

# Arduino Code for Syringe Pump Operation

Jacy Robbins, Nathan Park, Isabella Blauzdys, Rylan Hunt

```
#include <AccelStepper.h>

// Motor & control pins

#define STEP_PIN 3

#define DIR_PIN 4


// LED pins

#define GREEN_LED 9

#define RED_LED 8


// Switch pins

#define CONTROL_SWITCH_PIN 12 // Toggle switch: run (LOW) / pause (HIGH)

#define LIMIT_SWITCH_PIN 10 // Limit switch (normally closed)


AccelStepper stepper(AccelStepper::DRIVER, STEP_PIN, DIR_PIN);


// ===== USER SETTINGS =====

const int syringeSize_mL = 10; // 10 for 10mL syringe, 20 for 20mL syringe

const float targetFlowRate_mL_per_min = 30; // Target flow rate in mL/min

// =====


// Internal constants

const float MAX_STEPS_PER_SEC = 1000.0;

const float STEP_DISTANCE_MM = 0.04; // Updated mechanical step size
```

```
// Variables

float area_mm2 = 0.0;

float distancePerML_mm = 0.0;

float motorSpeedStepsPerSec = 0.0;


// States

bool systemInitialized = false;

bool limitTriggered = false;


void setup() {

    // Setup pins

    pinMode(CONTROL_SWITCH_PIN, INPUT_PULLUP);

    pinMode(LIMIT_SWITCH_PIN, INPUT_PULLUP);

    pinMode(GREEN_LED, OUTPUT);

    pinMode(RED_LED, OUTPUT);


    // Set syringe area

    if (syringeSize_mL == 10) {

        area_mm2 = 165.0;

    } else if (syringeSize_mL == 20) {

        area_mm2 = 283.5;

    } else {

        area_mm2 = 165.0;

    }


    // Calculate linear conversion
```

```
distancePerML_mm = 1000.0 / area_mm2;

float mm_per_min = targetFlowRate_mL_per_min * distancePerML_mm;

float mm_per_sec = mm_per_min / 60.0;

motorSpeedStepsPerSec = mm_per_sec / STEP_DISTANCE_MM;


if (motorSpeedStepsPerSec > MAX_STEPS_PER_SEC) {

    motorSpeedStepsPerSec = MAX_STEPS_PER_SEC;

}


stepper.setMaxSpeed(MAX_STEPS_PER_SEC);
stepper.setSpeed(motorSpeedStepsPerSec);


// Wait until toggle switch is flipped LOW to initialize
while (digitalRead(CONTROL_SWITCH_PIN) == HIGH) {

    // Wait for switch to be flipped to LOW (start)

}

delay(50); // small debounce

systemInitialized = true;

}


void loop() {

    // Read limit switch

    limitTriggered = digitalRead(LIMIT_SWITCH_PIN) == HIGH;


    // Read current toggle switch state

    bool switchState = digitalRead(CONTROL_SWITCH_PIN);
```

```
if (limitTriggered) {  
    // LIMIT TRIGGERED: Immediate full stop  
  
    digitalWrite(GREEN_LED, LOW);  
  
    digitalWrite(RED_LED, HIGH);  
  
    return;  
}
```

```
if (systemInitialized) {  
    if (switchState == LOW) {  
        // Switch LOW: Motor runs  
  
        stepper.runSpeed();  
  
        digitalWrite(GREEN_LED, HIGH);  
  
        digitalWrite(RED_LED, LOW);  
    } else {  
        // Switch HIGH: Motor paused  
  
        digitalWrite(GREEN_LED, HIGH); // Yellow (Green + Red ON)  
  
        digitalWrite(RED_LED, HIGH);  
  
        delay(10); // stabilize LEDs  
    }  
}  
}
```