



Universidade de Brasília - UnB
Departamento de Engenharia Eletrica - ENE

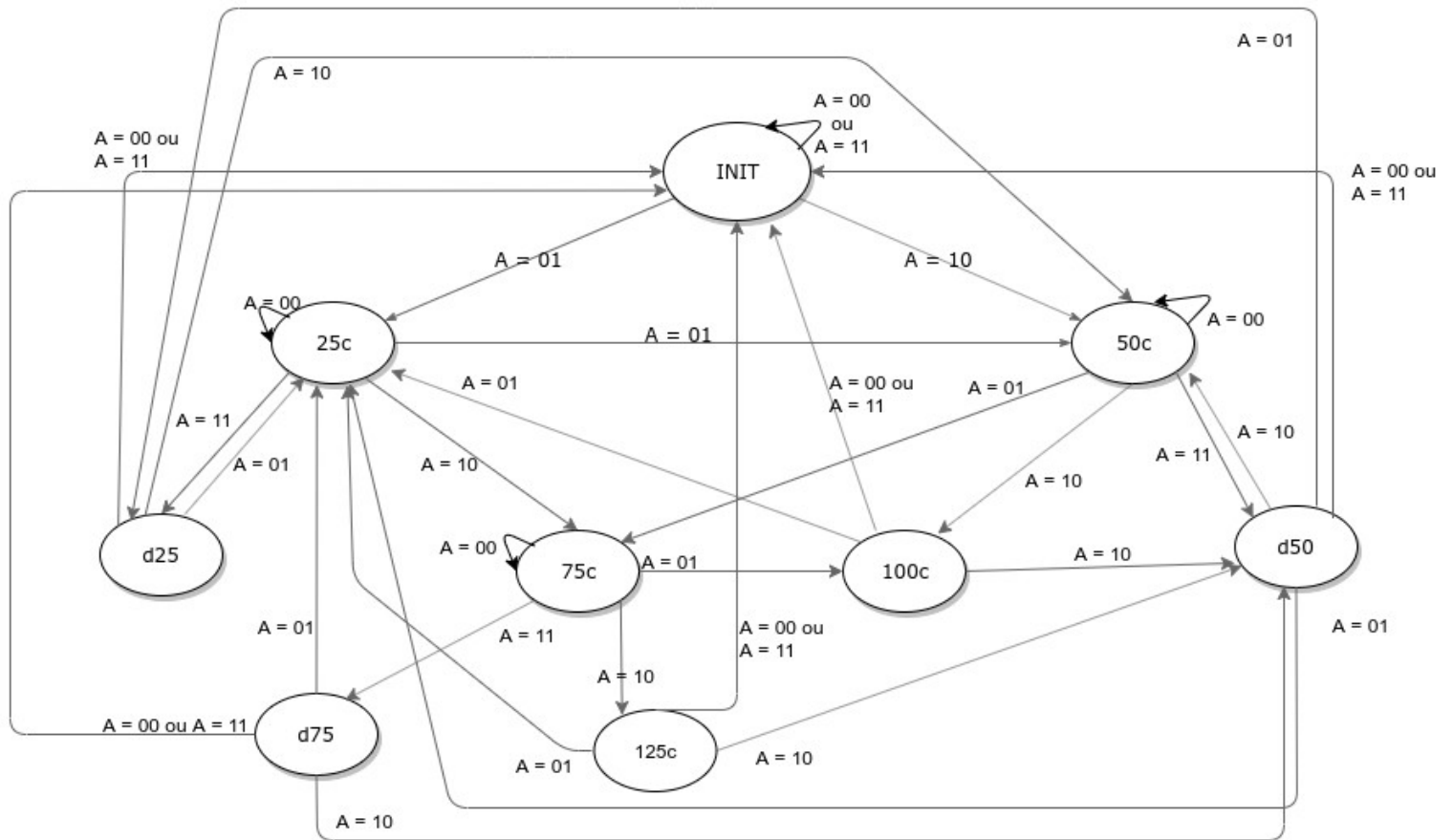
Pré-relatório Experimento 7
Máquina de estados síncrona Moore

Carlos Eduardo Taborda Lottermann
180041509

Prof. Luís Fernando Ramos Molinaro

Brasília, DF
2019

Diagrama de estados:



Código em VHDL:

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

entity Visto is
    Port ( A : in  STD_LOGIC_VECTOR (1 downto 0);
          CLK : in  STD_LOGIC;
          RESET : in STD_LOGIC;
          R, OUT_25, OUT_50 : out STD_LOGIC);
end Visto;

architecture Visto_op of Visto is
    type state is (INIT, c25, c50, c75, c100, c125, D25, D50, D75);
    signal estadoAtual, proxEstado : state;
begin

    sync_process: process(CLK, RESET)
    begin
        if (RESET = '1') then
            estadoAtual <= INIT;
        elsif rising_edge(CLK) then
            estadoAtual <= proxEstado;
        end if;
    end process;

    comb_process : process(estadoAtual, A)
    begin
        case estadoAtual is
            when INIT =>
                R <= '1';
                OUT_25 <= '0';
                OUT_50 <= '0';
            case A is
                when "00" => proxEstado <= INIT;
                when "01" => proxEstado <= c25;
                when "10" => proxEstado <= c50;
                when "11" => proxEstado <= INIT;
```

```

        when others => null;
    end case;

when c25 =>
    R <= '1';
    OUT_25 <= '0';
    OUT_50 <= '0';
    case A is
        when "00" => proxEstado <= c25;
        when "01" => proxEstado <= c50;
        when "10" => proxEstado <= c75;
        when "11" => proxEstado <= d25;
        when others => null;
    end case;

when c50 =>
    R <= '1';
    OUT_25 <= '0';
    OUT_50 <= '0';
    case A is
        when "00" => proxEstado <= c50;
        when "01" => proxEstado <= c75;
        when "10" => proxEstado <= c100;
        when "11" => proxEstado <= d50;
        when others => null;
    end case;

when c75 =>
    R <= '1';
    OUT_25 <= '0';
    OUT_50 <= '0';
    case A is
        when "00" => proxEstado <= c75;
        when "01" => proxEstado <= c100;
        when "10" => proxEstado <= c125;
        when "11" => proxEstado <= d75;
    end case;

```

```

        when others => null;

    end case;

when c100 =>
    R <= '1';
    OUT_25 <= '0';
    OUT_50 <= '0';
    case A is
        when "00" => proxEstado <= INIT;
        when "01" => proxEstado <= c25;
        when "10" => proxEstado <= c50;
        when "11" => proxEstado <= INIT;
        when others => null;
    end case;

when c125 =>
    R <= '1';
    OUT_25 <= '1';
    OUT_50 <= '0';
    case A is
        when "00" => proxEstado <= INIT;
        when "01" => proxEstado <= c25;
        when "10" => proxEstado <= c50;
        when "11" => proxEstado <= INIT;
        when others => null;
    end case;

when d25 =>
    R <= '0';
    OUT_25 <= '1';
    OUT_50 <= '0';
    case A is
        when "00" => proxEstado <= INIT;
        when "01" => proxEstado <= c25;
        when "10" => proxEstado <= c50;
        when "11" => proxEstado <= INIT;

```

```

        when others => null;
    end case;

when d50 =>
    R <= '0';
    OUT_25 <= '0';
    OUT_50 <= '1';
    case A is
        when "00" => proxEstado <= INIT;
        when "01" => proxEstado <= c25;
        when "10" => proxEstado <= c50;
        when "11" => proxEstado <= INIT;
        when others => null;
    end case;

when d75 =>
    R <= '0';
    OUT_25 <= '1';
    OUT_50 <= '1';
    case A is
        when "00" => proxEstado <= INIT;
        when "01" => proxEstado <= c25;
        when "10" => proxEstado <= c50;
        when "11" => proxEstado <= INIT;
        when others => null;
    end case;

end case;

end process;

end Visto_op;

```