



USP - ICMC - SSC
SSC 0610 - Eng. Comp. - 2o. Semestre 2010

Disciplina de
Organização de Computadores I

Prof. Fernando Santos Osório

Email: fosorio [at] { icmc. usp. br , gmail. com }

Página Pessoal: <http://www.icmc.usp.br/~fosorio/>

Estagiário PAE Maurício Dias - Email: [macccdias \[at\] gmail.com](mailto:macccdias@gmail.com)

Material on-line Wiki ICMC - <http://wiki.icmc.usp.br/index.php/Ssc-610>

1

Agosto 2010

Aula 09s

Aula 09 – Microprocessadores

Conteúdos Abordados:

- 1. Microprocessador Intel 8086**
- 2. Programação 8086 e 80x86**
 - Assembly, Assembler
 - Emulador 8086
 - Interrupções de Hardware e de Software
BIOS, BDOS (IBM-PC)

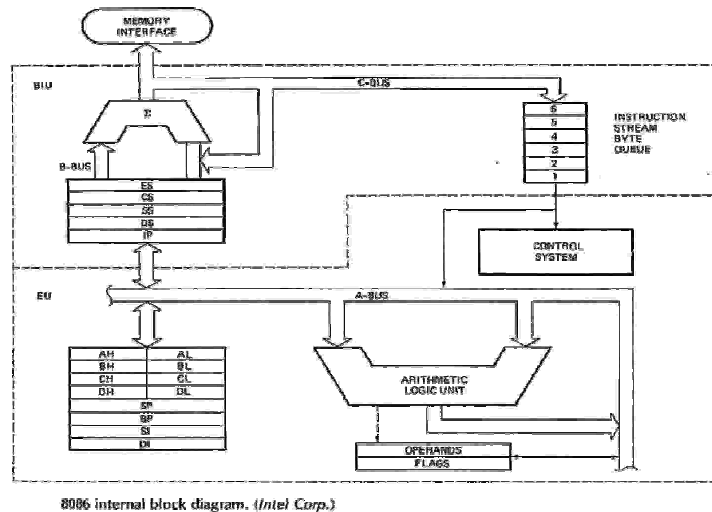
2

Out. 2009

O material a seguir é baseado no Livro “Arquitetura de Computadores Pessoais” de Raul Weber e nas transparências do prof. Weber obtidas no site <ftp://ftp.inf.ufg.br/pub/inf112/> e <http://www.inf.ufg.br/~fmc/arqcomp/Intel.ppt>

1. Microprocessador Intel 8086

Arquiteturas Intel 8086 / 8088 (16 bits)



3

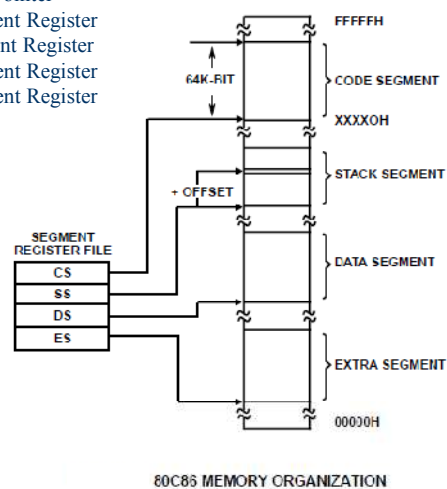
Out. 2009

1. Microprocessador Intel 8086

Arquiteturas Intel 8086 / 8088 (16 bits)

AX - Accumulator
BX - Base Register
CX - Count Register
DX - Data Register
SP - Stack Pointer
BP - Base Pointer
SI - Source Index Register
DI - Destination Register
IP - Instruction Pointer
CS - Code Segment Register
DS - Data Segment Register
SS - Stack Segment Register
ES - Extra Segment Register

TYPE OF MEMORY REFERENCE	DEFAULT SEGMENT BASE	ALTERNATE SEGMENT BASE	OFFSET
Instruction Fetch	CS	None	IP
Stack Operation	SS	None	SP
Variable (except following)	DS	CS, ES, SS	Effective Address
String Source	DS	CS, ES, SS	SI
String Destination	ES	None	DI
BP Used As Base Register	SS	CS, DS, ES	Effective Address



4

Out. 2009

1. Microprocessador Intel 8086

Arquiteturas Intel 8086 / 8088 (16 bits)

AX - Accumulator	SP - Stack Pointer	IP - Instruction Pointer
BX - Base Register	BP - Base Pointer	CS - Code Segment Register
CX - Count Register	SI - Source Index Register	DS - Data Segment Register
DX - Data Register	DI - Destination Register	SS - Stack Segment Register
		ES - Extra Segment Register

FLAGS:

Bit 0 - CF Carry Flag - Set by carry out of msb
 Bit 2 - PF Parity Flag - Set if result has even parity
 Bit 4 - AF Auxiliary Flag - for BCD arithmetic
 Bit 6 - ZF Zero Flag - Set if result is zero
 Bit 7 - SF Sign Flag = msb of result
 Bit 8 - TF Single Step Trap Flag
 Bit 9 - IF Interrupt Enable Flag
 Bit 10 - DF String Instruction Direction Flag
 Bit 11 - OF Overflow Flag

Bits 1, 3, 5, 12-15 are undefined.

5

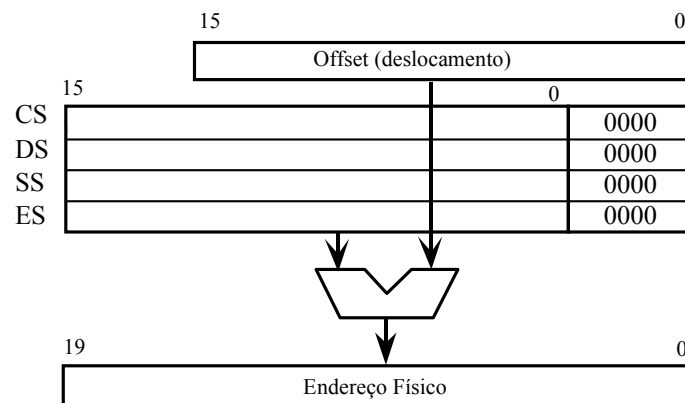
Out. 2009

1. Microprocessador Intel 8086

Arquiteturas Intel 8086 / 8088 (16 bits)

Endereçamento da memória...

Gerando 20 bits a partir de registradores de 16 bits



6

Out. 2009

1. Microprocessador Intel 8086

Arquiteturas Intel 8086 / 8088 (16 bits)

Endereçamento da memória...

Gerando 20 bits a partir de registradores de 16 bits

- Arquitetura de 16 bits
- Comunicação com a memória em 16 (8086) ou 8 (8088) bits
- Capacidade máxima de memória de 1 MByte
- 14 registradores (4 dado, 4 endereço, 4 segmento, ponteiro do programa, flags)
- endereço físico = segmento * 16 + deslocamento
- 85 instruções básicas... Listadas a seguir

7

Out. 2009

AAA	ASCII Adjust After Addition	AAD	ASCII Adjust Before Division
AAM	ASCII Adjust After Multiply	AAS	ASCII Adjust After Subtract
ADC	Add with Carry	ADD	Add
AND	Logical And	CALL	Call Procedure
CBW	Convert Byte to Word	CLC	Clear Carry Flag
CLD	Clear Direction Flag	CLI	Clear Interrupt Flag
CMC	Complement Carry Flag	CMP	Compare
CMP SB/W	Compare strings (Byte, Word)	CWD	Convert Word to Double
DAA	Decimal Adjust After Division	DAS	Decimal Adjust after Subtract
DEC	Decrement	DIV	Unsigned Divide
ESC	Escape (for FPU)	HLT	Halt
IDIV	Signed Divide	IMUL	Signed Multiply
IN	Input from Port	INC	Increment
INT	Software Interrupt	INTO	Interrupt on Overflow
IRET	Interrupt Return (16 bits)	IRETD	Interrupt Return (32 bits)
JAE, JNB	Jump if Above	JAE, JNB	Jump if Above or Equal
JB, JNAE	Jump if below	JBE, JNA	Jump if Below or Equal
JC	Jump on Carry	JCXZ	Jump if CX is Zero
JE, JZ	Jump if Equal (Zero)	JG, JNLE	Jump if Greater
JGE, JNL	Jump if Greater or Equal	JL, JNGE	Jump if Less
JLE, JNG	Jump if Less or Equal	JMP	Jump unconditionally
JNC	Jump if not Carry	JNE, JNZ	Jump if not Equal
JNP, JPO	Jump if Parity Odd	JO, JNO	Jump on Overflow, not Overf.
JP, JPE	Jump if Parity Even	JS, JNS	Jump on Sign, not Sign
LAHF	Load Flags into AH	LDS, LES	Load Far Pointer (DS, ES)
LEA	Load Effective Address	LOCK	Lock the Bus
LODSB/W	Load String (Byte, Word)	LOOP	Loop
LOOPE/Z	Loop while Equal	LOOPNE/Z	Loop while not Equal
MOV	Move Data	MOVS B/W	Move String (Byte, Word)
MUL	Unsigned Multiply	NEG	Two's Complement Negation
NOP	No Operation	NOT	One's Complement Negation
OR	Inclusive Or	OUT	Output to Port
POP	Pop from Stack	POPF	Pop Flags
PUSH	Push into Stack	PUSHF	Push Flags
RCL	Rotate with Carry Left	RCR	Rotate with Carry Right
REP	Repeat String	REPE/Z	Repeat while Equal (Zero)
REPNE/Z	Repeat while not Equal (Zero)	RET, RETF	Return from Procedure
ROL	Rotate Left	ROR	Rotate Right
SAHF	Store AH into Flags	SAL	Shift Arithmetic Left
SAR	Shift Arithmetic Right	SBB	Subtract with Borrow
SCASB/W	Scan String (Byte, Word)	SHL	Shift Left
SHR	Shift Right	STC	Set Carry Flag
STD	Set Direction Flag	STI	Set Interrupt Flag
STOSB/W	Store String (Byte, Word)	SUB	Subtract
TEST	Logical Compare	WAIT	Wait (coprocessor)
XCHG	Exchange	XLAT	Translate
XOR	Exclusive Or		

1. Microprocessador Intel 8086

Arquiteturas Intel 80x86

- IA 16 (8086, 80186, 80286)
 - Modo real, 16 bits
- IA 32 (80386, 80486, 80586)
 - Modo protegido, 32 bits
 - Modo virtual real, 16 bits
- IA 64
 - Itanium, 64 bits
- IA 32e, AMD64, x86-64, EM64T
 - Modo 64-bit
 - Modo de compatibilidade, 32 bits

1. Microprocessador Intel 8086

Arquiteturas Intel 8086

- Arquitetura de 16 bits
- Comunicação com a memória em 16 (8086) ou 8 (8088) bits
- Capacidade máxima de memória de 1 MByte
- 14 registradores (4 dado, 4 endereço, 4 segmento, ponteiro do programa, flags)
- endereço físico = segmento * 16 + deslocamento
- 85 instruções básicas
- co-processador: 8087 (67 instruções básicas)
- sem cache, sem memória virtual
- somente opera no modo real

1. Microprocessador Intel 8086

Arquiteturas Intel 80186

- Idêntico ao 8086
- Comunicação com a memória em 16 bits
- Capacidade máxima de memória de 1 MByte
- 14 registradores
- Endereço físico = segmento * 16 + deslocamento
- Sete instruções extras ($85 + 7 = 92$ instruções básicas)
- Co-processador: 80187 (idêntico ao 8087)
- Sem cache, sem memória virtual

1. Microprocessador Intel 8086

Arquiteturas Intel 80286

- Modos real (8086) e virtual (modo protegido)
- Comunicação com a memória em 16 bits
- Capacidade máxima de memória de 16 MByte
- 14 registradores (os do 8086)
- Endereço físico ou virtual
- 15 instruções extras ($92 + 15 = 107$ instruções básicas)
- Co-processador: 80287
- Sem cache
- Memória virtual segmentada

1. Microprocessador Intel 8086

Arquiteturas Intel 80386

- Modos real (8086), virtual (protegido) e virtual86
- Comunicação com a memória em 16 (SX) ou 32 bits (DX)
- Capacidade máxima de memória de 4 GByte
- 14 registradores (os do 8086, estendidos para 32 bits) e mais 2 registradores de segmento
- Endereço físico ou virtual
- 44 instruções extras ($107 + 44 = 151$ instruções básicas)
- Memória virtual segmentada e paginada (opcional)
- Co-processador: 80387 ($67 + 7 - 1 = 73$ instruções básicas)

13

Out. 2009

1. Microprocessador Intel 8086

Arquiteturas Intel 80386

	31	16	15		
EAX			AH	AL	acumulador
EBX			BH	BL	base
ECX			CH	CL	contador
EDX			DH	DL	dado
ESP				SP	ponteiro para pilha
EBP				BP	ponteiro base
ESI				SI	índice fonte
EDI				DI	índice destino
	31	16	15		0
EIP				IP	apontador de instruções
EF				FLAGS	flags

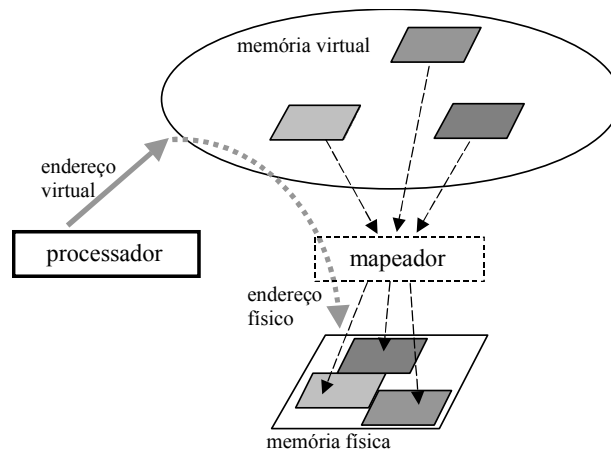
14

Out. 2009

1. Microprocessador Intel 8086

Arquiteturas Intel 80386

Gerência de memória virtual



1. Microprocessador Intel 8086

Arquiteturas Intel 80486

- Idêntico ao 386
- Modos real (8086), virtual e virtual86
- Comunicação com a memória em 32 bits
- Capacidade máxima de memória de 4 GByte
- 16 registradores (os do 80386, também em 32 bits)
- Endereço físico ou virtual
- 6 instruções extras ($151 + 6 = 157$ instruções básicas)
- Memória virtual segmentada e paginada (opcional)
- Co-processador: 80487 para 80486SX
integrado no 80486DX
- Com cache de 8 KByte

1. Microprocessador Intel 8086

Arquiteturas Intel 80586 - Intel Pentium

- Re-estruturação do 486
- Modos real (8086), virtual e virtual86
- Comunicação com a memória em 64 bits
- Capacidade máxima de memória de 4 GByte
- 16 registradores (os do 80386, também em 32 bits)
- Endereço físico ou virtual
- 5 instruções extras ($157 + 5 = 162$ instruções básicas)
- Memória virtual segmentada (sempre) e paginada (opcional)
- Co-processador: integrado
- Com cache de 16 KByte (2 x 8 KByte)

17

Out. 2009

1. Microprocessador Intel 8086

Arquiteturas Intel 80586 - Intel Pentium

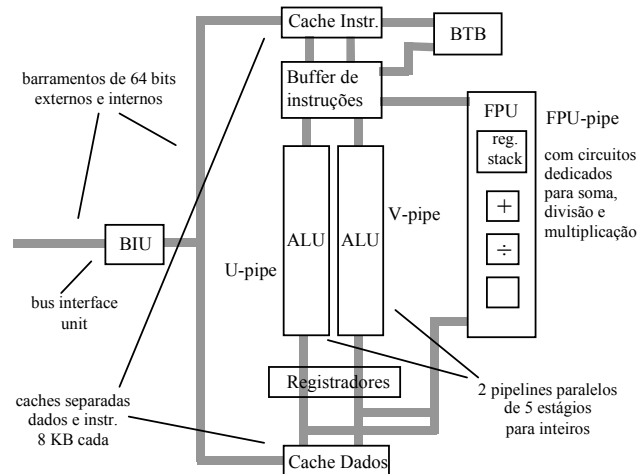
- 2 pipelines para inteiros, operando em paralelo
- cada pipeline inteiro consta de 5 estágios:
 - busca de instrução (a partir da cache de instruções),
 - decodificação de instrução,
 - geração de endereço,
 - execução,
 - escrita (write back).
- FPU também em pipeline (mas não em paralelo)
- Operação super-escalar: mais de uma instrução pronta em um ciclo de relógio

18

Out. 2009

1. Microprocessador Intel 8086

Arquiteturas Intel 80586 - Intel Pentium

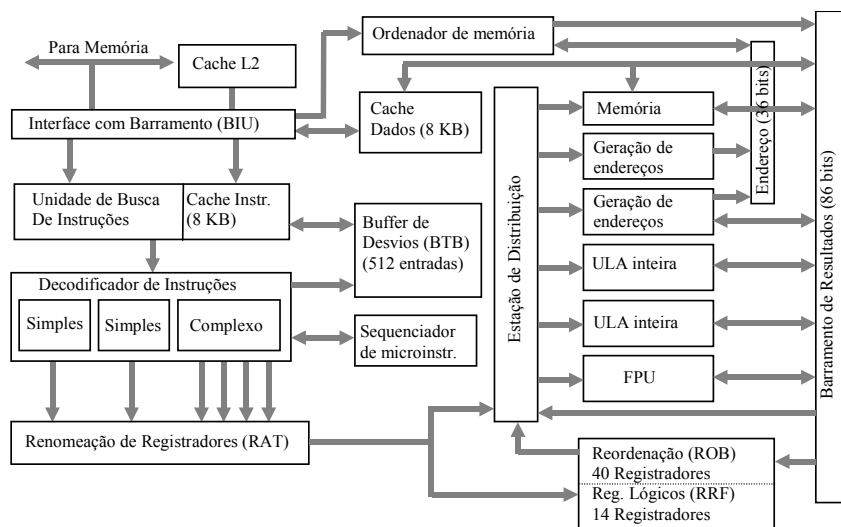


19

Out. 2009

1. Microprocessador Intel 8086

Arquiteturas Intel 80586 - Intel Pentium Pro



20

Out. 2009

2. Programação Intel 8086

Arquiteturas Intel 80x86 (16 bits)

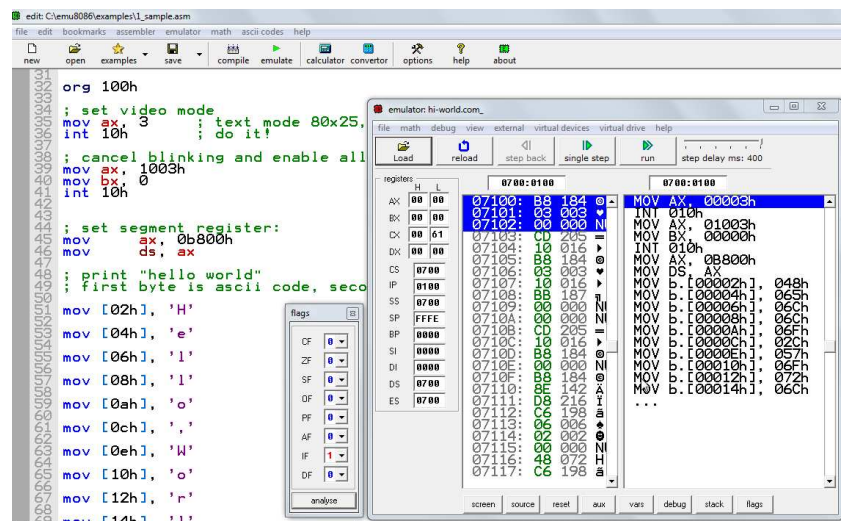


21

Out. 2009

2. Programação Intel 8086

Arquiteturas Intel 80x86 (16 bits)



22

Out. 2009

2. Programação Intel 8086

Arquiteturas Intel 80x86 (16 bits)



Dos Box
<http://www.dosbox.com/>

sourceforge FIND AND DEVELOP OPEN SOURCE SOFTWARE

[Find Software](#) [Develop](#) [Create Project](#) [Blog](#) [Site Support](#) [About](#)

SourceForge.net > Find Software > DOSBox DOS Emulator



DOSBox DOS Emulator by harekiet, qbix79

[Summary](#) [Files](#) [Support](#) [Develop](#)

DOSBox emulates a full x86 pc with sound and dos. Its main use is to run old dosgames on platforms which don't have dos(win2K/XP/linux/FreeBSD/Mac OS X)

Download Now!

DOSBox0.74-win32-install... (1.4 MB)

OR [View all files](#)

23

Out. 2009

2. Programação Intel 8086

Arquiteturas Intel 80x86 (16 bits)

>> **gcc -S hello.c** <<

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    printf("\n\nHello World!\n\n");
    system("PAUSE");
    return 0;
}
```

```
.file      "hello.c"
.def      __main; .scl 2; .type 32; .endef
.section .rdata,"dr"
LC0:
    .ascii "\12\12Hello World!\12\12\0"
LC1:
    .ascii "PAUSE\0"
    .text
.globl __main
.def      __main; .scl 2; .type 32; .endef
```

hello.s

```
_main:
    pushl    %ebp
    movl     %esp, %ebp
    subl     $8, %esp
    andl     $-16, %esp
    movl     $0, %eax
    addl     $15, %eax
    addl     $15, %eax
    shrl     $4, %eax
    sall     $4, %eax
    movl     %eax, -4(%ebp)
    movl     -4(%ebp), %eax
    call     __alloca
    call     __main
    movl     $LC0, (%esp)
    call     __printf
    movl     $LC1, (%esp)
    call     __system
    movl     $0, %eax
    leave
    ret
.def      _system; .scl 3; .type 32; .endef
.def      __printf; .scl 3; .type 32; .endef
```

24

Out. 2009

2. Programação Intel 8086

Arquiteturas Intel 80x86 (16 bits)

>> gcc hello.c <<

Incluindo código em assembly junto ao programa “C”

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    printf("\n\nHello World!\n\n");

    __asm__
    (
        "movw $0x0E41,%AX\n"
        "movw $0x0000,%BX\n"
        "int $0x10\n"
    );
    system("PAUSE");
    return 0;
}
```

25

Out. 2009

2. Programação Intel 8086

Arquiteturas Intel 80x86 (16 bits)

Interrupções de Hardware – IBM PC (AT 286)

Linha de Pedido	Vetor (dec.)	Utilização	barramento ISA 16
IRQ0	8	relógio de tempo real (timer tick)	
IRQ1	9	teclado	
IRQ2	10	acesso a PIC subordinada (cascateamento)	
IRQ8	112	atualização da hora do dia no BIOS	
IRQ9	113	vídeo no PS/2, uso geral em um PC (placa de rede)	
IRQ10	114	uso geral (tipicamente controlador de CD)	
IRQ11	115	uso geral (tipicamente placa SCSI)	
IRQ12	116	mouse no PS/2 - uso geral em um PC	
IRQ13	117	coprocessador aritmético	
IRQ14	118	controlador de disco rígido (controladora IDE primária)	
IRQ15	119	uso geral (controladora secundária de disco)	
IRQ3	11	placa serial COM2 e COM4	
IRQ4	12	placa serial COM1 e COM3	
IRQ5	13	uso geral (tipicamente placa de som)	
IRQ6	14	controladora de disquete	
IRQ7	15	controladora de impressora (LPT1)	

26

Out. 2009

2. Programação Intel 8086

Arquiteturas Intel 80x86 (16 bits)

Interrupções de Software – BIOS

Serviços do BIOS - vetores de 10H a 1FH e 40H a 5FH

- 10H - serviços de vídeo
- 11H - configuração do equipamento
- 12H - tamanho de memória
- 13H - serviços de disco
- 14H - comunicações (portas seriais RS232)
- 15H - serviços de cassete (PC) ou extensões de E/S (AT)
- 16H - serviços de teclado
- 17H - serviços de impressora
- 18H - ativação do interpretador BASIC
- 19H - tratamento de bootstrap
- 1AH - hora do dia
- 1BH - tratamento de Ctrl-Break
- 1CH - relógio de tempo real
- 67H - serviços de memória expandida

27

Out. 2009

2. Programação Intel 8086

Arquiteturas Intel 80x86 (16 bits)

Interrupções de Software – BIOS

Serviços do DOS - vetores de 20H a 3FH

- 20H - término de programa
- 21H - serviços do DOS
- 24H - tratamento de erro
- 25H - leitura absoluta de disco
- 26H - escrita absoluta de disco
- 27H - término de programa residente
- 2AH - serviços de rede DOS
- 2EH - comando de execução DOS
- 2FH - controle de spooler de impressão DOS
- 33H - serviços do driver de mouse

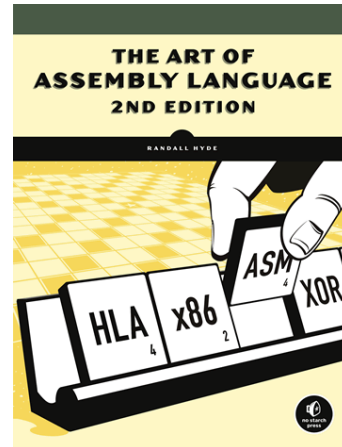
28

Out. 2009

2. Programação Intel 8086

Arquiteturas Intel 80x86 (16 bits)

Programação em
Linguagem de Máquina para o 8086
Interrupções de HW e SW



Ref.: <http://webster.cs.ucr.edu/AoA/>
http://webster.cs.ucr.edu/AoA/DOS/pdf/0_AoAPDF.html



INFORMAÇÕES SOBRE A DISCIPLINA

USP - Universidade de São Paulo - São Carlos, SP
ICMC - Instituto de Ciências Matemáticas e de Computação
SSC - Departamento de Sistemas de Computação

Prof. Fernando Santos OSÓRIO

Web institucional: <http://www.icmc.usp.br/ssc/>

Página pessoal: <http://www.icmc.usp.br/~fosorio/>

E-mail: fosorio [at] icmc. usp. br ou fosorio [at] gmail. com

Disciplina de Organização de Computadores I / Eng. Comp.

Estagiário PAE: Maurício A. Dias

Web disciplina: <http://wiki.icmc.usp.br/index.php/Ssc-610>

> Programa, Material de Aulas, Critérios de Avaliação,

> Lista de Exercícios, Trabalhos Práticos, Datas das Provas