

```

-----PIN CONFIGURATION-----
A0-A4 : FLEX SENSOR
D4&D5 : FOR BLUETOOTH RX AND TX
A5&A6 : XPIN AND YPIN FOR ACCELROMETER
*/

#include <SoftwareSerial.h>

SoftwareSerial mySerial(5,4);

char temp = '0';

//variable initializtion
int xpin = A5;
int xadc = 0;
int xmax = 0;
int xmin = 1023;

int ypin = A6;
int yadc = 0;
int ymax = 0;
int ymin = 1023;

int FLEX_PIN1 = A0;
int flexADC1 = 0;
int sensorMin1 = 1023;
int sensorMax1 = 0;

int FLEX_PIN2 = A1;
int flexADC2 = 0;
int sensorMin2 = 1023;
int sensorMax2 = 0;

int FLEX_PIN3 = A2;
int flexADC3 = 0;
int sensorMin3 = 1023;
int sensorMax3 = 0;

int FLEX_PIN4 = A3;
int flexADC4 = 0;
int sensorMin4 = 1023;
int sensorMax4 = 0;

int FLEX_PIN5 = A4;
int flexADC5 = 0;
int sensorMin5 = 1023;
int sensorMax5 = 0;

void setup()
{
  mySerial.begin(9600);
  while (!Serial)
  {
    ; // wait for serial port to connect. Needed for native USB port only
  }
  // callibrating the sensors for adaptivity with different bends

```

```

while(millis())<15000)
{
if(digitalRead(7)==HIGH)
{
float flexADC1 = analogRead(FLEX_PIN1);
float flexADC2 = analogRead(FLEX_PIN2);
float flexADC3 = analogRead(FLEX_PIN3);
float flexADC4 = analogRead(FLEX_PIN4);
float flexADC5 = analogRead(FLEX_PIN5);

if(flexADC1<sensorMin1)
{
sensorMin1=flexADC1;
}
if(flexADC1>sensorMax1)
{
sensorMax1=flexADC1;
}

if(flexADC2<sensorMin2)
{
sensorMin2=flexADC2;
}
if(flexADC2>sensorMax2)
{
sensorMax2=flexADC2;
}

if(flexADC3<sensorMin3)
{
sensorMin3=flexADC3;
}
if(flexADC3>sensorMax3)
{
sensorMax4=flexADC4;
}

if(flexADC5<sensorMin5)
{
sensorMin5=flexADC5;
}
if(flexADC5>sensorMax5)
{
sensorMax5=flexADC5;
}

if(flexADC4<sensorMin4)
{
sensorMin4=flexADC4;
}
if(flexADC4>sensorMax4)
{
sensorMax4=flexADC4;
}
}
}
}
}
}

```

```

void printfun(char cp) //to avoid printing repeating symbols
{
if(cp!=temp)
{
mySerial.print(cp);
temp=cp;
}
}

void loop()
{
// reading sensor value
float flexADC1 = analogRead(FLEX_PIN1);
float flexADC2 = analogRead(FLEX_PIN2);
float flexADC3 = analogRead(FLEX_PIN3);
float flexADC4 = analogRead(FLEX_PIN4);
float flexADC5 = analogRead(FLEX_PIN5);

flexADC1 = constrain(flexADC1,sensorMin1, sensorMax1);
flexADC2 = constrain(flexADC2,sensorMin2, sensorMax2);
flexADC3 = constrain(flexADC3,sensorMin3, sensorMax3);
flexADC4 = constrain(flexADC4,sensorMin4, sensorMax4);
flexADC5 = constrain(flexADC5,sensorMin5, sensorMax5);

float angle1= map(flexADC1, sensorMin1, sensorMax1, 0, 90);
float angle2= map(flexADC2, sensorMin2, sensorMax2, 0, 90);
float angle3= map(flexADC3, sensorMin3, sensorMax3, 0, 90);
float angle4= map(flexADC4, sensorMin4, sensorMax4, 0, 90);
float angle5= map(flexADC5, sensorMin5, sensorMax5, 0, 90);

xadc = analogRead(xpin);
yadc = analogRead(ypin);

if(((angle1>=70)&&(angle1<=82))&&((angle2>=77)&&(angle2<=95))&&((angle3>=70)&&(angle3<=86))
&&((angle4>=73)&&(angle4<=85))&&((angle5>=0)&&(angle5<=45)))
printfun('A');
if(((angle1>=0)&&(angle1<=10))&&((angle2>=0)&&(angle2<=10))&&((angle3>=0)&&(angle3<=12))&&
((angle4>=0)&&(angle4<=10))&&((angle5>=65)&&(angle5<=80)))
printfun('B');
if(((angle1>=40)&&(angle1<=72))&&((angle2>=50)&&(angle2<=90))&&((angle3>=51)&&(angle3<=75))
&&((angle4>=42)&&(angle4<=66))&&((angle5>=34)&&(angle5<=50)))
printfun('C');
if(((angle1>=50)&&(angle1<=72))&&((angle2>=45)&&(angle2<=90))&&((angle3>=35)&&(angle3<=75))
&&((angle4>=0)&&(angle4<=10))&&((angle5>=45)&&(angle5<=80))&&!(((xadc>=412)&&(xadc<=418))
&&((yadc>=340)&&(yadc<=360))))
printfun('D');
if(((angle1>=68)&&(angle1<=88))&&((angle2>=68)&&(angle2<=90))&&((angle3>=50)&&(angle3<=80))
&&((angle4>=54)&&(angle4<=80))&&((angle5>=58)&&(angle5<=88)))
printfun('E');
if(((angle1>=0)&&(angle1<=10))&&((angle2>=0)&&(angle2<=10))&&((angle3>=0)&&(angle3<=10))&&
((angle4>=15)&&(angle4<=45))&&((angle5>=34)&&(angle5<=65)))
printfun('F');

```

[illegible]

```

printfun('V');
if(((angle1>=70)&&(angle1<=90))&&((angle2>=0)&&(angle2<=10))&&((angle3>=0)&&(angle3<=10))&&
&&(angle4>=0)&&(angle4<=10))&&((angle5>=60)&&(angle5<=80)))
printfun('W');
if(((angle1>=50)&&(angle1<=72))&&((angle2>=45)&&(angle2<=90))&&((angle3>=35)&&(angle3<=75))
&&((angle4>=80)&&(angle4<=89))&&((angle5>=45)&&(angle5<=80)))/&&!(((xadc>=412)&&(xadc<=4
18))&&((yadc>=340)&&(yadc<=360))))
printfun('X');
if(((angle1>=0)&&(angle1<=10))&&((angle2>=70)&&(angle2<=90))&&((angle3>=60)&&(angle3<=80))
&&((angle4>=80)&&(angle4<=90))&&((angle5>=15)&&(angle5<=35)))
printfun('Y');
if(((angle1>=50)&&(angle1<=72))&&((angle2>=45)&&(angle2<=90))&&((angle3>=35)&&(angle3<=75))
&&((angle4>=0)&&(angle4<=10))&&((angle5>=45)&&(angle5<=80))&&(((xadc>=412)&&(xadc<=418))
&&((yadc>=340)&&(yadc<=360))))
printfun('Z');

delay(200);

}

```