

## Interface Hardware-Software:

[www.cin.ufpe.br/~if817](http://www.cin.ufpe.br/~if817)

Prof. Sergio Queiroz

[srmq@cin.ufpe.br](mailto:srmq@cin.ufpe.br)



(Slides baseados nos cursos anteriores ministrados pelos Profs. Eduardo Tavares e Paulo Maciel)

# Apresentação

Nome: Interface Hardware-Software (IF817)

Página: [www.cin.ufpe.br/~if817](http://www.cin.ufpe.br/~if817)

Carga Horária: 75 horas

Faltas: 25% da carga horária

Horário: Terças (15-17) e Quintas (13-15)

Aulas Teóricas (Sala D-001) e Práticas (Lab. C007)

Monitores (já confirmados, deveremos ter outros também):

Vandi Alves de Lira Neto (valn)

Rodrigo Bruno de Carvalho Cavalcanti (rbcc)

Andre Olivino da Silva (aos)

# Ementa

Processador Pentium: Visão Geral

Assembly

Assemblers

Interrupção e E/S

BIOS

Programação de Ponto Flutuante

Tópicos em Linguagem C

Programação Linux

# Avaliação

## **1ª Nota:**

- Mini-teste
- Projeto
- Prova

## **2ª Nota**

- Mini-teste
- Projeto
- Prova

## **Final:**

- Prova

# Bibliografía

## **Básica:**

- The Intel Microprocessor Family: Hardware and Software Principles and Applications. James Antonakos
- A Guide to Assembly Programming Language in Linux, Dadamudi, Springer.

## **Complementar:**

- Pentium Processor System Architecture. Addison Wesley.
- Assembly Language Step-by-Step. Programming with DOS and Linux. Jeff Duntemann. Wiley.
- Assembly Language Step-by-Step. Programming with Linux. Jeff Duntemann. Wiley.
- Linux Assembly Language Programming. Bob Nevln. Prentice Hall.
- Professional Assembly Language. Richard Blum. Wiley.

# Estilos de arquitetura

- Vejamos um programa para computar a expressão  $C = A + B$

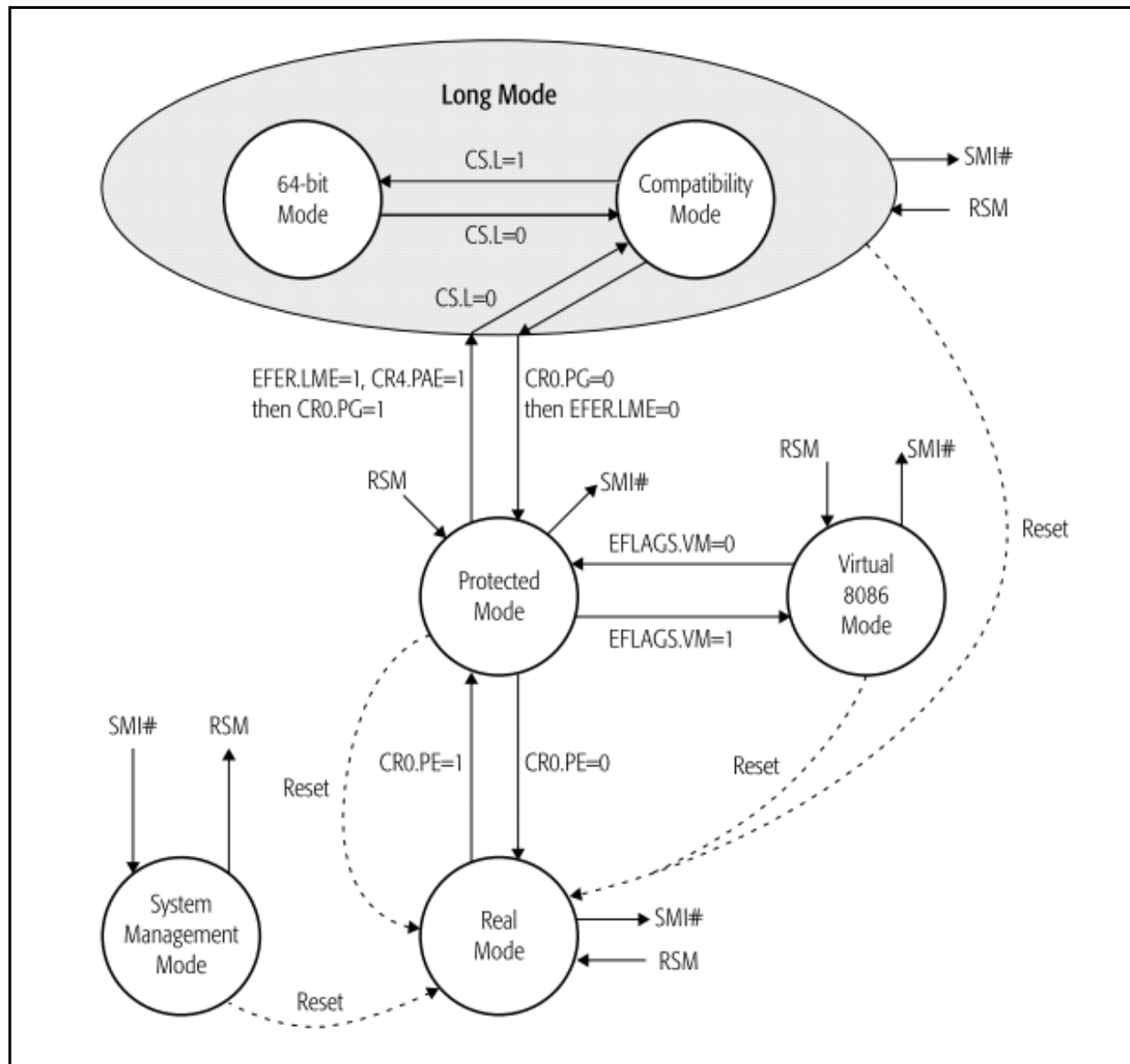
Arquitetura de pilha	Arquitetura de Acumulador	Registrador-Memória	Memória-Memória	Registrador (load-store) [RISC]
Push A	Load A	Load r1,A	Add C,B,A	Load r1,A
Push B	Add B	Add r1,B		Load r2,B
Add	Store C	Store C,r1		Add r3,r1,r2
Pop C				Store C,r3

E nos processadores x86?

# x86 = CISC = **Complex Instruction Set Computer**

- Os processadores da família x86 possuem instruções que se enquadram em todos os estilos do slide anterior
  - Exemplo:
    - Unidade de ponto flutuante x87: usa majoritariamente uma arquitetura de pilha
    - Registrador AX: tem o papel de acumulador em várias versões de instruções aritméticas
    - Várias instruções têm versões com operandos entre registrador e memória
    - ...
- A evolução da família ao longo de mais de 30 anos mantendo a compatibilidade com versões anteriores acrescenta complexidade
  - Processadores x86 atuais podem ser usados como 16bit, 32bit, 64bit e continuam compatíveis com os programas escritos para o 8086, **de 1978!**

# X86 – Modos de Operação



# X86 – Formato de instrução (16/32 bits)

Instruction Prefixes	Opcode	ModR/M	SIB	Displacement	Immediate
Até 4 prefixos de 1 byte cada (opcional)	1, 2 ou 3 bytes	1 byte (se necessário)	1 byte (se necessário)	Deslocamento do endereço: 1, 2 ou 4 bytes ou nenhum	Dados imediatos: 1, 2 ou 4 bytes ou nenhum

Passam informações adicionais à instrução. Exemplo: usar os 32bits do registrador (estando em modo 16 bit); prefixo de repetição para operações em strings

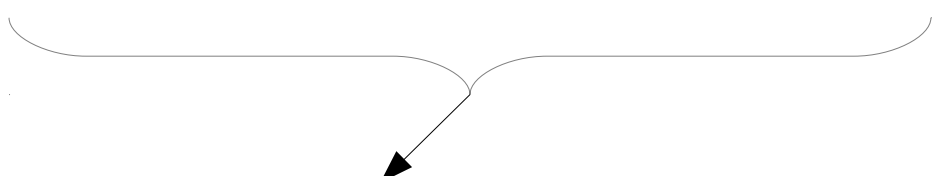
# X86 – Formato de instrução (16/32 bits)

Instruction Prefixes	Opcode	ModR/M	SIB	Displacement	Immediate
Até 4 prefixos de 1 byte cada (opcional)	1, 2 ou 3 bytes	1 byte (se necessário)	1 byte (se necessário)	Deslocamento do endereço: 1, 2 ou 4 bytes ou nenhum	Dados imediatos: 1, 2 ou 4 bytes ou nenhum

Opcode da instrução. Algumas instruções já codificam informações adicionais no próprio Opcode. Exemplo: direção da operação (incremento/decremento)

# X86 – Formato de instrução (16/32 bits)

Instruction Prefixes	Opcode	ModR/M	SIB	Displacement	Immediate
Até 4 prefixos de 1 byte cada (opcional)	1, 2 ou 3 bytes	1 byte (se necessário)	1 byte (se necessário)	Deslocamento do endereço: 1, 2 ou 4 bytes ou nenhum	Dados imediatos: 1, 2 ou 4 bytes ou nenhum



Codificam as informações de endereçamento

# X86 – Formato de instrução (16/32 bits)

Instruction Prefixes	Opcode	ModR/M	SIB	Displacement	Immediate
Até 4 prefixos de 1 byte cada (opcional)	1, 2 ou 3 bytes	1 byte (se necessário)	1 byte (se necessário)	Deslocamento do endereço: 1, 2 ou 4 bytes ou nenhum	Dados imediatos: 1, 2 ou 4 bytes ou nenhum

Se a instrução trazer dados imediatos, eles podem ser de 1, 2 ou 4 bytes

# Principais registradores no x86 (16/32bits)

