

**Análise e Síntese de Funções Combinacionais de Múltiplas Saídas**

Laboratório 2 (Circuitos Digitais I)

Março, 2019

Turma 1COMP

# Adriano Soares Rodrigues e Matheus Vidal de Menezes

Prof.º Osamu Saotome

Instituto Tecnológico de Aeronáutica (ITA)

São José dos Campos, São Paulo, Brasil.

{sadrianorod, [matheusvidaldemenezes}@gmail.com](mailto:matheusvidaldemenezes%7d@gmail.com)

1. **Introdução**

No início do século XX, John Ambrose Fleming criou a primeira válvula termiônica, dispositivo que alavancou a eletrônica e o rumo do desenvolvimento tecnológico: possibilitando a criação do rádio e também de televisores. As válvulas termiônicas consistem basicamente por um invólucro de vidro, metal ou cerâmica a vácuo ou com algum tipo de gás. No interior das válvulas tem-se vários elementos metálicos internos chamados de anodo, catodo, grade e filamento conforme retratado na Figura 1.



Figura 1. Exemplo de válvula termiônica com filamento de 12V e base de 7 pinos.

Após o advento das válvulas termiónicas, foram desenvolvidos os transístores, representados na Figura 2, que são basicamente semicondutores capazes de desempenhar a mesma função de uma válvula, *i.e.*, controlar o fluxo de corrente, entretanto sem dissipar tanto calor, sendo menor e mais econômico. Por conta do sucesso da substituição das válvulas, estas tiveram sua produção em larga escala interrompidas.



Figura 2. Réplica do primeiro transístor, inventado no Bell Labs, 23 de dezembro, 1947 (Foto: Reprodução/Wikipédia).

É importante salientar também que, com o progresso tecnológico dos transístores, criou-se a famosa família de circuitos integrados TTL, Transistor-Transistor Logic, responsáveis pelo desenvolvimento de portas lógicas, que, na verdade, são circuitos transistorizados, tecnologia essencial para o surgimento dos computadores pessoais (Personal Computer - PC) de hoje.

1. **Objetivo**

Diante desse breve contexto histórico, o objetivo da segunda prática laboratorial de EEA-21 Circuitos Digitais mostra-se de grande importância. Isso, porque introduz novos circuitos e, como na primeira prática, trata da familiarização das portas lógicas estudadas teoricamente e do aprendizado quanto a utilização destas em simulações, via o *software* *Quartus*® *13.01*, de circuitos de múltiplas saídas, tais como: demultiplexadores, codificadores e decodificadores.

1. **Tarefas**

**Problema 4.1)**

A tabela verdade para um DEMUX 1x4 é feita, conforme a Tabela 1.

4 saídas 🡪 2 chaves seletoras

Tabela 1. Tabela lógica, ou tabela verdade, referente ao problema 4.1 que possui E como entrada e , , e como saídas. Além de e como chaves seletoras.

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

Assim, temos para cada saída:

Saídas:

Note que as expressões já estão minimizadas, visto que se fizéssemos os devidos Mapas de Karnaugh, veríamos que cada uma possuirá somente 1 elemento para ser envolvido. Note ainda que se , todos os valores de saída serão zeros, funcionando como uma entrada de habilitação.

Para cada saída, podemos “negar” duas vezes a expressão, de modo a obter resultados, conforme a Eq.1.

(1)

Aplicando Lei de De Morgan de produto para soma, temos uma porta NOR de 3 entradas:

(2)

Repetindo o processo para as outras saídas, obtemos:

(3)

(4)

(5)

Logo, o diagrama esquemático do circuito pode ser feito apenas com apenas 1 tipo de porta lógica: NOR, conforme a Figura 1. Note que podemos fazer uma inversora com uma porta NOR de duas entradas iguais, *i.e.*, .

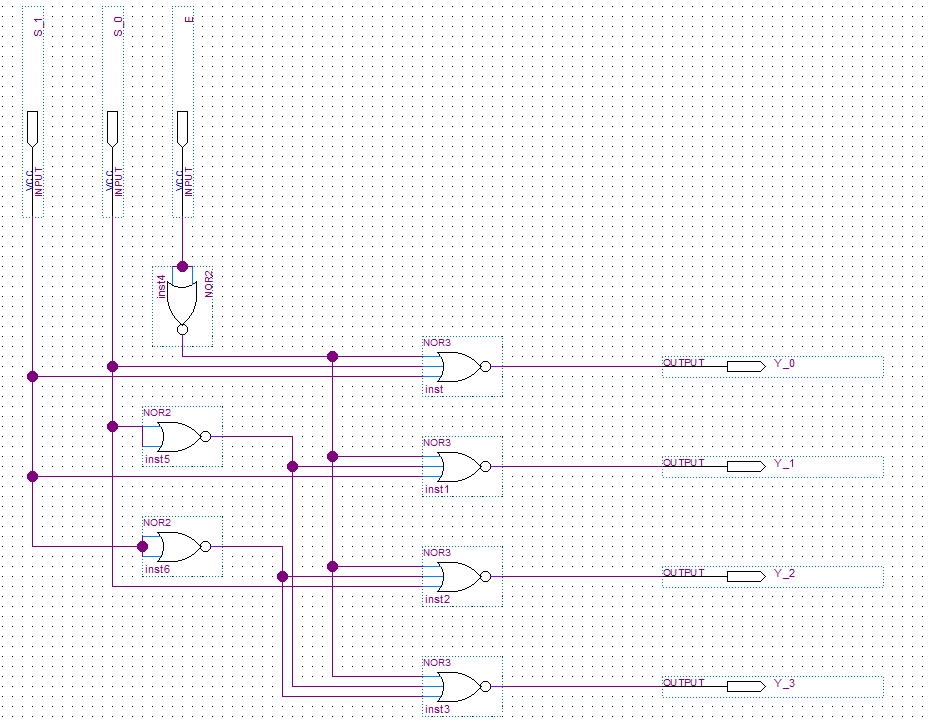


Figure 1. Circuito montado para o DEMUX 1x4 com apenas portas NOR.

Após a compilação e da simulação do sistema, obtemos o diagrama de temporização, conforme a Figura 2.

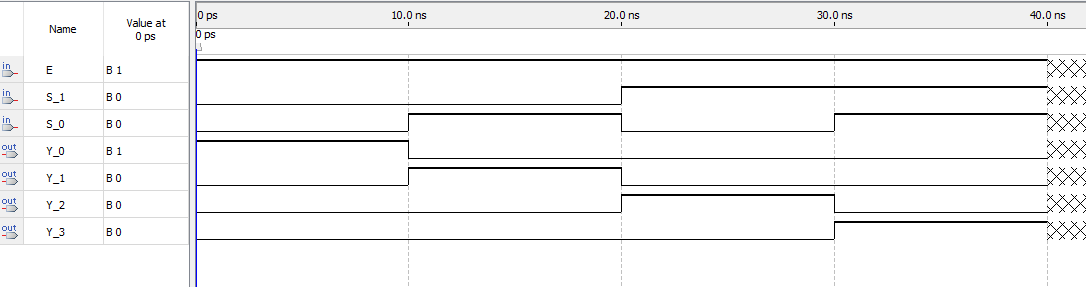


Figure 2. Diagrama de Temporização para o circuito da Figura 1 do problema 4.1.

Note que, de fato, a simulação corrobora para o resultado encontrado, as saídas Y\_0, Y\_1, Y\_2 e Y\_3 possuem os valores de acordo com a tabela verdade, vide Tabela 1.

**Problema 4.2)**

Seja e , os bits dos números em binário e Gray, respectivamente, , com a quantidade de bits. Podemos construir a tabela verdade para números de bits, conforme a Tabela 2.

Tabela 2. Tabela verdade de um número binário e de Gray, para 3 bits.

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

Pelo método de construção, , colunas em amarelo na Tabela 2. Para os outros bits, escrevendo os bits em função dos , temos a Eq.6.

Para ,

(6)

Note que, com em evidência, a soma das duas primeiras parcelas pode ser simplificada por uma porta XOR e a soma das duas seguintes, com em evidência, por uma porta XNOR, conforme a Eq. 7.

(7)

Analogamente, para ,

(8)

Note que a soma novamente pode ser simplificada por uma porta XOR, conforme a Eq. 9.

(9)

Note que a simplificação para tanto , quanto para , já são mínimas, dado os Mapas de Karnaugh mostrados respectivamente nas Tabelas 3 e 4.

Tabela 3. Mapa de Karnaugh para a simplificação da variável no problema 4.2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **00** | **01** | **11** | **10** |
| **0** | 0 | **1** | 0 | **1** |
| **1** | **1** | 0 | **1** | 0 |

Que é o resultado previamente obtido, vide Eq. 6.

Tabela 4. Mapa de Karnaugh para a simplificação da variável no problema 4.2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **00** | **01** | **11** | **10** |
| **0** | 0 | **1** | 0 | **1** |
| **1** | 0 | **1** | 0 | **1** |

Que é o resultado previamente obtido, vide Eq. 8.

Logo, o diagrama esquemático do circuito pode ser feito apenas com 4 tipos de portas lógicas: XOR, XNOR, OR e AND, conforme a Figura 3**.**

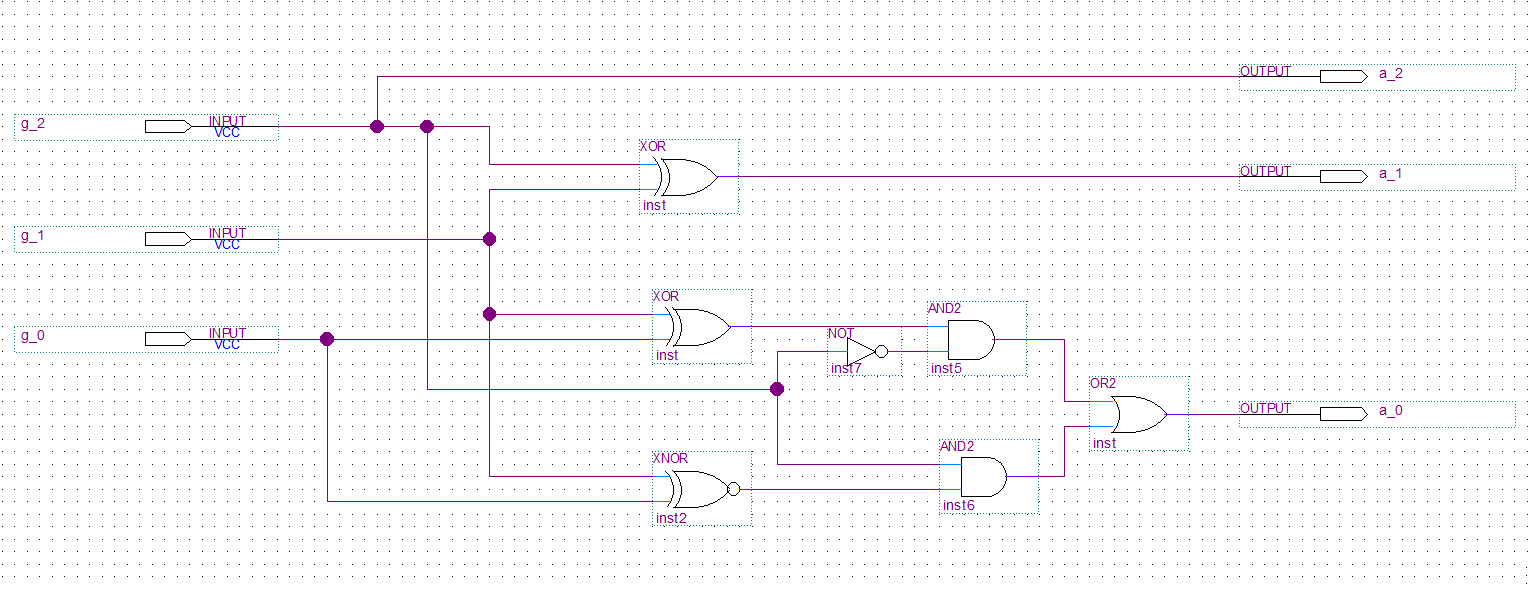


Figura 3. Circuito do problema 4.2 a ser simulado e simplificado.

Após a compilação e da simulação do sistema, obtemos o diagrama de temporização, conforme a Figura 4.

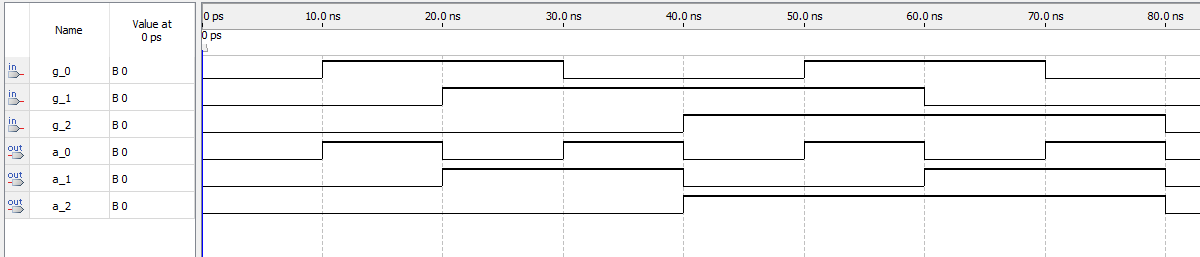


Figura 4. Diagrama de Temporização para o circuito da Figura 3 do problema 4.2.

Note que, de fato, a simulação corrobora para o resultado encontrado, as saídas a\_0, a\_1 e a\_2 possuem os valores de acordo com a tabela verdade, vide Tabela 2.

**Problema 4.3)**

A tabela verdade para um DECOD 3x8 é feita, conforme a Tabela 5.

8 saídas 🡪 3 sinais de entrada

Tabela 5. Tabela lógica, ou tabela verdade, referente ao problema 4.3 que possui , , como entradas binárias. Além de , , , , , , , onde apenas uma das saídas recebe o valor 1 para marcar o valor da entrada em binário através do índice decimal de cada uma das chaves .

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

Assim, temos para cada saída:

Saídas: e

Note que as expressões já estão minimizadas, visto que se fizéssemos os devidos Mapas de Karnaugh, veríamos que cada uma possuirá somente 1 elemento para ser envolvido.

Para incluir o caso de , temos:

Saídas: e

O diagrama esquemático do circuito pode ser feito, conforme a Figura 5.

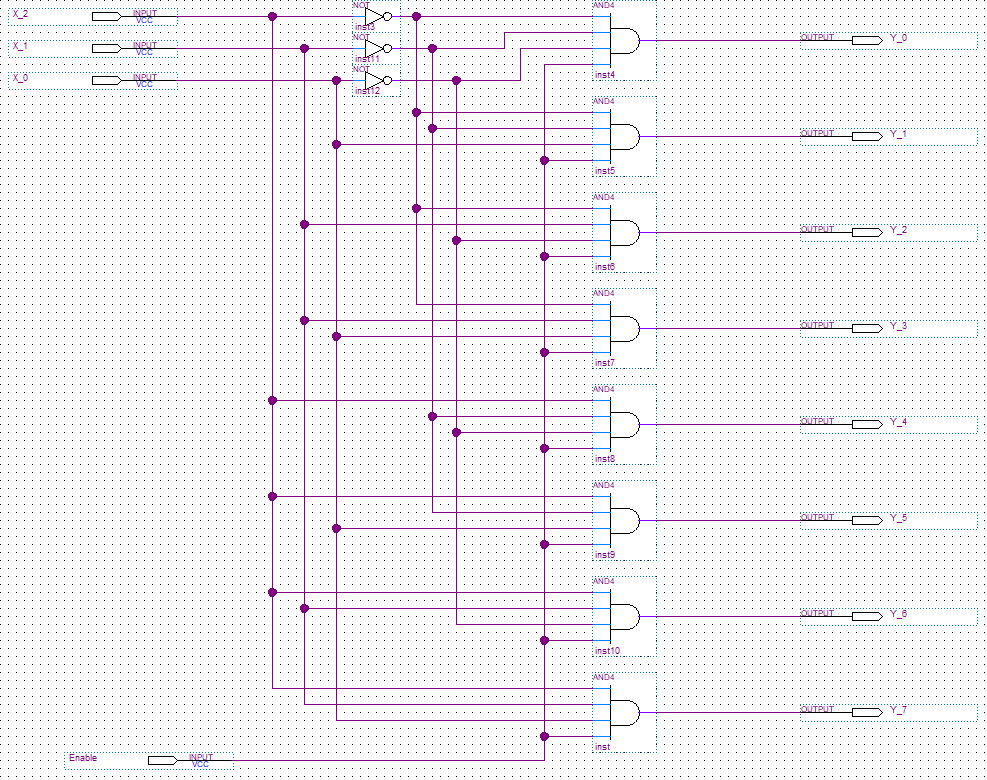


Figure 5. Circuito montado para o DECOD 3x8 (entrada binário puro).

Após a compilação e da simulação do sistema, obtemos o diagrama de temporização, conforme a Figura 6.

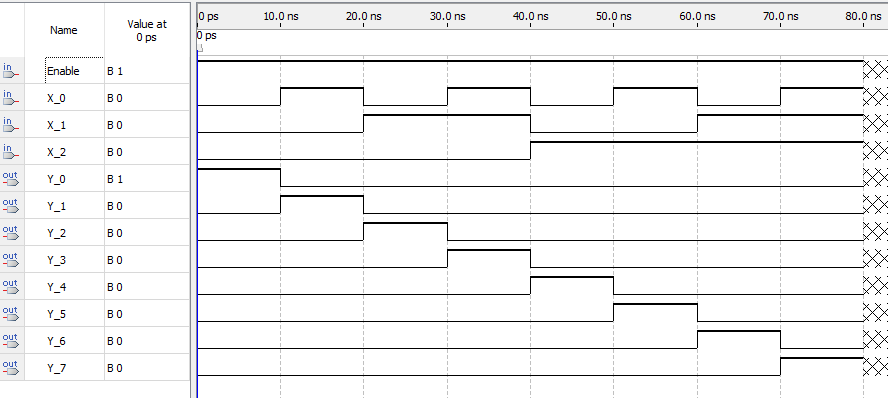


Figure 6. Diagrama de Temporização para o circuito da Figura 5 do problema 4.3. Note .

Note que, de fato, a simulação corrobora para o resultado encontrado, as saídas Y\_0, Y\_1, Y\_2, Y\_3, Y\_4, Y\_5, Y\_6 e Y\_7 possuem os valores de acordo com a tabela verdade, vide Tabela 5.

**Problema 4.4)**

1. A tabela verdade pedida para o codificador de prioridade 4x2 pedido na questão é representada na Tabela 6.

Tabela 6. Tabela lógica, ou tabela verdade, referente ao problema 4.4 que possui , , e como entradas, considerando seus índices decimais sempre com prioridade para o maior valor, neste caso seria a entrada de íncide 3. Além das saídas , (com valores em binários) e A como indicadora se todas as entradas foram nulas ou não.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | ***A*** |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | x | 0 | 1 | 1 |
| 0 | 1 | x | x | 1 | 0 | 1 |
| 1 | X | x | x | 1 | 1 | 1 |

Para a saída **,** temos a soma dos produtos considerando os maxtermos das entradas na Tabela 6.

(10)

Utilizando a propriedade que , teremos então que:

(11)

Por DeMorgan, temos:

(12)

Como **=** 1, podemos simplificar mais ainda a expressão:

(13)

Utilizando a propriedade

(14)

Finalmente, por DeMorgan, novamente, teremos que:

(15)

Analogamente, para a saída **,** temos o produto da soma (POS) abaixo, considerando os maxtermos das entradas estudadas.

(16)

Utilizando a propriedade que , teremos então que :

(17)

Por DeMorgan teremos a expressão abaixo:

(18)

Pela propriedade **,** teremos então

(19)

Finalmente, para a saída A, temos o produto da soma (POS) abaixo, considerando os maxtermos das entradas estudadas.

(20)

Assim, temos as saídas:

**b)**

O diagrama esquemático do circuito pode ser feito, conforme a Figura 7.

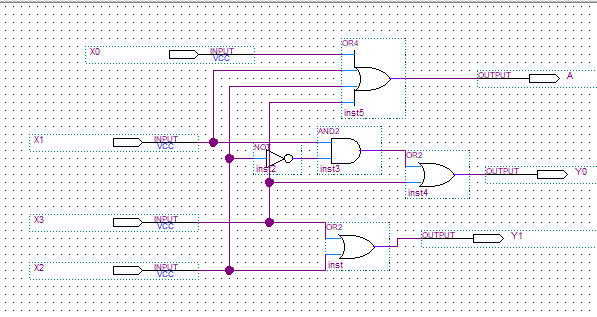


Figure 7. Circuito montado para o COD 4x2 (saída em binário puro).

**c)**

Após a compilação e da simulação do sistema, obtemos o diagrama de temporização, conforme a Figura 8.

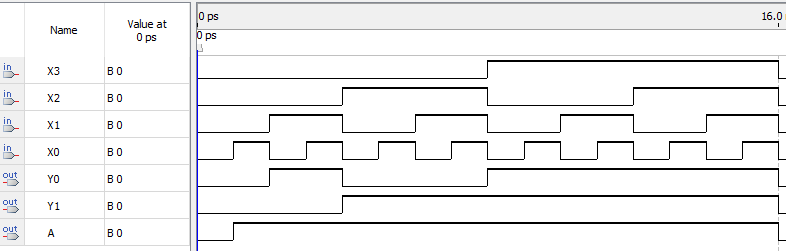


Figure 8. Diagrama de Temporização para o circuito da Figura 7 do problema 4.4.

É importante salientar que as repostas dadas no diagrama de temporização estão de acordo com as respostas esperadas no estudo de um codificador 4x2 de prioridade.

**Problema 4.5)**

**a)**

A tabela verdade pedida para o Bloco H pedido na questão é representada logo abaixo.

Tabela 6. Tabela lógica, ou tabela verdade, referente ao problema 4.5 que possui como entradas, sendo utilizadas como resultante das entradas obtidas apenas para fins de análise do diagrama temporal para verificar a validade dos valores lógicos das saídas, entretanto para a black box Bloco H . Além das saídas , e e A como indicadora se todas as entradas foram nulas ou não.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  | **A** |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |
| 0 | 0 | x | x | x | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | x | x | x | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | x | x | x | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | x | x | x | 1 | 1 | 1 | 1 | 1 |

Assim, simplificando as expressões de saída utilizando os mintermos na tabela verdade teremos que:

(21)

(22)

(23)

(24)

**b)**

O diagrama esquemático do circuito bem como o diagrama temporal do Bloco H são inseridos abaixo, conforme a Figura 9 e 10, respectivamente.

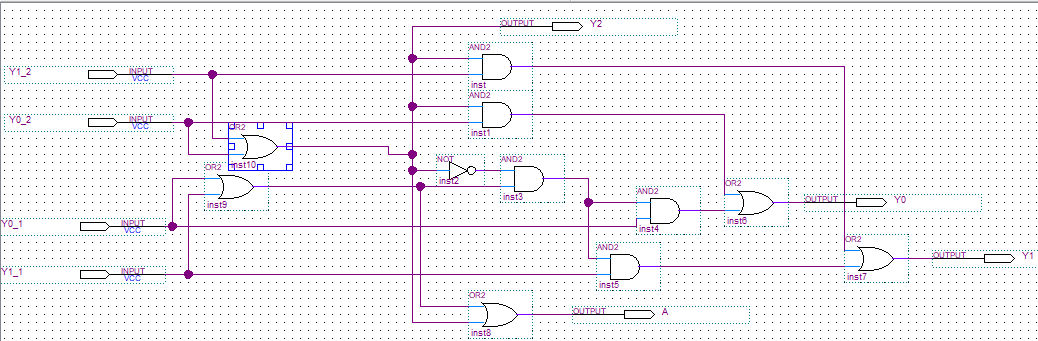


Figure 9. Circuito montado para a simulação do diagrama temporal do Bloco H.

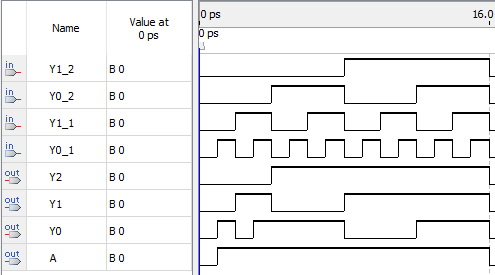


Figure 10. Diagrama de Temporização para o circuito da Figura 9 do problema 4.5.

**c)**

O diagrama esquemático do circuito bem como o diagrama temporal do dispositivo completo, utilizando os codificadores da questão 4.4 e o bloco H criado nessa questão são inseridos abaixo, conforme a Figura 11 e 12, respectivamente.

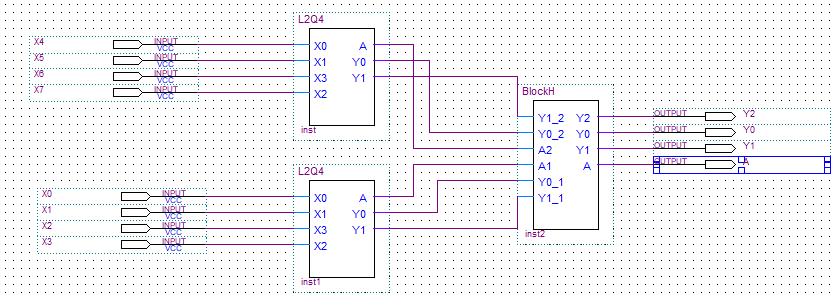


Figure 11. Circuito montado para a simulação do diagrama temporal do Codificador de Prioridade 8x3, onde L2Q4 são os codificadores de prioridade 4x2 criados na questõ 4.4.

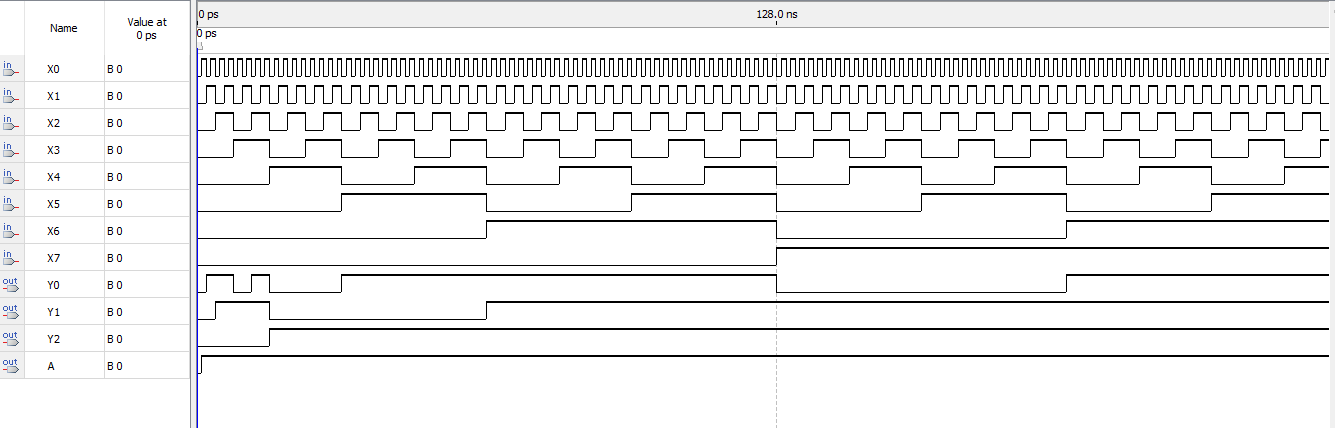


Figure 12. Diagrama de Temporização para o circuito final do Codificador de Prioridade 8x3 da Figura 11 do problema 4.5.

É importante salientar que as repostas dadas no diagrama de temporização estão de acordo com as respostas esperadas no estudo de um codificador 8x3 de prioridade.

**IV. Conclusão**

As atividades laboratoriais desenvolvidas, bem como o aprendizado na utilização do software *Quartus*® *13.01* foram de grande importância para desenvolver os conhecimentos teóricos e práticos a respeito dos circuitos digitais, principalmente sobre o funcionamento de demultiplexadores, codificadores e decodificadores, tanto os convencionais quanto os de prioridade (quando são inseridas mais de uma tecla no input).

Do experimento realizado, foi possível, embora toda simplicidade em sua metodologia, conseguir obter resultados coerentes com a teoria de circuitos digitais. Além disso, tanto o software, quanto as técnicas aplicadas são de grande importância para o aprendizado da frente de EEA-21.