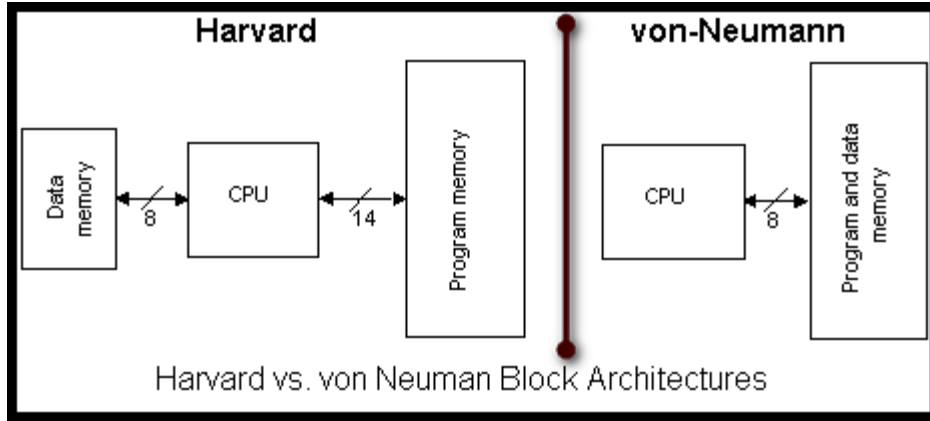


VON NEUMANN VE HARVARD MİMARİLERİ

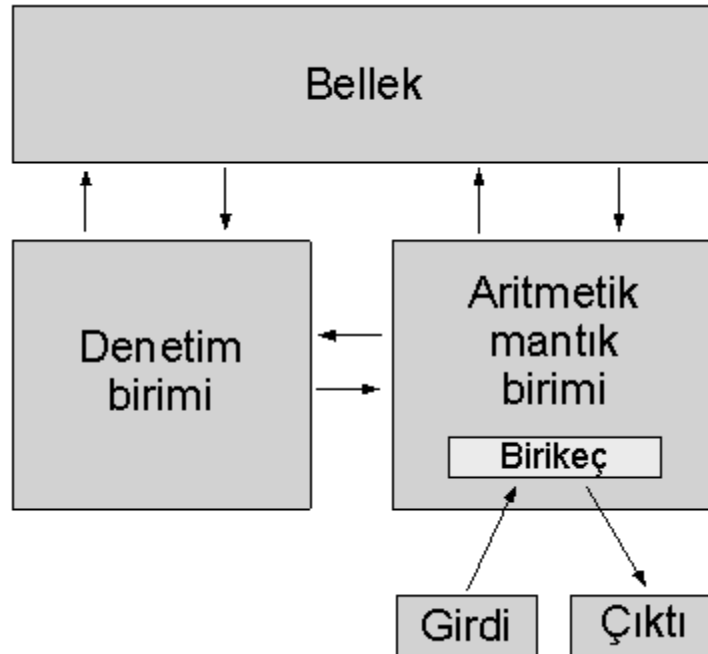
Harvard & Von Neumann Mimarileri

Harvard mimarisi, veri ve komutların Merkezi İşlem Birimine (*MİB* veya *CPU*) giden kanallarının **ayrılması** ile oluşturulmuş bilgisayar tasarımıdır. Von Neumann mimarisi, veri ve komutları **tek bir yığın** (*depolama*) biriminde bulunduran bilgisayar tasarımıdır.

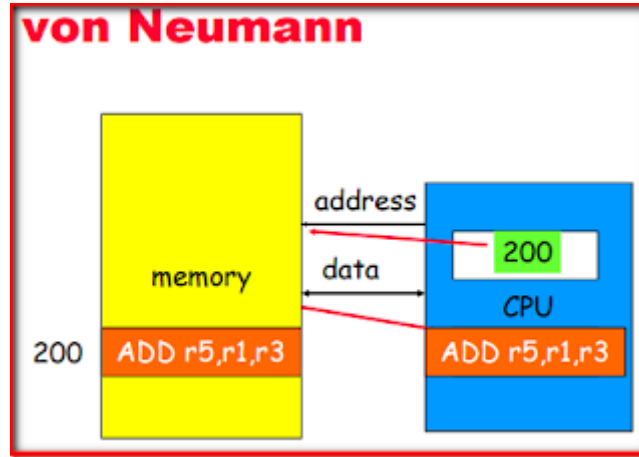


Von Neumann mimarisi

Verilerin ve program kodlarının aynı hafıza birimi üzerinde bulunduran tasarımıdır. Von Neumann mimarisinin temel tasarımı aşağıdaki gibidir:



"von Neumann mimarisi" isim olarak John von Neumann'ın 1945 tarihli makalesine dayanır. Bellek ile Merkezi işlem biriminin (MİB) hız açısından farklı gelişmesi **von Neumann dar geçidi** olarak bilinen soruna yol açmıştır. Bu sorun MİB ile bellek arası veri taşıma hızının, bellek miktarına göre çok düşük olmasından kaynaklanmaktadır. Bu nedenle, CPU zamanının büyük çoğunluğunu bellekten istenilen verinin gelmesini beklemekle geçirir. Son yıllarda CPU'ların hızları ile bellek erişim hızlarının arasındaki farkın açılması ile bu sorun daha da büyümüştür. Sorunu hafifletmek adına *cache memory* kullanılmaktadır ve *branch prediction yöntemi* geliştirilmiştir. von Neumann mimarisinin dar geçit sorunu dışında, en olumsuz yanı ise hatalı yazılımların (*buffer overflow gibi*) kendilerine, işletim sistemine ve hatta diğer yazılımlara zarar verebilme olasılığıdır.

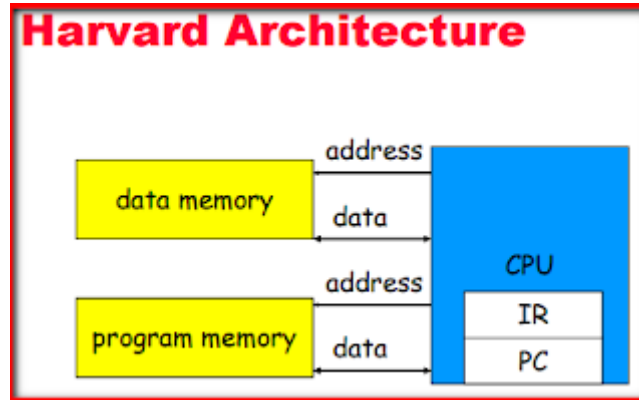


Harvard mimarisi

Harvard mimarisi, ismini ilk kez bu mimariyi kullanan bilgisayar **Harvard Mark I**'den almıştır. Bu mimariyi kullanan makinalar, veriler ile komutlar arasında herhangi bir köprü bulundurmazlar. Veri adresi ile program (*komut*) adresinin adresleme boyutları farklıdır. Harvard mimarisinin temel tasarımı aşağıdaki gibidir:



Günümüz bilgisayarlarında tam anlamıyla kullanıldığı söylenilemez. Yine de Von Neumann mimarisi ve Harvard mimarisinden ortak özellikler günümüz teknolojisinde kullanılmaktadır.



Von Neumann vs Harvard

- Von Neumann mimarisinde, işlemcinin doğası gereği ya **komutlarla** ya da **verilerle** uğraşmaktadır. Çünkü ikisi de aynı belleği paylaşmaktadır. İkisinin de aynı anda bellekten getirilerek işlenmesi durumu söz konusu değildir.
- Harvard mimarisini kullanan bir bilgisayarda ise komut ve veriler ayrı tutulduğu için, işlemci aynı esnada hem komutları değerlendirip hem verileri işleyebilir. Bir önbelleğe de gerek yoktur. Bu Harvard mimarisine bir avantaj sağlasa da Von Neumann mimarisinde komutlar verilerle bir tutulduğundan, program kendi kendine değişim gösterebilir.
- Harvard mimarisinde komutlar ile veriler arasında bir kanal yoktur bu yüzden kodların içine veri gömülmüş programlar çalıştırılırken veya kendi kendine değişim gösterilecek programlar için Von Neumann mimarisi temel alınır.

- Bellek adresleri aısından Harvard mimarisi iki ayrı adres kullandıđından; boş komut adresi boş veri adresinden de farklı olacaktır. Von Neumann mimarisinde ise ikisi de aynı adresi paylaşır.

Deđiştirilmiş Harvard mimarisi

Deđiştirilmiş Harvard mimarisi, Harvard mimarisindeki **veri/komut** bağlantısının eksikliđini gidermesi amacıyla yapılan tasarımsal düzenlemelere verilen isimdir. İşlemci halen veri ve komut erişimine de sahip olsa da aralarında bağlantılar mevcuttur. Bu düzenlemelerden en önemlisi aynı bellek adresi tarafından desteklenen iki ayrı önbellek kullanmasıdır. Biri komutları, biri verileri tutar. **Önbellekte işlem** yapılırken Harvard mimarisinin, **asıl bellekte işlem** yapılırken Von Neumann mimarisinin uygulandıđını söyleyebiliriz. Günümüzde kullanılan yeni bilgisayar teknolojilerinde buna benzer mimariler sıkça kullanılmaktadır fakat isimlendirilmesinde ne Von Neumann ne de Harvard demek doğru deđildir. Ayrıca komutları, okunabilir verilermiř gibi göstermek de deđiştirilmiş Harvard mimarisine örnek gösterilebilir.

Kaynak: <https://systemprogramminglearningtutorial.blogspot.com/2016/11/harvard-von-neumann-mimarileri.html>