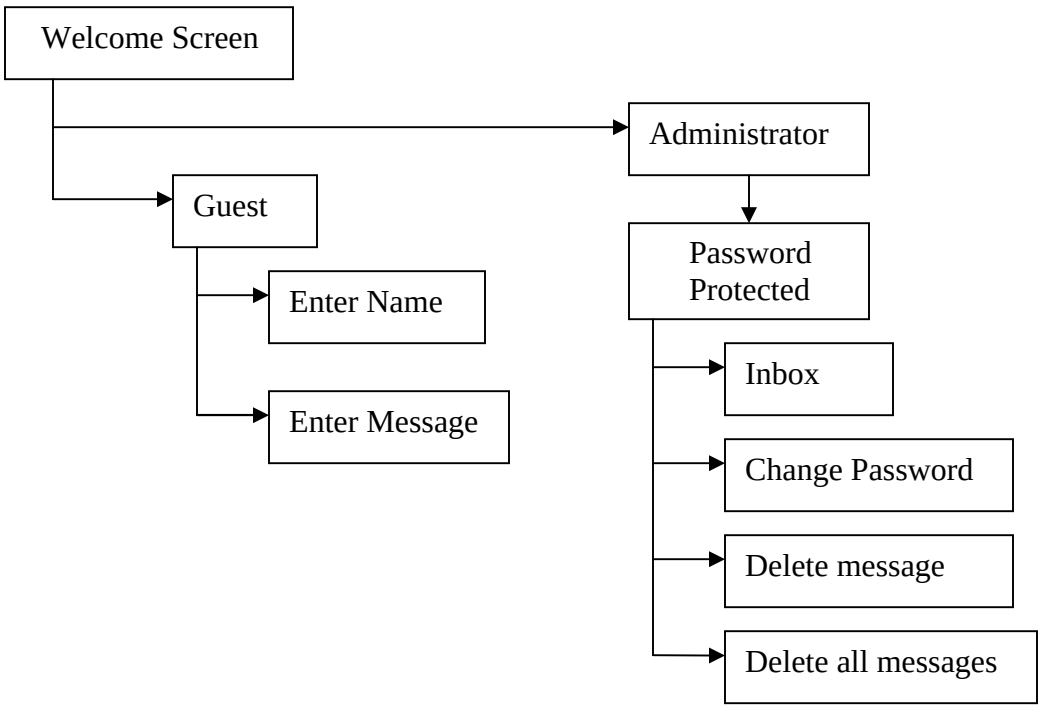
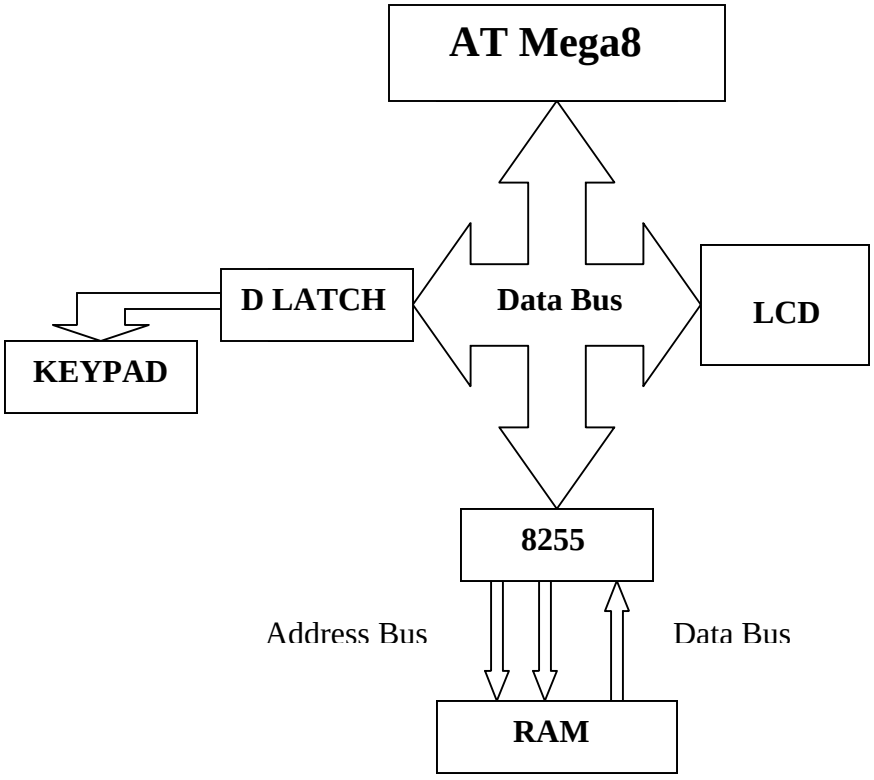
This article describes the hardware and software implementation of a Door Messaging System. Our main aim was to develop an embedded, menu driven door messaging system allowing the user to store and retrieve messages in a password protected environment. This system is a battery operated stand alone system.



The system is accessible in two modes- Guest and Administrator. When accessed as a guest, the name and message are accepted and stored. When accessed in the password protected administrator mode, the user is able to retrieve messages. The administrator is offered a menu of options like accessing the inbox, deleting messages and changing the password. To achieve the aforementioned features, the AT mega8 had to be interfaced with a 32kx8 bit RAM, a 2x16 LCD display and a switch-based keypad (16 push-button switches). A common data bus (all pins of port B) interacted with the keypad, the LCD and the memory simultaneously.



The **32kx8 bit RAM** has 28 pins, 15 address pins and 8 I/O pins. In addition to this there are 3 control signals. One to select the chip and the other two are to enable the chip in read or write mode. Keeping in mind the heavy requirement of port pins by the memory, an 8255 was introduced, to extend our ports.

The 8 data pins of 8255 are connected to the common data bus of our system, which is shared by the LCD and the keypad (via the D-latch).

The **keypad** was built using 16 individual push-button switches arranged in a 4x4 matrix. All row terminals of a switch are connected and likewise all column terminals are connected. Each column is scanned one after the other and as soon as a switch is pressed direct electrical connection between the leads of the respective column and row are made and so the switch pressed can be traced by the system. Since the keypad uses a shared data bus it needs to be isolated when not in use. To make this possible, a **D latch** is put between the data bus and the keypad. When we want to access the data bus, the D latch is kept in transparent mode and at all other times it is kept in high impedance mode so that the keypad does not load the data bus.

The External SRAM

The **8255** uses an 8 bit data bus and 5 control signals. In return it furnishes three 8-bit ports A,B and C. The chip can be programmed (using the control signals) to interact with only one of the three ports at a time. Port C of 8255 is connected to the 8 I/O pins of the memory whereas Port A and Port B of 8255 are connected to the address pins of the memory i.e. Port A to the lower 8 bits of the address pins and Port B to the upper 7 bits of the address pins.

Software Implementation

The source code is written in C and compiled using AVR-GCC compiler.

1 . Functions for LCD operation:-

Specific instructions for initializing and displaying characters on the LCD are found in the data sheet of the LCD. The instructions and data to be displayed on the LCD are put on the data bus and are received by the LCD using control signals. The LCD control signals are sent through Port C of the AT mega8. Since the LCD does not recognize the ASCII code for the backspace key there needs to be a special function to deal with this case. This function reads the current address, takes the cursor back one position while clearing the previously entered character.

2. Function for keypad operation

This function waits for the user to enter a key and returns the respective value.T his function boasts of being a blocking as well as a non blocking function. Until a key is hit it behaves as a blocking function. The keys are overloaded with multiple characters for different number of hits of the same key in rapid succession. Once a key is hit, the function behaves as a non blocking function, waiting to see if the same key is hit again within the stipulated time. debounce in switches is taken care of by a function that returns 1 if debouncing is still taking place and 0 if the switch is now in stable position. It also waits for a certain amount of time to let the debouncing subside.

3. Menu driven system functions

The contents of the menu are stored as an array. Manipulation of two variables pointing to the two displayed menu options controls the circular scrolling of the menu. As different scroll keys are entered, the two variables start pointing to different options of the menu which are then displayed on the screen. When the user selects an option, the function returns a number corresponding to the selected option.

A special function is dedicated to the display of messages on the LCD. This function displays the message, two lines at a time, on the LCD. It also allows the user to scroll up and down the message. Firstly it calculates the number of delimiters in the message and saves the position of each. Two variables are used to point two delimiters at a time that mark the beginning of the two strings that are to be displayed. The two strings are then retrieved and displayed on the LCD. On pressing the scroll keys the delimiters that are being pointed to are changed in such a way that the user can scroll up and down.

When the user selects an option, the function returns a number corresponding to the selected option.

A special function is dedicated to the display of messages on the LCD. This function displays the message, two lines at a time, on the LCD. It also allows the user to scroll up and down the message. Firstly it calculates the number of delimiters in the message and saves the position of each. Two variables are used to point two delimiters at a time that mark the beginning of the two strings that are to be displayed. The two strings are then retrieved and displayed on the LCD. On pressing the scroll keys the delimiters that are being pointed to are changed in such a way that the user can scroll up and down.