

Bellekler

Bellekler

Ortak giriş/çıkışlara, yazma ve okuma kontrol sinyallerine sahip eşit uzunluktaki saklayıcıların bir tümdevre içerisinde sıralanmasıyla hafıza (bellek) yapısı elde edilir. Çeşitli türleri mevcuttur:

- Salt Okunur Bellekler(ROM)
 - Salt Okunur Bellekler(Read Only Memory-ROM)
 - Programlanabilir Salt Okunur Bellekler (PROM)
 - Silinebilir Programlanabilir Salt Okunur Bellekler (EPROM)
- Oku/Yaz Bellekler(RAM)
 - Statik Oku-Yaz Bellekler (SRAM)
 - Dinamik Oku-Yaz Bellekler (DRAM)

Salt Okunur Bellek

(Read Only Memory-ROM)

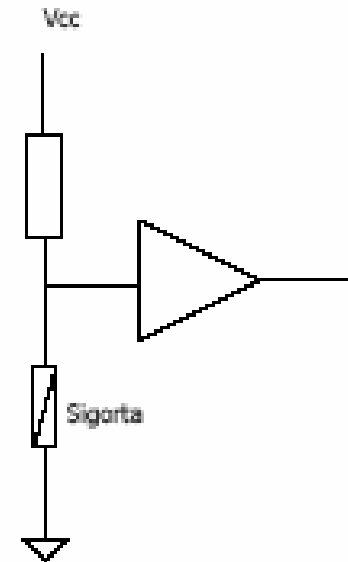


Üzerindeki veriler sadece okunabilir, yazılamaz.

- Kişisel bilgisayarlar
- Sabit programla çalışan bilgisayarlar
- Sistem programlarında kullanılır.

Programlanabilir Salt Okunur Bellek (PROM)

- Bellek gözleri için nikel-krom veya polikristal maddelerinden oluşan bir sigorta bulunur. Üretildiklerinde bellek gözleri 0 veya 1 konumundadır.
- Yüksek bir akım uygulandığında sigorta atar.
- Sadece bir kez programlanabilir.
- Üzerindeki verilen silinmesi mümkün değildir.



Silinebilir-Programlanabilir Salt Okunur Bellek (EPROM)

- Üretildiklerinde tüm bellek gözleri elektriksel olarak 1 konumundadır.
- Elektriksel olarak programlanabilirler.
- Morötesi ışık kullanarak silinebilirler.
- Belli sayıda silme ve programlama işlemi gerçekleştirilebilir.

Elektriksel Olarak Silinebilir Programlanabilir Salt Okunur Bellek (EEPROM)

- Daha önceki verileri silmeye gerek yoktur.
- Programlama işlemi belirlenen sayıda tekrar tekrar gerçekleştirilir.

Flash Bellek

- EEPROM tipi belleklere benzemektedir.
- Veriler, elektriksel olarak bloklar halinde silinmektedir.
- Taşınabilir USB depolama cihazlarında sıklıkla kullanılmaktadır.

Oku-Yaz Bellekler (Random Access Memory-RAM)

- Veriler elektriksel olarak depolanırlar.
- Güç kesilmelerinde üzerindeki veriler kaybolur.

Static RAM-SRAM



Bellek üzerine bir veri yazıldığında, tümdevreye enerji verildiği ve yazılan yere başka bir kelime yazılmadığı sürece, bu kelime hafızada saklı kalır. Bu yüzden verinin tazelenmesi gibi bir durum söz konusu değildir.



Ön belleklerde tercih edilir.

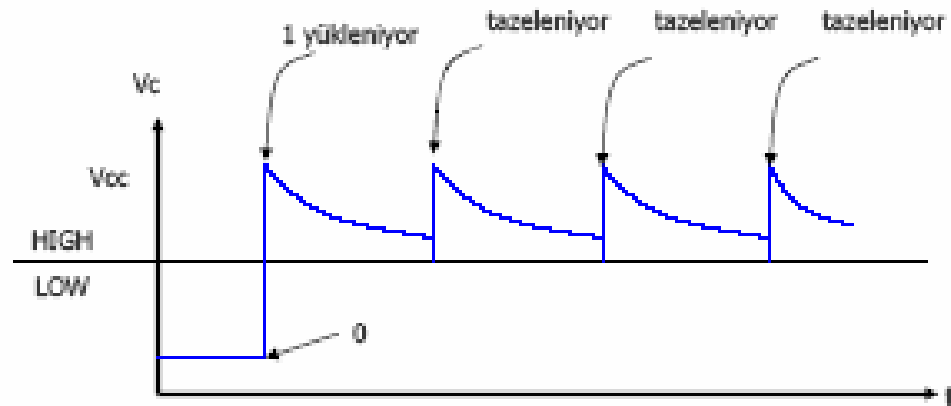
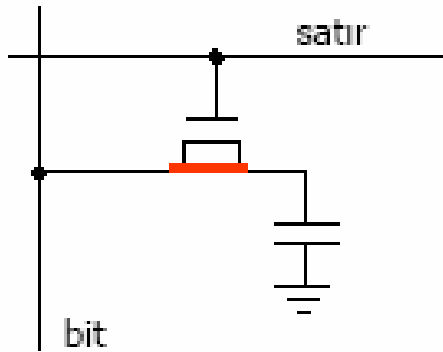


Hızlı ve pahalı bir bellek türüdür.

Dynamic RAM-DRAM

Her bir hücrede saklanan verinin, periyodik bir şekilde okunup tekrar geri yazılarak tazelenmesi gerekmektedir. Aksi halde bellek hücresindeki veri kaybolur.

- Bir kondansatör ve transistörden oluşur.
- Veri, kondansatörün doldurulması ile saklanmaktadır. Fakat sistem üzerinde istenmeyen kaçak akımlar olduğundan kondansatörün sık sık tazelenmesi gerekmektedir.



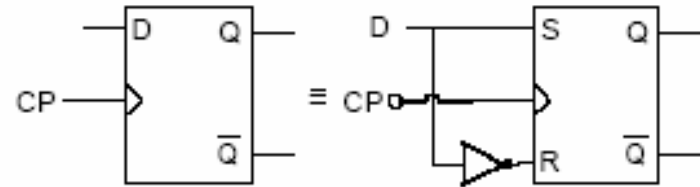
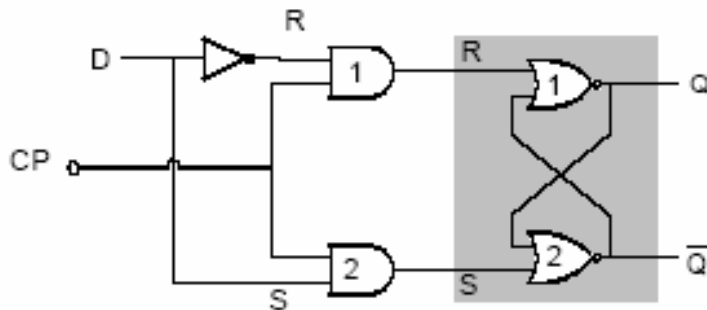
Bellek Düzeni

Bellek tümdevresinin kapasitesi arttıkça kullanılacak uç sayısı da **logaritmik** olarak artmaktadır.

-  Temel bellek birimi gözedir.
-  Gözelerin yanyana gelmesiyle gözler oluşur.
-  Gözelerin bir araya gelmesiyle de bellek oluşur.

Temel Bellek Birimi

Temel bellek birimi D tipi flip-floptur.



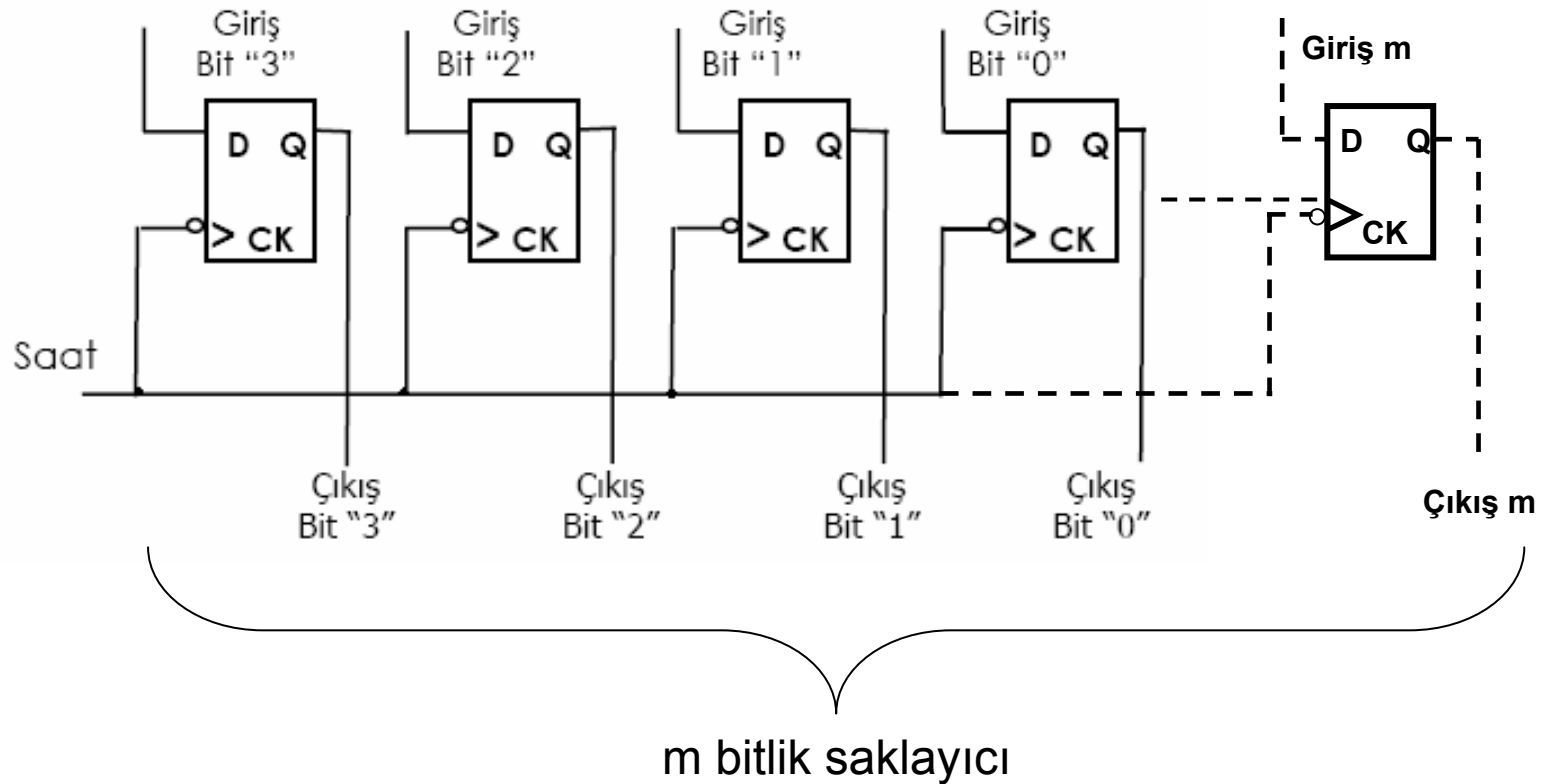
CP	D	Q_{n+1}	$\overline{Q}_{n+1}$
↓	x	Q_n	$\overline{Q}_n$
↑	0	0	1
↑	1	1	0

Değişim yok

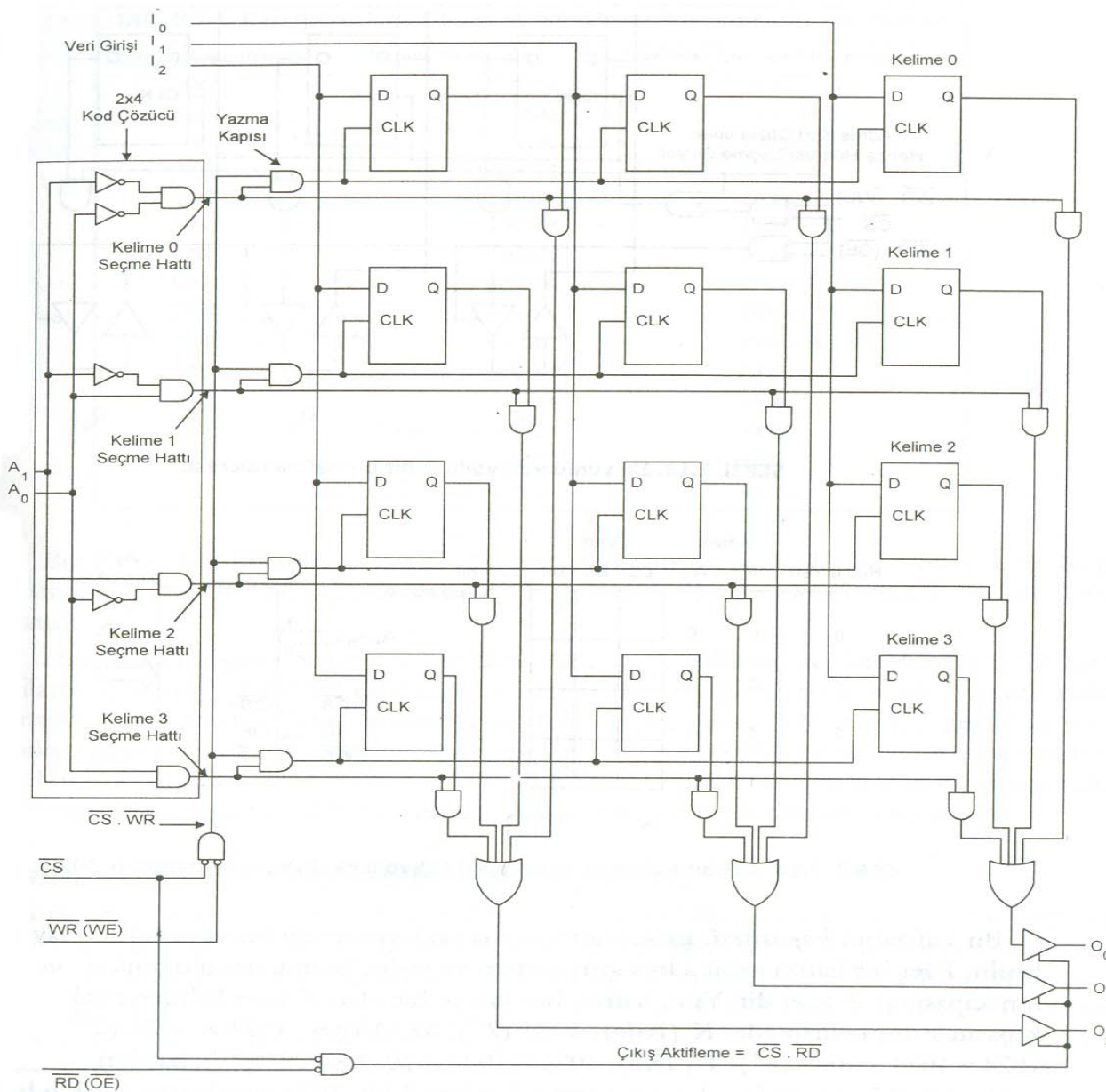
Silme

Kurma

m Bit Saklayıcı



Örnek Bellek Yapısı



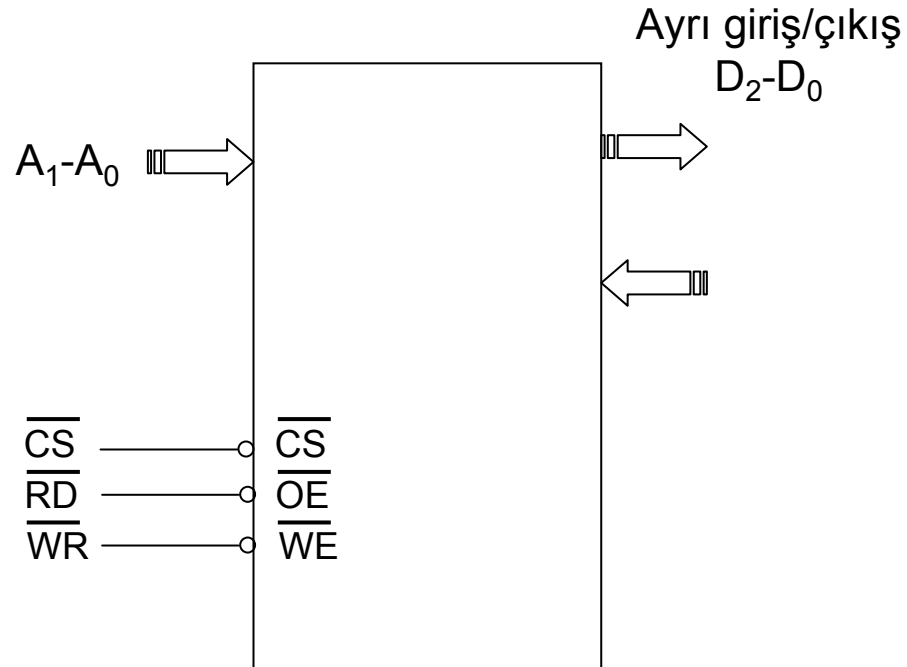
4x3 bit hafızanın lojik diyagramı. Her bir satır, toplam 4 tane olan, 3 bit kelimedenden birine karşılık gelir.

Bir hafıza okuma veya yazma işleminde, her zaman 3 bit bir kelime okunur veya yazılır.

Örnek Bellek Yapısı (devam)

Aktiflenecek Hücre Yeri	A1	A0
0	0	0
1	0	1
2	1	0
3	1	1

Kavramsal Gösterim



Blok Gösterim



Her bir saklayıcı, tek bir adrese sahip farklı fiziksel bir yer (location) işgal eder. Diğer bir deyişle, hafıza adreslenebilir saklayıcılar yığını olarak düşünülebilir.



Hafıza tümdevresinin içinde, adres girişlerinin kodunu çözüp içindeki saklayıcılardan birini seçmek için, bir kod çözücü lojik bulunur.



$\overline{CS}$ (Chip Select) sinyali, hafıza tümdevresinin aktif olması için, bir dış lojik tarafından üretilmesi gereken ilk sinyaldir.



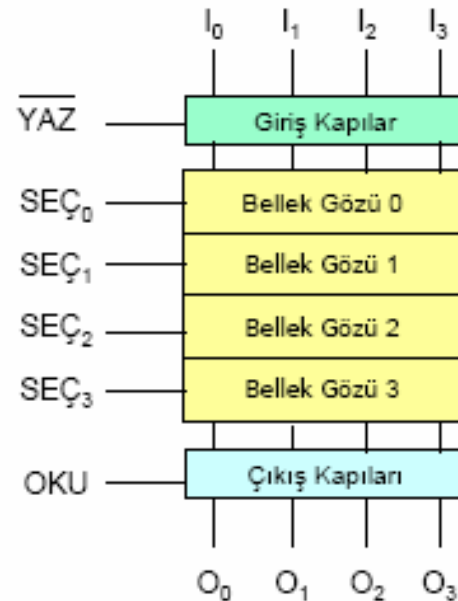
$\overline{WR}$ ve $\overline{RD}$ sinyallerinin durumlarına göre, seçili hafıza hücrelerine yazma veya okuma gerçekleştirilir.



$\overline{WE}$ ve $\overline{OE}$ sinyalleri genelde bir hafıza tümdevresi üzerinde, yazma ve okuma işlemlerini aktiflemeyi belirten girişler olarak gösterilir.

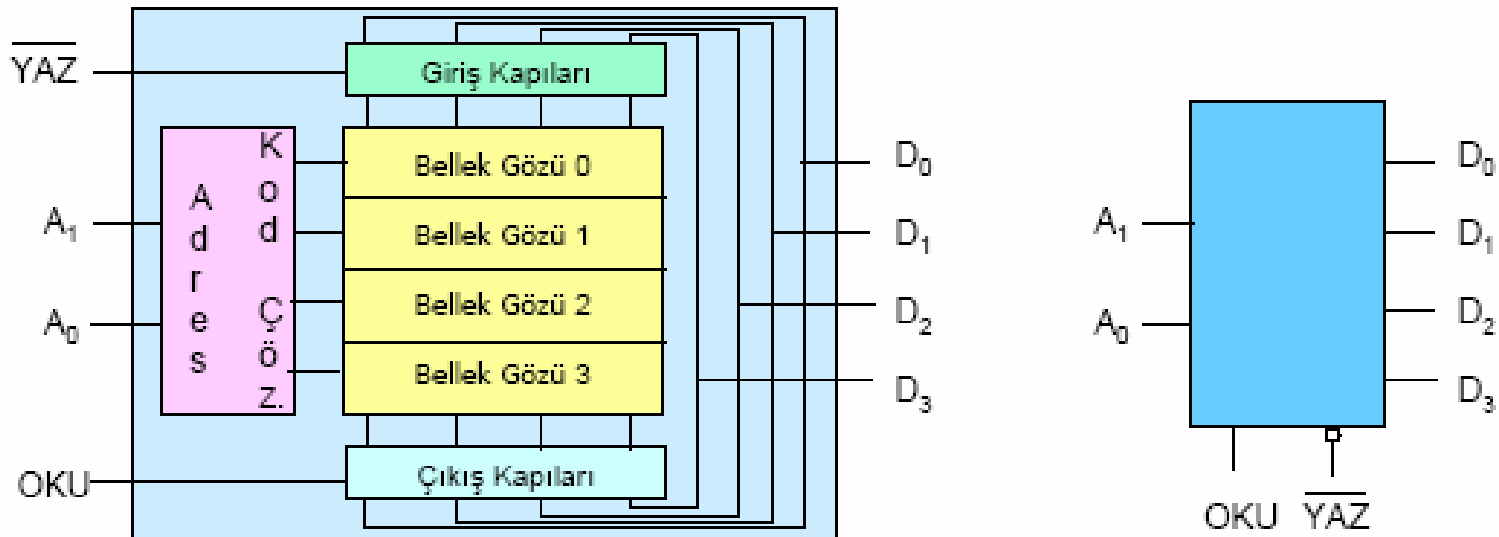
Bellek Adreslemesi

- ✓ Oku ve Yaz işaretleri ile verinin yönü belirlenir ve uygun seçme ucu aktiflenerek istenilen bellek gözü seçilir.
- ✓ Aynı anda sadece bir tek bellek gözü seçilebilir.

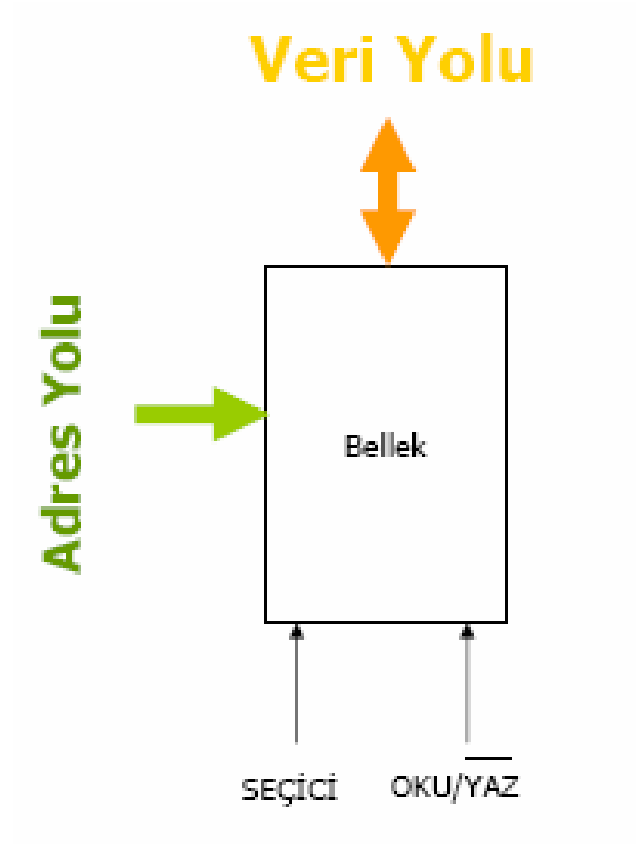


Bellek Tasarımı

Üç durumlu bufferlar bulunduğu için giriş ve çıkış veri aktarım uçları aynı uçlar olabilir.



Belleğin Dıştan Görünümü



Hafıza Kapasitesi



Bir hafızanın kapasitesi, maksimum sayıda saklayabileceği bit veya kelime sayısı ile belirtilir. Eğer bir hafıza n -bit adres giriş hattına ve m -bit kelime uzunluğuna sahipse hafızanın kapasitesi $2^n \times m$ 'dir.

Örn: 1Kx8'lik bir hafıza, her biri 8-bit genişlikte olan 1024 tane hücreyi ifade etmektedir.

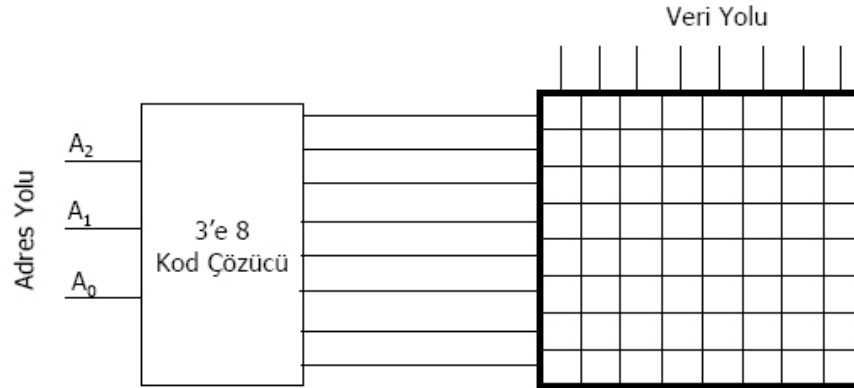
Örnek

Hafıza kapasitesi	Kelime sayısı	Adresleme için gerekli bit sayısı
1K	1024	10 bit
2K	2048	11 bit
4K	4096	12 bit
8K	8192	13 bit
16K	16384	14 bit
32K	32768	15 bit
64K	65536	16 bit
512K	8 x 64K	19-bit

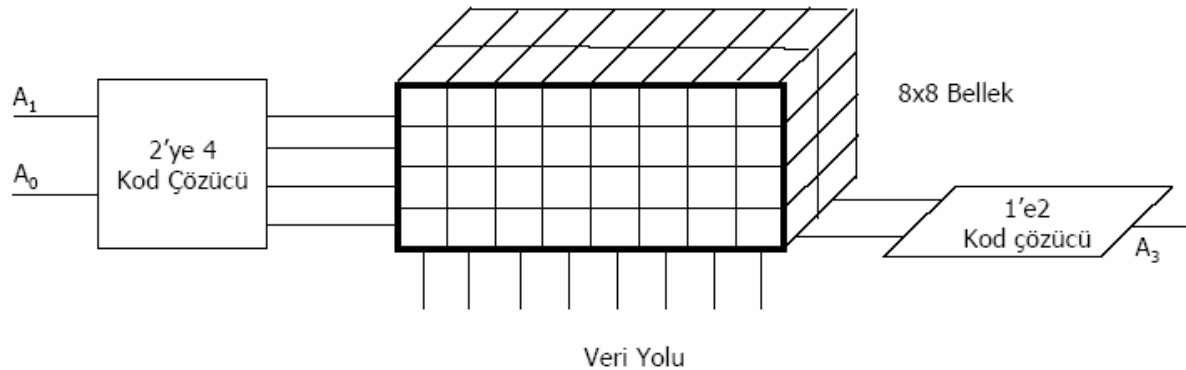
Adresleme



Tek seviyeli kod çözümü



İki seviyeli kod çözümü



Zamanlama

Bir hafızanın çalışabilmesi için, uygulanan adres, veri ve kontrol sinyalleri için zamanlama sınırlamaları bulunur. Hafıza sinyallerini analiz etmek için en basit işlem okuma çevrimidir. Bir **okuma** çevriminde aşağıda olaylar gerçekleşir:



Hafıza girişlerine adres uygulanır.



$\overline{CS}$ girişine lojik 0 uygulanarak SRAM seçilir.



Seçilen hafıza hücresinin içeriği, erişim zamanına (access time) eşit bir süre sonra ,veri çıkışlarında gözlenir.

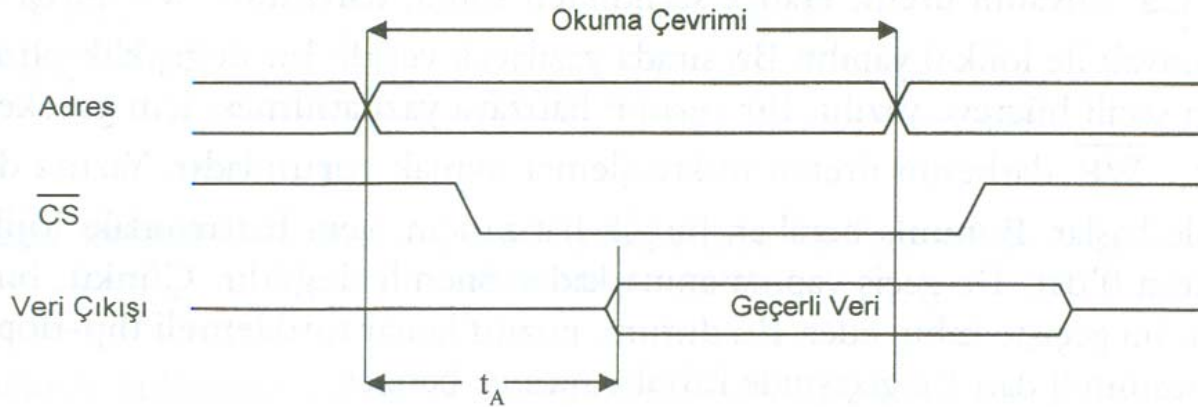


Adres hatları diğer bir hafıza hücreesine yazmak veya okumak için değiştirilebilir.

Bir **yazma** çevriminde aşağıda olaylar gerçekleşir:

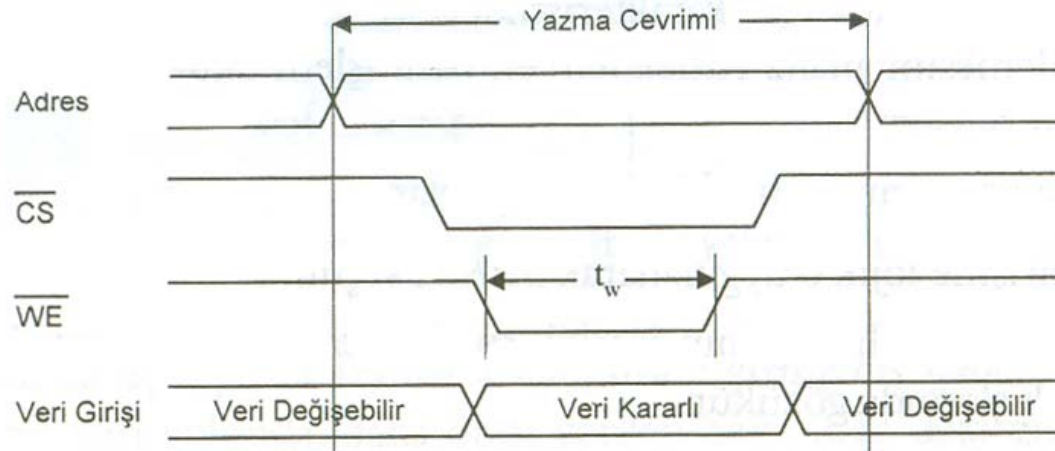
-  Hafıza girişlerine adres uygulanır.
-  $\overline{CS}$ girişine lojik 0 uygulanarak SRAM seçilir.
-  Hafızaya yazılacak veri, hafıza veri girişlerine uygulanır.
-  $\overline{WE}$ hattı aktiflenir.
-  Adres hatları, diğer bir hafıza hücresine yazmak veya okumak için değiştirilebilir.

Örnek (Read)



- ✓ Hafıza adresi, mikroişlemci tarafından sağlanır.
- ✓ Mikroişlemcinin çalışması sırasında bu hatları gözleyen bir kod çözücü, erişilmek istenen hafıza biriminin adresinin üretilmesinden kısa bir süre sonra, bu kodu çözerek $\overline{CS}$ sinyalini üretir.
- ✓ Adres üretiminden sonra, verinin çıkışta okumaya hazır olması için geçen zamana erişim zamanı t_A denir. En az bu kadar bir zaman sonra, mikroişlemci $\overline{RD}$ sinyalini aktif yaparak veriyi okur.

Örnek (Write)



- ✓ Adres ve kontrol sinyallerinin kodunu çözüen kod çözücü, $\overline{CS}$ sinyalini üretir.
- ✓ Hafıza seçildikten sonra, hafızanın $\overline{WR}$ sinyali aktiflenir. Bu sırada yazılacak veride bir değişiklik olmaz ve son olarak bu veri seçilmiş olan hücreye yazılır.
- ✓ Bir verinin hafızaya yazılabilmesi için gereken minimum süreye t_w , $\overline{WR}$ darbesini üreten mikroişlemci uymak zorundadır.

Bellek Haritası

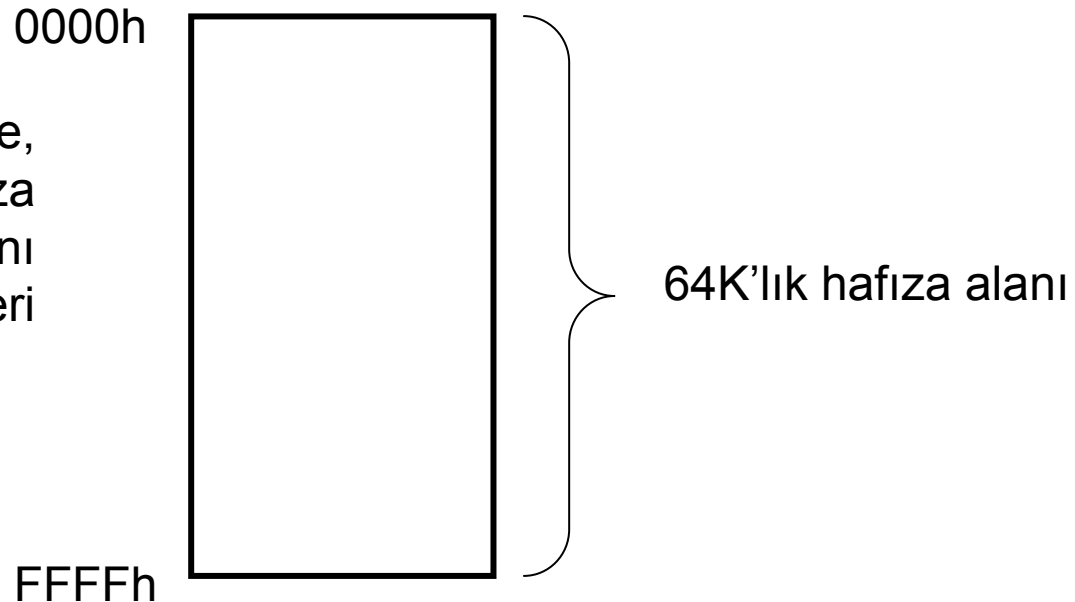
Aşağıdaki temel kavramların bilinmesine gereksinim vardır:

- Kullanılan mikroişlemcinin adres ve veri yolu genişliği
- İşlemcinin yürütmeye başlama adresi
- İşlemcinin hafıza birimleriyle haberleşmede kullandığı yol kontrol sinyalleri

✓ Bir mikroişlemcinin sağladığı adres bitlerinin sayısı, işlemcinin doğrudan adresleyebileceği hafıza hücrelerinin sayısını, diğer bir deyişle hafıza adres alanını (memory address space) belirtir.

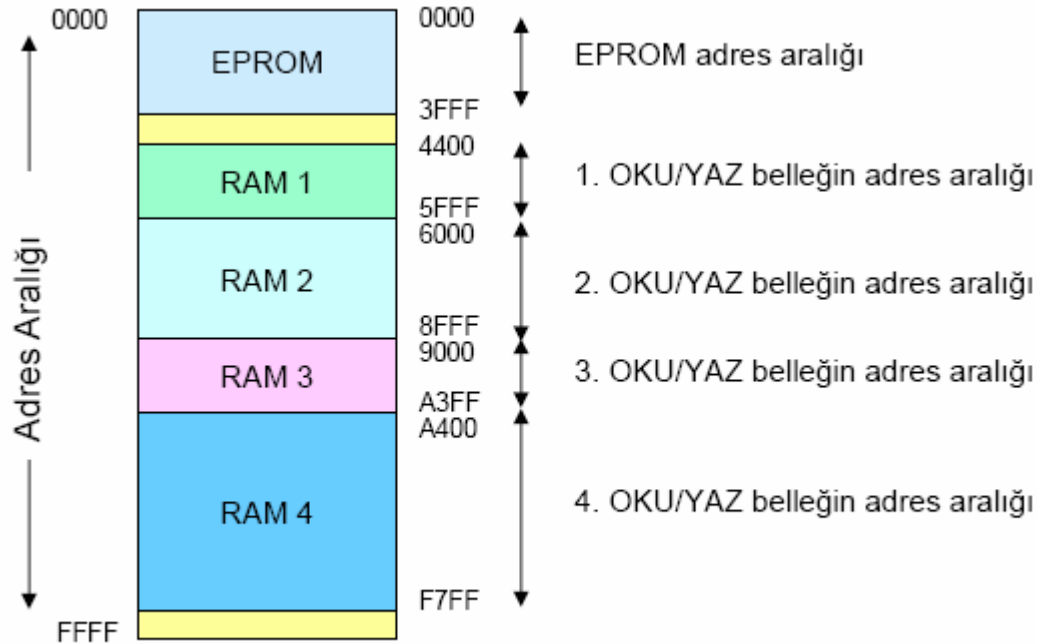
✓ Bir işlemciye ait hafıza alanının bir şekilde gösterilmesine hafıza haritası (memory map) denir. Bu harita şekli üzerinde, mikroşlemcili sistemin kullandığı veya kullanabileceği hafıza modüllerinin sınırları gösterilir.

✓ Bu alan içerisinde, değişik hafıza birimlerinin yanı sıra, I/O birimleri de bulunur.



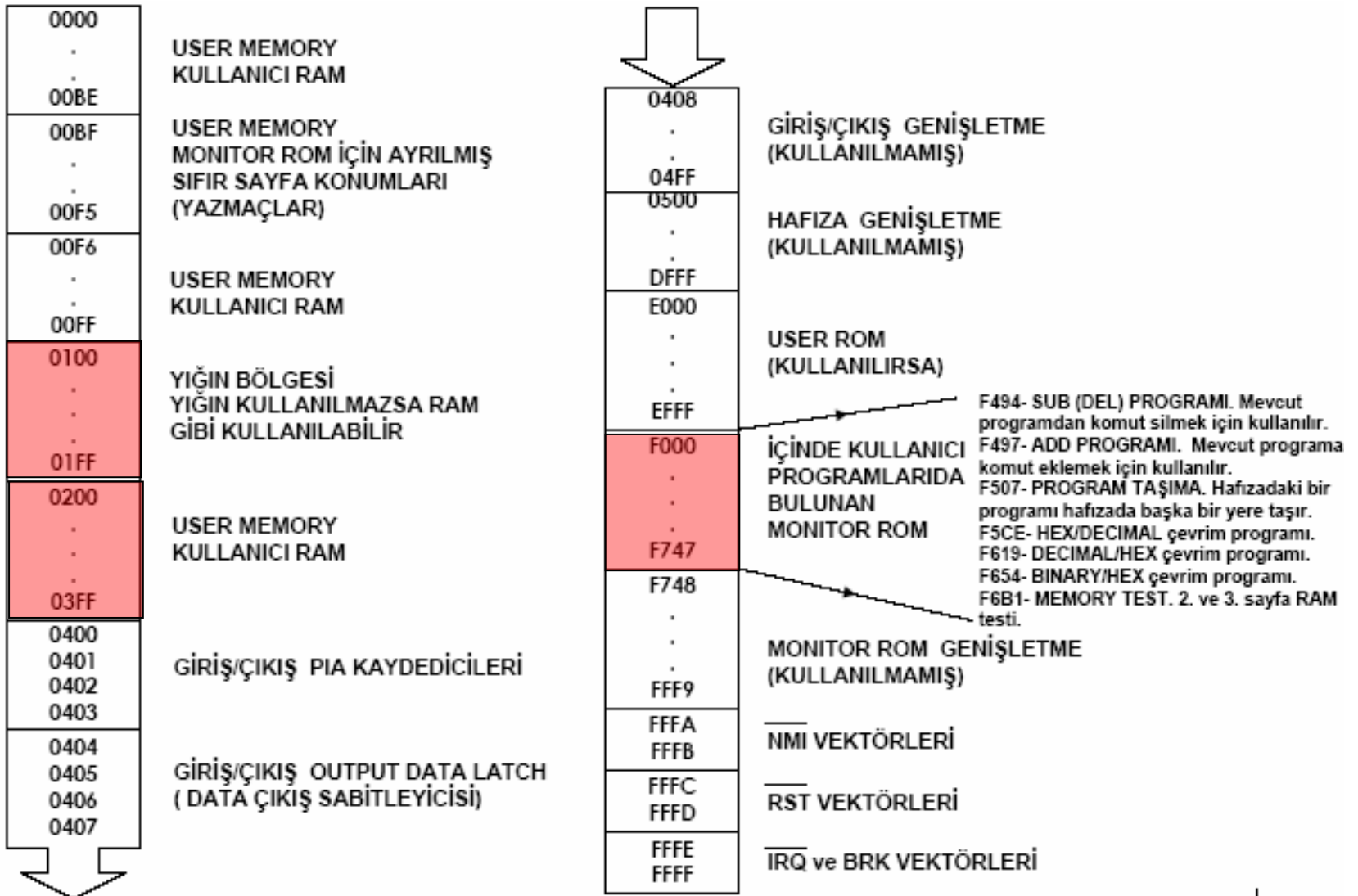
16 bit adres yoluna sahip bir işlemcinin hafıza haritası

✓ Bir mikroişlemci, program yürütmeye, sıfır veya yüksek bir adresten başlar. Bu adreste ise mutlaka bir program komutunun olması gerekir.



✓ Mikroişlemcinin çalışmaya başladığı hafızanın ilk adresinde, genelde sistem programlarının olduğu, bir çeşit ROM hafıza bulunur. **Örn:** 8086 ve 8088'de programa başlama adresi yüksek adrestir(FFFF0h).

6502 Mikroişlemcisi'nin Hafıza Haritası

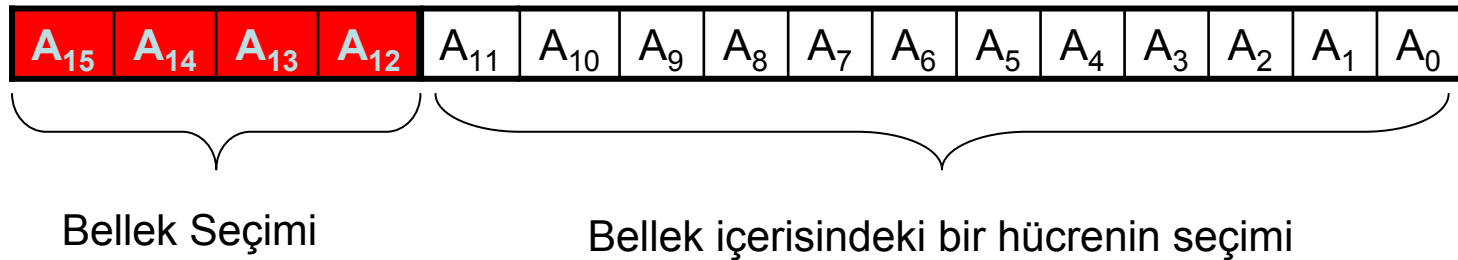


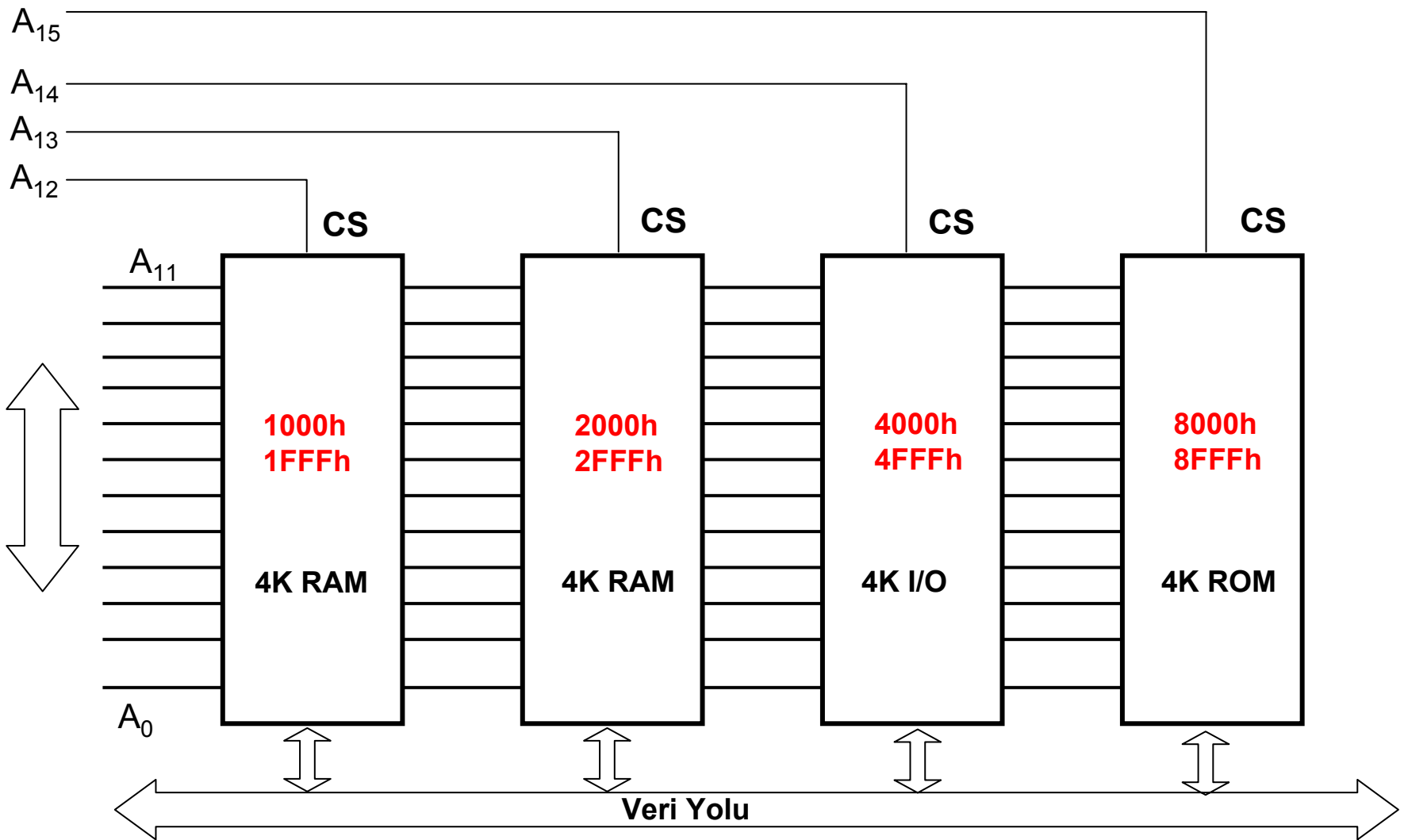
Hafıza Tasarımındaki Adımlar

- Uygulama için gereken ROM ve RWM miktarı yaklaşık olarak belirlenir.
- Her hafıza çeşidi için, mikroişlemcinin hafıza adres alanı göz önüne alınarak, hafıza sınırları belirlenir ve ilk hafıza haritası çizilir.
- Kullanılacak hafıza birimleri seçilir.
- Gereken kelime sayısı ve kelime uzunluğunu verecek şekilde hafıza birimleri birleştirilerek detaylı hafıza haritası oluşturulur.
- Adres ve kontrol sinyallerinin kodunu çözecek olan kod çözücü lojik tasarlanır.
- Ortak yoldan yalıtım sağlamak için Bufferlama kullanılabilir.
- Tasarım sırasında mikroişlemcinin hızı göz önüne alınmalıdır.

✓ Eğer kullanılan sistemde tek bir bellek lojiğine gerek varsa veya gerek duyulan bellek kapasitesini tek bir lojik sağlıyorsa bellek adreslemesi kolaylaşacaktır. Fakat birden fazla bellek lojiğinin olduğu durumlarda kod çözme işlemine gereksinim vardır.

✓ 16 bitlik adres hatlarının olduğu bir sistemde 4Kbit x 8=4KB'lık bellek lojiği bulunmaktadır. Bu 4K'lık alan 12 bit (2^{12}) ile adreslenebilir. Geriye kalan 4 bitlik adres hattı ise doğru bellek lojiğinin seçiminde kullanılır. Yani.





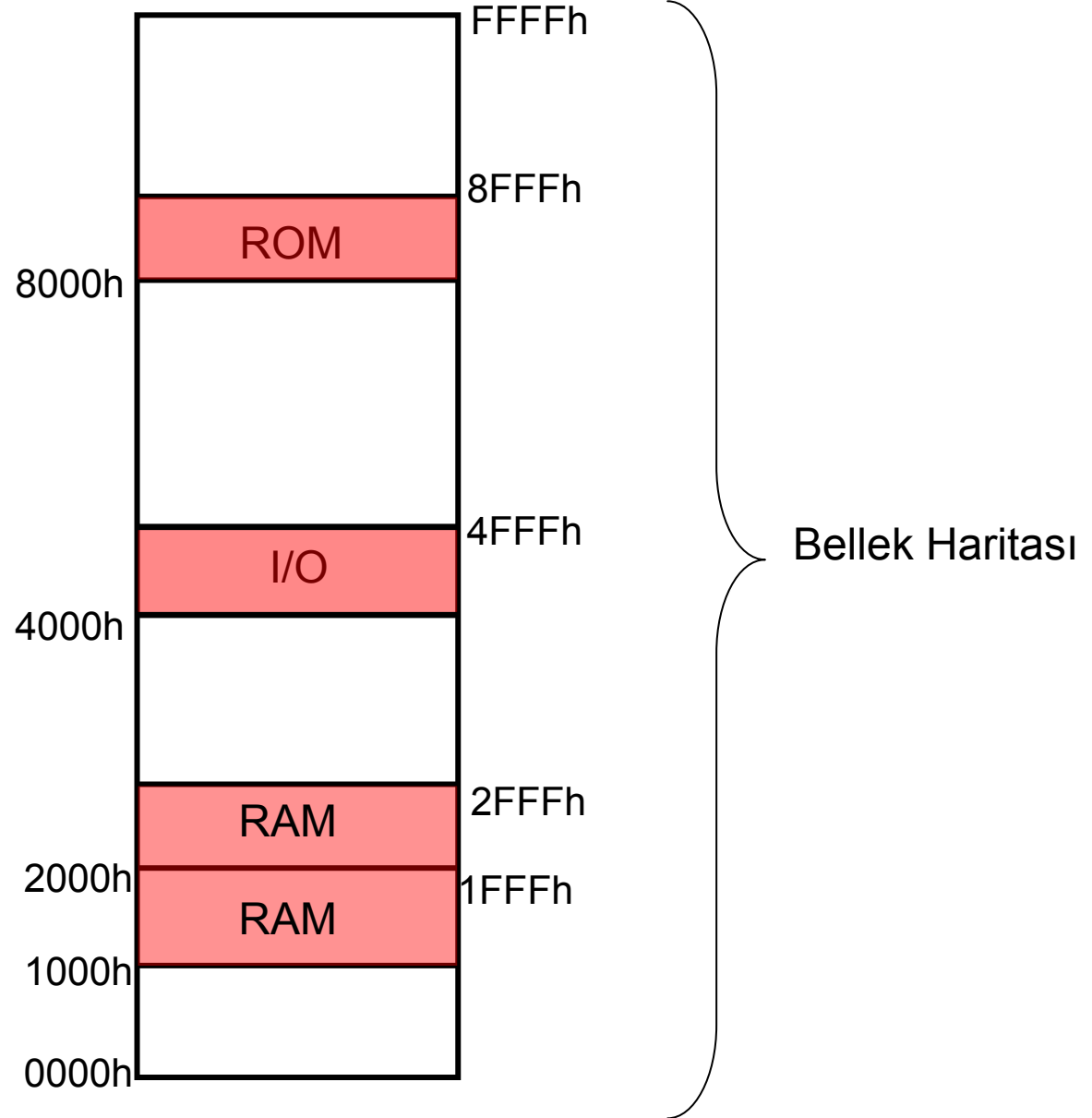
Adres kod çözücüsü kullanmadan yüksek değerlikli adres hatlarından 4 tanesini doğrudan CS girişlerine bağlayarak da kod çözümü yapılabilir.



Aynı anda 2 bellek çipinin seçilirse aranan bilginin hangisinde olduğu bilinmediğinden sistem kilitlenmesi olur.

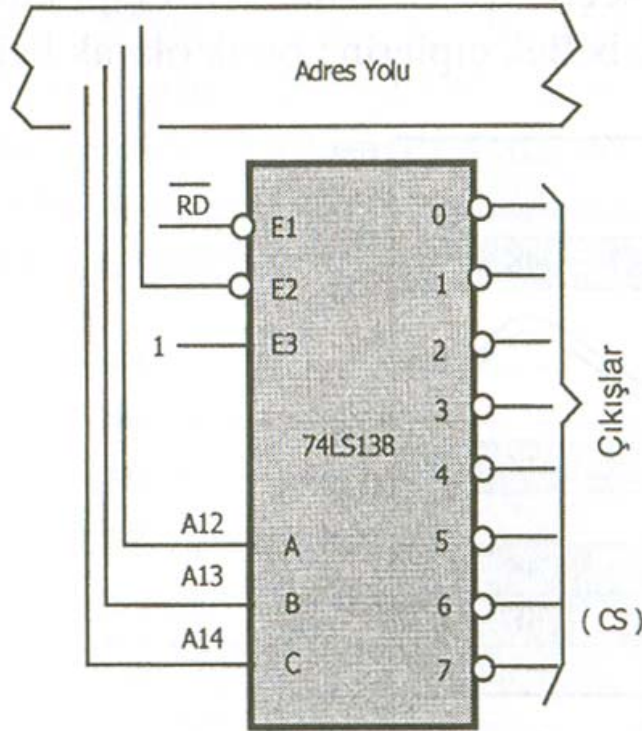


Kullanılmayan alanların bulunduğu adreslerden birisi seçildiğinde aynı anda birden fazla lojik seçilebilir.



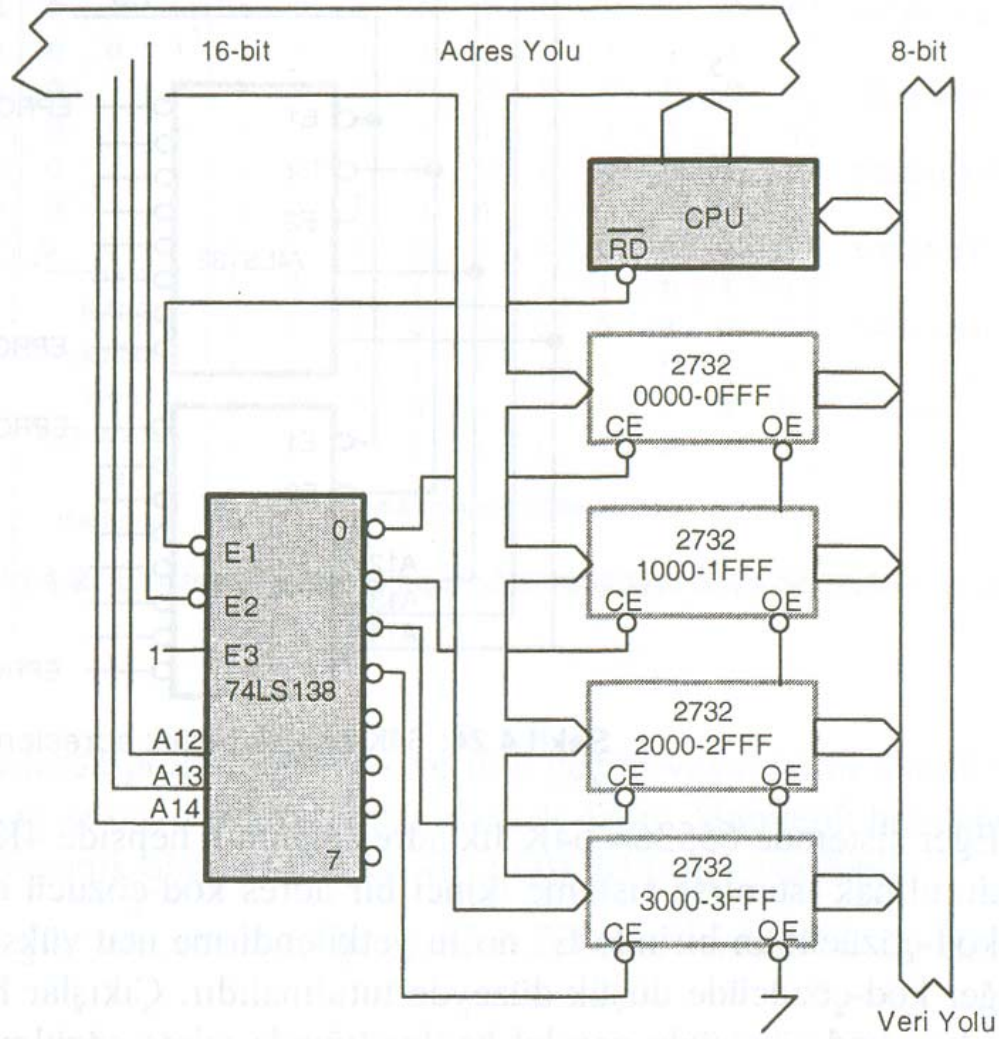


Önceden bahsedilmiş olan kod çözücü yapısı kullanarak da sistemin tasarımını yapmak mümkündür. Aşağıdaki kod çözücü 16Kx4'lük bellek lojiklerinin adreslenmesinde kullanılmaktadır.



3X8'lik bir kod çözücü

Girişler						Çıkışlar							
Yetkiler			Seçme										
E1	E2	E3	C	B	A	0	1	2	3	4	5	6	7
1	X	X	X	X	X	1	1	1	1	1	1	1	1
X	1	X	X	X	X	1	1	1	1	1	1	1	1
X	X	0	X	X	X	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	1	1	1	1	1	1	1
0	0	1	0	0	1	1	0	1	1	1	1	1	1
0	0	1	0	1	0	1	1	0	1	1	1	1	1
0	0	1	0	1	1	1	1	1	0	1	1	1	1
0	0	1	1	0	0	1	1	1	1	0	1	1	1
0	0	1	1	0	1	1	1	1	1	1	0	1	1
0	0	1	1	1	0	1	1	1	1	1	1	0	1
0	0	1	1	1	1	1	1	1	1	1	1	1	0



16 Kb'lık bellek lojikleri ile oluşturulan sistemin kod çözümü

Örnek

8 bitlik veri yolu ve 16 bitlik adres yolu olan bir mikroişlemcili sisteminde, 8K*8'lik RAM ve ROM lojikleri kullanarak 24K*8'lik RAM belleği ve 16K*8'lik ROM belleği tasarlanacaktır. Oku/Yaz belleğin başlangıç adresi \$0000, salt oku belleğin başlangıç adresi ise \$8000'dir.

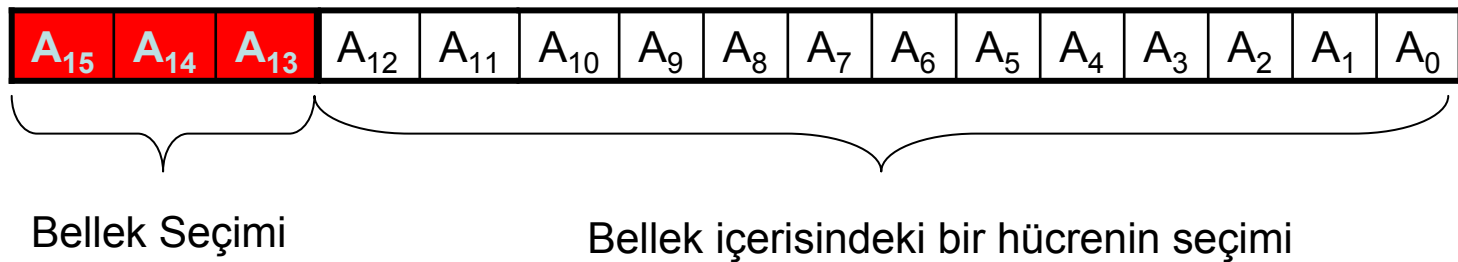
a) Buna göre sistemin bellek haritası nasıl olacaktır ?

b) Böyle bir sistem için gerekli olan kod çözücü yapısı nasıl olmalıdır?

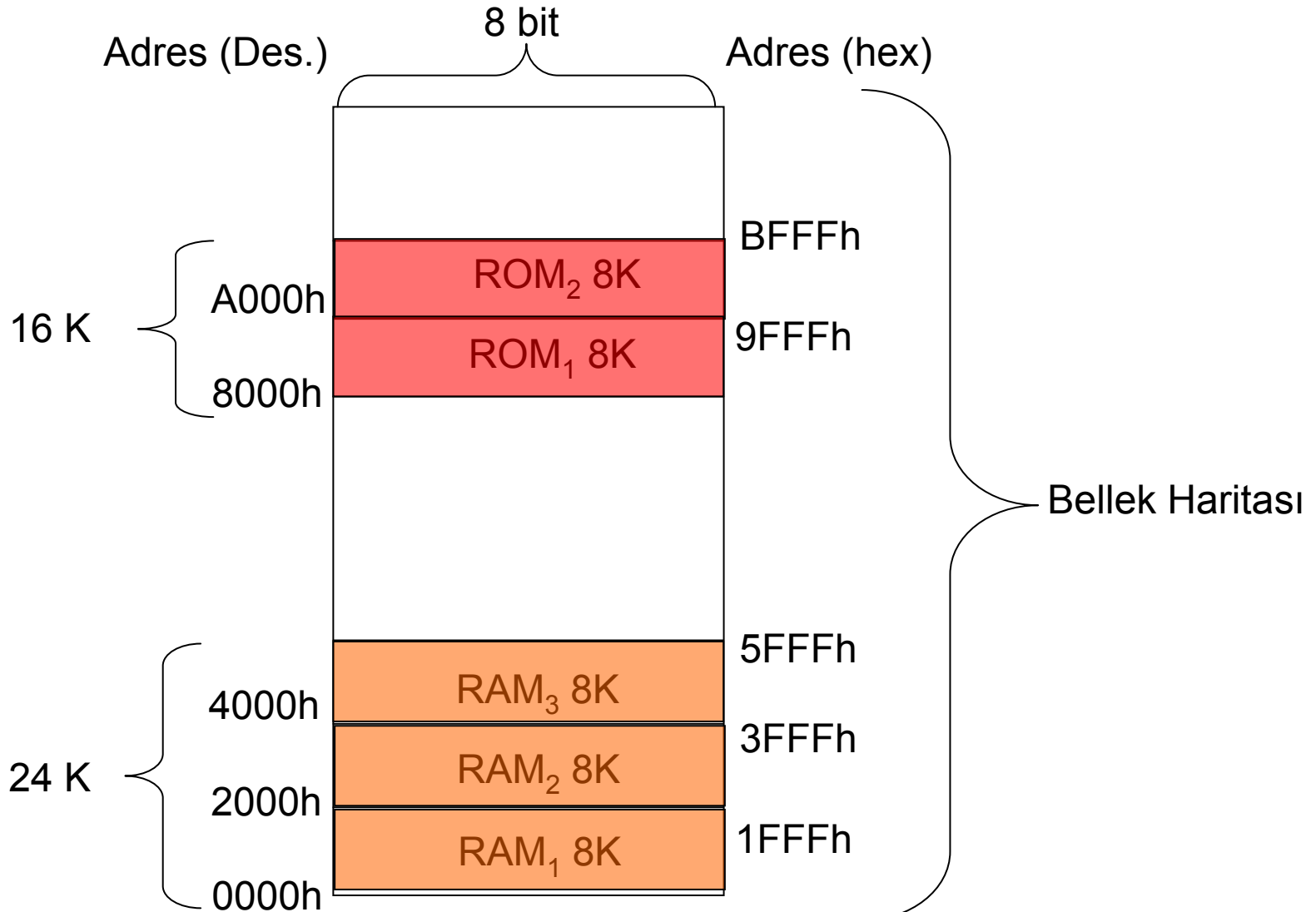
Çözüm



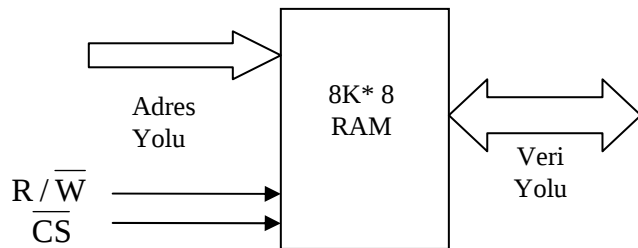
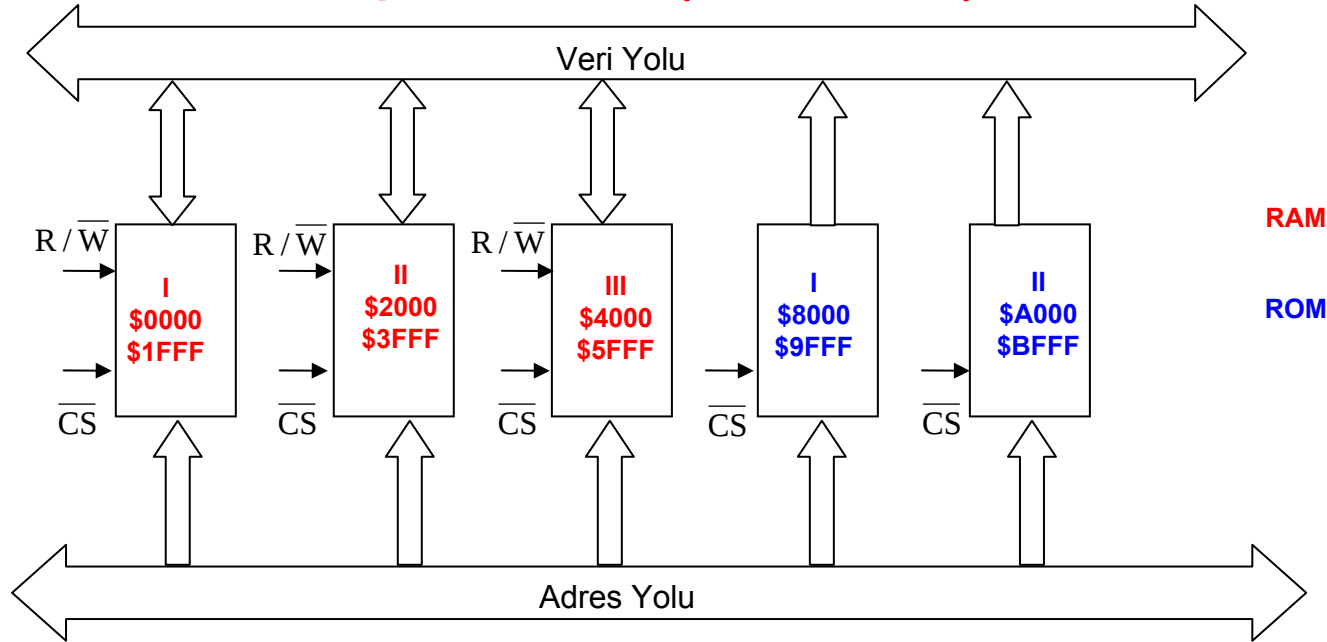
Öncelikli olarak 8K gözü olan bir bellek içerisindeki her bir konumu seçmek için 13 bit gerekmektedir (A_0 - A_{12}). Geriye kalan 3 bit ise farklı hücreleri seçmek için kullanılmalıdır.



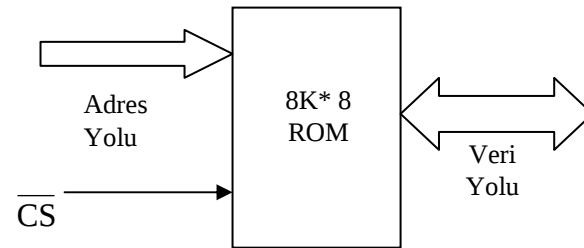
Çözüm (devam)



Çözüm (devam)



RAM Chipi'nin
Blok Diyagramı



ROM Chipi'nin
Blok Diyagramı

Çözüm (devam)

RAM ve ROM belleğe sahip böyle bir sistemin kullanım tablosu aşağıdaki gibi olsun.

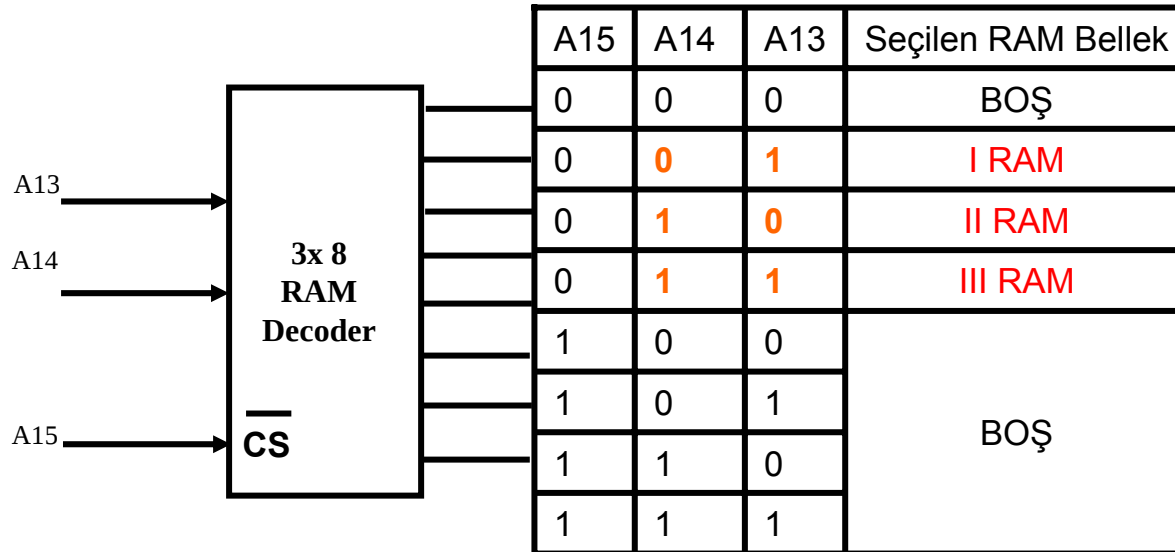
A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	Bellek
0	0	1	X	X	X	X	X	X	X	X	X	X	X	X	X	I (RAM)
0	1	0	X	X	X	X	X	X	X	X	X	X	X	X	X	II(RAM)
0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	X	III(RAM)
1	1	0	X	X	X	X	X	X	X	X	X	X	X	X	X	I (ROM)
1	1	1	X	X	X	X	X	X	X	X	X	X	X	X	X	II (ROM)



Bu durumda RAM ve ROM belleğin seçilebilmesi için A15, A14 ve A13'yi kod çözme işlemine tabi tutmak yeterlidir.

Çözüm (devam)

✓ Bu durumda RAM belleğin seçilebilmesi için A15, A14 ve A13'yi kod çözme işlemine tabi tutmak yeterlidir.



Çözüm (devam)



ROM belleğin seçilebilmesi için sadece A13' ü kod çözme işlemine tabi tutmak yeterlidir.

