



TOBB Ekonomi ve Teknoloji Üniversitesi

Elektrik – Elektronik Mühendisliği Bölümü

ELE 504 – Dijital Kontrol Sistemleri

Final Ödevi Raporu

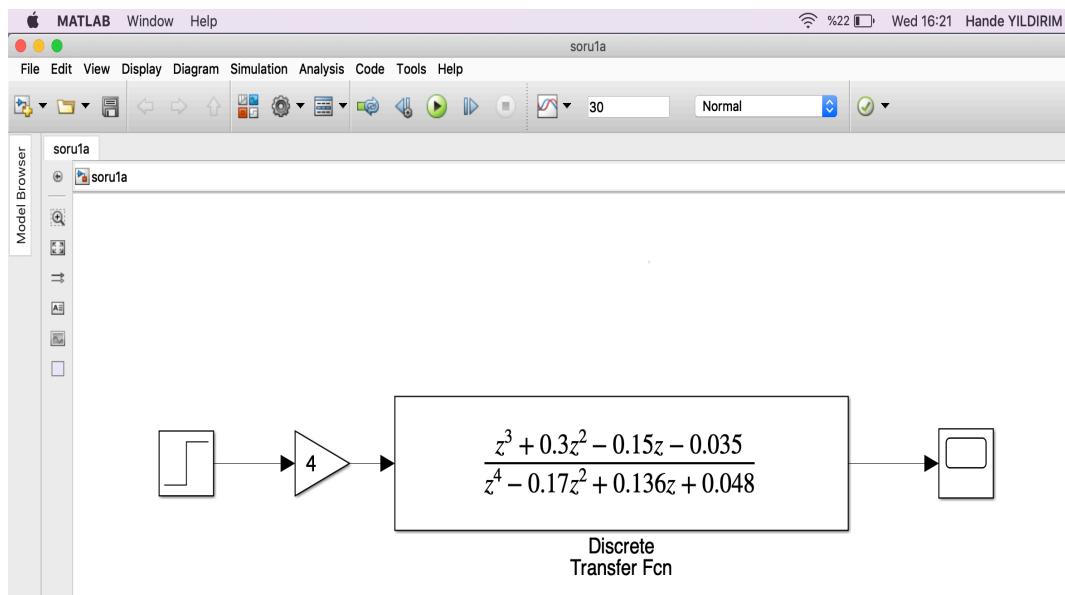
Adı Soyadı	Hande Yıldırım
Numara	141201047
Tarih	08.04.2020

SORU 1

Bu soru ayrik zaman transfer fonksiyonu aşağıda denklem 1.1'de yer alan sisteme uygun olarak çözülmüştür. Burada örnekleme periyodu $T = 1$ saniyedir.

$$G(z) = \frac{4*(z+0.7)*(z-0.5)*(z+0.1)}{(z+0.5)*(z+0.3)*(z^2 - 0.8z + 0.32)} \quad (1.1)$$

- a. Öncelikle sistem Simulink ortamında “Transfer Function” bloğu ile şekil 1.a.1’deki gibi gerçeklenmiştir. Bu gerçeklenme olmadan önce sistem aşağıdaki şekil 1.a.2’de gösterildiği gibi elle yazılarak açılmıştır daha sonra simulinke eklenmiştir.



Şekil 1.a.1: Transfer Function Bloğu ile Gerçeklenen Devre

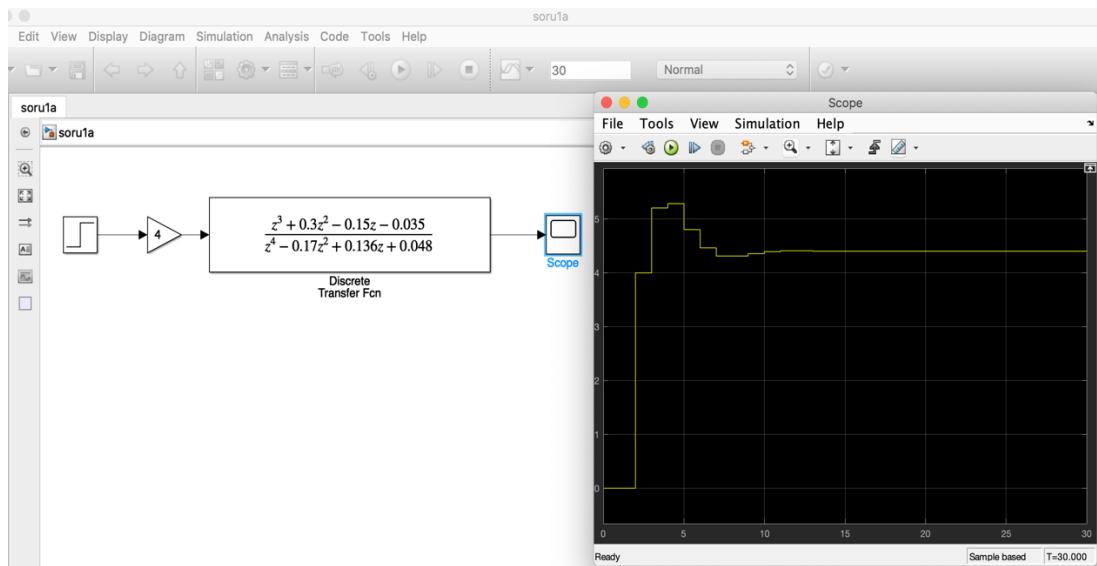
Bu devre gerçeklenirken öncelikle “discrete transfer function” bloğu kullanılarak sistemin açılmış hali yazılmıştır. Hemen öncesine de denklemde kat sayı olarak yer alan 4 kazanç olarak devreye eklenmiştir. Girdi olarak devreye örnekleme zamanı 1 olan bir “step” verilmiştir ve bunun sonucu olarak elde edilen çıktı “scope” kullanılarak gözlemlenmiştir.

SORU 1A)

$$\begin{aligned}
 G(z) &= \frac{4(z+0.1)(z-0.5)(z+0.1)}{(z+0.5)(z+0.1)(z^2-0.8z+0.08)} \\
 &= \frac{4(z^2+0.1z-0.15)(z+0.1)}{(z^2+0.8z+0.15)(z^2-0.8z+0.08)} \\
 &= \frac{4(z^3+0.1z^2-0.15z+0.2z-0.15z-0.035)}{z^4-0.8z^3+0.22z^2+0.8z^3-0.64z^2+0.156z+0.15z-0.12z+0.048} \\
 &= \frac{4(z^3+0.1z^2-0.15z-0.035)}{z^4-0.17z^2+0.136z+0.048} = \frac{4z^3+0.12z^2-0.67-0.14}{z^4-0.17z^2+0.136z+0.048} \quad \left. \begin{array}{l} \rightarrow m=3 \\ \rightarrow n=4 \end{array} \right\} \text{N>km}
 \end{aligned}$$

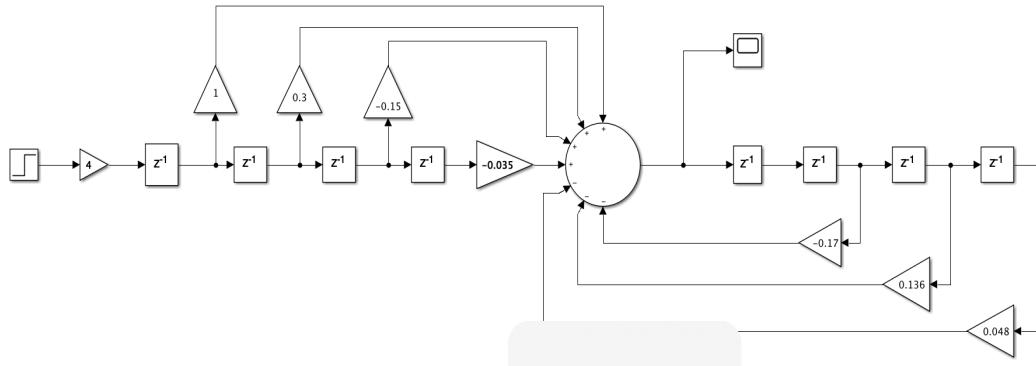
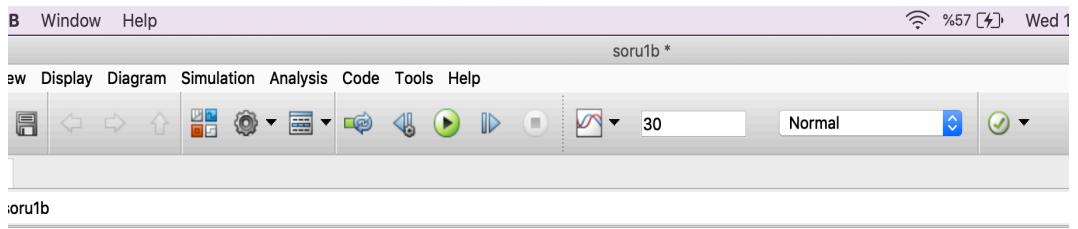
Şekil 1.a.2: Denklem 1.1'in Açılmış Hali

Bunun sonucunda simülasyon 30 saniye boyunca çalıştırıldığında elde edilen grafik aşağıdaki şekildeki şekil 1.a.3'teki gibi elde edilmiştir.



Şekil 1.a.3: Simulink Üzerinde Transfer Fonksiyonu ve Onun Ürettiği Grafik

- b.** Bu kısımda sistem Simulink ortamında “Doğrudan Programlama” yöntemi ile şekildeki gibi gerçekleştirmiştir. Bu gerçeklenme olmadan önce $G(z)$ sistemi aşağıdaki şekildeki şekildeki 1.b.2'deki forma getirilir.



Şekil 1.b.1: Doğrudan Programlama ile Gerçeklenen Devre

$$\begin{aligned}
 G(z) &= 4 \cdot \frac{z^3 + 0.1z^2 - 0.15z - 0.035}{z^4 - 0.17z^3 + 0.126z^2 + 0.048} \times \frac{z^{-4}}{z^{-4}} \\
 &= 4 \cdot \frac{\cancel{z^3} + \cancel{0.1z^2} - \cancel{0.15z} - \cancel{0.035} z^{-4}}{\cancel{z^4} - \cancel{0.17z^3} + \cancel{0.126z^2} + \cancel{0.048} z^{-4}} = \frac{0.1z^3 + 0.15z^2 - 0.6z + 0.14 z^{-4}}{12^0 - 0.17z^3 + 0.126z^2 + 0.048z^{-4}}
 \end{aligned}$$

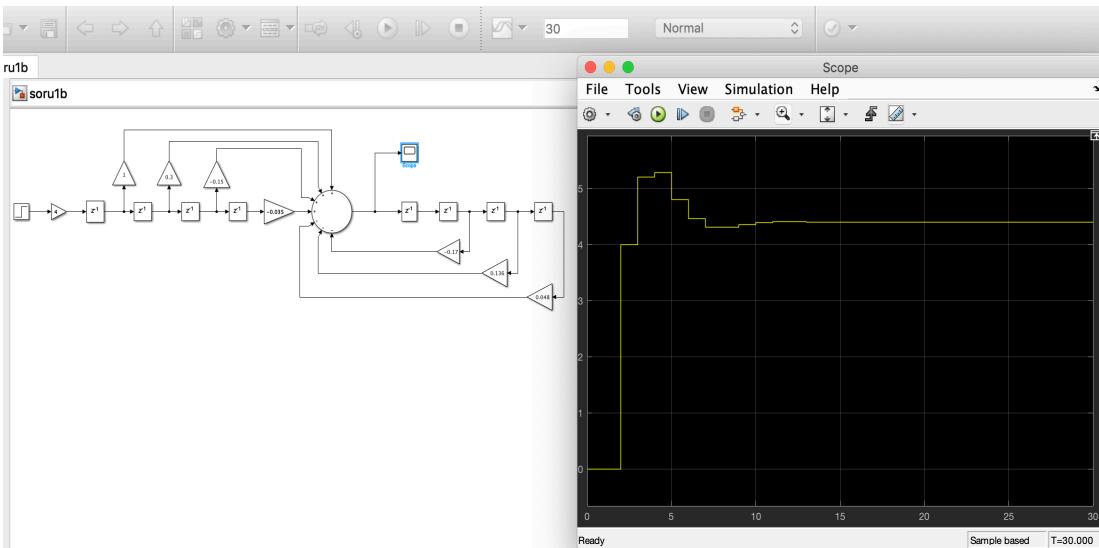
Şekil 1.b.2: Denklem 1.1'in Doğrudan Programlamaya Uygun Olarak Düzenlenmiş Hali

$G(z)$ 'nin şekil 1.b.2'deki yeni formu aslında denklem 1.2'deki gibi ifade edilebilir ve buradan da $Y(z)$ denklem 1.3'teki gibi de çekilebilir:

$$G(z) = Y(z)/X(z) \quad (1.2)$$

$$\begin{aligned}
 Y(z) &= -a_1 z^{-1} Y(z) - a_2 z^{-2} Y(z) - a_3 z^{-3} Y(z) - \cdots - a_n z^{-n} Y(z) + \\
 b_0 X(z) + b_1 z^{-1} X(z) + b_2 z^{-2} X(z) + b_3 z^{-3} X(z) + \cdots + b_m z^{-m} X(z)
 \end{aligned} \quad (1.3)$$

Burada a_0 'ın beklenildiği gibi 1 olduğu gözlemlenmiştir. Bu durumda devrenin 30 saniye boyunca gerçekleşmesi sonucu elde edilen grafik aşağıdaki gibi şekil 1.b.3'deki gibidir:



Şekil 1.b.3: Simulink Üzerinde Doğrudan Programlanan Devre ve Onun Ürettiği Grafik

- c. Burada sistem Simulink ortamında “Standart Programlama” yöntemi ile şekil 1.c.1’deki gibi gerçekleşmiştir. Bu gerçekleşme olmadan önce şekil 1.b.2’de gösterilen $G(z)$ sistemi denklem 1.4’teki formuna getirilir:

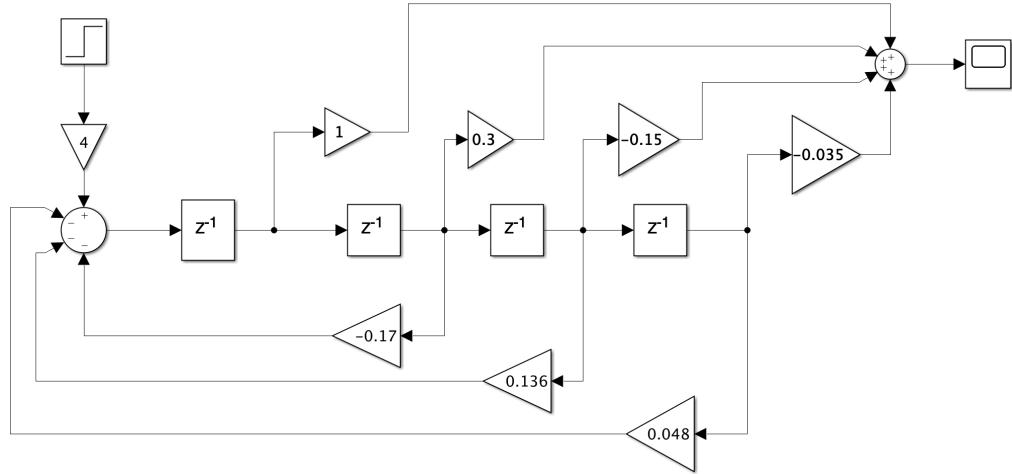
$$G(z) = \frac{Y(z)}{X(z)} = \frac{Y(z)}{H(z)} * \frac{H(z)}{X(z)} \quad (1.4)$$

Daha sonrasında bu fonksiyonlar genel denkleme bağlı olarak aşağıdaki denklem 1.5 ve denklem 1.6’daki formlarındaki gibi ayırtılırlar:

$$\frac{Y(z)}{H(z)} = b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_m z^{-m} \quad (1.5)$$

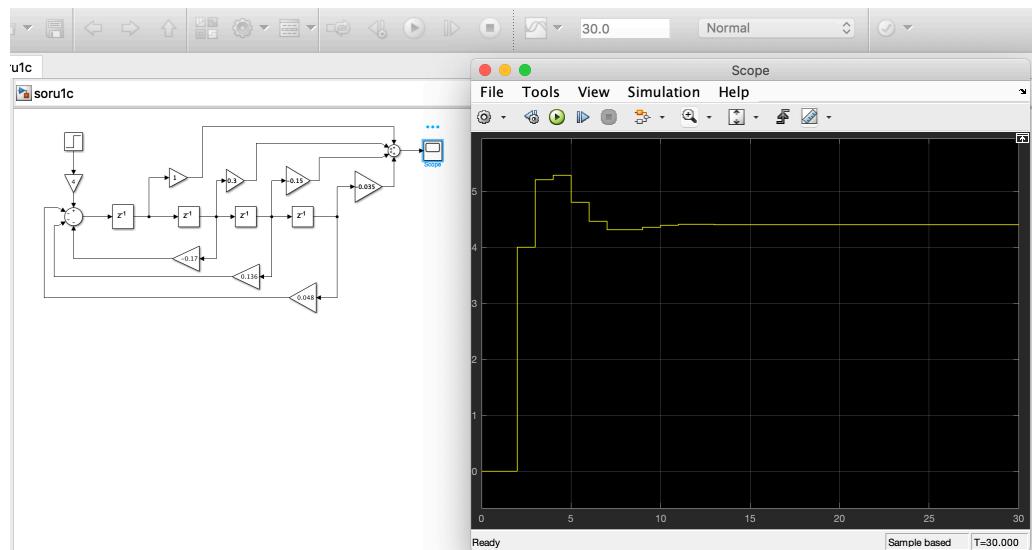
$$\frac{H(z)}{X(z)} = \frac{1}{1+a_1 z^{-1}+a_2 z^{-2}+\dots+a_n z^{-n}} \quad (1.6)$$

Yani Simulink’te modelleme yapılırken aşağıdaki şekil 1.c.1’de bulunan devrenin öncelikle üst tarafı (z^{-1} = delay fonksiyonlarından üst taraf) sonrasında da alt tarafı (z^{-1} = delay fonksiyonlarından alt taraf) modellenmiş ve daha sonra birleştirilerek genel olarak devre 30 saniye boyunca standart programlama ile gerçekleşmiştir.



Şekil 1.c.1: Standart Programlama ile Gerçeklenen Devre

Bu devreye girdi olarak diğer kısımlarda da yapıldığı gibi örnekleme zamanı 1 olan olan bir “step source” verilmiştir. Devrenin genel çıktısı ise devrenin çıktısının olduğu yere koyulan bir “scope” kullanarak gözlemlenmiştir. Bunun sonucunda elde edilen grafik aşağıda şekil 1.c.2’de gösterildiği gibidir:



Şekil 1.c.2: Simulink Üzerinde Standart Programlanan Devre ve Onun Ürettiği Grafik

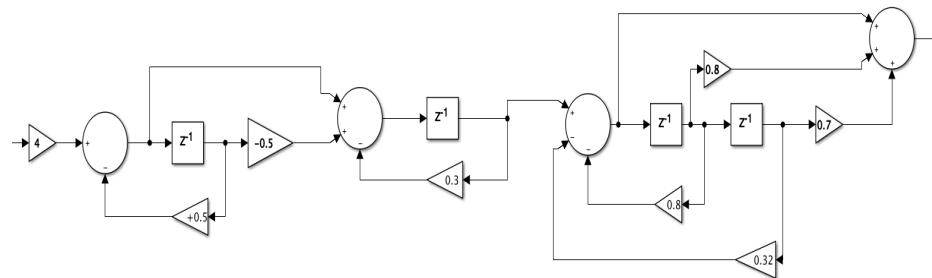
Yukarıda b şıklında bulunan doğrudan ve c şıklıkında bulunan standart programlama işlemlerinde sistem bir bütün olarak düşünülperek bu şekilde gerçeklenmiştir. Fakat yüksek dereceli sistemlerle uğraşırken sistem düşük dereceli (1. ve 2. dereceli) alt sistemlere ayırip ayrı ayrı gerçekleyip, sonra bunların birleştirildiği durumlar da olabilir. Bunun içinde aşağıda

d şıklındaki gibi seri ve e şıklındaki gibi de paralel programlama yöntemleri de kullanılabilir. Bu sistemler genelde daha modüler bir yaklaşım içermektedirler.

- d. Bu kısımda ise sistem Simulink ortamında “Seri Programlama” yöntemi ile şekil 1.d.2’deki gibi gerçekleşmiştir. Bu gerçekleşme olmadan önce $G(z)$ sistemi denklem 1.7’de gösterildiği gibi yazılabilir:

$$G(z) = K * G_1(z) * G_2(z) * G_3(z) * \dots * G_p(z) \quad (1.7)$$

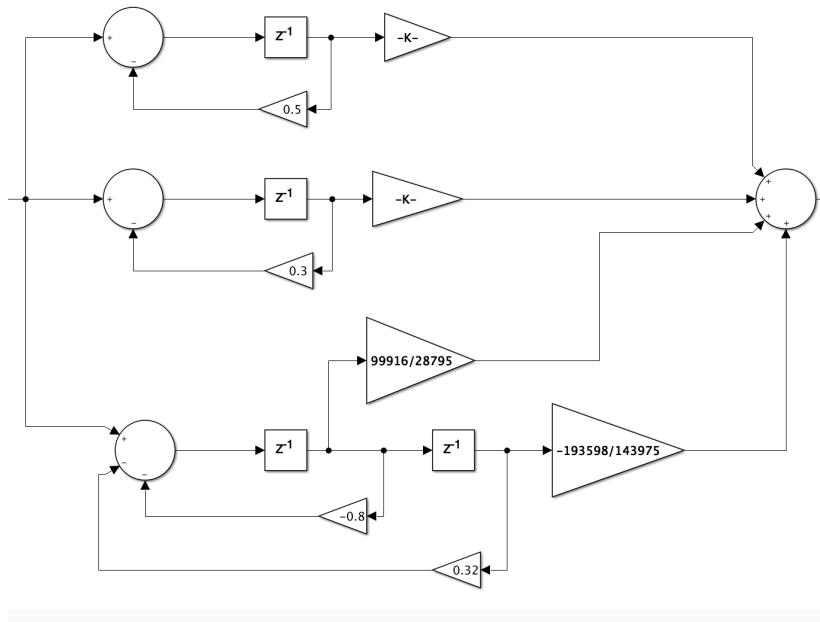
Burada $G_i(z)$ ’ler basit (1. ve 2. dereceden) ve alt sistemlerdirler. Ayrıca 1. dereceden terimler reel kutup/sıfırlardan, 2. dereceden terimler ise kompleks kutup/sıfırlardan meydana gelmektedirler. Yani buna sonuç olarak; öncelikle 1. dereceden terimlerin bulunduğu $G_i(z)$ ’ler Simulink ortamında gerçekleşir daha sonrasında 2. dereceden terimlerin bulunduğu $G_i(z)$ ’ler Simulink ortamında gerçekleşirler ve en son şekil 1.d.2’de verildiği gibi birleştirilirler.



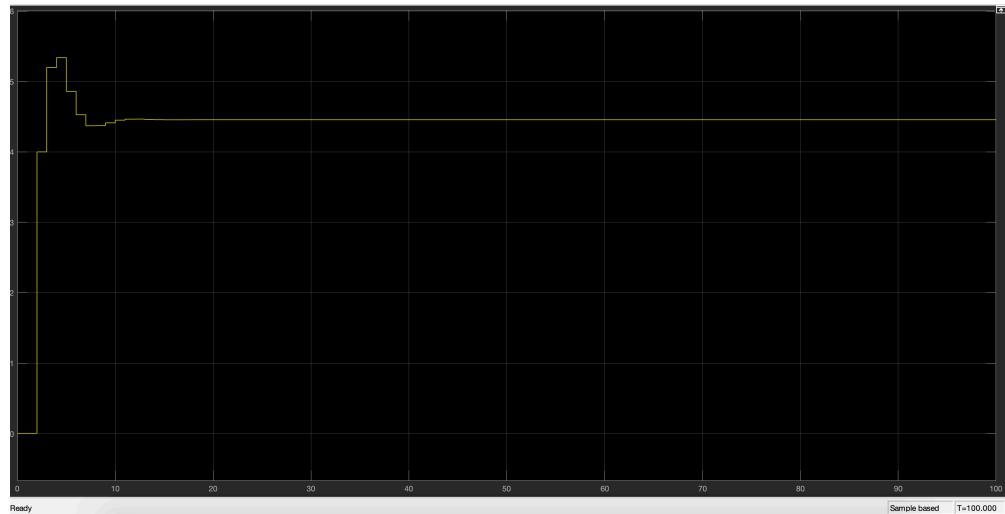
Sekil 1.d.1: Seri Programlama ile Gerçeklenen Devre

Seri programlanmış olan yukarıdaki devre 1saniyelik örneklemeye zamanında simüle edildiği zaman scope’tan elde edilecek olan grafik ise aşağıdaki şekil 1.d.2’de verildiği gibi olacaktır.

e. Simulink ortamında son olarak paralel programlama yöntemi ile $G(z)$ sistemi şekil 1.e.1'de verildiği gibi gerçeklenmiştir. Bunu yaparken seri programlama sonucu elde edilen şekil 1.d.1'deki denklemler kullanılmıştır. Buna göre ayrı ayrı elde edilen alt sistemler birbirlerine paralel olarak bağlanarak sistem 30 saniye boyu simülé edilmiştir. Bunun sonucunda elde edilen grafik ise şekil 1.d.2'de verildiği gibidir.



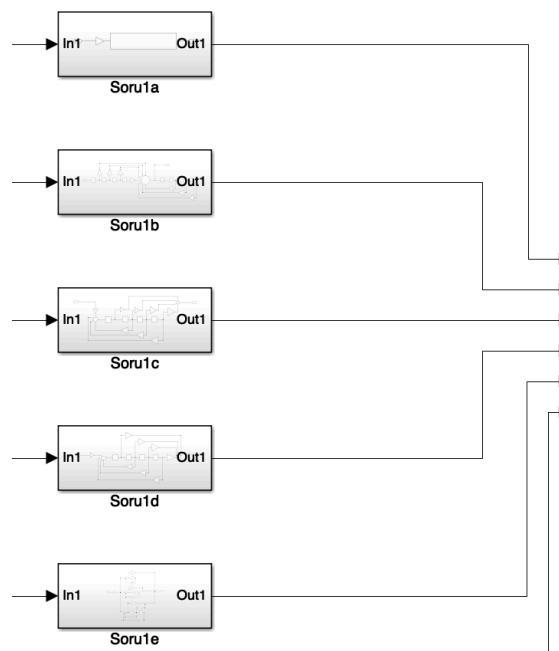
Şekil 31.e.1: Paralel Programlama ile Gerçeklenen Devre



Şekil 1.e.2: Paralel Programlama ile Elde Edilen Grafik

Yukarıdaki 5 şıkta da elde edilen tüm grafikler kıyaslandıkları zaman hepsinin birbirinin aynısı oldukları gözlemlenmiştir. Bu durumda yapılan tüm aşamaların başarıyla sonuçlandığı ve grafiklerden de gözlemlendiği gibi gerçekleşen her bir devre belli bir süre salınım ve aşım yaptıktan sonra beklenildiği gibi 1'e yakınsamaktadır.

- f. Bu şıkta daha önceden a-e şıkları arasında yer alan tüm gerçeklemelerin tek bir Simulink model dosyası altında alt alta birleştirilmeleri istenmiştir. Bunu yaparken her bir sistem “subsystem” bloğu içerisine yerleştirilmiştir. Bu durum aşağıdaki şekil 1.f.1’de gösterildiği gibi gerçekleşmiştir:

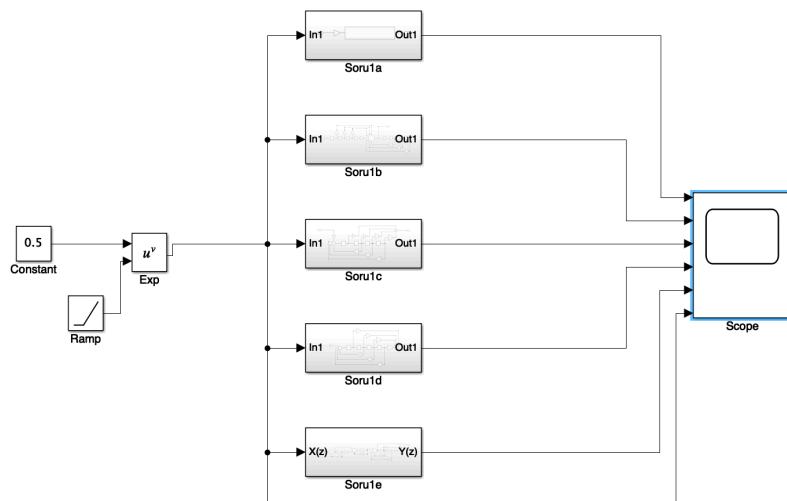


Şekil 1.f.1: Tüm Gerçeklemelerin Tek Bir Simulink Dosyası İçerisinde Yer Alması

- g. Sistemlerin hepsinin girişine aşağıda denklem 1.8’de verildiği gibi tanımlanan bir $x(k)$ giriş sinyali 10 saniye boyunca uygulanmış ve daha sonrasında da scope bağlanarak hem girdinin hem de çıktıların grafikleri aynı grafik içerisinde aşağıdaki şekil 1.g.2’deki gibi elde edilmiştir.

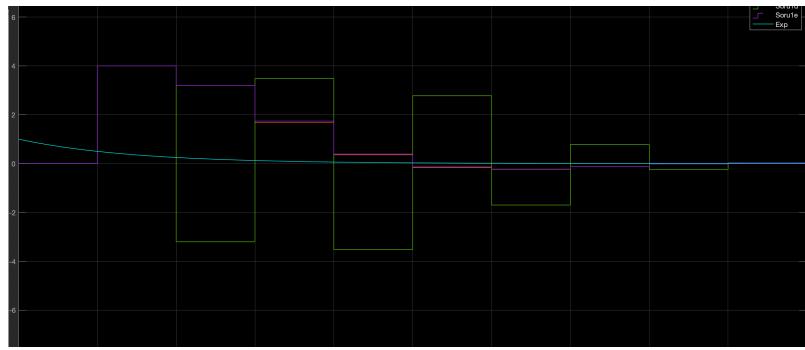
$$x(k) = \begin{cases} 0.5^k, & k \geq 0 \\ 0, & k < 0 \end{cases} \quad (1.8)$$

Bu girdiye bakıldığı zaman ilk anlaşılan girdinin bir parabolik denklem olduğunu söyleyebiliriz. Bu nedenle girdi olarak parabolik kaynak sağlayabilmek amacıyla öncelikle “ramp” koyulmuş sonrasında da buna “integrator” bağlanarak parabolik girdi verilmiştir ve scope ile çıktılar elde edilmiştir. Burada çıktıların hepsi için grafik üzerinde legendalar renkleri belirlenmiştir. Fakat buna rağmen tek 1 adet çıktı grafiği gözlemlenebilmektedir. Bunun sebebi olarağın aslında çıktıların hepsinin aynı grafiğe sahip olmaları ve bu nedenle üst üste binerek tek 1 renkte gözükmeleri olduğu söylenebilir. Ayrıca bu durum her bir gerçeklemenin başarılı olarak yapıldığının da kanıtıdır. Bu gerçeklemelerin yapıldığı devre şeması ise aşağıdaki şekil 1.g.1'deki gibidir:



Şekil 1.g.1: Her Bir Sistemin Aynı Simulink Dosyası İçerisinde Gerçeklenmesi

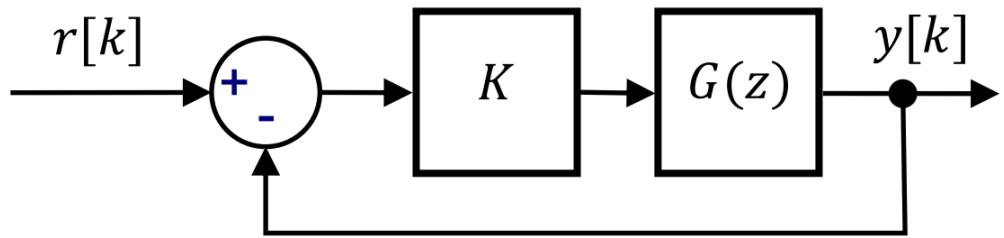
Bu devre şemasının 10 saniye boyunca simüle edilmesi sonucu elde edilen grafik ise aşağıdaki şekil 1.g.2'deki gibidir:



- h.** Bu şıkta denklem 1.8'de verilen $x(k)$ sinyali için z dönüşümü ile $X(z)$ sinyalinin bulunması, daha sonrasında bu sinyalin $G(z)$ 'ye uygulanarak $Y(z)$ çıkışının elde edilmesi ve son olarak z dönüşümünün tersini uygulayarak $y[k]$ çıkış sinyalinin bulunması ve bunun çizdirilmesi beklenmiştir ve $y[k]$ sinyali elde edilmiştir.

$$\begin{aligned}
 x[k] &= \begin{cases} 0.5^k, & k \geq 0 \\ 0, & k < 0 \end{cases} \rightarrow X(z) = ? \\
 X(z) &= \sum_{k=0}^{\infty} x[k] z^{-k} \\
 &= \sum_{k=0}^{\infty} 0.5^k z^{-k} = \sum_{k=0}^{\infty} (0.5z)^k \\
 &= \frac{1}{1-0.5z^{-1}} = \frac{z}{z-0.5} = \frac{2z}{2z-1} //
 G(z) &= \frac{4(z+0.2)(z+0.5)(z+0.1)}{(z+0.5)(z+0.1)(z-0.8z+0.12)} \\
 X(z) \rightarrow [G(z)] \rightarrow Y(z) & \Rightarrow \frac{Y(z)}{X(z)} = 8 \cdot \frac{(z+0.2)(z-0.5)(z+0.1)}{(2z-1)(z+0.5)(z+0.1)(z-0.8z+0.12)} \\
 &= \frac{a_1}{(2z-1)} + \frac{a_2}{(z+0.5)} + \frac{a_3}{(z+0.1)} + \frac{a_4 + a_5 z}{(z^2 - 0.8z + 0.12)} \\
 \Rightarrow a_1 \left[\frac{1}{2z-1} \right] &= 8 \cdot \frac{(z+0.2)(z-0.5)}{z^2 - 1} \xrightarrow[z=\frac{1}{2}]{} 0 \\
 a_2 \left[\frac{1}{z+0.5} \right] &= 8 \cdot \frac{(-0.5+0.2)(-0.5-0.5)(-0.5+0.1)}{(2z^2 - 1)(-0.5+0.1)(-0.5+0.12)} \\
 &= \frac{8(-0.3)(+0.4)}{+2z^2 + 0.1z (-0.25 + 0.4 + 0.12)} = \frac{3.2}{1.84} \approx 1.65 \\
 a_3 \left[\frac{1}{z+0.1} \right] &= 8 \cdot \frac{(0.4)(+0.8)(-0.2)}{(+1.6)(+0.2)(0.1^2 + 0.24 + 0.12)} \\
 &= \frac{-0.512}{0.208} \approx -2.46
 \end{aligned}$$

- i.** Bu şıkta aşağıdaki şekil 1.i.1'de görüldüğü gibi bir kapalı çevrim kontrol sistemi oluşturulmuştur.



Şekil 1.i.1: Oluşturulan Kapalı Çevrim Sistemi

Buna bağlı olarak bu sistemin kararlı olduğu K değerlerinin bulunması beklenmiştir. Bu K değerini bulabilmek için Jury testinden faydalانılmıştır. Jury testini yaparken 3 farklı kontrol yapılması gerekmektedir. Bu durum ve yapılan işlemler aşağıdaki gibi şekil 1.i.2'de gösterilmiştir.

SORU (i)

$$G(z) = \frac{4z^3 + 0.12z^2 - 0.6z - 0.14}{z^4 - 0.17z^3 + 0.196z^2 + 0.048}$$

$$KGTF = \frac{K \cdot G(z)}{1 + K \cdot G(z)} = \frac{K \cdot (4z^3 + 0.12z^2 - 0.6z - 0.14)}{1 + K \cdot (4z^3 + 0.12z^2 - 0.6z - 0.14)}$$

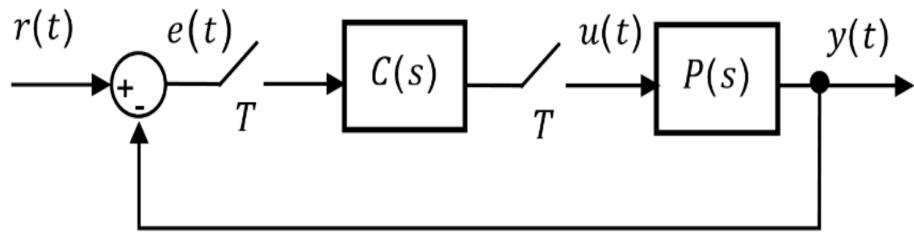
Jury testi ile batalım:

- 1) $|a_4| < a_0$, olmalı $\Rightarrow |-0.14K + 0.048| < 1 \Rightarrow -1 < -0.14K + 0.048 < 1 \Rightarrow -1.048 < -0.14K < 0.952 \Rightarrow -6.8 < K < 7.5$
- 2) $P(1) > 0$ olmalı $\Rightarrow 1 + K \cdot 1 + 1^2(0.12K - 0.17) + 1(-0.6K + 0.196) + (-0.14K + 0.048) > 0 \Rightarrow 3.38K + 1.048 > 0 \Rightarrow K > 0.3$
- 3) $P(-1) > 0$ olmalı $\Rightarrow (-1)^4 + 1K(-1)^3 + (-1)^2(0.12K - 0.17) + (-1)(-0.6K + 0.196) + (-0.14K + 0.048) > 0 \Rightarrow 1 - 4K + 0.12K - 0.17 + 0.6K - 0.17K - 0.14K + 0.048 > 0 \Rightarrow 0.742 > 3.42K \Rightarrow K < 0.22 \Rightarrow -0.3 < K < 0.22$

Şekil 1.i.2: K Değerinin Bulunduğu İşlemler

Bu görselden görüldüğü gibi K değeri -0.3 'ten büyük ve 0.22 'den küçük olmalıdır. Yani K değeri için $-0.3 < K < 0.22$ aralığında olmalıdır denilir.

SORU 2



Şekil 2.1: Soruda Verilmiş Olan Kapalı Çevrim Sistemi

Bu soruda yukarıda şekil 2.1'de verilen kapalı çevrim sisteminde örneklemme periyodu $T = 1$ saniye olup $P(s)$ için z dönüşümü aşağıdaki denklem 2.1'deki gibi verilmiştir.

$$P(z) = \frac{1}{z^2 - 1.4z + 0.48} \quad (2.1)$$

Burada $C(s) = K_p$ şeklinde bir P kontrolcüsüdür ve $K_p > 0$ 'dır. Bunun yanı sıra kapalı çevrim sönüm oranı olan ζ , 0.7'den büyük ve kapalı çevrim doğal frekansı w_n 'de $\pi/8T$ 'den büyüktür. Burada T değeri 1 olarak alınacağı için w_n 'in de $\pi/8$ 'den büyük olması beklenmektedir. Buna göre sırasıyla aşağıdaki işlemleri gerçekleştirilmişlerdir:

- a. Yukarı verilen şartlar için aşım (M_p) ve %2'lik oturma zamanının (t_s) olması gereken değer aralıkları aşağıdaki denklem 2.2 ve 2.3'teki şeklindedirler:

$$M_p = e^{-\left(\frac{\zeta\pi}{\sqrt{1-\zeta^2}}\right)} = e^{-\left(\frac{0.7\pi}{\sqrt{1-0.7^2}}\right)} \cong 0.013 \quad (2.2)$$

Bu işlem sonucuna bakılarak M_p aşım değerinin 0.013'ten büyük olması beklenmektedir. Çünkü denklem 2.2'deki hesaplama ζ 'nın limit değeri olan 0.7'ye göre yapılmıştır. Kapalı çevrim sönüm oranının artması M_p aşımını düşüreceğinden, M_p 'nin değer aralığı $M_p < 0.013$ şeklinde olmalıdır.

$$t_{\%2} = \frac{4}{\zeta\omega_n} = \frac{4}{\frac{0.7\pi}{8}} = 14.56 \text{ s} \quad (2.3)$$

Yukarıdaki denklem sonucuna göre $t_{\%2}$ değerinin 14.56 saniyeden küçük olması gerekmektedir. Çünkü denklem 2.3'teki hesaplama ζ 'nın limit değeri olan 0.7'ye ve w_n 'nin limit değeri olan $\pi/8$ göre yapılmıştır. Yani normalde bu değerlerden büyük veya küçük değerlerin seçilmesi gerekmektedir. Bunun için de ζ 'nın ve w_n 'nin artması $t_{\%2}$ değerini düşürecekinden, $t_{\%2}$ değer aralığı $t_{\%2} < 14.56$ saniye olmalıdır.

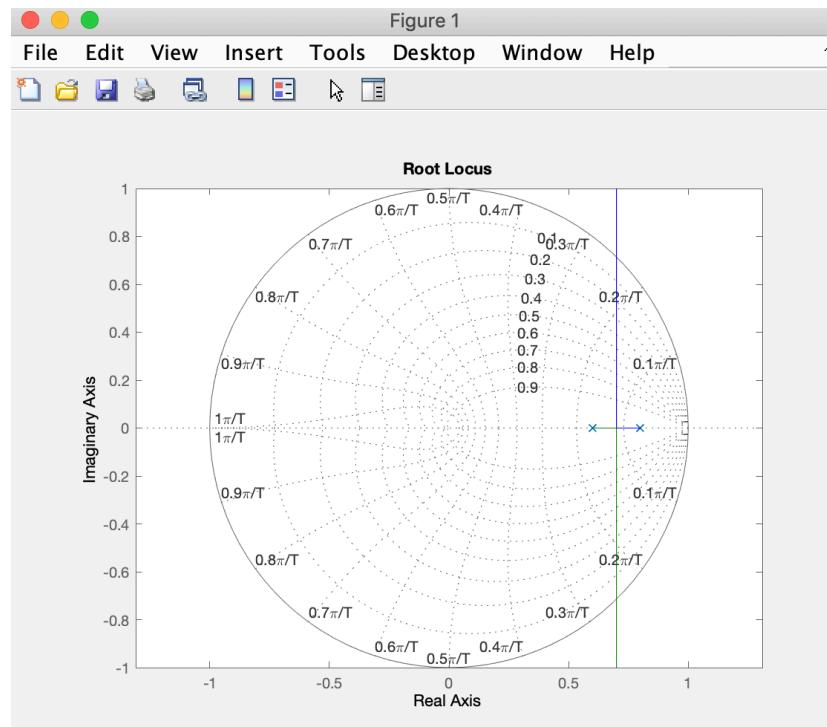
- b.** Bu sistemin kök yer eğrisini çizdirmek için bir Matlab Script kullanılarak şekil 2.b.1'de gösterildiği gibi bir kod parçası yazılmıştır. Bu grafik rlocus komutu kullanılarak çizdirilirse aşağıdaki şekil 2.3'teki gibi bir grafik elde edilir:

```
%Soru2b, Hande Yıldırım, 141201047
%verilen P sistemi için kök yer eğrisini çiziren grafik için yazılan kod

%P sistemi transfer fonksiyonu cinsinden oluşturuldu
P = tf([1],[1 -1.4 0.48],-1)
% Kok yer eğrisi rlocus ile çizdirildi.
rlocus(P)
zgrid
% eksenlerin sınırları axis ile belirlendi
axis equal
```

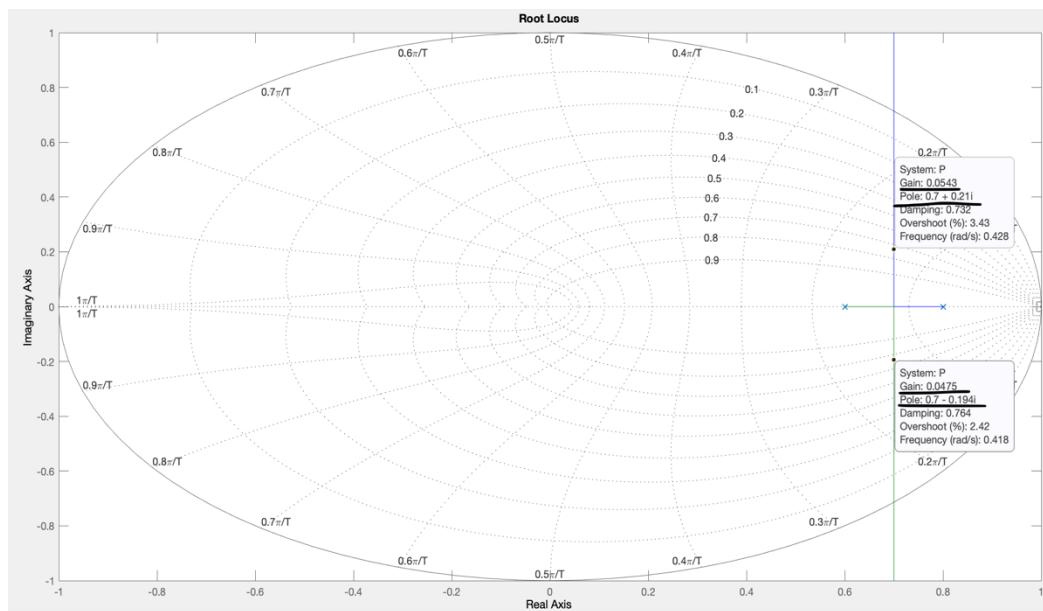
Şekil 2.b.1: Kök Yer Eğrisi İçin Yazılan Kod Parçası

Buradaki kodda öncelikle P sistemi bir transfer fonksiyon olarak tanımlanmıştır. Sonrasında rlocus ile grafik çizdirilip, zgrid komutu ile de kılavuz çizgileri belirlenmiştir. Son olarak ise eksen uzunlukları axis komutu kullanılarak eşit olarak ayarlanmıştır.



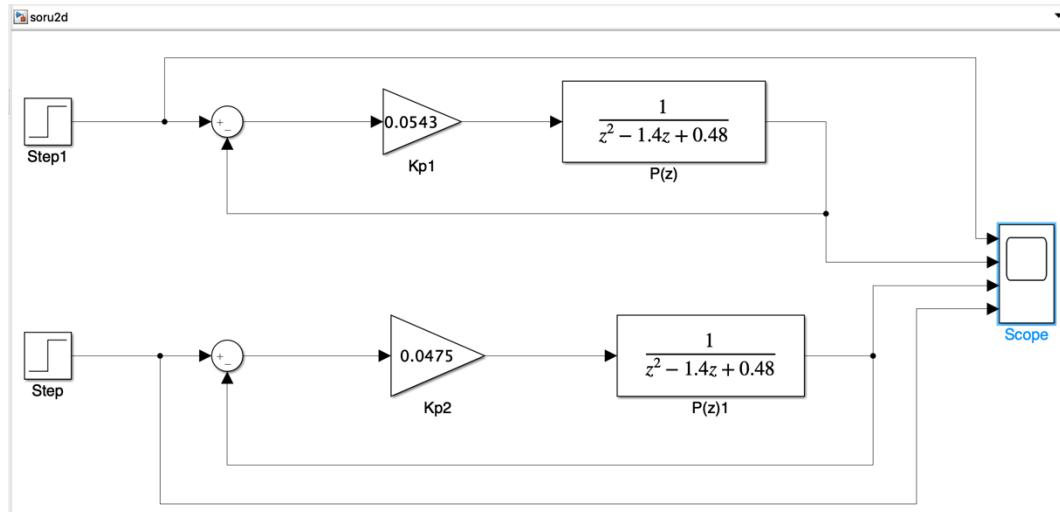
Şekil 2.b.2: Kök Yer Eğrisini Gösteren Grafik

- c. Bu aşamada çizdirilen grafik kullanılarak grafik üzerinde iki farklı K_p değerinin seçimi yapılmıştır. Seçilen bu K_p değerleri aşağıdaki şekil 2.c.1'deki şekilde siyah kutucuklar içerisinde de alındığı gibi $K_{P1} = 0.0543$ ve $K_{P2} = 0.0475$ şeklindedirler.



Şekil 2.c.1: Seçilen İki Farklı K_p Kontrolü Değerlerini Gösteren Grafik

d. Bu K_p değerlerine bağlı olarak Simulink üzerinde kapalı çevrim sistemleri aşağıdaki şekil 2.d.1'deki gibi tekrar kurulmuştur.



Şekil 2.d.1: Yeni KP Değerlerine Uygun Oluşturulan Kapalı Çevrim Sistemleri

Bu sistemlerin birim basamak cevapları ve giriş sinyalleri aşağıdaki şekil 2.d.2'deki grafikteki gibi elde edilmişlerdir. Hangi grafiğin hangi sisteme bağlı oldukları kullanılarak grafik üzerinde gösterilmiştir.



Şekil 2.d.2: K_{p1} ve K_{p2} Kontrolcülerinin Kullanımı Sonucu Sistemden Elde Edilen Grafikler

Yukarıdaki şekil 2.d.2'den de anlaşılacağı gibi Tip0 için e_{ss} değerleri yani hatalar mevcuttur. Bu hata değerleri aşağıdaki denklem 2.4a, denklem 2.4b, denklem 2.5a ve denklem 2.5bdteki gibi elde edilebilir.

$$e_{ss1} = \frac{1}{1+K_{P1}} = \frac{1}{1+0.0563} \cong 0.946 \quad (2.4a)$$

$$e_{ss2} = \frac{1}{1+K_{P2}} = \frac{1}{1+0.0475} \cong 0.955 \quad (2.4a)$$

Sürekli hal hatasının küçük olması için K mümkün olduğunca büyük olmalı. Seçilebilecek maksimum K değeri için aşağıda şekil 2.d.3'teki işlemler yapıldı:

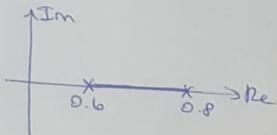
$$P(z) = \frac{1}{z^2 - 1.4z + 0.48} = \frac{1}{(z-0.6)(z-0.8)}$$

$$C(z) = K_p \quad \left. \right\} AGTF = C(z) \cdot P(z)$$

$$AGTF = \frac{K_p}{(z-0.6)(z-0.8)}$$

↓
kutuplar = 0.6, 0.8
sifirlar = yok

1) reel eksen



2) asymptoller

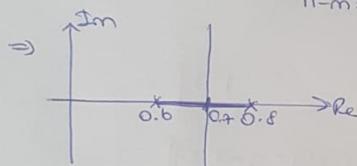
$$n-m = 2-0 = 2 \text{ tane}$$

$$\alpha_{all} = \frac{180^\circ (2k+1)}{n-m}$$

$$= \frac{90^\circ (2k+1)}{1} \rightarrow 90^\circ$$

$$270^\circ$$

$$\text{kesim noktası (Ta)} = \frac{[0.6 + 0.8] - [0]}{n-m=2} = 0.7$$



3) Aynılma/Birleşme Noktaları

$$KGTF = \frac{\frac{K_p}{(z-0.6)(z-0.8)}}{1 + \frac{K_p}{(z-0.6)(z-0.8)}} = \frac{K_{GTF}}{z^2 - 1.4z + 0.48 + K_{GTF}}$$

\Rightarrow Karakteristik Denklemleri $z^2 - 1.4z + 0.48 + K_{GTF} = 0$

$$\Rightarrow \frac{dK_{GTF}}{dz} = 2z - 1 = 0 \Rightarrow z = 0.5$$

$$\tan \beta = \frac{\sqrt{1-z^2}}{z} = \frac{\sqrt{1-0.5^2}}{0.5} = 1.02 \Rightarrow \boxed{\beta < 45^\circ} \text{ olmali}$$

\Rightarrow Bu deire matlab'da çizdirilince

$$K_{GTF} = |s_1 - (0.6)| \cdot |s_1 - (0.8)|$$

grafikte bulunsan
pole = $0.7 + 0.21j$
bşpn'ı içinde
 \Rightarrow

Karakteristik Denklemleri $\rightarrow z^2 - 1.4z + 0.48 + K_{GTF} = 0$

$$\therefore w_n = -1.4 \Rightarrow w_n = -0.7 \rightarrow j > 0.7 \Rightarrow w_n > -1$$

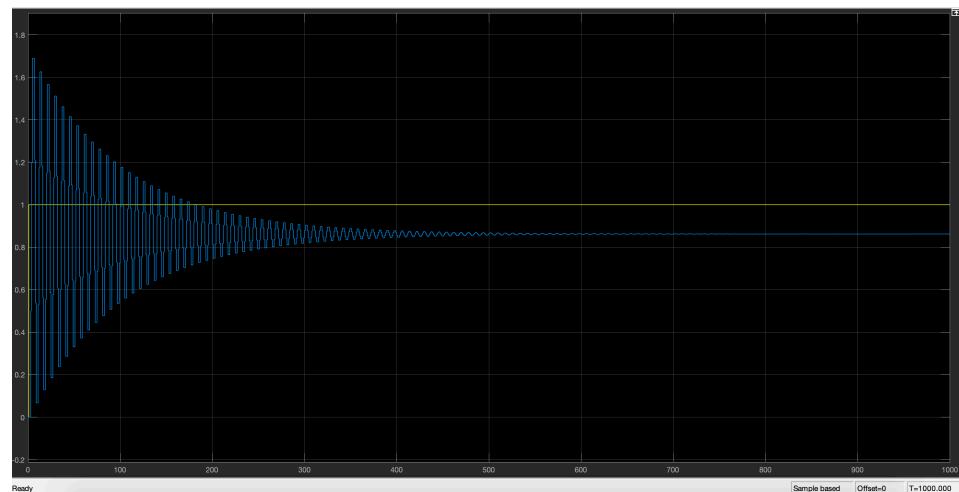
$$w_n^2 = 0.48 - k$$

$$\Rightarrow k = w_n^2 - 0.48$$

$$\Rightarrow K_{GTF} \leq 0.52 \xrightarrow{K_{GTF} \leq 0.52} e_{ss} = \frac{1}{1 + 0.52} = 0.66$$

Şekil 2.d.3: K_p Değer Seçimi

Bu görseller sonucunda bulunan maksimum K değerine göre tekrar devre tasarlandı ve simüle edildi. Böylece aşağıda şekil 2.d.4'teki grafik elde edildi.



Daha sonrasında tekrar aşım ve oturma zaman değerleri bulunmaya çalışıldı. Fakat burada sonuçlar yaklaşık olarak elde edilmiştir. Çünkü hata çıktıgı için sistem tam 1'e tam olarak yakınsayamamıştır. Buradan:

$P(z)$ için:

$$M_{P1} \cong 1.7 * \%3.43 = 0.5831 \quad (2.4a)$$

$$t_{\%2_1} = 600s \quad (2.4b)$$

- e. Elde edilen bu aşım ve oturma değerleri yukarıdaki a şıkkında teorik olarak elde edilen aşım ve oturma değerleri ile kıyaslandıkları zamanmayı olmadıkları görülür. Bu durum hem yapılan teorik hesaplamaların hem de bilgisayar simülasyonu ile yapılan çıkarımların ne yazık ki tutarlı olmadıklarını ve elde edilen sonuç doğru olmadığını bir kanıtıdır denilebilir. Buna sebep ise e_{ss} yani hata değeridir denilebilir.