G02_HW07

Group 02
HW 07
2019/10/29

ID	Name	Your works	Times you spend	Self score	TA
108202529	葉揚昀	What is forced oscillation, equation of forced oscillation Arduino	9hr	8	
108202009	田家瑋	Resonance Arduino	6hr	5	
108202016	張家菖	Bearing and screw	7hr	7	

Forced oscillation

1. Definition:

Forced oscillations occur when an oscillating system is driven by a periodic force that is external to the oscillating system.

2. Feature:

(1) Steady-state

Let's take a boy seated in a playground swing for example (shown in fg.1). He and the swing constitute (very nearly) a simple pendulum, which has a natural frequency v_N of oscillation. Now suppose his father stands behind the swing and periodically pushes on the oscillating boy but with a driving period T_D .

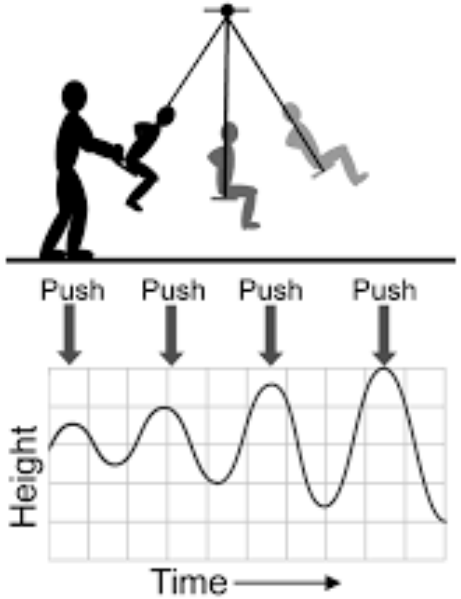
Above feature can be expressed as:

Natural frequency: $V_N = \frac{1}{T_N} = \frac{\omega_N}{2\pi}$ →(oscillator: a boy and swing)

Driving period: $T_D = \frac{1}{V_D} = \frac{2\pi}{\omega_D}$ →(driving force)

Finally, if the oscillator is moving at the frequency of the driver (whether the system is near resonance or not), the oscillations are said to have reached the steady-state condition.

Forced oscillation



fg.1 A periodic force put on swing
source:
https://www.northernhighlands.org/cms/lib/NJ01000179/Centricity/Domain/89/CPO%20Physics/PFC%20eBook%20by%20Chapter/Chapter%2019%20Harmonic%20Motion.pdf

In fg.1 demonstrates the periodic force which makes the amplitude of swing (oscillator) higher. But, as long as the oscillator is moving at the frequency of the driver, it will reach the steady-state condition.

Forced oscillation

(2) Expression of periodic force (one example In fg.2)

The driving force of forced oscillation can be shown as

$F(t) = F_0 cos(\omega_D t)$

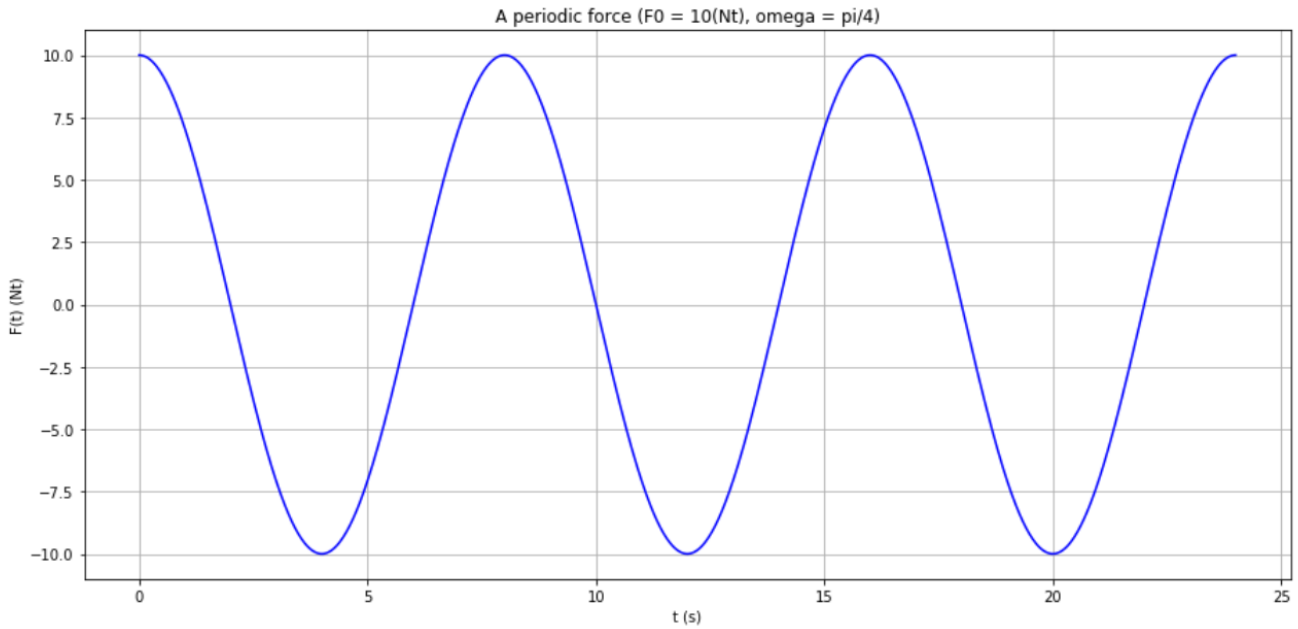
$F(t)$: function of driving force

F_0: amplitude of F(t)

ω_D: regular frequency of driving force

Accordingly, the driving force F(t) of amplitude F_0 that varies periodically with time.

F_0 (Nt)



t (s)

fg.2 The example of a periodic force (F_0 = 10(Nt), ω_D = π/4)

Forced oscillation

3. Equation of motion of Forced oscillation

(1) When the external force is applied to a damped oscillator. we can represent it as

(Ⅰ) $F(t) = F_0 cos(\omega_D t)$

(2) Thus, at this time, the forces acting on the oscillator are its restoring force, the resistance and a time-dependent driving force.

(Ⅱ) $\Sigma \mathrm{F(t)} = ma(t) = -kx(t) - bv(t) + F_0 cos\omega_D t$

restoring force　resistance　driving force

(3) Substitute $a(t) = \mathrm{d^2x/dt^2}$ and $v(t) = \mathrm{dx/dt}$ in equation (Ⅱ).

(Ⅲ) $m(\mathrm{d^2x/dt^2}) + kx + b(\mathrm{dx/dt}) = F_0 cos\omega_D t$

(4) When we apply the driving periodic force, the oscillations with natural frequency die out and the body then oscillates with the driven frequency.

(Ⅳ) $\mathrm{x(t)} = A\cos(\omega_D t + \phi)$

Forced oscillation

3. Equation of motion of Forced oscillation

(5) In function (Ⅳ) the magnitude of amplitude A can be shown as

(Ⅴ) $$A = \frac{F_0}{\sqrt{(k - m\omega_D^2)^2 + \omega_D^2 c^2}}$$

(6) Then, simultaneously divide numerator and denominator in equation (Ⅴ) by m. We get

(Ⅵ) $$A = \frac{\frac{F_0}{m}}{\sqrt{(\frac{k}{m} - \omega_D^2)^2 + \omega_D^2 (\frac{c}{m})^2}} = \frac{f_0}{\sqrt{(\omega_N^2 - \omega_D^2)^2 + \omega_D^2 (\Gamma)^2}}$$

In equation (Ⅵ), $\omega_N = \sqrt{\frac{k}{m}}$, $\Gamma = \frac{c}{m}$

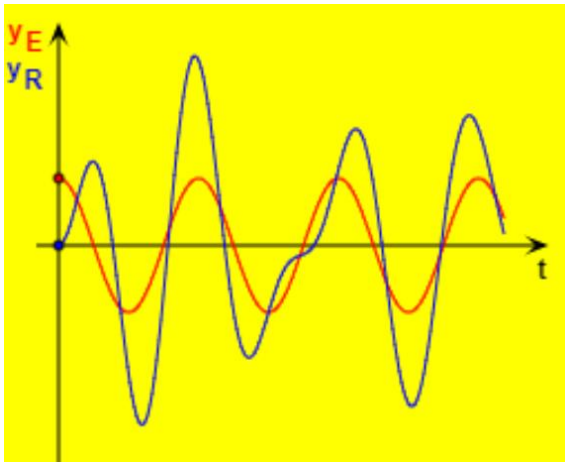
resonance

Resonance happens when the frequency of the driving force is equal to the natural frequency of the oscillation . At the moment , the amplitude of the oscillation would come to the maximum value with mere movement of the driving force.

For example , the amplitude of the spring S.H.O has the largest value when $\omega_D = \omega_N$ see next page .

resonance

Fig1: $\omega < \omega_N$
$\omega_D = 2\ rad/s$



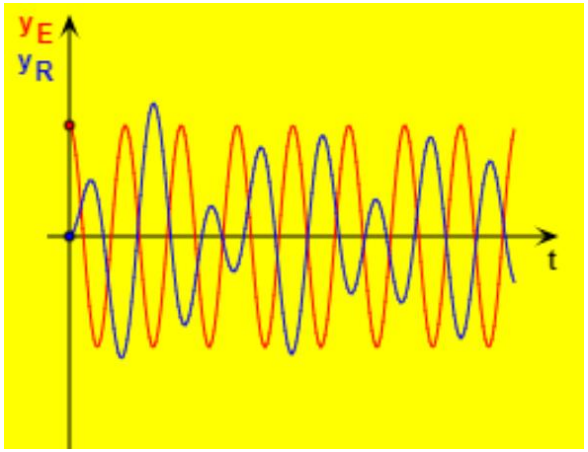
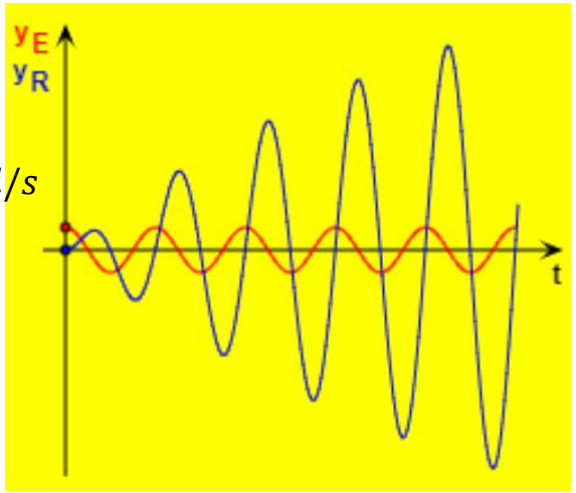


Fig2: $\omega > \omega_N$
$\omega_D = 5\ rad/s$

Fig3: $\omega = \omega_N$
$\omega_D = 3.16\ rad/s$



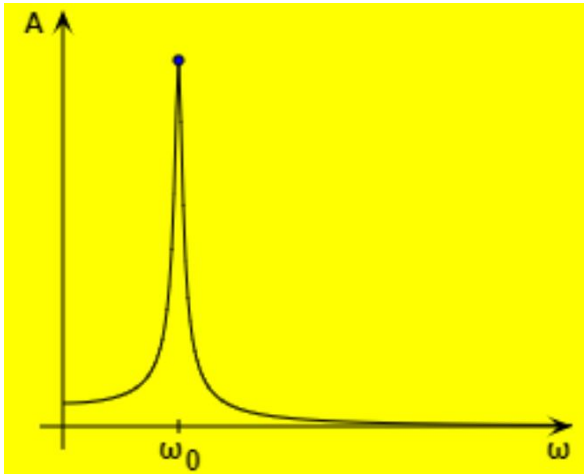


Fig4 : the maximum amplitude is arise at the $\omega = \omega_N$

$\omega_N = 3.16\ rad/s$

From(https://www.walter-fendt.de/html5/phen/resonance_en.htm)

yE : the displacement of force
yR : the displacement of object

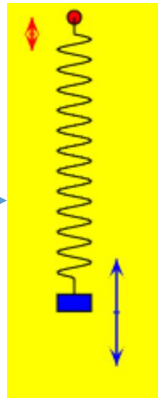


Fig 5

what is bearing

Bearing is often seen at the machine with rotating . The major benefit of using bearing is:

1.Reducing friction : The design of the structure of bearing allow machine rotating with less energy dissipation .

2.Providing fixed direction : While rotating , the machine may interfere by tiny move and therefore deviate from the origin place . Bearing would fix the rotation and prevent it .

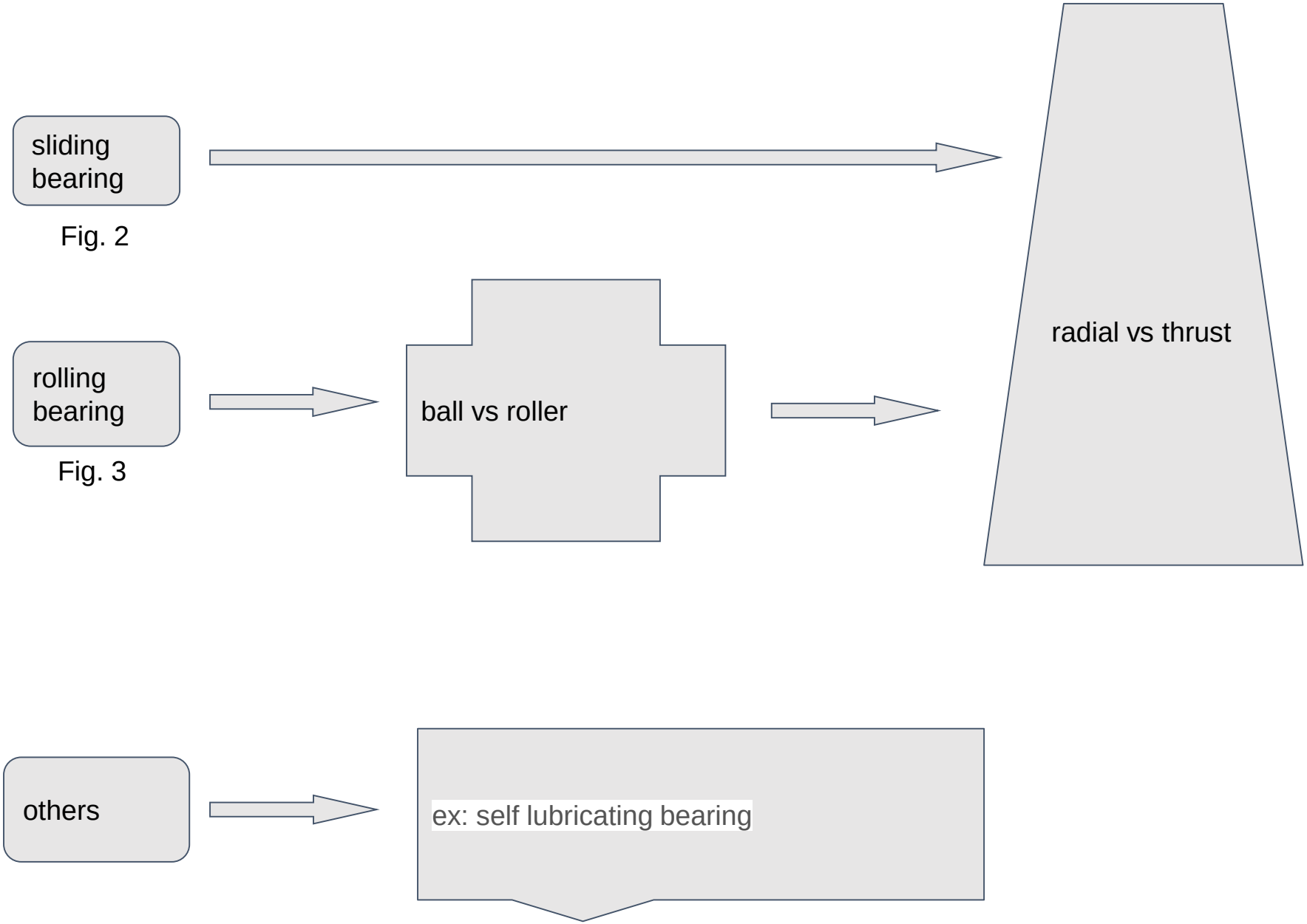

sliding bearing
rolling bearing
ball vs roller
radial vs thrust
others
ex: self lubricating bearing

Fig. 2

Fig. 3

Fig.1 sliding bearing

Fig. 2 rolling bearing

From: https://reurl.cc/VazpRN

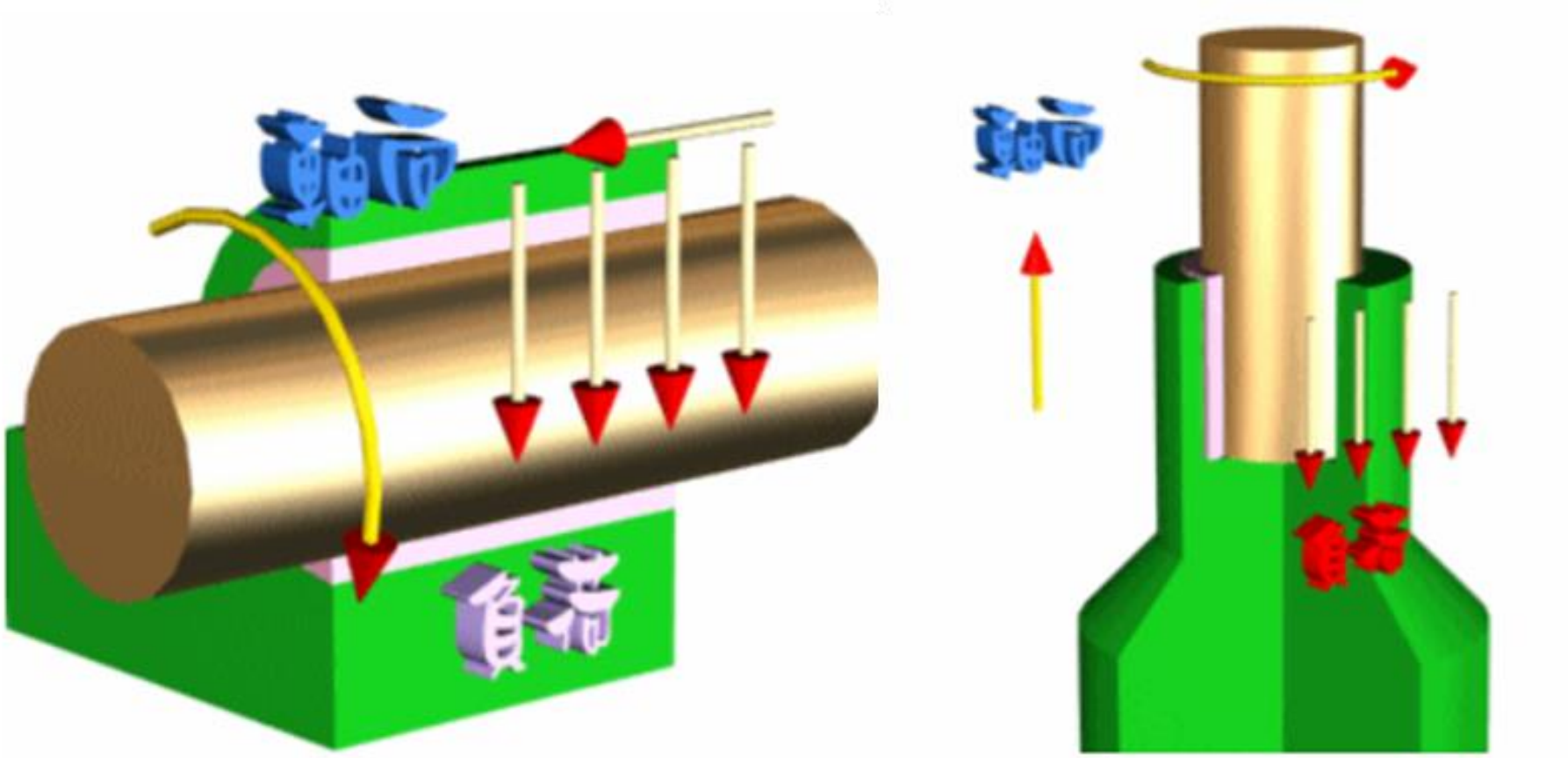


Fig. 3 Radial bearing vs thrust bearing
http://mail.scvs.ntpc.edu.tw/~gupeen/top1/chap6/htm/chap6-11.htm

Fig. 4 bearings about lathe

(For sliding bearing example)

https://reurl.cc/ObM757

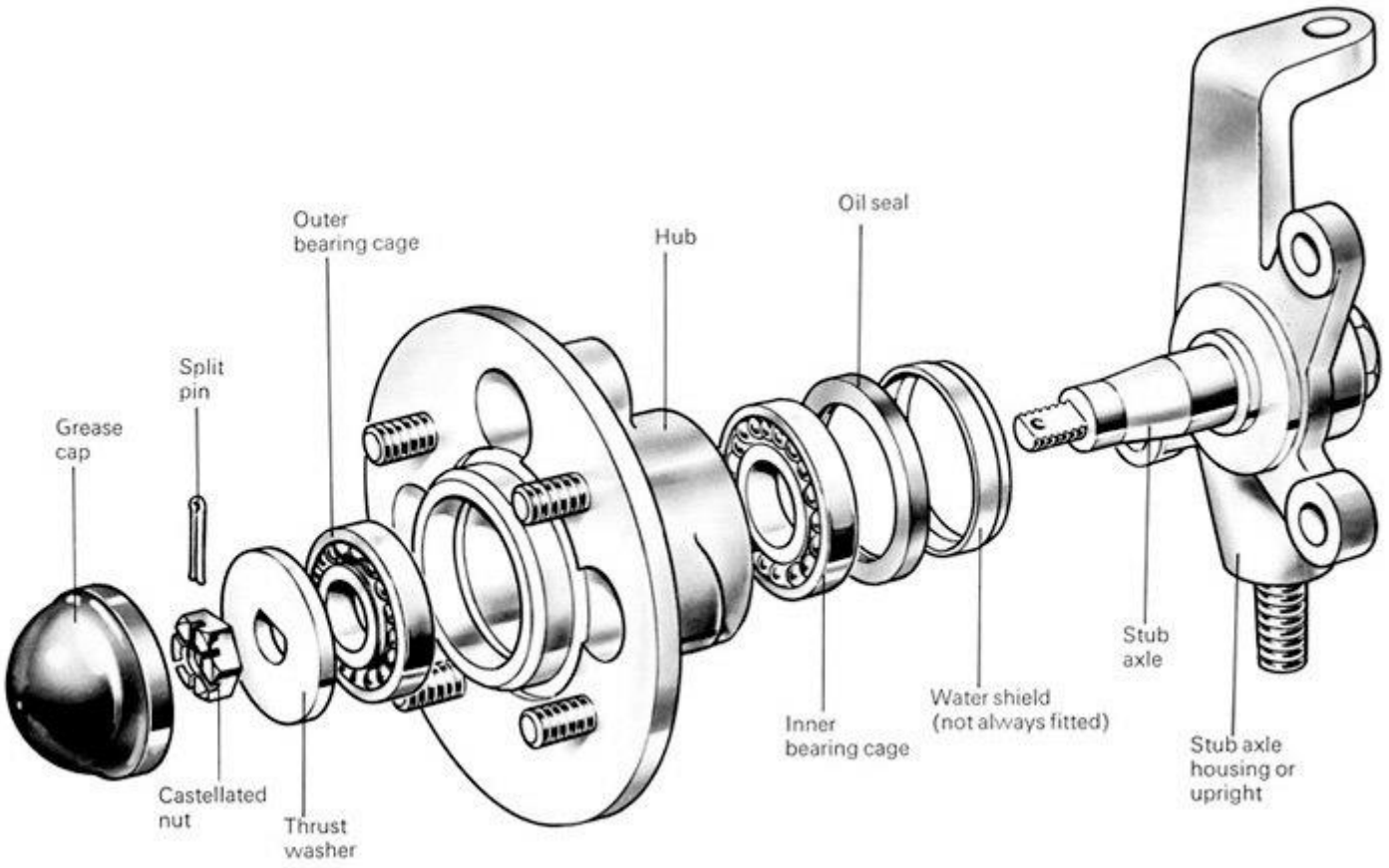


Fig. 4 bearings on car wheel
(For rolling bearing example)
https://www.carid.com/articles/wheel-bearings.html

Sliding bearing:

1. high speed
2. heavy duty
3. high precision
4. intermittent operating conditions.

Defect:

Large starting friction

Rolling bearing:

friction low

maintenance cost low

Defect:

1. High cost and expensive price.

2. The required manufacturing technology is high, and the design is difficult.

3. It is difficult to bear for impact load and heavy load.

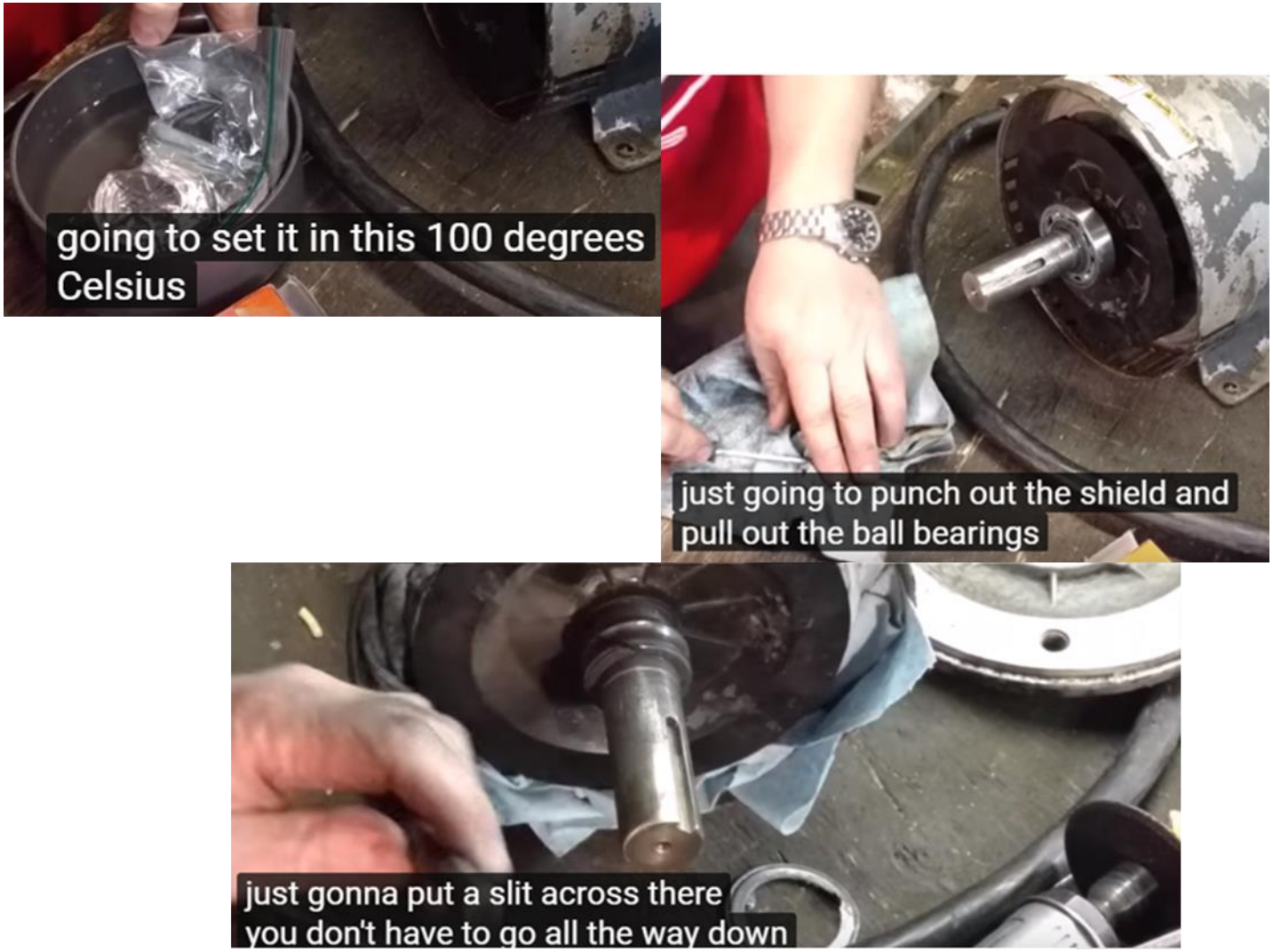


Fig. 6 install and remove bearing
https://www.youtube.com/watch?v=rtzOPCIOSDc

Inch thread marked with: W__*N__.

The number behind W is referred to as the outer diameter of the thread.

N is the thread pitch, how many threads per inch.

The metric thread is marked with: M__*P__.

The number behind M refers to the outer diameter of the thread. It means the outer diameter of the thread "mm"

P also refers to the thread pitch, but expressed as the distance of each tooth distance "mm"

(Inch thread pitch indicates how many threads per inch

The metric thread pitch indicates the distance between each tooth distance)

Drilling size = screw outer diameter - pitch

E.g:

Metric

M3 X 0.5 (outer diameter = 3mm, pitch = 0.5mm)

Drill size = 3mm - 0.5 = 2.5mm

M5 X 0.8

Drill size = 5mm - 0.8 = 4.2mm

Inch screw

1/8"-40T (outer diameter = 1/8" = 3.175mm, 40 teeth per inch)

Drill size = 3.175mm - 0.635mm = 2.54mm

(M)公制粗牙螺纹

规格	建议钻孔径	2级螺纹内径	
		上限	下限
M1.0×0.25	0.75	0.785	0.729
M1.1×0.25	0.85	0.885	0.828
M1.2×0.25	0.95	0.985	0.929
M1.4×0.3	1.10	1.142	1.075
M1.6×0.35	1.25	1.321	1.221
M1.7×0.35	1.35	1.421	1.321
M1.8×0.35	1.45	1.521	1.421
M2×0.4	1.60	1.679	1.567
M2.2×0.45	1.75	1.838	1.713
M2.3×0.4	1.90	1.979	1.867
M2.5×0.45	2.10	2.138	2.013
M2.6×0.45	2.20	2.238	2.113
M3×0.5	2.50	2.599	2.459
M3.5×0.6	2.90	3.010	2.850

Fig. 8 screw type

https://kknews.cc/collect/xvm299q.html

PF（G）英制管螺纹

规格	标准径	最大	最小
1/16–28	6.7	6.843	6.561
1/8–28	8.7	8.848	8.566
1/4–19	11.7	11.890	11.445
3/8–19	15.2	15.395	14.950
1/2–14	19.0	19.172	18.631
5/8–14	21.0	21.128	20.587
3/4–14	24.5	24.658	24.117
7/8–14	28.2	28.418	27.877
1 –11	30.6	30.931	30.291
1 1/2–11	35.2	35.579	34.939
1 1/4–11	39.2	39.592	38.952
1 1/2–11	45.0	45.485	44.845
1 3/4–11	51.0	51.428	50.788
2 –11	57.0	57.296	56.656

Fig. 9 screw type

https://kknews.cc/collect/xvm299q.html

Set screw: secure an object within or against another object, normally without using a nut. The screw head will touch other objects, so use it to reduce the length of the screw head.

Self-tapping screws: they can be tapped by themselves without drilling first.

Fig. 10 two types of screw
(From : https://reurl.cc/EKjZYn)

Arduino coding

Breathing LED & ordered lighting array

```
void loop() {
  for(int i=0;i<=2;i=i+1){          /*The outter  "for loop" make 3 LED light take turns*/
    digitalWrite(LED1,LOW);
    digitalWrite(LED2,LOW);
    digitalWrite(LED3,LOW);
    if(i%3==0){
      for(int i=0;i<256;i=i+1){    /*This inner "for loop" make LED light as breathing*/
        analogWrite(LED1,i);
        delay(5);
      }
      for(int i=255;i>=0;i=i-1){
        analogWrite(LED1,i);
        delay(5);
      }
    }
```

Arduino coding

Breathing LED & ordered lighting array

```
else if(i%3==1){
      for(int i=0;i<256;i=i+1){
      analogWrite(LED2,i);
      delay(5);
      }
      for(int i=255;i>=0;i=i-1){
        analogWrite(LED2,i);
        delay(5);
      }
    }
    else if(i%3==2){
      for(int i=0;i<256;i=i+1){
      analogWrite(LED3,i);
      delay(5);
      }
      for(int i=255;i>=0;i=i-1){
      analogWrite(LED3,i);
      delay(5);
      }
```

Arduino coding

Brightness of LED with potentiometer

```
int analogPin_A3 = 17;
int val = 0;
int LED1 = 3;
void setup()
{
  Serial.begin(9600);
  pinMode(LED1,OUTPUT);
}

void loop()
{

  val = analogRead(analogPin_A3);    /*Read the value of voltage and convert into 0~1023*/
  Serial.println(val);
  analogWrite(LED1,val/4);               /*Write the value of voltage which be divided into 0~256*/
}
```

Summary-葉揚昀 108202529

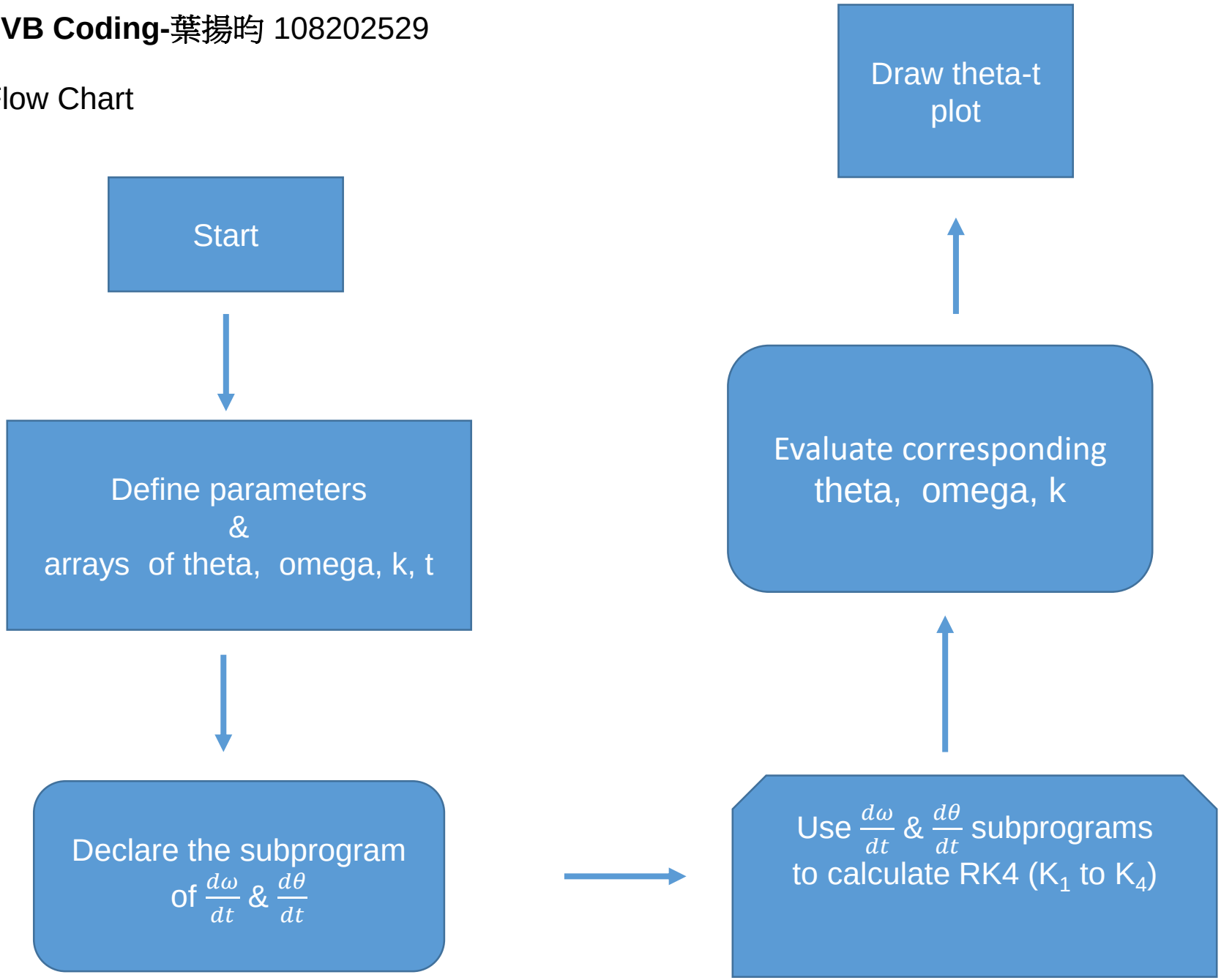
In this week, I had been learning various kinds of things through different ways. One was my groupmate and me discussing about whether CMOS has a structure of color filter or not. Due to all of us hadn't researched any information that had mentioned about CMOS's color filter, we made two assumptions which pixels of CMOS have the ability for converting electron packet into specific voltage through inducing electron packet kinetic energy, or CMOS actually has a color filter but this information hadn't talked about it.

Another, I had more clear conceptions of numerical aperture, PWM, and duty cycle form each groups' presentation. Let's take numerical aperture and PWM for example. In numerical aperture, to simplify, if we increase numerical aperture, the resolution will be higher. (The form of numerical is NA = nsin θ or focal length of the lens / diameter of the lens aperture). In PWM, as a result of duty cycle can be shown as $duty\ cycle = \frac{t_{HIGH}}{t_{period}} \cdot 100\%$, it having the capability of simulating analog signal by using duty cycle of digital form.

The other was Arduino's coding and circuit of breadboard to motherboard. Based on the reason of one of my groupmate quite value our groupmate whether realize Arduino or not, he assigned this week's Arduino work to me. Fortunately, although I hadn't learned Arduino before, he would teach some basic program syntax and how to connect the wires in correct position. Eventually, I used for-loop, analogWrite, and analogRead to make breathing LED and convert the analog signal of variable resistor into an output signal for LED.

VB Coding-葉揚昀 108202529

Flow Chart



VB Coding-葉揚昀 108202529

Define parameters & arrays of theta, omega, k, t

```
 Dim L As Single = CSng(L_text.Text)
        Dim dt As Single = CSng(time_step_text.Text)
        Dim t(100000.0), theta(100000.0), omega(100000.0) As Single
        Dim k1_theta(100000.0), k2_theta(100000.0), k3_theta(100000.0), k4_theta(100000.0)
As Single
        Dim k1_omega(100000.0), k2_omega(100000.0), k3_omega(100000.0), k4_omega(100000.0)
As Single
        t(0) = 0
        theta(0) = CSng(initial_theta_text.Text)
        omega(0) = CSng(initial_omega_text.Text)
```

VB Coding-葉揚昀 108202529

Declare the subprogram of $\frac{d\omega}{dt}$ & $\frac{d\theta}{dt}$

```
    Private Sub Cal_omega_dot(ByVal Cal_omega As Single, ByVal Cal_theta As Single, ByRef
Cal_omega_dot As Single)
        Dim g As Single = 9.8
        Dim L As Single = CSng(L_text.Text)
        Dim c As Single = CSng(c_text.Text)
        Cal_omega_dot = -g / L * Sin(Cal_theta) - c * Cal_omega
    End Sub

    Private Sub Cal_theta_dot(ByVal Cal_omega As Single, ByRef Cal_theta_dot As Single)
        Cal_theta_dot = Cal_omega
    End Sub
```

VB Coding-葉揚昀 108202529

Use $\frac{d\omega}{dt}$ & $\frac{d\theta}{dt}$ subprograms to calculate RK4 (K_1 to K_4)

```
For i As Integer = 1 To 50000 Step 1
        Cal_omega_dot(omega(i - 1), theta(i - 1), k1_omega(i - 1))
        Cal_theta_dot(omega(i - 1), k1_theta(i - 1))
        Cal_omega_dot(omega(i - 1) + dt / 2 * k1_omega(i - 1), theta(i - 1) + dt / 2
* k1_theta(i - 1), k2_omega(i - 1))
        Cal_theta_dot(omega(i - 1) + dt / 2 * k1_omega(i - 1), k2_theta(i - 1))
        Cal_omega_dot(omega(i - 1) + dt / 2 * k2_omega(i - 1), theta(i - 1) + dt / 2
* k2_theta(i - 1), k3_omega(i - 1))
        Cal_theta_dot(omega(i - 1) + dt / 2 * k2_omega(i - 1), k3_theta(i - 1))
        Cal_omega_dot(omega(i - 1) + dt * k3_omega(i - 1), theta(i - 1) + dt *
k3_theta(i - 1), k4_omega(i - 1))
        Cal_theta_dot(omega(i - 1) + dt * k3_omega(i - 1), k4_theta(i - 1))
```

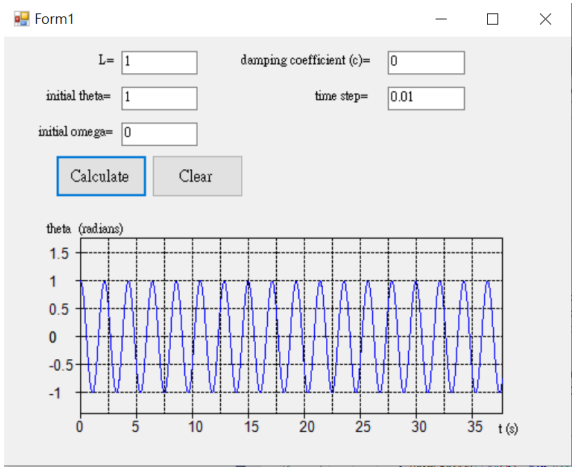
VB Coding-葉揚昀 108202529

Evaluate corresponding theta, omega, k & draw theta-t plot

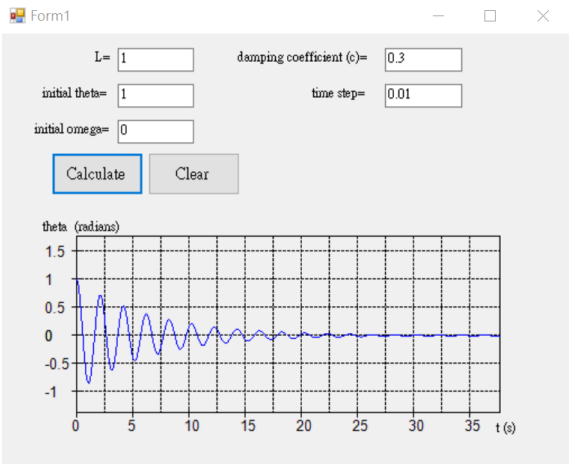
```
            omega(i) = omega(i - 1) + 1 / 6 * dt * (k1_omega(i - 1) + 2 * k2_omega(i - 1)
+ 2 * k3_omega(i - 1) + k4_omega(i - 1))
            theta(i) = theta(i - 1) + 1 / 6 * dt * (k1_theta(i - 1) + 2 * k2_theta(i - 1)
+ 2 * k3_theta(i - 1) + k4_theta(i - 1))
            t(i) = t(i - 1) + dt


            a.DrawLine(p, 10 * t(i - 1) + 30, -50 * theta(i - 1) + ph_half, 10 * t(i) +
30, -50 * theta(i) + ph_half)
            If 10 * t(i) > (pw - 50) Then Exit For
        Next
    End Sub
```

VB Coding-葉揚昀 108202529

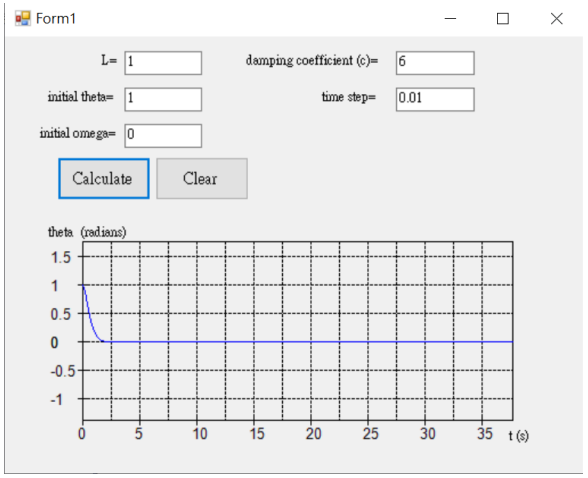
Result (L=1(m) , $\theta_0 = 1(radian)$, $\omega_0 = 0(1/s)$, dt = 0.01(s))



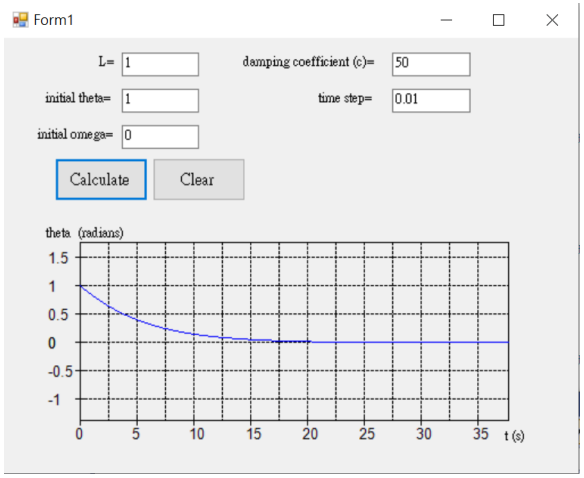
Fg.1 harmonic oscillation (c=0)



Fg.2 under damping (c=0.3)



Fg.3 critical damping (c=6)



Fg.4 over damping (c=50)

Plan of machining project-葉揚昀 108202529

I have done the cylinder last week and milled six surfaces of the square cylinder this week.

In next week, I plan to:

1. Mill the square cylinder to 57 mm.
2. Use a drill bit to drill a 6.9 mm hole and use a pipe tap to tap in square cylinder (M8 x 1.25 tapped).

If I have extra time I will:

1. Make one groove and two hole with countersunk drill and milling cutter in the workpiece.
2. Chamfer the square cylinder .

summary

This week I want to talk some of question about CMOS and CCD stunning in my head . “ Why CMOS has more noise than CCD “ , and “ How to calculate pixels “ .

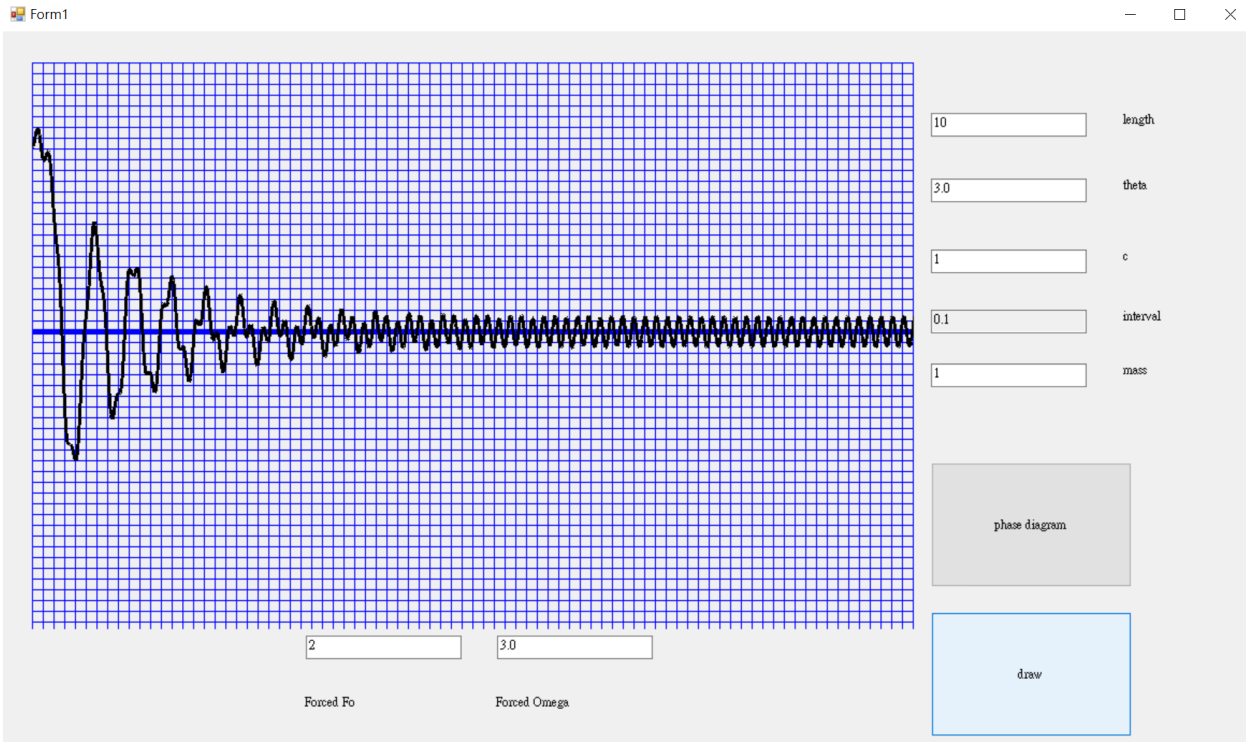
The first question raise from a website . It says that “ CMOS has less noise than CCD “ (I forget the reason) when most of the document I have ever seen shows that CCD cause less noise . And there is another team mention the former on the representation on the class . It’s my fault I didn’t ask them that .

The second question hit my head bitterly . Since there is color filter on CCD before the photons hit the photodiode , they use 4 pixels to describe 1 pixel of the real image . the question is : Which pixel is the pixel we are talking . the filtered-color pixel? Or the combined pixel ? I couldn’t find the answer on the website . For the conclusion : There are lots of information we can’t find online . We are hard to determine which is true .

At the end , I have some ideas about the professor ask today . we were talking about the structure and we mentioned that CMOS may use the equation $E = hf$ and measure the kinetic energy of the electronic and transform into voltage . Professor says “ NO WAY! “ at that time . My thought is that the voltage shouldn’t be a smooth wave due to different Kinetic energy of electronic . And therefore , we can use Transformation of Fourier to analyze the energy of different kind wave and combine into a real pixel of the image . I just give my weak point , never mind .

Vb_practice

I‘ve finish RK4 last week , so I add periodic force to drive the oscillation . Notice , that my force provide from the gravity is using $Sin(\theta)$ instead of θ . So the consequence of the graph may be different from the expactation .



Vb_practice

- For t As Single = dt To times Step dt
- 'theta_0o is point about to puting into the K1'
- theta_0o = theta_0(CInt(t / dt) - 1)
- theta_1o = theta_1(CInt(t / dt) - 1)
- 'theta_0I(0) is K1 , theta1I(0) is L1'
- theta_0I(0) = theta_1o
- theta_1I(0) = -g / l * Sin(theta_0o) - c / (M * l) * theta_1o + F0 * Cos(Ome * t)

- 'theta_0o is point about to puting into the K2'
- theta_0o = theta_0(CInt(t / dt) - 1) + dt / 2 * theta_0I(0)
- theta_1o = theta_1(CInt(t / dt) - 1) + dt / 2 * theta_1I(0)
- 'theta_0I(1) is K2 , theta1I(1) is L2'
- theta_0I(1) = theta_1o
- theta_1I(1) = -g / l * Sin(theta_0o) + -c / M / l * theta_1o + F0 * Cos(Ome * t)

- 'theta_0o is point about to puting into the K3'
- theta_0o = theta_0(CInt(t / dt) - 1) + dt / 2 * theta_0I(1)
- theta_1o = theta_1(CInt(t / dt) - 1) + dt / 2 * theta_1I(1)

Vb_practice

- 'theta_0I(2) is K3 , theta1I(2) is L3'
- theta_0I(2) = theta_1o
- theta_1I(2) = -g / l * Sin(theta_0o) + -c / M / l * theta_1o + F0 * Cos(Ome * t)

- 'theta_0o is point about to puting into the K4'
- theta_0o = theta_0(CInt(t / dt) - 1) + dt * theta_0I(2)
- theta_1o = theta_1(CInt(t / dt) - 1) + dt * theta_1I(2)
- 'theta_0I(3) is K4 , theta1I(3) is L4'
- theta_0I(3) = theta_1o
- theta_1I(3) = -g / l * Sin(theta_0o) + -c / M / l * theta_1o + F0 * Cos(Ome * t)

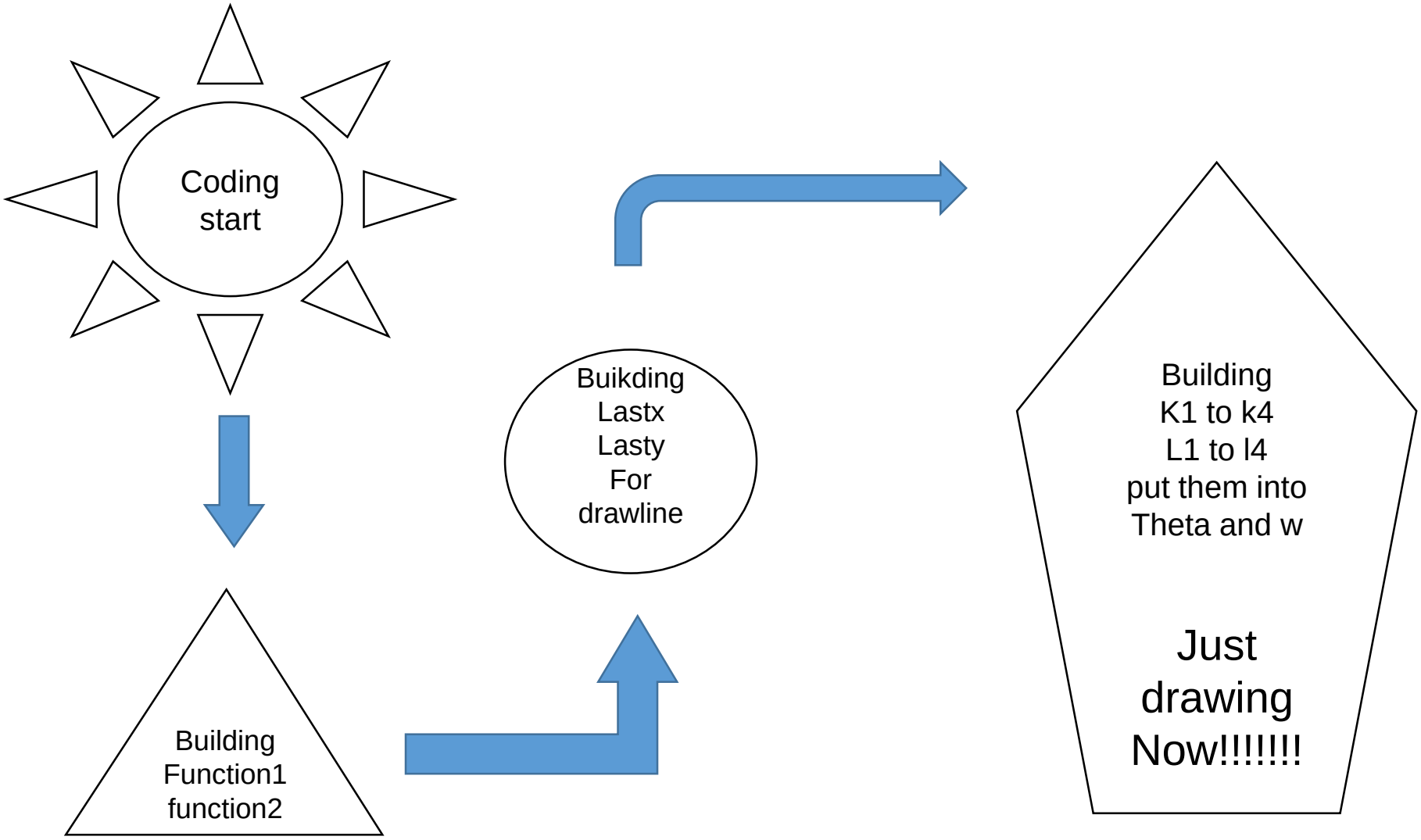
- 'Calculate the average K and L , and then output the nex point'
- theta_0(CInt(t / dt)) = theta_0(CInt(t / dt) - 1) + dt / 6 * (theta_0I(0) + 2 * theta_0I(1) + 2 * theta_0I(2) + theta_0I(3))
- theta_1(CInt(t / dt)) = theta_1(CInt(t / dt) - 1) + dt / 6 * (theta_1I(0) + 2 * theta_1I(1) + 2 * theta_1I(2) + theta_1I(3))

Machining project

Next week , I am going to finish milling the cylinder by lathe .

summary_張家菖

This week, I learned five issues. First, I knew the three important factors of photographing, including shutter speed, the ISO, and the numerical aperture. We can take shutter speed for a length of time. And the ISO means the sensitivity of light. The numerical aperture means the opening extent of an aperture. I also understood two kinds of semiconductors, called a CCD and a CMOS. We compared them with costs, principles, complexions, power consumptions, and so on. Third, I learned the PWM and a duty cycle. We can use a special way to calculate a wave, in which we can transform a digital wave into an analog wave. We can use the duty cycle to simulate a numerical wave into an analog wave by the ratio of one to zero. Forth, I knew more types of coding of Arduino. In order to make the light change continuously and learn to use programs to change the voltage, I tried to use the function, called "analog write". And I also tried to use the function of arrays. All of these made me surprise because I had learned the c++ before. It made me miss the past time. Fifth, I started to cut my cuboid. I tried to cut the cuboid general structure. And I found that it is hard to frame coordinate system. Because it is my second opportunity to cut a cuboid, I scared that I will fail again. With fear, I carefully processed the cuboid.


Coding
start
Buikding
Lastx
Lasty
For
drawline
Building
K1 to k4
L1 to l4
put them into
Theta and w
Just
drawing
Now!!!!!!!
Building
Function1
function2

```
Public Function f1(ByRef theta1 As Decimal, ByRef w1 As Decimal, ByRef time1 As Decimal)
        Dim l As Single = Val(TextBox1.Text)          ‘長度
        Dim b As Single = Val(TextBox3.Text)          ‘阻尼
        Dim theta As Single = Val(TextBox2.Text)      ‘角度(弧度)
        Return w1
    End Function

Public Function f2(ByRef theta2 As Decimal, ByRef w2 As Decimal, ByRef time2 As Decimal)
        Dim l As Single = Val(TextBox1.Text)          ‘長度
        Dim b As Single = Val(TextBox3.Text)          ‘阻尼
        Dim theta As Single = Val(TextBox2.Text)      ‘角度(弧度)
        Return -9.8 / l * Sin(theta2) - b * w2
    End Function
```

```
For t = 1 To 300 Step 1
        lasty = theta
        lastx = t
        k1 = f1(theta, w, t) * h
        l1 = f2(theta, w, t) * h
        k2 = f1(theta + 0.5 * k1, w + 0.5 * l1, t + 0.5) * h
        l2 = f2(theta + 0.5 * k1, w + 0.5 * l1, t + 0.5) * h
        k3 = f1(theta + 0.5 * k2, w + 0.5 * l2, t + 0.5) * h
        l3 = f2(theta + 0.5 * k2, w + 0.5 * l2, t + 0.5) * h
        k4 = f1(theta + k3, w + l3, t + h) * h
        l4 = f2(theta + k3, w + l3, t + h) * h
        theta = theta + (k1 + 2 * (k2 + k3) + k4) / 6
        w = w + (l1 + 2 * (l2 + l3) + l4) / 6
        g2.DrawLine(penn, lastx, 5 * lasty, t, 5 * theta)
    Next t
```