

Создание калькулятора на базе платы Arduino

Терехов Захар Станиславович

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье описан процесс создания калькулятора с цифровой панелью. Для создания используется плата Arduino, переменный резистор, цифровая панель 4 на 4 и LCD панель. Созданный макет позволяет проводить вычисления с операциями сложение, умножение, вычитание и деление как на обычном калькуляторе.

Ключевые слова: Arduino, LCD панель, цифровая клавиатура

Creating a calculator based on the Arduino board

Terekhov Zakhar Stanislavovich

Sholom-Aleichem Priamursky State University

student

Abstract

This article describes the process of creating a calculator with a digital panel. For creation, the Arduino board, a variable resistor, a 4 by 4 digital panel and an LCD panel are used. The created layout allows you to perform calculations with the operations of addition, multiplication, subtraction and division as on a conventional calculator.

Keywords: Arduino, LCD panel, numeric keypad

Клавиатуры позволяют пользователям вводить данные во время работы программы. Часто клавиатура необходима для ввода данных в систему Arduino, а мембранные клавиатуры являются экономичным решением для многих приложений. Они довольно тонкие и могут быть легко установлены в любом месте.

Цель исследования – создать калькулятор с цифровой панелью на базе платы Arduino.

Ранее этим вопросом интересовались А.В. Пилипенко, А.П. Пилипенко развивали тему «Программное обеспечение для автоматизации помещений на платформе arduino» [1] в которой обсуждается программа для проектов автоматизации жилых зданий на базе платформ Arduino Uno или Arduino Mega, которая позволяет реализовать сбор и обработку данных, выдачу рекомендаций на светодиодные панели, управление кондиционером и вентиляцией, ведение отчетности о количестве срабатываний, управление запорными механизмами и кодовыми замками, управление осветительными

и гидравлическими приборами. М. Нидилько, Д. Митарова с темой «Разработка проекта на микроконтроллерной плате» [2], а подробнее про различные свободные программные продукты, которые используются в разных областях и сферах человеческой деятельности. Сфера программирования не является исключением. Так, к примеру, существуют свободное ПО для программирования микроконтроллерных плат. Остановимся подробнее на том, что же такое свободное программное обеспечение, микроконтроллерные платы и как они связаны между собой, а также какие проекты можно собрать из этого сочетания. А.В. Пилипенко, О.В. Пилипенко, А.П. Пилипенко опубликовали статью «Разработка модульной системы автоматического управления помещением» [3] рассказали, как решить проблему внедрения автоматических и автоматизированных систем управления помещением с целью повышения энергоэффективности помещения и благосостояния владельцев. Авторы рассказывают о сложности выбора аппаратного обеспечения, используемого для разработки автоматизированных систем, и приводят пример разработанной модульной автоматической системы. Авторами представляются алгоритмы, исходные коды на языке программирования C++ и отладочные схемы подсистем, а также результаты внедрения, которые и подтверждают актуальность разработки подобных систем управления.

Для этого потребуется:

- Плата Arduino
- Макетная плата
- Соединительные провода
- Цифровая панель 4 на 4
- LCD панель 16 на 2
- Переменный резистор на 10 кОм

Схема подключения представлена на рисунке 1.

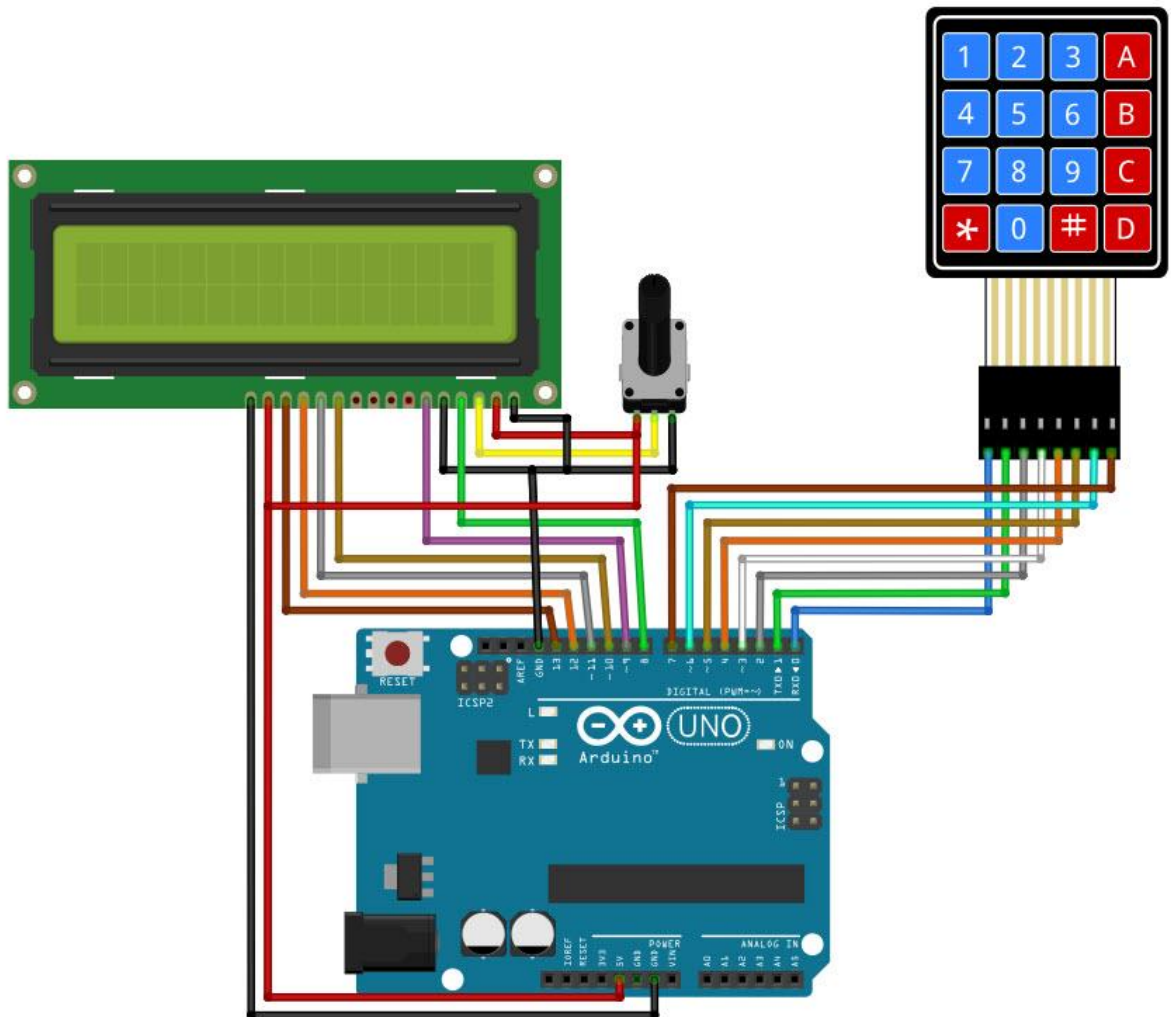


Рис. 1 Схема подключения к плате Arduino

Кнопки на клавиатуре расположены в строках и столбцах. Клавиатура 4X4 имеет 4 строки и 4 столбца:

Под каждым ключом находится мембранный переключатель. Каждый переключатель в ряду соединен с другими переключателями под подушкой. Каждый переключатель в столбце подключен одинаково - одна сторона переключателя соединена со всеми другими переключателями в этом столбце. Каждая строка и столбец выводятся на один контакт, всего 8 контактов на клавиатуре 4X4:

Нажатие кнопки замыкает переключатель между столбцом и трассировкой строки, позволяя току протекать между контактом столбца и контактом строки.

```
#include <LiquidCrystal.h>
#include <Keypad.h>

const byte Row_Id = 4;
const byte Cols_Id = 4;

char kbtns_matrix[Row_Id][Cols_Id] = {
    {'7', '8', '9', 'D'},
```

```
{'4', '5', '6', 'C'},
{'1', '2', '3', 'B'},
{'*', '0', '#', 'A'}
};

byte PinRow[Row_Id] = { 0, 1, 2, 3 };
byte PinCol[Cols_Id] = { 4, 5, 6, 7 };

Keypad KeyPd = Keypad( makeKeymap(kbtns_matrix), PinRow, PinCol,
Row_Id, Cols_Id );

const int rs = 8, en = 9, d4 = 10, d5 = 11, d6 = 12, d7 = 13;
LiquidCrystal display(rs, en, d4, d5, d6, d7);

long Number1, Number2, Nbr;
char kbtn, act;
boolean result = false;

void setup() {
}

void loop() {

    kbtn = KeyPd.getKey();

    if (kbtn != NO_KEY)
        DetectButtons();

    if (result == true)
        CalculateResult();

    DisplayResult();
}

void DetectButtons()
{
    display.clear();
    if (kbtn == '*')
    {
        Serial1.print("Button Cancel");
        Nbr = Number1 = Number2 = 0;
        result = false;
    }

    if (kbtn == '1')
    { Serial1.print("Button 1");
      if (Nbr == 0)
          Nbr = 1;
      else
          Nbr = (Nbr * 10) + 1;
    }

    if (kbtn == '4')
    { Serial1.print("Button 4");
      if (Nbr == 0)
          Nbr = 4;
      else
          Nbr = (Nbr * 10) + 4;
    }
}
```

```
}

if (kbtn == '7')
{ Seriall.print("Button 7");
  if (Nbr == 0)
    Nbr = 7;
  else
    Nbr = (Nbr * 10) + 7;
}

if (kbtn == '0')
{ Seriall.print("Button 0");
  if (Nbr == 0)
    Nbr = 0;
  else
    Nbr = (Nbr * 10) + 0;
}

if (kbtn == '2')
{ Seriall.print("Button 2");
  if (Nbr == 0)
    Nbr = 2;
  else
    Nbr = (Nbr * 10) + 2;
}

if (kbtn == '5')
{ Seriall.print("Button 5");
  if (Nbr == 0)
    Nbr = 5;
  else
    Nbr = (Nbr * 10) + 5;
}

if (kbtn == '8')
{ Seriall.print("Button 8");
  if (Nbr == 0)
    Nbr = 8;
  else
    Nbr = (Nbr * 10) + 8;
}

if (kbtn == '#')
{ Seriall.print("Button Equal");
  Number2 = Nbr;
  result = true;
}

if (kbtn == '3')
{ Seriall.print("Button 3");
  if (Nbr == 0)
    Nbr = 3;
  else
    Nbr = (Nbr * 10) + 3;
}
```

```
if (kbtn == '6')
{ Seriall.print("Button 6");
  if (Nbr == 0)
    Nbr = 6;
  else
    Nbr = (Nbr * 10) + 6;
}

if (kbtn == '9')
{ Seriall.print("Button 9");
  if (Nbr == 0)
    Nbr = 9;
  else
    Nbr = (Nbr * 10) + 9;
}

if (kbtn == 'A' || kbtn == 'B' || kbtn == 'C' || kbtn == 'D')
{
  Number1 = Nbr;
  Nbr = 0;
  if (kbtn == 'A')
  {
    Seriall. ("Addition");
    act = '+';
  }
  if (kbtn == 'B')
  {
    Seriall.print("Subtract");
    act = '-';
  }
  if (kbtn == 'C')
  {
    Seriall.print("Multiplication");
    act = '*';
  }
  if (kbtn == 'D')
  {
    Seriall.print("Devesion");
    act = '/';
  }

  delay(100);
}

}

void CalculateResult()
{
  if (act == '+')
    Nbr = Number1 + Number2;

  if (act == '-')
    Nbr = Number1 - Number2;

  if (act == '*')
    Nbr = Number1 * Number2;

  if (act == '/')
```

```
    Nbr = Number1 / Number2;
}

void DisplayResult()
{
    display.setCursor(0, 0);
    display.print(Number1); display.print(act); display.print(Number2);

    if (result == true)
    {
        display.print(" =");
        display.print(Nbr);
    }

    display.setCursor(0, 1);
    display.print(Nbr);
}
```

Результат работы можно увидеть на рисунке 2.

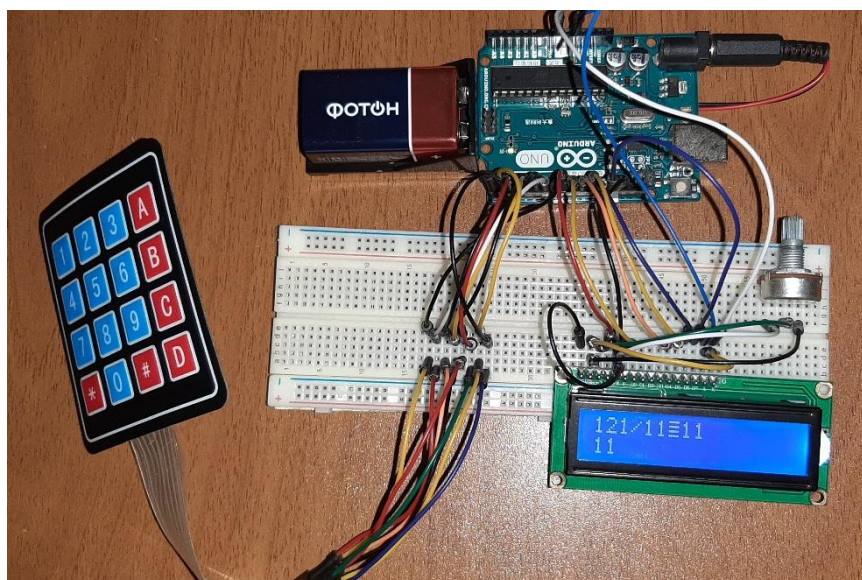


Рис. 2 Схема в собранном состоянии

Вывод

Результатом статьи стал работающий калькулятор на Arduino имитирующий работу настоящего калькулятора. Концепция работы с подобными клавиатурами позволяет делать системы безопасности и любые другие схемы, когда данные вводятся самим пользователем.

Библиографический список

1. Пилипенко А.В., Пилипенко А.П. Программное обеспечение для автоматизации помещений на платформе arduino // Свидетельство о регистрации программы для ЭВМ RU 2015662105, 17.11.2015. Заявка № 2015618867 от 28.09.2015. URL: <https://elibrary.ru/item.asp?id=39339631> (Дата обращения: 07.01.2020)
2. Нидилько М., Митарова Д. Разработка проекта на микроконтроллерной

- плате // В сборнике: Свободное программное обеспечение в высшей школе Сборник тезисов Четырнадцатой конференции . Ответственный редактор В.Л. Черный. 2019. С. 36-38. URL: <https://elibrary.ru/item.asp?id=37016847> (Дата обращения: 07.01.2020)
3. Пилипенко А.В., Пилипенко О.В., Пилипенко А.П. Разработка модульной системы автоматического управления помещением // Промышленные АСУ и контроллеры. 2015. № 10. С. 32-39. URL: <https://elibrary.ru/item.asp?id=24343712> (Дата обращения: 07.01.2020)