

# Bölüm 3: Bellek Yönetimi

İşletim Sistemleri

# Bellek Yönetimi

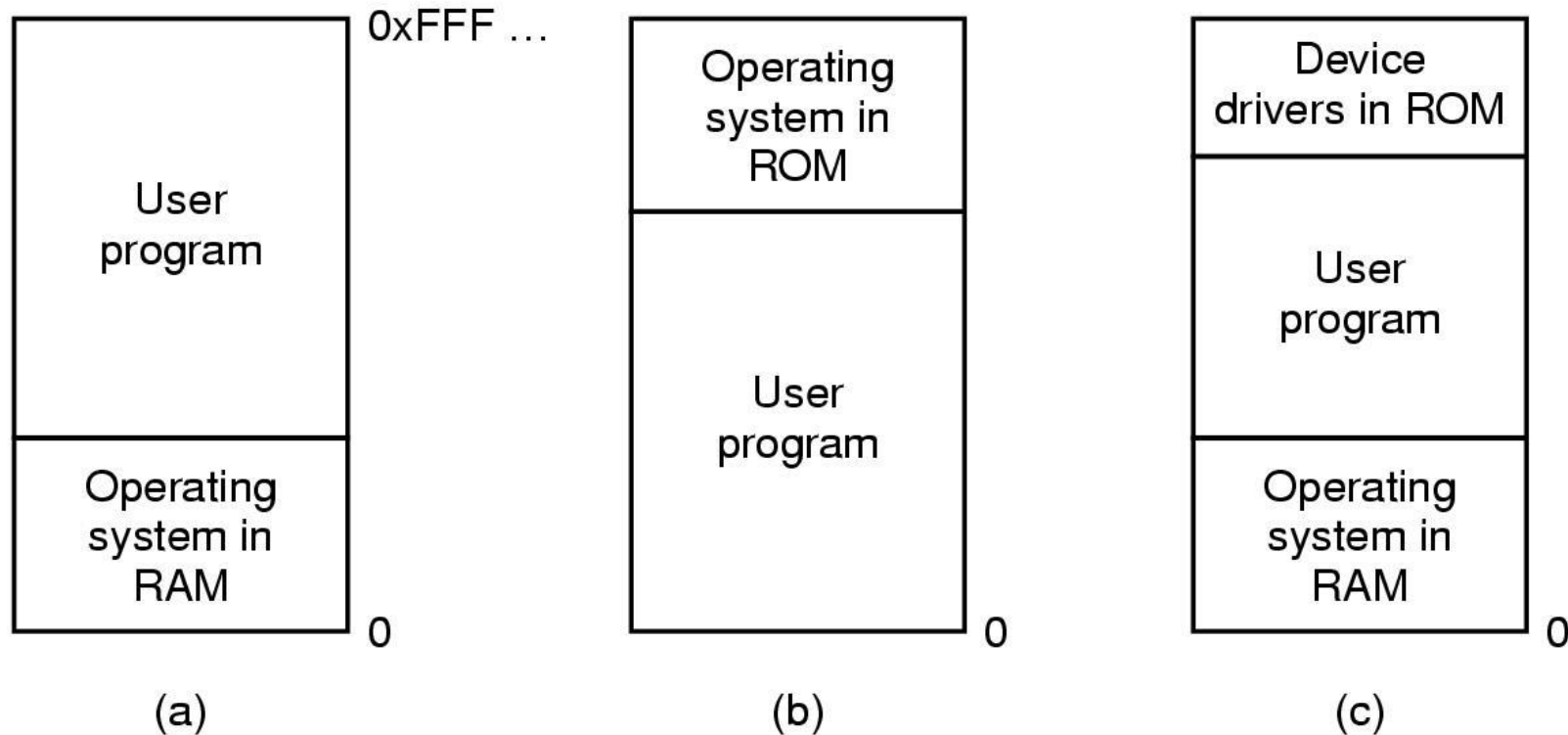
- Bellek (RAM) önemli ve nadir bulunan bir kaynaktır
  - Programlar, genişleyerek kendilerine sunulan belleği doldururlar
- Programcının istediği
  - Bellek gizli, sonsuz büyük, sonsuz hızlı, ve kalıcı olmalıdır...
- Gerçekte olan
  - İnsanların aklına gelen en iyi şey: bellek hiyerarşisi
  - Yazmaç, önbellek, bellek, disk, teyp
- Bellek yöneticisi
  - Belleği verimli bir şekilde yönetir
  - Boşalan bellek alanlarını takip eder, programlara bellek tahsis eder

# Soyutlama Yapılmadığında

- Eski anabilgisayar, mini bilgisayarlar, kişisel bilgisayarlarda bellek soyutlaması yoktu...
  - MOV REGISTER1, 1000
    - fiziksel bellek adresi 1000'in içeriğini yazmaca taşır
- Bellekte aynı anda iki program yer alamaz

# Soyutlama Yapılmadığında

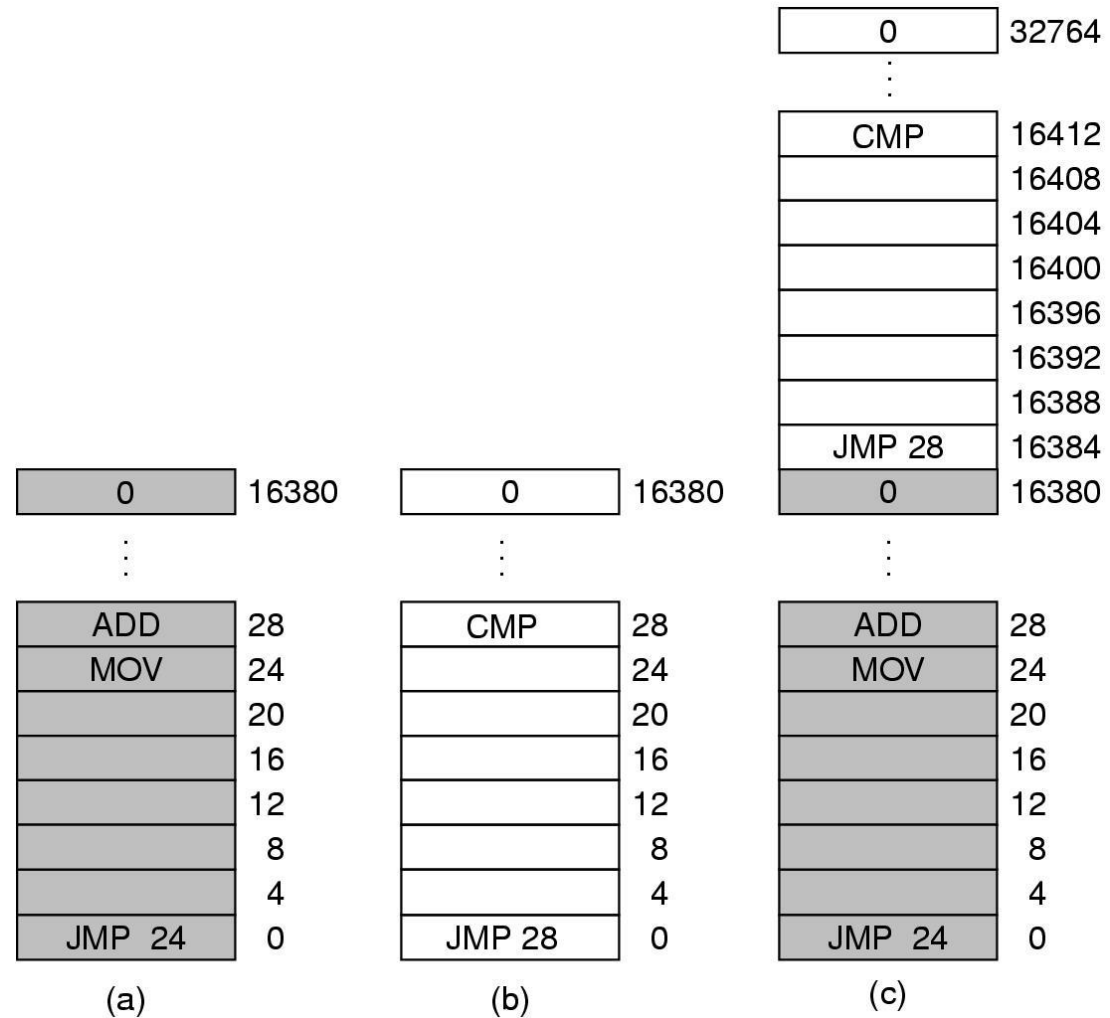
- İşletim sistemi ve bir süreç ile belleğin düzenlenmesi



# Soyutlama Yapılmadığında Problemler

- IBM 360
- 2 KB bloklara bölünmüş bellek ve her biri 4 bit koruma anahtarına sahip
- PSW 4 bitlik bir koruma anahtarına sahip
- Donanım, PSW anahtarından farklı bir koruma koduyla belleğe erişme girişimlerini yakalar.

# Yer Değiştirme Problemi



# Soyutlama Olmamasının Dezavantajı

- Sorun, her iki programın da mutlak fiziksel belleğe referans vermesidir.
- İnsanlar mahrem bir alana, yani yerel adreslere sahip olabilmek isterler.
- IBM 360
  - İkinci program belleğe yüklenirken adresler değiştirilir
  - Statik yer değiştirme
    - 16384'e bir program yüklenirken, her adrese bir sabit değer eklenir.
    - Yükleme yavaşlatır, ek bilgi gerektirir
- Gömülü ve akıllı sistemlerde soyutlama olmadan bellek yönetimi var

# Soyutlama: Adres Uzayı

- Fiziksel adresi programcılara gösterme (not expose)
  - işletim sistemi çökmesi
  - Parallelleştirmek zor
- Çözülmesi gereken iki problem:
  - Koruma
  - yer değiştirme
- Adres alanı:
  - Belleği adreslemek için bir dizi bellek süreci kullanılabilir
  - Her sürecin birbirinden bağımsız kendi adres alanı vardır.
  - Nasıl?

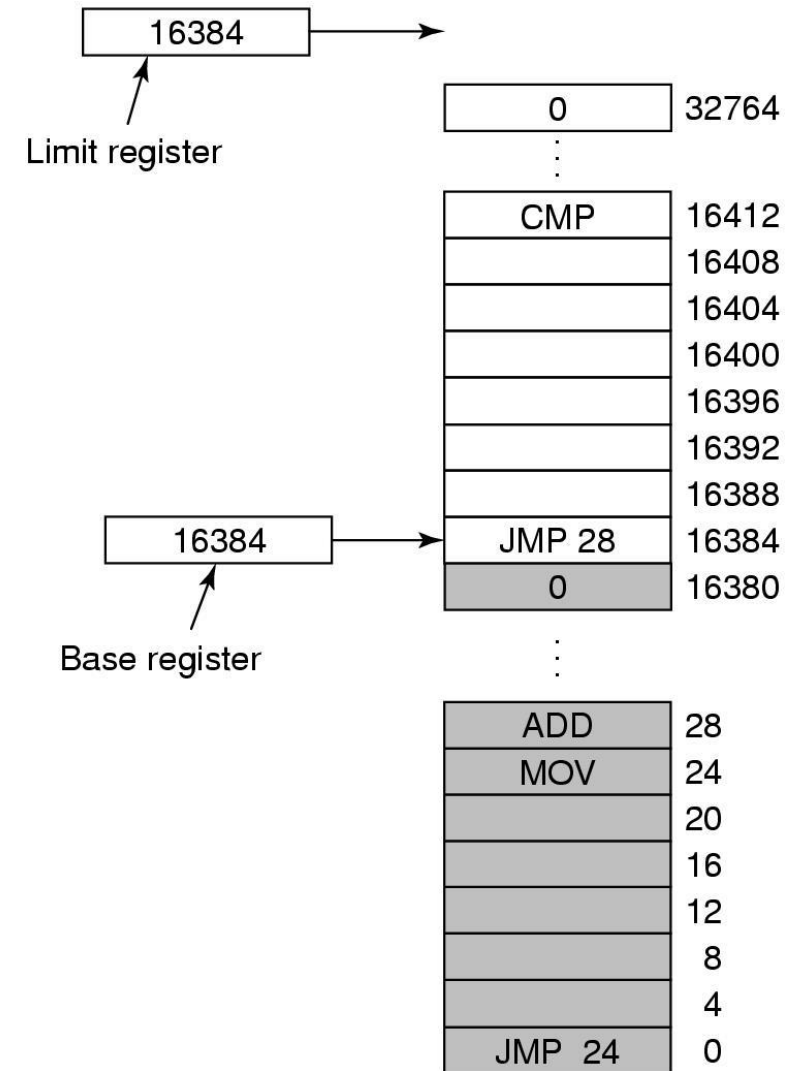


# Dinamik Yer Değiştirme

- İşlemciye iki özel yazmaç:
  - taban ve limit
- Program ardışık bir boşluğa yüklenecek
- Yükleme sırasında yer değiştirme yok
- Süreç çalıştırıldığında ve bir adrese referans verildiğinde, CPU otomatik olarak limit değerini aşıp aşmadığını kontrol ederek taban değerini o adrese ekler

# Taban ve Limit Yazmaçları

- Her bellek erişiminde bir ekleme ve karşılaştırma yapılması gerekiyor



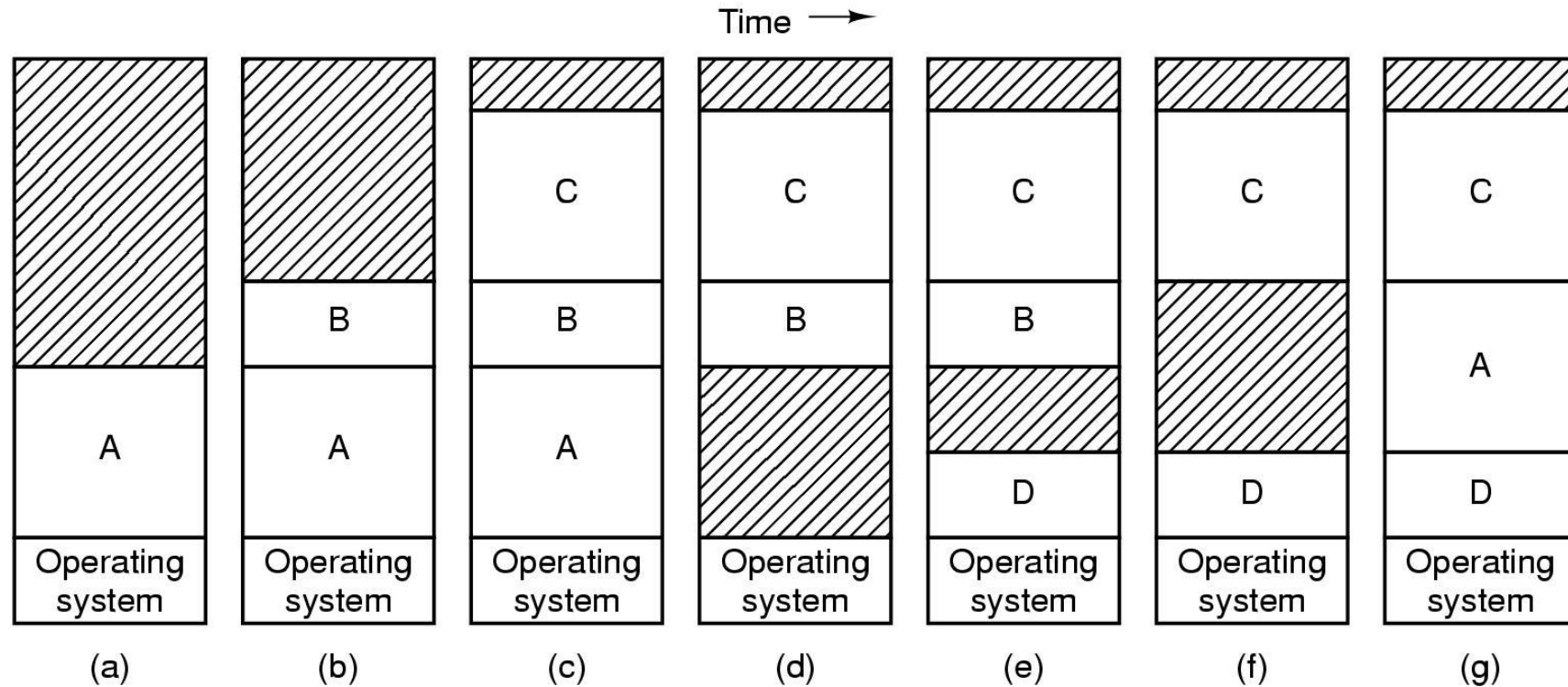
(c)

# Değiş Tokuş Yapmak (swap)

- Sistemde arka planda çalışan bir çok sunucu süreci vardır
- Fiziksel bellek tüm programları tutacak kadar büyük değildir
  - Değiş tokuş
    - Programları belleğe getir, değiş tokuş yapıp götür
  - Sanal bellek
    - Programları kısmen bellekte olsalar bile çalıştır

# Bellek Düzeni Değişimi

- Süreçler belleğe girip çıktıkça bellek tahsisi değişir. Gölge bölgeler kullanılmayan bellektir.

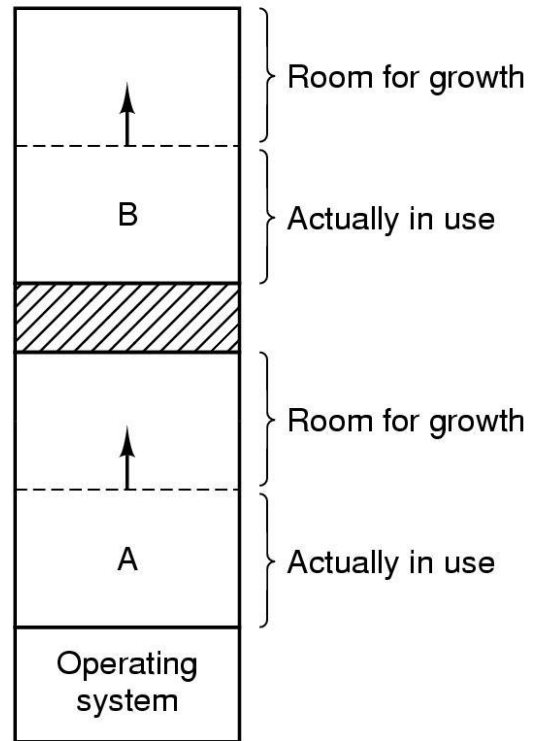


# Problemler

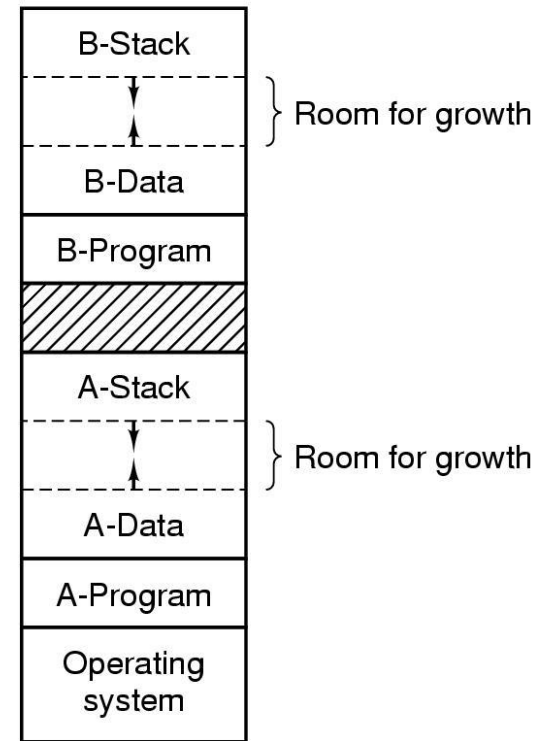
- Giriş ve çıkış takası sonrası farklı adresler
  - Statik yer değiştirme/dinamik yer değiştirme
- Bellek delikleri (hole)
  - Bellek sıkıştırma
    - İşlemci zamanı gerekir
    - 20 ns'de 4 byte taşır, ardından 1 GB'ı sıkıştırmak için 5 saniye
- Bir program için ayrılan bellek miktarı ne kadar?
  - Programlar büyüme eğilimindedir
  - Hem veri kesimi (segment) hem de yığın

# Bellekte Alan Tahsisi

- (a) Büyüyen veri segmenti için. (b) Büyüyen yığın, ve veri kesmi için.



(a)



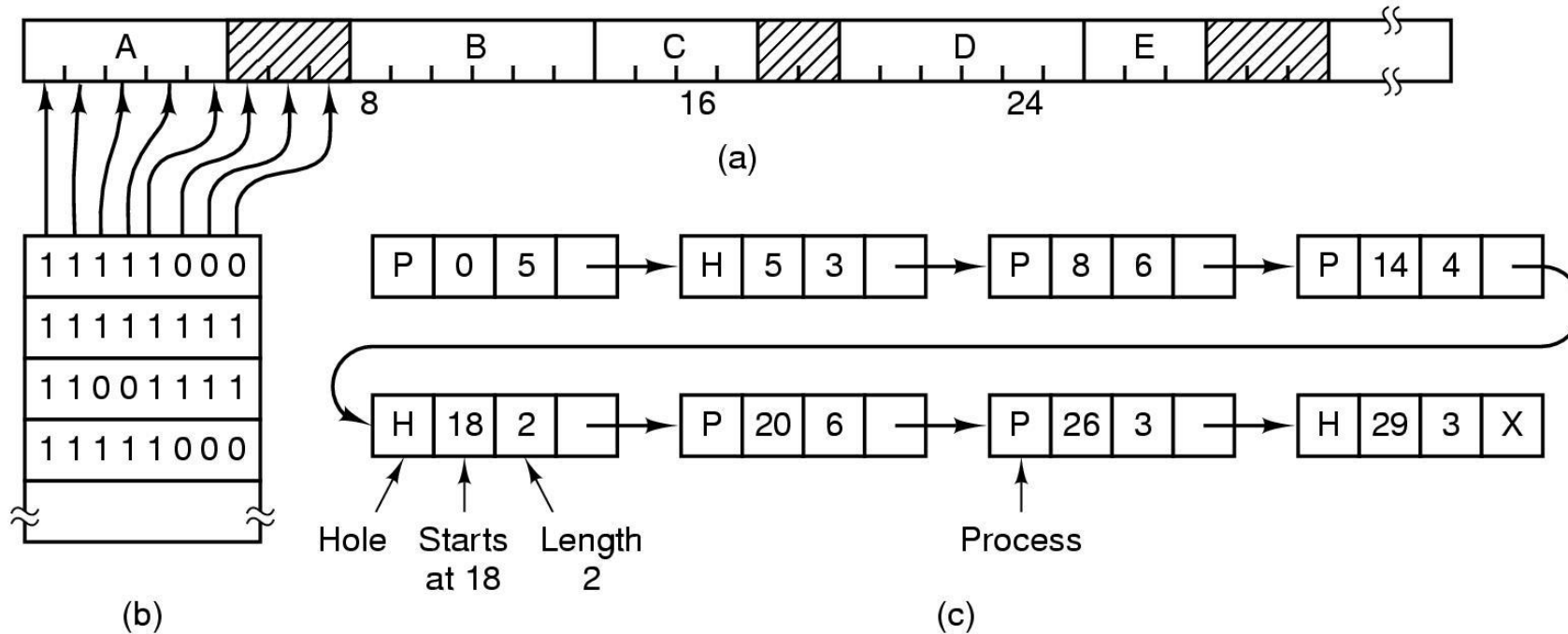
(b)

# Bellekte Boş Alan Yönetimi

- Biteşlem (bitmap) ve bağlı listeler
- Biteşlem
  - Bellek, ayırma birimlerine bölünmüştür (birkaç sözcükten KB boyutuna kadar)
  - Her birime karşılık gelen, biteşlem'de bir bit var
  - İstenen uzunlukta boş alan bulmak zor

# Biteşlem ile Bellek Yönetimi

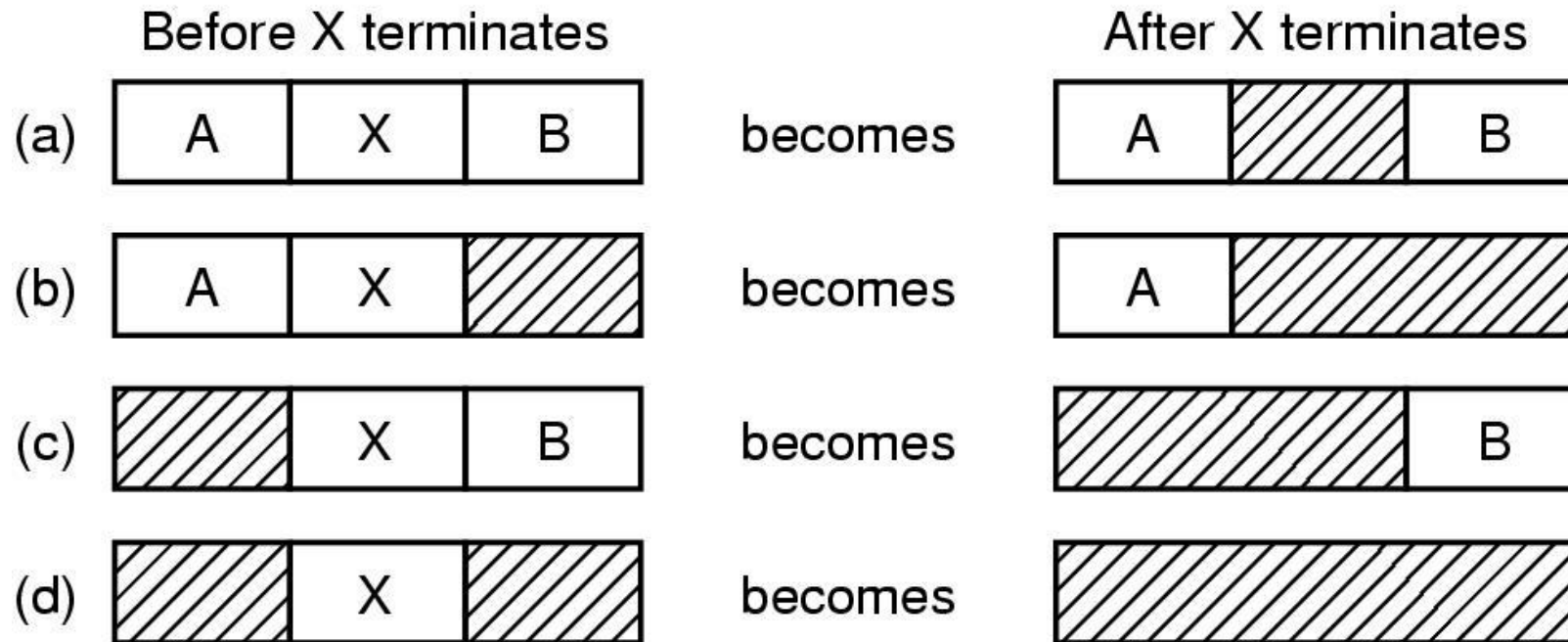
(a) Belleğin bir bölümü, beş işlem ve üç delik var. İm işaretleri, bellek ayırma birimlerini gösterir. Gölge bölgeler (bit eşlemde 0) boştur. (b) ilgili bit eşlem. (c) bağlı liste gösterimi.





# Bağlı Liste ile Bellek Yönetimi

- X Sürecini sonlandırdıktan sonra oluşan bellek düzeni.



# Boş Alan Yönetimi – Bağlı Liste

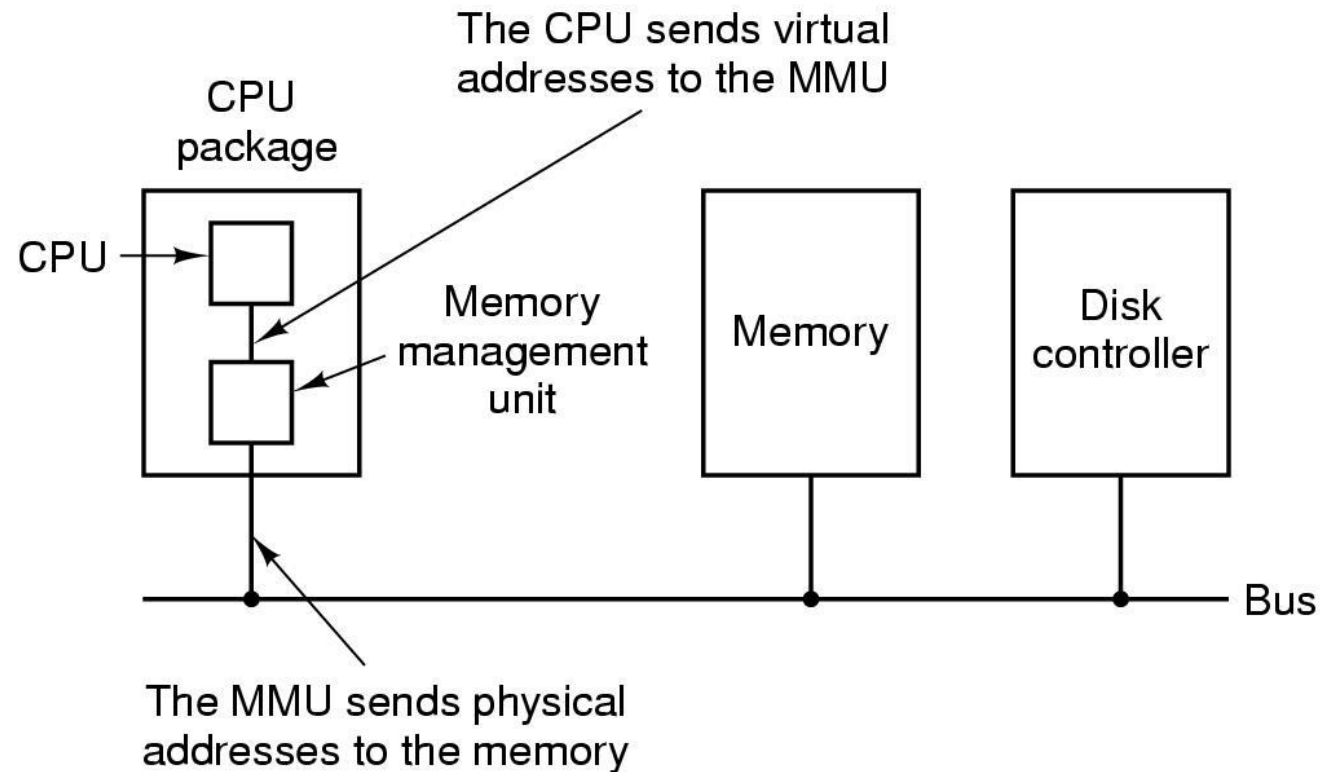
- Çift bağlı liste
- Programlara boş hafıza nasıl tahsis edilir?
  - İlk uyan (first fit)
    - hızlı; başlangıç daha sık kullanılır; büyük bir boş alanı kırmak
  - Sonraki uyan (next fit)
    - Her seferinde en son kullanılan yerden
  - En iyi uyan (best fit)
    - Tüm listeyi arayıp, gerekli boyuta yakın deliği bulan
  - En kötü uyan (worst fit)
    - En büyük deliği bulur
  - Hızlı uyan (quick fit)
    - İşlemler ve delikler için ayrı kuyruklarda tutulur

# Sanal Bellek – Şişkin (bloat) Yazılım Yönetimi

- Programların belleğe sığmayacak kadar büyük olduğu yerler
- Programlar tarafından bölünmek kötü bir fikir
- Sanal bellek
  - Her programın kendi adres alanı vardır
  - Adres alanı, sayfa adı verilen parçalara bölünmüştür.
  - Her sayfa bitişik bir alandır ve fiziksel adresle eşlenir
  - Tüm sayfaların fiziksel bellekte olması gerekmez.
  - İşletim sistemi, sayfa adreslerini ve fiziksel adresleri ihtiyaç anında eşler
  - Gerekli bir sayfa bellekte olmadığında, işletim sistemi halleder
  - Her sayfa değiş tokuşa ihtiyaç duyar

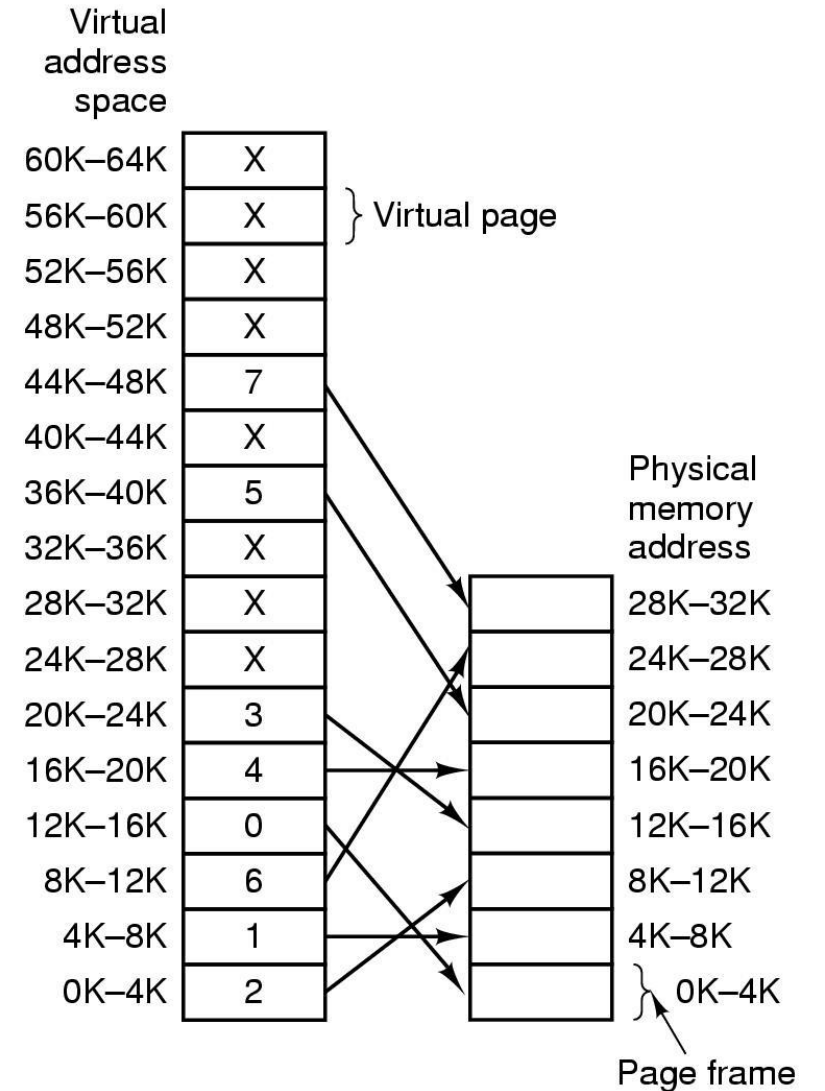
# Sanal Bellek - Sayfalama

- MMU'nun konumu ve işlevi – işlemcinin bir parçası olarak gösterilir



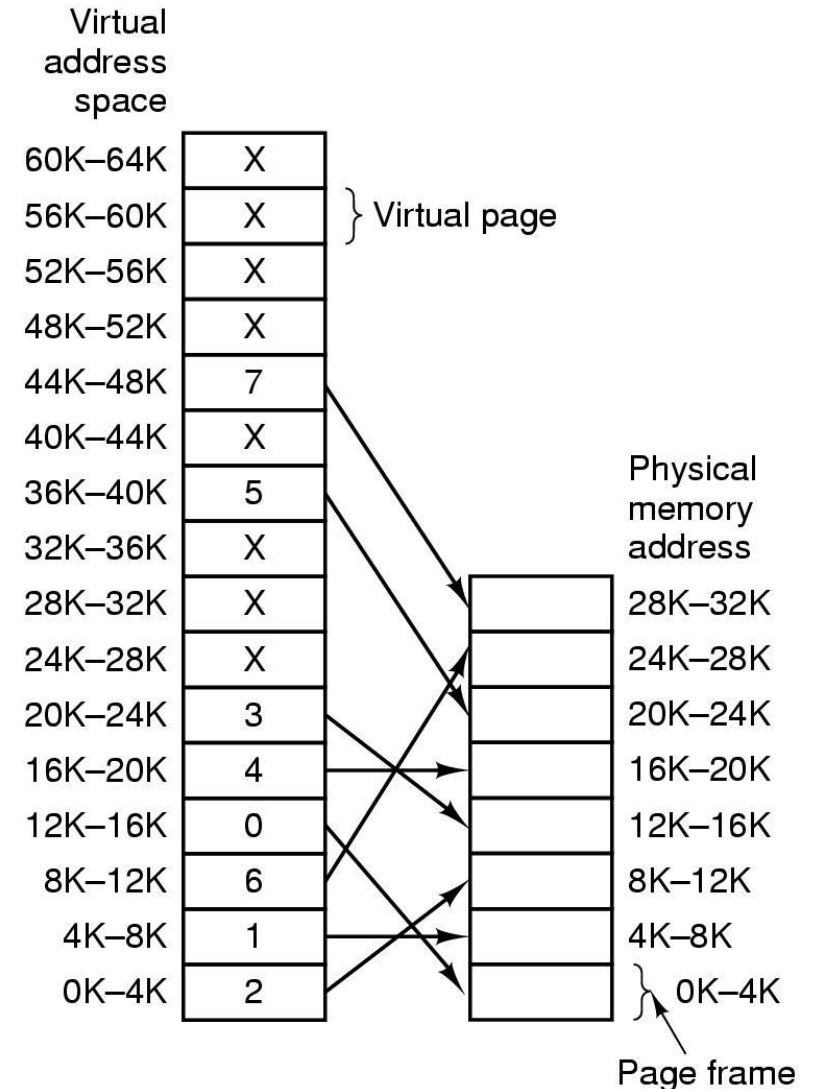
# Sanal Bellek – Sayfa Tablosu

Sayfa tablosu (page table) sanal ve fiziksel bellek adresleri arasındaki ilişkiyi tutar.



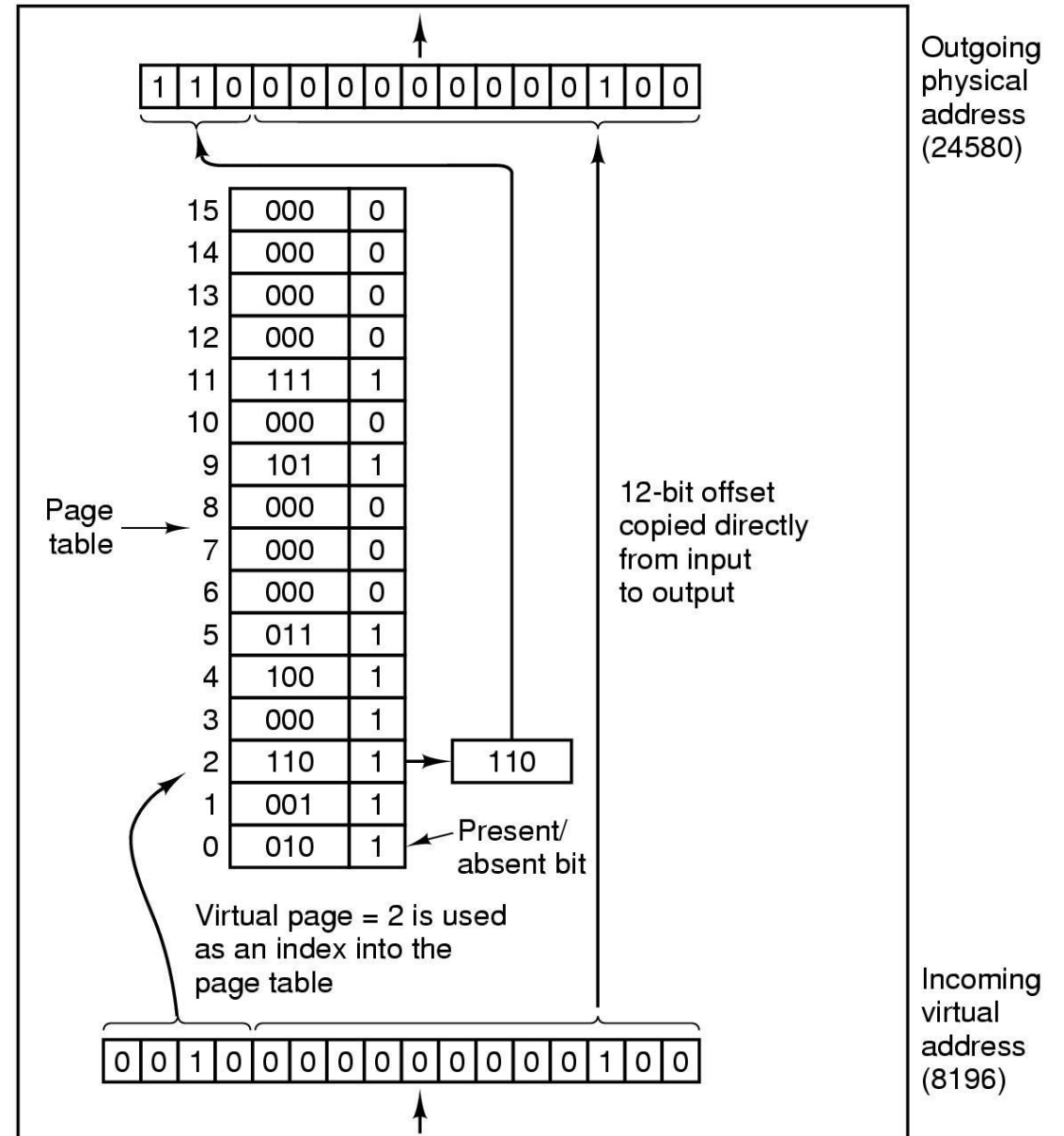
# Bellek Yönetim Birimi

- MMU( memory management unit)
  - CPU: MOV REG, 0
  - MMU: MOV REG, 8192
  - CPU: MOV REG 8192
  - MMU: MOV REG 24567
  - CPU:MOV REG 20500
  - MMU:MOV REG 12308
  - CPU: MOV REG 32780
  - MMU: page fault



# Bellek Yönetim Birimi

- 16 adet 4 KB sayfalı MMU'nun dahili çalışması



# Sanal Adres Eşleme

- Sanal adres, sanal sayfa numarası ve ofset
- 16 bit adres: 4 KB sayfa boyutu; 16 sayfa
- Sanal sayfa numarası: sayfa tablosundaki indis (index)
- Sayfa tablosunun amacı
  - Sanal sayfaları sayfa çerçevelerine eşleme



# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama



# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama



# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama



# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

# Sanal Bellek - Sayfalama

SON