

بسمه تعالی

تمرین دوم درس سیستم های نهفته و بیدرنگ - محمدامیر سالاری - ۹۳۳۱۰۳۷

تشریح مسئله:

این احتمال وجود دارد که وقتی روز و ساعت از روی داده‌ی یک ثانیه‌ی قبل خوانده شده است، مقادیر در روال اینتراپت تغییر کنند و با مراجعه دوباره به برنامه‌ی GetDateTime مقادیر دقیقه و ثانیه را از مقادیر جدید بخوانیم و در نتیجه ساعت خوانده شده اشتباه خواهد بود.

راه حل ۱:

در صورت آمدن اینتراپت یک ثانیه Flag یک پرچم را تنظیم می‌کنیم و اجرای برنامه‌ی گرفتن تاریخ و زمان را ادامه می‌دهیم و پس از آن اگر Flag ست شده بود یک بار دیگر روال آپدیت زمان‌ها را انجام می‌دهیم. البته می‌توان به جای Flag اینتراپت از آرایه هم استفاده کرد تا بتوانیم تا چند بار آمدن اینتراپت را ذخیره کنیم. فقط باید این تابع را در برنامه‌ی main و در یک حلقه‌ی بی‌نهایت تکرار کنیم.

```
1 void DateTimeISR(void){
2     interruptFlag = true;
3 }
4 void update(void){
5     TimerVal.second++;
6     if(TimerVal.second > 59){
7         TimerVal.second = 0;
8         TimerVal.minute++;
9         if(TimerVal.minute > 59){
10             TimerVal.minute = 0;
11             TimerVal.hour++;
12             if(TimerVal.hour > 23){
13                 TimerVal.hour = 0;
14                 TimerVal.day++;
15             }
16         }
17     }
18     interruptFlag = false;
19 }
20
21 void GetDateTime(DateTimeType *DT){
22     DT.day = TimerVal.day;
23     DT.hour = TimerVal.hour;
24     DT.minute = TimerVal.minute;
25     DT.second = TimerVal.second;
26     if(interruptFlag == true){
27         update();
28     }
29 }
30 void main(){
31     ...
32     if(interruptFlag == true)
33         update();
34     ...
35 }
```

مزیت این روش این است که زمانی در استراکت DT قرار می‌گیرد هیچ گاه مقدار اشتباهی نخواهد بود و امکان ندارد تعدادی از فیلدها مربوط به یک ثانیه قبل و تعدادی مربوط به اکنون باشند. اما همواره ممکن است تابع گرفتن تاریخ و زمان به اندازه‌ی کمتر یا مساوی یک ثانیه از مقدار واقعی عقب باشد.

راه حل ۲:

یک متغیر global از نوع struct مربوط به زمان می‌سازیم و در تابع روال اینتراپت مقادیر این متغیر گلوبال را تغییر می‌دهیم. حال در داخل تابع GetDateTime به جای ریختن مقادیر موجود در فیلدهای TimerVal در فیلدهای استراکتی که DT (پوینتر ورودی تابع) به آن اشاره می‌کند تنها آدرس TimerVal را داخل مقدار DT می‌ریزیم. یعنی DT در یک دستور به TimerVal اشاره خواهد کرد پس مشکل تغییر مقادیر در حین اجرای تابع GetDateTime به وجود نخواهد آمد.

کد:

```
1  DateTime TimerVal;
2  void DateTimeISR(void) {
3      TimerVal.second++;
4      if(TimerVal.second > 59) {
5          TimerVal.second = 0;
6          TimerVal.minute++;
7          if(TimerVal.minute > 59) {
8              TimerVal.minute = 0;
9              TimerVal.hour++;
10             if(TimerVal.hour > 23) {
11                 TimerVal.hour = 0;
12                 TimerVal.day++;
13             }
14         }
15     }
16 }
17
18 void GetDateTime(DateTime *DT) {
19     DT = &TimerVal;
20 }
```

راه حل ۳:

چون نمی‌خواهیم زمان صحیح را از دست بدهیم پس وقتی اینتراپت یک ثانیه آمد نمی‌توانیم تابع گرفتن تاریخ و زمان را ادامه بدهیم و حتما باید ابتدا روتین سرویس اینتراپت را انجام بدهیم. بنابراین به دو متغیر باینری برای صحبت کردن روتین سرویس و تابع گرفتن زمان و تاریخ نیاز پیدا می‌کنیم:

- متغیر getDateTimelsRunning که تابع گرفتن تاریخ و زمان در ابتدای اجرای خود آن را یک و در انتهای اجرای خود دوباره صفر می‌کند. مقدار اولیه‌ی این متغیر صفر است.
- متغیر resetYourRunning که در انتهای روتین اینتراپت اگر متغیر getDateTimelsRunning یک باشد یک می‌شود و در آخر تابع گرفتن تاریخ و زمان چک می‌شود و اگر یک بود تابع گرفتن تاریخ و زمان دوباره خودش را صدا می‌زند تا فیلدها درست مقدار دهی شوند.

```

1 void DateTimeISR(void) {
2     TimerVal.second++;
3     if(TimerVal.second > 59){
4         TimerVal.second = 0;
5         TimerVal.minute++;
6         if(TimerVal.minute > 59){
7             TimerVal.minute = 0;
8             TimerVal.hour++;
9             if(TimerVal.hour > 23){
10                TimerVal.hour = 0;
11                TimerVal.day++;
12            }
13        }
14    }
15    if(getDateTimeIsRunning == true){
16        resetYourRunning = true;
17    }
18 }

23 void GetDateTime(DateTimeType *DT){
24     getDateTimeIsRunning = true;
25     DT.day = TimerVal.day;
26     DT.hour = TimerVal.hour;
27     DT.minute = TimerVal.minute;
28     DT.second = TimerVal.second;
29     if(resetYourRunning = true){
30         getDateTimeIsRunning = false;
31         resetYourRunning = false;
32         GetDateTime(DT);
33     }
34     getDateTimeIsRunning = false;
35 }

```