

# UNIVERSIDADE DE SÃO PAULO

Escola de Engenharia de São Carlos  
SEL0621 - Projetos de Circuitos Integrados Digitais I  
Prof. Dr. João Pereira do Carmo

---

## Projeto 11

Davi Diório Mendes	7546989
Nivaldo Henrique Bondança	7143909

---



14 de novembro de 2014

# *Lista de Figuras*

1	Prescaler 32/33. . . . .	p. 5
2	Simulação no <i>ModelSim</i> do divisor por 8 . . . . .	p. 7
3	Caminho crítico gerado pelo <i>LeonardoSpectrum</i> do divisor por 8 . . . . .	p. 7
4	Esquemático gerado pelo <i>LeonardoSpectrum</i> do divisor por 8 . . . . .	p. 7
5	Esquemático gerado no <i>ICStation</i> do divisor por 8 . . . . .	p. 8
6	Máxima frequência de operação . . . . .	p. 8
7	<i>Layout</i> do circuito divisor por 8 . . . . .	p. 9
8	Máxima frequência de operação (R+C+CC) . . . . .	p. 9
9	Simulação no <i>ModelSim</i> do <i>Prescaler</i> completo, SM = '1' . . . . .	p. 10
10	Simulação no <i>ModelSim</i> do <i>Prescaler</i> completo, SM = '0' . . . . .	p. 10
11	Esquemático do circuito, modelo hierarquico . . . . .	p. 11
12	Esquemático do circuito, modelo <i>flat</i> . . . . .	p. 11
13	Caminho crítico gerado pelo <i>LeonardoSpectrum</i> do <i>Prescaler</i> completo . . . . .	p. 11
14	<i>Layout</i> do <i>Prescaler</i> completo . . . . .	p. 12
15	<i>Layout</i> do <i>Prescaler</i> completo, modelo <i>flat</i> . . . . .	p. 13
16	Máxima frequência de operação (R+C+CC) . . . . .	p. 13
17	Razão de potência consumida por frequência . . . . .	p. 14

# *Lista de Tabelas*

1	Máximas frequência de operação. . . . .	p. 10
---	---	-------

# ***Códigos Fontes***

1	Descrição do divisor por 8 . . . . .	p. 5
2	Descrição do <i>Prescaler</i> completo . . . . .	p. 10

## Resumo

## Introdução

O objetivo deste projeto é trabalhar com alguns circuitos em VHDL e explorar hierarquias. Também será visto o ModelSim para simulação de VHDL.

1. Considere o circuito prescaler da **Figura 1**.

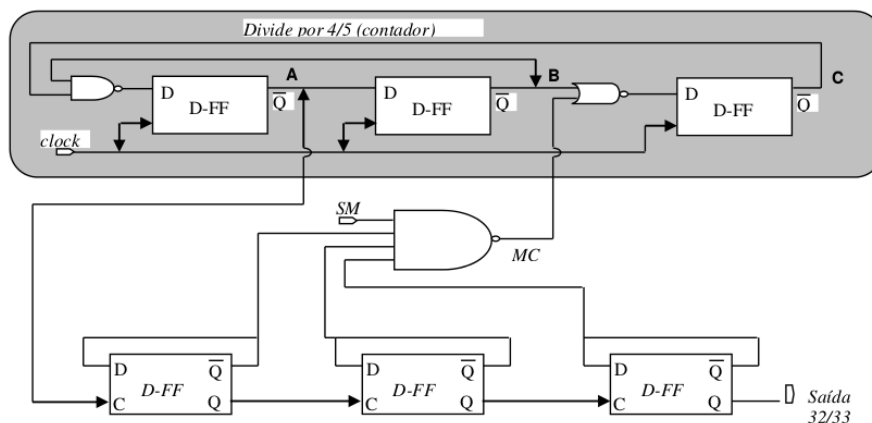


Figura 1: Prescaler 32/33.

O circuito composto pelos blocos não hachurados (três *D-flip flops* e o NAND de quatro entradas) compõe um contador assíncrono de 0 até 7 (conta 8). O circuito pode ser visto como tendo dois sinais de entrada (os sinais SM e um clock) e dois sinais de saída (os sinais MC e saída). A saída MC é igual a “0” apenas quando SM = “1” e o contador esta no “0” da contagem. O sinal saída é o clock dividido por 8.

2.\* Descreva o contador assíncrono em *VHDL* (não fazer a descrição estrutural que repete o esquemático, mas sim em alto nível). Utilize o manual de *VHDL* do *software* Leonardo (*LeonardoSpectrum HDL Synthesis Manual*) para ver modelos se necessário.

A descrição do circuito está representada em **Código fonte 1**.

Código Fonte 1: Descrição do divisor por 8

```

1 entity div_8 is
2
3     port ( SM, CLK : in  bit;
4           MC, S   : out bit );
5
6 end div_8;
7
8 architecture div_8_imp of div_8 is
9 begin

```

```
10
11     registers : process (CLK, SM)
12         variable c : integer range 0 to 7;
13     begin
14         if (CLK'event and CLK = '1') then
15             if(c = 7) then
16                 c := 0;
17             else
18                 c := c + 1;
19             end if;
20
21             if (c > 3) then
22                 S <= '1';
23             else
24                 S <= '0';
25             end if;
26         end if;
27
28         if (c = 0 and SM='1') then
29             MC <= '0';
30         else
31             MC <= '1';
32         end if;
33     end process;
34
35 end div_8_imp;
```

**3.\*** Simule no *ModelSim* o circuito e verifique seu funcionamento. Para isso:

- crie uma pasta de trabalho, *work*, caso ainda não tenha feito;
- compile o seu *VHDL*;
- inicie a simulação, *Simule > Start Simulation*. Caso não sejam abertas as janelas de *wave* e *objectcs*, abri-las via comandos *View - Wave* e *View - Objects*.
- crie os sinais de entrada (na janela de *Objects*, *Create Wave*);
- coloque na janela *waves* os sinais de saída que deseja ver (“arraste” os sinais de uma janela a outra);
- execute a simulação.

Apresente o resultado gráfico da simulação.

O resultado da simulação se encontra na **Figura 2**.

**4.** Carregue o circuito *VHDL*, sintetize e otimize para *delay* utilizando a biblioteca da *AMS*.

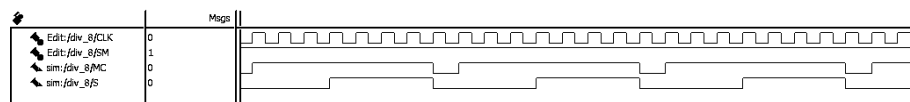


Figura 2: Simulação no *ModelSim* do divisor por 8

5.\*Compare a implementação feita no *Leonardo* com a da **Figura 1**. Qual é a maior diferença?

A maior diferença é que a implementação da **Figura 1** é assíncrona, enquanto a feita em VHDL gerada pelo *LeonardoSpectrum* (antes de ser otimizada) é síncrona.

6.\*Verifique o caminho crítico e estime a máxima frequência de operação que este circuito pode atingir (apresente a figura com caminho crítico).

O caminho crítico está representado na **Figura 3**. Como o atraso do circuito é de  $1,19ns$  sua máxima frequência de operação será  $0,84GHz$ .

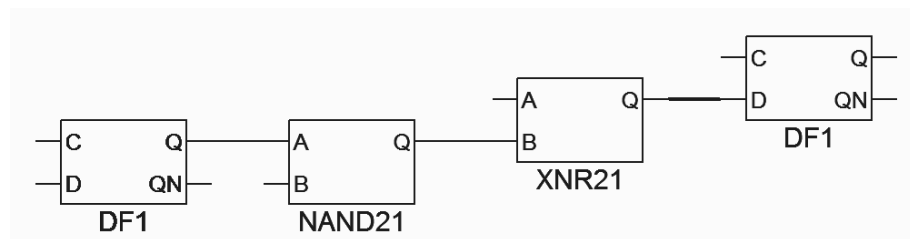


Figura 3: Caminho crítico gerado pelo *LeonardoSpectrum* do divisor por 8

7.\*Apresente no relatório o VHDL e o esquemático final que obteve.

O esquemático do circuito está representado na **Figura 4**.

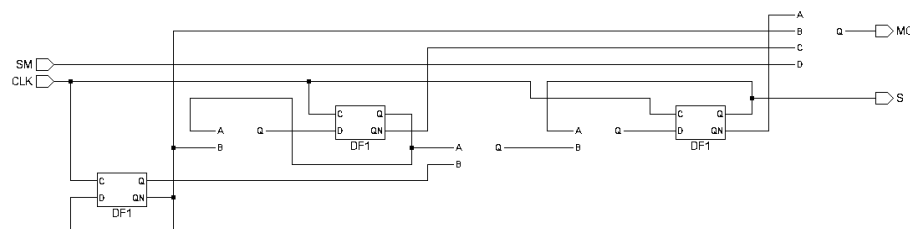


Figura 4: Esquemático gerado pelo *LeonardoSpectrum* do divisor por 8

8.\*Exporte o circuito gerado para o *ICStation* e gere o esquemático. A partir do esquemático extraia o circuito para simulação e determine a máxima frequência de operação.

O esquemático do circuito está representado na **Figura 5**. Assim como mostra o gráfico da **Figura 6**, a máxima frequência de operação é  $1,10GHz$ .

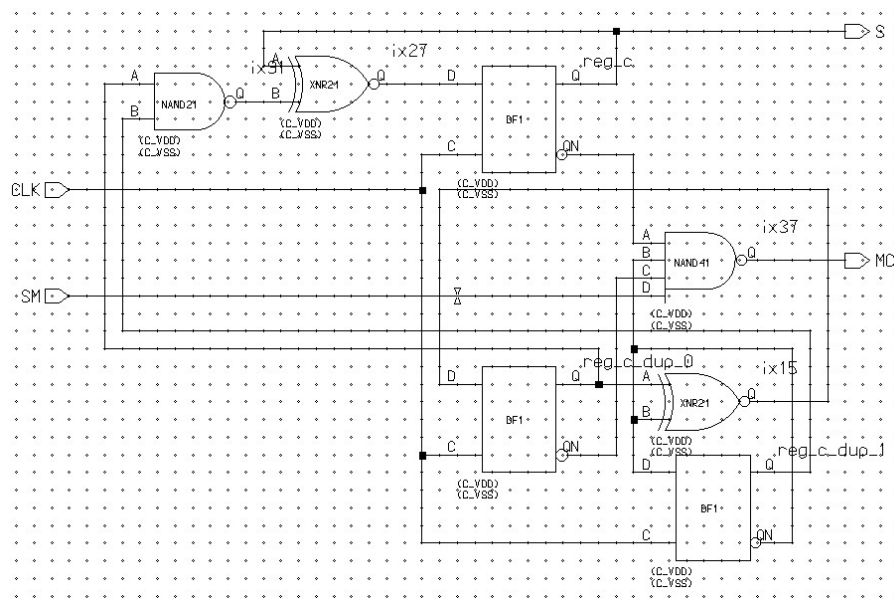


Figura 5: Esquemático gerado no *ICStation* do divisor por 8

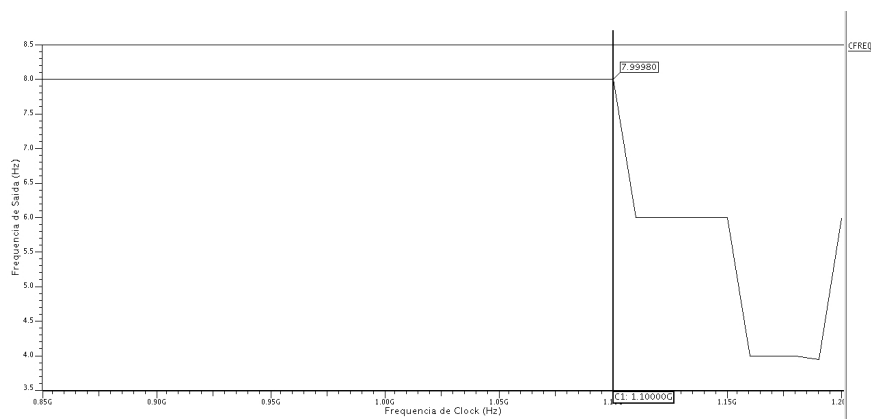


Figura 6: Máxima frequência de operação

**9.**Faça o *placement* automático das células. Dentro do menu *Place & Route* utilize os comando *MvOnRw* (*move on row*) e *MvInRow* (*move into rows*) para reposicionar as células para melhorar o *layout*.

**10. \*Faça o roteamento das células sem usar poli (utilize no roteamento do VDD e do VSS linhas de  $1,8\mu m$ . Qual é a máxima corrente que pode passar pela alimentação nesse caso?).**

De acordo com as especificações do MET1, sua densidade de corrente é  $1,0mA/\mu m$ . Portanto, a máxima corrente será  $1,8mA$ .

11. \*Coloque os *ports*, passe o *DRC*, elimine os erros e faça o LVS. Apresente o *layout* obtido no relatório.

O *layout* do circuito está representado na **Figura 7**.



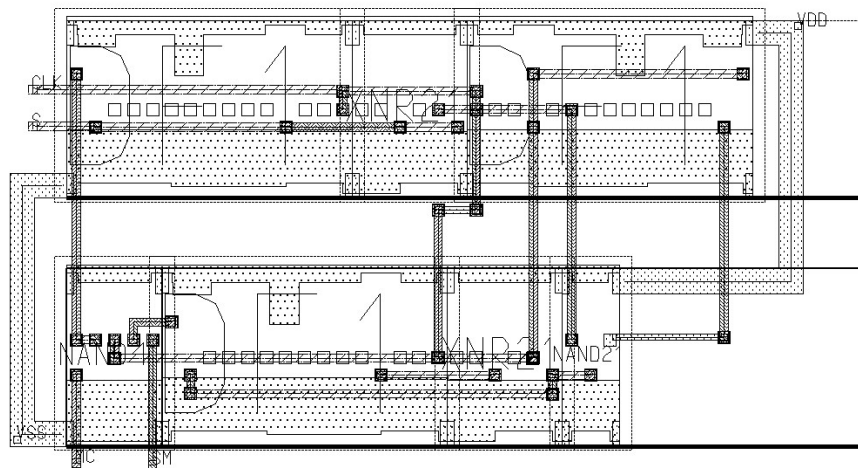


Figura 7: *Layout* do circuito divisor por 8

12. \*Simule o circuito extraído do *layout* (R+C+CC) e determine sua máxima velocidade.

Assim como mostra o gráfico da **Figura 8**, a máxima frequência de operação é 0,77GHz.

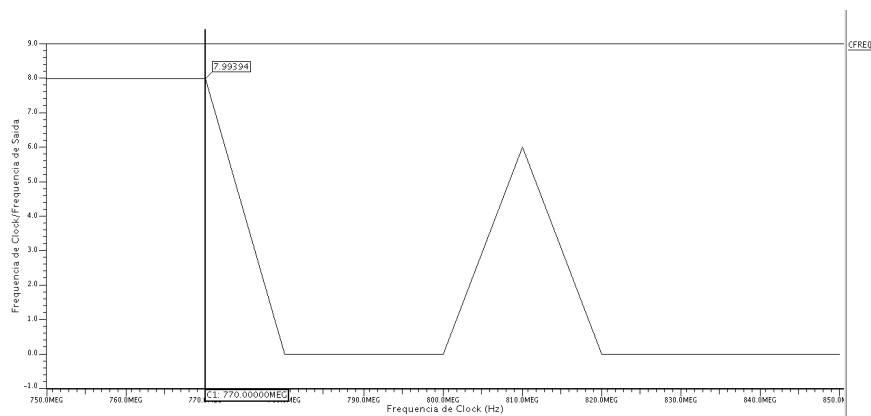


Figura 8: Máxima frequência de operação (R+C+CC)

13. \*Compare os resultados de velocidade obtidos nas questões 6, 8 e 12.

Após uma análise dos dados, percebe-se que o melhor resultado veio da simulação do esquemático, o que faz sentido, devido a ele utilizar modelos um tanto ideais do circuito, sem simulação de elementos parasitas como a simulação pelo *layout* apresenta. A simulação R+C+CC é a simulação mais próxima do real, portanto os resultados obtidos na simulação do *layout* é a mais confiável.

14. \*Com os dois blocos já escritos em VHDL, contador 4/5 e contador 8, monte o *Prescaler* completo VHDL (agora descrevendo um bloco conectado ao outro). Apresente o VHDL.

Tabela 1: Máximas frequência de operação.

Modelo	Frequência (GHz)
<i>Critical Path</i>	0,84
Esquemático	1,10
<i>Layout</i>	0,77

A descrição do circuito está representada em **Código fonte 2**.

Código Fonte 2: Descrição do *Prescaler* completo

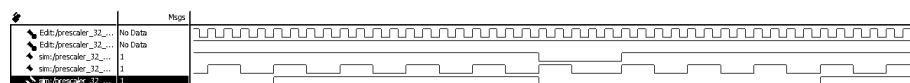
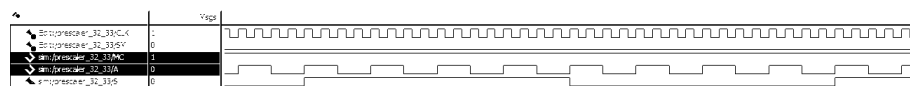
```

1  entity prescaler_32_33 is
2
3      port(SM, CLK : IN  BIT;
4            S      : OUT BIT );
5
6  end prescaler_32_33;
7
8  architecture prescaler_32_33_imp of prescaler_32_33 is
9      signal MC, A : BIT;
10  begin
11
12      div_4_5 : entity work.divisor_4_5 port map(MC, CLK, A);
13      div8    : entity work.div_8      port map(SM, A, MC, S);
14
15  end prescaler_32_33_imp;

```

**15.** \*Simule no *ModelSim* o circuito e verifique seu funcionamento. Apresente os resultados gráficos da simulação.

O resultado das simulações encontram-se nas **Figura 9**, **Figura 10**.

Figura 9: Simulação no *ModelSim* do *Prescaler* completo, SM = '1'Figura 10: Simulação no *ModelSim* do *Prescaler* completo, SM = '0'

**16.** Faça a síntese e teste duas possibilidades disponíveis no Leonardo: mantendo a hierarquia e *flat* (apresente os esquemáticos)

As duas possibilidades de esquemático estão representadas nas **Figura 11** e **Figura 12**.

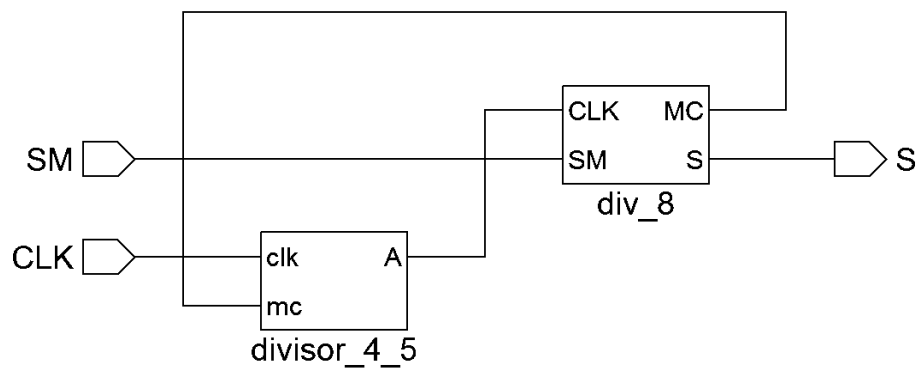


Figura 11: Esquemático do circuito, modelo hierarquico

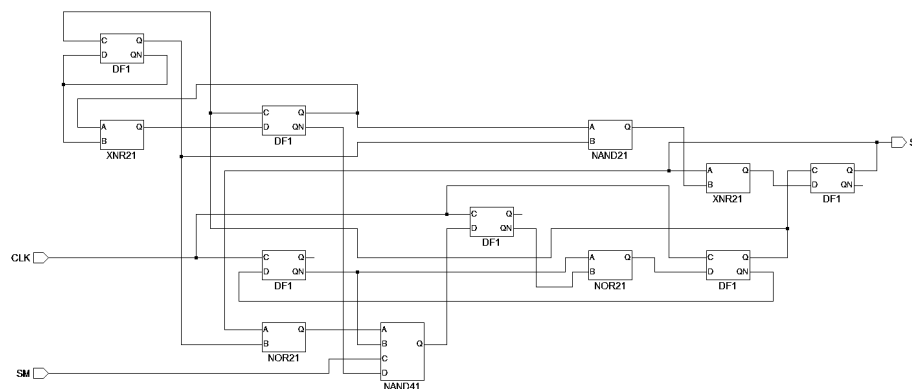
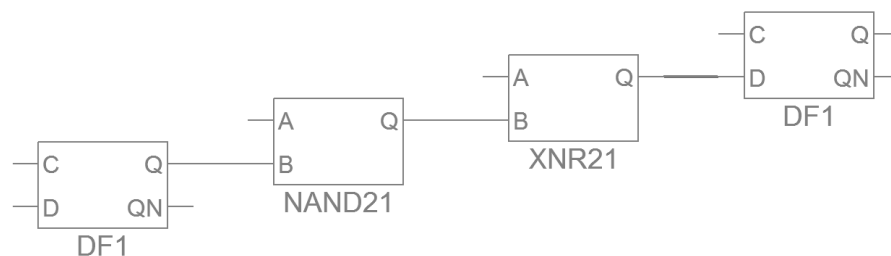


Figura 12: Esquemático do circuito, modelo flat

17. \*Verifique, para o caso *flat*, qual é o caminho crítico (apresente a figura com caminho crítico e indique no esquemático completo onde ele está). Está correto?

O caminho crítico está representado na **Figura 13** e na **Figura ??**. Como o atraso do circuito é de  $1,27ns$  sua máxima frequência de operação será  $0,787GHz$ .

Figura 13: Caminho crítico gerado pelo *LeonardoSpectrum* do *Prescaler* completo

18. Para os dois casos exporte o circuito para o *ICStation*.

19. \*Gere o esquemático em cada caso (deixe-os em bibliotecas diferentes). Qual é a diferença em cada caso? Quando é interessante usar um ou outro?

No caso da hierarquia preservada, não é possível ver o que está dentro dos blocos, o que limita um pouco a flexibilidade de otimização do circuito, mas simplifica a visualização e o trabalho com eles quando o circuito é bastante grande. Já o caso *flat* permite uma maior otimização, mesmo pelo programa *LeonardoSpectrum*.

Ou seja, quando se deseja otimizar bastante, é mais sensato utilizar o modelo *flat*. Quando se deseja melhor modularização e organização dos blocos, é mais sábio utilizar a hierarquia.

**20.** \*A partir do esquemático com hierarquia, monte o *layout* do prescaler completo (aproveite os *layouts* já feitos). Passe o DRC e faça o *LVS*. Apresente o *layout* obtido no relatório.

O *layout* do circuito está representado na **Figura 14**. O modelo *flat*

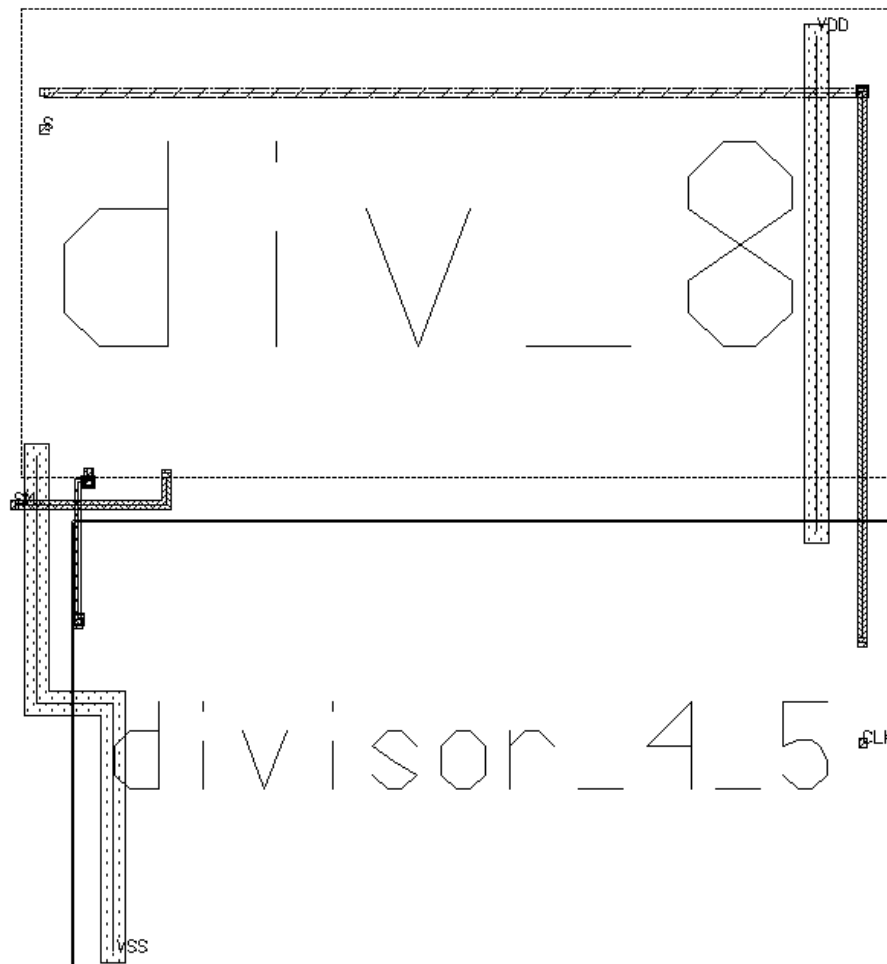


Figura 14: *Layout* do Prescaler completo

**21.** \*Extraia o circuito (R+C+CC), determine a máxima velocidade de operação e, nessa velocidade, determine a razão (potência consumida)/GHz. Determine também a área do circuito.

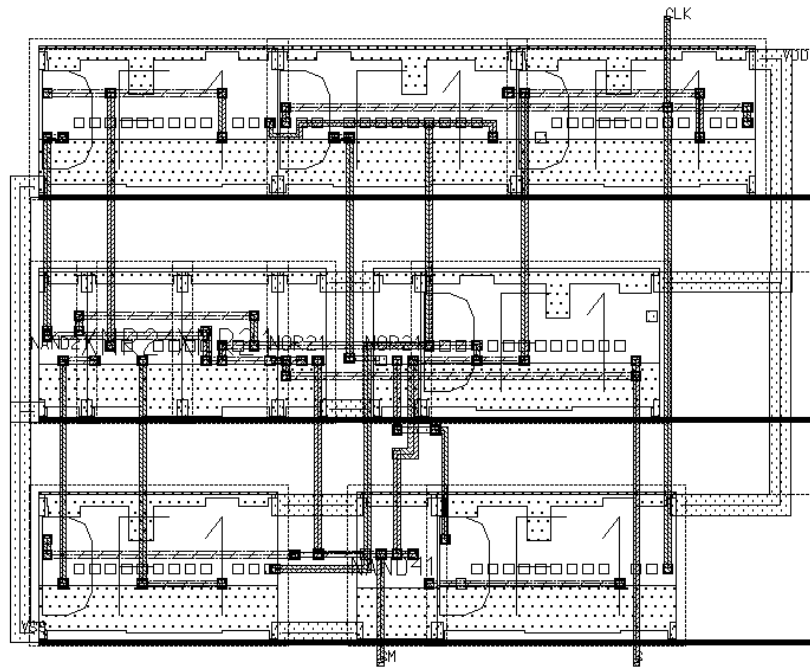


Figura 15: Layout do Prescaler completo, modelo *flat*

Assim como mostra o gráfico da **Figura 16**, a máxima frequência de operação é  $0.915\text{GHz}$ . A razão de potência consumida, em  $0.915\text{GHz}$ , é de  $1,99\text{mW}$ , o que implica uma relação de  $2,17\text{mW}/\text{GHz}$ , assim como mostra a **Figura 17**. As dimensões do circuito são  $63,793\mu\text{m}$  e  $69,300\mu\text{m}$ , o que implica em uma área total de  $4.421\mu\text{m}^2$ .

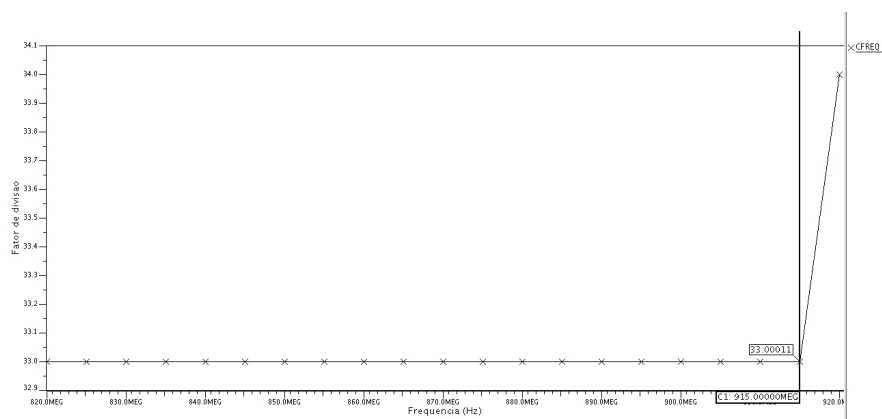


Figura 16: Máxima frequência de operação (R+C+CC)

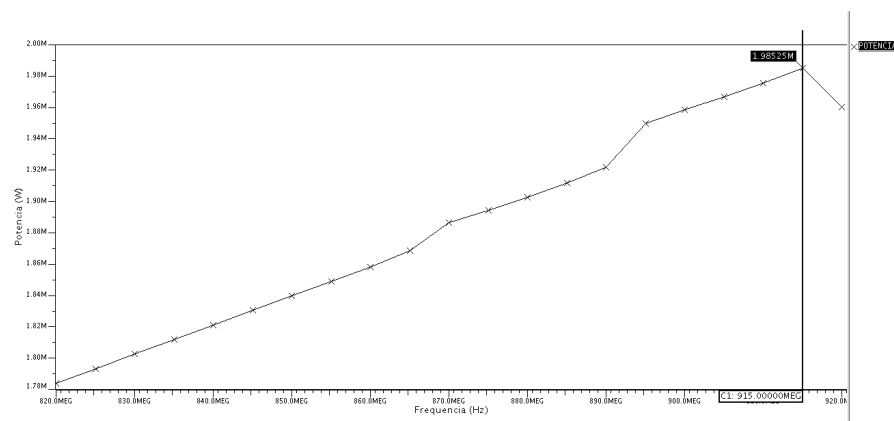


Figura 17: Razão de potência consumida por frequência