José Augusto N. G. Manzano

LINGUAGEM ASSEMBLY

INTRODUÇÃO AO PADRÃO INTEL 8086

```
=============================================================
[LINE]     LOC: MACHINE CODE     SOURCE
=============================================================

[   1]          :                ORG 100H
[   2]          :
[   3]     0100: B8 0A 00        MOV AX, 000AH
[   4]     0103: BB 0B 00        MOV BX, 000BH
[   5]     0106: 50              PUSH AX
[   6]     0107: 53              PUSH BX
[   7]     0108: E8 01 14        CALL PROCEDIMENTO
[   8]     010B: 40              INC AX
[   9]     010C: 43              INC BX
[  10]     010D: E8 01 14        CALL PROCEDIMENTO
[  11]     0110: 5B              POP BX
[  12]     0111: 58              POP AX
[  13]     0112: CD 20           INT 20H
[  14]          :
[  15]     0114: B8 01 00        PROCEDIMENTO: MOV AX, 0001H
[  16]     0117: BB 02 00                      MOV BX, 0002H
[  17]     011A: 40                            INC AX
[  18]     011B: 43                            INC BX
[  19]     011C: C3                            RET

=============================================================
```

EDIÇÃO PROPES VIVENS

José Augusto N. G. Manzano

ORCID: 0000-0001-9248-7765

Linguagem Assembly

INTRODUÇÃO AO PADRÃO INTEL 8086

São Paulo
2021 - Propes Vivens

ISBN 978-65-00-24498-4

Dados Internacionais de Catalogação na Publicação (CIP)
(Câmara Brasileira do Livro, SP, Brasil)

Manzano, José Augusto N. G.
Linguagem Assembly [livro eletrônico] : introdução ao padrão Intel 8086 / José Augusto N. G. Manzano. -- São Paulo : Propes Vivens, 2021.
PDF

Bibliografia.
ISBN 978-65-00-24498-4

1. Intel 8086 (Microprocessador) 2. Linguagens de montagem (Computadores) 3. Microcomputadores IBM I. Título.

21-70472 CDD-005.136

Índices para catálogo sistemático:

1. Assembly : Linguagem de programação : Computadores : Processamento de dados 005.136

Cibele Maria Dias - Bibliotecária - CRB-8/9427

Revisões

1.00 - 13/05/2021 - lançamento
1.01 – 30/06/2022 – atualização e revisão após mudanças de versões de alguns dos softwares usados

Produção e Editoração: José Augusto Navarro Garcia Manzano
Imagens: canva.com e freepng.es (ibm-pc)
https://www.freepng.es/png-evckf6/

Edição: Propes Vivens

FERRAMENTAS UTILIZADAS

Produto: **EMU8086 - MICROPROCESSOR EMULATOR [Versão 4.08r11]**
Sítio: http://www.emu8086.com (descontinuado)
Alternativa: https://github.com/AhmadNaserTurnkeySolutions/emu8086
Custo:

Private Licence	4,95 USD
1 Year School Licence	52,00 USD
Permanent Size Licence	295,00 USD

Produto: **ENHANCED DEBUG [Versão 1.32b]**
Sítio: https://pcdosretro.github.io/enhdebug.htm

Produto: **VDOS [Versão 2022.05.01]**
Sítio: https://vdos.info/

Produto: **VDOSPLUS [Versão 2015.11.01]**
Sítio: http://vdosplus.org/

REQUISITOS DE HARDWARE E SOFTWARE

Requisitos Comuns

- 1 MB de memória RAM;
- 10 MB de espaço em disco para instalação do compilador;
- Sistema Operacional Windows XP, Vista, 7, 8, 8.1 ou 10;
- Monitor com 1024 x 768 pontos ou superior para melhor visualização;
- Mouse ou outro periférico de apontamento;
- Modem e acesso à Internet;
- Programa emu8086 adquirido com custo financeiro;
- Programa DOS Debug versão 1.25.

OBS.: No Microsoft Windows 7 de 32 bits após atualizações poderão ocorrer problemas na execução do programa emu8086. Neste caso é aconselhável trabalhar com o programa em modo XP Mode. Use o modo XP Mode para rodar os programas debug do próprio XP ou o emu8086 quando estiver em uso a versão de 64 bits do sistema operacional Microsoft Windows 7 ou 10. Outra dica é executar todos os programas como administrador para minimizar eventuais problemas.

Foram feitos testes de execução do programa emu8086 no Microsoft Windows 8.1 e 10 de 32 e 64 bits e aparentemente não ocorreram problemas. No entanto na versão 64 bits o programa Microsoft Debug não é encontrado e não poderá ser executado. No entanto, há uma alternativa para executar do programa no sistema operacional Windows 8, 8.1 e 10 de 64 bits por meio do programa de virtualização vDos retratado nesta obra.

Contato com o autor: *certificado@mail.com*

Colocar em assunto o título *Linguagem Assenbly*.

Somente serão respondidos e-mails com dúvidas relacionadas, estritamente, ao que é apresentado neste trabalho. Qualquer outro tipo de dúvida ou questão fora deste contexto, não será atendida.

RESTRIÇÃO OPERACIONAL

Devido a estrutura operacional de uso dos microprocessadores Intel 8086/8088 e do foco deste trabalho devem ser consideradas as seguintes restrições operacionais:

- Os programas apresentados nesta obra são desenvolvidos numa esfera simplista, apresentam apenas os recursos em média discutidos nos capítulos que são focados;
- Não são tratadas as entradas de valores numéricos com ponto flutuante, apenas entradas de dados baseada em valores numéricos inteiros e dados alfabéticos;
- O objetivo deste trabalho é ser uma introdução didática, o mais simples possível, apenas e tão somente para leitores que não conhecem absolutamente nada em programação Assembly;
- Não são tratados nenhum aspecto considerado intermediário ou avançado no nível de desenvolvimento dos programas ou do conteúdo da obra;
- Não é apresentado nenhum programa contextualizando o uso da linguagem Assembly para componentes de automação ou de uso comercial.

AGRADECIMENTOS

À minha esposa Sandra e à minha filha Audrey, motivos de constante inspiração ao meu trabalho. Pela paciência nos momentos de ausência que tenho, quando estou absorto em escrever, dedicando-me ao ensino;

A Peter Norton, bandeirante desbravador, que sempre direcionou seu trabalho literário ao conhecimento mais refinado no uso dos microcomputadores padrão IBM-PC e principalmente ao processo de entendimento do funcionamento dos microprocessadores (micro controladores) Intel, como o fez com John Socha em sua obra Peter Norton's Assembly Language Book for IBM-PC.

A Alexander Popov que, com seu brilhantismo profissional, iniciou o desenvolvimento da ferramenta de estudo da linguagem de programação Assembly emu8086, e também a Tomasz Grystar, desenvolvedor do programa Flat Assembler que tem sua contribuição registrada neste trabalho, por tornarem o estudo da programação em baixo nível muito agradável e de fácil compreensão.

Ao amigo Stephen P. Morse, engenheiro que desenvolveu o microprocessador 8086/8088 e quando o fez, não imaginou que sua criação seria utilizada para compor um dos microcomputadores mais utilizados no mundo. Obrigado amigo pelo auxilio que ofereceu a mim quando interagimos sobre seu trabalho.

Há ainda uma pessoa que merece grande atenção, você, leitor, que me prestigia adquirindo esta e outras obras de minha autoria, sempre de forma legal, na livraria de sua confiança. Devido à sua postura tenho incentivo para continuar escrevendo, mesmo num País em que está enraizada a cultura da pirataria, das famigeradas apostilas e da cópia ilegal do trabalho alheio. Muito obrigado por sua confiança e respeito.

A todos os alunos que passaram e passam por minhas mãos, que acreditaram e acreditam na minha pessoa e seguiram e seguem as orientações passadas; por me incentivarem continuamente quando me questionam sobre temas que ainda não conheço, por me levarem a um patamar maior por exigirem assim que eu pesquise mais.

Vida longa e próspera.

SUMÁRIO

Parte I - Noções preliminares

Capítulo 1 - Introdução

Capítulo 2 - Conceitos fundamentais

Parte II - Programação com Enhanced DEBUG

Capítulo 3 - Ferramentas de desenvolvimento

Capítulo 9 - Saltos, decisões e repetições

Capítulo 10 - Detalhes complementares

Capítulo 11 - Recursos específicos

Capítulo 12 - Como em alto nível

Parte IV - Apêndice

PREFÁCIO

O objetivo central deste trabalho é ser um veículo de apresentação dos princípios preliminares e básicos que norteiam o início do aprendizado da linguagem de programação Assembly 8086/8088 de microcomputadores IBM-PC em baixo nível. Não há pretensão de explorar o tema com profundidade. Este livro é direcionado tão somente a leitores iniciantes. É uma introdução a atividade de programação em baixo nível. Se o leitor possui noções de programação em linguagem Assembly ou programação em baixo nível com *opcodes*, este material não é para você, pois não lhe acrescentará novos conhecimentos.

A escolha do modelo Assembly 8086/8088 para este estudo e das funções de programação básica de um microprocessador padrão Intel ou AMD se justifica por dois motivos: primeiro pelo fato de ser a mais usada e difundida para o ensino e aprendizado de programação de computadores em baixo nível em várias escolas espalhadas pelo mundo; segundo pelo fato das ferramentas *debug* e *emu8086* fazerem uso do padrão Intel 8086/8088.

Nos vários cursos de linguagem de programação Assembly nas mais variadas instituições de ensino superior espalhadas pelo mundo, normalmente escolhe-se o modelo 8086/8088 de microprocessador para o início do estudo, pois a família de computadores IBM-PC se desenvolveu e tornou-se popular sobre esta arquitetura. Não adianta tentar apresentar os últimos recursos do último modelo de um microprocessador Intel ou AMD para um iniciante que não conhece as bases mínimas desse ambiente de desenvolvimento.

O livro está dividido em quatro partes lógicas, nas quais se procurou agrupar o tema em áreas de concentração operacional (o que não é algo simples em se tratando de programação em baixo nível, pois os temas facilmente se misturam), no sentido de formarem as partes mínimas do todo, para o desenvolvimento do entendimento mental voltado à programação de computadores padrão IBM-PC em baixo nível.

A parte I apresenta as noções preliminares do estudo. São estudados a diferença entre assembly e assembler; o contexto do sistema computacional, a organização básica dos dados, os registradores, as interrupções, os segmentos, os deslocamentos e endereçamento de memória. Essa parte dá a noção básica relacionada à arquitetura interna do modelo de microprocessador 8086/8088.

A parte II destaca o uso do programa DEBUGX. Aqui são apresentados conceitos de aritmética em modo hexadecimal, representação de valores negativos, cálculos com códigos de máquina, movimentação de dados, apresentação de caracteres, valores binários e hexadecimais, utilização básica dos registradores de estado, apresentação inicial de salto condicional, entrada de dados por teclado, utilização de procedimentos, de pilha e o controle da consistência da entrada de dados. É pertinente salientar que boa parte dos programas exemplo utilizados foram baseados ou são adaptações dos programas encontrados na obra dos autores NORTON & SOCHA de 1987, que melhor descreveram as bases iniciais para o aprendizado da programação em baixo nível. Em alguns trechos dos capítulos desta parte são usados elementos colorizados para permitir melhor visualização dos elementos técnicos abordados.

A parte III trata das informações de aperfeiçoamento em relação ao estudo da parte anterior. É feita uma descrição dos recursos básicos da ferramenta emu8086, e são apresentadas informações teóricas de manipulação de registradores e dados (uma abordagem mais prática encontrada no final da obra). Outros pontos apresentados são a definição de tipos de dados e de cálculos matemáticos mais elaborados, o uso de saltos com decisões e laços de repetição, o uso de instruções lógicas, utilização de macros em complementação ao procedimento, apresentação mais detalhada de segmentos e deslocamentos, criação de biblioteca externa, apresentação de valores negativos, utilização da biblioteca de recursos que acompanha a ferramenta emu8086, abordagem de endereçamento e acesso à memória.

A parte IV apresenta anexo com informações complementares.

Por se tratar de um trabalho de apresentação dos recursos básicos da linguagem de programação Assembly 8086/8088, não é feito um estudo aprofundado das instruções do microprocessador Intel 8086/8088. Apresenta–se em torno de 60% do conjunto de instruções, o que já dá muito material para um estudo bem demorado.

Como sempre, a todos um grande abraço e um bom aprendizado. Que este trabalho seja de grande proveito e o investimento financeiro, ora nele feito, seja válido e proveitoso no final do seu estudo desta obra.

Augusto Manzano

"O aprendizado é um tesouro que segue seu dono por todos os lugares." - Provérbio chinês

SOBRE O AUTOR

José Augusto Navarro Garcia Manzano é professor com mestrado possuindo formação em Análise e Desenvolvimento de Sistemas, Ciências Econômicas e Licenciatura em Matemática atuando na área de Tecnologia da Informação, Computação e Informática (desenvolvimento de software, ensino e treinamento) desde 1986. Participou do desenvolvimento de aplicações computacionais para áreas de telecomunicações e comércio. Na carreira docente iniciou sua atividade docente em cursos livres, trabalhando posteriormente em empresas de treinamento e atuando desde então nos ensinos técnico e superior.

Atualmente é professor com dedicação exclusiva no IFSP (Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, antiga Escola Técnica Federal). Em sua carreira desenvolveu competências e habilidades para ministrar componentes curriculares de Lógica de Programação (Algoritmos), Estrutura de Dados, Microinformática, Informática, Linguagens de Programação Estruturada, Linguagens de Programação Orientada a Objetos, Engenharia de Software, Sistemas de Informação, Engenharia da Informação, Arquitetura de Computadores e Tecnologias Web. Possui conhecimento de uso de diversas linguagens de programação imperativas, descritivas e de marcação, tais como: BASIC (clássico e estruturado), COBOL, COMAL, Logo, Scratch, Assembly, Pascal, FORTRAN (77 e 90), C, C++, D, Java, F#, Modula-2, C#, Lua, HTML, XHTML, CSS, Javascript, VBA, Ada, Rust, Python, LISP, Haskell, Standard ML, OCaml, Miranda, Hope, Groovy, Julia e Elixir. Possui mais de uma centena de obras publicadas, além de diversos artigos no Brasil e exterior.

.