**PO4\_DGELV\_DIGITAL ELEVATOR**

**HSI DOCUMENT**

**Version 1.0**

**Draft**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 1/24/2020 | 1.0 | Initial Draft, specifying used hardware. | Sara Abdallah Ahmed Youssif |
|  |  |  |  |

**Table of Contents**

Table of Contents

[1 Introduction 4](#_Toc30724599)

## 1.2 Specification Objectives…………………………………………………………………………………………………………………….4

1.[2 System Overview: 5](#_Toc30724602)

[2.1 Definition: 5](#_Toc30724603)

[2.2 Hardware Requirment: 5](#_Toc30724604)

[2.3 Block Digram](#_Toc30724605) 7

[3 Hardware Connection:](#_Toc30724606) 8

# 1 Introduction

This section introduces the Hardware Software Interface (HSI) for a digital Elevator with lock system and specify the hardware used in the system.

## 1.2 Specification Objectives

The objectives of this specification are to:

* Provide a Hardware system design system overview for the digital Elevator with lock system

# 2 System Overview:

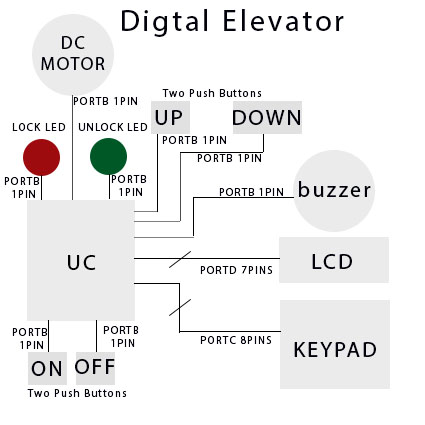
## 2.1 Definition:

The digital Elevator has lock system for 10 users that can be configured and edited by the user with high security sequence to ensure that only approved users can use the elevator.

## 2.2 Hardware Requirement

|  |  |  |
| --- | --- | --- |
| **Feature** | **REQ\_ID** | **Used Hardware** |
| **Microcontroller** | REQ\_ PO4\_DGELV \_HSI\_01\_V1 | The system will be controlled using AVR microcontroller. |
| **small model for simulation** | REQ\_ PO4\_DGELV \_HSI\_02\_V1 | The System will have DC motor to Indicate the movement of the elevator clock wise direction for moving up and anti-clock wise for moving down. |
| **elevator direction selection** | REQ\_ PO4\_DGELV \_HSI\_03\_V1 | The system will have two push buttons to select direction up and down. |
| **reset the system** | REQ\_ PO4\_DGELV \_HSI\_04\_V1 | The system will have two push buttons on and off buttons used to rest the system for 2 sec. |
| **Screen for displaying data** | REQ\_ PO4\_DGELV \_HSI\_05\_V1 | The system will have LCD screen CLCD (Character Liquid Crystal Diodes) That will show the user name and password. |
| **Alarm subsystem** | REQ\_ PO4\_DGELV \_HSI\_06\_V1 | The system will use buzzer to indicate for wrong password after three trial. |
| **indicator for lock and unlock** | REQ\_ PO4\_DGELV \_HSI\_07\_V1 | The system will have two leds (green and red), the green one for unlock and the red one for lock. |
| **Username and password enter hardware** | REQ\_ PO4\_DGELV \_HSI\_08\_V1 | The system will have keypad to enter this name and password |

2.3 Block Diagram:



3. Hardware connection:

|  |  |  |
| --- | --- | --- |
| **Feature** | **REQ\_ID** | **Description** |
| **AVR** | REQ\_ PO4\_DGELV \_HSI\_01\_V1 | * AVR have four PORTS and every PORT has eight pin DIO. |
| **DC Motor** | REQ\_ PO4\_DGELV \_HSI\_02\_V1 | * The DC motor will connect to PORTB and use one pin.   **#output Pins** |
| **Up/down push buttons** | REQ\_ PO4\_DGELV \_HSI\_03\_V1 | * The two push buttons will connect to PORTB and use two pins.   **#input Pins** |
| **ON/OFF push buttons** | REQ\_ PO4\_DGELV \_HSI\_04\_V1 | * The two push buttons will connect to PORTB and use two pins.   **#input Pins** |
| **CLCD** | REQ\_ PO4\_DGELV \_HSI\_05\_V1 | * CLCD will connected PORTD and use four pin for sending data and three pin for controlling the data.   **#output Pins** |
| **buzzer** | REQ\_ PO4\_DGELV \_HSI\_06\_V1 | * Buzzer will connect to PORTB and use one pin.   **#output Pins** |
| **Two leds** | REQ\_ PO4\_DGELV \_HSI\_07\_V1 | * The green and red led will connect to PORTB and use two pins.   **#output Pins** |
| **KEYPAD** | REQ\_ PO4\_DGELV \_HSI\_08\_V1 | * The KEYPD 4\*4 will connect to PORTC four pins as input and four pins as output. * The keypad has 16 buttons, the first 10 buttons is for entering number from 0-9 and the other 6 buttons for entering the alphabetic characters every button from those six bottoms represent from 4-5 character and by pressing the button we can change the change the characters.   **#input and output Pins** |