



Amplifier Design

จัดทำโดย

นายธรรมราช	งามอยู่	รหัสนักศึกษา	62070502431
นายปัญญากร	อริกะ	รหัสนักศึกษา	62070502439
นางสาวพิมพ์นารา	เริงสำราญ	รหัสนักศึกษา	62070502478

เสนอ

ผศ.ดร. ธอริน อีร์เตชวานิชกุล

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชา

ENE450 Applied Communications and Transmission Lines

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์และโทรคมนาคม คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

Problem 10: Design an amplifier for maximum gain at 2.5 GHz using single-stub matching sections. (Use short-circuited shunt stub for input matching network and open - circuited shunt stub for the output network.) The transistor is NE46134 NPN microwave transistor with the following scattering parameter ($Z_0 = 50\Omega$): $S_{11} = 0.314\angle 57.3^\circ$, $S_{21} = 1.109\angle 11.7^\circ$, $S_{12} = 0.482\angle 28^\circ$, $S_{22} = 0.423\angle -135.9^\circ$ Also, determine the transducer power gain. Draw the schematic and the microstrip-line layout of the circuit. Check the stability and the transducer power gain of the circuit at 1 GHz, given scattering parameters at 1 GHz: $S_{11} = 0.352\angle 132.4^\circ$, $S_{21} = 2.254\angle 54.4^\circ$, $S_{12} = 0.239\angle 59.2^\circ$, $S_{22} = 0.210\angle -124.2^\circ$

วัตถุประสงค์การทดลอง

1. เพื่อศึกษาการออกแบบวงจร Amplifier ด้วยการใช้ single-stub matching sections
2. เพื่อใช้ค่า Scattering ของทรานซิสเตอร์มาทำการคำนวณสำหรับการออกแบบวงจร
3. เพื่อใช้โปรแกรม MATLAB ในการคำนวณและตรวจสอบ Stability ของวงจรทั้งทางฝั่ง input และ output

ขั้นตอนการทดลอง

1. หาค่า $|\Delta|$ และ K เพื่อตรวจสอบความเสถียรภาพ (Stability)

$$|\Delta| = |s_{11}s_{22} - s_{12}s_{21}| = |(0.314\angle 57.3^\circ)(0.423\angle -135.9^\circ) - (0.239\angle 59.2^\circ)(2.254\angle 54.4^\circ)|$$

$$|\Delta| = 0.609$$

$$k = \frac{1 - |S_{11}|^2 - |S_{22}|^2 + |\Delta|^2}{2|S_{12}S_{21}|} = \frac{1 - |0.314|^2 - |0.423|^2 + |0.609|^2}{2|(0.482)(1.109)|}$$

$$k = 1.023$$

ผลจากการคำนวณบนโปรแกรม MATLAB

```
S_abs = 1.109000e+00
--Step 1: K-Delta stability test
Delta: |Delta| = 0.608844, deg of Delta = -129.225812
K = 1.02253
Unconditionally stable
```

2. คำนวณหาค่า Γ_s

จากการคำนวณ $\Gamma_s = \frac{B_1 \pm \sqrt{B_1^2 - 4|C_1|^2}}{2C_1}$

$$\Delta = S_{11}S_{22} - S_{12}S_{21}$$

$$\Delta = (0.314 \angle 57.3^\circ)(0.423 \angle -135.9^\circ) - (0.239 \angle 59.2^\circ)(2.254 \angle 54.4^\circ)$$

$$\Delta = 0.609 \angle -129.225^\circ$$

หาค่าของ B_1 จากสมการ $B_1 = 1 + |S_{11}|^2 - |S_{22}|^2 - |\Delta|^2$

$$B_1 = 1 + |0.314 \angle 57.3^\circ|^2 - |0.423 \angle -135.9^\circ|^2 - |0.609 \angle -129.225^\circ|^2$$

$$B_1 = 0.548786$$

หาค่าของ C_1 จากสมการ $C_1 = S_{11} - \Delta S_{22}^*$

$$C_1 = 0.314 \angle 57.3^\circ - (0.609 \angle -129.225^\circ)(0.423 \angle -135.9^\circ)^*$$

$$C_1 = 0.25 \angle 110.205^\circ$$

$$|C_1| = 0.2493030$$

นำค่า B_1 และ C_1 ไปแทนในสมการ Γ_s

$$\Gamma_s = \frac{0.548786 \pm \sqrt{0.548786^2 - 4|0.2493030|^2}}{2(0.25 \angle 110.205^\circ)}$$

$$\Gamma_s = 1.55608 \angle -110.205^\circ, 0.63906 \angle -110.205^\circ$$

ผลจากการคำนวณบนโปรแกรม MATLAB

```
--Step 2: GammaS
GammaS: |GammaS| = 1.556644, deg of GammaS = -110.190098
GammaS: |GammaS| = 0.642408, deg of GammaS = -110.190098
choose the 2nd solution because it is inside the input stability region
gammaS = -0.221718
```

จะสามารถสรุปได้ว่า ต้องเลือกใช้ค่า $\Gamma_s = 0.63906 \angle -110.205^\circ$ เพราะหาค่าของ Γ_s เป็นค่าที่อยู่ใน Input stability region มีขนาดค่าน้อยกว่า 1 ในการใช้คำนวณขั้นตอนต่อไป

3. คำนวณหาค่า impedance Z_s

สามารถคำนวณได้ตามสมการ

$$Z_s = Z_0 \left(\frac{1+\Gamma_s}{1-\Gamma_s} \right) = 50 \left(\frac{1+0.63906\angle -110.205^\circ}{1-0.63906\angle -110.205^\circ} \right) = 36.131 \angle -64.032^\circ \Omega$$

ผลจากการคำนวณบนโปรแกรม MATLAB

```
--Step 3: Zs
Zs = 15.820933 + j(-32.483476)
|Zs| = 36.131401, deg of Zs = -64.031808
```

4. ออกแบบวงจร input matching

ผลจากการคำนวณบนโปรแกรม MATLAB เพื่อใช้สำหรับออกแบบ short-circuited shunt stub tuner for input matching network

```
--Step 4: input matching circuit
[Single-stub shunt tuner] 2 solution(s):
Solution #1
Distance of the stub: d/lambda = 0.333558
Short circuit: ls/lambda = 0.0855975
Open circuit: lo/lambda = 0.335598

Solution #2
Distance of the stub: d/lambda = 0.472526
Short circuit: ls/lambda = 0.414402
Open circuit: lo/lambda = 0.164402
```

กราฟที่ได้จากโปรแกรม MATLAB

กราฟความสัมพันธ์ระหว่าง ขนาดสัมประสิทธิ์การสะท้อน $|\Gamma|$ เมื่อเทียบกับความถี่ของทั้ง Solution 1 และ Solution 2 เพื่อเปรียบเทียบ Fractional Bandwidth

Input: Short – circuit

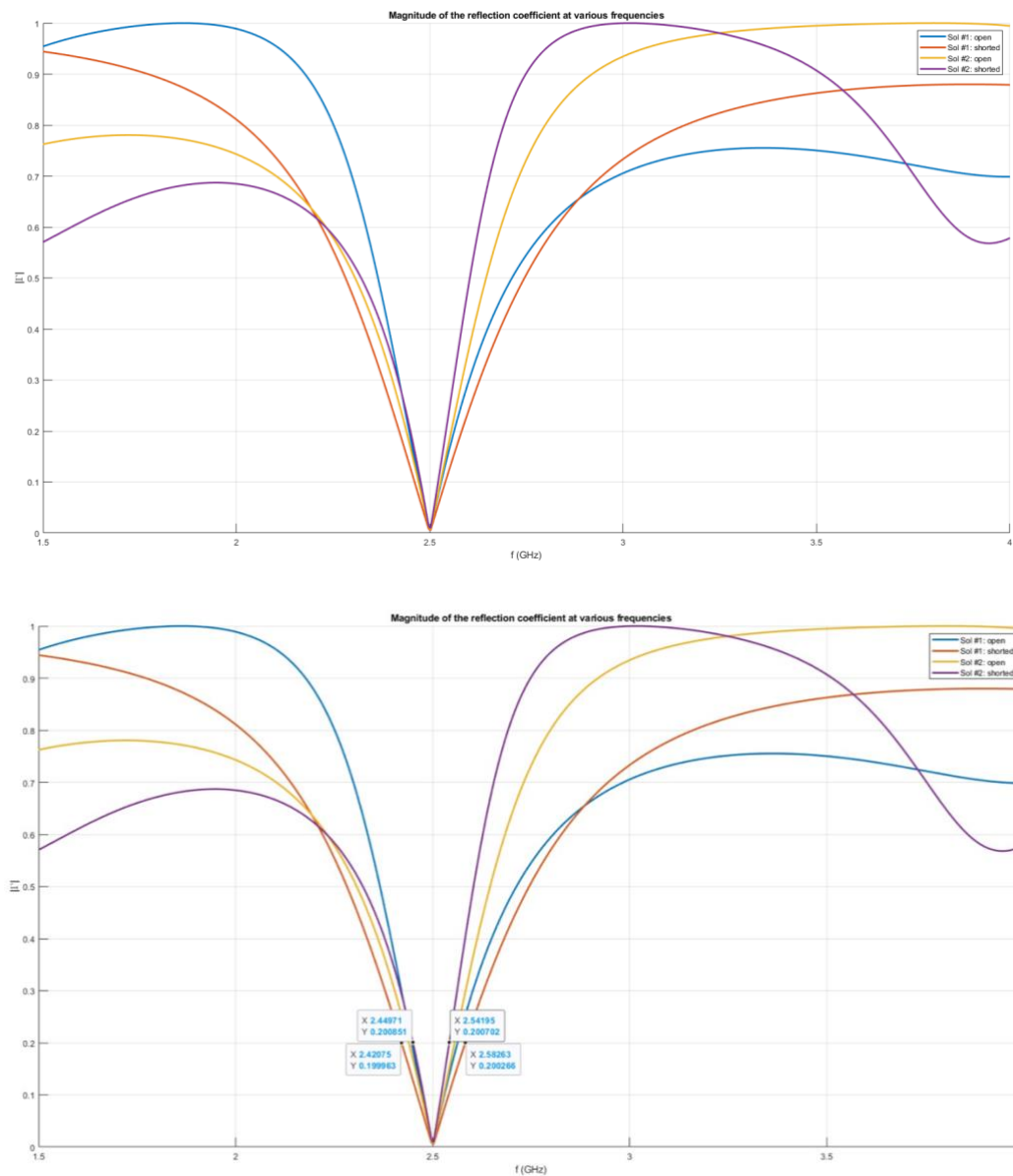


figure 1 กราฟแสดงความสัมพันธ์ระหว่าง $|\Gamma|$ กับความถี่

Table 1 ตารางแสดงค่า $\frac{d}{\lambda}$, $\frac{l_s}{\lambda}$, และ $\frac{l_o}{\lambda}$ ที่คำนวณได้จากโปรแกรม MATLAB

Parameters	Solution 1	Solution 2
$\frac{d}{\lambda}$	0.333558	0.472526
$\frac{l_s}{\lambda}$	0.0855975	0.414402
$\frac{l_o}{\lambda}$	0.335598	0.164402

Table 2 ตารางแสดงค่า f_{\max} (GHz) และ f_{\min} (GHz) ที่ตำแหน่ง $|\Gamma| = 0.2$

	Solution 1		Solution 2	
	f_{\max} (GHz)	f_{\min} (GHz)	f_{\max} (GHz)	f_{\min} (GHz)
short - circuit	2.58263	2.42075	2.54195	2.44971

จากการ Design input matching circuit โดยเราเลือกใช้ short – circuited shunt stub tuner ประกอบไปด้วย 2 Solution พบว่าทั้งสอง Solution มีความกว้างของ Bandwidth ใกล้เคียงกันที่ $|\Gamma| = 0.2$ นั้น เราจะสามารถหาค่า fractional bandwidth (FBW) ของทั้งสอง Solution ได้จากสมการ $FBW = \frac{f_{\max} - f_{\min}}{f_c}$ เมื่อค่า $f_c = 2.5$ GHz ดังนี้

Table 3 ตารางแสดงการคำนวณค่า fractional bandwidth (FBW)

	Solution 1	Solution 2
short - circuit	$\frac{2.58263 - 2.42075}{2.5} = 0.064752$	$\frac{2.54195 - 2.44971}{2.5} = 0.036896$

Table 4 ตารางแสดงค่าร้อยละของ fractional bandwidth (FBW)

	Solution 1	Solution 2
short - circuit	6.4752 %	3.6896 %

จะสังเกตเห็นว่าค่า Fractional Bandwidth ของ Solution 1 นั้นมีค่ามากกว่าของ Solution 2 จึงเลือกใช้ Solution 1 ที่มีค่า $\frac{d}{\lambda} = 0.333558$ และมีค่า $\frac{l_s}{\lambda} = 0.0855975$ ซึ่งเป็นวิธีที่มีค่า FBW สูงที่สุด คือ 0.064752 หรือคิดเป็นร้อยละ 6.4752 สามารถวาดเป็นรูปวงจรได้ ดังนี้

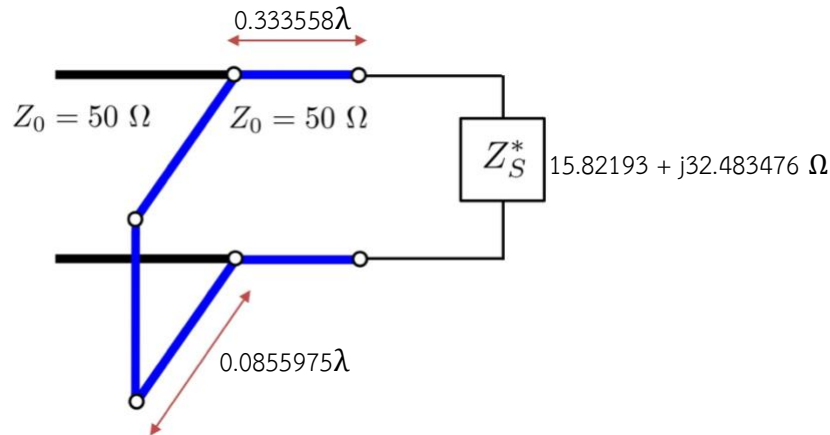


figure 2

ทำการแทนที่ input matching circuit แบบ short-circuited shunt stub

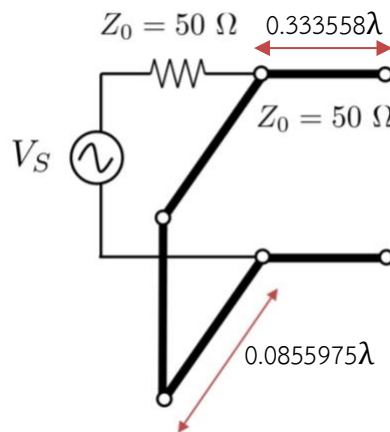


figure 3

5. คำนวณหาค่าของ Γ_L

$$\text{จากการคำนวณ } \Gamma_L = \frac{B_2 + \sqrt{B_2^2 - 4|C_2|^2}}{2C_2}$$

$$\Delta = 0.609 \angle -129.225^\circ$$

$$\text{จากการคำนวณ } B_2 = 1 + |S_{22}|^2 - |S_{11}|^2 - |\Delta|^2$$

$$B_1 = 1 + |0.423 \angle -135.9^\circ|^2 - |0.314 \angle 57.3^\circ|^2 - |0.609 \angle -129.225^\circ|^2$$

$$B_2 = 0.709452$$

$$\text{จากการคำนวณ } C_2 = S_{22} - \Delta S_{11}^*$$

$$C_2 = 0.423 \angle -135.9^\circ - (0.609 \angle -129.225^\circ)(0.314 \angle 57.3^\circ)^*$$

$$|C_2| = 0.335955$$

$$C_2 = 0.335955 \angle -109.796^\circ$$

นำค่า B_2 และ C_2 ไปแทนในสมการ Γ_L

$$\Gamma_L = \frac{0.709452 \pm \sqrt{0.548786^2 - 4|0.335955|^2}}{2(0.335955 \angle -109.796^\circ)}$$

$$\Gamma_L = 1.39479 \angle 109.2796^\circ, 0.71695 \angle 109.796^\circ$$

ผลจากการคำนวณบนโปรแกรม MATLAB

```
--Step 5: GammaL
GammaL: |GammaL| = 1.395772, deg of GammaL = 109.804247
GammaL: |GammaL| = 0.716449, deg of GammaL = 109.804247
choose the 2nd solution because it is inside the input stability region
```

สามารถสรุปได้ว่า ต้องเลือกใช้ค่า $\Gamma_L = 0.71695 \angle 109.796^\circ$ เพราะหาค่าของ Γ_L เป็นค่าที่อยู่ใน input stability region ซึ่งต้องมีค่าน้อยกว่า 1 เพื่อใช้ในการคำนวณในขั้นตอนต่อไป

6. คำนวณหาค่าของ Impedance Z_L

จากการคำนวณโดยใช้สมการ $Z_L = Z_0 \left(\frac{1+\Gamma_L}{1-\Gamma_L} \right)$

$$Z_L = 50 \left(\frac{1 + 0.71695 \angle 109.796^\circ}{1 - 0.71695 \angle 109.796^\circ} \right) = 12.149 + j33.73 \, \Omega = 35.85 \angle 70.19^\circ \, \Omega$$

ผลจากการคำนวณบนโปรแกรม MATLAB

```
--Step 6: Z1
Z1 = 12.174958 + j(33.724398)
|Z1| = 35.854771, deg of Z1 = 70.149741
```

7. ออกแบบวงจร output matching

ผลจากการคำนวณบนโปรแกรม MATLAB เพื่อใช้สำหรับออกแบบ short-circuited shunt stub tuner for input matching network

```
--Step 7: output matching circuit
[Single-stub shunt tuner] 2 solution(s):
Solution #1
Distance of the stub: d/lambda = 0.0360526
Short circuit: ls/lambda = 0.0721116
Open circuit: lo/lambda = 0.322112

Solution #2
Distance of the stub: d/lambda = 0.158936
Short circuit: ls/lambda = 0.427888
Open circuit: lo/lambda = 0.177888
```

กราฟที่ได้จากโปรแกรม MATLAB

กราฟความสัมพันธ์ระหว่าง ขนาดสัมประสิทธิ์การสะท้อน $|\Gamma|$ เมื่อเทียบกับความถี่ของทั้ง Solution 1 และ Solution 2 เพื่อเปรียบเทียบ Fractional Bandwidth

Output: Open – circuit

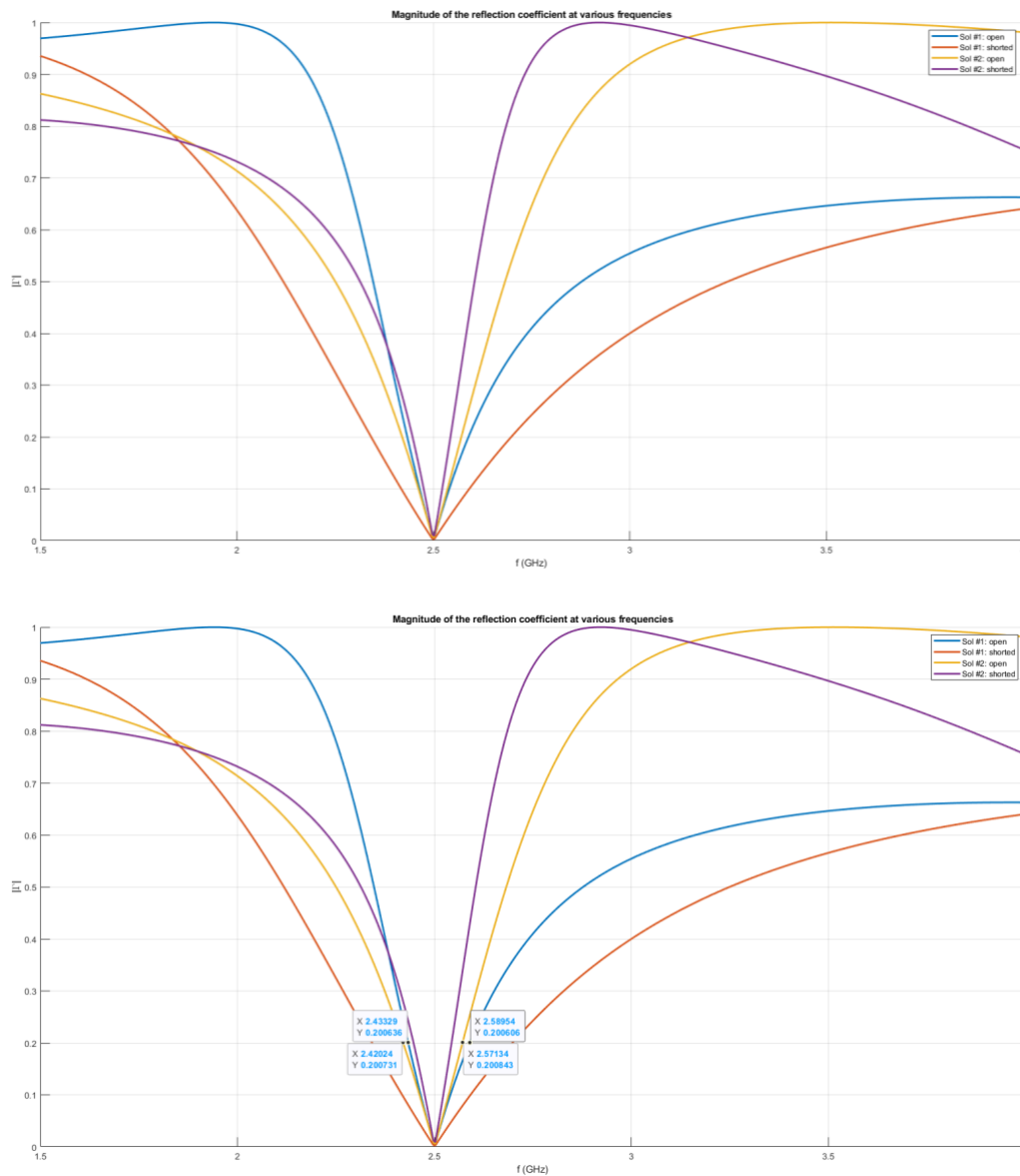


figure 4 กราฟแสดงความสัมพันธ์ระหว่าง $|\Gamma|$ กับความถี่

Table 5 ตารางแสดงค่า $\frac{d}{\lambda}$, $\frac{l_s}{\lambda}$ และ $\frac{l_o}{\lambda}$ ที่คำนวณได้จากโปรแกรม MATLAB

Parameters	Solution 1	Solution 2
$\frac{d}{\lambda}$	0.0360526	0.158936
$\frac{l_s}{\lambda}$	0.0721116	0.427888
$\frac{l_o}{\lambda}$	0.322112	0.177888

Table 6 ตารางแสดงค่า f_{\max} (GHz) และ f_{\min} (GHz) ที่ตำแหน่ง $|\Gamma| = 0.2$

	Solution 1		Solution 2	
	f_{\max} (GHz)	f_{\min} (GHz)	f_{\max} (GHz)	f_{\min} (GHz)
open - circuit	2.58954	2.43329	2.57134	2.43329

จากการ Design output matching circuit โดยเราเลือกใช้ open – circuited shunt stub tuner ประกอบไปด้วย 2 Solution พบว่าทั้งสอง Solution มีความกว้างของ Bandwidth ใกล้เคียงกันที่ $|\Gamma| = 0.2$ นั้น เราจะสามารถหาค่า fractional bandwidth (FBW) ของทั้งสอง Solution ได้จากสมการ $FBW = \frac{f_{\max} - f_{\min}}{f_c}$ เมื่อค่า $f_c = 2.5$ GHz ดังนี้

Table 7 ตารางแสดงการคำนวณค่า fractional bandwidth (FBW)

	Solution 1	Solution 2
open - circuit	$\frac{2.58954 - 2.43329}{2.5} = 0.0625$	$\frac{2.57134 - 2.42024}{2.5} = 0.06044$

Table 8 ตารางแสดงค่าร้อยละของ fractional bandwidth (FBW)

	Solution 1	Solution 2
open - circuit	6.25 %	6.044 %

จะสังเกตเห็นว่าค่า Fractional Bandwidth ของทั้งสอง Solution มีความใกล้เคียงกันมาก เราจึงจำเป็นต้องพิจารณาที่ความยาวของเส้นสายส่ง ซึ่งจาก Table 5 จะพบว่า Solution 2 มีความยาวของเส้นสายส่งที่สั้นกว่า Solution 1 ดังนั้นจึงเลือกใช้ Solution 2 ที่มีค่า $\frac{d}{\lambda} = 0.158936$ และมีค่า $\frac{l_0}{\lambda} = 0.177888$ มีค่า FBW เท่ากับ 0.06044 หรือคิดเป็นร้อยละ 6.044 สามารถวาดเป็นรูปวงจรได้ ดังนี้

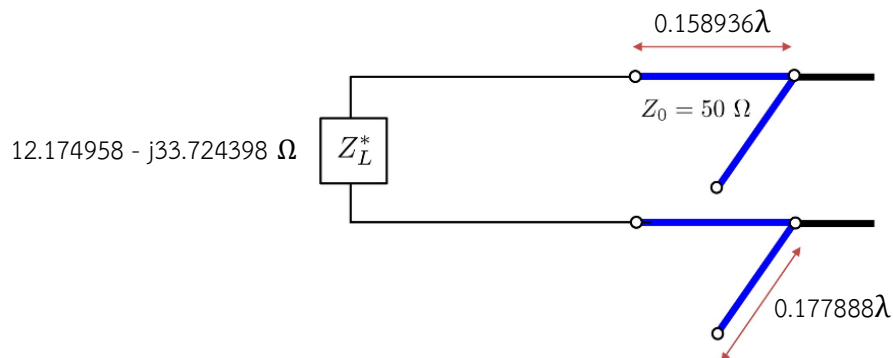


figure 5

ทำการแทนที่ output matching circuit แบบ open - circuited shunt stub

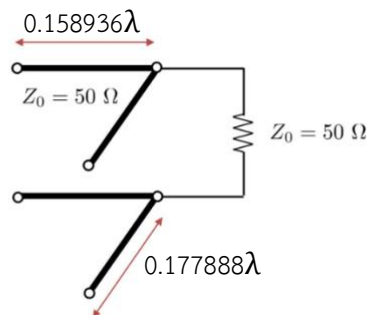


figure 6

8. ตรวจสอบความเสถียรภาพ (Stability) ของวงจร

ผลจากการคำนวณบนโปรแกรม MATLAB

```
--Step 8: Careful check of stability
The input port of transistor is...stable
The output port of transistor is...stable
```

9. คำนวณหาค่า Transducer power gain ของ Amplifier

คำนวณหาค่าของ G_s จากสมการ $G_s = \frac{1}{1-|\Gamma_s|^2}$

$$G_s = \frac{1}{1-|0.63906|^2} = 1.6903 = 2.2797 \text{ dB}$$

คำนวณหาค่าของ G_0 จากสมการ $G_0 = |S_{21}|^2$

$$G_0 = |1.109|^2 = 1.22988 = 0.8986 \text{ dB}$$

คำนวณหาค่าของ G_L จากสมการ $G_L = \frac{1-|\Gamma_s|^2}{|1-S_{22}\Gamma_L|^2}$

$$G_L = \frac{1-|0.71695|^2}{|1-(0.423)(0.71695)|^2} = 0.8879 = -0.5163 \text{ dB}$$

คำนวณหาค่าของ G_{Tmax} จากสมการ $G_s = \frac{1}{1-|\Gamma_s|^2} |S_{21}|^2 \frac{1-|\Gamma_s|^2}{|1-S_{22}\Gamma_L|^2}$

$$G_{Tmax} = 1.6903 \times 1.22988 \times 0.8879 = 1.8458 = 2.6618 \text{ dB}$$

เมื่อนำผลลัพธ์ของ gain ทั้ง Source, Load และ G ในหน่วย dB จากการคำนวณด้วย MATLAB มาบวก

กันจะได้เป็น Transduction power gain เป็น 2.6618 dB

ผลจากการคำนวณบนโปรแกรม MATLAB

```
--Step 9: Gains
Gs = 1.702672 = 2.311310 dB
G0 = 1.229881 = 0.898631 dB
GL = 0.888925 = -0.511347 dB
GTmax = 1.861484 = 2.698594 dB
```

Table 9 ตารางแสดงค่า Gain ต่าง ๆ

G_s	G_0	G_L	G_{Tmax}
1.702672	1.229881	0.888925	1.861484
2.311310 dB	0.898631 dB	-0.511347 dB	2.698594 dB

Draw the schematic and the microstrip-line layout of the circuit

The schematic

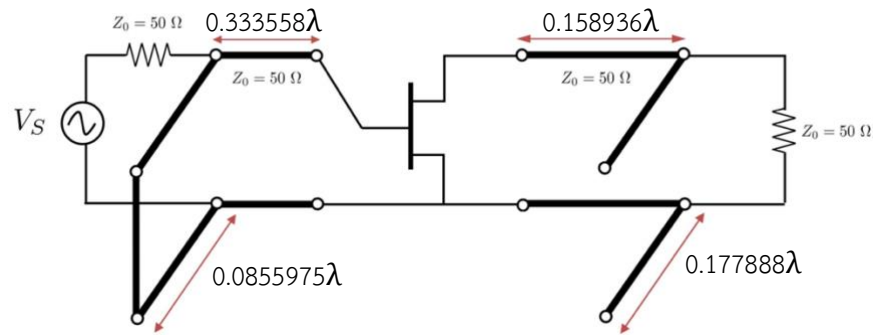


figure 7 The schematic

The microstrip – line layout

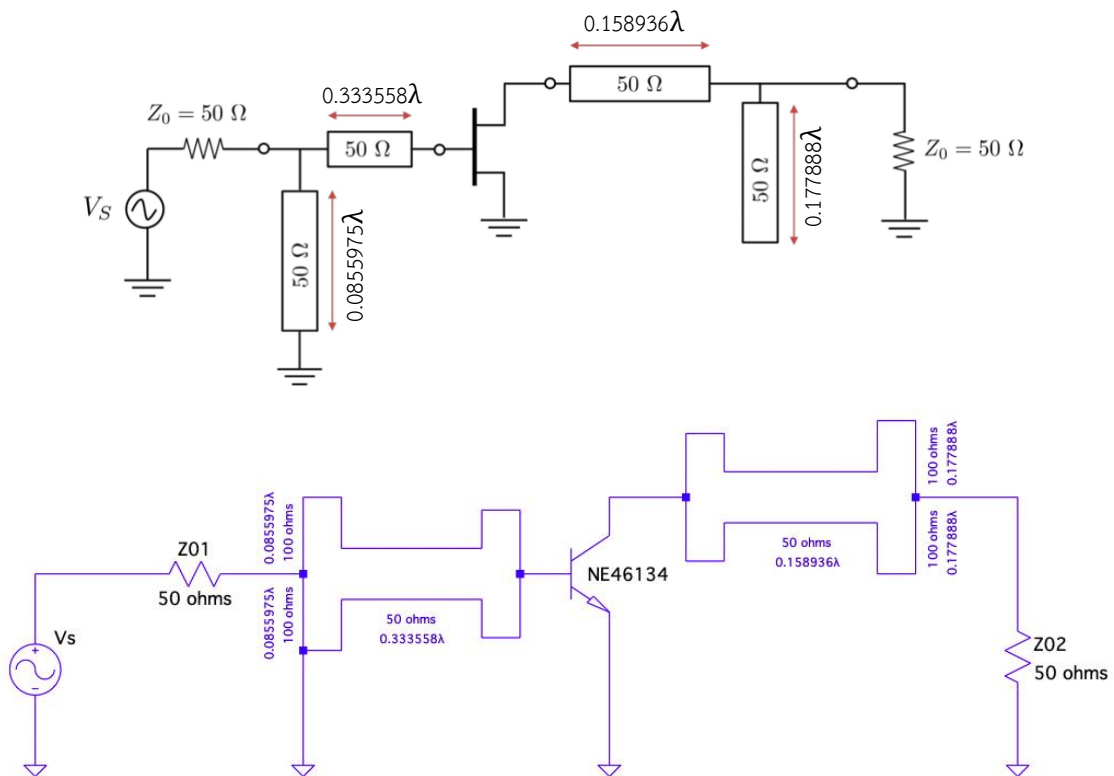


figure 8 The microstrip – line layout

Check the stability and the transducer power gain of the circuit at 1 GHz

จะทำการตรวจสอบว่า stability ในความถี่อื่นด้วย โดยการจะเช็คค่าที่ความถี่ของ 1 GHz มีการนำค่าของ Scattering ของความถี่นั้นมาทำการคำนวณ โดยการ

1. หาค่าสัมประสิทธิ์การสะท้อนที่ source (Γ_s) และ Γ_{out} ที่ความถี่เท่ากับ 1 GHz

ผลจากการคำนวณค่าสัมประสิทธิ์การสะท้อน โดยใช้โปรแกรม MATLAB

```
Za is 0.000000+10.925570j ohm
Zeq_in is 2.278566+10.427677j ohm
Zs is 8.598982+85.385289j ohm
Gamma_s is 0.916318exp(j60.328872)
Gamma_out is 0.501318exp(j-166.860249)
```

1.1 คำนวณหาค่า Z_A

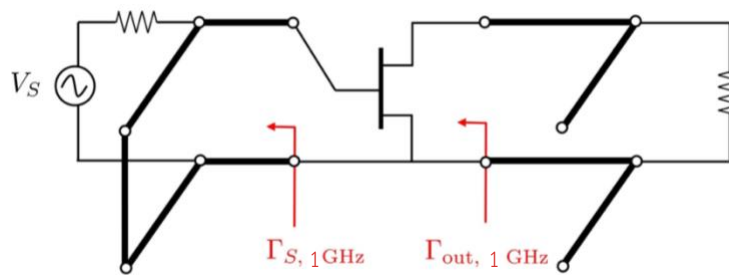


figure 9

- 1.2 เราสามารถคำนวณได้จากสมการ $Z_A = jZ_0 \tan(\beta l)$ เพราะเป็น short-circuited shunt stub โดยจะได้ว่า $Z_A = j10.926 \Omega$

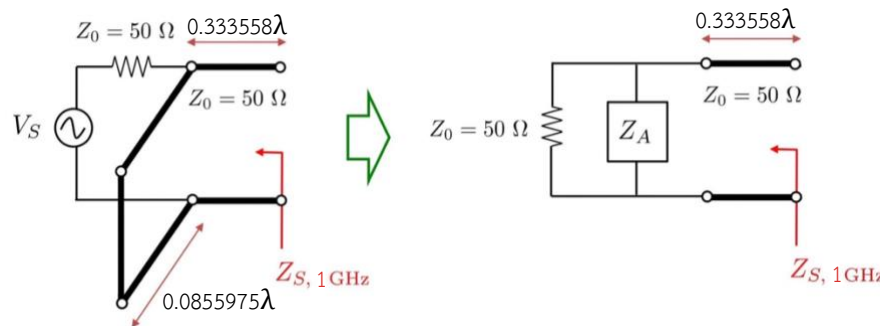


figure 10

1.3 คำนวณหาค่าของ Z_{eq}

เราจะสามารถคำนวณหาค่าของ Z_{eq} ได้จากสมการ $Z_{eq} = Z_0 \parallel Z_A = \frac{Z_0 Z_A}{Z_0 + Z_A}$ โดยเราจะได้ว่า
 $Z_{eq} = 2.279 + j10.428 \Omega$

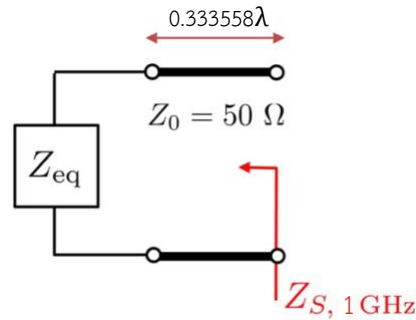


figure 11

1.4 คำนวณหาค่าของ $Z_{S,1\text{GHz}}$

เราสามารถคำนวณได้จากสมการ $Z_{S,1\text{GHz}} = \frac{Z_{eq} + jZ_0 \tan(\beta d)}{Z_0 + Z_{eq} \tan(\beta d)} = 8.599 + j85.385 \Omega$

1.5 หาค่าสัมประสิทธิ์การสะท้อนที่ load (Γ_s) และ Γ_{out} ที่ความถี่เท่ากับ 1 GHz

$$\Gamma_s = \frac{Z_{S,1\text{GHz}} - Z_0}{Z_{S,1\text{GHz}} + Z_0} = 0.916 \angle 60.329^\circ$$

$$\Gamma_{out} = S_{22} + \frac{S_{12}S_{21}\Gamma_s}{1 - S_{11}\Gamma_s} = 0.501 \angle -166.86^\circ$$

2. หาค่าสัมประสิทธิ์การสะท้อนที่ load (Γ_L) และ Γ_{in} ที่ความถี่เท่ากับ 1 GHz

ผลจากการคำนวณค่าสัมประสิทธิ์การสะท้อน โดยใช้โปรแกรม MATLAB

```
Zb is 0.000000-104.283904j ohm
Zeq_out is 40.654298-19.492125j ohm
ZL is 32.495215-8.190615j ohm
Gamma_L is 0.233125exp(j-149.254683)
Gamma_in is 0.232557exp(j124.482714)
```

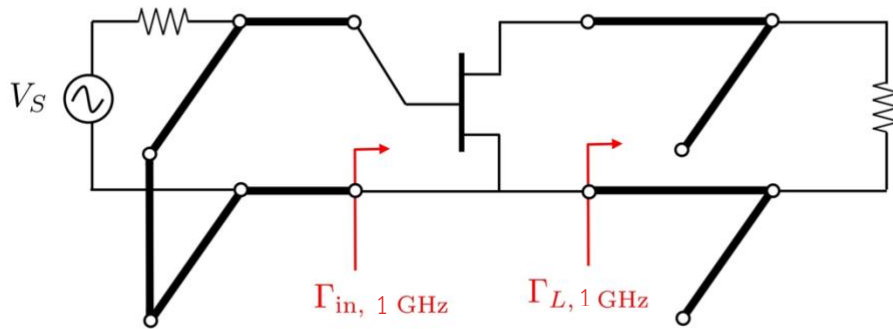



figure 12

2.1 คำนวณหาค่าของ Z_B

เราสามารถคำนวณได้จากสมการ $Z_B = -jZ_0 \cot(\beta l)$ เพราะเป็น open - circuited shunt stub
จะได้ว่า $Z_B = -j104.284 \, \Omega$

2.2 คำนวณหาค่าของ Z_{eq}

เราสามารถคำนวณได้จากสมการ $Z_{eq} = Z_0 \parallel Z_A = \frac{Z_0 Z_B}{Z_0 + Z_B} = 40.654 - j19.492 \, \Omega$

2.3 คำนวณหาค่าของ Z_L

เราสามารถคำนวณได้จากสมการ $Z_{L,1\text{GHz}} = \frac{Z_{eq} + jZ_0 \tan(\beta d)}{Z_0 + Z_{eq} \tan(\beta d)} = 32.495 - j8.191 \, \Omega$

2.4 หาค่าสัมประสิทธิ์การสะท้อนที่ load (Γ_L) และ Γ_{in} ที่ความถี่เท่ากับ 1 GHz

$$\Gamma_L = \frac{Z_{L,1\text{GHz}} - Z_0}{Z_{L,1\text{GHz}} + Z_0} = 0.233 \angle -149.255^\circ$$

$$\Gamma_{in} = s_{11} + \frac{s_{12}s_{21}\Gamma_L}{1 - s_{22}\Gamma_L} = 0.232 \angle 124.483^\circ$$

3. ทำการหาค่าของ Transducer power gain

ผลจากการคำนวณค่าของ Transducer power gain โดยโปรแกรม MATLAB คือ

```
Transducer power gain 0.532727
Transducer power gain in dB is -2.734949 dB
```

จากผลการคำนวณโดยใช้โปรแกรม MATLAB ในการหาค่าของ Transducer power gain มีค่าเท่ากับ 0.532727 หรือ -2.734949 dB ที่ความถี่ 1 GHz

Table 10 ตารางสรุปค่า parameter ต่าง ๆ ที่คำนวณได้ที่ความถี่ 2.5 GHz

Parameter	Value
Γ_s	$0.916 \angle 60.329^\circ$
Γ_{out}	$0.501 \angle -166.86^\circ$
Γ_{in}	$0.232 \angle 124.483^\circ$
Γ_L	$0.233 \angle -149.255^\circ$
G_T	-2.734949 dB

จากตารางข้างต้นได้ทำการสรุปรวมค่าต่าง ๆ สำหรับการตรวจสอบความเสถียรภาพ (stability) และค่าของ Transducer power gain พบว่ามีค่า $|\Gamma_s| < 1$ ที่มีผลทำให้ $\Gamma_{out} < 1$ ตามไปด้วยด้วย และมี $|\Gamma_L| < 1$ ที่มีผลทำให้ $\Gamma_{in} < 1$ ดังนั้นเราจึงสามารถสรุปได้ว่าที่ความถี่ 1 GHz นั้น Amplifier มีความเสถียรภาพ (stability) และ Transducer power gain เท่ากับ -0.532727 หรือ -2.734949 dB

สรุปผลการทดลอง (Conclusion)

จากการทดลองเรื่อง Amplifier design เพื่อใช้โปรแกรม MATLAB ในการคำนวณและตรวจสอบความเสถียรภาพ (Stability) ของวงจรทั้งทางฝั่ง input และ output ที่ความถี่ 2.5 GHz โดยใช้ transistor ชนิด NE46134 NPN วงจรที่ใช้ในการทดลองจะเป็น short-circuited shunt stub สำหรับ input matching network และ open - circuited shunt stub สำหรับ output matching network โดยเริ่มต้นเราทำการทดลองผ่านวิธี K- Δ Test ปรากฏว่าทรานซิสเตอร์ (Transistor) ในวงจรของเรานั้นมีความเสถียรอย่างไม่มีเงื่อนไข (Unconditionally stable) เนื่องจากค่า K ที่คำนวณได้จากโปรแกรม MATLAB มีค่าเท่ากับ 1.02253 ซึ่งมีค่ามากกว่า 1 และ $|\Delta|$ ที่คำนวณได้จากโปรแกรม MATLAB มีค่าเท่ากับ 0.608844 ซึ่งมีค่าน้อยกว่า 1 เป็นไปตามทฤษฎีของ Rollet's condition ในส่วนถัดมาจะทำการคำนวณค่า Γ_S โดยผลการคำนวณจากโปรแกรม MATLAB จะให้ค่า Γ_S ออกมา 2 ค่าเท่ากับ $1.556644 \angle -110.19^\circ$ และ $0.642408 \angle -110.19^\circ$ แต่เราต้องเลือกค่าที่อยู่ใน input stability region หรือเลือกผลที่ให้ค่า magnitude น้อยกว่า 1 ดังนั้นเราจึงต้องทำการเลือกใช้ค่า $\Gamma_S = 0.642408 \angle -110.19^\circ$ ในส่วนต่อมาจะทำการคำนวณค่า impedance Z_S ได้ออกมามีค่าเท่ากับ $36.131 \angle -64.032^\circ \Omega$ จากนั้นเราจะทำการออกแบบ input matching circuit โดยเราเลือกใช้วงจรเป็นแบบ short - circuited shunt stub tuner ประกอบไปด้วย 2 Solution เมื่อสังเกตจากกราฟความสัมพันธ์ระหว่างขนาดสัมประสิทธิ์การสะท้อน $|\Gamma|$ เมื่อเทียบกับความถี่ของทั้ง Solution 1 และ Solution 2 เพื่อเปรียบเทียบ Fractional Bandwidth พบว่าทั้งสอง Solution มีความกว้างของ Bandwidth ที่ $|\Gamma| = 0.2$ นั้นเราสามารถหาค่า fractional bandwidth (FBW) ของทั้งสอง Solution เพื่อเปรียบเทียบ ได้จากสมการ $FBW = \frac{f_{max} - f_{min}}{f_c}$ เมื่อค่าของ $f_c = 2.5$ GHz จะสังเกตเห็นว่าค่า Fractional Bandwidth ของ Solution 1 นั้นมีค่ามากกว่าของ Solution 2 จึงเลือกใช้ Solution 1 ที่มีค่า $\frac{d}{\lambda} = 0.333558$ และมีค่า $\frac{l_s}{\lambda} = 0.0855975$ ซึ่งเป็น Solution ที่มีค่า FBW สูงที่สุด คือ 0.064752 หรือคิดเป็นร้อยละ 6.4752 นำค่า d และ l_s ไปใช้ออกแบบวงจรในส่วนของ input matching circuit ถัดมาเราจะทำการคำนวณค่า Γ_L โดยผลการคำนวณจากโปรแกรม MATLAB จะให้ค่า Γ_S ออกมา 2 ค่าเท่ากับ $1.39479 \angle 109.804247^\circ$ และ $0.716449 \angle 109.804247^\circ$ ดังนั้นเราจึงจำเป็นต้องเลือกใช้ค่า $\Gamma_L = 0.716449 \angle 109.804247^\circ$ เพราะค่าของ Γ_L ต้องใช้ค่าที่อยู่ใน input stability region หรือต้องเลือกค่าที่ magnitude น้อยกว่า 1 แล้วนำค่าที่เราเลือกไปใช้คำนวณ impedance Z_L ได้ออกมามีค่าเท่ากับ $35.855 \angle 70.15^\circ \Omega$ ต่อมาเราจะทำการออกแบบ output matching circuit โดยเราได้เลือกใช้วงจรเป็นแบบ open - circuited shunt stub tuner ประกอบไปด้วย 2 Solution เมื่อสังเกตจากกราฟความสัมพันธ์ระหว่าง

ขนาดสัมประสิทธิ์การสะท้อน $|\Gamma|$ หากเทียบกับความถี่ของทั้ง Solution 1 และ Solution 2 เพื่อเปรียบเทียบ Fractional Bandwidth นั้นจะพบว่าทั้งสอง Solution มีความกว้างของ Bandwidth ใกล้เคียงกันที่ $|\Gamma| = 0.2$ เราจะสามารถหาค่า fractional bandwidth (FBW) ของทั้งสอง Solution ได้จากสมการ $FBW = \frac{f_{\max} - f_{\min}}{f_c}$ เมื่อค่า $f_c = 2.5$ GHz เนื่องจากค่า Fractional Bandwidth ของ Solution 1 นั้น มีค่าใกล้เคียงกันมากกับค่า Fractional Bandwidth ของ Solution 2 เราจึงจำเป็นต้องพิจารณาที่ความยาวของเส้นสายส่งแทน ซึ่งจาก Table 5 นั้นจะพบว่า Solution 2 มีความยาวของเส้นสายส่งที่สั้นกว่า Solution 1 ดังนั้นจึงเลือกใช้ Solution 2 ที่มีค่า $\frac{d}{\lambda} = 0.158936$ และมีค่า $\frac{l_0}{\lambda} = 0.177888$ มีค่า FBW เท่ากับ 0.06044 หรือคิดเป็นร้อยละ 6.044 จากนั้นนำค่า d และ l_0 ไปใช้ออกแบบวงจรในส่วนของ output matching circuit ในส่วนต่อไปเราจะนำวงจรที่เราออกแบบไปวิเคราะห์ด้วยโปรแกรม MATLAB เพื่อตรวจสอบความเสถียรของวงจร (Stability) พบว่าทั้ง input matching circuit และ output matching circuit มีความเสถียรภาพที่ความถี่ 2.5 GHz เพราะมีค่า $|\Gamma_s|$ และ $|\Gamma_L|$ ที่น้อยกว่า 1 ทั้งสองค่า ต่อมาจะทำการคำนวณหาค่า Transducer power gain ($G_{T_{\max}}$) ของ Amplifier ได้ค่าจากการโปรแกรม MATLAB มีค่าเท่ากับ 1.861484 หรือ 2.698594 dB และได้ค่า gain ต่าง ๆ ดัง Table 9

ในส่วนนี้จะเป็นการวิเคราะห์ผลการทดลอง เมื่อเราลองนำวงจรที่เราออกแบบได้ไปตรวจสอบความเสถียรที่ความถี่ 1 GHz โดยมีการนำค่าของ Scattering ของความถี่ที่เราต้องการตรวจสอบมาทำการคำนวณด้วยเช่นกัน เริ่มต้นเราจะคำนวณค่าสัมประสิทธิ์การสะท้อนที่ source (Γ_s) และ Γ_{out} ได้ผลลัพธ์จากการคำนวณด้วยโปรแกรม MATLAB มีค่า Γ_s เท่ากับ $0.916 \angle 60.329^\circ$ และค่า Γ_{out} เท่ากับ $0.501 \angle -166.86^\circ$ จากนั้นเราจะคำนวณค่าสัมประสิทธิ์การสะท้อนที่ load (Γ_L) และ Γ_{in} ได้ผลลัพธ์จากการคำนวณด้วยโปรแกรม MATLAB มีค่า Γ_L เท่ากับ $0.233 \angle -149.255^\circ$ และค่า Γ_{in} เท่ากับ $0.232 \angle 124.483^\circ$ จะสังเกตได้ชัดเจนว่าค่า magnitude ของ Γ_{in} และ Γ_{out} นั้นมีค่าที่น้อยกว่า 1 ซึ่งเป็นเพราะ $|\Gamma_s| < 1$ มีผลทำให้ $\Gamma_{out} < 1$ ตามไปด้วยด้วย และมี $|\Gamma_L| < 1$ มีผลทำให้ $\Gamma_{in} < 1$ ดังนั้นสามารถสรุปได้ว่า วงจรที่เราออกแบบนั้นมีความเสถียรภาพ (Stability) ที่ความถี่ 1 GHz และที่ความถี่นี้มีค่า Transducer power gain = -0.532727 หรือ -2.734949 dB

MATLAB Code

Check the stability and the transducer power gain of the circuit at 2.5 GHz

```
% Single-stage amplifier design for maximum gain, at 2.5 GHz
% EIE/ENE 450 Applied Communications and Transmission Lines
function amplifier_design_for_class
%-----%
% adjustable parameters %
%-----%
clear all;
Z0 = 50; % line impedance (Ohms)
% scattering parameter at 2.5GHz
f = 2.5e9; % frequency (Hz)
S_abs = [0.314 0.482; 1.109 0.423]; % magnitudes of the scattering parameters
S_deg = [57.3 28; 11.7 -135.9]; % angles of the scattering parameters (degrees)
fprintf('S_abs = %d', S_abs(2,1));
% scattering parameters at other frequencies, to check the stability
% and gain
check{1}.f = 1e9; % frequency to check (Hz), i.e., 1 GHz
check{1}.S_abs = [0.352 0.239; 2.254 0.210]; % magnitudes of the scattering parameters
at 1 GHz
check{1}.S_deg = [132.4 59.2; 54.4 -124.2]; % angles of the scattering parameters
(degrees) at 1 GHz
%-----%
% amplifier design at 2.5 GHz %
%-----%
S = S_abs .* exp( j*pi/180*S_deg );

%---- Part 1: check stability ----
Delta = det(S);
abs_Delta = abs( Delta ); % magnitude of Delta
K = (1 - ( S_abs(1,1) )^2 - ( S_abs(2,2) )^2 + ...
abs_Delta^2 ) / ( 2*abs( S(1,2)*S(2,1) ) ); % paramter K
% print out
fprintf('\n--Step 1: K-Delta stability test\n');
fprintf('Delta: |Delta| = %f, deg of Delta = %f\n', abs(Delta), angle(Delta)*180/pi );
fprintf('K = %g\n', K );
% print out the result
if ( K > 1 && abs_Delta < 1 )
    fprintf('Unconditionally stable\n');
else
    fprintf('Conditionally stable\n');
end

%---- Part 2: GammaS ----
B1 = 1 + (S_abs(1,1))^2 - (S_abs(2,2))^2 - abs_Delta^2;
C1 = S(1,1) - Delta*conj(S(2,2));
GammaS = ( B1 + [1 -1]*sqrt( B1^2 - 4*(abs(C1))^2 ) ) / (2*C1);
% print out answer
fprintf('\n--Step 2: GammaS\n');
for g = GammaS
    fprintf('GammaS: |GammaS| = %f, deg of GammaS = %f\n', abs(g), angle(g)*180/pi );
end
fprintf('choose the 2nd solution because it is inside the input stability region\n');
gammaS = GammaS(2);
fprintf('gammaS = %f', gammaS);

%---- Part 3: Zs ----
Zs = Z0*( 1 + gammaS ) / ( 1 - gammaS );
```

```

fprintf('\n--Step 3: Zs\n');
fprintf('Zs = %f + j(%f)\n', real(Zs), imag(Zs) );
fprintf('|Zs| = %f, deg of Zs = %f\n', abs(Zs), angle(Zs)*180/pi );

%---- Part 4: input matching circuit -----
tuner_in = single_shunt_stub( conj(Zs), Z0 );
% printout only open-circuited shunt stub
nsol = length(tuner_in);
fprintf('\n--Step 4: input matching circuit\n');
fprintf(1, '[Single-stub shunt tuner] %d solution(s):', nsol );
for k=1:nsol
    fprintf(1, '\nSolution #%d\n', k );
    fprintf(1, ' Distance of the stub: d/lambda = %g\n', tuner_in{k}.d );
    fprintf(1, ' Short circuit: ls/lambda = %g\n', tuner_in{k}.ls );
    fprintf(1, ' Open circuit: lo/lambda = %g\n', tuner_in{k}.lo );
end

%---- Part 5: GammaL -----
B2 = 1 + (S_abs(2,2))^2 - (S_abs(1,1))^2 - abs_Delta^2;
C2 = S(2,2) - Delta*conj(S(1,1));
GammaL = ( B2 + [1 -1]*sqrt( B2^2 - 4*(abs(C2))^2 ) ) / (2*C2);
% print out answer
fprintf('\n--Step 5: GammaL\n');
for g=GammaL
    fprintf('GammaL: |GammaL| = %f, deg of GammaL = %f\n', abs(g), angle(g)*180/pi );
end
fprintf('choose the 2nd solution because it is inside the input stability region\n');
gammaL = GammaL(2);

%---- Part 6: Zl -----
Zl = Z0*( 1 + gammaL ) / ( 1 - gammaL );
fprintf('\n--Step 6: Zl\n');
fprintf('Zl = %f + j(%f)\n', real(Zl), imag(Zl) );
fprintf('|Zl| = %f, deg of Zl = %f\n', abs(Zl), angle(Zl)*180/pi );

%---- Part 7: output matching circuit -----
tuner_out = single_shunt_stub1( conj(Zl), Z0 );
% printout only open-circuited shunt stub
nsol = length(tuner_out);
fprintf('\n--Step 7: output matching circuit\n');
fprintf(1, '[Single-stub shunt tuner] %d solution(s):', nsol );
for k=1:nsol
    fprintf(1, '\nSolution #%d\n', k );
    fprintf(1, ' Distance of the stub: d/lambda = %g\n', tuner_out{k}.d );
    fprintf(1, ' Short circuit: ls/lambda = %g\n', tuner_out{k}.ls );
    fprintf(1, ' Open circuit: lo/lambda = %g\n', tuner_out{k}.lo );
end

%---- Part 8: Check stability
gamma_out = S(2,2) + S(1,2)*S(2,1)*gammaS/( 1-S(1,1)*gammaS );
gamma_in = S(1,1) + S(1,2)*S(2,1)*gammaL/( 1-S(2,2)*gammaL );
fprintf('\n--Step 8: Careful check of stability\n');
fprintf('The input port of transistor is...');
if ( abs(gammaS) < 1 && abs(gamma_out) < 1 )
    fprintf('stable\n');
else
    fprintf('unstable\n');
end
fprintf('The output port of transistor is...');
if ( abs(gammaL) < 1 && abs(gamma_in) < 1 )
    fprintf('stable\n');
else
    fprintf('unstable\n');
end

```

```

end

%---- Part 9: Compute the tranducer gain
Gs = 1/(1-(abs(gammaS))^2);
G0 = (S_abs(2,1))*(S_abs(2,1));
GL = (1-(abs(gammaL))^2) / ( abs(1-S(2,2)*gammaL) )^2;
GTmax = Gs * G0 * GL;
fprintf('\n--Step 9: Gains\n');
fprintf('Gs = %f = %f dB\n', Gs, 10*log10(Gs) );
fprintf('G0 = %f = %f dB\n', G0, 10*log10(G0) );
fprintf('GL = %f = %f dB\n', GL, 10*log10(GL) );
fprintf('GTmax = %f = %f dB\n', GTmax, 10*log10(GTmax) );
%-----%
% gain and stability at other frequencies %
%-----%
ncheck = length(check); % number of frequencies to check
for k=1:ncheck

    fprintf('\n\nFrequency = %g Hz\n', check{k}.f );

    % scattering parameter at frequency check{k}.f
    SS = check{k}.S_abs .* exp( j*pi/180*check{k}.S_deg );

    % stability at the input port of transistor
    % input impedance seen looking into the input matching network
    % {2} for the second solution of the single shunt stub
    Z = Zin_shunt_stub( f, tuner_in{2}.d, tuner_in{2}.lo, Z0, Z0, 'Short', check{k}.f );
    gamS = (Z - Z0)/(Z + Z0); % reflection coefficient at the source side
    gam_out = SS(2,2) + SS(1,2)*SS(2,1)*gamS/( 1-SS(1,1)*gamS );
    % stability at the ouput port of transistor
    % {2} for the second solution of the single shunt stub
    Z = Zin_shunt_stub( f, tuner_out{2}.d, tuner_out{2}.lo, Z0, Z0, 'Open', check{k}.f );
    gamL = (Z - Z0)/(Z + Z0); % reflection coefficient at the load side
    gam_in = SS(1,1) + SS(1,2)*SS(2,1)*gamL/( 1-SS(2,2)*gamL );

    fprintf(' gamS: |gamS| = %f, deg of gamS = %f\n', abs(gamS), angle(gamS)*180/pi );
    fprintf(' gam_out: |gam_out| = %f, deg of gam_out = %f\n', abs(gam_out),
angle(gam_out)*180/pi );
    fprintf(' gamL: |gamL| = %f, deg of gamL = %f\n', abs(gamL), angle(gamL)*180/pi );
    fprintf(' gam_in: |gam_in| = %f, deg of gam_in = %f\n', abs(gam_in),
angle(gam_in)*180/pi );
    fprintf('\n The input port of transistor is...');
    if ( abs(gamS) < 1 && abs(gam_out) < 1 )
        fprintf('stable\n');
    else
        fprintf('unstable\n');
    end
    fprintf(' The output port of transistor is...');
    if ( abs(gamL) < 1 && abs(gam_in) < 1 )
        fprintf('stable\n');
    else
        fprintf('unstable\n');
    end
end

%---- Compute the tranducer gain
% Note: This calculation comes from equation (12.13) of the textbook
% Equation (12.37) will not work here because (12.37) is only for
% the maximum gain (conjugate matching), at 2.5 GHz
Gs = (1-(abs(gamS))^2) / ( abs(1-gamS*gam_in) )^2;
G0 = (check{k}.S_abs(2,1))^2;
GL = (1-(abs(gamL))^2) / ( abs(1-SS(2,2)*gamL) )^2;
GTmax = Gs * G0 * GL;
fprintf('\n Gains:\n');

```

```

fprintf(' Gs = %f = %f dB\n', Gs, 10*log10(Gs) );
fprintf(' G0 = %f = %f dB\n', G0, 10*log10(G0) );
fprintf(' GL = %f = %f dB\n', GL, 10*log10(GL) );
fprintf(' GT = %f = %f dB\n', GTmax, 10*log10(GTmax) );
end
end
% Matches a load Zload to the line with impedance Z0 using a
% shunt-stub tuner
% Input:
% ZL - a complex number for the load impedance (Ohm)
% Z0 - a real number for the characteristic impedance of the
% transmission line (Ohm)
% Output:
% tuner - a cell, where
% tuner{k}.d is (the length from the line to the load)/lambda
% tuner{k}.ls is (the length of the short-circuited stub)/lambda
% tuner{k}.lo is (the length of the open-circuited stub)/lambda
% Here, length(tuner) is the number of solutions
function tuner = single_shunt_stub( ZL, Z0 )
    RL = real(ZL); % load resistance
    XL = imag(ZL); % load reactance
    Y0 = 1/Z0; % characteristic admittance of the line
    % obtain t
    if ( RL ~= Z0 )
        % two solution. Put them in a vector 't'
        t = ( XL + (-1).^[0 1] * sqrt( RL*( (Z0-RL)^2 + XL^2 )/Z0 ) ) ...
            / ( RL - Z0 );
    else
        % one solution
        t = -XL/(2*Z0);
    end
    % obtain B
    B = ( RL^2*t - (Z0-XL*t).*(XL + Z0*t) ) ...
        ./ ( Z0*(RL^2 + (XL + Z0*t).^2 ) );
    % obtain normalized length, norm_ls = ls/lambda, of the short-circuit stub
    norm_ls = atan( Y0./B ) / (2*pi);
    norm_ls( norm_ls < 0 ) = norm_ls( norm_ls < 0 ) + 1/2;
    % obtain normalized length, norm_lo = lo/lambda, of the open-circuit stub
    norm_lo = -atan( B/Y0 ) / (2*pi);
    norm_lo( norm_lo < 0 ) = norm_lo( norm_lo < 0 ) + 1/2;
    % obtain the normalized distance, norm_d = d/lambda, of the stub
    % Note that t can be a vector, so we can have multiple solutions of norm_d
    norm_d = atan( t ) / (2*pi);
    norm_d( t<0 ) = norm_d( t<0 ) + 1/2;
    % prepare the output
    nsol = length( norm_d ); % number of solution
    tuner = cell( nsol );
    for k=1:nsol
        tuner{k}.d = norm_d(k);
        tuner{k}.ls = norm_ls(k);
        tuner{k}.lo = norm_lo(k);
    end
    %-----%
    % Gamma vs frequency %
    %-----%
    f0 = 2.5*10^9; % (Hz) frequency for which the load is ZL as given
    f = linspace( 1.5*10^9, 4*10^9, 500 ); % (Hz) range of frequencies to plot
    Z0 = 50; % (Ohm) characteristic impedance
    % Suppose ZL is a series of RC or of RL
    if ( XL < 0 ) % series of R and C
        C = 1/(-XL*2*pi*f0); % capacitance (F) in the series
        ZL_f = RL + 1./(j*2*pi*f*C); % load impedance, as a function of frequency
        % show the components in the load

```



```

else
    L = XL / (2*pi*f0); % inductance (H) in the series
    ZL_f = RL + j*2*pi*f*L; % load impedance, as a function of frequency
    % show the components in the load
end
% compute |Gamma| vs frequency
figure(1);
clf;
hold all;
leg = cell( 1, 2*nsol ); % legend
for k=1:nsol
    % impedance down a length d from the load, equation (5.7) in the
    % textbook
    tan_term = tan( 2*pi*norm_d(k)*f/f0 ); % tan (beta*d)
    Z = Z0*( ZL_f + j*Z0*tan_term ) ./ (Z0 + j*ZL_f.*tan_term);

    % impedances of the stub transmission line
    % circuit
    Z_stub_o = -j*Z0*cot( 2*pi*norm_lo(k)*f/f0 ); % open stub
    Z_stub_s = j*Z0*tan( 2*pi*norm_ls(k)*f/f0 ); % short-circuited stub

    % the over-all input impedance looking into the matching network:
    % A parallel of Z and Z_sub
    Zin_o = Z .* Z_stub_o ./ (Z + Z_stub_o); % open stub
    Zin_s = Z .* Z_stub_s ./ (Z + Z_stub_s); % short-circuited stub

    % Gamma
    Gamma_o = (Zin_o - Z0) ./ (Zin_o + Z0); % open stub
    Gamma_s = (Zin_s - Z0) ./ (Zin_s + Z0); % short-circuited stub

    % plot and set legend
    figure(1);
    plot( f/10^9, [ abs(Gamma_o) ; abs(Gamma_s) ], 'Linewidth', 2 );
    leg{2*k-1} = sprintf('Sol #%d: open', k );
    leg{2*k} = sprintf('Sol #%d: shorted', k );
    grid on;
end
legend( leg );
title('Magnitude of the reflection coefficient at various frequencies');
xlabel('f (GHz)');
ylabel('| \Gamma |');
end
function tuner = single_shunt_stub1( ZL, Z0 )
    RL = real(ZL); % load resistance
    XL = imag(ZL); % load reactance
    Y0 = 1/Z0; % characteristic admittance of the line
    % obtain t
    if ( RL ~= Z0 )
        % two solution. Put them in a vector 't'
        t = ( XL + (-1).^[0 1] * sqrt( RL*( (Z0-RL)^2 + XL^2 )/Z0 ) ) ...
            / ( RL - Z0 );
    else
        % one solution
        t = -XL/(2*Z0);
    end
    % obtain B
    B = ( RL^2*t - (Z0-XL*t).*(XL + Z0*t) ) ...
        ./ ( Z0*(RL^2 + (XL + Z0*t).^2 ) );
    % obtain normalized length, norm_ls = ls/lambda, of the short-circuit stub
    norm_ls = atan( Y0./B ) / (2*pi);
    norm_ls( norm_ls < 0 ) = norm_ls( norm_ls < 0 ) + 1/2;
    % obtain normalized length, norm_lo = lo/lambda, of the open-circuit stub
    norm_lo = -atan( B/Y0 ) / (2*pi);

```

```

norm_lo( norm_lo < 0 ) = norm_lo( norm_lo < 0 ) + 1/2;
% obtain the normalized distance, norm_d = d/lambda, of the stub
% Note that t can be a vector, so we can have multiple solutions of norm_d
norm_d = atan( t ) / (2*pi);
norm_d( t<0 ) = norm_d( t<0 ) + 1/2;
% prepare the output
nsol = length( norm_d ); % number of solution
tuner = cell( nsol );
for k=1:nsol
    tuner{k}.d = norm_d(k);
    tuner{k}.ls = norm_ls(k);
    tuner{k}.lo = norm_lo(k);
end
%-----%
% Gamma vs frequency %
%-----%
f0 = 2.5*10^9; % (Hz) frequency for which the load is ZL as given
f = linspace( 1.5*10^9, 4*10^9, 500 ); % (Hz) range of frequencies to plot
Z0 = 50; % (Ohm) characteristic impedance
% Suppose ZL is a series of RC or of RL
if ( XL < 0 ) % series of R and C
    C = 1/(-XL*2*pi*f0); % capacitance (F) in the series
    ZL_f = RL + 1./(j*2*pi*f*C); % load impedance, as a function of frequency
    % show the components in the load
else
    L = XL / (2*pi*f0); % inductance (H) in the series
    ZL_f = RL + j*2*pi*f*L; % load impedance, as a function of frequency
    % show the components in the load
end
% compute |Gamma| vs frequency
figure(2);
clf;
hold all;
leg = cell( 1, 2*nsol ); % legend
for k=1:nsol
    % impedance down a length d from the load, equation (5.7) in the
    % textbook
    tan_term = tan( 2*pi*norm_d(k)*f/f0 ); % tan (beta*d)
    Z = Z0*( ZL_f + j*Z0*tan_term ) ./ (Z0 + j*ZL_f.*tan_term);

    % impedances of the stub transmission line
    % circuit
    Z_stub_o = -j*Z0*cot( 2*pi*norm_lo(k)*f/f0 ); % open stub
    Z_stub_s = j*Z0*tan( 2*pi*norm_ls(k)*f/f0 ); % short-circuited stub

    % the over-all input impedance looking into the matching network:
    % A parallel of Z and Z_sub
    Zin_o = Z .* Z_stub_o ./ (Z + Z_stub_o); % open stub
    Zin_s = Z .* Z_stub_s ./ (Z + Z_stub_s); % short-circuited stub

    % Gamma
    Gamma_o = (Zin_o - Z0) ./ (Zin_o + Z0); % open stub
    Gamma_s = (Zin_s - Z0) ./ (Zin_s + Z0); % short-circuited stub

    % plot and set legend
    figure(2);
    plot( f/10^9, [ abs(Gamma_o) ; abs(Gamma_s) ], 'Linewidth', 2 );
    leg{2*k-1} = sprintf('Sol #%d: open', k );
    leg{2*k} = sprintf('Sol #%d: shorted', k );
    grid on;
end
legend( leg );
title('Magnitude of the reflection coefficient at various frequencies');

```

```

xlabel('f (GHz)');
ylabel('| \Gamma |');
end
% Obtain the input impedance seen looking into a single-stub shunt tuner at
% a given frequency
% Input:
% f - frequency where the shunt tuner is designed for (Hz)
% d - (length of the line)/lambda, where lambda = wavelength on the
% transmission line for frequency f
% l - (length of the stub)/lambda
% Zload - the impedance of the load, a complex number (Ohm)
% Z0 - the impedance of the transmission line,
% a complex number although usually be real (Ohm)
% type - type of the stub, either 'Open' (for open-circuited) or 'Close'
% for (for closed-circuit)
% f_given - frequency to find the input impedance (Hz)
% Output:
% Z - the input impedance seen
function Z = Zin_shunt_stub( f, d, l, Zload, Z0, type, f_given )
% input impedance of the stub at the given frequency
beta_l = 2*pi*f_given/f * l; % beta (at the given frequency) * (stub length)
if ( type(1) == 'O' )
Zstub = -j*Z0*cot( beta_l );
else
Zstub = j*Z0*tan( beta_l );
end

% Zload || Zstub
Zeq = Zload * Zstub / (Zload + Zstub);

% input impedance of a line of length d terminated by a load Zeq
beta_d = 2*pi*f_given/f * d; % beta (at the given frequency) * (line length)
t = tan( beta_d );
Z = Z0*( Zeq + j*Z0*t)/( Z0 + j*Zeq*t );

fprintf(' >>> debugging: Zstub = %f + j(%f)Ohm\n', real(Zstub),imag(Zstub));
fprintf(' >>> debugging: Zeq = %f + j(%f) Ohm\n', real(Zeq), imag(Zeq) );
fprintf(' >>> debugging: Z = %f + j(%f) Ohm\n', real(Z), imag(Z) );
grid on
end

```

Check the stability and the transducer power gain of the circuit at 1 GHz

```
clear all;
% S parameter at 1GHz
S_abs = [0.352 0.239; 2.254 0.210]; % magnitudes of the scattering parameters
S_deg = [132.4 59.2; 54.4 -124.2]; % angles of the scattering parameters (degrees)
S = S_abs .* exp( j*pi/180*S_deg );

%input matching network
%find ZA
ls_in = 0.0855975;
d_in = 0.333558;
f_old = 2.5e9;
f_new = 1.0e9;
Zo=50;
Za =j*Zo*tan((2*pi*ls_in*f_new)/f_old); %short ckt
%find Zeq_in
Zeq_in =(Zo*Za)/(Zo+Za);
%find Zs at 2.5GHz
Zs =Zo*((Zeq_in+j*Zo*tan((2*pi*d_in*f_new)/f_old))...
/((Zo+j*Zeq_in*tan((2*pi*d_in*f_new)/f_old))));
%find gamma_s at 2.5GHz
gamma_s =(Zs-Zo)/(Zs+Zo);
%find gamma_out at 2.5GHz
gamma_out = S(2,2)+((S(2,1)*S(1,2)*gamma_s)/(1-S(1,1)*gamma_s));
fprintf('Za is %f+j ohm\n', real(Za), imag(Za));
fprintf('Zeq_in is %f+j ohm\n', real(Zeq_in), imag(Zeq_in));
fprintf('Zs is %f+j ohm\n', real(Zs), imag(Zs));
fprintf('Gamma_s is %fexp(j%f)\n',abs( gamma_s),(180/pi)*angle( gamma_s));
fprintf('Gamma_out is %fexp(j%f)\n',abs( gamma_out),(180/pi)*angle( gamma_out));

% output matching network
%find ZB
lo_out = 0.177888;
d_out = 0.158936;
Zb =-j*Zo*cot((2*pi*lo_out*f_new)/f_old); %open ckt
%find Zeq_out
Zeq_out =(Zo*Zb)/(Zo+Zb);
%find ZL at 2.5GHz
ZL =Zo*((Zeq_out + j*Zo*tan((2*pi*d_out*f_new)/f_old))...
/((Zo + j*Zeq_out*tan((2*pi*d_out*f_new)/f_old))));
%find gamma_L at 2.5GHz
gamma_L =(ZL-Zo)/(ZL+Zo);
%find gamma_in at 2.5GHz
gamma_in = S(1,1)+((S(2,1)*S(1,2)*gamma_L)/(1-S(2,2)*gamma_L));
fprintf('Zb is %f+j ohm\n', real(Zb), imag(Zb));
fprintf('Zeq_out is %f+j ohm\n', real(Zeq_out), imag(Zeq_out));
fprintf('ZL is %f+j ohm\n', real(ZL), imag(ZL));
fprintf('Gamma_L is %fexp(j%f)\n',abs( gamma_L),(180/pi)*angle( gamma_L));
fprintf('Gamma_in is %fexp(j%f)\n',abs( gamma_in),(180/pi)*angle( gamma_in));

%find transducer power gain by gamma_L, gamma_in, gamma_s
GT = (1-abs(gamma_s)^2)*abs(S(2,1))*abs(S(2,1))*(1-abs(gamma_L)^2);
GT = GT/(abs(1 - gamma_in*gamma_s)*abs(1 - gamma_in*gamma_s));
GT = GT/(abs(1 - S(2,2)*gamma_L)*abs(1 - S(2,2)*gamma_L));
fprintf('Transducer power gain %f\n', GT);
GT = 10*log10(GT);
fprintf('Transducer power gain in dB is %f dB\n', GT);
```