

بسم الله الرحمن الرحيم

گزارش پروژه سوم سیستم های نهفته

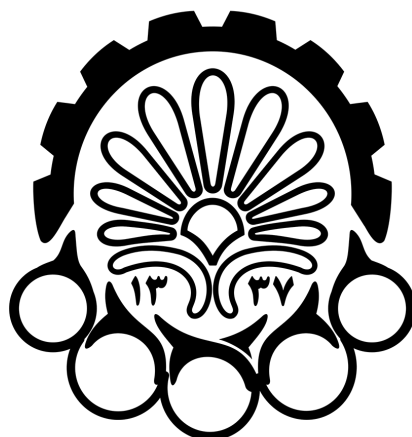
پاییز ۹۷

امیرحسین عباسی
محمدرضا حسینی

۹۳۳۱۰۰۸-۹۳۳۱۰۷۰

دانشگاه امیرکبیر

دانشکده کامپیوتر و فناوری اطلاعات

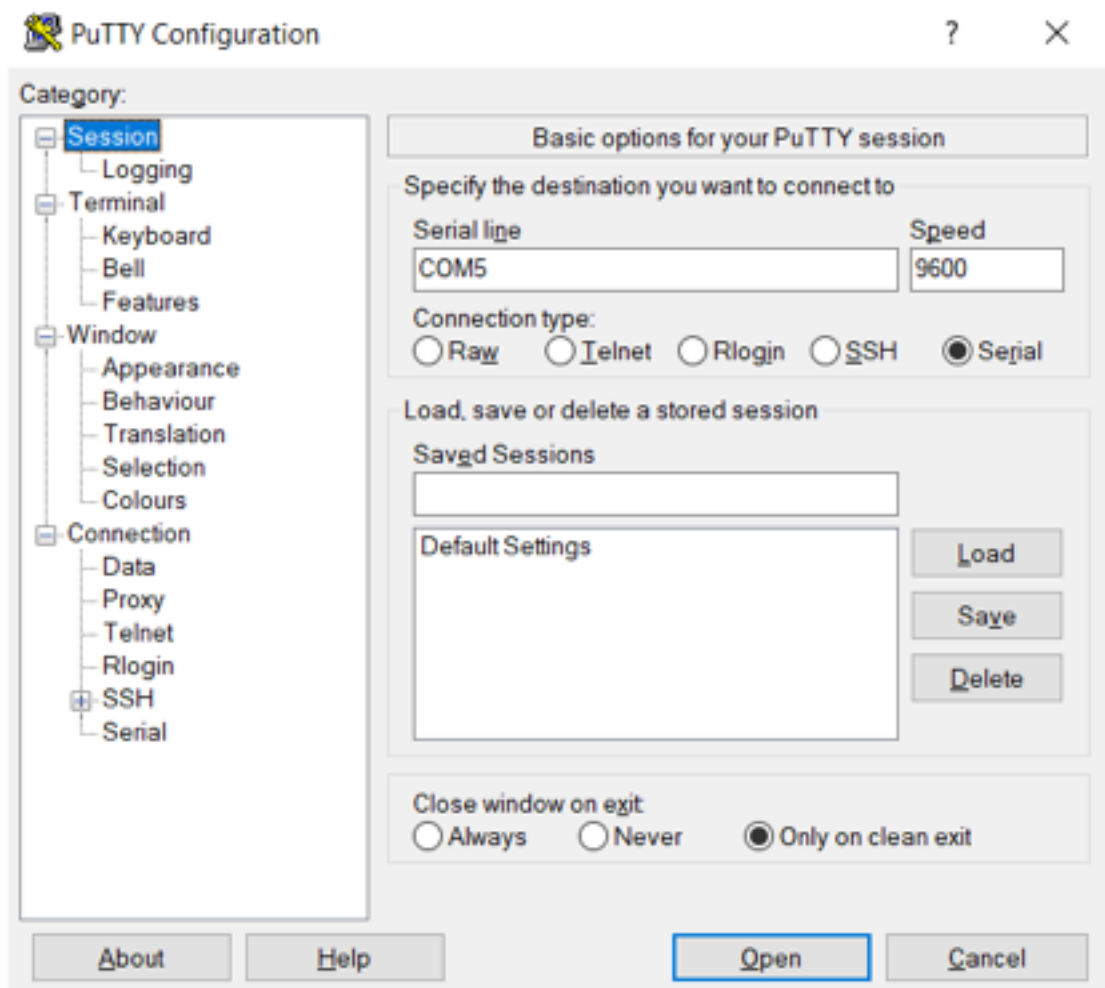


دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

هدف از این پروژه اتصال برد KL25Z به کامپیوتر به منظور برقراری ارتباط با آن بود .

در این پروژه از ترمینال Putty استفاده شد که در تنظیمات آن اتصال به صورت serial تنظیم شد و line مورد نظر نیز COM5 بود :



اتصالات اجزای مدار و تنظیمات اولیه به شرح زیر است :

```
PwmOut audioOut(PTB0);  
PwmOut Led(LED1);  
InterruptIn B_1(PTD0);  
InterruptIn B_2(PTA5);  
Serial pc(USBTX, USBRX);  
AnalogIn temp(PTE30);
```

پس از آن کاراکترهای ورودی مورد نظر را در آرایه ذخیره کردیم :

```
printf ("\n\n -$-> Enter your command: ");  
char input[20]="";  
scanf("%s", input);  
checkInput(input);
```

سپس با تابع checkInput با توجه به ورودی مورد نظر ، عملیات لازم برای هر مورد صورت میگیرد :

```
void checkInput(char input[]){  
    char str1[2] = {'o' , 'n'};  
    char str2[3] = {'o' , 'f' , 'f'};  
    char str3[4] = {'t' , 'e' , 'm' , 'p'};  
    char str4[4] = {'p' , 'l' , 'a' , 'y'};  
    bool isSame;  
    isSame = true;  
    for(int i=0;i<3;i++){  
        if(input[i]!=str2[i]){  
            isSame=false;  
            break;  
        }  
    }  
    if(isSame){  
        Led = 1 ;  
        printf ("\n\n - off");  
    }  
    isSame = true;  
    for(int i=0;i<2;i++){  
        if(input[i]!=str1[i]){  
            isSame=false;  
        }  
    }  
}
```

```
        break;
    }
}
if(isSame){
    Led = 0;
    printf ("\n\n - on");
}
isSame = true;
for(int i=0;i<4;i++){
    if(input[i]!=str3[i]){
        isSame=false;
        break;
    }
}
if(isSame){
    getTemp();
    printf ("\n\n - temp");
}
isSame = true;
for(int i=0;i<4;i++){
    if(input[i]!=str4[i]){
        isSame=false;
        break;
    }
}
if(isSame){
    playSong();
    printf ("\n\n - play");
}
}
```

برای ورودی on مقدار led برابر با 1 میشود و برای ورودی off مقدار آن صفر میشود .

برای ورودی temp تابع getTemp صدا میشود که در هر 5 میلی ثانیه یک نمونه از سنسور دما را ذخیره میکند و میانگین 30000 نمونه را محاسبه و چاپ میکند :

```
void getTemp() {  
    float temperature=0,sum=0,avg=0;  
    int i;  
    for(i=0;i<30000;i++){  
        sum += temp.read();  
        wait(0.005);  
    }  
    avg = sum/(30000);  
    temperature=(avg*3.685503686*30000);  
    printf ("\n\n - temp: %f",temperature);  
}
```

برای ورودی play تابع playSong صدا شده و با توجه به نت ها و زمانبندی انجام شده ، با استفاده از buzzer ملودی مورد نظر پخش میشود :

```
void playSong() {  
    float duration[20]=  
{0.4,0.4,0.4,0.4,0.8,0.4,0.4,0.4,0.4,0.4,0.4,0.8,0.4,0.4,0.2,0.2,0.4,0.4,0.8};  
    float note[20]=  
{440,440,440,466.16,440,440,466.16,440,440,440,466.16,440,440,466.16,440,392,440,466.16,440,392};  
    float vol = 0.25;  
    int len = 20;  
    PwmOut audioOut(PTB0);  
    for(int i=0;i<len;i++){  
        audioOut.period(1.0/note[i]);  
        audioOut = vol;  
        wait(duration[i]);  
        audioOut = 0;  
    }  
}
```

```
}  
}
```

برای اولویت بندی جهت فشردن هر یک از button ها از تنظیمات NVIC استفاده شده و همچنین کلیدها به صورت pull up و حساس به لبه بالارونده تنظیم شده ان که به شرح زیر است :

```
B_1.mode(PullUp) ;  
B_2.mode(PullUp) ;  
B_1.rise(&int_1);  
B_2.rise(&int_2);  
NVIC_SetPriority(PORTA_IRQn, 0);  
NVIC_SetPriority(PORTD_IRQn, 1);
```

پس از فشردن هر یک از کلیدها تابع مربوط به آن صدا شده و خروجیهای مورد نظر چاپ شده و تاخیرهای لازم اجرا میشود :

```
void int_1(){  
    printf("User interrupt 1 has been received \n");  
    wait(5);  
    printf("intrupt1 is finished \n");  
}  
void int_2(){  
    printf("User interrupt 2 has been received \n");  
    wait(5);  
    printf("intrupt2 is finished \n");  
}
```