Project Report

CSE331L Section 12

Group 4

Faculty: Dr. Md Arifur Rahman

Group Members' Information

Name	**ID**
Sarowar Alam Minhaj	2021175642
Md Maruf Enayet	2013607642
Fahim Muntashir	2021183642
Md. Saife Bhuyain	2021437642

Introduction

Electronic Voting Machines (EVMs) provide a secure, efficient, and user-friendly method for conducting elections. This project implements an EVM system using an STM32 microcontroller (Blue Pill), with the help of a Nondeterministic Finite Automata (NFA) to manage state transitions. The EVM features a 16x2 LCD display, and user inputs are collected using a 4x4 matrix keypad. This report details the EVM's system design, implementation, and testing.

System Overview

The main components of our EVM system are:

- **STM32 Blue Pill**: The microcontroller that serves as the core of the system.
- **ST-Link V2:** Used to upload code into the microcontroller.
- **16x2 LCD Display**: Displays system prompts, voting options, and results.
- **4x4 Keypad**: Allows users to interact with the system by selecting options and confirming votes.
- **DHT11 Temperature Sensor**: The system can also check the surrounding temperature and relative humidity.

Methodology

Hardware Interfacing

STM32 Blue Pill:
The STM32 Blue Pill microcontroller is chosen for its affordability, efficiency, and compatibility with various peripherals. It acts as the central processing unit of the EVM.

16x2 LCD Display:
A 16x2 LCD display is used to show prompts, candidate options, and voting confirmation messages. Interfacing is done via the **I2C2** protocol instead of I2C1 because we need those pins to connect the keypad.

4x4 Keypad:
A 4x4 matrix keypad collects user input. Each key corresponds to a candidate or a specific operation. Key presses are scanned and decoded using **(B9 - A15)** pins on the STM32.

DHT11 Temperature Sensor:
A temperature sensor is connected to the circuit through the **B12** pin of the STM32. It can show real-time ambient temperatures along with relative humidity.

Software Design

Nondeterministic Finite Automata (NFA):
The NFA manages the state transitions of the EVM. States and their description is listed in the table below.

States in NFA

State	Options	Description
PROLOGUE	N/A	Initial state of the system. Will be redirected to HOME when the Passkey is entered
HOME	[Admin] 1 Menu # Exit [Voter] 1 Vote # Exit	Home screen
MENU	1 Stats 3 Take Vote 3 Publish 4 Temperature	Menu screen. Admin choose different options from here
VOTE	1 Cand_A 3 Cand_B 3 Cand_C 4 Cand_D	Voting screen for voters
CONFIRMATION	1 Yes * Back	Confirmation dialog for the voter
STAT	1 Vote counts * Back # Exit	Different statistics options are listed on this screen
COUNT	* Back # Exit	Vote counts of each candidate
SUCCESS	N/A	Voting succeeded
ADMIN_LOGIN	N/A	Admin logged in successfully
PUBLISH	* Back # Exit	Vote published. Results will be displayed here
TEMPERATURE	* Back # Exit	Temperature and Humidity can be checked on this screen

The source code of the implementation can be found here:
https://github.com/sarwar76200/STM32-EVM

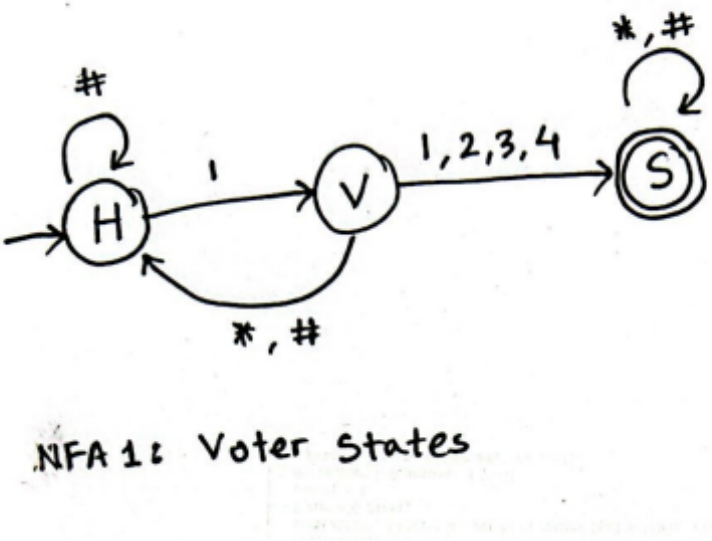

Fig 1: Transition between `Voter` NFA states

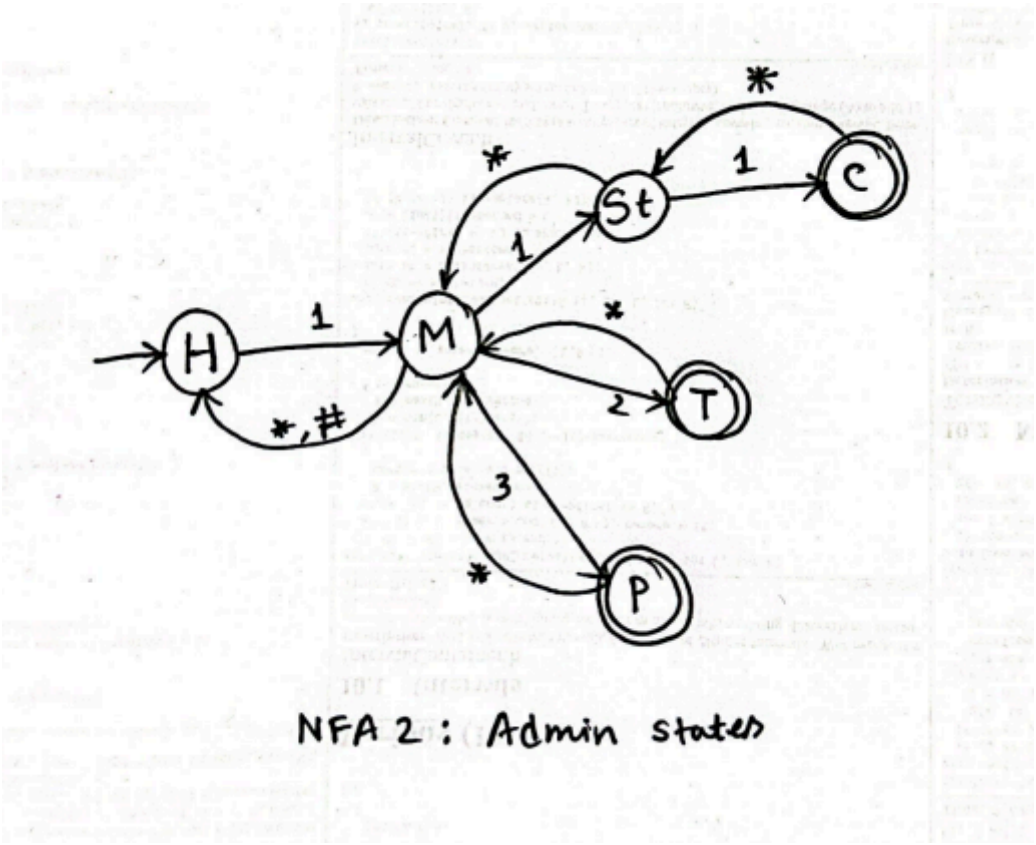

Fig 2: Transition between `Admin` NFA states

The overall flow of the system

1. At first, the system is in `PROLOGUE_STATE`.

2. It continuously scans for user keypresses, and as soon as it gets a substring that matches the `Passkey`, it immediately redirects to the `HOME_STATE` with admin privileges.

3. The admin can do a lot of things, such as Take Votes, See Statistics, Publish Vote Results, etc. He or she can navigate between the states with the keypad, a "`*`" to go back, and a "`#`" to return home.

4. If the admin chooses the option to `Take Vote`, he will be redirected to the `HOME_STATE` again, but as a voter this time.

5. A voter has very limited functionalities; he or she can only cast his or her vote and cannot access the menu.

6. Once the voter is done with his or her vote, he or she will see a confirmation window, once confirmed, he or she will see a success message. After that, the system will be frozen for him or her, and he or she can no longer interact with the system.

7. Only when an admin enters the `Passkey` it will be redirected to the `HOME_STATE`, and the system will be functional again.

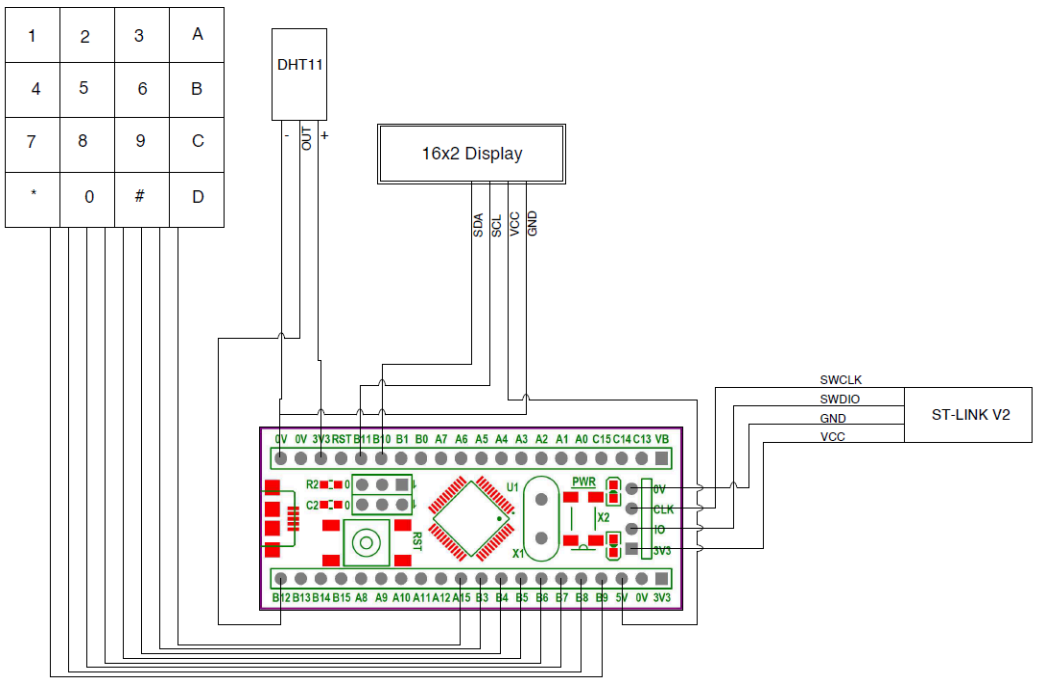

Fig 3: Circuit Diagram of the EVM system

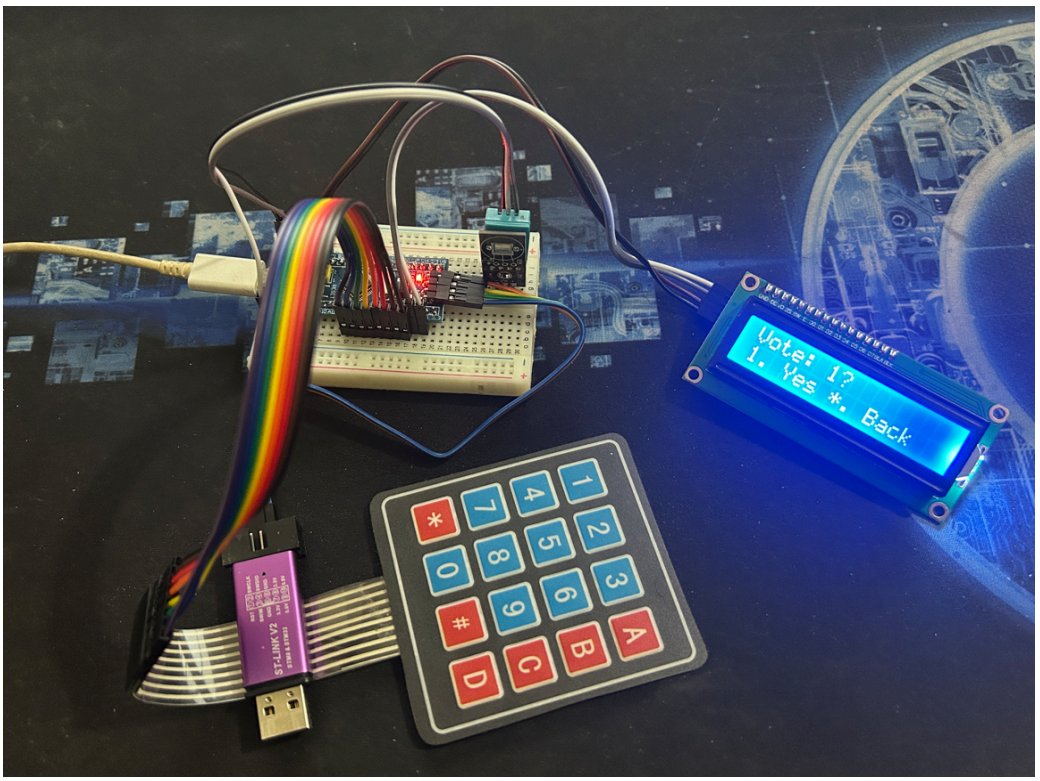

Fig 4: A picture of the circuit with the components

Testing and Results

The EVM was tested quite a bit to make sure everything worked properly. Some of the key findings are:

- The overall system works smoothly without any problems.
- Transitions between the states are done without any errors.
- Admin control was handled properly.
- The UI is responsive and intuitive for the users.

However, we've encountered some unexpected behaviors due to the faulty keypad. Using a different keypad would solve the problem.

Conclusion

This project demonstrates the implementation of a reliable EVM system using an STM32 microcontroller. The use of an NFA ensures structured state management, while the LCD and keypad provide an intuitive user interface. The system is scalable and can be extended with additional features such as wireless communication or biometric authentication.