

Küçük Adam Bilgisayar - CPU simülatörü - yardım sayfası

Little Man Computer (LMC) 1965 yılında Dr. Stuart Madnick tarafından oluşturulan bir bilgisayar öğretim modelidir ve tüm temel özelliklere sahiptir.

Orijinal konsept CPU içinde bir listede kodlanmış komutları yerine getiren küçük bir adamdı.

CPU içinde şunlar vardır:

- 100 Bellek yerleri (00 dan 99 a kadar numaralı).
- Bir Aritmetik Birimi - hesaplamaları yapmak için.
- Bir Akümülatör - son işlem veya hesaplamının sonucunu saklamak için.
- Bir Program Sayacı - yapılacak bir sonraki kumandanın adresini saklamak için.
- Bir Talimat Kayıt Birimi - bellekten okunan kumandanın üst basamağını tutmak için.
- Adres Birimi - bellekten okunan kumandanın alt iki basamağını tutmak için kayıt.
- Veri Giriş Birimi - gereken numaraları girebilmek için.
- Veri Çıkış Birimi - Herhangi bir sayı yazmak için bir yazış alanı.
- Program Sayacı sadece 2 basamaklı rakamları (00-99) tutar ve Akümülatör 3 haneli rakamları ve bir de işareti (-999 dan +999) tutar. Çıkış alanı iki sütun veri gösterir ve dolunca eski bilgiler kaybolur.

Bir talimat (kumanda) bellekten okunduğu zaman üst hanesi (Op kod) ne yapmaya karar vermek için kullanılır. Bize daha kolay programlar yazmaya yardımcı olması için her talimata (kumandaya) fonksiyonla ilgili bir kod verilmiştir. ASSEMBLER (tercümeçi) bunların isimlerini gereken kodlara çevirir. Ardındaki iki basamak yeri de kumanda / talimatla ilişkili bir adres olarak kullanılır.

Kumandalar şunlardır:

KOD	İSİM	AÇIKLAMA
0	HLT	HALT - Dur (Küçük Adam dinlenecek).
1	ADD	Akümlatöre bellek adresinin içeriğini ekle.
2	SUB	SUBTRACT - Akümülatörden bellek adresi içeriğini çıkart.
3	STA veya STO	STORE ACC or STORE - Akümülatör içeriğini bellek adresine yükle.
4		Bu kod kullanılmaz ve bir hata verir.
5	LDA	LOAD ACC - Bellek adresinin içeriğini Akümülatöre yükle.
6	BRA	BRANCH ALWAYS - Verilen kumanda adresine git.

(Bir sonraki talimat adresi olarak verilen adresi kullan).

7	BRZ	BRANCH on ZERO - Akümülatör SIFIR ise verilen kumanda adresine git.
8	BRP	BRANCH on POSITIVE - Akümülatördeki numara pozitif ise verilen kumanda adresine git.
9	INP veya OUT	Giriş ya da çıkış. Adres 01 ise INPUT (giriş) ten klavyeyi oku. Adres 02 ise Akümülatörü OUTPUT (çıkış) a kopyalayın.
9	OTC	OUTPUT CHARACTER - Adres 22 ise akümülatörü bir karakter/harf olarak yaz.
	DAT	DATA - verileri içeren bir konum belirtmek için kullanılır.

LMC bilgisayarı bellek içeriklerinin talimat veya veri olduğunu bilmez. Modern bilgisayarlar bellek içeriklerini TALİMAT ve VERİ diye ikiye ayırabilirler. DAT bellek içeriklerinin VERİ olduğunu assembler'a (tercümana) anlatmak için yapılan sanal bir talimattır.

Bu uygulama çıkışa hem rakamların (OUT) hem de harflerin (OTC) yazılmasını mümkün kılar. Gerçek bir bilgisayarda sadece OTC talimatı olur ve program çıkış yazmadan önce gerekirse harfleri numaralar dönüştürür. Aynı şey giriş için de geçerlidir. Fakat kolay olsun diye NUMARA girişi sağlayan INP kumandası sağlanmıştır.

Çevirici (Assembler) ayrıca bellek adreslerine isim (etiket) vermemizi de sağlar. Herhangi bir kelime eğer bir talimat değilse, etiket olarak kabul edilir. Eğer ETİKET bir talimat öncesi veya DAT önünde ise bu oradaki adresi temsil eder. Eğer ETİKET talimat sonrasında veya DAT arkasında ise o zaman bellek adresi etiketin yerine ikame edilir. LMC solundaki iki alan Assembler (çevirici) işlevine sunulan program tasarımı ve çeviricinin bu programı sayılara dönüştürülmüş etiketlerini gösteren liste yazısıdır.

Ayrıca bu uygulamada programın içine yorum da koyabilirsiniz. // ile başlayan bir satır assembler tarafından göz ardı edilir. // programda her yerde kullanılabilir. Çevirici her satırda şunları arar: isteğe bağlı ETİKET, TALİMAT / KUMANDA kodu, ve gerekirse kod için lazım olan BELLEK ADRESİ. Bunun haricinde satırdaki herşey göz ardı edilir. Bu yüzden de talimatlar arkasına yorumlarınızı da koyabilirsiniz. (// kullanmak bu durumda iyi bir yöntemdir). Şuna dikkat edin: Gereken bir adresi atlarsanız çevirici normalde adres yerine 0 (sıfır) kullanır. Fakat gereken adresi atlatıp yorum eklerseniz bir HATA alırsınız.

Çevirici hem büyük hem de küçük harf kabul eder ve talimatlar için karışımı da kullanılabilir. Fakat ETİKETLER için büyük/küçük harflerin önemli olduğunu unutmayın. BAŞLA, başla ve Başla üç farklı etiketlerdir.

İşlemci değişik şekillerde programlanabilir: talimatları numara halinde bellek içine direkt yazarak, talimatların KOD isimlerini derleme alanına girerek, veya talimat kodlarını içeren bir dosya hazırlayarak ve onu doğrudan derleme alanına yükleyerek. Program derleme alanına yüklendikten sonra ÇEVİRİCİ assembly lisan kodlarını sayılara dönüştürür ve belleğe yükler. Programlamayı kolaylaştırmak için derleme alanında yapılan her değişiklikte çevirici otomatik olarak lisan kodlarını numaraya dönüştürür.

SELECT (Seçenek) düğmesi aşağıdaki programları seçmenizi sağlar:

add	TOPLAMA - iki sayının toplamını yaz. İki numara girişi ister.
add/subtr	TOPLA/ÇIKAR - üç rakam girin. İlk iki rakamın toplamını ve üçüncü eksi birinci rakam sonucunu yaz.
max	MAKSİMUM - iki rakam girin. Büyük olanı yaz.
is sum 1-n	1-n TOPLAMI - Girilen rakamın birden rakama kadar olan numaraların toplamı olup olmadığını yaz. Eğer rakam 1-n toplamı olansa rakamı yaz, değilse 0 (sıfır) yaz.

square	KARE - Girilen rakamın karesini yaz.
times table	ÇARPIM TABLOSU - Bir rakam girin. O rakama kadar gelişen çarpım tablosunu yaz.
bubble sort	BALON SIRALAMASI - 20 kadar rakam girin. Son rakam olarak 0 (sıfır) girin. Sıralama başlar ve rakamlar en ufaktan en büyüğe göre sıralanır.
overwrite	ÜZERİNE YAZMAK - Bellek üzerine yazma hem talimatlar hem de veri için kullanılan bellek çekincelerini göstermektedir. Program çalışırken, girilen numaralar bellekte 0 (sıfır) adresinden başlayarak depolanır. Bellek konumu 10 a ulaşıldığında program TALİMATLARI üzerlerine yazılarak VERİYE değiştirilir. LMC verileri bir talimat şeklinde yorumlamaya çalışırken çalışması bozulur ve durur.
ascii	ASCII Bilgi Değişimi İçin Amerikan Standart Kodu anlamına gelir ve bunlar harflerin bilgisayarlar tarafından manipüle edilebilmesini sağlayan kodlama yoludur. Orijinal kod yazdırılabilir karakterleri temsil etmek için 033-126 arası numaraları, kontrol fonksiyonları temsil etmek için de 000-031 ve 127 numaraları kullanır. 32 boşluk karakteri için kullanılır ve normal halde (yazılmasa bile) yazdırılabilir bir karakter olarak kabul edilir. LMC sadece 10 numarayı YENİSATIR kontrol fonksiyonu için kullanır. Bu program 32 den 126 ya kadar numaraları karakter olarak yazar.
ascii table	ASCII TABLOSU - program 32 ile 96 arasındaki kodları sırasıyla numara - boşluk - harf şeklinde yazar.

LMC simülatörünü kullanmaya başlamak için şunu öneririz:

- SELECT - açılan listeden bir program seç.
- RUN düğmesine basın. Program çalışmaya başlar.

Yürütme orta / yavaş hızda başlar ve bunu hızlandırmak için >> düğmesini veya yavaşlatmak için << düğmesini kullanabilirsiniz. En düşük beş hızlar yönerge yürütme akışını gösterir:

- FETCH AL - bir sonraki yürütülecek talimat içeriği alın. Aynı zamanda Program Sayacına 1 eklenir.
- EXECUTE YAP - yüklü talimatı çöz (deşifre et) ve istenen işlemi gerçekleştir.

Herhangi bir noktada STOP (dur) ve tekrar RUN (çalış) yapabilirsiniz. DUR yürütme duraklar; eğer yürütme akışı vizyonda değilse, geçerli talimat durmadan önce tamamlanır. Düşük hızlarda duraklama anında olur ve geçerli talimatı bitirmek için tekrar DUR tuşuna basabilirsiniz. (Anında iptal etmek isterseniz RESET kullanın.) STOP (dur) dan sonra, RUN (çalış) veya STEP (adım) seçebilirsiniz.

STEP (adım) düğmesi (yavaş hızda) program talimatlarını birer birer yapar. RESET (resetlemek) kumandası ile program başlangıcına dönlür. (LMC Program Sayacı = 0 olur). Veya da Program Sayacına herhangi bir bellek adresi girerek ayarlayabilirsiniz.

Programa girişler değişik alanlarda yapılır: Eğer program çalışıyorsa INPUT kutusuna numara yazılır. Program durmuşsa değişik bellek konumlarına, Program Sayacına, veya da sol taraftaki program yükleme alanına girişler veya ayarlamalar yapılabilir. Çeviricinin yazdığı Kod alanı veya CPU yazmaçları (Program Sayacı haricinde) değiştirilemez. Bellek konumlarına, INPUT (GİRİŞ) kutusu ve Program Sayacı alanlarına yalnız numara girişler kabul edilir.

Simülatörde hiçbir programı diske kaydetme yeteneği yoktur. Ama çevirici alanına yazılan programı kopyalayıp herhangi bir düzenleyici ile değiştirip sonra tekrar çevirici alanına yapııştırarak veya diske yazıp oradan LOAD (Yükle) kumandası ile çevirici alanına alarak aynı amaç sağlanabilir.

OPTIONS (SEÇENEKLER) düğmesi şu yararlı özelliklere sahiptir:

- clear memory - Tüm bellek yerlere sıfır koyun.
- show op codes - op kodlarını göster. Çevirici tercüme alanında Kodların adından sonra sayısal talimat kodunu göster.
- show decimal - Çevirici tercüme alanında sırayla bellek içeriğini, talimat kodunu, ve bellek adresini göster.
- hide op codes - op kodlarını gizle. Sadece talimat adını ve adresini göster.
- default fast - Hızlı varsayılan. En yüksek hızda çalışmaya başla (talimat yüklemeyi gösterme).
- default normal - Normal varsayılan. Orta hızda çalışmaya başla (talimat yüklemeyi göster).
- default slow - Yavaş varsayılan. Yavaş hızda çalışmaya başla (talimat yüklemeyi göster).

Eğer kod gösterilme tipi değiştirilirse Assembler (çevirici) otomatikman yeni tipi oluşturmak için çalışır.