

Design de computadores

Relógio Digital

André Toyama e Gabriel Rios

Introdução:

Este projeto se trata da criação de um relógio digital em VHDL para a placa FPGA DE2-115. As funções atribuídas para o relógio são a contagem dos segundos em tempo real, se baseando no clock interno da placa, um modo de contagem acelerada e um modo setup onde é possível definir qualquer horário escolhido.

Como funciona:

O relógio funciona fazendo suas operações através de uma unidade lógica aritmética (ULA), onde a atualização do horário ocorre por meio de uma soma. Assim, é somado “1” a cada um segundo na unidade do segundo, e quando necessário, adiciona-se “1” na dezena do segundo e assim por diante.

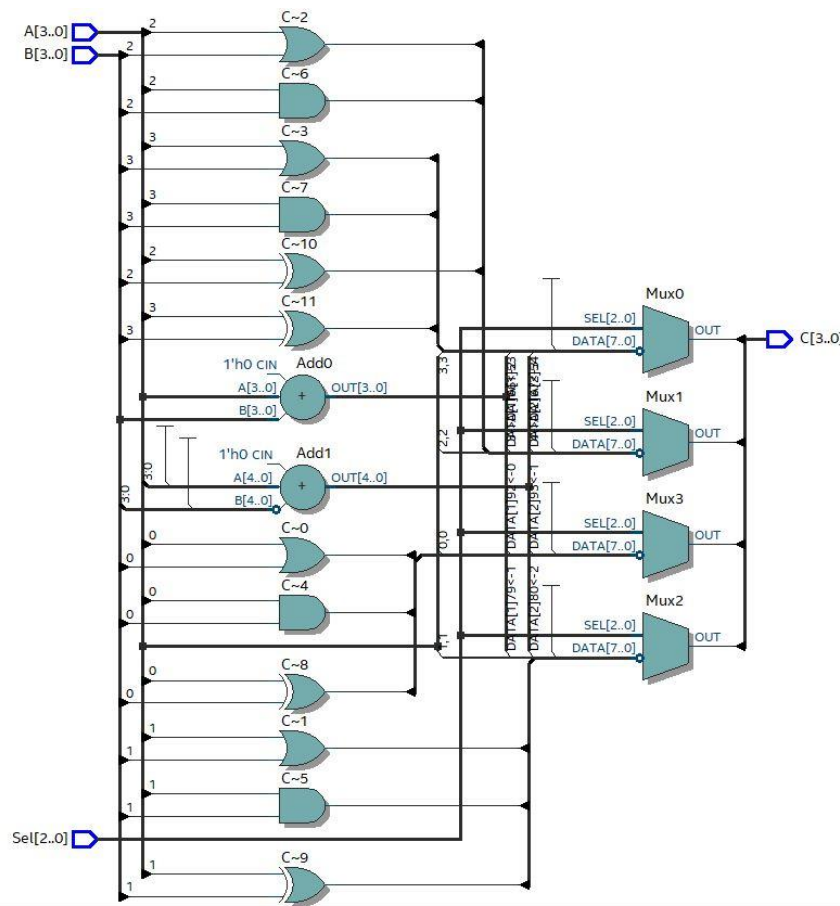


Figura 1 ULA

Além da ULA, o projeto possui registradores para armazenar e sincronizar os valores que serão mostrados nos displays da placa:

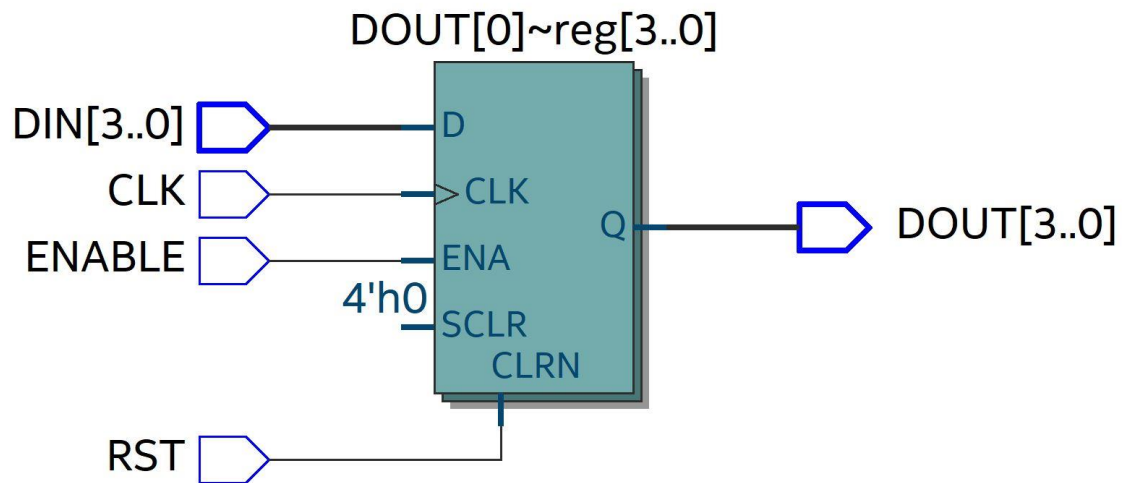


Figura 2 Registrador genérico

Os registradores possuem um “enable” para que o registrador possa atualizar o seu valor de saída baseado no “clock”. Para que o registrador correto seja selecionado, já com seu “enable” ativado, é necessário um “MUX”:

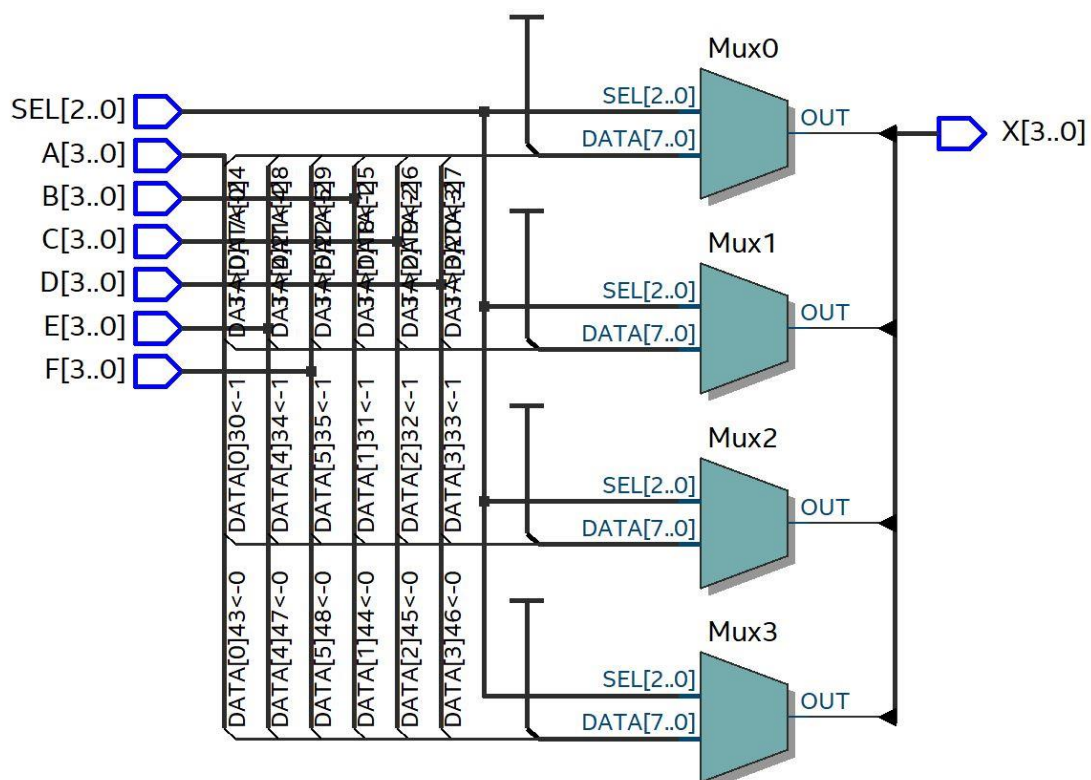


Figura 3 Mux de registradores

A partir do seletor, é selecionado o registradores cujo valor de tempo será atualizado.

Um relógio deve atualizar seu horário a cada 1 segundo. Porém, a lógica do circuito irá ser processada na velocidade da FPGA em questão, que em termos de clock é igual a 50MHz. Assim, é necessário dividir a parte lógica dos processos em 2:

- CLOCK_50, clock da FPGA
- auxClock, clock de 1 segundo

A imagem abaixo mostra a divisão destes clocks, em VHDL:

```
process(CLOCK_50, auxClock)
begin
    if(CLOCK_50'event and CLOCK_50='1') then
        count <= count+1;

        if(count = compare_clk) then
            auxClock <= not auxClock;
            count <= 1;
            if(auxClock='0') then
                auxUlaB := "0001";
            end if;
        else
            auxUlaB := "0000";
        end if;
    end if;
end process;
```

Figura 4 Divisão de clocks na lógica

O funcionamento do relógio como um todo esta baseado em uma máquina de estados (SMUC), onde temos basicamente um estado para cada medida de tempo (unidade dos segundos, dezena dos segundos, unidade dos minutos, etc) e estados de transição de um para o outro, sem contar alguns outros estados necessários. A máquina de estados roda sua lógica com o clock de 50Hz da placa, porém a parte que realiza a atualização do horário só é chamada em pulsos de 1 segundo. Basicamente quando o nosso clock auxiliar, que tem como base de tempo 1 segundo, está em borda de subida, a máquina de estados executa a lógica do relógio, de acrescentar 1 segundo e checar quais atualizações são necessárias baseado nesse segundo. Por exemplo ao atingirmos 10 segundos devemos incrementar a dezena dos segundos e assim por diante.

A máquina de estados age como unidade de controle, assim temos nosso projeto bem dividido em Unidade de Controle, Fluxo de Dados e Top-Level Entity, sem contar alguns componentes extras. Também temos uma segunda maquina de estados mais simples (SM), que é responsável pela troca de modos do relógio.

Modos:

A transição de modos do relógio se dá por uma máquina de estados, onde cada estado é um modo e cada estado de transição se dá por um respectivo botão.

- Setup:

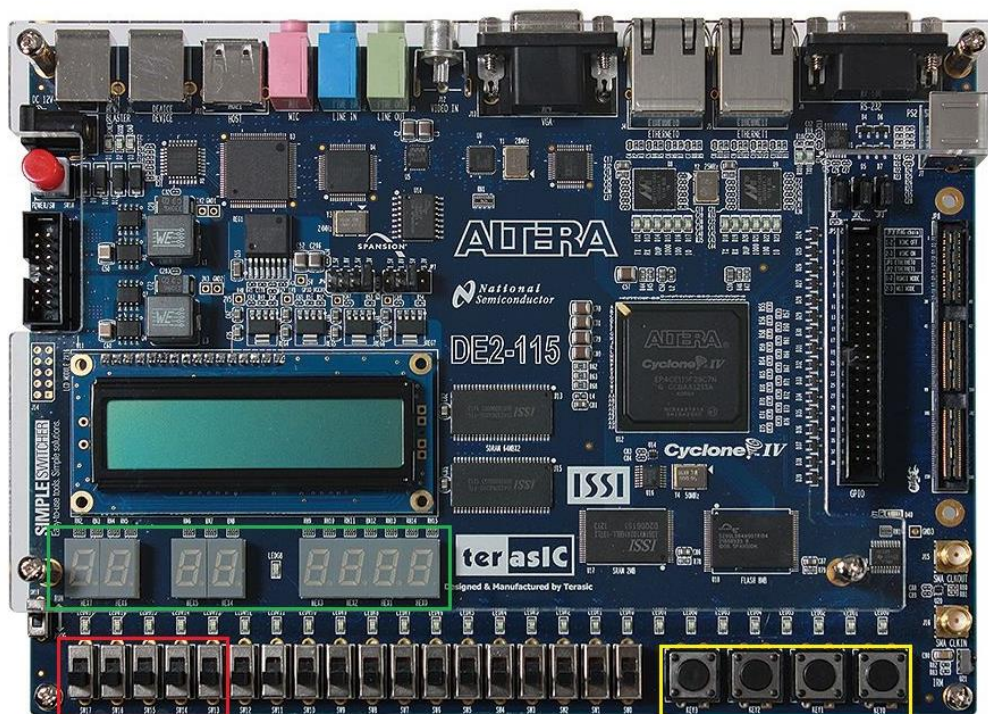
Modo para realizar o ajuste de horário do relógio. O tempo é parado para que haja o ajuste, para fazer o tempo continuar basta voltar para o modo “normal”.

- Acelerado:

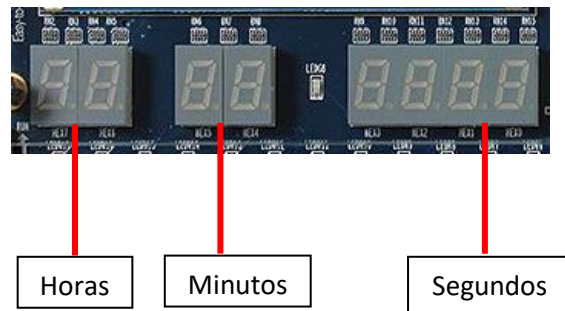
Funciona como o modo normal, porém o tempo é acelerado em 96x, para que a passagem das 24 horas ocorra em 15 minutos.

Como Utilizar:

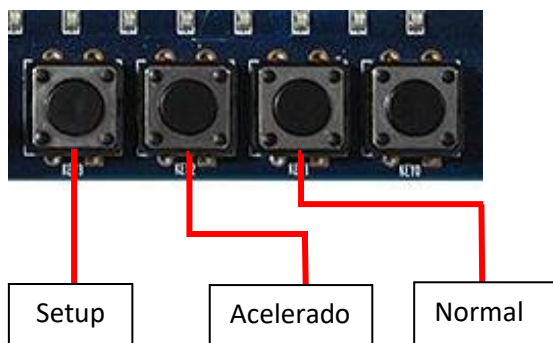
Na imagem da placa abaixo temos destacado as partes da placa necessárias para a utilização do relógio.



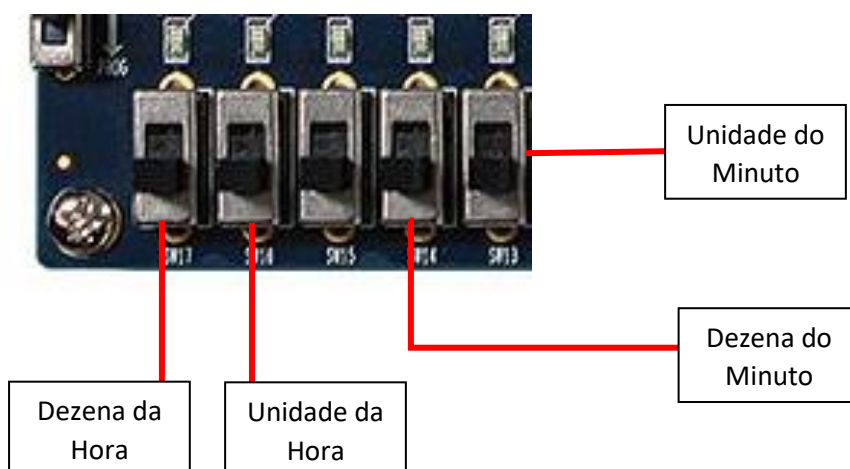
Em verde, temos destacado o display da placa, onde veremos a projeção dos valores de tempo.



Em amarelo temos os botões de transição de modo, onde podemos trocar o modo do relógio.



Em vermelho, temos as chaves utilizadas para o controle do horário no modo setup. Elas só serão utilizadas no modo setup, caso contrário não vão fazer nada. Estando no modo setup, basta ativar a chave do valor em que você quer mudar e apertar novamente o botão de setup, assim acrescentando 1 no valor escolhido.



Complicações do Projeto:

Embora tenhamos uma versão que aparenta ser bem mais funcional do que nossa versão final, ela está com a arquitetura do projeto errada, assim foi necessário a reformulação inteira do projeto em um curto período de tempo. Embora tenha ocorrido algumas perdas de funcionalidades e aparição de bugs, a arquitetura se encontra de forma correta, assim resultando em um aprendizado melhor.