**Nama : Dian Puspitasari**

**NIM : L200130119**

**Kelas : A**

**Tugas 02 dan Tugas 03**

**Kelompok Set Instruksi**

1. **Data processing: Arithmetic dan Logic Instructions**

Data Processing (Pengolahan Data) adalah : 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.

1. Operasi set instruksi untuk aritmatika :
   * ADD                 : penjumlahan
   * SUBTRACT     : pengurangan
   * MULTIPLY      : perkalian
   * DIVIDE            : pembagian
   * ABSOLUTE
   * NEGATIVE
   * DECREMENT
   * INCREMENT
2. Operasi set instruksi untuk operasi logika :
   * AND, OR, NOT, EXOR     : melakukan operasi logika tertentu terhadap bit
   * COMPARE                         : melakukan perbandingan logika.
   * TEST                                   : menguji kondisi tertentu.
   * SHIFT                                 : melakukan penggeseran bit-bit operand
   * ROTATE                             : melakukan pemutaran bit-bit operand
3. **Data storage: Memory instructions**

Data Storage (Penyimpanan Data) adalah : instruksi-instruksi memori. Instruksi-instruksi memori diperlukan untuk memindah data yang terdapat di memori dan register.

Operasi set instruksi untuk pemindahan data :

* MOVE                        : memindahkan word atau blok dari sumber ke tujuan
* STORE                       : menyimpan word dari prosesor ke memori.
* LOAD                         : memuat word dari memori ke prosesor.
* EXCHANGE               : menukar isi sumber ke tujuan.
* CLEAR / RESET         : memindahkan word 0 ke tujuan.
* SET                             : memindahkan word 1 ke tujuan.
* PUSH                          : memindahkan word dari sumber ke bagian paling atas stack.
* POP                             : memindahkan word dari stack teratas ke tujuan..

1. **Data Movement: I/O instructions**

Data Movement (Perpindahan Data) adalah : instruksi I/O. Instruksi-instruksi I/O diperlukan untuk memindahkan program dan data ke dalam memori dan mengembalikan hasil komputansi kepada pengguna.

Operasi set instruksi Input / Ouput :

* INPUT              : memindahkan data dari pernagkat I/O tertentu ke tujuan
* OUTPUT          : memindahkan data dari sumber tertentu ke perangkat I/O
* START I/O       : memindahkan instruksi ke prosesor I/O untuk mengawali operasi I/O
* TEST I/O          : memindahkan informasi dari sistem I/O ke tujuan

1. **Control: Test and branch instructions**

Control (Kontrol) adalah : instruksi pemeriksaan dan percabangan. Instruksi-instruksi control digunakan untuk memeriksa nilai data, status komputansi dan mencabangkan ke set instruksi lain.

Operasi set instruksi untuk pemindahan kontrol :

* JUMP                                   : pemindahan tidak bersyarat dan memuat PC dengan   alamat tertentu.
* JUMP BERSYARAT           : menguji persyaratan tertentu danmemuat PC dengan  alamat tertentu atau tidak melakukan apa tergantung dari  persyaratan.
* JUMP SUBRUTIN              : melompat ke  alamat tertentu.
* RETURN                             : mengganti isi PC dan register lainnya yang berasal dari lokasi tertentu.
* EXECUTE                           : mengambil operand dari lokasi tertentu dan mengeksekusisebagai instruksi
* SKIP                                     : menambah PC sehingga melompati instruksi   
                                                      berikutnya.
* SKIP BERSYARAT            : melompat atau tidak melakukan apa-apa berdasarkan pada persyaratan
* HALT                                : menghentikan eksekusi program.
* WAIT (HOLD)                  : melanjutkan eksekusi pada saat persyaratan dipenuhi.
* NO OPERATION              : tidak ada operasi yang dilakukan.

**Pengaturan Prioritas Interupsi dalam Sistem Komputer**

**Interupsi atau interrupt adalah** suatu permintaan khusus pada mikroprocessor untuk melakukan sesuatu, jika terjadi interupsi maka komputer akan menghentikan dahulu apa yang sedang dikerjakan dan melakukan apa yang di minta oleh yang menginterupsi, setelah selesai maka aliran program akan kembali ke pernyataan program sebelum terjadinya interupsi. Interupsi merupakan sub rutin yang sudah tersedia dalam memori komputer.

1. **Jenis Interupsi**

**a.      Internal HW interruptions :**

1)   Ditimbulkan/digenerasi oleh peristiwa tertentu yang terjadi pada waktu/selama eksekusi program.

2)   Diatur oleh HW dan tidak dapat dirubah.

3)   Contoh : tipe interrupt untuk counter clock internal; HW call interrupt ini untuk memaintance “time to date.”

**b.      External HW interrupstion**

1)   Ditimbulkan/digenerasi oleh devais peripheral, seperti keyboard, printers, mouse, dsb.

2)   Biasa juga ditimbulkan/digenerasi oleh Co-processor

3)   Tidak mungkin mendeaktivekan

4)   Tidak dikirim langsung ke CPU, melainkan ke IC yang memiliki fungsi untuk menghandle secara eksklusive interrupts ini.

1. **Software interruption :**

* Diaktifkan langsung oleh assembler melalui sejumlah interuosi  
  yg diharapkan dgn instruksi INT.
* Terdapat dua jenis:
* **BIOS Interruptions**

Interrupt 00h - 1Fh (0-31) adalah interrupt BIOS dan standar di semua komputer baik yang menggunakan sistem operasi DOS atau bukan. Lokasi Interrupt Vector table-nya ada di alamat absolut 0000h-007Fh Interrupt BIOS dibawah ini telah dipastikan kegunaannya oleh sistem untuk keperluan khusus tidak boleh diubah oleh program  
seperti lainnya :  
1. DEVIDE BY ZERO Jika terjadi pembagian dengan nol maka  
proses akan segera dihentikan  
2. SINGLE STEP Untuk melaksanakan/mengeksekusi instruksi  
satu per satu  
3. NMI Pelayanan terhadap NMI (Non Maskable Interrupt) yaitu  
interupsi yang tak dapat dicegah  
4. BREAK POINT Jika suatu program menyebabkan overflow  
flag menjadi 1, interrupt ini akan melayani pencegahannya  
dan memberi tanda error

**Vector interrupt ROM BIOS**

Definisi : interrupt yang disediakan oleh BIOS (int 0 sampai int 1Fh)

|  |  |  |
| --- | --- | --- |
| **NO** | **Service** | **Fungsi** |
| INT 5 | - | Tombol print screen |
| INT 9 | - | Saat tombol keyboard lain ditekan, KB mengirim sandi lacak ke CPU bukan ke karakter ASCII |
| INT 10 | - | Mengubah mode video pada register AL 0-6 > layar grafik, 7 > layar monochrom.  Register input : AH=00, AL=nomor mode video. Contoh: Mov AL,5 ; Mov AH,0 ; Int 10 |
|  | 1 | Register input : AH=01h, CH=awal garis bentuk cursor, CL=akhir garis bentuk cursor. |
|  | 2 | Mengubah posisi kursor  Register input : AH=02h, BH=nomor halaman  tampilan, DH nomor baris (dimulai 00),  DL=nomor kolom (dimulai 00) |
|  | 3 | Register input : AH=03h, BH=nomor halaman  tampilan  Register output : CH=awal garis bentuk  cursor, CL=akhir garis bentuk cursor , DH  nomor baris (dimulai 00), DL=nomor kolom  (dimulai 00) |
|  | 6 | Register input : AH=06h, AL=jumlah baris  untuk digulung, BH=atribut untuk baris  kosong, CH,CL=koordinat kiri atas window,  DH,DL=koordinat kanan bawah |
|  | 7 | Scroll down window  Register input sama dengan service 6 |
|  | 8 | Melihat nilai karakter ASCII yang ada di layar.  Cara :  • Atur posisi cursor dengan fungsi 3  • Atur BH dengan nomor halaman 0  • Isi AH dengan 08h  • Jalankan int 10  • AL =Nilai ASCII, AH=atribut |

* **DOS Interruptions**  
  Interrupt 20h-FFh (32-255) adalah interrupt DOS, dan hanya ada pada komputer yang menggunakan sistem operasi DOS dan interrupt handlernya di load ke memori oleh DOS pada saat DOS digunakan. Lokasi interrupt vector tablenya ada di alamat absolut 07Fh-3FFh

**Vector Interrupt ROM DOS**

Definisi : Interrupt yang disediakan oleh DOS (Int di atas 1Fh)

|  |  |  |
| --- | --- | --- |
| **No** | **Service** | **Fungsi** |
| INT 20 | - | Menghentikan proses komputer terhadap suatu program COM |
| INT 21 | 01 | Membaca karakter dari keyboard dan menampilkan di layar.  Register input : AH=01h  Register Output : AL=kode ASCII |
|  | 02 | Mencetak karakter ke layar. Register input : AH=02 |
|  | 05 | Output ke printer  Register input : AH=05  Register output : DL |
|  | 09 | Mencetak string ke layar  Register input : AH=09 |
| 22h | - | Terminate code |
| 23h | - | Ctrl-Break code |
| 24h | - | Critical Error Handler |
| 25h | - | Absolute Disk Read |