

MİKROİŞLEMCİ SİSTEMLERİ

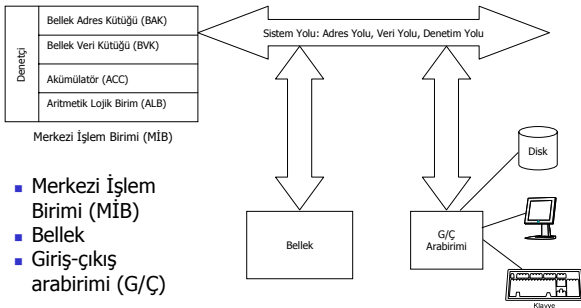
Yrd. Doç. Dr. Şule Gündüz Öğüdücü
<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Bilgisayar Yapısı

- Bilgisayar verilen verileri, belirlenen bir programa göre işleyen, istenildiğinde saklayabilen, gerektiği zaman geriye verebilen sayısal bir alettir.
 - Belleme yeteneği
 - Hesaplama yeteneği
 - Karar verme yeteneği
 - Giriş-çıkış yeteneği

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Bilgisayar Temel Birimleri



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

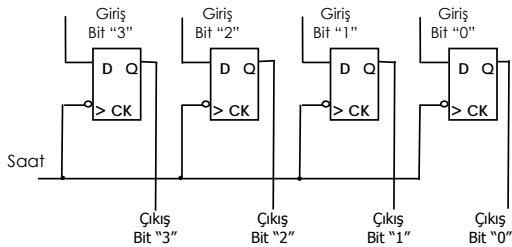
MİB Yapısı

- Bellek Adres Kütüğü (BAK): MİB'den giden veya MİB'e gelen verilerin gideceği veya geldiği yerin adresini belirtir. Adres yolunu bağıdır.
- Bellek Veri Kütüğü (BVK): MİB'den giden veya MİB'e gelen verilerin alınıp verildiği bir iskele. Veri yoluna bağıdır.
- Akümlatör (ACC): BVK üzerinden gelen veriler akümülatöre alınır.
- Aritmetik Lojik Birim (ALB): ACC'deki veri üzerinde yapılacak işlemler ALB tarafından yürütülür.
- Denetçi (DEN): Tüm bilgisayarın yönetimi ile ilgili bilgileri toplamak ve üretmekle görevlidir.

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Kütükler

- Aynı saat işaretinin uygulandığı flip-flop'lardan oluşur.



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Kütükler

- MİB, ALB, denetçi ve MİB'nin diğer birimleri tarafından çabuk erişilebilen saklama alanlarıdır.
- Farklı mikroişlemcilerde kütüklerin sayısı ve boyutu da farklı olabilir.
- Genel Amaçlı kütükler: MİB'nin yürüttüğü işlemlerde veri saklamak için kullanılırlar (Akümülatör).
- Özel amaçlı kütükler: Her MİB işleminde gerekli olan veriyi saklamak için kullanılırlar. (BAK, BVK).

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Aritmetik Lojik Birim

- MİB tarafından yürütülen bütün işlemleri yerine getirir.
 - Aritmetik
 - Lojik
- Bilgisayar mimarisi ALB'in boyunu belirler.

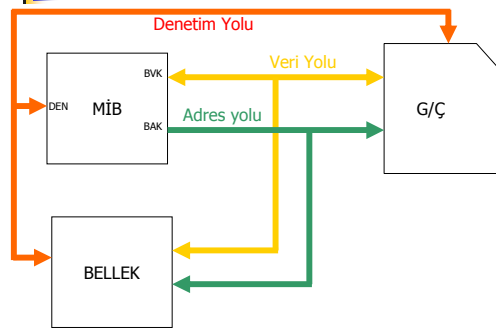
<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Denetçi

- MİB içindeki kontrol işlemleri
 - Komutların sıralanması ve yürütülmesi
 - Veri yolu kontrolü
 - MİB içindeki yolların kontrolü
 - :
- MİB dışındaki kontrol işlemleri
 - Sistem yolunun kontrolü
 - Kesme işlemleri
 - Sistemin durumu
 - Bellek yönetimi
 - :

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Bilgisayar Bağlantıları



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

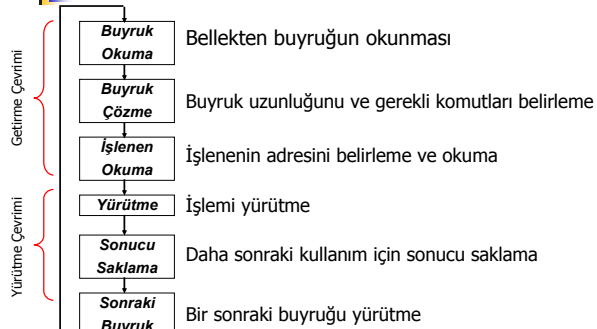
Bilgisayarın Çalışma Düzeni

- Örnek: X ve Y sayılarının 3000 ve 3001 nolu bellek gözlerine yazılması, sayıların toplanması sonucun 3002 nolu bellek gözüne yazılması

Bellek Adresi	İşlem
1001	X'i ACC'ye yükle
1002	ACC'nin içeriğini 3000 nolu bellek gözüne yaz
1003	Y'i ACC'ye yükle
1004	ACC'nin içeriğini 3001 nolu bellek gözüne yaz
1005	3000 sayılı bellek gözünün içeriğini ACC'ye yükle
1006	3001 sayılı bellek gözünün içeriğini ACC'ye ekle
1007	ACC'nin içeriğini 3002 nolu bellek gözüne yaz

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Buyruk Yürütülmesi



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Buyruklar

- Bilgisayara bir işlem yaptırmak için kullanılan adımlar
 - 3+1 adresli buyruk kalıbı
 - 3 adresli buyruk kalıbı
 - 2 adresli buyruk kalıbı
 - 1 adresli buyruk kalıbı
 - 0 adresli buyruk kalıbı

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Buyruklar

- 3+1 Adresli buyruk kalıbı: TOP X,Y,Z,N

Komut	1.İşlenen adresi	2.İşlenen adresi	Sonucun adresi	Bir sonraki buyruğun adresi
-------	------------------	------------------	----------------	-----------------------------

- 3 Adresli buyruk kalıbı: TOP X,Y,Z

Komut	1.İşlenen adresi	2.İşlenen adresi	Sonucun adresi
-------	------------------	------------------	----------------

- 2 Adresli buyruk kalıbı: TOP X,Y

Komut	1. İşlenen adresi	2. İşlenen adresi
-------	-------------------	-------------------

- 1 Adresli buyruk kalıbı: TOP A,X

Komut	Kütük Adı	İşlenen adresi
-------	-----------	----------------

- 0 Adresli buyruk kalıbı: YİĞ A, ÇEK A

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Bellek

- Bilgisayarın çalışmasına yön verecek programı ve programın üzerinde çalışacağı verileri saklar.
 - Salt Oku Bellekler
 - Salt Oku Bellek (ROM)
 - Programlanabilir Salt Oku Bellek (PROM)
 - Silinebilir Programlanabilir Salt Oku Bellek (EPROM)
 - Oku/Yaz Bellekler (RAM)
 - Statik Oku/Yaz Bellek (SRAM)
 - Dinamik Oku/Yaz Bellek (DRAM)

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

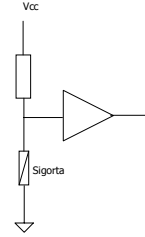
Salt Okunabilir Bellek

- Read Only Memory (ROM)
 - Yarı iletken malzeme kullanılır. Veriler üretim sırasında yazılır.
 - Veri okunabilir ancak yazılamaz.
 - Kullanım alanları:
 - Özel amaçlı bilgisayarlar
 - Sabit programla çalışan bilgisayarlar
 - Sistem programları

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Programlanabilir Salt Oku Bellek (PROM)

- Programmable ROM
- Her bellek gözesi için bir sigorta
 - Üretildikleri zaman bütün gözleri 1 veya 0
 - Üretim sonrasında özel bir aletle sigorta konumu değiştirilerek programlanır.
- Sadece bir kere programlanabilir.
- Veriler silinemez.



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Silinebilir Programlanabilir Salt Oku Bellek

- Erasable PROM (EPROM)
- Üretildiklerinde tüm bellek gözleri 1 konumunda
 - Elektriksel olarak programlanabilir.
 - Mor ötesi ışık ile silinir.
 - Silme ve programlama tekrarlanabilir. (belli bir sayıda)
 - Yazmadan önce tüm bellek silinir ve tekrar yazılır.

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Elektriksel Silinebilir Programlanabilir Salt Oku Bellek

- Electrically Erasable PROM (EEPROM)
- Daha önceki veri silinmeden programlanabilir.
- İstenen bellek gözlerine istenen değer yazılabilir.
- Programlama tekrarlanabilir. (belli bir sayıda)

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Flash Bellek

- Programlarken hızlı olması nedeniyle flash bellek olarak adlandırılıyor.
- Elektriksel olarak siliniyor.
- Veri bloklar halinde silinebilir.
- Her bit için bir tranzistor kullanılır.

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

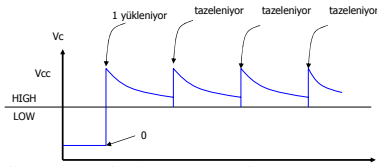
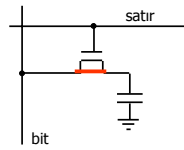
Oku/Yaz Bellek (RAM)

- Random Access Memory (RAM)
- Okuma ve yazma işlemi elektriksel olarak yapılır.
- Güç kesildiği anda veriler kaybolur.
- İki tip RAM
 - Statik RAM
 - Dinamik RAM

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Dinamik Oku/Yaz Bellek (DRAM)

- Bir kapasite ve tranzistörden oluşur.
- Veri kapasitenin doldurulması ile saklanıyor
- Kapasite kaçak akımlar nedeniyle boşaldığı için belleğin tazelenmesi gerekiyor.



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

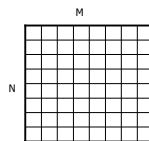
Statik Oku/Yaz Bellek (SRAM)

- Herbir göze bir flip-flop.
- Verinin kaybolmaması için belleğin tazelenmesi gerekmemektedir.
- Daha hızlı ve daha pahalıdır.
- Genellikle cep bellekler için kullanılır.

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Bellek Düzeni

- Belleğin temel birimi göze (bit)
- Gözelerin yanyana gelmesiyle gözler oluşur
- Bellek gözlerinin üst üste gelmesiyle bellek oluşur
 - NxM boyutunda matris
 - N: satır sayısı (göz sayısı)
 - M: Sütun sayısı (göze sayısı)



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

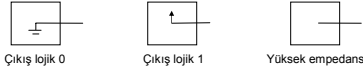
Belleğe Erişme

- MİB'nin belleğe (RAM ya da ROM) erişme (okuma ya da yazma işlemleri için) adımları :
 - Doğru bellek birimini seçmesi (adres yolundaki hatların bir kısmını kullanarak)
 - Bellek içinde doğru gözü seçmesi (adres yolundaki diğer hatları kullanarak)
 - Veriye erişme (veri yolunu kullanarak)

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Üç Konumlu Kapılar

- Bellek birimlerinde yaygın olarak kullanılan bir devre
- Üç konumu bulunan lojik kapı
 - Lojik 0, lojik 1 ve yüksek empedans
 - Yüksek empedans konumunda çıkıştan tamamen yalıtılmış



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Üç Konumlu Kapılar

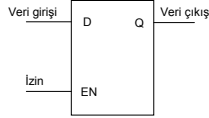
- İki girişi bir çıkışı bulunan bir devre
 - Lojik kapı girişi
 - İzin girişi
 - Lojik 0 ise kapı girişine uygulanan girişin etkisi çıkışa aktarılır
 - Lojik 1 ise çıkış yüksek empedans konumunda



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Temel Bellek Elemanı

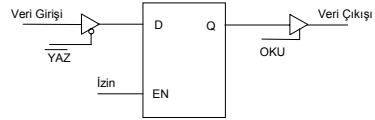
- Temel Bellek elemanı D tutucu
 - Giriş
 - Çıkış
 - İzin girişi



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Temel Bellek Elemanı

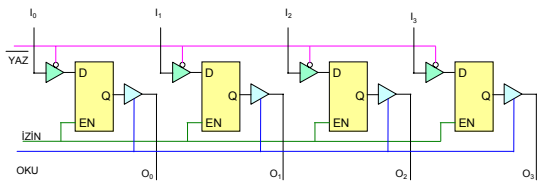
- Giriş ve çıkış veri yoluna bağlı
- Girişe ve çıkışa üç konumlu kapılar bağlanarak giriş ve çıkış istenildiği zaman veri yolundan yalıtılıyor.



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

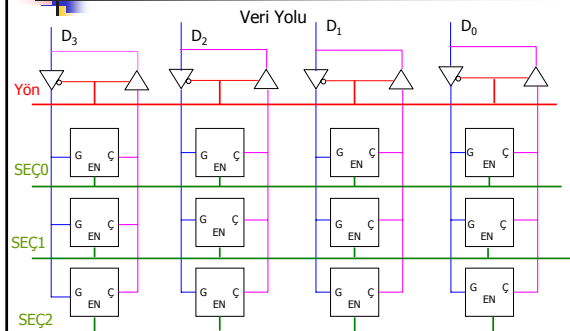
Bellek Gözü

- 4 tutucu birlikte bağlanırsa 4 bitlik bir bellek gözü elde edilir



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

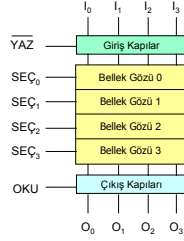
Bellek Düzeni



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Bellek Adresleme

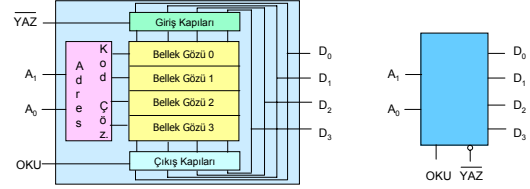
- OKU ve YAZ işaretleri ile verinin yönü belirlenir: bellekten okunur ya da belleğe yazılır
- uygun SEÇ girişi kullanılarak istenilen bellek gözü seçilir
 - aynı anda sadece tek bir SEÇ girişi aktif hale getirilir
 - bellek kırımlığına gelen adres yolu sayısını azaltmak için SEÇ işareti kodlanarak oluşturulur



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

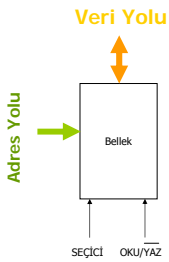
Bellek Tasarımı

- giriş ve çıkışta üç konumlu kapılar bulunduğu için bellek girişi ve çıkışı için aynı uçlar kullanılabilir



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Bir Belleğin Dıştan Görünümü

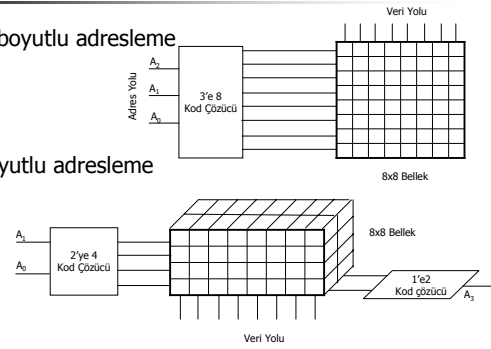


- Veri Yolu: Bellek gözlerinin bağlantısını sağlar.
- Adres Yolu: Bellek gözlerinin seçilmesi için adres bilgilerini taşır.
- Oku/Yaz: Veri akışını belirler.
- Seçici: Etkin olduğu durumda belleğin seçilmesini sağlar.

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Adresleme

- Tek boyutlu adresleme
- 2 Boyutlu adresleme



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Bellek Boyutu

- bellek boyutu boyu ve eni ile ölçülür (boy x en)
 - bellek gözü sayısı: boy
 - her gözdeki göze sayısı: en
- bellek boyutu bellek kırımlığına gelen adres yolu sayısının bir fonksiyonu
 - $\text{bellek gözü sayısı} = 2^{(\text{adres yolu sayısı})}$
 - adres yolu sayısı 10 olan bir bellekteki göz sayısı $2^{10} = 1024$ bellek gözü (1K)
 - bellek gözü sayısı 4096 olan bir belleği adres yolu sayısı $\log_2 4096 = 12$ adres yolu

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

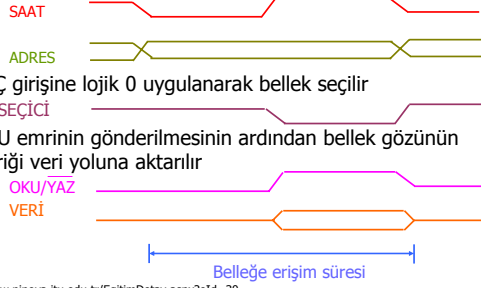
Belleğe Erişme

- Belleğe Erişme Zamanı: Bir bellek gözünün içeriğinin okunması ve bir bellek gözüne veri yazılması için gereken süre.

<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Okuma İşlemi

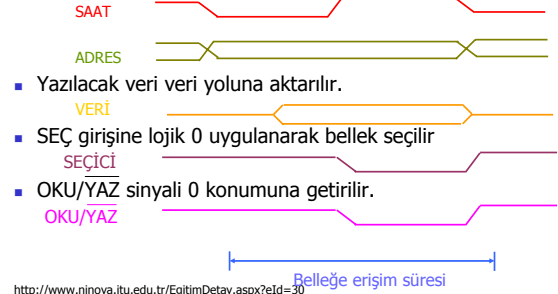
- Saat işaretinin 1'den 0'a inmesiyle adres yoluna adres bilgileri konur.



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>

Yazma İşlemi

- Saat işaretinin 1'den 0'a inmesiyle adres yoluna adres bilgileri konur.



<http://www.ninova.itu.edu.tr/EgitimDetay.aspx?eId=30>