

Министерство Образования и Исследований Республики Молдова
Технический Университет Молдовы
Факультет Вычислительной Техники, Информатики и Микроэлектроники
Департамент Программной Инженерии и Автоматики

Лабораторная работа №4

по предмету «Интернет вещи»

Выполнил:

ст. гр. ТІ-196

Н. Шарафудинов

Проверил:

А. Бырназ

Тема: Работа с двигателями и драйверами.

Задание:

Создать приложение на базе микроконтроллера, которое будет управлять приводными устройствами

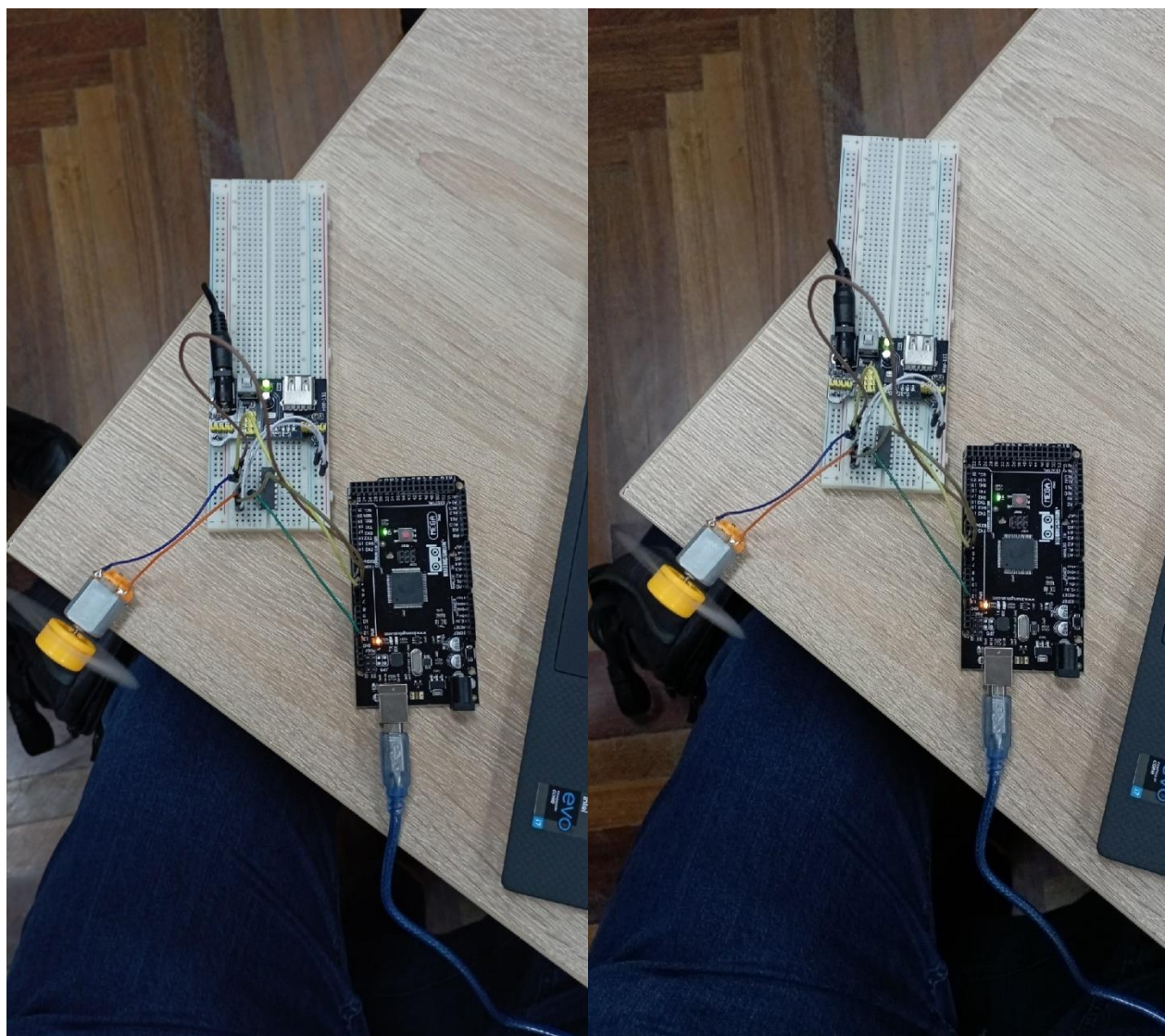
с помощью команд, получаемых от последовательного интерфейса и сообщающая састаяния системы на ЖК-дисплей.

Исполнительные механизмы будут следующими:

1. электрическая лампочка через реле с командами включения и выключения
2. Двигатель постоянного тока с командами установки мощности двигателя в диапазоне (-100% .. 100%), т.е. вперед и назад, через драйвер L298 или DRV8833.

Драйверы периферийного управления будут выполнены на разных уровнях абстракции

Схема



Вывод

В данной лабораторной работе я познакомился с работой драйверов в Arduino, используя драйвер L293D для работы с драйвером.

Приложение

```
// Motor B connections
#define MSG_LEN 100
int enB = 3;
int in3 = 5;
int in4 = 4;
int L = 9;
int R = 10;
unsigned long timing = 0;

static int serial_putchar(char c, FILE *f) {
    return Serial.write(c);
}

static int serial_getchar(FILE *) {
    while (Serial.available() <= 0) {}
    return Serial.read();
}

static FILE serial_stdinout = { .buf = NULL, .unget = 0, .flags = _FDEV_SETUP_RW,
    .size = 0, .len = 0, .put = serial_putchar, .get = serial_getchar, .udata = 0 };

void setup() {
    Serial.begin(9600);
    // Set up stdout and stdin
    stdout = &serial_stdinout;
    stdin = &serial_stdinout;
    stderr = &serial_stdinout;

    // Set all the motor control pins to outputs
    pinMode(enB, OUTPUT);
    pinMode(in3, OUTPUT);
    pinMode(in4, OUTPUT);

    pinMode(L, OUTPUT);
    pinMode(R, OUTPUT);

    // Turn off motors - Initial state
    digitalWrite(in3, LOW);
```

```

    digitalWrite(in4, LOW);
}

void loop() {

    int speed = 255;
    char msg[MSG_LEN] = { 0 };
    bool direction;
    printf("\nEnter [-100; 100] ");

    scanf("%s", msg);
    speed = my_getnbr(msg);
    printf("%d", speed);

    if (speed < 0) {
        speed *= -1;
        direction = false;
    } else {
        direction = true;
    }

    directionControl(speed, direction);
    delay(100);
}

void directionControl(int speed, bool dir) {

    float duty = map(speed, 0, 100, 0, 255);
    analogWrite(enB, duty);
    if (dir) {
        digitalWrite(R, HIGH);
        digitalWrite(L, LOW);

        digitalWrite(in3, HIGH);
        digitalWrite(in4, LOW);
    } else {
        digitalWrite(R, LOW);
        digitalWrite(L, HIGH);

        digitalWrite(in3, LOW);
        digitalWrite(in4, HIGH);
    }
}

int my_getnbr(char *str) {
    int result;

```

```
int puiss;

result = 0;
puiss = 1;
while (('-' == (*str)) || ((*str) == '+')) {
    if (*str == '-')
        puiss = puiss * -1;
    str++;
}
while ((*str >= '0') && (*str <= '9')) {
    result = (result * 10) + ((*str) - '0');
    str++;
}
return (result * puiss);
}
```