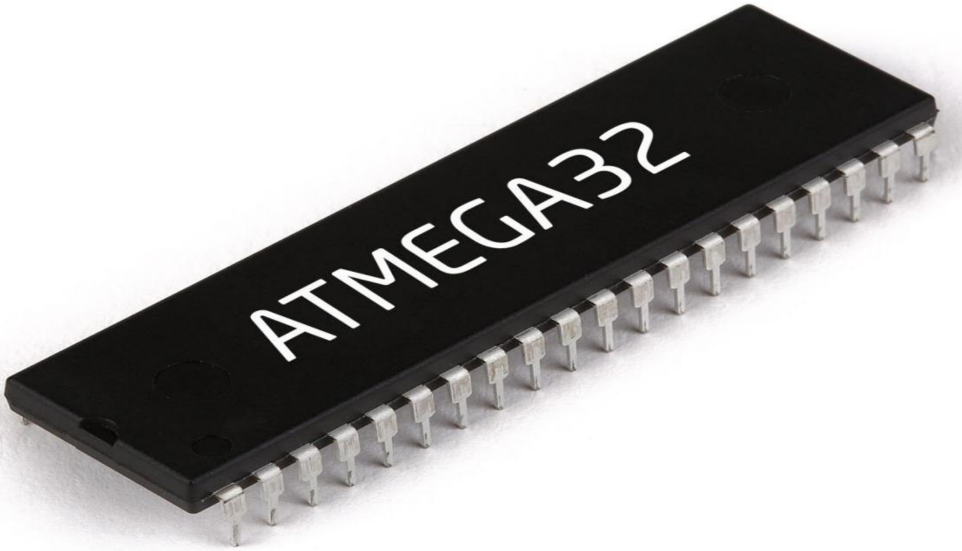
Motor Speed controller project

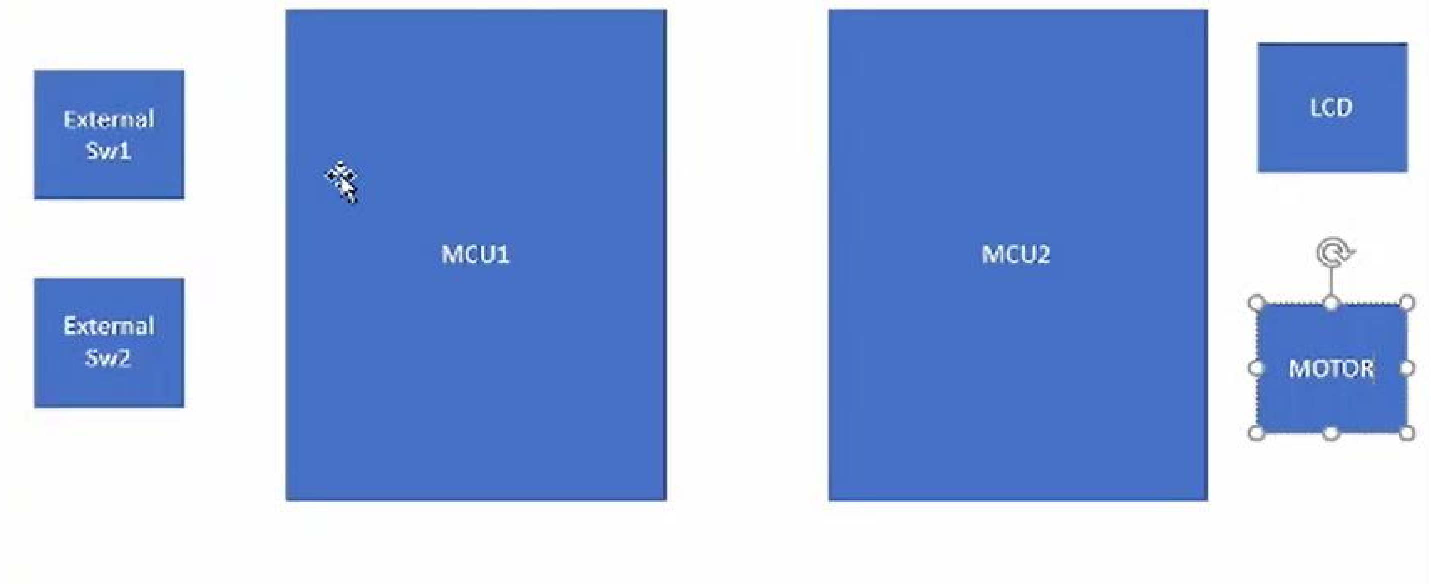


Team four members :

1. Ahmed Hany
2. Abdulrahman Badr
3. Ghada Ragab
4. Mahmoud Abd Elwahab
5. Maryham Melad Gerges
6. Mostafa Mamdouh

Contents

Brief description:



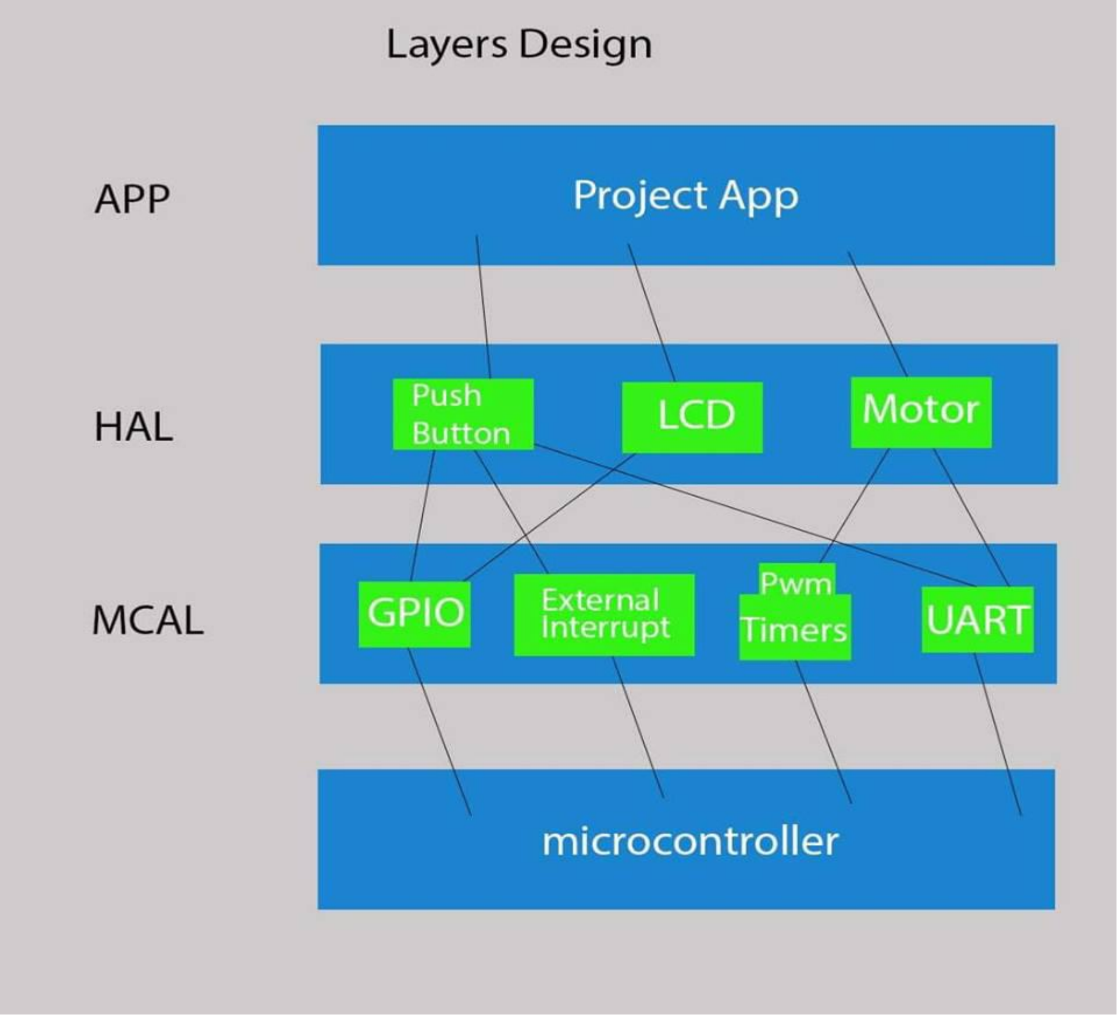
Our project consists of:

- **External Sw1:** is responsible for incrementing the speed.
- **External Sw2:** is responsible for decrementing the speed.

These switches are connected to the first microcontroller (MCU1).

- **MCU1:** is responsible for saving the input speeds given by the two external switches then it will transfer them to the second microcontroller (MCU2).
- **MCU2:** is responsible for updating the motor speed after taking all the speed data from MCU1.
- **LCD :** is responsible for representing the current motor speed on the screen.
- **Motor:** The DC motor can be controlled to increase or decrease via a switch.

Layers Design:



MCAL layer :

- **GPIO:** **driver responsible for input/outputs Microcontroller pins.**
- **External interrupt:** **it handles any external interrupt (such as external switches or push button interrupt).**
- **Timers:** **The time is calculated by the frequency of the CPU through the timer unit, and it is used in the delay calculation.**
- **PWM**: **PWM is a method through which we can generate variable voltage by turning on and off the power that's going to the electronic device at a fast rate. The average voltage depends on the duty cycle of the signal, or the amount of**

time the signal is ON versus the amount of time the signal is OFF in a single period of time.
So its implemented functions include setting certain duty cycle needed to set certain value for the motor speed.

- UART : handles the communication between the two microcontrollers.(MCU1 &MCU2)

HAL layer :

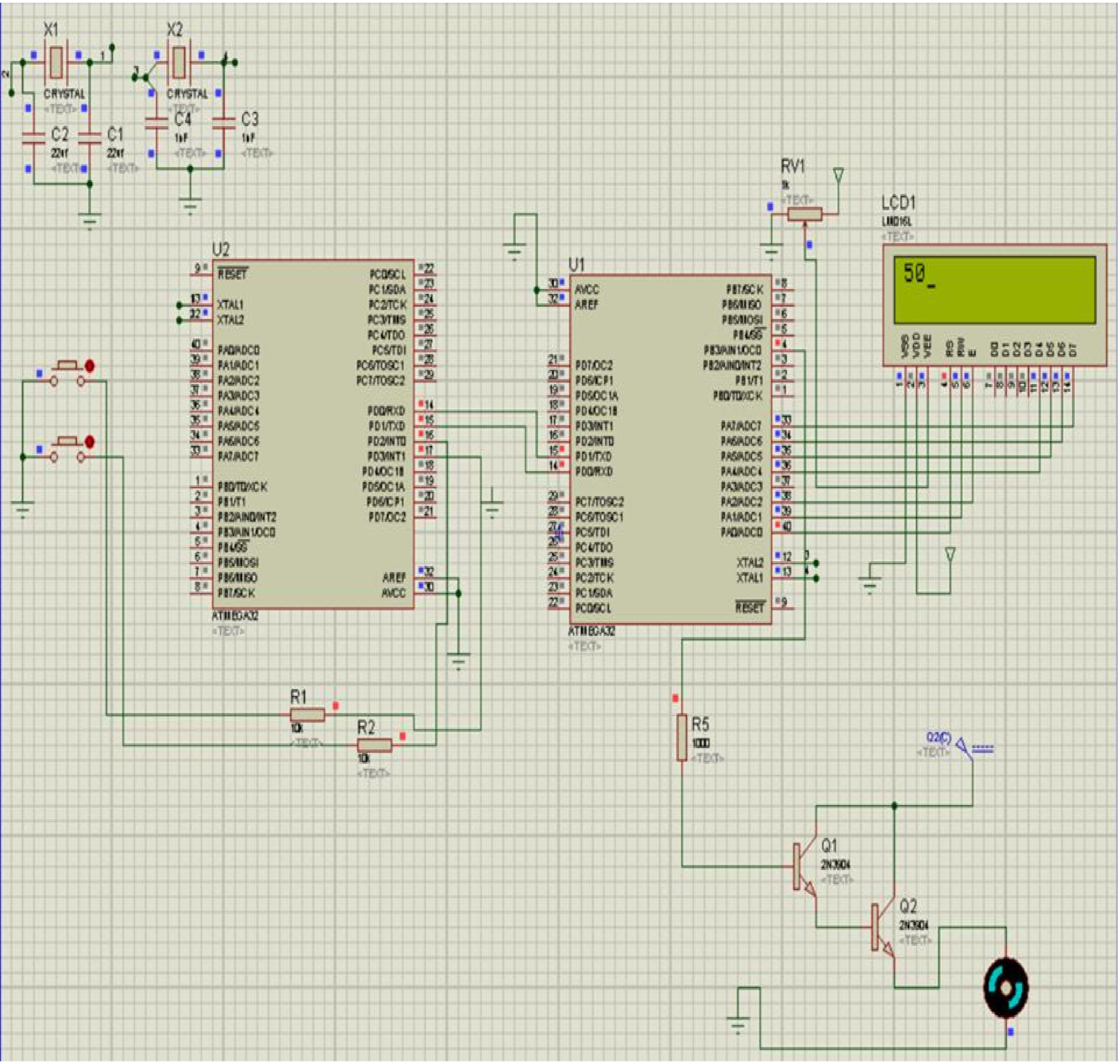
- LCD: driver handles showing the current motor speed on the screen.
- Push button: it generates interrupt to be handled by CPU by changing the speed value and transfer it to the UART.

Motor :We receive the value of the speed of the motor in the second MCU2 and the motor operates with it.

APP layer:

It includes how project is handled in main.

Detailed Proteus Design



How we handled corner cases

1-Speed limits from zero% to 100%

array has the values of the Duty Cycle to run the motor at speed rates from 0 to 100. We did index by passing it to array and not just the last element or the first element in Functions speed up and down .

Code in function speedup because it is not exceeds the maximum speed

```
        if(index >MaxNumSpeed)
        index=MaxNumSpeed;
```

Code in function slow down because it is not exceeds the Minimum speed

```
    if(index<MinNumSpeed)
        index=MinNumSpeed;
```

2-If there were two interrupt, they cut each other

I configure various interrupts and two or more interrupts happen at the same time. it depends on the interrupt priority. For AVR architecture it is simple. The lower the vector address, the higher the priority. Have a look again RESET has the highest priority as might expect and other units later.

Consequently, if there are two interrupts , int0 is carried out with int1, then the int0 will be executed, and then int1.It will appear in reality, but the simulation will not appear.may be It can appear in the simulation or do our daily time in interrupt zero .

Team member role during the project :

1. **Ahmed Hany :** Lcd-Motor-External interrupt-UART-main1-main2

2. **Abdulrahman Badr:** UART-main1-main2-GPIO-Layers Design

3. **Ghada Ragab:** GPIO-Timers-main1

4. **Mahmoud Abd Elwahab:** GPIO-External interrupt-main1-main2-Protuess-Layers Design

5. **Maryham Melad Gerges:**Timers-Presentation Project

6. **Mostafa Mamdouh :** GPIO