

Escola Politécnica da Universidade de São Paulo  
**Departamento de Engenharia de Sistemas Eletrônicos - PSI**

**PSI-3452- Projeto de Circuitos Integrados Digitais e Analógicos**

**LAB 2: Transistor NMOS- Desenho de Layout, Extração  
de Circuito Elétrico e Simulação (2017)**

## **1 - OBJETIVOS**

Esta sessão de lab visa dar continuidade à familiarização do estudante com a metodologia de projeto dedicado com aplicativo IC Station da Mentor Graphics e o programa de simulação de circuitos ELDO. Especificamente, objetiva:

- 1) a geração do leiaute de um transistor NMOS e a observação das regras de projeto utilizadas, dadas em dimensão  $\lambda$ .
- 2) Realizar a extração do diagrama elétrico deste transistor, realizar a simulação e compreender o modelo de transistor utilizado.

## **2 - PARTE TEÓRICA**

### **2.1. O transistor NMOS – funcionalidade**

O modelo mais simples do transistor MOS para o programa SPICE, o chamado NÍVEL 1, prevê que o transistor se comporte de acordo com as equações abaixo. Este é o modelo normalmente usado no desenvolvimento de equações de corrente, do dreno para a fonte ( $I_{ds}$ ), para se determinar as primeiras aproximações de comportamento de circuitos MOS, baseados nas tensões entre porta e fonte ( $V_{gs}$ ) e entre dreno e fonte ( $V_{ds}$ ):

$I_{ds} = 0$  , para  $V_{gs} < \text{ou} = V_{tn}$  (chamada de região de corte)

$I_{ds} = \beta \left( (V_{gs} - V_{tn}) V_{ds} - V_{ds}^2 / 2 \right)$  , para  $V_{gs} > V_{tn}$  e  $V_{ds} < V_{gs} - V_{tn}$  (chamada região linear)

$I_{ds} = \frac{\beta}{2} (V_{gs} - V_{tn})^2$  , para  $V_{gs} > V_{tn}$  e  $V_{ds} > \text{ou} = V_{gs} - V_{tn}$  (chamada região de saturação)

com  $\beta = \mu_n C_{ox} \frac{W}{L}$  ou  $\beta = \mu_n \frac{C_{ox}}{e_{ox}} \frac{W}{L}$  , onde

W e L- largura e o comprimento do transistor NMOS;

$\mu_n$  - mobilidade dos elétrons;

$C_{ox}$ - capacitância por unidade de área da porta do transistor;

$T_{ox}$ - a espessura do óxido fino, sob o silício-poli de porta;

$e_{ox}$ - constante dielétrica do óxido de silício ();

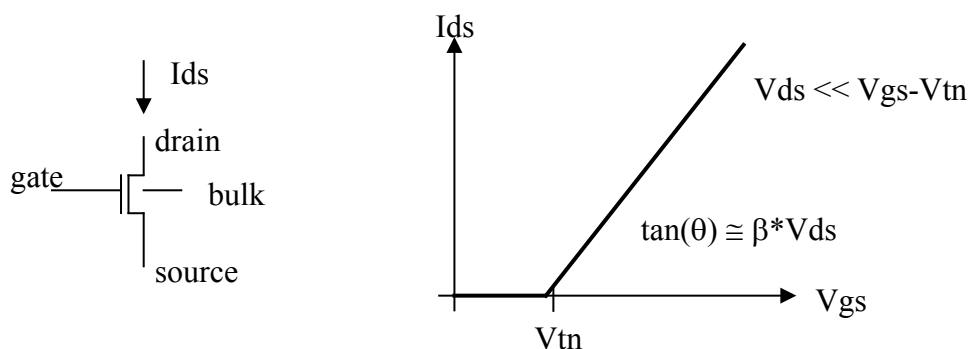
$V_{tn}$ - tensão de limiar de condução.

$V_{tn}$ ,  $\mu_n$  e  $C_{ox}$  (ou  $T_{ox}$ , espessura do óxido) são parâmetros de cunho tecnológico que devem ser fornecidos pelo fabricante em forma de parâmetros de modelo de transistor para simuladores elétricos (SPICE, ELDO, etc.).

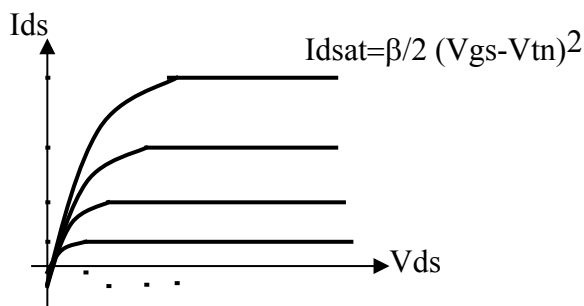
O modelo acima apresenta uma precisão razoável quando trata-se de transistores de geometria grande, porém, em escalas submicrônicas atuais, ele não é adequado. Outros parâmetros, referentes a efeitos de segunda ordem como, por exemplo, a velocidade de saturação por campo elétrico, devem ser utilizados. É o caso do modelo NÍVEL 53 do ELDO que adotamos nesta experiência.

Uma primeira curva interessante de se estudar é a resultante do experimento colocando-se uma tensão baixa em  $V_{ds}$  e aumentando gradualmente a tensão  $V_{gs}$ . Supondo

que a condição seja tal que o transistor venha da região de corte, quando  $V_{gs} < V_{tn}$  e vá direto para a região de linear, e supondo que  $V_{ds} \ll V_{gs} - V_{tn}$ , a equação do transistor nesta região será:  $I_{ds} \approx \beta V_{ds} (V_{gs} - V_{tn})$ . Veja pela curva abaixo que os valores de  $V_{tn}$  e de  $\beta$  podem ser obtidos. Tendo  $W$  e  $L$ , o valor de ' $\mu_n \text{ Cox}$ ' também pode ser obtido.



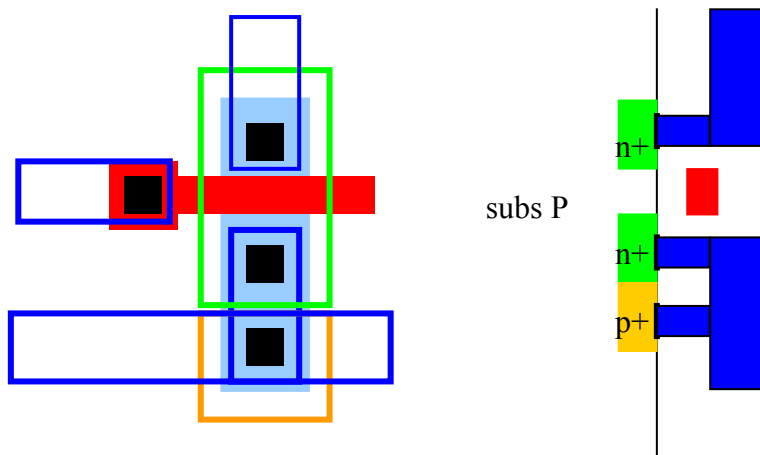
Uma outra curva interessante de se obter com as equações anteriores é a que mostra o transistor conduzindo quando tanto a tensão  $V_{gs}$  como a  $V_{ds}$  variam. Esta curva é mostrada na figura a seguir ( $I_{ds} \times V_{ds}$ ).



## 2.2. O Transistor NMOS – construção

Um desenho do transistor NMOS é mostrado na figura a seguir. Perceba que, além do transistor em si, é necessário fazer o contato de substrato, no caso para o substrato P, que, na tecnologia TSMC-035, será o próprio material da lâmina do chip. Portanto, para um transistor NMOS não será necessário desenhar o Poço N; este será necessário apenas quando houver necessidade de desenhar transistores canal P.

- Poço n
- Região Ativa
- Si-poli
- Seleção p+
- Seleção n+
- Contacto
- Metal 1



## 2.3. Extração de circuitos

### 2.3.1. Subcircuitos

Os arquivos tipo SPICE apresentam o objeto 'subckt' como forma de modular e hierarquizar a descrição de um circuito, facilitando a sua leitura. Funciona como um módulo de um circuito ou de um programa.

Um subcircuito tem como interface os seus nós. Por exemplo, um diodo tem dois nós e um circuito somador de um bit pode ter, por exemplo, cinco nós. A sintaxe para um subcircuito é mostrada a seguir:

**.SUBCKT Nome\_do\_subcircuito nome\_nó\_1 nome\_nó\_2 etc**

⇒ componentes internos do subcircuito e suas ligações

**.ENDS**

Os nomes dos nós são, neste contexto, denominados locais. Os subcircuitos existem para serem utilizados em outros circuitos ou subcircuitos. Ao ser incorporado dentro de um circuito maior, o subcircuito é instanciado da seguinte forma:

**X\_Nome\_da\_instância nome\_nó\_A nome\_nó\_B Nome\_do\_subcircuito**

Neste contexto, os nós, na definição da instanciação são denominados globais. A ordem dos nós é fundamental, pois a associação é feita na sequência das listas.

a) Exemplo 1:

Descrição:

```
. SUBCKT somador carry_in  carry_out  in_1  in_2  out
=>Descrição de transistors interconectados
. ENDS
```

No exemplo do somador acima, temos quatro nós (locais).

Fazemos duas instanciações (são construídos dois blocos somadores):

```
X_somador_1 c_i_1 co_1 in_1_A in_1_B out_1 somador
X_somador_2 c_i_2 co_2 in_2_A in_2_B out_2 somador
```

b) Exemplo 2:

No caso do exemplo 1, se a primeira instanciação fosse da forma

```
X_somador_1 c_i_1 co_1 out_1 in_1_A in_1_B somador
```

provavelmente teríamos um erro, pois o nó global out\_1 (saída) seria associado ao nó local in\_1 (entrada).

### 2.3.2. Arquivos Extraídos

No fluxo de projeto de circuitos integrados dedicados, após a edição do leiaute, é comum realizar-se a extração do diagrama elétrico do circuito leiautado, para a sua simulação. A simulação permite: a) verificar a funcionalidade do circuito, detectando eventuais erros inseri-

dos pelo projetista; b) confirmar as características elétricas, uma vez que todos os elementos parasitários (decorrentes de interconexão e capacitâncias intrínsecas dos transistores) podem ser extraídos e incorporados à descrição do circuito.

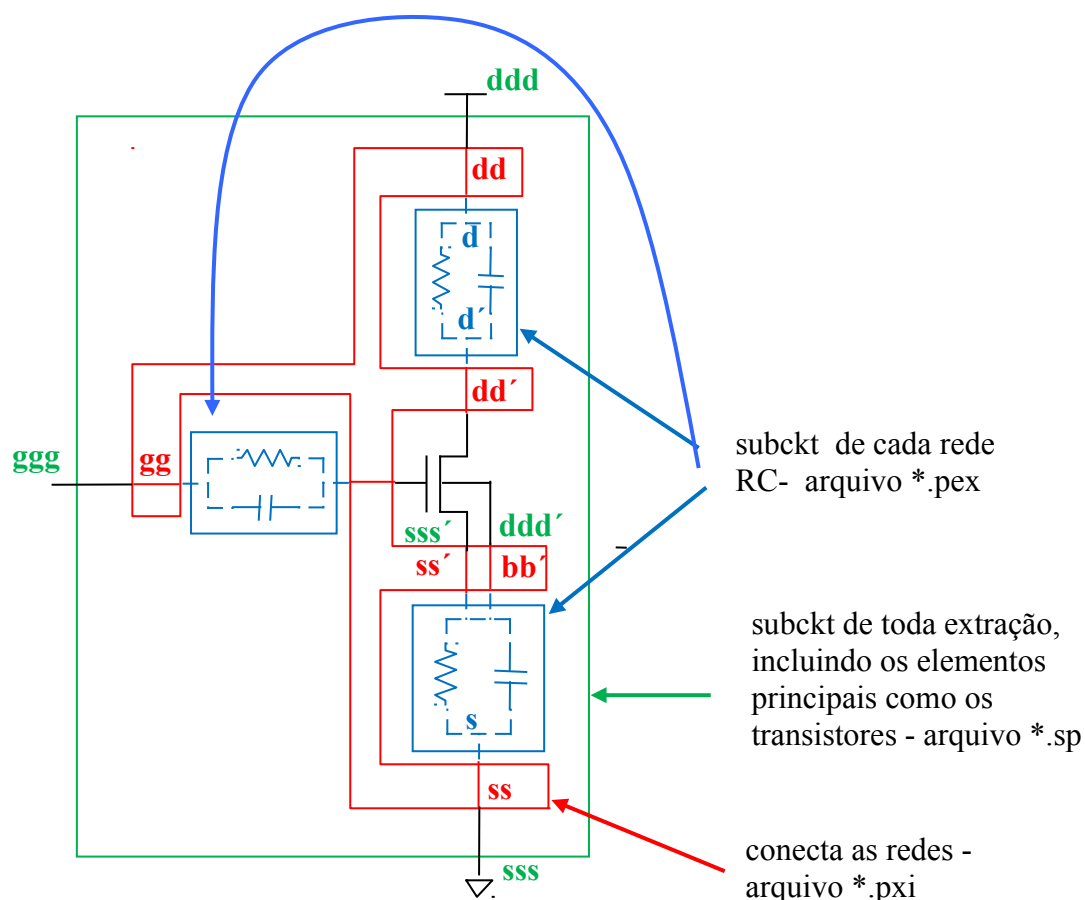
No ambiente de projeto do IC Station da Mentor, a ferramenta que realiza a extração é o programa CALIBRE. Após a extração, um conjunto de arquivos descritivos do circuito do tipo SPICE é gerado. A figura abaixo ilustra o resultado da extração de um leiaute específico de um transistor. Por depender da forma como o projetista constrói o seu leiaute, é improvável que duas extrações de um mesmo circuito sejam idênticas.

Os arquivos são os seguintes:

**\*.sp.pex**- arquivo contendo todos os subcircuitos extraídos, em forma de capacitâncias e resistências, correspondentes às interligações de metal, silício-poli, etc. do leiaute (comando 'subckt' do formato SPICE). Cada polígono desenhado para cada camada será extraído em forma de componente elétrico. Como ilustrado no exemplo da figura abaixo, cada subcircuito (em azul) conterá nós internos, com os seus nomes locais (como d e s).

**\*.sp.\*.xi**- arquivo que contém as interconexões entre subcircuitos e destes para outros nós globais do arquivo global de extração. No exemplo da figura, correspondem às conexões contidas na área em vermelho; o subcircuito superior à direita, por exemplo, tem os seus nomes locais, d e d', conectados pelos nós globais, dd e dd'.

**\*.sp** – arquivo do leiaute completo extraído, na forma de um subcircuito global (principal). Contém componentes como os transistores e instâncias de capacitores e resistores; também engloba nomes globais dos nós que foram alocados durante a edição do leiaute, e que são conectados via os componentes do arquivo \*.sp.\*.xi. No exemplo da figura, na área verde, o nome ddd foi definido pelo projetista no seu leiaute e torna-se a um nó global de \*.sp.



### **3. PARTE EXPERIMENTAL**

Os experimentos a seguir referem-se a um transistor a ser construído com a tecnologia TSMC 0,35  $\mu\text{m}$ .

#### **3.0. Preparo Inicial**

Faça os procedimentos de preparação já vistos em experiência anterior (no sistema operacional Linux), em particular, para esta sessão:

→ Crie diretório:  $\sim/psi3452/lab2$  (para visualização e acompanhamento, use o navegador de pastas *konqueror*)

**Observação:** para completar a folha de respostas, você deve usar a ferramenta de edição de texto, LibreOffice, seguindo o procedimento do documento “**Usando os servidores de aplicativos xserver1 e xserver2**”.

#### **3.1. Obtenção de parâmetros do transistor**

Abra o arquivo `/tools/mgc_tree/adk3_1/technology/ic/models/tsmc035.mod` com o Kwrite ou o Kate, e observe os parâmetros fornecidos. Tente comparar alguns dos parâmetros com o do modelo Nível 1, visto em teoria (lembre-se que trata-se de um modelo muito mais moderno e sofisticado com atribuição de nível superior).

⇒ **Completar na folha de respostas com as informações solicitadas (item 3.1)**

#### **3.2. Definição das dimensões do transistor canal N.**

As dimensões serão definidas de acordo com o número USP do aluno, da seguinte forma:

**Largura do transistor ( $W_n$  em  $\lambda$ )** é o penúltimo algarismo do seu número USP. Se for  $< 5$ , some o valor 5 a ele.

**Comprimento do transistor ( $L_n$ )** é formado com o último algarismo do seu número USP. Considere o algarismo como  $k$ , então a dimensão é  $(k \bmod 4)$ . Caso o resultante for  $< 2$ , some o valor 2 a ele.

Exemplo 1: #USP 3164124 leva a  $W_n = \underline{7\lambda}$  (ou seja,  $2+5$ ) e  $L_n = \underline{2\lambda}$  (pois  $k=0$ )

Exemplo 2: #USP 3164167 leva a  $W_n = \underline{6\lambda}$  e  $L_n = \underline{3\lambda}$ .

**Observação:** Como usaremos a tecnologia 0,35 $\mu\text{m}$ , no exemplo acima, temos que  **$\lambda=0,2\mu\text{m}$**  (dentro da opção MOSIS SCMOS). Logo, o transistor terá:

Exemplo 1:  $W_n = 7\lambda = \underline{1,4\mu\text{m}}$  e  $L_n = 2\lambda = \underline{0,4\mu\text{m}}$ .

Exemplo 2:  $W_n = 6\lambda = \underline{1,2\mu\text{m}}$  e  $L_n = 3\lambda = \underline{0,6\mu\text{m}}$ .

⇒ **Completar a folha de respostas (item 3.2)**

### 3.3. Levantamento de curva teórica $I_{ds}$ x ( $V_{gs}$ , $V_{ds}$ ) com o modelo simplificado

Vamos agora calcular valores de corrente  $I_{ds}$  x ( $V_{gs}$ ,  $V_{ds}$ ) com as equações simplificadas dadas na parte de teoria.

Obs. Os parâmetros que você separou na seção 3.1 são considerados básicos, isto é, nas equações bastante complexas, utilizadas no simulador de circuitos, sofrem uma série de correções por parte de outros parâmetros do modelo SPICE. As correções atuam, quase na sua totalidade, no sentido de diminuir o valor da corrente  $I_{ds}$ . Desta forma, calcular a corrente com as equações simplificadas dadas na seção de teoria e com os parâmetros básicos levará provavelmente a uma **superestimação** do valor de  $I_{ds}$ .

Para facilitar a obtenção da curva teórica simplificada do transistor, uma planilha será fornecida. Copie o arquivo ***IdsVds.ods*** que está em */home/disc/psi3452/lab2* para a sua recém-criada pasta *~/lab2*. No **konqueror**, clique duas vezes sobre o ícone ***IdsVds.ods*** e o aplicativo de planilha do **LibreOffice** se abrirá.

Na parte superior da planilha, substitua os valores arbitrários pelos que você encontrou nas seções 3.1 e 3.2. Atente para as **unidades** dos parâmetros.

**Observação:** A constante dielétrica do óxido  $\epsilon_{ox}$  já é fornecida na planilha.

A seguir, para cada um dos valores de  $V_{gs}=2V$  e  $V_{gs}=3V$ , calcule os valores de  $V_{dsat}$  e inclua-os na planilha. Como um exemplo, os valores calculados de  $V_{dsat}$  poderiam ser (caso  $V_t=0,68V$ ):

	$V_{gs}(V)$	$V_{gs}(V)$
	3	2
$V_{dsat}(V)$	2,32	1,32

Com isso, as curvas  $I_{ds}$  x ( $V_{gs}=2V$ ,  $V_{ds}$ ) e  $I_{ds}$  x ( $V_{gs}=3V$ ,  $V_{ds}$ ) poderão ser automaticamente geradas. A planilha já prepara uma sequência de valores de  $V_{ds}$  espaçadas de 0,2V. Para ter o número certo de pontos faça o seguinte (veja a ilustração abaixo):

- selecione as células inferiores e coloque o cursor no canto inferior direito da célula da direita (deve aparecer uma cruzeta) e arraste pra baixo;
- faça de tal forma que tenha pelo menos cinco pontos **acima** da tensão de saturação para  $V_{gs}=3V$

$V_{ds}(V)$	$I_{ds}(mA)$ ( $V_{gs}=3V$ )	$I_{ds}(mA)$ ( $V_{gs}=2V$ )
0	0	0
0,2	0,089920064	0,05167116
0,4	0,172190347	0,095692539
0,6	0,246810849	0,132064137
0,8	0,313781571	0,160785954
1	0,373102511	0,181857991
1,2	0,424773671	0,195280246

⇒ **Siga as instruções da folha de respostas (item 3.3)**

### 3.4. Desenhando o transistor no IC Station

No Linux:

- Abra um terminal e vá para este diretório `~/psi3452/lab2`
- Tecle `source /home/disc/psi3452/perfil_mgc1` para ativar as variáveis do ambiente. Lembre-se sempre de realizar esta tarefa quando abrir um terminal- este alerta não será mais repetido futuramente...
- Tecle `adk_ic &` para disparar a ferramenta de edição IC Station.
- Abra uma nova célula (utilizando o comando *New* e preenchendo, em seguida, a janela que se abre). Neste manual, adotaremos **NMOS** como nome da célula. Utilize o mesmo processo e as mesmas regras adotadas na sessão de lab 1.

Durante a edição do seu leiaute você deverá obedecer às regras do processo CMOS da TSMC035. Algumas regras importantes foram relacionadas indicadas no anexo 1 da apostila do lab1. Utilize-as como referência.

ATENÇÃO: o programa DRC deverá ser usado durante a edição para verificar se nenhuma regra de projeto está sendo violada.

No desenho de um transistor, as máscaras (aqui chamadas de camadas) devem ser inseridas na sequência que tem sido apresentada nas aulas de teoria:

- ❖ Nwell (apenas para transistor PMOS) (utilizada na próxima sessão de lab)
- ❖ Active
- ❖ Poly
- ❖ N\_ Plus\_Select
- ❖ P\_ Plus\_Select (apenas para transistor PMOS) (utilizada na próxima sessão de lab)
- ❖ Contact\_to\_Poly
- ❖ Contact\_to\_Active
- ❖ Metal 1
- ❖ Via (utilizada na próxima sessão de lab)
- ❖ Metal2 (utilizada na próxima sessão de lab)

**Observação:** Como já visto na teoria, a camada P\_ Plus\_Select pode ser usada para definir a região de contato de substrato. Entretanto, na nossa sessão com o IC Station, usaremos uma outra forma para isto, como poderá ser vista mais adiante.

Os tamanhos das camadas são computadas da seguinte forma (medidas em  $\lambda$ ):

- A camada **Active** deve ter a dimensão  $W_n \times (2 * DRC6\_1 + 2 * DRC6\_2 + 2 * DRC6\_4 + L_n) \Rightarrow$  calcule o seu valor
- A camada **Poly** deve estar centralizada em relação à camada Active e ter  $L_n \times (W_n + 2 * DRC3\_3) \Rightarrow$  calcule o seu valor

⇒ **Completar a folha de respostas (item 3.4.A)**

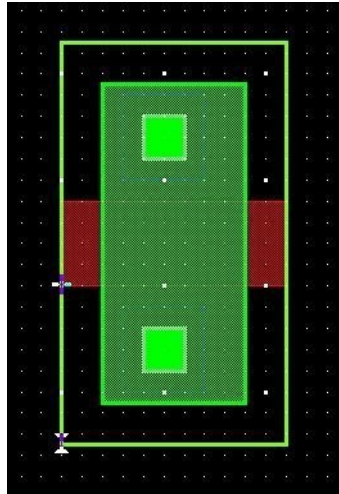


A figura resultante, exceto pela largura e comprimento do canal, deve ser similar à mostrada a seguir. Use-a como referência para o seu desenho no IC Station.

→ Posicione as camadas de *Active* e *Poly*, procurando centralizá-los (em 0,0).

**Observação:** Não se esqueça de gravar o seu projeto a cada mudança!! A ocorrência de algum pane no software e a perda de edições realizadas e não gravadas é um evento extremamente frustrante e desagradável !!!

**Observação 2:** Após a gravação, o seu desenho NÃO estará mais reservado para edição. Para continuar editando o seu trabalho, reserve o seu desenho novamente para edição, como instruído na sessão de lab 1.



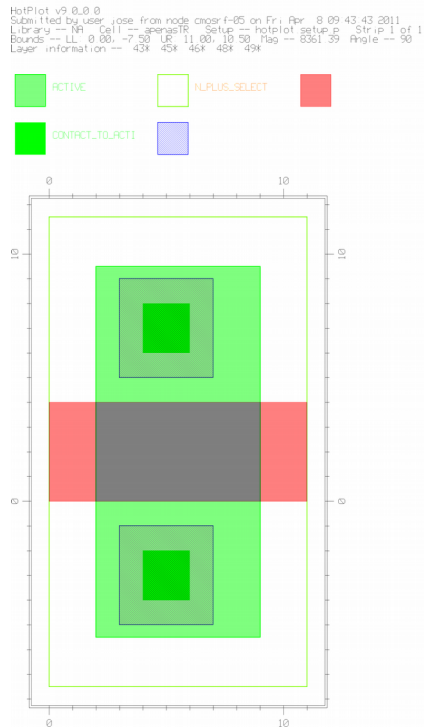
- A camada *N Plus* deve conter a camada *Active*, com um avanço extra de  $2\lambda$  (DRC4\_2) em todos os lados.
- A camada *Contact* (dos contatos de fonte e dreno) devem ter exatamente  $2\lambda \times 2\lambda$  (DRC6\_1) e devem estar distantes de  $2\lambda$  (DRC6\_4) da camada *Poly* e estar contidos na camada *Active*, com, no mínimo,  $1,5\lambda$  (DRC6\_2) de cada lado.
- A máscara *Metall* deve cobrir e conter os contatos, com um avanço extra de, pelo menos,  $1\lambda$  (DRC7\_3) de cada lado.
- Se não o fez ainda, grave o seu desenho.

Ative agora a verificação do leiaute (DRC) (siga as instruções da sessão de lab. 1) para observar se algum erro foi cometido. Se você cometeu algum erro que viole as regras acima mencionadas, corrija-as e reative o seu DRC até que não ocorram mais indicações de erros.

**Atenção:** haverá um único erro que você não será capaz de corrigir neste momento. Qual é ele? Pense o porquê.

A próxima ação é gravar o desenho do leiaute no formato **TIFF** (como realizado na sessão do lab1). Ela deve ter a forma similar ao da figura abaixo, porém com as medidas derivadas do número USP. Verifique que ela apresente as dimensões corretas.

⇒ Siga as instruções da folha de respostas (item 3.4.B)



### 3.5. Adicionando o contato de substrato

O substrato precisa ser polarizado e um contato deve ser criado para este fim. Para implementar o contato de substrato será utilizada uma célula da biblioteca padrão que acompanha o pacote ADK.

Na pasta `/tools/mgc_tree/adk3_1/technology/ic/process/tmsc035` estão armazenados diversos blocos lógicos digitais (inversores, nands, nors, etc...). Com o navegador de pastas, verifique o seu conteúdo.

Na pasta `/tools/mgc_tree/adk3_1/technology/ic/process/tmsc035_via` estão armazenadas células de contato de substrato possíveis de serem usadas nesta tecnologia. Com o navegador de pastas, verifique o seu conteúdo.

Usaremos a célula `'pwell_contact'` que se encontra nesta segunda pasta. Para isto, no *IC Station*, acione na barra de menu (na parte superior da tela) o comando

→ **Add→Instance**

e procure (usando o *browser*) a célula acima. Em seguida, na tela de desenho, acrescente esta célula encostando-a no transistor já desenhado pela parte inferior. Use a tecla *Esc* (do teclado) para sair deste modo.

Para ver todas as camadas desenhadas, inclusive as das células:

→ Selecione todo o leiaute com: **Select→Select→All**

→ Visualize com: **View→Peek→Peek** e na caixa selecione o Number of Level **99** e tecla OK.

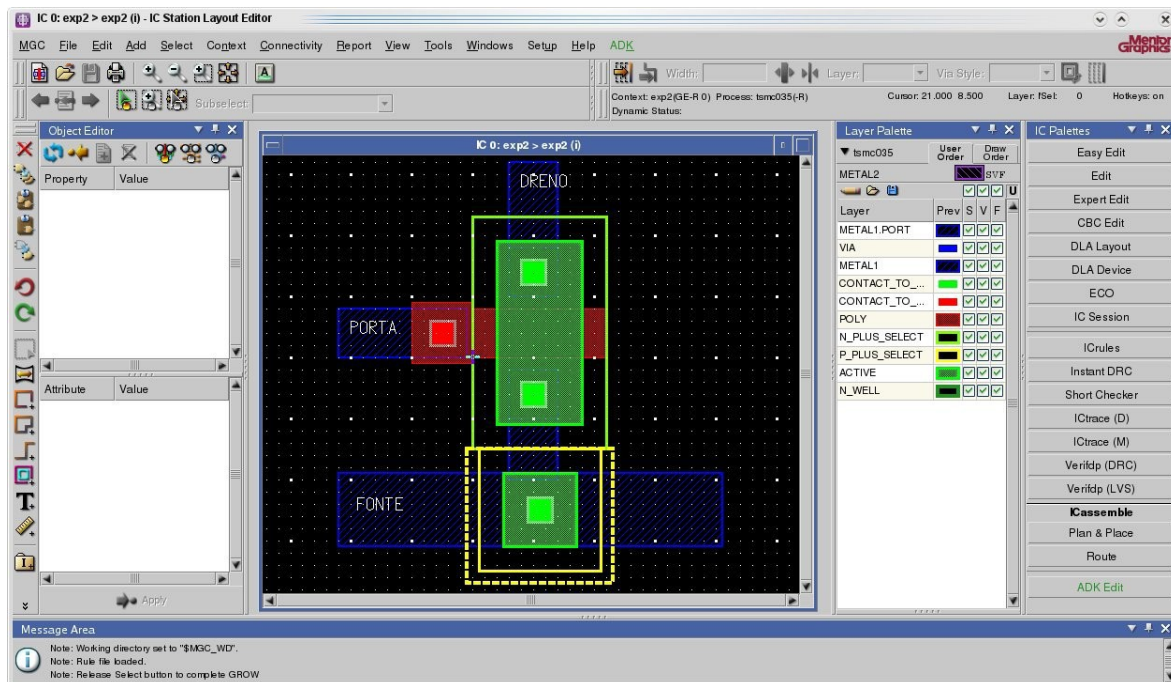
→ Deselecione todo o leiaute (*F2* com o ponteiro do *mouse* sobre a grade, mas fora do leiaute).

→ Se o leiaute não couber na tela use a opção **View -> All**.

⇒ **Completar a folha de respostas (item 3.5)**

### **3.6. Inserção de portos (*port*) ligados ao METAL1 e seus rótulos (*labels*)**

Para definir os portos do circuito e associá-los a rótulos (nomes dos nós), algumas camadas deverão ser adicionadas, como ilustrado na figura a seguir.



#### **3.6.1. METAL 1 para dreno, fonte e substrato**

Primeiramente, linhas de METAL 1 serão acrescentadas para conectar aos contatos de fonte, dreno e substrato. Veja qual é a largura mínima do METAL 1 de acordo com as regras de projeto (Apêndice 1 da apostila do lab 1)

Conecte a fonte do NMOS com o contato de substrato usando a camada METAL1, como na figura a seguir (adotaremos esta condição em que ambos estão em curto e serão polarizados em GND). Há diversas formas para fazer tal ligação. Você pode desenhar um retângulo de METAL1 encostando-o no quadrado de METAL1 já desenhado e cobrindo o contato de substrato. Outra forma (mais elegante) é aumentar o tamanho do quadrado de METAL1 já desenhado sobre o contato da fonte de forma a envolver o contato de substrato (usando o comando *Stretch*). Fica ao critério do aluno como realizar tal passo.

Desenhe mais uma tira vertical de Metal1 cobrindo o contato de dreno conforme mostrado na figura abaixo.

Desenhe mais uma tira horizontal de Metal1 cobrindo o contato de fonte conforme mostrado na figura abaixo.

### **3.6.2. METAL 1 e contato de POLY**

Agora, o aluno deverá acrescentar um contato que conecte a camada de METAL 1 ao de POLY, como indicado na figura.

Primeiramente, acrescente um quadrado de POLY de dimensão  $5\lambda \times 5\lambda$  (DRC5\_1+2\*DRC5\_2) separado em  $2\lambda$  do canal do transistor (DRC3\_5 diz que separação pode ser  $1\lambda$ , mas para respeitar a regra de espaçamento entre metais, use  $2\lambda$ ). Cuide para que este quadrado de POLY esteja ligado à porta do transistor (uma ligação feita com o próprio POLY).

A seguir, acrescente uma camada de contato entre a camada de Metal 1 e o de POLY, usando o comando '**Easy Edit -> Shape**' e selecionando a camada '**Contact\_to\_Poly**'. Gere-o com a dimensão  $2\lambda \times 2\lambda$  (DRC5\_1) e centralize-o no quadrado de POLY desenhado no passo anterior.

Finalmente, cubra o contato com um retângulo de METAL1 de dimensão  $4\lambda \times 4\lambda$  (DRC5\_1+2 DRC7\_3).

Para finalizar, faça uma extensão de METAL1 como mostra a figura.

### **3.6.3. Portos e Rótulos- definindo nós elétricos do circuito**

Adicione texto para cada terminal (PORTA, FONTE, DRENO, por exemplo) ligado às tiras de METAL1 que você acabou de desenhar. Para isto, a camada **METAL1.PORT** será usada. Cada camada de porto corresponderá a um nó elétrico (porto) da célula que será usado como terminal de comunicação desta célula com alguma outra célula irmã, ou à sua célula mãe (de maior hierarquia).

**Atenção:** A camada **METAL1.PORT** NÃO corresponde a uma máscara, sendo usada apenas para fixar os portos da célula.

Para adicionar as portas, use o comando '**Add→Text**'.

Observe se em '**Object Editor**' o *Layer Name* seja 'METAL1.PORT'. Se não estiver, isto pode ser feito tanto usando a *Layer Palette*, como escrevendo diretamente.

Também, no '**Object Editor**' coloque o nome do porto a ser usado (DRENO, por exemplo) no campo *Value*.

Para que este valor (nome) seja efetivado pelo '**Object Editor**' tecele RETURN

Em seguida, posicione o cursor (grudado no cursor está o nome que você usou) sobre o METAL1 que servirá de porto e clique o botão da esquerda do mouse para implementar.

Repita o procedimento para, uma para a fonte e porta. A figura resultante deve ser similar com a mostrada acima.

Não se esqueça de gravar a sua célula e depois volte a habilitá-la para edição.

Faça a verificação do leiaute (DRC). Se houver erros, elimine-os até que nenhum erro apareça.

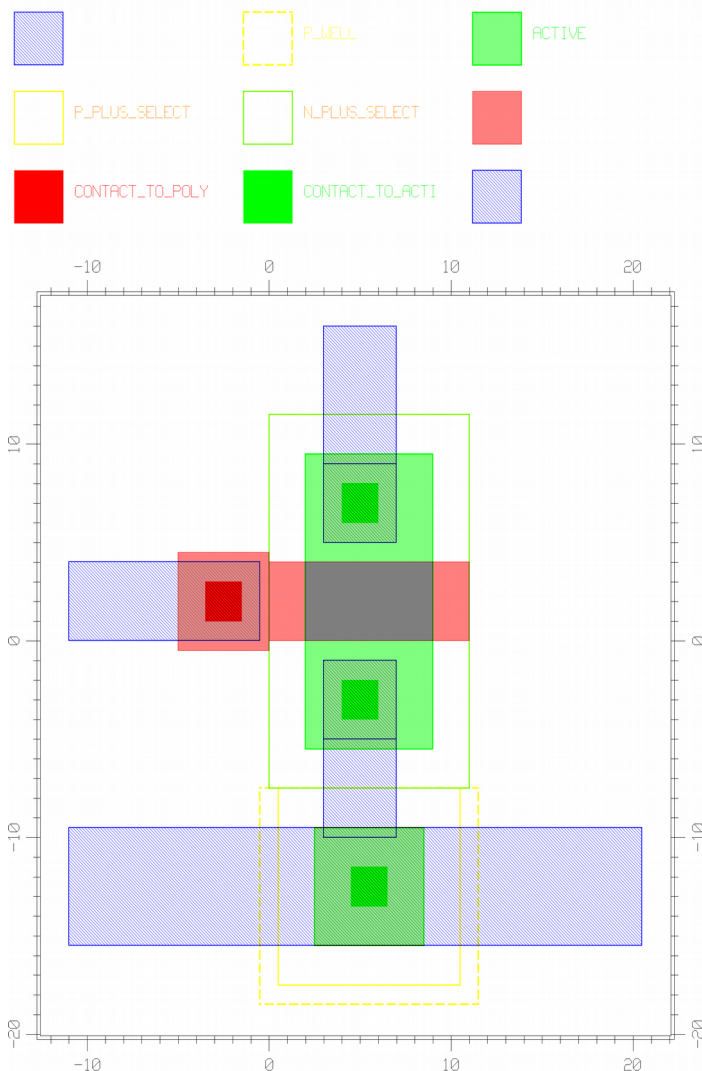
Grave o leiaute (que deve ser parecida com a figura a seguir) em formato TIFF.

⇒ **Sigas as instruções e complete a folha de respostas (item 3.6)**

```

HotPlot v9 0.0.0
Submitted by user jose from node cmosrf-05 on Fri, Apr 8 09:45:19 2011
Library -- NA Cell -- exp2 Setup -- hotplot setup.p Strip 1 of 1
Bounds -- LL: -11 00, -18 50 UR: 20 50, 15 50 Flag -- 4041 61 Angle -- 90
Layer information -- 2* 41* 43* 44* 45* 46* 47* 48* 49*
4113

```



### 3.7. Extração do diagrama elétrico

Como relatado na seção de teoria, pode-se agora realizar a extração do diagrama elétrico para sua posterior simulação. A extração é feita com o programa CALIBRE. Para isto siga os seguintes passos:

- Acione **Tools**→**Calibre**→**Run PEX**.
- Na janela que aparece após a licença ser adquirida, preencha o campo Load Runset com o arquivo `/tools/mgc_tree/adk3_1/technology/calibre/pex.tsmc035.runset`, que contém algumas das variáveis usadas na extração)
- Confirme clicando em OK
- Clique em **Run Pex** e aguarde.

Três arquivos são gerados: ‘NMOS.sp’, ‘NMOS.sp.pex’ e ‘NMOS.sp.NMOS.pxi’. Ob-

serve que o nome depende do nome da célula que originou os arquivos, no caso, NMOS (caso você tenha mantido este nome). Reveja a seção 2.3 para entender qual é o conteúdo destes arquivos.

⇒ **Sigas as instruções e complete a folha de respostas (item 3.7)**

### **3.8. Simulação de Ids x Vds usando o ELDO e visualização usando o EZWAVE**

O ELDO é o simulador elétrico da Mentor Graphics (muito semelhante ao SPICE).

O arquivo *NMOS.sp* (criado anteriormente) corresponde ao seu leiaute, porém está em forma de subcircuito e precisa ser instanciado em algum arquivo topo. Use o programa **Kwrite** para criar um arquivo semelhante ao do exemplo mostrado a seguir. Grave o arquivo com um nome qualquer à sua escolha. No exemplo abaixo o arquivo foi denominado de **exp2IdsVds.sp**.

**Observação:** O arquivo exemplo **exp2IdsVds.sp** a seguir encontra-se na pasta **/home/disc/psi3452/lab2**. Se quiser, copie este arquivo para a sua pasta e apenas edite as alterações que forem necessárias.

**Atenção:** No exemplo, o *xEXP2* é o nome de uma instância do sub-circuito NMOS que está descrito em *NMOS.sp*. A ordem em que os terminais globais são descritos **deve ser a mesma** do subcircuito NMOS em *NMOS.sp*. O extrator **pode designar os terminais em diferentes sequências**, dependendo da posição geométrica dos mesmos no leiaute específico do aluno.

**Observação 1:** A primeira linha do arquivo é **sempre** vista pelo ELDO como **comentário**.

**Observação 2:** A descrição do modelo de transistor é incluída automaticamente no seu arquivo topo de simulação pelo comando `‘.include /tools/mgc_tree/adk3_1/technology/ic/models/tsmc035.mod’` (reveja o item 3.1).

=====

#### **ARQUIVO EXEMPLO (exp2IdsVds.sp)**

Simulacao do transistor NMOS no ELDO para determinar a curva Ids x Vds

```
.include "NMOS.sp"
.include /tools/mgc_tree/adk3_1/technology/ic/models/tsmc035.mod
xEXP2 porta dreno 0 NMOS
vdreno dreno 0 3.3
vporta porta 0 3.3
.dc vdreno 0 3.3 0.1 vporta 0 3.3 0.5
.plot dc i(vdreno)
.end
```

=====

**Atenção:** o manual do ELDO encontra-se em **/tools/mgc\_tree/ams/docs/pdfdocs/**. Verifique-o, procurando, por exemplo, os capítulos de **fontes independentes, elementos passivos, sub-circuitos e decrição do mosfet**. Encontrou-os?

Outros elementos a serem adicionados no arquivo de simulação são as fontes de tensão ou corrente, além da definição do tipo de análise será encaminhada na simulação. O arquivo `exp2IdsVds.sp` está preparado para uma simulação `IdsXVds` (para diferentes valores de  $V_{gs}$ ). Tenha certeza que consegue associar os comandos de tensão com a simulação desejada.

Para rodar o ELDO você deve acessar a janela e pasta de onde rodou o *script* 'perfil\_mgc1'. Para disparar o simulador tecle:

→ **eldo exp2IdsVds.sp** (a sintaxe do comando é `eldo nome_do_seu_arquivo_de_simulação`).

Corrija eventuais erros até que não haja problemas tanto de compilação como de execução.

O programa **ezwave** é usado para visualizar gráficos. Para rodar o ezwave tecle:

→ **ezwave exp2IdsVds.wdb** (a sintaxe do comando é `ezwave nome_escolhido_por_você.wdb` – este arquivo é gerado pelo ELDO).

No **ezwave**, abra a pasta '**exp2IdsVds**' na janela da esquerda e arraste o arquivo '**DC**' para a janela à direita.

A curva resultante deve ser parecida com a mostrada na próxima figura.

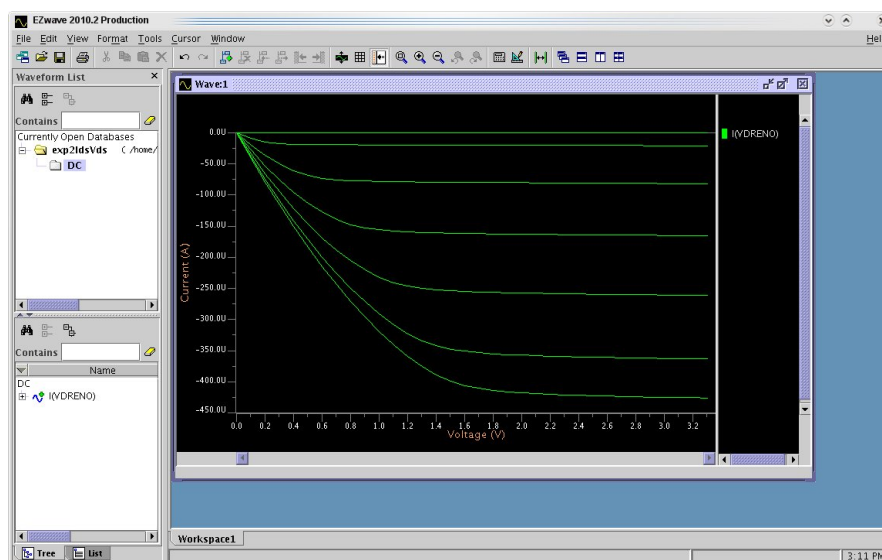
**Observação:** A corrente mostrada é negativa. Você esperava isto para um transistor canal N?

Para obter a curva em formato portátil para outros aplicativos clique:

→ **File→Print**

→ Na janela que aparece clique em File e digite um nome(**IdsVds**, por exemplo).

→ Tecle **OK**.



Em sua pasta de trabalho aparecerá o arquivo **IdsVds.ps** (formato post-script). Para converter este arquivo, use o programa GIMP.

Dispare o **GIMP**:

→ **gimp**

Abra o arquivo com o comando **File→Open** e na janela que aparece selecione o arquivo **IdsVds.ps** confirmando com OK.

Aparecerá uma nova janela perguntando se você quer importar este arquivo; responda **Importar**. Escolha o nome com a extensão '.jpg', ou seja, acione o comando **File→SaveAs**, e escolha o nome **IdsVds.jpg**. Grave o arquivo, com a resolução proposta de 85%, e o arquivo exportado para jpeg aparecerá no diretório.

Com o auxílio do arquivo Excel da seção 3.3 e da curva gerada, deve-se comparar os valores obtidos pela simulação com o do modelo simplificado. Para cada uma das tensões  $V_{GS}=2.0V$  e  $=3.0V$ , escolha os mesmos dois pontos de  $V_{ds}$  definidos na Seção 3.3.

Usando os valores simulados como referência, por serem mais precisas, calcule os erros observados:

$$\text{Erro (\%)} = 100 * (\text{Ids\_idealizado} - \text{Ids\_simulado}) / \text{Ids\_simulado}$$

⇒ **Sigas as instruções e complete a folha de respostas (item 3.8)**

### **3.9. Simulação de Ids x Vgs usando o ELDO e visualização usando o EZWAVE**

Um novo arquivo deve ser criado para a simulação IdsXVgs (parametrizado por valores de  $V_{ds}$ ). Crie um novo arquivo de simulação cujo nome, por exemplo, será **exp2IdsVgs.sp** (você pode usar outro nome, se desejar). A forma mais simples de criar este arquivo é criar uma cópia do seu arquivo anterior e modificá-lo, como mostrado abaixo (não se esqueça de modificar a primeira linha, de comentário).

=====

#### **ARQUIVO EXEMPLO (exp2IdsVgs.sp)**

*Simulacao no eldo do Transistor Ids x Vgs nome exp2IdsVgs.sp*

*.include "NMOS.sp"*

*.include /tools/mgc\_tree/adk3\_1/technology/ic/models/tsmc035.mod*

*xEXP2 porta dreno 0 NMOS*

*vdreno dreno 0 0.1*

*vporta porta 0 3.3*

*.dc vporta 0 1 0.01*

*.plot dc i(vdreno)*

*.end*

=====

Tenha certeza que consegue associar os comandos de tensão no arquivo exp2IdsVgs.sp com a simulação desejada.

Não se esqueça que para rodar o ELDO, você deve utilizar a janela e diretório onde rodou o 'perfil\_mgc1'.

Tecla:

→ **eldo exp2IdsVgs.sp.**



Verifique se não ocorreram erros na execução (corrija-os se houver).

Rode o **ezwave**:

→ **ezwave exp2IdsVgs.wdb**

No **ezwave**, abra na janela a esquerda a pasta '**exp2IdsVgs**' e arraste o arquivo 'DC' para a janela à direita.

A figura resultante deve ser parecida com a seguinte. Repetindo o procedimento anterior, ou seja, gerando o '.ps' e depois o '.jpg', grave a curva.

⇒ **Sigas as instruções e complete a folha de respostas (item 3.9)**

