

 free_arduino 27 января 2015 в 14:07

Паяльная станция своими руками на базе Arduino

DIY или Сделай сам

Из песочницы

Всем привет! Как-то я затронул тему паяльной станции на Arduino и сразу меня завалили вопросами (как/где/когда). Учитывая массовость запросов, я решил написать обзор простой паяльной станции (только паяльник) на базе Arduino.

Почему Arduino? Ведь существует уйма контроллеров быстрее и дешевле. В таких случаях я обычно отвечаю: — *Дёшево, практично, быс*

Действительно, ведь Arduino Pro Mini сейчас стоит 1,63\$ за 1 шт (недавно прислали), а atmega8 стоит 1\$ (оптовая цена). Получается, что п. Pro Mini с обвесом (кварц, конденсаторы, стабилизаторы) стоит не так-то и дорого, плюс ко всему экономит время. Также время очень сильно экономит IDE-оболочка для Arduino, легко и быстро в ней справляется даже школьник. Учитывая популярность и дешевизну я решил собрать именно на Arduino.

Для создания паяльной станции нам первым делом нужна ручка паяльной станции, зачастую это китайские станции типа 907 A1322 939.



Начнём

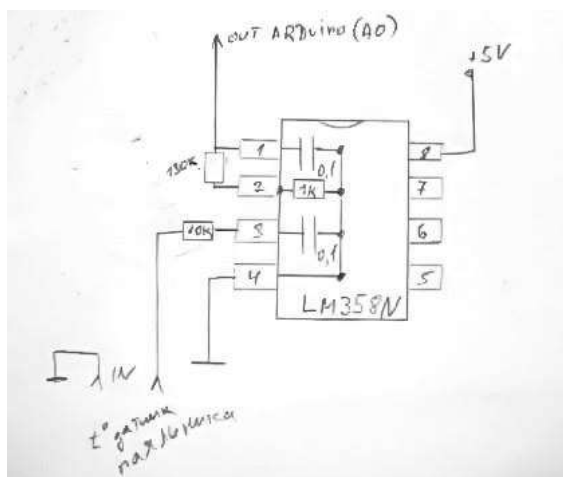
Характеристики ручки:

Напряжение: 24V DC

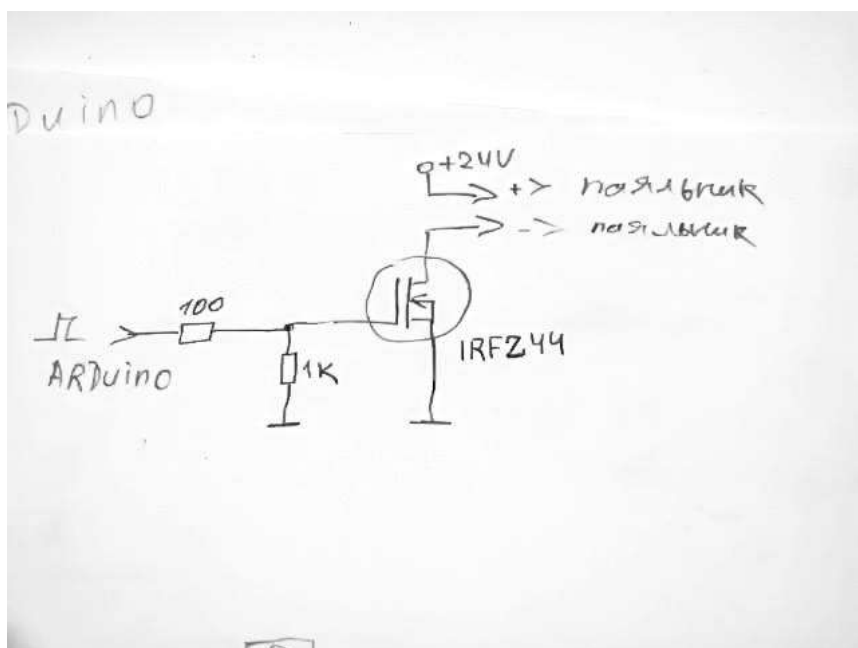
Мощность: 50W (60W)

Температура: 200°C~ 480°C

Для управления ручкой паяльника нам первым делом нужно снимать данные с датчика температуры, в этом нам поможет **LM358N**. Эта схема работает у меня почти 2 года.

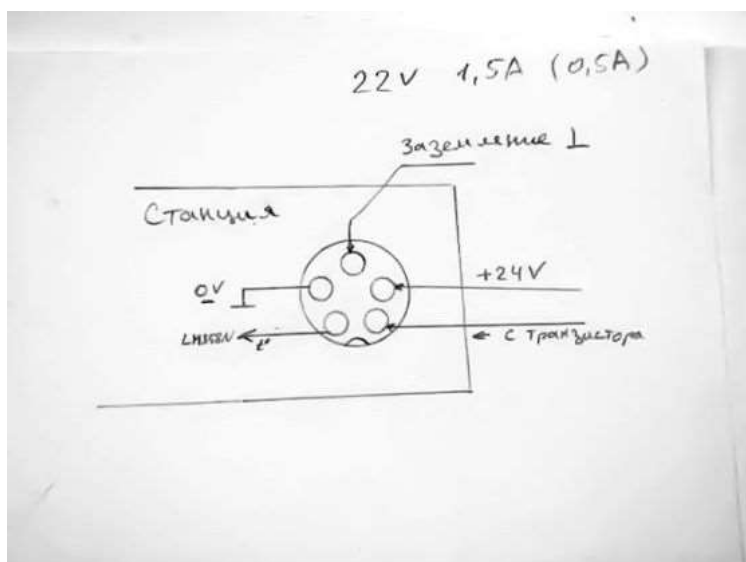


Далее нам нужно управлять(включать и выключать) нагревательный элемент паяльника, в этом на поможет импульсный транзистор **IRFZ44** подключение очень простое:



Хочу обратить Ваше внимание на будущий режим работы нагревательного элемента. Его мы будем включать в три этапа путём ШИМ-моду. При старте программы будет включаться почти максимальная мощность (скважность 90%), при приближении к заданной температуре мощность понижается (скважность 35-45%), и при минимальной разнице между текущей и заданной температуры мощность держится на минимуме (скважность 30-35%). Таким образом мы устраняем инерцию перегрева. Повторюсь, паяльная станция стабильно работает почти 2 года, и термоэлемент не находится в постоянной предельной нагрузке (что продлевает его жизнь). Все настройки в программе можно отредактировать.

Подключать ручку нужно по схеме:



Обратите внимание, разъём на панели станции, а не на ручке.

Очень настаиваю: проверяйте ручки перед пуском, раскрутите и проверьте целостность нагревательного элемента, а также правильность распылки проводов на разъёме.

Далее нам нужен контроллер. Для демонстрации я выбрал Arduino Uno – как самый популярный и удобный. Заметьте, что паяльную станцию делаю блочной и это даёт возможность самому выбрать контроллер. Также нам нужны две кнопки подтянутые к +5В сопротивлениям 10кОм и сегментный индикатор на три разряда. Выводы сегментов я подключил через сопротивления 100 Ом.

ANODES:

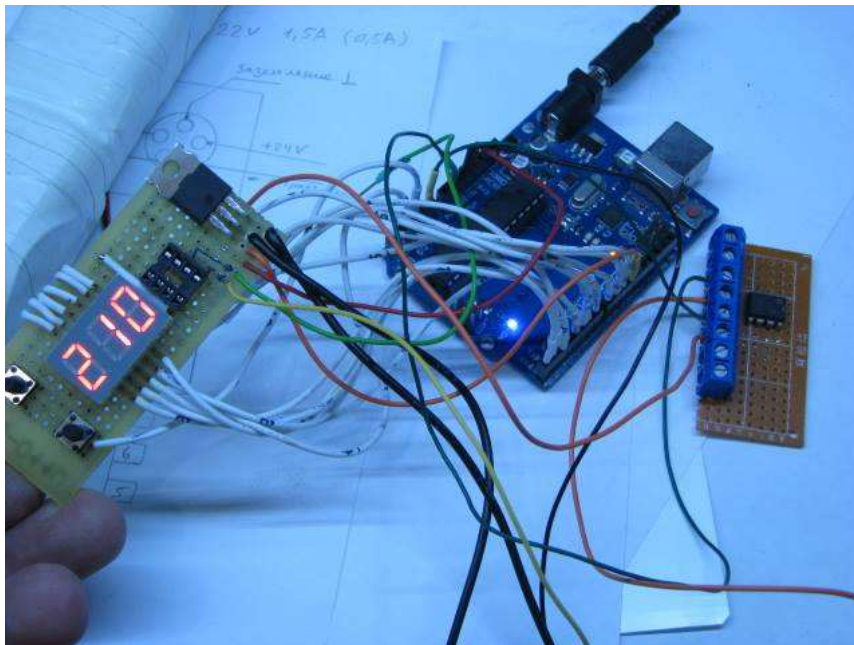
D0 — a
D1 — b
D2 — c
D3 — d
D4 — e
D5 — f
D6 — g
D7 — dp (точка)

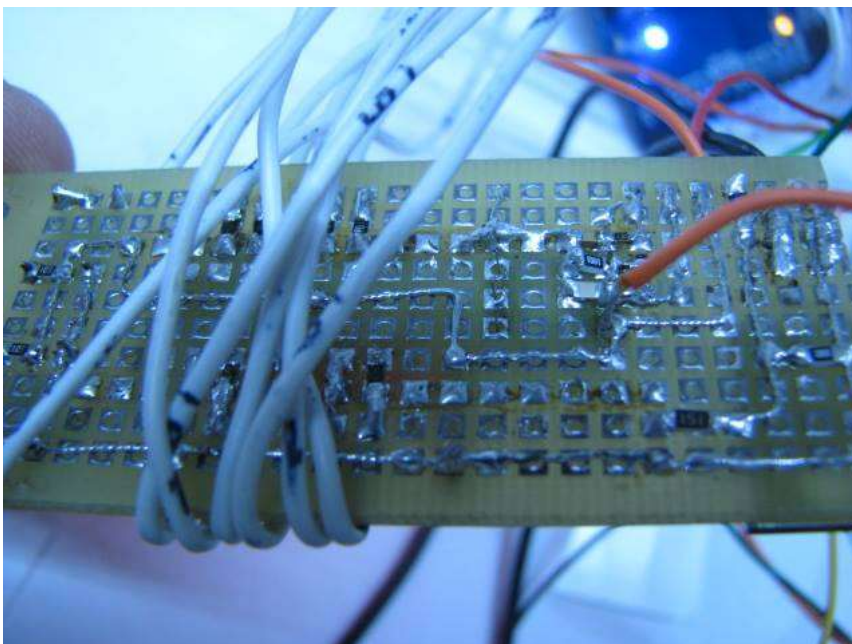
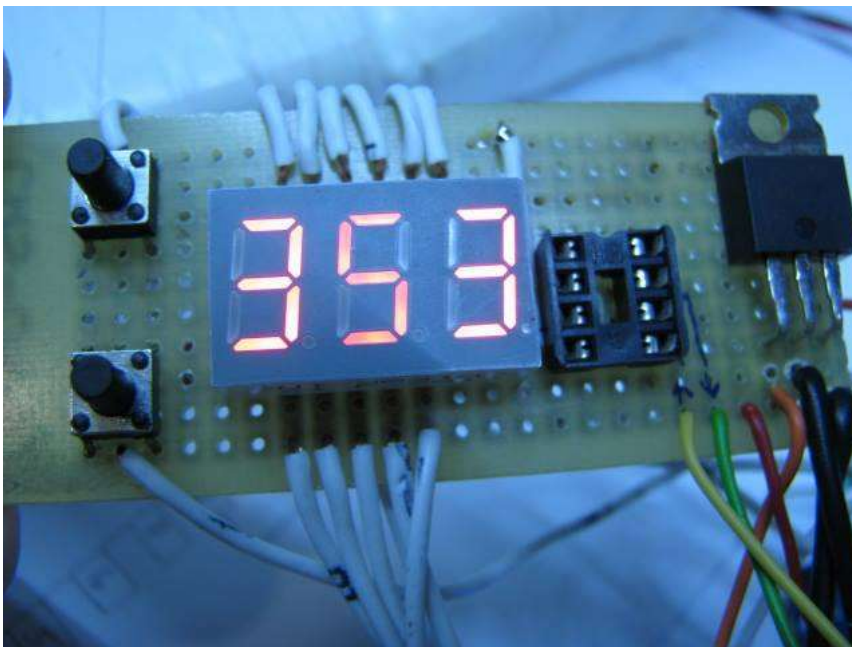
CATHODES:

D8 — cathode 3
D9 — cathode 2
D10 — cathode 1

Хочу также заметить, что кнопки мы сажаем на аналоговые пины 3 и 2. И в программе я их опрашиваю как аналог. Сделал я это для того, чтобы не вводить в заблуждение молодое поколение. Не каждый знает где найти пин 14, 15 и 16. А учитывая, что скорости достаточно и памяти в контроллере много, то так будет проще.

Давайте посмотрим что получилось:





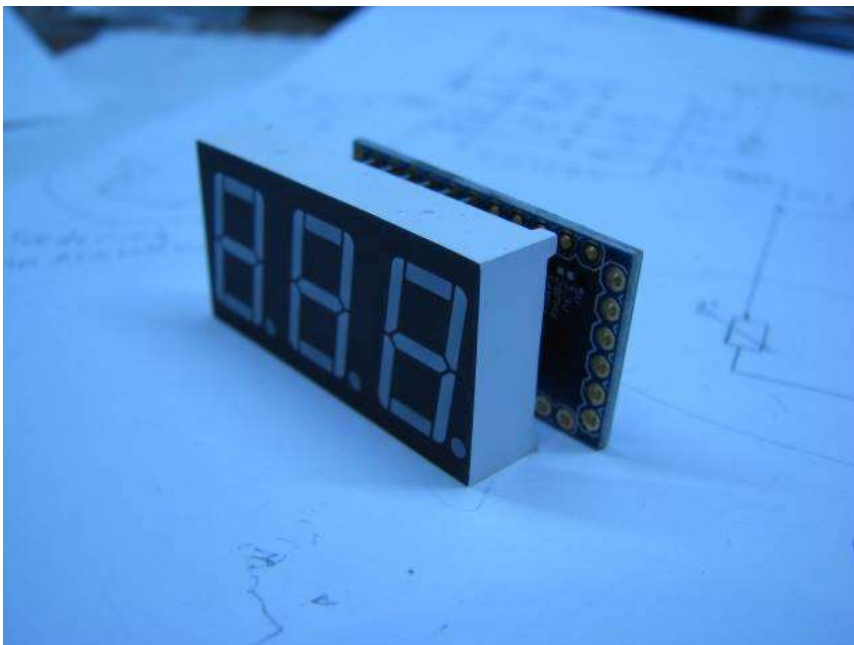
Вы можете заметить пустую панельку возле индикатора, это заготовка под LM358N, просто аналог KA358 показал плохие результаты в раб. Поэтому я воспользовался блоком термодатчиков на LM358N для паяльной станции с феном.

 image

 image

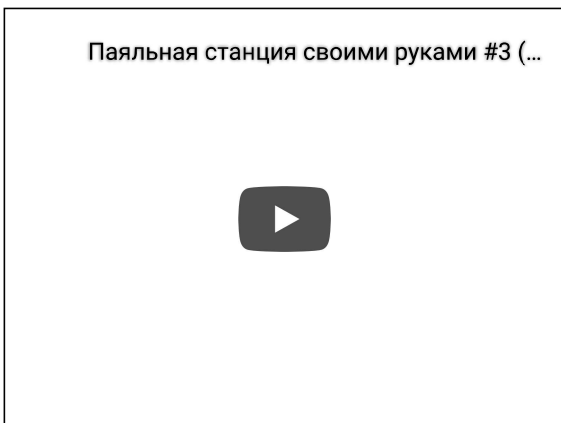
Далее необходимо выбрать источник питания. Я взял блок питания от какого-то ноутбука на 22V 3A, его хватает с запасом. Потребление при старте паяльника 1,5 A а при поддержке температуры 0,5A. Поэтому выбирайте себе подходящий блок питания, желательно 24V DC 2A.

На фото выше видно жмут проводов и многих это пугает. Поймите, это демо, вариант под любой контроллер, станцию можно собрать и компактно, к примеру:



 image

Это наглядный пример для реализации Вашего проекта паяльной станции. Видео, которое наглядно поможет понять Вам как собрать самс



Вот тест программы, писал под версией IDE 1.5.2. Учтите всё вышесказанное и сильно не критикуйте (программу пытался написать просто доступно).

```
/*  
  
// Пины подключения индикаторов  
ANODES:  
D0 - a  
D1 - b  
D2 - c  
D3 - d  
D4 - e  
D5 - f  
D6 - g  
D7 - dp (digital point)  
  
      a  
      *  
f *      * b  
  * g *  
      *  
e *      * c  
  * d *  
      * # dp  
  
CATHODES:
```

```

D8 - cathode 3
D9 - cathode 2
D10 - cathode 1
*/

// ----- не изменять, это для Сегментов -----
byte const digits[] = {
    B00111111, B00000110, B01011011, B01001111, B01100110, B01101101, B01111101, B00000111, B01111111, B01101111};

int digit_common_pins[]={8,9,10}; // пины для разрядов сегментов(при изменении убедитесь что Ваш порт не используется)
int refresh_delay = 5;
int count_delay = 300; // COUNTING SECONDS IF count_delay = 1000
long actual_count_delay = 0;
long actual_refresh_delay = 0;
int increment = 0; //Стартовое значение на сегментах
int max_digits=3; // Кол-во знаков
int current_digit=max_digits-1;
int increment_max = pow(10,max_digits);
// ----- не изменять, это для Сегментов -----

//----- переменные паяльника -----
int knup = 3; //Пин кнопки вверх in(красный светодиод)
int kn dn = 2; //Пин кнопки вниз in(синий светодиод)

int nagr = 11; // пин вывода нагревательного элемента(через транзистор)
int tin = 0; // Пин Датчика температуры IN Analog через LM358N
int tdat = 0; //Переменная Датчика температура
int ustt = 210; // Выставленная температура по умолчанию (+ увеличение и уменьшение при нажатии кнопок)
int mintemp = 140; // Минимальная температура
int maxtemp = 310; // Максимальная температура
int nshim = 0; // Начальное значение шим для нагрузки

void setup(){

    pinMode(nagr,OUTPUT); // Порт нагрузки (паяльника) настраиваем на выход
    analogWrite(nagr, nshim); //Вывод шим в нагрузку паяльника (выводим 0 - старт с выключенным паяльником - пока не опр
лим состояние температуры)

    // ----- не изменять, это для Сегментов -----
    DDRD = B11111111;
    for (int y=0;y<max_digits;y++)
    {
        pinMode(digit_common_pins[y],OUTPUT);
        digitalWrite(digit_common_pins[y], HIGH);
    }
    // ----- не изменять, это для Сегментов -----

}

void loop() {
    show(increment); // Вывести значение переменной на экран (LED)

    if (tdat < ustt ){ // Если температура паяльника ниже установленной температуры то:

        if ((ustt - tdat) < 16 & (ustt - tdat) > 6 ) // Проверяем разницу между установленной температурой и текущей т
ьника,
            // Если разница меньше 10 градусов, то
            {
                nshim = 99; // Понижаем мощность нагрева (шим 0-255, мы делаем 99) - таким образом мы убирае
инерцию перегрева
            }
        else if ((ustt - tdat) < 7 & (ustt - tdat) > 3)
            {
                nshim = 80; // Понижаем мощность нагрева (шим 0-255, мы
делаем 99) - таким образом мы убираем инерцию перегрева
            }

        else if ((ustt - tdat) < 4 )

```

```

        {
            nshim = 45; // Понижаем мощность нагрева (шим 0-255, мы делаем 99)
        }

        таким образом мы убираем инерцию перегрева

        else {
            nshim = 230; // Иначе поднимаем мощность нагрева (шим 0-255, мы делаем 230) на максимум
            быстрого нагрева до нужной температуры
        }

        analogWrite(nagr, nshim); //Вывод в шим порт (на транзистор) значение мощности
    }
    else { //Иначе (если температура паяльника равняется или выше установленной)

        nshim = 0; // Выключаем мощность нагрева (шим 0-255 мы делаем 0) - таким образом мы отключаем паяльник
        analogWrite(nagr, nshim); //Вывод в шим порт (на транзистор) значение мощности
    }

    if(millis() - actual_count_delay > count_delay) // это для сегментов

    {
        actual_count_delay = millis();
        // Здесь мы пишем нашу прогу по считыванию состояния кнопок (это место в счетчик не будет тормозить вывод на сегменты

        tdat = analogRead(tin); // Считать состояние датчика температуры и присвоить tdat
        tdat =map(tdat,0,430,25,310); // калибровка п умолчанию 0,430,25,310
        increment = tdat; // присвоить текущее значение температуры переменной сегмента

        if (analogRead(kndn) < 1) // Если нажата синяя кнопка, то понизить температуру на 5
        {
            if( ustt <= mintemp || (ustt-5) <= mintemp )
            {
                ustt= mintemp;
                increment = ustt;
            }

            else {
                ustt=ustt-5;
                increment = ustt;
            }

        }

        else if (analogRead(knup) < 1) // Если нажата красная кнопка, то повысить температуру на 5
        {
            ustt=ustt+5;
            if( ustt >=maxtemp)
            {
                ustt= maxtemp;
            }

            increment = ustt;
        }

    }

}

}

void show(int value) { //----- подпрограмма для вывода на сегменты - лучше не изменять -----
    -----
    int digits_array[]={};
    int y=0;
    boolean empty_most_significant = true;

    if(millis() - actual_refresh_delay >= refresh_delay)
    {

```



```

for (int z=max_digits-1;z>=0;z--)
{

    digits_array[z] = value / pow(10,z); //rounding down by converting from float to int

    if(digits_array[z] != 0 ) empty_most_significant = false; // DON'T SHOW LEADING ZEROS

    value = value - digits_array[z] * pow(10,z);


    if(z==current_digit)
    {
        if(!empty_most_significant || z==0){ // DON'T SHOW LEADING ZEROS EXCEPT FOR THE LEAST SIGNIFICANT

            PORTD = digits[digits_array[z]];

        }

        else
        {
            PORTD = B00000000;
        }

        digitalWrite(digit_common_pins[z], LOW);
    }else{
        digitalWrite(digit_common_pins[z], HIGH);
    }

}

    current_digit--;
    if(current_digit < 0)
    {
        current_digit= max_digits; // NEED AN EXTRA REFRESH CYCLE TO CLEAR ALL DIGITS
    }

    actual_refresh_delay =  millis();
}

}

```

Очень надеюсь, что Вам это как-то поможет в создании своего проекта.

Tags: arduino, uno, паяльник, паяльная станция, своими руками

↑ +38 ↓ 549 👁 176k 💬 73



19,0
Карма

0,0
Рейтинг

43
Подписчики

@free_arduino
Пользователь

Поделиться публикацией

ПОХОЖИЕ ПУБЛИКАЦИИ

5 октября 2016 в 18:41

Управление шлагбаумом с помощью Arduino UNO и радиопередатчика 433 МГц

↑ +21 👁 76,5k 📖 174 💬 33