Science, Engineering & Education, 9, (1), 2024, 25-30

Implementation of a Home Automation System

Marin Dochev, Simeon Genchev*

VHS of Telecommunications, Sofia, Bulgaria

Received 23 September 2024, Accepted 23 October 2024
DOI: 10.59957/see.v9.i1.2024.4

ABSTRACT

This report presents an innovative home automation system using Arduino and a variety of components, such as sensors, motors, and a Bluetooth module. Through the mobile device application developed through the MIT App Inventor, users can easily manage and monitor the various aspects of their home. The project not only demonstrates the technical capabilities of the system but also highlights the importance of smart technologies in enhancing convenience and security in the home environment.

Keywords: home automation, internet of things (IoT), arduino system, bluetooth communication, smart home systems, sensor integration, access control, RFID technology, environmental monitoring, renewable energy simulation, embedded systems, wireless communication, security systems, automation control, OLED display, servo motors, ultrasonic sensors, DHT sensors, user interface design.

INTRODUCTION

Due to the growing interest in smart technologies and the possibilities of automation of home systems around the world, this report considers the implementation of a home automation system using a development system. The project aims to demonstrate the control and automation capabilities of various devices in the home using Arduino. Google Home and Amazon Alexa are well-known examples of such automation.

EXPERIMENTAL

At the core of the project is the Arduino Mega R3 development board. It connects with:

- RFID, motion, distance, touch, smoke and temperature sensors
- Oled display, Bluetooth module, siren, servo motor, finned motor,
- Second servo motor, matrix keyboard, button and LEDs.

The code managing the development board is written in Arduino IDE [1].

The application for system control over Bluetooth is developed using MIT App Inventor [2].

The diagrams are made with Fritzing [3].

Some of the techniques are from Arduino projects book [4].

Correspondence to: Simeon Genchev, VHS of Telecommunications, Sofia, Bulgaria, E-mail: simeon.genchew@gmail.com

RESULTS AND DISCUSSION

The project features a model house where the development board and its connected components are installed.

Fig. 1 shows the connection diagram of the components. The Arduino Mega reads the data sent from the: motion, smoke, RFID, matrix keyboard, potentiometer, button, Bluetooth, touch, temperature and distance sensors and modules, and then controls the Oled display, siren, servo motors, finned motor, blue LED, RGB LED and the LED strip. The 12V battery is placed for storing the energy from the "wind turbine".

First room

In the most inaccessible part of the home are located the components that do not need regular access like the siren and Bluetooth module HC-05.

The Bluetooth module HC-05 (Fig. 2) provides a connection between the house and a mobile device, and the siren is part of the signalling system. These components do not need regular access, so they can be placed in less accessible areas like tall places, attic, under stairs, behind big appliances, etc. Although these components don't need regular access, they must be reachable.

Roof and Attic

The smoke sensor (Fig. 3) is in the attic. It monitors oxide levels in the air, activating a red LED and the alarm system, if levels exceed a certain limit.

A servo motor then simulates a ventilation system. Different smoke levels are simulated by a potentiometer.

Also included in the project is the simulation of a wind generator (Fig. 3) using a motor to drive fins located on the roof. This motor is controlled

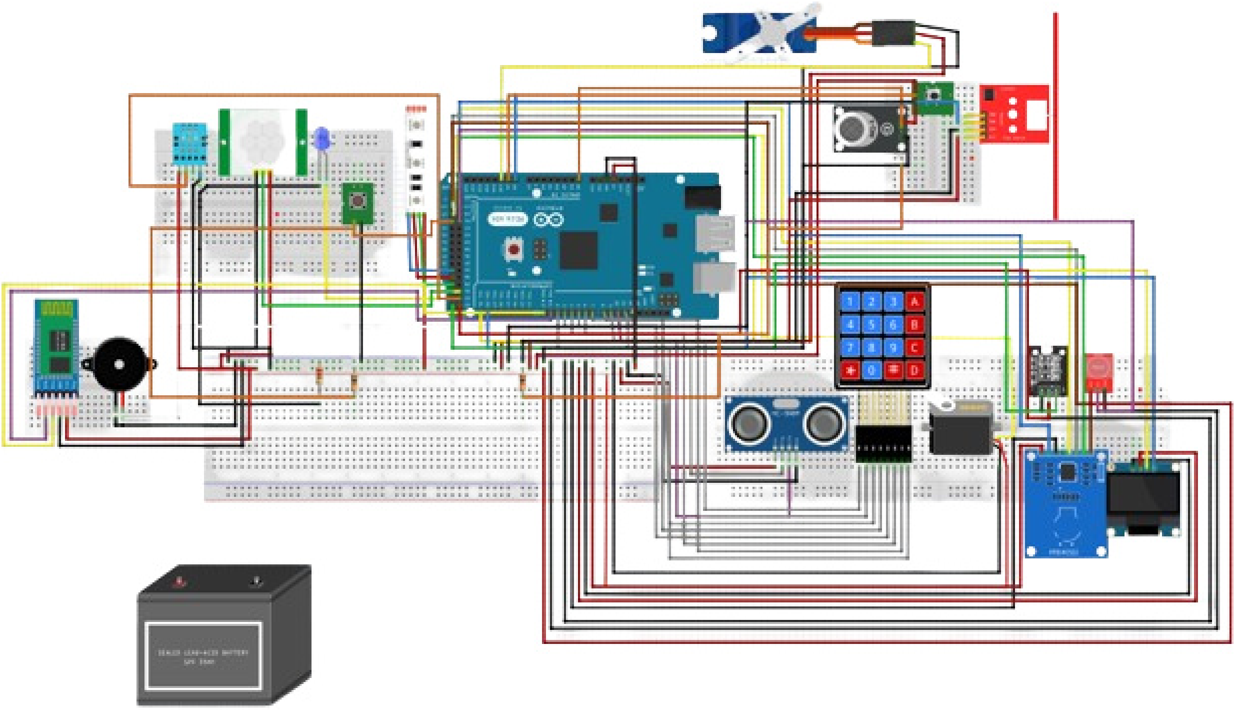

Fig. 1. Circuit of the home automation system.

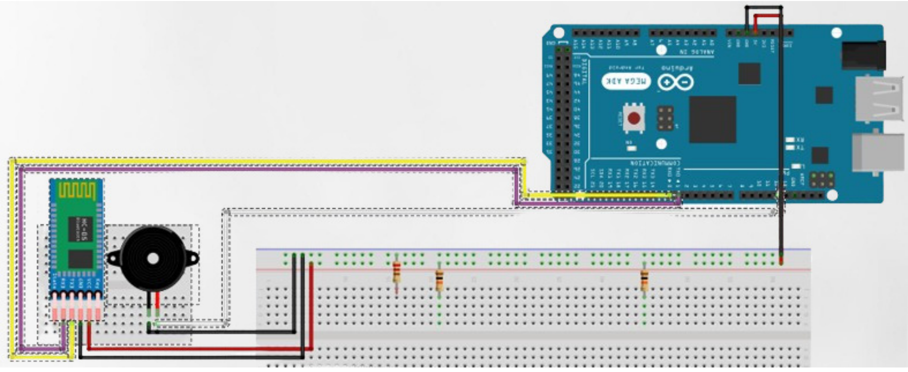

Fig. 2. Connecting the HC-05 Bluetooth module to Arduino.

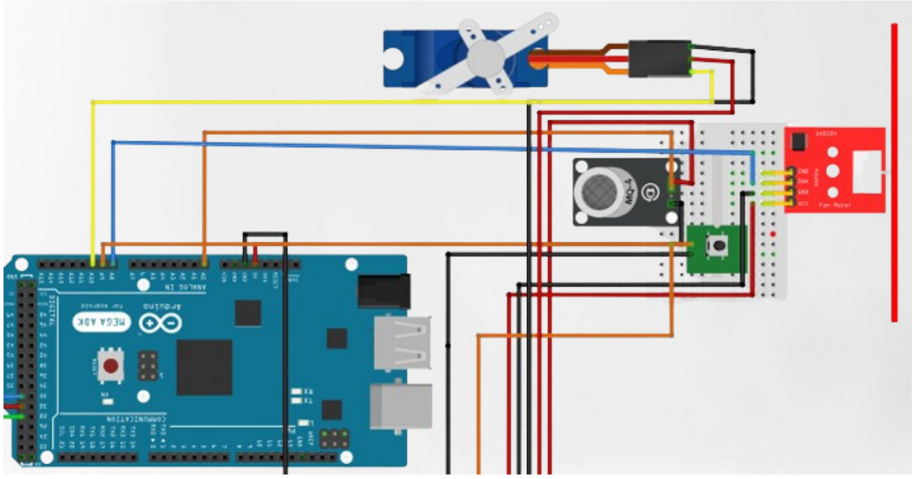

Fig. 3. Connecting the smoke sensor to Arduino.

by a potentiometer that simulates wind and is powered by a 3.7 V battery. Ideally, the energy generated by the wind turbine would be stored in a 12 V accumulator, but since the motor is only simulating a wind generator, the energy cannot be practically stored in the accumulator. This part of the project shows the potential of this kind of project to use any kind of renewable energy.

Access control system

In Fig. 4 the realization of the door access control is shown. It can be divided into several sections:

- ***Using a password to unlock the system***. Through a matrix keyboard, letters and numbers can be inputted to enter a password. If it is correct, the alarm won't go off and the door will unlock.
- ***With calibrated RFID card or Key Fob.*** With the RFID sensor is another way to turn off the alarm and unlock the door. The RFID key uses radio waves to transmit signals, which are picked up by the sensor and if the read code matches that of the authorized one, the door is unlocked, and the alarm is turned off.
- ***Simulating a fingerprint scanner***. The touch sensor is a third way to turn off the system and unlock the door. When there is contact, the circuit closes inside the sensor, and there is a flow of current, that sends data to the Arduino to unlock the system.
- ***Via Bluetooth.*** Through a device, the mobile app and the HC-05, the alarm and the door can be unlocked as well. A mobile device must turn on the Bluetooth option, then through

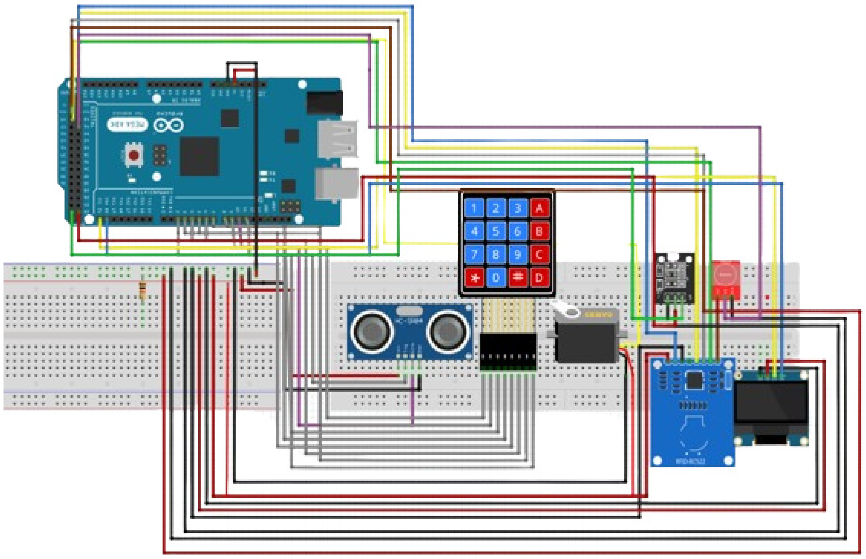

Fig. 4. The realization of door access control.

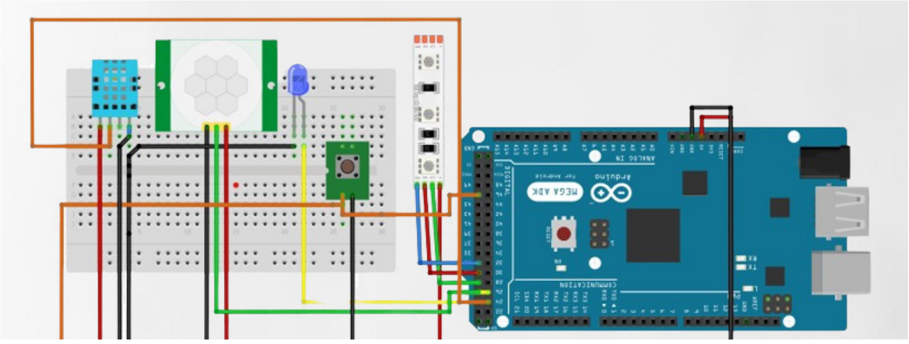

Fig. 5. The schematic solution of the temperature and motion monitoring system.

the mobile app the phone connects to the HC-05, from there the house can be controlled.

The two-tone LED glows depending on the attempt to unlock the alarm. If it is successful, it glows green, if not, red.

The servo motor on Fig. 4. locks the door (by rotating at 90°). And by the door access control it can be unlocked (rotating it at 90°).

In Fig. 4 the Oled display connected to the access system is also visible. It can monitor the password entered. It also shows whether the attempt to turn off the alarm is successful. It also displays data from the temperature sensor, as well as data from the smoke sensor.

The ultrasonic sensor is also included in the system that measures the distance to the door. If it is less than 6 cm. and the correct password is not entered, the alarm goes off.

Second room

In Fig. 5 the schematic solution of the system monitoring the room temperature and humidity, and the presence or absence of movement is shown. The temperature sensor measures the temperature and humidity of the room; by pressing the button on the scheme the temperature and humidity readings are shown on the display. The motion sensor detects changes in the infrared spectrum. When a person moves into the area of the sensor, he emits heat (infrared radiation), the sensor receives that heat and sends data to the Arduino. As a result, a blue LED indicates that there is movement and if the alarm is on, it activates the buzzer until it is turned off or the movement stops.

In the figure, the presence of LED strip is also visible. It simulates the lighting of the home. From the mobile device it can be turned on or off and the colour of the light can be changed. From the keyboard (from a place) the LED strip can also be controlled (by buttons A and B).

Mobile application

The mobile application uses Bluetooth to communicate with the Arduino Mega.

The communication between the HC-05 Bluetooth module, Arduino Mega, and the mobile app is based on serial communication (UART). The HC-05 establishes a wireless Bluetooth connection with the mobile app, which uses MIT App Inventor's Bluetooth Client to send and receive data. This data is transmitted to the Arduino Mega via its hardware serial port, typically using Serial1, which is one of the multiple hardware serial ports that Arduino Mega offers. The Arduino processes the incoming data and can send responses back through the same serial interface. Both devices communicate at a consistent baud rate (commonly 9600 bps), ensuring reliable, asynchronous data exchange.

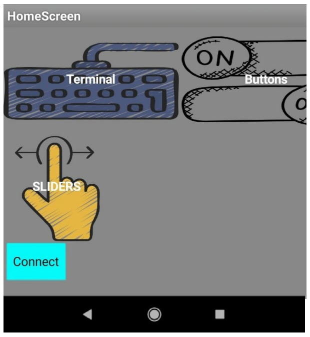

Fig. 6. Home screen of the mobile application.

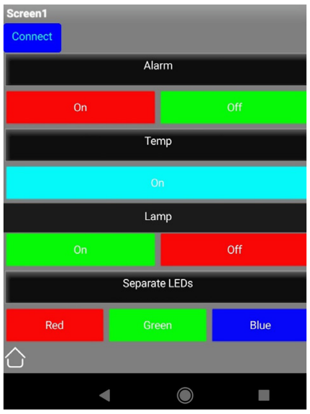

Fig. 7. Main menu with control buttons.

When the user opens the application the home screen (Fig. 6.) will load up, which has three buttons: "Terminal", "Buttons" and "Sliders". In the Terminal page the user can send and receive data to and from Arduino. In the Buttons (Fig. 7) page the user can directly control the different aspects of the house. The buttons for the "Alarm" can turn of and off the Access control system, the button for the temperature "Temp" makes the temperature and humidity appear on the Oled display for 5 sec, the buttons for the "Lamp" can turn on and off all of the LED's strip LEDs (so it glows with white colour) and the "Separate LEDs" buttons can turn On or Off the different colours of the LED strip, and make different colour combinations. The "Sliders" page has sliders that are not configured with a function.

Code explanation

The code is a key part in this and every Arduino project. It has three key components, the first is the naming of the components(pins), initialization of the variables, making the main functions and library calling, its usually in the beginning or the end of the code, before the setup(), or after the loop(). For example:

#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <Keypad.h>
#include <MFRC522.h>
#include <Servo.h>
#include <DHT.h>

"*<Adafruit_GFX.h>* "and "*<Adafruit_SSD1306.h>*" are libraries controlling the Oled display. "*<Keypad.h>*" is the library responsible for reading and processing the data from the keyboard. *<MFRC522.h>* is the library that controls and read the data from the RFID module. *<Servo.h>* controls the servo motors. *<DHT.h>* processes the data from the DHT sensor.

byte rowPins[ROWS] = {9, 8, 7, 6};
byte colPins[COLS] = {5, 4, 3, 2};

These lines of code define which pins are connected to the output column and row pins of the matrix keyboard. They initialize the variables “rowPins” and “colPins” as well.

Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

The definition of the key mapping is done before the void setup() also.

In the void setup() are the functions that will be runed once at the start of the project, for example:

dht.begin();
Serial.begin(9600);
delay(100);
SPI.begin();
rfid.PCD_Init();

Most functions are called here to initialize the used sensors, modules and communication methods.

pinMode(BUTTON,INPUT);
pinMode(MQOut, INPUT);
pinMode(PIR,INPUT);
pinMode(Touch, INPUT);
pinMode(echoPin, INPUT);
pinMode(buzzer, OUTPUT);

In the setup() are also the “pinMode” functions, they configure a specific pin on the Arduino Mega to behave either as an input, output.

The void loop() function in Arduino runs repeatedly after the setup() function. It contains the main code that will execute continuously. For example:

TempDisplay();
RFID();
BTHControl();

TempDisplay(); is the function that is responsible for displaying the temperature and humidity when either the button in the “house” or the button on the mobile application are pressed.

RFID(); is responsible for the RFID locking mechanism.

The BTHControl(); is responsible for all the Bluetooth communication.

CONCLUSIONS

This project demonstrates the control and management of temperature, humidity, movement, smoke, and door access, along with the use of renewable energy sources using Arduino’s development board and software [1]. Arduino’s development board and software [1] support the realization of a project like this, offering flexibility and easily programmable control capabilities for various devices and systems. Through the developed mobile application and Bluetooth, the project can be remotely controlled.

Future improvements could include:

- The Bluetooth module is replaced by a Wi-Fi module, thus achieving greater communication distances between the mobile device and the layout.
- The second servo motor was replaced by a fan of 5v (MF40100V1-A99).
- Replace the touch sensor with a fingerprint sensor (AS608).
- The matrix keyboard is replaced by a membrane keyboard.
- A second Oled display can be added.
- A camera module can be added to monitor the home.

REFERENCES

1. Arduino IDE 2.3.2. 2019. https://www.arduino.cc/en/software
2. MIT App Inventor. https://appinventor.mit.edu
3. Fritzing Fab, 2024. https://fritzing.org
4. S. Fitzgerald, M. Shiloh, T. Igoe, Arduino projects book, 2015.