❏ 215

Smart Home Automation System Based on Arduino

Bouzid Mohamed Amine[1], Chaib Fatima Zohra[2], Hamani Ilyes[3], Aid Lahcen[4], Allaoui Tayeb[5]
[1]Department of Electrical Engineering, ICEPS Laboratory, Djilali Liabes University, Algeria
[2,3,4] Department of Computer Science, IbnKhaldoun University, Algeria
[5]Department of Electrical Engineering, IbnKhaldoun University, Algeria

Article Info	ABSTRACT
	Intelligent home environments are environments that attempt to ease the user's life in different ways and make it more comfortable using the technology. This paper present the principle of realizing a home control interface based on an Arduino UNO R3 board equipped with a programmable ATmega328 microcontroller. The sketch to be uploaded to the Arduino is written in C language. A desktop application was created in C# language to communicate with the Arduino via the serial port, thus facilitating repetitive tasks such as lighting control, opening/closing doors. A temperature/humidity sensor, an alarm (motion sensor+buzzer), garden lighting and automatic control of the TV via an IR receiver+IR transmitter have been added without forgetting manual control.

Corresponding Author:

Bouzid Mohamed Amine,
ICEPS Laboratory, Departement of Electrical Engineering,
Djilali Liabes University,
BP 89 22000 Sidi-Bel-Abbès-Algérie.
Email: bouzid19amine@gmail.com

1. INTRODUCTION

Home automation includes all technologies (electronics, computers and telecommunications) that allow the automation of the equipment of the home (lighting, heating, shutters, television and alarm), to improve the quality of our life. Home automation has several advantages [1–3]:

- Secures Home Through Automated Door Locks.
- Adds Safety Through Lighting Control.
- Saves Money and Increases Convenience.
- The aid to dependent persons.
- Increase your home's energy efficiency by remotely powering off systems and appliances when they aren't in use.
- One of the biggest hidden benefits that come with home automation is peace of mind. No more worrying if you turned off the lights locked the door or turned off the television.

This area of application is very important and knows a steady increase in the future [4]. Since the research on smart homes spread over a large area, it is not possible to mention each concept or solution in this field. Several project based on arduino board have been discussed in the literature. In [5–7], the bluetooth is used for the exchange of data within a smart home environment. In [8], many projects were presented and discussed. Each project integrates a variety of inexpensive sensors, actuators and microcontrollers that have their own functions. A. K. Dennis [9] makes the project of home automation easy using Raspberry Pi and Arduino with a comprehensive guide.

These are just a few examples of this area and it has been demonstrated that it is already possible to create an environment that tries to meet the needs of the inhabitants. The main objective of this study is to

offer a smart home automation system using Arduino microcontroller. The system can control lights, temperature, humidity, alarms and other household appliances.

The room's light can be controlled manually or via the desktop application. The DHT11 temperature & humidity sensor is used to control the smart home environment. The automatic switching on and off of the light is controlled by the Light Dependent Resistor (LDR) which determines the day light intensity. Also to incorporate security in our system, a motion detector is integrated using Passive Infrared Sensor (PIR) to detect movement in the home when the security system is turned on.

2. HARDWARE PLATFORM

The hardware part consists of a digital computer, an Arduino Uno board, Light Detecting Resistors, motion sensor, an infrared receiver and a transmitter, temperature humidity sensor (DHT11), DC Motor, which are being discussed along with their specific functions. Proposed system architecture is mentioned in Figure 1.
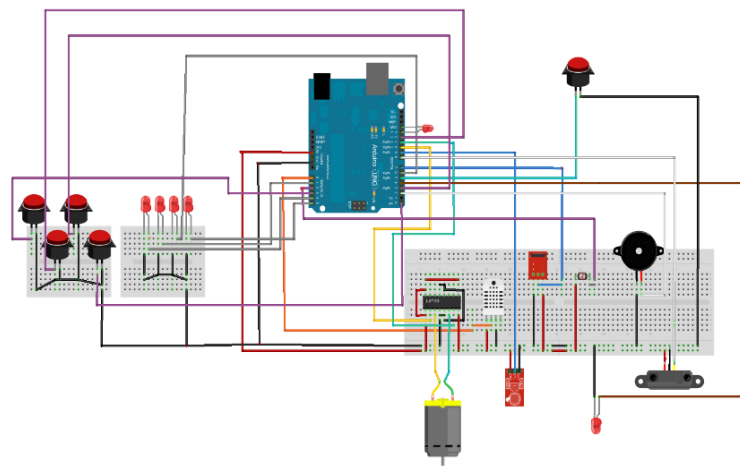

Figure 1. Smart home automation system

2.1. Lighting system

In a traditional electrical installation, the switch and the lamp are connected by the same wire. So, to realize an evolutionary lighting system, we used 5 lamps, one reserved for the garden which will be controlled by a photo-resistance according to the degree of the luminosity (It will be ON as soon as it gets dark). The other 4 lamps will be controlled manually and by the application (the application checks the state of the lamps and controls them). Each lamp is connected to a relay connected to the Arduino, when it is powered by a voltage of 5V; it will open its contact which will allow the necessary electric current to light the lamp.

2.2. The door control system

This part requires the use of a driver called L293D to manage the motors. The L293D circuit is connected to the Arduino. The motor rotates in both directions.

2.3. Security system

In this system, it was only necessary to use a presence sensor and a buzzer. The sensor detects the change at the infrared of the objects if there is a change, its value takes 1 otherwise it takes 0.

2.4. Temperature and humidity control system

To calculate the temperature and humidity, the DHT11 sensor was used, which provides a digital signal proportional to the temperature and humidity measured by the sensor. This ensures high reliability, excellent long-term stability and a time of very rapid response.

2.5. Remote control system

For the remote control system, two components are needed: an infrared receiver and a transmitter. The receiver is used for the cloning of a remote control and its storage in the database and the transmitter to use the remotes available on the database. When a button on the remote control is pressed, it emits an infrared signal that will be captured by the receiver and with the IRemote library the signal will be converted to hexadecimal or integer.

3. PRESENTATION OF THE DESKTOP APPLICATION

In this section, it is explained the results of research and at the same time is given the comprehensive discussion. Results can be presented in figures, graphs, tables and others that make the reader understand easily [2,5]. The discussion can be made in several sub-chapters.

3.1. Window1: Login

When launching the application, the login window is the first window to appear as shown in Figure 2. It contains two fields to enter the user name and password, two buttons: one for the login and the other for closing the application. When the login button is pressed, the application will check if the name exists in the database, if so, it will compare it with the password stored in the database associated with it, if the name and the password are correct the window MENU will be displayed and the type of session will be transmitted to a variable (session) in the 2nd window. Otherwise, an error message will be displayed.

Figure 2. Login window

3.2. Window2: MENU

This window consists of three partsand as shown in Figure 3:
− A bar at the top that displays information about temperature, humidity, date and time of calculation and a button to refresh if the Arduino is connected.
− A bar on the right that contains buttons to access other windows (the session management button is only displayed if the type of session used during the login is administrator).
− At the bottom, a label informs us if the Arduino is connected or not.

Figure 3. Menu window

3.3. Window 3: Remote control

The user selects one of the remote controls available in the database. A panel containing a remote control interface will be displayed with 23 buttons as shown in Figure 4. The buttons have three events: default when the button does nothing, focus when the mouse is above the button and the event Pressed.

To add a new remote control, there is a button that will display a panel consisting of a field to enter the name of the new remote control, a button to start the operation, a label that will serve as a guide (Indicating the button on the remote control to be pressed), a label that will display the received code and a stop button to stop the operation. The operation can be interrupted by clicking the STOP button.
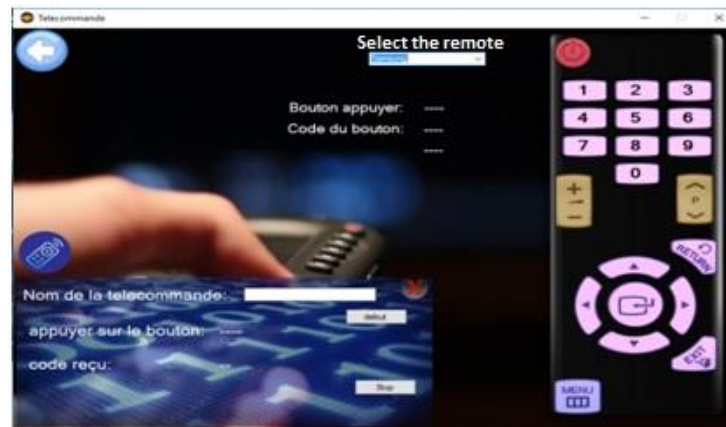

Figure 4. Remote control window

3.4. Window 4: Motion Sensor

This window contains an image and a label that will be changed when there is a movement and the label of the presence of Arduino, but the program of this window uses the multithreading. In Figure 5 is motion sensor window.

Figure 5. Motion sensor window

3.5. Window 5: Control of lights and Doors

This window contains the plan of the house and a button for each room that will allow us to control the light of these rooms, a button for a door, an image that will indicate if the garden lamp is on or off and a button refresh to update all the states of the lamps. In Figure 6 is light and doors control window. The lamp buttons will perform 2 functions, one to turn on or off the selected lamp and the other to refresh. The door knob is more accurate than that of the lamps because if clicked to close the motor will rotate in a specific direction, and turn in the opposite direction to open.
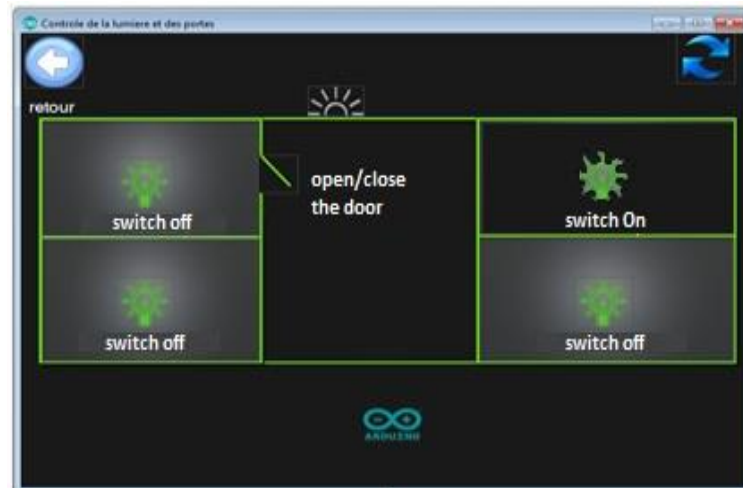
Figure 6. Light and doors control window

3.6. Window 6: Sessions management

It consists of a bar to the right to access other windows, a DataGridView that displays the names of users with their password, and a button to confirm the changes on the list. In Figure 7 is sessions management window.
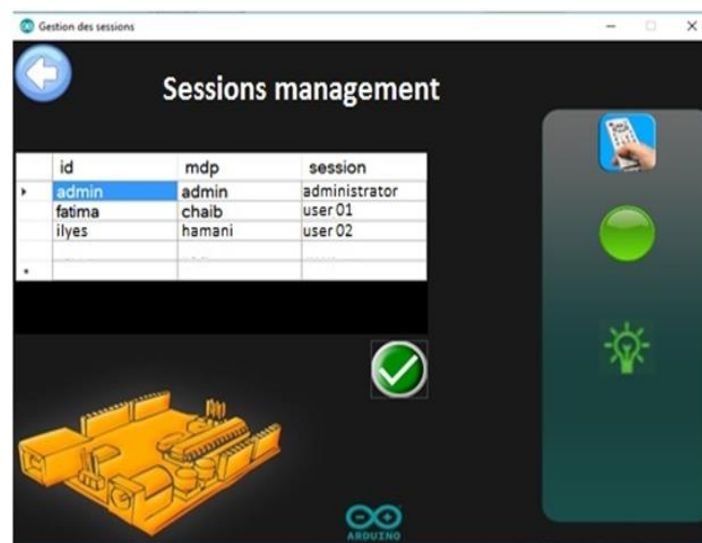

Figure 7. Sessions management window

4. CONCLUSION

In this paper, architecture for smart home control and monitoring system using Arduino is proposed and implemented. It gives a basic idea of how to control different home appliances and provide security by using Arduino Uno controlled from a desktop application.In our project, we tried to implement an embedded system that meets the main functions of home automation for the management of lighting, habitat security, and temperature & humidity control. For these reasons, a desktop application was created to interact with an Arduino via the serial port. Future work will integrate SMS and call alerts, a smart meter of energy consumed in the house, a Bluetooth control by creating a wireless network in the home environment to control and monitor the smart home environment.

REFERENCES

[1] C. Withanage, R. Ashok, C. Yuen, and K. Otto, "A comparison of the popular home automation technologies," in 2014 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA), 2014, pp. 600–605.

[2] R. Harper, Inside the Smart Home. Springer Science & Business Media, 2006.

[3] M. Chan, E. Campo, D. Estève, and J.-Y. Fourniols, "Smart homes — Current features and future perspectives," Maturitas, vol. 64, no. 2, pp. 90–97, Oct. 2009.

[4] M. Chan, D. Estève, C. Escriba, and E. Campo, "A review of smart homes—Present state and future challenges," Comput. Methods Programs Biomed., vol. 91, no. 1, pp. 55–81, Jul. 2008.

[5] R. Piyare and M. Tazil, "Bluetooth based home automation system using cell phone," in 2011 IEEE 15th International Symposium on Consumer Electronics (ISCE), 2011, pp. 192–195.

[6] S. Kumar and S. R. Lee, "Android based smart home system with control via Bluetooth and internet connectivity," in The 18th IEEE International Symposium on Consumer Electronics (ISCE 2014), 2014, pp. 1–2.

[7] J. Potts and S. Sukittanon, "Exploiting Bluetooth on Android mobile devices for home security application," in 2012 Proceedings of IEEE Southeastcon, 2012, pp. 1–4.

[8] M. Riley, Programming Your Home: Automate with Arduino, Android, and Your Computer. Pragmatic Bookshelf, 2012.

[9] A. K. Dennis, Raspberry Pi Home Automation with Arduino. Packt Publishing Ltd, 2013.