The Robotics Kinematics

MÕJÂĤED NÒ.ÓR at 7/8/2020 11:46 PM

- We need to declare the variables that we will use and include the libaries

```
#include <Keypad.h>
int LI=40,L2=45,CHECK;
float ANG1, ANG2 ,ANGL1, ANGL2,X,Y,ALLANG=50,INVANG;

#include <Servo.h>
Servo motor1 ;
Servo motor2 ;



char keys[4][3]=
{{'1','2','3'},{'4','5','6'},{'7','8','9'},{' ','9',' '}};

byte rows [4]={A2 ,A3 , A4, A5};
byte cols [3]={8,9,10};
Keypad userINPUTS = Keypad(makeKeymap(keys),rows,cols,4,3);


#include <LiquidCrystal.h>
// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
 motor1.attach(7);
 motor2.attach(6);

 lcd.begin(16, 2);
 lcd.print("hello, world!");
}

void loop() {
 // set the cursor to column 0, line 1
 // (note: line 1 is the second row, since counting begins with 0):
 lcd.setCursor(0, 1);
 // print the number of seconds since reset:
 lcd.print(millis() / 1000);
 lcd.clear(); //clear word on led
  CHECK=forwaORinver();



}
```

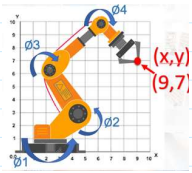
```
 if (CHECK==1) {
   int ANGL11,ANGL12,ANGL1,
   ANGL21,ANGL22,ANGL2;

lcd.clear();
lcd.print ("Enter anlgle 1");
while (ANGL11==0) {ANGL11=userINPUTS.getKey();} //wait user to
enter first number of Q1
lcd.setCursor (0,1);
ANGL11=ANGL1-48; // CONVERT THE Number From ASCII
lcd.print(ANGL11);
while (ANGL12==0) {ANGL12=userINPUTS.getKey();} //wait user to
enter second number of Q1
ANGL12=ANGL12-48;
lcd. print (ANGL12);
delay (500);
ANGL1=ANGL11*10+ANGL12;
lcd.clear();
lcd.print ("Enter anlgle 2");
while (ANGL21==0)
{ANGL21=userINPUTS.getKey();} //wait user to enter first number
of Q1
lcd.setCursor (0,1);
ANGL21=ANGL21-48; // CONVERT THE Number From ASCII
lcd.print(ANGL21);
while (ANGL22==0) {ANGL22=userINPUTS.getKey();} //wait user to
enter second number of Q1
ANGL22=ANGL22-48;
lcd. print (ANGL22);
delay (500);
ANGL2=ANGL21*10+ANGL22;
forward(); // CallING forward function
}
 else
 {  int X11=0,X12=0,X
   ,Y11=0,Y12=0,Y,ul;
```

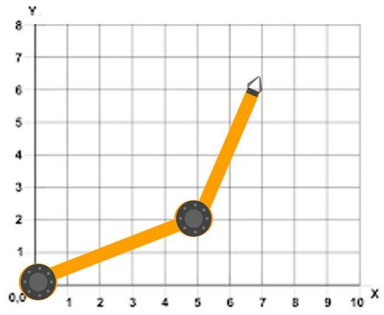
```
lcd.clear();
lcd.print ("Enter value of X. postion");
while (X11==0)
{X11=userINPUTS.getKey();}
X11=X11-48;
lcd.setCursor (0,1);
lcd. print (X11);
while (X12==0)
{X12=userINPUTS.getKey();}
X12=X12-48;
lcd. print (X12);
delay (500);
X=X11*10+X12;
lcd.clear();
lcd.print ("Enter value of Y");
while (Y11==0)
{Y11=userINPUTS.getKey();}
Y11=Y11-48;
lcd.setCursor (0,1);
lcd. print (Y11);
while (Y12==0)
{Y12=userINPUTS.getKey();}
Y12=Y12-48;
lcd. print (Y12);
delay (500);
Y=Y11*10+Y12;
invers(); //call invers function
```

#We have two types of-

1#/ Forward Kinematics :-



- knowns
  - the angles (ϕ_1, ϕ_2)
  - the tall of the arm. (L_1, L_2)
- nknows
  - the end effector postion. (x, y)
  - the angle of the end effector. (θ).



$$\cos\phi_1 = \frac{x_1}{L_1}$$

$$x_1 = \cos\phi_1 \cdot L_1$$

$$y_1 = \sin\phi_1 \cdot L_1$$

$$x_2 = (\cos\phi_1 + \phi_2) \cdot L_2$$

$$y_2 = \sin(\phi_1 + \phi_2) \cdot L_2$$

Lastly, we want to find the end effector postion (x, y)

$$x = x_1 + x_2 = L_1\cos\phi_1 + L_2\cos(\phi_1 + \phi_2)$$

$$y = y_1 + y_2 = L_1\sin\phi_2 + L_2\sin(\phi_1 + \phi_2)$$

θ between x, y is $\theta = \phi_1 + \phi_2$

We need to change these equations to code, as follow:

```
int forwaORinver()
{{
char IND=0;
motol.write (0);
moto2.write (0);
lcd.clear();
lcd.print ("Press 1 FOR forward ");
lcd. setCursor (0,1);
```

```
lcd.print ("or 2 FOR Inverse");
while (IND==0)
{ IND = userINPUTS.getKey();}
IND=IND-48;
if (IND==1) {
lcd.clear();
lcd.print ("Forward Kinematic");
lcd. setCursor (0,1);
lcd.print (1);
delay (2000);
return 1;}
else if (IND==2) {
lcd.clear();
lcd.print ("Inverse Kinemitic");
lcd. setCursor (0,1);
lcd.print (2);
delay (2000);
return 2;}
}

void forward () {
//this function to calculate Q1 &Q2
motol.write (ANGL1);
moto2.write (ANGL2);
lcd.clear();
ANG1=ANGL1*PI/180;//Degree to radian
ANG2=ANGL2*PI/180;: //Degree to radian
X=(L1*cos (ANG1) ) + (L2* cos (ANG1+ANG2));
Y=(L1*sin(ANG1))+(L2*sin(ANG1+ANG2));
lcd.print ("The postion of the end effector");
lcd.setCurser(5, 1);
lcd.print(":");
lcd.setCurser(0, 1);
lcd. print (X);
lcd.setCurser(6, 1);
lcd.print (Y);
Serial.print ("ANG1");
Serial.println(ql);
Serial.print ("ANG2");
Serial.println(q2);
delay (7000);

}
```

2/ inverse kinematics :-

Knowns
- the postion of end effector and angle.
- the length of the arm

unKnowns
- the angle of the first Joint
- the angle of the seand Joint.

Triangle (1)

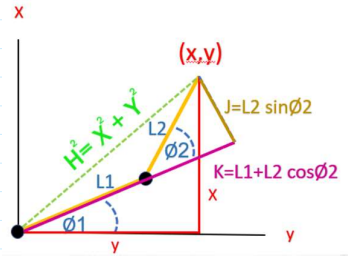
$H^2 = X^2 + Y^2$

Triangle (2)

$J = L2 \sin \phi 2$

$K = L1 + L2 \cos \phi 2$

$H^2 = J^2 + K^2$



$\therefore x^2 + y^2 = J^2 + K^2$

$= (L2 \sin \phi 2)^2 + (L1 + L2 \cos \phi 2)^2$

$= L2^2 \sin \phi 2^2 + L1^2 + L2^2 \cos \phi 2^2 + 2L1L2 \cos \phi 2$

$= L2^2 (\sin^2 \phi 2 + \cos^2 \phi 2) + L1^2 + 2L1L2 \cos \phi 2$

$x^2 + y^2 = L2^2 (1) + L1^2 + 2L1L2 \cos \phi 2$

Lastly, To find the desired angle.

$r^2 = L1^2 + L2^2 - 2L1L2\cos\phi 2$

Lastly, To find the desired angle.

$\cos\phi 2 = (x^2 + y^2 - L1^2 - L2^2)/(2L1 \cdot L2)$

$\phi 2 = \cos^{-1}\left((x^2 + y^2 - L1^2 - L2^2)/(2L1.L2)\right)$

$\phi 1 = \theta - \phi 2$

We need to change these equations to the following code :-

```
void invers() {
// this fuction to determine  end effector of arm depending on value of x, y
lcd.clear();
INVANG= (pow (X, 2) tpow (X, 2) -pow (L1, 2) -pow (L2, 2) ) / (2*L1*L2);
ANGL2=acos (INVANG);
ANG2=ANGL2*180/PI;
ANG1=ALLANG-ANG2;
lcd.print ("Angle of Arm:");
lcd.setCurser (0,1);
lcd. print ("ANG1=");
lcd.print (ANG1);
cd.setCurseor (0,1);
lcd. print ("&");
lcd.setCurser (9,1);
lcd. print ("ANG2=");
lcd.print (ANG2);
Serial.print(ANG1);
motol.write(ANG1);
moto2.write(ANG2);
 delay (7000);
}}
```