

1. Lagrangian Mechanics

Lagrangian 역학

1788년에 이탈리아 태생의 수학자이자 천문학자인 Joseph-Louis Lagrange에 의해 소개된 역학을 표현하는 방법의 뉴턴 역학(고전 역학)과는 다른 또 다른 형태이다. Lagrangian mechanics는 시스템의 이동 경로를 해석하는데 있어, Lagrange equation을 해석하여 해당 시스템을 해석한다.

뉴턴 역학에서 외부에서 물체에 미치는 힘에 중점을 두고 벡터량들을 주로 다루는 반면, Lagrangian 역학에서는 물체의 운동 에너지와 위치 에너지와 같은 스칼라량을 주로 다루게 된다.

Generalized 좌표계

뉴턴 역학은 벡터량을 주로 다루기 때문에 벡터량들을 다루기 쉬운 직교좌표계를 주로 사용한다. 반면 Lagrangian 역학에서는 구속 조건에 의한 자유도의 개수만큼의 Generalized 좌표계를 사용한다. 여기서 Generalized 좌표계는 물리적 계를 더 쉽게 분석하기 위해 설정하는 좌표계를 말한다. Generalized 좌표계를 사용하게 되면 구속력에 의한 조건을 제거하여 계산을 더 쉽게 할 수 있다.

여기서 자유도는 물체의 운동을 표현하기 위해 필요한 변수의 개수를 말하며, 자유도의 개수는 3차원의 공간에서 $3N-C$ 에 해당한다. 여기서 N 은 시스템의 입자의 개수이고, C 는 구속 조건의 개수이다.

Lagrangian & Lagrange's equations

Lagrangian L 은 다음과 같다.

$$L = T - V$$

여기서 T 는 시스템의 각 입자들의 Kinetic 에너지의 합이며, V 는 각 입자들의 Potential 에너지의 합이다.

Lagrange's equation은

$$\frac{\partial L}{\partial r_k} - \frac{d}{dt} \frac{\partial L}{\partial \dot{r}_k} + \sum_{i=1}^C \lambda_i \frac{\partial f_i}{\partial r_k} = 0$$

이며, 여기서 k 는 $1 \sim N$ 이며, r_k 는 각 입자의 Position 벡터, C 는 구속 조건의 개수 λ_i 해당 입자의 Lagrange multiplier, f_i 는 구속 조건 식이다.

각각의 r_k 를 n 개의 generalized 좌표계의 position 벡터 q 로 나타내면

$$r_k = r_k(q, t) = (x_k(q, t), y_k(q, t), z_k(q, t), t)$$

로 나타낼 수 있다.

이 때 \dot{q} 은 generalized velocity이고, 직교 좌표계서의 속도 v 는 다음과 같다.

$$v_k = \sum_{j=1}^n \frac{\partial r_k}{\partial q_j} \dot{q}_j + \frac{\partial r_k}{\partial t}$$

Generalized 좌표계로 Lagrange's equation을 다시 나타내면

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_j} \right) = \frac{\partial L}{\partial q_j}$$

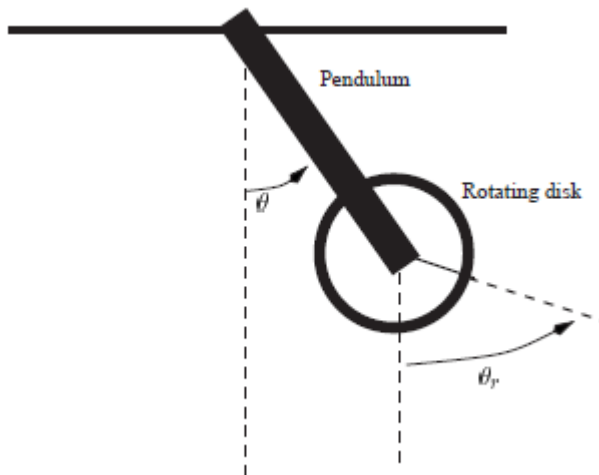
로 나타낼 수 있으며, 이는 첫 번째 Lagrange's equation과 달리 generalized 좌표계로 표현했기 때문에, 구속 조건을 고려하지 않은 자유도가 $3N-C$ 가 되며, 이 방정식은 Euler-Lagrange equations이라고도 한다.

만약 구속 조건을 무시하면 안 될 경우에는 구속조건 식을 고려해야 한다. m 개의 구속조건을 고려해야 한다면 Lagrange's equation은 다음과 같다.

$$\frac{\partial L}{\partial q_j} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_j} \right) + \sum_{i=1}^m \lambda_i \frac{\partial f_i}{\partial r_k} = 0$$

여기서 $Q_j = \sum_{i=1}^m \lambda_i \frac{\partial f_i}{\partial r_k}$ 로 정의하고, 이를 일반화 구속력이라고 한다.

2. Mechanical & Motor Modeling of the Inverted Reaction Pendulum System



$$\theta = \varphi$$

$$\theta_r = \varphi + \varphi_r$$

φ 는 수직 좌표계와 Pendulum 사이의 각도로 Pendulum의 고정된 부분에 위치한 Encoder로 측정된 각도이며, φ_r 는 Pendulum과 Wheel 사이의 상대 각도로, Pendulum과 Wheel이 연결된 부분에 위치한 Encoder로 측정된 각도이다. θ 는 수직 좌표계와 Pendulum 사이의 각도이며, θ_r 은 수직 좌표계와 Wheel 사이의 각도이다.

Reaction Wheel Pendulum은 θ 와 θ_r 두 개의 자유도를 가지고 있으며, Lagrange equation으로 해석하기 위한 Parameters는 다음과 같다.

m_p : pendulum의 질량

m_r : rotor의 질량

$m = m_p + m_r$: pendulum과 rotor의 질량

J_p : pendulum의 무게중심에서의 질량관성모멘트

J_r : rotor의 무게중심에서의 질량관성모멘트

l_p : pivot에서 pendulum의 무게중심 까지의 거리

l_r : pivot에서 rotor의 무게중심 까지의 거리

l : pivot에서 pendulum과 rotor의 무게 중심까지의 거리

$$m = m_p + m_r$$

$$ml = m_p l_p + m_r l_r$$

$$J = J_p + m_p l_p^2 + m_r l_r^2$$

Lagrange's equation을 이용하여 해석하기 위하여 kinetic energy T 와 potential energy V 를 구한다.

$$T = \frac{1}{2}J\dot{\theta}^2 + \frac{1}{2}J_r\dot{\theta}_r^2$$

$$V = mgl(1 - \cos \theta)$$

Lagrangian은 다음과 같다.

$$L = T - V = \frac{1}{2}J\dot{\theta}^2 + \frac{1}{2}J_r\dot{\theta}_r^2 + mgl(1 - \cos \theta)$$

Lagrange's equation을 풀기 위해 편미분을 구하면 다음과 같다.

$$\frac{\partial L}{\partial \theta} = -mgl \sin \theta$$

$$\frac{\partial L}{\partial \dot{\theta}} = J\dot{\theta}$$

$$\frac{\partial L}{\partial \theta_r} = 0$$

$$\frac{\partial L}{\partial \dot{\theta}_r} = J_r\dot{\theta}_r$$

구속력에 해당하는 rotor의 토크를 구하면

$$\tau = kI$$

가 되며, 이는 pendulum에 작용하는 토크는 $-\tau$ 가 된다.

위에서 구한 식으로 Lagrange's equation을 나타내면

$$J\ddot{\theta} + mgl \sin \theta = -kI$$

$$J_r\ddot{\theta}_r = kI$$

이고, 각각은 pendulum과 rotor의 Lagrange's equation이 된다.

양 변을 각각 질량관성모멘트로 나누면 아래와 같이 된다.

$$\ddot{\theta} + \frac{mgl}{J} \sin \theta = -\frac{k}{J} I$$

$$\ddot{\theta}_r = \frac{k}{J_r} I$$

외부의 입력이 없다고 가정하면

$$\ddot{\theta} + \frac{mgl}{J} \sin \theta = 0$$

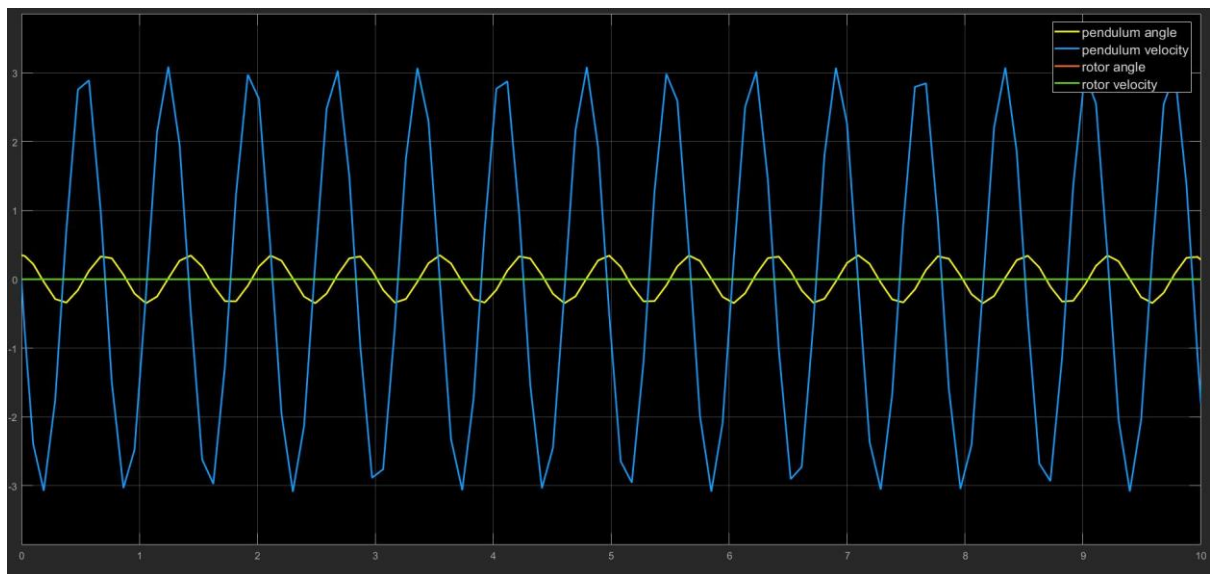
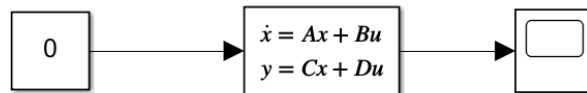
$$\ddot{\theta}_r = 0$$

이고 이를 Linearization을 하여 State Space Form으로 나타내면 다음과 같다.

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{\theta}_r \\ \ddot{\theta}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{mgl}{J} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \theta_r \\ \dot{\theta}_r \end{bmatrix}$$

$$\theta = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \theta_r \\ \dot{\theta}_r \end{bmatrix}$$

외부의 입력이 없을 때 20도의 위치에서 pendulum을 놓았을 때 pendulum과 rotor의 움직임을 simulink로 구현하여 도시하면 다음과 같다.



그러나 여기서 rotor의 움직임을 예측할 수 없는데, 그 이유는 rotor와 pendulum 사이의 마찰을 고려되지 않았기 때문이다. 그 결과, 실제로는 rotor가 pendulum에 고정되어 pendulum과 함께 진동하게 된다.

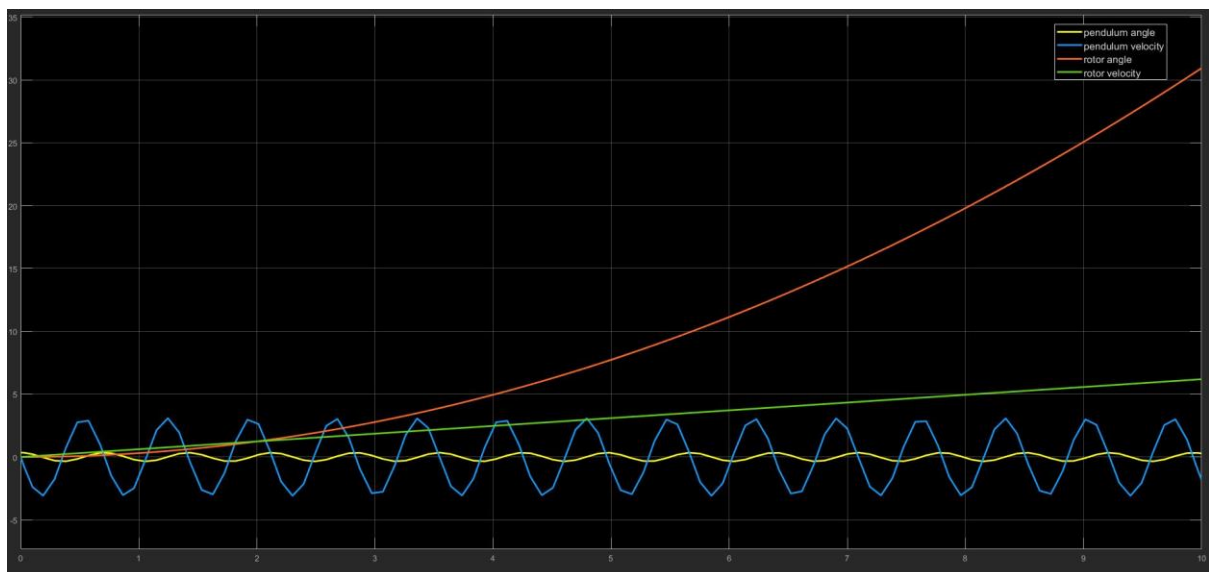
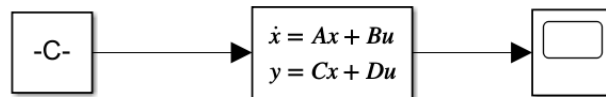
다음으로 모터의 토크가 외부의 입력으로 주어졌을 때 고려하면, 모터의 dynamics는 다음과 같다.

를 State Space Form으로 나타내면 다음과 같다.

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{\theta}_r \\ \ddot{\theta}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{mgl}{J} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \theta_r \\ \dot{\theta}_r \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{k}{J} \\ 0 \\ \frac{k}{J_r} \end{bmatrix} I$$

$$\theta = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \theta_r \\ \dot{\theta}_r \end{bmatrix}$$

이때 입력 전류 I에 일정한 Step 입력(motor torque)을 주었을 때 초기 위치가 20도일 때의 pendulum과 rotor의 움직임을 simulink로 구현하여 도시하면 다음과 같다.



모터의 Dynamics를 고려하면

$$L \frac{dI}{dt} + RI = V - k_b \dot{\phi}_r = V - k_b (\dot{\theta}_r - \dot{\theta})$$

$$I = \frac{1}{R} (V - k_b (\dot{\theta}_r - \dot{\theta})) \quad (L \approx 0)$$

이고,

$$\tau = kI$$

이므로

$$\tau = \frac{k}{R} (V - k_b (\dot{\theta}_r - \dot{\theta}))$$

가 된다.

위에서 구한 모터의 Dynamics를 토크를 입력으로 한 식을 다시 정리하면 다음과 같다.

$$\ddot{\theta} + \frac{mgl}{J} \sin \theta = -\frac{k}{JR} (V - k_b (\dot{\theta}_r - \dot{\theta}))$$

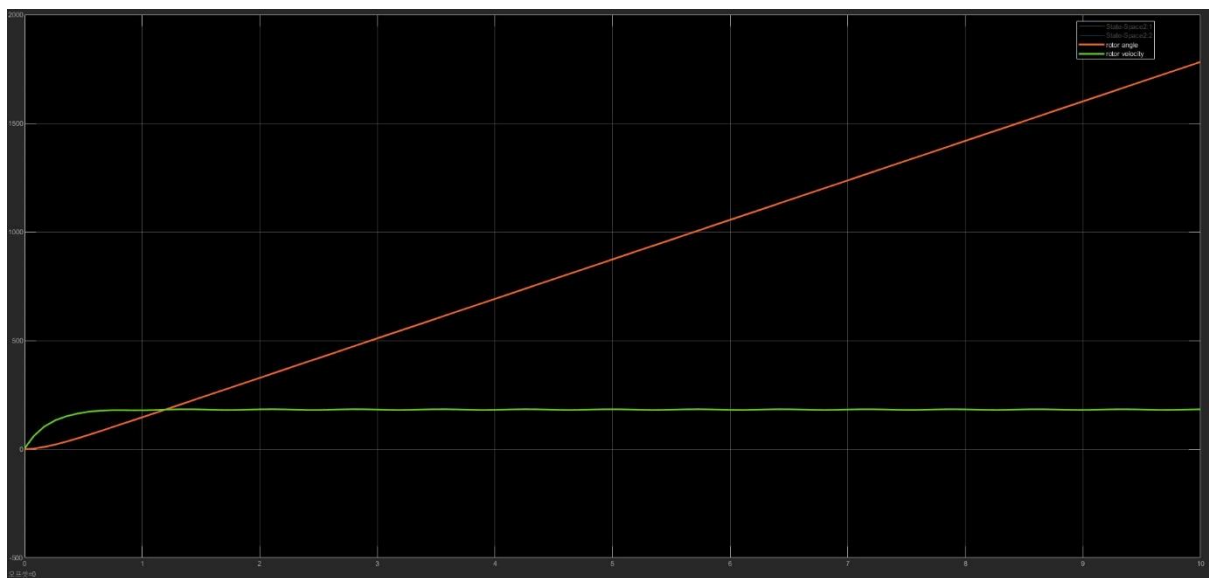
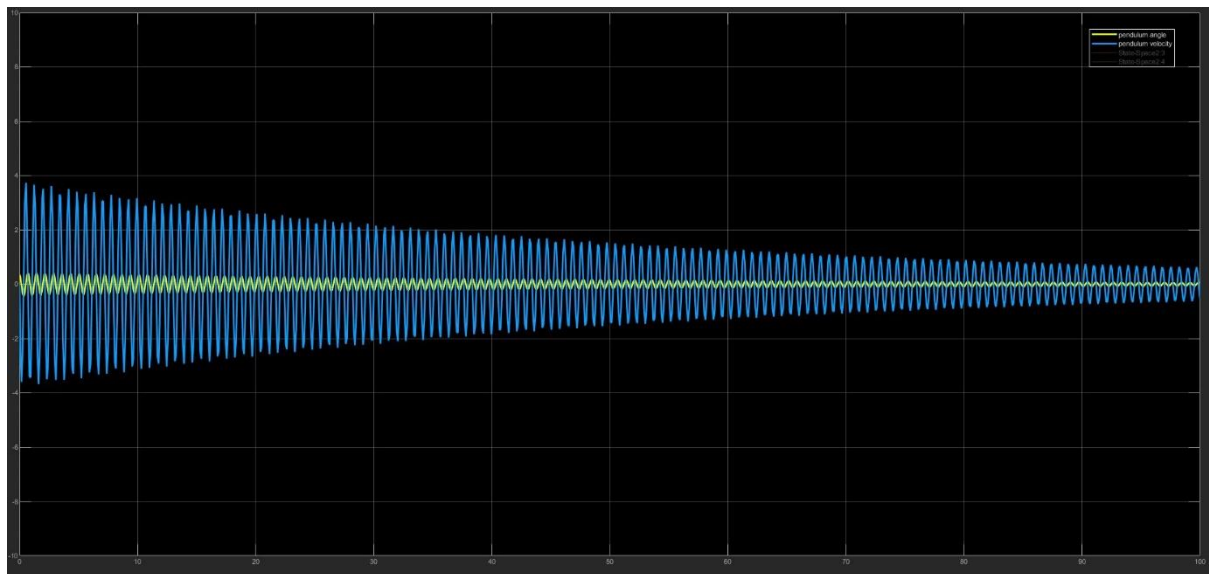
$$\ddot{\theta}_r = \frac{k}{J_r R} (V - k_b (\dot{\theta}_r - \dot{\theta}))$$

따라서 Motor dynamics를 고려했을 때 State space는 다음과 같다.

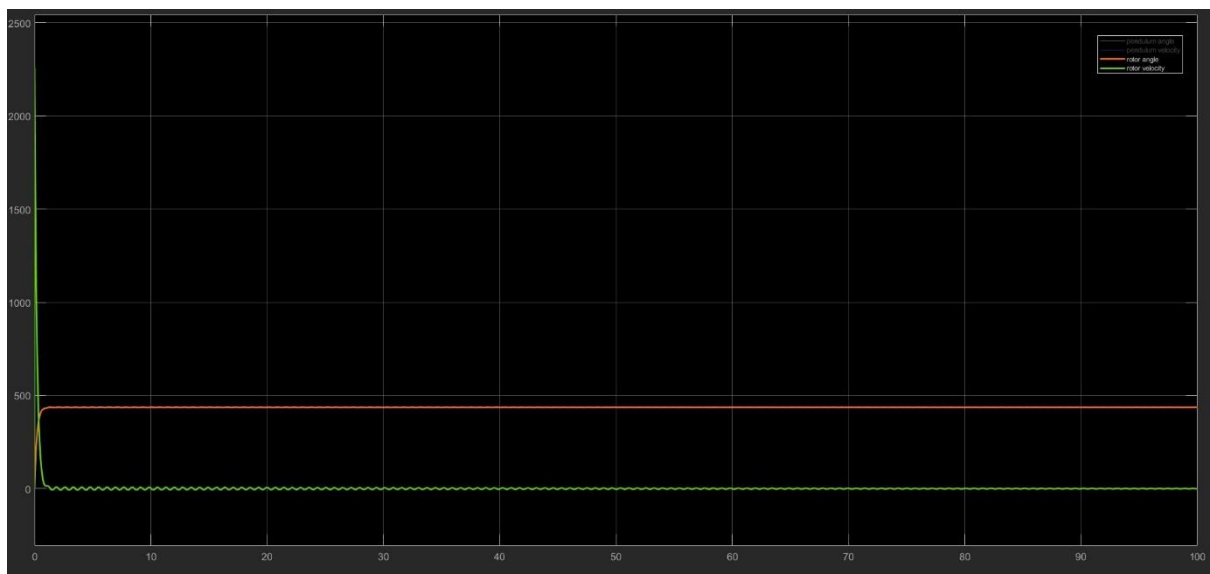
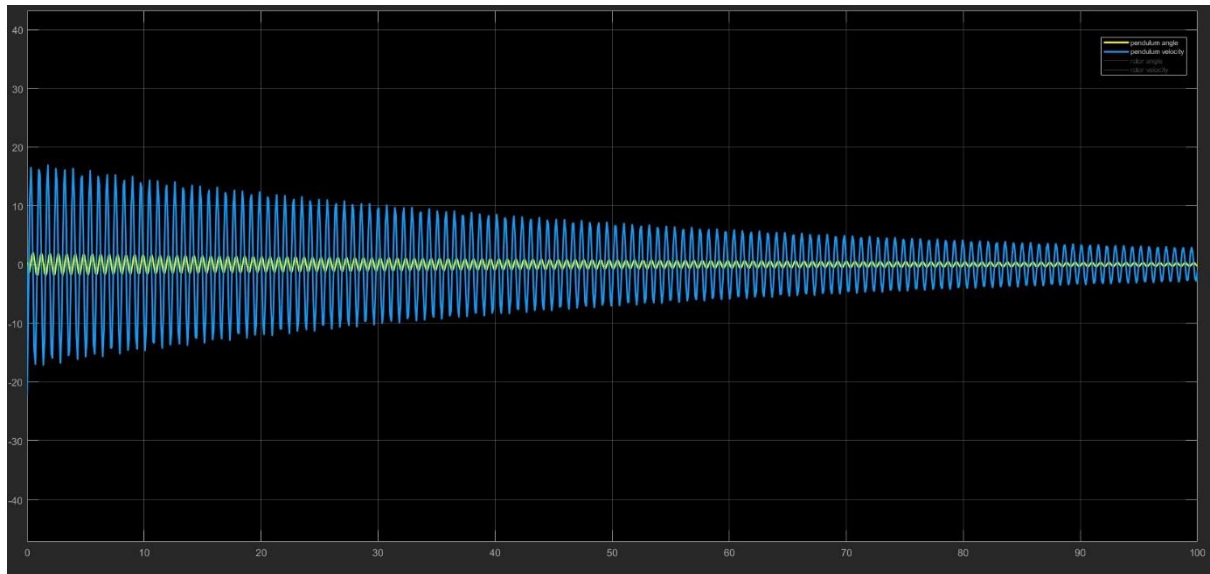
$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{\theta}_r \\ \ddot{\theta}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{mgl}{J} & -\frac{kk_b}{JR} & 0 & \frac{kk_b}{JR} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{kk_b}{J_r R} & 0 & -\frac{kk_b}{J_r R} \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \theta_r \\ \dot{\theta}_r \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{k}{JR} \\ 0 \\ \frac{k}{J_r R} \end{bmatrix} V$$

$$\theta_p = [1 \quad 0 \quad 0 \quad 0] \begin{bmatrix} \theta_p \\ \dot{\theta}_p \\ \theta_r \\ \dot{\theta}_r \end{bmatrix}$$

입력 전압 V 에 일정한 Step 입력(motor torque)을 주었을 때 초기위치가 20도일 때의 pendulum과 rotor의 움직임을 simulink로 구현하여 도시하면 다음과 같다.



입력을 Impulse 입력으로 주어졌을 때 다음과 같다.



Upright position에서의 linearization을 하기 위하여 먼저 upright position의 equilibrium state를 먼저 찾아야 한다.

Pendulum의 외부 입력이 없을 때의 Lagrange's equation은 아래와 같으며

$$\ddot{\theta} + \frac{mgl}{J} \sin \theta = 0$$

$$\ddot{\theta}_r = 0$$

Equilibrium point를 찾기 위해서 다음의 식을 풀면,

$$f(\theta) = \frac{d\theta}{dt} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \dot{\theta} \\ -\frac{mgl}{J} \sin \theta \end{bmatrix} = 0$$

$$\begin{cases} \dot{\theta} = 0 \\ -\frac{mgl}{J} \sin \theta = 0 \end{cases}$$

$$\therefore \theta = 0, \pi, 2\pi, \dots$$

과 같다.

Upright position의 equilibrium을 찾아야 하므로

$$\theta = \pi$$

이고, 이 때의 값을 이용하여 Taylor series expansion을 이용하여 linearization을 하면

$$f(\theta) = f(\pi) + \left. \frac{df}{d\theta} \right|_{\theta=\pi} (\theta - \pi)$$

$$f(\theta) = 0 - \frac{mgl}{J} \cos \pi (\theta - \pi) = \frac{mgl}{J} \sin \theta$$

$$\therefore \sin \theta = \pi - \theta$$

가 된다.

여기서

$$\theta - \pi = \theta_p$$

라고 하면, linearization된 dynamics equation은 다음과 같다.

$$\ddot{\theta}_p - \frac{mgl}{J} \theta_p = -\frac{k}{JR} (V - k_b(\dot{\theta}_r - \dot{\theta}_p))$$

$$\ddot{\theta}_r = \frac{k}{J_r R} (V - k_b(\dot{\theta}_r - \dot{\theta}_p))$$

여기서 Pendulum과 Motor 사이의 발생하는 토크를 F라고 하고 k_f 를 Pendulum과 Motor 사이의 마찰계수라고 하면,

$$F = k_f \dot{\theta}_r$$

이고, 이를 dynamics equation으로 다시 나타내면

$$\ddot{\theta}_p - \frac{mgl}{J} \theta_p = -\frac{k}{JR} (V - k_b(\dot{\theta}_r - \dot{\theta}_p)) + \frac{k_f}{J} \dot{\theta}_r$$

$$\ddot{\theta}_r = \frac{k}{J_r R} (V - k_b(\dot{\theta}_r - \dot{\theta}_p)) - \frac{k_f}{J_r} \dot{\theta}_r$$

따라서 이를 state-space equation으로 나타내면

$$\begin{bmatrix} \dot{\theta}_p \\ \ddot{\theta}_p \\ \dot{\theta}_r \\ \ddot{\theta}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{mgl}{J} & -\frac{kk_b}{JR} & 0 & \frac{kk_b}{JR} + \frac{k_f}{J} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{kk_b}{J_r R} & 0 & -\frac{kk_b}{J_r R} - \frac{k_f}{J_r} \end{bmatrix} \begin{bmatrix} \theta_p \\ \dot{\theta}_p \\ \theta_r \\ \dot{\theta}_r \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{k}{JR} \\ 0 \\ \frac{k}{J_r R} \end{bmatrix} V$$

$$\theta_p = [1 \quad 0 \quad 0 \quad 0] \begin{bmatrix} \theta_p \\ \dot{\theta}_p \\ \theta_r \\ \dot{\theta}_r \end{bmatrix}$$

과 같이 나타낼 수 있다.

여기서 $\dot{\theta}_r$ 은 다른 state에 영향을 주지 않기 때문에 고려하지 않으면,

$$\begin{bmatrix} \dot{\theta}_p \\ \ddot{\theta}_p \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \frac{mgl}{J} & -\frac{kk_b}{JR} & \frac{kk_b}{JR} + \frac{k_f}{J} \\ 0 & \frac{kk_b}{J_r R} & -\frac{kk_b}{J_r R} - \frac{k_f}{J_r} \end{bmatrix} \begin{bmatrix} \theta_p \\ \dot{\theta}_p \\ \theta_r \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{k}{JR} \\ \frac{k}{J_r R} \end{bmatrix} V$$

$$\theta_p = [1 \quad 0 \quad 0] \begin{bmatrix} \theta_p \\ \dot{\theta}_p \\ \dot{\theta}_r \end{bmatrix}$$

가 된다.

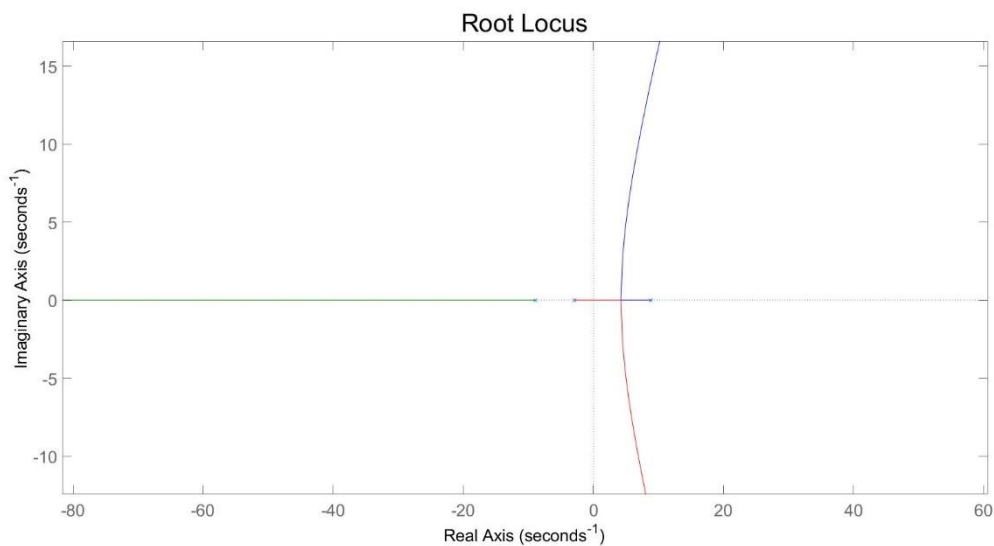
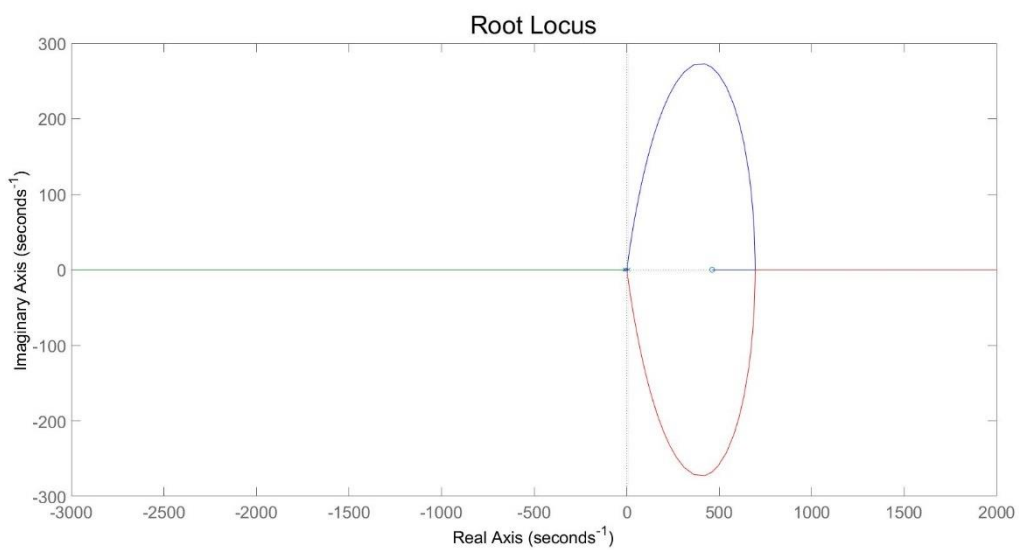
PID 제어를 하기 전 해당 system의 stability를 확인하기 위해 system의 pole, zero를 확인해보면

$$G(s) = C * (sI - A)^{-1} * B$$

$$G(s) = \frac{-0.049529(s - 462.4)}{(s - 8.861)(s + 8.919)(s + 2.981)}$$

이므로, system의 pole은 8.861, -8.919, -2.981로 한 개의 pole이 RHP에 있으므로 기본적으로 불안정한 시스템이다.

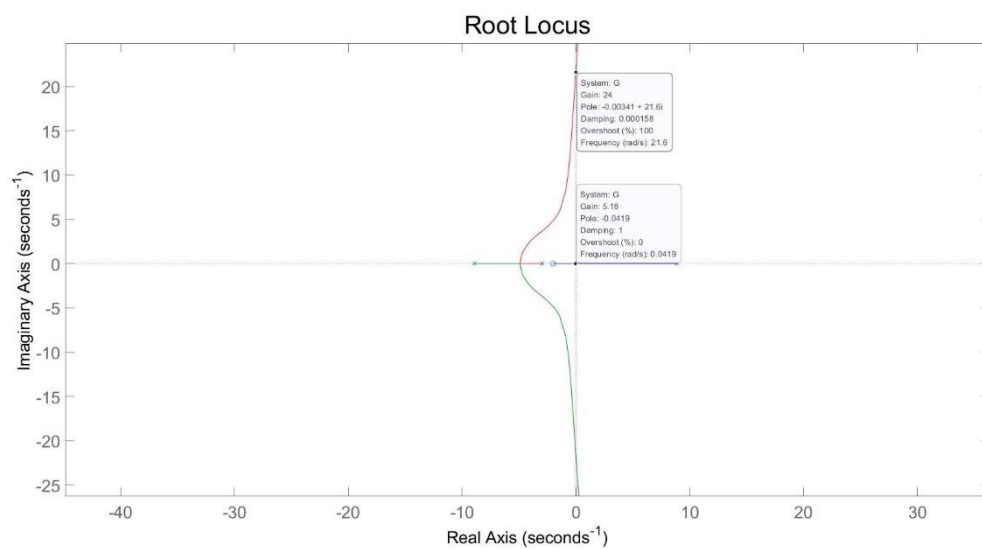
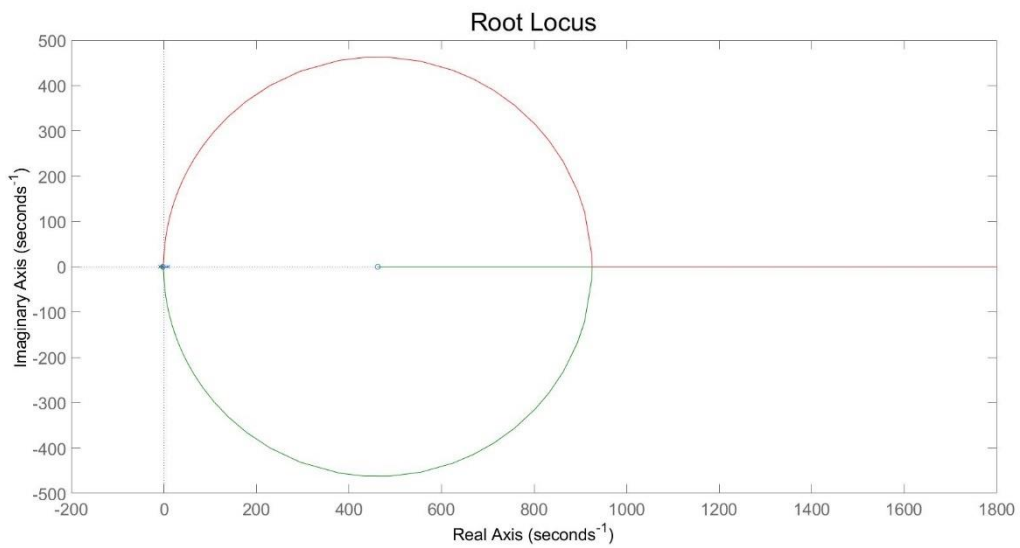
G(s)의 root locus를 그려보면 아래와 같고, 그래프에서 볼 수 있듯이, 게인 값을 변화시켜도 불안정한 시스템임을 알 수 있다.



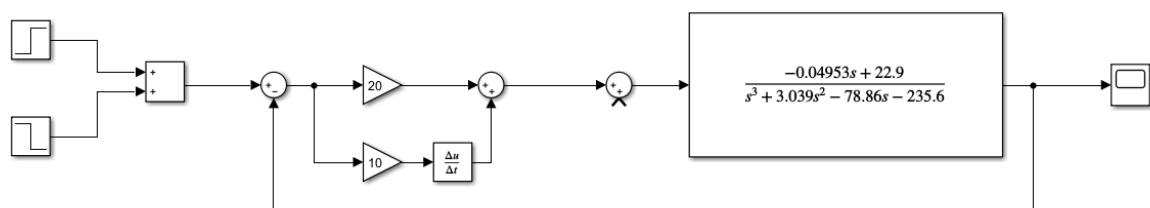
따라서 RHP에 있는 pole을 LHP로 끌고 오기 위하여 -2.981과 0점 사이에 zero를 추가하고자 한다. Zero -2를 추가하면,

$$G(s) = \frac{-0.049529(s - 462.4)(s + 2)}{(s - 8.861)(s + 8.919)(s + 2.981)}$$

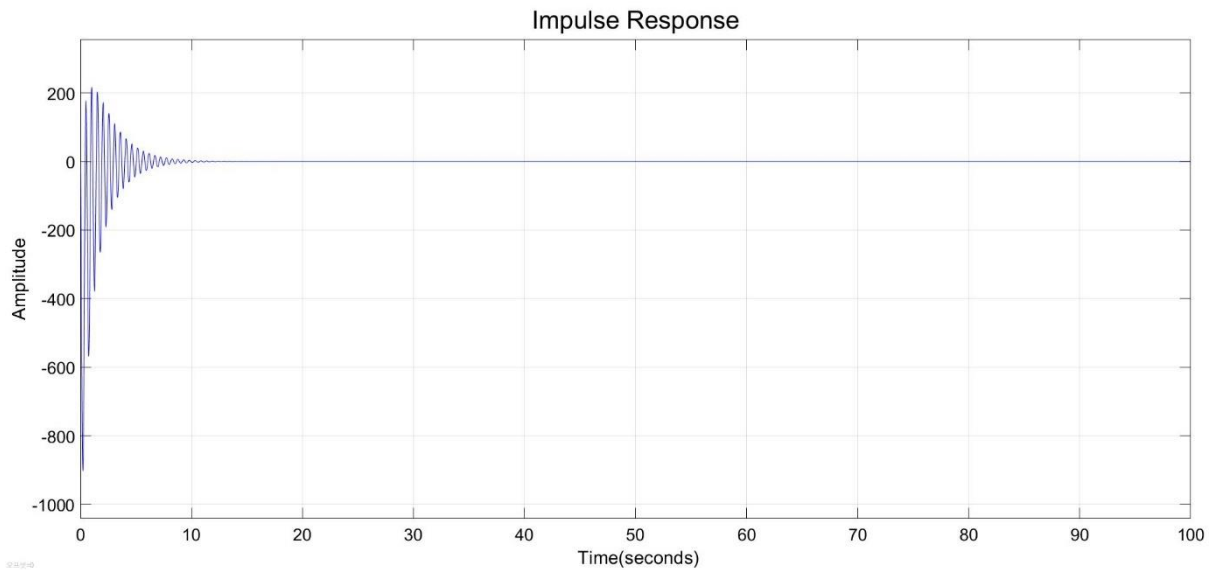
와 같이 되고, root locus를 그려보면 다음과 같고, Gain이 5.18 이상이고, 24 이하일 때 system의 모든 pole이 LHP에 있는 것을 알 수 있다.



따라서 PD 제어기로 $10(s+2)$ 로 설정을 하면 그림과 같이 나타낼 수 있다.



임펄스 응답을 확인해보면 다음과 같이 수렴함을 알 수 있다.



State compensator control을 진행하기 위해 Controllability와 Observability를 확인해보면

$$C = \begin{bmatrix} 0 & -0.0495 & 23.0523 \\ -0.0495 & 23.0523 & -73.9641 \\ 9.0760 & -27.5828 & 95.2586 \end{bmatrix}$$

$$O = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 77.6046 & -0.0027 & 2.5399 \end{bmatrix}$$

이고 각각의 rank는 3이므로, 해당 system은 controllable하고 observable하다는 것을 알 수 있다.

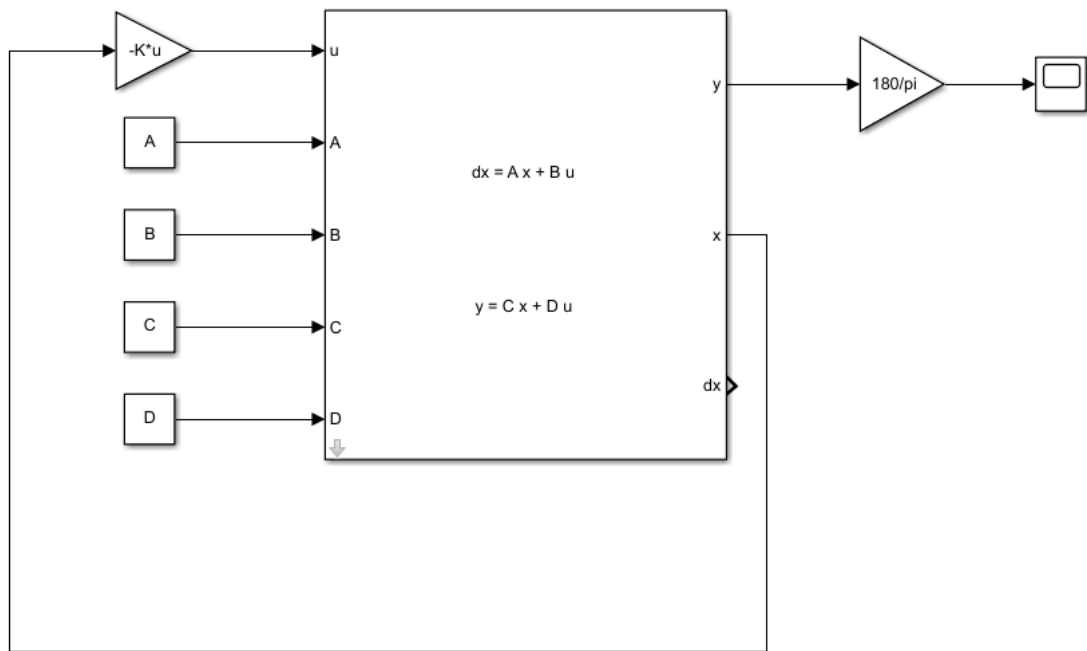
설계하고자 하는 system을 Damping ratio = 0.7, Natural frequency = 1로 진행하고, 나머지 pole의 위치를 -100으로 하면, 원하는 pole은 아래와 같다.

$$p_c = [-100 \quad -0.7084 \quad -0.6917]$$

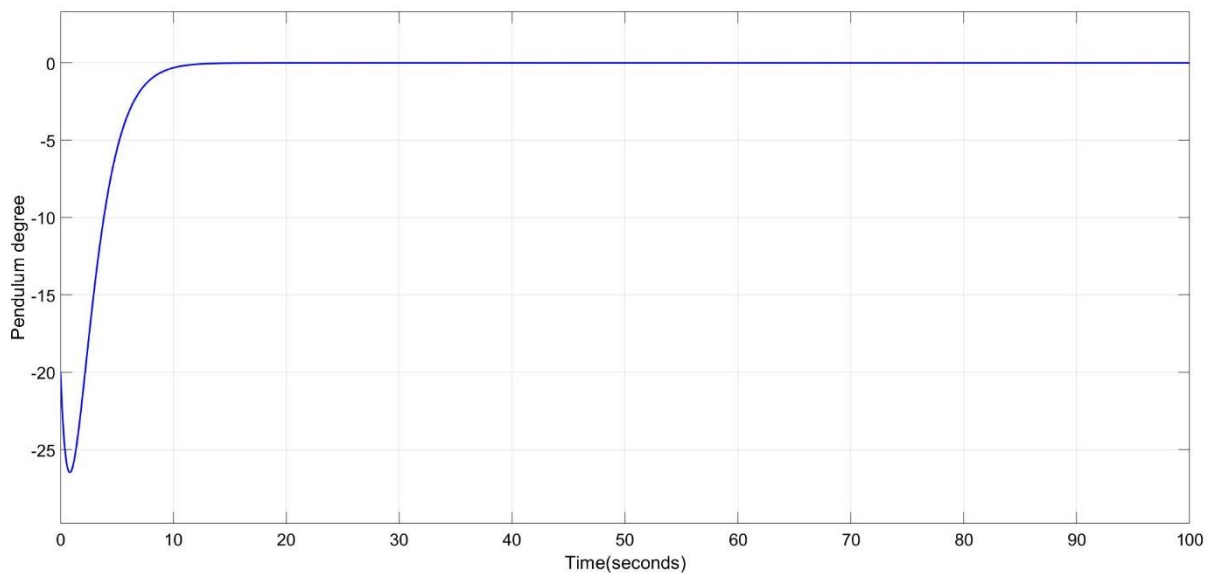
이 때 K 값을 구하면

$$K = [347.4657 \quad 10.3299 \quad 10.8938]$$

이며, state feedback control system은 다음과 같으며,



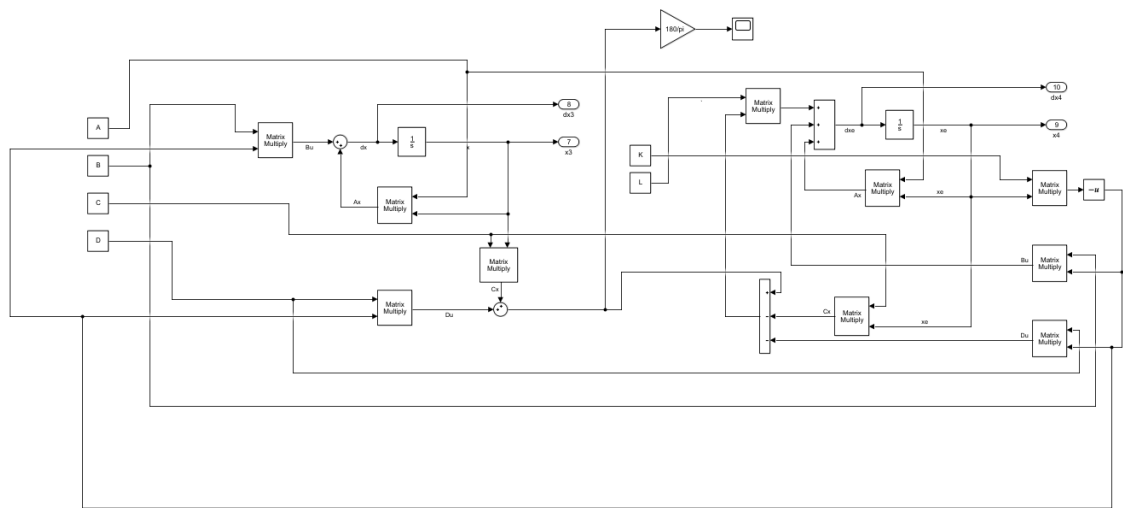
초기 각도가 -20도일 때 Pendulum의 각도를 나타내면 다음과 같다.



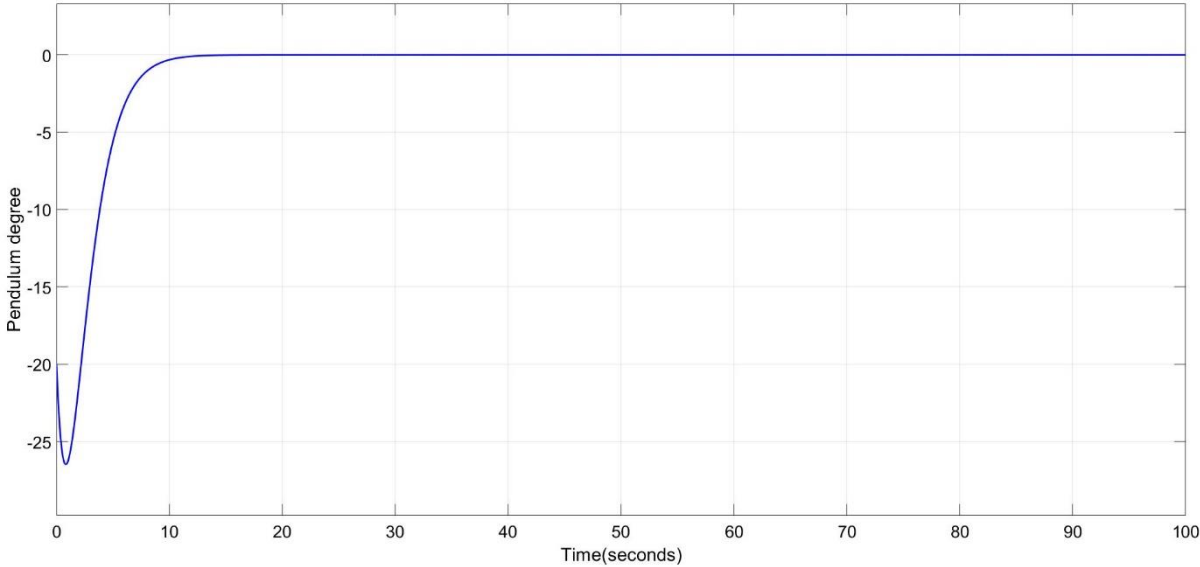
Estimator를 설계하기 위해, estimator의 pole을 system pole의 3배로 선정하였다. 따라서 estimator의 L을 구하면

$$L = [301.1612 \quad 428.1090 \quad 251.2236]^T$$

이고, estimator를 포함한 system을 simulink로 나타내면 다음과 같다.



동일하게 초기 각도가 -20일 때 pendulum degree를 보면 다음과 같다.



Non-linear system과 Linear system의 성능을 비교하고자 한다.

Non-linear system은 아래와 같이 나타낼 수 있으며,

$$\ddot{\theta} + \frac{mgl}{J} \sin \theta = -\frac{k}{JR} (V - k_b(\dot{\theta}_r - \dot{\theta})) + \frac{k_f}{J} \dot{\theta}_r$$

$$\ddot{\theta}_r = \frac{k}{J_r R} (V - k_b(\dot{\theta}_r - \dot{\theta})) - \frac{k_f}{J_r} \dot{\theta}_r$$

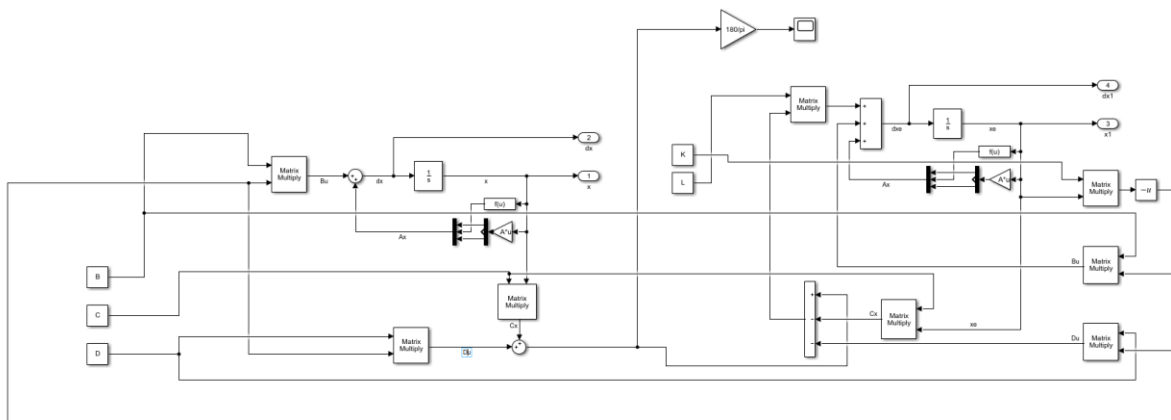
$$\theta - \pi = \theta_p$$

이므로,

$$\ddot{\theta}_p - \frac{mgl}{J} \sin \theta_p = -\frac{k}{JR} (V - k_b(\dot{\theta}_r - \dot{\theta}_p)) + \frac{k_f}{J} \dot{\theta}_r$$

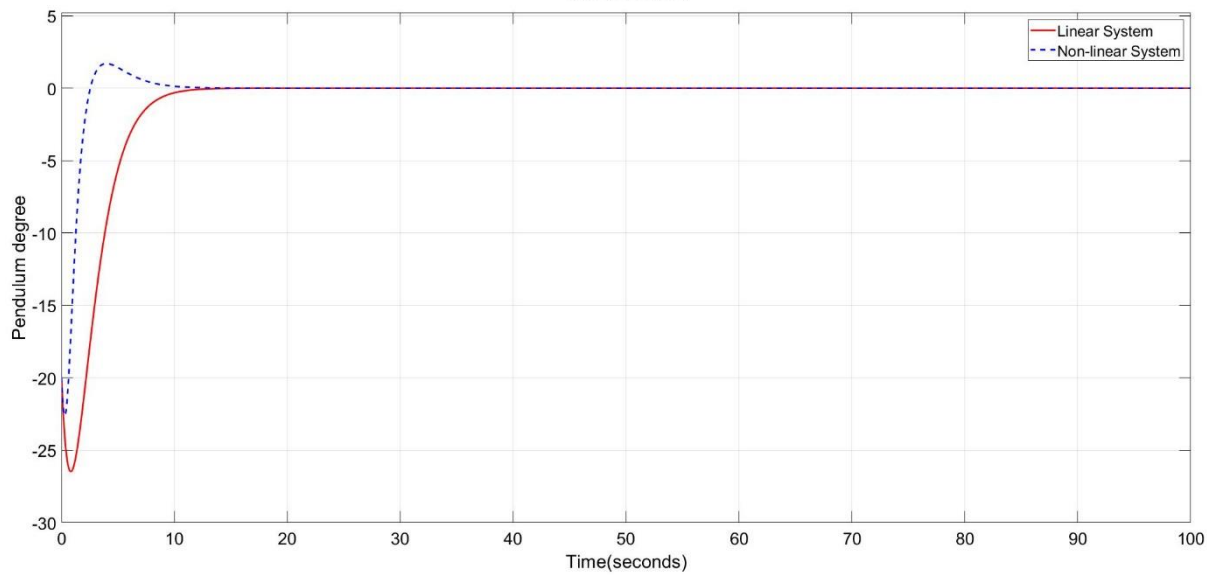
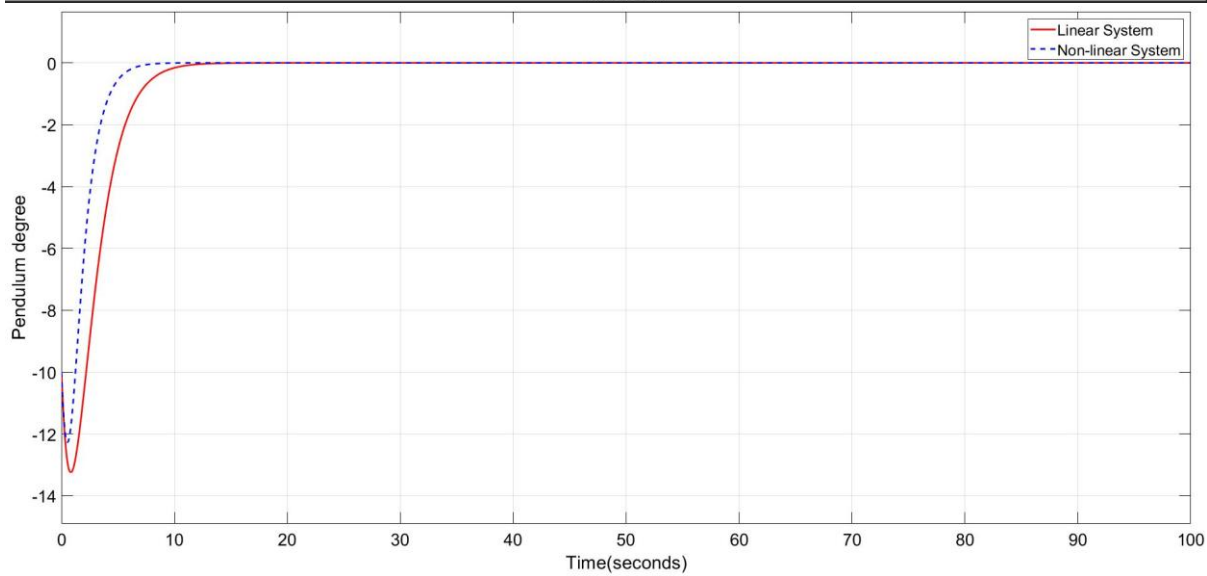
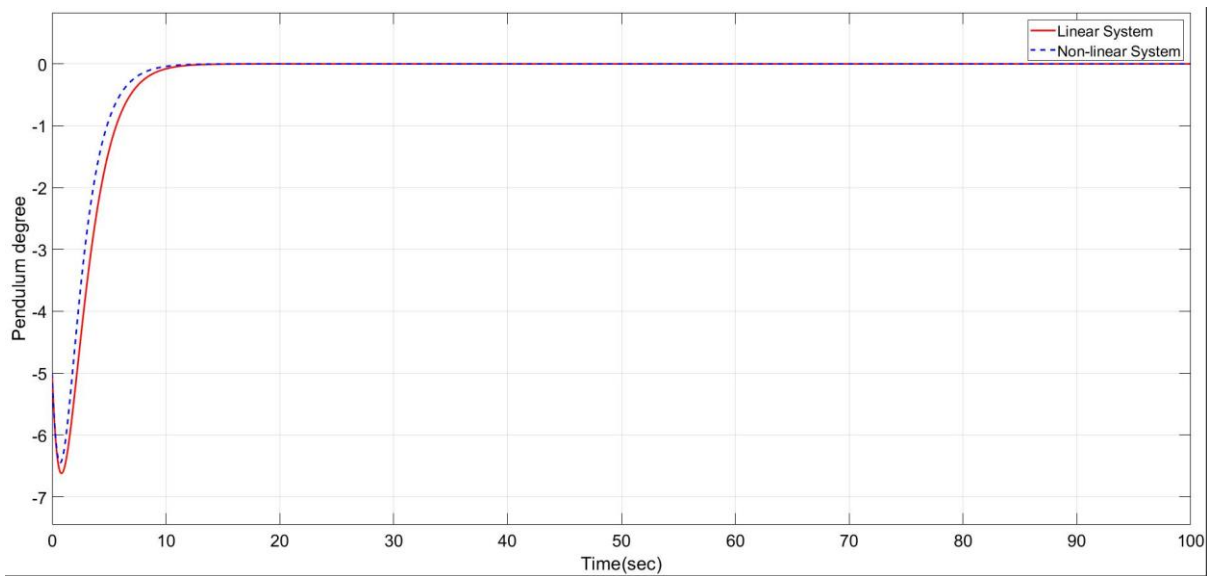
$$\ddot{\theta}_r = \frac{k}{J_r R} (V - k_b(\dot{\theta}_r - \dot{\theta}_p)) - \frac{k_f}{J_r} \dot{\theta}_r$$

이를 Simulink로 동일하게 나타내면 다음과 같다.



여기서 non-linear term인 $\sin \theta_p$ 를 표현하기 위해 A matrix와 x의 연산의 결과를 첫 번째 항과 세 번째 항만 사용하고, 두 번째 항은 직접 계산하여 사용하였다.

Non-linear system과 Linear system을 초기위치가 각각 -5, -10, -20일 때를 비교하면, 초기 각도가 equilibrium에서 멀어질수록 system의 동작에 차이가 점점 커지는 것을 알 수 있다.



3. Swing Up & Stabilizing Control

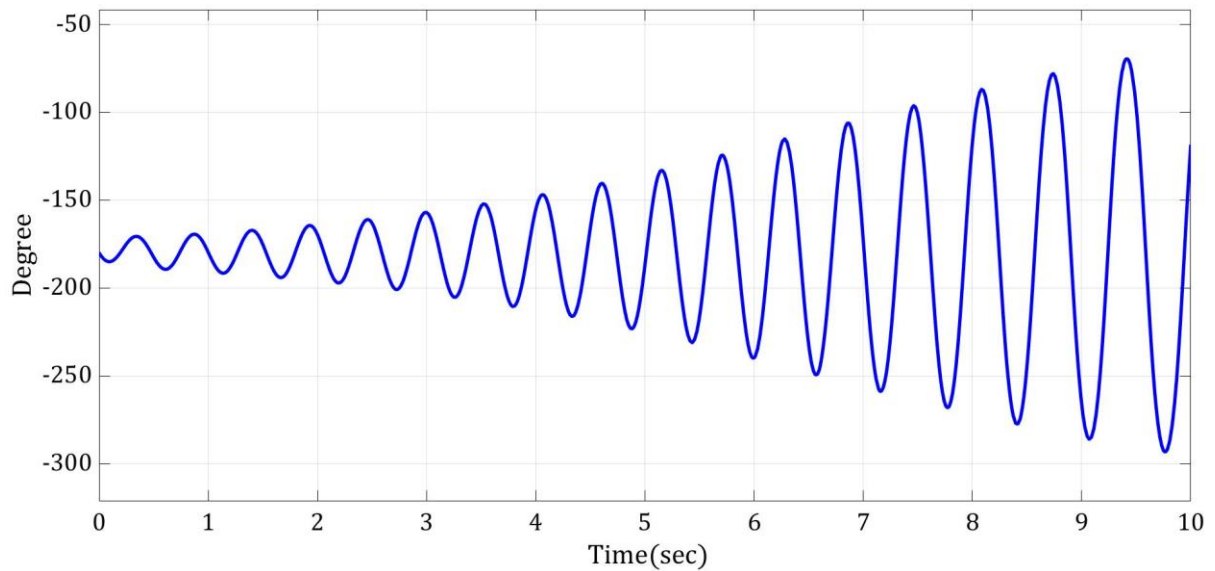
Designing swing up control

Swing up control을 하기 위해 아래와 같은 식을 이용하고자 하였으나, 초기 상태가 $\dot{\theta}_u$ 이 0인 경우, Swing up을 시작하지 않는 문제가 있었고, Swing up을 해서 Upright position에 도달하는데 시간이 오래 걸렸음.

$$y = -k\dot{\theta}_u(E - E_{ref})u$$

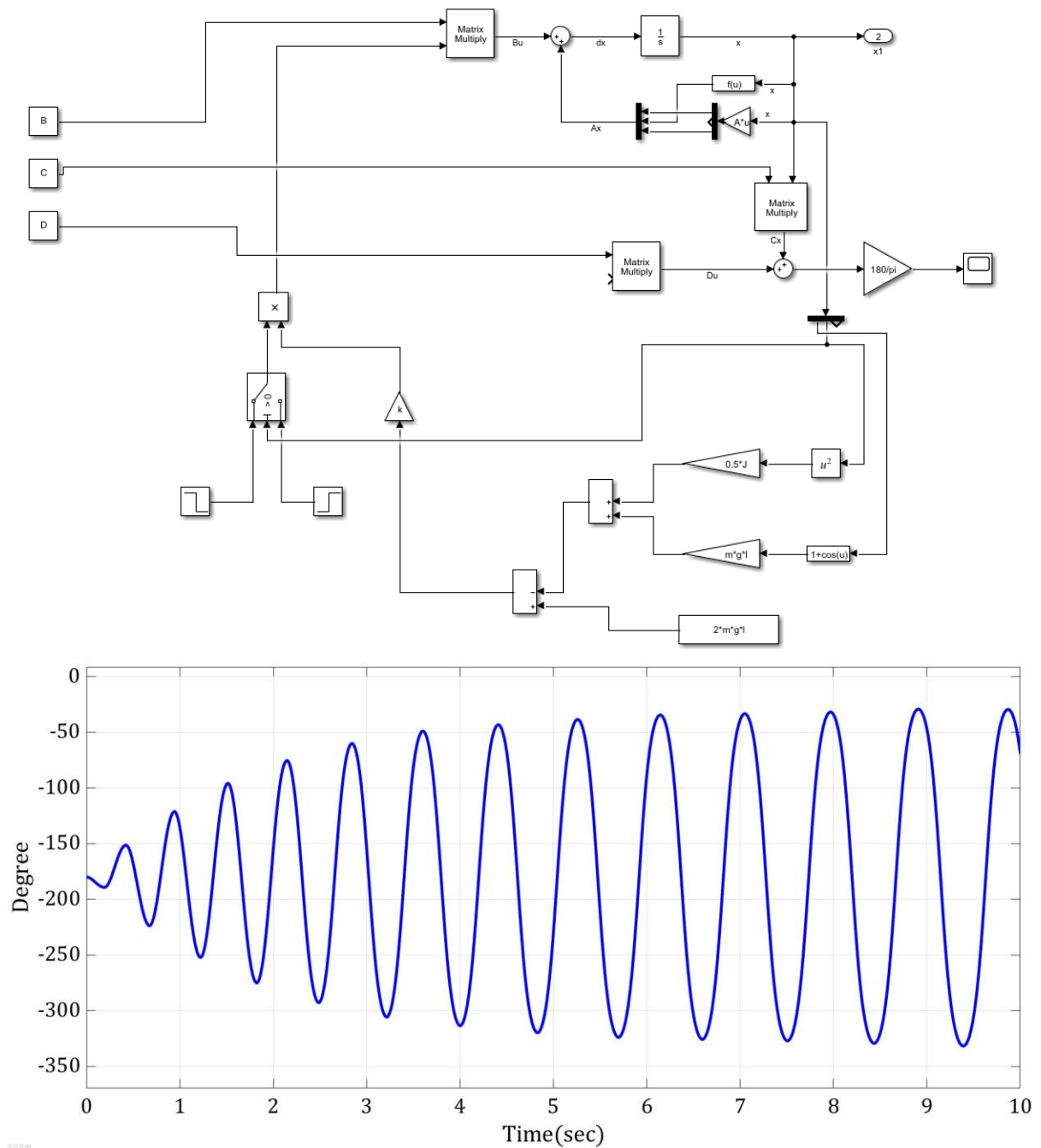
$$E = \frac{1}{2}J\dot{\theta}_u^2 + mgl(1 + \cos\theta_u)$$

$$E_{ref} = 2mgl$$



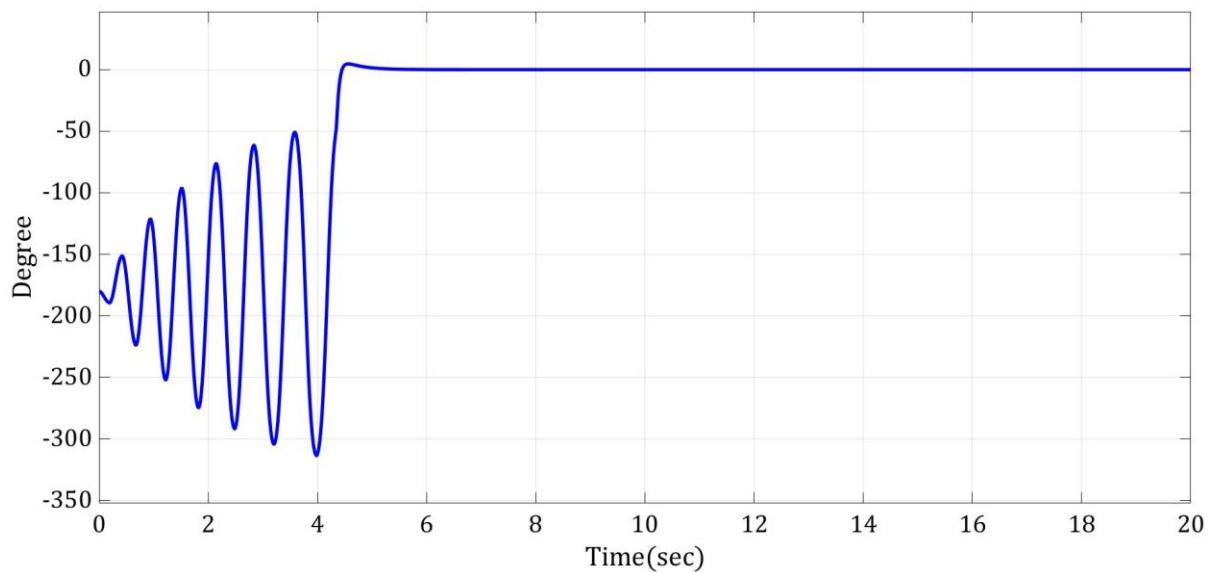
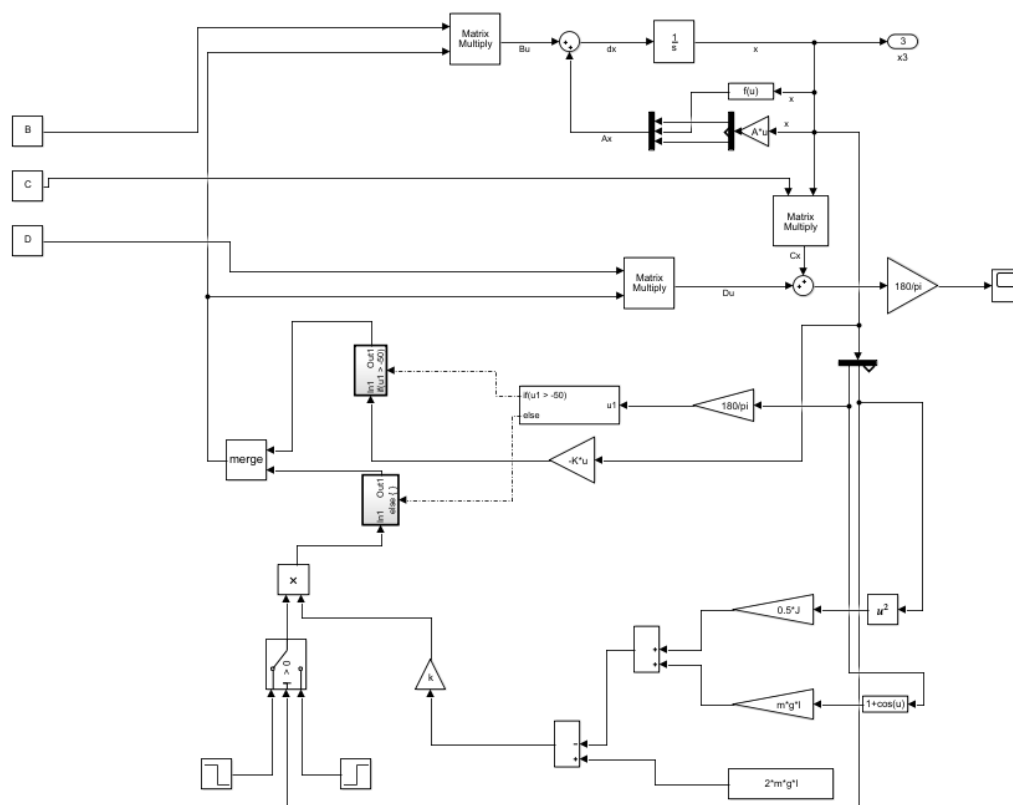
동일한 아이디어를 활용해서 $\dot{\theta}_u$ 의 부호만 사용하였음. 따라서 0일 때는 +방향으로 Swing을 하게 함으로 $\dot{\theta}_u$ 이 0이어도 Swing을 하고, Magnitude는 항상 1이므로 Swing up의 시간도 단축되었음.

$$y = \begin{cases} -k(E - E_{ref})u & \dot{\theta}_u \geq 0 \\ k(E - E_{ref})u & \dot{\theta}_u < 0 \end{cases}$$



Incorporate controller

초기 각도 -180도에서 Swing up을 하다가 -50도에서 Stabilizing control로 스위칭 되도록 Simulink를 구현하였으며, 시뮬레이션 결과 -50도에서 Stabilizing control로 스위칭 되어 0도에서 수렴함을 알 수 있다.



Discretize the controller

우선 Continuous system을 MATLAB을 이용하여 Zero order holder 방식으로 Discrete System을 구하였음

```
[DA, DB, DC, DD] = c2dm(A,B,C,D,Ts,'zoh');
```

Discretize system에서의 Desired closed loop pole과 Desired observer pole을 구하기 위해서 Continuous system에서의 Desired closed loop pole과 Desired observer pole인 $[-100, -15, -10]$, $[-300, -45, -30]$ 을 아래와 같이 MATLAB으로 구하였음.

$$Z = e^{T_s * s} \quad (T_s = 0.001 \text{ sec}, 1\text{KHz})$$

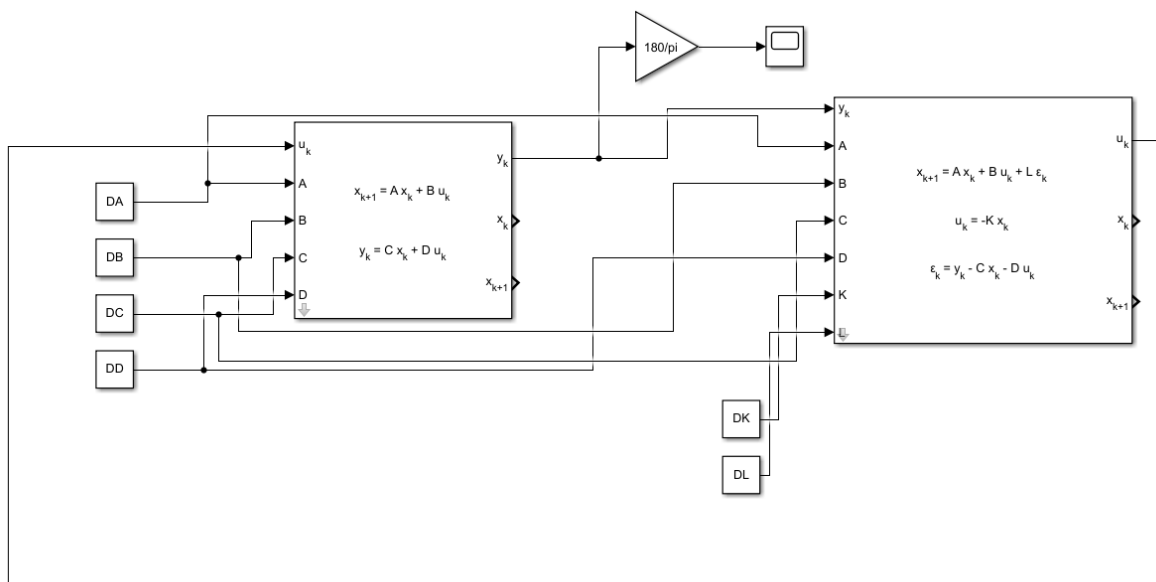
$$D_CL_Pole = \exp(T_s * CL_Pole);$$

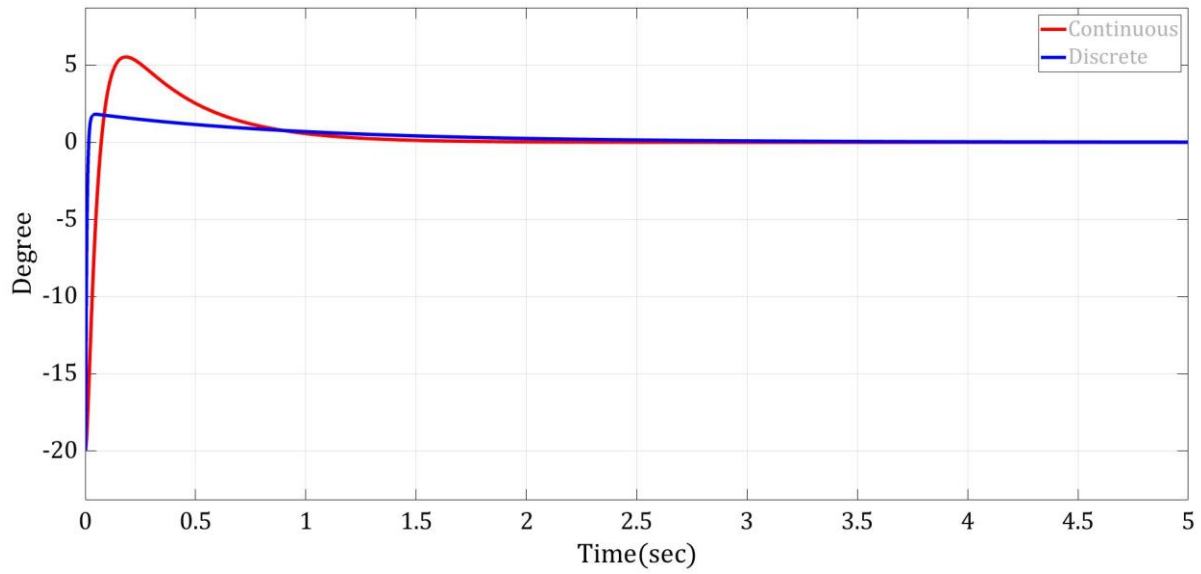
$$D_Ob_Pole = \exp(T_s * Ob_Pole);$$

원하는 system의 pole을 만들기 위해 K값과 L값을 다음과 같이 구하고, Simulink를 구현하였음.

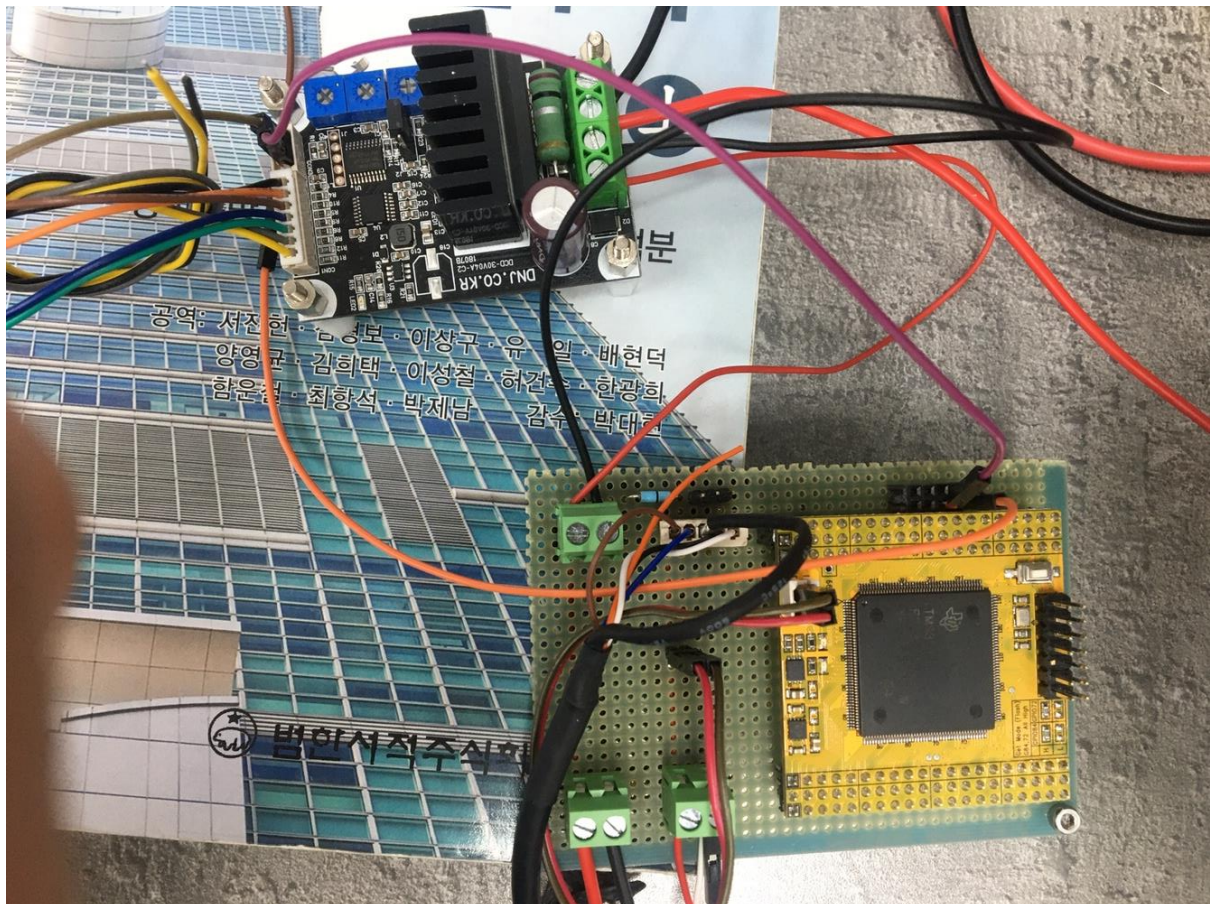
$$DK = \text{place}(DA, DB, D_CL_Pole)$$

$$DL = \text{place}(DA', DC', D_Ob_Pole)'$$





Controller는 TMS320F28377D를 사용하였으며, 모터 드라이버는 기존에 수령한 모터 드라이버의 출력이 나오지 않아 다른 모터 드라이버(DCD-30V04A-C2)를 사용함.



실제로 구현한 결과 기존에 선정했던 Pole을 사용했을 때 Upright position에 수렴하지 않아, Desired feedback pole과 Desired observer pole을 수정하며 적절한 게인을 선정하였다.

