

# Laboratório de Sistemas Digitais

## Projeto nº 9 - Marcador de Basquetebol

### Relatório do Projeto Final

Rui Oliveira 89216 | Pedro Silva 89228

- **Introdução** (breve descrição geral, objetivos e resumo das funcionalidades)

O nosso projeto “Marcador de Basquetebol” não poderia ser mais específico, pelo menos em relação ao nome, ao que tínhamos que fazer. Os objetivos eram implementar um temporizador decrescente em que o tempo começasse nos 10 minutos, apresentar os resultados de ambas as equipas e também as faltas, os descontos de tempo e o período em que se encontra o jogo.

Teríamos também que ter em atenção que quando o tempo fosse superior ou igual a 1 minuto de jogo, o tempo deveria ser apresentado em (minutos:segundos) e quando fosse inferior a 1 minuto deveria apresentar o tempo em (segundos:centésimos).

Em relação aos pontos e faltas das equipas, deveria haver um botão para remover pontos ou faltas, caso houvesse um erro da pessoa que está a anotar o resultado.

Caso conseguíssemos fazer o que nos era pedido anteriormente, teríamos o objetivo de colocar algumas informações no LCD e a produção de sons do evento.

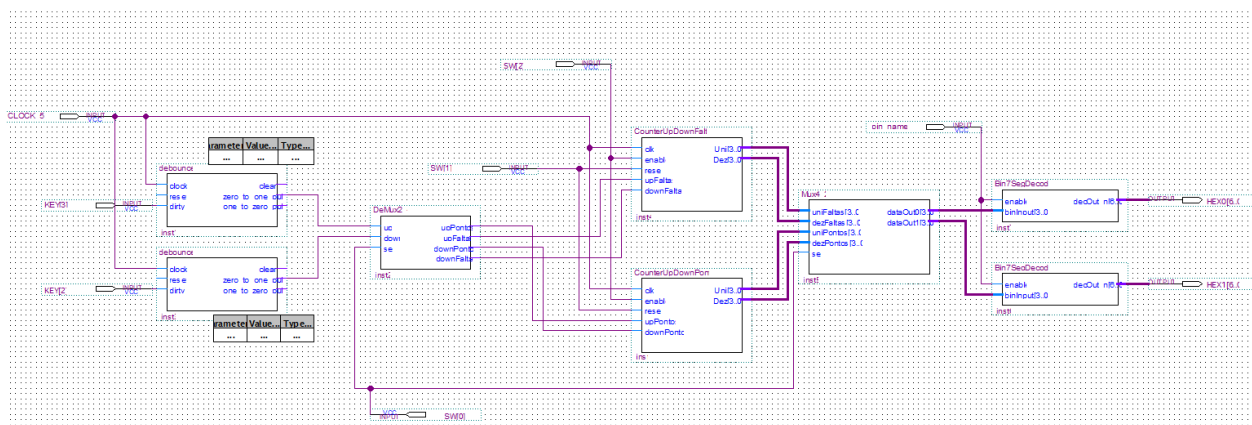
- **Arquitetura** (descrição da estrutura conceptual do sistema com pelo menos uma figura ilustrativa)

#### Marcador

Na elaboração dos contadores de pontos e faltas do nosso projeto definimos inicialmente instanciar um select (SW[1]) partilhado entre um DeMux2\_4 e um Mux6\_2 que pudesse alternar entre a contagem das respetivas variáveis, com o cuidado de só ser possível incrementar (KEY[3] e KEY[1]) ou decrementar (KEY[2] e KEY[0]) se deste modo as tivéssemos selecionado.

Assim nos displays HEX4 HEX5 para a equipa A e HEX6 HEX7 para a equipa B poderíamos ver os pontos e as faltas de ambas equipas.

Na figura seguinte está representado o modelo **só para uma equipa** dos dois contadores e dos restantes elementos. Nota: Neste momento do projeto ainda não existiam nem a unidade de controlo (Máquina de Estados) nem debouncers para todos os switches.

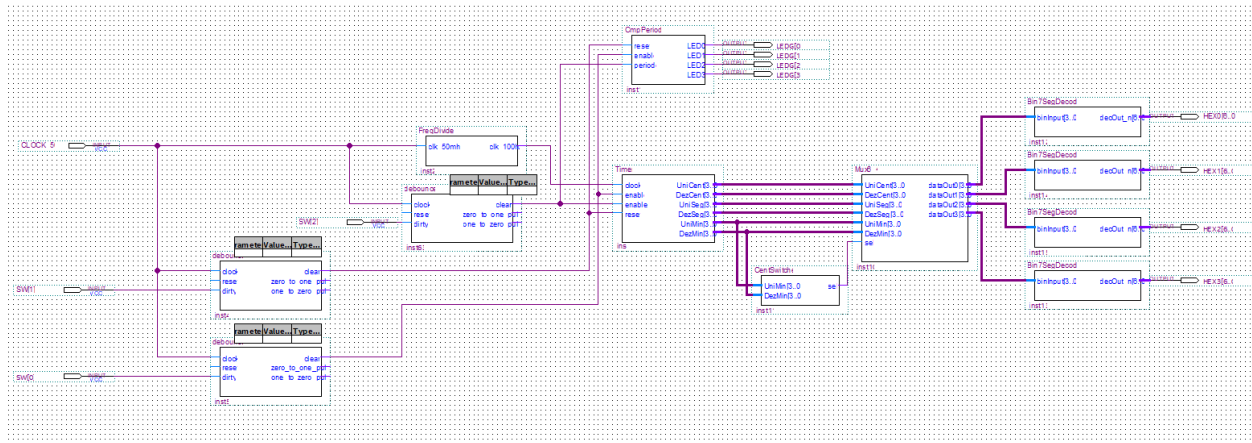


Timer

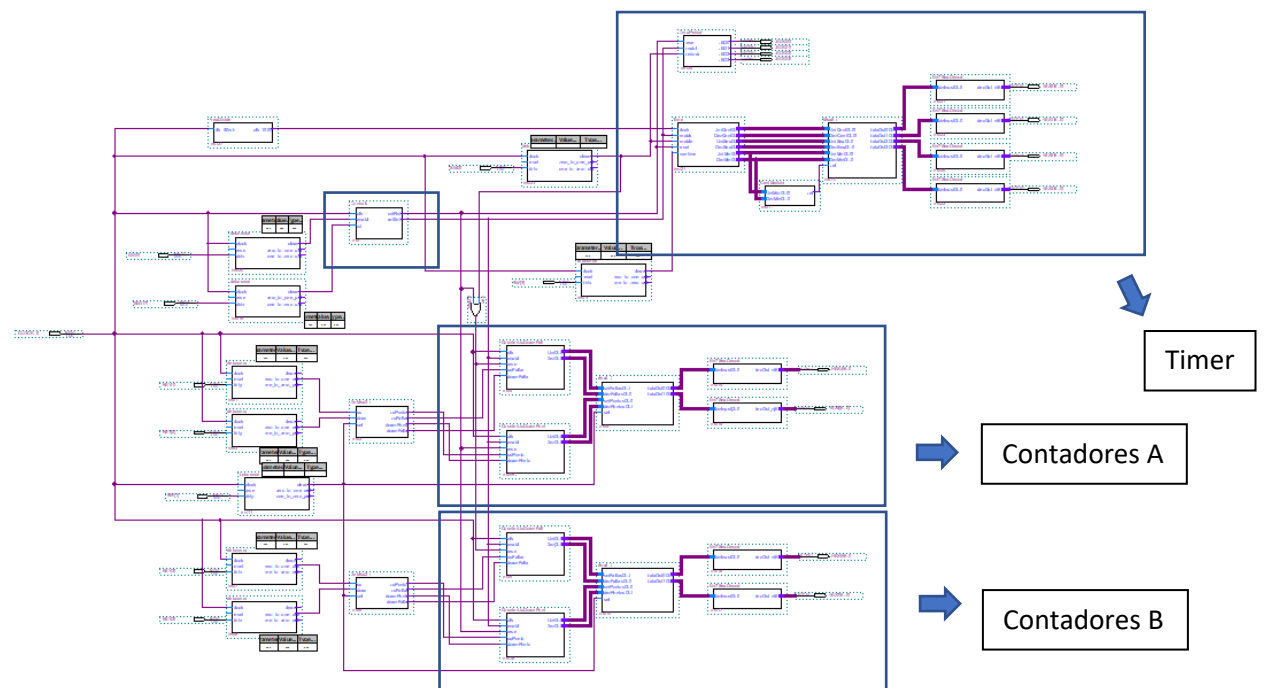
Tal como nos contadores acima descritos, para a elaboração do Timer foi necessário gerir a utilização dos displays disponíveis. Assim era apresentado em 4 dígitos o tempo remanescente em minutos e segundos até o tempo ser inferior a 1 minuto. Neste instante nos 4 displays é apresentado os segundos os segundos e os centésimos.

Desta maneira foi necessário criar um divisor de frequências para 100Hz.

Em relação aos períodos de jogo, cada vez que um período acaba e ligamos o SW[3] é incrementado nos LEDG um período. Nota: Mais tarde foi também instanciado o OverTime.

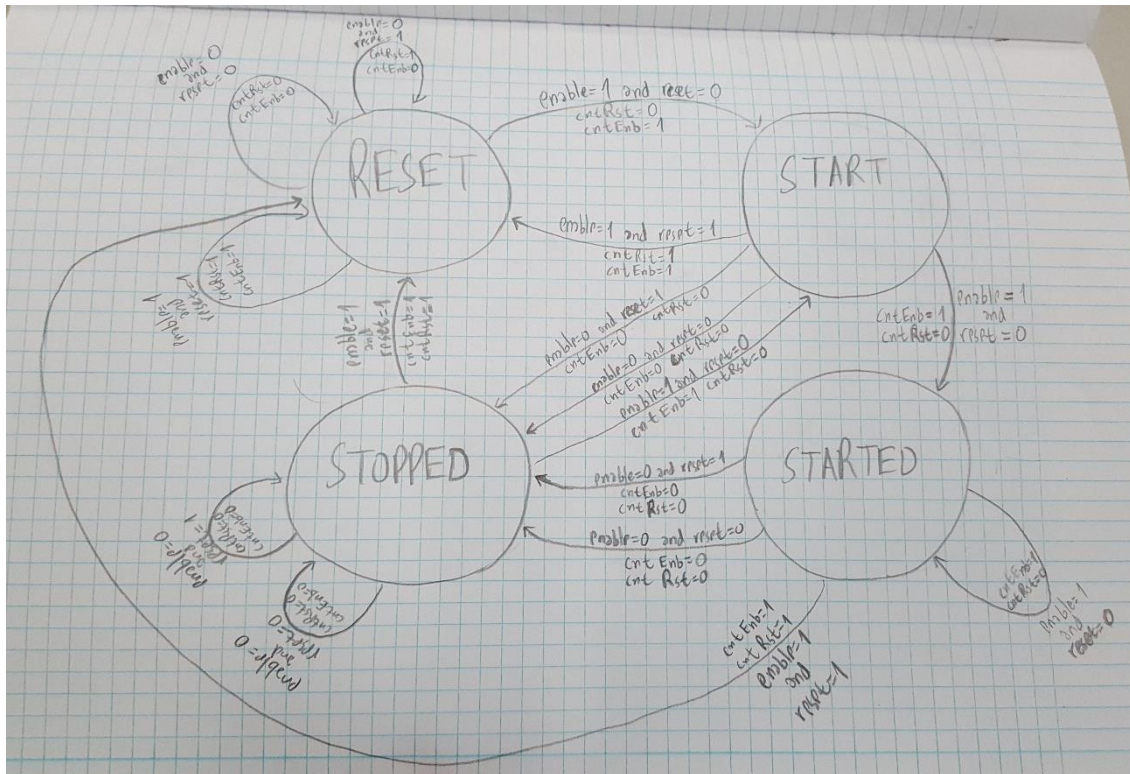


## Arquitetura Geral do Projeto



- **Implementação** (incluindo representação gráfica das máquinas de estado finitos implementadas – se aplicável, aspetos de implementação mais relevantes e ligação a periféricos do kit)

A nossa Unidade de Controlo possui 4 estados. Estes são Reset, Start, Started, Stopped. Consiste numa máquina de Mealy e tem controlo total sobre todos os elementos do projeto não incluindo o Reset das faltas no término dos períodos onde partilha uma porta logica OR.



#### Manual do utilizador:

O nosso projeto será regulado por vários SW's e KEY's.

O SW[0] será o Enable principal. Se este Switch estiver desligado, o tempo estará sempre parado e não será possível incrementar períodos, pontos e faltas.

O SW[1] será o Switch que vai alternar entre pontos e faltas. Caso esteja desligado, irão ser mostrados os pontos de ambas as equipas nos HEX[4], HEX[5], HEX[6] e HEX[7] e apenas será possível incrementar pontos nas equipas. Caso esteja ligado, irão ser mostradas os faltas de ambas as equipas nos HEX[4], HEX[5], HEX[6] e HEX[7] e apenas será possível incrementar faltas nas equipas.

O SW[2] será utilizado apenas quando o tempo for 00:00. Serve para repor o tempo a 10 minutos e ao mesmo tempo incrementar um período nas LEDG's e também fazer reset nas faltas.

O SW[3] será o Start/Pause mas apenas quando o tempo for diferente de 00:00, quando estiver a 0 será Start, caso esteja a 1 será Pause; caso o tempo seja 00:00 e o jogo estiver no fim do 4º período empatado, ao ligar o Switch o tempo será reposto a 5 minutos para jogar o Overtime e decidir quem é o vencedor. Caso volte a ficar empatado volta-se a repetir o processo.

SW[0] →	ENABLE	KEY[0] →	RETIRAR PONTOS / FALTAS	EQUIPA	B
SW[1] →	PONTOS / FALTAS	KEY[1] →	ADICIONAR	"	B
SW[2] →	NOVO PERÍODO	KEY[2] →	RETIRAR	"	A
SW[3] →	OVERTIME E START / PAUSE	KEY[3] →	ADICIONAR	"	A
SW[5] →	DESCONTO	TEMPO	EQUIPA	B	
SW[6] →	"	"	"	B	
SW[7] →	"	"	"	B	
SW[8] →	"	"	"	A	
SW[9] →	"	"	"	A	
SW[10] →	"	"	"	A	
SW[17] →	RESET	HEX0 →	UNIDADES	SEGUNDOS / CENTÉSIMOS	
LEDG[0-3] →	NÚMERO DO PERÍODO	HEX1 →	DEZENAS	"	"
LEDR[0-2] →	DESCONTOS DE TEMPO	EQUIPA	B		
LEDR[15-17] →	DESCONTOS DE TEMPO	EQUIPA	A		
		HEX2 →	UNIDADES	MINUTOS / SEGUNDOS	
		HEX3 →	DEZENAS	"	"
		HEX4 →	UNIDADES	PONTOS / FALTAS	EQUIPA B
		HEX5 →	DEZENAS	"	"
		HEX6 →	UNIDADES	"	"
		HEX7 →	DEZENAS	"	"

O SW[5], SW[6] e SW[7] estão ligados às LEDR[0], LEDR[1] e LEDR[2] respetivamente.

Representam os descontos de tempo da equipa B, cada Switch acende uma LEDR.

O SW[8], SW[9] e SW[10] estão ligados às LEDR[17], LEDR[16] e LEDR[15] respetivamente.

Representam os descontos de tempo da equipa A, cada Switch acende uma LEDR.

O SW[17] representa o Reset. Como o Enable tem prioridade no processamento da FPGA, para o Reset funcionar necessitam de estar o SW[17] e o SW[0] ambos ligados. Ao efetuar o Reset, tempo, pontos, faltas e períodos voltam ao estado inicial.

O KEY[3] adiciona pontos ou faltas, dependendo do SW[1], à equipa A. Cada clique adiciona 1 ponto ou falta.

O KEY[2] retira pontos ou faltas, dependendo do SW[1], à equipa A. Cada clique retira 1 ponto ou falta.

O KEY[1] adiciona pontos ou faltas, dependendo do SW[1], à equipa B. Cada clique adiciona 1 ponto ou falta.

O KEY[0] retira pontos ou faltas, dependendo do SW[1], à equipa B. Cada clique retira 1 ponto ou falta.

As LEDG[0], LEDG[1], LEDG[2] e LEDG[3] representam o período de jogo. A LEDG[0] estará sempre ligada já que o jogo tem que estar pelo menos no 1º período. Cada vez que o jogo for reposto nos 10 minutos, uma nova LEDG irá acender-se.

O HEX0 representará as unidades dos segundos enquanto o tempo for superior ou igual a 1 minuto, quando o tempo for inferior a 1 minuto, irá apresentar as unidades dos centésimos de segundo.

O HEX1 representará as dezenas dos segundos enquanto o tempo for superior ou igual a 1 minuto, quando o tempo for inferior a 1 minuto, irá apresentar as dezenas dos centésimos de segundo.



O HEX2 representará as unidades dos minutos enquanto o tempo for superior ou igual a 1 minuto, quando o tempo for inferior a 1 minuto, irá apresentar as unidades dos segundos.

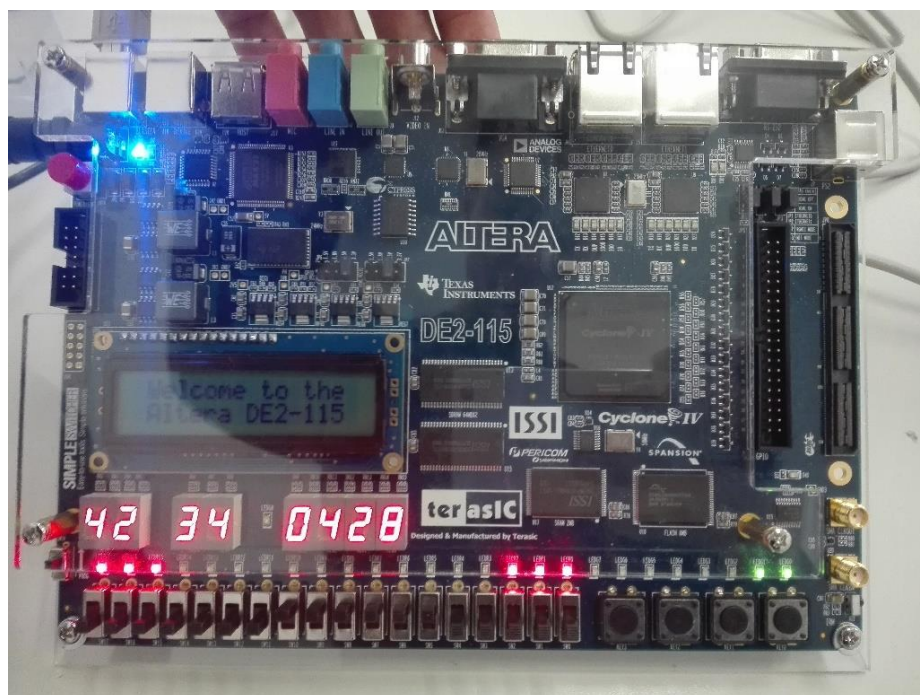
O HEX3 representará as dezenas dos minutos enquanto o tempo for superior ou igual a 1 minuto, quando o tempo for inferior a 1 minuto, irá apresentar as dezenas dos segundos.

O HEX4 representa as unidades dos pontos/faltas da equipa B.

O HEX5 representa as dezenas dos pontos/faltas da equipa B.

O HEX6 representa as unidades dos pontos/faltas da equipa A.

O HEX7 representa as dezenas dos pontos/faltas da equipa A.



- **Conclusão** (discussão da forma como o trabalho realizado foi ao encontro dos objetivos definidos)

Com a elaboração deste projeto podemos desenvolver o conhecimento em linguagem VHDL também como a utilização do Quartus como ferramenta principal de simulação.

Atribuímos uma percentagem de 50% para cada um dos elementos do nosso grupo e a nossa autoavaliação do projeto é de 15 valores.