

รายงานความคืบหน้า Mini-Project
การสร้าง GUI Application ด้วย MATLAB App Designer

จัดทำโดย

นาย ปริญญา ใจหาญ 6201012620155

นาย ปิณณวัฒน์ กิ่งแก้ว 6201012620163

นางสาว ตุลยา สารโพคา 6201012620341

นาย ธนากร บริบูรณ์ 6201012630045

เสนอ

อาจารย์เรวัต ศิริโกคาภิรมย์

ภาคเรียนที่ 1 ปีการศึกษา 2564

คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

สาขาวิศวกรรมคอมพิวเตอร์

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

Mini-Project: การสร้าง GUI Application ด้วย MATLAB App Designer

หัวข้อที่เลือก: การวิเคราะห์วงจร RLC และหาผลตอบสนองต่อสัญญาณ input

วัตถุประสงค์: เพื่อสร้าง Application ที่ผู้ใช้สามารถนำไปใช้วิเคราะห์วงจร RLC และดูผลตอบสนองที่ออกมาได้ด้วย MATLAB App Design

ขอบเขตของงาน:

- ผู้ใช้สามารถตั้งค่าของ Resistor, Inductor และ Capacitor ได้
- ผู้ใช้สามารถเลือกได้ว่าจะใช้แรงดันตกคร่อมที่ตัวใดเป็น output ได้
- โปรแกรมสามารถแสดงผลตอบสนองต่อสัญญาณ input เป็นรูปกราฟในช่วงเวลาที่ผู้ใช้กำหนดได้
- โปรแกรมสามารถแสดงรูปกราฟสำหรับ Frequency Response ได้

หลักการและทฤษฎีที่นำมาใช้:

Transfer Function คือ แบบจำลองทางคณิตศาสตร์ที่อยู่ในรูปอัตราส่วนระหว่าง output กับ input ใน s โดเมน เขียนได้เป็น

$$G(s) = \frac{Y(s)}{X(s)}$$

โดยที่ $Y(s)$ คือ output และ $X(s)$ คือ input

ใน RLC สามารถหา Transfer Function ได้จากการแปลง Laplace กับสมการเหล่านี้

- Kirchoff's voltage law

$$v(t) - v_R(t) - v_L(t) - v_C(t) = 0 \quad (\text{eq. 1})$$

- Resistor (Ohm's law)

$$v_R(t) = Ri(t) \quad (\text{eq. 2})$$

- Inductor

$$v_L(t) = L \frac{di}{dt} \quad (\text{eq. 3})$$

- Capacitor

$$i(t) = C \frac{dv_C}{dt} \quad (\text{eq. 4})$$

เมื่อแปลง Laplace ออกมาแล้วจะได้เป็น

$$V(s) - V_R(s) - V_L(s) - V_C(s) = 0$$

$$V_R(s) = RI(s)$$

$$V_L(s) = LsI(s)$$

$$I(s) = CsV_C(s)$$

โดยเมื่อเราแก้ 4 สมการข้างต้นจะทำให้ได้ค่าของแรงดันตกคร่อมและค่ากระแสดังนี้

$$I(s) = \frac{V(s)}{R + Ls + 1/Cs}$$

$$V_R(s) = \frac{RV(s)}{R + Ls + 1/Cs}$$

$$V_L(s) = \frac{sLV(s)}{R + Ls + 1/Cs}$$

$$V_C(s) = \frac{V(s)}{RCs + LCs^2 + 1}$$

เมื่อทำการย้ายข้างสมการแล้วจะได้

$$\frac{V_R(s)}{V(s)} = \frac{R}{R + Ls + 1/Cs}$$

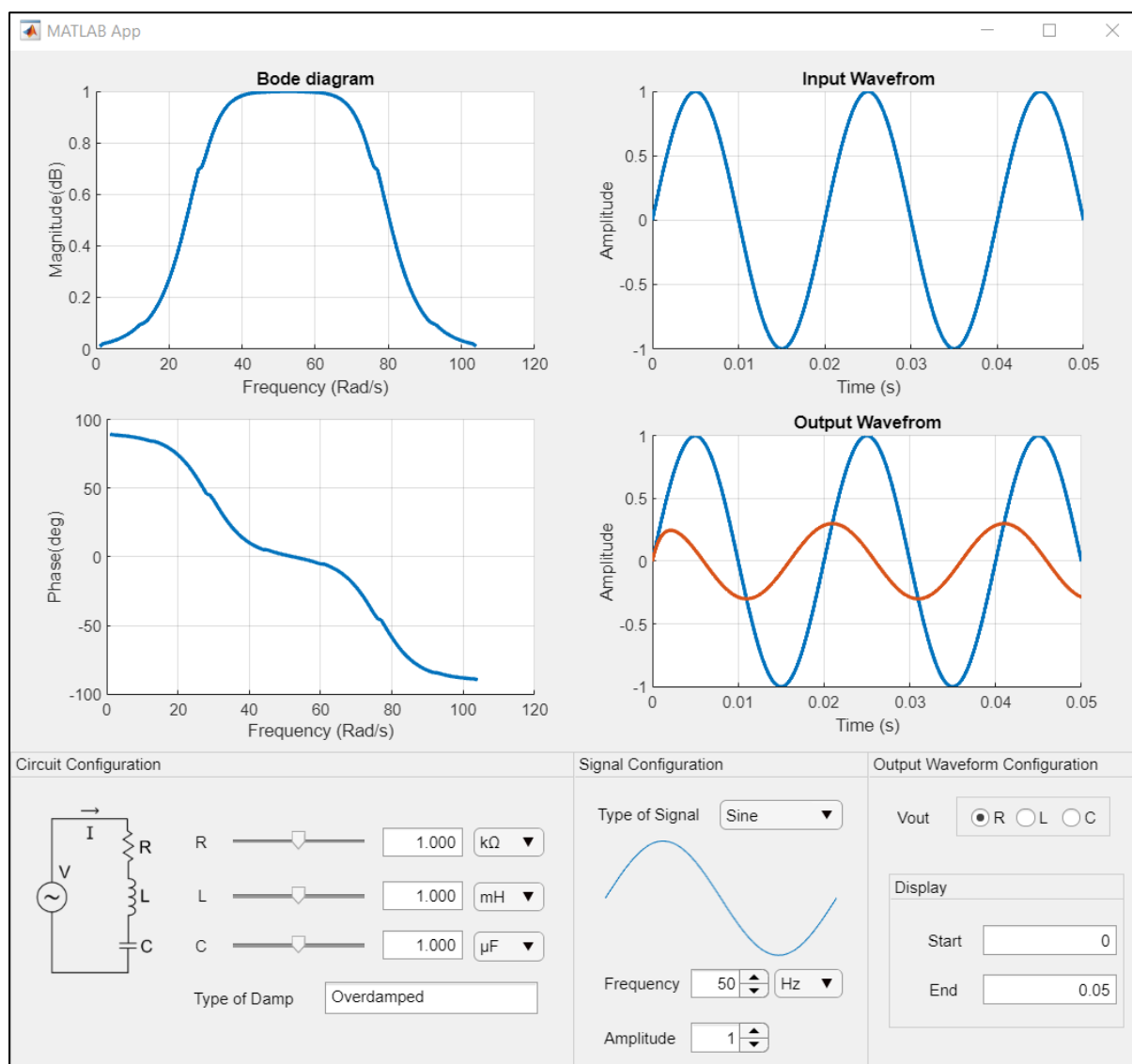
$$\frac{V_L(s)}{V(s)} = \frac{Ls}{R + Ls + 1/Cs}$$

$$\frac{V_C(s)}{V(s)} = \frac{1}{RCs + LCs^2 + 1}$$

ซึ่งเป็นสมการที่อยู่ในรูปของ V_{out}/V_{in} หรือ Transfer Function

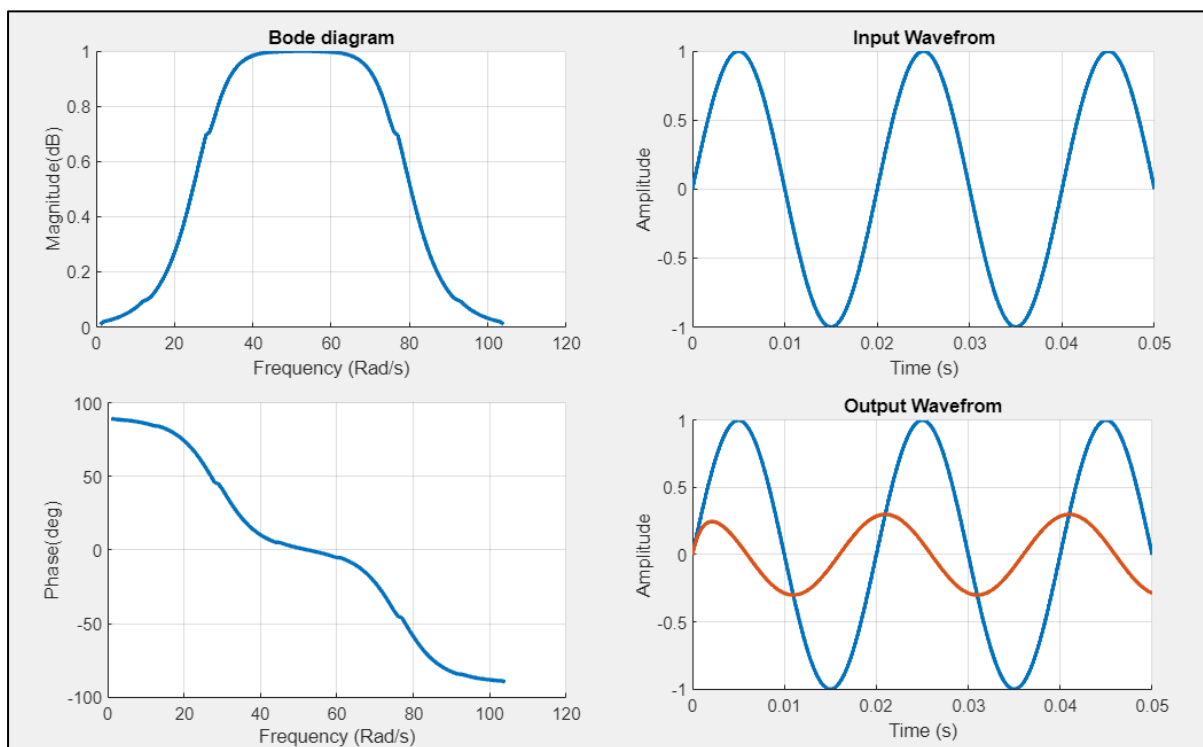
สิ่งที่ได้ดำเนินการไปแล้ว:

- GUI ของ application จาก MATLAB App Design



ใน GUI จะประกอบไปด้วยส่วนต่างๆดังนี้

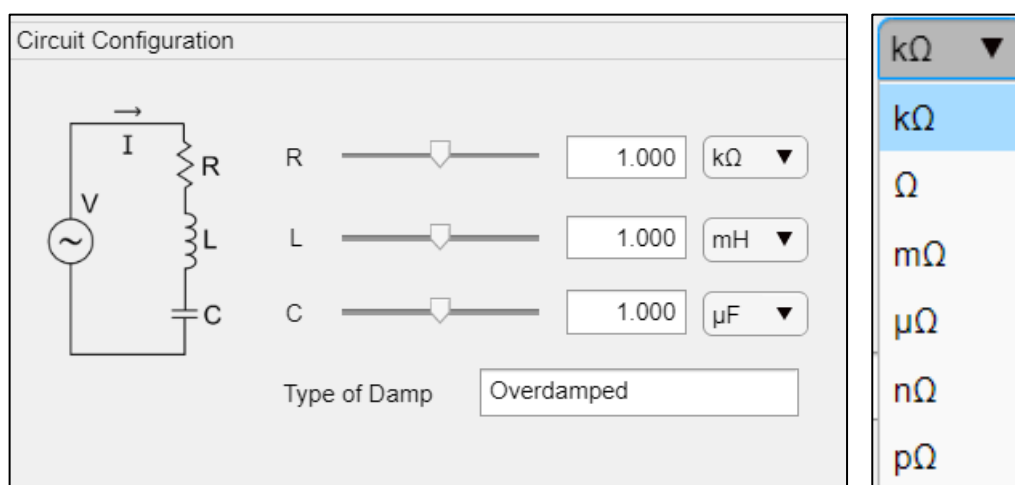
1. ส่วนแสดงผลของกราฟต่างๆ



2. ส่วนของ panel ตั้งค่า

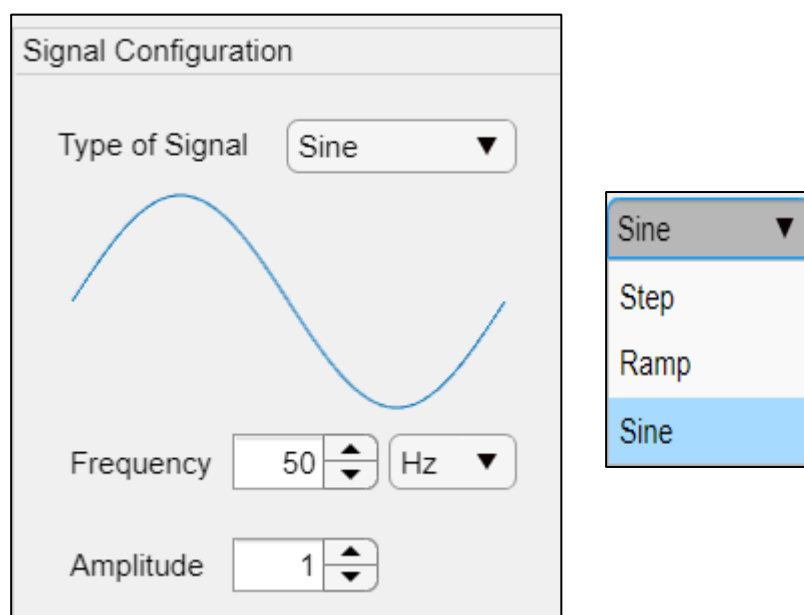
2.1 ส่วนของการตั้งค่า R, L และ C ของตัว Circuit ประกอบไปด้วย

- รูปตัวอย่างของ circuit
- Slide-bar และ Field ใ้รับค่าของ R, L และ C ที่กำหนด
- Dropdown ใช้เลือกขนาดของหน่วย Ω , H และ F
- Field แสดงประเภทของ Damp



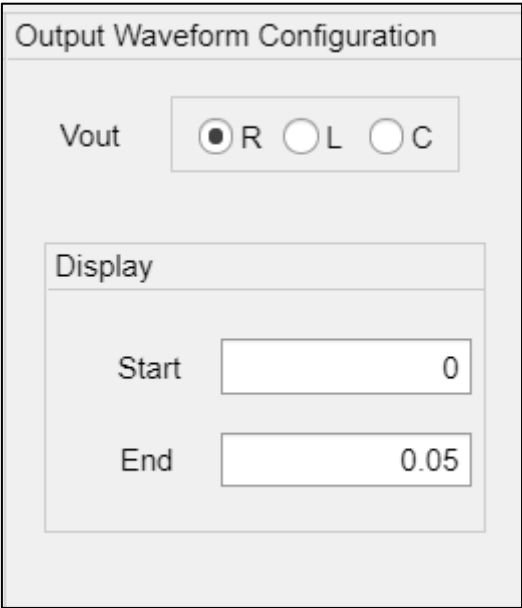
2.2 ส่วนของการตั้งค่า input signal

- Dropdown ใช้เลือกรูปแบบของสัญญาณ input
- รูปตัวอย่างของสัญญาณ input ที่ได้เลือกไว้
- Spinner สำหรับเพิ่มหรือลด Frequency
- Dropdown ใช้เลือกขนาดของหน่วย Hz
- Spinner สำหรับเพิ่มหรือลด Amplitude



2.3 ส่วนของการตั้งค่า output waveform

- Button Group สำหรับเลือกจะใช้แรงดันตกคร่อมที่ R, L หรือ C เป็น output
- Display Panel สำหรับตั้งค่าช่วงเวลาที่จะแสดงรูปของกราฟออกมา



The image shows a software dialog box titled "Output Waveform Configuration". It contains two main sections. The first section, labeled "Vout", has three radio buttons for selecting the output variable: "R" (which is selected), "L", and "C". The second section, labeled "Display", contains two input fields: "Start" with the value "0" and "End" with the value "0.05".

Section	Parameter	Value
Vout	R	Selected
	L	Not Selected
	C	Not Selected
Display	Start	0
	End	0.05

หลักการทำงาน

การทำงานของโปรแกรมนี้คือเมื่อเราเริ่มตั้งค่าต่างๆ ใน panel โปรแกรมจะทำการคำนวณแล้วแสดงกราฟออกมาตามค่าที่เราได้ตั้งเอาไว้ทันที โดยฟังก์ชันหลักๆที่จะใช้มีอยู่ 3 ฟังก์ชันคือ ฟังก์ชันที่ใช้หาประเภทของ damp, ฟังก์ชันที่ใช้ในการแสดง Frequency Response และฟังก์ชันที่ใช้แสดงผลตอบสนองต่อสัญญาณ input ซึ่ง 3 ฟังก์ชันนี้จะทำงานทุกครั้งที่มีการเปลี่ยนแปลงเกิดขึ้นใน panel ตั้งค่า

ตัวอย่างโค้ด

- โค้ดที่ใช้ในการหาประเภทของ Damp

```
methods (Access = public)
  function results = finddamp(app)
    rx = app.r*app.ru ;
    lx = app.l*app.lu ;
    cx = app.c*app.cu ;

    z=(rx/2)*(sqrt(cx/lx));
    if z>1
      app.TypeOfDamp.Value = "Overdamped" ;
    elseif z==1
      app.TypeOfDamp.Value = "Critically damped" ;
    elseif z<1
      app.TypeOfDamp.Value = "Underdamped" ;
    end
  end
end
```

โค้ดนี้จะใช้ในการหาประเภทของ damp โดยจะใช้ค่าของ R, L และ C ที่เรากำหนดเอาไว้ นำมาคูณกับขนาดของหน่วยแล้วนำมาคำนวณหาค่า Z เพื่อนำไปเทียบกับ 1 โดยอ้างอิงจาก

Circuit Type	Series RLC	Parallel RLC
Damping Factor	$\zeta = \frac{R}{2} \sqrt{\frac{C}{L}}$	$\zeta = \frac{1}{2R} \sqrt{\frac{L}{C}}$
Resonance Frequency	$\omega_o = \frac{1}{\sqrt{LC}}$	$\omega_o = \frac{1}{\sqrt{LC}}$

ที่มา https://en.wikibooks.org/wiki/Circuit_Theory/RLC_Circuits

The damping ratio is a system parameter, denoted by ζ (zeta), that can vary from **undamped** ($\zeta = 0$), **underdamped** ($\zeta < 1$) through **critically damped** ($\zeta = 1$) to **overdamped** ($\zeta > 1$).

ที่มา <https://en.wikipedia.org/wiki/Damping>

- โค้ดที่ใช้ในการแสดงกราฟของ Frequency Response

```

methods (Access = public)
    function results = cal_tranfer(app)
        rx = app.r*app.ru ;
        lx = app.l*app.lu ;
        cx = app.c*app.cu ;

        if app.vout == "R"
            G = tf([rx 0],[lx rx 1/cx]);
        elseif app.vout == "L"
            G = tf([lx 0 0],[lx rx 1/cx]);
        elseif app.vout == "C"
            G = tf([1],[lx*cx rx*cx 1]);
        end
        [mag,phase] = bode(G);
        plot(app.Magnitude_Graph,squeeze(mag),'LineWidth',2);
        plot(app.Phase_Graph,squeeze(phase),'LineWidth',2);
    end
end

```

โค้ดนี้จะใช้ในการแสดงกราฟของ Frequency Response โดยจะนำค่าของ R, L และ C ที่เราตั้งค่าเอาไว้นำมาคูณกับขนาดของหน่วย แล้วตรวจสอบว่าเราเลือกที่จะดูแรงดันตกคร่อมของ R, L หรือ C เพื่อเลือกใช้ Transfer Function จากนั้นนำไปเป็น input ให้ฟังก์ชัน tf ของ MATLAB เพื่อใช้ในการสร้าง Frequency Response จาก Transfer Function แล้วให้แสดงเป็นกราฟออกมาโดยจะแยกเป็นกราฟ Magnitude และ Phase โดย Transfer Function นั้นได้ทำการอ้างอิงมาจาก

$$\frac{V_R(s)}{V(s)} = \frac{R}{R + Ls + 1/Cs}$$

$$\frac{V_L(s)}{V(s)} = \frac{Ls}{R + Ls + 1/Cs}$$

$$\frac{V_C(s)}{V(s)} = \frac{1}{RCs + LCs^2 + 1}$$

- โค้ดที่ใช้ในการแสดงกราฟของ output

```

methods (Access = private)

function results = cal_vout(app)
    rx = app.r*app.ru ;
    lx = app.l*app.lu ;
    cx = app.c*app.cu ;

    t=(app.st:1/(app.f*100):app.lt) ;
    unitstep = t>=0;

    if app.vout == "R"
        G = tf([rx 0],[lx rx 1/cx]);
        if app.SignalType.Value == "Step"
            vr = app.a*unitstep;
        elseif app.SignalType.Value == "Ramp"
            vr = app.a*t.*unitstep;
            %vr = app.a*sawtooth(2*pi*app.f*t);
        elseif app.SignalType.Value == "Sine"
            vr = app.a*sin((2*pi*app.f*t));
        end
        [y,t] = lsim(G,vr,t);
        plot(app.Input_wave,t,vr,'LineWidth',2);
        plot(app.Output_wave,t,vr,t,y,'LineWidth',2);

    elseif app.vout == "L"
        G = tf([lx 0],[lx rx 1/cx]);
        if app.SignalType.Value == "Step"
            vl = app.a*unitstep;
        elseif app.SignalType.Value == "Ramp"
            vl = app.a*t.*unitstep;
            %vl = app.a*sawtooth((2*pi*app.f*t));
        elseif app.SignalType.Value == "Sine"
            vl = app.a*sin((2*pi*app.f*t));
        end
        [y,t] = lsim(G,vl,t);
        plot(app.Input_wave,t,vl,'LineWidth',2);
        plot(app.Output_wave,t,vl,t,y,'LineWidth',2);

    elseif app.vout == "C"
        G = tf([1],[lx*cx rx*cx 1]);
        if app.SignalType.Value == "Step"
            vc = app.a*unitstep;
        elseif app.SignalType.Value == "Ramp"
            vc = app.a*t.*unitstep;
            %vc = app.a*sawtooth((2*pi*app.f*t));
        elseif app.SignalType.Value == "Sine"
            vc = app.a*sin((2*pi*app.f*t));
        end
        [y,t] = lsim(G,vc,t);
        plot(app.Input_wave,t,vc,'LineWidth',2);
        plot(app.Output_wave,t,vc,t,y,'LineWidth',2);
    end
end
end

```

โค้ดนี้จะใช้ในการแสดงกราฟของผลตอบสนองต่อสัญญาณ input โดยจะนำค่าของ R, L และ C ที่เราตั้งค่าเอาไว้นำมาคูณกับขนาดของหน่วย จากนั้นไปทำการเช็คที่เราเลือกที่จะดูแรงดันตกคร่อมของ R, L หรือ C เพื่อเลือกใช้ Transfer Function ของตัวนั้นๆ แล้วทำการเช็คที่สัญญาณที่ได้รับเป็นสัญญาณแบบใด เพื่อที่จะได้เลือกใช้ Vin ได้อย่างถูกต้อง จากนั้นใช้ lsim ซึ่งเป็นฟังก์ชันของ MATLAB มาช่วยในการแสดงกราฟโดยจะใช้ Transfer Function, Vin และ t ซึ่งเป็นช่วงที่เราจะดูรูปกราฟมาใช้เป็น input

ปัญหาและอุปสรรค

- ชนงานโปรเจค CPF จากวิชา Software Engineering
- ผู้จัดทำยังไม่สามารถระบุได้ว่ารูปกราฟเมื่อมีสัญญาณ input เป็นรูปแบบของ ramp ที่ออกมามีความถูกต้องหรือไม่
- สามารถขยับกราฟได้เมื่อนำเมาส์ไปคลิกค้างแล้วลาก ส่งผลให้เมื่อกราฟเปลี่ยนแปลงจะแสดงผลผิดพลาด

เอกสารอ้างอิง

<https://www.mathworks.com/help/control/ref/lti.bode.html>

<https://cookierobotics.com/010/?fbclid=IwAR1i8X52EsnoUNnklDRdgRqSg6KT7Wz4KLugsTyhQd3BbVdWlbKhBlGkO2A>

<https://en.wikipedia.org/wiki/Damping>

https://en.wikibooks.org/wiki/Circuit_Theory/RLC_Circuits

<https://www.mathworks.com/help/control/ref/lti.lsim.html>

https://www.youtube.com/watch?v=iga-YS6VbyE&list=PLDzzooPYRJ_XNBukbCkOM-TBWFfeO5Ud4I