

# PROJETO 3

## *Projeto de Circuitos Integrados Analógicos (Sel0618 - set./2014)*

Nesta experiência iniciaremos o uso de células da biblioteca de células da AMS. Para isso será desenvolvido um circuito *Prescaler* 32/33 e feitas simulações sobre ele.

1. Verifique no arquivo

[/local/tools/dkit/ams\\_3.70\\_mgc/www/index.html](/local/tools/dkit/ams_3.70_mgc/www/index.html)

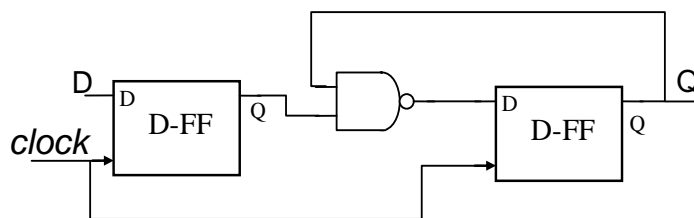
as informações sobre as células digitais disponíveis no processo AMS 0.35  $\mu\text{m}$ . Entenda os dados fornecidos (Standar Cell DataSheet – C35 - Core Cells).

2. \*Procure o *layout* da célula **DL1** e o abra no IcStation. Levante e **desenhe o circuito esquemático** desta célula (não precisa determinar as dimensões dos transistores). Observe com cuidado o *layout* feito.
3. \*Observe que na célula **DL1** as linhas dos sinais de  $V_{DD}$  e  $V_{SS}$  correm nas extremidades superior e inferior. Qual é a dimensão vertical destas linhas e qual a distância entre elas. Abra outra célula da biblioteca e verifique as dimensões das linhas de  $V_{DD}$  e  $V_{SS}$  e a distância entre elas. São iguais em todas as células? Seria/é interessante que fossem/sejam iguais, por quê?

Dimensão vertical: 0.75

Distância: 11.5

4. \*Considere o circuito da **Figura 1**. Desenhe seu **esquemático** utilizando as células **DF3**, **NAND22**. Como sinal de entrada deve colocar o **clock** e **D**; como sinal de saída, **Q**. Gere o símbolo para a célula e faça todas as verificações necessárias. Certifique-se que não haja erros ou *warnings*.



**Figura 1.** Exemplo de esquemático.

5. \*A partir das informações dos blocos que compõe este circuito estime o valor máximo da frequência do *clock* que o circuito pode suportar (considere a entrada  $D = "1"$ , utilize o pior

caso entre subida e descida das portas e considere que os sinais são rápidos). Apresente os cálculos.

6. \*O que significam tempos de *holding* e de *set-up* em *flip-flops*. Qual seria o valor máximo do *clock*, do exercício acima, caso o tempo de *set-up* fosse de 0,1 ns.
7. \*Gere, a partir do esquemático, um arquivo **netlist** para o ELDO. Simule o circuito com os parâmetros típicos e determine a **máxima velocidade (clock)** do circuito. Considere
  - a entrada  $D = "1"$ ;
  - $V_{DD} = 3\text{ V}$ ;
  - o sinal de *clock* com (tempo de subida) = (tempo de descida) =  $0,1 \cdot \text{Período}$ .Apresente as linhas de comando e sinais utilizados.
8. \*Gere agora o **layout para o circuito** (use para isso o *designviepoint* não o *schematic*). Neste *layout* deve-se tomar cuidado com:
  - a área total do circuito;
  - o uso correto dos metais e poli como camadas de conexão;
  - a posição e o tamanho dos *ports* de entrada e saída;
  - a largura das linhas de  $V_{DD}$  e  $V_{SS}$ , não inferiores a  $1\text{ }\mu\text{m}$  (use **Route – ARoute NEt Classe**).Faça a verificação com o **DRC** (CALIBRE), passe o **LVS** e elimine todos os erros.  
**Apresente a figura do layout no relatório.**
9. \*Faça a extração do circuito via o Calibre com a opção **C+CC** (deve ser usada uma configuração que garanta que o nome dos nós venha do *schematic*). Simule o circuito com parâmetros típicos e determine sua máxima velocidade. Utilize as mesmas condições do exercício 7.
10. \*Extraia agora com a opção **R+C+CC**. Simule o circuito com parâmetros típicos e determine sua máxima velocidade. Utilize as mesmas condições do exercício 7.
11. \***Monte uma tabela** com os resultados obtidos nos exercícios **5, 7, 9 e 10**. **Compare e comente** os resultados.
12. \*Determine a área total do circuito que desenhou.
13. Considere agora o circuito da **Figura 2**.

O circuito é um *Prescaler 32/33*, bloco comum na implementação de sintetizadores de frequência para RF (detalhes sobre o funcionamento de sintetizadores no fim da apostila). A parte de cima do circuito, hachurada, é um divisor 4/5 síncrono, cujo o valor de divisão depende do sinal **MC** (*modulus counter*); a parte de baixo é um contador assíncrono modulo

8. O circuito completo divide o sinal de **clock** por 32 quando **SM** = “0” ou por 33 quando **SM** = “1”.

14. \*A partir dos dados dos blocos que compõe o divisor 4/5 estime o máximo **clock** que o circuito pode suportar (considere que são usados na implementação os blocos **DF1**, **NAND23** e **NOR23** e utilize o pior caso entre subida e descida das portas). Apresente os cálculos.

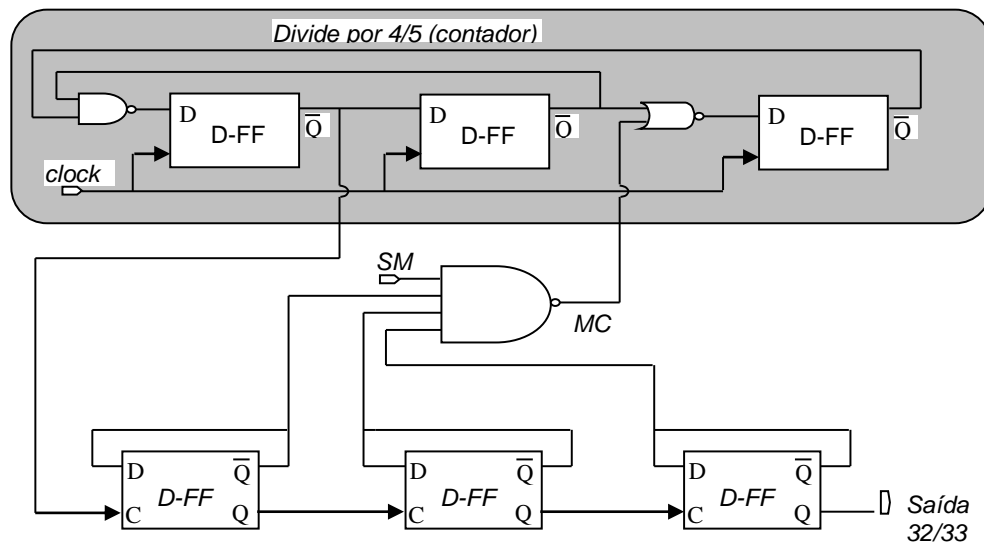


Figura 2. Prescaler 32/33.

15. Desenhe o **esquemático** do divisor 4/5 utilizando as células **DF1**, **NAND23** e **NOR23**. Como sinal de entrada deve ter o **clock** e **MC**; como sinal de saída o **sinal  $\neg Q$  do D-FF mais a esquerda**. Gere o símbolo para a célula e faça a verificação do esquemático e do símbolo. Certifique-se que não haja erros ou mesmo *warnings*.
16. \*Gere, a partir do esquemático, um arquivo **netlist** para o ELDO. Simule o circuito com parâmetros típicos e determine a máxima velocidade e consumo do circuito (consumo em **mW/GHz**). Considere
- MC = “0”, para o circuito fazer a divisão por 5;
  - $V_{DD} = 3\text{ V}$ ;
  - o sinal de **clock** com (tempo de subida) = (tempo de descida) =  $0,1 \cdot \text{Período}$ .

**Apresente as linhas de comando e sinais utilizados.**

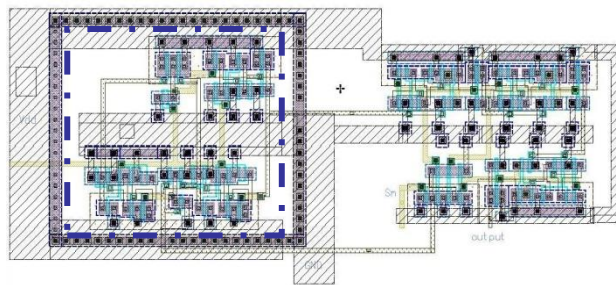
17. \*Gere agora o **layout do circuito**. Neste **layout** deve tomar cuidado com:

- a área total do circuito;
- o uso correto dos metais e poli como camadas de conexão;
- a posição e o tamanho dos **ports** de entrada e saída;
- a largura das linhas de  $V_{DD}$  e  $V_{SS}$ , não inferiores a  $1\text{ }\mu\text{m}$ .

Faça a verificação com o **DRC** (CALIBRE), passe o **LVS** e elimine todos os erros.

**Apresente a figura do *layout* no relatório.**

18. \*Faça a extração do circuito via o Calibre com a opção **C+CC**. Simule o circuito com parâmetros típicos e determine a máxima velocidade e consumo do circuito. Considere as mesmas condições do exercício 16.
19. \*Extraia agora com a opção **R+C+CC**. Simule o circuito com parâmetros típicos e determine a máxima velocidade e consumo do circuito. Considere as mesmas condições do exercício 16.
20. \***Monte uma tabela** com os resultados obtidos nos exercícios **14, 16, 18 e 19**. **Compare e comente** os resultados.
21. \*Determine a área total do circuito que desenhou.



**Figura 3.** *Layout* do circuito *Prescaler* 32/33 na tecnologia AMS 0.35 (dimensões de 82  $\mu\text{m}$  x 35  $\mu\text{m}$ ). O divisor 4/5 está indicado com um quadrado tracejado.

Observação: A Figura 3 apresenta o *layout* de um *Prescaler* 32/33 completo. Por resultados de simulação, este circuito funciona a 2,4 GHz com consumo de 2,43 mW (parâmetros típicos,  $V_{DD} = 3\text{ V}$ ). O circuito implementado, por sua vez, funcionou a 1,7 GHz com consumo de 2,31 mW. Tanto a velocidade máxima como grande parte do consumo se deve ao divisor 4/5. Assim podemos, a grosso modo, dizer que o consumo e a velocidade deste divisor é igual a do *Prescaler*.

## Sintetizador de Freqüências

Sintetizadores de Freqüências são circuitos que fornecem sinais em diferentes freqüências e são empregados na sintonia de canais dentro de uma banda de transmissão RF. Um Sintetizador pode ser implementado como um PLL (*phase locked loop*) onde é colocado um divisor programável no seu caminho de realimentação. Pela programação deste divisor pode-se controlar a freqüência do sinal gerado. Esta é a chamada arquitetura *integer N* que é mostrada na Figura 4. Os blocos que compõe tal arquitetura são: o **PD** (*phase detector*), o **LPF** (*low pass*

filter), o **VCO** (*voltage controlled oscillator*), e o **divisor M** (divisor programável chamado *Pulse Swallow*).

Para o funcionamento do circuito deve-se aplicar um sinal de referências externo, com uma frequência  $f_{Ref}$ , ao **PD**. O **VCO** e o Sintetizador não estabilizam enquanto o sinal de saída do **divisor M** não atingir frequência  $f_{Ref}$ . Por outro lado, quando isto ocorrer teremos o chamado estado *locked* e o sinal de saída terá frequência  $f_{osc} = Mf_{Ref}$ , que dependerá do valor de  $M$ . Dessa forma, desde que possamos modificar o valor de  $M$ , teremos controle sobre a frequência de saída.

O circuito **divisor M** ou *Pulse Swallow* é implementado com um **Prescaler**, um contador programável chamado **Swallow Counter** e um **contador principal**. O funcionamento do divisor inicia com a carga dos dois contadores, o **principal**, com um valor fixo  $P$ , e o **Swallow Counter**, com um valor programável  $S$ . O valor  $S$  deve ser menor ou igual a  $P$ ,  $S \leq P$ . No estado inicial o valor do sinal **SM** fica no nível lógico ALTO, o que faz com que o circuito **Prescaler** divida a frequência do sinal de saída do **VCO** por  $N+1$ . Dessa forma, a cada  $N+1$  ciclos, os valores do **contador principal** e do **Swallow Counter** serão reduzidos em um, até que o **Swallow Counter** chegue a zero. Neste momento o valor do sinal **SM** é alterado e o **Prescaler** passa a dividir por  $N$  o sinal de saída do **VCO**. Agora, a cada  $N$  ciclos, o **contador principal** será reduzido em um, até que também chegue a zero. Os contadores nesse instante são reinicializados com  $P$  e  $S$  e o processo é repetido. Com isso teremos que em um ciclo completo de operação do circuito Sintetizador o **Prescaler** divide  $S$  vezes por  $N+1$  e divide  $(P-S)$  vezes por  $N$ . O resultado final na saída do **Pulse Swallow** será a divisão do seu sinal de entrada, que é a saída do **VCO**, por  $M$ , onde

$$M = (N+1)S + N(P-S) = NP + S.$$

Quando o Sintetizador de Frequências estiver no estado *locked*, ele apresentará uma frequência de saída igual a

$$f_{osc} = M f_{Ref} = (NP + S)f_{Ref}.$$

Controlamos, portanto, a frequência de saída do **VCO** com o valor  $S$  programado no **Swallow Counter**.

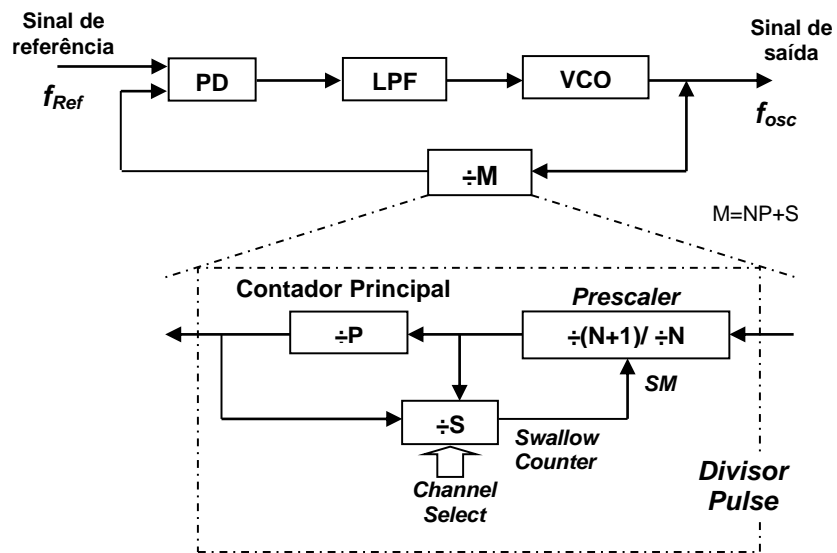


Figura 4. Arquitetura *Integer-N*.