RESEARCH WORK-4

**\*Why arduino is popular and why still it is not commercially used?**

The most important factor is that the entry barrier is very low - you can get from unpacking the Arduino to having your first LED to blink in less than 5 minutes. The programming environment and the language itself are very easy to understand even for people who have not programmed before, but it still allows more advanced folks to dig deeper into the possibilities. because it is more available in the market and cheaper than other boards. Arduino has an easy environment to code and test.

## There are many other microcontrollers and development boards, but Arduino’s accessibility, easy-to-understand hardware design, and simple software make it appealing to different types of users. There are also other reasons why Arduino is the preferred choice for its thousands of followers despite the emergence of many other competitors: Affordable, **Cross-platform,** **Easy to Program,** **Broad Libraries of Codes** ,**Open-Source Software and Hardware.**

## Arduinos are intended for experimentation and learning, often with breadboards and loose wires that eventually break if vibrated. The boards are not protected against harsh, dirty or electrically noisy environments. This sort of setup is unreliable, and not suitable for industrial operation.

## \* List out advantages and disadvantages of Arduino.

## *Advantages of Arduino*

**Easy to Use**

Arduino is super easy to learn. Any beginner can learn Arduino with little or no knowledge of programming. It uses a simplified version of C/C++ language which is easily adaptable. It is an ideal board for beginners. Arduino is a plug and play electronic board with a simple and minimalistic hardware interface.

**Inexpensive Hardware**

Arduino boards are available at low cost. To start building our first program we just need an Arduino board as its development tool **IDE** is freely available. There is no need for any external hardware, multiple projects can be made using single Arduino boards which cut overall project cost by a great margin.

**Active User Community**

Arduino is an open-source community with many online supports available on Arduino forum. Thanks to open-source nature of Arduino projects, as a lot of work on multiple topics has already been done. Multiple Arduino projects starting from fan controllers to smart homes have been designed already.

**Cross Platform Support**

Arduino is not only limited to Windows it is also available across multiple platforms like Linux and macOS. Most of the microcontrollers can only be programmed using Windows but not in case with Arduino.

**Tons of Libraries**

Many Arduino libraries are available to control external sensors, modules, and motors. Functionality of Arduino boards can be extended using these libraries. Libraries provide the user free hand to play with hardware and manipulate data easily.

***Disadvantages of Arduino***

**Lack of Multitasking**

Arduino boards are limited to run only a single program at a time. Other competitor boards like Raspberry Pi offers multitasking functionality. Like multicore CPUs that can run multiple programs without slowing the speed of the overall system, Arduino lacks this ability and we must close one sketch to execute the other one.

**Not Optimized for Performance**

Microcontrollers used in most Arduino boards are not ready to give their full performance. The Arduino development environment is optimized for beginners so they can easily create sketches. All this optimization comes at the cost of reducing the microcontroller’s overall power ability. If the same microcontroller is used with AVR development the performance will be increased by manifolds.

**Lack of Communications**

Arduino boards are limited in terms of Bluetooth and Wi-Fi support. Popular Arduino boards like UNO don’t have in-built communications support; we must interface external hardware modules to enable these functions. Arduino ships few boards with these technologies, but overall cost increased compared to other boards available in the market.

**Limited Support for Programming Languages**

Arduino boards are mostly programmed in C or C++ language. Arduino lacks support for some very famous languages like Java, Python and JavaScript. Although Arduino doesn’t directly support Python with IDE but can be programmed using some open-source libraries like **pySerial.**

**Less Memory Storage Capacity**

One of the main features which Arduino lacks is it has limited memory storage. Arduino UNO has 2kb of SRAM and 32kb of flash memory which can only store sketches with hundreds of lines. Due to this Arduino has limited scope in the robotics world and cannot be used in industrial scale projects.

**\*Does Arduino uses c/c++ ? Describe it.**

The Arduino language is a subset of C/C++, where you can also use assembly for ultra-low level code.

When saying “programming on Arduino”, in fact you don’t program the Arduino board itself, but the microcontroller inside the board. For example, the Arduino Uno has a AtMega328p microcontroller.

And usually, when writing code for microcontrollers, performance matters a lot. That’s why you need to have strong and fast programming languages. C and C++ are among the most powerful languages you can find. Those are quite fast and stable, and thus are a good choice for microcontrollers.

you can use both C and C++ to program the microcontrollers. The Arduino project also provides an Integrated Development Environment (IDE) based on the Processing language project. In fact, you can code an Arduino program in any language for a compiler that produces binary machine code for the target processor. When this compatibility is there, there should not be any issues.