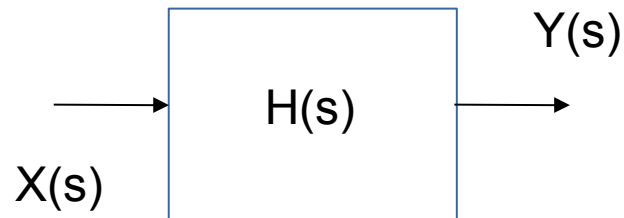


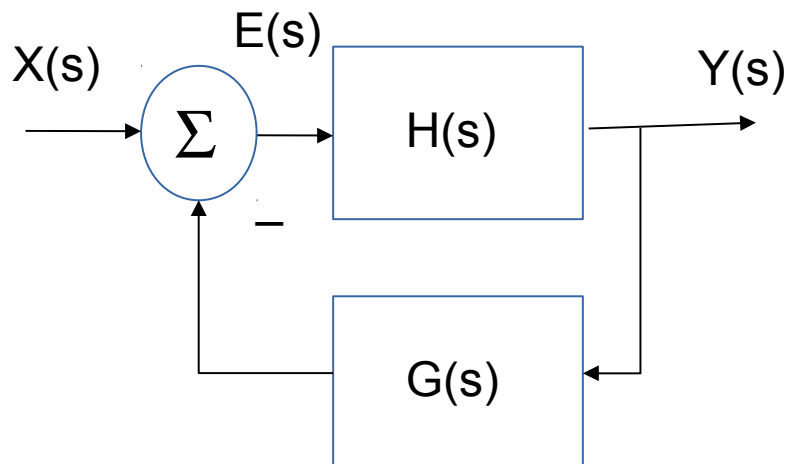
# Close Loop Transfer Function



Start with open loop, where plant  $H(s)$  can be a stepper motor for example, so we have

$$Y(s) = H(s) X(s) \quad \dots \quad (1)$$

Now we can add a sensor to form a feedback loop as follows



So, sensor  $G(s)$  forms a feedback loop as shown in Figure 2.

$$E(s) = X(s) - G(s) Y(s) \quad \dots \quad (2)$$

$$Y(s) = H(s) E(s) \quad \dots \quad (3)$$

From (2) we have

$$X(s) = E(s) + G(s) Y(s) \quad \dots \quad (4)$$

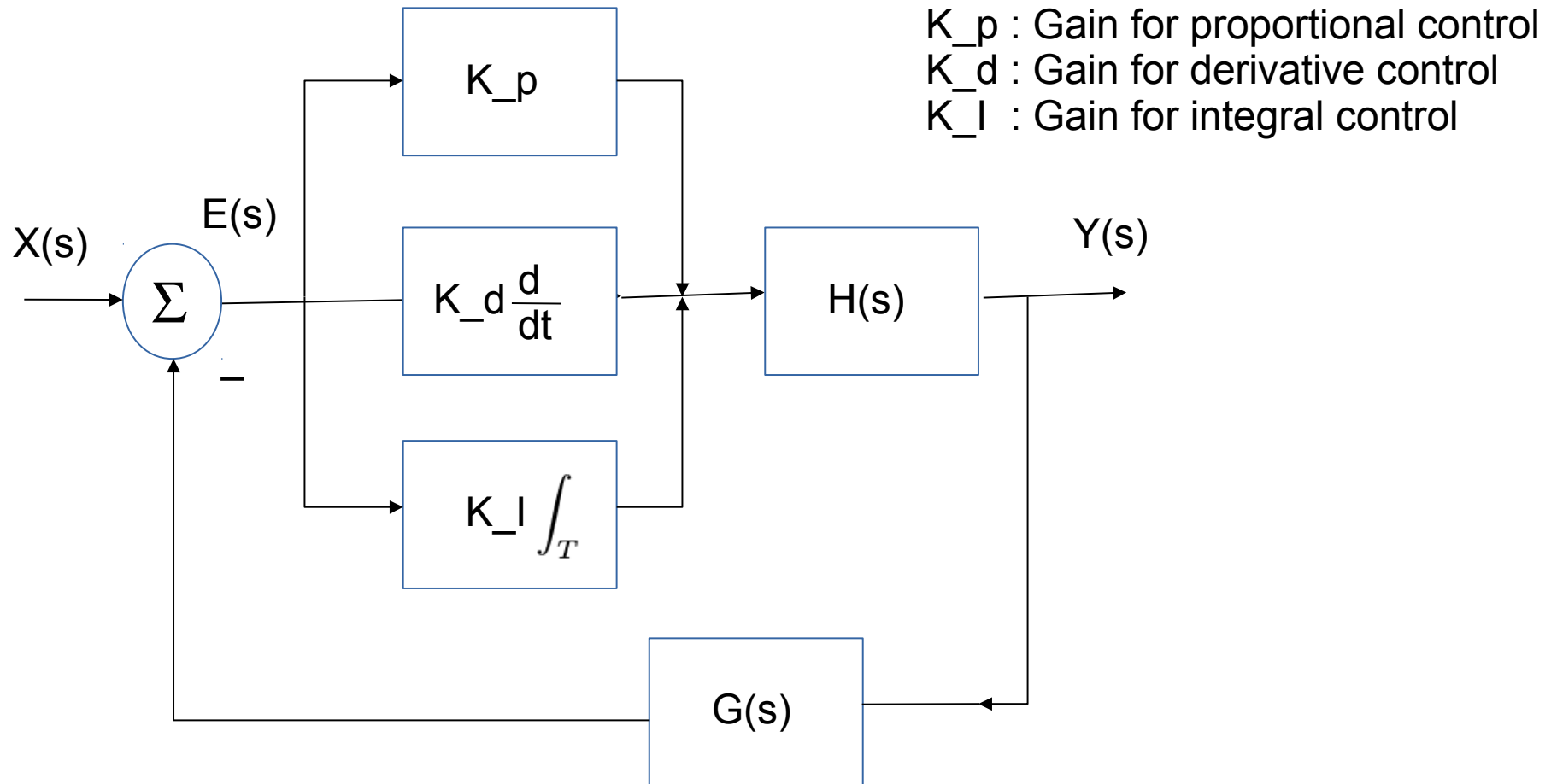
Substitute (3), we have

$$X(s) = E(s) + G(s) H(s) E(s)$$

Hence, the transfer function is

$$\frac{Y(s)}{X(s)} = \frac{H(s)}{1 + G(s) H(s)} \quad \dots \quad (5)$$

# PID Control

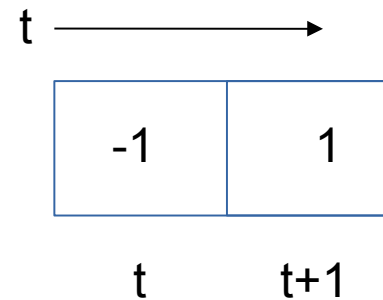


# Central Difference And Its Kernel

Forward difference:

$$FD = \frac{de(t)}{dt} \approx e(t+1) - e(t)$$

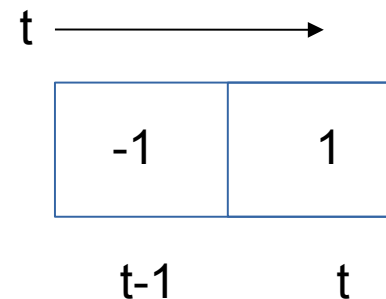
Map to a kernel



Backward difference:

$$BD = \frac{de(t)}{dt} \approx e(t) - e(t-1)$$

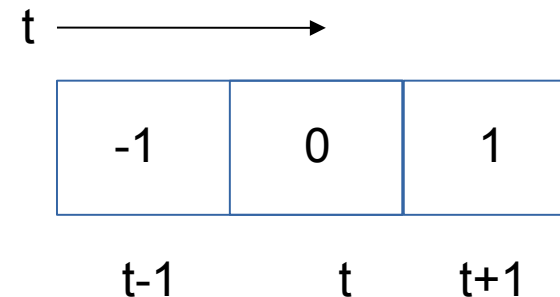
Map to a kernel



Central difference:

$$CD = \frac{1}{2}(FD + BD) = \frac{1}{2}[e(t+1) - e(t-1)]$$

Map to a kernel



# LoG Kernel And Convolution

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/(2\sigma^2)} \quad \dots(1)$$

Then taking the 2<sup>nd</sup> order derivative of equation (1), as shown below (note I have done 2D example on LoG, in the case of 1D, LoG, just let 2<sup>nd</sup> independent variable  $y = 0$ ).

$$\frac{\partial^2}{\partial x^2} G(x,y) = -\frac{1}{\sqrt{2\pi}\sigma^3} e^{-\frac{x^2+y^2}{2\sigma^2}} + \frac{x^2}{\sqrt{2\pi}\sigma^5} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \dots (4)$$

Based on the result from equation (4) (let  $y = 0$  for 1D case), so we can construct K by K kernel (K as an odd number) as illustrated below:

Example: Illustration of K by K kernel



Evaluate each coefficient of the K by K kernel, by setting the  $x = 0$  at the center of the kernel, and then you can find coefficient for both  $x = +1$  and  $x = -1$  (symmetric, they are equal, and so on.)

# LoG Kernel And Convolution

## Derive LoG Kernel

Example: LoG stands for Laplace of Gaussian  
First, Laplace operator is given as:

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \quad \dots (1)$$

First, 1D Gaussian function  

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Then, 2D Gaussian function,  

$$G(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu_x)^2 + (y-\mu_y)^2}{2\sigma^2}} \quad \dots (5)$$

Assume  $\mu_x = \mu_y = 0$

$$\frac{\partial}{\partial x} G(x, y) = -\frac{x}{\sqrt{2\pi}\sigma^3} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \dots (3)$$

Hence

$$\frac{\partial^2}{\partial x^2} G(x, y) = -\frac{1}{\sqrt{2\pi}\sigma^3} e^{-\frac{x^2+y^2}{2\sigma^2}} + \frac{x^2}{\sqrt{2\pi}\sigma^5} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \dots (4)$$

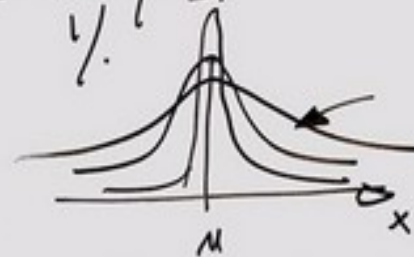
$$\frac{\partial^2}{\partial y^2} G(x, y) = -\frac{1}{\sqrt{2\pi}\sigma^3} e^{-\frac{x^2+y^2}{2\sigma^2}} + \frac{y^2}{\sqrt{2\pi}\sigma^5} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \dots (5)$$

$$\nabla^2 G(x, y) = \frac{x^2+y^2-2\sigma^2}{\sqrt{2\pi}\sigma^5} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

# 1D LoG Kernel

March 5th, 2018 ONPE242 Embedded Hardware Harry Li.

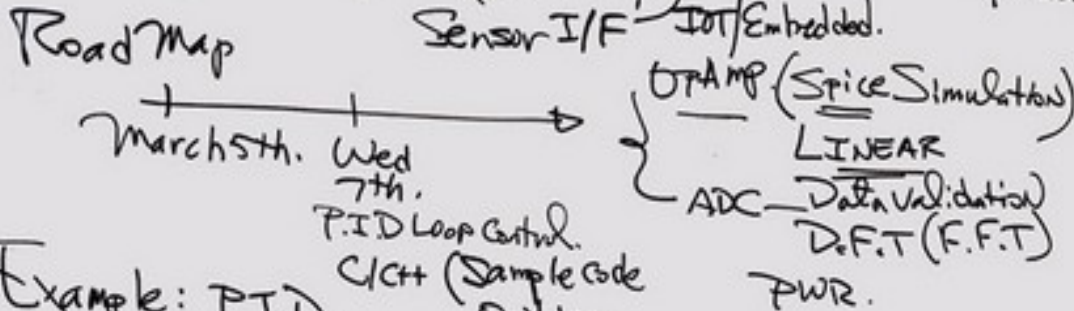
- 1) Midterm Exam: March 21st. (Wed)
- 2) Review 1 (ONGPP), Review 2 (ON PWM Timing) Posted on github/hualili
- 3) Lab 1 (Close Loop w/ FWA Motor Drive) Due March 19th. (Monday)



$$G(x,y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (\text{see P.P.T.})$$

- ①  $\sigma^4$  More Noisy Data
- ② Kernel size  $\uparrow$ , to process more noisy data.

Let  $y=0$ , then  $\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}$  ... (2)



Example: P.I.D.

$K_P, K_I$

$K_I \sum e^2(t)$  C.D. =  $\frac{1}{2} [F.D. + B.D.]$  Kernel

$\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-u)^2}{2\sigma^2}}$  ... (1)

Log  $\Delta^2 G(x,y)$

Robust Kernel Log

Computational C/C++

Kernel  $\begin{bmatrix} -1 & +1 \end{bmatrix}$

3x1 G.I. Rows.

$\left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) G(x,y) \dots (1)$

$\frac{\partial}{\partial x} G(x,y) = \frac{\partial}{\partial x} \left( \frac{\partial}{\partial x} G(x,y) \right)$

Where "Chain" Rule

$\frac{\partial}{\partial x} G(x,y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-u)^2+y^2}{2\sigma^2}} \cdot \frac{d}{dx} \left( -\frac{(x-u)^2}{2\sigma^2} \right)$

(See eqn (3) on P.P.T.)

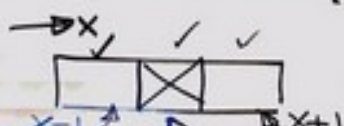
Then  $\frac{\partial}{\partial x} \left( \frac{x}{\sqrt{2\pi\sigma^3}} e^{-\frac{x^2+y^2}{2\sigma^2}} \right)$  See Eqn (4) on P.P.T. for  $y$



# Calculate PID of Error

CMPE242 March 5th 2018 2/ Harry Li

Derive the kernel. (K by 1)



1° Identify the center of the kernel.

2° Kernel weights (coefficients)

$$\nabla^2 G(x, y)|_{y=0} = \nabla^2 G(x) \text{ (from Eqn (a))}$$

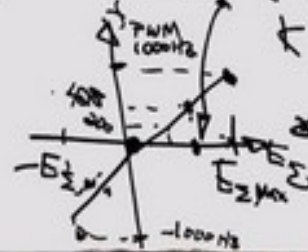
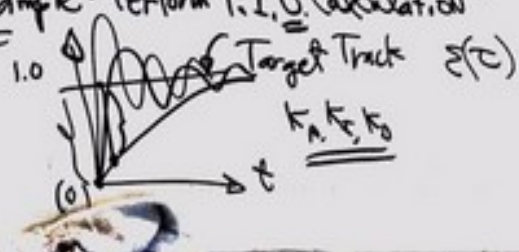
$$\text{a) Kernel Location, } x=0, \nabla^2 G(x)|_{x=0} = \frac{1}{\sqrt{2\pi}\sigma^2} (x-\delta^2) e^{-\frac{x^2}{2\sigma^2}}$$

$$= \frac{1}{\sqrt{2\pi}} (0-\delta^2) e^{-\frac{0}{2\sigma^2}} = -\frac{\delta^2}{\sqrt{2\pi}\sigma^2} \cdot 1 = -\frac{1}{\sqrt{2\pi}\sigma^2}$$

$$\text{a) Kernel Location, } x=1, \nabla^2 G(x)|_{x=1} = \frac{1}{\sqrt{2\pi}\sigma^2} (1-\delta^2) e^{-\frac{1^2}{2\sigma^2}}$$

$$\text{a) Kernel Location } x=-1, \nabla^2 G(x)|_{x=-1} = \nabla^2 G(x)|_{x=1}$$

Example: Perform P.I.D. Calculation



Sensor Current Location  $\rightarrow$  Compare w/ the target desired Location  $\rightarrow$  Error  $e(t)$

Location	$E(t)$	$K_p E(t)$	$K_i \frac{d}{dt} E(t)$	$K_d \frac{d}{dt} E(t)$
$L(0)$	$D(0) - L(0) = E(0)$	$K_p E(0)$	—	—
$L(1)$	$D(1) - L(1) = E(1)$	$K_p E(1)$	$K_i (E(1) - E(0))$	$K_d (E(1) - E(0))$
$L(2)$	$D(2) - L(2) = E(2)$	$K_p E(2)$	$K_i (E(2) - E(1))$	$K_d (E(2) - E(1))$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$L(k)$	$D(k) - L(k) = E(k)$	$K_p E(k)$	$K_i (E(k) - E(k-1))$	$K_d (E(k) - E(k-1))$

$x=0$

replace by C.D.

$$\frac{1}{2} K_d (E(2) - E(1))$$

$$\frac{1}{2} K_i (E(2) - E(1))$$

$$\frac{1}{2} K_d (E(k) - E(k-1))$$

$$E(s) = \sum_{t=0}^{\infty} E(t) e^{-st}$$

$$K_p E(t) + K_i \frac{d}{dt} E(t) + K_d \frac{d}{dt} E(t)$$

# Convolution With LoG

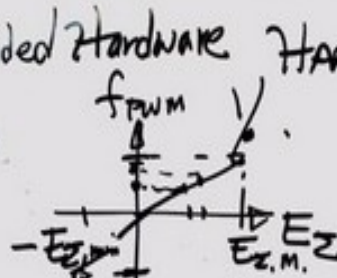
March 7, 18 CMPE242 Embedded Hardware Harry Li

1) March 19th, Lab 1 Due

2) March 21st, midterm.

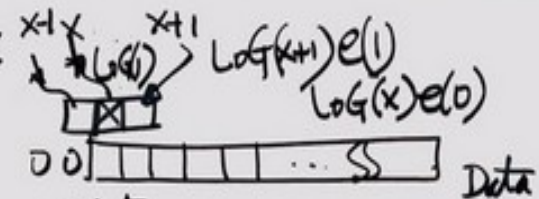
(Review 1 & 2 On github;  
Review 3 & 4 to be posted)

3) PID C code Sample to  
Be Posted on github today



$$E_c = K_P E(t) + K_D \frac{d}{dt} E(t) + K_I \int E(t) dt$$

$$F.D. = E(t+1) - E(t)$$



$$\int_{\Omega} e(\tau) K(t-\tau) d\tau \dots (1)$$

$$\sum_{j \in \Omega} e(j) K(i-j) \dots (1^*)$$

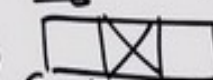
(1) Shift  
(2) Multiplication

(3) Summation/Addition  
"Under" the kernel

Example: Derivative Control.  $K_D \frac{d}{dt} E(t)$

Compute  $K_D \frac{d}{dt} E(t)$  w LoG kernel

LoG kernel, 3x1



First, Evaluate LoG kernel

$E(t)$	$K_D \frac{d}{dt} E(t)$
$\tau=0$ 1.0	—
$\tau=1$ 1.5	0.5 (1.5-1.0)
$\tau=2$ 2.25	0.75 (2.25-1.5)
$\vdots$ 1.8	-0.45 (1.8-2.25)
0.8	-1.0 (0.8-1.8)
0.9	0.1 (0.9-0.8)

$$\text{LoG}(1D \text{ Convolution}) \rightarrow \text{First, Evaluate LoG kernel}$$

$$\text{LoG}(x+1)e(1) + \text{LoG}(x)e(0) = 1 \cdot \text{LoG}(0) + 1.5 \cdot \text{LoG}(1)$$

$$\text{LoG}(x+1)e(2) + \text{LoG}(x)e(1) + \text{LoG}(x-1)e(0) = 2.25 \text{LoG}(x) + 1.5 \text{LoG}(x) + 1 \cdot \text{LoG}(x-1) \rightarrow 1.5 \text{LoG}(x) + \text{LoG}(x+1) (1.0 + 2.25)$$

$$\text{LoG}(x+1)e(3) + \text{LoG}(x)e(2) + \text{LoG}(x-1)e(1) = 1.8 \text{LoG}(x+1) + 2.25 \text{LoG}(x) + 1.5 \text{LoG}(x-1) \rightarrow$$

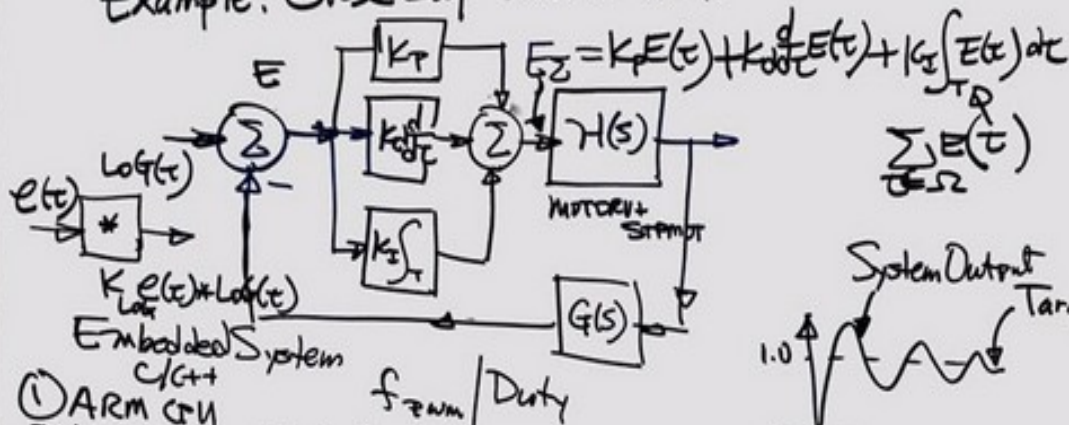
$$\text{LoG}(x+1)e(4) + \text{LoG}(x)e(3) + \text{LoG}(x-1)e(2) = 0.8 \text{LoG}(x+1) + 1.8 \text{LoG}(x) + 2.25 \text{LoG}(x-1) \rightarrow$$



# E-Sigma Mapping To PWM Control

CNPE242 Embedded Hardware March 7th, 2018 HL.2/.

Example: Close Loop P.I.D Control.



int Mapping-Control (float e-Sigma,  
C/C++ module. int freq-PWM,  
int duty-cycle);

For  $LoG(t) * e(t)$ , the controller:

$$E_{\Sigma} = K_P E(t) + K_{LoG} e(t) * LoG(t) + K_I \int e(t) dt \quad \dots (3)$$



$$a = \frac{y_2 - y_1}{x_2 - x_1} = \frac{2000}{E_{\Sigma, \max} - E_{\Sigma, \min}} \quad \dots (2)$$

$$b = -ax_1 + y_1 = 0 \quad \dots (2')$$

① ARM CPU  
PWM Output  
② MOTOR  
(motor Drive)

Write C/C++  
Mapping-Control:

$$\frac{y - y_1}{y_2 - y_1} = \frac{x - x_1}{x_2 - x_1}$$

$$y - y_1 = \frac{x - x_1}{x_2 - x_1} (y_2 - y_1)$$

$$y = \frac{y_2 - y_1}{x_2 - x_1} (x - x_1) + y_1$$

$$y = ax + b \quad \dots (1)$$

where

$$a = \frac{y_2 - y_1}{x_2 - x_1}$$

$$E_{\Sigma}(3) = K_P E(3) + K_D \frac{dE(3)}{dt} + K_I \int_{t=0}^{t=3} E(t) dt$$

if  $K_P = 0$

$$= K_D (-0.45) + K_I (E(0) + E(1) + E(2) + E(3))$$

$$= -0.45 K_D + K_I (1.0^2 + 1.5^2 + 2.25^2 + 1.8^2) \quad | K_D = K_I = 1$$

$$= -0.45 + 10.5 = 10.05$$