Elevator UART Embedded System
Embedded Assessment #1

Seif Salama – 202200639

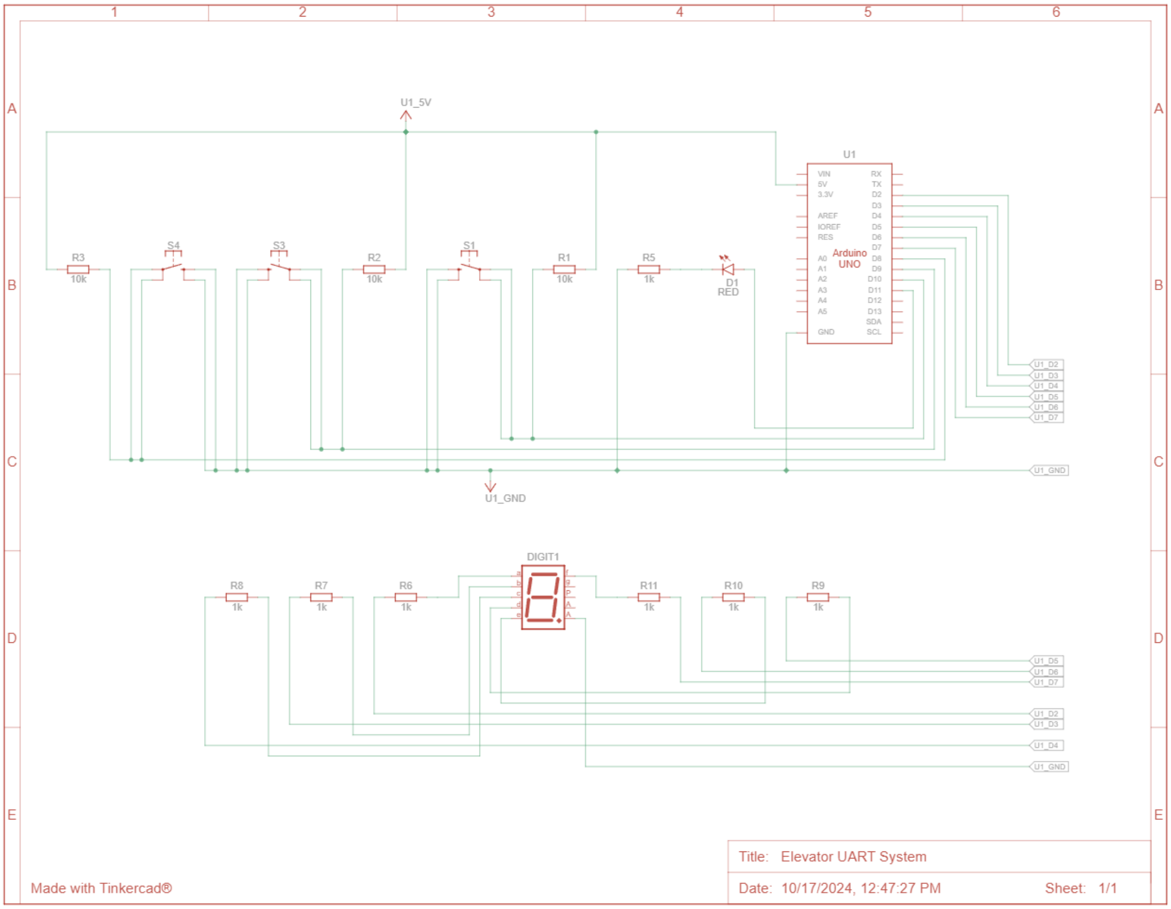
GitHub: Repository

Table of Contents

Introduction

The purpose of this project is to design a UART-controlled embedded system to simulate an elevator, where various components like buttons, a 7-segment display, a buzzer, and an LED are integrated. The system allows manual control over the elevator's functions, such as changing the floor display or responding to emergency button presses. The UART interface serves as the communication link for external commands, enabling precise control over system operations. This project not only highlights the use of UART in embedded systems but also models how manual approval can be used to ensure emergency procedures are properly handled.

System Design



The system is built around an ATmega328P microcontroller, which manages the core functionalities of the elevator simulation. The system is equipped with multiple buttons that represent floor selection and an emergency button. The output devices include a 7-segment display for showing the current floor, an LED for emergency signaling, and a buzzer for alert sounds. The UART interface acts as the communication bridge, allowing external commands to control and monitor the system's behavior. The buttons, display, and other peripherals are connected directly to the ATmega, and the microcontroller responds to commands sent through the UART interface by performing specific actions, such as turning on the LED or updating the display.

Digital Inputs:

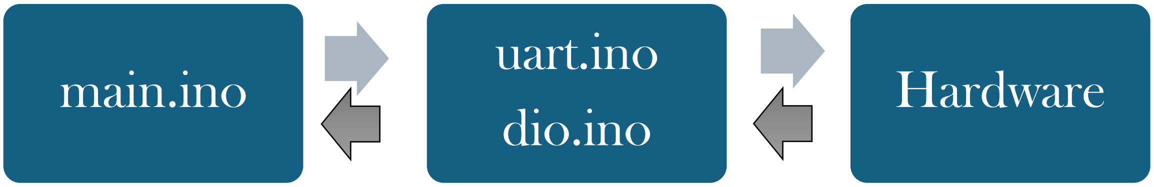
- Button 0 and Button 1: User selects desired floor of elevator
- Emergency Button: User declares an emergency

Digital Outputs:

- Red LED: Lights up when UART sends command after emergency button is pressed
- 7 Segment Display: displays current floor number based on UART command after button 0 or button 1 is pressed

Design Architecture

The system's architecture relies on the interaction between the Application Layer, Middleware, and Hardware components. Each layer communicates with the others to ensure the system functions smoothly and responds correctly.



1. Application Layer (main.ino):

Communicates with Middleware (dio.ino, uart.ino):

- The Application Layer sends commands to the DIO middleware to control the 7-segment display and LED by setting pin states.
- It receives incoming UART commands from the UART middleware through an RX interrupt. The Application Layer processes these commands and instructs the DIO middleware to take actions based on them.
- It sends feedback messages back to the UART middleware to confirm successful actions (e.g., “Command: LED ON”).

2. Middleware:

DIO (dio.ino):

- Communicates with the Application Layer by exposing functions to set pin directions and control the state of the pins connected to the 7-segment display and LED.
- Communicates with Hardware by directly controlling the state and direction of LED and 7-segment display.

UART (uart.ino):

- Communicates with the Application Layer through UART interrupts. When a command is received via UART, the UART middleware stores it in a buffer. The Application Layer reads this buffer to process the command.

- Communicates with external devices through the UART interface, sending and receiving commands like led_on, led_off, and displaying them via serial communication.

3. Hardware:

Communicates with Middleware (dio.ino):

The Buttons (floor selection and emergency) send input signals to the DIO middleware, which processes the button presses and sends to the Application Layer.

The 7-Segment Display and LED are controlled by the DIO middleware, which sets their corresponding GPIO pins HIGH or LOW based on instructions from the Application Layer.

Test Cases

Test Case	Process	Expected Outcome	Actual Outcome
Turning on LED through UART	Write “led_on” in serial monitor	LED turns on and “Command: Red LED ON” is printed	LED turns on and “Command: Red LED ON” is printed
Turning off LED through UART	Write “led_off” in serial monitor	LED turns off and “Command: Red LED OFF” is printed	LED turns off and “Command: Red LED OFF” is printed
Turning on/off LED or 7 Segment through misspelt command or upper-case letters	Write “led on” in serial monitor Write “LED_ON” in serial monitor Write “led_oh” in serial monitor	“Not a valid command” is printed	“Not a valid command” is printed
Press button x	Button is pressed once	“Button x Pressed” is printed	“Button x Pressed” is printed
Hold button x	Button is pressed for a long time	“Button x Pressed” is printed repeatedly after delay is over	“Button x Pressed” is printed repeatedly after delay is over
Press button x twice	Button is pressed twice	“Button x is Pressed” if printed twice	“Button x is Pressed” if printed twice
Displaying 0 on 7 Segment Display	Write “7Segment_0” in serial monitor	0 is displayed and “Command: 0 Displayed” is printed	0 is displayed and “Command: 0 Displayed” is printed
Displaying 1 on 7 Segment Display	Write “7Segment_1” in serial monitor	1 is displayed and “Command: 1 Displayed” is printed	1 is displayed and “Command: 1 Displayed” is printed

Bill of Materials

Item	Price (Total)	Quantity	Link
Arduino Uno R3 Clone	465	1	Arduino Uno Rev3 By UGE Electronics (uge-one.com)
Red LED	0.35	1	LED 3mm Super Bright Red Color WR | RAM Electronics website (ram-e-shop.com)
7 Segment Display	7.5	1	7 Segment 0.5" Common Cathode | RAM Electronics website (ram-e-shop.com)
Push Button	1.5 (4.5)	3	Press 4pin 12x12x8 mm Tack Switch Standard Stem | RAM Electronics website (ram-e-shop.com)
Resistor (1k Ohm)	0.5 (3.5)	7	Resistor 1 KOHM – Future Electronics Egypt (fut-electronics.com)
Resistor (10k Ohm)	0.5 (1.5)	3	Resistor 10 KOHM – Future Electronics Egypt (fut-electronics.com)
Breadboard	45	1	Breadboard – (fut-electronics.com)
	527.35 EGP		

Optimized Bill of Materials

Item	Description	Price (Total)	Quantity	Link
ATmega328P	Microcontroller	220	1	ATMega328P – (fut-electronics.com)
16 MHz Crystal Oscillator	UART Clock	4	1	Oscillator | RAM Electronics website
22pF Capacitors	Used with crystal oscillator	0.5 (1)	2	Capacitor 22pF – (uge-one.com)
5V Voltage Regulator	-	8	1	Voltage Regulator 7805 TO220 5V – 7809 (uge-one.com)
FTDI Adapter	USB to TTL – purchased once	150 - excluded	1	FTDI | RAM Electronics website
Red LED	-	0.35	1	LED Red Color | RAM Electronics website
7 Segment Display	-	7.5	1	7 Segment Common Cathode | RAM Electronics website
Push Button	-	1.5 (4.5)	3	Button | RAM Electronics website
Resistor (1k Ohm)	-	0.5 (3.5)	7	Resistor 1 KOHM – (fut-electronics.com)
Resistor (10k Ohm)	-	0.5 (1.5)	3	Resistor 10 KOHM – fut-electronics.com)
Breadboard	-	45	1	Breadboard – (fut-electronics.com)
	Total Excluding FTDI	295.35 EGP		

Conclusion

In summary, this project successfully integrates UART communication with an embedded elevator system, enabling efficient control of key components like the LED, buzzer, and display. The system's design has been optimized using an ATmega328P, making it more affordable and scalable. Through UART commands, users can manage various functions manually, ensuring that emergency responses are handled effectively. This setup simulates a practical industrial scenario, emphasizing manual intervention for safety in emergencies. The implemented test cases verified that the system operates as expected and has the potential for future improvements.

Appendix

GitHub Repository: https://github.com/sseifsalama/Elevator-UART-Embedded

Demo Video: Project Demo - Seif Salama 202200639.mp4

Demo Video URL Link: https://elsewedyedu1-my.sharepoint.com/:v:/g/personal/ss2200639_tkh_edu_eg/EY8BpedC3XtDqJW3Q5G5fscBYKmS1M2sHCE-lMwb2etb7g?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOiJPbmVEcml2ZUZvckJ1c2luZXNzIiwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYiIsInJlZmVycmFsTW9kZSI6InZpZXciLCJyZWZlcnJhbFZpZXciOiJNeUZpbGVzTGlua0NvcHkifX0&e=2g6UYe