



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DE SANTA CATARINA

CAMPUS SÃO JOSÉ

Disciplina: Dispositivos Lógicos Programáveis 1

Professor: Marcos Moecke

Alunos: Alex Magno, Theodor Konrad Wojcikiewicz,
Guilherme Roque

Relatório Grupo Semafórico

São José, 12 de dezembro de 2018.

Sumário:

Sumário:	1
Introdução:	2
Fundamentação Teórica	3
O Sistema de Grupo Semafórico	3
Figura 2 - Grupo semafórico de pedestres	4
Funcionamento	5
Estado padrão - Sem solicitação de pedestres	6
Estado 1 - Com solicitação de pedestres	6
Desenvolvimento:	8
Resultados	8
Componentes	11
Divisor de Clock (div_clk)	11
Sensor de Fluxo (sensor)	12
Display (display)	13
Conversor Binário para SSD (bin2ssd)	14
Contador (contador)	15
Máquina de Estado (maquinaDeEstado)	16
Hardware e Pinagens	18
Conclusão	27
Bibliografia:	28
Anexos	29

Introdução:

Tendo em vista a falta de sincronia dos semáforos presentes em nossa cidade, buscamos desenvolver uma alternativa, baseada em FPGA, utilizando o princípio de máquinas de estado.

No presente momento, as sinalleiras utilizam sistemas complexos que exigem sincronização, tornando o controle de trânsito ineficiente e totalmente dependente da central.

Além disso, os sistemas de controle de trânsito de carros e pedestres não são inteligentes, o que cria situações que poderiam ser evitadas.

Situações como semáforos de veículos abertos, sem fluxo de carros, enquanto pedestres desejam atravessar. Ou, por outro lado, sinalleiras de carro fechadas e de pedestres abertas sem fluxo de pedestres mas com alto fluxo de veículos.

Este projeto visa a criação de uma alternativa, um grupo de semáforos inteligente, formado por 4 sinalleiras, sensores que verificam a presença de carros, e displays de contagem de tempo.

Esta alternativa verifica o fluxo de veículos e o botão de pedestres, para permitir a passagem de um ou de outro, de acordo com a necessidade. Além disso, as sinalleiras possuem contagem de tempo próprias, que permitem que um pedestre transite com segurança em ambas as vias.

Fundamentação Teórica

As máquinas de estado são uma alternativa para a programação convencional de circuitos sequenciais, pois utilizam a abstração matemática para criar algoritmos. Estes, por sua vez, são convertidos para a linguagem de programação desejada e utilizados para solucionar determinado problema.

De maneira geral, são sistemas que recebem várias entradas e redirecionam para determinados estados pré estabelecidos, utilizando-se de condições já determinadas anteriormente.

O Sistema de Grupo Semafórico

Este projeto consiste em dois sistemas de sinaleiras, uma para carros e outra para pedestres, que juntas fazem parte do grupo semafórico. O sistema foi totalmente desenvolvido utilizando os princípios de máquina de estado para organização do fluxo, na busca de uma melhor mobilidade no trânsito.

Consiste, em sua composição, de dois sensores de fluxo de veículos, dois botões de requisição de passagem de pedestres e quatro sinaleiras, duas delas possuem três estados (amarelo, verde e vermelho) e as outras apenas dois (vermelho e verde).

Além disso, ambos os sistemas possuem *displays* para transmitir o tempo restante em segundos (conforme Figura 1 e 2). O grupo semafórico de pedestres ainda possui um botão para requisitar a passagem na via, em ambos os lados.

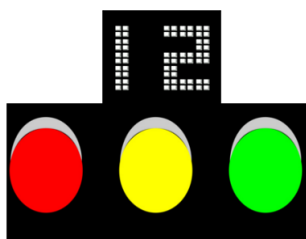


Figura 1 - Grupo semafórico de carros

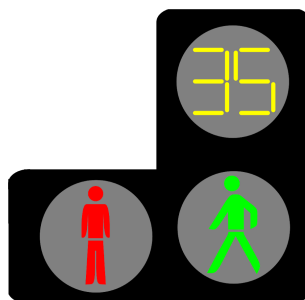


Figura 2 - Grupo semafórico de pedestres

Funcionamento

O sistema será instalado em uma via de mão dupla, com velocidade máxima permitida (V_{max}), juntamente com um sensor de movimento de veículos. Estes serão dispostos em cada lado da via a uma distância (D_s) da faixa de segurança.

Em cada lado da faixa teremos um GSP voltado para a calçada oposta, que facilita a leitura da informação pelos pedestres. Além disso, um GSC localizado no centro da via, permite a visualização pelos motoristas que vêm dos dois lados.

A distância do sensor do semáforo deve ser:

- Suficiente para que um veículo, já tendo alcançado o sensor, consiga passar pelo semáforo à velocidade mínima da via antes de o sinal fechar;
- Segura para que um veículo em alta velocidade, que não tenha alcançado o sensor, possa parar sem dificuldades antes do sinal fechar;

Legenda:

- 1 - Sensor de presença
- 2 - GSP
- 3 - GSC

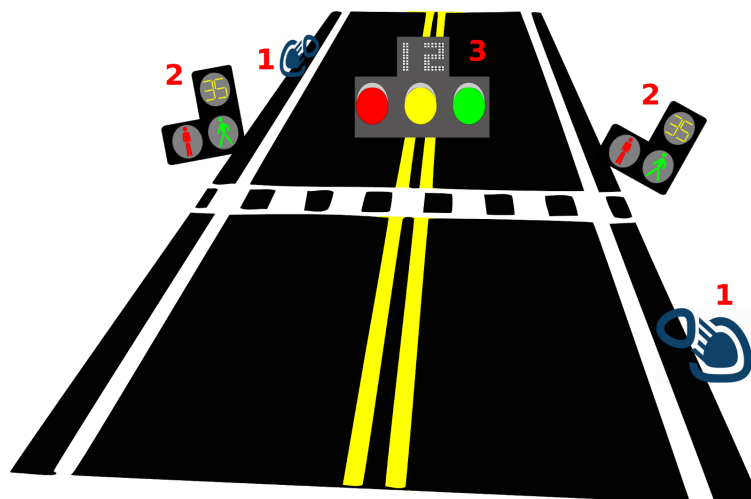


Figura 3 - Sistema de grupo semafórico

Estado padrão - Sem solicitação de pedestres

Neste caso o GSC permanece em espera sem nenhuma indicação, com passagem livre para os veículos. O GSP, por outro lado, indica vermelho para a passagem de pedestres.

Estado 1 - Com solicitação de pedestres

Considerando que o sistema esteja no estado padrão, ao receber uma solicitação de passagem de pedestre, este verifica o sensor de fluxo de veículos (entre a faixa de pedestre o semáforo). Após a verificação, se:

- Não houver sinal indicando presença de veículos nos últimos (T_c) segundos, o GSC indica amarelo por (T_{am}) segundos e, em seguida, vermelho.
- Detectar presença de veículos nos últimos (T_c) segundos, o GSC inicia uma contagem regressiva de (T_c) segundos. Logo após, indica um sinal amarelo por (T_{am}) segundos e, em seguida, vermelho.
- O GSC muda para vermelho, o GSP indica verde para pedestres por (T_p) segundos.
- Terminar os (T_p) segundos, o GSP permanece vermelho, enquanto o GSC troca para verde, por (T_{vd}) segundos. Neste caso se:
 - em algum momento o fluxo de veículos cessar, com solicitação de pedestres para atravessar a via, o GSC inicia seu processo. Neste instante ele fecha para então o GSP abrir.
 - em nenhum caso o fluxo de carros parar, com solicitação de pedestres, o GSC inicia seu processo. Ele então fecha apenas após o término dos (T_{vd}) segundos.
 - momento algum existir solicitação de pedestres, após os (T_{vd}) segundos o sistema retorna para o estado padrão.

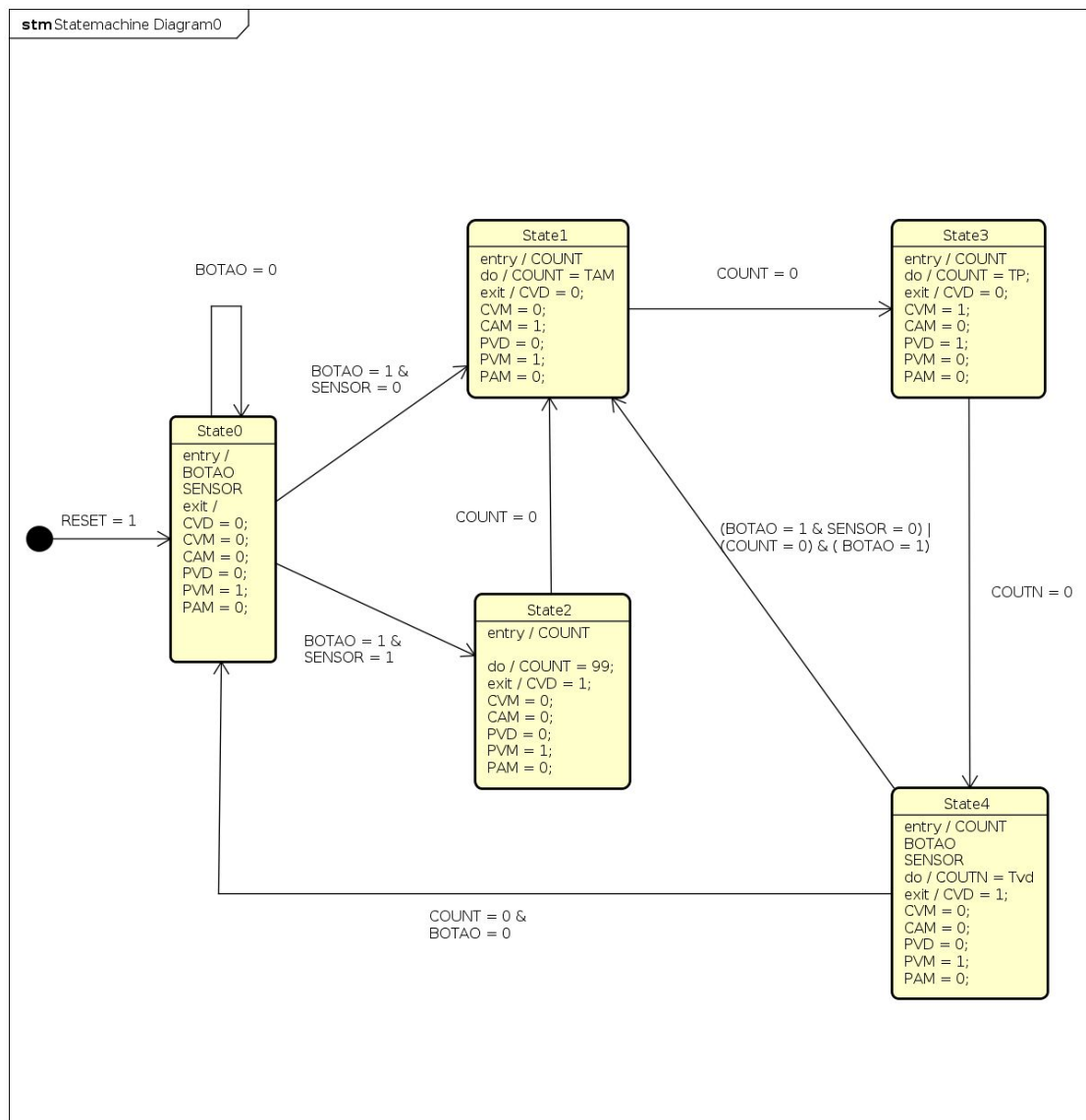


Figura 4 - Diagrama de estados do sistema

Vale ressaltar que o diagrama acima (Figura 4) o intuito de apenas representar, de maneira geral, os estados da máquina.

Alguns daqueles sinais foram trocados durante a etapa de desenvolvimento, permanecendo como ativos alto. O reset do sistema, leds de iluminação da via, indicadores de fluxo (sensor) e botões de solicitação dos pedestres foram os sinais que tiveram esta mudança. Vale lembrar também, que a variável count representa a contagem regressiva do tempo em segundos e que pode ser visualizada nos displays de sete segmentos disponíveis na via.

Desenvolvimento:

Para o desenvolvimento do Grupo Semafórico foi utilizado o software Quartus II edição 13, disponibilizado pela Intel.

O sistema foi programado utilizando os princípios de máquina de estado e foi compilado na linguagem de hardware VHDL (Very High Speed Integrated Circuit Hardware Description Language).

É uma linguagem que permite declarações concorrentes e sequenciais, além de instanciação de componentes e fácil incorporação no hardware (baixo nível).

O software Quartus permite realizar simulações sem a necessidade de hardware, diretamente no simulador incorporado. Além disso, permite trabalhar com softwares integrados como o Multisim, utilizado na simulação de nosso sistema.

Após os testes, o código é implementado no kit DE2-115 que utiliza o FPGA Cyclone IV E da fabricante Altera, para síntese do circuito físico.

Resultados

Após a compilação do código e da configuração das condições de simulação (realizadas nos programas acima), foi possível coletar os dados descritos neste tópico.

Nosso sistema é formado por apenas uma máquina de estado, responsável por indicar quais semáforos abrem e quais fecham, verificar os sensores de fluxo de veículos, observar se algum pedestre deseja atravessar, além de mostrar nos displays os tempos de cada estado.

Na imagem abaixo (Figura 5), podemos visualizar o circuito gerado pelo código VHDL, formado por 158 elementos lógicos e 57 pinos utilizados.

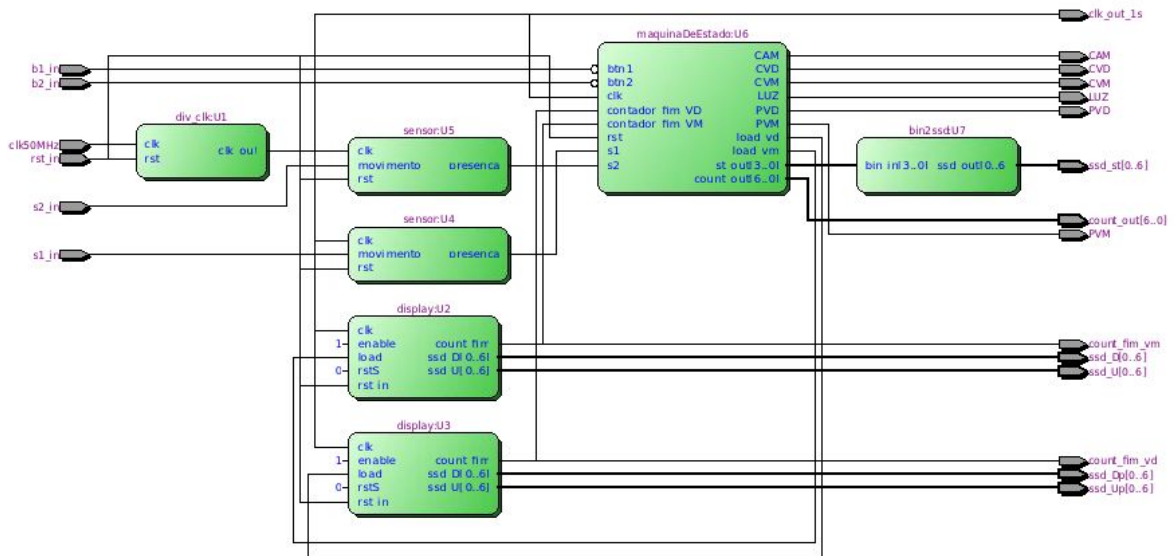


Figura 5 - Circuito completo do sistema

Neste circuito podemos observar os componentes dos quais fazem parte o sistema.

Em primeiro lugar temos a máquina de estado (cujo componente possui o mesmo nome, U6), como principal controlador do sistema. Em seguida podemos ver o divisor de clock (div_clk, U1), como responsável pela divisão do clock de entrada.

Visualizamos ainda os sensores (U4 e U5) e os displays (U2 e U3), responsáveis pelo processamento das entradas (s1_in e s2_in) e pela exibição dos contadores de pedestres e veículos, respectivamente.

Além destes, ainda temos as entradas do circuito (b1_in e b2_in), que simulam os sinais dos botões de solicitação dos pedestres e o clock do FPGA utilizado (clk50Mhz). Observamos também o reset geral do sistema (rst_in) e o sinal responsável pela simulação dos fluxo de veículos pelos sensores(s1_in e s2_in).

Por fim, temos as saídas do circuito, dos *displays* de sete segmentos (ssd_Up, ssd_Dp, ssd_U, ssd_D) e dos estados da sinaleira (PVD, PVM, LUZ, CVM, CVD, CAM).

Lembrando que o visor de sete segmentos é responsável por mostrar o contador de tempo restante, tanto no semáforo dos veículos (GSC), quanto no dos

pedestres (GSP). Com relação ao estados da sinaleira, nos referimos às cores (verde, amarelo e vermelho para carros e verde e vermelho para pedestres).

Ressaltamos, ainda, que os componentes bin2ssd (U7) e as saídas count_out e clk_out_1s serviram apenas testes de implementação na placa.

Por fim, após a compilação dos componentes acima, na linguagem VHDL, realizamos os testes no simulador MultiSim, para buscar quaisquer tipos de erros, antes da implementação no kit FPGA (conforme figura 6).

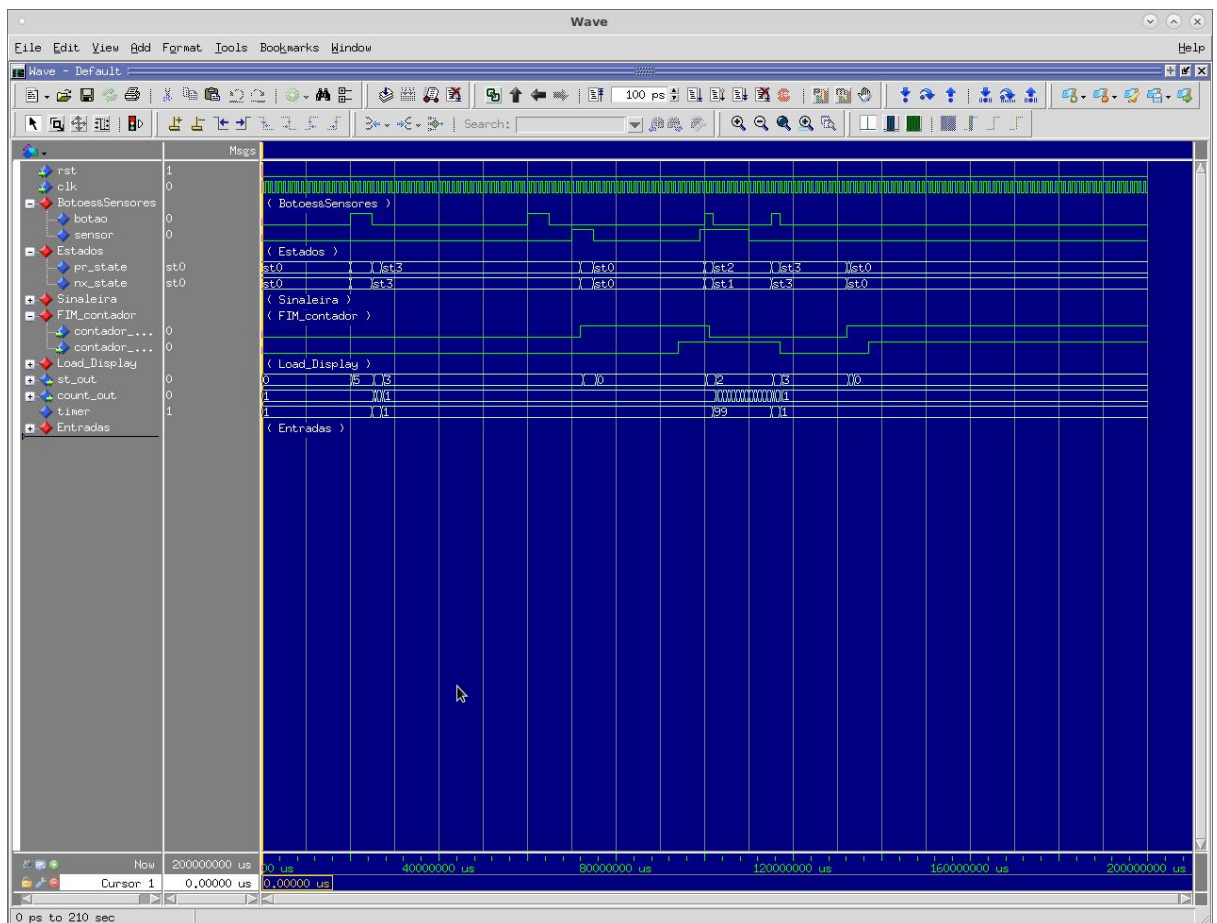


Figura 6 - Simulação do Sistema no Multisim

Componentes

No tópico abaixo, uma breve descrição dos componentes utilizados em nosso sistema, especificados de acordo com suas funções.

Cada componente é o equivalente a um programa próprio (.vhd) e juntos formam o sistema do Grupo Semafórico.

Divisor de Clock (div_clk)

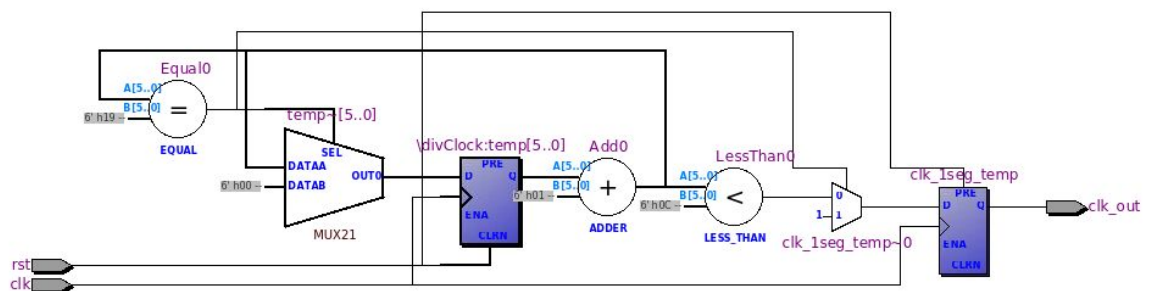


Figura 7 - Divisor de Clock

Entradas: clk,rst

Saídas: clk_out

Elementos lógicos: 13

Descrição: Responsável por criar um sinal de clock de período de 1 segundo a partir de um sinal de clock fornecido pela placa de 50MHz.

Funcionamento: Possui um contador interno que incrementa a cada transição do clock de entrada. Quando este alcança o valor de referente a meio segundo, o div_clk muda o estado da saída.

Sensor de Fluxo (sensor)

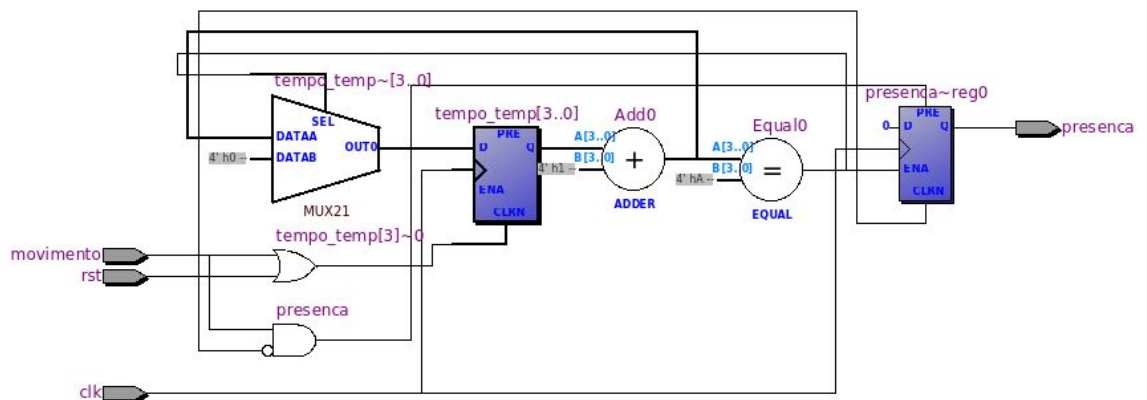


Figura 8 - Sensor de Fluxo de Veículos

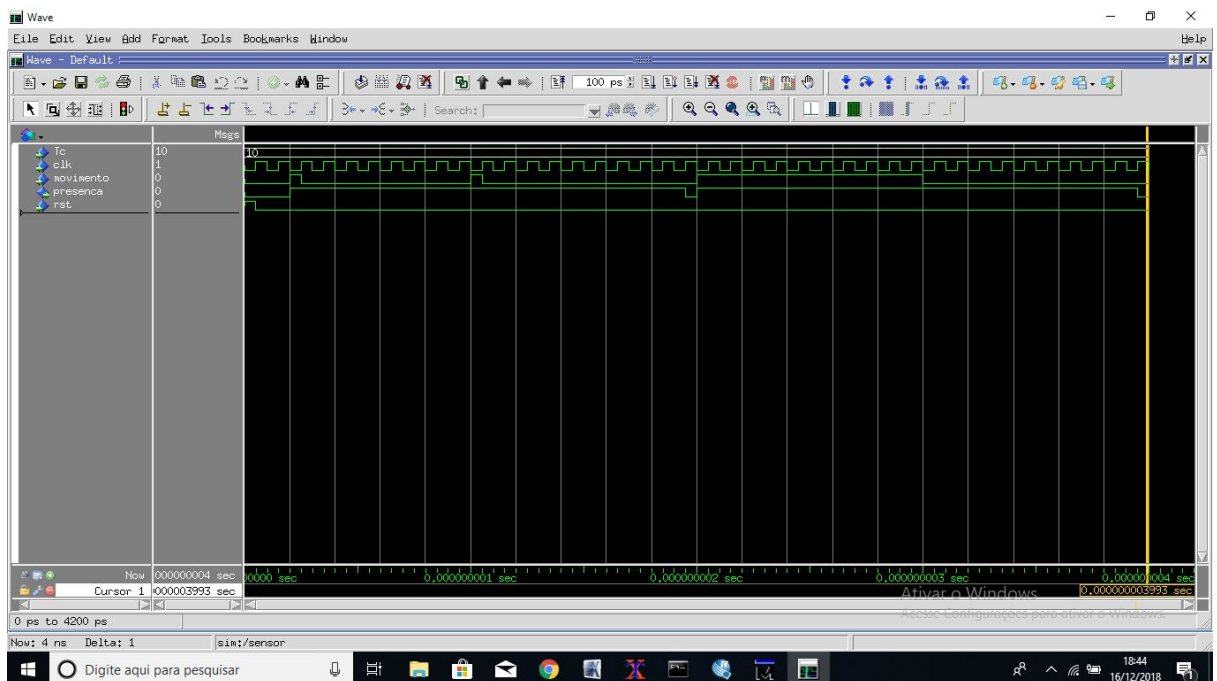


Figura 9 - Simulação do Sensor

Entradas: clk, movimento, rst

Saídas: presenca

Elementos lógicos: 10

Descrição: Detecta o movimento de veículos entre o sensor e a faixa de pedestre e retorna um valor (1 com fluxo, 0 sem fluxo).

Funcionamento: Cada vez que um veículo passa pelo sensor, um contador interno é zerado. Este contador é responsável por definir o tempo em segundos entre o evento anterior (último veículo a passar pelo sensor) e o momento atual.

Se houver a presença de veículos entre o sensor e a faixa de pedestre, a saída “presença” permanece ativa (nível lógico 1).

Após Tc segundos, o sensor indica que não há mais presença de veículos (nível lógico troca para 0).

Display (display)

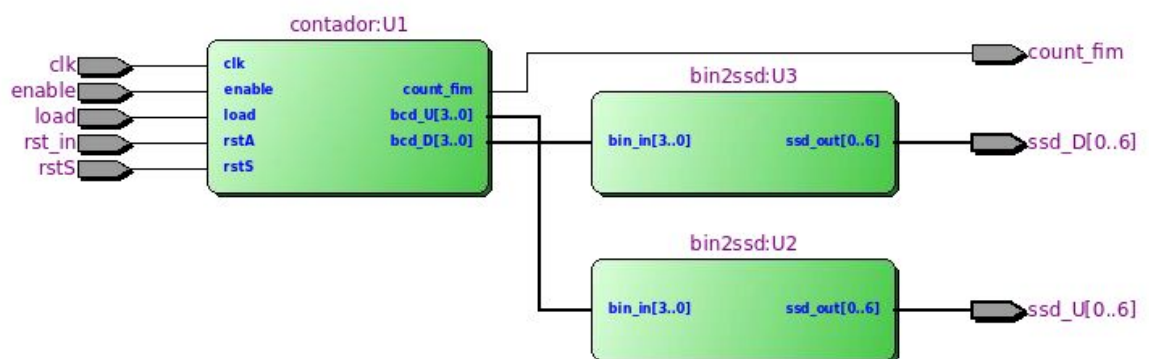


Figura 10 - Display

Entradas: clk, rst_in, enable, load, rstS,enable

Saídas: ssd_d, ssd_u, count_fim

Elementos lógicos: 33

Descrição: Componente responsável por agrupar os componentes do contador e do conversor de binário para ssd e transmiti-los para os displays de sete segmentos.

Funcionamento: O cliente define o tempo padrão em segundos, através da variáveis D e U (dezena e unidade, respectivamente). A partir deste ponto, o contador entra em ação quando a máquina de estado emite um sinal na entrada load e ele começa a decrementar, ao mesmo tempo que este valor é mostrado no

display (através das variáveis `ssd_d` e `ssd_u`). Vale ressaltar que este componente recebe os valores de outros dois e não realiza nada além de concentrar os sinais.

Conversor Binário para SSD (bin2ssd)

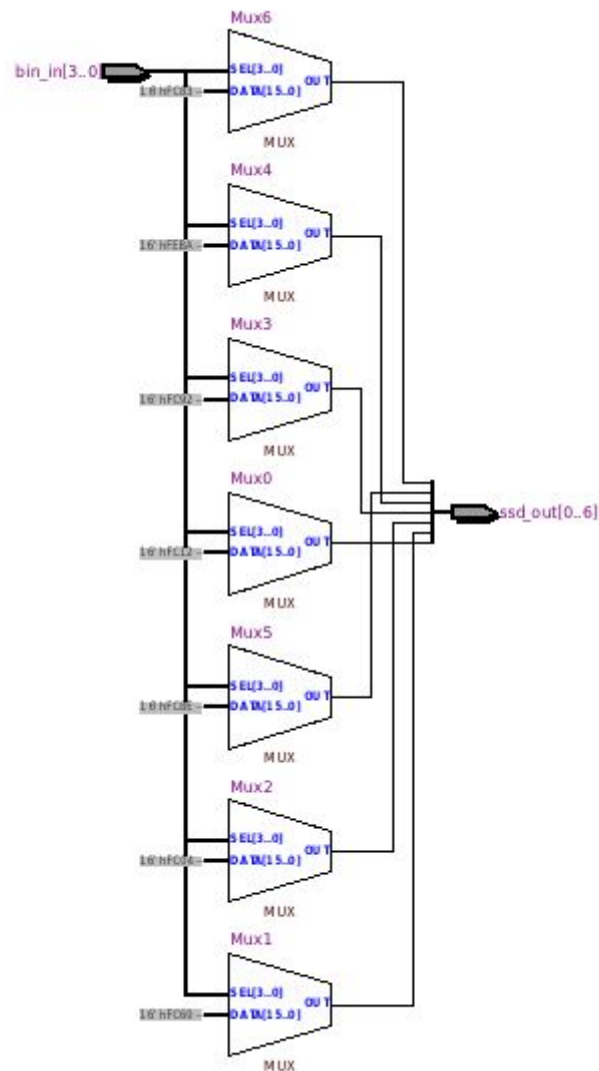


Figura 11 - Conversor Binário para SSD

Entradas: `bin_in`

Saídas: `ssd_out`

Elementos lógicos: 7

Descrição: Responsável pela codificação do display, através da conversão do sinal em binário para display de sete segmentos.

Funcionamento: Recebe o valor em binário e converte internamente para SSD (seven segment display), e transmite o valor final para a saída (ssd_out).

Contador (contador)

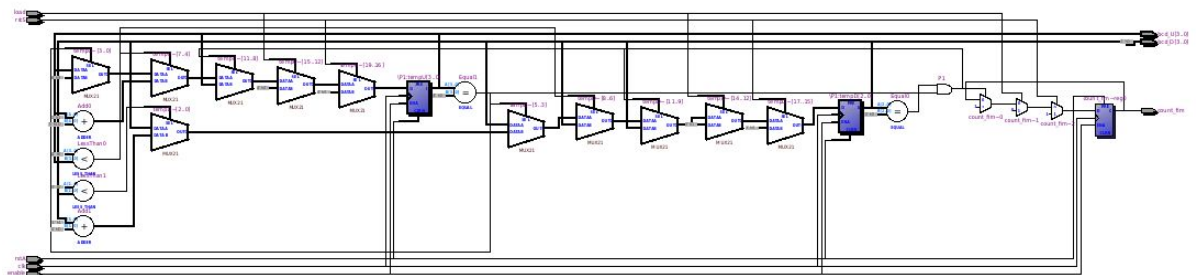


Figura 12 - Contador

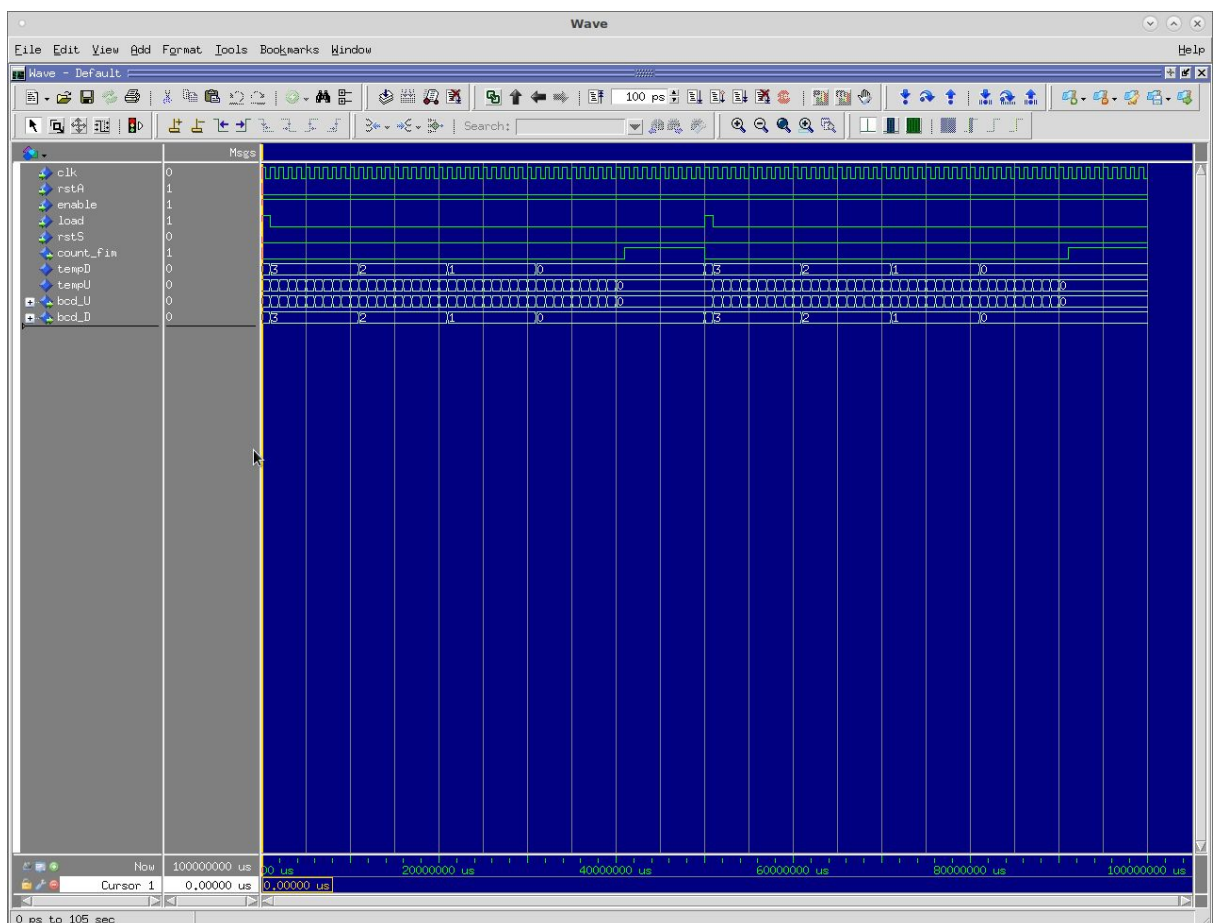


Figura 13 - Simulação do Contador

Entradas: clk, rstA, enable, load, rstS

Saídas: bcd_u, bcd_d, count_fim

Elementos lógicos: 19

Descrição: componente que realiza a contagem regressiva do tempo internamente, transmitindo o sinal para o próximo componente.

Funcionamento: após a definição do cliente do tamanho da contagem, através das variáveis genéricas D e U, inicia o decremento do tempo, transmitindo o valor para as saídas.

Máquina de Estado (maquinaDeEstado)

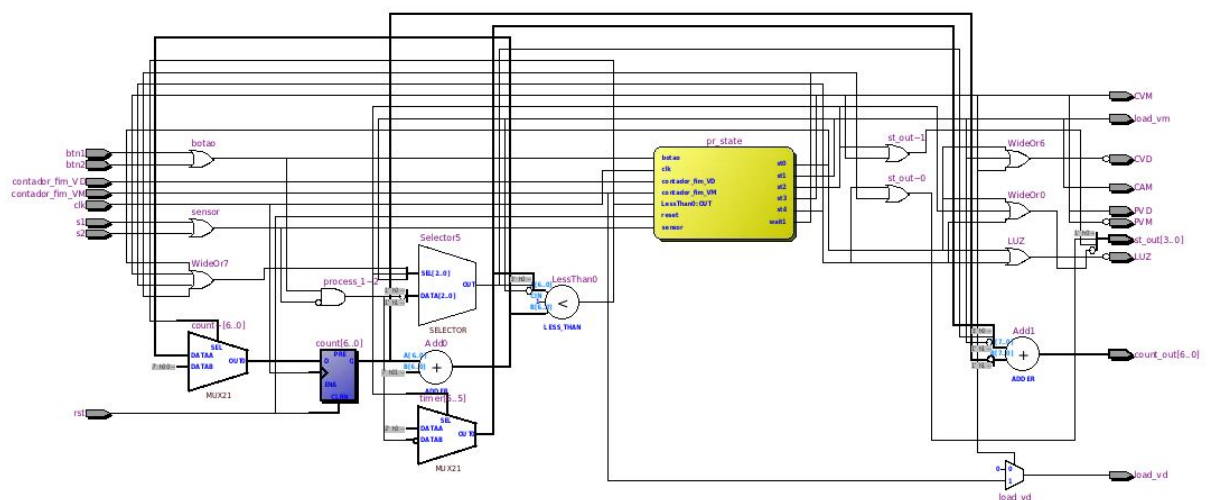


Figura 14 - Máquina de Estado

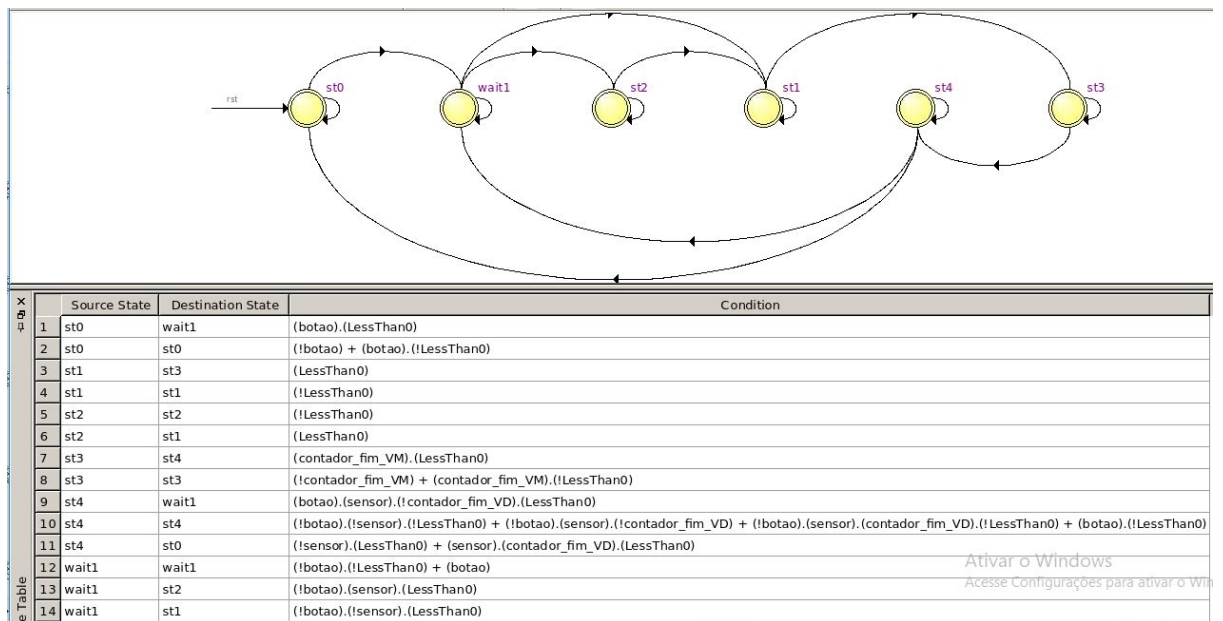


Figura 14 - Visualizador de Máquinas de Estado

Entradas: s1, s2, btn1, btn2, contador_fim_VD, contador_fim_VM

Saídas: load_vm, load_vd, CVD, CVM, CAM, PVD, PVM, LUZ

Elementos lógicos: 42 elementos

Descrição: é o componente mais importante do sistema, sendo responsável por controlar os estados do sistema.

Funcionamento: define os estados das sinaleiras, através dos algoritmos criados a partir da técnica de máquina de estado.

Hardware e Pinagens

Nosso sistema foi implementado na placa FPGA da Altera, modelo DE2-115, com a utilização dos leds, chaves e displays do próprio hardware.

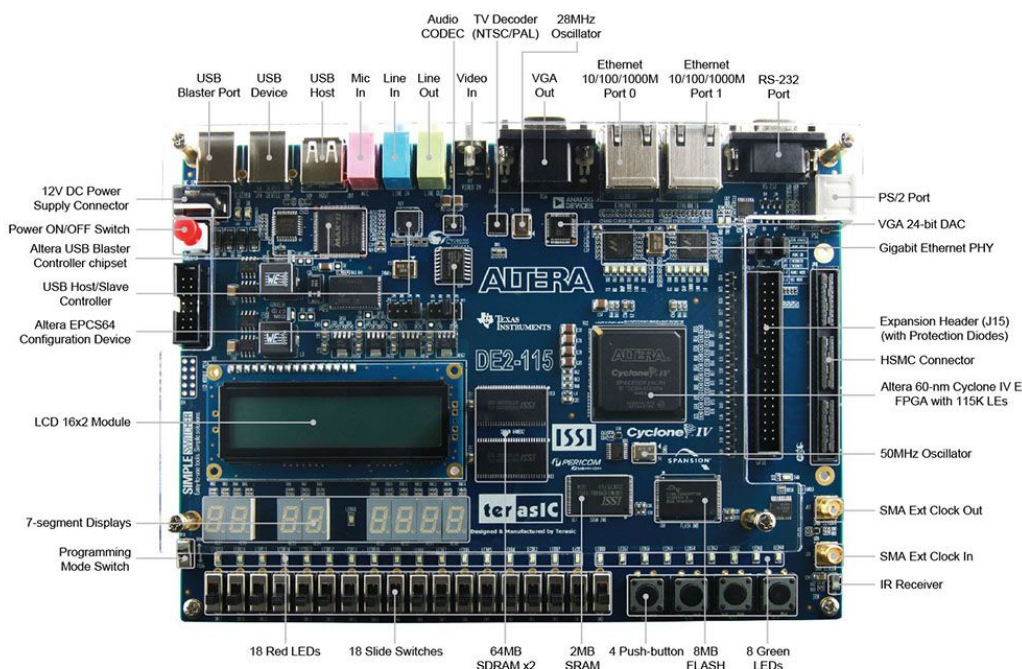


Figura 15 - Kit Altera DE2-115

Como podemos visualizar na imagem acima (Figura 15), esta placa possui várias opções de utilização, sendo necessário consultar a especificação da placa para buscar a melhor configuração para a nossa necessidade.

Na elaboração do Grupo Semaforico foi necessário utilizar duas chaves (slide switch) , cinco leds e dois botões (push button).

Os leds são utilizados para a visualização dos estados das sinaleiras que, no caso dos veículos, são três estados (verde, amarelo e vermelho) e no caso dos pedestres dois estados (verde e vermelho). No entanto, como o hardware não oferece leds de cor amarela, utilizamos a cor verde.

Considerando este pormenor, a configuração do sinaleiro de veículos (GSC) e dos pedestres (GSP) foram configuradas de acordo com a tabela abaixo:

Nome do Sinal	Número do Pino	Cor
LEDG[0]	PIN_E21	Verde Veículos
LEDG[8]	PIN_F17	Amarelo Veículos
LEDR[0]	PIN_G19	Vermelho Veículos
LEDG[1]	PIN_E22	Verde Pedestre
LEDR[1]	PIN_F19	Vermelho Pedestre
HEX0[6]	PIN_H22	Luz de Iluminação da via

Os leds de pinos PIN_E21, PIN_F17 e PIN_G19, são responsáveis por transmitir os estados da sinaleira de carros (GSC), com o primeiro led sendo verde, o segundo amarelo e o terceiro vermelho.

Ainda considerando a mesma tabela, podemos visualizar os leds responsáveis pelos estados do semáforo de pedestres (GSP). Utilizamos o PIN_E22 para verde e o PIN_F19 para vermelho. Para a iluminação da via durante a noite, utilizamos o PIN_H22.

Nosso sistema ainda prevê a utilização de sensores de fluxo de veículos, além de uma função que reseta tudo para a configuração de fábrica. Para estas funções designamos as seguintes chaves:

Nome do Sinal	Número do Pino	Função
SW[0]	PIN_AB28	Reset
SW[17]	PIN_Y23	Sensor de Fluxo de Veículos Via 1
SW[16]	PIN_Y24	Sensor de Fluxo de Veículos Via 2

Como podemos observar, a primeira chave, PIN_AB28, é responsável por realizar um reset completo do sistema. Este força o estado padrão do Grupo Semafórico, que é a sinaleira vermelha para pedestre e verde para veículos.

Quanto às outras chaves na mesma tabela (PIN_AC28 e PIN_AC27), são utilizadas para simular o sensor de presença de veículos, com a chave ligada (nível lógico 1) possui fluxo e com a chave desligada não possui (nível lógico 0).

Na tabela abaixo, temos a descrição dos botões utilizados pelos pedestres para requisitar a passagem na via.

Nome do Sinal	Número do Pino	Função
KEY[0]	PIN_M23	Botão Requisição de Passagem Pedestre Via 1
KEY[3]	PIN_R24	Botão Requisição de Passagem Pedestre Via 2

Como o sistema foi desenvolvido para duas vias, cada lado tem seu próprio botão, configurados como PIN_M23 E PIN_M21 respectivamente. Estes permanecem inativos (nível lógico 0) até a sua utilização.

Com relação aos displays, podemos observar nos quadros abaixo as configurações utilizadas, com as duas primeiras tabelas sendo vinculadas para os veículos e as duas últimas para os pedestres.

Apesar do sistema prever oito displays de sete segmentos, utilizaremos apenas quatro, vinculando dois deles para cada grupo semafórico (GSC e GSP), representando o tempo com unidade e dezena.

Display 1 Contagem de Tempo para Veículos			
Segmento	Nome do Sinal	Número do Pino	Função

a	HEX7[0]	PIN_AD17	Display Contagem Tempo Carro
b	HEX7[1]	PIN_AE17	Display Contagem Tempo Carro
c	HEX7[2]	PIN_AG17	Display Contagem Tempo Carro
d	HEX7[3]	PIN_AH17	Display Contagem Tempo Carro
e	HEX7[4]	PIN_AF17	Display Contagem Tempo Carro
f	HEX7[5]	PIN_AG18	Display Contagem Tempo Carro
g	HEX7[6]	PIN_AA14	Display Contagem Tempo Carro

Display 2 Contagem de Tempo para Veículos			
Segmento	Nome do Sinal	Número do Pino	Função
a	HEX6[0]	PIN_AA17	Display

			Contagem Tempo Carro
b	HEX6[1]	PIN_AB16	Display Contagem Tempo Carro
c	HEX6[2]	PIN_AA16	Display Contagem Tempo Carro
d	HEX6[3]	PIN_AB17	Display Contagem Tempo Carro
e	HEX6[4]	PIN_AB15	Display Contagem Tempo Carro
f	HEX6[5]	PIN_AA15	Display Contagem Tempo Carro
g	HEX6[6]	PIN_AC17	Display Contagem Tempo Carro

Display 1 Contagem de Tempo para Pedestres			
Segmento	Nome do Sinal	Número do Pino	Função
a	HEX5[0]	PIN_AD18	Display

			Contagem Tempo Pedestre
b	HEX5[1]	PIN_AC18	Display Contagem Tempo Pedestre
c	HEX5[2]	PIN_AB18	Display Contagem Tempo Pedestre
d	HEX5[3]	PIN_AH19	Display Contagem Tempo Pedestre
e	HEX5[4]	PIN_AG19	Display Contagem Tempo Pedestre
f	HEX5[5]	PIN_AF18	Display Contagem Tempo Pedestre
g	HEX5[6]	PIN_AH18	Display Contagem Tempo Pedestre

Display 2 Contagem Tempo para Pedestres			
Segmento	Nome do Sinal	Número do Pino	Função
a	HEX4[0]	PIN_AB19	Display

			Contagem Tempo Pedestre
b	HEX4[1]	PIN_AA19	Display Contagem Tempo Pedestre
c	HEX4[2]	PIN_AG21	Display Contagem Tempo Pedestre
d	HEX4[3]	PIN_AH21	Display Contagem Tempo Pedestre
e	HEX4[4]	PIN_AE19	Display Contagem Tempo Pedestre
f	HEX4[5]	PIN_AF19	Display Contagem Tempo Pedestre
g	HEX4[6]	PIN_AE18	Display Contagem Tempo Pedestre

Os displays utilizados nas tabelas acima seguem a lógica de funcionamento da figura abaixo, sendo configurados com sete segmentos, de zero a seis.

Ao transmitir um contador de tempo, é necessário transmitir dois números, formados por mais de um segmento (de zero a seis), totalizando 6 pinos utilizados por número. Nosso contador permite transmitir de zero a noventa e nove segundos e utilizado dois displays de sete segmentos como podemos visualizar na imagem abaixo.

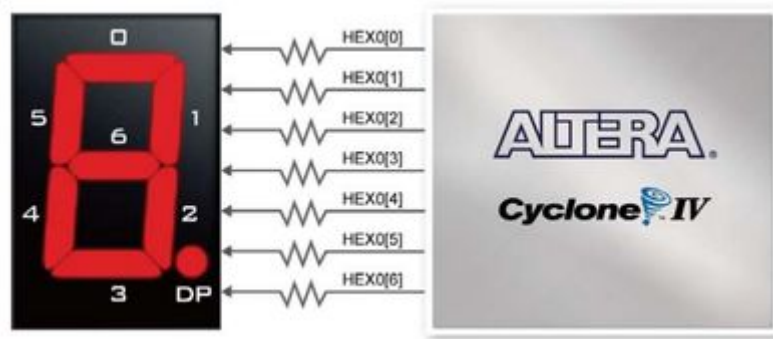


Figura 16 - Display de Sete Segmentos

Para os grupos semafóricos utilizamos os dois primeiros displays da placa para os veículos (HEX 7 e HEX 6) e os dois do meio para os pedestres (HEX 5 e HEX 4), conforme a imagem abaixo(Figura 17):



Figura 17 - Displays de Sete Segmentos disponíveis no Kit DE2-115

O último display (HEX0) foi configurado para iluminar a via para uma passagem mais segura dos pedestres. O display HEX3 foi utilizado para demonstrar o presente estado da via, de acordo com a máquina de estado, apenas para facilitar o teste na placa. Na tabela abaixo podemos visualizar essas configurações:

Display Máquina de Estado			
Segmentos	Nome do Sinal	Número do Pino	Função
a	HEX3[0]	PIN_V21	Display de Estado
b	HEX3[1]	PIN_U21	Display de Estado

c	HEX3[2]	PIN_AB20	Display de Estado
d	HEX3[3]	PIN_AA21	Display de Estado
e	HEX3[4]	PIN_AD24	Display de Estado
f	HEX3[5]	PIN_AF23	Display de Estado
g	HEX3[6]	PIN_Y19	Display de Estado

Conclusão

O projeto teve como objetivo avaliar o aprendizado dos conceitos ensinados durante o semestre, sobre o uso de projetos hierárquicos e de máquina de estados para desenvolver sistemas complexos em linguagem VHDL. Ele foi totalmente pensado e desenvolvido em um curto espaço de tempo, o que resultou em alguns problemas ao longo da sua implementação. Uma das dificuldades foi entender o que foi requisitado, além de deduzir qual a melhor forma para fazê-lo, de acordo com o que o cliente pediu, para então transmiti-lo para as especificações. Também foi um desafio elaborar a máquina de estado, pois gerou muitas dúvidas conceituais.

Como alguns componentes já haviam sido criados em projetos anteriores durante as aulas, demos mais ênfase na máquina de estado. Na primeira ideia, pensamos em criar duas máquinas de estados, uma responsável pelo controle geral do sistema e outra responsável pelo controle das sinaleiras. Após uma reestruturação do projeto, decidimos criar apenas uma, de forma temporal e atemporal, dependendo do caso. Isto se fez necessário para tratar algumas mudanças de estados, que poderiam ser acionadas tanto pelo fim da contagem regressiva (indicada por alguma sinaleira), quanto por alguma ação do pedestre (estas combinadas com as indicações dos sensores de presença de veículos).

Outro componente que gerou dificuldades foi o display, pois inicialmente o objetivo era criar um contador regressivo que deveria receber um sinal externo. Este serviria para iniciar sua contagem e, ao terminar, emitir um sinal para a máquina de estado. Foi então criado um flip-flop do tipo D com *load* para decrementar os valores e um *count_fim* para emitir o sinal ao terminar sua contagem. Para o correto funcionamento, o *load* deve acionar sempre um sinal de clock antes da mudança de estado.

Apesar das dificuldades apresentadas, o desenvolvimento do sistema se deu de forma satisfatória, atendendo completamente os requisitos do cliente.

Bibliografia:

Disponível em:
“<https://blog.markshead.com/869/state-machines-computer-science/>”. Acessado em 12 de dezembro de 2018.

Disponível em : <https://pt.wikipedia.org/wiki/VHDL>. Acessado em 13 de dezembro de 2018.

Disponível em:
“https://wiki.sj.ifsc.edu.br/wiki/index.php/Interfaces_de_entrada_e_saida_da_DE2-115”. Acessado em 13 de dezembro de 2018.

Disponível em:
“https://www.terasic.com.tw/attachment/archive/502/image/image_74_thumb.jpg”.
Acessado em 13 de dezembro de 2018.

Anexos

DESCRIÇÃO	NOME	VALOR
Número de vias	Nv	2
Velocidade mínima da via	Vmin	30 km / h
Velocidade máxima da via	Vmax	60 km / h
Distância do sensor do GSC	Ds	80 m
Tempo do sinal GSC amarelo	Tam	2 s
Tempo do sinal GSC verde	Tvd	90 s
Tempo de passagem do pedestre da via	Tp	5 * Nv
Tempo de passagem do veículo entre sensor e GSC	Tc	$Ds / (Vmin / 3,6)$

Tabela 4 - Tabela de variáveis