**CHAPTER 1**

**Introduction and Mission**

* 1. **Introduction**

Cubiccode is one of the fastest and consistently growing companies in the Web Development, Digital Marketing, Growth Hacking, Mobile Application Development, Software Solutions & IT Service sector with hundreds of happy clients across the nation. We are a professional organization with wide experience in this sector providing the best solution for businesses to help them grow. One of the best web designing & development and Digital Marketing Company in Belgaum. Cubiccode was recognized in the Top 10 web development startup of 2019 by Silicon India.

Cubiccode is one of the fastest and consistently growing companies in the Web Development, Digital Marketing, Growth Hacking, Mobile Application Development, Software Solutions & IT Service sector with hundreds of happy clients across the nation. We are a professional organization with wide experience in this sector providing the best solution for businesses to help them grow. One of the best web designing & development and Digital Marketing Company in Belgaum. Currently, we serve in fourteen different cities across the nation. We aim to lead not just the creative world but also the technology world with our imaginative, passionate and dedicated team of twenty-three young guns. We are happy and grateful to have built a good foundation for our customers in the early stage of our business career. At Cubiccode, we are a unique captivating mix of Innovativeness and Diligence with the additional sprinkle of creativity! Cubiccode was recognized in the Top 10 web development startup of 2019 by Silicon India.

* 1. **Mission**

Our mission to provide considerable results that turn traffic into business. We understand that each and every business is different and the demands of every client are individualistic and so we tailor their strategies to suit the needs of the clients with the intention of maximizing the effectiveness for all digital solutions.

**CHAPTER 2**

**Embedded systems and IOT Activities**

**2.1Embedded systems**

An embedded system is a controller programmed and controlled by a real-time operating system (RTOS) with a dedicated function within a larger mechanical or electrical system, often with real-time computing constraints. It is embedded as part of a complete device often including hardware and mechanical parts. Embedded systems control many devices in common use today. Ninety-eight percent of all microprocessors manufactured are used in embedded systems.

Examples of properties of typical embedded computers when compared with general-purpose counterparts are low power consumption, small size, rugged operating ranges, and low per-unit cost. This comes at the price of limited processing resources, which make them significantly more difficult to program and to interact with. However, by building intelligence mechanisms on top of the hardware, taking advantage of possible existing sensors and the existence of a network of embedded units, one can both optimally manage available resources at the unit and network levels as well as provide augmented functions, well beyond those available. For example, intelligent techniques can be designed to manage power consumption of embedded systems.

Embedded systems are widely used because of their compact size, efficiency and because of the reason that they can be used in variety of versions of a particular product. The embedded system works properly because of the usage of microprocessors. They are used for control, user interface, signal processing and many other tasks.

Microprocessor is nothing but a complete CPU on a single chip. So, an embedded system is any device that includes a programmable computer but is not itself. The microprocessors help us to make everything work on a single chip. The reason to use them is that they allow the designers to create multiple versions of a single product.

The design process of embedded systems includes the following steps:

1. Requirements
2. Specification
3. Architecture
4. Components
5. System Integration
6. Testing

The design process is very important to understand how the embedded systems’ design is laid down. Requirements is the phase where the user requirements are collected in an informal way. The needs are just scribbled down and then further sent to specification phase where the requirements are neatly written down as a formal description. The mock-ups are done for the requirement validation in the first phase.

The architecture phase involves building of models which depicts the end phase. Once the model is approved by user the components phase is implemented where we gather the components and in the system integration phase the components are assembled to design the final product. Components may be hardware or software.

CHARACTERISTICS:

1. An embedded system is software embedded into computer hardware that makes a system dedicated to be used for variety of application.
2. Embedded system generally used for do specific task that provide real-time output on the basis of various characteristics of an embedded system.
3. Embedded system may contain a smaller part within a larger device that used for serving the more specific application to perform variety of task using hardware-software intermixing configuration.
4. It provides high reliability and real-time computation ability.
5. It is used the most because of the compact size and more efficiency.
   1. IOT

IoT is creating a giant network where all the devices are connected to each other and providing them with the capability to interact with each other. This is driving the automation to a next level where devices will communicate with each other and make decisions on their own with less or without any human interventions. The person who coined the term “Internet of Things “. The term “The Internet of Things” (IoT) was coined by Kevin Ashton in a presentation to Proctor & Gamble in 1999.

The ‘Thing’ in IoT can be any device with any kind of built-in-sensors with the ability to collect and transfer data over a network with less or without manual intervention.

The embedded technology in the object helps them to interact with internal states and the external environment, which in turn helps in decisions making process.

In a nutshell, IoT is a concept that connects all the devices to the internet and let them communicate with each other over the internet. IoT is a giant network of connected devices – all of which gather and share data about how they are used and the environments in which they are operated. By doing so, each of your devices will be learning from the experience of other devices, as humans do. IoT is trying to expand the interdependence in human- i.e. interact, contribute and collaborate to things.

IoT systems have applications across industries through their unique flexibility and ability to be suitable in any environment. They enhance data collection, automation, operations, and much more through smart devices and powerful enabling technology.

IoT ADVANTAGES:

The advantages of IoT span across every area of lifestyle and business. Here is a list of some of the advantages that IoT has to offer −

* Improved Customer Engagement − Current analytics suffer from blind-spots and significant flaws in accuracy; and as noted, engagement remains passive. IoT completely transforms this to achieve richer and more effective engagement with audiences.
* Technology Optimization − The same technologies and data which improve the customer experience also improve device use, and aid in more potent improvements to technology. IoT unlocks a world of critical functional and field data.
* Reduced Waste − IoT makes areas of improvement clear. Current analytics give us superficial insight, but IoT provides real-world information leading to more effective management of resources.
* Enhanced Data Collection − Modern data collection suffers from its limitations and its design for passive use. IoT breaks it out of those spaces, and places it exactly where humans really want to go to analyze our world. It allows an accurate picture of everything.
* IoT DISADVANTAGES:
* Though IoT delivers an impressive set of benefits, it also presents a significant set of challenges. Here is a list of some its major issues −
* Security − IoT creates an ecosystem of constantly connected devices communicating over networks. The system offers little control despite any security measures. This leaves users exposed to various kinds of attackers.
* Privacy − The sophistication of IoT provides substantial personal data in extreme detail without the user's active participation.
* Complexity − Some find IoT systems complicated in terms of design, deployment, and maintenance given their use of multiple technologies and a large set of new enabling technologies.
* Flexibility − Many are concerned about the flexibility of an IoT system to integrate easily with another. They worry about finding themselves with several conflicting or locked systems.
* Compliance − IoT, like any other technology in the realm of business, must comply with regulations. Its complexity makes the issue of compliance seem incredibly challenging when many consider standard software compliance a battle.

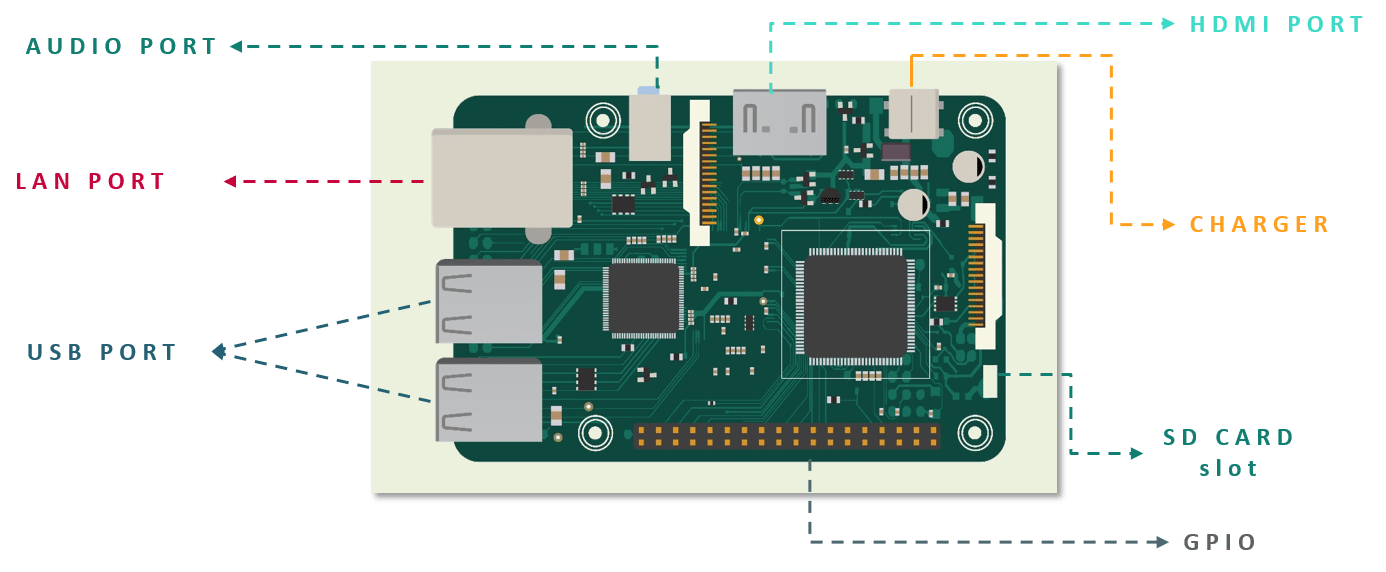
**CHAPTER 3**

**Raspberry Pie and Internship Activities**

**3.1 Raspberry Pie**

The Raspberry Pi is a small credit-card sized computer developed in the United Kingdom by the Raspberry Pi Foundation. Their primary motive was to support & promote the teaching of basic computer science in schools and in developing countries. Far more than expected, it did wonders in many fields, such as robotics. More than 5 million Raspberry Pi have been sold before February 2015, according to the Raspberry Pi Foundation.

The current version is RPi 3 which was released in February 2016. The below figure gives a physical demonstration of it.



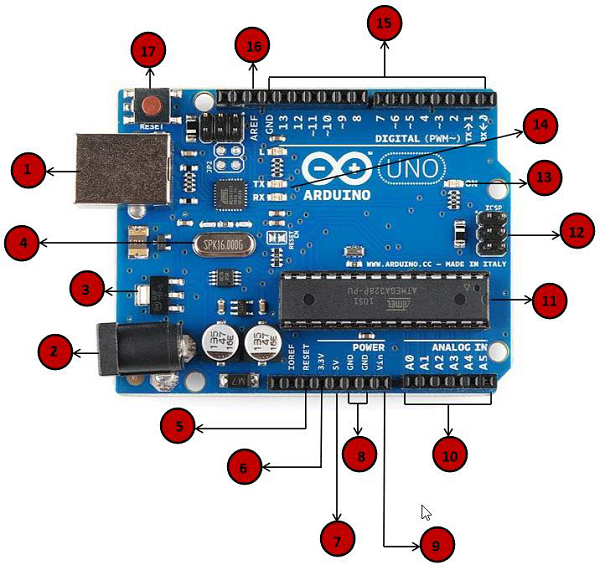
CPU speed ranges from 700 MHz to 1.2 GHz for the Pi 3 & has 1 GB of RAM. SD cards are used to store the operating system and program memory. It has four USB slots, HDMI and composite video output, and a 3.5 mm phone jack for audio. For lower level output it has a number of GPIO pins which support common protocols like I²C. Pi 3 is also equipped with Wi-Fi 802.11n and Bluetooth.

It is used for multiple purposes. It could be used as a general computer, for browsing the internet, playing HD videos, making spreadsheets & word-processing or playing games. But nowadays it is mostly being used for making IoT projects like infra-red cameras, security systems, music machines & detectors for weather stations. As we earlier discussed, Raspberry Pi is getting immense popularity in the field of robotics due to its portable size and good processing power required for standalone systems. There is a list of operating systems supported by Raspberry Pi like RISC OS Pi, FreeBSD, NetBSD, Plan 9 from Bell Labs, Windows 10 IoT Core, xv6, Haiku, HelenOS, Genode OS Framework and many more. The Raspberry Pi Foundation recommends the use of Raspbian, a Debian-based Linux operating system.

**3.2 Activities**

* ARDUINO UNO:

It is the best board to get started with the coding. Arduino UNO coding is done in Arduino IDE.



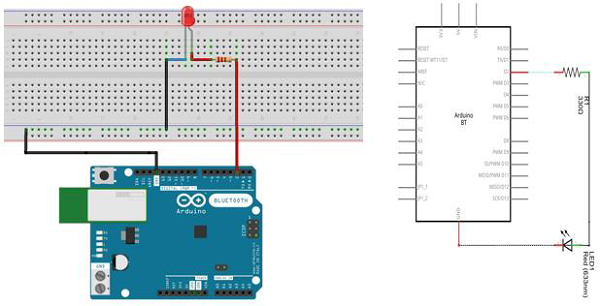
1. Blinking of an LED:

Blinking of led is a simple project where we can see that once the program is loaded into the Arduino memory the LED starts blinking. The speed of Blinking of an led depends upon the delay given in the program. This delay can be altered.

Components:

1. Arduino UNO and cable
2. Jump wires
3. Led
4. Register
5. Breadboard
6. Power source
7. Executable program

Circuit Diagram:



Arduino Code:

// the setup function runs once when you press reset or power the board

void setup () {// initialize digital pin 13 as an output.

pinMode (2, OUTPUT);

}

// the loop function runs over and over again forever

void loop () {

digitalWrite (2, HIGH); // turn the LED on (HIGH is the voltage level)

delay (1000); // wait for a second

digitalWrite (2, LOW); // turn the LED off by making the voltage LOW

delay (1000); // wait for a second

}

Output:

The pin number 2 is set to output. So, once the power source is given, program is loaded and pin is connected to pin number 2 the LED starts blinking.

The delay given here is 1000ms so every second we can see the led blinking.

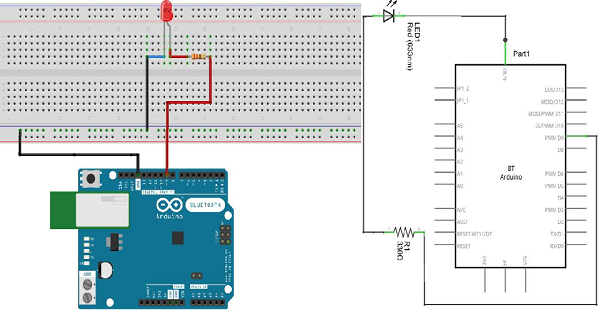
1. Fading of an LED

Blinking of an LED is a digital process whereas the fading is always considered as analogue process since there is a continuous motion. There is a continuous process where first the led’s brightness goes on increasing and slowing goes on fading.

Components:

* Arduino UNO and cable
* Jump wires
* Led
* Register
* Breadboard
* Power source
* Executable program

Circuit Diagram:



Arduino Code:

int led = 9; // the PWM pin the LED is attached to

int brightness = 0; // how bright the LED is

int fadeAmount = 5; // how many points to fade the LED by

// the setup routine runs once when you press reset:

void setup ()

{

// declare pin 9 to be an output:

pinMode (led, OUTPUT);

}

// the loop routine runs over and over again forever:

void loop () {

// set the brightness of pin 9:

analogWrite (led, brightness);

// change the brightness for next time through the loop:

brightness = brightness + fadeAmount;

// reverse the direction of the fading at the ends of the fade:

if (brightness == 0 || brightness == 255)

{

fadeAmount = -fadeAmount;

}

// wait for 30 milliseconds to see the dimming effect

delay (300);

}

Output:

The pin 9 is with a tilt sign in the Arduino board which means it is an analogue pin. So, for this

project the analogue pins are used. The output is that led goes on increasing its brightness and then decreases by certain amount of fading.

1. Password protected Arduino controlled servo gate

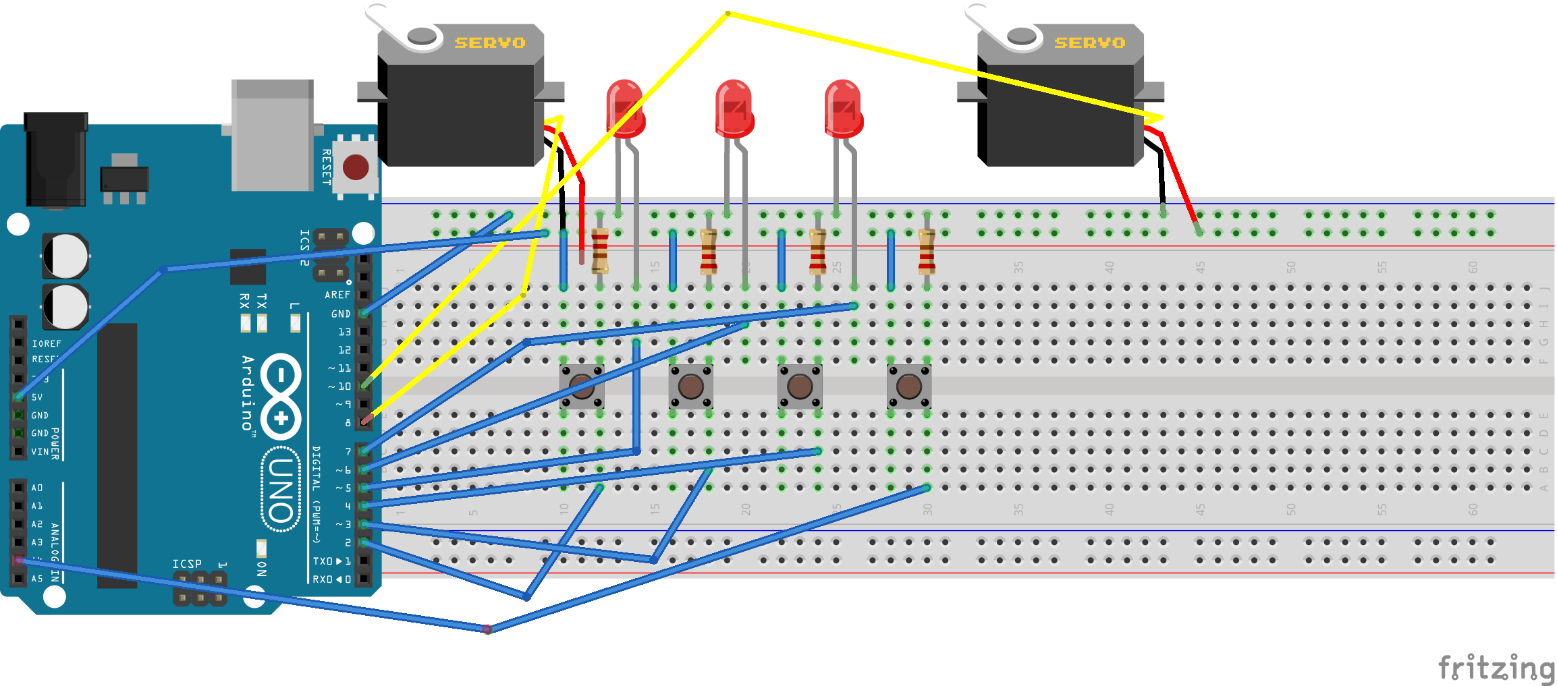
This project is about servo motors which are controlled by a push button. The servo motors only work when a certain algorithm is followed. So, this can be used to simulate the doors which are able to open only when certain password is followed.

Components:

* Arduino UNO and cable
* Push buttons
* Servo Motor
* Breadboard
* Jump wire
* Register
* Power source
* Executable program

.

Circuit Diagram:



Arduino code:

#include<Servo.h>

Servo myservo;

Servo myservo1;

int led1=5;

int led2=6;

int led3=7;

int push1=2;

int push2=3;

int push3=4;

int push4=A4;

int counter1=0;

int counter2=0;

int counter3=0;

int counter4=0;

void setup() {

// put your setup code here, to run once:

pinMode(led1, OUTPUT);

pinMode(led2, OUTPUT);

pinMode(led3, OUTPUT);

pinMode(push1, INPUT);

pinMode(push2, INPUT);

pinMode(push3, INPUT);

pinMode (push4, INPUT);

Serial.Begin(9600);

myservo. attach(9);

myservo1.attach(10);

}

void loop() {

int buttonState1=digitalRead(push1);

int buttonState2=digitalRead(push2);

int buttonState3=digitalRead(push3);

int buttonState4=digitalRead(push4);

if(buttonState1==1){

counter1=1;

}

if(buttonState1==1 && buttonState3==1) {

counter2=2;

}

if (buttonState2==1 && buttonState3==1) {

counter3=3;

}

if(buttonState4==1) {

counter4=4;

}

Serial.println("counter2");

Serial.println(counter2);

delay (50);

Serial.println("counter1");

Serial.println(counter1);

delay(50);

Serial.println("counter3");

Serial.println(counter3);

delay(50);

if(counter1==1 && counter2==2 && counter3==3){

digitalWrite (led1, HIGH);

digitalWrite (led2, HIGH);

digitalWrite (led3, HIGH);

myservo.write(90);

myservo1.write(0);

}

else {

digitalWrite (led1, LOW);

digitalWrite (led2, LOW);

digitalWrite (led3, LOW);

myservo. write (0);

myservo1.write(75);

}

if(buttonState4==1) {

counter1=0;

counter2=0;

counter3=0;

}

Serial.println("buttonState4");

Serial.println(buttonState4);

delay (50)

}

Output:

This project is all about a password-protected gate. The gate is made up of two servo motors and opens when a fixed algorithm is followed.

The project consists of two servo motors, three LEDs, four push buttons (three for gate and one for reset purpose). When the algorithm is followed correctly, the gate opens. Algorithmis like this...

When push button 1 is pressed once keeping the 2 other pushbutton at off state. Afterthis, pressing push button 1 and push button 3 at once and keeping push button 2 as it is. After this, pressing push button 2 and push button 3 at once and keeping push button 1 as it is. When this process is followed, the gate opens and all the LEDs glow. When push button 4 is pressed, all the values of push button get 0 or reset.

**CHAPTER 4**

**Conclusion of The Report**

**4.1 Conclusion**

Thus, I conclude that the ESIOT internship at cubiccode gave me an opportunity to learn all of these things and even more than that. The corporate behavior as well as the importance of IoT is understood very well. The idea presentation conducted at office gave me an opportunity to speak up in front of everyone.

IoT is been introduced so that the man power is reduced and more comfort is given to the customers. By the invent of embedded systems and IoT we are moving more towards automation. Each concept is clear with better examples. Thus, we should understand that IoT has become an inseparable part of our lives.