

گزارش پروژه آخر درس سیستم های نهفته و بی درنگ

تینا صداقت ۹۳۳۱۰۴۴

۵ thread برای این پروژه تعریف می کنیم که در ادامه در مورد هر کدام توضیح می دهیم:

یک thread برای task1 تعریف می شود که وظیفه ی آپدیت کردن زمان هر 0.1 ثانیه را برعهده دارد. باید thread های مربوط به task های ۲ و ۴ در این thread، به ترتیب هر ۵ ثانیه یک بار و هر ۱ ثانیه یک بار صدا زده شوند که این کار به کمک یک counter به نام count انجام شده است. چون هر ۱ ثانیه به معنای ۱۰ بار اجرا شدن task1 و هر ۵ ثانیه به معنای ۵۰ بار اجرا شدن task1 است، به کمک باقیمانده گرفتن مقدار count بر ۱۰ و ۵۰ می توان event مربوط به thread مورد نظر را فعال کرد.

```
1. void updateTime() { //task1
2.     timer1+=0.1;
3.     count++;
4.     //printf("timer: %f \n",timer1); updating but not showing
5.     if(count%10==0){
6.         t4.start(updateAndDisplayTime);
7.     }
8.     if(count%50==0){
9.         t2.start(readTemperature);
10.    }
11. }
```

برای task4 که باید هر ۱ ثانیه یک بار زمان را آپدیت کند و نمایش دهد و push button های مربوط به نمایش دادن مقدار ها در این بخش تعریف می شوند هم باید یک thread تعریف کرد. کد مربوط به push button مانند پروژه ۰ نوشته شده است.

```
1. void updateAndDisplayTime(){ //task4
2.     timer4+=1;
3.     printf("timer: %f \n",timer4);
4.
5.
6.
7.     new_in = SW1;
8.     if(new_in==1 && old_in == 0 ){
9.         printf("Temperature (in Celsius) is %4.2f \r\n",T);
10.        printf("associated time is %4.2f \r\n", timer4);
11.    }
12.    old_in=new_in;
13.
14.    new_in2 = SW2;
15.    if(new_in2==1 && old_in2 == 0 ){
16.        printf("Average Temperature (in Celsius) is %4.2f \r\n",average);
17.    }
18.    old_in2=new_in2;
19.
20.
21. }
```

برای task2 که باید مقدار دما را از سنسور دما (بخش امتیازی) بخواند یک thread داریم که در task1 هر ۵ ثانیه یک بار اجرا می شود. در این task باید task3 هم اجرا شود.

```
1. void readTemperature(){ //task2
2.     T=T1*K;
3.     temps[index]=T;
```

```

4.
5.     if(index==32){
6.         index=0;
7.         test=1;
8.     }
9.     //printf("Temperature (in Celsius) is %4.2f \r\n",T);
10.    t3.start(getAverage);
11.    index++;
12. }

```

Task3 هم یک thread است که مقادیر خوانده شده از سنسور دما را در یک آرایه قرار می دهد و میانگین حساب میکند.

برای کنترل کردن اینکه اگر کمتر از ۳۲ مقدار خوانده شده باشد میانگین درست حساب شود، یک flag به نام test تعریف کردیم که ابتدا مقدارش ۰ و تا زمانی که index آرایه به ۳۲ نرسد، ۰ باقی می ماند و برای محاسبه میانگین باید جمع دما ها تقسیم بر index آرایه شود. اما بعد از اینکه ۳۲ مقدار خوانده شد برای محاسبه میانگین باید جمع دما ها تقسیم بر همان عدد ۳۲ شود.

```

1. void getAverage(){//task3
2.     for(int i=0;i<32;i++){
3.         average+=temps[i];
4.     }
5.     if (test==0){
6.         average/=index;
7.     }
8.     else if (test ==1){
9.         average/=32;
10.    }
11.    //printf("average: %4.2f \n",average);
12. }

```

بخش امتیازی مربوط به خواندن دما از سنسور دما انجام شده است و LCD هم برای آن فعال شده است.