Question 1

He-Ne pure Guassain TEM00 mode at $\lambda_0 = 632.8nm$,$P = 5mw$, Divergence Angle $\theta = 1mRads$

$$\theta = \left(\frac{2\lambda_0}{\pi * n * \omega_0}\right)(Eq3.4.2)$$

$$\omega_0 = \left(\frac{2\lambda_0}{\pi * n * \theta}\right) = \left(\frac{2 * 632.8nm}{\pi * n * 1 * 10^{-3} Rads}\right) = 0.402mm$$

Spot size $\omega_0 = 0.402mm$

$$P = \frac{1}{2} * \frac{E_0^2}{\eta} * \left(\frac{\pi * @w_0^2}{2}\right)$$

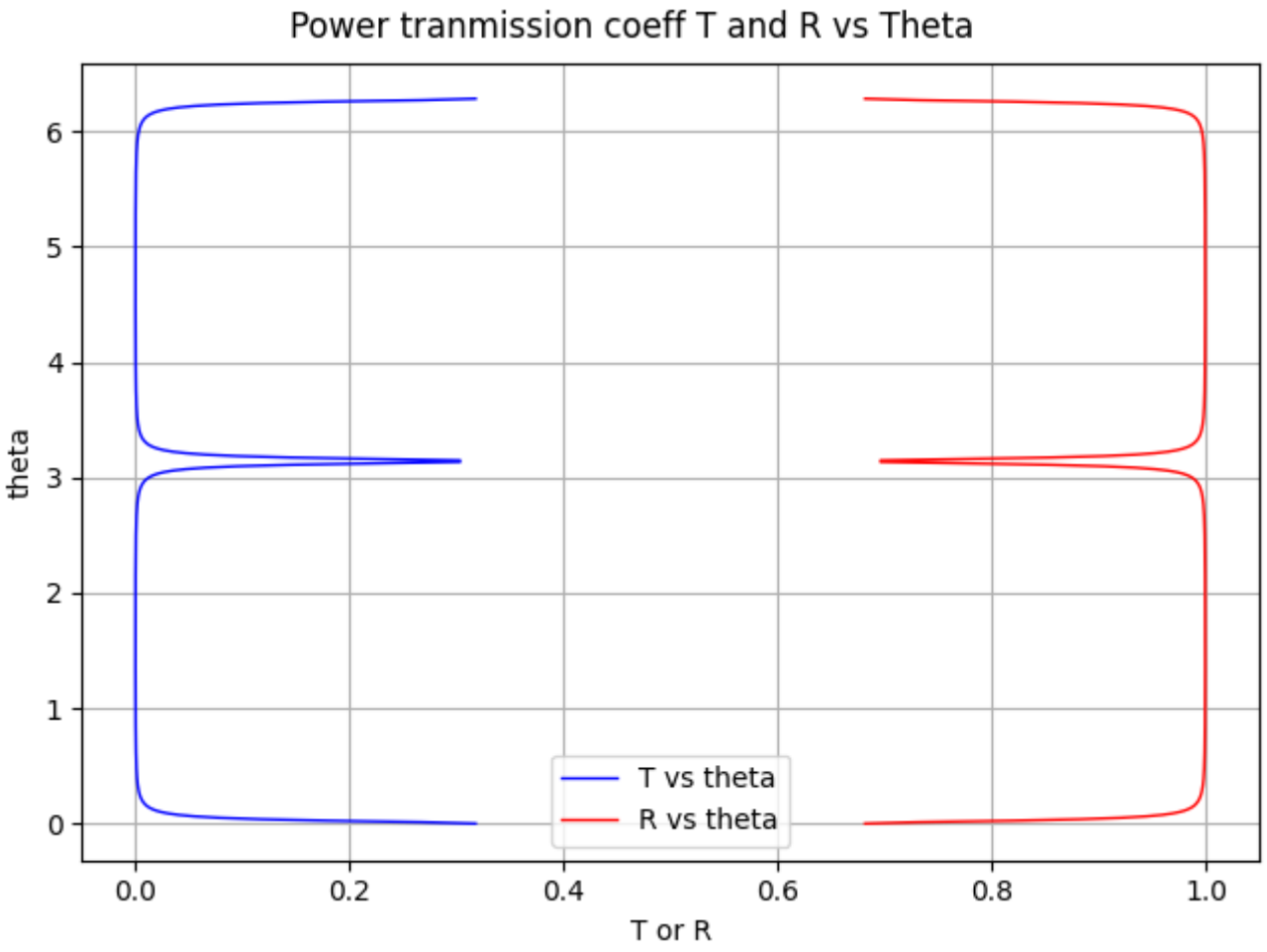
$$E_0^2 = 2 * P * \eta * \frac{2}{\pi * \omega_0^2} = 2.5mW * 367.7\Omega \frac{*2}{\pi * (0.402mm)}$$

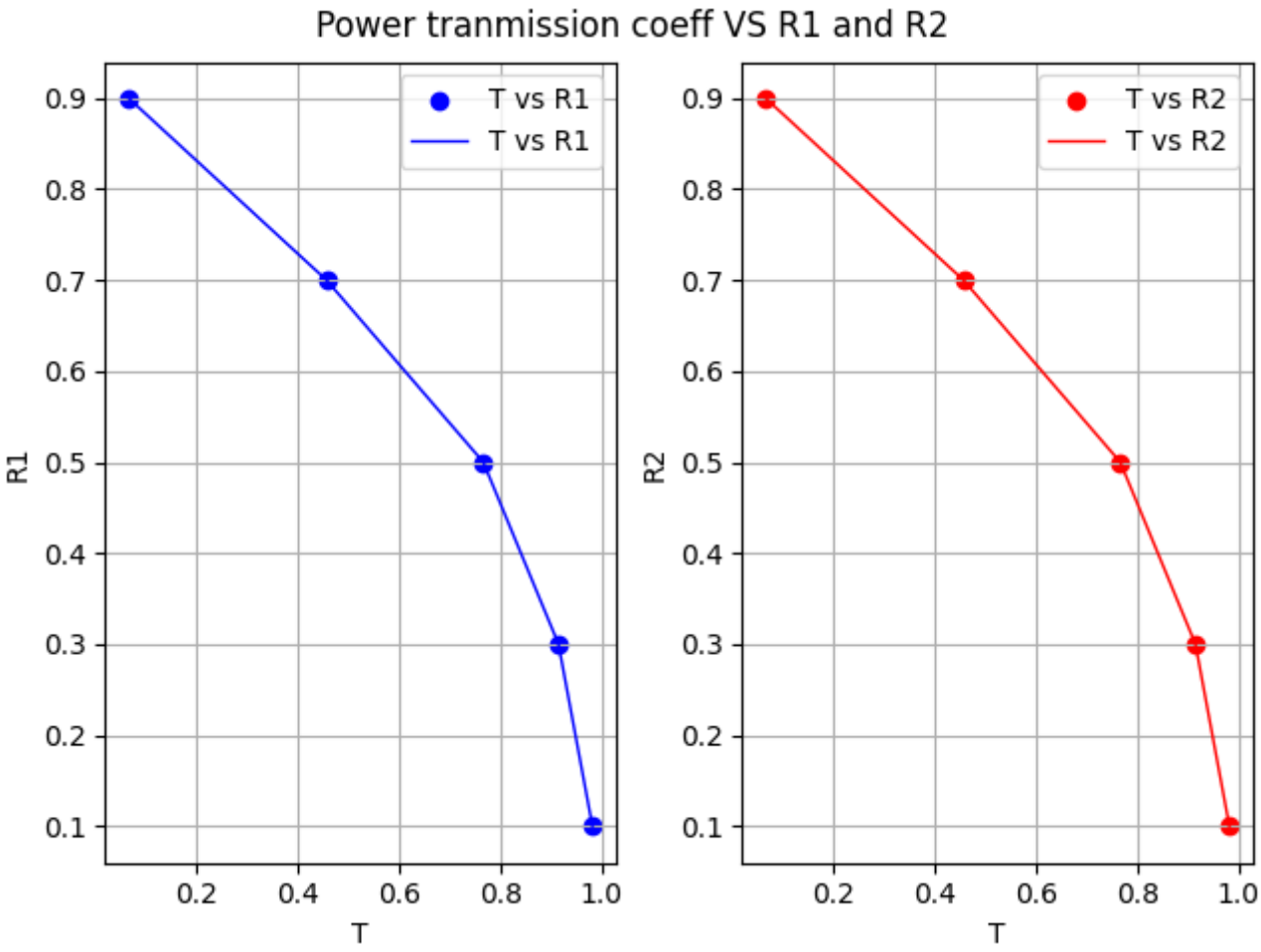
$$E_0^2 = 1.442 * 10^7 \frac{W}{m^2}$$

$$E_0 = \sqrt{E_0^2} = 3797.87 \frac{W}{m}$$

Peak intesnity $E_0^2 = 1.442 * 10^7 \frac{W}{m^2}$
Peak electric field $E_0 = \sqrt{E_0^2} = 3797.87 \frac{W}{m}$

Question 2


Power tranmission coeff T and R vs Theta
theta
T or R
T vs theta
R vs theta
0
1
2
3
4
5
6
0.0
0.2
0.4
0.6
0.8
1.0


Power tranmission coeff VS R1 and R2
T vs R1
T vs R1
R1
T
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.2
0.4
0.6
0.8
1.0
T vs R2
T vs R2
R2
T

Code

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

#---------------------------------
# Q2.1 defining basic variables
theta_1 = np.linspace(0,2*np.pi, 500)
R1_1 = 0.99
R2_1 = 0.9

# defining the power transmission coefficient
T_1 = ((1-R1_1)*(1-R2_1))/((1-np.sqrt(R1_1*R2_1))**2+(4*np.sqrt(R1_1*R2_1)*
(np.sin(theta_1))**2))

# defining R
R_1 = 1-T_1

# plotting
plt.subplots(layout="constrained")
plt.suptitle("Power tranmission coeff T and R vs Theta")

ax1 = plt.subplot(111)
plt.plot(T, theta, c='Blue',linewidth = 1, label = "T vs theta")
plt.xlabel("T or R")
plt.ylabel("theta")

ax2 = plt.subplot(111)
plt.plot(R, theta, c='red',linewidth = 1, label = "R vs theta")
plt.grid()
plt.legend()

plt.show()
#---------------------------------
# Q2.2 defining basic variables
R1_2 = np.array([0.1,0.3,0.5,0.7,0.9])
R2_2 = np.array([0.1,0.3,0.5,0.7,0.9])
theta_2 = (2*np.pi)+(1/16*np.pi)

# defining the power transmission coefficient
T_2 = ((1-R1_2)*(1-R2_2))/((1-np.sqrt(R1_2*R2_2))**2+(4*np.sqrt(R1_2*R2_2)*
(np.sin(theta_2))**2))

# defining R
```

```
R_2 = 1-T_2

# plotting
ax1 = plt.subplot(121)
plt.scatter(T, R1, c='Blue',linewidth = 1, label = "T vs R1")
plt.plot(T, R1, c='Blue',linewidth = 1, label = "T vs R1")
plt.xlabel("T")
plt.ylabel("R1")
plt.title("")
plt.grid()
plt.legend()

ax2 = plt.subplot(122)
plt.scatter(T, R2, c='Red',linewidth = 1, label = "T vs R2")
plt.plot(T, R2, c='Red',linewidth = 1, label = "T vs R2")
plt.xlabel("T")
plt.ylabel("R2")
plt.title("")
plt.grid()
plt.legend()

plt.show()
```