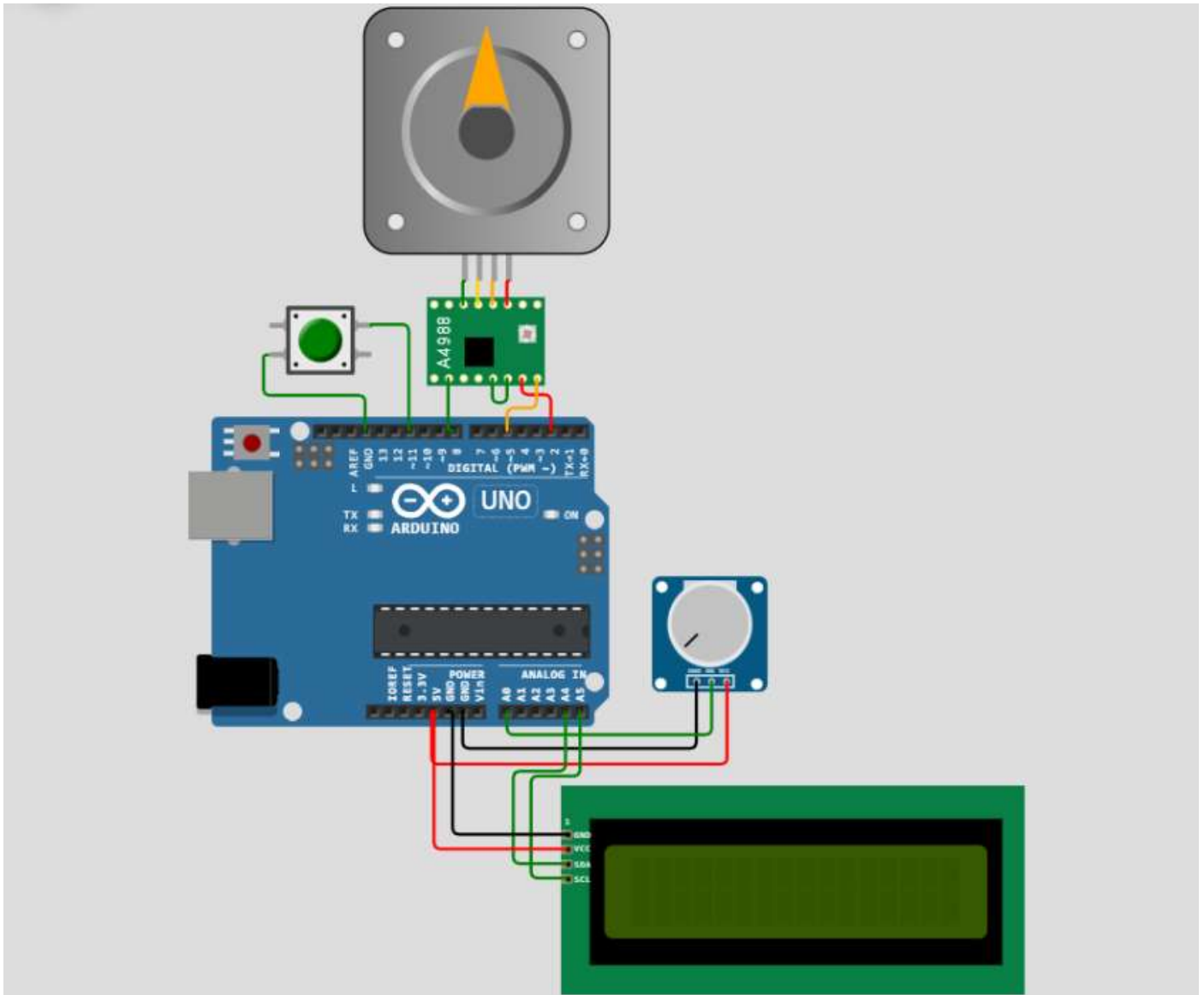


## GUIDA LINEARE 2



```
#include <LiquidCrystal_I2C.h>
#define I2C_ADDR 0x27
#define LCD_COLUMNS 20
#define LCD_LINES 4
LiquidCrystal_I2C lcd(I2C_ADDR, LCD_COLUMNS, LCD_LINES);

// FC finecorsa posizione HOME
// rotazione oraria --> allontana slitta dal FC
// rotazione oraria --> avvicina slitta al FC
#define DIR_PIN 5 // X
#define STEP_PIN 2 // X
#define EN_PIN 8 // pin abilitazioe driver
#define FC_PIN 11 // driver

int steps_x_round= 200; // 200 steps al giro
int pot; // potenziometro per velocità
int delay_step=1000;
int FC_STATE = LOW; // stato finecorsa
int HOME_STATE=LOW; // per sapere se sono a HOME
int FLAG_STOP=LOW;
float position;
```

```

void setup() {
  Serial.begin(115200);
  lcd.begin(16, 2);

  pinMode(DIR_PIN, OUTPUT);
  pinMode(STEP_PIN, OUTPUT);
  pinMode(EN_PIN, OUTPUT);
  pinMode(FC_PIN, INPUT_PULLUP);
  digitalWrite(EN_PIN, LOW);

  // Attivo LCD
  lcd.init(); lcd.backlight();
  lcd.setCursor(0, 0); lcd.print("...");

  delay(1000);
}

void loop() {
  //HOME antioraria --> vado alla posizione di riposo
  stepHOME(false, DIR_PIN, STEP_PIN, 20000); // 200 mm di corsa max
  delay(1000);

  if (FLAG_STOP== LOW) {
    //oraria--> mi spoto di 5mm --> 5/2mm=2.5 * 200 passi=500 step
    step(true, DIR_PIN, STEP_PIN, 500);
    FLAG_STOP= true;
  }

  delay(1000);
}

// dir = true= oraria
void step(boolean dir, byte dirPin, byte stepperPin, int steps)
{
  digitalWrite(dirPin, dir);
  for (int i = 0; i < steps; i++) {
    FC_STATE = digitalRead(FC_PIN);
    if (FC_STATE == LOW) {
      Serial.println("premuto");
      HOME_STATE= HIGH;
      position= 0.0;
      break;
    }
    if (FC_STATE == HIGH) {
      Serial.println("non premuto");
      digitalWrite(stepperPin, HIGH);
      delayMicroseconds(delay_step);
      digitalWrite(stepperPin, LOW);
      delayMicroseconds(delay_step);

      position= 2.0 * i / steps_x_round;
      Serial.println("mm "); Serial.println(position);
      lcd.setCursor(0, 0); lcd.print("mm  "); lcd.print(position);
    }
  }
  position= position + 2.0 / steps_x_round;
  Serial.println("mm "); Serial.println(position);
  lcd.setCursor(0, 0); lcd.print("mm  "); lcd.print(position);
}

```

```
void stepHOME(boolean dir, byte dirPin, byte stepperPin, int steps)
{
  // SE NON SONO A HOME
  if (HOME_STATE == LOW) {
    digitalWrite(dirPin, dir);
    delay(100);
    for (int i = 0; i < steps; i++) {
      FC_STATE = digitalRead(FC_PIN);
      if (FC_STATE == LOW) {
        Serial.println("premuto");
        HOME_STATE= HIGH;
        position= 0.0;
        break;
      }
      else if (FC_STATE == HIGH) {
        Serial.println("non premuto");
        digitalWrite(stepperPin, HIGH);
        delayMicroseconds(delay_step);
        digitalWrite(stepperPin, LOW);
        delayMicroseconds(delay_step);
      }
    }
  }
}
```