**TUGAS**

**INSTRUCTION SET**

**Disusun Guna Memenuhi Tugas**

**Organisasi dan Arsitetur Komputer**

**Pengampu : Bana Handaga, Dr. Ir, M.T.**



**Oleh:**

**ARI PURNOMO JATI**

**L 200150116**

**PROGRAM STUDI INFORMATIKA**

**UNIVERSITAS MUHAMMADIYAH SURAKARTA**

**2017**

**INSTRUCTION SET**

Set Instruksi (Instruction Set atau Instruction Set Architecture (ISA)) didefinisikan sebagai suatu aspek dalam arsitektur komputer yang dapat dilihat oleh para pemrogram. Secara umum, ISA ini mencakup jenis data yang didukung, jenis instruksi yang dipakai, jenis register, mode pengalamatan, arsitektur memori, penanganan interupsi, eksepsi, dan operasi I/O eksternalnya (jika ada). ISA merupakan sebuah spesifikasi dari kumpulan semua kode-kode biner (opcode) yang diimplementasikan dalam bentuk aslinya (native form) dalam sebuah desain prosesor tertentu. Kumpulan opcode tersebut, umumnya disebut sebagai bahasa mesin (machine language) untuk ISA yang bersangkutan. ISA yang populer digunakan adalah set instruksi untuk chip Intel x86, IA-64, IBM PowerPC, Motorola 68000, Sun SPARC, DEC Alpha,dan lain-lain. ISA kadang-kadang digunakan untuk membedakan kumpulan karakteristik yang disebut di atas dengan mikroarsitektur prosesor, yang merupakan kumpulan teknik desain prosesor untuk mengimplementasikan set instruksi (mencakup microcode, pipeline, sistem cache, manajemen daya, dan lainnya).

Pada beberapa mesin, semua instruksi memiliki panjang yang sama, pada mesin-mesin yang lain mungkin terdapat banyak panjang berbeda. Instruksi-instruksi mungkin lebih pendek dari, memiliki panjang yang sama seperti, atau lebih panjang dari panjang word. Membuat semua instruksi memiliki panjang yang sama lebih muda dilakukan dan membuat pengkodean lebih mudah tetapi sering memboroskan ruang, karena semua instruksi dengan demikian harus sama panjang seperti instruksi yang paling panjang.

**JENIS JENIS INSTRUKSI**

1. Data Processing (Pengolahan Data) : Instruksi-instruksi aritmetika dan logika. Instruksi aritmetika memiliki kemampuan untuk mengolah data numeric, sedangkan instruksi logika beroperasi pada bit-bit word sebagai bit bukan sebagai bilangan. Operasi-operasi tersebut dilakukan terutama untuk data di register CPU.
2. Data Storage (Penyimpanan Data): Instruksi-instruksi memori. Instruksi-instruksi memori diperlukan untuk memindah data yang terdapat di memori dan register.
3. Data Movement (Perpindahan Data): Instruksi I/O. Instruksi-instruksi I/O diperlukan untuk memindahkan program dan data ke dalam memori dan mengembalikan hasil komputansi kepada pengguna.
4. Control (Kontrol): instruksi pemeriksaan dan percabangan. Instruksi-instruksi control digunakan untuk memeriksa nilai data, status komputansi dan mencabangkan ke set instruksi lain

ISA yang diimplementasikan dalam bentuk perangkat keras :

Alpha AXP (DEC Alpha) ARM (Acorn RISC Machine) (Advanced RISC Machine now ARM Ltd) IA-64 (Itanium/Itanium 2) MIPS Motorola 68k PA-RISC (HP Precision Architecture) IBM POWER IBM PowerPC SPARC SuperH (Hitachi) System/360 Tricore (Infineon) Transputer (STMicroelectronics) VAX (Digital Equipment Corporation) x86 (IA-32, Pentium, Athlon) (AMD64, EM64T) ISA yang diimplementasikan dalam bentuk perangkat lunak lalu dibuat perangkat kerasnya :

p-Code (UCSD p-System Version III on Western Digital Pascal Micro-Engine) Java virtual machine (ARM Jazelle, PicoJava) FORTH ISA yang tidak pernah diimplementasikan dalam bentuk perangkat keras :

SECD machine ALGOL Object Code Elemen - elemen dari instruction Instruction terdiri dari beberapa elemen yaaitu :

- Operation Code (Op Code) - kode perintah operasi - Source Operand reference - Operand penampung nilai yang akan diproses. - Result Operand reference - Operand penampung nilai hasil proses. - Next Instruction Reference - Penghubung ke instruksi berikutnya. Operand merupakan obyek dari suatu Op code, operand biasanya ditampung pada salah satu tempat penyimpanan berikut: - Main memory (or virtual memory or cache) - CPU register - I/O device

Mode Pengalamatan (Addressing Mode) untuk Operand Terdapat beberapa mode pengalamatan operand, antara lain: - Immediate - Direct - Indirect - Register

Mode Immediate Addressing Merupakan mode pengalamatan operand secara langsung, pada mode ini operand merupakan bagian dari instruction. Operand merupakan area alamat (address field) dari suatu nilai yang akan diproses

Contoh: ADD 5 Keterangan: Tambahkan nilai 5 dengan nilai di register accumulator dan simpan hasilnya di register accumulator. Karakteristik mode immediate: - No memory reference to fetch data tidak memakai referensi memory untuk mengambil data - Fast : cepat - Limited range : terbatas dalam jangkauan nilai Mode Direct Addressing Merupakan mode pengalamatan operand dimana area alamat (address field) berisi alamat dari suatu nilai yang akan diproses. Effective Address (EA) = Address field (A) EA = A Contoh: ADD A Keterangan: - Cari di memory pada alamat A untuk operand (Look in memory at address A for operand). - Tambahkan isi yang ada pada alamat A dengan nilai di register accumulator dan simpan hasilnya di register accumulator. (Add contents of cell A to accumulator). Karakteristik: - Single memory reference to access data : Menggunakan memory untuk mengakses data - No additional calculations to work out effective address : Tidak memerlukan kalkulasi untuk mendapatkan effective address - Limited address space : Address space yang terbatas. Mode Indirect Addressing Merupakan mode pengalamatan operand dimana area alamat (address field) berisi alamat dari suatu alamat yang akan menunjukkan alamat dari suatu nilai yang akan diproses. (Memory cell pointed to by address field contains the address of (pointer to) the operand) EA = ( A ) Keterangan: - Cari di memory alamat A, cari alamat yang tertulis pada A untuk operand ( Look in A, find address (A) and look there for operand ). Contoh: ADD ( A ) Keterangan: - Tambahkan isi dari cell yang alamatnya ditunjukkan oleh isi yang terdapat pada A dengan nilai yang ada di register accumulator dan simpan hasilnya di register accumulator. (Add contents of cell pointed to by contents of A to accumulator). Karakteristik: - Large address space : memerlukan space address yang besar. - May be nested, multilevel, cascaded : Dapat dibuat nested (bersarang), multilevel dan cascade (bertumpuk). - Multiple memory accesses to find operand : pengaksesan memory yang multiple untuk

mendapatkan operand sehingga mengakibatkan proses mode ini agak lebih lambat. Mode Register Addressing Merupakan mode pengalamatan operand dimana operand yang akan diproses ditampung/disimpan dalam register yang namanya ditulis di area alamat (address field). (Operand is held in register named in address field) EA = R Karakteristik: - Limited number of registers : terbatas pada jumlah register yang hanya sedikit - Very small address field needed : karena addres yang kecil pada register, maka: - Shorter instructions : instruksinya lebih pendek - Faster instruction fetch : pemasukan data lebih cepat - No memory access : tidak memerlukan akses ke memory - Very fast execution : eksekusi sangat cepat - Very limited address space : tetapi space alamat (address) sangat terbatas