



## Tugas II

Disusun Oleh:

Nama : Hasna Azizah

NIM : L200160168

Makul / Kelas : Organisasi dan Arsitektur Komputer / C

Dosen : Bana Handaga

UNIVERSITAS MUHAMMADIYAH SURAKARTA  
SURAKARTA  
2017

Soal: Cari daftar instruction set untuk processor INTEL dan ARM (android)!

Jawab :

- **DAFTAR INSTRUCTION SET UNTUK PROCESSOR INTEL**

**A. Set Instruksi Pada Microprocessor 8088**

**1. Transfer Data**

Set instruksi 8088 mencakup 14 transfer data yang menggerakkan byte atau kata data antara memori dan register selain juga antara akumulator dan port I/O.

- a) General Purpose (Umum) : MOV, PUSH, POP, XCHG, SXLAT
- b) Input/Output : In, Out
- c) Address Transfer (Pemindahan Alamat) : LEA, KDS, KES
- d) Flag Transfer (Pemindahan Flag) : LAHF, SAHF, PUSHF, POPF

**2. Arithmetic Instructions (Instruksi Perhitungan)**

Microprocessor 8088 mampu melakukan penambahan, pengurangan, pengalian dan pembagian data selain bytes dan kata. Sistem ini menambah dan mengurangi dengan menggunakan bytes atau kata yang bertanda atau tak bertanda dan data BCD atau ASCII.

- a) Addition (Penjumlahan) : ADD, ADC, AAA, DAA, INC
- b) Substraction (Pengurangan) : SUB, SBB, AAS, DAS, DES, NEG, CMP
- c) Multiplication (Perkalian) : MUL, IMUL, AAM
- d) Division (Pembagian) : DIV, IDIV, AAD
- e) Sign Extension : CBW, CWD

**3. Manipulasi Bit**

Instruksi ini mencakup operasi logika, shift dan rotasi.

- a) Logical (Logika) : AND, OR, XOR, NOT, TEST
- b) Shift (Geser) : SAL/SHL, SAR, SHR
- c) Rotate (Gulung) : ROL, ROR, RCL, RCR

**4. Instruksi String**

Instruksi string digunakan untuk memanipulasi string data dalam memori. Setiap string tersusun baik dari bytes maupun kata dan hingga mencapai 64 K bytes panjangnya

- a) Direction : CLD, STD
- b) Repeat Prefixes : REP, REPE/REPZ, REPNE/REPNZ,
- c) Move : MOVSB, MOVSW
- d) Compare : CMPS, CMPSB, CMPSW
- e) Scan : CSAS, SCASB, SCASW
- f) Load and Store : LODS, LODSB, LODSW, STOS, STOSB, STOSW

## 5. Transfer Program

Instruksi transfer program mencakup *jump*, *call*, dan *return*, instruksi yang sudah kita kenal dalam 8085.

- a) Unconditional transfer (Pemindahan) : CAL, RET, JMP
- b) Conditional transfer (Pemindahan Bersyarat) : JA/JNBE, JAE/JNB, JB/JNAE/JC, JBE/JNA, JCXZ, JE/JZ, JG/JNLE, JGF/JNL, JL/JNGE, JLE/JNG, JNC, JNE/JNZ, JNO, JNP/JPO, JNS, JO, JP/JPE, JS
- c) Loop/Jump : LOOP, LOOPE/LOOPZ, LOOPNE/LOOPNZ

## 6. Kontrol Prosesor

Instruksi kontrol prosesor memungkinkan dan tidak memungkinkan interupsi, memodifikasi bit flag dan mensinkronisasi kejadian eksternal.

- a) Flag Operations : CLC, STC, CMC, CLD, STD, CLI, STI,
- b) External Synchronization

## 7. Interrupt : INT, INTO, IRET

## SEKILAS INTEL 8088

Intel 8088 adalah prosesor mikro buatan Intel berbasis pada 8086, dengan 16-bit register dan menggunakan 8-bit external data bus. Intel 8088 merupakan prosesor yang digunakan pada IBM PC. 8088 ditargetkan pada sistem yang ekonomis, diikuti oleh penggunaan desain 8-bit. Jalur bus yang lebar dalam circuit boards masih sangatlah mahal ketika ini di luncurkan. Queue yang unggul dari 8088 adalah 4 bytes, sebagai penggunaan dalam 8086 6 bytes. 8088 termasuk keturunan dari 80188, 80288, 80186, 80286, 80386, 80486, dan 80388, microcontroller seperti yang masih digunakan sekarang. Clone yang populer dengan menggunakan 8088 adalah Model D, dimana tombol pilihan dapat berjalan pada clock 4.77 MHZ atau 7.16 MHZ.

- **DAFTAR INSTRUCTION SET UNTUK PROCESSOR ARM**

## B. Set Instruksi Pada Microprocessor 8088

### 1. Register

Dalam arsitektur ARM, memori adalah byte-addressable, menggunakan alamat 32-bit, dan register prosesor memiliki panjang 32-bit. Dua panjang operand digunakan dalam memindahkan data antara memori dan register prosesor : byte (8 bit) dan word (32 bit).

#### Struktur Register

- Register prosesor yang digunakan oleh program aplikasi ditampilkan terdapat enam belas register 32 bit berlabel R0 hingga R15 yang terdiri dari lima belas general-purpose register (R0 hingga R14) dan register program counter (PC), R15, yang terdiri dari 15.
- General purpose register dapat menyimpan alamat memori atau operand data. Current program status register (CPSR), atau cukup disebut register status, menyimpan condition code flag (N, Z, C, V), interrupt disable flag, dan bit mode prosesor.

### 2. Instruksi Akses Memory

Fitur yang membedakan dan agak tidak biasa dari prosesor ARM adalah semua instruksinya dieksekusi secara conditional, tergantung pada kondisi yang ditetapkan pada instruksi tersebut. Instruksi tersebut dieksekusi hanya jika keadaan saat ini dari condition code flag prosesor memenuhi kondisi yang ditetapkan dalam bit  $b_{31-28}$  dari instruksi tersebut. Jika tidak prosesor melanjutkan ke instruksi berikutnya salah satu kondisi tersebut digunakan untuk mengindikasikan bahwa instruksi tersebut selalu dieksekusi.

#### Mode Pengalamatan Memori ada 2:

- a) Metode dasar untuk mengalami operand memori adalah membangkitkan effective address, EA, dari operand tersebut dengan menambahkan offset bertanda keisi base register  $R_n$ , yang ditentukan dalam instruksi. Besarnya offset tersebut dapat berupa nilai immediate yang terdapat dalam 12 bit low order instruksi atau isi dari register

ketiga, Rm, yang dinamai dengan 4 bit low order tanda arah offset terdapat dalam field OP-code.

b) Operand Load/Store Multiple

Selain instruksi load dan store untuk operand tunggal, terdapat 2 instruksi untuk me-load dan menyimpan banyak operand. Instruksi itu disebut instruksi transfer block. Sub set apapun dari general purpose register load atau disimpan. Hanya operand word yang diperbolehkan, dan OP code yang digunakan dalam load multiple dan store multiple. Operand memori harus berada dalam lokasi word yang berurutan.

3. Instruksi Move Register

Menggunakan Rd , Rm

4. Instruksi Aritmatika

Ekspresi bahasa assembly dasar untuk instruksi aritmatika adalah

- Op code                      Rd, Rn, Rm

Dimana operasi yang ditetapkan oleh OP code dilakukan menggunakan operand dalam general-purpose register Rn dan Rm. Hasilnya diletakkan dalam register Rd.

Misalnya, instruksi

- ADD                              R0, R2, R4

Menjalankan operasi

$R0 \leftarrow [R2] + [R4]$

Dan instruksi SUB              R0, R6, R5 menjalankan operasi  $R0 \leftarrow [R6] - [R5]$

5. Instruksi Logika

Operasi logika AND, OR, XOR, dan Bit-clear diimplementasikan oleh instruksi OP code AND, ORR, EOR, dan BIC. Kode tersebut memiliki format yang sama dengan instruksi aritmatika. Instruksi AND Rd, Rn, Rm menjalankan operasi

$Rd \leftarrow [Rn] \wedge [Rm]$

Yang merupakan bitwise logical AND antara operand dalam register Rn dan Rm. Misalnya, jika register R0 berisi pola hexadesimal 02FA62CA dan R1 berisi pola 0000FFFF, maka instruksi AND R0, R0, R1 akan menyebabkan pola 000062CA diletakkan dalam register R0.

6. Instruksi Branch

Instruksi branch CONDITIONAL berisi offset 24-bit, 2'-complement, bertanda yang ditambahkan ke isi ter-update Program Counter untuk menghasilkan alamat target branch.

Instruksi Branch dieksekusi dengan cara yang sama seperti instruksi ARM yang lain, yaitu dieksekusi hanya jika keadaan terbaru condition code flag berhubungan dengan kondisi ditetapkan dalam field condition instruksi tersebut.