

به نام خداوند بخشنده مهربان

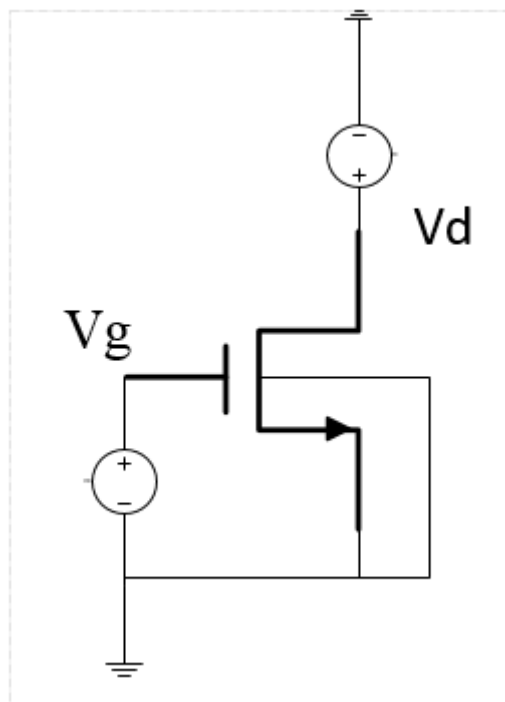
جزوه آموزشی نرم افزار HSPICE

برای نوشتن یک برنامه ساده در نرم افزار hspice ، نیاز هست که برنامه در یک فایل text نوشته شود و با پسوند sp. ذخیره شود.

اولین خط برنامه خوانده نمی شود، به طور کلی بجز سطر اول برنامه که شامل عنوان برنامه می باشد: ترتیب سطرها در نوشتن برنامه و اجرای آن هیچ اثری ندارد. ترتیب برنامه ها برای این است که اشکالات برنامه راحت تر گرفته شود.

نرم افزار به حروف کوچک و بزرگ حساس نیست و تمامی دستورات با . شروع میشوند.

در ادامه یک نمونه برنامه برای یک ترانزیستور nmos نوشته شده که تک تک خطوط برنامه را توضیح خواهیم داد:



```

My Program

*****LIBRARY*****
.protect
.LIB'mm018.1' TT
.unprotect
*****

.option post=2
*****PARAMETERS*****
.param VDDVAL=1.8

*****CIRCUIT DESCRIPTION*****

M1 D G S B NMOS L=180nm W=1um

*****ANALYSIS*****
vg g 0 vddval
vd d 0 vddval
vb b 0 0
vs s 0 0

.dc vg 0 vddval .01
*.dc vg 0 vddval 0.01 sweep vb 0 .3 .1
*.dc vd 0 vddval .01 sweep vg 0 2 .4
*.dc vd 0 vddval .01 sweep vg 0 .5 .1

.probe dc i(m1)
.op

.end

```

خط اول شامل اسم برنامه می باشد که دلخواه است میتوان حتی از ***** استفاده کرد (دقت شود که نمیتوان از المان های مدار برای اسم برنامه استفاده کرد، مثلا nmos اسم مناسبی نیست)

فراخوانی کتابخانه:

برای فراخوانی کتابخانه از دستور زیر استفاده میشود:

.lib 'اسم کتابخانه' tt or SS or FF or FS or SF

هریک از tt و.. کاربرد خود را دارند و معمولا برای تحلیل کرنر ترانزیستورها استفاده میشود.(نکته:حتما باید کتابخانه درمسیری که برنامه نوشته شده کپی شود)

برای مثال tt: برای ترانزیستورهای معمولی باسرعت متوسط استفاده میشود و ss هم nmos و هم pmos دارای سرعت کم هستند، ff برای سرعت بالا و sf,fs یکی سرعت بالا و یکی سرعت پایین است(منظور nmos و pmos است)

با نوشتن دستور **protect,unprotect**. کتابخانه در فایل lis. نمایش داده نخواهد شد.

دستور option post::

بالین دستور میتوان نوع خروجی که در Scope نمایش داده خواهد شد را تعیین کرد (توجه: استفاده از این دستور در hspice2014 ضروری است)

این دستور میتواند مقادیر ۰ و ۱ و ۲ و ۳ را به خود اختصاص دهد.

۰: بدون خروجی (خروجی نمایش داده نخواهد شد) ۱: خروجی باینری (برای محاسبه slew rate مناسب است)

۲: ASCII ۳: new wave binary

دستور param::

از این دستور برای تعریف پارامتر استفاده میشود :

مقدار پارامتر = اسم پارامتر param.

که در اینجا منبع ولتاژ مدار به صورت پارامتر تعریف شده است.

معرفی ترانزیستور:

m {Model} w= l= Drain Gate Source Bulk (نام ترانزیستور)

منظور از درین و گیت و... اسم گره هایی است که برای درین و گیت و... ترانزیستور گذاشته ایم.

*معرفی المان دوسر:

value Node- Node+ (اسم ترانزیستور) r

برای خازن نیز به همین صورت است فقط اول C مینویسیم.

معرفی منابع ولتاژ DC :

value dc Node- Node+ (اسم منبع) V

تحلیل dc :

اگر بخواهیم تحلیل برای یک منبع ولتاژ انجام دهیم

step end start اسم منبع ولتاژ .dc

دستور sweep:

برای رسم یک پارامتر یا ولتاژ برحسب تغییرات ولتاژ یک گره معمولاً از این دستور استفاده میشود، توجه شود: زمانی که از این دستور استفاده میشود باید کلیه تحلیل های dc ستاره دار و غیرفعال شوند.

step end start param name sweep

دستور **probe**:

با نوشتن این دستور میتوان مشخصات المان دلخواه را در خروجی **scope** دید.

probe analysis type v,i,p(eleman name)

با این دستور میتوان شکل موج ولتاژ، جریان یا توان یک المان خاص یا یک گره را در خروجی دید، که در نمونه برنامه از این دستور برای نمایش جریان **dc** ترانزیستور **m1** استفاده شده است.

دستور **op**:

از این دستور برای تحلیل نقاط کار و نمایش آن در فایل **lis**. استفاده میشود.

دستور **end**:

در پایان هر برنامه حتما این دستور باید نوشته شود.

بعد از نوشتن برنامه و ذخیره آن ، از **hspice** ، **open** میکنیم و بعد **simulate** را میزنیم، بعد از انجام شبیه سازی **Edit LL** را میزنیم که برنامه **error** نداشته باشد.

بعد از انجام این کارها و برطرف کردن **error** های برنامه ، **cscope** را میزنیم و **open** را میزنیم.

برای مشاهده خروجی تحلیل **dc** از فایل با پسوند **.sw0** ، خروجی **ac** پسوند **.ac0** و تحلیل **tran** از فایل **.tr0** استفاده میکنیم که در ادامه این تحلیل ها گفته خواهد شد.