TI DSP, MCU 및 Xilinx Zynq FPGA 프로그래밍 전문가 과정

강사 – Innova Lee(이상훈)
gcccompil3r@gmail.com
학생 – GJ (박현우)
uc820@naver.com

목차

수학

1) 계수가 상수항인 2계 미분방정식

○ 계수가 상수항인 2계 미분방정식

$ay'' + by' + cy = 0 \xleftarrow{\text{대입}}$ $y = e^{rx}$, $y' = r \cdot e^{rx}$, $y'' = r^2 \cdot e^{rx}$

$\Rightarrow ar^2e^{rx} + bre^{rx} + ce^{rx} = 0$

$e^{rx}(\underbrace{ar^2 + br + c}_{=0}) = 0$

$ar^2 + br + c$ (특성방정식)

1) r 값이 서로 다를 경우

$y_1 = e^{r_1x}$, $y_2 = e^{r_2x}$ $\left(\because y'' + \overset{p(x)}{\frac{b}{a}}y' + \frac{c}{a}y = 0\right)$

$y_1 = C_1 \cdot e^{r_1x} + C_2 \cdot e^{r_2x}$ //

2) 중근일 경우

$y_1 = e^{rx}$, $y_2 = y_1 u$ $\left(\because r = -\frac{b \pm \sqrt{b^2 - 4ac}}{2a}\right)$

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

$\int u'dx = x$

$\therefore y_2 = xe^{rx}$

$y = c_1e^{rx} + c_2xe^{rx}$ //

3) 복소근일 경우 (if: $r_1 = \lambda + iu$, $r_2 = \lambda - iu$)

$y_1 = e^{(\lambda + iu)x} = e^{\lambda x}e^{iux} = e^{\lambda x}(\cos(ux) + i\sin(ux))$

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

$y_1 + y_2 = 2e^{\lambda x} \cdot \cos ux$

$y_1 - y_2 = 2ie^{\lambda x}\sin ux$

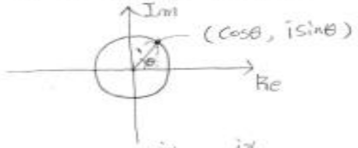
$\therefore y = c_1e^{\lambda x}\cos ux + c_2e^{\lambda x}\sin ux$ //

※ 오일러 공식

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

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

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



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

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

• 연습문제

1. $y'' + 5y' + 6y = 0$, $y(0) = 1$, $y'(0) = 3$

2. $3y'' - 12y' + 12y = 0$, $y(0) = 3$, $y'(1) = 7$

2) Cauch – Euler 방정식

○ Cauch – Euler 방정식

$$a_n x^n y^{(n)} + a_{n-1}x^{n-1}y^{(n-1)} + \cdots + a_1 x y' + a_0 y = g(x)$$

$\boxed{\text{○ 이계 Cauch-Euler}}$

$$a_2 x^2 y'' + a_1 x y' + a_0 y = 0$$

$$x^2 y'' + \underbrace{\frac{a_1}{a_2}}_{a} x y' + \underbrace{\frac{a_0}{a_2}}_{b} y = 0 \quad (\text{표준형})$$

대입

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

$\Rightarrow x^2(m(m-1)x^{m-2}) + a x m x^{m-1} + b x^m = 0$

$(m^2-m)x^m + a m x^m + b x^m = 0$

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

코시-오일러 특성방정식

1) 서로 다른 두 실근

$m = m_1, m_2$

$y = c_1 x^{m_1} + c_2 x^{m_2}$ //

2) 중근 (계수 내림법)

$y_1 = c_1 x^m$, $y_2 = y_1 u$

$y'' + a x^{-1} y' + b x^{-2} y = 0$

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

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

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

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

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

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

$y = c_1 x^m + c_2 x^m \ln x$ //

3) 복소근

$m = \lambda \pm iu$

$y_1 = x^\lambda x^{iu} = x^\lambda e^{iu\ln x}$

$\Rightarrow x^\lambda\{\cos(u\ln x) + i\sin(u\ln x)\}$

$\therefore y = c_1 x^\lambda \cos(u\ln x) + c_2 x^\lambda \sin(u\ln x)$ //

3) 비동차 2계 미분방정식

○ 비동차 2계 미분 방정식

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

- y_p = 특정해
- y_g = 일반해 $y_g = y_p + y_h$
- y_h = 동차해 $y_h = y_g - y_p$

① $y_h \rightarrow g(x) = 0$ 증명

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

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

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

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

$$\underbrace{y_g'' + a(x)y_g' + b(x)y_g}_{g(x)} - (\underbrace{y_p'' + a(x)y_p' + b(x)y_p}_{g(x)}) = \overset{0}{\overset{/\!/}{g(x)}}$$

* 특정해 (y_p) 조건

1) $g(x)$가 다항식인 경우

$$A_n x^n + A_{n-1}x^{n-1} + \cdots + A_0$$

2) $g(x)$가 지수함수인 경우

$$Ae^{mx}$$

3) $g(x)$가 삼각함수인 경우

$$A\cos(mx) + B\sin(mx)$$

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

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

$y_p = Ax^2 + Bx + C = \underline{2x^2 - 8x + 16}$

$y_p' = 2Ax + B$

$y_p'' = 2A$

$2A + 4Ax + 2B + Ax^2 + Bx + C$

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

$A = 2, \ B = -8, \ C = 16$

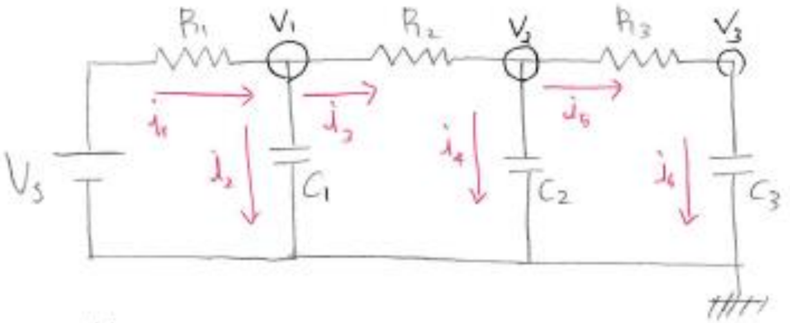
$4A + B = 0$

$2A + 2B + C = 4$

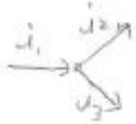
$\therefore \underline{y_g = C_1 e^{-x} + C_2 x e^{-x} + 2x^2 - 8x + 16}$ //

4) 라플라스 변환이 필요한 이유와 연립 미분방정식

○ 라플라스 변환이 필요한 이유와 연립미분 방정식



① 전류는 높은 곳에서 낮은 곳으로 흐름.



$i_1 = i_2 + i_3$

$i = C \cdot \frac{dV}{dt}$, $V = L \cdot \frac{di}{dt}$

1. $\frac{V_s - V_1}{R_1} = C_1 \frac{dV_1}{dt} + \frac{V_1 - V_2}{R_2}$

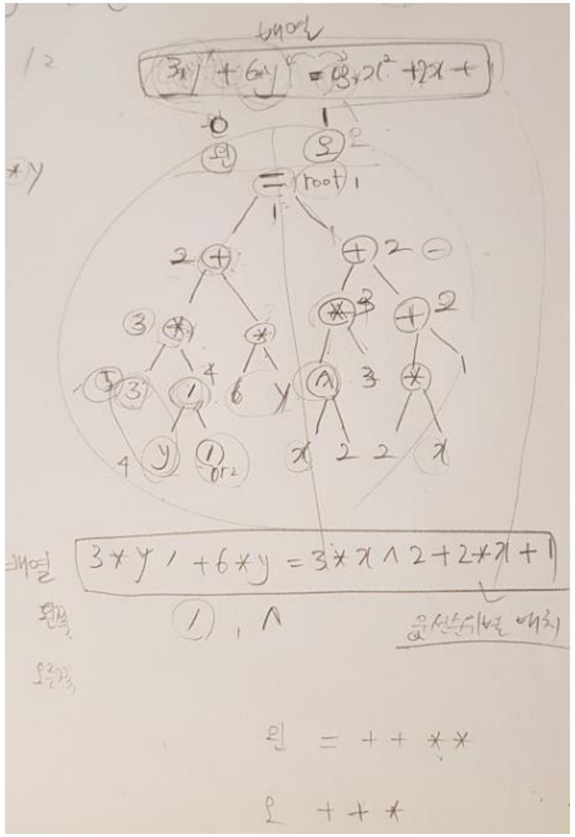
2. $\frac{V_1 - V_2}{R_2} = C_2 \frac{dV_2}{dt} + \frac{V_2 - V_3}{R_3}$

3. $\frac{V_2 - V_3}{R_3} = C_3 \frac{dV_3}{dt}$

→ 라플라스 변환 필요.

5) Expression tree 프로그래밍

<concept>



<header>

```c

#include<stdio.h>
#include<malloc.h>
#include<string.h>
#include<math.h>
#include<stdlib.h>
#include<unistd.h>

#define LEFT    0
#define RIGHT   1

enum operand{
    /*equal, plus, minus, mul, div, prime, exponent, constant*/
    EQ = 1,
    PLUS = 2,
    MINUS = 2,
    MUL = 3,
    DIV = 3,
    PRI = 4,
    EXPO = 4,
    CONS = 5
}

typedef struct expression_node{
    int prio;
    int lr;
}expr;

typedef struct expr_tree{

    char op;

    struct expr_tree *left;
    struct expr_tree *right;
}expr_t;

expr *get_expr_t_node(void){

    expr_t *tmp = (expr_t *)malloc(sizeof(expr_t));
    tmp->left = NULL;
    tmp->right = NULL;
    return tmp;
}

void expr_t_ins(expr_t **root, expr ex, char op ){

    if(*root == NULL){
        *root = get_expr_t_node();
        (*root)->op = op;
    }

}

#endif
```

Expression tree 컨셉 잡는 것도 그렇고 만드는데,

생각이 좀 많이 필요할 것 같습니다. 만드려고 노력중입니다.