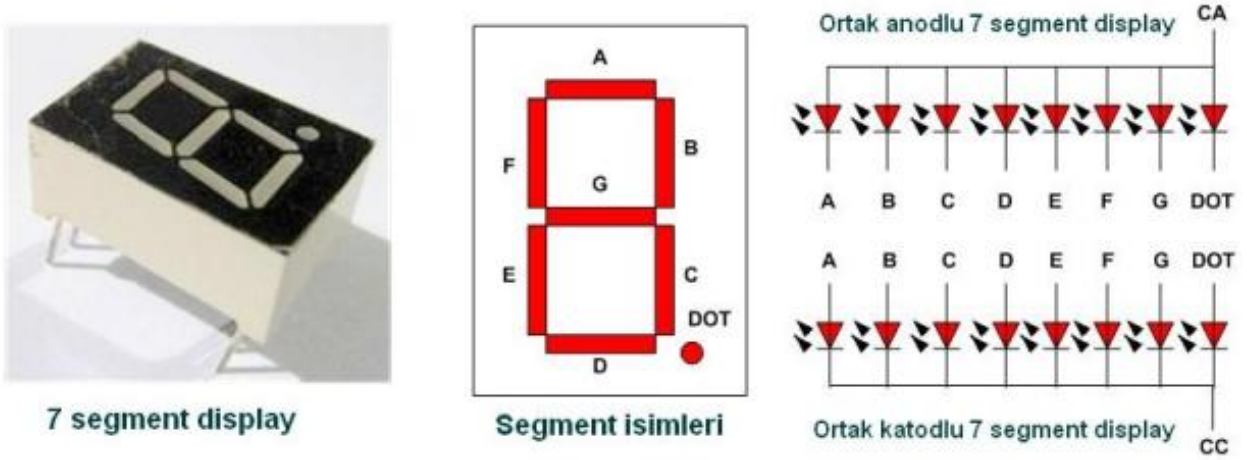


Uygulama 4: 7 Segment Display Uygulaması.

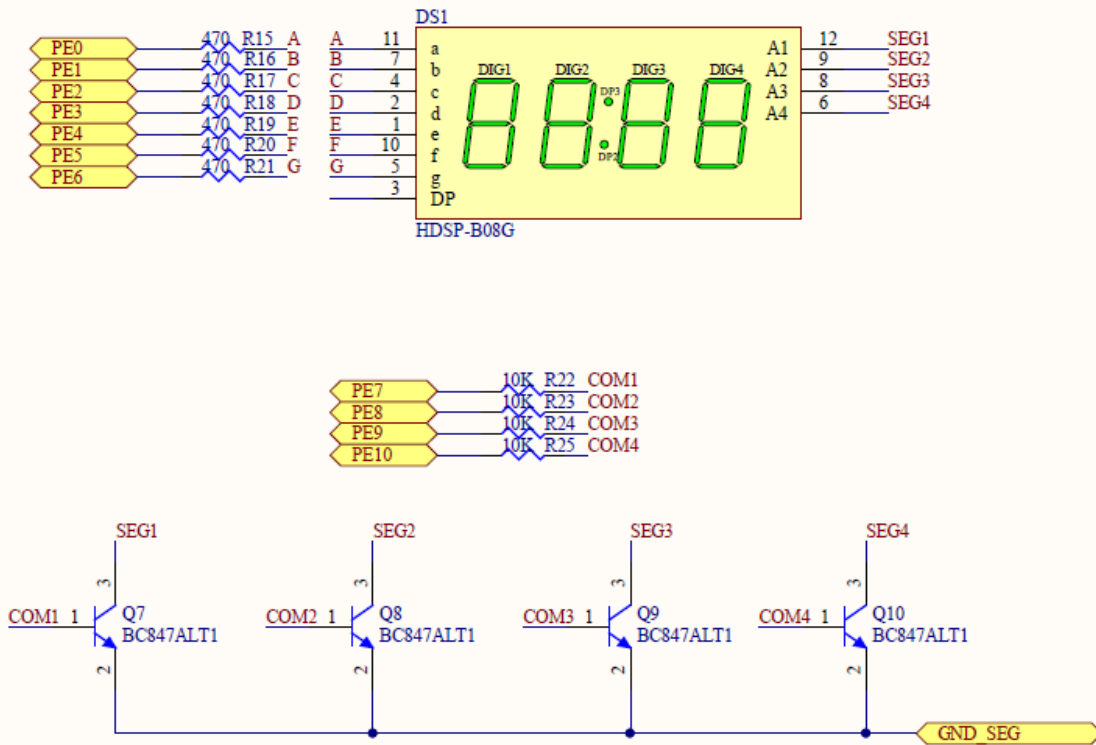
7 Segment display, asansörlerden, beyaz eşyalara, uydu alıcılarından akaryakıt pompalarına pek çok cihazda gösterge amaçlı olarak kullanılmaktadır. 7 segment displaylerin isimleri 7 parçalı yapılarından kaynaklanmaktadır. Bu parçalar kullanılarak bütün rakamlar ve hatta bazı harfler oluşturulabilmektedir.



Şekil 4.1: 7 Segment display ve iç yapısı.

7 segment displayler farklı boyutlarda ve farklı güçlerde üretilmekle birlikte iç yapıları da ortak anot ve ortak katot olmak üzere iki türdür. Küçük boyutlu ve küçük güçle çalışan 7 segment displayler mikrodenetleyiciler ile kullanılacaksa kullanım kolaylığı açısından ortak katotlusunu tercih etmek daha makuldür.

ARMapp-18 uygulama setinde kullanılan 7 segment display 4 haneli ve ortak katotlu bir 7 segment displaydir. 7 segment display modülün devre şeması şekil 4.2' de görülebilir.



Şekil 4.2: ARMapp-18 Uygulama setindeki buzzer, power led, buton ve led modüllerine ait şemalar.

ARMapp-18 uygulama setinde kullanılan 7 segment display modülünde her hane birer NPN transistör vasıtası ile kontrol edilmektedir. Hangi hane çalıştırılmak isteniyor ise o haneye ait olan transistör iletime geçirilerek istenilen rakam o hanede gösterilir. Örneğin 1234 sayısını displayde göstermek istersek önce 1 rakamını gösterecek değerleri displayin ABCDEFG pinlerine gönderilir. Daha sonra Q7 transistörünün beyzine gerilim uygulanarak iletime geçirilir (bu esnada diğer transistörler yalıtımda olmalıdır yani beyz uçlarına 0 gelmelidir). Bu halde 4' lü displayin 1. hanesinde '1' rakamını görürüz. Bu şekilde 5 ms civarında bir bekleme yapılarak 2. Hanede '2' rakamı gösterilir ve yine 5 ms beklenir. Bu işlemler diğer haneler için de gerçekleştirilir ve tekrar başa dönülerek 1. Hanede yine '1' rakamı gösterilir. Bu işlem 5 ms aralıklarla çok hızlı yapıldığı için bir göz yanılması oluşarak bütün rakamlar aynı anda görünür. Bu tekniğe tarama tekniği ismi verilir. Bu sayede bütün displaylerin segment bağlantıları için ayrı ayrı bağlantılar kullanılmaz ve pin sayısından tasarruf edilir.

Uygulama Kodları:

```

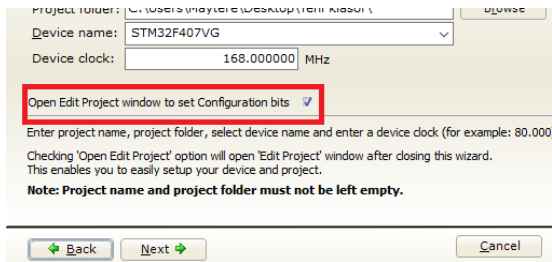
1 ///////////////////////////////////////////////////////////////////
2 //*****//
3 // 4 Haneli 7 segment Display kullanımı. //
4 // MikroC v6.2 - STM32F407VG //
5 // ARMapp-18 Deney Seti için yazılmıştır //
6 //*****//
7 // http://elektrovadi.com //
8 // http://mikrodunya.wordpress.com //
9 ///////////////////////////////////////////////////////////////////
10
11
12
13 unsigned int sayilar[10]={0x3F,0x06,0x5B,0x4F,0x66, // 7Segment displayde gösterilecek rakamların kodları 'sayilar'
14 // 0x6D,0x7D,0x07,0x7F,0x6F}; // dizisi olarak tanımlanıyor.
15 unsigned int i=0,j=0; // i ve j isminde iki adet unsigned int tipinde değişken tanımlanıyor
16
17 void display(unsigned int i) // 'display' isminde tanımlanmış ve işaretsiz tamsayı türünde
18 { // değer alan fonksiyon tanımlanıyor.
19 char binler,yuzler,onlar,birler; // char tipinde basamaklara ait sayıl değerleri tanımlanıyor.
20 binler=i/1000; // binler basamağının sayı değeri hesaplanıyor.
21 yuzler=(i/100)%10; // yuzler basamağının sayı değeri hesaplanıyor.
22 onlar=(i/10)%10; // onlar basamağının sayı değeri hesaplanıyor.
23 birler=i%10; // birler basamağının sayı değeri hesaplanıyor.
24
25 GPIOE_ODR=0x0080 | sayilar[binler]; // Binler basamağının sayı değerine karşılık gelen 7 segment kodu
26 delay_ms(2); // diziden çekilerek displayin en soldaki hanesinde gösteriliyor ve
27 // 2 ms beklenerek sayının displayde görülebilmesi sağlanıyor.
28
29 GPIOE_ODR=0x0100 | sayilar[yuzler]; // Yuzler basamağının sayı değerine karşılık gelen 7 segment kodu
30 delay_ms(2); // diziden çekilerek displayin en soldaki hanesinde gösteriliyor ve
31 // 2 ms beklenerek sayının displayde görülebilmesi sağlanıyor.
32
33 GPIOE_ODR=0x0200 | sayilar[onlar]; // Onlar basamağının sayı değerine karşılık gelen 7 segment kodu
34 delay_ms(2); // diziden çekilerek displayin en soldaki hanesinde gösteriliyor ve
35 // 2 ms beklenerek sayının displayde görülebilmesi sağlanıyor.
36
37 GPIOE_ODR=0x0400 | sayilar[birler]; // Birler basamağının sayı değerine karşılık gelen 7 segment kodu
38 delay_ms(2); // diziden çekilerek displayin en soldaki hanesinde gösteriliyor ve
39 // 2 ms beklenerek sayının displayde görülebilmesi sağlanıyor.
40 }
41
42 void main() // Ana program bloğu
43 {
44 GPIO_Digital_Output(&GPIOE_BASE, _GPIO_PINMASK_0| // 'E' portunun 0,1,2,3,4,5,6,7,8,9 ve 10. pinleri
45 // _GPIO_PINMASK_1| // dijital çıkış olarak tanımlanıyor.
46 // _GPIO_PINMASK_2|
47 // _GPIO_PINMASK_3|
48 // _GPIO_PINMASK_4|
49 // _GPIO_PINMASK_5|
50 // _GPIO_PINMASK_6|
51 // _GPIO_PINMASK_7|
52 // _GPIO_PINMASK_8|
53 // _GPIO_PINMASK_9|
54 // _GPIO_PINMASK_10);
55
56 while(1) // Sonsuz döngü.
57 {
58 for(i=0;i<10000;i++) // 0'dan başlayıp 9999 olana kadar sayıları displayde gösteren kod.
59 {
60 for(j=0;j<5;j++)display(i); // aynı sayı 5 kez gösterilerek sayı artışında gecikme sağlanıyor.
61 }
62 }
63 }

```

Kod kısmında dikkat edilecek husus sonsuz döngü içerisinde kullanılan iç içe iki for döngüsünün kullanımınıdır. Bu döngülerden dışarıda olanı 0-9999 arası sayma işlemini yaparken içteki for döngüsü tarama işleminin yapıldığı “display” fonksiyonunu çağıran for döngüsüdür. İçteki for döngüsünde dikkat edilecek olursa “display” fonksiyonu 5 kez çağırılmakta ve aynı sayı 5 kez görüntülenmektedir. Eğer bu sayı arttırılırsa aynı sayıda daha uzun süre kalınacağından sayma hızı düşecektir.



İşlem Basamakları:

- 1- MikroC Pro for ARM programını çalıştırınız.
- 2- “Project” sekmesinden “New Project” seçeneğine tıklayınız.
- 3- Açılan ekrandan “Standart Project” seçilerek “Next” butonuna tıklayınız.
- 4- Açılan pencereden proje ismi, projenin kaydedileceği dizin, kullanılacak mikrodenetleyiciyi (STM32F407VG) ve mikrodenetleyici frekansını (168.000000 MHz) belirleyerek ve “Open Edit Project window to set Configuration bits” seçeneğini işaretleyerek “Next” butonuna tıklayınız.



Project folder: [C:\Users\...]\Documents\...
Device name: STM32F407VG
Device clock: 168.000000 MHz
Open Edit Project window to set Configuration bits
Enter project name, project folder, select device name and enter a device clock (for example: 80.000).
Checking 'Open Edit Project' option will open 'Edit Project' window after closing this wizard.
This enables you to easily setup your device and project.
Note: Project name and project folder must not be left empty.

Back Next Cancel

- 5- Açılan pencerede “Finish” butonuna tıklayarak proje oluşturma aşamasını bitiriniz.
- 6- Açılan pencerede sağ tarafta bulunan Load Scheme butonuna tıklayınız ve açılan pencereden “HSI 16 MHz_PLL 168 MHz Clock Ayarı.cfgsch” dosyasını seçiniz.
- 7- “OK” butonuna tıklayarak proje oluşturma işlemini tamamlayınız.
- 8- Açılan “.c” uzantılı sayfada kodları yazınız ve  ikonuna tıklayarak projeyi kaydediniz.
- 9- ARMapp-18 setini USB kablo kullanarak bilgisayara bağlayınız.
- 10- Piyano anahtarlardan sağ taraftakinde bulunan 7 SEGMENT anahtarını açarak, bu modülün GND bağlantısını sağlayınız.
- 11-  “Build and Program” ikonuna tıklanarak veya (Ctrl+F11) tuşlarına aynı anda basarak yazılan programı derleyip STM32F407VG mikrodenetleyicisine yükleyiniz.
- 12- Programın çalışmasını gözlemleyiniz.

Sorular:

- 1- Bu uygulamada “delay_ms” gibi bir gecikme fonksiyonunun neden kullanılmadığını açıklayınız.
- 2- Menü tuş takımı modülünü kullanarak yukarı butonuna tıkladığında yukarıya doğru sayan, aşağı butonuna basıldığında aşağı doğru sayan programı yazınız.