

WEEK 7: Overview of the Development of Arduino based projects using open-source simulation software

Arduino is an open-source electronics platform based on free, flexible, easy-to-use hardware and software. for creators and developers. Arduino boards are able to read inputs - light on a sensor, a finger on a button and turn it into an output - activating a motor, turning on an LED. By sending a set of instructions to the microcontroller on the board we can tell the board what to do by using the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing. This platform allows us to create different types of single-board microcomputers.

Arduino UNO is a Microcontroller developed using ATmega328P. It has 14 digital I/O pins. From among these 14 pins, 6 are used for PWM output. Besides these Uno also has 6 analog inputs, a 16 MHz quartz crystal. It also contains a USB connection along with a power jack, an ICSP header, and a reset button. This Uno can be viewed as a plug and play device. To get started, one simply has to connect it to the computer using a USB connection or can power it using an **AC-DC adapter** or battery. For programming an Arduino Uno, an **Arduino IDE** is required. To work on **Arduino**, one has to install this IDE on the computer and write programs using it.

Arduino Hardware

It consists of a simple base board that has the microcontroller and its support circuitry with connectors to connect to plug in modules and a USB interface to download code from the PC. The commonest ones are called Arduino Uno and R3, which use Atmel ATmega328P microcontroller, but there are now several others with larger chips.

Arduino Software

The software is an open-source development environment, written in Java that can run under Linux, MAC or Windows. It runs a simple programming language called Wiring, which makes it fairly easy to write scripts to make the microcontroller carry out tasks.

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with

buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them.

Programs written using Arduino Software (IDE) are called **sketches**. These sketches are written in the text editor and are saved with the file extension. `.ino`. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

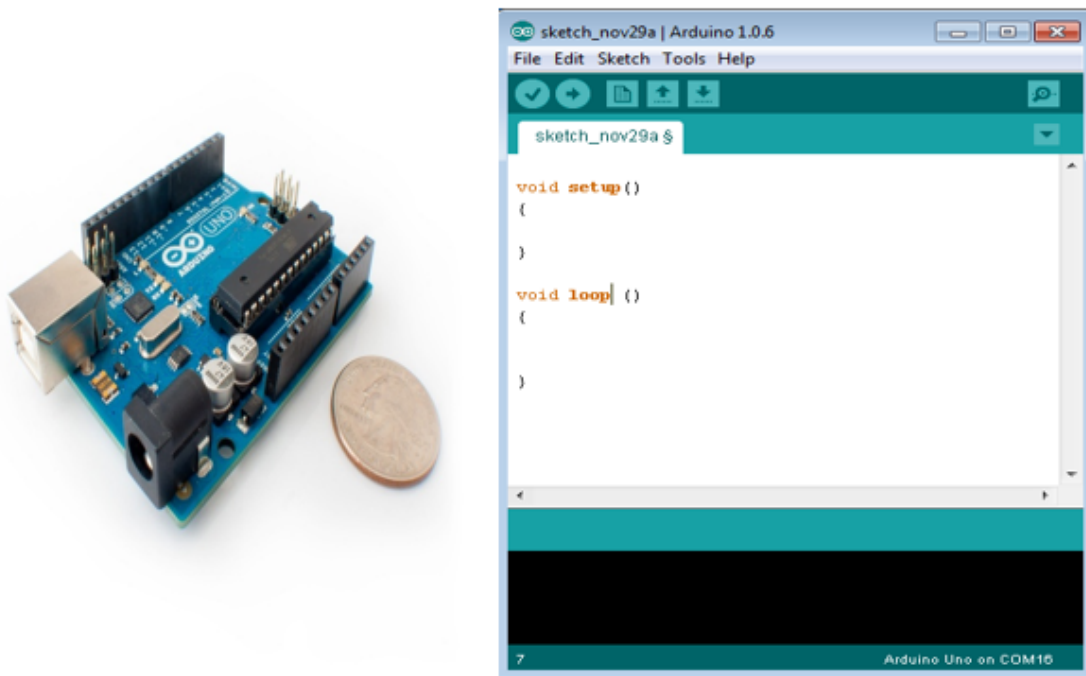


Fig : Arduino Board with Arduino IDE

Advantages of Arduino based projects

The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage.

- Not much knowledge required to get started
- Fairly low cost, depending on shields you need
- Lots of sketches and shields available

- No external programmer or power supply needed
- Easy to use (simple c/c++ programming)
- Inexpensive hardware (multiple projects can be done)
- Active user community
- Cross platform support (work on windows, linux, macos)
- Tons of libraries
- It is very good for carrying out a specific project you may have in mind.

Arduino boards based on ATMEGA328 microcontroller

Board Name	Operating Volt	Clock Speed	Digital i/o	Analog Inputs	PWM	UART	Programming Interface
Arduino Uno R3	5V	16MHz	14	6	6	1	USB via ATmega16U2
Arduino Uno R3 SMD	5V	16MHz	14	6	6	1	USB via ATmega16U2
Red Board	5V	16MHz	14	6	6	1	USB via FTDI
Arduino Pro 3.3v/8 MHz	3.3V	8MHz	14	6	6	1	FTDI-Compatible Header
Arduino Pro 5V/16MHz	5V	16MHz	14	6	6	1	FTDI-Compatible Header
Arduino mini 05	5V	16MHz	14	8	6	1	FTDI-Compatible Header
Arduino Pro mini 3.3v/8mhz	3.3V	8MHz	14	8	6	1	FTDI-Compatible Header

Table:Arduino boards based on ATMEGA328 microcontroller

Differentiate Arduino controller and other microcontrollers

ARDUINO	8051 MICROCONTROLLER
Arduino is not a microcontroller, but a system based on an AVR with its own IDE.	It is just a microcontroller belongs to 8 bit family of microcontroller.
Has 32KB of flash memory and 2KB of SRAM	Has 128 bytes of RAM but no EEPROM.
Operating voltage 6V to 20V via direct current barrel jack or into Vin pin.	Operating voltage +5V
Applications: home automation system, IoT etc	Applications: remote controls, power tools, auto engine control systems, toys and more

Table :Differences between Arduino and 8051

The difference between an Arduino development board and a microcontroller chip is that the development board contains almost every essential part needed to start a project on the board itself. It has a microcontroller chip and ports for connection or extension. You can program the chip with inbuilt examples available on Arduino IDE (software) and observe the output right on the board, or by interfacing with external circuits.

On the other hand, for a single microcontroller chip (such as Atmega328 microcontroller), you need a lot of things. First, you need to buy a separate programmer board to program the chip. For building a project or experimentation with the chip, you need a breadboard or veroboard for circuit wiring/connections with other components, a DC power supply for the project, jumper wires, etc. You also need to search for program examples from various sources, books, and the Internet.

Comparison of Arduino with PLC

An Arduino is a microcontroller development board that has many functionalities. These capabilities are enabled by the assortment of peripherals found on the board (besides the microcontroller chip), such as input/output pins, USB port, etc. The whole purpose of the Arduino is to make coding, electronics, and circuitry available, understandable, and accessible to everyone.

A PLC (Programmable Logic Controller) is basically a more robust version of the Arduino that's meant to handle tasks better than the Arduino. For instance, it can work in harsh conditions whereas the Arduino cannot. PLC is mainly used for industrial purposes.

Arduino	PLC
Needed External Components to Work as PLC	Does not need additional external components
Universally accepted	Promoted mainly in Industries
Low cost	High cost
Need to learn basic programming to rewrite the Arduino program	Only needed basic operating technique for reprogramming the PLC
Reprogramming is relatively difficult	Reprogramming is relatively easy
Satisfactory performance	High performance
Cannot work in harsh conditions	Can work on harsh conditions
Compact and Small	Bulky and heavy

More communication options	Less communication options
Easy to replace and repair	Hard to replace and repair
Lesser options for choosing	Many options for choosing

Table :Differences between Arduino and PLC