

Óbudai egyetem



Bánki Donát Gépészmérnöki Kar

Cruise Control(Tempomat)

Projektum

Rendszer és irányításelmélet

Mechatronika MSC levelező szak

Kihelyezett tagozat, Szabadka

Hallgatók: Szegedi Mihály(P0KPOE)
Papp Dorottya(IAV5VX)
Sápi Róbert(A9D8R0)
Törőcsik Áron(U6KRFW)

Előadó: Dr. Pletl Szilveszter

Szabadka, 2018.

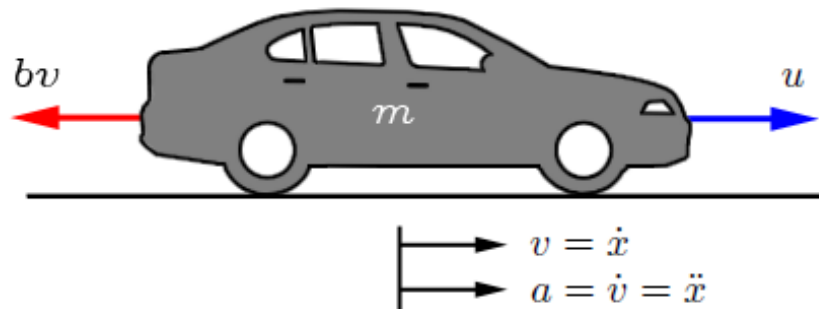
Tartalomjegyzék

A rendszer modellezése	3
A rendszer fizikai modellje	3
Egyenletek.....	3
Paraméterek.....	4
Állapottér modell	4
Átviteli függvény	4
Rendszer elemzése	5
Rendszer modell és paraméterek	5
Teljesítmény meghatározás.....	5
Nyílt hurkú lépés-válasz függvény	5
PID vezérlő	6
Simulink Modellezés	8
Vezérlőtervezés	11

A rendszer modellezése

A rendszer fizikai modellje

Az automatikus sebességtartó automatika kiváló példája a sok modern járműben található visszacsatolás-vezérlési rendszernek. A sebességszabályozó rendszer célja, hogy a külső hatások ellenére állandó járműsebességet tartson fenn, például szél- vagy közúti minőségváltozást. Ezt a jármű sebességének mérésével végezzük, összehasonlítva a kívánt vagy referencia sebességgel, és automatikusan szabályozzuk a fojtószelepet szabályozási algoritmus szerint.



A gépjármű modellezése

Itt a jármű dinamikájának egy egyszerű modelljét vesszük figyelembe, amelyet a „free-body diagram”-ban (FBD) mutatunk be. A m tömegű jármű melyet egy u erő hajt. Az u erő az út / gumibroncs felületén keletkező erőt jelenti. Ehhez az egyszerűsített modellhez feltételezzük, hogy közvetlenül ellenőrizhetjük ezt az erőt, és figyelmen kívül hagyjuk a hajtómű, a gumibroncsok stb. dinamikáját, amelyek az erő létrehozását eredményezik. A gördülési ellenállás és a légellenállás miatt a „ bv ” rezisztív erőről feltételezzük, hogy lineárisan változik a jármű sebességével (v), és a jármű mozgásával ellentétes irányba hat.

Egyenletek

Ezekkel a feltevésekkel egy elsőrendű tömegállandós rendszer maradt. Összeadva az erők az x -irányba és Newton 2. törvényének alkalmazásakor a következő egyenletrendszerre jutunk:

$$m\dot{v} + bv = u$$

Mivel a jármű sebességének ellenőrzésére vagyunk kíváncsiak, a kimeneti egyenlet az alábbiak szerint kerül kiválasztásra:

$$y = u$$

Paraméterek

Ehhez a példához tegyük fel, hogy a rendszer paraméterei:

- jármű tömege(m): 1000 kg
- Csillapítási együttható: 50 Ns/m

Állapottér modell

Az elsőrendű rendszereknek csak egyetlen energiatárolási módjuk van, ebben az esetben az autó kinetikus energiája, ezért csak egy állapotváltozó szükséges, a sebesség. Az állapottér reprezentáció ezért:

$$\dot{x} = [\dot{v}] = \left[\frac{-b}{m} \right] [v] + \left[\frac{1}{m} \right] [u]$$

$$y = [1][v]$$

Ezt az állapot-tér modellt MATLAB-ba tesszük a következő parancsokkal:

```
m = 1000;  
b = 50;  
A = -b/m;  
B = 1/m;  
C = 1;  
D = 0;  
cruise_ss = ss(A,B,C,D).
```

Átviteli függvény

A vezérlés differenciálegyenletének Laplace transzformációját figyelembe véve, és nullázó kezdeti feltételeket feltételezve a sebességtartó automatika átviteli függvényét a következőképpen találjuk:

$$P(s) = \frac{V(s)}{U(s)} = \frac{1}{ms + b} \left[\frac{m/s}{N} \right]$$

Az átviteli függvény modellt MATLAB-ba tesszük a következő parancsokkal:

```
s = tf('s');  
P_cruise = 1/(m*s+b);
```

Rendszer elemzése

Rendszer modell és paraméterek

A sebességszabályozó probléma átviteli függvény modellje az alábbiakban található.

$$P(s) = \frac{V(s)}{U(s)} = \frac{1}{ms + b} \left[\frac{m/s}{N} \right]$$

A példában használt paraméterek a következők:

- jármű tömege(m): 1000kg
- csillapítási tényező(b): 50Ns/m
- névleges szabályzóerő(u): 500N

Teljesítmény meghatározás

A következő lépés olyan tervezési kritériumok kidolgozása, amelyeket a kompenzált rendszernek el kell érnie. Amikor a motor egy 500 Newtonos erőt ad, akkor az autó eléri a maximális sebességét 10m/s. Az autónak képesnek kell lennie arra, hogy 5 másodpercen belül elérje ezt a sebességet. Maximális megengedett túllövés 10%, az egyensúlyi hiba pedig 2% -os legyen.

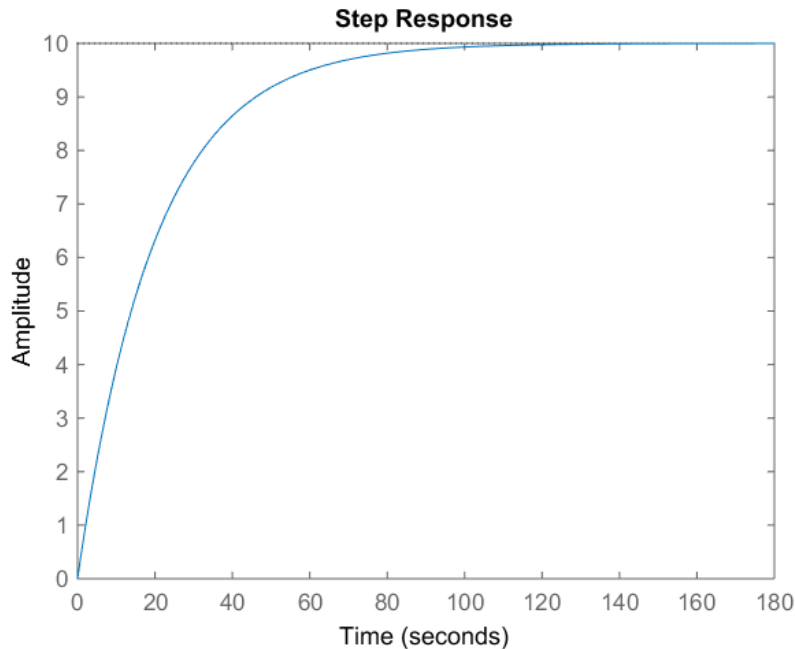
Nyílt hurkú lépés-válasz függvény

A rendszer nyílt hurkú válaszát, visszacsatolás nélküli vezérlés nélkül, az 500 Newton egy lépcsős bemeneti erejét a MATLAB-ban szimulálják az alábbiak szerint:

```

m = 1000;
b = 50;
u = 500;
s = tf('s');
P_cruise = 1/(m*s+b);
step(u*P_cruise):

```



Látjuk, hogy a nyílt hurkú rendszer nem mutat túllövést vagy oszcillációkat (az elsőrendű rendszerek jellemzői), és elérheti a kívánt 10 m / s állandó sebességet; azonban az gyorsulási idő túlságosan lassú, 60 s. Ezért olyan visszacsatolásvezérlőt kell kialakítanunk, amely jelentősen felgyorsítja a választ, anélkül, hogy negatív hatással lenne a többi dinamikus teljesítménymutatóra.

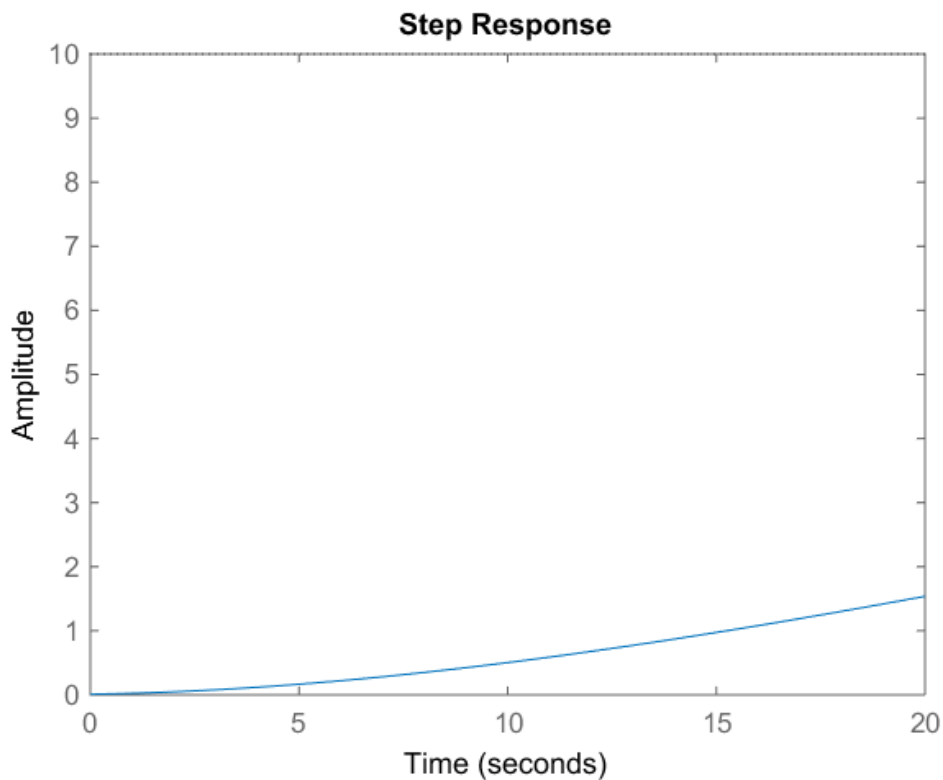
PID vezérlő

Ebben a konkrét példában nem volt szükség származékos vezérlő megvalósítására a szükséges kimenet eléréséhez. A PID szabályozóval ($C = K_p + \frac{K_i}{s} + K_d s$) rendelkező sebességszabályozó rendszer zárt hurkú átviteli funkciója a következő:

$$T(s) = \frac{Y(s)}{R(s)} = \frac{P(s)C(s)}{1 + P(s)C(s)} = \frac{K_p s^2 + K_p s + K_i}{(m + K_d)s^2 + (b + K_p)s + K_i}$$

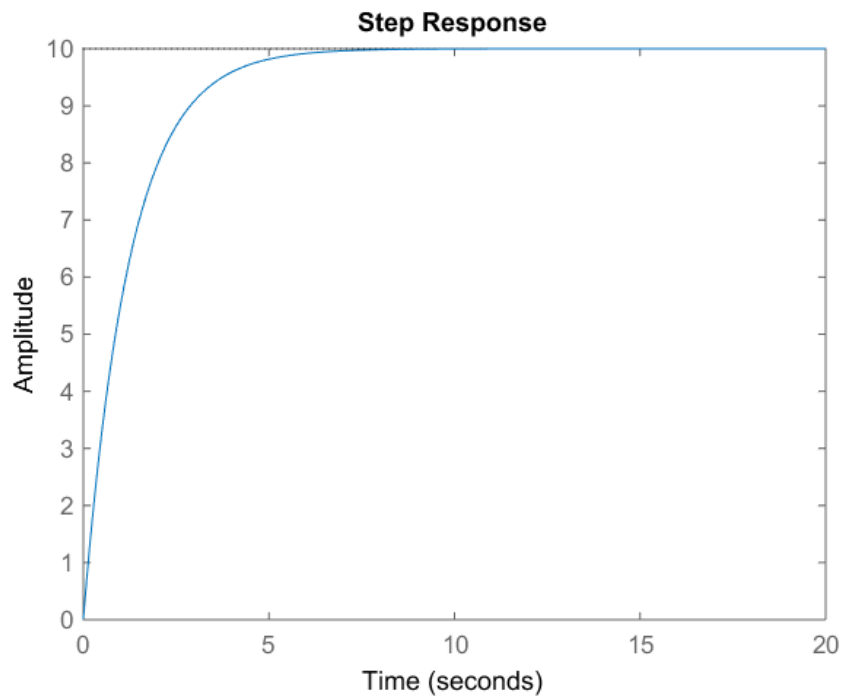
Legyen K_p egyenlő 1, K_i egyenlő 1 és K_d egyenlő 1, és írja be az alábbi parancsokat egy új m-fájlba.

```
Kp = 1;  
Ki = 1;  
Kd = 1;  
C = pid(Kp,Ki,Kd);  
T = feedback(C*P_cruise,1);
```



A K_p , K_i és K_d értékek beállítása a után a következő eredményt kaptuk:

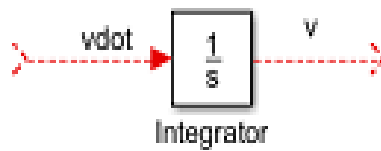
```
Kp = 800;  
Ki = 40;  
Kd = 1;  
C = pid(Kp,Ki,Kd);  
T = feedback(C*P_cruise,1);  
step(r*T,t)  
axis([0 20 0 10])
```



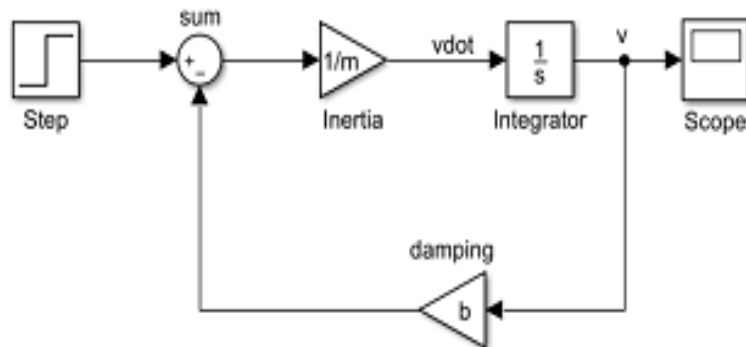
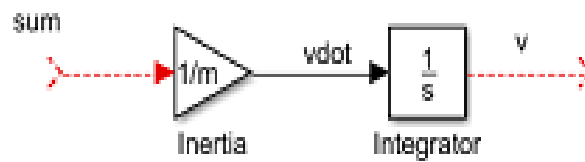
Simulink Modellezés

A sebességszabályzó modellje viszonylag egyszerű. Ha feltételezzük, hogy a gördülési ellenállás és a légellenállás arányos az autó sebességével, akkor a probléma az alábbiakban bemutatott egyszerű tömeg- és szeleprendszerre korlátozódik. Ezt a rendszert a tömegre ható erők összegzésével és a gyorsulás integrálásával a sebesség növelésével modellezzük. A Simulink alkalmazást megnyitva először a gyorsulás integrálódását fogjuk modellezni.

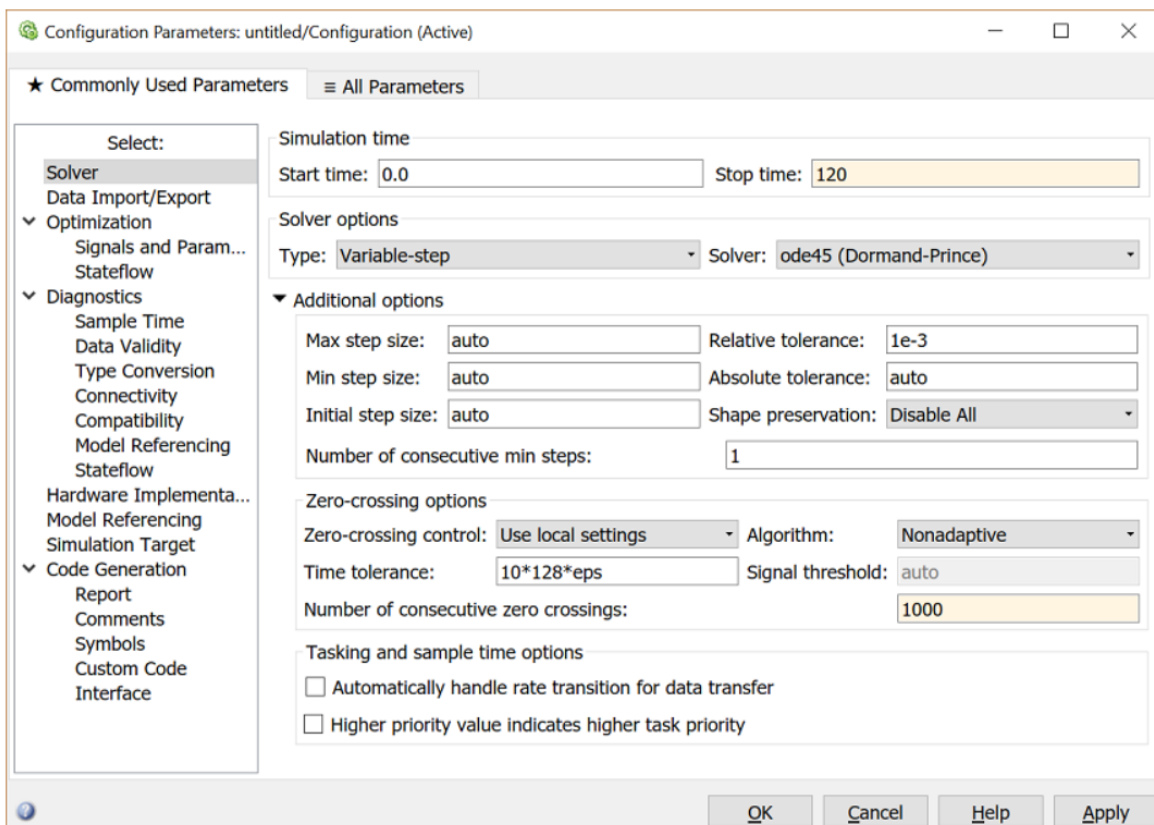
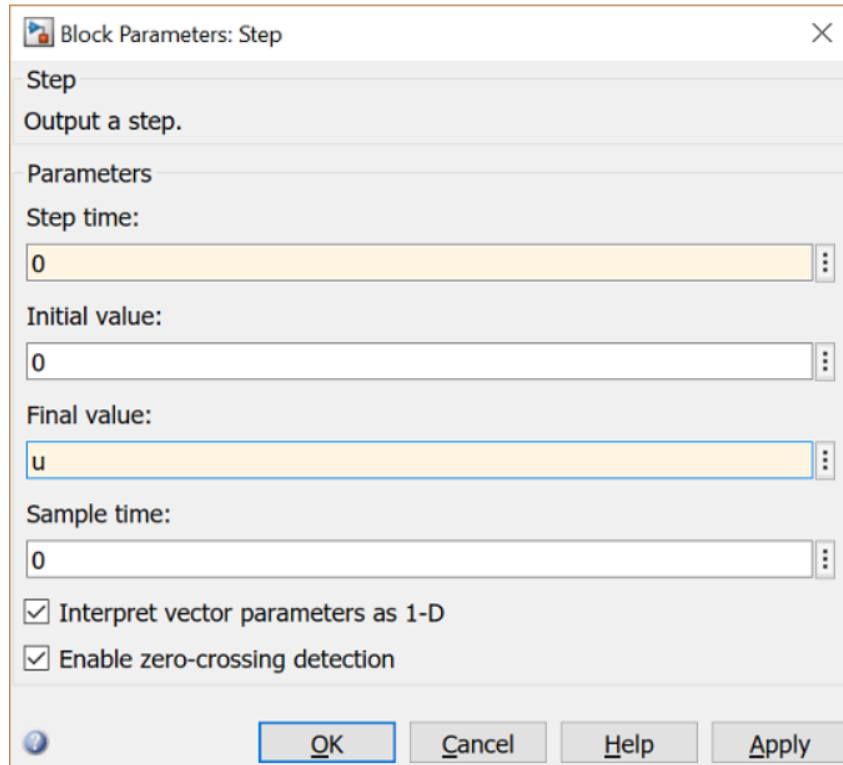
$$\int \frac{dv}{dt} dt = v$$



Mivel a gyorsulás (dv / dt) megegyezik a tömeggel osztott erők összegével, a bejövő jelet a tömeggel osztjuk el.



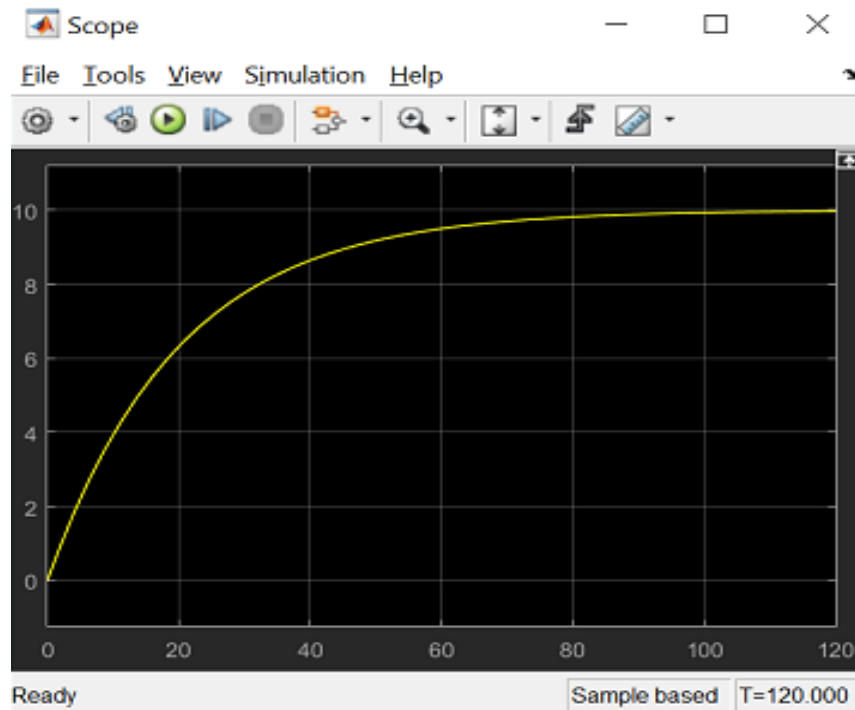
Hogy a szimuláció megfelelő megfelelő grafikon-t mutasson, a következő paramétereket kell beállítani:



Ezután be kell állítani a fizikai paramétereket a MATLAB parancssorában:

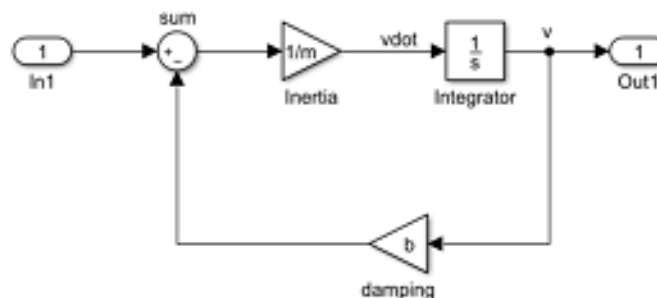
```
m = 1000;  
b = 50;  
u = 500;
```

Az eredmény az alábbi kimeneten látható:



Vezérlőtervezés

A rendszer lineáris modellje (az állami térben vagy az átviteli függvényformában) egy Simulink modellből a MATLAB-ban nyitható. Ez az In1 és az Out1 blokkok használatával és a MATLAB függvény `linmod` használatával történik.



```

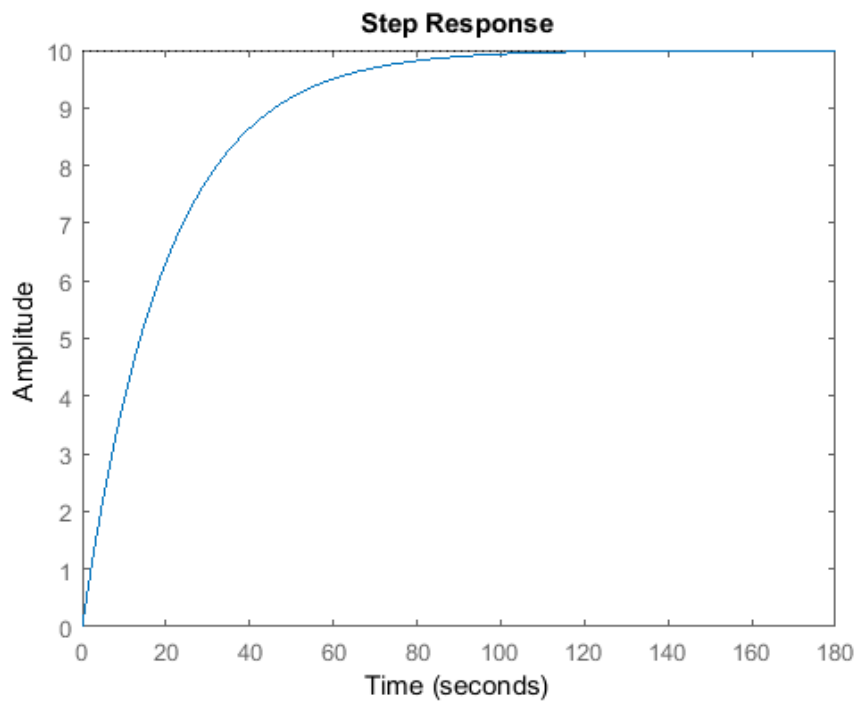
m = 1000;
b = 50;
u = 500;
[A,B,C,D] = linmod('ccmodel')
cruise_ss = ss(A,B,C,D);

A = -0.0500
B = 1.0000e-03
C = 1
D = 0

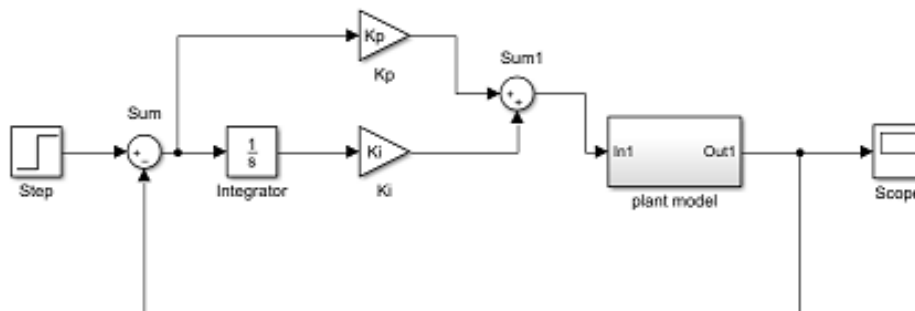
```

A modellkivonás igazolásához a kibővített átviteli függvény nyílt hurkú lépéses válaszát generáljuk a MATLAB-ban. A számlálót 500-mal meg kell szorozni, hogy szimuláljuk az 500 N lépések bemenetet. Adjuk meg a következő parancsot a MATLAB-ban.

```
step(u*cruise_ss)
```



Mint láthatjuk a célsebesség eléréséhez majdnem egy percre van szükség . A szabályzót a következő PI szabályzó modelljével szerkesztjük meg:



A tervezési követelmények 5 másodpercesnél kisebb emelkedési időt tartalmaztak, ezért 10 másodpercig szimuláljuk a kimenetet. Most be kell állítani a fizikai paramétereket. Futtassuk a következő parancsokat a MATLAB parancssorában.

```
m = 1000;
b = 50;
r = 10;
Kp = 800;
Ki = 40;
```

