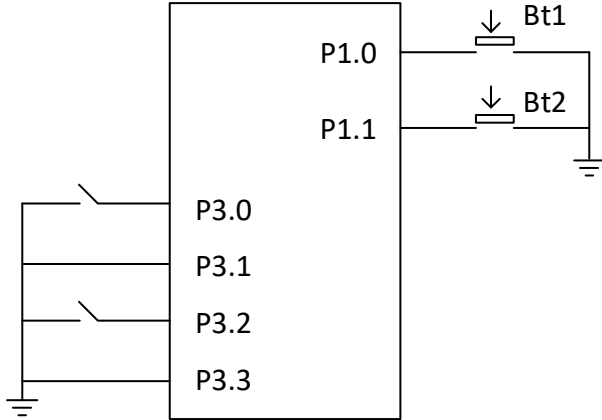


1. P3'te yüksek nibble'daki deęer bilinmiyor. P3'ün düşük nibble'da 10d deęeri vardır. Bt1 butonuna basıldığında p3'ten alınan düşük nibble deęerin (10d=0Ah) 2 katını, Bt2 butonuna basıldığında ise 3 katını alınız. Sonucu P0'da gözlemleyiniz.

Pinlerin kontrolünü (butonlara basılıp basılmadığını) jb veya jnb komutu kullanarak yapılacaktır.



2. Yukarıdaki arpma işlemlerini alt programda yapınız. Bir alt programınız olacak o sadece 2 katını alacak, dięer bir alt programınız ise 3 katını alacak.

3. 2. Sorudaki programı, Keil programını adım adım alıřtırırken acall kodu alıřmadan önce döneęiniz adresi belirleyiniz. Sp deęişimleri ile sp adreslerin gösterdięi içeriklerini gözlemleyiniz.

4. Ařaęıdaki programı yazınız. Veritabanı etiketinin program adresini gözlemleyiniz. Ayrıca veri tabanındaki deęerleri ve bunların program hafızadaki adreslerini gözlemleyiniz. Program adresi hem debug modunda dissambly penceresinde hemde memory kısmında c:0x00 řeklinde gözlemleyebilirsiniz.

```
cs1.a51
1  #include "aduc841.h"
2  org 0000
3  sjmp ayar
4  ayar:
5  ; burada mikro
6  ; ayarlari yapilir
7  basla:
8  mov r0,#04h
9  mov a,#33h
10
11  sjmp basla
12
13  veritabani: db 19d,
14               8d,
15               17d,
16               6d,
17               15d,
18               4d,
19               13d,
20               2d,
21               11d,
22               0d // 10 adet veri
23  ;veritabani: db 19d,8d,17d,6d,15d,4d,13d,2d,11d,0d // bu řekilde daha rahat olusturulabilirdi.
24  end
```

Bundan sonraki sorularda veri tabanındaki deęerler bilinmedięi sadece veri tabanındaki veri sayısının 10 olduęu bilinerek ařaęıdaki soruları cevaplayınız.

5. 4. Soruda hazırladıęınız programı řu řekilde gncelleyelim: Veri tabanındaki deęerlerin bilinmedięi sadece veri tabanında 10 veri olduęu bilgisi ile veri tabanındaki 8. deęeri (yani 1d verisini) p0 da gsteriniz.

6. 4. Soruda hazırladığınız programı şu şekilde güncelleyelim: Bir 10 luk döngü ile (**djnz**) p0 da tek tek veri tabanındaki bilgileri aşağıda verilen a ve b adımlarında verilen şekilde p0 da gözlemleyiniz.

a. İlk önce veri tabanındaki 10. veriden başlayarak geriye doğru 0. veriye kadar olan verileri p0 da tek tek gözlemle. (0d->p0'da, ardından 11d->p0'da, ardından 2d->p0'da.....son olarak 19d->p0'da olacak şekilde )

b. Bu biraz zor. Sonraya bırakılabilir. Yine djnz kullanarak 0. veriden 10. veriye kadar olan verileri tek tek p0'da gözlemle.

7. 6. Sorudaki kodu cjne ile hem a hem de b şıkkı için yapınız. Bu sefer b şıkkı kolaylıkla yapılabilir.

8. 4. Soruda hazırladığınız programı şu şekilde güncelleyelim: Veri tabanında bir arama yapma programı yapılacaktır. Örneğin veri tabanında 6d değeri var ise p0.1 yakınız, yok ise p0.2 yakınız. Örneğin veri tabanında 16d değeri var ise p0.1 yakınız yok ise p0.2 yakınız.

9. 4. Soruda hazırladığınız programı şu şekilde güncelleyelim: Bir döngü ile veri tabanındaki tek indisli bilgileri tek tek p0'da gözlemleyiniz. Yani 1. 3. 5. 7. ve 9. sıradaki bilgileri

10. 4. Soruda hazırladığınız programı şu şekilde güncelleyelim: Bir döngü ile veri tabanındaki tek indisli bilgileri toplayalım ve sonucu p0'da gözlemleyelim.

11. 4. Soruda hazırladığınız programı şu şekilde güncelleyelim: Bir döngü ile veri tabanındaki tek indisli bilgileri toplayalım ve sonucu r7'ye yazalım, ardından çift indisli bilgileri toplayalım ve sonucu r6'ya yazalım. Ve r7-r6 çıkartma işlemini gerçekleştiriniz. Sonuç pozitif ise p0.1 deki ledi yakınız, negatif ise p0.2deki ledi yakınız.