

```
[13]: # UTS no 2
import scipy.signal as sig # untuk analisis sistem sinyal, termasuk model state-space dan respons sistem
import matplotlib.pyplot as plt # untuk membuat grafik
import numpy as np # untuk manipulasi array dan operasi numerik

# Simulation parameters
x0 = [0,0] # kondisi awal (state awal)

start = 0
stop = 30
step = 1
t = np.arange(start,stop,step) # vektor waktu dari 0 sampai 29 dengan langkah 1

K = 3 # gain sistem
T = 5 # konstanta waktu

# State space model
A = [[-1/T,0],[0,0]]
B = [[K/T],[0]]
C = [[1,0]]
D = 0

sys = sig.StateSpace(A,B,C,D) # Membuat representasi sistem dalam bentuk state-space

# Step response
t,y = sig.step(sys, x0, t) # Menghitung respons sistem terhadap input tangga (step input) dengan kondisi awal nol

# plotting
plt.plot(t,y)
plt.title("Step Response")
plt.xlabel("t")
plt.ylabel("y")
plt.grid()
plt.show()
```

```
[14]: # UTS no 2
import scipy.signal as sig # untuk analisis sistem sinyal, termasuk model state-space dan respons sistem
import matplotlib.pyplot as plt # untuk membuat grafik
import numpy as np # untuk manipulasi array dan operasi numerik

# Simulation parameters
x0 = [0,0] # kondisi awal (state awal)

start = 0
stop = 30
step = 1
t = np.arange(start,stop,step) # vektor waktu dari 0 sampai 29 dengan langkah 1

K = 3 # gain sistem
T = 10 # konstanta waktu

# State space model
A = [[-1/T,0],[0,0]]
B = [[K/T],[0]]
C = [[1,0]]
D = 0

sys = sig.StateSpace(A,B,C,D) # Membuat representasi sistem dalam bentuk state-space

# Step response
t,y = sig.step(sys, x0, t) # Menghitung respons sistem terhadap input tangga (step input) dengan kondisi awal nol

# plotting
plt.plot(t,y)
plt.title("Step Response")
plt.xlabel("t")
plt.ylabel("y")
plt.grid()
plt.show()
```

```
[15]: # UTS no 2
import scipy.signal as sig # untuk analisis sistem sinyal, termasuk model state-space dan respons sistem
import matplotlib.pyplot as plt # untuk membuat grafik
import numpy as np # untuk manipulasi array dan operasi numerik

# Simulation parameters
x0 = [0,0] # kondisi awal (state awal)

start = 0
stop = 30
step = 1
t = np.arange(start,stop,step) # vektor waktu dari 0 sampai 29 dengan langkah 1

K = 3 # gain sistem
T = 30 # konstanta waktu

# State space model
A = [[-1/T,0],[0,0]]
B = [[K/T],[0]]
C = [[1,0]]
D = 0

sys = sig.StateSpace(A,B,C,D) # Membuat representasi sistem dalam bentuk state-space

# Step response
t,y = sig.step(sys, x0, t) # Menghitung respons sistem terhadap input tangga (step input) dengan kondisi awal nol

# Plotting
plt.plot(t,y)
plt.title("Step Response")
plt.xlabel("t")
plt.ylabel("y")
plt.grid()
plt.show()
```