**EA773 Laboratório de Circuitos Lógicos**

**Relatório Roteiro 2**

**Autor:** Mateus Henrique Silva Araújo

**RA: 184940**

**Data:** 27/08/2022

1. **Projeto do módulo “complementa”:**
   1. **Escopo:**

Projeto de um circuito combinacional capaz de realizar o complemento bit-a-bit de uma entrada de 4 bits, mediante a permissão de uma entrada de controle (enable).

* 1. **Especificação de alto-nível:**

*Entradas:*

com e ;

(entrada enable)

*Saídas:*

come *;*

*Função:*

; com

* 1. **Especificação binária:**

*Entradas:*

com e ;

(entrada enable)

*Saídas:*

come

*Função de saída: Tabela verdade*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| 0 | 0 | 0 | 0 |  |  |  |  |
| 0 | 0 | 0 | 1 |  |  |  |  |
| 0 | 0 | 1 | 0 |  |  |  |  |
| 0 | 0 | 1 | 1 |  |  |  |  |
| 0 | 1 | 0 | 0 |  |  |  |  |
| 0 | 1 | 0 | 1 |  |  |  |  |
| 0 | 1 | 1 | 0 |  |  |  |  |
| 0 | 1 | 1 | 1 |  |  |  |  |
| 1 | 0 | 0 | 0 |  |  |  |  |
| 1 | 0 | 0 | 1 |  |  |  |  |
| 1 | 0 | 1 | 0 |  |  |  |  |
| 1 | 0 | 1 | 1 |  |  |  |  |
| 1 | 1 | 0 | 0 |  |  |  |  |
| 1 | 1 | 0 | 1 |  |  |  |  |
| 1 | 1 | 1 | 0 |  |  |  |  |
| 1 | 1 | 1 | 1 |  |  |  |  |

* 1. **Minimizações:**



|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | | | | |  |  | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | | | | |  |  | | | | | |

* 1. **Esquemático do circuito:**

Diagrama, Esquemático

Descrição gerada automaticamenteA figura 1 apresenta o esquemático do módulo complementa, implementado usando as funções minimizadas acima.

Figura 1: Diagrama esquemático do módulo complementa

* 1. **Simulações:**

**Interface gráfica do usuário, Aplicativo, Word

Descrição gerada automaticamenteInterface gráfica do usuário, Aplicativo, Word

Descrição gerada automaticamente**Para as simulações, as entradas foram configuradas de modo a assumirem todas as configurações possíveis. A simulação funcional é apresentada na figura 2 e a simulação temporal, na figura 3.

Figura 3: simulação temporal do módulo complementa

Figura 2: simulação funcional do módulo complementa

1. **Projeto do módulo “zera”:**
   1. **Escopo:**

Projeto de um combinacional capaz de receber uma entrada de 4 bits e zerá-la, mediante a permissão de uma entrada de controle (enable).

* 1. **Especificação de alto-nível:**

*Entradas:*

com e ;

(entrada enable);

*Saídas:*

come *;*

*Função:*

; com

* 1. **Especificação binária:**

*Entradas:*

com e ;

(entrada enable);

*Saídas:*

come

*Função: Tabela verdade*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |  |
| 0 | 0 | 1 | 0 | 0 | 0 |  | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 |  |  |
| 0 | 1 | 0 | 0 | 0 |  | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |  | 0 |  |
| 0 | 1 | 1 | 0 | 0 |  |  | 0 |
| 0 | 1 | 1 | 1 | 0 |  |  |  |
| 1 | 0 | 0 | 0 |  | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 |  | 0 | 0 |  |
| 1 | 0 | 1 | 0 |  | 0 |  | 0 |
| 1 | 0 | 1 | 1 |  | 0 |  |  |
| 1 | 1 | 0 | 0 |  |  | 0 | 0 |
| 1 | 1 | 0 | 1 |  |  | 0 |  |
| 1 | 1 | 1 | 0 |  |  |  | 0 |
| 1 | 1 | 1 | 1 |  |  |  |  |

* 1. **Minimizações:**



|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | | | | |  |  | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | | | | |  |  | | | | | |

* 1. **Esquemático do circuito:**

A figura 4 apresenta o esquemático do módulo zera, implementado usando as funções de minimização acima.

Uma imagem contendo Interface gráfica do usuário

Descrição gerada automaticamente

Figura 4: Diagrama esquemático do módulo zera

* 1. **Simulações:**

**Interface gráfica do usuário, Aplicativo, Word

Descrição gerada automaticamenteInterface gráfica do usuário, Aplicativo, Word

Descrição gerada automaticamente**Para as simulações, as entradas foram configuradas de modo a assumirem todas as configurações possíveis. A simulação funcional é apresentada na figura 5 e a simulação temporal, na figura 6. Nota-se que, em ambos os casos, o circuito obedece ao comportamento especificado anteriormente, zerando a entrada quando a entrada é igual a .

Figura 6: Simulação temporal do módulo zera

Figura 5: Simulação funcional do módulo zera

1. **Projeto do módulo “somador\_4\_bits”:**
   1. **Escopo:**

Projeto de um circuito combinacional, que recebe duas entradas de 4 bits, representando dois valores inteiros, e uma entrada de carry, capaz de efetuar a soma entre os dois valores da entrada (considerando o carry fornecido). As saídas devem ser um número de 4 bits (identificando o resultado da soma), bem como as bandeiras de estado (carry), (overflow), (negativo) e (zero).

* 1. **Módulos utilizados:**

Para a implementação da função de soma, particionaremos o somador em mais dois módulos, o módulo de meio somador e o módulo somador completo (que recebe dois bits e um carry, executa a soma e retorna tanto o resultado quanto o carry).

1. **Módulo “meio\_somador”:**

Circuito combinacional que recebe dois bits, executa a soma deles retorna e retorna tanto o resultado quanto o carry.

* 1. **Especificação binária:**

|  |  |
| --- | --- |
| *Entradas:* | *Saídas:* |
|  | (bit de geração) |
|  | (bit de propagação) |

|  |  |  |  |
| --- | --- | --- | --- |
| *Função: Tabela verdade* | | | |
|  |  |  |  |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

* 1. **Minimizações:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  | 0 | 0 |  |  | 0 | 1 |
|  | 0 | 1 |  |  | 1 | 0 |
|  | | |  |  | | |

* 1. **Esquemático:**

**Diagrama, Esquemático

Descrição gerada automaticamente**

Figura 7: Diagrama esquemático do módulo meio\_somador

* 1. **Simulações:**

**Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente**

Figura 8: Simulação funcional do módulo meio\_somador

**Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente**

Figura 9: Simulação temporal do módulo meio\_somador

1. **Módulo “somador\_completo”:**

Circuito combinacional que recebe dois bits e um carry, executa a soma deles retorna e retorna tanto o resultado quanto o carry. Para tal, empregaremos o módulo meio somador especificado anteriormente.

* 1. **Especificação binária:**

|  |  |
| --- | --- |
| *Entradas:* | *Saídas:* |
|  | (resultado da soma) |
|  | (carry out) |
|  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *Função: Tabela verdade* | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  |  |  |  | | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 1 | 1 | 0 | | 0 | 1 | 0 | 1 | 0 | | 0 | 1 | 1 | 0 | 1 | | 1 | 0 | 0 | 1 | 0 | | 1 | 0 | 1 | 0 | 1 | | 1 | 1 | 0 | 0 | 1 | | 1 | 1 | 1 | 1 | 1 | |

* 1. **Minimização:**

Usando o módulo meio somador, podemos implementar a tabela verdade acima. Para facilitar a compreensão das equações, vamos definir como a saída do circuito meio somador tendo como entradas e , bem como como a saída do circuito meio somador tendo como entradas e . Assim:

* 1. **Diagrama

     Descrição gerada automaticamenteEsquemático:**

Figura 10: Diagrama esquemático do módulo somador\_completo

* 1. **Interface gráfica do usuário, Aplicativo

     Descrição gerada automaticamenteSimulações:**

Figura 11: Simulação funcional do módulo somador\_completo

Figura 12: Simulação temporal do módulo somador\_completo

**Interface gráfica do usuário, Aplicativo, Tabela

Descrição gerada automaticamente**

* 1. **Especificação de alto-nível:**

*Entradas:*

com e ;

com e ;

(carry in)

*Saídas:*

come *;*

*; ; ;*

*Função:*

*;*

*; ;*

*;*

* 1. **Especificação binária:**

*Entradas:*

com e ;

com e ;

(carry in)

*Saídas:*

come *;*

*; ; ;*

*Função:*

Usando o módulo de somador completo especificado acima, podemos realizar a soma completa de cada bit considerando os dois bits de entrada e o carry da soma do bit menos significativo. Dessa forma, primeiramente definimos como a saída do somador completo tendo as entradas e , como a saída do somador completo tendo as entradas e e dado por :

Teremos que a saída , com será dada por:

Para as demais saídas, teremos:

, pois o carry produzido pela soma completa é igual ao carry do bit mais significativo;

, pois o bit mais significativo do resultado define se o resultado é negativo (caso ) ou positivo (caso);

, pois o resultado só é zero se todos os bits do resultado forem zero;

Para a bandeira de estado , como ela depende somente dos bits , e , tem-se a seguinte tabela verdade:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

* 1. **Minimizações:**

As saídas , , e já tiveram suas equações de saída reduzidas a partir de sua definição. Para a flag , fazemos:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  | 0 | 1 | 0 | 0 |
|  | 0 | 0 | 0 | 1 |

* 1. **Esquemático do circuito:**

Diagrama

Descrição gerada automaticamenteAssim como descrito anteriormente, o módulo do circuito “somador\_4\_bits” foi implementado utilizando o módulo “somador\_completo”. O circuito resultante está disposto na figura 13.

Figura 13: Diagrama esquemático do módulo somador\_4\_bits

* 1. **Simulações:**

**Interface gráfica do usuário, Aplicativo, Tabela

Descrição gerada automaticamenteInterface gráfica do usuário, Aplicativo, Word

Descrição gerada automaticamente**Como o circuito deste módulo apresenta 9 entradas e, portanto, possíveis combinações destas, não seria viável analisar todas as saídas geradas em tais casos. Portanto, como medida para contornar esse problema, uma tabela de valores de interesse foi montada e usada como entrada para a simulação (tabela 1). Os resultados estão expostos nas figuras 14 e 15.

Figura 15: simulação temporal do módulo somador\_4\_bits

Figura 14: simulação funcional do módulo somador\_4\_bits

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entradas | | | Saídas esperadas | | | | | *Tabela 1: combinações de interesse das entradas do somador e saídas esperadas para cada caso. As primeiras 8 combinações de entradas foram escolhidas de modo a contemplar todos os casos possíveis de ocorrências das flags. As demais foram tomadas aleatoriamente.* |
|  |  |  |  |  |  |  |  |
| 0100 | 0010 | 1 | 0111 | 0 | 0 | 0 | 0 |
| 0000 | 0000 | 0 | 0000 | 0 | 0 | 0 | 1 |
| 0011 | 1010 | 0 | 1101 | 0 | 0 | 1 | 0 |
| 0110 | 0010 | 1 | 1001 | 0 | 1 | 1 | 0 |
| 0011 | 1110 | 0 | 0001 | 1 | 0 | 0 | 0 |
| 0100 | 1011 | 1 | 0000 | 1 | 0 | 0 | 1 |
| 1100 | 1100 | 0 | 1000 | 1 | 0 | 1 | 0 |
| 1110 | 0000 | 0 | 1110 | 0 | 0 | 1 | 0 |
| 0001 | 0001 | 1 | 0011 | 0 | 0 | 0 | 0 |
| 0101 | 1000 | 0 | 1101 | 0 | 0 | 1 | 0 |
| 1001 | 0101 | 0 | 1110 | 0 | 0 | 1 | 0 |
| 0100 | 0001 | 1 | 0110 | 0 | 0 | 0 | 0 |
| 1001 | 1010 | 0 | 0011 | 1 | 1 | 0 | 0 |
| 1111 | 1111 | 1 | 1111 | 1 | 0 | 1 | 0 |
| 1000 | 0110 | 1 | 1111 | 0 | 0 | 1 | 0 |
| 0110 | 1010 | 1 | 0001 | 1 | 0 | 0 | 0 |

1. **Projeto do módulo “decodificador\_ULA”:**
   1. **Escopo:**

Projeto de um circuito combinacional capaz de atuar como decodificador da ULA construída. O circuito deve receber uma entrada seletora de 4 bits e um carry, enviando 4 sinais de controle como saída.

* 1. **Especificação de alto-nível:**

*Entradas:*

com e (entrada seletora);

(carry in);

*Saídas:*

*; ; ;*

*Função:*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Código () | Nome |  |  |  |  |
| 000x | ADD | 0 | 0 | 0 | 0 |
| 001x | ADDC | 0 | 0 | 0 |  |
| 010x | SUB | 0 | 1 | 0 | 1 |
| 0110 | INC | 0 | 0 | 1 | 1 |
| 0111 | DEC | 0 | 1 | 1 | 0 |
| 1000 | NEG | 1 | 0 | 1 | 1 |
| 1001 | CMPL | 1 | 0 | 1 | 0 |

* 1. **Especificação binária:**

*Entradas:*

com e (entrada seletora);

(carry in);

*Saídas:*

*; ; ;*

*Função: Tabela verdade*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 |  |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 |  |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | x | x | x | x |
| 1 | 0 | 1 | 1 | x | x | x | x |
| 1 | 1 | 0 | 0 | x | x | x | x |
| 1 | 1 | 0 | 1 | x | x | x | x |
| 1 | 1 | 1 | 0 | x | x | x | x |
| 1 | 1 | 1 | 1 | x | x | x | x |

* 1. **Minimizações:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | x | x | x | x |  |  | x | x | x | x |
|  |  |  | x | x |  |  |  |  | x | x |
|  | | | | |  |  | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | x | x | x | x |  |  | x | x | x | x |
|  |  |  | x | x |  |  |  |  | x | x |
|  | | | | |  |  | | | | |

* 1. **Esquemático do circuito:**

O circuito do módulo “decodificador\_ULA” foi implementado utilizando as funções minimizadas acima. Seu esquemático está exposto na figura 16.

Diagrama

Descrição gerada automaticamente

Figura 16: Diagrama esquemático do módulo somador\_4\_bits

* 1. **Simulações:**

**Interface gráfica do usuário, Aplicativo, Tabela

Descrição gerada automaticamenteInterface gráfica do usuário, Aplicativo, Tabela, Excel

Descrição gerada automaticamenteTela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente**Para as simulações, as entradas foram configuradas de modo a assumirem todas as configurações possíveis, até mesmo as consideradas inválidas para o circuito. A simulação funcional é apresentada na figura 17 e a simulação temporal, na figura 18. A figura 19 é uma simulação funcional adicional feita somente com as combinações das entradas de seleção válidas para o circuito.

Figura 19: Simulação funcional do módulo “decodificador\_ULA”, considerando somente combinações de entradas válidas

Figura 18: Simulação temporal do módulo “decodificador\_ULA”, considerando até combinações de entradas inválidas

Figura 17: Simulação funcional do módulo “decodificador\_ULA”, considerando até combinações de entradas inválidas

1. **Projeto do circuito ULA:**
   1. **Escopo:**

O circuito ULA é o projeto final deste relatório, pelo qual todos os demais módulos foram construídos. Esse circuito combinacional empregará instâncias de cada símbolo definido até então, devendo ser capaz de, a partir de uma entrada seletora de 4 bits, realizar uma determinada operação sobre duas entradas de 4 bits de operandos e uma entrada de 1 bit de carry.

* 1. **Especificação de alto-nível:**

*Entradas:*

com e ;

com e ;

(carry in);

*Saídas:*

come *;*

*; ; ;*

*Função:*

|  |  |  |  |
| --- | --- | --- | --- |
| Código () | Nome | Operação | Descrição |
| 000x | ADD |  | Adição de x com y |
| 001x | ADDC |  | Adição de x com y com carry |
| 010x | SUB |  | Subtração y de x |
| 0110 | INC |  | Incremente unitário de x |
| 0111 | DEC |  | Decremento unitário de x |
| 1000 | NEG |  | Negação aritmética de x em complemento de 2 |
| 1001 | CMPL |  | Complemento bit-a-bit de x |

* 1. **Especificação binária e Minimizações:**

*Entradas:*

com e ;

com e ;

(carry in);

*Saídas:*

come *;*

*; ; ;*

*Função:*

Para implementar as saídas desejadas do circuito ULA, serão empregues os módulos “complementa”, “zera”, “somador\_4\_bits” e “decodificador\_ULA”. Portanto, como medida de facilitar o entendimento de cada função, vamos definir primeiramente que:

* representa o vetor de 4 bits de saída do módulo “complementa”, tendo como entrada o vetor de 4 bits e o sinal de controle ;
* representa o vetor de 4 bits de saída do módulo “zera”, tendo como entrada o vetor de 4 bits e o sinal de controle ;
* representa o vetor de 4 bits de saída do módulo “somador\_4\_bits”, tendo como entrada os vetores de 4 bits e , bem como o bit de carry ;
* , , ou representam as flags de saída do módulo “somador\_4\_bits”, tendo como entrada os vetores de 4 bits e , bem como o bit de carry ;
* representa a saída do módulo “decodificador\_ULA”, tendo como entrada o vetor de 4 bits e o bit de carry ;
* representa a saída do módulo “decodificador\_ULA”, tendo como entrada o vetor de 4 bits e o bit de carry ;
* representa a saída do módulo “decodificador\_ULA”, tendo como entrada o vetor de 4 bits e o bit de carry ;
* representa a saída do módulo “decodificador\_ULA”, tendo como entrada o vetor de 4 bits e o bit de carry .

Dessa forma, as funções de cada circuito do projeto ULA podem ser descritos da seguinte maneira:

|  |
| --- |
|  |
|  |
|  |
|  |
|  |

* 1. **Esquemático do circuito:**

Interface gráfica do usuário

Descrição gerada automaticamente com confiança médiaO circuito foi implementado seguindo as funções descritas acima, o resultado está disposto na figura 20.

Figura 20: Diagrama esquemático do circuito ULA

* 1. **Simulações:**

Como o circuito recebe 13 entradas e, portanto, apresenta possíveis combinações destas, não seria viável analisar todas as saídas geradas em tais casos. Portanto, cada funcionalidade implementada foi testada separadamente com um conjunto de valores de entrada de interesse. Os conjuntos de valores testados e resultados esperados estão dispostos nas tabelas 2 a 8. Os resultados das simulações estão expostos nas figuras 21 a 36.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teste da função ADD | | | | | | | |  |
| Entradas | | | Saídas esperadas | | | | | *Tabela 2: combinações de interesse das entradas da ULA para o teste da funcionalidade ADD (entrada ).* |
|  |  |  |  |  |  |  |  |
| 0100 | 0010 | x | 0110 | 0 | 0 | 0 | 0 |
| 0000 | 0000 | x | 0000 | 0 | 0 | 0 | 1 |
| 0011 | 1010 | x | 1101 | 0 | 0 | 1 | 0 |
| 0110 | 0010 | x | 1000 | 0 | 1 | 1 | 0 |
| 0011 | 1110 | x | 0001 | 1 | 0 | 0 | 0 |
| 0100 | 1011 | x | 1111 | 0 | 0 | 1 | 0 |
| 1100 | 1100 | x | 1000 | 1 | 0 | 1 | 0 |
| 1110 | 0000 | x | 1110 | 0 | 0 | 1 | 0 |
| 0001 | 0001 | x | 0010 | 0 | 0 | 0 | 0 |
| 0101 | 1000 | x | 1101 | 0 | 0 | 1 | 0 |
| 1001 | 0101 | x | 1110 | 0 | 0 | 1 | 0 |
| 0100 | 0001 | x | 0101 | 0 | 0 | 0 | 0 |
| 1001 | 1010 | x | 0011 | 1 | 1 | 0 | 0 |
| 1111 | 1111 | x | 1110 | 1 | 0 | 1 | 0 |
| 1000 | 0110 | x | 1110 | 0 | 0 | 1 | 0 |
| 0110 | 1010 | x | 0000 | 1 | 0 | 0 | 1 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teste da função ADDC | | | | | | | |  |
| Entradas | | | Saídas esperadas | | | | | *Tabela 3: combinações de interesse das entradas da ULA para o teste da funcionalidade ADDC (entrada ).* |
|  |  |  |  |  |  |  |  |
| 0100 | 0010 | 1 | 0111 | 0 | 0 | 0 | 0 |
| 0000 | 0000 | 0 | 0000 | 0 | 0 | 0 | 1 |
| 0011 | 1010 | 0 | 1101 | 0 | 0 | 1 | 0 |
| 0110 | 0010 | 1 | 1001 | 0 | 1 | 1 | 0 |
| 0011 | 1110 | 0 | 0001 | 1 | 0 | 0 | 0 |
| 0100 | 1011 | 1 | 0000 | 1 | 0 | 0 | 1 |
| 1100 | 1100 | 0 | 1000 | 1 | 0 | 1 | 0 |
| 1110 | 0000 | 0 | 1110 | 0 | 0 | 1 | 0 |
| 0001 | 0001 | 1 | 0011 | 0 | 0 | 0 | 0 |
| 0101 | 1000 | 0 | 1101 | 0 | 0 | 1 | 0 |
| 1001 | 0101 | 0 | 1110 | 0 | 0 | 1 | 0 |
| 0100 | 0001 | 1 | 0110 | 0 | 0 | 0 | 0 |
| 1001 | 1010 | 0 | 0011 | 1 | 1 | 0 | 0 |
| 1111 | 1111 | 1 | 1111 | 1 | 0 | 1 | 0 |
| 1000 | 0110 | 1 | 1111 | 0 | 0 | 1 | 0 |
| 0110 | 1010 | 1 | 0001 | 1 | 0 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teste da função SUB | | | | | | | |  |
| Entradas | | | Saídas esperadas | | | | | *Tabela 4: combinações de interesse das entradas da ULA para o teste da funcionalidade SUB (entrada ).* |
|  |  |  |  |  |  |  |  |
| 0100 | 0010 | x | 0010 | 1 | 0 | 0 | 0 |
| 0000 | 0000 | x | 0000 | 1 | 0 | 0 | 1 |
| 0011 | 1010 | x | 1001 | 0 | 1 | 1 | 0 |
| 0110 | 0010 | x | 0100 | 1 | 0 | 0 | 0 |
| 0011 | 1110 | x | 0101 | 0 | 0 | 0 | 0 |
| 0100 | 1011 | x | 1001 | 0 | 1 | 1 | 0 |
| 1100 | 1100 | x | 0000 | 1 | 0 | 0 | 1 |
| 1110 | 0000 | x | 1110 | 1 | 0 | 1 | 0 |
| 0001 | 0001 | x | 0000 | 1 | 0 | 0 | 1 |
| 0101 | 1000 | x | 1101 | 0 | 1 | 1 | 0 |
| 1001 | 0101 | x | 0100 | 1 | 1 | 0 | 0 |
| 0100 | 0001 | x | 0011 | 1 | 0 | 0 | 0 |
| 1001 | 1010 | x | 1111 | 0 | 0 | 1 | 0 |
| 1111 | 1111 | x | 0000 | 1 | 0 | 0 | 1 |
| 1000 | 0110 | x | 0010 | 1 | 1 | 0 | 0 |
| 0110 | 1010 | x | 1100 | 0 | 1 | 1 | 0 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teste da função INC | | | | | | | |  |
| Entradas | | | Saídas esperadas | | | | | *Tabela 5: combinações de interesse das entradas da ULA para o teste da funcionalidade SUB (entrada ).* |
|  |  |  |  |  |  |  |  |
| 0000 | x | x | 0001 | 0 | 0 | 0 | 0 |
| 0001 | x | x | 0010 | 0 | 0 | 0 | 0 |
| 0010 | x | x | 0011 | 0 | 0 | 0 | 0 |
| 0011 | x | x | 0100 | 0 | 0 | 0 | 0 |
| 0100 | x | x | 0101 | 0 | 0 | 0 | 0 |
| 0101 | x | x | 0110 | 0 | 0 | 0 | 0 |
| 0110 | x | x | 0111 | 0 | 0 | 0 | 0 |
| 0111 | x | x | 1000 | 0 | 1 | 0 | 0 |
| 1000 | x | x | 1001 | 0 | 0 | 1 | 0 |
| 1001 | x | x | 1010 | 0 | 0 | 1 | 0 |
| 1010 | x | x | 1011 | 0 | 0 | 1 | 0 |
| 1011 | x | x | 1100 | 0 | 0 | 1 | 0 |
| 1100 | x | x | 1101 | 0 | 0 | 1 | 0 |
| 1101 | x | x | 1110 | 0 | 0 | 1 | 0 |
| 1110 | x | x | 1111 | 0 | 0 | 1 | 0 |
| 1111 | x | x | 0000 | 1 | 0 | 0 | 1 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teste da função DEC | | | | | | | |  |
| Entradas | | | Saídas esperadas | | | | | *Tabela 6: combinações de interesse das entradas da ULA para o teste da funcionalidade SUB (entrada ).* |
|  |  |  |  |  |  |  |  |
| 0000 | x | x | 1111 | 0 | 0 | 1 | 0 |
| 0001 | x | x | 0000 | 1 | 0 | 0 | 1 |
| 0010 | x | x | 0001 | 1 | 0 | 0 | 0 |
| 0011 | x | x | 0010 | 1 | 0 | 0 | 0 |
| 0100 | x | x | 0011 | 1 | 0 | 0 | 0 |
| 0101 | x | x | 0100 | 1 | 0 | 0 | 0 |
| 0110 | x | x | 0101 | 1 | 0 | 0 | 0 |
| 0111 | x | x | 0110 | 1 | 0 | 0 | 0 |
| 1000 | x | x | 0111 | 1 | 1 | 0 | 0 |
| 1001 | x | x | 1000 | 1 | 0 | 1 | 0 |
| 1010 | x | x | 1001 | 1 | 0 | 1 | 0 |
| 1011 | x | x | 1010 | 1 | 0 | 1 | 0 |
| 1100 | x | x | 1011 | 1 | 0 | 1 | 0 |
| 1101 | x | x | 1100 | 1 | 0 | 1 | 0 |
| 1110 | x | x | 1101 | 1 | 0 | 1 | 0 |
| 1111 | x | x | 1110 | 1 | 0 | 1 | 0 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teste da função NEG | | | | | | | |  |
| Entradas | | | Saídas esperadas | | | | | *Tabela 7: combinações de interesse das entradas da ULA para o teste da funcionalidade SUB (entrada ).* |
|  |  |  |  |  |  |  |  |
| 0000 | x | x | 0000 | 1 | 0 | 0 | 1 |
| 0001 | x | x | 1111 | 0 | 0 | 1 | 0 |
| 0010 | x | x | 1110 | 0 | 0 | 1 | 0 |
| 0011 | x | x | 1101 | 0 | 0 | 1 | 0 |
| 0100 | x | x | 1100 | 0 | 0 | 1 | 0 |
| 0101 | x | x | 1011 | 0 | 0 | 1 | 0 |
| 0110 | x | x | 1010 | 0 | 0 | 1 | 0 |
| 0111 | x | x | 1001 | 0 | 0 | 1 | 0 |
| 1000 | x | x | 1000 | 0 | 1 | 1 | 0 |
| 1001 | x | x | 0111 | 0 | 0 | 0 | 0 |
| 1010 | x | x | 0110 | 0 | 0 | 0 | 0 |
| 1011 | x | x | 0101 | 0 | 0 | 0 | 0 |
| 1100 | x | x | 0100 | 0 | 0 | 0 | 0 |
| 1101 | x | x | 0011 | 0 | 0 | 0 | 0 |
| 1110 | x | x | 0010 | 0 | 0 | 0 | 0 |
| 1111 | x | x | 0001 | 0 | 0 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teste da função CMPL | | | | | | | |  |
| Entradas | | | Saídas esperadas | | | | | *Tabela 8: combinações de interesse das entradas da ULA para o teste da funcionalidade CMPL (entrada ).* |
|  |  |  |  |  |  |  |  |
| 0000 | x | x | 1111 | 0 | 0 | 1 | 0 |
| 0001 | x | x | 1110 | 0 | 0 | 1 | 0 |
| 0010 | x | x | 1101 | 0 | 0 | 1 | 0 |
| 0011 | x | x | 1100 | 0 | 0 | 1 | 0 |
| 0100 | x | x | 1011 | 0 | 0 | 1 | 0 |
| 0101 | x | x | 1010 | 0 | 0 | 1 | 0 |
| 0110 | x | x | 1001 | 0 | 0 | 1 | 0 |
| 0111 | x | x | 1000 | 0 | 0 | 1 | 0 |
| 1000 | x | x | 0111 | 0 | 0 | 0 | 0 |
| 1001 | x | x | 0110 | 0 | 0 | 0 | 0 |
| 1010 | x | x | 0101 | 0 | 0 | 0 | 0 |
| 1011 | x | x | 0100 | 0 | 0 | 0 | 0 |
| 1100 | x | x | 0011 | 0 | 0 | 0 | 0 |
| 1101 | x | x | 0010 | 0 | 0 | 0 | 0 |
| 1110 | x | x | 0001 | 0 | 0 | 0 | 0 |
| 1111 | x | x | 0000 | 0 | 0 | 0 | 1 |

**Interface gráfica do usuário, Aplicativo, Tabela, Word

Descrição gerada automaticamenteInterface gráfica do usuário, Aplicativo, Tabela, Excel

Descrição gerada automaticamente**Interface gráfica do usuário, Aplicativo, Tabela, Word, Excel

Descrição gerada automaticamente

Figura 23: Simulação temporal da função ADD (com a entrada seletora sendo 0001)

Figura 22: Simulação funcional da função ADD (com a entrada seletora sendo 0001)

Figura 21: Simulação funcional da função ADD (com a entrada seletora sendo 0000)

**Tabela

Descrição gerada automaticamente com confiança médiaTela de computador com texto preto sobre fundo branco

Descrição gerada automaticamenteTela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente**

Figura 25: Simulação funcional da função ADDC (com a entrada seletora sendo 0011)

Figura 26: Simulação temporal da função ADDC (com a entrada seletora sendo 0011)

Figura 24: Simulação funcional da função ADDC (com a entrada seletora sendo 0010)

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente com confiança médiaInterface gráfica do usuário

Descrição gerada automaticamente com confiança médiaInterface gráfica do usuário, Aplicativo, Tabela, Excel

Descrição gerada automaticamente

Figura 29: Simulação temporal da função SUB (com a entrada seletora sendo 0101)

Figura 28: Simulação funcional da função SUB (com a entrada seletora sendo 0101)

Figura 27: Simulação funcional da função SUB (com a entrada seletora sendo 0100)

**Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamenteTela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente**Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamenteTela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

Figura 31: Simulação temporal da função INC

Figura 33: Simulação temporal da função DEC

Figura 32: Simulação funcional da função DEC

Figura 30: Simulação funcional da função INC

Tela de computador com fundo branco

Descrição gerada automaticamente com confiança médiaTela de computador com texto preto sobre fundo branco

Descrição gerada automaticamenteTela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente com confiança média

Figura 36: Simulação funcional da função CMPL

Figura 35: Simulação temporal da função NEG

Figura 34: Simulação funcional da função NEG

* 1. **Interface gráfica do usuário, Aplicativo, Tabela, Excel

     Descrição gerada automaticamenteEncapsulamento do módulo ULA e testes finais:**

Figura 37: Simulação temporal da função CMPL

Texto

Descrição gerada automaticamente com confiança baixaAssim como pedido no roteiro desta atividade, o circuito resultante do projeto ULA foi encapsulado em um módulo e teve suas entradas e saídas associadas aos pinos respectivos. O circuito resultante está disposto no diagrama da figura 38.

Figura 38: Esquemático do circuito de teste da ULA.

Todos os testes executados anteriormente para o circuito ULA foram refeitos para o circuito de teste da ULA, seus resultados são mostrados nas figuras 39 a 55.

|  |
| --- |
| *Figura 39: Simulação funcional da função ADD (com entrada seletora 0000)* |

|  |
| --- |
| Figura 40: Simulação funcional da função ADD (com entrada seletora 0001) |
| *Figura 41: Simulação temporal da função ADD (com entrada seletora 0001)* |
| *Figura 42: Simulação funcional da função ADDC (com entrada seletora 0010)* |
| *Figura 43: Simulação funcional da função ADDC (com entrada seletora 0011)* |
| *Figura 43: Simulação temporal da função ADDC (com entrada seletora 0011)* |
| *Figura 44: Simulação funcional da função SUB (com entrada seletora 0100)* |
| *Figura 45: Simulação funcional da função SUB (com entrada seletora 0101)* |
| *Figura 46: Simulação temporal da função SUB (com entrada seletora 0101)* |
| *Figura 47: Simulação funcional da função INC (com entrada seletora 0110)* |
| *Figura 48: Simulação temporal da função INC (com entrada seletora 0110)* |
| *Figura 49: Simulação funcional da função DEC (com entrada seletora 0111)* |
| *Figura 50: Simulação temporal da função DEC (com entrada seletora 0111)* |
| *Figura 51: Simulação funcional da função NEG (com entrada seletora 1000)* |
| *Figura 52: Simulação temporal da função NEG (com entrada seletora 1000)* |
| *Figura 53: Simulação funcional da função CMPL (com entrada seletora 1001)* |
| *Figura 54: Simulação temporal da função CMPL (com entrada seletora 1001)* |