LaPSE

Laboratório de Prototipação Digital
e Sistemas Embarcados

Universidade do Vale do Rio dos Sinos - UNISINOS

Prototipação em *PLDs*

Introdução aos *PLDs*

Autor: Prof. Rodrigo Marques de Figueiredo

Agenda

- Definição;
- Contextualização;
- Escalas de Integração;
- Histórico;
- *Market Share*;
- Aplicações;
- Conceitos Básicos;
- Estrutura Eletrônica;
- Sinais;
- Interfaceamento;
- Fontes;
- Bibliografia Recomendada.

Definição

- *PLD – Programmable Logic Device*;
- São dispositivos eletrônicos construídos com uma estrutura configurável (programável);
- Não é baseado em uma arquitetura específica, mas sim em circuitos lógicos genéricos;
- Estes circuitos genéricos são capazes de reproduzir virtualmente quaisquer arquiteturas já existentes, ou ainda, arquiteturas dedicadas a aplicação totalmente novas e originais.

Contextualização

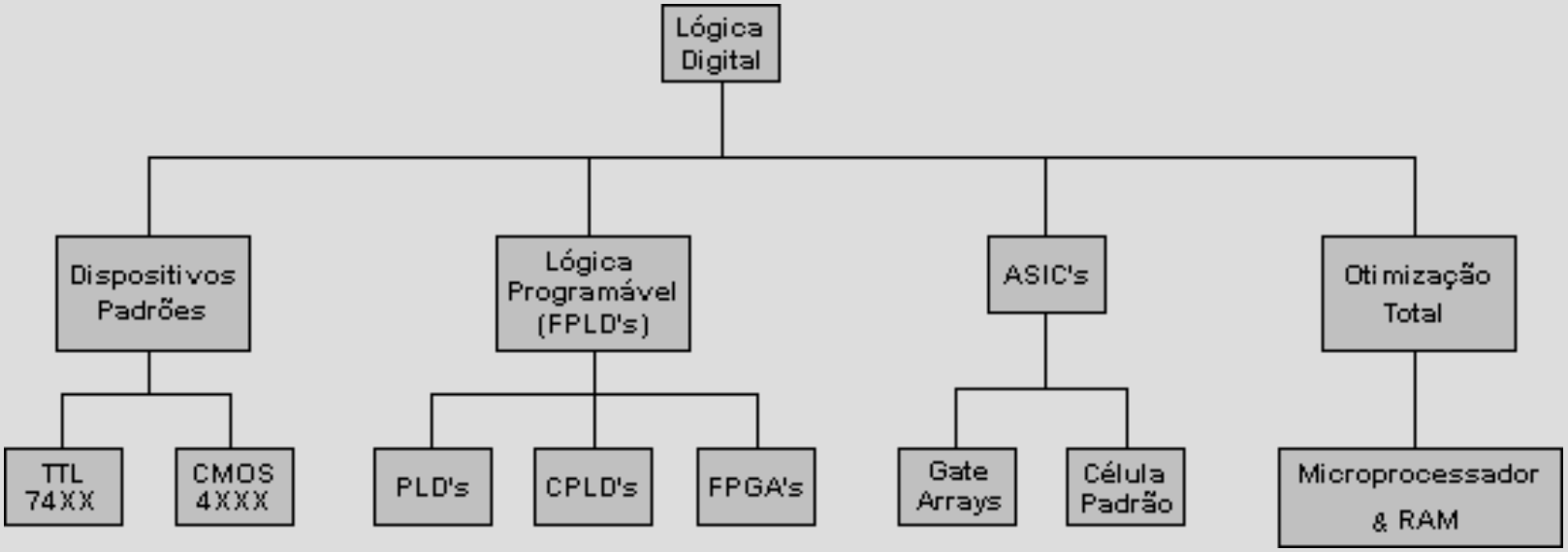
- A tecnologia dos dispositivos eletrônicos evoluiu dramaticamente nos últimos 50 anos;
- O transistor foi inventado no final dos anos 40;
- Popularizou-se nos anos 50;
- Em 1956 o cientista *Geoffrey W. A. Dummer* tentou sem sucesso fabricar o primeiro circuito integrado (CI) baseado em transistores;
- Entre 1956 e 1958 muitas outras tentativas por outros cientistas foram realizadas, tendo sucesso em 1958;
- Este CI foi chamado inicialmente de Circuito do Estado Sólido;

Contextualização

- No início dos anos 60 houve então a grande revolução da eletrônica moderna com a criação da patente do Circuito Unitário, em 20 de abril de 1961;
- A partir desta data popularizou-se a idéia do uso de CIs em projetos tanto digitais como analógicos;
- Começou-se então a criação de CIs com diferentes escalas de integração, o que criou uma classificação de gerações de CIs utilizando esta métrica;
- Atualmente é praticamente impossível pensar em um projeto que possa dispensar totalmente o uso de CIs.

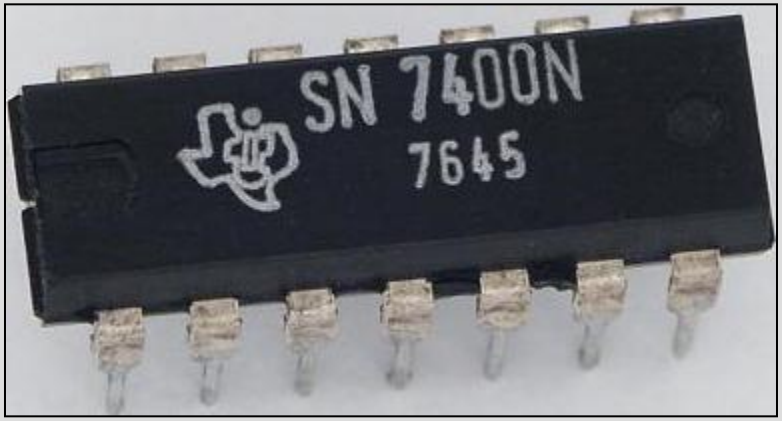
Contextualização

- Lógica Digital - Organização



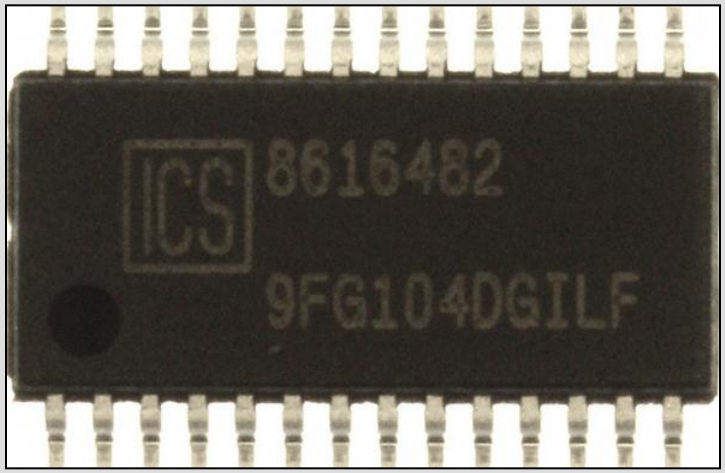
Escalas de Integração

- *SSI – Small Scale Integration:*
  - Produzidos no início dos anos 60;
  - Continham dezenas de transistores em sua arquitetura;
  - Utilizados nos projetos *Apollo* e *Minuteman*;
  - Em um prazo de 2 anos o consumo de CI por parte destes dois projetos baixou o custo de cada CI de US$ 1,000.00 para menos de US$ 25.00.



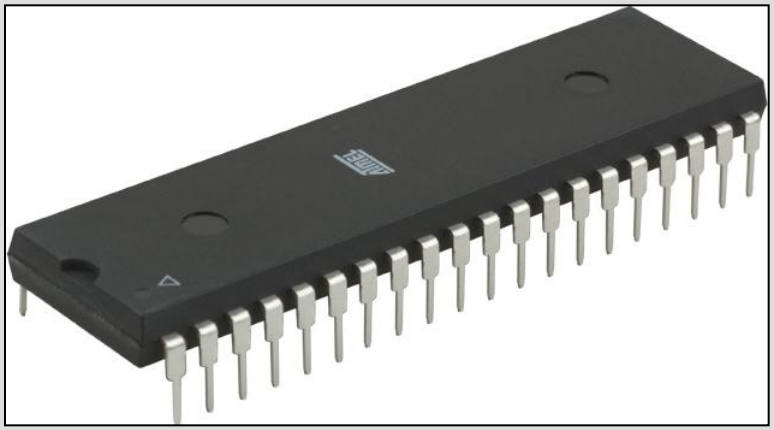
Escalas de Integração

- *MSI – Medium Scale Integration:*
  - Produzidos no final dos anos 60;
  - Continham centenas de transistores em sua arquitetura;
  - Possibilitava a implementação de circuitos (e conseqüentemente projetos) mais complexos que os CIs da geração anterior.



Escalas de Integração

- *LSI – Large Scale Integration:*
  - Produzidos na metade dos anos 70;
  - Continham dezenas de milhares de transistores em sua arquitetura;
  - O mercado alvo para sua produção era a indústria de computadores.



Escalas de Integração

- *VLSI – Very Large Scale Integration:*
  - Desenvolvidos e produzidos na metade dos anos 80;
  - Contém milhões de transistores em sua arquitetura;
  - Utilizados para praticamente todos os fins.

Escalas de Integração

- *ULSI – Ultra Large Scale Integration:*
  - Última palavra em CIs;
  - Utiliza nanotecnologia;
  - Introduzido no mercado em 2002.

Escalas de Integração

- *CPLDs* e *FPGAs*:
  - São produzidos com as tecnologias *VLSI* e *ULSI*;
  - Os *FPGAs* mais modernos, lançados a partir de 2004, salvo raras exceções são fabricados utilizando tecnologia *ULSI*;
  - Existem outros circuitos *PLDs* que fazem uso de tecnologias mais antigas, como *LSI* ou até mesmo *MSI*, porém estes estão caindo em desuso.

Histórico

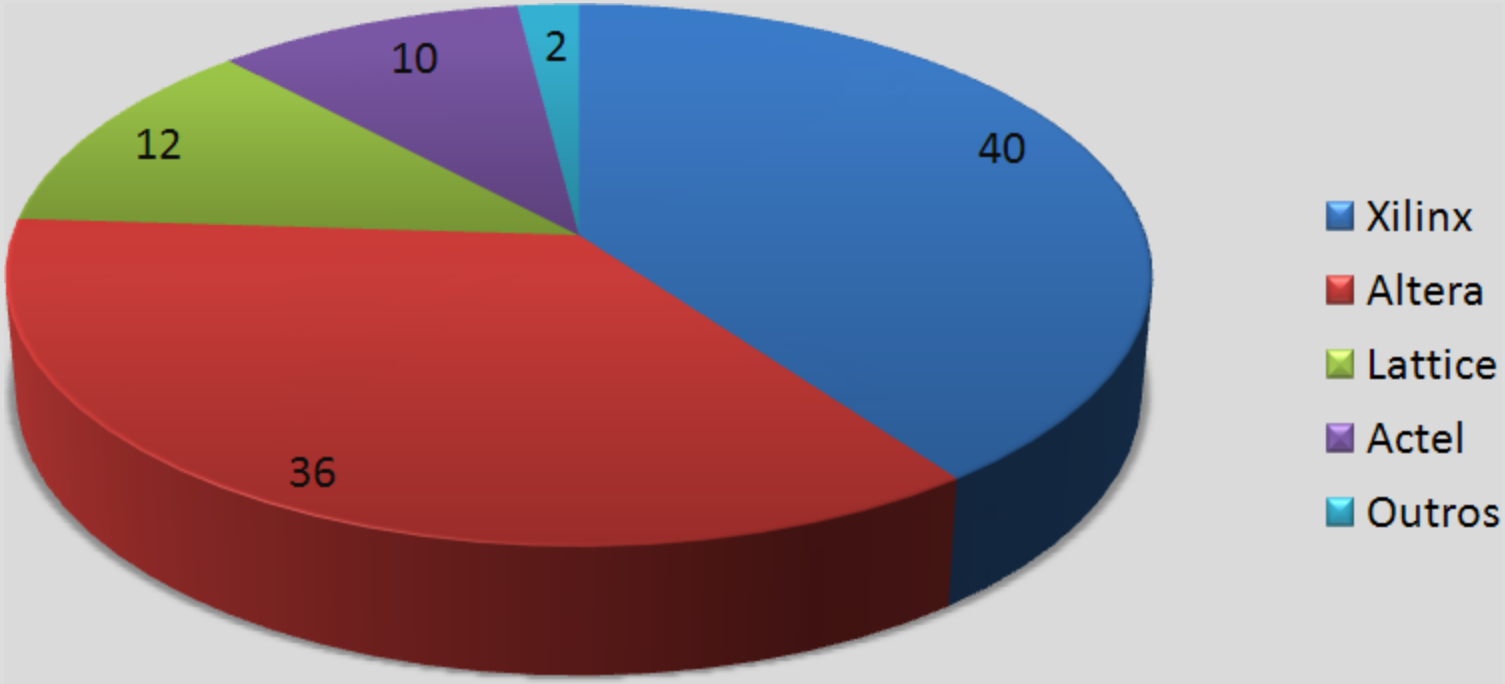
- Desde final dos anos 60 já se tinha a idéia de disponibilizar aos projetistas o poder de escolher a arquitetura comportamental de um CI;
- Em 1970 a *Texas Instruments* criou a *PLA* (*Programmable Logic Array*), colocando esta tecnologia em um dispositivo chamado **TMS2000**;
- Em 1971 a *General Electric* (*GE*) desenvolveu uma tecnologia baseda em *PROM*, apagada por ultravioleta, ao qual chamou de *EPLD*;
- Em 1973 a *National Instruments* introduziu o uso desta tecnologia (*PLA*) em um dispositivo chamado **DM7575**;
- Em 1974 a *GE* juntou-se a *Monolithic Memories*, quando desenvolveram em conjunto *PALA* (*Programmable Associative Logic Array*). Esta foi a semente da posterior arquitetura dos *FPGAs*.

Histórico

- Em 1983 primeira padronização de *HDLs* por parte do **IEEE** (padronizando o *VHDL* e o *Verilog*);
- Com isso tem-se o início da popularização em termos de mercado para os *PLDs*;
- Nascem as primeiras empresas especializadas em *PLDs*: *Altera* e *Lattice* (83), *Xilinx* (84) e *Actel* (85);
- Anos 90 entrada das tecnologias com interfaces *gigabit*;
- Ainda nos anos 90 *cores* de *PPC* são integrados à *FPGAs*;
- Anos 2000 adoção de tecnologias de 40nm;
- 2004 *VHDL-AMS* e primeiros *FPGA* de circuitos analógicos programáveis;
- 2009 *cores ARM* (32 bits) são integrados à *FPGAs*.
- 2010 tecnologia de 28nm é introduzida;

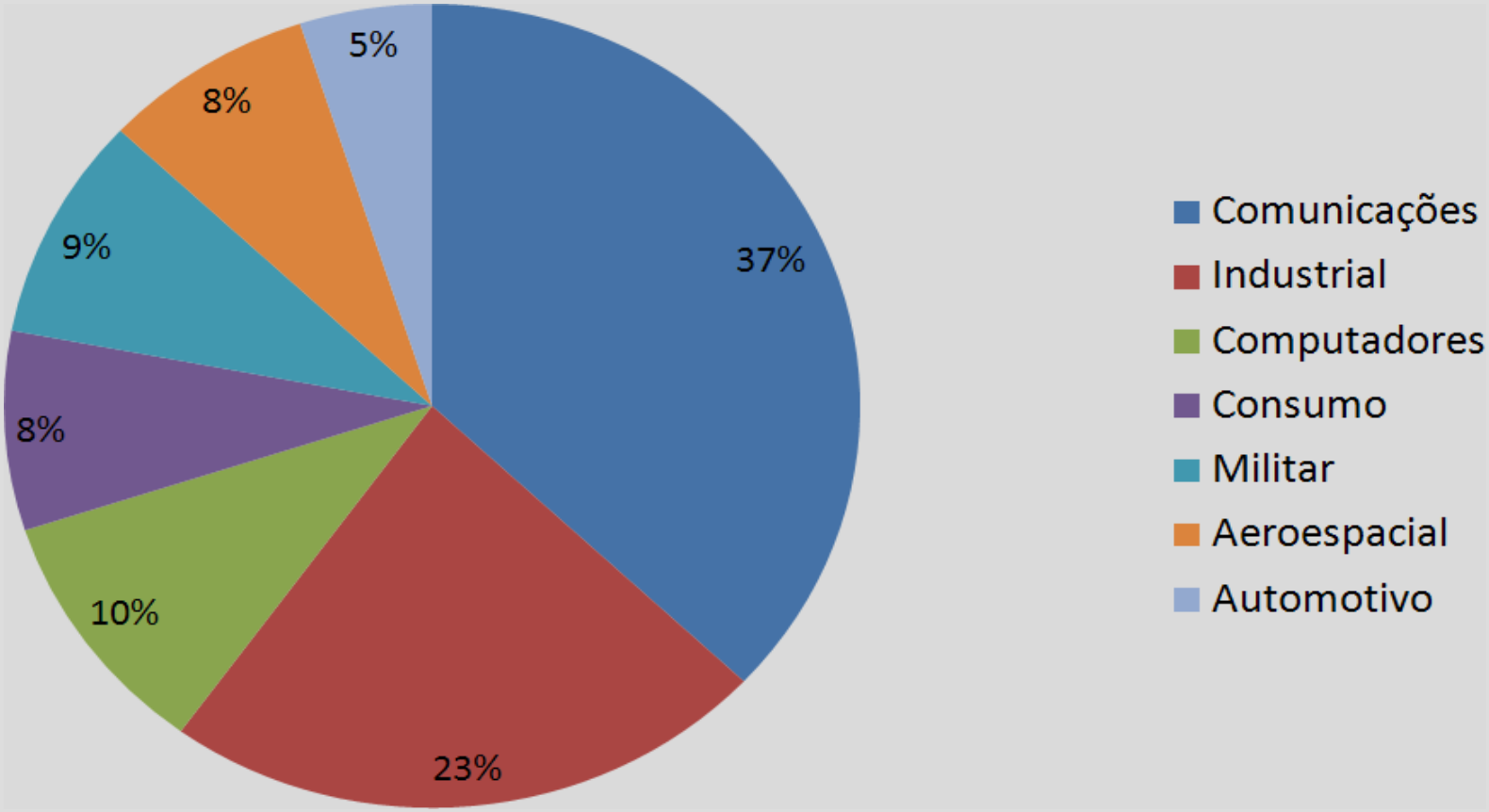
Market Share

- Fabricantes:



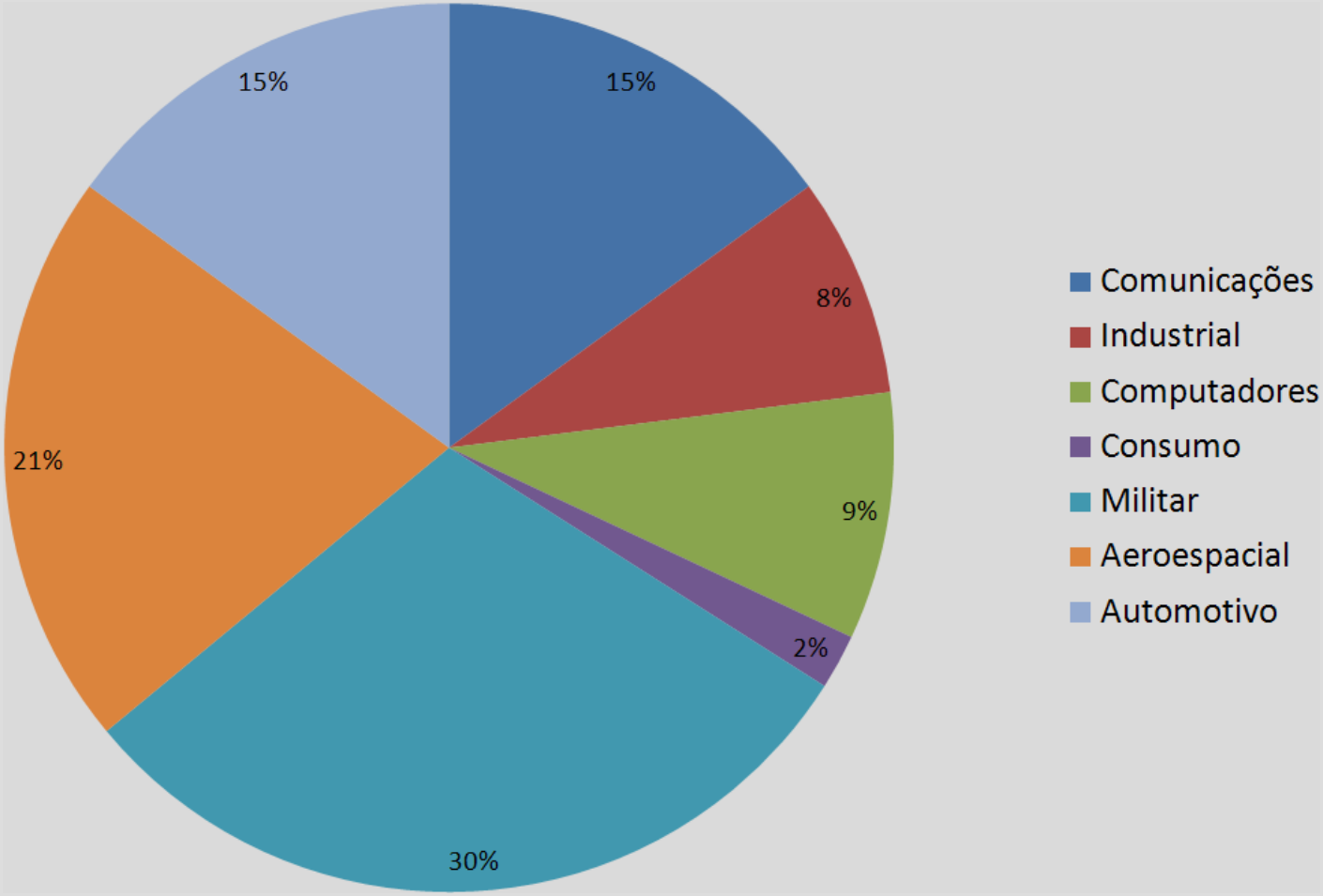
Market Share

- Áreas de Aplicação:



Market Share

- Investimento nas Áreas de Aplicação:

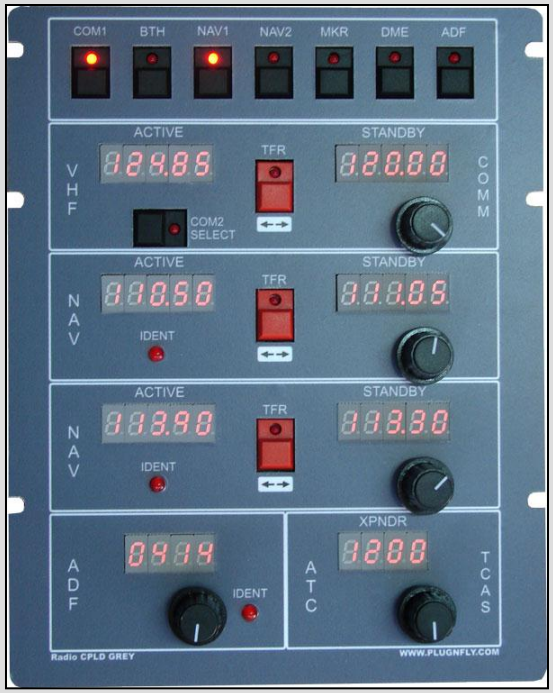


Aplicações

- As aplicações de *PLDs* são bastante diversas, indo desde aplicações *low-end* até aplicações *high-end*:
  - **Aplicações *low-end*:**
    - SPLDs;
    - CPLDs;
    - FPGAs (pequenos – com poucas macrocélulas).
  - **Aplicações *middle-end*:**
    - CPLDs;
    - FPGAs.
  - **Aplicações *high-end*:**
    - CPLDs;
    - FPGAs (grandes – muitas macrocélulas).

Aplicações

- Aplicações *low-end*:
  - Substituição de itens obsoletados;
  - Condensação de componentes para diminuir espaço em PCI;
  - Integração de soluções proprietárias com soluções de terceiros.



Aplicações

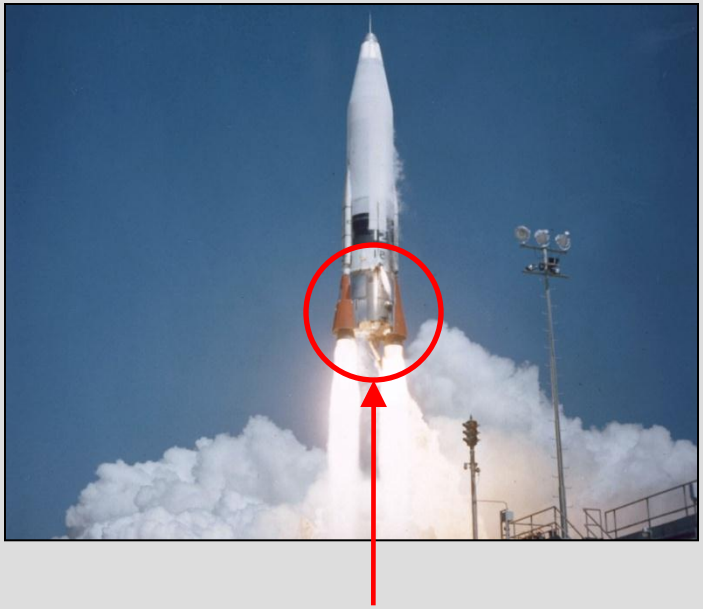
- Aplicações *middle-end*:
  - Co-processadores;
  - *DSPs* simples (detectores de *DTMF*);
  - Conversores de interface (modulação digital);
  - Computação paralela;
  - Multiprocessadores;
  - Arquiteturas de processamento proprietárias.

Aplicações

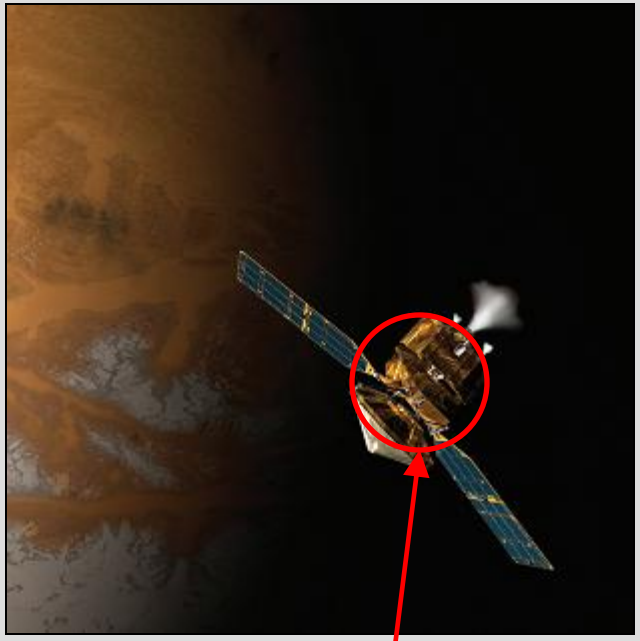
- Aplicações *high-end*:
  - Co-processadores;
  - *DSPs* avançados (reconhecimento facial);
  - Interfaces de alta velocidade;
  - Computação paralela massiva;
  - Multiprocessadores com *multithreads*;
  - Aplicações de alta confiabilidade.

Aplicações

- Aplicações *high-end*:



DSPs: controle de aceleração, posição, angulação, pressão dos gases, ...



DSPs: posição, células de carga, processamento de imagens, ...

Aplicações

- Aplicações *high-end*:

DSPs: sistema de visão, controles de aceleração, posição, angulação, ...

Aplicações

- Aplicações *high-end*:

DSPs: visão multiespectral.

DSPs: controle do ângulo de ataque (mantém nível do canhão independente da base).

DSPs: reconhecimento de alvo (amigo ou inimigo)

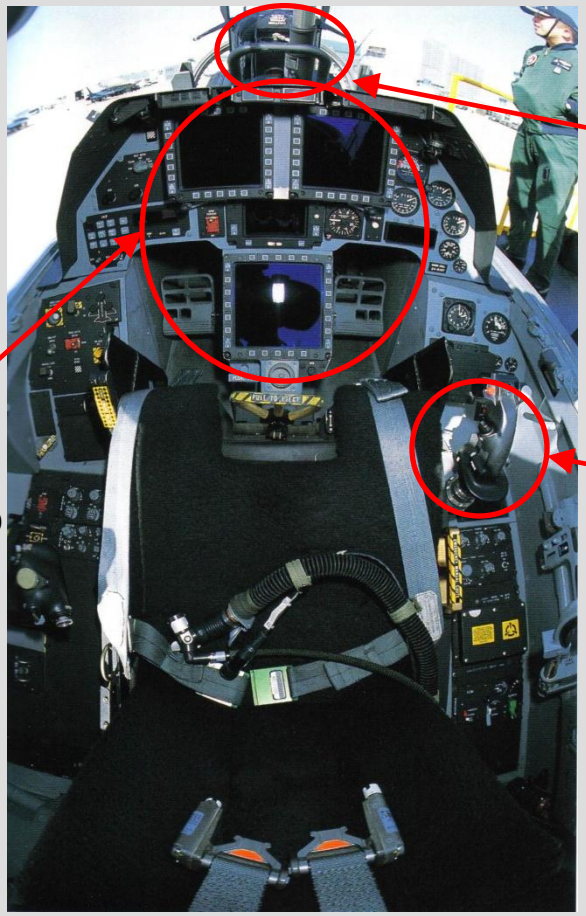
Aplicações

- Aplicações *high-end*:

Modernização
de aeronaves

Aplicações

- Aplicações *high-end*:



HUD: *Head-up Display*

DSPs: visão multiespectral; Informações de status; Processamento do sensoriamento da aeronave.

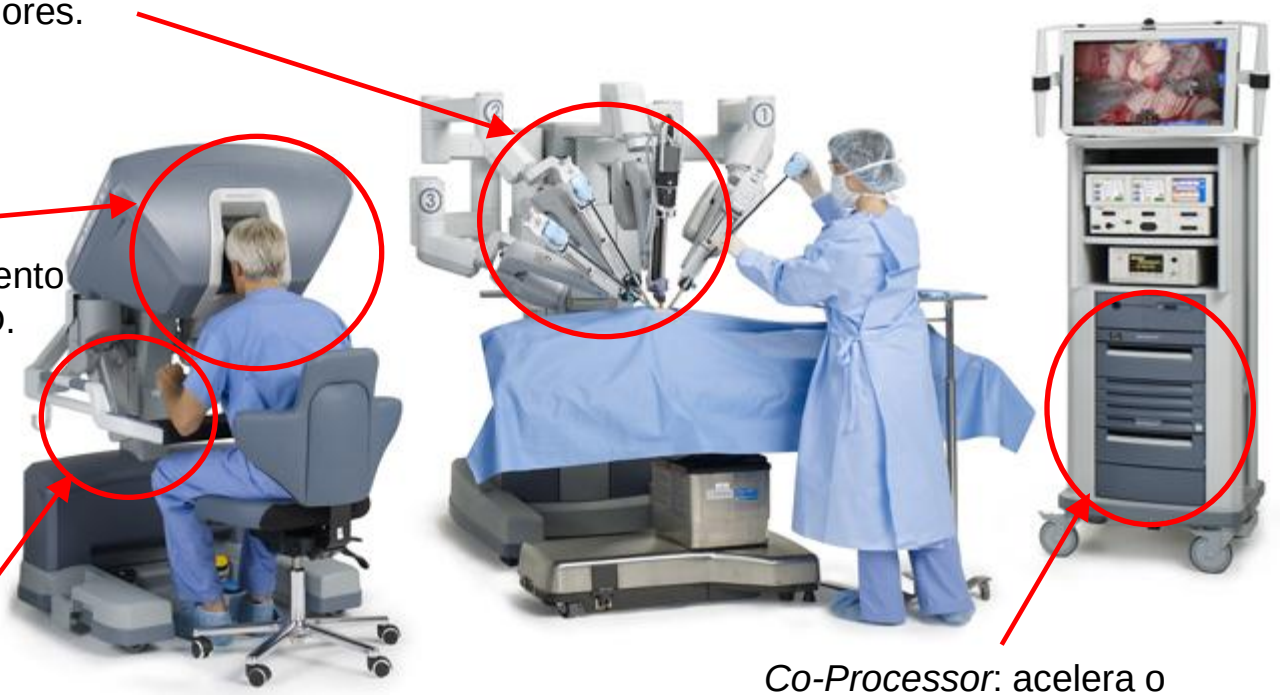
DSP: Comandos servo assistidos

Aplicações

- Aplicações *high-end*:

DSPs: Cálculo *online* da posição dos atuadores.

DSPs: Processamento de vídeo 3D.



DSPs: Cáculo da posição dos sensores.

Co-Processor: acelera o processamento dos dados dos sensores para passar para os atuadores

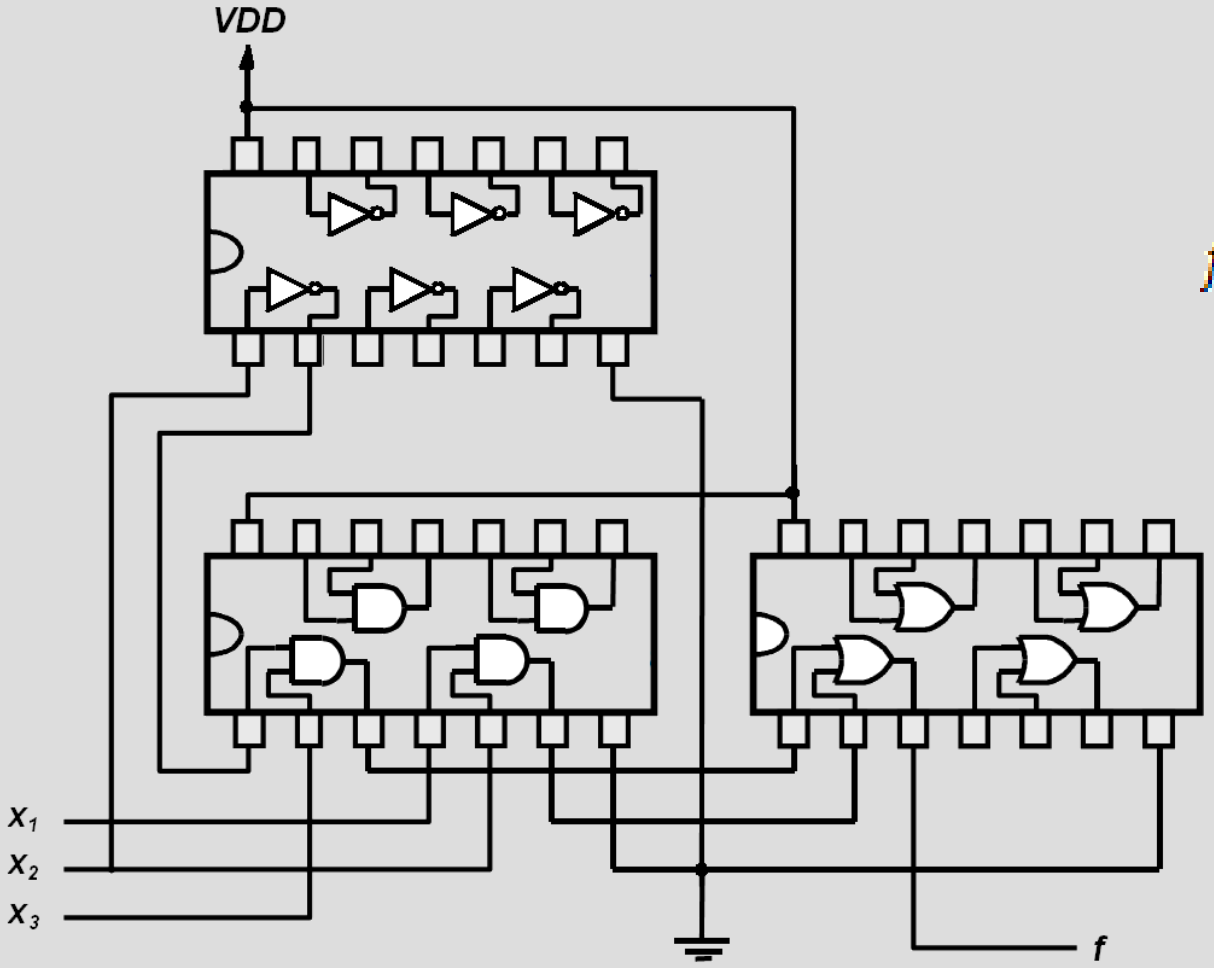
Estrutura Eletrônica

- Tipos:
  - Existem diversos tipos de *PLDs*, e é muito importante ter-se em mente suas vantagens e desvantagens de uso;
  - Estes tipos são classificados principalmente por sua tecnologia de implementação, mas também por características elétricas definidas pela tecnologia de implementação;
  - São os tipos de *PLDs*:

- *PLA;*
- *GAL;*
- *SPLD;*
- *EPLD;*
- *CPLD;*
- *FPGA.*

Conceitos Básicos

- Implementação de Sistemas Digitais:



$$f = x_1 x_2 + \overline{x_2} x_3$$

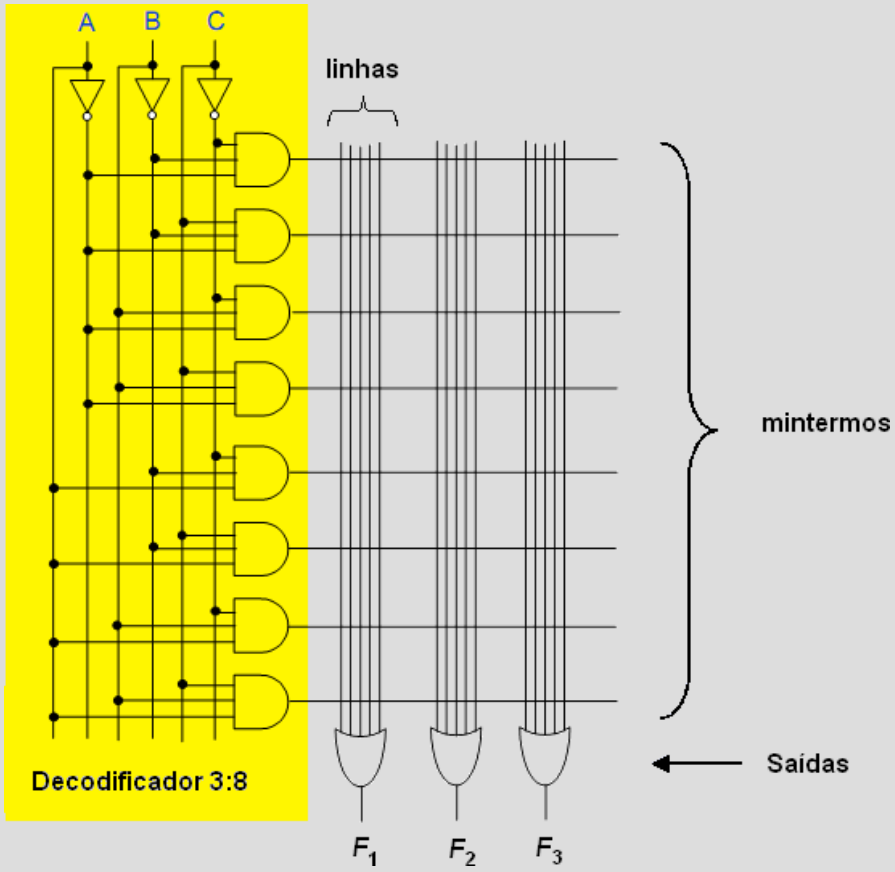
Conceitos Básicos

- Implementação de Sistemas Digitais:
  - Uma maneira mais simples de se apresentar este sistema é através de sua tabela verdade:

x_1	x_2	x_3	f
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

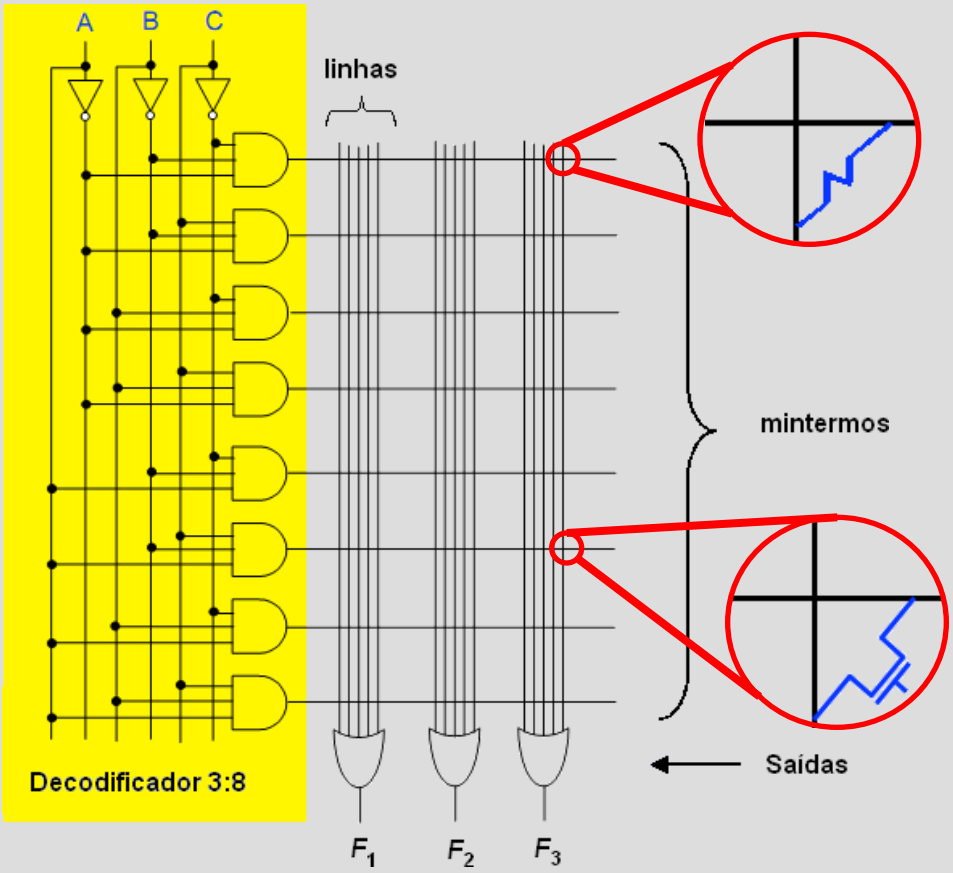
Conceitos Básicos

- Implementação de Sistemas Digitais:
  - Essa tabela verdade pode ser convertida para circuito lógico através de uma matriz de conexões:



Conceitos Básicos

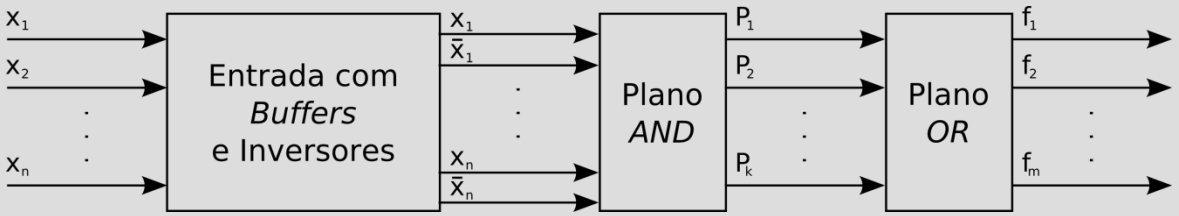
- Implementação de Sistemas Digitais:



- As ligações são feitas por técnicas de gravação de chaves lógicas;
- Estas chaves lógicas podem ser ou fusíveis ou antifusíveis;

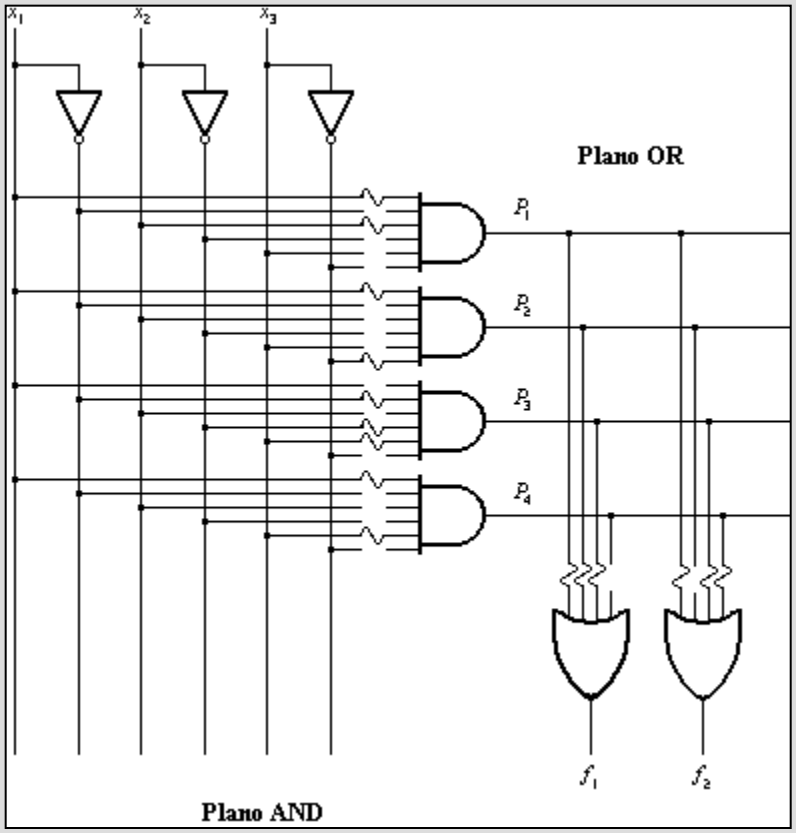
Estrutura Eletrônica

- *PLA*:
  - Sua sigla significa *Programmable Logic Array*;
  - Foi o primeiro *PLD* desenvolvido;
  - Possui um arranjo que é capaz de descrever uma lógica através da soma de produtos; de produtos;
  - São muito eficientes em termos de área requerida para sua implementação;
  - Só podiam ser gravadas uma única vez (funcionavam como uma *ROM*).



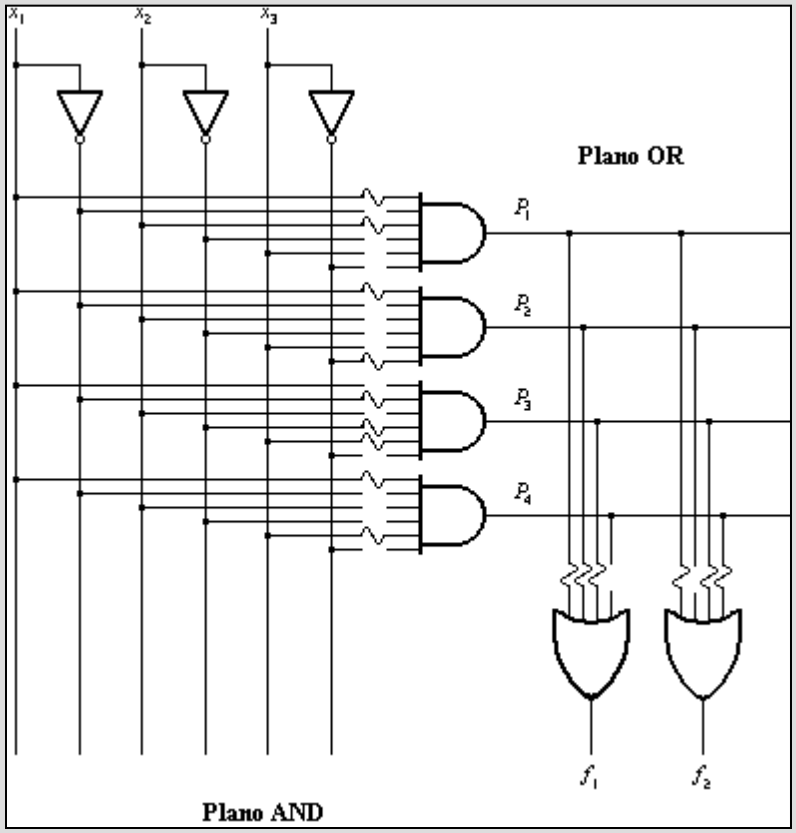
Estrutura Eletrônica

- *PLA*:
  - Sua estrutura interna é a seguinte:



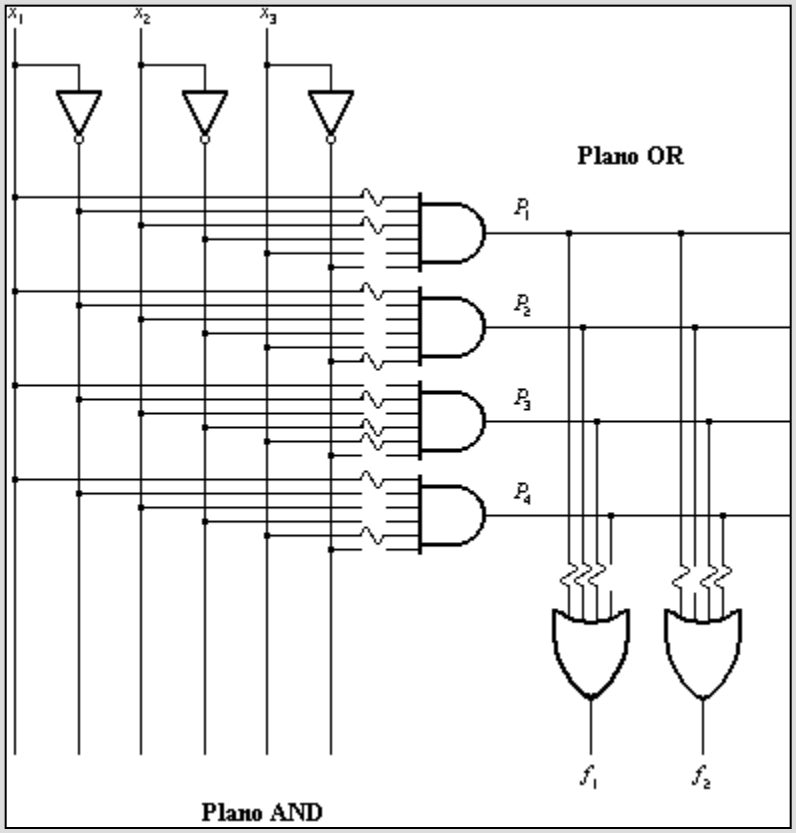
Estrutura Eletrônica

- *PLA*:
  - Sua estrutura interna simplificada é a seguinte:



Estrutura Eletrônica

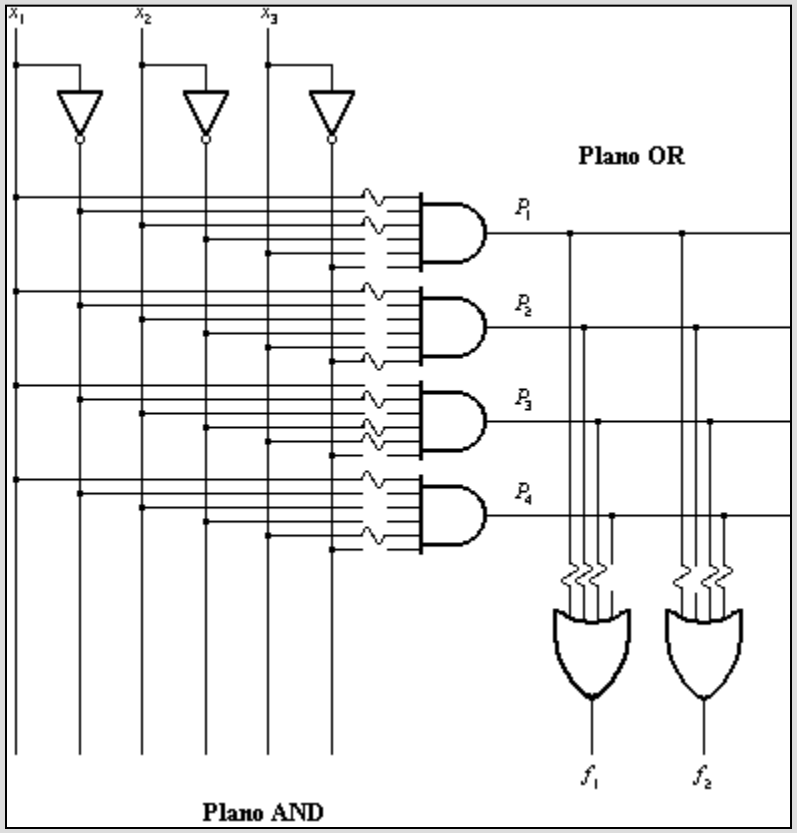
- *PLA*:
  - Sua estrutura interna simplificada é a seguinte:



$$P_1 = x_1 \cdot x_2$$

Estrutura Eletrônica

- *PLA*:
  - Sua estrutura interna simplificada é a seguinte:

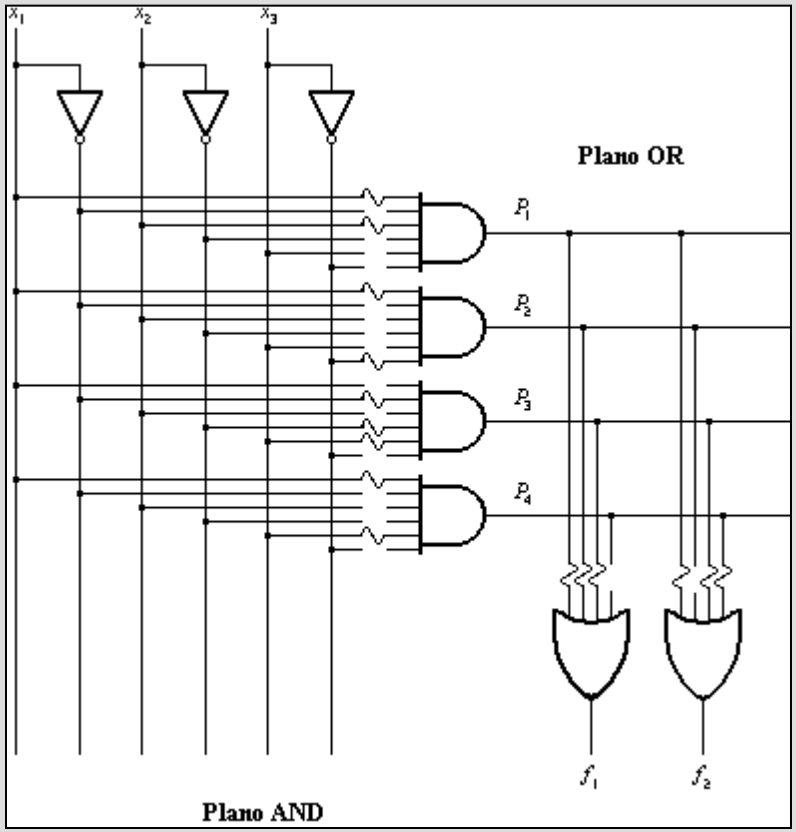


$$P_1 = x_1 \cdot x_2$$

$$P_2 = x_1 \cdot \bar{x}_3$$

Estrutura Eletrônica

- *PLA*:
  - Sua estrutura interna simplificada é a seguinte:



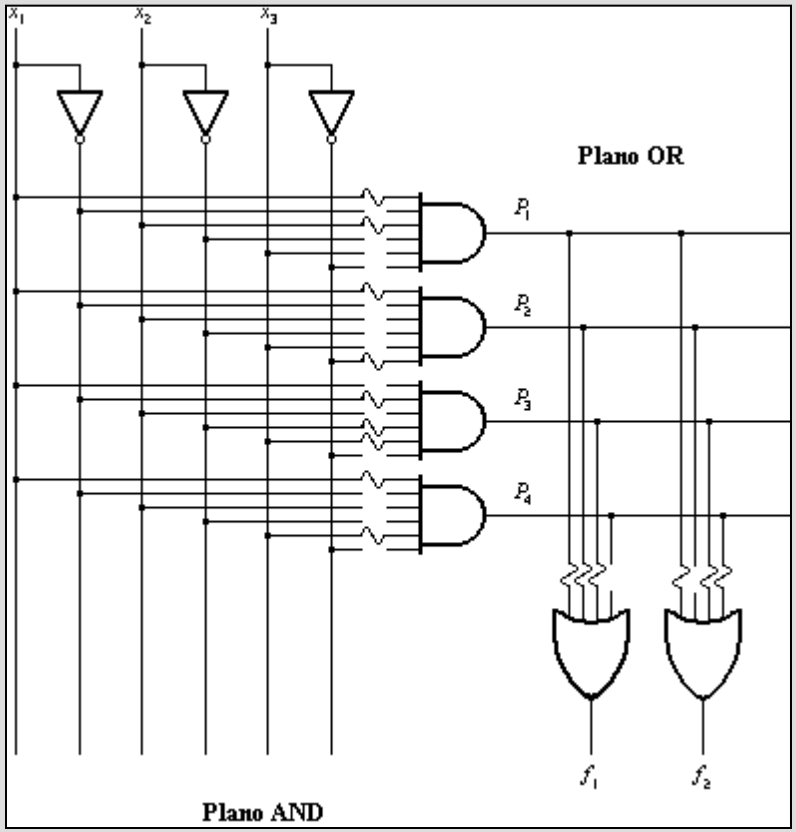
$$P_1 = x_1 \cdot x_2$$

$$P_2 = x_1 \cdot \bar{x}_3$$

$$P_3 = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

Estrutura Eletrônica

- *PLA*:
  - Sua estrutura interna simplificada é a seguinte:



$$P_1 = x_1 \cdot x_2$$

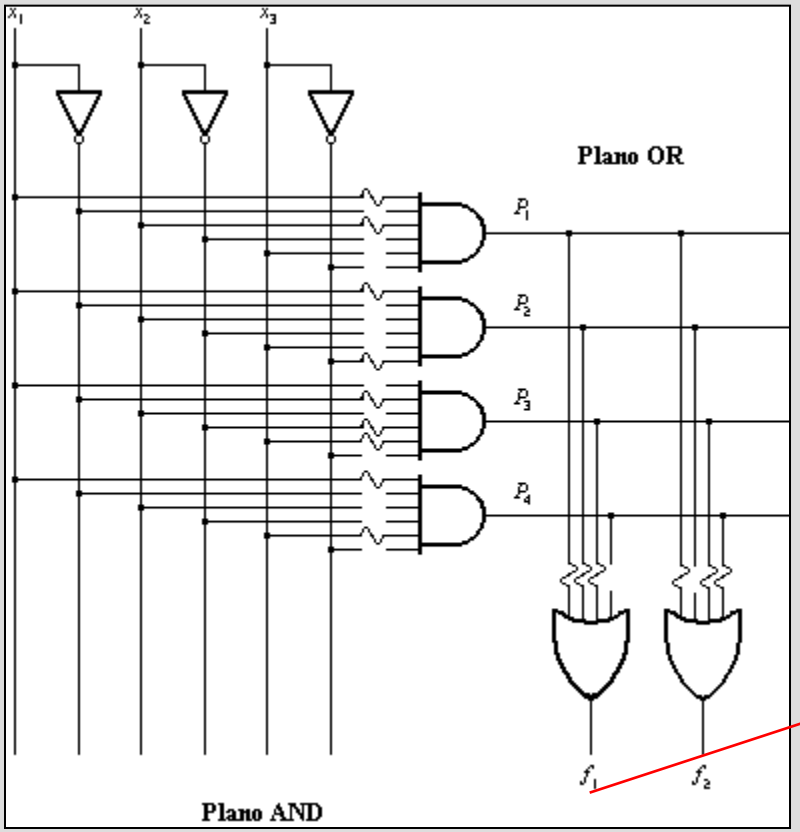
$$P_2 = x_1 \cdot \bar{x}_3$$

$$P_3 = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

$$P_4 = x_1 \cdot x_3$$

Estrutura Eletrônica

- *PLA*:
  - Sua estrutura interna simplificada é a seguinte:



$$P_1 = x_1 \cdot x_2$$

$$P_2 = x_1 \cdot \bar{x}_3$$

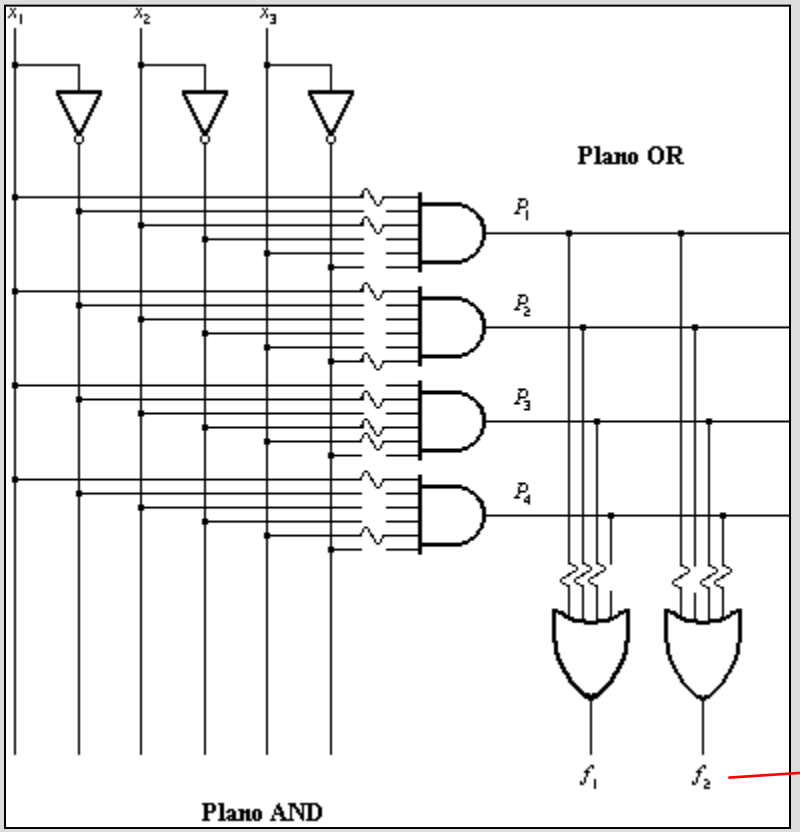
$$P_3 = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

$$P_4 = x_1 \cdot x_3$$

$$f_1 = x_1 \cdot x_2 + x_1 \cdot \bar{x}_3 + \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

Estrutura Eletrônica

- *PLA*:
  - Sua estrutura interna simplificada é a seguinte:



$$P_1 = x_1 \cdot x_2$$

$$P_2 = x_1 \cdot \bar{x}_3$$

$$P_3 = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

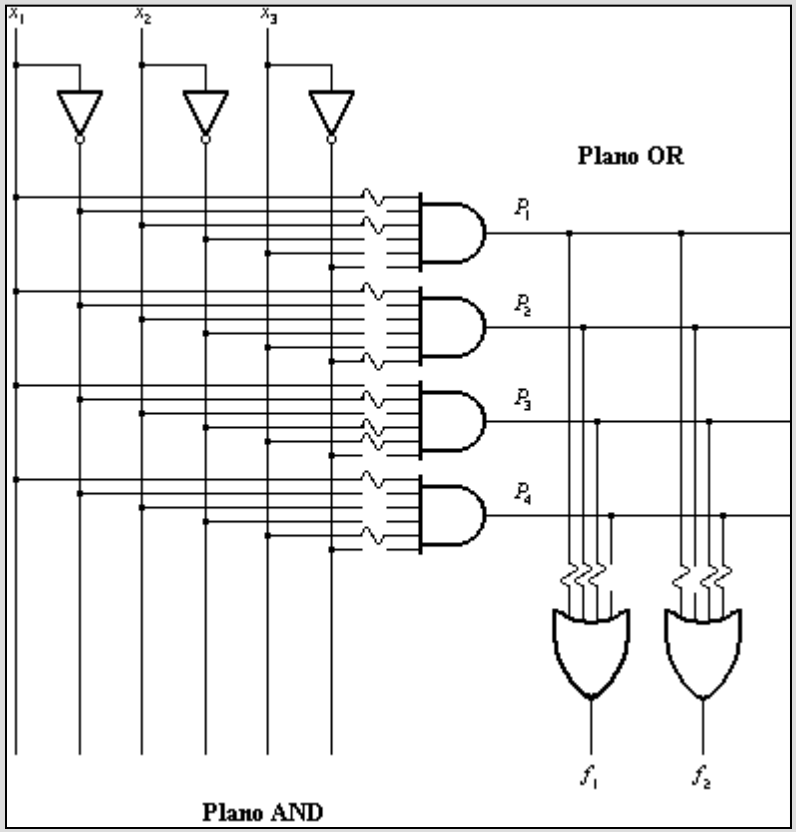
$$P_4 = x_1 \cdot x_3$$

$$f_1 = x_1 \cdot x_2 + x_1 \cdot \bar{x}_3 + \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

$$f_2 = x_1 \cdot x_2 + \bar{x}_1 \cdot \bar{x}_2 \cdot x_3 + x_1 \cdot x_3$$

Estrutura Eletrônica

- *PLA*:
  - Sua estrutura interna simplificada é a seguinte:



$$P_1 = x_1 \cdot x_2$$

$$P_2 = x_1 \cdot \bar{x}_3$$

$$P_3 = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

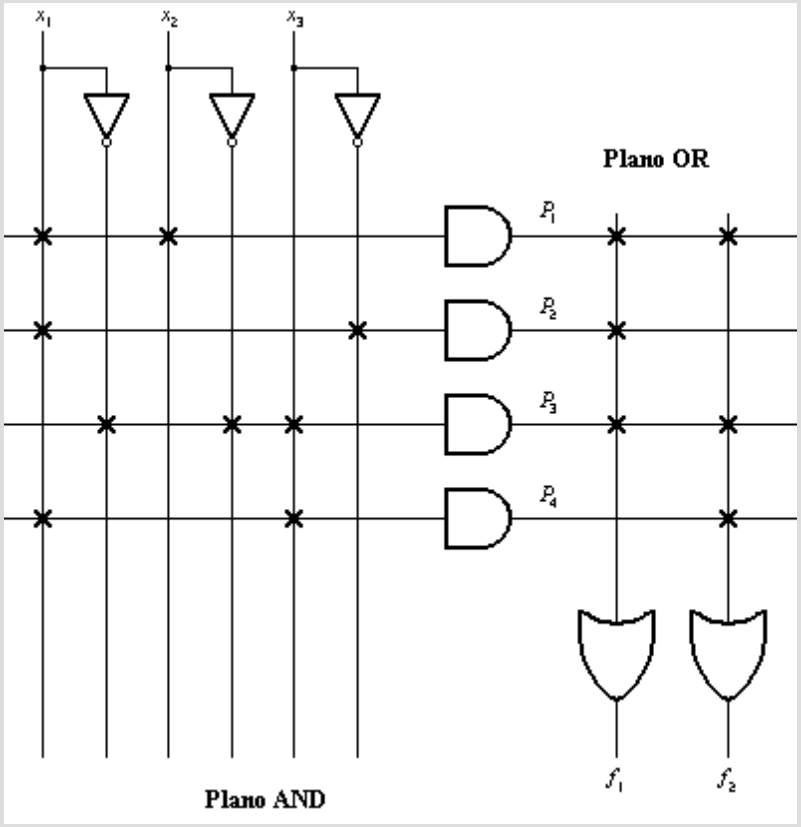
$$P_4 = x_1 \cdot x_3$$

$$f_1 = x_1 \cdot x_2 + x_1 \cdot \bar{x}_3 + \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

$$f_2 = x_1 \cdot x_2 + \bar{x}_1 \cdot \bar{x}_2 \cdot x_3 + x_1 \cdot x_3$$

Estrutura Eletrônica

- *PLA*:
  - Sua estrutura interna simplificada é a seguinte:



$$P_1 = x_1 \cdot x_2$$

$$P_2 = x_1 \cdot \bar{x}_3$$

$$P_3 = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

$$P_4 = x_1 \cdot x_3$$

$$f_1 = x_1 \cdot x_2 + x_1 \cdot \bar{x}_3 + \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

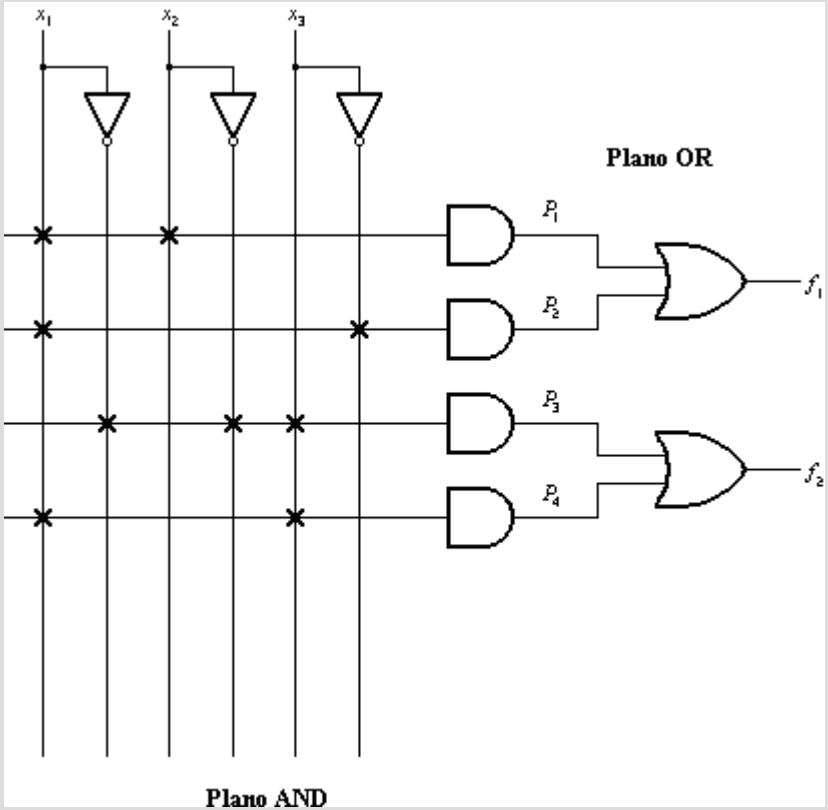
$$f_2 = x_1 \cdot x_2 + \bar{x}_1 \cdot \bar{x}_2 \cdot x_3 + x_1 \cdot x_3$$

Estrutura Eletrônica

- *PAL*:
  - Sua sigla significa *Programmable Array Logic*;
  - Em uma *PAL* somente o plano *AND* é programável, sendo o plano *OR* fixo;
  - Isso reduz a implementação de um elemento crítico para a construção de um *PLD* que são as chaves programáveis;
  - Além de ser um de difícil implementação com precisão necessária, as chaves programáveis reduzem a performance (velocidade e potência) dos circuitos digitais;
  - Apresenta uma menor flexibilidade de implementação em relação as *PLAs*;
  - Para compensar são oferecidas em uma ampla gama de dimensões em termos de números de entrada e de saída;
  - Assim como as *PLAs* só podem ser programadas uma única vez.

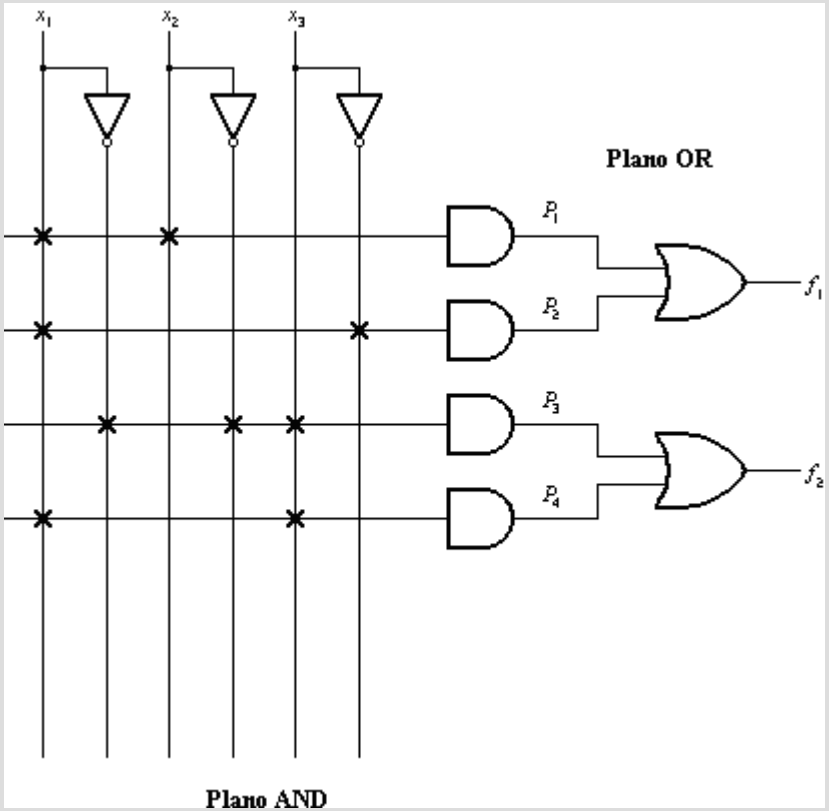
Estrutura Eletrônica

- *PAL*:
  - Sua estrutura interna simplificada é a seguinte:



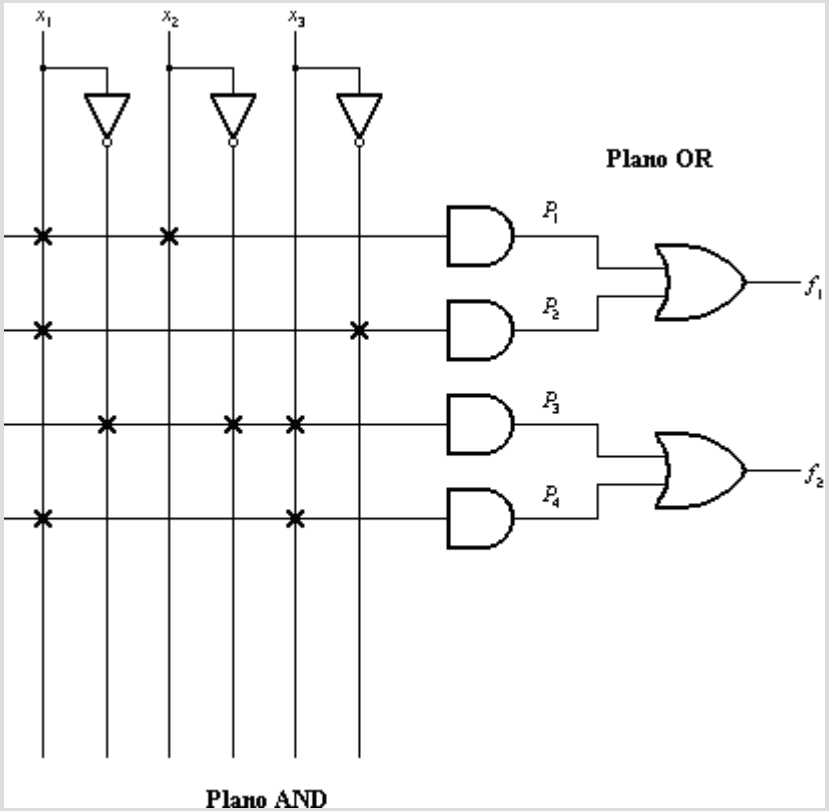
Estrutura Eletrônica

- *PAL*:
  - Sua estrutura interna simplificada é a seguinte:



Estrutura Eletrônica

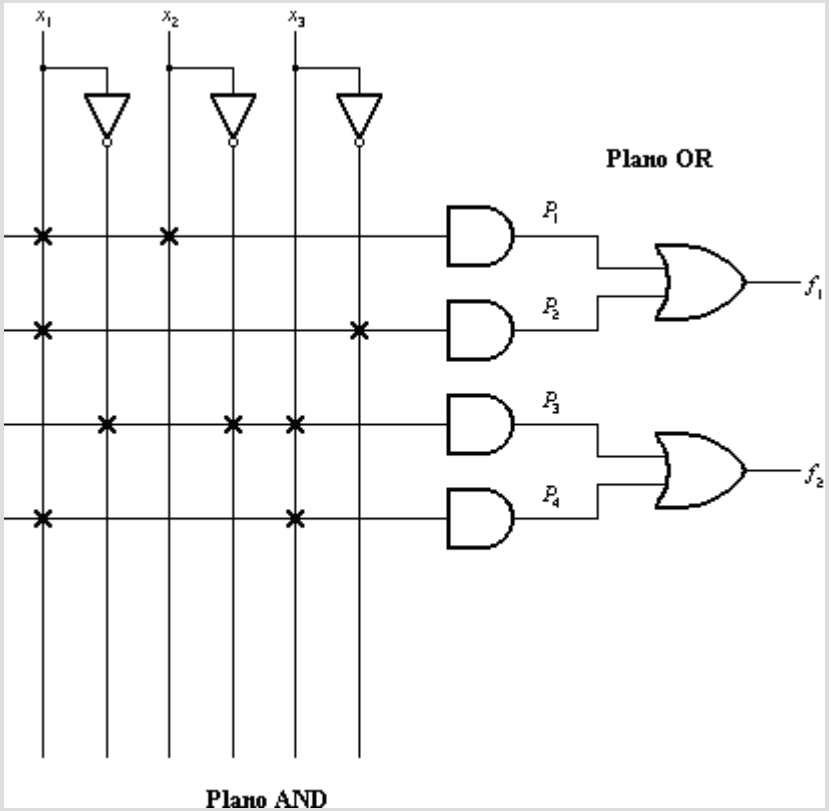
- *PAL*:
  - Sua estrutura interna simplificada é a seguinte:



$$f_1 = x_1 \cdot x_2 + x_1 \cdot \bar{x}_3$$

Estrutura Eletrônica

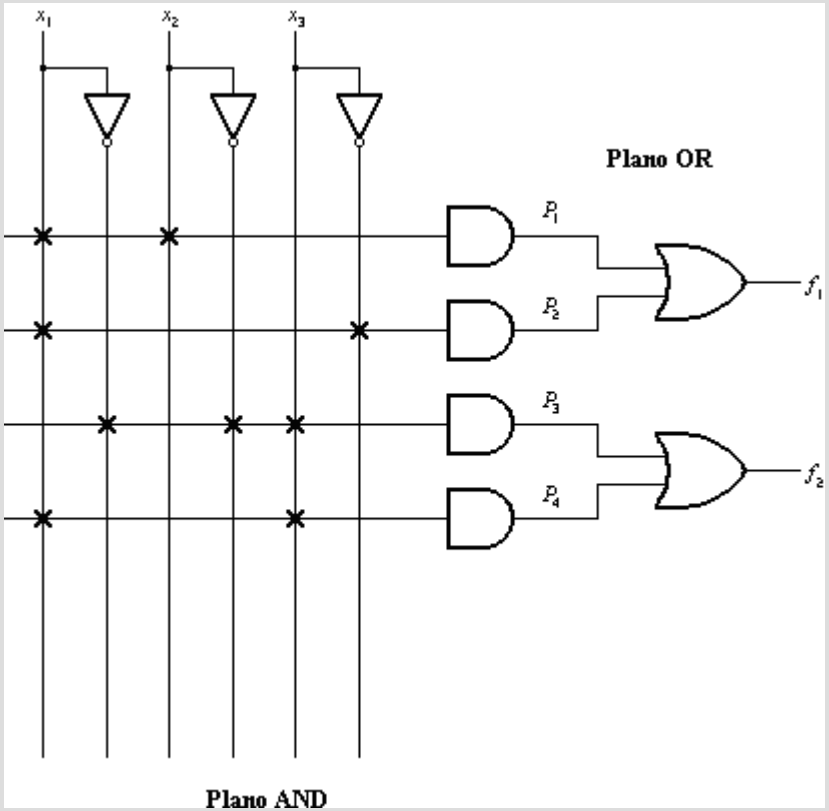
- *PAL*:
  - Sua estrutura interna simplificada é a seguinte:



$$f_1 = x_1 \cdot x_2 + x_1 \cdot \bar{x}_3$$

Estrutura Eletrônica

- *PAL*:
  - Sua estrutura interna simplificada é a seguinte:

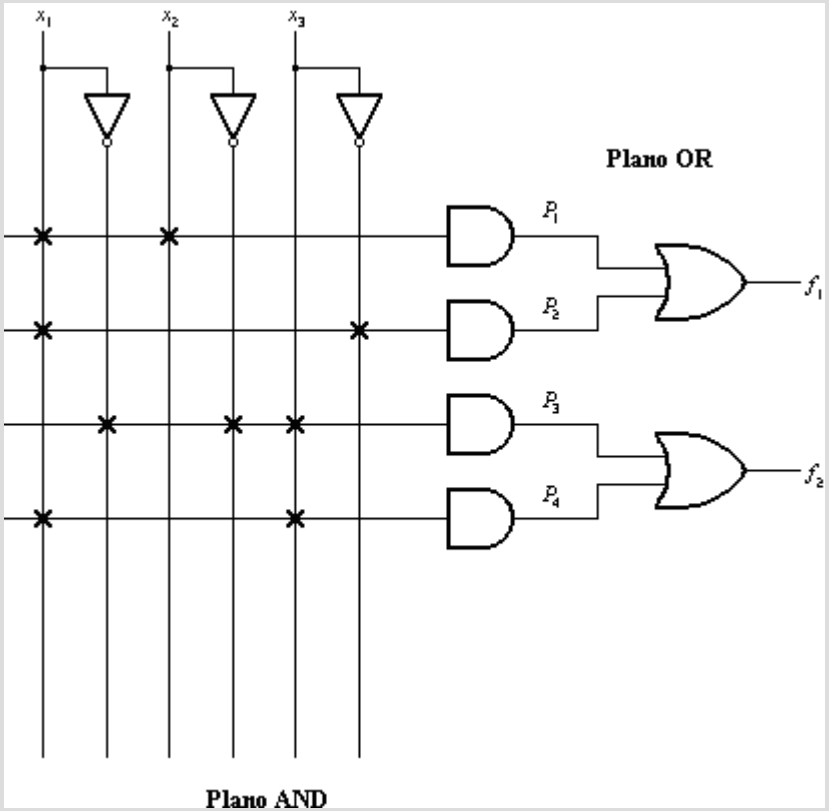


$$f_1 = x_1 \cdot x_2 + x_1 \cdot \bar{x}_3$$

$$f_2 = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3 + x_1 \cdot x_3$$

Estrutura Eletrônica

- *PAL*:
  - Sua estrutura interna simplificada é a seguinte:

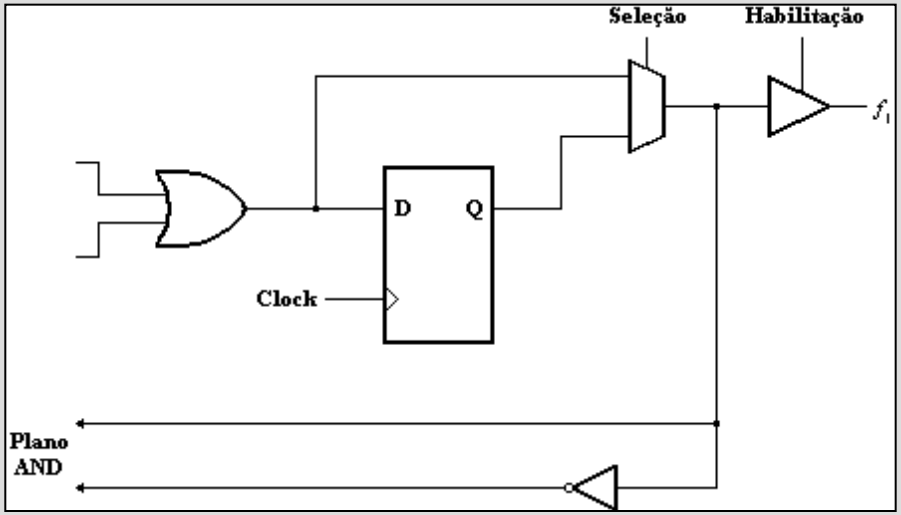


$$f_1 = x_1 \cdot x_2 + x_1 \cdot \bar{x}_3$$

$$f_2 = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3 + x_1 \cdot x_3$$

Estrutura Eletrônica

- *PAL*:
  - Para tornar a arquitetura de uma PAL ainda mais flexível, pode-se adicionar às saídas desta um circuito lógico extra;



  - A este conjunto (*PAL* + Circuito lógico extra) dá-se o nome de **macrocélula**.

Estrutura Eletrônica

- *PAL*:
  - O flip-flop cria a possibilidade de armazenar o valor produzido para a saída do Plano *OR* (elemento memória);
  - Em sua saída (do Flip-Flop) tem-se um multiplexador que permite
  - selecionar entre o sinal com (síncrono) ou sem registro (assíncrono);
  - Na saída da macrocélula tem-se ainda um *tri-state* buffer, que possibilita colocar esta em alta-impedância (possibilitando dois sentidos e três implementações de dados *in*, *out* ou *inout*);
  - O sinal de saída (antes do *tri-state buffer*) é realimentado para o Plano *AND*;
  - Com esta realimentação ampliam-se os horizontes de possibilidades de implementações lógicas com uma macrocélula.

Estrutura Eletrônica

- *GAL*:
  - Sua sigla significa *Generic Array Logic*;
  - Criada em 1985 pela *Lattice Semiconductors;*
  - Oossui as mesmas propriedades das PALs, porém podem ser apagadas e reprogramadas;
  - As GALs são utilizadas como circuitos básicos de PLDs modernos, tais como os PLDs, por apresentar esta característica.

Estrutura Eletrônica

- *PAL* x *GAL*:
  - Atualmente quando refere-se a *PAL* provavelmente esta-se falando de um elemento *GAL*;
  - Isto ocorre porque como *GAL* era o nome dado por um fabricante a uma arquitetura *PAL*, apenas com uma técnica diferente de gravação;
  - A arquitetura de uma *GAL* é uma *PAL*;
  - Portanto, tanto no meio industrial como no acadêmico as primeiras *PALs* somente graváveis, como as *GALs* (por vezes chamadas de *PALs* regraváveis) são chamadas de *PALs*;
  - Nos *PLDs*, onde são implementadas *PALs* regraváveis, estes elementos são chamados apenas de *PALs*.
  - Não é errado fazer essa simplificação de nomes, pois a arquitetura é realmente a mesma.

Estrutura Eletrônica

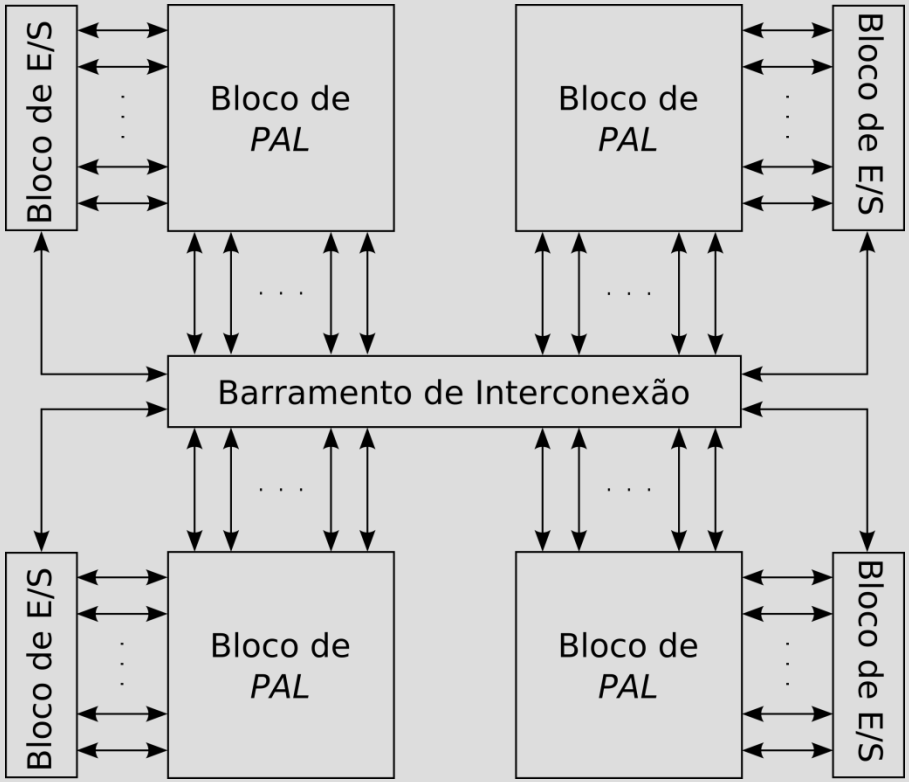
- *SPLD*:
  - Primeira Geração:
    - Utilizavam como arquitetura básica *PLAs*;
    - Proviam alta densidade de lógica disponível (para a época);
    - Alta flexibilidade, pois podia-se programar tanto no plano *AND* como no plano *OR*.
  - Segunda Geração:
    - Utilizavam como arquitetura básica *PALs*;
    - Proviam uma densidade de lógica média (para a época);
    - Menor flexibilidade, porém mais rápidos que os *SPLDs* de primeira geração, por terem uma menor quantidade de chaves lógicas (que reduz a performance).

Estrutura Eletrônica

- *CPLD*:
  - Sua sigla significa *Complex Programmable Logic Device*;
  - É um arranjo de *PALs*, macrocélulas interconectadas por uma linha central dedicada;
  - Possui aspectos de *timing* simples e determinístico;
  - Possui facilidade de roteamento (por ter esta linha dedicada a
  - internconexão);
  - Tem uma ampla e rápida rede de chaves lógicas;
  - Atualmente têm-se várias ferramentas de *CAD* para auxílio no projeto para estes dispositivos;
  - Sua gravação é dada com a tecnologia *E2PROM* ou *Flash*.

Estrutura Eletrônica

- *CPLD*:
  - Arquitetura de um *CPLD*:



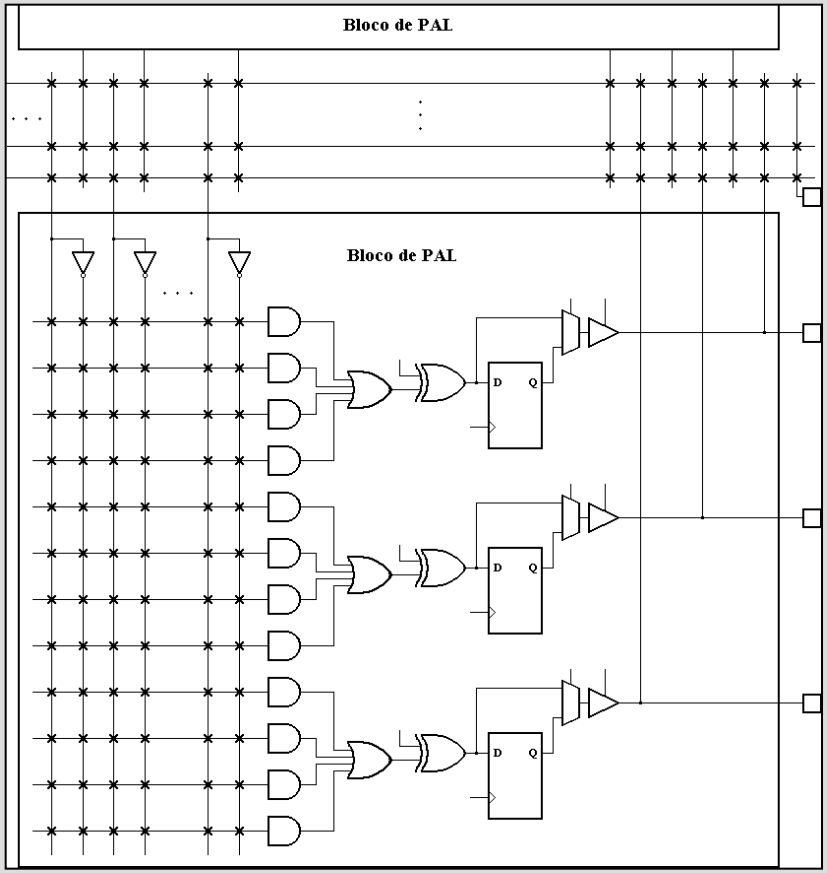
- *CPLDs* comerciais possuem em cada bloco de *PAL* normalmente 16 macrocélulas;
- Cada macrocélula é composta por entre 5 ou 20 entradas de portas *OR*;
- As saídas das portas *OR* são ligadas aos flip-flops por meio de uma porta *XOR*;
- O tri-state buffer liga a saída da lógica do bloco de *PAL* ao pino do *CPLD*;
- As ligações entre os blocos de *PAL* também são reprogramáveis.

UNISINOS
Somos infinitas possibilidades

LaPSE
Laboratório de Prototipação Digital e Sistemas Embarcados

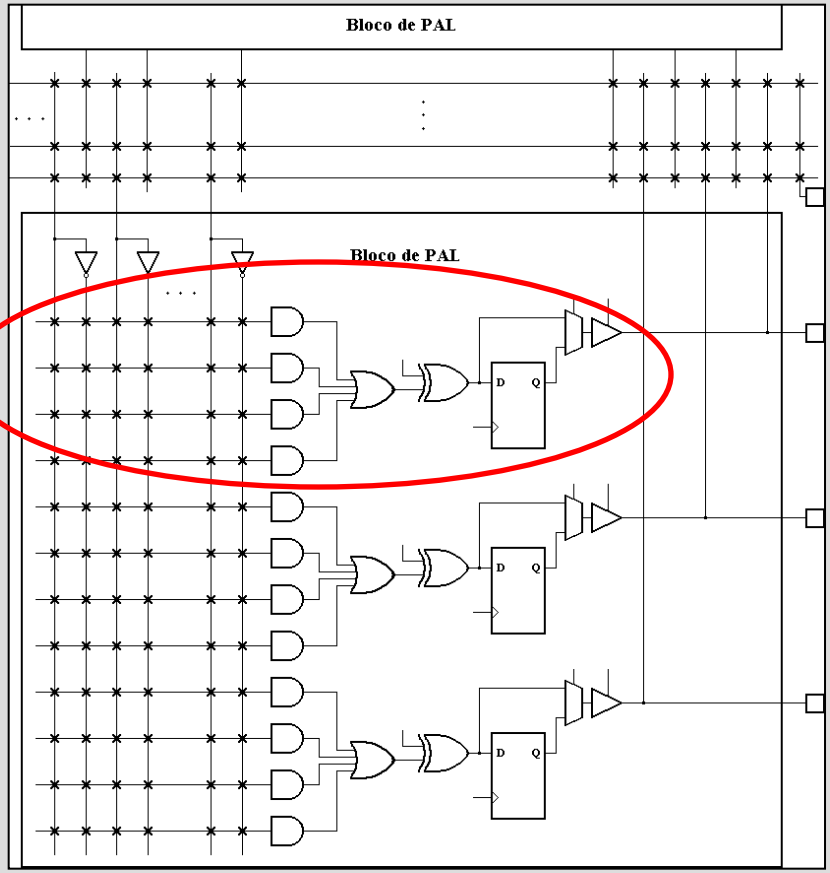
Estrutura Eletrônica

- *CPLD*:
  - Apresentação da arquitetura de um bloco de *PAL*:



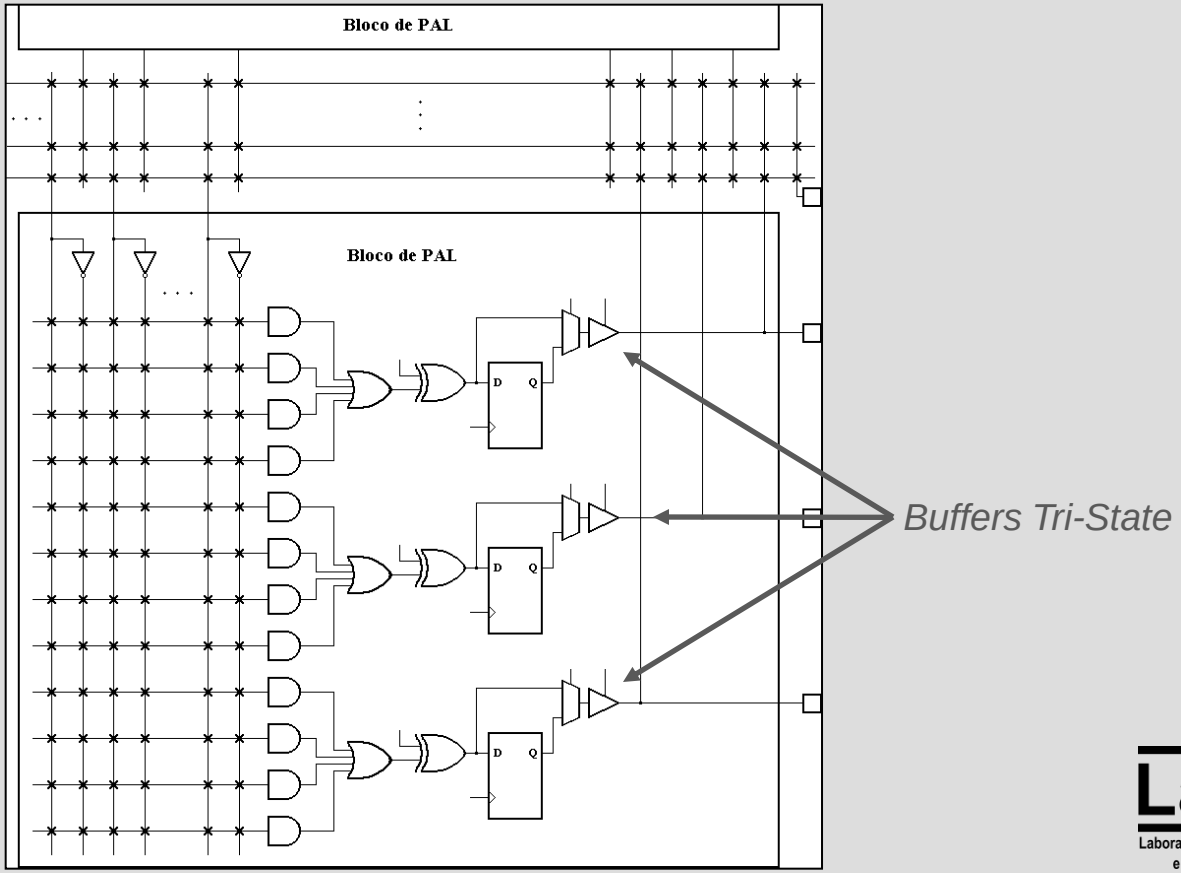
Estrutura Eletrônica

- *CPLD*:
  - Apresentação da arquitetura de um bloco de *PAL*:



Estrutura Eletrônica

- *CPLD*:
  - Apresentação da arquitetura de um bloco de *PAL*:



Estrutura Eletrônica

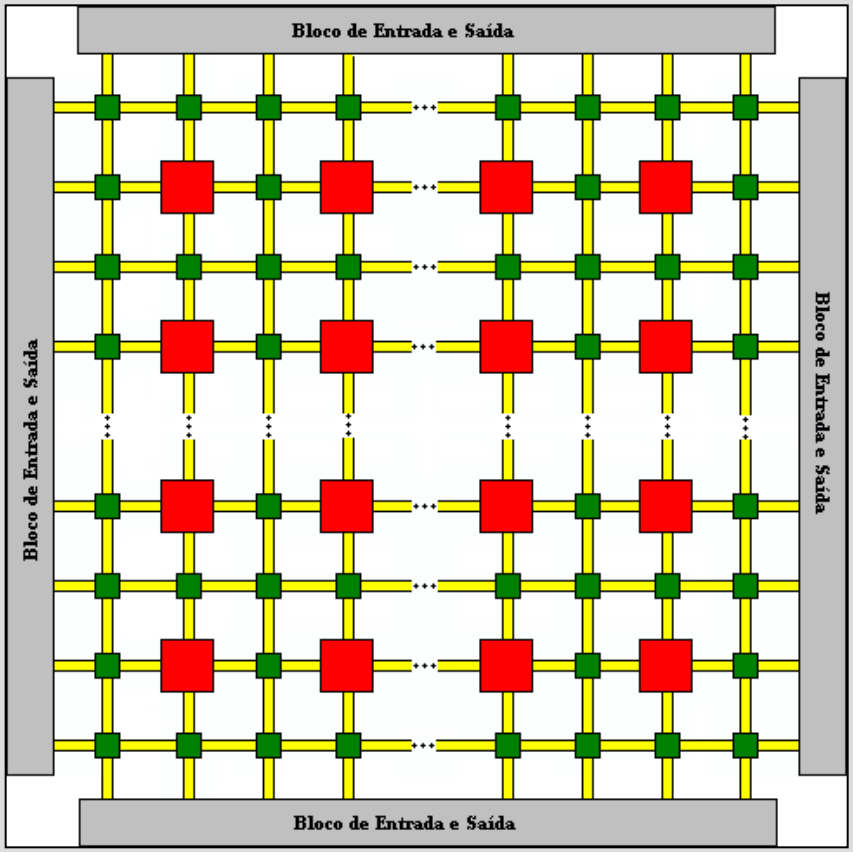
- *CPLD*:
  - Vantagens do uso de *CPLDs*:
    - Fácil desenvolvimento;
    - Baixo custo de desenvolvimento;
    - Retorno financeiro mais rápido;
    - Redução da área de PCI (Placa de Circuito Impresso);
    - Possibilidade de agregar alto índice de propriedade intelectual (aumetando assim o valor agregado do produto final.

Estrutura Eletrônica

- *FPGAs*:
  - Sua sigla significa *Field Programmable Gate Array*;
  - Não são baseados em planos *AND* e *OR*, como os demais *PLDs*;
  - Ele é baseado em blocos lógicos programáveis chamados de *look-up tables* (*LUTs*);
  - Possui quatro elementos de arquitetura básicos:
    - *LUTs* (blocos lógicos)
    - Barramentos de interconexão;
    - Chaves programáveis;
    - Blocos de E/S (Entrada/Saída, que ligam a lógica interna aos pinos do *FGPA*).

Estrutura Eletrônica

- *FPGAs*:
  - Arquitetura de um *FPGA*:



Estrutura Eletrônica

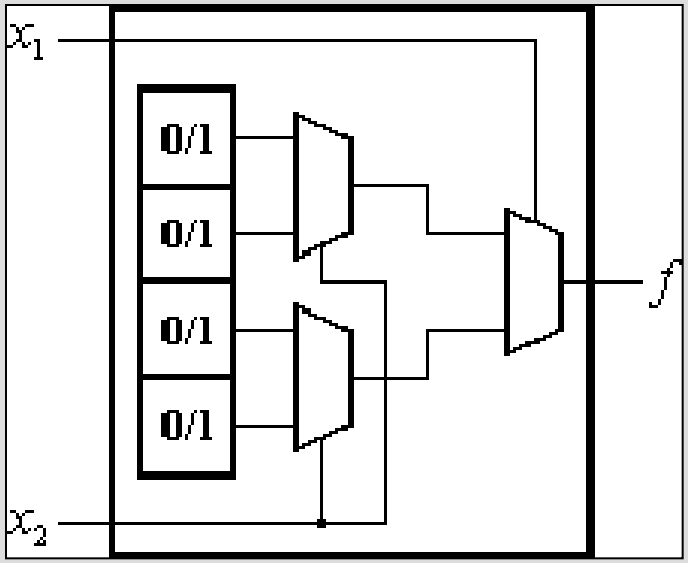
- *FPGAs*:
  - São divididos em dois tipo, definidos pelo seu método de programação:
    - Reprogramáveis:
      - Baseados em tecnologia *SRAM*;
      - Trazem a flexibilidade para o desenvolvimento e *upgrade* de projetos;
      - Mais comuns e difundidos do meio, tanto que ao se falar em *FPGAs* logo se pensa em componentes reprogramáveis;
      - Utilizam a construção baseadas em interconexão de *LUTs*.
    - *One-Time Programmable* (*OTP*):
      - baseado um anti-fusívieis;
      - Uilizados somente em projetos maduros e amplamente testados, ou em aplicações com alto volume, ou ainda que tenham a necessidade de um *boot* rápido do sistema;
      - Utiliza a construção tradicional de portas lógicas, como nas *PLAs* e *PALs*.

Estrutura Eletrônica

- *FPGAs*:
  - *LUT – Lookup Table*:
    - São uma espécie de tabela de consulta rápida;
    - Possuem células de armazenamento de tabelas verdade de circuito lógicos;
    - Com a utilização de *LUTs* pode-se utilizar simplificações *booleanas* para gerar tabelas verdades mais compactas, com menos dispendimento de recursos computacionais;
    - Torna a tradução de uma *HDL* mais fácil para o *hardware* propriamente dito;
    - As *LUTs* podem assumir virtualmente qualquer tamanho, porém atualmente os *FPGAs* são constituídos por *LUTs* de 8 a 64 bits;
    - Normalmente um *FPGA* utiliza somente um tamanho de *LUT* para econimizar na fabricação da máscara de silício da matriz.

Estrutura Eletrônica

- *FPGAs*:
  - *LUT – Lookup Table*:
    - Uma *LUT* possibilita ao projetista, por meio do sintetizador, gravar uma tabela verdade customizada para sua aplicação;
    - Abaixo o exemplo das possibilidades de gravação de uma *LUT* de 2 entradas;

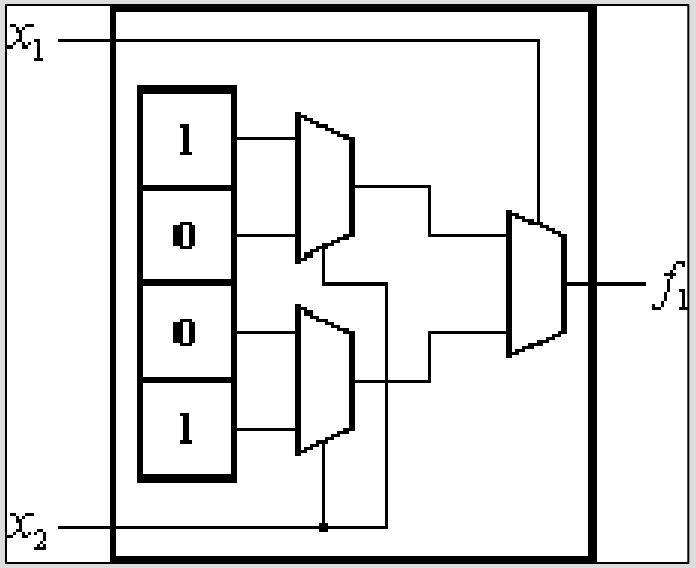


Estrutura Eletrônica

- *FPGAs*:
  - *LUT – Lookup Table*:
    - Exemplo de uma *LUT* de 2 entradas já gravada com sua respectiva tabela verdade:

x_1	x_2	f_1
0	0	1
0	1	0
1	0	0
1	1	1

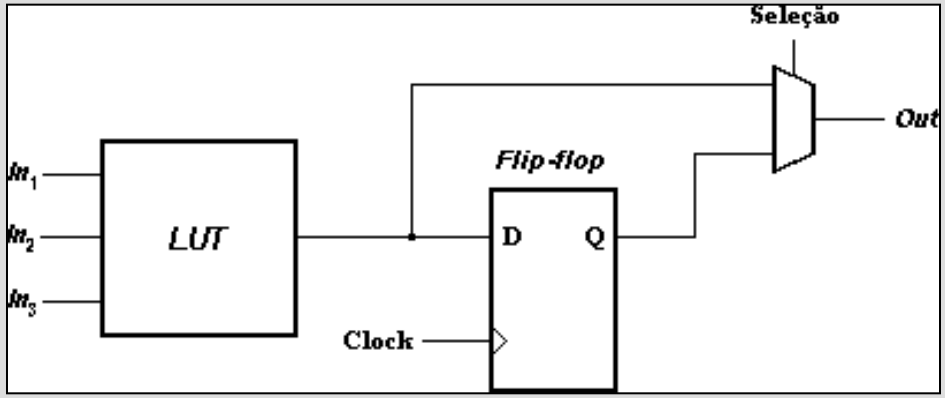
(a) *LUT* - Tabela Verdade



(b) *LUT - Hardware*

Estrutura Eletrônica

- *FPGAs*:
  - *LUT – Lookup Table*:
    - De maneira análoga às *PLAs*, existe um circuito extra colocado na saída das *LUTs* que amplificam o potencial de aplicação das *LUTs*;



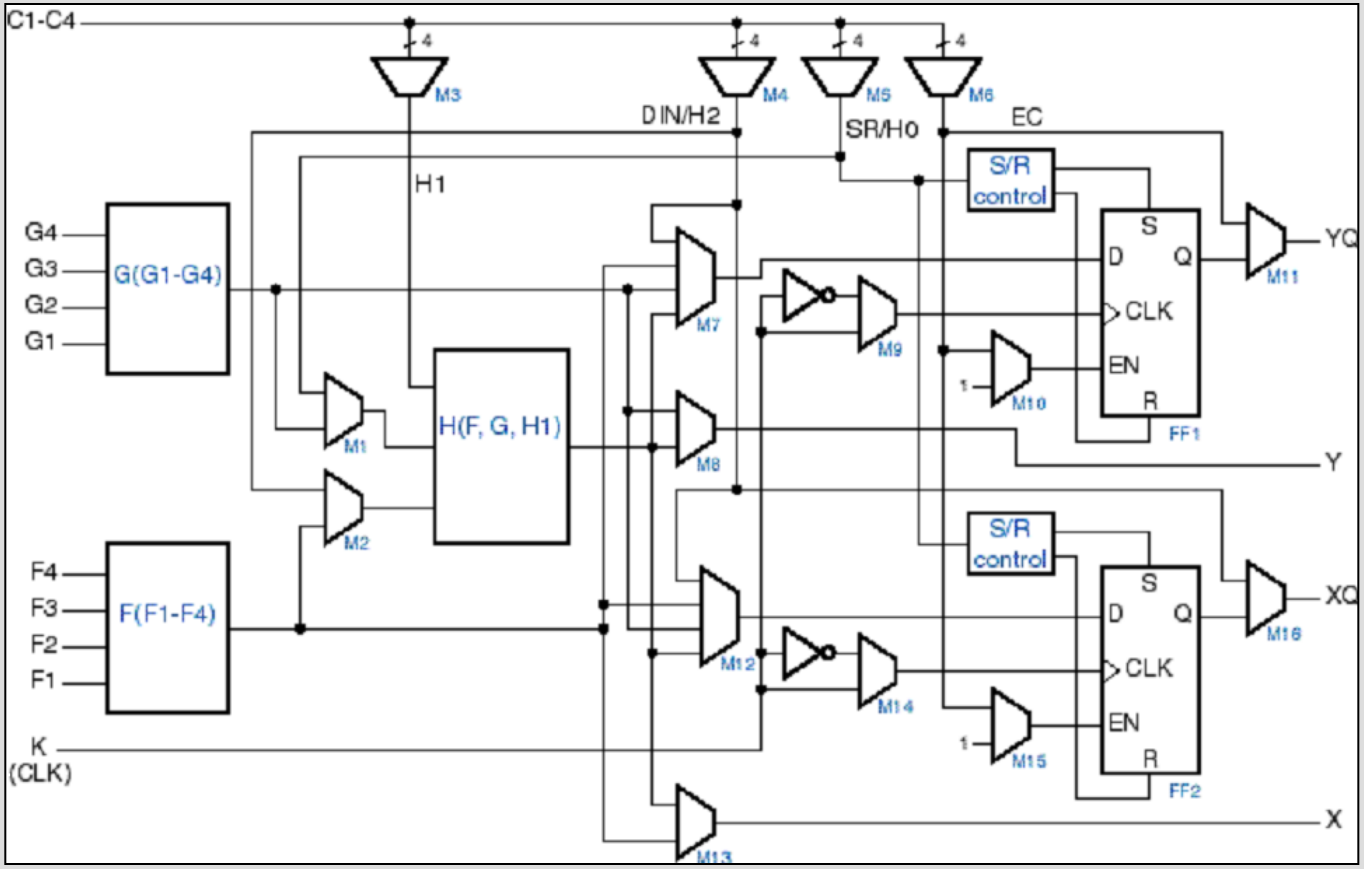
    - Este circuito extra possibilita a escolha entre a utilização da lógica meramente combinacional, ou ainda a utilização do efeito memória pelo uso de um *Flip-Flop*.

Estrutura Eletrônica

- *FPGAs*:
  - *CLB – Configurable Logic Block*:
    - Outro termo bastante utilizado pelos fabricantes de *FPGAs*;
    - Com a evolução da tecnologia para construção de circuitos integrados, e com a intenção de aumentar a flexibilidade dos *FPGAs*, foi criada uma esta estrutura;
    - Ela utiliza-se de *LUTs* para implementar tabelas verdades complexas, *Flip-Flops* para implementações de registros e atrasos de sinais, além de multiplexadores (de alta performance) para seleção dos sinais para a implementação das lógicas desejadas;
    - Os fabricantes como *Altera* e *Xilinx* tratam estes blocos como sendo os blocos básicos, não mais tratando as LUTs como tal;
    - Outros fabricantes como *Lattice*, *Actel* e *Quick Logic*, utilizam nomes próprios para estas estruturas, mas sempre mostras o equivalente em termos de *CLBs*.

Estrutura Eletrônica

- *FPGAs*:
  - CLB – *Configurable Logic Block*:
    - Exemplo de um *CLB* (*FPGA Xilinx*):

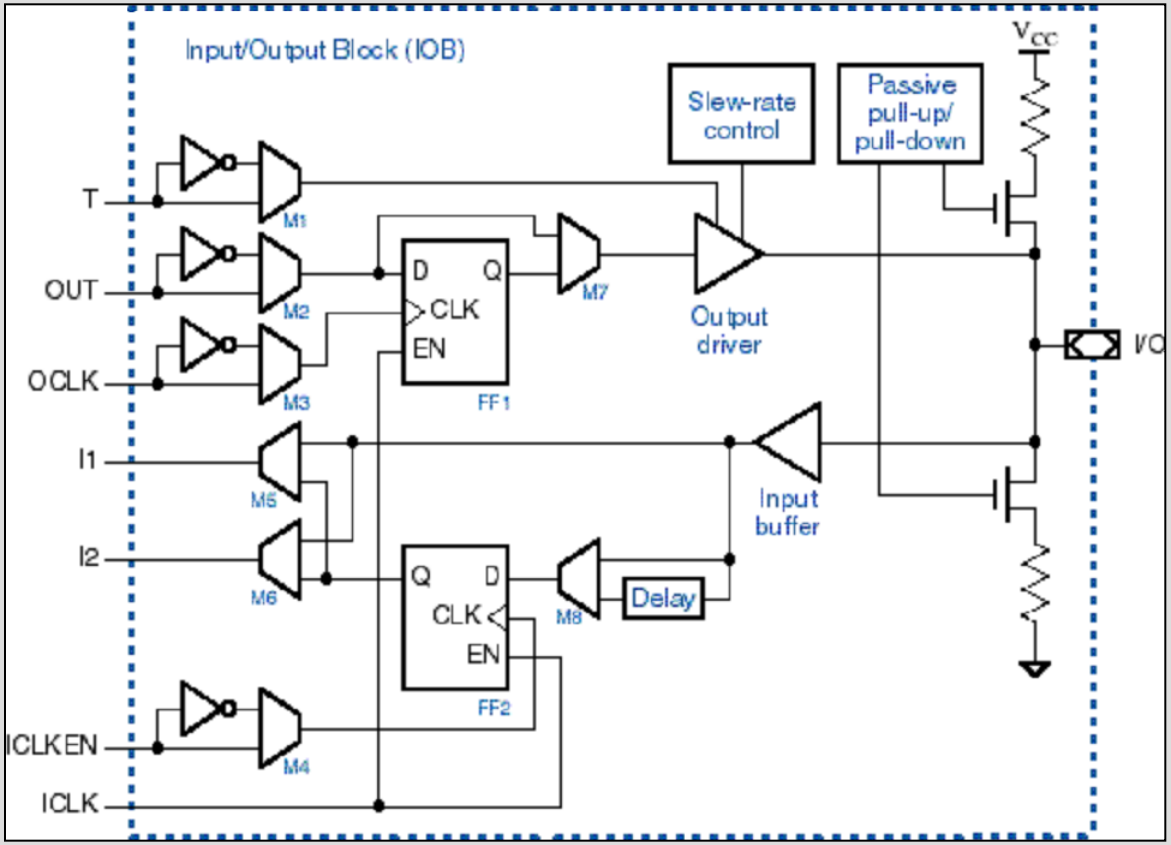


Estrutura Eletrônica

- *FPGAs*:
  - *IOB – Input/Output Block*:
    - São os blocos de lógica responsáveis pela ligação com com os pinos de entrada/saída dos *FPGAs*;
    - Garantem os níveis de tensão e corrente das interfaces;
    - São capazes de implementar interfaces tanto *single ended* como diferenciais;
    - Possibilitam também a implementação de *pull-ups* e *pull-downs* (normalmente fracos) nos sinais de entrada e saída;
    - Podem gerar estado de alta impedância *(tri-state)* para a implementação de canais bidirecionais (portas de *inout*);
    - São dependentes da alimentação do banco a que eles correspondem, para a implementação de cada uma destas características.

Estrutura Eletrônica

- *FPGAs*:
  - *IOB – Input/Output Block*:
    - Exemplo de um *IOB* (*FPGA Xilinx*):



Sinais

- Sinais Digitais:
  - Os sinais digitais assim como os analógicos possuem características a serem levadas em consideração quando utilizados em um projeto;
  - Serão abordadas as principais características a serem avaliadas, analisadas e respeitadas durante o projeto de um sistema digital;
  - Existem características relevantes para todos os sinais digitais e outras específicas para os sinais de controle, ou seja, os sinais de *clock*.

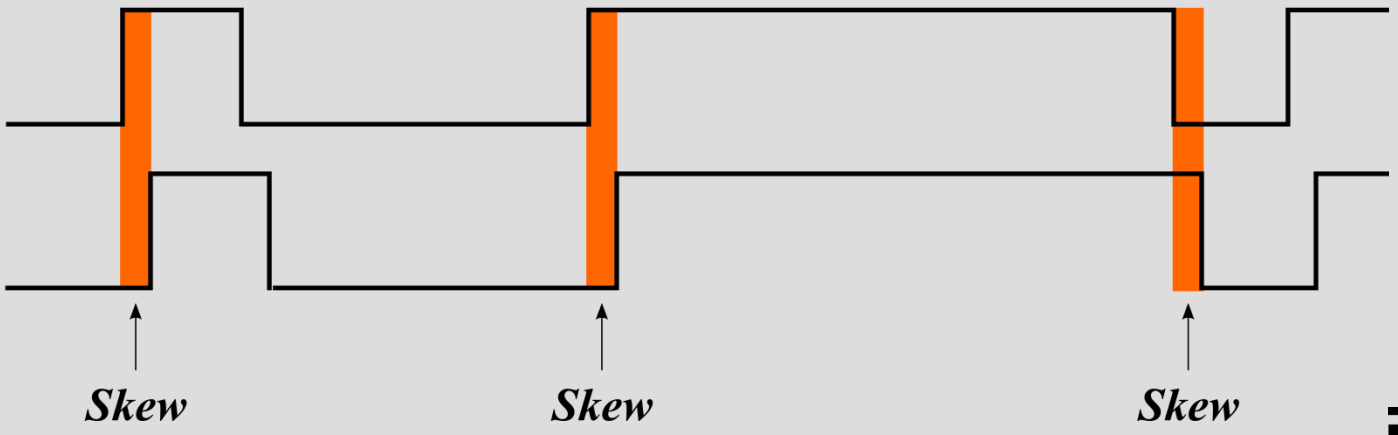
    IMPORTANTE: ter em mente que o sinal digital é um sinal analógico que possui níveis de interpretação pré-definidos!

Sinais

- Sinais Digitais:
  - As principais características a serem conhecidas e reconhecidas em sinais digitais são:
    - *Skew*;
    - *Slew Rate*;
    - *Setup Time* e *Hold Time*;
    - *Fan-in* e *Fan-out*;
    - *Glitch*;
    - *Jitter* e *Wander*.

Sinais

- *Skew*:
  - Escorregamento do sinal digital causado pelo atraso de propagação do mesmo pelo caminho elétrico;
  - Este escorregamento pode ser causado também pelo chamado tempo de porta (*port timing*), que nada mais é do que o atraso que o sinal digital sofre ao ser processado por uma porta lógica, ou por um conjunto destas.

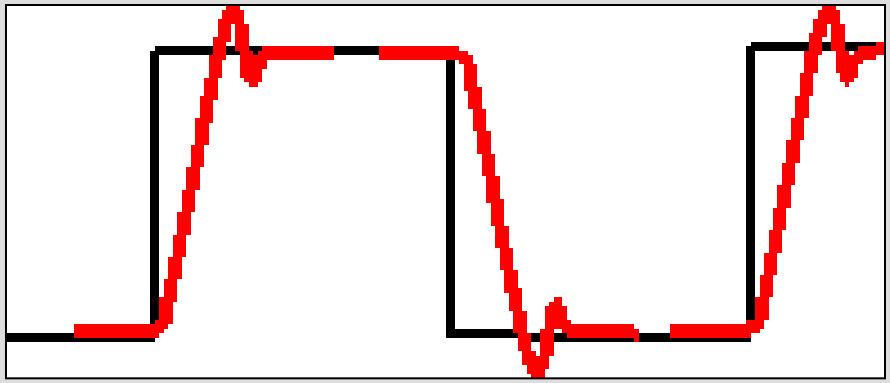


Sinais

- *Slew Rate*:
  - Resposta de uma porta digital ao comutar de nível;
  - É ele que definirá a freqüência máxima de operação de um circuito, ou dispositivo digital;
  - Um *Slew Rate* muito alto torna o dispositivo mais lento;
  - Um *Slew Rate* muito baixo torna o dispositivo mais rápido porém, suscetível a ruídos internos (*bouncing*) e externos (eletromagnéticos ou *crosstalking*);
  - O *Slew Rate* está ligado diretamente aos *Setup* e *Hold Time*, porém ele ajuda a defini-los e não o contrário.

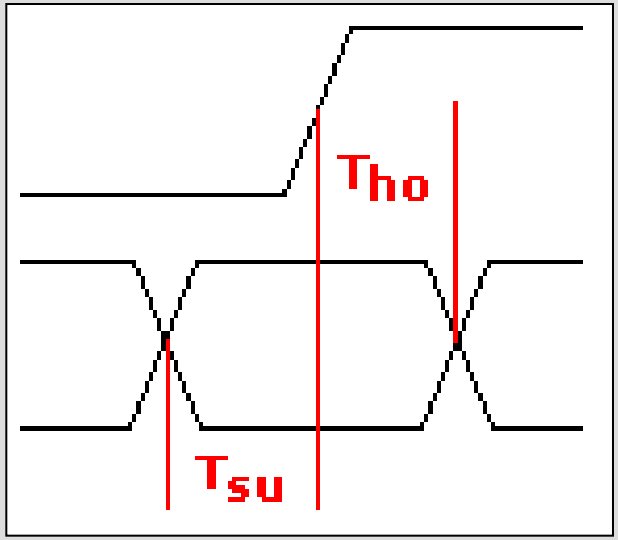
Sinais

- *Slew Rate*:
  - Definido por: $$SR = \left. \frac{dV_{ou}}{dt} \right|_{max}$$



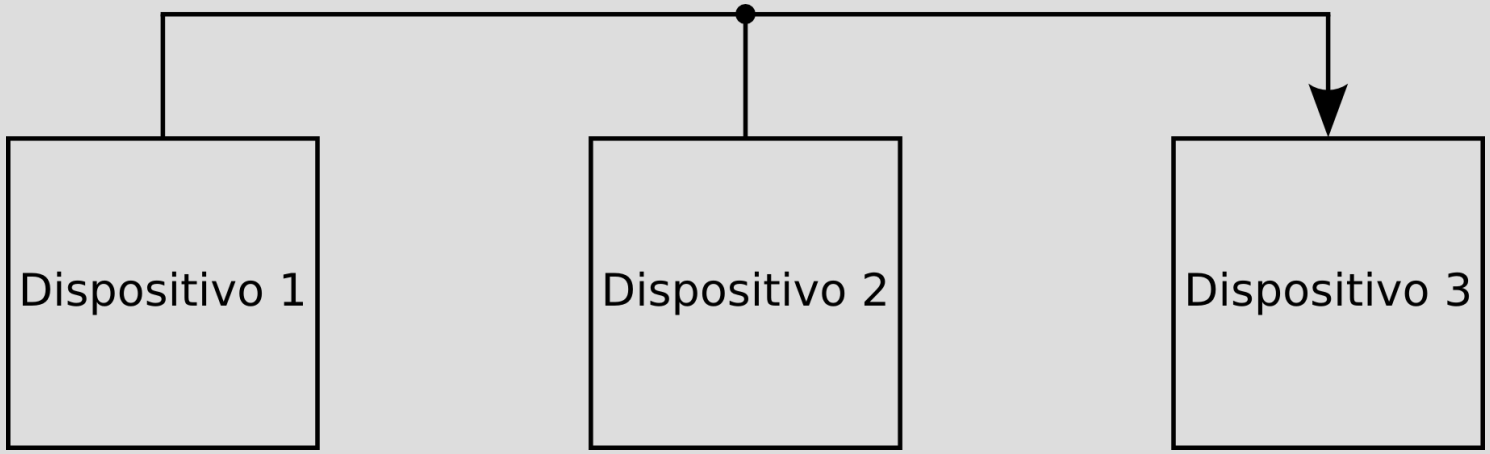
Sinais

- *Setup Time & Hold Time*:
  - ***Tsetup***: ou *Tsu*, é o tempo mínimo para que um sinal (dado) deve ser estabilizado para estar pronto a ser avaliado;
  - ***Thold***: ou *Tho*, é o tempo mínimo para que um sinal (dado) deve ficar estabilizado após ser avaliados.



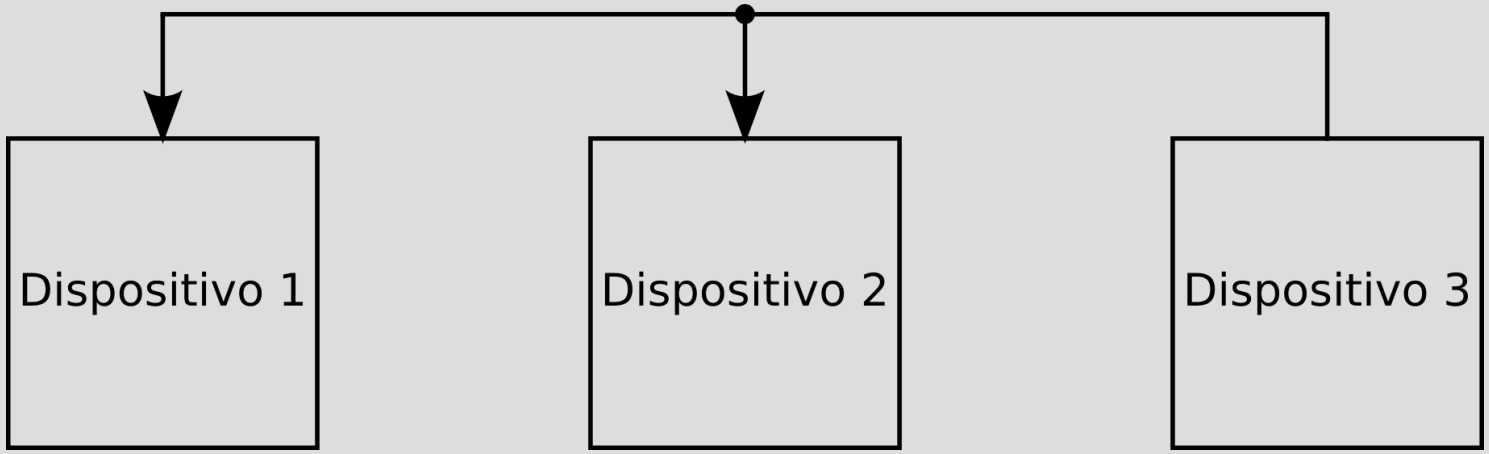
Sinais

- *Fan-in & Fan-out*:
  - ***Fan-in***: capacidade máxima de um elemento digital fazer o *drive* de sinais em sua entrada. Normalmente em portas lógicas o *fan-in* é igual ao número de entradas:



Sinais

- *Fan-in & Fan-out*:
  - ***Fan-out***: capacidade máxima de um elemento digital fazer o *drive* de sinais para sua saída. Cada saída possuirá um *fan-out* específico, pois este é totalmente dependente da construção eletrônica da porta:

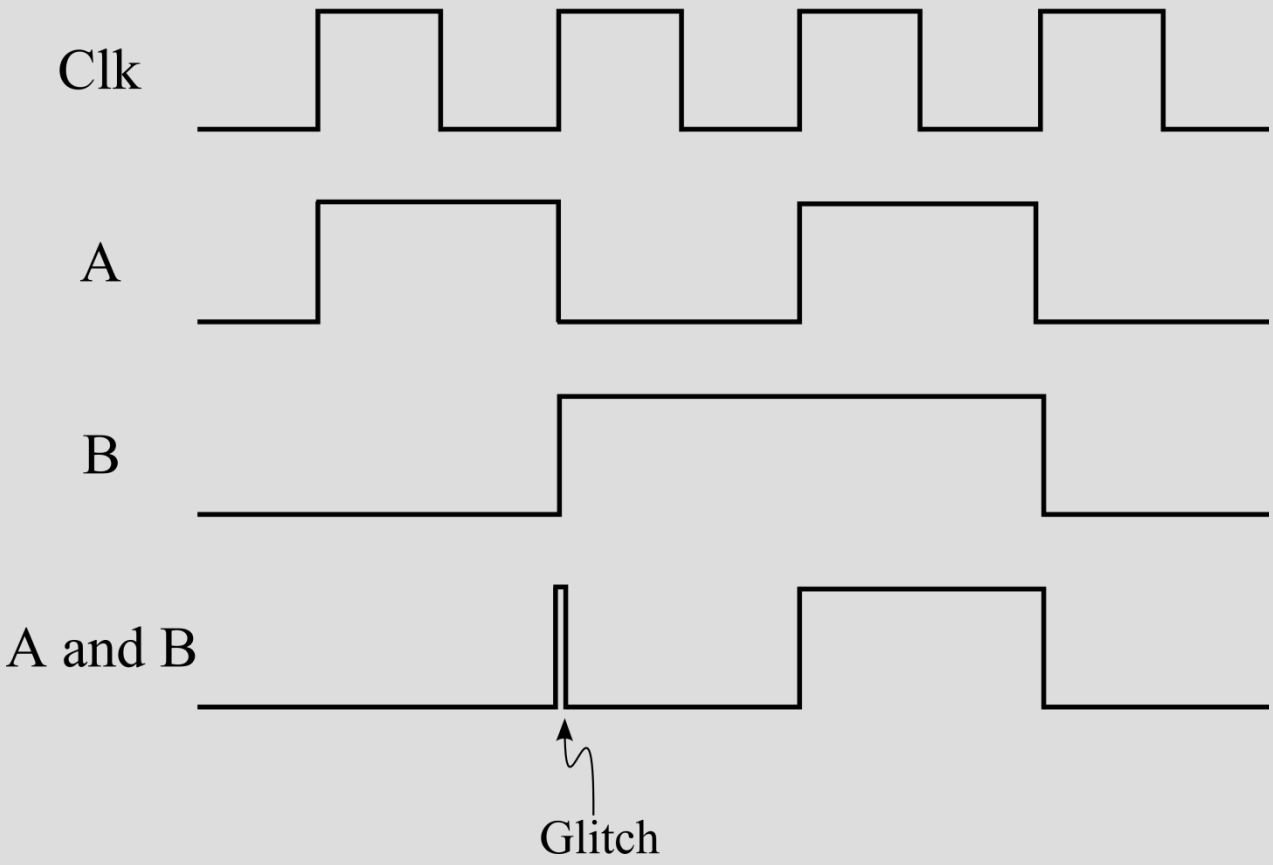


Sinais

- *Glitch*:
  - Sinal espúrio causado pela instabilidade no momento de análise de um sinal (assincronicidade);
  - Normalmente ocorre quando a transição do sinal analisado acontece junto com a transição do sinal de controle;
  - Com o uso de *Latches* deve-se cuidar para não deixar-se uma condição na qual *glitches* possam aparecer;
  - Utilizando *Flip-Flops* deve-se cuidar para garantir que a borda de análise do *clock* nunca coincida com a borda do sinal a ser avaliado.

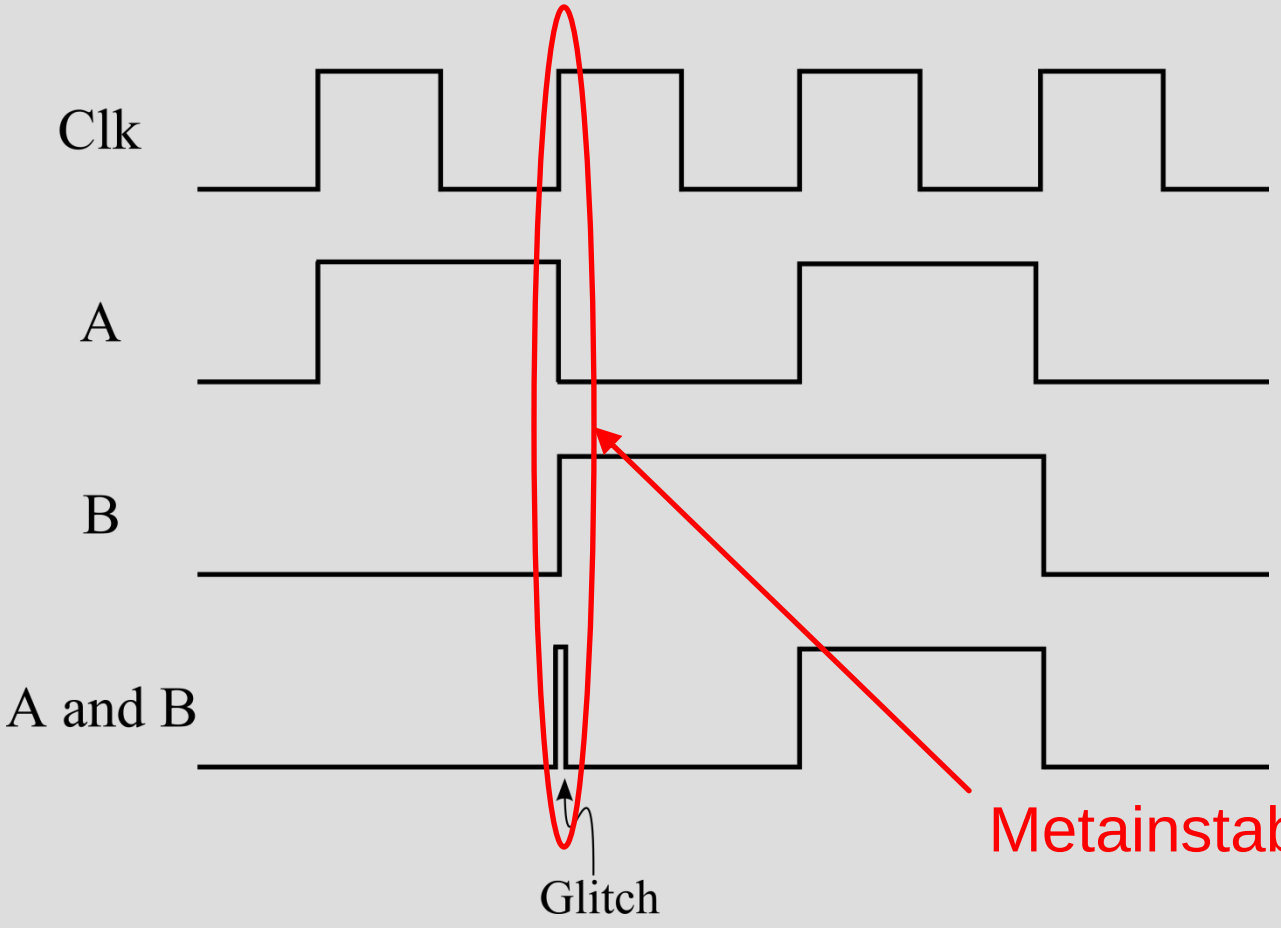
Sinais

- *Glitch*:



Sinais

- *Glitch*:



Sinais

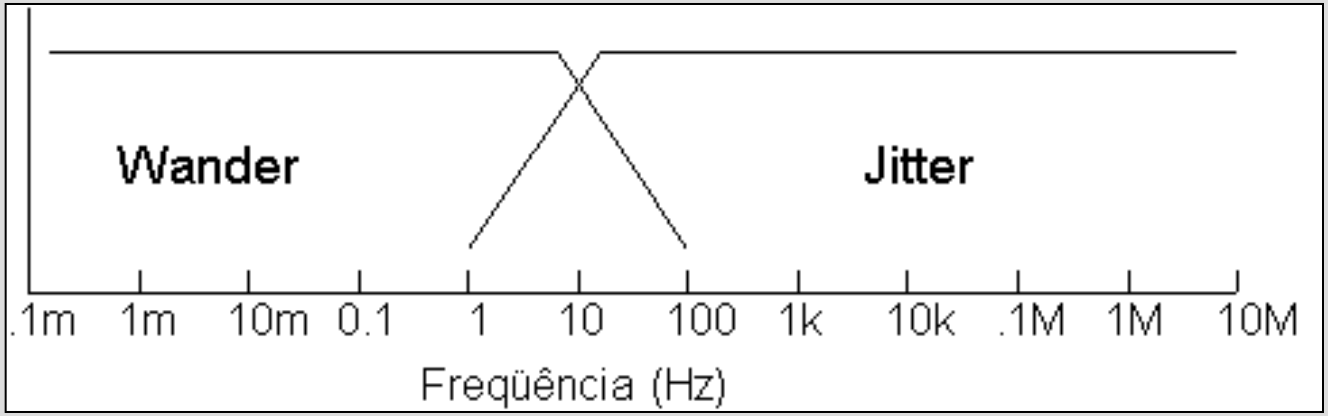
- *Jitter & Wander*:
  - O desvio ou deslocamento de um sinal periódico em torno de seu formato fundamental, seja em freqüência, período, *duty cycle*, amplitude ou ainda fase;
  - Em sistemas digitais *jitter* e *wander* referem-se sempre a período, freqüência ou *duty-cycle*;
  - Causam problemas relacionados ao *timing* (temporização) do circuito, podendo gerar perda de sincronia parcial ou permanente, o que leva a perda de informação.

Sinais

- *Jitter & Wander*:
  - O desvio ou deslocamento de um sinal periódico em torno de seu formato fundamental, seja em freqüência, período, *duty cycle*, amplitude ou ainda fase;
  - Em sistemas digitais *jitter* e *wander* referem-se sempre a período, freqüência ou *duty-cycle*;
  - Causam problemas relacionados ao *timing* (temporização) do circuito, podendo gerar perda de sincronia parcial ou permanente, o que leva a perda de informação.

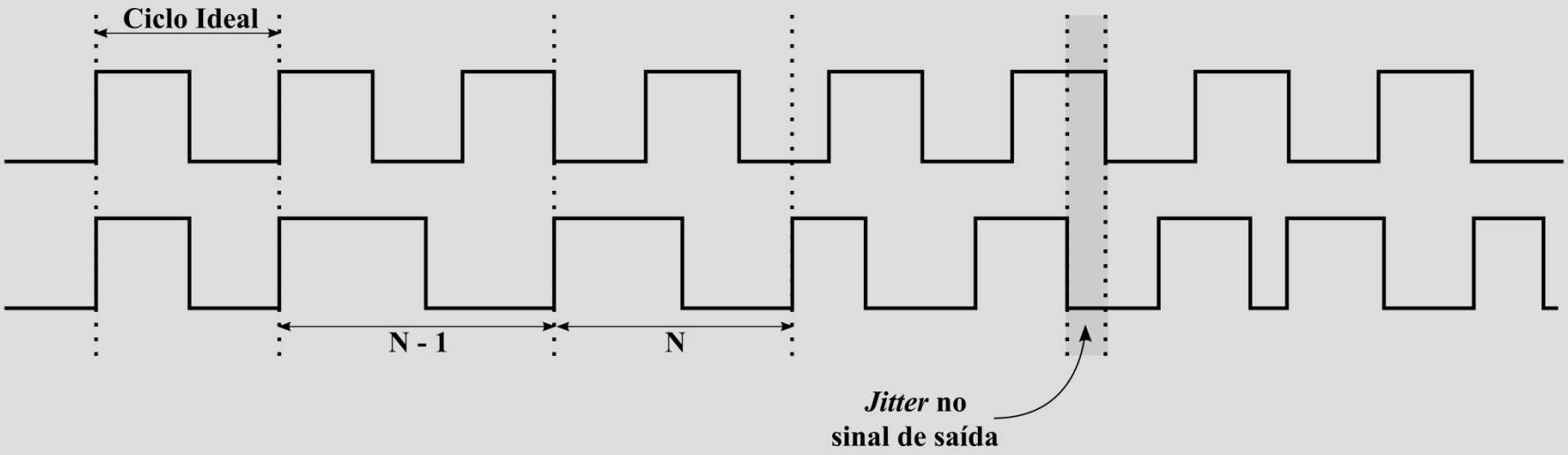
Sinais

- *Jitter & Wander*:



Sinais

- *Jitter & Wander*:
  - Para sistemas digitais o mais crítico dos problemas ocorridos com *clock* é o jitter.



Sinais

- *Jitter & Wander*:
  - *Jitter RMS*:

$$J_{RMS} = \sqrt{\frac{1}{n-1} \cdot \sum_{i=1}^{n} \left(T_i - \overline{T}\right)^2}$$

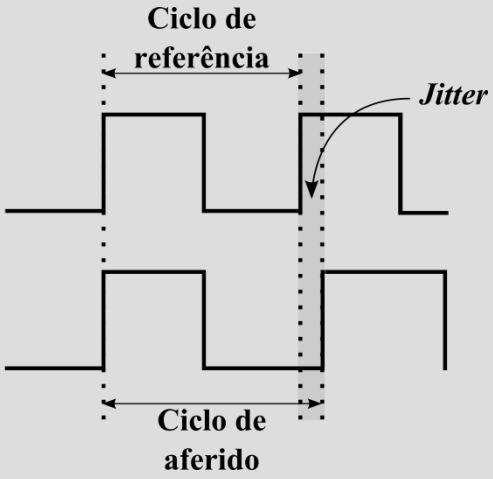
n : *número de amostras;*
T_i : *período dado para cada amostra;*
$\overline{T}$: *média dos períodos medidos dados pela equação* $\overline{T} = \frac{1}{n} \cdot \sum_{i=1}^{n} T_i$

Sinais

- *Jitter & Wander*:
  - *Jitter* Ciclo-a-Ciclo:

$$J_{c2c} = P_n - P_{n-1}$$

P_n : Período medido na amosta n
P_{n-1}: Período medido na amosta $n-1$



Sinais

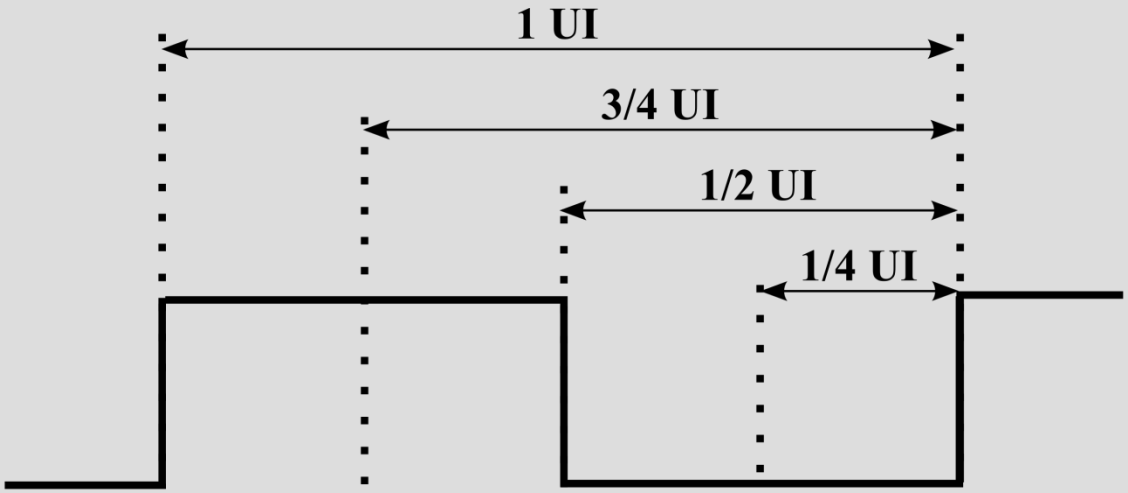
- *Jitter & Wander*:
  - *Jitter* de Distorção de *Duty Cycle*:

$$J_{DCD} = \frac{H_{period}}{H_{period} + L_{period}} \cdot 100\%$$

H_{period} : Perído em nível lógico alto;
L_{period} : Perído em nível lógico baixo.

Sinais

- *Jitter & Wander*:
  - São especificados em Unidade Intervalar (UI):



1 UI = 360º

3/4 UI = 270º

1/2 UI = 180º

1/4 UI = 90º

1 UI = Período do sinal analisado.

Sinais

- *Jitter & Wander*:
  - Exemplo de Especificação:

O *Jitter* permitido para uma interface *STM-1* (*SDH - Synchronous Digital Hierarchy*) é definida em termos de pico-a-pico, como 2.6 mUI. Sendo que a interface *STM-1* possui freqüência de transmissão de 155M52Hz.

$$1UI = \frac{1}{155{,}52MHz} = 6{,}45ns$$

Portanto:

$$2{,}6mUI = \frac{6{,}45ns}{1000} \cdot 2{,}6 = 16{,}77\,ps$$

Sinais

- Diagrama de Olho:
  - O diagrama de olho provê uma útil informação visual que auxilia na avaliação de problemas em transmissões digitais;
  - Faz uma análise qualitativa da transmissão;
  - Pode dar uma informação quantitativa aproximada;
  - Através de uma análise criteriosa e cuidadosa pode dar uma aproximação inicial sobre a quantidade de skew e jitter presentes no sinal;

Sinais

- Diagrama de Olho:

  - A técnica de aquisição da imagem do diagrama de olho baseia em pegar um padrão de variação de dados no barramento ou canal de transmissão;
  - Pode-se criar este padrão (recomendável) ou então analisar o barramento em condição normal de funcionamento;

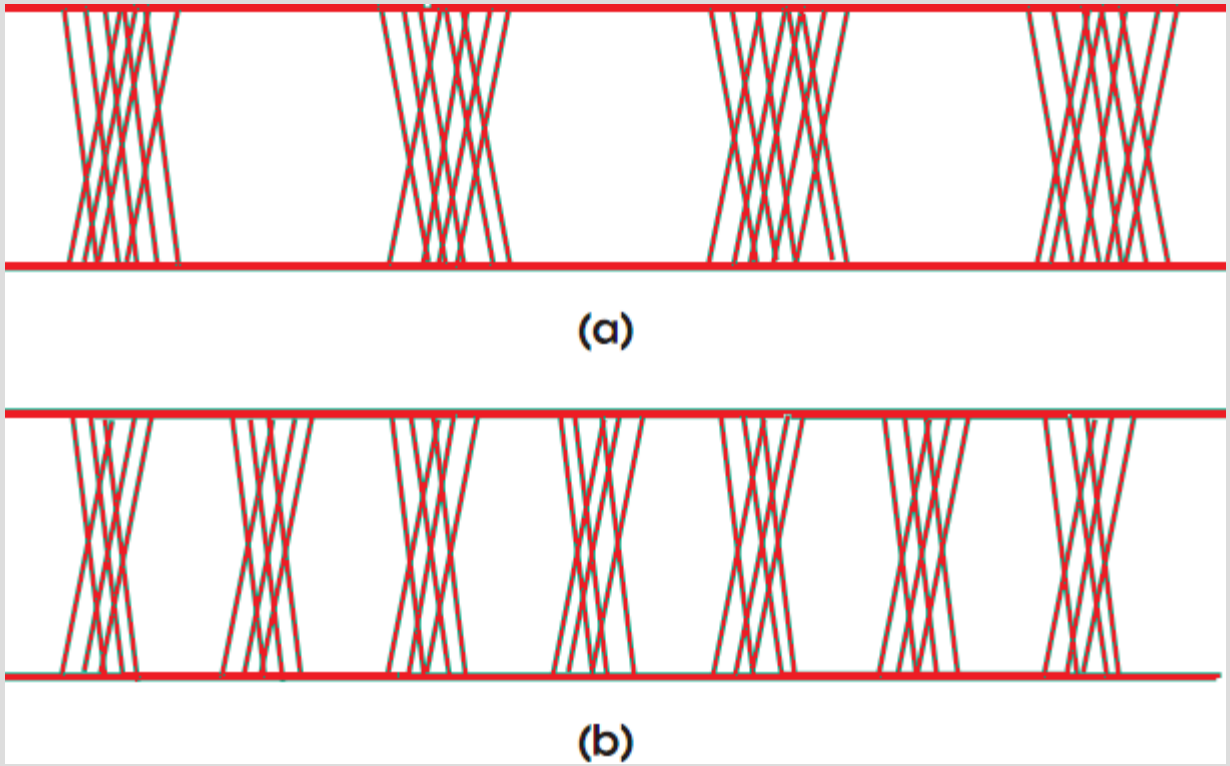
IMPORTANTE: Não será válida a análise do diagrama de olho (não haverá olho) em sinais repetitivos do mesmo padrão.

Sinais

- Diagrama de Olho:
  - Idealmente:

Sinais

- Diagrama de Olho:
  - Na prática:



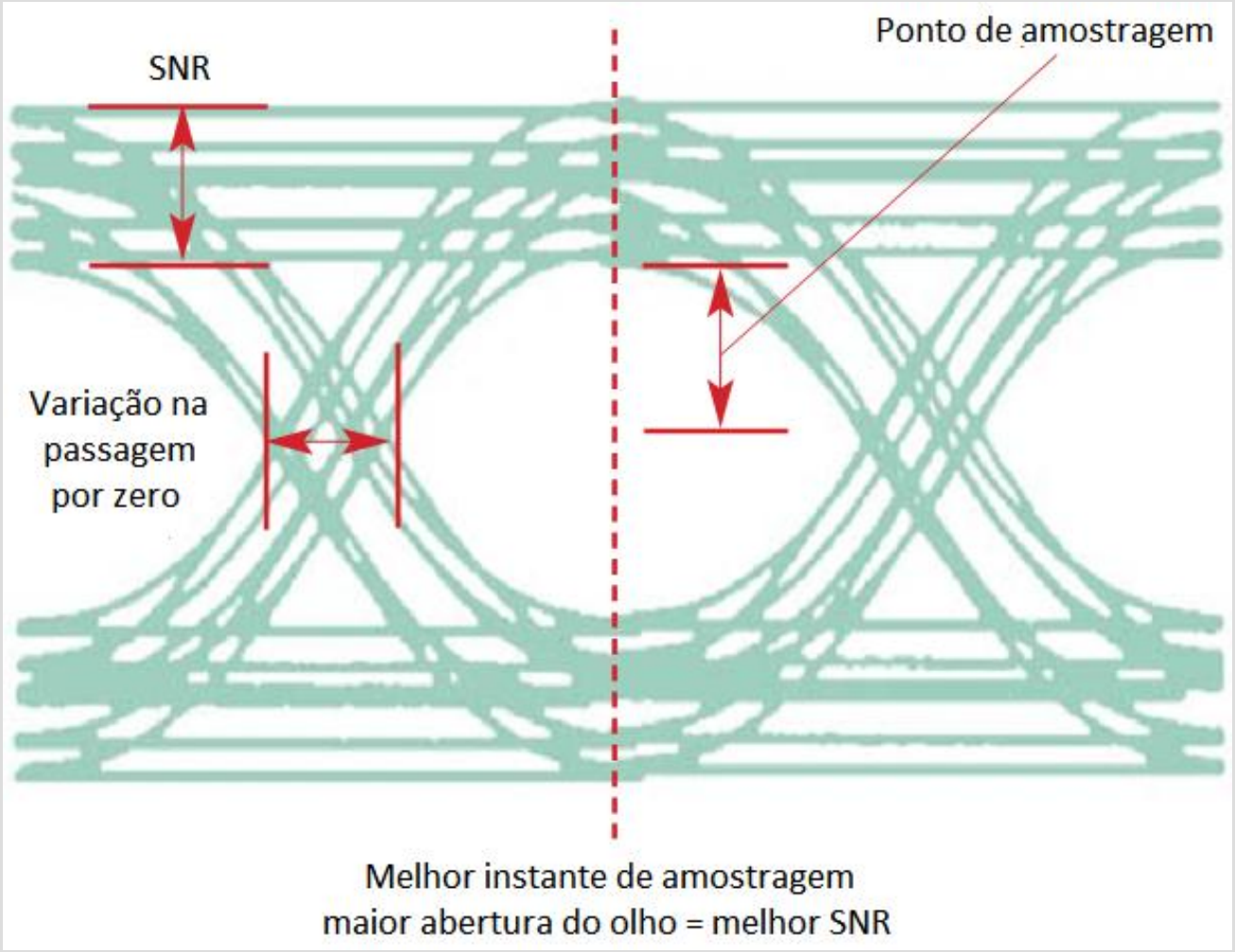
Sinais

- Diagrama de Olho:
  - Na prática:
    - Em **a** temos uma maior quantidade de *jitter*;
    - Em **b** temos apesar de uma menor quantidade de *jitter* uma abertura de olha menor.

IMPORTANTE: em **b** temos uma maior taxa de dados, o que explica a diferença de abertura de olho, apesar da melhor qualidade do sinal.

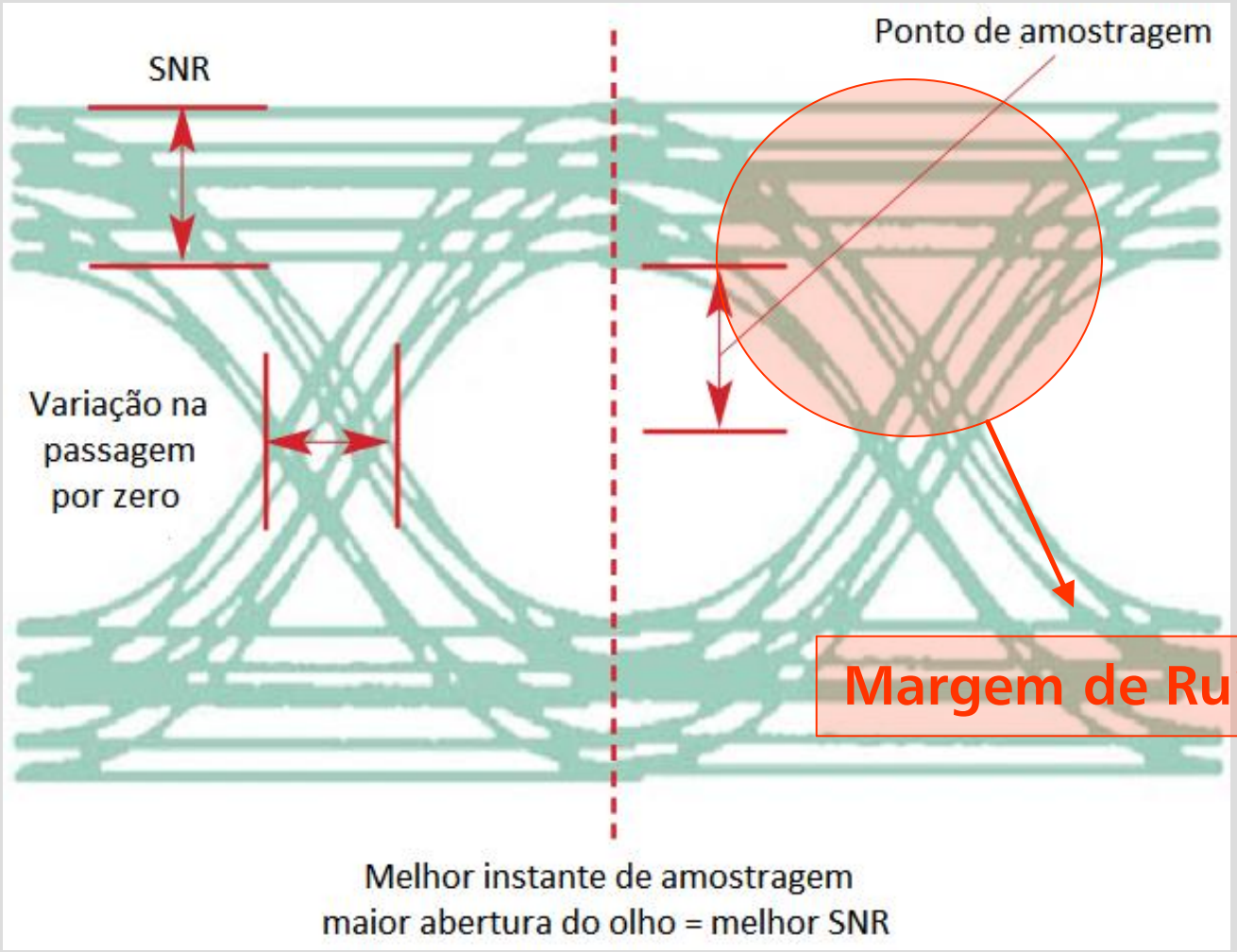
Sinais

- Diagrama de Olho – Parâmetros:



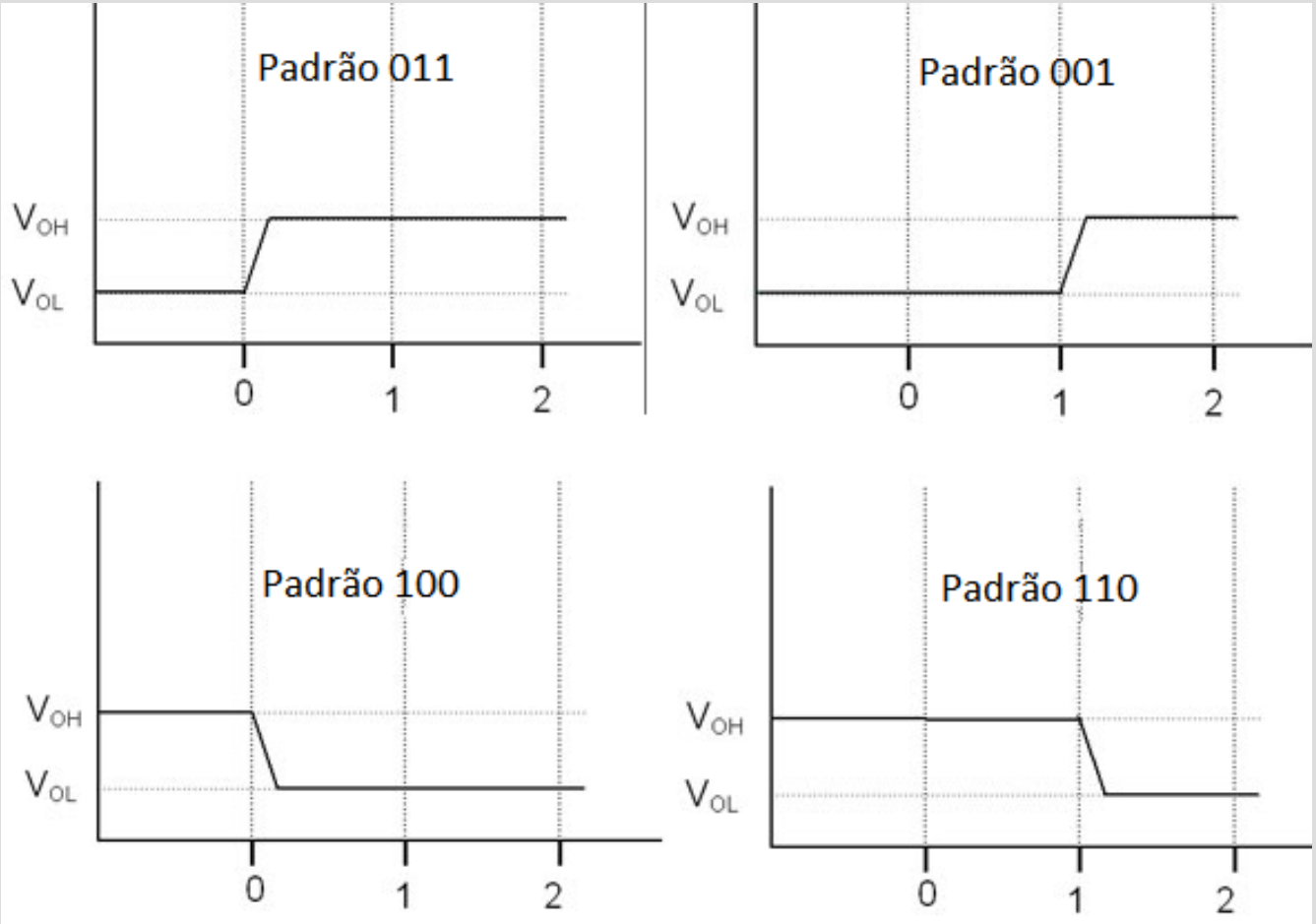
Sinais

- Diagrama de Olho - Parâmetros:



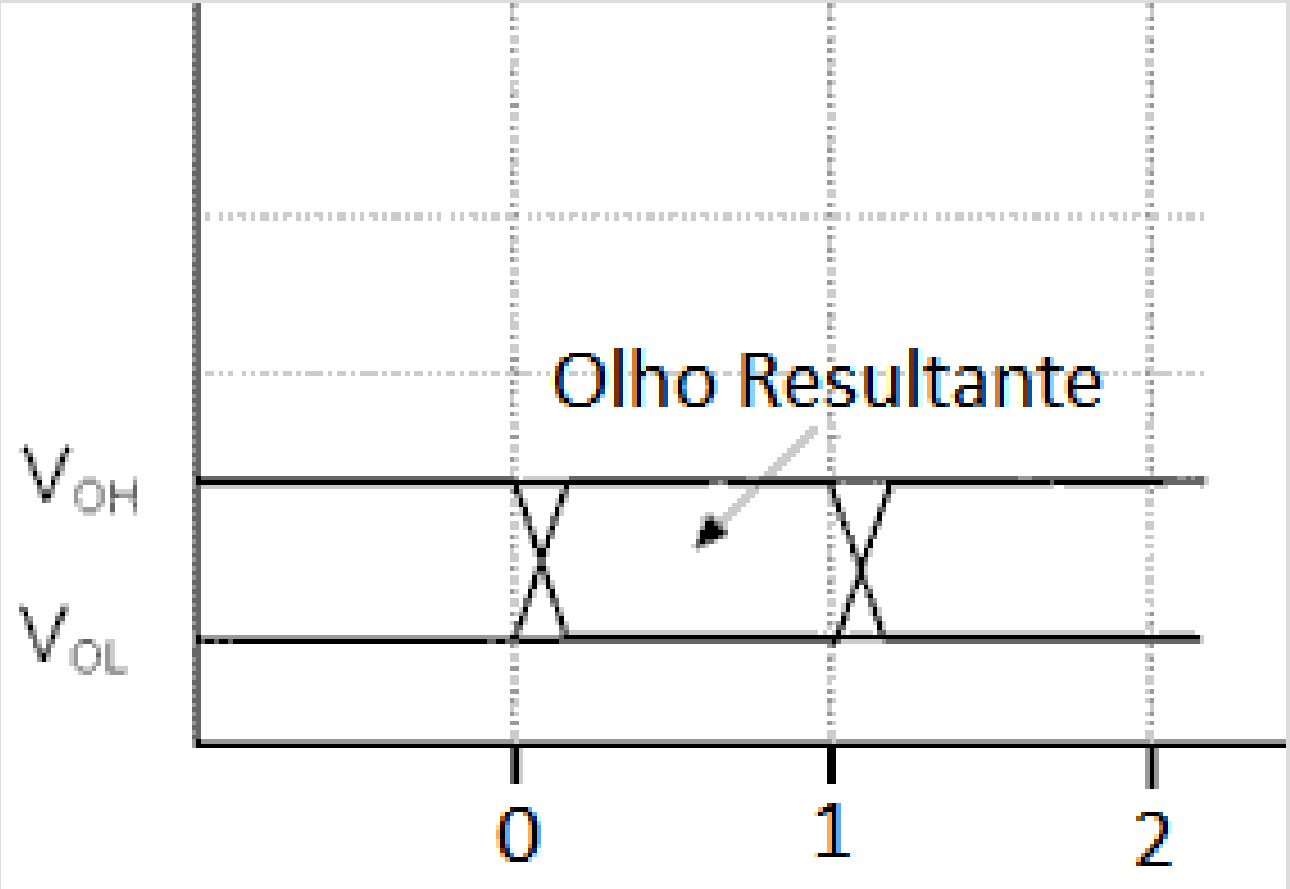
Sinais

- Diagrama de Olho - Padrões:



Sinais

- Diagrama de Olho - Padrões:

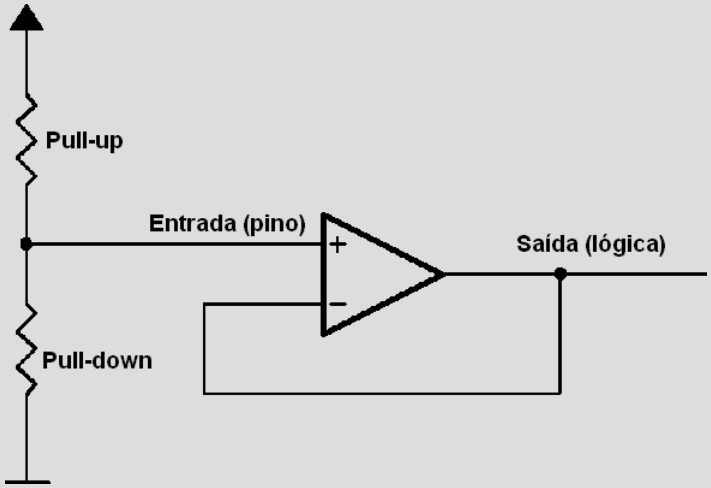


Interfaceamento

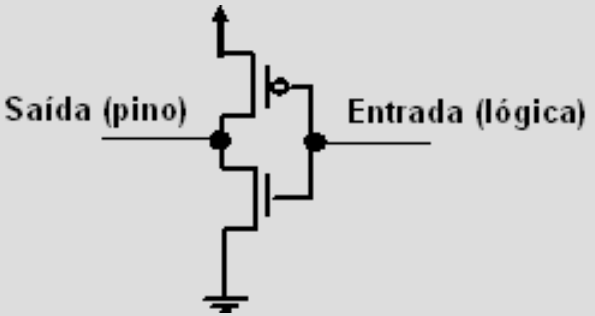
- Interfaces elétricas de *FPGAs*:
  - As aplicações modernas superaram a um certo tempo os padrões de interfaceamento *TTL* e *CMOS*;
  - Principalmente pelo aumento da velocidade de sinalização e distância entre transmissores e receptores;
  - Cada interface elétrica busca melhorar um aspecto não coberto ou fragilizado na aplicação de sua predecessora;
  - Atualmente tem-se uma gama muito grande de interfaces elétrica tanto para sistemas analógicos como para digitais;
  - Interfaces digitais evoluíram de maneira exponencial nos anos 90, com as novas exigências do mercado para com emissão eletromagnética.

Interfaceamento

- Interfaces elétricas de *FPGAs*:
  - *LVTTL*:
    - *Low Voltage TTL*;
    - Para uso de propósito geral;
    - Padrão de 3V3;
    - Possui um *buffer* de entrada e um *buffer push-pull* na saída.



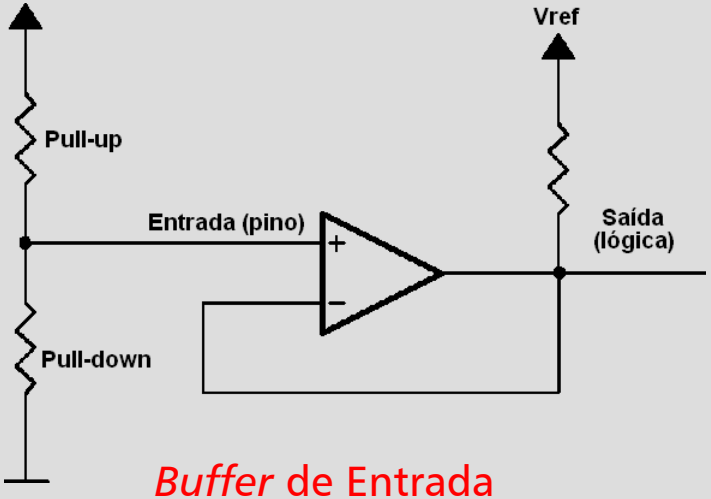
Buffer de Entrada



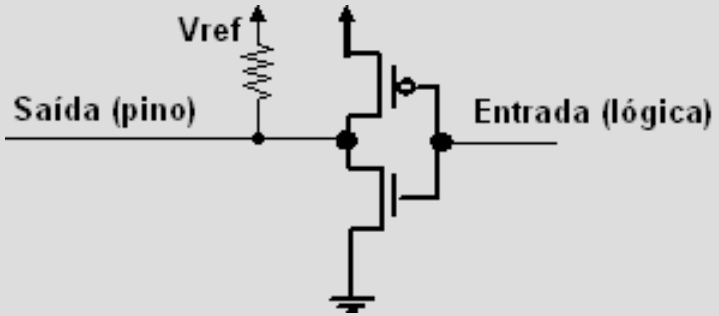
Buffer de Saída

Interfaceamento

- Interfaces elétricas de *FPGAs*:
  - *LVCMOS:*
    - *Low Voltage CMOS*;
    - Para uso de propósito geral;
    - Padrão de 1V2 a 3V3 (Requer alimentação nos pinos auxiliares);
    - Possui a mesma configuração de entrada e saída do *LVTTL*, porém com a adição de referências auxiliares (Vref).



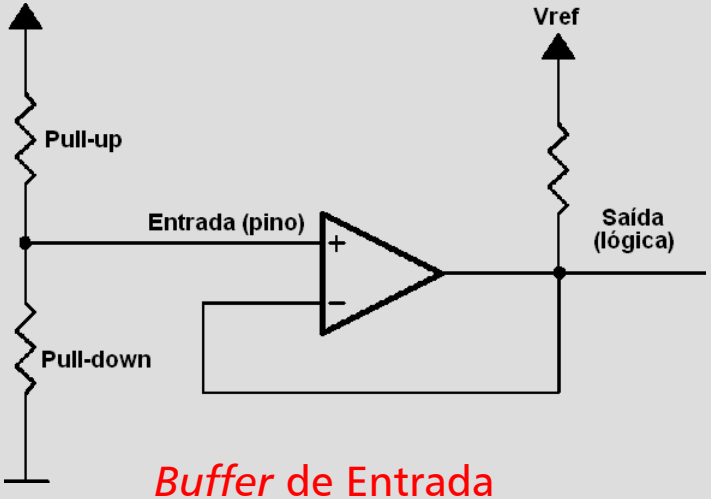
Buffer de Entrada



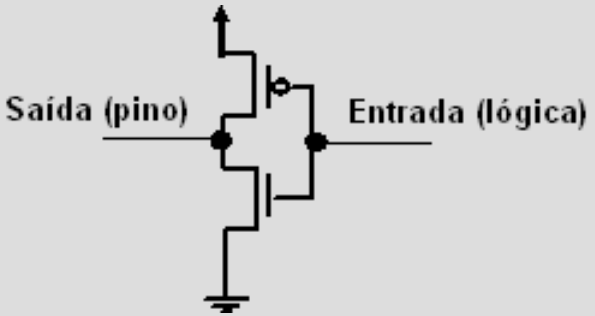
Buffer de Saída

Interfaceamento

- Interfaces elétricas de *FPGAs*:
  - *PCI*:
    - *Peripheral Component Interface*;
    - Para uso em barramentos com freqüências típicas de 33 a 66MHz;
    - Utiliza os *buffers* de saída do *LVTTL* e de entrada do *LVCMOS*;
    - Trabalha a 3V3.



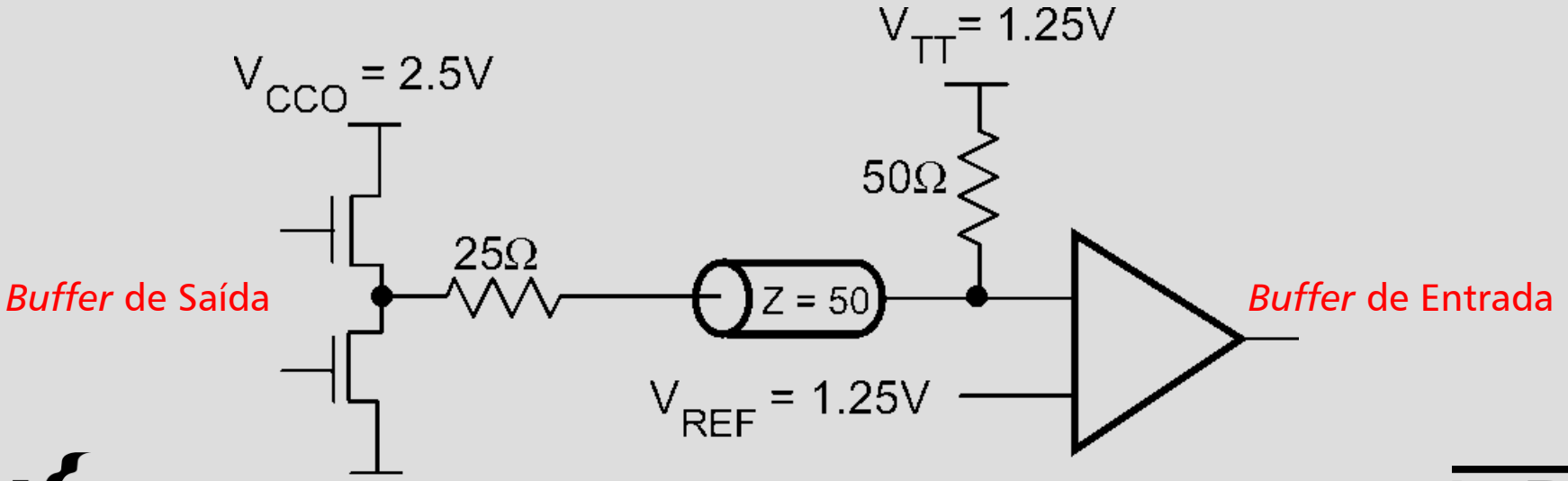
Buffer de Entrada



Buffer de Saída

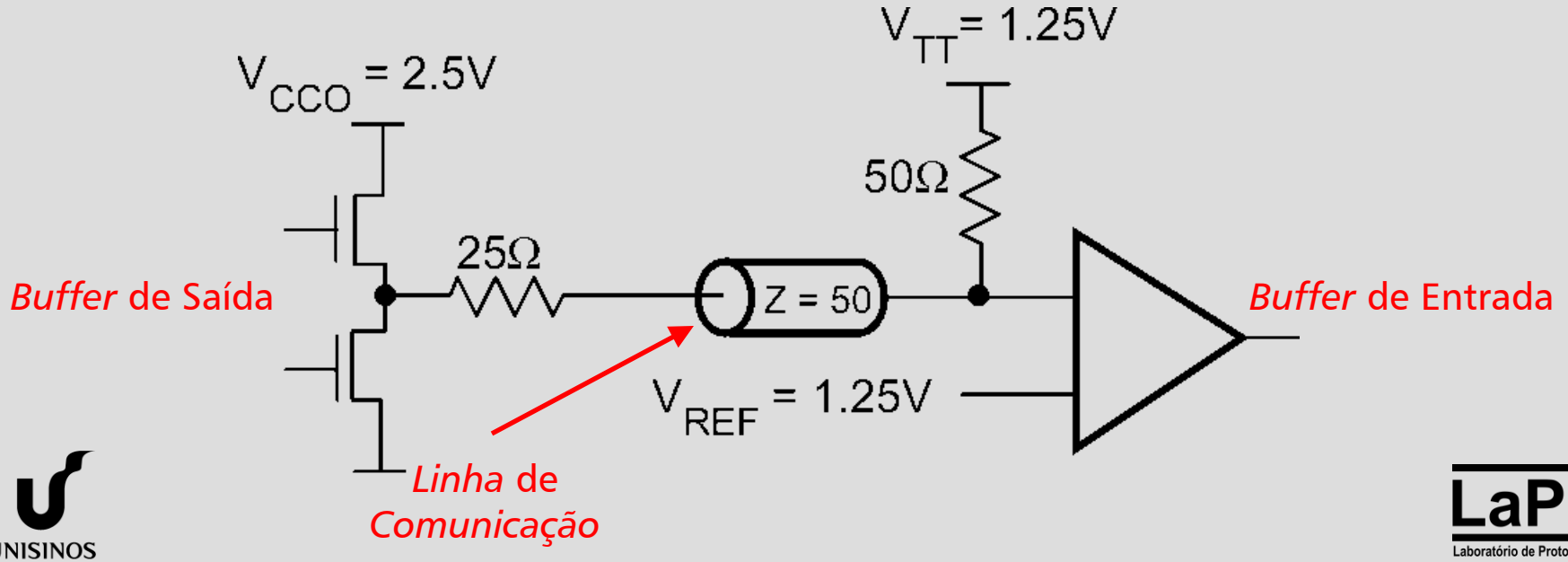
Interfaceamento

- Interfaces elétricas de *FPGAs*:
  - *HSTL*:
    - *High Speed Tranceiver Logic*;
    - Para uso de propósito geral em alta velocidade (200 MHz);
    - Utiliza padrão de tensão de 1V5 a 1V8;
    - Necessita de alimentação auxiliar.



Interfaceamento

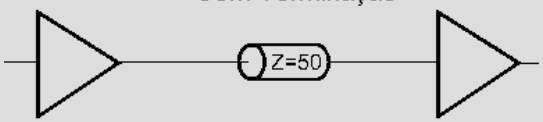
- Interfaces elétricas de *FPGAs*:
  - *HSTL*:
    - *High Speed Tranceiver Logic*;
    - Para uso de propósito geral em alta velocidade (200 MHz);
    - Utiliza padrão de tensão de 1V5 a 1V8;
    - Necessita de alimentação auxiliar.



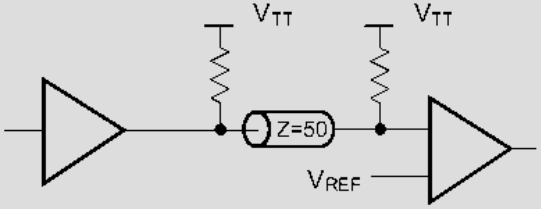
Interfaceamento

- Interfaces elétricas de *FPGAs*:
  - *HSTL*:
    - Possui diversas configurações.

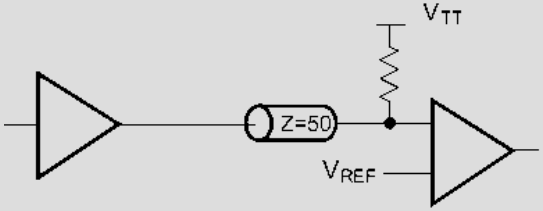
Sem Terminação



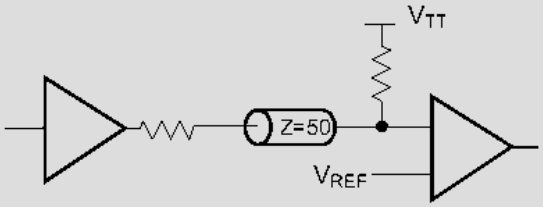
Terminação Dupla Paralela



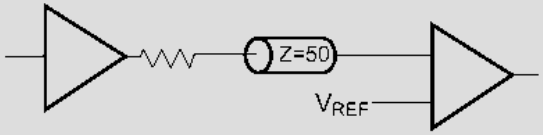
Sem Terminação de Saída
Com Terminação Paralela de Entrada



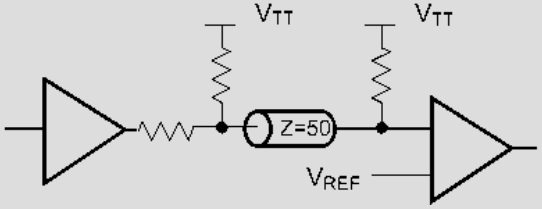
Terminação Série de Saída
Terminação Paralela de Entrada



Terminação Série de Saída



Terminação Série/Paralele de Saída
Terminação Paralelo de Entrada

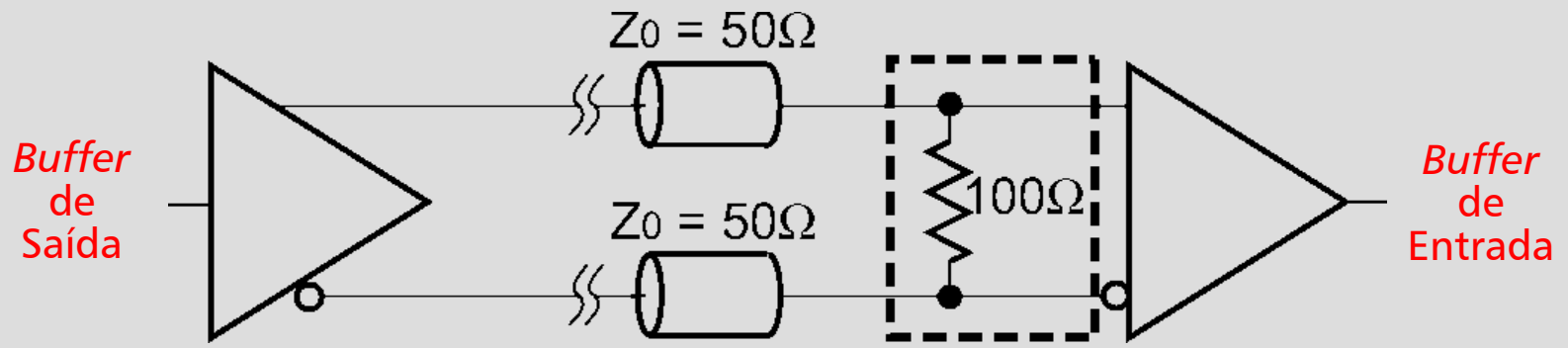


Interfaceamento

- Interfaces elétricas de *FPGAs*:
  - *SSTL x*:
    - *Stub Series Terminated Logic xV*;
    - Para uso em barramentos de memória;
    - Utiliza padrão de tensão de 1V8 (*SSTL18*) a 3V3 *SSTL3*;
    - Necessita de alimentação auxiliar;
    - Utiliza o mesmo *buffer* utilizado pelo padrão de tensão *HSTL*.

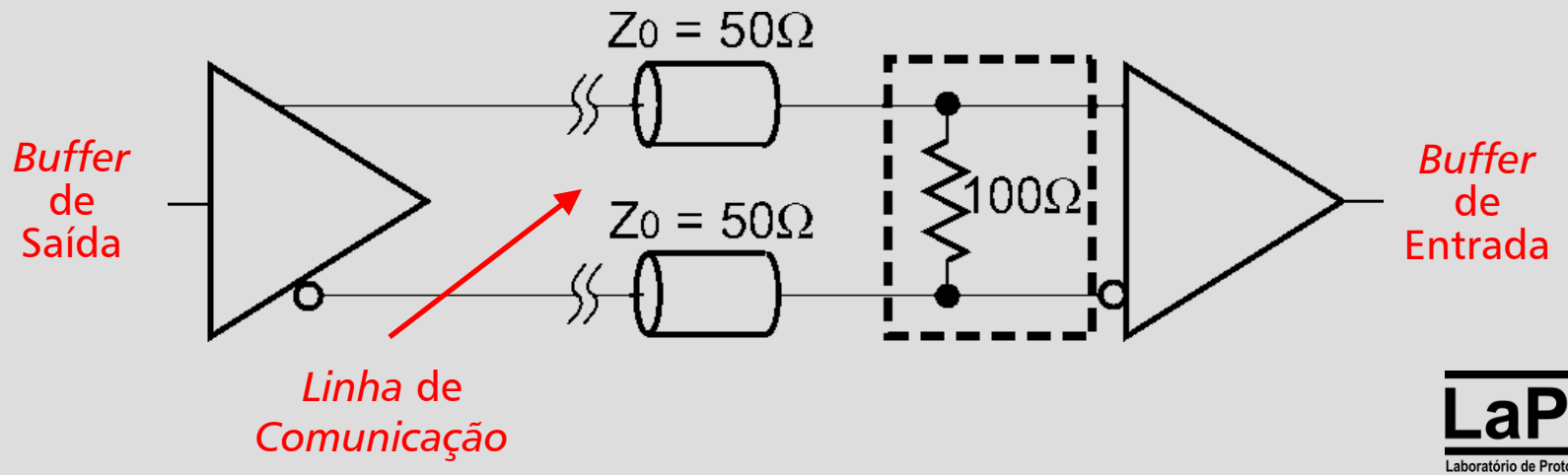
Interfaceamento

- Interfaces elétricas de *FPGAs*:
  - *LVDS*:
    - *Low Voltage Differencial Signal*;
    - Para uso de propósito geral de alta velocidade diferencial (333 MHz);
    - Requer o uso de 2 pinos vizinhos do *FPGA* (já definidos pela arquitetura);
    - Necessita de terminação externa via resistor;
    - Utiliza 3V3 e 2V5 como padrão de sinalização.



Interfaceamento

- Interfaces elétricas de *FPGAs*:
  - *LVDS*:
    - *Low Voltage Differencial Signal*;
    - Para uso de propósito geral de alta velocidade diferencial (333 MHz);
    - Requer o uso de 2 pinos vizinhos do *FPGA* (já definidos pela arquitetura);
    - Necessita de terminação externa via resistor;
    - Utiliza 3V3 e 2V5 como padrão de sinalização.

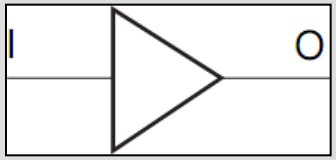


Interfaceamento

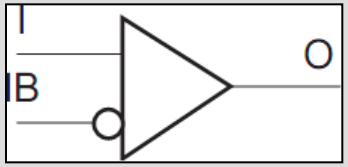
- Interfaces elétricas de *FPGAs*:
  - *LVDS Extended*:
    - Possui maior potência de transmissão que o *LVDS*.
  - *BLVDS*:
    - *LVDS* para aplicações bidirecionais.
  - *LVPECL:*
    - Similar ao *LVDS* porém com potência de transmissão superior ao *LVDS Extended*.
  - *TMDS*:
    - Próprio para transmissão de alta fidelidade em alta velocidade.
  - *RSDS*:
    - Próprio para *drives* de controle de *LCDs*.
  - *PPDS*:
    - Variação do *RSDS* com suporte a maiores velocidades

Interfaceamento

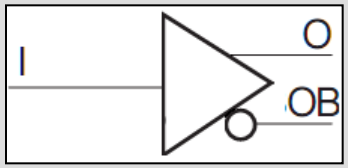
- Interfaces elétricas de *FPGAs*:
  - Os pinos dos *FPGAs* salvo algumas exceções são configuráveis quanto ao sentido de troca de informação;
  - Para tanto existem alguns blocos de implementação que explicitam este desejo por parte do projetista:



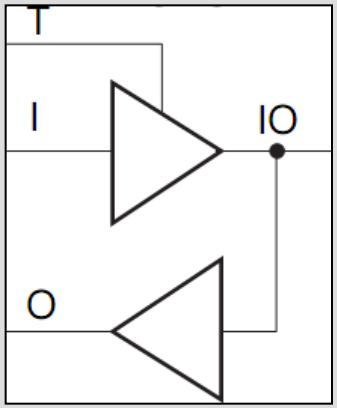
IBUF/
OBUF



IBUFDS



OBUF



IOBUF

Interfaceamento

- Interfaces elétricas de *FPGAs*:
  - *MGT*:
    - *Multigigabit Transceiver*;
    - Transmissão *full-duplex*;
    - Velocidade variável de 600 Mpbs até 3.125 Gbps;
    - Não necessita de componentes externos;
    - Função de *auto lock reference*;
    - Cinco níveis de potência de transmissão configuráveis;
    - Quatro níveis de *pre-enphasis* programáveis;
    - Acoplamento *AC* e *DC*;
    - Impedância configurável (50 Ω ou 75Ω);
    - Preâmbulo de 10 *bits* programável;
    - *Loopbacks* programáveis em tempo de operação.

Fontes

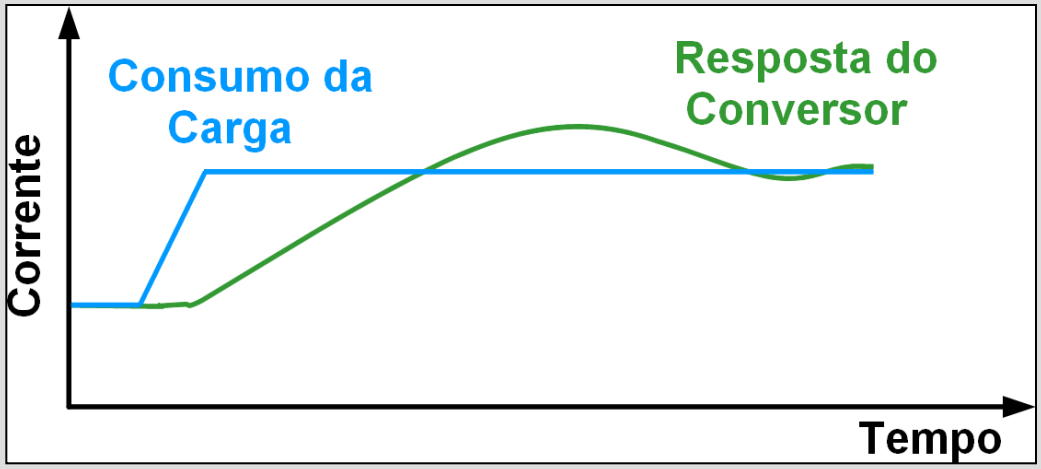
- Fontes para *PLDs*:
  - Com o avançar da tecnologia os *PLDs* passaram por processos de:
    - Miniaturilização;
    - Aumento da velocidade de processamento;
    - Aumento da velocidade de suas interfaces elétricas;
    - Aumento da potência consumida.
  - Outra característica interessante é a necessidade de um número cada vez maior de diferentes tensões para garantir o funcionamento dos PLDs modernos;
  - Isto se deve ao fato da especialização de cada grupo interno destes dispositivos (*IOBs*, *MGTs*, *DCMs*, *LUTs*, Cores de Lógica Digital e *Cores* de microprocessadores);
  - Quanto maior a velocidade permitida pelo *PLD*, maior será sei consumo.

Fontes

- Fontes para *PLDs*:
  - Para chegar as velocidades máximas atuais os fabricantes baixaram a alimentação dos *cores*;
  - Com a corrente aumenta e a dissipação por *Efeito Joule* aumenta;
  - Fontes de *PLDs* tem diferentes níveis de exigência e por conseqüência diferentes níveis de complexidade:
    - *SPLDs* podem trabalhar com tolerâncias tensão e *ripple* de 10%;
    - *CPLDs* toleram tensão e *ripple* em 5%;
    - *FPGAs* (*low-end* e *medium-end*) toleram tensão em 5% e *ripple em* 2%*;*
    - *FPGAs* (*high-end*) toleram tensão e *ripple* em 2%;
    - *FPGAs* especializados (*Aplicações Militar/Aeroespacial*) toleram tensão e *ripple* em no máximo 1%.

Fontes

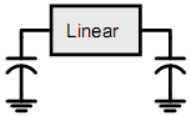
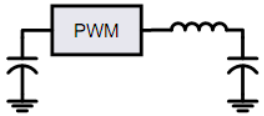
- Requisitos para Fontes de *PLDs*:
  - Necessidade de pouca variação de tensão quando do chaveamento entre modos de consumo:
    - É bastante comum um *PLD* ficar em trabalhando em um modo de baixo consumo (sem processamento) e chavear para um modo de alto consumo (com processamento) de maneira freqüente.



  - Fontes com baixo ruído e baixa emissão eletromagnética;
  - Rápida resposta ao degrau;
  - Seqüenciamento de partida;

Fontes

- Requisitos para Fontes de *PLDs*:
  - Para atender os requisitos mostrados anteriormente o tipo de regulador de tensão ideal é o linear;
  - Porém este regulador tem uma baixa eficiência o que o torna por vezes impossível de ser utilizado;

	Regulador Linear	Switching Mode Power Supply (SMPS)	Módulo DC/DC	Módulo de Potência Integrado
				
Eficiência	Baixo	Alto	Alto	Alto
Ruído	Baixo	Alto	Alto	Alto
Custo	Baixo	Médio	Alto	Alto
Tamanho	Pequeno	Médio	Médio-Grande	Pequeno
Desenvolvimento	Baixo	Médio-Alto	Baixo	Baixo
Performance*	Alto	Médio	Baixo	Alto

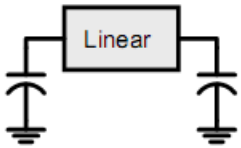
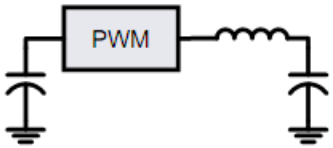
*Performance no transiente!

UNISINOS
Somos infinitas possibilidades

LaPSE
Laboratório de Prototipação Digital e Sistemas Embarcados

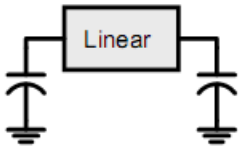
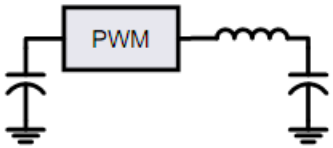
Fontes

- Requisitos para Fontes de *PLDs*:

	Regulador Linear	Switching Mode Power Supply (SMPS)	Módulo DC/DC	Módulo de Potência Integrado
				
Eficiência	Baixo	Alto	Alto	Alto
Ruído	Baixo	Alto	Alto	Alto
Custo	Baixo	Médio	Alto	Alto
Tamanho	Pequeno	Médio	Médio-Grande	Pequeno
Desenvolvimento	Baixo	Médio-Alto	Baixo	Baixo
Performance*	Alto	Médio	Baixo	Alto

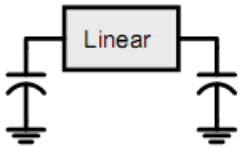
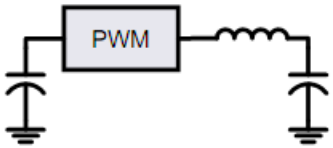
Fontes

- Requisitos para Fontes de *PLDs*:

	Regulador Linear	Switching Mode Power Supply (SMPS)	Módulo DC/DC	Módulo de Potência Integrado
				
Eficiência	Baixo	Alto	Alto	Alto
Ruído	Baixo	Alto	Alto	Alto
Custo	Baixo	Médio	Alto	Alto
Tamanho	Pequeno	Médio	Médio-Grande	Pequeno
Desenvolvimento	Baixo	Médio-Alto	Baixo	Baixo
Performance*	Alto	Médio	Baixo	Alto

Fontes

- Requisitos para Fontes de *PLDs*:

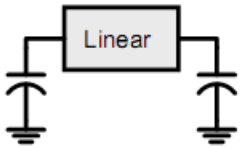
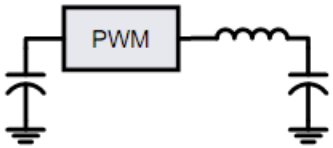
	Regulador Linear	Switching Mode Power Supply (SMPS)	Módulo DC/DC	Módulo de Potência Integrado
				
Eficiência	Baixo	Alto	Alto	Alto
Ruído	Baixo	Alto	Alto	Alto
Custo	Baixo	Médio	Alto	Alto
Tamanho	Pequeno	Médio	Médio-Grande	Pequeno
Desenvolvimento	Baixo	Médio-Alto	Baixo	Baixo
Performance*	Alto	Médio	Baixo	Alto

UNISINOS
Somos infinitas possibilidades

LaPSE
Laboratório de Prototipação Digital e Sistemas Embarcados

Fontes

- Requisitos para Fontes de *PLDs*:

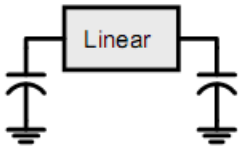
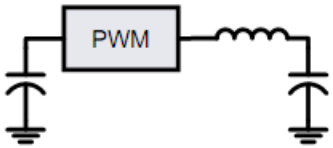
	Regulador Linear	Switching Mode Power Supply (SMPS)	Módulo DC/DC	Módulo de Potência Integrado
				
Eficiência	Baixo	Alto	Alto	Alto
Ruído	Baixo	Alto	Alto	Alto
Custo	Baixo	Médio	Alto	Alto
Tamanho	Pequeno	Médio	Médio-Grande	Pequeno
Desenvolvimento	Baixo	Médio-Alto	Baixo	Baixo
Performance*	Alto	Médio	Baixo	Alto

UNISINOS
Somos infinitas possibilidades

LaPSE
Laboratório de Prototipação Digital e Sistemas Embarcados

Fontes

- Requisitos para Fontes de *PLDs*:

	Regulador Linear	Switching Mode Power Supply (SMPS)	Módulo DC/DC	Módulo de Potência Integrado
				
Eficiência	Baixo	Alto	Alto	Alto
Ruído	Baixo	Alto	Alto	Alto
Custo	Baixo	Médio	Alto	Alto
Tamanho	Pequeno	Médio	Médio-Grande	Pequeno
Desenvolvimento	Baixo	Médio-Alto	Baixo	Baixo
Performance*	Alto	Médio	Baixo	Alto

UNISINOS
Somos infinitas possibilidades

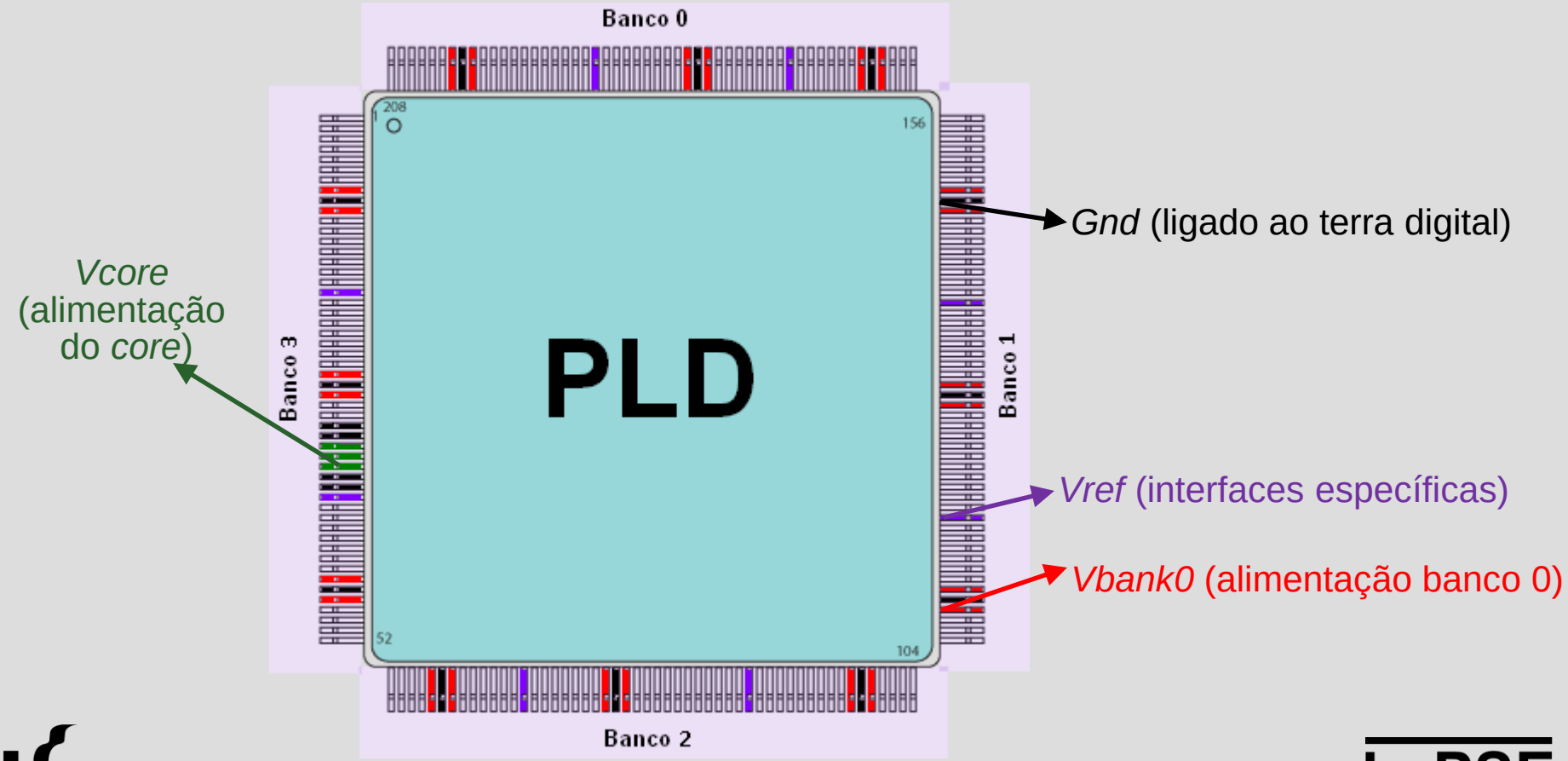
LaPSE
Laboratório de Prototipação Digital e Sistemas Embarcados

Fontes

- Fontes para *PLDs*:
  - *PLDs* possuem bancos de alimentação, que definem as possibilidades de padrões de interface nos pinos relacionados aos mesmos;
  - É bastante comum ter-se ao menos 2 tensões diferentes para a alimentação de um *CPLD* e 3 para *FPGAs*;
    - *CPLD*: bancos de alimentação;
    - *FPGA*: bancos de alimentação, tensões auxiliares e alimentação do *core*;
  - Utilizando uma solução conjunta de módulos conversores *DC-DC* e de *POL* (*Point of Load*) pode-se cobrir de maneira eficiente as diferentes tensões necessárias.

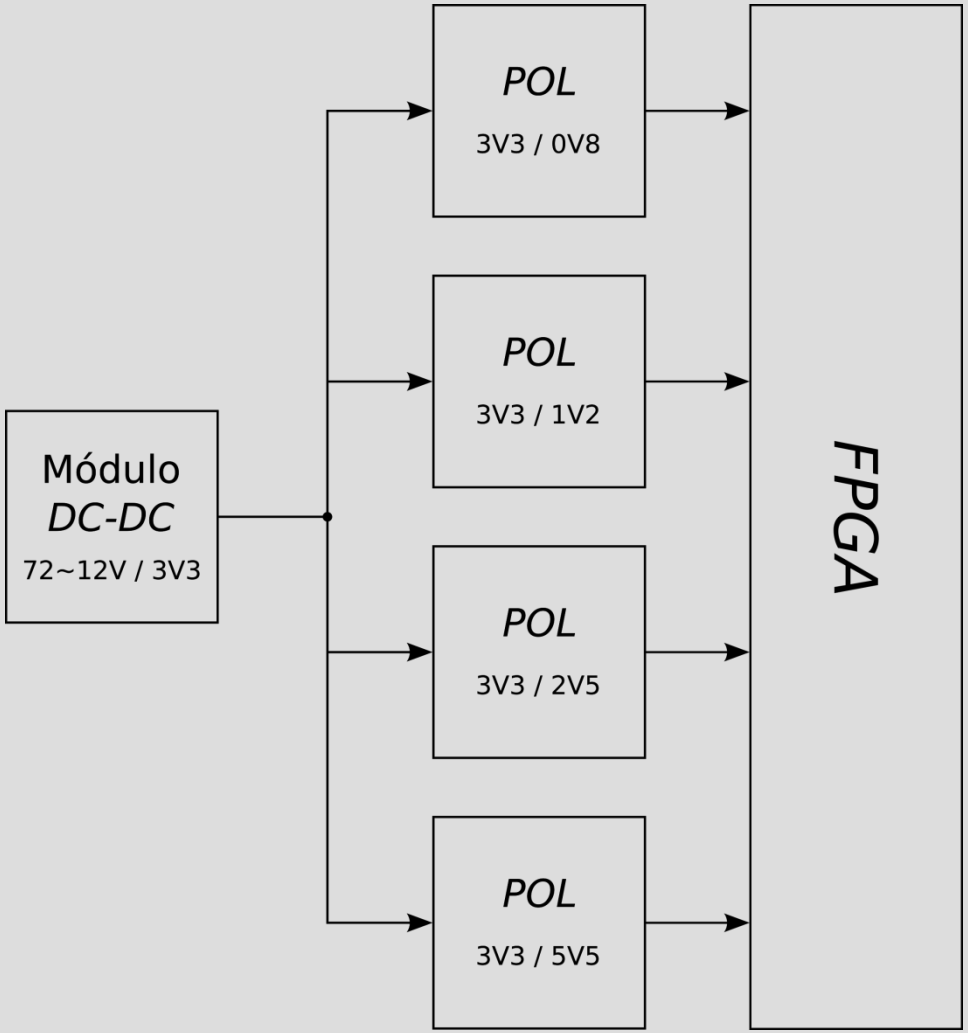
Fontes

- Fontes para *PLDs*:



Fontes

- Fontes para *PLDs*:



LaPSE

Laboratório de Prototipação Digital
e Sistemas Embarcados

Universidade do Vale do Rio dos Sinos - UNISINOS

Prototipação em *PLDs*

Obrigado!