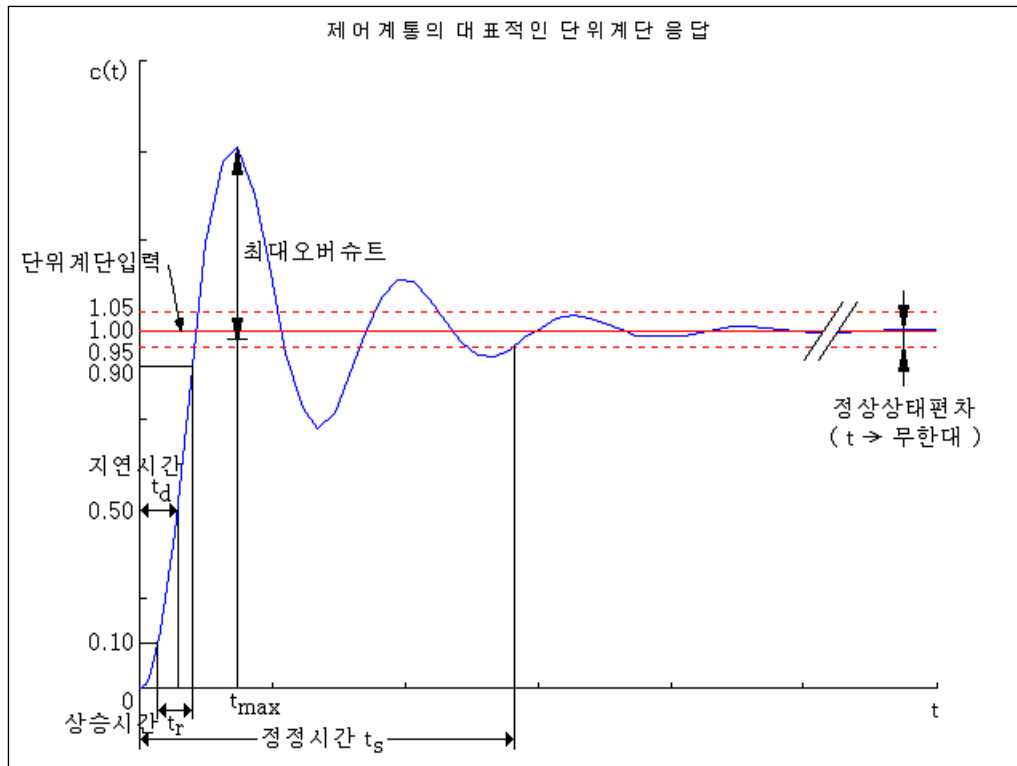


## 1. 용어의 정의



### 1) 최대 오버슈트(Maximum Overshoot)

최대 오버슈트 =  $C_{\max}$  (단위 계단 응답의 최대치) -  $C_{ss}$  (단위 계단 응답의 정상상태치)

### 2) 지연 시간(Delay Time)

지연시간  $t_d$ 는 계단응답이 최종치의 50%에 도달하는데 필요한 시간으로 정의한다.

### 3) 상승시간(Rise Time)

상승시간  $t_r$ 는 계단응답이 최종치의 50%에 도달하는데 필요한 시간으로 정의 한다.

### 4) 보정시간(Settling Time)

정정시간  $t_s$ 는 계단 응답이 감소하여 그 응답 최종치의 특정 백분율 이내에 들어가는데 필요한 시간으로 정의 한다. 보통 사용되는 양은 5%이다.

### 5) 단위 계단 응답(unit-step response)

입력이 단위 계단함수 일 때 제어계통의 응답을 말한다.

### 6) 전달 함수

전달함수는  $G(s) = \frac{\text{출력}(Y(s))}{\text{입력}(U(s))}$  이다.

## 2. 안정도 판별법

설계목적으로 선형제어시스템의 안정도 판별을 하는 경우에는 미지의 파라미터가 특성방정식에 내장되므로 근을 구하는 프로그램을 사용하는 것이 불가능하다. 더욱이 절대안정도만 관계한다면  $s$ -평면의  $j\omega$  축에 관계되는 특정방정식에 관한 정보만이 필요하다. 완전히 구할 필요는 없다. 근을 직접 구하지 않고 선형제어시스템의 안정도를 결정하는 방법중 잘 알려져 있는 것들은 주로 다음과 같다

안정도 판별법	특징
Routh-Hurwitz 법	특정방정식이 상수계수를 갖는 선형 시불변시스템의 절대 안정도에 관한 정보를 제공하는 대수적 방법이다. 이 판별법은 특성방정식의 근 중 그 어느 하나가 $s$ -평면의 우반면에 놓여 있는지를 검사한다. $s$ -평면의 $j\omega$ 축 위나 $s$ -평면의 우반면에 놓여 있는 특정근의 수도 또한 제시할 수 있다.
Nyquist 선도	루프전달함수의 Nyquist 특성을 관찰함으로써 $s$ -평면 우반부에 있는 페루프전달함수의 극점과 영점의 수의 차이에 관한 정보를 주는 반도표적 방법이다. 페루프전달함수의 극점은 그 특성방정식의 근이다. Nyquist 도표에 대한 데이터는 ACSP 소프트웨어에서의 FREQRP 또는 MATLAB의 LTIVIEW를 사용하여 얻을 수 있다.
근궤적 선도	이 선도는 어떤 계통 파라미터가 변할 때의 특성방정식의 궤적이다. 이 근궤적은 변화하는 파라미터에 관계되는 안정도의 명백한 그림을 제공한다. 근궤적 구성에 관한 법칙에 의하여 실제 특성방정식의 근을 풀지 않고 대략 근궤적을 그릴 수 있다. 근궤적은 ACSP 소프트웨어에서의 ROOTLOC1 또는 MATLAB의 LTIVIEW로 정확히 그릴 수 있다.
Bode 선도	이 선도는 루프전달함수 $G(j\omega)H(j\omega)$ 의 크기를 dB로, $G(j\omega)H(j\omega)$ 의 위상을 각도로 모든 주파수 $\omega$ 에 대하여 그린 도표이다. 이 Bode 선도는 ACSP소프트웨어에서의 FREQRP 또는 MATLAB의 LTIVIEW로 그릴 수 있다.
PZMAP	영점과 극점을 통해 안정도를 확인해 주는 선도이다. 만약 모든 영점과 극점이 허수축을 기준으로 하여 좌반부에 있으면 그 시스템은 안정하다고 말할 수가 있다.

## 1. 도입

PID 제어기는 현장에서 가장 많이 적용되는 궤환제어기이다. 단순함에 비해 뛰어난 성능 때문이다. 따라서 본 도입진자 시스템에 PID 제어기를 자연스럽게 도입해 보게 되었다. 단, 여기서는 간단하게 SISO 시스템으로 생각하여 시스템의 해석과 파라미터의 결정이 간단하도록 하였다.

## 2. 전달함수의 유도

본 도입진자 시스템의 운동방정식은 다음과 같이 주어져 있다.

$$J_t \ddot{\Phi} - \frac{1}{2} m_b r l \ddot{\Theta} + C_\Phi \dot{\Phi} = k_m u \quad (\text{식 1-1})$$

$$- \frac{1}{2} m_b r l \ddot{\Phi} + (J_b + J_{Ex}) \ddot{\Theta} + C_\Theta \dot{\Theta} - m_b g \frac{l}{2} \Theta = 0 \quad (\text{식 1-2})$$

변수들을

$$J_t = J_\Phi$$

$$N = -\frac{1}{2} m_b r L$$

$$J_\Theta = J_b + J_{Ex}$$

$$G = m_b g \frac{L}{2}$$

로 치환하면,

$$J_{tot} \ddot{\Phi} + N \ddot{\Theta} + C \dot{\Phi} = k_m u \quad (\text{식 1-1'})$$

$$N \ddot{\Phi} + J_\Theta \ddot{\Theta} - G \Theta = 0 \quad (\text{식 1-2'})$$

여기서 전달함수를 구하는 방법에는 고전적 방법(Classical Control ; Laplace 변환에 의한 방법)과 현대적 방법(Modern Control ; Matrix 통해 상태 방정식을 이용하는 방법)으로 나누어 진다. 각각에 대해 알아보면 다음과 같다.

### 1) 고전적 방법(Classical Control)

위의 운동방정식에 대하여 Laplace 변환을 한다. 우선 첫 번째 운동방정식을 Laplace 변환하면

$$J_t \bar{\Phi}(s) s^2 + N \bar{\Theta}(s) s^2 + C \bar{\Phi}(s) s = k_m \bar{u}(s) \quad (\text{식 2-1})$$

이 주어지고, 두 번째 운동방정식을 Laplace 변환하면

$$N \bar{\Phi}(s) s^2 + J_\Theta \bar{\Theta}(s) s^2 - G \bar{\Theta}(s) = 0 \quad (\text{식 2-2})$$

이 주어진다.

여기서 첫 번째 라플라스 변환식을  $\bar{\Phi}(s)$ 에 관하여 정리한 후 두 번째 라플라스 변환식에 대입함으로써  $\bar{\Phi}(s)$ 를 소거하면

$$\frac{\bar{\Theta}(s)}{\bar{u}(s)} = \frac{K_m N s^2}{(N^2 - J_\Theta J_\Phi) s^4 - C J_\Theta s^3 + G J_\Phi s^2 + C G s}$$

를 얻을 수 있다. 이것이 시스템의 전달함수이다.

### 2) 현대적 방법(Modern Control)

최초의 운동방정식을 행렬로 표현하면 다음을 얻을 수 있다.

$$\begin{pmatrix} J_\Phi & N \\ N & J_\Theta \end{pmatrix} \begin{pmatrix} \ddot{\Phi} \\ \ddot{\Theta} \end{pmatrix} + \begin{pmatrix} C_\Phi & 0 \\ 0 & C_\Theta \end{pmatrix} \begin{pmatrix} \dot{\Phi} \\ \dot{\Theta} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & -\frac{m}{2} g l \end{pmatrix} \begin{pmatrix} \Phi \\ \Theta \end{pmatrix} = \begin{bmatrix} K_m \\ 0 \end{bmatrix} u$$

상태 방정식으로 변환시킨다.

$$Z = \begin{bmatrix} \Phi \\ \Theta \end{bmatrix}$$

$$M = \begin{pmatrix} J_\phi & N \\ N & J_\theta \end{pmatrix}$$

$$C = \begin{pmatrix} C_\phi & 0 \\ 0 & C_\theta \end{pmatrix}$$

$$K = \begin{pmatrix} 0 & 0 \\ 0 & -\frac{m}{2}gL \end{pmatrix}$$

$$Bs = \begin{bmatrix} K_m \\ 0 \end{bmatrix}$$

$$MZ + CZ + KZ = Bs u$$

$$x = \begin{bmatrix} Z \\ \dot{Z} \end{bmatrix} \quad \dot{x} = \begin{bmatrix} \dot{Z} \\ \ddot{Z} \end{bmatrix}$$

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \phi \\ \theta \\ \dot{\phi} \\ \dot{\theta} \end{bmatrix}$$

$$\dot{x} = Ax + Bu$$

$$A = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ M^{-1}Bs \end{bmatrix}$$

여기 상태 방정식에서 systf=tf(sys)의 명령어를 사용하여 구하면 된다.

상태공간에서부터 전달함수를 구하는 MATLAB 프로그램

```
clear
Jphi = 0.0026;
Jthe = 0.0105;
N = -0.0018;
G=1.472*10^-1;
Cphi=3.15/1000;
km = 18.9/1000;
Cthe=0.0;
M = [ Jphi N; N Jthe]
C = [ Cphi 0; 0 Cthe]
K = [ 0 0; 0 -G]
Bs = [km; 0]
A=[ zeros(2,2) eye(2); -inv(M)*K -inv(M)*C]
B = [ 0; 0; inv(M)*Bs ]
C = [ 0 1 0 0]
D = [ 0 ]
sys=ss(A,B,C,D);
systf=tf(sys)
```

각각에 대해 물성치를 대입한 결과는 다음과 같다.

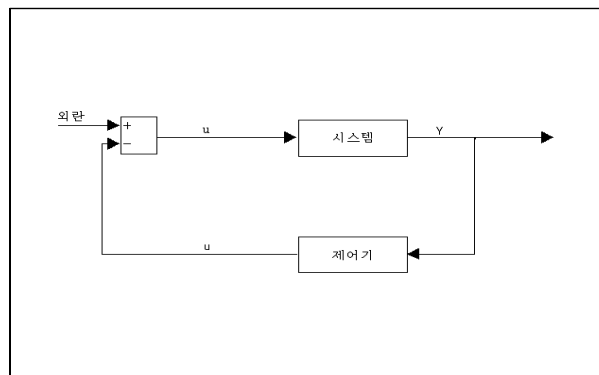
## 결 과

Transfer function:

$$\frac{1.414 s^2}{s^4 + 1.375 s^3 - 15.91 s^2 - 19.27 s}$$

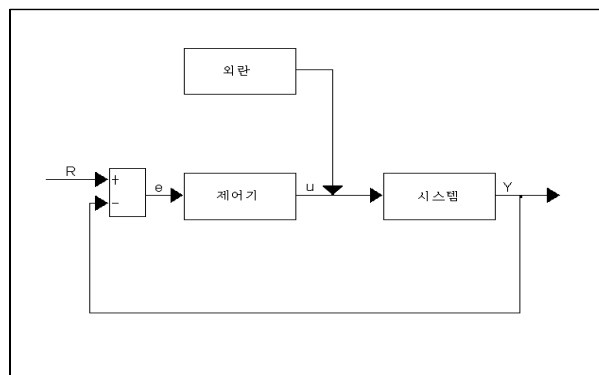
이제 PID 제어기가 들어간 페루프 제어시스템에 관하여 생각해 본다. PID 제어기라고 하는것은  $K_p$ ,  $K_i$ ,  $K_d$  라는 비례성분( $K_p$ ), 적분성분( $K_i$ ), 미분성분( $K_d$ )를 이용하여 제어기를 설계하는 것을 말한다. 이번 Rotating Inverted Pendulum에서는 다음과 같은 feedback system(직렬 보상)을 사용한다.

(1) 일반적인 경우



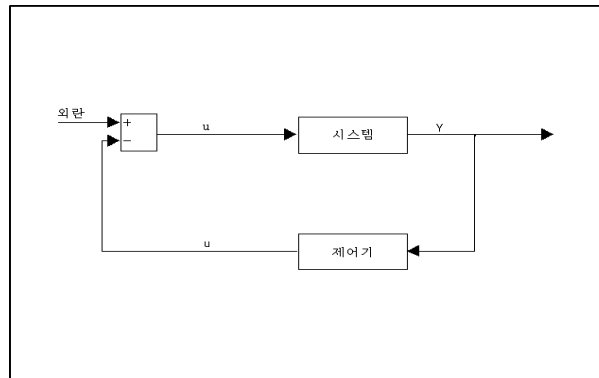
일반적인 경우

(2) 외란이 있는 경우



외란이 있는 경우

(3) 기준 입력이 0이고 외란이 있을 경우



기준입력인 0이고 외란이 있는 경우

### 3. PID제어기 설계 과정

1) 실제 PID제어기 설계의 대략적인 개요

(1) PI부분이 PD부분과 종속 접속된 PID제어기를 생각한다. PID 제어기 전달함수는 다음과 같이 쓸 수 있다.

$$G_c(s) = K_p + K_d s + \frac{K_i}{s}$$

(2) PI 부분만이 효과가 있다고 생각하고 응답의 상승시간에 대한 요구를 만족하도록  $K_p$ ,  $K_i$ 의 값을 택한다. 계통의 정상상태 편차는 일차(One order)로 개선된다. 현 단계에서는 최대 오버슈트가 크더라도 염두에 두지 않는다.

(3) 이제 최대 오버슈트를 감소시키기 위해 PD 부분을 고려한다. 제동 요구에 맞게  $K_d$ 의 값을 선택한다.

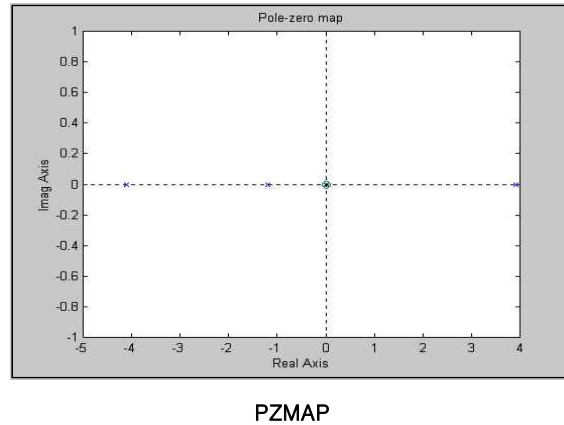
(4) 이상의 과정을 반복하여  $K_p$ ,  $K_i$ ,  $K_d$ 값을 조정해 나간다.

2) 실제 model에 따른 PID제어기 설계 과정

제어기가 들어가지 않은 독립진자 시스템의 전달 함수는 다음과 같다.

$$G_s(s) = \frac{1.414s^2}{s^4 + 1.375s^3 - 15.91s^2 - 19.27s}$$

안정도 판별법(Pzamp)에 따라 안정도를 확인하면,

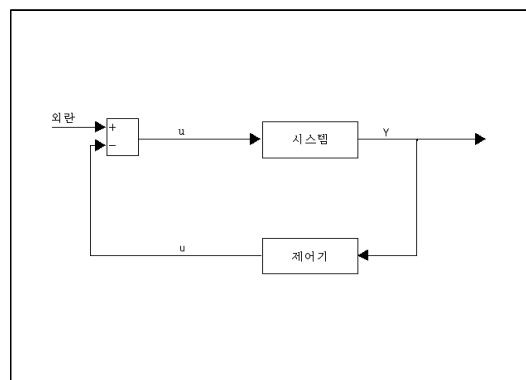


위 그림을 보면 우반부에 극점이 존재함을 볼수가 있다, 이는 곧 시스템이 불안정하다는 증거이다. 우리는 PID제어기를 통해 이를 안정화시켜야 한다.

(1) 단일 페루프 방식의 PID 제어기 전달함수는 다음과 같이 쓸 수 있다.

$$G_c(S) = K_p + K_d s + \frac{K_i}{s}$$

우리가 사용하게 될 시스템의 개략도는 아래와 같다.



사용하는 제어시스템의 개략도

(2) PI부분에 적당한 Gain값 정하기 ( $K_D=0$  이라고 가정)

$$G_c(S) = K_p + \frac{K_i}{s}$$

이 단계에서 대략적이거나  $K_p, K_i$ 는 잡는 방법에는 Routh - Hurwitz 판별법을 이용한다. Routh - Hurwitz 판별법에 따라  $K_p, K_i$ 의 값을 구하면

$$K_p > 11.25$$

$$K_D > 1.35$$

위의 값의 범위에 있을 때에만 안정화 될 수 있다고 생각할 수 있다.

이 때  $K_p, K_i$ 의 비로서 최초 제어 상태를 본 다음 적절한 형태(simulink에서의 그래프)가 나올 때 까지

$K_p, K_I$ 의 비를 증가 시켜라. 즉  $K_p$ 의 값을  $1.35 \times \alpha$ 로  $\alpha$ 를 1에서 10까지 정도화 시킨다음 가장 적절한  $\alpha$ 를 지정한다(Routh - Hurwitz 판별법에 대한 자세한 내용은 부록으로 첨가했음).

여기서는  $\alpha$ 값을 6으로 잡았다. (여기서  $K_p, K_I$ 의 비율이 결정됨)

아래의 표에 따라 일정 비율로 각각  $K_p, K_I$ 를 증가 시킨다.

이득값	Rise time	Overshoot	Settling time	정상상태오차
$K_p$ (비례이득)	감소한다	증가한다	변화없음	감소한다
$K_I$ (적분이득)	감소한다	증가한다	증가한다	제거한다
$K_D$ (미분이득)	변화없다	감소한다	감소한다	변화없다

### (3) 페루프 전달함수

$$T(s) = \frac{G_s(s)}{1 + G_s(s)G_c(s)}$$

이 이상적으로 안정화 되는  $K_d$ 의 값을 선택한다.

(4) 이상의 과정을 반복하여  $K_p, K_I, K_d$ 값을 조정해 나간다. 결과로서 각각의 이득 값을 다음과 같이 추정하면

$$K_p = 1500$$

$$K_I = 30$$

$$K_D = 750$$

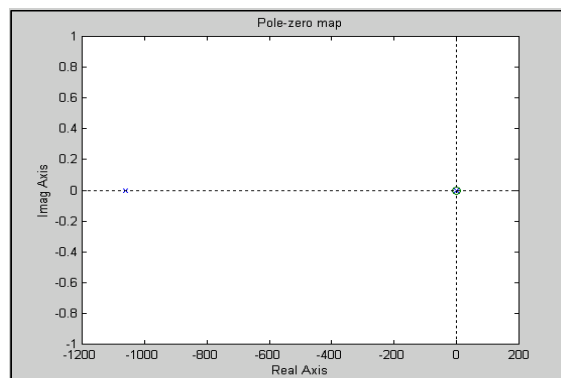
PID 제어기의 전달함수는 다음과 같다.

$$G_c(s) = \frac{750s^2 + 1500s + 30}{s}$$

위의 둘을 합한 전체 페루프전달함수는 다음과 같다.

$$T(s) = \frac{G_s(s)}{1 + G_s(s)G_c(s)} = \frac{1.414s^3}{s^5 + 1062s^4 + 2105s^3 + 23.15s^2}$$

PID 제어기가 들어간 페루프제어시스템의 pzmap 그래프는 아래와 같다.

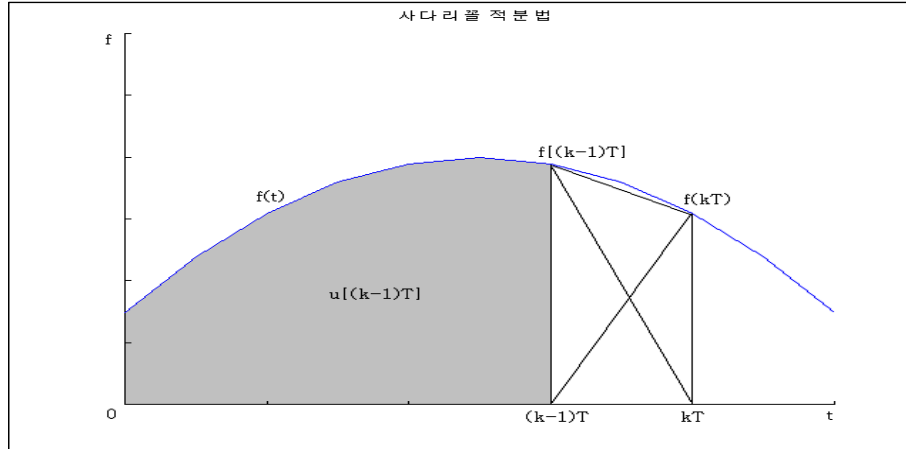


PZMAP

우반평면의 극점이 좌반평면쪽으로 이동했음을 알 수 있다. 이는 곧 우리가 결정한 PID 제어기는 이론적으로 도입진자시스템을 안정화시켜준다는 것을 말한다.



### 3) PID 제어기의 Digital control law



디지털 PID 제어기를 위한 사다리꼴 적분법

연속계의 PID 보상기의 전달함수는 다음과 같다.

$$G_c(s) = K_p + \frac{K_I}{s} + K_D s = \frac{K_D s^2 + K_p s + K_I}{s}$$

이와 같은 함수를 디지털적으로 근사화(여기서 샘플링 주기의 선택은 극히 중요하며 샘플링 주기의 값은 충분히 작아서 디지털화가 아주 정확하여야 한다)하는데 사용될 수 있는 많은 수치적분법이 있다. 함수의 면적을 수치적으로 근사화시키는 3가지 기본적인 방법은 사다리꼴 적분(trapezoidal integration), 전향 거형 적분(forward - rectangular integration)과 역방향 - 거형 적분 (backward - rectangular integration)이다. 이 중에서 사다리꼴 적분에 대해서만 이 논문에서는 거론하도록 한다.

사다리꼴 적분법(trapezoidal integration)은 위의 그림처럼 함수  $f(t)$ 의 아래 면적을 사다리꼴 급수로 근사화하는 것이다.  $t = kT$ 에서 계산된  $f(t)$ 의 적분을  $u(kT)$ 로 표기 하면 다음 식을 얻는다.

$$u(kT) = u[(k-1)T] + \frac{T}{2} [f(kT) + f[(k-1)T]]$$

여기서  $(k-1)T \leq t \leq kT$  에서  $f(t)$  아래의 영역은 그 구간내에 사다리의 면적으로 근사화된다. 위의 식의 방식대로 PID제어기에 양변에 Z-변환을 취하면 아래와 같은 디지털 적분기의 전달함수를 얻는다. 위의 식의 방식대로 연속계 PID 제어기의 양변에 Z-변환을 취하면 아래와 같은 디지털 적분기의 전달함수를 얻을 수 있다.

$$G(z) = \frac{u}{x} = \frac{az^2 + bz + c}{z(z-1)}$$

$$a = -k_p + \frac{Tk_I}{2} + \frac{k_d}{T}$$

$$b = \frac{Tk_I}{2} - k_p - 2\frac{k_d}{T}$$

$$c = \frac{k_d}{T}$$

### 4) 디지털 PID 제어기의 설계

실제 C-프로그램상에서 디지털 PID 제어기는 다음과 같은 알고리즘으로 작동한다.

## 디지털 PID 제어기의 알고리즘

$$x_{m2} = 0$$

$$x_{m1} = 0$$

$$u_{m1} = 0$$

$$u_k = u_{k-1} + ax_k + bx_{k-1} + cx_{k-2}$$

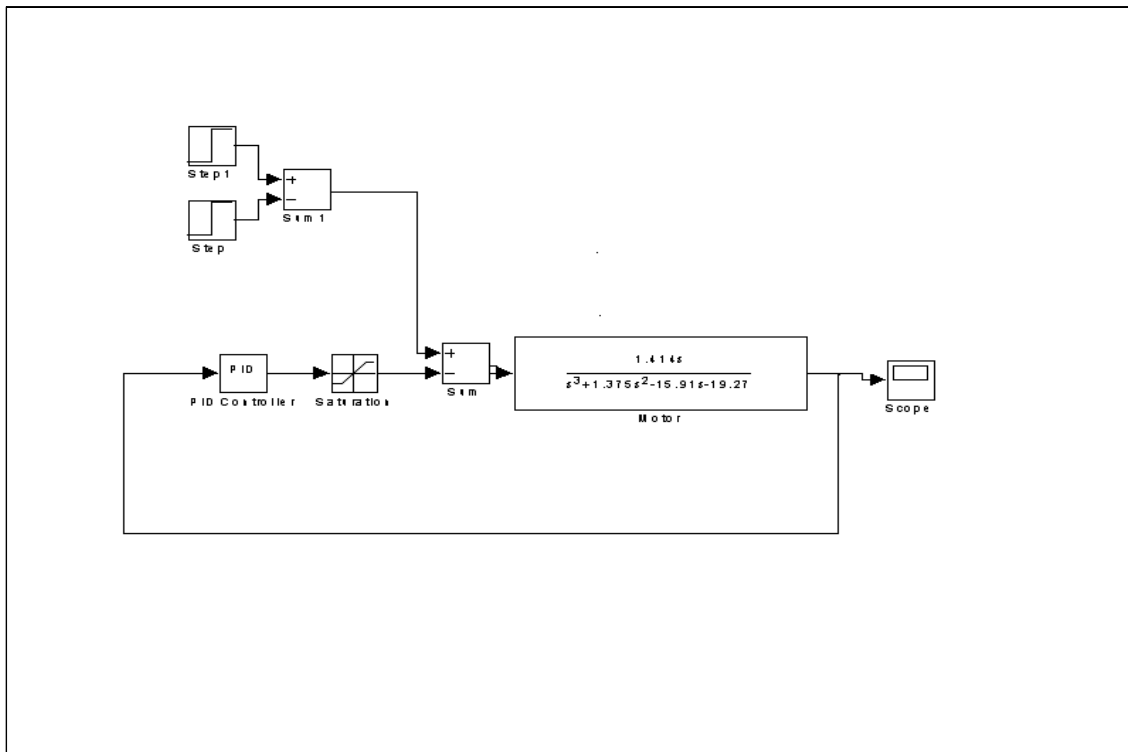
$$x_{m2} = x_{m1}$$

$$x_{m1} = x$$

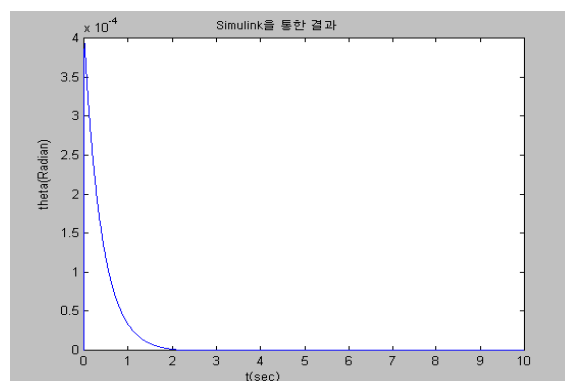
$$u_{m1} = u$$

### 4. 시뮬레이션

Simulink를 통한 이론 결과

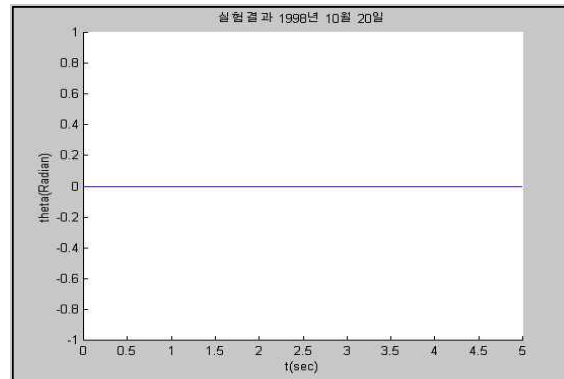


PID 제어시스템의 시뮬링크 모델



Bar의 변위

## 5. 실제 실험결과



Bar의 변위

Data값이 눈에 보이지 않는 Noise때문에 작은 외란에 대해서만 실험을 할 수 밖에 없었다. 그리고 엔코더의 분해능은 0.006rad 이다. 실제 상황에서는 움직이나 결과에서 보면 항상 0인 것 처럼 보이나 데이터로 보면 0.001초동안 제어가 이루어진다.

## 6. 결론

PID제어기 적용에 있어 PID에 관한 이론적인 내용이 많이 부족하여 나름대로의 방식으로 PID Gain 값을 찾는 방법을 생각함에 있어 여러 가지 문제점이 보였다.

Simulink상에서 이론적인 실험을 통해  $K_p, K_i, K_d$ 값의 특성과 다른 참고 문헌에서의 Gain값의 특징에 대해 실제 Modeling에 적용했을 때 그 변화는 눈으로 확인해도 전혀 예상하지 못한 결과를 얻게 되었다. 그것은 System의 Noise와 그 외 여러 가지 예상하지 못한 외란의 작용에 의해서라고 생각된다.

오차의 원인으로는

- 1) 인식할 수 있는 엔코더의 범위가 너무 크다.
- 2) 회로상에서 Noise가 Random하게 작용한다.
- 3) 환경적인 요인에 의하여 발생할 수 있는 요인을 지적 할 수 있다.

앞에서 실험한 Gain값이 현재 우리가 보유하고 있는 최적의 Gain값이 아니라 저의 나름대로의 방식을 고려한 최적의 방식임을 여기서 분명히 밝힌다. 물론 여기서 고려한 사항은 모터에 과부하가 걸리지 않는 범위내에서 우리가 알 수 없는 오차를 최대한 무시하도록 하기위해서 최대값으로  $K_p$ 값을 설정했다.

Trial and Error에 의해서 결정한 Gain값에 의하면 물론 더욱 강력한 제어기 설계가 가능하지만 여기서는 순수 학문의 적용이라는 취지에 맞추어 측정하지 못한 외란에 의한 영향을 무시하였다.