

Alt Seviye Programlama BLM2021

Öğr. Grv. Furkan ÇAKMAK



Ders Tanıtım Formu ve Konular

BLM2021
Alt Seviye
Programlama

Hafta 13

Hafta	Tarih	Konular
1	08.10.2020	Alt seviye dilinin özellikleri, sayı ve kodlama sistemleri, 80x86 ailesi işlemcileri, yazmaçları ve bayrakları ile kesim organizasyonu
2	15.10.2020	Komutlar (veri aktarımı, aritmetik ve dallanma)
3	22.10.2020	Komutlar (çevrim, bayraklar, mantıksal, öteleme, döndürme)
4	29.10.2020	Cumhuriyet Bayramı
5	05.11.2020	Komutlar (katar işlemleri, ön ekler)
6	12.11.2020	Adresleme modları, alt seviye programlama araçları, sözde komutlar
7	19.11.2020	Çalışma ortamının hazırlanması ve debug kullanımı
8	26.11.2020	EXE tipinde alt seviye programlama
9	03.12.2020	COM tipinde alt seviye programlama
10	10.12.2020	Yordam ve makro kullanımları
11	17.12.2020	Alt-programlar ve parametre aktarma yöntemleri
12	24.12.2020	Ortak kesim kullanımı ve EXTRN/PUBLIC tanımlamaları
13	31.12.2020	Kesme, vektör tablosu
14	07.01.2020	Alt seviye programlama dilinin yüksek seviyeli diller ile birlikte kullanılması

Kesme (Interrupt) Nedir?

- Çeşitli kriterler ve önceliklerle program işleyişlerinin kesintiye uğratılarak eş zamanlı çalışmanın sağlanması interrupt mekanizması ile sağlanır.
- Yavaş birimler ile hızlı birimler arasındaki bağlantıyı sağlarken zaman kaybetmeyi engellemek için kullanılır.
- Çevre birimleri işlemciden daha yavaş çalışır.
 - Mouse - Klavye
 - Yazıcı
 - Monitör
- Diğer birimler
 - RAM

İşlemci ile Yavaş Birimler Arasındaki İlişki

BLM2021
Alt Seviye
Programlama

Hafta 13

- Busy waiting
- Polling
- Interrupt
 - Zaman kesmesi
 - Grafik kesmesi



Kesme Çeşitleri

BLM2021
Alt Seviye
Programlama

Hafta 13

- Temel kesmeler
 - Donanım Kesmeleri
 - Yazılım Kesmeleri
- Interrupt Handler
- Dahili kesmeler
 - Logical interrupt
 - Sıfıra bölme
 - Adım adım çalıştırma
 - Vb.
- Harici kesmeler
 - Maskelenemez Kesmeler (NMI)
 - Maskelenebilir Kesmeler (INTR)
 - IF durumu önemli !
 - 8259A bütünleşik entegresi üzerinden sürdürülür.



Kesme Öncelikleri

BLM2021
Alt Seviye
Programlama

Hafta 13

Öncelik	Kesme	Açıklama
1	<i>Divide by Zero</i>	<i>Hatana neden olan program sonlandırılır. Sistem güvenliğini yitirir. Tekrar başlatılması gerekebilir.</i>
2	<i>INT #</i>	<i>Yazılım olarak numarası verilen kesmenin çağırılması</i>
3	<i>INTO</i>	<i>Aritmetik işlemin (MUL/IMUL) sonucunu değerlendirmek içindir. JO/JNO koşullu dallanma komutları da bu amaçla kullanılır.</i>
4	<i>NMI</i>	<i>INT 02H – Bellek üzerinde oluşan kritik hatalarda kullanılır.</i>
5	<i>INTR</i>	<i>8259A entegresi üzerinde oluşan donanım kesmeleridir.</i>
6	<i>Single step</i>	<i>INT 01H programın adım adım çalışmasını sağlayan kesmedir.</i>

Kesme Oluştuğunda Yapılan İşlemler

1. Bayraklar yığında saklanır (PUSHF)
 2. TF = 0
 3. IF = 0 (CLI)
 4. CS yazmacı yığına atılır (PUSH CS)
 5. IP yazmacı yığına atılır (PUSH IP)
 6. INT çalıştırılır (# x 4 sonucunda vektör tablosundaki adrese erişilir.)
 - Alınan ilk word değeri IP yazmacına,
 - İkinci word değeri CS yazmacına yerleştirilir.
- Kesme servis programını yazan;
 - Yazmaçlarla ilgili durumları değerlendirmeli,
 - IRET komutu ile kesmeden dönmelidir.

Vektör Tablosunun Görevi ve Konumu

- Kesme servis programları gerek BIOS gerekse İşletim Sistemi tarafından sağlanan değişik uzunluktaki kod parçalarıdır.
- Bilgisayar açılışında belleğe yerleştirilirler.
- Farklı bellek alanlarında bulunan bu kodlara erişmek için Vektör Tablosu kullanılır.
- Vektör Tablosunun tek görevi kesmelerin başlangıç adreslerini tutmaktır.
- 00000H - 003FFH fiziksel adresleri arasında bulunan 1024 byte'lık alanda bulunur.
- Her servisin adresi 2 word'le ifade edilir (Offset ve kesim adresi).
- Yani $1024 / 4 = 256$ tane kesme vardır.
- Kullanıcılar kesme yazacaklarsa;
 - INT 60H - INT 67H arası bu iş için ayrılmıştır.
 - Mevcut olan servisin yerine yeni bir kesme yazılabilir.

Vektör Tablosunda Değişiklik Nasıl Yapılır?

- Programcı kendi kesmesini yazabilir.
- Var olan bir kesmeden hemen önce devreye girebilecek bir kesme yazabilir.
- Her iki durumda da yazılan kesme vektör tablosuna yerleştirilmelidir.
- Bunun için yapılması gereken işlemler;
 1. Kesmenin TSR ile belleğe yerleştirilmiş olması gerekir. (TSR -> Terminate and Stay Resident)
 2. Yazılan kodun adresinin (offset + kesim) Vektör Tablosunda ilgili alanlara yazılması gerekir (Kesmeler disable edilmelidir. Edilmezse ?)
 3. Değiştirilecek kesmenin adresi saklanmalıdır.
 4. INT 21H - Fonksiyon 15H (Set Interrupt Vector) kullanılarak yazılan kesme kodunun adresi vektör tablosuna yazılmalıdır.
 5. Kesmelere izin verilmelidir.
 1. Kendi kesmemizi yazmışsak IRET ile dönmeliyiz.
 2. Orijinal kesmeden önce çalışmışsak JMP ile orijinal kesme adresine gitmeli ve akışa oradan devam etmeliyiz.
 6. Kullanım tamamlandıktan sonra orijinal kesmenin adresi vektör tablosuna yazılmalıdır.

Interrupt Servis Programı Yazarken Nelere Dikkat Edilmelidir?

BLM2021
Alt Seviye
Programlama

Hafta 13

- Kesme kodu içerisinde girildiğinde kesmelere izin verilmelidir.
- Kesme içerisinde yazmaçların bir önceki değerlerinin korunması gerekmektedir.
- Kesme programı 1'den fazla kesim kullanacak ise bu değerler kesmeye gitmeden önce ilgili yazmaçlara yüklenmelidir.
- Donanım kesmesi yerine (BIOS) geçecek kesme yazılacaksa İşletim Sistemine ait (DOS) hiçbir fonksiyon kullanılmamalıdır.
- Kesme programı TSR ile belleğe yerleştirilmesi gerekmektedir.
- Yığına veri göndermişsek onları kesme içerisinde temizleyecek şekilde hareket sergilemeliyiz.

Kesmeleri Kullanırken

- 00H-1FH arasında BIOS kesmeleri,
- 20H-2FH arasında DOS kesmeleri bulunur.
- Kesmeler çağırılırken,
 - Fonksiyon numarası AH'a,
 - Eğer varsa alt fonksiyon numarası AL'ye aktarılır.
 - INT komutu kullanılır.
- Örnek;
 - MOV AH, 1H
 - INT 21H

Sabırla Dinlediğiniz İçin Teşekkürler

BLM2021
Alt Seviye
Programlama

Hafta 13

