

Simulador de Lógica Digital

Este enunciado é deliberadamente aberto e omissivo em relação a alguns pormenores como por exemplo detalhes relativos ao interface de utilizador, tipo de ações que podem sujeitas a undo/redo. O cumprimento das restrições e indicações de implementação é mais relevante do que esse tipo de detalhes, que ficam em aberto. Quaisquer dúvidas podem ser esclarecidas junto do Professor.

Resumo

Pretende-se construir um sistema de desenvolvimento e simulação de sistemas digitais. Este sistema deverá permitir aos seus utilizadores criarem e editarem modelos que descrevem circuitos digitais. O funcionamento destes modelos em seguida ser testado através de simulação. Nesta fase de desenvolvimento do sistema, pretende-se apenas abordar os circuitos assíncronos.

Módulos

O sistema deve permitir a criação modelos de sistemas digitais tal como o apresentado na Fig. 1. Nesta é descrito o módulo com nome “Teste”, com três entradas (A,B,C) e duas saídas (X,Y)¹. Cada projecto de um novo circuito é concretizado através da criação de um módulo adequado. Os módulos podem, por sua vez, ser utilizados de forma hierárquica para criar novos módulos, tal como apresentado na Fig. 2.

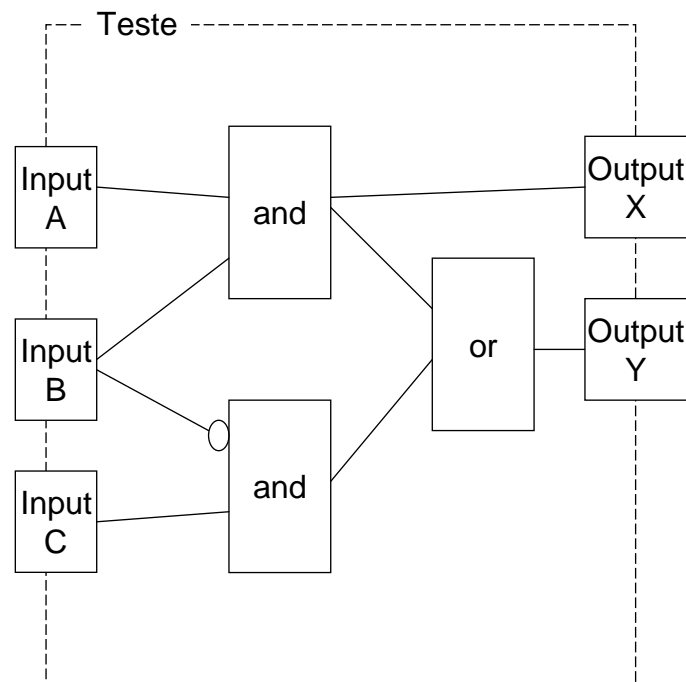


Fig. 1 – Módulo de teste

¹ A aparência gráfica dos componentes é meramente indicativa: podem ser adoptadas outras soluções.

Projecto

Um projecto é, essencialmente, um conjunto de módulos. Um dos módulos, e apenas um, pode ser definido como o módulo Global, representando o circuito completo. Os outros módulos serão utilizados para realizar o circuito descrito no módulo global. Os módulos podem ter dependências hierárquicas, mas não circulares.

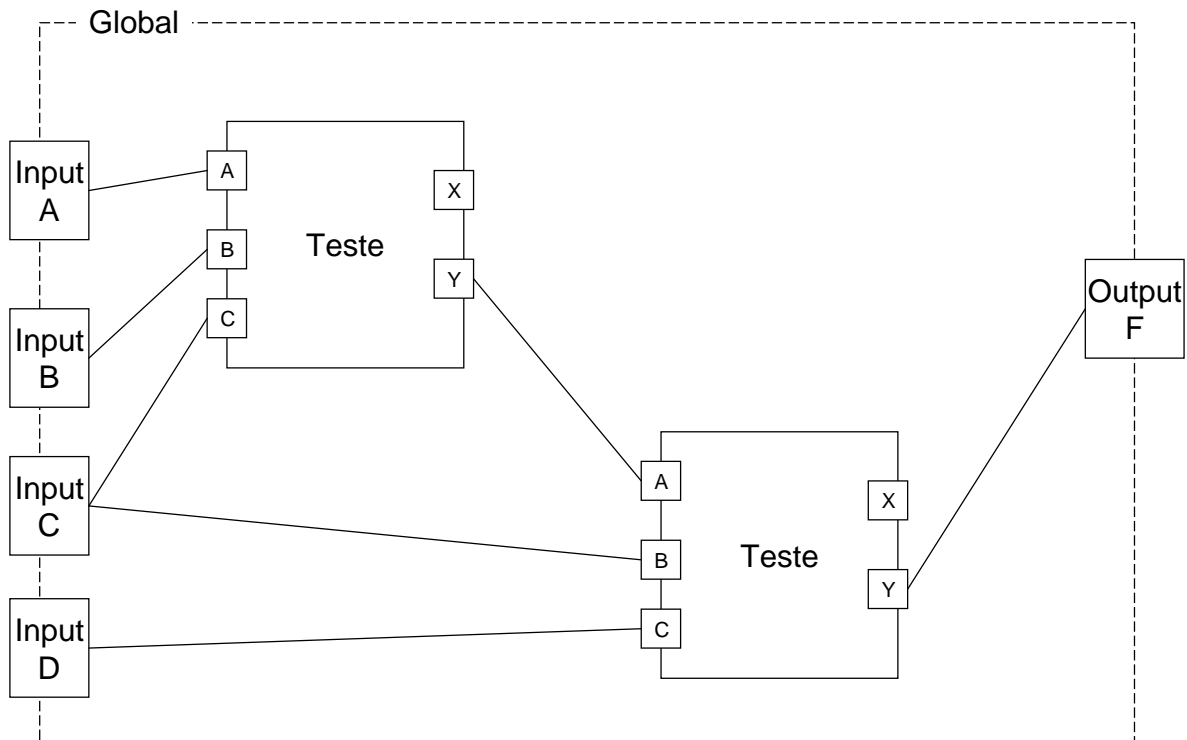


Fig. 2 – Exemplo de Módulo Global

Funcionalidades

O sistema deverá oferecer um interface de utilizador adequado para criar e editar os projectos e módulos, bem como correr as simulações e testes. O editor deverá suportar o undo/redo de operações de edição, bem como suportar diferentes algoritmos para desenho de ligações entre componentes (ver adiante).

O sistema deve solicitar ao utilizador a sua identificação no início de cada sessão. Esta informação é usada (apenas) para gerir o histórico de ficheiros recentes do utilizador. Esta informação (nomes de utilizadores e respetivo historico de ficheiros) deve ser persistida na BD.

O projeto do utilizador pode ser guardado num formato próprio, a ser definido pelos alunos.

O sistema também deve permitir a exportação e importação a partir de ficheiros em formato BLIF (*Berkeley Logic Interchange Format*). A especificação deste formato pode ser encontrada aqui:

<https://www.cse.iitb.ac.in/~supratik/courses/cs226/spr16/blif.pdf>

Não é necessário considerar os aspetos do formato BLIF que não são aplicáveis a este trabalho (por exemplo, máquinas de estado ou relógio).

Como exemplo, o circuito da Fig.1 poderia ser descrito na notação BLIF através de :

```

.model Teste
.inputs A B C
.outputs X Y

.names A B and1
11 1
.names B C and2
01 1
.names and1 and2 or1
1- 1
-1 1
.names and1 X
1 1
.names or1 Y
1 1
.end

```

O sistema deve permitir a simulação dos circuitos. Esta capacidade é importante para verificar o comportamento correcto dos mesmos.

O primeiro passo para a simulação é a definição de um *signal* de entrada para o módulo em teste. Como exemplo da informação contida num sinal que percorre todas as combinações de entrada possíveis para o módulo da Fig. 1, temos:

```

A: 01010101
B: 00110011
C: 00001111

```

A aplicação deve permitir a criação/edição de sinais de forma adequada, através de um interface adequado. Estes sinais podem ser persistidos a nível do módulo, para posterior reutilização.

Estando definido um sinal aplicável à entrada do módulo, o utilizador pode solicitar a simulação do comportamento do circuito. O resultado será um outro sinal que descreve o resultado do mesmo, por exemplo:

```

X: 00010001
Y: 00011101

```

A ferramenta deverá permitir ao utilizador a visualização do sinal de saída de forma adequada.

O utilizador pode definir testes de validação. Um teste consiste num par de sinais que representam a entrada e saída esperada. O teste é feito sob pedido do utilizador, aplicando a entrada ao circuito e verificando se a saída obtida corresponde à saída esperada armazenada na definição do teste. Os testes são gravados no projeto, conjuntamente com os vários módulos, para posterior reutilização.

Restrições de Implementação

O sistema deve estar estruturado em camadas, Na implementação deste trabalho, deverão ser utilizados os seguintes padrões da forma indicada:

Estratégia: Deve ser utilizada uma estratégia para implementar a ligação entre os diversos componentes no interface com o utilizador. Há pelo menos duas estratégias possíveis:

- *Ligação Direta:* Uma linha reta entre os dois pinos a ligar
- *Ligação Ortogonal:* Um ligação composta por 3 segmentos de reta (2x verticais e um horizontal ou um vertical e 2x horizontal)

Adaptador: Devem ser usados adaptadores para ler informação de um projeto (quer seja em formato próprio ou formato BLIF).

Builder: Devem ser usados *builders* para exportar/gravar informação de um projeto para formato próprio ou formato BLIF, assim como para gerar um relatório em formato html sobre o resultado dos testes de validação do projeto.

Comandos: Devem ser usados comandos para implementar as funcionalidades de undo/redo das operações de edição do circuito.

Camadas e Fachada: Deve ser utilizada uma arquitetura em camadas, com separação clara entre interface e lógica Deve existir uma fachada de acesso à lógica do sistema.

Fábricas de Objetos: Devem ser usadas fábricas de objetos quando apropriado.

Máquina de Estados: Deve ser usada uma máquina de estados para validar o sequenciamento de pelo menos parte das operações de edição do utilizador.

Grupos e Entrega do Trabalho

O trabalho deve ser realizado por grupos de três alunos, sendo sujeito a defesa onde deverão estar presentes todos os elementos do grupo. A defesa é marcada em data a combinar individualmente entre o professor e cada grupo. O trabalho pode ser entregue em qualquer altura deste ano letivo, desde que respeitando as regras em vigor. Após a sua entrega e defesa, nota do trabalho será lançada na próxima época de avaliação disponível (do mesmo ano letivo) à qual o aluno tenha acesso.

O trabalho (código fonte, executáveis) deverá ser enviado ao professor através do moodle ou por email. (pode ser enviado um *link* para *cloud* onde se encontre o trabalho)