Bu üç seviye arasındaki ayrım çok belirsiz olabilir. Örneğin, çoğu uyumlu dil donanımı doğrudan değiştirmenize izin verir. Aynı şekilde, iyi stoklanmış bir DSP fonksiyonları kütüphanesine sahip yüksek seviyeli bir dil, bir uygulama paketi olmaya çok yakındır. Bu üç kategorinin amacı, neyi manipüle ettiğinizi anlamaktır: (1) donanım, (2) soyut değişkenler veya (3) tüm prosedürler ve algoritmalar.

Daha yüksek bir seviyede çalışırken, son makine kodunun bellek kullanımı, hız ve hassasiyet açısından daha az verimli olmasını bekleyin.

Hangi programlama dilini kullanacağınız kim olduğunuza ve ne yapmayı planladığınıza bağlıdır. Çoğu bilgisayar bilimcisi ve programcısı C’yi kullanır. Güç, esneklik ve modülerliğe sahiptir. C o kadar popüler ki, fakat DSP uygulamasında C dışında bir şey programlanıyor çünkü ilk olarak, DSP o kadar hızlı büyümüştür ki, değişmesi oldukça yavaş olan askeri ve devlet kurumları FORTRAN ve PASCAL gibi dillerin modunda takılıp kalmıştır. İkinci olarak, bazı uygulamalar yalnızca montaj programlamasıyla elde edilebilen en yüksek verimliliği gerektirir.Üçüncüsü, C, özellikle yarı zamanlı programcılar için öğrenmesi kolay bir dil değildir. Bu nedenle DSP tekniklerine ihtiyaç duyanlar basitliğinden dolayı genellikle BASIC'e yönelir.

Genel bir kural olarak, montajda yazılan bir alt yordamın, karşılaştırılabilir üst düzey programdan 1.5 ila 3.0 kat daha hızlı olmasını bekleyin. Kesin değeri bilmenin tek yolu, kodu yazmak ve hız testleri yapmaktır. Kişisel bilgisayarların hızı her yıl yaklaşık % 40 arttığından, montajda bir rutin yazmak, donanım teknolojisinde yaklaşık iki yıllık bir sıçramaya eşdeğerdir.

Çoğu profesyonel programcı, derleme kullanma fikrine oldukça kırılır ve BASIC'i önerirseniz şaka yapar. Mantıkları oldukça basit: montaj ve BASIC, iyi yazılım uygulamalarının kullanılmasını engelliyor. İyi kod taşınabilir, modüler ve anlaşılması kolay olmalıdır. Zayıf montaj yapısı ve BASIC, bu standartlara ulaşmayı zorlaştırır.

Diyelim ki C'de bir program yazıyorsunuz ve rakibiniz aynı programı assembly'de yazıyor. Son kullanıcının ilk izlenimi, programınızın iki kat daha yavaş olduğu için önemsiz olduğu olacaktır. Hiç kimse derlemede büyük programlar yazmanızı önermez, sadece programın hızlı yürütülmesi gereken kısımları önerir. Örneğin, DSP yazılım kütüphanelerindeki pek çok işlev montajda yazılır ve daha sonra C'de yazılmış daha büyük programlardan erişilir.

**Yürütme hızı : Donanım**

Hesaplama gücü o kadar hızlı artıyor ki, konuyla ilgili herhangi bir kitap yayınlanmadan eskimiş olacak. Orijinal IBM PC, 4.77 MHz saat hızı ve 8 bit veri yoluna sahip 8088 mikroişlemciye dayanan 1981'de tanıtıldı. Bunu her 3-4 yılda bir tanıtılan yeni nesil kişisel bilgisayarlar izledi.Bu yeni sistemlerin her biri, bilgi işlem hızını önceki teknolojiye göre yaklaşık beş kat artırdı. 1996'da saat hızı 200 MHz'e ve veri yolu 32 bite yükseldi. Diğer iyileştirmelerle, bu, yalnızca 15 yılda bilgi işlem gücünde yaklaşık binlik bir artışla sonuçlandı.

Hızla değişen bu alanda güncel bilgileri elde etmenin tek yolu doğrudan üreticilerden: reklamlar, şartname sayfaları, fiyat listeleri vb. Performans verileri için dergilere ve günlük gazetenize bakın. Bilgisayar gücünün mevcut durumunu öğrenmek basitçe yeterli değildir ; nasıl geliştiğini anlamanız ve izlemeniz gerekir.

Yürütme hızının bilgisayar donanımı tarafından nasıl sınırlandırıldığına bakalım. Bilgisayarlar birçok alt sistemden oluştuğundan, belirli bir görevi yürütmek için gereken süre iki ana faktöre bağlı olacaktır: (1) bireysel alt sistemlerin hızı ve (2) bu bloklar arasında veri aktarımı için geçen süre.Merkezi İşlem Birimi (CPU) sistemin kalbidir. Daha önce açıklandığı gibi, her biri 32 bit tutabilen bir düzine kadar kayıttan oluşur.

Daha karmaşık matematik, verilerin matematik yardımcı işlemcisi olarak adlandırılan ALU, özel bir donanım devresine aktarılmasıyla ele alınır. ALU, CPU ile aynı çipte bulunabilir veya ayrı bir elektronik cihaz olabilir. Örneğin, iki kayan nokta sayısının eklenmesi, CPU'nun ALU’ya 8 baytı (her sayı için 4) ve verilerle ne yapılacağını açıklayan birkaç baytı aktarmasını gerektirecektir. Kısa bir hesaplama süresinden sonra, matematik işlemcisi, toplam olan kayan nokta numarasını içeren dört baytı CPU'ya geri gönderir.

En ucuz bilgisayar sistemlerinin bir matematik işlemcisi yoktur veya bunu yalnızca bir seçenek olarak sunar. Bu düşük performanslı sistemler, donanımı yazılımla değiştirir. Matematiksel işlevlerin her biri, doğrudan CPU içinde ele alınabilen temel ikili işlemlere bölünmüştür. Bu aynı sonucu sağlarken, yürütme süresi çok daha yavaştır, örneğin 10 ila 20 kat.

Çoğu kişisel bilgisayar yazılımı, bir matematik işlemcisi ile veya olmadan kullanılabilir. Bu, derleyicinin her iki durumu da ele almak için makine kodu üretmesini sağlayarak gerçekleştirilir, bunların tümü son çalıştırılabilir programda depolanır. Sabit nokta değişkenleri (tamsayılar) içeren hesaplamalar, bir matematik işlemcisinin varlığından etkilenmez, çünkü bunlar CPU içinde işlenirken, DSP'nin ve kayan nokta hesaplamalarını kullanan diğer hesaplama programlarının yürütme hızı bir büyüklük sırası olabilir.

Çoğu bilgisayar sisteminde CPU ve ana bellek ayrı yongalarda bulunur. Açık nedenlerden dolayı, ana belleğin çok büyük ve çok hızlı olmasını istersiniz. Maalesef bu, belleği çok pahalı hale getirir. Ana bellek ile CPU arasındaki veri aktarımı, hız için çok yaygın bir darboğazdır. Bu sorunu aşmanın yaygın bir tekniği, bir bellek önbelleği kullanmaktır.