

```

//display
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
//LiquidCrystal_I2C lcd(0x27);
char disp(16), buff(5);

//pin-pin motor-----
#define kiri 6
#define dirKi2 31
#define dirKi1 33
#define dirKa1 35
#define dirKa2 37
#define kanan 7

//-----

//variable PID-----
unsigned long lastTime;
double Output, Setpoint = 23 ,setpoint=10;
double errSum, lastErr;
double kp = 2 , kd = 1 ; //17 12 30
double kp1 = 2, kd1 = 4;
byte kecepatan = 60;

//-----

//tombol
#define tombol1 digitalRead(14)
#define tombol2 digitalRead(15)

//pin-pin ultrasonik-----
#define trigKa 9
#define echoKa 8
#define trigDe 5

```

```

#define echoDe 4
#define trigKi 3
#define echoKi 2
int jarakKa, jarakKi, jarakDe;
//-----

//pin-pin flame sensor-----
const int adcFire[4] = {A0, A1, A2, A3};
byte chFire[4];
byte fire;
//-----

#define semprot A0

void setup() {
  // put your setup code here, to run once:
  inisialMotor();
  inisialFire();
  inisialUltra();
  inisialDisplay();
  inisialPixy ();
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  utama();
  bacaApi();
  // if (fire > 0)
  // {
  //   navigasiApi();
  // }

```

```

// else
// {
//tampil();
// followWallKiri();
//  bacaUltra();
// }
// followWall();
//  bacaUltra();
//jalan(-130,-130);
// }
}

void inisialFire() {
  pinMode(semprot, OUTPUT);
  digitalWrite(semprot, HIGH);
  byte i = 0;
  for (i = 0; i < 4; i++) {
    pinMode(adcFire[i], INPUT);
  }
}

void bacaApi() {
  byte i = 0;
  int batasNilai = 500;
  for (i = 0; i <= 4; i++) {
    if (analogRead(adcFire[i]) < batasNilai) {
      chFire[i] = 1;
    }
    else {
      chFire[i] = 0;
    }
  }
}

```

```
Serial.print(chFire[0]); Serial.print(chFire[1]); Serial.print(chFire[2]); Serial.print(chFire[3]);  
Serial.println(chFire[4]);  
  
fire = (chFire[0] * 16) + (chFire[1] * 8) + (chFire[2] * 4) + (chFire[3] * 2) + (chFire[4] * 1);  
  
lcd.setCursor(5, 1);  
  
lcd.print(chFire[0]); lcd.print(chFire[1]); lcd.print(chFire[2]); lcd.print(chFire[3]); lcd.print(chFire[4]);  
  
}
```

```
void cekApi() {  
  while (1) {  
    bacaApi();  
    if (fire > 0)  
      navigasiApi();  
  }  
}
```

```
void navigasiApi() {  
  int PV;  
  while (1) {  
    bacaApi();  
    switch (fire) {  
      case B10000:  
        PV = 5;  
        jalan(-50, 50);  
        break;  
  
      case B11000:  
        PV = 5;  
        jalan(-50, 50);  
        break;
```

case B01000:

PV = 5;

jalan(-50, 50);

break;

case B11100:

PV = 5;

jalan(-50, 50);

break;

case B01100:

PV = 5;

jalan(-50, 50);

break;

case B11110:

PV = 5;

jalan(-50, 50);

break;

case B01110:

PV = 0;

jalan(50, 50);

padamkan();

break;

case B11111:

PV = 0;

jalan(60, 90);

padamkan();

```
break;

case B00100:
    PV = 0;
    jalan(60,90);
//    lcd.clear();
//    while (1) {
//        //        bacaUltra();
//        jalan(60, 90);
//        if (analogRead(A3) > 800) {
//            //            padamkan();
//            lcd.setCursor(0, 0); lcd.print(analogRead(A3));
//            henti();
//        }
//    }
break;
```

```
case B01111:
    PV = (-5);
    jalan(50, -50);
    break;
```

```
case B00111:
    PV = (-5);
    jalan(50, -50);
    break;
```

```
case B00110:
    PV = (-5);
```

```
jalan(50, -50);  
break;
```

```
case B00010:  
    PV = (-5);  
    jalan(50, -50);  
    break;
```

```
case B00011:  
    PV = (-5);  
    jalan(50, -50);  
    break;
```

```
case B00001:  
    PV = (-5);  
    jalan(50, -50);  
    break;
```

```
case B00000:  
    if (PV > 0) {  
        jalan(-50, 50);  
    }  
    if (PV < 0) {  
        jalan(50, -50);  
    }  
    if (PV = 0) {  
        jalan(50, 50);  
    }  
    break;
```

```
}
```

```

}
}

void padamkan() {
    jalan(0, 0);
    digitalWrite(semprot, LOW);
    while (1) {
        bacaApi();
        goyang();
        if (fire == 0) {
            digitalWrite(semprot, HIGH);
            henti();
            // loop();
        }
    }
}

void utama() {
    lcd.setCursor(0, 0); lcd.print(" Mulai ");

    if (!tombol1) {
jalanKanan:
        lcd.clear();
        while (1) {
            lcd.setCursor(0, 1); lcd.print("kanan");
            bacaApi();
            if (fire > 0) {
                navigasiApi();
            }
        }
    }
}

```



```
else {
    followWallKanan();
}
if (!tombol2) {
    goto jalanKiri;
}
}
if (!tombol2) {
jalanKiri:
    lcd.clear();
    while (1) {
        lcd.setCursor(0, 1); lcd.print("kiri");
        bacaApi();
        if (fire > 0) {
            navigasiApi();
        }
        else {
            followWallKiri();
        }
        if (!tombol1) {
            goto jalanKanan;
        }
    }
}
void inisialMotor() {
    pinMode (dirKa1, OUTPUT);
    pinMode (dirKa2, OUTPUT);
    pinMode (dirKi1, OUTPUT);
```

```
pinMode (dirKi2, OUTPUT);
}
void inisialDisplay() {
  lcd.begin(16, 2);
  lcd.backlight();
  lcd.setCursor(0, 0); lcd.print("  daus BIG  ");
  delay(800);
  lcd.clear();
  pinMode(14, INPUT_PULLUP);
  pinMode(15, INPUT_PULLUP);
}
void jalan(int moKi, int moKa) {
  if (moKi > 0) {
    digitalWrite(dirKi1, LOW); digitalWrite(dirKi2, HIGH);
  }
  else {
    moKi = abs(moKi);
    digitalWrite(dirKi1, HIGH); digitalWrite(dirKi2, LOW);
  }
  analogWrite(kiri, (moKi + 30));

  if (moKa > 0) {
    digitalWrite(dirKa1, HIGH); digitalWrite(dirKa2, LOW);
  }
  else {
    moKa = abs(moKa);
    digitalWrite(dirKa1, LOW); digitalWrite(dirKa2, HIGH);
  }
  analogWrite(kanan, moKa);
}
```

```
void henti() {
  while (1) {
    digitalWrite(dirKa1, LOW); digitalWrite(dirKa2, LOW); analogWrite(kanan, 0);
    digitalWrite(dirKi1, LOW); digitalWrite(dirKi2, LOW); analogWrite(kiri, 0);
    bacaApi();
    if (fire > 0) {
      navigasiApi();
    }
  }
}

void goyang() {
  jalan(80, -80); delay(300);
  jalan(-80, 80); delay(300);
  jalan(0, 0); delay(80);
  jalan(-80, 80); delay(280);
  jalan(80, -80); delay(300);
  jalan(0, 0); delay(80);
}

void loop()
{

  static int i = 0;
  int j;
  char buf[32];
  blocks = pixy.getBlocks();

  if (blocks)
  {
    i++;
    if (i%50==0)
```

```
{
  if(pixy.blocks[j].x>200)
  {
    Serial.println("Right");
    belokkanan();
  }
  else if(pixy.blocks[j].x<60)
  {
    Serial.println("Left");
    belokkiri();
  }
  else if(pixy.blocks[j].x>=60 && pixy.blocks[j].x<=200)
  {
    Serial.println("Middle");
    maju();
  }

  else{
    Serial.println("No Object Detected");
    berhenti();
  }

}

}

}

void inisialUltra() {
  pinMode(trigKa, OUTPUT); pinMode(echoKa, INPUT);
  pinMode(trigDe, OUTPUT); pinMode(echoDe, INPUT);
```

```
pinMode(trigKi, OUTPUT); pinMode(echoKi, INPUT);  
}
```

```
void bacaUltra() {  
  // lcd.clear();  
  //ultraKanan-----  
  digitalWrite(trigKa, LOW);  
  digitalWrite(trigKa, HIGH);  
  delayMicroseconds(10);  
  digitalWrite(trigKa, LOW);  
  jarakKa = pulseIn(echoKa, HIGH) / 58;  
  jarakKa = constrain(jarakKa, 3, 30);  
  Serial.println(jarakKa);  
  // lcd.setCursor(0,0);lcd.print("      ");  
  // lcd.setCursor(10, 0); lcd.print(jarakKa);  
  //-----  
  
  //ultraDe-----  
  digitalWrite(trigDe, LOW);  
  digitalWrite(trigDe, HIGH);  
  delayMicroseconds(10);  
  digitalWrite(trigDe, LOW);  
  jarakDe = pulseIn(echoDe, HIGH) / 58;  
  jarakDe = constrain(jarakDe, 3, 30);  
  Serial.println(jarakDe);  
  // lcd.setCursor(5, 0); lcd.print(jarakDe);  
  //-----  
  
  //  
  // //ultraKi-----  
  digitalWrite(trigKi, LOW);
```

```

digitalWrite(trigKi, HIGH);
delayMicroseconds(10);
digitalWrite(trigKi, LOW);
jarakKi = pulseIn(echoKi, HIGH) / 58;
jarakKi = constrain(jarakKi, 3, 30);
Serial.println(jarakKi);
// lcd.setCursor(0, 0); lcd.print(jarakKi);
// //-----
lcd.setCursor(0, 0);
sprintf(dis, " %2d ", jarakKi); lcd.println(dis);
sprintf(dis, " %2d ", jarakDe); lcd.println(dis);
sprintf(dis, " %2d ", jarakKa); lcd.println(dis);
// lcd.display();
}

```

```

void followWallKanan() {
  bacaUltra();
  if (jarakDe <= 17 ) {
    jalan(-50, 100);
    // delay(150);
  }
  else {
    unsigned long now = millis();
    double timeChange = (double)(now - lastTime);
    double error = Setpoint - jarakKa;
    double dErr = (error - lastErr) / timeChange;
    Output = kp * error + kd * dErr;
    int motorKiri = kecepatan - Output;
    int motorKanan = kecepatan + Output;
    motorKiri = constrain(motorKiri, -100, 150);
  }
}

```

```

motorKanan = constrain(motorKanan, -75, 125);
Serial.print("jarak=="); Serial.println(jarakKa);
Serial.print("kiri=="); Serial.println(motorKiri);
Serial.print("kanan=="); Serial.println(motorKanan);
jalan(motorKiri, motorKanan);
lastErr = error;
lastTime = now;
}
}

```

```

void followWallKiri() {
  bacaUltra();
  if (jarakDe <= 17 ) {
    jalan(100, -50);
    // delay(150);
  }
  else {
    unsigned long now = millis();
    double timeChange = (double)(now - lastTime);
    double error = setpoint - jarakKi;
    double dErr = (error - lastErr) / timeChange;
    Output = kp1 * error + kd1 * dErr;
    int motorKiri = kecepatan + Output;
    int motorKanan = kecepatan - Output;
    motorKiri = constrain(motorKiri, -100, 150);
    motorKanan = constrain(motorKanan, -75, 125);
    Serial.print("jarak=="); Serial.println(jarakKi);
    Serial.print("kiri=="); Serial.println(motorKiri);
    Serial.print("kanan=="); Serial.println(motorKanan);
    jalan(motorKiri, motorKanan);
  }
}

```

```
lastErr = error;  
lastTime = now;  
}  
}
```