

BIL 362 Mikroİşlemciler

M.Ali Akcayol
Gazi Üniversitesi
Bilgisayar Mühendisliği Bölümü

Konular

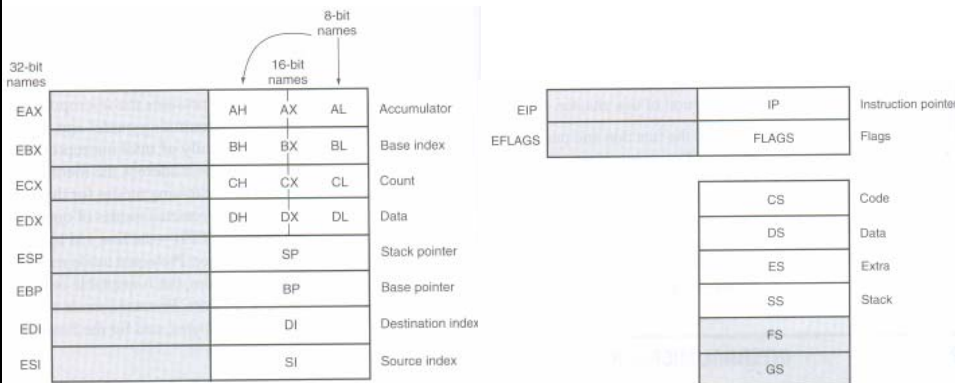
- Internal microprocessor architecture
- Real mode adresleme
- Protected mode adresleme

Internal Microprocessor Architecture

- Bir program yazmadan önce mikroişlemcinin iç yapısının bilinmesi zorunludur.
- **Program visible** model 8086 dan Pentium mikroişlemciye kadar tüm mikroişlemcilerde intruction'lar tarafından değiştirilebilen register'ları ifade eder.
- **Program invisible** model 80286 ve üstü mikroişlemcilerde bulunan ve uygulama programı tarafından doğrudan adreslenemeyen tüm register'ları ifade eder.
- Program-invisible register'lar protected memory sistemini kontrol etmek gibi işler için kullanılır.
- 8086, 8088 ve 80286 işlemcileri **16 bit** iç mimariye sahiptir. 80386 dan Pentium 4 işlemciye kadar iç mimari **32 bit** register'lara sahiptir.

Internal Microprocessor Architecture

- Koyu gri register'lar 80386 - Pentium 4 işlemcilerde vardır.



Internal Microprocessor Architecture

- Program 8, 16 ve 32 bit register'ları içerebilir.
- 8 bit register'lar AH, AL, BH, BL, CH, CL, DH ve DL dir. (ADD AL, AH instruction'ı AL register'ına AH register'ını ekler. Sadece AL değeri değiştirir.)
- 16 bit register'lar AX, BX, CX, DX, SP, BP, DI, SI, IP, FLAGS, CS, DS, ES, SS, FS ve GS dir. (ADD DX, CX instruction'ı DX register'ına CX register'ını ekler. Sadece DX değeri değiştirir.)
- 32 bit register'lar EAX, EBX, ECX, EDX, ESP, EBP, EDI, ESI, EIP ve EFLAGS.
- 32 bit extended register'lar ve 16 bit FS ve GS register'ları sadece 80386 ve üstü işlemcilerde bulunur. (ADD ECX, EBX instruction'ı ECX register'ına EBX register'ını ekler. Sadece ECX değeri değiştirir.)

Internal Microprocessor Architecture

- Bazı registerlar genel veya çok amaçlıdır (general or multipurpose purpose) bazı register'lar ise özel amaçlıdır (special purpose).

Genel amaçlı register'lar

EAX (Accumulator)

- 32 bit olarak kullanılırsa EAX, 16 bit kullanılırsa AX ve 8 bit olarak kullanılırsa AH veya AL olarak yazılır.
- 8 veya 16 bit olarak kullanıldığında sadece ilk 8 veya 16 bit değeri değiştirir diğer kısım değişmez.
- Çarpma ve bölme gibi işlemlerde accumulator olarak kullanılır.
- 80386 ve üstü işlemcilerde EAX hafızada offset adresini tutmak için kullanılabilir.

EBX (Base index)

- EBX, BX, BH ve BL olarak kullanılabilir.
- Tüm mikroişlemciler için hafızada offset adresini de tutabilir.

Internal Microprocessor Architecture

Genel amaçlı register'lar (devam)

ECX (Count)

- 80386 ve üstü işlemcilerde hafızada offset adresi tutar.
- String, döngü, shift ve rotate gibi instruction'lar count değeri kullanır.
- Shift ve rotate instruction'ları CL, string repeat instruction'ları (REP, REPE, REPNE) CX ve loop instruction'ları (LOOP) CX veya ECX kullanır.

EDX (Data)

- Çarpma işleminin sonucunun bir kısmını veya bölme işleminde bölünenin bir kısmını tutar.
- 80386 ve üstü işlemcilerde hafızada data adreslemek için kullanılabilir.

EBP (Base pointer)

- Tüm işlemcilerde hafızada bir alanı adreslemek için kullanılır.
- BP veya EBP olarak kullanılabilir.

Internal Microprocessor Architecture

Genel amaçlı register'lar (devam)

EDI (Destination index)

- Genellikle string instruction'larda hedef string adresleme amacıyla kullanılır.
- DI (16 bit) veya EDI (32 bit) olarak kullanılabilir.

ESI (Source index)

- Genellikle string instruction'larda kaynak string adresleme amacıyla kullanılır.
- SI (16 bit) veya ESI (32 bit) olarak kullanılabilir.

Internal Microprocessor Architecture

Özel amaçlı register'lar

EIP (Instruction pointer)

- Code segment içindeki bir sonraki instruction'ı adresler.
- Mikroişlemci real mode'da çalışırken IP (16 bit), protected mode'da çalışırken EIP (32 bit) kullanılır.
- Instruction pointer **call** ve **jump** instruction'ları ile değiştirilebilir.

ESP (Stack pointer)

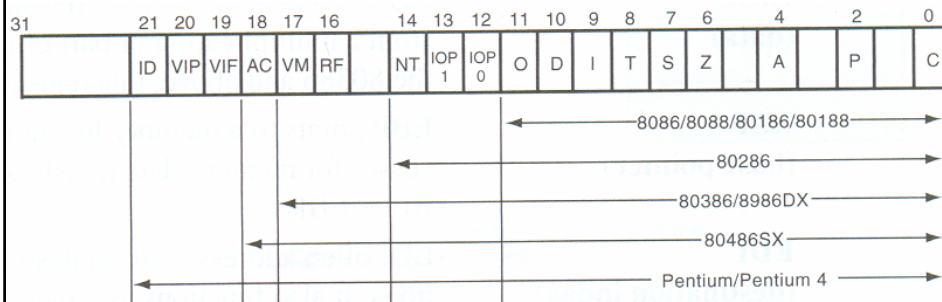
- Hafızada stack alanının adreslenmesi için kullanılır.
- SP (16 bit) veya ESP (32 bit) olarak kullanılabilir.

Internal Microprocessor Architecture

Özel amaçlı register'lar (devam)

EFLAGS

- Mikroişlemcinin durumunu gösterir veya kontrol etmek için kullanılır.
- FLAGS 8086/8088 - Pentium 4 'e kadar upward compatible'dır.



C (Carry)

- Toplama ve çıkarma işleminde taşıma bitini tutar.

Internal Microprocessor Architecture

Özel amaçlı register'lar (devam)

P (Parity)

- Odd parity (tek eşlik) için 0, even parity (çift eşlik) için 1 değerini alır.
- Parity bir bit serisindeki 1 lerin sayısıdır. Veri iletişimde hata denetimi için kullanılır.
- İlk Intel işlemcilerinde hata denetimi mikroışlemcide yapıldı. Günümüzde parity check işlemi ağ kartlarında yapılır.

A (Auxiliary carry)

- BCD toplama ve çıkarma işleminde kullanılır.

Z (Zero)

- Aritmetik ve mantık işlemin sonucunun sıfır olduğunu ($Z = 1$ için) belirtir. Diğer durumda ($Z = 0$) işlem sonucu sıfırdan farklıdır.

S (Sign)

- Aritmetik ve mantık işlemin sonucunun işaretini tutar. ($S = 1$ için sonuç negatif, $S = 0$ için işaret pozitif)

Internal Microprocessor Architecture

Özel amaçlı register'lar (devam)

T (Trap)

- Mikroışlemcinin debug modunda çalışmasını ($T = 1$ için) sağlar.

I (Interrupt)

- Interrupt biti INTR giriş pinini kontrol eder.
- STI (Set I flag) ve CLI (Clear I flag) instruction'larıyla değiştirilebilir.

D (Direction)

- DI ve SI register'larının artma veya azalma modunda olmasını denetler.
- $D = 1$ için register'lar otomatik olarak azalır, $D = 0$ için artar.
- STD (Set direction) ve CLD (Clear direction) instruction'ları değiştirir.

O (Overflow)

- İşaretili sayılarda toplama ve çıkarma sonucunda makinenin kapasitesinden büyük değer oluşursa overflow oluşur.
- $7FH (+127) + 01H (+1) = 80H (-128)$ **8 bit toplama işlemi**
- $01111111 + 00000001 = 10000000$

Internal Microprocessor Architecture

Segment register'lar

CS (Code)

- Code segment mikroişlemcinin kullandığı kodu tutan kısımdır.
- CS, register code segment kısmının başlangıç adresini tutar.
- CS boyutu real mode'da 64 KByte, protected mode'da bir tanımlayıcı (descriptor) ile başlangıç adresi ve boyutu belirlenir.
- 8088 ve 80286 için 64 KByte 80386 ve üstü için mikroişlemci protected mode' da çalışırken 4GByte olabilir.

DS (Data)

- Data segment program tarafından kullanılan dataları tutar.
- DS, Data segment başlangıç adresini tutar.
- 8088 ve 80286 için 64 KByte 80386 ve üstü için mikroişlemci protected mode' da çalışırken 4GByte olabilir.

Internal Microprocessor Architecture

Segment register'lar (devam)

ES (Extra)

- Extra segment bazı string instruction'lar tarafından hedef dataya ulaşmak için kullanılır.

SS (Stack)

- Stack segment hafızada stack olarak kullanılan kısmı tutar.
- BP ile stack segment içinde adresleme yapılabilir.

FS ve GS

- 80386 ve üstü işlemcilerde kullanılırlar.
- Windows tarafından kullanılırlar.

Real Mode Hafıza Adresleme

- 80286 ve üstü işlemciler real mode veya protected mode'da çalışabilirler.
- 8086/8088 işlemciler doğrudan real mode'da çalışırlar.
- Real mode çalışmada hafızada sadece 1 MB alan adreslenebilir. (İşlemci Pentium bile olsa)
- 1 MB hafıza real memory, conventional memory veya DOS memory olarak adlandırılır.
- Tüm bilgisayarlar ilk açıldığında real mode'da çalışır.

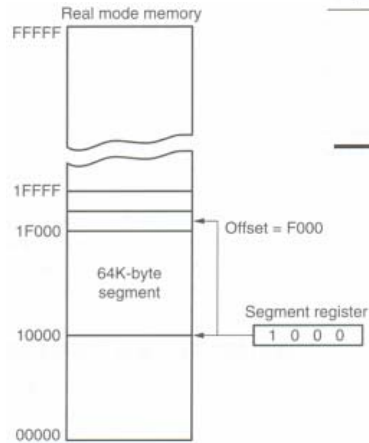
Real Mode Hafıza Adresleme

Segment ve Offset

- Real mode'da segment adres ve offset adres'in birleşimiyle hafızada istenen alana erişilir.
- Segment adres segment register'ların birisindedir ve 64 KB segment kısmının başlangıç adresini tutar.
- Offset adres 64 KB segment içerisindeki bir adresi belirtir.
- Real mode'da segment boyutu her zaman 64 KB dır.
- Real mode'da segment başlangıç adresi için segment register değerinin en sağına 0H eklenerek 20-bit adres elde edilir.
- Segment register = 1200H ise segment başlangıç adresi 12000H olur.
- En sağa hep 0H eklendiği için segment başlangıçları 16 byte aralıklarla başlayabilir. (paragraph)

Real Mode Hafıza Adresleme

Segment ve Offset (devam)



Segment Register	Starting Address	Ending Address
2000H	20000H	2FFFFH
2001H	20010H	3000FH
2100H	21000H	30FFFFH
AB00H	AB000H	BAFFFFH
1234H	12340H	2233FH

- Segment bitiş adresi başlangıç adresine FFFFH eklenerek bulunur.
- Segment register değeri = 3000H ise segment başlangıç adresi 30000H ve bitiş adresi 30000H + FFFFH = 3FFFFH olur.

Real Mode Hafıza Adresleme

Segment ve Offset (devam)

- İstenen hafıza alanına segment başlangıç adresine offset adresi eklenerek bulunur.
- Segment başlangıç adresi = 1000H ve offset adres = 2000H ise mikroşlemci 12000H hafıza alanını adresler.
- Segment ve offset adresi 1000:2000 şeklinde yazılır. (Segment başlangıcı = 1000H ve offset adresi = 2000H)

Default Segment ve Offset Register'ları

- Mikroşlemci bir segment ve offset adresini belirli kuralları uygulayarak hesaplar.
- CS her zaman IP ile birlikte kullanılır. CS code segment başlangıç adresini ve IP ise segment içindeki adresi (bir sonra çalıştırılacak instruction adresi) tutar.
- Birleşim CS:IP veya CS:EIP şeklinde gösterilir.
- CS = 1400H, IP/EIP = 1200H, CS:IP = 14000H+1200H = 15200H

Real Mode Hafıza Adresleme

Default Segment ve Offset Register'ları (devam)

16-bit

Segment	Offset	Special Purpose
CS	IP	Instruction address
SS	SP or BP	Stack address
DS	BX, DI, SI, an 8- or 16-bit number	Data address
ES	DI for string instructions	String destination address

- Stack içindeki data stack segment ve stack pointer veya base pointer register'ları ile gösterilir.
- Birleşim **SS:SP(SS:ESP)** veya **SS:BP(SS:EBP)** şeklindedir.
- $SS = 2000H$, $BP = 3000H$, $SS:BP = 20000H + 3000H = 23000H$

Real Mode Hafıza Adresleme

Default Segment ve Offset Register'ları (devam)

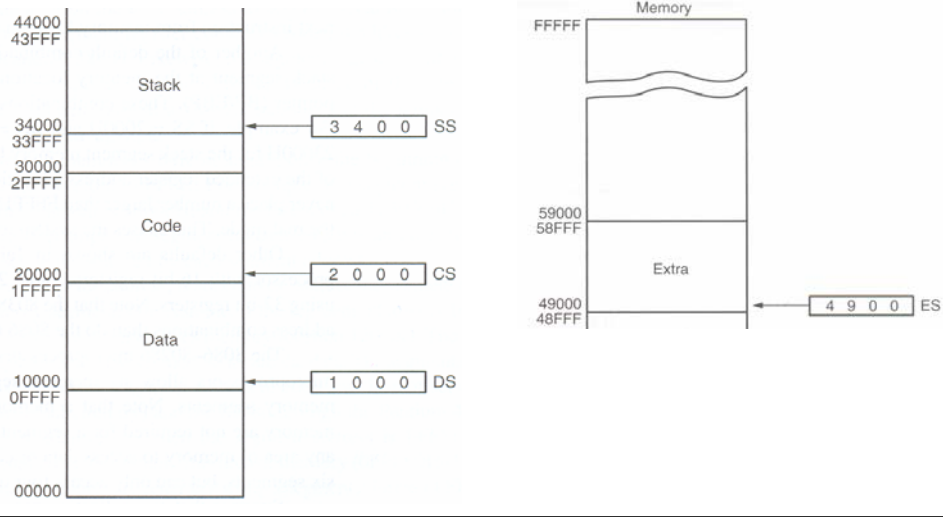
32-bit

Segment	Offset	Special Purpose
CS	EIP	Instruction address
SS	ESP or EBP	Stack address
DS	EAX, EBX, ECX, EDX, ESI, EDI, an 8- or 32-bit number	Data address
ES	EDI for string instructions	String destination address
FS	No default	General address
GS	No default	General address

- 80386 ve üstü işlemciler daha çok segment ve offset adres seçim opsiyonuna sahiptir.

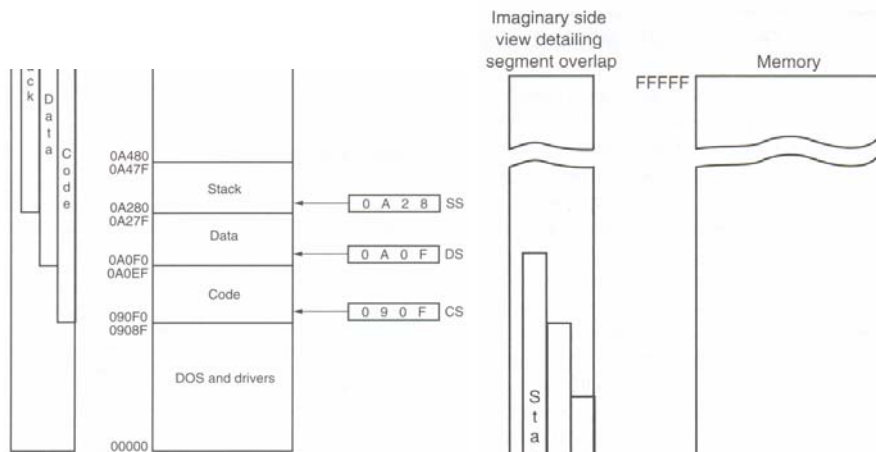
Real Mode Hafıza Adresleme

Segmentlerin örnek yerleşimi



Real Mode Hafıza Adresleme

Bir uygulama programının hafızaya yerleşimi

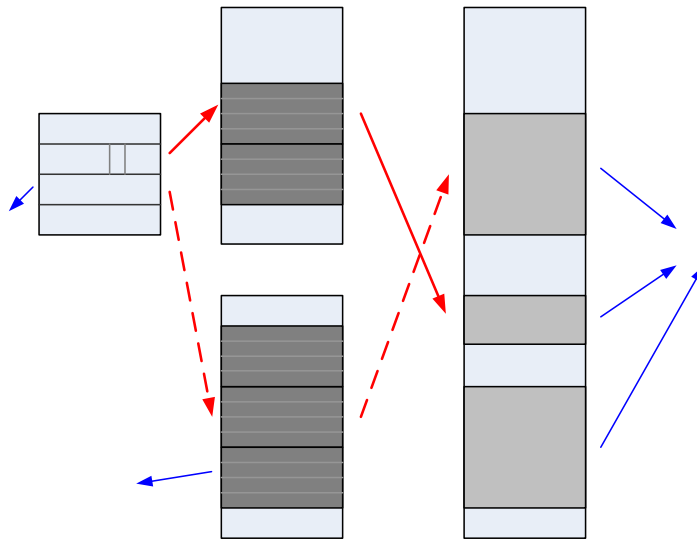


- Code, data ve stack segment 64KB alanı kullanmıyor.
- Segmentlerde overlap (üstüste çıkışma) olabilir.

Protected Mode Hafıza Adresleme

- 80286 ve üstü işlemcilerde 1 MB hafıza alanının üstündeki alanlara ulaşmaya izin verir. (Windows protected mode'da çalışır)
- Offset adres protected mode'da segment içerisinde istenen adresi belirler.
- Segment register bir **selector** saklar ve **descriptor** tablosunda bir **descriptor** seçer.
- Descriptor segment başlangıç adresini, boyutunu ve erişim haklarını belirler.
- 80386 ve üstünde offset adres 32 bittir ve segment boyutu 4 GB olabilir.

Protected Mode Hafıza Adresleme



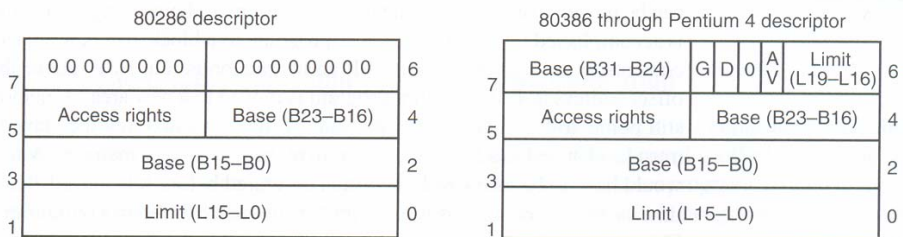
Protected Mode Hafıza Adresleme

Selectors ve Descriptors

- Selector'lar segment register'larda saklanır ve iki tabloda bulunan 8192 descriptor'dan birisini seçer.
- Global descriptor tablosu tüm programların kullanacağı segmentlere ilişkin bilgiyi tutar.
- Local descriptor tablosu bir uygulamanın kullanacağı segmentlere ilişkin bilgiyi tutar.
- İki descriptor tablosunda toplam 16384 segment oluşturulabilir.
- Bir segment boyutu 4 GB olabilir. Toplam adreslenebilir hafıza alanı = 4 GB x 16384 = 64 TB.
- Her descriptor 8 byte'tır. Her descriptor tablosunun kapladığı alan = 8192 x 8 = 64 KB tır.
- 80386-Pentium 4 descriptor'ları farklı yapıdadır.

Protected Mode Hafıza Adresleme

Selectors ve Descriptors (devam)



- **Base address** kısmı segmentin başlangıç adresini tutar.
- **Limit** kısmı segment'in son offset adresini tutar.

Protected Mode Hafıza Adresleme

Selectors ve Descriptors (devam)

- **G (Granularity bit)** segment limitini 4K ile çarparak büyütür.

Base = Start = 10000000H

G = 0

End = Base + Limit = 10000000H + 001FFH = 100001FFH

Base = Start = 10000000H

G = 1

End = Base + Limit = 10000000H + 001FFFFH = 101FFFFFH

-
- **AV (Available)** 80386 ve üstünde segmentin kullanılabilir olma durumunu belirler.
 - AV = 1 available, AV = 0 not available

Protected Mode Hafıza Adresleme

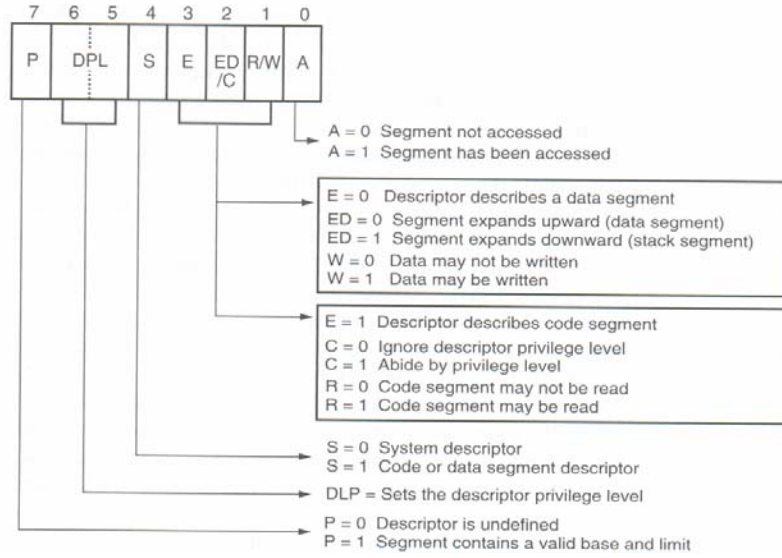
Selectors ve Descriptors (devam)

- **D bit** 80386 ve üstü için çalışma modunu belirler. (real / protected mode)
- D = 0, 16 bit instructions, 16 bit offset adres, 16 bit register.
- D = 1, 32 bit instructions, 32 bit offset adres, 32 bit register.
- **Access rights** byte protected mode segment'ine erişim denetimi yapar.

Protected Mode Hafıza Adresleme

Selectors ve Descriptors (devam)

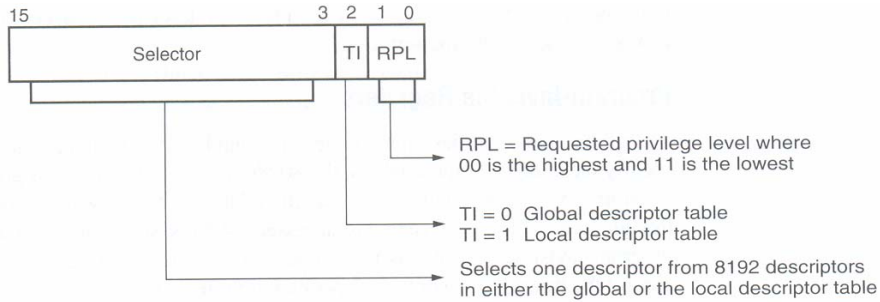
Segment erişim hakları



Protected Mode Hafıza Adresleme

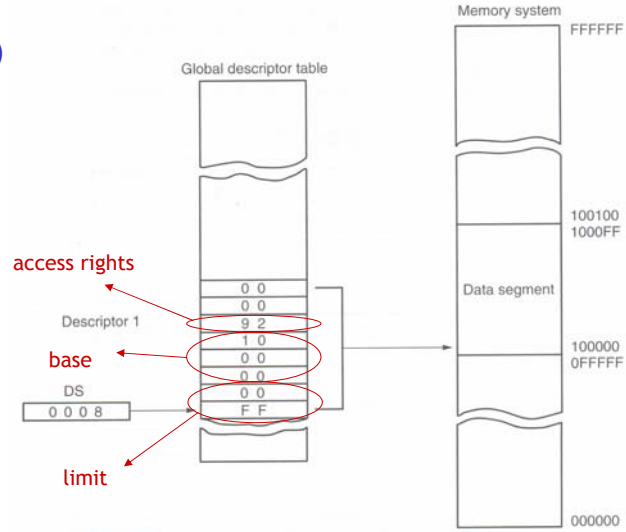
Selectors ve Descriptors (devam)

- Descriptor seçimi segment register ile yapılır.
- Segment register'ın 13 biti 8192 descriptor'dan birini seçer.
- TI = 0, global TI = 1 local descriptor tablosundan seçim yapar.
- RPL (Requested Privilege Level), 00 highest, 11 lowest.
- Eğer RPL descriptor içindeki DPL'den öncelikli ise erişim hakkı verilir. (RPL = 10, DPL = 11)
- Windows kernel işlemleri için 00, sürücü programları için 11 kullanır.



Protected Mode Hafıza Adresleme

Selectors Descriptors (devam)



DS = 8 = 0000000000001000, TI = 0(global), RPL = 00(highest), Selector = 0000000000001 (Descriptor 1)

Protected Mode Hafıza Adresleme

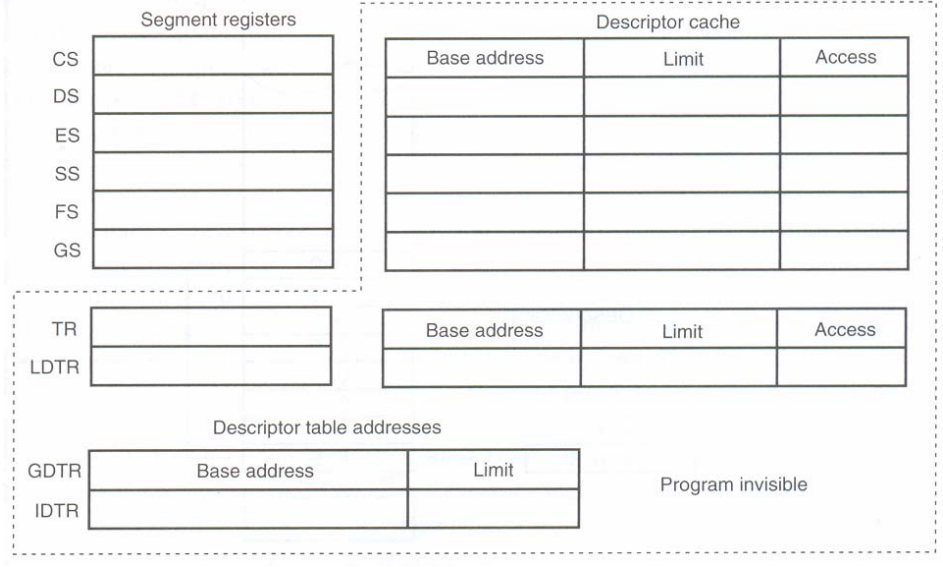
Program-Invisible Registers

- Global ve local descriptor tabloları 80286-Pentium 4 işlemcilerde vardır.
- Bu tablolara erişim ve adresleme için program-invisible register'lar kullanılır.
- Program-invisible register'lar sadece protected mode'da kullanılırlar.
- GDTR (Global Descriptor Table Register) ve IDTR (Interrupt Descriptor Table Register) tabloların base adreslerini ve limitlerini tutar.
- LDTR (Local Descriptor Table Register) local descriptor tablosuna erişmek için kullanılır.
- TR (Task Register) görevler (bir program veya prosedür) arasında yaklaşık 17µs de geçiş yapmayı sağlar.
- Multitasking TR tarafından sıralı geçişlerle gerçekleştirilir.



Protected Mode Hafıza Adresleme

Program-Invisible Registers (devam)



Ödev

- Memory swapping, partitioning, paging hakkında rapor hazırlayınız.