Digital Systems II

Lab1 : ALU

By:

Ahmad Abdallah Waheeb (04)

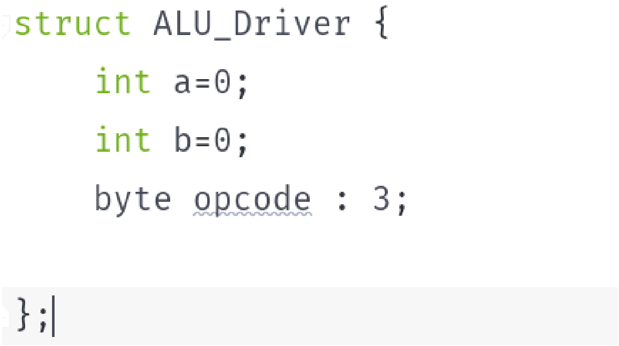
Ahmad Mohamed El-Zeny (06)

• Problem Statement:

Using Arduino along with a 16 X 2 LCD Keypad Shield to implement a simple ALU that supports multiple operations.

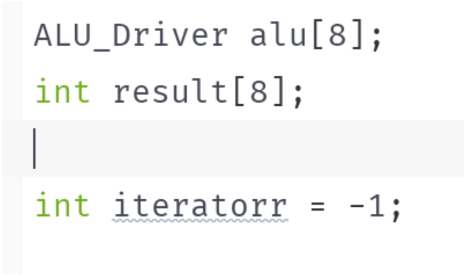
• Design:

We implemented two auxiliary modules. The first module is (ALU Driver) module. it stores different sets of values for (A, B, opCode).



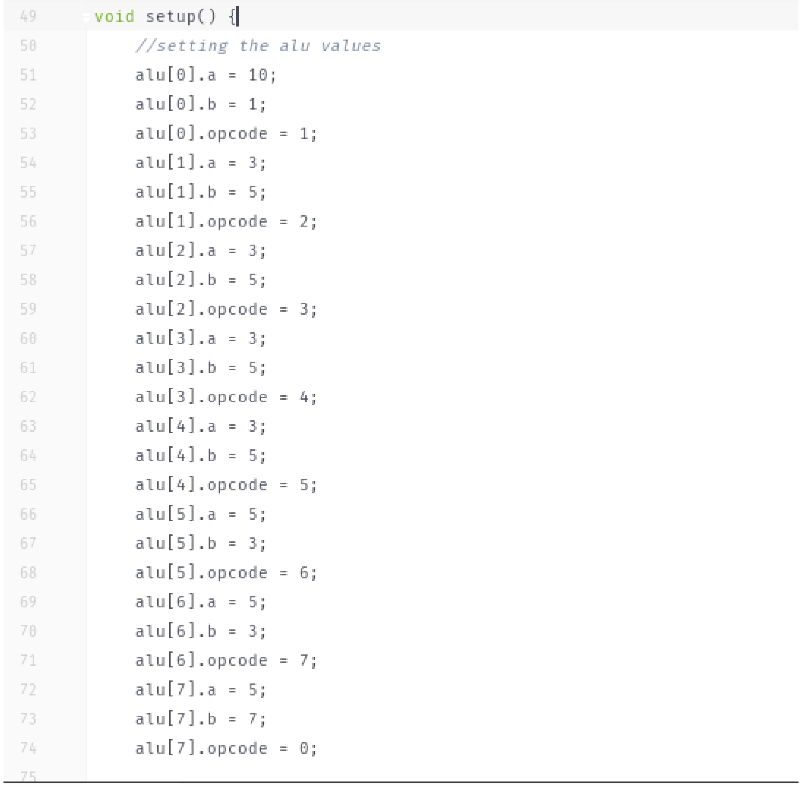
```
struct ALU_Driver {
    int a=0;
    int b=0;
    byte opcode : 3;

};
```

The second module (Result Array) stores the ALU result for the equivalent set of inputs
given by the ALU Driver.



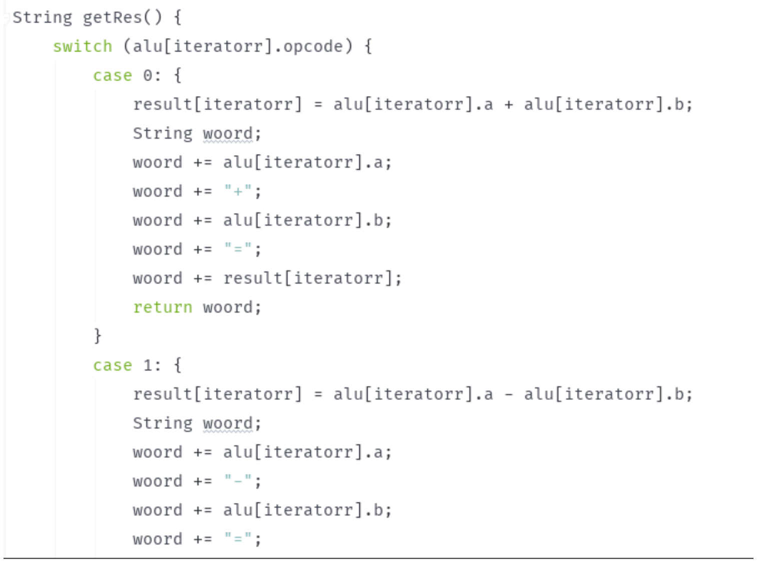
```
ALU_Driver alu[8];
int result[8];

int iteratorr = -1;
```

We added 8 different sets of Values (A,B,opCode) one for each operation.



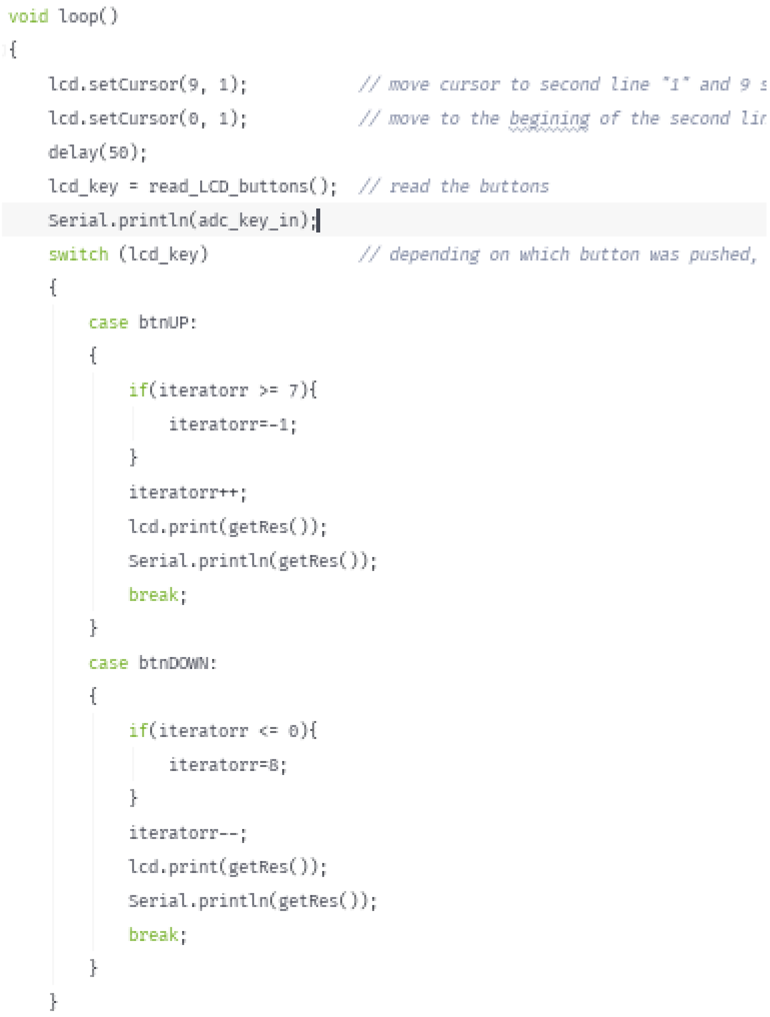
```
void setup() {
    //setting the alu values
    alu[0].a = 10;
    alu[0].b = 1;
    alu[0].opcode = 1;
    alu[1].a = 3;
    alu[1].b = 5;
    alu[1].opcode = 2;
    alu[2].a = 3;
    alu[2].b = 5;
    alu[2].opcode = 3;
    alu[3].a = 3;
    alu[3].b = 5;
    alu[3].opcode = 4;
    alu[4].a = 3;
    alu[4].b = 5;
    alu[4].opcode = 5;
    alu[5].a = 5;
    alu[5].b = 3;
    alu[5].opcode = 6;
    alu[6].a = 5;
    alu[6].b = 3;
    alu[6].opcode = 7;
    alu[7].a = 5;
    alu[7].b = 7;
    alu[7].opcode = 0;
```

We set up a function that performs different operations depending on the opCode value then stores the result at the results array.



```
String getRes() {
    switch (alu[iteratorr].opcode) {
        case 0: {
            result[iteratorr] = alu[iteratorr].a + alu[iteratorr].b;
            String woord;
            woord += alu[iteratorr].a;
            woord += "+";
            woord += alu[iteratorr].b;
            woord += "=";
            woord += result[iteratorr];
            return woord;
        }
        case 1: {
            result[iteratorr] = alu[iteratorr].a - alu[iteratorr].b;
            String woord;
            woord += alu[iteratorr].a;
            woord += "-";
            woord += alu[iteratorr].b;
            woord += "=";
```

Finally, at the loop we iterate on the results array to return the performed operation through the getRes() function.



```
void loop()
{
    lcd.setCursor(9, 1);           // move cursor to second line "1" and 9 s
    lcd.setCursor(0, 1);           // move to the begining of the second lir
    delay(50);
    lcd_key = read_LCD_buttons();  // read the buttons
    Serial.println(adc_key_in);
    switch (lcd_key)               // depending on which button was pushed,
    {
        case btnUP:
        {
            if(iteratorr >= 7){
                iteratorr=-1;
            }
            iteratorr++;
            lcd.print(getRes());
            Serial.println(getRes());
            break;
        }
        case btnDOWN:
        {
            if(iteratorr <= 0){
                iteratorr=8;
            }
            iteratorr--;
            lcd.print(getRes());
            Serial.println(getRes());
            break;
        }
    }
```