

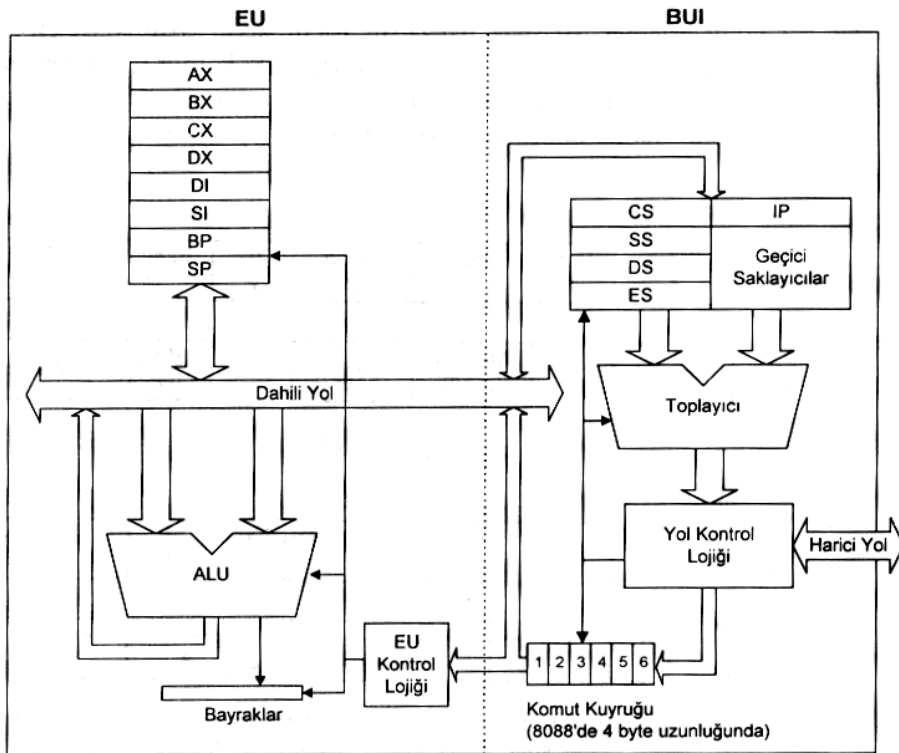
BÖLÜM 3

INTEL AİLESİNİN 16 BİTLİK MİKROİŞLEMCİLERİ

3.1 8086/8088 MİKROİŞLEMCİSİ

3.1.1 İÇ MİMARİSİ

Şekil 3.1'de x86 ailesinin 16-bit çekirdek mimarisinin basitleştirilmiş bir gösterimi verilmiştir. Mikroişlemci temel iki ayrı çalışma birimine sahiptir. Bu birimler: **Yürütme birimi (Execution Unit – EU)** ve **yol arabirimi (Bus Interface Unit – BIU)**. EU komutları yorumlamakta (decode) ve yürütmektedir. BIU ise yol işlerini (işlem kodu okuma, operand okuma, veri saklama ve I/O cihazlarıyla haberleşme gibi) gerçekleştirmektedir.



Şekil 3.1 8086 / 8088 mikroişlemcisinin iç mimarisi

3.1.2 YÜRÜTME BİRİMİ

EU komutları çözme ve komutları yürütme için bir kontrol birimine; aritmetik ve lojik işlemler için bir ALU'ya; genel amaçlı saklayıcılara (AX, BX, CX ve DX); işaretçi (BP ve SP) ve indis (SI, DI) saklayıcıları ile bayraklar (flags) saklayıcısına sahiptir. EU içindeki kontrol birimi, makine dilindeki komutları yorumlamakta ve komutları

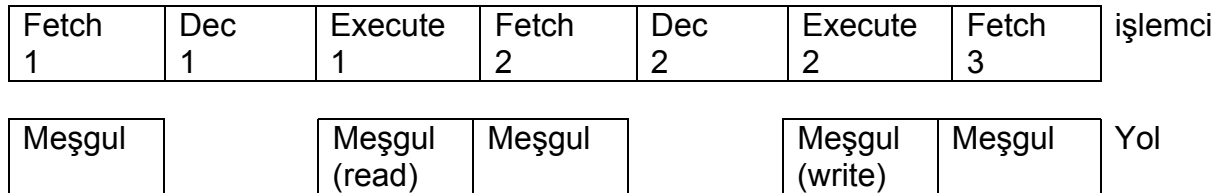
yürütmek için gerekli işlemleri kontrol etmektedir. EU komut byte'larını BIU tarafından **komut kuyruğuna (instruction queue)** yerleştirilme sırasıyla, komut kuyruğundan almaktadır. Eğer EU hafızadan bir operand'a ihtiyaç duyarsa veya bir sonucu hafızada saklamak isterse istenilen işlemi BIU'e yönlendirir. EU, bir işlem kodu okuma veya saklama işlemi için, BIU tarafından fiziksel adresi hesaplamada gerekenleri sağlamaktadır. ALU, operand'larını, dahili yol üzerinden, genel amaçlı saklayıcılardan, komut byte'larından veya BIU'dan (örneğin, hafızadan okunan operand'lar) almaktadır. ALU 8-bit veya 16-bit işlem yapma kapasitesine sahiptir.

3.1.3 YOL ARABİRİMİ

BIU, bütün dış yol işlemlerini kontrol eden bir yol kontrol birimine; EU için komut byte'larını tutan komut kuyruğuna; fiziksel hafıza adresleri üretme için bir toplayıcıya; 4 segment saklayıcısına (CS, SS, DS ve ES); komut işaretçisine (IP) ve verileri geçici olarak saklamada kullanılan bazı iç saklayıcılara sahiptir. BIU, hafıza ve I/O işlemleri dahil, bütün dış yol işlemlerini kontrol etmekten sorumludur. Tümdevre harici adres yolu, belli bir hafıza hücresini veya I/O port'unu seçmede kullanılır.

BIU, komut byte'larını okur ve onları EU için komut kuyruğuna yerleştirir. BIU en fazla 4 (8088 için) veya 6 (8086 için) byte komut kodunu okuyabilir. EU, bir veri okuma veya yazma işlemine ihtiyaç duymadığı veya program akışında bir dallanma olmadığı sürece, BIU komut byte'larını önden okuma için serbesttir. Bu şekilde komut önden okuma, BIU ve EU birimlerinin paralel çalışmasına imkan tanır. Bu sayede işlemcinin veri işlem hızı artar. Bu şekilde çalışan bilgisayar mimarisine **işhatlı mimari (pipelined architecture)** denir. Şekil 3.2'de klasik bir 8-bit işlemcideki, örneğin 8085A'daki ile işhatlı mimariye sahip 8086/8088 mikroişlemcilerindeki yol zamanlaması görülmektedir.

Şekil 3.2 (a)'da işlemci, sırasıyla hafızadan işlem kodu okuma (fetch), kod çözme (decode) ve yürütme (execute) işlemlerini sırayla yapıp iki komut için bitirmekte ve üçüncü komutun işlem kodunu okumaktadır. Birinci komutun yürütmesinde, hafızadan bir veri okuma işlemi ve ikinci komutun yürütmesinde ise, hafızaya bir yazma işlemi yapılmaktadır.



a)



b)

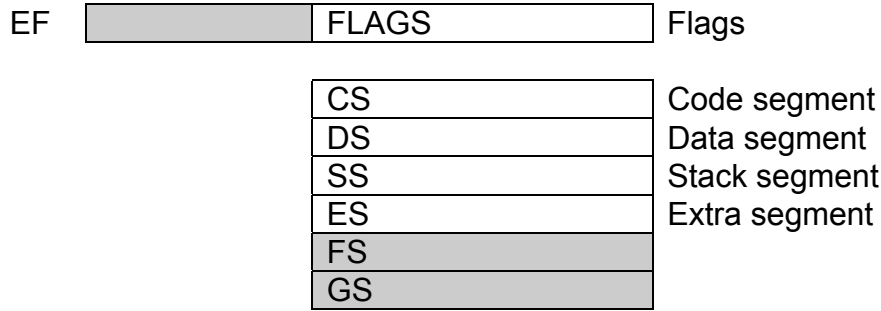
Şekil 3.2 Yol zamanlaması a) 8085A mikroişlemcisi, b) 8086 /8088 mikroişlemcisi

Şekil 3.2 (b)' deki x86 işlemcisinde, birinci komutun yürütülmesi ile ikinci komutun okuma işlemi (fetch 2) üst üste gelmekte, yani aynı anda paralel yapılmaktadır. İkinci komutun yürütülmesi de üçüncü komutun okunması (fetch 3) ile paralel yapılmaktadır. Klasik yapıdaki mikroişlemcide, yukarıda şekil 3.2 (a)' da verilen zamanlama diyagramı sonunda, üçüncü komutun işlem kodu okunurken, yaklaşık aynı zamanda, işhatlı yapıdaki işlemci, üçüncü komutun yürütmesini bitirmiş ve dördüncü komutun işlem kodunda okumuş durumdadır. Klasik mimaride mikroişlemci kod çözümü yaparken, (bazen de yürütme sırasında, operand gerektirmeyen komutlarda)yol, boşta kalmakta, yolda bir iş yapılmamaktadır. Buna karşın, işhatlı mimaride yol (genelde) boş kalmamaktadır. Verilen bu zamanlama diyagramından görüldüğü gibi, işhatlı mimari yapı mikroişlemcinin veri işlem hızını artırmaktadır.

3.1.4 SAKLAYICILAR

Şekil 3.3 intel x86 ailesinde yer alan mikroişlemcilerin temel saklayıcılarını göstermektedir. Gri olarak gölgelendirilmiş alanlar, 80386, 80486 ve pentium gibi 32-bit işlemcilerde bulunur. X86 saklayıcıları 3 grupta toplanabilir. Genel amaçlı saklayıcılar, **işaretçi (pointer)** ve **indis (index)** saklayıcıları ve **segment** saklayıcıları. Bu gruplara ek olarak, CPU'ya ait çeşitli durumları gösteren, örneğin bir aritmetik veya lojik işlem sonucu gibi, bir **bayrak (flag)** saklayıcısı da bulunmaktadır.

	32	16 15	8 7	0	
EAX		AH	AL		AX Accumulator
EBX		BH	BL		BX Base index
ECX		CH	CL		CX Count
EDX		DH	DL		DX Data
ESP		SP			Stack pointer
EBP		BP			Base pointer
EDI		SI			Source index
ESI		DI			Destination ind.
EIP		IP			Instruction poin.



Şekil 3.3 8086 / 8088' den Pentiuma kadar olan temel x86 saklayıcıları (gri alanlar 80386 ve daha sonraki işlemcilerde vardır).

3.1.5 GENEL -AMAÇLI SAKLAYICILAR

Bu grupta yer alan saklayıcılar, programcı tarafından çok değişik amaçlarla kullanılabilir. Genel amaçlı saklayıcılardan her biri, 32-bit (EAX, EBX, ECX ve EDX), 16-bit (AX, BX, CX ve DX) veya 8-bit (AH, AL, BH, BL, CH, CL) olarak kullanılabilir. Daha sonraki bölümlerde açıklanacak bazı komutlar, bu saklayıcıları özel işlemlerde kullandıkları için, her birine bu işlemleri belirten bir isim verilmiştir. (accumulator, base, count ve data) Saklayıcılar, assembly dilde, iki veya üç harfli gösterimlerle belirtilir. Örneğin, accumulator (ACC) EAX, AX, AH veya AL ile ifade edilir. Genel-amaçlı saklayıcıların temel fonksiyonları aşağıda sırasıyla sunulacaktır. Daha sonraki bölümlerde ise kullanımları örneklerle anlatılacaktır.

- **AX (Accumulator-Toplam)** : Bir aritmetik veya lojik işlemden sonra sonucu içerdiği (tuttuğu) için bu şekilde adlandırılmıştır. EAX, AH veya AL şeklinde adreslenir.
- **BX (Base- Taban)** : Hafızada yer alan bir verinin taban (ofset) adresini veya XLAT (translate) komutuyla erişilen bir tablo verisinin taban adresini içermeye sık olarak kullanılır. EBX, BH veya BL şeklinde adreslenir.
- **CX (Count –sayma)** : Bit kaydırma (shift) veya döndürme (rotate) gibi işlemlerde, bit -sayısını (count) tutmada string veya LOOP komutundaki işlem sayısını belirtmede döngü sayacı olarak kullanılır. ECX, CH veya CL şeklinde adreslenir.
- **DX (Data –veri)** : Özellikle, 16- veya 32-bit çarpma işlemlerinden sonra, sonucun yüksek değerli kısmını, bir bölme işleminden önce bölünen sayının yüksek değerli kısmını ve değişken I/O komutunda I/O port numarasını tutma işlemlerinde kullanılır. EDX, DH veya DL şeklinde adreslenir.

3.1.6 İŞARETÇİ VE İNDİS SAKLAYICILARI

İşaretçi ve indis saklayıcıları, genel amaçlı olarak da kullanılabilmelerine rağmen, genellikle, hafızada yer alan operand'lara erişimde indis veya işaretçi olarak kullanılır. Bu saklayıcılar 8086/8088, 80186 ve 80286 işlemcilerinde 16-bit, 80386, 80486'da 32-bit'dir.

- **SP (Stack pointer-yığın işaretçisi)** : En son giren ilk çıkar (Last-In First-Out-LİFO) şeklinde çalışan, yığın (stack) hafızada yer alan bir veriyi adreslemede kullanılır. Bu saklayıcı, PUSH ve POP komutları yürütülürken; bir alt program CALL komutu ile çağrıldığında veya programdan bir RET komutu ile döndürüldüğünde ; kesme işlemlerinde otomatik olarak CPU tarafından verileri yığın hafızaya yazmada ve kesme programında geri dönerken tekrar geri okumada kullanılmaktadır. ESP 32-bit işaretçi olarak 32-bit işlemcilerde kullanılır.
- **BP (Base pointer- taban işaretçisi)** :Yığın hafızada yer alan bir veri dizisini (data array) adreslemede kullanılır. EBP, 32-bit işaretçi olarak 32-bit işlemcilerde kullanılır.
- **SI (Source Index-kaynak indisi)** : string komutlarında kaynak veriyi dolaylı adreslemede kullanılır. ESI, 32-Bit işaretçi olarak 32-bit işlemcilerde kullanılır.
- **DI (Destination Index- Hedef indisi)** : String komutlarında hedef veriyi dolaylı adreslemede kullanılır. EDI, 32-bit işaretçi olarak 32-bit işlemcilerde kullanılır.
- **IP (Instruction Pointer-Komut İşaretçisi)** : Her zaman mikroişlemci tarafından yürütülecek bir sonraki komutu adreslemede kullanılır. IP içeriğinin CS x 10H ile toplanmasıyla, bir sonraki komutun gerçekte olan fiziksel hafıza adresi bulunur. EIP, 32-bit işaretçi olarak 32-bit işlemcilerde kullanılır.

3.1.7 SEGMENT SAKLAYICILARI

Mikroişlemcideki diğer saklayıcılarla beraber hafıza adresleri üretmede kullanılır. Ailenin 8086-80486 ürünlerinde 4 veya 6 tane segment saklayıcısı bulunmaktadır. 80286 ve 32-bit işlemcilerine ait özel çalışma modları olan, **gerçek mod (real mode)** ve **korumalı modda (protected mode)** bu saklayıcıların fonksiyonları değişmektedir. Kitapta, x86 ailesinin 16-bit çekirdeğinin (8086/8088) saklayıcı kümesi görüleceği için, bu saklayıcıların 32-bit fonksiyonlarına fazla değinilmeyecektir. Aşağıda kısaca bu saklayıcıların görevleri belirtilmektedir.

- **CS (Code Segment)** : Kod segment, hafızanın, programları ve alt programları tutan bir bölümdür.CS saklayıcısı, program kodunun başlangıç taban adresini belirler. CS, 8086-80286 mikroişlemcilerinde 64 Kbyte, 80386 ve 80486'da 4 GB ile sınırlıdır.
- **DS (Data Segment)** : Veri segment'i, bir program tarafından kullanılan verilerin bulunduğu hafıza alanıdır.DS saklayıcısı, bu veri alanının başlangıç taban adresini belirler.

- **ES (Extra Segment)** : Bazı string komutlarında kullanılan ek veri alanıdır.
- **SS (Stack Segment)** : Yığın için kullanılan hafıza alanını belirler. Yığın segment'ine yazılacak veya okunacak verinin adresi, SP saklayıcısı tarafından belirlenir. BP saklayıcısı da SS'de bulunan veriyi adreslemede kullanılır.
- **FS ve GS** : Bu ek segment saklayıcılar, 80386 ve 80486'da bulunup programlarda ek hafıza segment'leri olarak kullanılabilir.

3.1.8 BAYRAKLAR (FLAGS) SAKLAYICISI

Bayraklar (flags), mikroişlemcinin çalışmasını belirlediği gibi, çalışma sırasındaki durumunu da yansıtır. Şekil 3.4, 8086/8088 mikroişlemcilerinin bayraklar (flags) sayıcısını göstermektedir. 8086/8088' den pentium'a kadar olan mikroişlemcilerde, bu bayraklar uyumludur. Buna karşın, 8086-80286 mikroişlemcilerinde bayrak sayıcısı 16-bit iken 80386 pentium işlemcilerinde EFLAGS (Extended flags) saklayıcısı 32-bit içermektedir.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				O	D	I	T	S	Z		A		P		C

Şekil 3.4. 8086/8088 Bayraklar saklayıcısı

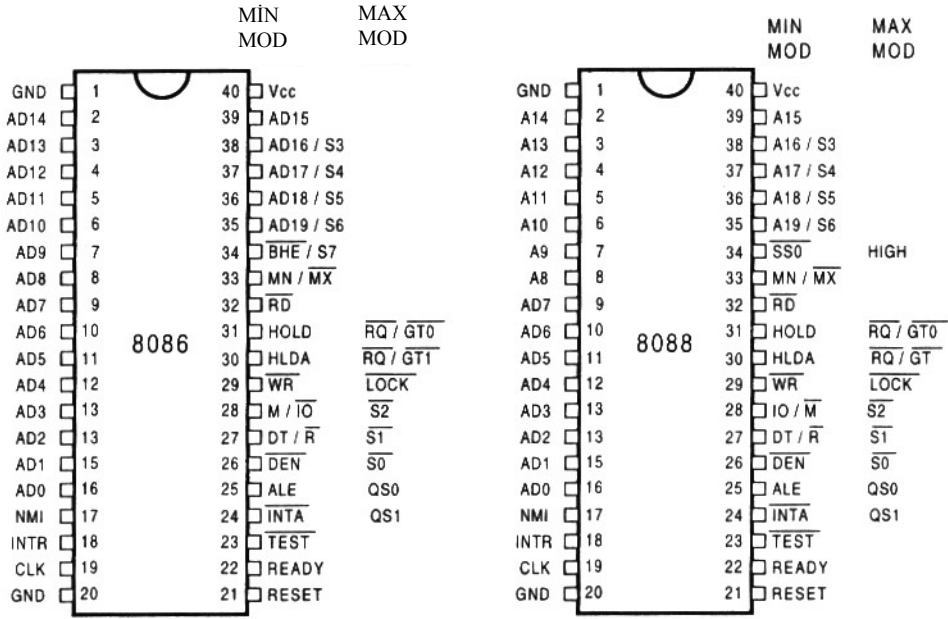
Bayrak bit'leri, birçok aritmetik ve lojik komut yürütüldükten sonra değişmektedir. Bayrakların bazıları, mikroişlemcinin çalışmasını kontrol etmek amacıyla da kullanılır. Aşağıda bayraklar ve fonksiyonları kısaca anlatılmaktadır. İleriki bölümlerde 8086/8088 komutları anlatıldıkça, bayrak bit'leri hakkında ek detaylar verilecektir.

- **C (Carry)** : Bir aritmetik işlemde, toplamadan sonraki **eldeyi (carry)** veya çıkarmadan sonraki **ödüncü (borrow)** belirtir. Programlarda hata durumu, özel işlem durumları ve sonuçları ile ilgili Boolean bayrak olarak da kullanılır.
- **P (Parity)** : Tek eşlik (odd parity) işlemi, lojik 0; çift eşlik (even parity) işlemi, lojik 1 ile gösterilir. Eşlik, tek veya çift olarak belirtilen, bir byte veya word'deki 1'lerin sayısıdır. Örneğin, ikili olarak belirtilen bir sayı (00110100) 3 tane 1'e sahipse, o sayı Tek Eşlik'tir. Eğer bir sayı 1'lere sahip değilse (bütün bit'ler "0"), o sayı Çift Eşlik olarak kabul edilir.
- **A (Auxiliary carry)** : Yapılan bir işlem sonucunda, bit pozisyonları 3 ve 4 arasında olan (en sağdaki bit sıfır pozisyonundadır), toplamadan sonraki eldeyi ve çıkarmadan sonraki ödüncü belirtir. Bu özel bayrak bit'i BCD bir toplama veya çıkarma işleminden sonra, AL'nin değerini **ayarlamak (adjust)** için, DAA ve DAS komutları tarafından test edilir. Diğer durumlarda, A bayrak bit'i mikroişlemci tarafından kullanılmaz.

-
- **Z (Zero)** : Bir aritmetik ve lojik işlem sonucunun sıfır olduğunu belirtir. Eğer $Z=1$ ise, sonuç sıfırdır ve eğer $Z=0$ ise sonuç sıfır değildir.
 - **S (Sign)** : Bir toplama veya çıkarma işleminden sonra, sonucun aritmetik işaretini belirtir. Eğer $S=1$ ise, işaret (Sign) 1'lenir (set) veya negatiftir. Eğer $S=0$ ise, işaret temizlenir (cleared) veya pozitifdir. Bayrakları etkileyen bir komuttan sonra, en değerli bit pozisyonu S bit'ine yerleştirilir.
 - **T (Trap)** : Eğer Trap bayrağı 1'lenmiş ise, tümdevre hata takip (debugging) işlemi devreye girer.
 - **I (Interrupt)** : Mikroişlemci tümdevresinin kesme isteği giriş bacağı INTR, harici kesme isteği (Interrupt Request) işlemi kontrol eder. Eğer $I=1$ ise, INTR bacağı aktif yapılır (enabled) ve eğer $I=0$ ise, INTR bacağı pasif (disabled) olur. Aktif olma durumunda, mikroişlemci tümdevre INTR ucu üzerinden gelen kesmelere cevap verir. Aksi durumda, gelen kesme istekleri mikroişlemci tarafından ihmal edilir, yani cevap verilmez. 'I' bayrak bit'inin durumu, STI (Set I Flag) ve CLI (Clear I Flag) komutlarıyla kontrol edilir.
 - **D (Direction)** : String komutları yürütülürken, DI ve/veya SI saklayıcılarının artırılması veya azaltılması işlemlerinin seçimini kontrol eder. Eğer $D=1$ ise, saklayıcılar otomatik olarak azaltılır ve eğer $D=0$ ise, saklayıcılar otomatik olarak artırılır. D bayrağı STD (Set Direction) komutuyla 1'lenir ve CLD (Clear Direction) komutuyla 0'lanır.
 - **O (Overflow)** : Taşma (overflow), işaretli sayıların toplandığında veya çıkartıldıklarında oluşan bir durumdur. Taşma, işlem sonucunun hedef saklayıcıya sığmadığını gösterir. Örneğin, 8-bit saklayıcılarda işaretli sayı aritmetiğinde, 7FH (+127) sayısı 01H ile toplandığında sonuç 80H (-128) olur. Bu sonuç işaretli toplama için O- bayrağı ile belirtilen bir taşma durumudur. İşaretsiz işlemler için, O-bayrağı ihmal edilir.

3.1.9 8086 / 8088 DIŞ MİMARİSİ

Şekil 3.5 8088 ve 8086 mikroişlemcilerinin tümdevre uçlarını göstermektedir. Her iki tümdevrede 40-uç olup aralarında çok az farklılıklar bulunur. En büyük fark veri yolu uzunluklarındadır. 8088 8-bit veri yoluna sahiptir. Buna karşın, 8086'da 16-bit veri yolu bulunur.



Şekil 3.5 8086 / 8088 mikroişlemcileri

8086/8088 işlemcileri için MN/MX ucuyla belirlenen iki çalışma modu vardır. Minimum mod çalışması, bu uca +5V uygulayarak; maximum mod, bu ucu topraklayarak elde edilir. Her iki mod farklı kontrol yapılarına neden olur. Bu farklı modlarda farklı sinyaller üretilir.

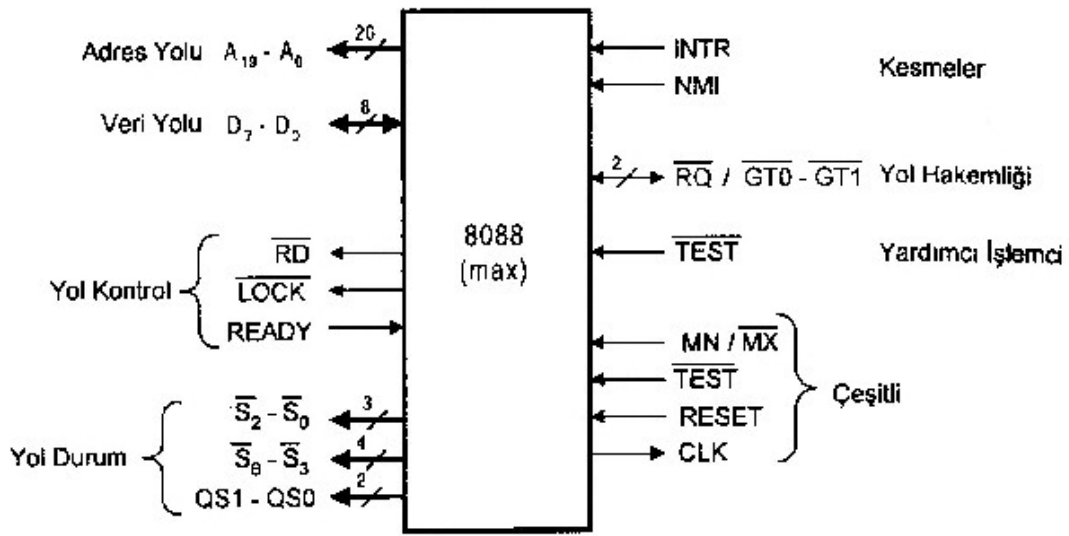
Yukarıda belirtildiği gibi, iki işlemci arasındaki dış mimari olarak en büyük fark, veri yolları uzunlukları arasında bulunur. Bununla beraber, kontrol sinyallerinden birinde de ufak bir fark vardır. 8088'deki M/IO sinyaline karşın, 8086'da IO/M bulunmaktadır. Diğer bir donanım farkı, her iki tümdevrenin 34 numaralı ucunda bulunur. 8088'de bu uç SS0 ve 8086'da ise BHE/S₇'dir.

Şekil 3.6 ve 3.7'de 8088 işlemcisinin tümdevre sinyallerinin her iki çalışma moduna göre gruplanmış yapıları görülmektedir. Benzeri şekiller 8086 işlemcisi için de çizilebilir.

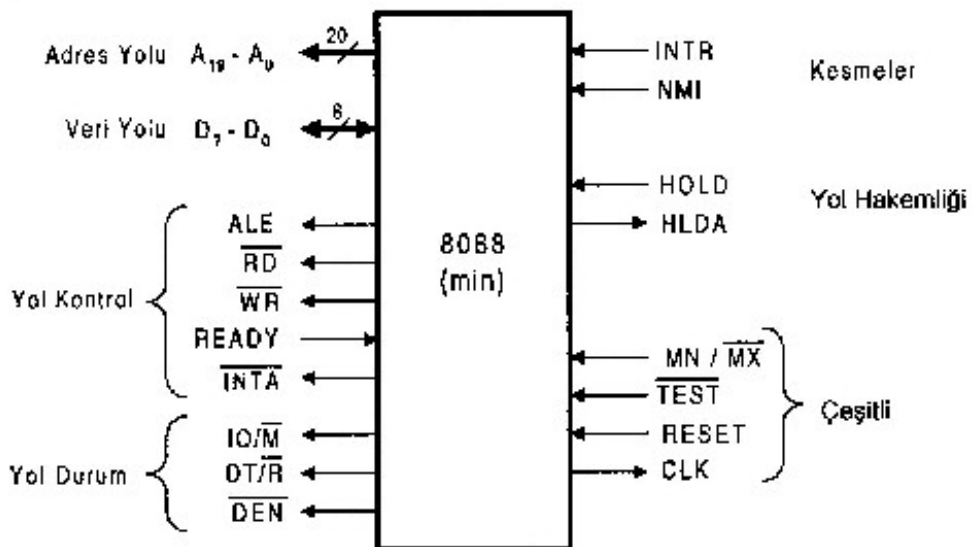
Aşağıda her iki işlemcinin tümdevre uçları ve fonksiyonları açıklanacaktır.

- **AD₇ –AD₀** (8088 ve 8086)- **Adres/veri yolu** : 8088'in seçimli (multiplexed) Adres /veri yolunu oluşturur. ALE sinyali aktif olduğunda (lojik 1), hafıza adresinin veya I/O port numarasının en sağ düşük 8-bit'ini tutar. ALE pasif olduğunda (lojik 0) veri, bu yolda bulunur. Bu yol, bir tutma tasdikinde (hold acknowledge) yüksek empedans durumundadır.

- **A₁₅ –A₈ (8088) – Adres yolu** : Bir yol çeviriminde, hafıza adres bit'lerinin üst yarısını sağlar. Bu adres hatları, bir tutma tasdikinde yüksek empedans durumundadır.
- **AD₁₅ –AD₈ (8086) –Adres/veri yolu** : 8086'nın seçimli Adres/veri yolunun üst bölümünü oluşturur. Bu hatlar, ALE lojik 1 olduğunda A₁₅-A₈ bitlerini içerir. Bu yol, bir tutma tasdikinde yüksek empedans durumundadır.
- **A₁₉ / S₆ , A₁₈ / S₅ , A₁₇ / S₄ , A₁₆ / S₃ –Adres /Durum yolu** :Seçimli yol olup A₁₉-A₁₆ adres hatları ile S₆-S₃ durum bitlerini içerir. Bu sinyaller bir tutma tasdikinde yüksek empedans durumundadır.



Şekil 3.6 8088 işlemcisinin max modu



Şekil 3.6 8088 işlemcisinin min modu

S₆ durum bit'i lojik '0' da kalır. S₅ bit'i IF bayrak bit'inin durumunu belirtir ve S₄ ve S₃ bir yol çevriminde, tablo 3.1 'de verildiği gibi, hangi segmente erişildiğini gösterir.

TABLO 3.1. S ₄ ve S ₃ Fonksiyonları		
S ₄	S ₃	Fonksiyon
0	0	ES
0	1	SS
1	0	CS
1	1	DS

- **RD- Okuma (Read)**: Lojik 0 olduğunda mikroişlemcinin hafızadan veya bir giriş cihazından veri okuduğunu dış lojiğe bildirir. Bu sinyal, bir tutma tasdikinde yüksek empedans durumundadır.
- **READY- Hazır** : Mikroişlemcinin bir giriş sinyali olup işlemcinin zamanlamasına **bekleme durumları (wait states)** ekleme için kullanılabilir. Eğer bu giriş, lojik 0 yapılırsa, mikroişlemci bekleme durumuna girer ve boşta kalır. Bekleme durumu istenmiyor ise normalde bu giriş lojik 1 durumunda kalmalıdır.
- **INTR – Interrupt Request (Kesme isteği)** : Donanım kesmesi için kullanılır. IF=1 olduğunda INTR girişi lojik 1 yapılırsa, x86 işlemcisi bir dış donanım kesmesi almış olur ve bunun neticesi, o an yürütülen komut tamamlandıktan sonra, kesme tasdik (INTA) çevrimine girer.
- **TEST – Test** : WAIT komutu tarafından test edilen bir giriştir. Eğer TEST lojik 0 ise, WAIT komutu bir NOP gibi davranır. Eğer TEST lojik 1 ise, WAIT komutu TEST girişinin lojik 0 olmasını bekler. Bu uç 8087 nümerik işlemcisiyle beraber kullanılır.
- **NMI – Nonmaskable Interrupt (Maskelenemeyen kesme)** : INTR gibi, bir dış donanım kesme girişidir. Bu kesmenin oluşması için IF bit'inin 1 olması gerekmez.
- **RESET –Reset** : Bu giriş, en az 4 saat periyodu lojik 1 seviyesinde tutulduğunda mikroişlemci kendisini sıfırlar ve işlemci FFFF0h adresindeki komuttan itibaren çalışmaya başlar. Ayrıca IF bayrağı da sıfırlanır.
- **CLK – Clock (saat)** : Mikroişlemcinin temel zamanlama sinyalini sağlar.
- **VCC – Pover supply (Besleme kaynağı)** : +5V besleme
- **GND – Ground (Toprak)** : İşlemci iki toprak girişine sahip olup her ikisinin de toprağa bağlanması gerekmektedir.
- **MN / MX – Minumum / Maximum mode** : İşlemcinin minumum veya maximum modlarından birinde çalışmasını seçme girişidir. Bu girişin +5V olması durumunda, işlemci minumum modunda, toprağa bağlanması durumunda maximum modda çalışır.
- **BHE / S₇ – Bus High Enable (Yol Yüksek Aktif)** : 8086 işlemcisinde, bir okuma veya yazma çevriminde, veri yolunun üst 8- bit'inin (D₁₅ – D₈) aktif olduğunu dış lojiğe bildirir. S₇ 'nin durumu her zaman lojik 1'dir.

3.1.10 MİNİMUM MOD UÇLARI

8086/8088 işlemcisinin minimum mod çalışması MN /MX ucunun +5V'a bağlanmasıyla sağlanır.

- **IO / M (8088), M / IO (8086) – Input / Output or Memory (Giriş / Çıkış veya Hafıza) :** Mikroişlemcinin bir I / O cihazı veya hafıza modülü üzerinde işlem yaptığını dış lojiğe bildirir. Bu sinyal, bir tutma tasdikinde yüksek empedans durumundadır.
- **WR – Yaz (write) :** Bu sinyal lojik 0 olarak mikroişlemcinin bir hafıza veya I/O cihazına bir veri yazdığını belirtir. Bu sinyal, bir tutma tasdikinde yüksek empedans durumundadır.
- **INTA – Interrupt Acknowledge (kesme tasdik) :** INTR girişine işlemcinin cevap sinyalidir. Bu sinyal genelde bir kesme isteğine karşılık olarak, bir kesme vektör numarasının dışardan okunmasında, dış lojik tarafından kullanılır.
- **ALE – Address Latch Enable (Adres Latch Aktif) :** Lojik 1 seviyesi ile işlemcinin Adres / veri yolunda adres bilgisinin olduğunu belirtir.
- **DT /R – Data Transmit / Receive (veri Gönderme/ Alma) :** Lojik 1 olduğunda mikroişlemcinin dışarıya veri yolladığında ve lojik 0 olduğunda ise, mikroişlemcinin dışardan veri okuduğunu belirtir. Bu sinyal dış veri yolu buffer'larını aktif etmede kullanılır.
- **DEN – Data Bus Enable (Veri Yolu Aktif) :** Harici veri yol buffer'larını aktif etmede kullanılır.
- **HOLD – Hold (tutma) :** Doğrudan hafızaya erişim (Direct Memory Access – DMA) tek girişidir. Eğer bu giriş lojik 1 yapılırsa, mikroişlemci çalışmasını durdurur ve adres, veri ve kontrol yolunu yüksek empedans durumuna koyar. Bu girişin lojik 0 olma durumunda işlemci normal çalışmasına devam eder.
- **HLDA – Hold Acknowledge (Tutma Tasdik) :** İşlemcinin tutma durumuna girdiğini belirtir.
- **SS0 (8088) – Status 0 (Durum 0) :** Mikroişlemcinin maximum mod çalışmasındaki S0 ucuna karşı gelir.Tablo 3.2'de görüldüğü gibi, bu sinyalin IO/M ve DT/R sinyalleriyle beraber kullanılmalarıyla, bir yol çevriminin durumu dış lojiğe bildirilir.

IO / M	DT / R	SS0	Fonksiyon
0	0	0	Kesme Tasdik
0	0	1	Hafıza Okuma
0	1	0	Hafıza Yazma
0	1	1	Durma (Halt)
1	0	0	Kod Erişim
1	0	1	I / O Okuma
1	1	0	I / O Yazma
1	1	1	Pasif

3.1.11 MAXİMUM MOD UÇLARI

Mikroişlemcinin diğer harici yardımcı işlemlerle (Nümerik ve I / O işlemcisi gibi) beraber kullanılabilmesi için MN / MX ucunun toprağa bağlanması gerekir. Bu moddaki sinyaller ve fonksiyonları aşağıda verilmektedir.

- **S2, S1 ve S0 – Status Bits (Durum Bit’leri)** : O anki yol durumunu belirtirler. Bu sinyallerin normalde 8288 yol denetleyicisi (Bus controller) tarafından kodu çözülür.Tablo 3.3 bu durum sinyalinin fonksiyonlarını göstermektedir.

S2	S1	S0	Fonksiyon
0	0	0	Kesme Kabul
0	0	1	I / O Okuma
0	1	0	I / O Yazma
0	1	1	Durma (Halt)
1	0	0	Kod Erişim
1	0	1	Hafıza Okuma
1	1	0	Hafıza Yazma
1	1	1	Pasif

- **RQ / GT0 ve RQ / GT1 – Request / Grant (İstek / Kabul)** : İki yönlü olan bu uçlar DMA işleminde kullanılır.
- **LOCK – Lock (kilit)** : Mikroişlemcinin çıkış sinyali olarak, sistemdeki istenen çevre birimlerini kilitlemek amacıyla kullanılır. Bu uç LOCK ön ekinin (prefix) herhangi bir komutun önünde kullanılmasıyla aktif olur.
- **QS1 ve QS2 – Queue Status (Kuyruk Durumu)** : Dahili komut kuyruğunun durumunu belirtir. Bu sinyaller nümerik işlemci 8087 tarafından kullanılır. Bu uçların durumları Tablo 3.4’de görülmektedir.

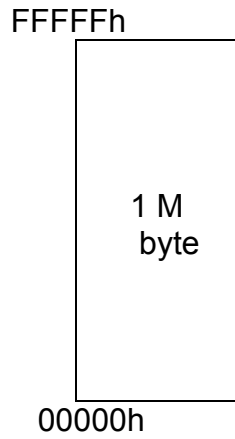
QS1	QS0	Fonksiyon
0	0	Bir işlem yok (kuyruk boşta)
0	1	İlk byte bir işlem kodu
1	0	Kuyruk boş
1	1	Bir sonraki byte bir işlem kodu

3.1.12 8086 /8088 HAFIZA MİMARİSİ

8086 / 8088 mikroişlemcisi 20-bit adres yolu ile toplam 1048576 (1M) byte hafıza hücresi adresleyebilmesine karşın, her iki işlemcinin fiziksel hafıza yapıları farklıdır.

Bununla beraber, bu işlemcilerin lojik hafızaları şekil 3.8 de görüldüğü gibi aynıdır. **Lojik hafıza**, genellikle yazılım tarafından programcıya görülen hafızaya verilen isimdir. Bu hafıza, donanım tasarımcısı tarafından görülen, gerçek hafıza yapısını oluşturan **fiziksel hafızadan** farklı olabilir.

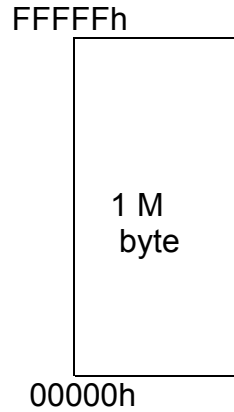
8088 ve 8086 mikroişlemcilerinin lojik hafızası 0000h tan başlayıp FFFFh'a kadar uzanır.Lojik hafıza genişliği bir byte (8-bit) olup bu adreslerin uzunluğu 1M byte hafıza bloğunu belirtir.mikroişlemci tarafından adreslenen 16-bit bir hafıza kelimesi, herhangi bir byte adresinden başlar ve peş peşe 2 byte işgal eder.



Şekil 3.8 8086 / 8088 lojik hafıza haritası

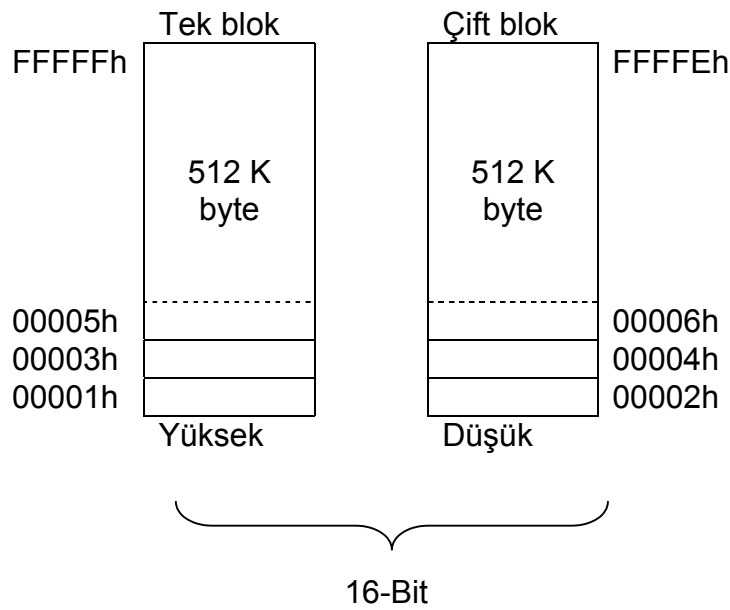
8086 mikroişlemcisinin fiziksel hafıza haritaları genişlik olarak birbirinden farklıdır.8088 hafızası 8-bit, 8086 hafızası ise 16-bit genişliğindedir.Programcı için hafıza her zaman 8-bit genişliğinde olmasına rağmen, fark sadece donanım tasarımcısı için bulunur.

Şekil 3.9 8088 mikroişlemcisinin fiziksel hafıza alanını göstermektedir.bu hafıza haritası şekil 3.8'de verilen lojik hafıza haritası ile aynıdır.8088'in hafıza arabirimi 8085a işlemcisine benzemektedir.



Şekil 3.9.8088 fiziksel hafıza haritası

8086 mikroişlemcisinin fiziksel hafıza haritası şekil 3.10'da görülmektedir. 8088'den farklı olarak ki ayrı hafıza bloğu içermektedir. Tek blok (yüksek hafıza) ve çift blok (düşük hafıza). Her 8086 bloğu 512k*8 olup toplam adreslenebilir hafıza yine 1M byte'tır. bu yapının avantajı, 8086, byte veya kelime (16-bit, word) verisini doğrudan adresleyebilmektedir. bundan dolayı, 8086, 16-bit bir kelimeyi bir işlemde okuyup yazabilmektedir. s(verinin adresinin çift olduğu sağlandığında). Buna karşın, 8088, 16-bit veri aktarımı için 2 okuma veya yazmaya gerek duyar. 8086 yazılımı daha hızlı çalışır. Çünkü 8086 bir çok komuta ve 16-bit veriye 8088'in iki katı hızında erişir.



Şekil 3.10 8086 fiziksel hafıza yapısı

3.1.13 SEGMENT'Lİ HAFIZA YAPISI

8086/8088 mikroişlemcisinde hafıza erişim, segment saklayıcıları yoluyla yapılır. Herbir segment hafıza bloğu 64k byte'tır. Şekil 3.11'de segment saklayıcıları ile adreslenen bir hafıza haritası örneği görülmektedir. hafıza alanında aynı anda 4 farklı segment bulunabilmektedir. Bunlar: kod segment (CS), veri segment (DS), ekstra segment (es) ve yığın segmentidir(SS).

AFFFFh	Yığın segmenti	
A0000h		SS=A000
8FFFFh	Ekstra segment	
80000h		ES=8000
3FFFFh	Veri segmenti	
30000h		DS=3000
1FFFFh	Kod segmenti	
10000h		CS=1000

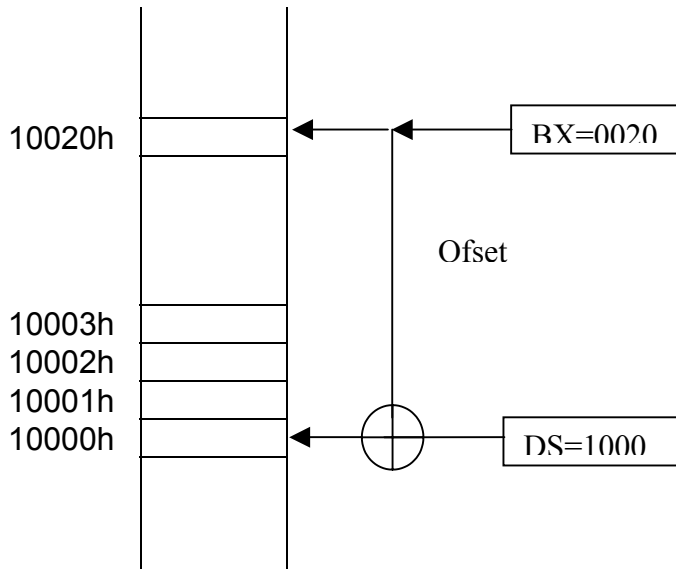
Şekil 3.11 Segment saklayıcıları ile adreslenen bir hafıza haritası örneği

Her segment saklayıcısı, 20-bit adresin 16-bit kısmını tutar. segment saklayıcıda bulunan 16-bit adresin düşük değerli bölümüne, 0h(0000₂) eklenir. Ayrıca bir ofset (indis) bu adresle toplanarak, donanım tarafından otomatik olarak 20-bit adres elde edilir. Bu işleme lojik adresin fiziksel adrese çevrilmesi denir. segment saklayıcılarına göre aşağıda açıklanacağı gibi, ofset, değişik saklayıcılarda gelir.

Eğer bir 8086/8088 sistemi sadece 64k byte hafıza içeriyorsa, bütün 4 segment saklayıcıları 0000h ile yüklenir ve segmentler üst üste çıkarılır. Bu durumda adres aralığı X0000h-XFFFFh arasında değişir. (X herhangi bir 16-lı rakamdır)

Her bir segment saklayıcısı, özel bir fonksiyona sahiptir ve bir veya daha çok indis veya işaretçi saklayıcısı ile ilişkilidir. bir hafıza adresi üretmek için, bir segment saklayıcısının içeriği, bir ofset adres adres tutan bir indis veya işaretçi saklayıcısına eklenir.

Şekil 3.12'de bir komut tarafından adreslenen bir verinin, fiziksel adresinin üretimi görülmektedir. Bu örnekte, veri segment saklayıcısı DS 1000h içermekte, yani veri segment 10000h fiziksel adresinden itibaren başlamaktadır. Taban saklayıcısı BX'te ofset adres 0020h bulunmakta, ve böylece fiziksel adres 10020h veya 1000h*10h+0020h olmaktadır



Şekil 3.12 Veri segmentinde bulunan bir hafıza hücresine erişim

Kod segment CS, program ve veri alanı olarak kullanılabilmesine rağmen, genelde, program kodlarının bulunduğu alandır. 8086/8088 tarafından yürütülecek bir sonraki komutun adresi, komut işaretçisi IP'ye $CS \cdot 10h$ içeriğinin toplanmasıyla elde edilir.

Veri segment'i DS, bir çok komut ve adresleme modu tarafından erişilen program verilerini tutar. Hafızadaki verinin adresi, BX, SI veya DI saklayıcılarından birine $DS \cdot 10h$ içeriğinin toplanmasıyla elde edilir.

Yığın (stack) segment'i SS, LIFO (last In First Out) tarzında çalışmaktadır. bir yığın hücresinin adresi, SP'nin içeriği artı $SS \cdot 10h$ 'tir. BP tarafından adreslenen veri de normalde yığın segment'inde bulunur.

ES string komutları tarafından kullanılan veri alanıdır. bir string komutu yürütüldüğünde hedef aders DI artı $ES \cdot 10h$, kaynak veri ise, SI artı $ES \cdot 10h$ 'tir.

3.1.14 SEGMENT'Lİ HAFIZA YAPISININ AVANTAJLARI

Yukarıda anlatılan segment'li hafıza yapısı, ilk bakışta şaşırtıcı ve zor görülebilir. bu hafıza yapısı için hatırlatılması gerekenler özetle şunlardır: Program işlem kodları CS alanından okunmakta, program verileri DS ve ES alanlarında saklanmaktadır. yığın işlemleri ise, SP ve BP saklayıcılarını kullanarak SS üzerinde işlem yapar.

Ayrı kod ve veri alanlarının olmasının ilk avantajı, bir programın, farklı veri blokları üzerinde çalışabilmesidir. bu işlem, DS saklayıcısına farklı bir bloğa işaret eden yeni bir adresin yüklenmesiyle yapılır.

Segment'li hafıza yapısının en büyük avantajı , lojik adresler üreten x86 programlarının hafızanın herhangi bir yerine yüklenip çalıştırılabilmesidir.Bunun nedeni lojik adreslerin her zaman.CS taban adresinden bağımsız olarak, 0000h ile FFFFh arasında değişmesidir.

Çok görevli (multi-tasking) x86 tabanlı bir ortamı veya Windows 95/98/NT işletim sistemini düşünelim.Bir çalışan aktif programın geçici olarak sabit diske saklandığını ve onun yerine, yeni bir programın getirildiğini farz edelim.bu çeşit programlar hafızanın herhangi bir yerinde çalışacakları için, **tekrar yerleştirilebilir (relocatable)** olarak adlandırılır. segment saklayıcılarının yani program, veri ve yığın saklayıcılarının taban adreslerinin değiştirilmesiyle, programlar hafızanın herhangi bir yerinde çalışabilmektedir.

Bölüm 3'te gördüğümüz 8085A ve benzeri birçok 8-bit mikroişlemci ve mikro denetleyicide, bu şekilde bir hafıza adres üretimi olmadığı için programların tekrar yerleştirilebilirliği mümkün değildir. Yani, 8085A işlemcisi için program ve verilerin başlangıç adresleri , ORG (origin) gibi assembler ifadeleri ile belirtilip kod üretimi yapıldıktan sonra bu adreslere, program ve veriler yüklenebilir.bu şekilde program ve veriler bir bakıma, hafızaya kök salar ve başka bir yere taşınmaz.program ve verileri başka bir adrese yükleyebilmek için ORG adresindeki adresler değiştirilir, sonra tekrar derleme yapılır ve yeni ikili kod üretilir.bu sayede yeni adres yerleşimi için kod elde edilir.

3.2 80286 MİKROİŞLEMCSİ

8086/8088 mikroişlemcisinin halefi 80286 dir.bu mikroişlemcinin kullanıcı açısından 8086/8088'e göre 3 temel üstünlüğü bulunmaktadır.

- Birincisi, **gerçek (real)** mod ile **korunmalı (protected)** moda sahip olmasıdır. Donanım **hafıza yönetim (memory management)** sistemi sayesinde hafızada birden çok programın güvenli bir şekilde çalışması mümkün hale gelmiştir.
- İkincisi, 8086 gibi, 8088'den farklı olarak 16-bit veri yoluna sahiptir. Bus hafıza-mikroişlemci arasındaki veri akış band genişliğini iki katına çıkarır. Ayrıca, fiziksel hafıza 2^{20} byte'dan (1M byte) 2^{24} byte'a (16M byte) genişletilmiştir.
- Üçüncüsü, daha hızlıdır ve daha yüksek saat hızlarında çalışabilir.bu faktörle, 80286 tabanlı bir sistemi 8088 tabanlı bir sisteme göre, 5 ile 10 kat daha hızlı yapmaktadır.

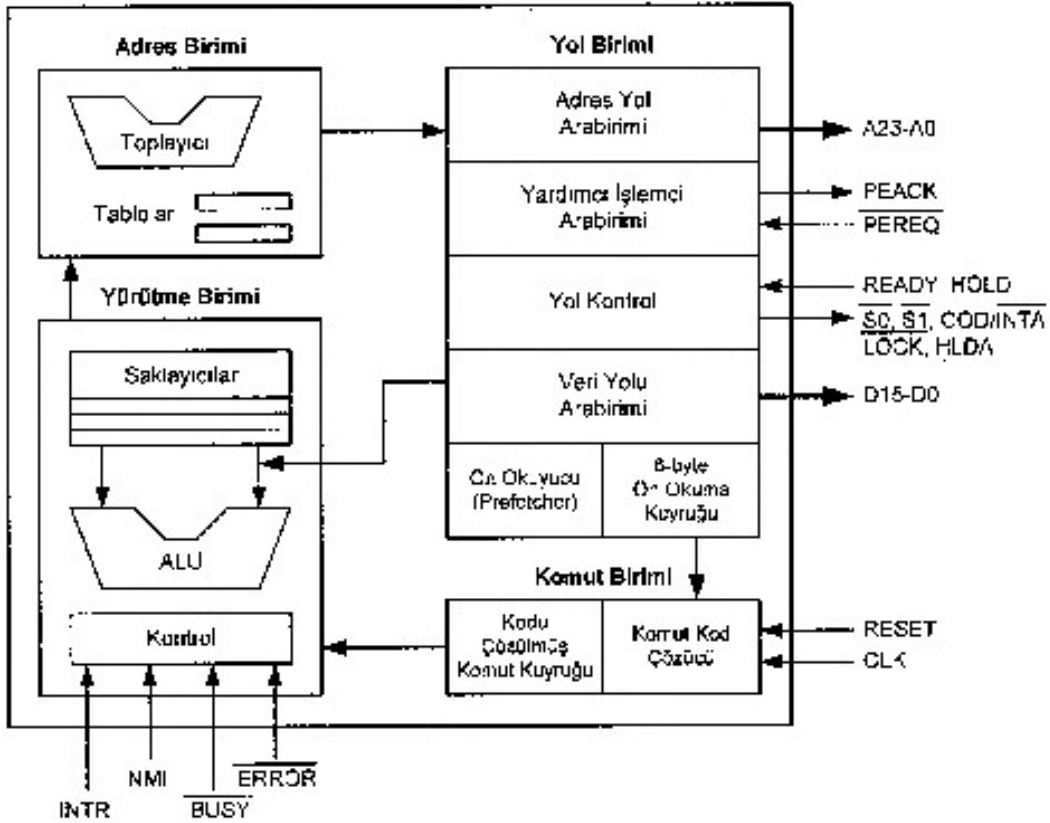
Ayrıca, 80286 mikroişlemcisi yeni adresleme ve hafıza koruma özelliklerini desteklemek için yeni ek komutlara sahiptir.bu işlemci IBM PC/AT ve bazı PS/2 bilgisayarlarının ana mikroişlemcisi olmuştur.

3.2.1 İÇ MİMARISI

80286 bu yüksek performansı basitleştirilmiş hali şekil 3.13'de görülen, içinde bulunan birbirinden bağımsız 4 fonksiyonel birim sayesinde sağlar.yol birimi (bus unit) CPU için gerektiğinde işlem kodu ve veri okuma/saklama gibi bütün yol işlemlerini yerine getirir.CPU, eğer yapacak başka bir işlemi yoksa, 6 byte2a kadar konutları önceden oku ve bunları komut birimine gönderir.

Komut birimi, yol birimi tarafından okunmuş ham verileri alır ve sonraki yürütme için kodunu çözer. üç taneye kadar tam kodu çözülmüş komut, bu birimde bir anda bulunabilir.kodu çözülmüş komutların CPU içinde hazır olarak bulunması CPU yürütme hızını artırır.

Yürütme birimi (execution unit) komut biriminden gelen komutları işler .Bazı konutlar adres içermektedir.bu adresler daha sonraki işlemler için adres birimine verilir.adres birimi (address unit)bütün adresleme ve **görüntü hafıza (virtual memory)** işlemlerini yerine getirir.(görüntü hafıza, bir programın bilgisayarın fiziksel olarak sahip olduğu hafızadan daha fazla hafıza kullanılabilmesini sağlayan bir tekniktir.Program parçalarının, yürütme sırasında gerektiğinde, hafıza ile disk arasında değiştirme prensibine dayanır.) Adres biriminin çıkışı, okuma ve yazma adreslemesi için yol birimine iletilir.



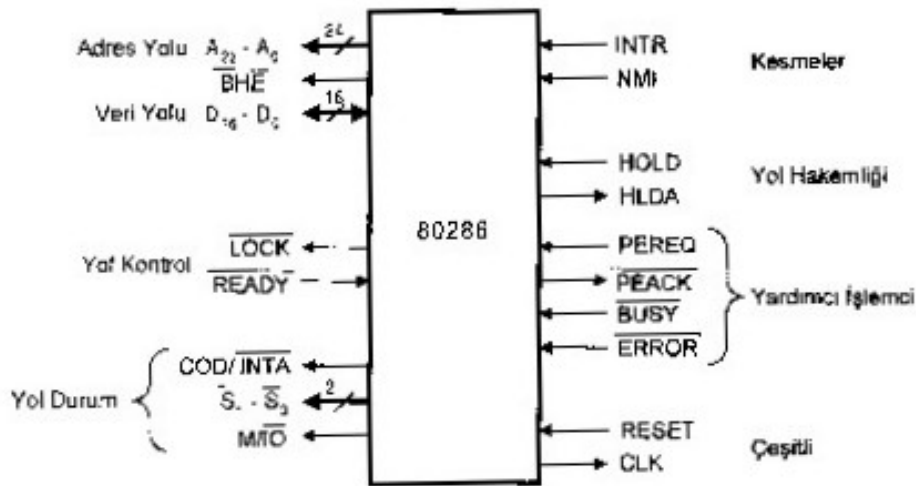
Şekil 3.13 80286 işlemcisinin iç yapısı

3.2.2 DIŐ MİMARİSİ

80286 tümdevresi, 40 sinyalden daha fazla gerektirdiđi için, intel firması 80286 için daha önceki tümdevre kılıflarından farklı bir yapı sundu. Bazı sinyalleri çođullamalı (multiplexed) yapıp 48 uç sunmak yerine, her tarafında 17 ucu olan, kare 68 uçlu bir tümdevre kılıfı kullandı. Bu paket 24 adres sinyali, 16 veri sinyali, 16 kontrol sinyali, saat, güç, toprak ve bađlı olmayan birkaç uca sahiptir. Őekil 3.14 80286 mikroişlemci tümdevresinde yer alan sinyallerin lojik gruplandırılmış yapısını göstermektedir.

Daha önce gördüğümüz 8086/8088 mikroişlemcisinde olduđu gibi, minimum/maksimum mod ayrımı 80286'da yoktur. 80286 mikroişlemcisi 24-bit adres ve 16-bit veri sinyallerine sahiptir.

Adreslemeyle ilgili, 8086 mikroişlemcisinde olduđu gibi, BHE (bus high enable) sinyali bulunmaktadır.



Őekil 3.14 80286 işlemcisinin uçları

80286'nın yol durum sinyalleri, 8086/8088'den oldukça farklı olmasına rağmen, temel olarak yaptıkları görevler aynıdır. Bu sinyaller, hafızadan veya I/O biriminden okuma, hafızaya veya I/O birimine yazma veya başka bir yol ayrımını gösterir. S0, S1, MIO ve COD/INTA sinyalleri, 16 farklı yol durumunu belirtir. Bu durumlardan 7 tanesi faydalıdır, geri kalanlarının bir fonksiyonu yoktur. Intel eski ürettiđi tümdevreler ile uyumluluđu gözetmiş olmasaydı daha temiz bir tasarım yapabilirdi. Aslında bu tasarım yaklaşımı Intel'in bütün mikroişlemcileri için geçerlidir.

Yol kontrol sinyallerinden olan LOCK ve READY, 8086/8088'deki benzeri görevlere sahiptir. (yolu kitleme ve yol çevrimlerine bekleme durumları ekleme). Kesme girişleri INTR ve NMI de aynıdır.

Tutma (HOLD) ve tutma kabul (HKDA – Hold acknowledge) yol hakemliđi (bus arbitration) iřleminde kullanılır.Ortak yolu kullanmak , isteyen bir iřlemci, HOLD giriřini aktif duruma getirir.80286 bu isteđi grdkten sonra, yeni yol hakemine yolu bırakırken, yol uęlarını yksek empedans konumuna getirerek ve HLDA sinyalini aktif yapar.yola hakim olan cihaz, yolu bıraktıđı zaman, HOLD sinyalini bırakır (tersini alır)ve 80286 normal ęalıřmasına devam eder.

80286 yardımcı iřlemci , ięin 4 tane sinyale sahiptir.Bu sinyallerin yanında, 80286 diđer mikroiřlemcilerde grdğmz ve aynı fonksiyonlara sahip RESET, CLK, gę ve toprak gibi sinyallere de sahiptir.