

UNIVERSIDADE DE SÃO PAULO

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

Projeto 6

Davi Diório Mendes	7546989
Nivaldo Henrique Bondança	7143909



22 de agosto de 2014

Lista de Figuras

1	Porta lógica <i>CMOS</i> . Implementa a função $\overline{AB+C}$	p. 5
2	Atraso de subida e descida por capacitância de carga obtido pela simulação do <i>viewpoint</i>	p. 10
3	Sinais de entrada e saída utilizados na simulação do <i>viewpoint</i>	p. 13
4	Atraso de subida e descida por capacitância de carga.	p. 13
5	Sinais de entrada e de saída da porta lógica.	p. 14
6	<i>Layout</i> da porta lógica apresentada na figura 1.	p. 14
7	<i>Layout</i> do inversor.	p. 15
8	Atraso alto-alto e baixo-baixo por capacitância de carga para a porta lógica não inversora.	p. 16
9	Sinais de entrada e saída utilizados na simulação da porta lógica não inversora.	p. 19
10	<i>Layout</i> da porta lógica não inversora.	p. 20
11	Atraso alto-alto e baixo-baixo para a porta lógica não inversora obtidos através de simulação do circuito com capacitâncias parasitas.	p. 21
12	Sinais de entrada utilizados na simulação e saída obtida para a porta lógica não inversora	p. 21
13	Histograma de ocorrências de atraso alto-alto obtido pela simulação de Monte Carlo sobre a porta lógica não inversora.	p. 23
14	Histograma de ocorrências de atraso baixo-baixo obtido pela simulação de Monte Carlo sobre a porta lógica não inversora.	p. 23
15	Sinal de entrada <i>C</i> utilizado durante a simulação de Monte Carlo sobre a porta lógica não inversora e sinal de saída <i>OUT</i> no melhor caso, caso típico e pior caso obtidos pelo simulação.	p. 25

Lista de Tabelas

- 1 Coeficientes angular e linear para as curvas de atraso de descida e subida por
capacitância de carga. p. 12
- 2 Atrasos obtidos para diferentes capacitâncias de carga através da simulação
da porta lógica não inversora. p. 19
- 3 Coeficientes angular e linear para as curvas de atraso alto-alto e baixo-baixo,
obtidas na simulação da porta lógica não inversora com circuitos parasitas. . . p. 22

Códigos Fontes

1	<i>Netlist</i> gerado a partir do <i>viewpoint</i>	p. 7
2	Comandos utilizados para a medição de atraso de subida e descida por capacidade de carga.	p. 9
3	Arquivo utilizado para a simulação da porta lógica não inversora	p. 16

Nesta experiência é dada continuidade ao que foi ensinado no semestre anterior. Será visto aqui como se extrai o *netlist* para simulação já a partir do esquemático, permitindo a verificação da sua funcionalidade. Também será visto como fazer a comparação entre *layout* e esquemático. Por fim é introduzida a simulação do tipo Monte Carlo que possibilita analisar o comportamento dos circuitos com as variações dos parâmetros dos transistores.

1. *Considere a porta lógica *CMOS* estática que implementa a função lógica $\overline{AB + C}$. Tendo o transistor *NMOS* de menor dimensão um $W = 2\mu m$ e $L = 0,35\mu m$, determine as dimensões de todos transistores de forma que:

- i (atraso de propagação na descida com $ABC = "110"$) = (atraso de propagação na descida com $ABC = "001"$) = (pior atraso de propagação na subida);
- ii Todos transistores *PMOS* tenham as mesmas dimensões.

(deixe indicado os valores usados)

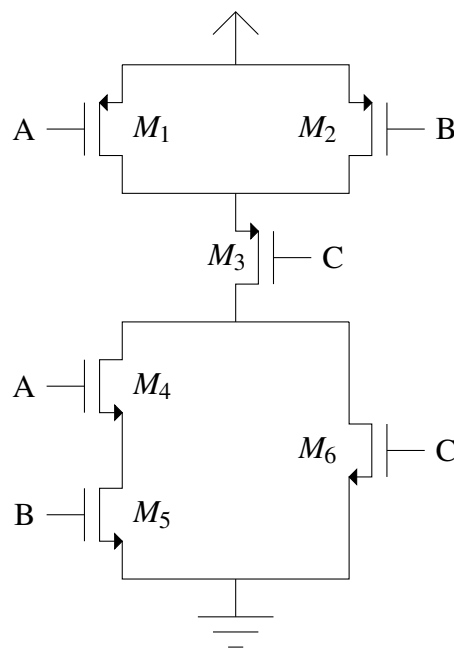


Figura 1: Porta lógica *CMOS*. Implementa a função $\overline{AB + C}$.

Seja o tempo de propagação dado pela seguinte relação:

$$t_p = \frac{1,6 \cdot C_L}{\mu \cdot \left(\frac{W}{L}\right) \cdot C_{ox} \cdot V_{DD}} \quad (1)$$

No caso 110, teremos os transistores M_4 e M_5 abertos, realizando a descida do sinal. Para o cálculo tempo de propagação na descida, teremos um transistor equivalente com as seguintes

dimensões:

$$W_{eq} = W_{M_4, M_5} \quad (2)$$

$$L_{eq} = 2 \cdot L \quad (3)$$

Por outro lado, no caso 001, somente o transistor M_6 abre. Desta forma as dimensões equivalentes são as mesmas dimensões do transistor M_6 :

$$W_{eq} = W_{M_6} \quad (4)$$

$$L_{eq} = L \quad (5)$$

Desta forma temos:

$$t_{pHL(110)} = \frac{1,6 \cdot C_L}{\mu_N \cdot \left(\frac{W_{M_4, M_5}}{2 \cdot L} \right) \cdot C_{ox} \cdot V_{DD}} \quad (6)$$

$$t_{pHL(001)} = \frac{1,6 \cdot C_L}{\mu_N \cdot \left(\frac{W_{M_6}}{L} \right) \cdot C_{ox} \cdot V_{DD}} \quad (7)$$

e, ao igualarmos (6) e (7), eliminaremos todas as variáveis não abordadas até o momento, obtendo a relação:

$$\frac{W_{M_4, M_5}}{2 \cdot L} = \frac{W_{M_6}}{L} \quad (8)$$

$$\Rightarrow W_{M_4, M_5} = 2 \cdot W_{M_6} \quad (9)$$

Como $W_{M_6} < W_{M_4, M_5} \Rightarrow W_{M_6} = 2\mu m$. Logo:

$$W_{M_4, M_5} = 2 \cdot 2\mu m = 4\mu m \quad (10)$$

$$\Rightarrow W_{M_4} = W_{M_5} = 4\mu m \quad (11)$$

Para o análise dos transistores *PMOS*, tomamos como os piores casos $ABC = 101$ ou $ABC = 011$. Os transistores equivalente — utilizado no cálculo do tempo de propagação — de ambos os casos possuem as mesmas dimensões, mostradas a seguir:

$$W_{eq} = W_P \quad (12)$$

$$L_{eq} = 2 \cdot L \quad (13)$$

Desta forma, podemos modelar o pior atraso de propagação na subida como:

$$t_{pLH(101|011)} = \frac{1,6 \cdot C_L}{\mu_P \cdot \left(\frac{W_P}{2 \cdot L}\right) \cdot C_{ox} \cdot V_{DD}} \quad (14)$$

Igualando as equações (7) e (14), analogamente ao que foi feito para os transistores *NMOS*, podemos calcular W_P :

$$W_P = 2 \cdot \frac{\mu_N}{\mu_P} \cdot W_{M_6} \quad (15)$$

$$\Rightarrow W_P \approx 12,90 \mu m \quad (16)$$

para $\mu_N = 476 \text{ cm}^2/\text{Vs}$ e $\mu_P = 148 \text{ cm}^2/\text{Vs}$.

2. Faça o circuito esquemático da porta *CMOS* e gere seu símbolo. Faça todas as verificações necessárias no esquemático e no símbolo não deixando nenhum erro ou *warning*. Não esqueça de ligar o *bulk* dos transistores.

3. Crie para o esquemático um *designview point* para poder gerar arquivos para simulação e comparar com *layout*. Para isso execute o comando **Hit-Kit Utilites – Create Viewpoint** na parte superior da janela do *Design Architecture*. No menu que aparece coloque o nome de seu arquivo esquemático no *Design Path*. Deixe a opção *device*.

4. Agora gere um arquivo *netlist* para o *ELDO*. Para isso entre no *Simulation mode* (coluna esquerda da janela do *Design Architecture*, último comando). No menu selecione o *vpt_c35b4_device*, que foi gerado no item anterior, e dê *OK*.

Obs.: A partir da nova janela podemos retornar à anterior executando novamente o último comando da coluna à esquerda.

5. *Gere o *netlist* executando o comando apropriado na coluna à esquerda. Com outro comando nessa coluna, o *ASCII Results*, verifique os resultados na opção *view netlist*. Acrescente o *netlist* ao relatório.

O *netlist* gerado automaticamente a partir do *viewpoint* está descrito no código fonte 1.

Código Fonte 1: *Netlist* gerado a partir do *viewpoint*

```

1 *
2 * .CONNECT statements
3 *
4 .CONNECT GROUND 0
5
6
7 * ELDO netlist generated with ICnet by 'cad' on Wed Aug 13 2014 at 16:18:27
8
9 *
```

```

10 * Globals.
11 *
12 .global VDD VSS
13
14 *
15 * MAIN CELL: Component pathname : $lab6/default.group/logic.views/ex2
16 *
17     M_PMOS_C S C N$209 VDD MODP w=1.290000e-05 l=3.500000e-07 as=1.096500e-11
18 + ad=1.096500e-11 ps=1.460000e-05 pd=1.460000e-05 nrs=3.294574e-02 nrd=3.294574e-02
19     M_PMOS_B N$209 B VDD VDD MODP w=1.290000e-05 l=3.500000e-07 as=1.096500e-11
20 + ad=1.096500e-11 ps=1.460000e-05 pd=1.460000e-05 nrs=3.294574e-02 nrd=3.294574e-02
21     M_PMOS_A N$209 A VDD VDD MODP w=1.290000e-05 l=3.500000e-07 as=1.096500e-11
22 + ad=1.096500e-11 ps=1.460000e-05 pd=1.460000e-05 nrs=3.294574e-02 nrd=3.294574e-02
23     M_NMOS_C S C VSS VSS MODN w=2.000000e-06 l=3.500000e-07 as=1.700000e-12
24 + ad=1.700000e-12 ps=3.700000e-06 pd=3.700000e-06 nrs=2.125000e-01 nrd=2.125000e-01
25     M_NMOS_B N$207 B VSS VSS MODN w=4.000000e-06 l=3.500000e-07 as=3.400000e-12
26 + ad=3.400000e-12 ps=5.700000e-06 pd=5.700000e-06 nrs=1.062500e-01 nrd=1.062500e-01
27     M_NMOS_A S A N$207 VSS MODN w=4.000000e-06 l=3.500000e-07 as=3.400000e-12
28 + ad=3.400000e-12 ps=5.700000e-06 pd=5.700000e-06 nrs=1.062500e-01 nrd=1.062500e-01
29 *
30 .end

```

6. *Como são calculadas as áreas e perímetros de dreno e *source* no circuito extraído pelo esquemático (relação usada)?

Conforme o manual *ELDO equations*, temos que:

$$AS = W \cdot XA \quad (17)$$

$$AD = W \cdot XA \quad (18)$$

$$PS = W + 2 \cdot XA \quad (19)$$

$$PD = W + 2 \cdot XA, \quad (20)$$

onde *AS* é a área de *source*, *AD* é a área de dreno, *PS* é o perímetro de *source*, *PD* é o perímetro de dreno e *XA* é o comprimento da difusão. Neste caso o comprimento da difusão é $XA = 0.85\mu m$.

7. Observe que no *netlist* não aparecem as capacitâncias parasitas que são geradas quando é feita a extração do *layout*. Prepare um arquivo para simulação e, com os parâmetros típicos, $V_{DD} = 3V$, determine o atraso de propagação na subida *versus* capacitância de carga (entrada com onda quadrada com *rise/fall time* pequenos (1% do período, por exemplo)). Escolha os sinais de entrada de forma a obter a pior situação, ao menos cinco valores para a capacitância de carga, e escolha o período dos sinais de forma a obter resultados corretos. Faça o mesmo para o atraso de propagação na descida *versus* capacitância de carga.

8. *Apresente os gráficos da questão anterior e copie os comandos de medida e sinais de

entrada que usou no *ELDO*.

Os atrasos medidos na questão anterior podem ser vistos na imagem 2. Os comandos utilizados em suas medidas estão disponíveis no Código Fonte 2. As formas de onda dos sinais de entrada *A*, *B* e *C* e a saída *S* obtida pelas simulação estão na imagem 3.

Quando mantemos o sinal *A* em nível lógico alto e o sinal *C* em nível lógico baixo, o circuito lógico da questão 1 funciona como uma porta inversora com entrada em *B* e saída *S*. Pode-se observar na figura 3 que, como esperado, a saída é a negação de *C*.

Código Fonte 2: Comandos utilizados para a medição de atraso de subida e descida por capacitância de carga.

```

1 *
2 * .CONNECT statements
3 *
4 .CONNECT GROUND 0
5
6 * ELDO netlist generated with ICnet by 'cad' on Wed Aug 13 2014 at 14:50:34
7
8 *
9 * Globals.
10 *
11 .global VDD VSS
12
13 *
14 * MAIN CELL: Component pathname : $lab6/default.group/logic.views/ex2
15 *
16     M_PMOS_C S C N$209 VDD MODP w=1.290000e-05 l=3.500000e-07 as=1.096500e-11
17 + ad=1.096500e-11 ps=1.460000e-05 pd=1.460000e-05 nrs=3.294574e-02 nrd=3.294574e-02
18     M_PMOS_B N$209 B VDD VDD MODP w=1.290000e-05 l=3.500000e-07 as=1.096500e-11
19 + ad=1.096500e-11 ps=1.460000e-05 pd=1.460000e-05 nrs=3.294574e-02 nrd=3.294574e-02
20     M_PMOS_A N$209 A VDD VDD MODP w=1.290000e-05 l=3.500000e-07 as=1.096500e-11
21 + ad=1.096500e-11 ps=1.460000e-05 pd=1.460000e-05 nrs=3.294574e-02 nrd=3.294574e-02
22     M_NMOS_C S C VSS VSS MODN w=2.000000e-06 l=3.500000e-07 as=1.700000e-12
23 + ad=1.700000e-12 ps=3.700000e-06 pd=3.700000e-06 nrs=2.125000e-01 nrd=2.125000e-01
24     M_NMOS_B N$207 B VSS VSS MODN w=4.000000e-06 l=3.500000e-07 as=3.400000e-12
25 + ad=3.400000e-12 ps=5.700000e-06 pd=5.700000e-06 nrs=1.062500e-01 nrd=1.062500e-01
26     M_NMOS_A S A N$207 VSS MODN w=4.000000e-06 l=3.500000e-07 as=3.400000e-12
27 + ad=3.400000e-12 ps=5.700000e-06 pd=5.700000e-06 nrs=1.062500e-01 nrd=1.062500e-01
28 *
29 .PARAM cload=100f
30 C1 S VSS cload
31
32 *** pulso ***
33 .PARAM tvdd=3V
34 .PARAM frequencia=230MegHz
35 .PARAM periodo='1/frequencia'
36 .PARAM delay=0s
37 .PARAM tSubida='0.01*periodo'
38 .PARAM tDescida='0.01*periodo'
39 .PARAM larguraPulso='0.49*periodo'

```

```

40 .PARAM valto='tvdd'
41 .PARAM vbaixo=0V
42
43 *** varredura ***
44 .PARAM vStep=50f
45 .PARAM vInit=100f
46 .PARAM vEnd=300f
47
48 Vcc VDD VSS tvdd
49 Vgnd VSS 0 0
50
51 Va A VSS tvdd
52 Vb B VSS PULSE (vbaixo valto delay tSubida tDescida larguraPulso periodo)
53 Vc C VSS 0
54
55 .TRAN 1ns '20*periodo' '5*periodo' 0.1ns SWEEP cload INCR vStep vInit vEnd
56 .PROBE TRANS V(A) V(B) V(C) V(S)
57
58 *** Delay ***
59 .MEAS TRAN DEL_H2L TRIG V(B) VAL=1.5V RISE=5 TARG V(S) VAL=1.5 FALL=5
60 .MEAS TRAN DEL_L2H TRIG V(B) VAL=1.5V FALL=5 TARG V(S) VAL=1.5 RISE=5
61
62 .INC '~/para_prova/tipico.mod'
63
64 .end

```

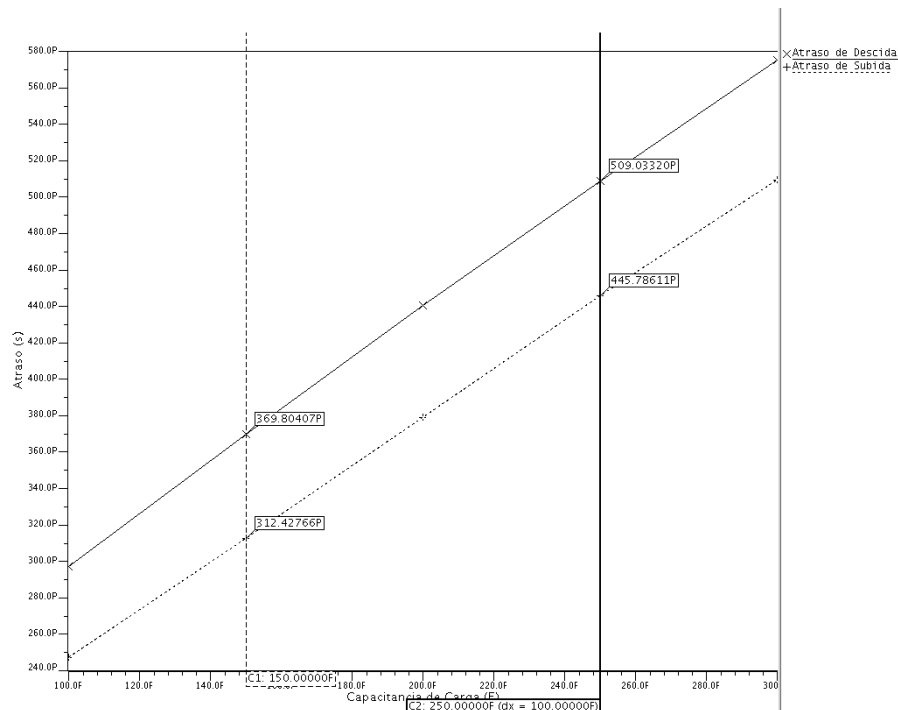


Figura 2: Atraso de subida e descida por capacitância de carga obtido pela simulação do *view-point*.

9. Faça agora o *layout* da porta (utilize o *designview point* gerado para geração). No *layout* deve-se tomar cuidado com:

- área total do circuito;
- o uso correto dos metais e poli como camadas de conexão;
- a posição dos *ports* de entrada e saída.

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

Obs.: Veja as opções do **Route – ARoute Commands – Setup – Display** para melhor fazer o roteamento manual.

10. Utilize o comando **Connectivity – port – Add to Port** para ampliar as áreas dos *ports*. Para isso selecione o *shape* que deseja acrescentar a um *port* e então execute o comando.

11. *Como se pode acrescentar aos *ports* V_{DD} e V_{SS} as regiões de *source* dos transistores sem transformarmos os transistores em *flatten*?

Para adicionar o *source* ao V_{DD} ou ao V_{SS} deve-se adicionar um novo *shape* de metal 1, sobre o metal do *source*. Este *shape* pode ser adicionado aos *ports* V_{DD} ou V_{SS} . Isso funciona pois dois *shapes* sobrepostos de mesma *layer* se comportam como um único *shape*. Logo, conectar algo em um *MET1* sobre o *source* de um transistor, é o mesmo que conectar diretamente ao *source* em questão.

12. Uma vez acrescentadas aos *ports* todas as regiões desejadas, faça nova verificação com o *DRC*.

13. Vamos agora fazer a última verificação do circuito: comparação entre o *layout* e o esquemático que o gerou (*Layout vs. Schematic* ou simplesmente *LVS*). Para isso, dentro do *ICStation* feche o circuito lógico associado ao *layout* (**File – Logic – close** , menu superior). Execute então os comandos *IcTrace(M)* e *LVS* (menu à direita). No menu que aparece complete o *source name* (nome do *netlist* para comparar que esta no ... `logic.view/name/.../vpt_c35b4_device`) e na opção *Abort on Supply Error*, deixe *NO*. Verifique a função das outras opções que estão disponíveis.

14. Ao dar o *OK*, é feita a comparação entre os *netlists* extraídos do *layout* e do esquemático. Para ver se há ou não erros execute o comando ***IcTrace(M) – LVS – Report – LVS*** . Verifique também as outras opções no menu de *Report*. Caso não houver erros aparecerá a "*carinha feliz*" se houver, serão fornecidas informações sobre os erros.

15. Para determinar onde estão os erros pode ser usado o comando ***IcTrace(M) – discrepancies***. A opção *first* aí mostra o primeiro erro; a opção *next*, o próximo. Caso tenha tido algum erro tente achá-lo com esses comandos; se não teve erros, modifique o *layout* (apague alguma conexão) para poder praticar (caso não pratique na aula poderá ter surpresas na prova).

Obs.: o comando *discreps* deixa selecionada uma ou mais regiões do circuito. Essas regiões selecionadas são deselecionadas apenas pelo comando ***IcTrace(M) – unshow – all***. Algo similar acontece com o *DRC*. Obs.: O comando ***IcTrace(M) – netlist*** também serve, como o *PEX (CALIBRE)*, para gerar *netlists* para o *ELDO*.

16. *Uma vez feitas as verificações com *DRC* e *LVS*, caso não tenha sido encontrado nenhum erro, o *layout* estará pronto para uso. Agora, extraia o circuito de simulação a partir do *layout* (opção C+CC) e repita as simulações feitas no item 7. Apresente os gráficos com resultados (gere uma figura do *layout* e inclua no trabalho).

O gráfico com os atrasos de subida e descida estão na figura 4 e os sinais de entrada utilizados e saída obtida pela simulação estão na figura 5. O *layout* pode ser visto na figura 6.

17. *Para as curvas atraso de propagação na subida e descida *versus* carga, geradas a partir do *layout*, calcule as inclinações e o pontos de cruzamento com o eixo Y (eixo de tempo).

A partir dos dados da figura 4, obtemos:

$$\alpha_D = \frac{\Delta y}{\Delta x} = 1386,8 \quad (21)$$

$$\alpha_S = \frac{\Delta y}{\Delta x} = 1335,8 \quad (22)$$

onde α_D e α_S são os coeficientes angulares dos atrasos de descida e subida, respectivamente.

Seja a o atraso de descida dado por:

$$y_D = 1386,8 \cdot x_D + y_{0D} \quad (23)$$

Substituindo um ponto na fórmula, podemos obter y_0 :

$$y_{0D} = 497,42 \cdot 10^{-12} - 1386,8 \cdot 250 \cdot 10^{-15} = 0,15 \cdot 10^{-9} \quad (24)$$

Analogamente, calcula-se o y_0 de subida:

$$y_{0S} = 438,8 \cdot 10^{-12} - 1335,8 \cdot 250 \cdot 10^{-15} = 0,10 \cdot 10^{-9} \quad (25)$$

Os resultados destes cálculos estão condensados na tabela 1.

Tabela 1: Coeficientes angular e linear para as curvas de atraso de descida e subida por capacidade de carga.

	Atraso de Descida	Atraso de Subida
Coeficiente Angular	1386,8	1335,8
Coeficiente Linear	$0,15 \cdot 10^{-9}$	$0,10 \cdot 10^{-9}$

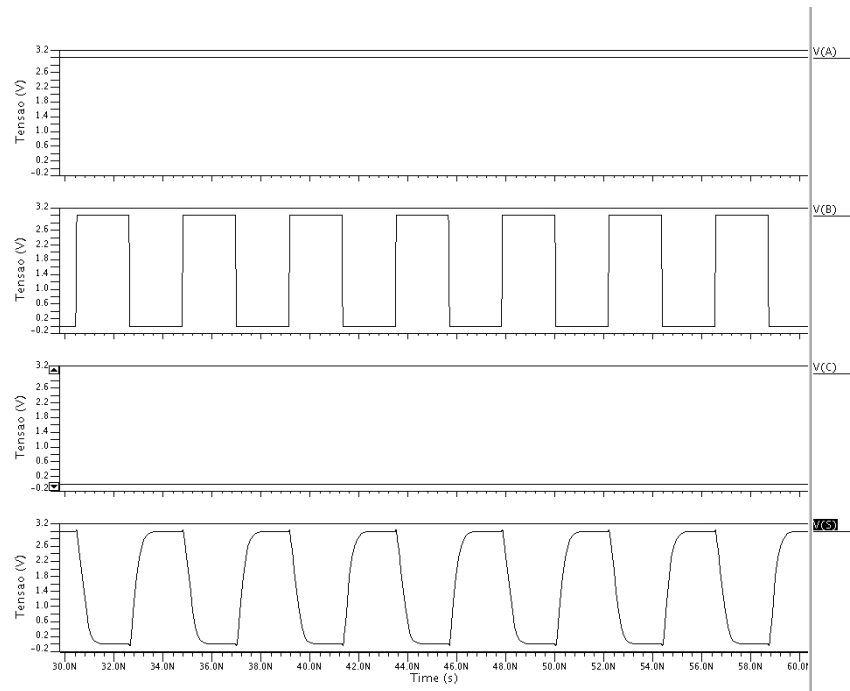


Figura 3: Sinais de entrada e saída utilizados na simulação do *viewpoint*.

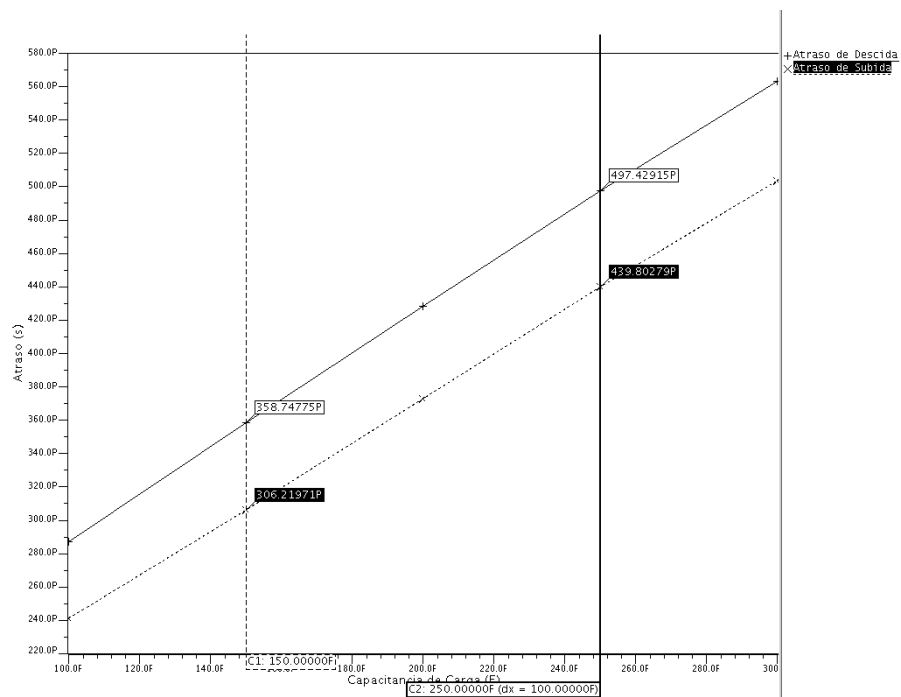


Figura 4: Atraso de subida e descida por capacitância de carga.

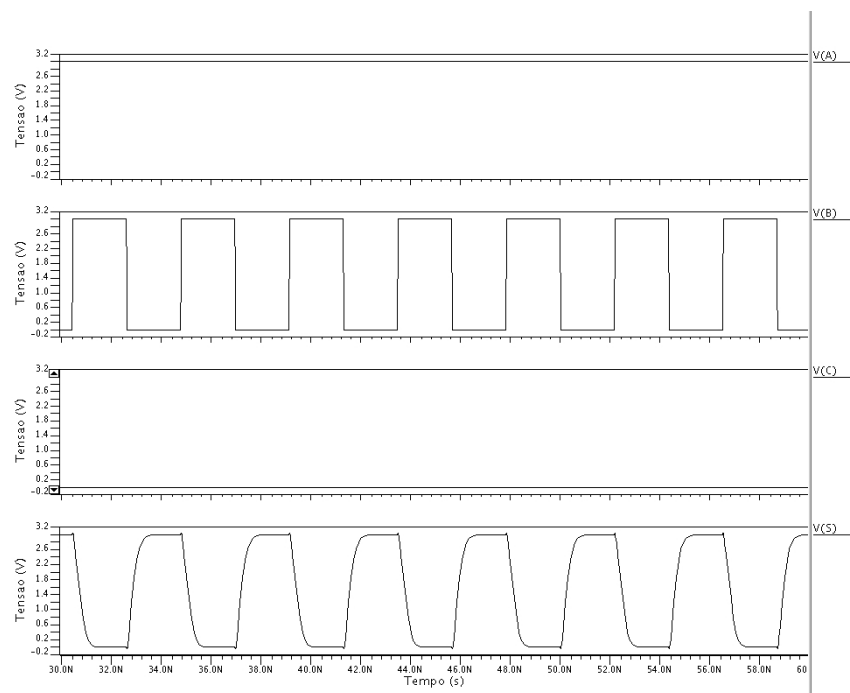


Figura 5: Sinais de entrada e de saída da porta lógica.

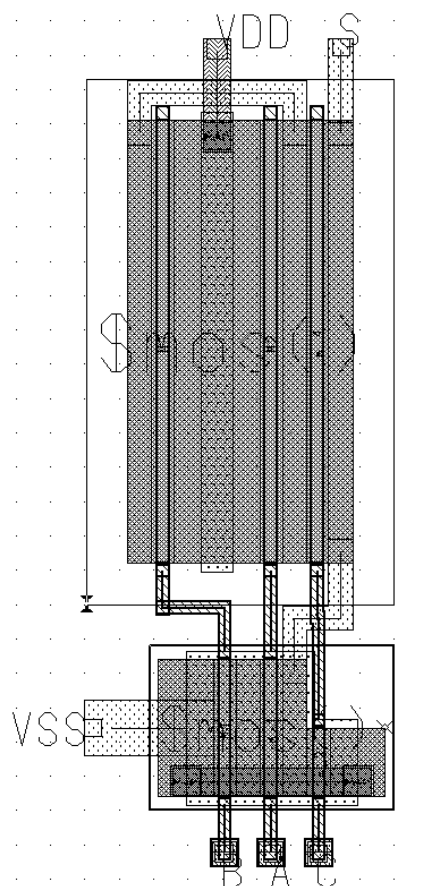


Figura 6: Layout da porta lógica apresentada na figura 1.

18. *Comente as diferenças entre os resultados encontrados nas questões 8 e 16/17? Dê as razões para elas.

19. *Faça um inversor com $W_N = 2\mu m$ e $L_N = 0,35\mu m$. Faça o esquemático, símbolo e *layout*. Passe as verificações no esquemático e símbolo. O *layout* deve ser feito com cuidado para ter área pequena, utilização correta de metais/poli e *ports* de tamanho conveniente. Passe o *DRC* no *layout* e faça o *LVS* deixando a célula pronta para uso. Acrescente ao relatório o *layout* feito.

O *layout* do inversor está apresentado na imagem 7.

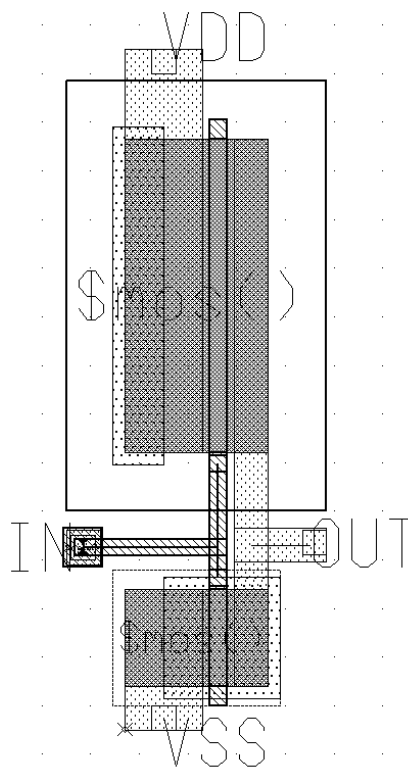


Figura 7: *Layout* do inversor.

20. A partir das duas células desenhadas, monte o esquemático de uma nova célula que executa a função lógica $(ab + c)$. Gere o seu símbolo e faça todas as verificações necessárias.

Obs.: antes de realizar o item 20, deve-se acrescentar aos símbolos anteriores a propriedade *phy_comp* com a posição do *layout* de cada célula.

21. Gere a partir do esquemático o arquivo para simulação com o *ELDO* (acrescente o *netlist* ao relatório). Simule com os parâmetros típicos, $V_{DD} = 3V$, e determine o atraso de propagação na subida *versus* capacitância de carga (entrada com onda quadrada com *rise/fall time* igual a 1%). Escolha os sinais de entrada de forma a obter a pior situação, ao menos cinco valores para a capacitância de carga e escolha o período dos sinais de forma a obter resultados

corretos. Faça o mesmo para o atraso de propagação na descida *versus* capacitância de carga.

22. *Desenhe os gráficos da questão anterior e copie os comandos de medida e sinais de entrada que usou no *ELDO*.

Os atrasos estão descritos na figura 8. Os sinais de entrada utilizados na simulação e a saída obtida podem ser vistos na figura 9. O código utilizado para a simulação está descrito no código fonte 3

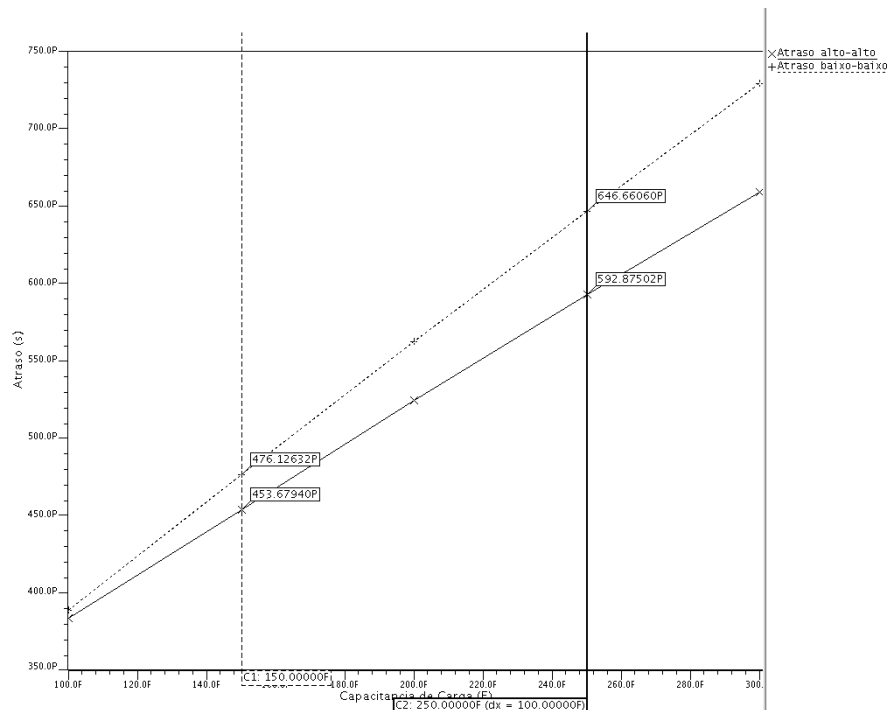


Figura 8: Atraso alto-alto e baixo-baixo por capacitância de carga para a porta lógica não inversora.

Código Fonte 3: Arquivo utilizado para a simulação da porta lógica não inversora

```

1 *
2 * .CONNECT statements
3 *
4 .CONNECT GROUND 0
5
6 * ELDO netlist generated with ICnet by 'cad' on Mon Aug 18 2014 at 11:31:24
7
8 *
9 * Globals.
10 *
11 .global VDD VSS
12
13 *
14 * Component pathname : $lab_seis/default.group/logic.views/inversor
15 *
16 .subckt INVERSOR OUT IN
17

```



```

18     M_2 OUT IN VSS VSS MODN w=2.000000e-06 l=3.500000e-07 as=1.700000e-12
19 +   ad=1.700000e-12 ps=3.700000e-06 pd=3.700000e-06 nrs=2.125000e-01 nrd=2.125000e-01
20     M_1 OUT IN VDD VDD MODP w=6.450000e-06 l=3.500000e-07 as=5.482500e-12
21 +   ad=5.482500e-12 ps=8.150000e-06 pd=8.150000e-06 nrs=6.589147e-02 nrd=6.589147e-02
22 .ends INVERSOR
23
24 *
25 * Component pathname : $lab_seis/default.group/logic.views/exdois
26 *
27 .subckt EXDOIS  S A B C
28
29     M3 N$209 B VDD VDD MODP w=1.290000e-05 l=3.500000e-07 as=1.096500e-11
30 +   ad=1.096500e-11 ps=1.460000e-05 pd=1.460000e-05 nrs=3.294574e-02 nrd=3.294574e-02
31     M2 N$209 A VDD VDD MODP w=1.290000e-05 l=3.500000e-07 as=1.096500e-11
32 +   ad=1.096500e-11 ps=1.460000e-05 pd=1.460000e-05 nrs=3.294574e-02 nrd=3.294574e-02
33     M_4 S C N$209 VDD MODP w=1.290000e-05 l=3.500000e-07 as=1.096500e-11
34 +   ad=1.096500e-11 ps=1.460000e-05 pd=1.460000e-05 nrs=3.294574e-02 nrd=3.294574e-02
35     M_3 S C VSS VSS MODN w=2.000000e-06 l=3.500000e-07 as=1.700000e-12
36 +   ad=1.700000e-12 ps=3.700000e-06 pd=3.700000e-06 nrs=2.125000e-01 nrd=2.125000e-01
37     M1 N$207 B VSS VSS MODN w=4.000000e-06 l=3.500000e-07 as=3.400000e-12
38 +   ad=3.400000e-12 ps=5.700000e-06 pd=5.700000e-06 nrs=1.062500e-01 nrd=1.062500e-01
39     M_1 S A N$207 VSS MODN w=4.000000e-06 l=3.500000e-07 as=3.400000e-12
40 +   ad=3.400000e-12 ps=5.700000e-06 pd=5.700000e-06 nrs=1.062500e-01 nrd=1.062500e-01
41 .ends EXDOIS
42
43 *
44 * MAIN CELL: Component pathname : $lab_seis/default.group/logic.views/exvinte
45 *
46
47 *** varredura ***
48 .PARAM cload=100f
49 .PARAM vStep=50f
50 .PARAM vInit=100f
51 .PARAM vEnd=300f
52
53     X_INVERSOR2 S N$5 INVERSOR
54     X_EXDOIS2 N$5 A B C EXDOIS
55     Cl S VSS cload
56
57 *** pulso ***
58 .PARAM tvdd=3V
59 .PARAM frequencia=230MegHz
60 .PARAM periodo='1/frequencia'
61 .PARAM delay=0s
62 .PARAM tSubida='0.01*periodo'
63 .PARAM tDescida='0.01*periodo'
64 .PARAM larguraPulso='0.49*periodo'
65 .PARAM valto='tvdd'
66 .PARAM vbaixo=0V
67
68 VDD VDD 0 tvdd
69 VSS VSS 0 0
70 VIN VIN 0 PULSE (vbaixo valto delay tSubida tDescida larguraPulso periodo)
71

```

```

72 .CONNECT A VDD
73 .CONNECT B VIN
74 .CONNECT C VSS
75
76 *** ANALISE TRANSIENTE ***
77 .PARAM tprint=1ns
78 .PARAM hmax='0.1*tprint'
79
80 .TRAN 1ns '20*periodo' '5*periodo' 0.1ns SWEEP cload INCR vStep vInit vEnd
81
82 *** Delay ***
83 .MEAS TRAN DEL_H2H TRIG V(B) VAL=1.5V RISE=5 TARG V(S) VAL=1.5 RISE=5
84 .MEAS TRAN DEL_L2L TRIG V(B) VAL=1.5V FALL=5 TARG V(S) VAL=1.5 FALL=5
85
86 .PROBE TRANS V(A) V(B) V(C) V(S)
87 *
88 .include "~/para_prova/tipico.mod"
89
90 .end

```

23. Faça o *layout* final da célula. Utilize nas linhas de metal que ligam o V_{DD} e o V_{SS} largura sempre superior ou igual a $1,0\mu m$. Para isso veja e utilize o comando **Route – ARoutre NEt Classe – Edit** que permite especificar as características de conexão de qualquer sinal. Utilize a opção *New/Edit* para fornecer as características desejadas e a opção *Assign* para associá-las a um sinal (e apenas um). Coloque os sinais de V_{DD} e o V_{SS} com metais de $1,0\mu m$.

Obs.: Veja que quando se esta executando o comando **Route – Iroute Commands – Run**, a tecla *w* pode ser utilizada para alterar a largura da linha desenhada.

24. *Termine *layout* da célula, passe o *DRC* e faça o *LVS*. Gere uma figura do *layout* mostrando todos os níveis e inclua no trabalho.

O *layout* da porta lógica não inversora está na figura 10.

25. *Agora extraia o circuito de simulação a partir do *layout* (opção C+CC) e repita as simulações feitas no item 22. Apresente gráficos e tabelas com os resultados.

O gráfico de atraso por capacitância de carga está representado na figura 11. Os sinais de entrada utilizados na simulação e a saída da porta lógica não inversora estão na figura 12. Na tabela 2 podemos observar os valores dos atrasos para cada capacitância de carga utilizada na simulação.

Vale observar que, com a utilização do inversor na saída da porta lógica, ao mantermos o sinal *A* em alto e *C* em baixo obtemos uma porta repetidora com entrada *C* e saída *S*. Pode-se ver na figura 12 que realmente o sinal de saída acompanha os níveis de tensão do sinal de entrada.

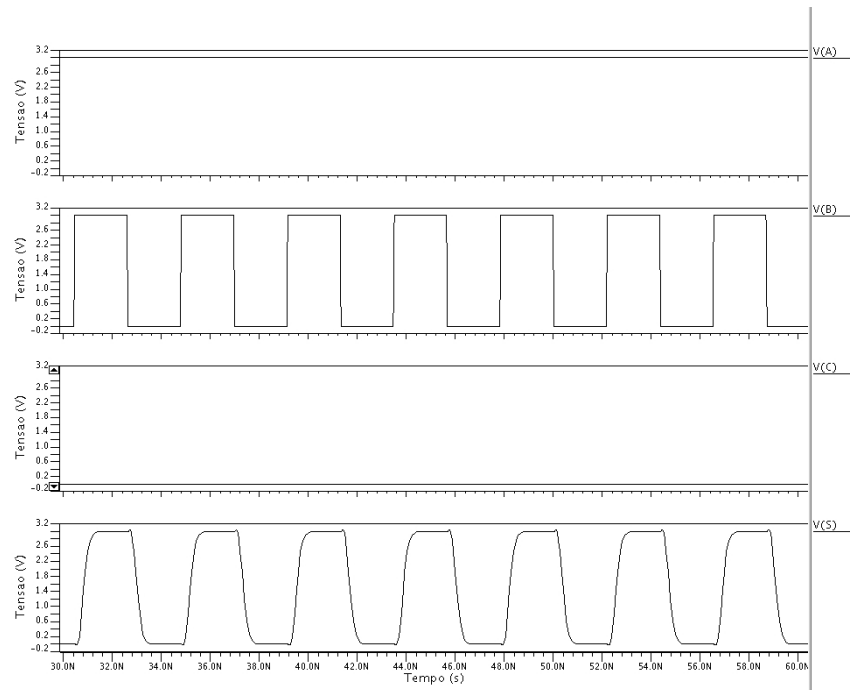


Figura 9: Sinais de entrada e saída utilizados na simulação da porta lógica não inversora.

Tabela 2: Atrasos obtidos para diferentes capacitâncias de carga através da simulação da porta lógica não inversora.

Capacitância de Carga (fF)	Atraso alto-alto (ps)	Atraso baixo-baixo (ps)
100	3.7184E-10	3.8480E-10
150	4.4253E-10	4.7144E-10
200	5.1329E-10	5.5719E-10
250	5.8176E-10	6.4243E-10
300	6.4756E-10	7.2473E-10

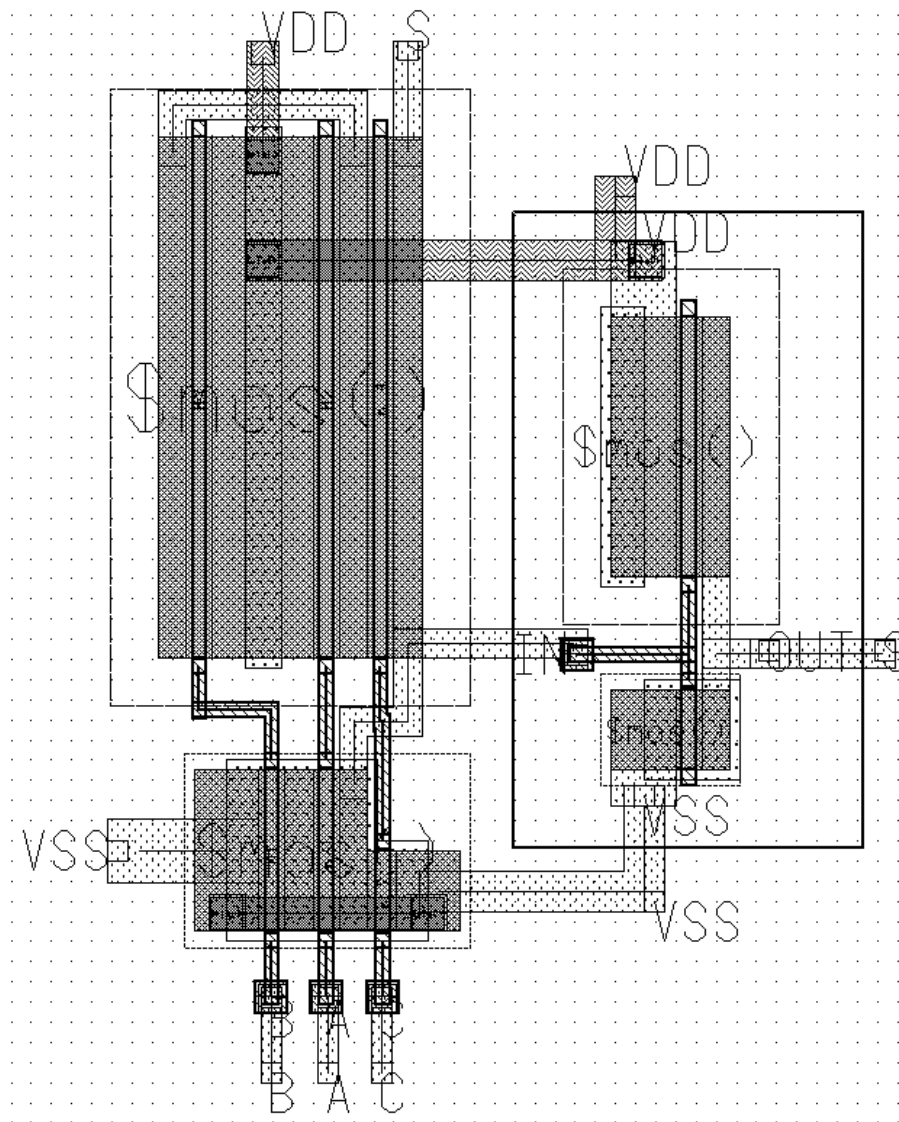


Figura 10: *Layout* da porta lógica não inversora.

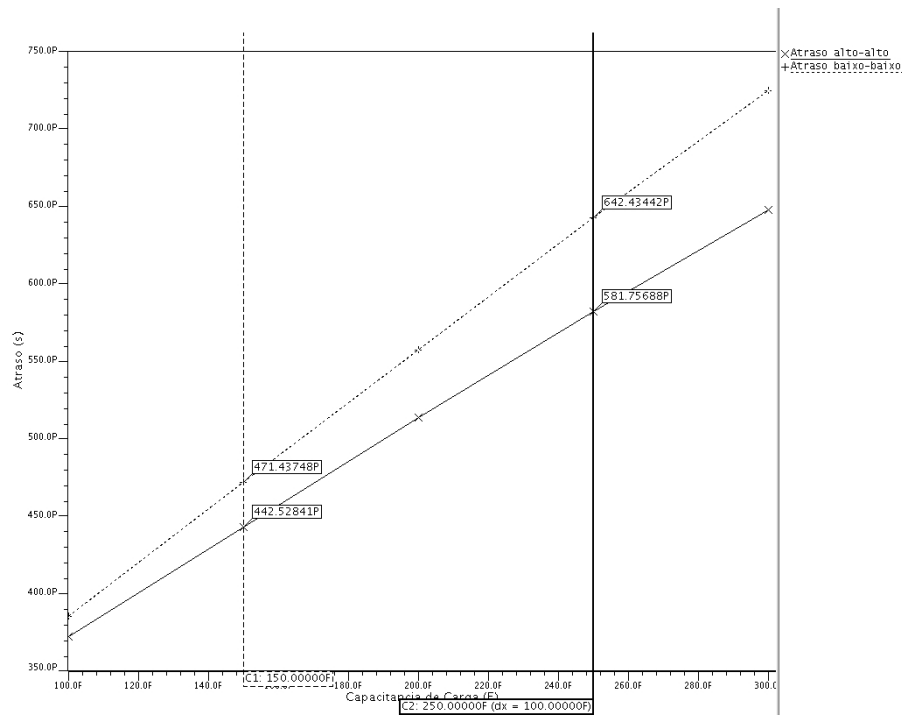


Figura 11: Atraso alto-alto e baixo-baixo para a porta lógica não inversora obtidos através de simulação do circuito com capacitâncias parasitas.

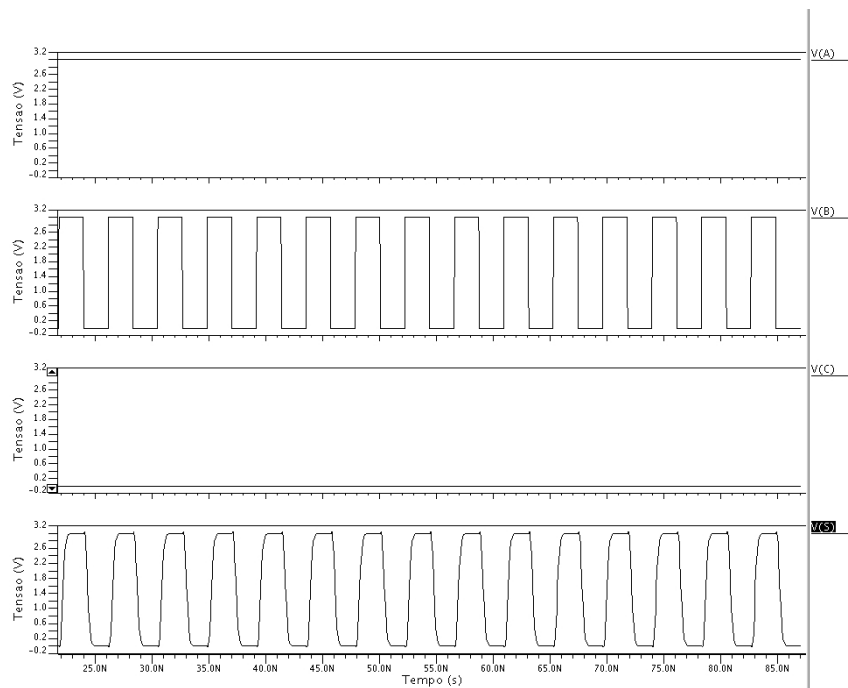


Figura 12: Sinais de entrada utilizados na simulação e saída obtida para a porta lógica não inversora

26. *Para as curvas tempo de propagação na subida e descida geradas a partir do *layout*, calcule as inclinações e os pontos de cruzamento com o eixo Y (eixo de tempo).

*** Calcular!!! A partir das equações xxxx calculou-se os coeficientes angular e linear. Estes estão apresentados na tabela 3.

Tabela 3: Coeficientes angular e linear para as curvas de atraso alto-alto e baixo-baixo, obtidas na simulação da porta lógica não inversora com circuitos parasitas.

Atraso	Coef. Angular	Coef. Linear
Alto-alto	xxx	xxx
Baixo-baixo	yyy	yyy

27. *Gere novamente os tempos de propagação na subida e descida utilizando agora os comandos (faça os ajustes necessários para seu circuito)

```

1 Va a 0 3V
2 Vb b 0 0
3 Vc c 0 pulse (0 3 0 1p 1p 2n 4n)
4 .tran 1n 40n 0n 1p
5 .meas tran delayF trig v(c) val=1.5 fall=6 targ v(out) val=1.5 fall=6
6 .meas tran delayR trig v(c) val=1.5 rise=6 targ v(out) val=1.5 rise=6
7 Cl out 0 30fF

```

Utilizando os comandos listados, obtemos atraso alto-alto de 20,6ps e atraso baixo-baixo de 20.4ps.

28. *Vamos realizar agora a *simulação de Monte Carlo*. Nesta simulação são realizadas, na verdade, várias simulações com parâmetros diferentes e podemos conhecer o comportamento do circuito para diversas condições de fabricação. Utilizando os comandos abaixo realize *Monte Carlo* (faça os ajustes necessários para seu circuito, não coloque no arquivo o modelo do transistor). Forneça os gráficos da tensão em c e na saída (valor típico e piores casos) e os gráficos do número de saídas *versus delayF* e *delayR*.

```

1 Va a 0 3V
2 Vb b 0 0
3 Vc c 0 pulse (0 3 0 1p 1p 2n 4n)
4 .tran 1n 30n 0n 10p
5 .meas tran delayF trig v(c) val=1.5 fall=5 targ v(out) val=1.5 fall=5
6 .meas tran delayR trig v(c) val=1.5 rise=5 targ v(out) val=1.5 rise=5
7 Cl out 0 30fF
8
9 .option SST_MTHREAD=1
10 * MONTE CARLO
11 .MC 100 NBBINS=20
12 .INCLUDE /local/tools/dkit/ams_3.70_mgc/eldo/c35/profile.opt
13 .LIB /local/tools/dkit/ams_3.70_mgc/eldo/c35/wc53.lib mc

```

Os histogramas de ocorrências de atrasos estão nas figuras 13 e 14. Na figura 15 podemos observar o sinal de entrada e o comportamento da saída típica, melhor e pior casos.

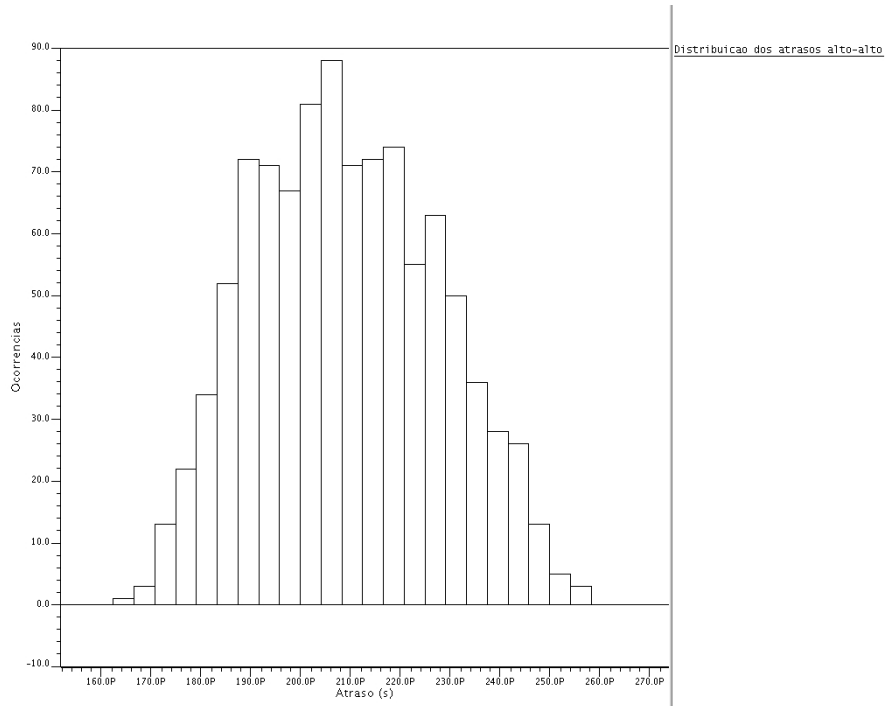


Figura 13: Histograma de ocorrências de atraso alto-alto obtido pela simulação de Monte Carlo sobre a porta lógica não inversora.

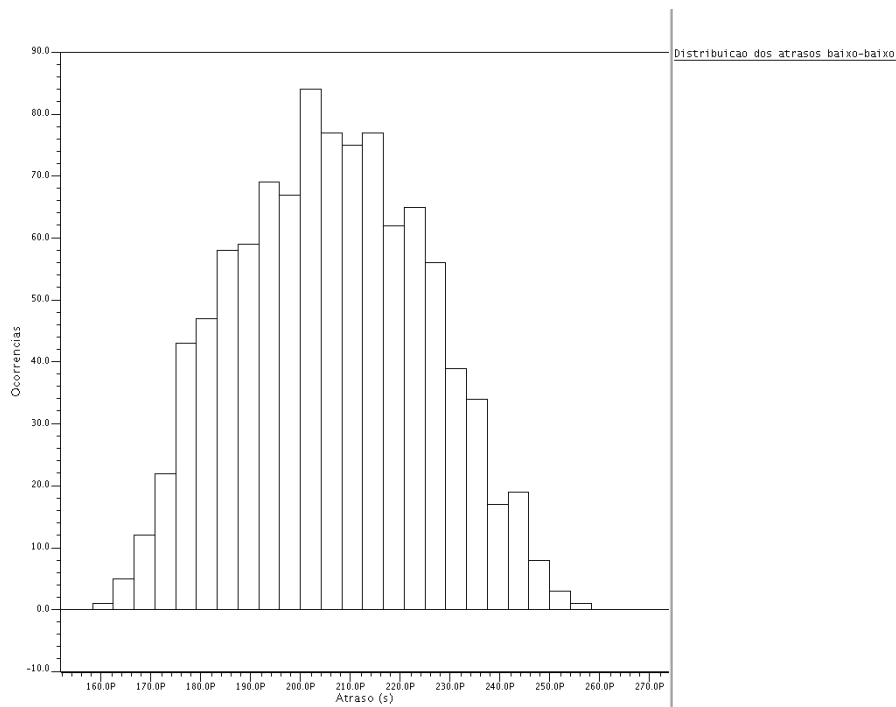


Figura 14: Histograma de ocorrências de atraso baixo-baixo obtido pela simulação de Monte Carlo sobre a porta lógica não inversora.

29. *Utilize o comando **Report – Windows** do *ICStation* para determinar o tamanho da célula (coloque o tamanho de sua célula no relatório).

Conforme o *Windows report* do *mentor*, temos que:

```
--- Cell Extent:      [[-1.693, 0.969], [20.481, 28.600]]
```

logo temos a área A da célula como:

$$A = \Delta x \cdot \Delta y \quad (26)$$

$$A = [20.481\mu m - (-1.693\mu m)] \cdot (28.600\mu m - 0.969\mu m) \quad (27)$$

$$\Rightarrow A = 612,7\mu m^2 \quad (28)$$

Assim a área da célula é de $612,7\mu m^2$

manual do Mentor-ELDO: /local/tools/mentor/shared/pdfdocs/eldo_ur.pdf

modelos dos transistores: /local/tools/dkit/ams_3.70_mgc/eldo/c35.

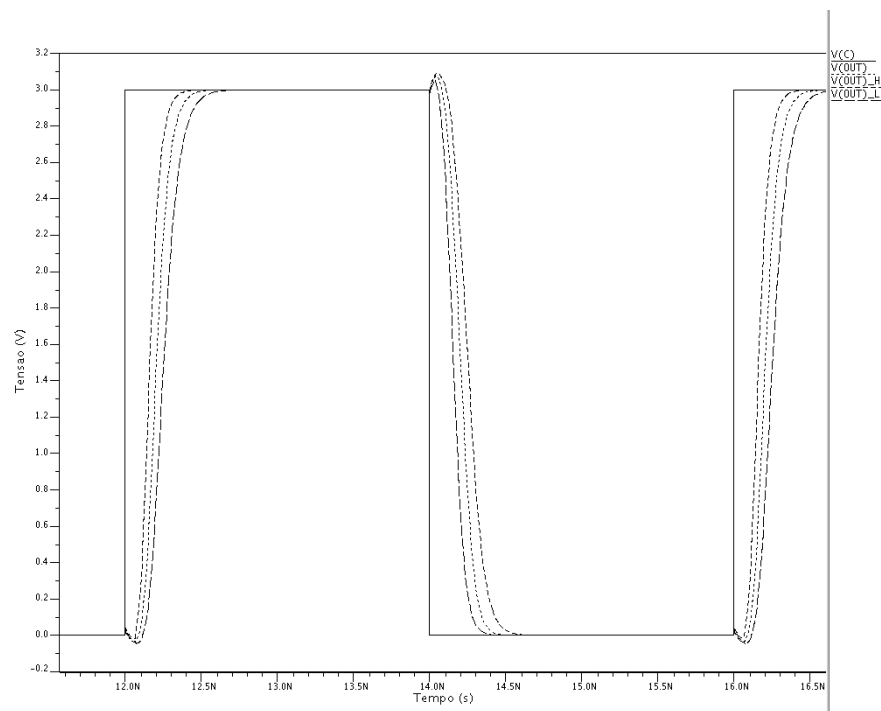


Figura 15: Sinal de entrada *C* utilizado durante a simulação de Monte Carlo sobre a porta lógica não inversora e sinal de saída *OUT* no melhor caso, caso típico e pior caso obtidos pelo simulação.