

UNIVERSIDADE DE SÃO PAULO
ESCOLA POLITÉCNICA
DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO E
SISTEMAS DIGITAIS
PCS3635 - LABORATÓRIO DIGITAL I



PLANEJAMENTO DA EXPERIÊNCIA 7

Felipe Luis Korbes - NUSP: 13682893
Henrique Eduardo dos Santos de Souza - NUSP: 13679972
João Felipe de Souza Melo - NUSP: 13682913

Turma: 5

Professor: Reginaldo Arakaki

Data da experiência: 28/02/2024

São Paulo

2024

Sumário

1. Introdução e Objetivos.....	1
2. Atividades pré-laboratório.....	1
2.1. Atividade 1: Projeto Lógico.....	1
2.1.1 Testes.....	5
2.2. Atividade 2 – Implementação e Síntese.....	12
3. Planejamento da aula prática.....	16
4. Relatório.....	17
4.1. Atividade 2: Testes e sinais de depuração.....	17
4.2. Atividade 3 – Desafio: Modificação do Sistema Digital.....	17
4.2.1. Testes e sinais de depuração.....	17

1. Introdução e Objetivos

Esta experiência consiste em projetar e implementar o circuito digital do Genius, em seu modo desafio “você inventa a sequência”, numa placa FPGA. A especificação é que o usuário, jogando nesse modo do jogo, seja capaz de jogar diversas rodadas, ao final de cada uma das quais há uma adição de uma jogada nova, que será salva na memória. Essa implementação também incorpora um mecanismo de timeout para demora na jogada, seja inédita seja já existente.

2. Atividades pré-laboratório

2.1. Atividade 1: Projeto Lógico

Iniciou-se a montagem do circuito mediante a adaptação do projeto da experiência 6, de forma incremental. Modificou-se a unidade de controle, com adição de estado e sinais de condição e controle para a nova proposta. Quanto aos estados, adicionou-se um denominado *mostra_led*, no qual o circuito exibe nos leds o conteúdo inicial da memória antes da primeira jogada da primeira rodada. A nova configuração da máquina de estados é descrita pelo diagrama da figura 1.

A fim de disponibilizar a visualização do sinal por 2 segundos, acrescentou-se um contador ao fluxo de dados, cujo RCO indica o fim deste estado de amostragem e o início da rodada. Os sinais de entrada, conta e reset deste contador são controlados pela Unidade de Controle com a seguinte lógica: Quando o estado atual é *mostra_led* então ambos são HIGH, caso contrário são LOW. As entradas e saídas da entidade de alto nível *circuito_jogo_base*, bem como sua lógica interna, foram ajustadas para acomodar essas alterações e respeitar as especificações do projeto e a interface proposta, ambos da apostila do experimento.

A estrutura de fluxo de dados do projeto é composta por dois contadores binários do tipo m com reset e contagens assíncronas (um para a contagem da memória, um para as rodadas) e contadores do tipo 163 (um para o timeout e outro para a amostragem do led da posição inicial), uma RAM que armazena as jogadas a serem realizadas em tempo de execução, um registrador que registra a jogada feita pelo jogador, dois comparadores, sendo um utilizado para comparar as jogadas e outro como controle para as rodadas, um *edge_detector*, uma porta OR que colaboram para o registro da jogada do jogador, e uma multiplexador controlado pelo sinal de conta do contador adicionado. A nova configuração do fluxo de dados encontra-se na figura 2.

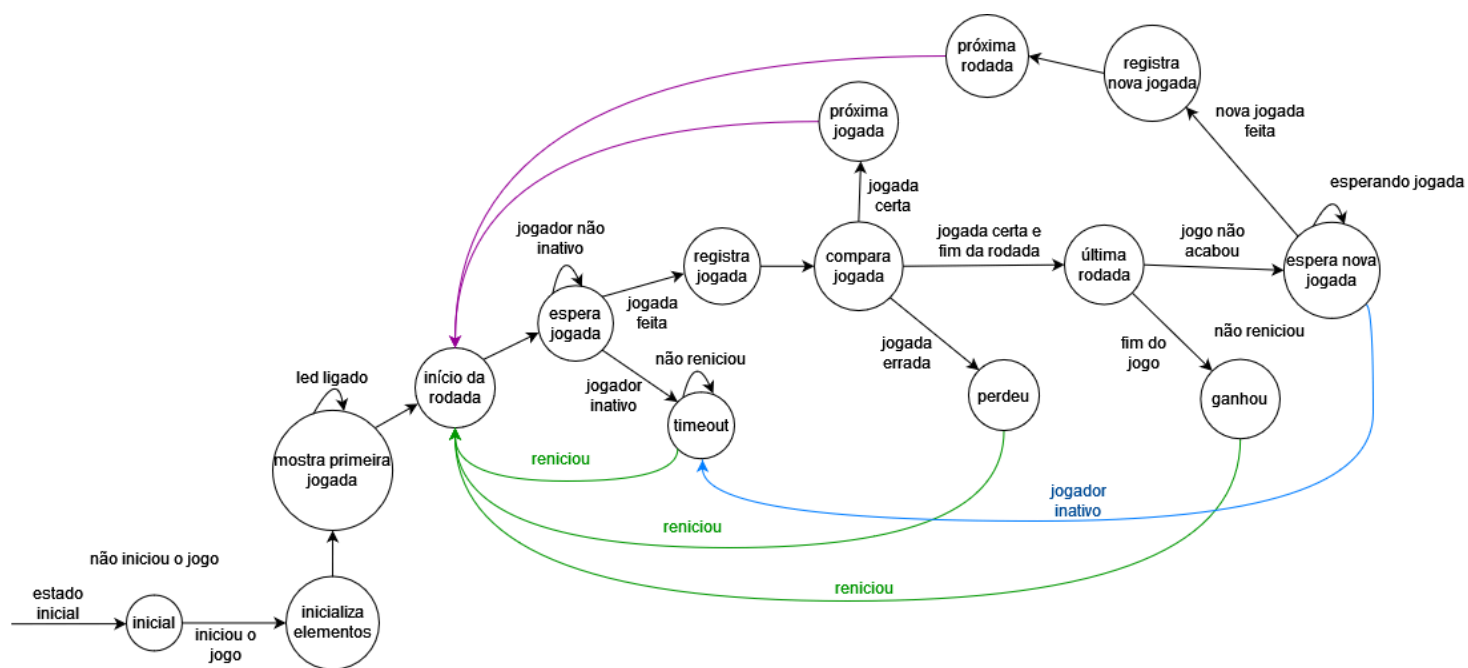
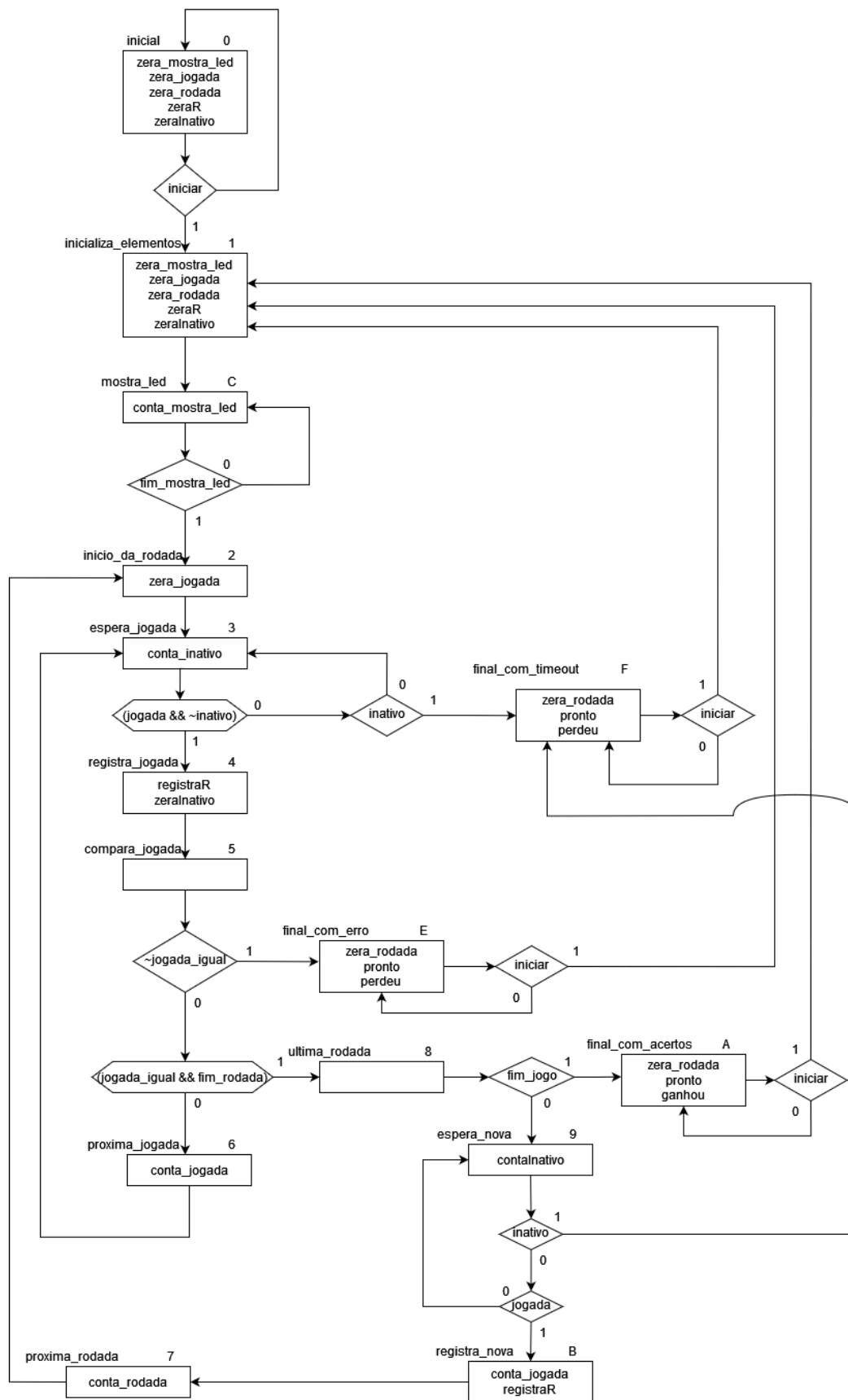


Figura 1 - Diagrama de transição de estados de alto nível

Associado a essa nova configuração de máquina de estados, o diagrama de transição de estados, da figura 2, sintetiza na forma de um fluxograma uma outra visualização mais simples do fluxo lógico do circuito e seu funcionamento.



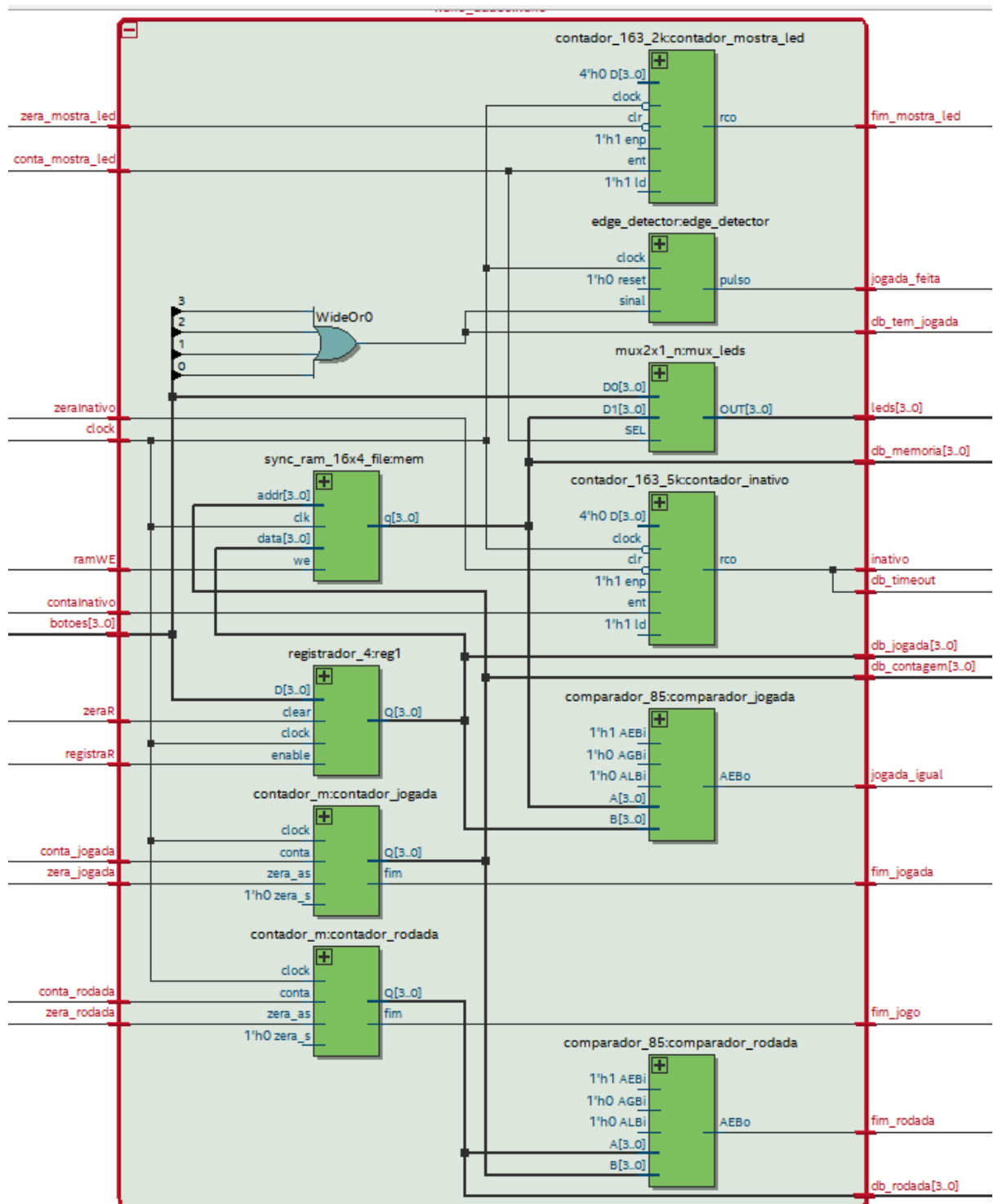


Figura 3 - Diagrama de bloco do fluxo de dados

Verificando-se a consistência da nova proposta com o projeto do circuito, verificou-se atendimento aos requisitos e às novas necessidades. Para verificar então sua funcionalidade e corretude, definiu-se um plano de testes, descrito na seção 2.1.1.

2.1.1 Testes

Para essa implementação, o plano de testes consistiu em quatro cenários de testes, que verificam o comportamento do timeout para jogadas inéditas e para jogadas já registradas na memória RAM (testes 1 e 2), o erro numa jogada de uma certa rodada (teste 3) e o final bem sucedido, com o acerto de todos os valores (teste 4). Os testes 1, 2, 3 e 4 estão descritos em detalhes nas tabelas 1, 2, 3 e 4.

Cenário de Teste 1 - Timeout de Jogada já feita na 1ª Jogada da 2ª rodada					
	Operação	Sinais de controle	Resultado esperado	Resultado observado	Veredito
c.i.	Condições iniciais	reset=0 iniciar=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	
1	Resetar o circuito	reset=1 iniciar=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	
2	Iniciar	reset=0 iniciar=1	ganhou=0 perdeu=0 pronto=0 db_iniciar=0-1 db_igual=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0-1 db_igual=0	

			db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=1-C-2-3	db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=1-C-2-3	
3	Jogar até rodada 2	reset=0 iniciar=0 botões=0-1 0-1-2 (timeout)	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0-1 db_tem_jogada=0 timeout=0 db_jogadafeita=1-2-4 db_rodada=1-2 db_contagem=0-1-2 db_memoria=1-2-4 db_estado=3-4-5-8-9-B-7-2 e 3-4-5-F	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0-1 db_tem_jogada=0 timeout=0 db_jogadafeita=1-2-4 db_rodada=1-2 db_contagem=0-1-2 db_memoria=1-2-4 db_estado=3-4-5-8-9-B-7-2-3 e 3-4-5-F	

Tabela 1 - Cenário de teste 1, com Timeout de jogada já na 1ª jogada da 2ª rodada

Cenário de Teste 2 - Timeout ne 1ª Jogada inédita da 2ª rodada					
	Operação	Sinais de controle	Resultado esperado	Resultado observado	Veredito
c.i.	Condições iniciais	reset=0 iniciar=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	
1	Resetar o circuito	reset=1 iniciar=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0=0	

				db_memoria=1 db_estado=0	
2	Iniciar	reset=0 iniciar=1	ganhou=0 perdeu=0 pronto=0 db_iniciar=0-1 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=1-C-2-3	ganhou=0 perdeu=0 pronto=0 db_iniciar=0-1 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=1-C-2-3	
3	Jogar até rodada 2	reset=0 iniciar=0 botões=0-1 0-1-2 0-1-2 (timeout)	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0-1 db_tem_jogada=0 timeout=0 db_jogadafeita=1-2-4 db_rodada=1-2-3 db_contagem=1-2 db_memoria=1-2-4 db_estado=3-4-5-8-9-B-7-2 E-3-4-5-6-3-4-5-8-9-F	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0-1 db_tem_jogada=0 timeout=0 db_jogadafeita=1-2-4 db_rodada=1-2-3 db_contagem=1-2 db_memoria=1-2-4 db_estado=3-4-5-8-9-B-7-2 E-3-4-5-6-3-4-5-8-9-F	

Tabela 2 - Timeout na primeira jogada inédita da segunda rodada

Cenário de Teste 3 - Erro na 2ª Jogada da 2ª rodada					
	Operação	Sinais de controle	Resultado esperado	Resultado observado	Veredito
c.i.	Condições iniciais	reset=0 iniciar=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0 db_estado=0	
1	Resetar o circuito	reset=1 iniciar=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0	

			db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	
2	Iniciar	reset=0 iniciar=1	ganhou=0 perdeu=0 pronto=0 db_iniciar=0-1 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=1-C-2-3	ganhou=0 perdeu=0 pronto=0 db_iniciar=0-1 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=1-C-2-3	
3	Jogar até rodada 2	reset=0 iniciar=0 botões=0-1 0-1-2 0-4(erro)	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0-1 db_tem_jogada=0 timeout=0 db_jogadafeita=1-2-4 db_rodada=1-2-3 db_contagem=1-2 db_memoria=1-2-4 db_estado=3-4-5-8-9-B-7-2 e 3-4-5-6-3-4-5-E	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0-1 db_tem_jogada=0 timeout=0 db_jogadafeita=1-2-4 db_rodada=1-2-3 db_contagem=1-2 db_memoria=1-2-4 db_estado=3-4-5-8-9-B-7-2 e 3-4-5-6-3-4-5-E	

Tabela 3 - Cenário de Teste 3, com erro na 2ª Jogada da 2ª rodada

Cenário de Teste 4 - Acertar todas as rodadas					
	Operação	Sinais de controle	Resultado esperado	Resultado observado	Veredito
c.i.	Condições iniciais	reset=0 iniciar=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	
1	Resetar o circuito	reset=1 iniciar=0	ganhou=0 perdeu=0	ganhou=0 perdeu=0	

			pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	pronto=0 db_iniciar=0 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=0	
2	Iniciar	reset=0 iniciar=1	ganhou=0 perdeu=0 pronto=0 db_iniciar=0-1 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=1-C-2-3	ganhou=0 perdeu=0 pronto=0 db_iniciar=0-1 db_igual=0 db_tem_jogada=0 timeout=0 db_jogadafeita=0 db_rodada=0 db_contagem=0 db_memoria=1 db_estado=1-C-2-3	
3	Jogar até a rodada E	reset=0 iniciar=0 botões= 0-1 0-1-2 0-1-2-3 0-1-2-3-2 0-1-2-3-2-1 0-1-2-3-2-1-0 0-1-2-3-2-1-0-0 0-1-2-3-2-1-0-0-1 0-1-2-3-2-1-0-0-1-1 0-1-2-3-2-1-0-0-1-1-2 0-1-2-3-2-1-0-0-1-1-2-2 0-1-2-3-2-1-0-0-1-1-2-2-3 0-1-2-3-2-1-0-0-1-1-2-2-3-3 0-1-2-3-2-1-0-0-1-1-2-2-3-3-0 0	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0-1 db_tem_jogada=0-1 timeout=0 db_jogadafeita=1-2-4-8 db_rodada=1-2-3-4-5-6-7-8-9-A-B-C-D-E db_contagem=1-2-3-4-5-6-7-8-9-A-B-C-D-E db_memoria=1-2-4-8 db_estado=3-4-5-8-9-B-7-2 e 3-4-5-6-3-4-5-A	ganhou=0 perdeu=0 pronto=0 db_iniciar=0 db_igual=0-1 db_tem_jogada=0-1 timeout=0 db_jogadafeita=1-2-4-8 db_rodada=1-2-3-4-5-6-7-8-9-A-B-C-D-E db_contagem=1-2-3-4-5-6-7-8-9-A-B-C-D-E db_memoria=1-2-4-8 db_estado=3-4-5-8-9-B-7-2 e 3-4-5-6-3-4-5-A	
4	Jogar última rodada	reset=0 iniciar=0 botões=0-1-2-3-2-1-0-0-1-1-2-2-3-3-1-2	ganhou=1 perdeu=0 pronto=1 db_iniciar=0 db_igual=0-1 db_tem_jogada=0 timeout=0 db_jogadafeita=1-2-4-8 db_rodada=E-F db_contagem=1-2-3-4-5-6-7-8-9-A-B-C-D-E-F db_memoria=1-2-4-8 db_estado=4-5-6-3 E-4-5-8-A	ganhou=1 perdeu=0 pronto=1 db_iniciar=0 db_igual=0-1 db_tem_jogada=0 timeout=0 db_jogadafeita=1-2-4-8 db_rodada=E-F db_contagem=1-2-3-4-5-6-7-8-9-A-B-C-D-E-F db_memoria=1-2-4-8 db_estado=4-5-6-3 E-4-5-8-A	

Tabela 4 - Cenário de teste 4, com acerto de todas as rodadas

Para cada um desses testes, produziu-se um testbench em verilog correspondente, posteriormente simulado no software ModelSim, o que gerou as formas de onda das figuras 4, 5, 6 e 7, relativas aos cenários de testes 1, 2, 3 e 4. As formas de onda evidenciaram o comportamento esperado do circuito e demonstraram sua corretude.

Nas formas de onda dos cenários de testes 1 e 2, das figuras 4 e 5, nessa ordem, apresentam explicitamente o sinal alto de timeout, perdeu e pronto ao final do tempo máximo de espera da nova jogada, tal qual esperado.

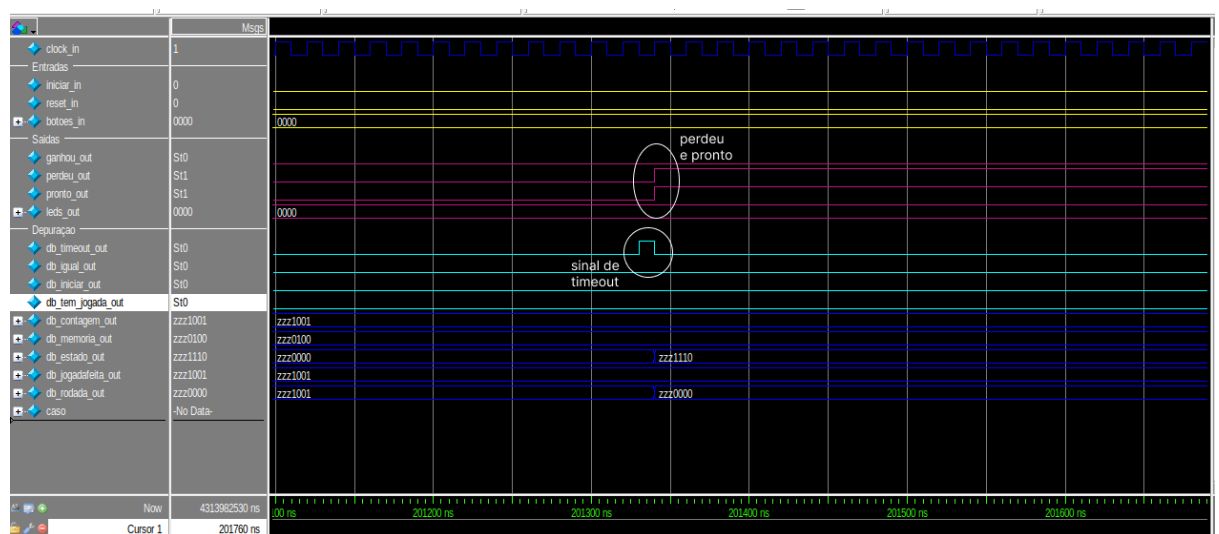


Figura 4 - Waveforms do teste 1, gerados pelo ModelSim

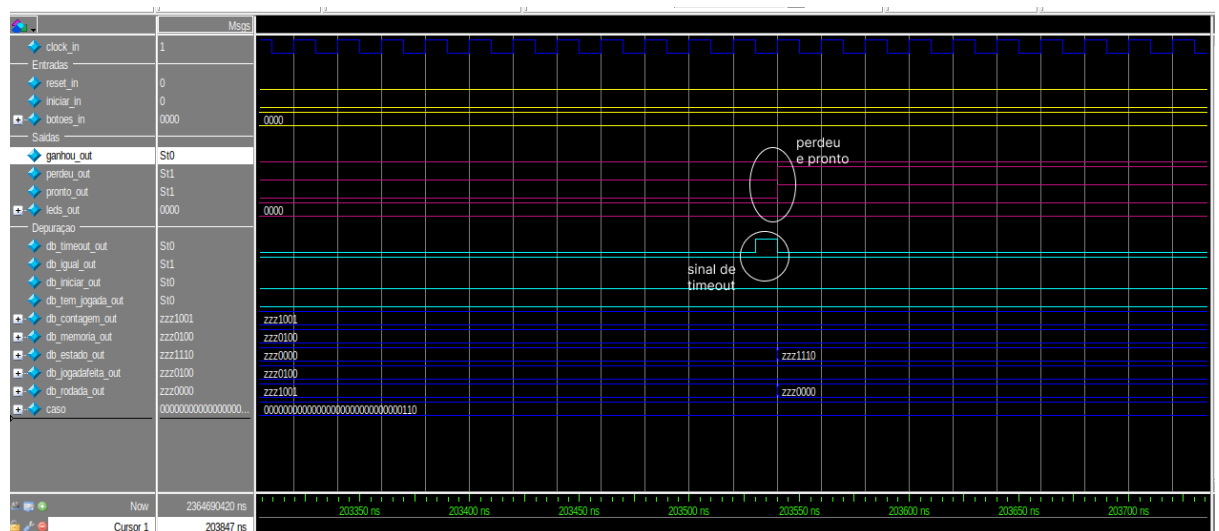


Figura 5 - Waveforms do teste 2, gerados pelo ModelSim

Já para o teste do cenário 3, da figura 6, verificou-se o sinal alto para perdeu e pronto, após a jogada incorreta, e sem o sinal de timeout, já que o erro foi por

conta de um botão apertado incorretamente pelo usuário, não uma demora excessiva na inserção da próxima entrada. Assim, ele comportou-se conforme o esperado.

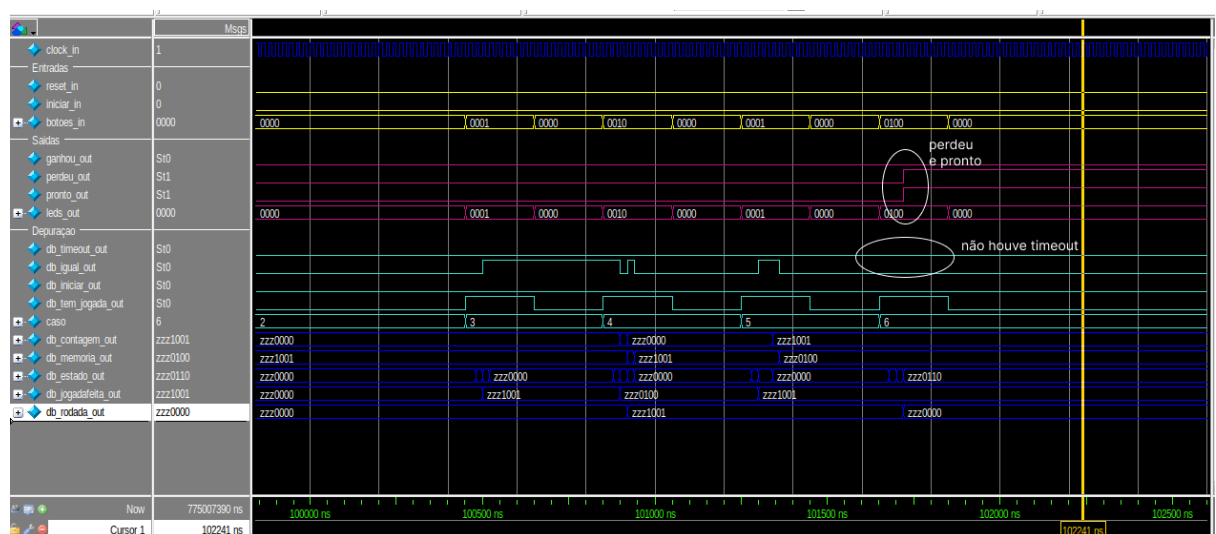


Figura 6 - Waveforms do teste 3, gerados pelo ModelSim

Por fim, o cenário de teste 4, com formas de onda na figura 7, mostrou corretude de todos os sinais para cada uma das rodadas e, na rodada final, apresentou sinais altos para ganhou e pronto, também sem apresentar timeout. Portanto, ele também mostrou corretude dessa funcionalidade.

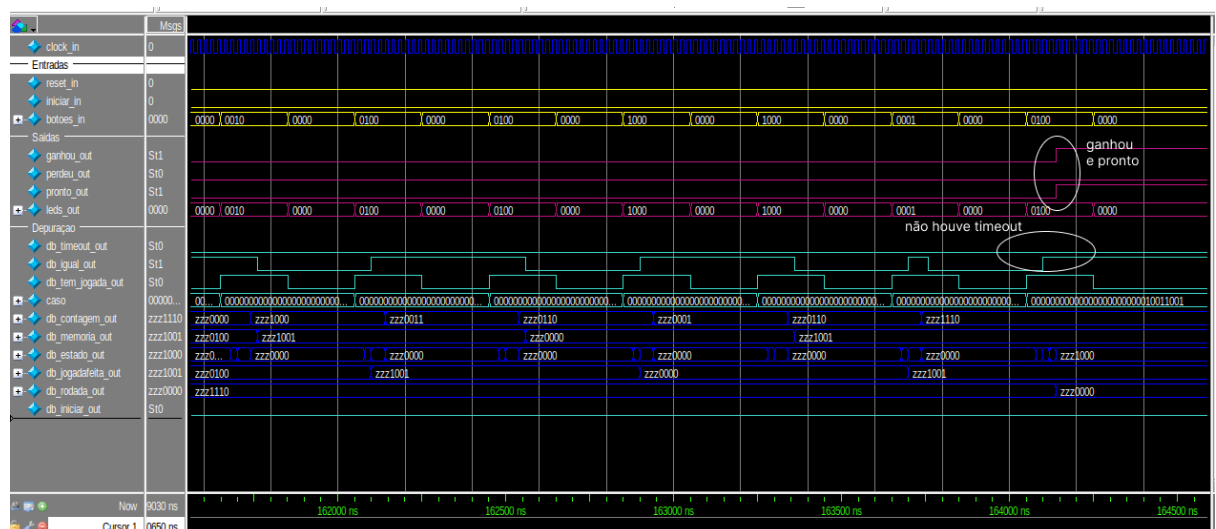


Figura 7 - Waveforms do teste 4, gerados pelo ModelSim

2.2. Atividade 2 – Implementação e Síntese

Para implementação na placa FPGA, definiu-se a associação de pinos e sinais entre a placa FPGA e a Analog Discovery tal qual na tabela 5.

Sinal	Pino na Placa DE0-CV	Pino na FPGA	Analog Discovery
CLOCK	GPIO_0_D0	PIN_N16	StaticIO – LED – DIO0 e Patterns – Clock – 1 kHz
RESET	GPIO_0_D1	PIN_B16	StaticIO – Button 0/1 – DIO1
INICIAR	GPIO_0_D3	PIN_C16	StaticIO – Button 0/1 – DIO2
BOTOES(0)	GPIO_0_D11	PIN_R22	StaticIO – Button 0/1 – DIO4
BOTOES(1)	GPIO_0_D13	PIN_T22	StaticIO – Button 0/1 – DIO5
BOTOES(2)	GPIO_0_D15	PIN_N19	StaticIO – Button 0/1 – DIO6
BOTOES(3)	GPIO_0_D17	PIN_P19	StaticIO – Button 0/1 – DIO7
PERDEU	GPIO_1_D11	PIN_J18	StaticIO - LED - DIO8
GANHOU	GPIO_1_D13	PIN_G11	StaticIO - LED - DIO9
PRONTO	GPIO_1_D15	PIN_J11	StaticIO - LED - DIO10
LEDS(0)	GPIO_1_D17	PIN_A15	StaticIO - LED - DIO12
LEDS(1)	GPIO_1_D19	PIN_L8	StaticIO - LED - DIO13
LEDS(2)	GPIO_1_D21	PIN_B15	StaticIO - LED - DIO14
LEDS(3)	GPIO_1_D23	PIN_E14	StaticIO - LED - DIO15
DB_IGUAL	Led LEDR7	PIN_U1	-
DB_INICIAR	Led LEDR1	PIN_AA1	-
DB_TEM_JOGA DA	Led LEDR2	PIN_W2	-
TIMEOUT	Led LEDR3	PIN_Y3	-

DB_CONTAGEM	Display HEX0	DB_CONTAGEM[0]=PIN_U21 DB_CONTAGEM[1]=PIN_V21 DB_CONTAGEM[2]=PIN_W22 DB_CONTAGEM[3]=PIN_W21 DB_CONTAGEM[4]=PIN_Y22 DB_CONTAGEM[5]=PIN_Y21 DB_CONTAGEM[6]=PIN_AA22	-
DB_MEMORIA	Display HEX1	DB_MEMORIA[0]=PIN_AA20 DB_MEMORIA[1]=PIN_AB20 DB_MEMORIA[2]=PIN_AA19 DB_MEMORIA[3]=PIN_AA18 DB_MEMORIA[4]=PIN_AB18 DB_MEMORIA[5]=PIN_AA17 DB_MEMORIA[6]=PIN_U22	-
DB_JOGADAFEITE	Display HEX2	DB_JOGADAFEITA[0]=PIN_Y19 DB_JOGADAFEITA[1]=PIN_AB17 DB_JOGADAFEITA[2]=PIN_AA10 DB_JOGADAFEITA[3]=PIN_Y14 DB_JOGADAFEITA[4]=PIN_V14 DB_JOGADAFEITA[5]=PIN_AB22 DB_JOGADAFEITA[6]=PIN_AB21	-
DB_RODADA	Display HEX3	DB_RODADA[0]=PIN_Y16 DB_RODADA[1]=PIN_W16 DB_RODADA[2]=PIN_Y17 DB_RODADA[3]=PIN_V16 DB_RODADA[4]=PIN_U17 DB_RODADA[5]=PIN_V18 DB_RODADA[6]=PIN_V19	-
DB_ESTADO	Display HEX5	DB_ESTADO[0]=PIN_N9 DB_ESTADO[1]=PIN_M8 DB_ESTADO[2]=PIN_T14 DB_ESTADO[3]=PIN_P14 DB_ESTADO[4]=PIN_C1 DB_ESTADO[5]=PIN_C2 DB_ESTADO[6]=PIN_W19	-

Tabela 5 - Designação de Pinos

Com a síntese do circuito descrito em verilog, o software Intel Quartus Prime gerou o diagrama de blocos da figura 8, consistente com o projeto feito nas etapas anteriores e com as modificações propostas. Além disso, o Quartus também produziu a visualização da máquina de estados finita, contida na figura 9. Essa configuração de estados é equivalente à projetada para circuito, tal qual mostrado na figura 1.

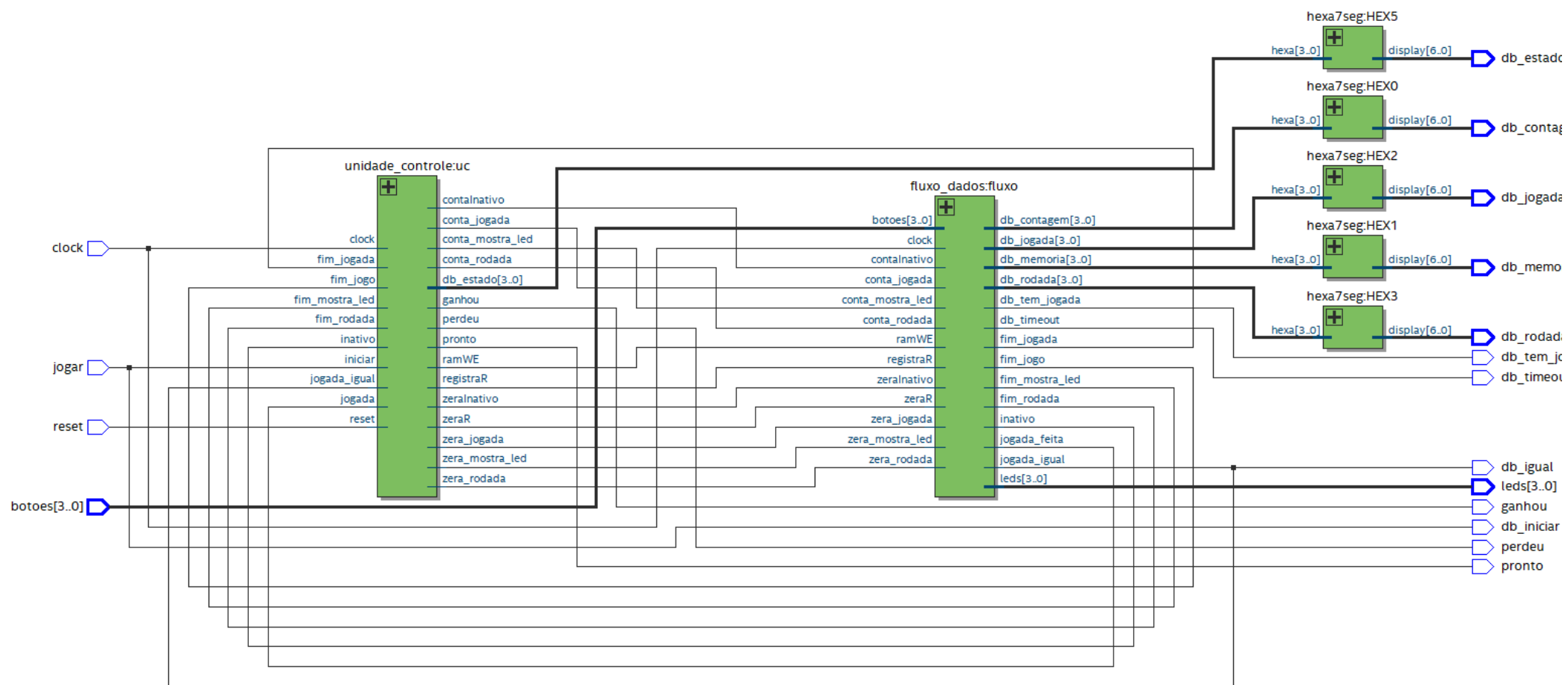


Figura 8 - Diagrama de blocos do circuito

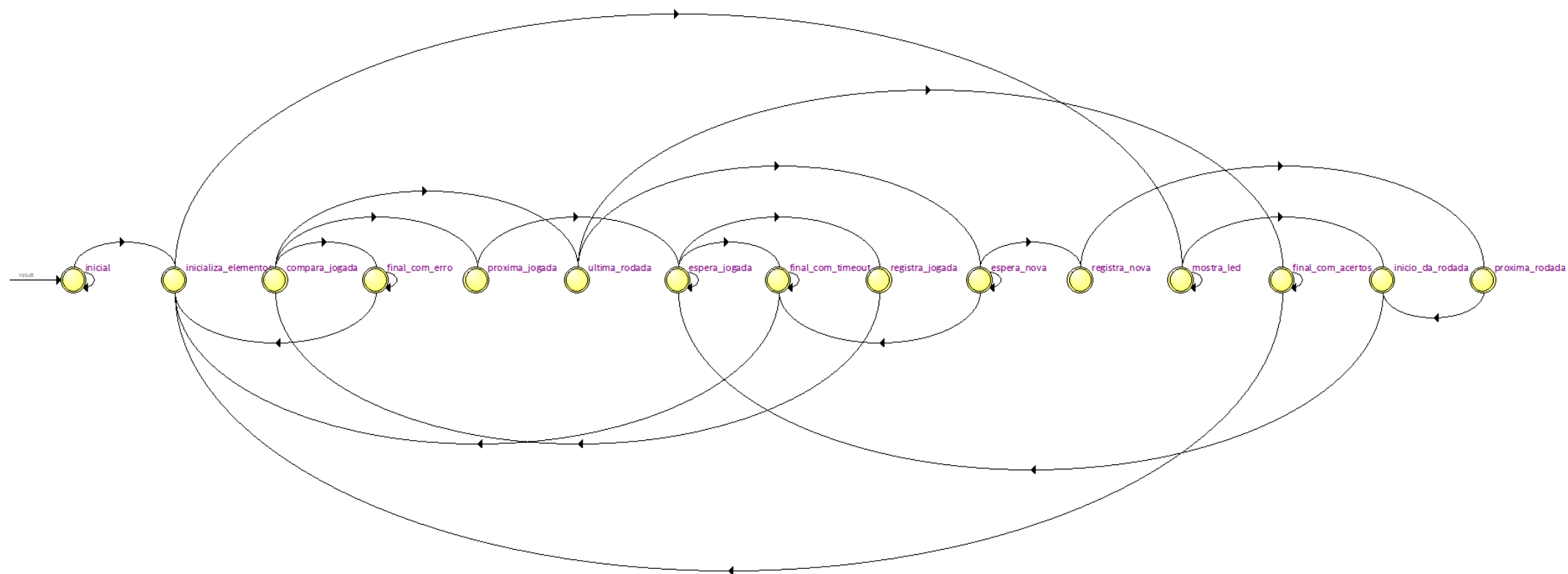


Figura 9 - Diagrama de transição de estados gerada pelo quartus

3. Planejamento da aula prática

Iniciará-se a aula prática baixando os arquivos do projeto, armazenados em repositório remoto do github, para o diretório local. Em seguida, abrirá-se o projeto do circuito no Intel Quartus Prime e compilará-se ele, a fim de verificar se não há problemas da versão mais atual do código, seja em função de sintaxe seja em função de qualquer outra incompatibilidade. Em seguida, conectará-se os pinos da Analog Discovery à placa FPGA, de acordo com o Pin Planner e a figura 10 abaixo. A fim de facilitar a leitura e a interpretação dos dados dos displays e dos leds, trará-se diagramas e imagens impressas, com informações úteis para a montagem.

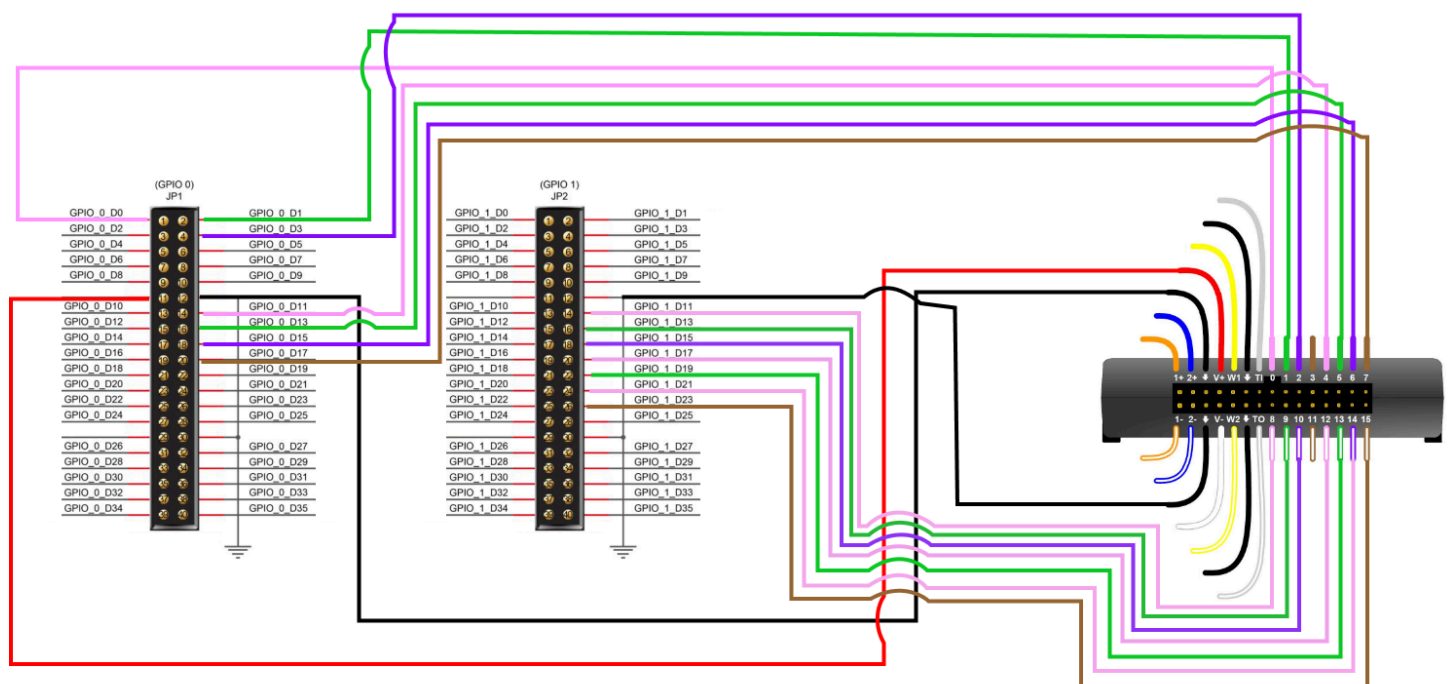


Figura 10 - Ligações dos GPIO's com o Analog Discovery

Com essas etapas concluídas, realizará-se a síntese do circuito no Quartus e, com o sucesso dessa operação, programará-se a placa. Finalmente, iniciará-se então a bateria de testes. Eles serão realizados em sequência, e suas entradas serão dadas de acordo com as descritas pelas tabelas de teste 1, 2 e 3. Em cada teste, deve-se seguir o procedimento padrão de ativar os sinais de entrada em ordem e observar os sinais de depuração, nos displays e no led. Havendo compatibilidade de todos os sinais, marcará-se um verificado na etapa do teste correspondente.

Caso haja divergência, repetirá-se o teste, desde a primeira etapa, para eliminar possíveis erros de operação, como demora excessiva para jogar, que poderia provocar timeout. Com a persistência do erro, deve-se lançar hipóteses sobre as possíveis fontes de erro e revisar o circuito. A revisão deve iniciar-se pelo módulo defeituoso, depois para os módulos que entram diretamente em contato com ele e, em seguida, deve-se revisar o diagrama de blocos e a lógica da máquina de estados. nessa abordagem do mais específico para o mais geral, procura-se o erro em regiões cada vez mais abrangentes, que começam das proximidades do ponto de falha do circuito. Falhando esse processo, deve-se suspeitar de alguma falha física do circuito, como má conexão dos cabos e problemas na placa FPGA ou da Analog Discovery.

Finalizados os testes, procederá-se então ao desafio. Começará-se por uma reflexão sobre uma nova lógica, em relação a que estados e transições devem ser modificados ou incluídos. A partir daí, deve-se validar a nova lógica, verificando a nível de pensamento se ela não prejudica as funcionalidades anteriores e se incorpora as novas. Em seguida, inicia-se uma fase de implementação, com a programação das novas mudanças no verilog. Finalmente, compila-se e programa-se a placa FPGA com o novo circuito e inicia-se a onda de testes.

Testa-se primeiro a retrocompatibilidade com a lógica do circuito anterior onde for cabível, para verificar que a adição traz novas funcionalidades mas não perturba as demais, já feitas. Assim, procede-se então para o teste da nova funcionalidade, proposta pelo desafio. Sendo aprovado, dá-se por encerrada a experiência e completa-se o relatório.

4. Relatório

4.1. Atividade 2: Testes e sinais de depuração

4.2. Atividade 3 – Desafio: Modificação do Sistema Digital

4.2.1. Testes e sinais de depuração