**آزمایش 5**

**تایمر**

**1. چگونه میتوان 4 رقم متفاوت را همزمان بر روی 7segment نمایش داد؟**

مقدار تأخیر تایمر را بر روی یک عدد بسیار کوچک، مثلا 1ms تنظیم می‌کنیم و در هر بار اجرای تابع تایمر، تنها یکی از ارقام عدد مورد نظر را نمایش می‌دهیم. چون تغییر رقم به سرعت اتفاق می‌افتد، چشم انسان قادر به تشخیص تغییرات نیست و همه ارقام متفاوت را به صورت ثابت در کنار هم می‌بیند.

**2. در تنظیمات اولیه، تایمر را چگونه config کردید؟**

در پنجره Code Wizard از منوی سمت چپ، Timers/Counters را انتخاب می‌کنیم.

مقدار Period را روی 1 میلی‌ثانیه قرار می‌دهیم.

روی Apply کلیک می‌کنیم تا تنظیمات اعمال شود.

**3. در تنظیمات اولیه، پورت B و D و INT0 و INT1 را چگونه config کردید؟**

مقادیر 0 تا 7 پورت B را بر روی output تنظیم می‌کنیم.

مقادیر 4 تا 7 پورت D را بر روی output تنظیم می‌کنیم.

وقفه‌های 0 و 1 را فعال کرده و روی raising edge قرار می‌دهیم.

**4. در کد برنامه، زمان 1 ثانیه را چگونه اندازه گرفتید؟**

چون تایمر را روی 1ms کانفیگ کردیم، با استفاده از متغیر کمکی delay و delay\_count و ایجاد شرط در بدنه تابع تایمر، کاری می‌کنیم تا از هر 1000 بار اجرای آن تابع، فقط یک بار به مقدار عدد ما یک واحد اضافه شود. یعنی delay را روی 1000 قرار می‌دهیم. از آنجا که وقفه ایجاد شده خیلی بیشتر از 1 ثانیه به نظر می‌رسد، مقدار آن را کاهش داده و مثلاً روی 200 تنظیم می‌کنیم.

**5. کد برنامه**

char SEG\_DATA[] = {0x3F, 0x06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07, 0x7F, 0x6F};

int num = 0;

char d0 = 0;

char d1 = 0;

char d2 = 0;

char d3 = 0;

char di = 0;

int delay = 200;

int delay\_count = 0;

int is\_counting = 0;

void setDigits() {

    d0 = (num / 1) % 10;

    d1 = (num / 10) % 10;

    d2 = (num / 100) % 10;

    d3 = (num / 1000) % 10;

}

void switchDigit() {

    setDigits();

    PORTD.7 = 1;

    PORTD.6 = 1;

    PORTD.5 = 1;

    PORTD.4 = 1;

*switch*(di) {

*case* 0:

           PORTD.7 = 0;

           PORTB = SEG\_DATA[d0];

*break*;

*case* 1:

           PORTD.6 = 0;

           PORTB = SEG\_DATA[d1];

*break*;

*case* 2:

           PORTD.5 = 0;

           PORTB = SEG\_DATA[d2];

*break*;

*case* 3:

           PORTD.4 = 0;

           PORTB = SEG\_DATA[d3];

*break*;

    }

    di = (di + 1) % 4;

}

*// External Interrupt 0 service routine*

interrupt [EXT\_INT0] void ext\_int0\_isr(void)

{

*// Place your code here*

    num = 0;

    is\_counting = 1;

}

*// External Interrupt 1 service routine*

interrupt [EXT\_INT1] void ext\_int1\_isr(void)

{

*// Place your code here*

    is\_counting = 0;

    num = 0;

}

*// Timer 0 overflow interrupt service routine*

interrupt [TIM0\_OVF] void timer0\_ovf\_isr(void)

{

*// Reinitialize Timer 0 value*

TCNT0=0x53;

*// Place your code here*

    switchDigit();

*if* (delay\_count == 0 && is\_counting) {

        num = (num + 1) % 10000;

    }

    delay\_count = (delay\_count + 1) % delay;

}