

# **HAMMING CODE – ARQUITETURAS DE ALTO DESEMPENHO**

---

Hugo Moinheiro, 84931

Mariana Pinto, 84792

# ENCODER – PARALLEL VERSION

Devido à ordenação da matriz foi necessário inverter os bits de entrada para cálculo dos bits de paridade.

Com a redução das equações conseguiu-se:

- 18 xors com 3 tempos de atraso

```
begin

xorA: xor4 port map (d(10), d(4), d(3), d(0), xor_a);
xorB: xor4 port map (d(5), d(2), d(1), d(0), xor_b);
par1: xor4 port map (d(9), d(8), d(2), xor_a, p1);
par2: xor4 port map (d(7), d(6), d(1), xor_a, p2);
par3: xor4 port map (d(9), d(7), d(4), xor_b, p3);
par4: xor4 port map (d(8), d(6), d(3), xor_b, p4);

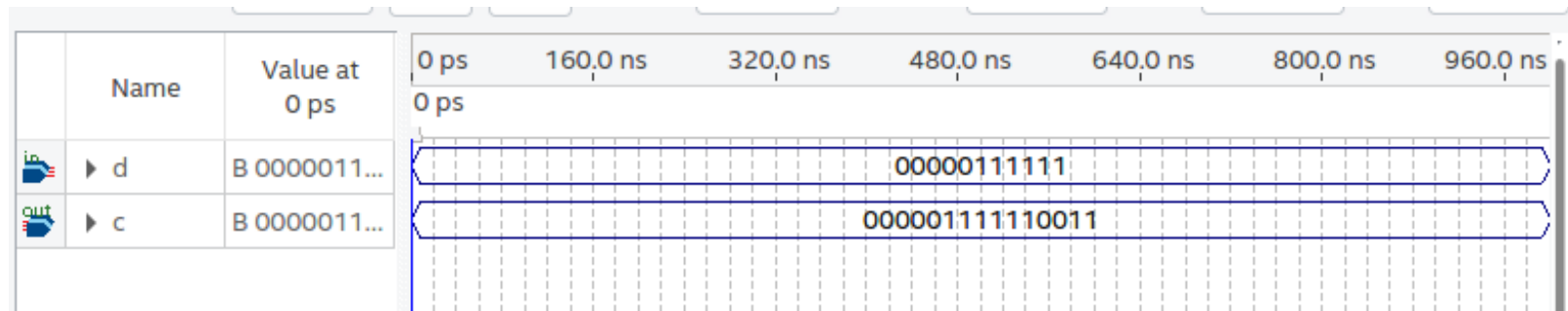
c <= d(10 downto 0) & p1 & p2 & p3 & p4;

end structural;
```

# ENCODER – PARALLEL VERSION TESTS

Validation Program

```
message m(m1,m2, ... ,m11) = 0 0 0 0 0 1 1 1 1 1 1  
encoded message x(x1,x2, ... ,x15) = 0 0 0 0 0 1 1 1 1 1 1 0 0 1 1
```



Simulation Program

# DECODER – PARALLEL VERSION

Descodificador para corrigir o erro em relação ao valor dado pelos bits de paridade.

Dado o valor do erro do descodificador

Com a redução das equações conseguiu-se:

- 22 xors com 4 tempos de atraso

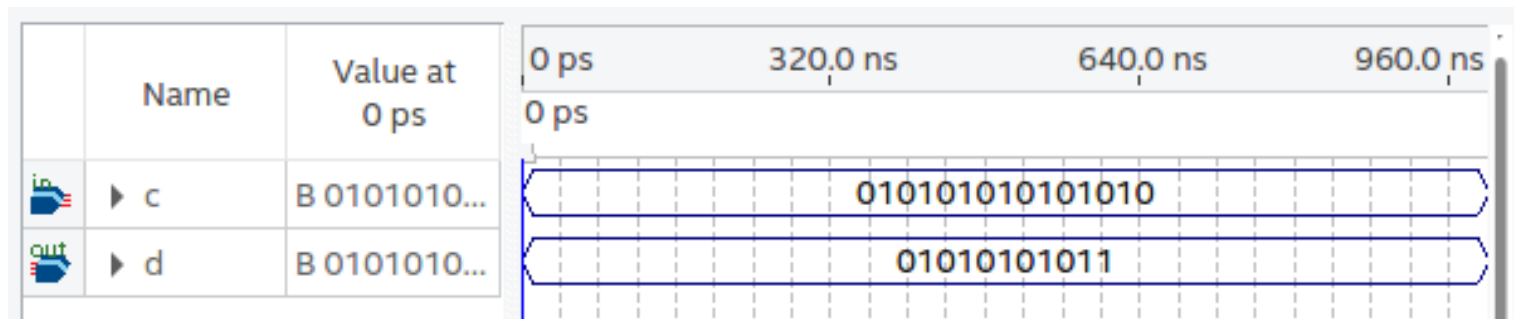
```
begin
  process(par)
  begin
    if (par = "1100") then
      word <= "10000000000";
    elsif (par = "1010") then
      word <= "01000000000";
    elsif (par = "1001") then
      word <= "00100000000";
    elsif (par = "0110") then
      word <= "00010000000";
    elsif (par = "0101") then
      word <= "00001000000";
    elsif (par = "0011") then
      word <= "00000100000";
    elsif (par = "1110") then
      word <= "00000010000";
    elsif (par = "1101") then
      word <= "00000001000";
    elsif (par = "1011") then
      word <= "00000000100";
    elsif (par = "0111") then
      word <= "00000000010";
    elsif (par = "1111") then
      word <= "00000000001";
    end if;
  end process;
end structural;
```

```
process(temp, parWord)
begin
  temp <= c(22 downto 4);
  if (parWord = "1100") then
    temp(10) <= not c(14);
  elsif (parWord = "1010") then
    temp(9) <= not c(13);
  elsif (parWord = "1001") then
    temp(8) <= not c(12);
  elsif (parWord = "0110") then
    temp(7) <= not c(11);
  elsif (parWord = "0101") then
    temp(6) <= not c(10);
  elsif (parWord = "0011") then
    temp(5) <= not c(9);
  elsif (parWord = "1110") then
    temp(4) <= not c(8);
  elsif (parWord = "1101") then
    temp(3) <= not c(7);
  elsif (parWord = "1011") then
    temp(2) <= not c(6);
  elsif (parWord = "0111") then
    temp(1) <= not c(5);
  elsif (parWord = "1111") then
    temp(0) <= not c(4);
  end if;
end process;
d <= temp;
```

# DECODER – PARALLEL VERSION TESTS

Validation Program

```
distorted encoded message y(y1,y2, ... ,y15) = 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0  
retrieved message ml(ml1,ml2, ... ,ml11) = 0 1 0 1 0 1 0 1 0 1 1
```



Simulation Program