



**Vysoká škola báňská – Technická univerzita Ostrava**



# **PROGRAMOVATELNÉ LOGICKÉ PRVKY**

Studijní opora

Jaroslav Zdrálek

Ostrava 2007

Recenze: **Ing. Zdeňka Chmelíková, Ph.D.**

Název: Programovatelné logické prvky, studijní opora

Autor: Doc. Ing. Jaroslav Zdrálek, Ph.D.

Vydání: první, 2007

Počet stran: 164

Vydavatel a tisk: Ediční středisko VŠB – TUO

Studijní materiály pro studijní obor Elektronika a sdělovací technika a obor Telekomunikační technika na Fakultě elektrotechniky a informatiky VŠB – TU Ostrava.

Jazyková korektura: nebyla provedena.

Určeno pro projekt: Operační program Rozvoj lidských zdrojů

Název: E-learningové prvky pro podporu výuky odborných a technických předmětů

Číslo: CZ.O4.01.3/3.2.15.2/0326

Realizace: VŠB – Technická univerzita Ostrava

Projekt je spolufinancován z prostředků ESF a státního rozpočtu ČR

© Jaroslav Zdrálek

© VŠB – Technická univerzita Ostrava

**ISBN 978-80-248-1502-2**

## POKYNY KE STUDIU

### Programovatelné logické obvody

Pro předmět Programovatelné logické prvky oboru Elektronika a sdělovací technika a oboru Telekomunikační technika jste obdrželi studijní balík obsahující:

- integrované skriptum pro distanční studium obsahující i pokyny ke studiu,
- CD-ROM s tutoriálem pro obsluhu návrhového systému,
- harmonogram průběhu semestru a rozvrh prezenční části,
- rozdělení studentů do skupin k jednotlivým tutorům a kontakty na tutorý,
- kontakt na studijní oddělení.

#### Prerekvizity

Pro studium tohoto předmětu se předpokládá absolvování předmětu Logické obvody.

#### Cílem předmětu

je seznámení se základními pojmy programovatelných prvků a jejich logickými a elektrickými vlastnostmi, které mají vliv na realizaci číslicových systémů. Po prostudování modulu by měl student být schopen používat programovatelné logické prvky CPLD a FPGA pro realizaci číslicových systémů.

#### Pro koho je předmět určen

Modul je zařazen do bakalářského studia oboru Elektronika a sdělovací technika studijního programu Elektronika, sdělovací a výpočetní technika a oboru Telekomunikační technika studijního programu Informační a komunikační technologie, ale může jej studovat i zájemce z kteréhokoliv jiného oboru, pokud splňuje požadované prerekvizity.

Skriptum se dělí na části, kapitoly, které odpovídají logickému dělení studované látky, ale nejsou stejně obsáhlé. Předpokládaná doba ke studiu kapitoly se může výrazně lišit, proto jsou velké kapitoly děleny dále na číslované podkapitoly a těm odpovídá níže popsaná struktura.

#### Při studiu každé kapitoly doporučujeme následující postup:

**Čas ke studiu: 30 minut**



Na úvod kapitoly je uveden **čas** potřebný k prostudování látky. Čas je orientační a může vám sloužit jako hrubé vodítko pro rozvržení studia celého předmětu či kapitoly. Někomu se čas může zdát příliš dlouhý, někomu naopak. Jsou studenti, kteří se s touto problematikou ještě nikdy nesetkali a naopak takoví, kteří již v tomto oboru mají bohaté zkušenosti.

**Cíl** Po prostudování této kapitoly budete umět



- popsat ...
- definovat ...
- vyřešit ...

Okamžitě potom jsou uvedeny cíle, kterých máte dosáhnout po prostudování této kapitoly – konkrétní dovednosti, znalosti.

## Výklad



### Použití

Následuje vlastní výklad studované látky, zavedení nových pojmů, jejich vysvětlení, vše doprovázeno obrázky, tabulkami, řešenými příklady, odkazy na animace.

Na pravé straně textu jsou uvedeny marginálie, jedná se o pojmy, které jsou v daném textu blíže rozebírány.

## Otázky kap. 1



Pro ověření, že jste dobře a úplně látku kapitoly zvládli, máte k dispozici několik teoretických otázek. Tato část zároveň slouží k shrnutí pojmů, vlastností, které si máte osvojit. Tato forma otázek byla použita místo odseku shrnutí pojmů.

## Odměna a odpočinek



Tento odsek Vás seznámí s historickým pohledem na vývoj integrovaných obvodů, hlavně na programovatelné logické prvky typu PROM, PAL, GAL, FPGA a CPLD. Součástí je také historický pohled na vývojové systémy, které úzce souvisí s použitím programovatelných prvků.

# Obsah

|                                           |           |
|-------------------------------------------|-----------|
| <b>Obsah.....</b>                         | <b>1</b>  |
| <b>Předmluva .....</b>                    | <b>5</b>  |
| <b>1 Úvodem .....</b>                     | <b>1</b>  |
| 1.1 Polozákaznické obvody .....           | 2         |
| 1.2 Význam PLD v praxi .....              | 3         |
| <b>2 Základní pojmy .....</b>             | <b>5</b>  |
| 2.1 Propojky .....                        | 8         |
| <b>3 Paměti PROM .....</b>                | <b>11</b> |
| 3.1 Paměť PROM jako AND a OR matice ..... | 11        |
| 3.2 Základní parametry paměti PROM .....  | 12        |
| 3.3 EPROM paměti .....                    | 13        |
| 3.4 Sériové paměti PROM .....             | 14        |
| 3.5 Význam PROM pamětí .....              | 15        |
| <b>4 Obvody PLA .....</b>                 | <b>17</b> |
| 4.1 Základy obvodu PLA .....              | 17        |
| 4.2 Význam obvodů PLA .....               | 18        |
| <b>5 Obvody PAL .....</b>                 | <b>19</b> |
| 5.1 Základy PAL obvodů .....              | 19        |
| 5.2 Zapojení PAL obvodu .....             | 21        |
| 5.3 Význam PAL obvodů .....               | 22        |
| <b>6 Obvody GAL .....</b>                 | <b>23</b> |
| 6.1 OLMC buňka .....                      | 23        |
| 6.2 Další vlastnosti GAL obvodů .....     | 24        |
| 6.3 Zapojení obvodu GAL22V10 .....        | 25        |
| 6.4 Označování GAL obvodů .....           | 27        |
| 6.5 Význam GAL obvodů .....               | 27        |

|           |                                                        |           |
|-----------|--------------------------------------------------------|-----------|
| <b>7</b>  | <b>Základní stavební bloky prvků CPLD a FPGA .....</b> | <b>29</b> |
| 7.1       | Logická síť na principu AND a OR matice .....          | 30        |
| 7.2       | Paměťový element .....                                 | 33        |
| 7.3       | Logická síť na principu LUT tabulek .....              | 35        |
| 7.4       | Buňka jako hradlo .....                                | 38        |
| 7.5       | Vstupně-výstupní buňka .....                           | 39        |
| 7.6       | Rozvody hodinových signálů.....                        | 42        |
| 7.7       | Správce hodinového signálu .....                       | 43        |
| 7.8       | Paměťové bloky .....                                   | 43        |
| 7.9       | Procesor .....                                         | 46        |
| 7.10      | Doplňující bloky PLD prvku .....                       | 47        |
| 7.11      | Časový model .....                                     | 47        |
| <b>8</b>  | <b>Obvody CPLD .....</b>                               | <b>49</b> |
| 8.1       | Základ CPLD obvodu .....                               | 49        |
| 8.2       | PLD obvod.....                                         | 50        |
| 8.3       | Propojovací síť a rozvody hodinového signálu .....     | 51        |
| 8.4       | Vstupně-výstupní buňky .....                           | 52        |
| 8.5       | Pouzdra .....                                          | 52        |
| 8.6       | Doplňující bloky .....                                 | 52        |
| <b>9</b>  | <b>Obvody FPGA .....</b>                               | <b>54</b> |
| 9.1       | Základy FPGA obvodu .....                              | 54        |
| 9.2       | Propojovací síť a rozvody hodinového signálu .....     | 55        |
| 9.3       | Programovatelné propojky.....                          | 55        |
| 9.4       | Vstupně-výstupní buňky .....                           | 56        |
| 9.5       | Pouzdra .....                                          | 56        |
| 9.6       | Bloky.....                                             | 56        |
| <b>10</b> | <b>Systémové vlastnosti PLD obvodů.....</b>            | <b>58</b> |
| 10.1      | Boundary Scan.....                                     | 58        |
| 10.2      | Testování I/O pinů .....                               | 63        |
| 10.3      | Systém programování ISP, ISC .....                     | 64        |
| 10.4      | BSDL jazyk .....                                       | 66        |
| 10.5      | IBIS model.....                                        | 66        |
| 10.6      | VITAL .....                                            | 66        |

|                                                                     |                                                        |            |
|---------------------------------------------------------------------|--------------------------------------------------------|------------|
| 10.7                                                                | Intellectual Properties – IP .....                     | 66         |
| 10.8                                                                | PCI kompatibilní .....                                 | 67         |
| 10.9                                                                | Pouzdra .....                                          | 67         |
| 10.10                                                               | Provozní podmínky .....                                | 67         |
| <b>11</b>                                                           | <b>Základní vlastnosti PLD prvků .....</b>             | <b>69</b>  |
| 11.1                                                                | Bezpečnostní propojka - Security Fuse .....            | 69         |
| 11.2                                                                | Nulování po náběhu napětí - Power-Up Reset .....       | 69         |
| 11.3                                                                | Elektronický podpis - Electronic Signature .....       | 69         |
| 11.4                                                                | Nahrání registru – Register Preload .....              | 69         |
| 11.5                                                                | Třístavový výstup .....                                | 70         |
| 11.6                                                                | Vedení nebo spoj .....                                 | 71         |
| 11.7                                                                | Omezující diody – Clamping Diode .....                 | 74         |
| 11.8                                                                | Odporové přizpůsobení .....                            | 75         |
| 11.9                                                                | Sběrnice .....                                         | 75         |
| 11.10                                                               | Časový rozdíl - Skew Time .....                        | 77         |
| 11.11                                                               | Strmost – Slew Rate .....                              | 78         |
| 11.12                                                               | Proudové poměry na vodiči .....                        | 79         |
| 11.13                                                               | Velikost výstupního proudu .....                       | 79         |
| 11.14                                                               | Napájecí proud hradla při změně výstupu .....          | 79         |
| 11.15                                                               | Rozvody napájecích napětí .....                        | 80         |
| 11.16                                                               | Oscilace při třetím stavu .....                        | 82         |
| 11.17                                                               | Bus Hold .....                                         | 83         |
| 11.18                                                               | Metastabilita .....                                    | 83         |
| 11.19                                                               | Překlápěcí napětí $V_{TH}$ – Threshold Level .....     | 87         |
| 11.20                                                               | Definice L a H úrovně .....                            | 87         |
| 11.21                                                               | Diferenciální logika LVDS .....                        | 89         |
| 11.22                                                               | Práce pod napětím – Live Insertion, Hot Swapping ..... | 92         |
| 11.23                                                               | Rozvody hodin .....                                    | 94         |
| <b>Příloha A - Značky používané k znázornění logické sítě .....</b> |                                                        | <b>98</b>  |
| <b>Příloha B – Termíny .....</b>                                    |                                                        | <b>100</b> |
| <b>Literatura .....</b>                                             |                                                        | <b>104</b> |

|                              |            |
|------------------------------|------------|
| <b>Internet.....</b>         | <b>107</b> |
| <b>Závěrečný test.....</b>   | <b>108</b> |
| <b>Správné odpovědi.....</b> | <b>114</b> |

## Předmluva

Tento materiál je zpracován pro potřeby elearn výuky hlavně pro posluchače kombinované formy studia oboru Telekomunikace. Tento materiál vznikl přepracováním původních skript s názvem Programovatelné logické prvky vydaných na VŠB-TU Ostrava v roce 2006. Zároveň je tento materiál doplněn o další podkapitoly. Jedná se hlavně o testování pomocí Boundary Scan.

Tento materiál je určen posluchačům jako doplněk k přednáškám z předmětu Programovatelné logické prvky, který je přednášen na Fakultě elektrotechniky a informatiky VŠB–TU Ostrava.

Programovatelné logické prvky jsou integrované obvody, které slouží k realizaci číslicových systémů a toto skriptum objasňuje jejich základní zapojení a hlavně vlastnosti, které s danou problematikou souvisí.

Pojem programovatelné logické prvky je překlad z anglického termínu Programmable Logic Device a v dalším budu používat zkratku PLD odvozenou z anglických slov. Dávám přednost používání anglických termínů včetně zkratk, protože tyto termíny nám umožňují rychleji se orientovat v zahraniční literatuře.

Při studiu těchto skript se předpokládá znalost logických obvodů a jejich základních zapojení. Programovatelné logické prvky jsou dnes jednou z alternativ jejich realizace.

Závěrem bych chtěl poděkovat Ing. Zdeňce Chmelíkové, Ph.D. za velmi cenné připomínky jak po stránce obsahové tak i věcné.

Autor

Ostrava, 2007

# 1. ÚVODEM

**Čas ke studiu: 30 minut**



**Cíl** Po prostudování této kapitoly budete umět



- Základní orientaci v typech integrovaných obvodů pro realizaci číslicových systémů
- Principy polozákaznických obvodů
- Význam programovatelných logických obvodů

## Výklad



Báze obvodů pro realizaci číslicových systémů je dnes rozsáhlá množina a je možné ji charakterizovat na základě různých kritérií. Jednoznačnou definici kritérií nelze stanovit, protože se mění rozvojem technologie a prosazováním vyšších stupňů do nižších. Osobně dávám přednost dělení, které je uvedeno dále.

Základem i dnes zůstávají obvody malé integrace, které můžeme také nazvat jako standardní obvody. Jedná se o skupinu obvodů řady 74, které stály u začátku technologie TTL – Transistor-Transistor Logic, která byla definována v 60. letech 20. století.

**TTL logika**

Další skupinu tvoří polozákaznické obvody, které se vyrábějí ve spolupráci mezi výrobcem obvodu a zákazníkem. Zákazník většinou specifikuje funkci, podílí se na návrhu zapojení na logické úrovni a výrobce jej potom realizuje na předem dané struktuře. Jedná se skupinu obvodů, kde základ tvoří hradlová pole a standardní buňky.

Skupina zákaznických obvodů – ASIC – Application Specific Integrated Circuit je skupina, která je navrhována přesně na míru požadavků zákazníka. Jedná se o jedinečné zapojení a jedinečnou výrobu obvodů. Návrh a výroba ASIC obvodů je úzce svázána s jeho výrobcem.

**ASIC  
zákaznické  
obvody**

Skupinu programovatelných logických obvodů PLD někdo zařazuje do polozákaznických obvodů, někdo ji definuje jako samostatnou skupinu. Já se přimlouvám za samostatnou skupinu, protože na definování obsahu neboli funkce obvodu již není nutná spolupráce s výrobcem. Výrobce totiž dodává univerzální zapojení, ve kterém může zákazník naprogramováním realizovat logickou síť definovaných vlastností.

Programovatelné logické prvky, Programmable Logic Devices - PLD jsou moderní součástky pro realizaci logických sítí. Vznikly postupně s rozvojem technologie výroby polovodičových prvků a dnes dokáží na ploše několika čtverečních milimetrů soustředit zapojení, které je ekvivalentem až miliónů hradel. Výhodou programovatelných logických prvků je jejich možnost, že sám uživatel definuje funkci, kterou bude prvek plnit. Způsob zadávání logické sítě do prvku se nazývá programování prvku. Po naprogramování každý programovatelný logický prvek realizuje logickou síť.

**PLD -  
Programmable  
Logic  
Devices**

## 1.1 Polozákaznické obvody

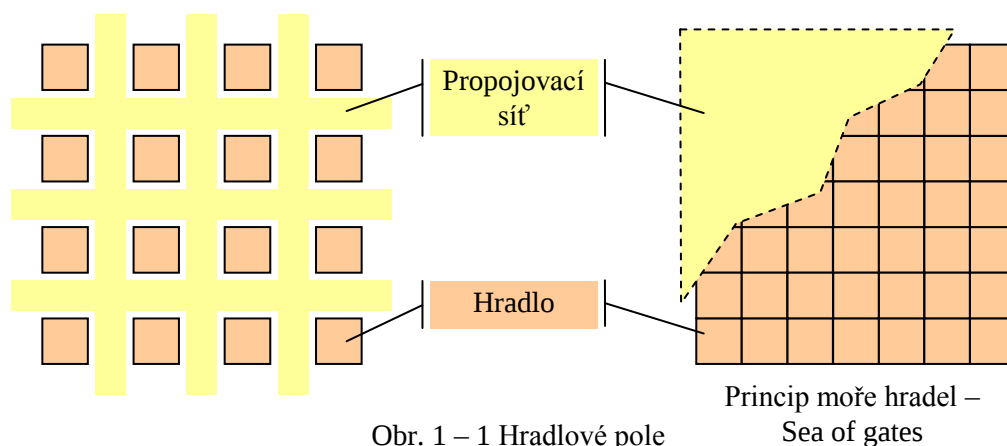
Polozákaznické obvody charakterizují jako obvody, které se vyrábějí ve spolupráci mezi výrobcem obvodu a zákazníkem. Zákazník jako návrhář již ve svém návrhu musí zohlednit požadavky výrobce. Do této skupiny zařazují dva význačné principy návrhu a výroby obvodů. Jedná se o hradlové pole a standardní buňky. Princip obou technologií je zřejmý při pohledu na čip integrovaného obvodu.

**Polozákaznické  
obvody**

Jakékoliv logické zapojení můžeme realizovat pomocí jednoho typu hradla, např. hradlem NAND o 4 vstupech. Realizace kombinační sítě pomocí hradel NAND není problém, tento princip je základem logických obvodů. Nutno si ale také uvědomit, že pomocí hradel jsme schopni realizovat i klopné obvody. Tato zapojení již mohou být předem dána formou knihovny. Detaily lze nalézt v literatuře [Diviš, Chmelíková a Zdrálek 2005]. Hradlo se tím stává univerzálním stavebním prvkem, z kterého lze realizovat libovolné logické zapojení.

**Hradlové  
pole**

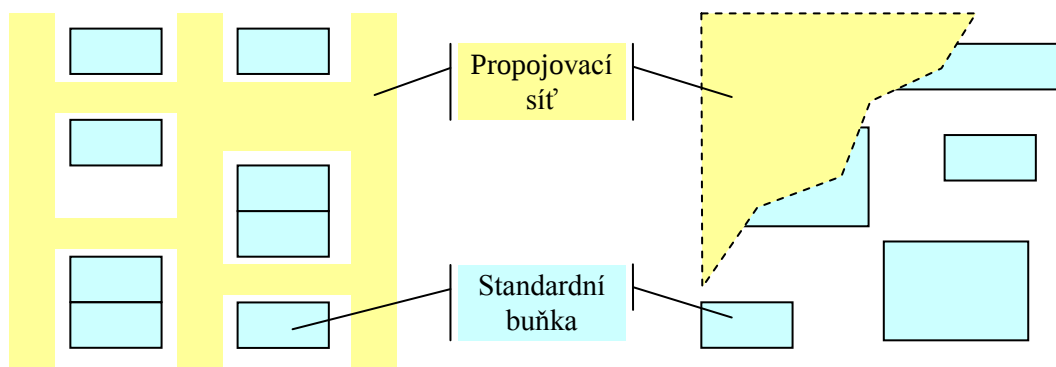
Hradlové pole je vlastně čip, který obsahuje tisíce hradel jednoho typu, obr. 1 – 1. Zákaznický návrh se na hradlovém poli vytváří pouze propojením. Výrobce nejdříve vyrobí čip, který obsahuje pouze nepropojená hradla. Propojovací síť je vytvořena podle zákaznického návrhu. Výrobce jenom generuje propojovací masky podle zákaznického návrhu. Masky pro výrobu hradel jsou stejné, nezávislé na zákazníkovi.



Obr. 1 – 1 Hradlové pole

První varianta hradlového pole je, že propojovací síť je mimo hradla. Druhá varianta je, že hradla jsou těsně vedle sebe a propojovací síť je umístěna nad nimi. Tento princip se označuje názvem moře hradel – sea of gates. Samozřejmě, že princip moře hradel umožňuje na jednotku plochy umístit mnohem více hradel než první princip.

**Moře  
hradla –  
sea of  
gates**



Obr. 1 – 2 Standardní buňky

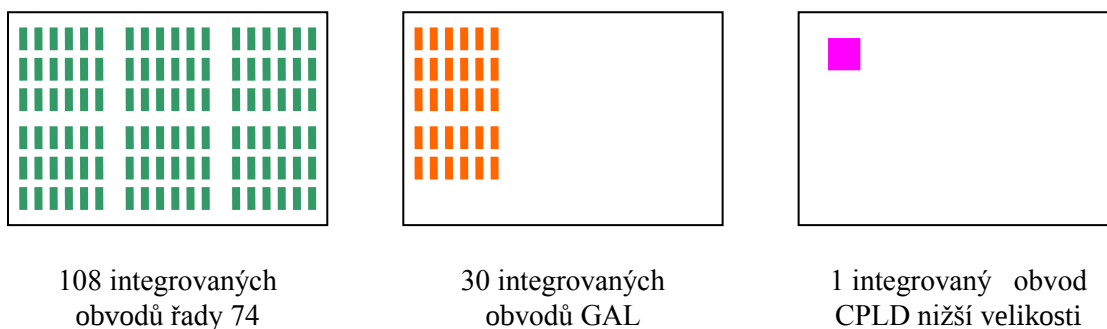
Jiný princip je technologie standardních buněk. V tomto případě výrobce má navrženy typické stavební bloky. Jedná se například o 8 bitový registr, D klopný obvod, 4 bitový binární čítač, multiplexery a demultiplexery, 4 bitovou binární sčítačku, různé typy hradel a jiné bloky. Pro tyto bloky má zároveň zpracované potřebné masky pro výrobu. Všechny informace jsou uloženy do knihovny návrhového systému. Zákazník jako návrhář navrhuje svůj systém pomocí doporučených bloků z knihovny. Výrobce si převezme návrh a vygeneruje celkové masky pro výrobu čipu. Tyto celkové masky skládá ze známých masek jednotlivých bloků. Návrhový systém navrhuje rozložení jednotlivých bloků na čipu a jejich vzájemné propojení. Pohled na čip standardních buněk je na obr. 1 – 2, kde jsou různé principy pro rozložení standardních buněk. V prvním případě mají standardní buňky stejnou konstantní šířku a umísťují se do sloupců. V druhém případě jsou standardní buňky různé velké a umísťují se libovolně na čipu.

**Standardní  
buňky**

## 1.2 Význam PLD v praxi

Význam PLD obvodů v praxi je možné chápat v několika směrech. První je ve zvyšování integrace, a tím ve snižování počtu integrovaných obvodů k realizaci daného zapojení, obr. 1 - 3. Výsledkem je zmenšení rozměru desky plošného spoje a zároveň menší plocha, kterou zabírají integrované obvody. Jedná se asi o 50-ti násobný pokles zastavěné plochy v porovnání s realizací pomocí standardních obvodů řady 74. Potom skutečnou velikost desky určuje konektor desky a počet propojovaných signálů na okolí. Z pohledu spotřeby elektrické energie dochází k jejímu poklesu.

**Význam PLD v  
praxi**



Obr. 1 - 3 Význam zvyšování integrace

Další význam je ve zkrácení cyklu vývoj, výroba a použití výrobku. Nutno si uvědomit, že návrhové systémy, které má návrhář, poskytují prostředky jednak pro návrh a také pro simulaci a programování PLD prvku. Tím je možné měnit návrh velmi rychle, řádově minuty, hodiny. Nehledě, že tyto změny nemusí mít vliv na zapojení desky plošných spojů. Proto zde dochází k výraznému zkrácení cyklu vývoj, výroba a použití. Při porovnání s manuálními postupy používanými v minulosti jako je kreslení schémat, návrh desky plošných spojů, ověření a zpracování změn dochází dnes použitím PLD prvků k výrazným časovým úsporám.

Dalším významným principem, který může zkrátit cyklus vývoj, výroba a použití je program IP – Intellectual Properties u výrobců PLD obvodů. Jedná se o program v rámci kterého si můžeme zakoupit již vyřešený programový zápis číslicového bloku a tento zápis použít ve vlastním řešení. Nákupem vyřešeného bloku se urychluje návrh vlastního číslicového systému. Jako již vyřešené bloky lze zakoupit například komunikaci na PCI sběrnici, USB řadič, Ethernet řadič, procesor a jiné.

Aplikace technologie ISC umožňuje programovat PLD prvek osazený na desce plošných spojů. Tento princip je také součástí zkrácení cyklu vývoj, výroba a použití, ale také přímo zlevňuje výrobu. Na programování PLD prvků nemusí být speciální zařízení jako jsou hardwarové programátory. Další význam technologie ISC je možnost upgradů obsahu PLD prvků, ke kterým může docházet během používání.

Nezanedbatelný význam je i možnost změny zapojení principem reprogramování. Jedná se o skutečnost, že v prvotním návrhu se mohou vyskytnout chyby, které lze dalším návrhem odstranit a tyto změny uplatnit již na vyrobených či dokonce distribuovaných systémech.

Význam PLD obvodů lze spatřovat i v ochraně zapojení proti neautorizovanému kopírování pomocí bezpečnostní propojky – Security Fuse. Tímto způsobem je znemožněno kopírování zapojení konkurencí a ochrana vlastnických práv k zapojení.

## Otázky kap. 1



- 1.1 Vyjmenujte jednotlivé typy integrovaných obvodů.
- 1.2 Popište základní principy polozákaznických obvodů.
- 1.3 Popište význam programovatelných logických prvků.

## Odměna a odpočinek



V této odpočinkové části se seznámíme s historií programovatelných logických obvodů a návrhovými systémy. Aspoň si uvědomíme, že tyto obvody mají již dlouhou historii a není to jenom záležitost posledních let. Je zde uvedeno i několik jmen a společností, které jsou spojeny s vývojem programovatelných logických obvodů. Mnoho společností, které se podílely na vývoji již dnes neexistuje, buď zanikly nebo byly koupeny větší společností nebo oblast programovatelných obvodů prodaly jiným společnostem.

S integrovanými obvody ale souvisí jedna význačná událost a to je objevení tranzistoru. Bez něj si dnes nedovedeme představit elektronické systémy.

Dříve než se budu věnovat historii PLD prvků, rád bych se zmínil o dvou meznících, které princip polovodičů významně ovlivnily. První je objevení tranzistoru v roce 1947 v Bellových laboratořích, literatura [Lucent 1997] a [Internet 06]. Zadáni pro tento objev byl praktický problém – jak nahradit elektronky, které byly příliš objemné, neefektivní a drahé pro komunikaci. Práce na tomto problému začaly před druhou světovou válkou a pokračovaly po ní, kdy skupina vědců John Bardeen, Viliam Shockley a Walter Brittain prováděla série experimentů zaměřených na použití pevných materiálů raději než vakuových trubec pro zesílení signálu. Objevitelé tranzistoru obdrželi Nobelovu cenu za fyziku v roce 1956.

**Objevení  
tranzistoru**

## 2. ZÁKLADNÍ POJMY

**Čas ke studiu: 1 hodina**



**Cíl** Po prostudování této kapitoly budete umět



- Nový způsob kreslení logických hradel s více vstupy
- Nový způsob kreslení logické sítě používaný v programovatelných logických prvcích
- Propojky pro vytváření zapojení

**Výklad**



V dalším výkladu budeme používat pojmy, zápisy a značky, které si nejdříve vysvětlíme a jejich znalost nám umožní správné a rychlé pochopení funkce bloku či logické sítě.

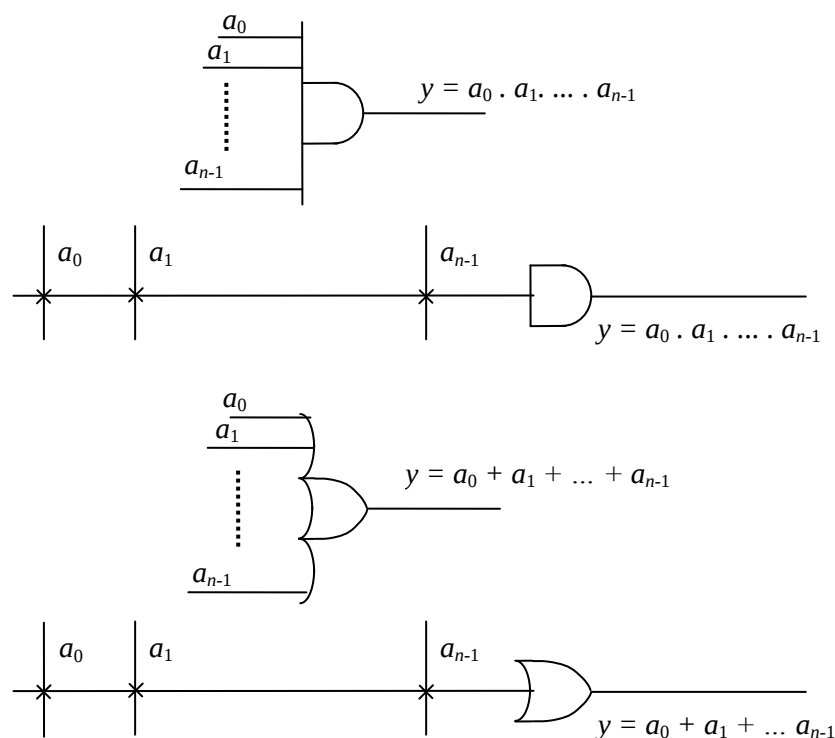
**Úvod**

Teorie číslicových systémů používá logické hodnoty 0 a 1 a binární číselné hodnoty 0 a 1. Nutno mít na paměti, že logický součet ( $1 + 1 = 1$ ) dává úplně jiný výsledek než binární aritmetický součet ( $1 + 1 = 10$ ). Skutečností však je, že binární aritmetické operace realizujeme pomocí logických operací.

V katalogu hradel se jejich funkce definuje pomocí napěťových úrovní L a H, Low a High. Automaticky se předpokládá následné vzájemné přiřazení těchto symbolů: logické hodnotě 0 se přiřazuje napěťová úroveň L a logické hodnotě 1 se přiřazuje napěťová úroveň H. Jedná se o kladnou logiku. Opačné přiřazení se nazývá negativní logika a v praxi se používá výjimečně.

V literatuře se používá několik způsobů zápisu negace. Z logických obvodů jsme zvyklí používat pro negace vodorovnou čáru nad názvem signálu. V psaném textu a hlavně v programovacích jazycích se používají i jiné zápisy. V programovacím jazyku HDL se negace píše pomocí vykřičníku před názvem signálu, např. !Dekodovani. V přímé formě signálu se vykřičník nepíše, Dekodovani. Jiný možný způsob zápisu negace signálu v psaném textu je pomocí písmena L za názvem signálu, např. Rovno\_FFFF\_L. V přímé formě se název signálu píše s písmenem H, Rovno\_FFFF\_H. Další způsob je použití slova non před nebo za názvem signálu.

Programovatelné logické prvky PLD jsou součástky využívající specifické zapojení základních logických členů. Pro znázornění zapojení PLD prvku se používají základní značky bloků, operací, se kterými se setkáváme v logických obvodech, příloha A. V dalším budeme používat značky podle amerických zvyklostí, podle kterých kreslí všichni výrobci PLD obvodů. Dále si objasníme způsob kreslení hradel o více vstupech, obr. 2 – 1, které se běžně používají v PLD prvcích.



Kreslení hradla  
AND o více  
vstupech

Kreslení hradla  
OR o více  
vstupech

Obr. 2 - 1 Vícestupňová hradla

Hradlo o více vstupech se kreslí s jednou čarou na vstupu, kterou křížují ve svislém směru čáry odpovídající vstupním signálům. To, že daný vstupní signál vstupuje do logické operace, vyjadřujeme speciálním symbolem "x" nebo "•" v místě křížení čar. U programovatelných logických prvků může být toto spojení pevné nebo programovatelné. Pevné spojení, které označujeme symbolem "•" je dáno na trvalo a definuje jej výrobce PLD. Programovatelné spojení je spojení, které definuje sám uživatel a označujeme ho symbolem "x".

Značení  
propojení

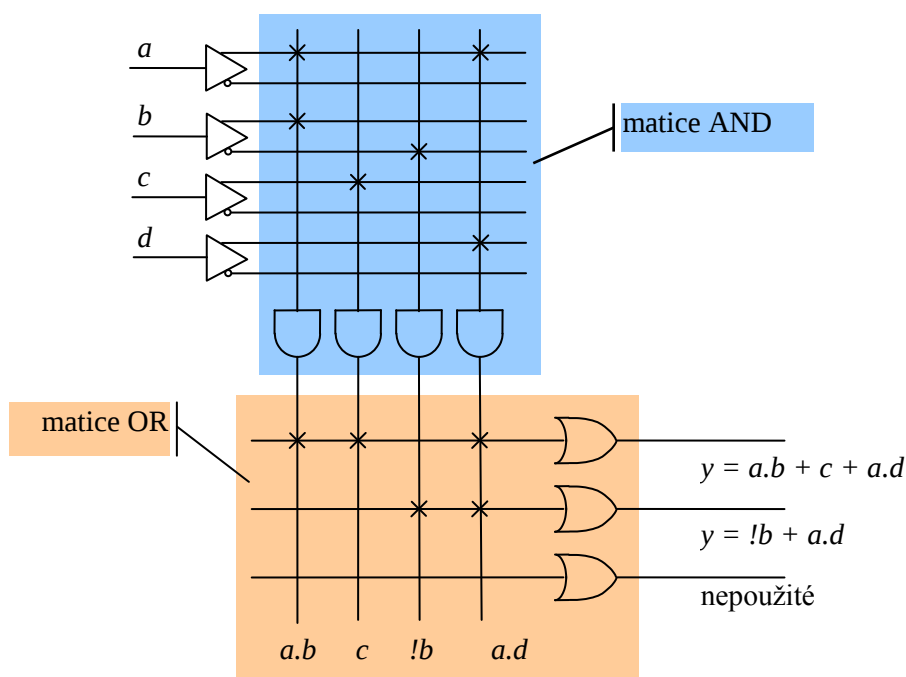
V logických obvodech jsou Boolean výrazy ( $\mathcal{B}$ -výrazy) základem pro návrh logické sítě. Z nich pro programovatelné logické prvky má význam výraz ve tvaru DNF - Disjunktivní Normálové Formy, součtové normálové formy. Základ formy tvoří součiny jednotlivých signálů a jednotlivé součiny se sčítají. Jedná se o součet součinů. Jednotlivé součiny se nazývají součinnové termy, Product Terms, obr. 2 - 2. Počet součinnových termů je důležitým parametrem PAL obvodů.

Součtová  
normálová  
forma - DNF

$$y = \underbrace{a \cdot b}_{\text{Součinnové termy}} + \underbrace{c \cdot d \cdot e \cdot f}_{\text{Součinnové termy}} + \underbrace{g}_{\text{Součinnové termy}}$$

Obr. 2 - 2 Součtová normálová forma

Od základního tvaru součtové normálové formy se odvozuje základní realizace logické sítě, obr. 2 - 3. Jedná se o realizaci pomocí hradel AND a OR, které vytváří specifické zapojení označované jako matice AND a matice OR. V místě křížení čar v jednotlivých maticích jsou propojky, které umožňují na sobě kolmé vodiče vzájemně propojit, a tím vytvářet logickou síť odpovídající Boolean výrazu v součtové normálové formě.



Realizace  
součtové  
normálové  
formy - DNF

Obr. 2 - 3 Matice AND a OR

Vstupní signály jsou přivedeny do programovatelné matice AND přes oddělovací budiče, které mají dva výstupy odpovídající vstupnímu signálu v přímém a negovaném tvaru. V programovatelné matici AND se vytvářejí jednotlivé součinné termy pomocí propojek. Výsledný součinný term z výstupu součinného hradla je přiveden do součtové matice, kde se vytváří součet jednotlivých součinných termů a na výstupu součtového hradla je výsledný výraz.

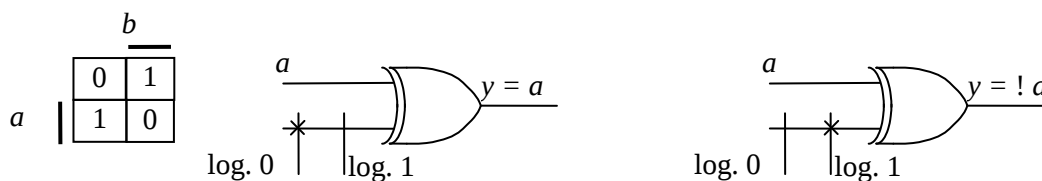
| $c$ | $b$ | $a$ | $y$ |
|-----|-----|-----|-----|
| 0   | 0   | 0   | 1   |
| 0   | 0   | 1   | 0   |
| 0   | 1   | 0   | 0   |
| 0   | 1   | 1   | 1   |
| 1   | 0   | 0   | 1   |
| 1   | 0   | 1   | 0   |
| 1   | 1   | 0   | 0   |
| 1   | 1   | 1   | 0   |

$$y = !c.b.!a + !c.b.a + c.!b.!a$$

Úplná součtová  
normálová  
forma - DNF

Obr. 2 - 4 Úplná disjunktivní normálová forma

Další odvozeným tvarem  $\mathcal{B}$ -výrazu je UDNF – Úplná Disjunktivní Normálová Forma. UDNF je tvar DNF, kdy každý součinný term obsahuje všechny vstupní proměnné  $\mathcal{B}$ -funkce v přímé nebo negované formě, obr. 2 – 4



Obr. 2 - 5 Programovatelný invertor

V programovatelných logických prvcích se dost často používá hradlo XOR ve funkci programovatelného invertoru. Funkci hradla XOR pro dvě vstupní proměnné popisuje Karnaughova mapa, obr. 2 - 5. Z ní vyplývá, když např. na vstup  $b$  přivedeme logickou nulu, tak výstup odpovídá hodnotě druhého vstupu a když na vstup  $b$  přivedeme trvale logickou jedničku, výstup odpovídá negaci druhého vstupu.

## 2.1 Propojky

Propojky slouží k propojování vodičů, a tím k vytváření požadované logické sítě. Propojky mohou vodiče spojit nebo ne a jejich stav určuje buď výrobce nebo návrhář. V anglické literatuře se dost často pro označení propojky používá termín „fuse“. Každá propojka programovatelného prvku může být ve stavu, kdy vodiče propojuje nebo ne. Proces, který definuje jeden z těchto stavů propojky, se nazývá programování. Programování prvku se uskutečňuje na speciálních zařízeních nazývaných programátor nebo speciálními postupy. Jinak, programování prvku je speciální procedura předepsaná přesně výrobcem. Takovéto propojky nazýváme programovatelné. Existují i propojky pevné, tyto propojky propojují vodiče napevno a jejich realizace a umístění je dáno výrobcem.

Pokud výše uvedené matice AND a OR používají pouze jeden typ propojky, potom se tyto matice nazývají pevné nebo programovatelné, pevná matice AND, programovatelná matice AND, pevná matice OR a programovatelná matice OR. Propojky dělíme podle způsobu programování na maskou programovatelné a programovatelné u uživatele.

Maskou programovatelné propojky jsou jednou programovatelné. Jedná se o propojky, které se nastavují pomocí masky pro výrobu integrovaného obvodu ve výrobním závodu. Z pohledu uživatele to značí odevzdat na základě dohody výrobcí polovodičů nastavení propojek a výrobce zajistí jejich naprogramování formou masky. Tato technologie se uplatňovala v počátcích programovatelných prvků v 70tých letech, hlavně u ROM pamětí.

U prvků s programovatelnými propojkami se určuje stav propojky – propojeno nebo ne. Pro určení stavu propojky uživatel používá speciální přístroje jako jsou programátory programovatelných prvků nebo postupy pod označením ISP - In System Programming, dnes ISC – In System Configurable. Skupina prvků s programovatelnými propojkami se dále dělí na:

- Jednou programovatelné prvky. Jedná se o prvky, kde propojka je výrobcem nastavena do výchozí hodnoty a programováním se propojka nastaví do opačné hodnoty. Zpětné uvedení propojky do výchozí polohy není možné. Typická realizace takovéto propojky je propojení vodičů tenkým drátkem (fuse), který se proudovým impulsem propálí. Z pohledu uživatele to značí, že v případě určitých chyb v návrhu je nutné pro jejich opravu použít nový prvek. Základní nevýhodou těchto propojek pro výrobce je jejich nemožnost testování nedestructivním způsobem. Nelze otestovat propojky jejich naprogramováním a pak prvek použít. Tento problém odstraňují principy vícenásobného programování.
- Vícenásobně programovatelné prvky. Jedná se o prvky, které lze po naprogramování vymazat a opět programovat, propojku lze uvést do původního stavu. Pro výrobce

**Programovatelný invertor**

**Programovatelné a pevné propojky**

**Programovatelné matice AND a OR**

**Maskou programovatelné propojky**

**Programovatelné propojky u uživatele**

**Jednou programovatelné propojky**

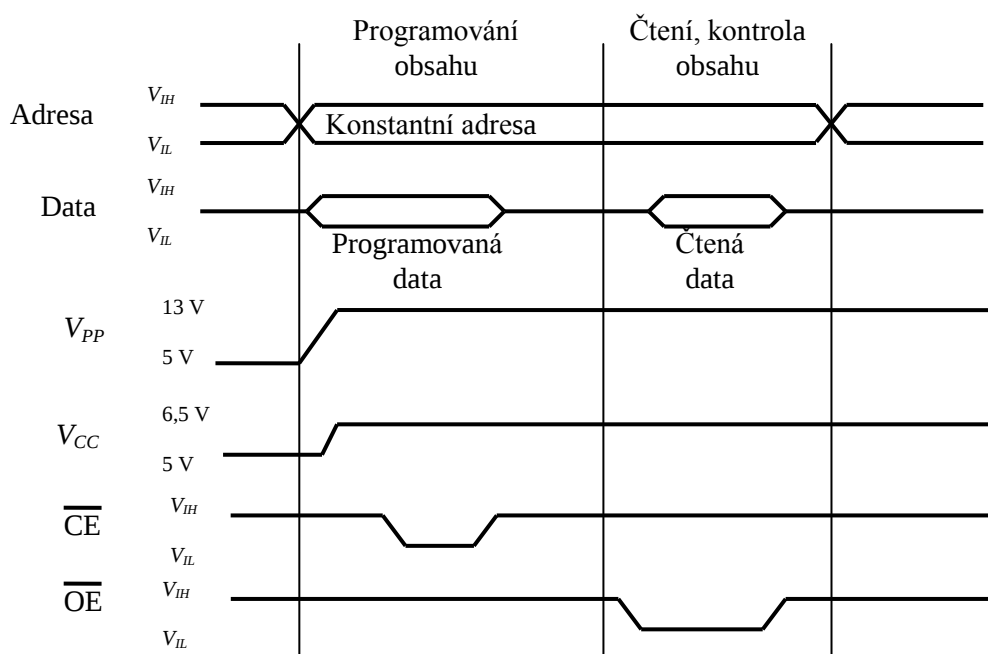
**Vícenásobně programovatelné propojky**

prvků tato technologie přinesla možnost kompletně testovat vyrobený prvek, včetně propojek. Skupina se dále dělí podle způsobu vymazání stavu propojky na:

- Technologie EPROM. Jedná se o prvky, u kterých se vymazání provádí pomocí UV-světla, ultrafialového světla, kterým se osvítl přímo polovodičová struktura prvku. Pouzdro tohoto prvku má speciální skleněné okénko, které je umístěné na vrchní straně pouzdra prvku, přímo nad polovodičovou strukturou. Dnes tato technologie zaniká.
- Technologie EEPROM. Jedná se o prvky, u kterých se vymazání provádí elektricky. Samotné vymazání se provádí v programátorech nebo pomocí postupů ISP.

Programátor programovatelných logických prvků je speciální zařízení k určení stavu programovatelných propojek, které se děje speciálními postupy danými výrobcem prvku. Programování se většinou provádí při zvýšeném napájecím napětí prvku po určitou dobu. Příklad programovacího postupu pro programovatelný prvek typu EPROM je na obr. 2 - 6. V obrázku nejsou zakresleny jednotlivé časy mezi jednotlivými signály.

**Ukázka  
programovacích  
o postupu**



Obr. 2 - 6 Ukázka programovacího předpisu pro prvek typu EPROM

Z pohledu spolehlivosti se u programování zabýváme parametry:

- Počet garantovaných zápisů nebo cyklů. Jedná se o parametr, který určuje kolikrát lze prvek programovat. Reálná hodnota je od 1 000 až do 1 miliónu cyklů, [Xilinx 2003a] a [Atmel 2004].
- Data Retention. Jedná se o čas, po který prvek neztratí svůj obsah. Nutno si uvědomit, že stav propojky je uchovávan ve formě elektrického náboje na kondensátoru nebo na plovoucí G elektrodě CMOS tranzistoru. Detailní výklad je mimo rozsah těchto skript. Reálná hodnota je od 10 do 40 roků, [Xilinx 2003a] a [Atmel 2004].
- Správnost postupu při programování. Nevhodné programování může způsobit v prvku latentní poruchu, která se projeví jako trvalá porucha časem. Výrobci programovatelných prvků tento problém řeší tím, že garantují správnou funkci prvku, který byl naprogramován na jimi schválených nebo vyrobených programátorech.

**Parametry  
spolehlivosti u  
programování**

**Otázky kap. 2**

- 2.1 Jak se kreslí logická hradla s více vstupy?
- 2.2 Které hradlo se používá jako programovatelný invertor a proč? (Vysvětlit pravdivostní tabulku hradla.)
- 2.3 Součtová normálová forma, úplná součtová normálová forma, součinný term?
- 2.4 Realizace B-funkce pomocí matice AND a OR.
- 2.5 Jaké typy propojek používáme v maticích programovatelných logických prvků?

**Odměna a odpočinek**

Dalším významným mezníkem pro integrované obvody je slavné tvrzení pana Moora, „Počet tranzistorů v integrovaných obvodech se přibližně každé dva roky zdvojnásobí“, literatura [Computerworld 2005] a [Busniss World 2006]. Původně toto tvrzení, které bylo prvně publikováno v časopise Electronics Magazine v roce 1965, obsahovalo periodu jeden rok. V roce 1975 své původní tvrzení opravil na každé dva roky. Sám pan Moore tvrdí, že nikdy neřekl 18 měsíců, což je nejčastější výklad jeho tvrzení. Pan Moore se v roce 1969 stal spoluzakladatelem společnosti Intel. Jeho tvrzení, dnes označované jako Moorův zákon, potvrdila historie. Jaké bude pokračování tohoto tvrzení ukáže čas. Podle výzkumné zprávy společnosti Intel přestane Moore zákon platit kolem 2021, kdy bude 16 nm technologie vylepšená jedním či dvěma kroky k miniaturizaci.

**Moore zákon**

Historie PLD obvodů je krátká, ale za to bohatá. První PLD obvody byly PROM paměti, které se objevily na trhu kolem roku 1970. V té době se pojem PLD ještě nepoužíval a aplikace pamětí pro realizaci logických sítí byla v začátcích. Proto se PROM paměti nepokládají za počátek PLD odvodů.

**První PLD jsou  
PROM paměti**

### 3. PAMĚTI PROM

**Čas ke studiu: 1 hodina**



**Cíl** Po prostudování této kapitoly budete umět



- Princip činnosti PROM paměti
- Aplikace PROM paměti jako logický kombinační obvod

**Výklad**



Paměť PROM - Programmable Read Only Memory je logický prvek, který slouží k uchování informací. Dále budu používat toto všeobecné označení, které je nezávislé na způsobu programování propojek. Paměť PROM má vstupní signály, které nazýváme adresní vstupy nebo adresa a výstupní signály, které nazýváme datové výstupy nebo data a pomocné vstupní řídicí signály. Adresné signály se obvykle označují  $A_{n-1}, A_{n-2}, \dots, A_1, A_0$  a datové výstupy  $Q_{m-1}, \dots, Q_1, Q_0$ , blíže obr. 3 – 2. K jedné konkrétní adrese je přiřazena jedna paměťová buňka. Počet paměťových buněk je dán počtem adresných signálů a je roven  $2^n$ . Každá paměťová buňka má délku  $m$  bitů, každému bitu paměťové buňky odpovídá příslušný datový výstup. Počet datových výstupních signálů PROM paměti odpovídá počtu bitů paměťové buňky. Každá paměťová buňka může obsahovat libovolnou kombinaci hodnot bitů. Obsah každé paměťové buňky definuje uživatel. Paměti PROM charakterizují pojmy organizace paměti a kapacita paměti. Organizaci paměti vyjadřujeme počtem paměťových buněk a počtem bitů v buňce ve tvaru  $2^n \times m$  bitů. Pro počet paměťových buněk se často používá pojem počet slov o délce  $m$  bitů nebo při počtu 8 bitů v buňce přímo  $2^n$  bytů. Kapacita paměti je celkový počet bitů v paměťovém prvku a rovná se součinu počtu paměťových buněk a počtu bitů v buňce.

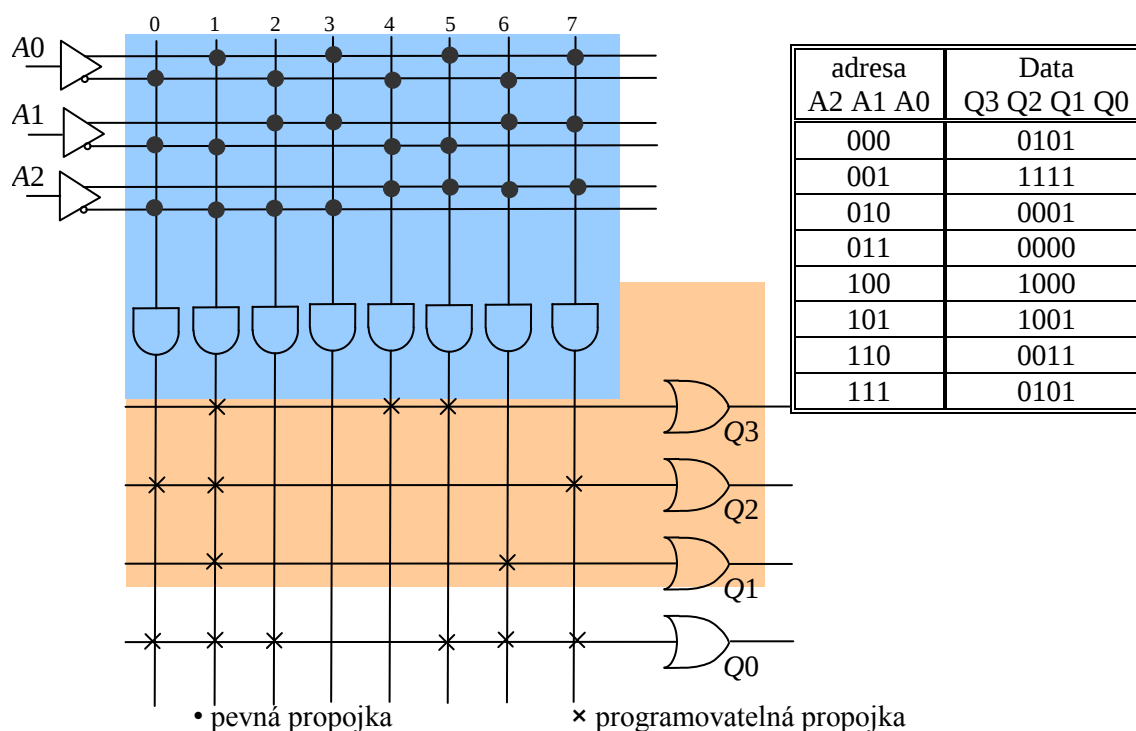
**Pojmy spojené  
s PROM  
paměti**

Z pohledu historie byl přístup k datům paralelní. Paměť má paralelní vstup adres a paralelní výstup dat. Dnes se přechází k používání paměti se sériovým přístupem. Tím se dosahuje velké snížení počtu pinů integrovaného obvodu a také zmenšení zabírané plochy prvkem na desce plošného spoje.

#### 3.1 Paměť PROM jako AND a OR matice

Z pohledu uplatnění teorie AND a OR matice má paměť PROM pevnou matici AND a programovatelnou matici OR. Do pevné matice AND vstupují adresné signály a na matici se realizuje dekodér z binárního kódu do kódu "1 z  $n$ ". Požadovaný počet vstupů do každého součinnového hradla je  $n$  a počet hradel, a tím i součinnových termů je  $2^n$ . Příklad PROM paměti s organizací 8 slov x 4 bity je na obr. 3 - 1. Pevná AND matice vytváří adresný dekodér 1 ze 8. Jeden sloupec v programovatelné OR matici je vlastně jedna paměťová buňka délky 4 bity. Pro popis obsahu paměti PROM se především využívá forma tabulky, obr. 3 - 1. Při tomto zápisu je ke každé adrese přiřazen obsah paměťové buňky. U paměti s vyšší organizací se pro přehlednost používají vyšší číselné soustavy, hlavně šestnáctková, výjimečně osmičková. Nepoužitým bitům se přiřazuje většinou konstantní hodnota. Zápis obsahu paměti formou tabulky je východiskem pro programování PROM paměti.

**PROM paměť  
jako AND a  
OR matice**



Obr. 3 - 1 Paměť PROM s organizací 8 x 4 bitů

Paměť PROM je logický kombinační obvod, který lze popsat i pomocí  $\mathcal{B}$ -výrazu ve tvaru UDNF, úplné disjunktivní normálové formy.

$$Q3 = !A2 !A1 A0 + A2 !A1 !A0 + A2 !A1 A0$$

$$Q2 = !A2 !A1 !A0 + !A2 !A1 A0 + A2 A1 A0$$

$$Q1 = !A2 !A1 A0 + A2 A1 !A0$$

$$Q0 = !A2 !A1 !A0 + !A2 !A1 A0 + !A2 A1 !A0 + A2 !A1 A0 + A2 A1 !A0 + A2 A1 A0$$

Právě tyto úvahy jsou základem realizace logického kombinačního obvodu pomocí PROM paměti. Tabulka obsahu paměti odpovídá pravdivostní tabulce logického kombinačního obvodu až na názvy signálů. U logického kombinačního obvodu hovoříme o vstupních a výstupních proměnných a u paměti hovoříme o adrese a výstupních datech. Návrh spočívá v přiřazení jednotlivých signálů. Pro  $\mathcal{B}$ -funkci o  $n$  proměnných  $x$  je nejvhodnější přiřazení  $A_0 = x_0$ ,  $A_1 = x_1$  až  $A_n = x_n$  a nepoužité adresné vstupy musí být ošetřeny na konstantu. Pomocí paměti PROM o organizaci  $2^n \times m$  bitů lze realizovat maximálně  $m$   $\mathcal{B}$ -funkcí každou o maximálně  $n$  vstupních proměnných.

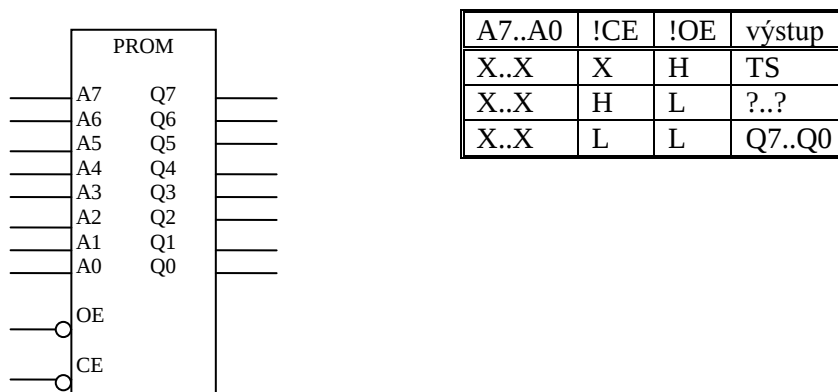
**PROM paměť  
jako  
kombinační  
logický obvod**

**Princip návrhu  
kombinačního  
obvodu pomocí  
PROM paměti**

### 3.2 Základní parametry paměti PROM

Schematická značka paměti PROM je na obr. 3 - 2. Ve skutečnosti jsou paměti doplněny dalšími řídicími signály a to výběr prvku CS, Chip Select a povolení výstupu OE, Output Enable. Signál CS, výběr prvku aktivuje prvek, uvádí jej do aktivního stavu. Signál OE, povolení výstupu řídí výstupní budiče pro datové signály. V případě třístavového výstupu signál OE řídí třetí stav výstupu paměti.

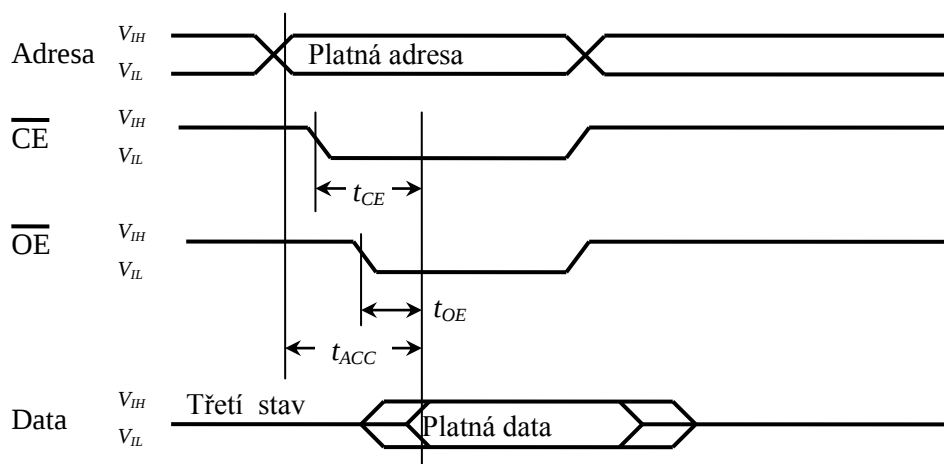
**Parametry  
PROM paměti**



Obr. 3 – 2 Paměť PROM, schematická značka a její obsah

Za základní technické parametry mimo organizace paměti se považuje přístupová doba paměti, obr. 3 - 3. Přístupová doba je čas, za který se ustálí datový výstup paměti od změn vstupních signálů. Za základní přístupovou dobu se považuje přístupová doba od adresy, která je označována  $t_{ACC}$ . Další přístupové doby jsou  $t_{CE}$  od povolení obvodu a  $t_{OE}$  od povolení výstupu. Výrobce garantuje, že data na výstupu paměti po uplynutí přístupové době jsou stabilní nebo-li platná. Pokud neuplyne příslušná přístupová doba nelze data na výstupu považovat za správná. Všechny přístupové doby jsou v katalogu definovány maximální hodnotou a potom typickou hodnotou.

Časové  
parametry  
PROM paměti



Obr. 3 - 3 Přístupová doba paměti

Za základní výhodu PROM paměti je považována možnost realizace **libovolné**  $\mathcal{B}$ -funkce nebo  $\mathcal{B}$ -funkcí, které splňují podmínku počtu adresných vstupů a počtu datových výstupů. V současnosti se vyrábějí PROM paměti různých kapacit a organizací. Nejmenší vyráběná a prodávána paměť s organizací 32 x 8 bitů, s kapacitou 256 bitů je SN74188 fy Texas (70tá léta), dnes označena JBP18S030 s přístupovou dobou od adresy 50 ns. Jedná se o jednu programovatelnou PROM paměť. Dnes se vyrábějí EPROM paměti s organizací 1 G x 8 bitů, s kapacitou 8 Gbit a více. Jedná se o reprogramovatelnou paměť. Mezi tím existuje široká škála pamětí s různou organizací, s různou kapacitou a přístupovou dobou.

Reálné paměti  
PROM

### 3.3 EPROM paměti

V této stati s názvem EPROM paměti se budeme zabývat pamětmi typu EEPROM a Flash, které lze opětovně programovat a elektricky vymazat. Velký význam pro tyto vlastnosti má umístění

násobiče napětí přímo na křemíkový čip a aplikace řídicího signálu WRITE. Tyto nové technologie či nové módy činnosti EPROM paměti umožňují zapisovat do paměti EPROM přímo z okolních obvodů, například procesoru. Zápis do paměti EPROM na desce plošného spoje má význam jednak při ladění obsahu paměti a také možnost změny obsahu v průběhu životnosti zařízení. Dnes se změnou obsahu můžeme setkat např. u osobních počítačů, kde se provádí upgrade BIOSu, upgrade firmware některých periférií. Možnost zápisu obsahu EPROM pomocí speciálních programovacích zařízení zůstává.

EEPROM paměť pracuje jako energetická nezávislá RAM paměť. Každou paměťovou buňku lze číst nebo do ní zapisovat. Paměť EEPROM nelze vymazat, lze do ní pouze zapsat samé nuly, postupně do všech buněk.

**Paměti  
EEPROM**

Naproti tomu do FLASH paměti se data zapisují klasickým programováním. FLASH paměť je možné vymazat, akce mazání vymaže definovaný blok FLASH paměti, většinou se tento blok rovná celé FLASH paměti. Čtení obou pamětí je stejné.

**Paměti  
FLASH**

| Sekvence         | Poč. cyk. | 1. cyklus |       | 2. cyklus |      | 3. cyklus |      | 4. cyklus |      | 5. cyklus |      | 6. cyklus |      |
|------------------|-----------|-----------|-------|-----------|------|-----------|------|-----------|------|-----------|------|-----------|------|
|                  |           | Adr.      | Data  | Adr.      | Data | Adr.      | Data | Adr.      | Data | Adr.      | Data | Adr.      | Data |
| Read             | 1         | adr       | D out |           |      |           |      |           |      |           |      |           |      |
| Chip erase       | 6         | 555       | AA    | AAA       | 55   | 555       | 80   | 555       | AA   | AAA       | 55   | 555       | 10   |
| Sector Erase     | 6         | 555       | AA    | AAA       | 55   | 555       | 80   | 555       | AA   | AAA       | 55   | Adr sec   | 30   |
| Byte program     | 4         | 555       | AA    | AAA       | 55   | 555       | A0   | adr       | D in |           |      |           |      |
| Boot block       | 6         | 555       | AA    | AAA       | 55   | 555       | 80   | 555       | AA   | AAA       | 55   | 555       | 40   |
| Product ID entry | 3         | 555       | AA    | AAA       | 55   | 555       | 90   |           |      |           |      |           |      |
| Product ID exit  | 3         | 555       | AA    | AAA       | 55   | 555       | F0   |           |      |           |      |           |      |
| Product ID exit  | 1         | XXX       | F0    |           |      |           |      |           |      |           |      |           |      |

Obr. 3 - 4 Ukázka softwarových sekvencí EPROM paměti

Některé typy EPROM pamětí dnes mají nové registry, které identifikují jednak výrobce EPROM paměti a také konkrétní typ paměti. Pro tuto vlastnost se používá pojem Product Identification – identifikace výrobku, který obsahuje Manufacture Code – kód výrobce a Device code – kód zařízení. Tato identifikace se uplatní hlavně při programování na desce plošného spoje, kde je možné před změnou obsahu si zkontrolovat správnost osazení, a tím i uplatnit správné postupy pro danou paměť. Data Product Identification se získávají buď hardwarově nebo softwarově. Hardwarový způsob je dán katalogovým listem. Při softwarovém způsobu se používají sekvence dat.

**Identifikace  
EPROM  
paměti**

Programování na desce plošného spoje se také provádí pomocí softwarových sekvencí dat. Jedná se o definovaný sled zápisů předem definovaných dat na předem definované adresy. Příklad softwarových sekvencí je na obr. 3 - 4. Paměti EPROM dost často obsahují informace, které je vhodné chránit proti neoprávněnému přepisu, např. BIOS. Proto některé paměti EPROM obsahují možnost blokovat zápis do některé části či celé paměti. Tato blokace může být jednak hardwarová (zapojení signálů WRITE přes propojku) nebo pomocí softwarových sekvencí dat.

**Zápis do  
EEPROM  
paměti  
sekvencemi dat**

### 3.4 Sériové paměti PROM

Dosud jsme se zabývali pamětmi s paralelním způsobem komunikace. Ale plocha pouzdra, ve kterém je paměť umístěna, je u velkých kapacit také velká a je vlastně dána počtem vývodů integrovaného obvodu. Takové řešení má vlastnosti:

- Zabírá velkou plochu na desce plošných spojů. Např. paměť 128 K x 8 bitů v pouzdru TSOP zabírá 160 mm<sup>2</sup>, [Atmel 2003].
- Počet pinů integrovaného obvodu je velký, a tím i počet propojovacích vodičů, kterými je integrovaný obvod zapojen na svoje okolí.

**Sériové paměti  
PROM**

- Přístup k datům je paralelní, a tím i rychlý, řádově desítky nanosekund.

Ale jsou i aplikace, které vyžadují pravý opak, kdy paměť zabere na desce plošných spojů málo místa a přístup může být mnohem pomalejší. Proto se dnes vyrábějí paměti se sériovým přístupem. Dnes se prosazují dva způsoby 2-drátový interface nebo SPI interface.

### 3.5 Význam PROM paměti

Paměti PROM jsou nejstarší PLD prvky a po uvedení na trh se začaly používat k realizaci  $\mathcal{B}$ -funkcí. Základní vlastnosti jsou:

- Možnost realizace **libovolné**  $\mathcal{B}$ -funkce o  $n$ -vstupech, kde  $n$  je zároveň počet adresných vstupů PROM paměti. U PLA, PAL a GAL obvodů je tato možnost omezena počtem součinných termů.
- Možnost v jednom integrovaném obvodu realizovat až  $m$   $\mathcal{B}$ -funkcí, kde  $m$  je počet výstupů z PROM paměti.
- Realizace  $m$   $\mathcal{B}$ -funkcí, každá maximálně o  $n$  vstupech v jednom integrovaném obvodu. V porovnání s klasickou realizací pomocí hradel vzniká velká úspora počtu potřebných integrovaných obvodů k jejich realizaci, a tím i úspora plochy na desce plošných spojů. Zpoždění takto realizovaného obvodu obvykle bývá větší než realizace pomocí hradel.
- Jednoduché sestavení dat pro programování, protože obsah PROM paměti odpovídá pravdivostní tabulce  $\mathcal{B}$ -funkce, za podmínky vhodného přeznačení vstupů a výstupů.

**Význam  
paměti PROM.  
Porovnání  
s PLA, PAL a  
GAL obvody.**

### Otázky kap.3



- 3.1 Popište PROM paměť, hlavně se zaměřte na názvy jednotlivých vstupních a výstupních signálů, organizace paměti.
- 3.2 PROM paměti jako kombinační obvod?
- 3.3 Popište princip návrhu kombinačního logického obvodu pomocí paměti PROM?
- 3.4 Popište význam EEPROM a Flash paměti?
- 3.5 Jaký je význam sériového přístupu k obsahu paměti PROM?
- 3.6 Porovnejte PROM paměti s PLA, PAL a GAL architekturou?

### Odměna a odpočinek



Historie principu programovatelných matic AND a OR spadá před rok 1975, kdy tyto obvody vyráběla společnost National Semiconductors jako maskou programovatelné obvody. Dnes se považuje za vznik programovatelných logických obvodů rok 1975, kdy společnost Signetics Corporation aplikovala programovatelné propojky u programovatelných matic AND a OR ve svém obvodu 83S100, později pod označením PLS100. Jednalo se již o obvody programovatelné u zákazníka. Těmto obvodům dala název FPLA – Field Programmable Logic Array. Tvůrcem obvodů byl Napoleon Calvan, který je označován za otce programovatelných logických obvodů. Dnes jsou tyto obvody označovány jako PLA.

První obvody FPLA nenašly výraznou odezvu u návrhářů číslicových systémů. Obvody vykazovaly velké zpoždění. Další nevýhodou byl způsob návrhu, kdy návrhář musel přímo definovat hodnoty propojek, a tím vytvářet logické zapojení a také speciální programátor pro jejich programování. Neexistence návrhového prostředku a také jejich cena byly příčinou malého rozšíření v praxi.

Další významnou událostí je uvedení PAL obvodů společnosti MMI – Monolithic Memories Inc. Za tvůrce PAL obvodů je považován John Martin Brikner. Jedná se o obvody s programovatelnou AND maticí a pevnou OR maticí. Aplikace pevné součtové matice snížila požadavky na plochu křemíku a přispěla k menšímu zpoždění PAL obvodů vůči PLA. Počet součinnových termů pro každý součet byl stanoven na základě empirických zkušeností při návrhu číslicových systémů, kdy jako základ bylo zvoleno 8 součinnových termů pro jeden součet. Způsob programování PAL obvodů byl obdobný jako u paměti PROM. První katalog byl zveřejněn v roce 1978. K prvním PAL obvodům patří PAL16L8, PAL16R4, PAL16R6 a PAL16R8.

## 4. OBVODY PLA

Čas ke studiu: 15 minut



**Cíl** Po prostudování této kapitoly budete umět



- Princip PLA obvodů

### Výklad

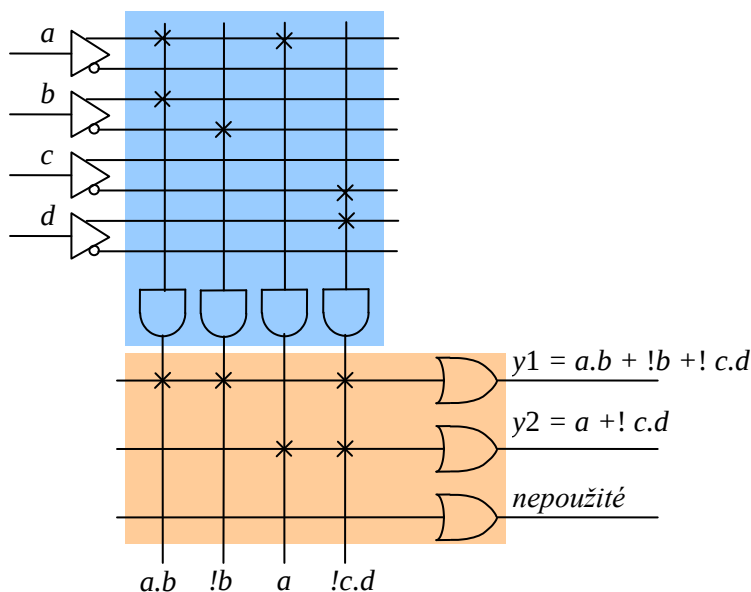


Obvody PLA byly jedny z prvních, které se v minulosti začaly vyrábět. Avšak z trhu je vytlačily obvody PAL. Dnes se s principem PLA můžeme setkat u některých CPLD obvodů.

#### 4.1 Základy obvodu PLA

Obvody PLA, Programmable Logic Array je skupina obvodů PLD, která vychází z teorie programovatelných matic AND a OR, obr. 4 - 1. Obě matice jsou programovatelné.

**AND a OR  
matice**



Obr. 4 - 1 Princip obvodů PLA

Obvody PLA realizují zapojení kombinační sítě, které vychází z DNF – součtové normálové formy. V programovatelné součinové matici se vytvářejí jednotlivé součinové termy ze vstupních signálů v přímém a negovaném tvaru. V programovatelné součtové matici se potom provede součet jednotlivých součinových termů.

Na obrázku je PLA struktura určená pouze pro logickou kombinační síť. Dalším pokračováním je umístění paměťového elementu za součtové hradlo, a tím možnost vytvářet logické sekvenční

obvody. S těmito variantami se blíže seznámíme u PAL a GAL obvodů, popřípadě CPLD obvodů.

## 4.2 Význam obvodů PLA

Význam PLA obvodů je dnes spíše historický, protože jako samostatná skupina se v minulosti neprosadily. PLA obvody byly vytlačeny PAL obvody a dnes GAL obvody. Základní vlastnosti jsou:

**Význam PLA obvodů**

- Možnost skupinové minimalizace, tím že jeden součinnový term mohl být použit ve více výstupních funkcích.
- Počet součinnových termů pro výstupní funkci není omezen jako u obvodů PAL.

### Otázky kap. 4



- 4.1 Popište princip architektury PLA.
- 4.2 Jaký je význam PLA architektury?

### Odměna a odpočinek



Je čas na skutečnou kávu.

## 5. OBVODY PAL

Čas ke studiu: 1 hodina



**Cíl** Po prostudování této kapitoly budete umět



- Princip PAL obvodů
- Nový způsob kreslení AND a OR matic, který se používá v praxi
- Aplikace kombinačního výstupu a výstupu s registrem

### Výklad



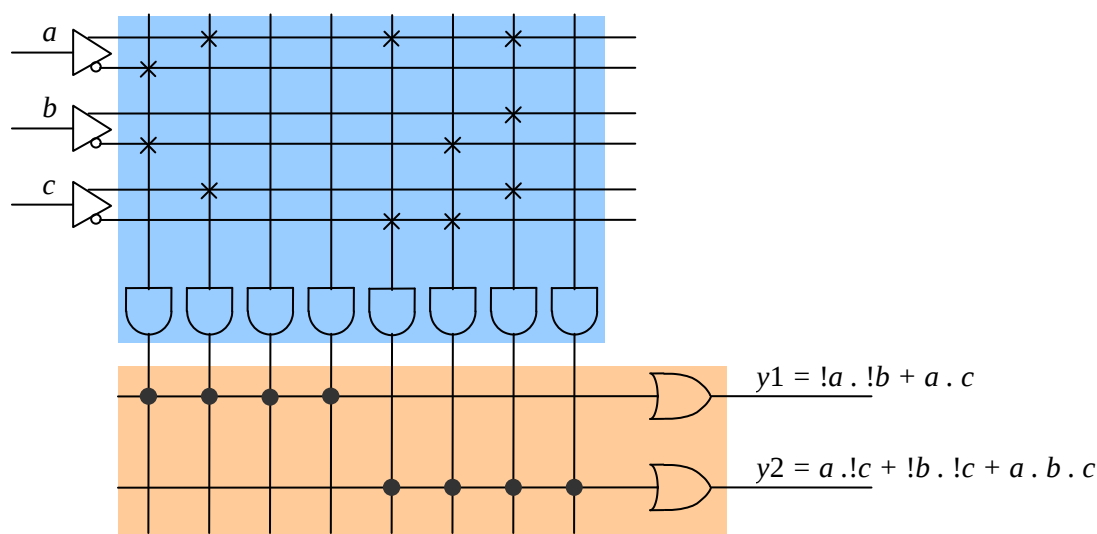
Obvody PAL – Programmable Array Logic jsou variantou aplikace součinnové a součtové matice. V době jejich vzniku doznaly velkého praktického rozšíření a byly jednou programovatelné. Obvody PAL byly časem nahrazeny obvody GAL.

### 5.1 Základy PAL obvodů

Obvody PAL, Programmable Array Logic je skupina obvodů vycházející z teorie matic AND a OR, kde matice AND je programovatelná a matice OR je pevná. Základní architektura obvodu PAL je na obr. 5 - 1.

**AND a OR  
matice**

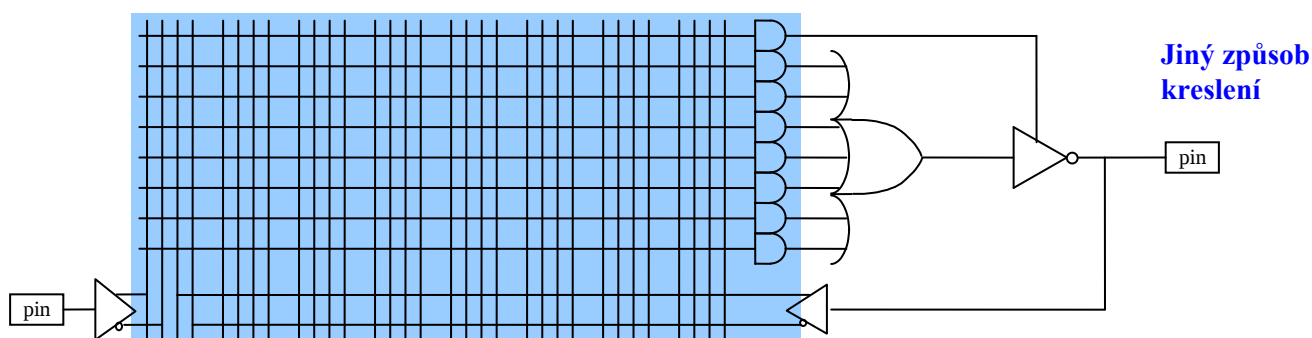
Kombinační logická síť se zapojuje na základě DNF – součtové normálové formy. Podle obrázku může každý výstup použít do logického součtu maximálně 4 součinnové termy. Výraz  $y_1$  má pouze dva součinnové termy, v zapojení se použijí pouze dvě součinnová hradla a dvě zůstanou nepoužitá. Nepoužitá součinnová hradla neovlivňují výsledný výraz  $y_1$ . Z obrázku je vidět, že důležitým parametrem PAL obvodů je počet možných součinnových termů.



**Programovatelná AND a pevná OR matice**

Obr. 5 - 1 Principiální zapojení PAL obvodů

Počet součinných termů vstupujících do součtu je pevný a je dán konkrétním typem PAL obvodu. Počet výstupů, a tím i počet součtových hradel je také dán konkrétním typem obvodu. U skutečných PAL obvodů můžeme najít různé modifikace zapojení. Příklad možného zapojení kombinačního výstupu s třetím stavem a zpětnovazebním signálem je na obr. 5 - 2. Z hlediska zapojení se součtová matice nerozkresluje a součinná hradla se kreslí přímo na vstupy součtového hradla.



Obr. 5 - 2 Zapojení kombinačního obvodu PAL

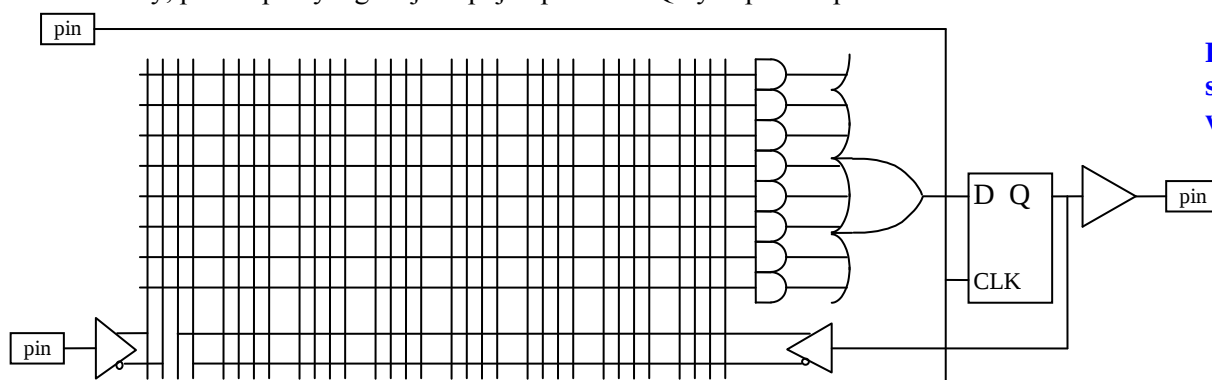
Jedná se o zapojení, kde součinná matice má 8 vstupních signálů a každý signál je rozveden v přímém a negovaném tvaru. Pomocí součinného hradla lze vytvořit součin až 8-mi signálů. Do součinné matice jsou přivedeny nejen vstupní signály z pinů ale také zpětné signály odvozené od výstupních signálů. Samotný výstup obsahuje výstupní třístavový budič s negací. Třetí stav je řízen povolovacím signálem, který je v tomto případě generován jako součin v součinné matici. Zpětný signál je napojen na výstupní pin obvodu. Tento princip umožňuje použití výstupního pinu několika způsoby:

**PAL obvod  
s kombinačním  
výstupem**

- Jako vstupního (výstup je trvale ve třetím stavu).
- Možnost propojit výstup součtového hradla zpět do součinné matice, a tím rozšířit možnosti realizovaných  $\mathcal{B}$ -výrazů. Nevýhodou je zvýšení zpoždění.
- Možnost vytvářet asynchronní sekvenční obvody. Dnes se asynchronní sekvenční obvody doporučuje nahrazovat synchronními sekvenčními obvody.
- Možnost vytvářet synchronní sekvenční obvody, v tomto případě výstup je se synchronním paměťovým elementem, viz obr. 5 - 3.

Další možná modifikace výstupu je zapojení paměťového elementu na výstup součtového hradla, obr. 5 - 3. Jako paměťový element se často používá D klopný obvod. Hodinový signál se v tomto případě přivádí na pin obvodu. D klopný obvod obvykle nemá vstup pro nastavení nebo nulování výstupu. V tom případě platí, že klopný obvod po náběhu napětí přechází do předem definovaného stavu, obvykle je to L-úroveň na výstupu klopného obvodu. Výstup není třístavový, proto zpětný signál je napojen přímo na Q výstup D klopného obvodu.

**PAL obvod  
s registrovým  
výstupem**

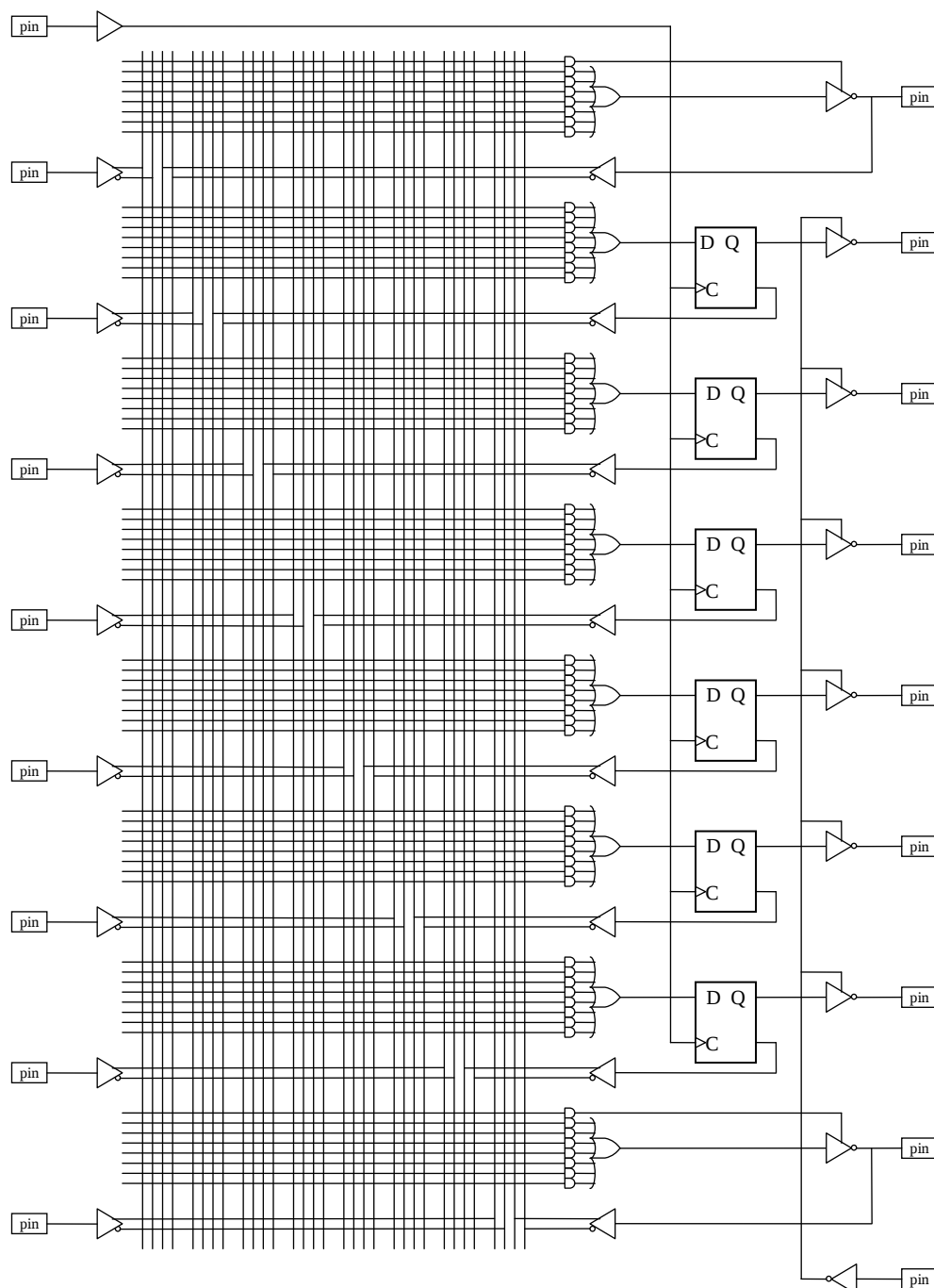


Obr. 5 - 3 Zapojení PAL s paměťovým elementem

## 5.2 Zapojení PAL obvodu

Zapojení PAL obvodů je dáno konkrétním typem a nalezneme je v katalogu. Na obr. 5 - 4 je zapojení obvodu PAL16R6. Jedná se o obvod se 16 vstupy do součinové matice, 8 výstupy, z nichž 2 jsou kombinační a 6 výstupů je s paměťovým elementem. Všechny výstupy jsou třístavové.

U obvodů PAL se prvně objevila vlastnost bezpečnostního bitu, Security Fuse. Jedná se speciální bit, který po odpálení zamezuje čtení stavu propojek z integrovaného obvodu. Tato vlastnost je vhodná pro ochranu zapojení navrženého obvodu, návrhář si chrání svoje zapojení proti neoprávněnému použití, kopírování.



**Skutečné  
zapojení PAL  
obvodu**

Obr. 5 - 4 Zapojení obvodu PAL16R6

### 5.3 Význam PAL obvodů

Význam PAL obvodů je dnes spíše historický, protože se dnes přechází k používání GAL obvodů. Základní vlastnosti jsou:

- Realizace  $B$ -funkce je omezena počtem součinnových termů pro každý výstup.
- PAL obvody, co se týče zpoždění, jsou rychlejší než PLA a PROM paměti. Aplikace pevné součtové matice snížilo zpoždění.
- Aplikace bezpečnostní propojky – Security Fuse.
- Bylo vyráběno hodně typů podle konkrétního zapojení výstupu.
- Při uvedení na trh byl k dispozici první návrhový prostředek pod názvem Fuse assembler. Tím bylo zautomatizováno generování nastavení jednotlivých propojek.
- Byly úspěšnější než PLA obvody.
- Původně byly PAL obvody jednou programovatelné, dnes se objevují i reprogramovatelné a hlavně jsou nahrazeny obvody GAL.

**Význam PAL  
obvodu**

#### Otázky kap. 5



- 5.1 Popište princip architektury PAL.
- 5.2 Jak se aplikuje kombinační výstup s třetím stavem do PAL struktury?
- 5.3 Jak se aplikuje registr do PAL struktury?
- 5.4 Jaký je význam PAL architektury?

#### Odměna a odpočinek



PAL obvody doznaly významného rozšíření v praxi. Za základní důvody rozšíření jsou považovány skutečnosti:

- Menší zpoždění obvodů v porovnání s PLA a PROM obvody.
- Způsob programování, který byl obdobný jako u PROM pamětí. V té době používané programátory vyžadovaly pouze malé úpravy.
- Zveřejnění zdrojového textu prvního Fuse assembler - assembleru propojek v programovacím jazyku FORTRAN. Tento program umožňoval vygenerovat zápis nastavení jednotlivých propojek obvodu PAL na základě logických výrazů. Tento program můžeme považovat za první vývojový systém PLD obvodů.

Dalším významným krokem bylo zavedení OLMC buňky – Output Logic Macro Cell v roce 1983 společností AMD v obvodu pod označením PAL22V10. Jedná se o zapojení, které umožňuje realizovat výstup jako kombinační nebo s paměťovým elementem – D klopný obvod a v obou případech s aktivní L nebo H úrovní výstupu. Hlavní význam OLMC buňky je možnost náhrady mnoho typů PAL obvodů pouze jedním typem. Dnes OLMC buňku spojujeme s GAL obvody.

## 6. OBVODY GAL

Čas ke studiu: 1 hodina



**Cíl** Po prostudování této kapitoly budete umět



- Princip GAL obvodů
- Zapojení a význam OLMC buňky
- Elektronický podpis

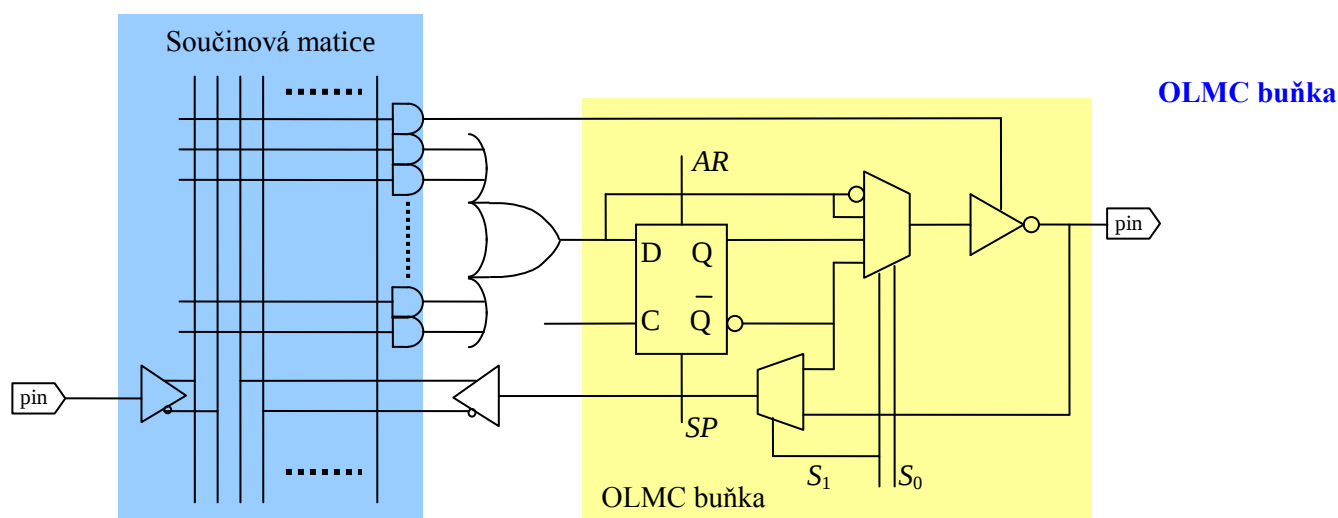
**Výklad**



Obvody typu GAL, Generic Array Logic jsou obvody, které navazují na obvody PAL a rozšiřují je o dvě významné vlastnosti. První vlastností je reprogramovatelnost GAL obvodů, čímž vzniká skupina elektricky mazatelných obvodů. Druhou vlastností je rozšíření používání OLMC - Output Logic Macro Cell, která umožňuje jeden obvod využívat jako kombinační nebo s paměťovými elementy. Poprvé byla OLMC buňka aplikována v PAL struktuře společnosti AMD, dnes se OLMC buňka spojuje hlavně s GAL obvody. Obvody GAL vycházejí z obvodů PAL, mají programovatelnou matici AND a pevnou matici OR a na výstup součtového hradla je zařazena OLMC. Zapojení logické kombinační sítě vychází z DNF – součtové normálové formy. GAL obvody dnes tvoří skupinu PLD prvků označovanou jako standardní obvody.

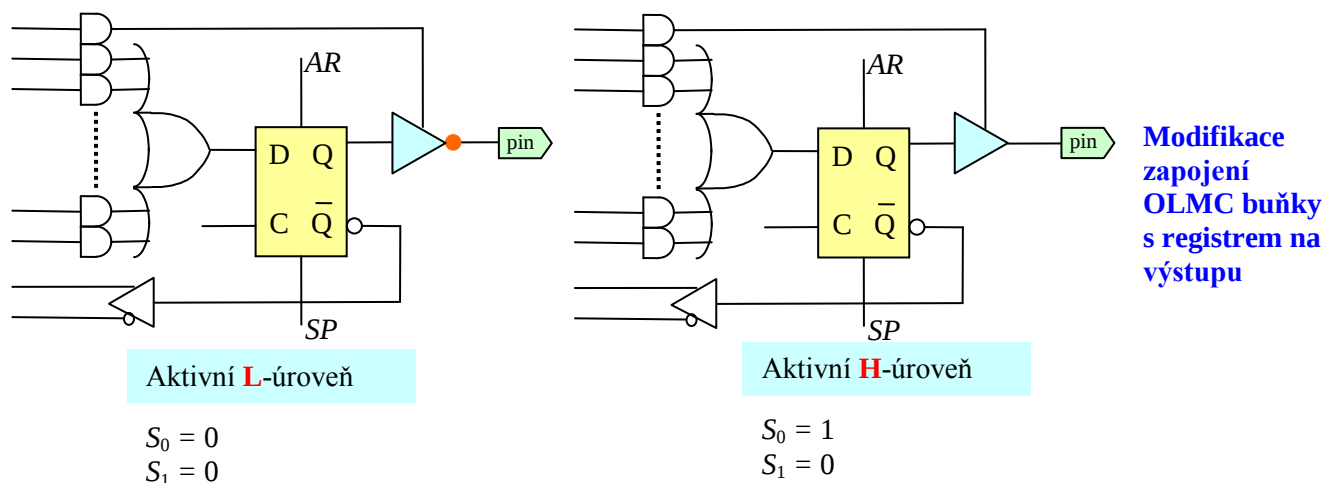
### 6.1 OLMC buňka

GAL obvody mají základní strukturu s programovatelnou maticí AND a pevnou maticí OR. Na každý výstup součtového hradla je umístěna buňka z názvem Output Logic Macro Cell, jedno z možných zapojení je obr. 6 - 1. Jedná se o OLMC buňku obvodu GAL22V10.

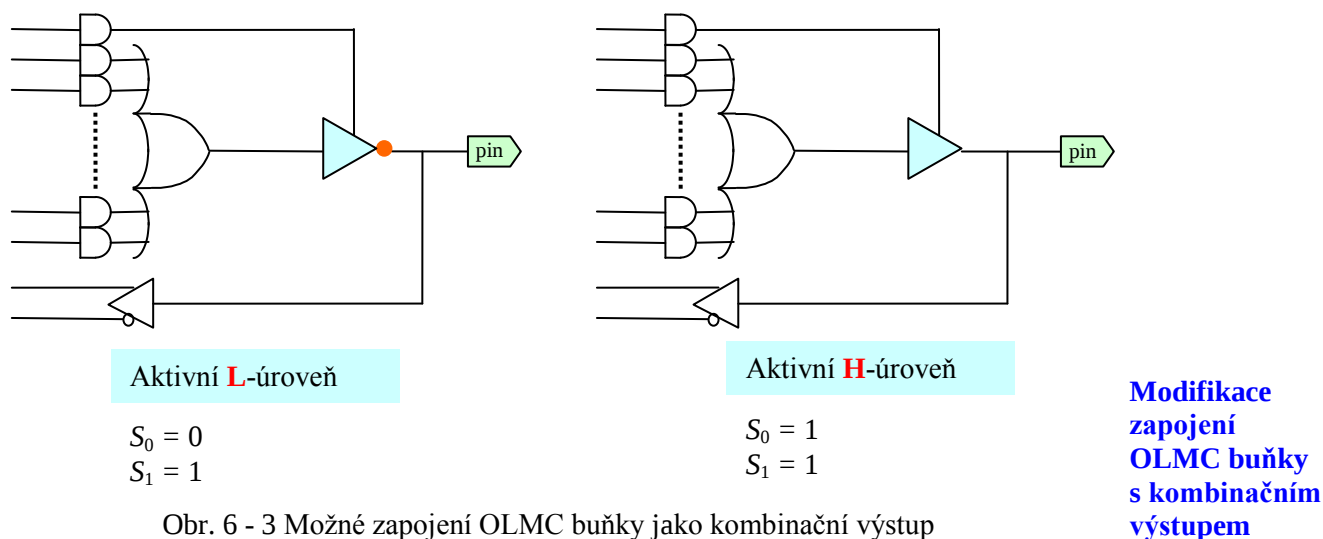


Obr. 6 - 1 OLMC buňka obvodu GAL22V10

OLMC buňka je konfigurovatelná pomocí programovatelných propojek  $S_0$  a  $S_1$ , které umožňují konfigurovat buňku jako kombinační nebo registrový výstup. Všechny možné konfigurace jsou uvedeny na obr. 6 - 2 a obr. 6 - 3. Jedná se o výstup s paměťovým elementem nebo jako kombinační. Další rozdíl je v aktivnosti výstupu, v přímém tvaru výstup nemá invertor a v negovaném tvaru jej obsahuje. Jako paměťový element je většinou D klopný obvod. Signál AR – Asynchronous Reset je asynchronní nulování a signál SP je Synchronous Preset – synchronní nastavení. Signály AR, SP a CLK mohou být odvozeny od pinu nebo generovány jako součin v programovatelné AND matici.



Obr. 6 - 2 Možné zapojení OLMC buňky s registrem na výstupu



Obr. 6 - 3 Možné zapojení OLMC buňky jako kombinační výstup

## 6.2 Další vlastnosti GAL obvodů

U GAL obvodů je aplikován bezpečnostní bit, Security Fuse, který zabráňuje neautorizovanému **Bezpečnostní bit** čtení obsahu GAL obvodu. Jednou naprogramovaný bezpečnostní bit brání čtení pole propojek a neumožňuje další programování nebo verifikaci pole propojek. Bezpečnostní bit se dá vymazat pouze s vymazáním celého obsahu GAL obvodu. Tato akce se nazývá Bulk Erase, kdy **Security Fuse** se vymaže zapojení obvodu včetně bezpečnostního bitu.

Elektronický podpis, Electronic Signature, je paměťové programovatelné slovo a je to libovolná kombinace bitů navržená návrhářem obvodu, která se doporučuje používat k identifikaci obsahu

obvodu. Může to být číslo obsahu obvodu, číslo revize obsahu, datum programování, skladové číslo a pod. Data pamatována v elektronickém podpisu nemají žádný vliv na funkci obvodu a mají pouze význam informační. Elektronický podpis může být čten na programátoru obvodu verifikační procedurou bez ohledu na nastavení bezpečnostního bitu.

**Elektronický  
podpis**

**Electronic  
signature**

U obvodů GAL se také prvně objevila funkce přednastavení registru – Register Preload. Jedná se o zjednodušení testování GAL obvodů po naprogramování, ve kterém je realizován sekvenční obvod. Jedná se o možnost pomocí speciálního postupu nastavit paměťové elementy do testem definovaného stavu a z tohoto stavu pokračovat v testování sekvenčního obvodu. Funkce Register Preload se většinou zapíná pomocí zvýšeného napětí na definovaném pinu obvodu a data se do paměťových elementů zapisují sériově nebo paralelně.

**Přednastavení  
registru**

Funkce Power-Up Reset, nulování po náběhu napětí, zajišťuje nulování paměťových elementů GAL obvodu. Skutečný výstup je H nebo L úroveň, podle konfigurace OLMC, kde se volí aktivnost výstupu. Nulování paměťových elementů po náběhu napětí má význam u sekvenčních obvodů, kdy se tato funkce využívá k návrhu výchozího stavu sekvenčního obvodu.

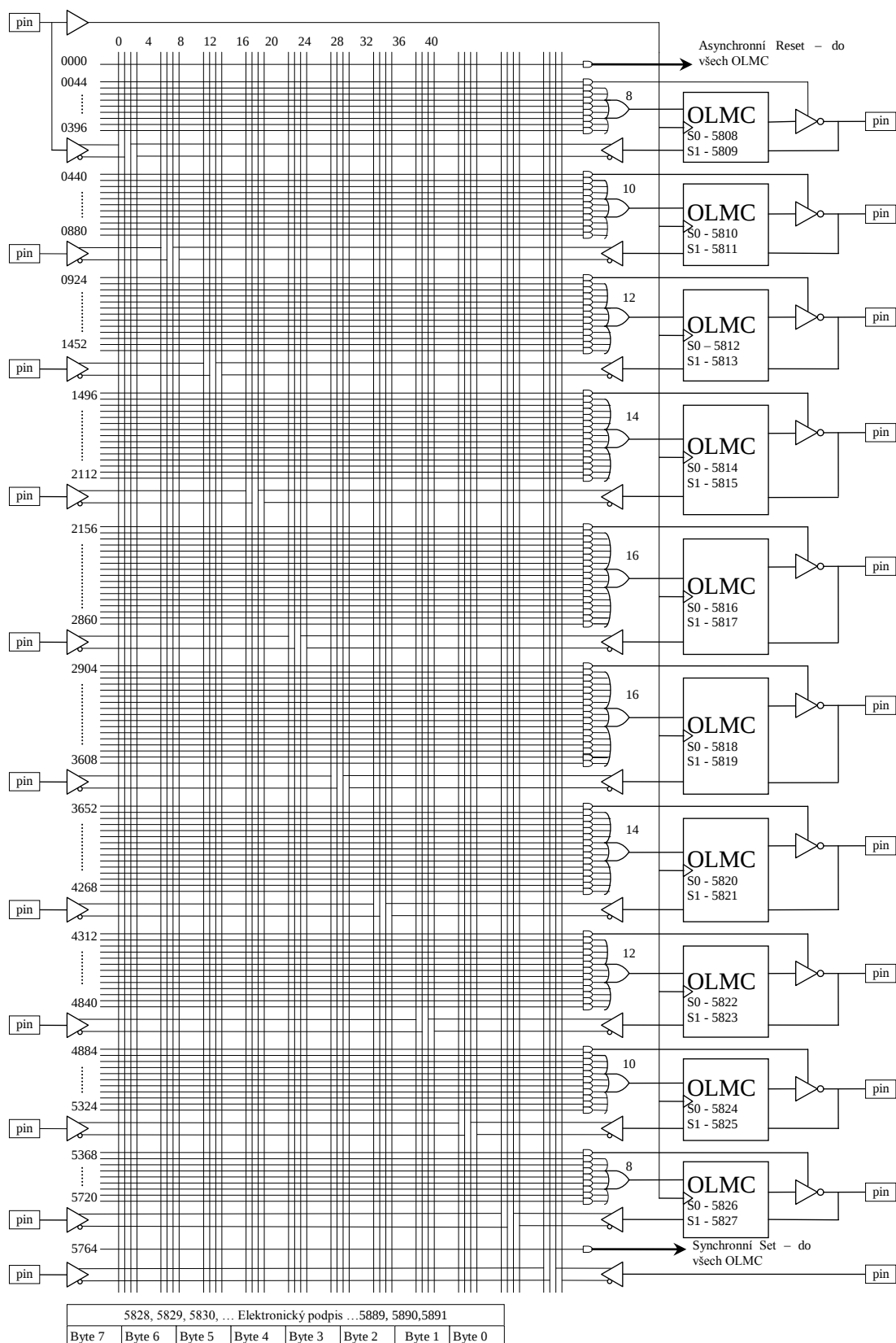
### 6.3 Zapojení obvodu GAL22V10

V praxi je možné se setkat s GAL obvody pod základním označení GAL16V8, GAL20V8, GAL22V10. Celkové zapojení obvodu GAL22V10 je na obr. 6 – 4 a používá OLMC buňku podle obr. 6 - 1. Obvod má 10 buněk. Počet součinnových termů pro každou OLMC buňku je jiný a pevně daný, viz obrázek. Krajní buňky mají 8 součinnových termů a směrem do středu se počet zvyšuje až na 16 součinnových termů. Jako hodinový signál pro paměťový element je použit signál z pinu obvodu. Nastavení a nulování každého paměťového elementu je odvozeno jako součinnový term a je společné pro všechny OLMC buňky.

**Přednastavení  
registru**

Vstupní signály do součinnové matice jsou jednak ze vstupních pinů a z OLMC buňky. V případě konfigurace OLMC buňky s paměťovým elementem jsou signály do součinnové matice napojeny na výstup paměťového elementu a v případě konfigurace jako kombinační výstup jsou signály do součinnové matice napojeny na pin obvodu.

Na obrázku jsou také zakresleny adresy jednotlivých programovatelných propojek. Adresu 0 má programovatelná propojka v součinnové matici vlevo nahoře a adresu 5 807 má propojka vpravo dole. Propojky s adresou 5 808 až 5 827 konfiguruje OLMC buňku a propojky 5 828 až 5 891 jsou určeny pro elektronický podpis. Obvod GAL22V10 má tedy 5 892 programovatelných propojek. Ruční vytváření logické sítě je nepředstavitelné, náročné, proto se již používají návrhové vývojové systémy.

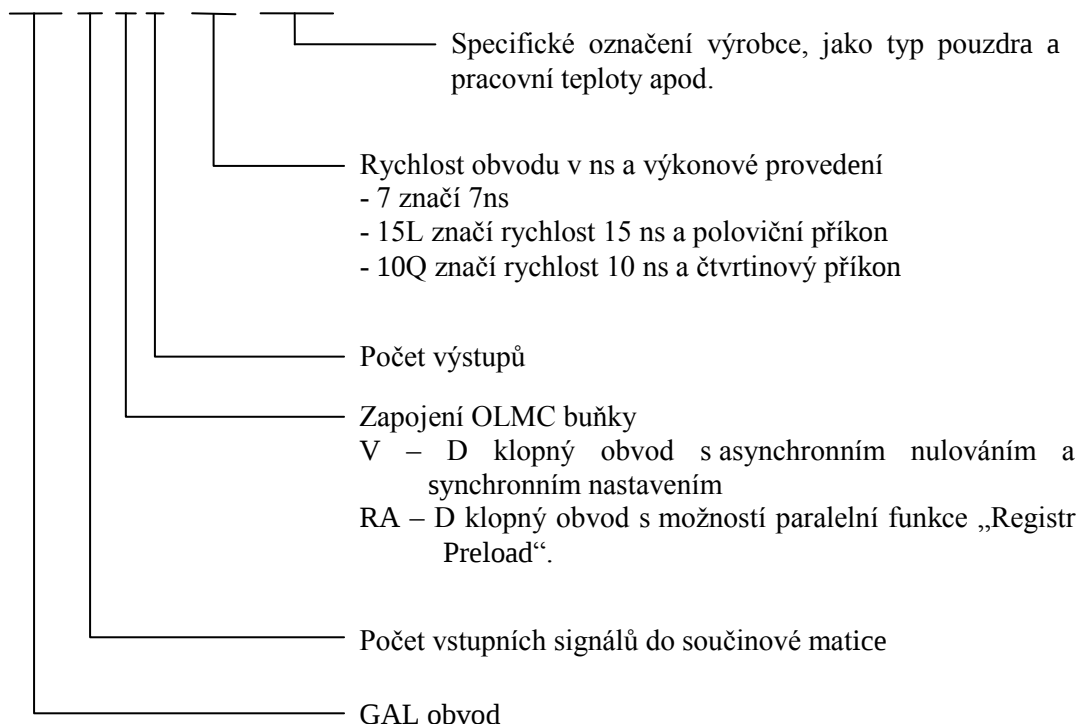


Obr. 6 - 4 Zapojení obvodu GAL22V10

## 6.4 Označování GAL obvodů

Značení GAL obvodů je obdobné jako PAL obvodů a je zde důležitý počet vstupů do součinové matice a počet výstupů. Organizace a význam jednotlivých částí značení je zřejmá z následujícího obrázku.

GAL 22 V 10 15L - xxxx



**Označování  
GAL obvodů**

Obr. 6 - 5 Značení GAL obvodů

## 6.5 Význam GAL obvodů

Význam GAL obvodů spočívá:

- Aplikace OLMC buňky. Protože OLMC buňka může být použita jako kombinační nebo paměťový výstup, potom jeden typ GAL obvodu nahrazuje několik typů PAL obvodů. Například, obvod GAL20V8 dokáže pinově i funkčně nahradit obvody PAL typu 14L8, 14H8, 14P8, 16L6, 16H6, 16P6, 18L4, 18H4, 18P4, 20L2, 20H2, 20P2, 20R8, 20RP8, 20R6, 20RP6, 20R4, 20RP4, 20L8, 20H8, 20P8.
- Z aplikace OLMC buňky vyplývá možnost použití pouze jednoho typu GAL obvodů pro všechny PLD prvky na desce plošného spoje. Je to zmenšení počtu typů součástek na desce plošného spoje.
- Zavedení opětovného programování má význam jednak pro návrháře ale i pro výrobce. Reprogramovatelnost GAL obvodů umožňuje pozdější zavádění změn zapojení GAL obvodů. Je to významný rozdíl vůči PAL obvodům. Reprogramovatelnost umožňuje také testování GAL obvodů ve výrobě.
- Aplikace elektronického podpisu, který umožňuje identifikaci obsahu GAL obvodů.
- Funkce Register Preload, možnost testování sekvenčních obvodů z testem definovaného stavu ve speciálních zařízeních, např. programátor.
- Funkce nulování paměťových elementů po náběhu napětí.

**Význam GAL  
obvodů**

**Otázky kap. 6**

- 6.1 Popište princip architektury GAL.
- 6.2 Popište funkci OLMC buňky.
- 6.3 Jaké jsou další vlastnosti GAL obvodů v porovnání s PAL obvody?
- 6.4 Jaký je význam GAL architektury?

**Odměna a odpočinek**

Další směr vývoje PLD prvků se ubíral k reprogramovatelnosti propojek obdobně jako u PROM pamětí. V roce 1984 společnost Altera Corp. uvádí na trh obvody řady EP300 jako EPLD – Erasable PLD, kde mazání bylo pomocí ultrafialovým zářením. Další směr v reprogramovatelnosti vede k elektrickému principu, obdobně jako u PROM pamětí.

V roce 1985 uvádí společnost Lattice Semiconductor Corp. novou řadu PLD obvodů pod označením GAL – Generic Array Logic. Jedná se hlavně o zavedení elektrického mazání obsahu, aplikace OLMC buňky a CMOS technologie, která vede ke snížení spotřeby elektrické energie. Nejznámější a dodnes používané obvody jsou GAL20V8 a GAL22V10.

Výrobci PLD obvodů se snažili o co nejlepší využití plochy křemíku integrovaného obvodu, a proto se objevuje nová architektura pod označením FPGA – Field Programmable Gate Array, která vychází z teorie polozákaznických obvodů - hradlových polí.

Průkopníkem této architektury byla společnost Xilinx v roce 1985. Tyto obvody společnost označovala jako LCA – Logic Cell Array. Základem je blok CLB, kde se vytváří jedna libovolná logická funkce o čtyřech proměnných. Při pohledu na křemík, obvod má uprostřed matici CLB bloků, které jsou na okraji obklopeny vstupně-výstupními buňkami. Vše je propojeno programovatelnou sítí, která umožňuje propojit podle požadavků jednotlivé bloky do logické sítě. Za další významnou vlastnost lze považovat realizaci propojek pomocí statické RAM paměti. Pole buněk statické RAM paměti, které řídilo propojky, bylo umístěno pod CLB buňkami. Samotné obvody byly energeticky závislé, proto při každém zapnutí napájení musel být nahrán jejich obsah z EPROM paměti nebo jiného zdroje. Tuto skutečnost lze považovat za výhodu i nevýhodu. Výhoda spočívá v rychlém principu naprogramování, zavedení nové modifikace zapojení. Časem se přidalo používání buněk statické RAM paměti jako klasické paměti k uchovávání informace. Nevýhodou je energetická závislost. Princip statické RAM paměti je používán u některých typů FPGA obvodů dodnes.

## 7. ZÁKLADNÍ STAVEBNÍ BLOKY PRVKŮ CPLD A FPGA

**Čas ke studiu:** 2 hodiny



**Cíl** Po prostudování této kapitoly budete umět

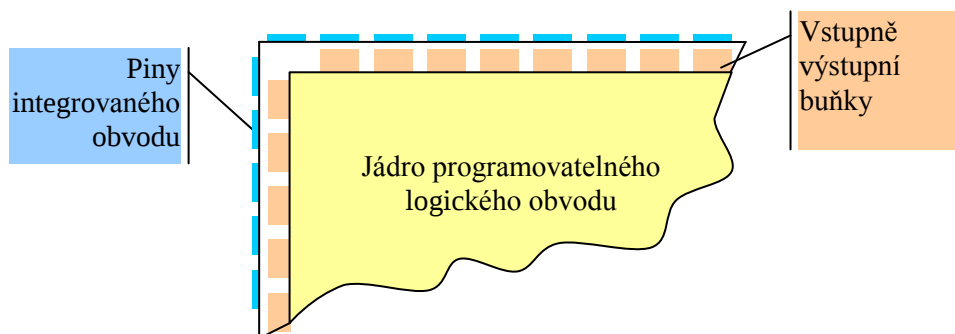


- Základní stavební bloky vyšších struktur programovatelných logických obvodů jako jsou CPLD a FPGA
- Vytváření  $\mathcal{B}$ -funkcí pomocí LUT tabulek

**Výklad**



Pokračováním PAL a GAL prvků jsou CPLD – Complex Programmable Logic Devices a FPGA – Field Programmable Gate Array obvody. Tyto obvody mají dnes mnoho společných vlastností, zapojení, které si jednotlivě objasníme. Principy zvyšování integrace vedly i k změně znázorňování obvodů, kdy se dává přednost pohledu na organizaci čipu v integrovaném obvodu, obr. 7 - 1.



**Nový princip  
znázorňování  
zapojení PLD  
prvku**

Obr. 7 - 1 Pohled na křemík PLD obvodu

Obvody obsahují vstupně-výstupní buňky a jádro obvodu. Jádro obvodu slouží k vytváření logické sítě a vstupně-výstupní buňky jej propojují na piny integrovaného obvodu. Vstupně-výstupní buňky jsou umístěny na okraji křemíku a jsou napojeny na jádro programovatelnou propojovací sítí.

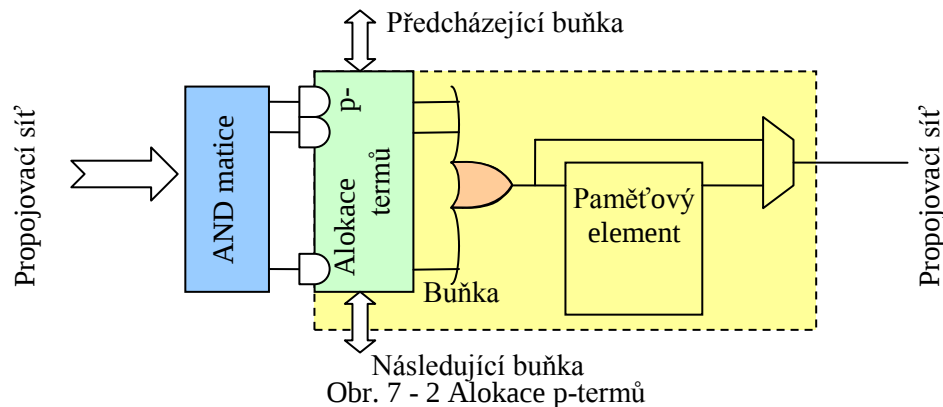
Programovatelné propojovací sítě se již detailně nerozkreslují pro svoji rozsáhlou a složitou strukturu. Nutno si uvědomit, že tyto sítě mají desetitisíce až statisíce programovatelných propojek a spojů různých vlastností. Jako vlastnosti je možno uvést délku spojů, hierarchické uspořádání spojů, možnost propojit každou druhou, čtvrtou buňku apod. Detailní znalost propojovací sítě pro návrháře není nutná, protože konečné propojení vždy provádí návrhový systém. Proto všechny detaily o propojovací sítí musí znát návrhový systém.

**Propojovací  
sítě PLD  
prvku**

## 7.1 Logická síť na principu AND a OR matice

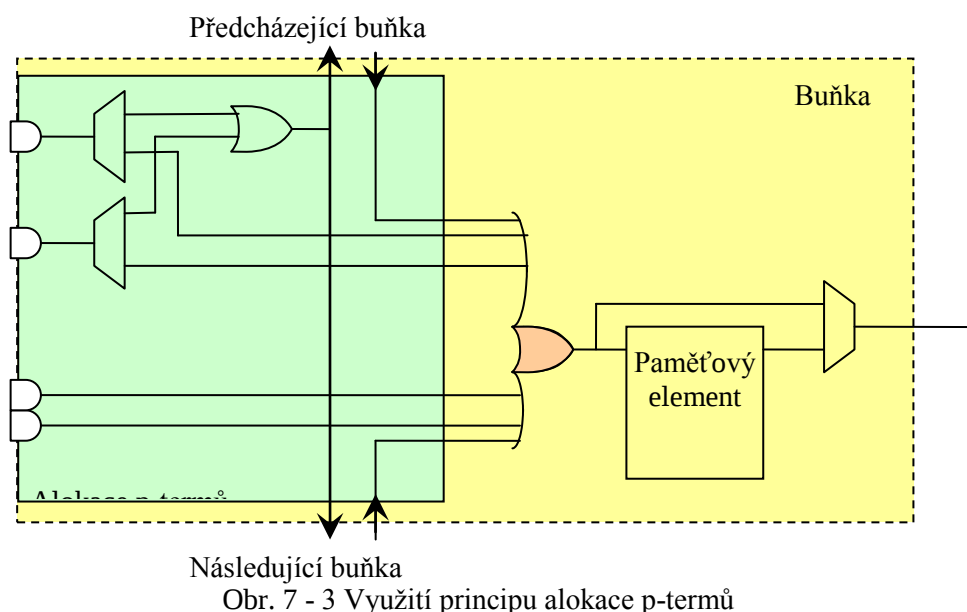
V těchto vyšších strukturách se převážně používá modifikované zapojení AND a OR principu včetně OLMC buňky, které jsou známé z GAL obvodů. Základem je programovatelná AND matice a pevná OR matice. Výjimečně se lze setkat s PLA principem. Pro úplnost nutno chápat, že vstup a výstup uvedených zapojení je vyveden do programovatelné propojovací sítě. Počet vstupních signálů do součinné matice jsou desítky signálů. Buňky jsou v porovnání z GAL zapojením modifikovány o další vlastnosti, které si uvedeme dále. Znárodnované obrázky se budou týkat jedné buňky, ve skutečném obvodu jsou tyto buňky aplikovány vícenásobně.

**Aplikace AND a OR matice**

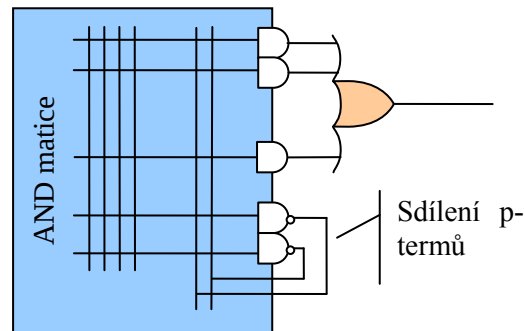


PAL a GAL obvody mají pevně přiřazený počet součinných termů na každé OR hradlo. U CPLD obvodů je většinou počet součinných termů na OR hradlo proměnlivý pomocí bloku alokace p-termů, obr. 7 - 2. Blok alokace p-termů umožňuje volné součinné hradla, která nebyla použita pro daný součet, použít v sousedních buňkách. Přenos nepoužitých součinných termů do jiných součtových hradel lze provést přímo nebo pomocí pomocného součtu v bloku. Princip alokace p-termů a jejich použití pomocí součtového hradla je na obr. 7 - 3. Základem je aplikace demultiplexerů na výstupu součinného hradla, který umožňuje použít součinný výstup na součet v dalším pomocném hradle nebo v přímo v součtovém hradle buňky. Výstup pomocného součtového hradla je propojen na sousední buňky, které jej mohou využít. Ne u všech součinných hradel musí být aplikován demultiplexer, vše závisí na konkrétním typu. Alokace p-termů umožňuje realizaci rozsáhlejších výrazů, které obsahují hodně součinných termů. Tento princip může vést i k úspoře logických prvků a ke zmenšení zpoždění logické sítě.

**Alokace p-termů**



**Možné zapojení principu alokace p-termů**

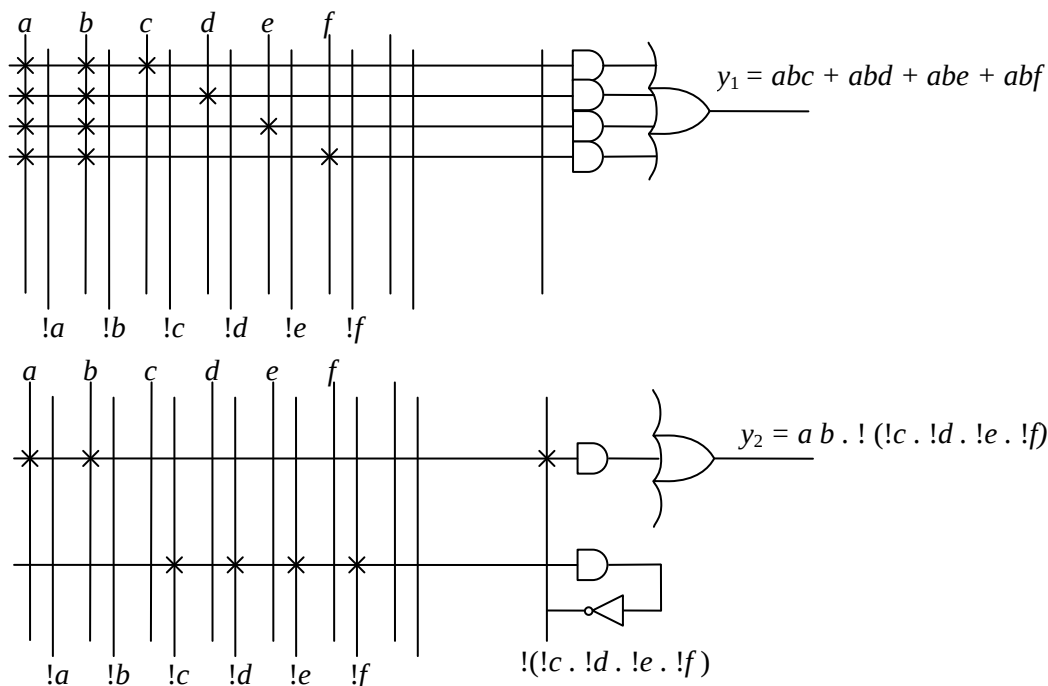


Obr. 7 - 4 Sdílení součinných termů

Dalším zapojením, se kterým se můžeme setkat je sdílení součinných termů, obr. 7 - 4. Toto zapojení se využívá u některých minimalizací součtových výrazů, kdy lze ze součinných termů vytknout společnou část, výraz  $y_1$  (7.1) se dá upravit na výraz  $y_2$  (7.2) a na součet aplikovat De-Morganův zákon, který mění operaci logického součtu na logický součin

$$y_1 = abc + abd + abe + abf \quad (7.1)$$

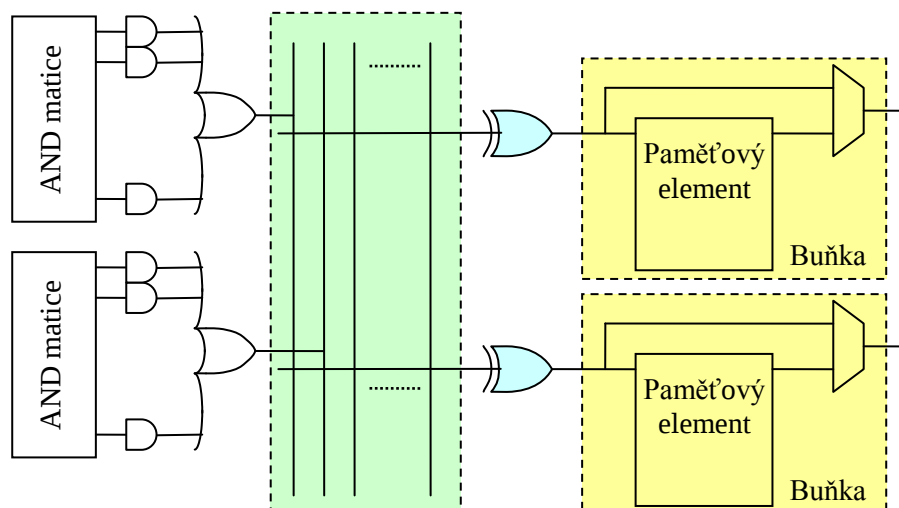
$$y_2 = a.b(c + d + e + f) = ab(\overline{\overline{c.d.e.f}}) \quad (7.2)$$



Obr. 7 - 5 Využití sdílení součinných termů

Význam sdílení součinných termů je v realizaci, kdy sdílení součinných termů vede ke snížení počtu použitých součinných hradel. Výraz  $y_1$  lze realizovat pomocí čtyř součinných hradel, ale k realizaci výrazu  $y_2$  stačí pouze dvě součinné hradla, obr. 7 - 5. Při realizaci výrazu  $y_2$  došlo k úspoře součinných hradel, které mohou být alokovány do jiných buněk, například principem alokace p-termů. Při větším počtu součinných termů ve výrazu  $y_1$  je úspora větší, protože počet součinných hradel pro výraz  $y_2$  je vždy dva.

Jiné řešení sdílení je na obr. 7 - 6. Výstupy součtového hradla jsou vedeny do další programovatelné matice s definovaným typem hradla, např. XOR. Tato doplňující programovatelná síť umožňuje vytvářet další operace, a tím rozšířit možnosti pro realizaci kombinační sítě. Minimalizace pomocí XOR hradel jsou náročné a lze je popsat v literatuře [Frištacký, Kolesár, Kolenička, Hlavatý 1986].

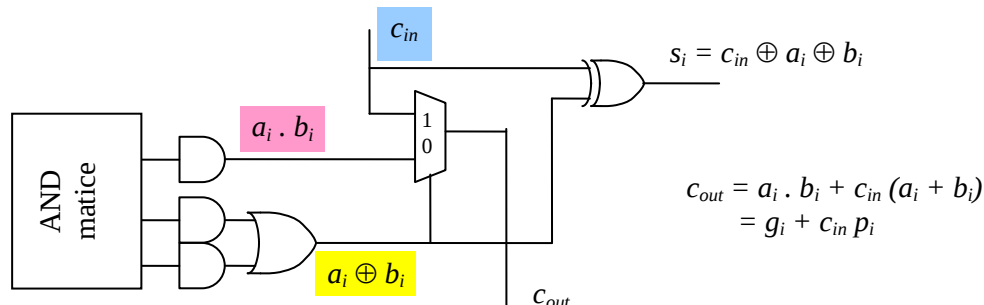


**Aplikace  
sdílené XOR  
matice**

Obr. 7 - 6 Doplňující propojovací síť s hradly XOR

Pod pojmem přenosová logika se chápou zapojení, která umožňují vytvářet iterační obvody. V PLD prvcích se setkáváme s přizpůsobenými zapojeními, která využívají pouze jeden přenosový bit a jsou navržena hlavně pro aplikaci binárního součtu nebo rozdílu.

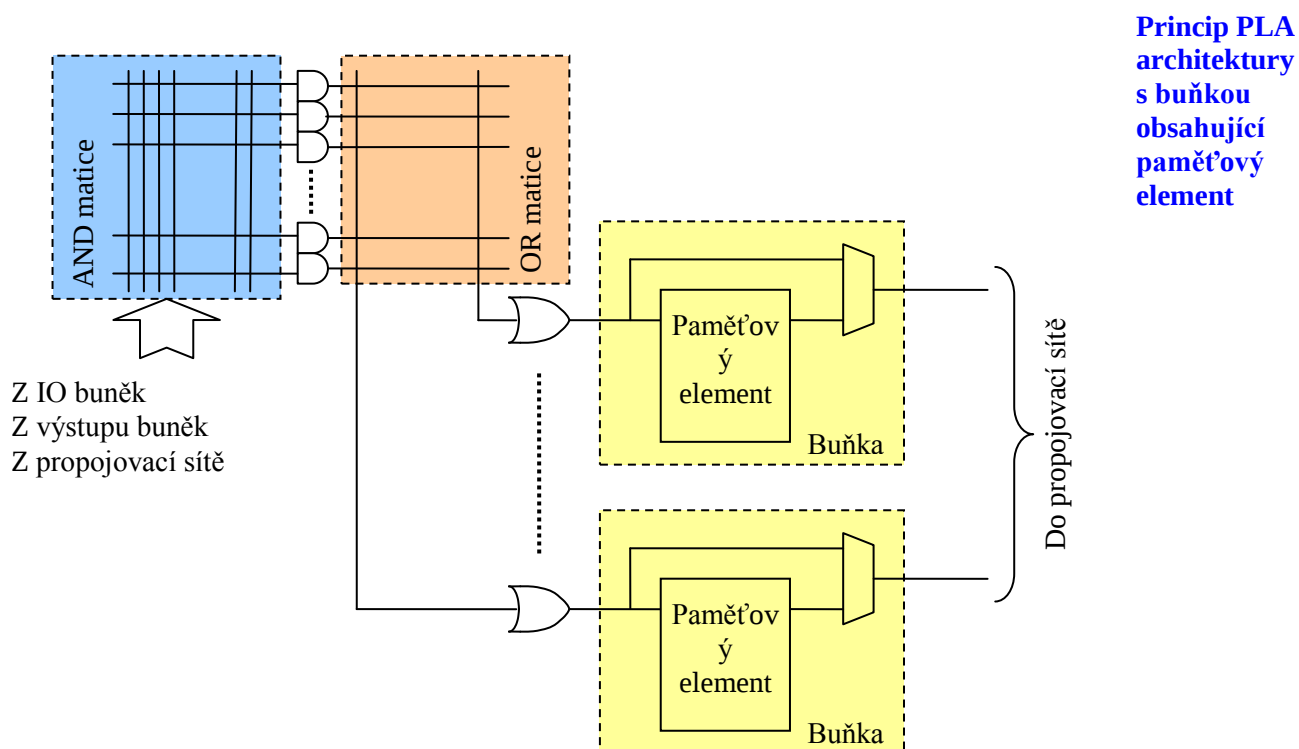
**Iterační  
obvody a  
přenosová  
logika**



Obr. 7 - 7 Zapojení buňky pro 1 bit aritmetického součtu

Zapojení jedné buňky 1 bitové binární sčítačky je na obr. 7 - 7 a je realizováno na základě generačního a propagačního přenosu. Buňka generuje přenos  $g_i$  rovný 0 nebo 1, když  $a_i = b_i = 0$  nebo  $a_i = b_i = 1$ . Tato situace je vyjádřena součinem  $a_i \cdot b_i$  na datovém vstupu multiplexeru a podmínkou  $a_i \text{ xor } b_i = 0$  na řídicím vstupu multiplexeru. Propagační přenos  $p_i$  značí, že buňka přenáší vstupní přenos na výstup buňky  $c_{out} = c_{in}$ , když  $a_i \neq b_i$ . Tato situace je vyjádřena signálem  $c_{in}$  na datovém vstupu multiplexeru a podmínkou  $a_i \text{ xor } b_i = 1$  na řídicím vstupu multiplexeru.

**Aplikace na  
jednobitovou  
binární  
sčítačku**

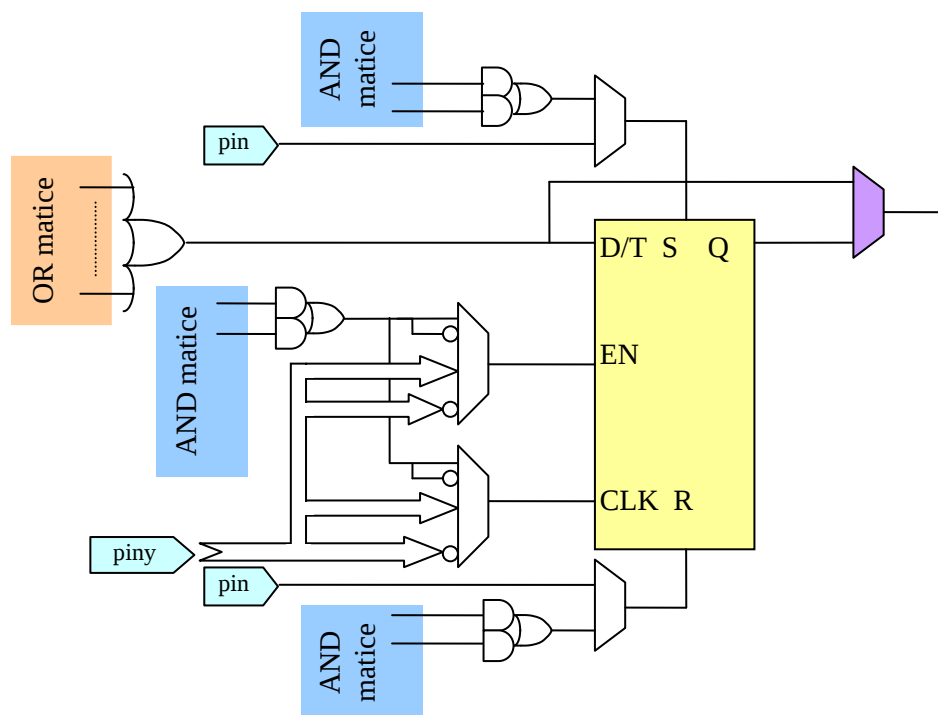


Obr. 7 - 8 Základní zapojení CPLD prvku pomocí struktury PLA

Uplatnění PLA principu pro realizaci logické sítě je na obr. 7 - 8. Za OR hradlo je zapojena buňka s paměťovým elementem a multiplexerem, který dovoluje konfigurovat výstup buňky jako kombinační nebo registrový. Programovatelná AND i OR matice umožňuje lepší využití součinnových hradel, protože pro každý součet se použije vždy potřebný počet součinnových hradel a odpadá potřeba bloku alokace součinnových termů.

## 7.2 Paměťový element

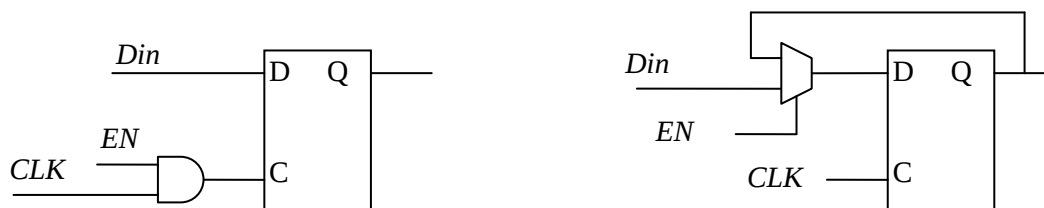
Za kombinační sítě buňky následuje paměťový element s multiplexerem jako v OLMC buňce GAL obvodů, viz předcházející obrázky. Multiplexer umožňuje buňku použít jako kombinační obvod nebo jako buňku s paměťovým elementem. V PLD prvcích se používá pojem paměťový element jako všeobecné označení, protože představuje různé typy klopných obvodů s různými vlastnostmi. Nejčastěji se používá D klopný obvod, popřípadě D/T klopný obvod. Jiné typy klopných obvodů se vyskytují ojediněle.



Obr. 7 - 9 D/T klopný obvod jako paměťový element

Paměťový element jako D/T klopný obvod je na obr. 7 - 9. Tento element buď pracuje jako D nebo T klopný obvod, výběr se provádí programovatelnými propojkami. Součástí paměťového elementu je i povolovací vstup, který povoluje synchronní činnost klopného obvodu. Před hodinovým a povolovacím vstupem bývají předřazeny multiplexery pro výběr signálů z možných zdrojů. Zdrojem těchto signálů může být logická síť, kterou navrhuje uživatel nebo signál odvozený od pinů PLD prvků. Aplikace multiplexeru na hodinovém vstupu klopného obvodu umožňuje i volit, na kterou hranu bude klopný obvod reagovat, vzestupnou nebo sestupnou.

**D/T klopný obvod jako paměťový element**



Obr. 7 - 10 Povolovací vstup D klopného obvodu

Použití povolovacího vstupu je možné chápat jako zapojení na obr. 7 - 10. Hodinový signál pro D klopný obvod je vytvářen na součinném hradle. Tento princip se běžně používá při zápisu dat do registru. Jiná modifikace je uplatněním multiplexeru na D vstupu klopného obvodu. V tomto případě, je-li signál  $EN=0$  se výstup  $Q$  opětovně přivádí na  $D$  vstup. Jedná se o NOP operaci, výstup se nemění. Je-li signál  $EN=1$  provede se zápis nové informace ze signálu  $D_{in}$ .

**Povolení hodin pro D/T klopný obvod**

Paměťový element může reagovat na hranu nebo úroveň hodinového signálu. D klopný obvod, který reaguje na hranu se nazývá D-FF, D – Flip Flop a D klopný obvod reagující na úroveň se nazývá D – Latch. T klopný obvod reaguje pouze na hranu (flip flop).

Dalšími vstupy paměťového elementu jsou nastavení - set a nulování - reset výstupu  $Q$ . Příslušné signály se vytvářejí buď logickou sítí nebo jsou odvozeny od vyhrazených pinů

**Nulování a nastavení pro D/T klopný obvod**

integrovaného obvodu. Uvedené signály mohou být vytvářeny pro každý paměťový element nebo pro více paměťových elementů, které vytvářejí určitý blok. Společné nastavení a nulování pro blok paměťových elementů je častější řešení.

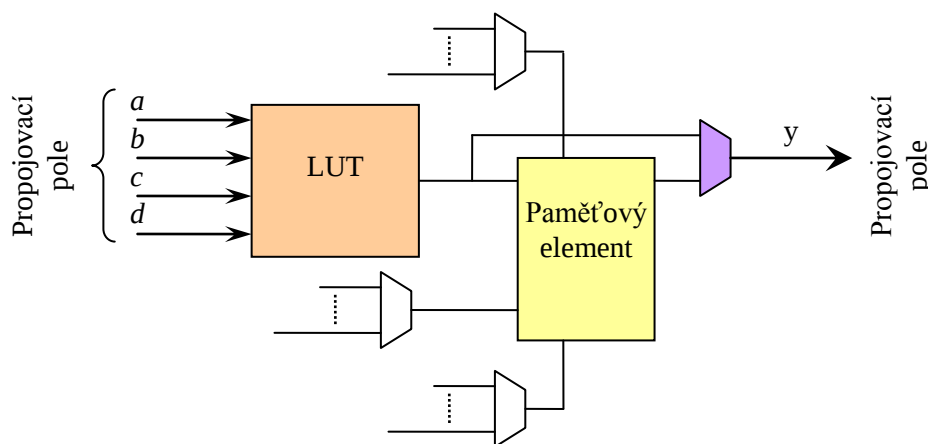
Nastavení a nulování může být synchronní nebo asynchronní. Asynchronní nulování a nastavení není závislé na hodinovém signálu, naproti synchronní se provede pouze s hodinovým signálem. Akce nulování a nastavení má vždy přednost před datovým vstupem D/T klopného obvodu. Existují i zapojení paměťového elementu bez nulování a nastavení. V tomto případě se uplatňuje akce nulování po náběhu napětí, power up-reset.

**Synchronní a  
asynchronní  
nulování a  
nastavení**

### 7.3 Logická síť na principu LUT tabulek

Buňky na základě LUT tabulek – Look Up Table vůbec nesouvisí s teorií AND a OR matice. Základní zapojení buňky s LUT tabulkou je na obr. 7 - 11. Buňka s LUT tabulkou je logické zapojení, které dokáže pracovat jako kombinační síť nebo kombinační síť s paměťovým elementem.

**LUT tabulka**



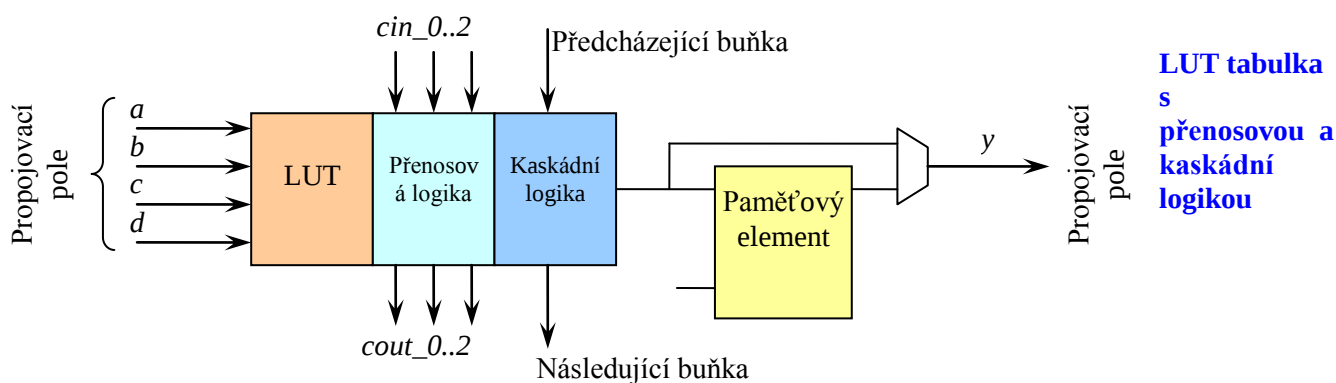
Obr. 7 - 11 Základní zapojení buňky s LUT tabulkou

Základ tvoří LUT tabulka, která odpovídá logické kombinační síti a dokáže realizovat libovolnou  $\mathcal{B}$ -funkci o čtyřech vstupních proměnných. Počet vstupních proměnných signálů do LUT tabulky bývá malý, typicky 4 signály. Buňka dále obsahuje paměťový element a multiplexer na výstupu buňky, který umožňuje konfigurovat výstup jako kombinační, přímo z LUT tabulky nebo jako výstup s paměťovým elementem.

Skutečné zapojení, a tím i funkci buňky definují programovatelné propojky. Vstupy a výstup buňky jsou zapojeny na programovatelnou propojovací síť. Jako paměťový element se převážně používá D klopný obvod nebo D/T klopný obvod, skutečný typ závisí na konkrétním typu PLD prvku.

Mezi základní modifikace zapojení buňky s LUT tabulkou je doplnění zapojení buňky o přenosovou logiku a kaskádní logiku, obr. 7 - 12. Zapojení pro přenosovou logiku umožňuje vytvářet iterační obvody, hlavně binární aritmetický součet nebo rozdíl. U LUT tabulky pro přenosovou logiku může být počet bitů určených pro přenos i více bitů než jeden. Počet vstupních proměnných podílejících se na vytváření výstupní a přenosové funkce se většinou zmenšuje na dvě vstupní proměnné. Signály pro přenosové a kaskádní zapojení se propojují přímo mezi buňkami, mimo propojovací síť. Takovéto řešení propojení dosahuje menší zpoždění signálů.

**Iterační  
obvody a LUT  
tabulka**



Obr. 7 - 12 Zapojení přenosové a kaskádní logiky v LUT buňce

Kaskádní zapojení je určeno pro realizaci součinu nebo součtu mnoha libovolných  $\mathcal{B}$ -funkcí (7.3) a (7.5). Jedná se o výrazy, kdy každá buňka vytváří svoji  $\mathcal{B}$ -funkci  $F_i$ , které se potom logicky násobí nebo sčítají.  $\mathcal{B}$ -funkce  $F_i$  mohou mít různé nebo stejné vstupní proměnné v závislosti na aplikaci. Jednou variantou výrazu (7.3) je, že  $\mathcal{B}$ -funkce  $F_i$  jsou vlastně logický součin různých vstupních proměnných, čímž získáme logický součin o  $n$  proměnných, (7.4). Jinou aplikací výrazu (7.3) je využití pro KNF – součinnovou normálovou formu.

$$y = y_{(n/4)-1} \text{ and } \dots \text{ and } y_1 \text{ and } y_0 =$$

$$F_{(n/4)-1}(x_{n-1}, x_{n-2}, x_{n-3}, x_{n-4}) \text{ and } \dots \text{ and } F_1(x_7, x_6, x_5, x_4) \text{ and } F_0(x_3, x_2, x_1, x_0) \quad (7.3)$$

$$y = x_{n-1} \text{ and } x_{n-2} \text{ and } x_{n-3} \dots \text{ and } x_2 \text{ and } x_1 \text{ and } x_0 \quad (7.4)$$

$$y = y_{(n/4)-1} \text{ or } \dots \text{ or } y_1 \text{ or } y_0 =$$

$$F_{(n/4)-1}(x_{n-1}, x_{n-2}, x_{n-3}, x_{n-4}) \text{ or } \dots \text{ or } F_1(x_7, x_6, x_5, x_4) \text{ or } F_0(x_3, x_2, x_1, x_0) \quad (7.5)$$

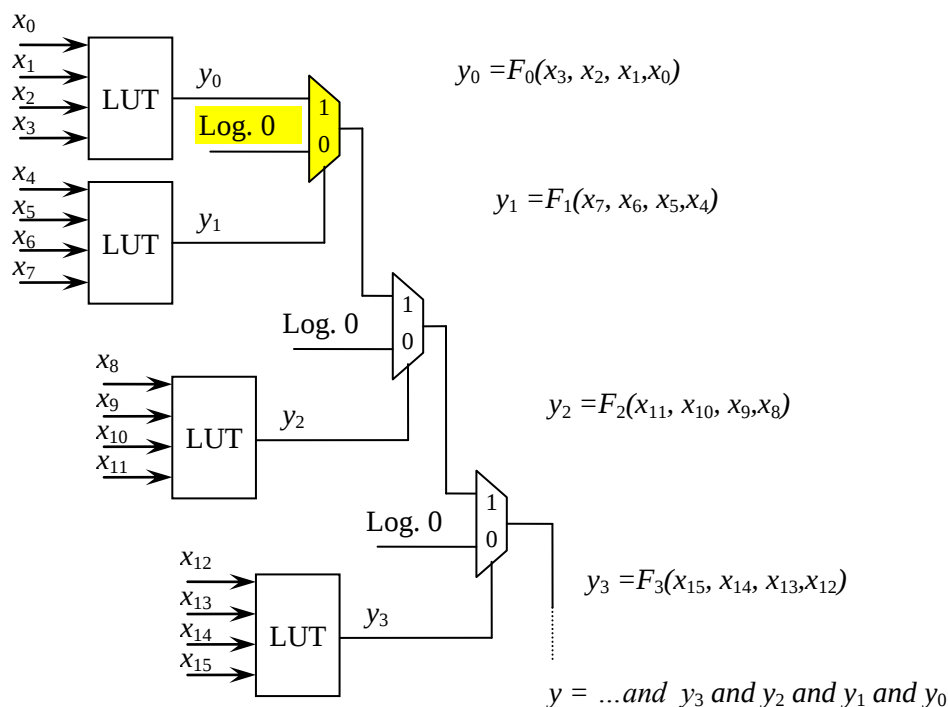
$$y = x_{n-1} \text{ or } x_{n-2} \text{ or } x_{n-3} \dots \text{ or } x_2 \text{ or } x_1 \text{ or } x_0 \quad (7.6)$$

Naproti tomu, výraz (7.5) může být modifikován na logický součet o  $n$  proměnných (7.6), kdy jednotlivé  $\mathcal{B}$ -funkce  $F_i$  jsou logický součet. Jiná modifikace je použití pro DNF - součtovou normálovou formu.

LUT tabulka  
s  
přenosovou a  
kaskádní  
logikou

Kaskádní  
logika

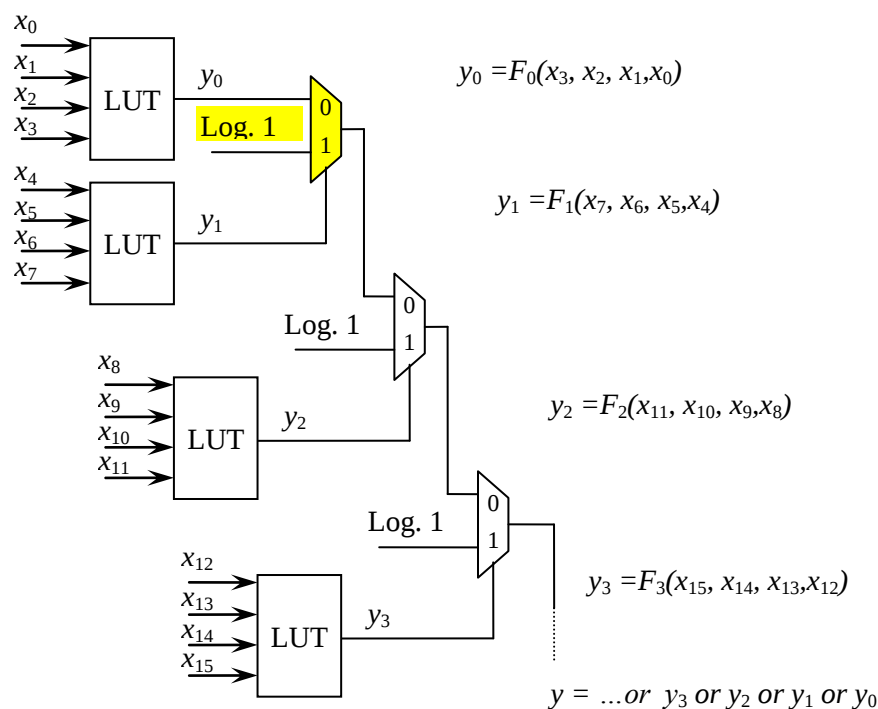
Kaskádní  
logika s  
operací  
logického  
součtu



Obr. 7 -13 Kaskádní zapojení logického součinu

**Zapojení  
kaskádního  
součinu**

Kaskádní zapojení se vytváří pomocí multiplexerů, pomocí kterých se vytváří součin nebo součet, obr. 7 – 13 a obr. 7 – 14. Pro vytváření logického součinu pomocí multiplexeru se využívá definice logického součinu „Logický součin je roven log. 1, když oba operandy jsou současně rovny log. 1.“. Pro vytváření logického součtu se využívá definice „Logický součet je roven log. 1, pokud aspoň jeden operand má hodnotu log. 1“, literatura [Diviš, Chmelíková a Zdrálek 2005].

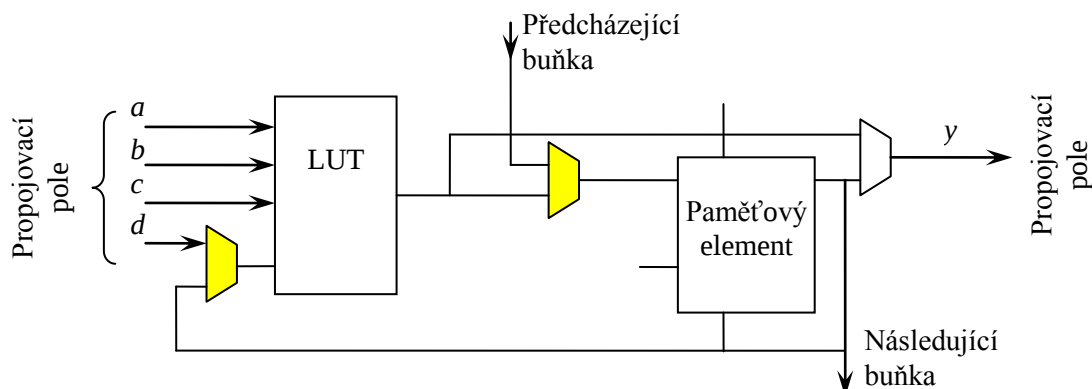


Obr. 7 -14 Kaskádní zapojení logického součtu

**Zapojení  
kaskádního  
součtu**

Jiná modifikace zapojení LUT buňky je, že výstup klopného obvodu je přiveden přes multiplexer na vstup LUT tabulky místo jedné vstupní proměnné, obr. 7 - 15. Tato volba zapojení pomáhá při realizaci sekvenčních obvodů, kdy může dojít k zmenšení zpoždění kombinační sítě vytvářející následující stav, a tím k možnému zvýšení hodinové frekvence.

**Propojení výstupu D klopného obvodu na vstup LUT**



Obr. 7 - 15 Modifikace zapojení buňky s paměťovým elementem

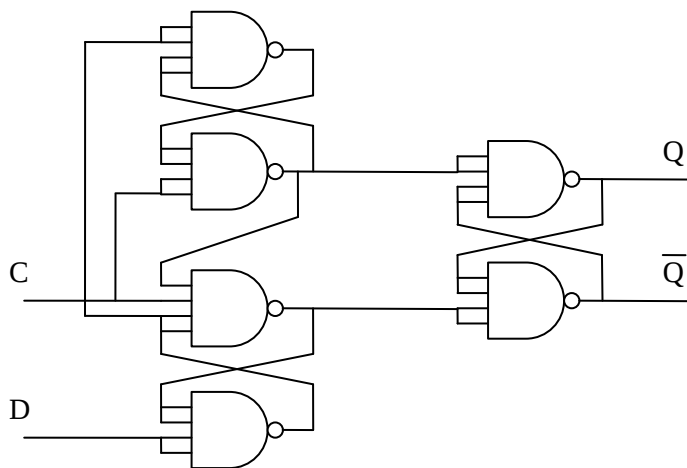
Další modifikace spočívá v možnosti propojit klopné obvody sousedních buněk jako posuvný registr. Možnost použít LUT tabulku k vytváření kombinačního obvodu zůstává. Uvedená zapojení se dají realizovat propojením přes propojovací síť, ale toto řešení má menší zpoždění.

**Propojení D klopných obvodů jako sériový registr**

## 7.4 Buňka jako hradlo

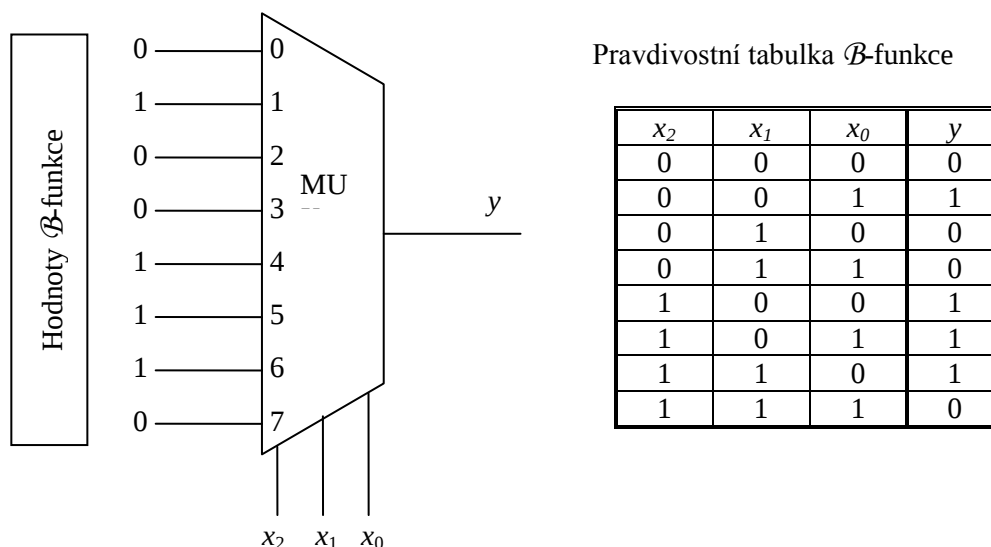
U FPGA obvodů se objevuje řešení, kdy buňka má velmi jednoduché zapojení. Celkové zapojení obvodů pak používá jenom jeden typ buňky nebo několik typů. Buňka může obsahovat zapojení pouze kombinačního obvodu, pouze paměťový element nebo kombinaci obou typů. Jako příklad, kdy obvod obsahuje pouze kombinační buňku, lze uvést buňku jako hradlo NAND o 4 vstupech. V tomto případě se všechny prvky logické sítě musí sestavit pouze z tohoto hradla.

**Buňka jako hradlo**



Obr. 7-16 Zapojení D klopného obvodu pomocí hradel

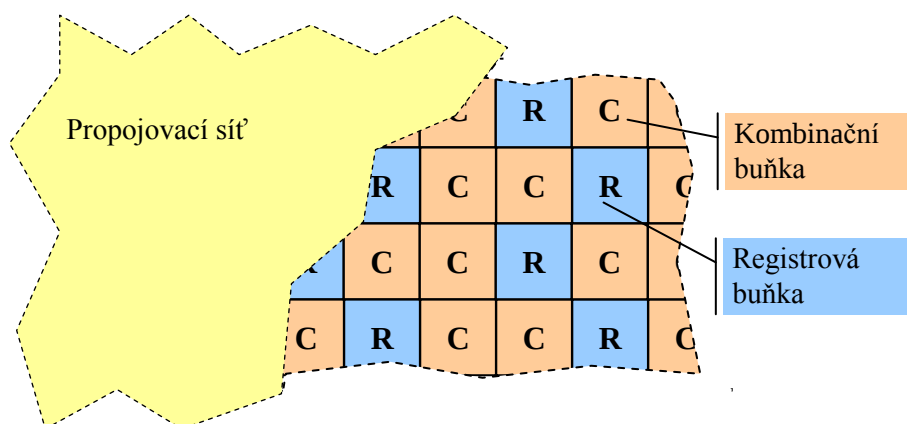
Potřebné paměťové elementy je také nutno sestavovat z těchto hradel, obr. 7-16. Toto sestavení je většinou již dáno výrobcem jako element knihovny. Sestavení kombinační logické sítě není problém.



Obr. 7-17x Buňka jako multiplexer

Jiné řešení kombinační buňky je zapojení, které dokáže realizovat libovolnou  $\mathcal{B}$ -funkci o definovaném počtu proměnných, například 3 vstupních proměnných, obr. 7 - 17. Použití této buňky jako paměťového elementu není možné. Paměťový element ani nelze sestavit vhodným zapojením kombinačních buněk, jedná se čistě pouze o kombinační buňku. Potom obvod také obsahuje buňky s paměťovým elementem a jiné použití není možné.

**Buňka jako multiplexer**



Obr. 7 – 18 Uspořádání PLD prvku principem moře hradel

Konečný PLD prvek obsahuje vhodný počet kombinačních a paměťových buněk. Příklad pohledu na čip je na obr. 7 – 18 a jedná se o uspořádání moře buněk – sea of gates. Jedná se o uspořádání, kdy jednotlivé buňky jsou těsně vedle sebe a propojovací síť je nad nimi. Jedná se o uspořádání v prostoru.

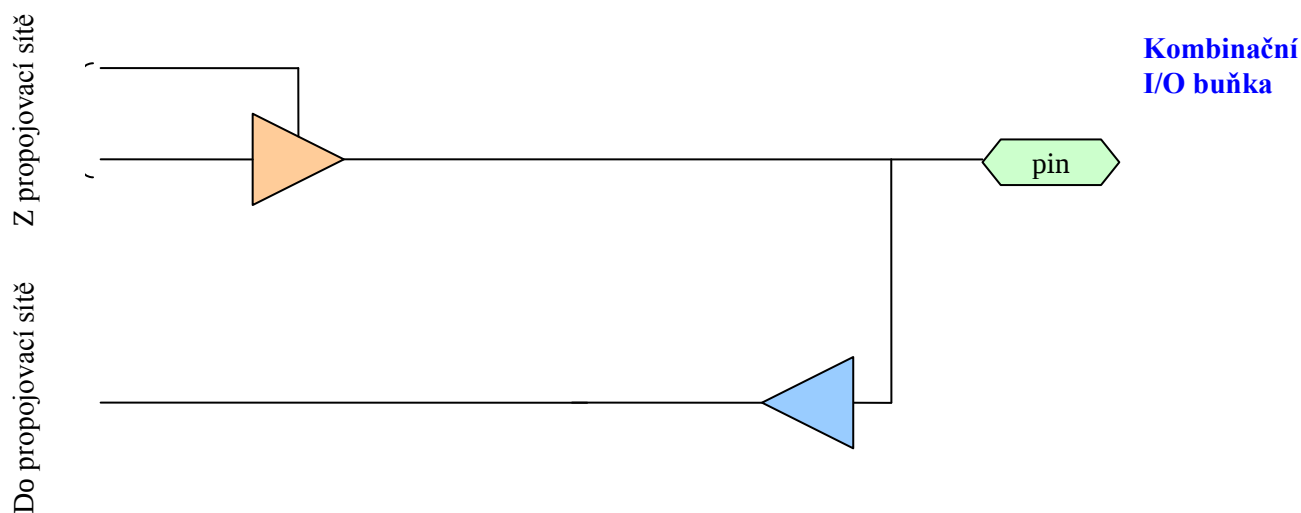
**PLD prvek jako moře hradel**

## 7.5 Vstupně-výstupní buňka

Vstupně-výstupní buňka je často označována jako IO buňka – Input Output Cell, která propojuje interní logiku obvodu s okolím PLD prvku. IO buňka může být pouze kombinační nebo s paměťovými elementy. Zapojení pouze kombinační IO buňky je na obr. 7 - 19. IO buňka je obousměrná. To značí, že ji lze konfigurovat buď jako vstupní, dále jako výstupní nebo jako

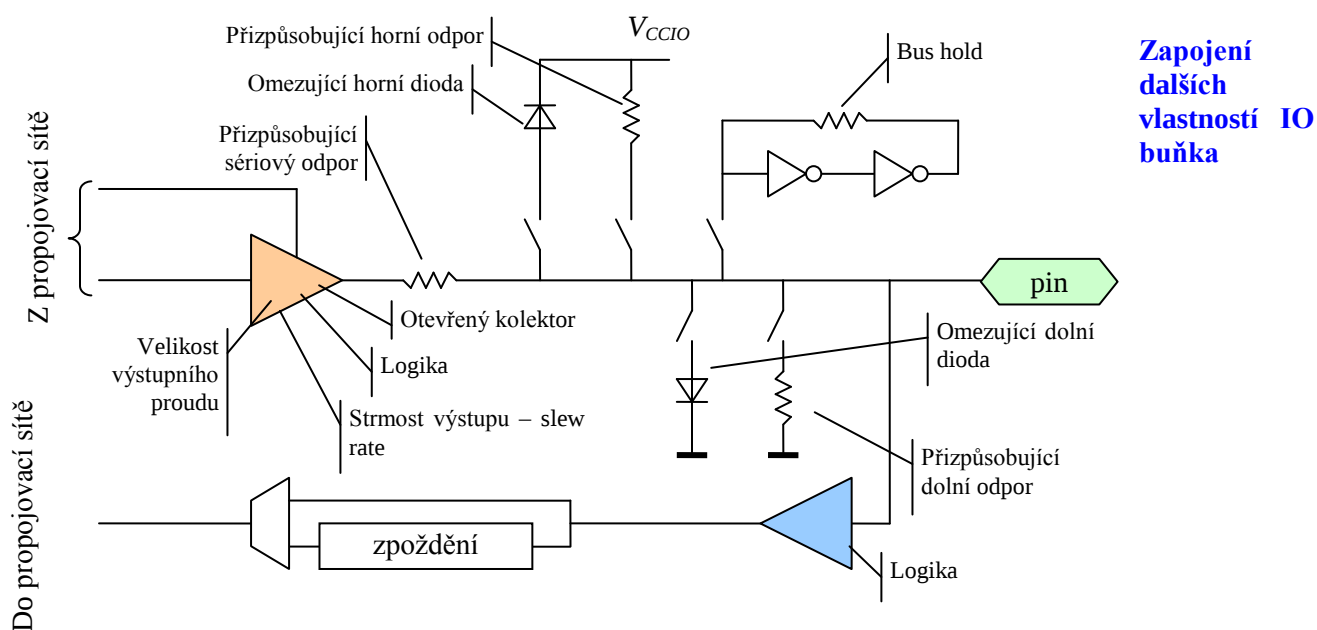
obousměrnou – vstupně-výstupní. Výstupní budič je proto třístavový a jeho stav je řízen povolovacím signálem z logiky.

Aplikace jako vstupní buňky je dáno pouze tím, že povolovací signál třetího stavu bude aktivní, výstupní budič bude v třetím stavu. IO buňka je napojena na interní propojovací síť, a tím je umožněn velký výběr zdrojů pro výstup. Při propojování IO buněk na interní logiku se můžeme setkat s různými omezeními, které závisí na konkrétním PLD prvku.

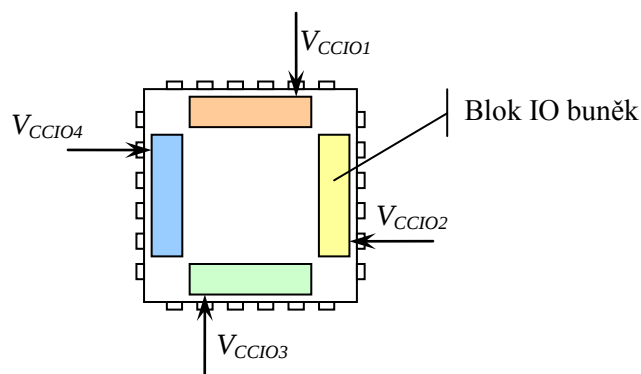


Obr. 7 - 19 Základní IO buňka bez paměťových elementů

Skutečné zapojení IO buňky je mnohem složitější a může být doplněno o obvody uvedené na obr. 7 - 20. Jedná se o obvody a vlastnosti, které určují výsledné vlastnosti IO buňky na pinu integrovaného obvodu. U skutečné IO buňky mohou být aplikovány pouze některé obvody nebo vlastnosti. Popis jednotlivých vlastností je v samostatné kapitole.



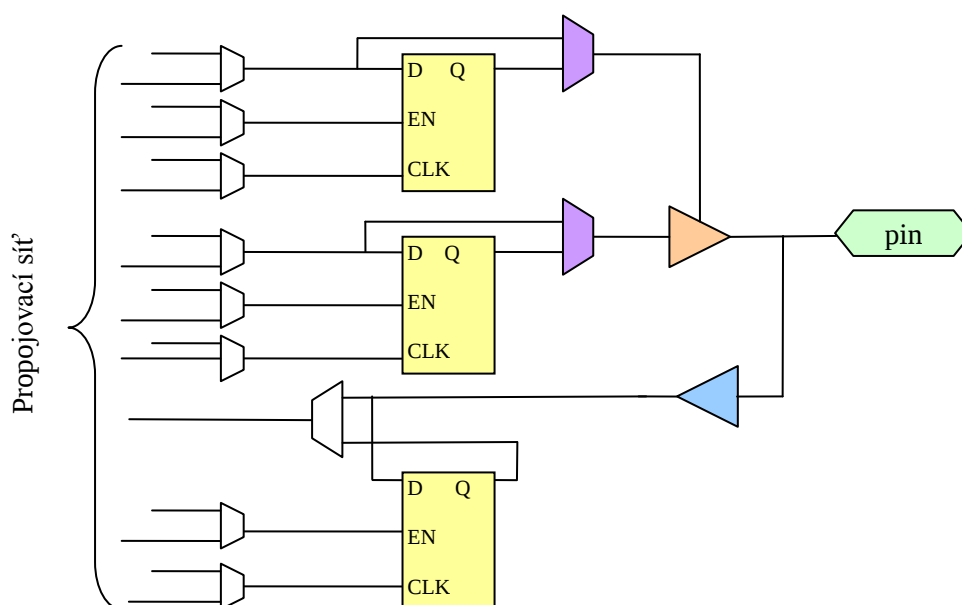
Obr. 7 - 20 Zapojení IO buňky s dalšími možnými obvody



**Bloky  
buněk**      **I/O**

Obr. 7 - 21 Vytváření IO bloku a přiřazení napájecího napětí

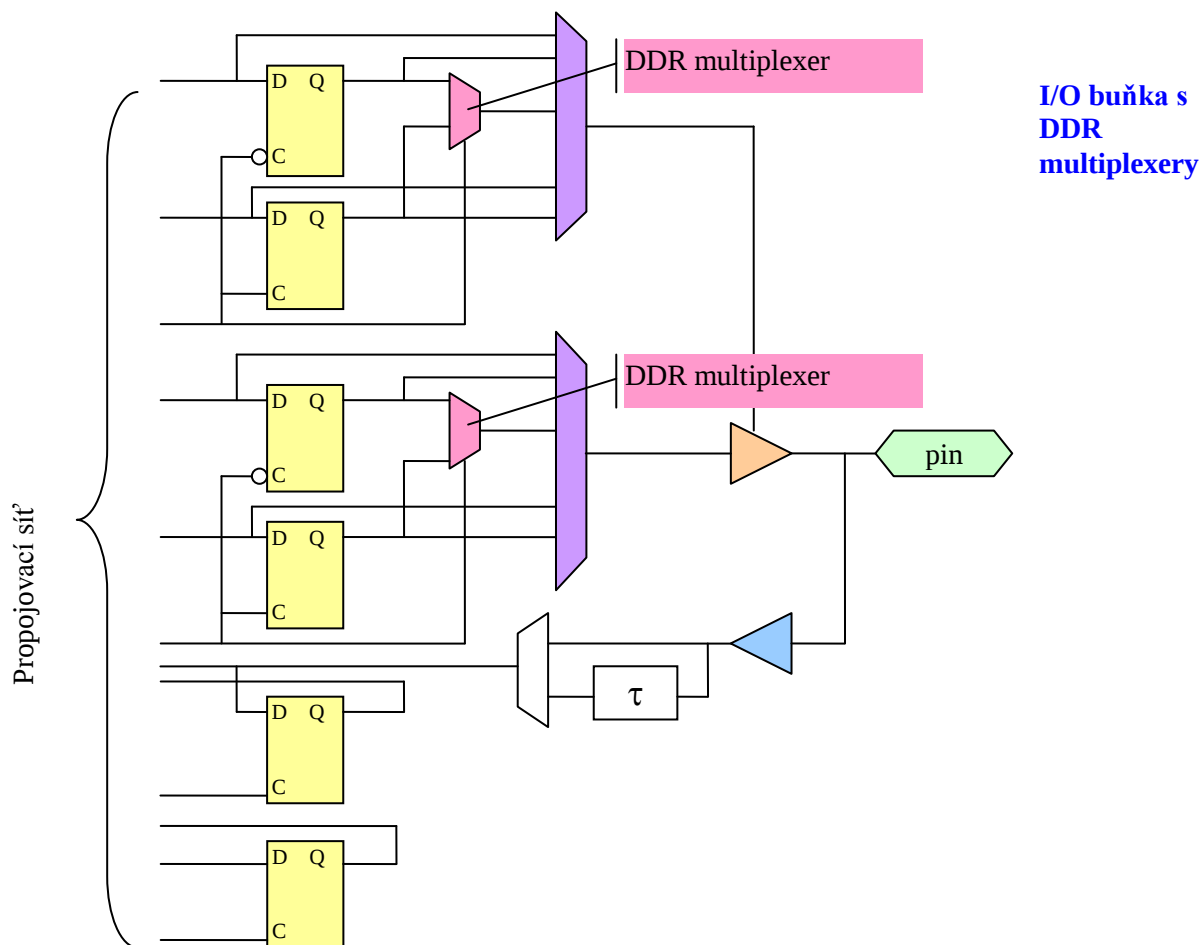
Číslicové systémy pracují s různými napěťovými definicemi úrovní L a H a různou definicí překlápěcího napětí – threshold level  $V_{TH}$ . Tyto definice se slučují pod pojmem logika. Vstupně-výstupní buňku je možné konfigurovat pro různé logiky. Proto vstupně-výstupní buňky se slučují do bloků a celému bloku se definuje jedna logika. Tyto bloky mají samostatné napájecí napětí  $V_{CCIO}$ , které je vyvedeno na samostatný pin obvodu, obr. 7 - 21. Hodnota tohoto napájecího napětí je součástí definice logiky, ve které bude tento blok pracovat. Počet vstupně-výstupních buněk v bloku a počet bloků v PLD prvku je dán konkrétním typem. Toto řešení umožňuje zapojit PLD prvek do různých logik současně.



**I/O buňka s  
paměťovým  
elementem**

Obr. 7 - 22 Základní zapojení IO buňky s paměťovými elementy

IO buňka může být realizována s paměťovým elementem, většinou se jedná o D klopný obvod, obr. 7 - 22. Na každý vodič IO buňky je zařazen D klopný obvod. Zapojení multiplexeru dovoluje konfigurovat daný vodič jako kombinační nebo s paměťovým elementem. Paměťové elementy mají i povolovací vstup pro hodinový vstup a také mohou být doplněny o vstupy pro nulování nebo nastavení. Samozřejmě, že po náběhu napětí se očekává nastavení L úrovně na výstupu.



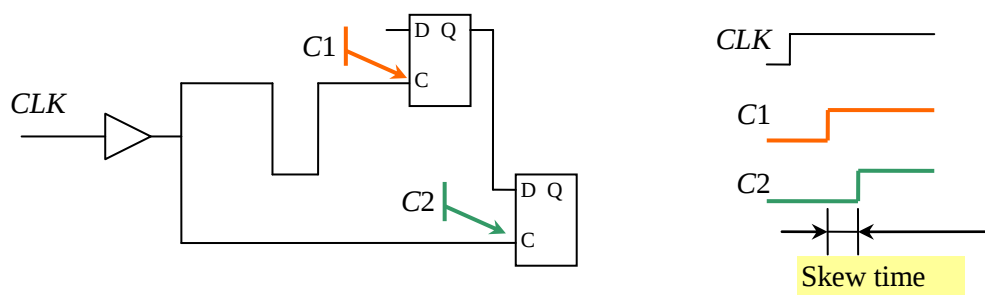
Obr. 7 - 23 Zapojení IO buňky s dvěma D klopnými obvody na vodič

Zapojení IO buňky může být modifikováno aplikováním dvou paměťových elementů, obr. 7 - 23. Vstupy a výstupy paměťových elementů jsou zapojeny na propojovací síť, a tím je umožněn výběr vstupních signálů pro paměťové elementy. Samozřejmě, že IO buňka obsahuje multiplexer pro možnost kombinačního výstupu nebo výstupu s paměťovým elementem. Dva paměťové elementy se dají použít pro zapojení principu DDR nebo pro eliminaci metastabilního stavu na vstupu PLD prvku. Princip DDR souvisí s přenosem informace, kdy v každé půl periodě hodin se přenáší jeden bit. Toto zapojení se uplatňuje v komunikaci s pamětmi. Pro eliminaci metastabilního stavu klopných obvodů se používá sériové řazení paměťových elementů.

Skutečné zapojení IO buňky může být ještě doplněno o Boundary Scan Register, který slouží k testování PLD prvku a desky plošného spoje. V IO buňce se aplikují tři buňky Boundary Scan Registru, pro vstupní signál, pro výstupní signál a pro signál řízení třetího stavu. Každá buňka může být realizována jedním nebo dvěma D klopnými obvody a zapojení buňky registru je dáno definicí Boundary Scan. Více v samostatné kapitole.

## 7.6 Rozvody hodinových signálů

Rozvod hodinových signálů je velice složitá problematika, kterou v rámci PLD prvku zvládá jeho výrobce. Jedná se o splnění podmínky, že vzorkovací hrana musí být u všech klopných obvodů ve stejném čase, ideální případ. V praxi se jedná, aby rozdílový čas – skew time byl minimální, obr. 7 - 24.



Obr. 7 - 24 Problém rozvodu hodinového signálu

Nutno si uvědomit, že spoj od budiče po hodinový vstup má určité zpoždění. Pokud bude toto zpoždění ve větvích různé, dojde k časovému posuvu na hodinových vstupech klopného obvodu, a tím k vzniku časového rozdílu – skew time. Při velké hodnotě rozdílového času může být funkce registru nebo sekvenčního obvodu ohrožena. Typický příklad je sériový posuvný registr, kdy může dojít k ztrátě informace.

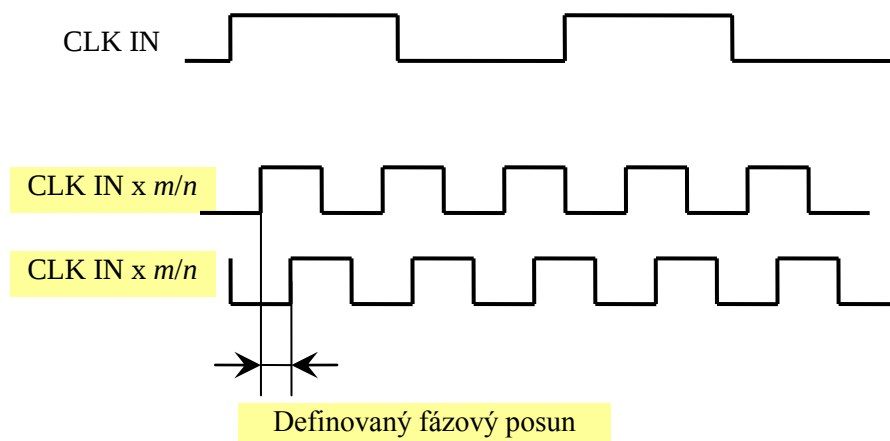
**Důvod**

Uživatel nemusí být s touto problematikou detailně seznámen, protože výrobce PLD prvku věnuje této problematice velkou pozornost a problém se snaží eliminovat. Skutečné rozvody hodinových signálů v PLD prvcích jsou realizovány samostatnou a složitou sítí.

**Realita rozvodu hodinového signálu**

## 7.7 Správce hodinového signálu

Dnešní PLD prvky používají hodinový signál až stovky MHz. Generovat takové hodiny mimo PLD prvek a navrhnout jejich rozvod do více prvků je velice náročné. Za další, v zapojení může být použito více hodinových signálů s definicí vzájemného časového posuvu. Proto výrobci přímo v PLD prvku vytvořili speciální obvody pro generování hodinových signálů pod různými názvy, ale ve všeobecnosti se jedná o správu hodinového signálu. Jeho základní vlastnosti je násobení vstupního hodinového signálu definovanými koeficienty pomocí fázového závěsu - PLL obvod. Výstupem může být několik hodinových signálů různé frekvence s možností definovat fázové nebo časové posuvy mezi nimi, obr. 7 - 25. Podrobněji o dané problematice pojednává například literatura [Altera 2003], [Actel 2004], [Latticesemi 2003] a [Xilinx 2003b].

**Správce hodinového signálu**

Obr. 7 - 25 Možná činnost správce hodinového signálu

## 7.8 Paměťové bloky

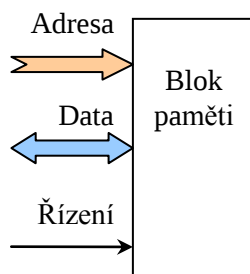
PLD prvky dnes obsahují paměti, které mohou sloužit k uchovávání informací. Tyto paměti jsou v PLD prvku vytvořeny jednak jako samostatné zapojení nebo existuje varianta, kdy část PLD prvku sloužící k vytváření logické sítě může být konfigurována jako paměť. Tyto paměti mají

**Typy paměťových bloků a přístupy k datům**

různé provedení a možné zapojení. U každého prvku je definován způsob napojení na propojovací síť a buňky vytvářející logickou síť. Jedná se o paralelní napojení nebo sériové napojení. Obě verze ještě v synchronním nebo asynchronním provedení. Paralelní napojení značí, že adresa a data jsou k paměti přivedena samostatně, paralelně. Synchronní napojení značí, že se komunikace probíhá synchronně podle hodinového signálu. V některých PLD prvcích se můžeme také setkat s blokem EPROM paměti, která slouží k uchování informací získaných v průběhu zpracování nebo k uchování konfigurace pro zpracování i v případě, že prvek bude bez napájení.

Zajímavější je aplikace SRAM paměti - Static RAM, kdy se můžeme setkat s různými řešeními a použitím. Organizace paměti je konfigurovatelná v daných mezích, například SRAM paměť o kapacitě 1 024 bitů může být organizována jako 1 024 x 1 bit, 512 x 2 bity až 32 x 32 bitů. Skutečné kapacity závisí na konkrétním PLD prvku. Z pohledu přístupu k datům se může jednat o asynchronní přístup nebo synchronní. Asynchronní přístup je dán signály povolení a zápis, enable and write nezávislé na hodinovém signálu. U synchronního přístupu jsou všechny akce v paměti jako čtení a zápis prováděny podle hodinového signálu. Další možné konfigurace SRAM paměti jsou uvedeny dále.

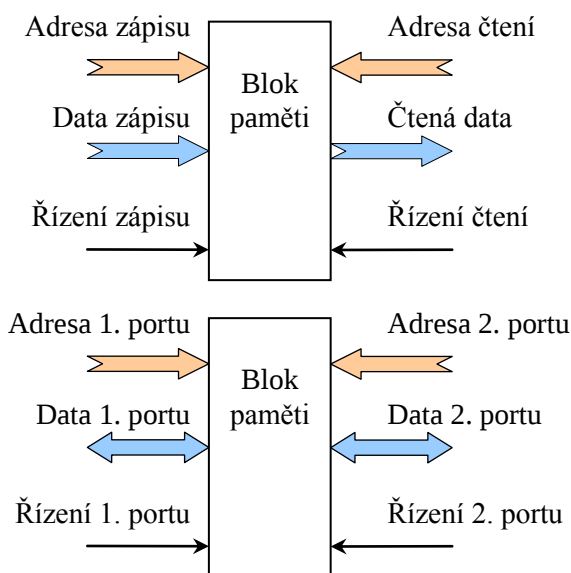
**Statická RAM  
paměť**



**Jednoportová  
paměť**

Obr. 7 – 26 Jednoportová paměť

**Jednoportová paměť**, jedná se o klasickou SRAM paměť. K paměťové informaci lze přistupovat pomocí jedné sady - adresy, dat a řídicích signálů, obr 7 - 26.



**Dvouportová  
paměť**

Obr. 7 - 27 Principy dvouportové paměti

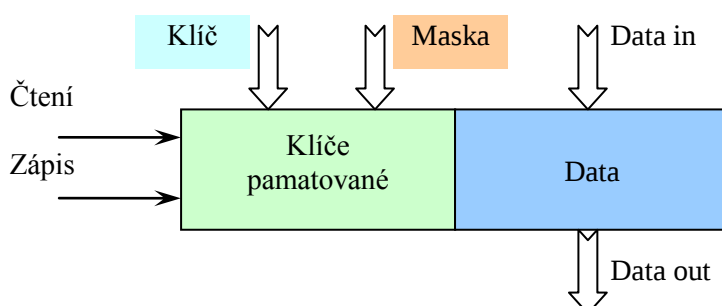
**Dvouportová paměť**, jedná se o přístup k paměťové informaci ze dvou různých nezávislých zdrojů, obr. 7 - 27. Paměť má dva porty a podle jejich použití se dvouportová paměť dělí na dvě skupiny. První skupina jsou paměti, kdy jeden port slouží k zápisu informace na danou adresu a druhý port k čtení informace z dané adresy. Paměť má samostatný vstup pro data a adresu zápisu a další vstup pro adresu čtení a výstup pro čtené data. Druhá skupina je paměť, kdy každý port může informaci do paměti zapisovat a číst nezávisle na druhém portu.

**Sériový posuvný registr**, paměť může být konfigurována do funkce sériového posuvného registru.

Sériový  
posuvný  
registr

**FIFO, LIFO – fronta, zásobník**, jedná se o zapojení paměti jako fronta, FIFO – First In First Out nebo zásobník, LIFO – Last In First Out. Potřebné obvody pro jejich činnost jsou již vytvořeny a uživatel nepracuje s adresou buňky ale pouze s principem činnosti. Má k dispozici pouze signály čtení a zápis, popřípadě signalizační signály o překročení kapacity.

FIFO a LIFO  
paměť



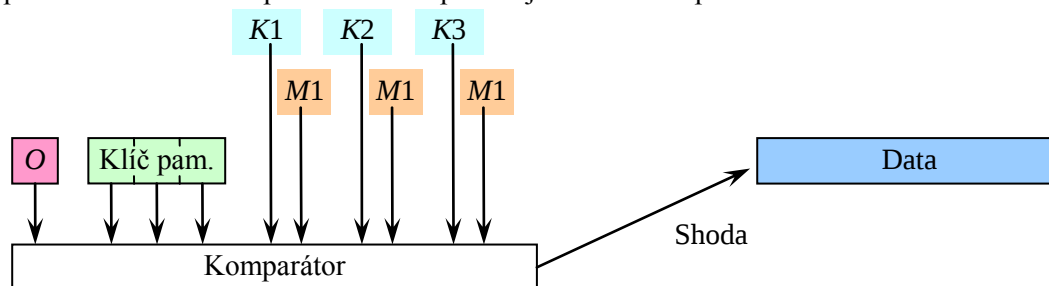
Obr. 7 - 28 Blokové zapojení CAM paměti

**CAM paměť, Contents Access Memory**, je dalším významným typem paměti, viz obr. 7 - 28. Jedná se o paměť, kde se k pamatované položce přistupuje podle klíče. S každou paměťovou buňkou dat je pevně spojen klíč. Klíč je kombinace bitů, která charakterizuje data podle představ uživatele. Právě tento klíč se používá k čtení a zápisu CAM paměti. Asociativní paměť se používá k vyhledávání dat podle klíče, třídění dat nebo jako cache paměť. Při čtení CAM paměti se na vstup přivede klíč, který se porovná s pamatovanými klíči. Řádek paměti, kde nastane shoda se vstupním klíčem, se aktivuje a příslušná data se objeví na výstupu paměti. Maska je modifikací CAM paměti a určuje, které bity klíče mají být porovnávány. Zápis do CAM paměti je komplikovanější a má dvě základní varianty. První je zápis nového klíče a dat do volné buňky CAM paměti. Druhá varianta je pouze změna dat pro daný klíč.

CAM paměť  
-  
paměť  
adresovaná  
obsahem

Jedna buňka CAM paměti je na obr. 7 – 29. Každá buňka se skládá ze tří polí. Každá buňka obsahuje příznak *O* délky jeden bit, který určuje, zda buňka obsahuje data nebo je volná. Příznak je skrytý bit a je určen pouze pro interní potřebu CAM paměti. Pole pamatovaný klíč a data slouží k uchování informace. Každá buňka má komparátor, který porovná pamatovaný klíč se vstupním klíčem *K1* až *K3*. Výstupní signál komparátoru aktivuje pouze jedno pole data, které je pevně spojeno s polem pamatovaného klíče. Signály masky *M1* až *M3* mohou být dalším vstupním signálem CAM paměti a určují, které bity klíče *K1* až *K3* se budou používat pro porovnání. Základní operace s CAM paměti jsou čtení a zápis.

Princip  
přístupu  
pomocí klíče



Obr. 7 - 29 Možné zapojení jedné buňky CAM paměti

Zápis se provádí do volné buňky a zapíše se hodnota klíče a příslušná data. Bit obsazení buňky se nastaví na hodnotu obsazeno. Jiný zápis může přepsat pouze pole data novým záznamem pouze u buněk, kde došlo ke shodě klíčů.

**Princip  
přístupu  
pomocí klíče  
Čtení z CAM  
paměti**

Při čtení se na vstup CAM paměti přivede klíč K1 až K3, který je přiveden do každé buňky, obr. 7 – 29. V každé obsazené buňce se provede porovnání na shodu. V případě shody se na výstupu datového pole objeví datová informace spojená s klíčem. Vstupní klíč může být maskován, maska určuje, zda příslušný bit klíče bude použit pro výběr, zda bit klíče bude zahrnut do vyhodnocení shody nebo ne.

Toto je pouze základní princip, který neřeší krajní situace. Tyto krajní situace nutno potom signalizovat dalšími signály. Jedná se o situace:

**Možné krajní  
situace při  
činnosti CAM  
paměti**

- Jak se bude chovat CAM paměť, když všechny buňky budou obsazeny a je požadavek na další zápis. Zapsat a kam nebo nezapsat.
- Jakým způsobem nulovat buňku CAM paměti.
- V případě čtení může nastat situace, že dvě a více buněk vyhodnotí shodu. Potom je nutno definovat, zda výstup bude pouze jeden, první buňka nebo více, sekvence dat ze všech buněk.
- Zápis dat do volné buňky nebo podle klíče.

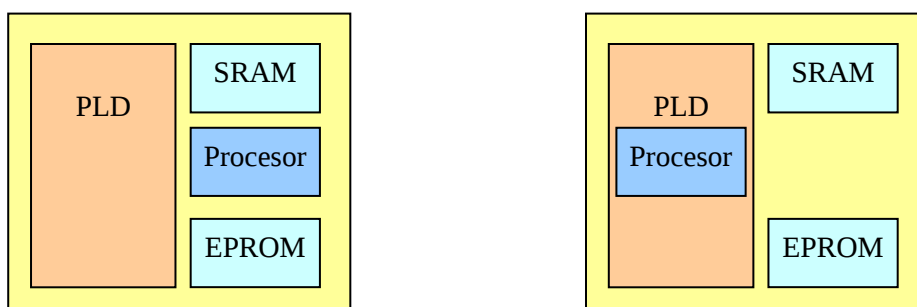
## 7.9 Procesor

Dnes PLD prvky mohou obsahovat i procesor, který lze použít k návrhu číslicového systému. Některé části systému lze řešit softwarově a jiné hardwarově. Zde se setkáváme s dvojím provedením, soft a hard procesor. Hard procesor je hardwarový procesor, který je jako blok umístěný na křemíku a definovaným způsobem napojen na obvody PLD prvku. Může se jednat o mikroprocesor 8051 pro jednoduché aplikace až po Power PC procesor, pro složitější aplikace. V případě výkonných procesorů pro složité aplikace obsahuje PLD prvek ještě další speciální bloky.

**Hard procesor**

Soft procesor je vlastně programový zápis, který se dodává jako IP. Soft procesor se vytváří z části PLD prvku naprogramováním a zbytek PLD prvku se použije pro podpůrné obvody procesoru, speciální zapojení podporující procesor.

**Soft procesor**



Hard procesor

Soft procesor

Obr. 7- 30 Hard a soft procesor v PLD prvku

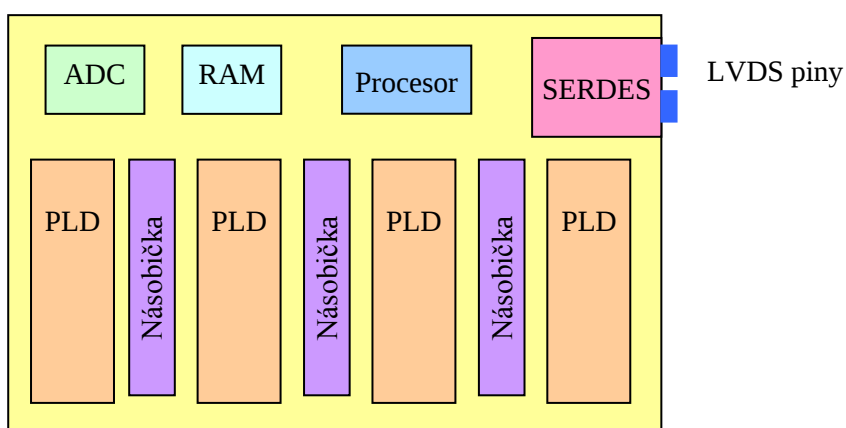
Návrhář rozhoduje o použití procesoru v návrhu číslicového systému, a tím i o řešení některých situací programově. S tím souvisí i speciální nadstavba vývojového systému, která propojuje programový zápis s hardwarovým řešením, hlavně v oblasti simulace. Vývoj procesorových aplikací se zásadně předpokládá na bázi ANSI C jazyku. Zvládnutí assembleru pro několik typů procesorů a optimálního zápisu je náročné a proto je vhodné tyto činnosti přenechat překladačům.

## 7.10 Doplnující bloky PLD prvku

PLD prvky, hlavně nejvyšší integrace obsahují přímo na křemíku specializované obvody pro různé aplikace. Běžně to jsou paměťové bloky a procesory, které jsou uvedené výše. Výčet všech bloků není možný, protože se stále doplňuje o nové bloky. Proto si zde uvedeme jen některé a jejich možné použití.

- Hardwarové násobičky. Hlavní použití je pro zpracování číslicových signálů, typicky číslicové filtry.
- A/D převodníky. Pro číslicové zpracování signálu, typicky číslicové filtry, řízení.
- Obvody pro vysílání a příjem sériové informace SERDES – Serializer - Deserializer. Zde se jedná o sériové toky řádově Gbit za sekundu. Aplikace SDH, 10Gbps Ethernet a podobně.

**Další  
doplnující  
bloky PLD  
prvku**



Obr. 7- 31 Další možné doplňující bloky PLD prvku

## 7.11 Časový model

Pro každý PLD prvek je zpracován časový model, který popisuje jednotlivé elementy, spoje v prvku hodnotou zpoždění signálů. Nutno si uvědomit, že programovatelné propojky jsou aktivní prvky, které mají zpoždění. Tím i propojovací síť má zpoždění. Zpoždění jednotlivých cest, hradel, paměťových elementů pak udává celkové časové vlastnosti návrhu v PLD prvku.

Vývojové systémy obsahují časové modely konkrétních PLD prvků a provádějí časovou analýzu a výpočet zpoždění signálů od vstupu na výstup. Časová analýza se týká maximálně použité frekvence pro synchronizaci a zajištění potřebných časů předstihu a přesahu dat na paměťových elementech. Pro návrháře není důležitá znalost časového modelu, jenom musí využívat její výsledky.

### Otázky kap. 7



- 7.1 Popište princip, význam alokace a sdílení součinných termů v AND OR architektuře PLD prvku.
- 7.2 Popište možné vlastnosti paměťového elementu v PLD prvku.
- 7.3 Popište princip LUT tabulky.

- 7.4 Použití přenosové logiky v PLD obvodech.
- 7.5 Popište princip kaskádní logiky.
- 7.6 Popište vstupně-výstupní buňku PLD prvku.
- 7.7 Popište význam rozvodu hodin a správce hodinového signálu v PLD prvcích.
- 7.8 Popište možné typy paměťových bloků v PLD prvcích.
- 7.9 Popište princip CAM paměti.
- 7.10 Popište další možné bloky, se kterými se lze setkat v PLD obvodech.

### **Odměna a odpočinek**



Dalším trendem konce 80. let je snaha zvýšit integraci PLD obvodů umístěním více PAL nebo GAL obvodů do jednoho integrovaného obvodu. Při pohledu na křemík jsou na kraji vstupně-výstupní buňky a uprostřed několik GAL obvodů, které jsou propojeny další programovatelnou propojovací sítí. Jinak, obvody mají dvě propojovací sítě, první v samotném GAL obvodu jako programovatelnou AND matici a druhou, nadřazenou propojovací síť, která propojuje jednotlivé GAL obvody mezi sebou a zároveň se vstupně-výstupními buňkami. Tyto obvody dostaly označení CPLD – Complex Programmable Logic Device. Dnes přesné vymezení CPLD obvodů je pro mne neznámé, protože se lze setkat s CPLD obvody i s LUT bloky – Look Up Tables, které byly dominantní pro FPGA.

Na přelomu 80. a 90. let společnost Texas Instruments uvádí na trh FPGA obvody založené na dvou typech buněk, kde každá má specifické zapojení. Jedna buňka je určena pro realizaci kombinační sítě a druhá pro paměťový element. Dnes se s tímto principem setkáváme u společnosti Actel.

## 8. OBVODY CPLD

**Čas ke studiu:** 1 hodina



**Cíl** Po prostudování této kapitoly budete umět



- Základní princip CPLD obvodu

**Výklad**



Vymezení CPLD - Complex PLD obvodů je dnes pro mne osobně problematické. Jednak lze CPLD obvody chápat jako další vývojový stupeň obvodů PAL a GAL, zvyšování integrace, protože nejdříve začínaly na principu AND a OR matice. Dnes, přebírají některé prvky dříve typické pro FPGA, hlavně uplatněním buňky s LUT tabulkou. Rozdíly mezi CPLD a FPGA, které vyplývají s technologií výroby, jsou důležité pro výrobce, a proto CPLD prvky já osobně charakterizuji skutečnostmi, vlastnostmi, které jsou důležité pro jejich použití.

**Vymezení  
CPLD prvku**

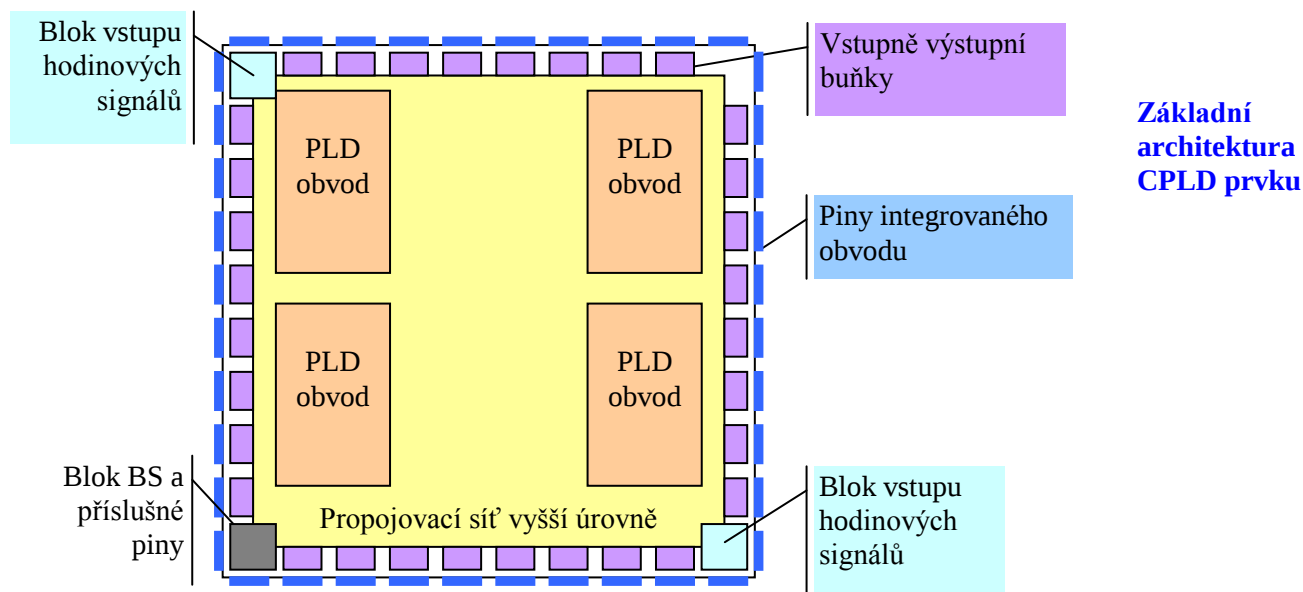
- Aplikace dvojúrovňové propojovací sítě.
- Aplikace vstupně-výstupních buněk, hlavně bez paměťového elementu.
- Reprogramovatelné prvky, s energeticky nezávislými propojkami.
- Počet programovatelných propojek jde až do jednotek miliónů.
- Stupeň integrace. Dnes CPLD prvky mají počet ekvivalentních hradel od několika tisíců až po stovky tisíc ekvivalentních hradel.
- CPLD prvky jsou střední třídou mezi standardními prvky a FPGA prvky.

V budoucnosti se dá očekávat, že dnes typické zapojení, vlastnosti pro FPGA obvody se začnou objevovat v CPLD obvodech.

### 8.1 Základ CPLD obvodu

Nezákladnější popis CPLD obvodu je nejlépe při pohledu na křemík, jedná se o řez integrovaným obvodem. Základ CPLD prvku tvoří PLD obvody, IO buňky a propojovací síť vyšší úrovně, obr. 8 - 1. PLD obvod je v převážně typu GAL s OLMC buňkou. Dnes se na tomto místě objevují i zapojení pomocí buněk s LUT, které jsou známe s FPGA obvodů. PLD obvod má vlastní programovatelnou propojovací síť a je možné v něm vytvářet zapojení logické sítě. Propojovací síť PLD obvodu nazývám propojovací sítí nižší úrovně. Druhá také programovatelná propojovací síť je propojovací síť vyšší úrovně, která propojuje jednotlivé PLD obvody mezi sebou, a tím umožňuje vytvářet logické sítě větší složitosti. PLD obvod je propojen s propojovací sítí vyšší úrovně desítkami signálů, řekněme 32 nebo 64 vstupních signálů a 16 nebo 32 výstupních signálů z PLD obvodu. Základní rozdíl mezi jednotlivými úrovněmi propojovacích sítí je ve zpoždění signálu. Zapojení, které využije pouze propojovací síť PLD obvodu bude mít menší zpoždění než zapojení, které využije i propojovací síť vyšší

úrovně. Na kraji křemíku jsou aplikované vstupně-výstupní buňky, které mohou být zapojeny přímo na PLD obvody nebo na propojovací síť vyšší úrovně.



Obr. 8 - 1 Základní pohled na CPLD prvek

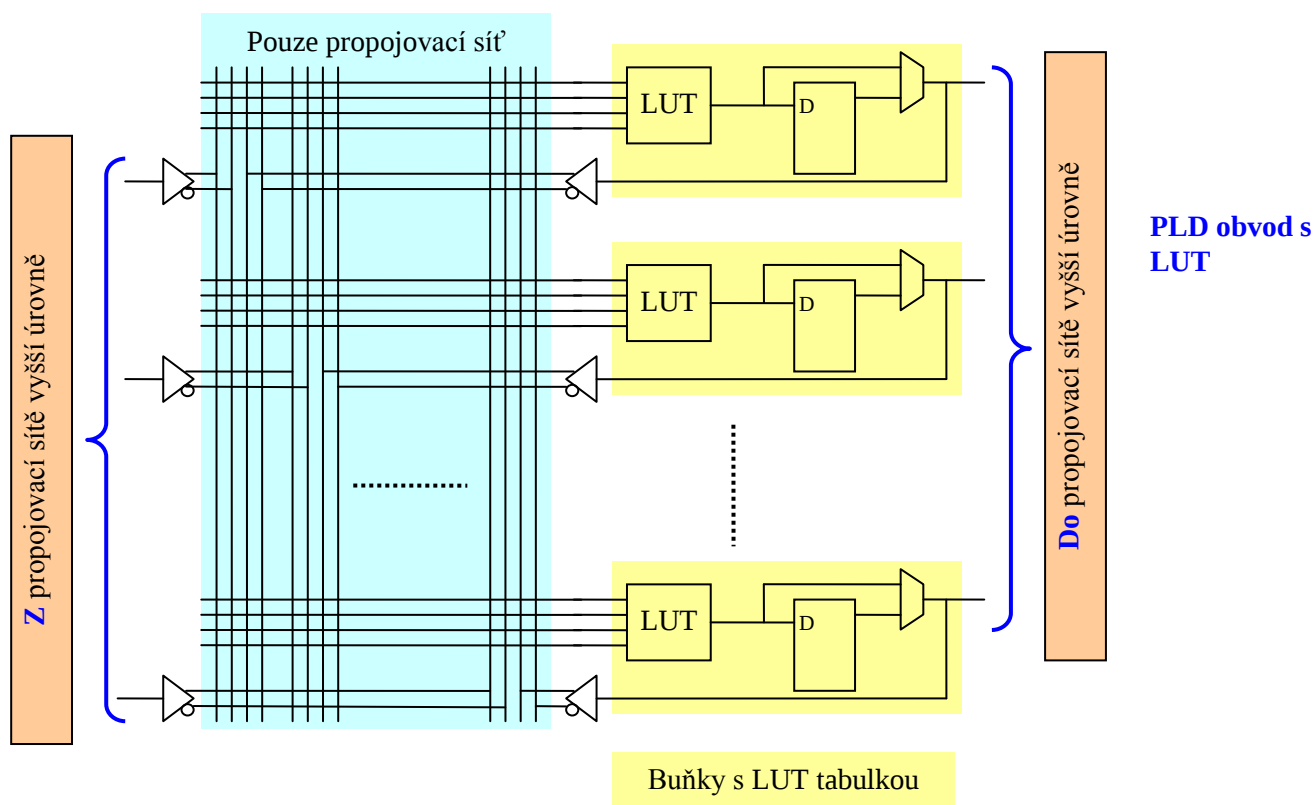
Zde je použit pojem PLD obvod, ve skutečnosti každý výrobce tento blok nazývá vlastním názvem. Počet PLD obvodů v CPLD prvcích není definován a závisí na konkrétním prvku a výrobci. Potom počet PLD obvodů v CPLD prvku je udáván jako základní charakteristika obvodu.

Počet paměťových elementů v CPLD prvku závisí na počtu PLD obvodů a bývá až stovky, tisíce paměťových elementů. Většinou se jedná o D klopný obvod.

Mnohdy PLD obvody mají i speciální možnosti propojovat jednotlivé buňky, mohou obsahovat zapojení pro přenosovou logiku, propojení paměťových elementů pro sériový registr a podobně.

## 8.2 PLD obvod

Skutečné zapojení PLD obvodu a jeho název závisí na konkrétním typu a výrobci. Převážně se zde uplatňuje zapojení s programovatelnou AND a pevnou OR maticí s následnou buňkou s paměťovým elementem. Jedná se o základní GAL princip. Lze se setkat i s aplikací PLA struktury, programovatelná AND i OR matice. Počet buněk na jeden PLD obvod bývá typicky 16. Propojovací síť první úrovně je v tomto případě tvořena programovatelnou maticí AND.



Obr. 8 – 2 Architektura PLD obvodu s LUT tabulkou

Někteří výrobci použijí PLD obvod založený na buňce s LUT tabulkou. V tomto případě je před buňkami propojovací síť bez logické funkce, která umožňuje propojit vstupy LUT tabulky na vstupní signály. Několik LUT buněk s propojovací sítí potom tvoří PLD obvod.

### 8.3 Propojovací síť a rozvody hodinového signálu

CPLD prvek má dvě propojovací sítě s různými vlastnostmi, hlavně co se týče zpoždění signálů. Propojovací síť nižší úrovně má menší zpoždění než propojovací síť vyšší úrovně. Propojovací síť nižší úrovně, která je součástí PLD prvku, umožňuje i speciální propojení pro přenosovou logiku, sériové propojení paměťových elementů apod. Detailní znalost všech možností propojení je důležitá pouze pro návrhový systém, a ne pro uživatele.

**Propojovací síť  
vyšší a nižší  
úrovně**

V CPLD prvcích jsou specializované sítě pro rozvod hodinového signálu, aby se zajistil minimální rozdílový čas – skew time na hodinových vstupech paměťových elementů. CPLD prvek většinou obsahuje několik hodinových signálů. Frekvence hodinového signálu bývá až do 500 MHz. Dříve se pouze používaly pro vstup hodinového signálu pouze vyhrazené piny, ale dnes se můžeme setkat i s aplikací správce hodinových signálů. Správce hodinových signálů generuje interní hodinové signály na základě externího hodinového signálu. Interní hodinové signály jsou násobkem externího signálu a mohou mít různé časové posuvy. Výhodou tohoto řešení je, že frekvence externího hodinového signálu, který je přiveden na pin CPLD prvku, může být mnohonásobně menší než frekvence interních hodinových signálů. Koeficient násobení a velikost časových posuvů interních hodinových signálů jsou programovatelné návrhářem zapojení. Nižší externí frekvence hodinového signálu je vhodnější pro rozvod na desce plošného spoje.

**Hodinové  
rozvody**

## 8.4 Vstupně-výstupní buňky

CPLD prvky většinou používají vstupně-výstupní buňku bez paměťového elementu, výjimečně s paměťovým elementem. Vstupní, výstupní signály a signály pro povolení třetího stavu jsou zapojeny buď přímo do PLD obvodu nebo na propojovací síť vyšší úrovně.

**Vstupně –  
výstupní buňky**

Vlastnosti samotného pinu, budiče a vstupního hradla jsou dány konkrétním typem a výrobcem PLD prvku. Jedná se o vlastnosti jako jsou přízpůsobovací odpory, logika, ošetřující diody a další.

Dnes do IO buňky proniká Boundary Scan architektura. Jedná se o zapracování Boundary Scan registru do vstupně-výstupní buňky. Boundary Scan architektura, která slouží k testování číslicového systému.

## 8.5 Pouzdra

CPLD obvody jsou v různých pouzdrech pro různá prostředí. Počet pinů se pohybuje od 40 až po stovky pinů. Ne všechny piny jsou určeny pro vstup a výstup navrhovaného systému. Mezi důležité piny patří napájení obvodu, které je řešeno pomocí několika pinů. Vícenásobné napájení eliminuje možnost vzniku falešných impulsů v důsledku posuvu zemnicího potenciálu a strmosti - slew rate signálů. Další část pinů je věnována hodinovým vstupům a také Boundary Scan interface (IEEE 1149.1).

## 8.6 Doplnující bloky

Dnes se u CPLD obvodů můžeme setkat s aplikací statické RAM paměti a u některých typů s soft procesorem. Ostatní specializované bloky se zatím neobjevují.

### Otázky kap. 8



- 8.1 Popište typické znaky CPLD obvodů.
- 8.2 Popište základní princip organizace CPLD obvodu a převážný princip PLD prvku.

### Odměna a odpočinek



Dalším významným mezníkem bylo zavedení technologie ISP – In System Programming. Jedná se o možnost programování obsahu PLD obvodu přímo na zapojené desce plošného spoje. Mezi průkopníky této technologie byla společnost Lattice Semiconductors, která vyvinula svůj vlastní systém a představila jej v roce 1992. Výrobci PLD obvodů se však sjednotili na technologii ISP na základě JTAG – Boundary Scan. Technologie ISP využívá interface Boundary Scan a byla standardizována jako IEEE 1532 v roce 2002 pod názvem In System Configuration.

Koncem 90. let minulého století výrobci PLD obvodů do svých obvodů, hlavně FPGA obvodů, začínají doplňovat specifické zapojení jako jsou obvody přijímače a vysílače pro sériový přenos informace o stovkách megabitů až po jednotky gigabitů za sekundu. Jedná se o obvody pro Ethernet 1 Gbit/s a 10 Gbit/s, SDH a jiné aplikace. Další doplňována zapojení jsou paměti RAM, EPROM, hardwarové násobičky, hardwarové procesory a jiné.

Se samotnou historií PLD prvků souvisí i vývojové systémy, které přispěly svou mírou k rozšíření PLD prvků. Dnes se tyto systémy označují zkratkou EDA – Electronic Design

Automation nebo ECAD – Electronic Computer-Aided Design. Prvním průkopníkem byl již zmíněný Fuse assembler, který generoval zapojení propojek PAL obvodů z logických výrazů. Tento první program přispěl k rozšíření používání PAL obvodů tím, že automatizoval návrh.

## 9. OBVODY FPGA

**Čas ke studiu: 1 hodina**



**Cíl** Po prostudování této kapitoly budete umět



- Základní princip FPGA obvodu

**Výklad**



Obvody FPGA – Field Programmable Gate Array považují za nejvyšší stupeň PLD prvků. Jejich přesné vymezení je pro mne problematické, obdobně jako u CPLD obvodů. FPGA obvody vycházejí z hradlových polí, nepoužívají princip AND OR matice. Typickými znaky FPGA obvodů v minulosti bylo použití LUT buňky, programovatelné propojovací sítě v jedné úrovni a propojky na bázi statické RAM paměti. Z pohledu uživatele charakterizují FPGA obvody následně:

**Vymezení  
FPGA**

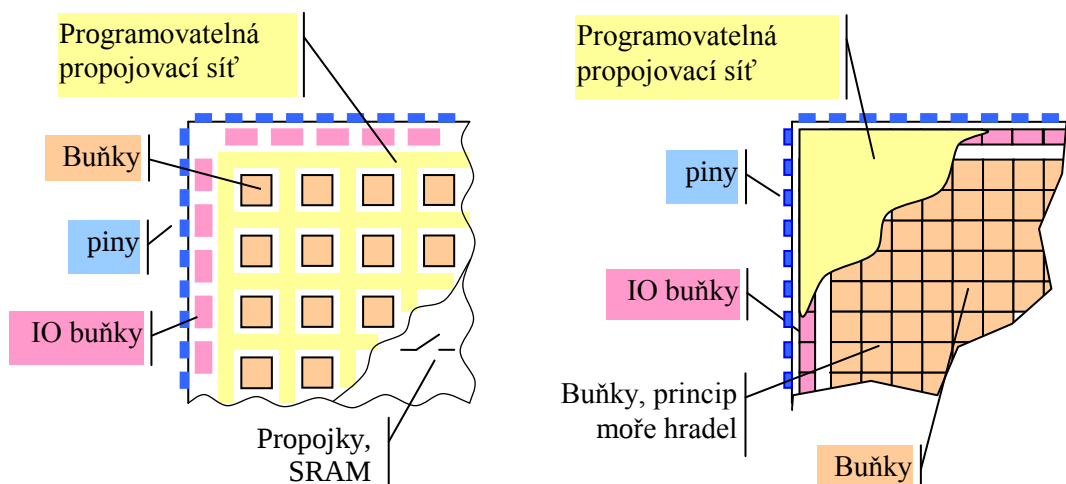
- Aplikace propojovací sítě v jedné úrovni, která propojuje buňky vytvářející logickou síť a vstupně-výstupní buňky.
- Aplikace vstupně-výstupních buněk s paměťovým elementem, výjimečně bez.
- Reprogramovatelné prvky, aplikace energeticky závislých propojek na principu statické RAM paměti, u některých typů také energetických nezávislých propojek.
- Počet programovatelných propojek je jednotky až desítky miliónů propojek.
- Stupeň integrace. Dnes FPGA prvky mají počet ekvivalentních hradel od několika desetitisíců až po milióny ekvivalentních hradel.
- FPGA prvky jsou nejvyšší třídou PLD prvků.

### 9.1 Základy FPGA obvodu

Základní popis je vhodné začít pohledem na křemík FPGA prvku, obr. 9 - 1. Základem je buňka, která dokáže vytvářet určitou logickou funkci s nebo bez paměťového elementu. FPGA prvek pak obsahuje několik desítek tisíc až milióny takových buněk. Pro zajištění vstupu a výstupu se používají IO buňky jako u CPLD obvodů.

Původně se jednalo o umístění buněk, propojovací sítě a IO buněk v jedné vrstvě. Statická RAM paměť pro řízení propojek byla umístěná pod touto vrstvou. Dnes se spíše upřednostňuje uspořádání, kdy nejvyšší vrstva je propojovací síť s programovatelnými propojkami, pod ní je vrstva buněk a IO buněk. Uspořádáním buněk těsně vedle sebe umožnilo podstatně zvýšit hustotu buněk na křemíku, protože vrstva již neobsahuje propojovací síť. Tato síť je umístěna nad buňkami. Pro uspořádání buněk těsně vedle sebe se používá pojem moře hradel – sea of gates.

**Základní  
architektura  
FPGA prvku**



Obr. 9 - 1 Základní pohled na křemík pro FPGA obvody

Buňka je převážně realizována pomocí LUT tabulky s paměťovým elementem. Některé FPGA obvody mají v jedné buňce i dvě LUT tabulky s dvěma paměťovými elementy. Lze se setkat i s aplikací jiných speciálních zapojení, které tvoří buňku, viz kapitola 7.

Počet buněk FPGA obvodů na bázi LUT tabulky se dnes pohybuje od desítek tisíc až po statisíce. V případě jednodušších buněk může FPGA obvod obsahovat až milióny buněk. Obdobný je i počet klopných obvodů.

## 9.2 Propojovací síť a rozvody hodinového signálu

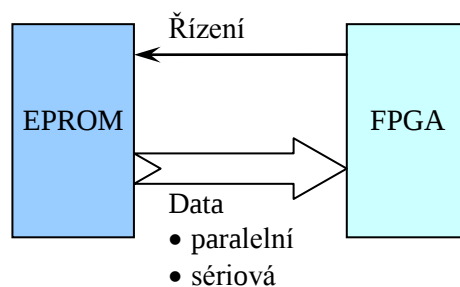
Propojovací síť je v jedné úrovni. Obsahuje spoje různých vlastností, které umožňují propojit jednotlivé buňky optimálním způsobem a hlavně s co nejmenším zpožděním. Proto se zde můžeme setkat se speciálními spoji propojující sousední buňky, propojující každou druhou, čtvrtou buňku, s třístavovými spoji přes celý FPGA obvod, spoji určenými pro napojení vstupně-výstupních buněk apod. Detailní znalost propojovací sítě mají vývojové systémy, pro uživatele je to jenom informativní záležitost.

FPGA obvody obsahují samostatnou programovatelnou síť pro rozvod hodinového signálu k jednotlivým klopným obvodům, aby byl zajištěn minimální rozdílový čas – skew time na hodinových vstupech klopných obvodů. Jako zdroje hodinového signálu se dnes zásadně používají správci hodin.

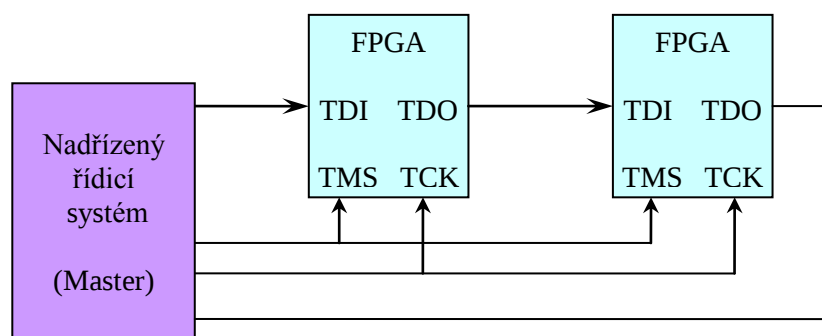
## 9.3 Programovatelné propojky

Pro FPGA obvody bylo význačné použití programovatelných propojek, jejichž stav je uchováván ve statické RAM paměti. Tím se FPGA obvody stávají energeticky závislé a po každém zapnutí napájení je nutno do nich nahrát obsah, který definuje propojení. Proto se FPGA obvody často kombinují s EPROM pamětí, která uchovává obsah FPGA obvodů. Většinou se jedná o speciální EPROM paměti, které jsou pro tento účel přímo určené. EPROM obvody a FPGA obvod se propojují definovaným způsobem podle katalogu. Samozřejmě, že paměť EPROM může být nahrazena jiným systémem pro uchování dat, např. počítačem. **Propojky pomocí SRAM paměti**

Základní způsob nahrávání dat je sériový nebo paralelní způsob, obr. 9 - 2.



Obr. 9 – 2 Nahrávání obsahu FPGA obvodu paralelním způsobem



Obr. 9 – 3 Aplikace BS interface pro nahrávání obsahu FPGA obvodu

V případě sériového způsobu se dost často používá Boundary Scan interface (IEEE 1149.1), obr. 9 - 3. Samozřejmě, že existují i FPGA obvody s energeticky nezávislými propojkami, ale je jich méně.

## 9.4 Vstupně-výstupní buňky

**Základní  
architektura  
FPGA prvku**

Vstupně-výstupní buňky FPGA obvodů jsou většinou s paměťovým elementem, ale i s dvěma elementy. Dva paměťové elementy se dají použít pro vytvoření DDR interface na paměť nebo pro eliminaci metastabilního stavu. Vlastnosti samotného pinu, budiče a vstupního hradla jsou dané konkrétním typem a výrobcem PLD prvku. Jedná se o vlastnosti jako přizpůsobovací odpory, logika, ošetřující diody a další.

Vstupně-výstupní buňka FPGA obvodů dále obsahuje buňky Boundary Scan registru, které jsou určené pro testování. Jedná se o prostředek umožňující snadné testování DFT – Design For Test. Mnohdy může být základem vestavěného testování BIST - Built In Self Test.

## 9.5 Pouzdra

FPGA obvody se dodávají v různých pouzdrech pro různá prostředí. Počet pinů je dán konkrétním typem a nejvyšší prvky mají až 1 500 pinů při rozměrech pouzdra 4 x 4 cm (pouzdro BGA 1508). Napájení obvodu je přes několik pinů, hlavně zemnicí piny, aby se vyloučila možnost generovat falešné impulsy v důsledku posuvu potenciálu země. Jedná se o důsledky slew rate signálů. Další část pinů je věnována hodinovým vstupům, signálům určených pro nahrávání obsahu a také Boundary Scan interface.

## 9.6 Bloky

FPGA obvody vždy byly špičkou PLD prvků, a proto jsou do nich zapracovány další podporné specializované bloky jako jsou:

- Statické RAM paměti. Jedná se o několik bloků paměti, kdy celková kapacita může dosáhnout až jednotek Mbitů.
- Hardwarové násobičky, A/D a D/A převodníky pro číslicové zpracování signálů.
- Obvody SERDES pro zpracování sériového toku až do jednotek Gbitů za sekundu.
- Hardwarové procesory.

## Otázky kap. 9



- 9.1 Popište typické znaky FPGA obvodů.
- 9.2 Popište způsob programování FPGA obvodů.

## Odměna a odpočinek



V roce 1983 se objevuje PLD kompilátor pod názvem CUPL – Universal Compiler for Programmable Logic od společnosti ASSISTED Technology. Program CUPL podporoval implementaci více obvodů od různých firem. Dnes se s označením CUPL můžeme setkat např. v systému ORCAD.

Dalším mezníkem v oblasti návrhových systémů bylo uvedení systému ABEL – Advance Boolean Expression Language společnosti DATA I/O v roce 1984. Dnes se s pojmem ABEL setkáváme jako s definicí jazyka pro návrh číslicových systémů, i když je dnes mnohdy nahrazován pojmem HDL. Tento jazyk umožnil popisovat chování číslicového systému nejen pomocí Boolean výrazů (booleovských výrazů) ale i nových zápisů. U kombinačních obvodů je to zápis pomocí pravdivostní tabulky a u sekvenčních obvodů popis chování pomocí programových zápisů.

Rozšiřování používání PLD obvodů a používání návrhových systémů vedlo výrobce PLD obvodů k definici formátu souboru určujícího obsah PLD. Tento formát definoval výbor Joint Electron Device Engineering Council pod označením JEDEC Standard No 3, který definuje formát souboru obsahu PLD prvku, který dokáže programátory PLD prvků načíst a podle něj naprogramovat PLD prvek. Tento předpis prošel revizemi a používá se dodnes.

V roce 1987 je definován nový jazyk pro popis číslicových systémů pod označením VHDL jako standard IEEE 1076-1987, který je po dalších modifikacích používán dodnes. U vzniku VHDL jazyka stála snaha DoD of USA – Department of Defense USA, ministerstvo obrany USA, které požadovalo sjednotit popis číslicových systémů, protože v té době existovalo několik vzájemně nekompatibilních jazyků. U vzniku VHDL jazyka stály společnosti IBM, Texas Instruments a Intermetrics. Zkratka jazyka VHDL je VHSIC Hardware Description Language, kde VHSIC je Very High Speed Integrated Circuits.

## 10. SYSTÉMOVÉ VLASTNOSTI PLD OBVODŮ

**Čas ke studiu: 2 hodiny**



**Cíl** Po prostudování této kapitoly budete umět



- Základní princip FPGA obvodu

**Výklad**



U PLD prvků se můžeme setkat s pojmy, které mohou ovlivňovat výběr pro návrh a použití PLD prvku. Dále uvedené systémové pojmy, vlastnosti, mnohá z nich jsou definována standardy.

### 10.1 Boundary Scan

BS - Boundary Scan je dán standardem IEEE 1149.1 s názvem „Test Access Port and Boundary Scan Architecture“. První standard byl schválen v roce 1990 a poslední verze je z roku 2001. Boundary Scan je prostředek pro snadné testování DFT – Design For Test. Boundary Scan je vhodným prostředkem i pro vestavěné testování číslicových systémů, BIST – Built In Self Test. BS architektura byla vyvíjena seskupením JTAG – Join Test Access Group. Bližší informace jsou v literatuře [Texas 1998], [Texas 1997c], [Bleeker, Eijnden, Jong 1993] a [Parker 2003].

**Digitální  
Boundary Scan**

Standard IEEE 1149.1 popisuje implementaci Boundary Scan v pro digitální piny, které jsou vstupem a výstupem číslicových systémů. Dalším vývojem vznikly nové modifikace standardů, které jsou zaměřené na testování jiných typů pinů. Jedná se o standard IEEE 1149.4, který popisuje implementaci Boundary Scan pro analogové piny – Analog Boundary Scan. Standard 1149.6 se zabývá testováním vylepšeným I/O pinů – Advanced I/O Testing. Jedná se o piny určené pro sériový přenos dat s vysokou kapacitou, jednotky Gbitů za sekundu. Poslední standardem je IEEE 1532, který implementuje programování programovatelných obvodů přímo na čipu. Název principu je ISC – In System Configuration. Analogový Boundary Scan není dále uveden, popis je v literatuře [Parker 2003].

**Varianty  
Boundary Scan**

Dále se zaměřím hlavně na implementaci Boundary Scan pro digitální piny. BS slouží k testování jádra integrovaného obvodu a okolí integrovaného obvodu bez žádného vlivu na navrhnoutou funkčnost obvodu. Význam BS pro digitální piny spočívá:

- V možnosti testovat propojení mezi jednotlivými integrovanými obvody.
- V možnosti testovat jádro integrovaného obvodu, hlavně pomocí spouštění vestavěné diagnostiky.
- V možnosti testovat části zapojení integrovaných obvodů, které nepodporují princip BS. Jedná se o test logického clusteru.
- V možnosti kontroly správného osazení desky plošného spoje integrovanými obvody včetně správné verze obsahu PLD prvku nebo firmware.

**Význam  
Boundary Scan**

- Jednotný způsob testování pro vývoj, výrobu a servis. Úspora je v ekonomických nákladech na testovací přístroje a tvorbu testu číslicového systému pro jednotlivé etapy.
- Možnost vzdáleného testování. Servisní technik může testovat vzdáleně připojený systém přes komunikační kanál, například Internet.
- Znalosti pracovníka o testovaném systému mohou být minimální, protože pracuje pouze s testy v nadřazeném řídicím systému.

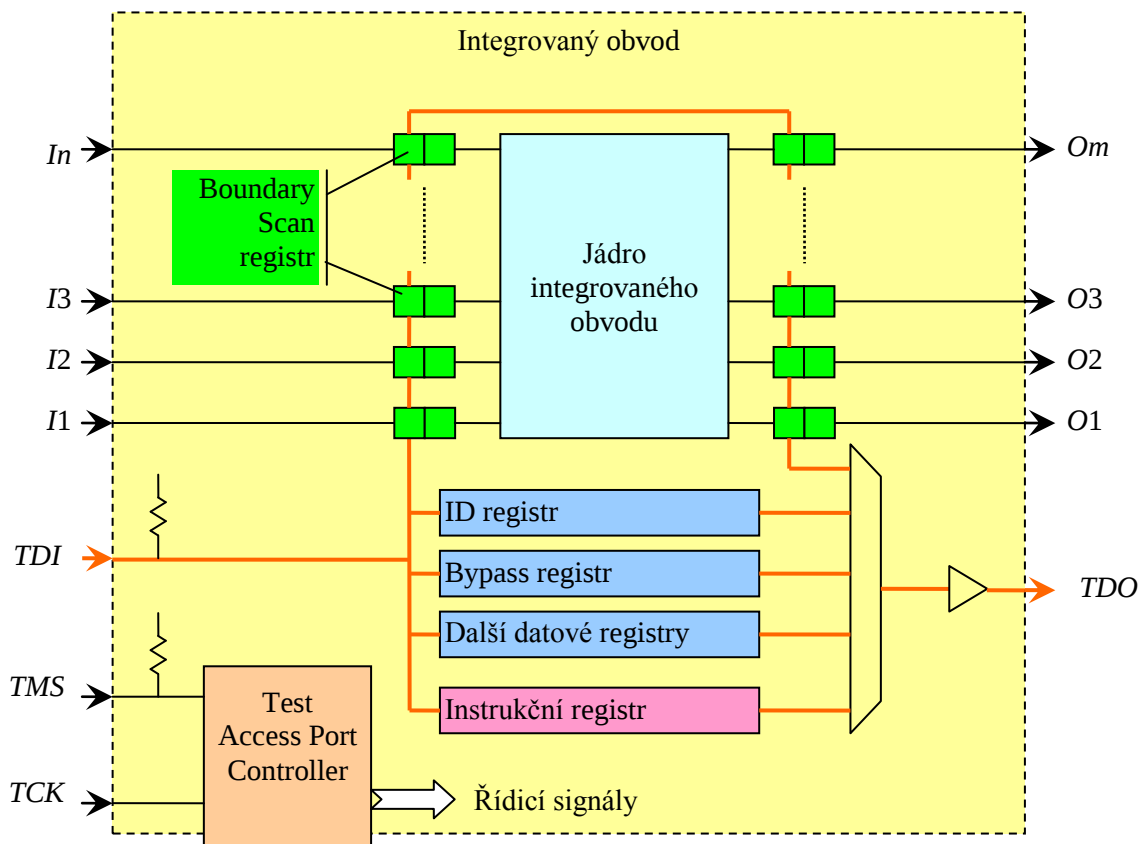
BS architektura má definované povinné prvky, doporučené a uživatelské prvky. Doporučené nebo-li volitelné prvky standard popisuje, použití závisí na výrobci obvodu. Uživatelské prvky vlastní návrhy výrobce, které zapracovává do Boundary Scan architektury. Použití těchto prvků je na výrobci, většinou se jedná o vylepšení testování, ladění procesorů. Typickým příkladem uživatelských prvků je ISC - In System Configuration. Tato technologie vznikala v 90-letech minulého století na bázi uživatelských prvků a standardem se stala v roce 2003. Skutečné vlastnosti BS architektury závisí na aplikovaných prvcích architektury v konkrétním integrovaném obvodu.

**Povinné,  
volitelné a  
uživatelské  
prvky**

Mezi povinné prvky BS architektury patří:

- TAP – Test Access Port, jedná se o rozhraní, pomocí kterého BS architektura v integrovaném obvodu komunikuje s nadřazeným systémem.
- TAP Controller, jedná se o řadič, který řídí činnost BS architektury.
- Instrukční registr, jedná se o registr, který udává akce, jaké bude BS architektura vykonávat. Jsou to testování interní nebo externí logiky, snímání dat apod.
- Soubor datových registrů potřebných pro testování. Jedná se hlavně o Boundary Scan registr.

**Povinné prvky**



**Architektura  
Boundary Scan**

Obr. 10 - 1 Boundary Scan architektura

Boundary Scan architektura a její aplikace v integrovaném obvodu je na obr. 10 - 1. Bloky BS architektury žádným způsobem neovlivňují navrženou činnost obvodu a slouží pouze k testování. Integrovaný obvod s BS architekturou pracuje ve dvou režimech, funkčním a testovacím. Funkční režim je dán navrženou funkcí a BS architektura ji žádným způsobem neovlivňuje. V testovacím režimu je BS architektura aktivována a je možné do zapojení zadávat hodnoty signálů a snímat hodnoty signálů integrovaného obvodu.

Základem BS architektury je buňka Boundary Scan registru, která je umístěná u každého pinu. Tato buňka musí být transparentní pro normální činnost obvodu. Ale v testovací režimu musí umožňovat vzorkovat vstupní data, vnucovat svá data do výstupu, zpřístupňovat svůj obsah. Právě pro přístup k datům Boundary Scan registru, ale i ostatních registrů byl zvolen sériový přístup. Potom jednotlivé registry BS architektury jsou sériové registry, kdy čtení a zápis jejich obsahu se provádí sériově pomocí signálu *TDI*, *TDO* a hodinového signálu *TCK*. Počet bitů jednotlivých registrů závisí na konkrétním typu integrovaného obvodu. Registry se dělí na instrukční registr a datové registry.

Instrukční registr je povinný registr a uchovává instrukce, které definují akce Boundary Scan testování a zároveň určuje, který datový registr bude zařazen do sériového řetězce *TDI* a *TDO*. Datové registry slouží k uchovávání informací o testování nebo dalších pomocných informacích. Datové registry jsou:

**Instrukční  
registr**

- Povinné datové registry jsou:
  - BS registr - Boundary Scan registr je umístěný na vstupech a výstupech integrovaného obvodu a slouží při testování k vnucení testovacích vzorků a snímání odezev. Pokud se testování neprovádí, potom registr žádným způsobem neovlivňuje navrženou činnost obvodu. **Boundary Scan registr**
  - Bypass registr je vždy jednobitový registr a slouží k vyřazení daného integrovaného obvodu ze sériového řetězce *TDI* a *TDO*. **Bypass registr**
  - ID registr – IDentification registr, je 32 bitový identifikační registr, který obsahuje kód výrobce integrovaného obvodu a kód konkrétního typu obvodu. Je přístupný pouze pro čtení. **ID registr**
- BS architektura umožňuje definovat další doporučené datové registry:
  - User Code registr, který obsahuje uživatelem definovaný kód. V případě PLD prvku se může jednat o kód verze návrhu. **User Code registr**
  - Registry ISC architektury, slouží k programování obvodu.
  - Další registry definované výrobcem integrovaného obvodu.

Dalším povinným blokem je TAP řadič – Test Access Port Controller, který řídí všechny činnosti Boundary Scan architektury uvnitř integrovaného obvodu. TAP řadič je synchronní sekvenční obvod, který má vstupní signály *TMS* a *TCK*. Výstupní signály řadiče řídí bloky BS architektury.

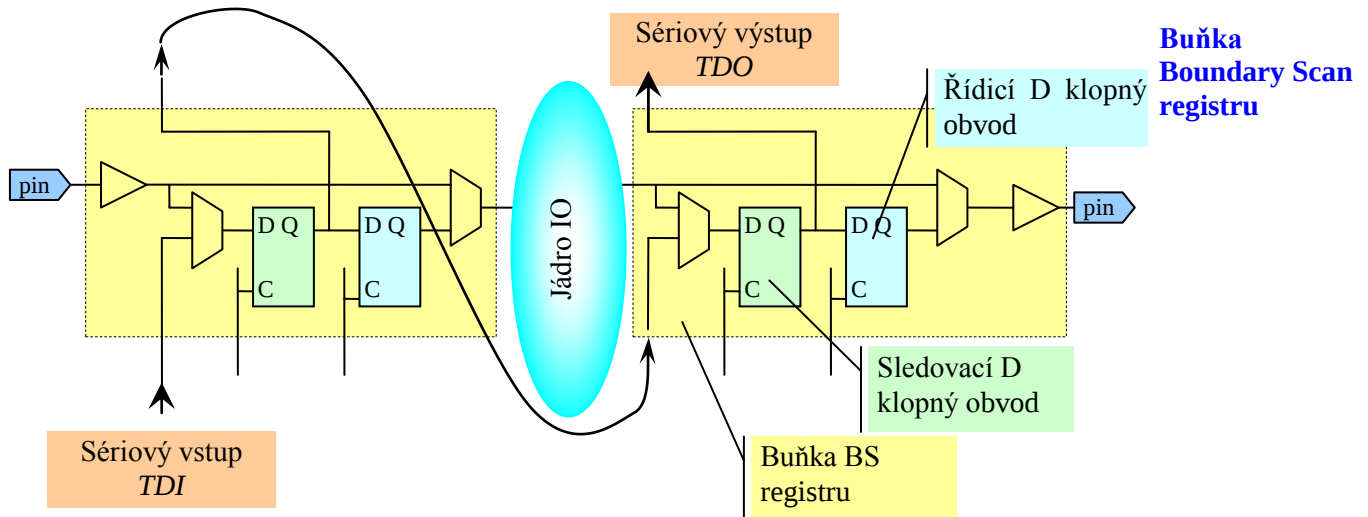
**TAP řadič**

TAP – Test Access Port je povinný interface, pomocí kterého Boundary Scan obvody integrovaného obvodu komunikují s nadřazeným systémem. TAP je vlastně interface a představuje signály:

- *TDI* – Test Data In, sériový vstup dat, které jsou synchronizovány hodinovým signálem *TCK*.
- *TDO* – Test Data Out, sériový výstup dat, které jsou synchronizovány hodinovým signálem *TCK*.
- *TCK* – Test Clock, hodinový synchronizační signál.
- *TMS* – Test Mode Select, řídicí signál pro TAP řadič.

- TRST – Test ReSeT, nepovinný signál pro asynchronní nulování Boundary Scan architektury.

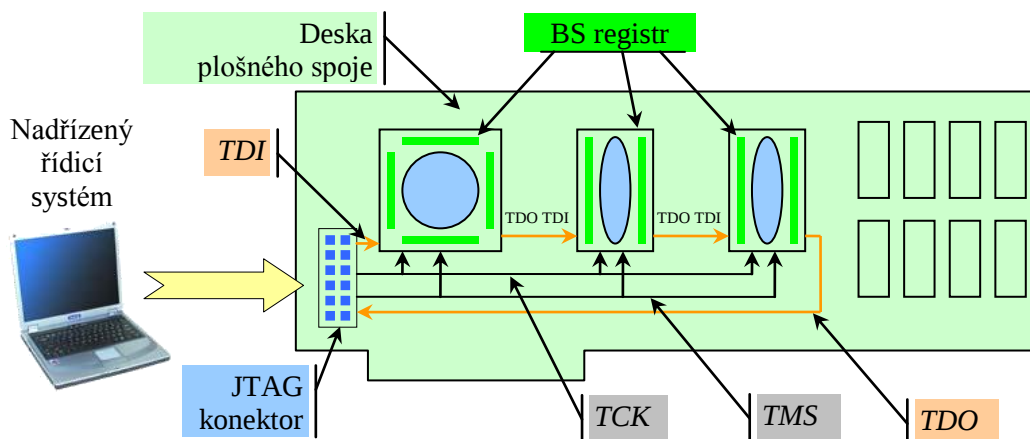
Základ BS vychází ze zásad strukturovaného návrhu a testování ICT – In Circuits Test. Základem je povinný Boundary Scan registr, který je umístěný u pinů integrovaného obvodu. Zapojení jednoho bitu BS registru je více a je dáno standardem, nejčastěji používané zapojení pro vstup a výstup je na obr. 10 - 2.



Obr. 10 - 2 Zapojení Boundary Scan registru

Zapojení pro vstup a výstup jsou stejná, liší se pouze v napojení na okolí. Aplikace Boundary Scan registru nijak neovlivňuje navrženou činnost obvodu. Pouze se využívá pro testování. Buňka BS registru se vlastně skládá ze dvou klopných obvodů, první s názvem sledovací – Capture DFF a druhý řídicí – Control DFF. Sledovací část vytváří sérioparalelní registr. Sériový princip se používá pro čtení a zápis obsahu z nadřazeného řídicího systému a paralelní princip k paralelnímu sejmutí dat vstupujících do buňky, tj. z pinu integrovaného obvodu nebo z jeho jádra. Řídicí část vytváří paralelní registr, do kterého se přepisují data ze sledovacího registru. Přepis je řízen TAP řadičem. Výstup řídicího registru pak může budit signál do jádra integrovaného obvodu nebo na výstupní pin.

Základem testování je možnost Boundary Scan registru sejmut paralelní data na vstupu a paralelní vnucení dat na výstup. Čtení sejmutých dat a zápis nových dat se provádí sériovým způsobem do sledovací části BS registru. Data pro zápis se umísťují na signál TDI a sejmutá data jsou čtena na signálu TDO.



Obr. 10 - 3 Zapojení BS na desce plošného spoje

Zapojení BS architektury na desce plošného spoje je na obr. 10 - 3. Signály *TMS* a *TCK* se k jednotlivým integrovaným obvodům připojují paralelně a signály *TDI* a *TDO* vytvářejí sériové propojení. Do sériového řetězce mezi signály *TDI* a *TDO* mohou být zařazeny různé datové registry BS architektury, proto sériový řetězec může být tvořen stejnými nebo různými registry několika integrovaných obvodů. V každém integrovaném obvodu je do řetězce zařazen pouze jeden registr. Pokud integrovaný obvod v dané fázi testování nemá význam, potom se do řetězce *TDI* a *TDO* zařazuje Bypass registr daného integrovaného obvodu. Připomínám, že Bypass registr je dlouhý 1 bit. Instrukční registr každého integrovaného obvodu určuje, který datový registr bude zařazen do sériového řetězce *TDI* a *TDO*. K instrukčnímu registru se přistupuje u všech integrovaných obvodů současně a přístup je dán stavem TAP řadiče.

Součástí DPS je i konektor pro připojení nadřazeného číslicového systému, který provádí testování. Tento konektor je často označován jako JTAG konektor a v případě PC se napojení provádí přímo na paralelní port s vhodným driverem nebo přes speciální blok, který mění PC interface na BS interface. Může se jednat USB – BS interface a podobné varianty. Nadřazený číslicový systém řídí průběh testování a ve svých pamětech má testovací vzorky pro danou DPS. BS architektura umožňuje i testovat shluky integrovaných obvodů, které nemají v sobě zapracovanou BS architekturu. Předpokládá se, že vstup a výstup tohoto shluku je zapojen na integrované obvody s BS architekturou.

**JTAG  
konektor**

Propojení nadřazeného řídicího systému pomocí paralelního rozhraní PC se používá pro jednoduché varianty, kdy se nevyžaduje vysoká přenosová kapacita na signálech *TDI* a *TDO* BS architektury. Propojení pomocí speciálních bloků je určeno pro náročné řešení s vysokou přenosovou kapacitou BS architektury.

BS architektura obsahuje instrukční registr, a tím pádem instrukce, které definují jednotlivé činnosti. Instrukce se dělí na tři skupiny, povinné, doporučené a vlastní, definované výrobcem. Jaké principy testování a akcí lze provádět v integrovaném obvodu, vyplývá z konkrétně realizovaných instrukcí v daném PLD obvodu.

Mezi povinné instrukce patří:

- EXTEST, umožňuje testování propojení mezi jednotlivými obvody a testování shluků. Instrukce pracuje s BS registrem.
- SAMPLE, vzorkuje paralelní data do sledovacího registru a provádí sériový posuv dat ve sledovací části BS registru. Jedná se o čtení sejmutých dat a zároveň zápis nových dat pro následný test.
- PRELOAD, instrukce se používá k přepisu dat ze sledovací části do řídicí části BS registru – Capture DFF to Control DFF. Po ukončení sériového posuvu se nová data přepíší do řídicí části BS registru.
- BYPASS, zařazuje do řetězce *TDI* a *TDO* Bypass registr, integrovaný obvod zůstává ve funkčním stavu. Jedná se vlastně o vyřazení integrovaného obvodu ze sériového řetězce *TDI* a *TDO*, protože délka Bypass registru je 1 bit.

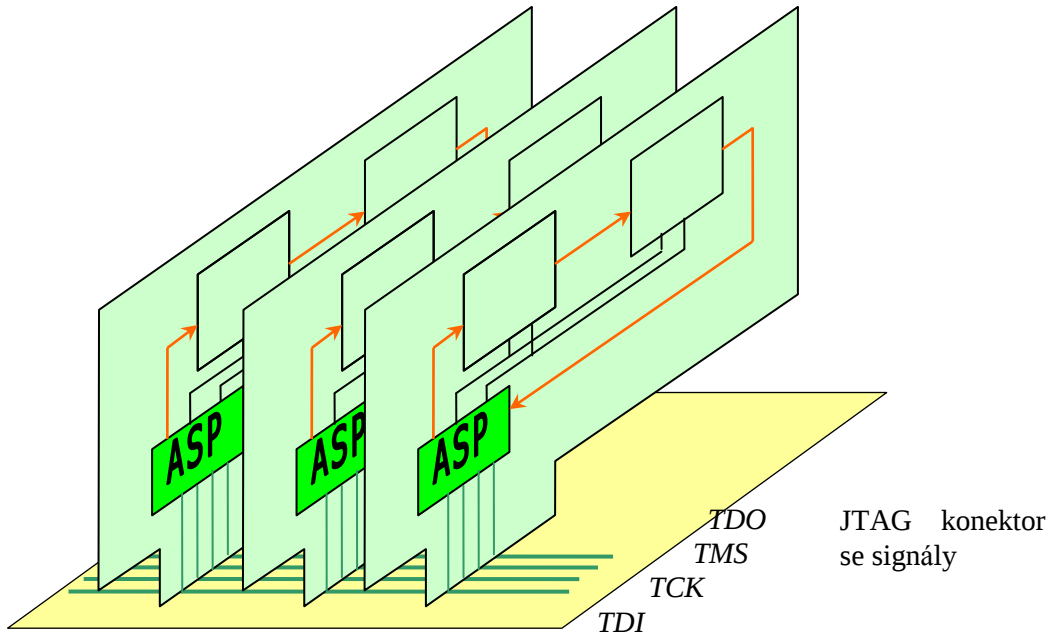
**Instrukce  
Boundary Scan**

Mezi nepovinné instrukce patří:

- INTTEST, umožňuje testovat jádro integrovaného obvodu. Instrukce pracuje s BS registrem.
- RUNBIST, spouští vestavěné testování, které je zapracováno do integrovaného obvodu principem BIST.
- CLAMP, instrukce zajistí, že výstupní signály jsou buzeny obsahem řídicí části BS registru.
- HIGHZ, instrukce nastaví výstupní budiče do třetího stavu.
- ID CODE, instrukce čte obsah ID registru.
- USER CODE, instrukce čte a zapisuje obsah User code registru.

BS architektura je dále rozpracována pro testování číslicového systému, který se skládá z několika desek plošných spojů, obr. 10 - 4. V tomto případě deska plošného spoje obsahuje řadič BS architektury - ASP, kdy na jedné straně je TAP interface, který je rozveden po desce plošného spoje a na druhé sběrnicové řešení pomocí signálů *TDI*, *TDO*, *TMS* a *TCK*. Toto zapojení umožňuje provádět testování desek plošných spojů autonomně. Tím testy mohou běžet paralelně, čímž dojde k výraznému zkrácení doby testování celého číslicového systému.

**Boundary Scan  
v systému**



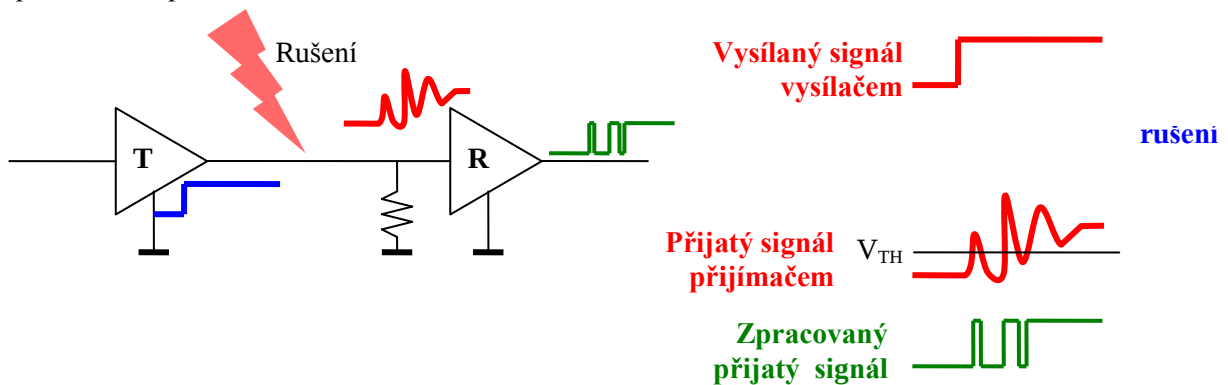
Obr. 10 - 4 Aplikace BS architektury v číslicovém systému

BS architektura může mít v každém typu integrovaného obvodu jiné vlastnosti. Jedná se o zapojení buňky BS registru a jeho délky, délky instrukčního registru, instrukční kódy pro nepovinné instrukce a další vlastnosti. Všechny tyto vlastnosti jsou popsány v BSDL souboru pro každý integrovaný obvod.

## 10.2 Testování I/O pinů

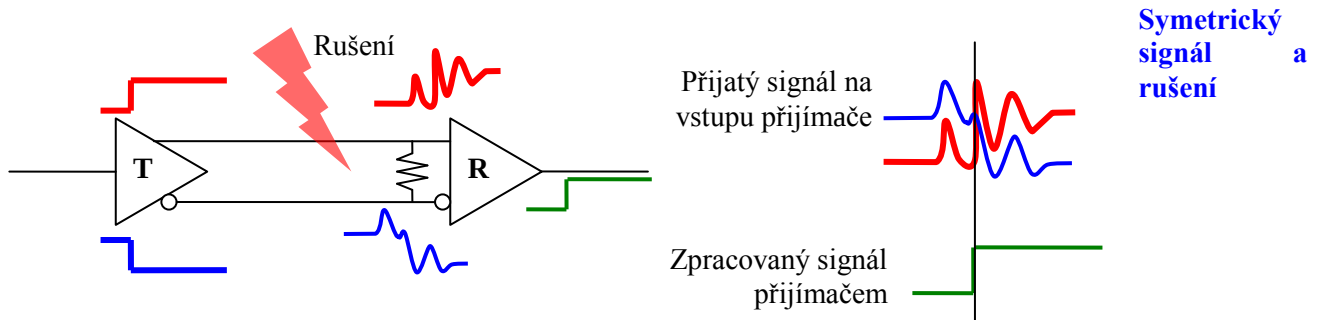
Při přenosu dat o vysokých rychlostech řádově Gbity za sekundu se jednoznačně používá sériový princip. Upouští se o paralelního principu, kde je obtížné zvládnout časování přenosu, rozdílový čas - skew time. Dále je to neúměrný nárůst potřebných počtu pinů na integrovaném obvodu. O této skutečnosti například svědčí přechod od paralelního připojení pevných disků v osobním počítači k sériovému rozhraní – SATA.

**Vylepšené  
testování I/O  
pinů**



Obr. 10-5 Přenos signálu nesymetrickým vedením

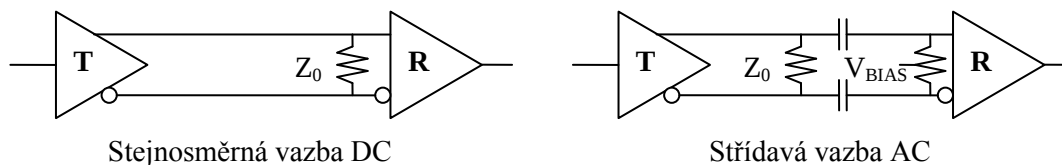
Pro tyto sériové přenosy se dává přednost použití diferenciálního signálu z důvodů rušení. Na obr. 10 – 5 je vidět přenos nesymetrického signálu a vliv rušení. Přijatý signál má zákmity v důsledku rušení. Zákmity jsou způsobeny definicí překlápěcího napětí  $V_{TH}$  u přijímače.



Obr. 10-6 Přenos signálu symetrickým vedením

Na obr. 10 - 6 je vidět přenos diferenciálního nebo-li symetrického signálu a vliv rušení. Diferenciální signál přenáší informaci v rozdílu dvou úrovní na dvou vodičích. Dá se předpokládat, že rušení ovlivňuje oba vodiče stejně, protože jsou taženy v blízkosti sebe. Rozdíl napětíových úrovní však zůstává stejný. Přijatý signál je roven vyslanému signálu. Rozdíl je způsobem tím, že v případě nesymetrického přenosu je jeden vodič pevný, má konstantní napětíovou úroveň.

Vysílač a přijímač mohou být propojeny stejnosměrnou – DC nebo střídavou – AC vazbou. Střídavá vazba AC používá kapacitu k oddělení vysílače a přijímače, obr 10-7.



Obr 10 – 7 Stejnosměrná DC a střídavá AC vazba

Z pohledu testování propojení vysílače a přijímače potřebuje k testování střídavý signál. Digitální Boundary Scan podle 1149.1 umožňuje používat pouze stejnosměrné signály a tím vznikl problém jak testovat střídavé propojení. Proto vznikl standard 1149.6 vylepšené testování I/O pinů. Tento standard nazývá na původní standard 1149.1 digitální Boundary Scan a doplňuje jej o možnost generovat střídavý signál na straně vysílače a jeho vyhodnocování na straně příjmu. V důsledku těchto změn dochází k definici nových buněk Boundary Scan registru pro vysílání a vyhodnocování na straně příjmu. Detailní popis je v literatuře [Parker 2003].

Nárůst logiky z důvodů vylepšeného testování v integrovaném obvodu je malý. Nutno si uvědomit, že počet speciálních pinů určených k takovému přenosu je malý, řádově jednotky maximálně desítky pinů. Většinou se jedná o piny, které souvisejí s doplňujícími obvody typu SERDES. Pouze u těchto pinů se aplikuje vylepšené testování I/O pinů podle 1149.6.

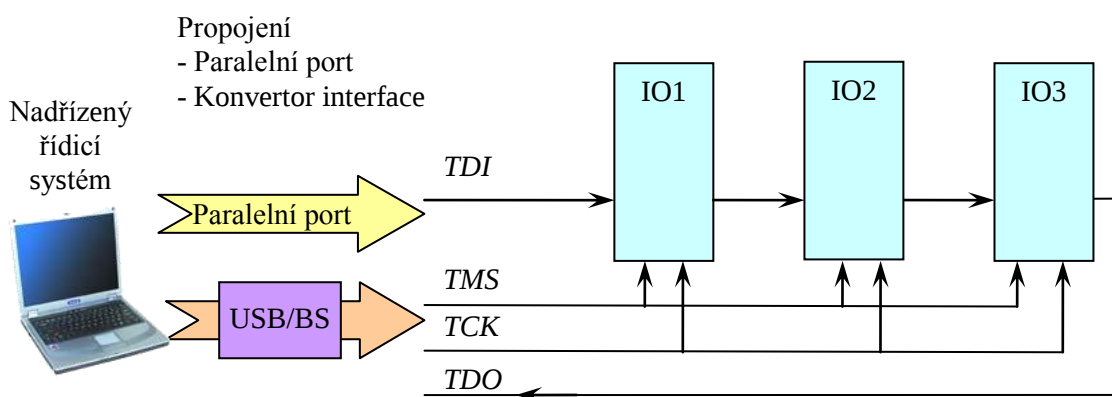
### 10.3 Systém programování ISP, ISC

Technologický rozvoj výroby integrovaných obvodů časem umožnil vytvářet na přímo čipu zdroje napětí potřebné pro programování. Proto speciální programátory zanikají a moderní programovatelné prvky vytvářejí tyto zapojení v integrovaném obvodu na křemíku. Tento technologický pokrok vede ke vzniku ISP – In System Programming, dnes se ujmá nové označení této technologie ISC – In System Configurable. Jedná se technologii programování osazených integrovaných obvodů na deskách plošných spojů. Odpadá nutnost používat speciální hardwarové programátory, stačí se pouze k integrovanému obvodu připojit a nahrát do něj data.

Technologie ISC je dána standardem IEEE 1532 a je rozšířením BS architektury. Význam ISC lze spatřovat ve všech fázích vzniku výrobku a to ve vývoji, výrobě a servisu.

- Pro oblast vývoje je to hlavně zkrácení jeho celkové délky, a tím zkrácení času uvedení výrobku na trh. Jinými slovy, jedná se o možnost provádět návrh desek plošných spojů a obsahů PLD obvodů současně.
- Pro oblast výroby je to hlavně redukce výrobních nákladů, redukce počtu skladových položek a zlepšení parametrů spolehlivosti navrhovaného systému. Pro programování PLD obvodů nejsou potřebné speciální hardwarové programátory. Ve skladě stačí skladovat pouze čisté, nenaprogramované obvody nikoliv jako dřív stejné PLD obvody, ale s různým obsahem. PLD obvody nemusí být osazovány do patič z důvodu přístupu pro testování, upgradu obvodu novým obsahem nebo vytváření konfigurací na přání.
- Pro oblast servisu a podpory je to možnost upgradu a oprav ze vzdáleného stanoviště. Princip upgradu zároveň umožňuje vytvářet konfigurace systému na přání. Princip vzdáleného přístupu umožňuje spravovat, opravovat systém na dálku, tj. servisní pracovník se připojí k systému pomocí počítačové sítě ze servisního centra bez nutnosti vycestování.

### Význam ISC



Obr. 10 - 8 Použití ISC technologie

Základní zapojení pro ISC programování integrovaných obvodů je na obr. 10 – 8. Jedná se o klasické připojení pomocí BS architektury, protože ISC je jejím rozšířením. Ze systému BS používá stejný interface pro napojení nadřízeného systému. Rozšíření BS architektury spočívá v doplnění o další datové registry sloužící k programování. Naproti tomu ISC systém může využívat prvky BS architektury a to hlavně ID – identifikační registr a User Code registr. ISC může kontrolovat, zda jsou osazeny správné integrované obvody pomocí ID registru. Dále do User Code registru umístit kód obsahu PLD prvku a zajistit PLD prvek bezpečnostní propojkou – Security Fuse.

### Použití ISC

Výrobci PLD prvků nabízejí pro své vlastní PLD prvky programovací systém, většinou i ve verzi volně dostupné.

Se systémem ISC úzce souvisí jednak popis registrů a postupů programování PLD prvku a také tvar dat obsahu PLD prvku. K popisu ISC systému v PLD prvku se používá BSDL popis a tento popis umožňuje při programování kombinovat různé PLD prvky od různých výrobců. ISC také definuje tvar dat, které definují obsah PLD prvku. Tyto soubory jsou výstupem návrhových systémů a dnes se používají formáty:

- JEDEC, jedná se o nejstarší standard souboru obsahující data pro programování PLD prvku. Je to ASCII soubor, který lze textovými editory číst.

- ISC soubor, tvar dat definovaný standardem IEEE 1532.

## 10.4 BSDL jazyk

BSDL – Boundary Scan Description Language je jazyk pro popis BS architektury a je dán standardem IEEE 1149.1b, bližší popis je v literatuře [Bleeker, Eijnden, Jong 1993]. BSDL jazyk vychází z VHDL jazyka. V každém PLD prvku může být BS architektura implementována jiným způsobem, například délka jednotlivých registrů, kódy nepovinných instrukcí, zapojení BS registru a přiřazení k jednotlivým pinům, datové registry definované výrobcem apod. Proto ke každému prvku je k dispozici BSDL soubor, který detailně popisuje implementaci BS architektury v PLD prvku.

**BSDL -  
Boundary Scan  
Description  
Language**

BSDL soubor pro daný prvek je nutné mít v případě, kdy provádím testování, tvorbu testů nebo programování.

## 10.5 IBIS model

IBIS je model popisující vlastnosti vstupních hradel, výstupních budičů a pinů integrovaného obvodu, literatura [Actel 2002]. Jedná se voltampérové charakteristiky, parazitní indukce a kapacity spojené s pinem, strmost hran – Slew rate apod. IBIS model neobsahuje časové charakteristiky pro časovou simulaci. Nutno si uvědomit, že návrhové systémy desky plošných spojů dokáží simulovat přenosy mezi jednotlivými integrovanými obvody. Výsledkem simulace může být upozornění na špatné přizpůsobení vedení, eliminace elektromagnetické kompatibility (EMC) apod. IBIS model se dodává ve tvaru souboru, který je sestaven pro každý PLD prvek.

**IBIS**

## 10.6 VITAL

Ve vývojových systémech PLD prvků se provádí časové simulace chování navrženého číslicového systému v konkrétním PLD prvku. Automaticky se předpokládá popis navrženého číslicového systému pomocí jazyků VHDL nebo Verilog. Z toho důvodu je nutné vývojovému systému zadat časové modely konkrétních PLD prvků. Za druhé, jedná se o náročnost a časovou délku výkonu simulace u konkrétního prvku. Při použití standardních možností jazyka VHDL je tato časová simulace neúměrně dlouhá. Proto bylo přikročeno k vytvoření standardu IEEE 1076.4 s názvem VITAL – VHDL Initiative Toward ASIC Libraries, který řeší tyto problémy. VITAL obsahuje specializované rutiny pro popis chování a časovou specifikaci ASIC buněk. Detailní znalost standardu VITAL je důležitá pro výrobce PLD obvodů, kteří vytvářejí knihovny pro jednotlivé prvky. Pro uživatele je důležité mít simulátory, které jsou VITAL kompatibilní, to je VITAL specifikace je zapracována přímo do jádra simulátoru. Uživatelé pouze využívají VITAL specifikaci. Bližší informace jsou v literatuře [Internet 04], [Internet 05].

**VITAL -  
VHDL  
Initiative  
Toward  
ASIC  
Libraries**

## 10.7 Intellectual Properties – IP

IP - Intellectual Properties je program, který organizují výrobci PLD obvodů nebo jiné společnosti zabývající se vývojem číslicových systémů. Jako zákazník si mohu v rámci programu koupit již navržený číslicový systém zapsaný v jazyku VHDL nebo Verilog. Účelem je zkrácení vývoje nového zapojení nákupem již vyřešeného bloku a ne vymyšlením toho, co je již dobře známo a vyřešeno. Použitím bloků z IP návrhář zkracuje časovou délku návrhu nového číslicového systému.

**IP –  
Intellectual  
Properties**

Výrobci PLD obvodů shromažďují vyřešené bloky číslicových systémů, které potom nabízejí svým zákazníkům. Mezi vyřešenými bloky můžeme najít PCI interface, UART - sérioparalelní převodník dat pro RS 232, Ethernet adaptér 10/100 Mbit/s, Ethernet adaptér 1 Gbit/s, šifrovací a dešifrovací algoritmy, soft procesory a jiné.

## 10.8 PCI kompatibilní

Jedná se o PCI – Peripheral Component Interconnect, sběrnici personálního počítače v dnešních PC počítačích. PCI byla prvně definována společností Intel v roce 1993 a dnes je definice sběrnice spravována organizací pod názvem PCI SIG – PCI Special Interest Group, která ji dále rozvíjí a vydává její specifikaci. Bližší informace najdete v literatuře [Internet 03], [Altera 1995].

**PCI  
kompatibilní**

Specifikace PCI sběrnice vzhledem k PLD prvkům specifikuje elektrické a přenosové parametry výstupních budičů a vstupních přijímačů. Jedná se např. o:

- Elektrické parametry, jako je definice napětíových L a H úrovní, proudové zátěže, voltampérové charakteristiky vstupu a výstupu.
- Přenosové vlastnosti, jako jsou parazitní indukce a kapacity související s piny integrovaného obvodu, voltampérové charakteristiky vstupu a výstupu, a tím i charakteristické přenosové impedance, skew time jednotlivých budičů, strmost hran signálů a další.
- Všechny výstupní budiče a vstupní přijímače musí být v jednom pouzdru.

## 10.9 Pouzdra

PLD prvky se vyrábějí v různých pouzdrech, které mají různý počet pinů. Mezi nejstarší pouzdro patří DIP pouzdro, ve kterém se vyráběly nejstarší PLD prvky. Dnes se s ním můžeme setkat u GAL obvodů. CPLD a FPGA obvody mají již počet pinů vyšší a používají jiné pouzdra. V současnosti se vyrábí FPGA obvod s počtem 1 508 pinů v pouzdru BGA. Běžná pouzdra mají stovky pinů. Za základní vlastnosti pouzdra lze považovat:

**Pouzdra**

- Konstrukční rozměry pouzdra, jako je jeho plošná velikost, uspořádání pinů, rozteč mezi piny a jejich tvar.
- Ztrátový výkon pouzdra, tepelný odpor pouzdra. Jedná se o elektrický příkon jaký dokáže pouzdro uchládit. Nutno se uvědomit, že elektrický příkon PLD prvků je několik wattů a špičkově může být až desítky wattů.
- Možnost uchycení chladičů.

Výběr pouzdra pro konkrétní realizaci většinou závisí na technologii výroby desky plošného spoje. Detailní specifikaci pouzder lze nalézt u každého katalogového listu PLD prvku.

## 10.10 Provozní podmínky

Provozní podmínky určují vlastnosti okolního prostředí, kdy při jejich splnění je výrobcem garantována správná činnost integrovaného obvodu. Jedná se o podmínky pro teplotu pouzdra nebo PN přechodu uvnitř integrovaného obvodu, vlhkost, otřesy, radiační záření apod. Mezi základní podmínku patří teplota přechodu  $T_j$  a teplota okolí  $T_A$ . Vztah mezi těmito teplotami je dán tepelným odporem pouzdra. Provozní podmínky se označují názvem prostředí a jedná se o:

**Provozní  
podmínky**

- Commercial, komerční prostředí. Běžné podmínky se kterými se setkáváme u elektronických zařízeních pro běžné použití. Někdy se toto prostředí označuje jako kancelářské prostředí. Teplota okolí  $T_A$  je 0 °C až 70 °C. Některé společnosti definují toto prostředí teplotou přechodu  $T_j$ , například 0 až +85 °C (90 °C).
- Industrial, průmyslové prostředí. Jedná se o těžší podmínky, které se mohou vyskytnout v průmyslovém prostředí. Toto prostředí je charakterizováno teplotou okolí  $T_A$  je -40 °C až +85 °C. Některé společnosti definují toto prostředí teplotou přechodu  $T_j$ , například -40 °C až +100 °C (+105 °C).

- Automotive, automobilní prostředí. Teplota přechodu  $T_j$  je  $-40\text{ }^{\circ}\text{C}$  až  $+140\text{ }^{\circ}\text{C}$ . Teplota okolí  $T_A$  je  $-40\text{ }^{\circ}\text{C}$  až  $+125\text{ }^{\circ}\text{C}$  ( $+150\text{ }^{\circ}\text{C}$ ).
- Military, vojenské prostředí. Teplota okolí  $T_C$  je  $-55\text{ }^{\circ}\text{C}$  až  $+125\text{ }^{\circ}\text{C}$ .

Jiná definice provozních podmínek je podle standardů QPL a modernějšího QML, literatura [QPL 1993] a [QML 2003]. QPL standardy definují vlastnosti integrovaných obvodů podle kategorie prostředí. QML standardy definují místo prostředí kvalitativní třídy elektronických součástek. Aplikace těchto standardů se hlavně využívá pro vojenskou, raketovou a kosmickou techniku. Tyto standardy definují i vlastní značení obvodů.

## Otázky kap. 10



- 10.1 Popište digitální Boundary Scan.
- 10.2 Popište vylepšené testování I/O pinů, hlavně důvody.
- 10.3 Popište význam BSDL, VITAL, IBIS souborů a pojmu PCI kompatibilní.
- 10.4 Popište technologii ISC – In System Configuration.
- 10.5 Popište princip IP.

## Odměna a odpočinek



V roce 1985 vznikl jazyk pro návrh číslicových systémů pod názvem Verilog. Tento jazyk byl standardizován až v roce 1995 jako IEEE 1364. Jazyk Verilog je porovnáván s C jazykem a VHDL jazyk s jazykem ADA nebo Pascalem. V roce 2005 došlo k další modifikaci jazyka Verilog na SystemVerilog jako standard IEEE 1800. SystemVerilog obsahuje specifické metody návrhu, vylepšené nastavení jazyka, jazyk pro tvorbu testů včetně pokrytí a nastavení, API – aplikační programový interface a DPI – přímý programový interface. SystemVerilog je považován za první sjednocený HDVL jazyk – Hardware Description and Verification Language, jazyk pro popis a verifikaci hardwaru. Bližší informace lze nalézt na [Internet 01], [Internet 02], [Internet 07] a [Internet 08].

Dnes se můžeme setkat s novým prostředkem pod názvem SystemC. Tento jazyk navazuje na původní jazyk C++, který je doplněn o knihovny pro popis číslicových obvodů na systémové úrovni. Knihovny zavádějí nové definice datových typů, a prvků jazyka jako je signál, interface, port, procesy apod. V současnosti je volně dostupný SystemC ve verzi 2.0, který je zároveň standardem IEEE 1666. Dnes se připravuje verze 3.0, která má umožňovat kombinovaný návrh pomocí hardwarových a softwarových prostředků. SystemC je vyvíjený neziskovým sdružením OSCI – Open SystemC Initiative jako otevřený standard (open source) pro systémový návrh číslicových systémů. Bližší informace na [Internet 09] a [Internet 10].

V oblasti návrhu zapojení se už koncem 80. let používají vývojové systémy a objevuje se trend, kdy každá významná společnost vyrábějící PLD obvody si vytvoří svůj návrhový systém.

## 11. ZÁKLADNÍ VLASTNOSTI PLD PRVKŮ

**Čas ke studiu:** 3 hodiny



**Cíl** Po prostudování této kapitoly budete umět



- Význam a použití pojmů, se kterými se lze setkat při použití programovatelného prvku. Jedná se hlavně o elektrické vlastnosti související s použitím.

**Výklad**



U PLD prvků se setkáváme s různými pojmy, které určují konkrétní vlastnosti. V této kapitole se budeme zabývat jejich jednotlivými vlastnostmi bez ohledu, u kterých typů PLD prvků se s nimi setkáváme.

### 11.1 Bezpečnostní propojka - Security Fuse

Jedná se o bezpečnostní propojku, která zabraňuje neoprávněnému kopírování obsahu prvku. Tento princip lze chápat jako ochranu duševního vlastnictví, ochranu proti krádeži zapojení. Po naprogramování prvku bez bezpečnostní propojky lze obsah PLD číst a kontrolovat správnost naprogramování (akce programátoru Verify). Po nastavení bezpečnostní propojky již nelze číst skutečný obsah PLD prvku. Bezpečnostní propojka se nuluje pouze, když se maže celý obsah prvku.

**Bezpečnostní  
pojistka –  
Security Fuse**

### 11.2 Nulování po náběhu napětí - Power-Up Reset

PLD prvky s paměťovými elementy nemusejí mít speciální signály pro nastavení a nulování paměťových elementů. Po náběhu napětí se paměťové elementy automaticky nulují. Skutečný výstup na pinu PLD prvku je pak závislý na nastavení aktivity výstupu. Tato vlastnost se hlavně využívá u sekvenčních obvodů, kdy se nulový stav paměťových elementů využívá jako výchozí stav sekvenčního obvodu.

**Nulování po  
náběhu  
napětí**

### 11.3 Elektronický podpis - Electronic Signature

Elektronický podpis je paměťové slovo, které je součástí PLD prvku a slouží hlavně k identifikaci obsahu PLD prvku. Elektronický podpis může být např. 64 bitové slovo a jeho obsah určuje návrhář. Elektronický podpis může obsahovat kód obsahu prvku včetně verze, adresu návrháře apod. S pojmem elektronický podpis se setkáváme u GAL obvodů. U CPLD a FPGA obvodů se k stejnému účelu používá User Code registr, který je součástí BS architektury. Elektronický podpis lze číst pouze na programátoru nebo pomocí BS architektury a ISC.

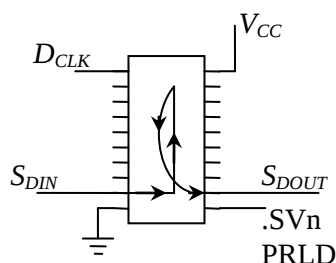
**Elektronický  
podpis -  
Electronic  
Signature**

### 11.4 Nahrání registru – Register Preload

S touto vlastností se setkáváme u PLD prvků typu PAL a GAL. Jedná se výhradně o diagnostickou vlastnost, kdy je možné při testování nahrát do paměťových elementů

**Nahrání  
registru**

definovanou hodnotu a z této hodnoty vykonat test. Tato vlastnost je hlavně vhodná pro testování sekvenčních obvodů s větším počtem stavů. Nahrání registru se provádí za speciálního stavu PLD prvku, kdy na definovaný pin se připojí zvýšené napětí, v ABEL-HDL jazyku se jedná o aplikaci tečkového rozšíření .SVn. Samotné nahrání informace může být paralelní nebo sériové. Této vlastnosti se dá hlavně využít při testování PLD prvku v hardwarovém programátoru. V aplikačním zapojení se téměř nevyužívá. Na obr. 11 - 1 je znázorněné sériové nahrání registru pro prvek GAL.

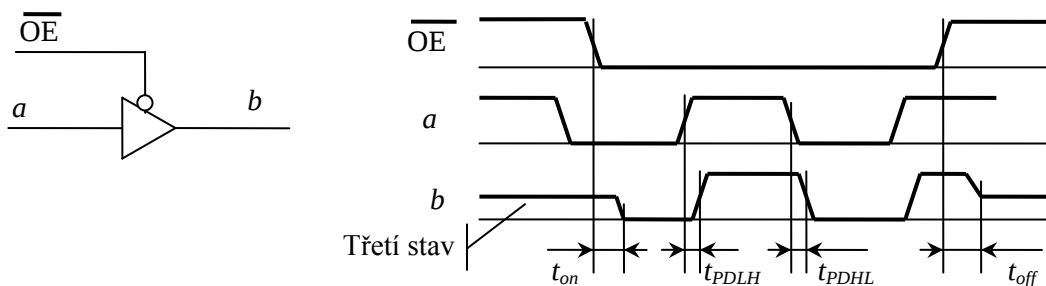


Obr. 11 - 1 Princip sériového nahrání obsahu registru GAL16V8

### 11.5 Třístavový výstup

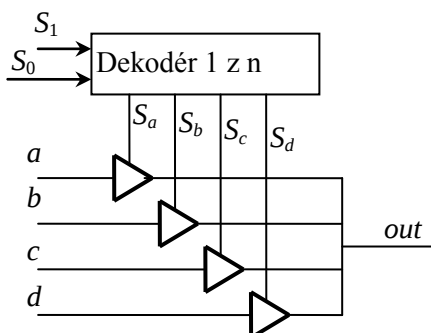
Jedná se o speciální zapojení výstupu hradla, kdy oba výstupní tranzistory jsou zavřeny. Tento stav hradla se také označuje jako stav vysoké impedance nebo zkráceně Z stav. Třetí stav je řízen speciálním signálem, často označovaný jako OE – Output Enable, povolení výstupu. Aplikace třístavových výstupů u hradel umožňuje propojování výstupu hradel.

**Třístavový výstup**



Obr. 11 - 2 Budič s třístavovým výstupem

Třístavových výstupů se hlavně využívá pro návrh obousměrných sběrnicových signálů, jako náhrada multiplexerů apod. Schématická značka budiče třetího stavu včetně časového průběhu je na obr. 11 - 2. Povolovací signál je aktivní v L úrovni, po dobu L úrovně výstup budiče odpovídá aktivnímu stavu a výstup odpovídá vstupnímu signálu, po dobu H úrovně je výstup v třetím stavu. V obrázku jsou i zakreslená zpoždění s označením jednotlivých časů.



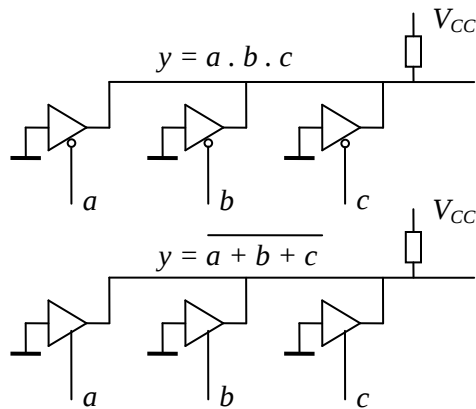
Pravdivostní tabulka dekodéru 1 z n

| $S_1$ | $S_0$ | $S_a$ | $S_b$ | $S_c$ | $S_d$ |
|-------|-------|-------|-------|-------|-------|
| L     | L     | H     | L     | L     | L     |
| L     | H     | L     | H     | L     | L     |
| H     | L     | L     | L     | H     | L     |
| H     | H     | L     | L     | L     | H     |

**Multiplexer pomocí třístavových výstupů**

Obr. 11 - 3 Multiplexer 4 na 1 pomocí budičů s třetím stavem

Použití třístavového výstupu v aplikaci jako multiplexer 4 na 1 je na obr. 11 - 3. Multiplexer je obvod, kde výstup odpovídá jednomu ze vstupních signálů na základě výběrových signálů. Kombinace výběrových signálů  $S_0$  a  $S_1$  určují jeden ze vstupů, kterému bude odpovídat výstup. Budič odpovídající vybranému vstupu bude v aktivním stavu a zbývající ve třetím stavu. Výstupy ve třetím stavu neovlivňují aktivní výstup. Situaci, aby pouze jeden z budičů byl aktivní a ostatní ve třetím stavu zajišťují signály  $S_a$  až  $S_d$ , které jsou v kódu 1 z  $n$ . Pouze jeden ze čtyř signálů má povolovací úroveň pro budič s třetím stavem.



Pravdivostní tabulka pro TS budič

| OE | In | Out |
|----|----|-----|
| H  | X  | TS  |
| L  | L  | L   |
| L  | H  | H   |

Pravdivostní tabulka pro TS budič

| OE | In | Out |
|----|----|-----|
| L  | X  | TS  |
| H  | L  | L   |
| H  | H  | H   |

Logické operace pomocí třístavového výstupu

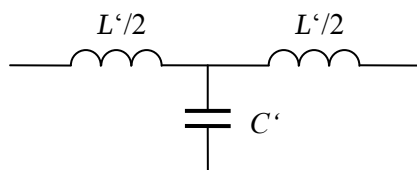
Obr. 11 - 4 Použití třetího stavu jako montážní součin

Zajímavý způsob je využití budičů s třetím stavem pro realizaci některých  $\mathcal{B}$ -výrazů. Jedná se o vytvoření montážního součinu a negace montážního součtu. Typ je dán podle aktivní úrovně, která povoluje třetí stav.

## 11.6 Vedení nebo spoj

Spoje mezi jednotlivými hradly mohou být chápány pouze jako spoje s ohmickým odporem nebo jako vedení. Nutno si uvědomit, že světlo urazí za 1 ns vzdálenost 30 cm ( $\tau=3,3$  ns/m), což už může odpovídat spojům na desce plošného spoje. U vedení je tato vzdálenost ještě menší. U bezztrátového přenosového vedení (ohmická složka je nulová) je náhradní model uveden na obr. 11 - 5 včetně důležitých vzorců.

Vedení nebo spoj



$L'$  - je charakteristická indukce na jednotku délky

$C'$  - je charakteristická kapacita na jednotku délky

Impedance vedení  $Z_0 = \sqrt{\frac{L'}{C'}}$

Zpoždění vedení  $\tau = \sqrt{L' C'}$

Obr. 11 - 5 Náhradní schéma bezztrátového vedení

Indukce a kapacita se zadává na jednotku délky, např. cm, potom impedance vedení je v Ohmech na cm a zpoždění v sekundách na cm. V číslicových systémech se můžeme setkat s vodiči se základními parametry, které jsou uvedené v tabulce na obr. 11 - 6. Z tabulky je vidět, že elektrický signál je pomalejší než světlo a uražené vzdálenosti za jednotku času budou menší, literatura [Becke, Forstner, Haseloff, Huchzermeier 1997].

Hodnoty pro vybrané vedení

| Typ spoje                    | $L'$ [nH/cm] | $C'$ [pF/cm] | $Z_0$ [ $\Omega$ ] | $\tau$ [ns/m] |
|------------------------------|--------------|--------------|--------------------|---------------|
| Spoj (daleko od GND)         | 20           | 0,06         | 600                | $\approx 4$   |
| Twistovaný pár               | 5 – 10       | 0,5 – 1      | 80 – 120           | 5             |
| Plochý kabel                 | 5 – 10       | 0,5 – 1      | 80 – 120           | 5             |
| Spoj na desce plošných spojů | 5 – 10       | 0,5 – 1,5    | 70 – 100           | $\approx 5$   |
| Koaxiální kabel              | 2,5          | 1,0          | 50                 | 5             |
| Sběrníkový vodič             | 5 – 10       | 10 – 30      | 20 – 40            | 10 - 20       |

Zdroj „Digital Design Seminar“ fy Texas Instrument

Obr. 11 - 6 Typické hodnoty vybraných spojů

Proto vzniká otázka, od jaké délky spoje je vhodné uvažovat o spoji jako o vedení. V praxi se doporučuje použít pravidlo, literatura [Becke, Forstner, Haseloff, Huchzermeier 1997] :

- Když je spoj delší než podíl času náběžné hrany signálu a dvojnásobku zpoždění, vztah (11.1), je vhodné použít teorii vedení. V opačném případě je možné spoj chápat jako ohmický nebo spoj s čistou kapacitní zátěží.

**Vedení nebo spoj - vzoreček**

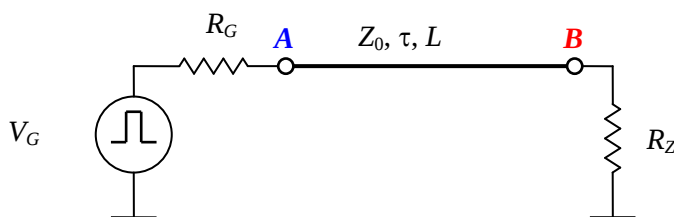
$$L = \frac{t_r}{2 \times \tau} \quad (11.1)$$

kde  $t_r$  je čas náběžné hrany signálu, který je spojem přenášen,

$\tau$  je zpoždění vedení na jednotku délky.

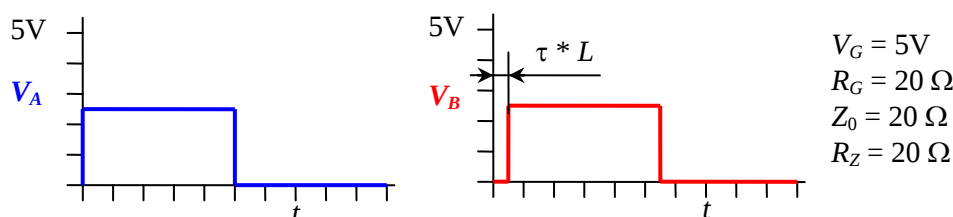
Na obr. 11 - 7 je zapojení idealizovaného vedení pro účely simulace, které se vyznačuje tím, že vedení je bezztrátové a odpor zdroje i zátěže je lineární. Ve skutečnosti jsou tyto odpory nelineární. Zdroj napětí je ideální a generuje impuls o napětí 5 V,  $R_G$  je vnitřní odpor zdroje,  $Z_0$  je impedance vedení,  $R_Z$  je zatěžovací odpor vedení. Ideálem z hlediska vedení je, aby všechny impedance byly stejné, a tím se dosáhne ideálního přenosu bez deformace impulsu, obr. 11 - 8.

**Hodnoty pro vybrané vedení**



Obr. 11 - 7 Náhradní zapojení pro simulaci na vedení

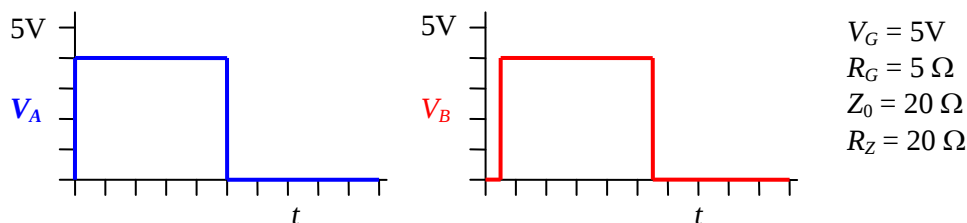
Na obr. 10 - 8 je průběh napětí na začátku vedení  $V_A$  a na konci vedení  $V_B$  v případě přizpůsobeného vedení. Přenášený impuls je bez deformace a napětíová úroveň přenášeného impulsu je poloviční vůči zdroji napětí. Napětíová úroveň je dána děličem z odporů  $R_G$  a  $Z_0$ .



**Ideálně přizpůsobené vedení**

Obr. 11 - 8 Odporové přizpůsobení vedení – varianta a

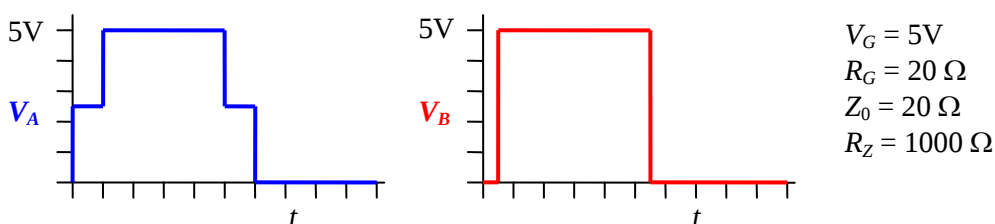
V praxi se dost často vyskytuje situace, že podmínku odporového přizpůsobení na začátku a konci vedení nelze splnit. I tyto situace se dají využít k přenosu impulsů bez deformace. Toto je často nemožné a v praxi jsou využitelné i jiné kombinace znázorněné na dalších obrázcích. Na obrázcích je znázorněn průběh napětí na začátku a konci vedení, bod A a B. Signál v bodě B je zpožděn.



**Přizpůsobení  
vedení  
pouze  
konci** na

Obr. 11 - 9 Odporové přizpůsobení vedení pouze na konci – varianta b

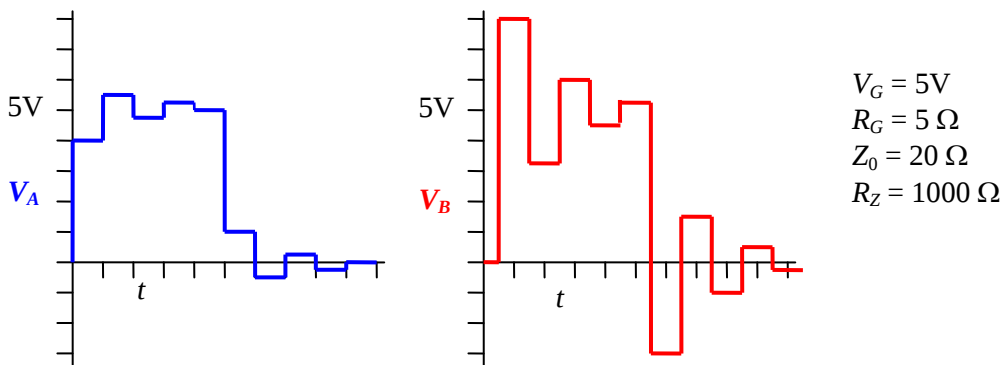
Na obr. 11 – 9 je situace, kdy na vstupu není odporové přizpůsobení, odpor  $R_G < Z_0$  a konec vedení je odporově přizpůsoben,  $Z_0 = R_Z$ . Impuls je přenášen bez deformace, ale napěťová úroveň je vyšší a je dána odporovým děličem z odporů  $R_G$  a  $Z_0$ . Na obr. 11 – 10 je situace, kdy vstup vedení je odporově přizpůsoben  $R_G = Z_0$  a na konci vedení je odpor zátěže mnohonásobně vyšší než impedance vedení,  $Z_0 < R_Z$ . U této situace dochází k deformaci impulsu pouze na vstupu vedení, ale na výstupu je impuls bez deformace. První napěťový skok na vstupu vedení je vždy poloviční napětí zdroje a druhý je blízký napětí zdroje. Napěťová úroveň impulsu na konci vedení je blízká zdroji napětí. I tato kombinace se dá využít k přenosu impulsu, protože na výstupu je impuls bez deformace. Tato situace se také často vyskytuje u sběrnicových vodičů.



**Přizpůsobení  
vedení  
pouze  
začátku** na

Obr. 11 - 10 Odporové přizpůsobení vedení pouze na začátku – varianta c

Pokud vedení není odporově přizpůsobeno ani na začátku ani na konci vedení dochází k deformaci přenášeného impulsu překmity do plusových i minusových hodnot napětí, obr. 11 - 11. Toto je reálná situace při propojení výstupu hradla se vstupem následujícího hradla, když si uvědomíme, že výstupní odpor hradla je malý a vstupní odpor hradla velký, hlavně u CMOS technologie.



**Nepřizpůsobení  
vedení**

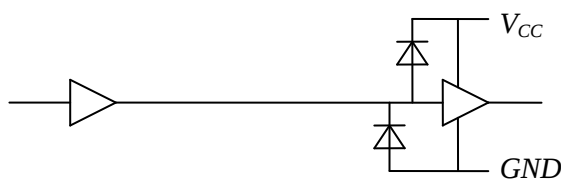
Obr. 11 - 11 Odporové nepřizpůsobení vedení – varianta d

Uvedené průběhy platí pouze pro lineární prvky. V praxi jako zdroj pro vedení používáme hradlo nebo budič s nelineární voltampérovou charakteristikou výstupu. Reálné vedení má odporovou složku je ztrátové. Zatěžovací odpor vedení také nemusí být lineární. Průběhy signálů na vedení pro lineární obvody je možné vypočítat podle lattice diagramu [Becke, Forstner, Haseloff, Huchzermeier, 1997] nebo pro nelineární a lineární obvody grafickou metodou podle Bergeron diagramu [Becke, Forstner, Haseloff, Huchzermeier, 1997], [Texas 1996a].

## 11.7 Omezující diody – Clamping Diode

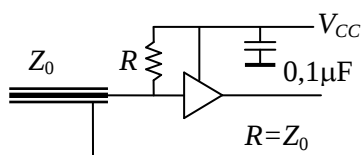
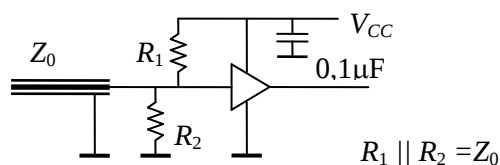
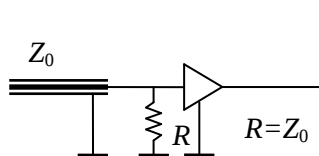
Omezující diody – clamping diode je aplikace diod na omezení překmitů signálů, které vznikají na nepřizpůsobeném vedení. Situace na obr. 11 - 11 se běžně vyskytuje v praxi. Jedná se o propojení výstupu hradla se vstupem hradla bez přizpůsobovacích odporů. Překmity jsou také dány nelineárními charakteristikami vstupu a výstupu hradel. Napětí, které je mimo toleranci, může zničit vstup nebo výstup hradla. Proto se používají omezující diody podle obr. 11 - 12.

**Omezující diody – Clamping Diode**



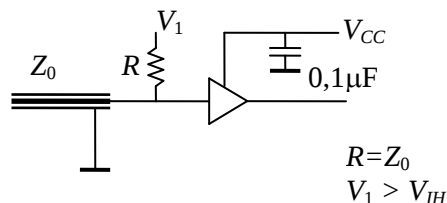
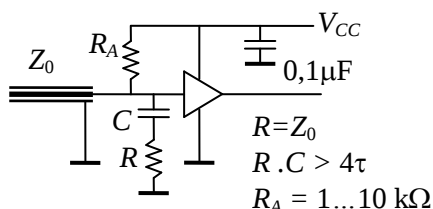
Obr. 11 - 12 Aplikace omezujících diod

Překmitý přesahující napájecí napětí jsou svedeny diodou nahoru do napájecího napětí a překmity pod 0 V jsou svedeny diodou dolů do zemního vodiče. U současných číslicových obvodů jsou většinou vstupy vybavené omezujícími diodami. Pokud ne, je možné použít sólo omezujících diod nebo speciálních obvodů pro omezení překmitů - clamping circuit.



$$V_{CC} \frac{R_2}{R_1 + R_2} \geq V_{IH}$$

**Odporové přizpůsobení vedení**



Obr. 11 - 13 Odporové přizpůsobení na konci vedení

## 11.8 Odporové přizpůsobení

Jiný způsob omezení překmitů je odporové přizpůsobení na výstupu vedení, aplikace situace podle obr. 11 - 9. Možné způsoby odporového přizpůsobení na konci vedení jsou na obr. 11 - 13.

Při výběru odporového přizpůsobení je nutno mít na paměti, že se zvyšuje proudové namáhání výstupu hradla. Např. pro vedení s  $Z_0 = 100 \Omega$  při aplikaci jednoho odporu  $R$  buď na  $V_{CC} = 5 \text{ V}$  nebo  $GND$  teče výstupem hradla proud  $I_O$  nejméně o hodnotě  $V_{CC}/R = 5/100 = 50 \text{ mA}$ . Proto se raději volí přizpůsobení pomocí odporového děliče, kde tento proud je menší. Typický dělič pro  $Z_0 = 100 \Omega$  je  $R_1 = 180 \Omega$  a  $R_2 = 220 \Omega$ . Jiné řešení je použití dalšího zdroje s menším napětím, který také snižuje proud výstupem hradla. Toto řešení se často používá ve sběrnicevých systémech, například GTL budiče sběrnice.

**Odporové  
přizpůsobení  
vedení**

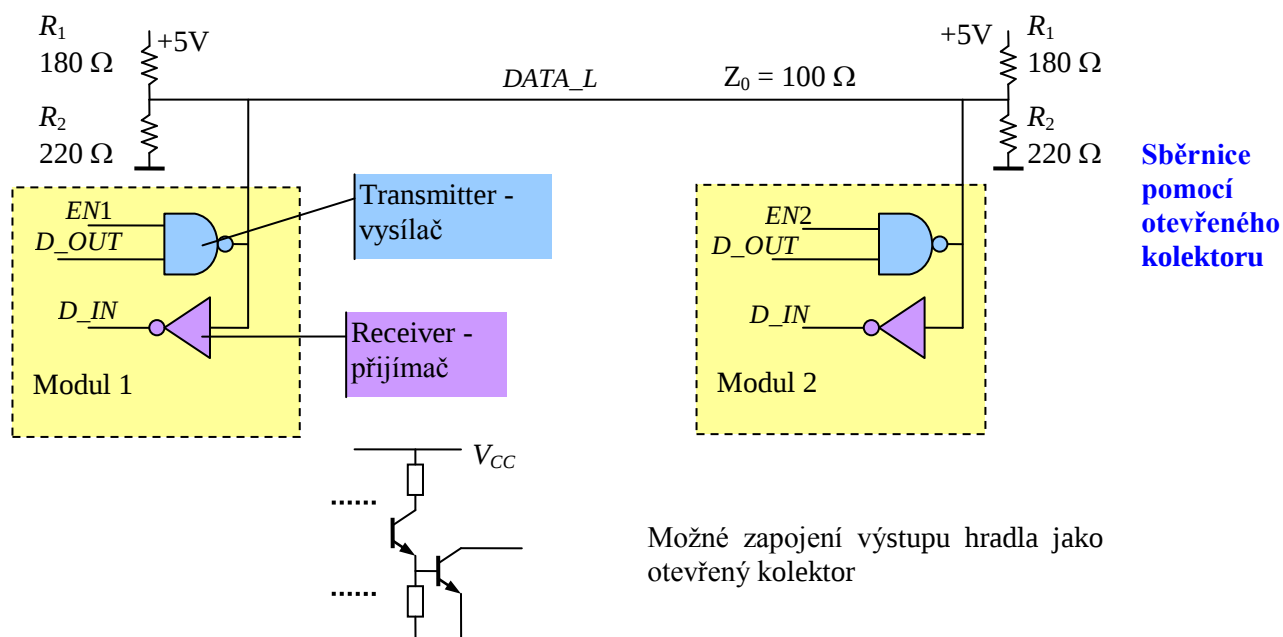
## 11.9 Sběrnice

Se sběrnicemi se setkáváme v mnoha číslicových systémech a slouží k přenosu dat mezi dvěma bloky. Základem je použití jednoho vodiče pro přenos dat oběma směry. Takový vodič nebo signál sběrnice se často označuje jako obousměrný. K realizaci se používají buď budiče s třetím stavem nebo budiče s otevřeným kolektorem. Z aplikací sběrnice se také setkáváme nejen k propojení bloků číslicových systémů, ale i jiných aplikacích, například paměťové pole.

**Sběrnice**

Sběrnicevodič realizovaný pomocí otevřeného kolektoru je na obr. 11 - 14. V uvedeném příkladu sběrnicevodič má impedanci  $Z_0 = 100 \Omega$  a využívá se hlavně pro dlouhé sběrnice. V klidovém stavu mají všechny výstupy vysílačů sběrnice H úroveň, což je zajištěno pomocí povolovacích signálů budičů v každém modulu  $ENx = L$ . Přenos dat se provádí tak, že pouze jeden vysílač je povolen ( $EN1 = H$ ), tím data na sběrnicevodiči odpovídají vysílaným datům (z modulu 1). Zbývající moduly mají vysílače nepovoleny, pouze mohou přijímat data ze sběrnicevodiče. Data se šíří zleva doprava. V opačné situaci, kdy bude pouze povolen vysílač na modulu 2, se data budou šířit zprava do leva. Proto se tomuto vodiči říká obousměrný.

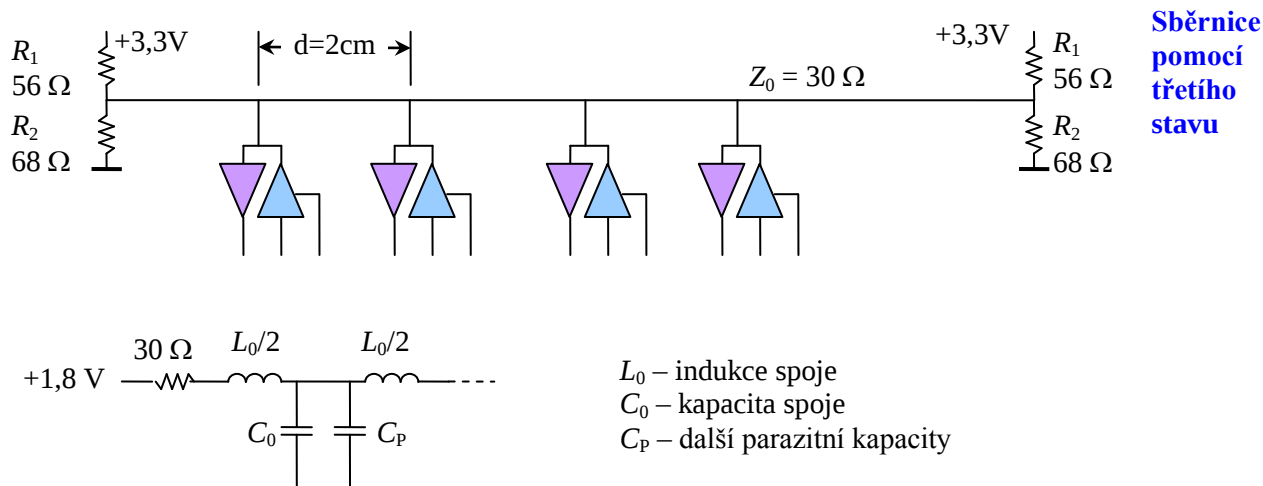
**Obousměrný  
přenos dat**



Obr. 11 - 14 Obousměrný sběrnicevodič pomocí otevřeného kolektoru

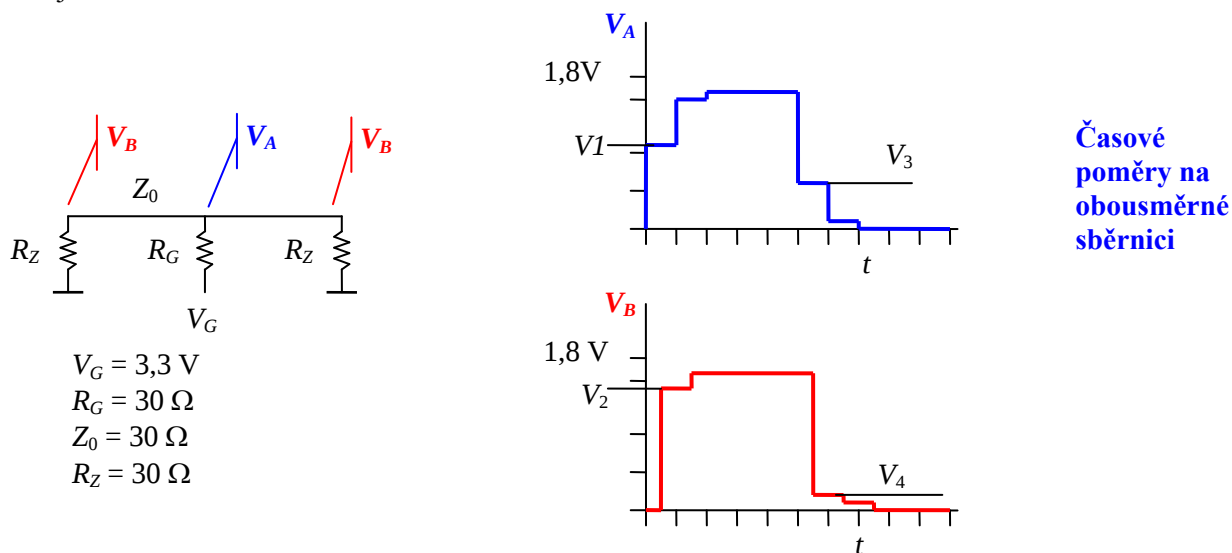
V příkladu je uvedena impedance vedení  $100\ \Omega$ , což odpovídá dvěma zkrouceným vodičům (twisted pair) a nejsou zde započteny další parazitní kapacity nebo indukce od vstupů a výstupů hradel, konektorů.

Jiné řešení obousměrného sběrnicevého vodiče je pomocí vysílačů s třetím stavem. Na obr. 11 - 15 je příklad, kdy sběrnicevému vodiči jsou připojeny ve velmi krátkých vzdálenostech vysílače a přijímače sběrnice. Pak se změní impedance vedení v důsledku dalších parazitních kapacit a indukci. Jedná se o parazitní kapacity a indukce vstupů a výstupů hradel, kapacity a indukce pinů konektorů a další.



Obr. 11 - 15 Krátká třístavová sběrnice

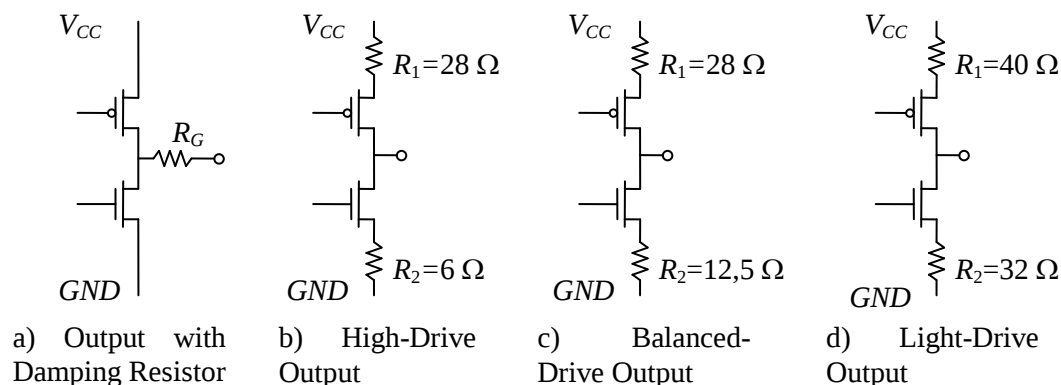
Z pohledu přenosu dat na sběrnicevému vodiči vzniká problém, že zdroj nemusí být na začátku vedení, ale může být uprostřed. Pro takovýto zdroj je potom impedance vedení poloviční, protože se jedná o paralelní kombinace impedance vedení. Modelovaný průběh při lineárních náhradách je na obr. 11 - 16.



Obr. 11 - 16 Tvar impulsů na sběrnici

V této situaci dochází k odrazům a na obrázku jsou vyznačeny důležité hodnoty napětí, které vznikají jako první odraz. Hodnoty  $V_1$  a  $V_2$  by měly být vyšší než je  $V_H$  - napětí H úrovně, hodnoty  $V_3$  a  $V_4$  by měly být menší než  $V_L$  - napětí L úrovně. V nejhorším případě lze definice napětí nahradit překlápěcím napětím hradla  $V_{TH}$  - threshold level. V případě nesplnění této

podmínky je nutné ke zpoždění signálu připočítat dvojnásobek zpoždění vedení. Jiný způsob řešení problému je  $R_G < Z_0$  a  $R_Z > Z_0$ , kdy na vedení vznikají napěťové překmity a ty se eliminují omezujícími diodami.



Obr. 11 - 17 Verze výstupů

Na trhu jsou sběrnicové budiče s třístavovým výstupem s různými vlastnostmi. Liší se hlavně zapojením výstupu, a tím i vlastnostmi výstupních voltampérových charakteristik, které ovlivňují odporové přizpůsobení vedení, obr. 11 - 17.

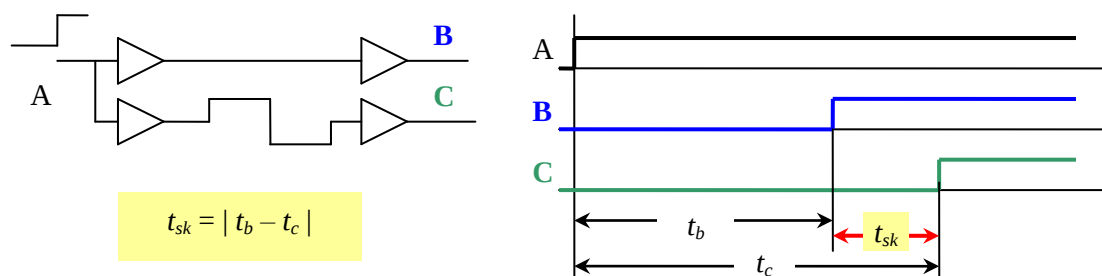
U verze a) se impedance výstupu přizpůsobuje impedanci vedení aplikací odporu  $R_G$ . Verze b) je zapojení, kdy odpory  $R_1$  a  $R_2$  reprezentují odpor kanálů FET tranzistorů. U verzí c) a d) je snaha tyto odpory zvětšit, a tím omezit energii pro tvorbu překmitů v signálu. Potom není nutné odporové přizpůsobení na koncích vedení.

Jiný způsob řešení problémů jsou speciální obvody navržené pouze pro sběrnicové použití. Jedná se o obvody označované jako GTL, GTLP, BTL.

## 11.10 Časový rozdíl - Skew Time

Skew time je časový rozdíl mezi dvěma přijatými signály v místě příjmu, které byly vyslány ve stejném čase a ze stejného místa, obr. 11 - 18. Jsou dva signály, mezi zdrojem a cílem. Každý signál se šíří v jiném prostředí, protože úplně identické prostředí neexistuje. Rozdílnost může být způsobena jinými okolními signály, různou délkou vodičů, různým zpožděním hradel na straně příjmu a vysílání apod.

Časový rozdíl – skew time



Obr. 11 - 18 Definice skew time

Skew time je zohledněn u každé sběrnice. Vzorkování dat ze sběrnice na straně příjmu musí zohledňovat skew time, tím se zaručí, že všechna data na datových vodičích budou při vzorkování platná.

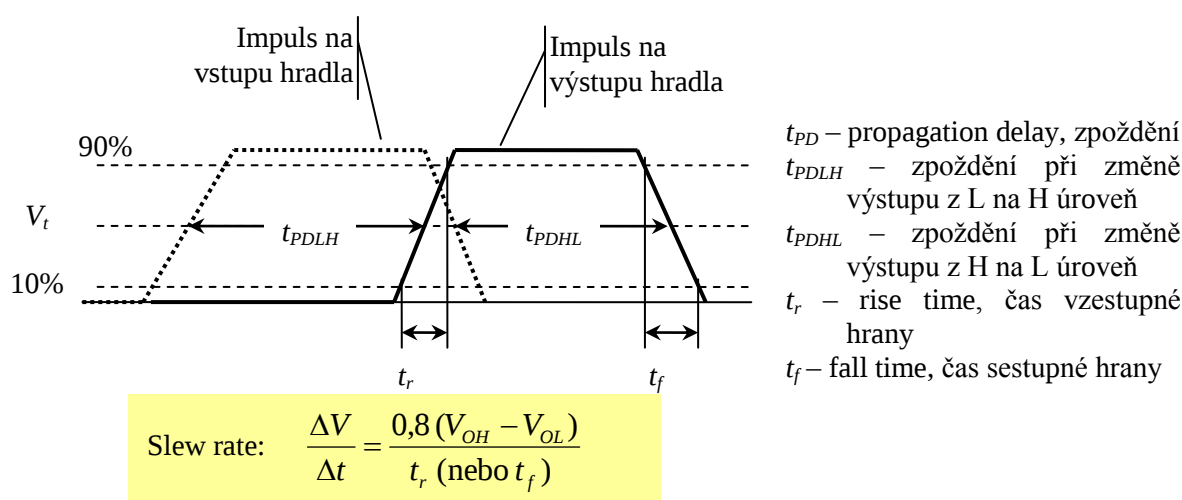
Skew time musí být minimalizován u každého sériového registru. Jedná se o rozvod hodinového signálu, kdy hrana u všech klopných obvodů by měla nastat v jednom čase. Potom je skew time nulový.

Skew time v hodinových rozvodech musí být minimalizován u každého sekvenčního obvodu jinak by mohlo dojít k narušení činnosti. Proto se u PLD prvků věnuje velká pozornost rozvodu hodinového signálu.

### 11.11 Strmost – Slew Rate

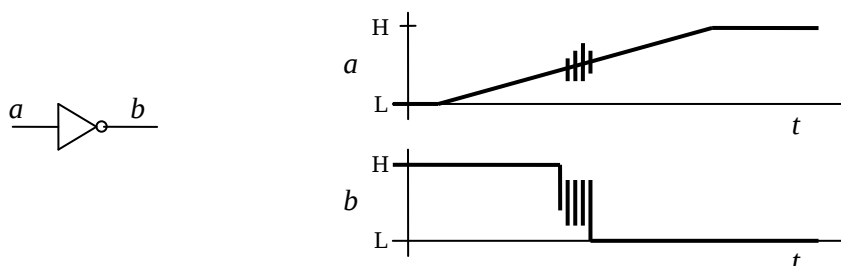
Strmost – slew rate

Skutečná hrana číslicového signálu trvá určitý čas, není nulová jak se obvykle kreslí. Čas trvání hrany je definován podle obr. 11 - 19. Strmost hrany u hradel je pak definována uvedeným vzorcem. Strmost u hradel je v rozmezí 0,5 až 2 V/ns a závisí na konkrétním typu.



Obr. 11 - 19 Strmost hrany – slew rate

U hradel je definované překlápěcí napětí  $V_{TH}$  – threshold voltage. Pokud strmost hrany je malá, může dojít k zámkům na vstupním a výstupním signálu hradla, obr. 11 - 20. Malá strmost hrany signálu  $a$  může být také způsobena velkou kapacitní zátěží na signálu  $a$ .



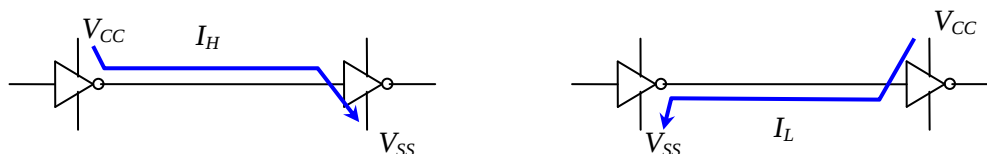
Obr. 11 - 20 Malá strmost hrany na vstupu hradla

Velká strmost hran také není výhodná, protože hrany generují přeslechy do jiných signálů v přímé úměře. Z pohledu elektromagnetické kompatibility je velká strmost hran zdrojem rušení. Vše vychází z Fourierovy řady, kdy spektrum pravoúhlého signálu má v porovnání se skosenými hranami mnohem vyšší amplitudy harmonických. Z pohledu elektromagnetické kompatibility je výhodná malá strmost hran.

### 11.12 Proudové poměry na vodiči

Směry proudu na vodiči přenášející číselný signál z výstupu hradla na vstup jsou znázorněny na obr. 11 - 21. Když je na vodiči úroveň H, potom proud  $I_H$  teče z výstupu hradla do vstupu. Když je na vodiči úroveň L, potom proud  $I_L$  teče ze vstupu hradla do výstupu. Při dnešních hradlech, kdy vstupy jsou realizovány pomocí CMOS tranzistorů, jsou tyto proudy malé.

**Směry proudů pro L a H úroveň**



Obr. 11 - 21 Směry proudu  $I_H$  a  $I_L$

Při bipolárních vstupech jsou tyto proudy mnohem vyšší. Ale v situaci, kdy k vodiči jsou připojeny odpory na napájecí napětí  $V_{CC}$  nebo  $V_{SS}$ , se hodnoty proudu výrazně mění. Tato situace nastává při odporovém přizpůsobení vedení na konci.

### 11.13 Velikost výstupního proudu

Každá logika má kromě definice úrovně L a H také definované velikosti proudu vstupu a výstupu. Tyto definice proudu potom určují kolik vstupů lze připojit na jeden výstup nebo jak použít přizpůsobovací odpory vedení na  $V_{CC}$  a  $GND$ , či jaké budou nabíjecí a vybíjecí proudy kapacitní zátěže. Nutno si uvědomit, že vstup CMOS hradla je kapacitní zátěž.

**Velikost výstupního proudu**

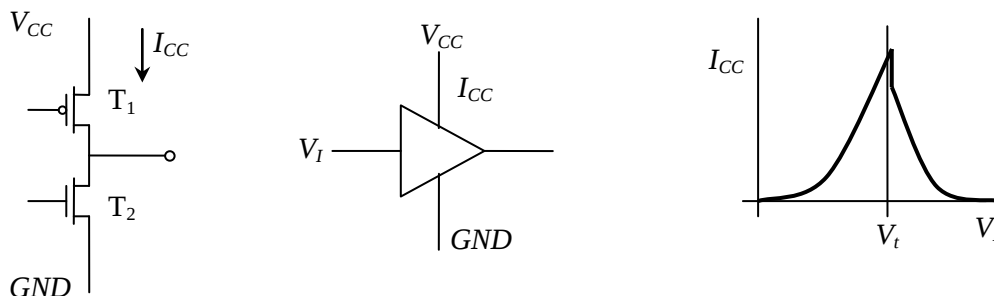
Výstupní budiče IO buněk mohou obsahovat programovatelnou vlastnost nastavení výstupního proudu  $I_{OH}$  a  $I_{OL}$ . Velikost proudu pro úroveň L a H se většinou nastavují na stejnou hodnotu v definovaných krocích. Skutečné hodnoty závisí na konkrétním typu PLD prvku. Rozsah nastavení velikosti výstupního proudu  $I_{OH}$  a  $I_{OL}$  bývá od 2 mA až 50 mA v několika krocích.

Omezení výstupního proudu se dá využít na omezení rušení do sousedních spojů, na eliminaci posuvu potenciálů v rozvodech napájení.

### 11.14 Napájecí proud hradla při změně výstupu

Funkce hradla z pohledu výstupu se dá zjednodušeně vysvětlovat tak, že pokud je na výstupu úroveň H, tranzistor  $T_1$  je otevřený a tranzistor  $T_2$  je uzavřen. Pokud je na výstupu úroveň L je situace opačná. Pokud je na výstupu třetí stav, jsou oba tranzistory zavřeny. Ale pokud se mění výstup z úrovně H na úroveň L, dochází k změně stavu tranzistorů za krátký čas. Ale po krátký časový okamžik jsou oba tranzistory částečně otevřeny a teče jimi proud, viz obr. 11 - 22, literatura [scla015].

**Proud hradla při změně výstupu**



Obr. 11 - 22 Proud hradla při změně výstupu

### 11.15 Rozvody napájecích napětí

V číslicových systémech je možné vygenerovat falešné impulsy pomocí nevhodných zemnicích vodičů. Tato situace může nastat u sběrnicových vysílačů. Vše vychází z poznatku, že všechny spoje mají indukci. Parazitní indukce jsou jednak v samotném integrovaném obvodu a také v rozvodu napájení na desce plošného spoje. Na obr. 11 – 23 je situace, kdy v jednom integrovaném obvodu je 8 budičů použito jako vysílač sběrnice a devátý budič je použitý obyčejný pro signál xy, který propojuje výstup a vstup hradla. Jedná se reálnou situaci, která může nastat v bloku IO buněk PLD prvku. Mezi zemnicím pinem integrovaného obvodu a jednotlivými budiči je spoj s indukcí  $L_{GND}$ . Uvedená indukce, i když je malá, dokáže za určitých podmínek na krátký čas vygenerovat falešný impuls na výstupu 9-tého budiče. Falešný impuls se dá pouze vysvětlit posuvem potenciálu v bodě A, čímž se úroveň L signálu xy se posune o  $\Delta V$  vůči GND. Vše začíná vygenerováním napětí  $\Delta V$  na indukci  $L_{GND}$ , které je přibližně dáno rovnicí (11.2).

**Rozvody  
napájecích  
napětí**

$$\Delta V \approx \frac{\Delta \Phi}{\Delta t} \approx L \frac{\Delta I}{\Delta t}, \quad (11.2)$$

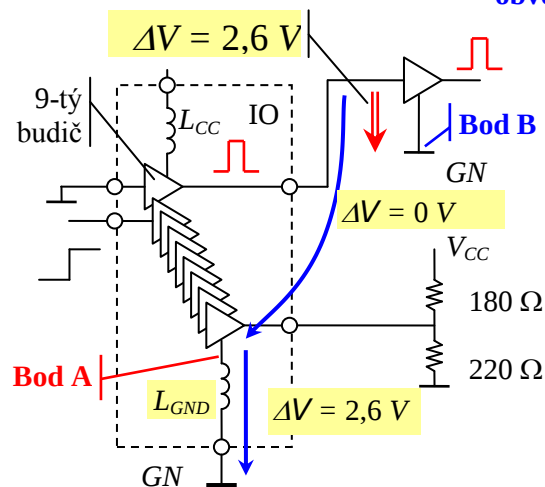
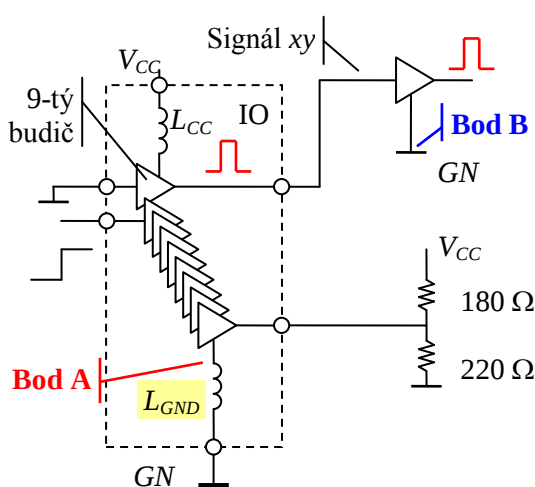
kde  $\Delta V$  je změna napětí, napěťový impuls,

$\Delta \Phi$  je změna magnetického toku

$L$  je indukce spojů,

$\Delta I$  je změna proudu,

$\Delta t$  je změna času.



**Situace v  
integrovaném  
obvodu**

Obr. 11 - 23 Impuls způsobený indukcí spoje v rámci integrovaného obvodu

Změna proudu je dána zdroji:

- Proudovým impulsem způsobeným změnou výstupu z úrovně H do úrovně L. Je-li pro jeden výstup proudový impuls 5 mA, potom při 8 výstupech je to 40 mA.
- Změnou směru proudu  $I_{OH}$  na  $I_{OL}$ . Tato změna je významná v případě, že na výstupu budičů jsou přizpůsobující odpory vedení a také počtem budičů. Při uvedeném ošetření je změna proudu  $8 \times 5V/180\Omega = 220$  mA.

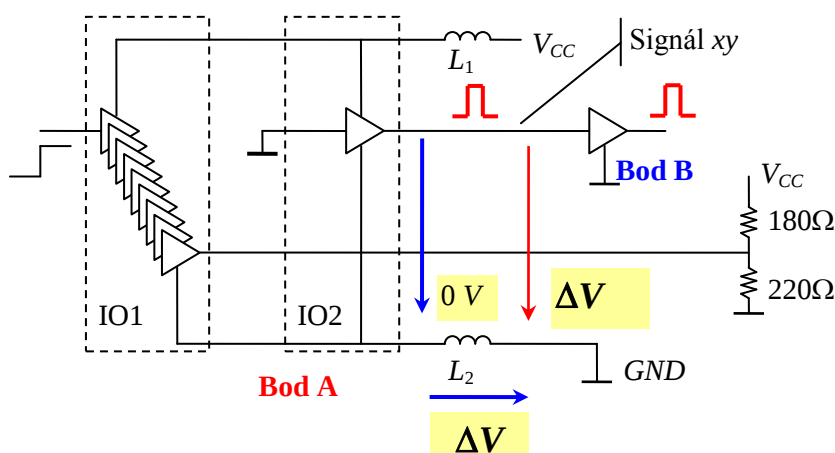
**Úvaha pro  
výpočet**

- Proudovými špičkami při nabíjení a vybíjení kapacitní zátěže. V následujícím příkladu s těmito změnami nebudeme počítat.

Za změnu času dosadíme čas hrany logického signálu, který je  $t_r = 1$  ns. Indukce v zemnicím spoji integrovaného obvodu odhaduji, řekněme  $L_{GND} = 10$  nH. Potom

$$\Delta V \approx 10 \cdot 10^{-9} \frac{(40 + 220) \cdot 10^{-3}}{1 \cdot 10^{-9}} \approx 2,6V. \quad (11.3)$$

Uvedené napětí  $\Delta V$  je mezi bodem A a zemnicím potenciálem  $GND$ . Tím u samostatného hradla, které má na výstupu úroveň L, dojde k napěťovému posuvu zemnicího potenciálu o  $\Delta V$  vůči  $GND$ , a tím pádem výstup má úroveň H vůči  $GND$ . Přijímací hradlo, kde nenastává napěťový posuv zemnicího potenciálu o  $\Delta V$ , tuto situaci vyhodnotí jako úroveň H a přenesle je na výstup. Tato situace nastává pouze po krátký čas, proto výsledkem je krátký impuls úrovně H. Velikost indukovaného napětí  $\Delta V$  je přímo úměrná změně proudu a velikosti parazitních indukci, nepřímo úměrná strmosti hrany.



Situace na  
desce  
plošného spoje

Obr. 11 – 24 Impuls způsobený indukcí spoje na desce plošného spoje

Tyto úvahy platí jednak pro samotný integrovaný obvod a také pro desku plošného spoje, obr. 11 - 24. V případě desky plošného spoje pracujeme s indukci spojů pro rozvod napájecího napětí. Možnosti změny zemnicího potenciálu mají v praxi následující důsledky:

Na úrovni PLD prvku.

- Větší PLD prvky v pouzdrech o větším počtu pinů mají napájecí vodiče vyvedené na více pinů. Tím se snižuje velikost indukci  $L_{GND}$  a  $L_{CC}$  v napájecích spojích mezi pinem a křemíkem.
- Výstupní budič v IO buňce PLD prvku má programovatelnou vlastnost slew rate. Šikmější hrana zmenšuje velikost indukovaného napětí a zároveň snižuje EMC.

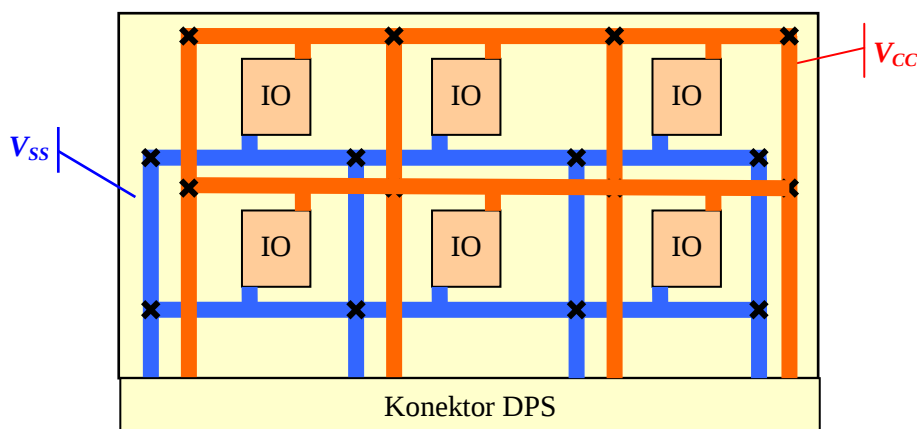
Dopady na  
úrovni PLD  
prvku

Na úrovni desky plošného spoje.

- Rozvody napětí. V případě dvojvrstvé desky se rozvody provádějí širokými spoji až k napájecím pinům integrovaného obvodu. Zároveň, rozvody napětí se organizují do mřížky, obr. 11 - 25. Lepší řešení je použití čtyřvrstevných desek plošných spojů. Napájecí napětí na desku plošného spoje se přivádějí přes více pinů konektoru.

Dopady na  
úrovni desky  
plošného spoje

- Snížením strmosti hran aplikací kondensátorů. Například známe D-konektory se vyrábějí i v provedení, kdy každý pin má kapacitu vůči zemi.
- Omezením výstupních proudů výstupního budiče IO buňky.



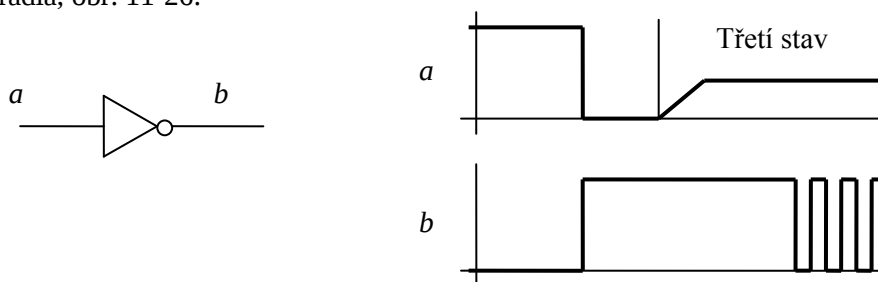
Doporučení pro  
dvojvrstvou  
desku  
plošného spoje

Obr. 11 - 25 Organizace rozvodů napájecího napětí na dvojvrstvé DPS

### 11.16 Oscilace při třetím stavu

Studujme situaci, kdy třístavový vodič není žádným způsobem ošetřen a je uveden do třetího stavu. Pro přijímač to znamená, že na vstupu je hrana s malou strmostí a napěťová úroveň na vstupu přijímačů není definovaná. Tato situace může způsobit po určitém čase oscilaci na výstupu hradla, obr. 11-26.

Oscilace při  
třetím stavu



Obr. 11 - 26 Oscilace při třetím stavu

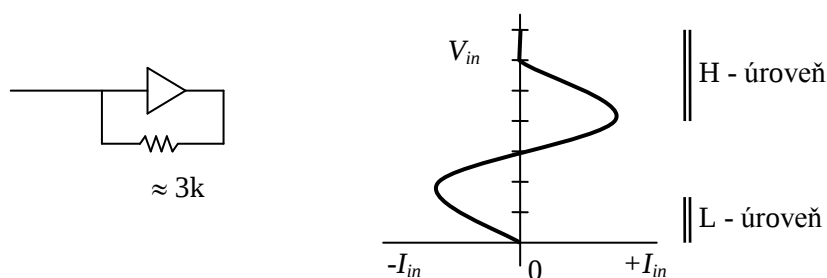
Někdy dokonce může dojít k oscilaci na samém třístavovém vodiči, literatura [Becke, Forstner, Haseloff, Huchzermeier, 1997]. Z pohledu návrhu není vhodné, aby v zapojení docházelo k samovolným oscilacím a návrhář by vlastně o nich nevěděl. Vzniku oscilací, které vznikají na vodiči s třetím stavem, lze zabránit tím, že:

- Vždy musí být jeden zdroj aktivní na vodiči. Situace, že na vodiči je pouze třetí stav, je vyloučena.
- Vodič s třetím stavem je ošetřen pomocí odporu nahoru na  $V_{CC}$  nebo odporovým děličem.
- Aplikovat obvod paměti – bus hold. Dnes mnohé obvody s třístavovým výstupem tento obvod již obsahují. Bus hold je také programovatelná vlastnost výstupního budiče IO buňky PLD prvku.

### 11.17 Bus Hold

Jedná se o speciální zapojení paměti, které je schopno si zapamatovat poslední logickou hodnotu na vodiči. Zapojení a voltampérová charakteristika obvodu je na obr. 11 - 27.

**Bus Hold or  
Week Keeper**



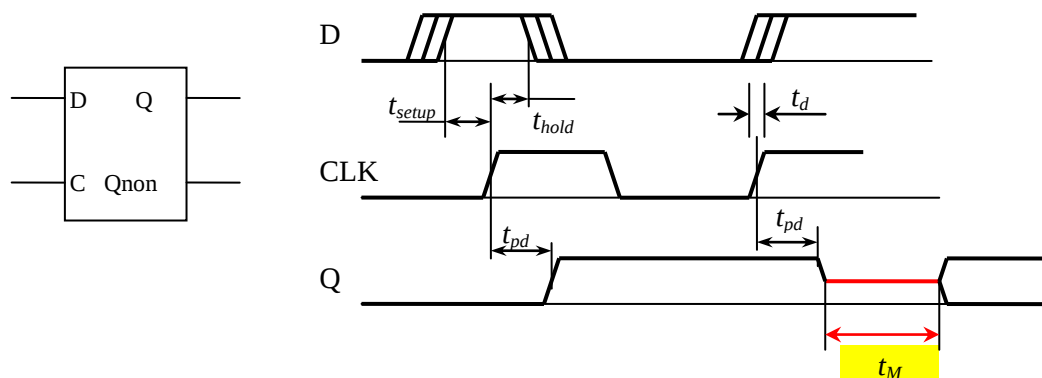
Obr. 11 - 27 Obvod bus hold

V případě třístavového vodiče a uvedení všech vysílačů do třetího stavu, obvod bus hold udržuje třístavový vodič na poslední známe logické úrovni. Zabraňuje přechodu na plávající úroveň, a tím zabraňuje možným oscilacím. V některé literatuře se také můžeme setkat s názvem Week Keeper.

### 11.18 Metastabilita

S metastabilitou se setkáváme u paměťových elementů a je to označení chování, kdy výstup je mimo tolerance L a H úrovně. Tento stav končí po náhodném čase náhodným přechodem do úrovně L nebo H. Tuto situaci si vysvětlíme na D klopném obvodu. Podle katalogu jsou definované časy  $t_{setup}$  a  $t_{hold}$  mezi datovým vstupem a hodinovým signálem, obr. 11 - 28. Čas  $t_{setup}$  označovaný jako předstih určuje poslední možnou změnu datového vstupu před hranou hodinového vstupu. Čas  $t_{hold}$  označovaný jako přesah udává, jak dlouho musí být úroveň datového vstupu konstantní po hraně hodinového signálu. Pokud jsou tyto časy splněny, potom je zajištěno, že výstup přejde do odpovídající úrovně se zpožděním  $t_{pd}$ . V opačném případě výrobce negarantuje změnu výstupu podle chování D klopného obvodu. Časy jsou definovány v katalogu a při jejich splnění výrobce garantuje správnou činnost D klopného obvodu. Bližší informace o metastabilním stavu najdete v literatuře [Latticesemi 2004], [Latticesemi 2003], [Altera 1999], [Becke, Forstner, Haseloff, Huchzermeier, 1997], [Texas 1997b], [Xilinx 1997a] a [Xilinx 1997b].

**Metastabilita**



**Vznik  
metastabilního  
stavu**

Obr. 11 - 28 Vznik metastabilního stavu

Pokud nastává změna vstupu D v časovém intervalu  $t_d$ , ve kterém nejsou splněny časy pro předstih a přesah, může výstup D klopného obvodu přejít do metastabilního stavu, obr. 11 - 28. Na výstupu se objeví nedefinovaná logická úroveň, která trvá náhodný čas  $t_M$ . Tento stav končí

přechodem do definované logické úrovně L nebo H zcela náhodně. Chování D klopného obvodu je zcela náhodné. Tomuto jevu říkáme metastabilní stav. K metastabilnímu stavu dochází, pokud ke změně na D vstupu dochází v časovém intervalu  $t_d$ , kterému říkáme časové okno.

Tato situace běžně nastává pokud se propojují dva nezávislé systémy. Signál D vstupu generuje první systém, který je zcela nezávislý na hodinovém signálu druhého systému. První systém má asynchronní chování vůči druhému systému. Například, D vstup generuje člověk tlačítkem nebo čidlo, které snímá průchod předmětu daným bodem. Tyto akce jsou zcela nezávislé na hodinovém vstupu D klopného obvodu. Jiný příklad je na sběrnících, například komunikace mezi PC a tiskárnou, akce žádost o další data je zcela asynchronní vůči hodinovému signálu, který tuto událost zapisuje jako příznak do registru řadiče v PC. Nebo žádost o přerušení od periferního zařízení je asynchronní vůči hodinovému signálu řadiče přerušení v mikroprocesoru.

Metastabilní jev označujeme jako poruchu a čas mezi jevy označujeme jako střední dobu mezi poruchami – MTBF. Výpočet MTBF je dán vzorcem (11.4), literatura [Becke, Forstner, **Výpočty** Haseloff, Huchzermeier, 1997], [Texas 1997b] a [Altera 1999].

$$MTBF = \frac{1}{f_{in} \cdot f_{clk} \cdot t_d}, \quad (11.4)$$

kde  $f_{in}$  je frekvence na vstupu D,

$f_{clk}$  je frekvence signálu na vstupu CLK,

$t_d$  je časové okno pro vznik metastabilního jevu.

Pokud do vzorce dosadíme hodnoty,  $f_{in} = 5$  MHz,  $f_{clk} = 20$  MHz,  $t_d = 30$  ps potom dochází k metastabilnímu jevu se střední dobou mezi poruchami  $MTBF = 1/(5\text{MHz} \cdot 20\text{Mhz} \cdot 30\text{ps}) = 333 \mu\text{s}$ , což je dost často.

Časová délka trvání metastabilního jevu je náhodná a pohybuje se v rozmezí od 3 ns do 50 ns v závislosti na technologické řadě klopného obvodu (AS, ALS, AHC, LV, ...). Znalost délky metastabilního stavu se dá zohlednit v návrhu, jedná se o vhodné načasování hrany, která vzorkuje výstup klopného obvodu. Proto byl na základě experimentů odvozen vzorec (11.5), který vyjadřuje, jak střední doba mezi poruchami závisí na časové délce metastabilního stavu  $t_M$ .

$$MTBF = \frac{\exp(T \cdot t_M)}{f_{in} \cdot f_{clk} \cdot T_O}, \quad (11.5)$$

kde  $MTBF$  je střední doba mezi metastabilními jevy (v sekundách) delšími než  $t_M$ ,

$t_M$  je časová délka metastabilního jevu v nanosekundách,

$f_{in}$  je frekvence signálu datového vstupu klopného obvodu,

$f_{clk}$  je frekvence hodinového signálu synchronního vstupu klopného obvodu,

$T$  a  $T_O$  jsou experimentálně odvozené konstanty pro jednotlivé technologické řady (S, AS, F, HC, ...).

Pro následující zapojení, obr. 11 - 29 je možné použít vzorec (11.5), kde datové vstupy klopných obvodů FF1 až FF $n$  jsou časově nezávislé, nejsou synchronizovány s hodinovým vstupem CLK klopného obvodu. Návrh musí zabezpečit, aby na vstupu D klopného obvodu FF11 byla správná úroveň po odeznění metastabilního stavu, rovnice (11.6).

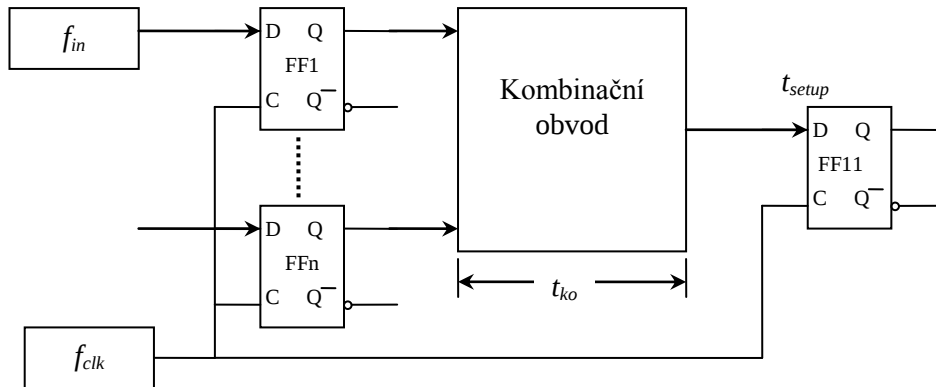
$$\frac{10^9}{f_{clk}} = t_M + t_{ko} + t_{setup}, \quad (11.6)$$

kde  $f_{clk}$  je frekvence hodinového signálu,

$t_M$  je délka metastabilního stavu v nanosekundách,

$t_{ko}$  je zpoždění kombinačního obvodu v nanosekundách,

$t_{setup}$  je čas předstihu v nanosekundách pro klopný obvod FF11.



Obr. 11 - 29 Zapojení s metastabilním stavem

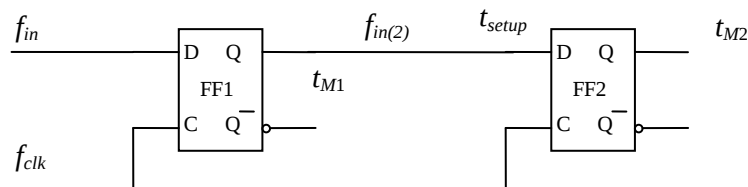
Výpočet pro  
uvedené  
zapojení

Zabývejme se konkrétním použitím D klopných obvodů 74S74 a 74AS74 v obr. 11 - 29. Obvod 74S74 má  $t_{setup} = 3 \text{ ns}$ , a  $T = 0,36 \text{ ns}^{-1}$  a  $T_O = 1,3 \times 10^{-9} \text{ s}$ ; obvod 74AS74 má  $t_{setup} = 4,5 \text{ ns}$  a  $T = 4 \text{ ns}^{-1}$  a  $T_O = 1,4 \times 10^{-9} \text{ s}$ , literatura [Texas 1997b]. Předpokládejme zpoždění kombinační sítě  $t_{ko} = 10 \text{ ns}$ , frekvenci hodinového signálu  $f_{clk} = 20 \text{ MHz}$  a frekvenci datového vstupu  $f_{in} = 5 \text{ MHz}$ . Po dosazení konkrétních hodnot do vzorce (11.6) dostáváme maximální délku metastabilního stavu na výstupu D klopných obvodů FF1 až FFn, který nenaruší správnou činnost obvodu, pro obvod 74S74  $t_M = 37 \text{ ns}$  a pro obvod 74AS74  $t_M = 35,5 \text{ ns}$ . Překročení těchto časů způsobí na vstupu D klopného obvodu FF11 nesprávnou úroveň a střední doba mezi těmito poruchami se vypočte podle vzorce (11.5), pro obvod 74S74 je  $MTBF \approx 4,7 \text{ sekund}$  a pro obvod 74AS74 je  $MTBF \approx 10^{37} \text{ roků}$ . Správná činnost obvodu bude narušena metastabilním stavem D klopných obvodů FF1 až FFn každých 4,7 s při použití obvodů 74S74 a  $10^{37} \text{ roků}$  při použití obvodu 74AS74. Správný výběr technologické rodiny může eliminovat chyby obvodu způsobené metastabilním stavem. Toto je první způsob, jak eliminovat metastabilní jev v daném zapojení.

Eliminace  
metastabilního  
stavu

Výběr rodiny S74  
nebo AS74

$MTBF \approx 10^{37} \text{ roků}$   
pro 74AS74



Obr. 11 - 30 Obvod s dvoustavovou synchronizací

Zapojení  
eliminující  
metastabilní stav  
Dvoustavová  
synchronizace

Jiný způsob eliminace metastabilního stavu je aplikace obvodu s dvoustavovou synchronizací podle obr. 11 - 30. V tomto případě je střední doba mezi metastabilním jevem na výstupu druhého klopného obvodu dána vzorcem (11.7) a (11.8).

$$f_{in(2)} = \frac{1}{MTBF(1)} = \frac{f_{in} \cdot f_{clk} \cdot T_{O(1)}}{\exp\left(T_{(1)} \left(\frac{10^9}{f_{clk}} - t_{setup}\right)\right)} \quad (11.7)$$

$$MTBF(2) = \frac{\exp(T_{(2)} \cdot t_{M2})}{f_{in(2)} \cdot f_{clk} \cdot T_{O(2)}} \quad (11.8)$$

Vztah (11.7) vyjadřuje frekvenci metastabilních jevů na výstupu prvního D klopného obvodu FF1, které jsou delší než perioda hodinového signálu minus předstih druhého klopného obvodu,  $t_{M1} = (10^9/f_{clk}) - t_{setup}$ . Kratší metastabilní jevy než  $t_{M1}$  nemají vliv na činnost obvodu. Vztah (11.8) vyjadřuje střední dobu metastabilního jevu na výstupu druhého D klopného obvodu FF2.

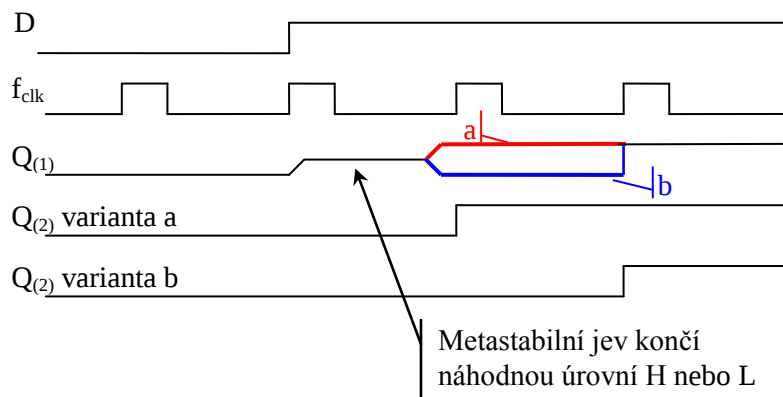
Pokud v obr. 11 - 29 nahradíme D klopné obvody FF1 až FF<sub>n</sub> principem dvoustavové synchronizace podle obr. 11 - 30 a použijeme všechny D klopné obvody 74S74 s předstihem  $t_{setup} = 3$  ns, a konstantami  $T = 0,36$  ns<sup>-1</sup> a  $T_O = 1,3 \times 10^{-9}$  s, délka metastabilního stavu  $t_{M2}$  druhého obvodu je určena vztahem (11.6). Frekvence metastabilního jevu na vstupu druhého D klopného obvodu je 0,0058 Hz, vztah (11.9) a střední doba metastabilního jevu delšího než  $t_{M2} = 37$  ns na výstupu druhého D klopného obvodu je asi 127 roků, vztah (11.10). Toto je druhý způsob zabránění metastabilních jevů. Pro obvody 74AS74 je střední doba metastabilního jevu, který je delší než  $t_{M2} = 35,5$  ns na výstupu druhého klopného obvodu, je asi  $4 \times 10^{105}$  roků.

**MTBF ≈ 127 roků  
pro 74S74**

$$f_{in(2)} = \frac{1}{MTBF(1)} = \frac{5 \text{ MHz} \cdot 20 \text{ MHz} \cdot 1,3 \cdot 10^{-9}}{\exp\left(0,36 \text{ ns}^{-1} \cdot \left(\frac{10^9}{20 \text{ MHz}} - 3 \text{ ns}\right)\right)} \approx 0,0058 \text{ Hz} , \quad (11.9)$$

$$MTBF(2) = \frac{\exp(0,36 \text{ ns}^{-1} \cdot 37 \text{ ns})}{172 \text{ Hz} \cdot 20 \text{ MHz} \cdot 1,3 \cdot 10^{-9} \text{ s}} \approx 127 \text{ roků} . \quad (11.10)$$

Na obr. 11 - 31 jsou časové průběhy situace dvoustavové synchronizace pro eliminaci metastabilního jevu podle obr. 11 - 30. Na obrázku je situace, kdy metastabilní jev nastane na výstupu prvního D klopného obvodu a je kratší než perioda vzorkovacích hodin. Metastabilní jev na výstupu prvního D klopného obvodu  $Q_{(1)}$  končí náhodnou úrovní H nebo L, varianta a nebo b. Výstup  $Q_{(2)}$  je v pořádku, pouze dojde ke zpoždění vzorkovaného signálu o jednu nebo dvě periody vzorkovacího signálu.



**Časové  
průběhy pro  
dvoustavovou  
synchronizaci**

Obr. 11 - 31 Princip dvoustavové synchronizace

Problém s metastabilním stavem se odstraňuje způsoby:

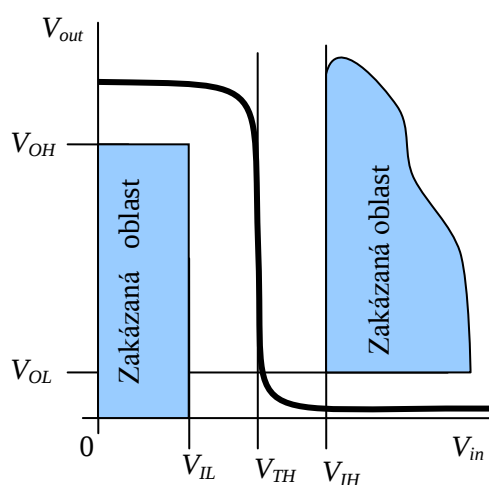
- Výběrem vhodné technologické řady pro klopný obvod.
- Pomocí dvoustavové synchronizace, obr. 11 – 30 a obr. 11 - 31.

**Způsoby  
odstranění  
metastability**

### 11.19 Překlápěcí napětí $V_{TH}$ – Threshold Level

Logická hradla pracují s L a H úrovněmi, kterým odpovídá určité rozmezí napětí. Pokud si všimneme funkce obyčejného negačního hradla s pohledu napětí, potom převodová charakteristika, která udává vztah výstupního napětí jako funkci vstupního napětí, je na obr. 11 – 32. Z obrázku je jasné vidět, že výstupní napětí rychle mění svou hodnotu okolo hodnoty vstupního napětí  $V_{TH}$ . Pokud je vstupní napětí větší než  $V_{TH}$ , potom se chápe jako H úroveň a obráceně, pokud je vstupní napětí menší než  $V_{TH}$ , potom se chápe jako L úroveň. Překlápěcí napětí  $V_{TH}$  nelze použít pro definici L a H úrovní, protože jeho hodnota je typická pro každou logiku, nikoliv přesně daná.

**Překlápěcí  
napětí  
Threshold  
Level**



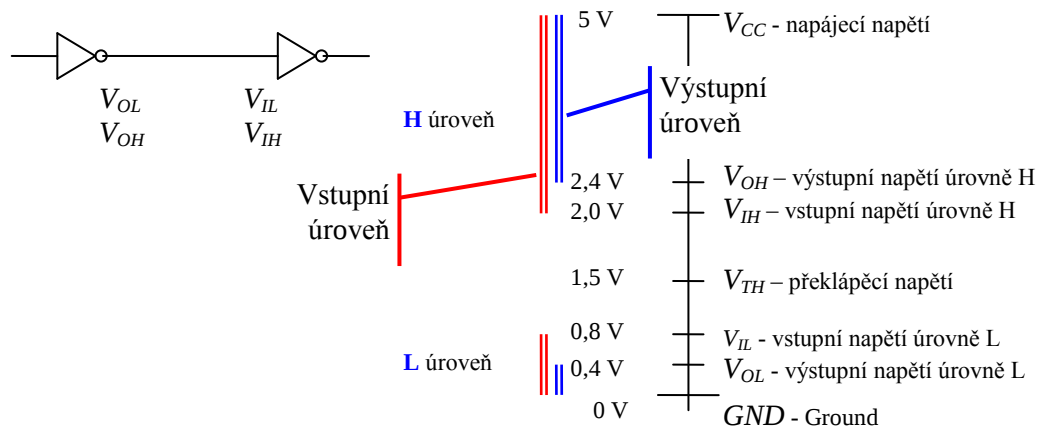
Obr. 11 – 32 Převodová charakteristika invertoru

Nehledě k tomu, že vstupní napětí pohybující se okolo překlápěcího napětí  $V_{TH}$  způsobuje za určitých podmínek i oscilace na výstupu hradla. Nezapojený vstup hradla většinou mívá hodnotu napětí o málo vyšší, než je hodnota překlápěcího napětí  $V_{TH}$ , proto někdy nezapojené vstupy neovlivňují činnost hradla, např. AND. Z pohledu zdroje napětí, je tento zdroj velmi měkký zdroj, a tím pádem velmi náchylný na rušení z okolí. Proto se nezapojené vstupy musí vždy ošetřovat definovanou L nebo H úrovní z tvrdého zdroje.

### 11.20 Definice L a H úrovně

TTL logika vznikla již v 60. letech 20. století a byla realizována pomocí bipolární technologie. Postupem času se k realizaci začaly používat nově vyvinuté technologie jako je CMOS, BiCMOS a také přechod na snížené napájecí napětí, od kterého se odvozuje název logiky - 3,3 V logika, dnes 1,8 V logika. Definici L a H úrovní je také přesně vidět na přechodové charakteristice invertoru, obr. 11 – 32. Původní TTL logika byla 5 V, a tím i prvně definované napěťové rozsahy pro L a H úroveň, obr. 11 - 33.

**Definice L a  
H úrovně**



**Definice pro  
TTL 5V  
logiku**

Obr. 11 - 33 Definice L a H úrovně v 5 V TTL logice

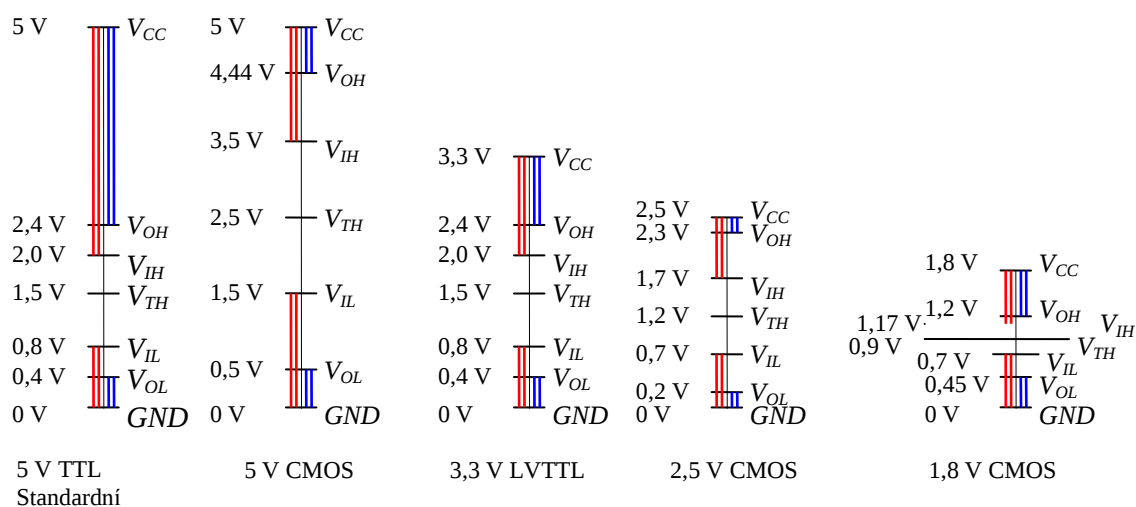
Hraniční hodnoty napětí pro L a H úroveň jsou pro vstup a výstup různé, kde výstup má přísnější definici. H úrovní odpovídá napětí, které je větší než  $V_{IH} = 2,0\text{ V}$  na vstupu a větší než  $V_{OH} = 2,4\text{ V}$  na výstupu hradla. Z druhé strany je definice ohraničena napájecím napětím  $V_{CC} = 5\text{ V}$ .

L úrovní odpovídá napětí, které je menší než  $V_{IL} = 0,8\text{ V}$  na vstupu a menší než  $V_{OL} = 0,4\text{ V}$  na výstupu hradla. Z druhé strany je definice ohraničena napětím  $GND = 0\text{ V}$ . Další významné napětí je překlápěcí napětí  $V_{TH}$  – threshold level. Toto napětí rozděluje vstupní rozsah napětí na dvě oblasti, kdy je vstupní napětí chápáno jako úroveň L a kdy jako úroveň H.

Rozdíl napětí mezi vstupní a výstupní definicí pro danou úroveň se říká šumová imunita a je rovna  $0,4\text{ V}$ . V tomto rozdílu je náhodné napětí, které může být do spojovacího vodiče indukováno z okolí a stejnosměrné napětí vytvořené na odporu vodiče či pomocném sériovém odporu, vloženém do série s vodičem. **Šumová imunita**

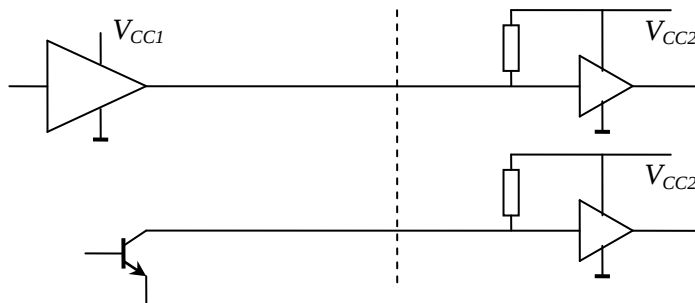
Rozvojem technologie vznikly nové definice L a H úrovní, které jsou zobrazeny na obr. 11 - 34. Na obrázku jsou uvedeny definice význačných logik. Dále uvedu možné názvy logik, se kterými se můžeme setkat u PLD prvků. Jedná se o 3,3V PCI, 3,3V LVTTL/LVCMOS, 2,5V LVTTL/CMOS, 1,8V LVTTL/LVCMOS, 1,5V LVTTL/LVCMOS a diferenciální logika LVDS, LVPECL, 1,8V HSTTL, 1,5V HSTTL, SSTTL. Úplná specifikace všech logik je nad rámec těchto skript a další bližší informace lze nalézt v literatuře [Texas 2003a] a [Texas 2004].

**Další logiky**



Obr. 11 - 34 Definice L a H úrovní u ostatních logik

Pokud je číslicový systém realizovaný pouze v jedné logice není žádný problém. Pokud se v návrhu objeví i jiné logiky, je nutno zajistit přizpůsobení jednotlivých logik. V některých případech propojení různých logik je možné přímé propojení jindy je nutnost použít převodníku úrovní. Pro logiku s různým napájecím napětím je možné použít jako převodníky hradla s otevřeným kolektorem, obr. 11 - 35 nebo pouze ošetření standardního výstupu pomocí odporů. Vše závisí na použité logice. Blíže literatura [Texas 1996c], [Texas 2003] a [Texas 2004].



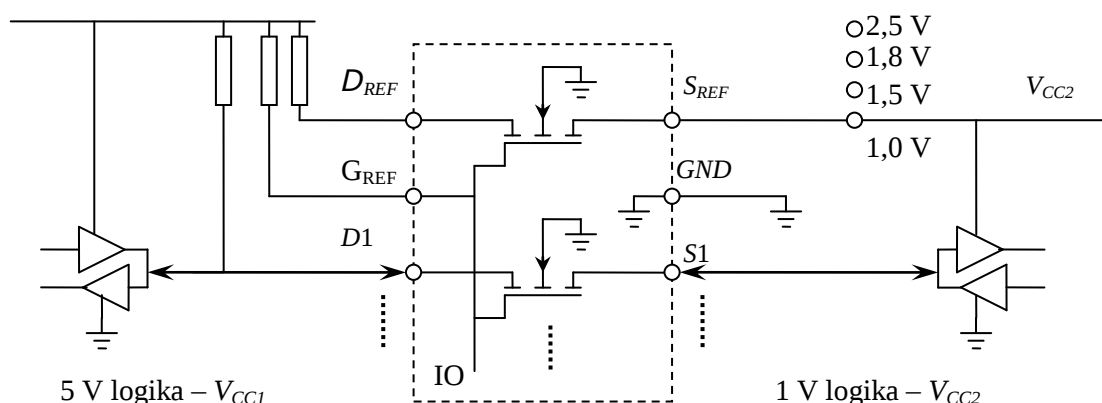
**Propojování  
dvou logik**

Logika s napájecím napětím  $V_{CC1}$

Logika s napájecím napětím  $V_{CC2}$

Obr. 11 - 35 Možný převodník úrovní H a L pro různé logiky s  $V_{CC1}$  a  $V_{CC2}$

Dnes se hlavně ve sběrnicevých systémech, jako převodník úrovní H a L pro různé napájecí napětí, uplatňují modifikace sběrnicevých přepínačů – bus switch, literatura [Philips 2002] a [Texas 1999]. Příklad využití sběrnicevého přepínače jako převodníku je na obr. 11 - 36.



**Propojování  
dvou logik**

Obr. 11 - 36 Aplikace obvodů bus switch jako převodník úrovní mezi logikami

## 11.21 Diferenciální logika LVDS

Informace při diferenciálním signálu je dána rozdílem napětí a jeho orientací mezi dvěma signály, vodiči. Proto diferenciální signály vytvářejí pár a pro přenos je nutné použít dva signální vodiče, plus a mínus. Základní výhodou diferenciálních signálů je nezávislost na napájecím napětí zdroje, nízké vyzařování (EMI) a vysoká odolnost proti vnějšímu rušení. Pojem diferenciální signál je ekvivalentní symetrickému signálu. Dnes se uplatňuje hlavně diferenciální logika LVDS – Low Voltage Differential Signaling.

**Diferenciální  
logika LVDS**

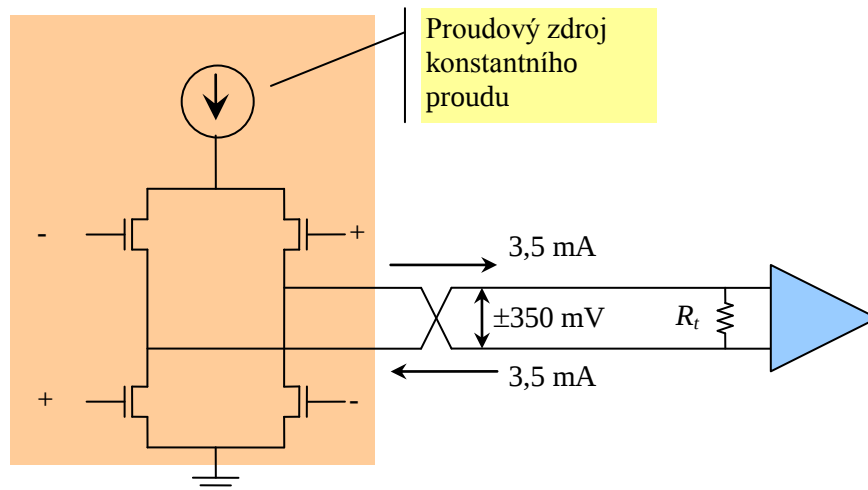
**neboli  
symetrický  
signál**

LVDS je diferenciální logika určená pro aplikace v komunikacích, serverech, perifériích a počítačích, kde se vyžaduje rychlý přenos dat. Při porovnání LVDS logiky se staršími standardy diferenciální logiky, tak LVDS poskytuje levné, nízko příkonové řešení při vysoké přenosové kapacitě. Detailnější popis a použití je v literatuře [Fairchild 2000] a [Texas 2003b].

LVDS je definována standardem TIA/EIA-644. Přenosová kapacita je standardem definována až 655 Mbit/s, dnes některá firemní řešení tuto hranici zvyšují. LVDS je založena na

konstantním proudovým zdroji, který vytváří na ukončovacím odporu plovoucí diferenci, obr. 11 - 37.

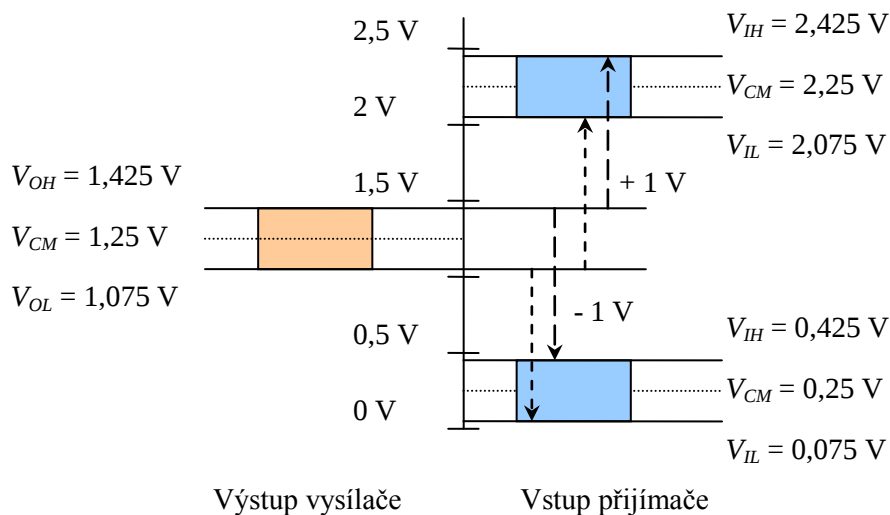
**Diferenciální signál** poskytuje hodně výhod než nesymetrické technologie. LVDS signál na straně vysílače má diferenci  $350\text{ mV}$  se středem okolo  $V_{CM} = 1,25\text{ V}$  a je nezávislý na napájecím napětí, obr. 11 – 38. Tato definice signálu umožňuje jednoduchý přechod i pro aplikaci zdrojů s nízkým napětím. **LVDS logika**



Zdroj: Fairchild Semiconductor; AN-5017

Obr. 11 - 37 Základní schéma vysílače

Diferenciální signál má plovoucí **společný bod**. Proto přijímač ignoruje jakékoliv rušení, které se indukuje do vodičů. Přijímač pouze vyhodnocuje rozdíl napětí mezi vodiči, obr. 11 - 38. Přijímač má společný plovoucí bod  $V_{CM}$  v rozmezí napětí  $0,25\text{ V}$  až  $2,25\text{ V}$ . Přijímač pracuje s posuvem společného bodu  $V_{CM}$  o  $\pm 1\text{ V}$  vůči vysílači. Definice nízkého rozdílu signálu vylepšuje integritu signálu pro vysoké rychlosti přenosu.



**Definice úrovní LVDS logiky**

Zdroj: Fairchild Semiconductor; AN-5017

Obr. 11 - 38 Šumový rozsah společného bodu

LVDS logika má **ochranu proti poruchám**, která zvyšuje spolehlivost přenosu a chrání ho proti poruchám. Ochrana proti poruchám zaručuje, že výstup přijímače bude na úrovni HIGH, **LVDS logika a ochrana proti poruchám**

když vstup přijímače je mimo definované podmínky. Bez této ochrany by jakýkoliv externí šum na vstupu přijímače mohl způsobit neznáme stavy na výstupu.

Výstup přijímače přechází do stavu HIGH v případech, kdy vstup přijímače je nezapojen nebo zkratován, není připojen zdroj signálu nebo zdroj je vypnutý. Výstup přijímače také přechází do stavu HIGH, když diferenciální signál zůstane v definovaném rozpětí po abnormální periodu času.

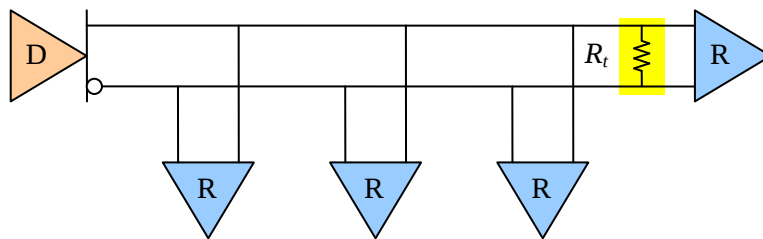
Tato ochrana má hodně výhod pro návrháře. Například v některých aplikacích může dojít k situaci, kdy všechny LVDS přijímače nebudou použity. Ochrana proti poruchám zaručuje, že výstupy nezapojených přijímačů budou trvale ve stavu HIGH.

Standard definuje způsob **ukončení** pomocí ukončovacího odporu v rozmezí hodnot  $90\ \Omega$  až  $132\ \Omega$ . V praxi se většinou používá hodnota  $100\ \Omega$ . Tento odpor zajišťuje jednak odporové přizpůsobení vedení a také generuje rozdílové napětí pro vstup přijímače.

Ukončovací  
odpor LVDS  
logiky

LVDS pro přenos používá dva vodiče jako zkroucený pár nebo spoje na desce plošného spoje. Charakteristická impedance uvedených vedení je okolo  $100\ \Omega$ . LVDS logika z pohledu přenosu informace se používá v konfiguracích:

- Point-to-Point, jedná se o propojení bod – bod, kdy je jeden vysílač a přijímač, obr. 11 - 39. Ukončovací odpor je na vstupu přijímače.

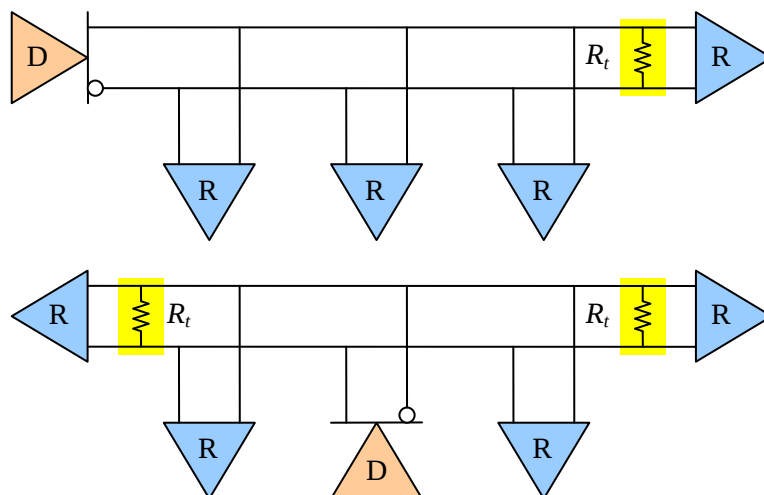


Aplikace  
LVDS logiky

Zdroj: Fairchild Semiconductor; AN-5017

Obr. 11 - 39 Zapojení bod-bod

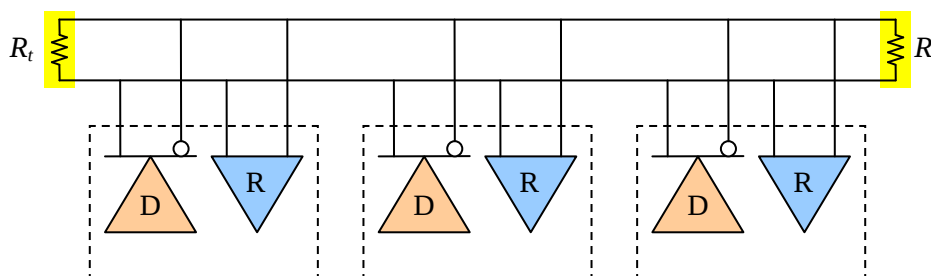
- Multi-Drop Configuration, jedná se o zapojení s jedním vysílačem a více přijímačů, obr. 11 - 40. Vysílač z pohledu vedení může být umístěn na jednom z jeho konců nebo uprostřed vedení. Pokud je vysílač na jednom konci vedení, potom ukončovací odpor je na jeho druhém konci, nejvzdálenějším přijímači. V případě umístění vysílače do středu vedení musí být ukončovací odpory na obou jeho koncích. Z pohledu vysílače se jedná o buzení paralelní kombinace obou odporů, při hodnotě ukončovacích odporů  $100\ \Omega$  musí vysílač budit vedení dvojnásobným proudem.



Aplikace  
LVDS logiky

Zdroj: Fairchild Semiconductor; AN-5017

Obr. 11 - 40 Zapojení s více přijímači



Aplikace  
LVDS logiky

Zdroj: Fairchild Semiconductor; AN-5017

Obr. 11 - 41 Více bodové zapojení – sběrnice

Multi-point Configuration, jedná se o sběrnicové zapojení, kdy na vedení je více vysílačů a přijímačů, obr. 11 - 41. Ukončovací odpory jsou na obou koncích sběrnice a proto z pohledu vysílače se jedná o paralelní zapojení ukončovacích odporů. Pro hodnotu ukončovacích odporů  $100\ \Omega$  musí vysílač budit sběrnici dvojnásobným proudem, to je  $7\ \text{mA}$ .

## 11.22 Práce pod napětím – Live Insertion, Hot Swapping

V dnešní době se dost často vyžaduje vyjmutí či zasunutí modulu do systému bez jeho vypnutí. Jevy, které mohou při této činnosti vzniknout, mohou vést až k zničení součástky. Proto vstupy a výstupy jsou vybavované obvody, které zamezují jejich zničení při vyjmutí či zasunutí pod napětím. Význam této vlastnosti je ve vylepšení servisu, zvýšení spolehlivosti. Samozřejmě tyto vylepšení se musí odrazit i v samotném návrhu modulu a také v jeho ceně. Při návrhu obvodů a modulů umožňující práci pod napětím se setkáváme s problémy, které jsou rozděleny do tří úrovní. Blíže literatura [Texas 1996b], [Texas 2003a], [Magdaluyo 2000], [Philips 1995]a [IDT 1996].

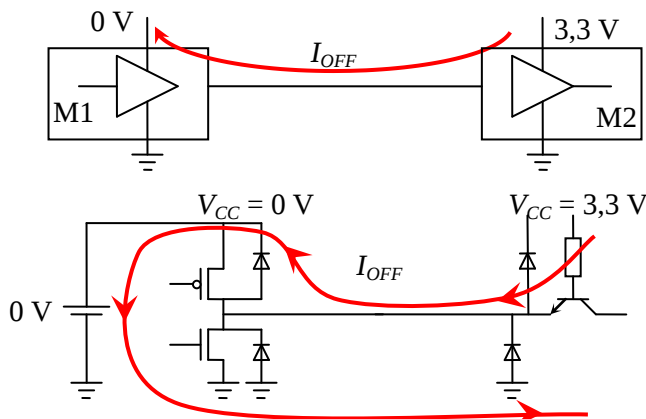
Práce pod  
napětím

**Úroveň 1 – vyrovnávací proud  $I_{OFF}$ .** Uvědomme si, že na vstupech hradla mohou být ochranné diody proti překmitům a na výstupu hradla jsou parazitní diody dané realizací hradla. Dále, jsou propojena dvě hradla jako součást dvou modulů a jeden modul je bez napětí a druhý s napětím, obr. 11 - 42.

Práce pod  
napětím –  
úroveň 1

V modulu bez napětí lze považovat rozvod  $V_{CC}$  za zkratovaný s  $GND$ , a tím vzniká možnost proudu tekoucího z modulu s napájením přes signální vodič do modulu bez napájení. Podle

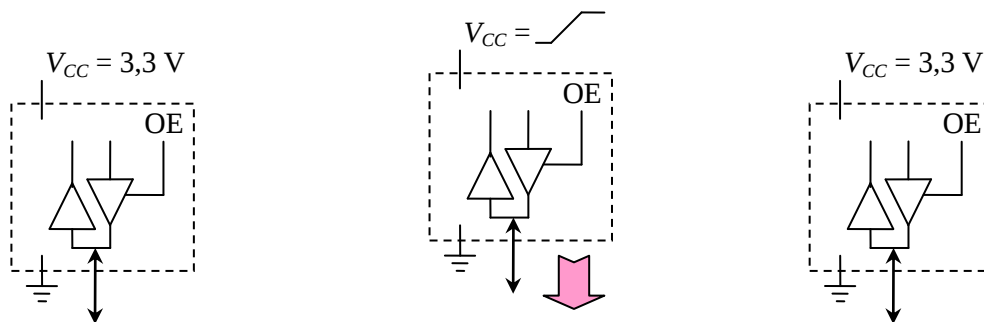
velikosti proudu a směru může dojít ke zničení obvodů. Velikost a směr proudu závisí na konkrétním zapojení vstupů a výstupů hradel včetně ochranných prvků. Proto vstupy a výstupy musí být řešeny tak, aby nedošlo k jejich zničení. Tento požadavek se promítá do zapojení integrovaného obvodu a v katalogu jej najdeme vyjádřený pomocí proudu  $I_{OFF}$  při  $V_{CC} = 0\text{ V}$ ,  $I_{OFF}$  při  $0\text{ V} < V_I$  nebo  $I_{OFF}$  při  $V_O < V_{CC}$ .



Obr. 11 - 42 Práce pod napětím, úroveň 1

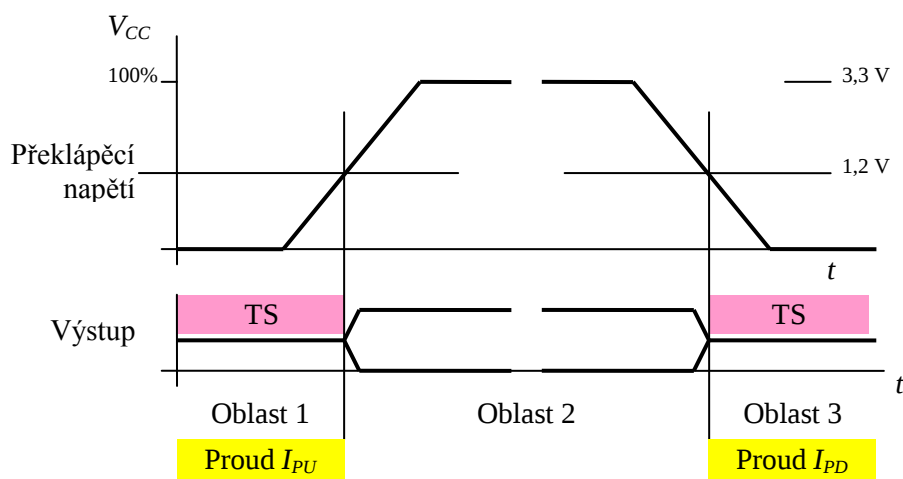
**Úroveň 2 – akce při náběhu napětí, PU3S.** Při zasouvání a vyjímání modulu ze systému pod napětím dochází u integrovaných obvodů k akci připojení napětí (power up) nebo odpojení napětí (power down), obr. 11 - 43. Práce pod napětím vyžaduje, aby po dobu náběhu a odpojení napětí byl definován stav a elektrické parametry třístavového výstupu, aby nedošlo k nežádoucímu ovlivnění zbylé funkce systému.

Práce pod  
napětím –  
úroveň 2



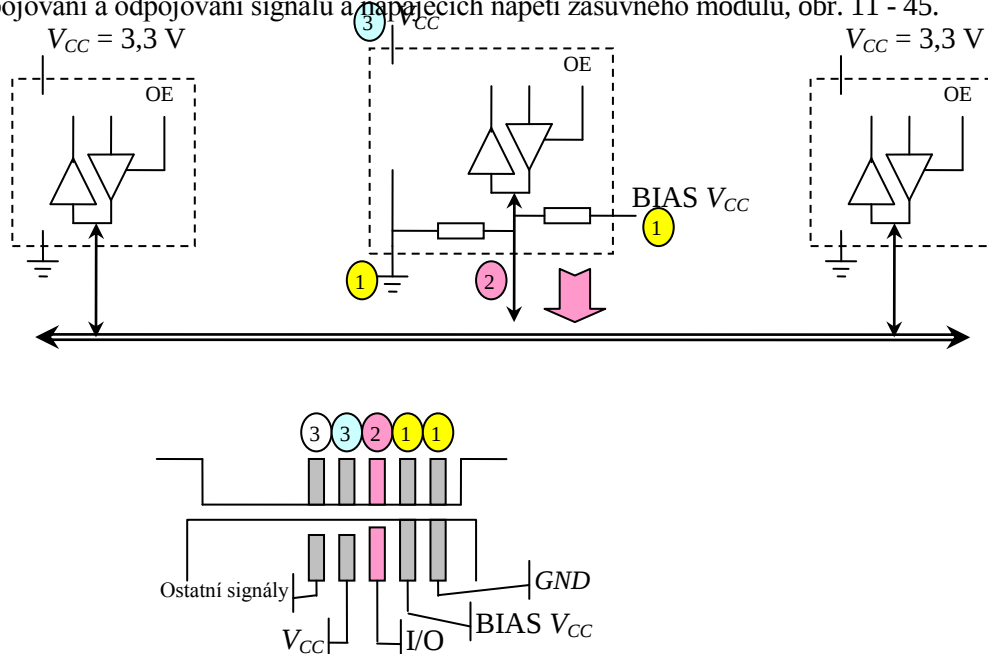
Obr. 11 - 43 Princip PU3S, úroveň 2

Vlastnosti třístavového výstupu po dobu náběhu a výpadku napětí jsou na obr. 11 - 44. V oblasti 1, kdy napájecí napětí stoupá a je menší než překlápěcí napětí, musí být výstup v třetím stavu. Tento stav zjišťuje samotný budič svým zapojením. Tato situace se v katalogu popisuje proudem  $I_{UP}$ . V oblasti 2 je výstup již řízen povoločím signálem třetího stavu. Pokud má být výstup ještě v třetím stavu musí tuto situaci zajistit návrhář aktivním signálem povolení třetího stavu. Oblast 3 má stejné vlastnosti jako oblast 1, pokud napájecí napětí klesne pod překlápěcí napětí, musí budič sám zajistit, aby výstup byl v třetím stavu. Tato situace se v katalogu popisuje proudem  $I_{PD}$ .



Obr. 11 - 44 Třetí stav při náběhu a vypnutí napětí

**Úroveň 3 – přizpůsobovací odpory, konektor.** Při zasouvání a vysouvání modulu, který je připojen na sběrnici systému, se mění přenosové vlastnosti vedení a na sběrnicových vodičích dochází ke krátkodobým pulsům opačné polaroty než je signál - glitches. Vzniku těchto krátkodobých impulsů se zabráňuje jednak samostatným napájením přizpůsobovacích odporů sběrnice a sekvencí připojování napájecího napětí, nejdřív napájení přizpůsobovacích odporů a potom napájecí napětí jádra modulu. Tento princip připojování napájecího napětí vyžaduje speciální přímý konektor, kdy jednotlivé piny mají různou délku, která zajišťuje vhodné pořadí připojování a odpojování signálů a napájecích napětí zásuvného modulu, obr. 11 - 45.



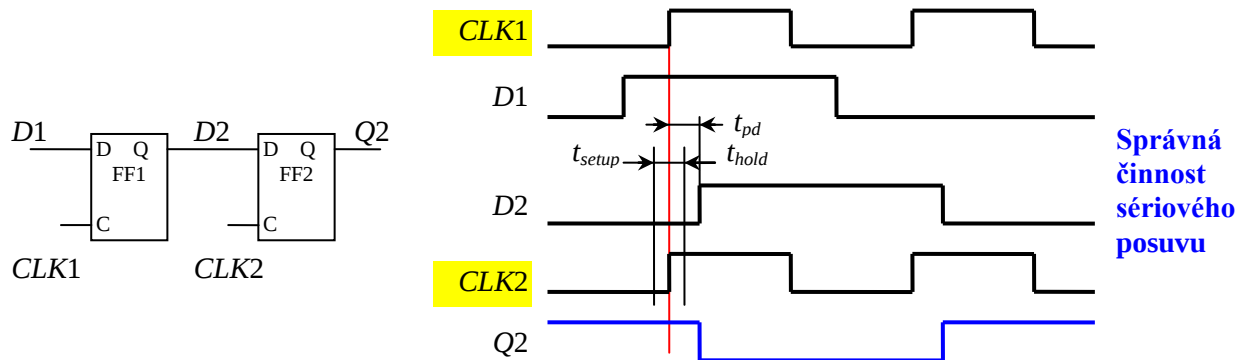
Obr. 11 - 45 Zapojení konektoru

### 11.23 Rozvody hodin

Hodinový signál zajišťuje synchronizaci celého zapojení. Je proto důležité, aby všechny hodinové signály na vstupu paměťového elementu byly ve stejný okamžik a měly nulový časový rozdíl – skew time. Nesplnění této podmínky se nejvíce projeví na posuvném registru, jehož funkčnost je založena na zpoždění paměťového elementu a existence předstihu a přesahu

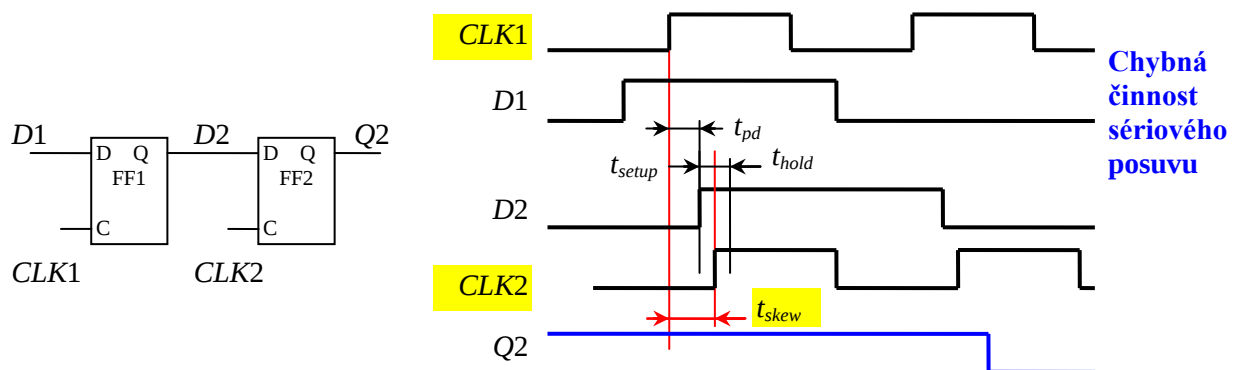
Rozvody hodin

na datovém vstupu, obr. 11 - 46. Důležitá podmínka činnosti je, že hrany na hodinových vstupech D klopných obvodů mají nulový skew time.



Obr. 11 - 46 Správná činnost posuvného registru

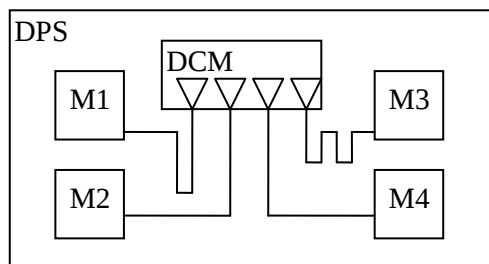
Uvedené zapojení pracuje správně i při malém posuvu hran na hodinových vstupech. Akceptovatelná hodnota skew time na hodinových vstupech je dána skutečným zpožděním D klopného obvodu  $t_{pd}$  a požadovanými hodnotami předstihu  $t_{setup}$  a přesahu  $t_{hold}$  na datovém vstupu. Pro správnou činnost je potřebná hodnota skew time nikoliv zpoždění hodinového signálu od zdroje ke vstupům.



Obr. 11 - 47 Časový posuv hodin CLK2

Na obr. 11 - 47 je situace, kdy není splněna podmínka akceptovatelné hodnoty skew time. Hodnoty předstihu a přesahu jsou posunuty v čase tak, že druhý D klopný obvod vzorkuje již novou hodnotu výstupu předcházejícího odvodu. Dochází ke ztrátě informace a nesprávné činnosti posuvného registru. Skew time na hodinových vstupech vzniká nesprávným rozvodem hodin, spoj k druhému hodinovému vstupu má velké zpoždění.

Výrobci PLD prvku věnují velkou pozornost rozvodu hodinového signálu. Tyto rozsáhlé sítě jsou navrženy tak, aby vykazovaly definovaný skew time mezi všemi hodinovými vstupy paměťových elementů



**Doporučení  
pro rozvod  
hodin**

**nulový skew  
time**

Obr. 11 - 48 Vhodné varianty rozvodu hodinového signálu

I na desce plošného spoje je nutné věnovat pozornost rozvodu hodinového signálu a snažit se dosáhnout minimálního skew time mezi všemi hodinovými vstupy integrovaných obvodů. Hodinové signály by měly mít stejnou délku od zdroje ke všem hodinovým vstupům, obr. 11 - 48. Je výhodné, aby každá větev měla svůj samotný zdroj. Z teorie vedení je nevhodné větvit hodinový signál, protože v místě větvení dochází k odporovému nepřizpůsobení se svými důsledky.

## Otázky kap. 11



- 11.1 Popište význam bezpečnostní pojistky, nulování po náběhu napětí a elektronického podpisu.
- 11.2 Popište princip a význam třístavového výstupu.
- 11.3 Popište použitelné varianty při přenosu pomocí vedení.
- 11.4 Popište význam omezujících diod.
- 11.5 Popište význam a varianty odporového přizpůsobení vedení.
- 11.6 Popište sběrníkový vodič pomocí otevřeného kolektoru.
- 11.7 Popište sběrníkový vodič pomocí třístavového výstupu.
- 11.8 Vysvětlete časový rozdíl signálů, skew time.
- 11.9 Vysvětlete strmost hran, slew rate a důsledek malé strmosti.
- 11.10 Vysvětlete směry proudů na signálním vodiči pro úroveň L a H a časový průběh napájecího proudu při změně úrovně výstupu.
- 11.11 Vysvětlete vliv rozvodů napětí na generování falešných impulsů.
- 11.12 Způsob odstranění oscilací výstupu způsobených třetím stavem na vstupu.
- 11.13 Vysvětlete pojem metastabilita.
- 11.14 Jaké jsou způsoby odstranění metastabilního jevu?
- 11.15 Vysvětlete pojem překlápěcí napětí, definici L a H úrovně pro 5 V TTL logiku. Vyjmenujte další možné logiky.
- 11.16 Popište definici a význam diferenciální logiky LVDS.
- 11.17 Popište možné zapojení diferenciální logiky LVDS při přenosech.
- 11.18 Co je to práce pod napětím?
- 11.19 Popište problém práce pod napětím podle jednotlivých úrovní.

11.20 Význam rozvodu hodin, důsledky časových posuvů hodinových signálů.

11.21 Možné způsoby eliminace časových posuvů v hodinových rozvodech.

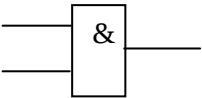
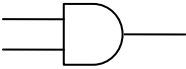
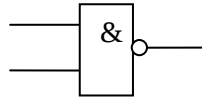
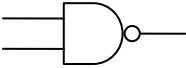
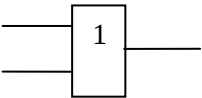
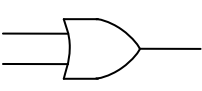
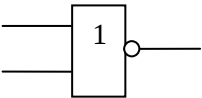
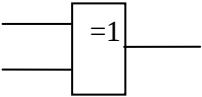
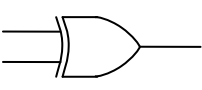
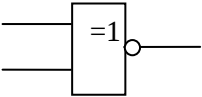
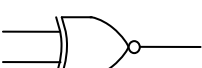
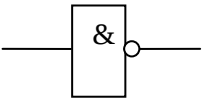
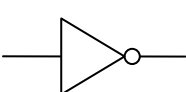
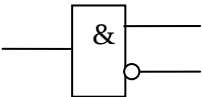
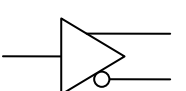
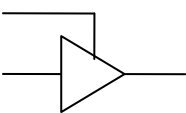
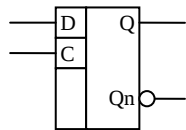
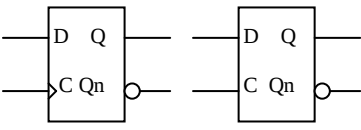
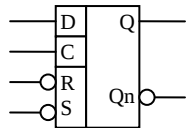
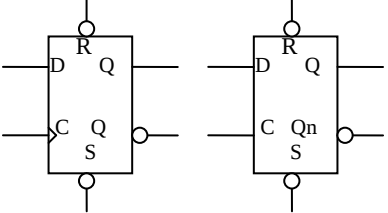
## Odměna a odpočinek

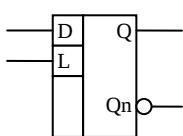
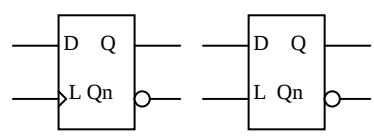
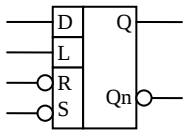
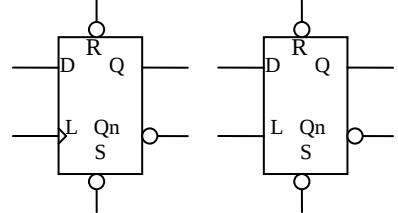
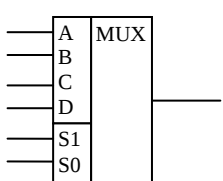
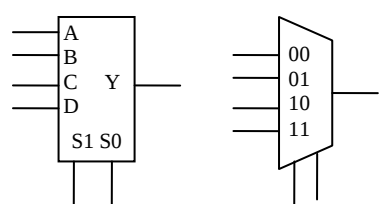
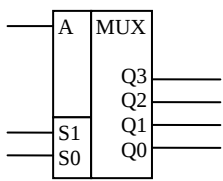
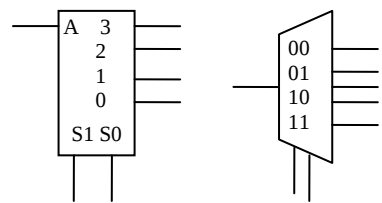
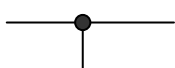
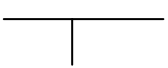
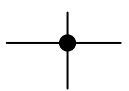
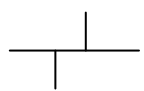


V dnešní době se PLD obvody rozdělují na tři skupiny podle hustoty integrace. Jedná se o:

- Standardní PLD. Jedná se o obvody hlavně řady GAL, které se používají dodnes. Typickým představitelem je obvod GAL22V10.
- CPLD – Complex PLD. Jedná se o střední integraci obvodů, které většinou obsahují dvě úrovně propojovacích sítí. Obvody se vyrábějí v širokém spektru s počtem ekvivalentních hradel od 3 000 až 100 000.
- FPGA obvody. Jedná se o obvody s nejvyšší integrací, kde počet ekvivalentních hradel jde do miliónu.

## Příloha A - Značky používané k znázornění logické sítě

| Název prvku                                        | Naše značení                                                                        | Značka podle ANSI                                                                    |
|----------------------------------------------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| AND hradlo                                         |    |    |
| NAND hradlo                                        |    |    |
| OR hradlo                                          |    |    |
| NOR hradlo                                         |    |    |
| XOR hradlo                                         |    |    |
| XNOR hradlo                                        |   |   |
| Invertor                                           |  |  |
| Budič s přímým a negovaným výstupem                |  |  |
| Budič s třístavovým výstupem                       |                                                                                     |  |
| D klopný obvod reagující na hranu bez R a S vstupů |  |  |
| D klopný obvod reagující na hranu s R a S vstupy   |  |  |

|                                                            |                                                                                     |                                                                                      |
|------------------------------------------------------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| <b>D klopný obvod reagující na úroveň bez R a S vstupů</b> |    |    |
| <b>D klopný obvod reagující na úroveň s R a S vstupy</b>   |    |    |
| <b>Multiplexer</b>                                         |    |    |
| <b>Demultiplexer</b>                                       |   |   |
| <b>Odbočka na vodiči</b>                                   |  |  |
| <b>Křížení vodičů</b>                                      |  |  |

## Příloha B – Termíny

|                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ABEL – HDL            | ABEL Hardware Description Language, první programovací jazyk pro návrh číslicových systémů, který uvedla na trh společnost DATA I/O v roce 1983.                                                                                                                                                                                                                                                                                                                                                                      |
| Accellera             | Skupina zabývající tvorbou nových standardů pro jazyky Verilog a VHDL za účelem zvýšení produktivity návrhářských prací. Jedná se o definici nových standardů a formátů dat pro výměnu, zpracování nových postupů návrhů apod.                                                                                                                                                                                                                                                                                        |
| ASIC                  | Application Specific Integrated Circuits – integrované obvody pro aplikace, většinou se jedná o plně zákaznický návrh.                                                                                                                                                                                                                                                                                                                                                                                                |
| $\mathcal{B}$ -funkce | Boolean funkce je zobrazení definičního oboru na obor hodnot, kde definiční obor je množina, která obsahuje vektory definované délky z logických hodnot 0 a 1. Obor hodnot je množina o dvou logických hodnotách 0 a 1. Jedná se o jednoznačný zápis chování logického kombinačního obvodu. K popisu se často používá pravdivostní tabulka. V praxi se často používá rozšířená $\mathcal{B}$ -funkce, která používá tři hodnoty, 0, 1 a hodnotu X – neurčitá hodnota. Rozšířená $\mathcal{B}$ -funkce zkracuje zápis. |
| BIST                  | Built In Self Test, prostředky vestavěného testování.                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| BS                    | Boundary Scan, architektura určená pro testování podle standardu IEEE 1149.                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| BSDL                  | Boundary Scan Description Language, jazyk pro popis Boundary Scan architektury, která je implementovaná v integrovaném obvodu.                                                                                                                                                                                                                                                                                                                                                                                        |
| BS registr            | Boundary Scan registr, který se hlavně podílí na výkonu testů v rámci BS architektury.                                                                                                                                                                                                                                                                                                                                                                                                                                |
| $\mathcal{B}$ -výraz  | Zápis chování logického kombinačního obvodu pomocí logických operací. Jednu $\mathcal{B}$ -funkci lze zapsat více $\mathcal{B}$ -výrazy.                                                                                                                                                                                                                                                                                                                                                                              |
| CAD                   | Computer Aided Design, návrh pomocí počítače.                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| CAM                   | Contests Access Memory, asociativní paměť, paměť, kde se k datům přistupuje pomocí klíče.                                                                                                                                                                                                                                                                                                                                                                                                                             |
| CPLD                  | Complex Programmable Logic Device, střední kategorie PLD prvků.                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| DDR                   | Double Data Rate, dvojnásobná přenosová kapacita aplikovaná u synchronního přenosu. Data jsou vzorkována na obě hrany hodinového impulsu (vzestupná a sestupná). V jedné periodě hodin jsou přenášeny dva bity informace.                                                                                                                                                                                                                                                                                             |
| DFT                   | Design For Test, návrh pro snadné testování                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| DNF                   | Disjunktivní Normálová Forma - viz součtová normálová forma.                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| DPS                   | Deska Plošného Spoje.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| Dynamická RAM         | Dynamická RAM paměť uchovává informaci ve formě náboje na kondensátoru. Kondensátor se časem vybíjí, proto dynamické paměti mají režim obnovení informace (náboje) s anglickým názvem refresh. V porovnání se statickou RAM jsou pomalejší, ale mají vyšší kapacity.                                                                                                                                                                                                                                                  |
| EDA                   | Electronic Design Automation – automatizace elektronického návrhu.                                                                                                                                                                                                                                                                                                                                                                                                                                                    |

---

|               |                                                                                                                                                                                                                                 |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ECAD          | Electronic Computer-Aided Design – elektronický návrh pomocí počítače.                                                                                                                                                          |
| EEPROM        | Electrically Erasable PROM, elektricky mazatelná a programovatelná paměť PROM. Umožňuje práci s jednotlivými buňkami paměti. Často označována jako NV RAM.                                                                      |
| EPROM         | Erasable Programmable Read Only Memory, PROM paměť s možností vymazání a opětovného programování, reprogramovatelnost.                                                                                                          |
| FIFO          | First In First Out, paměťová struktura s funkcí fronty.                                                                                                                                                                         |
| Firmware      | Programy pro procesor, obsahy PLD prvků, obsahy PROM pamětí, prostě vše co určuje vlastnosti číslicového systému a je ukryto jeho uživateli.                                                                                    |
| Flash Memory  | Elektricky mazatelná a programovatelná PROM paměť. Mazání se uskutečňuje nad větším počtem buněk paměti nazývaných sektory (16 buněk až celý obsah).                                                                            |
| FPGA          | Field Programmable Gate Array, nejvyšší kategorie PLD prvků.                                                                                                                                                                    |
| GAL           | Generic Array Logic, obvody se strukturou programovatelné AND matice a pevné OR matice s následnou OLMC buňkou. GAL obvody jsou reprogramovatelné. OLMC buňka umožňuje výstup obvodu buď kombinační nebo s paměťovým elementem. |
| Hard procesor | Procesor jako blok, který je umístěn jako doplňující blok do PLD prvku. Opak je soft procesor.                                                                                                                                  |
| HDL           | Hardware Description Language. Označení pro skupinu programovacích jazyků pro návrh číslicových systémů. Většinou se jedná o ABEL-HDL, VHDL a Verilog.                                                                          |
| HDVL          | Hardware Description and Verification Language, jazyk pro popis a verifikaci hardwaru. Jedná se o rozšíření jazyka HDL o možnosti testování návrhu. Prvním představitelem této skupiny je SystemVerilog.                        |
| IBIS          | I/O Buffer Information Specification, model popisující vlastnosti vstupu a výstupu integrovaného obvodu pro potřeby simulace spojů na desce plošné spoje.                                                                       |
| ICT           | In Circuits Test – princip testování, kdy se pomocí speciálních sond (hřebíky) snímají a vnučují hodnoty signálů do zapojení. ICT se aplikuje na desky plošných spojů.                                                          |
| IEEE          | Institute of Electrical and Electronics Engineers, Inc., je nezisková organizace, která sdružuje inženýry, vědce z oblasti elektrických, elektronických a počítačových oborů. Je známa vydáváním standardů.                     |
| IO            | Integrovaný obvod.<br>Input, Output – vstup a výstup.                                                                                                                                                                           |
| IO Cell       | In Out Cell, IO buňka, vstupně-výstupní buňka.                                                                                                                                                                                  |
| IO buňka      | Vstupně-výstupní buňka PLD obvodu.                                                                                                                                                                                              |
| IP            | Intellectual Properties, systém, kdy lze zakoupit hotový návrh bloku                                                                                                                                                            |

---

|                      |                                                                                                                                                                                                                                 |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                      | číslicového systému.                                                                                                                                                                                                            |
| ISC                  | In System Configurable, systém pro programování PLD obvodů, který je přímo zapracován v integrovaném obvodu. ISC je dáno standardem IEEE 1532.                                                                                  |
| ISP                  | In System Programming, dříve používaný pojem pro systém programování PLD obvodů, který je zapracován přímo v integrovaném obvodu. Dnes je tento pojem nahrazován pojmem ISC.                                                    |
| JEDEC                | Join Electron Device Engineering Council. Tato zkratka je spojována s formátem souborů, který obsahuje data pro programování PLD prvků.                                                                                         |
| JTAG                 | Join Test Access Group, skupina, která se podílela na vývoji BS a standardu IEEE 1149.                                                                                                                                          |
| KNF                  | Konjunktivní Normálová Forma – součinnová normálová forma.                                                                                                                                                                      |
| LIFO                 | Last In First Out, paměťová struktura s funkcí zásobníku.                                                                                                                                                                       |
| LVDS                 | Low Voltage Differential Signaling. Diferenciální logika pro přenos o vysokých přenosových kapacitách.                                                                                                                          |
| MTBF                 | Time Between Failure, střední doba mezi poruchami.                                                                                                                                                                              |
| MTTF                 | Mean Time To Failure, střední doba do poruchy.                                                                                                                                                                                  |
| NVM                  | Non-Volatile Memory, paměť s přímým přístupem, kde data se málo mění. Někdy se používá i termín energeticky nezávislá paměť.                                                                                                    |
| OLMC                 | Output Logic Macro Cell, buňka, která může být konfigurována s kombinačním výstupem nebo s paměťovým elementem.                                                                                                                 |
| OTP                  | One Time Programmable, označení pro jednu programovatelný prvek.                                                                                                                                                                |
| PAL                  | Programmable Array Logic, nejpobulárnější princip programovatelné logiky založený na programovatelné matici AND a pevné matici OR.                                                                                              |
| PCB                  | Printed Circuits Board, viz DPS – deska plošného spoje.                                                                                                                                                                         |
| PCI                  | Peripheral Component Interface, standardizována sběrnice PC počítačů.                                                                                                                                                           |
| PLA                  | Programmable Logic Array, princip programovatelné logiky založený na programovatelné matici AND a programovatelné matici OR.                                                                                                    |
| PLD                  | Programmable Logic Device, programovatelný logický prvek. Univerzální struktura pro realizaci logické sítě.                                                                                                                     |
| Pravdivostní tabulka | Zápis $\mathcal{B}$ -funkce do tabulky, kde ke každé kombinaci vstupních proměnných je přiřazena výstupní hodnota.                                                                                                              |
| Product term         | Viz součinnový term.                                                                                                                                                                                                            |
| PROM                 | Programmable Read Only Memory, paměť určená pouze pro čtení, jejichž obsah lze definovat speciálními postupy – programováním u uživatele. PROM lze také chápat jako PLD prvek s pevnou AND maticí a programovatelnou OR maticí. |
| QML                  | Quality Manufacturers List, soubor doporučení pro výrobu elektronických součástek určených pro speciální použití se zaměřením na kvalitu. Modernější verze od QPL.                                                              |
| QPL                  | Quality Part List, soubor doporučení pro výrobu elektronických součástek určených pro speciální použití. Předchůdce QML.                                                                                                        |
| RAM                  | Random Access Memory, paměť s náhodným přístupem. K datům se                                                                                                                                                                    |

|                                |                                                                                                                                                                                                                                                         |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                | přístupuje pomocí adresy. Varianty RAM paměti jsou statická RAM, dynamická RAM, jednoportová a dvouportová paměť.                                                                                                                                       |
| ROM                            | Read Only Memory, paměť určená pouze pro čtení. Používá se jako všeobecné označení nebo pro paměti jejichž obsah je zadán výrobcem v průběhu výroby prvku.                                                                                              |
| Soft processor                 | Procesor, který je dán zápisem ve VHDL nebo Verilog jazyku a může být umístěn do PLD prvku. Opak hard procesoru.                                                                                                                                        |
| Součinná normálová forma       | $\mathcal{B}$ -výraz ve tvaru součinu součtových termů,<br>např. $(a + b).(c + d) \dots$                                                                                                                                                                |
| Součinný term                  | $\mathcal{B}$ -výraz ve tvaru součinu proměnných nebo $\mathcal{B}$ -výrazů,<br>např. $a.b.c \dots$                                                                                                                                                     |
| Součtová normálová forma       | $\mathcal{B}$ -výraz ve tvaru součtu součinných termů,<br>např. $a.b + c.d \dots$                                                                                                                                                                       |
| Součtový term                  | $\mathcal{B}$ -výraz ve tvaru součtu proměnných nebo $\mathcal{B}$ -výrazů,<br>např. $a + b + c \dots$                                                                                                                                                  |
| Standard PLD                   | Nejnižší skupina PLD prvků, dnes se do této kategorie hlavně řadí PAL a GAL prvky.                                                                                                                                                                      |
| Statická RAM                   | Data v paměti jsou uchovávána pomocí klopného obvodu. V porovnání s dynamickou, statická RAM je rychlejší, mívají menší kapacity a nepotřebuje režim obnovení informace – refresh.                                                                      |
| SystemC                        | Systém pro popis a verifikaci návrhu. Jedná se o aplikaci jazyka C++ a knihoven. Systém kombinuje hardwarové a softwarové řešení návrhu číslicového systému.                                                                                            |
| SystemVerilog                  | Systém pro popis a verifikaci návrhu, který vychází z jazyka Verilog. Jazyk HDVL, který je rozšířením jazyka Verilog o možnosti testování a ověření návrhu.                                                                                             |
| Truth table                    | Pravdivostní tabulka pro zápis $\mathcal{B}$ -funkce.                                                                                                                                                                                                   |
| UDNF                           | Úplná Disjunktivní Normálová Forma – viz úplná součtová normálová forma.                                                                                                                                                                                |
| Úplná součtová normálová forma | $\mathcal{B}$ -výraz ve tvaru součtu součinů, kde v každém součinném termu jsou použity všechny proměnné $\mathcal{B}$ -funkce. Například pro $\mathcal{B}$ -funkci o třech vstupních proměnných je UDNF ve tvaru<br>$a.b.c + !a.b.c + a.!b.!c + \dots$ |
| Verilog                        | Programovací jazyk pro návrh číslicových systému. Je dán standardem IEEE 1364.                                                                                                                                                                          |
| VHDL                           | VHSIC Hardware Description Language, kde VHSIC je Very High Speed Integrated Circuits, programovací jazyk pro návrh číslicových systémů. Je dán standardem IEEE 1076.                                                                                   |
| VITAL                          | VITAL je standard pro zápis knihoven, které popisují konkrétní PLD prvek a slouží k časové simulaci číslicového návrhu v konkrétním PLD prvku. VITAL je dán standardem IEEE 1076.4.                                                                     |

## Literatura

- [Actel 2002] IBIS Models: Background and Usage, Technical brief; Ibis\_AN.pdf; Actel Corporation 2002
- [Actel 2004] Using ProASICPLUS Clock Conditioning Circuits; PAPLUSdynamicAN; Actel Corporation; 2004
- [Altera 1990] The Maximalist Handbook; Altera Corporation; 1990
- [Altera 1995] PCI Compliance of Altera Device; AB 140; Altera Corporation; 1995
- [Altera 1999] Metastability in Altera Devices; AN042; Altera Corporation; 1999
- [Altera 2003] Using the ClockLock & ClockBoost PLL Features in APEX Devices; AN115; Altera Corporation; 2003
- [Atmel 2003] 1-megabit (128K x 8) Single 2.7-volt Battery-Voltage™ Flash Memory - AT49BV001A, AT49BV001AN, AT49BV001AT, AT49BV001ANT; doc3364; Atmel Corporation; 2003
- [Atmel 2004] 2-wire Serial EEPROMs 128K (16,384 x 8), 256K (32,768 x 8) AT24C128, AT24C256; doc0670; Atmel Corporation; 2004
- [Becke, Forstner, Haseloff, Huchzermeier, 1997]  
G. Becke, P. Forstner P, E. Haseloff, J. Huchzermeier: Digital Design Seminar, Texas Instruments; 1997
- [Bleeker, Eijnden, Jong 1993]  
H. Bleeker, P. Eijnden, F. Jong: Boundary-Scan Test, A practical Approach; Kluwer Academic Publishers 1993; ISBN 0792392965
- [Business World 2006]  
J. Kapoun: Gordon Moore, spoluzakladatel společnosti Intel; Business World 4/2006; ISSN 1213-1709
- [Computerworld 2005]  
P. Louda: Moorův zákon: stálce v oblasti IT; Computerworld 16/2005; ISSN 1210-9924
- [Diviš, Chmelíková a Zdrálek 2005]  
Z. Diviš, Z. Chmelíková a J. Zdrálek: Logické obvody; skriptu VŠB-TU Ostrava 2005; ISBN 80-248-0829-3
- [Fairchild 2000] LVDS Fundamentals; AN-5017; Fairchild Semiconductor 2000
- [Frištacký, Kolesár, Kolenička, Hlavatý 1986]  
N. Frištacký, M. Kolesár, J. Kolenička, J. Hlavatý: Logické systémy; ALFA Bratislava 1986
- [IDT 1996] S. Hronik: Designing for Hot Insertion and Multiple Power Supply systems; AN-158; Integrated Devices Technology, Inc.; 1996
- [Latticesemi 2003] isoLSI/GAL Matastability Report; Lattice Semiconductors Corporation; 2003

- 
- [Latticesemi 2004] Metastability in Lattice Devices; TN1055; lattice Semiconductors; 2004
- [Latticesemi 2004] sysCLOCK PLL Usage Guide for ispXPGA, ispGDX2, ispXPLD and ispMACH5000VG Devices; TN1003; Lattice Semiconductors Corporation; 2004
- [Lucent 1997] Product selection guide, 50<sup>th</sup> Anniversary – 1<sup>st</sup> Transistor; Lucent Technologies 1997
- [Magdaluyo 2000] M. Magdaluyo: Hot-swap your way to high availability; magazine EDN March 30, 2000
- [Musil a kol. 1994] V. Musil, V. Stříbrný, B. Weber, J. Kolouch, M. Přádka, J. Popelek, P. Hazdra, Z. Rozehnal: Návrh integrovaných obvodů; PC-DIR 1994; ISBN 80-214-0561-9
- [Parker 2003] K. P. Parker: The Boundary-Scan Handbook; Kluwer Academic Publishers 2003; ISBN 1-4020-7496-4
- [Pericom 2002] P. Li: The Behavior of CMOS in Hot Insertion; AB39; Pericom Semiconductor Corporation; 2002
- [Pericom 2003] P. Li: LVTC Logic family for live insertion using standard CMOS process; AN62; Pericom Semiconductor Corporation; 2003
- [Philips 1995] Philips PLDs support live insertion applications; AN051; Philips Semiconductors; 1995
- [Philips 2002] Bus Switches; Philips 2002
- [QPL 1991] MIL-M-38510J; Military specification microcircuits, General specification for ...
- [QML 2003] MIL-PRF-38535; Integrated circuits (microcircuits) manufacturing, general requirements for ...
- [Texas 1996a] The Bergeron Method, A Graphic Method for Determining Line Reflections in Transient Phenomena; sdya014; Texas Instruments; 1996
- [Texas 1996b] E. Haseloff: Live insertion; sdya012; Texas Instruments; 1996
- [Texas 1996c] R. Ammar: Understanding Advanced Bus-Interface Products; scaa029; Texas Instruments; 1996
- [Texas 1997a] L. Katz: Bus-Interface Devices With Output-Damping Resistors or Reduced-Drive Outputs; scba012a; Texas Instruments; 1997
- [Texas 1997b] E. Haseloff: Metastable Response in 5-V Logic Circuits; sdya006; Texas Instruments; 1997
- [Texas 1997c] IEEE STD 1149.1 (JTAG) Testability, Primer; ssya002c; Texas Instrument; 1997
- [Texas 1998] Boundary Scan Logic, Fully compliant with IEEE STD 1149.1 (JTAG); Product bulletin Texas Instrument; 1998
-

- 
- [Texas 1999] S. Feodorov, R. Ammar: Voltage Translation (5 V, 3.3 V, 2.5 V, 1.8 V), Switching Standards, and Bus Contention; scya006; Texas Instrument; 1999
- [Texas 2001] E. Haseloff: Bus Hold Circuit; scla015; Texas Instruments; 2001
- [Texas 2003a] Logic Selection Guide, Second Half 2003; sdyu001t; Texas Instrument; 2003
- [Texas 2003b] F. Dehmelt: Suitable LVDS Architectures; slla147; Texas Instruments 2003
- [Texas 2004] Logic Selection Guide, First Half 2004; sdyu001u; Texas Instrument; 2004
- [Xilinx 1997a] Metastability Considerations; XAPP077; Xilinx Inc.; 1997
- [Xilinx 1997b] Metastable Recovery; XAPP094; Xilinx Inc.; 1997
- [Xilinx 2003a] CoolRunner XPLA3 CPLD; ds012; Xilinx Inc.; 2003
- [Xilinx 2003b] Using Digital Clock Managers (DCMs) in Spartan-3 FPGAs; XAPP462; Xilinx Inc.; 2003

## Internet

- [Internet 01] [www.verilog.net](http://www.verilog.net); www stránky věnované Verilogu, včetně volných programů.
- [Internet 02] [www.asic-world.com/verilog/verifaq1.html](http://www.asic-world.com/verilog/verifaq1.html); www stránky se základními otázkami o Verilogu.
- [Internet 03] [www.pcisig.com](http://www.pcisig.com); www stránky specifikace sběrnice PCI.
- [Internet 04] <http://csdl.computer.org/comp/proceedings/glsvlsi/1997/7904/00/79040051abs.htm>
- [Internet 05] [http://www2.ele.ufes.br/~ailson/digital2/A Hardware Eng Guide to VHDL.pdf](http://www2.ele.ufes.br/~ailson/digital2/A%20Hardware%20Eng%20Guide%20to%20VHDL.pdf)
- [Internet 06] Transistor History 101 by Bob McGarrah;  
<http://users.arczip.com/rmcgarra2/xstrhist.html>
- [Internet 07] <http://www.accellera.org>
- [Internet 08] <http://www.systemverilog.org>
- [Internet 09] <http://www.systemC.org>
- [Internet 10] <http://www.eda.org>

## Závěrečný test



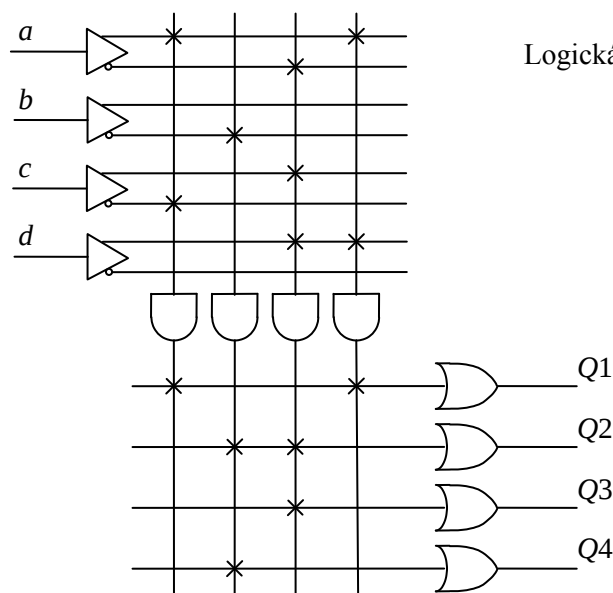
Otázka 1 Výraz  $ab + c$  je

- a) v úplném součtovém normálovém tvaru
- b) v úplném součtinovém normálovém tvaru
- c) v součtovém normálovém tvaru
- d) v součtinovém normálovém tvaru

Otázka 2 Výraz  $abc + ab(\text{not } c)$  je

- a) v úplném součtovém normálovém tvaru
- b) v úplném součtinovém normálovém tvaru
- c) v součtovém normálovém tvaru
- d) v součtinovém normálovém tvaru

Na obrázku je logická síť, a další otázky se na něj budou odvolávat



Logická síť pro závěrečný test

Obr. T - 1

Otázka 3 Na obrázku T - 1 výstupu Q1 odpovídá rovnice

- a)  $y = (a + !c) \cdot (a + d)$
- b)  $y = a \cdot !c + a \cdot d$
- c)  $y = !a \cdot c \cdot d$
- d)  $y = !b$

Otázka 4 Na obrázku T - 1 výstupu Q2 odpovídá rovnice

- a)  $y = !a \cdot c \cdot d$
- b)  $y = (!a + c + d)$
- c)  $y = !b + !a \cdot c \cdot d$
- d)  $y = !b \cdot (!a + c + d)$

Otázka 5 Na obrázku T - 1 výstupu Q3 odpovídá rovnice

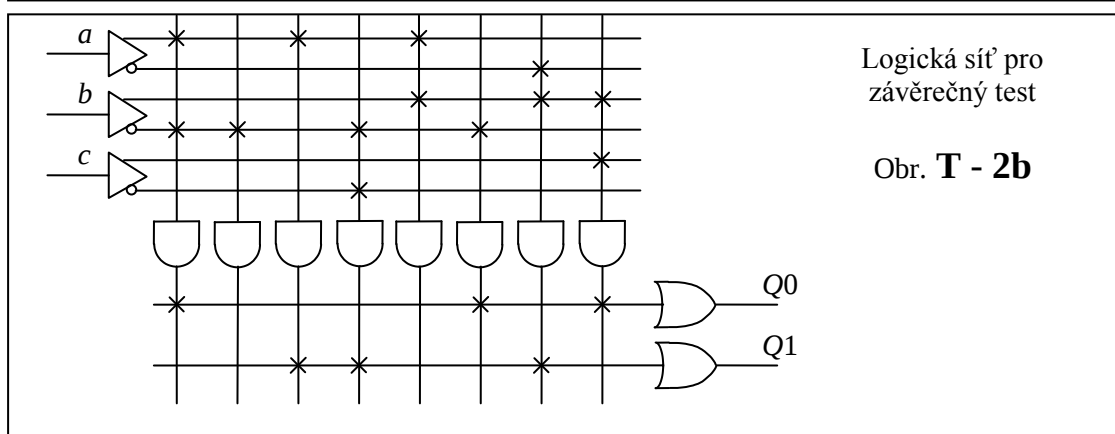
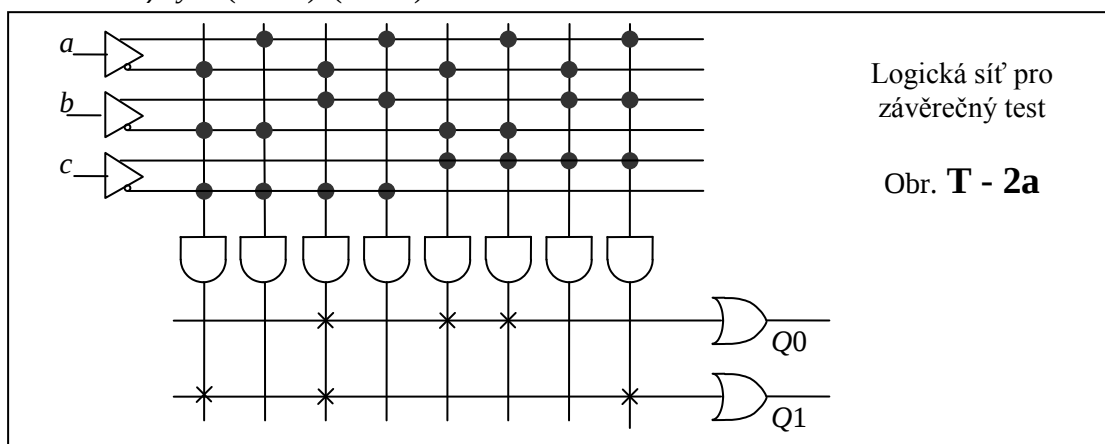
- a)  $y = !a . c . d$
- b)  $y = a . !c . !d$
- c)  $y = !b$
- d)  $y = b$

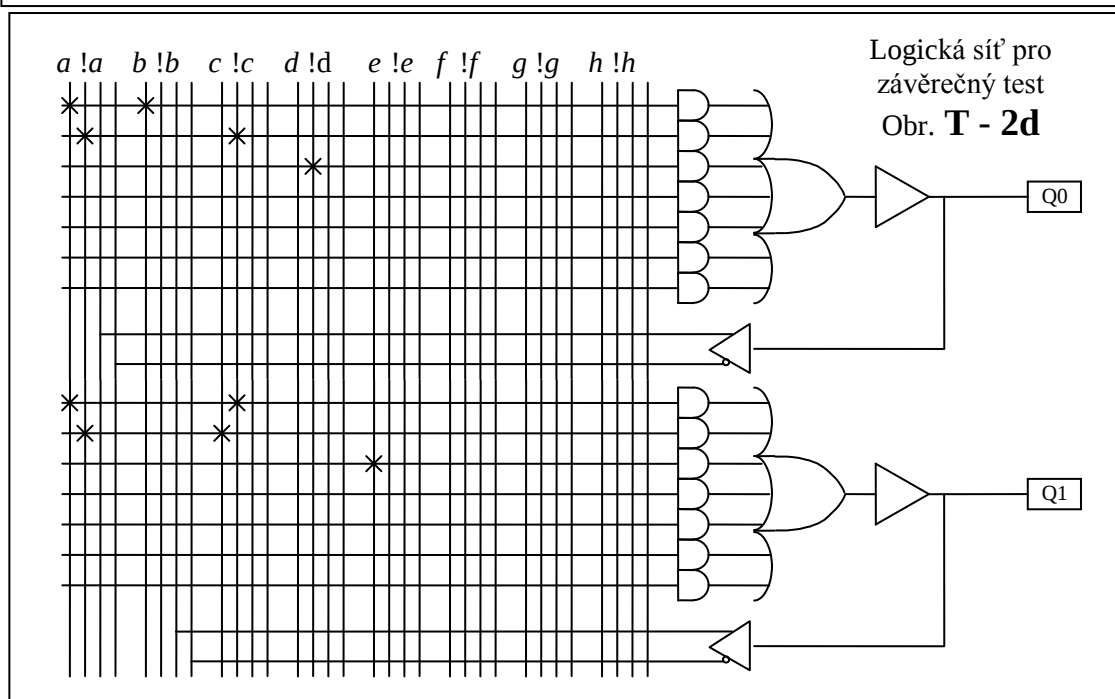
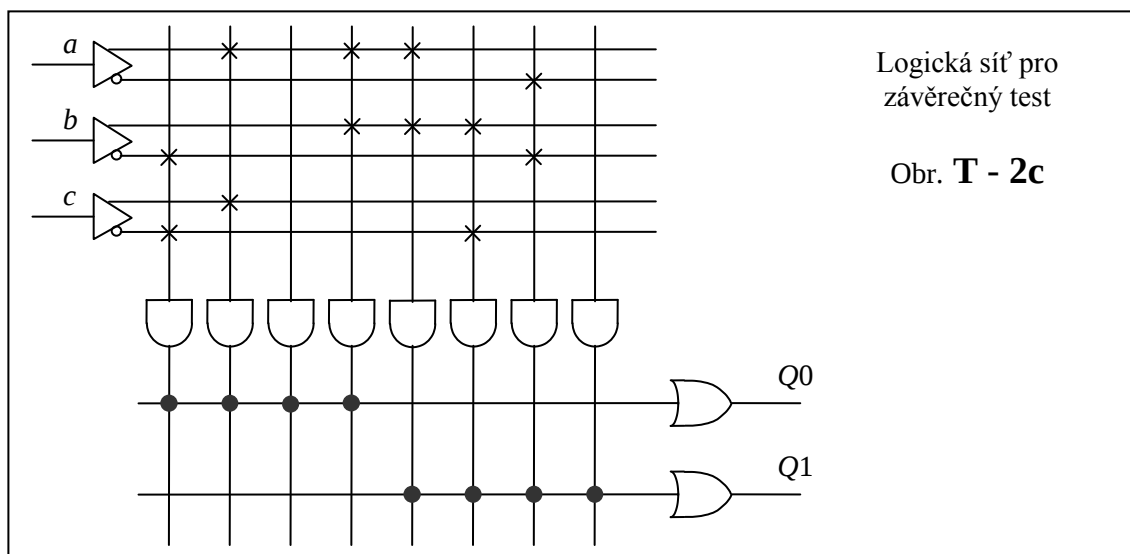
Otázka 6 Na obrázku T - 1 výstupu Q4 odpovídá rovnice

- a)  $y = !a . c . d$
- b)  $y = a . !c . !d$
- c)  $y = !b$
- d)  $y = b$

Otázka 7 Jsou dány následující rovnice. Ke každé rovnici určete obrázek, na kterém je realizována. Existuje i varianta, že rovnice nemá na obrázcích příslušnou logickou síť, pro správné řešení se rovnici přiřadí obrázek **T - 2e**.

- a)  $y = !b$
- b)  $y = b$
- c)  $y = !c . !b + c . a + b . a$
- d)  $y = (c + b) . (!c + !a) . (!b + !a)$
- e)  $y = !c . !a + b . a + !d$
- f)  $y = (c + a) . (!b + !a) . d$
- g)  $y = !c . !b . !a + !c . b . !a + c . b . a$
- h)  $y = (c + b + a) . (c + !b + a) . (!c + !b + !a)$
- i)  $y = !c . b . !a + b . a + c . !b . a + !c . b . !a$
- j)  $y = (c + !b + a) . (b + a) . (!c + b + !a) . (c + !b + a)$
- k)  $y = a + !c . !b + b . !a$
- l)  $y = !a . (c + b) . (!b + a)$
- m)  $y = !c . a + c . !a + e$
- n)  $y = (c + !a) . (!c + a) . !e$





Jedná se o jiné zapojení zapojení logické sítě, které  
není na výše uvedených obrázcích

Logická síť pro  
závěrečný test

Obr. **T - 2e**

Otázka 8 Obvody PROM mají

- a) obě matice AND a OR neprogramovatelné
- b) AND matici pevnou a OR matici programovatelnou
- c) AND matici programovatelnou a OR matici pevnou
- d) obě matice AND a OR programovatelné

- Otázka 9 Obvody PROM mají organizaci  $2^n \times m$  ,
- a) kde  $n$  je počet adresných vstupů a  $m$  počet datových výstupů
  - b) kde  $n$  je počet datových vstupů a  $m$  je počet adresných výstupů
- Otázka 10 Obvody PROM s organizací organizaci  $2^n \times m$  , mohou realizovat
- a) libovolnou  $\mathcal{B}$  funkci o  $n$  vstupech a  $m$  výstupech
  - b) libovolnou  $\mathcal{B}$  funkci o  $m$  vstupech a  $n$  výstupech
  - c) libovolnou  $\mathcal{B}$  funkci o  $2^n$  vstupech a  $m$  výstupech
  - d) libovolnou  $\mathcal{B}$  funkci o  $2^m$  vstupech a  $n$  výstupech
- Otázka 11 Pro realizaci zapojení v PROM obvodech se používá
- a)  $\mathcal{B}$  - výraz v součtovém normálovém tvaru
  - b)  $\mathcal{B}$  - výraz v úplném součtovém normálovém tvaru
  - c)  $\mathcal{B}$  - výraz v součinnovém tvaru
  - d)  $\mathcal{B}$  - výraz v úplném součinnovém tvaru
- Otázka 12 Obvody PLA mají
- a) obě matice AND a OR neprogramovatelné
  - b) AND matici pevnou a OR matici programovatelnou
  - c) AND matici programovatelnou a OR matici pevnou
  - d) obě matice AND a OR programovatelné
- Otázka 13 Pro realizaci zapojení v PLA obvodech se používá
- a)  $\mathcal{B}$  - výraz v součtovém normálovém tvaru
  - b)  $\mathcal{B}$  - výraz v součinnovém tvaru
  - c)  $\mathcal{B}$  - výraz v úplném součinnovém tvaru
- Otázka 14 Obvody PAL mají
- a) obě matice AND a OR neprogramovatelné
  - b) AND matici pevnou a OR matici programovatelnou
  - c) AND matici programovatelnou a OR matici pevnou
  - d) obě matice AND a OR programovatelné
- Otázka 15 PAL obvody mají
- a) Rekonfigurovatelný výstup a to buď jako kombinační nebo s registrem
  - b) Výstup je daný typem PAL obvodu a nelze jej měnit
- Otázka 16 Pro realizaci zapojení v PAL obvodech se používá
- a)  $\mathcal{B}$  - výraz v součtovém normálovém tvaru
  - b)  $\mathcal{B}$  - výraz v součinnovém tvaru
  - c)  $\mathcal{B}$  - výraz v úplném součinnovém tvaru
- Otázka 17 Obvody GAL mají
- a) obě matice AND a OR neprogramovatelné
  - b) AND matici pevnou a OR matici programovatelnou
  - c) AND matici programovatelnou a OR matici pevnou
  - d) obě matice AND a OR programovatelné
- Otázka 18 Obvody GAL používají na svém výstupu
- a) OLMC buňku - Output Logic MacroCell
  - b) LUT tabulku – Lookup Table
  - c) Metastabilní buňku

- Otázka 19 GAL obvody jsou
- a) vícekrát programovatelné, mazání je většinou elektrické
  - b) programovatelné maskou při samotné výrobě u výrobce
  - c) jednou programovatelné, OTP
- Otázka 20 OLMC buňka slouží
- a) ke konfiguraci výstupu GAL obvodu a to jako kombinační nebo registrový výstup
  - b) ke konfiguraci vstupu GAL obvodu a to jako kombinační nebo registrový výstup
  - c) k propojení AND a OR matice.
- Otázka 21 GAL obvod dokáže nahradit
- a) pouze jeden typ PAL stejné organizace
  - b) více typů PAL obvodů
  - c) jedině PAL obvody s nízkým příkonem
- Otázka 22 Počet součinných termů součtového výrazu omezuje architekturu
- a) PROM obvodů
  - b) PLA obvodů
  - c) PAL nebo GAL obvodů
- Otázka 23 Pro realizaci zapojení v GAL obvodech se používá
- a)  $\mathcal{B}$  - výraz v součtovém normálovém tvaru
  - b)  $\mathcal{B}$  - výraz v součinném tvaru
  - c)  $\mathcal{B}$  - výraz v úplném součinném tvaru
- Otázka 24 Obvody CPLD se vyznačují
- a) GAL strukturou
  - b) dvěma propojovacími sítěmi, nadřazená propojuje jednotlivé PLD obvody
  - c) jednou rozsáhlou propojovací sítí
- Otázka 25 Obvody FPGA se vyznačují
- a) GAL strukturou
  - b) dvěma propojovacími sítěmi, nadřazená propojuje jednotlivé PLD obvody
  - c) jednou rozsáhlou propojovací sítí
- Otázka 26 Obvody FPGA se vyznačují
- a) jednou programovatelnou propojkou
  - b) vícekrát programovatelnou propojkou
  - c) propojkami na bázi statické RAM
- Otázka 27 Pojem elektronický podpis nejlépe vystihuje
- a) jedná se o bit, který zabráňuje čtení obsahu PLD prvku, z důvodů nelegálního kopírování obsahu
  - b) speciální zapojení PLD prvku, kdy se nahrává obsah paměťových elementů z důvodů testování
  - c) skupina bitů definovaná návrhářem, doporučuje se, aby obsahovala kód obsahu PLD prvků
- Otázka 28 Boundary Scan v PLD prvcích nejlépe charakterizuje věta
- a) jedná se o speciální soubor ke každému PLD prvku, který obsahuje popis zapojení PLD prvku pro účely časové simulace.
  - b) jedná se o speciální soubor, který obsahuje popis vstupních a výstupních signálů PLD prvku pro účely simulace.
  - c) jedná se o speciální architekturu, která je určena k testování PLD prvku, desky plošného spoje i celého systému

Otázka 29 Zkratku IP v PLD prvcích nejlépe vystihuje věta

- a) jedná se o implementaci Internet Protokolu v PLD prvku
- b) Intellectual Properties. Jedná se o speciální zapojení zapsané jazyku VHDL nebo Verilog, které lze získat bez poplatků nebo za úplatu. Jedná se např. o zapojení Ethernet adaptéru, procesorů, šifrovacích algoritmů a jiné.
- c) In Programming. Jedná se o programování PLD prvků přímo na desce plošného spoje.

Otázka 30 Pojem metastabilita nejlépe vystihuje věta

- a) jedná se o generování falešných impulzů na základě indukčního zákona. Původem jsou proudové změny hlavně zemnicích vodičích způsobené změnou směru proudů.
- b) jedná se o diferenciální logiku, která se používá pro sériový přenos o vysokých kapacitách řádově jednotky Gbitů za sekundu a hlavně je odolná proti vnějšímu rušení.
- c) jedná se o nestandardní chování D klopného obvodu, kdy výstup má nedefinovanou úroveň. Tento jev důsledkem nedodržení časových poměrů na hodinovém a datovém vstupu D klopného obvodu.

Otázka 31 Třístavová logika se dá charakterizovat

- a) výstup hradla obsahuje zapojení s otevřeným kolektorem pro realizaci montážních operací a vytváření obousměrných sběrnic.
- b) jedná se o speciální označení hradel, které na výstup mají definované dvě základní úrovně L a H, ale s vysokým proudovým zatížením řádově desítky mA. Základní použití je pro vytváření obousměrných sběrnic.
- c) jedná se o speciální hradla, které mají na výstupu definovány L, H úroveň a třetí stav (stav vysoké impedance). Základní použití je pro vytváření sběrnic.

## Správné odpovědi

### Správné odpovědi k závěrečnému testu



1 - c                  2 - a                  3 - b                  4 - c                  6 - c

Otázka 7 a varianty

Rovnice 7a – obrázek Test 2e

Rovnice 7b -      obrázek Test 2e

Rovnice 7c – obrázek Test 2c

Rovnice 7d – obrázek Test 2e

Rovnice 7e – obrázek Test 2d

Rovnice 7f – obrázek Test 2e

Rovnice 7g - obrázek Test 2a

Rovnice 7h - obrázek Test 2e

Rovnice 7i - obrázek Test 2e

Rovnice 7j - obrázek Test 2e

Rovnice 7k - obrázek Test 2b

Rovnice 7l - obrázek Test 2e

Rovnice 7m - obrázek Test 2d

Rovnice 7n - obrázek Test 2e

Otázka 8 a další

8 - b                  9 - a                  10 - a                  11 - b                  12 - d                  13 - a

14 - c                  15 - b                  16 - a                  17 - c                  18 - a                  19 - a

20 - a                  21 - b                  22 - c                  23 - a                  24 - b                  25 - c

26 - c                  27 - c                  28 - c                  29 - b                  30 - c                  31 - c