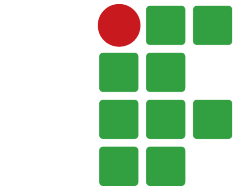

INSTITUTO FEDERAL
Santa Catarina

Câmpus
São José

Relatorio 01

DISPOSITIVOS LÓGICOS PROGRAMÁVEIS II (DLP029007)

Rhenzo Hideki Silva Kajikawa

20 de Setembro de 2023

Sumário

1. Caminho crítico dos operadores

Faça a implementação de um somador para valores de 16 bits sem sinal. Escreva 5 implementações em VHDL para as operações abaixo. Sintetize usando o Quartus e o dispositivo da DE2-115.

a) a+b

b) a + “0000000000000001”

c) a + “0000000010000000”

d) a + “1000000000000000”

e) a + “1010101010101010”

Verifique a área (LE) e atraso (ns) para cada implementação.

Discussão:

Qual implementação usou menos área? Por quê? Como o delay se comportou?

Comente as diferenças entre os valores de área e delay obtidos nas operações a), b) e c).

2. Resultados

2.1. Código

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

entity AP1 is

  generic
  (
    DATA_WIDTH : natural := 16
  );

  port
  (
    a    : in std_logic_vector((DATA_WIDTH-1) downto 0);
    b    : in std_logic_vector((DATA_WIDTH-1) downto 0);
    s : out std_logic_vector((DATA_WIDTH-1) downto 0)
  );

end entity;

architecture rtl of AP1 is
  signal soma : unsigned(DATA_WIDTH-1 downto 0);
  --constant b: std_logic_vector(DATA_WIDTH-1 downto 0):= "0000000000000001";
  --constant b: std_logic_vector(DATA_WIDTH-1 downto 0):= "0000000010000000";
  --constant b: std_logic_vector(DATA_WIDTH-1 downto 0):= "1000000000000000";
  --constant b: std_logic_vector(DATA_WIDTH-1 downto 0):= "1010101010101010";
begin
  soma <= unsigned(a) + unsigned(b);
  s <= std_logic_vector(soma);

end rtl;
```

2.2. Tabela

	Área(LE)	Delay Completo (ns)	Delay sem IO (ns)
A	16 / 114,480 (< 1 %)	12.853	9.24
B	15 / 114,480 (< 1 %)	11.263	6.966
C	8 / 114,480 (< 1 %)	9.474	5.803
D	0 / 114,480 (0 %)	8.510	3.455
E	14 / 114,480 (< 1 %)	11.932	7.515

2.3. Qual implementação usou menos área? Por quê? Como o delay se comportou?

A implementação com menos área e delay é a implementação *D*. Isso ocorre pois ela é uma implementação que pode acontecer com somadores localizados nos buffers de IO da placa , isso se deve pois no IO buffers são encontrados somadores de 1 bits e a soma ocorre com o valor de b fixo de “1000000000000000”. Por esse motivo , não aparece no relatorio gerado pelo Quartus nenhum elemento lógico sendo utilizado e também por não depender de nenhum somador seu delay acaba sendo baixo.

2.4. Comente as diferenças entre os valores de área e delay obtidos nas operações a), b) e c)

A diferenças das operações de *A*, *B* e *C* se devem ao fato das diferentes posições possiveis com o valor de bit 1. A operação *A* tem a maior área pois é necessária garantir que a soma cubra todos o valores de bits, pois existem 2 entradas aleatórias. A operação *B* tem o valor de *b* fixo de “0000000000000001” , dessa forma foi utilizado 1 somador a menos , pois ainda é necessário garantir a soma do carry caso ocorra. Por fim a operação c tem como valor “0000000010000000”, como os bits que precedem o 1 são 0 não irão gerar nenhum Carry para a soma então é apenas necessário garantir somadores apartir do bit 1.