Study of arduino microcontroller board

Article · March 2022

CITATIONS
72

READS
64,418

2 authors, including:

Alisher Shakirovich Ismailov
Tashkent State University of Economics
41 PUBLICATIONS **183** CITATIONS

SEE PROFILE

Study of arduino microcontroller board

Alisher Shakirovich Ismailov
Zafar Botirovich Jo‘rayev
Andijan Machine Building Institute

Abstract: This paper will discuss Arduino microcontroller working principle and applications. In this paper, we will also discuss how the Arduino microcontroller can be used as a tool for study and research works. Arduino microcontroller can provide a quick tool for developing small projects that involve sensors. Arduino microcontroller is easy to learn, and easy to program. Arduino IDE can program Arduino microcontroller. Arduino IDE is a tool to write a program for Arduino boards. Arduino IDE is open-source software that can be downloaded and installed on the computer is free. Arduino IDE provides many ready-to-use libraries. Using these libraries Arduino developers will save a lot of time. This paper provides a glimpse of the type of Arduino boards, working principles, software implementation, and Arduino applications.

Keywords: Arduino, Microcontroller, Hardware, Software, Open-source platform, Sensors

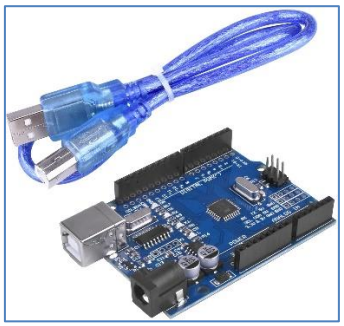
INTRODUCTION

Arduino microcontroller is an open-source that can be easily programmed and can update at any time. First Arduino was introduced in 2005. Arduino microcontroller was originally designed for professionals and students to develop devices that can interact with the environment using sensors.

Arduino microcontrollers have inputs and outputs that can be used to get information and based on received data Arduino can send output. Arduino microcontrollers can also send and receive data via the internet using HTTP requests. The simple microcontroller that can be connected to the internet is Esp board. Esp microcontrollers can be connected to a Wi-Fi server or they can act as a Wi-Fi server.

Arduino platform can be divided into two: Hardware and Software.

Arduino uses hardware known as the Arduino development board. Arduino software for developing the code is known as the Arduino IDE (Integrated Development Environment). Built-up with the 8-bit Atmel AVR microcontrollers that are manufactured by Atmel or a 32-bit Atmel ARM, these microcontrollers can be programmed easily using the C or C++ language in the Arduino IDE.



The Arduino board can also be used to upload a new code to the Arduino board by using a USB cable to upload. The Arduino IDE provides a simplified integrated platform that can run on almost all personal computers and users can write programs for Arduino using C or C++ programming language.

There are many Arduino microcontroller boards available on the market. To use proper Arduino board depending on the project requests little research. Each Arduino board has different specifications and capabilities.

There are many reasons to use Arduino microcontrollers instead of other microcontrollers.

Arduino Co-founder Massimo Banzi mentioned some very important reasons to use Arduino boards.

- Active User Community: Arduino users can post a message and share their experiences. If you occur a problem, using the Arduino board, you post your problem to a community platform then other users normally will give solutions or guidance for solving the problem [1].
- Growth of Arduino: Arduino microcontrollers are cheaper than their competitors are. This makes it perfect for newcomers to get started quickly [1].
- Inexpensive Hardware: Arduino platform is free to use from the official website. The only thing users pay is for Arduino hardware [1].
- Arduino Board as a Programmer

- Multi-platform Environment: The Arduino IDE is multiplatform software that can run on several platforms including Microsoft, Linux, and Mac OS X making the user community even larger [1].

DIFFERENT TYPES OF ARDUINO BOARDS

In the Arduino board family there, many types of Arduino boards are available. For wireless communication, boards such as Arduino BT come with a built-in Bluetooth module. These built-in modules can also be available separately which can then be interfaced to it. These modules are known as Shield.

Some of the popular Arduino Shields:

- Arduino Ethernet shield: This shield allows an Arduino board to connect to the internet by Ethernet library and to read and write an SD card using the SD library [2].
- Arduino Wireless shield: this shield allows the Arduino board to communicate wirelessly using Zigbee [2].
- Arduino Motor Driver Shield: this shield allows Arduino boards to interface with the driver of a motor etc. [2].

Fig. 1. Arduino Shields - Ethernet, Wireless, and Motor Driver.

Here is a list of the different types of Arduino Boards available along with their microcontroller type, crystal frequency, and availabilities of auto-reset facility:

Table 1. Heading and text fonts

Arduino Type	Microcontroller	Clock Speed
Arduino Uno	ATmega328	16 MHz with auto-reset
Arduino Duemilanove / ATmega328	ATmega328	16 MHz with auto-reset
Arduino Nano	ATmega328	16 MHz with auto-reset
Arduino Mega 2560 or Mega ADK	ATmega2560	16 MHz with auto-reset
Arduino Leonardo	ATmega32u4	16 MHz with auto-reset
Arduino Mini w/ ATmega328	ATmega328	16 MHz with auto-reset
Arduino Fio.	ATmega328	8 MHz with auto-reset
Arduino BT w/ ATmega328	ATmega328	16 MHz with auto-reset
Lily Pad Arduino w/ ATmega328	ATmega328	8 MHz with auto-reset
Arduino Pro or Pro Mini	ATmega328	16 MHz with auto-reset
Arduino NG	ATmega8	16 MHz with auto-reset

ELEMENTS OF ARDUINO BOARDS

Elements of an Arduino Board can be done into two categories:

- *Hardware*
- *Software*

Hardware

The Arduino microcontroller consists of many components. Here are some of those main components and their functionality:

- Microcontroller: This is the brain of the development board that can receive and sends information or command to the peripheral devices connected to it. The microcontroller differs from board to board; it also has various specifications.
- External Power Supply: This power supply is used to power the Arduino microcontroller with a DC voltage ranging from 9 - 12 volts.
- USB plug: This plug is a very important port in the Arduino board. It is used to upload a program to the microcontroller using a USB cable. The USB cable has a DC power of 5V that powers the Arduino board in cases when the External Power Supply is absent.
- Internal Programmer
- Reset button
- Analog Pins: These pins are used for the analog input/output. The number of analog pins also varies from board to board.
- Digital I/O Pins: These pins are used for the digital input/output. The number of these digital pins also varies from board to board.
- Power and GND Pins: There are pins on the development board that provides 3.3, 5 volts and ground through them

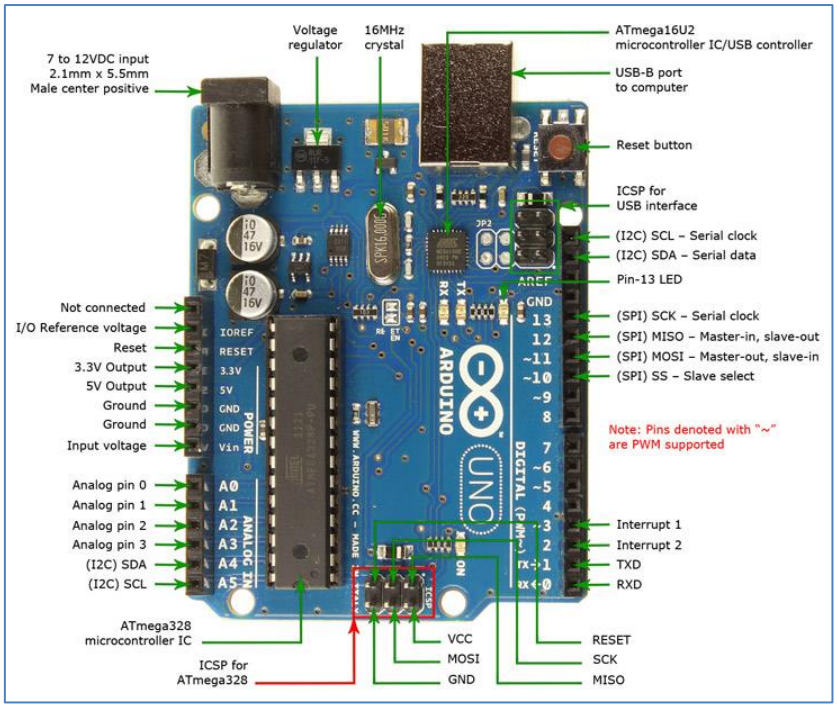


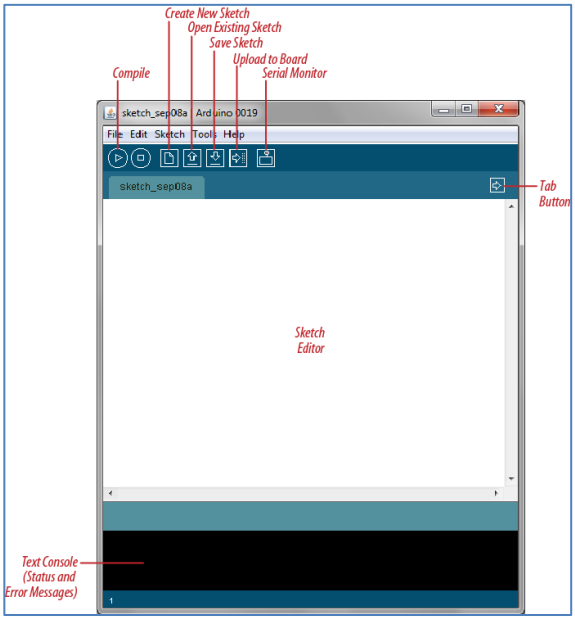
Fig. 2. A labeled diagram of an Arduino Board and an IDE.

Software

The program code written for the Arduino board is also called a sketch. The software used for developing such sketches for an Arduino is an Arduino IDE. This

IDE contains the following parts in it:

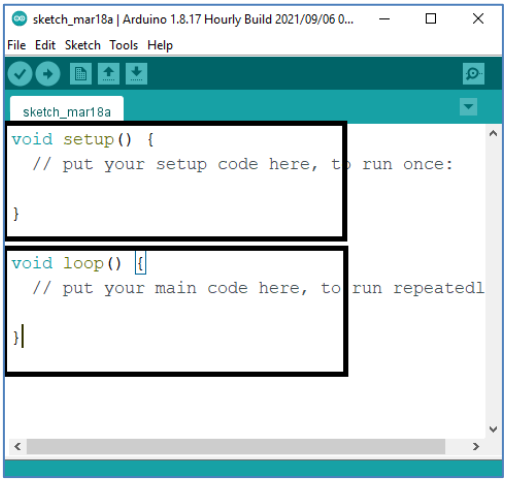
- Text editor: This is where programming code can be written C/C++ programming language.
- Message area: It displays an error and gives feedback on saving and exporting the code.
- Text: The console displays text output by the Arduino environment including complete error messages and other information
- Console Toolbar: This toolbar includes some buttons like Compile, Upload, New, Open, Save, and Serial Monitor.



PROGRAMMING BASICS

Now we discuss programming techniques of Arduino sketch in the Arduino IDE platform. There are two main functions every sketch will always have, they are:

- void setup ()
- void loop ()



1) void setup() function

This is the first process that begins when the Arduino starts functioning. This function is executed only once throughout the entire program's functioning. The setup function contains the initialization of every pin we intend to use in the project for input or output. Here is an example of how it should be written:

```
void setup() {
  // put your setup code here, to run once:
  pinMode(2, INPUT);
  pinMode(3, OUTPUT);
}
```

In this example, we have set pin 2 as an INPUT and pin 3 as an OUTPUT.

It also contains the initialization of the Serial Monitor. A serial monitor is used to know the data that are being sent serially to any peripheral device.

```
void setup() {
  // put your setup code here, to run once:
  pinMode(2, INPUT);
  pinMode(3, OUTPUT);

  Serial.begin(9600);
}
```

Note: Before using any variables for programming it is necessary to define them above the function "void setup()"

2) void loop() function

This function is the next important function in Sketch. Void loop() sketch runs infinitely. Once you write code on void loop() it will never stop.

There are few built-in functions available in Arduino IDE. Following explains each built-in functions:

digitalWrite() function

digitalWrite() function accepts two parameter; first parameter is digital pin number and

HIGH or a LOW value to a digital pin. For example:

digitalWrite(2, HIGH);

If the pin has been set as an OUTPUT with pinMode(), its voltage will be set to the

Corresponding value: 5V for HIGH, and 0V for LOW.

Example:

```
void loop() {
  // put your main code here, to run repeatedl
  digitalWrite(3, HIGH);
  delay(1000);
  digitalWrite(3, LOW);
  delay(1000);
}
```

In the above code digital pin 3 become on for 1 second and off for 1 second

infinitely.

digitalRead() function

If the pin is set as an INPUT, digitalWrite() will enable (HIGH) or disable (LOW) the internal pullup on the input pin.

analogWrite() function

Writes an analog value (PWM wave) to a pin. Can be used to light a LED at varying brightness or drive a motor at various speeds. After a call to analogWrite(), the pin will generate a steady rectangular wave of the specified duty cycle until the next call to analogWrite() (or a call to digitalRead() or digitalWrite()) on the same pin.

analogRead()

Reads the value from the specified analog pin. The Arduino board contains a 6 channel (8 channels on the Mini and Nano, 16 on the Mega), 10-bit analog to digital converter.

This means that it will map input voltages between 0 and 5 volts into integer values between 0 and 1023. This yields a resolution between 5 volts / 1024 unit readings alternatively, .0049 volts (4.9 mV) per unit.

It takes about 100 microseconds (0.0001 s) to read an analog input, so the maximum reading rate is about 10,000 times a second.

CONCLUSIONS

In this paper, we have studied the working principle of Arduino, its hardware/software features and its applications as to where it is currently being used and where all it can be used. We have also learned how to write sketches for Arduino in its IDE (software).

Developing new ideas with Arduino is endless, with the help of this paper we have learned to build new devices of our own to create and implement innovative things. From wearable fashion to space research, the possibilities of using an Arduino to learn and develop new ideas are infinite. Though it does have its limitations, a great tool can be used in learning.

References

[1] ARDUINO.CC, “Arduino – Introduction”, 2015 [Online] Available: http://arduino.cc/en/Guide/Introduction. [Accessed: 25- Feb - 2015].

[2] Arduino.cc, 'Arduino - Products', 2015. [Online]. Available: http://arduino.cc/en/Main/Products. [Accessed: 25- Feb- 2015].

[3] ArduPilot Mega, 'ArduPilot Mega', 2015. [Online]. Available: http://www.ardupilot.co.uk/. [Accessed: 23- Nov- 2015].

[4] Wikipedia, 'ArduSat', 2015. [Online]. Available: http://en.wikipedia.org/wiki/ArduSat. [Accessed: 23- FEB- 2015].

[5] Kickstarter, 'ArduSat - Your Arduino Experiment in Space', 2015. [Online].

Available: https://www.kickstarter.com/projects/575960623/ardusat-your-arduino-experiment-in-space. [Accessed: 23- Feb- 2015].

[6] DIY Space Exploration, 'Ardusat - Your Personal Satellite Built on Arduino', 2013. [Online]. Available: http://www.diyspaceexploration.com/ardusat-your-personal-satellite. [Accessed: 13- Sep- 2015].

[7] J. Brodkin, '11 Arduino projects that require major hacking skills—or a bit of insanity', Ars Technica, 2013. [Online]. Available: http://arstechnica.com/information-technology/2013/05/11-arduino- projects-that-require-major-hacking-skills-or-a-bit-of-insanity/2/. [Accessed: 25- Nov- 2015].

[8] Galadima, A.A., "Arduino as a learning tool," in Electronics, Computer and Computation (ICECCO), 2014 11th International Conference on , vol., no., pp.1-4, Sept. 29 2014-Oct. 1 2014. doi: 10.1109/ICECCO.2014.6997577

[9] Badamasi, Y.A., "The working principle of an Arduino," in Electronics, Computer and Computation (ICECCO), 2014 11th International Conference on , vol., no., pp.1-4, Sept. 29 2014-Oct. 1 2014. doi: 10.1109/ICECCO.2014.6997578

[10] LilyPad Arduino, 'LilyPad Arduino', 2015. [Online]. Available: http://lilypadarduino.org/. [Accessed: 13- Sep- 2015].