**Microprocesadores**

La **arquitectura abierta** es configurable, tanto del punto de vista del software como del hardware; mientras que la **arquitectura cerrada** no lo es, ya que es de propósito específico.

Un **procesador** procesa bits que obtiene de una memoria, los opera y almacena el resultado de nuevo en memoria. Una **palabra de memoria** es la cantidad de bits a los que se puede acceder por vez.

La **capacidad de direccionamiento** es igual a 2\*…\*n tal que n=cantidad de bits del bus de direcciones.

El **numero de registros internos** es la cantidad de registros con la que cuenta el microprocesador.

**Registros**

Siempre serán de 16 bits.

Pueden ser:

* **De cálculo:**
  + **AH|AL**: acumulador
  + **BH|BL**: base
  + **CH|CL**: contador
  + **DH|DL**: datos
* **De punteros:** utilizados para desplazarse dentro de un bloque o zona de memoria.
  + **IP**: puntero de instrucciones
  + **SP**: puntero de pila (a pila)
  + **BP**: base de pila (a pila)
  + **SI**: índice fuente (indexado)
  + **DI**: índice destino (indexado)
* **De estado**: alojan a todas las banderas aritméticas.
  + **OF (desbordamiento)**: indicador de error. Se pone en 1 cuando la operación excede los 16 bits o en un cambio de signo en un desplazamiento.
  + **DF (dirección en operaciones con cadenas)**: vale 1 cuando se recorre la cadena de izquierda a derecha.
  + **IF (indicador de interrupción)**: vale 1 si se permite reconocimiento de interrupciones.
  + **TF (modo traza)**: vale 1 si se ejecuta paso por paso. Usado en fase de depuración.
  + **SF (indicador de signo)**: vale 1 cuando la operación da resultado negativo.
  + **ZF (indicador de cero)**: vale 1 cuando la operación da resultado cero.
  + **AF (acarreo auxiliar)**: vale 1 cuando hay acarreo en el bit 3.
  + **PF (paridad)**: vale 1 cuando la suma de unos de la operación da par.
  + **CF (bit de acarreo)**: vale 1 cuando hay acarreo en la operación.
* **De segmento**:
  + **CS (segmento de código):** contiene el valor de segmento donde se encuentra el código.
  + **DS (segmento de datos):** contiene el segmento donde están los datos.
  + **ES (segmento extra de datos):** accede a otro segmento con más datos.
  + **SS (segmento de pila):** valor del segmento en donde se encuentra la pila.

**Modos de direccionamiento**

|  |  |
| --- | --- |
| MOV AX,BX | Direccionamiento de Registro (No accede a MP) |
| MOV AX,204 | Direccionamiento Inmediato (No accede a MP) |
| MOV AX,[10] | Direccionamiento Directo (valor de la posición de memoria DS:10, Accede a MP) |
| MOV AX,TABLA | Direccionamiento Directo |
| MOV AX,[BX] | Direccionamiento Indirecto mediante Registro (DS:[BX]) |
| MOV AX,[BX]+2 | Direccionamiento por Registro Base (valor en la posición de memoria+2) |
| MOV AX,TABLA[DI] | Direccionamiento Indexado (valor de tabla sumado al contenido de DI) |
| MOV AX,TABLA[DI][BX] | Direccionamiento Indexado respecto de una base (valor de tabla sumado al contenido de DI y BX) |
| Si el operando se deduce | Direccionamiento Implícito |

**Pilas**

La CPU utiliza la pila para:

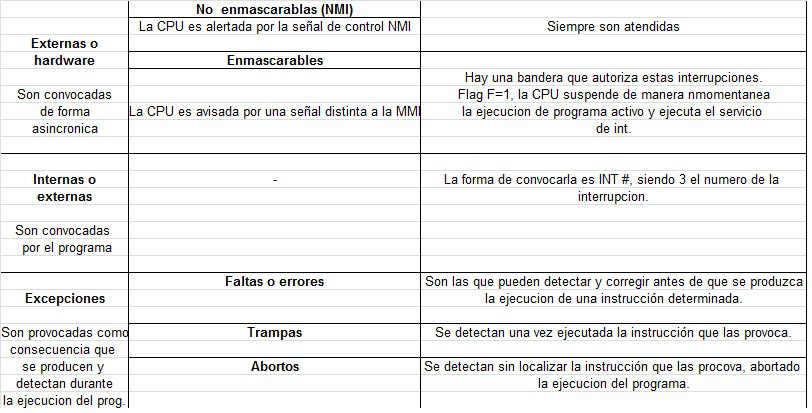
* Almacenar la dirección de retorno IP.
* Almacenar el estado del procesador cuando se produce una interrupción. Los registros que apila son el CS y el IP y estado de Flags.
* Pasar parámetros entre procedimientos.

El acceso a la pila se realiza mediante los registros punteros SP y BP. El SP es el registro que contiene la dirección del próximo elemento de la pila vacío. La pila no se ve porque no es un registro, solo se ve el decremento del puntero de la pila SP. El registro SP se va decrementado antes de que se ingresen los datos en la pila, luego de extraerlos se incrementa.

**Interrupciones**

Son acontecimientos que provocan la desviación del flujo de control de la CPU, que pasa a ejecutar un servicio que interprete el manejo de la interrupción.

* **Interrupciones internas o interrupciones software**: son convocadas por el programa en ejecución. Cabe destacar que son interrupciones programadas y suspenden momentáneamente él programa.
* **Interrupciones externas o hardware**: recibe la señal de interrupción desde afuera, son convocadas de forma asincrónica, no dependen del programa en ejecución, detecta el controlador de interrupciones programable – avanzado (PIC-APIC). Pueden ser no mascarables (NMI – cuando la CPU es avisada por la señal NMI siempre es atendida) o mascarables (la CPU es avisada por otra señal. INTR con el flag IF que si esta en 1 se atiende la interrupción, de otra forma no).
* **Excepciones**: causadas por anomalías. Las faltas o errores pueden detectarse antes de ejecutar la instrucción, las trampas se detectan una vez ejecutada la instrucción y el aborto se detecta sin necesidad de localizar la instrucción, abortando la ejecución del programa.



Cuando se detecta una interrupción podemos indicar 3 pasos:

* Resguardo del contexto en la Pila.
* Ejecución del servicio asociado a la interrupción.
* Restauración del contexto.