

1. Simulação de amplificador

Os parâmetros S de um amplificador GaAs MESFET ($Z_0=50\Omega$) são:

$f(\text{GHz})$	S_{11}	S_{12}	S_{21}	S_{22}
3.0	$0.80\angle -89^\circ$	$0.03\angle 56^\circ$	$2.86\angle 99^\circ$	$0.76\angle -41^\circ$
4.0	$0.72\angle -116^\circ$	$0.03\angle 57^\circ$	$2.60\angle 76^\circ$	$0.73\angle -54^\circ$
5.0	$0.66\angle -142^\circ$	$0.03\angle 62^\circ$	$2.39\angle 54^\circ$	$0.72\angle -68^\circ$

Temos que em 3 GHz o amplificador tem estabilidade condicional porque $K < 1$ e $\mu < 1$:

```

1 delta = s11*s22 - s12*s21;
2 K = (1 - (abs(s11)**2) - (abs(s22)**2) + (abs(delta)**2))/(2*abs(s12*s21));
3 mu = (1-abs(s11)**2)/(abs(s22 - (delta*s11.conjugate())) + abs(s12*s21));

print(abs(delta) , "< 1")
print(K , "> 1")
print(mu , "> 1")

0.5916268455125475 < 1
0.771691866731513 > 1
0.9360041977016981 > 1

```

Já em 4 GHz temos que o amplificador tem estabilidade incondicional porque atendeu os critérios abaixo:

```

1 delta = s11*s22 - s12*s21;
2 K = (1 - (abs(s11)**2) - (abs(s22)**2) + (abs(delta)**2))/(2*abs(s12*s21));
3 mu = (1-abs(s11)**2)/(abs(s22 - (delta*s11.conjugate())) + abs(s12*s21));

print(abs(delta) , "< 1")
print(K , "> 1")
print(mu , "> 1")

0.48752686574033216 < 1
1.1947592616576408 > 1
1.0400684251172099 > 1

```

Por último, em 5 GHz temos que o amplificador tem estabilidade incondicional porque atendeu os critérios abaixo:

```

1 delta = s11*s22 - s12*s21;
2 K = (1 - (abs(s11)**2) - (abs(s22)**2) + (abs(delta)**2))/(2*abs(s12*s21));
3 mu = (1-abs(s11)**2)/(abs(s22 - (delta*s11.conjugate())) + abs(s12*s21));

print(abs(delta) , "< 1")
print(K , "> 1")
print(mu , "> 1")

0.4176867952758771 < 1
1.5373937165120812 > 1
1.0893970391088454 > 1

```

Como na frequência de 4 GHz o amplificador tem estabilidade incondicional é necessário que $\Gamma_{in} < 1$ e $\Gamma_{out} < 1$ para TODOS ZS e ZL, logo abaixo encontra-se o cálculo do Γ_L e Γ_s e do ganho de transdutância máximo.

```
0 s 1 B1 = 1 + abs(s11)**2 - abs(s22)**2 - abs(delta)**2
2 B2 = 1 + abs(s22)**2 - abs(s11)**2 - abs(delta)**2
3 C1 = s11 - delta*s22.conjugate()
4 C2 = s22 - delta*s11.conjugate()
5 Ts1 = (B1 + cmath.sqrt(B1**2 - 4*(abs(C1))**2))/(2*C1); print(abs(Ts1))
6 Ts2 = (B1 - cmath.sqrt(B1**2 - 4*(abs(C1))**2))/(2*C1); print(abs(Ts2))
7 Tl1 = (B2 + cmath.sqrt(B2**2 - 4*(abs(C2))**2))/(2*C2); print(abs(Tl1))
8 Tl2 = (B2 - cmath.sqrt(B2**2 - 4*(abs(C2))**2))/(2*C2); print(abs(Tl2))

1.1471054160629666
0.871759461682385
1.141173709496366
0.8762907799911809

1 GTmax = ((abs(s21)**2) * (1-abs(Tl2)**2)) / ((1 - abs(Ts2)**2) * (abs(1 - s22*Tl2))**2); print('\n',GTmax)

46.88345061751798
```