**Mobile Home Applications**

(AVR PROJECT)

(Omar Temsah)

Table of Contents

[1. Introduction: 3](#_Toc135775321)

[2. System Design: 4](#_Toc135775322)

[2.1. Schematic Diagram: 4](#_Toc135775323)

[3. Implementation: 5](#_Toc135775324)

[3.1. Programming steps: 5](#_Toc135775325)

[3.2. Source Code: 6](#_Toc135775326)

[4. Testing: 6](#_Toc135775327)

[4.1. Results and videos: 6](#_Toc135775328)

[5. Conclusion: 7](#_Toc135775329)

[5.1. Future Upgrades: 7](#_Toc135775330)

# Introduction:

This document describes the development of a mobile controlled home system. The system allows users to control various aspects of their home from their mobile device, such as lights, door locks and security system. The system was developed using an AVR microcontroller atmega32 with LEDs and a servo motor.

The purpose of this project was to create a system that would allow users to have more control over their home from anywhere inside the house using a Bluetooth module. The goals of the project were to:

* Develop a system that is easy to use and reliable.
* Use open-source hardware and software to reduce costs.
* Create a system that is scalable and can be easily expanded to control additional devices.

The scope of this project included the development of the hardware and software for the system. The hardware components included the Atmega32 (AVR microcontroller), LEDs, and servo motor. The software components included the user interface, the communication protocol, and the control logic.

The system was developed using a waterfall development methodology. The first step was to develop a requirements document that specified the functionality of the system. The next step was to design the hardware and software architecture. The final step was to implement the hardware and software.

The system was tested using a variety of methods, including unit testing, integration testing, and system testing. The system was also evaluated by a group of users to ensure that it was easy to use and reliable.

The system was successfully developed and deployed.

The next steps for this project are to expand the system to be controlled by multiple users at the same time, to control additional devices, and to make the system more user-friendly.

The system architecture consists of three main components: the Raspberry Pi, the sensors, and the actuators.

# System Design:

The Atmega32 is the central processing unit of the system. It is responsible for running the software and controlling the LEDs and Servo Motor.

The servo motor is used to control the locks in the home.

The software architecture consists of three main components: the user interface, the communication protocol, and the control logic.

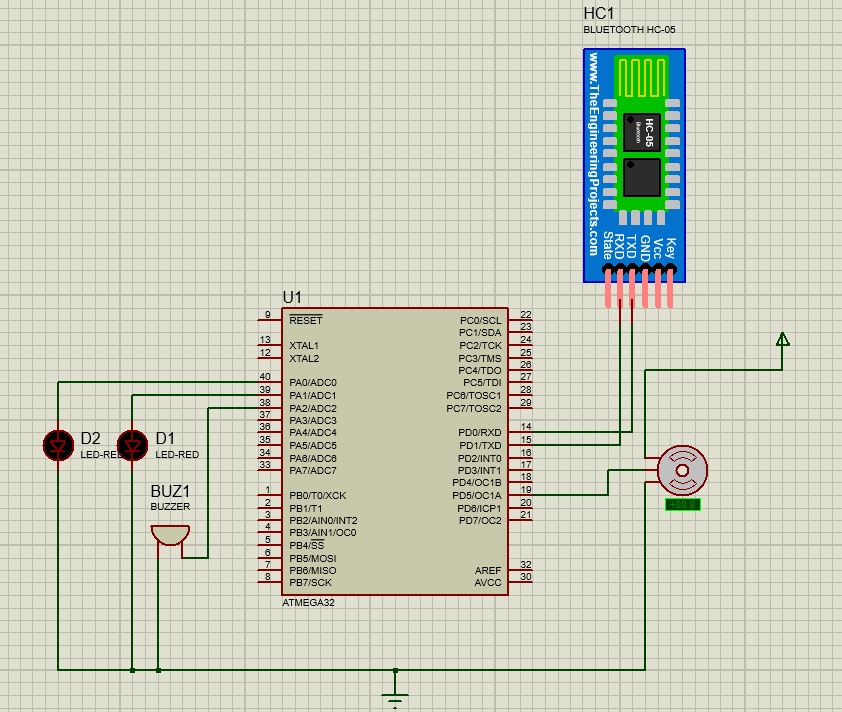
The user interface is used by the user to control the system. The user interface allows the user to send commands to the system and to receive feedback from the system.

The communication protocol used is USART with a Bluetooth (HC-05) to communicate between the Atmega32 and the mobile phone. The communication protocol is responsible for sending and receiving data between the Atmega32 and the mobile phone.

The control logic is responsible for controlling the system based on the commands sent by the user. The control logic is responsible for ensuring that the system operates correctly.

The system is designed to be scalable and can be easily expanded to control additional devices.

## Schematic Diagram:



# Implementation:

## Programming steps:

1. The implementation was done by first programming the necessary drivers for the USART communication protocol as to first test the main functions of the system that is to receive commands from the user mobile and for the Atmega32 to send feedback to the user mobile as to confirm that the commands have been received.
2. The second step is to implement the login system, the system consists of 10 users each with there unique password stored in an array of characters, the user first get asked to enter their username then to enter their password the systems allow for 3 tries if the user fails the first and second time he gets a message saying that his user or password is wrong if he enters it a 3rd time falsely the systems activates an alarm and locks the user for 1 min, in the case the user enters his user and password correctly he is then allowed to send commands to the system and control the connected devices to the Atmega32 with his/her phone.
3. The third step is to add the commands and their corresponding conditions like LedOnA which activites led A or in a bigger scale room A, another command example would be LogOut which allows the user to disconnect from the system and allows another user to connect to the system.
4. Finally adding a snooze or inactivity timer which uses a timer to count a whole minute in real time if the timer passes 1 min the system automatically resets disconnecting the user if the user inputs anything or disables the snooze the timer is reset or disabled.

## Source Code:

In this section a link for the code can be found.

<https://drive.google.com/drive/folders/1CfzlvgTxVXk9GHqSPPcond77SbLOQEkC?usp=sharing>

# Testing:

The system has undergone three tests unit testing, integration testing and system testing.

* Unit Testing:

Unit testing is the lowest level of software testing. It is performed by the developer to test individual units of code, such as functions or methods. The goal of unit testing is to ensure that each unit of code works as expected.

* Integration Testing:

Integration testing is the next level of software testing. It is performed by the developer to test how individual units of code interact with each other. The goal of integration testing is to ensure that the units of code work together as expected.

* System Testing:

System testing is the highest level of software testing. It is performed by a tester to test the entire system. The goal of system testing is to ensure that the system meets all its requirements and works as expected.

The final developed system can be observed in the next section.

## Results and videos:

The system passed all the tests and works flawlessly, and the tests have been repeated many times as to confirm the same expected behavior of the code, also the code have been commented as to ease for anyone editing the main code to know where to go and edit to fine tune the system to their liking.

This is a link to the test video of the final system.

<https://drive.google.com/drive/folders/1CfzlvgTxVXk9GHqSPPcond77SbLOQEkC?usp=sharing>

# Conclusion:

In conclusion, the mobile controlled home system was successfully developed. The system has several advantages over traditional home automation systems. It is more affordable, and easier to use. The system is also scalable and can be easily expanded to control additional devices.

The system has several potential applications. It can be used to control lights, thermostats, security systems, and other devices in the home. The system can also be used to monitor the home for security and safety purposes.

The system is a valuable tool for homeowners who want to improve the comfort, security, and efficiency of their homes. The system is also a valuable tool for businesses that want to improve the efficiency of their operations.

## Future Upgrades:

* a future to allow user to add new users and password, and to allow to change or delete an existing user account.
* A config file for the main code to eliminate the possibility of changing some of core functions in system which would lead to the system malfunction.
* Allow multiple users to use the system simultaneously.
* Adopt to Wi-Fi instead of Bluetooth so it can be used anywhere.
* Dedicated mobile application to make the system simpler and better looking.