

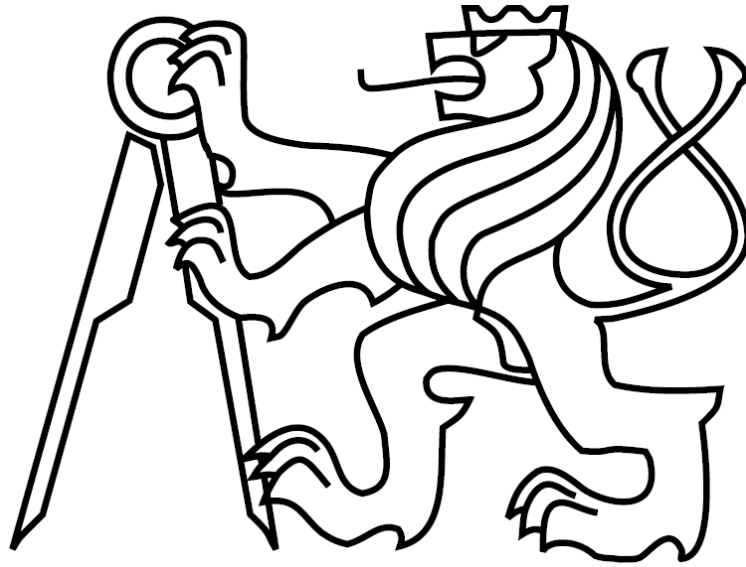
České vysoké učení technické v Praze
Fakulta elektrotechnická

DIPLOMOVÁ PRÁCE

Praha 2008

Martin Zatřepálek

České vysoké učení technické v Praze
Fakulta elektrotechnická



Diplomová práce

**Zabezpečovací zařízení pro železniční stanici založené na
FPGA**

Martin Zatřepálek

Vedoucí práce: Ing. Kubalík Pavel, Ph.D.

Studijní program: Elektrotechnika a informatika, strukturovaný, magisterský

Obor: Výpočetní technika - Projektování číslicových obvodů

květen 2008

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 užití tohoto školního díla ve smyslu § 60 Zákona č.121/2000 Sb. , o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne

.....

Podpis

Anotace

Tato práce se zabývá návrhem a realizací staničního zabezpečovacího zařízení do přípravku s FPGA ve VHDL. Analytická část se zabývá výběrem možností způsobu realizace tohoto zařízení s inspirací ve stávajícím reléovém řešení. Jsou zde popsány požadavky na zařízení, které dodržují bezpečnost železniční dopravy, aby nemohlo dojít k ohrožení lidských životů. Řešení je rozděleno na bloky dle topologie stanice a prvků použitých v kolejišti. Zařízení bylo úspěšně testováno na jednoduchém nádraží, kde splňovalo všechny požadavky, které na něj byly kladeny. Pomocí bloků, které byly vytvořeny s důrazem na jejich univerzálnost, se dá postavit staniční zabezpečovací zařízení pro jakoukoli složitější železniční stanici, která se skládá z bloků navržených v této práci.

Abstract

This thesis deals with a design and implementation of station safety system into the FPGA in VHDL. The analytical part comprises of a selection of available possibilities ways of realization this system with inspiration in the current relay solution. There are described requirements on system, which keep safety of railway traffic. Solution is resolved into the blocks by the topology and elements used in the yard. System was successfully tested on simple station, where fulfills all requirements, which were demanded. Blocks were designed forcefully on their universality. We can set up system for any complex station, which consists of in this thesis created blocks.

Obsah

1	Úvod.....	1
2	Vymezení řešeného problému, cílů a požadavků	2
3	Analýza	3
3.1	Analýza stávajících zařízení	3
3.1.1	Úvod	3
3.1.2	Obecný popis staničního zabezpečovacího zařízení	3
3.1.3	Reléové staniční zabezpečovací zařízení.....	5
3.1.3.1	Bezpečnost	6
3.1.3.2	Požadavky na zřízení	6
3.1.3.3	Bloky Reléového staničního zabezpečovacího zařízení	7
3.1.3.4	Jednoduché nádraží	8
3.1.3.5	Rozhraní bloků	8
3.1.3.6	Stavění cest	9
3.1.3.7	Rušení cest	9
3.1.3.8	Vybavování vlaku.....	10
3.1.4	Závěr.....	11
3.2	Možnosti řešení.....	12
3.2.1	Úvod	12
3.2.2	Zachování stávajících bloků reléového staničního zabezpečovacího zařízení .	12
3.2.2.1	Jedno FPGA jeden blok reléového staničního zařízení	13
3.2.2.2	Jedno FPGA více bloků reléového staničního zařízení	14
3.2.3	Vlastní bloky.....	14
3.2.3.1	Jednoduché nádraží	14
3.2.3.2	Seznam navržených bloků	15
3.2.3.3	Požadavky na zařízení	16
3.2.3.4	Rozhraní.....	16
3.2.4	Testovací aplikace	16
3.2.5	Závěr.....	17
4	Návrh Řešení	18
5	Řešení.....	19
5.1	Základní principy	19
5.1.1	Obecný popis vstupů a výstupů bloků.....	19
5.1.2	Komunikace mezi bloky	19

5.1.3	Stavění cest	20
5.1.4	Rušení cest	22
5.1.5	Vybavování vlaku	23
5.1.6	Návěstidla	23
5.2	Blok Vjzd	25
5.2.1	Popis bloku	25
5.2.2	Požadavky na blok	26
5.3	Blok M.....	26
5.3.1	Popis bloku	26
5.3.2	Požadavky na blok	26
5.4	Blok SD	26
5.4.1	Popis bloku	26
5.4.2	Požadavky na blok	28
5.5	Blok HQ	29
5.5.1	Popis bloku	29
5.5.2	Požadavky na blok	29
5.6	Blok K	30
5.6.1	Popis bloku	30
5.6.2	Požadavky na blok	30
5.7	Implementace jednoduchého nádraží.....	30
5.7.1	Testovací aplikace	30
5.7.1.1	Protokol testovací aplikace	32
5.7.2	Využití prostředků.....	33
5.8	Složitější nádraží.....	34
6	Testování.....	36
6.1	Úvod.....	36
6.2	Testování jednotlivých bloků	36
6.3	Testování na jednoduchém nádraží.....	36
6.4	Testování na složitějším nádraží.....	39
6.5	Výsledky.....	39
6.6	Závěr	40
7	Uživatelská příručka	41
7.1	Popis zapojení jednotlivých bloků a jejich propojování	41
8	Zhodnocení.....	43

9	Závěr	44
10	Literatura a použité vývojové programy	45
11	Seznam obrázků.....	46
12	Seznam Tabulek	47
13	Přílohy	48
13.1	Schémata logických automatů	48
13.2	Testovací aplikace – jednoduché nádraží.....	56
13.2.1	Příkazy	56
13.2.2	Popis registrů	56
13.2.3	Popis bitů v registrech	57
13.2.4	Adresy registrů.....	58
13.3	Obsah CD	60

Přehled použitých zkratek a pojmů:

- **FPGA** – Field Programmable Gate Array – Obvod s vnitřní strukturou umožňující realizaci logických funkcí.
- **CLB** – Základní stavební prvek v FPGA, pro realizaci logických funkcí.
- **LUT** – Paměťový element obsažený v CLB.
- **Slice** – Poloviční řez CLB.
- **FF** – Flip-Flop – hranový klopný obvod
- **Struna** – Signály které převážně procházejí přes všechny bloky.
- **VHDL** – Programovací jazyk použitelný pro simulaci i pro syntézu.

1 Úvod

Cílem této diplomové práce je prozkoumat možnost nahrazení stávajícího zabezpečovacího staničního zařízení, které je na bázi relátkové logiky. Tato zařízení jsou na železnicích používána sice desítky let, ale neexistuje k nim adekvátní uspokojivá dokumentace. Samotné zařízení je postaveno z univerzálních relátkových bloků. Udržování takového zařízení je relativně nákladné, a proto cílem tohoto projektu je zmapovat a navrhnout hardwarovou náhradu za zařízení tohoto typu.

Požadavkem na navržené zařízení bude samozřejmě zachování funkce a hlavně bezpečnosti. Systém bloků by měl být zachován tak, aby z jedné vytvořených a odzkoušených bloků šlo vytvořit složitější staniční zařízení na základě jeho topologie. Měla by být vytvořena přesná pravidla, způsob propojování jednotlivých bloků a rozhraní k ovládacímu zařízení a kolejišti. Tato diplomová práce je součástí většího projektu. Jiná diplomová práce má za úkol vytvořit grafické rozhraní na ovládání staničního zabezpečovacího zařízení. Ve výsledku by s mou diplomovou prací měla jít propojit a fungovat jako celek. Navržené jednoduché nádraží bude implementováno a demonstrováno na FPGA Spartan 3E.

2 Vymezení řešeného problému, cílů a požadavků

Požadavky na projekt:

- Vytvoření bloků ve VHDL pro jednoduché nádraží jako náhradu za reléové staniční zabezpečovací zařízení.
- Univerzálnost bloků – Bloky mají být navrženy tak, aby z nich šlo mimo jednoduché testované nádraží sestavit i jakékoli složitější, které obsahuje stejné prvky v kolejišti.
- Zachování funkce – Vytvořené zapojení bloků musí samozřejmě ve výsledku dávat stejnou funkci jako stávající zařízení. Pro tuto práci má být uvažováno reléové staniční zabezpečovací zařízení.
- Zachování bezpečnosti – Základní požadavek, který bude řešen stejně jako u reléového staničního zabezpečovacího zařízení, kde je definován bezpečný stav, do kterého se v případě potřeby přechází.
- Jednoduché nádraží implementovat a o odzkoušet v přípravku s FPGA.
- Definování protokolu k ovládacímu zařízení – Pro jednoduché nádraží bude navržen řadič a jednoduchý komunikační protokol pro ovládání zařízení.
- Definování pravidel pro zapojování bloků - Vytvořit jednoznačná pravidla pro zapojování bloků dle topologie a prvků ve stanici.
- Dle možnosti vyzkoušet zapojení složitějšího nádraží, které má stejné prvky v kolejišti jako zadané jednoduché nádraží.

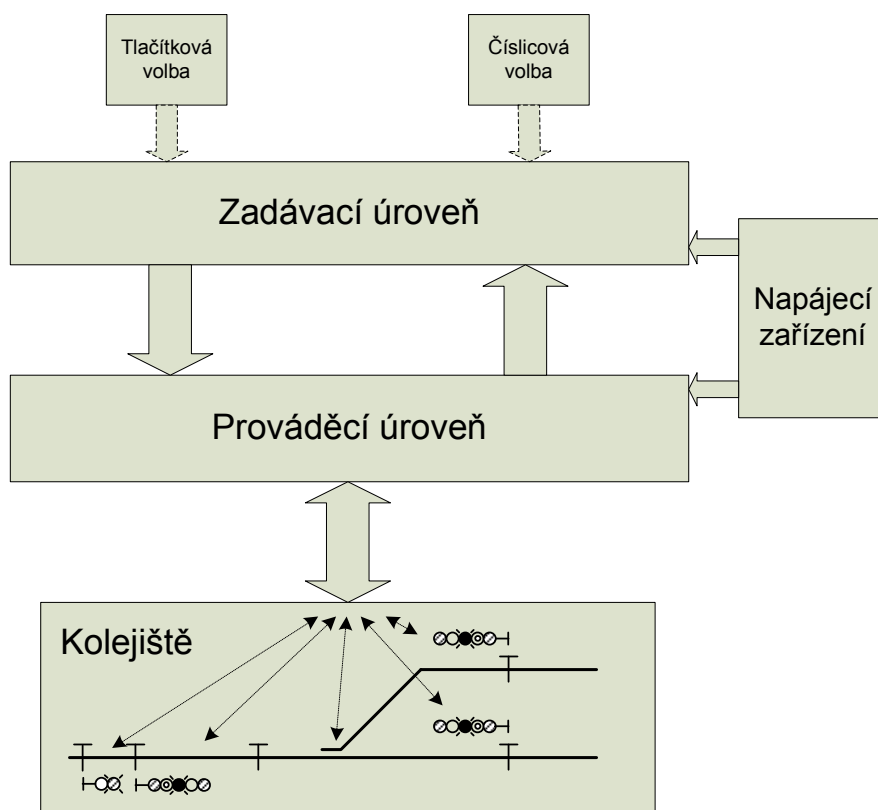
3 Analýza

3.1 Analýza stávajících zařízení

3.1.1 Úvod

V této části analýzy si ujasníme funkci a zapojení staničního zabezpečovacího zařízení jako celku. Dále se blíže zaměříme na reléové staniční zařízení, které by měla tato práce být schopna nahradit. Při studiu stávajících řešení byly využity zdroje uvedené v [1], [2] a dokumentace získaná od Ing. Martin Leso, Ph.D.

3.1.2 Obecný popis staničního zabezpečovacího zařízení



Obr. 1 : Obecné železniční staniční zabezpečovací zařízení

Na obrázku č.1 je znázorněno obecné zapojení železničního staničního zabezpečovacího zařízení. Jednotlivé bloky plní následující funkce:

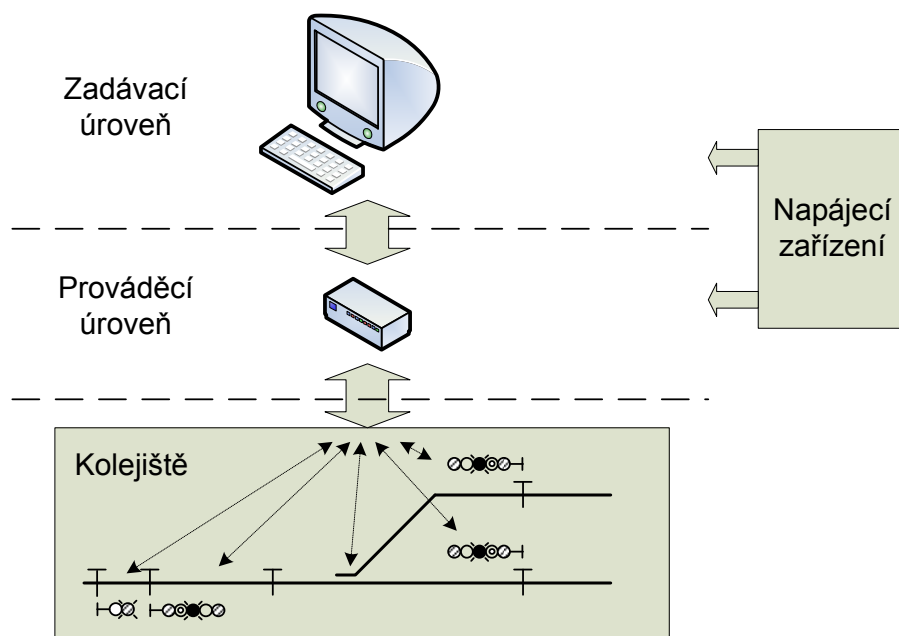
- **Napájecí zařízení** – Blok, který zajišťuje kvalitní a zálohované napájení zadávací a prováděcí úrovně. Tento blok nás v návrhu práce v podstatě nemusí zajímat, protože v případě reálného využití této práce by byl k dispozici.

- **Tlačítková a číslicová volba** – Jedná se v podstatě o zařízení, které zajišťuje interakci mezi člověkem a strojem. Rozdíl mezi tlačítkovou a číslicovou volbou je relativně malý. U tlačítkové volby máme k dispozici pult s nákresem stanice a u každého návěstidla nebo nějakého funkčního prvku, který se dá ovládat, jsou tlačítka. U číslicové volby máme k dispozici pole tlačítek a zadáváme jednoznačnou sekvenci, podle které zařízení pozná co po něm požadujeme.
- **Zadávací úroveň** – Jedná se o rozhraní mezi různými druhy voleb a prováděcí úrovní. Předělává tedy požadavky z voleb do signálů srozumitelných pro prováděcí úroveň. Zároveň zajišťuje zobrazování aktuálního stavu prvků v kolejišti a prováděcí úrovně. Tato část nemá přímý vliv na bezpečnost dopravy a proto se u ní zabezpečení neřeší a je většinou složena z běžných prvků.
- **Prováděcí úroveň** – Na tuto úroveň jsou kladeny největší bezpečnostní nároky. Je zodpovědná za samotné stavění či rušení cest, vybavování vlaku atd. Většinou se skládá z logických automatů, které svou činností zamezují vznikům zakázaných stavů. Právě v této úrovni se nachází srdce reléového staničního zabezpečovacího zařízení. Logické funkce tvoří bloky relátek, které se propojují dle topologie kolejiště. Tato diplomová práce by právě měla nahradit reléové bloky za automaty v FPGA.
- **Kolejiště** – V této úrovni nalezneme členy, které přímo zajišťují sběr dat nebo provádějí nějaké změny v kolejišti. Získává se odsud v podstatě přesný stav na kolejišti pro prováděcí úroveň nebo se tento stav mění na základě signálů z prováděcí úrovně. Členy jsou pomocí fyzikálních principů konstruovány s ohledem na bezpečnost, tak, aby se jakákoli porucha či nežádoucí změna projevila bezpečnějším stavem. Například při poruše povolující návěsti se rozsvítí povolující návěst s nižší rychlostí.

Na následujícím obrázku č.2 je předpokládaný model zapojení navrženého systému.
Popis částí:

- **Zadávací úroveň** – Ovládací a zobrazovací systém vzniklý z jiné diplomové práce. V této úrovni se neřeší samostatná bezpečnost, ale musí být zajištěno správné zobrazení informací získaných z prováděcí úrovně.
- **Prováděcí úroveň** – Logické automaty v FPGA plní úlohu reléových bloků (náplň této diplomové práce). Samotné automaty v prováděcí úrovni mohou být rozděleny do více FPGA, případně se mohou navzájem kontrolovat a vyhodnocovat (není náplň této práce).

- **Kolejiště** – K reálnému kolejišti nebo jeho fyzickému modelu se asi nedostaneme, takže ho budeme simulovat generovanými signály (vstupy do prováděcí úrovně např. obsazení úseku atd.). Tímto způsobem můžeme simulovat i zakázané stavy v kolejišti, případně některé chyby.
- **Napájecí zařízení** – Není cílem této práce, uvažujeme kvalitní bezproblémové napájení. Samotné výpadky napájení nebo problémy spojené s napájením jsou už dnes kvalitně řešeny záložními zdroji, případně přechodem do bezpečnějšího stavu a v posledním případě předpisy.



Obr. 2 : Model zapojení navrženého systému

3.1.3 Reléové staniční zabezpečovací zařízení

Je to typ staničního zabezpečovacího zařízení, který umožňuje stavění a rušení vlakových a posunovacích cest. Kontroluje správnost vybavování vlaku a má hlavní i seřadovací návěstidla závislá na poloze všech projížděných i odvratných výhybek a výkolejek a samozřejmě na volnosti stavěné cesty. Dále zajišťuje svou funkcí, aby se nepostavily současně zakázané cesty. Samotné zařízení je rozděleno do bloků dle topologie stanice a dle prvků umístěných v kolejišti. Každý blok plní funkci logického automatu a nachází se v definovaném stavu. Tato technologie je relativně stará a bohužel přesná adekvátní dokumentace k funkčnosti jednotlivých bloků chybí.

3.1.3.1 Bezpečnost

Bezpečnost je zajištěna definováním „bezpečného stavu“, do kterého se přechází v případě nefunkčnosti některých částí zařízení nebo neadekvátních stavů kolejiště. „Bezpečný stav“ vlastně znamená, že se všechen provoz zastaví a na návěstidlech se zobrazí zakazující znak. Takovéto stavy jsou nežádoucí a když k nim dojde, i přesto se musí zajistit doprava. V takovém případě se přechází na provoz dle pravidel v předpisech D1 a jiných. Samozřejmě se těmto stavům snažíme předcházet, takže je kladen důraz na spolehlivost a dostupnost zařízení.

3.1.3.2 Požadavky na zřízení

Zařízení musí umožňovat:

- Vjezdy vlaků ze všech traťových kolejí na všechny dopravní koleje určené pro vjezdy vlaků.
- Odjezdy vlaků ze všech dopravních kolejí na všechny traťové koleje.
- Posunovací cesty podle požadavků dopravní technologie.

Před rozsvícením povolující návěsti pro jízdní cestu musí být splněné požadavky:

- Všechny projížděné výhybky ve vlakové cestě, začínající obvodem výhybek přilehlých k hlavnímu návěstidlu na začátku vlakové cesty, musí být postavené ve správné poloze.
- Všechny výhybky uprostřed staniční koleje, ze kterých se staví vlaková cesta a které nepatří mezi výhybky přilehlé k předcházejícímu návěstidlu, musí být postavené ve správné poloze.
- Všechny výhybky a výkolejky, které zajišťují boční ochranu stavěné vlakové cesty, musí být v odvrácené poloze.
- Nesmí být postavena žádná současně zakázaná cesta.
- Musí být dané předepsané souhlasy.
- Traťové zabezpečovací zařízení musí dovolovat odjezd vlaku.
- Světelné návěstidlo, u kterého cesta končí, nesmí být zhasnuté.
- Všechna výstražná světelná zařízení v jízdní cestě musí dávat s předepsaným předstihem výstrahu.
- Musí být vykonaný závěr cesty. (Jsou dva typy: Předběžný závěr – Vlak není v rozhodujícím úseku před návěstidlem. Úplný závěr – Vlak je v rozhodujícím úseku před návěstidlem.)
- Volnost vlakové cesty musí být kontrolována v celé svoji délce.

Další požadavky:

- Všechny příjezdy musí být zabezpečené výstražným světelným zařízením.
- Volnost posunovací cesty musí být kontrolována od návěstidla, které posun povoluje až za poslední ústředně přestavovanou výhybku.
- Při jízdě vlaku se musí povolující návěst změnit na zakazující obsazením prvního úseku ve vlakové cestě.
- Při posunu se povolující návěst musí změnit na zakazující okamžitě po uvolnění prvního úseku za návěstidlem.
- Závěr cesty se ruší automaticky postupným obsazením a uvolněním úseků.

Závěr neuskutečněné cesty se může zrušit:

- Po změně povolující návěsti na návěst zakazující.
- Se zpožděním tak, aby vlak nebo posunující díl, který už nestačil zastavit před návěstidlem, mohl jízdu bezpečně dokončit.

Zpoždění při rušení závěru neuskutečněné cesty:

- Za 5 sekund, jestliže nebyl uskutečněn úplný závěr (vlak není v rozhodujícím úseku před návěstidlem).
- Za 1 minutu, když byl uskutečněn úplný závěr posunovací cesty.
- Za 3 minuty, když byl uskutečněn úplný závěr vlakové cesty.

3.1.3.3 Bloky Reléového staničního zabezpečovacího zařízení

Prováděcí úroveň se u reléového staničního zabezpečovacího zařízení skládá z 11 bloků. Tyto bloky jsou zasazovány do panelů a propojeny dle topologie stanice. Každý blok je samostatný logický automat nebo jeho určitá část. Základní jejich funkce se dají odvodit z názvů. Bohužel přesný popis nebo dokumentace nejsou k dispozici. Schémata bloků naleznete v příloze.

Seznam bloků reléového staničního zabezpečovacího zařízení:

H - Závislostní blok hlavního návěstidla

W - Doplnkový blok hlavního návěstidla

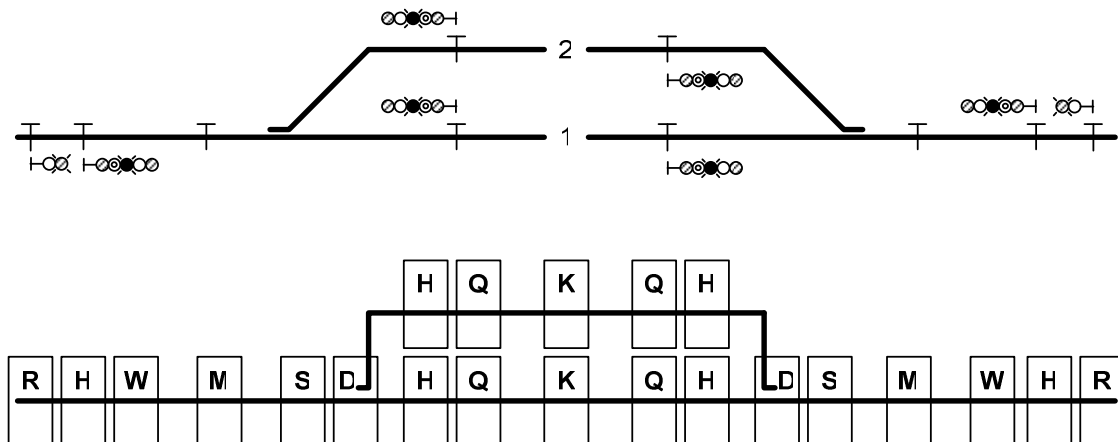
Q - Doplnkový blok odjezdového návěstidla

R - Rychlostní blok

- A - Blok seřadovacího návěstidla mezi výhybkami
- B - Blok seřadovacího návěstidla u bezvýhybkového úseku
- C - Blok seřadovacího návěstidla u kusé koleje
- M - Blok bezvýhybkového izolovaného úseku
- S - Blok výhybkového izolovaného úseku
- K - Blok dopravní koleje
- D - Blok výhybkový

3.1.3.4 Jednoduché nádraží

Na následujícím obrázku je vidět v horní části náčrt jednoduchého nádraží a pod ním k němu odpovídající reléové bloky. Trojice bloků R,H,W v podstatě ovládá vjezdové návěstidlo. Blok M je přiřazen k bezvýhybkovému úseku, kde hlídá obsazování daného úseku a správné vybavování vlaku. Dvojice S,D je přiřazena k jednoduché výhybce a má za úkol starat se o správné přehození výhybky, kontrolu dorazů a vybavování vlaku. Dále dvojice H, Q ovládá odjezdové návěstidlo a blok K je přiřazen ke staniční koleji.



Obr. 3 : Základní nádraží s reléovými bloky

3.1.3.5 Rozhraní bloků

Reléové bloky jsou výhradně logické automaty pracující s logickými hodnotami sepnuto a nesepnuto. Takže se v podstatě jedná o binární hodnoty. Vstupy a výstupy bloků by se daly rozdělit do tří skupin:

- **Signály mezi bloky:**

U reléových bloků se mluví o takzvaných strunách. Bloky jsou zapojeny do série a předávají si určité informace. Tyto struny jdou tedy převážně skrz všechny bloky. Z toho plyne, že určitou část vstupů/výstupů mají bloky stejné. Jedná se hlavně o informaci o uzavírání nebo stavění cesty, jejich rušení a návěstních znacích.

- **Signály do zadávací úrovně**

Tyto signály směřují do nebo ze zadávací úrovně. Informace směřující ven jsou hlavně o různých stavech zařízení, informace do bloků jsou převážně příkazy na postavení a rušení vlakových nebo posunovacích cest.

- **Signály do kolejiště**

Signály směřující do kolejiště jsou hlavně na přestavění výhybek a informace o znacích na návěstidlech. Informace směřující z kolejiště do bloků jsou třeba dorazy výhybek, obsazenost úseků atd. Prvky v kolejišti jsou od prováděcí úrovně striktně galvanicky odděleny.

3.1.3.6 Stavění cest

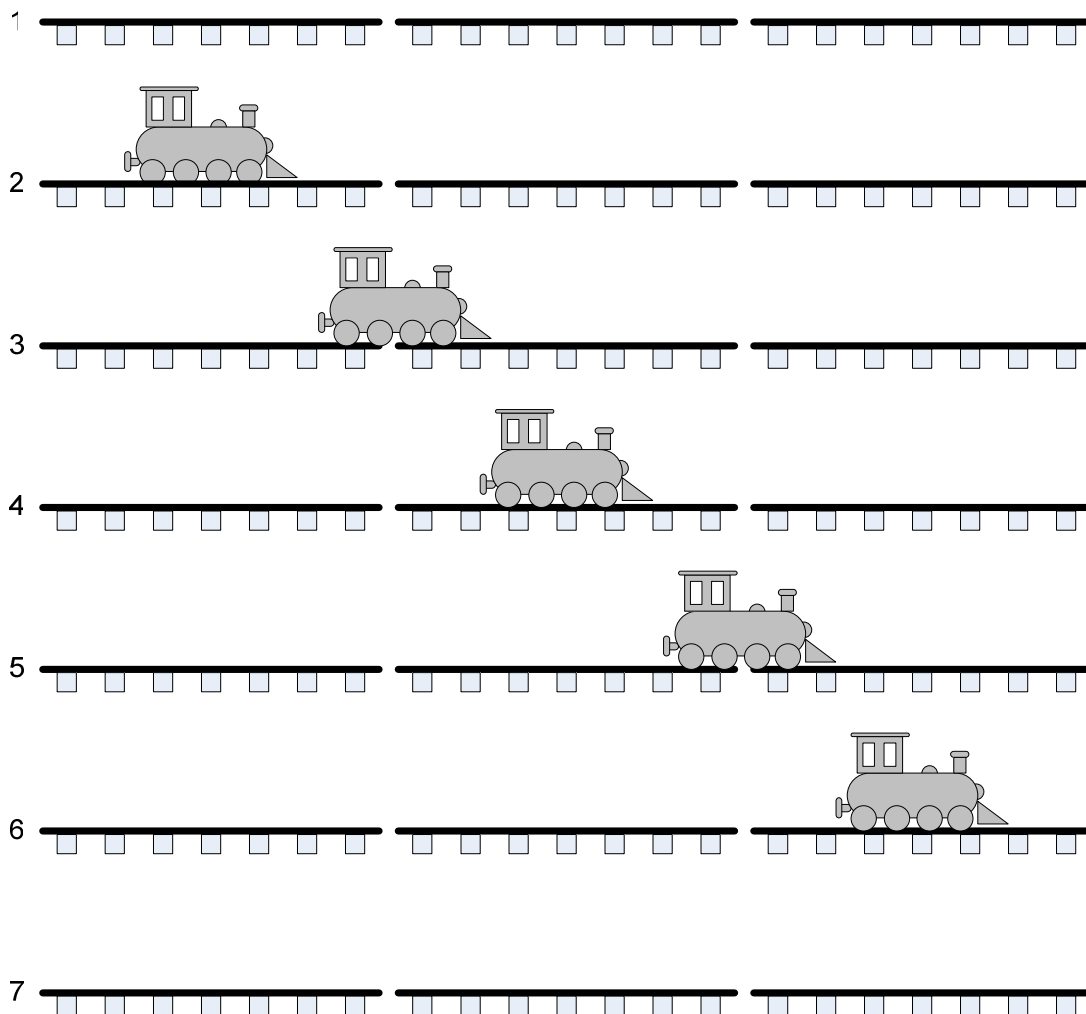
U reléového staničního zabezpečovacího zařízení se stavění cest provádí postupným zadáním počátku a cíle (odkud kam má být postavena cesta). U tlačítkové volby se cesta zadává od počátečního návěstidla k cílovému návěstidlu. Poté se na pultu rozsvítí indikace stavění cesty a v tuto chvíli nelze stavět žádnou jinou. Cesty se tedy do systému zadávají sériově. Po určení počátku a cíle bloky naleznou cestu, zkontrolují, zda jsou úseky obsažené v cestě volné, výhybky se postaví do správného směru a cesta se uzavře. Po uzavření cesty se rozsvítí u počátečního návěstidla odpovídající povolující návěst.

3.1.3.7 Rušení cest

U reléového staničního zabezpečovacího zařízení se rušení cest provádí postupným vytažením počátečního a cílového tlačítka. V takovémto případě se cesta zruší okamžitě, pokud nebyla ještě uzavřena. Pokud cesta již byla uzavřena, tak se povolující návěst na počátečním návěstidle změní na zakazující. Samotné zrušení cesty však trvá různou dobu v závislosti na typu cesty a obsazenosti rozhodujícího úseku před počátečním návěstidlem. Pokud rozhodující úsek je volný, cesta se zruší za 5 sekund. Pokud je obsazen a jedná se o vlakovou cestu, tak rušení bude trvat 3 minuty. V případě posunovací cesty a obsazenosti rozhodujícího úseku by to byla 1 minuta.

3.1.3.8 Vybavování vlaku

Na následujícím obrázku je znázorněno 7 případů, na kterých je pod obrázkem popsáno správné vybavení vlaku. Kolejiště, jak je vidět, vždy obsahuje 3 oddělené úseky, které mohou být ve stavu volno (vlak tam není) nebo obsazeno (vlak je v daném úseku). Pokud je právě na rozmezí a zasahuje do dvou úseků, tak jsou obsazené oba úseky. Popis je brán hlavně z pohledu prostředního úseku.



Obr. 4 : Vybavování vlaku

- **Úsek 1**

Vlak se nenachází v žádném úseku. Cesta je v tomto případě postavena zleva doprava a prostřední úsek čeká, až se obsadí úsek předcházející, tedy levý úsek. Pokud by se obsadil prostřední úsek dříve než levý, nastává chyba.

- **Úsek 2**

Vlak vjel na levý úsek. Blok, který má na starosti levý úsek, dává informaci bloku pro prostřední úsek, že je obsazen. Nyní se tedy očekává, že se obsadí prostřední úsek dříve než se uvolní levý úsek. Pokud by se tak nestalo, nastává chyba.

- **Úsek 3**

Vlak vjíždí do prostředního úseku. Je tedy v obou zároveň. Nyní se tedy očekává uvolnění levého úseku, popřípadě pokud by byl vlak hodně dlouhý, tak obsazení pravého úseku. Pokud by se uvolnil prostřední úsek, dojde k chybě.

- **Úsek 4**

Vlak opustil levý úsek. Blok pro prostřední úsek ví, že vlak je jen v jeho úseku a očekává tedy, že obsadí úsek následující za ním, tedy pravý. Pokud by byl prostřední úsek uvolněn, vlak by jakoby „na chvíli zmizel“, dojde k chybě.

- **Úsek 5**

Vlak vjíždí do pravého úseku. Nyní jsou opět obsazeny dva úseky najednou. Prostřední blok očekává uvolnění svého úseku. Pokud by se dříve uvolnil pravý úsek, dojde k chybě.

- **Úsek 6**

Vlak opustil prostřední úsek. Tímto práce pro prostřední blok končí a tento blok už není uzavřený. Může se tedy použít pro další cesty.

- **Úsek 7**

Vlak opustil i pravý úsek. Vybavování vlaku proběhlo úspěšně.

Vlak je tedy bloky kontrolován, aby nezmizel, popřípadě se nerozdělil na dva. Správné vybavování vlaku tedy probíhá tak, že vlak, který je například v prvním úseku, musí nejdříve obsadit úsek následující, tedy druhý ve směru jízdy, dříve než opustí první úsek. A opět, pokud je obsazen druhý úsek, musí vlak nejdříve obsadit třetí úsek než uvolní druhý úsek.

3.1.4 Závěr

V této části jsme si tedy ukázali zapojení obecného staničního zabezpečovacího zařízení, popsali jsme si jeho funkci a v obr.č.2 jsem navrhl model zapojení s FPGA. Stručně jsme si

ukázali základní funkce reléového staničního zabezpečovacího zařízení (stavění, rušení cest atd.) a seznámili jsme se s požadavky, které jsou na toto zařízení kladeny. Jelikož toto staniční zařízení má být pro tuto diplomovou práci inspirativní, budeme z těchto požadavků určitě dále vycházet.

3.2 Možnosti řešení

3.2.1 Úvod

V této části analýzy se budeme zabývat různými možnostmi předělání reléového zabezpečovacího zařízení do logických automatů pro FPGA. Na konci této části se rozhodneme jakým směrem tedy bude tato diplomová práce směřovat.

3.2.2 Zachování stávajících bloků reléového staničního zabezpečovacího zařízení

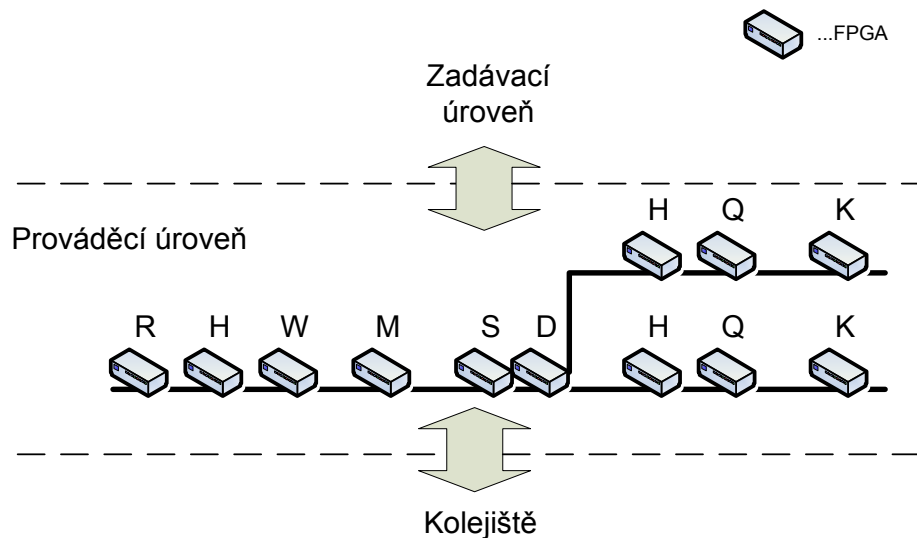
Jedna ze základních možností je tedy vzít stávající bloky reléového staničního zabezpečovacího zařízení a přepsat je do logických automatů. Při studiu a pokusech o tuto možnost jsem narazil na několik následujících problémů:

- **Neexistuje adekvátní dokumentace k blokům** – Není nikde řečeno co přesně má jaký blok dělat při určitých vstupech, či jak má reagovat. Sice jsme se po dlouhém shánění dostali ke schémátům bloků (viz. Příloha) ale tato schémata jsou bez popisu.
- **Vstupy / Výstupy bloků jsou většinou obousměrné** – Jeden blok má v průměru něco okolo 40 vstupů / výstupů. Při teoretickém rozdělení každého vodiče na dva, jeden vstupní a jeden výstupní, jsme na 80 vodičích, a když uvážíme, že některé vodiče mohou mít i více jak dvě hodnoty, tak nám počet vodičů, který bychom museli pro každý blok použít, dále roste.
- **Neschopnost oddělit volnou vazbu od základních funkcí** – Každý blok obsahuje dále navíc různou logiku pro volnou vazbu. Tyto vstupy nebo výstupy se dají různě zapojit dle specifických anomálií stanice. Tato volná vazba není příliš využívaná a pro samotnou hlavní funkci bloku je nepotřebná. V našem případě máme uvažovat standardní stanici a proto nás nezajímá. Případné přidání nových vstupů nebo výstupů do vytvořených automatů pro FPGA by neměl být problém.

- **Samotné propojení bloků není zárukou funkčnosti** – Dověděli jsme se, že při aplikaci bloků na novou stanici je zprovoznění uměním. Samotné propojení bloků nezajišťuje funkčnost. Mezi bloky se musí přidávat různé zpožďovací členy (RC články) atd. Sice relátka použitá v blocích jsou přísně typizovaná (odpad kotvy gravitací, nesvařitelné kontakty atd.), ale pravděpodobně mají různé doby odpadu nebo sepnutí, a proto aby se zajistilo správné pořadí spínání relátek v jednotlivých blocích, musí se pravděpodobně k nim přidávat tyto přídatné obvody.

3.2.2.1 Jedno FPGA jeden blok reléového staničního zařízení

Na obrázku č.5 vidíme náčrtek prováděcí úrovně s FPGA zařízeními. Na každý blok reléového staničního zařízení připadá jedno FPGA. Takovéto zapojení má výhodu v tom, že se zachovává topologie stanice, jako u reléového staničního zařízení a čistě teoreticky, kdyby bloky s FPGA měly naprosto stejné chování jako bloky reléového staničního zařízení, tak by se tyto dvě technologie daly kombinovat. Postupně by se daly reléové bloky vyměňovat za bloky s FPGA.

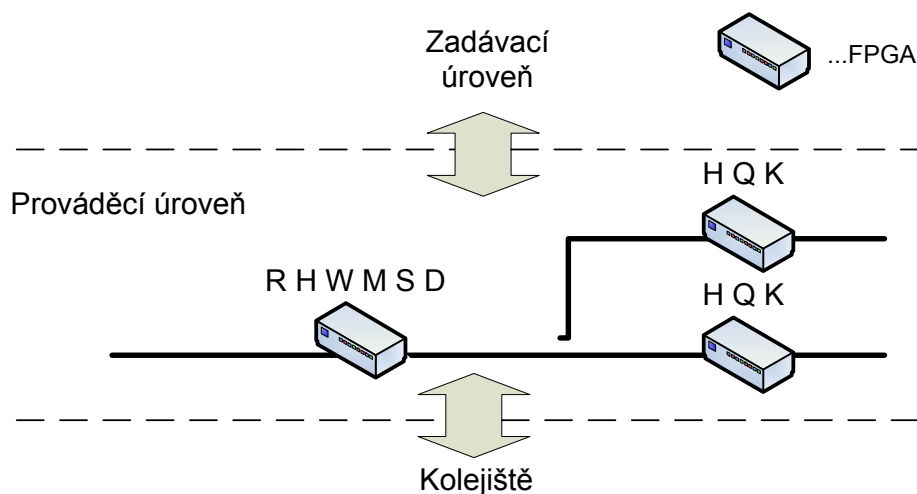


Obr. 5 : Jedno FPGA jeden blok reléového staničního zařízení

Logické automaty bloků nejsou extrémně složité, takže by FPGA mělo využito málo prostředků. Daly by se tedy použít menší levnější FPGA. Zbytek prostředků by se také mohl použít například na zdvojení automatu pro zvýšení bezpečnosti nebo na nějaké diagnostické funkce.

3.2.2.2 Jedno FPGA více bloků reléového staničního zařízení

Na následujícím obrázku je náčrt tohoto zapojení. V FPGA je více bloků odpovídajících bloků reléového staničního zařízení. Odhaduji, že menší nádraží by se do jednoho FPGA vešlo celé. Tímto by došlo k velké úspoře místa a prostředků oproti reléovým blokům (Jeden reléový blok váží cca 13 kg a má rozměry odhadem 30 x 40 x 20 cm, skládají se do panelů. Většinou zabírají celou místnost.).



Obr. 6 : Jedno FPGA více bloků reléového staničního zařízení

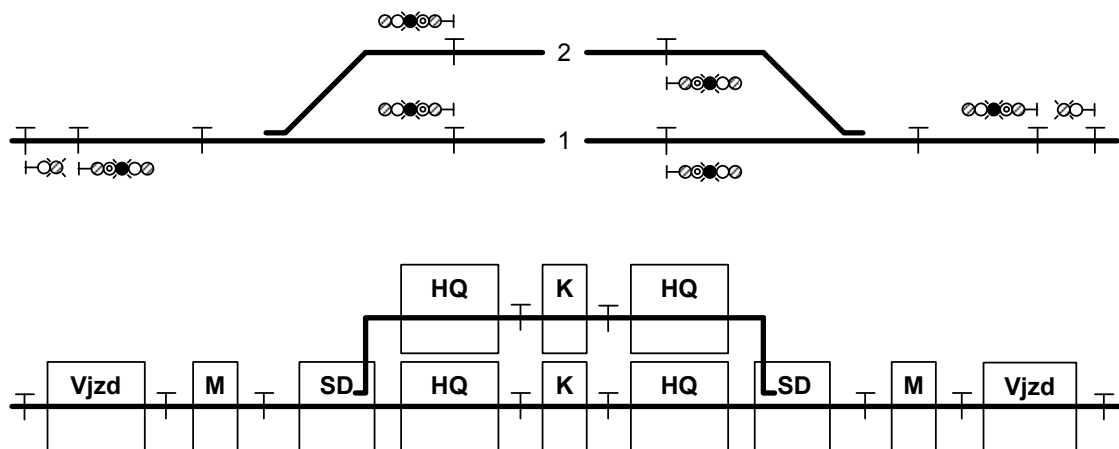
3.2.3 Vlastní bloky

Další možností je vytvořit si vlastní bloky. Inspirovat se u reléového staničního zabezpečovacího zařízení, ale v konečném důsledku je navrhnout po svém. Tímto se zbavíme nežádoucích problémů, které nás u reléového staničního zabezpečovacího zařízení potkávaly, jako třeba nedostatek dokumentace, potřeba zpoždovacích členů, množství vstupů / výstupů a neschopnost oddělit zapojení volných vazeb, které my pro jednoduché nádraží nepotřebujeme.

3.2.3.1 Jednoduché nádraží

Na následujícím obrázku vidíte jednoduché nádraží s navrženými bloky. Bloků je méně než u reléového zabezpečovacího zařízení. Je to z důvodů, protože některé bloky u reléového zařízení pracovaly spolu, byly rozděleny do dvou proto, aby měly typizovanou velikost a váhu. Pro nás toto není omezujícím faktorem a proto pro každý specifický člen v kolejišti

můžeme mít jeden určitý blok. V takovémto jednoduchém nádraží neuvažujeme posunovací cesty.



Obr. 7 : Jednoduché nádraží s navrženými bloky

3.2.3.2 Seznam navržených bloků

Jednoduché nádraží obsahuje tyto navržené bloky:

- **Blok Vjzd** – Tento blok je přiřazen k hlavnímu vjezdovému návěstidlu. Měl by tedy ovládat Hlavní návěstidlo, jeho předzvěst a umožňuje od něj začít stavět vlakovou cestu nebo být cílem vlakové cesty. Od něj postavená vlaková cesta může být zrušena. Blok sleduje obsazenost nebo volnost úseku před hlavním návěstidlem.
- **Blok M** – Jeho hlavní náplní je kontrola obsazenosti nebo volnosti bezvýhybkového úseku, ke kterému je přiřazen. Dále kontroluje správnost vybavování vlaku, tzv. řapání.
- **Blok SD** – Tento blok je přiřazen k výhybce, pro kterou generuje příkazy na přestavění a kontroluje dorazy, zda je výhybka ve správné konečné poloze. Dále blok také kontroluje správnost vybavování vlaku.
- **Blok HQ** - Je přiřazen k hlavnímu odjezdovému návěstidlu. Toto návěstidlo tedy ovládá a umožňuje od něj začít stavět vlakovou cestu nebo být cílem vlakové cesty. Samozřejmě také umožňuje rušení od něj postavených cest.

- **Blok K** – Tento blok je přiřazen ke staniční koleji. Kontroluje tedy obsazenost svého úseku a správnost vybavování vlaku.

3.2.3.3 Požadavky na zařízení

Je nutné, aby ve výsledku navržené staniční zabezpečovací zařízení dělalo to samé jako reléové. Tudiž musíme splnit požadavky uvedené v odstavci 3.1.3.2. Tímto dosáhneme správnou funkci našeho návrhu. Bereme v úvahu jen ty požadavky, které se týkají našeho základního nádraží, tedy v podstatě všechny kromě těch, co se týkají posunovacích cest.

3.2.3.4 Rozhraní

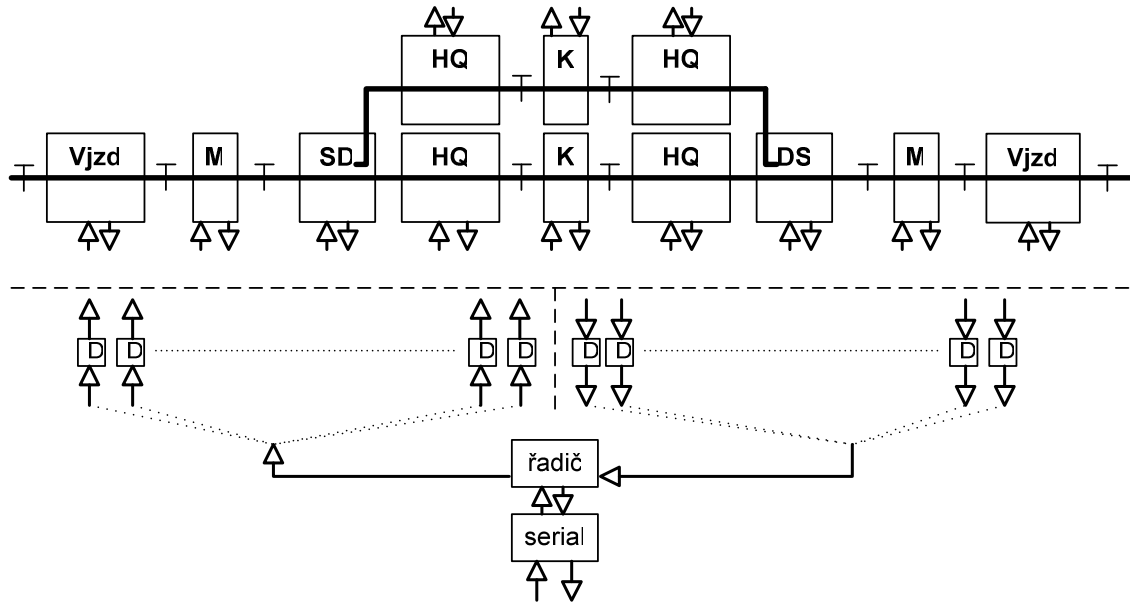
Rozhraní bude prakticky stejné jako u reléového staničního zařízení. Bude rozděleno do tří následujících kategorií:

- Signály mezi bloky
- Signály do zadávací úrovně
- Signály do kolejiště.

3.2.4 Testovací aplikace

Bude navržena testovací aplikace, na které vytvořené bloky otestuji. Tuto aplikaci připojíme k blokům v FPGA a budeme moci sledovat jejich stav, případně jejich činnost. Vzhledem k počtu výstupů a vstupů bude nejlepší stav jednotlivých bloků nádraží zobrazovat na LCD v přípravku FPGA. Nejjednodušší rozhraní k PC je asi sériový port, má výhodu, že má málo signálů (Rx,Tx) a ve VHDL je běžně dostupný. Ten do aplikace tedy také zahrneme a budeme všechny výstupy, které by měly jít do kolejiště nebo do zadávací úrovně, směřovat do PC, kde je podrobíme kontrole. Ze sériové linky budeme moci také generovat příkazy pro stavění, rušení cest a simulaci vybavování vlaku. Bude tedy potřeba vyvinout řadič, který bude zajišťovat posílání výstupů, které se změnily na sériovou linku a zápis příkazů, které přišly do správných bloků.

Na následujícím obrázku je vidět model testovací aplikace. Obsahuje řadič, sériovou linku a pole vstupních a výstupních registrů, které jsou připojeny na vstupy a výstupy navržených bloků. Pro správné rozlišení jednotlivých bloků bude mít každý registr jednoznačnou adresu.



Obr. 8 : Návrh zapojení testovací aplikace

3.2.5 Závěr

Seznámili jsme se tedy s různými možnostmi předělání reléových bloků staničního zabezpečovacího zařízení do logických automatů nebo napsání bloků vlastních. Jako první jsem se pokoušel o předělání stávajících bloků do logických automatů. Tato činnost však díky malé dokumentaci vedla na neúměrně složitá zapojení z důvodů, které jsou uvedeny v odstavci 3.2.2. Vzhledem k tomu se předělávání bloků do VHDL zdá jako problémové a upustil jsem od něj. Jako lepší řešení považuji kompletní předělání bloků s ohledem pouze na požadavky jako celku, tedy na požadavky uvedené v odstavci 3.1.3.2. Vzhledem k tomu, že logické automaty nebudou vycházet ze schémat reléových bloků, bude muset být kladen velký důraz na jejich testování, aby jako celek splnily uvedené požadavky na zařízení. Dále zde byl navržen hrubý model testovací aplikace, kterou po simulaci v PC použijeme na otestování zařízení.

4 Návrh Řešení

Řešení bude tedy postaveno na vlastně vyvinutých blocích. Bude se jednat o vjezdový blok, blok M, SD, HQ a K. Nákres jednoduchého nádraží s odpovídajícími bloky je v analýze na obrázku č.7. Stavění a rušení cest se bude moci provádět vždy od návěstidla k návěstidlu v daném směru. Hlavní část bloků bude tvořit vždy logický automat. Navíc v nich pravděpodobně bude nějaká kombinační logika pro generování specifických výstupů. Bloky, které budou ovládat návěstidla, budou obsahovat „časovač“, který v případě rušení cest bude odpočítávat požadovaný časový úsek.

Samotné hledání cest bude provedeno lavinovitým šířením přes volné bloky určitého tokenu, který když dojde k cílovému návěstidlu, tak na něj bude odpovězeno jiným tokenem. Všechny logické automaty, přes které tento zpětný token projde, přejdou do stavu, kdy přes ně už nepůjde postavit jiná cesta. Vygenerují se adekvátní příkazy pro prvky v kolejišti a po jejich dokončení se cesta uzavře a rozsvítí se povolující návěst. Rušení cest se bude skládat ze dvou cyklů. Nejprve bloky přejdou do rušícího stavu a budou čekat až doběhne odpovídající čas ke zrušení nebo k obnovení cesty příjezdem vlaku na rušenou cestu. Správné vybavování vlaku budou kontrolovat všechny bloky kromě HQ, které jsou přiřazeny do úseku, kde kontroluje ťapání blok SD.

Pro jednoduché nádraží bude vyvinuta testovací aplikace a vše se naimplementuje do FPGA SPARTAN 3E, kde proběhne testování. Pomocí rozhraní přes sériovou linku se zařízení otestuje s jinou diplomovou prací, ve které se vyvíjí ovládací pult.

5 Řešení

5.1 Základní principy

5.1.1 Obecný popis vstupů a výstupů bloků

Každý blok má vstupy nebo výstupy rozdělené dle jejich použití, jak je uvedeno v analýze. Vstupy jsou značeny písmenem **I** a výstupy písmen **V**. Pokud se jedná o signály mezi bloky, je přidáno písmeno **B** a pokud se jedná o signály k ovládacímu zařízení, je přidáno písmeno **O**. U bloků se můžeme setkat tedy s následujícími signály:

- *I_název* – vstup z kolejiště
- *IB_název* – vstup signálu z jiného bloku
- *IO_název* – vstup z ovládacího zařízení
- *V_název* – výstup k prvkům v kolejišti
- *VB_název* – výstup k jiným blokům
- *VO_název* – výstup k ovládacímu pultu

5.1.2 Komunikace mezi bloky

V případě reléového staničního zabezpečovacího zařízení se mluvilo o „strunách“, byly to signály, které si bloky v podstatě předávaly a šířily se převážně přes všechny bloky. Zde je takto můžeme také nazývat. Jedná se o struny **STATUS** a **ZRUS**. Každý blok, kromě určitých výjimek, obsahuje vstupy a výstupy na obě strany pro tyto struny. Pro strunu **STATUS** i **ZRUS** je tato výjimka vjezdový blok, který samozřejmě svůj stav nešíří ven mimo stanici. Pro strunu **ZRUS** je to blok **K** (nebo může být i jiný), který je před blokem odjezdového návěstidla **HQ**, kde zaniká rušení cesty, protože cesta se vždy staví i ruší od jednoho návěstidla k druhému.

Struna **STATUS** je 4 bitový signál, který nabývá hodnot:

- „0000“ – **FREE** – Daný blok je volný, jeho úsek není obsazen ani se přes něj nestaví nebo neruší cesta.
- „0001“ – **INFORM** – Tento token se šíří přes volné bloky při hledání cesty.
- „0010“ – **RESERVED** – Pokud se najde cesta přes volné bloky k cílovému návěstidlu (k cílovému bloku se dostane token **INFORM**), tak cílový blok vrací tento token. Bloky na této cestě se předběžně uzamknou a už lze cestu zrušit jen rušícím procesem nebo dostavěním cesty a vybavením vlaku.

- „0011“ – **CLOSED** – Cílový blok po vygenerování tokenu **RESERVED** zkontroluje náležitosti, které má mít splněné pro uzavření cesty a pošle **CLOSED**. Bloky, které jsou v **RESERVED**, přejdou tedy do stavu **CLOSED** až po kontrole svých povinností. (Např. pro blok výhybky SD, je to kontrola dorazu a červených na okolních návěstidlech, případně kontrola dorazu odvratných výhybek).
- „0100“ – **RESERVED** – Úsek u daného bloku je obsazen vlakem, případně může být blok v chybovém stavu.
- Ostatní hodnoty jsou jako rezerva

Struna **ZRUS** je 2 bitový signál, který nabývá hodnot:

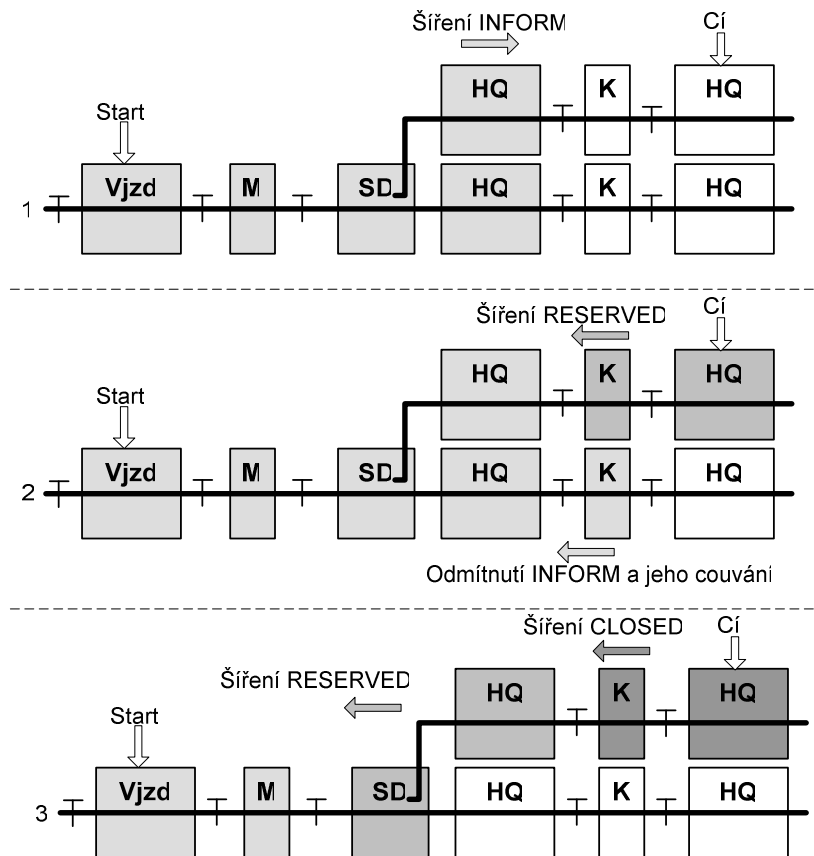
- „00“ – Cesta se neruší – Základní stav
- „01“ – První token pro rušení cesty
- „10“ – Druhý token pro rušení cesty
- „11“ – rezerva

5.1.3 Stavění cest

Postavení cesty se skládá z několika kroků z pohledu ovládacího zařízení:

- 1) Přivedením **log. 1** na vstup **Start** u počátečního bloku.
- 2) Přivedením **log. 1** na vstup **Cíl** u cílového bloku.
- 3) Přivedením **log. 0** na vstup **Start** u počátečního bloku.
- 4) Od počátečního bloku dostaneme informaci (výstup **Cesta**), že se cesta začala stavět.
- 5) Čekáme na informaci od počátečního bloku (výstup **Cesta**), zda se cesta postavila úspěšně či nikoli.
- 6) Přivedeme **log. 0** na vstup **Cíl** u cílového bloku.

Tímto způsobem se zařízení pokusí postavit cestu od vybraného počátku k cíli. Kroky 1 až 3 jsou zadání startu a cíle. Po 3. kroku dá počáteční blok informaci o začátku stavění cesty (krok 4) a vyšle směrem kam má být hledána cesta token **INFORM**. Ten se lavinovitě bude šířit přes volné bloky až do chvíle, než narazí na blok, který je odmítne nebo na cílový blok (v obrázku č.9 náčrt 1).



Obr. 9 : Vybrané části šíření signálů při stavění cesty

Pokud šíření v některém směru narazí na blok, který token odmítne (buď je blok obsazen nebo je součástí jiné cesty nebo to je návěst a není cílová v daném směru), začnou se bloky od tohoto místa nazpátek zase uvolňovat. Pokud je nalezen cílový blok (existuje tedy mezi počátečním a cílovým blokem spojení přes volné bloky), odpoví tento blok tokenem **RESERVED** (v obrázku č.9 nákr. 2).

Token **RESERVED** se bude šířit nazpět k počátečnímu bloku a všechny bloky na této cestě přejdou do stavu, kdy už přes ně nebude možno postavit další cestu. Koncový blok zkontroluje svou návěst (svítí-li červená) a odešle token **CLOSED** (v obrázku č.9 nákr. 3) . Postupně jak se přes bloky tento token šíří, tak bloky kontrolují to co mají (např. SD dorazí výhybky) a poté co dojde k počátečnímu bloku, tak se na návěstidle rozsvítí povolující návěst. Tímto je stavění cesty uzavřeno a počáteční blok podá informaci o úspěšnosti postavení cesty. V Posledním 6 kroku se zruší, že daný blok má být cílový.

Pokud je potřeba postavit cestu přes více návěstidel, začne se od začátku po jednotlivých úsecích, protože poté, co je cesta dostavěna, může být od původního cílového bloku postavena další v daném směru. V Našem jednoduchém nádraží to znamená třeba případ, kdy bychom chtěli postavit cestu skrz celé nádraží, průjezd vlaku zleva doprava. Postavíme tedy

nejdříve cestu s počátkem levého vjezdového bloku a cílovým návěstidlem u staniční koleje. A až se cesta postaví, tak postavíme další část od staniční koleje k pravému vjezdovému návěstidlu.

5.1.4 Rušení cest

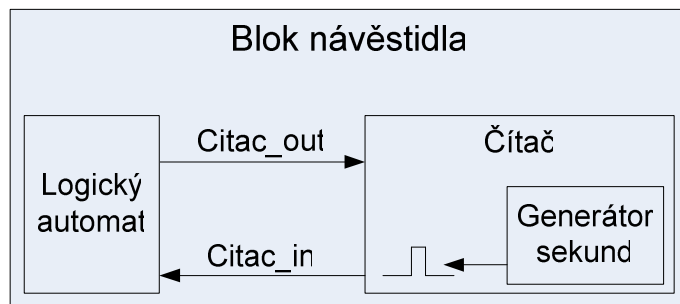
Zrušení cesty se skládá z pohledu ovládacího zařízení z následujících kroků:

- 1) Přivedením **log. 1** na vstupy **Start** a **Cíl** u počátečního bloku.
- 2) Přivedením **log. 0** na vstupy **Start** a **Cíl** u počátečního bloku.

Po provedení této sekvence vyšle počáteční blok informaci o první fázi rušení cesty (struna **ZRUS**). Na počátečním návěstidle se rozsvítí zakazující návěst a dle následujících kritérií se pokračuje. Přejít do druhé fáze rušení je proveden:

- **Okamžitě**, pokud cesta nebyla dostavěna. To znamená, že se vlastně povolující návěst nerozsvítí, tudíž může být cesta zrušena okamžitě.
- **Za 5 sekund**, pokud cesta byla uzavřena a vlak není v rozhodujícím úseku před návěstidlem.
- **Za 3 minuty**, pokud cesta byla uzavřena a vlak je v rozhodujícím úseku před návěstidlem. Je to vlastně čas, po kterou dobu se fyzicky drží cesta postavená, kdyby náhodou vlak, který byl v rozhodujícím úseku nestihl zastavit a vjel do rušící se cesty. Pokud se tak stane, cesta se nezruší.

V každém bloku, který náleží k nějakému návěstidlu, je jednoduchý čítač, který odpočítává dané časové úseky. Tento čítač má vstup, kterým se určuje délka časového úseku, který má být odměřen (informace z logického automatu) a výstup zda časový úsek uplynul. Tento čítač obsahuje generátor sekund, na základě kterého čítá. Blokové schéma je na následujícím obrázku:



Obr. 10 : Blokové schéma návěstidla s čítačem

Signál **Cítac_out** je 2 bitový a je to signál, pomocí kterého logický automat ovládá čítač. Signál nabývá následujících hodnot:

- „00“ – **nečítej** – žádný časový úsek není potřeba odměřit
- „01“ – **čítej 5 sekund** – čítač odpočítá 5 sekund a nastaví svůj výstup na log. 1
- „11“ – **čítej 3 minuty** – čítač odpočítá 3 minuty a nastaví svůj výstup na log. 1
- „10“ – rezerva – např. pro budoucí možné rozšíření pro rušení posunovacích cest

Signál **Cítac_in** je jednobitový vstup do logického automatu, který čítač nastaví na log. 1 poté, co načítá požadovaný časový úsek. Jinak je tento signál v log. 0.

5.1.5 Vybavování vlaku

Postup správného vybavování vlaku je popsán v analýze v části 3.1.3.8. Tento postup kontrolují bloky uvnitř postavené cesty. Z pohledu takového bloku se jedná o následující kontrolu průjezdu vlaku (cesta je postavena zleva doprava, opačný směr by byl analogický):

- 1) Můj úsek je volný, jinak nastává chyba.
- 2) Dostávám informaci o obsazení levého sousedního úseku.
- 3) Můj úsek je obsazen a zároveň zůstal obsazen sousední levý úsek.
- 4) Od této chvíle se sousední levý úsek může uvolnit.
- 5) Sousední pravý úsek je obsazen a zároveň zůstal obsazen můj úsek.
- 6) Od této chvíle se může můj úsek uvolnit, ale jen pokud je levý úsek již uvolněn.
- 7) Pravý sousední úsek se může uvolnit, ale jen pokud můj úsek je již uvolněn.

Nedodržetím těchto pravidel se blok dostává do chybového stavu. Pokud jsou tato pravidla dodržena, bloky se uvolní a lze je použít k dalšímu postavení cesty.

5.1.6 Návěstidla

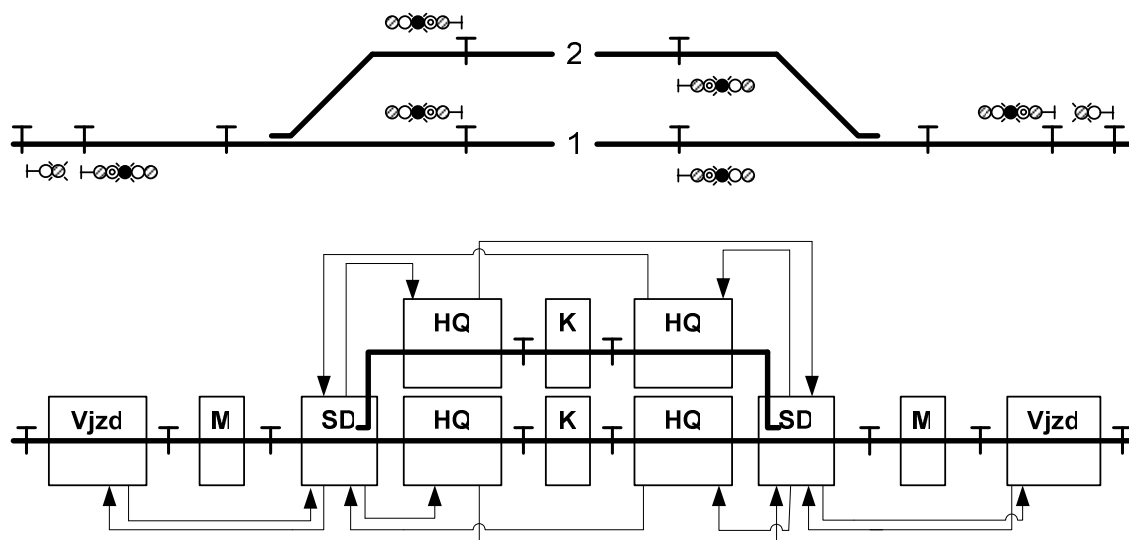
K ovládání každého návěstidla je přiřazen určitý blok. Na každém návěstidle může být zobrazeno více informací. V našem případě jednoduchého nádraží svítí na návěsti **zakazující návěst** (červená) nebo jsou zde zobrazeny následující dvě informace:

- **Aktuální rychlost v úseku**
- **Informace co zobrazuje následující návěst v cestě**

První možnost, **aktuální rychlost v úseku**, je v daném bloku 3 bitový výstup **V_HLN** a může nabývat následujících hodnot:

- 1) „000“ – červená
- 2) „001“ – oranžová
- 3) „010“ – 40 km/h
- 4) „011“ – 60 km/h
- 5) „100“ – 80 km/h
- 6) „101“ – 100 km/h

Informace co zobrazuje následující návěst v cestě (předzvěst) je v daném bloku 3 bitový výstup **V_HLN_pr** a nabývá stejných hodnot jako **V_HLN** pouze je rozdíl v hodnotě „000“, protože pokud je na následující návěsti červená, tak na této předzvěsti je „001“ - oranžová. Hodnotu „000“ tento signál nabývá, pokud je i na **V_HLN** „000“ (červená). Předzvěstní část návěstidla by tedy na hodnotu „000“ neměla nic zobrazovat, protože pokud je na naší návěsti červená, tak je informace co je na následujícím návěstidle zbytečná a mohla by mást.



Obr. 11 : Návrh blokového zapojení signálů pro návěsti

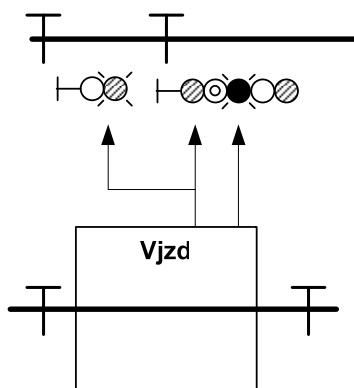
Na obrázku č.11 je znázorněn blokový návrh zapojení signálů pro návěsti. Každá šipka představuje dvojici signálů. Ve dvojici je vždy aktuální rychlost v tomto úseku (tu můžou

bloky upravovat, např. výhybky dle směru kam jsou nastaveny) a rychlost na předchozím návěstidle (tu bloky upravovat nemohou, tam je hodnota, která má svítit na návěstidle v části předzvěst). Bloky jsou propojeny tak, že do daného návěstního bloku jde informace z následujícího návěstního bloku ve stejném směru. Hodnoty se šíří i přes výhybkový blok SD, aby aktuální rychlost mohla být upravena v závislosti na směru výhybky.

5.2 Blok Vjzd

5.2.1 Popis bloku

Je to blok, který se umísťuje na rozhraní mezi stanicí a tratí. Na tomto místě je vždy vchodové návěstidlo a tento blok ho ovládá. Umožňuje tedy vjezd a odjezd vlaků z nebo do stanice. Část nádraží s tímto blokem je znázorněna na následujícím obrázku.



Obr. 12 : Blok Vjzd

Tento blok má na starosti hlavní vjezdové návěstidlo. Toto návěstidlo mívá svou vlastní předzvěst a bývá umístěno někde dále na trati, aby přijíždějící vlaky věděly s předstihem, co svítí na hlavním vjezdovém návěstidle. Pro tuto předzvěst není z bloku vyveden samotný výstup, protože je to zbytečné. Jedná se o jednoduchý dekodér nebo spíše způsob zapojení, které by bylo u předzvěsti a zajišťovalo by, že místo červené (ta na předzvěsti není) zobrazuje oranžovou, jinak rychlosti zobrazuje stejně.

V příloze najdete náčrt logického automatu, z kterého se blok **Vjzd** skládá. Náčrt je rozdělen do dvou částí. První část řeší směr do stanice. Druhá část řeší směr ze stanice. Základem jsou dva stavy obsazeno a volno, z těchto dvou stavů lze začít stavět cestu, kdežto být cílem cesty umožňuje blok, jen pokud je ve stavu volno. Blok obsahuje čítač pro rušení cesty a jednoduchou logiku pro návěst, která zajišťuje aby svítila červená, pokud není postavena cesta. Pokud je postavena, cesta svítí obě části hlavní návěsti (aktuální povolená rychlost a informace co je na dalším návěstidle v cestě). Rozhraní bloku je popsáno přímo v něm u VHDL kódu.

5.2.2 Požadavky na blok

- Kontrola obsazenosti příjezdového, odjezdového traťového úseku
- Umožňovat vjezd vlaků do stanice. (Stavění cesty a blok **vjzd** je počátkem)
- Umožňovat odjezd vlaků na traťovou kolej. (Stavění cesty, blok **Vjzd** je cílem)
- Rušení od tohoto bloku postavených cest.
- Kontrola znaků na Hlavní návěsti.
- Zobrazování správné informace o rychlosti.

5.3 Blok M

5.3.1 Popis bloku

Tento blok se umísťuje na bezvýhybkový úsek. Jeho základní náplní je kontrola obsazenosti úseku a kontrola správnosti vybavení vlaku. Pokud je úsek volný, umožňuje přes něj postavit cestu. Tento blok je stranově symetrický, ale aby nedošlo k záměně signálů, jsou signály pojmenovány L , R. V příloze naleznete náčrtek logického automatu, ze kterého se blok skládá. V tomto náčrsku je jen jeden směr, druhý je analogický.

5.3.2 Požadavky na blok

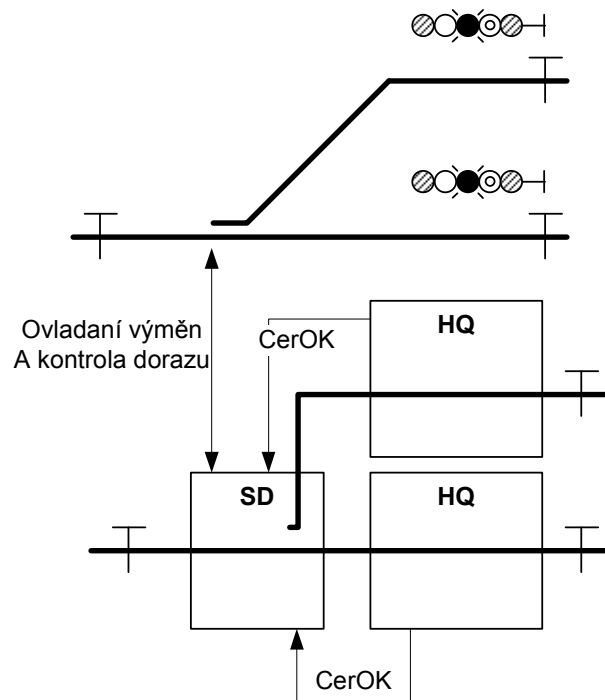
- Kontrola obsazenosti úseku.
- Umožňovat stavění cesty přes blok.
- Umožňovat rušení cesty přes blok.
- Kontrola správného vybavení vlaku.

5.4 Blok SD

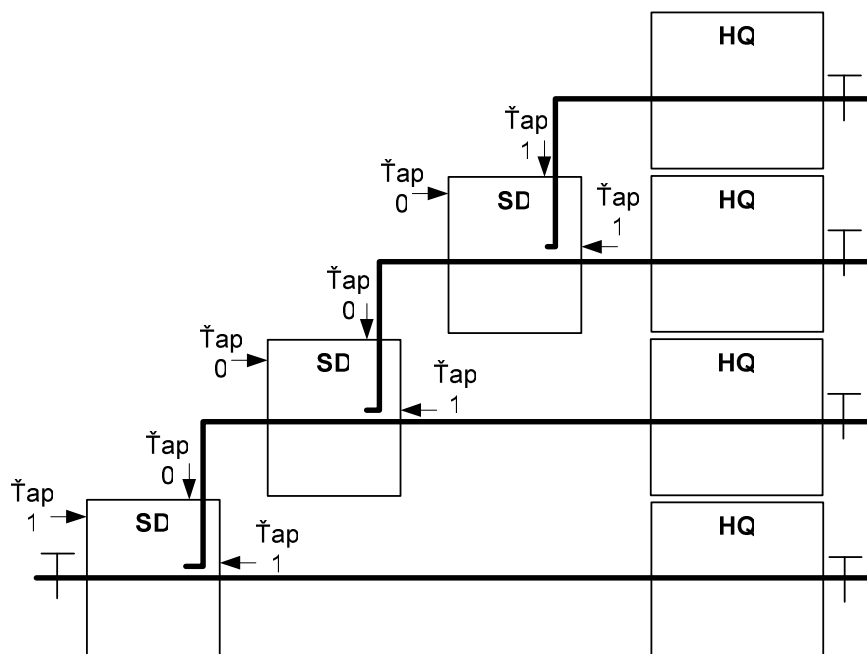
5.4.1 Popis bloku

Tento blok je určen k ovládní výhybky. Umožňuje přehození výhybky dle nalezené cesty, kontrolu správnosti dorazů výhybky a kontroluje řapání vlaku. Náčrtek logického automatu pro tento blok naleznete v příloze. Náčrtek má dvě části, každou pro jeden směr. Vždy je tam nakreslen jeden případ (např. když se staví cesta rovně), ten druhý je analogický (cesta se

staví do odbočky). Zapojení bloku do kolejiště je znázorněno na následujícím obrázku. Podmínkou pro postavení cesty v daném směru je, že v druhém směru je na návěsti v pořádku rozsvícená zakazující návěst.



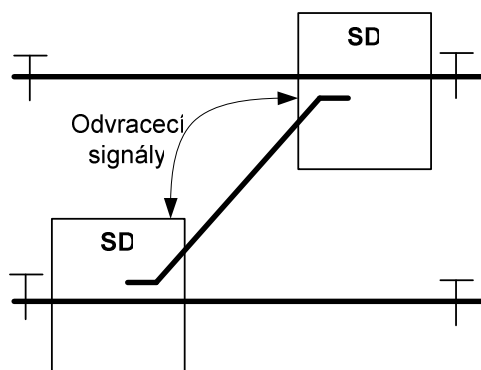
Obr. 13 : Zapojení bloku SD



Obr. 14 : Zapojení více bloků SD

Blok SD umožňuje zapojení více těchto bloků za sebou i v jednom úseku. Tedy že všechny tyto bloky jsou připojeny na stejný kolejový obvod (Mají stejnou informaci o obsazení). V takovém případě samozřejmě nemůže vnitřní blok kontrolovat vybavování vlaku. Toto je ošetřeno dalšími vstupy do bloku SD, kterými se říká, zda mají kontrolovat ťapání, neboli zda je jejich výstup už poslední v dané větvi. Toto je znázorněno na obrázku číslo 14. Pokud je na vstupu **Ťap** log. 1, tak se ťapání kontroluje, pokud je tam log. 0, tak blok ťapání nekontroluje.

Dalším požadavkem je, aby se některé výhybky navzájem ovlivňovaly. Je to takzvaná boční ochrana a spočívá v tom, že pokud se přes jednu výhybku postaví cesta, tak jiná výhybka, která jí má dělat ochranu, se odvrátí, aby nemohl omylem vjet vlak do postavené cesty z jiného směru než byla cesta postavena. Pokud mají dvě výhybky plnit funkci boční ochrany, propojí se signály k tomu určenými. Tyto signály jsou 4, dva vstupní a dva výstupní. Tento případ je nakreslen na následujícím obrázku. Pokud je tedy požadavek na postavení cesty rovně na spodní koleji, tak se blok SD u horní koleje odvrátí, tedy výhybka se přehodí také do směru rovně, aby nemohl vlak z horní koleje ohrozit vlak na spodní koleji.



Obr. 15 : SD boční ochrana

Dále jsou v bloku SD dva vstupy pro definování rychlosti do daného směru. Blok může být použit k různým typům vyhybek, které jsou konstruovány na různé rychlosti. Těmito vstupy se tedy nastavuje, kolik může jet vlak v daném směru. Jsou to vstupy **rychl_r** (směr rovně) a **rychl_odb** (směr do odbočky). Tyto hodnoty ovlivňují rychlosti zobrazované na návěsti, podle toho kudy je postavena cesta.

5.4.2 Požadavky na blok

- Kontrola obsazenosti úseku
- Ovládání výhybky

- Kontrola dorazů výhybky
- Možnost hledat cestu v obou větvích najednou
- Možnost stavění a rušení cesty
- Kontrola červené ve větví než se staví cesta
- Odvracení výměn určených k boční ochraně
- Kontrola t'apání
- Možnost nastavení rychlosti do každého směru.

5.5 Blok HQ

5.5.1 Popis bloku

Tento blok je přiřazen k hlavnímu odjezdovému návěstidlu. Tento blok nekontroluje volnost úseku, protože to má na starosti jiný blok, a to většinou SD. Tento blok umožňuje postavení a rušení cest a umožňuje postavit k němu cestu v daném směru co je návěstidlo. Jeho logický automat naleznete v příloze. Skládá se ze dvou částí, pro každý směr jedna. Základ je stejný jako u bloku **Vjzd**, pouze blok nekontroluje obsazenost úseku a navíc umožňuje hledat cestu dál ve směru, v kterém návěstidlo není. Pokud se k němu dostane požadavek na hledání cesty ve směru v jakém je návěstidlo, požadavek u něj končí, ať už je blok cílem nebo ne. Pokud blok je cílem, generuje token **RESERVED**, a pokud cílem není, tak nereaguje a token **INFORM** se začne stahovat zpět.

5.5.2 Požadavky na blok

- Umožňovat vjezd vlaků do stanice.(Stavění cesty a blok **HQ** je cílem)
- Umožňovat odjezd vlaků na traťovou kolej. (Stavění cesty, blok **HQ** je počátkem)
- Rušení od tohoto bloku postavených cest.
- Kontrola znaků na Hlavní návěsti.
- Zobrazování správné informace o rychlosti.

5.6 Blok K

5.6.1 Popis bloku

Tento blok se přiřazuje ke staniční koleji. Jeho základní náplní je kontrola obsazenosti úseku a kontrola správnosti vybavování vlaku. Pokud je úsek volný, umožňuje přes něj postavit cestu. Oproti bloku **M** umožňuje tento blok svým zapojením, že vlak na tomto úseku může skončit a odjet na druhou stranu. Tento blok je stranově symetrický, ale aby nedošlo k záměně signálů, jsou signály pojmenovány *L* , *R*. V příloze naleznete nákres logického automatu, ze kterého se blok skládá. V tomto nákresu je jen jeden směr, druhý je analogický.

5.6.2 Požadavky na blok

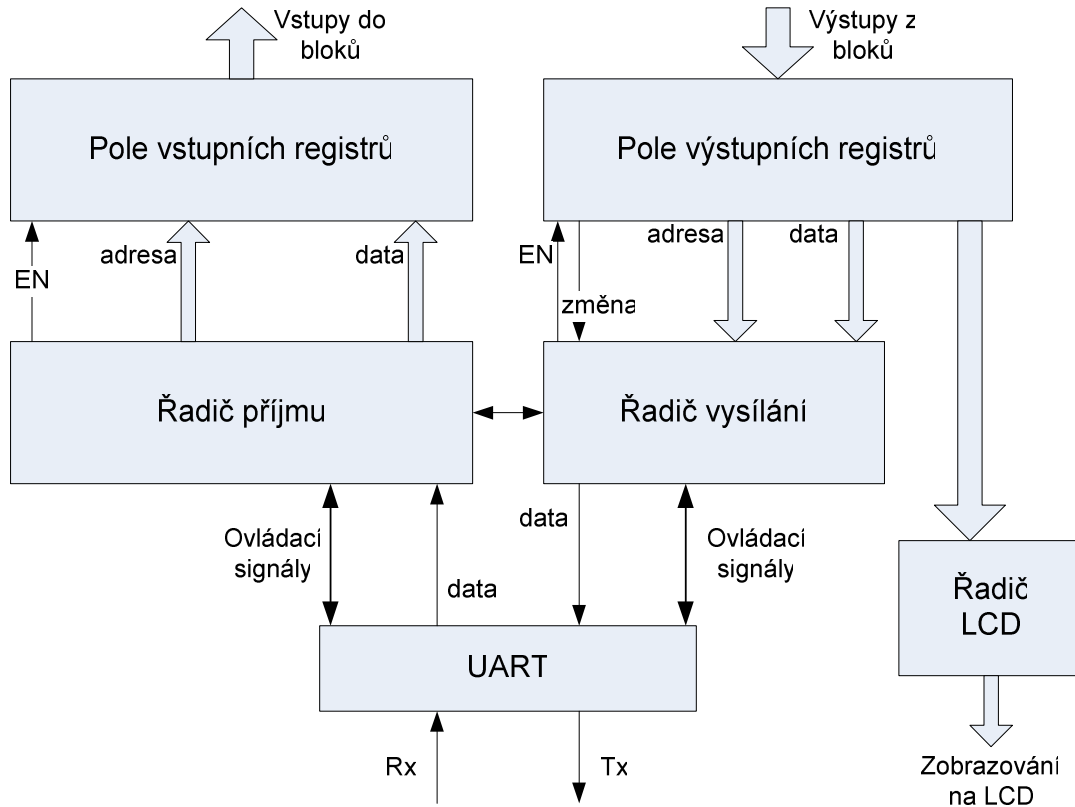
- Kontrola obsazenosti úseku.
- Umožňovat stavění cesty přes blok.
- Umožňovat rušení cesty přes blok.
- Kontrola správného vybavení vlaku.

5.7 Implementace jednoduchého nádraží

V projektu **ISE** v příloze naleznete VHDL kódy jednotlivých bloků. Ve schématu **nadrazi.sch** je zapojeno jednoduché nádraží z obrázku č. 7. Toto nádraží je postaveno z vytvořených bloků v rámci této diplomové práce.

5.7.1 Testovací aplikace

Pro jednoduché nádraží byla vyvinuta testovací aplikace, aby se zařízení mohlo otestovat v přípravku se SPARTAN 3E. Aplikace obsahuje pole registrů, které se dají adresovat. Tímto můžeme zapisovat hodnoty přímo do vstupních registrů, pro který blok chceme. Případně nám přijde na sériovou linku adresa a data a my podle adresy poznáme, ke kterému bloku a jakému jeho výstupu hodnota odpovídá. Aplikace je navržena tak, že posílá jen hodnoty které se změnily. Schematický nákres si můžete prohlédnout na následujícím obrázku:



Obr. 16 : Schéma testovací aplikace

- **Pole vstupních registrů**

Jedná se o pole registrů, které jsou připojeny na datovou a adresovou sběrnici. Pokud řadič příjmu vystaví na tyto sběrnice adresu a data a povolí zápis EN, zapíše se hodnota do registru s danou adresou. Výstupy registrů jsou už přímo připojeny k blokům.

- **Pole výstupních registrů**

Registry v tomto poli mají připojeny vstup přímo na výstupy z bloků. Do těchto registrů se hodnota generovaná z bloků zapisuje okamžitě. Každý tento registr má ještě jeden svůj „stínový registr“, kam se hodnota zapisuje až na povolení od řadiče. Mezi těmito dvěma registry je komparátor, který řadiči sděluje, že došlo ke změně.

- **Řadič příjmu**

Řadič příjmu čeká na příjem dat od UARTU. Jakmile má k dispozici adresu a data, tak je vystaví na sběrnici a povolí zápis do registru. Také umožňuje s řadičem vysílání „test živosti“,

což znamená, že pokud je přijata speciální sekvence, tak řadič příjmu uvědomí řadič vysílání a ten na ni odpoví odpovídajícím způsobem.

- **Řadič vysílání**

Tento řadič čeká na signál **ZMĚNA**, který mu říká, že na některém z výstupů došlo ke změně. Na tento signál reaguje tak, že odešle přes sériovou linku adresu a data, která se změnila a povolí signálem **EN** zápis do „stínového“ registru. Tímto se v obou registrech ocitne stejná hodnota a komparátor pro dané dva registry přestane dávat signál **ZMĚNA**.

- **Řadič LCD**

V tomto bloku je řadič pro LCD. Je zde startovací sekvence pro správné rozchození LCD na přípravku a jsou v něm napevno dány pozice, kde se jaký výstup z bloků má na LCD zobrazovat. Na LCD je názorně vidět v jakém stavu jsou bloky, které cesty jsou uzavřené, kde se nachází vlak, či jaké rychlosti zobrazují návěsti.

- **UART**

Tento blok nebyl vytvořen v rámci této diplomové práce. Byl stáhnut z www.opencores.org. Bylo mu jen mírně upraveno rozhraní, aby byl použitelný pro mou aplikaci.

5.7.1.1 Protokol testovací aplikace

Protokol je vytvořen obecně s dostatečnou rezervou pro rozšíření aplikace. Pro testovací účely nám postačí jen základní funkce:

- Po sériové lince můžeme ovládat vstupy bloků
- Po sériové lince nám přichází změny na výstupech bloků
- Můžeme otestovat „živost přípravku“

Všechna komunikace po sériové lince probíhá po 3 bajtech:

- 1) příkaz
- 2) adresa
- 3) data

Takže pokud chceme nastavit nějaký vstupní registr, musíme znát jeho adresu. Seznam adresovatelných registrů a jejich popis bitů pro jednoduché nádraží naleznete v příloze. Příkazy, které jsou nutné pro testovatelnost v přípravku a byly naimplementovány:

- **A** (ascii hodnota) - příkaz pro zápis do registru (směr z UARTU do FPGA), nebo pokud jsme přijali trojici bajtů s příkazem A (směr z FPGA do UARTU), tak se hodnota ve výstupním registru na dané adrese změnila na přijatá data.
- **X** (ascii hodnota) – test živosti přípravku na adrese a datech nezáleží, může se odeslat třeba XYZ. Přípravek vždy odpoví XXX.

5.7.2 Využití prostředků

Blok	FF [%]	LUT[%]	Slices[%]	IOB[%]
Vjzd	1	1	2	18
M	1	1	1	10
SD	1	2	2	36
HQ	1	2	2	22
K	1	1	1	9
Jednoduché nádraží s testovací aplikací	9	30	33	5

Tab. 1 Využití prostředků FPGA

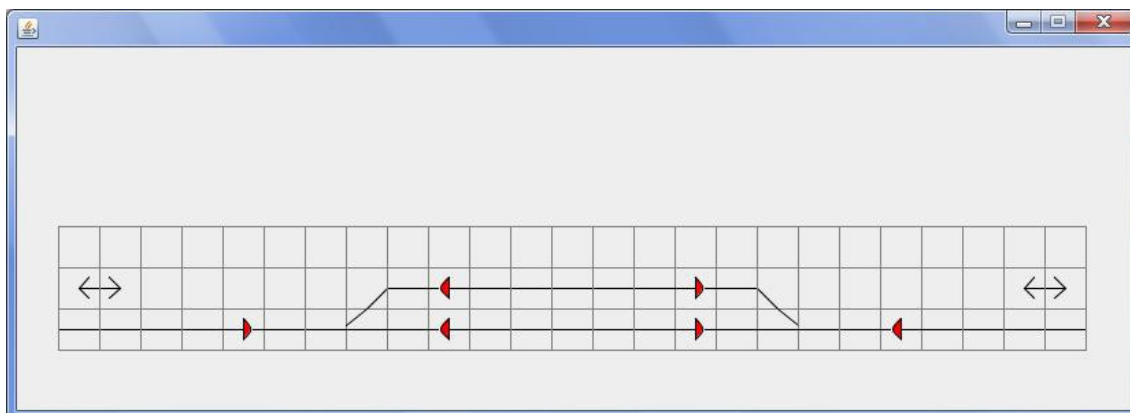
Využití prostředků v tabulce je popsáno v procentech. Bylo měřeno na FPGA SPARTAN 3E, ten disponuje následujícím počtem jednotlivých parametrů:

SPARTAN 3E	
FF	9 312
LUT	9 312
SLICES	4 656
IOBs	232

Tab. 2 Prostředky SPARTAN 3E

Z předcházejících tabulek je vidět, že jednoduché nádraží se do tohoto FPGA bezproblémově vešlo. Zhruba jednu čtvrtinu z využitých prostředků zabírá testovací aplikace. Do FPGA by tedy šlo naimplementovat větší nádraží, případně u hodně velkého nádraží by šly logické automaty rozdělit do více FPGA.

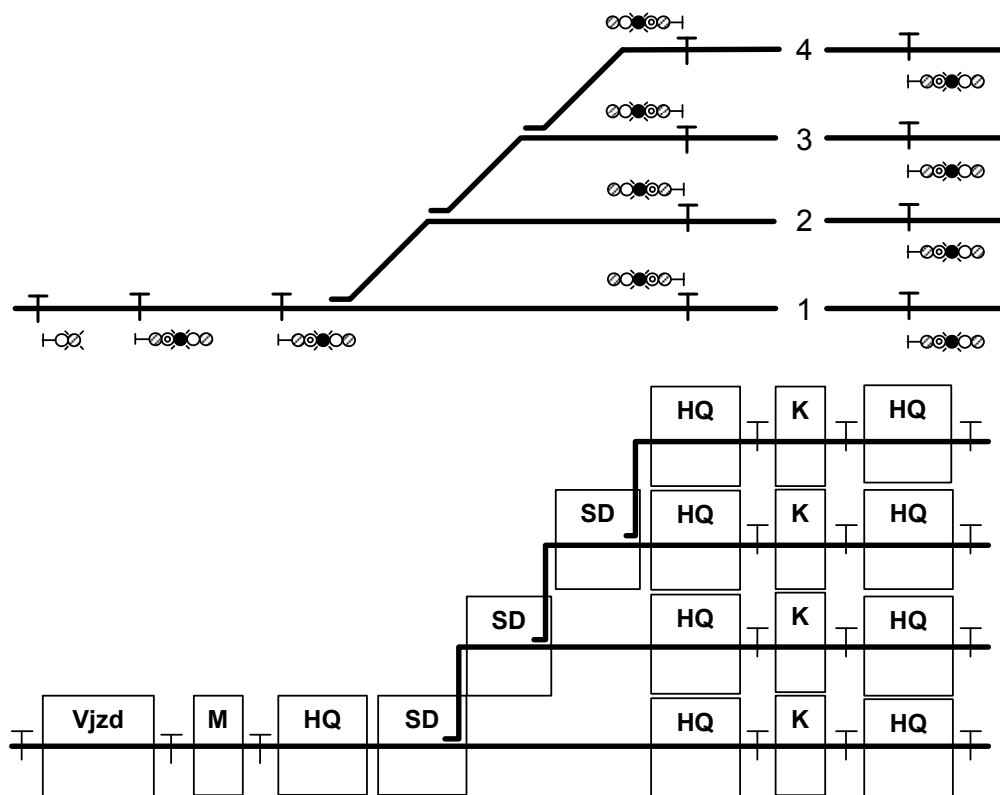
Na následujícím obrázku je náhled ovládací aplikace vytvořené v rámci jiné diplomové práce s kterou má práce komunikuje po sériové lince.



Obr. 17 : Náhled na ovládací pult

5.8 Složitější nádraží

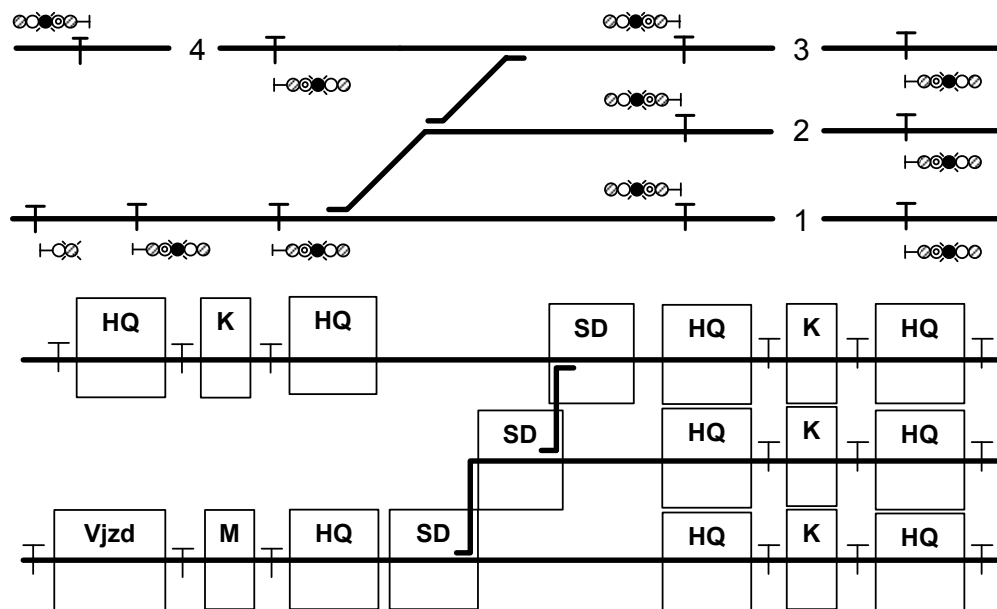
V projektu byly vytvořeny i dvě části složitějšího nádraží, na kterých byla testována jiná zapojení bloků než je v jednoduchém nádraží. Tato zapojení jsou znázorněna na následujících obrázcích:



Obr. 18 : Složitější nádraží 1-více staničních kolejí

Na obrázku číslo 18 je znázorněna část složitějšího nádraží s příjezdem z jedné traťové koleje a čtyřmi staničními kolejemi. Na tomto nádraží bylo vyvinuto a odzkoušeno správné chování bloků SD, pokud jich je více za sebou a jsou na jednom vybavovacím úseku. Zbytek nádraží pokračující vpravo není nakreslen, protože pro odzkoušení funkčnosti není potřeba.

Na obrázku číslo 19 je nádraží, kde byly zapojeny výhybky jako boční ochrana. Bylo zde tedy odzkoušeno správné odvracení vyhybek. Zbytek nádraží není opět znázorněn, protože pro funkci není potřeba.



Obr. 19 : Složitější nádraží 2-odvrat výhybky

6 Testování

6.1 Úvod

V této kapitole je popsán postup a způsob testování zařízení, které bylo provedeno. Testování je rozděleno do několika bodů tak, jak bylo postupně prováděno.

6.2 Testování jednotlivých bloků

Každý blok, který byl vyvinut, byl postupně už při vývoji podrobován testování a to zejména v ModelSimu, kde byla kontrolována jeho správná funkčnost a správnost výstupů. Bloky byly nejdříve jednotlivě zkontrolovány a poté testovány jako černé krabičky. Postupně jsem bloky spojoval, sledoval a kontroloval komunikaci mezi bloky. Na základě zjištěných nesrovnalostí a chyb byly bloky nesčetněkrát upravovány až do stávající podoby, kdy plní požadavky na jednotlivé bloky a jako celek plní požadavky na zařízení uvedené v kapitole 3.1.3.2.

6.3 Testování na jednoduchém nádraží

Pro účely testování bylo implementováno jednoduché testovací nádraží, které je znázorněno na obrázku č.7. Toto nádraží bylo testováno v ModelSimu a poté v přípravku za pomoci vyvinuté testovací aplikace. Požadavky, které byly kontrolovány, jsou v následující tabulce. Jsou tam uvedeny Requirements, kdy s lichým číslem je vždy kontrola v ModelSimu a se sudým číslem kontrola na přípravku.

Modelsim	Přípravek	Slovní popis požadavku
Req.1	Req.2	Po uvedení do provozu přechod do bezpečného stavu a dále do stavu dle vstupů.
Req.3	Req.4	Postavení vlakové cesty z levé příjezdové cesty na 1. staniční kolej.
Req.5	Req.6	Postavení vlakové cesty z levé příjezdové cesty na 2. staniční kolej.
Req.7	Req.8	Postavení vlakové cesty z pravé příjezdové cesty na 1. staniční kolej.
Req.9	Req.10	Postavení vlakové cesty z pravé příjezdové cesty na 2. staniční kolej.

Req.11	Req.12	Vjezd vlaku a jeho vybavování z levé příjezdové cesty na 1. staniční kolej.
Req.13	Req.14	Vjezd vlaku a jeho vybavování z levé příjezdové cesty na 2. staniční kolej.
Req.15	Req.16	Vjezd vlaku a jeho vybavování z pravé příjezdové cesty na 1. staniční kolej.
Req.17	Req.18	Vjezd vlaku a jeho vybavování z pravé příjezdové cesty na 2. staniční kolej.
Req.19	Req.20	Postavení vlakové cesty z 1. staniční koleje na levou traťovou kolej.
Req.21	Req.22	Postavení vlakové cesty z 2. staniční koleje na levou traťovou kolej.
Req.23	Req.24	Postavení vlakové cesty z 1. staniční koleje na pravou traťovou kolej.
Req.25	Req.26	Postavení vlakové cesty z 2. staniční koleje na pravou traťovou kolej.
Req.27	Req.28	Odjezd vlaku a jeho vybavování z 1. staniční koleje na levou traťovou kolej.
Req.29	Req.30	Odjezd vlaku a jeho vybavování z 2. staniční koleje na levou traťovou kolej.
Req.31	Req.32	Odjezd vlaku a jeho vybavování z 1. staniční koleje na pravou traťovou kolej.
Req.33	Req.34	Odjezd vlaku a jeho vybavování z 2. staniční koleje na pravou traťovou kolej.
Req.35	Req.36	Rušení postavené cesty z levé traťové koleje na 1. staniční, pokud v rozhodujícím úseku není vlak.
Req.37	Req.38	Rušení postavené cesty z levé traťové koleje na 2. staniční, pokud v rozhodujícím úseku není vlak.
Req.39	Req.40	Rušení postavené cesty z pravé traťové koleje na 1. staniční, pokud v rozhodujícím úseku není vlak.
Req.41	Req.42	Rušení postavené cesty z pravé traťové koleje na 2. staniční, pokud v rozhodujícím úseku není vlak.
Req.43	Req.44	Rušení postavené cesty z levé traťové koleje na 1. staniční, pokud v rozhodujícím úseku je vlak.
Req.45	Req.46	Rušení postavené cesty z levé traťové koleje na 2. staniční, pokud v rozhodujícím úseku je vlak.
Req.47	Req.48	Rušení postavené cesty z pravé traťové koleje na 1. staniční, pokud v rozhodujícím úseku je vlak.

Req.49	Req.50	Rušení postavené cesty z pravé traťové koleje na 2. staniční, pokud v rozhodujícím úseku je vlak.
Req.51	Req.52	Rušení postavené cesty z levé traťové koleje na 1. staniční, pokud v rozhodujícím úseku je vlak a vjede do rušené cesty.
Req.53	Req.54	Rušení postavené cesty z levé traťové koleje na 2. staniční, pokud v rozhodujícím úseku je vlak a vjede do rušené cesty.
Req.55	Req.56	Rušení postavené cesty z pravé traťové koleje na 1. staniční, pokud v rozhodujícím úseku je vlak a vjede do rušené cesty.
Req.57	Req.58	Rušení postavené cesty z pravé traťové koleje na 2. staniční, pokud v rozhodujícím úseku je vlak a vjede do rušené cesty.
Req.59	Req.60	Rušení postavené cesty z 1. staniční koleje na levou traťovou kolej, pokud v rozhodujícím úseku není vlak.
Req.61	Req.62	Rušení postavené cesty z 2. staniční koleje na levou traťovou kolej, pokud v rozhodujícím úseku není vlak.
Req.63	Req.64	Rušení postavené cesty z 1. staniční koleje na pravou traťovou kolej, pokud v rozhodujícím úseku není vlak.
Req.65	Req.66	Rušení postavené cesty z 2. staniční koleje na pravou traťovou kolej, pokud v rozhodujícím úseku není vlak.
Req.67	Req.68	Rušení postavené cesty z 1. staniční koleje na levou traťovou kolej, pokud v rozhodujícím úseku je vlak.
Req.69	Req.70	Rušení postavené cesty z 2. staniční koleje na levou traťovou kolej, pokud v rozhodujícím úseku je vlak.
Req.71	Req.72	Rušení postavené cesty z 1. staniční koleje na pravou traťovou kolej, pokud v rozhodujícím úseku je vlak.
Req.73	Req.74	Rušení postavené cesty z 2. staniční koleje na pravou traťovou kolej, pokud v rozhodujícím úseku je vlak.
Req.75	Req.76	Rušení postavené cesty z 1. staniční koleje na levou traťovou kolej, pokud v rozhodujícím úseku je vlak a vjede do rušené cesty.
Req.77	Req.78	Rušení postavené cesty z 2. staniční koleje na levou traťovou kolej, pokud v rozhodujícím úseku je vlak a vjede do rušené cesty.
Req.79	Req.80	Rušení postavené cesty z 1. staniční koleje na pravou traťovou kolej, pokud v rozhodujícím úseku je vlak a vjede do rušené cesty.
Req.81	Req.82	Rušení postavené cesty z 1. staniční koleje na pravou traťovou kolej, pokud v rozhodujícím úseku je vlak a vjede do rušené cesty.

Tab. 3 Testované požadavky

Při každém požadavku byly kontrolovány výstupy generované jednotlivými bloky. Zvláště signály k výhybkám, zobrazované znaky na návěstidlech, jejich zobrazení ve správný okamžik atd.

6.4 Testování na složitějším nádraží

Testování proběhlo i na složitějších částech nádražích, které jsou uvedeny na obrázcích číslo 17 a 18. V prvním případě bylo testováno hlavně správné vybavování vlaku a ve druhém případě správné odvracení vyhybek.

6.5 Výsledky

Požadavky	Výsledek
Req.1, Req.2	V pořádku
Req.3, Req.4	V pořádku
Req.5, Req.6	V pořádku
Req.7, Req.8	V pořádku
Req.9, Req.10	V pořádku
Req.11, Req.12	V pořádku
Req.13, Req.14	V pořádku
Req.15, Req.16	V pořádku
Req.17, Req.18	V pořádku
Req.19, Req.20	V pořádku
Req.21, Req.22	V pořádku
Req.23, Req.24	V pořádku
Req.25, Req.26	V pořádku
Req.27, Req.28	V pořádku
Req.29, Req.30	V pořádku
Req.31, Req.32	V pořádku
Req.33, Req.34	V pořádku
Req.35, Req.36	V pořádku
Req.37, Req.38	V pořádku
Req.39, Req.40	V pořádku
Req.41, Req.42	V pořádku
Req.43, Req.44	V pořádku

Req.45, Req.46	V pořádku
Req.47, Req.48	V pořádku
Req.49, Req.50	V pořádku
Req.51, Req.52	V pořádku
Req.53, Req.54	V pořádku
Req.55, Req.56	V pořádku
Req.57, Req.58	V pořádku
Req.59, Req.60	V pořádku
Req.61, Req.62	V pořádku
Req.63, Req.64	V pořádku
Req.65, Req.66	V pořádku
Req.67, Req.68	V pořádku
Req.69, Req.70	V pořádku
Req.71, Req.72	V pořádku
Req.73, Req.74	V pořádku
Req.75, Req.76	V pořádku
Req.77, Req.78	V pořádku
Req.79, Req.80	V pořádku
Req.81, Req.82	V pořádku

Tab. 4 Výsledky testovaných požadavků

6.6 Závěr

Výsledky testovaných požadavků jsou shrnuty v předchozí tabulce. Všechny testy jednotlivých bloků a testy na jednoduchém nádraží proběhly úspěšně. Testy na složitějších nádražích proběhly taktéž úspěšně. I přesto, že problematika a možnosti kombinací na vstupech je příliš mnoho a nelze je všechny v přijatelném čase otestovat, se dá říci, že zařízení je funkční, protože všechny testy provedené na jednoduchém i složitějších nádražích dopadly úspěšně a zařízení se chová dle požadavků, které na něj byly kladeny jako na celek. Na závěr proběhlo i úspěšné testování s ovládacím pultem vytvořeným v rámci jiné diplomové práce.

7 Uživatelská příručka

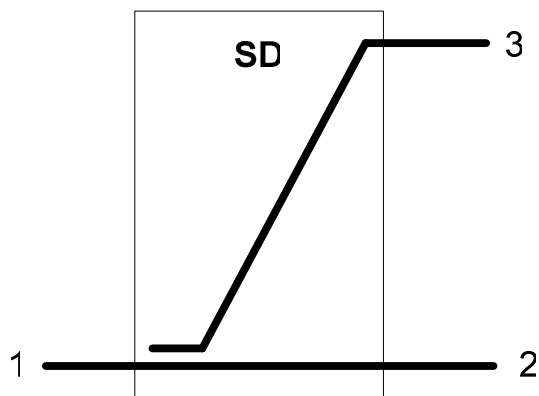
Na základě vyvinutých bloků lze postavit staniční zabezpečovací zařízení pro jakoukoli stanici složenou ze stejných prvků v kolejišti, jaké jsou v jednoduchém nádraží uvedeném na obrázku č. 7. Jedná se o následující bloky a jim odpovídající prvky:

- **Vjzd** – Vjezdové návěstidlo
- **M** – Bezvýhybkový úsek
- **SD** – Jednoduchá výhybka
- **HQ** – Odjezdové návěstidlo
- **K** – Staniční kolej

7.1 Popis zapojení jednotlivých bloků a jejich propojování

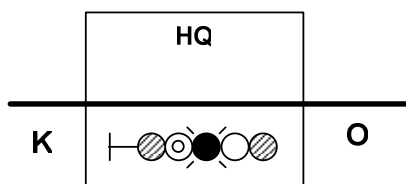
Bloky **Vjzd**, **SD**, **HQ** nejsou stranově symetrické a musíme při jejich zapojování dbát na jejich správnou orientaci:

- **Vjzd** sice není symetrický, ale v podstatě nejde otočit, protože je to koncový blok a nemá žádné dva stejné signály, které by se daly zaměnit.
- **SD** není symetrický. Na jednu stranu jsou signály pro jednu kolej a na druhou by měly být signály pro dvě koleje. Ve VHDL kódu mají vstupy a výstupy vždy označení **1,2,3** a je vhodné ho zachovat podle následujícího obrázku:



Obr. 20 : Číslování vývodů bloku SD

- **HQ** není symetrický. Jeho signály, které jsou vyvedeny do obou směrů, mají označení **K** a **O**. Blok je vytvořen tak, že návěstidlo je pro směr od **K** k **O** dle následujícího obrázku:



Obr. 21 : Označení vývodů HQ

Nejjednodušší způsob jak bloky zapojit je pomocí tohoto postupu:

- 1) Naskládám si bloky dle prvků v kolejišti.
- 2) Propojím struny. V postateř se stačí držet pravidla, že se vždy propojují signály se stejným jménem (až na označení směrů, nebo číslování u výhybek) a samozřejmě propojované signály jsou opačného typu (IN připojuji na OUT, OUT připojuji na IN). Takto se propojí vždy sousední bloky a vznikne základ nádraží.
- 3) Propojí se signály pro šíření znaků na návěstech **p_akt** a **p_pr**, postupuje se stejným způsobem jako v předcházejícím bodě, pouze se nemusí jednat o sousední bloky. Některé bloky totiž do těchto signálů nezasahují. Vždy ale propojujeme dva nejbližší signály v daném směru. Způsob zapojení u jednoduchého nádraží je naznačen na obrázku číslo 11.
- 4) U bloků SD se nastaví rychlosti do jednotlivých směrů pomocí vstupů **rychl_r** a **rychl_odb**. (rychlosti jsou stejně kódované jako návěstidla viz, vysvětlení hodnot výstupu **HLN** v příloze.
- 5) U bloků SD se nastaví výstupy **t'apani**. Viz odstavec 5.4.1. popis obrázku 14.
- 6) Propojí se signály pro odvracení výhybek, které mají fungovat jako boční ochrana. Jedná se o signály: **odvrat_in**, **odvraceno_in**, **odvrat_out** a **odvraceno_out**. Pokud v daném směru výhybky není boční ochrana, tak se pro daný směr na vstup **odvrat_in** přivede log. 0 a na vstup **odvraceno_in** log. 1.
- 7) Tímto je zařízení zapojené a plně funkční. Už jenom záleží jakým způsobem se mu budou budít signály na vstupech. Pokud by se chtěla použít vyvinutá testovací aplikace pro složitější nádraží, musí se do vstupních a výstupních registrových polí přidat vždy odpovídající adresovatelné registry.

8 Zhodnocení

Cílem této diplomové práce bylo vytvořit staniční zabezpečovací zařízení pro jednoduché nádraží. Vytvořit testovací aplikaci a odzkoušet zařízení v přípravku. Dle možností pak sestavit složitější nádraží. Těchto cílů bylo dosaženo za použití logických automatů vytvořených ve VHDL kódu pomocí metody tří procesů. Zařízení splňuje požadavky uvedené v kapitole 3.1.3.2 a jako celek se tedy na jednoduchém nádraží, na kterém probíhalo testování, chová shodně jako reléové staniční zabezpečovací zařízení. Univerzálnost bloků byla demonstrována sestavením částí složitějšího nádraží. Jednoduché nádraží bylo implementováno v FPGA SPARTAN 3E a odzkoušeno s ovládacím pultem z jiné diplomové práce.

Možnost rozšíření této práce by mohlo směřovat k doděláním bloků pro prvky, které nebyly součástí našeho jednoduchého nádraží, jako třeba implementace posunovacích cest (návesti pro ně) a dále například křížová výhybka. Nyní zařízení funguje dle reléového staničního zařízení tak, že cesty se musí zadávat postupně od jednoho návěstidla k dalšímu. Jednoduchou úpravou bloku HQ by šlo docílit automatického hledání cest tak, aby se dlouhé cesty nemusely rozkládat na jednotlivé úseky mezi návěstidly a stavět postupně.

9 Závěr

Výsledkem této diplomové práce jsou univerzální bloky ve VHDL, ze kterých lze postavit staniční zabezpečovací zařízení jako alternativní náhradu za reléové staniční zabezpečovací zařízení. Toto zařízení umožňuje bezpečné stavění nebo rušení vlakových cest a kontrolu správného vybavování vlaku. Generuje ovládací signály pro výhybky a hlavní vjezdová a odjezdová návěstidla. Umožňuje způsob boční ochrany pomocí odvracení výhybek. Dále v této práci byla vytvořena testovací aplikace s protokolem přes sériovou linku pro zadané jednoduché nádraží, na kterém proběhlo testování. Bloky byly otestovány i na složitějších částech nádraží, kde bez problémů plnily požadavky na svoji činnost. Jednoduché nádraží bylo naimplementováno do FPGA a odzkoušeno s ovládacím pultem z jiné diplomové práce.

10 Literatura a použité vývojové programy

[1] Ing. Martin Leso, Ph.D. – Disertační práce

[2] Dušan Križan, Zabezpečovací technika I a II, nakladatelství NADAS

[3] Komponenty ve VHDL www.opencores.org

11 Seznam obrázků

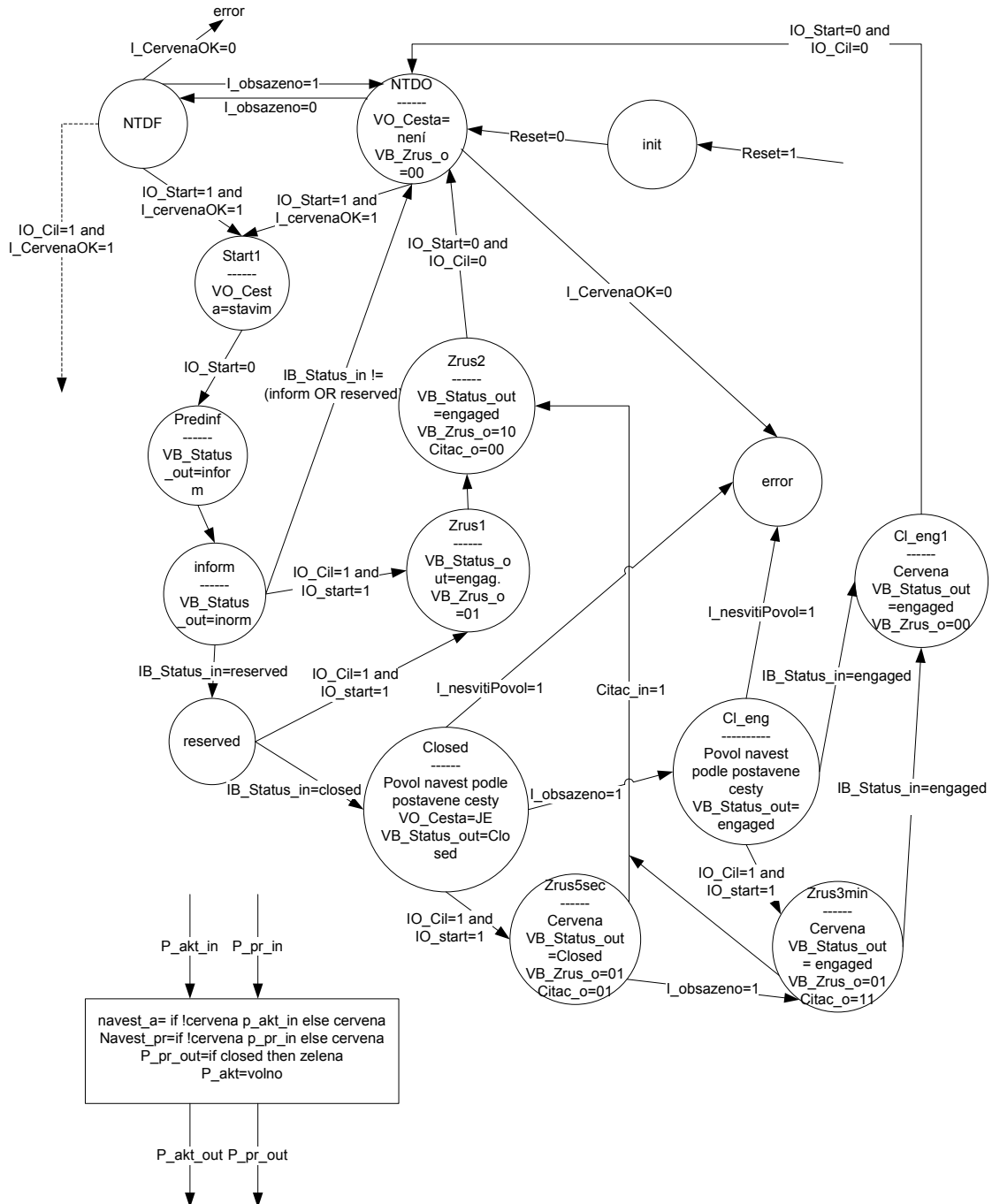
Obr. 1 : Obecné železniční staniční zabezpečovací zařízení	3
Obr. 2 : Model zapojení navrženého systému.....	5
Obr. 3 : Základní nádraží s reléovými bloky	8
Obr. 4 : Vybavování vlaku.....	10
Obr. 5 : Jedno FPGA jeden blok reléového staničního zařízení	13
Obr. 6 : Jedno FPGA více bloků reléového staničního zařízení	14
Obr. 7 : Jednoduché nádraží s navrženými bloky	15
Obr. 8 : Návrh zapojení testovací aplikace	17
Obr. 9 : Vybrané části šíření signálů při stavění cesty	21
Obr. 10 : Blokové schéma návěstidla s čítačem.....	22
Obr. 11 : Nákres blokového zapojení signálů pro návěsti.....	24
Obr. 12 : Blok Vjzd	25
Obr. 13 : Zapojení bloku SD.....	27
Obr. 14 : Zapojení více bloků SD	27
Obr. 15 : SD boční ochrana.....	28
Obr. 16 : Schéma testovací aplikace	31
Obr. 17 : Náhled na ovládací pult	34
Obr. 18 : Složitější nádraží 1-více staničních kolejí.....	34
Obr. 19 : Složitější nádraží 2-odvrat výhybky	35
Obr. 20 : Číslování vývodů bloku SD	41
Obr. 21 : Označení vývodů HQ.....	42
Obr. 22 : Automat VJZD – část 1.....	48
Obr. 23 : Automat VJZD – část 2.....	49
Obr. 24 : Automat M	50
Obr. 25 : Automat K.....	51
Obr. 26 : Automat SD – část 1	52
Obr. 27 : Automat SD – část 2.....	53
Obr. 28 : Automat HQ – část 1	54
Obr. 29 : Automat HQ – část 2	55

12 Seznam Tabulek

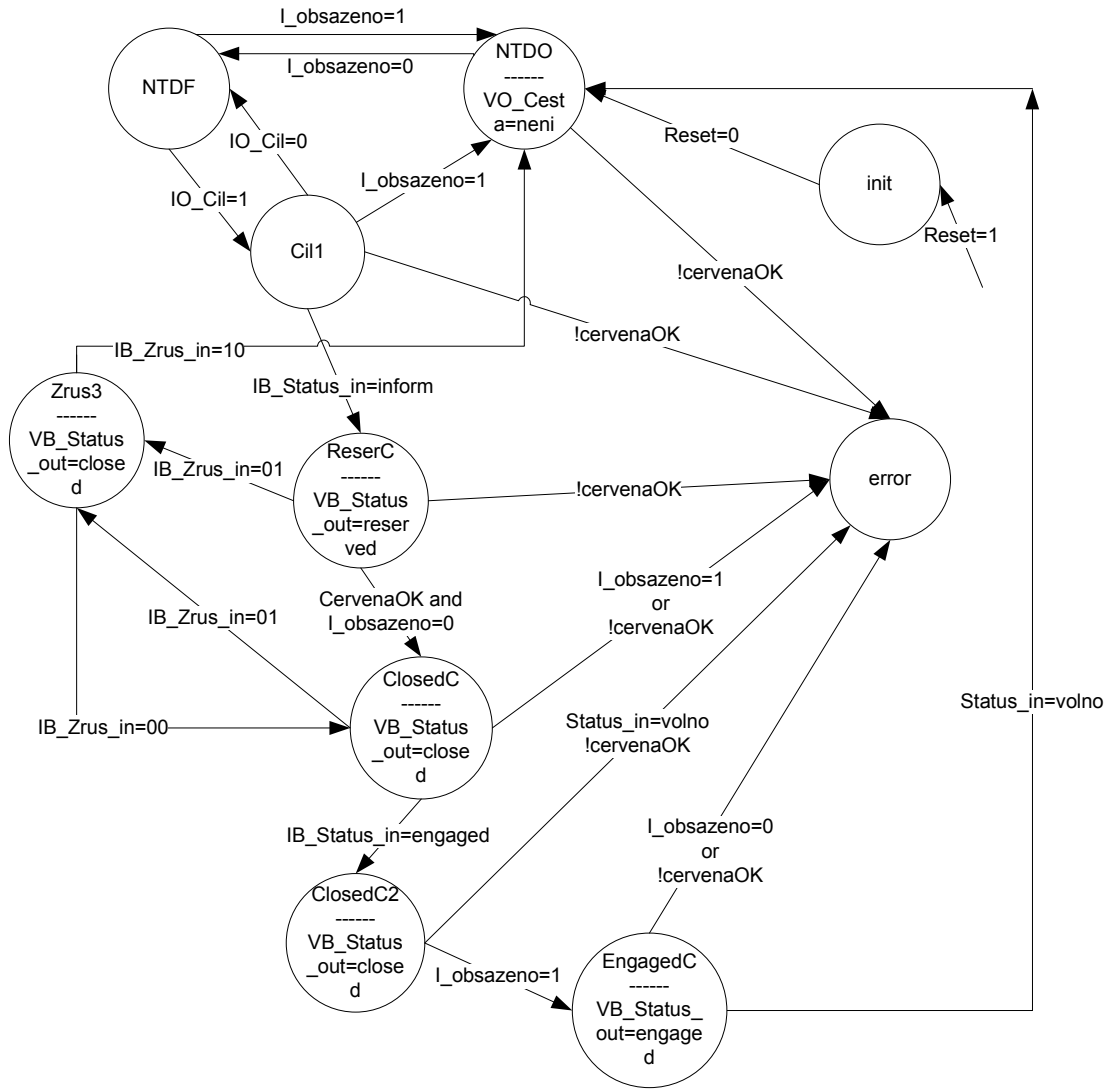
Tab. 1	Využití prostředků FPGA.....	33
Tab. 2	Prostředky SPARTAN 3E.....	33
Tab. 3	Testované požadavky	38
Tab. 4	Výsledky testovaných požadavků.....	40

13 Přílohy

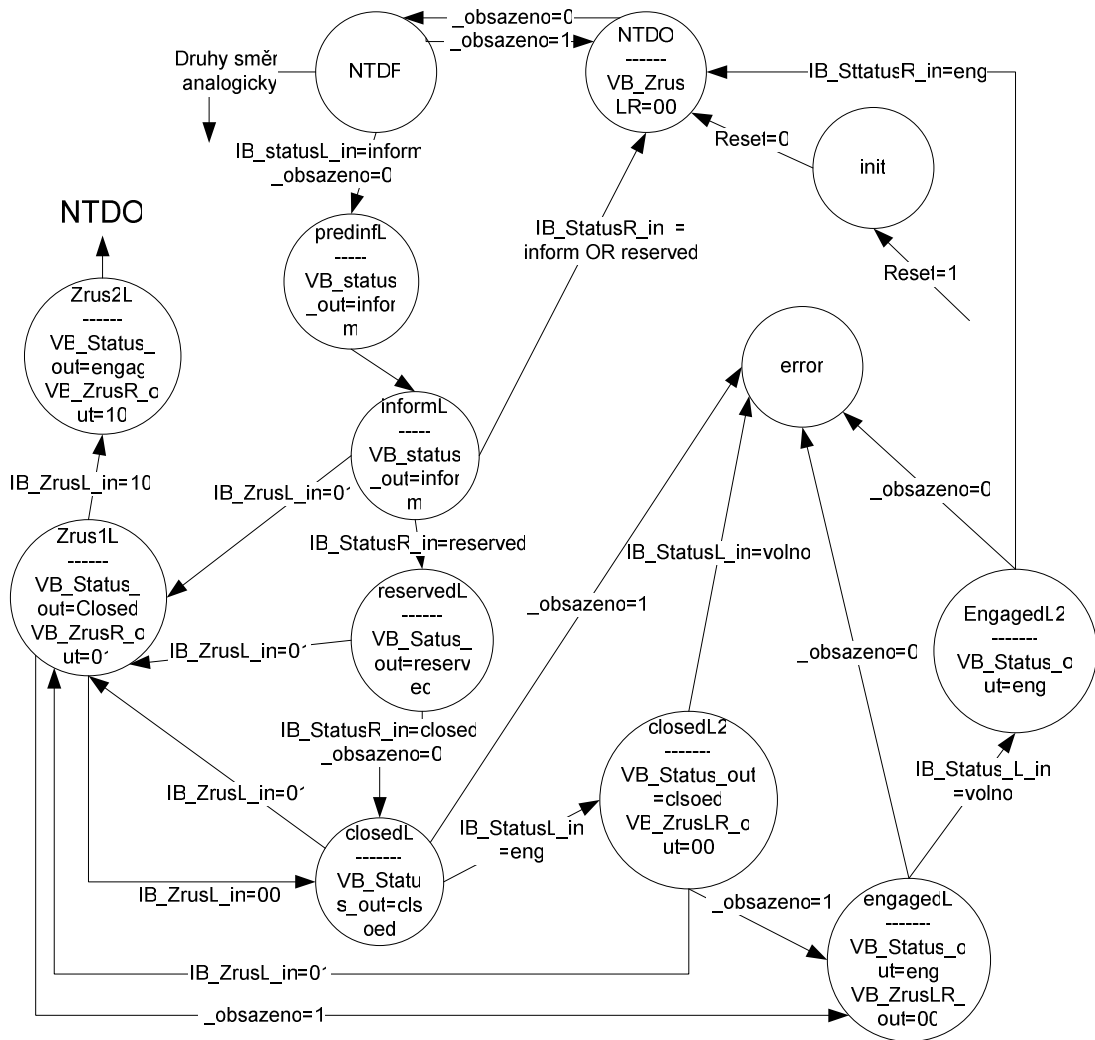
13.1 Schémata logických automatů



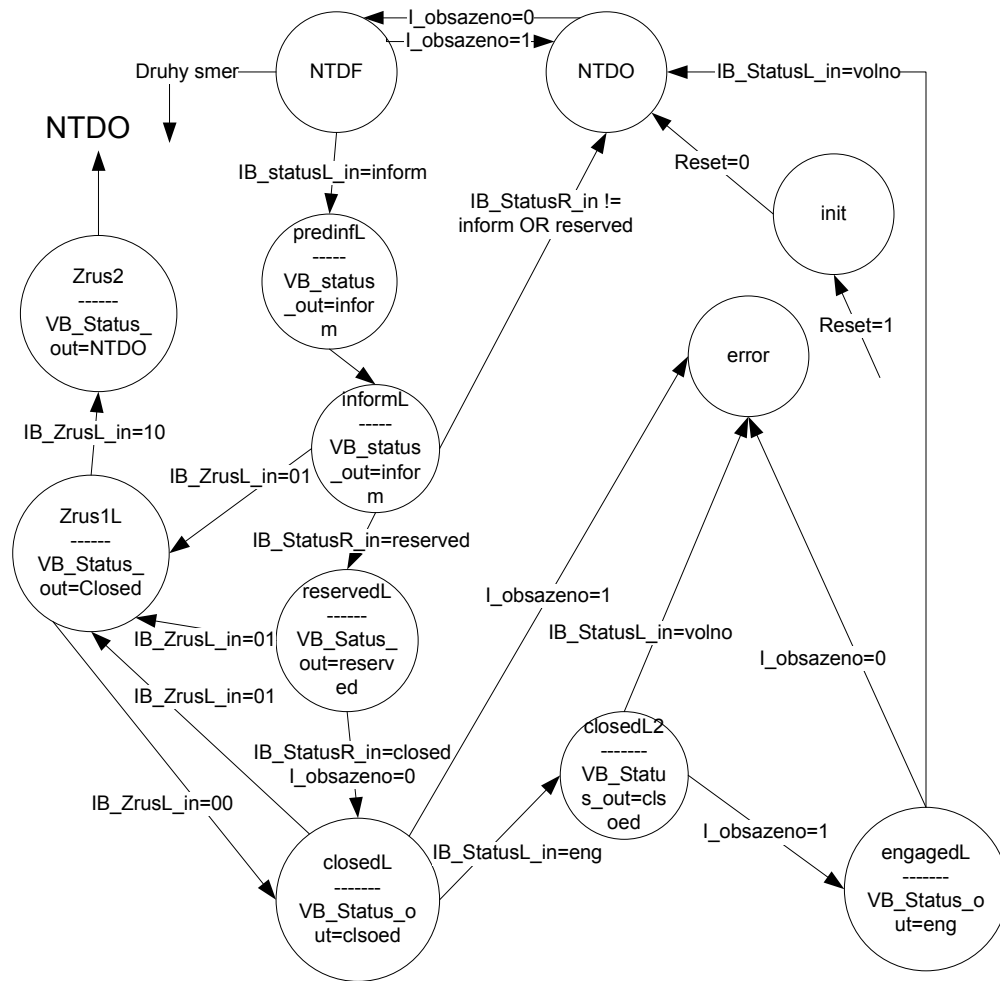
Obr. 22 : Automat VJZD – část 1



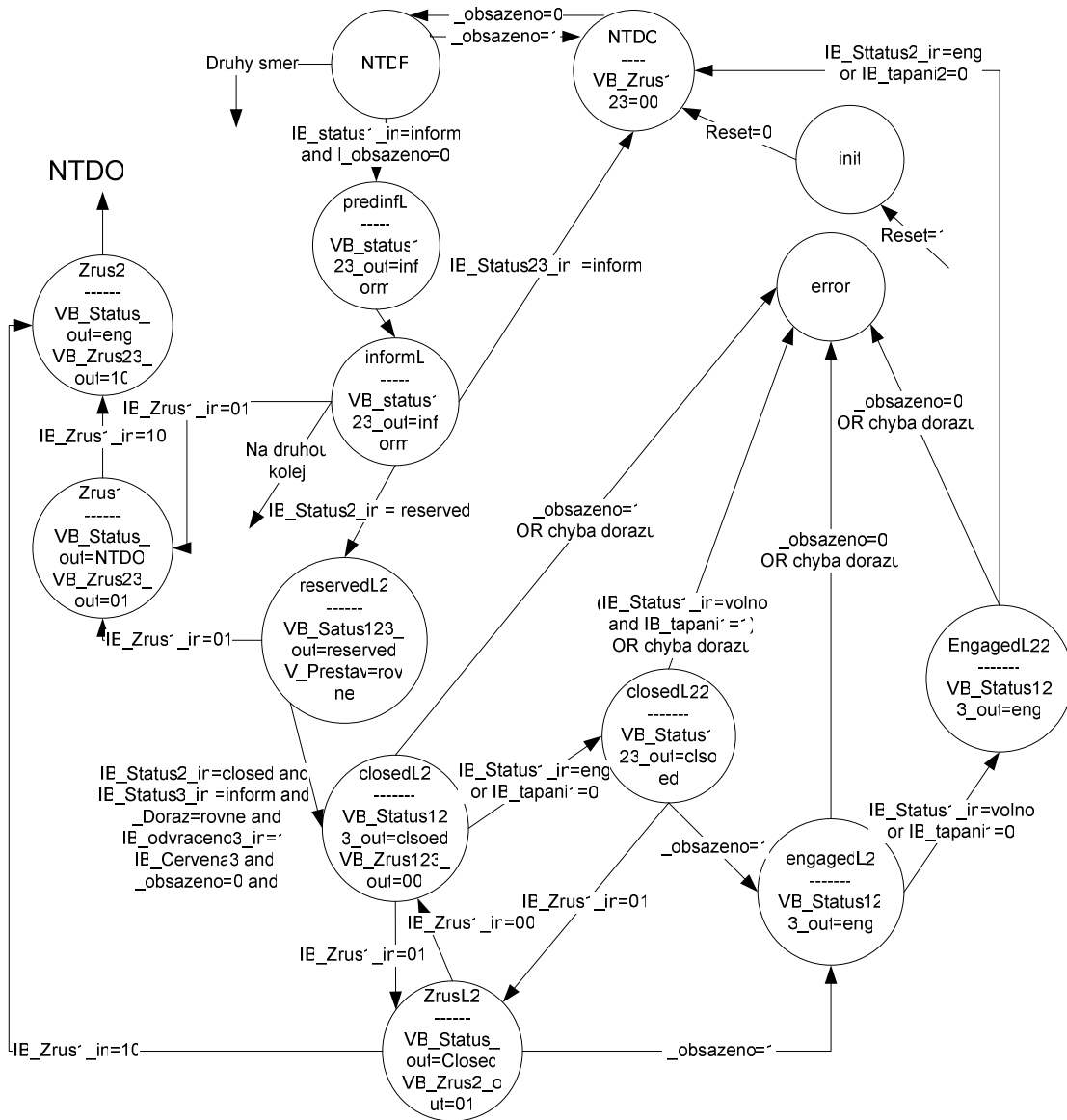
Obr. 23 : Automat VJZD – část 2



Obr. 24 : Automat M



Obr. 25 : Automat K

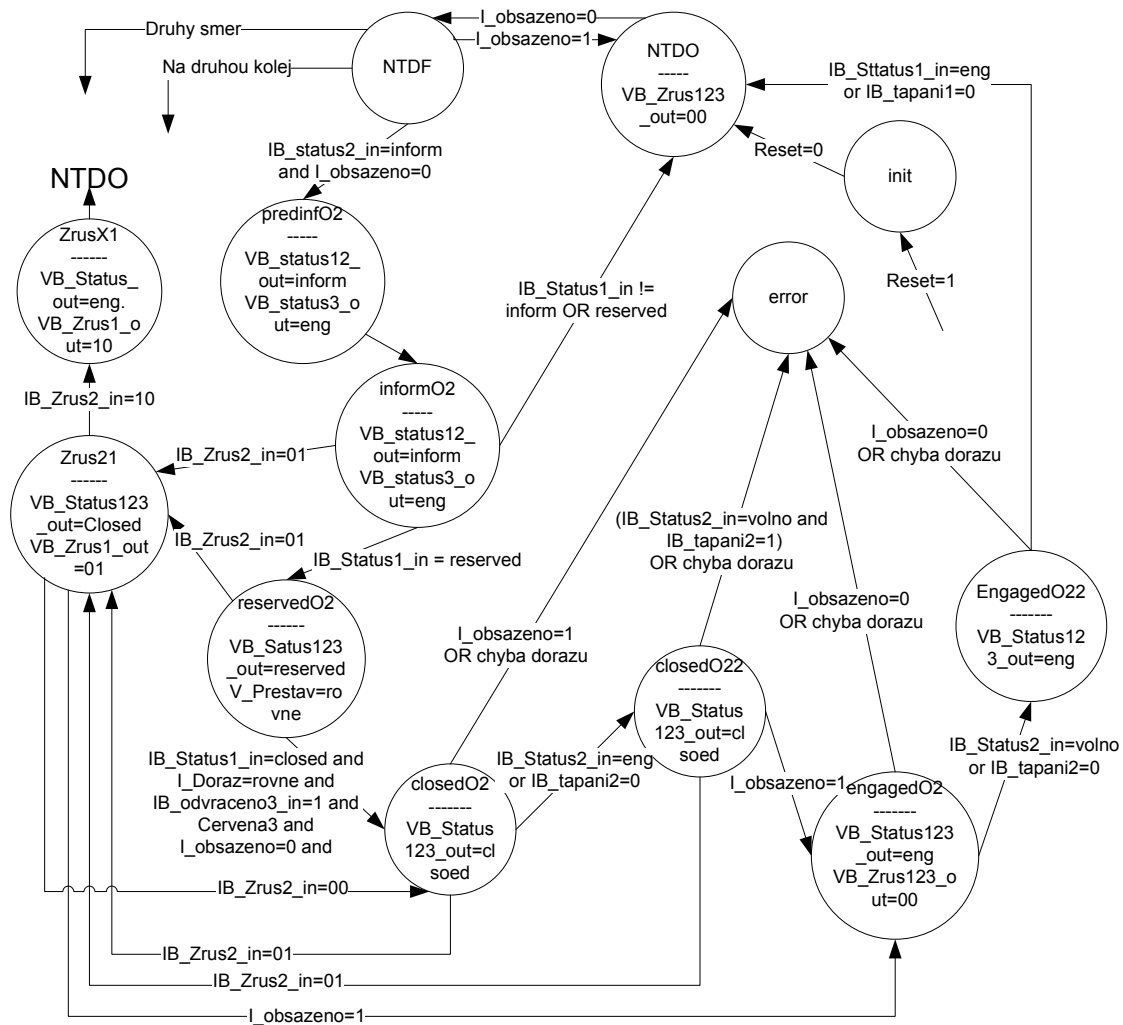


```

If rovne then
    pR23_pr_our=pL'_pr_ir
    pR23_akt_our=pL'_akt_ir if (Rychl_R>pL'_pr_ir) else rychl_r
    pL'_pr_our=pR2_pr_ir
    pL'_akt_our=pR2_akt_ir if (Rychl_R>pL2_pr_ir) else rychl_r
    else
        pR23_pr_our=pL'_pr_ir
        pR23_akt_our=pL'_akt_ir if (Rychl_odb>pL'_pr_ir) else rychl_odb
        pL'_pr_our=pR3_pr_ir
        pL'_akt_our=pR3_akt_ir if (Rychl_odb>pL3_pr_ir) else rychl_odb

```

Obr. 26 : Automat SD – část 1



Obr. 27 : Automat SD – část 2

13.2 Testovací aplikace – jednoduché nádraží

13.2.1 Příkazy

- A – Zapiš hodnotu do registru (při příjmu), informace o změně výstupu (při vysílání)
- X – Test živosti přípravku (přípravek odpoví XXX)

13.2.2 Popis registrů

VJZD registry:

RegIO 2 bity... ..	1.bit-start , 0.bit-cil
RegI 3bity	0.bit- obsazeno, 1.bit-cervenaOK , 2.bit-nesvitipovol
RegVO 5 bitu	0-2 stav, 3-4 cesta
RegV 6 bitu	0-2 HLN, 3-5 HLNpr

M registry:

RegI 1 bit	Obsazeno 0.bit
RegVO 3 bity....	Stav 0-2 bit

SD registry:

RegI ...3 bity ...	0.bit obsazeno, 2-3bit doraz
RegV...2 bity...	prestav 0-1. bit
RegVO....3bity...	stav 0-2 bit

HQ registry:

RegIO... 2 bity ...	1.bit-start , 0.bit-cil
RegI... 2 bity ...	0.bit cervenaOK, 1. bit nesviti povol
RegV... 6 bitu ...	0-2 HLN, 3-5 HLN pr
RegVO... 5bitu...	0-2 stav, 3-4 cesta

K registry:

RegI 1 bit	Obsazeno 0.bit
RegVO 3 bity....	Stav 0-2, bit

13.2.3 Popis bitů v registrech

HLN:

000-cervena

001- oranzova

010- 40 km/h

011-60 km/h

100- 80km/h

101- 100 km/h

110- zelena

HLNpr:

000-nemá se nic zobrazovat

001- oranzova

010- 40 km/h

011-60 km/h

100- 80km/h

101- 100 km/h

110- zelena

STAV:

000-rusim... blok je v rusici fazi

001-volny... blok je volny

010-reseved ... stavi se nebo se hleda cesta

011-closedpres tento blok je uzavrena cesta

100- Error chyba

111- vlak..... blok je obsazený vlakem

Popis ostatních bitu:

- Start - zapsaním jednotky se bude hledat cesta...(zacne se hledat a z po opětovném zapsaní 0)
- Cil- konec cesty..... musí byt jedna dokud se cesta nenajde , po postaveni cesty zapsat 0...

- Obsazeno --- 0..vlak tu není , 1- vlak tu je
- CervenaOK--- informace o funkci cervene 1- v poradku.... 0 nefunkcnost....
Takže pokud se nebudou simulovat poruchy , tak trvale 1
- NesvitiPovol--- chyba povolující navesti..... 0... v poradku, 1 ...chyba....
Takže pokud se nebudou simulovat poruchy , tak trvale 0
- Prestav--- draty k vyhybce, na jakou stranu se ma přestavit....00- rovne, 11-
sikmo
- Doraz --- doraz vyhybky..info pro blok... 00 rovne, 11 sikmo, 01 nebo 10
prestavuje se
- Cesta--- 00 není cesta , 01... stavi se, 11, postaveno

13.2.4 Adresy registrů

Jednoduché načrtnutí nádraží:

HQ1 K1 HQ3

VJZD_L M_L SD_L HQ2 K2 HQ4 SE_R M_R VJZD_R

Vstupní registry (do nich lze zapisovat):

Vjzd_L_regIO... adresa 0100 0000
 Vjzd_L_regI..... adresa 0100 0001
 Vjzd_R_regIO... adresa 0100 0010
 Vjzd_R_regI..... adresa 0100 0011
 M_L_regI..... adresa 0100 0100
 M_R_regI..... adresa 0100 0101
 SD_L_regI..... adresa 0100 0110
 SD_R_regI..... adresa 0100 0111
 K1_regI..... adresa 0100 1000
 K2_regI..... adresa 0100 1001
 HQ1_regIO... adresa 0100 1010
 HQ1_regI..... adresa 0100 1011
 HQ2_regIO... adresa 0100 1100
 HQ2_regI..... adresa 0100 1101

HQ3_regIO...	adresa 0100 1110
HQ3_regI.....	adresa 0100 1111
HQ4_regIO...	adresa 0101 0000
HQ4_regI.....	adresa 0101 0001

Výstupní registry (jejich změny přichází po sériové lince):

VJZD_L_regVO	adresa 0101 0100
VJZD_L_regV	adresa 0101 0101
VJZD_R_regVO	adresa 0101 0110
VJZD_R_regV	adresa 0101 0111
HQ1_regVO	adresa 0101 1000
HQ1_regV	adresa 0101 0101
HQ2_regVO	adresa 0101 1010
HQ2_regV	adresa 0101 1011
HQ3_regVO	adresa 0101 1100
HQ3_regV	adresa 0101 1101
HQ4_regVO	adresa 0101 1110
HQ4_regV	adresa 0101 1111
M_L_regV	adresa 0110 0000
M_R_regV	adresa 0110 0001
SD_L_regV	adresa 0110 0010
SD_L_regV0	adresa 0110 0110
SD_R_regV	adresa 0110 0011
SD_R_regV0	adresa 0110 0111
K1_regV	adresa 0110 0100
K2_regV	adresa 0110 0101

Příklad příkazu např. obsazení úseku u **LVjzd** je trojicí bajtů vyjádřených v ascii jako AA3.

13.3 Obsah CD

- V adresáři **Schemata** naleznete schémata vytvořených logických automatů
- V adresáři **Releovka** naleznete schémata reléových bloků
- V adresáři **Dokumentace** naleznete získanou dokumentaci k reléovému staničnímu zabezpečovacímu zařízení
- V adresáři **Projekt** naleznete projekt v ISE se zdrojovými kódy ve VHDL. Soubory bloků:
 - **HQ.vhd**
 - **K.vhd**
 - **M.vhd**
 - **SD.vhd**
 - **VJZD.vhd**
 - Ostatní VHDL kódy patří k testovací aplikaci
- V adresáři **Bitstream** naleznete bitstream pro desku SPARTAN 3E Starter Kit, komunikující s jinou diplomovou prací pomocí sériové linky.