

УДК 004.42

## АВТОМАТИЗИРОВАННОЕ УДАЛЕННОЕ УПРАВЛЕНИЕ УСТРОЙСТВАМИ ПРИ ПОМОЩИ АРДУИНО

Столяров В. Г., Голубев Л. П.

Киевский национальный университет технологий и дизайна

*В статье рассматриваются возможности автоматизированного удаленного управления устройствами при помощи Ардуино. Приведен пример создания автоматизированной системы удаленного управления температурой и влажностью при помощи микропроцессорной системы Ардуино Уно.*

**Ключевые слова:** Ардуино, Wi-Fi, Bluetooth, микропроцессор, автоматизация, система управление, температура, влажность

Сегодня микропроцессорное управление является доминирующим в системах автоматизированного управления. Огромное число созданных систем управления, работающих на предприятиях и в быту является тому подтверждением. Разработка протоколов удаленного доступа к устройствам, таких как Ethernet, Bluetooth и Wi-Fi, значительно увеличило область применения подобных систем. Теперь появилась возможность управлять устройством, находясь на значительном расстоянии от него [8, 9].

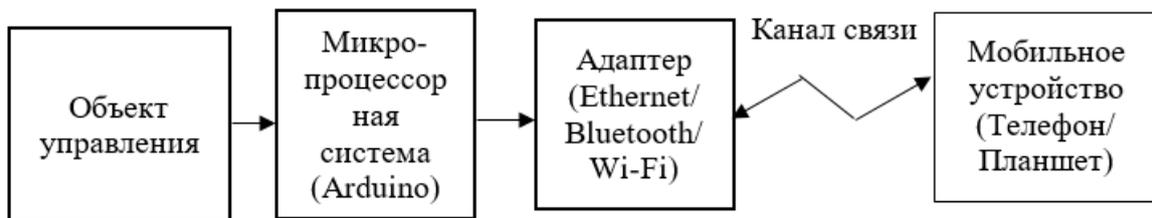


Рис. 1. Схема автоматизированной системы удаленного управления

В соответствии с этой схемой (рис. 1), с помощью мобильного устройства можно как получать информацию с устройства, так и передавать ее.

### **Постановка задачи**

Поэтому возникает задача в разработке программных средств, позволяющих осуществлять удаленный обмен информацией с устройством не зависимо от выбранного канала связи (Ethernet, Bluetooth, Wi-Fi). Кроме этого, необходимо разработать интерфейс пользователя мобильного устройства.

**Результаты исследования**

В результате наших исследований разработана автоматизированная система удаленного контроля и управления температурой и влажностью помещения. В ходе проектирования системы был разработан интерфейс мобильного устройства под управление операционной системы Андроид [3, 7], позволяющий наблюдать показания датчика DHT-11 и включать при необходимости либо вентилятор, либо нагревательный элемент.

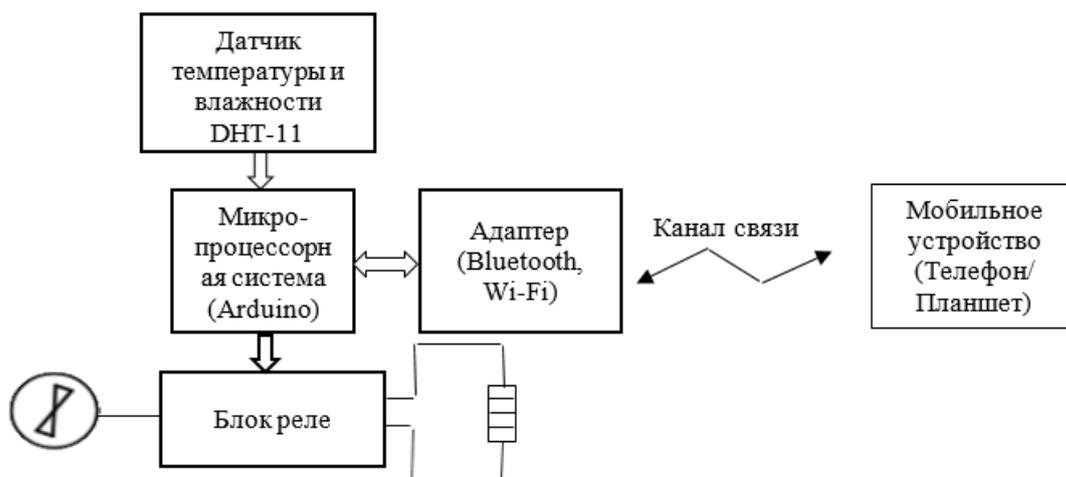


Рис. 2. Структурная схема автоматизированной системы удаленного управления температурой и влажностью

Рассмотрим более детально основные компоненты схемы:

Микропроцессорная система Arduino Uno

Arduino Uno – это устройство на основе микроконтроллера ATmega328 (datasheet). В его состав входит все необходимое для удобной работы с микроконтроллером: 14 цифровых входов/выходов (из них 6 могут использоваться в качестве ШИМ-выходов), 6 аналоговых входов, кварцевый резонатор на 16 МГц, разъем USB, разъем питания, разъем для внутрисхемного программирования (ICSP) и кнопка сброса. Для начала работы с устройством достаточно просто подать питание от AC/DC-адаптера или батарейки, либо подключить его к компьютеру посредством USB-кабеля [4, 5].

## Характеристики

Микроконтроллер	ATmega328
Рабочее напряжение	5В
Напряжение питания (рекомендуемое)	7-12В
Напряжение питания	6-20В

(предельное)	
Цифровые входы/выходы	14 (из них 6 могут использоваться в качестве ШИМ-выходов)
Аналоговые входы	6
Максимальный ток одного вывода	40 мА
Максимальный выходной ток вывода 3.3V	50 мА
Flash-память	32 КБ (ATmega328) из которых 0.5 КБ используются загрузчиком
SRAM	2 КБ (ATmega328)
EEPROM	1 КБ (ATmega328)
Тактовая частота	16 МГц

На рис. 3 представлен внешний вид Bluetooth модуля HC-06. Bluetooth модуль HC-06 – приемно-передатчик с антенной, на подложке, интерфейс RS232, построенный на чипе BC417 компании Cambridge Silicon Radio.



Рис. 3. Bluetooth модуль HC-06

Bluetooth модуль позволяет подключать модули Arduino к телефонам, PDA, наладонникам, смартфонам, ноутбукам, и любым другим устройствам, имеющим Bluetooth, работающий в режиме master. Позволяет передавать данные на Arduino контроллер через стандартный интерфейс RS-232.

Для использования Bluetooth модуля нужно сначала на его основе собрать прототип: подключить питание, контроллер, устройство с master Bluetooth. Потом нужно записать в Arduino контроллер специальную программу. Потом на компьютере нужно создать COM порт по Bluetooth соединению. После этого понадобится программа-терминал для отправки данных через COM порт по Bluetooth соединению,

например, бесплатная Termite. Потом нужно провести настройку и тестирование модуля с помощью AT-команд и начинать работу. Также есть возможность подключения к Linux и Android. При подключении к модулю питания, на корпусе должен загореться красный светодиод. Корпус модуля обтянут пленкой для защиты от статического электричества и загрязнения.

Управление Bluetooth модулем осуществляется из устройства master Bluetooth (телефон, PDA, ноутбук и проч.). Данные передаются по стандарту UART, который вшит в большинстве встраиваемых систем (этот порт имеется почти во всех микроконтроллерах, либо легко организуется программно). Также возможно управление с помощью AT-команд.

Bluetooth модуль с антенной имеет один 4-х пиновый штыревой интерфейс.

Обозначение контактов:

- VCC (напряжение питания);
- GND (общий контакт);
- RXD (принимаемые данные);
- TXD (передаваемые данные).

Также на корпусе есть две клеммы, которые по умолчанию не задействованы, но могут использоваться:

- STATE – сюда дублируется сигнал с встроенного светодиода, когда модуль активен светодиод мигает, когда связь установлена – горит;
- EN – включение/выключение, если подать на этот контакт логическую единицу (5 В), то модуль выключится, если логический ноль (или пин будет не задействован), то модуль будет работать.

Для подключения модуля к Ардуино необходимо соединить контакты устройств в соответствии с таблицей 1:

Таблица 1

Bluetooth HC-05/06	Arduino
VCC	+5V
GND	GND
TX	RX
RX	TX

Внешний вид устройства представлен на рис. 4. Миниатюрный WiFi модуль на базе новейшей микросхемы **ESP8266** со встроенным стеком протокола TCP/IP и управлением AT-командами. Чип создан для использования в устройствах автоматизации – в умных розетках, IP-камерах, беспроводных сенсорах, носимой электронике и так далее. Одним словом, ESP8266 появился на свет, чтобы стать мозгом грядущего «Интернета вещей».

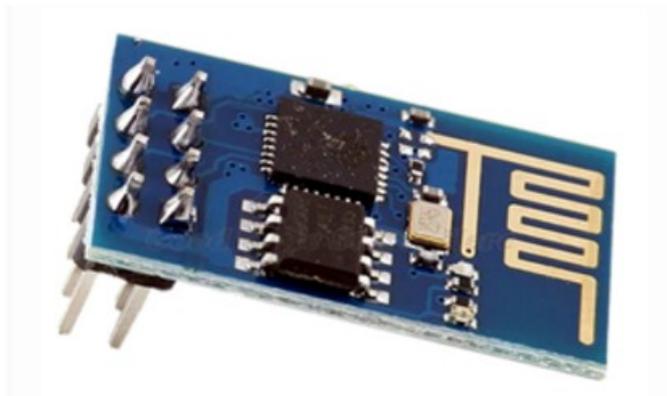


Рис. 4. Wi-Fi-модуль ESP8266

Предусмотрено два варианта использования чипа:

1) в виде моста UART-WIFI, когда модуль на базе ESP8266 подключается к существующему решению на базе любого другого микроконтроллера и управляется AT-командами, обеспечивая связь решения с инфраструктурой Wi-Fi;

2) реализуя новое решение, использующее сам чип ESP8266 в качестве управляющего микроконтроллера.

#### Характеристики:

- Микромодуль: ESP8266MOD;
- Полная совместимость выводов с ESP-12;
- Объем флеш-памяти: 4Мб;
- Напряжение питания: 3.3В;
- Рабочая частота: 2.4 ГГц;
- Мощность излучения: +24dbm;
- Порты GPIO: доступны во всех исполнениях модуля с шагом 2.0мм;
- Тип антенны: встроенная;
- Беспроводные режимы: точка доступа / softAP / SoftAP + точка доступа;
- Беспроводный протокол: 802.11 b /g /n;

- Поддержка безопасности: Wi-Fi @ 2.4GHz, поддержка WPA / WPA2 режима безопасности;
- Управление: встроенными AT-командами.

Выводы (пины):

- Vcc – питание, +3,3В (максимально 3,6 В);
- GND – общий;
- TXD – передача данных (уровень 3,3 В);
- RXD – приём данных (уровень 3,3 В);
- CH\_PD – выключение модуля (низкий уровень активный, для включения модуля следует подать Vcc);
- GPIO0 – вывод общего назначения 0;
- GPIO2 – вывод общего назначения 2;
- RST – сброс модуля (низкий уровень активный).

Алгоритм работы по созданию автоматизированной системы удаленного управления состоит из следующих шагов:

1. Собрать аппаратную часть системы;
2. Зайти на сайт [www.remotexy.com](http://www.remotexy.com)
3. Создать графический интерфейс управления системой (разместить соответствующие элементы управления);
4. Загрузить на мобильное устройство приложение RemoteXY, работающее под управлением ОС Андроид;
5. В микропроцессорную систему Ардуино загрузить разработанный скетч, позволяющий управлять удаленными устройствами (в нашем случае вентилятором и нагревательным элементом);
6. Запустить на мобильном устройстве приложение RemoteXT;
7. Выбрать тип подключения (Bluetooth/WiFi), подключиться к сети и выполнить авторизацию;
8. На экране мобильного устройства проконтролировать параметры системы – текущую влажность и температуру;
9. При необходимости с помощью элементов управления Slider – установить новые требуемые значения температуры;
10. С помощью соответствующих индикаторов проконтролировать состояние устройств управления температурой и влажностью.

С помощью программы RemoteXY был разработан следующий графический интерфейс системы (рис. 5):

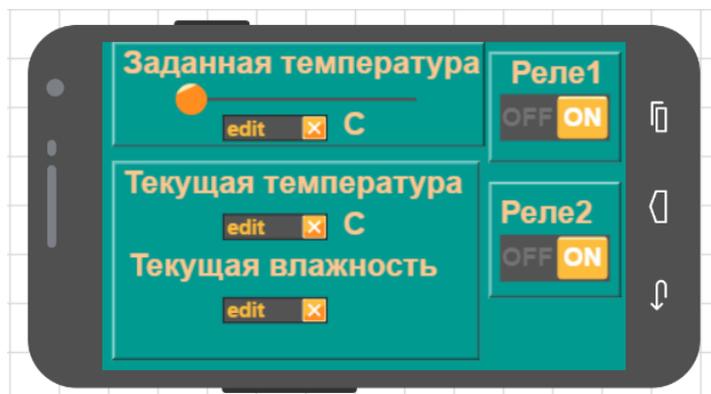


Рис. 5. Графический интерфейс в мобильном устройстве

Элементы управления, использованные в интерфейсе [3, 7]:

 Элемент управления «Поле ввода» предназначен для ввода текстовой строки или числа с клавиатуры в графическом интерфейсе. Введенное значение передается в контроллер в соответствующее поле структуры RemoteXY. Поле ввода поддерживает ввод произвольного текста, ввод целого числа, ввод вещественного числа;

 Выключатель позволяет передать на микроконтроллер одно из двух фиксированных положений: включено или отключено. Переключатель фиксирует свое положение без необходимости постоянного удерживания. Для изменения состояния выключателя необходимо пальцем передвинуть его движок или нажать пальцем на соответствующую сторону выключателя;

 Слайдер является пропорциональным элементом управления и позволяет передать на микроконтроллер плавное изменение своего положения. Он позволяет плавно изменять значение от 0 до 100 (целые числа).

Фрагмент программного скетча, загружаемого в Ардуино:

```
////////////////////////////////////  
// RemoteXY include library //  
// use library version 2.2.2 or up //  
// use ANDROID app version 3.4.1 or up //  
////////////////////////////////////
```

```
/* определение режима соединения и подключение библиотеки RemoteXY */
#define REMOTEXY_MODE__HC05_SOFTSERIAL
#include <SoftwareSerial.h>

#include "RemoteXY.h";

/* настройки соединения */
#define REMOTEXY_SERIAL_RX 2
#define REMOTEXY_SERIAL_TX 3
#define REMOTEXY_SERIAL_SPEED 9600

/* конфигурация интерфейса */
unsigned char RemoteXY_CONF[] =
  { 3,33,57,1,4,5,4,128,5,11
    ,57,7,2,2,0,68,11,28,8,2
    ,79,78,0,79,70,70,0,2,0,68
    ,29,28,8,2,79,78,0,79,70,70
    ,0,67,4,22,20,20,6,4,11,67};

/* структура определяет все переменные вашего интерфейса управления */
struct {

  /* input variable */
  signed char slider_setTemp; /* =0..100 положение слайдера */
  unsigned char switch_relay3; /* =1 если переключатель включен и =0 если отключен */
  unsigned char switch_relay4; /* =1 если переключатель включен и =0 если отключен */

  /* output variable */
  char text_setTemp[11]; /* =строка оканчивающаяся нулем UNICODE */
  char text_valTemp[11]; /* =строка оканчивающаяся нулем UNICODE */
  char text_valHum[11]; /* =строка оканчивающаяся нулем UNICODE */

  /* other variable */
```

```
unsigned char connect_flag; /* =1 if wire connected, else =0 */

} RemoteXY;

////////////////////////////////////
//      END RemoteXY include      //
////////////////////////////////////
#include "DHT.h"

#define PIN_RELAY_COLD 7
#define PIN_RELAY_HOT 6
#define PIN_RELAY_3 5
#define PIN_RELAY_4 4
#define PIN_DHT 8

DHT dht(PIN_DHT, DHT11);
void setup()
{
  RemoteXY_Init ();
  // TODO you setup code
  pinMode (PIN_RELAY_COLD, OUTPUT);
  pinMode (PIN_RELAY_HOT, OUTPUT);
  pinMode (PIN_RELAY_3, OUTPUT);
  pinMode (PIN_RELAY_4, OUTPUT);
}
void loop()
{
  RemoteXY_Handler ();
  // TODO you loop code
  //////////////////////////////////
  // Логика задачі управління температурой //
  //////////////////////////////////
}
```

```
// получаем значение установленной температуры от 10 до 40 град.ц.  
int setTemp = RemoteXY.slider_setTemp/100.0*30.0+10.0;  
  
// передаем установленную температуру обратно на смартфон  
itoa (setTemp, RemoteXY.text_setTemp, 10);  
  
// опрашиваем датчик DHT11  
int valTemp = dht.readTemperature();  
int valHum = dht.readHumidity();  
  
// отправляем значения температуры и влажности в интерфейс  
itoa (valTemp, RemoteXY.text_valTemp, 10);  
itoa (valHum, RemoteXY.text_valHum, 10);  
  
// логика управления по заданной температуре  
digitalWrite(PIN_RELAY_COLD, (valTemp>setTemp)?HIGH:LOW);  
digitalWrite(PIN_RELAY_HOT, (valTemp<setTemp)?HIGH:LOW);  
  
// устанавливаем реле 1 и 2 в соответствии с выключателями интерфейса  
digitalWrite(PIN_RELAY_3, (RemoteXY.switch_relay1==0)?LOW:HIG);  
digitalWrite(PIN_RELAY_4, (RemoteXY.switch_relay2==0)?LOW:HIG);  
}
```

С помощью библиотеки RemoteXY устанавливается режим и настройки соединения (пины RX, TX и скорость соединения). Далее формируется интерфейс мобильного устройства, на основании информации, заданной в режиме редактора. Определяются пины подключения датчика DHT-11 и реле включения вентилятора и нагревательного элемента. Далее реализован алгоритм управления температурой управляемого объекта, осуществляя включение вентилятора или нагревательного элемента.

### ***Выводы***

Разработанная система позволяет удаленно наблюдать за параметрами объекта управления (температурой и влажностью). Система позволяет удаленно управлять температурой объекта как по каналам Bluetooth, так и с помощью Wi-Fi.

Имеется возможность дистанционно устанавливать заданную температуру с помощью мобильного устройства. При этом, в случае необходимости, автоматически включается вентилятор для уменьшения температуры или нагревательный элемент для ее увеличения.

#### ЛИТЕРАТУРА

1. Дейтел П. Android для программистов. Создаем приложения / П. Дейтел, Х. Дейтел, Э. Дейтел, М. Моргано. – СПб. : БХВ-Петербург, 2013. – 415 с.
2. Хашими С. Разработка приложений для Android. / С. Хашими, С. Коматинени, Д. Маклин – СПб. : БХВ-Петербург, 2011. – 1024 с.
3. Петин В. Arduino и Raspberry Pi в проектах Internet of Things. / В. Петин. – СПб. : БХВ-Петербург, 2016. – 320 с.
4. Таранушенко С. Arduino, датчики и сети для связи устройств / С. Таранушенко. – СПб. : БХВ-Петербург, 2015. – 544 с.
5. Карвинен Т. Делаем сенсоры. Проекты сенсорных устройств на базе Arduino и Raspberry Pi. / Т. Карвинен, К. Карвинен, В. Валтокари. – М. : Вильямс, 2015. – 448 с.
6. Блум Дж. Изучаем Arduino. Инструменты и методы технического волшебства. / Дж. Блум – СПб. : БХВ-Петербург, 2015. – 336 с.
7. Рето Майер Android 2. Программирование приложений для планшетных компьютеров и смартфонов. / Майер Рето. – М. : Эксмо, 2011. – 672 с.
8. Соммер У. Программирование микроконтроллерных плат Arduino/Freduino. / У. Соммер. – СПб. : БХВ-Петербург, 2015. – 256 с.
9. Стюарт Болл Р. Аналоговые интерфейсы микроконтроллеров. / Р. Стюарт Болл – М. : Издательский дом «Додэка XXI», 2007. – 360 с.

#### **Автоматизоване віддалене управління пристроями за допомогою Ардуіно Столяров В. Г., Голубєв Л. П.**

*Київський національний університет технологій та дизайну*

*У статті розглядаються можливості автоматизованого керування пристроями за допомогою Ардуіно. Наведено приклад створення автоматизованої системи віддаленого управління температурою і вологістю за допомогою мікропроцесорної системи Ардуіно Уно.*

**Ключові слова:** *Ардуіно, Wi-Fi, Bluetooth, мікропроцесор, автоматизація, система управління, температура, вологість*

*Automated remote control devices using Arduino*

*Stolyarov V. G., Golubev L. P.*

*Kiev National University of Technology and Design*

*The article discusses the possibility of an automated remote device management using the Arduino. An example of the creation of the automated remote control system of temperature and humidity using a microprocessor system Arduino Uno.*

**Keywords:** *Arduino, Wi-Fi, Bluetooth, a microprocessor, automation, system management, temperature, humidity*