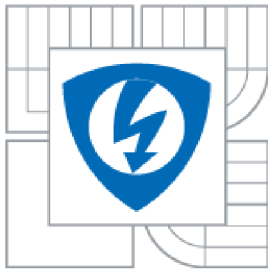




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ
ÚSTAV RADIOELEKTRONIKY**

**FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF RADIOELECTRONICS**

IMPLEMENTACE ALGORITMU PRO MĚŘENÍ PARAMETRŮ ENERGETICKÝCH MATERIÁLŮ V OBVODU FPGA (VEŘEJNÁ VERZE)

**IMPLEMENTATION OF ALGORITHM FOR ENERGETIC MATERIAL MEASUREMENT
IN FPGA DEVICE (PUBLIC VERSION)**

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

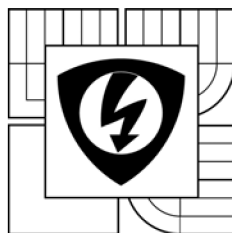
AUTOR PRÁCE
AUTHOR

Bc. JIŘÍ SLOVÁK

VEDOUCÍ PRÁCE
SUPERVISOR

doc. Ing. ROMAN MARŠÁLEK, Ph.D.

BRNO 2014



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav radioelektroniky

Diplomová práce

magisterský navazující studijní obor
Elektronika a sdělovací technika

Student: Bc. Jiří Slovák

Ročník: 2

ID: 125631

Akademický rok: 2013/14

NÁZEV TÉMATU:

Implementace algoritmu pro měření parametrů energetických materiálů v obvodu FPGA

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s principy používaných algoritmů zpracování signálu v zařízení na měření parametrů energetických materiálů. Sestavte vlastní simulaci v prostředí MATLAB, která z měřeného jedno či vícekanálového záznamu vypočítá požadované parametry průběhu exploze (například detonační rychlost). Výsledky zpracujte do přehledné technické zprávy.

V jazyce VHDL popište a v prostředí ISE simulujte Vámi používaný algoritmus zpracování signálu pro zařízení na měření parametrů energetických materiálů. Algoritmus implementujte do dodané desky s FPGA a A/D převodníky. Činnost algoritmu ověřte na sadě záznamů změřených signálů. Výsledky zpracujte do přehledné technické zprávy.

DOPORUČENÁ LITERATURA:

[1] CHAN, E.M. Low-Cost Optoelectronic Devices to Measure Velocity of Detonation, Proceedings of SPIE Vol. 5649, 2005, p. 586-594.

[2] PROAKIS John G. Digital Communications. New York: McGraw-Hill Book, 1995.

Termín zadání: 10.2.2014

Termín odevzdání: 23.5.2014

Vedoucí práce: doc. Ing. Roman Maršálek, Ph.D.

Konzultanti diplomové práce:

doc. Ing. Tomáš Kratochvíl, Ph.D.

předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové 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

V textu diplomové práce je nejprve stručně pojednáno o problematice měření parametrů energetických materiálů v obecné rovině. Důraz je kladen především na popis detonační rychlosti a stručný rozbor nejčastěji používaných metod. Převážná část textu je věnována návrhu a popisu systému, jenž byl vytvořen v prostředí ISE Design Suite s využitím jazyka VHDL. Vývoj byl prováděn s ohledem na pozdější integraci do desky s FPGA a A/D převodníky. Funkčnost algoritmu detekce, implementovaného na základě modelu, byla ověřena v závěrečné části práce simulací zpracování reálných signálů optických sond.

KLÍČOVÁ SLOVA

Energetický materiál, VoD, Optická metoda, MATLAB, VHDL, FPGA

ABSTRACT

In the text of the master's thesis, it is at first briefly referred about Energetic Material Measurement topic in general. Emphasis is placed especially at the description of the Velocity of Detonation and short analysis of selected measurement method. The most significant part of the paper is dedicated to the design and description of the system that was created in ISE Design Suite environment using VHDL language. The development was performed with respect to oncoming integration into the board with FPGA and A/D converters. The operation of detection algorithm which was created based on the MATLAB model was verified in the final part of the thesis by simulation of processing of real optical probe signals.

KEYWORDS

Energetic Material, VoD, Optical Method, MATLAB, VHDL, FPGA

SLOVÁK, J. Implementace algoritmu pro měření parametrů energetických materiálů v obvodu FPGA (Veřejná verze). Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2014. 26 s., 9 s. příloh. Vedoucí diplomové práce doc. Ing. Roman Maršálek, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svoji diplomovou práci na téma Implementace algoritmu pro měření parametrů energetických materiálů v obvodu FPGA jsem vypracoval samostatně pod vedením vedoucího diplomové 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením této práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne

.....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce doc. Ing. Romanu Maršálkovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne

.....

(podpis autora)

OBSAH

Seznam obrázků	VIII
Seznam tabulek	VIII
Úvod	1
1 Detonační rychlost a metody měření	2
1.1 Detonační rychlost (VoD) a další parametry	2
1.2 Metody měření VoD	2
1.2.1 Optická metoda měření VoD	3
2 Návrh algoritmu a model systému	4
2.1 Signály optických sond	4
2.1.1 Charakter signálu a nežádoucí jevy	4
2.1.2 Definice pozice pulzu, okamžik detekce	5
2.2 Vstupní data pro návrh modelu	6
2.2.1 Parametry vstupních dat.....	6
2.2.2 Formát vstupních dat.....	6
2.3 Výpočet detonační rychlosti	6
2.3.1 Výpočet VoD	6
2.3.2 Zobrazení výstupních dat.....	7
3 Implementace systému v obvodu FPGA	8
3.1 Podmínky implementace, použité prostředky.....	8
3.1.1 HW prostředky pro zpracování	8
3.1.2 Použitá zařízení	9
3.1.3 Odhad nároků na paměť pro záznam	10
3.1.4 Vývojové prostředí.....	10
3.1.5 Parametrizace návrhu, soubory konstant	11
3.1.6 Speciální konstrukce jazyka VHDL, Record Types.....	11
3.2 Struktura implementovaného systému	12
3.3 Rozhraní ADC ADS4229.....	13
3.3.1 Režim přenosu, LVDS	13
3.3.2 ADC jako zdroj hodinového signálu	14

3.3.3	Datové rozhraní	15
3.3.4	Konfigurační rozhraní.....	15
3.4	Blok Detector	16
3.4.1	Schéma bloku, vstupy a výstupy.....	16
3.5	Záznam signálu, blok Acquisition.....	17
3.5.1	Schéma bloku, vstupy, výstupy a komponenty	17
3.5.2	FIFO zásobník, Pre-Trigger okno	17
3.6	Použití BRAM IP Core	18
3.6.1	Popis bloku.....	18
3.7	Zpracování signálu, blok Processing	19
3.8	Použití SW mikroprocesoru Picoblaze	19
3.8.1	Popis a vlastnosti	19
3.8.2	Připojení Picoblaze k FPGA, Vstupní a výstupní multiplex	20
4	Vyhodnocení výsledků simulací	21
4.1	Nástroje simulace bloku TOP	21
4.1.1	Testování řídicích signálů	21
4.1.2	Testování funkce algoritmu	22
4.1.3	Waveform soubory	23
	Závěr	24
	Literatura	25
	Seznam zkratk	26
	Seznam příloh	27

SEZNAM OBRÁZKŮ

Obr. 2.1:	Příklad saturace špičky impulzu (Tetryl)	4
Obr. 2.2:	Příklad výskytu parazitních impulzů (Semtex)	5
Obr. 2.3:	Metody detekce pozice pulzu	5
Obr. 3.1:	Obecná struktura FPGA, převzato z [5].....	8
Obr. 3.2:	Jednoduché schéma systému použitého pro implementaci	9
Obr. 3.3:	Příklad komplementárních signálů LVDS s offsetem V_{OCM} , převzato z [8]..	13
Obr. 3.4:	Využití technologie DDR pro přenos vzorků signálu, převzato z [8]	14
Obr. 3.5:	Řetězec konverze diferenčního hodinového signálu	14
Obr. 3.6:	Řetězec konverze datového DDR signálu	15
Obr. 3.7:	Ukázka průběhů signálu SPI rozhraní ADS4229, převzato z [8]	16
Obr. 3.8:	Schematický symbol bloku Detector	16
Obr. 3.9:	Schematický symbol bloku Acquisition	17
Obr. 3.10:	Princip kruhového zásobníku	18
Obr. 3.11:	Schematický symbol BRAM v režimu Simple Dual-Port.....	18
Obr. 3.12:	Schematický symbol bloku Processing.....	19
Obr. 3.13:	Architektura mikroprocesoru Picoblaze, převzato z [11]	20
Obr. 3.14:	Multiplexace vstupů mikroprocesoru Picoblaze [11]	20
Obr. 4.1:	Řetězec simulace zpracování záznamů reálných signálů.....	22

SEZNAM TABULEK

Tab. 1.1:	Přehled metod měření VoD [1], [3]	3
Tab. 2.1:	Parametry záznamů, dostupných z předchozích měření, 2 varianty.....	6
Tab. 3.1:	Vybrané parametry použitého FPGA [7].....	9
Tab. 3.2:	Tabulka paměťových nároků měřeného záznamu	10

ÚVOD

Energetické materiály, tedy materiály, které lze charakterizovat tím, že při jejich chemické přeměně dochází k uvolnění značného množství energie, jsou dnes využívány v mnoha odvětvích, ať už se jedná o pohonné hmoty v dopravních prostředcích, zdroje energie pro domácnosti nebo výbušné směsi, užívané pro vojenské nebo stavební účely. Nezbytným předpokladem pro využití těchto materiálů je schopnost kvantifikovat jejich účinky, čili určit jejich parametry.

Měření parametrů některých podskupin těchto materiálů může být poměrně problematické, avšak o to důležitější. Typickým příkladem je naposled zmiňovaná skupina výbušnin, jejichž analýza bude hlavní funkcí zařízení, pro které byl algoritmus navrhován. V první kapitole diplomové práce je tedy krátce pojednáno o metodách, jež jsou pro jejich analýzu používány a je objasněn význam měřeného parametru, označovaného jako VoD (Velocity of Detonation).

Kapitola druhá se zabývá návrhem algoritmu pro detekci pozic impulzů v signálech optických sond. Nejprve je pojednáno o očekávaném charakteru signálu a možných parazitních jevech, ovlivňujících výsledky měření. Objevuje se také diskuze o významu pojmu pozice impulzu. Základní principy jsou popsány prostřednictvím modelu systému, jenž byl v rámci semestrální práce vytvořen v prostředí MATLAB.

Stěžejní třetí kapitola textu obsahuje stručný rozbor praktické realizace systému v jazyce VHDL na vývojové platformě firmy Xilinx. Kompletní popis je součástí neveřejné verze diplomové práce.

V závěrečné části práce je nastíněno s jakými prostředky a s využitím jakých nástrojů byl navržený systém testován. Součástí příloh je krátké vyhodnocení výsledků získaných pomocí simulace zpracování reálných signálů.

1 DETONAČNÍ RYCHLOST A METODY MĚŘENÍ

Jak bylo naznačeno v úvodu práce, kvantifikace účinků výbušnin, tedy podskupiny energetických materiálů, je poměrně náročný proces, který však má zásadní význam pro jejich použití. Následující kapitola slouží jako krátký úvod do problematiky měření parametrů těchto látek.

1.1 Detonační rychlost (VoD) a další parametry

Vlastnosti výbušnin, tedy látek schopných v extrémně krátkém čase uvolnit velké množství tepelné energie, jsou definovány řadou parametrů. Jmenovitě lze mluvit např. o hustotě výbušniny, detonačním tlaku, citlivosti, voděodolnosti, tepelné stabilitě či detonační rychlosti. Pro diplomovou práci je vhodné, objasnit zejména poslední ze jmenovaných parametrů.

Detonační rychlost, často označovaná zkratkou **VoD (Velocity of Detonation)**, je jedním z nejvýznamnějších parametrů výbušných materiálů. Označuje rychlost, se kterou se při detonaci šíří rázová vlna tělem výbušniny a je ovlivněna chemickým složením výbušniny. Její znalost je jedním z hlavních faktorů při volbě optimálního množství materiálu pro bezpečný a dostatečně efektivní průběh řízené exploze.

Hodnoty VoD se typicky pohybují v řádech tisíců m/s a značně tedy převyšují např. rychlost zvuku ve vakuu. Tento fakt výraznou mírou ovlivňuje používanou metodiku měření, kdy je třeba v extrémně krátkém čase zaznamenat a případně i určitým způsobem zpracovat, velké množství hodnot. Nejčastěji se jedná o intervaly od vyšších desítek mikrosekund až po milisekundy v závislosti na délce a typu výbušniny [1].

1.2 Metody měření VoD

Existuje poměrně široká škála metod, kterými lze detonační rychlost analyzovat, diplomová práce však není jejich studií a následující kapitola tedy obsahuje pouze krátký přehled pro doplnění pozadí samotného návrhu.

Metody měření VoD lze dělit dle různých hledisek. Existují metody, umožňující měřit rychlost spojitě v celé délce analyzovaného sloupce výbušniny, naproti tomu metody určené k měření pouze v určitých bodech např. pomocí optických sond. Odlišným způsobem je třeba přistupovat k měření tohoto parametru v otevřeném prostředí v porovnání s analýzou řízených explozí v prostoru uzavřeného charakteru. Tabulka 1.1 představuje krátký souhrn nejpoužívanějších metod se stručným popisem principu, na němž jsou založeny [1], [2].

Tab. 1.1: Přehled metod měření VoD [1], [3]

Metody	Stručný popis
Fotografické	Světlo, generované při explozi, je snímáno vysokorychlostními kamerami v reálném čase. Z grafu závislosti pozice čela vlny na čase pak, lze jednoduše vypočítat VoD.
Optické	Point to Point metody. S využitím optických vláken schopných přenést vznikající světelný signál jsou zaznamenány okamžiky průchodu rázové vlny body známých pozic. Z časů pulzů a rozdílů vzdáleností sond lze spočítat VoD.
Elektrické	Využívají změn elektrického odporu při postupné detonaci vodiče umístěného do (příp. okolo) těla nálože. Alternativou je použití ionizačních snímačů (snímače jsou se vznikem ionizace postupně vodivě spojeny).
Dautricheho	Je využito toho, že dvě vlny šířící se různou rychlostí urazí za stejnou dobu různou dráhu. Výbušnina je doplněna zápalnicí známé detonační rychlosti. VoD lze dopočítat z rozdílu vzdáleností geometrického středu mezi přípojnými body zápalnice a skutečného místa setkání obou vln.

1.2.1 Optická metoda měření VoD

Optická metoda spočívá ve využití konečného počtu sond zavedených do těla výbušniny. Jednotlivé sondy jsou tvořeny optickými vlákny, schopnými zachytit a přenést světelné záblesky vznikající při explozi. Výstup těchto vláken je pak možno po vhodném optoelektronickém a následně analogově digitálním převodu analyzovat dostatečně rychlým signálovým procesorem.

Okamžik záblesku a následné destrukce vlákna se v průběhu signálu projeví ostrým maximem. Hlavním úkolem procesoru je spolehlivá detekce těchto maxim v jednotlivých kanálech a následný výpočet VoD. Dílčí kroky zpracování signálu ze sond a matematický popis výpočtu VoD jsou podrobně rozebrány v kapitolách, zabývajících se popisem algoritmu.

Z výše uvedeného popisu je zřejmé, že výstupem zařízení, implementujících tuto metodu, bude vektor hodnot detonační rychlosti, odpovídající umístění optovláknových sond [4].

2 NÁVRH ALGORITMU A MODEL SYSTÉMU

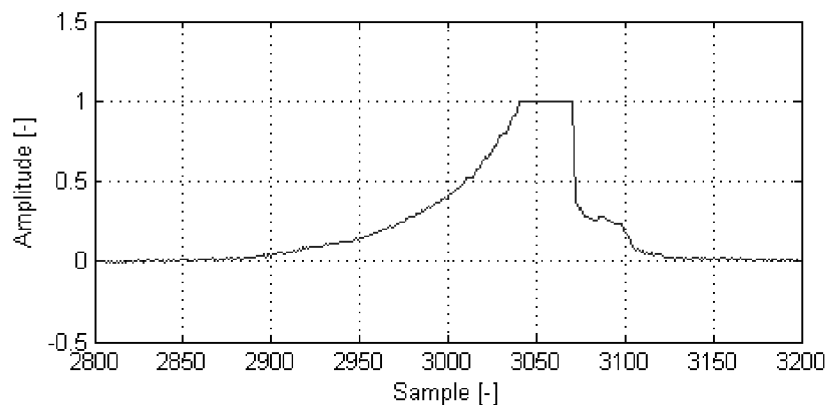
Při implementaci algoritmu do hardware je vhodné vycházet ze systémového modelu, jenž usnadní pozdější návrh a potvrdí jeho funkčnost. Kapitola druhá využívá model, vytvořený v rámci semestrálního projektu v prostředí MATLAB, pro studii výchozích podmínek pro návrh, zejména však pro teoretický rozbor samotného algoritmu. Kroky zpracování signálu jsou podrobně rozebrány v jednotlivých podkapitolách.

2.1 Signály optických sond

Pro návrh samotného číslicového systému je vhodné nejprve zvážit, jaký charakter bude mít vstupní signál, tedy signál, jenž bude systémem zpracováván. V případě použití optických sond lze vyjít z principu popisovaného v předchozí kapitole.

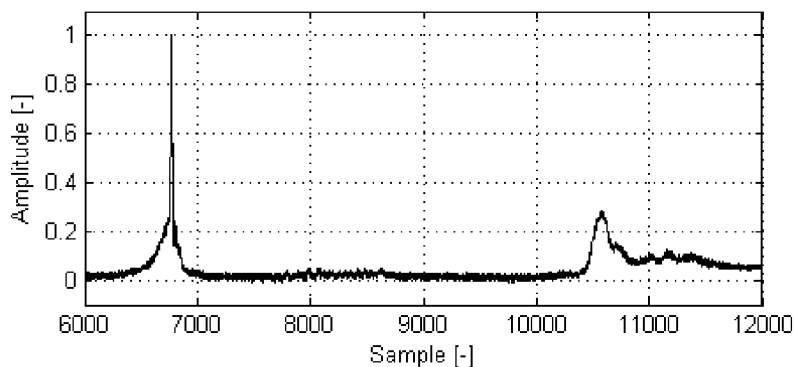
2.1.1 Charakter signálu a nežádoucí jevy

V prvé řadě bude použit vícekanálový záznam. Kanál, který jako první zareaguje na zvýšení úrovně signálu při explozi, bude zdrojem spouštěcího pulzu (Trigger). Vzhledem k použití optických vláken lze očekávat velmi nízké úrovně šumu v klidovém stavu, naproti tomu průchod rázové vlny (tím pádem i vznikajícího světelného signálu) pozicí umístění sondy se projeví relativně úzkým impulzem velmi vysoké amplitudy. Úroveň maxim může dosahovat saturační úrovně použitých A/D převodníků, špička pulzu tedy může být oříznuta na maximální hodnoty jejich rozsahu. Situace je znázorněna na obr. 2.1, jenž zobrazuje impulz v signálu získaném z měření prováděného na výbušnině Tetryl.



Obr. 2.1: Příklad saturace špičky impulzu (Tetryl)

Dalším možnou komplikací, jež se může v signálu vyskytnout, jsou parazitní impulzy, vznikající zejména za špičkami hledanými při destrukci optického vlákna. Průběh na obr. 2.2 znázorňuje poměrně výrazný parazitní impulz (přibližně od vzorku 10000) v signálu, měřeném při explozi výbuštiny Semtex.

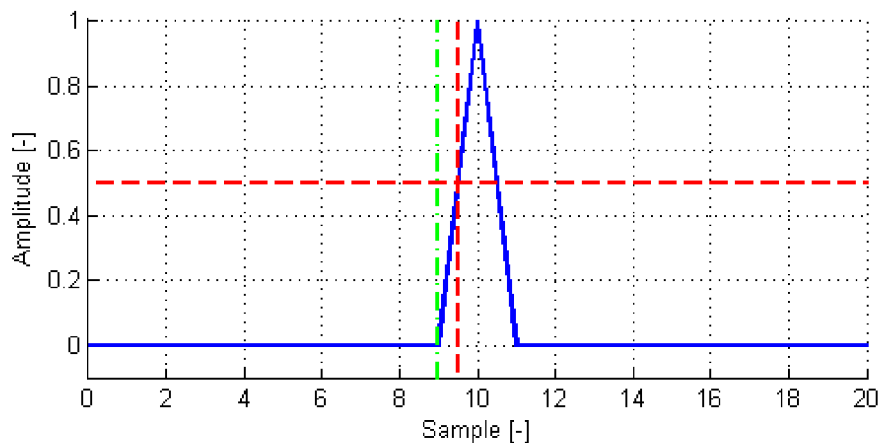


Obr. 2.2: Příklad výskytu parazitních impulzů (Semtex)

2.1.2 Definice pozice pulzu, okamžik detekce

Při zpracovávání signálu je důležité rozhodnout, jak nejlépe definovat pozici impulzu pro správné vyhodnocení naměřených dat. V průběhu návrhu bylo uvažováno několik variant.

Běžnou praxí je např. použití pevně nebo adaptivně vytvořeného prahu, při jehož překročení je zaznamenán příslušný časový okamžik, případně číslo aktuálního vzorku. Nevýhodou metody je nutnost vhodně definovat hodnotu prahu, značná nesourodost tvarů impulzů různých výbušnin nebo např. absence jakékoliv rezistence vůči parazitním impulzům, předcházejícím impulzu hledanému. Na obr. 2.3 je tato metoda naznačena červenými čarami, přičemž vodorovná představuje práh a svislá hledanou hodnotu.



Obr. 2.3: Metody detekce pozice pulzu

2.2 Vstupní data pro návrh modelu

Podkladem pro vytvoření algoritmu byla data, naměřená ve spolupráci s Fakultou chemicko-technologickou Univerzity Pardubice.

2.2.1 Parametry vstupních dat

Záznam dat byl proveden s použitím osciloskopu s parametry, uvedenými v tabulce 2.1, do souborů s příponou *.csv v číselném formátu s pohyblivou řádovou čárkou, neboli **float**. Součástí jsou i přiložené textové soubory, obsahující informace o umístění optických sond, nutné pro výpočet výsledné detonační rychlosti.

Tab. 2.1: Parametry záznamů, dostupných z předchozích měření, 2 varianty

Vzorkovací perioda [ns]	0,8	1,6
Vzorkovací frekvence [GHz]	1,25	0,625
Počet vzorků [-]	125000	62500
Délka záznamu [μ s]	100	100
Počet kanálů [-]	3 + Trigger	2 nebo 3 + Trigger

2.2.2 Formát vstupních dat

Původní varianta algoritmu pracuje se vstupními daty přímo bez jakékoliv konverze, což umožňuje snadnější návrh systému a funkcí jednotlivých bloků. Vzhledem k faktu, že ve výsledné implementaci nebude systém pracovat s formátem typu float, nýbrž s celými čísly (příp. zlomkovými s pevnou řádovou čárkou), byla však navržena také druhá varianta algoritmu. Tato varianta využívá konverze vstupních dat k simulaci problémů, nastávajících při práci s omezeným rozsahem čísel typu **Integer** (celých čísel).

Ke konverzi typů jsou použity vestavěné funkce prostředí MATLAB. Konkrétně jsou použity funkce `int16()` a `int32()`, tedy převod na znaménková celá čísla s délkou 16 nebo 32 bitů. Vyšší počet bitů je nutno použít zejména při násobení, kdy je výsledný rozsah dán součtem délek jednotlivých součinitelů.

2.3 Výpočet detonační rychlosti

2.3.1 Výpočet VoD

Na základě hodnot získaných pomocí vybraného detekčního algoritmu lze poměrně snadno vypočítat výslednou detonační rychlost, přičemž v modelu je tento výpočet realizován funkcí `get_velocity()`.

Pro výpočet je využito znalosti relativní vzdálenosti jednotlivých optických sond a získaných pozic pulzů. Na základě těchto hodnot je možno využít elementární vzorec pro výpočet rychlosti pomocí dráhy s a času t (vztah (2.1)) a upravit jej do potřebné formy, kterou vyjadřuje vztah (2.2).

$$v = \frac{s}{t} \quad (2.1)$$

$$VoD = 0,001 \cdot \frac{\Delta dist}{T_{down} \cdot \Delta pos} \quad (2.2)$$

$$T_{down} = \frac{N}{F_s}, \quad (2.3)$$

kde **$\Delta dist$** představuje vzdálenost sousedních sond v metrech (konstanta 0,001 představuje převod hodnoty načtené z textového souboru v milimetrech na metry), **Δpos** rozdíl pozic impulsů v daných kanálech a **T_{down}** vzorkovací periodu po decimaci vzorkovací frekvence **F_s** faktorem **N** .

Alternativní výsledek lze získat např. záměnou pozice impulsu (Δpos) za jiný příznačný okamžik signálu. V modelu je k výpočtu využito také rozdílu poloh maxim a minim diferenciálu a hodnoty takto získané je tedy možno porovnat.

2.3.2 Zobrazení výstupních dat

Vzhledem k faktu, že model slouží k návrhu a demonstraci funkčnosti algoritmu, je kladen důraz na vizuální zobrazení důležitých průběhů.

O zobrazení důležitých dat, jež jsou předávány jako parametry jednotlivých funkcí se stará funkce `display_values()`. Kromě signálů získaných pomocí příslušné varianty předzpracování je zobrazen také původní průběh analyzovaného signálu.

3 IMPLEMENTACE SYSTÉMU V OBVODU FPGA

Model, jehož část byla popsána v předchozí kapitole, představuje podklad pro implementaci algoritmu detekce do cílového zařízení. V rámci diplomové práce však byl vytvářen kompletní systém, který má několik základních požadavků.

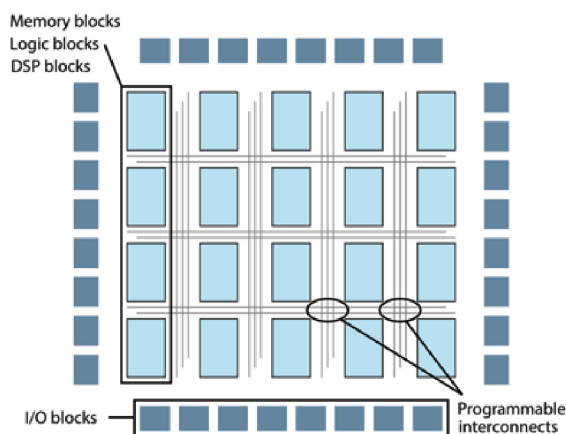
Hlavními z těchto požadavků jsou schopnost detekovat přítomnost impulsu ve vstupním signálu, zaznamenat dostatečné množství dat do vnitřní blokové paměti, detekovat v tomto množství přesné pozice impulsů a v neposlední řadě také umožnit přenos naměřených hodnot do počítače. Implementace tohoto systému je stěžejní částí diplomové práce a její podrobný popis je součástí neveřejné verze.

3.1 Podmínky implementace, použité prostředky

3.1.1 HW prostředky pro zpracování

Jak bylo naznačeno v první kapitole textu, děje, které jsou předmětem analýzy navrhovaného systému, se vyznačují velmi krátkou dobou trvání a je proto nutné použít dostatečně výkonné prostředky pro jejich záznam a zpracování.

Nejčastěji jsou v těchto případech voleny obvody ze skupiny mikroprocesorů, označované jako digitální signálové procesory, tedy DSP (v poslední době také procesory ARM s DSP bloky). Jedná se o instrukčně založené systémy, jež je možno programovat pomocí vyšších programovacích jazyků (např. C) a je tedy možné implementovat poměrně složité algoritmy. Typická je sériová povaha realizace algoritmů (i přes určitou míru paralelizace na úrovni jádra procesoru). Výhodou je zejména relativně nízká cena těchto zařízení.



Obr. 3.1: Obecná struktura FPGA, převzato z [5]

Další nejčastěji volenou skupinou, zejména v náročnějších aplikacích, jsou logická hradlová pole označovaná jako **FPGA (Field Programmable Gate Array)**. Na obr. 3.1 je znázorněna zjednodušená struktura, jejímž základem je pole konfigurovatelných logických bloků (znázorněné světle modrou barvou), obklopené vstupně výstupními piny zařízení. Kromě univerzálních logických bloků se v obvodech objevují také dedikované komponenty (např. DSP bloky, DCM, blokovaná paměť RAM a další).

Požadovaná funkce zařízení je zajištěna spojením různých funkčních bloků pomocí programovatelných propojů (Programmable Interconnects). Díky paralelní povaze obvodů je možno dosahovat výrazně vyšších výpočetních výkonů při značně nižších pracovních kmitočtech. Obvody se vyznačují velmi rychlou reakcí na změnu vstupního signálu v porovnání s výše zmíněnými signálovými procesory. Typickou aplikací je např. zpracování mnoha paralelních toků dat poměrně vysoké frekvence. Paralelismus však v tomto případě znamená také značně náročnější vývoj složitějších algoritmů.

3.1.2 Použitá zařízení

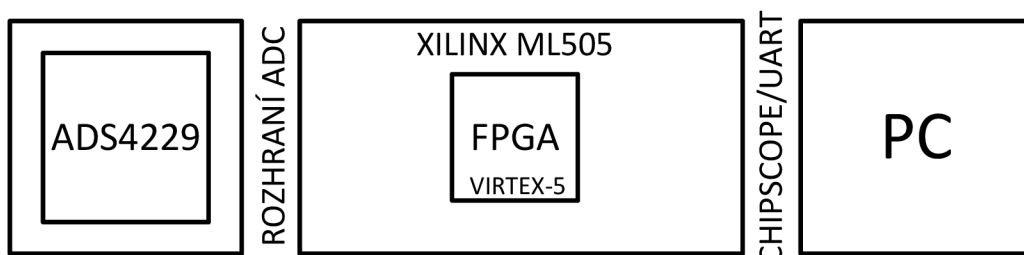
Algoritmus byl navrhován pro vývojovou desku s označením **Xilinx ML505**. Hlavní součástí vývojové desky je obvod FPGA **VIRTEX-5**, osazený ve variantě **XC5VLX50T**, jehož nejdůležitější parametry jsou shrnuty v tabulce 3.1. Deska kromě čipu FPGA nabízí celou řadu užitečných periférií. Jmenovitě se jedná např. o DDR2 paměťové moduly, LCD displej, uživatelsky dostupné přepínače a LED diody či řada vstupně výstupních portů, které mohou posloužit k přenosu naměřených dat do počítače (např. RS-232, Ethernet a další) [6].

Tab. 3.1: Vybrané parametry použitého FPGA [7]

Zařízení	Logické buňky	Slices	Distribuovaná RAM (Kb)	DSP Slices	Bloková RAM (Kb)	Uživatelské I/O
XC5VLX50T	46080	7200	480	48	2160	480

Zdrojem dat pro samotné FPGA je analogově-digitální převodník značky Texas Instruments s označením **ADS4229**. Jedná se o dvoukanálový převodník schopný vzorkovat s frekvencí až **250MSPS** v bitové hloubce až **12 bitů**. Konfigurace probíhá prostřednictvím kombinace sériového rozhraní SPI a paralelních kontrolních pinů a umožňuje nastavení značného množství parametrů a módů. Přestože tvorba rozhraní mezi zmíněným ADC a FPGA není předmětem diplomové práce, je jeho funkčnost stěžejní pro funkčnost celého systému a je o něm tedy krátce pojednáno v jedné z podkapitol zabývajících se popisem jednotlivých komponent [8].

Jednoduché schéma na obr. 3.2 slouží pro lepší představu o zapojení, použitým při vývoji systému.



Obr. 3.2: Jednoduché schéma systému použitého pro implementaci

3.1.3 Odhad nároků na paměť pro záznam

Pro odhad požadavků na paměť je nutno vyjít z několika známých parametrů. V závislosti na zvolené konfiguraci použitého A/D převodníku a jeho nastavení, bude definován vzorkovací kmitočet a počet bitů na vzorek. V projektu byly zvoleny maximální hodnoty, uvedené v předchozí kapitole a vzorkování tedy probíhá na **12 bitových hladin** (počítá se však i s variantou 14 bitů), při frekvenci **250 MSPS**. Předpokládaná délka záznamu je **100 až 200 μs**. Posledním potřebným parametrem je počet kanálů. Během návrhu na desce ML505 bylo počítáno s použitím pouze jednoho převodníku a tedy celkem **dvou** kanálů. Tabulka 3.2 přehledně zobrazuje očekávané nároky na paměť pro různé kombinace těchto parametrů. Pro výpočet byl použit vztah (3.1), u něhož je také uveden příklad výpočtu pro konkrétní hodnoty.

$$X = N_b \cdot M_{ch} \cdot F_s \cdot t_z = 12 \cdot 8 \cdot 250 \cdot 10^6 \cdot 100 \cdot 10^{-6} = \mathbf{2,4\ Mbit}, \quad (3.1)$$

kde N_b značí počet bitů převodníku, M_{ch} počet kanálů, F_s vzorkovací frekvenci a t_z délku záznamu. V tabulce je také uvedeno, jaká část blokové paměti RAM pro danou kombinaci je v konkrétním FPGA využita. Lze pozorovat, že např. při použití osmi kanálů pro záznam, je vnitřní paměť obvodu VIRTEX nedostatečná.

Tab. 3.2: Tabulka paměťových nároků měřeného záznamu

Parametr	Varianta							
	2	2	2	2	4	4	8	8
Počet kanálů [-]	2	2	2	2	4	4	8	8
Počet bitů ADC [-]	12	12	14	14	12	12	12	12
Vzorkovací frekvence [MSPS]	250	250	250	250	250	250	250	250
Délka záznamu [μs]	100	200	100	200	100	200	100	200
Celková paměť [Kb]	600	1200	700	1400	1200	2400	2400	4800
Využití paměti (VIRTEX-5) [%]	27,8	55,6	32,4	64,8	55,6	Více než 100%		

3.1.4 Vývojové prostředí

Značnou výhodou plynoucí z použití obvodu FPGA jsou široké možnosti paralelního programování s využitím jazyka VHDL.

Tradičním vývojovým prostředím pro logické obvody firmy Xilinx je již dlouhou dobu komplexní nástroj nazývaný **ISE Design Suite**. Toto prostředí umožňuje provést kompletní návrh algoritmu, včetně jeho syntézy, produkující logické RTL schéma, až po mapování jednotlivých pinů a flash generovaného bitstream souboru do cílového zařízení, například prostřednictvím rozhraní JTAG. Samozřejmostí je přítomnost simulátoru iSIM, jenž je nezbytným prostředkem pro efektivní návrh a verifikaci funkce algoritmů. Vzhledem k použití čipu VIRTEX-5 bylo toto prostředí zvoleno pro tvorbu diplomové práce ve variantě **WebPack**.

3.1.5 Parametrizace návrhu, soubory konstant

Důležitou vlastností správně navržených digitálních systémů je možnost jejich opakovaného použití (**design reuse**), tedy vlastnost, zajišťující minimální množství potřebných úprav kódu při změně některého z výchozích parametrů. Příkladem může být např. změna bitové hloubky vstupního slova, změna délky záznamu nebo dalších parametrů uvedených v tabulce 3.2.

Stejně jako v jiných programovacích jazycích, je i v jazyce VHDL možno tuto vlastnost zajistit s využitím parametrizace konstantních hodnot. Konkrétním prostředkem, jenž byl využit při vývoji diplomové práce, jsou soubory nazývané *packages*, jež je možno přiřadit na začátek každého zdrojového kódu.

Příkladem balíčku v návrhu je soubor `param_pkg.vhd`, kde se objevují deklarace nejrůznějších konstant. Konstantou je definovaná šířka vstupního slova, šířka adresního prostoru a mnoho dalších hodnot používaných napříč celým návrhem. Při deklaracích odpovídajících signálů je pak místo pevné hodnoty zvolen alias této konstanty.

Díky parametrickému přístupu znamená pro návrh změna většiny vstupních parametrů pouze nutnost upravit hodnoty konstant v balíčku a opětovné vygenerování IP komponent (např. IP blokové paměti RAM), kde není možno definovat tyto hodnoty parametricky.

3.1.6 Speciální konstrukce jazyka VHDL, Record Types

Častým problémem rozsáhlejších digitálních návrhů je velké množství obecně proměnných v systému. V jazyce VHDL hovoříme zejména o vstupních a výstupních portech komponent a vnitřních signálech. Obzvláště při strukturálním popisu znamená přidání či odebrání portu komponenty zdlouhavou úpravu v podobě editace kódu na několika různých místech. Řešením může být použití speciální konstrukce jazyka VHDL označované jako kompozitní typy, v tomto případě konkrétně **Record Types**.

Record types jsou typy podobné strukturám známým z vyšších programovacích jazyků. Umožňují sdružovat skupiny signálů, jak je naznačeno v následující deklaraci typu. Při kombinaci různých typů je však nutno dbát na zachování syntetizovatelnosti kódu.

```
type agu_out_t is
  record
    ADDR          : std_logic_vector (ADDR_WIDTH-1 downto 0);
    ADDR_END      : std_logic;
    DV            : std_logic;
  end record;

constant AGU_OUT_INIT : agu_out_t := ((others => '0'), '0', '0');
```

Zobrazený příklad představuje deklaraci typu pro výstupní porty komponenty adresního generátoru (Address Generation Unit). Z kódu je patrné, že použití Records umožňuje vícenásobné použití běžných názvů signálů, bez nutnosti identifikace

příslušné komponenty (např. signál DV). Rozlišení tedy probíhá na základě názvu celého záznamu. U signálu ADDR lze také pozorovat parametrizaci délky vektoru zmiňovanou v předchozí podkapitole. Současně s deklarací příslušného typu je vhodné vytvořit také inicializační konstantu, kterou lze využít následujícím způsobem při inicializaci odpovídajícího signálu

```
signal agu_tang_out    : agu_out_t := AGU_OUT_INIT;
```

Records také významnou mírou přispívají k zpřehlednění strukturálního kódu při přiřazení vnitřních signálů k portům komponenty. Následující ukázka kódu, představuje instanciaci komponenty adresního generátoru. Protože vstupní port bloku addr_gen s názvem **AGU_IN** představuje stejný datový typ, jako v předchozích příkladech popsany signál **agu_tang_out**, lze je propojit přímo. Pro přiřazení všech signálů, zahrnutých v signálu typu **agu_out_t** tedy stačí použít pouze jeden řádek kódu. Tato vlastnost je výhodná obzvláště při opakovaném použití dané komponenty.

```
ADDR_TANGENT_GEN : addr_gen
  Port map (
    CLK      => CLK,
    INIT_R   => agu_tang_rst,
    AGU_IN  => agu_tang_in,
    AGU_OUT  => agu_tang_out
  );
```

Pro přístup k hodnotám uloženým v signálech uvnitř záznamu lze využít následující syntaxe

```
if (agu_tang_out.DV = '1') then
```

Použití Records znamená značnou úsporu potřebných úkonů při změnách na vstupně výstupních rozhraních vytvářených komponent. V kombinaci s použitím balíčků popsanych v předchozí kapitole znamená např. přidání výstupního portu změnu pouze na jednom místě kódu. V diplomové práci jsou všechny záznamy a další uživatelské typy deklarovány v balíčku `types_pkg.vhd`.

3.2 Struktura implementovaného systému

Systém se skládá z několika základních součástí, mezi které patří bloky rozhraní A/D převodníku, blok Detector včetně jeho součástí (Acquisition, BRAM a Processing) a SW mikroprocesor Picoblaze. Kompletní popis těchto součástí je uveden v neveřejné verzi diplomové práce. V následujících podkapitolách se objevuje stručný přehled vybraných bloků.

3.3 Rozhraní ADC ADS4229

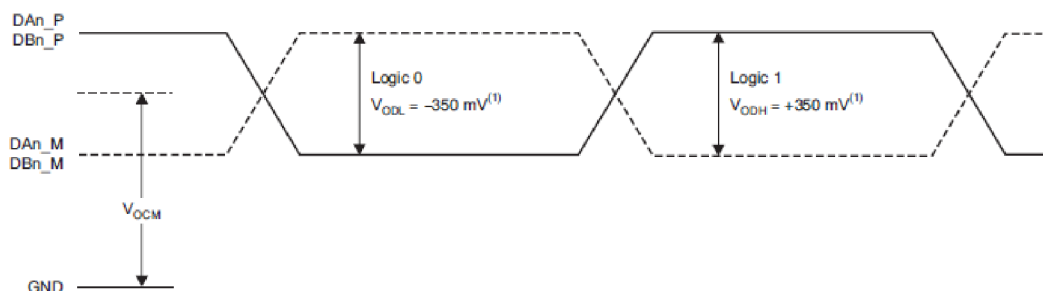
Pro správnou detekci požadovaných hodnot v naměřeném průběhu je zcela zásadní, aby byla přijímaná data interpretována vhodným způsobem. Do role A/D převodníku byl zvolen čip firmy Texas Instruments ADS4229.

3.3.1 Režim přenosu, LVDS

Důležitým krokem při návrhu komunikace mezi převodníkem a zařízením je volba přenosového režimu. ADS4229 umožňuje použít dva režimy přenosu dat.

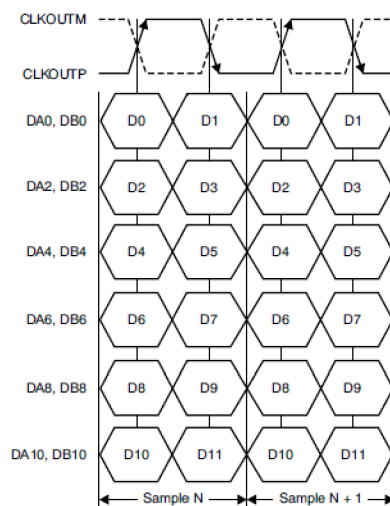
Prvním je režim CMOS, jenž představuje SDR (Single Data Rate) variantu, tedy data jsou vysílána pouze se sestupnou hranou hodinového signálu po celých bitových slovech a každému bitu připadá jeden výstupní pin. Jedná se o Single-Ended standard, výhodný obzvláště díky poměrně jednoduchosti rozhraní. Kvůli nízké odolnosti vůči šumu je však nutno použít poměrně vysoké napěťové úrovně a rychlosti přechodu mezi stavy značně limitují mezní kmitočty. Řešením je použití diferenčních standardů.

LVDS (Low Voltage Differential Signaling) představuje diferenční standard, jenž byl představen v roce 1994 a je dnes používán v rozličných oblastech výpočetní techniky. Princip spočívá v použití dvou komplementárních signálů (obrázek 3.3), přenášených párem vodičů, přičemž hledaná hodnota je dána rozdílem jejich napěťových úrovní. Diferenční povaha přenosu zajišťuje výrazně vyšší odolnost vůči šumu a interferencím. Je tedy možno použít menší rozkmit napětí a tím dosáhnout spolehlivého přenosu i na vyšších kmitočtech (řádově až jednotky GHz). Vzhledem k faktu, že realizovaný systém pracuje s vzorkovacím kmitočtem až 250MSPS, byla zvolena právě varianta LVDS.



Obr. 3.3: Příklad komplementárních signálů LVDS s offsetem V_{OCM} , převzato z [8]

Z principu LVDS je zřejmé, že použití diferenčních párů znamená dvojnásobné nároky na počet pinů pro přenos. V případě převodníku ADS4229 je situace řešena pomocí technologie **DDR** (Dual Data Rate), která umožňuje přenášet užitečná data dvojnásobnou rychlostí, za použití vzestupné i sestupné hrany hodinového signálu. Na obrázku 3.4 lze pozorovat, že pro přenos kompletního vzorku stačí opět pouze 12 pinů, tentokrát však využitých jako 6 párů a s příslušnou hranou hodinového signálu dojde k přenosu sudých nebo lichých bitů slova [8].



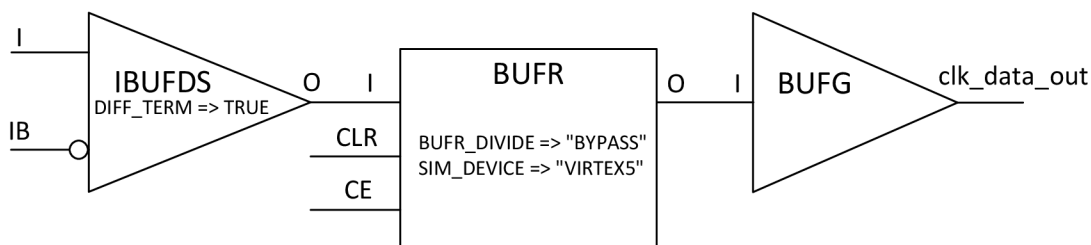
Obr. 3.4: Využití technologie DDR pro přenos vzorků signálu, převzato z [8]

3.3.2 ADC jako zdroj hodinového signálu

Systém, jenž je v rámci práce vytvářen, lze zařadit do skupiny digitálních systémů, označovaných jako **Source Synchronous Systems**. Z toho vyplývá, že kromě datových signálů je převodník také zdrojem diferenčního hodinového signálu.

Vzhledem k použití režimu LVDS je hodinový signál z převodníku přenášen pomocí diferenčního páru, v návrhu označeného jako **CLK_ADC_P** a **CLK_ADC_N**.

Obr. 3.5 zobrazuje řetězec, který slouží pro převod diferenčního hodinového signálu a jeho vhodnou úpravu pro použití v FPGA.



Obr. 3.5: Řetězec konverze diferenčního hodinového signálu

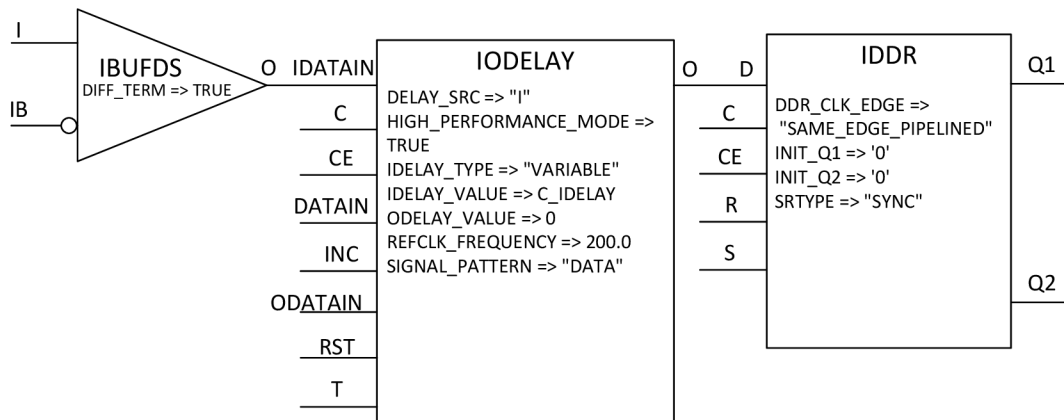
Prvním blokem je buffer **IBUFDS**, umožňující převod LVDS signálu na vstupech **I** a komplementárním **IB** na single-ended signál. Generická hodnota `DIFF_TERM` je použita pro nastavení varianty zakončení diferenčního páru vodičů.

Po konverzi je signál přiveden do regionálního bufferu pro hodinový signál (**BUFR**), protože diferenční hodinový signál převodníku je připojen na piny FPGA, označené jako **CC** (Clock Capable). Tyto piny neumožňují přímé připojení do bufferu pro globální rozvod hodinového signálu. Generická hodnota `SIM_DEVICE` definuje typ použitého hradlového pole. Výsledný signál není třeba dělit, je tedy proveden bypass dělicí logiky (nastavení generické hodnoty `BUFR_DIVIDE`) [9].

Výstup bloku **BUFG** je tedy výsledným hodinovým signálem o frekvenci 250MHz, jenž je v kódu označen jako **clk_data_out**.

3.3.3 Datové rozhraní

Při zpracování datových signálů je situace podobná jako v případě hodinového signálu. Situaci znázorňuje blokové schéma na obr. 3.6.



Obr. 3.6: Řetězec konverze datového DDR signálu

Prvním krokem je opět převod diferenčního signálu na single-ended pomocí bloku **IBUFDS**. Kvůli zajištění ideálního okamžiku vzorkování signálu je na výstup zařazen blok **IDELAY**, jenž představuje konfigurovatelné zpoždění signálu. Blok má opět řadu generických parametrů.

Pomocí parametru **DELAY_SRC** lze zvolit, který z portů bude použit jako vstup bloku (na obr. 3.6 je to port **IDATAIN**). **HIGH_PERFORMANCE_MODE** umožňuje snížit jitter (nežádoucí odchylka od periodicity) za cenu mírně vyšší spotřeby. **IDELAY_TYPE** slouží k rozlišení, jestli je zpoždění fixní nebo variabilní. Dvojice **IDELAY_VALUE** a **ODELAY_VALUE** slouží k definici hodnoty zpoždění (nastavitelné v krocích od 0 po 63). Zbývající parametry představují referenční kmitočet pro blok (z externího zdroje) a informaci o tom, zda se jedná o datový nebo hodinový signál.

Posledním krokem k převodu DDR signálu do SDR podoby je použití bloku **IDDR**. Režim **SAME_EDGE_PIPELINED** pomocí přídatného registru zajišťuje, že na výstupech **Q1** a **Q2** jsou k dispozici příslušné páry sudých a lichých bitů (viz. obr. 3.4) v okamžiku stejné aktivní hrany [9].

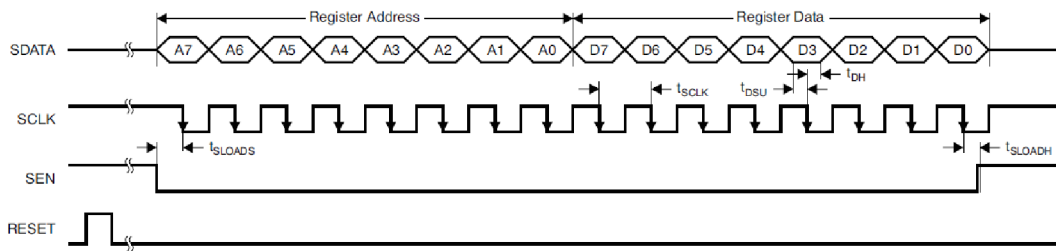
Výsledný signál představuje souvislý tok vzorků o délce 12 bitů, přenášený s frekvencí 250MHz pro každý kanál.

3.3.4 Konfigurační rozhraní

Pro konfiguraci převodníku do požadovaného režimu byla výrobcem implementována dvě základní rozhraní, paralelní a sériové.

Paralelní rozhraní představují piny, v katalogovém listu označené jako **CTRL1** až **CTRL3**, v kombinaci s definovaným napětím na pinech **SEN** a **SCLK**. Různými konfiguracemi úrovní lze prostřednictvím tohoto rozhraní nastavovat např. režim přenosu, číselný formát výstupu nebo režim napájení. Pro aktivaci módu paralelní konfigurace je třeba držet pin **RESET** ve vysoké hodnotě.

Plnou kontrolu nad parametry zařízení však nabízí sériové rozhraní SPI, reprezentované piny SCLK, SEN (aktivní v nízké úrovni), SDATA, SDOUT a značným množstvím vnitřních registrů. SCLK představuje hodinový kmitočet, který je generován zdrojem konfiguračních dat (v tomto případě FPGA) a jeho hodnota může nabývat hodnot od nízkých jednotek Hz až k frekvenci 20 MHz. Obrázek 3.7 znázorňuje průběhy signálů rozhraní při odeslání jednoho rámce. Nezobrazený port SDOUT slouží ke čtení obsahů registrů po aktivaci módu READOUT [8].



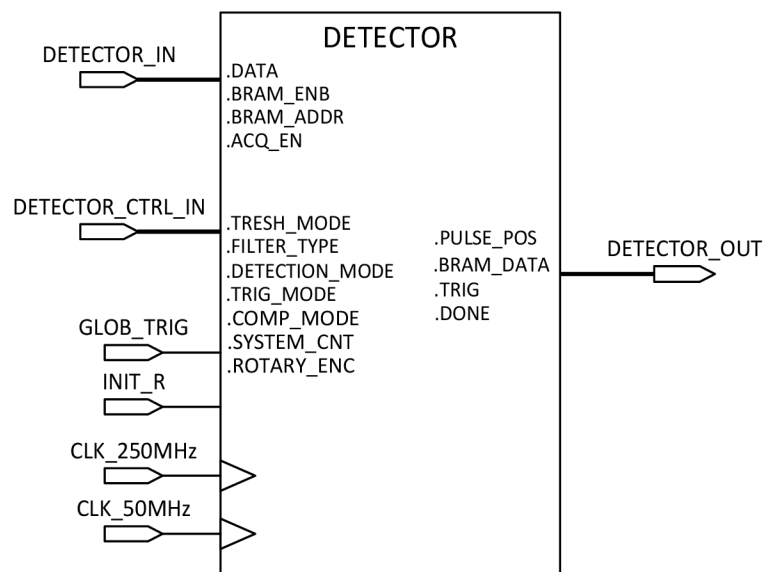
Obr. 3.7: Ukázka průběhů signálu SPI rozhraní ADS4229, převzato z [8]

Rozhraní SPI je v projektu implementováno jako jednoduchý stavový automat, řízený mikroprocesorem Picoblaze. Generace dostatečně pomalého signálu je zajištěna pomocí v určitých intervalech generovaných Clock Enable signálů.

3.4 Blok Detector

Každému kanálu v návrhu přísluší jeden blok Detector. Jedná se o nejdůležitější blok z hlediska záznamu a zpracování signálu.

3.4.1 Schéma bloku, vstupy a výstupy



Obr. 3.8: Schematický symbol bloku Detector

Schematický symbol na obr. 3.8 znázorňuje vstupní a výstupní porty bloku Detector. Tlustá černá čára označuje porty kompozitního typu Records, popsáno v podkapitole 3.1.6. Názvy s tečkovou konvencí představují porty, jež jsou v jednotlivých Records zahrnuty.

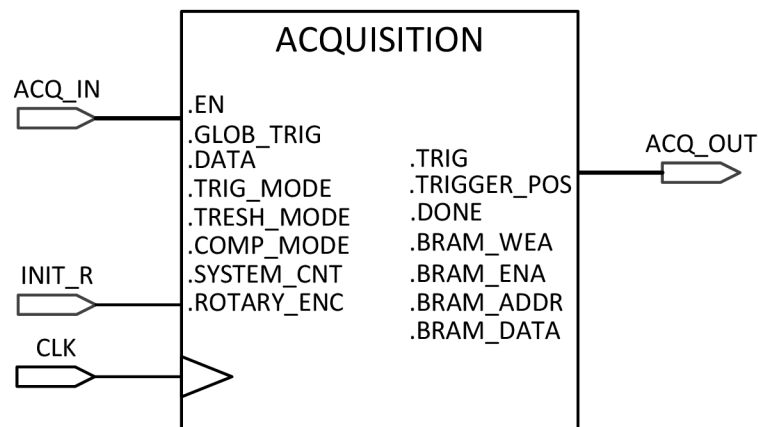
Chod celého bloku je zajištěn pomocí dvojice hodinových signálů CLK_250MHz a CLK_50MHz, přičemž jak naznačuje schematická značka, aktivní hranou je hrana vzestupná.

Přestože reset zařízení bude prováděn na úrovni celého obvodu FPGA, je součástí designu také port **INIT_R**, umožňující inicializaci zařízení na defaultní hodnoty a opětovné zahájení procesu detekce.

3.5 Záznam signálu, blok Acquisition

Blok Acquisition slouží k detekci impulsu a záznamu požadovaného počtu vzorků. Díky vysokému taktovacímu kmitočtu (250MHz) se jedná o nejnáročnější část systému z hlediska časování.

3.5.1 Schéma bloku, vstupy, výstupy a komponenty

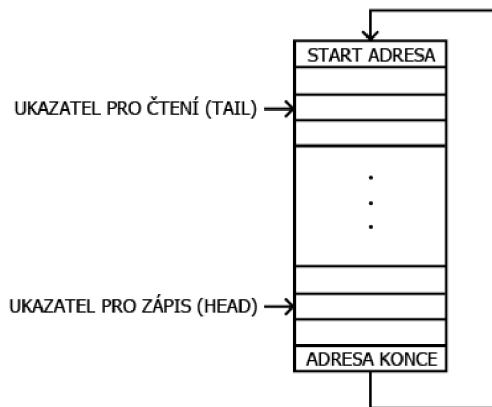


Obr. 3.9: Schematický symbol bloku Acquisition

Ze schematického symbolu na obr. 3.9 je zřejmé, že do bloku vstupuje řada řídicích signálů pro nastavení parametrů měření. Hodinový signál CLK s aktivní vzestupnou hranou je propojen s portem CLK_250MHz bloku Detector

3.5.2 FIFO zásobník, Pre-Trigger okno

Při záznamu signálů optických sond je vhodné, aby byly hodnoty zaznamenány s určitým předstihem a k dispozici tedy byl i signál před pozicí triggeru. V návrhu je tato funkce zajištěna pomocí jednoduchého synchronního FIFO (First-In, First-Out) zásobníku, jenž je implementován jako kruhový buffer, znázorněný na obr. 3.10. Základní princip spočívá v kruhovém adresování určitého paměťového bloku. V případě, že ukazatel pro čtení nebo pro zápis dosáhne poslední vyhrazené adresy, vrací se na nejnižší pozici.



Obr. 3.10: Princip kruhového zásobníku

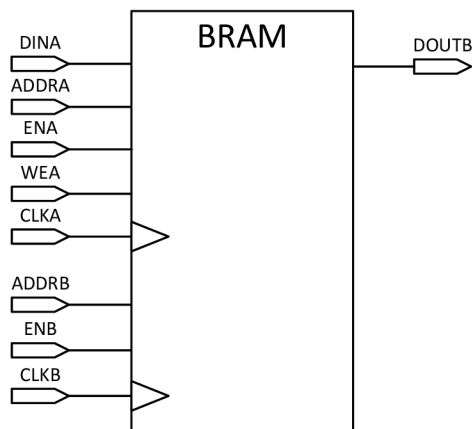
3.6 Použití BRAM IP Core

Součástí každého bloku Detector je paměť BRAM předem zvolené velikosti, určené na základě požadované délky záznamu. Tuto paměť tvoří dedikované bloky, jejichž popisem se zabývá následující podkapitola.

3.6.1 Popis bloku

Základními paměťovými prvky čipu VIRTEX-5 jsou bloky velikosti 36kb (RAMB36). Těchto bloků je v zařízení k dispozici 60, což odpovídá hodnotě 2160kbit paměti, uvedené v tab. 3.1. Bloky RAMB36 je možno dále dělit do dvou nezávislých bloků velikosti 18kb. Paměť obsahuje dva plnohodnotné nezávislé porty a na základě jejich konfigurace ji lze provozovat ve třech základních režimech.

Režimy **Single-Port** a **True Dual-Port** využívají příslušné porty pro zápis i čtení obsahu paměti, přičemž mezi operacemi je přepínáno pomocí vyhrazených signálů. Při použití těchto režimů je třeba dbát na zamezení možnosti vzniku kolize dat (např. pokus o zápis obou portů na stejnou adresu) a definici chování výstupního portu v průběhu zápisu.



Obr. 3.11: Schematický symbol BRAM v režimu Simple Dual-Port

Při implementaci systému, však bylo zvoleno použití režimu **Simple Dual-Port**. Tento režim využívá jeden z portů výhradně pro zápis a druhý pro čtení zaznamenaných dat. Na obr. 3.11 je zobrazen symbol bloku BRAM, jenž byl generován jako IP (Intellectual Property) Core firmy Xilinx s názvem **Block RAM Generator** [10].

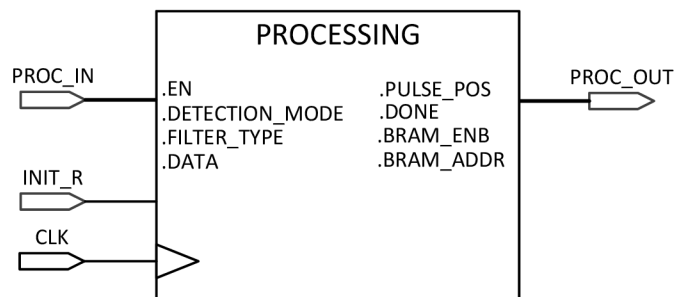
Z popisu vstupních a výstupních portů vyplývá, že port A je určen pro zápis. Zdrojem vstupních dat je port **DINA**. K aktivaci portu dochází při nastavení vysoké úrovně na portu **ENA**. Záznam na adresu, definovanou pomocí hodnoty na portu **ADDRA**, je však zahájen až při aktivaci zápisu, nastavením portu **WEA** (Write Enable A).

V případě portu B je situace podobná. Port je však určen pouze pro čtení dat a není tedy třeba využívat WE. Výstupní data jsou dostupná na portu **DOUBT**.

Dvojice signálů **CLKA** a **CLKB** naznačuje, že funkce obou popsanych portů může být asynchronní.

3.7 Zpracování signálu, blok Processing

V případě úspěšného provedení záznamu hodnot následuje krok zpracování signálu, jenž spočívá v nalezení přesných vzájemných pozic impulzů v jednotlivých kanálech zařízení. Detekce těchto pozic je stěžejním krokem pro určení rychlosti detonace.



Obr. 3.12: Schematický symbol bloku Processing

Samotný proces zpracování je aktivován při ukončení záznamu pomocí signálu **EN**, jenž je navázán na výstup **DONE** bloku Acquisition. Trojice signálů **DATA**, **BRAM_ENB** a **BRAM_ADDR** představuje rozhraní pro přístup k zaznamenaným datům.

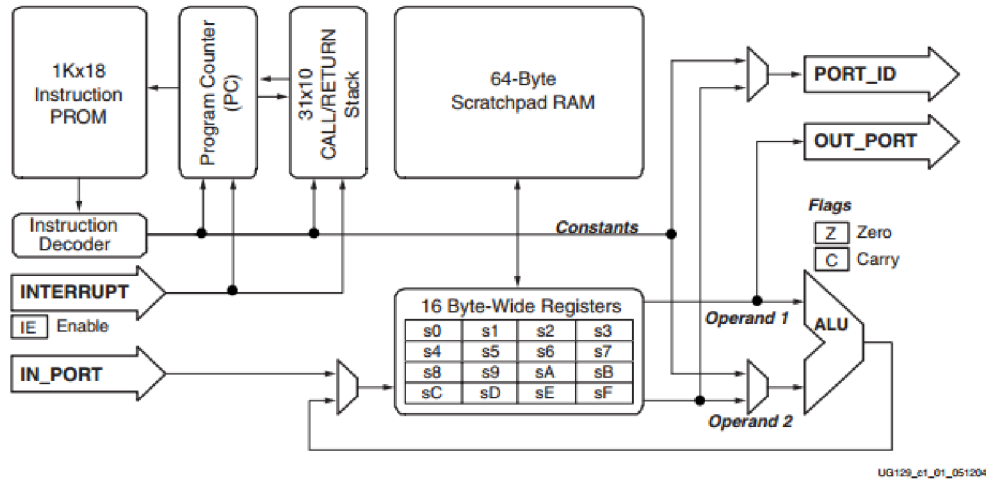
3.8 Použití SW mikroprocesoru Picoblaze

Pro řízení sekvenčně orientovaných součástí návrhu byl použit již dříve v textu zmiňovaný softwarový mikroprocesor Picoblaze.

3.8.1 Popis a vlastnosti

Picoblaze, konkrétně jeho verze KCPSM3, jež byla použita v návrhu, představuje jednoduchý osmibitový mikrokontrolér typu RISC (redukovaná instrukční sada), jehož architektura je zobrazena na obr. 3.13.

Ve schématu lze nalézt běžné součásti mikrokontrolérů jako např. aritmeticko-logickou jednotku (ALU), 16 uživatelských registrů, programový čítač či programovou read-only paměť (PROM), jež je implementována s využitím jednoho 18kb bloku BRAM. Samozřejmě je podpora systémových přerušení [11].



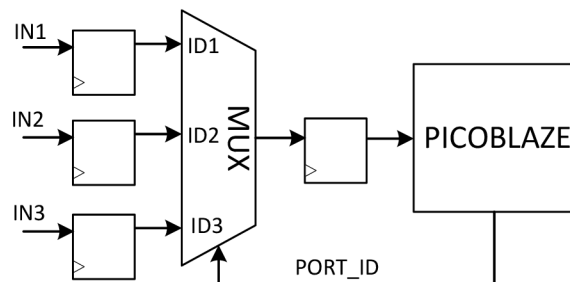
Obr. 3.13: Architektura mikroprocesoru Picoblaze, převzato z [11]

Stejně jako běžné mikrokontroléry, má i Picoblaze čistě sekvenční povahu a hodí se tedy pro implementaci jednoduchých, obzvláště řídicích aplikací. Pro programování kódu v jazyce symbolických adres dle instrukční sady mikroprocesoru je možno využít vývojové prostředí pBlazeIDE firmy Mediatronix.

3.8.2 Připojení Picoblaze k FPGA, Vstupní a výstupní multiplex

Zařazení procesoru Picoblaze do existujícího designu, znamená použití hotového jádra, dostupného např. z webových stránek společnosti Xilinx, přestože se nejedná o oficiálně podporovaný produkt.

Připojení volitelných signálů na vstup či výstup procesoru je možno provést s využitím multiplexorů. Obr. 3.14 zobrazuje princip multiplexace vstupů. Hodnota signálu PORT_ID je periodicky nastavována dle definovaných ID vstupních portů. V případě požadavku procesoru pro čtení z daného portu, je v okamžiku, kdy je hodnota PORT_ID rovna jeho ID generován signál READ_STROBE, který zajistí načtení hodnoty vstupu do procesoru. Princip multiplexace výstupů je analogický.



Obr. 3.14: Multiplexace vstupů mikroprocesoru Picoblaze [11]

4 VYHODNOCENÍ VÝSLEDKŮ SIMULACÍ

Kapitola čtvrtá se zabývá popisem simulačních technik, jež byly využity pro verifikaci funkce vytvořeného systému.

4.1 Nástroje simulace bloku TOP

Základním nástrojem pro simulaci v jazyce VHDL jsou soubory, označované jako **Testbench**. Tyto soubory umožňují použít speciální konstrukce jazyka, jež nejsou syntetizovatelné v reálném HW, ale poskytují možnost řídit časové parametry signálů. Přestože během vývoje systému byly vytvořeny soubory Testbench pro mnohé z vyvíjených komponent, v práci je popsán pouze soubor, analyzující kompletní systém, tedy `top_tb.vhd`.

Výše zmíněný Testbench bloku TOP, byl navrhován tak, aby kromě ověření funkčnosti implementovaného algoritmu umožnil také testovat reakci systému na změny řídicích signálů. Základní struktura testovacích souborů je tvořena několika základními součástmi.

Samotný blok TOP, je v kódu reprezentován jako komponenta, jejíž porty jsou připojeny k vnitřním signálům. Kód dále obsahuje procesy, určené pro generaci hodinových signálů pomocí konstrukcí, umožňujících zařadit do procesu definovatelné zpoždění. Změny řídicích signálů jsou obvykle prováděny prostřednictvím procesu, označovaného jako **Stimulus Process**. Podle toho jestli jsou testovány řídicí signály nebo algoritmus jsou použity některé ze součástí, popsanych v následujících podkapitolách (nepoužité součásti je v kódu třeba označit jako komentáře).

4.1.1 Testování řídicích signálů

V případě testování řídicích signálů a přechodů mezi jednotlivými režimy zařízení je pro generaci vstupních dat využito dvou procesů. První proces představuje generátor pseudonáhodných posloupností, jenž je implementován s použitím funkce **uniform**, jak je naznačeno v následujícím úseku kódu, představujícím sekvenční proces, řízený vzestupnou hranou hodinového signálu.

```
rand_gen : process (clk)
    variable seed1, seed2 : positive := 1;
    variable rand : real;
    variable range_of_rand : real := 40.0;

begin
    if rising_edge(CLK) then
        uniform(seed1, seed2, rand);
        sample_CH1_in <= std_logic_vector(to_unsigned(integer
            (rand*range_of_rand)*gen_pulse, SAMPLE_WIDTH));
    end if;
end process;
```

Samotná funkce `uniform` je součástí slohy `IEEE.math_real`. Vstupní hodnoty `seed1` a `seed2` slouží k prvotní inicializaci pseudonáhodného generátoru. Na základě jejich inicializačních hodnot je rozlišena výstupní pseudonáhodná posloupnost, jež je generována jako reálné číslo v rozsahu 0 až 1. Pro získání hodnot v požadovaném rozsahu stačí výstupní hodnotu `rand` vynásobit maximální hodnotou tohoto rozsahu (`range_of_rand`) [12].

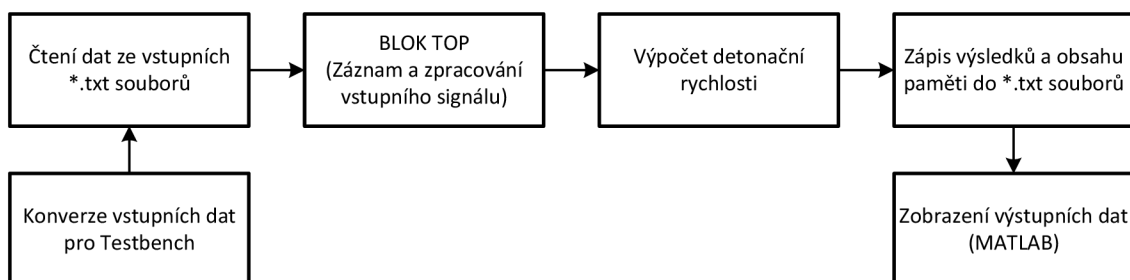
Signál, jenž je generován výše uvedeným způsobem, je dále násoben hodnotou signálu `gen_pulse`, jenž představuje impuls, generovaný separátním procesem v požadovaném čase simulace. Ke generaci impulsu libovolného tvaru lze využít několika příkazů `wait`, jak je naznačeno v následujícím úseku kódu.

```
wait for 100us; gen_pulse <= 20;
wait for 8ns;   gen_pulse <= 40;
wait for 8ns;   gen_pulse <= 80;
wait for 8ns;   gen_pulse <= 100;
wait for 8ns;   gen_pulse <= 100;
wait for 8ns;   gen_pulse <= 20;
wait for 8ns;   gen_pulse <= 1;
```

Správnou funkci systému, lze otestovat různými kombinacemi řídicích signálů nastavených v procesu Stimulus (např. `INIT_R` nebo signálů, jež jsou součástí `Record Detector_ctrl_in`, viz. podkapitola 3.4.1). K vyhodnocení výsledků slouží tzv. **Waveform** soubory, jež jsou popsány v podkapitole 4.1.3.

4.1.2 Testování funkce algoritmu

Testování funkčnosti algoritmu bylo provedeno s využitím záznamů reálných signálů, měřených ve spolupráci s Fakultou chemicko-technologickou Univerzity Pardubice. Pro návrh byla použita převážně data měřená pomocí osciloskopu, jejichž parametry jsou shrnuty v tabulce 2.1. V závěrečné části vývoje byla funkčnost algoritmu ověřena také na datech, naměřených přímo s využitím převodníku ADS4229 a zaznamenaných pomocí nástroje **Chipscope** (zpráva z měření a jeho vyhodnocení je součástí příloh). Celý řetězec, jenž je pro testování algoritmu použit, je zobrazen na obr. 4.1.



Obr. 4.1: Řetězec simulace zpracování záznamů reálných signálů

Aby bylo použití záznamů v testovacím kódu co nejjednodušší, je data nejprve nutno upravit do vhodné formy. Kód je navržen pro práci s textovými soubory, jež mají hodnoty vzorků uloženy na jednotlivých řádcích. Pro konverzi osciloskopických záznamů byl v prostředí MATLAB vytvořen skript s názvem `csv_to_TB.m`. K převodu záznamů z Chipscope, jež byly exportovány s příponou `*.prn`, lze využít skript

`prn_to_TB.m`, jenž je založen na skriptu `Data_input_PRN.m` (autorem je Ing. Michal Kubíček, Ph.D.), doplněném o vytvoření textového souboru ve formátu vhodném pro simulaci.

Čtení dat z generovaných textových souborů je implementováno prostřednictvím procesů `read_File_CHx`, kde `x` představuje číslo kanálu. V deklarační části procesu je třeba uvést kompletní cestu k umístění vybraného textového souboru na disku počítače. Samotné čtení probíhá prostřednictvím funkcí `readline` (přečtení řádku souboru) a `read` (přečtení hodnoty na daném řádku), jež jsou součástí slohy `std.textio`. Hodnoty vzorků, vyčtené z textových souborů jsou přivedeny přímo na datové vstupy bloků `Detector` (`detector_in.DATA`) a simulují tedy výstupní data převodníků po úpravě, popsané v podkapitole 3.3.3.

Pro výpočet detonační rychlosti dle vztahu (2.2) je třeba do proměnných `x_chy` (`y` odpovídá číslu kanálu) v procesu `get_VOD` zadat vzájemné pozice vláken, odpovídající jednotlivým kanálům, v milimetrech. Výsledná detonační rychlost je dostupná jako hodnota signálu `detected_VOD`.

Po dokončení výše uvedeného výpočtu je zahájen proces uložení obsahu blokové RAM do textového souboru. K vytvoření souboru a záznamu hodnot je využito funkcí `write` a `writeline`. Výstupem simulace je `x` textových souborů s názvy `OUTPUT_CHx.txt` (obsah BRAM) podle počtu použitých kanálů a soubor `OUTPUT_RESULTS.txt`, jenž obsahuje hodnoty nalezených pozic impulzů a výslednou detonační rychlost. Výchozím úložištěm pro uvedené soubory je kořenová složka projektu.

K zobrazení obsahu výstupních souborů lze využít skript `show_data.m`. Při spuštění je nutné zvolit soubor s názvem `OUTPUT_CH1.txt`, ostatní soubory jsou načteny automaticky.

4.1.3 Waveform soubory

Pro analýzu průběhu libovolného signálu či komponenty návrhu je při simulaci možno využít tzv. Waveform souborů. Jedná se o soubory s příponou `*.wcfg`, prostřednictvím kterých je uživateli umožněno uložit aktuální nastavení sledovaných signálů v okně simulace a není tedy nutné opakovaně nastavovat formát interpretace, barevné rozlišení signálů či jejich uspořádání.

V návrhu bylo vytvořeno několik Waveform souborů, pro snadnou analýzu nejdůležitějších bloků. Soubory jsou přiloženy spolu se zdrojovými kódy testovacích souborů.

ZÁVĚR

V rámci diplomové práce bylo nejprve stručně pojednáno o principech, používaných v zařízeních na měření parametrů energetických materiálů. Důraz byl kladen na popis detonační rychlosti a principu optické metody.

Na základě získaných poznatků a s využitím záznamů reálných signálů byl v prostředí MATLAB vytvořen funkční model, umožňující detekovat přesnou pozici impulzu v jedno nebo vícekanálovém záznamu signálů optických sond. Model byl navržen ve variantě, pracující s čísly s plovoucí řádovou čárkou i v celočíselné variantě, umožňující simulovat problémy, související s omezeným číselným rozsahem hardwarové implementace.

Třetí kapitola textu, poskytuje stručný popis základní struktury plně implementovatelného systému, jehož návrh byl hlavní náplní diplomové práce a jenž byl navrhován pro použití na vývojové platformě ML505 s čipem VIRTEX-5 firmy Xilinx. Detailní popis návrhu je součástí neveřejné verze diplomové práce.

V rámci návrhu systému byl na základě dříve vytvořeného modelu úspěšně implementován algoritmus detekce, jehož funkčnost byla ověřena na sadě záznamů, dostupných z měření reálných signálů optických sond. Popis a vyhodnocení jednotlivých měření jsou uvedeny jako součást příloh.

V době dokončení diplomové práce byl projekt, jehož součástí je návrh uvedeného systému, ve fázi testování. Ve finálním návrhu bude možno dosáhnout poměrně velkého zjednodušení kódu odstraněním parametrů, jež byly pro testovací účely navrženy ve více variantách.

LITERATURA

- [1] National Institute of Rock Mechanics. Evaluation of explosives performance through in-the-hole detonation velocity measurement [online]. 2001 [cit. 2013-12-15]. Dostupné z: <http://arblast.osmre.gov/downloads/International/India%20papers/VOD-S_T-final.pdf>
- [2] TETE, Aruna D., A.Y. DESHMUKH a R.R. YERPUDE. Velocity of detonation (VOD) measurement techniques practical approach. *International Journal of Engineering and Technology* [online]. 2013 [cit. 2013-12-15]. Dostupné z: <<http://www.sciencepubco.com/index.php/ijet/article/download/1023/817>>
- [3] MEYER, Rudolf, Josef KÖHLER a Axel HOMBURG. Explosives, 6th, Completely Revised Edition [online]. Germany: Wiley-Vch, 2007. ISBN 978-3-527-31656-4. Dostupné z: <<http://miningandblasting.files.wordpress.com/2009/09/explosives-6th-edition-by-meyer-kohler-and-homburg-2007.pdf>>
- [4] CHAN, Edwin M., Vivian LEE, Samuel P. MICKAN a Phil J. DAVIES. Low-Cost Optoelectronic Devices to Measure Velocity of Detonation [online]. 2005 [cit. 2013-12-15]. Dostupné z: <http://www.eleceng.adelaide.edu.au/thz/publications/chan_2005_proc_spie.pdf>
- [5] The evolution of FPGA coprocessing. STRICKLAND, MIKE. ALTERA. *Hearst Electronic Products* [online]. San Jose, CA, 2010 [cit. 2014-05-02]. Dostupné z: <http://www.electronicproducts.com/Digital_ICs/Standard_and_Programmable_Logic/The_evolution_of_FPGA_coprocessing.aspx>
- [6] XILINX, Inc. *ML505/ML506/M L507 Evaluation Platform: User Guide (UG347, v3.1.2)* [online]. 2011 [cit. 2014-05-05]. Dostupné z: <http://www.xilinx.com/support/documentation/boards_and_kits/ug347.pdf>
- [7] XILINX, Inc. *Virtex-5 Family Overview: Product Specification (DS100, v5.0)* [online]. 2009 [cit. 2014-05-10]. Dostupné z: <http://www.xilinx.com/support/documentation/data_sheets/ds100.pdf>
- [8] Texas Instruments. *ADS4229: Dual-Channel, 12-Bit, 250-MSPS Ultralow-Power AD* [online]. 2011 [cit. 2014-05-10]. Dostupné z: <<http://www.ti.com/lit/ds/symlink/ads4229.pdf>>
- [9] XILINX, Inc. *Virtex-5 FPGA: User Guide (UG190, v5.4)* [online]. 2012 [cit. 2014-05-07]. Dostupné z: <http://www.xilinx.com/support/documentation/user_guides/ug190.pdf>Xilinx. Virtex-5
- [10] XILINX, Inc. *LogiCORE IP Block Memory Generator v7.3: Product Guide (PG058)* [online]. 2012 [cit. 2014-05-12]. Dostupné z: <http://www.xilinx.com/support/documentation/ip_documentation/blk_mem_gen/v7_3/pg058-blk-mem-gen.pdf>
- [11] XILINX, Inc. *PicoBlaze 8-bit Embedded Microcontroller: User Guide (UG129)* [online]. 2011 [cit. 2014-05-17]. Dostupné z: <http://www.xilinx.com/support/documentation/ip_documentation/ug129.pdf>
- [12] Generating random numbers in a VHDL testbench. VHDL Coding Tips and Tricks [online]. 2013 [cit. 2014-05-17]. Dostupné z: <<http://vhdlguru.blogspot.cz/2013/08/generating-random-numbers-in-vhdl.html>>

SEZNAM ZKRATEK

A/D – Analog-to-Digital

ADC – Analog-to-Digital Converter

ARM – Advanced RISC Machine

DCM – Digital Clock Manager

DSP – Digital Signal Processing (Processor)

DDR – Dual Data Rate

FIFO – First-In First-Out

FPGA – Field Programmable Gate Array

IP – Intellectual Property

LED – Light Emitting Diode

LVDS – Low Voltage Differential Signaling

JTAG – Joint Test Action Group

RAM – Random Access Memory

ROM – Read Only Memory

RTL – Register Transfer Level

SDR – Single Data Rate

SPI – Serial Peripheral Interface

VHDL – VHSIC (Very High Speed Integrated Circuit) Hardware Description Language

VoD – Velocity of Detonation

SEZNAM PŘÍLOH

A	Analýza dostupných dat za použití různých konfigurací modelu	28
B	Popis a vyhodnocení záznamů pořízených pomocí ADS4229 a ChipScope	29
B.1	Měření s použitím skleněných vláken.....	30
B.2	Měření s použitím plastových vláken	34

A ANALÝZA DOSTUPNÝCH DAT ZA POUŽITÍ RŮZNÝCH KONFIGURACÍ MODELU

Tabulka A.1 obsahuje srovnání detonačních rychlostí, detekovaných s využitím tří různých variant principu detekce a způsobu předzpracování signálu.

Tab. A.1: Porovnání detekovaných hodnot při různých konfiguracích modelu

Tabulka hodnot detekované VoD v závislosti na konfiguraci							
Soubor s měřením (výbušnina)	Princip detekce	Varianta předzpracování signálu					
		I.		II.		III.	
		VoD _{P21}	VoD _{P32}	VoD _{P21}	VoD _{P32}	VoD _{P21}	VoD _{P32}
A-IX-1-78g	A	5806	6203	5794	6213	5806	6203
	B	5783	6219	5781	6219	5783	6219
	C	5796	6215	5792	6216	5796	6215
Bleskovice	A	6778	6616	6783	6613	6778	6616
	B	6804	6608	6804	6605	6804	6608
	C	6770	6605	6770	6611	6770	6605
Np2_04	A	4908	4198	4908	4177	4908	4198
	B	4851	4265	4846	4271	4851	4265
	C	4923	4159	4963	4067	4923	4159
Np10_07	A	4791	5421	4794	5421	4791	5421
	B	4822	5421	4808	5421	4822	5421
	C	4799	5421	4797	5424	4799	5421
PI-Np10	A	7962	7700	8011	7659	7962	7662
	B	7957	7672	8034	7682	7957	7677
	C	7975	7721	7993	7596	7975	7598
Politol	A	2358	1897	2364	1898	2358	1898
	B	2370	1889	2368	1905	2370	1904
	C	2360	1902	2359	1901	2359	1902
Semtex	A	8016	7095	7984	7100	8011	7100
	B	8016	7100	7921	7100	8011	7100
	C	8016	7095	8005	7100	8011	7100
Tetryl-74g	A	5924	6090	5922	6090	5922	6092
	B	5879	6105	5907	6074	5879	6105
	C	5924	6087	5929	6089	5924	6087

Pozn. Všechny hodnoty uvedeny v m/s.

B POPIS A VYHODNOCENÍ ZÁZNAMŮ POŘÍZENÝCH POMOCÍ ADS4229 A CHIPSCOPE

Dne 16.5.2014 byla provedena sada měření s využitím desky s převodníkem ADS4229, propojeným s vývojovou deskou ML505. Testy byly provedeny v detonační komoře chemicko-technologické fakulty Univerzity Pardubice. Kód VHDL, jenž byl použit pro konfiguraci čipu VIRTEX-5, byl redukován pouze na bloky rozhraní A/D převodníku, procesoru Picoblaze (ovládání periférií desky) a jednoduchý proces, generující trigger.

Záznam obou kanálů převodníku byl proveden pomocí nástroje Chipscope v délce 32768 vzorků na kanál. Současně s použitím Chipscope byl také proveden záznam s využitím Osciloskopu Tektronix.

Během měření byly testovány obě varianty optických vláken, tedy plastové i skleněné, přičemž při použití skleněné varianty byl signál rozdělen splitterem pro osciloskopický záznam.



Obr. C.1: Fotografie měřicí aparatury



Obr. C.2: Příprava výbušniny Semtex pro měření

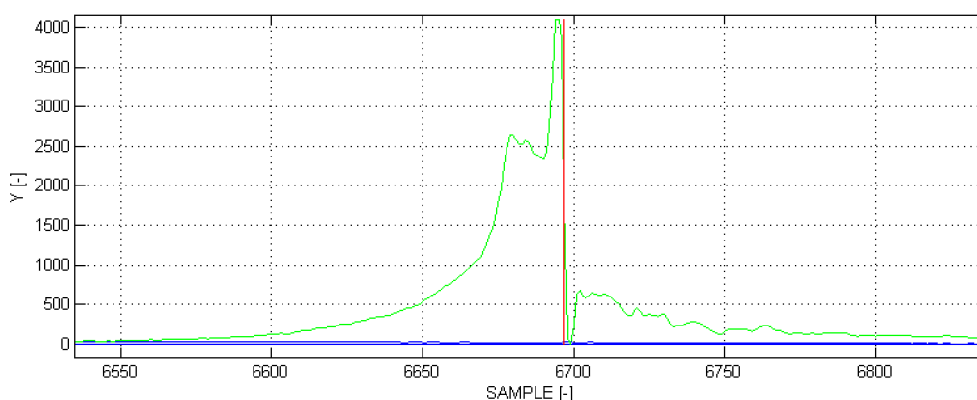
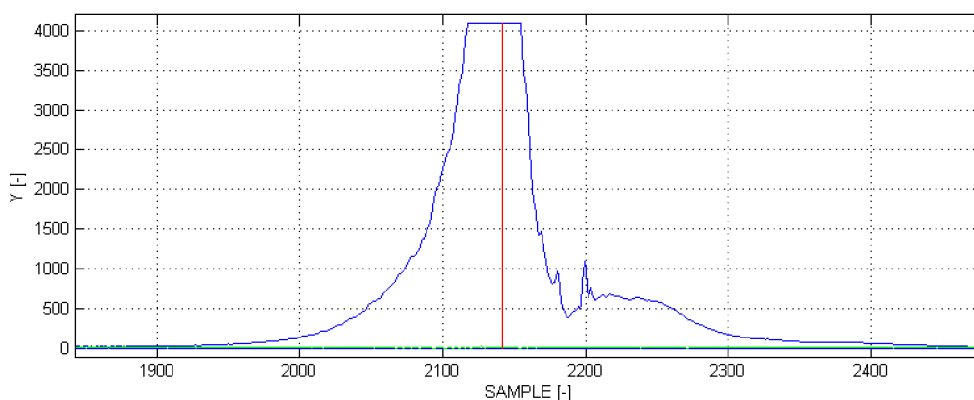
B.1 Měření s použitím skleněných vláken

Následující část příloh obsahuje vyhodnocení měření na jednotlivých výbušninách, zaznamenaných s využitím nástroje **Chipscope**. ΔX v tabulkách hodnot představuje rozdíl poloh obou kanálů (tzn. vzdálenost mezi dvěma použitými optickými vlákny). Barvy signálů zobrazených v detailech pulzů neodpovídají barvám ve schématech měření. Indexy A a D, uvedené u popisu detonačních rychlostí, slouží k rozlišení použitých metod detekce.

1. BLESKOVICE

Pozn. Analyzovaný průběh odpovídá druhému odpalu Bleskovice.

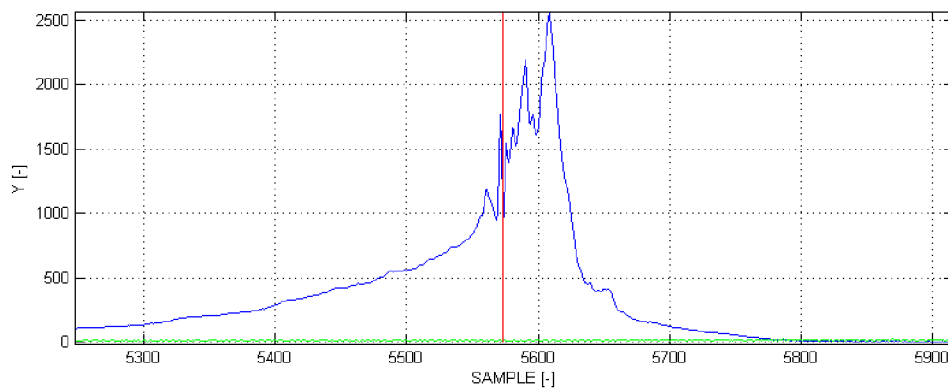
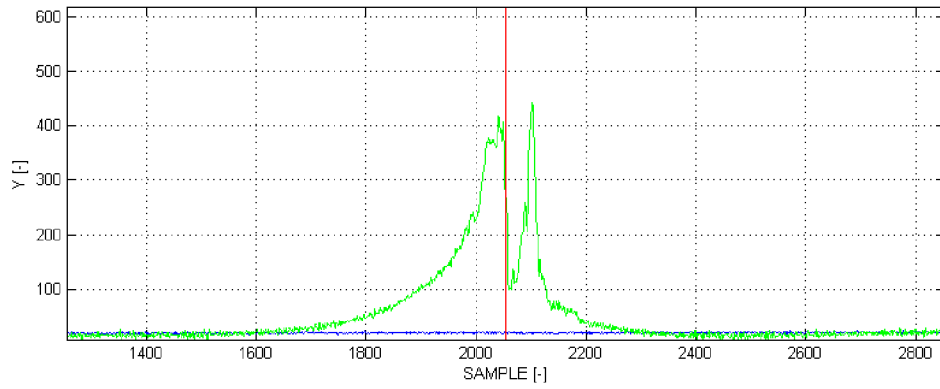
TRESHOLD [-]	500	ΔX [mm]	126
VoD_A [$m \cdot s^{-1}$]	6915	VoD_D [$m \cdot s^{-1}$]	6873



Obr. C.3: Bleskovice. Měřeno skleněnými vlákny. Tvar impulzů a vyznačení detekované pozice

2. SEMTEX

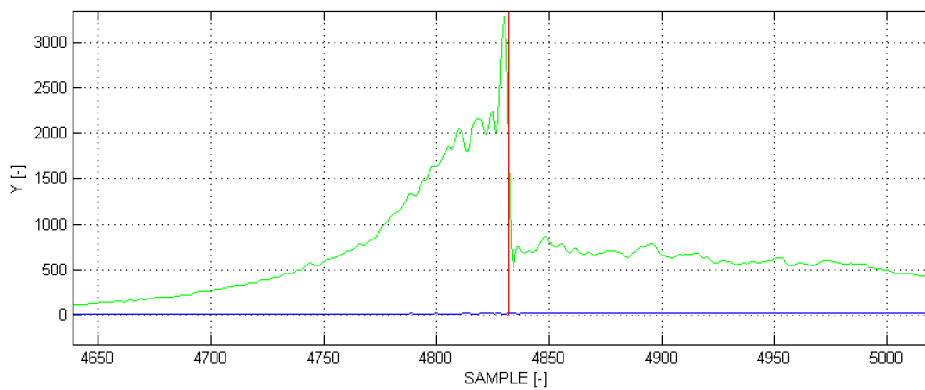
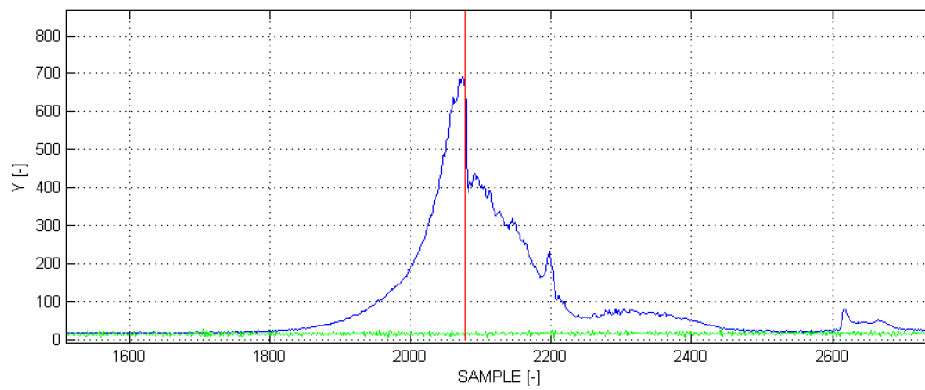
TRESHOLD [-]	500	ΔX [mm]	100
VoD_A [$m.s^{-1}$]	7104	VoD_D [$m.s^{-1}$]	6998



Obr. C.4: Semtex. Měřeno skleněnými vlákny. Tvar impulzů a vyznačení detekované pozice

3. EMULZE PULSAR

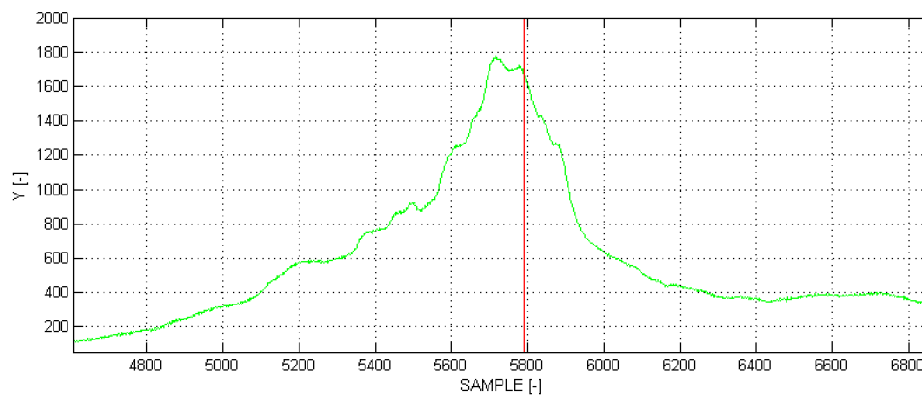
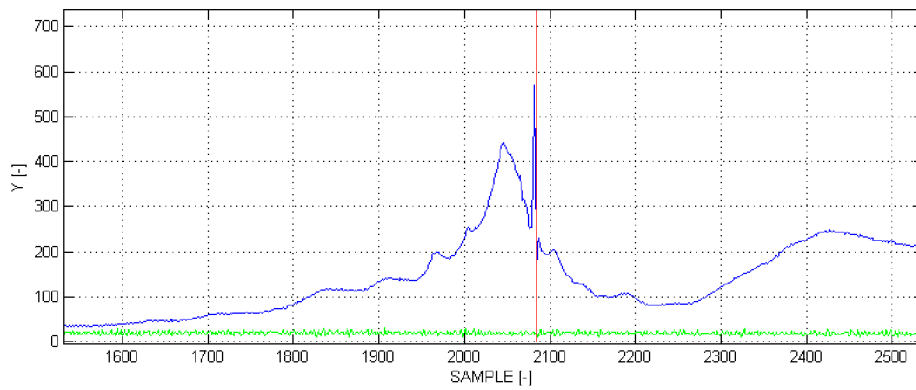
TRESHOLD [-]	200	ΔX [mm]	62
VoD_A [$m.s^{-1}$]	5628	VoD_D [$m.s^{-1}$]	5581



Obr. C.5: Emulze Pulsar. Měřeno skleněnými vlákny. Tvar impulzů a vyznačení detekované pozice

4. T/H 50/50

TRESHOLD [-]	200	ΔX [mm]	37
VoD_A [$m.s^{-1}$]	2495	VoD_D [$m.s^{-1}$]	2566



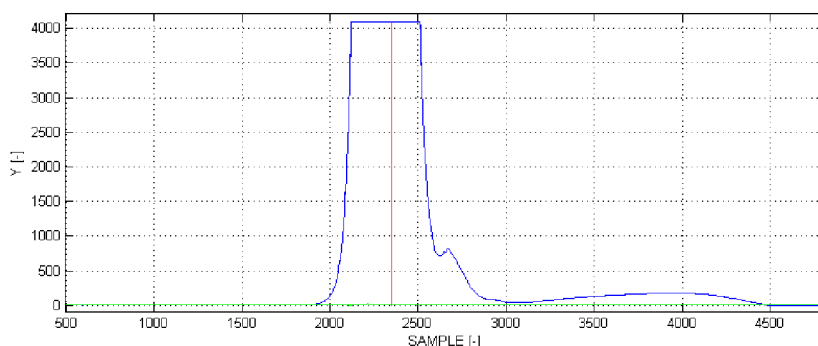
Obr. C.6: T/H 50/50. Měřeno skleněnými vlákny. Tvar impulzů a vyznačení detekované pozice

B.2 Měření s použitím plastových vláken

Z výsledných průběhů testování měření pomocí plastových vláken je zřejmé, že absence utlumovacích článků, jež v předchozích měřeních představoval optický splitter (umožňuje dělit pouze signál skleněných vláken), způsobuje značnou saturaci výstupu převodníků. Pro výraznou podobnost průběhů přesaturovaných signálů, je u většiny vzorků zobrazen detail pouze jednoho z impulzů.

1. BLESKOVICE

TRESHOLD [-]	1000	ΔX [mm]	144
VoD_A [$m.s^{-1}$]	6903	VoD_D [$m.s^{-1}$]	6820

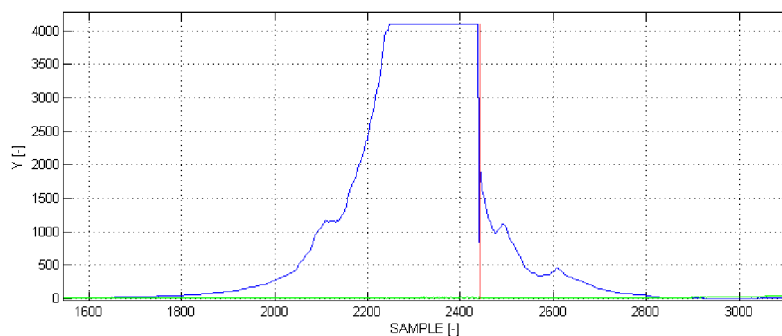


Obr. C.7: Bleskovice. Měřeno plastovými vlákny. Tvar impulzu a vyznačení detekované pozice

2. SEMTEX

Pozn. Lze pozorovat, že ostrý zákmit v oblasti sestupné hrany impulzu způsobuje značné zkreslení při detekci jeho pozice. Řešením může být použití jiného typu předzpracování signálu.

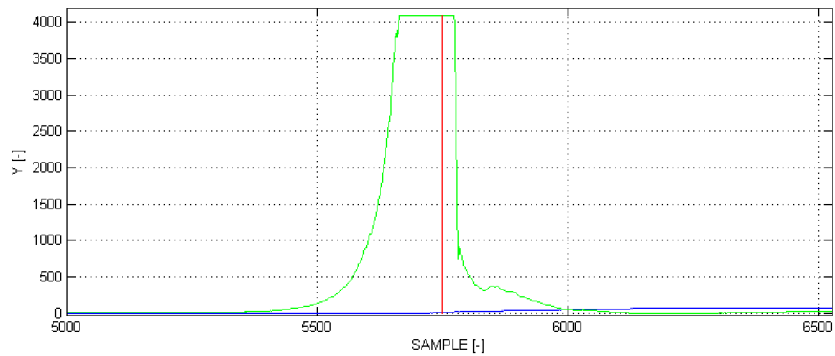
TRESHOLD [-]	500	ΔX [mm]	41
VoD_A [$m.s^{-1}$]	7487	VoD_D [$m.s^{-1}$]	7700



Obr. C.8: Semtex. Měřeno plastovými vlákny. Tvar impulzu a vyznačení detekované pozice

3. EMULZE PULSAR

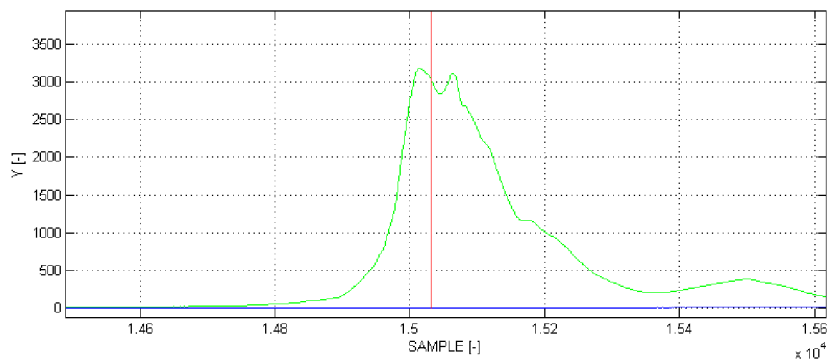
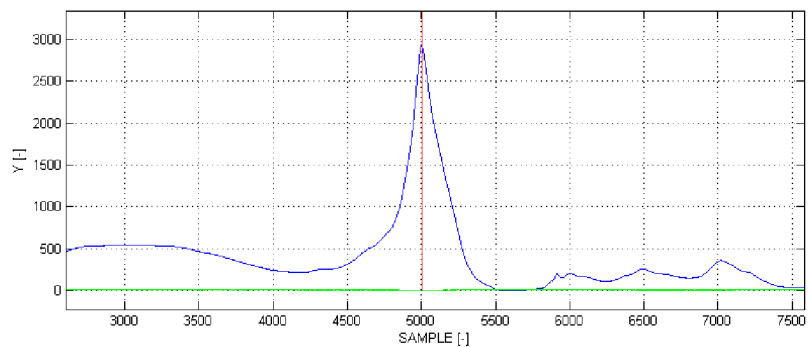
TRESHOLD [-]	300	ΔX [mm]	72
VoD_A [$m \cdot s^{-1}$]	5333	VoD_D [$m \cdot s^{-1}$]	5138



Obr. C.9: Emulze Pulsar. Měřeno plastovými vlákny. Tvar impulzu a vyznačení detekované pozice

4. T/H 50/50

TRESHOLD [-]	300	ΔX [mm]	58
VoD_A [$m \cdot s^{-1}$]	2517	VoD_D [$m \cdot s^{-1}$]	2515



Obr. C.10: T/H 50/50. Měřeno plastovými vlákny. Tvar impulzu a vyznačení detekované pozice