

**T.C.**  
**GAZİ ÜNİVERSİTESİ**  
**MÜHENDİSLİK FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

HAZIRLAYAN  
**141180018**  
**Kürşat ÇAKAL**

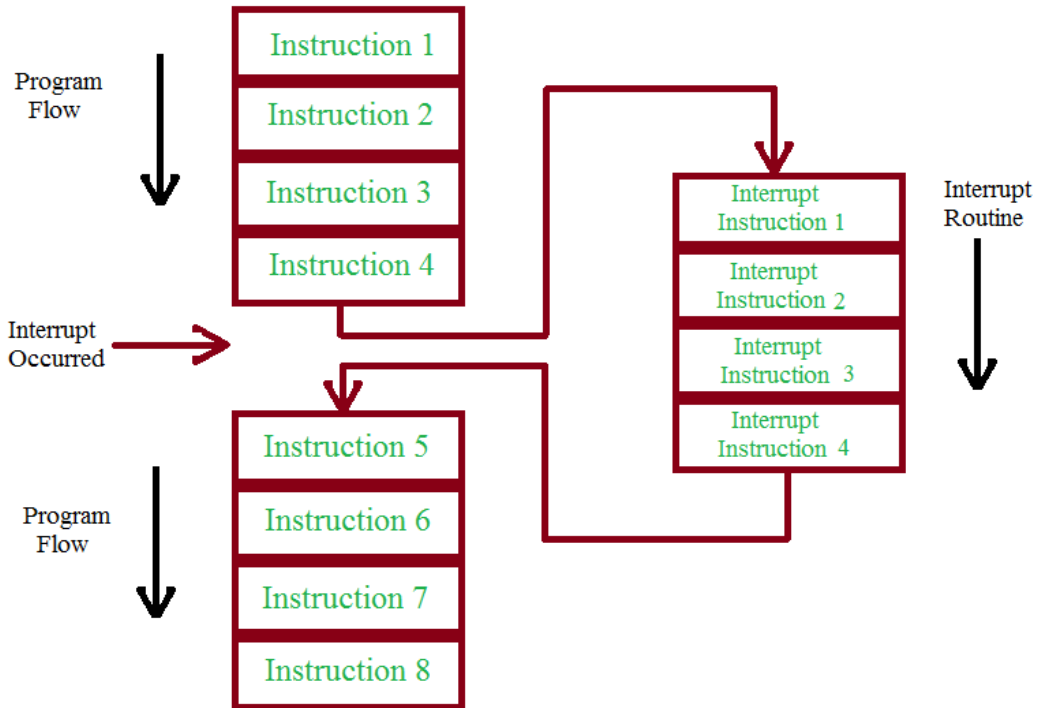
DERS  
**Bilgisayar Mimarisi**  
**BM311**

ARAŞTIRMA KONUSU  
**Interrupt çeşitleri**  
**Interrupt kullanım amaçları**  
**Farklı mimarilerinde interrupt performansları**

ANKARA 2017

## Interrupt Nedir ?

Interrupt en genel anlamıyla işlemcimizde anlık olarak işlenmekte olan olayın her hangi bir anında kesilmesi ve başka bir iş yapılmasına denir. İşlem kesildikten sonra alt işlem bitirilene kadar üst işlem tamamlanmaz alt işlem tamamlandıktan sonra üst işlem parçacığına devam edilir. Dikkat edilmesi gereken nokta bence hangi olayların kesme olup hangilerinin kesme olmadığıdır. Eğer bir işlemin normal akışı içerisindeki olaylar başka bir olayı engelliyorsa bu her zaman interrupt olmaz.



Örnek olarak bizim 10 farklı işlemi içinde barındıran sıralı adımlardan oluşan bir fonksiyonumuz var ve 7. Aşamasında 10 adım sonucunda oluşacak sonuç bir anlığına duruyor ve alakasız bir işlem yapılıyor bu durum interrupt değildir. Çünkü fonksiyonumuz bu adımı içermekte ve olumsuz da olsa 7. Aşamanın olması beklenmektedir. Fakat bu 10 adımlık fonksiyonumuz boyunca internet bağlantısının kopması ve internet bağlantısı yeniden sağlandığında işlemin kaldığı yerden devam etmesi bir interrupt olarak tanımlanabilir çünkü internet bağlantısının kopması beklenen bir durum değildir. Aynı trafikte kırmızı ışık yanarken durmamız beklenen bir durum fakat yeşil ışık yanıyorken önümüzden ambulans geçmesi sonucu durmamız işlemi interrupt olduğu gibi.

## **Interrupt'un Kullanılma Amacı Nedir ?**

Interrupt sadece bir işlemi durdurup onun durduğu andan itibaren başka bir işlemin çalışması demek değildir. Yani biz bir fonksiyon içinde olayın ana amacının dışında başka bir işlem yapmak için illaki bu akışı kesip durdurmamıza gerek yok işlem devam ederken arada başka bir işlemi başlatabiliriz. Eğer kesme yapılarak başlatılan işlemin sonucu ana olayın devam etmesi için gerekli işleri engellemiyorsa bu şekilde çoklu işlem yapabiliriz. Örnek verecek olursak bir yazılım geliştirme sürecinde asenkron olarak oluşturulan methodlar kullanarak aslında temelde bir interrupt çalıştırmış oluyoruz. Kısaca başlıklandırarak olursak interrupt aşağıdaki amaçlar için kullanılır.

- Birçok işlemi aynı anda gerçekleştirebilmek.
- İşlem bekletmekten kaçınmak.
- Senkronizasyon gerektiren uygulama ve sistemlerin tasarlamak.

Bu kısım için kaynak 3 yardımıyla öğrenilmiş ve çıkarım yapılmıştır.

## **Interrupt Çeşitleri Nelerdir ?**

Gömülü sistemler,elektrik-elektronik ve mikroişlemci başta olmak üzere farklı alanlara göre interrupt çeşitleri değişken başlıklar altında çeşitlendirilmektedir. Kaynaklarımızdan 1. Kaynağa göre interrupt 3 çeşit olarak ayrılmaktadır. Kaynaklarımızdan 2'nciye göre ise interrupt 2 çeşit olarak ayrılmaktadır. Araştırma sonucunda anlaşılan şudur ki aslında hepsi için doğru olduğunu söylemek mümkün. Her alan kendisiyle yakından ilgili kısımları başlıklandırmıştır. Interruptlar gömülü donanımlar, yazılımlar ve harici donanımlar olmak üzere işleyen bir sürecin nereden gelen etkiyle kesmeye uğradığına göre ayrılırlar. Üç farklı başlıkta bunları inceleyeceğiz.

- 1) Internal Interrupt
- 2) Software Interrupt
- 3) External Interrupt

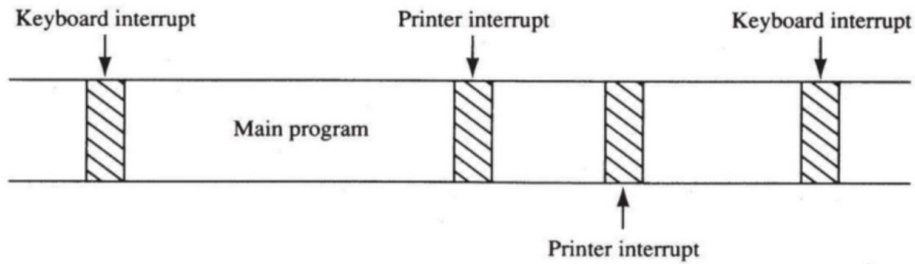
## 1-) Internal Interrupt Nedir?

Internal kesmeler sistemdeki harici olmayan yani direk sistemin içindeki donanımsal yapı üzerindeki bileşenler içinde olur. Donanım üzerindeki program veya işlemin yürütülmesi esnasında yine donanım üzerinden gelen bir komut veya işlem aracılığı sayesinde kesme meydana gelir. Bu tip kesmeler genellikle kullanıcı tarafından gerçekleştirilmezler yani kullanıcı erişimine kapalıdır. Donanım üzerindeki mimariye göre çalıştığı için doğal bir şekilde gerçekleşir.

Bir işletim sistemi görevden göreve geçtiğinde bir kesme olur. İşletim sisteminin belirli bir zamanda bir işi gerçekleştirebildiğini Hyper-Threading konusunda öğrenmiştik. İşletim sistemi aslında ayrıntılı bir sistem kullanarak, kullanıcı için uygun olan yollarla daha fazla görev gerçekleştirmek için fazlasını yapıyor gibi görünür. Dahili kesme işlemi çevre birimlerinin veya yazılımsal bir komut sürecinin etkisi olmadan gerçekleşir. Kaynak 6 yardımıyla öğrenilmiştir.

## 2-) External Interrupt Nedir?

External kesmeler sistemi o sisteme dahil olmayan dış donanım araçlarıyla kullanarak sistem üzerinde oluşturulan kesme türleridir. Buradaki anlatımda sistemden kasıt bilgisayar, uçak, tank, drone ve gemi gibi gömülü donanımlarının yanı sıra harici donanımlar bağlanarak sisteme etki bırakılan araçlar olabilir. Yani çevre birimleri tarafından sisteme kullanıcı etkisi sonucu bırakılan kesme türüdür. Kaynak 7 yardımıyla öğrenilmiştir.



Klavye ve fare araçlarıyla sebep olduğumuz işlemlerin bir çoğu, yazıcı ile yaptığımız işlemler ve oyun konsolu ile bilgisayarda oyun oynamamız gibi işlemleri bu kesme türüne örnek olarak verebiliriz. Fakat external yani harici kesme türlerini sadece kullanıcıların çevre birimlerini kullanarak sebep olduğu işlemler ile sınırlayamayız. Programlanmış bir çevre birimi içerdiği program algoritmasına göre bilgisayarın işlemcisinden bir işlem isteyebilir bu durumda sistemin daha önceden bu ana değin süre gelen yaptığı işlemlere uygulanan işlem de kesme olarak nitelendirilebilir.

Karmaşıklığı CISC mimarisine göre daha sadedir. CISC mimarisinin aksine daha basit bir yaklaşımla geliştirilmiş ve belleğe sadece “Load” ve “Store” komutlarıyla erişim sağlanmaktadır. Komut kümesi daha küçüktür. Microişlemcinin interrupt

düzenine göre yapılan her iş parçacığının ardından interrupt'a sebebiyet veren bir durum olup olmadığı kontrol edilir. Ve tüm bu özellikleriyle CISC mimarisine göre daha hızlı bir interrupt performansına sahiptir.

### EPIC Mimarisinde Interrupt Performansı

Bu mimari tipinde RISC ve CISC mimarilerinin üstün özellikleri bir arada bulunmuştur. Programdaki paralelliği tanımlayarak hangi işlemin bir başkasından bağımsız olduğunu belirler ve donanıma bildirir. Tahmin kullanma , çevrim başına birden çok komut çalıştırabilme , bellek gecikmesini önleme ve daha bir çok özelliğiyle mikroişlemci yapısının interrupt performansı diğer 2 mimariye göre çok daha hızlıdır çünkü paralel işlem yapabilme kabiliyetine sahip olması iş parçacıklarına kesme yapıp başka iş parçacıklarının işlenmesine olanak sağlar. Kaynak 4 yardımıyla öğrenilmiştir.

### **SONUÇ**

Interrupt'un ne olduğu, kullanıma amacı , farklı çeşitlerinin nasıl çalıştığı ve nasıl tepkiler gösterdiği , farklı mimariler üzerindeki performans değerleri araştırılarak öğrenilmiş bu başlıklara dair detaylı bilgiler edinilmiştir. Buna ek olarak farklı microişlemci mimarilerinin çalışma prensipleri somut bir terim üzerinden daha iyi anlaşılmıştır. Interrupt tek bir işlemin merkezde olması ve sıralı işlemler mantığıyla ilerlenmesinin getirdiği problemleri ortadan kaldırarak en yavaş çalışan mimari yapıda bile daha fonksiyonel ve performanslı bir yapı sunmuştur.

## KAYNAKLAR

- [1] <http://ecomputernotes.com/fundamental/input-output-and-memory/what-is-interrupt-types-of-interrupts>
- [2] [https://www.tutorialspoint.com/microprocessor/microprocessor\\_8086\\_interrupts.htm](https://www.tutorialspoint.com/microprocessor/microprocessor_8086_interrupts.htm)
- [3] <http://csit-on-move.blogspot.com/2013/07/why-interrupt-is-required.html>
- [4] <https://www.yumpu.com/tr/document/view/19404060/ymt216-mikroislemciler-ve-programlama/29>
- [5] <http://www.mademir.com/2010/09/interrupt-cesitleri-kullanlma-alarlar.html>
- [6] <https://www.techopedia.com/definition/7772/internal-interrupt>
- [7] <https://www.techopedia.com/definition/7772/external-interrupt>
- [8] <https://www.techopedia.com/definition/7772/software-interrupt>