LAMPIRAN A
Listing Program Arduino

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <HX711_ADC.h>
#include <Servo.h>
LiquidCrystal_I2C lcd(0x27, 20, 4);
HX711_ADC LoadCell(23, 25);//Dout,SCK
Servo servo;
#define RF 48//RELLAY AC
#define RN 50
#define R24 2//RELLAY 24
#define Lsw1 24//LIMITSWITCH
#define Lsw2 22
#define servPin 4
#define M1 4
#define M2 5
#define M3 6
#define M4 7
#define prox 8
#define suhuPin A14
#define button1 11
#define button2 13
#define button3 12
#define enStep 51
```

```
#define motorStep 39
#define dirStep 37
int setSuhu = 65;//setting suhu jepitan
int waktuJepit = 2500;//setting delay jepitan
int delayTarik = 1700;
bool limit1, limit2, limitBawah, button100, button250,
buttonStart, Start;
long int timeSw, millisBucket;;
const int calVal_calVal_eepromAdress = 0;
unsigned long t = 0;
float gram;
int setGram, setMode;
float temp;
bool Show, plastik;
String startStatus;
void setup() {
  Serial.begin(9600);
  lcd.init();
  lcd.backlight();
  servo.attach(3);
  Serial.println("Starting...");
  //INISIALISASI PIN
  pinMode(RF, OUTPUT);
  pinMode(RN, OUTPUT);
  pinMode(R24, OUTPUT);
  pinMode(M1, OUTPUT);
```

```
pinMode(M2, OUTPUT);
pinMode(M3, OUTPUT);
pinMode(M4, OUTPUT);
pinMode(enStep, OUTPUT);
pinMode(motorStep, OUTPUT);
pinMode(dirStep, INPUT);
pinMode(Lsw1, INPUT);
pinMode(Lsw2, INPUT);
pinMode(prox, INPUT);
pinMode(suhuPin, INPUT);
pinMode(button1, INPUT);
pinMode(button2, INPUT);
pinMode(button3, INPUT);
lcd.setCursor(0, 0);
lcd.print("Mesin Pengemas..");
lcd.setCursor(0, 1);
lcd.print("SetUp....");
digitalWrite(RF, HIGH);
digitalWrite(RN, HIGH);
digitalWrite(R24, HIGH);
servo.write(129);
///KALIBRARI LOADCELL
float calibrationValue; // calibration value
calibrationValue = 470.0;
LoadCell.begin();
unsigned long stabilizingtime = 2000;
```

```
      boolean _tare = true;
      LoadCell.start(stabilizingtime, _tare);
      if (LoadCell.getTareTimeoutFlag()) {
      Serial.println("Timeout, check MCU>HX711 wiring and pin
     designations");
      }
     else {
     LoadCell.setCalFactor(calibrationValue);
      Serial.println("Startup is complete");
  }
  while (!LoadCell.update());
  Serial.print("Calibration value: ");
  Serial.println(LoadCell.getCalFactor());
  Serial.print("HX711 measured conversion time ms: ");
  Serial.println(LoadCell.getConversionTime());
  Serial.print("HX711 measured sampling rate HZ: ");
  Serial.println(LoadCell.getSPS());
  Serial.print("HX711 measured settlingtime ms: ");
   Serial.println(LoadCell.getSettlingTime());
 Serial.println("Note that the settling time may increase
significantly if you use delay() in your sketch!");
      if (LoadCell.getSPS() < 7) {
        Serial.println("!!Sampling rate is lower than specification,
check MCU>HX711 wiring and pin designations");
      }
      else if (LoadCell.getSPS() > 100) {
```

```
        Serial.println("!!Sampling rate is higher than specification,
check MCU>HX711 wiring and pin designations");
      }
      lcd.clear();

    }
    int mode, set;
    void loop() {
      heaterSteadyTemp();//MONITORING SUHU
      bacaSwitch();//BACA LIMITSWITCH
      loadCell();//MONITORING LOADCELL
      kalibrasiMekanik();
      showLcd();
      jalan();
      if (button100 == 1) {
        if (set > 2)set = 0;
        else {
          set += 1;
        }
        delay(300);
      }
      if (button250 == 1) {
        if (setMode > 1)setMode = 0;
        else {
          setMode += 1;
        }
```

```
    delay(300);
  }
  if (set == 1)setGram = 100;
  else if (set == 2)setGram = 250;
  else {
    setGram = 0;
  }
}

int packing = 1;
long int millisPack, millisPress;
bool start;
void jalan() {//JALAN PACKING
  if (setMode == 0) {
    packing = 1;
    start = 0;
    startStatus = "OFF";
  }
  else if (setMode == 1) {
    start = 1;
    startStatus = "ON";
  }
  Serial.println(packing);
  switch (packing) {
    case 1:
      analogWrite(M1, 0);
```

```
analogWrite(M2, 0);
analogWrite(M3, 0);
analogWrite(M4, 0);
servo.write(129);
digitalWrite(R24, HIGH);
Show = 1;
if (start == 1 && setGram != 0) {
  packing = 2;
  Show = 0;
  millisPress = millis();
  lcd.clear();
  lcd.setCursor(0, 1);
  lcd.print("Proses");
  lcd.setCursor(8, 1);
  lcd.print(packing);
}
else if (start == 1 && setGram == 0) {
  for (int i = 0; i < 3; i++) {
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print("Set Isian!!!");
    delay(500);
  }
  setMode = 0;
}
break;
```

```
case 2:
  digitalWrite(R24, LOW);
  if (limitBawah == 1) {
    digitalWrite(R24, HIGH);
    digitalWrite(RN, LOW);
    digitalWrite(RF, LOW);
    delay(waktuJepit);
    packing = 33;
  }
  break;
case 33:
  digitalWrite(R24, LOW);
  delay(600);
  digitalWrite(R24, HIGH);
  delay(500);
  packing = 3;
  break;
case 3:
  analogWrite(M3, 0);
  analogWrite(M4, 180);
  delay(delayTarik);
  analogWrite(M3, 0);
  analogWrite(M4, 0);
  packing = 34;
  break;
case 34:
```

```
  digitalWrite(R24, LOW);
  if (limitBawah == 1) {
    digitalWrite(R24, HIGH);
    digitalWrite(RN, LOW);
    digitalWrite(RF, LOW);
    delay(waktuJepit);
    packing = 344;
  }
  break;
case 344:
  digitalWrite(R24, LOW);
  delay(600);
  digitalWrite(R24, HIGH);
  delay(1000);
  packing = 4;
  break;
case 4:
  servo.write(170);
  if (gram > setGram) {
    servo.write(129);
    delay(1500);
    packing = 54;
  }
  break;
case 54:
  analogWrite(M3, 0);
```

```
      analogWrite(M4, 180);
      delay(delayTarik);
      analogWrite(M3, 0);
      analogWrite(M4, 0);
      packing = 5;
      break;
    case 5:
      bucket(1, 2000);
      delay(2000);
      bucket(0, 2000);
      delay(2000);
      packing = 34;
      break;
  }
}
void kalibrasiMekanik() {
  if (limit1 == 1 && start != 1) {
    analogWrite(M3, 0);
    analogWrite(M4, 180);
  }

  if (limit2 == 1 && start != 1) {
    digitalWrite(R24, LOW);
  }
}
```

```
void bucket(bool arah, int lama) {//GERAK PENAMPUNG
  if (arah == 1) {
    analogWrite(M1, 0);
    analogWrite(M2, 30);//turun
    delay(lama);
    analogWrite(M1, 0);
    analogWrite(M2, 0);//turun
  }
  else {
    analogWrite(M1, 100);
    analogWrite(M2, 0);//naik
    delay(lama);
    analogWrite(M1, 0);
    analogWrite(M2, 0);//turun
  }
}

void showLcd() {
  lcd.setCursor(0, 1);
  lcd.print("gram :");
  lcd.setCursor(8, 1);
  lcd.print(gram);

  lcd.setCursor(0, 2);
  lcd.print("suhu :");
  lcd.setCursor(7, 2);
```

```
      lcd.print(temp);

      lcd.setCursor(0, 0);
      lcd.print("Isian :");
      lcd.setCursor(8, 0);
      lcd.print(setGram);

      lcd.setCursor(0, 3);
      lcd.print("sistem:");
      lcd.setCursor(8, 3);
      lcd.print(startStatus);
    }
    float getTemperature(char unit) {//FUNGSI BACA SUHU
      const float R1 = 10000;  // Nilai Resistor ke GND 10K / 10000
ohm
      const float c1 = 1.009249522e-03, c2 = 2.378405444e-04, c3
= 2.019202697e-07;
      int adc = analogRead(suhuPin);
      float R2 = R1 * (1023.0 / (float)adc - 1.0);
      float logR2 = log(R2);
      float T = (1.0 / (c1 + c2 * logR2 + c3 * logR2 * logR2 *
logR2));  // KElvin
      float Tc = T - 273.15;                                  // CELCIUS
      float Tf = (Tc * 9.0) / 5.0 + 32.0;                     //
FARENHEIT
```

```
  if (unit == 'K') return T;
  else if (unit == 'F') return Tf;
  else return Tc;
 }

 long int millisSuhu;
 void heaterSteadyTemp() {//FUNGSI MONITORING
HEATER
  if (millis() - millisSuhu > 500) {
   lcd.clear();
   temp = getTemperature('C');
   //   if (temp > 90 + 2) {
   //     digitalWrite(RN, HIGH);
   //     digitalWrite(RF, HIGH);
   //   }
   //   else if (temp < 90 - 2) {
   digitalWrite(RN, LOW);
   digitalWrite(RF, LOW);
   //   }
   millisSuhu = millis();
  }
 }
```

```
void loadCell() {//FUNGSI BACA LOADCELL
  static boolean newDataReady = 0;
  if (LoadCell.update()) newDataReady = true;
  if (newDataReady) {
    if (millis() - t > 500) {
      gram = LoadCell.getData();
      newDataReady = 0;
      t = millis();
    }
  }
  if (Serial.available() > 0) {
    char inByte = Serial.read();
    if (inByte == 't') LoadCell.tareNoDelay();
  }
  if (LoadCell.getTareStatus() == true) {
    Serial.println("Tare complete");
  }
}

void bacaSwitch() {// FUNGSI BACA LIMIT SWITCH
  if (millis() - timeSw > 50) {
    button100 = digitalRead(button3);
    button250 = digitalRead(button2);
```

```
        limitBawah = digitalRead(dirStep);
        limit1 = digitalRead(Lsw1);
        limit2 = digitalRead(Lsw2);
        //    Serial.println(String(button100) + "\t" +
String(button250) + "\t" + String(setMode) + "\t" + String(set) +
"\t" + String(setGram) + "\t" + String(limitBawah));
        timeSw = millis();
      }
```

LAMPIRAN B

Dokumentasi Penggunaan Alat

B. Cara Penggunaan Alat

No	Kegiatan	Dokumentasi
1	Masukan plastik roll ke pipa pembentuk kemasan.	
2	Masukan kedua ujung plastik ke roda penarik, pastikan terjepit.	
3	Masukan biji kopi ke wadah penampungan	
4	Hubungkan colokan ke	

	stop kontak sumber AC	
5	Tekan tombol power merah untuk menyalakan alat	
6	Pada LCD akan muncul notifikasi seperti pada gambar hingga proses kalibrasi selesai	
7	Jika tampilan LCD sudah seperti gambar maka alat siap digunakan.	
8	Tekan tombol sebelah kanan untuk menentukan set isian	

	yaitu 100g atau 250g	
9	Tekan tombol sebelah kiri untuk menjalankan alat	
10	Tekan tombol power merah untuk memberhentikan proses pengemasan.	

C. Hasil

Gambar hasil 250 gram & 100 gram