

T.C.
GAZİ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

HAZIRLAYAN

141180018

Kürşat ÇAKAL

DERS

Bilgisayar Mimarisi

BM311

ARAŞTIRMA KONUSU

Intel i7 Mikroişlemcilerin Komut Yapıları ve Komut Kümeleri

ANKARA 2017

Mikroişlemcide Komut Yapıları ve Komut Kümeleri

Öncelikle mikroişlemcilerde komutlara kısaca değinmemizin yazımın devamı için yararlı olacağını düşünüyorum. Mikro işlemciler bellekte saklanan belli bit dizilerine göre işlemler yapmak için tasarlanmışlardır. Bu bit dizilerine komut denir. Mikroişlemciler daha üretim aşamasında iken yapacağı işlemler tasarlanarak bu sisteme eklenir. Ve devamında dahil edileceği sistem bu alt işlemlerin birleştirilmesiyle çıktılar üretebilmektedir. Mikroişlemcilerde komut kümelerinin farklılık göstermesinin bir çok sebebi vardır. En önemli sebep olarak mikro işlemci üreticilerinin ve mimarilerinin farklı olması her işlemci ailesinin yeni komutlara sahip olmasıdır. Bunun sonucu olarak da mikro işlemlerinin ortak olarak okuyup işlem yapabilecekleri komut yapıları ve komut kümelerinin olması çok mümkün olmamaktadır. Hatta bu yazımda değerlendireceğimiz gibi aynı aileye sahip mikro işlemcilerin bile komut yapılarında ve komut kümelerinde önemli farklılıklar gözlenebilmektedir. Genel olarak mikro işlemcilerin çalışmasını sağlayan komutları 3 farklı başlıkta inceleyecek ve bu başlıklar altında değerlendirmemizi yapacağız. Kaynak 2 ve kaynak 3 yardımıyla bu bilgiler yorumlanmıştır.

Aritmetik komutlar

Aritmetik komutlar bellekte saklanan veri üzerinde çeşitli aritmetik ve mantıksal işlemler yapılmasını sağlarlar (toplama, çıkarma, mantıksal ve işlemi, sağa ve sola bit kaydırma işlemleri gibi).

Girdi/Çıktı komutları

Girdi/çıktı komutları işlemcinin dış birimlerle bağlantısını sağlayan kapılarını (port) kontrol eder.

Kontrol komutları

Kontrol komutları ise mikro işlemcinin iç durumunu değiştiren komutlardır (programın devam edeceği yeri değiştirme, aritmetik işlemlerde yardımcı olan kütükleri değiştirme, 'stack pointer' işlemleri gibi). Her mikro işlemcinin komut kümesinde bu üç türden komutlar mutlaka bulunacak, ancak komutların hangi bit dizilerine karşılık geldikleri, işlemlerin nasıl yürütüldüğü farklı olacaktır. Kaynak 3 yardımıyla bu bilgiler yorumlanmıştır. Bu bilgilere ek olarak mikroişlemcilerle ilgili değinmemiz gereken küçük farklılıklar vardır.

C.P.U.=Central Processing Unit

M.P.U=Micro Processing Unit

M.C.U.=Micro Controller Unit

Merkezi işlem birimi bir mikro işlemcidir. Bazı mikro işlemciler bütün paketler halindedirler. CPU çevre birimleri ile çalıştığı gibi mikro işlemcilerde çevre birimlerine ihtiyaç duymaktadırlar. Bu sebeple araştırma ödevinde bahsi geçen mikro işlemci ve intel firmasının üretmiş olduğu işlemciler ve mikro işlemciler instruction set architecture ibaresi ile değerlendirilecektir. Bilgisayarlarımızda Central Processing Unit bir mikro işlemcidir.

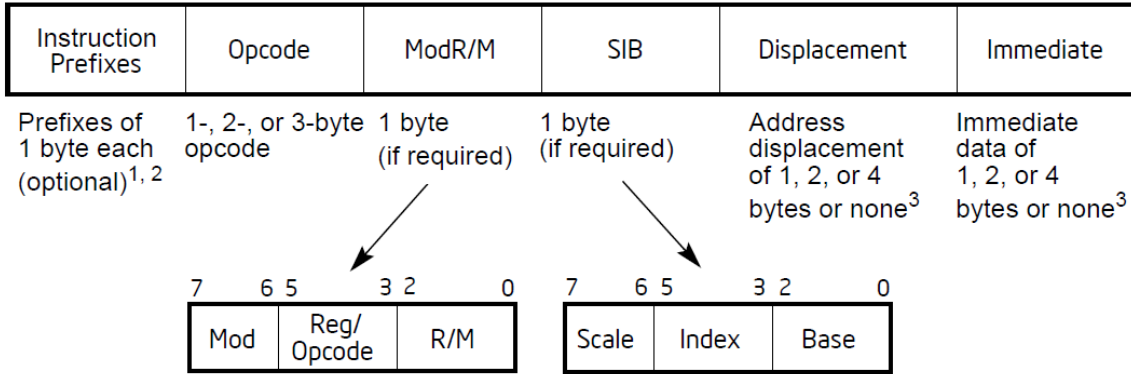
Farklı işlemler için farklı komut sistemleri mevcuttur bu komut yapılarını yazılımcılar işlemciye giden işlemlerde kullanıldığında işlemcinin performansını düzenlemeye yönelik çalışmalar yapmaktadırlar. Her bir komut seti değişken amaçlar için kullanılıyor ve kullanıldığı alanda fark yaratacak bir performans sergiliyor ve işlemcilerde standart olarak marka sınıflandırması yapılmadan bulunuyor. Markalar ortak komut setleri oluşturabiliyor yani sadece farklı adlar ile oluşturulup kullanım amacı aynı oluyor. Performans farkları da komut setlerinin işlemcilerde nasıl kullanıldığına göre ortaya çıkıyor.

Bir ISA (Instruction set architecture), opcode dizisinin bir özelliklerini ve belirli bir işlemcinin uyguladığı yerel komutları içerir. Bir komut kümesi mimarisi, belirli bir işlemcide kullanılan bir dizi tasarım tekniği olan mikro mimariden farklıdır. Farklı mikro mimarilere sahip işlemciler ortak bir yönerge seti paylaşabilir. Bu bölümdeki bilgiler kaynak 5 yardımıyla öğrenilmiş ve yorumlanmıştır. Çoğu komut kümesine ortak işlemlerin örnekleri şunlardır:

- Bir kayıt bir sabit değere atanır.
- Veriler konumundan bir kayıt defterine kopyalanır veya bu işlemin tersi yapılır.
- Bir kaydın içeriğini, genellikle bir yükleme ve depolama işlemi olarak adlandırılan bir hesaplamanın sonucunu depolamak için kullanılır.
- Donanım aygıtlarından veri okumak ve yazmak, iki kayıt değerini eklemek, çıkartmak, çarpım veya bölmek, sonucu bir kayıt defterine yerleştirmek, büyük olasılıkla bir durum kaydında bir veya daha fazla koşul kodu ayarlamak için kullanılır.

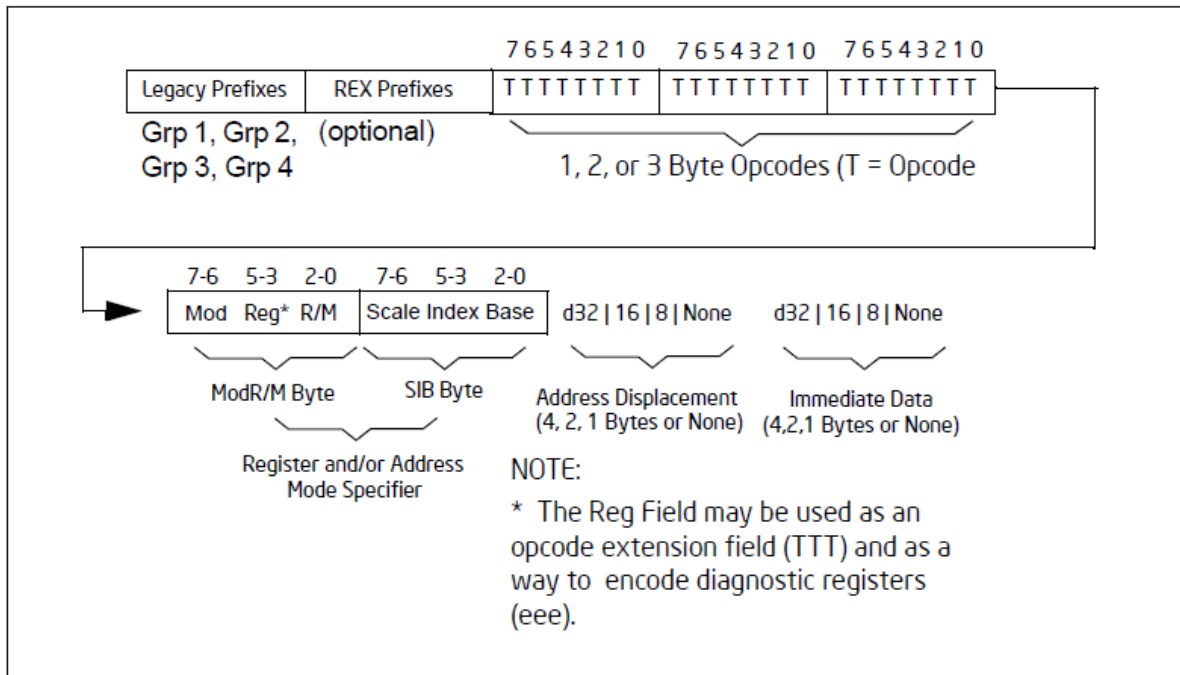
Intel i7 komut kümesi mimarisini incelemekten önce geçmişten günümüze bu alandaki değişikliklerin genel teması hakkında konuşmakta fayda var. Tasarlanmış olan her mikro işlemci yapısının farklı genel amaçları ve mimarisi vardı. Bu geçişlerin her biri yeni ve değiştirilmiş talimatlara ihtiyaç duymaktaydı. İhtiyaçlar sonucunda sanallaştırma, yeni SIMD yönerge setleri, AES şifreleme hızlandırması ve daha bir çok özellik yeni yapıları desteklemek için son on yılda yeni talimatlar eklendi. Bu konuda kaynak 6 yardımıyla Intel geliştirmiş olduğu bütün mikro işlemci teknolojilerinin detayını ulaşabilirsiniz.

Intel i7 Mikroişlemci Komut Yapıları



Şekil 1

Intel 64 ve IA mimarileri komut kodlamaları Şekil 2-1'de gösterilen formatın alt kümeleridir. Talimatlar isteğe bağlı komut önekleri (herhangi bir sırada), birincil opcode baytları 3 btye değerine kadar olmak sureti ile, birincil opcode baytları ModR / M baytı ve bazen SIB (Ölçek-Dizin-Tabanı) içeren adresleme-biçim belirleyici (gerekirse) bayt, yer değiştirme (gerekirse) ve immediate bir veri alanı (gerekirse) parçalarından oluşmaktadır.



İşlemcilerdeki bu komut desteklerindeki asıl amaç, birbiri ile aynı olan işlemleri yaparken izlenmesi gereken yolun işlemciye tanıtılarak performans artışı sağlamaktır. Örnek uygulama olarak, sinyal işleme, grafik işleme verilebilir. Tipik olarak tekrarlanan işlemleri işlemciye hazır bir komut olarak vererek işlemcinin performansını artırmayı hedefler. Bu kısımdaki bilgiler 7 numaralı kaynak yardımıyla elde edilmiş ve yorumlanmıştır.

Legacy Prefixes

Yeni talimat uzantılarında F2H, F3H ve 66H haricinde bunlar isteğe bağlıdır. Eski ön ekler REX ön eklerinden önce yerleştirilmelidir.

REX Prefixes

REX örnekleri opcode haritasının bir satırına yayılan ve 40H - 4FH girişlerini kaplayan 16 opcode dizisidir.

Opcode

Bir talimat için birincil opcode, talimatın bir ila üç bayt arasında kodlanır. Birincil opcode, daha küçük kodlama alanları tanımlanabilir. Bu alanlar çalışma sınıfına göre değişir seslendirdi.

Bu kısımdaki bilgiler 7 numaralı kaynak yardımıyla elde edilmiş ve yorumlanmıştır.

Table B-1. Special Fields Within Instruction Encodings

| Field Name | Description | Number of Bits |
|------------|---|----------------|
| reg | General-register specifier (see Table B-4 or B-5). | 3 |
| w | Specifies if data is byte or full-sized, where full-sized is 16 or 32 bits (see Table B-6). | 1 |
| s | Specifies sign extension of an immediate field (see Table B-7). | 1 |
| sreg2 | Segment register specifier for CS, SS, DS, ES (see Table B-8). | 2 |
| sreg3 | Segment register specifier for CS, SS, DS, ES, FS, GS (see Table B-8). | 3 |
| eee | Specifies a special-purpose (control or debug) register (see Table B-9). | 3 |
| tttn | For conditional instructions, specifies a condition asserted or negated (see Table B-12). | 4 |
| d | Specifies direction of data operation (see Table B-11). | 1 |

KAYNAKLAR

- [1] <https://ark.intel.com/products/series/95544/7th-Generation-Intel-Core-i7-Processors>
- [2] https://en.0wikipedia.org/wiki/Instruction_set_architecture
- [3] https://tr.0wikipedia.org/wiki/Mikroişlemci_komutları
- [4] <https://en.0wikipedia.org/wiki/Intel>
- [5] www.sdsc.edu/~allans/cs141/L2.ISA.pdf
- [6] https://en.0wikipedia.org/wiki/List_of_Intel_Core_i7_microprocessors
- [7] <https://www.intel.com/content/dam/www/public/us/en/documents/manuals/64-ia-32-architectures-software-developer-instruction-set-reference-manual-325383.pdf>