MOBI PARKING NAVIGATOR SYSTEM

(SENSOR MONITORING)

SHAROS AZUANUDDIN BIN MAT SIDEK

THESIS SUBMITTED IN FULFILMENT OF THE DEGREE OF BACHELOR OF COMPUTER SCIENCE (COMPUTER SYSTEMS & NETWORKS) WITH HONOURS

FACULTY OF COMPUTER SYSTEMS AND SOFTWARE ENGINEERING

2013

ABSTRACT

The vehicles in UMP Gambang Campus keep increasing; problem to get parking spaces is very difficult especially during peak hours. That is also a general problem that occurs in the parking spaces in front of FSKKP building; staff and guest have to circling around the parking spaces a few times to find parking. Mobi Parking Navigator System (Sensor Monitoring Module) is proposed and will be implementing at FSKKP office parking spaces. This system consists of three different modules which are Sensors module (Infrared sensor), Electronic module (Raspberry pi) and System module (Web based system). All the modules will be combining together and build this system full functioning. A system that using an infrared sensor been developed, this system will detect the free parking spaces with infrared sensor and save it to database. Users that have Mobi Parking Navigator System Apps can get information about the parking from this smartphone application. With this system trouble in finding parking spaces will become easier, less cost and less frustration to the driver.

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Overview

In global world today, transportation is very important for everyone. Transportation usually used in working, to go somewhere, and to make easy human life. Because of this matter, parking lot space has become a very huge problem; it becomes needs that are very important to avoid wasting space in the modern era. Drivers often becomes frustrated when arrive at the shopping complex during weekends day. Some might even around the parking lot several times trying to find space to park their cars. They all face the same problem and have to spend more time to park their vehicle especially during the peak hours.

Infrared sensors are electronic devices that emit and detect infrared radiation to detect some aspects of its surroundings. Thermal infrared sensors can measure objects, heats of object, and detect movement. An infrared sensor is also called as an IR sensor is widely used in application involving object detection and avoidance.

Electronic devices that can detect the physical movement in the given area and can transform motion into an electric signal, it connected to a devices or indicator that will showed the resulting object that detected. Infrared sensors have been used in a wide variety of applications and it is mainly used in transportation area.

Nowadays the application of microcontroller is widely used for consumer. Microcontroller is a board functional computer system on a small chip. Microcontroller board contains a processor core, a memory and a programmable input/output peripheral. Its emphasize high integration, unlike a microprocessor which only contains a CPU. Adding up to the usual arithmetic and logic elements of a microprocessor, microcontrollers is include an integrated CPU, a memory which is a small amount of RAM, program memory or both of it and a peripherals as capable of input and output.

Microcontroller board operates at relatively very low speed and power compared to microprocessors, but this sufficient for typical applications. Microcontroller

consumes relatively a little power and will generally have an ability to maintain the functionality while waiting for a process such as button press or the interrupt. Microcontrollers are used in automatically controlled products and devices such as automobile engine control systems, remote controls and appliances. By reducing the size, cost, and power consumption compared to a design using a separate microprocessor, memory, and input and output devices, microcontrollers make it economical to electronically control many more processes. Operations of a microcontroller are difficult to be demonstrated without an application.

The infrared sensor parking is designed by combining several equipment which are the infrared sensor, various types of IC, resistor, capacitor, diode and transistor. The infrared transducer works as a sensor to interpret the echoes from the objects or human. While the RF transmitter and receivers are the devices that transmit and receive the signal from the transducer. This system save the data detected by receiving information from the sensor that detects the object and it is very useful to detect object even though it will be in a darker environment.

Monitoring System is a main part in MPNS, it is a system that allows administrator to monitoring real time detecting activity in the parking and also can manage sensor and parking information. Monitoring system is developed based on hardware and software component that consist of three different modules which are Sensors module (Infrared sensor), Electronic module (Raspberry pi) and System module (Web based system). The infrared sensors sent the data of detecting to the raspberry pi and store it in database, the web based system will manage the data and display it.

1.2 Problem Statement

The vehicles in UMP Gambang Campus keep increasing; problem to get parking spaces is very difficult especially during peak hours. That is also a general problem that occurs in the parking spaces in front of FSKKP building; staff and guest have to circling around the parking spaces a few times to find parking.

There is no system or tool develops to solve this parking spaces problem, the parking does not have a systematic system or tool to overcome the problem. The parking also cannot be managing.

1.3 Motivation

The vehicles in UMP Gambang Campus keep increasing; problem to get parking spaces is very difficult especially during peak hours. That is also a general problem that occurs in the parking spaces in front of FSKKP building.

To solve this problem a system that using an infrared sensor been developed, this system will detect the free parking spaces with infrared sensor and save it to database. User with Mobi Parking Navigator System Apps, can get information about parking from this smartphone application.

With this system trouble in finding free parking spaces will become easier, less cost and less frustration. And with this system time can be save more and can optimize the use of smartphones and Wi-Fi available.

1.4 Project Objective

The main objective of the project is to develop an application and a system that made easy to the people to find empty parking.

The objectives of this project are:

i. To identify software and hardware to detect whatever have an empty parking or not by using infrared sensor.

ii. To design database and system interface for managing and monitoring the parking space.

iii. To develop system between software and hardware to receive, save and update the data that received from infrared sensor.

1.5 Project Scope

The general scopes of this project are:

i. Staff of FSKKP, UMP Gambang.
ii. Guest that come to FSKKP office building.
iii. Mainly focus at parking spaces in front of FSKKP office building.

The specific scopes for this system are:

i. Administrator:
   Use to monitor and manage the parking information..

Device used in this project:

i. Computer
   - A computer that used to develop code and system.

ii. Adjustable Infrared Sensor (SN-E18-D80NK)
   - A detection sensor that will located at the front or below of the parking spaces. It is enables to detect an object that in front or below of it and can returned signal to the receiver.

iii. Raspberry Pi (Model B)
   - A microcontroller that allows receiving a signal from the sensor and then converting it to computer understandable language to store in the database.

Tools used in create codes, build interfaces and database:

1. Notepad++
2. XAMPP

1.6 Methodology

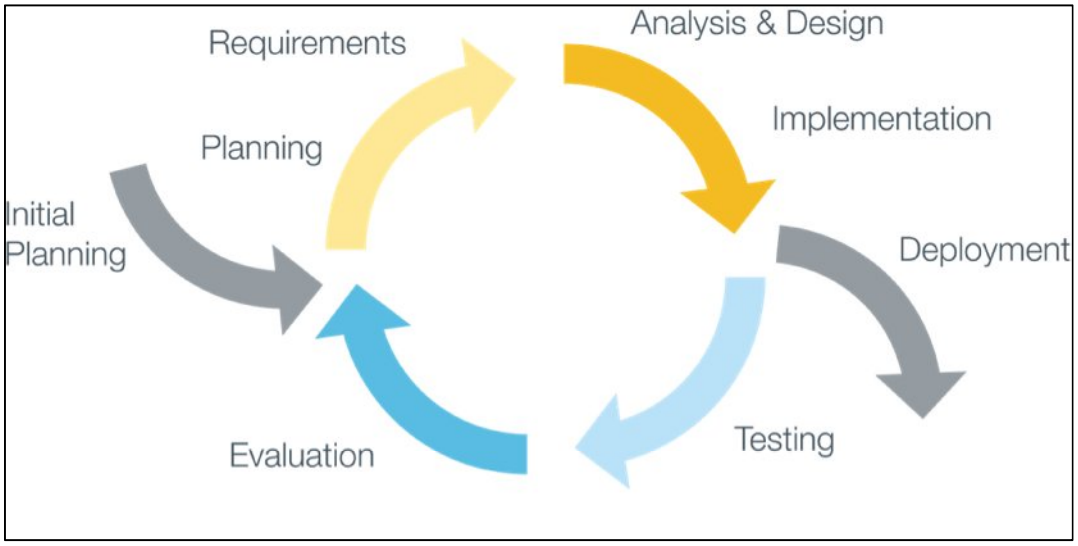

Figure 1.6.1 – Iterative Development Model.

For this project, Iterative Development Model has been chosen as the methodology to develop the system. In this methodology have eight phases that need to be completed in order to complete the project. First phase is initial planning, where all possible requirement of the system that wants to developing will be captured in this phase.

Second phase is planning, in this phase the introduction has been done and timeline for the project has been making. The general research on finding the information about the problem is done in order to get more knowledge about the problem. The problems need to be analyzed and understood because a solution must be created.

Third phase is requirement phase, all the information like the flow and the concept of the software and hardware of the system will be gathers here.

Fourth phase is analysis and design phase, where the development of the project will begin here. All the tools and coding will be use and make in this phase. A complete literature review of hardware and software has been identified and analyze. The concepts of system and sensor work are important to make sure that the device will work properly and can detect the obstacles. Other than that, the suitable type of microcontroller needed to be selected and understand to maximize the usage of the microcontroller.

Fifth phase is implementation phase, in this phase all software and hardware that is suitable will be choosing in making the project. The design of system interface for monitoring system, database design and prototype design will be make in this phase. A design of context diagram and data flow diagram for the project also has been made in this phase.

Six phases is testing phase, upon the completion of implementation the testing will be run on the system to test the system, board and sensor functionality. Any error or not function part will be fixing in this phase.

Seven phases is evaluation phase, in this phase the system will be evaluated by developer and administrator to see their feedback about the system. If there has a problem while implement and testing the system, the whole process will be started again.

Eight phases is deployment phase, in this last phase a complete full functioning system has been developed.

1.7 Theoretical Framework

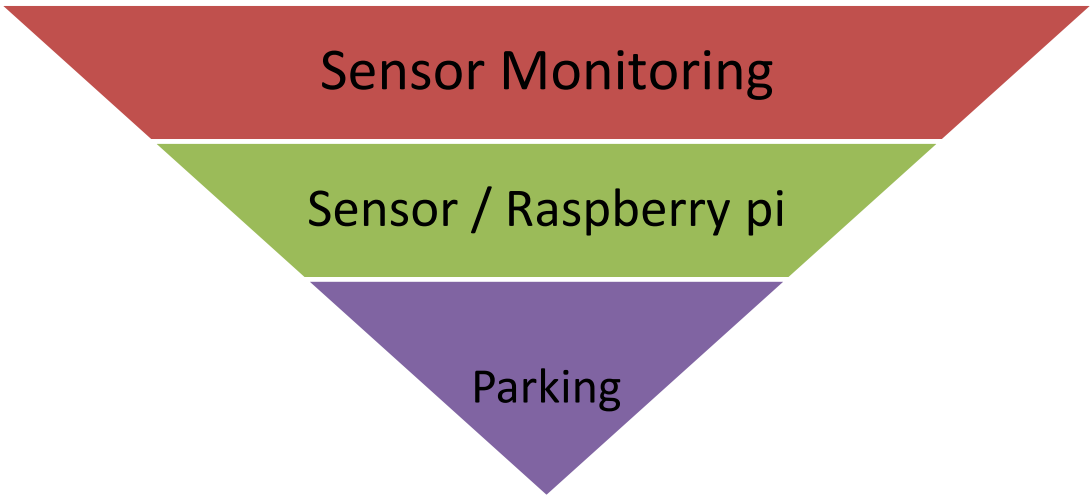

Figure 1.7.1 – Part of theoretical framework.

In the first part of theoretical framework is parking, parking is an act or practice of temporarily leaving a vehicle or manoeuvring a vehicle into a certain location. There are 2 type of parking available that is on street parking and off street parking.

The second part of theoretical framework is sensor, sensor also known as detector that use as convertor to measure the physical quantity and sent to microcontroller to converts it into signal that can be understand by a device or a computer. And another part is raspberry pi, that is also known as a mini computer with the size of card credit and have same functionality as a real computer.

The third part of theoretical framework is sensor monitoring system, sensor monitoring is the system in computer that do all the functionalities like monitoring, collection of data and transmission of data to the user. The more explaining about parking, sensor and sensor monitoring will be cover in chapter 2.

1.8 Thesis Organization

This thesis consists of six chapters. Chapter 1 will be discussing about the introduction of the system, problem statements, objectives and scope of the project.

Chapter 2 is a literature review that discuss about the existing system and hardware that are used in that existing system.

Chapter 3 is a proposed work that will discuss about the design of the system and what hardware and software will be used.

Chapter 4 is an implementations part that will discuss about the compilation of the system, hardware, tool and the system itself. The process in developing the system will be explained in this chapter.

Chapter 5 is result and discussion. The elements have in this chapter is include result analysis and future works. To see thesis organization chart refer to appendix A.

Chapter 6 is the conclusion of the project. This last chapter will make a conclusion of the entire part of the project.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Literature review part is a study that has been done related to the equipment and other needs that are used during the development process of sensor monitoring system was done. Functions available in the project has been reviewed and clarified in each of its functionality.

Study done on the development of the project has also been described in as much as possible, especially in parts of the main concerns in the development of projects such as the determination of the main components of the system that have been developed.

Parking Monitoring System with Infrared Sensor Features is design for in or outdoor parking spaces. The main part of this project is sensor. It is about how to manipulate infrared sensors as an indicator to detect the car in the parking monitoring system.

This system consists of three difference modules which are Sensors module (Infrared sensor), Electronic module (Microcontroller) and Software module (Sensor monitoring system). All the modules will be combining together and build this system full functioning. Each module carries own functioning and special features which will be discussed in detail in this chapter.

In completing the project, a literature reviews have been made on several resources. The theory, design and description plus the details about the project have taken as guidance in completing the project. By this chapter, an overview of the journal and some system that similar to the project and related project design is present.

2.2 Parking

Parking by generally is mean action of stopping a vehicle and leaving it unoccupied, parking is very important to the driver to park their vehicles safely, fast and systematically. Parking spaces lot can be in a parking garage, in parking lot or on a city street. [1]

Parking usually designated by a white paint on tar rectangle indicated by three lines at the top, left and right of the designated area. The vehicles should fits inside the space, either by parallel parking, perpendicular parking or an angled parking. There are 2 types of parking spaces there are on-street parking and another one is off-street parking.

2.2.1 On-Street Parking

On-street parking means the vehicles are parked on the sides of the street itself. This will be usually controlled by a government agency itself. Common types of on-street parking are parallel parking and 90 degree parking. [2]

2.2.2 Off-Street Parking

Off-street parking means the parking is away from main stream of traffic. In many urban centres, some are exclusively allotted for parking by either public agencies or private firms. The common types of off-street parking are garages, self-parking or attendant-parking. [2]

2.3 Sensor Monitoring

A monitoring system that will have a graphical interface where people can understand and can observe the environment of the monitoring, sensors will connect with microcontroller and microcontroller will connect with computer.

The infrared sensors detection any physical object appearing in front of it, translating physical world attributes into values by microcontroller and sent the data to the computer. The system in the computer will do all the functionalities like monitoring, collection of data, and transmission of data to the user.

2.3.1 NanoMon: An Adaptable Sensor Network Monitoring Software

NanoMon is a software use for monitoring network sensor. It can enables users to use and interpret sensor network status visually by using providing decent of visualization component for sensor network monitoring like topology, sensor data, sensor list, node status, received packet log and other.[3]

NanoMon can also support and manage multiple sets of sensor network information like a history of receiving sensing data from sensor and it can store sensor data that related with. It also provides a plug-in capability for adding or removing custom graphical user interface component at needed.

The framework for sensor network monitoring is showing in figure 2.3.1.1. From the figure it shows 4 part of NanoMon that is wireless sensor networks, monitoring server, databases and monitoring clients.

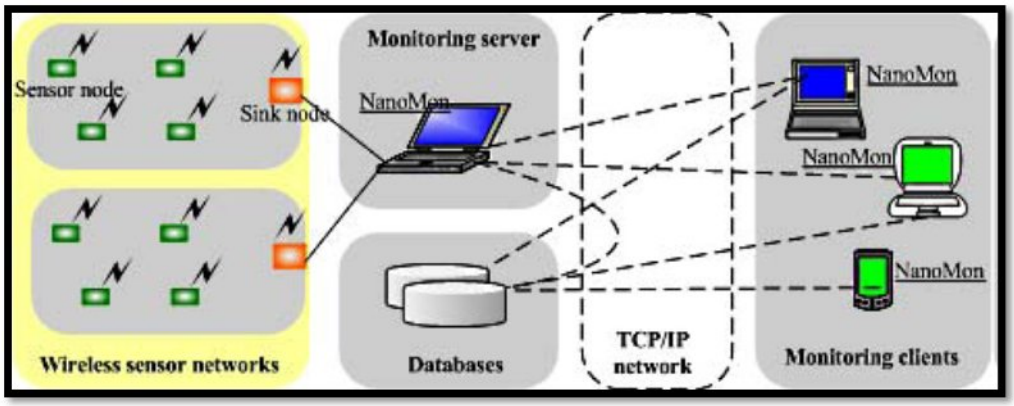

Figure 2.3.1.1 – Sensor network monitoring framework of NanoMon.

This software includes micro-sensor nodes that loading sensor OS, sensor applications, routing facilities, and sink node. Information will generated by a sensor nodes that forwarded to a designated sink node. Each of sink nodes sends the collected data and topology information to the monitoring server.

After monitoring server receives the sensor network data the from sink nodes, it will filters, arranges and displays them in turn. And then it transmits data to the database server via TCP /IP network and store them. The monitoring server can forward all sensor data to monitoring clients that connected via TCP/IP.

Monitoring clients is the computer programs that utilizing the sensor data and topology information of a sensor network that received from monitoring servers. Monitoring clients and monitoring servers can communicate each other through TCP/IP network. When past data related with a sensor network is requested by a user, the monitoring client will connects to its database server via TCP/IP network and retrieves the data from it.

2.3.2 MOTE-VIEW: A Sensor Network Monitoring and Management Tool

Mote-view is been designed to simplify, from the perspective of end user, administration of such sensor networks. In particular, introduce a method to reliably perform the calibrated unit conversion of all the sensor readings, as well as a set of abstraction layers to allow for extensibility.[4]

Typically, a sensor network deployment involve three distinct software tiers that is first the mote tier, second the server tier, and last is the client tier. The figure 2.3.2.1 shows the software use in this 3 tier.

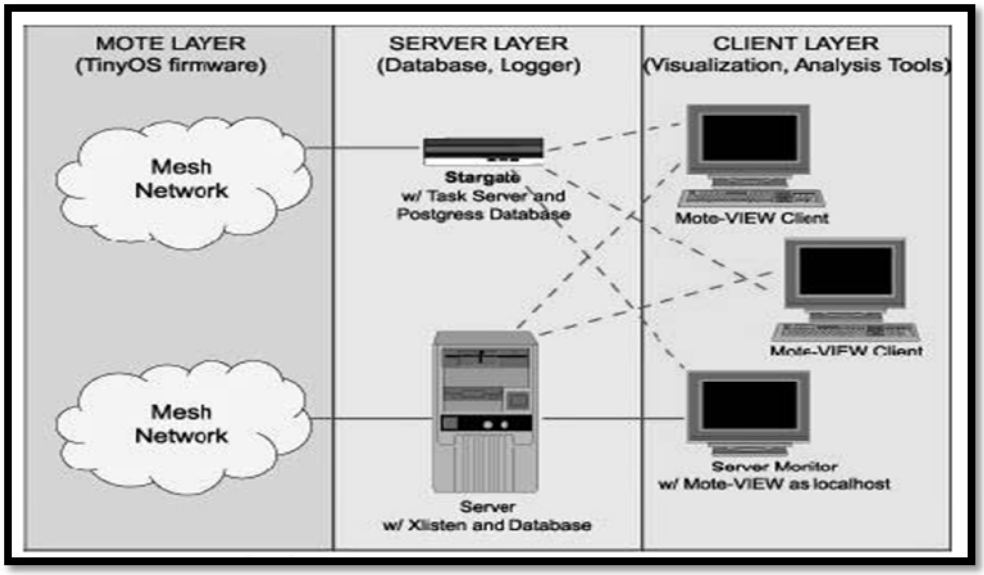

Figure 2.3.2.1 – Software use in typical deployment.

Mote tier embedded the software that runs on the mote hardware like tiny micro-threaded operating system (example TinyOS), firmware application and sensor board driver. A server tier provides a data logging, database storage and a service for forward the sensor data that coming from mote gateway.

The client tier provides with Graphical User Interface (GUI) for manage and visualizing the server and mote tier to be designing to run well on end user platform like in notebook or smartphone.

MOTE-VIEW is a client tier application that been designed to provide the users of wireless sensor networks an interface for end to end management and supervision of a deployment. It focuses particularly on solving the three main problems described earlier: data overload, health monitoring, and the visualization.

2.3.3 SpyGlass: A Wireless Sensor Network Visualizer

SpyGlass is visualizing for the sensor network that can be used to debugging and evaluation the network sensor. Visualization frameworks consist of 3 functions that is wireless sensor network, the gateways at the sensor network and at the visualization software.

It is flexible and has a plug-in architecture means most of its inner component can be easily exchanged or can easily extend. SpyGlass is not only used as visualizing wireless sensor network it also can record activities and can playing it back. [5]

The framework for sensor network monitoring is showing in figure 2.3.2.1. From the figure it shows that information from the sensor network is forwarded to the gateways and then transferred to the visualizer.

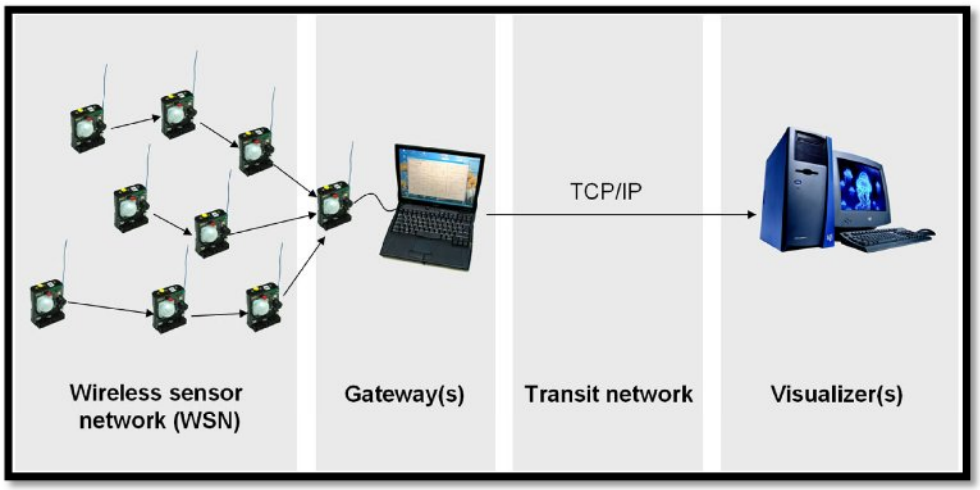

Figure 2.3.2.1 – Sensor network monitoring framework of SpyGlass.

In sensor network part, each sensor node can collects the data using its sensors and derives the new information. If the new information is generated, it will be forwarded to the gateway nodes. Gateways nodes will connect to the sensor network to some other TCP/IP based on the network depend on the setup implemented.

On connection part, it will sends over the contents of it ring buffer. When a new data arrives via the serial link, both will be stored into the buffer and forwarded to all connected visualizers. The circular buffer will enables the system to bridge the time gap of transit network failures or to provide data to visualization stations which connect at a later point in time.

For the network connection between the gateway PC and the machine that running the visualization component, all kinds part of TCP/IP based on networks including the LAN, WLAN or GPRS can be used.

2.3.4 A comparison between NanoMon and SpyGlass software

The similarities between NanoMon and SpyGlass is they are been implemented in Java language that offer platform independence. Others comparison of NanoMon and SkyGlass capability is show in table 2.3.3.1.

Tools / Features	NanoMon	MOTE-VIEW	SpyGlass
Database	MySQL	TinyDB	None
Multiple WSN monitoring	Support	Not support	Not support
Plug-in capability for custom GUI	Support	Support	Support
Sensor node control based on GUI	Not support	Support	Not support
Application platform	Platform independent (implemented in Java)	Windows XP & 2000	Platform independent (implemented in Java)
WSN platform	.Nano-24(NanoQplus) .ETRI-SSN(NanoQplus) .MicaZ(NanoQplus)	Mica-series (TinyOS)	Mica-series (TinyOS)

Table 2.3.3.1 – Capability comparison between NanoMon and SpyGlass

2.4 Microcontroller

Microcontrollers are integrated chips that are often part of the embedded system. The microcontroller includes a CPU, RAM, ROM, I / O ports, and timers like a standard computer but because they are designed to carry only a single specific task to control a single system, they are much smaller and simpler so that they can include all the necessary functions on a single chip. [6]

2.4.1 Arduino Uno

Arduino Uno is a single board microcontroller that designed to make the process of using electronics in multidisciplinary projects become more accessible. The hardware

consists of a simple open source hardware board designed around an 8-bit Atmel AVR microcontroller, though a new model has been designed around a 32-bit Atmel ARM. The software is consists of a standard programming language compiler and a boot loader that can executes on the microcontroller. [7]

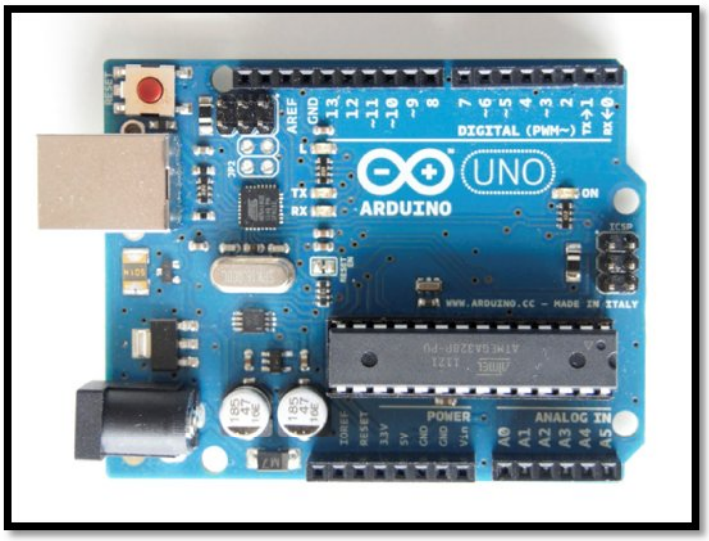

Figure 2.4.1.1 – Arduino Uno board.

2.4.2 Raspberry Pi (Model B)

Raspberry Pi is a credit card sized single board computer that developed in the UK by the Raspberry Pi Foundation with the intention of promoting the teaching of basic computer science in schools. The Raspberry Pi is manufactured through licensed manufacturing deals with Element 14/Premier Farnell and RS Components. Both of these companies sell the Raspberry Pi online. [8]

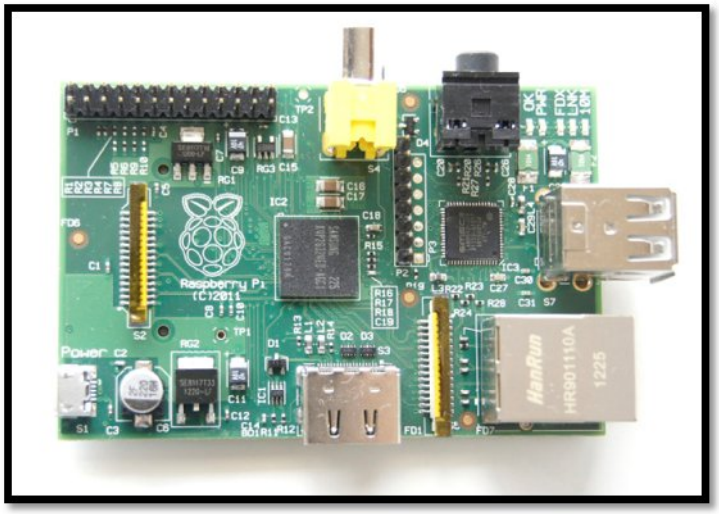

Figure 2.4.2.1 – Raspberry Pi board.

2.4.3 BeagleBone

BeagleBoard is a low power open source hardware single board computer produced by Texas Instruments in association with Digi-Key. The BeagleBoard was also designed with open source software development in mind and as a way of demonstrating the Texas Instrument's OMAP3530 system-on-a-chip. The board was developed by a small team of engineers as an educational board that could be used in colleges around the world to teach open source hardware and open source software capabilities. It is also sold to the public under the Creative Commons share alike license. [9]

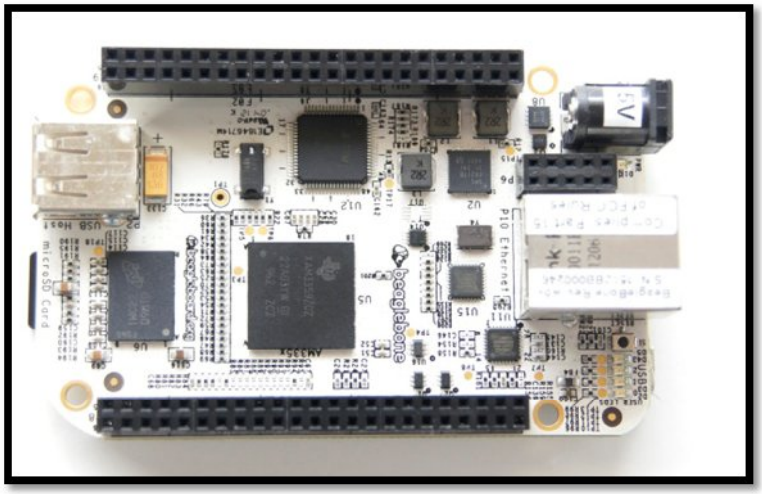

Figure 2.4.3.1 – BeagleBone board.

2.4.4 A comparison between Arduino Uno, Raspberry Pi and BeagleBone

In figure 2.4.4.1 show the available port on the Arduino Uno, BeagleBone and Raspberry Pi board. Notice the BeagleBone and Raspberry Pi have Ethernet ports on the board but Arduino Uno board don't have Ethernet ports.

Figure 2.4.4.1 – (From left to right)
The Arduino Uno, BeagleBone and Raspberry Pi
(Notice the Ethernet ports on the BeagleBone and Raspberry Pi)

In table 2.4.4.2 show the features comprising between Arduino Uno, Raspberry Pi (Model B) and BeagleBone board.

Name	Arduino Uno	Raspberry Pi	BeagleBone
Model Tested	R3	Model B	Rev A5
Price	$29.95	$35	$89
Size	2.95"x2.10"	3.37"x2.125"	3.4"x2.1"
Processor	ATMega 328	ARM11	ARM Cortex-A8
Clock Speed	16MHz	700MHz	700MHz
RAM	2KB	256MB	256MB
Flash	32KB	(SD Card)	4GB(microSD)
EEPROM	1KB		
Input Voltage	7-12v	5v	5v
Min Power	42mA (.3W)	700mA (3.5W)	170mA (.85W)
Digital GPIO	14	8	66
Analog Input	6 10-bit	N/A	7 12-bit
PWM	6		8
TWI/I2C	2	1	2
SPI	1	1	1
UART	1	1	5
Dev IDE	Arduino Tool	IDLE, Scratch, Squeak/Linux	Python, Scratch, Squeak, Cloud9/Linux
Ethernet	N/A	10/100	10/100
USB Master	N/A	2 USB 2.0	1 USB 2.0
Video Out	N/A	HDMI, Composite	N/A
Audio Output	N/A	HDMI, Analog	Analog

Table 2.4.4.2 – Comparing the microcontroller platform.