surveillance robot

Submitted in Partial fulfillment of the Requirments
for the degree of Master
in
the Computer Sciences

May 2016

Helgurd Aziz Mirza

"Thanks to my solid academic training, today I can write hundreds of words on virtually any topic without possessing a shred of information, which is how I got a good job in journalism."

Dave Barry

Abstract

beside researching the Security vulnerability in the future of Robots there is an implementation of Programming in aim to bring the partial of the theory to the live. This project implements a security robot suitable for warehouse or other similar buildings such as a library. The robot follows a pre-defined path and takes images of surroundings which are compared with previous images to detect any changes. The robot is Build on Raspberry Pi and Brick pi fro Lego EV3 Mind-storms equipped with one colour sensors and 3 Motors, standard Raspberry pi camera, software used Python.the aim of this Project is mainly to research many different aspects of Raspberry pi board a and vulnerability in the future of Robotics System build with this little tiny board.

Acknowledgements

My thanks go out to all lecturers at ITB who have held my hand through the development of this project. Specifically, thanks to Arnold Hansman, my supervisor, who showed massive faith in taking on and internally promoting this Thesis and kept me on the right track during the project development.....

Motivation

Security is an important element of business and homes as well. Crime in Ireland is increasing recording to central static office in Ireland[1] and is likely to further increase with recession. Alarm systems although increasing in sophistication, are still inadequate for a total security solution. Need for flexible security other than alarm systems is become more important. Traditional alarm system have a fix position and in many case are easy to hack while mobile security system have the advantage of moving this make it harder to be hacked, In many cases, many individual security systems would be required to fully implement a totally secure system for a building. Robotics fill this need as only one camera needed instead of many camera, finally traditional system required installations and wiring for each component in the system while mobile robot will use wireless communication ,this can save money and installation time that required in traditional system. however the security on the Robotic system itself should be taken in to account.

Aims

In this project going to build a security robot using Lego EV3 Mindstrom on Raspberypi BrickPi with python. However during achieving this project found that one color sensor it is not sufficient to follow line therefore the tools which decided changed to more powerful an cheaper in price.

- Scan the area and keep recording changing to environment to keep tracking and mapping the area , when change happen the system will take one off the following action previously set-up by the administrator.
- Records any changes in environment to the map and notifying the system administrator.
- Recorded change and rise alarm.

Objective

Robot will have the following:

- The robot will equipped with camera to help to recognize any change in environment, has motion capture.
- The robot will follow a path.
- Have remote control functionality so administrator can gave order to move robot to specific position though internet and can take a look by using on robot camera.

Scope

the scope of the robot environment will be warehouse or building have the same layout.

Problem statement

The big company's in the past, store and storage's warehouse, night time security been in the risk always. Preserving some places put human lives at high risk. In order to overcome this kind of dangerous tasks and to spare some workers live, a Maze solving robot has been created.

In order to overcome this hazardous job and to save the life of the Army, a maze solving robot has been designed and developed. This maze the solver is helpful in the situations whenever around is a danger to human life. The robot moves all the way through the places and recognizes and reports the aim and comes back by the optimum path by avoiding dead ends. The robot has a cam mounted to take a shot for any an expected move, and to take an image of the surrounded area, and make a comparison next time visiting if anything changed to raise the alarm. By deploying such a robot will make places much safer, because the robot can't blink the eyes, never get tired, does not need a break. Hence, the Surveillance robot will make the life more secure where humans supposed to be in danger.

Related Work

The area of surveillance robots is really popular. A huge amount of work carried out in navigational algorithms and control system of wireless surveillance robots. A standard theme is also the use of a camera on the robot to be able to receive live video feedback. Wireless robots use of the Arduino micro controller has now been applied, but wireless communication occurs making use of the Zigbee protocol, which limits the product range of the robot. A robot which performs image processing utilizing the camera on an Android smart cell phone has additionally been implemented. However, this technique limited by the processing power of the phone. Our project on raspberry pi will meet what previous micro controller devices have not yet fulfilled the requirement of this surveillance robot.

Proposed Solution

This paper presents followed surveillance robot, however security solution that integrates vision, intelligent algorithm and robot technology has been mentioned . however the system can be more advanced if cooperating with few network camera over a head the map by this will grading large facilities will be more secure, and robot it self can be secured from stolen. Those cameras will help find the robot in case of robot network is down, stuck somewhere, mechanical problems.

Research Methodology

Most of my researchs have been don in schoolor, and mendeley academic papers. been looking inside projects. visited Maplin in blanchardstown, to discuss the clients demand on surveillance robots and the Technology that they providing, and what exactly the most salable devices regarding surveillances.

Contents

List of Figures

Chapter 1

Introduction to the Surveillance Robot's

The Introduction

The Robot movement waves on its way. No longer, robots are only science fiction. Robots are everywhere, on the March, in Hospitals, in the battlefields, yet over the next few years, the robots will dominate every aspect of our life, all over the places. The fantastic thing is that any person with little electronic board called raspberry-pi can turn out to be a robotics. Deploying indoor and outdoor, security in big demands nowadays, where monitoring and observing of some locations such as airports, complex, crowded centres, became impassable by human in contrast to the computerized robot. Capabilities of security methods, advanced from the classic passive role. The classic models could only observe events and create an alert of Hazard, for example, Electric Dog 1912[2], since Greek ancient time[3][4] human tried to create a humanoid- human-like robot in hope to interact with the humans and environments, and their dreams almost came true, just in the last few years. Many robot security platforms along with a remarkable number of algorithms developed particularly people and object detection algorithms.

In General talk, the next generation of the robot research shall be conducted on the multi-functioning robot, to be able to work almost everywhere and under any situations rather than design for each task different robot, the multi-task robot can be used specifically in Hospitals and firework assistance.

The Mobile multi -task systems are still in their early stages of improvement; Many Issues are open for the researchers yet. The robot needs some exclusive sensors to

be able to perform as a multi- function robot, i.e. Recognition, taste, touch, smell, etc... Our aim of this project is a four-wheeled raspberry pi based surveillance robot, which will be monitoring an area of the interest of our needed. As human being, our ability to observe the locations or crowded places is limited. The Aim of this project to assist security guards, and decrease the risk of their life in many instances such as the Battlefields, diffusing bombs, etc..... However, the sensible systems, on the mobile based system are still in their early stages of improvement yet; many Issues are still open to the researcher. To obtain an intelligent robot, should be the robot containing of some exclusive sensors to be able to achieve the desired tasks as recognition, taste, touch, smell, etc.[5].

While upgrading a simple robot design to the complex, there will be a challenge and many drawbacks, however, the developers do not need to invent new algorithms, but they only need to know how to imply what already is there. Nevertheless, the big data and power sources will be restrained the developers from thinking how to attach all censers together to one single robot to perform: fast path tracking, voice recognition, image processing, quick decision making, etc...., alternatively to one multi-task functioning robot, there is an S+T algorithm[6].The goal of the activity can be divided between more than one robot, by using the S+T algorithm. The S+T algorithm will enable the robot to ask each other among them to complete the task in case one of the robots not capable of finish his work by itself. To carry out MRS, we need a stabilized communication system to allow robots to confirm their position and exchanges information between them, and by applying the communication systems on the robot, it will result to increase the efficiency of MRS7.

for more about the Brick pi see page 10 figure 1.4

Literature Review

There are wide ranges of researches and developments taking place using the raspberry pi nowadays;due to the fact its cheap and incredibly small when compared with other computers available on the market know.

1.1 The Technology

In this project, we used python as the libraries, the reasons for the libraries are only available in python. Makes the creating of the user interface easier and the user can view and to control the movement of the robot. However, in this project robot will following the predefined coloured line as the path. That means probably not necessary to design a user interface to drive and navigate the robot. However, still python will be used in programming the robot. The wireless technology used to transfer date from raspberry pi and user and vice versa.

1.2 Problem Definitions

The Primary goal of this thesis is to research and develop a surveillance indoor robot, and This is done by investigating different aspects in terms of the robotic system. There are endless terms and definitions to be researched. However, this thesis will only highlight a few important subjects.

1. what does Autonomous Robot mean?

   A robot that will take steps to decide how to choice the algorithms that comply with the surrounded environment without human discussion and interaction. However, there is a long way to deciding that the robot autonomous reached to our desired point, yet robot needed to be developed to update and maintenance by itself without human interaction.

2. Up to what level will the robot should recognize objects and it is properties to be able to live in the human environments and avoid causing problems?

   a robot need Huge database to recognize the object's properties, hence we need a faster processor and bigger hard drives for a robot to respond at the time. Otherwise robots could cause threat on human's life.
   Need countless of sensors. Which makes it difficult even to think of designing A robot. There is a different ways to approach this problem. However, the security to

handle the data by The robot is remaining an issue, i.e. If data processed remotely from a website that dedicated to one or more robots, the data will be exposed in case if the robot is stolen.

3. which type of environment the platform needs to be designed for?

   There have to be boundaries of some type or kind of the working platform.

4. Test design that is complete meets certain requirements? It will meet if the robot platform is ideal for its purpose. And should comply with a set Requirement characteristics. Hence, the Geometric has to be analysed before stepping and designing the robot.[8] However, endless problems exist in terms of designing, recognition, planning, etc... For further reading, I recommend " Springer Handbook of Robotics ".[8]

1.3 What is Surveillance Robot's?

A lot of work carried out in navigational algorithms, and wireless surveillance robot's control system. The subject of surveillance robots is fairly popular, the aim of building a surveillance robot so it hopes to help security and law enforcement. However, this project will only be used indoors. It will function while a motion detector detects movement, then the camera will start the recording or take snapshots of the area, and by the second pass will take the a picture again and comparing to the first one if anything changed will try to raise the alarm. Everything should done remotely. To have more clear idea on this project please see figure 1.1

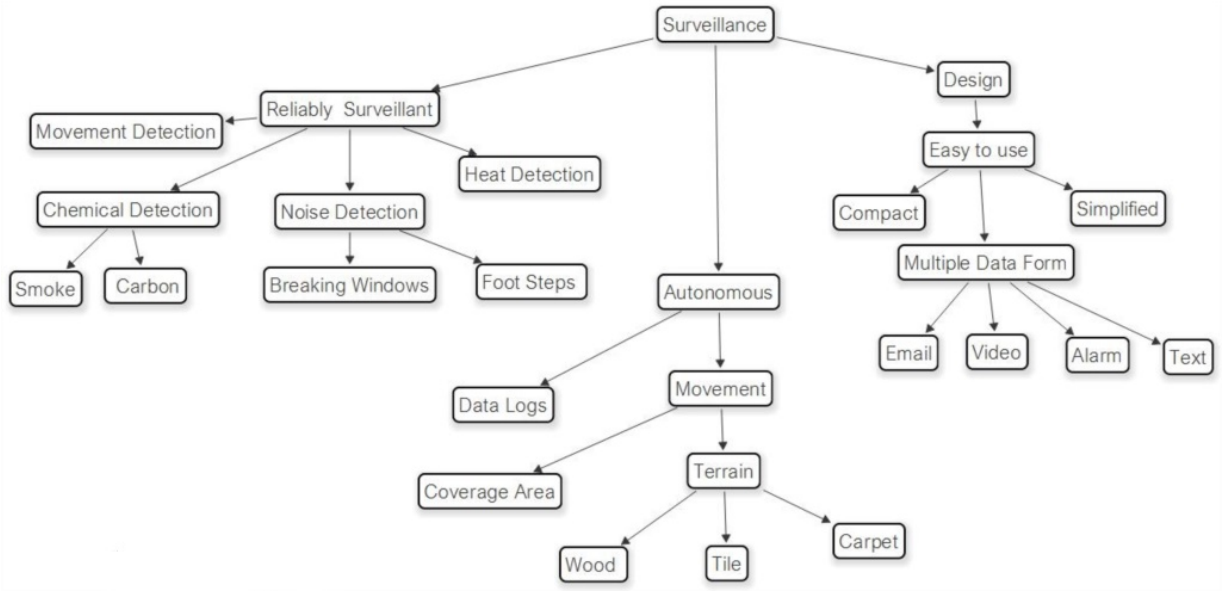


FIGURE 1.1: Pre-Design Basic Robot Digram

1.4 The Hardware

It is the most important to decide what hardware to choose for this type of project, thus we have to do wisely research to find out different models and marks,because once all parts and tools have been combined, and data stored it will be very difficult to change the project. Therefore, we have decided to study all other devices function as that available and can be used in our project.

1.5 Raspberry Pi

The low-cost Computer as small as a credit card, which could be plugged into TV, or Computer Monitors, with use standard keyboard and mouse. This little device has the capability to be used by people no matter what age are they, it comes with programming software pre-installed called Scratch, out of the packet which allow Even Kids to go and give a programming try. Yet! Many different Advanced programming ID could be installed on the system. It's doing anything that you could expect from computer to do. It can be connected to The internet, Ethernet or wireless. It's capable of playing games, to writing and making spreadsheets, or playing high-definition video.

The raspberry pi is a great product and its set-up from it's ancestors. however, there is a huge different between latest model and older one in terms of functionality and handling threats. the tree of its predecessors are as follows:

- Raspberry Pi B 26 pin GPIO headers
- Raspberry Pi A 26 pin GPIO headers
- Raspberry Pi B+ 40 pin GPIO headers and B+ has 4 USB connectors.
- Raspberry Pi A+ 40 pin GPIO headers
- Raspberry Pi 2 Model: 900MHz quad-core - 40 pin GPIO headers and 4 USB connectors etc..

in February 2015, the latest generation raspberry pi2 Model B released and is one of the best (simple single small) board computers available at the moment yet added awesome reputation to the Raspberry pi family[1] either used in fields like Educational, commercial or hobbyist.[9]

[1] 500,000 pi2 Model B sold an almost 5 million from all family together sold around the World

The primary differences between the B+ and the Pi 2 is its quad core CPU and All the I/O ports on the Pi 2 are nearly identical to the B+ but Raspberry 2 main features are as follow :

- A 900MHz quad-core ARM Cortex-A7 CPU 1GB RAM.
- 4 USB ports.
- 40 GPIO pins.
- Full HDMI port.
- Ethernet port.
- Combined 3.5mm audio jack and composite video.
- Camera interface (CSI).
- Display interface (DSI).
- Micro SD card slot.
- VideoCore IV 3D graphics core.
- Because it has an ARMv7 processor, it can run the full range of ARM GNU/Linux distributions, including Snappy Ubuntu Core, as well as Microsoft Windows 10.

The Raspberry Pi 2 has an identical form factor to the previous (Pi 1) Model B+ and has complete compatibility with Raspberry Pi 1.

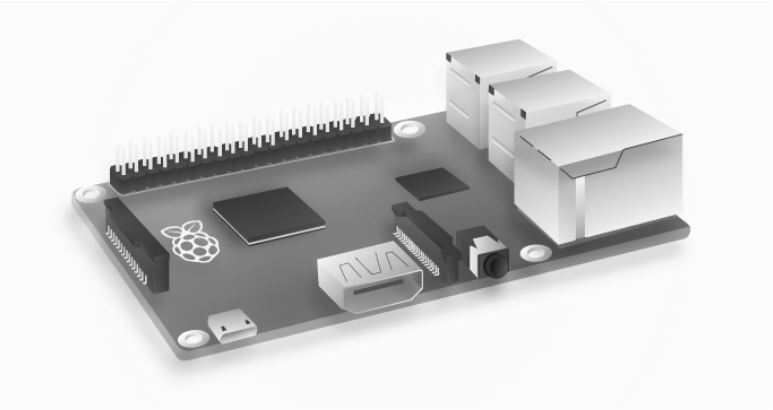


FIGURE 1.2: Raspberrypi2 Board

1.6 Downside of the Raspberry pi

What is not good in raspberry pi despite every possibility of the raspberry pi, still will be much more to be considered about before trying to use this pocket-sized PC. For e.g. How it is easy to get development tools and how often can get technical supports for software and hardware, in case you stuck somewhere while you are in the middle of the development. Usually developing software will take greater parts in building a Robotics system, so it is will be wiser if investigating software elements and attributes in wider ranges. As Raspberry pi2 Model B is ARMv7 Multi-core processor, full Gigabyte of RAM, all improvements from it are an ancestor move it from bringing a toy to a real computer. However, just because it is ARM processor and not regular PC CPU, so it will limit the developers by architecture and power abilities due to some portions of Linux code have not been completely ported to the ARM Processors to prove what being here claimed about it is the downfall is a laser scanner at by the time of writing of this thesis, there is no prove to utilize to do any 3D mapping. Raspberry pi will be reset in the presence of a bright light when it comes to the advertisement, we always see that raspberry pi 2 compatible to install Windows 10, but it is likely to an ARM Version not what's being built for PC. RPI will let a developer determine which input-output (GPIO) pins to choice of sensors and motors, not to confuse the readers of this project, despite saying raspberry pi, is ARM processor yet it is like a computer; you can install Linux on it decide what size of memory you want and install software like OpenCV on it. Decide your own sensors and motors, those have restricted when you decide to use EV3/NXT. Raspberry pi 2 Model B is a great potential for small computation.[10]

1.7 EV3 Lego Mindstorm

LEGO inc. Released series programmable robotics Kits in 1998 called Mindstorms, that has since been used in graduate level and researching the area in the college's particular in Computer Science Schools The third generation of Mindstorms released in 2013 called EV3 with 300MHz ARM9 CPU, with 64MB RAM. The environment in the development of EV3 IDE is Linux based firmware, that is using data flow graphical consisted of icons performed by drag and drop. In line. The main disadvantage for the real developer the IDE is conducted on a virtual machine. The virtual machine is included in the Linux-dependent firmware running on EV3. The graphical drag and drop made the environment very friendly to Student or anyone wants to start learning to program the Robots. Working on EV3 don not required to identify parts, for example,'s Motors and sensors, eventually, EV3 will teach learners the logic and planning and how to work with the real-world programming system, and it is the threshold of translating the knowledge which gained in programming EV3 into real software projects. It should bear in mind; there is some downside for the programmers who are currently comfortable with common language development environments for e.g. C or C++, they will be challenging to write complicated applications with the graphical programming language. They find the lack of real-time multi-processor support and Virtual machine creating awful real-time performance, the software exclusively made for EV3, which causing hard to extend. The most critical for me to not a choice and decide not to go with EV3 despite my lack of programming experience, the matter of fact EV3 Lego only has one process. The kernel constructed to switch context's threads every 10ms that mean even if your thread has given priority still need 10ms until the other thread released the CPU.[11]

Some Developers have done the EV3 reverse - engineered to builds models that completely different from what was origin software platform came with. The system, called ev3dev which will bring the ability of the programming EV3 to the another level, provides the developer the possibility to program the EV3 the way they do want. They aimed to create a low-level driver framework, to control quite everything connected e.g. Motors and sensors etc. EV3ev is based on Linux operating system, which operates on varied LEGO including the Rainstorms EV3. Eventually, ev3dev is compatible with many popular scripting languages, that will give developer right away with their favorite language and libraries. The team who has reverse - engineered still in learning stage and everything at the time of written this Thesis, they published should always be regarded as experimental and it is subject to major change, for this reason, EV3 not thought on this project.[12]

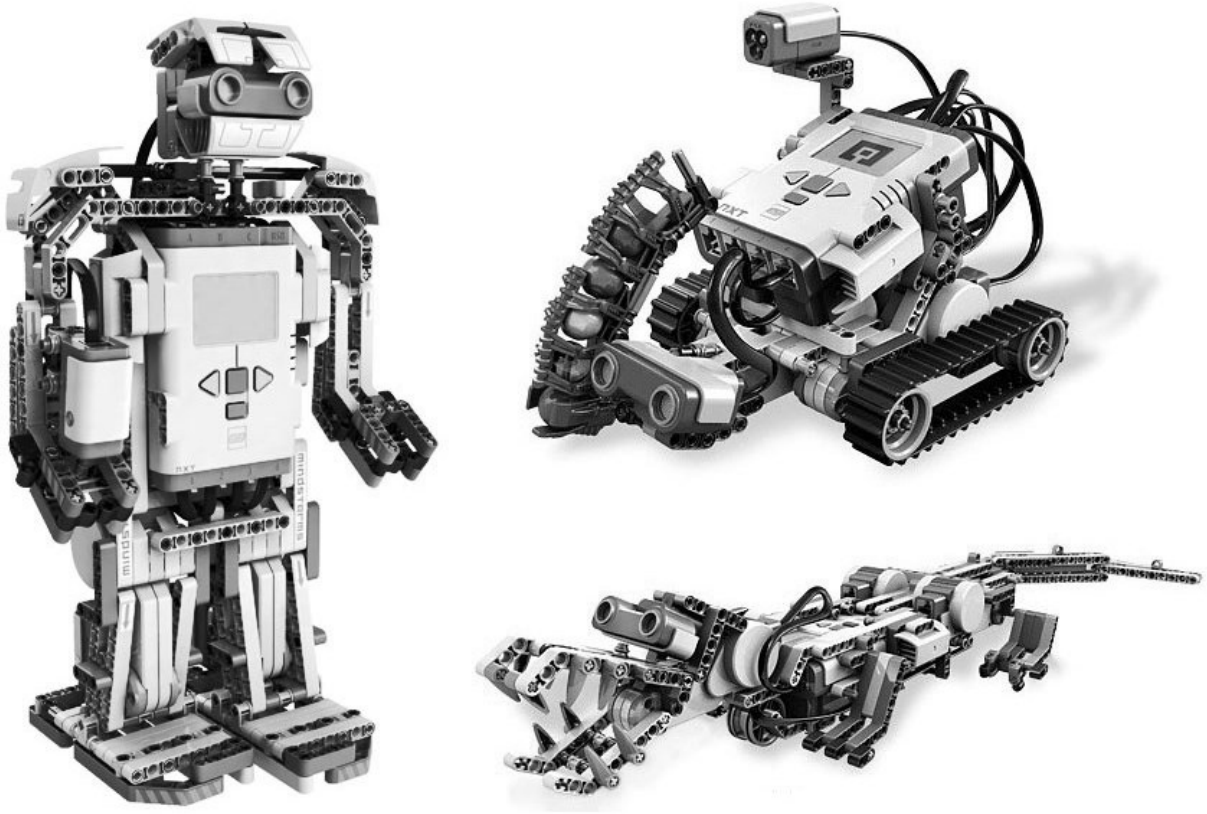


FIGURE 1.3: EV3 Lego used to build a robot

1.8 BrickPi

Brick Pi is a board that is slide-on from Dexter Industries that converts the Raspberry Pi into a robot. The BrickPi helps to connect LEGO mindstorm's sensors, motors, and parts to the raspberry pi; it will effortlessly switch the credit card size raspberry pi board computer into a robot and empower it. This thesis will help to set up all the software and other dependencies for the BrickPi along with python codes to configure and set the Brick pi to respond to the raspberry pi GPIO. Equipment you will need:

- Raspberry Pi (with an SD Card and WiFi dongle or Ethernet).
- BrickPi Board.
- Battery Pack.[13]

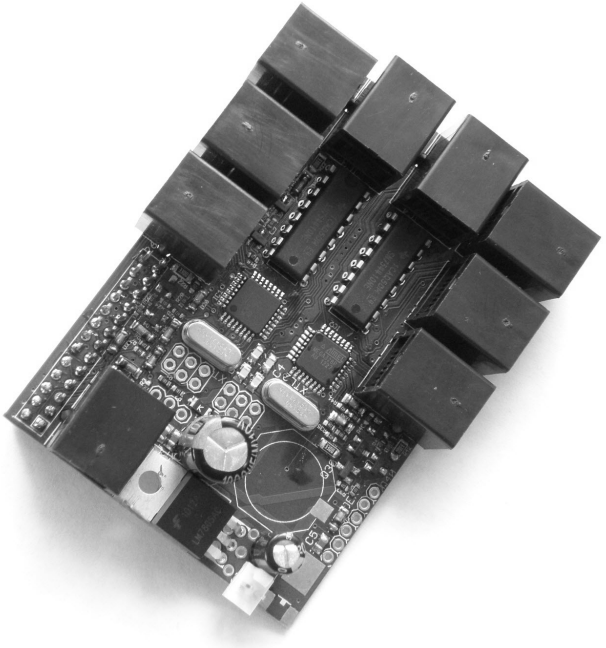


FIGURE 1.4: Brickpi

BrickPi – Control Your LEGO Mindstorms sensors and motors with Raspberry Pi.

Thanks to the myriad of embedded development board available, nowadays it is easier than ever to build things in your own household. From extremely simple connected switches or relays for home automation purposes, to more advanced, full-fledged robots, the only sky is the limit, and since additive manufacturing has become mainstream and accessible, you could do 3D print virtually any physical part you need for your project. Nevertheless, what if you could combine the unique way of building with LEGO bricks with the power and versatility offered by the Raspberry Pi embedded board.

[14]

FIGURE 1.5: mindstorm parts combinde with raspberrypi brickpi

Brick pi Created by American company Dexter Industries, BrickPi is a two-part system that allows you to bring these two amazing development platforms together. It allows you to connect LEGO Mindstorm's motors and sensors to the Raspberry Pi, as well as mount the whole assembly to your project with LEGO Technic bricks and parts.

The brick pi has the capability to link five NXT sensors or servo motors, with raspberry pi.

Via standard Mindstorms RJ12-style sockets. Two-way communication is enabled, allowing for motor encoder data to be read. Both analog and digital I2C sensors are supported on four ports, and only digital sensors on the fifth. There are no limitations, compared to the NXT intelligent brick, which means the unit can be replaced entirely with no limitations. The BrickPi circuit board itself is the same size as the Raspberry Pi Model B or B+ units and attaches to it via the standard 26-pin header. It is built around two ATmega328 microcontroller chips running on Arduino firmware, which shares a serial channel to communicate with the raspberry pi. Brick pi also supplies power to all hardware attached to it. It supplies 5VDC through a voltage regulator to all the logical components — the Raspberry Pi, MCU, NXT sensors and other hardware, while motors are powered directly from the 9-12VDC power source.

The kit also includes a sturdy acrylic enclosure which can hold the BrickPi board together with a Raspberry Pi and features standard sized mounting holes for receiving LEGO bricks. Programming the BrickPi can be done in Python, C, visually in Scratch, with Wyliodrin, and much more supported by Raspi. Pretty complex function libraries are readily available for download from the BrickPi repository as well as example programs and other information to get you started. Installing BrickPi to the Raspberry Pi can be done in two ways, either by downloading a modified Raspbian image or by modifying your own. To avoid the hassle, you can also buy an SD card with a modified Raspbian image from Dexter Industries.[14] for how to install and set-up Brick pi module refer to chapter 3 section 3.3 Install and Run the Brick Pi

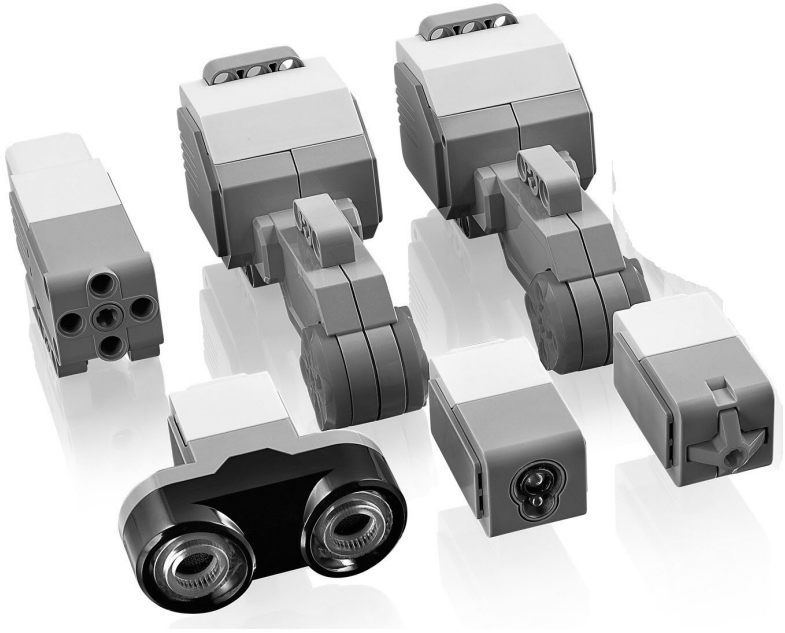


FIGURE 1.6: Mindstorm Motors and Sensors

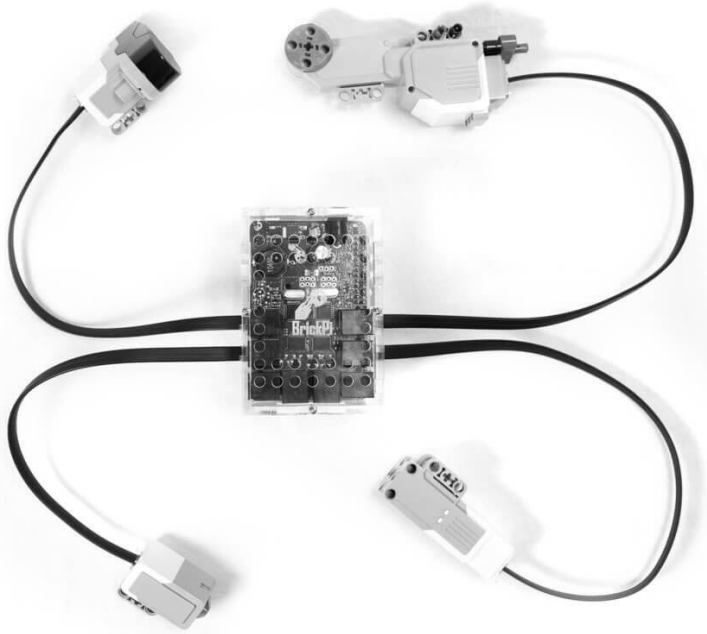


FIGURE 1.7: Mindstorm Parts connected Brickpi

1.9 The Future of Raspberry-pi and NXT or EV3.

In terms of the future, despite there is no solid declaration from the official founding organization of raspberry pie about how the future will be for the Raspberry pi and We are not in the position to predict what the future of raspberry pi will look like either. Logic tells us that, because the technology in the Progress, particularly in the trends toward of the nanotechnology.[2]. In the very near future medicine scientists along with computer scientists sending robots into the Human's body, which have been built with a Raspberry pi or NXT systems, embedded software modified and developed for that purpose. At this, moment imagined tens of thousands of tiny microscopic robots can swallow exactly with the teaspoon that sent to the most delicate affected area within the body precisely. Let we call it a "Nano - Ambulance,"Nano- Clinic". In the very soon future, the science fiction writers are told that they have to look for something more advanced than Nanotechnology and Humanoid robot to write about. " You put the mechanical surgeon inside the blood vessel, and it goes through the heart and 'looks' around," Feynman "[3] said," It finds out which valve is the faulty one and takes a little knife and slices it out " [15]

FIGURE 1.8: Three generations of Programmable Bricks: RCX (left), NXT (middle), EV3 (right)

[2]science whose goal is to create a computer chip and devices that are thousands of times smaller than current technologies permitted.

[3]The first to suggest that you could one day swallow the surgeon was beloved physicist and Nobel Prize winner Richard Feynman. He coined the idea in the provocative 1959 talk " There's plenty of room at the bottom "

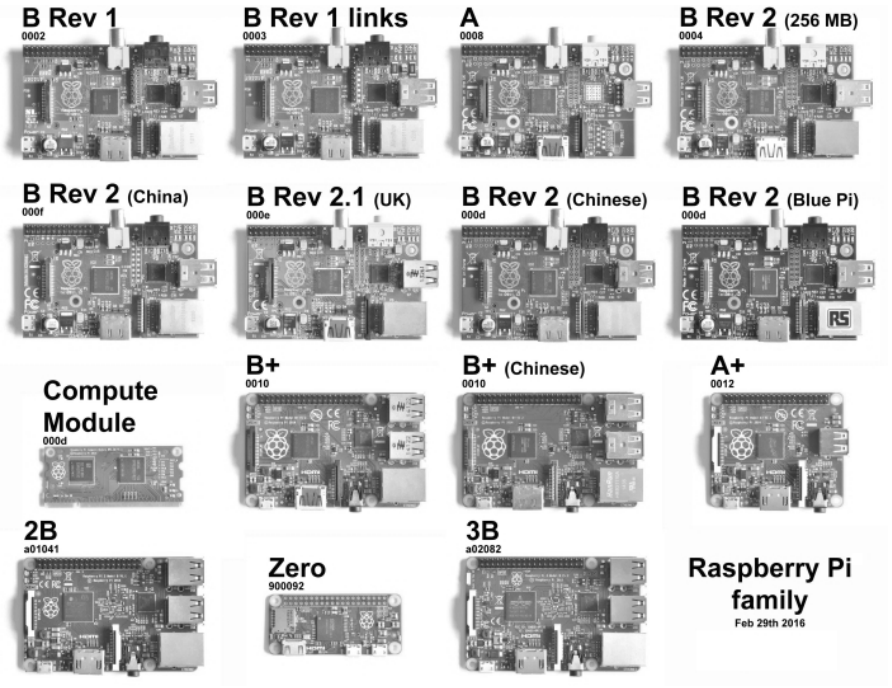


FIGURE 1.9: Raspberry Pi Family

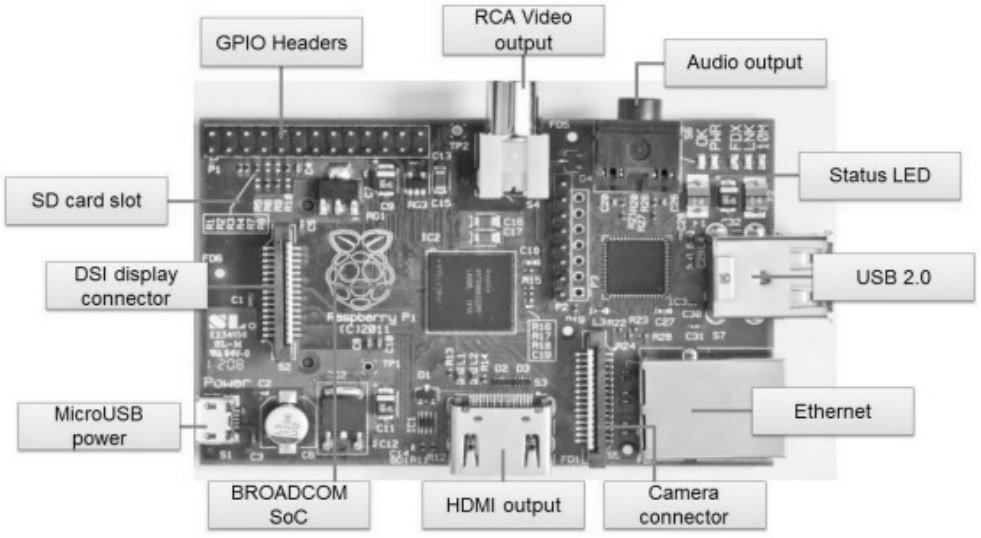


FIGURE 1.10: Detaild Raspberry pi

1.10 Arduino

Arduino the controller that is most and well-known nowadays. Arduino is a microcontroller it has been sized as small as raspberry pi's size, it comes with USB plugs, in order to connect to the personal computer and furthermore, which can be wired to as much as external electronic devices you would like e.g. Laser-diodes, motors, sensors, etc..... same as all other different computer pockets size platforms you will find diverse types and models of Arduino in the market; it has 14 digital input and output pins. However, it has no Ethernet port embedded, but it can be attached to an external after purchasing it as an extra part of course, As it is a microcontroller, it does not need any interpreter, In origin it does not designed to work with python as a raspberry pi does, it is an open source, after some modification, it can support, the python programming language. Arduino IDE based on a programming language which also supports the programming language C and C++. As mentioned Arduino is an open source it is, it is launched in 2005. To send or retrieve data from Arduino it has to be connected with the computer. However, Arduino can be used as a stand alone. There is certainly a sizable community of construction, programming, Electronics happy to share their skills and knowledge on the Internet.

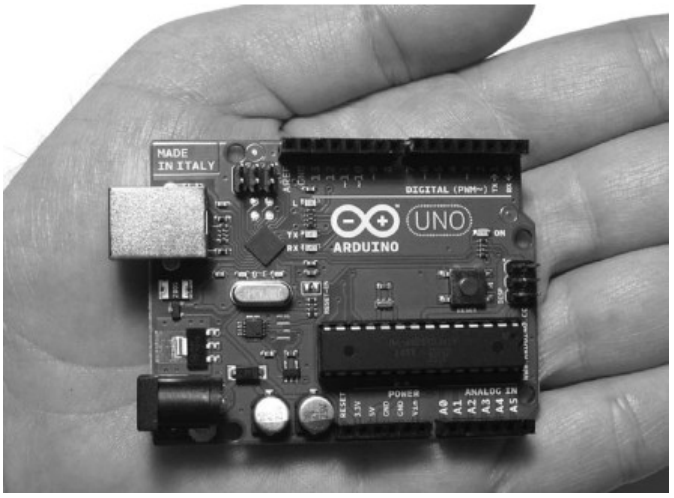


FIGURE 1.11: Arduino

1.11 Banan pi

Shenzhen LaMaker Technology. Co, Ltd produced a single-board computer called Banan pi. This board almost similar to raspberry pi in design and shape, all software's that created for Raspberry pi can run on the Banana pi with a slight or no customization at all. It has an ability to connect a very big SATA hard drive, with A20 CPU at 1GHz. yet, in term of processor, if compared to Raspberry pi new model 2B it is in the half way in comparison to raspberry pi, because has got a quad-core 3.6 gigahertz a processing power. Banana PI's supporting community is very small. However, it keeps growing.

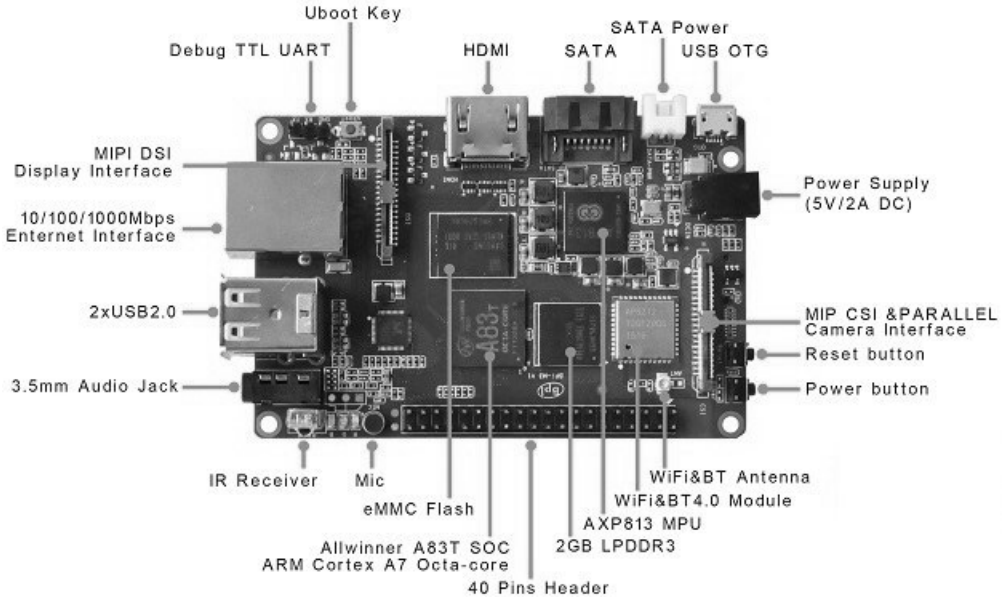


FIGURE 1.12: Banan PI

1.12 The Hardware Evaluation

Following investigation all simple and small-sized boards they all, a tiny computer and all looks very much the same for Starter user, However, they are different, in fact, they are very different devices. So what is the difference, and what is made raspberry pi preferable to be chosen for this project? Let us point out quickly some features of Three of them as an answer.

1. Arduino:
   Arduino: it is a microcontroller; Arduino can be programmed in C, but cannot run an operating system by itself. It cannot be connected to the Internet despite has a big community support, and large of component's compatibility yet, due to the absence of the interface cannot be taken into consideration in this project.[16]

2. Raspberry PI:
   it is efficient software and hardware. Due to easy OS installation and fast performances, raspberry pi has produced huge ready projects online, which mean a vast community to make contact with and get support anytime.

   You get nearly anything you think about, just like the add-on and compatible extra devices with raspberry pi for your project.

   It can proceed to any project, especially it is awesome for college students as a starter kit on another hand it is very cheap to buy.

   The only thing close to this project could be considered as Cons: it lacks a good Android OS. [17]

3. Banana Pi:
   Banana pi delivers excellent CPU and Memory and the longest list of features. However, it lacks in the high quality and reliability and consistency. It is too high priced compared to the other two boards. It has a very modest community, not that many accessories available as raspberry pi does.

   confusing new starter by leading them to a few different websites for same purpose, OS installation is not so easy hence sometimes push the starters to give up the projects or switch to different platform, despite having few certain features like on-board Wi-fi, IR sensor, Fast Ethernet, Android.[18]

4. EV3/NXT

   The matter that is only EV3 make it advanced over the pi is embedded in with LCD and does not need Brick pi to mountain motors and sensors unless you need to build the most complex project that could need more sensors to run. Bear in

mind as mentioned before, the raspberry pi a more faster and lighter. The matter of fact, raspberry pi has got more USB slots, also has HDMI, Ethernet too. there will be a fun and stunning robot if raspberry pi and EV3 have combined into one project Or the sensors via Brick pi, besides fun yet you will get the stunning robot.[9] [12]

following the extensive researching, in this thesis found that Raspberry PI in term of educational will be the right choice, it will stark and almost can build everything on it, in other hand it is cheaper and easy to learn, unless, you decided to go more complicated projects, there are few others have been studied for the sake of this project. Nevertheless, the are beyond the reach of this project. As a result, they are up to the level of the Knowledge of the students, and the type of the projects they would like to under take e.g.. BeagleBone, PCDuino, Pine A64....

1.13 Similar Projects & Techniques

The same sort of project has been implemented already in past time either as a commercial or as hobby (DIY) projects, each of them with a different selection of features. In the Following section, it shall be offered an overview that is a brief of DIY task and a commercial product currently available on the market.

1. PiLarm:
   Portable Raspberry Pi Room Alarm PiLarm (Jeff Highsmith, 2013) is an excellent motivation for this task. PiLarm is an easy house security alert using a matrix keypad, a speaker, a camera, a motion sensor and LEDs. Highsmith's system in motion spotting is forwarding notes on Tweeter and an email, to inform the related person about suspected entrance. Moreover, It enables some regular voice guidance through the embedded speakers, to notify the owner all about the system history or raise the alarm to inform the people. The project has straightforward python functions, and the framework of the execution is even simpler to understand. [19]

2. Arduino Anti-Theft Alarm Shield:
   Arduino Anti-Theft Alarm Shield is a total anti-theft alarm suitable with any type of Sensors, it can recognize the RFID, web server, depending on an Arduino Mega (IngGaro, 2014). The method can be managed and tweaked from the integrated web application, And it is the own control panel, providing LCD, capacitive control keys, and RFID Recognition. Moreover, the system sends email and SMS notification once it is Alarmed, and a backup power supply is around in case of origin power fault occur. [20]

3. Extra researches:

   Commercial Home Security Systems (in Maplin) An additional source of motivation, is the significant number of commercial security systems. Commercial items might enable a lot to understand and Initiate thinking regarding what features a real-world Home Security System needs and the ways the that user should have interaction within the system daily. The tips that have been accumulated from the commercial goods had not been from just a Pair of systems, however, a combination of many products from various companies on-line, Which made a huge impact on the feature's requirements. The research in the commercial section aimed to see what clients are exactly looking for. What are the best security system and their current optimum sale. I have been in touch with "Maplin Blanchardstwon," they just introduced me a CCTV that could be integrated with Smart-phones with email notification to raise the alarm and IP surveillance camera that offers live streaming video and audio. [21]

1.14 Hardware Investigation summery

None of the investigated projects could meet clients' requirements, and the outcome was like this: the commercial products are more surveillance camera than a home security techniques. It provides several excellent services like live streaming and email notification, but certain crucial features not provided as an alarm sound and expandability. Keep in mind if you decided to build a perfect surveillance system, it should be considered the investment that One company able to put in the projects and the required technology.

Chapter 2

The Designing

2.1 Overview Of Component diagram

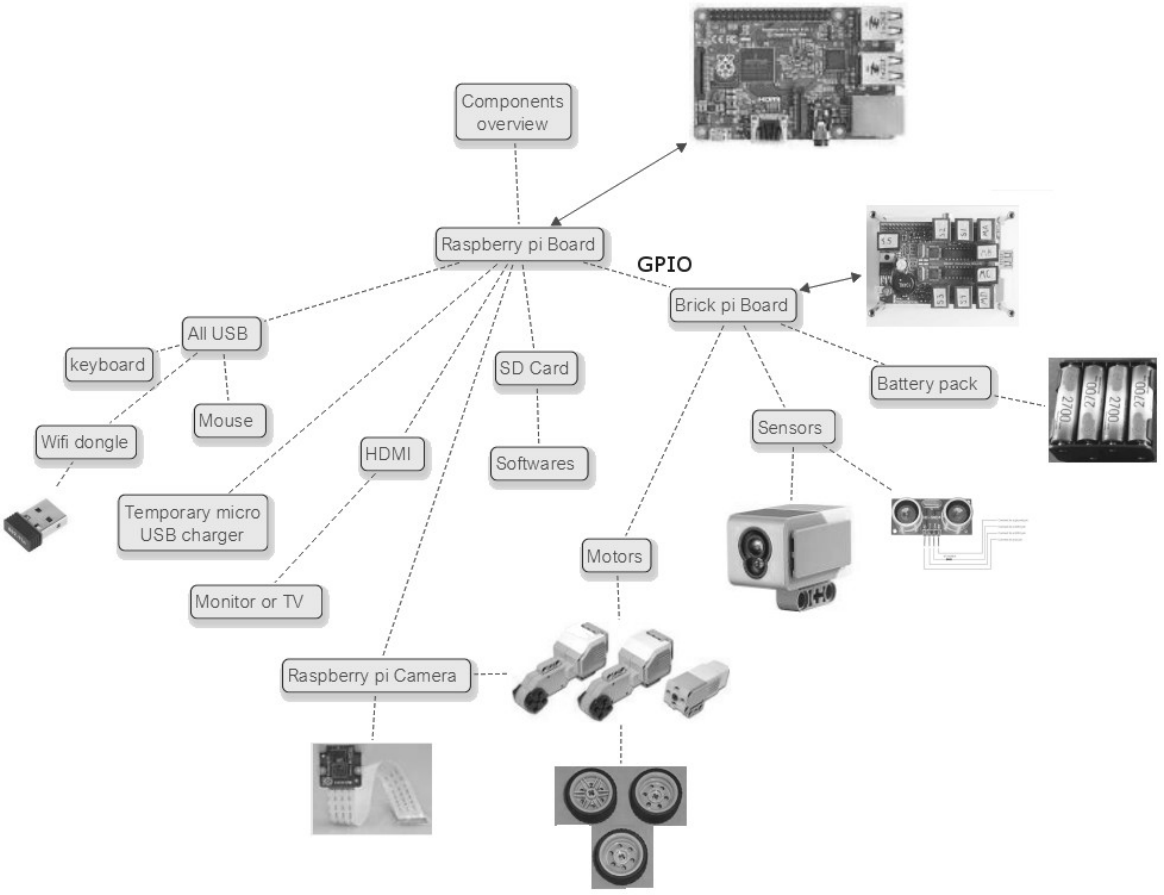


FIGURE 2.1: Component Diagram

2.2 The Robot types

In fact, there are lots of ways how you might be likely to define different types of robots. The robots which look like a human are the most complicated one to make; however, the majority of people are imagining that the robots must sound like a person if they have been asked. In reality, anything performs a task with or without human instructions are called a robot. Nevertheless, a robot has to be designed to perform all tasks given with qualities and exact time expected. It totally depends upon the tasks are given, whether the robot is small or big, fixed at the position or can move. Each work means qualities regarding functions that robot has programmed for to be achieved. Hence, to divide robots into types is the best to classify them by their application and the method they move or behaviors, to understand better what we mean by sorting and classifying robots we[22] should ask ourselves at least three Questions:

- What it does?
- How it does it?
- When it does?

Basically any robot should consisted at least one of these components:

- Sensors
- Effectors
- Actuators
- Controllers
- effector's Arms, Artificial Intelligence, other components that help it achieve mobility.

Divisions made as follow:

- Industrial or Heavy-duty robots.
- Home or household robots.
- The medical robot's
- Service robots
- Military robots

- Entertainment robots
- Space robots
- Hobby and competition robots
- Polar robots

If we would like to talk about a particular robot, type the application alone does not offer sufficient details. We need to classify them too, for examples " Automated Guided Vehicle " thought over as an industrial robot while we consider the stationary Arm that performing some certain tasks as industrial robots as well, therefore, to identify them exactly for the reader to understand it is a good practice to classify them as follows:

1. **Stationary robots (including robotic arms with a global axis of movement)**
    1.1. Cartesians/Gantry robots
    1.2. Cylindrical robots
    1.3. Spherical robots
    1.4. SCARA robots
    1.5. Articulated robots (robotic arms)
    1.6. Parallel robots
2. **Wheeled robots**
    2.1. Single wheel (ball) robots,
    2.2. Two-wheel robots
    2.3. Three and more wheel robots
3. **Legged robots**
    3.1. Bipedal robots (humanoid robots)
    3.2. Tripodal robots
    3.3. quadrupedal robots
    3.4. Hexapod robots
4. **Others**
    4.1. Swimming robots
    4.2. Flying robots
    4.3. Mobile spherical robots (robotic balls)
    4.4. Swarm robots

While some types of robots have mentioned, it is a good practice to talk about some other types; to make this subject General, and give the readers a total overview of what types of the robot are available and yet, there are many more we are not knocking at them, bear in mind, one year after this paper probably readers some new invented robot type and labeled different name to what we have them here. So besides physical robots which consist of hardware, and it is tangible and visible; We have a virtual robot that it does not exist in real life. But it is a combination of some blocks of programs, as they had the better-named software, where they are residing inside our computers, for example, " world wide web search engine." If you compare my work " wheeled Sentry's robot " to what's on hand nowadays, it should be considered as a basic design despite all the functions and sensors has been used to develop it. Complicated and modern robot will need bigger investment and available lab and more time to research. However, wheeled robots are the most preferred mobile robot, specifically for beginners, usually need the minimum investment. The complex type which includes autonomous humanoid needed a lot of freedom, numerous of motors, and sensors[23].

This Thesis is about a robot which it is categorized as "wheeled sentry robot," therefore, I will contrast some of the advantages and disadvantages of the wheeled robot as follows.

1. **Advantages**
   (a) Usually low-cost
   (b) Simple design and construction
   (c) Fast moving
   (d) Six wheels can replace a track system
   (e) Diameter, width, material, weight, tread, etc.
   (f) Easy to program
   (g) not require many motors or parts
   (h) Weight is balanced
2. **Disadvantages**
   (a) May lose traction (slip)
   (b) Small contact area (small rectangle or line)
   (c) cannot be used on the rough or rocky ground.

2.3 Designing The Robot

Developing robots suitable for safely and correctly interacting with humans in real-world settings is one of the greatest challenges in robotics. In fact, there is no ideal design, except for that if we decided to adapt some extra functions, along with required tools for some specific purposes. There could be numerous techniques and methods available, all depends on strategies, similar to designing cars by different makers, as almost all inner of the car have the same components, however, the cars appear different. We should admit that designing the robot somehow probably differing from designing cars, as we all know that, first vehicles goal is to drive, the other extra design and components all are most likely luxuries. While developing robots, there are different purpose for each robots in their designing, and the purposes are all depend on the environments; they are created for to act and adapt with a different situations beside walk and drive around.

It is the matter of the fact, the most consideration that is clear in the design associated with the robot is the characteristics of the workplace area. The designer has to investigate the workspace traits before planning how the robot should have designed, and the designer has to identify the workplace characteristics, these are fundamental problems in the mechanism designing[24] . We will start with the most important components that could be used indoor surveillance robot. The design based on flatted ground.

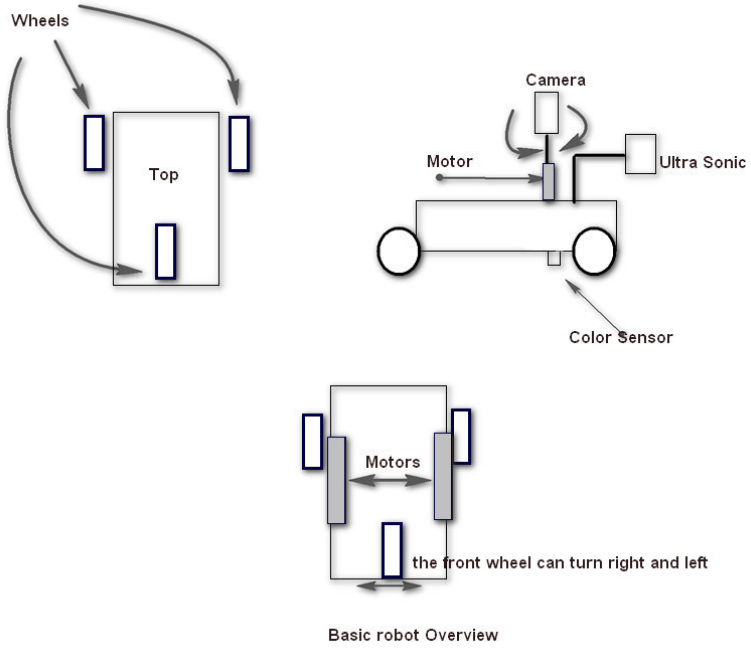


FIGURE 2.2: robot Designing Graphic

The design of surveillance robot provided with a camera and some sensors. The robot is a mobile in all directions, with an arm to hold the camera attached to the motor to

rotate it horizontally in 360 Degrees to give the robot the ability to track all the area that has surrounded, Because the development of the Robot has to be constructed with the Lego Mindstorms NXT most achieve with limited parts, and motors. This Robot recruitment is basically for security application: It has to accomplish the following tasks using Raspberry Pi and NXT sensors via brick pi as a bridge.

- The robot has ability to rotate in 360 Degrees horizontally.
- The robot must be able to go forward, backward, right and left at all times.
- The robot must able to take a photo of surrounded area and on the second visit has to take another picture and compare it. During the undertaking, the robot has to find any unexpected motion with motion sensor and send alerts to the user. Should be-be able to save data and transmit it to the server.

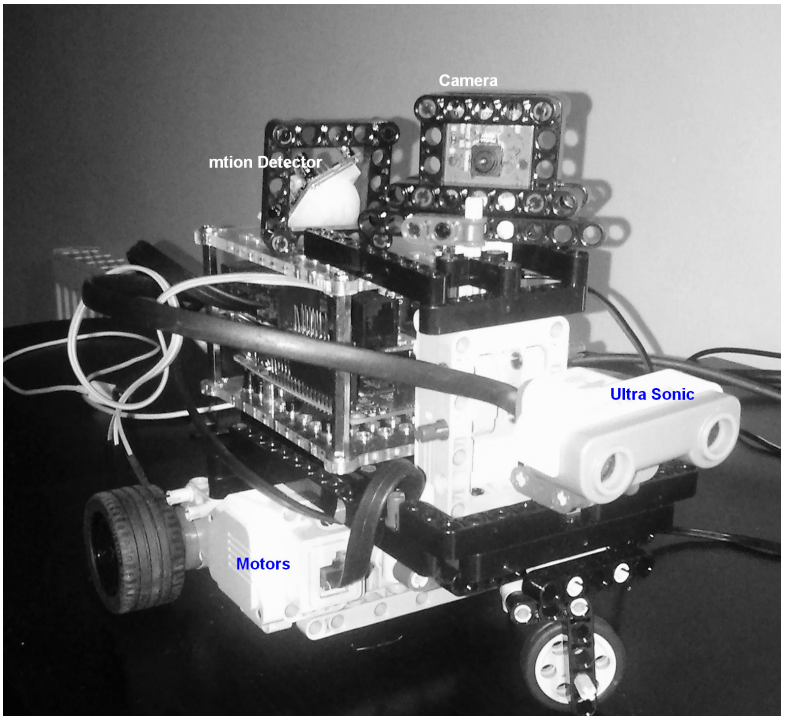


FIGURE 2.3: Origin Design

The parts have been used to build this robot in Figure 2.3 are as following :

1. Raspberry pi board B2.
2. Brick pi.
3. Three servo-motors.
4. ultra sonic.
5. motion detector.
6. raspberry pi camera[1]
7. colour detector sensor.
8. Mindstorm Bricks.
9. Wifi dongle[2].

2.4 The Sensors

The most important component in the robotic system are sensors, in fact, what makes the robot is a robot, it is the combination of the set of the sensors. They would be very useful technologies if they collaborated with each other. You can see many of sensors nowadays for different purposes in this section only related sensors described. Bear in mind the sensors are used in this project very basic, and they usually employed in the educational domains, however, for the more complicated robot system, there are many different advanced categories.

The LEGO NXT and EV3 sets use connectors are comparable to the regular telephone plugs. Word of advice! If phone plugs used with LEGO's sensor instead of Origin, it could damage the entire Brick Pi and the sensor element itself! The fact is that this

[1]module has a five megapixel fixed-focus camera that supports 1080p30, 720p60 and VGA90 video modes, as well as stills capture. It attaches via ribbon cable to the CSI port on the Raspberry Pi

[2]-BCM43143 chipset -802.11b/g/n -150Mbps maximum throughput

connector only comes with a LEGO sensor set and not available anywhere else to buy. There is another model called DEC connector it is usable in the Electronic Hobby shops, but need some work to justify the latch as it is on the wrong sides. The NXT connectors using 6-position their futures analogue and digital interfaces. every little information about connectors can be found on this link in the reference[25].

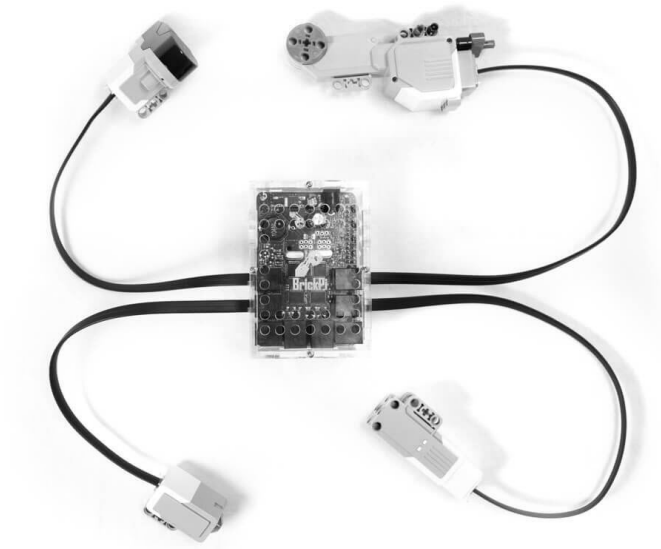


FIGURE 2.4: The Sensors connected to Brick Pi

NXT Sensor Interface Pinout

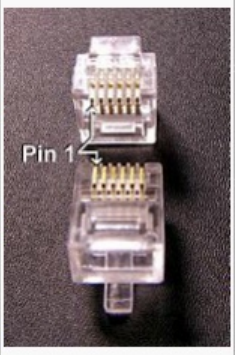
Pin	Name	Function	Color	Pin Numbering
1	ANA	Analog interface / +9V	white	
2	GND	Ground	black	
3	GND	Ground	red	
4	IPOWERA	+4.3V Supply	green	
5	DIGIAI0	I^2C Clock (SCL), RS-485 A	yellow	
6	DIGIAI1	I^2C Data (SDA), RS-485 B	blue	

FIGURE 2.5: NEXT Sensor Interface Pinout

2.5 The Colour Sensor

The colour sensor is a sensor that in digitally mode, can discover the colour or intensity level of light that enters a tiny window of the face of the sensor. The EV3 sensor may be used in three separate ways e.g. Colour function, Reflected light Strength Mode, Ambient Light Strength mode. In this project we will only need to set the colour sensor to recognize the colour mode, Hence we would need to only highlight that mode. In that mode it will understands seven colours, as they are White, Black, Blue, Green, Yellow, red, white, and Brown. This capability to distinguish between colours indicates your robot could be developed to type coloured balls or blocks, understand the names of colours as it has noticed, or end activity once it spots red[26].

FIGURE 2.6: Colour Sensor

Nearly, what we have been mentioned so far were only about robots that used in Educational range, However, there are better and more complicated sensors or motors in Industrial section, but they are far high priced to be used for educational purpose, for e.g. ColorMax-1000 Colour Sensors for Automation Applications,see Figure 2.7 , it is set to read up to 15 colours, and RGB strength, and many more features[27].

FIGURE 2.7: ColourNMix Senosor

2.6 The EV3 Servo

A Servo is a small device that has got an output shaft. This shaft may be set specific positions that are angular sending the servo a coded signal. Servos it is a geared DC motor with a control that has integrated and a potentiometer for detecting the turning angle of the motor. The majority servos work inside of a pre-defined angle, normally between 0 and 180 degrees. The servos have three wires: power, ground, and signal. Simple, small servo motors typically operate with 5V power. Servos are incredibly beneficial in robotic systems. The motors are small, and so are most truly powerful for their size. A servo that is standard 42 Oz/inches of torque, which is pretty strong for its size. It also draws power proportional to the mechanical load. Servo doesn't consume energy that is much [3][28]

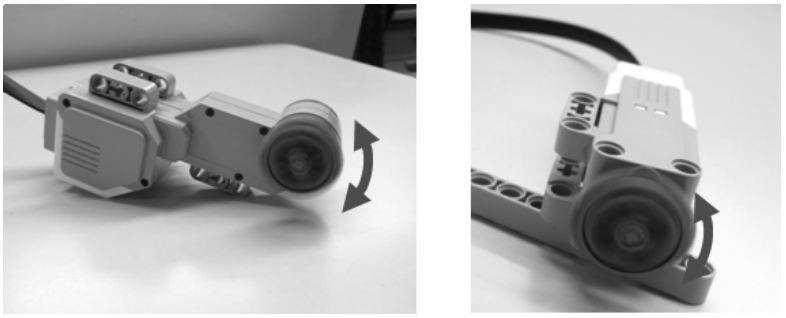


FIGURE 2.8: EV3 Servo

As we promised just to talk about what are important components in this project. At this point in this thesis we are not going into detail for further parts any more except for power supply as we see it the important section to knock at it here and it is relevant to mobile robotic system. Probably, by the time you read this paper probably they are out of dated, due to the Technology is processing swiftly. However, for the sake of references, the design diagram had been added, if as a designer need to be instructed to the up to date information about each part, if names just goggled, you will be directed to hundreds if not thousands web pages and books that they are rich in suggestions that fulfil your requirements.

[3] The servo is the motor that has linear or actuator, the actuator that allows for precise regulation of angular or position that is linear

2.7 Robotic power supply & RaspberryPi

In general, the primary source of Mobile Robots depends on the battery. Robot subsystems will need a range of voltage levels. Thus, we must insist on robot supplies provide subsystems with each effort necessary. Most often the Robots have designed to work a multi-purpose. It Would be ideal to provide a DC battery system combination of switching method, to choose from a wide variety of DC converter. The drive Motors when runs will draw power from the primary DC battery system. Hence, separation and splitting up of voltage are suitable for subsystems of robots, It is utilizing through flyback converter topology implemented to provide voltage control and separation when subsequent buck converters are applied to match the other voltages demands. The Raspberry Pi 2 Model B for use in schools, it offers more flexibility for learners than the leaner (Pi 1) Model A+. The Pi2 B is more useful for embedded projects and tasks that require tiny power.[29]
However while robots getting designed the essential element, it is a Power system. Therefore, this function has to be accurately measured and scientifically calculated before the robot initiated. Hence, will be ideal to talk about what power-source is Raspberry Pi2 will need, and how much will consume compared to its earlier counterparts Raspberry Pi A and Pi B+.
Raspberry pi2 at a load of full four core will use the same power as model B+ at idle[30]. Bear in mind, that electric energy consumption in every Robot, will depend on the functionality and tasks and Hardware or software that will be in use, that means more parts linked need more power.. The "wifi" adapter, for example, it is very common tool, that attached to almost every single devices that they have to function on the networks. Wifi connectivity will slightly take additional power, and there are innumerable tools and features that Robotic system relied on beside wifi adapter, all based upon which performance Robots made for[4][31]. On that point are various of a different power source to select.

For robots to perform efficiently, would be a good idea if batteries and photovoltaic or cells have combined together in one project collectively[32]. As this project, mostly scope on how to build a surveillance robot with image processing, and will highlighting the related topics, However, will not concentrate on it or go through the building steps. The aim of highlighting subjects it is only for purpose of understanding the robot requirements. As mobile robots have to go around need to be autonomous and for that required the robot to be supplied ed with cordless power to enhance their mobility.

[4] According to TrendNets website, that Wifi Dongles will pull anywhere between 100mA to 500mA, based on brand and models

	A+	A	B+	B	Pi 2 B
	/mA	/mA	/mA	/mA	/mA
Idling	100	140	200	360	230
Loading LXDE	130	190	230	400	310
Watch 1080p Video	140	200	240	420	290
Shoot 1080p Video	230	320	330	480	350

FIGURE 2.9: the Comparison between all Pi power consumption

FIGURE 2.10

conclusion

The robot modality and mission poses challenges that are distinct power. Generally speaking, battery power recommended over internal combustion due to the logistical difficulty in transporting liquids that are flammable. yet, what's needed for every robot rescue application is primarily As yet not known.

Chapter 3

Installation

3.1 The Installation

This chapter will guide through the installation of the software to show how to run The raspberry pi, and will over view the Path planing. Installation of Operating system required to get a new out of the box. To get a new NOOBS, please refer to the origin home of raspberry pi reference [33]. The Raspbian OS NOOBS have been chosen for this project. However, there are more than one OS as a third party available, for more information See figure 3.1

There is no doubt Raspberry Pi is a computer by itself, but still needed a few things like software and hardware to control it remotely. The application tools which are needed during the implementation of proposed this project has been given just below:

- a version that is optimized of the raspberry pi to support programming in python when it comes to the servos.
- Xming server who will connect to, access the Raspberry Pi remotely by using Putty.
- In this project, we will control the sensors and the servo and the raspberry pi via programming language Python.
- Html and JavaScript for sending the pictures.
- a web server to stream the video.

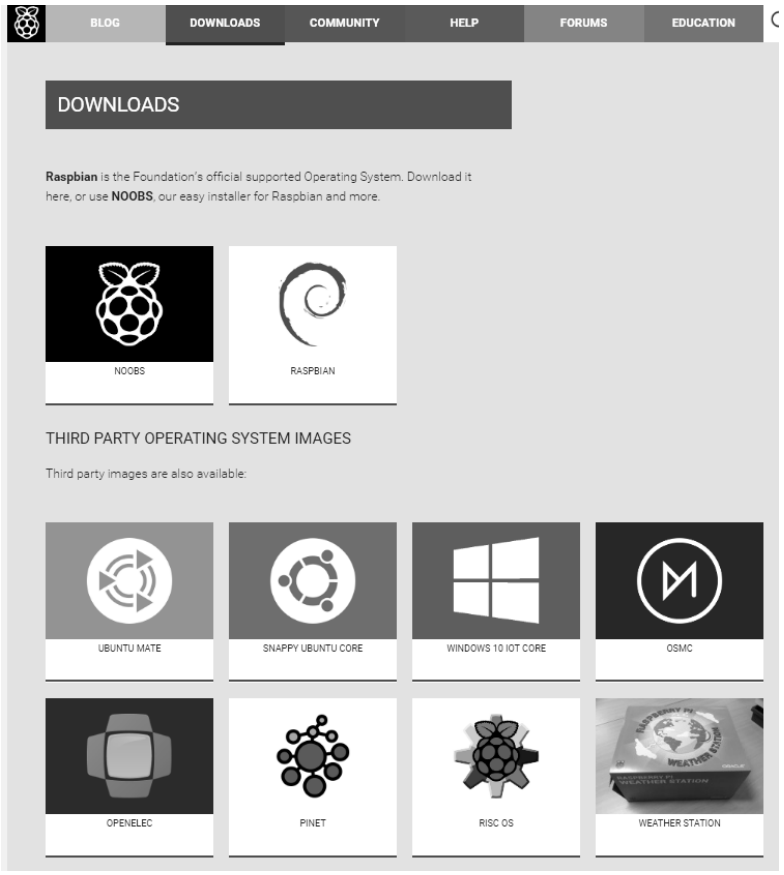


FIGURE 3.1: Raspbery pi different operating systmes

Installing the Raspbian "wheezy" on the raspberry pi. The raspberry pi powerful raspbian is continuous LXDE[1]. Rasbian[2] is essentially a complete Linux distribution for the Raspberry Pi and comes with a GUI based off of LXDE and pre-installed programs for word processing, games, development, and so on. As a starter, you will need a TV or Monitor, Keyboard, and mouse and they all have to be connected to raspberry pie. The Steps are as follows:

- If you raspberry pi board not bought from origin distributor, it is a big chance that you don't have the pre-installed Rasbian SD[3]-card with it. You have to go back to this reference [33] to get all information and cleaner version of the raspbian.
- unzip the image file of raspbian after downloading to SD card using Win32Diskimager tool.
- once the card it is ready and all files have been extracted to SD card, time to go.

[1]The name LXDE stands for "Lightweight X11 Desktop Environment"
[2]Why Raspbian advised? Independently of The Foundation taken care of it. IT is Based on ARM (arhf) the port structure formerly designed for ARMv7.
[3]4 GB SD card or more recommended, However, minimum of 2 GB SD card is required.

FIGURE 3.2: Succeded Raspbian installation message

- The SD card has to be inserted in the raspberry pi micro SD card and raspberry pi powered on, There would be a led indicating while the raspbian is loaded, the message as shown in figure 3.2 and figure 3.3 after installation is succeeded.

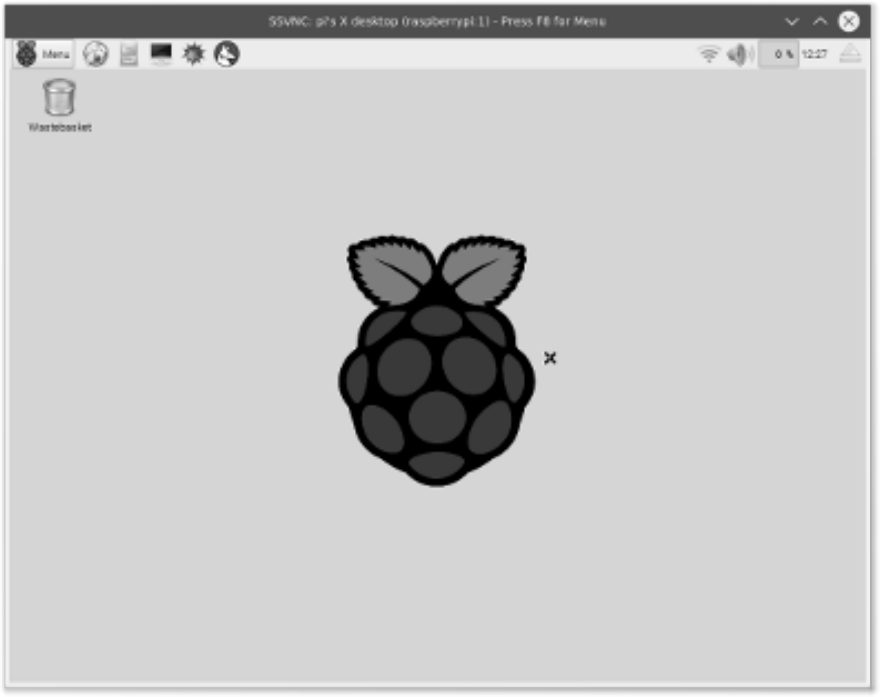


FIGURE 3.3: after Installation of raspbian on raspberry pi

3.2 Installing putty and Xming server

Using the main monitor or TV is not a practical way to run raspberry pi always, hence, In this project, the keyboard and mouse only needed once when the raspberry pi configured for the first time, only Wi-Fi connection has to be made manually. Best way to connect to the raspberry Pi, it is a remotely connection via SSH on putty[4] software, it is a tiny software can be installed on the desktop and along with Xming[5] server[34] . So the Extra monitor not necessary too, the pi will be controlled via its IP address with SSH terminal. Bear in mind, there are a plenty of SSH terminal to be used, however the putty is have been chosen in this project. So the Wi-Fi dongle or internet cable must be plugged into Ethernet. first you need to update package:

cod: $sudo apt-get update

cod :$sudo apt-get

Its very easy to install Putty and Xming, only Putty need to be configured as shown in figure 3.4, click open and that is it should get windows like figure 3.7

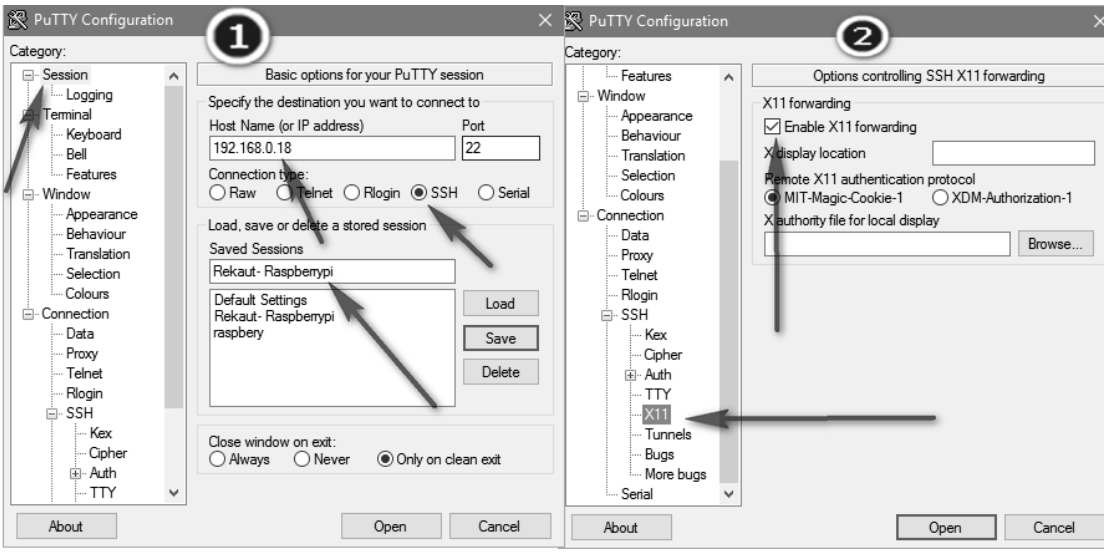


FIGURE 3.4: Setting Putty with Raspberry pi IP

As there are many different ways to connect remotely to your PI, it is would be a good idea to show another, even easier way, however, this method simply depending on what and which OS the PC is running, this is a shortcut way to connect to your raspberry pi it is a faster and clear image if connected to any Windows10 Remote Desktop Connection. It comes ready with Windows at most need to enter the raspberry pi IP Number that obtained from beginning while it has been connected to a TV or monitor while installing and configuring the raspbian OS, see the steps in the figures below.

[4]PuTTY is Project Xming's preferred and integrated X terminal emulator for Microsoft Windows

[5]Xming is the leading X Window system server of Microsoft Windows.it is fast and lean,very simpl to install and portable

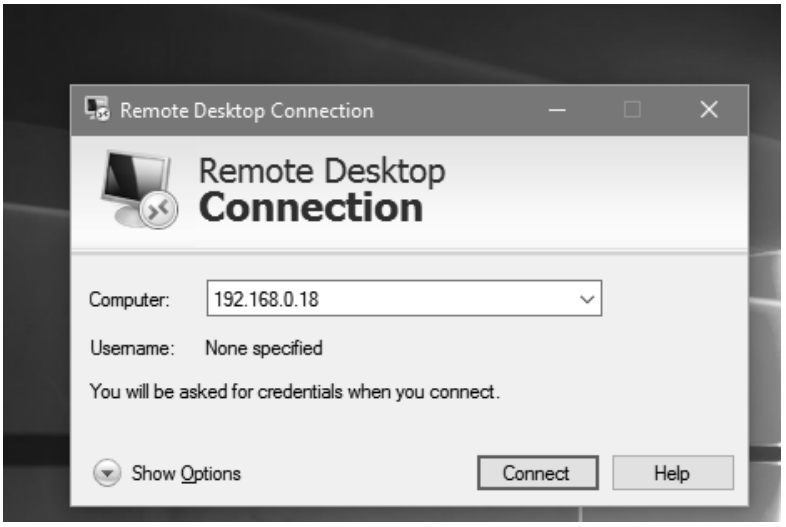


FIGURE 3.5: Remote Desktop Connection Win10

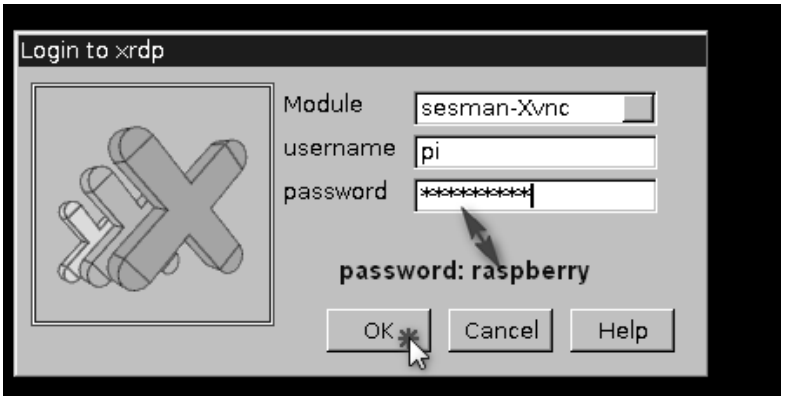


FIGURE 3.6: Connection to the Hosted devise

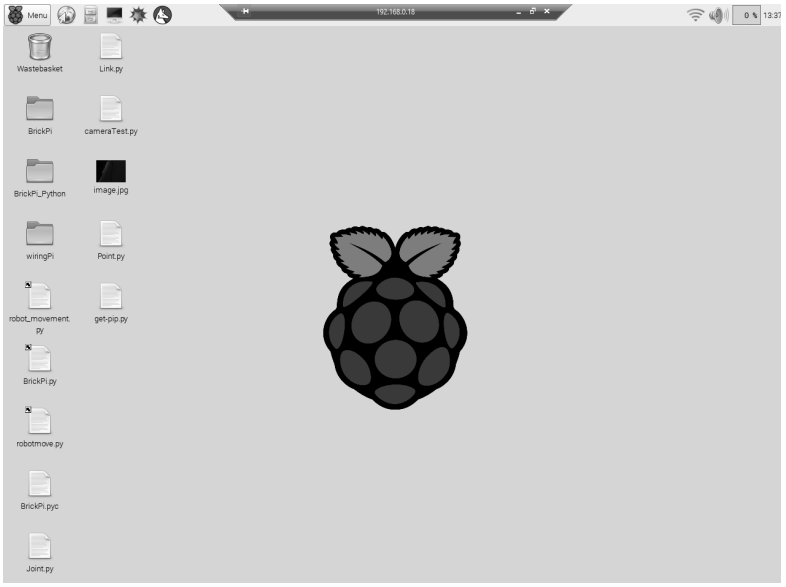


FIGURE 3.7: opend Raspberry pi applicaiton on PC remotly via Wi-Fi

3.3 Install and Run the Brick Pi

In the chapter 1 section 1.8 BrickPialready mentioned as one of power full extra part which playing the bridges role again it is a board that is slide-on from Dexter Industries that converts the Raspberry Pi into a robot. The BrickPi helps to connect LEGO mindstorm's sensors, motors, and parts to the raspberry pi; it will effortlessly switch the credit card size raspberry pi board computer into a robot and empower it. This thesis will help to set up all the software and other dependencies for the BrickPi along with python codes to configure and set the Brick pi to respond to the raspberry pi. However it has a exclusive module code from Dexter Industries to be installed and executed to run sensors and motors on it is ports, for more information and documentation see Reference [35].

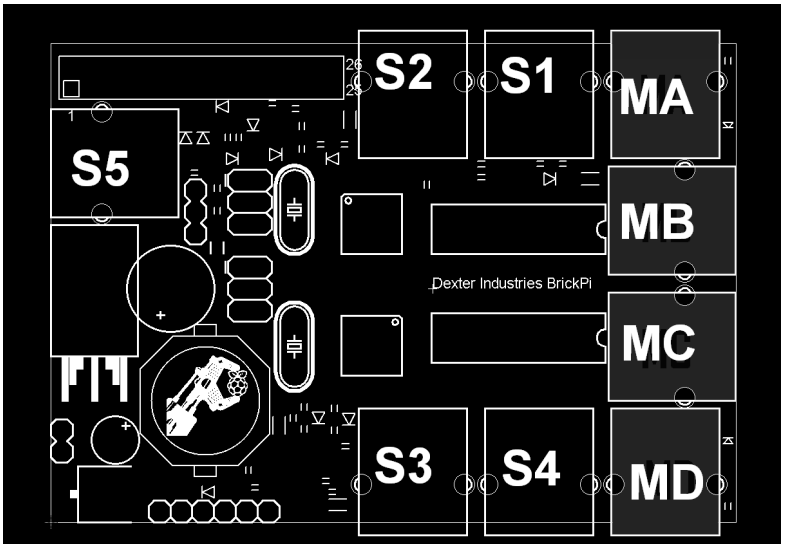


FIGURE 3.8: Brick pi motor and sensor ports

Test Failed: At This Point, all parts are combined. It is time to conduct the test to see whether everything is responding as it is expected. the LEGO-Motor-Test.py file loaded from DEXTER to test motor ports, the code executed but failed to operate despite the console attempt to output that the motors trying to run.

The solution: It found that the SD-card had been formatted and knew image reinstalled after format, That means package has to be downloaded and installed manually

while origin SD card formatted and data wiped away. With this piece of code running from raspberry pi's console will do the job.

```
git clone https://github.com/DexterInd/BrickPi.git
a folder named BrickPi will download on raspberry pi,
directory containing file named setup
will be pointing it is directory by writing:
$cd BrickPi/Setupfile
there is a script called install.sh has  to be made executable,
$sudo chmod +x install.sh
Run the install script
$sudo ./install.sh
```

test performed again all motors working fine.

3.4 Robot Programming Languages

By the time getting onto the programming phase, the robot parts should have been assembled, so that means we have an experiment on how and which sensors and electronics parts to choice, in result However, remember that the robot will remain a piece of metal, no matter how beautiful and nicely look like the robot, or how expensive is the robot unless it is becoming alive. For a Robot to become live, it will need a programming Language to control the robot. Mainly this research will concentrate to find the security issues, rather than Implementation to identify these problems we need to look at the performance of the programming Language, and it is the vulnerability to the attack.

3.4.1 Why Python?

Developing programs for Robots it is a complicated part, especially with the growing of Robotics industrials, while robots More or less to dominate all aspects of our life, the programming languages have to be taken into consideration upon the design of the Robots, as different hardware needs various functions. Programming requires the massive size of the codes to design an advanced robot. That would make re-using of the code, not a natural process, the software for The robots must contain a huge pile of coding, starting from easy level to the challenging level, at the same time usually need

to provide software integration.

There are many languages available that might be used in the programming of the robots. As part of my research, I realized that once it gets to talking about robots technology one way or another has to have Python to be there to talk about. This encouraged me to ask myself why simply Python is so desirable to programming most of the Robots, is that only because Python is supported language for ROS? No, it's not just that.
(ROS)[6] Robot Operating System is compatible with Python, C, and Lisp, despite there is not helpful documentation to start off easy with ROS but still it is the preferred application that would make sense to say Python is popular in that sphere. The main ROS client libraries (C++, Python, LISP) are all ROS's Programming languages then again, at this point we have to ask ourself as a Robot designer while there are few different free source Languages libraries available alongside with Python on the same framework what will make Python distinguished to the rest?[36] despite (ROS) is the largest framework but it is not the best, the best framework in Robotic system designing is not exist yet.As part of this research containing implementations of the code in Python, thus it would be good to take advantage and talk about factors that Python have been selected for.

3.5 Summery: why Python?

Since, Raspberry Pi is a Linux system based, to program it almost altogether the same as any Linux system. In fact, it is a single board Linux computer. It has nearly all the compilers, for e.g. JRE, Apache, C, PHP, anything else has ported to that CPU. Despite The Raspberry Pi Foundation suggests Python as an official language for learners. However, It is not compulsory to make use of Python to program your Pi. It's all regarding possibility libraries are currently supplied in the language in which developers are comfortable with[37]. Crucial that the Language is chosen allows all necessary implementations or performs the required functionality for, e.g., face detection.
So why Python?

- First thing's first, There is a big community, they recommend such numerous different things. The developer never lost or stuck.
- It is a simple and straightforward syntaxes. The point, it is very easy for beginners.
- Despite it is easy, yet it is a very powerful language can be used in making websites, games, and computer programming.

[6] Robotics research center, Willow Garage invented ROS to solve the common platform problem.

- There is numerous libraries and tutorial available on the Internet.
- To use it with raspberry pi it can define raspberry pi pins effortlessly.
- Open CV supports Python, and many image-processing tools are available.
- As said the standard Raspbian Linux distribution which comes with raspberry pi; it is well set up with Python.
- It is faster than the others on the raspberry pi as " Hello World" tried on Java and Python, but Python executed the code one-second faster.
- Python contains extensions for Raspberry Pi GPIO.
- Prototype is quick to boot-up with python.
- Python is excellent for a short road.
- It is very painful to write file I/O, string parsing and (web sockets)[7] in C++ compared to Python which it would take 5-minutes task.

[7]Web-socket is a protocol providing full-duplex communication channels over a single TCP connection

Chapter 4

Path Planing

4.1 Overview of Path-planning

Eventually, the context for this project has established, it is time to look at the task that is specific mobile robots require to achieve. For a robot that is mobile to be regarded a functional robot, either it is a self-driving car or surveillance robot,it should be in a position to navigate its environment. Autonomous navigation systems in robotics consisted of three factors:

- Mapping
- localization
- Path planning

An autonomization of robots will demand to create a risk-free environment to navigate safely and avoid collisions to other subjects or people. The aim of mobility is to accomplish its mission independently, after inputting some code to program the robot, for all intents and purposes. The path planning should meet satisfied constraint conditions. It is already having noted that the most important is to steer clear of known and unknown obstacles. To navigate from start to end point with minimal energy consumption and shorter time planner, in meanwhile, we expect robots to operate as quickly possible; precisely. Unfortunately, in the real world, there are always complex and surrounding dynamic and unstructured environments! Hence, it is really hard to develop the entire system to deal and handle all encountered problems. As issues not entirely wiped out yet, researchers are still trying to find the optimal path from A to B. The term that will be measuring the optimality is considered travel distances and confidence value.

The slope and roughness of the surfaces also have to be considered as a measure of path efficiency. To mobilize the robot to travel safely, requires a path - planning. For that surrounding maps needed, where the robot can confirm its location on that map. The robot can localize itself,able following a plan that supplied and capable of escaping the obstacles while running. So for that developer need to learn how to create a plan, or localize a robot and find how to deal with uncertain position information, all those functions called a path- planning[38]. In general the path planning algorithms for the shortest path planning:

- plan a path from an existing place to the current goal, it should be calculated by distance transform method[39]
- Stick to the path.
- in the case of facing an Obstacle tray to change to a new plan from current position to the goal by distance transform algorithm. In this project the robot will stop by obstacle and raise alarm or sending an image of the obstacle.
- all steps has to be repeated if for any reason stopped functioning or robot lost until the goal is reached.
- all time and distance, obstacles should be recorded during the action.[40]

Essentially, there will be many different path points on a grid-based map. That means too many solutions take on a journey between two points. The robot not allowed to try all paths available, hence there is an algorithm to select its path and aim of the algorithm is to:

- the path generated between start and End is simple to follow if not facing any obstacles.
- the paths that are generated have to be a big differentiator as much as possible to let robot discover innovative solutions. [41]

In a very short description, pointed out path planning it is an essential to mobilize robot collision free, so it's would be ideal to mention the most effective algorithms that can be used in the path-planning. Bear in mind if we want to research path planning as a Topic by itself will need few research papers.[42]

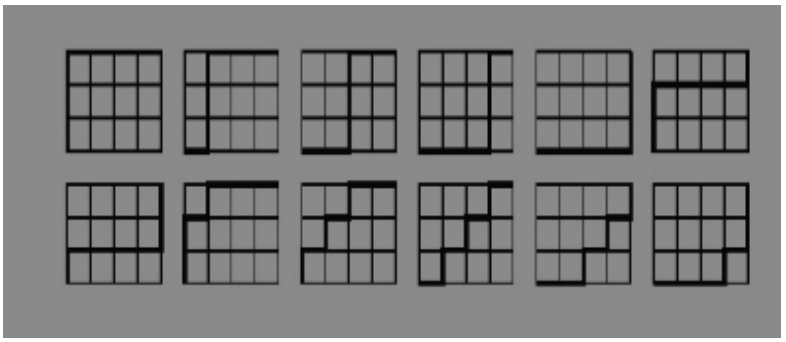


FIGURE 4.1: 3x4 grid divided into paths segments where the robot can choose between them

Different Algorithms:

- the Graph Search Algorithm - based on node-edge notation
- The Breadth - First Algorithm: Breadth-first algorithm runs with the technique branching from the starting cell to the neighbour cells (just traversable cells), un-traversable cells, cells out of boundaries are discarded) until the goal cell is found[42]
- The Dijkstra Algorithm.
- A-Star (A*) this is the most common and efficient used algorithm in the shortest path.
- Multi- Goal Cells: This algorithm is applied to the cell multiple goal which in the reality multi-goal cells scenario is most common.
- Maze solving algorithm. we will applying this algorithm for robot to follow the Line in this project.

NOTE: In this paper not all related algorithms have been mentioned!

4.1.1 Localization

In this section tried to find the best solution that indoor surveillance robot possible to find, with the available sensor and tools. Since Lego Mindstorms EV3 has been used in this project, that means limited sensors (light sensor or colour)sensor and camera mounted on the robot body. GPS is the best known positioning method so far, however, because the robot designed to work inside building, GPS is not a good option even though there is a third-party GPS sensor for the Raspberry pi. Therefore searching for alternative is not easy specially when the robot environment is not standard and keeps

changing continuously for example inside our houses the furniture and stuffs are always keep moving around, and some times disappear for the robot by putting them in closet. Again the sentry mobile robots has to be Autonomous or controlled by the server or Hybrid system which combination of Autonomous and controlled from any computer, The ideas of Autonomous robots are able to gain details about their environments and work with a prolonged time frame without human involvement. Simpler robots that are autonomous need infrared or sensors that are called ultrasound to see obstacles, and permitting them to navigate across the obstacles without human control. More complex robots use stereo vision to see their environments; cameras provide them with depth perception, and the software allows them to find and classify objects in real time. Demand's on the robotic system is increasing. Robots need application systems to

perform indoor and outdoor tasks in a variety of environments for the robot to be able to a self-localization. A system that allows a robot to switch between indoor and outdoor environments.[43] for indoor navigation method, there is a system called Identification (RFID)[1] The RFID tag is used as a position's tag on the wall and the antenna of an RFID tag sensor mounted on a mobile robot, the robot will pass the labels and read them all tags have a unique ID number which supply information on the robot's position[44]. A (GPS)[2] The receiver will get signals from several satellites; that is why the numbers and the position will change. The GPS is fascinating to be used for outdoor path planning in robot navigation. As it does not need the individual devices reflectors or targets. In a dynamic environment, full details will alter together while obstacles move, and that movement of the details will raise the difficulty of the path planning in dynamic environments. That makes some of the planning methods not good enough to be implemented for examples: Visibility Graph[45], Voronoi diagrams Grids[46] instates a genetic algorithm introduced by Wang Sillitoe and Davide[47] to tackle problems in the moving obstacles. Then again, this technique has downsides. Therefore, traditional path planning methods, such as Visibility Graph, Voronoi diagrams, and Grid[46][45], Are unsuitable for planning in dynamic environments. Recently, Wang, Sillitoe, and Davide introduced a genetic algorithm based navigation method to deal with the path planning problem in an environment with moving obstacles. However, this approach still has drawbacks: local minima results may appear. But still research in shows significant change levels, improves quality of the algorithm needed. Either in static or dynamic environments, path planning has extensively studied for the decade, in particular, for the

[1]RFID: is a technology similar in theory to bar codes

[2]GPS:stand for Global Positioning System. A worldwide MEOsatellite system that is navigational by 24 satellites orbiting the earth and they corresponding with the receivers on earth..

multiple robots as their goal to move multiple robots in a typical work environment from start to the goal configuration in hope to avoid collisions with obstacles and damage it. Eventually, critical problems in the collection of robots are interacting with the motions coordinate in the same environment. If a team of robots is sharing the same workspace irrespective of the objective, they should escape interferences between them, For that, there are a variety of ways to approach. The solution relies on goals of the robotic group, and often the paths are clearly planned and coordinated in advance. For instance, in a warehouse, the team of robots will increase the system of autonomous by directing each of them to the task that inherited, either time, space, or functionality. As we aren't discussed in this paper extensively about algorithms that have been used as a solution in that area, rather we only want to show the readers, there are tons and tons of documents available to go through and get information depending on what type of robots intended to build. Bear in mind there are many open issues regarding multi-robot path planning and coordination remained yet.

4.1.2 Localization Testing and Problems:

- First effort :

A first alternative solution would be to put on fixed position sensors. That robot may use to determine its position inside of the room or the building, figure 4.2 either by receiving a radio signal from the transmitters in the room or simply by using an infrared sensor which they fit against the walls or floor every time the robot pass one re-calculate his position.It is, in fact, The Trilateration method which is determining the robot's position based distance measurements to beacon sources, there would be three or more transmitters mounted on the rampart in the room. However, this technique is very costly for education purposes.[48] There are more than one problems with this method beside it is incur high cost there is other problems problems accrue because it is work only when the robot is with in the triangle or the range of sensors. [49]

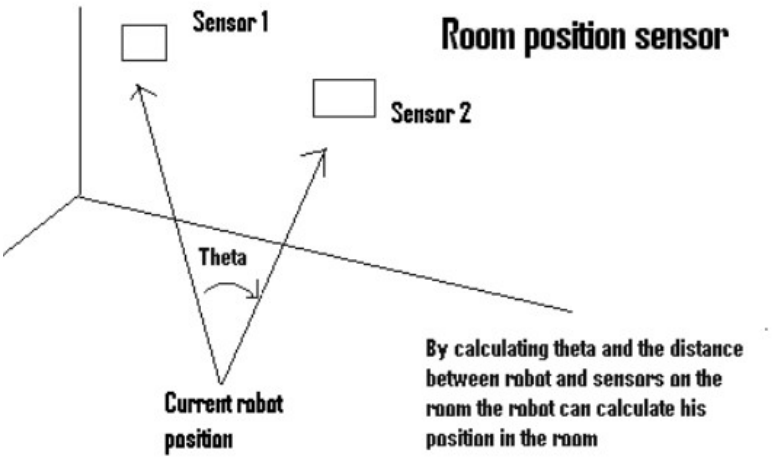


FIGURE 4.2: Simple Room position sensors mounted on the Wall

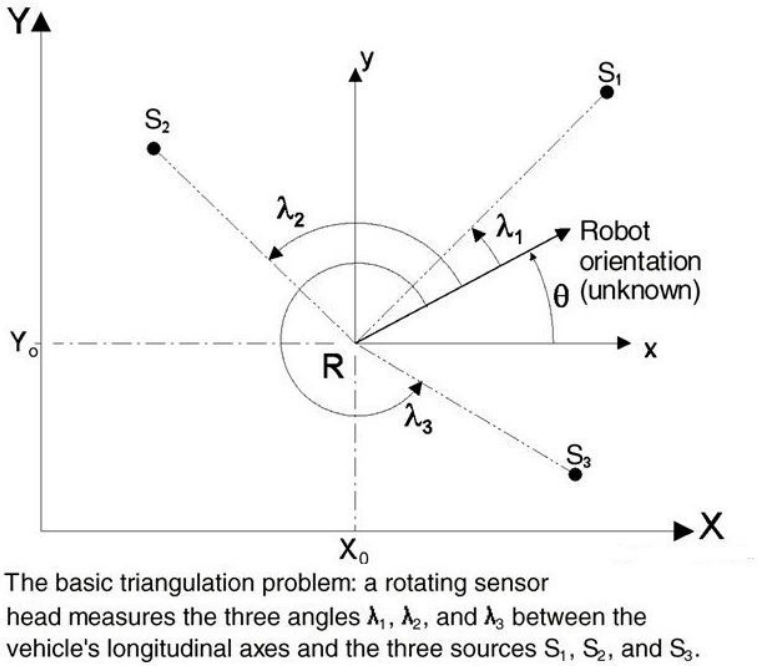


FIGURE 4.3: Trilateration method

- Second effort:

The second effort, is using the light sensor or the camera, by sketching lines across the floor in a distinctive colour or size to represent a position that is different in the room or the building. the robot has to drive over the line to keep reading these lines by this will determine his location. This technique will rise one problem. In the most rooms or buildings, particularly in (offices, houses, halls) drawing lines on the floor, are unpractical. It is impossible sometimes Due to floor materials e.g... carpets. However, this scheme would be suitable to be used in warehouses and libraries where the building has divided into sections. Despite this, techniques are not preferred for solving the positioning here. However, in this project, it will be chosen as a line track method.

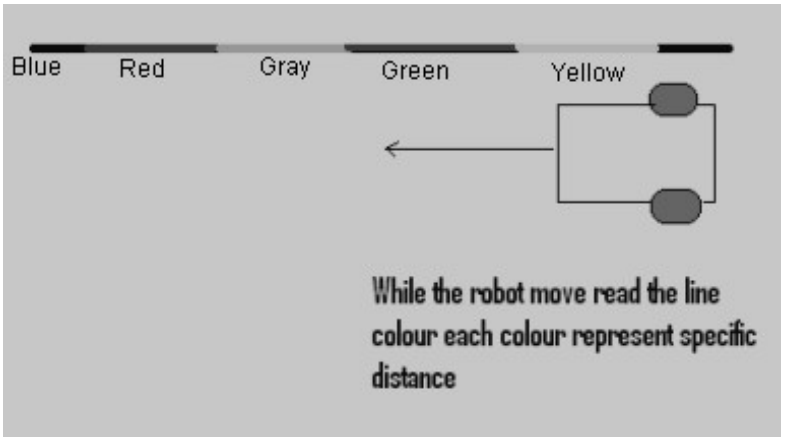


FIGURE 4.4: Robot reading coloured Line to determine it is positions

- Localization solution

Navigation is a difficult task, because there are some complex issues. Some navigation complexity's issues are limits of the computational power. Difficulties in the detection and recognition of objects. Challenges to avoid collisions with objects, a problem which set to use of information provided by the environment. Due to the limited budget for this project for the localization, web camera has been used. Despite, nowadays the technology, providing very advanced indoor Navigator sensors and tools. Unfortunately, most of this navigation systems are used commercially, which limiting individual researcher students to get benefited from it or in the small individual projects , for instance; a GPS repeaters[50]. It tried the cheaper and simpler method at the moment for this project. deployed a web cam to stream over the local host or the Internet from anywhere. By this, any-time the robot can be found where it is locations about, and this done by installing a motion module onto the robot board, of course, it is Linux OS, for, e.g.,... By inputting these codes.

```
\textit{\$ sudo apt-get install motion
$ sudo apt-get install libv4l-0
$ sudo apt-get install uvccapture

the next step to modify the motion can file to set the daemon to ON and web-localhost to off.

First we will edit the motion.conf file by typing:
$sudo nano /etc/motion/motion.conf

It is a  big file.
edit are:
# DAEMON = OFF (turn it  to ON)
# Webcam\_localhost = ON (Switch it  to OFF)
# ffmpeg\_bps 100000 (from 500000
$ motion start
$ motion stop for exite the software.}
```

The port number has to be forwarded to the modem. If the camera opened from outside the local host, google your modem IP to find your modem IP from outside. Tail the port number to your IP, for e.g... 79.97.62.225:8081. Before that the motion has to be started from Lx-command line. The camera mounts on another motor, and this is certainly for the low budget, at least an individual can see where maybe the robot precisely about through the network camera mounted on it. This method is definitely contrary to the Single Web camera [51], which using a fixed camera on the wall to observe the Robot passing by and sending an image to confirm the location. However, in this project, the single cheap web camera mounted on the robot and any time, a user from the controller like to see where the robot is can see through any website by inputting the IP and Port number either the local network IP or to connect to the outside worldwide. There would be some security issues and IP vulnerability. However, the port number range can be changed to be avoided from scanning or at least interfering with other devices. The Intrusion and port scanning can be limited, but it is impossible to protect against all attacks, particularly for using the web camera from the website, and the port 80 will be visible throughout the world.

Future work As navigation is a core capability of the mobile robot. Since back to year 1960s, the navigational systems have been researched by researchers, it has been only since the early 1990s was this robot affordable, it was on board sensors and enough computing power. As a consequence, most researchers studied and developed the navigation techniques, Applying simulation and assumptions about how real robots will be physically functional. Two of the most widespread assumptions of these researchers

proved unfortunate in the review. First, it is assumed that the robot it may accurately be located in each update. This assumption is based on another assumption: that the sensors could give a representation that is accurate in the world. This is most of the time not correct. The sensors are often noisy and also have vulnerabilities. Therefore, a robot must operate in the presence of uncertainty[52][53].

The future work and for an indoor purpose based on that assumptions, the Camera as a localization sensor very cheaper to be used. Hence, instead of live streaming, it is easier to stick some Numbers or letters in locations that the robot is driving through. By identifying and recognizing these figures will be able to confirm where it positioned on the map. In that situation, the robot has to send an image to be processed for example The OpenCV image processor, first has to convert the image to grayscale and then extract the background of the picture to get exact text. Again for the future work I will limit attacks by setting a good firewall and Intrusion Prevention. system(IPS)[53][54].

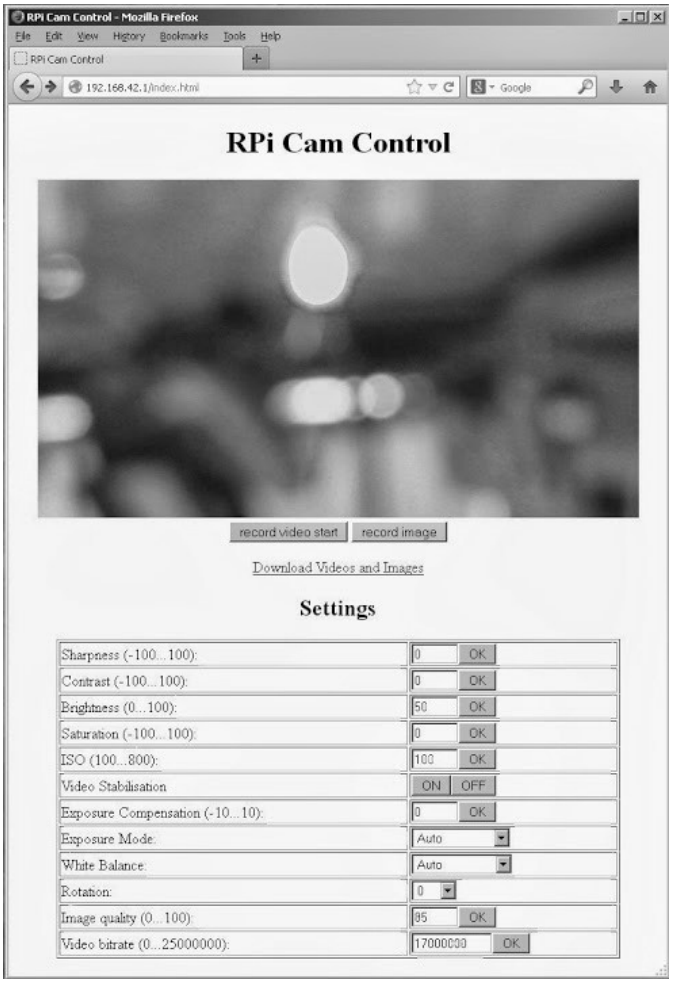


FIGURE 4.5: The website interface Html and php, that used for Localization the robot via Images and live video streaming.

4.2 Line Tracking

In the complicated following line track projects, only one sensor would not be enough for an agent to track the line. In fact, with a single sensor the robot can not be a line tracker; it simply follows one of the two edges of the line. The robot with a single sensor does not know if it has been drifting from the line or not and arise problem while robot attempting to turn or meeting the cross lines. Hence, the one sensor has provided to this project it will limit line tracking options. In another hand, I would prefer the maze track solver, which the robot has to have at least three sensors, and the best is five sensors. Most robot parts are linked with the GPIO port of the Raspberry Pi. The exception that is only the Wi-Fi USB dongle, which is used by the application on the Internet to perform start, stop and shut down. The Wi-Fi will give access to Raspberry Pi through SSH connectivity in a remote way to the developers. The motors are controlled via (PWM), using 'DRV8835 Dual Motor Driver Kit for Raspberry Pi B+' the board is connected straight to GPIO and fill pins 1 to 34. The motors powered along with 5v, which controlled through switching step up and steps down regulator, sees figure 4.6 Which can be attached to the side of the DRV8835 Dual motor driver kit.

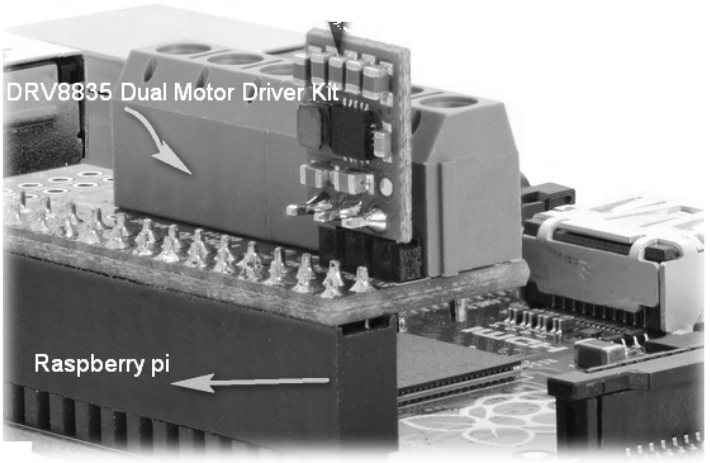


FIGURE 4.6: Pololu DRV8835 Dual Motor Driver Kit for Raspberry Pi

Line Tracking Algorithm for this project: The first step to putting a robot on the track was programmed only to follow the path; it does not need any effort. # if the middle sensor on the back right and left in the white, forward motor = sat to Max speed. #if the left sensor on the black middle and right at the white will drift to the left vise verse. However, For simple line follow what has been done, it is just a simplified PID Control Algorithm, and the reason to use PID is to minimize the errors of what we are trying to control for more info see figure .... The second part of the line following in this project will be a Maze Algorithm solution.

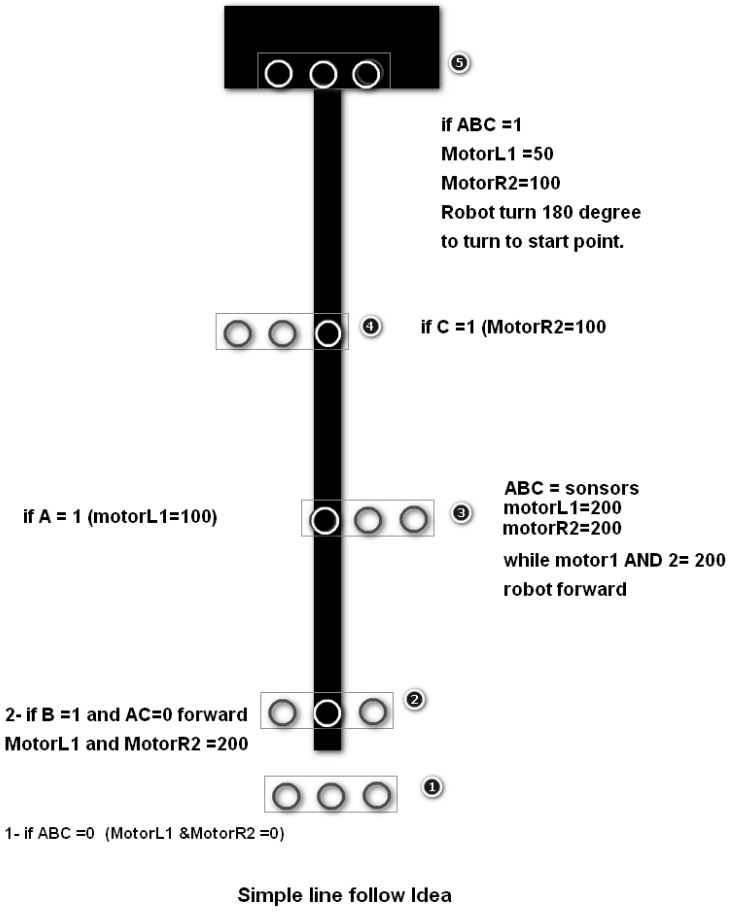


FIGURE 4.7: Line following algorithm based on PID control algorithm

4.3 Maze solving Algorithm

You can get a certain amount of different algorithms to resolve a maze, which is, automated methods for the clear answer for the maze.

Mouse randomly, plus the followers of the wall, designed pledge, and Trémaux's algorithms to be used the maze by the traveler won't have prior knowledge of the maze. Even though the filling dead-end and shorter algorithms designed the course to be utilized by an individual or a PC that one may start to see the whole maze program at the same time. It is known mazes usually do not contain loops as 'just connected' or 'perfect' maze, and equal to a tree in graph theory. And thus, some algorithm to solve the maze is closed linked t graph theory. Intuitively, if one pulled and stretched tracks within the maze in the means that is correct, it could be the outcome to resemble a tree[55].

4.3.1 Wall follower

The traversing maze rule which identified as a wall follower, usually known as either left or right-hand rule. Continuing to keep one hand in contact with one wall the maze will indeed be assured not to be lost. It will bring the robot to the end point or the exit if any exist, otherwise, will bring it back to the start point in the loop manner until finding the door out if any there[55].

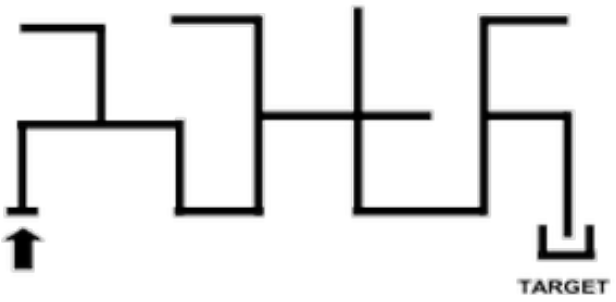


FIGURE 4.8: Maz blue print

In this project the same line follower sensors used for the maze solver, commonly maze solving requires five or more sensors, by passing robot the cross line it can be noticed a little before deciding right or left to go, it is happening to correct the shortage of few other sensors.

4.3.2 Approach for handling the maze

- The focus on (the black disk) is placed wherever in the maze.
- The robot is laid wherever in the maze. Robot lookup and detects the final point following the left-hand rule.
- Robot determines the quickest path by an algorithm similar to the dead-end filling.
- Robot moves through the shortest path back to its beginning state and "nods its head".

The Algorithm has been made simple as follow:

- If robot able to go left
- if not left carry on drive straight

- if robot can turn right turn right
- when reaching dead end turn 180 degree.

the codes for the maze and line follower is published and free for use can be seen on the link provided, as tested by this project there is no problems and code can be modified very easily .
http://www.retas.de/thomas/raspberrypi/pibot-a/robot-ms.py.html

```
# -----------------------------------------------------------
# Read sensor input
def read_sensors(*sensor):

    L = GPIO.input(GPIO_left)
    M = GPIO.input(GPIO_middle)
    R = GPIO.input(GPIO_right)

    if len(sensor) == 0: return (L, M, R)

    elif sensor[0] == "left":   return L
    elif sensor[0] == "middle": return M
    elif sensor[0] == "right":  return R

# -----------------------------------------------------------
# Drive some distance, time to sleep is calculated from
# given value (val) and result of calibration (cal).
def drive(val):

    sec = val * cal/500
    sleep (sec)

# -----------------------------------------------------------
# Calibrate: Drive two 180 degree turns in both directions
# and measure time needed.
def calibrate():

    tl1 = turn("left")
    tl2 = turn("left")
    tr1 = turn("right")
    tr2 = turn("right")
    cal = (tl1 + tl2 + tr1 + tr2) / 4
    print ("CAL:", tl1, tl2, tr1, tr2, "=>", cal)

    return cal

# -----------------------------------------------------------
# Turn left or right: at first leave the black line just
# under the sensors (if there is a line), then continue
# turning until the black line is reached again.
def turn(dir):

    if dir == "left":
        motors.setSpeeds(-v3, v3)
    else:
        motors.setSpeeds(v3, -v3)
```

FIGURE 4.9: soem code for Maze solving

Chapter 5

Image processing and Motion detection

5.1 Motion Detection

The raspberry pi camera is a small printed circuit board with a camera on it. The camera comes with the ribbon cable that can be connected right away to raspberry pi board to the slot made to its origin. Despite it has a tiny size but still have a good quality compared to it is a size which it is 5MP, however, should be noticed there is a very high-quality PX small camera builds into mobile calls. That's what is available and by the time of writing this thesis, 8MP out which it would be even great. Now we are building a simple robot and so what is next? What is the use of part of metal just driving around and make a loud noise It is the time to look to hook something to the robot and make it useful for, e.g., Motion detection, face recognition or two images comparing.

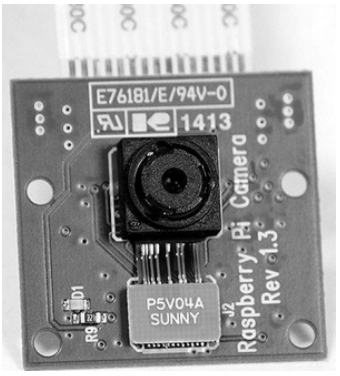


FIGURE 5.1: Raspberry pi camera

To be able to provide for email notifications to send, the operating system software which allows emails to be sent as needed. The software is called SSMTP; it is the program that enables a system to pass email from local computer to the host. SSMTP ideal for situations that require the alerts to be sent. So it is useful when notifications that are sent. The SSMTP can be installed using the command below. of courser the raspberry pi raspbian always has to be updated before installing a new modules.
$sudo apt-get install ssmtp

Next, the motion is the program that is main, which allows the camera to take a snapshot as soon as the motion it detects. Together with the motion, also, it has special packages, which can as well be installed to enable the program work correctly.

```
sudo apt-get install motion
sudo apt-get install -y libjpeg62 libjpeg62-dev
libavformat53 libavformat-dev libavcodec53 libavcodec-
dev libavutil51 libavutil-dev libc6-dev zlib1g-dev
libmysqlclient18 libmysqlclient-dev libpq5 libpq-dev
```

The majority programs get configuration file that allows the user to make any adjustments to modify or advance this system of which the program run on it, or to comply the the program to the system which installed on. while the file motion.conf is opened make sure to change
daemon to ON
threshold 300
target_dir /home/pi/Securityshot
on_picture_save python camera.py
To detect movement the Threshold amount of pixel has to be changed to 3000 just to make sure that the image captured after movement noticed in front of the camera, the directory it is the place where the taken images will be saved on. It is passable to make any directory even to save the taken images on the USB only need to write a directory destination after target_dir/ The algorithm is if motion detected taken a photo and sending an email[56] . Whenever a picture saved this bit of code will run to send, and this code has to be enabled in configuration file again and it is designed to use SSMPT and send notifications each time pictures are saved.An email and users password has to be set up in the conf file, And the code is:
on_picture_save python camera.py for more clarity on conf file please have look at figure

Every time any modification that are made to the conf file the raspberry pi has to reboot. Now it is time to take a picture with the command below.

. /Motion -n -c motion-mmalcam.conf

```
import subprocess
import smtplib
import socket
from email.mime.text import MIMEText
import datetime
import os
#Destination email
to = 'Type your email address'
#user gmail account
gmail_user = 'type your email address'
#gmail password
gmail_password = 'type your password'
#everything is verbatim from this line on down
smtpserver = smtplib.SMTP('smtp.gmail.com', 587)
smtpserver.ehlo()
smtpserver.starttls()
smtpserver.ehlo()
smtpserver.login(gmail_user, gmail_password)
today = datetime.date.today()
arg = 'ip route list'
proc = subprocess.Popen(arg, shell=True,
stdout=subprocess.PIPE)
data = proc.communicate()
split_data = data[0].split()
local_ip = split_data[split_data.index('src') + 1]
public_ip = os.system('wget http://ipecho.net/plain -o - -q >
test.txt; echo')
public_ip = open('test.txt', 'r').read()
msg_content = 'Alarm detected. Public ip is %s:8081, local ip is
%s' % (public_ip,local_ip)
msg = MIMEText(msg_content)
msg['Subject'] = 'Raspberry Pi Alarm on %s' % today
msg['From'] = gmail_user
msg['To'] = to
smtpserver.sendmail(gmail_user, [to], msg.as_string())
smtpserver.quit()camera.py
```

FIGURE 5.2: Motion conf file

It has to be opened from Firefox with Raspberry pi IP and port Number, which usually it is 8081 This raspberry pi IP address is 192.168.0.18:8081. This is how IP and Port should appear in the Firefox.

A test has been done with no issues

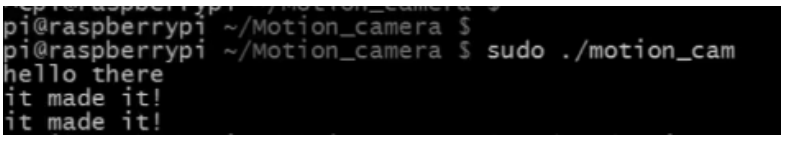


FIGURE 5.3: Test from Lxcommand

5.2 Image comparing

The image comparison can be done in OpenCV in a more professional way, However, due to this projects limitation of SD Card which open CV file almost 16 GB big, and it would take over 12 Hours to upload it to Raspberry Pi, and deploy it on the raspberrypi. During research discovered that many user of raspberry pi community members experienced that OpenCV is slow on raspberry pi module B+, their published youtube tutorials encouraged me to try different methods to find a temporary solution for the time being at least. The alternative and a lighter software simpleCV selected.
Unfortunately, the idea can be implemented, but have to do manually, which means each time images has to be taken manually and uploaded to the software on the raspberry pi and the software will compare the images. How to compare images manually? despite there is not plenty of tutorials, but simplecv is my framework for image processing.
How dose it work? It can be achieved by starting to apply template matching, And there is another practical method comes with simplecv, called simple subtraction, by detecting the pixel of motion detection from the previous image; the method used is the simple subtraction in detecting motion. At that case the user behind the robot has to take a template picture of the area where robot due to be employed, and than take another picture any time needed, or in case the motion detector rising alarms then by help of simpleCV software and some code can pictures be matched to the source image. Matching Templates: The gaol is to detect the highest matching area and this Idea of matching will be developed in the future to comparison between feature of two images and subtract the pixel to find any small detail if have been changed[57]. The Template matching methods can be achieved by applying correlation coefficient matching methods (CV_TM_CCOEFF) These procedures match a template Towards the image relative to its mean, hence, the best. The match is going to be 1 and a perfect mismatch is –1; The value of 0 only ensures that there is absolutely no correlation.

$$R(x,y) = \sum_{x',y'} (T'(x',y') \cdot I(x+x', y+y'))$$

where

$$T'(x',y') = T(x',y') - 1/(w \cdot h) \cdot \sum_{x'',y''} T(x'',y'')$$
$$I'(x+x', y+y') = I(x+x', y+y') - 1/(w \cdot h) \cdot \sum_{x'',y''} I(x+x'', y+y'')$$

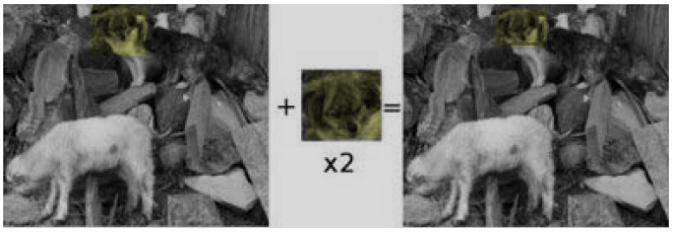


FIGURE 5.4: Test image highst matching area detected

simple subtraction: It will shows the pixels have been changed. This is what surveillance robot about to observe an area and rise alarm whenever any objects move or stolen. The method is if pixel values always zero that indicates no thing have been changed,but if it is not zero it needs attention. in this method user could figure out about 0.8 or 80% alter in pixels. and this change value can use this change as a threshold, for example send an email if the value more that 80% of the pixels change and using threshold can minimize the chance of false positives happening. please see the code below and the images[56].

```
>>> cam = Camera()
>>> prev = cam.getImage()
>>> current = cam.getImage()
>>> diff = current - prev
>>> diff.show()

>>> area = diff.width * diff.height
307200 #this is our image area in pixels
>>> matrix = diff.getNumpy()
>>> matrix.shape
(640, 480, 3)
>>> flat = matrix.flatten()
>>> counter = 0
>>> for i in flat:
        if flat[i] == 0: #if black
                counter += 1

>>> percent_change = float(counter) / float(len(flat))
>>> print percent_change
0.818358289930555
```

FIGURE 5.5

FIGURE 5.6: Image1 origin-image2 moved pixels - Image3 resulte found only moved pixels.

Experiments and Result

The approach described above has been implemented and tested on different robotic sensors for the raspberry pi raspberry pi platforms and EV3 tested in term of processors in different environments, the algorithm correlation coefficient matching methods has been tested. In all experiments, I will figured out, it was able to accurately go through map following line and solving maze algorithm.

Conclusion

In this paper tried to present a unique approaching to vision-based surveillance robot CV. with obstacles avoidance capabilities for general purpose robot in indoor environments. which the all robots should have it. few Algorithms are were implements in this robot. The robot also able to solve maze, and comparison between to images, also has ability to motion detect. The robot uses Three line follow sensors and ultra sonic sensor for obstacle avoidance, and it can be developed the system for swarm.

Future Work

there is more can be done with raspbery pi in the future, it has very powerful ability if an advanced developer handle it. The Robot can go to recharge itself automatically. robot be able to senses the fire smoke. their is so many fields yet for robot, In my opinion we are as a human being there are a distance between to day and the day we will reach the day that satisfied with perfect robot, that AI could update itself and recharge itself with energy that needed. The good thing for the future is that the robot not only sense and guessing, it will calculate and give the accurate results Back.

Bibliography

[1] Online Website. Central static criminal. *Central Static Criminal*, 72, December 2005-2013. URL http://www.cso.ie/multiquicktables/quickTables.aspx?id=cja01.

[2] Benjamin Miessner, Roberto Cordeschi, and Guglielmo Tamburrini. History Makers - Electric Dog 1912. pages 1–4, 2007.

[3] RobotShop Distribution Inc. History of Robotics: Timeline. page 11, 2008. URL http://www.robotshop.com/media/files/PDF/timeline.pdf.

[4] Ben J Challacombe, Mohammad Shamim Khan, Declan Murphy, and Prokar Dasgupta. The history of robotics in urology. *World journal of urology*, 24 (2):120–7, jun 2006. ISSN 0724-4983. doi: 10.1007/s00345-006-0067-1. URL http://www.ncbi.nlm.nih.gov/pubmed/16552597.

[5] Dipankar Das. O BJECT DETECTION FOR SERVICE ROBOT USING. 4(3): 17–29, 2014.

[6] Related Questions, Search Quora, Answer Notifications Halgurd, Shashi Kant, Emergent Semantics Researcher, I B M Watson, Question Stats, and Bharath Hariharan. Qwote What are some of the important problems in the field of AI that need to be solved within the next 5 10 years ? Why did Terence Tao choose. 2016 (7):2–3, 2016.

[7] Computer Science. Search & Rescue using Multi-Robot Systems. pages 1–14.

[8] Oussama khatib(Eds.) Bruno Siciliano. Eng. rami khalil. page 1629. 2008. ISBN 9781852339944. URL http://www.springer.com/gp/book/9783540239574.

[9] M. Gibbs. Ten reasons why the new raspberry pi 2 model b sbc is a killer product. *online- Network World*, Feb 22, 2015. URL http://www.networkworld.com/article/2887224/tech-primers/10-reasons-why-the-raspberry-pi-2-model-b-is-a-killer-product.html.

[10] Whitequark.org. Why raspberry pi is unsuitable for education - whitespace. *online*, 2012. URL http://whitequark.org/blog/2012/09/25/why-raspberry-pi-is-unsuitable-for-educatio.

[11] Bjorn B. Brandenburg and Shinpei Kato. the 10th annual workshop on operating systems platforms for embedded real-time applications. *the 26th Euromicro Conference on Real-Time Systems July 9–11, 2014, Madrid, Spain*, July 9–11, 2014. URL https://www.mpi-sws.org/~bbb/proceedings/ospert14-proceedings.pdf.

[12] v3dev.org. Ev3dev (2015). ev3dev home. [online]. *Website*, 2015. URL http://www.ev3dev.org/.

[13] dexter industries. Brickpi. 69. URL http://www.instructables.com/id/BrickPiSetup/.

[14] smashingrobotics. Brickpi. URL http://www.smashingrobotics.com/brickpi-control-lego-mindstorms-robots-raspberry-pi/.

[15] F. C. Jacopo Prisco. Will nanotech soon allow you to 'swallow the doctor'? - cnn.com. *online*, 2015. URL http://edition.cnn.com/2015/01/29/tech/mci-nanobots-eth/.

[16] Mike McRoberts. Beginners Guide to the Arduino Earthshine Design A Complete Beginners guide to the Arduino. *Legal Theory*, 2009. URL http://www.earthshinedesign.co.uk/.

[17] Online/Website. Arduino vs Raspberry Pi vs BeagleBone vs PCDuino — Random Nerd Tutorials. URL http://randomnerdtutorials.com/arduino-vs-raspberry-pi-vs-beaglebone-vs-pcduino/.

[18] Gus. Banana Pi Vs Raspberry Pi 2 : The Battle of the 2 Fruits. Comparasio:2–7. URL http://pimylifeup.com/banana-pi-vs-raspberry-pi-2/.

[19] By Jeff. PiLarm : Portable Raspberry Pi Room Alarm. *Website*, pages 1–3. URL http://makezine.com/projects/pilarm-portable-raspberry-pi-room-alarm/.

[20] Project Name, Project Sponsor, Peer Baneke, Project Manager, Ayesha Ali Version, Project Rationale, and Young Voices. Project details. (November 2012):1–33, 2013. URL https://hackaday.io/post/7360.

[21] Rekatu the Researcher. Maplin products. 2016. URL http://www.maplin.co.uk/c/cctv-and-security.

[22] Yihan Shao, Laszlo Fusti Molnar, Yousung Jung, Jörg Kussmann, Christian Ochsenfeld, Shawn T Brown, Andrew TB Gilbert, Lyudmila V Slipchenko, Sergey V Levchenko, Darragh P O'Neill, et al. Advances in methods and algorithms in a modern quantum chemistry program package. *Physical Chemistry Chemical Physics*, 8 (27):3172–3191, 2006.

[23] Robotics/Types of Robots - Wikibooks, open books for an open world, 2015. URL https://en.wikibooks.org/wiki/Robotics/Types{_}of{_}Robots.

[24] Oussama khatib(Eds.) Bruno Siciliano. *Eng. rami khalil.* 2008. ISBN 9781852339944.

[25] Michael Gasperi's LEGO Mindstorms. Michael Gasperi's LEGO Mindstorms NXT/RCX Sensor Input Page. URL http://www.extremenxt.com/lego.htm.

[26] General Search and Conference Information. User Guide. 2007. URL http://legoeducation.com.

[27] INC. EMX, Indutries. Color Sensor Products From EMX Inc., 2016. URL http://www.emxinc.com/colormax.html?gclid=CjwKEAjw0pa5BRCLmoKIx{_}HTh1wSJABk5F{_}4-GaFrUygx0ORmKf9BFUUtBWSyXnUcMyw6b-ffN3sS

[28] 'LEGO.com Education 45502 - Large Servo Motor', LEGO.com Education 45502 - Large Servo Motor, 2016 <https://education.lego.com/en-au/lego-education-product-database/mindstorms-ev3/45502-l-motor>LEGO.com Education 45502 - Large Servo Motor. *LEGO.com Education 45502 - Large Servo Motor*, 2016. URL https://education.lego.com/en-au/lego-education-product-database/mindstorms-ev3/45502-l-motor.

[29] Website. Raspberry pi 2 model b. 69. URL http://www.raspberrypi.org/products/raspberry-pi-2-model-b/.

[30] bushsteve. Raspberry pi 2 model b - more detail. *Electronics Weekly*, 21 Nov 2015. URL http://readwrite.com/2013/05/09/how-an-open-source-operating-system-jumpstarted-robotics-research.

[31] Trendnet.com. Raspberry pi 2 model b - more detail. *Website*, November 22 2015. URL https://www.trendnet.com/?todo=home.

[32] En.wikibooks.org. Robotics/components/power sources - wikibooks, open books for an open world. *Website*, November 22 2015. URL https://en.wikibooks.org/wiki/Robotics/Components/Power_Sources.

[33] Raspberry Pi NOOBS Setup. URL https://www.raspberrypi.org/help/noobs-setup/.

[34] Windows Os, Windows Pc, Command Line Interface, Windows Pcs, So Putty, Unix Guis, Unix Guis, and Linux Vm. Setting up Xming and Putty on your Windows laptop.

[35] TitleDexterInd/BrickPi_Python. URL https://github.com/DexterInd/BrickPi{_}Python.

[36] LAUREN ORSINI. How an open source operating system jumpstarted robotics research. *Online Source*, MAY 9, 2013. URL http://readwrite.com/2013/05/09/how-an-open-source-operating-system-jumpstarted-robotics-research.

[37] Laura Cassell and Alan Gauld. *Python Projects*. Wiley, 2014. ISBN 1118909194. URL https://books.google.com/books?id=Cty6BQAAQBAJ{&}pgis=1.

[38] D.B.Gennery. Traversability analysis and path planning for planetary rovers", autonomous robots. *PDF Book*, 1999. URL Kluwer,AcademicPublishers.

[39] The International Journal. the tenth internationalworkshop on algorithmicfoundations of robotics. *Journal*, 2013. URL www.sagepub.co.uk/journalsPermissions.nav.

[40] kristo@math.ut.ee jan@math.ut.ee alvo@ut.ee maarjakr@ut.ee. Learning method for mobile robot navigation in partially unknown environments. *PDF Journal*, —-.

[41] Maarja Kruusmaa Jan Willemson. Algorithmic generation of path fragment coversfor mobile robot path planning. *PDF File*, 2003.

[42] S. M. Masudur Rahman Al-ArifA. H. M. Iftekharul FerdousSohrab Hassan Nijami. omparative study of different path planning algorithms: A water based rescue system arabic. *International Journal of Computer Applications*, Volume 39 - Number 5, 2012.

[43] Mongi A. Abidi Brad Grinstead, Andreas Koschan. Hybrid self localization for a mobile robotic platform for indoor and outdoor environments. *PDF,The University of Tennessee 334 Ferris Hall 1508 Middle*, —-.

[44] tsukiyama@sie.dendai.ac.jp. Rfid based navigation system forin door mobile robots. *PDF*.

[45] K.a. Sikorski a. Borkowski M. Weigl, B. Siemiaatkowska. Gridbased mapping for autonomous mobile robot. *Sciencedirect.com (2015)*, 16 November 1992. URL http://www.sciencedirect.com/science/article/pii/092188909390004V.

[46] C. O'Dunlaing and C. Yap. A retraction method for planning the motion of a disc. *Journal of Algorithms*, vol. 6:pages 104–111, 1982.

URL https://www.researchgate.net/publication/245040768_A_retraction_method_for_planning_the_motion_of_a_disc.

[47] Ian Stllitoe Yang Wang, David Mulvaney. Genetic-based mobile robot path planning using vertex heuristics. *2006 IEEE Conference on Cybernetics and Intelligent Systems*, 2006. URL https://www.researchgate.net/publication/224673288_Genetic-based_Mobile_Robot_Path_Planning_using_Vertex_Heuristics.

[48] Light Sensors. Lego Robot Tutorials Follow a Line. pages 1–7, 2003.

[49] J. Borenstein, H. R. Everett, L. Feng, and D. Wehe. Mobile robot positioning: Sensors and techniques. *Journal of Robotic Systems*, 14(4):231–249, 1997. ISSN 07412223.

[50] GPS or GNSS signal repeaters — Ofcom. URL http://stakeholders.ofcom.org.uk/enforcement/spectrum-enforcement/gpsrepeaters/.

[51] I-Hsum Li, Ming-Chang Chen, Wei-Yen Wang, Shun-Feng Su, and To-Wen Lai. Mobile robot self-localization system using single webcam distance measurement technology in indoor environments. *Sensors (Basel, Switzerland)*, 14(2):2089–109, jan 2014. ISSN 1424-8220. doi: 10.3390/s140202089. URL http://www.mdpi.com/1424-8220/14/2/2089/htm.

[52] Joachim Hertzberg and Raja Chatila. AI Reasoning Methods for Robotics. *Springer Handbook of Robotics*, pages 207–223, 2008. doi: 10.1007/978-3-540-30301-5_10. URL http://dx.doi.org/10.1007/978-3-540-30301-5{_}10$\delimiter"026E30F$nhttp://link.springer.com/static-content/0.5480/pdf/297/chp:10.1007/978-3-540-30301-5{_}10.pdf?token=1350544024456--eab43db8747278a681f490ceae92a6f33f2ea39eee350d873953f03ccfa233ad95

[53] R Murphy. *AI Robotics*. 2000. ISBN 0262133830. URL http://scholar.google.com/scholar?hl=en{&}btnG=Search{&}q=intitle:Ai+robotics{#}5$\delimiter"026E30F$nhttp://scholar.google.com/scholar?hl=en{&}btnG=Search{&}q=intitle:AI+Robotics{#}5.

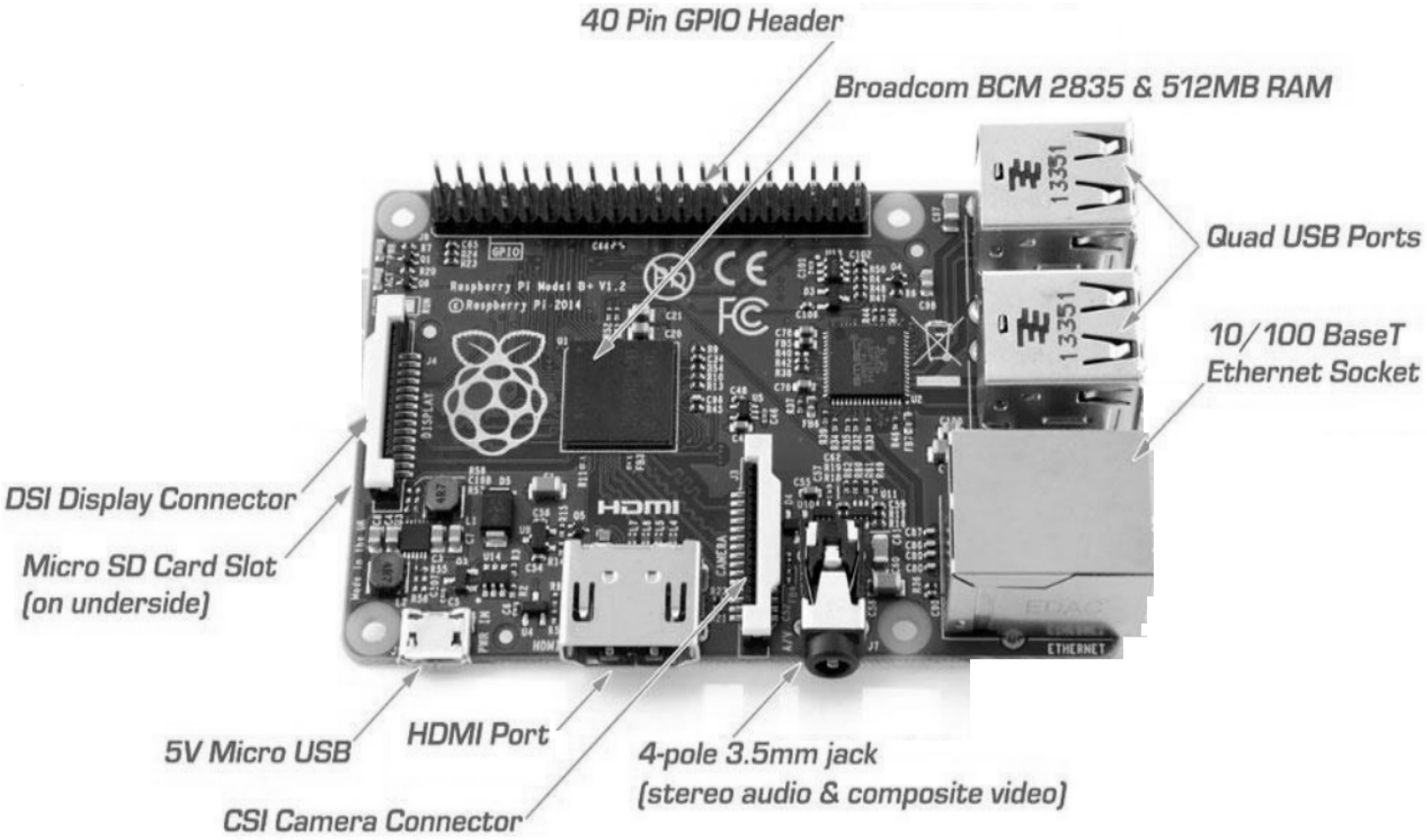
[54] José Antonio and Álvarez Ruiz. Master Thesis Active Scene Text Recognition for a Domestic Service Robot. *RoboCup 2012: Robot Soccer World Cup XVI*, page 75, 2011.

[55] D Venkata Vara, Prasad Me, and Chitra Devi J. Knowledge based Reinforcement Learning Robot in Maze Environment. 14(7):22–30, 2011.

[56] Sundas Zafar and Computer Engineering Technology. Notifications and Live Streaming Using Raspberry Pi.

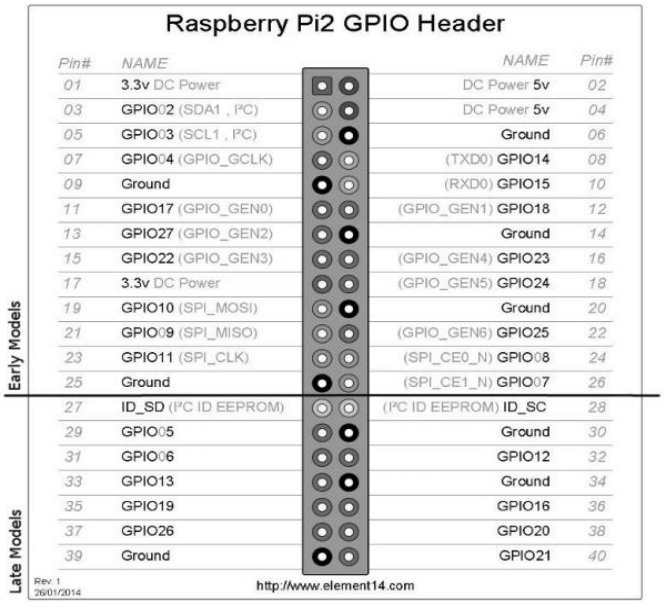
[57] Cuauhtemoc Carbajal. Computer Vision Using SimpleCV and the Raspberry Pi. page 142, 2012.

Starting to Build our Robot



Accessing GPIO Pins

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BCM)
GPIO.setup(23, GPIO.IN)
GPIO.setup(24, GPIO.OUT)
```



Python on Raspberry PI

- IDLE and Python Come pre-installed on Raspbian
- We will use our favourite editor to write python scripts

Configuring VNC

• Install VNC viewer from https://www.realvnc.com/ on your MAC or
Windows

• Raspbian already comes pre-installed with VNC

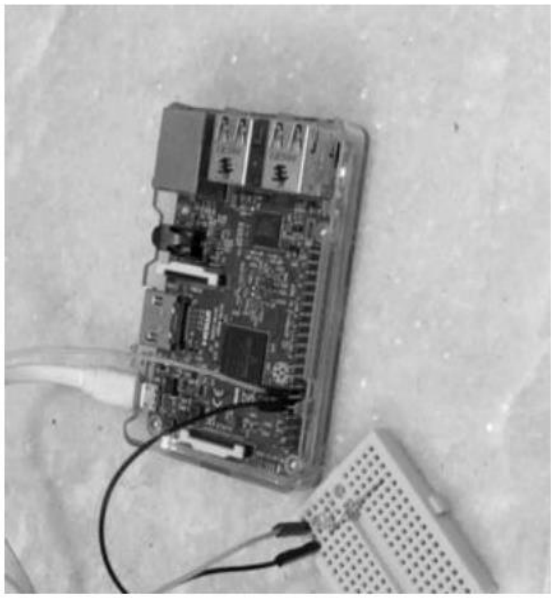
• You can connect to your PI over the same network or over the
internet

• Need to create an account with VNC

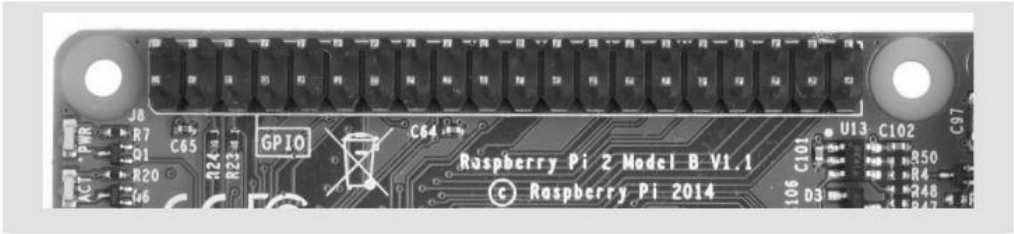
• Follow https://www.realvnc.com/en/connect/docs/raspberrypi.html#raspberry-pi-connect-cloud to connect to Raspberry

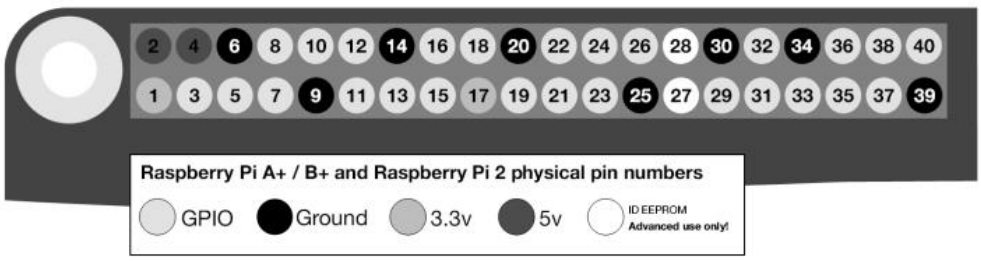
Task 1: Hello World

- Blinking an LED
  - Connect an LED to IO Port of Raspberry PI
  - Make it Blink
  - We will use pin 11 (GPIO 17)
  - And pin 9 (Gnd)

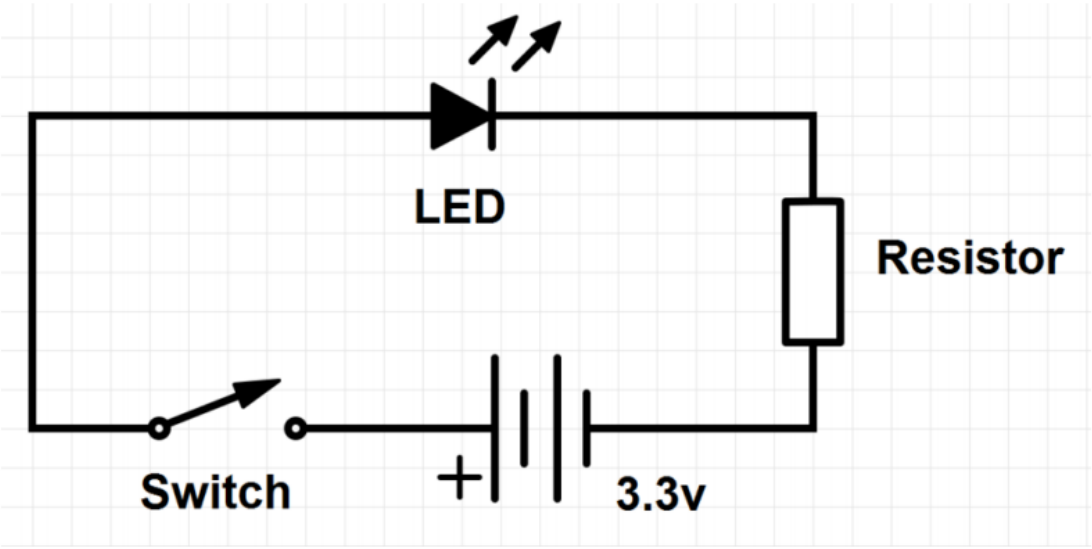


GPIO Pins





LED Circuit



What's in the Code?

```
#Blinking an LED
import RPi.GPIO as GPIO
import time

LedPin = 11    # pin11

def setup():
  GPIO.setmode(GPIO.BOARD)       # Numbers GPIOs by physical location
  GPIO.setup(LedPin, GPIO.OUT)   # Set LedPin's mode is output
  GPIO.output(LedPin, GPIO.HIGH) # Set LedPin high(+3.3V) to turn on led
```

BLINK : HELLO WORLD!

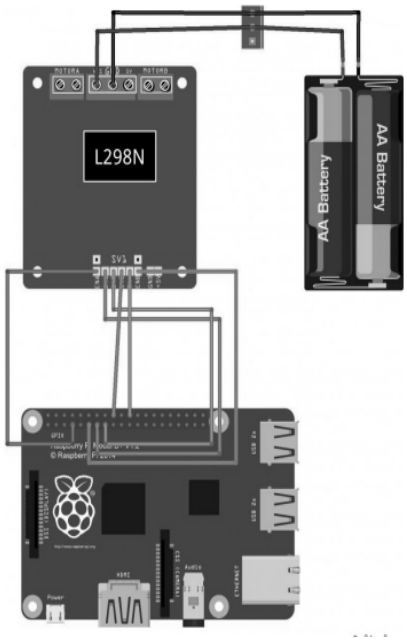
```
def blink():
  while True:
    GPIO.output(LedPin, GPIO.HIGH)  # led on
    time.sleep(0.2)
    GPIO.output(LedPin, GPIO.LOW) # led off
    time.sleep(0.2)

def destroy():
  GPIO.output(LedPin, GPIO.LOW)   # led off
  GPIO.cleanup()                  # Release resource
```

Main Function

```
if __name__ == '__main__':     # Program start from here
  setup()
  try:
    blink()
  except KeyboardInterrupt:  # When 'Ctrl+C' is pressed,
    destroy()
```

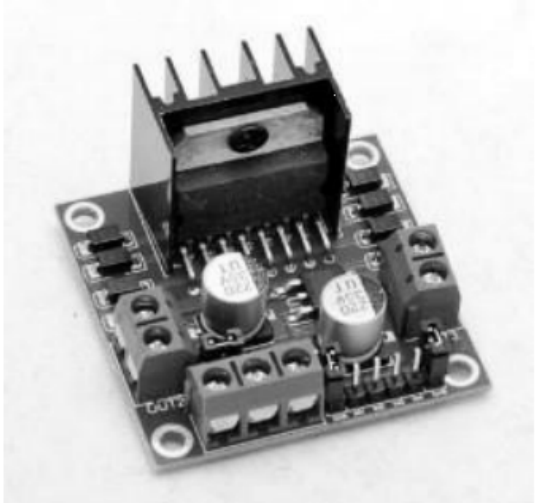
Task 2: Interfacing Motors

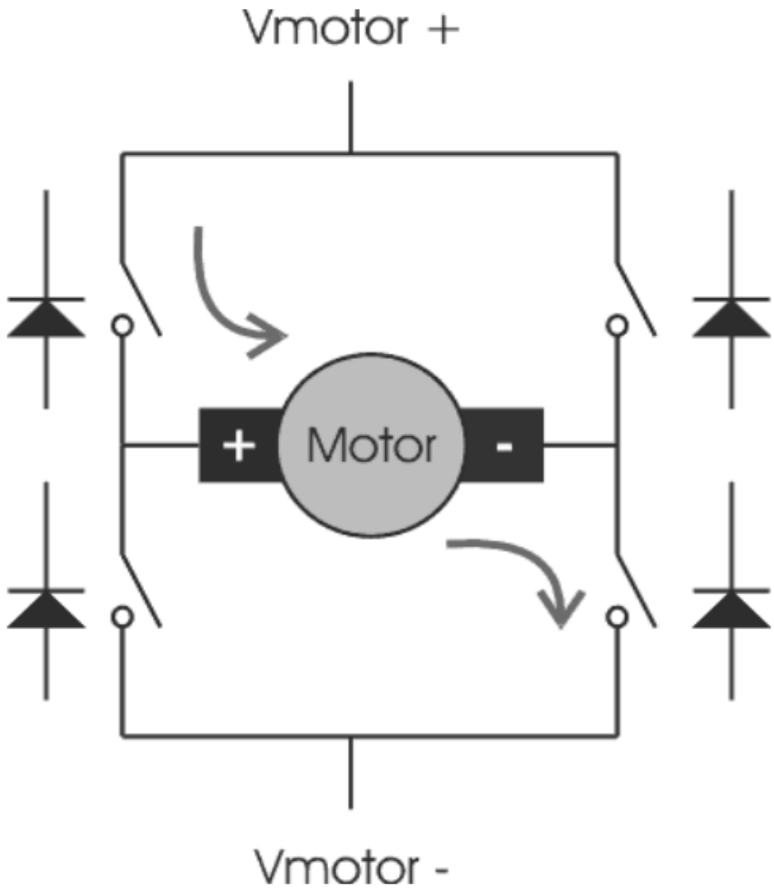
- Using CodiBot we will be using an L298 Hbridge to control the 2 motors for robot movement
- Moving Codi Forwards
- Moving Codi Backwards
- Turning Codi using Differential Drive



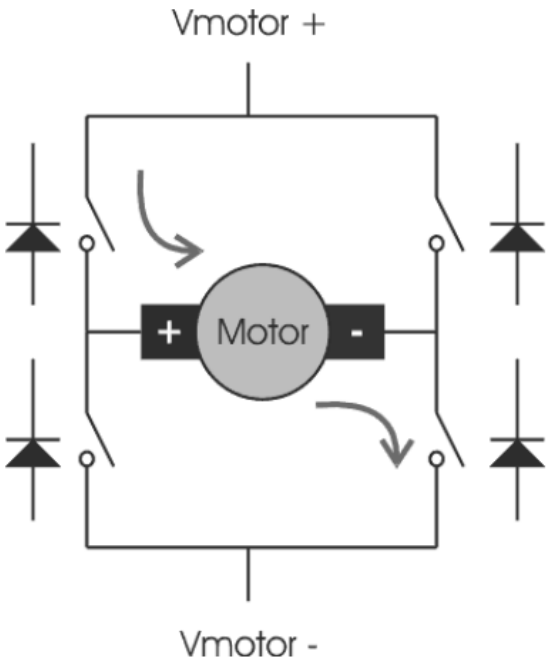
H-Bridge

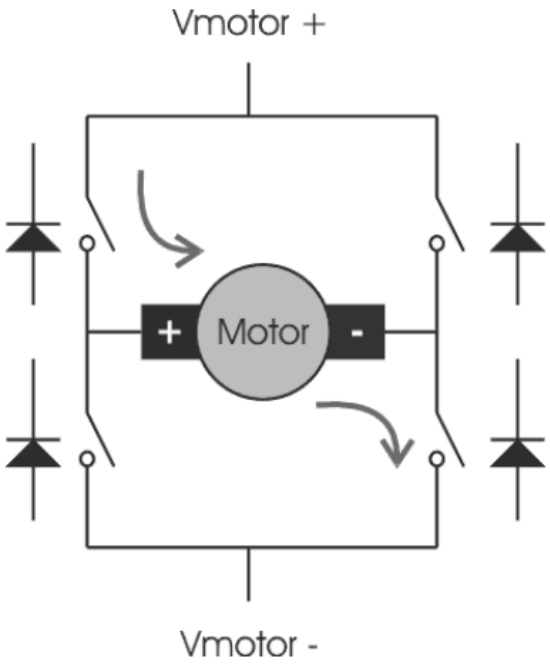
- Changing Directions





Differential Drive






```
# Import required libraries
import sys
import time
import RPi.GPIO as GPIO

GPIO.cleanup()

# Define GPIO signals to use
# Physical pins 29,31,33,35,37,40

RightMotor = 29
RM1a = 31
RM1b = 33

LeftMotor = 40
LM1a = 35
LM1b = 37

sleeptime=1
```

```
 def setup():
     GPIO.setmode(GPIO.BOARD)
     GPIO.setup(RightMotor, GPIO.OUT)
     GPIO.setup(LeftMotor, GPIO.OUT)

     GPIO.setup(RM1a, GPIO.OUT)
     GPIO.setup(RM1b, GPIO.OUT)
     GPIO.setup(LM1a, GPIO.OUT)
     GPIO.setup(LM1b, GPIO.OUT)
```

```
def forward(x):
    #H-Bridge Pin Settings
    GPIO.output(RM1a, GPIO.HIGH)
    GPIO.output(RM1b, GPIO.LOW)
    GPIO.output(LM1a, GPIO.HIGH)
    GPIO.output(LM1b, GPIO.LOW)

    #Turning Motors ON
    GPIO.output(RightMotor, GPIO.HIGH)
    GPIO.output(LeftMotor, GPIO.HIGH)
    print ("Moving Forward")
    time.sleep(x)
    GPIO.output(RightMotor, GPIO.LOW)
    GPIO.output(LeftMotor, GPIO.LOW)
```

```
def reverse(x):
    #H-Bridge Pin Settings
    GPIO.output(RM1a, GPIO.LOW)
    GPIO.output(RM1b, GPIO.HIGH)
    GPIO.output(LM1a, GPIO.LOW)
    GPIO.output(LM1b, GPIO.HIGH)

    #Turning Motors On
    GPIO.output(RightMotor, GPIO.HIGH)
    GPIO.output(LeftMotor, GPIO.HIGH)

    print ("backwarding running motor")
    time.sleep(x)

    GPIO.output(RightMotor, GPIO.LOW)
    GPIO.output(LeftMotor, GPIO.LOW)


def dance(x):
    #H-Bridge Pin Settings
    GPIO.output(RM1a, GPIO.LOW)
    GPIO.output(RM1b, GPIO.HIGH)
    GPIO.output(LM1a, GPIO.HIGH)
    GPIO.output(LM1b, GPIO.LOW)

    #Turning Motors On
    GPIO.output(RightMotor, GPIO.HIGH)
    GPIO.output(LeftMotor, GPIO.HIGH)

    print ("Dancing!")
    time.sleep(x)

    GPIO.output(RightMotor, GPIO.LOW)
    GPIO.output(LeftMotor, GPIO.LOW)

def destroy():
    print ("Stopping motor")
    GPIO.cleanup()
```

```
if __name__ == '__main__':     # Program start from here
  setup()
  try:
    reverse(5)
  except :  # When 'Ctrl+C' is pressed, the child program destroy() will
be  executed.
    destroy()

  time.sleep(1)
  try:
     forward(5)
  except:
     destroy()

  #time.sleep(1)
  #try:
  #    dance(5)
  #except:
  #    destroy()

  destroy()
```

```
#Blinking an LED
import RPi.GPIO as GPIO
import time

LedPin = 11    # pin11

def setup():
  GPIO.setmode(GPIO.BOARD)       # Numbers GPIOs by
physical location
  GPIO.setup(LedPin, GPIO.OUT)   # Set LedPin's mode is
output
  GPIO.output(LedPin, GPIO.HIGH) # Set LedPin high(+3.3V) to
turn on led

def blink():
  while True:
    GPIO.output(LedPin, GPIO.HIGH)  # led on
    time.sleep(0.2)
    GPIO.output(LedPin, GPIO.LOW) # led off
    time.sleep(0.2)

def destroy():
  GPIO.output(LedPin, GPIO.LOW)   # led off
  GPIO.cleanup()                  # Release resource

if __name__ == '__main__':     # Program start from here
  setup()
  try:
    blink()
  except KeyboardInterrupt:  # When 'Ctrl+C' is pressed, the child
program destroy() will be  executed.
    destroy()
```

```
#Sonar interface with the Raspberry PI

#Import Python libraries
import time
import RPi.GPIO as GPIO


GPIO.setmode(GPIO.BOARD)

GPIO_TRIGGER = 29    #GPIO_24
GPIO_ECHO = 40       #GPIO_25

#Set Pins as output and input
GPIO.setup(GPIO_TRIGGER,GPIO.OUT) # Trigger
GPIO.setup(GPIO_ECHO,GPIO.IN) # Echo

#Set Trigger low
GPIO.output(GPIO_TRIGGER, False)

#Allow module to settle
time.sleep(0.5)
```

```
def sonar():
    #Send 10us pulse to trigger
    GPIO.output(GPIO_TRIGGER, True)
    time.sleep(0.00001)
    GPIO.output(GPIO_TRIGGER, False)

    while GPIO.input(GPIO_ECHO)==0:
        start = time.time()

    while GPIO.input(GPIO_ECHO)==1:
        stop = time.time()

    #Calculate pulse length
    elapsed = stop-start

    #Distance pulse traveled in that time is time
    #multiplied by the speed of sound (cm/s)
    distance = elapsed * 34000

    #That was the total distance so half it for reaching the object
    distance = distance /2

    return distance

while True:
    time.sleep(0.3)

    distance = sonar()
    print (distance)
```