

## Roteiro do Projeto 1: Apresentação do Número de Matrícula

### 1. Introdução

O objetivo do projeto 1 do Laboratório de Sistemas Digitais é desenvolver um sistema digital combinacional que a partir do acionamento de três chaves apresenta o código da disciplina e o número da matrícula do aluno em um par de displays de sete segmentos.

Esse laboratório tem como objetivo exercitar a metodologia de desenvolvimento de projetos de engenharia apoiada em computador (CAE). Esse processo envolve a compreensão do problema, seu planejamento, desenvolvimento da solução lógica, integração dos subsistemas, implementação no ambiente computacional, simulação, testes, depuração do projeto, implementação física e registro dos resultados.

### 2. Planejamento do Projeto

O processo de elaboração do projeto consiste nas seguintes etapas:

#### **Projeto Lógico (preparatório ao laboratório – antes da aula):**

➤ Análise do projeto lógico do sistema digital para a elaboração das Tabelas Verdade, considerando as especificações do problema.

#### **Aula 5 (AVA):**

➤ Desenvolvimento do projeto lógico do sistema digital (tabelas verdade, mapas de Karnaugh e expressões lógicas simplificadas), considerando **duas formas de implementação**: com portas lógicas padrão (AND, OR, NOT) e com multiplexadores comercialmente disponíveis (**Avaliação Formativa**);

#### **Aula 6 (AVA):**

➤ Elaboração do diagrama esquemático da lógica principal do projeto (decodificador do aluno) utilizando os recursos de edição gráfica (*Block Diagram*) na ferramenta Quartus Prime Lite;

➤ O projeto lógico deve ser elaborado utilizando duas formas de implementação: com portas lógicas e com multiplexadores;

➤ Apresentação individual do diagrama esquemático do projeto do decodificador ao professor (**1 ponto**);

#### **Aula 7 (AVA):**

➤ Simulação funcional do decodificador do aluno na ferramenta Quartus Prime Lite;

➤ Apresentação individual da simulação funcional ao professor para comprovar o correto funcionamento do decodificador, utilizando as duas formas de implementação (**2 pontos**);

➤ Incorporação dos outros módulos que complementam o projeto lógico.

#### **Aula 8 (CGI):**

➤ Atribuição de pinos e preparação para configuração do dispositivo lógico programável (**FPGA**);

➤ Apresentação individual do projeto completo e teste projeto na placa **DE10-Lite** para avaliação da implementação pelo professor (**1 ponto\***);

➤ Entrega do relatório do projeto (conforme instruções no final do roteiro) (**1 ponto**).

**\*Observação:** Na impossibilidade da realização da configuração do FPGA (por indisponibilidade do acesso ao **CGI**) o relatório passará a valer **2 pontos**.

Nas aulas de apresentação o aluno deverá estar preparado para responder às seguintes questões:

- ✓ Demonstrar que a lógica de controle desenvolvida opera de acordo com a especificação;
- ✓ Descrever e exemplificar o funcionamento da lógica de controle (casos de teste considerados);
- ✓ Descrever a função dos componentes utilizados.

### 3. Descrição Funcional do Projeto

Deseja-se implementar um sistema digital que, a partir do acionamento de três chaves (**S2**, **S1**, **S0**), apresente o código da disciplina e o número da matrícula do aluno em dois displays de sete segmentos conforme a **Tabela 1**.

Os displays são do tipo LED (*Light Emitting Diode*) de sete segmentos com ligação em ânodo comum (todos conectados em Vcc). Esse tipo de display pode ser acionado por um decodificador comercialmente disponível (**74247**). A tabela funcional do decodificador **74247** é fornecida no final deste roteiro.

Devem ser utilizadas **duas formas de implementação** da lógica do **Decodificador\_do\_Aluno**: para o display **HEX0** devem ser utilizadas portas lógicas padrão (AND, OR, NOT); para o display **HEX1** devem ser utilizados multiplexadores (**74151**). A tabela funcional de um multiplexador **7151** é fornecida no final deste roteiro.

O número de matrícula do aluno possui nove dígitos, no formato: **D8 D7 D6 D5 D4 D3 D2 D1 – D0**, onde **D0** é o dígito de controle. Os dígitos devem ser apresentados sequencialmente nos displays conforme comutam-se as chaves de seleção (**S2 S1 S0**), conforme representado na **Tabela 1**. Observe que em algumas combinações das chaves os displays devem ficar apagados, essa condição é obtida com o código “1111” nas entradas DCBA do decodificador **74247**.

**Tabela 1:** Valor numérico que cada display deve apresentar conforme o número de matrícula do aluno

S2	S1	S0	DISPLAY HEX1	DISPLAY HEX0
0	0	0	Apagado	Apagado
0	0	1	3	5
0	1	0	1	2
0	1	1	Apagado	D8
1	0	0	D7	D6
1	0	1	D5	D4
1	1	0	D3	D2
1	1	1	D1	D0

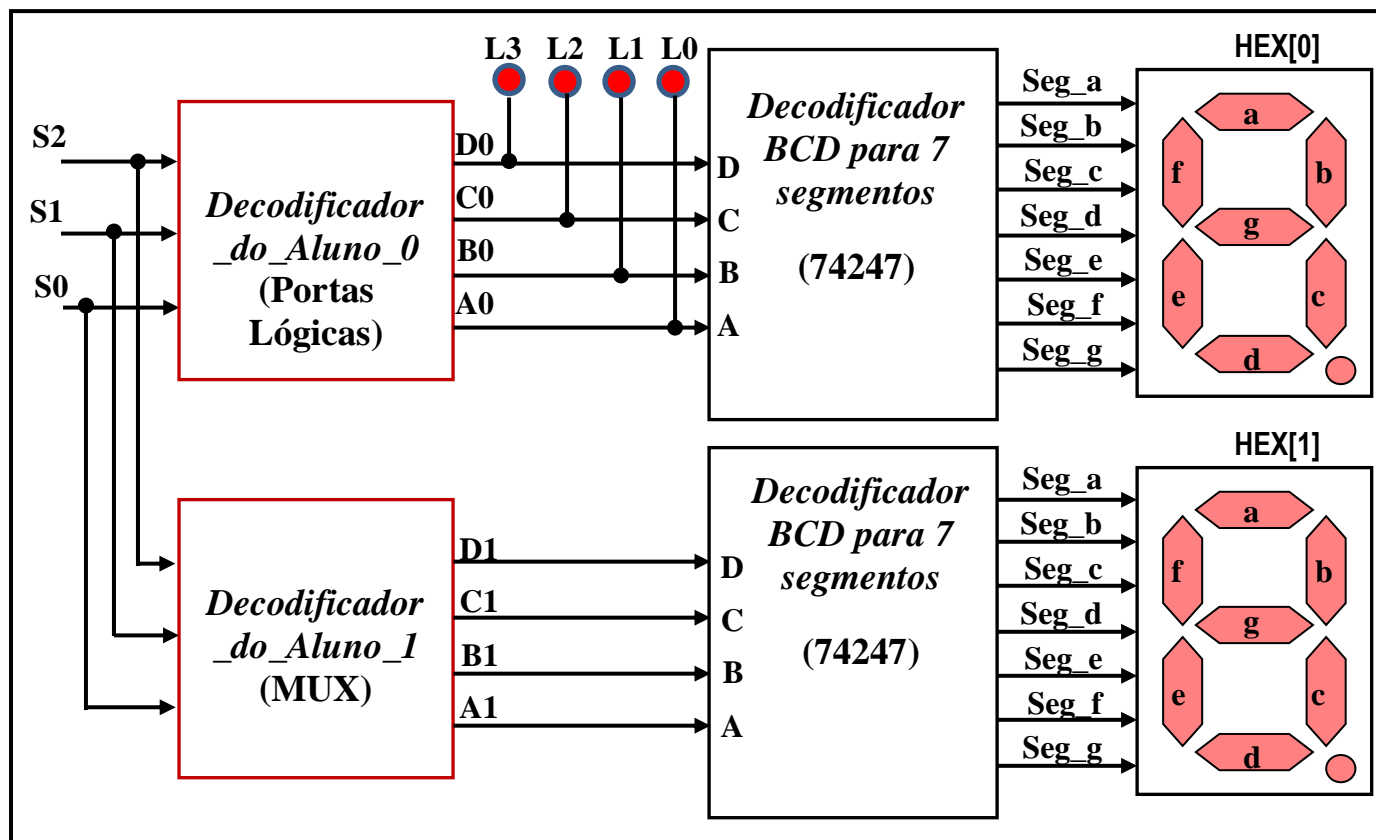
Fonte: Autor.

Por exemplo: caso o aluno tenha número de matrícula: **22118059-4**, seu projeto deve ter a seguinte tabela verdade:

S2	S1	S0	DISPLAY 1	DISPLAY 0
0	0	0	---	---
0	0	1	3	5
0	1	0	1	2
0	1	1	---	2
1	0	0	2	1
1	0	1	1	8
1	1	0	0	5
1	1	1	9	4

O diagrama de blocos completo do sistema digital para a apresentação do código da disciplina e do número da matrícula do aluno é representado na **Figura 1**.

**Figura 1:** Diagrama de blocos do sistema de apresentação do código da disciplina e número da matrícula



Fonte: Autor.

#### 4. Requisitos de Implementação de Projeto (Aula 5 no AVA)

O sistema de controle digital deve ser totalmente implementado utilizando o FPGA da família **MAX10** – modelo: **10M50DAF484C7G** existente na placa de desenvolvimento **DE10-Lite**.

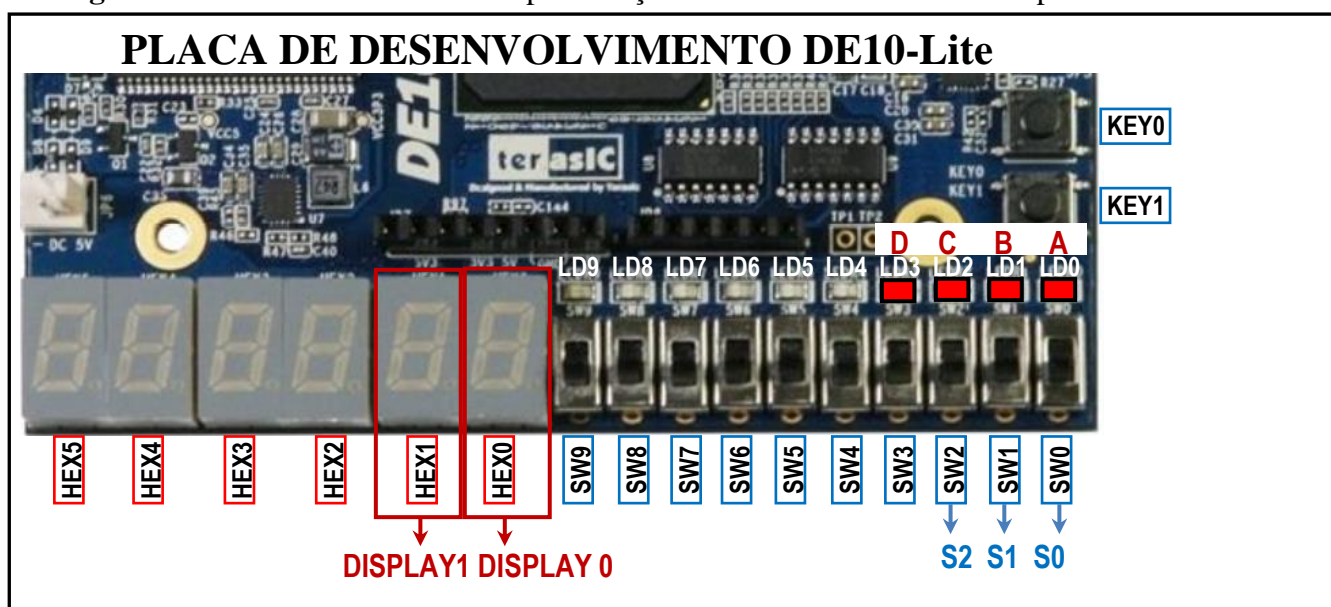
Cada aluno deve desenvolver **individualmente o projeto lógico** (Tabelas Verdade e equações lógicas minimizadas) para representar o sistema de controle de apresentação do seu número de matrícula. Esse projeto deve ser **desenvolvido e discutido com o professor na primeira aula de projeto (Aula 5)**.

Os requisitos específicos desse sistema digital são os seguintes:

- Conforme a **Figura 1** o projeto é constituído por quatro decodificadores: dois que devem ser projetados pelo aluno (**Decodificador do Aluno\_0** e **Decodificador do Aluno\_1**) e outros dois que serão utilizados componentes comerciais para executar a conversão BCD - Sete Segmentos (neste caso o **74247** é apropriado para o display de ânodo comum da placa **DE10-Lite**).
- O **Decodificador do Aluno\_0** deve ser projetado utilizando portas lógicas padrão (AND, OR, NOT);
- O **Decodificador do Aluno\_1** deve ser projetado utilizando multiplexadores 8x1, neste caso o Quartus Prime possui o modelo **74151**. A tabela funcional do multiplexador **74151** é fornecida no final deste roteiro.

- O **Decodificador\_do\_Aluno\_0** deve fornecer como saída um código (**D0 C0 B0 A0**) que deve ser apresentado nos LED's: **L3, L2, L1, L0** da placa **DE10-Lite**. Esse código deve ser tal que o display hexadecimal **HEX0** da placa **DE10-Lite** apresente o número desejado e permanecer apagado para a primeira combinação das chaves, conforme a **Tabela 1**
- O **Decodificador\_do\_Aluno\_1** deve fornecer como saída um código (**D1 C1 B1 A1**) que deve apresentar no display hexadecimal **HEX1** da placa **DE10-Lite** os números desejados e permanecer apagado para a primeira e última combinação das chaves, conforme a **Tabela 1**.
- Os sinais de entrada (**S2, S1 e S0**) devem ser conectados às chaves deslizantes **SW2, SW1 e SW0** da placa **DE10-Lite**, como mostrado na **Figura 2**.

**Figura 2:** Interfaces do sistema de apresentação do número da matrícula na placa **DE10-Lite**.



Fonte: TERASIC, 2020 (adaptado).

## 5. Apresentação da Implementação do Projeto Lógico (**Aula 6 no AVA**)

Na **Aula 6** o aluno deve apresentar a **realização do projeto lógico** do sistema, incluindo os decodificadores **Decodificador\_do\_Aluno\_0** e **Decodificador\_do\_Aluno\_1** implementados técnicas de projeto diferentes (portas lógicas padrão e multiplexadores). O circuito com esses dois componentes deve ser **implementado no ambiente da ferramenta Quartus Prime Lite**. Veja os procedimentos do arquivo: **Tutorial\_Quartus\_Prime\_Captura\_de\_Esquemático** (PRATES, 2019), disponível no Moodle.

## 6. Apresentação da Simulação (**Aula 7 no AVA**)

A apresentação das formas de onda da simulação do Projeto 1 no ambiente do Quartus Prime Lite (**Aula 7**) **pode restringir-se** à simulação funcional dos circuitos: **Decodificador\_do\_Aluno\_0** e **Decodificador\_doAluno\_1** (**circuítos com contorno em vermelho** na **Figura 1**), portanto não precisam ser incluídos os outros elementos do projeto (os decodificadores **74247** e respectivos pinos).

A simulação da lógica do **Decodificador\_do\_Aluno\_0** e **Decodificador\_do\_Aluno\_1** deve ser realizada **no modo funcional**, utilizando o editor de vetor de formas de onda. Nessa simulação devem ser apresentadas **todas as oito possíveis combinações dos sinais de entrada** para demonstrar a correta operação do sistema.

## 7. Conclusão do Projeto (Aula 8 no CGI)

Na **Aula 8** (no CGI o aluno deve incluir todos os componentes (decodificadores 74247 e respectivos pinos), assim como a atribuição da numeração dos pinos do FPGA para cada sinal de entrada e saída do projeto na placa **DE10-Lite**. A associação dos sinais do projeto de apresentação do código da disciplina e do número da matrícula do aluno com os pinos do FPGA da placa DE10-Lite é apresentada na **Tabela 2**.

**Tabela 2:** Numeração dos pinos para os sinais de entrada e saída do FPGA da placa **DE10-Lite**.

SINAL	PINO	DISPLAY HEX0	PINO	DISPLAY HEX1	PINO	LED	PINO
SW[2] – S2	D12	HEX0 – Seg_g	C17	HEX1 – Seg_g	B17	LD3 – D0	B10
SW[1] – S1	C11	HEX0 – Seg_f	D17	HEX1 – Seg_f	A18	LD2 – C0	A10
SW[0] – S0	C10	HEX0 – Seg_e	E16	HEX1 – Seg_e	A17	LD1 – B0	A9
		HEX0 – Seg_d	C16	HEX1 – Seg_d	B16	LD0 – A0	A8
		HEX0 – Seg_c	C15	HEX1 – Seg_c	E18		
		HEX0 – Seg_b	E15	HEX1 – Seg_b	D18		
		HEX0 – Seg_a	C14	HEX1 – Seg_a	C18		

Fonte: TERASIC, 2020 (adaptado).

Os procedimentos para a execução da atribuição de pinos do **FPGA** da família MAX10 - modelo: **10M50DAF484C7G** a um projeto lógico no ambiente do Quartus Prime Lite estão apresentados na **seção 7.3** do arquivo: **Tutorial\_Quartus\_Prime\_Captura\_de\_Esquemático** (PRATES, 2019), disponível no Moodle.

Os demais pinos disponíveis para chaves e LED's da placa **DE10-Lite** estão disponíveis no arquivo: **Associação de Pinos do FPGA da Placa DE10-Lite** (disponível nas referências de laboratório no Moodle).

**Atenção:** Não deixe conectores de saída (OUTPUT) **sem atribuição de pino**. O Quartus Prime associa automaticamente todos os conectores de saída a algum pino de I/O, **podendo ocorrer conflito** de um pino escolhido pela ferramenta com um sinal já utilizado na placa **DE10-Lite**.

O aluno pode utilizar a **Sala de Projetos (sala D3-07 ou D3-09)** para testar seu projeto antes da aula de apresentação (a chave da sala e a placa **DE10-Lite** podem ser solicitadas no almoxarifado do **CLE**).

## 8. Critérios de Elaboração do Relatório do Projeto

O projeto deve ser desenvolvido **individualmente**. Projetos copiados, total ou parcialmente, não serão considerados válidos, sendo atribuído zero para todos os projetos copiados;

As avaliações das apresentações e os relatórios serão individuais, sendo que cada aluno deve demonstrar o perfeito conhecimento de todas as etapas do projeto e teste;

O relatório deve ser entregue no formato de **arquivo em PDF (enviado pelo MOODLE)** e deve conter a documentação detalhada do projeto, permitindo que este seja entendido e executado por um outro projetista. O relatório deve apresentar, no mínimo, os seguintes itens:

1. **Capa** com o nome, número do aluno e turma;

### 2. Parte 1: Descrição do Problema:

A introdução deve apresentar um resumo dos objetivos do projeto com a especificação das características particulares da implementação realizada pelo aluno (tabela verdade preenchida com o código da disciplina e do número da matrícula do aluno).

### 3. Parte 2: Descrição da Realização do Projeto:

Devem ser descritos os vários elementos que compõem o projeto. Devem ser apresentados **em detalhe no mínimo** os seguintes elementos:

- **Tabela Verdade**: com as condições das saídas D0 C0 B0 A0 e D1 C1 B1 A1 necessárias para a apresentação os números do aluno;
- **Mapas de Karnaugh**: com as expressões resultantes da lógica de decodificação para os números do aluno;
- **Diagrama Esquemático-Lógico** completo do projeto do sistema, mostrando a interligação entre todos os elementos do projeto (portas lógicas, multiplexadores) e os pinos dos sinais de entrada e saída (gerada com “Print Screen” do editor de blocos do Quartus Prime – arquivo.bdf);
- **Formas de onda da simulação funcional** apresentadas na **Aula 7 de projeto** (geradas com “Print Screen” do simulador do Quartus Prime – arquivo.vwf);

### 4. Conclusão:

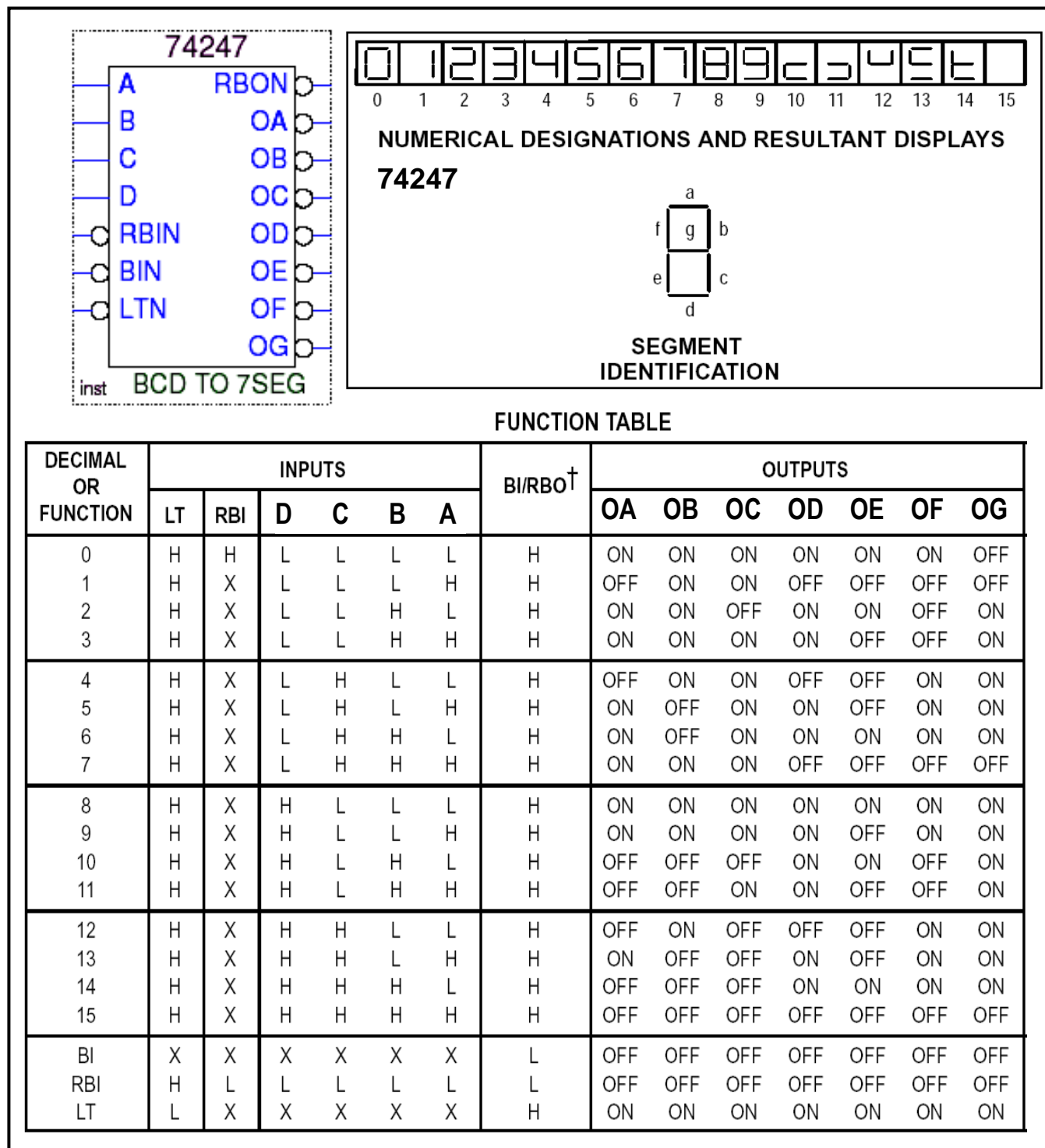
Na conclusão do relatório podem ser comentados: os objetivos do Projeto 1; as dificuldades encontradas e como foram superadas; o que o aluno faria de diferente se tivesse que executar esse projeto novamente.

**ATENÇÃO:** O *upload* de arquivos no Moodle é de inteira responsabilidade do aluno, que deve verificar se os arquivos foram efetivamente postados após o *upload*. Não serão aceitos arquivos postados fora do horário especificado para *upload* da atividade.



## 9. Especificação Funcional do Decodificador 74247

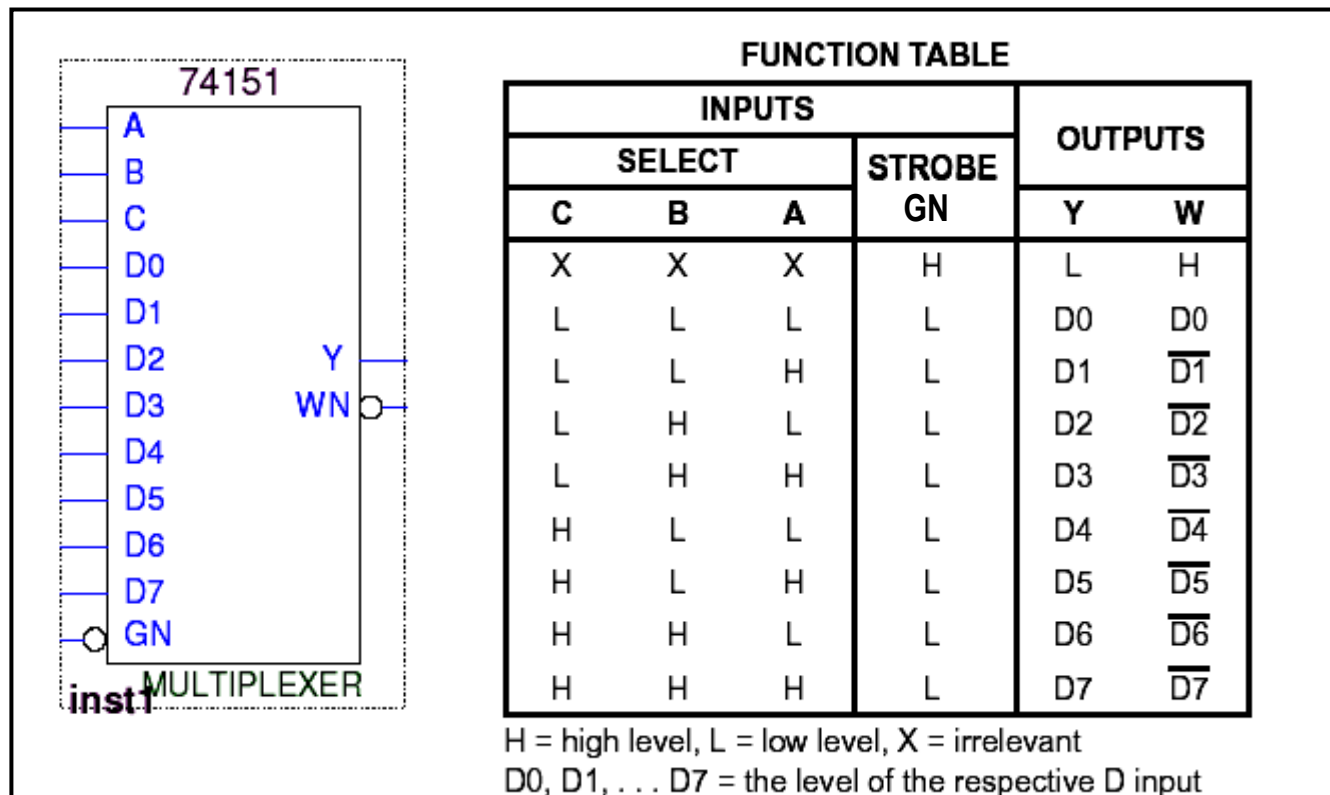
**Figura 3:** Tabela funcional e símbolo lógico do decodificador BCD para display de sete segmentos (74247).



Fonte: TEXAS (A), 2020 (adaptado).

## 10. Especificação Funcional do Multiplexador 74151

**Figura 4:** Tabela funcional e símbolo lógico do multiplexador de oito entradas (74151).



Fonte: TEXAS (B), 2020 (adaptado).

## 11. Referências Bibliográficas

TOCCI, Ronald J; WIDMER, Neal S.; MOSS, Gregory L. **Sistemas Digitais: Princípios e Aplicações**. Revisão técnica: Renato Giacomini. Tradução: Jorge Ritter. 11. Ed. São Paulo: Pearson Prentice Hall, 2011.

QUARTUS – Intel Quartus Prime – Lite, Version 16.1. Intel-FPGA (©Intel Corporation). Disponível em: (<https://fpgasoftware.intel.com/?edition=lite>). Acesso em: 07/07/2020.

PRATES, R. R. – Tutorial de Quartus Prime para Projeto de CPLD/FPGA (baseado em Captura de Esquemático). Programa de Iniciação Didática – Centro Universitário FEI, 2019.

TERASIC - DE10-LITE Board - User Manual. (©Terasic Inc). Disponível em: (<https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=234&No=1021&PartNo=4>). Acesso em: 06/07/2020.

TEXAS (A) – SN74LS247 – BCD-To-Seven-Segment Decoders/Driver datasheet. (©Texas Instruments Inc). Disponível em: (<https://www.ti.com/product/SN74LS247>). Acesso em 16/07/2020

TEXAS (B) – SN74LS151 – 8-Line To 1-Line Data Selectors/Multiplexers datasheet. (©Texas Instruments Inc). Disponível em: (<https://www.ti.com/product/SN74LS151>). Acesso em 16/07/2020