

```

1      AREA ders4, CODE
2      EXPORT __main
3
4      __main PROC
5
6      ;-----ACIKLAMA-----
7      ; Karsilastirma islemi
8      ;CMP R3,#0 ; Hedef: R3, Kaynak: #0
9
10     ;Bazi Branch Komutlari
11     ;B <etiket adi> : Kosulsuz olarak etiket adinin bulundugu kod satirina gider.
12     ;BEQ <etiket adi> : Karsilastirma sonucu esit ise etikete gider.
13     ;BNE <etiket adi> : Karsilastirma sonucu esit degil ise etikete gider.
14     ;BGT <etiket adi> : Karsilastirma sonucu buyuk ise etikete gider.
15     ;BLT <etiket adi> : Karsilastirma sonucu kucuk ise etikete gider.
16     ;BLE <etiket adi> : Karsilastirma sonucu kucuk veya esit ise etikete gider.
17     ;BGE <etiket adi> : Karsilastirma sonucu buyuk veya esit ise etikete gider.
18
19     ;-----
20
21     ;-----BASLANGIC-----
22
23     ;if(i == j) { f = g + h;} else{ f = g - h;}
24     MOV R0,#5 ;i degiskeni
25     MOV R1,#6 ;j
26     MOV R2,#20 ;g
27     MOV R3,#3 ;h
28     MOV R4,#0 ;f
29     CMP R0,R1
30     BEQ dogru0 ;R0,R1'e esit ise "dogru0" etiketinin oldugu kod satirina git.
31     SUB R4,R2,R3 ;BEQ komutu calismazsa yani esit degilse bu kod satirindan devam.
32     B son0 ;Buradan sonra, "dogru0" etiketinin calismamasi icin "son0" etiketinin oldugu
33     ;kod satirina kosulsuz olarak git.
34
35     dogru0
36         ADD R4,R2,R3
37     son0
38     ;-----BITIS-----
39
40     ;-----BASLANGIC-----
41
42     ; if(x>20 && x<25) { a = 1;}
43     MOV R0,#5 ;x degiskeni
44     CMP R0,#20
45     BLE son1 ;x 20den kucuk veya esitse "son1" etiketine git.
46     CMP R0,#25
47     BGE son1 ;x 25den buyuk veya esitse "son1" etiketine git.
48     MOV R1,#1 ;a degiskeni, BLE ve BGE saglanmazsa bu komut calisir.
49
50     son1 ;BLE veya BGE saglanirsa bu komut satiri calisir.
51     ;-----BITIS-----
52
53     ;-----BASLANGIC-----
54
55     ;if(x<=20 || x>=25) { a = 1}
56     MOV R0,#5 ;x
57     CMP R0,#20
58     BLE dogru1 ;x 20den kucuk veya esit ise "dogru1" etiketine git.
59     CMP R0,#25
60     BLT son2 ;x 25den kucuk ise "son2" etiketine git.
61
62     dogru1
63         MOV R1,#1 ;a
64
65     son2
66     ;-----BITIS-----
67
68     ;-----BASLANGIC-----
69
70     ;do{g--; i=i+j;}while(i!=h)
71     MOV R0,#9 ;g
72     MOV R1,#0 ;i
73     MOV R2,#1 ;j
74     MOV R3,#6 ;h
75     loop1
76         SUB R0,#1
77         ADD R1,R1,R2

```

```

78     CMP R1,R3
79     BNE loop1    ;i, h'ye esit degil ise loop1 etiketine git.
80     ;-----BITIS-----
81
82     ;-----BASLANGIC-----
83
84     ;sum =0;
85     ;for(i=0; i<10;i++){sum=sum+i}
86     MOV R0,#0    ;i
87     MOV R1,#0    ;sum
88     dongu
89     CMP R0,#10
90     BGE son3     ;i, 10dan buyuk veya esit ise "son3" etiketine git.
91     ADD R1,R1,R0
92     ADD R0,#1
93     B dongu      ;Kosulsuz olarak "dongu" etiketine git.
94     son3
95     ;-----BITIS-----
96
97     ;-----BASLANGIC-----
98
99     BL delay     ;Alt programi cagirmak icin bu komut kullanilir.
100    ;BL <etiket adi> seklinde yazilir.
101    ;-----BITIS-----
102
103    ;-----BASLANGIC-----
104
105    ;1 saniye araliklarla led yakip sondurme odevini delay alt programi ile yapiyoruz.
106    MOV R0,#0x1000
107    MOVT R0,#0x4001
108    LDR R1,[R0,#0x4]
109    ORR R1,#(1<<21)
110    AND R1,#~(1<<22)
111    STR R1,[R0,#0x4] ;PortC 13 numarali pin cikis olarak ayarlandi.
112
113    ;loop2
114
115    MOV R0,#0x1000
116    MOVT R0,#0x4001
117    LDR R1,[R0,#0xC]
118    ORR R1,#(1<<13)
119    STR R1,[R0,#0xC] ;13 numarali pin 1 olarak ayarlandi.
120
121    BL delay      ;1 saniye bekle
122
123    MOV R0,#0x1000
124    MOVT R0,#0x4001
125    LDR R1,[R0,#0xC]
126    AND R1,#(~(1<<13))
127    STR R1,[R0,#0xC] ;13 numarali pin 0 olarak ayarlandi.
128
129    BL delay      ;1 saniye bekle
130
131    ; B loop2 ;Sonsuz dongu olacagi icin alt satirlara gecmeyecektir o yuzden yorum satiri yaptim.
132    ;-----BITIS-----
133
134    ;-----BASLANGIC-----
135
136    ;void swap(char x, char y){
137    ;char t;
138    ;t=x;
139    ;x=y;
140    ;y=t;
141    ;}
142    ;Soru: verilen kodu assembly dilinde alt program kullanarak kodlayiniz.
143    MOV R0,#5     ;x
144    MOV R1,#6     ;y
145    BL swap
146    ;-----BITIS-----
147
148    ;-----BASLANGIC-----
149
150    ;SysTick ile 1 saniyelik bekleme
151    MOV R0,#0x1000
152    MOVT R0,#0x4002
153    MOV R1,#(1<<4)
154    STR R1,[R0,#0x18]

```

```

155
156     MOV R0,#0x1000
157     MOVT R0,#0x4001
158     LDR R1,[R0,#0x4]
159     ORR R1,#(1<<21)
160     AND R1,#~(1<<22)
161     STR R1,[R0,#0x4] ;PortC 13 numarali pin cikis olarak ayarlandi.
162
163     MOV R0,#0xE000E000 ;SysTick baslangic adresi
164     MOV32 R1,#9000000 ;Islemin bir saniye surmesi icin bu degeri yukluyoruz
165     STR R1,[R0,#0x14] ;Reload Value Register'ina R1 degerini yukledik.
166     MOV R1,#3
167     STR R1,[R0,#0x10] ;Control Register'inda Enable bitini ve Tickint bitini 1 yapiyoruz.
168
169     ;Burada "BX SysTick_Handler" yazmamiza gerek yok, Tickint aktif ettigimiz icin girilen deger
170     ;sifira ulastiginda interrupt ile "SysTick_Handler" alt programi calisacaktır.
171 ;-----BITIS-----
172
173 loop
174     B loop
175
176     ENDP ;"__main" programinin bittigini belirtir.
177
178 ;-----ACIKLAMA-----
179     ;Alt program bir method gibi dusunulebilir; "__main" methodu, "delay" methodu gibi.
180     ;Method tanimlarken etiket adinin yanina PROC yazilir.
181     ;Methodun bittigini belirtmek icin ENDP yazilir.
182 ;-----
183
184 ;-----ALT PROGRAMLAR-----
185
186 delay PROC
187     MOV32 R3,#10000000
188 dongu1
189     SUB R3,#1
190     CMP R3,#0
191     BNE dongu1
192
193     BX lr ;BX Link Register, "__main" methodundan cagirabilmemiz icin gerekli
194     ENDP ;"delay" programinin bittigini belirtir.
195
196 ;-----
197
198 swap PROC
199     ;R0: x, R1: y, R2: temp
200     MOV R2,R0 ;temp = x
201     MOV R0,R1 ;x = y
202     MOV R1,R2 ;y = temp
203
204     BX lr
205     ENDP ;"swap" programinin bittigini belirtir.
206
207 ;-----
208
209 SysTick_Handler PROC
210     EXPORT SysTick_Handler
211
212     MOV R5,#0x1000
213     MOVT R5,#0x4001
214     LDR R6,[R5,#0xC]
215     EOR R6,#(1<<13) ;EOR komutu ledi yaniyorsa sonduruyor, sonuyorsa yakiyor.
216     STR R6,[R5,#0xC]
217
218     BX lr
219     ENDP ;"SysTick_Handler" programinin bittigini belirtir.
220
221
222     END ;Assembly kodunun bittigini belirtir.
223

```