

# Relatório

Tópico: Aula 6 - Timers and Real-time Clock

Grupo: Adhemar Molon Neto

14687681

Alessandro Rodrigues Pereira da Silva

15436838

Link do Github: <>

## Part 4

### Descrição

Implementação de um circuito que recebe como entrada um número de 3 bits, representando uma letra entre A e H e, ao apertar um botão, um Led pisca a letra inserida, em código Morse. Também vem com um botão de Reset que o faz código Morse parar de piscar.

### Código VHDL

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

use IEEE.NUMERIC_STD.ALL;

ENTITY aula6part4 is
    port(
        Num                                : in std_logic_vector(2
downto 0);
        Enable, Reset, Clk                : in std_logic;
        led                                : out std_logic
    );
END aula6part4;
ARCHITECTURE main of aula6part4 is
    signal Tempol : std_logic_vector(26 downto 0);
    signal aceso : std_logic := '0';
    signal terminado : std_logic := '0';
    signal ativado : std_logic := '0';
    signal limite : INTEGER := 0;
    signal indice : INTEGER := 0;
    signal numero : INTEGER;

    -- "01011111010111100001000000" = 25000000
    type ARRAY_LIMITES is array (0 to 1) of INTEGER;
    signal limites : ARRAY_LIMITES := (0 => 25000000, 1 => 25000000);
    type ARRAY_AUX is array (0 to 4) of std_logic_vector(0 to 1);
    type ARRAY_PALAVRA is array (0 to 7) of ARRAY_AUX;
    constant alfabeto : ARRAY_PALAVRA := (
        ("10", "11", "00", "00", "00"), -- 0 1 - - - == .- = A
        ("11", "10", "10", "10", "00"), -- 1 0 0 0 - == -... = B
```

```

("11", "10", "11", "10", "00"), -- 1 0 1 0 - == -. = C
("11", "10", "10", "00", "00"), -- 1 0 0 - - == -.. = D
("10", "00", "00", "00", "00"), -- 0 - - - - == . = E
("10", "10", "11", "10", "00"), -- 0 0 1 0 - == ..- = F
("11", "11", "10", "00", "00"), -- 1 1 0 - - == --. = G
("10", "10", "10", "10", "00") -- 0 0 0 0 - == .... = H
);
begin

    process (Clk) is
    begin
        if(rising_edge(Clk)) then
            if(Enable = '0') then
                numero <= to_integer(unsigned(Num)); -- só serve
para botões, não alavancas
                ativado <= '1';
                terminado <= '0'; -- só botões
            end if;
            if(Reset = '0') then
                terminado <= '0';
                ativado <= '0';
                aceso <= '0';
                limite <= 0;
                indice <= 0;
            end if;
            if(ativado = '0') then
                Tempol <= (others => '0');
            elsif (ativado = '1' and terminado = '0') then

                if(to_integer(unsigned(Tempol)) < limites(limite)
) then
                    Tempol <= std_logic_vector(unsigned(Tempol) +
1);
                else
                    if(aceso = '1') then
                        aceso <= '0';
                        indice <= (indice + 1) mod 5;
                        limite <= 0;
                    elsif(aceso = '0') then
                        aceso <= '1';
                        limite <= 1;

                        if(alfabeto(numero)(indice) = "11") then
                            limites(1) <= 75000000;
                        elsif(alfabeto(numero)(indice) = "10")
then
                            limites(1) <= 25000000;
                        else

```

```

                                terminado <= '1';
                                indice <= 0;
                                limite <= 0;
                                aceso <= '0';
                            end if;
                        end if;
                    Tempol <= (others => '0');
                end if;
            end if;
        end process;

        led <= aceso;

end main;
```