

# Thực hành Lập Trình Nhúng Căn Bản

## Báo Cáo Lab 04

Thành viên nhóm:

Trần Thanh Duy 16520308

Lương Quốc Hải 16520327

Phan Thanh Duy 14521199

Viết chương trình thực hiện chức năng lưu và xóa EEPROM, hiển thị thông tin lên LCD;

```
char numbertochar(int8_t num)
{
    switch(num)
    {
        case 1: return 'a';
        case 2: return 'b';
        case 3: return 'c';
        case 4: return 'd';
        case 5: return 'e';
        case 6: return 'f';
        case 7: return 'g';
        case 8: return 'h';
        case 9: return 'i';
        case 0: return 'q';
        default: return 'q';
    }
}
```

Hàm này dùng để chuyển đổi số thứ tự phím bấm thành ký tự. Từ 1 tới 9 tương ứng với từ A tới Q. Giá trị mặc định của nó sẽ là Q.

```
print_lcd(1, "2-Doc EEPROM");
print_lcd(0, "1-Ghi EEPROM");

key = numbertochar(Scankey());
// scan keypad to input

if(key == 'a')
{
    while(key=='a')
    {
        key = numbertochar(Scankey());
        clr_all_panel();
        print_lcd(0, "Insert Data");
    }
}
```

Đầu tiên là hiển thị ra LCD 2 lựa chọn để người dùng thấy được.

Tiếp theo ta đọc keypad từ người dùng để đưa ra quyết định.

Sau khi người dùng nhấn keypad phím 1 thì sẽ vào chương trình ghi EEPROM.

Vòng while ở đây để giá trị được chính xác hơn.

```

int count = 0;
char temp[8];
while(count<8)
{
    key = numbertochar(Scankey());
    if(key!='q')
    {
        temp[count++] = key;

        print_lcd(2,temp);
        while(key!='q')
            key = numbertochar(Scankey());
    }
}

```

Nhận 8 chữ cái và in ra màn hình mỗi lần nhấn nút. Bao gồm một mảng 8 phần tử và chạy 1 vòng lặp để nhập các phần tử đó.

Nhấn nút thì sẽ chuyển tín hiệu nhấn thành chữ như hàm chuyển ở trên đã viết. Sau đó lưu chữ cái vào mảng, sau đó in chữ cái ra LCD.

Tiếp tục nhận tín hiệu từ keypad và thực hiện vòng lặp.

```

if(key == 'b')
{
    clr_all_panel();
    while(key=='b')
    {
        key = numbertochar(Scankey());
        print_lcd(0, "Out put data: ");
        char temp[8];
        if(Read_24LC64(0x00000000+4)=='q')
        {
            print_lcd(1,"EEPROM is empty");
        }
        else
        {
            for(int i=0;i<8;i++)
            {
                temp[i]= Read_24LC64(0x00000000+i+4);
            }

            print_lcd(2,temp);
        }
        for(int i=0;i<8;i++)
        {
            temp[i]= 0;
        }
        while(key=='q')
        {
            key = numbertochar(Scankey());
            clr_all_panel();
        }
        DrvSYS_Delay(5000);          // delay
    }
}

```

Lựa chọn 2 thì sẽ in ra màn hình giá trị đã được lưu trong EEPROM. Hàm `Read_24LC64(*address*)` dùng để đọc giá trị ở địa chỉ. Nếu đọc được giá trị 'q' tức là chưa nhận thì sẽ in ra màn hình "EEPROM is empty". Hoặc sẽ in ra tất cả giá trị đã ghi vào.

Hàm

```
void DrvI2C_Ctrl(E_I2C_PORT port, uint8_t start, uint8_t stop, uint8_t intFlag, uint8_t ack)
```

Để chỉnh control bit của I2C. intFlag là biến điều chỉnh chuyển trạng thái. Mỗi lần chuyển trạng thái thì hardware clear bit này. Để có thể thực hiện hành động kế tiếp ta dùng software (KeilC) set bit lên sau khi thực hiện một hành động.