

PSI-3451 Projeto de CI Lógicos Integrados

Aula 11- Conceitos relacionados ao Fluxo de Projeto de FPGAs

O objetivo desta aula é a exploração por parte do(a) aluno(a) de mais alguns aspectos do sintetizador lógico Leonardo Spectrum e , em uma segunda parte, das possibilidades providenciadas pelas ferramentas e opções fornecidas pelo programa de projeto de FPGAs, Quartus II (Altera/Intel).

Os conceitos relacionados à síntese de ASICs já foi assunto de apostila da aula 10, sendo que, nesta apostila, somente aspectos do projeto em FPGAs serão tratados. O aluno deve complementá-los com o material do livro-texto de Pong Chu. (seções 3.1 a 3.3)

--> **Em que implica o projeto de FPGA?** Quando se usa a terminologia "Projeto FPGA", assim como para ASICs, estamos entendendo que temos CIs que foram projetados com propósito(s) específico(s), ou seja, não seriam destinados a muitos diferentes tipos de atividade. Implica em projetos curtos (menos de um ano) devido ao alto reuso de blocos pré-projetados e na programação em um dispositivo já fabricado (em oposição ao processo de fabricação de ASICs). Usa-se esta estratégia quando um número relativamente pequeno de peças finais são necessárias.

1. Arquitetura de um FPGA

Field Programmable Gate Arrays ou FPGAs são dispositivos programáveis de variadas capacidades. Há famílias lógicas com tecnologias de fabricação de CIs dos nós tecnológicos mais avançados, assim como de tecnologias mais maduras e antigas. Há também oferecimento de dispositivos pequenos, com poucos pinos de entrada e saída, e células lógicas programáveis, assim como aqueles de grandes dimensões, apropriados para o mapeamento de sistemas digitais

A Figura 1 apresenta uma visão simplificada de parte de uma arquitetura típica de FPGAs, como é o caso da família Cyclone II que utilizamos em nossos exercícios (notar que uma FPGA pode ser composta de dezenas de milhares de células lógicas). A lógica de um circuito sendo projetado, após ser particionado em sub-blocos, é mapeado de forma distribuída entre as diversas células lógicas programáveis da figura; por outro lado, seguindo-se o esquema lógico/especificação do circuito, os blocos mapeados para as células lógicas são interligados (por suas respectivas entradas e saídas) através da programação de chaves que conectam trilhas verticais e horizontais (os blocos S da figura). A programação das lógicas e das interconexões dentro da FPGA é definida pelo programa de projeto Quartus e realizada por células de memória SRAM.

2. Fluxo de Projeto a partir de descrição RTL

A estratégia adotada no programa Quartus para o fluxo de projeto de um circuito/sistema apresenta uma similaridade com o fluxo de projeto de ASICs visto na aula 10. A Figura 2 apresenta uma ilustração deste fluxo de uma forma simplificada. Há uma etapa de síntese lógica onde uma descrição comportamental RTL (em VHDL, por exemplo) é transformada em um esquema lógico de blocos correspondentes a componentes típicos da arquitetura do FPGA. São eles: células combinacionais de LEs (memórias look-up-table/LUT de elementos lógicos) em modo normal ou

aritmético, registradores de LEs, etc (ver o pequeno manual da família Cyclone II). Estas tarefas são feitas dentro do conjunto *Analysis & Synthesis*.

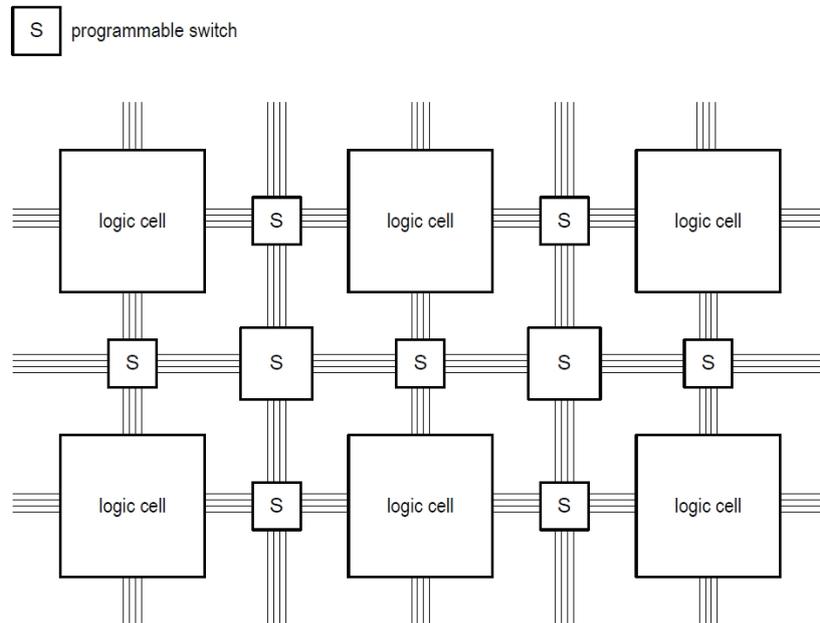


Figura 1. Visão simplificada da arquitetura de um FPGA

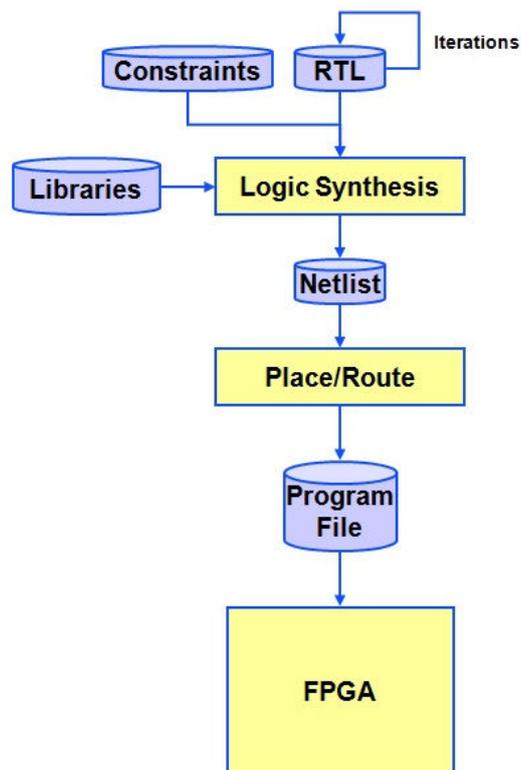


Figura 2. Fluxo de projeto

Após a etapa de síntese lógica, o esquema lógico é utilizado como entrada das tarefas de Place/Route, que também existe no fluxo de projeto de ASICs. O conjunto de tarefas são um tanto diferentes entre os casos ASICs e FPGAs. Para os primeiros, o leiaute do CI está ainda para ser feito, deixando o projetista livre para escolher uma planta-baixa om razão de aspecto livre (*floorplanning*) e dentro deste escolher onde posicionar as células (*placement*). No caso dos FPGAs, o chip já está fabricado e todos os LEs já estão dispostos nele. Desta forma, uma vez escolhido o dispositivo dentro da família, a tarefa de posicionamento consiste em definir para cada bloco do esquema lógico em qual LE físico do dispositivo alocar. Esta tarefa é feita pelo **Fitter**, que pode modificar o número de blocos originais (duplicando-os ou reduzindo-os) de modo a ajustar os atrasos no circuito. Os algoritmos do **fitter** também cuidam de otimizar os tempos de atraso na programação das chaves de interconexão.

A última etapa prevista na síntese do circuito em FPGA consiste na escolha/definição dos pinos do dispositivo aos quais as entradas e saídas do projeto (que já estão alocados nos seus LEs correspondentes) serão associados. Esta é uma tarefa do **Assembler**, que também gera o arquivo de programação/configuração final do FPGA.

A Tabela 1 lista comparativamente as diversas tarefas e representações resultantes nos fluxos ASIC e FPGA, realizados respectivamente pelo Leonardo Spectrum e Quartus.

Tabela 1- Quadro comparativo dos fluxos do Quartus e Leonardo Spectrum

Tarefa e Objetos	Fluxo Quartus	Fluxo Leonardo Spectrum
Circuito a ser sintetizado	RTL comportamental (VHDL)	RTL comportamental (VHDL)
Síntese Independente de Tecnologia	Função Analysis & Elaboration	Função Read
Biblioteca Independente de Tecnologia	Primitives e Operators (muxes, adders, registers, latches, etc.)	Primitives e Operators (logic cells, registers, incrementers, etc.)
Resultado de síntese	Representação interna (esquema de primitivas e operadores)	Representação interna (esquema de primitivas e operadores)
Síntese Dependente de Tecnologia	Realizado em Analysis & Synthesis	Optimize
Restrição para a Síntese Dependente de Tecnologia	Definição da Família do FPGA	Definição da foundry
Biblioteca de Tecnologia	LUTs de LEs (modo normal ou aritmético), registradores de LEs	Design kit - exemplo TSMC 0.36um ADK
Resultado da síntese lógica	Representação interna (esquema de LUTs, registradores, etc.)	Arquivo VHDL (gate-level)
Placement/Routing	Fitter	Ferramenta de Terceiros (ex. ICStation- P&R- Mentor)
Restrição para a tarefa	Definição do Dispositivo da Família	Células de leiaute (ex. ADK)
Resultado de placement e routing	Todos os blocos alocados em LEs/LABs físicos da FPGA e conectados (roteamento define os caminhos de cada conexão)	Células posicionadas em fileiras e conectadas (roteamento define os caminhos de cada conexão)
Finalização de Projeto	Assembler : 1- definição de I/Os 2- geração de arquivo de programação	Timing Closure Geração de máscaras