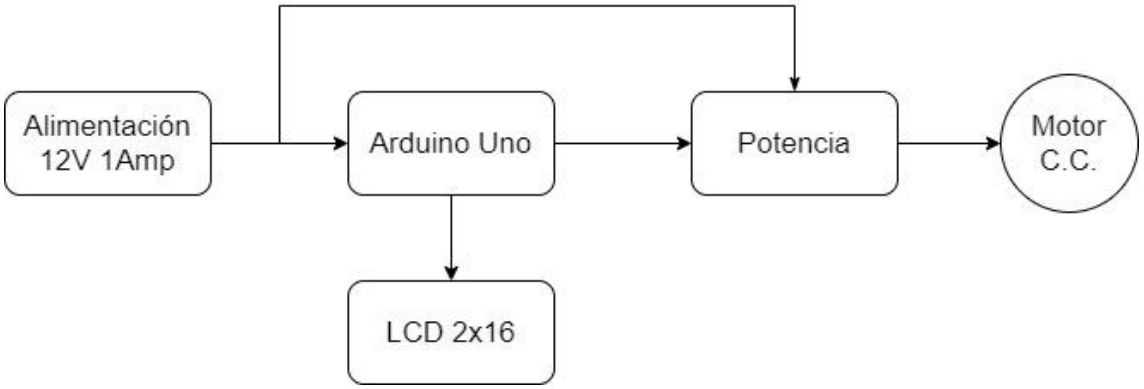
Responsable	IFIBIO “Houssay” UBA-CONICET	Ticket #
Dr. Esteban Valverde	Infusion pump for electrophysiology experimental setup	20200311
Start date: 11/03/2020		End date: 23/08/2021

This is the development of an infusion pump for the electrophysiology patch clamp setup of slices. To do this, he provided a peristaltic pump of the Goso brand AB11 of 12V – 7.5W. This pump is DC and reaches a maximum luxury of 50mL/min.

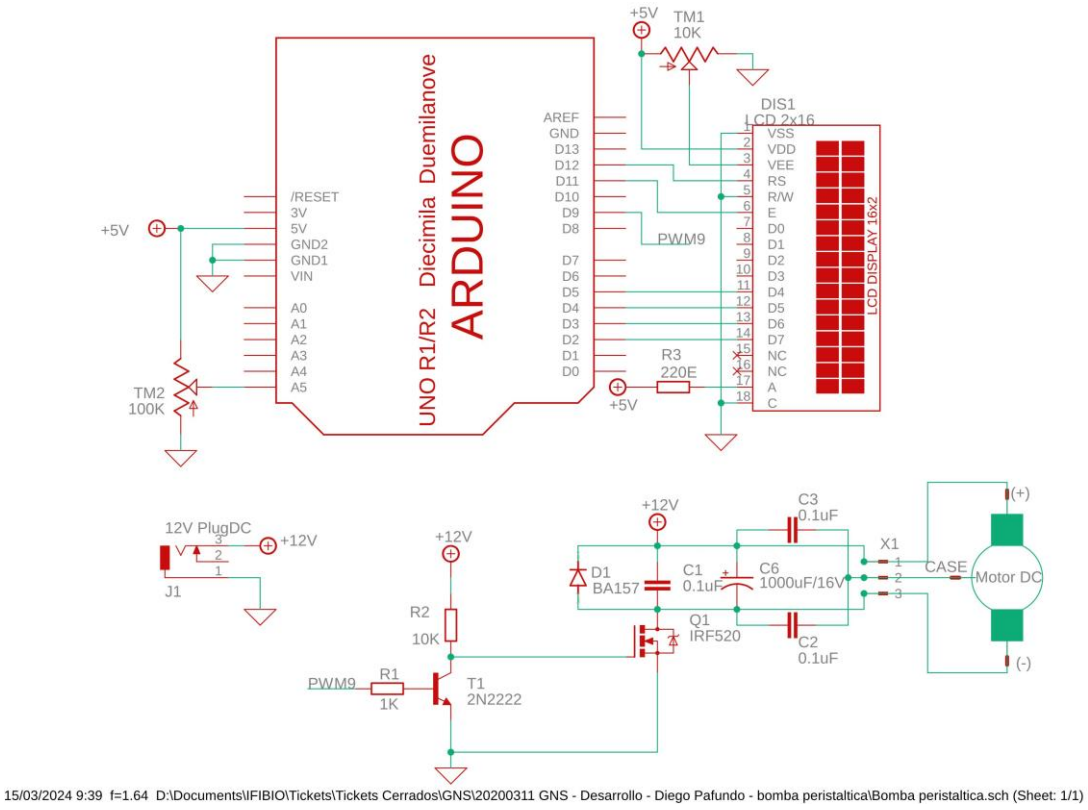
Schematic Circuit

It was proposed to develop a controller using an Arduino Uno board, in order to use PWM as a DC amplitude control for motor speed. In addition, the system must have a display to inform the chosen flow rate.



The schematic circuit is as follows

Responsable	IFIBIO “Houssay” UBA-CONICET	Ticket #
Dr. Esteban Valverde	Infusion pump for electrophysiology experimental setup	20200311
Start date: 11/03/2020		End date: 23/08/2021



15/03/2024 9:39 f=1.64 D:\Documents\IFIBIO\Tickets\Tickets Cerrados\GNS\20200311 GNS - Desarrollo - Diego Pafundo - bomba peristaltica\Bomba peristaltica.sch (Sheet: 1/1)

The TM1 potentiometer adjusts the speed. To do this, a continuous level is read at A5 and projected onto output 9 as a variation of the pulse width. This pin feeds the power circuit, so the DC value that reaches the motor will depend on the pulse width chosen.

On the other hand, the D1 diode filters out the spurious, as does the C6 electrolytic capacitor, so that the waveform between the motor terminals is as close to a continuous wave as possible. Finally, C1, C2 and C3 filter out the noise that the engine may generate.

The LCD circuit was taken from the Arduino display management tutorials, as well as the software and programming library.

Software Development

The software is basically a loop that reads the level of the potentiometer at A5 and sets a pulse width at PWM9. A calibration constant indicates on the LCD display the amount of mL/min that is passing through the pump. It should be noted that this value was obtained experimentally by measuring at maximum flow (12V) the amount of water that the pump was capable of pumping.

The flow chart of the software is as follows

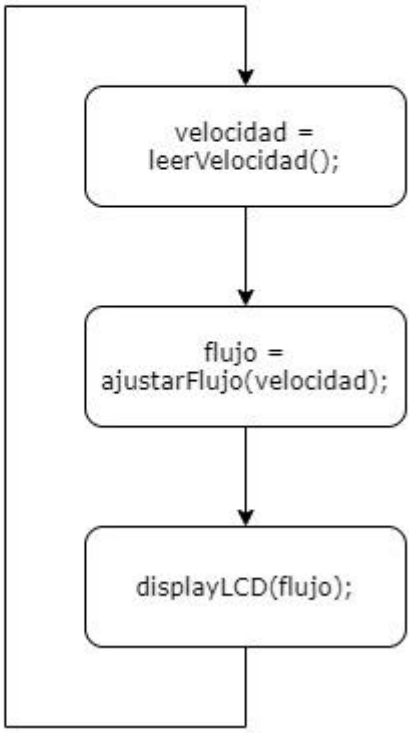
Setup general

void setup()

```
declara e inicializa constantes y variables globales

void setup()
{
    inicia pin analogico entrada y PWM salida
    inicia LCD
}

void loop(){}
```

Diagrama de flujo general

void loop()



The software listing is as follows

```
#include <LiquidCrystal.h>

/* MAPA DE PINES DEL ARDUINO UNO
D2: LCD DB7
D3: LCD DB6
D4: LCD DB5
D5: LCD DB4
D9: SALIDA PWM DEL MOTOR
D11: LCD EN
D12: LCD RS
A5: POTENCIOMETRO ENTRADA FLUJO
*/

// se midieron 500mL en 10min, entonces
#define FLUJOMAXIMO (500.0/10.0)

// initialize the library by associating any needed LCD
// interface pin with the arduino pin number it is connected to
```

```
// http://www.arduino.cc/en/Tutorial/LiquidCrystalDisplay
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

const int potenPin = A5; // leo el potenciometro de velocidad
const int motorPin = 9;  // ajusto el PWM de salida al motor
const int buzzerPin = A1;

void setup()
{
    // inicializo pines analogico de entrada y PWM de salida
    pinMode(potenPin, INPUT);
    pinMode(motorPin, OUTPUT);
    analogWrite(motorPin, 0);

    // inicializo buzzer de alarma apagado
    pinMode(buzzerPin, OUTPUT);
    digitalWrite(buzzerPin, LOW);

    // set up the LCD's number of columns and rows:
    lcd.begin(16, 2);

    // Print a message to the LCD.
    lcd.clear();
    lcd.print("    BOMBA DE    ");
    lcd.setCursor(0, 1);
    lcd.print("    INFUSION    ");
    delay(1000);
    lcd.clear();
}

unsigned char leerVelocidad(void)
{
    unsigned char velocidad;
    velocidad = analogRead(potenPin)/4;
    return velocidad;
}

unsigned char ajustarFlujo(unsigned char velocidad)
{
    unsigned char flujo;
    analogWrite(motorPin, velocidad);

    // ecuacion para convertir a flujo
    flujo = (float)velocidad * FLUJOMAXIMO / 255.0;
    return flujo;
}
```

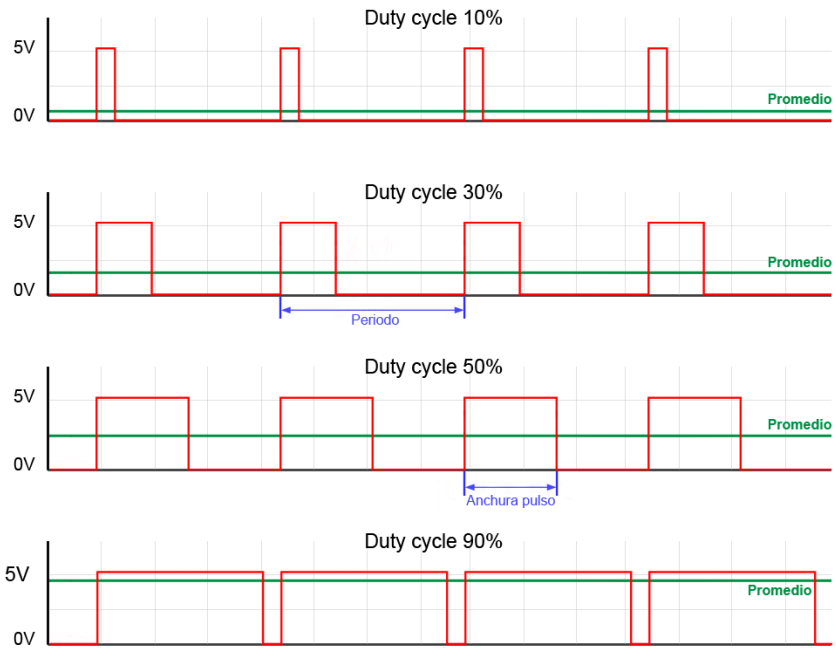
```
void loop()
{
    static char linea1[16];
    static char linea2[16];
    static char strflujototal[8];
    static char llenado = 255;
    static unsigned char i = 0;
    static unsigned char flujo = 0; // valor expresado en ml/min
    static unsigned int flujototal = 0; // expresado en ml
    static unsigned char velocidad=0, oldvelocidad=0; // 0-255

    //---------------------------------------------
    // leo velocidad y calculo el flujo
    velocidad = leerVelocidad();
    if(velocidad != oldvelocidad)
    {
        flujo = ajustarFlujo(velocidad);
        oldvelocidad = velocidad;
    }

    //---------------------------------------------
    // informacion al LCD
    flujototal += flujo;

    // Arduino no tiene sprintf para flotantes, uso “dtostrf”
    dtostrf((float)flujototal/60.0, 7, 1, strflujototal);
    sprintf(linea1, "TOTAL %7s ml", strflujototal);
    lcd.setCursor(0, 0);
    lcd.print(linea1);

    sprintf(linea2, "       %2d ml/min", flujo);
    lcd.setCursor(0, 1);
    lcd.print(linea2);

    // imprimo el cursor rotatorio de llenado
    lcd.setCursor(i, 1);
    lcd.print(llenado);

    //---------------------------------------------
    // cada 5 segundos, i retorna a 0
    i = (i+1) % 5;
    delay(1000);
}
```

Finished model

Responsable	IFIBIO “Houssay” UBA-CONICET	Ticket #
Dr. Esteban Valverde	Infusion pump for electrophysiology experimental setup	20200311
Start date: 11/03/2020		End date: 23/08/2021

Limitations

The Goso motor is a DC motor. The data sheets say that the flow rate is 0 – 50mL/min, depending on the DC voltage with which the motor is powered. On the other hand, the output of the PWM is a pulse train in which the activity cycle is varied. The frequency of the pulse train is 490Hz.



For an activity cycle close to 90%, the output of the PWM is practically a DC. However, for low activity cycle value, the output waveform is completely pulsatile. This makes it difficult to use the motor at these activity cycle values, precisely because the waveform is no longer continuous.

Responsable	IFIBIO “Houssay” UBA-CONICET	Ticket #
Dr. Esteban Valverde	Infusion pump for electrophysiology experimental setup	20200311
Start date: 11/03/2020		End date: 23/08/2021

The addition of a large capacitor between motor terminals has a grinding effect that improves the use of the pump at low flow.

Experimentally it was observed that between 15-50mL the pump works. At lower flow values, the engine does not have enough energy to start.

Ideally, a stepper motor would be used to have full control at any flow rate, but the design was made from this motor that Diego offered when making the design proposal for the infusion pump.