Development of a Microcontroller Based Robotic Arm

Jegede Olawale, Awodele Oludele, Ajayi Ayodele, and Ndong Miko Alejandro
Babcock University, Ilisan- Remo, Nigeria

jegede1@yahoo.com; delealways@yahoo.com
zeebaby05@yahoo.com; alejandro_mico@yahoo.com

Abstract

Robotic arm has become popular in the world of robotics. The essential part of the robotic arm is a programmable microcontroller based brick capable of driving basically three stepper motors design to form an *anthropomorphic* structure. The first design was for experimental use on a human-size industrial robot arm called PUMA 560 used to explore issues in versatile object handling and compliance control in grasp actions (Bejczy & Jau, 1986). This paper explains the method of interfacing the *robotic arm stepper motors* with the programmed 8051-based microcontroller which are used to control the robot operations. We have employed the assembly language in programming our microcontroller. A sample robot which can grab (by magnetizing) and release small objects (by demagnetizing) is built for demonstrating the method explained.

Keywords: MCU, PUMA, PIO, LATCH, I/O

Introduction

Background

Taking a look back at the history of robot development, a special kind of human-size *industrial robotic arm* called Programmable Universal Machine for Assembly (*PUMA*) came into existence. This type of robot is often termed *anthropomorphic* because of the similarities between its structure and the human arm. The individual joints are named after their human-arm counterparts. “It is worth noting that in our work, the hand is magnetic and not a generalized manipulator. In the proper sense of the word, manipulation is the function of the arm. The function of the arm is to position and orient the hand, act as a mechanical connection and power and sensing transmission link between the hand and the main body of the person. The full functional meaning of the arm rests in the hand” (Bejczy & Jau, 1986). Our work provides important elements that are required to build a simple robotic arm of very high quality. As stated earlier we are making use of the 8051-based microcontroller.

The 8051’s instruction set is optimized for one-bit operations that are often desired in *real world, real time operations*. (Pont, 2002).

Objective

The primary objective is to make the *Robotic arm*, which comprises of three stepper motors, to interface with the In-

---

tel 8051-based micro-controller. It provides more interfaces to the outside world and has larger memory to store many programs.

Scope

The scope of this work involves confirming the 8051 micro-controller Input/Output (I/O) signals are compatible with that of the *robotic arm* stepper motors and testing of the robot's motor signals through programming the 8051 microcontroller. Assembly programming is used to develop the programs for the EPROM 2732 on the 8051 micro-controller platform that takes robot's motor signal as I/O and controls the robot operation programmatically.

We have assumed that after figuring out the interface issues for the Robot with the 8051 micro-controller, the same knowledge can be extended to make very complex robots with enhanced functionality.

Approach

We were able to perform a detailed study of the *robotic arm* and the 8051 micro-controller. We tested the built *robotic arm*, and the stepper motors when the robot is loaded. We also learnt and familiarized with the 8051 micro-controller using assembly language, and converting the assembly language codes to hexadecimal codes using a development board.

Robotics

The word '*robotics'*, meaning the study of robots was coined by Isaac Asimov. Robotics involves elements of both *mechanical* and *electrical engineering,* as well as *control theory, computing* and now *artificial intelligence* (Selig, 1992). According to the Robot Institute of America, "*A robot is a reprogrammable, multifunctional manipulator designed to move materials, parts, tools or specialized devices through variable programmed motions for the performance of a variety of tasks"* (Robotics Research Group, n.d.)

The fact that a robot can be reprogrammed is important: it is definitely a characteristic of robots. In order to perform any useful task the robot must interface with the environment, which may comprise feeding devices, other robots, and most importantly people.

Mechanical Structure of the Arm

In constructing our arm, we made use of three stepper motors and gears since our structure is a three dimensional structure. A typical prototype that we employed is as shown in Figure 1. There is a stepper motor at the *base*, which allows for circular movement of the whole structure; another at the *shoulder* which allows for upward and downward movement of the arm; while the last stepper motor at the *wrist* allows for the picking of objects by the magnetic hand.

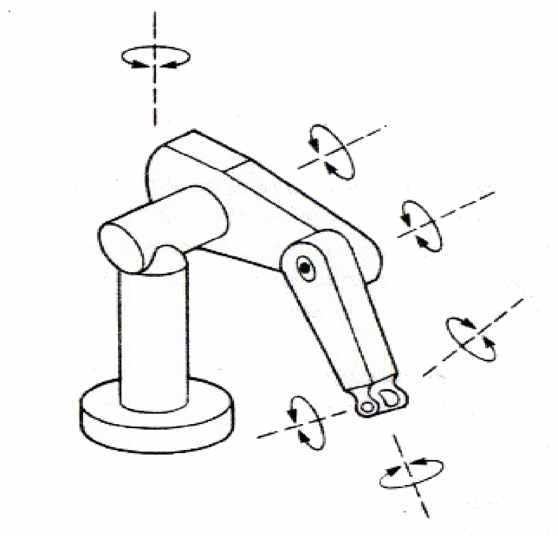

Figure 1. Anthropomorphic Type of Robot Design (Selig, 1992, p.29)

Micro-Controller

A microcontroller is an entire computer manufactured on a single chip. Microcontrollers are usually dedicated devices embedded within an application e.g. as engine controllers in automobiles and as exposure and focus controllers in cameras. In order to serve these applications, they have a high concentration of *on-chip facilities* such as *serial ports, parallel input/output ports, timers, counters, interrupt control, analog-to-digital converters, random access memory, read only memory, etc. The I/O, memory,* and on-chip peripherals of a microcontroller are selected depending on the specifics of the target application. Since microcontrollers are powerful digital processors, the degree of control and programmability they provide significantly enhances the effectiveness of the application.

Embedded control applications also distinguish the microcontroller from its relative, the general-purpose microprocessor. Embedded systems often require *real-time operation* and *multitasking* capabilities. Real-time operation refers to the fact that the embedded controller must be able to receive and process the signals from its environment as they are received. Multitasking is the capability to perform many functions in a *simultaneous or quasi-simultaneous manner* (Yeralan, S. & Emery, 2000, p. 2).

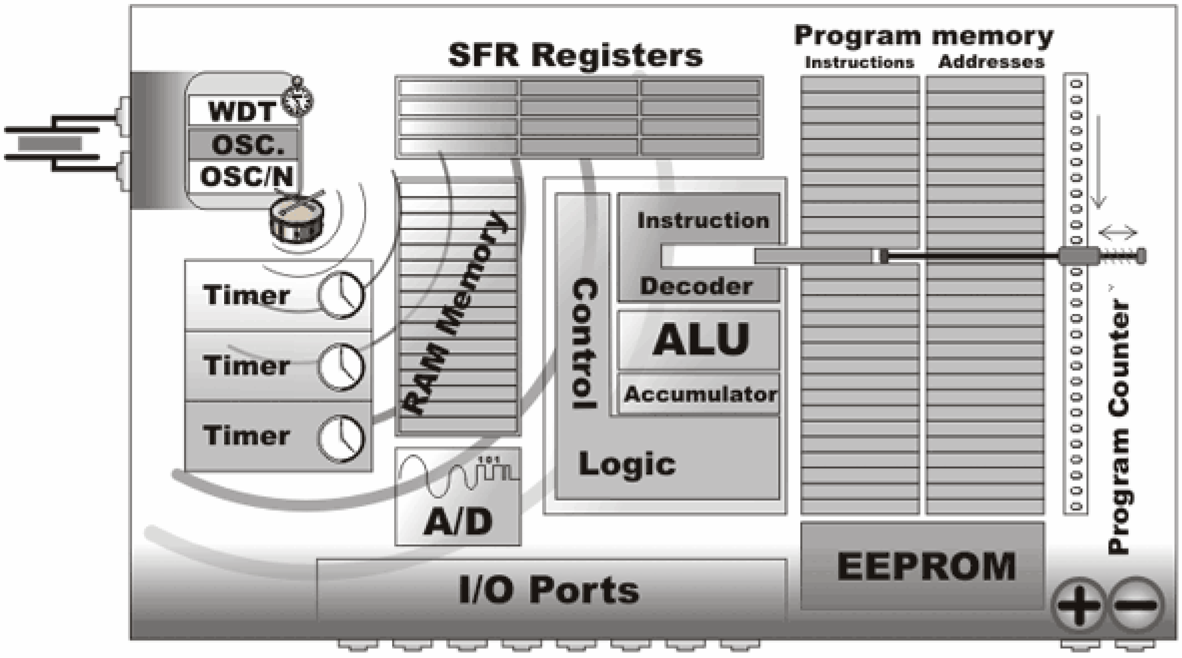

Figure 2. Block diagrams of a Microcontroller (microElectronika, 2004)

The various components of the MCU shown in Figure 2 are explained below:

Random Access Memory (RAM): RAM is used for temporary storage of data during runtime.

ROM: ROM is the memory which stores the program to be executed.

SFR Registers: Special Function Registers are special elements of RAM.

Program Counter: This is the "*engine*" which starts the program and points to the memory address of the instruction to be executed. Immediately upon its execution, value of counter increments by 1.

Control Logic: As the name implies, it which *supervises* and *controls* every aspect of operations within MCU, and it cannot be manipulated. It comprises several parts, the most important ones including: *instructions decoder*, *Arithmetical Logic Unit (ALU) and Accumulator*.

A/D Converter: A/D stands for *analog to digital.* They convert analog signals to digital signals.

I/O Ports: To be of any practical use, microcontrollers have ports which are connected to the pins on its case. Every pin can be designated as either input or output to suit user's needs.

Oscillator: This is the rhythm section of the MCU. The stable pace provided by this instrument allows harmonious and synchronous functioning of all other parts of MCU.

Timers: timers can be used for measuring time between two occurrences and can also behave like a counter. The *Watchdog Timer* resets the MCU every time it overflows, and the program execution starts anew (much as if the power had just been turned on).

Power Supply Circuit: this powers the MCU. (MikroElectronika, 2004).

Methodology

The method employed in designing and constructing the robotic arm are based on the operational characteristics and features of the microcontrollers, stepper motors, the electronic circuit diagram and most importantly the programming of the microcontroller and stepper motors.

Block Diagram

The block diagram of our work is as shown in Figure 3.

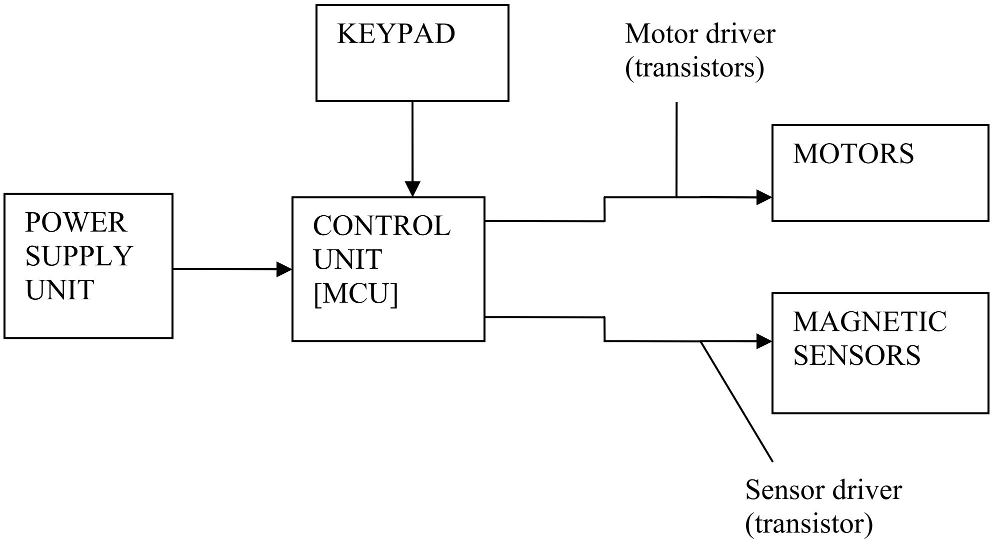

Figure 3. Block diagram

Circuit Diagram

The electronic circuit diagram of the development board is as shown in Figure 4. The connection of the identified components and devices are as shown. The components shown are: the *MCU*, the *LATCH 74LS373, the EPROM 2732, Intel 8255 PIO, diodes, resistors, capacitors, inductors, transistors,* and *op-amps.* This components work together to achieve the set goal of controlling the anthropomorphic-like arrangement of the stepper motor. The microcontroller is the processing device that coordinates all the activities of all the components for proper functioning.

Power Supply

This is used to power the whole system i.e. the *Control Unit, Magnetic Sensing Unit, and the Stepper Motors.* The transformer is a 220/12V step down transformer. We used a bridge rectifier to convert the 12V alternating current to direct current.

The unregulated output from the filtering circuit is fed into a voltage regulator LM7805 and LM7812. These two are chosen for the design because the LM7805 has an output of +5V which is required to power the *Control Unit*, and the *Magnetic Coil* while the LM7812 has an output of +12v which is required to power the *Stepper motors*.

The TIP41 connected to the IC regulators functions as an *emitter follower amplifier* making sure that at least the required voltage by the *Control Unit*, the *Magnetic Coil* and the *Stepper Motors* produced.

MCU 8051

This is the processor. It coordinates the operation of the robotic arm by collecting information from the EPROM, the LATCH, and the PIO; interprets and then execute the instructions. It is the heart of the whole system.

LATCH 74LS373

This is a D-type transparent latch. It is an 8 bit register that has 3 state bus driving outputs, full parallel access for loading, and buffer control inputs. It is transparent because when the enable ***EN****(enable)* input is high, the output will look exactly like the D input. This latch particularly separates the data and address information from the MCU before sending to the instructed destination. The *high-impedance state* and *increased high logic-level drive* provide these registers with the capability of being connected directly to and driving the bus lines in a bus-organized system without need for interface or pull-up components. These latches are particularly attractive for implementing buffer registers, I/O ports, bidirectional bus drivers, and working registers.

We are using this latch because there is a need to separate our 8 bit *data* and 8 bit *address* information from the common line of the MCU, and send them to the appropriate device(s).

8255 PIO

This is a programmable input/output device. It interfaces the connection between the 8051, the LATCH 74LS373, and the EPROM 2732 to external devices such as the stepper motors, (as is our own case) thereby allowing for communication.

(EPROM) 2732

EPROM stands for Electrically Programmable Read Only Memory. We made use of this external EPROM specifically because it makes the controller cheaper, allows for longer programs, and because its content can be changed during run time and can also be saved after the power is off.

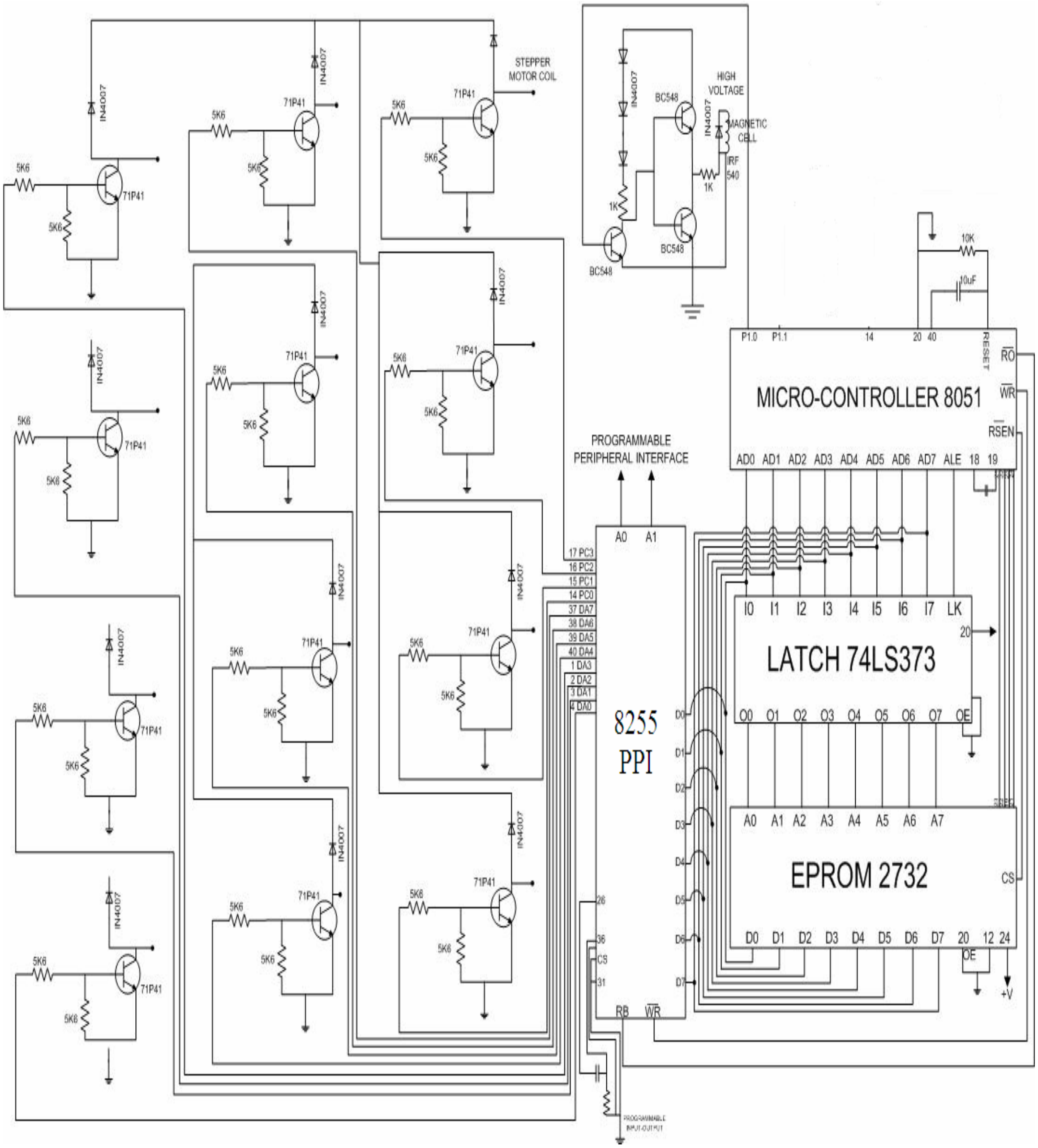

Figure 4. Control Unit
(Digitouch, 2006)

The overall diagrammatical layout of the complete circuit diagram of the whole control unit is shown in Figure 4.

Stepper Motor

The stepping motor is a motor that is driven and controlled by an electrical pulse train generated by the MCU (or other digital device). Each pulse drives the stepping motor by a fraction of one revolution, called the *step angle*.

The Magnetic Sensing Unit

The magnetic sensing unit consists of a magnetic coil which can be magnetized simply by the action of the port P1.0 of the 8051. The port 1.0 was made use of because when designated as output, each of the pin can be connected up to four TTL inputs. That is why we have connected the pin 1.0 to the magnetic coil through three TTL logic.

The design is such that on the downward movement of the wrist, the 8051 sends an electrical signal to the Darlington pair connected to the magnetic coil.

The magnetic sensing unit is powered on by three BC548 Darlington NPN pair transistor, through a diode each and a 5k resistor. The pair amplifies the current and makes the magnetic coil turn into magnet. Then any magnetic material could be picked (by attraction) and then movement continues. The magnetic material can then be dropped at another point when the wrist is made to come down, this also is an action from the 8051 as it withdraws the electrical signal from the coil.

Control Circuit

This is the control panel of the system as it oversees the operations of the mechanical arm, and the magnetic sensing unit. The MCU 8051 of the control unit acts as the brain of the control panel as it coordinates all the activities of the other devices. When power (+5V) was supplied to the control unit, the MCU started off by loading the program from the EPROM M2732A, *interpreted* and *executed* the instruction codes through the various operational principles which had been described in details in chapter three (session 3.2).

The 8051 then sends signal to the stepper motor which moves 9° per step. The stepper motor (M3) at the wrist first moves five times (45°) turning the gears to cause a downward movement of the hand. The stepper motor at the shoulder (M2) moves next stepping five times (45°) and makes the connected gears to cause the movement of the arm 45° forward. Then the stepper motor at the base(M1) moves either ten times (90°) or twenty times (180°), depending on the button pressed, causing the whole structure to turn from right to left(or vice versa) through the connected gears. The magnetic coil resting on the hand becomes magnetized immediately the last gear on the hand stops moving. Then, it magnetizes (picks) any magnetic material it can find and then M3 and M2 moves the arm up while M1 moves (rotates the structure) from left to right (or vice versa) and then the 8051 demagnetizes the magnetic coil thereby making the hand to drop the metallic object.

Results

This work is able to successfully accomplish the defined functionality. A sample robot which can rotate, magnetize an object, lower and raise its arm, by being controlled by the 8051 microcontroller is built successfully. The 8051-development board is soldered and it used the required procedure for the correct operation of the controller. The 8051 development board has been interfaced to the stepper motors such that the anthropomorphic like structure can be controlled from the buttons at the base of the structure (*robotic arm*).

There are four buttons being controlled by the control unit at the base of the arm:

- ON/OFF: the ON button puts on the system while the OFF button puts off the system
- START/STOP: the START button starts the movement of the whole arm from its reset point, while the STOP button takes the arm back to its reset button after completion of its movement.

- RIGHT-LEFT/LEFT-RIGHT: when this button is switched to the RIGHT-LEFT part it causes movement from right to left, while the LEFT-RIGHT part causes movement from left to right.
- 180/90: when the button is on 180, it causes a rotation of 180 degree of the base stepper motor, but when put on 90 degrees, it causes rotation of 90 degrees.

Conclusion

In this paper we have interfaced the robot with different kinds of I/O devices and our method allows for storing more programs to enhance more functionality. From our work, we deduced that in comparison to humans, robots can be much stronger and are therefore able to lift heavier weights and exert larger forces. They can be very precise in their movements, reduce labor costs, improve working conditions, reduce material wastage and improve product quality (Mair, 1988). This is why they're very important in industries because the overall objective of industrial engineering is productivity (Hicks, 1994, p. 61).

Meanwhile, intelligent Control is the discipline that implements Intelligent Machines (IMs) to perform *anthropormorphic tasks* with minimum supervision and interaction with a human operator. This project can be further enhanced to as a multi-disciplinary project involving electrical and computer engineers to work together to create more complex, intelligent robots controlled by the 8051 micro-controller.

References

Bejczy, A. K., & Jau, B. M. (1986). Smart hand systems for robotics and teleoperation. *Proceedings of the Sixth CISM-IFToMM Symposium on Theory and Practise of Robots and Manipulators*, Hermes Publishing.

Engelberger, J. F. (1989). *Robotics in service* (pp 224-226). Kogan Page.

Hall, D. V. (2004). *Microprocessors and interfacing, Programming and hardware* (pp 278-279). McGraw Hill.

Hicks, P. E. (1994). *Industrial engineering and management.* McGraw-Hill.

Mair, G. M. (1988). *Industrial robotics*. Prentice Hall.

microElectronika. (2004). *Architecture and programming of 8051 MCU* (Chapter 1).

Pont, M. J. (2002). *Embedded C*. Addison-Wesley.

Robotics Research Group. (n.d.). Learn More History. Retrieved from http://www.robotics.utexas.edu/rrg/learn_more/history/

Selig, J. M. (1992). *Introductory robotics.* Prentice Hall.

Valavanis, K. P., & Saridis, G. N. (1992). *Intelligent robotics.* Academic Publishers.

Yeralan, S., & Emery, H. (2000). *Programming and interfacing the 8051 microcontroller in C and Assembly*. Rigel Press.

Biographies

Jegede Olawale is a fresh graduate of Babcock University, Nigeria where he obtained a first class degree in Computer Engineering Technology. He is a member of the Nigeria Society of Engineers and the Nigeria Computer Society. He is a Cisco Certified Network Associate. His areas of interest include Artificial Intelligence, Telecommunications, Digital Electronics, Control and Networking. He has published works in several journals of international repute.

Awodele Oludele is a graduate of the University of Ilorin, Nigeria with a Master degree from the University of Agriculture, Abeokuta, Nigeria. He has had several years experience in teaching computer science courses at the university level. He is currently a lecturer in the department of Computer Science and Maths, Babcock University, Nigeria and He is currently undergoing his PhD program . He is a full member of the Nigeria Computer Society and the Computer Professional Registration Council of Nigeria. His areas of interest are Artificial Intelligence and Computer Architecture. He has published works in several journals of international repute.

Ajayi Ayodele is a fresh graduate of Babcock University, Nigeria where she obtained a degree in Computer Engineering Technology. She is a Cisco Certified Network Associate. Her areas of interest include Artificial Intelligence, Digital Electronics, and Networking. She has published works in several journals of international repute.

Ndong Miko Alejandro is a fresh graduate of Babcock University, Nigeria where he obtained a degree in Computer Engineering Technology. His areas of interest include Artificial Intelligence and Telecommunications. He has published works in several journals of international repute.