

Mikrobilgisayar Sistemleri ve Assembler

Bahar Dönemi

Konu Başlıkları

- Mikrobilgisayar sisteminin genel yapısı ,Mimariler,Merkezi işlem Birimi
- RAM ve ROM bellek özellikleri ve Çeşitleri
- Bir Mikroişlemcinin fiziksel yapısı ve bileşenleri
- Adres Yolu, Kontrol yolu ve Veri yolu
- Kaydediciler, ALU , Zamanlama ve kontrol Birimi
- Akümülatör, İndis kaydediciler, yığın kaydedici ,program sayıcı,Bayraklar
- Sayı sistemleri hakkında ön bilgi ve Adresleme Modları
- Assembly Dilinin Yapısı ve Temel Kavramları
- Assembly Dilinde kullanılan Komutlar
- Assembly Dilinde aritmetik işlemler

Giriş:

Bilgisayar sistemleri iki temel öğeden meydana gelir. Bunlar,

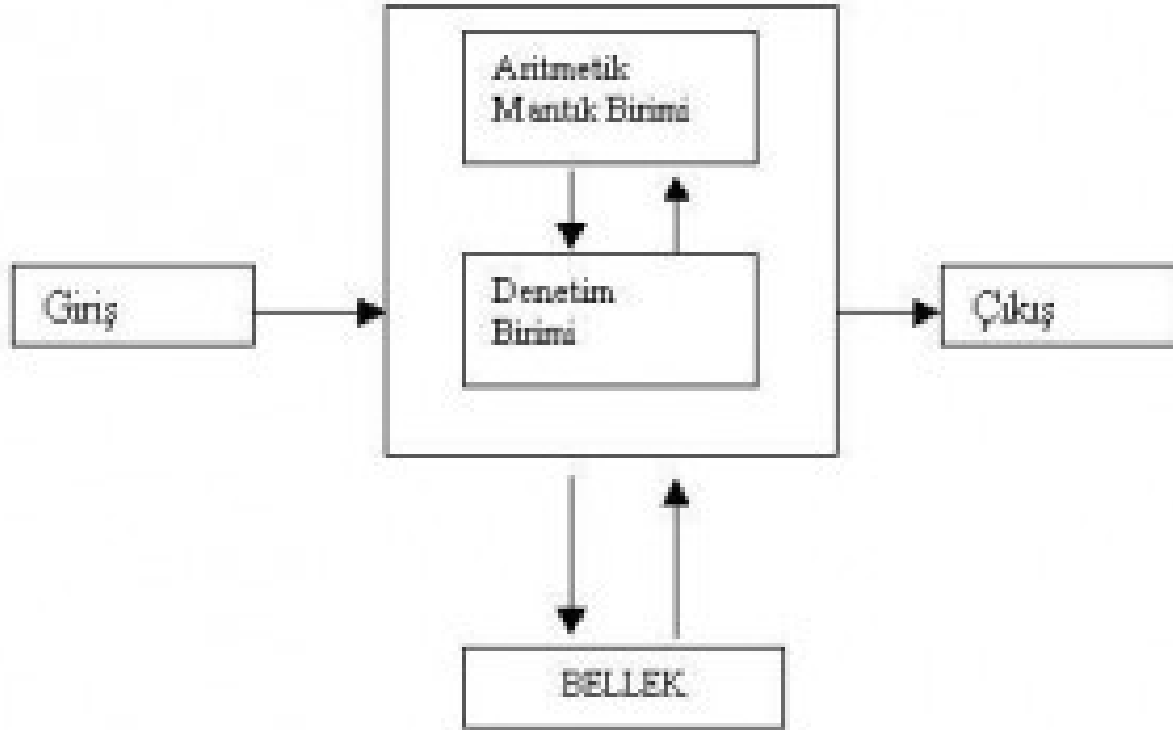
- Yazılım
- Donanım

YAZILIM: Donanımın hangi yöntemle göre çalışacağını gösteren bir sanal uygulamadır.

DONANIM: Yazılıma göre belli zamanlarda devreye girerek fonksiyonları yerine getirmekle görevlidir.

Bilgisayarı oluşturan bir sistemdeki temel elemanlar;

- Mikroişlemci
- Bellek
- Giriş/Çıkış birimleri



Mikroişlemcinin Temel Özellikleri

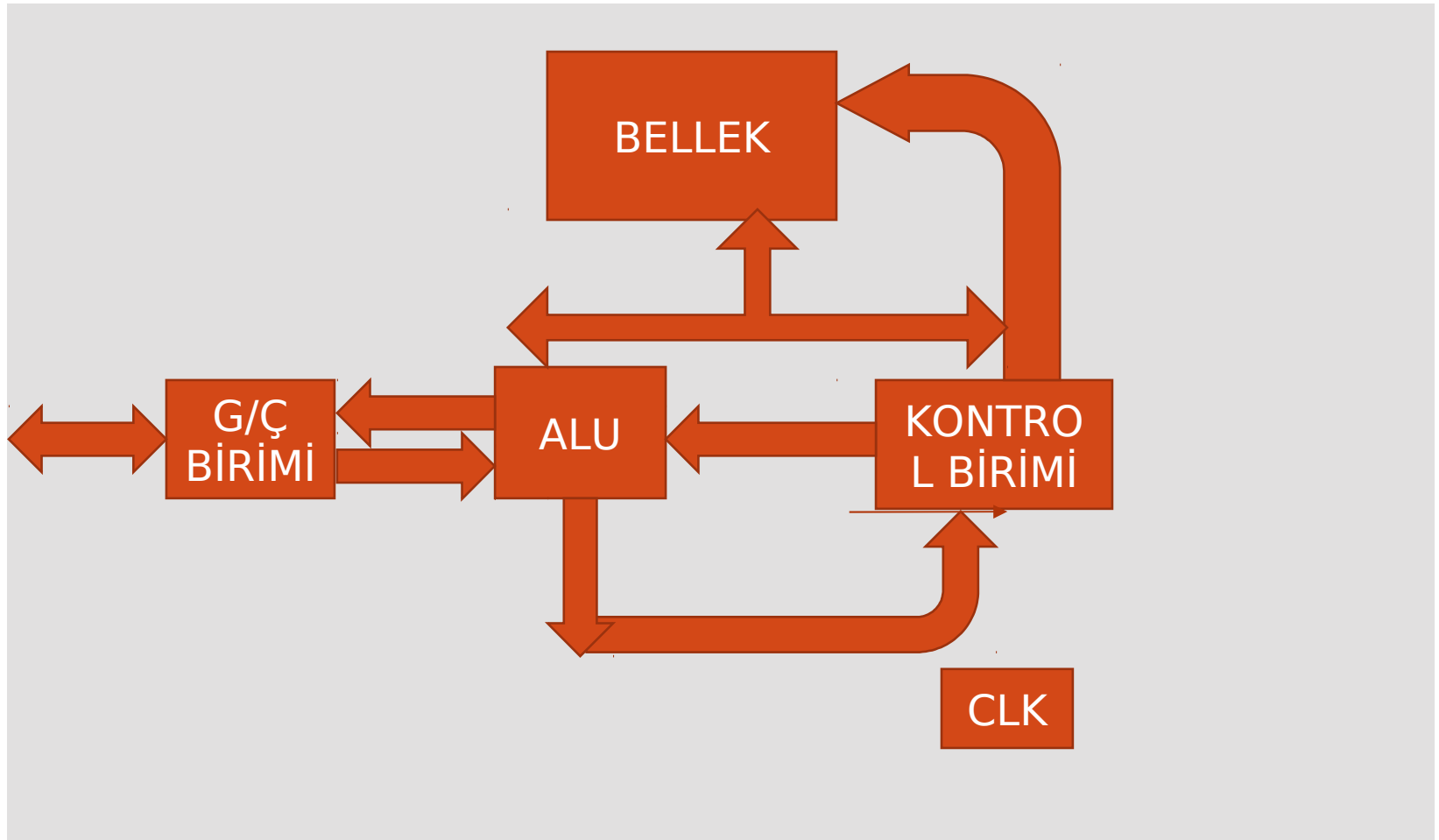
- Mikroişlemcinin bir defada işleyebileceği kelime uzunluğu
- Mikroişlemcinin tek bir komutu işleme hızı
- Mikroişlemcinin doğrudan adresleyeceği bellek büyüklüğü
- Programcının üzerinde çalışabileceği kaydedici çeşitleri
- Programcının kullanabileceği değişik türdeki komutlar
- Programcının bellek adreslerken gerek duyacağı adresleme modları

- **Kelime Uzunluęu:**Paralel olarak iřlenen veri bitlerinin sayıdır.Kelime, iřlemcideki genel amaęlı kaydedicilerin byklę ve aynı zamanda her bir bellek alanı kapasitesidir.
- **Komut iřleme Hızı :** Frekans ve zaman ters orantılıdır. iřlemcinin frekansı ne kadar yksekse, komut iřleme hızı dřer. Yani bu komutların daha kısa srede iřlenmesi anlamına gelir. ($F.T=1$)
- **Adres Byklę :** Bellekler mikroiřlemci tarafından adres yoluyla adreslenirler. Adres yolunun sayısı ne kadar fazla olursa adresleme kapasitesi de ona gre byk olur.
- **Kaydedici Sayısı Ve Yapısı**
- **Deęiřik Tipteki Komutlar :** Bir mikroiřlemcide komut sayısının ok olması sisteme kolaylık saęlar ama bizim iin nemli olan aynı zamanda bu komutların az cycle (devir, dolařım, dnme) ile alıřmasıdır.
- **Farklı Adresleme Metotları :** Bir verinin nasıl ve ne Őekilde yerleřtireleceęi veya bellekten nasıl ve hangi yntemle aęrılacaęının belirlenmesi (doęrudan, dolaylı, indisli adresleme gibi) programcıya ekstra kolaylık saęlar.

Mikroişlemciler farklı yapılarda olmasına rağmen temelde şu birimleri içerir :

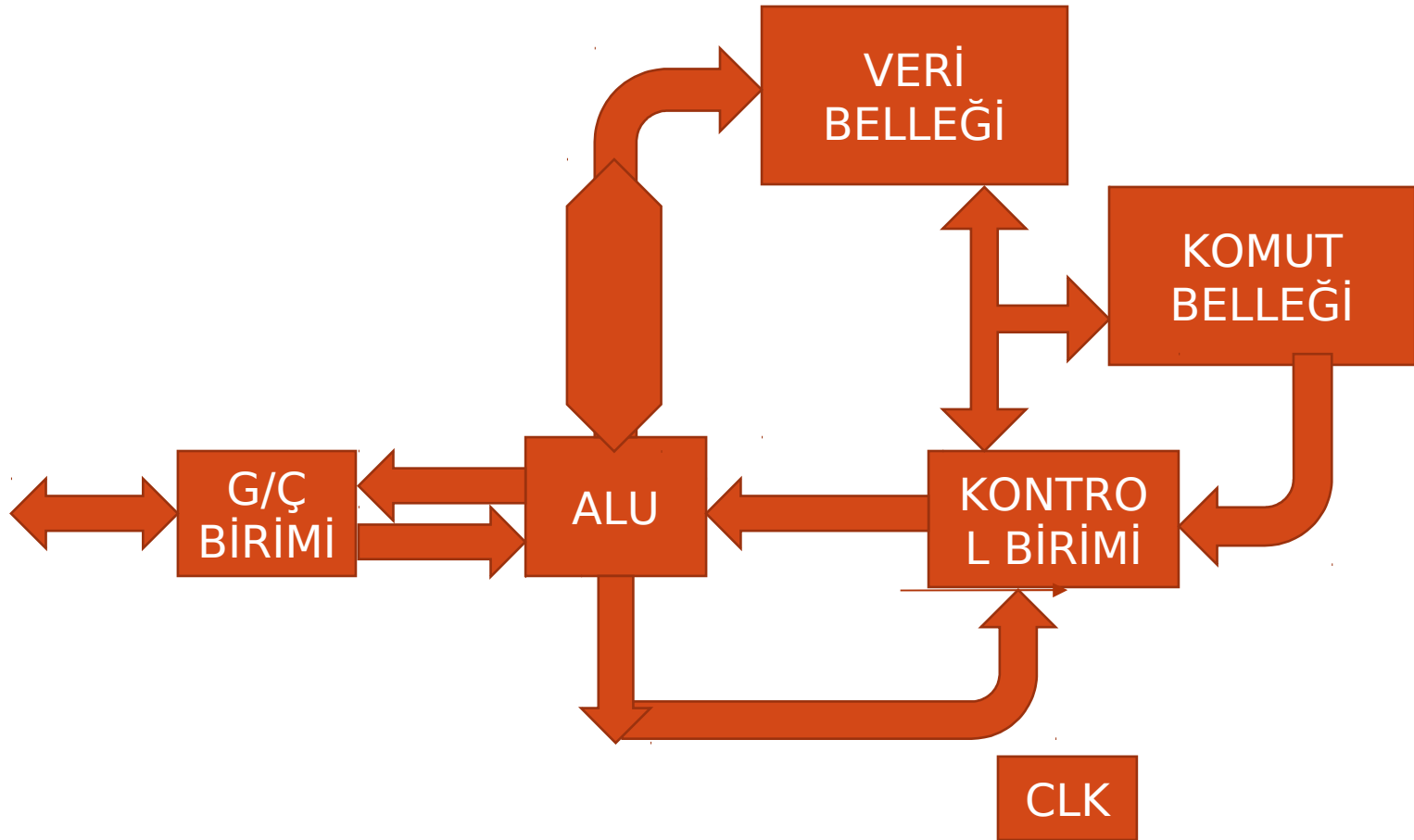
- Akümülatör
- Program Sayıcı (Program Counter - CP)
- Komut Saklayıcı (Instruction Register - IR)
- Komut Kod Çözücüsü
- Durum Saklayıcısı (Condition Code Register - CCR)
- Aritmetik Mantık Birimi (Arithmetic Logic Unit - ALU)
- Kontrol Birimi
- Yığın Göstergesi (Stack Point - SP)

Von Neuman Mimarisi



- Princeton adıyla da bilinen Von Neuman mimarisinde, veri ve komutlar bellekten tek bir yoldan mikroişlemciye getirilerek işlenmektedir. Program ve veri aynı bellekte bulunduğundan ,komut ve veri gerekli olduğunda aynı iletişim yolunu kullanmaktadır. Her bir komut için ayrı bir komut saykılı (fetch cycle) gerekir.
- Von Neuman mimarisinde veri bellekten alınıp işlendikten sonra tekrar belleğe gönderilmesinde çok zaman harcanır.
- Veri ve komutlar aynı bellekte depolandığından ,yanlışlıkla komut diye veri alanından kod getirilmesi sıkıntılara sebep olmaktadır.
- Bellek doğrusal olarak adreslenmiştir. Hem kod hem veri adresleri tek sıralı olarak adresler verilmiştir.

Harvard Mimarisi



- Veri ve komutlar ayrı belleklerde tutulmaktadır.Böylece veri ve komut aktarımında iletişim yolları da birbirinden bağımsız yapıdadır.
- Komutla veri aynı saykılada farklı iletişim yolundan ilgili bellekten alınıp işlemciye getirilebilir.
- Getirilen komut işlenip ilgili verisi veri belleğinden alınırken sıradaki komut ,komut belleğinden alınır.Önden alıp işleme,hızı iki katına çıkarabilir.

Mikroişlemci Mimarisi

CISC (Complex Instruction Set Computer - Karmaşık Komut Setli Bilgisayar)

Bu mimari programlanması kolay ve etkin bellek kullanımını sağlayan tasarım felsefesinin bir türüdür.

- Performans düşüklüğüne sebep olur.
- İşlemciyi daha karmaşık hale getirir.
- Yazılımı basitleştirir.

CISC mimarisinin karakteristik iki özelliğinden birisi; değişken uzunluktaki komutlar, diğeri ise karmaşık komutlardır. Değişken ve karmaşık uzunluktaki komutlar bellek tasarrufu sağlar. Karmaşık komutlar iki yada daha fazla komutu tek bir komut haline getirdikleri için hem bellekte hem programda yer alması gereken komut sayısından tasarruf sağlar.

CISC Mimarisinin Üstünlükleri

- Mikro programlama, assembly dilinin yürütülmesi kadar kolaydır ve sistemdeki kontrol biriminden daha ucuzdur.
- Yeni komutlar ve mikro kod ROM'a eklemenin kolaylığı tasarımcılara CISC makinelerini geriye doğru uyumlu yapmalarına izin verir.
- Herbir komut daha yetenekli olmaya başladığından, verilen bir görevi yürütmek için daha az komut kullanılır.
- Mikro program komut kümeleri, yüksek seviyeli dillerin yapılarına benzer biçimde yazılabildiğinden, derleyici karmaşık olmak zorunda değildir.

CISC Mimarisinin Dezavantajları

- Komut kodu ve çip donanımı bilgisayarların her kuşağıyla birlikte daha karmaşık hale gelmiştir.
- Makine performansını düşürür
- Çoğu özel güçlü komutlar geçerliliklerini doğrulamak için yeteri kadar sık kullanılmıyor. Tipik bir programda mevcut komutların yaklaşık %20sini kullanılıyor.
- Komutlar genellikle bayrak (durum) kodunu, komuta bir yan etki olarak kurar. Bu ise ek cycle (çevrim) yani bekleme demektir. Aynı zamanda, sıradaki komutlar işlem yapmadan önceden bayrak bitlerinin mevcut durumunu bilmek durumundadır. Buda yine ek cycle demektir.

RISC (Reduced Instruction Set Computer - İndirgenmiş Komut Takımı Bilgisayarı)

- Risc mimarisi, CISC mimarili işlemcilerin kötü yanlarını piyasanın tepkisi ve ona bir alternatif olara, işlemci mimari tasarımlarında söz sahibi olan [IBM](#), [Apple](#) ve [Motorola](#) gibi firmalarca sistematik bir şekilde geliştirilmiştir.

Risc Mimarisinin Özellikleri

- Risc mimarisi aynı anda birden çok komutun birden fazla birimde işlendiği iş hattı (pipelining) tekniği ve süperskalar yapılarının kullanımıyla yüksek bir performans sağlamıştır. Pipelining yöntemini şu şekilde açıklayabiliriz :

Çamaşır yıkamaya benzetebileceğimiz kullanışlı bir metottur. O anda sadece ihtiyacımız olan çamaşırları yıkamalı, kurulamalı ve katlamalıyız. 30 dakikada çamaşırları yıkamak, daha sonra 40 dakikada kurulama ve 20 dakikada katlama işlemini yapalım. Daha sonra 2inci parti çamaşırları makineye atalım, kurulayalım, katlayalım ve 3üncü ve 4üncü parti çamaşırlarımız için bu işlemleri tekrarlamış olalım. 18.00da başladığımız çamaşır işi için mümkün olduğu kadar çalıştık ama gece yarısına kadar sürdü.

- Fakat akıllı bir yaklaşımla bu süreç kısaltılabilir. İlk parti çamaşırımız yıkandıktan sonra onları kurulama işlemine geçirmeden hemen önce 2inci parti çamaşırını makineye atalım. Daha sonra ilk parti çamaşırını kuruladıktan sonra 2inci parti çamaşırını kurulamaya alabilir ve 3üncü parti çamaşırını makineye atabiliriz ve o esnada ilk parti çamaşırının ütüleme işlemini yapabiliriz. Böylelikle 2 ve 3üncü çamaşır yıkama işlemini bir iş hattına sokmuş olduk. Bu yöntemle tüm çamaşırın yıkanma, durulanma ve ütülenme işlemi 21,30 da bitirebildik.

Risc mimarisinin özelliklerini sıralamak gerekirse :

- Komut kümesi küçülmüştür, adresleme modları azaltılmıştır.
- Bir çevrimlik zamanda bir komut işlenir.
- Aynı uzunluk ve sabit formatta komut kümesine sahiptir.
- Ana belleğe sadece “load” ve “store” komutlarıyla erişir, operasyonları kaydedici üzerinde yapar.
- Bütün icra birimleri mikrokodlar kullanmadan donanımsal çalışır.
- Yüksek seviyeli dilleri destekler.
- Çok sayıda kaydediciye sahiptir.

RISC Mimarisinin Üstünlükleri

- Hız
- Basit donanım
- Kısa tasarım zamanı