

Instituto Tecnológico de Costa Rica

Principios de Sistemas Operativos

Ernesto Rivera Alvarado

Proyecto 1

Integrantes:

Jefri Cárdenas Villatoro

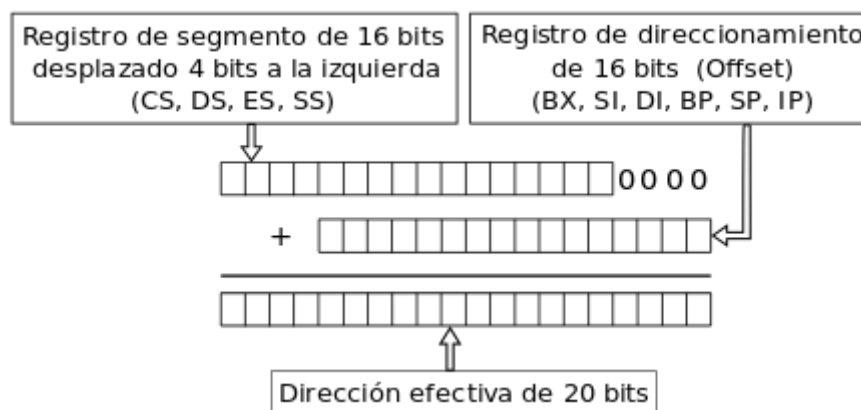
Edgar Parra Barillas

Ricardo Víquez Mora

Semestre I - 2020

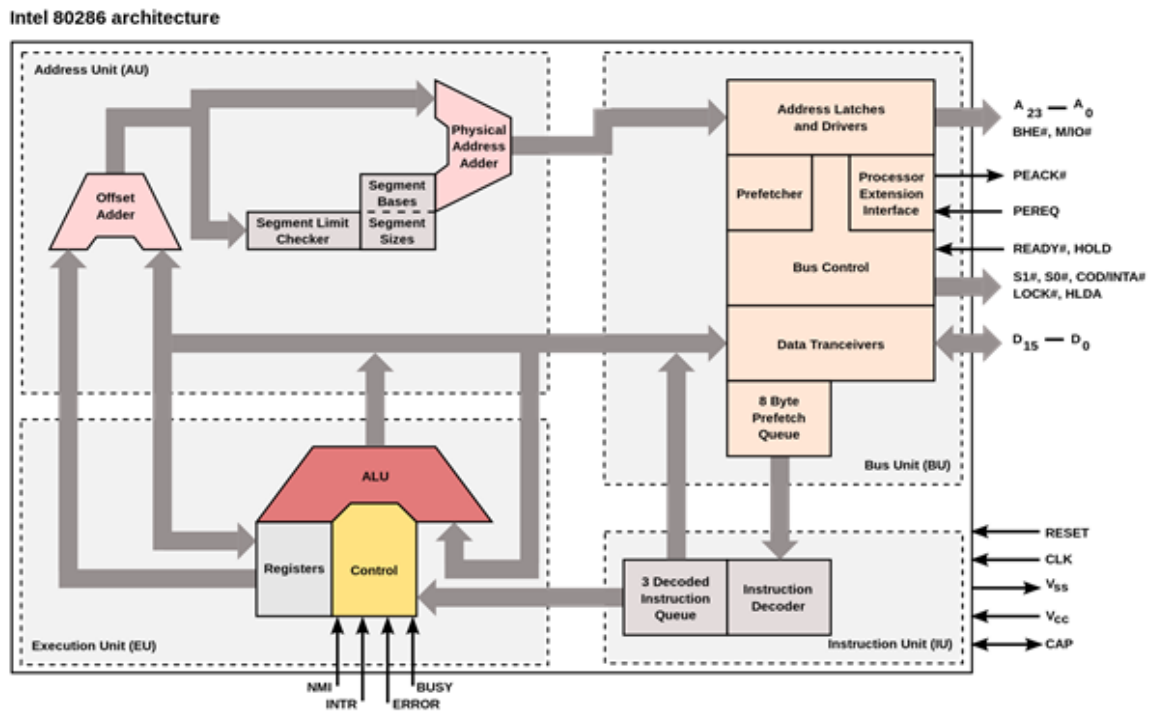
Modo Real

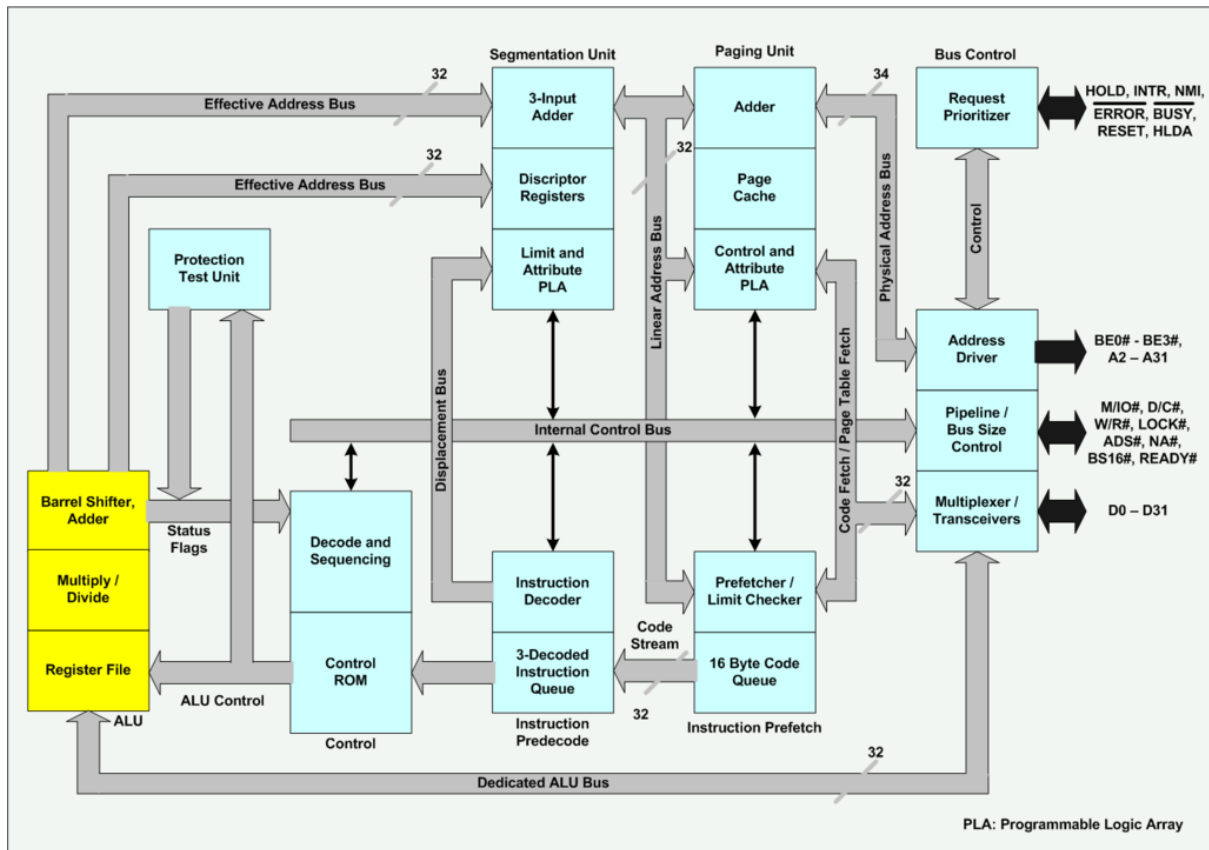
Modo Real : es un modo de operación del 8086 y posteriores CPUs del tipo de arquitectura x86. Caracterizado por 20 bits de espacio (dirección física = Segmento * 16 + Offset) de direcciones segmentado de solo 1 MB de memoria, acceso directo a las rutinas del BIOS por medio de programas, entrada/salida o periféricos sin protección de memoria. Esta memoria segmentada 1 MB es la llamada memoria real. Las dos partes de la memoria , el segmento ,es un fragmento de memoria de 64 KB , y el desplazamiento puede apuntar a cualquier parte dentro de ese fragmento. Para calcular la dirección real , multiplica la parte del segmento de la dirección por 16(o se desplaza a la izquierda 4 bits) , y luego agregue el desplazamiento. Con esta suma se obtiene la dirección de 20 bits.



80286 con una arquitectura de 16 bits

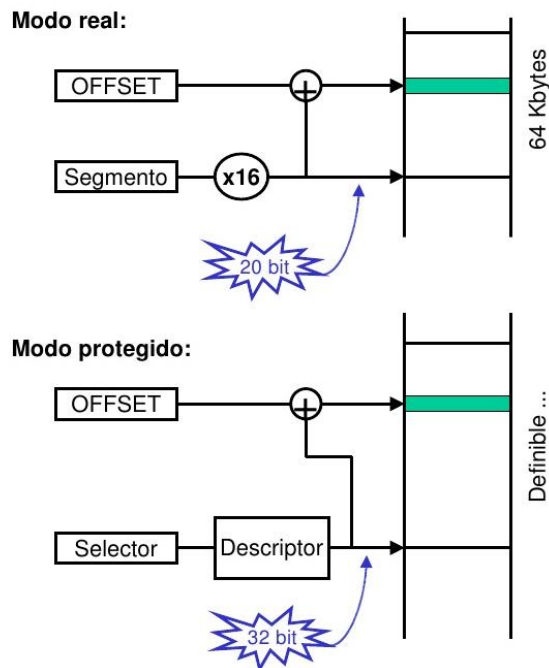
Esta es la arquitectura en la cual se implementa el modo real.





Modo protegido

Diseñado para mejorar la multitarea y la estabilidad del sistema, tales como la protección de la memoria y soporte de hardware para memoria virtual. La mayoría de los sistemas operativos x86 modernos se ejecutan en modo protegido. Los CPUs x86 comienzan en modo real en el momento del boot time para asegurar compatibilidad hacia atrás con los sistemas operativos heredados. La característica de protección de memoria es que previene que un programa pueda dañar la memoria de otra tarea o del núcleo del sistema operativo.



El descriptor especifica la ubicación del segmento en memoria, su longitud y sus derechos de acceso. Con un fragmento de memoria definible y no de 64KB.

Anillos de Protección

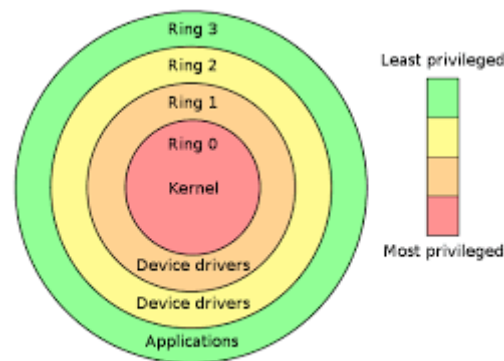
En modo protegido, hay 4 niveles de privilegios o anillos, del 0 al 3, siendo el anillo 0 el más privilegiado y 3 el menor. El uso de anillos permite que el software del sistema pueda controlar o restringir los accesos de las diferentes tareas a los datos. Se tienen entonces las instrucciones privilegiadas.

Definimos instrucciones privilegiadas aquellas instrucciones que pueden potencialmente dañar a otros procesos. Por ejemplo: operaciones de E/S, actualizar el reloj, desactivar interrupciones, manipular la MMU o el bit de protección.

En la mayoría de los entornos, el sistema operativo opera en el nivel 0 y las aplicaciones en el nivel 3.

El kernel es una parte del sistema operativo es el responsable de facilitar a los programas acceso seguro al hardware de la computadora. También se encarga de decidir que programa podría usar un espacio de hardware y durante cuánto tiempo. Como el hardware es limitado, los núcleos suelen implementar una serie de

abstracciones del hardware, él esconde la complejidad y facilita el uso al programador.



Bootloader

En cuanto al bootloader utilizado en el proyecto se utilizó GRUB (**Grand Unified Bootloader**), este es un pequeño programa almacenado en la tabla de particiones, es un administrador parte del proyecto GNU que se utiliza para cargar ya sea uno o varios sistemas operativos alojados en el computador a memoria (multiboot). Este gestor de arranque lo que hace es primero cargar un archivo pequeño de unos 8KB que consiste en 32 bits de indicadores (flags) seguido de la información de la imagen a ejecutar.

GRUB tiene las siguientes características destacadas:

- Compatible con multiboot
- Soporta múltiples arquitecturas de hardware y sistemas operativos como Linux y Windows.
- Ofrece una interfaz interactiva de línea de comandos tipo Bash para que los usuarios puedan ejecutar comandos de GRUB, así como interactuar con archivos de configuración.
- Permite el acceso al editor GRUB.

- Soporta configuración de contraseñas con cifrado para seguridad.
- Soporta el arranque desde una red combinada con varias otras características menores.

Este gestor permite los siguientes sistemas de archivos:

- ext2/ext3/ext4
- ReiserFS
- XFS
- UFS
- VFAT, FAT16 y FAT32
- NTFS
- JFS
- HFS

¿Cómo inicia el GRUB?

Este se ejecuta cuando el usuario enciende el ordenador presionando el botón de Power ON la BIOS (**Basic Input Output System**) hace un chequeo de los componentes de hardware, seguidamente este al arranque busca un dispositivo el cual tiene un orden establecido, por ejemplo disco duro, CD-ROM/DVD, memorias USB, etc. A continuación el BIOS carga en memoria el programa alojado el dispositivo alojado y pasa a un registro conocido como Master Boot Record (**MBR**), consiste en 512 bytes del disco duro. De la misma manera el programa cargado ya ubicado en alguna parte del disco duro, el bootloader se encarga de cargar la siguiente fase que corresponde a los siguientes 30KB, en el cual se recibe el control de memoria y se muestra al usuario un menú de inicio con los sistemas almacenados o disponibles como el siguiente

```
GNU GRUB version 0.97 (639K lower / 1046784K upper memory)

Ubuntu, kernel 2.6.20-15-generic
Ubuntu, kernel 2.6.20-25-generic (recovery mode)
Ubuntu, memtest86+
Other operating systems:
Windows Vista/Longhorn (loader)
Windows Vista/Longhorn (recuperación)

Use the ▲ and ▼ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for command-line.

The highlighted entry will be booted automatically in 7 seconds
```

luego el GRUB carga el kernel que el usuario seleccionó a memoria y le cede el control para así poder arrancar con el sistema y quedar listo para el inicio de sesión por parte del usuario.

Referencias

Burger, J. (2020). *Intel X86 Assembly Language & Microarchitecture - Real Mode | X86 Tutorial*. Recuperado de: <https://riptutorial.com/x86/example/12672/real-mode>.

Burger, J. (2020). *Intel X86 Assembly Language & Microarchitecture - Protected Mode | X86 Tutorial*. Recuperado de: <https://riptutorial.com/x86/example/12673/protected-mode>.

Mifsud, E. (2007). *El gestor de arranque GNU GRUB*. Recuperado de: [https://www.ecured.cu/GRUB_\(GRand_Unified_Bootloader\)](https://www.ecured.cu/GRUB_(GRand_Unified_Bootloader)).