
Laboratório Digital II - Relatório 2



Prof. Dr. Paulo Cugnasca

Bancada B5

Otávio Felipe de Freitas - 11261249

William Abe Fukushima - 11261771

Gabriel Pereira de Carvalho - 11257668

13 de setembro de 2021

Sumário

1	Objetivo e Abrangência do Projeto	3
2	Solução Técnica	4
2.1	Controle do Servo Motor	4
2.2	Simulação	5
2.3	Designação de Pinos para o <i>Analog Discovery</i>	7
2.4	Setup MQTT Dashboard	7
3	Códigos	9

1 Objetivo e Abrangência do Projeto

O objetivo deste projeto é a familiarização com circuitos de controle utilizando servomotores. A experiência envolve atividades introdutórias de servomotores para familiarização com a modulação PWM e a implementação de circuitos para controlar suas posições.

2 Solução Técnica

2.1 Controle do Servo Motor

Utilizando como referência a descrição do arquivo `circuito_pwm.vhd` fornecido pelo time de professores, alteramos a contagem máxima para 1000000. Portanto, como o período da FPGA é 20 ns, será obtido um período de 20 ms para a PWM. Dessa mesma forma, para obter as larguras equivalentes a 1 ms, 1.5 ms e 2 ms, o código [3] terá um sinal – `motor_width` – o qual possui valor de acordo com a posição do potenciômetro. O valor que essa variável adquire é apresentada na tabela [1].

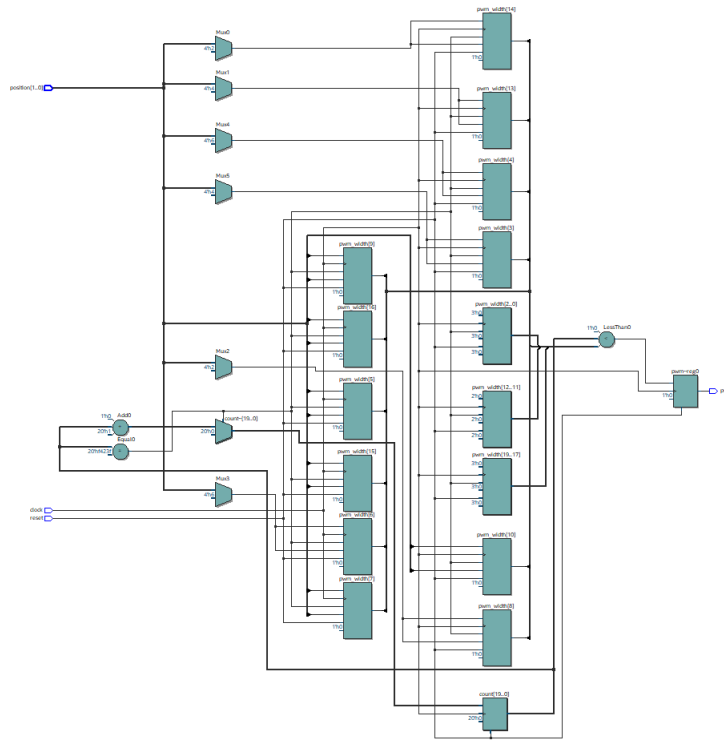


Figura 1: Diagrama RTL da descrição de `controle_servo.vhd`

Largura (ms)	Ciclos de Clock
1	50000.00
1.5	75000.00
2	100000.00

Tabela 1: Ciclos de clock para atingir a largura em ms

2.2 Simulação

Com o código do testbench fornecido, realizou-se a simulação no Modelsim e os seguintes resultados foram obtidos:

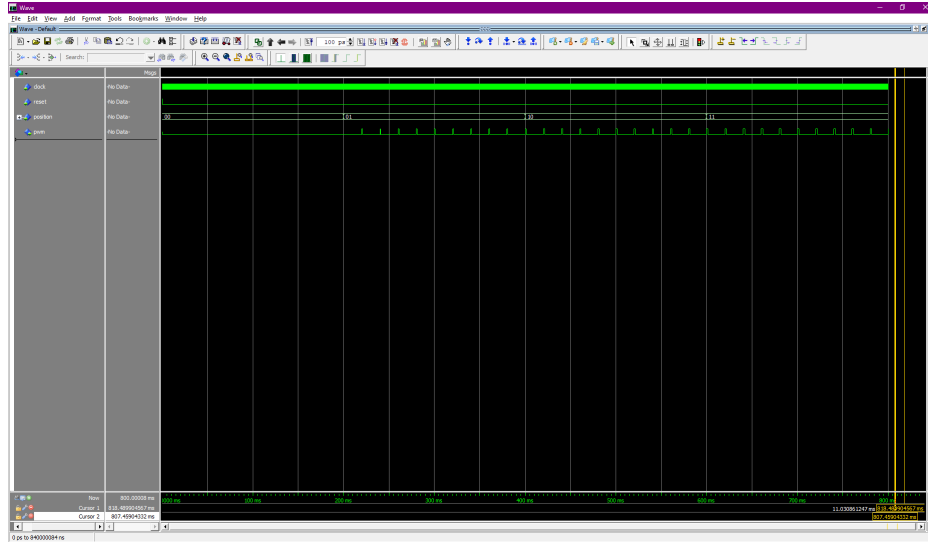


Figura 2: Captura de tela abrangendo toda a simulação

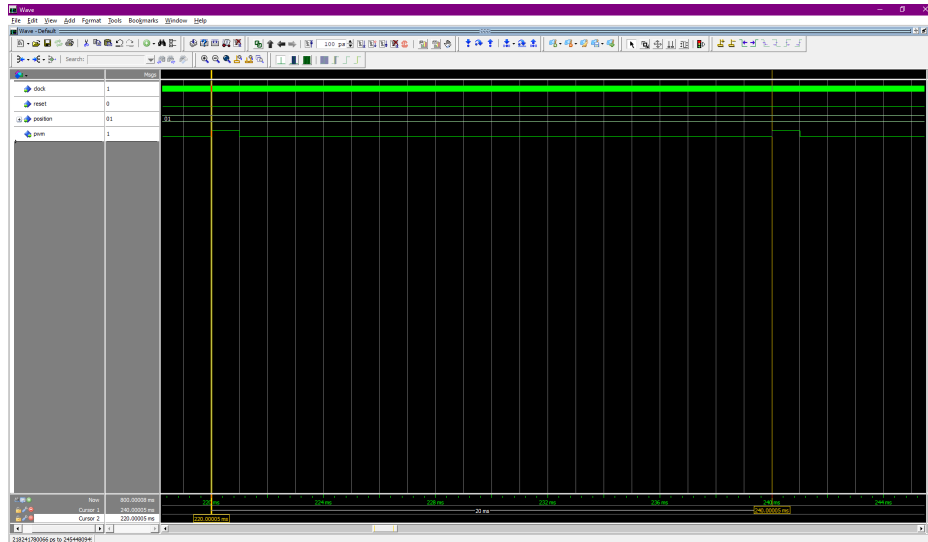


Figura 3: Período PWM

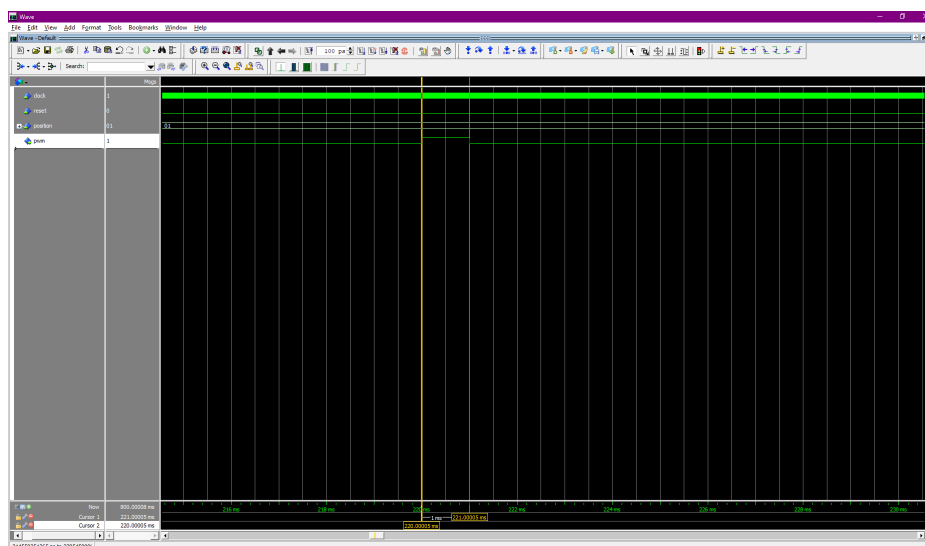


Figura 4: Tempo ativo do servo motor para posição "01"

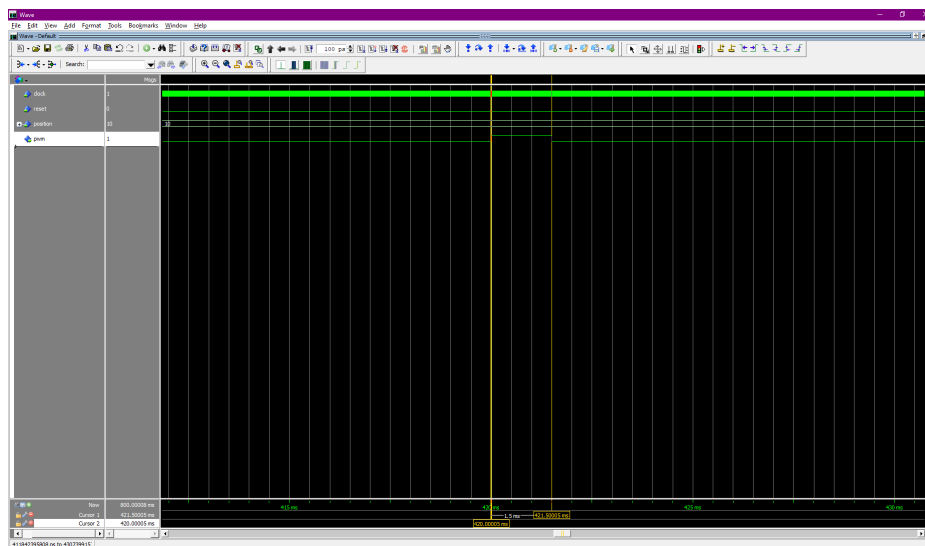


Figura 5: Tempo ativo do servo motor para posição "10"

2.3 Designação de Pinos para o Analog Discovery 2 SOLUÇÃO TÉCNICA

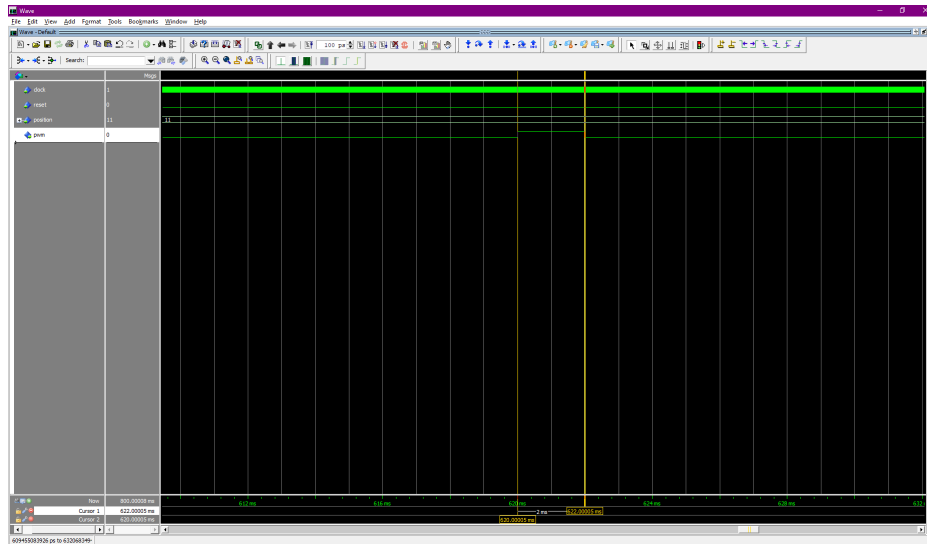


Figura 6: Tempo ativo do servo motor para posição "11"

2.3 Designação de Pinos para o Analog Discovery

Após os testes, foi feita a pinagem da placa FPGA e do equipamento Analog Discovery.

Sinal	Ligação na placa FPGA	Pino na FPGA	Analog Discovery
clock	CLK_50	PIN_M9	-
reset	GPIO_0.D27	PIN_P18	DIO0
position[0]	GPIO_0.D29	PIN_R17	DIO1
position[1]	GPIO_0.D31	PIN_T20	DIO2
pwm	GPIO_1.D35	PIN_K16	-

2.4 Setup MQTT Dashboard

Após a familiarização com os servomotores foi feito o setup do MQTT Dashboard. Conforme aprendemos na experiência 0 configuramos os widgets *switch/button* nas entradas E0,E1 e E2 para controlar o projeto via MQTT.



Figura 7: Tela do projeto no MQTT Dash para controle do servo

3 Códigos

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity controle_servo is
port (
    clock    : in  std_logic;
    reset    : in  std_logic;
    position : in  std_logic_vector(1 downto 0);
    pwm      : out std_logic
);
end controle_servo;

architecture rtl of controle_servo is

    constant MAX_COUNT : integer := 1000000;

    signal count      : integer range 0 to (MAX_COUNT - 1);
    signal pwm_width  : integer range 0 to (MAX_COUNT - 1);
    signal motor_width : integer range 0 to (MAX_COUNT - 1);

begin

    process (clock, reset, position)
    begin
        if (reset = '1') then
            count      <= 0;
            pwm        <= '0';
            pwm_width <= motor_width;
        elsif (rising_edge(clock)) then
            if (count < pwm_width) then
                pwm <= '1';
            else
                pwm <= '0';
            end if;

            if (count = MAX_COUNT - 1) then
                count      <= 0;
                pwm_width <= motor_width;
            else
                count <= count + 1;
            end if;
        end if;
    end process;
end architecture;
```

```
process (position)
begin
    case position is
        when "01" => motor_width <= 50000;
        when "10" => motor_width <= 75000;
        when "11" => motor_width <= 100000;
        when others => motor_width <= 0;
    end case;
end process;

end rtl;
```