

1. Şekil 1’de örnek MİB dilinde yazılmış bir program ve program çalışmadan önce bazı bellek gözlerinin içerikleri verilmiştir.

a. (10 puan) Bu programda yer alan boşlukları doldurunuz.

b. (15 puan) Programın sonlanması ardından Şekil 1’deki tabloda yer alan bellek gözlerinin , A ve B akümülatörlerinin içerikleri ne olur?

0010	20 05 0F FF	BAŞLA	YÜK	SK, \$0FFF	Bellek Adresi	Veri
0014	20 02 10 FF		YÜK	CD, \$10FF	0040	00
0018	4B 41		SİL	B	:	:
001A	70 45	CEVRİM	ART	SK	0FFF	FF
001C	70 42		ART	CD	1000	46
001E	00 60 00		YÜK	A, <SK+0>	1001	55
0021	1C 00 FF		KAR	A, -- (a)	1002	82
0024	81 -- (b)		DEE	BITTI	1003	A5
0026	01 40		YAZ	A, -- (c)	1004	FF
0028	50 41		ART	B	:	:
002A	80 EE		DAL	-- (d)	10FF	FF
002C	01 21 00 40	BITTI	YAZ	B, \$0040	1100	AA
0030	C3		KES		1101	FF
					1102	AA
					1103	55
					1104	FF
					1105	62

Şekil 1

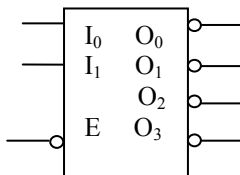
2. (35 puan) Bellekte \$1000 adresinden başlayarak 8 bitlik **işaretsiz** sayılardan oluşan bir dizi bulunmaktadır. Dizinin eleman sayısı ($n < 255$) bellekte \$000A adresindedir. Bu dizinin elemanlarından değeri bellekte \$000F adresinde bulunan sayıdan daha küçük olanları ile \$1100 adresinden başlayarak yeni bir dizi oluşturulacak ve bu yeni dizinin eleman sayısı \$000B adresine yazılacaktır. Bu işlemi yapan programı örnek MİB dilinde yazınız, gerekli açıklamaları kısaca yapınız.

3. 16 bit adres yolu ve 8 bit veri yolu olan bir bilgisayarın bellek haritası şu şekildedir:

- \$0000 adresinden başlayan 8K*8 düzeninde OKU/YAZ bellek
- \$3FFF adresinde biten 4K*8 düzeninde salt oku bellek

Tasarımda kullanılacak olan OKU/YAZ bellekler 2K*8 boyunda, salt oku bellekler 4K*8 boyundadır.

- a. (30 puan) Gerekli olan tasarımı aşağıda özellikleri verilen 2’ye 4 kod çözücü ve gerekli lojik kapıları kullanarak yapınız, örnek mikroişlemci ile olan bağlantılarını gösteriniz.
- b. (10 puan) OKU/YAZ belleğin başlangıç adresi \$4000 ve salt oku belleğin bitiş adresi \$7FFF olarak değiştirilirse tasarımda gerekli olan değişikliği yapınız.



E	I ₁	I ₀	O ₀	O ₁	O ₂	O ₃
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0