

## Laboratório sobre Introdução a Sistemas de CAD, Projeto com Esquemáticos e Circuitos Combinacionais

Prática: Implementação de somador de 8 bits – captura, hierarquia e simulação funcional

Recursos: Sistema de CAD Foundation da empresa Xilinx, Inc.

### Parte I - Objetivos

O objetivo estratégico deste laboratório é fornecer aos alunos ferramental básico usado no projeto, validação e implementação de sistemas digitais mediante emprego de software do tipo sistema de Projeto Auxiliado por Computador (PAC, do inglês, *Computer Aided Design*, ou CAD) para sistemas digitais eletrônicos (em inglês, Electronic CAD ou ECAD). Este conjunto de ferramentas é fundamental para o restante da disciplina. Ao final deste laboratório o aluno deve estar apto a usar os recursos básicos de um sistema comercial de CAD eletrônico para realizar a captura de projeto e a validação do mesmo através de simulação no nível lógico de abstração, bem como dominar os princípios de projeto hierárquico no nível lógico de abstração de sistemas digitais. O objetivo específico aqui é implementar de forma modular e hierárquica um somador de 8 bits, a partir de somadores de 1 bit. Será enfatizado durante a aula o processo de simulação lógica *funcional* (sem atraso nas portas lógicas) e utilização de *scripts*.

### Parte II - Projeto de um somador completo de um bit (em inglês, full-adder ou FA)

O circuito somador completo de 1 bit é capaz de somar três bits quaisquer, possuindo 3 entradas: A, B e Cin (Carry in, ou vai-um de entrada) e duas saídas: Soma e Cout (Carry-out, ou vai-um de saída).

Tarefa 1 - Determinar as equações do FA, utilizando tabela verdade e mapas de Karnaugh. **Faça isto agora!** Documente seu projeto de equações e coloque-o no relatório final.

Tarefa 2 – Use as orientações abaixo para realizar a implementação do diagrama de esquemáticos do seu somador de 1 bit.

1. Abrir o FOUNDATION (Figura 1).



Foundation.Ink

Ícone: Se não houver este ícone no desktop, executar `c:/fndtn/active/exe/pcm.exe`. Caso não esteja instalado na máquina, chamar o professor.

- seta 1: indica chamada ao editor de portas lógicas ou editor de esquemáticos.
- seta 2: indica chamada ao simulador de portas lógicas ou simplesmente simulador lógico.

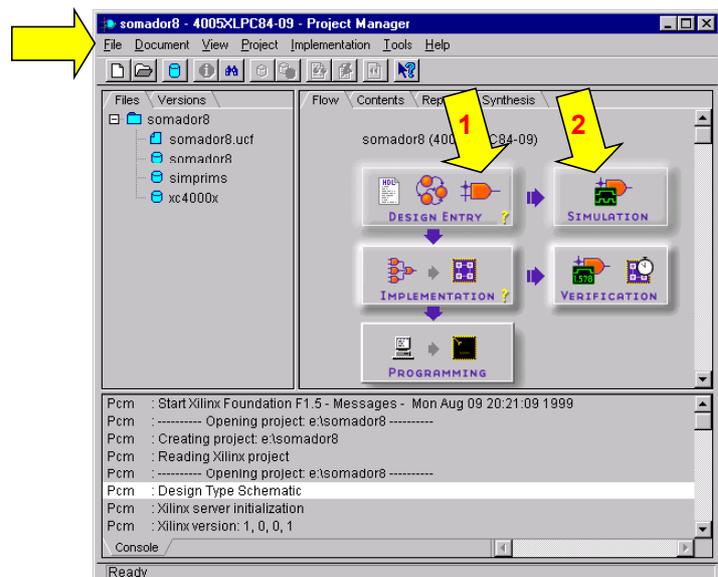


Figura 1 – Janela Principal do Gerenciador de Projeto.

2. Abrir um novo projeto (Figura 2): → **File** → **New project** (o símbolo → indica menus e/ou submenus)

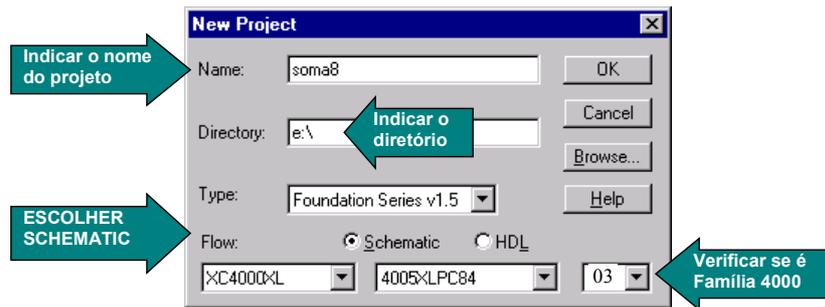


Figura 2 - Criação de um novo projeto.

MUITO IMPORTANTE:

- Trabalhar sempre no disco local (c:) (evite o disco h:, pois este é muito lento. Após o projeto realizado é possível armazenar o projeto em um arquivo .zip e guardá-lo no seu disco h:). Não utilizar disquete, exceto para fazer cópias de salvaguarda ao final do trabalho.
- Verificar se o dispositivo corresponde ao dispositivo da placa (família XC4000XL, componente 4005XLPC84 velocidade -3)

3. Clicar no botão "Schematic Editor" da janela principal (ver seta 1 - Figura 1).
4. Agora, a partir das equações, implemente o circuito utilizando a ferramenta editor de esquemáticos. Os comandos a serem utilizados são de 3 classes: inserção de símbolos, inserção de fios e inserção de pinos (os pontos de entrada e saída do circuito, ou seja, sua conexão ao mundo exterior). A janela inicial do editor é apresentada na Figura 3.

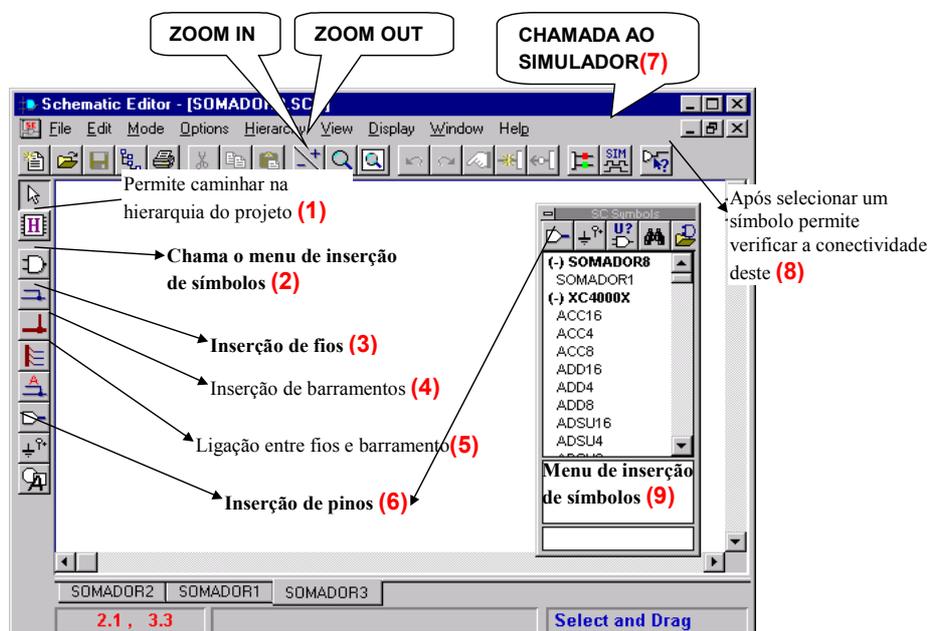


Figura 3 - Editor de Esquemático

Para criar o esquemático:

- insira as portas lógicas assim: clique no botão (2), no menu (9), selecione a porta lógica desejada ou escreva seu nome na parte inferior do menu.
- Insira os pinos de entrada/saída do somador, via botão (6). A janela da Figura 4 deve ser preenchida para cada pino.

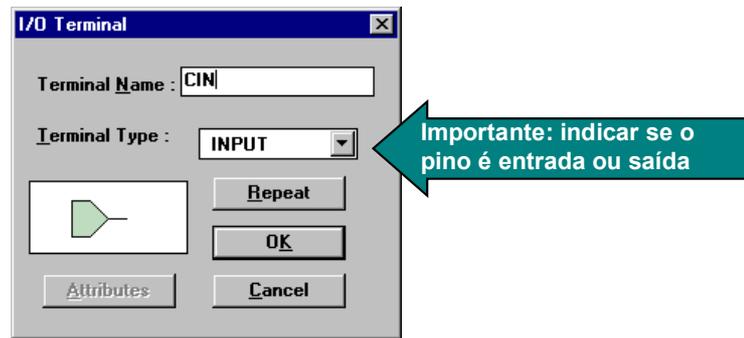


Figura 4 - Inserção de Pino.

- c. Realize as conexões entre as portas lógicas e os pinos, clicando no botão (3) - Figura 3, depois na primeira extremidade do fio, arrastando-se o mouse até a extremidade seguinte (podes clicar em pontos intermediários da conexão para realizar quebras), terminando-se com um clique na extremidade final da conexão.
- 5. Uma vez concluído o esquemático do somador completo (*full adder*), chamar o simulador via o botão (7) no editor de esquemático ou via seta 2 no gerenciador de projeto.

### Parte III - Simulação Lógica com a Ferramenta Foundation

- **Objetivo:** verificar se o comportamento do FA corresponde à especificação original, usando simulação exaustiva.

Tarefa 3 – Use as orientações abaixo para realizar a simulação do diagrama de esquemáticos do seu somador de 1 bit.

Para realizar a simulação do FA, utilizaremos um *script*. O script permite definir: (1) quais sinais serão definidos como entrada; (2) quais sinais serão observados e (3) o tempo de simulação.

Para invocar o editor de script selecione:

**tools → script editor → create empty script**

O script da Figura 5 exemplifica os comandos necessários à simulação do FA.

Tarefa 4 – Crie um arquivo contendo o script abaixo. Observe que as palavras reservadas são escritas em vermelho. Para conhecer a função de cada uma, simplesmente selecione o comando e pressione F1. Usar o help!

```

;simulacao relativa ao somador full-adder
delete_signals
restart
stepsize 10 ns

wfm a 0ns=0 (5ns=1 5ns=0)*1000
wfm b 0ns=0 (10ns=1 10ns=0)*1000
wfm cin 0ns=0 (20ns=1 20ns=0)*1000

watch sum
watch cout

sim 200ns
    
```

**COMENTÁRIO**

**comandos de inicialização**

**formas de onda (entradas)**

**sinais de saída**

**tempo de simulação**

**USEM O HELP!**

Figura 5 - Script para simulação do somador completo.

O comando básico para definir formas de onda periódicas é *waveform* ou *wfm*. A sintaxe do comando é: *wfm* <nome do sinal> <tempo inicial>=<valor> (<intervalo1>=<valor> <intervalo2>=<valor> <intervalo3>=<valor> .....)\*<número de repetições>

Uma vez escrito e armazenado o script, deve-se executá-lo. Para isto, fechar o editor de scripts e no simulador lógico fazer: **file** → **run script file** → [nome dado na edição do script]

Tarefa 5 - Feche ou minimize a janela correspondente ao resultado da execução do script, e analise as formas de onda geradas (Figura 6). A saídas geradas pelo *cout* e *sum* devem corresponder à tabela verdade.

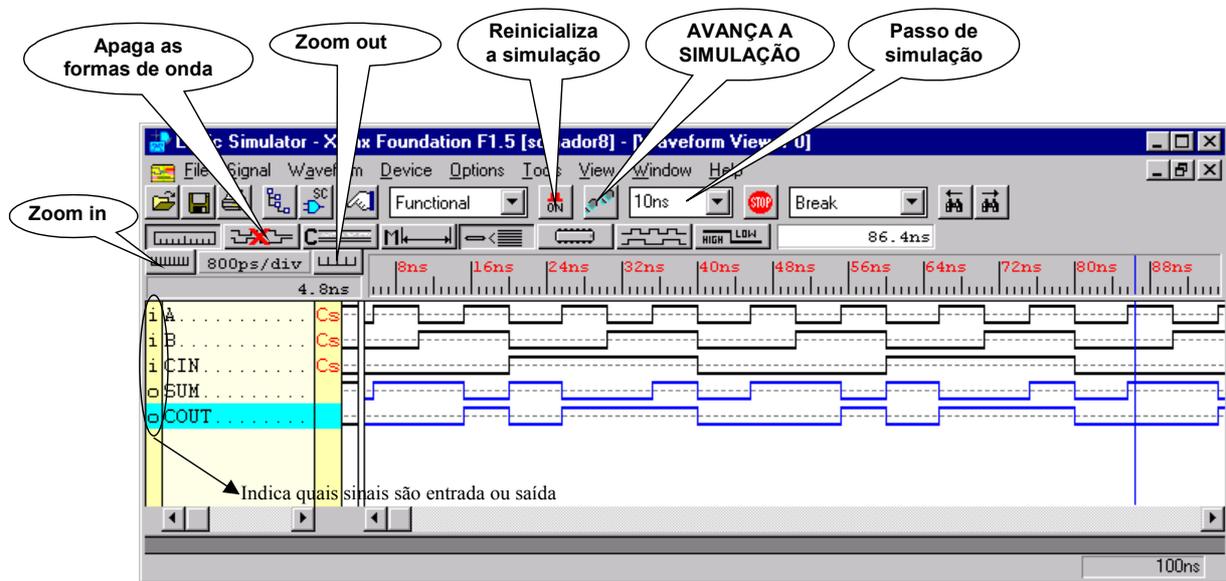


Figura 6 - Simulação do somador completo

Algumas dicas úteis:

1. Para reinicializar a simulação clicar no botão correspondente a esta função, e depois pode-se executar a simulação passo a passo no botão correspondente.
2. Para apagar as formas geradas antes de reinicializar uma simulação também há um botão correspondente.
3. Para visualizar um sinal interno do circuito, não definido no script. Ir no editor de esquemáticos e colocar um *probe* (ponteira) sobre o sinal desejado - Figura 7. No editor de esquemático é inserido um retângulo sobre o sinal, e no simulador um novo sinal. Reinicialize a simulação, apague as formas de onda e execute passo a passo a simulação. **FAÇA ESTE PROCEDIMENTO PARA OS SINAIS INTERNOS DO SOMADOR. ESTE É UM PROCEDIMENTO IMPORTANTE PARA DEPURAR CIRCUITOS COMPLEXOS.**

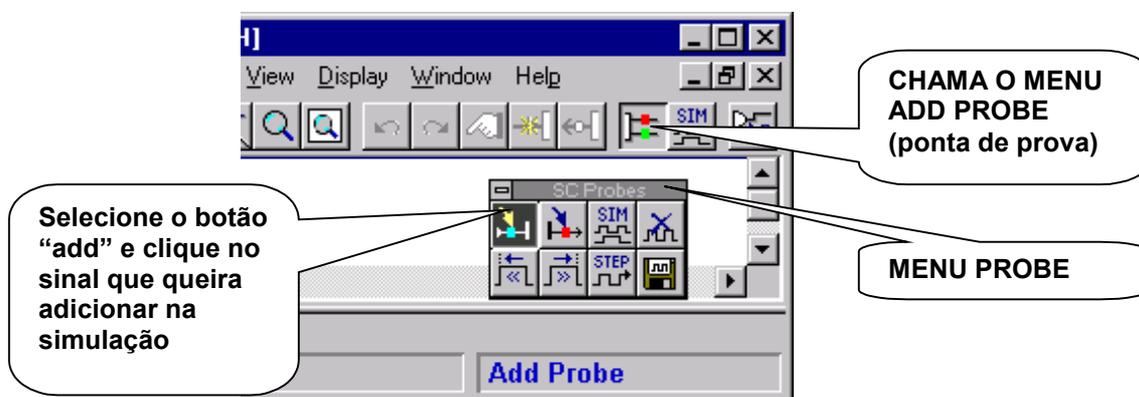


Figura 7 - Inserção de pontas de prova (probes).

Feche o simulador lógico e inicie a parte 3 do trabalho.

## Parte IV - Hierarquia: somador de 8 bits

Tarefa 6 – A partir do diagramas de esquemáticos do somador de 1 bit (FA) implemente um somador de 8 bits de forma hierárquica. Siga os passos abaixo.

1. A partir do esquemático do FA, criar um símbolo referente a este para ser utilizado pelo esquemático pai (somador de 8 bits).
  - Escolha a opção de menu (**Hierarchy** → **Create Macro Symbol from Current Sheet**). A janela da Figura 8 é apresentada, e normalmente deve-se apenas clicar em OK.

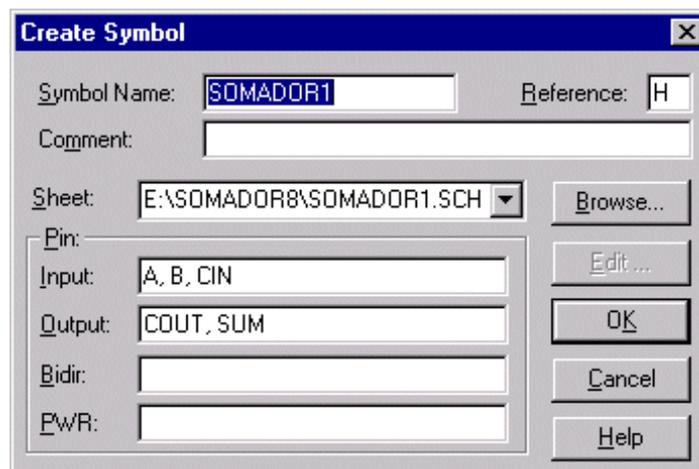
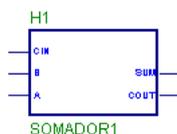
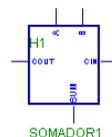


Figura 8- Criação de um símbolo a partir de uma folha de esquemáticos.

- Você pode editar as características do novo símbolo (nome, comentário, etc), clicando em seguida sobre o botão OK. Dica: responda *não* à pergunta “queres editar símbolo?” após o OK.
  - Agora, existe um novo símbolo que representa este circuito disponível na biblioteca, com o nome indicado na opção *symbol name*.
2. Abrir nova folha, (opção de menu File → New Sheet do editor de esquemáticos) e inserir nela OITO (8) células iguais, instâncias do símbolo que você acabou de criar no item 1. Para colocar as instâncias do somador recém criado, ir no menu de inserção de símbolos (botão 2 - Figura 3), e selecione o somador pela lista ou escreva seu nome na parte inferior do menu.
  3. Como a disposição das entradas e saídas dificulta a edição de esquemático, podemos modificar o símbolo gerado automaticamente para o somador de um bit, afim de simplificar a edição (Figura 9). Utilizar para isto a ferramenta de editor de símbolos.



Símbolo original do somador



Modificar o símbolo, clicando duas vezes no símbolo original, selecionado a opção *symbol editor*. No editor de símbolos colocar as entradas na parte superior, cin na direita alinhado com o cout na esquerda e a saída sum na parte inferior. Salvar o símbolo e voltar para o editor de esquemático.

Figura 9 - Modificação de um símbolo através do editor de símbolos.

4. O esquemático neste ponto deve ser semelhante ao esquemático da Figura 10.

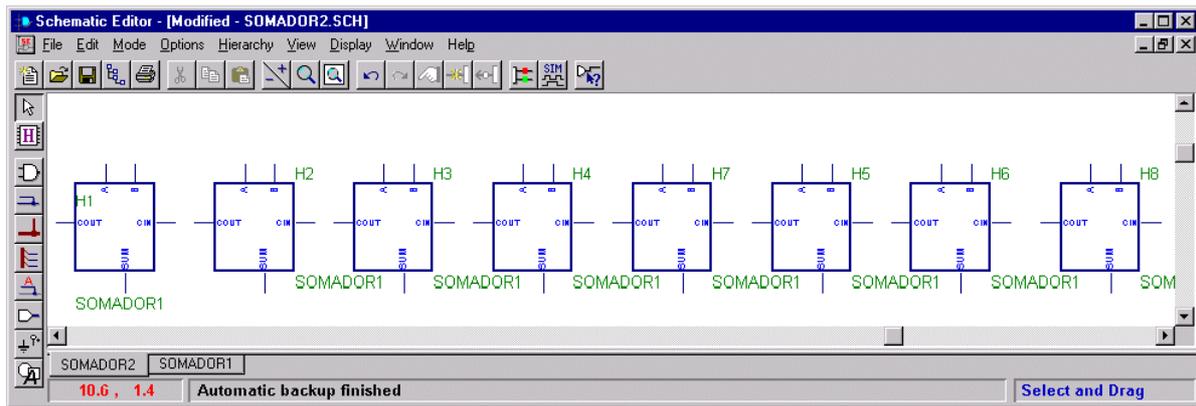


Figura 10 – Esquemático do somador de 8 bits, com 8 instâncias do FA.

5. Realizar as conexões relativas a propagação do vai um (carry out).
6. Inserir os pinos Cin e Cout.
7. Inserir os barramentos de 8 bits referentes as entradas A e B, e a saída Soma (botão 4 - Figura 3). Atenção à direção dos sinais (input/output). Procedimento: um clique no início do barramento e um duplo clique no final deste. Após o duplo clique aparecerá a janela da Figura 11, devendo-se indicar: nome do barramento, direção e número de bits.

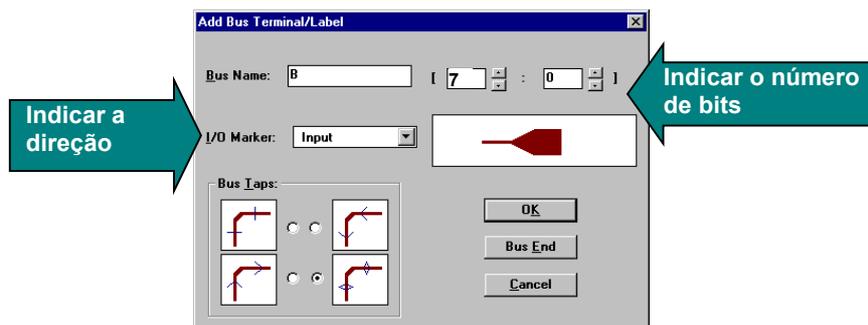


Figura 11 - Janela de inserção de barramento.

Neste ponto, o esquemático deve ser semelhante ao esquemático da Figura 12.

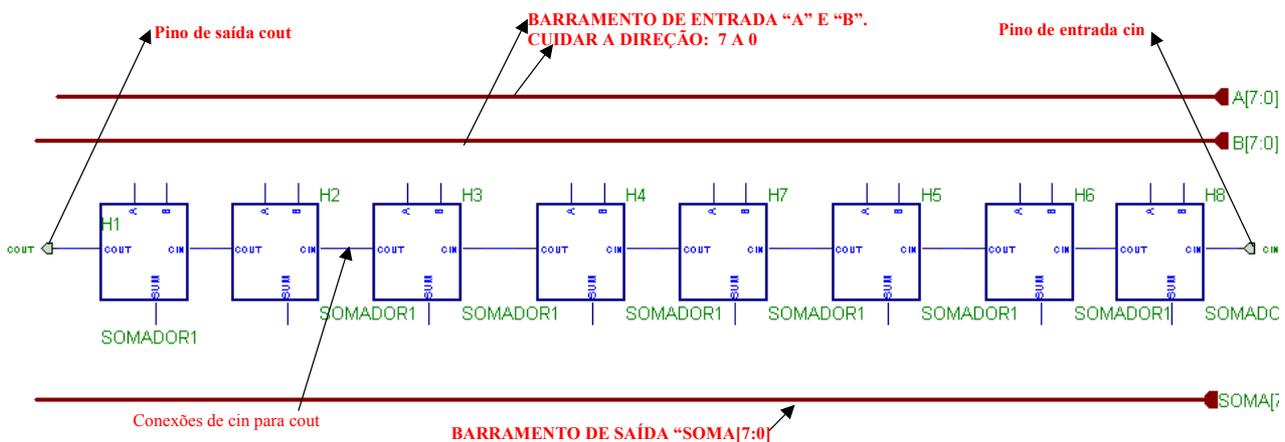


Figura 12 - Esquemático do somador de 8 bits, com inserção dos barramentos.

8. Após a inserção dos barramentos, ligar os fios dos FA aos barramentos.

**\*\* parte mais propensa a erros \*\***

Procedimentos:

**Método 1:** clicar no botão (5), clicar no barramento para selecioná-lo (**observar a mensagem “expand: bus tap A”** na parte inferior da janela), clicar nos pinos dos FA. A conexão se faz automaticamente.

**Método 2,** caso o método 1 não realize a conexão automaticamente: insira um fio entre o pino e o barramento, clique no botão (5), clicar no barramento para selecioná-lo (**observar a mensagem “expand: bus tap A”** na parte inferior da janela), selecionar o sinal (A0, A1, A2, ..) com os teclas  $\uparrow$  ou  $\downarrow$ , clicar no pino do FA. Observar o texto que será inserido sobre a conexão.

- Uma dica é, após a realização das conexões, verificá-las utilizando o botão do editor correspondente (procedimento: selecionar o símbolo e depois clicar no botão de conectividade).

O resultado final é apresentado na Figura 13.

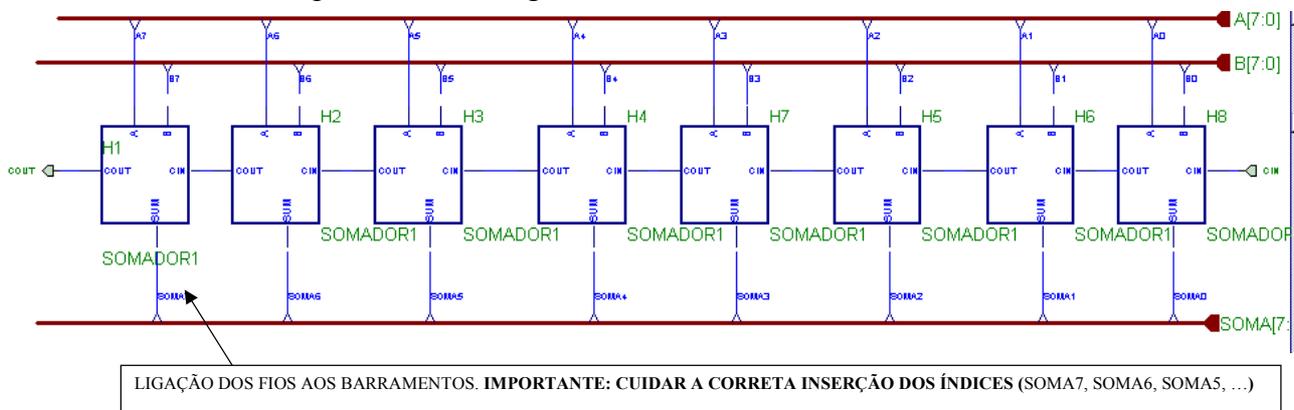


Figura 13 - Esquemático final do somador de 8 bits.

## Parte V – SIMULAÇÃO DO SOMADOR DE 8 BITS

Tarefa 7 – Simule o somador de 8 bits projetado anteriormente. Siga os passos abaixo.

1. Deve-se primeiro definir o *script* para simular este circuito. Um possível script é apresentado na Figura 14.

```
; simulacao relativa ao projeto somador de 8 bits
delete_signals
restart
stepsize 10 ns

; define os vetores a serem visualizados
vector A      a[7:0]
vector B      b[7:0]
vector SOMA   soma[7:0]

wfm A 0ns=10\H 40ns=15\H 40ns=30\H 40ns=50\H 40ns=60\H 40ns=05\H 40ns=09\H
wfm B 0ns=30\H 40ns=AB\H 40ns=FF\H 40ns=01\H 40ns=8C\H 40ns=8E\H 40ns=EE\H
wfm cin 0ns=0 140ns=1

watch cout
sim 280ns
```

Figura 14 - Script de simulação para o somador de 8 bits.

2. O resultado esperado para esta simulação é apresentado na Figura 15 (a interpretação desta simulação deverá constar no relatório):

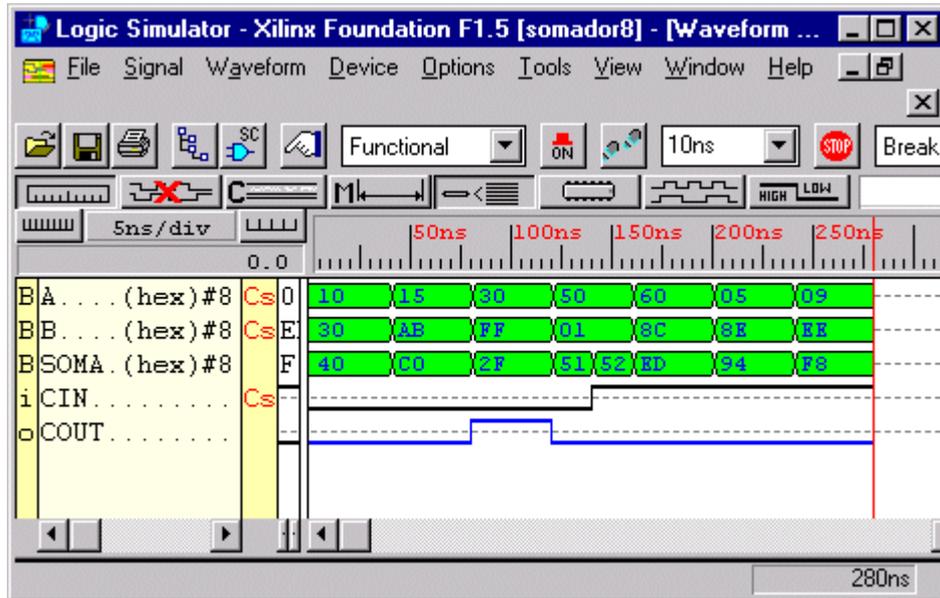


Figura 15 - Janela de simulação para o somador de 8 bits.

## Parte VI – A Fazer e Entregar

Execute todas as [Tarefas](#) propostas acima. Elabore um relatório contendo todas as suas simulações, bem como a resposta a todas as questões que eventualmente aparecem no texto. Entregue este relatório no prazo especificado na homepage da disciplina. Procure mostrar as operações que o grupo fez ao professor, em aula, o que contribuirá para a nota de participação em aula. Além das Tarefas específicas acima faça, antes de entregar o material.

[Tarefa 8](#) - Inserir no esquemático a lógica e um pino que identifica transbordo (overflow). Não se esqueça que o conceito de transbordo apenas faz sentido se se interpretam as entradas do somador como números representados em complemento de 2.

[Tarefa 9](#) – O grupo de alunos deve considerar no script casos onde haverá/não haverá vai um e transbordo, indicando na simulação estes casos.

No seu relatório coloque as dificuldades encontradas para realizar este estudo dirigido, tais como instruções pouco claras, erros de digitação, passos omitidos, etc. Guarde o projeto para próximas aulas.

Percentuais de nota atribuídos às Tarefas:

Item Avaliado	Percentual
Somador de 1 bit – Projeto, simulação e documentação	30%
Somador de 8 bits – Projeto simulação e documentação	40%
Inserção de Transbordo – Implementação simulação e documentação	30%