ROBOTICS I

INDEX:

1.INTRODUCTION

Robotics is a branch of engineering that involves the conception, design, manufacture, and operation of robots. This field overlaps with electronics, computer science, artificial intelligence, mechatronics, nanotechnology and bioengineering

Robotics involves design, construction, operation, and use of robots, as well as computer systems for their perception, control, sensory feedback, and information processing. The goal of robotics is to design intelligent machines that can help and assist humans in their day-to-day lives and keep everyone safe.

Robotics develops machines that can substitute for humans and replicate human actions. Robots can be used in many situations and for lots of purposes, but today many are used in dangerous environments (including inspection of radioactive materials, bomb detection and deactivation), manufacturing processes, or where humans cannot survive (e.g. in space, underwater, in high heat, and clean up and containment of hazardous materials and radiation). Robots can take on any form but some are made to resemble humans in appearance. This is said to help in the acceptance of a robot in certain replicative behaviors usually performed by people. Such robots attempt to replicate walking, lifting, speech, cognition, or any other human activity. Many of today's robots are inspired by nature, contributing to the field of bio-inspired robotics.

The concept of creating machines that can operate autonomously dates back to classical times, but research into the functionality and potential uses of robots did not grow substantially until the 20th century. Throughout history, it has been frequently assumed by various scholars, inventors, engineers, and technicians that robots will one day be able to mimic human behavior and manage tasks in a human-like fashion. Today, robotics is a rapidly growing field, as technological advances continue; researching, designing, and building new robots serve various practical purposes, whether domestically, commercially, or militarily. Many robots are built to do jobs that are hazardous to people, such as defusing bombs, finding survivors in unstable ruins, and exploring mines and shipwrecks. Robotics is also used in STEM (science, technology, engineering, and mathematics) as a teaching aid. The advent of nanorobots, microscopic robots that can be injected into the human body, could revolutionize medicine and human health.

2. MACHINE AUTOMATION VS ROBOTICS

Machine automation is any information technology that is designed to control the work of machines. Many modern machines have some form of computer control into them. In many cases, machines are also integrated with business systems that provide control inputs. These machines can perform a specific activity that can not be changed, for example the machines in a production line in a factory; they all are controlled by the factory automation system.

The term **robotics** implies a more sophisticated level of automation such as the ability to respond to sensory perceptions or artificial intelligence that learns and adapts

Machine automation defined as a machine that is controlled by information technology has its **value** for example:

- Improve products and services
- Improve productivity
- Free humans from repetitive and physically demanding or dangerous work

Automation risk is the potential for automation to replace jobs. Machines began to automate work in the 1760s, this process has led to social disruption at times but generally speaking, the jobs have historically been replaced with new professions. Technology appears to improve at an exponential or hyperbolic rate. This means that technology is currently improving at a much greater rate of speed that it did historically. As such , there is a potential for jobs to be replaced faster than people can adjust. This may lead to social instability, political change and economic problems.

- Physical tasks: Robotics and artificial intelligence are approaching the point that the can overcome obstacles and handle unpredictable situations. This allow physical work to be automated
- Health and safety: Areas such as medicine and transport
- Administration: Self service tools, electronic data processing and process automation are likely to continue to reduce administrative workloads.
- Stakeholder Interaction: For example an e-commerce firm that replaces a local shop
- Architecture and design: Design, coding, and engineering were viewed as too complex for automation. However, it is clear that software tools are able to eliminate low level work in these professions and this trend is only likely to accelerate.
- Arts and entertainment: Professions may shift towards creative work, governance, education, arts, entertainment and other areas that a society or culture values as human pursuit (1)

3. CONTROL SYSTEMS

A **control system** manages, commands, directs, or regulates the behavior of other devices or systems using control loops. It can range from a single home heating controller using a thermostat controlling a domestic boiler to large Industrial control systems which are used for controlling processes or machines. A control system may be operated by electricity, by mechanical means, by fluid pressure (liquid or gas), or by a combination of means. When a computer is involved in the control circuit, it is usually more convenient to operate all of the control systems electrically, although in mixtures are fairly common.

Control systems are used to enhance production, efficiency and safety in many areas, including: Agriculture, Industry, Electricity power plants, mines, pharmaceutical manufacturing, etc

A control system is a system, which provides the desired response by controlling the output. The following figure shows the simple block diagram of a control system.

Here, the control system is represented by a single block. Since, the output is controlled by varying input, the control system got this name. We will vary this input with some mechanism. In the next section on open loop and closed loop control systems, we will study in detail about the blocks inside the control system and how to vary this input in order to get the desired response.

Examples :

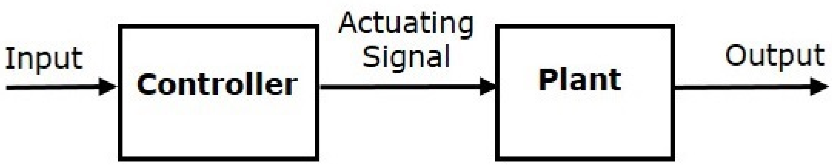
1. Traffic lights control system, washing machine . In the traffic lights control system, a sequence of input signals is applied to this control system and the output is one of the three lights that will be on for some duration of time. During this time, the other two lights will be off. Based on the traffic study at a particular junction, the on and off times of the lights can be determined. Accordingly, the input signal controls the output. So, the traffic lights control system operates on a time basis.
2. A simple weather station can be looked upon as the following:
   - Input: The heat from the sun causes the temperature sensor to produce data and this is sent to the computer.
   - Process: Data is received by the computer and it is processed. The processed data is displayed on the monitor as a graph.
   - Output: The temperature levels are printed out. This is one form of output.

3.1. Types of control systems

There are **two main types of control loops:** Open loops, which operate with human input, and closed loops, which are fully autonomous. Some loops can be switched between closed and open modes. When open, a switchable loop is manually controlled and when closed it is fully automated.

Open Loops:

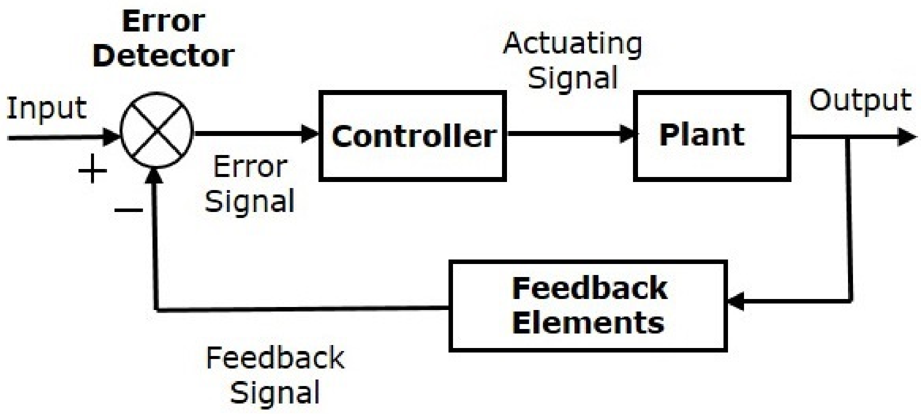
an input is applied to a controller and it produces an actuating signal or controlling signal. This signal is given as an input to a plant or process which is to be controlled. So, the plant produces an output, which is controlled. The traffic lights control system which we discussed earlier is an example of an open loop control system. (2)



Close Loops:

In closed loop control systems, output is fed back to the input. So, the control action is dependent on the desired output.

The following figure shows the block diagram of negative feedback closed loop control system.



The error detector produces an error signal, which is the difference between the input and the feedback signal. This feedback signal is obtained from the block (feedback elements) by considering the output of the overall system as an input to this block. Instead of the direct input, the error signal is applied as an input to a controller.

So, the controller produces an actuating signal which controls the plant. In this combination, the output of the control system is adjusted automatically till we get the desired response. Hence, the closed loop control systems are also called the automatic control systems. Traffic lights control system having sensor at the input is an example of a closed loop control system.(2)

The differences between the open loop and the closed loop control systems are mentioned in the following table.

Open Loop Control Systems	Closed Loop Control Systems
Control action is independent of the desired output.	Control action is dependent of the desired output.
Feedback path is not present.	Feedback path is present.
These are also called as **non-feedback control systems**.	These are also called as **feedback control systems**.
Easy to design.	Difficult to design.

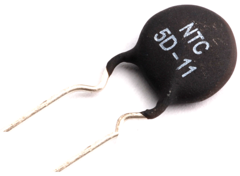
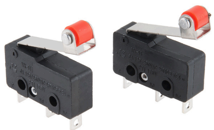
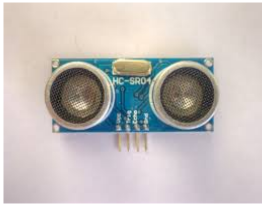
These are economical.	These are costlier.
Inaccurate.	Accurate.

3.2 Elements of control systems:

Control systems are composed by 3 main elements:

Sensors: A sensor is defined as a device that converts a physical stimulus into a readable output. The role of a sensor in a control and automation system is to detect and measure some physical effect, providing this information to the control system

The most common sensors in robotics are

			
LDR Light dependent resistor	NTC Temperature sensor	Limit sensor	Ultrasonic Sensor

Controller card / board : The controller card or board, or simply "controller," is a piece of hardware that acts as the interface between the motherboard and the other components of the computer. In this unit we are going to work with Arduino.

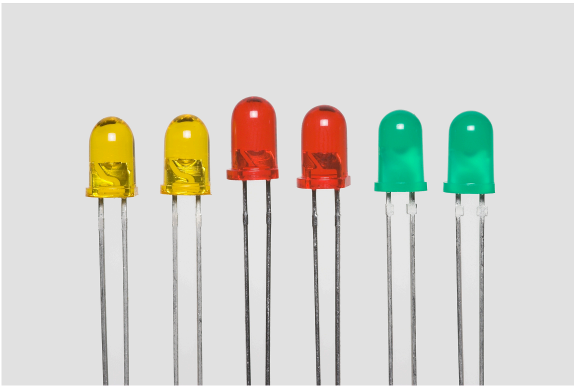
Actuators : An **actuator i**s a component of a machine that is responsible for moving and controlling a mechanism or system, for example by opening a valve. In simple terms, it is a "mover". An actuator requires a control signal and a source of energy. The control signal is relatively low energy and may be electric voltage or current, pneumatic or hydraulic pressure, or even human power. Its main energy source may be an electric current, hydraulic fluid pressure, or pneumatic pressure. When it receives a control signal, an actuator responds by converting the source's energy into mechanical motion. An actuator is the mechanism by which a control system acts upon an environment. The control system can be simple (a fixed mechanical or electronic system), software-based (e.g. a printer driver, robot control system), a human, or any other input

Some of the actuators :

LED: We have learned about LEDs in the electronic lesson.

A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration.[1] It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors.

Servomotors are not a specific class of motor, although the term *servomotor* is often used to refer to a motor suitable for use in a closed-loop control system. Servomotors are used in applications such as robotics, CNC machinery or automated manufacturing.

	
LED	Servomotor

4. WORKING WITH ARDUINO CONTROLLER BOARD

4.1. What is Arduino?

Arduino is an open-source electronics platform based on easy-to-use hardware and software. It's intended for anyone making interactive projects. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing. (3)

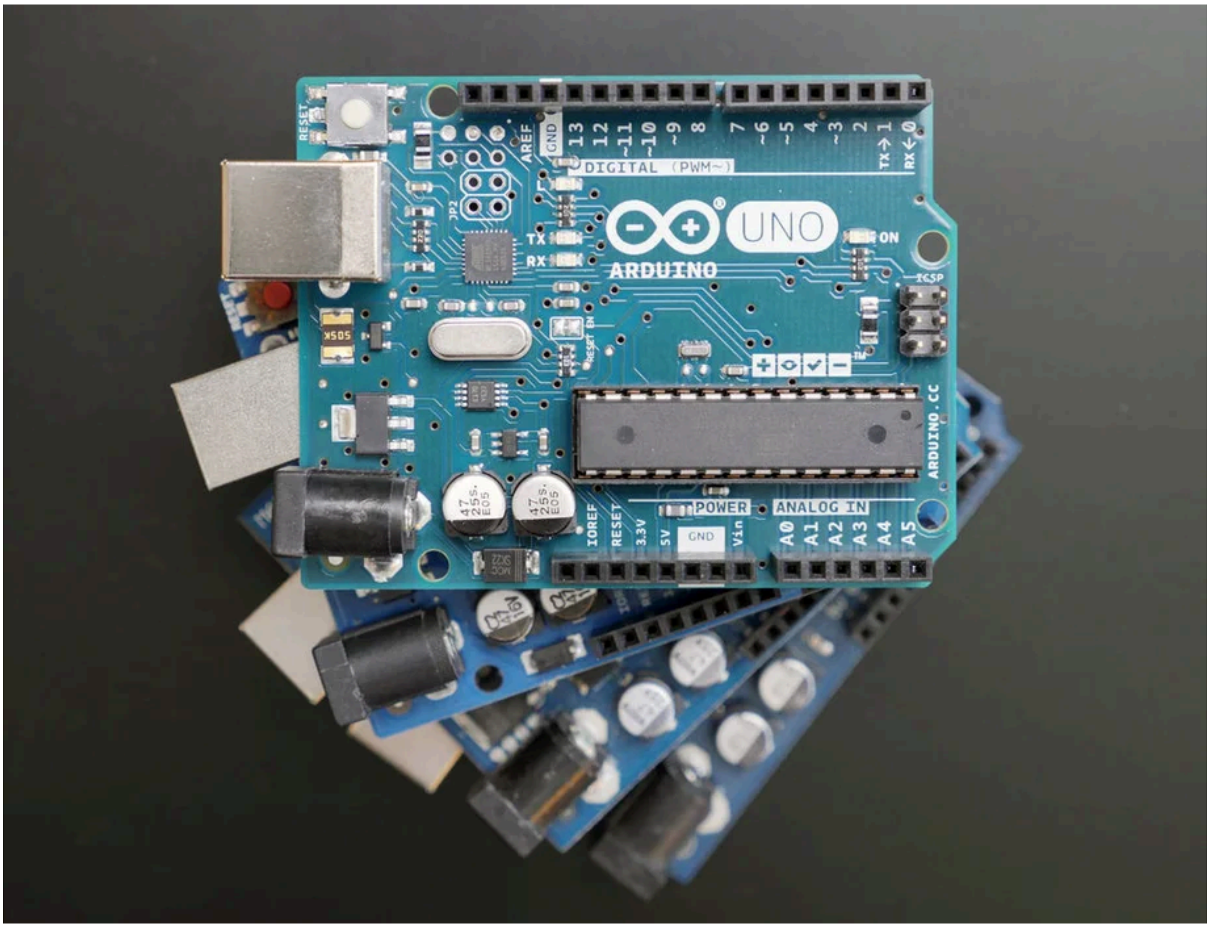
Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

Why Arduino;

- **Inexpensive** - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50
- **Cross-platform** - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
- **Simple, clear programming environment** - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing

programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.

- **Open source and extensible software** - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.
- **Open source and extensible hardware** - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the [bradboard version of the module] in order to understand how it works and save money.[e]



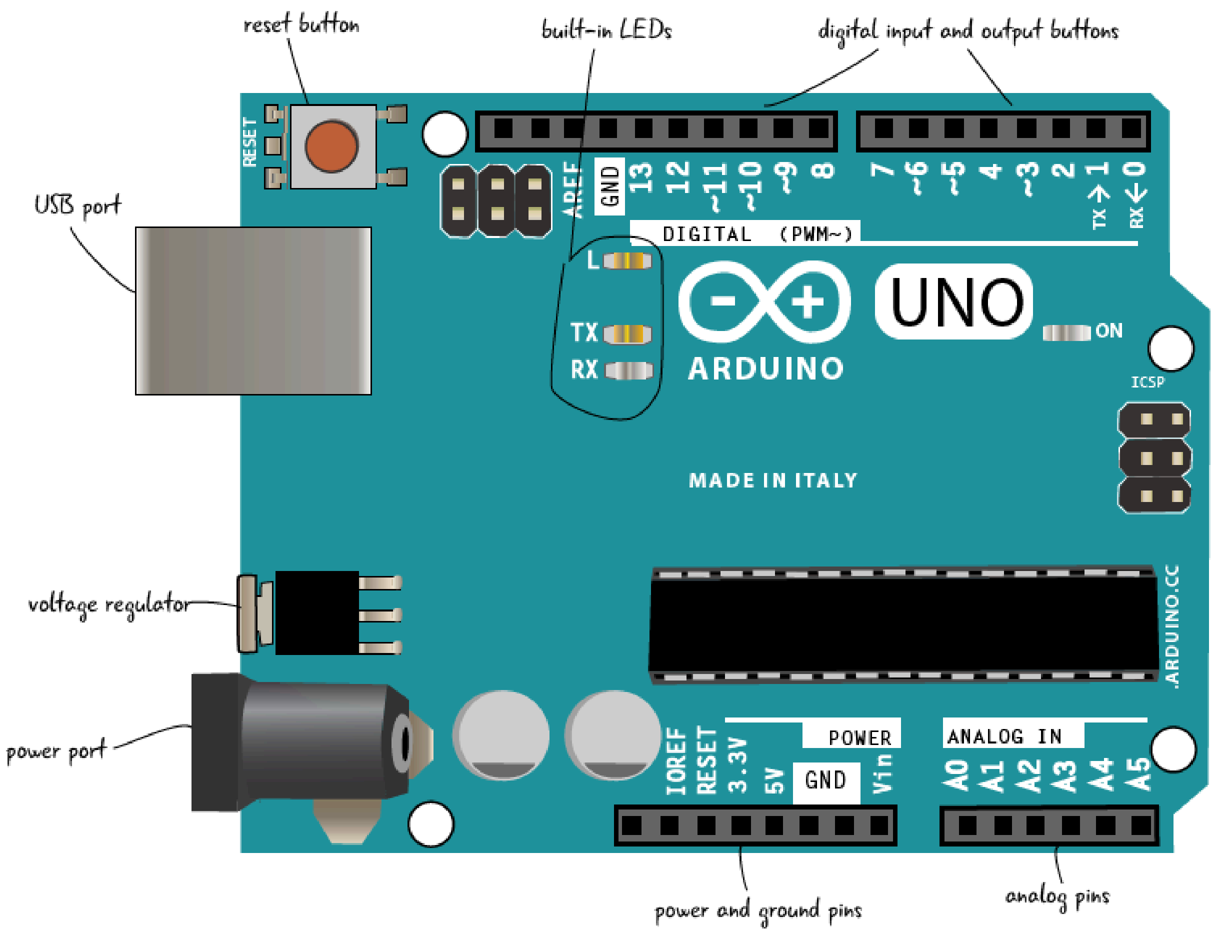
Picture from (4)

There are several versions of the Arduino as it has been around since 2005 and is constantly evolving. For the purpose of this book, we are concerned with the Arduino Uno. Your Arduino might not look exactly like this, as we have simplified the drawing in order to point out the sections that concern us. Since the Arduino is open source, you might also purchase a board that does not come directly from the Arduino organization. Just know that for this book we are focused on the Arduino Uno and compatible boards.

We will also need some additional electronic parts and a few tools to build projects with the Arduino. The most common ones are:

- Breadboard
- USB A-B cable
- 9 volt battery
- 9-12 volt power supply
- 9 volt battery cap or holder
- Assorted LEDs, a variety of colors
- Potentiometer
- Switches/buttons
- Photo resistor
- Speaker
- Servomotor
- Jumper wires

4.2. Parts of Arduino?



Picture from (5)

Reset Button: Much like turning your computer off and on again, some problems with the Arduino can be solved by pushing the reset button. This button will restart the code currently uploaded on your Arduino. The reset button may be in a different location on your board than in this diagram, but it is the only button.

USB Port: The USB port takes a standard A-to-B USB cable, often seen on printers or other computer peripherals. The USB port serves two purposes: First, it is the cable connection to a computer which allows you to program the board. The USB cord will also provide power for the Arduino if you're not using the power port (described below).

Voltage Regulator: The voltage regulator converts power plugged into the power port (described below) into the 5 volts and 1 amp standard used by the Arduino. BE CAREFUL! This component gets very hot.

Power Port: The power port includes a barrel-style connector which allows for either power straight from a wall source (often called a wall-wart) or from a battery. This power is used instead of the USB cable. The Arduino can take a wide range of voltages (5V – 20V) but will be damaged if power higher than that is connected.

We'll take a closer look at the other side of the board now.

USB Port: The USB port takes a standard A-to-B USB cable, often seen on printers or other computer peripherals. The USB port serves two purposes: First, it is the cable connection to a computer which allows you to program the board. The USB cord will also provide power for the Arduino if you're not using the power port (described below).

Voltage Regulator: The voltage regulator converts power plugged into the power port (described below) into the 5 volts and 1 amp standard used by the Arduino. BE CAREFUL! This component gets very hot.

Power Port: The power port includes a barrel-style connector which allows for either power straight from a wall source (often called a wall-wart) or from a battery. This power is used instead of the USB cable. The Arduino can take a wide range of voltages (5V – 20V) but will be damaged if power higher than that is connected.

We'll take a closer look at the other side of the board now. (5)

4.3.Steps for working with Arduino

1. Download and install arduino software
2. Plug your arduino into your computer with the usb cable. The green pin on the board will light up. Select the option for the automating installation of the software
3. Select the controllator option and select the drivers
4. Check the communication is correctly done.
5. Upload a program to the arduino board. I recommend to start with the Blink for example (see activity below)
6. Build the circuit in your breadboard and connect it to your arduino.

5. VOCABULARY

- Arduino
- Automated machine
- control loop
- control system
- Controller board
- Controller card

- Debugging
- Loop
- Robot
- Sketch

6. CREDITS AND RESOURCES

This unit was created by Stella Carrera under creative commons by sa license

Resources and links (en)

(1) https://simplicable.com/new/automation-risk
(2) https://www.tutorialspoint.com/control_systems/control_systems_introduction.htm
(3) https://www.arduino.cc/
(4) https://www.instructables.com/class/Arduino-Class/
(5) https://arduinotogo.com