnými přístroji
- chybí spolupráce s ostatními žáky/studenty

vzdálená laboratoř

výhody:

- ovládání a používání reálných komponent
- reálná data
- žádné časové omezení a blokování prostor

- nízké náklady

nevýhody:

- práce neprobíhá fyzicky v laboratoři

Z výše uvedeného porovnání jednotlivých typů laboratoří lze pozorovat průnik výhodných vlastností u vzdálených laboratoří, ke stejným závěrům dochází Williams a kol. [6], který popisuje trend postupného nahrazování reálných a virtuálních laboratoří laboratořemi vzdálenými.

Samotná simulace experimentu jednoduše nepřináší takovou hodnotu (výsledkovou, pedagogickou) jako reálný (byť vzdálený) experiment, v některých případech je však obtížné či téměř nemožné a nákladné sestavit reálný experiment. Navíc simulace může být například jednoduše spuštěna s upravenými parametry a tato změna parametrů zkoumána, což může být opět obtížné v případě reálného experimentu.

Williams a kol. [6] přichází s tezí, ve které tvrdí, že vzdálené laboratoře mohou přesvědčit studenty kombinovaných programů, aby přešli ke studiu prezenčnímu. Tato teze by se dala rozšířit na vliv vzdálených laboratoří při prezentaci konkrétní školy za účelem zvýšení povědomí o úrovni a možnostech školy a tak zvýšení šancí pro získání kvalitních žáků a studentů. Tento aspekt by při vývoji vzdálené laboratoře neměl být zanedbáván.

V ideálním případě by pro žáky či studenty neměl být z hlediska vzdělávacího procesu žádný rozdíl mezi vzdálenou laboratoří a skutečnou laboratoří.

Williams a kol. [6] navrhuje na základě jak kvalitativního tak kvantitativního výzkumu zkoumat tři modely laboratoří (a - fyzické laboratoře na půdě univerzity pro skupinovou výuku, b - počítačové simulace pro individuální výuku, c - vzdálená laboratoř pro individuální výuku) s hypotézou výzkumu v podobě tvrzení, že s pomocí vzdálené laboratoře lze dosáhnout srovnatelných a za některých okolností dokonce kvalitnější vzdělání studentů než v případě navštěvování klasických laboratoří, a pomocí tohoto výzkumu testovat zpětnou vazbu k návrhu konkrétní vzdálené laboratoře.

Garcia-Zubia a kol. [7] tvrdí na základě porovnání úspěšného či neúspěšného absolvování předmětu, v rámci něhož skupina prezenčních studentů řešila úlohy v klasické laboratoři a skupina kombinovaných studentů řešila úlohy pomocí vzdálené laboratoře typu WebLab-DEUSTO (viz kapitola 2.3.3) - kdy byla výrazně vyšší úspěšnost absolvování předmětu ve skupině studentů pracujících ve vzdálené laboratoři, že tato výuková metoda je tedy z hlediska efektivity učení výhodnější. V tomto hodnocení nicméně úplně postrádám diskuzi vlivu rozdílných charakteristik členů obou skupin, a výsledky tedy nepovažuji za relevantní. Návrh vhodného modelu, který lze aplikovat na výzkum v oblasti porovnání efektivity výuky v klasické laboratoři a ve vzdálené laboratoři, který zohledňuje mnohé

aspekty, které mohou z pohledu výběru studentů tak rozdílnosti porovnávaných laboratoří ovlivnit daný výzkum, přináší Corter [8].

Virtuální a vzdálené laboratoře určené pro žáky a studenty technicky zaměřených středních odborných, vyšších odborných či vysokých škol mohou plnit více druhů didaktických aspektů.

#### Virtuální a vzdálené laboratoře jako součást e-learningu:

Virtuální a vzdálené laboratoře jsou ideálním prvkem pro doplnění e-learningové výukové metody při výuce prakticky orientovaných předmětů či předmětů, kde je experiment vítaným - aktivizujícím prvkem. Virtuální nebo vzdálená laboratoř může být v tomto případě integrální součástí e-learningového kurzu a její úspěšný průchod pak podmínkou nutnou (nikoli postačující) pro úspěšné splnění cílů výukového kurzu.

#### Problémová metoda výuky ve virtuálních a vzdálených laboratořích:

Problémová metoda výuky by měla cílit na rozvinutí znalostí v dané problematice pomocí následujícího několikastupňového schématu:

- definice problému
- podrobná definice problému, definice subproblémů
- nalezení rámcového řešení, případná tvorba hypotéz
- ověření hypotéz
- praktické ověření řešení problému

Jednotlivé kroky mohou být procházeny iterativně v případě, že je během řešení narazí řešitelé na problém či rozpor v hypotézách.

Úlohy pro virtuální a vzdálené laboratoře mohou být s výhodou navrženy pro splnění předpokladů problémové metody výuky. Hluběji se problémovou metodou ve vztahu k virtuálním laboratořím zabývá Orduna a kol. [9].

#### Metoda projektů s využitím virtuálních a vzdálených laboratoří:

Metoda projektové výuky má ve své podstatě aspekty problémové metody výuky. Jedná se o dlouhodobější projekty, na kterých může pracovat širší tým žáků/studentů.

#### Přímá podpora praktických cvičení:

Virtuální či vzdálené laboratoře mohou v přímé podpoře praktických cvičení zastávat dvě základní úlohy:

- primární laboratoř - v případě, kdy se jedná o práci se zdravím ohrožujícím materiálem či přístroji, které nemohou být z legálních důvodů obsluhovány osobami bez příslušného vzdělání nebo nesplňujícími minimální věkový limit. Klasickým příkladem mohou být experimenty, které pro své provedení vyžadují práci s radioaktivním materiálem, vysokým napětím, výkonnými lasery.
- doplňková laboratoř - laboratoř sloužící pro samostatné procvičení úloh nebo pro vypracování samostatných domácích úloh. Tato laboratoř může být koncipovaná jako

samostatná a uzavřená pro přímé použití (výuka přímo probíhající s fyzickým využitím prostředků v učebně, laboratoři) nebo se může jednat o laboratoř kombinovanou, kdy stejné přípravky a laboratorní přístroje mohou být použity pro přímé využití v době konání seminárních nebo laboratorních cvičení a mimo tuto dobu mohou být použity právě ve formě virtuální nebo vzdálené laboratoře. Tento přístup přináší jak výhody, tak i negativní aspekty. Výhodou je, že žák/student může používat jednotnou platformu se kterou je dobře seznámen. Další výhodou je ekonomická, kdy je vybavení, může se jednat o poměrně vysoké investice, optimálně využíváno. Nevýhodou kombinované laboratoře je nutnost jejího zafixování, kdy žáci/studenti nemohou měnit topologii a propojení laboratoře během provádění experimentů či prací na vývojových deskách. Rigidní podoba je nutná právě z důvodu využití pro potřeby vzdálené laboratoře.

### 1.3 Cíle bakalářské práce

Hlavním cílem této práce je navrhnout vzdálenou laboratoř pro práci s programovatelnými hradlovými poli a podpořit tak výuku předmětů zaměřených na číslicovou techniku, jako jsou například předměty "Číslicová technika", "Technické vybavení" (předměty uvedené ve školním vzdělávacím plánu SPŠE Ječná).

Dílčí cíle práce jsou zjištění současných existujících vzdálených laboratoří určených pro výuku a to jak obecných, tak případně zaměřených na FPGA a dále zjištění specifik programovatelných hradlových polí a návrhového systému. K těmto specifikám je nutné přihlídnout během návrhu laboratoře, která by měla v co největší míře odpovídat reálné práci s FPGA.

Souhrnným cílem této práce je pak zvýšit povědomí o programovatelných hradlových polích jako o moderních technologiích a elektronických součástkách, které se dnes již běžně používají v technické praxi.

### 1.4 Členění textu

V kapitole 2 jsou zmíněny existující provozované vzdálené laboratoře určené pro podporu výuky na středních odborných případně na vysokých školách. V první části kapitoly jsou uvedeny vzdálené laboratoře bez ohledu na oborové zaměření, v druhé části kapitoly pak vzdálené laboratoře určené pro podporu výuky programovatelných hradlových polí.

V kapitole 3 je uvedena stručná historie technologie programovatelných hradlových polí a rozbor odlišností návrhu FPGA od návrhu klasických číslicových logických obvodů.

Kapitola 4 se zabývá koncepcí a potřebnými prvky, které jsou pro funkci vzdálené laboratoře pro podporu výuky programovatelných hradlových polí podstatné a kapitola 5 se pak zabývá konkrétně zvolenými prvky a prostředky jak standardně dostupnými tak navrženými pro konkrétní účely této laboratoře.

V kapitole 6 je pak provedeno shrnutí a možnosti pro pokračování projektu.

V příloze A je schéma navržené desky VLBRD-1 a motiv plošného spoje. Příloha B obsahuje přiřazení vstupně/výstupních pinů k FPGA vývojové desky Xilinx Spartan-3A/3AN Starter Kit Board.

# Kapitola 2

## Existující vzdálené laboratoře

### 2.1 Softwarové technologie použité ve vzdálených laboratořích

#### 2.1.1 JavaApplety

Java applet znamená aplikaci napsanou v jazyce Java, spouštěnou na osobním počítači prostřednictvím Java Virtual Machine mimo vlastní prohlížeč, ze kterého ale bývá volána.

#### 2.1.2 JavaSkript

JavaSkript je objektově orientovaný skriptovací interpretovaný programovací jazyk určený pro použití v rámci internetových stránek původně vyvinutý ve společnosti Netscape, ale v současné době velmi rozšířený a podporovaný prakticky všemi běžnými internetovými prohlížeči. Spouštění skriptu probíhá typicky na straně uživatelského osobního počítače.

#### 2.1.3 AJAX

AJAX (Asynchronous JavaScript and XML) je ve skutečnosti balík samostatně existujících webových technologií jako jsou:

- JavaScript
- XHTML (eXtensible Hypertext Markup Language)
- DOM (Document Object Model)
- CSS (Cascading Style Sheets)

a k tomuto balíku standardních komponent přidává technologii XMLHttpRequest. XMLHttpRequest umožňuje komunikaci mezi klientem a serverem, komunikace probíhá prostřednictvím HTTP (Hypertext Transfer Protocol) protokolu. Tato technologie umožňuje měnit obsah na části www stránky v prohlížeči na uživatelském osobním počítači bez nutnosti načítat znovu celou stránku.

Neboť jsou technologie použité v balíku AJAX standardními součástmi moderních internetových prohlížečů, které jsou již běžně dostupné také na mobilních zařízeních, je tato platforma jednoduše použitelná i na mobilních zařízeních.

#### 2.1.4 RDP

RDP (Remote Desktop Protocol) je integrovaná služba v systémech Microsoft Windows, vyvinutá firmou Microsoft. Umožňuje virtualizaci vzdálené plochy a vzdálený přístup ke vstupním zařízením systému (např. myš a klávesnice) a výstupům (tiskárna). V tomto případě přebírá z pohledu RDP vzdálený systém funkci serveru a aplikace spuštěná na operačním systému počítače, který k serveru vzdáleně přistupuje, přebírá funkci klienta. RDP pracuje na bázi a přináší rozšíření protokolů rodiny T.120 - standard definovaný ITU



(International Telegraph Union) jako protokol určený pro multimediální komunikace. Snímání obrazu na straně serveru zajišťuje speciální grafický ovladač. Nasnímaná data ve formě bitmapy nejsou pouze triviálně odeslána směrem ke klientské aplikaci, kromě komprese dat jsou využívány takzvané vyrovnávací paměti, díky kterým není třeba přenášet kompletní informaci o podobě zdrojové pracovní plochy, ale například informace o změnách, a přenášení objektů (nepřenáší se bitová mapa, ale příkazy). Tyto funkce zajišťují pro uživatele přijatelnou rychlost odezvy virtualizovaného systému na dnes běžných rychlostech připojení k internetu (od jednotek MBit výše). Přenos dat probíhá paralelně pomocí vícenásobného spojení a je šifrován pomocí algoritmu RC4. Protokol RDP podporuje práci se schránkou (CTRL+C, CTRL+V) na úrovni možnosti přenosu textů a grafických prvků.

## 2.1.5 RFB

RFB (Remote FrameBuffer) je protokol pro virtualizaci a vzdálený přístup pracovní ploše operačních systémů, které disponují grafickým interfacem. Tento protokol byl původně vyvinut firmou Olivetti a byl postupně doplňován užitečnými funkcemi, jako je přenos souborů nebo efektivnější komprese přenášených dat. V současné době lze najít produkty, které pracují na principu protokolu RFB a jsou nabízeny v podobě open-source. RFB protokol je založen na komunikaci server-klient, kde server pracuje na straně počítače a operačního systému, ke kterému je vzdáleně přistupováno. Nasnímaná data jsou pak po zakódování (kompresi) posílána ve formě obdélníků z pracovní plochy na straně serveru. Flexibilita v možnosti nastavení parametrů obdélníka z pracovní plochy během snímání a nastavení různých druhů kódování dat umožňují efektivní využití komunikační linky tak, aby byl nastaven rozumný kompromis mezi kvalitou zobrazení a rychlostí reakce systému, případně množstvím přenášených dat, pokud je toto důležitý parametr (například pokud je použita komunikační linka, která je účtovaná podle množství přenesených dat). Míra flexibility je dokonce taková, kdy se server a klient mohou vzájemně domluvit na používání takového kódování či zabezpečení přenosu dat, která nejsou součástí vlastního protokolu. V dnešní době RFB protokol umožňuje kromě standardní funkce virtualizace vzdálené plochy a ovládání vstupních zařízení vzdáleného počítače (klávesnice, myš) také obousměrný přenos souborů a práci se schránkou (CTRL+C, CTRL+V). Detailní popis protokolu RFB lze nalézt v [10].

Protokoly RDP a RFB dosahují při současných možnostech připojení k síti internet (jednotky MBit a výše) podobných parametrů. Protokol RDP na rozdíl od protokolu RFB používá přenášení objektů, tato vlastnost, byť je z hlediska velikosti datového toku výhodná, tak má některé nedostatky z hlediska implementace v operačních systémech Microsoft Windows a to, že se za jistých okolností nepřenáší varovná hlášení, která jsou jinak zobrazena na pracovní ploše počítače, ke které je přistupováno. U řešení na bázi protokolu RFB uživatel vidí přesně to, co je zobrazeno na vzdálené ploše, což považuji za výhodu.

## 2.1.6 VNC

VNC (Virtual Network Computing) je softwarový balík, který se skládá ze serveru a klienta, dohromady zajišťují vzdálený přístup k pracovní ploše a ovládání vstupů počítače (například myš, klávesnice). VNC je přímo založen na protokolu RFB. Existuje několik variant balíků VNC, které se o sebe vzájemně liší některými vlastnostmi a především pak druhem licence, pod kterou jsou distribuované (placená, freeware, GPL, open-source). Pravděpodobně nejznámější verzí je RealVNC, které pochází od původních tvůrců protokolu RFB, dále můžeme zmínit verze UltraVNC a TightVNC. Základ v protokolu RFB a jeho většinou kvalitní implementace umožňuje vzájemnou kompatibilitu mezi jednotlivými komponentami balíků VNC, tedy serverovou částí a klientskou částí, prakticky se tak lze připojit klientem k serverové aplikaci poskytované jinou společností.

Instalace aplikace klienta odpadá v případě použití verze Java VNC (Java applet) přímo pomocí webového prohlížeče, který podporuje a má povoleno spouštění Java appletů.

## 2.2 Obecné vzdálené laboratoře

### 2.2.1 LabVIEW

Systém vyvinutý a nabízený firmou National Instruments primárně určený pro automatizaci měřicího řetězce, ale také velmi často využíván pro tvorbu vzdálených laboratoří. Jedním z důvodů, proč je použití systému LabVIEW tak populární je fakt, že je používán širokou komunitou pro vytváření laboratoří a automatizovaných měření a že se jeho licence již v rámci školního prostředí vyskytují (toto platí převážně pro vysoké školy technického zaměření, ale ani střední školy technického zaměření nejsou výjimkou).

Pokud se zaměříme na možnosti použití při implementaci vzdálené laboratoře, nejdůležitějším vizualizačním prvkem je tzv. Remote Panel. Remote Panel je na uživatelském osobním počítači možné zobrazit dvěma způsoby. Prvním je pomocí nainstalovaného prostředí LabVIEW, druhým a pravděpodobně preferovanějším způsobem (ve vztahu ke vzdálené laboratoři) je zobrazení pomocí internetového prohlížeče. V tomto případě je spojení realizováno na bázi klient-server za pomoci vestavěného webového serveru prostředí LabVIEW na straně počítače vzdálené laboratoře.

Mezi základní vlastnosti systému LabVIEW patří:

- vizuální programovací jazyk: Programování v grafickém prostředí, "G" programovacím jazyku bez použití (ale také bez možnosti použití) některého z klasických programovacích jazyků je firmou National Instruments považováno za výhodu. Psaní takového programu je dostupné i pro laiky z pohledu programování. Nicméně tento pohled je dle mého názoru trochu sporný, neboť jsou stále potřeba znalosti z oblasti algoritmizace a ty jsou většinou spojené se znalostí alespoň některého ze základních programovacích jazyků. Systém psaní programu v textové

podobě vůbec nenabízí a v určitých oblastech je grafický zápis poměrně nepraktický.

- možnost publikovat výsledky a zprostředkovat řízení pomocí Remote panel viz výše
- podpora od různých výrobců měřících přístrojů pro integraci přístrojů do prostředí LabVIEW a široká komunita uživatelů

Použití Remote Panel je základním prvkem pro tvorbu vzdálených laboratoří které pro svoji činnost využívají systém LabVIEW, na straně uživatelského osobního počítače vyžaduje instalaci pluginu v internetovém prohlížeči a přináší některé problémy a omezení:

- bezpečnost
- omezení způsobená firewallem
- zpoždění reakce systému

### **2.2.2 iSES**

iSES (Internetové Školní Experimentální Studio) je systém pro vzdálené počítačové měření a řízení vyvinutý a spravovaný týmem Matematicko-fyzikální fakulty Univerzity Karlovy v Praze (v současné době na projektu pracuje František Lustig, Jiří Dvořák a Pavel Kuriščák).

Softwarová část systému ISES WEB Control kit byla původně postavená na Java appletech. Od roku 2013 jsou vzdálené laboratoře iSES provozovány na bázi JavaSkriptů. Použití JavaSkriptů přináší poměrně příjemný vedlejší efekt, se kterým autoři systému iSES aktivně pracují, a to je možnost spustit skript a tím i řídit vzdálenou laboratoř pomocí mobilních zařízení (tablety, chytré mobilní telefony), což může být pro některé žáky a studenty v dnešní době poměrně populární.

iSES je ve své podstatě softwarově hardwarová stavebnice, která je deklarována jako otevřená. Otevřenost v tomto případě znamená, že si uživatel - autor vzdálené laboratoře může laboratoř postavit na základě dodaných hardwarových komponent a pomocí skriptů předpřipravených v rámci softwarového balíku. Základními funkcemi jsou dálkové řízení, získávání a zpracování naměřených dat. Jedná se o komerční projekt nicméně cena komponent a softwarového balíku se zdá dle katalogových cen velmi příznivá vzhledem k podpoře a množství ukázkových vzdálených laboratoří.

Tento systém se výborně hodí pro realizace vzdálených laboratoří, jejichž cílem je provádění pevně daných experimentů, které pak lze využít ve výuce předmětů, jako jsou fyzika, chemie.

odkaz: <http://www.ises.info/index.php/en/ises>

### **2.2.3 VISIR**

VISIR (Virtual Instrument Systems In Reality for remote wiring and measurement of electronics circuits on breadboard) je laboratoř vyvinutá na BIT (Blekin Institute of Technology), Švédsko, je vzdálená laboratoř zaměřená na analogové elektronické obvody.

Umožňuje vytvářet zapojení a měřit veličiny. Základem je počítač s připojenými reléovými propojovacími jednotkami, které mají za úkol realizovat propojení mezi jednotlivými komponentami (napájecí zdroj, generátor, osciloskop, uživatelské prvky či obvody). Laboratoř VISIR je realizovaná na platformě LabVIEW s pomocnými programy napsanými v C++. Přihlašovací a rezervační systém je pak navržen v PHP/MySQL. Vizualizace vzdálené laboratoře má podobu nepájivého kontaktního pole (breadboard), pomocí kterého lze propojit jednotlivé prvky elektronického obvodu.

Laboratoř VISIR je k dispozici ve formě open source. Laboratoř VISIR popisuje Tawfik a kol. [11].

odkaz:

<http://visir.ieec.uned.es/trunk/sites/electronics/public/index.php?page=AboutPage#>

## 2.3 Vzdálené laboratoře pro práci s FPGA

### 2.3.1 VLAM

VLAM - Virtuální laboratoř pro vývoj aplikací s mikroprocesory a FPGA

Prototypová laboratoř, která kombinuje klasický lokální přístup (klasická laboratoř) a virtuální laboratoř. Pro práci v módu virtuální laboratoře jsou využívány virtuální pracovní stanice, které jsou spouštěny a provozovány na fyzickém serveru. Připojení k přípravkům a vývojovým deskám je pak realizováno pomocí sítě ethernet s využitím převodníků ethernet-USB (USB over ethernet) připojených přes GbE (Gigabite Ethernet) switch.

Projekt laboratoře VLAM byl zřízen Ministerstvem školství, mládeže a tělovýchovy České republiky a schválen v rámci programů na podporu výzkumu a vývoje. Řešiteli projektu jsou firma UNIS spol. s r.o., ÚTIA (Ústav teorie informace a automatizace AV ČR), a Ústav informačních systémů na Fakultě informačních technologií VUT Brno. Podrobné informace o projektu VLAM popisuje Dulík a kol. v publikaci [12], která v rámci tohoto projektu vznikla.

Jednotlivá pracoviště se skládají z:

- vývojové desky
- osciloskopu
- generátoru signálů
- osciloskopu

V rámci VLAM laboratoře byly realizovány 2 typy pracovišť s vývojovými deskami:

typ 1:

- FIT Flexis (mikrokontroléry Freescale)
- FITkit (mikrokontrolér MSP430 firmy Texas Instruments a FPGA XC3S50 firmy Xilinx)

typ 2:

- M68EVB08GB60 firmy Freescale
- Digilent Spartan 3E-1600 s FPGA XC3S1600E firmy Xilinx

### 2.3.2 FPGA Virtual Lab

FPGA Virtual Lab - Projekt KEGA 3/5238/07

Laboratoř se vzdáleným přístupem pro experimentální ověřování činnosti složitých rekonfigurovatelných systémů/obvodů implementovaných na bázi obvodů FPGA (citováno ze stránek projektu).

Laboratoř je postavena na základě programu LabVIEW, který je nainstalován na lokálním serveru v laboratoři. Konfigurační data programovatelného hradlového pole se nahrávají pomocí FTP do speciálně určeného adresáře a konfigurace FPGA je pak spuštěna a řízena pomocí prostředí v LabVIEW. Z diagnostických prostředků jsou k dispozici:

- digitální osciloskop
- logický analyzátor
- generátor signálů

Licenční server v rámci virtuální laboratoře umožňuje provozování vývojových prostředí pro práci s FPGA, například Quartus II firmy Altera a FPGA Advantage firmy Mentor Graphics.

Virtuální laboratoř tohoto projektu je dle dostupných informací určena pro práci s FPGA firmy Altera, konkrétní typ aktuálně instalované vývojové desky a tak i konkrétní typ FPGA není možné ze stránek projektu zjistit.

odkaz: <http://www.kemt-old.fe.i.tuke.sk/predmety/FPGA/kega/index.html>

### 2.3.3 WebLab-Deusto

WebLab-Deusto je balík nástrojů pro vývoj vzdálených laboratoří vyvinutý na univerzitě v Deusto. Základem tohoto setu jsou serverově orientované knihovní prvky pro jazyky Java, .NET, LabVIEW, Python, C, C++ a knihovní prvky pro stranu uživatelského osobního počítače pro Java applety a Javaskripty.

WebLab-Deusto umožňuje integraci již existujících vzdálených laboratoří, jak uvádí Orduna a kol. [9] a Orduña a kol[13]. Tato vlastnost je hlavní předností systému. Mezi provozovanými vzdálenými laboratořemi jsou tak mimo jiné laboratoře pro práci s FPGA případně CPLD.

odkaz: <http://weblab.deusto.es/website/>

## 2.4 Vzdálené laboratoře - shrnutí

Ze zkoumání existujících vzdálených laboratoří vyplynul poměrně zajímavý aspekt a tím je spolupráce mezi různými školami, univerzitami a vzájemné sdílení laboratoří, které by bylo v případě skutečných laboratoří nemožné. Vznikají projekty, které se snaží poskytnout prostředky (převážně softwarové), které umožňují integraci různých vzdálených laboratoří pomocí jednotného portálu. Také se ukazuje, že projekty s tematikou vzdálených laboratoří mají šanci při získávání podpory z veřejných financí.

# Kapitola 3

## Technologie programovatelných hradlových polí

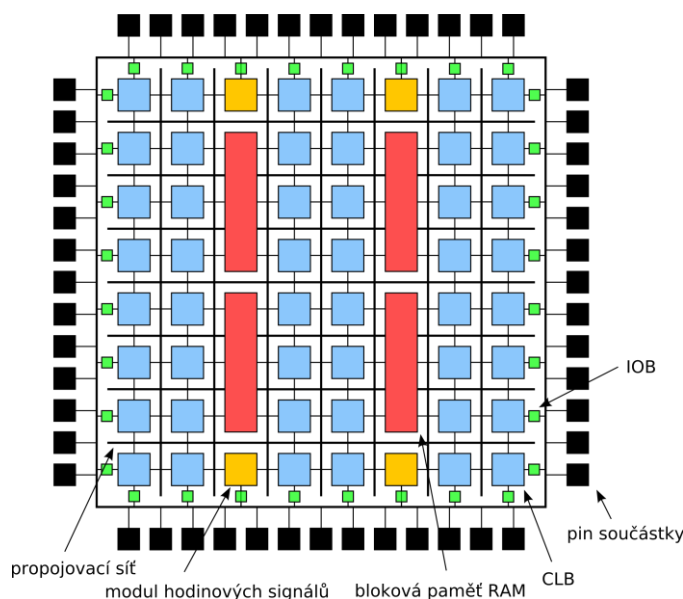
### 3.1 Historie a úvod

Historie programovatelných logických obvodů se začíná psát v 70. letech 20. století, kdy se objevují první komerčně dostupné obvody PLA (Programmable Logic Array) vyrobené technologií VLSI (Very-Large-Scale Integration). Tyto obvody umožňují implementovat logickou funkci ve tvaru součtu součinných termů (součinný term je logický výraz, který obsahuje pouze operátory typu logický součin). V roce 1978 přichází firma MMI (Monolithics Memories, Inc.) s programovatelnými logickými obvody typu PAL (Programmable Array Logic), které obsahují programovatelnou matici hradel typu logický součet, na každé logické hradlo lze přivést libovolnou kombinaci vstupních signálů s jediným omezením a tím je množství vstupů hradla logického součtu. Výstupy z programovatelné matice hradel jsou zavedeny do pevné matice hradel typu logický součin. Některé obvody typu PAL nabízí navíc integrované klopné obvody na výstupech z pevné matice hradel. V roce 1985 přichází firma Lattice Semiconductor Corp. s vylepšením programovatelných obvodů typu PAL a vzniká vícenásobně programovatelný logický obvod typu GAL (Generic Array Logic). Konečně v roce 1985 firma Xilinx, Inc. přináší první komerčně dostupné obvody typu FPGA (Field-Programmable Gate Array).

Obvody typu FPGA se od svých předchůdců liší diametrálně odlišnou architekturou a s tím také spojenou filosofií návrhu takového obvodu.

Programovatelné obvody typu FPGA ve své struktuře typicky obsahují bloky, viz obr. 1 (je použita terminologie firmy Xilinx, Inc.):

- logické bloky CLB (Configurable Logic Blocks), tyto bloky se dále skládají z generátorů logických funkcí tvořených buňkami paměti SRAM (Static Random Access Memory) ve formě vyhledávací tabulky LUT (Look-Up Table) a klopnými obvody
- propojovací síť, programovatelná matice s vysokou mírou symetrie, obsahuje spoje s různými parametry (délkou) a na tranzistorech řízených elektrickým polem (FET - Field Effect Transistor) založených spínačích. Speciální kategorií v propojovací síti jsou pak hodinové rozvody, které mají za úkol zajistit kvalitní rozvod řídicích hodinových signálů.
- vstupně/výstupní bloky IOB (Input/Output Block) které zajišťují vedení signálu od pinů (nožiček) součástky k propojovací síti v případě vstupních signálů a od propojovací sítě k pinům součástky v případě výstupních signálů
- pomocné bloky, kterými mohou být bloková paměť, obvody pro generování a úpravy hodinových signálů nebo celé komplexní jednotky (vestavěné procesory, aritmetické násobičky, bloky určené pro zpracování číslicových signálů)



Obr. 1: Typická struktura programovatelného hradlového pole

Funkci programovatelného hradlového pole určuje vnitřní konfigurační paměť, typicky typu SRAM. SRAM je typem volatilní paměti (po vypnutí napájení zapomene uložená data), je třeba pokaždé po zapnutí napájecího napětí tuto paměť naprogramovat, stejně tak po vynuceném resetu (rekonfiguraci). FPGA často nabízí různé způsoby nahrání dat do konfigurační paměti například automatické nahrání dat z externích pevných pamětí typu Flash EEPROM, většinou pak také podporují konfiguraci pomocí portu JTAG (standard, původně vyvinutém pro testování osazených desek plošných spojů).

Strukturou jednodušší programovatelné obvody než FPGA jsou obvody CPLD (Complex Programmable Logic Device), zjednodušeně se jedná o strukturu několikanásobných obvodů typu PAL. Tuto technologii zde zmiňuji, neboť je dále uvedena v příkladech existujících vzdálených laboratoří.

### 3.2 Návrh programovatelných hradlových polí typu FPGA

Návrh programovatelných hradlových polí se od návrhu klasických číslicových logických obvodů liší v zásadě ve dvou hlavních aspektech:

#### 1) množství a možnosti logických obvodů:

Programovatelné logické obvody typu FPGA nabízí uživatelům obrovské množství logických prvků. Například FPGA rodiny Virtex-7 firmy Xilinx nabízí velikosti matice logických bloků, které jsou ekvivalentní (ekvivalence se v tomto případě stanovuje velmi obtížně, jedná se tak o velmi volný a přibližný údaj) dvaceti milionům klasických logických hradel. To jsou však již obtížně dostupné elektronické součástky, převážně z finančních důvodů. Nicméně i u FPGA, které jsou finančně poměrně dobře dostupné (cena se pohybuje v řádu stokorun) se setkáváme s velikostmi ekvivalentními statisícům či jednotkám milionů

klasických logických hradel. Velikost matice konfigurovatelných logických bloků FPGA pak v podstatě znemožňuje použití klasických návrhových technik, které se učí a přednáší na středních odborných, vyšších odborných a vysokých školách v rámci předmětů, které se zabývají návrhem logických obvodů. Jedná se například o metody minimalizace (Karnaughova mapa, Svobodova mapa, přímá minimalizace úpravou logického výrazu podle pravidel Booleovy algebry), úprava logické funkce pomocí Rottovy mřížky atp. Tyto metody je možné v rámci FPGA použít pouze na návrh triviálních funkcí, pro které je ale použití FPGA prakticky neopodstatněné, i tak se jedná o filosoficky jiný způsob návrhu než je pro FPGA standardní. Standardní cesta návrhu obvodů typu FPGA spočívá v popisu navrhovaného obvodu pomocí programovacího jazyka určeného pro popis hardwaru a následném automatickém překladu pomocí návrhových nástrojů. V evropských zemích je pro popis nejčastěji volen jazyk VHDL (**V**ery-**H**igh-**S**peed **I**ntegrated **C**ircuits **H**ardware **D**escription **L**anguage), ve Spojených státech amerických pak jazyk Verilog. Existují i další způsoby popisu funkce programovatelných obvodů, jako příklad uvedu popis schématem (jedná se však spíše o blokové schéma, neboť se v něm používají i komponenty s vyšší funkcí než jen logická hradla, například čítače, posuvné registry, atp.). Obecně se doporučuje používat popis pomocí schématu při seznamování se s programovatelnými obvody, v technické praxi pak naprosto převládá popis pomocí programovacích jazyků. Kromě vlastního popisu funkce je dále třeba přiřadit jednotlivé vstupně/výstupní piny součástky daným funkcím a definovat časové závislosti (zpoždění, maximální frekvence).

Netvrdím v rámci pasáže uvedené výše, že by klasický návrh logických obvodů, tak jak se běžně vyučuje, neměl svůj smysl. Je tomu právě naopak. Základy návrhu kombinačních a sekvenčních logických obvodů jsou neoddiskutovatelným základem pro budoucího návrháře programovatelných logických obvodů, nicméně pro úspěšné uplatnění v technické praxi je třeba znát právě i moderní způsoby návrhu logických obvodů.

Aby byl žák či student schopný samostatně pracovat s programovatelnými obvody typu FPGA a navrhovat jednoduché úlohy, musí zvládat problematiku:

- popisu funkce zadaného logického obvodu pomocí programovacího jazyka určeného pro popis hardwaru, alternativně popis pomocí blokového schématu
- strukturu typického programovatelného hradlového pole typu FPGA
- stručný průběh automatického překladu, tak jak jej provádí návrhový systém - tento krok je důležitý z důvodu případného odhalování chyb během procesu popisu a návrhu funkce obvodu
- práce s návrhovým systémem - samozřejmě je zde rozdíl v závislosti na typu návrhového systému konkrétního výrobce FPGA či návrhových systémů určených pro obecné platformy, nicméně práce s návrhovými systémy je dle mého subjektivního hodnocení velmi podobná, tudíž se stačí seznámit s jedním návrhovým systémem (návrhový systém ISE firmy Xilinx Inc., který budu v navržené virtuální laboratoři používat je v tomto ohledu kvalitním reprezentantem).
- práce s fyzickým FPGA - například problematika nahrání konfiguračních dat



- ladění aplikace - hledání chyb v případě, že navržený obvod nepracuje tak jak má i přesto, že automatický překlad proběhl bez chyb či vážných varování, kterými dává návrhový systém najevo potenciálně problematické konstrukce v popisu funkce či v následné realizaci pomocí logických bloků a propojovacích matic (zjednodušeně řečeno).

## 2) principiální vlastnosti programovatelných logických obvodů typu FPGA:

Velmi důležitým aspektem, na který se dle mých zkušeností občas zapomíná, tkví v zásadní odlišnosti obvodu, který je realizovaný pomocí klasických logických hradel nebo pomocí logických hradel v programovatelných logických obvodech typu FPGA. Tato odlišnost spočívá v různé povaze zpoždění průchodu signálů jednotlivými prvky a propojeními.

U obvodů složených z klasických logických hradel za pomoci samostatných integrovaných obvodů (například populární řady TTL logických obvodů 7400, nebo CMOS 4000) dochází ke zpoždění signálu při průchodu logickým prvkem a zpoždění signálu při průchodu propojovacím vodičem, typicky měděným, se zanedbává. Neboť jsou u programovatelných logických obvodů typu FPGA propojovací sítě tvořeny mimo jiné také tranzistorovými spínači a propojovací sítě se nechovají jako klasická propojení tvořená metalickými vodiči, paradoxně dochází k většímu zpoždění signálu při průchodu propojovací sítí než při průchodu programovatelnou buňkou reprezentující logické hradlo. Tento zásadní rozdíl v praxi způsobuje, že obvody, které mohou fungovat, pokud je realizujeme pomocí klasických hradel a jednotlivých integrovaných obvodů nemusí fungovat při realizaci pomocí obvodů FPGA, pokud se jedná o návrh obsahující asynchronní prvky v případě sekvenčních obvodů, nebude takový obvod fungovat s pravděpodobností hraničící s jistotou. Také z tohoto důvodu je třeba se na návrh programovatelných hradlových nedívat pouze jako na prostředí, ve kterém se realizují klasicky navržené logické obvody, ale jako na svébytnou specifickou oblast návrhu číslicových obvodů, tímto způsobem je třeba nahlížet i během přípravy plánů a vlastní výuky předmětů zaměřených na číslicové obvody.

# Kapitola 4

## Návrh vzdálené laboratoře

Virtuální laboratoř pro práci s FPGA vyžaduje některá specifika například v porovnání s virtuálními laboratořemi zaměřenými na výuku fyziky či na provádění vzdálených experimentů.

Na trhu a v tomto případě, slovo trh uvádím zcela záměrně, existuje několik fungujících předpřipravených systémů, které vznikly na půdě univerzit, viz kapitola 2.3. Tyto systémy však nefungují na bázi licence GNU a jejich tvůrci je většinou nabízí ve formě hotových laboratoří na klíč nebo ve formě placených stavebnic.

Mezi výhody placených systémů patří především rychlost budování virtuální laboratoře, otestování vzájemné kompatibility použitých komponent, technická podpora ze strany poskytovatele systému, sady úloh. Nevýhodami jsou pak finanční zátěž pro zřizovatele laboratoře, případná uzavřenost systému, použité vývojové systémy a přípravy nemusí přesně odpovídat požadavkům pro výuku v daných předmětech.

### 4.1 Vývojový systém pro návrh FPGA

V projektu VLAM (kapitola 2.3.1) je zvolena metoda, kdy je vývojový systém firmy Xilinx nahrazen vlastní platformou pro návrh FPGA - grafický editor a editor jazyka VHDL, jednotlivé fáze návrhu FPGA pak tato nadstavbová platforma provádí za pomoci spouštění modulů z originálního návrhového systému. Autoři uvádí jako hlavní výhodu fakt, že originální návrhový systém je složitý pro potřeby výuky a použití jimi navržené nadstavbové platformy použití zjednoduší. S tímto přístupem a argumentací do jisté míry souhlasím, na druhou stranu je zde otázka, zda použití originálního návrhového systému při výuce nepřináší pro studenty benefit při jejich dalším profesním uplatnění v praxi. Přesto, že toto autoři projektu neuvádí, jedním z důvodů pro volbu nadstavbového systému mohl být problém navržené platformy s virtualizací grafického vývojového prostředí a z tohoto hlediska vlastní nadstavbová platforma více zapadá do zvoleného konceptu této konkrétní virtuální laboratoře (VLAM).

Z důvodu, aby se prostředky použité v navržené laboratoři co nejvíce blížily reálnému použití v technické praxi, budu využívat přímo návrhový systém od výrobce daného FPGA.

Rozhodoval jsem se mezi dvěma přístupy:

- 1) software návrhového systému je nainstalován lokálně + použití virtuální laboratoře například na základě java appletů
- 2) software návrhového systému je nainstalován na vzdáleném počítači

Nakonec jsem zvolil přístup, kdy je vývojový software nainstalován na vzdáleném počítači a to z následujících důvodů:

- instalace návrhového systému Xilinx ISE je časově náročná
- byť je systém poskytován zadarmo (ve verzi Webpack), je nutná registrace a v režimu Webpack pak probíhá internetová komunikace mezi počítačem a serverem firmy Xilinx - předávají se informace o jednotlivých návrzích jako forma politiky firmy Xilinx ve vztahu k volné licenci pro používání návrhových prostředků. Někteří žáci a studenti by s tímto faktem a následnou instalací návrhového systému na svůj počítač nemuseli souhlasit.
- návrhový systém Xilinx ISE je poměrně náročný na výkon PC a na volné místo na disku
- možnost konfigurace FPGA přímo z návrhového systému při použití originálních programovacích prostředků (tuto funkci je možné použít i vzdáleně pomocí ovladače komunikujícího přes TCP/IP, který je možné získat k originálnímu programovacímu kabelu firmy Xilinx, například k typu PlatformCable 2, ale instalace a provoz programovacího kabelu v tomto režimu jsou poměrně komplikované)

## 4.2 koncepce vzdálené laboratoře

Ladění (debugging) aplikací programovatelných hradlových polí je možné v případě virtuální laboratoře provádět z principu následujícími způsoby:

- 1) použití jediného FPGA, pro které se ladí aplikace a ve kterém se zároveň vytvoří diagnostické prostředí - logický obvod komunikující s příslušným návrhovým softwarem například pomocí rozhraní JTAG.

výhody:

- při použití vhodného softwarového vybavení - speciální logický analyzátor (pro FPGA firmy Xilinx je to program ChipScope Pro) je možné přímo sledovat signály uvnitř FPGA bez nutnosti vyvádět pomocné signály na piny FPGA a následně je analyzovat externím logickým analyzátozem

nevýhody:

- návrh je modifikován vložení diagnostického prostředí
- v případě softwaru ChipScope Pro se jedná o placený produkt

- 2) použití externího logického analyzátoru

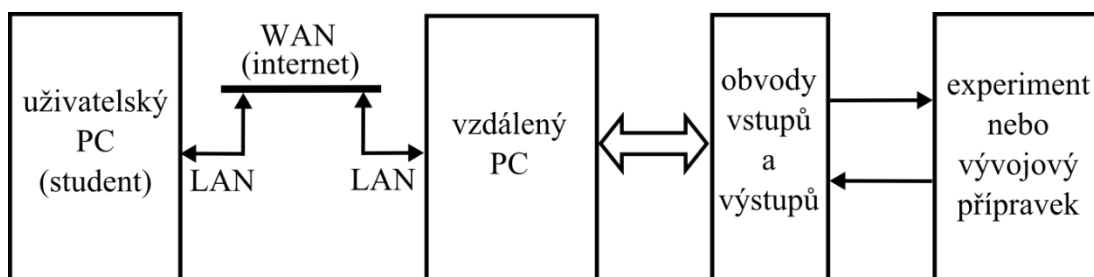
výhody:

- návrh FPGA není modifikován vloženou strukturou logického analyzátoru

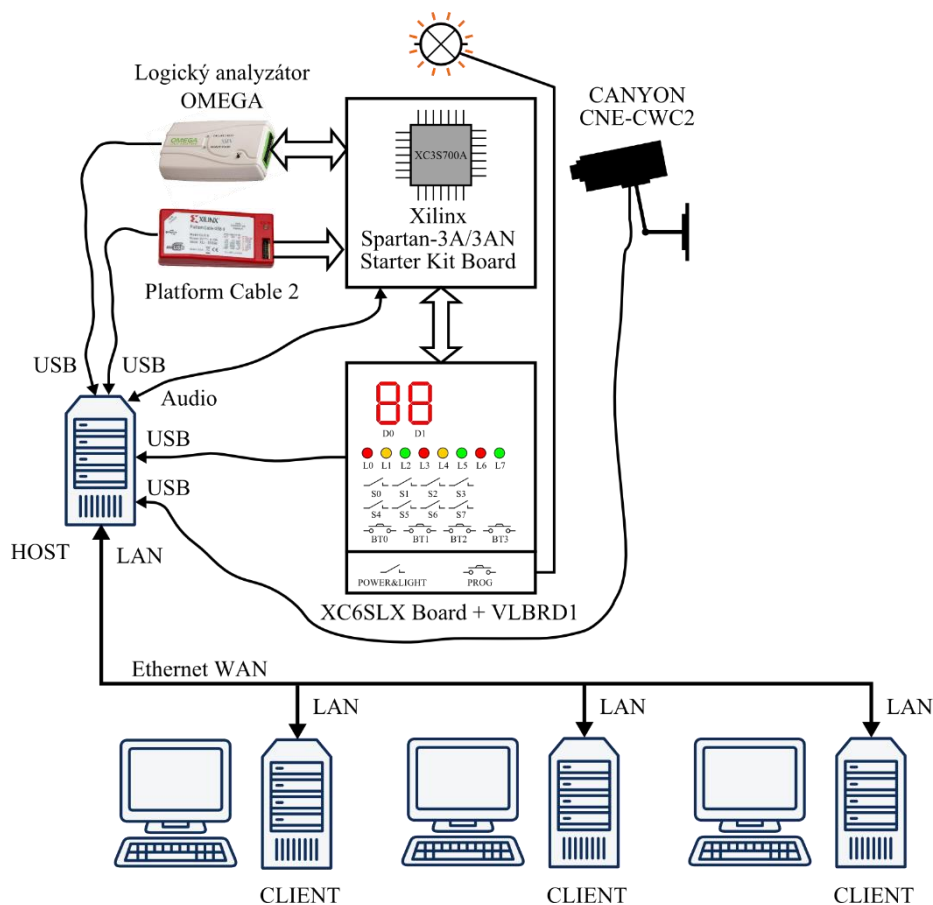
nevýhody:

- laděné signály je třeba vyvést na piny FPGA
- externí logické analyzátoři mají větší omezení v množství možných připojených signálů

Pro návrh virtuální laboratoře jsem zvolil variantu, která nevyžaduje software speciálního logického analyzátoru, který by komunikoval přímo s blokem integrovaným do návrhu FPGA. Platforma je postavena na samostatném modulu, viz obr. 2, který generuje vstupy a výstupy pro FPGA na vývojové desce. Tento externí modul, který obsahuje pomocné programovatelné hradlové pole, lze nazvat "virtuálním vývojovým modulem". Další vstupy a výstupy jsou tvořeny analyzátozem logických signálů, zvukovou kartou PC. Detailní pohled na systém vstupů a výstupů lze nalézt v blokovém schéma vzdálené laboratoře, viz obr. 3.



Obr. 2: Koncepce vzdálené laboratoře



Obr. 3: Blokové schéma vzdálené laboratoře

### 4.3 Systém virtualizace

Pro virtualizaci systému jsem vybral produkt Electromeet RemoteLab.

Electromeet RemoteLab je prostředí vzdálené laboratoře systému Electromeet poskytovaného firmou IDT Technologies. Z hlediska uživatele, který se ke vzdálené laboratoři přihlašuje, existují dvě varianty. První variantou je samostatná aplikace, kterou je možno nainstalovat na osobní počítač pod volnou licenci (stačí registrace), druhou pak spuštění ve webovém prohlížeči, který podporuje HTML5 verze 5 jazyka HTML). Prostředí Electromeet RemoteLab se skládá z několika komponent:

- virtualizace vzdálené plochy založené na protokolu RFB
- rezervační systém
- přenášení obrazu

Tento systém jsem vybral především ze dvou důvodů. Funkce serveru z hlediska uživatele, který se ke vzdálené laboratoři připojuje, přebírá server provozovaný systémem Electromeet. Na straně počítače umístěného v laboratoři tak není třeba serverové funkce instalovat. Systémem RemoteLab je k počítači v laboratoři připojen pomocí protokolu RFB a na tento počítač je tak třeba nainstalovat pouze serverovou část VNC.

Rezervační systém je plně v kompetenci serveru RemoteLab, který zajišťuje, že k počítači v laboratoři může přistupovat pouze uživatel, který má laboratoř rezervovánu.

### 4.4 Připojení k síti internet

Pro správnou funkci vzdálené laboratoře je třeba zajistit kvalitní internetové připojení a ideálně tzv. statickou a veřejnou IP adresu v rozsahu IP V4. IP (Internet Protocol) adresa je adresa, pomocí které je identifikován počítač v síti internet. Statická adresa znamená, že je počítači přidělena pevná adresa, která se tedy nemění po vypnutí a zapnutí počítače (počítači je vždy přidělena stejná IP adresa). Veřejná IP adresa znamená, že je tato IP adresa unikátní v celé síti Internet a počítač lze pomocí takové adresy jednoznačně identifikovat (adresa není takzvaně překládána). V4 je v současné době nejrozšířenější revize protokolu IP, jednou z vlastností této verze protokolu je velmi omezený rozsah adres v porovnání s dnešními požadavky na počet osobních počítačů a dalších zařízení připojených k síti internet. Není proto samozřejmé obdržet od poskytovatele připojení k síti internet statickou veřejnou IP adresu v rozsahu IP V4, pokud tato možnost existuje, jedná se většinou o zvlášť placenou službu.

V případě připojení počítače (ze softwarového hlediska se jedná o server, nicméně pro realizaci vzdálené laboratoře postačí po hardwarové stránce běžný výkonný osobní počítač) k internetu přes školní síť, je třeba zajistit součinnost se správcem počítačové sítě / IT specialistou, který má oprávnění nakonfigurovat například firewall a zpřístupnit tak potřebné porty pro komunikaci vzdálených uživatelů ze sítě internet s počítačem zpřístupňujícím laboratoř.

Pro funkci systému je třeba zajistit z pohledu připojení laboratoře k síti internet:

- instalace serverové části VNC na počítač v laboratoři
- zajistit průchodnost portu pro komunikaci VNC (typicky 5900)
- zajistit průchodnost portu pro přenos video dat z kamery (implicitně 1024)

# Kapitola 5

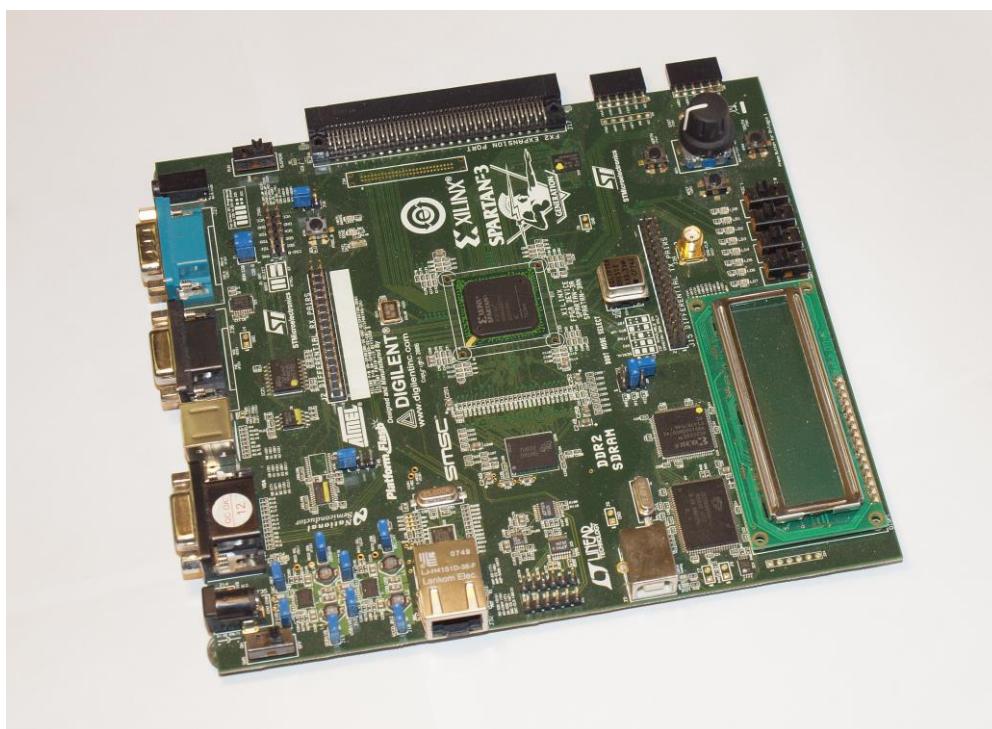
## Realizace vzdálené laboratoře

Vzdálená laboratoř se skládá především z následujících komponent popsaných níže:

- vývojová deska s FPGA Spartan-3A
- aplikace Virtuální deska FPGA
- modul VLBRD-1
- univerzální deska XC6SLX Board
- JTAG kabel Xilinx Platform Cable II
- logický analyzátor ASIX Omega
- aplikace osciloskop TrueRTA
- návrhový systém Xilinx ISE 14.7
- kamera Canyon CNE-CWC2
- osobní počítač
- systém Electromet RemoteLab

### 5.1 Vývojová deska s FPGA Spartan-3A

Jako vývojovou desku jsem zvolil Xilinx Spartan-3A/3AN Starter Kit Board viz obr. 4, který je osazen FPGA Xilinx XC3S700A řady Spartan-3A. Tato FPGA obsahuje matici 48x32 konfigurovatelných logických bloků CLB (celkem tedy tato FPGA disponuje 1472 CLB) a 360kBit blokové paměti RAM. Ekvivalentní počet hradel je přibližně 13000. Každá CLB obsahuje dva čtyřvstupové generátory logické funkce realizované vyhledávací tabulkou LUT. Více informací lze nalézt ve firemním katalogovém listu k řadě Spartan-3A [14]. Tento typ FPGA již nepatří k nejnovějším, na druhou stranu, dle vlastních zkušeností s výukou programovatelných hradlových polí, považuji za výhodu jeho jednodušší strukturu (novější řady FPGA nabízí různé integrované moduly, které strukturu vlastního FPGA činí složitější). Neboť se jedná o časem a praxí ověřené obvody, neobsahuje návrhový systém žádné chyby, které by začínajícího návrháře programovatelných logických obvodů mohly zbytečně zaskočit či dokonce odradit od další práce.



Obr. 4: Vývojová deska Xilinx Spartan-3A/3AN Starter Kit Board

Seznam vstupů a výstupů připojených k vývojové desce viz tab. 1.

prostředek	vstupy	výstupy
VLBRD-1	8x spínač 4x tlačítko	8x LED 2x sedmissegmentová zobrazovací jednotka 1x generátor hodinového signálu
logický analyzátor OMEGA	až 16 signálů	až 16 signálů
zvuková karta +TrueRTA	2	2

Tab. 1: Seznam vstupů a výstupů

## 5.2 Virtuální vývojový modul

Virtuální vývojový modul vytváří interface k vývojovému kitu "Xilinx Spartan-3A/3AN Starter Kit Board".

Virtuální vývojový modul se skládá ze softwarové a hardwarové části. Softwarová část je nazvána "Virtuální deska FPGA" a její grafický interface je na obr. 5. Tato softwarová aplikace je napsána v jazyce C++ v prostředí Borland C++ Builder.

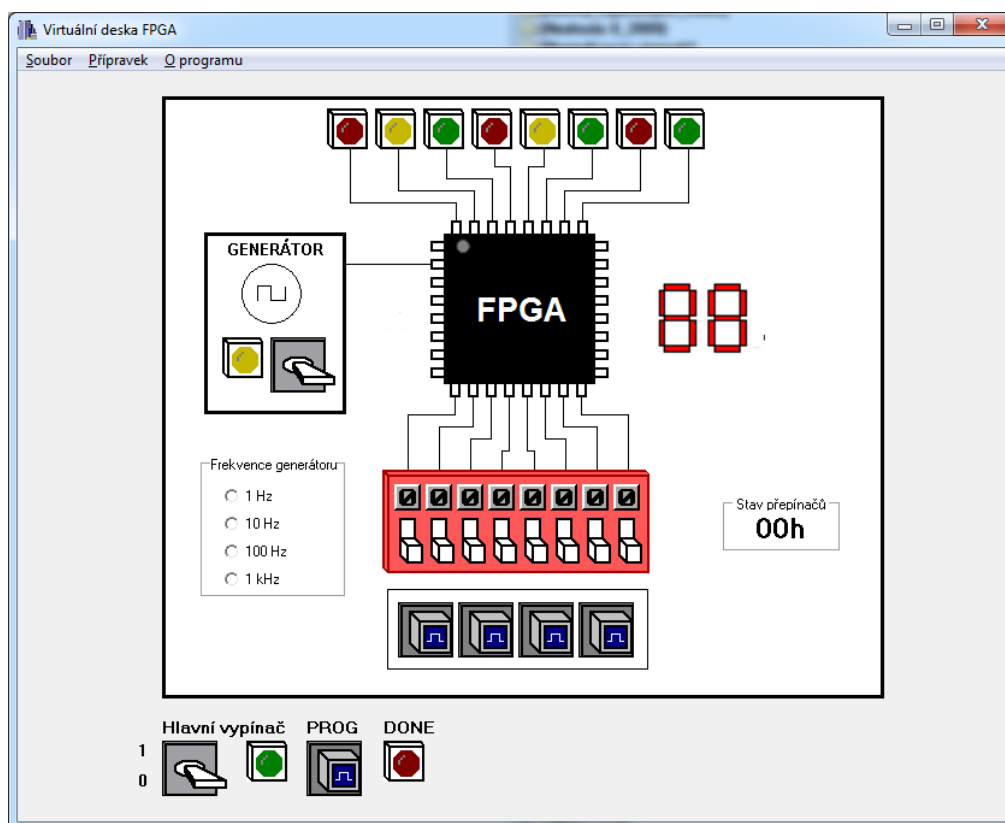


Aplikace zobrazuje přímo stav výstupů:

- dvě sedmsegmentové jednotky zapojené v přímém režimu (bez multiplexování)
- 8 LED (3ks zelené, 3ks červené, 2ks oranžové - tato kombinace dovoluje vytvořit například úlohu řízení jednoduché křižovatky s přechodem pro chodce)

Softwarová aplikace dále generuje následující vstupní signály:

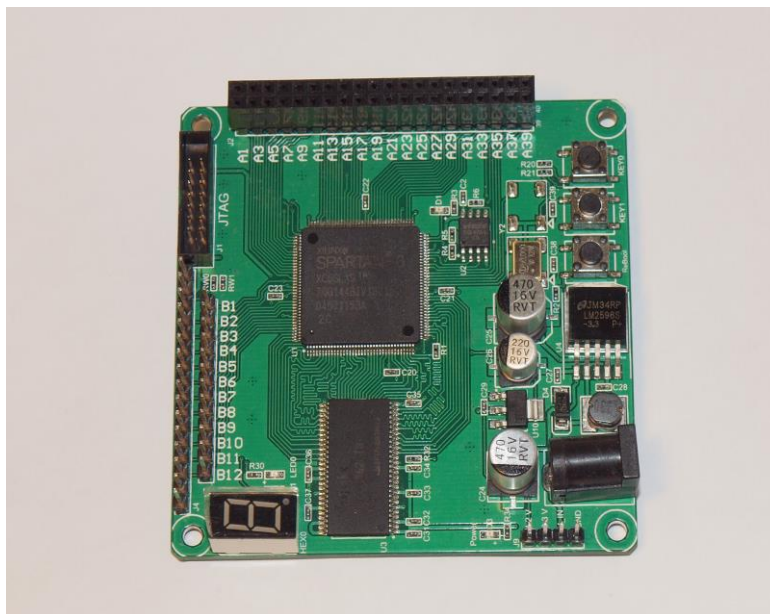
- ovládání zapnutí/vypnutí napájení napájecího napětí vývojového kitu Xilinx Spartan-3A/3AN Starter Kit Board
- 8 spínačů - použitelné buď samostatně pro generování hodnot logické 0 nebo logické 1 nebo jako součást 8 bitového vektoru, v tomto případě lze hodnotu celého vektoru zadat pomocí hexadecimálního čísla
- 4 tlačítka
- generátor periodického signálu (použitelný jako hodinový signál) o volitelné frekvenci 1Hz, 10Hz, 1kHz, 10kHz
- tlačítko ovládající signál -PROG FPGA, tento signál zahajuje rekonfiguraci programovatelného hradlového pole



Obr. 5: Aplikace Virtuální deska FPGA

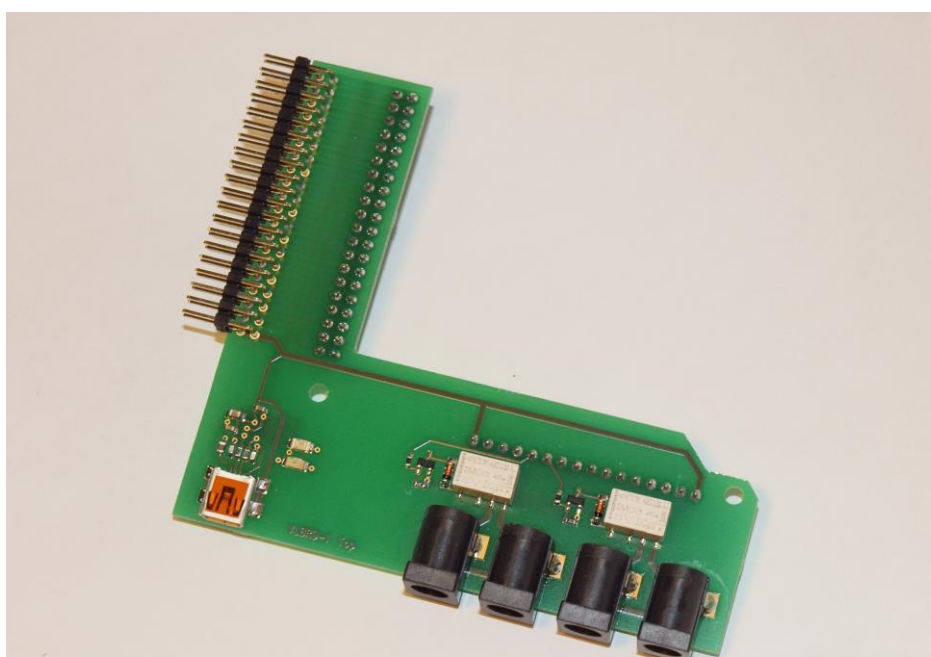
Jako platformu pomocného hradlového pole jsem pro tuto verzi realizace vzdálené laboratoře zvolil univerzální desku "XC6SLX Board", viz obr. 6, vybavenou FPGA z rodiny Xilinx Spartan-6, ke které jsem navrhl a vyrobil interfacovou desku VLBRD-1, viz obr. 7. Schéma zapojení desky VLBRD-1 a motiv tištěného spoje je v příloze A.

Deska VLBRD-1 lze pomocí systémového dvouřadého konektoru přímo připojit k univerzální desce XC6SLX Board, takto vzniklá soustava desek je na obr. 8.

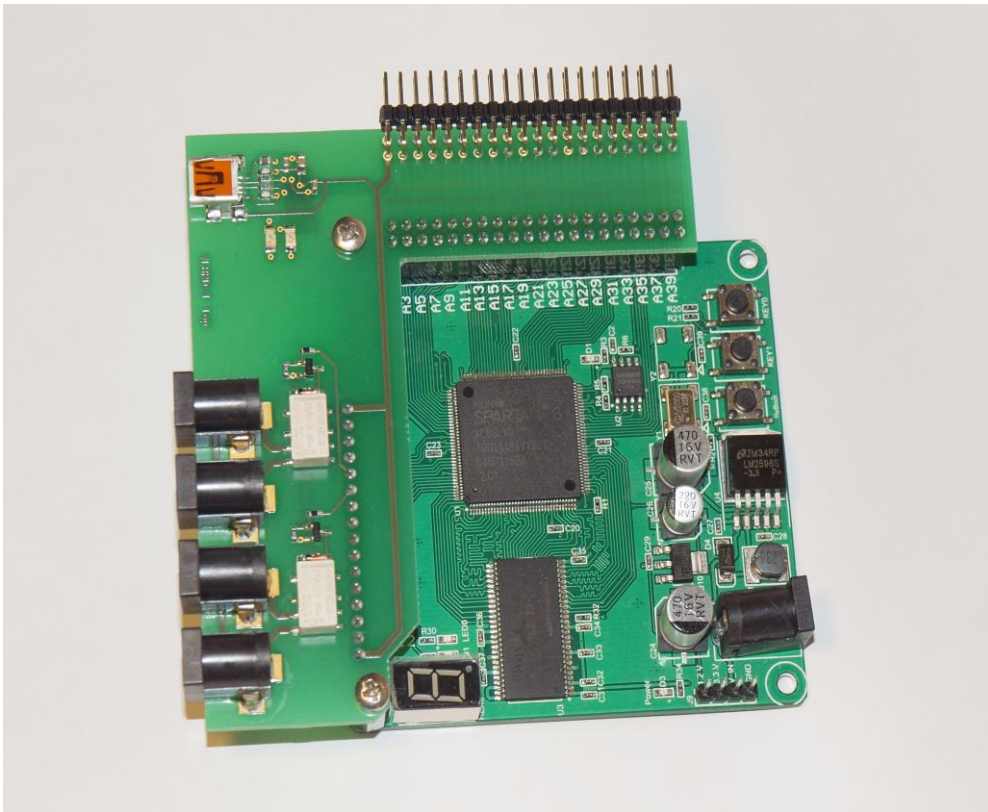


*Obr. 6: Univerzální deska XC6SLX Board*

Deska VLBRD-1 poskytuje sériovými rezistory chráněné signály, které mohou být jak vstupní, tak výstupní. Tyto signály jsou dále připojeny na vývojovou desku Xilinx Spartan-3A/3AN Starter Kit Board a zajišťují ve spolupráci s programem "Virtuální deska FPGA" generování vstupů (tlačítka, spínače, generátor periodických signálů), čtení výstupů (světlo emitující diody, sedmissegmentové zobrazovače). Obsluha napájení desky Xilinx Spartan-3A/3AN Starter Kit Board a obsluha zapnutí osvětlení je realizována pomocí dvou vzájemně a zároveň od zemního potenciálu desky VLBRD-1 galvanicky oddělených kontaktů dvou monostabilních relé.



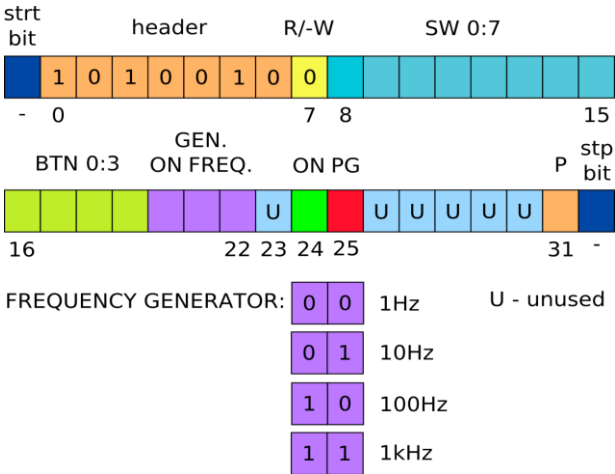
*Obr. 7: Deska VLBRD-1*



Obr. 8: Soustava desek XC6SLX Board a VLBRD-1

Komunikace mezi pomocným FPGA a PC v laboratoři je realizována pomocí desky VLBRD-1, která integruje funkci převodníku portu USB na sériový port typu UART. Komunikace probíhá rychlostí 115kBaud.

Pro zápis dat z aplikace "Virtuální deska FPGA" do pomocného FPGA jsem navrhl jednoduchý protokol, viz obr. 9 a tab. 2.

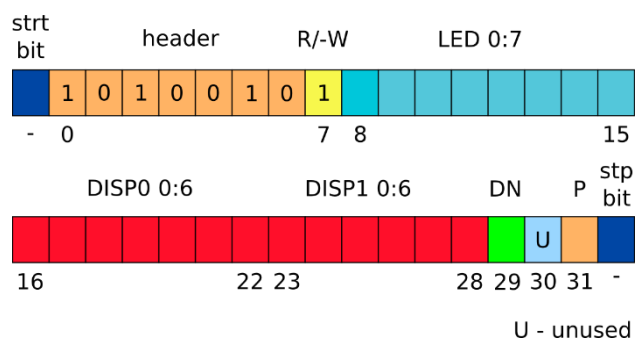


Obr. 9: Protokol vysílání dat

název bitu	popis
strt bit	start bit komunikace na UARTu
header	hlavička komunikace
R/-W	čtení/zápis; 1= čtení / 0= zápis z pohledu PC aplikace
SW	přepínače
BTN	tlačítka
GEN. ON	zapnutí generátoru; 1 = zapnuto
GEN. FREQ.	nastavení frekvence generátoru
ON	zapnutí napájení vývojové desky a osvětlení
PG	signál - PROG (spuštění rekonfigurace)
P	bit sudé parity, počítá se z celého rámce vyjma start bitu, stop bitu a vlastního paritního bitu
stp bit	stop bit komunikace na UARTu
U	nepoužitá pozice, může být 0 nebo 1

Tab. 2: Protokol vysílání dat

Pro čtení dat z pomocného FPGA aplikací aplikace "Virtuální deska FPGA" jsem pak navrhl protokol, viz obr. 10 a tab. 3.

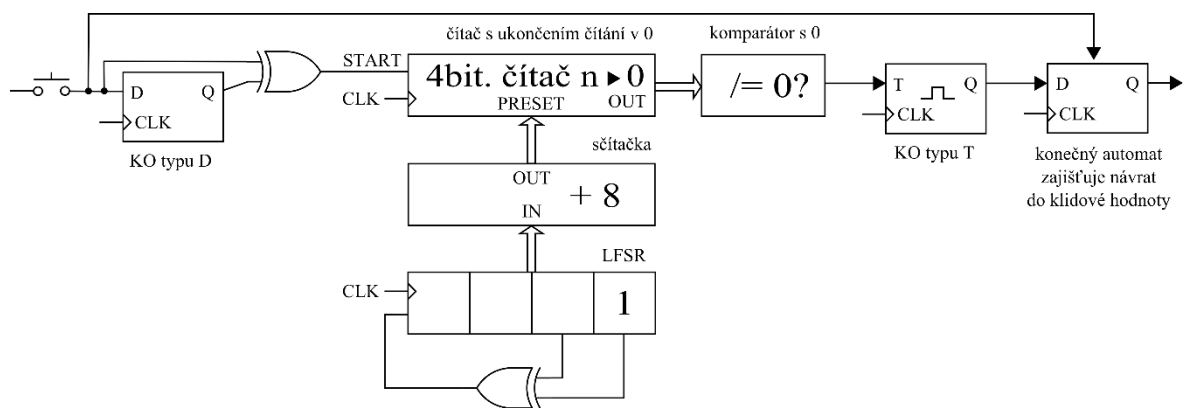


Obr. 10: Protokol příjmu dat

název bitu	popis
strt bit	start bit komunikace na UARTu
header	hlavička komunikace
R/-W	čtení/zápis; 1= čtení / 0= zápis z pohledu PC aplikace
LED	signalizační LED
DISP0	sedmsegmentová jednotka
DISP1	sedmsegmentová jednotka
DN	signál DONE (signalizace úspěšného dokončení konfigurace)
P	bit sudé parity, počítá se z celého rámce vyjma start bitu, stop bitu a vlastního paritního bitu
stp bit	stop bit komunikace na UARTu
U	nepoužitá pozice, může být 0 nebo 1

*Obr. 3: Protokol příjmu dat*

Vzdálené laboratoře často nabízí v jistých ohledech příliš "laboratorní" podmínky. Typickým příkladem může být simulace tlačítka. Reálné tlačítko vykazuje poměrně nepříjemné vlastnosti z hlediska dalšího zpracování signálu, neboť se chová jako mechanický systém a generuje jak při stisku, tak při uvolnění opakovaně měnící se signál - zákmity, které mohou způsobit ovlivnění funkce logických obvodů v případě, že tyto nejsou navrženy s ohledem na vlastnosti tlačítek nebo pokud nejsou v signálové cestě zařazeny obvody určené pro vhodné tvarování signálu (obvody filtrující zákmity). Toto je obzvlášť důležitý aspekt při návrhu programovatelných hradlových polí. Tyto elektronické součástky jsou určeny mimo jiné pro aplikace, kde pracují na frekvencích až stovky MHz a z tohoto důvodu reagují i na velmi rychlé změny signálu, které by například logické obvody klasické řady TTL 7400, používané v mnohých stavebnicích číslicových obvodů, na vstupech nezaregistrovaly. Pokud je funkce tlačítka simulována pomocí obecného generátoru logických signálů, a nebo pomocí I/O karty (zabudované v PC či externí), signál má ideální průběh bez zákmitů a vytvořená úloha se tak vzdaluje od reálné aplikace. Z tohoto důvodu jsem navrhl generátor stisku a uvolnění tlačítka viz obr. 11, který na základě generátoru náhodných čísel vytváří signál simulující zákmity tlačítek, tím jsou simulovány podmínky podobné reálné aplikaci. Generátor náhodných čísel pracuje na principu takzvaného lineárního zpětnovazebního posuvného registru. Blokové schéma generátoru stisku a uvolnění tlačítka je na obr. 11.



Obr. 11: Blokové schéma generátoru stisku/uvolnění tlačítka

Vývojová deska Xilinx Spartan-3A/3AN Starter Kit Board umožňuje připojení k FPGA pomocí systémového konektoru typu Hirose FX2 100. Neboť se nepodařilo sehnat opačný konektor k Hirose FX2 100 pro připojení na plochý kabel, je plochý kabel na straně vývojové desky připájen a na jeho opačném konci opatřen konektorem typu PFL (konektor na plochý kabel s dutinkami).

Výstup generátoru VLBRD-1 je připojen na signál "clk\_aux" vývojové desky pomocí konektoru typu SMA. Signál "clk\_aux" je přiveden k FPGA na vývojové desce na pin hodinového rozvodu a generátor je tak možné použít jako zdroj hodinového signálu pro FPGA Spartan-3A na vývojové desce a doplňuje tak standardní hodinový zdroj signálu na vývojové desce o frekvenci 50MHz.

Připojení jednotlivých periférií na FPGA vývojové desky je uvedeno v příloze B v podobě konfiguračního souboru (.ucf).

### 5.3 Xilinx ISE

Xilinx ISE je Komplexní vývojový softwarový nástroj firmy Xilinx. Základem je soubor parametricky spustitelných programů, které provádí jednotlivé kroky řetězce překladač programovatelných hradlových polí firmy Xilinx, konkrétně:

- syntéza kódu (VHDL, grafický zápis a další), v tomto kroku se snaží návrhový systém porozumět zadanému kódu
- mapování - úloha popsaná zadaným kódem je analyzována a znovu syntetizována za použití technologických možností dané architektury FPGA
- place&route - rozmístění syntetizovaných prvků po ploše konkrétního FPGA (konkrétního zástupce rodiny FPGA)
- časová verifikace - zkontroluje, zda návrh odpovídá zadaným časovým specifikacím
- generování konfiguračních dat

Další důležitou komponentou Xilinx ISE je pak grafické prostředí, které umožňuje psát zdrojový kód ve formě programovacího jazyka či schématu a po případné neúspěšné kontrole zdrojového kódu nebo syntéze označí potenciálně podezřelé řádky ve zdrojovém kódu. Součástí prostředí XilinxISE je také praktický simulátor Xilinx iSim.

Pro realizaci vzdálené laboratoře jsem vybral a nainstalovat verzi Xilinx ISE ve verzi 14.7.

## 5.4 Xilinx Platform Cable II

Platform Cable II, viz obr. 12, je takzvaný JTAG kabel firmy Xilinx Inc. Port JTAG umožňuje v případě FPGA firmy Xilinx Inc. nahrání dat do interní konfigurační paměti typu SRAM, komunikaci s již nakonfigurovaným FPGA (užitečné při ladění aplikace za běhu). Komunikace po portu JTAG je sériová. Přestože existují i JTAG kabely od jiných výrobců, Xilinx Platform Cable II jsem zvolil z důvodu nejlepší kompatibility s návrhovým systémem Xilinx ISE a také díky maximální komunikační frekvenci 24MHz, která je výhodná z důvodu rychlého nahrání konfiguračních dat.



Obr. 12: Xilinx Platform Cable II

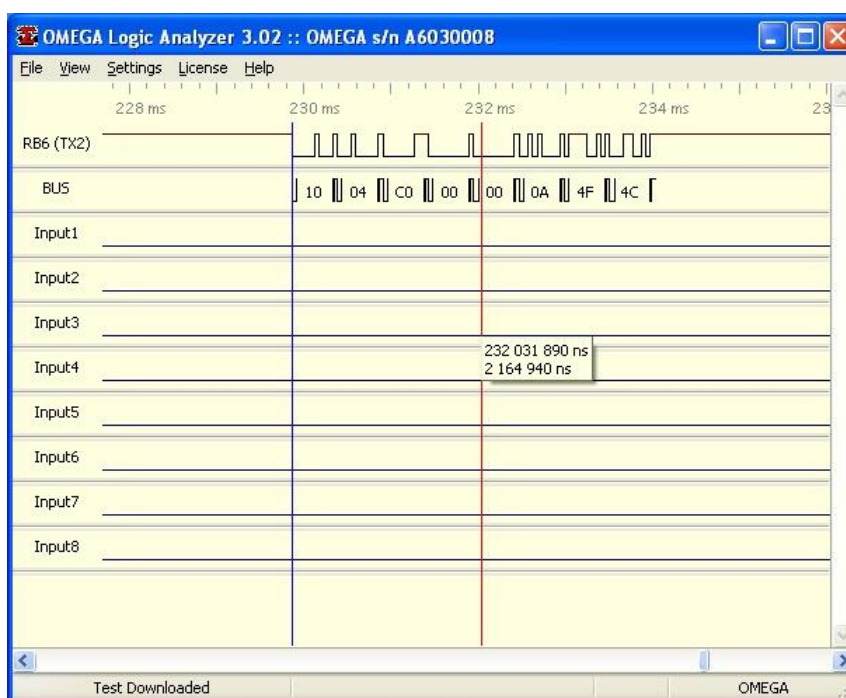
## 5.5 Logický analyzátor OMEGA

Logický analyzátor OMEGA (výrobek firmy ASIX s.r.o.), viz obr. 13, je logický analyzátor, který disponuje až 16 vstupy, vzorkováním až 400MHz. Interní paměť má velikost 512Mbit na ukládání naměřených vzorků, data jsou před uložením navíc komprimována. Komunikace s osobním počítačem probíhá prostřednictvím portu USB 2.0. K logickému analyzátoru dodávaná aplikace (Sigma&Omega Logic Analyzers) nabízí kromě standardní obsluhy logického analyzátoru navíc analyzátory některých komunikačních protokolů (SPI, I2C), pomocí aplikace je též možné na vybraných pinech generovat výstupní signály.



Obr. 13: Logický analyzátor OMEGA

Ukázka prostředí aplikace Sigma&Omega Logic Analyzers je na obr. 14.



Obr. 14: Softwarová aplikace logického analyzátoru OMEGA

Připojení logického analyzátoru / generátoru Omega:

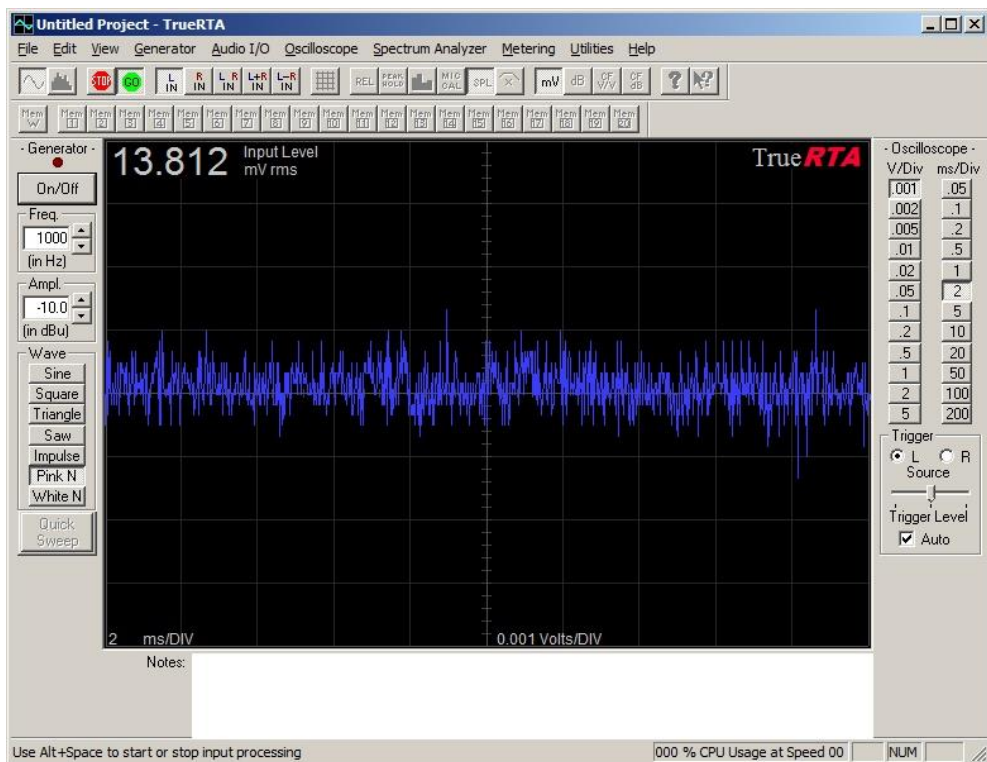
Logický analyzátor OMEGA disponuje 16 logickými vstupy nebo může být nakonfigurován do režimu generátoru 8 logických výstupů. Zvolil jsem proto konfiguraci, kdy je 8 signálů připojeno přímo na volné piny FPGA vývojové desky XC6SLX Board a 8 signálů je připojeno paralelně k signálům LED, které jsou zároveň vedeny na desku VLBRD-1.

## 5.6 Osciloskop

Aplikace TrueRTA nabízí možnost vytvořit s pomocí zvukové karty osobního počítače jednoduchý dvoukanálový osciloskop v rozsahu frekvencí odpovídajících zvukovému signálu a dvoukanálový generátor také v rozsahu frekvencí odpovídajících zvukovému signálu. Ve skutečnosti se, co se týče tohoto frekvenčního rozsahu, jedná o poměrně kvalitní nástroj, neboť A/D (analogově/číslíkové) a D/A (číslíkové/analogové) převodníky na zvukových kartách dosahují dobrých parametrů a v porovnání s klasickými digitálními osciloskopy dokonce větší napěťové rozlišení. Aplikace TrueRTA je dostupná zdarma s některými omezeními, ale i v této omezené verzi je pro použití jako součást virtuální laboratoře plně dostatečná (v našem případě je největším omezením nemožnost zobrazit zároveň oba dva vstupní kanály).

Ukázka prostředí aplikace TrueRTA je na obr. 15.





Obr. 15: Aplikace TrueRTA

## 5.7 Kamera Canyon CNE-CWC2

Kamera CNE-CWC2, viz obr. 16, poskytuje rozlišení 2Mpixely, díky kterému lze na vývojové desce rozeznat například stav signalizačních LED. Kamera je k osobnímu počítači připojena přes port USB.



Obr. 16: Kamera Canyon CNE-CWC2

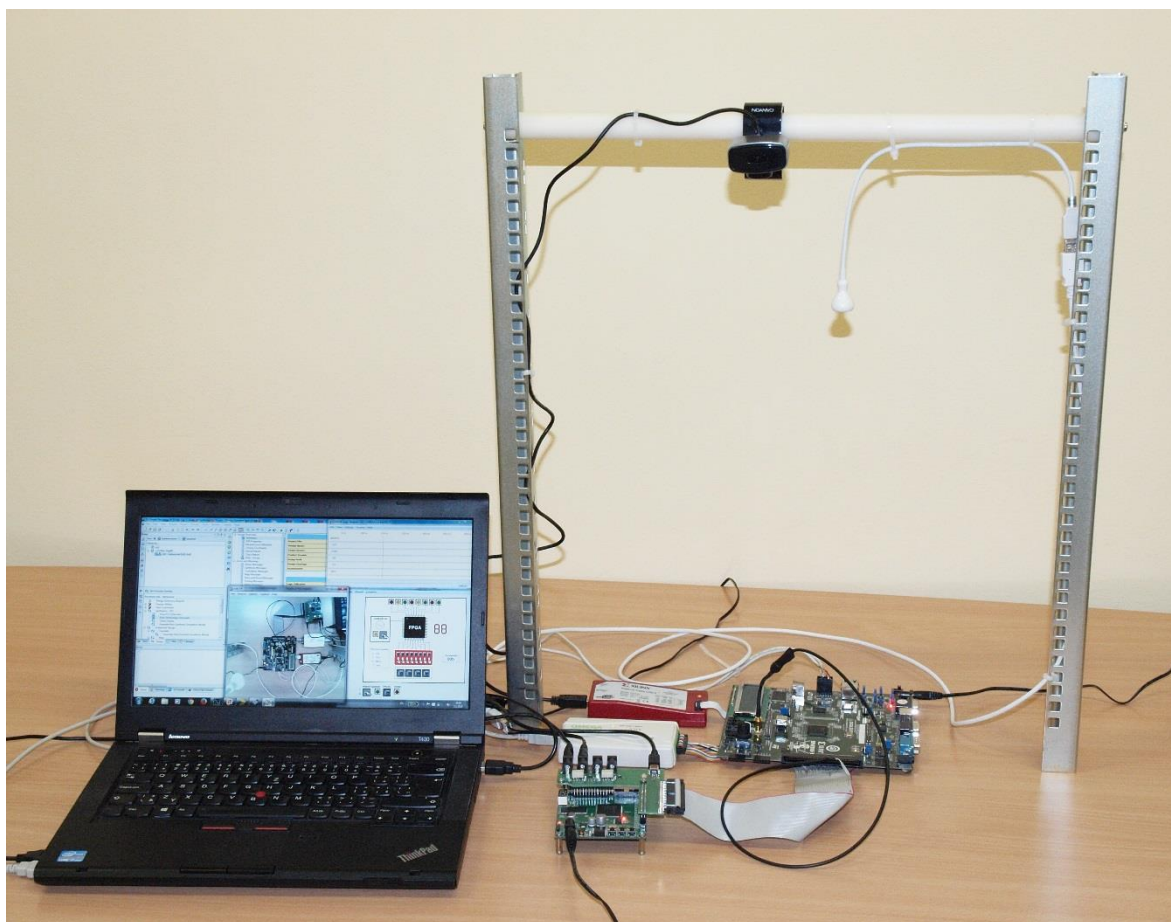
## 5.8 Editor pro psaní kódu

Návrh je možné vytvářet pomocí kódu v programovacím jazyce VHDL (preferovaný) a to buď pomocí editoru v prostředí Xilinx ISE nebo pomocí textového editoru Notepad++, který má předpřipravenou šablonu jazyka VHDL. Návrh je možné vytvářet také blokovým schématem a to za pomoci kreslicího nástroje v programu Xilinx ISE.

Notepad++ je textový editor vhodný pro psaní kódu. Poskytuje integrovanou podporu zvýraznění pro běžnou sadu programovacích jazyků, mezi které patří jazyk VHDL. Dovoluje vytvářet makra. Jednou ze základních vlastností pro efektivní psaní kódu je funkce takzvaného sloupcového bloku, který dovoluje pracovat s blokem znaků organizovaných ve sloupcích. Běžné textové editory dovolují pracovat pouze s bloky znaků organizovaných do řádku, což však z hlediska psaní kódu není optimální, neboť kód programu typicky respektuje specifickou grafickou strukturu (odsazování).

## 5.9 Sestava vzdálené laboratoře

Na obr. 17 je pohled na sestavenou vzdálenou laboratoř. Jako osobní počítač v laboratoři je provizorně použit notebook.



Obr. 17: Vzdálená laboratoř

# Kapitola 6

## Závěr

Hlavním cílem této práce bylo navrhnout vzdálenou laboratoř pro práci s programovatelnými hradlovými poli. Tento cíl se podařilo zcela naplnit, co se týče hardwarové části laboratoře a navržené aplikace pomocí které bylo vytvořeno rozhraní pro přístup k vývojové desce. Systém nicméně zatím vykazuje některé drobné nedostatky z hlediska vzdáleného připojení. Nepodařilo se zajistit konektivitu pomocí sítě internet, k tomuto bude potřeba součinnost partnerů na straně systému Electromeet RemoteLab, s kterými stále probíhá komunikace. Dále pro standardní provoz bude nutné zajistit umístění laboratoře a veřejnou pevnou IP adresu.

V rámci práce jsou uvedeny stávající existující vzdálené laboratoře jak pro obecné experimenty, tak pro práci v oblasti číslicové techniky a FPGA. V práci jsou popsány a rozebrány podstatné rozdíly návrhu úlohy pomocí programovatelných hradlových polí oproti návrhu pomocí klasických logických obvodů. Systém byl pak navržen tak, aby v maximální míře odpovídal návrhu reálné aplikace s FPGA.

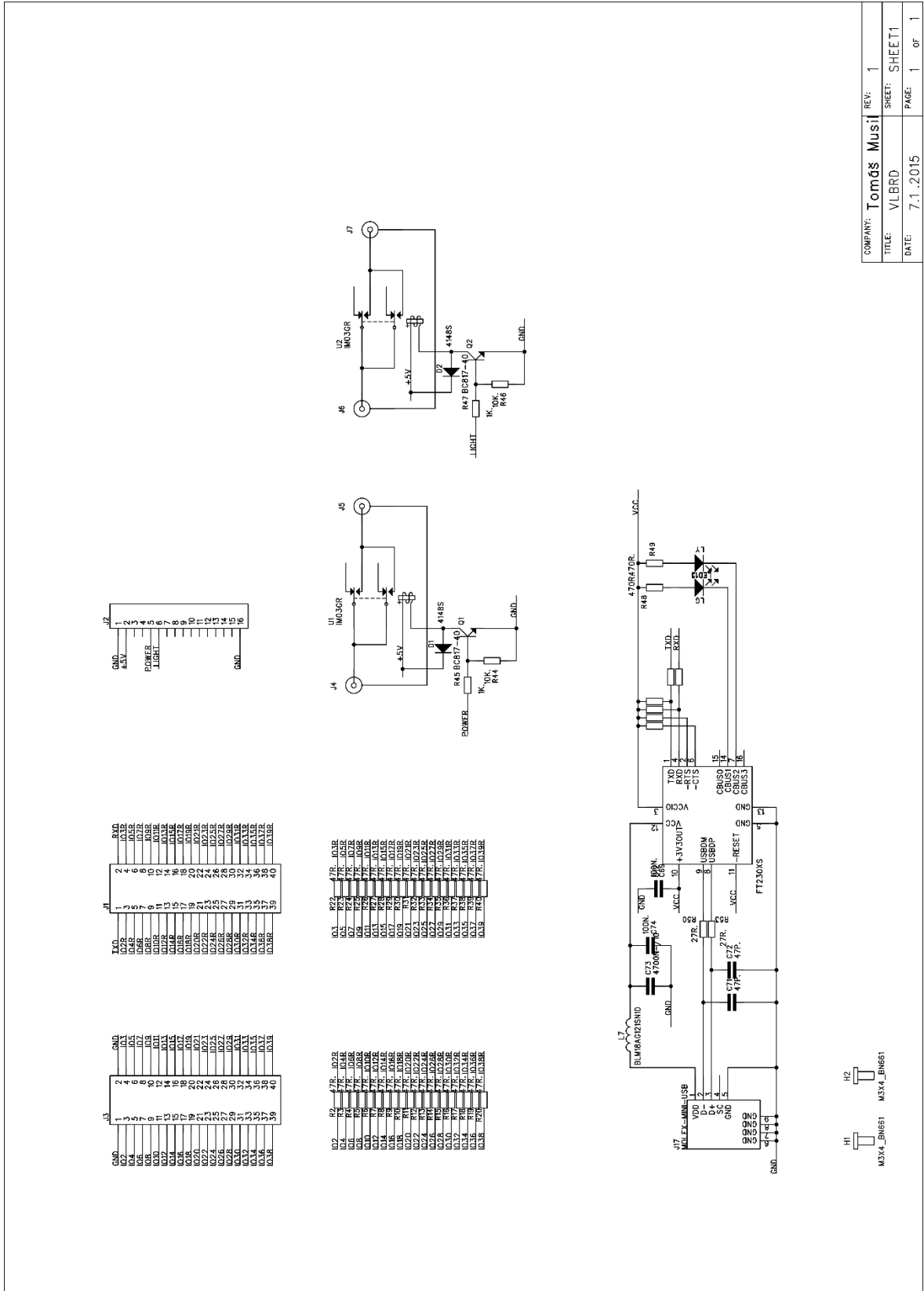
Do budoucna předpokládám v rámci pokračování projektu vzdálené laboratoře tvorbu vlastní webové aplikace, ve které bude integrován java klient VNC a odpadne tak závislost na firemních webových aplikacích. Dalším krokem by mohl být návrh elektronického modulu, který bude obsahovat obě FPGA jak vývojové, tak pomocné, na jedné desce plošných spojů.

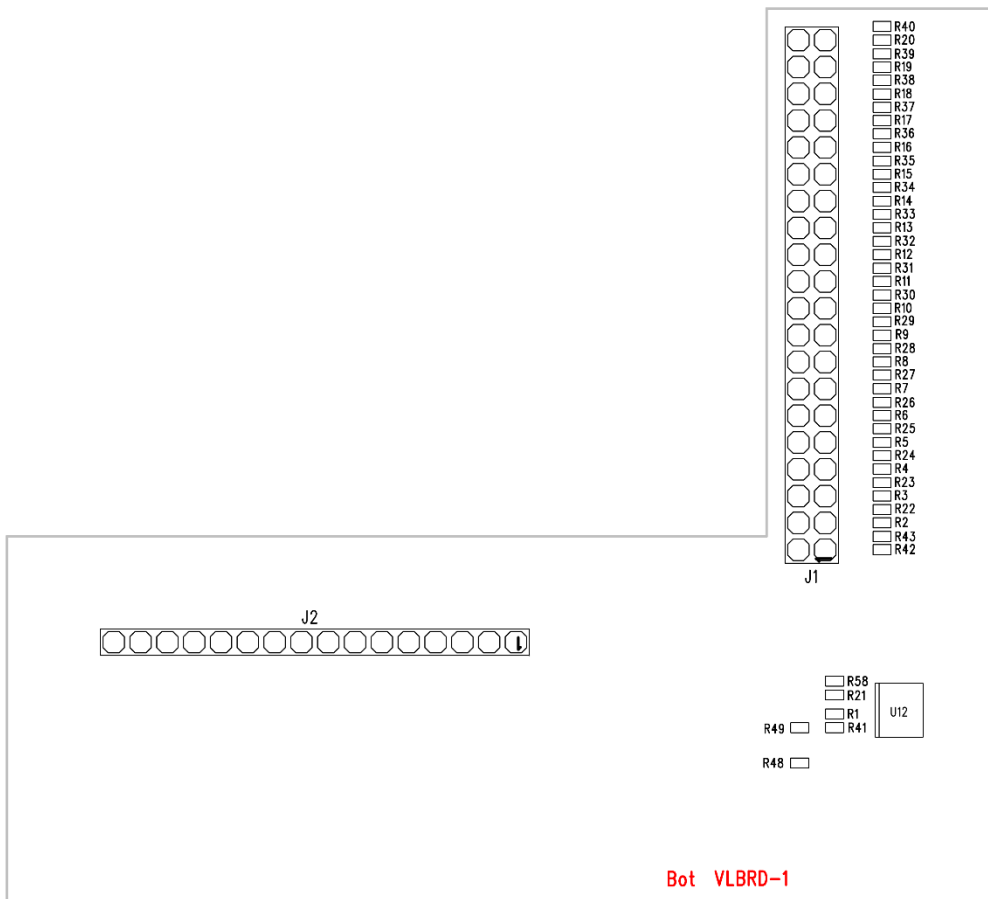
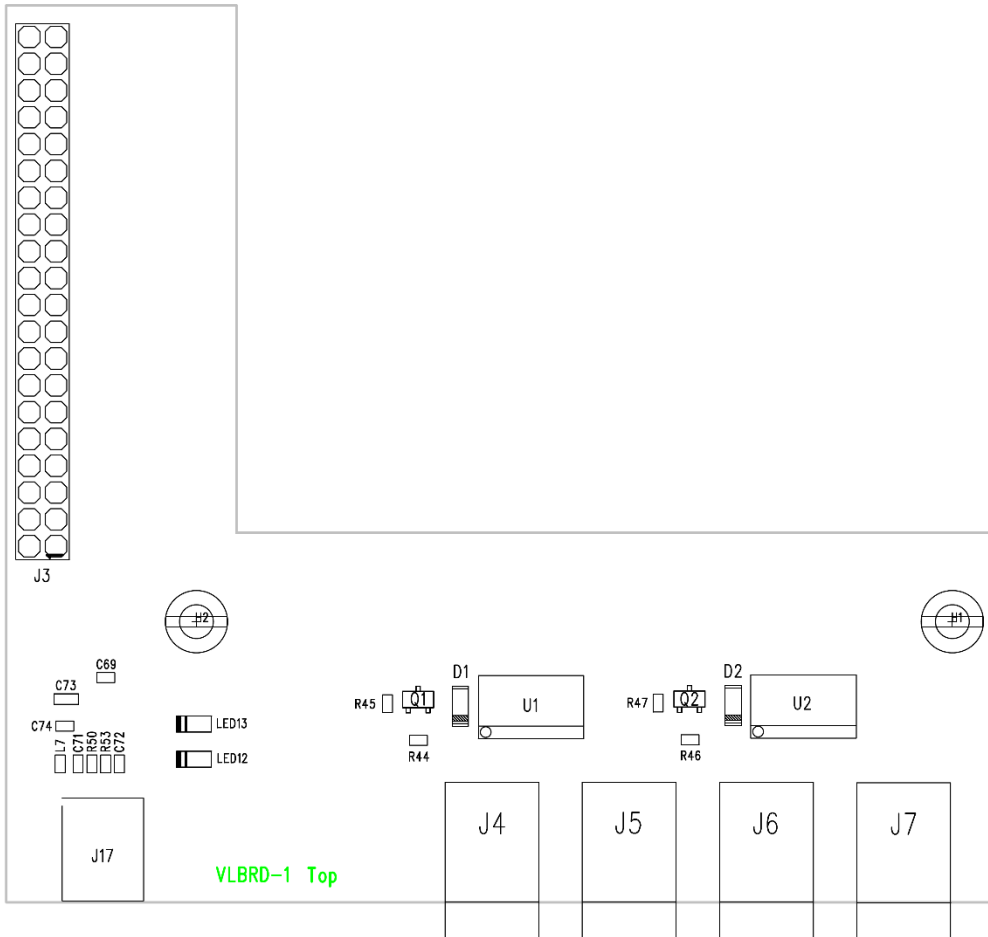
## Seznam použité literatury

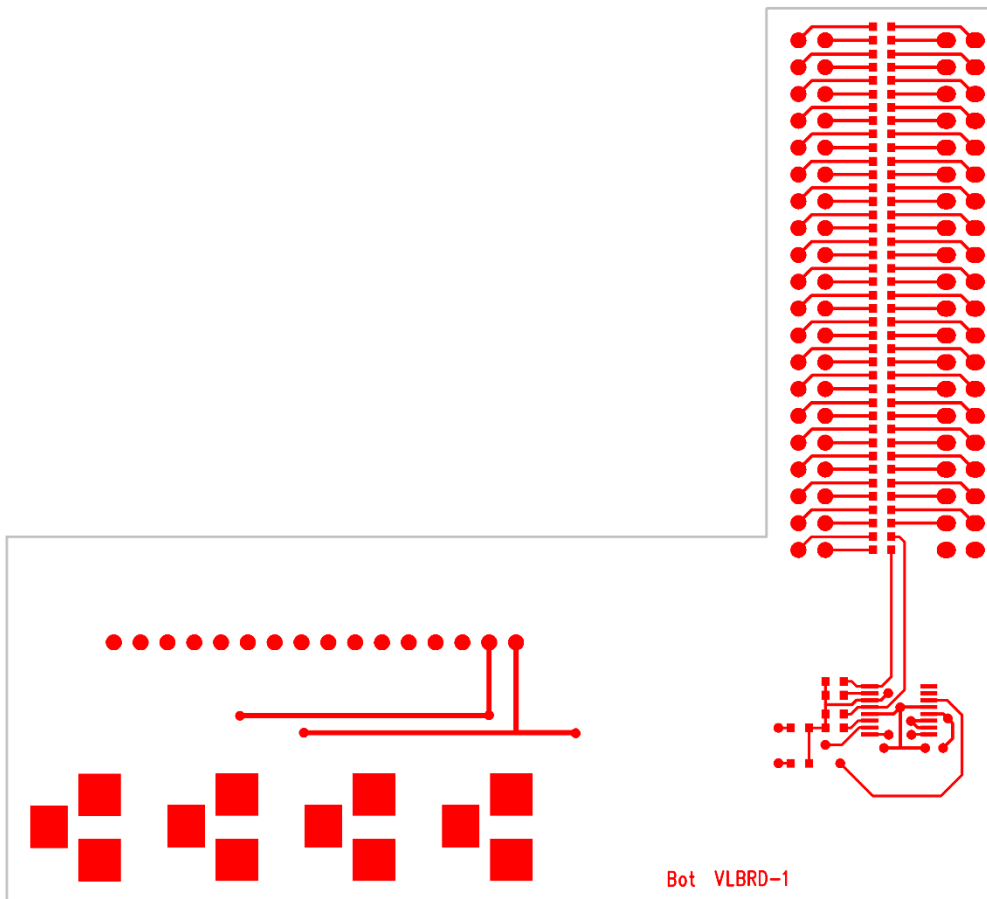
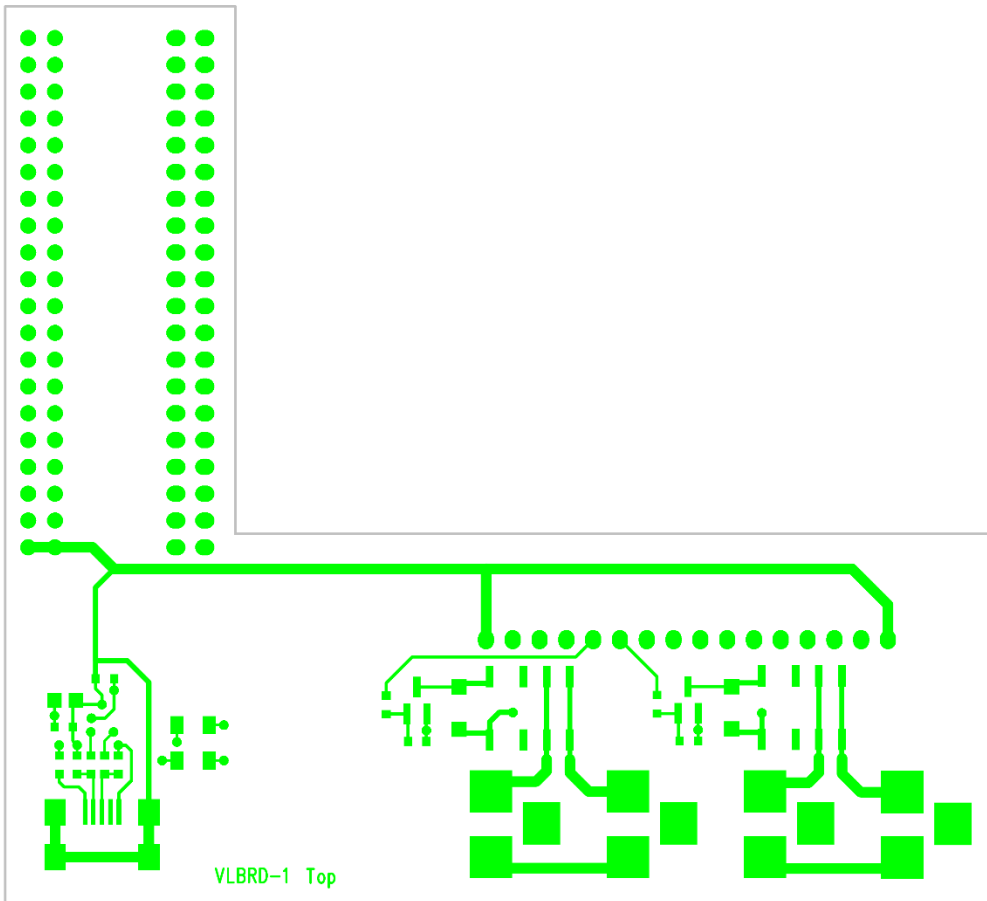
- [1] Vaněček, D. *Elektronické vzdělávání*. Praha: České vysoké učení technické v Praze, 2011. ISBN 9788001049525
- [2] Vaněček, D. *Virtuální a vzdálené laboratoře* [online]. Praha: Masarykův ústav vyšších studií. Dostupné z: <<http://jaroslavbalvin.eu/wp-content/uploads/2014/10/Vanecek.pdf>>
- [3] Látal, F. *Vzdáleně ovládané experimenty ve výuce fyziky* [online]. Disertační práce, Universita Palackého v Olomouci, Přírodovědecká fakulta, 2012. Dostupné z: <<http://theses.cz/id/wpnu70/>>
- [4] Nedic, Z., Machotka, J., Nafalski, A. Remote laboratories versus virtual and real laboratories, *Frontiers in Education*, 2003. FIE 2003 33rd Annual, vol.1, 2003, strana T3E-1-T3E-6. doi: 10.1109/FIE.2003.1263343
- [5] Schauer, F., Lustig, F., Dvořák, J., Ožvoldová, M. An easy-to-build remote laboratory with data transfer using the Internet School Experimental System, *European Journal of Physics*, vol. 29, n. 4, IOP Publishing Ltd, 2008
- [6] Williams, S., Blanchard, R., Mohammed, A., Bliss, M., Pancholi, R., Clowes, M. the development of a remote laboratory for distance learning at loughborough university, *EDULEARN13 Proceedings*, 2013. strana 4342-4349
- [7] Garcia-Zubia, J., Lopez-de-Ipina, D., Orduna, P., Hernandez-Jayo, U., Experience with WebLab-Deusto, *Industrial Electronics, 2006 IEEE International Symposium*, vol.4, 2006, strana 3190-3195. doi: 10.1109/ISIE.2006.296127
- [8] Corter, J.E., Nickerson, J.V., Esche, S.K., Chassapis, C. Remote versus Hands-On Labs, A Comparative Study, *34th Am. Soc. for Eng. Education (ASEE)/IEEE Frontiers in Education Conf.*, 2005
- [9] Orduna, P., Garcia-Zubia, J., Rodriguez-Gil, L., Irurzun, J., Lopez-de-Ipiña, D., Gazzola, F. Using LabVIEW remote panel in remote laboratories: Advantages and disadvantages, *Global Engineering Education Conference (EDUCON)*. 2012. IEEE, 2012, strana 1-7. doi: 10.1109/EDUCON.2012.6201134
- [10] Richardson T., The RFB Protocol Version 3.8, RealVNC Ltd (formerly of Olivetti Research Ltd / AT&T Labs Cambridge), 2010  
dostupné z: <<https://www.realvnc.com/docs/rfbproto.pdf>>
- [11] Tawfik, M., Sancristobal, E., Martin, S., Gil, R., Diaz, G., Colmenar, A., Peire, J., Castro, M., Nilsson, K., Zackrisson, J., Håkansson, L., Gustavsson, I. Virtual Instrument Systems in Reality (VISIR) for Remote Wiring and Measurement of Electronic Circuits on Breadboard. *Learning Technologies*. IEEE Transactions, vol.6, no.1, 2013, strana 60-72. doi: 10.1109/TLT.2012.20
- [12] Dulík, T., Křivka, Z., Kadlec, J., Bližňák, M., Budíková, V., Jiráček, O., Olšarová, N., Trbušek, J., Vašíček, Z. *Virtuální laboratoř pro vývoj aplikací s mikroprocesory a FPGA*. Brno: Akademické nakladatelství CERM sro., 2011. ISBN 978-80-7204-754-3
- [13] Orduña, P., Garcia-Zubia, J., Lopez-de-Ipina, D., Bailey, P.H., Hardison, J.L., DeLong, K., Harward, V.J. Sharing Laboratories across Different Remote Laboratory Systems, *Advanced Learning Technologies (ICALT)*, 2012 IEEE 12th International Conference, 2012, strana 493-494 doi: 10.1109/ICALT.2012.137
- [14] Spartan-3A Family Data Sheet, DS529, Xilinx, 2010  
dostupné z: <[http://www.xilinx.com/support/documentation/data\\_sheets/ds529.pdf](http://www.xilinx.com/support/documentation/data_sheets/ds529.pdf)>

# Přílohy

## Příloha A:







## Příloha B:

```
#-----  
# Timing constraints #  
#-----  
NET "clk50_pin" TNM_NET = clk50_pin;  
TIMESPEC TS_clk50_pin = PERIOD "clk50_pin" 20.0ns HIGH 50%;  
#-----  
# Clk #  
#-----  
NET "clk50_pin" LOC="E12" | IOSTANDARD = LVCMOS33;  
NET "clk_aux_pin" LOC="V12" | IOSTANDARD = LVCMOS33;  
#-----  
# LEDs #  
#-----  
NET "led0_pin" LOC="A13" | IOSTANDARD=LVCMOS33 | SLEW=SLOW |  
DRIVE=8;  
NET "led1_pin" LOC="B13" | IOSTANDARD=LVCMOS33 | SLEW=SLOW |  
DRIVE=8;  
NET "led2_pin" LOC="A14" | IOSTANDARD=LVCMOS33 | SLEW=SLOW |  
DRIVE=8;  
NET "led3_pin" LOC="B15" | IOSTANDARD=LVCMOS33 | SLEW=SLOW |  
DRIVE=8;  
NET "led4_pin" LOC="A15" | IOSTANDARD=LVCMOS33 | SLEW=SLOW |  
DRIVE=8;  
NET "led5_pin" LOC="A16" | IOSTANDARD=LVCMOS33 | SLEW=SLOW |  
DRIVE=8;  
NET "led6_pin" LOC="A17" | IOSTANDARD=LVCMOS33 | SLEW=SLOW |  
DRIVE=8;  
NET "led7_pin" LOC="B17" | IOSTANDARD=LVCMOS33 | SLEW=SLOW |  
DRIVE=8;  
#-----  
# Display #  
#-----  
NET "disp0_a_pin" LOC="A18" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "disp0_b_pin" LOC="C18" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "disp0_c_pin" LOC="A19" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "disp0_d_pin" LOC="B19" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "disp0_e_pin" LOC="A20" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "disp0_f_pin" LOC="B20" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "disp0_g_pin" LOC="C19" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "displ_a_pin" LOC="D19" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "displ_b_pin" LOC="D18" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "displ_c_pin" LOC="E17" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "displ_d_pin" LOC="D20" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "displ_e_pin" LOC="D21" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "displ_f_pin" LOC="D22" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
NET "displ_g_pin" LOC="E22" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;  
#-----  
# Switches #  
#-----  
NET "disp0_a_pin" LOC="F18" | IOSTANDARD=LVCMOS33;  
NET "disp0_b_pin" LOC="F19" | IOSTANDARD=LVCMOS33;  
NET "disp0_c_pin" LOC="F20" | IOSTANDARD=LVCMOS33;  
NET "disp0_d_pin" LOC="E20" | IOSTANDARD=LVCMOS33;  
NET "disp0_e_pin" LOC="G20" | IOSTANDARD=LVCMOS33;  
NET "disp0_f_pin" LOC="G19" | IOSTANDARD=LVCMOS33;  
NET "disp0_g_pin" LOC="H19" | IOSTANDARD=LVCMOS33;
```



```

NET "displ_a_pin" LOC="J18" | IOSTANDARD=LVCMOS33;
#-----
# Buttons #
#-----
NET "disp0_a_pin" LOC="K18" | IOSTANDARD=LVCMOS33;
NET "disp0_b_pin" LOC="K17" | IOSTANDARD=LVCMOS33;
NET "disp0_c_pin" LOC="K19" | IOSTANDARD=LVCMOS33;
NET "disp0_d_pin" LOC="K20" | IOSTANDARD=LVCMOS33;
#-----
# Logic analyser #
#-----
NET "la_io0" LOC="L19" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;
NET "la_io1" LOC="L18" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;
NET "la_io2" LOC="M20" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;
NET "la_io3" LOC="M18" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;
NET "la_io4" LOC="L20" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;
NET "la_io5" LOC="P20" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;
NET "la_io6" LOC="L21" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;
NET "la_io7" LOC="L22" | IOSTANDARD=LVCMOS33 | SLEW=SLOW | DRIVE=8;

```