

## Experiência 4 – Projeto de Sistemas Digitais Auxiliado por Computador

### 4.1) Objetivos

- Familiarizar o aluno com o projeto de sistemas digitais auxiliados por computador (CAE);
- Realizar a execução completa de um projeto de sistema digital e sua simulação utilizando uma ferramenta de projeto auxiliado por computador - Quartus Prime Lite – Versão 16.1 - Intel-FPGA (© Intel Corporation).
- QUARTUS PRIME-LITE);
- **Atividade individual**.

### 4.2) Material Necessário

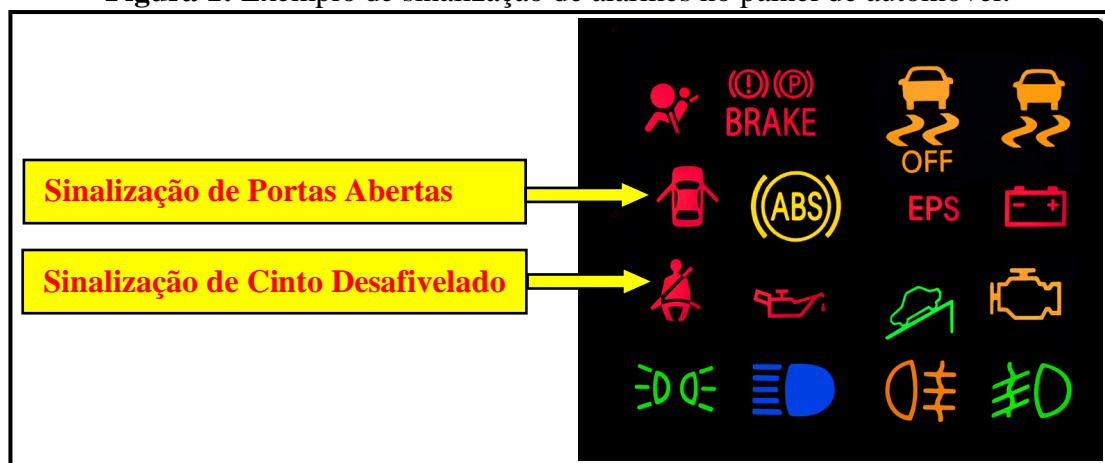
- Acesso ao software QUARTUS PRIME LITE - Versão 16.1 (© Intel Corporation).  
(<https://fpgasoftware.intel.com/?edition=lite>)  
Consulte os documentos “**Procedimento para Download e Instalação do Quartus Prime**” e “**Tutorial para Instalação do Quartus Prime Lite**” disponíveis nas “**Referências de Laboratório**” no Moodle.
- Documento: **Tutorial\_Quartus\_Prime\_para\_Projeto\_com\_Captura\_de\_Esquemático** disponível nas “**Referências de Laboratório**” no Moodle.

### 4.3) Procedimento Experimental

Conforme as instruções apresentadas nos capítulos 5, 6 7 e 8 do arquivo: **Tutorial\_Quartus\_Prime\_para\_Projeto\_com\_Captura\_de\_Esquemático** o aluno deve desenvolver e simular o projeto de um sistema de sinalização automotiva denominado **Luzes de Alerta**.

Esse sistema deve gerar três sinais de advertência para um painel automotivo (representado esquematicamente na **Figura 1**). Um sinal luminoso deve acender se existir alguma porta aberta, outro sinal luminoso deve acender se algum cinto de segurança não estiver afivelado e um sinal auditivo deve sinalizar que ao menos uma das situações anteriores ocorreu.

**Figura 1:** Exemplo de sinalização de alarmes no painel de automóvel.



Fonte: Autor.

Essas sinalizações devem ocorrer nas seguintes situações:

- **Sinalização de porta aberta**: deve ocorrer se a **chave de ignição está ligada** e uma das **portas está aberta**;

- **Sinalização de cinto desafivelado:** deve ocorrer se a chave de ignição está ligada e o cinto de segurança não está afivelado;
- **Alarme sonoro:** deve ocorrer se ao menos uma das sinalizações anteriores ocorreu.

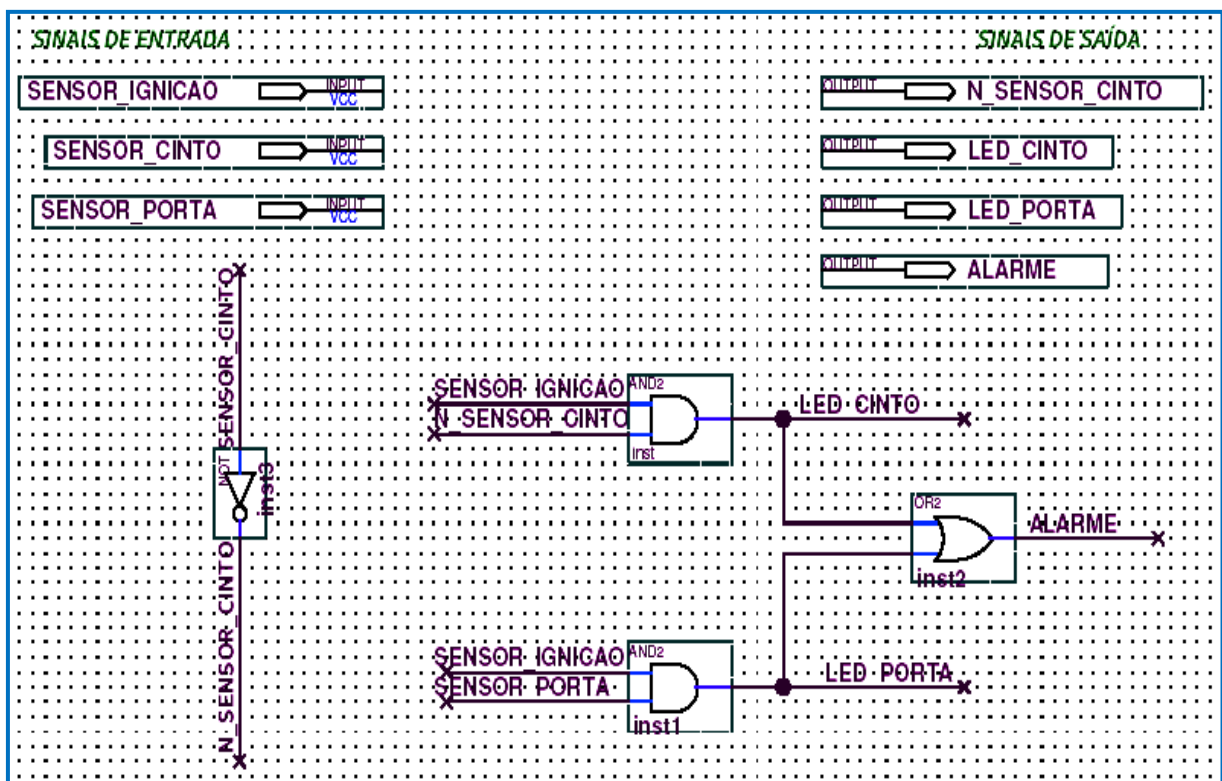
A ocorrência de cada um desses eventos deve ser sinalizada individualmente por saídas lógicas ligadas a Led's (**Led\_Porta** = '1' e **Led\_Cinto** = '1'), bem como deve existir uma saída lógica para a sinalização auditiva global (**Alarme** = '1'), indicando a ocorrência de qualquer um desses eventos.

Para a realização desse projeto consideraremos que existem os seguintes sinais lógicos de entrada:

- **Chave\_Ignicao:** se a chave de ignição está ligada esse sinal está em nível lógico 1;
- **Sensor\_Cinto:** se o cinto de segurança está afivelado este sinal está em nível lógico 1;
- **Sensor\_Porta:** se existe ao menos uma porta aberta este sinal está em nível lógico 1.

O diagrama esquemático-lógico da **Figura 2** apresenta uma possível realização do projeto **Luzes de Alerta** utilizando portas lógicas padronizadas (AND, OR e NOT). Nesse circuito foi incluída uma saída adicional para indicar, com lógica negada, a condição do sensor do cinto de segurança (**N\_Sensor\_Cinto**), de modo que essa saída é acionada (em nível lógico 1) quando o cinto de segurança está desafivelado.

**Figura 2:** Diagrama Esquemático-Lógico do circuito do Projeto Luzes\_de\_Alerta.



Fonte: Autor.

Esse projeto deve ser realizado no ambiente da ferramenta Quartus Prime-Lite (© Intel Corporation) e deve ser apresentada uma simulação funcional que demonstre o correto funcionamento do sistema.

**Observações sobre a implementação em FPGA:**

- O projeto deve ser criado no ambiente virtual (ambiente da ferramenta Quartus Prime Lite) e sua implementação baseia-se nos blocos lógicos de um único dispositivo lógico programável (**FPGA** - *Field Programmable Gate Array*), que substituem as portas lógicas padronizadas por células internas ao dispositivo. Dessa forma temos o sistema completo sendo **realizado por um único componente (SoC – System on Chip)**.
- Este método de projeto será utilizado como base para a implementação em FPGA da lógica proposta nos Projetos 1 e 2 do Laboratório de Sistemas Digitais.

**4.4) Procedimento de Projeto**

Utilizando a metodologia descrita no documento: **Tutorial\_Quartus\_Prime para Projeto com Captura de Esquemático** o aluno deve abrir um projeto no ambiente da ferramenta Quartus Prime, descrever o **Diagrama Esquemático-Lógico** do circuito **Luzes de Alerta** e realizar a simulação funcional do projeto.

O projeto do sistema de **Luzes de Alerta** deve ser desenvolvido individualmente para ser implementado no dispositivo lógico programável **FPGA** da família **MAX10** – Modelo: **10M50DAF484C7G** (dispositivo existente na placa de desenvolvimento **DE10-Lite** da THERASIC © Terasic.com).

Para a implementação desse projeto no ambiente do Quartus Prime-Lite devem ser executadas as seguintes etapas principais (para realizar essas etapas em detalhe deve ser seguido o procedimento do arquivo: **Tutorial\_Quartus\_Prime para Projeto com Captura de Esquemático**):

- a) Descrição do circuito com diagrama de blocos (*New> Design Files> Block Diagram*);
- b) Compilação do projeto no ambiente da ferramenta (*Processing> Start Compilation ou Processing> Start Analysis & Synthesis*);
- c) Descrição dos sinais lógicos para teste (*New> Verification/Debugging Files > University Program VWF*);
- d) Simulação funcional da lógica (*New> University Program VWF > Simulation Waveform Editor > Simulation > Run Functional Simulation*);
- e) Arquivamento do projeto (*Project> Archive Project*).

**4.5) Registro de Resultados**

Cada aluno deve postar no Moodle um relatório no formato **PDF** (segundo o **Modelo de Relatório de Projeto**) com no mínimo os seguintes itens:

- a) **Descrição do projeto**;
- b) **Tabela Verdade do sistema**;
- c) **Expressões lógicas minimizadas**;
- d) **Diagrama esquemático** do circuito gerado no Quartus Prime-Lite (arquivo.bdf);
- e) **Formas de onda da simulação funcional**, mostrando o comportamento do circuito para as oito combinações das variáveis de entrada (arquivo.vwf);
- f) **Conclusão** com a **comparação dos resultados** da simulação do Quartus Prime com as simulações no FALSTAD. Comentar vantagens e dificuldades encontradas.

Observação: O modelo desse relatório (disponível no Moodle) tem o formato MS-Word para facilidade de preenchimento, entretanto **deve ser postado no Moodle no formato PDF** (utilizar o recurso do MS-Word de *salvar como PDF*)

### 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.

FALSTAD, P. – Circuit Simulator, Version2.27js, GNU General Public License as published by the Free Software Foundation. Disponível em: (<https://www.falstad.com/circuit/circuitjs.html>). Acesso em: 04/07/2020.

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.

=====

### **ATENÇÃO: PREPARAÇÃO DO PROJETO 1:**

As atividades realizadas na **experiência 4** são essenciais para a realização da simulação do **Projeto 1** do laboratório, atividade individual a ser realizada na **próxima semana**.

**Para a realização do Projeto 1 na próxima aula de laboratório cada aluno deve analisar as especificações de projeto e preparar-se para a aula (antes da aula de laboratório) de modo a poder discutir com o professor as características das Tabelas Verdade necessárias para o desenvolvimento dos componentes do sistema.**

**Vide roteiro para a elaboração do Projeto 1 disponível no Moodle.**