

# Graphical User Interface (GUI) Timing Control System for DC Motors

Awuhe T. Samuel<sup>[1]</sup>, Tingir, T. James<sup>[2]</sup>, Ape G. Sewuese<sup>[3]</sup>

<sup>1,2,3</sup>Department of Electrical and Electronics Engineering, Federal University of Agriculture Makurdi,

P.M.B. 2373, Makurdi, Benue State., Nigeria

<sup>1</sup>Email: samzfrter@gmail.com, <sup>2</sup>Email: jaytinter@gmail.com, <sup>3</sup>Email: gillsewe@yahoo.com

## ABSTRACT

The research carried out is a Windows compatible GUI based timing control for DC Motors. The design in this research uses a Graphical User interface written using the C- sharp programming language, An Arduino Uno Microcontroller, a HC-05 Bluetooth Module and two DC motors. The GUI is installed on a Bluetooth enabled PC, which connects to the Arduino Uno Controller via the Bluetooth Module. The GUI gives Instruction to the controller on what motor to run for what period of time. Tests were run on the motors for different periods and the two motors were successfully run individually and together from the GUI installed on the PC.

**Keywords:** Dc Motors, GUI, Arduino, Bluetooth, Timing-Control

## I. INTRODUCTION

The Human Computer Interaction (HCI) concept of the early 60's was what birthed the idea of GUI later on. Graphical User Interfaces exist in almost every sphere of human living. GUIs have become useful in different applications like home security systems, power management and instrumentation amongst others. Some GUIs are developed for the blind and visually impaired human beings. Several popular graphical software (like visual basic, C#, Java applets, Abode flash CS4) integrate Graphical User Interface and Programming Scripting Function all-in-one for the interface designer and software developer to work seamlessly for a software development work. The control of DC motors is usually a difficult task. The conventional method of time control is inflexible and inconsistent because it is usually based on external factors like public power supply and so on. The GUI time control system is developed to control the running time of DC motors to enhance flexibility and reliability of the system. They have been existing researches on control of DC Motors. <sup>[1]</sup>Sigarev et al (2016) designed a real-time control system for DC motors. The design of their control system contained two parts: design of the hardware of the control system and design of the DC motor and developing the control system of DC motor by dint of

Matlab Real-Time Windows Target Toolbox. The system was tested and run with desirable results.

<sup>[2]</sup>Abdulrahman and Rosbi (2012) designed a Proportional Integral control (PI) and Artificial Intelligent control (AI) algorithm for control of DC Motors. The AI and PI controllers were analyzed using Matlab [Simulink] software. The DC motor was tested at varying speed and load characteristics due to its ease of controllability. The research was able to illustrate how the speed of the DC motor could be controlled using different controllers

<sup>[3]</sup>Khan *et al* (2015) designed a DC motor speed controlling technique under varying load condition. A Matlab simulation of system with no Load and full-load condition was performed on Simulink platform to observe the system response. The motor speed was kept constant in this experiment. The simulation result of the experiment showed that the motor was running approximately at a constant speed regardless of a motor load.

<sup>[4]</sup>Dubey and Srivastava (2018) designed a conventional PID controller which was used to control DC motors for various industrial processes from many years due to their simplicity in operation. PID controllers require mathematical models to control the plant for different process control applications. In this paper worked out, speed control analysis of DC motor using PID controller. Speed control of separately excited dc motor is done using armature voltage control method. PID controller basics are briefed and various tuning methods of PID controller are explained. Mathematical modeling of dc motor is done and is simulated in Simulink.

The system in this design incorporates a user interface with a Microcontroller and DC Motors.

## II. MATERIALS AND METHODS

The design in this research makes use of a graphical user interface (GUI), an Arduino Uno R3 (with ATmega 328 chip on board), a Bluetooth HC-05 module and two Micro DC Motor (10mA, 6V, 4700RPM).

### 2.1 Mode of Operation

Arduino Uno microcontroller is used as the platform. The DC motors that are connected to the microcontroller

via an BC547 NPN transistor in the common emitter configuration. Each Arduino digital pin supplies a current of 40mA at 5V. The 1N4001 diode is connected in parallel with each of the motors to prevent it from damage due to the electric potential generated when the motor suddenly stops.

The HC-05 Bluetooth module is used to establish a connection between the Arduino microcontroller and the GUI on the PC. The Controller then is connected to the DC motora to perform the required instruction.

**2.2 System Architecture**

The Block diagram of the design is shown in Figure 1. It has the GUI and Bluetooth HC-05 Module as inputs, The Arduino Uno Controller as the processing Unit and the DC Motors as Output

**2.3 Input**

The Input consists of the Windows Based GUI application and the HC-05 Bluetooth Module

**2.3.1 GUI Interface**

The GUI Application has multiple controls, that perform different actions depending on the button the user clicks on. The GUI application consists of four input textboxes and eight buttons. The textboxes are used to input information while the buttons perform different actions. It also has a menu and view option on the toolbar. The first textbox shows the time left for the selected motor to run. The second textbox tells the user the overall running time of the selected motor. The third textbox indicates the motor that is running since there are two motors. The combo-box has a dropdown menu which gives the user an option to select a motor to run. The preset buttons are used in presetting the running time of motor 1 and motor 2. The motor 1 and motor 2 buttons run the motors when the user wants to run any of the motors without setting the running time for any of them. The GUI (installed on a Bluetooth enabled PC) is connected to the controller via the HC-05 Bluetooth module. The screenshot of the GUI is shown in Figure 2.

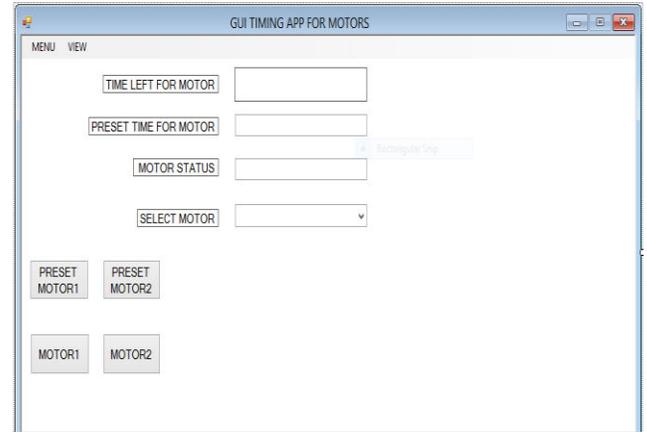


Figure 2: The GUI Application.

**2.3.2 Bluetooth HC-05 Module**

HC-05 Bluetooth module is a wireless device that uses Bluetooth technology. It is a low power, low cost, fast, secure and reliable communication channel for passing information from one device to another at a maximum range of 10 meters. It uses a short wave length, ultra high frequency (UHF) radio waves in the industrial scientific and medium radio band (ISM) from 2.4 to 2.485 GHZ from fixed and mobile devices. It connects the GUI and the Arduino Controller.. Figure 3 is a graphical representation of the Bluetooth Module.

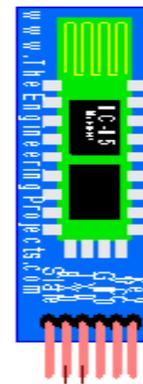


Figure 3: Graphical representation of Bluetooth HC-05 Module

**2.4 Processor Unit**

<sup>[5]</sup>The Arduino Uno R3 has an ATMEGA 328 microcontroller on board. It has 14 digital input/output pins (of which six can be used as outputs), 6 analogue inputs, ceramic resonator, a USB connection, power jack, an ICSP header, and a reset button. The Aduino Uno is connected to the Bluetooth Module trough the Tx and Rx pins ( Pin 0 and Pin 1) of the controller. The DC motors are connected to the controller via the BC547 NPN transistor from Pin 13 and 12. Figure 4 shows the Interfacing of the Bluetooth Module with the Arduino Uno R3

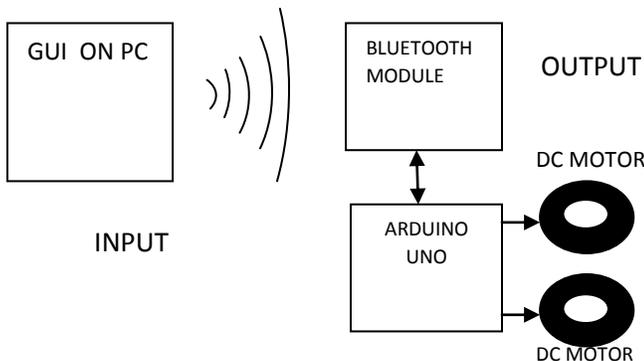


Figure 1. Block Diagram of the Timing Controller System

Figure 4, the voltage divider calculation is presented in the following section: Let  $V_{in}$  be the voltage from the TX pin of the Arduino Uno and  $V_{out}$  be the expected voltage from the Rx (voltage across  $R_4$  resistor), the equation is thus:

$$V_{out} = \frac{R_4}{R_4 + R_3} V_{in} \quad (1)$$

$$V_{out}(R_4 + R_3) = (R_4)V_{in} \quad (2)$$

$$V_{out}R_4 + V_{out}R_3 = V_{in} R_4 \quad (3)$$

$$V = IR \text{ hence } I = \frac{V}{R} \quad (4)$$

$$R = R_4 + R_3 \quad (5)$$

$$I = \frac{V}{R_4 + R_3} \quad (6)$$

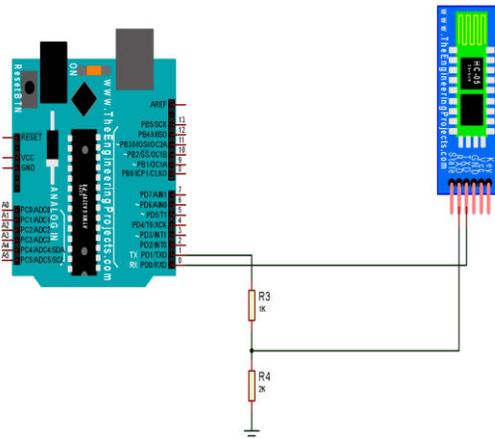


Figure 4: Interfacing Bluetooth Module with Arduino Uno.

## 2.5 Output

The Micro DC Motor is used as Output in the research.

### 2.5.1 DC Motor

A simple DC motor is that rotary part of electrical machines that converts direct current electrical energy to mechanical energy to run the machine because a machine is driven mechanically. DC motor is powered by direct current such as a battery or DC power supply. The DC motor used in this research is a 6V, 10mA, 4700rpm, DC motor. It has a radius of 12mm and thickness of 10mm. It is connected to the controller via a BC547 transistor. The equation for the base resistor is given as

$$R_B = \frac{V_{BB} - V_{BE}}{I_B} \quad (7)$$

The Connection of DC motor is shown in Figure 5.

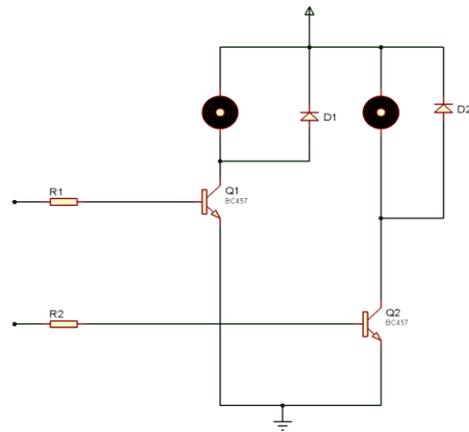


Figure 5: DC Motor Connection

### 2.5.2 Software Design and Flow Charts

The Software for the Home Automation System was written using <sup>[6]</sup>Visual Studio environment (Microsoft Corporation, 2017) The hardware design was simulated using <sup>[7]</sup>Proteus 8.4 (Labcenter Electronics, 2015) The flow chart for the program of the system is shown in Figure 6.

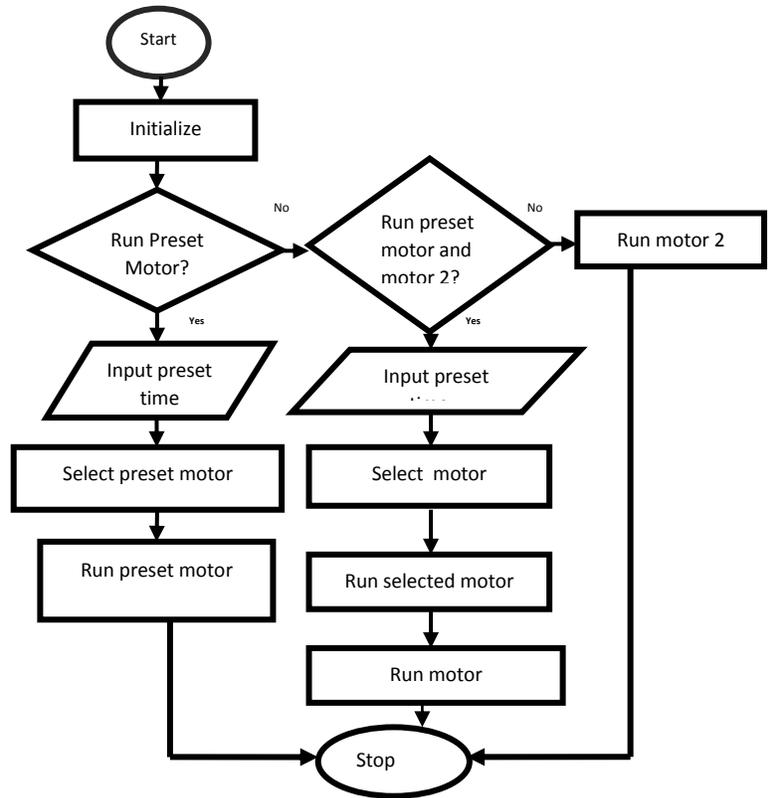


Figure 6: Flow Chart of the GUI Timing Control of DC Motors

## III. RESULTS

The GUI was programmed using C sharp programming language in visual studio 2012 while the Arduino

microcontroller was programmed using C language in Arduino IDE 1.8.

To establish communication between the PC and the hardware via Bluetooth the following steps were taken:

- Step 1: Turn on the Arduino Uno with HC-05 Bluetooth
- Step 2: Go to PC, open Bluetooth
- Step 3: Right click on Bluetooth icon.
- Step 4: Click on open settings
- Step 5: Turn on the Bluetooth on the PC and ensure it is paired with the HC-05 Bluetooth module. A screenshot of the PC/ Bluetooth pairing window is shown in Figure 7

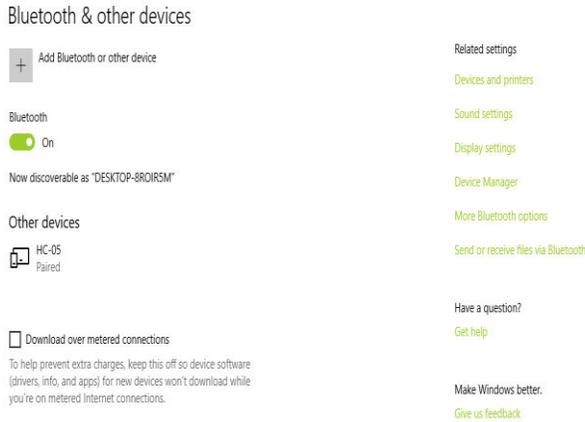


Figure 7: Bluetooth/PC pairing Window

The GUI Application is used to time the first motor for 25 seconds. When the user sets the time for motor 1 to run for 25 seconds, preset motor 1 is selected and the preset motor1 button is clicked so the motor will start running. The motor status indicates that motor 1 is running while the time left for the motor to run can be monitored by the user in the first textbox on the APP. In figure 10, the preset time for motor has elapsed and the left time for motor 1 to run reads 0 seconds. At the same time a message window comes up reminding the user that the preset time has elapsed with a question “Do you want to transfer control to motor 2”. If the user clicks on NO, the motor will stop but if the user clicks on CANCEL, the motor starts running again at 25 seconds. Screenshot is shown in Figure 10.

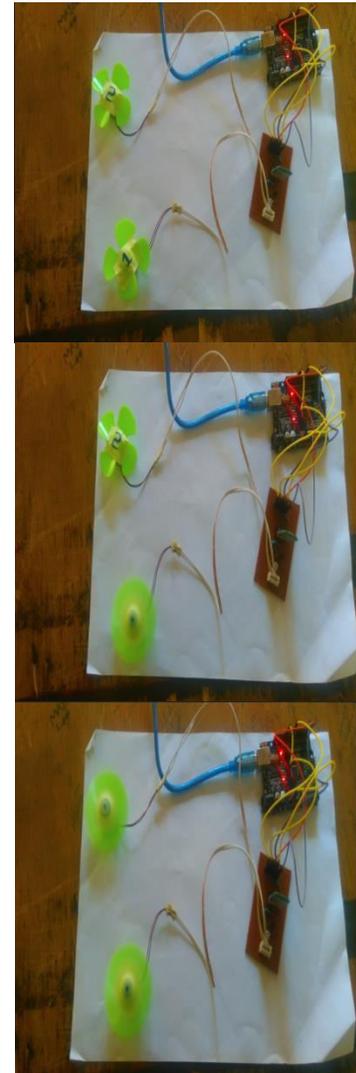


Figure 8 : Physical Implementation Showing Motors running at different instances

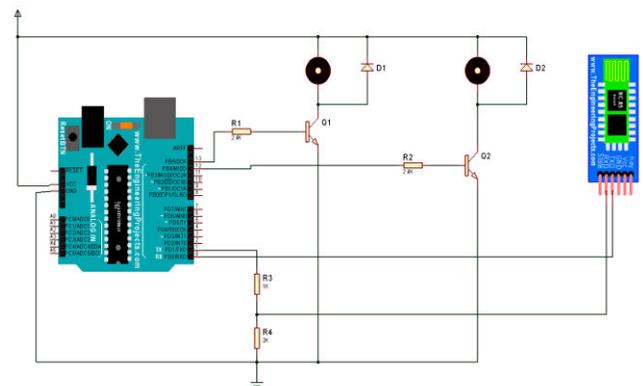


Figure 9: Circuit Diagram of the GUI Timing Control System.

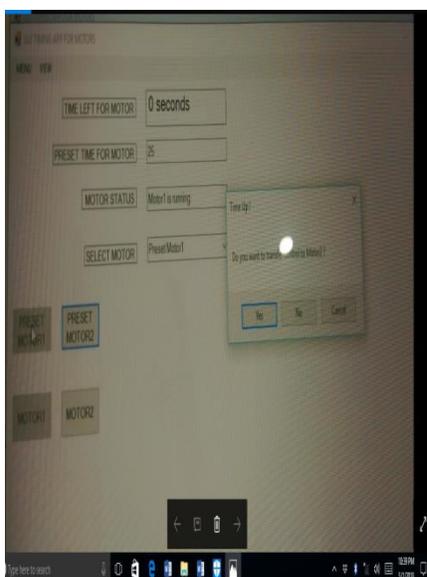
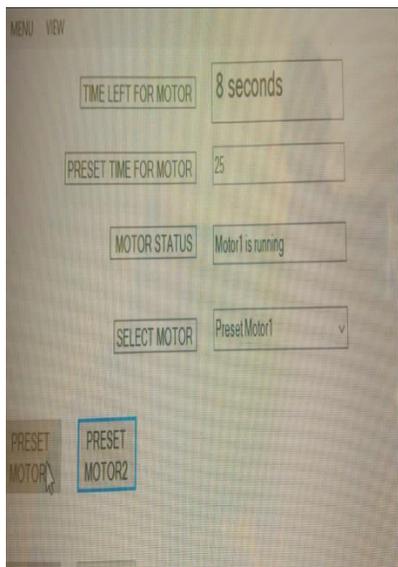


Figure 10: Screenshot of the GUI Application in Operation

*Russia Young Researchers in Electrical and Electronic Engineering Conference (EIconRusNW)*

[2] Abdulrahman A.A.E. and Rosbi B. M.(2012) Modelling and Simulation for Industrial DC Motor Using Intelligent Control. *Procedia Engineering* Volume 41, 2012, pp: 420-425

[3] Khan, M. R., Khan, A. A., Ghazali, U. (2015) Speed Control of DC Motor under Varying Load Using PID Controller *International Journal of Engineering (IJE)*, Volume 9, Issue 3. pp: 38-48

[4]Dubey, S. and Srivastava, S. K. (2018) A PID Controlled Real Time Analysis of DC Motors, *Research and Reviews*.

[5] Arduino UNO for beginners - Projects, Programming and Parts". *makerspaces.com*. Retrieved 4 February 2018

[6] Microsoft Corporation (2017) "Visual Studio 2017" [www.visualstudio.com](http://www.visualstudio.com) [accessed June, 2018]

[7] Proteus Design Suite 8.4, Release 8.4 SP0 (build 21079) with Advanced Simulation. © Labcenter Electronics, 2015.

#### IV. CONCLUSION

The research successfully developed a GUI timing control system for DC motors. During the tests, the time set for motor 1 to run was 25 seconds the results of the tests are shown in Figure 8 and Figure 10 respectively. The results obtained are in agreement with the objectives of the research which is to develop a GUI application that controls the timing of motors, design an Arduino based time controller for motors and to design a system that transfers the timing control from one motor to another.

#### REFERENCES

[1]Sigarev V. , Kuzmina T., Krasilnikov A. (2016) Real-time control system for a DC motor. *IEEE NW*