Available online www.jsaer.com

***Journal of Scientific and Engineering Research*, 2019, 6(7):82-88**

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

A Model Design of Smart Room Controller using Human Detection and Environmental Inputs

Moses O. Onibonoje

Department of Electrical Electronic and Computer Engineering, AfeBabalola University, Ado Ekiti, Nigeria
Emails: onibonojemo@abuad.edu.ng

Abstract The short communication paper models the design of distributed controllers for real-time operation of home interconnected appliances, based on human detection, sunlight detection and room temperature detection. This is with the aim of presenting a realizable model for home automated lighting system through human detection and environmental inputs. The system is modelled using the two position controller action, and implemented using resourceful autonomous devices. The system can execute its control actions remotely by responding to mobile sms codes, and can report its actions by sending alert sms to the mobile platform. The system has provided a reference model for mini smart homes and also eliminates the congestion of data execution with the previous inventions.

Keywords Automation, Controller, Distributed, Human detection

1. Introduction

Over the years, automation has had an incredible impact on various fields and in our everyday life. It has helped simplify majority of the tasks at hand to carry out. Process and procedures are carried out without or with little human interventions through automation. This has yielded to the development and use of controllers in operating and gauging domestic, industrial, commercial devices and equipment. The devices and equipment may include networks, boilers, aircrafts, ships stabilization, oven and others with controllers performing the human functions. The completely automated systems have had several applications with coverage from thermostat-controlled household boiler, to a very large industrial control system with verse input units and output control signals. Meanwhile, controller actions can include ranges of complexities from simple on-and-off to a multi-variable algorithms [1].

The rapid emergence and increasing popularity of home automation systems also result in few complexities and cost which have made it difficult for the end users to accept [2]. The benefits of home automations have been adequately explained [3-5]. The benefits include the better and economical allocation of energy resources leading to cost reduction, greater safety and security, and improved comfort. The elderly and people with disabilities also enjoy the great deal of support from home automation [6]. This is an application domain which provides a promising window for future exploits, research and opportunities. The desire for application of automation to people’s daily live has tremendously increased over the years. This is more pronounced in the allocation and usage of energy resources. The world energy use include 19% for lighting, in which 6% of emissions from greenhouse is attributed to the energy used for lighting [7].

The development of sensor technology has aided the reduction in electronics size, thereby making embedded systems desirable for many applications. The embedded system is a vital component unit of the switching system used in controllers. The advancement in technology as therefore encouraged the implementation of the embedded systems and controllers in automation of various human activities. Automation is gradually taking over the entire human interferences and consequently increasing the comfort quotient of the people. Automatic control of home lighting system is simple system in which the controller action can be experimented. The human ON/OFF control actions can be implemented to automatic regulate based on a predetermined and preset conditions. As documented by [8], the automated system which controls lighting and other gadgets helps to reduce the power consumed during unnecessary operations. Another approach was a system which uses the number of entrants and people leaving to determine the needed for switching the devices. The system did provides a reliable application for public utilities, and ensure maximum efficiency in operation. Also, [9] developed an economical smart home system without the complexity increase. The use of the off-the-shelf components helps in the reduction of the cost and licensing requirements. A paper was presented by [10] detailing the home automation system using Android application and ATmega Microcontroller. The system was designed and developed for a remote household appliance using the discussed Platform through GSM technology. Furthermore, appliances can therefore be remotely controlled using sms based system.

In this paper, a better method of fully automating the domestic appliances is proposed, using the controller design advantages, and combination of human and environmental factors detection as the system inputs. The system incorporates the arduino microcontrollers which control the sensing elements being deployed to monitor the defined device parameters within a location, the Xbee transceivers which interconnect the sensing nodes and the coordinating node, various sensors and the control relays. The basic electrical appliances could be remotely controlled through text messages. This will eliminate the congestion experienced from previously systems, and increase the adaptability of the disabled and elderly persons to the benefits of automation.

2. Material and Methods

The system is designed and implemented by using a single ON/OFF controller for switching the different interconnected appliances which have their inputs as environmental factors such as temperature and lights. The space of interest is divided into four quadrants, with each quadrant having a fan, lighting bulb and set of double plug sockets respectively. The model is developed and implemented

2.1. System Model

The model of the system is represented as a two-position controller being implemented with operational amplifier (Op-Amp) as shown in Figure 1

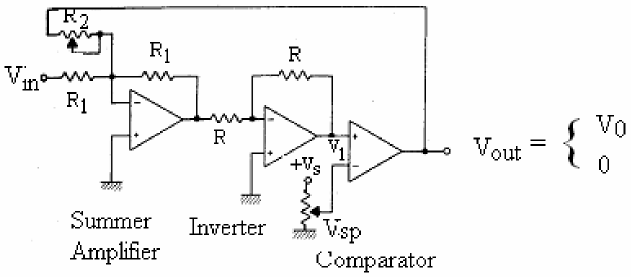

Figure 1: Two position controller with Op-Amp

Assumption: There is an adjustable neutral zone in the design, and the input voltage is V_{in}. If

$$V_{in} = V_H \quad (1)$$

Where V_H is the high voltage value

The comparator output switches to ON state defined as V_0

If

$$V_{in} < V_L \quad (2)$$

Where V_L is the low voltage value

The comparator output switches to OFF state defined as 0V.

The neutral zone of the controller will be defined by

$$N_Z = V_H - V_L \quad (3)$$

Assumption: At the start of the controller action, the output is in OFF state.

$$V_1 < V_{sp},\ V_{out} = 0 \quad (4)$$

Where V_1 is voltage at the comparator input, V_{sp} is the set point voltage, V_{out} is system output voltage

When $V_1 = V_{sp}$

$$V_1 = V_{in} + \frac{R_1}{R_2} V_{out} \quad (5)$$

The comparator will change to ON state when

$$V_1 = V_{in} = V_H \quad (6)$$

Therefore, the ON switch voltage is

$$V_H = V_{sp}, \quad (7)$$

The corresponding output voltage $V_{out} = V_0$.

Also,

$$V_1 = V_{in} + \frac{R_1}{R_2} V_o \quad (8)$$

If $V_{in} = V_L$, the comparator will switch to the OFF state, then

$$V_1 = V_{sp} = V_L + \frac{R_1}{R_2} V_o \quad (9)$$

This will give the OFF switching voltage as

$$V_L = V_{sp} - \frac{R_1}{R_2} V_o \quad (10)$$

The output waveform of the controller will be modelled as shown in Figure 2.

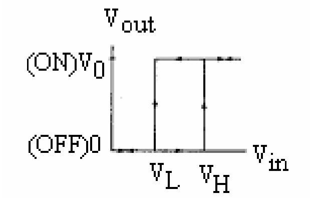

Figure 2: Output response of the model

The neutral zone width could therefore be adjusted by varying the value of R_2, and the location of the zone is relatively calculated using the difference between equation (7) and (10).

2.2. System Design

The system is designed with building blocks of the various parameters to be detected, which include: human detection, temperature, and light. The block diagram of the control unit in system is as shown in Figure 3, while the circuit diagram for the system design model is as shown in Figure 4.

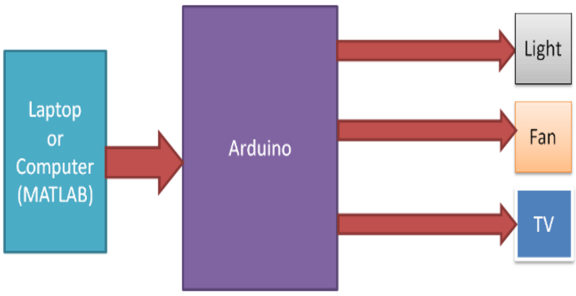

Figure 3: Block Diagram of the Control Unit in the System

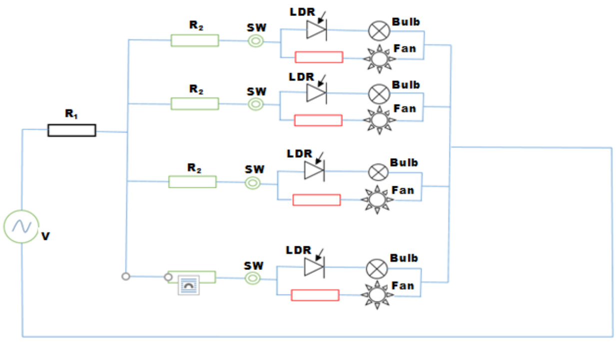

Figure 4: Circuit Diagram for the System Model

The first section of the circuit has the main power source and the primary PIR sensor. The secondary sensors are connected to the primary PIR sensor which has the various socket outlets connected to it. Other sensors for temperature and light detection and appliances are connected to the secondary circuit. The selected PIR sensor detects the human movement, the DHT22 sensor detects the temperature, while the LDR detects the light illumination in the monitored area. The detected parameters are communicated to the arduino microcontroller, which in turn processes them and give the control commands which is transmitted by the transceiver for execution at the coordinating node.

2.3. Software Design

The control unit and the nodes are programmed to execute the preset instructions through a much-linked algorithm. The algorithm for the system is implemented with MATLAB as listed in the appendix. The mobile/graphical user interface for the system is designed with LabVIEW, and the block diagram is as shown in Figure 5.

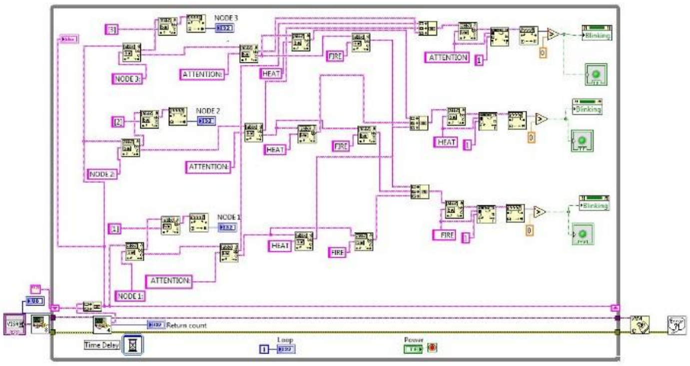

Figure 5: Block Diagram of the User Interface

3. Result and Conclusion

The above circuit is a model proposed for the power saving operation of a typical classroom with the use of various sensors. A wired communication is used to communicate the data from the computer to the arduinousing a MATLAB enabled interface. The various appliances like fan and lights in the classroom are controlled from the buttons on the Graphical User Interface (GUI).

The proposed control system which consists of 5 PIR sensors, four temperature sensors, four photosensors, four socket outlets, four lightbulbs and four fans. The first PIR sensor acts as the main circuit switch for the classroom and only allows power when the system detects a person(s) entering the class. The system switches off when it detects the same number of person(s) to have left the classroom. When the first PIR sensor detects a person(s) entering the class, it initially switches on all other components in the classroom. The other four PIR sensors are programmed with a delay of 60 secs, which gives enough time for the person(s) to settle in a desirable quadrant of the classroom. After the time delay, the other sensors that do not detect motion in their quadrant shut off until motion is later detected. The circuit arrangement was chosen because it allowed power to only flow in the quadrant where motion is detected, cutting down on wasted electricity.

Also, there is a set reference voltage and when the temperature of the room increases the output of the comparator goes high turning on the fan. The circuit also gives the flexibility of adjusting the reference temperature by adjusting the potentiometer. Future works can developed on the system by implementing a faster response and more secured system. The future approach could include the use of facial recognition and biometrics. This will enable the system to recognize the occupants of the monitored space within a particular point in time.

References

[1]. Chattoraj S. Smart Home Automation based on different sensors and Arduino as the master controller. 2015;5(10):1–4.

[2]. Bharmal B, Shahapurkar A, Aswalkar A. Automatic Home Lighting solutions using Human Detection,

Sunlight Intensity and Room Temperature. 2017;691–6.
[3]. Asadullah M, Raza A. An Overview of Home Automation Systems. 2017; (October).
[4]. Baidya N, S PK. A review paper on home automation. 2018; 4(1): 106–10.
[5]. R.R. Yuganandhine, Kiruthika. J, Manju. S, Mythili. KMR. Automatic Room Light and Fan Controller with Bi- Directional Visitor Counter. 2017;20–5.
[6]. M. O. Onibonoje, O. Folorunso, A. Ajibade, K. A. Adeniji, “An Integrated Approach to Automated Control for Air-Conditioned Home Apartments Using Wireless Sensor Network” Indian Journal of Science and Technology, vol 9 issue 40, 2016, pp. 1 - 9.
[7]. W. Maeston, and J. Ahn, “Composite multi-function relay system and its control method” Patent available from: https://patents.google.com/patent/KR101791255B1/en?q=triac&q=circuit+breaker &sort=new. Date Accessed: 21th February, 2019.
[8]. M. O. Onibonoje, and T. O. Olowu, “Real-Time Remote Monitoring and Automated Control of Granary Environmental Factors Using Wireless Sensor Network” In proceeding of 2017 IEEE International Conference on Power, Control, Signals & Instrumentation Engineering (ICPCSI), 2018.
[9]. M. O. Onibonoje, “Design of a Power Optimization Module in a Network of Arduino-Based Wireless Sensor Nodes” ARPN Journal of Engineering and Applied Sciences, vol 14, January 2019, pp. 101 – 105

Appendix

Algorithm:

```
clearar;
globalar;
ar=arduino(‘COM13’);
ar.pinMode(3, ‘OUTPUT’);
ar.pinMode(4, ‘OUTPUT’);
ar.pinMode(5, ‘OUTPUT’);
ar.pinMode(13, ‘OUTPUT’);
functionvarargout = Home_Automation_system(varargin)
% CLASS_AUTOMATION_SYSTEM MATLAB code for Home_Automation_system.fig
%      CLASS_AUTOMATION_SYSTEM, by itself, creates a new HOME_AUTOMATION_SYSTEM or
raises the existing
%      singleton*.
%
%      C = CLASS_AUTOMATION_SYSTEM returns the handle to a new CLASS_AUTOMATION_SYSTEM
or the handle to
%      the existing singleton*.
%
%      CLASS_AUTOMATION_SYSTEM('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in CLASS_AUTOMATION_SYSTEM.M with the given input arguments.
%
%      CLASS_AUTOMATION_SYSTEM('Property','Value',...) creates a new
CLASS_AUTOMATION_SYSTEM or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before Class_Automation_system_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to Class_Automation_system_OpeningFcn via varargin.
%

% Edit the above text to modify the response to help Home_Automation_system
```

```
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
'gui_Singleton',  gui_Singleton, ...
'gui_OpeningFcn', @Class_Automation_system_OpeningFcn, ...
'gui_OutputFcn',  @Class_Automation_system_OutputFcn, ...
'gui_LayoutFcn',  [] , ...
'gui_Callback',   []);
ifnargin&&ischar(varargin{1})
gui_State.gui_Callback = str2func(varargin{1});
end

ifnargout
[varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before Class_Automation_system is made visible.
functionClass_Automation_system_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Home_Automation_system (see VARARGIN)
% Choose default command line output for Class_Automation_system
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
% UIWAIT makes Class_Automation_system wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
functionvarargout = Class_Automation_system_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in light_on.
functionlight_on_Callback(hObject, eventdata, handles)
% hObject    handle to light_on (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
globalar;
ar.digitalWrite(3, 1);
```

```
ar.digitalWrite(13, 1);

% --- Executes on button press in light_off.
functionlight_off_Callback(hObject, eventdata, handles)
% hObject    handle to light_off (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
globalar;
ar.digitalWrite(3, 0);
ar.digitalWrite(13, 0);

% --- Executes on button press in fan_on.
functionfan_on_Callback(hObject, eventdata, handles)
% hObject    handle to fan_on (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
globalar;
ar.digitalWrite(4, 1);
```