

**Universidade de Aveiro**  
Dep. de Electrónica, Telecomunicações e Informática  
**Laboratório de Sistemas Digitais**  
**Proposta de Projeto Final**  
Ano letivo 2018/19

**Projeto nº 10 – Relógio Despertador Programável**

## **1. Introdução**

O objetivo deste trabalho é modelar em VHDL e testar na FPGA um relógio despertador programável. O relógio deve suportar além do modo normal de contagem de tempo, a programação de alarmes e configuração de horas e dia da semana.

Para a interação com o utilizador e para o controlo do relógio despertador devem ser consideradas as seguintes entradas e saídas:

### **Entradas:**

Modos de configuração (SW(0) e SW(1))

“00” – modo normal de contagem

“01” – modo de acerto de horas/dias da semana.

“11” – modo de programação da função despertador.

Programação e acerto

Up – KEY(0)

Down – KEY(1)

Seleção de dígitos (horas/minutos/dias) e stop – KEY(2)

Seleção de alarmes – KEY(3)

### **Saídas:**

HEX(0) – minutos, unidades

HEX(1) – minutos, dezenas

HEX(2) – horas, unidades

HEX(3) – horas, dezenas

HEX(4) – dia da semana

KEX(6) – número do alarme

HEX(7) – Modo de funcionamento

LEDG(8) – indicação de segundos

## **2. Descrição do funcionamento**

### **2.1. Funcionamento em modo de contagem**

- Quando se ativa o modo de contagem, o relógio executa a sua função normal de contagem de minutos/horas/dia da semana. Mostrando a passagem do tempo nos displays hexadecimais HEX(0) a HEX(3), o dia no display HEX(4) e a passagem dos segundos no LEDG(8), que deverá piscar à frequência de 1 Hz.
- No modo de contagem, o sistema deve verificar a ocorrência de alarmes previamente programados. Na eventualidade da contagem coincidir (minutos/horas/dia) com algum dos alarmes programados. Os displays HEX(0) and HEX(3) devem piscar à frequência de 1Hz durante um período de 30s ou até que o “stop” seja pressionado (KEY(2)).

### **2.2. Funcionamento em modo de acerto**

- Quando se ativa o modo de acerto, o relógio suspende a contagem normal e permite a interação do utilizador para definir as horas, minutos e dia da semana.
- A programação segue de forma sequencial dos minutos para o dia, permitindo ao utilizador, através da utilização dos botões de pressão up/down (KEY(0)/KEY(1)), acertar os valores pretendidos.
- O botão de pressão KEY(2) tem a função de mudar o dígito a configurar, seguindo a ordem sequencial estabelecida.

- No modo de acerto, os contadores são atuados com uma frequência de 4Hz, desde que os botões de pressão up/down estejam selecionados.

### 2.3. Funcionamento em modo de programação de alarmes

- Quando se ativa o modo de programação de alarmes, o relógio suspende a contagem normal e permite a interação do utilizador para definir as horas, minutos e dia da semana a que o(s) despertador deve atuar.
- A programação segue de forma sequencial como descrito no ponto 2.2, com a diferença da seleção do alarme. O botão de pressão KEY(3) é utilizado para este fim com dupla funcionalidade. Quando atuado por um período inferior a 1s, permite a seleção/programação de múltiplos alarmes. Se pressionado por um período superior a 1s abandona o modo de programação de alarmes.
- Um exemplo desta operação entrando no modo de programação, inicia-se a definição do alarme nº 0, no final o botão de pressão KEY(3) pressionado por um período curto (menor de 1s), permite a definição do alarme nº 1 e sucessivamente até ao máximo de 7 alarmes. Se KEY(3) for pressionado por um período longo, o processo de programação é terminado.

## 3. Implementação

Na implementação deste trabalho sugere-se um desenvolvimento faseado, de acordo com a descrição que se segue. É requisito obrigatório a utilização do sinal de relógio de 50 MHz (CLOCK\_50) para todos os componentes síncronos. É um fator de valorização do projeto a utilização de debouncers associados à utilização dos botões de pressão e de memórias RAM para armazenamento dos alarmes.

**Fase 1:** nesta fase deve ser implementada a funcionalidade do relógio, sem a função alarme, sem a função de acerto de horas e sem a função de programação de alarmes.

**Fase 2:** nesta fase deve ser implementada a função de acerto do relógio. Para tal sugere-se a utilização de máquinas de estado comunicantes.

**Fase 3:** nesta fase deve ser implementada a função de programação de alarmes. De notar que nesta fase, será necessário acrescentar um estado à máquina de estados geral (controlo de modo) e desenvolvida uma nova máquina de estados para a função de programação de alarmes.

## 3. Requisitos para obtenção de classificações superiores a 17 valores

Notas superiores a 17 valores requerem a implementação de duas funcionalidades adicionais.

- Na programação dos alarmes contemplar a possibilidade de permitir ou não a ocorrência de um alarme. Assim, apesar de programadas hora e dia, será necessário a definição de uma variável que indique se o alarme está ativo ou não. Na versão anterior, assumem-se todos os alarmes ativos por defeito.
- Implementar a função de “snooze”, ou seja, quando o despertador é ativado pelo relógio, desde que seja premido o botão de “snooze”, a sinalização de alarme é suspensa por períodos de 10 minutos até que seja premida a função de “stop”. O botão de pressão KEY(3) pode ser usado para esta função.

**Nota:** O *top-level* do sistema deve ser implementado, preferencialmente, com recurso a representação estrutural em VHDL.