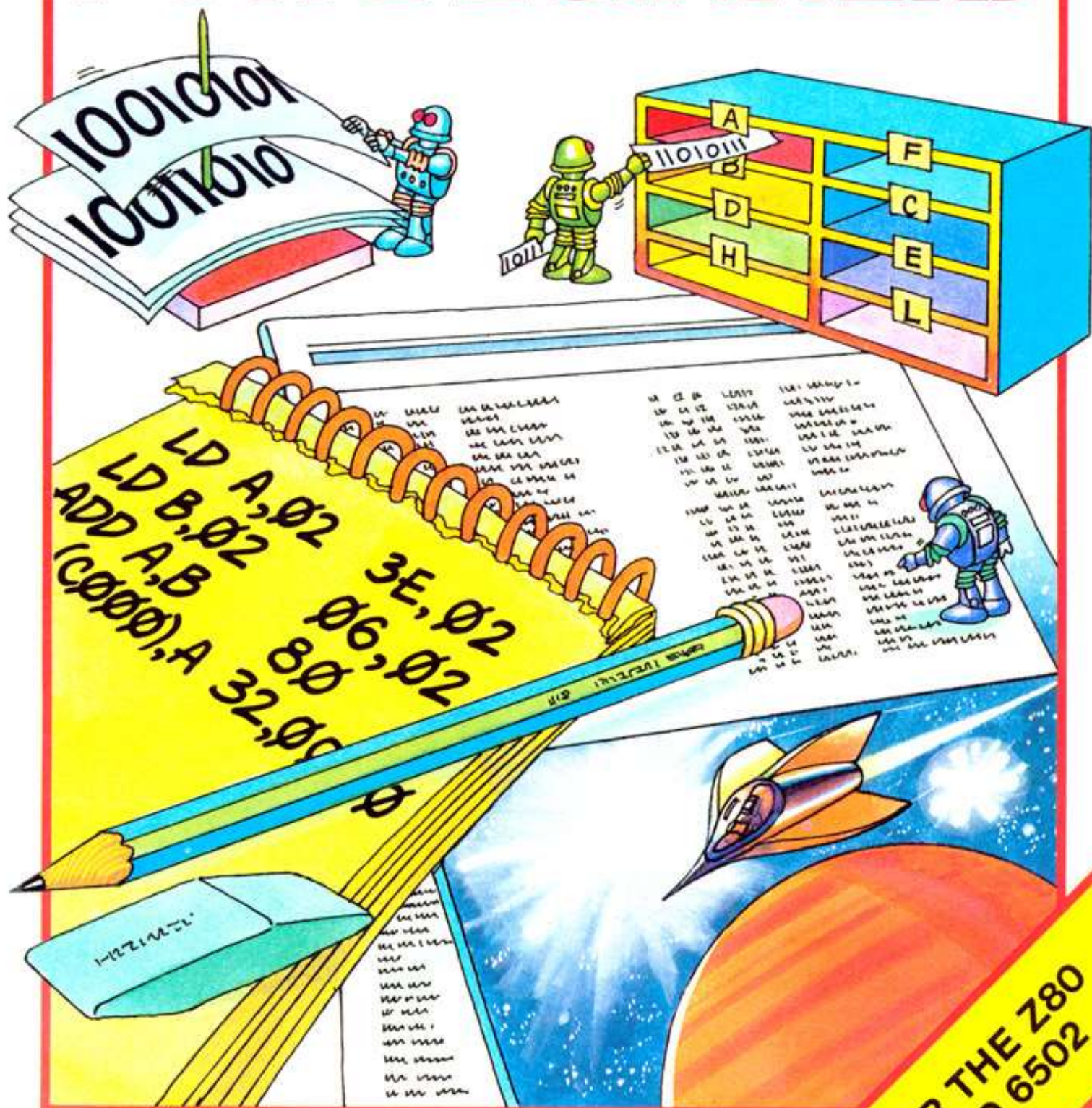


USBORNE INTRODUCTION TO



MACHINE CODE FOR BEGINNERS

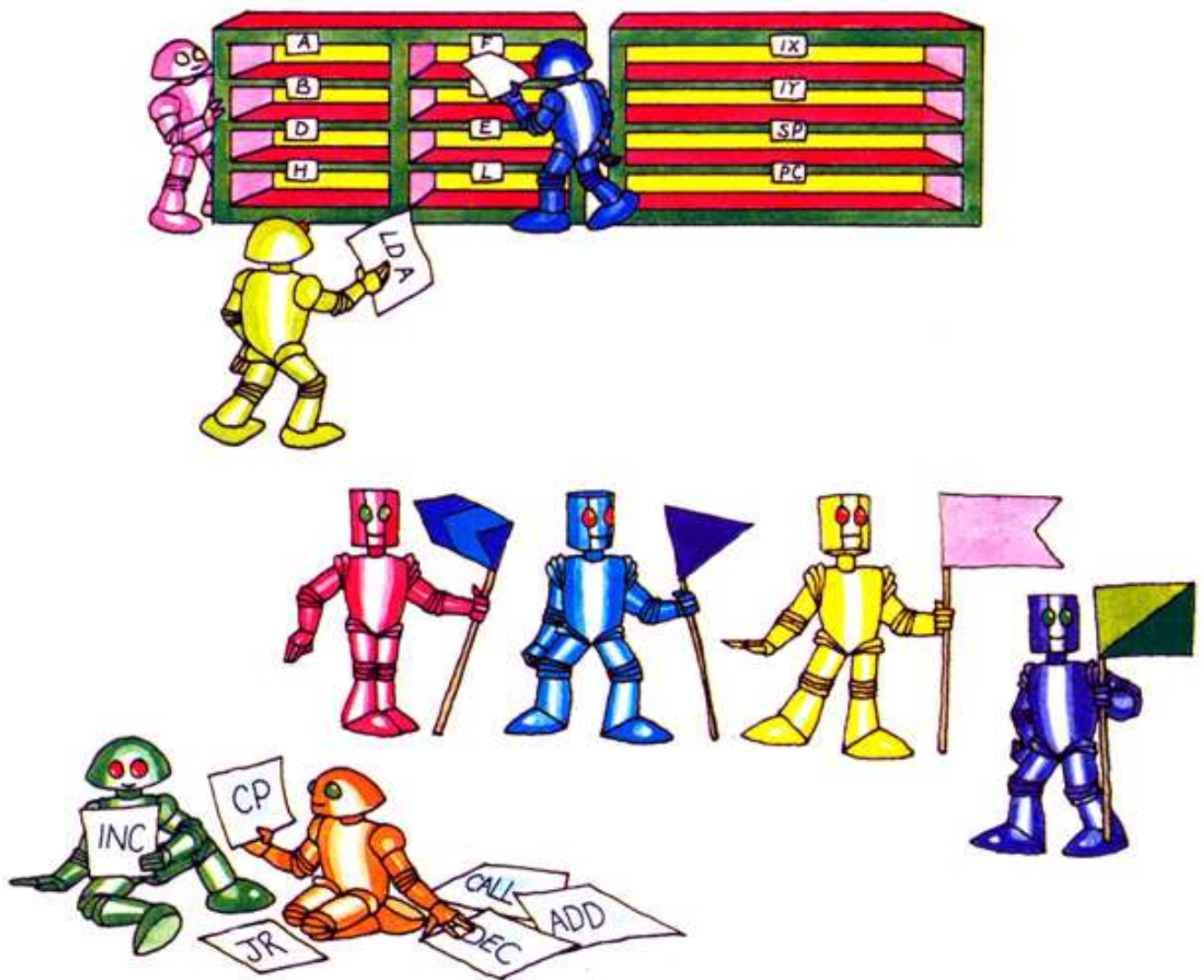


Usborne Computer Books

FOR THE Z80
AND 6502

USBORNE INTRODUCTION TO MACHINE CODE FOR BEGINNERS

Lisa Watts and Mike Wharton



Illustrated by Naomi Reed and Graham Round
Designed by Graham Round and Lynne Norman
6502 consultants: A. P. Stephenson and Chris Oxlade

İçerik

- 4 Makine kodu nedir ?
- 6 Bilgisayarını tanı
- 8 Bilgisayarların hafızası
- 11 Hex sayılar
- 12 Peeking and poking
- 14 Inside the CPU
- 16 Giving the CPU instructions
- 18 Translating a program into hex
- 20 Finding free RAM
- 23 Loading and running a program
- 27 Adding bytes from memory
- 28 Working with big numbers
- 29 The carry flag
- 30 Big number programs
- 32 Displaying a message on the screen
- 35 Jumping and branching
- 38 Screen flash program
- 40 Going further
- 41 Decimal/hex conversion charts
- 42 Z80 mnemonics and hex codes
- 45 6502 mnemonics and hex codes
- 46 Machine code words
- 48 Index

First published 1983 by Usborne Publishing Ltd, 20 Garrick Street, London WC2E 9BJ, England.

© 1983 Usborne Publishing

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the publisher.

The name Usborne and the device  are Trade Marks of Usborne Publishing Ltd.

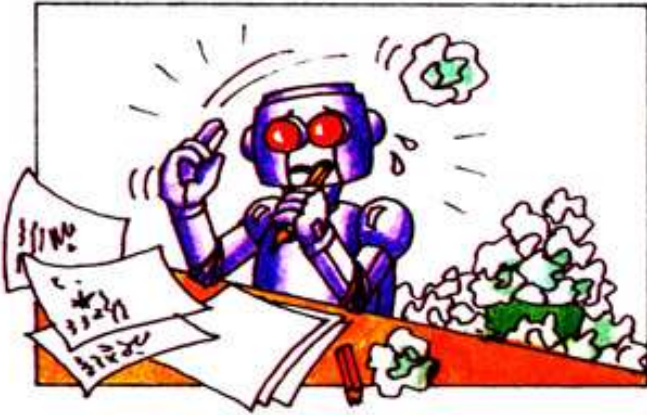
Printed in Spain by Printer Industria Gráfica, S. A. - Depósito Legal B. 33.755/1983

Kitap Hakkında

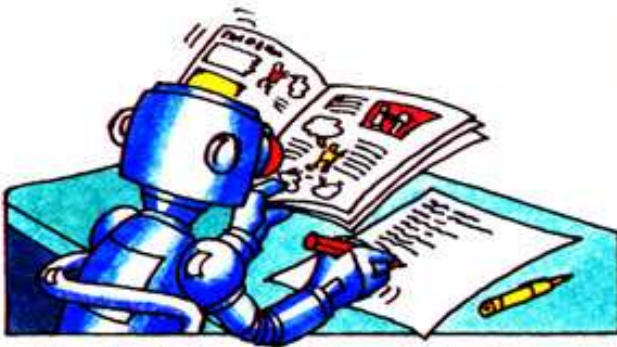
Bu kitabın amacı size makine kodunu adım adım öğretmek rehber olmaktır.

Makine kodu bir bilgisayarda tüm işi yapan koddur ve Makine kodu ile yazılan programlar BASIC'ge göre daha hızlı çalışır ve daha az yer kaplar.

Buna rağmen, Makine kodu BASIC'ge göre yazması daha zor ve anlaması daha az kolaydır.



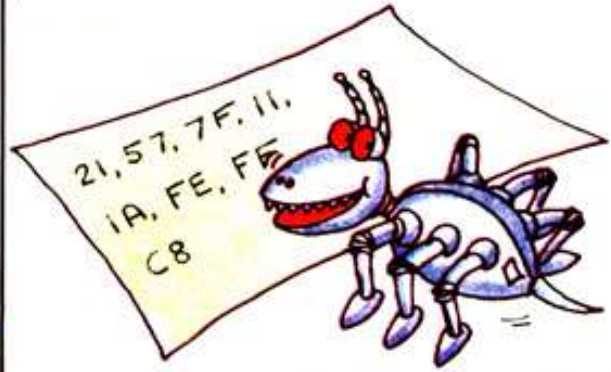
Kitap boyunca basit aşamalardan geçerek basit makine kodunun yapısını anlayacağız. Buna örnek olarak sana iki sayının toplanması veya ekrana yazı yazdırmak gibi örnekler vericek ve bu örnekleri kendi bilgisayarında nasıl uygulayacağını anlatacağız.



Bu kitap özel olarak Z80 veya 6502 mikroişlemcileri ile çalışan bilgisayarlar için yazıldı. Mikroişlemci bilgisayarların merkezi işlem birimini içeren çiptir ve bilgisayarlar farklı makine kodunu anlayan farklı mikroişlemcilerle gelir, yinede aynı mikroişlemciyi kullanan bilgisayarlar aynı zamanda aynı makine kodunu kullanırlar.

Makine kodu zor ve zahmetlidir bir çok uyulması gereken kural ve hatırlanması gereken küçük detay içerir. İlk başta makine kodunu okumayı ve anlamayı zor bulduysan endişelenmene gerek yok onlar sadece bir dizi sayı ve harf.

Hataların tespit edilmesi zorlayıcıdır eğer onları bulamazsan başına büyük dertler açabilir. Makine kodu ile çalıştığın sürece çok dikkatli ve metodik olmalı ve her şeyi iki veya üç defa kontrol etmelisin.

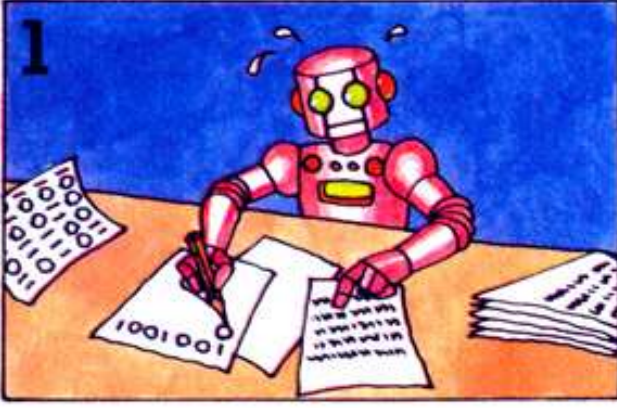


Gerçekten özel sebeplerin olmadığı sürece uzun makine kodu programları yazmanın bir anlamı yok bazı şeyler sadece BASIC kullanarak hallolabilir. Oyunların hızlarını arttırmak veya fantastik efektler yaratmak gibi görevlerin varsa makine koduna ihtiyaç duyabilirsin. Bu kitap sana BASIC programlarındaki makine kodu alt programlarını kullanarak programları nasıl daha heyecan verici hale getirebileceğini gösterecek.

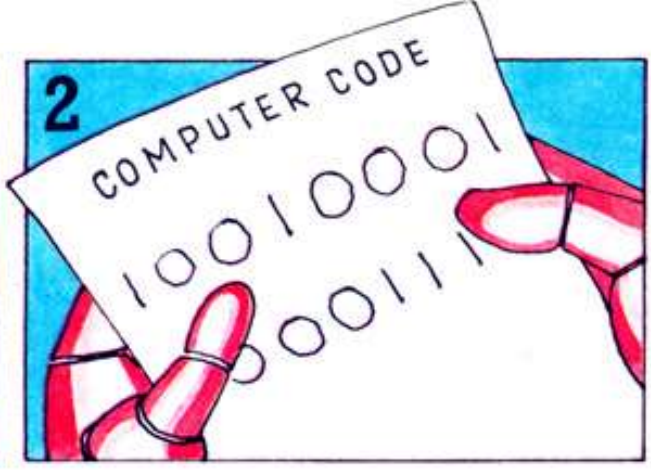


Kitabın arkasında, sen makine kodu yazarken yardımcı olması için bazı dönüşüm tabloları ve tüm jargonu açıklayan makine kodu kelimelerinin bir listesi var. Ayrıca orada kısa programlar yazmak için bir çok bulmaca ve fikir var. Bunlar cevaplarıyla birlikte sayfa 44'te.

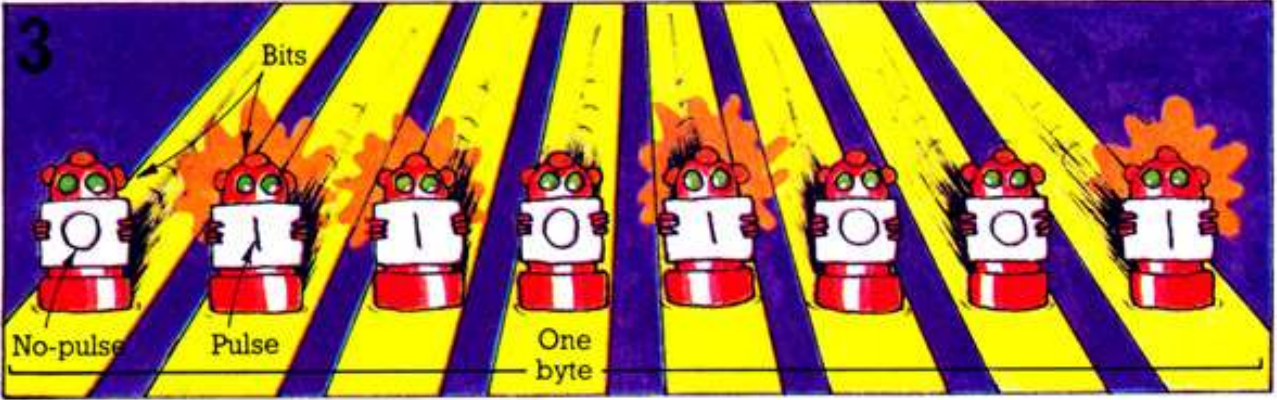
Makine kodu nedir?



Makine kodu bilgisayarın tüm işi yaptığı koddur. Bilgisayara BASIC programı verdiğin zaman tüm talimatlar ve veri bilgisayarın içinde makine koduna dönüşür.

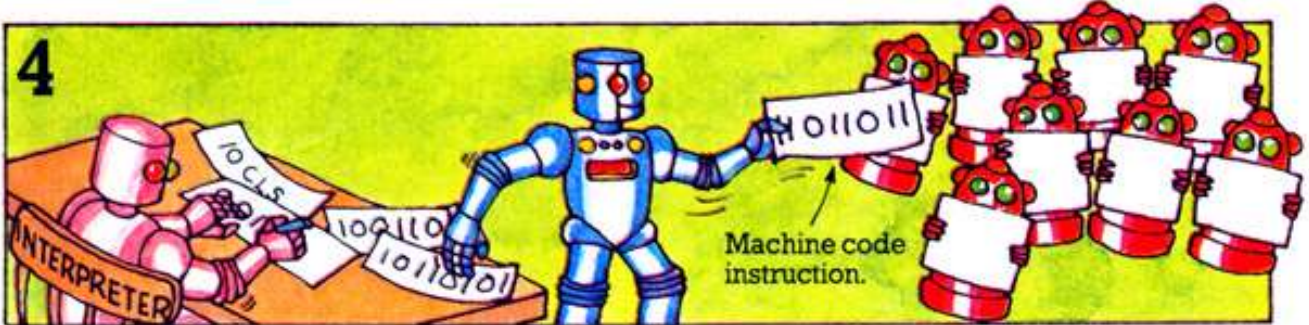


Makine kodunda, her bir talimat ve bilgi parçası ikili sayı(binary number) sisteminde temsil edilir. İkili sayı sistemi sadece 1 ve 0 şeklinde iki sayı kullanan sistemdir. İkili sayı sisteminde sadece 1'ler ve 0'lar kullanarak herhangi bir sayıyı yazabilirsin.



Bilgisayarın içinde ikili sayılar elektriksel sinyaller(pulses of electricity) olarak tanımlanır eğer sinyal varsa 1 yoksa 0'dır. Biz bu sinyallere bit deriz ve bu ikili rakamların bir kısaltmasıdır.

Bitler bilgisayarın içinde sekizli gruplar halinde akar ve her bir grup "byte" şeklinde çağrılır. Her byte sinyal varlığı veya yokluğu şeklinde bir talimat yada bir bilgi parçasını makine kodunda temsil eder.



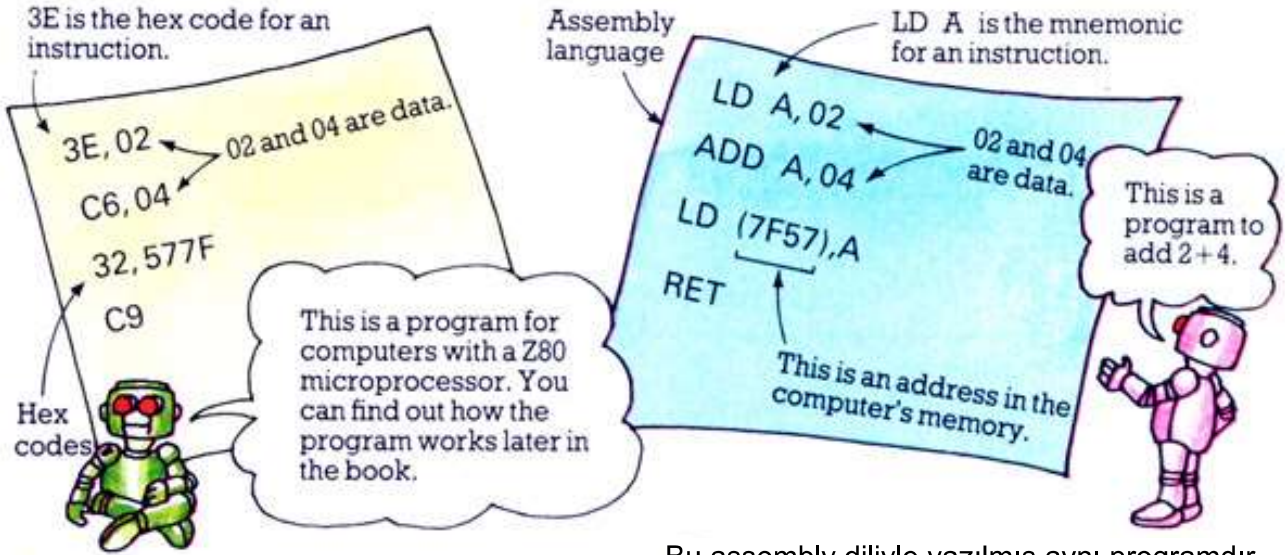
Bilgisayarın yapabileceği her görev ki bunlara örnek olarak iki sayıyı toplamak veya ekranı temizlemek gibi örnekler verilebilir, bunlar bir çok makine kodu talimatı içerir. Bilgisayarına bir BASIC komutu verdiğin zaman "interpreter" adında özel bir program çağrılır ve senin komutunu bilgisayarın anladığı makine kodu talimatlarına çevirir.

Makine kodu terimi, BASIC'ten çok daha yakın bir formda yazılmış programları ifade etmek içinde kullanılır. Ekranı temizlemek gibi bir makine kodu programında programın çalışabilmesi için tüm ayrı talimatları vermek zorundasın.

Makine kodunda programlama

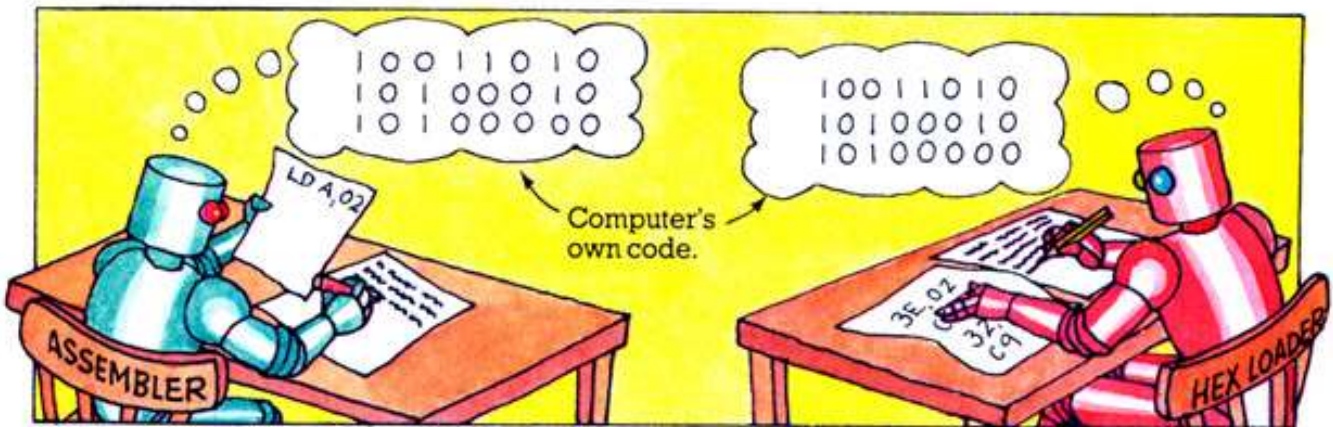
Bir çok makine kodu programı yazma yolu vardır, istersen tüm talimatları ikili sayılarla yazabilirsin ama bu sıkıcı olur onun yerine hexadecimal adıyla çağrılan başka bir sayı sistemini kullanabilirsin bu sayı sisteminin kısaltması hex'dir. Bir kere kullandıktan sonra hex sayı sistemi ile çalışmak ikili sayılara göre daha kolay olacaktır.

Makine kodu programları ayrıca "assembly language" şeklinde çağrılan dil ile yazılabilir. Assembly dilinde bilgisayara verilen her talimat "mnemonic" şeklinde tanımlanır bu kelime talimatın kısaltması gibidir.



Bu bölüm makine kodu programının hex'deki karşılığıdır. Hex sayı sisteminde 0-9 ve A-F arasında olan toplam onaltı sayı ve sembol vardır bunlar 0-15 arasındaki sayıları temsil eder.(Hex hakkında daha fazlasını sonra kitapda bulabilirsin.) Hex sayısı programın her satırının başlangıcında bir talimat ile başlar(örn. 3E). Bu binary kodundaki bir talimatın hex eşdeğeridir.

Bu assembly diliyle yazılmış aynı programdır. Her bir satır bir talimat için bir mnemonic içerir ve aynı satırın sol tarafında talimatın hex'deki karşılığı vardır. Örneğin mnemonic LD A'nın anlamı hex'deki 3E ile aynıdır. İki programda da her satırdaki talimat bilgisayarın kendi kodundaki tek talimatın eşdeğerini içerir.



Bilgisayara assembly dilinde bir program verirken "assembler" adında özel bir programa ihtiyacın var bu program içeride mnemonicleri bilgisayar koduna çevirir. Bazı bilgisayarlar zaten assemblera sahiptir diğerleri için kaset içinde bir assembler satın alabilir ve onu bilgisayarın hafızasına yükleyebilirsin. Alternatif olarak assembly dilindeki mnemonicleri kullanarak bir makine kodu programı yazabilir(bunları hatırlaması sayılardan daha kolaydır)

sonra onları ilk önce hex koduna çevirip bilgisayara verebilirsin. Bazı bilgisayarlar hex sayıları kabul edecektir ama diğerleri için onlara "hex loader" adında kısa bir program vermek zorundasın ki onları bilgisayar için dönüştürsün. Sayfa 24'te bir hex loader programı bulunmakta ki bu kitaptaki makine kodu programlarını yüklemek için kullanabilesin.

Bilgisayarını tanı

Makine kodu ile bir bilgisayar programladığın zaman her aşamada ona ne yapacağını tam olarak söylemek zorundasın örneğin veriyi nerede bulacak, nerede saklayacak veya nasıl ekrana yazdıracak gibi. (BASIC ile çalıştığın sürece, bilgisayarın içindeki özel programlar bu işleri senin için yapacaktır.) Bilgisayara doğru makine kodu talimatları vermek için bilgisayarın içinde neler döndüğüne dair iyi bir fikrin olmalı. Bu iki sayfadaki resimler evindeki bir bilgisayarın içindeki parçaları ve onların ne için var olduğunu gösterecek. Daha fazlasını gelecek bir kaç sayfada bulabilirsin.

Bir bilgisayarın içi

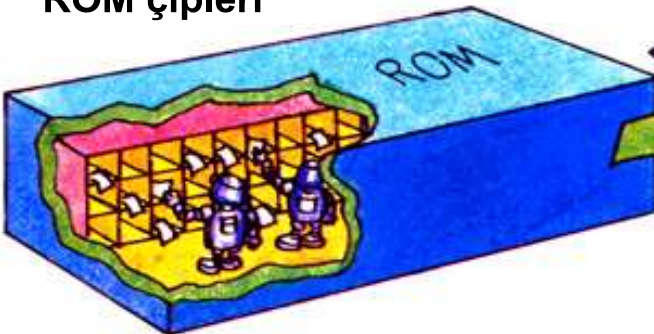


Mikrobilgisayarın klavyesinin içinde baskılı devre kartı bulunur ve elektrik akımının akması için metal yollara sahiptir. Bir çok çip borda basılı şekilde eklidir.

Çipler ne yapar

Bu resim bilgisayarın içindeki farklı çiplerin yaptığı işi gösterir. Çiplerin arasındaki byte'lar şeklinde olan mesaj akışını yani sekizli gruplardan oluşan bitlerin değerine göre veriyi ve talimatları temsil eder.

ROM çipleri



ROM sadece okunabilir hafıza anlamına gelir. Bilgisayara ne yapması gerektiğini söyleyen makine kodu talimatları ROM çiplerinde saklanır. Onlara sadece okunabilir hafıza denmesinin sebebi bilgisayarın sadece ROM'daki bilgiyi okuyabilip oraya yeni bilgi koyamamasıdır. Bilgisayarların çoğunda interpreter ROM'un içindedir.

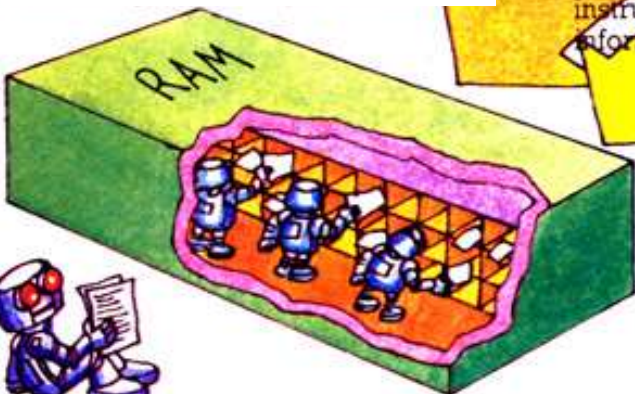
(Interpreter BASIC komutlarını bilgisayar koduna çeviren programdır.)

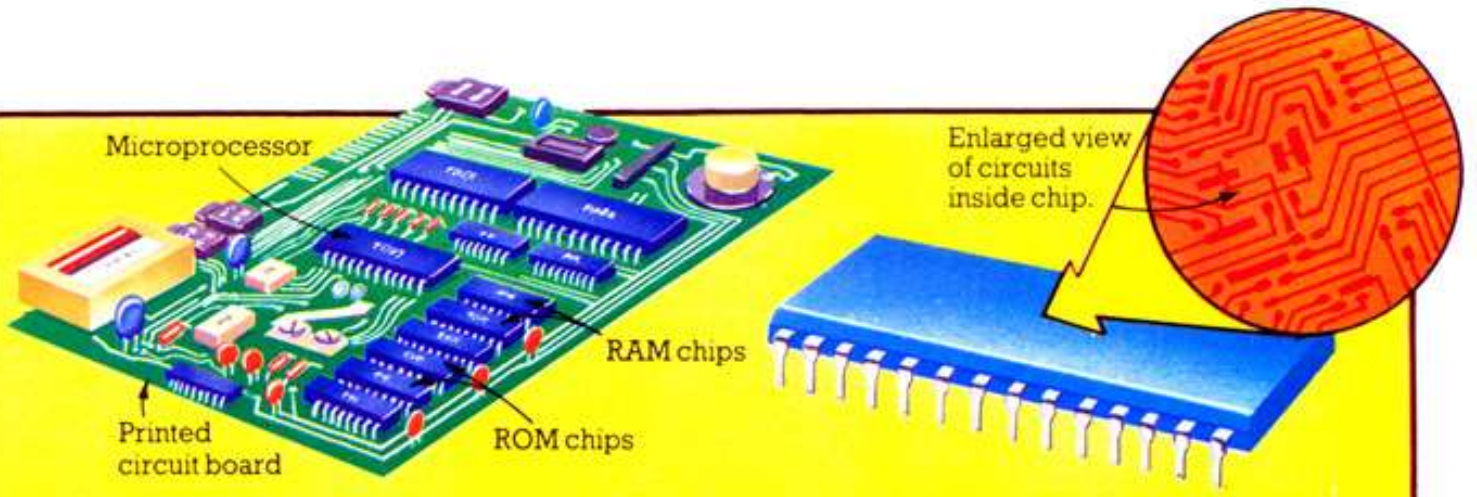


Bilgisayar kodundaki byte'lar çipler arasında devre borduna basılmış yollar ile akar. Farklı işleri gerçekleştirmek için üç farklı byte taşıma sistemi vardır. Her bir taşıma sistemi "bus" şeklinde adlandırılır.

RAM çipleri

RAM "Rastgele erişimli bellek" olarak adlandırılır. Bilgisayarın üzerinde çalıştığı programların depolandığı yerdir. Rastgele erişimli bellek diye çağrılır çünkü bilgisayar bilginin her parçasına hafızanın her yerinden ulaşabilir. Bilgisayarı kapattığında RAM'de depolanmış bilgi silinir.





Bir çip için doğru adlandırma "integrated circuit" şeklindedir ve her bir çipin içinde mikroskobik elektrik devreleri vardır. Tüm bilgisayarlar elektrik sinyallerinin akması ile çalışırlar ve bu sinyaller binary kodundaki talimatları temsil edip çiplerin içindeki devrelere doğru akar.

Farklı işleri gerçekleştirmek için farklı çipler vardır. Bu işleri yapan farklı türdeki çipler resmin altında gösterilmiştir.

Clock

Bu bir quartz crystal'dir, saniyede milyonkere sinyal verir ve sinyallerin akışını bilgisayarın içinde düzenler.



Mikroişlemci nedir

Mikroişlemci çipi içinde bilgisayarın merkezi işlem birimini diğer adıyla CPU'sunu saklar. CPU bilgisayarın tüm işi yaptığı yerdir. CPU hesaplamaları, veri parçalarını karşılaştırmayı, karar vermeyi ve bilgisayarın içindeki tüm diğer aktiviteleri düzenler. CPU'ya ne yapması gerektiğini ROM'un içindeki bilgi söyler.

Your computer's microprocessor

There are several different makes of microprocessor chip and all home computers use one or other of them. The most common makes are the Z80, found in Sinclair (Timex) computers, and the 6502, used in the Oric, the BBC micro, the VIC 20 and many others. Other examples of microprocessors are the 6809 used in the Dragon computer and the 9900 in the TI 99/4.

The different makes of microprocessor use different machine code instructions, but all computers with the same kind of microprocessor use the same machine code. There are also several different versions of each chip. For example, the Z80A is a faster version of the Z80 and the 6502A and 6510 (used in the Commodore 64) are versions of the 6502. They use the same machine code instructions, though, as the chips from which they were developed. This book is written specially for computers with microprocessors which understand Z80 or 6502 machine code.

Bilgisayar hafızası

Bilgisayar hafızasını düşünmenin en kolay yolu hafızanın küçük kutulardan oluştuğunu varsayıp her bir kutunun bir byte'ı sakladığını düşünmektir yani makine kodundaki bir talimat veya bir bilgi parçası bu kutularda saklanır. Hafızadaki her bir kutu "location" olarak adlandırılır ve her bir location bir numaraya sahiptir ve bu numaralar "address" şeklinde adlandırılır yani bilgisayar hafızadaki herhangi bir kutuyu bulabilir. Hafızadaki farklı alanlar farklı görevler için farklı bilgileri depolar ve bir tabloda verilen adres her bir alanın nerede başladığını gösterir ve "memory map" olarak adlandırılır. Makine kodunda programlama yaparken bilgisayara her bir talimatı veya bilgi parçasını nerede bulacağını yada nerede depolayacağını söylemek zorundasın. Bunu sana verilen hafıza konumlarının adresini kullanarak yaparsın. Hatta makine kodu programının kendisinin nerede depolanacağını bile söylemek zorundasın yani bilgisayarının memory map'ini tanıman gerekli.

The memory map

The picture on the right shows the memory map of a home computer. There should be a map for your computer in your manual. The memory is organized differently in different makes of computer, so your map will look different from this one.

The memory map may be drawn as a column like this, or horizontally. The address at which each of the different areas in the memory starts is given alongside the map and it may be a decimal number or a hex number, or both, as here. In this book hex numbers are distinguished by a & sign (ampersand) before the number. Your manual may use a different symbol, e.g. \$, %, or #.

The highest address in user RAM is called "RAMTOP", or on some computers, "HIMEM".

BASIC

This area holds the interpreter, the program which converts instructions in BASIC into the computer's binary code.

Operating system

This area contains a group of programs called the "operating system" or "monitor", which tell the computer how to operate. All the programs are in machine code. There are programs which tell it how to do mathematical calculations, programs to clear the screen, find a random number, scan the keyboard and all the other things the computer has to do in the course of its work.

