Set Instruksi
dan
Format Intruksi

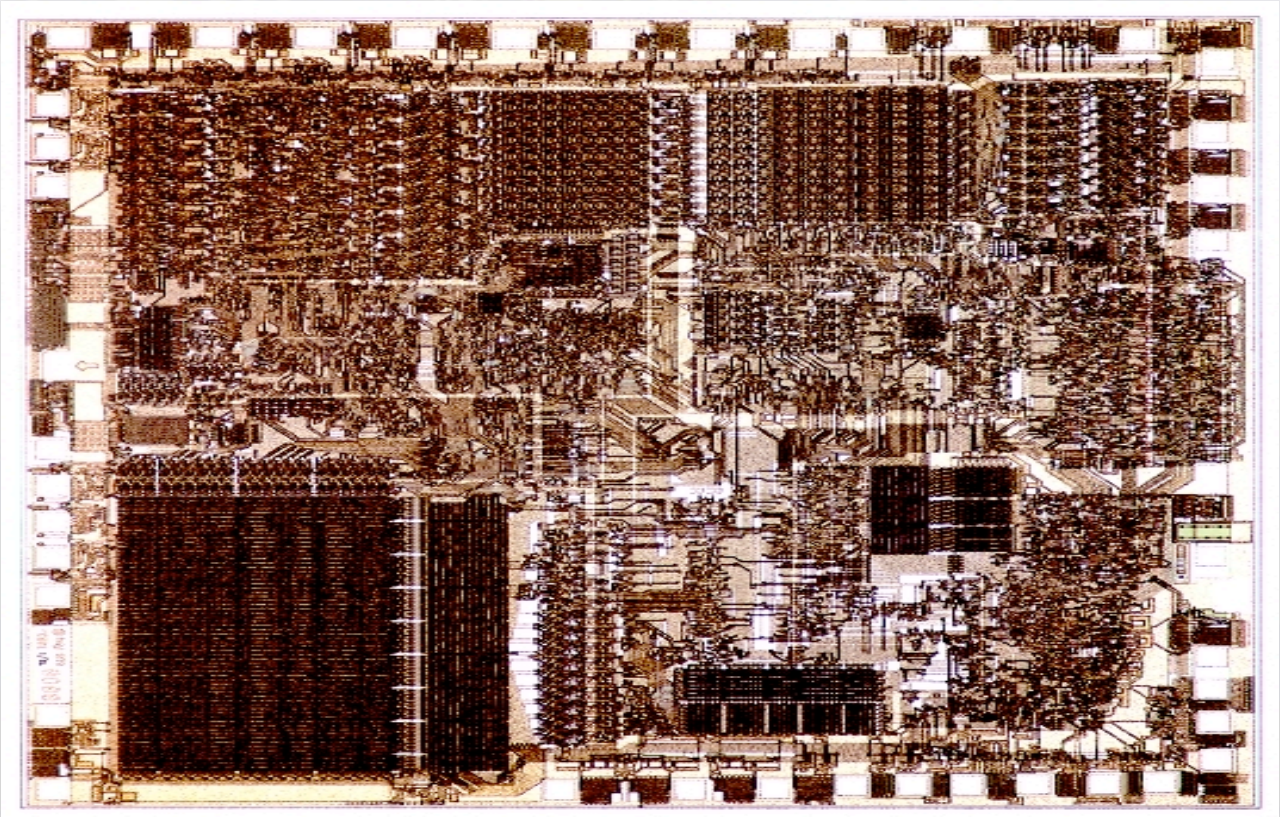
Judul Pokok Bahasan

Outline :

- High performance 16 bit
- Format Instruksi
- Set Instruksi
- Instruksi Data Transfer
- Instruksi Aritmatik
- Instruksi Logika
- Instruksi Lompatan

High Performance 16-bit Microprocessor

1978: 8086-8088



Picture Courtesy: Intel Corporation

Microprocessor History

1st Generation

- Early 1970's
- 4 - bit - nibble data
- Low Performance
- Limited System Capabilities
- Low Cost
- Like PPS 4
- Special Purpose Applications
- Uses: Calculators, Toys

2nd Generation

- 1973-74
- 8-bit - 1 byte wide
- Higher Performance
- Larger System Capabilities
- Greater ease of Programming
- Like 8085
- Uses:Electronic Instruments, Cash Registers, Printers

3rd Generation

- Mid 1970's
- 16 - bit
- Higher Performance
- Special and general purpose microcomputer applications
- Like **8086**
- Uses: Electronic instruments, Word Processing systems

The world's most popular architecture

- An extension of 8080 → 8 - bit Microprocessor from Intel Corporation and cousin of Z80.
- Mainly evolved due to requirements for larger memory systems.
- With Vast Instructions set
- **The World's most popular architecture 8086 -** 1st 16-bit microprocessor Introduced by Intel Corporation in 1978.

MIPS VS 8086

MIPS

- Introduced in 1985
- Separate operands for source and destination
- Not possible to have one of the operands in memory
- General Instruction format opcode destination, source1, source2

8086

- Introduced in 1978 by Intel
- One of the operands acts as both source and destination
- One of the operands can be in memory
- General Instruction format

opcode destination/source1, source2

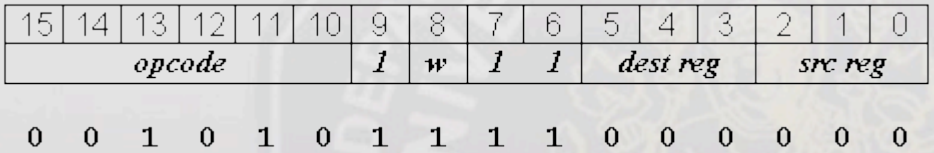
Instruction Format

16 - bit Instruction mode

Opcode 1-2 bytes	MOD REG-R/M 0-1 bytes	Displacement 0-1 bytes	Immediate 0-2 bytes

- **Opcode** selects the operation performed by the microprocessor
- **MOD** specifies the addressing mode for the selected instruction, also determines if displacement is present or not.
- **Displacement** specifies the amount of displacement for the address
- **Immediate** specifies the address in case of immediate addressing

Format of register-to-register instruction:



reg	register
000	AX
001	CX
010	DX
011	BX
100	SP
101	BP
110	SI
111	DI

opcode = 001010 means "Subtract"

w =1 means "word" (16 bits); 0 means "byte"

dest reg = address of destination register: 000 = AX

src reg = address of source register: 000 = AX

Instruction Set

The instructions in 8086, can be classified into 13 groups based on their functions.

1. Data Transfer Instructions
2. Arithmetic Instructions
5. Logic Instructions
6. Compare Instruction
7. Jump Instructions
8. Loop Instructions
9. Shift Instructions
10. Rotate Instructions
11. Flag Control Instructions
12. String Instructions
13. Input / Output Instructions
14. Interrupt Instructions
15. Subroutine and Subroutine handling instructions.

Data Transfer Instructions

Move data between its Internal registers or between an internal register and a storage location in memory

- MOV target,source

  target,source – reg,reg / reg,mem / mem,reg / mem,imme / reg,imme

- XCHG target,source

  target,source – reg,reg / reg,mem / mem,reg

Data Transfer Instructions

MOV	Move byte or word to register or memory
IN, OUT	Input byte or word from port, output word to port
LEA	Load effective address
LDS, LES	Load pointer using data segment, extra segment
PUSH, POP	Push word onto stack, pop word off stack
XCHG	Exchange byte or word
XLAT	Translate byte using look-up table

Arithmetic Instructions

Extensive complement of arithmetic instructions

- Addition Instructions - ADD, ADC, INC, AAA, DAA
- Subtraction Instructions - SUB, SBB, DEC, NEG, AAS, DAS
- Multiplication Instructions - MUL, IMUL, AAM
- Division Instructions - DIV, IDIV, AAD, CBW, CWD

ADD dest, source SUB dest, source

dest, source – reg , reg / reg, mem / mem, reg / mem, imme / reg, imme

INC dest, DEC dest, NEG dest

dest – reg / mem

Arithmetic Instructions

ADD, SUB	Add, subtract byte or word
ADC, SBB	Add, subtract byte or word and carry (borrow)
INC, DEC	Increment, decrement byte or word
NEG	Negate byte or word (two's complement)
CMP	Compare byte or word (subtract without storing)
MUL, DIV	Multiply, divide byte or word (unsigned)
IMUL, IDIV	Integer multiply, divide byte or word (signed)
CBW, CWD	Convert byte to word, word to double word (useful before multiply/divide)

Logic Instructions

NOT	Logical NOT of byte or word (one's complement)
AND	Logical AND of byte or word
OR	Logical OR of byte or word
XOR	Logical exclusive-OR of byte or word
TEST	Test byte or word (AND without storing)

Note: All instructions that have two operands, cannot have both operands to be memory locations.

Compare and Jump Instructions

Compare Instructions

Instruction to compare two 8 bit or 16 bit numbers

- CMP

Jump Instructions

- Unconditional
  - JMP
- Conditional
  - Jcc

Loop and Input/Output Instructions

Three instructions specifically designed for implementing loops.

- LOOP
- LOOPE/LOOPZ
- LOOPNE/LOOPNZ

Input/Output instructions

- IN
- OUT

Flag Control Instructions

Flag control instructions either monitor the status of executing instructions or control options available in its operation.

- LAHF
- SAHF
- CLC
- STC
- CMC
- CLI
- STI

Interrupt Instructions

Number of instructions for processing interrupts

- CLI
- STI
- INT n
- IRET
- INTO
- HLT
- WAIT

String Handling Instructions

Equipped to handle string operations

- MOVS
- MOVSB
- MOVSW
- CMPS
- SCAS
- LODS
- STOS

Rotate, Shift and Subroutine and Subroutine Handling Instructions

Shift Instructions

- Logical shift

  SHL, SHR

- Arithmetic shift

  SAL, SAR

Rotate Instructions

- ROL, ROR, RCL, RCR

- CALL and RET instructions
- PUSH and POP instructions

The Modern Microprocessor

- 8086 extended further to 8088, 16-bit Microprocessor, whose increased memory size and additional instructions led to more sophisticated applications for Microprocessor.
- Microprocessors were called CISC - Complex instruction set computers, because of the number and complexity of instructions.
- Very soon the memory offered by the 8088 became insufficient, in 1983 leading to the introduction of 80286, shortly before 80186/80188 were introduced. All being 16 bit.
- Demand yet increasing, in 1986, 32 -bit microprocessor 80386 was introduced followed by 80486 in 1989..

Referensi

- Brey, Barry, B., The Intel Microprocessors 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, and Pentium ProProcessor Architecture, Programming, and Interfacing, Fourth Edition, PHI Inc, USA, 1997. and Five Edition, 2003
- Brey, Barry, B., 8086/8088, 80286, 80386, and 80486 Assembly Language, Programming, Macmillan Publising Company, USA, 1994.
- Leventhal L.A., Introduction to Microprocessor : Software, Hardware, Programming, Phi Inc., 1978.
- Hall D.V., Microprocessor Interfacing : Programming and Hardware, McGraw-Hill, Singapore, 1986.
- Ananta, C., William JB., Frank Fox, Design of High-performance microprocessor circuit, IEEE Press, 2001
- Douglas V. Hall, Microprocessor and Interfacing: Programming and Hardware, McGraw-Hill, edition 2, 1991
- James A., Kenneth CM, Microcomputer Hardware, Software, and Troubleshooting for Engineering and Technology, Prentice Hall, 2000
- http://www.intel.com/
- http://www.hardwarebible.com/Microprocessors/8086.htm