

## **Использование микроконтроллера ESP8266 01. Программирование с помощью Arduino IDE**

*Худяков Сергей Викторович  
Амурский государственный университет  
Магистрант*

*Самохвалова Светлана Геннадьевна  
Амурский государственный университет  
кандидат технических наук, доцент кафедры ИУС*

### **Аннотация**

В работе приведено описание микроконтроллера, основные характеристики и возможности, рассмотрен функционал. Рассмотрено программирование микроконтроллера с помощью Arduino IDE. В качестве примера показана разработанная система удалённого мониторинга температуры и влажности.

**Ключевые слова:** микроконтроллер, Wi-Fi, ESP8266 01, Arduino, датчик температуры и влажности, скетч.

## **The using ESP8266 01. Programming with Arduino IDE**

*Hudjakov Sergej Viktorovich,  
Amur State University  
Undergraduates*

*Samohvalova Svetlana Gennad'evna  
Amur State University  
Ph.D., Associate Professor of IMS*

### **Abstract**

The article deal with the microcontroller description, key features and capabilities, is considered functional. Considered microcontroller programming with the Arduino IDE. As an example of the developed system of remote monitoring of temperature and humidity.

**Keywords:** microcontroller, ESP8266 01, Wi-Fi, Arduino, temperature and humidity sensor, a sketch.

Термин «Интернет вещей» сегодня можно услышать довольно часто. Данный термин предполагает интеграцию интернет соединения посредством Wi-Fi в небольшие и недорогие устройства, чтобы иметь возможность подключить их напрямую в инфраструктуру.

Данные системы можно встретить на рынке электроники для дома и офиса и они имеют общее название – системы «Умный дом». Умный дом

(smart home, также home automation) – система домашних устройств, способных выполнять действия и решать определенные задачи без участия человека. Наиболее распространенные примеры использования таких систем – автоматическое включение и выключение света, автоматическая коррекция работы отопительной системы или кондиционера и автоматическое уведомление о вторжении, возгорании или протечке воды.

Спрос на подобные системы сейчас очень быстро возрастает. Ими интересуются как рядовые разработчики, решающие использовать такие системы для своих бытовых нужд, так крупные компании для решения крупных задач по автоматизации.

Поэтому 2014 год стал для многих разработчиков-любителей знаменательным, так как большое количество производителей объявили о выпуске наборов микросхем. Данные микросхемы представляют собой как отдельные модули, так и полноценные микроконтроллеры, обладающие большим функционалом для решения различных задач.

В данной статье пойдет речь о микроконтроллере ESP8266 от компании Espressif (Китай). Пресс-релизы показали, как производитель гордится степенью интеграции элементов своего чипа. Так и есть, эта микросхема состоит всего лишь из семи элементов. Плата модуля практически пуста, компонентов – ничтожное количество. Все это естественно будет сказываться на стоимости конечных устройств в сторону удешевления.

Существует достаточно большое количество разновидностей микроконтроллера ESP8266, которые отличаются характеристиками, количеством входов и выходов, а также типом используемой антенны. По большому счёту ESP8266 нужно считать именно микроконтроллером, что будет понятно исходя из характеристик, представленных для одного из них ниже.

Далее будет рассмотрен микроконтроллер ESP8266 01, который представлен на рисунке 1.

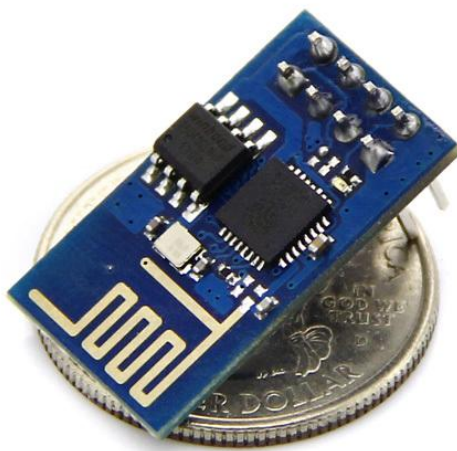


Рисунок 1 – Микроконтроллер ESP8266 01

Технические характеристики:

- Процессор: одноядерный Tensilica L106 частотой до 160 MHz;
- Поддерживаемые стандарты WI-FI: 802.11 b / g / n;
- Поддерживаемы типы шифрования: WEP, WPA, WPA2;
- Поддерживаемые режимы работы: Клиент(STA), Точка доступа(AP), Клиент+Точка доступа(STA+AP);
- Напряжение питания 1.7..3.6 В;
- Потребляемый ток: до 215мА в зависимости от режима работы;
- Количество GPIO: 16 (фактически до 11). Доступно на модулях: ESP-01 - 4, ESP-03 - 7+1, включая UART. Существуют и другие варианты модулей;
- Интерфейсы: 1 ADC, I2C, UART, SPI, PWM;
- Внешняя Flash память может быть установлена от 512кб до 4мб;
- RAM данных 80 кб, RAM инструкций - 64 кб.

В качестве основного языка программирования микроконтроллера ESP8266 используется C (в некоторых случаях C++) [1]. В зависимости от выбранного компилятора могут использоваться и другие языки.

Все языки, используемые для написания программ для микроконтроллера, являются объектно-ориентированными, поэтому написание хорошей программы (скетча или так называемой «прошивки») можно считать достаточно простым вопросом для продвинутого программиста.

Одним из компиляторов, который зарекомендовал себя среди разработчиков-любителей, является Arduino IDE.

Arduino IDE – это среда разработки, созданная с целью написания и загрузки программ на микроконтроллеры Arduino и не только. Она состоит из встроенного текстового редактора программного кода, области сообщений, окна вывода текста(консоли), панели инструментов с кнопками часто используемых команд и нескольких меню. Для загрузки программ и связи среда разработки подключается к аппаратной части Arduino.

Программа, написанная в среде разработки Arduino IDE, называется скетч. Скетч пишется в текстовом редакторе, имеющем инструменты вырезки/вставки, поиска/замены текста [2]. Во время сохранения и экспорта проекта в области сообщений появляются пояснения, также могут отображаться возникшие ошибки. Окно вывода текста (консоль) показывает сообщения Arduino, включающие полные отчеты об ошибках и другую информацию. Кнопки панели инструментов позволяют проверить и записать программу, создать, открыть и сохранить скетч, открыть мониторинг последовательной шины.

Arduino IDE для ESP8266 поддерживает все существующие на сегодняшний день модули ESP8266, включая модули с флэш большего, чем 512k объема. Поддерживаются модули NodeMCU (всех версий), Olimex-MOD-WiFi-ESP8266.

Для того чтобы компилятор Arduino IDE смог работать с микроконтроллером ESP8266 необходимо загрузить соответствующую библиотеку и выбрать платформу в панели инструментов (см. рисунок 2).

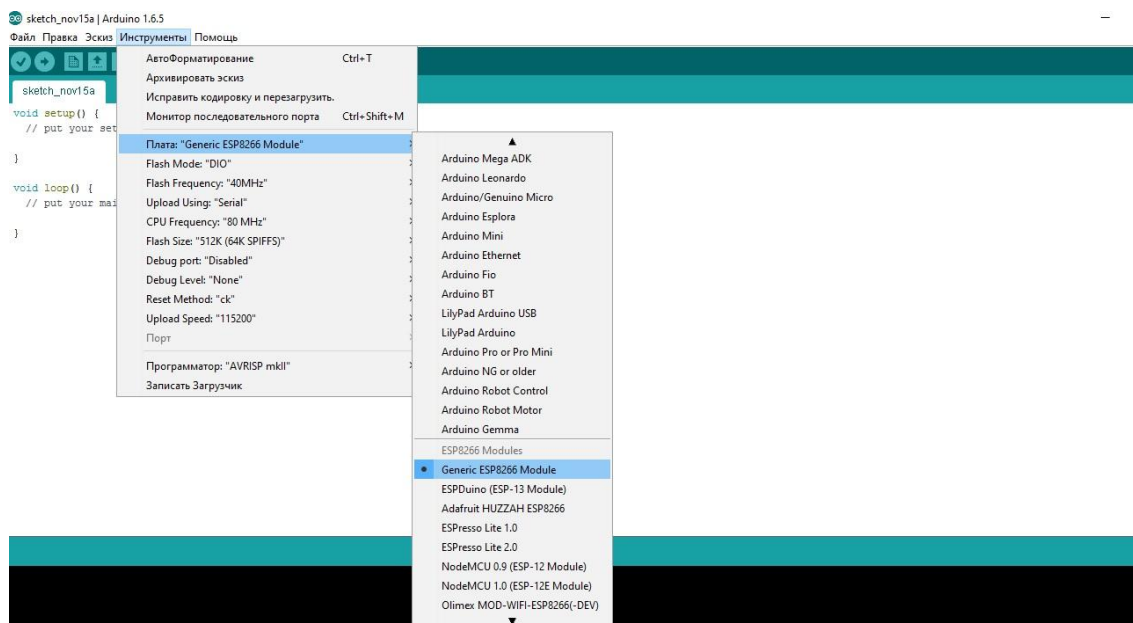


Рисунок 2 – Выбор платы в среде разработки Arduino IDE

После того как программа написана, её необходимо загрузить на микроконтроллер нажатием соответствующей кнопки в компиляторе Arduino IDE. Но перед этим нужно подключить ESP8266 01 к последовательному порту используя специальный адаптер USB TTL. Ниже будет представлена схема подключения для загрузки программы [3].

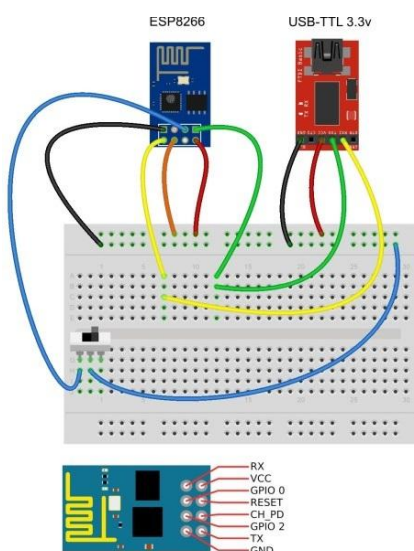


Рисунок 3 – Схема подключения микроконтроллера для загрузки программы

После загрузки программы необходимо отключить схему от последовательного порта и собрать схему уже в соответствии с выбранной архитектурой разработки.

Easy IoT – это программно-аппаратный комплекс от компании Elliptic Labs для различных устройств «Интернета вещей.» Суть проста. Благодаря ультразвуковым датчикам, некоторые устройства могут получить значительные преимущества. К примеру, техника будет реагировать на движения, появление в комнате пользователя, активировав соответствующий сценарий (включение или выключение света, какого-либо прибора и прочее). Всё это уже реализовано в готовых устройствах, просто на основе иных технологий. Перспективы Easy IoT достаточно велики, так как ультразвуковые датчики имеют определённые преимущества перед оптическими (проницаемость волн, направленность и так далее).

Благодаря данному сервису рядовые разработчики могут создавать системы автоматизации как для дома, так и для более глобальных целей. Примерами могут являться: передовая система сигнализации, системы дистанционного орошения растений, дистанционный контроль солнечных батарей, SMS контроль обогревателей, открывания дверей или даже системы управления технологическими процессами. Все это возможно с использованием EasyIoT.

### EasyIoT framework II architecture

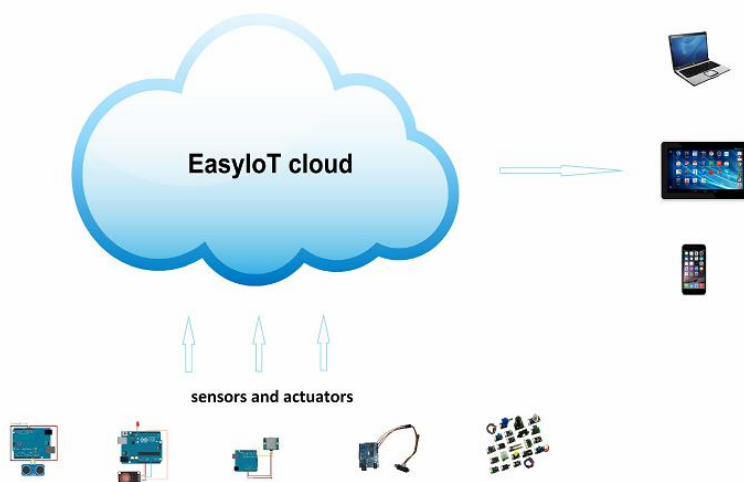


Рисунок 4 – Сервис EasyIoT

Сервис работает круглые сутки без перебоев, что можно считать безусловным преимуществом для разработки больших и надёжных систем.

В качестве примера ниже будет описана разработанная система для мониторинга температуры и влажности с использованием сервиса EasyIoT. Будут приведены характеристики используемых компонентов и представлен результат.

Для снятия значений температуры и влажности используется соответствующий датчик DHT22, выполненный в композитном корпусе и подключенный к цифровому входу микроконтроллера.

Технические параметры:

- Питание от 3 до 5В;

- Максимально потребляемый ток - 2.5мА при преобразовании (при запросе данных);
- Рассчитан на измерение уровня влажности в диапазоне от 0% до 100% (при этом точность измерений находится в диапазоне 2%-5%);
- Измеряет температуру в диапазоне от -40 до 125 градусов с точностью плюс-минус 0.5 градусов по Цельсию;
- Частота измерений до 0.5 Гц (одно измерение за 2 секунды);
- Размер корпуса: 15.1 мм x 25 мм x 7.7 мм.



Рисунок 5 – Датчик температуры и влажности

Также был использован микроконтроллер ESP8266 01, характеристики и описание которого представлены выше.

Так как для подключения питания необходимо напряжение 3.3 вольта был использован стабилизатор AMS1117.

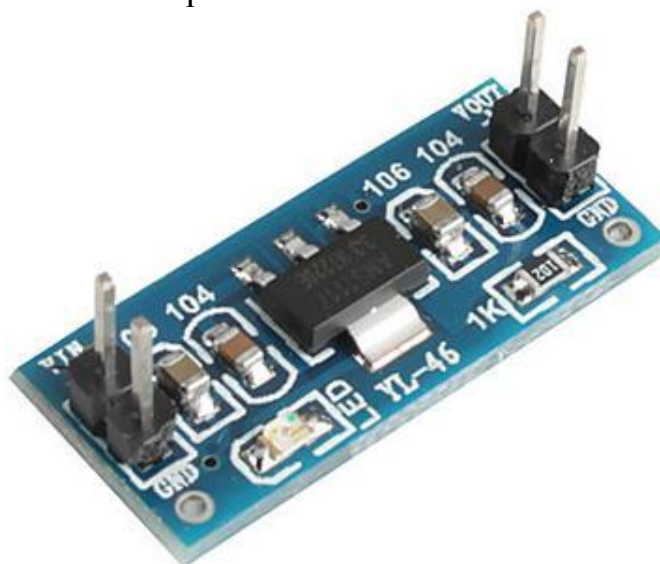


Рисунок 6 – Стабилизатор напряжения

Технические характеристики:

- Максимальный выходной ток – 1 А;
- Максимальное входное напряжение – 18 В;
- Температурный диапазон работы  $T = -20 \dots +125^{\circ}\text{C}$ ;

- Максимальная рассеиваемая мощность для корпуса SOT-223 –  $P_{max} = 0,8 \text{ Вт}$ ;
- Максимальная рассеиваемая мощность для корпуса TO-252 –  $P_{max} = 1,5 \text{ Вт}$ ;
- Тепловое сопротивление кристалл-корпус для корпуса SOT-223 –  $R_t = 15^\circ\text{C/Вт}$ ;
- Тепловое сопротивление кристалл-корпус для корпуса TO-252 –  $R_t = 3^\circ\text{C/Вт}$ ;
- Выключение при перегреве кристалла –  $T = 155^\circ\text{C}$ ;
- Тепловой гистерезис –  $\Delta T = 25^\circ\text{C}$ .

После загрузки скетча была собрана схема, представленная ниже, с учётом того, что источником питания является стабилизатор с подключенным к нему напряжением от четырёх АА батарей [4].

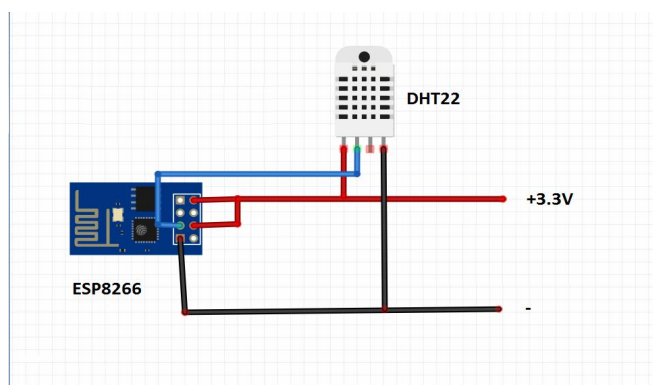


Рисунок 7 – Схема сборки

После подключения питания происходит соединение ESP8266 01 с точкой доступа Wi-Fi и данные сразу отправляются на сервис EasyIoT. Результаты представлены на рисунках 8 и 9.

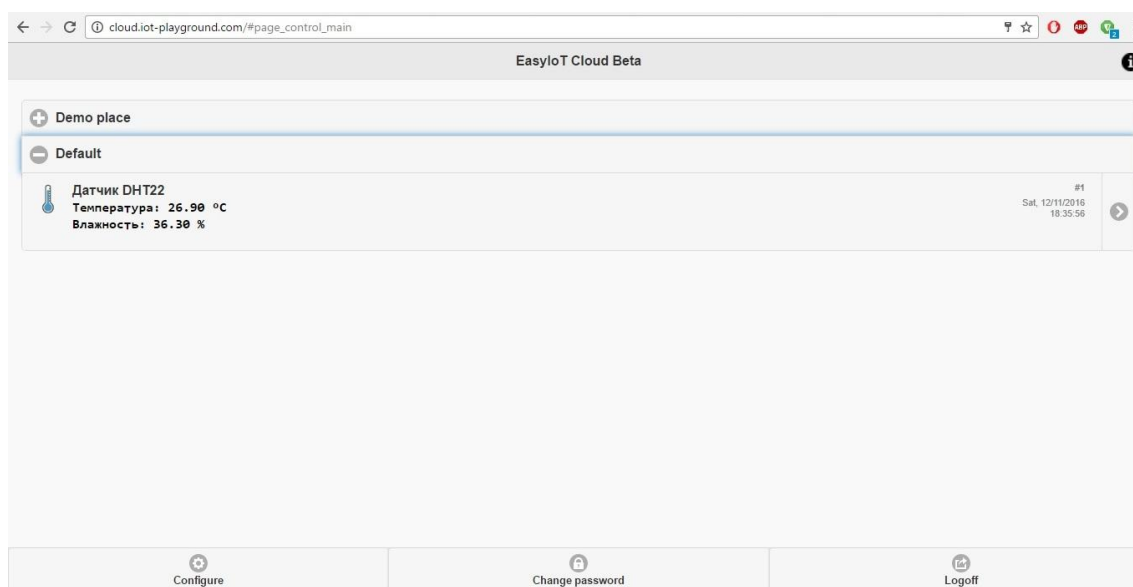


Рисунок 8 – Просмотр данных с сервиса EasyIoT

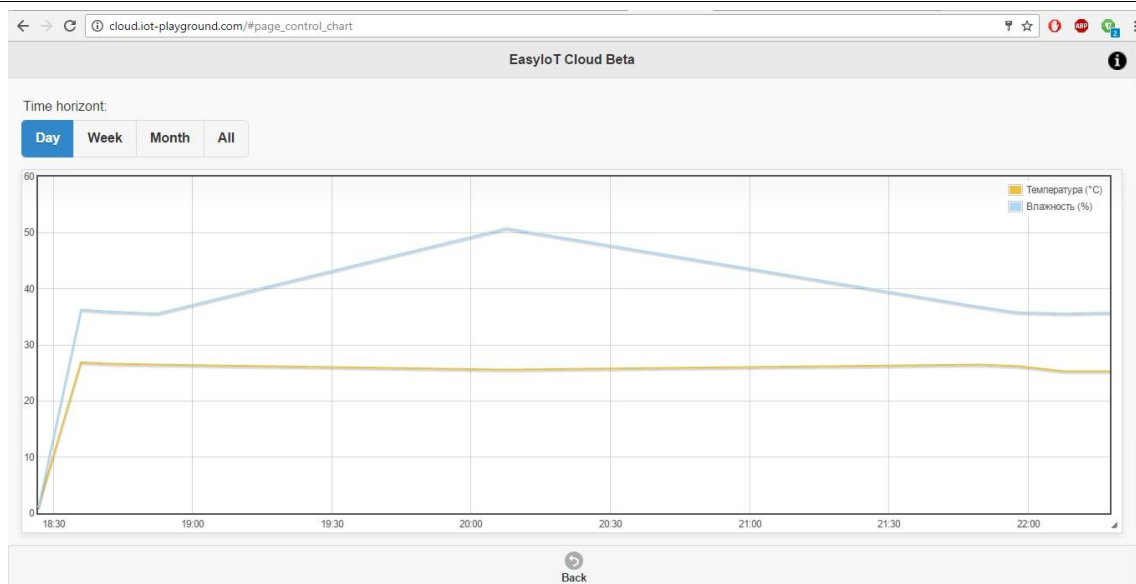


Рисунок 9 – Просмотр графиков изменения температуры и влажности

### Библиографический список

1. Прокопенко В. С. Программирование микроконтроллеров на языке С. СПб.: Корона-Век, 2012. С. 8–20.
2. Соммер У. Программирование микроконтроллерных плат Arduino /Freeduino. СПб.: БХВ-Петербург, 2012. С. 13 – 29.
3. Бродин В.Б. Калинин А.В. Системы на микроконтроллерах и БИС программируемой логики. М.: ЭКОМ, 2010. С. 303 – 340.
4. Белов А.В. Конструирование устройств на микроконтроллерах. М.: ЭКОМ, 2010. С. 7 – 60.