Министерство Образования и Исследований Республики Молдова

Технический Университет Молдовы

Факультет Вычислительной Техники, Информатики и Микроэлектроники

Департамент Программной Инженерии и Автоматики

**Лабораторная работа №3**

по предмету «IOT»

Выполнил: ст. гр. TI-196

М. Годунов

Проверил: А. Бырназ

Кишинев 2022

**ЛЛ3**

**Сбор данных. Датчики.**

**Определение проблемы:**

Создать приложение на основе микроконтроллера, которое будет принимать сигнал от источника сигнала, настраивать сигнал и отображать физический параметр на терминале (ЖК-дисплей и / или последовательный порт). Каждый студент выберет аналоговый или цифровой (не двоичный) датчик из прикрепленного PDF-файла или сайта: http://www.37sensors.com/.

1. Получить сигнал от датчика;

2. Кондиционировать сигнал с помощью цифровых фильтров и других методов;

3. Отображать данные на ЖК-дисплее и / или через последовательный порт.

**Ход работы**

Схема для Analog Joystick плюс MAX7219 показана на рисунке 1.

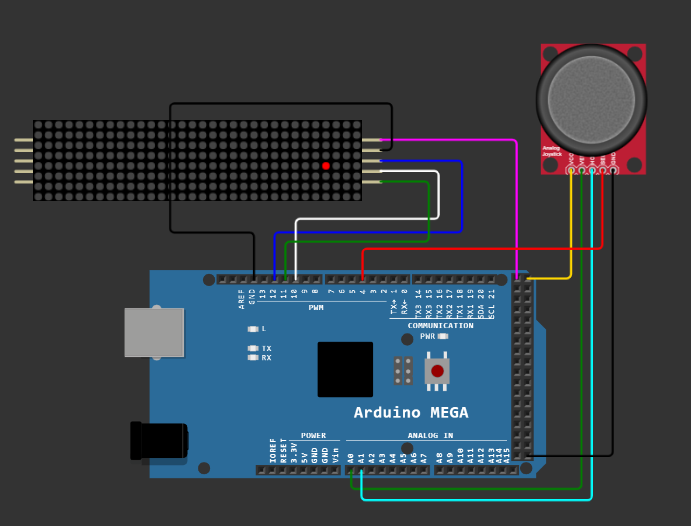


Рисунок 1 – схема.

Результат работы показан на рисунках 2, 3.

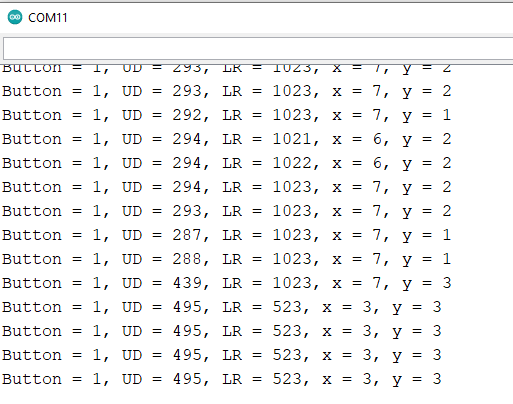
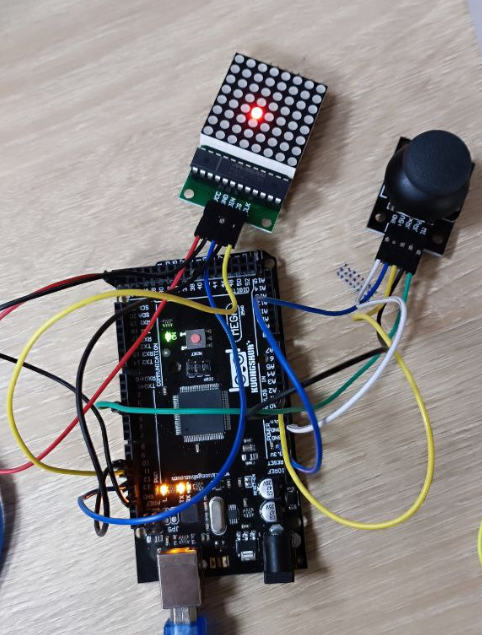


Рисунок 2 – Serial Monitor.



Рисунок 3 – Serial Plotter.

Фотография реальной схемы.



**Вывод**:

В данной лабораторной работе мы познакомились с тремя видами датчиков: Digital, Analog, Temporal. Для своей лабораторной работы я выбрал Analog датчик – Analog Joystick. В качестве дополнения я подключил LED матрицу MAX7219 для визуализации изменений.

Исходный код:

*#include <stdio.h>*

*#include <stdlib.h>*

*#include <LedControl.h>*

*int UD;*

*int LR;*

*int button=4;*

*static FILE vartio = {0};*

*/\**

*pin 12 is connected to the DataIn*

*pin 11 is connected to the CLK*

*pin 10 is connected to LOAD or CS*

*We have only a single MAX7219.\*/*

*LedControl lc = LedControl(12,11,10,1);*

*int ft\_write(char c, FILE \*f)*

*{*

*Serial.write(c);*

*return 0;*

*}*

*int ft\_read(FILE \*vartio){*

*while(!Serial.available());*

*return Serial.read();*

*}*

*void setup() {*

*// put your setup code here, to run once:*

*Serial.begin(9600);*

*Serial.println("Hello Arduino\n");*

*fdev\_setup\_stream(&vartio, ft\_write, ft\_read, \_FDEV\_SETUP\_RW);*

*stdin = &vartio;*

*stdout = &vartio;*

*lc.shutdown(0,false); //turn off power saving, enables display*

*lc.setIntensity(0,8); //brighttness 0-15*

*lc.clearDisplay(0);*

*pinMode(button, INPUT);*

*digitalWrite(button, HIGH);*

*}*

*void loop() {*

*// put your main code here, to run repeatedly:*

*UD = analogRead(A0); // analog value at A1 as UD;*

*LR = analogRead(A1); // analog value at A0 as LR*

*int state = digitalRead(button); // determines tha state of button (HIGH or LOW)*

*//scales down the value from 0-1023 to 0-7*

*char x\_translate = map(LR, 0, 1023, 0, 7);*

*char y\_translate = map(UD, 0, 1023, 0, 7);*

*printf("Button = ");*

*Serial.print(state, DEC);*

*printf(", UD = ");*

*Serial.print(UD, DEC);*

*printf(", LR = ");*

*Serial.print(LR, DEC);*

*printf(", x = ");*

*Serial.print(x\_translate, DEC);*

*printf(", y = ");*

*Serial.print(y\_translate, DEC);*

*printf("\n");*

*//clears display whenever a new value is updated*

*lc.clearDisplay(0);*

*lc.setLed(0, x\_translate, y\_translate, true);*

*//if button is pressed, LED is OFF and vice versa*

*if(state = 1){*

*lc.setLed(0,x\_translate, y\_translate, true);*

*}*

*else{*

*lc.setLed(0,x\_translate, y\_translate, false);*

*}*

*delay(150);*

*}*