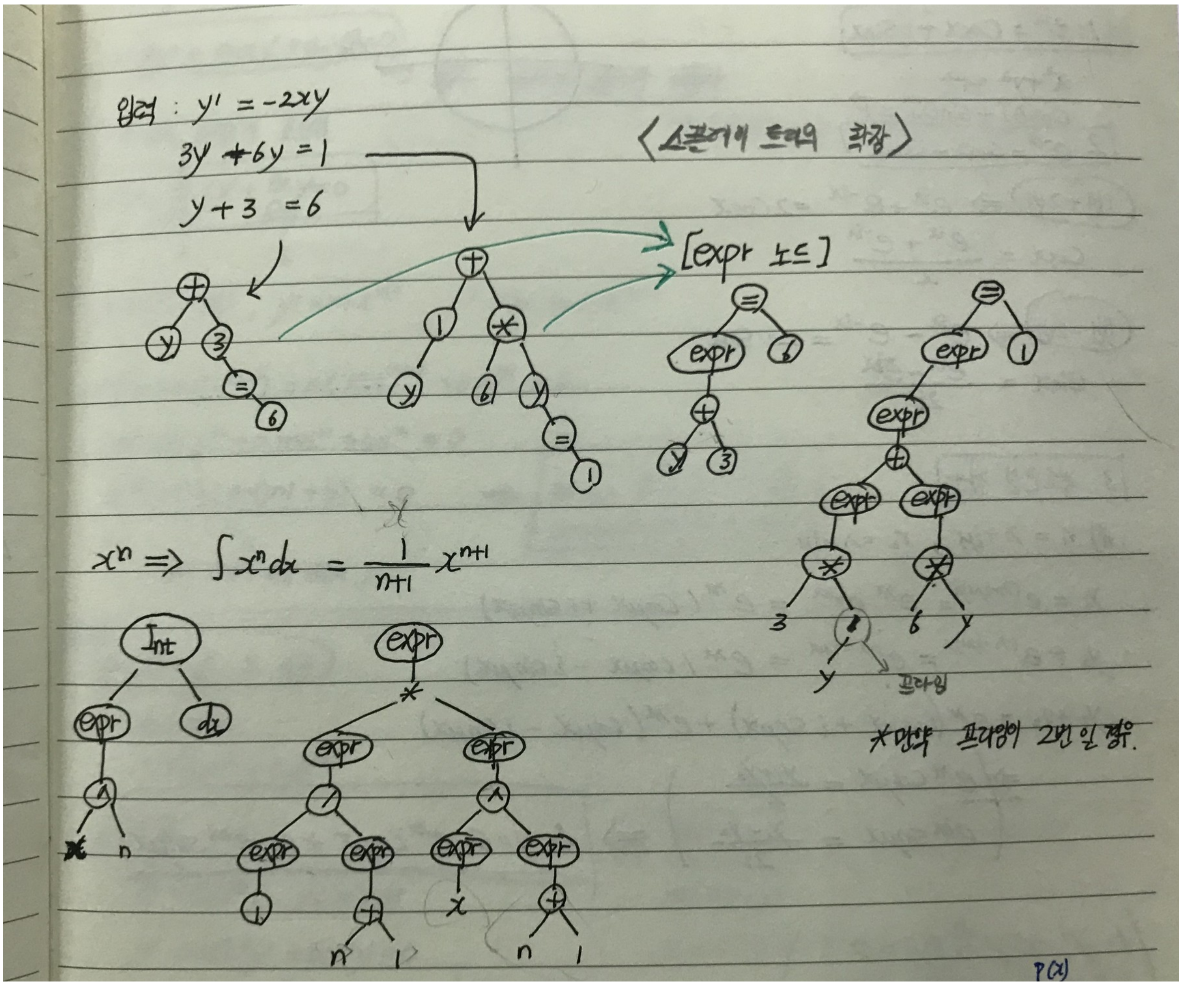
Xilinx Zynq FPGA,TI DSP, MCU 기반의 프로그래밍 전문가 과정

날 짜 : 2018 . 5. 22

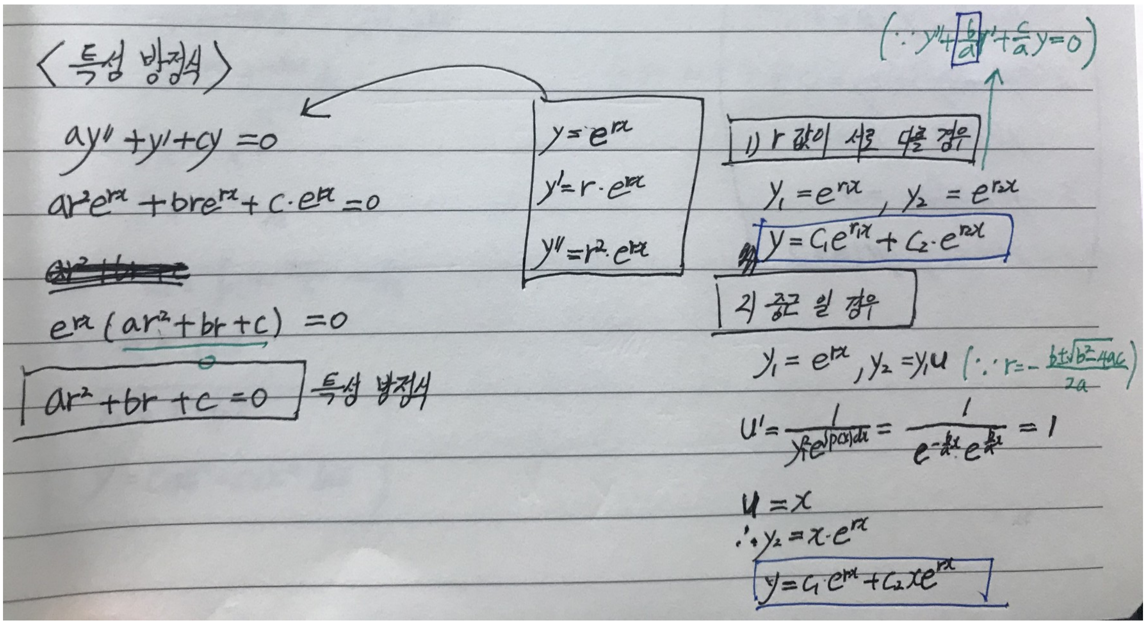
강사 – Innova Lee(이상훈)
gcccompil3r@gmail.com
학생 – 정한별
hanbulkr@gmail.com

수학

주말 숙제: 다시한번 설명.

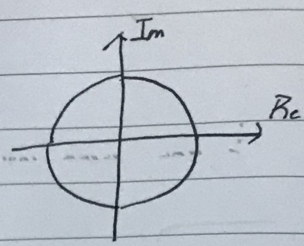


특성방정식.



오일러 공식 & 복소근일 경우

※ 오일러 공식



$$\boxed{1.\ e^{ix} = \cos x + i\sin x}$$

$$x^2+y^2=r^2$$

$$\cos\theta\hat{i}+\sin\theta\hat{j}=\vec{r}$$

$$\boxed{2.\ e^{-ix} = \cos x - i\sin x}$$

1번+2번 $\Rightarrow e^{ix}+e^{-ix} = 2\cos x$

$$\cos x = \frac{e^{ix}+e^{-ix}}{2}$$

1번-2번 $\Rightarrow e^{ix}-e^{-ix} = 2i\cdot\sin x$

$$\sin x = \frac{e^{ix}-e^{-ix}}{2i}$$

3. 복소근일 경우

if) $r_1 = \lambda + i\mu$, $r_2 = \lambda - i\mu$

$$y_1 = e^{(\lambda+i\mu)x} = e^{\lambda x}\cdot e^{i\mu x} = e^{\lambda x}(\cos\mu x + i\sin\mu x)$$

$$y_2 = e^{(\lambda-i\mu)x} = e^{\lambda x - i\mu x} = e^{\lambda x}(\cos\mu x - i\sin\mu x)$$

$$y_1+y_2 = e^{\lambda x}(\cos\mu x + i\sin\mu x) + e^{\lambda x}(\cos\mu x - i\sin\mu x)$$

$$\Rightarrow \begin{pmatrix} e^{\lambda x}\cos\mu x = \frac{y_1+y_2}{2} \\ e^{\lambda x}\sin\mu x = \frac{y_1-y_2}{2i} \end{pmatrix} \Rightarrow \boxed{\therefore\ y = c_1\cdot e^{\lambda x}\cos\mu x + c_2\cdot e^{\lambda x}\sin\mu x}$$

1. $y'' + 5y' + 6y = 0$
   $y(0)=1$, $y'(0)=3$
2. $3y'' + 12y' + 12y = 0$
   $y(0)=3$ $y'(0)=17$
3. $y'' + 4y' + 5y = 0$
   $y(0)=1$, $y'(0)=2$

cauchy - Euler(오일러) 방정식

연립 미분방정식이 | 미분방정식이 된다.

행렬이 가능

필터

라플라스 트랜스폼 → 미분 방정식 (ODE)

- Cauchy - Euler 방정식.

$a_2x^2y'' + a_1xy' + a_0y = 0$

↳ 동차 숫자가 있으면 비동차.

↳ 회로에서 쓴다.

계수값을 없애는게 표준형.

$x^2y'' + \underbrace{\frac{a_1}{a_2}}_{a}xy' + \underbrace{\frac{a_0}{a_2}}_{b}y = 0$

$y = x^m$ (가정) , $y' = m \cdot x^{m-1}$, $y'' = m(m-1)x^{m-2}$

$x^2(m(m-1)x^{m-2}) + ax\,mx^{m-1} + bx^m = 0$

$(m^2-m)x^m + amx^m + bx^m = 0$

$x^m(\underbrace{m^2+(a-1)m+b}_{=0}) = 0 \Rightarrow \boxed{m = -\frac{(a-1) \pm \sqrt{(a-1)^2-4b}}{2}}$

코시 오일러 특성 방정식.

문제

1) 서로 다른 두 실근

$m = m_1 m_2$

$y = c_1x^{m_1} + c_2x^{m_2}$

2) 중근

$y_1 = c_1x^m$, $y_2 = y_1 \cdot u$

$y'' + ax^{-1}y' + bx^{-2}y = 0$

$u' = \frac{1}{y_1^2 e^{\int p(x)dx}} = \frac{1}{x^{2m} e^{a\ln x}}$

$\int u' = \int x^{-2m} \cdot \underbrace{e^{-a\ln x}}_{x^{-a}} dx$

$= \int x^{-2m} \cdot x^{-a}$

$= \int x^{-2m-a} dx$

$\int x^{-2m-a}dx = \int x^{+2 \cdot \frac{a-1}{2} - a} dx$

$\int x^{-1} dx = \ln x$

$\therefore y_2 = x^m \ln x$

$\boxed{y = c_1x^m + c_2x^m \cdot \ln x}$

3) 복소근

$m = \lambda \pm i\mu$

$y_1 = x^{\lambda} \cdot x^{i\mu} = x^{\lambda} \cdot e^{i\mu \cdot \ln x}$

$= x^{\lambda}\{\cos(\mu \ln x) + i\sin(\mu \ln x)\}$

$\therefore y = c_1x^{\lambda}\cos(\mu \cdot \ln x) + c_2x^{\lambda} \cdot \sin(\mu \cdot \ln x)$

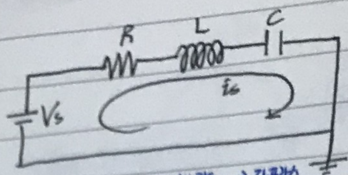
특정해

64 <특정해>

$y'' + a(x)y' + b(x)y = g(x)$

$\boxed{V_s = i_s R + L\frac{di}{dt} + \frac{1}{C}\int i_s\,dt}$

↳ 여기서 미분을 한번 한다.
그러면 2계 미분 방정식.



↳ 여기 들이 연결되면 계산 힘듦 → 라플라스 이용해야 함.
통신,

전류는 전기장과 반대.

$i_c = C\frac{dV_c}{dt}$ $q = CV$ $\frac{dq}{dt} = i$

★ 전원회로에서는
(코일과 콘덴서에 아주 짧게 전류가 흐른다.)
· 직류에도 걸린다는 말

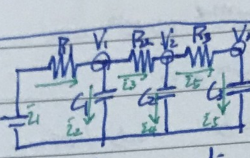
$y'' + a(x)y' + b(x)y = g(x)$

y_p = 특정해

y_g = 일반해 $y_g = y_p + y_h$

y_h = 동차해 $y_h = y_g - y_p$

$$\left(\begin{aligned} &y_g'' + a(x)y_g' + b(x)y_g = g(x) \\ &y_p'' + a(x)y_p' + b(x)y_p = g(x) \end{aligned}\right)$$



$i_c = C\frac{dV}{dt}$ $V_L = \frac{di}{dt}$

1. $\frac{V_s - V_1}{R_1} = C_1\frac{dV_1}{dt} + \frac{V_1 - V_2}{R_2}$
2. $\frac{V_1 - V_2}{R_2} = C_2\frac{dV_2}{dt} + \frac{V_2 - V_3}{R_3}$
3. $\frac{V_2 - V_3}{R_3} = C_3\frac{dV_3}{dt}$

↳ 라플라스 변환 필요.

$y_h'' + a(x)y_h' + b(x)y_h = g(x)$

$(y_g'' - y_p'') + a(x)(y_g' - y_p') + b(x)(y_g - y_p) = g(x)$

$\underbrace{y_g'' + a(x)y_g' + b(x)y_g}_{g(x)} - \underbrace{(y_p'' + a(x)y_p' + b(x)y_p)}_{g(x)} = \boxed{g(x)}$ $\Rightarrow 0$ → 동차는 0...

y_p(특정해) 조건

1) $g(x)$가 다항식인 경우
   $A_n x^n + A_{n-1}x^{n-1} \dots A_0$
2) $g(x)$가 지수함수인 경우
   Ae^{mx}
3) $g(x)$가 삼각함수인 경우
   $A\cos(mx) + B\sin(mx)$

$(y'' + 2y' + y = 2x^2 + 4)$

$y_h = C_1 e^{-x} + C_2 x e^{-x}$

$y_p = Ax^2 + Bx + C = 2x^2 - 8x + 16$

$y_p' = 2Ax + B$

$y_p'' = 2A$

· $2A + 4Ax + 2B + Ax^2 + Bx + C$

· $Ax^2 + (4A+B)x + (2A+2B+C) = 2x^2 + 4$

· $A = 2$, $B = -8$, $C = 16$

· $4A + B = 0$

· $2A + 2B + C = 4$

선생님이 주지 않은걸로 미방문제를 풀어봤음.

```
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include <math.h>

#define delta_x 0.001
#define E 2.71828182845904523536028740

void original_solve(double *A)
{
  double x = delta_x;
  double ii = x*-5000;
  printf("first num = %lf \n", ii);
  for(int i=1;i <= 10001;i++){
    A[i-1] = 3*pow(E,-pow(-5.000+x*(double)(i-1),2.0));
    printf("%11.6lf", A[i-1]);
    if(i%10 == 0)
      printf("\n\r");
  }
}

void second_solve(double *B)
{
  double x = delta_x;
  double ii = x*-5000;
  printf("first num = %lf \n", ii);
  for(int i=1;i < 5000;i++){
    B[i] = (-2*x*(x*(double)(i-1))+1)*B[i-1];
    printf("%11.6lf", B[i-1]);
    if(i%10 == 0)
      printf("\n\r");
  }
}

void percentage(double *A, double *B)
{
  for(int i=0; i<4999; i++)
  {
    printf("%11.6lf%%", A[i+5000]/B[i+1]*100);
    if(i%10 == 0)
      printf("\n\r");
  }
}

int main(void)
{
  printf("original solve is y = 3e^(-x^2).\n");
  printf("y' = -2xy , y(0) = 3. \n");

  printf("start 미방. \n");
```

```c
    double first_matrix[10001] = {0};
    double second_matrix[10001] = {0};

    second_matrix[0]=0;
    second_matrix[0] = 3.0;

    original_solve(first_matrix);
    printf("\n0 일때 : %.05lf \n", first_matrix[5000]);
    printf("\n");

    second_solve(second_matrix);
    printf("\n0 일때 : %.05lf \n", second_matrix[0]);

    percentage(first_matrix, second_matrix);

    return 0;
}
```

오차율은 x 가 증가함에 따라 크게 변화했다. 5 에 가까워 질때 쯤 9 프로 까지 차이난다. 아마도 0 에 가까워 질수록 자릿수가 너무 미세하여 컴퓨터가 표현하면서 오차가 커진것 같다.