

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/326316390>

Working Principle of Arduino and Using it as a Tool for Study and Research

Conference Paper · July 2018

DOI: 10.5121/ijcacs.2016.1203

CITATIONS

144

READS

91,212

1 author:



Leo Louis

2 PUBLICATIONS 145 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Implementation of closed loop based scan mechanism [View project](#)

WORKING PRINCIPLE OF ARDUINO AND USING IT AS A TOOL FOR STUDY AND RESEARCH

Leo Louis¹

¹Department of Electronics and Communication Engineering, Gujarat Technological University, Ahmedabad, India

ABSTRACT

This paper explores the working principle and applications of an Arduino board. This also explores on how it can be used as a tool for study and research works. Arduino board can provide a quick tool in development of VLSI test bench especially of sensors. Main advantages are fast processing and easy interface. Today, with increasing number of people using open source software and hardware devices day after day, technology is forming a new dimension by making complicated things look easier and interesting. These open sources provide free or virtually low costs, highly reliable and affordable technology. This paper provides a glimpse of type of Arduino boards, working principles, software implementation and their applications.

KEYWORDS

Arduino, Microcontroller, Hardware, Software, Open-source platform, VLSI, Sensors

1. INTRODUCTION

Arduino is an open source microcontroller which can be easily programmed, erased and reprogrammed at any instant of time. Introduced in 2005 the Arduino platform was designed to provide an inexpensive and easy way for hobbyists, students and professionals to create devices that interact with their environment using sensors and actuators. Based on simple microcontroller boards, it is an open source computing platform that is used for constructing and programming electronic devices. It is also capable of acting as a mini computer just like other microcontrollers by taking inputs and controlling the outputs for a variety of electronics devices.

It is also capable of receiving and sending information over the internet with the help of various Arduino shields, which are discussed in this paper. Arduino uses a hardware known as the Arduino development board and software for developing the code known as the Arduino IDE (Integrated Development Environment). Built up with the 8-bit Atmel AVR microcontroller's 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.

Unlike the other microcontroller boards in India, the Arduino boards entered the electronic market only a couple of years ago, and were restricted to small scale projects only. People associated with electronics are now gradually coming up and accepting the role of Arduino for their own projects. This development board can also be used to burn (upload) a new code to the board by simply using a USB cable to upload. The Arduino IDE provides a simplified integrated platform which can run on regular personal computers and allows users to write programs for Arduino using C or C++.

With so many Arduino boards available in the market, selecting a particular development board needs a variety of survey done with respect to their specifications and capabilities, which can be used for the project execution according to its specified applications.

2. NEED FOR ARDUINO

Why is there a need to use Arduino in specific? or What makes it different from others?

Massimo Banzi, a Co-founder of Arduino mentions some very important reasons for this question.

- 1) **Active User Community:** A group of people using a similar product can hold posted message conversations and share their experiences or solve the problems of the other users in the communities with their own experiences [1].

“If you start charging for everything, everything dies very quickly.” says Banzi, Arduino Co-founder.

- 2) **Growth of Arduino:** Arduino was developed with intent to provide an economical and trouble-free way for hobbyists, students and professionals to build devices that interact with their situation using sensors and actuators. This makes it perfect for newcomers to get started quickly [1].
- 3) **Inexpensive Hardware:** Since Arduino is an open source platform the software is not purchased and only the cost of buying the board or its parts is incurred, thus making it very cheap. The hardware designs are also available online for free from its official website [1].
- 4) **Arduino Board as a Programmer:** To make Arduino board function easy and also making it available everywhere these boards come with a USB cable for power requirements as well as functioning as a programmer [1].
- 5) **Multi-platform Environment:** The Arduino IDE is capable of running on a number of platforms including Microsoft, Linux and Mac OS X making the user community even larger [1].

3. TYPE OF ARDUINO BOARDS

Arduino boards are available with many different types of built-in modules in it. Boards such as Arduino BT come with a built-in Bluetooth module, for wireless communication. These built-in modules can also be available separately which can then be interfaced (mounted) to it. These modules are known as Shield.

Some of the most commonly used Shields are:

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

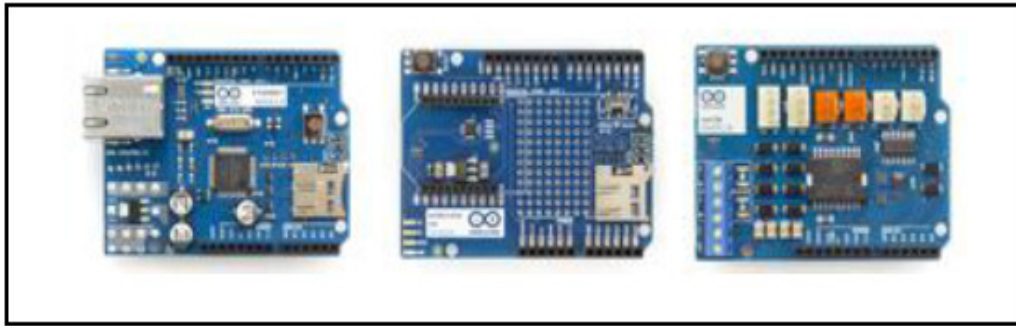


Fig. 1.Arduino Shields – Ethernet, Wireless and Motor Driver.

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

Table 1. Heading and text fonts

<i>Arduino Type</i>	<i>Microcontroller</i>	<i>Clock Speed</i>
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 Ethernet	Equivalent to Arduino UNO with an Ethernet shield	
Arduino Fio.	ATmega328	8 MHz with auto-reset
Arduino BT w/ ATmega328	ATmega328	16 MHz with auto-reset
LilyPad Arduino w/ ATmega328	ATmega328	8 MHz (3.3V) with auto-reset
Arduino Pro or Pro Mini	ATmega328	16 MHz with auto-reset
Arduino NG	ATmega8	16 MHz with auto-reset

4. ELEMENTS OF ARDUINO BOARDS

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

- *Hardware*
- *Software*

4.1. Hardware

The Arduino Development Board consists of many components that together makes it work. Here are some of those main component blocks that help in its functioning:

- **Microcontroller:** This is the heart of the development board, which works as a mini computer and can receive as well as send information or command to the peripheral devices connected to it. The microcontroller used differs from board to board; it also has its own various specifications.
- **External Power Supply:** This power supply is used to power the Arduino development board with a regulated voltage ranging from 9 – 12 volts.

- **USB plug:** This plug is a very important port in this board. It is used to upload (burn) a program to the microcontroller using a USB cable. It also has a regulated power of 5V which also powers the Arduino board in cases when the External Power Supply is absent.
- **Internal Programmer:** The developed software code can be uploaded to the microcontroller via USB port, without an external programmer.
- **Reset button:** This button is present on the board and can be used to resets the Arduino microcontroller.
- **Analog Pins:** There are some analog input pins ranging from A0 – A7 (*typical*). These pins are used for the analog input / output. The no. of analog pins also varies from board to board.
- **Digital I/O Pins:** There are some digital input pins also ranging from 2 to 16 (*typical*). These pins are used for the digital input / output. The no. of these digital pins also varies from board to board.
- **Power and GND Pins:** There are pins on the development board that provide 3.3, 5 volts and ground through them

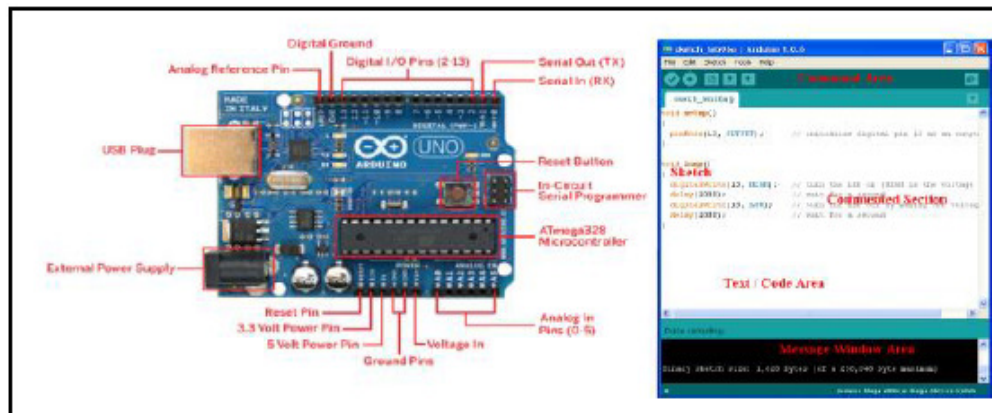


Fig. 2.A labeled diagram of an Arduino Board and an IDE.

4.2. Software

The program code written for Arduino is known as a sketch. The software used for developing such sketches for an Arduino is commonly known as the Arduino IDE. This IDE contains the following parts in it:

- **Text editor:** This is where the simplified code can be written using a simplified version of C++ programming language.
- **Message area:** It displays error and also gives a 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 contains various buttons like Verify, Upload, New, Open, Save and Serial Monitor. On the bottom right hand corner of the window there displays the Development Board and the Serial Port in use.

4.3. Features of Arduino IDE

- The project file or the sketches for a project are saved with the file extension .ino
- Features such as cut / copy / paste are supported in this IDE.
- There also is a facility for finding a particular word and replacing it with another by pressing the Ctrl + F buttons on the keyboard
- The most basic part or the skeleton of all Arduino code will have two functions

5. PROGRAMMING BASICS

Now we'll discuss about the programming techniques of Arduino sketch in the Arduino IDE. There are two main parts every sketch will always have, they are:

- void setup ()
- void loop ()

1) void setup():

This is the first routine that begins when the Arduino starts functioning. This function is executed only once throughout the entire program functioning.

The setup function contains the initialization of every pin we intend use in our project for input or output. Here is an example of how it should be written:

```
void setup()
{
  pinMode(pin, INPUT);
  pinMode(pin, OUTPUT);
}
```

Here the pin is the no. of the pin that is to be defined. INPUT / OUTPUT correspond to the mode in which the pin is to be used.

```
void setup()
{
  Serial.begin(9600);
}
```

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.

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

2) void loop():

This function is the next important function in the Sketch. It consists of that part of the code that needs to be continuously executed unlike the part of the code written in the setup function.

An example of a void loop is as follows:

```
void loop()
{
  digitalWrite(pin, HIGH);
}
```

Here digital Write is a function that writes a high or a low value to a digital pin. If the pin has been configured as an OUTPUT with pin Mode(), its voltage will be set to the corresponding value: 5V (or 3.3V on 3.3V boards) for HIGH, 0V (ground) for LOW.

Similarly if there is a need for delay in the sketch then there is another function that creates a delay in the execution of the code

```
delay(1000); //delay for a second
```

This creates a delay in the execution of the program for the time period specified (in milliseconds). Using the above two function lets create a sketch for blinking a led.

```
// this loop function executes only once
void setup()
{
  pinMode(13, OUTPUT); // initialize digital pin 13 as an output.
}

// this loop function executes forever
void loop()
{
  digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);           // wait for a second
  digitalWrite(13, LOW); // turn the LED off by making the voltage LOW
  delay(1000);           // wait for a second
}
```

Fig. 3.Arduino Shields – Ethernet, Wireless and Motor Driver.

6. APPLICATIONS

Arduino has endless applications as it has been used extensively for creating projects by hobbyist, amateurs and professional in various fields of engineering. Here are some of those amazing projects that have been developed on an Arduino platform:

1. Arduino Satellite (ArduSat)

ArduSat is an open source satellite completely based on Arduino to create a stage for space discoveries. Built by Spire previously known as NanoSatisfi, ArduSat collects various types of information's from the space environment, with the help of numerous sensors that includes temperature sensors, pressure sensors, cameras, GPS, spectrometer, and magnetometer etc with its programmable Arduino processors [4].

This platform also allows common public to experiment their projects in space. ArduSat can be used for photography from space, making a spectrograph of the sun, detecting high energy radiation, compiling temperature readings and observing meteors etc. [5] [6].

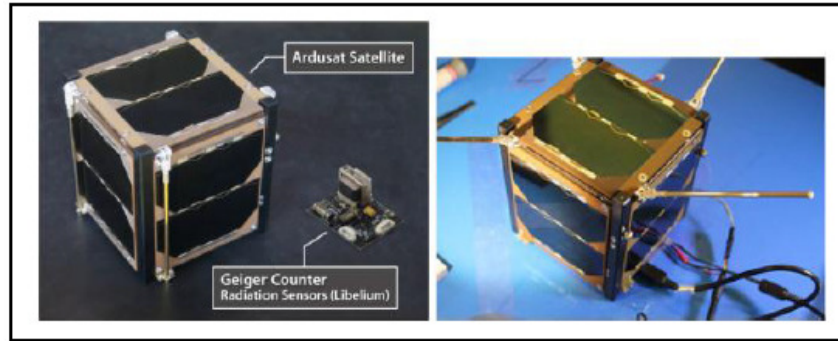


Fig. 4. Actual model of an ArduSat

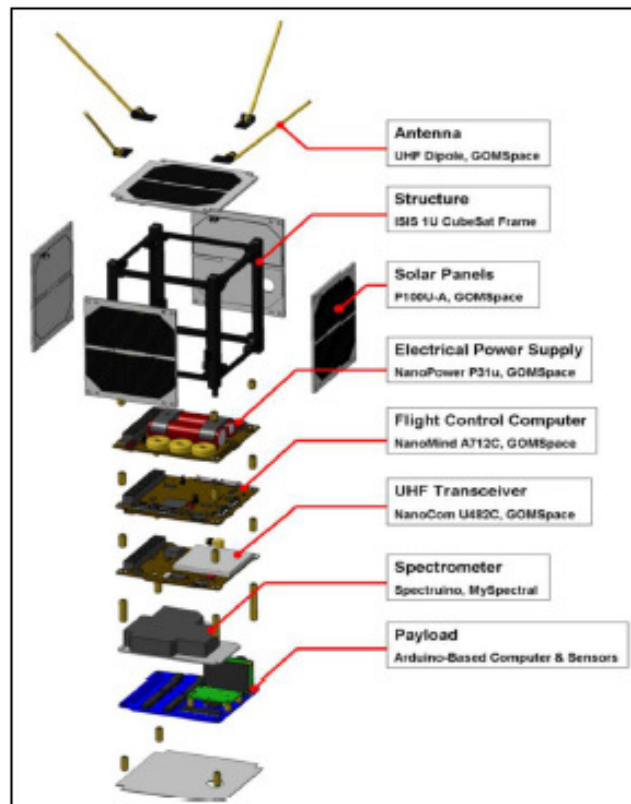


Fig. 5. ArduSat and its major components.[7]

2. ArduPilot (ArduPilotMega - APM)

ArduPilot is an unmanned aerial vehicle (UAV) based on the open source platform and built using Arduino Mega which is able to control independent multicopters, fixed-wing aircraft, traditional helicopters and ground rovers. [3]

It was created by the DIY Drones community in 2007 and was also an award winning platform of 2012 [3].

3. Lilypad Arduino

Just like the Google wearable's, Lilypad Arduino is a wearable version of Arduino developed and designed by Leah Buechley and SparkFun Electronics with the aim of building interactive electronic textiles or e-textiles [10].

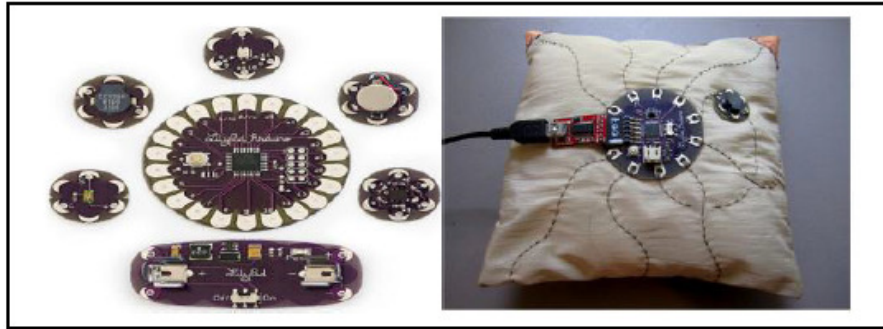


Fig. 6. Lily Arduino and so add-on components that can be sewn together and a Lilypad Arduino pillow.

These designs involve use of a number of modules sewn on the cloth with conductive thread to give it the required electrical connections. [10] The microcontroller can be programmed just like the normal Arduino boards using the usb-to-serial connection. This is a perfect illustration of a user community determined project with the business version of the kit designed by Leah and SparkFun Electronics.

This is an example of such designs, here is a jacket with turn signals that will let people know where you're going when on a bike.

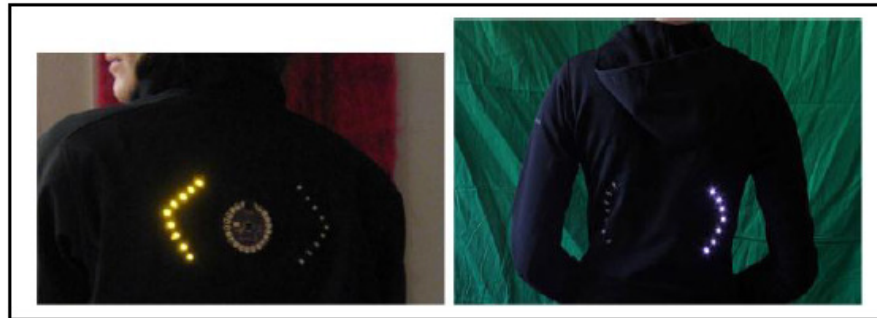


Fig. 7. Examples of jackets with turn signals powered by Lilypad Arduino sewn onto them

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 learnt how to write sketches for Arduino in its own IDE (software). Developing new ideas with Arduino is endless, with the help of this paper we have learnt 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 is infinite. Though it does have its own limitations, it is a great tool that 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. Brodtkin, '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].

AUTHORS

Leo Louis is currently in the final year of Bachelors of Engineering in Electronics and Communication from the Gujarat Technological University, India. He received his Diploma in Electronics and Communication from the Gujarat Technological University, India in 2012. His research interests include Automation and Robotics using Microcontrollers and Embedded Systems.

