Arranque del Sistema e Implementación de Procesos

Arquitectura interna Sistemas Operativos

Arquitectura x86

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Índice

- Historia
- Registros
- Repertorio de Instrucciones
- Modo Real
- Modo Protegido

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Índice – Historia

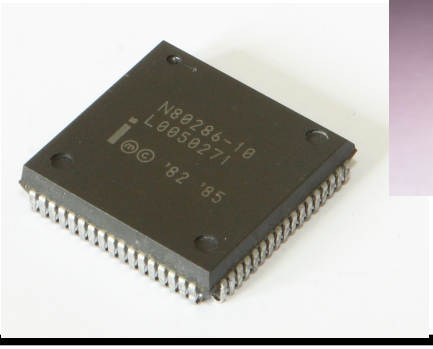
- Historia
  - Intel 16-bits
  - Intel IA-32
  - Intel P6
  - AMD Kryptonite
  - Intel P4

ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Intel 16-Bit

- Intel 8086/8088 (1978)
  - » IBM PC 5150 (1981) y el IBM PC-XT 5160 (1983)
  - » Bus Datos 16 Bits
  - » Bus Dir 20 Bits (compartidos con Bus Dir)
  - » Acceso segmentado *Segmento:Desplazamiento*
    - – Memoria Direccionable 1 MB (Segmentos 64KB)
  - » Coprocesador de punto flotante (8087)
  - » 80186 (Embedded) – iAPX 186 –

- Intel 80286 (1982): 1.5 µm
  - » IBM-AT (1984), BUS IDE, Bus Datos 16 bits, Dir 24 Bits
  - » MMU. Paginación -opcional-
  - » 2 Modos: Real y Modo Protegido 16-Bits
    - – Real: Mantener compatibilidad con 8086
    - – Protegido: Memoria Direccionable 16 MB



IBM

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Intel IA-32

- Intel 386 (1985): 1.5 – 1 µm
  - » Modo Protegido 32 Bits
    - – 4 GB de memoria direccionable
  - » Modo Virtual 8086 (MV86)
  - » clon AMD386 (1991)
- Intel486 (1989): 0.80 µm
  - » Pipelining, FPU y Cache integrada
  - » clones AMD486 (1993) y AMD5x86 (1995)
- Pentium (1993): 0.80 – 0.25 µm
  - » Superscalar dos vias
  - » Pentium MMX (1997) – 0.25 µm
    - – Repertorio MMX (1997) – 57 nuevas Instrucciones SIMD int, Reg MMX

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Intel P6

- Pentium Pro (1995): 0.50 – 0.35 µm
  - » Ejecución especulativa fuera de orden, Segmentación más profunda, Renombramiento de Registros, Nucleo RISC (µ-operaciones)
  - » *On-package* L2
  - » Physical Address Extension (PAE):
    - – Direcciones Fisicas 36-bits (64GB),
    - – Direcciones Virtuales 32-bits
- Pentium II (1997): 0.35 – 0.25 µm
  - » Celeron – Pentium II – Pentium II Xeon
- Pentium III (1999): 0.25 – 0.18 µm Evolución hasta el 2006!!!
  - » Katmai (May 1999 – 0.25): Repertorio SSE – SIMD FP
  - » Coppermine (Oct 1999 – 0.18): L2 integrada
  - » Tualatin (2001). Predecesor de:
    - – Intel Pentium M (2003) e Intel Core Duo/Solo (2006)

1995

1997

1999

2003

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

AMD Kryptonite

- AMD K5 (1995): 0.50 – 0.35 µm
  - » Basado en AMD 29k RISC
    - – Decodificador CISC/RISC
- AMD K6 MMX Enhanced (1997): 0.35 – 0.25 µm
- AMD K6-II/III (1998): 0.25 – 0.18 µm
  - » K6-2 (1998): Repertorio AMD 3DNow! - SIMD FP
  - » K6-III (1999): L2 integrada
- AMD K7 – Athlon (1999): 0.25 – 0.13 µm
  - » Athlon Classic (1999)
  - » Athlon XP (2001)

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Intel Pentium 4

- Willamette (2001) – 180nm
  - Arquitectura Netburst: Super-pipelining (GHz)
  - SSE2
- Northwood (2002) – 130nm
  - Intel Xeon (2003): Hyperthreading
- Gallatin (2003) – 130nm
  - Pentium 4 Extreme Edition
- Prescott (2004) – 90nm
  - SSE3
  - 3.8 GHZ
- Cedar Mill (2006) – 65 nm

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Índice – Registros

- Registros
  - » Registros IA-32
  - » Registros IA-32. Propósito General
  - » Registros IA-32. Registros Especiales

ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Registros IA-32 (I)

- 8 Registros Propósito 32 Bits

EAX
EBX
ECX
EDX

EBP
ESP
ESI
EDI

- Flags-Control, Puntero Instrucción

EFLAGS
EIP

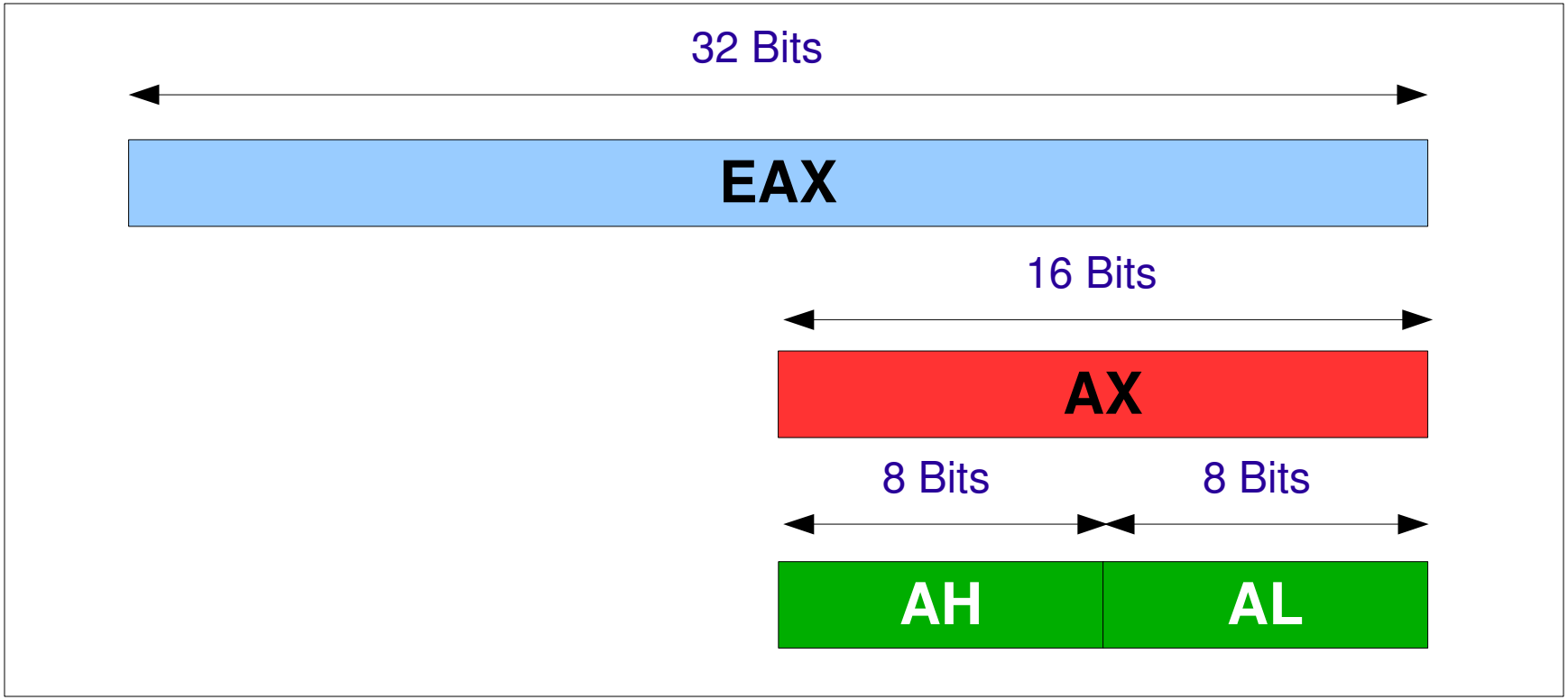
- 6 Registros Selectores Segmento 16 Bits

CS	ES
SS	FS
DS	GS

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Registros IA-32 (II)

- Para mantener compatibilidad con procesadores anteriores, los registros se solapan
  - » 16 Bits: AX, BX,CX, DX, BP, SP, SI, DI, FLAGS, IP
  - » 8 Bits: AH, LH, BH, BL, CH, CL, DH, DL



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Registros IA-32. Propósito General

- EAX, EBX, ECX, EDX, ESP, EBP, ESI, EDI son registros generales pero algunas instrucciones los tratan de forma especial.
  - EAX – Acumulador. Se usa implícitamente en división y multiplicación
  - EBX – Puntero a Datos en segmento DS
  - ECX – Contador bucles y strings
  - EDX – Puntero para operaciones I/O
  - ESP – puntero de pila
    - No se debe usar nunca para operaciones aritméticas, ni para transferencia de datos
  - EBP – Puntero a la pila para accesos con desplazamiento (marco de pila)
  - ESI – Puntero Datos en DS o Registro índice Fuente (Source) en transferencias
  - EDI – Puntero Datos en ES o Registro índice Destino en transferencias

ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

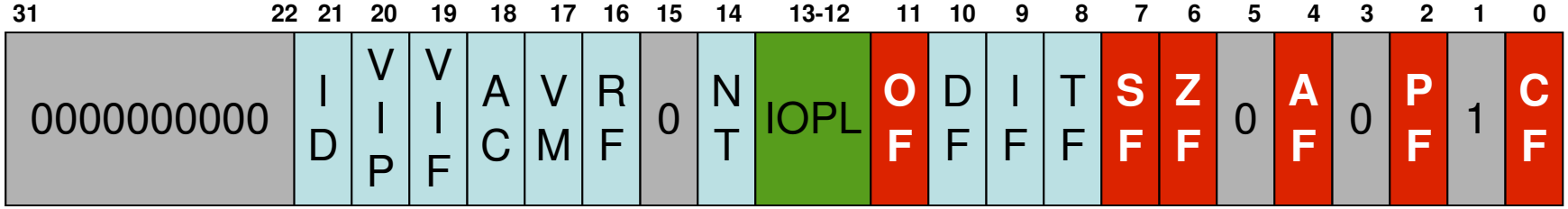
Registros IA-32. Registros Especiales (I)

- Registros de Segmento
  - CS – segmento de código
  - DS – segmento de datos
  - SS – segmento de pila
  - ES, FS, GS – segmentos adicionales
- EIP – Puntero de instrucción D
  - Desplazamiento dentro del segmento apuntado por CS

ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Registros IA-32. Registros Especiales (II)

- EFLAGS



= Bit reservados

= Flags de estado

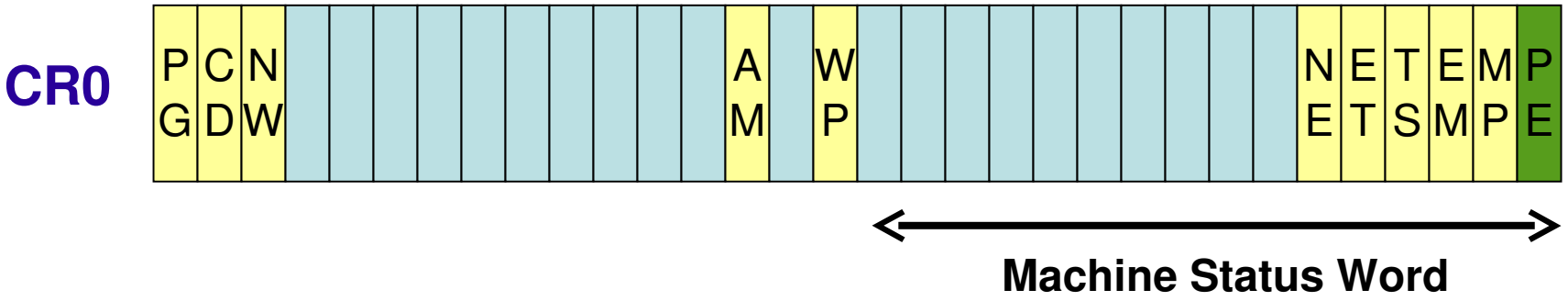
ZF = Zero Flag
SF = Sign Flag
CF = Carry Flag
PF = Parity Flag
OF = Overflow Flag
AF = Auxiliary Flag

IOPL = Nivel Privilegio I/O (0,1,2,3)
AC = Alignment-Check (1=yes, 0=no)
NT = Nested-Task (1=yes, 0=no)
RF = Resume Flag (1=yes, 0=no)
VM = Modo Virtual-8086 (1=yes, 0=no)
VIF = Flag Interrupción 'Virtual', Interrupt Virtual Pendiente?
ID = Instruccion CPUID si se puede cambiar el valor de este bit

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Registros IA-32. Registros Especiales (III)

- Crx [0-4]
  - » Determinan el modo operativo del procesador y las características de la tarea actual
  - » CR0: Paginación? (PG), Modo de Protección (PE) , Conmutación Tarea (TS)
  - » CR2: Si PG=1 y PE=1, Dirección que provocó Fallo de Página
  - » CR3: Si PG=1 y PE=1, Puntero Tabla de Paginas (primer nivel)



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Índice – Repertorio de Instrucciones

- Repertorio de Instrucciones
  - Sintaxis general
  - Orden little endian
  - Modos de direccionamiento
  - Repertorio de Instrucciones - manual -

ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Sintaxis general de instrucciones

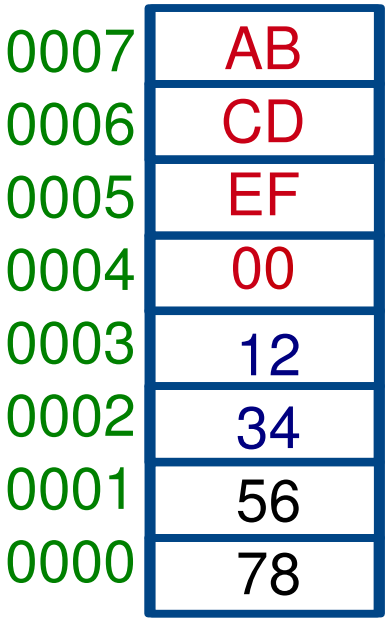
- Notación Intel – NASM, MASM
  - » Instrucción Destino, Fuente
    - mov eax,ebx !eax=ebx
    - add eax,ebx !eax=eax + ebx
  - » Instrucción Destino / Fuente
    - inc eax !eax=eax+1
    - push eax ! [esp]<-- eax, esp=esp-4
    - pop eax ! [esp]--> eax, esp=esp+4
  - » Instrucción Operando implícitos
    - pushad !guarda en la pila eax,ecx,edx,ebx,esp,ebp,esi,edi
    - popad !recupera de pila eax,ecx,edx,ebx,esp,ebp,esi,edi
- Notación AT&T – Gnu AS
  - » Instrucción Fuente, Destino
    - movl %ebx, %eax !eax=ebx l=double word, w=word, b = byte
    - addl %eax,%ebx !eax=eax + ebx

ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Orden Little Endian

- Todos los tipos de datos mayores que el byte, guardan sus bytes en orden little endian:
  - El byte menos significativo se guarda en la dirección más baja

- Ejemplo:
  - Palabra 16 bits: 5678h
  - Doble palabra 32 bits: 12345678h
  - Quad 64 bits: ABCDEF0012345678h



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modos de direccionamiento

- Operandos registro

  mov ax, ax

- Operandos constantes

  mov ax, 25 ; movw $25, %ax

  mov ecx, 8*8

- Direccionamiento Directo

  mov ax, (1000)

  mov esi, (_gdt + GDT_SELECTOR +2)

- Direccionamiento Indirecto

  mov ax, (bx)

- Direccionamiento Indexado

  mov ax, 4(ebx)

ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

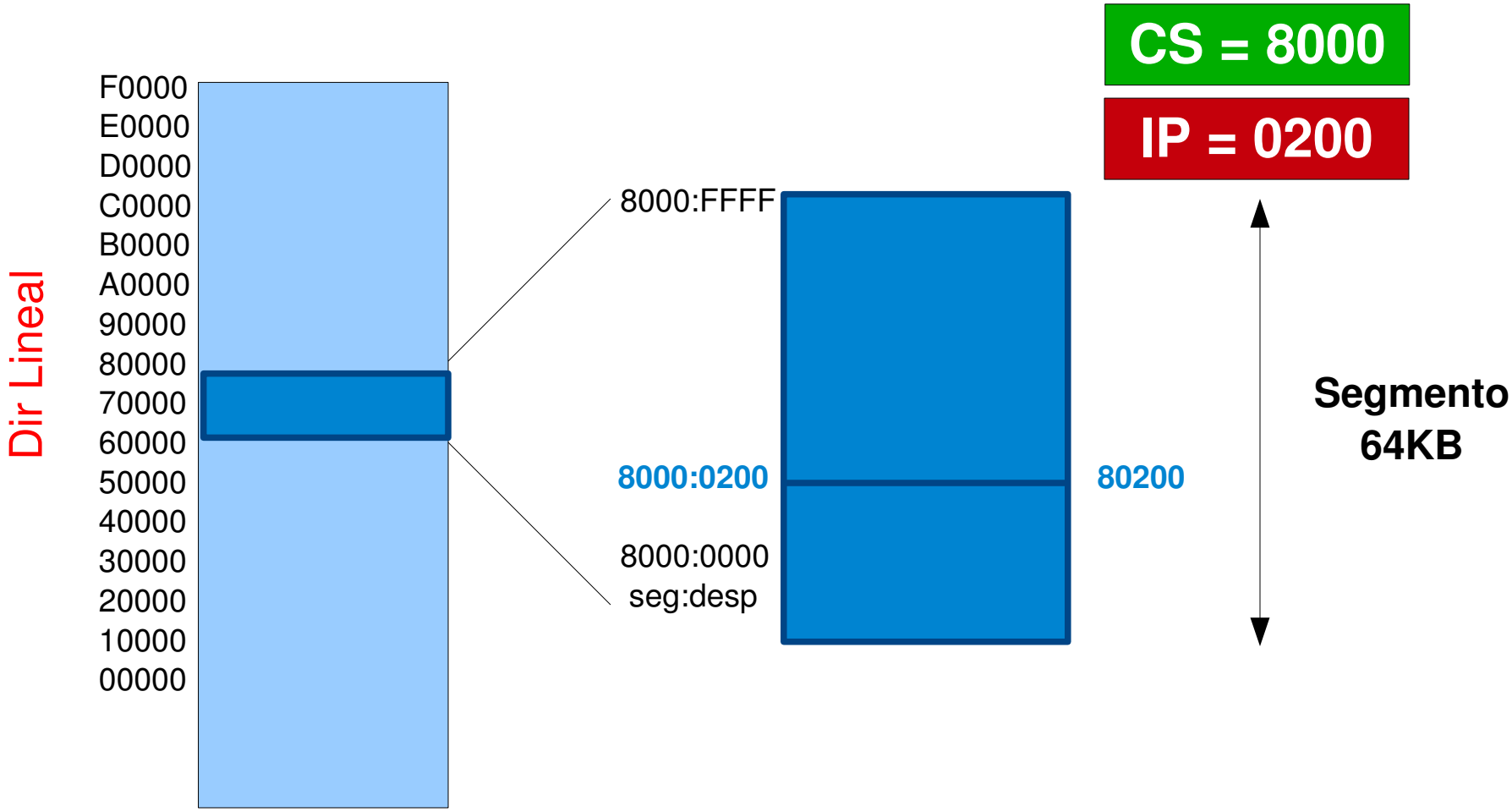
Modos de Operación IA-32

- Modo Real
  - Modo Nativo MS-DOS (monoprogramado)
  - Direcciones 20 Bits (Espacio Segmentado)
    - Espacio Direccionable 1MB
    - Punteros near, far, huge
  - El software tiene acceso a las rutinas de la BIOS y los periféricos
  - Todos los procesadores x86 arrancan en modo real
  - 8086 solo tiene un modo de operación semejante al modo real
- Modo protegido (32 Bits i386)
  - Modo Nativo de SO modernos (multiprogramación)
  - Direcciones 32 Bits (Espacio Segmentado, Paginación Opcional)
    - Espacio Direccionable 4GB
  - Selectores de segmento: índice tabla de segmentos
  - Modo Real Virtual
    - Virtualización: Modo real dentro de un contexto protegido y multitarea
    - Cada proceso dispone de su propio 8086

ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Real. Memoria Segmentada

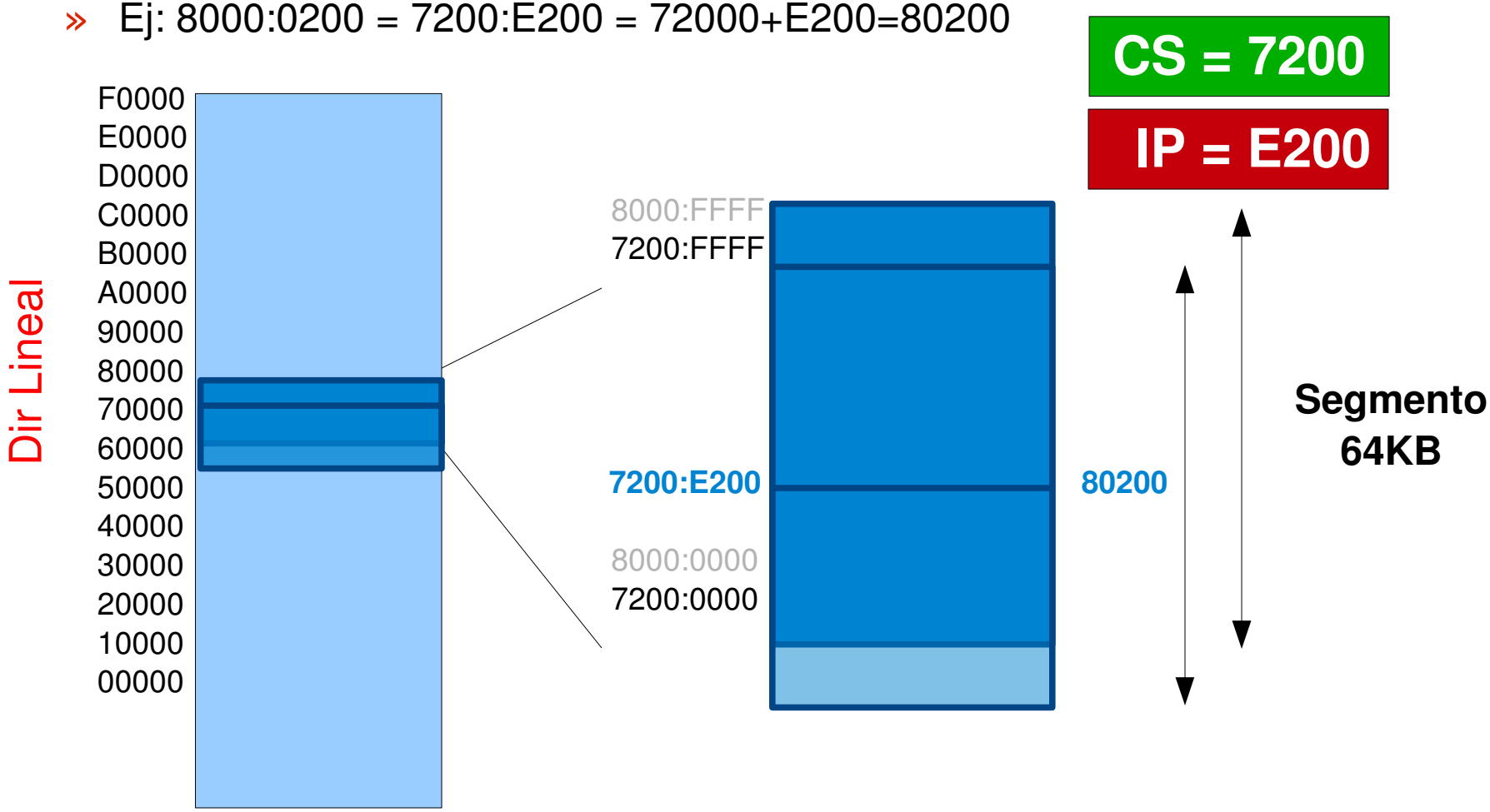
- Dirección: Segmento:Desplazamiento.
  - » Dirección lineal (absoluta) : 16*RegSegmento + Desplazamiento



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Real. Memoria Segmentada

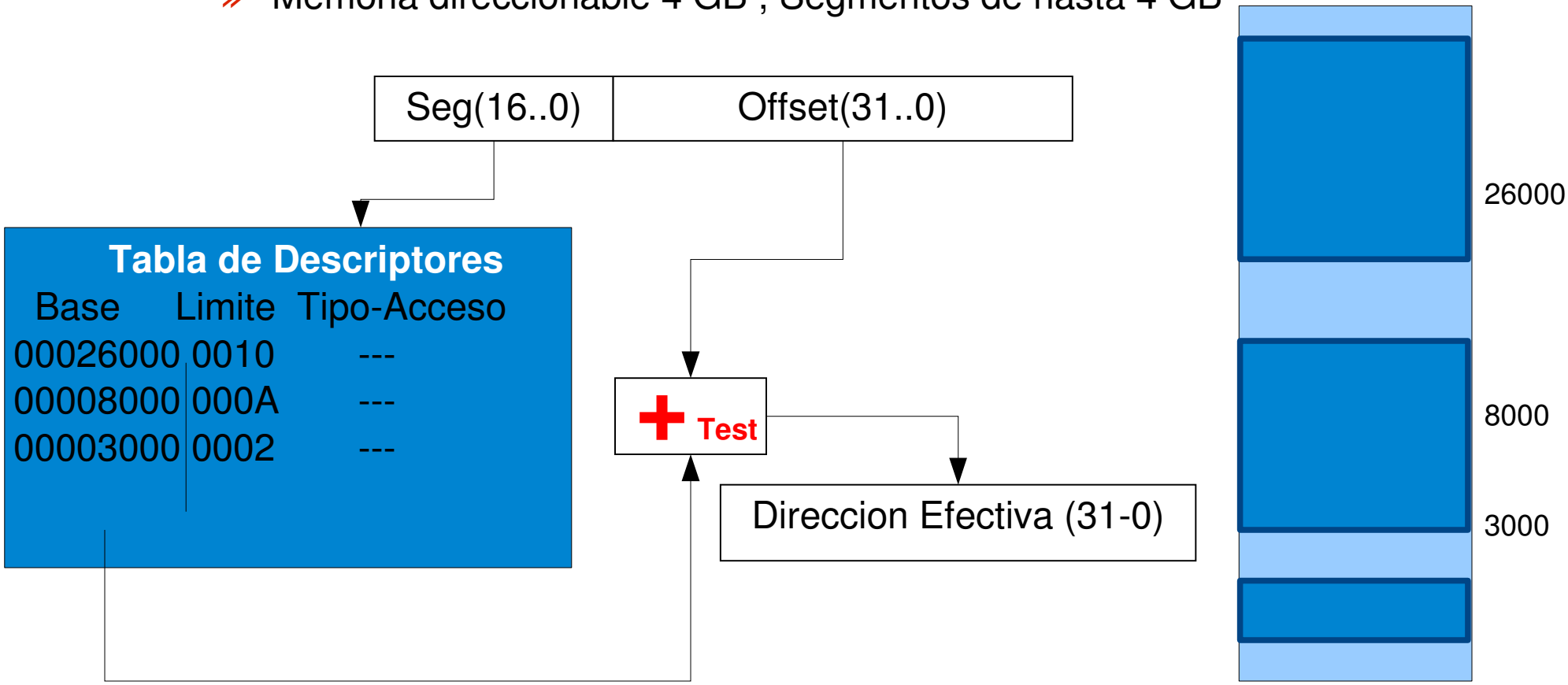
- Los segmentos pueden solaparse: A una misma dirección lineal le corresponden diferentes direcciones segmentadas
  - » Ej: 8000:0200 = 7200:E200 = 72000+E200=80200



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

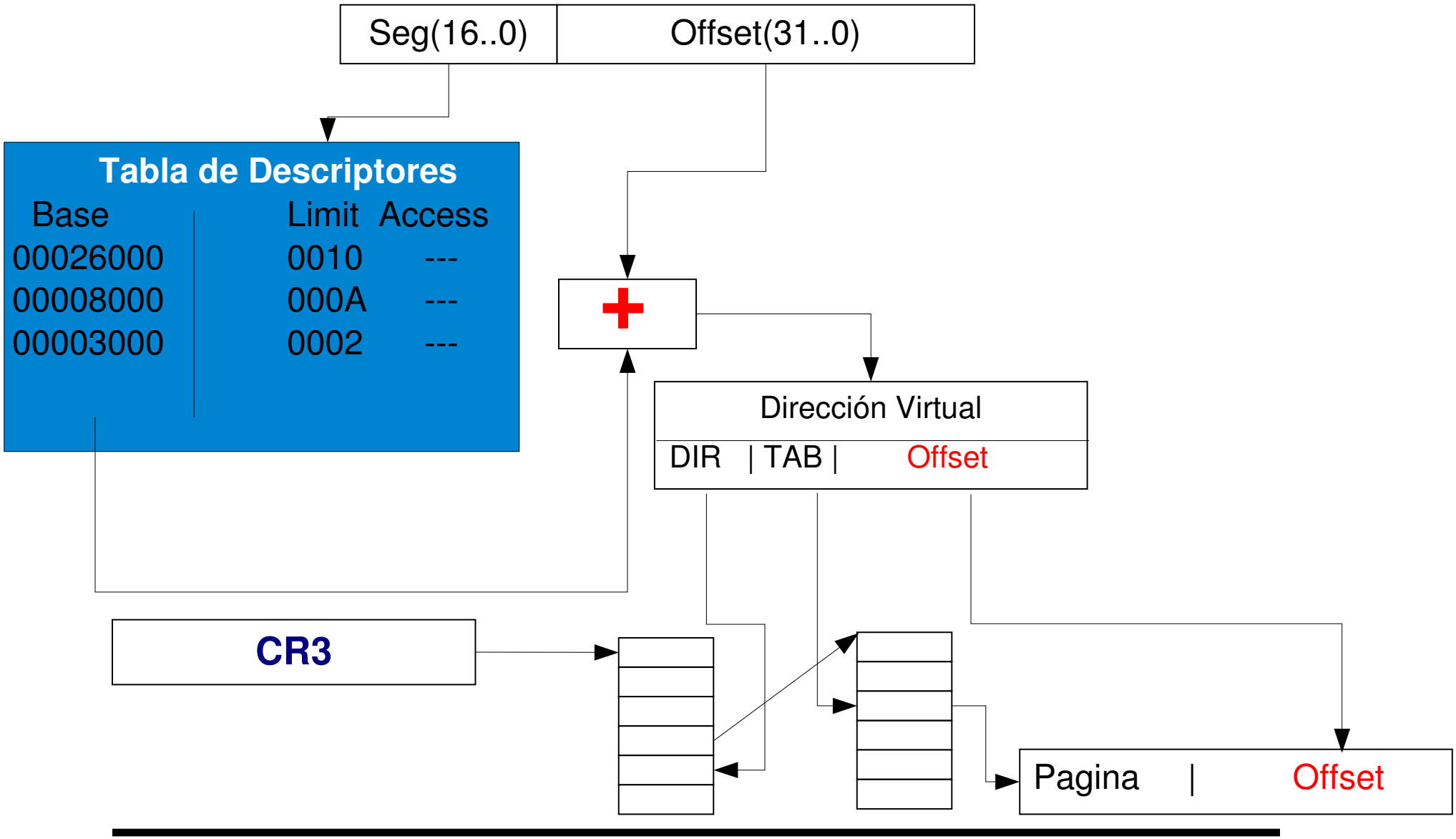
Modo Protegido

- La Segmentación Funciona Diferente
  - » Reg de segmentos apuntan a entradas de **Tabla de Descriptores** (segmentos)
  - » Memoria direccionable 4 GB , Segmentos de hasta 4 GB



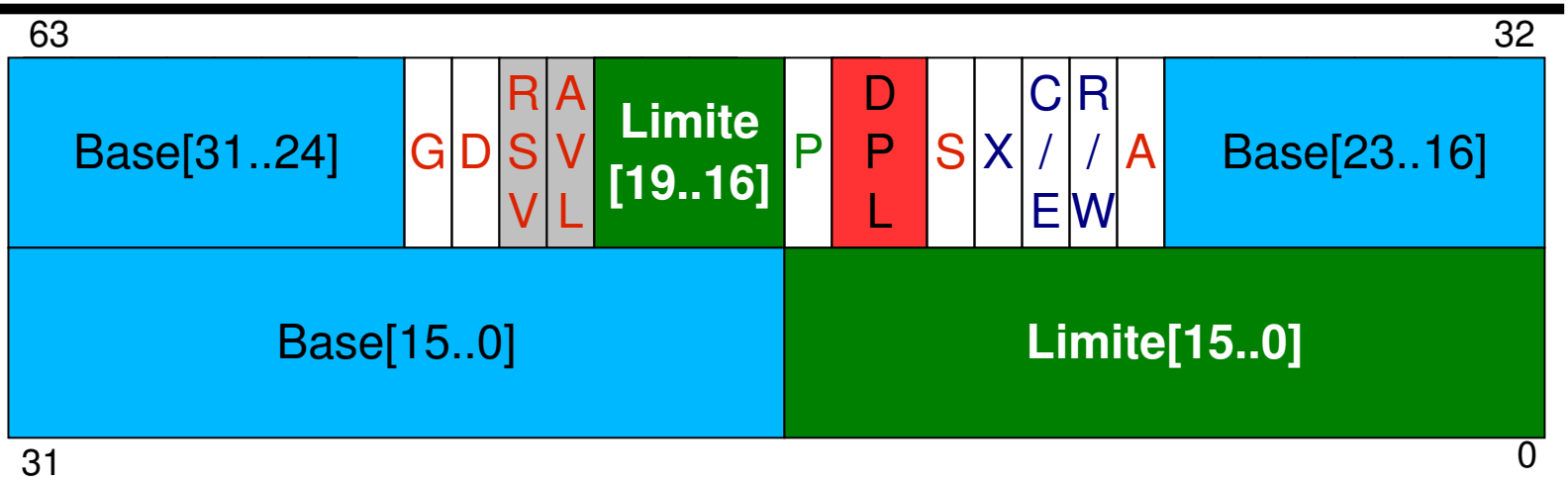
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Protegido. Paginación Opcional



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Protegido. Descriptor de Segmento (I)

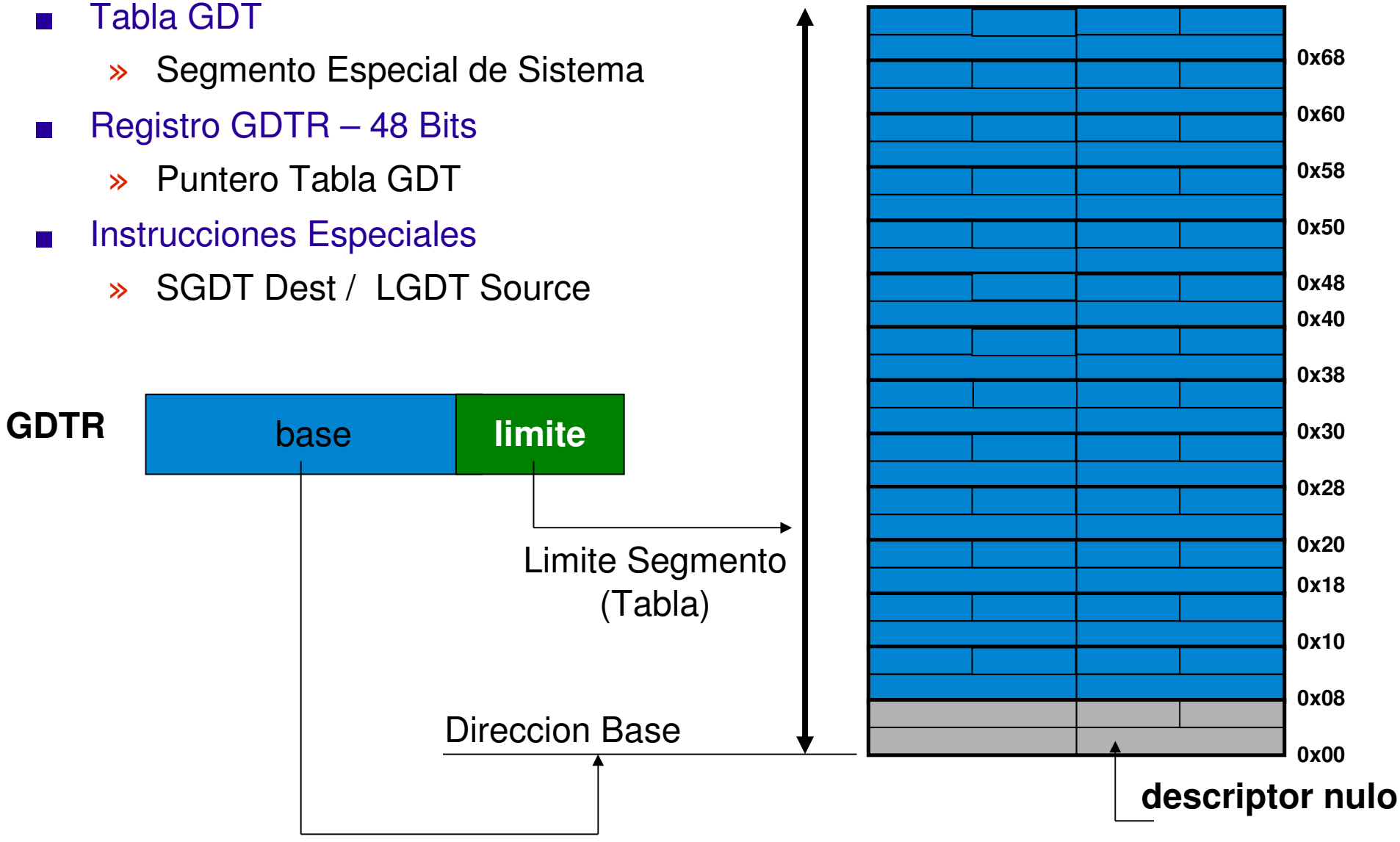


- P = Presencia/Cargado en memoria (1=si, 0=no)
- **DPL = Nivel de Privilegio (00=supervisor, 11=usuario)**
- S = Segmento de Sistema (0=si, 1=no)
- X = Ejecutable: 1=si (segmento codigo), 0=no (segmento datos)
- C/E = Conforming (1=si, 0=no) cuando Bit X=1
  Expansión-hacia-Abajo (1=si, 0=no) cuando X=0
- R/W = Readable (1=si, 0=no) cuando X-bit=1
  Writable (1=yes, 0=no) cuando X-bit=0
- A = El segmento ha sido accedido (1=si, 0=no)
- G = Granularidad Segmento: (0=byte, 1=pagina). El valor de Limite depende de G
- D = Operandos y Direcciones por defecto (0=16-bits, 1=32-bits)

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Protegido. Tabla de Descriptores Global – GDT (I)

- Tabla GDT
  - » Segmento Especial de Sistema
- Registro GDTR – 48 Bits
  - » Puntero Tabla GDT
- Instrucciones Especiales
  - » SGDT Dest / LGDT Source



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Protegido. Tabla de Descriptores Global – GDT (II)

- Ejemplo Información de la GDT (Minix)

BIOS

```
#define GDT_INDEX         1          /* GDT descriptor */
#define IDT_INDEX         2          /* IDT descriptor */
#define DS_INDEX          3          /* kernel DS */
#define ES_INDEX          4          /* kernel ES (386: flat 4 Gb at startup) */
#define SS_INDEX          5          /* kernel SS (386: monitor SS at startup) */
#define CS_INDEX          6          /* kernel CS */
#define MON_CS_INDEX      7          /* temp for BIOS (386: monitor CS at startup) */
#define TSS_INDEX         8          /* kernel TSS */
#define DS_286_INDEX      9          /* scratch 16-bit source segment */
#define ES_286_INDEX      10         /* scratch 16-bit destination segment */
#define A_INDEX           11         /* 64K memory segment at A0000 */
#define B_INDEX           12         /* 64K memory segment at B0000 */
#define C_INDEX           13         /* 64K memory segment at C0000 */
#define D_INDEX           14         /* 64K memory segment at D0000 */
#define FIRST_LDT_INDEX   15         /* rest of descriptors are LDT's */
```


ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Protegido. Tabla de Descriptores Global – GDT (III)

- Instrucciones Privilegiadas para leer/escribir en el registro GDTR
  - sgdt dest_6_bytes

    dest[0:15] <- GDTR(limite)

    dest[16:47] <- GDTR(base)

  - lgdt src_6_bytes

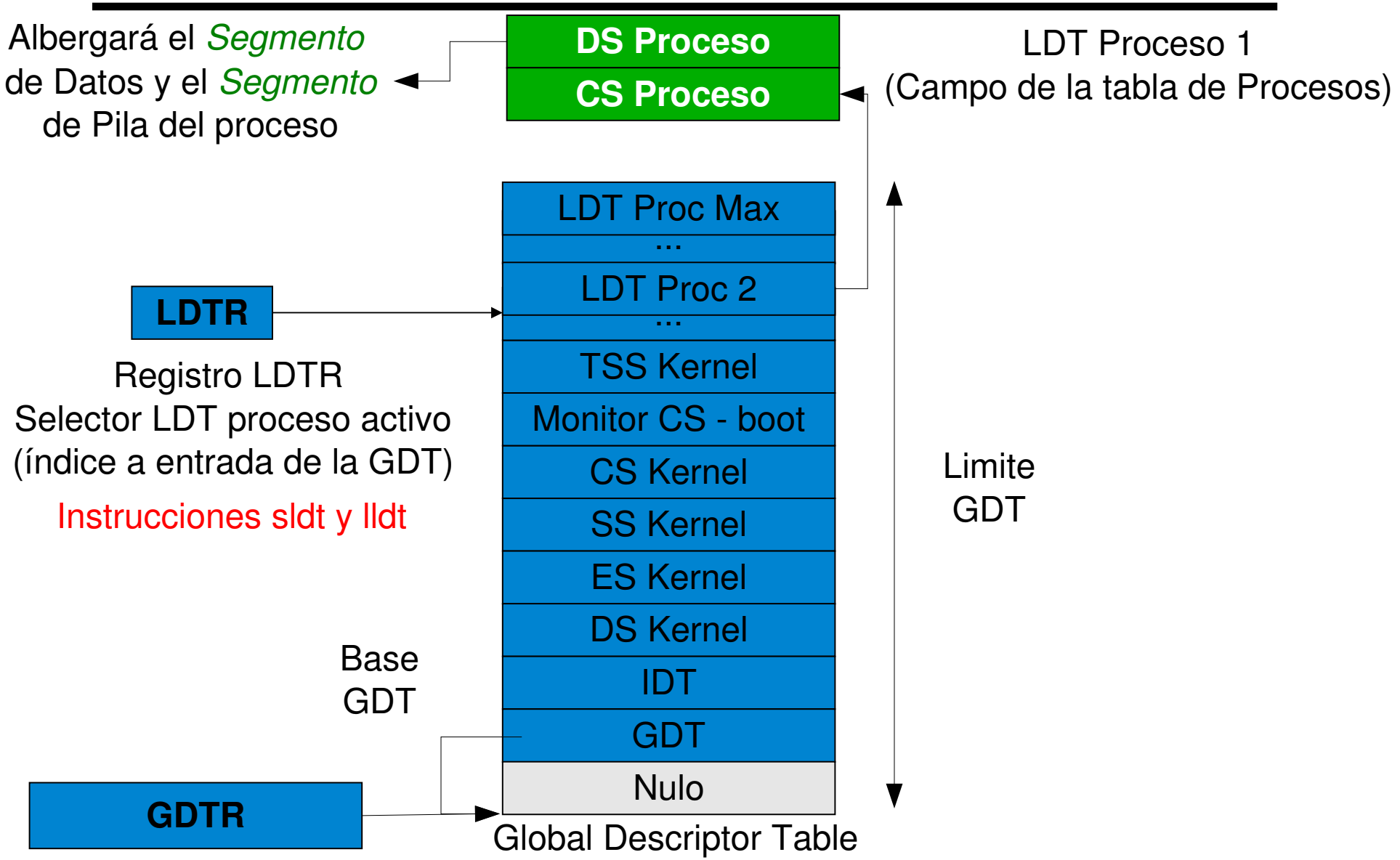
    GDTR(limite) ← src[0:15];

    GDTR(base) ← src[16:47];

ArTeCS

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

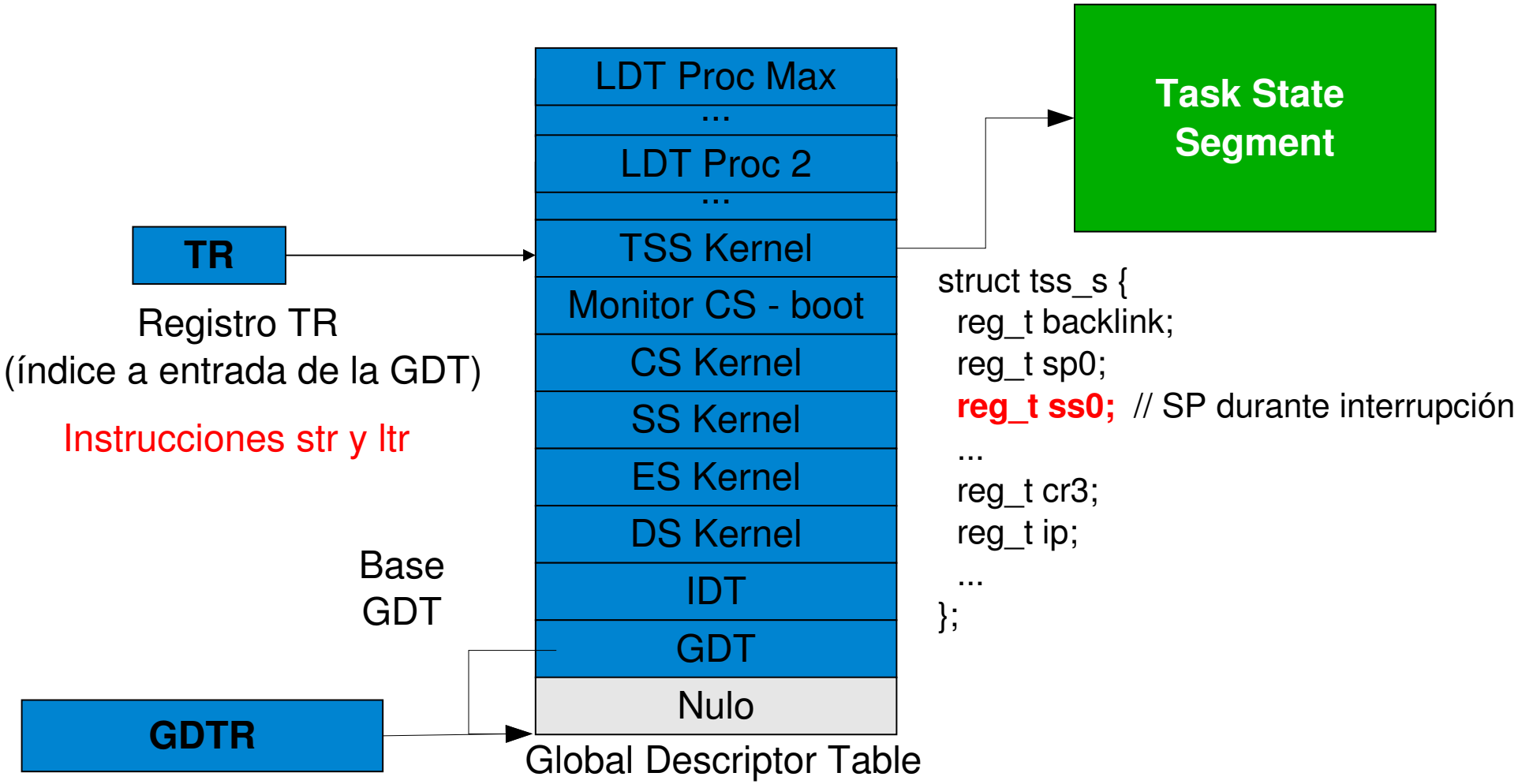
Modo Protegido. Tabla de Descriptores Local – LDT



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Protegido. Segmento Estado Tarea – TSS

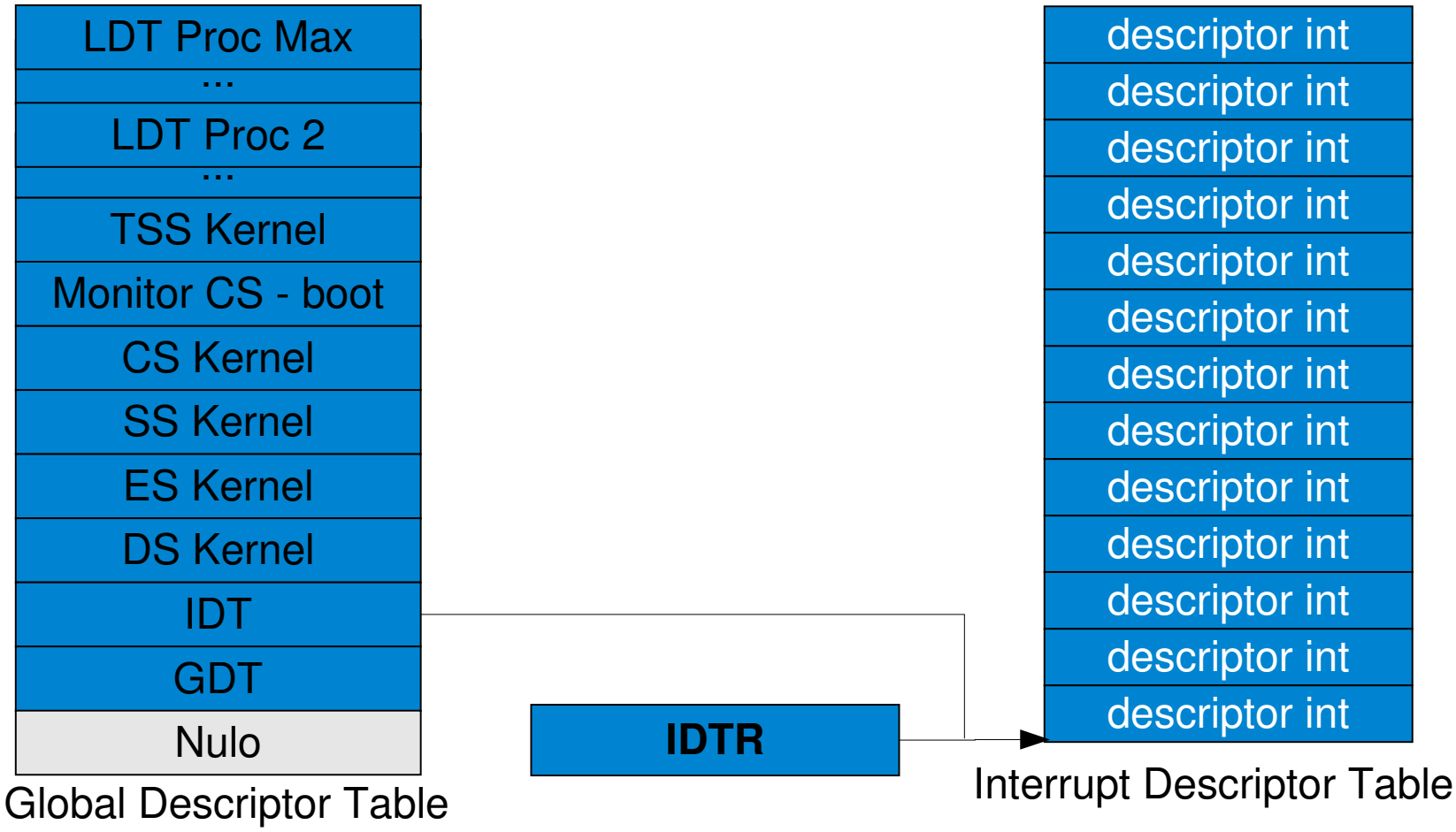
- TSS: Espacio para almacenar los registros del procesador y otra información que hay que guardar en una conmutación de tarea



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Protegido. Tabla Descriptores Interrupción – IDT

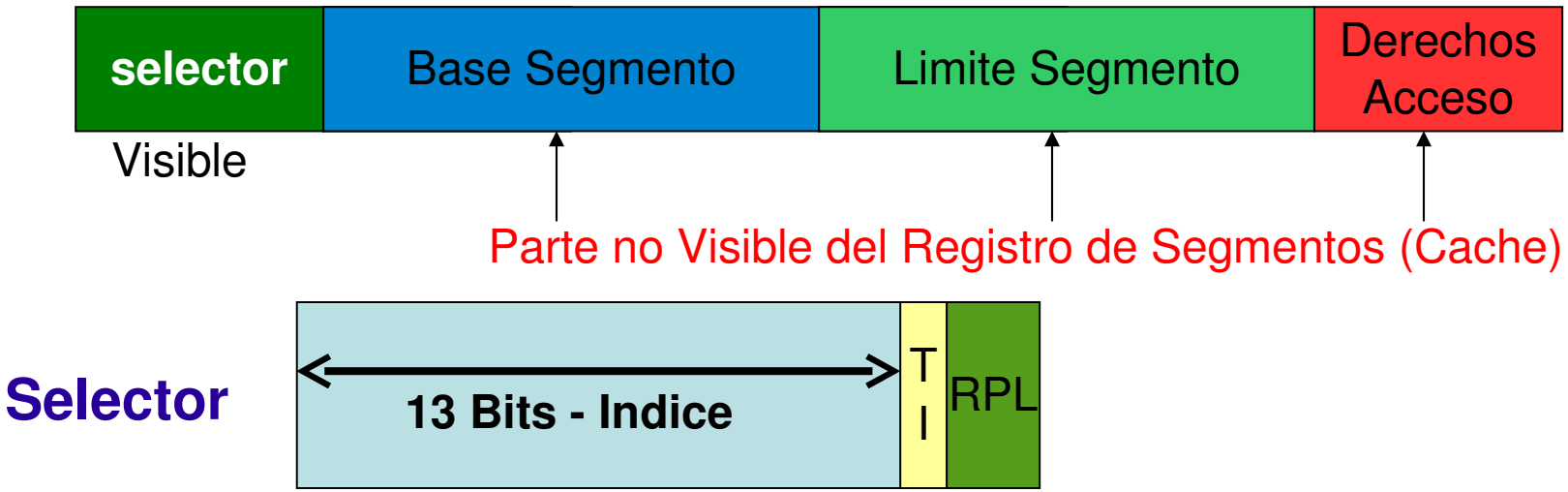
- IDT: interrupción gate descriptores



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Protegido. Registros de Segmento: Selector

- Registros de Segmentos
  - La parte oculta del registro de segmento se carga en el 1er acceso
  - TI: Table indicator: 0 = GDT, 1=LDT
  - RPL: Requested Privileged Level: 00-11
  - CPL: El nivel de privilegio actual –Current Privilege Level– es el RPL de CS
    - Solo es posible acceder a segmentos de datos de mayor nivel de prioridad a través de GATES



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Modo Protegido. Cambio Modo

- Entrar en modo protegido

```
# con interrupciones desabilitadas
        mov     eax, cr0        ; Obtener estado actual
        bts     eax, 0          ; set  bit 0 (bit PE)
        mov     cr0, eax        ; entra modo protegido
```

- Salir del modo protegido

```
# con interrupciones desabilitadas
        mov     eax, cr0        ; Obtener estado actual
        btr     eax, 0          ; reset  bit 0 (bit PE)
        mov     cr0, eax        ; entra modo protegido
```

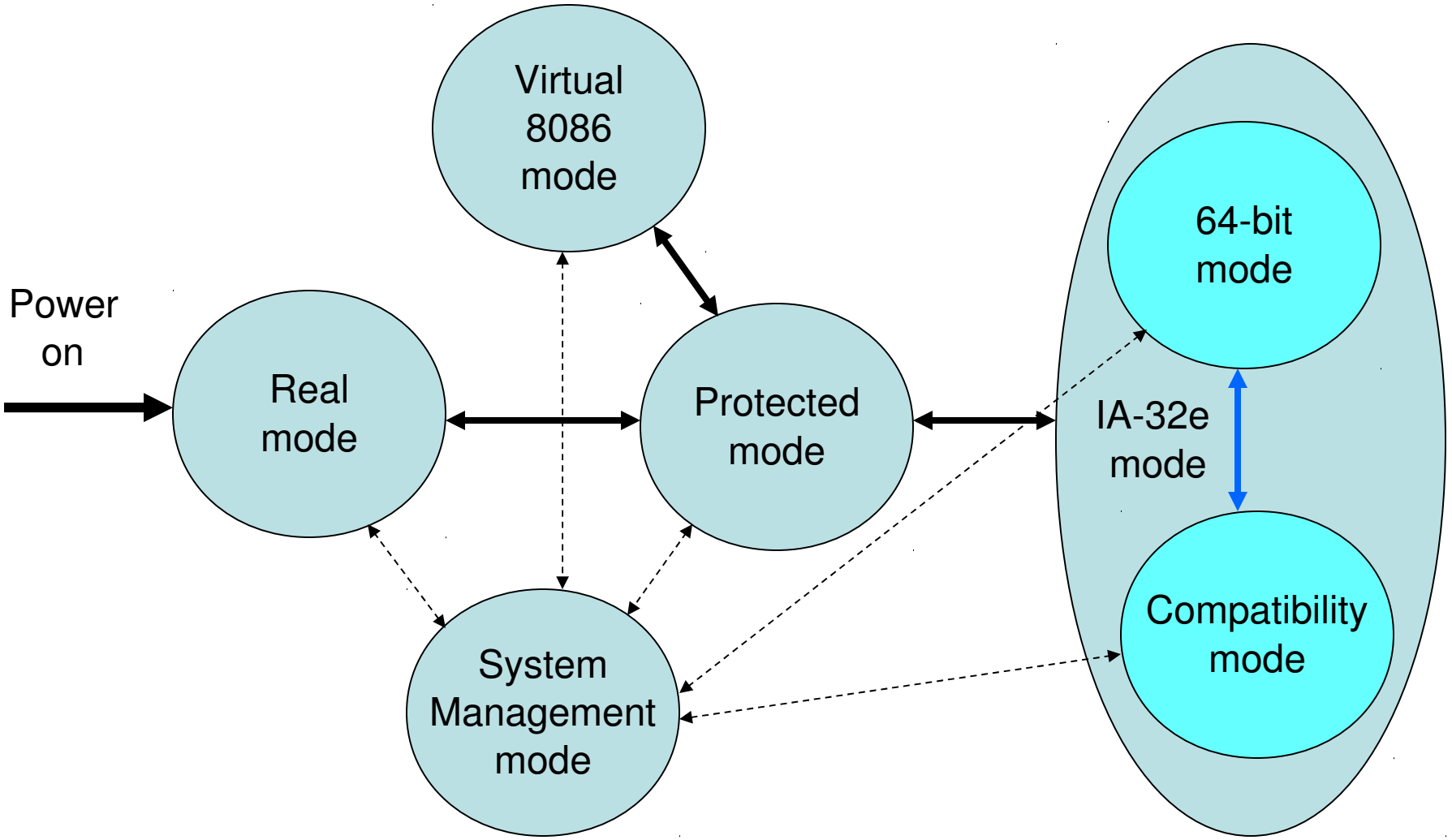

ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

AMD64 - x86-64

- AMD K8 – Arquitectura AMD-64
  - AMD Athlon 64 (2000) y AMD Opteron (2003)
  - Hypertransport
  - Arquitectura 64 bits
    - Registros de Propósito General Adicionales
  - Nuevo modo de operación (Long)
    - Direcciones Virtuales 64 Bits (Memoria Lineal)
    - Memoria Física de hasta 2^52 Bytes – limitado por entradas tabla de pagina –
      - (Actualidad) Memoria Fisica 2^40 Bytes

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

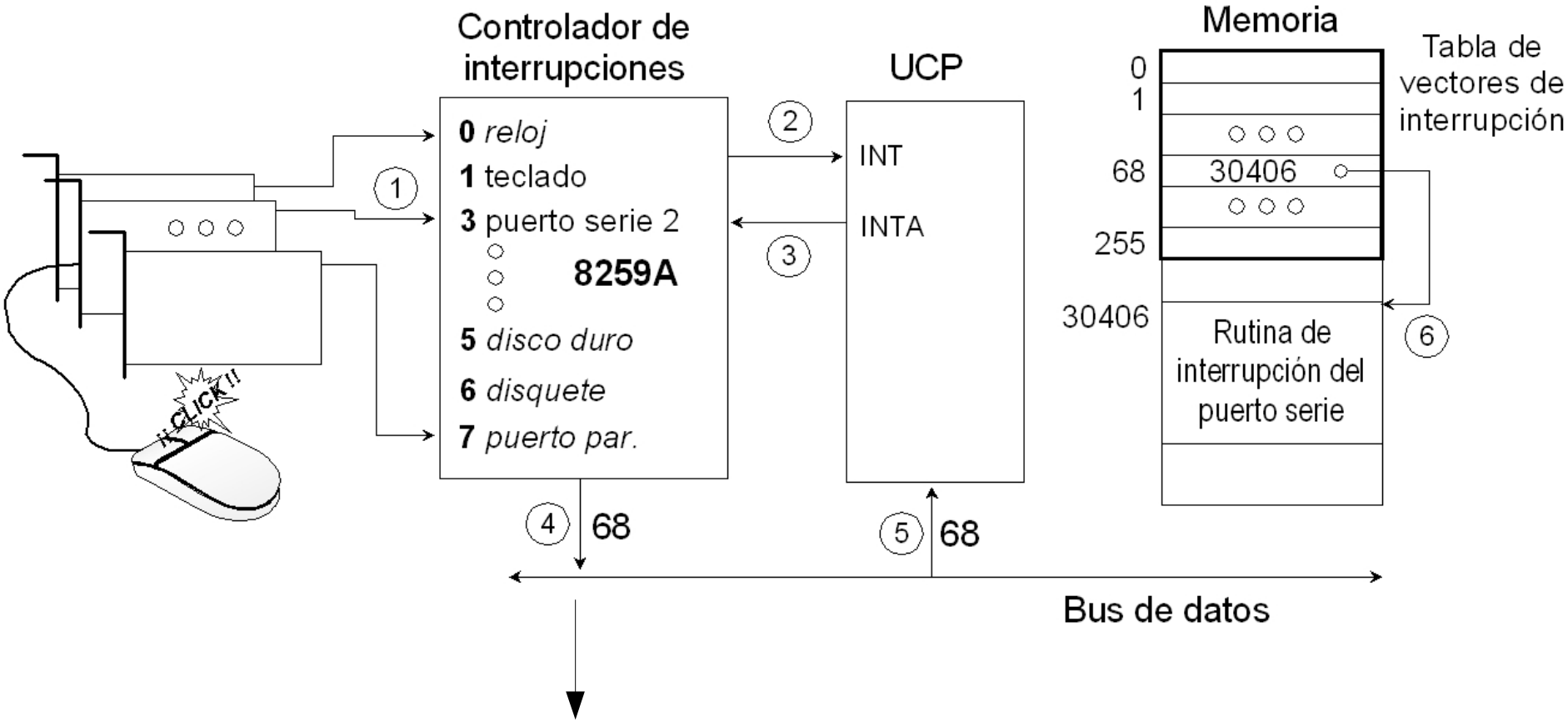
Modos x86-64



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

El hardware de interrupción – 8259

- El IBM PC y IBM XT venían equipados con un único controlador 8259

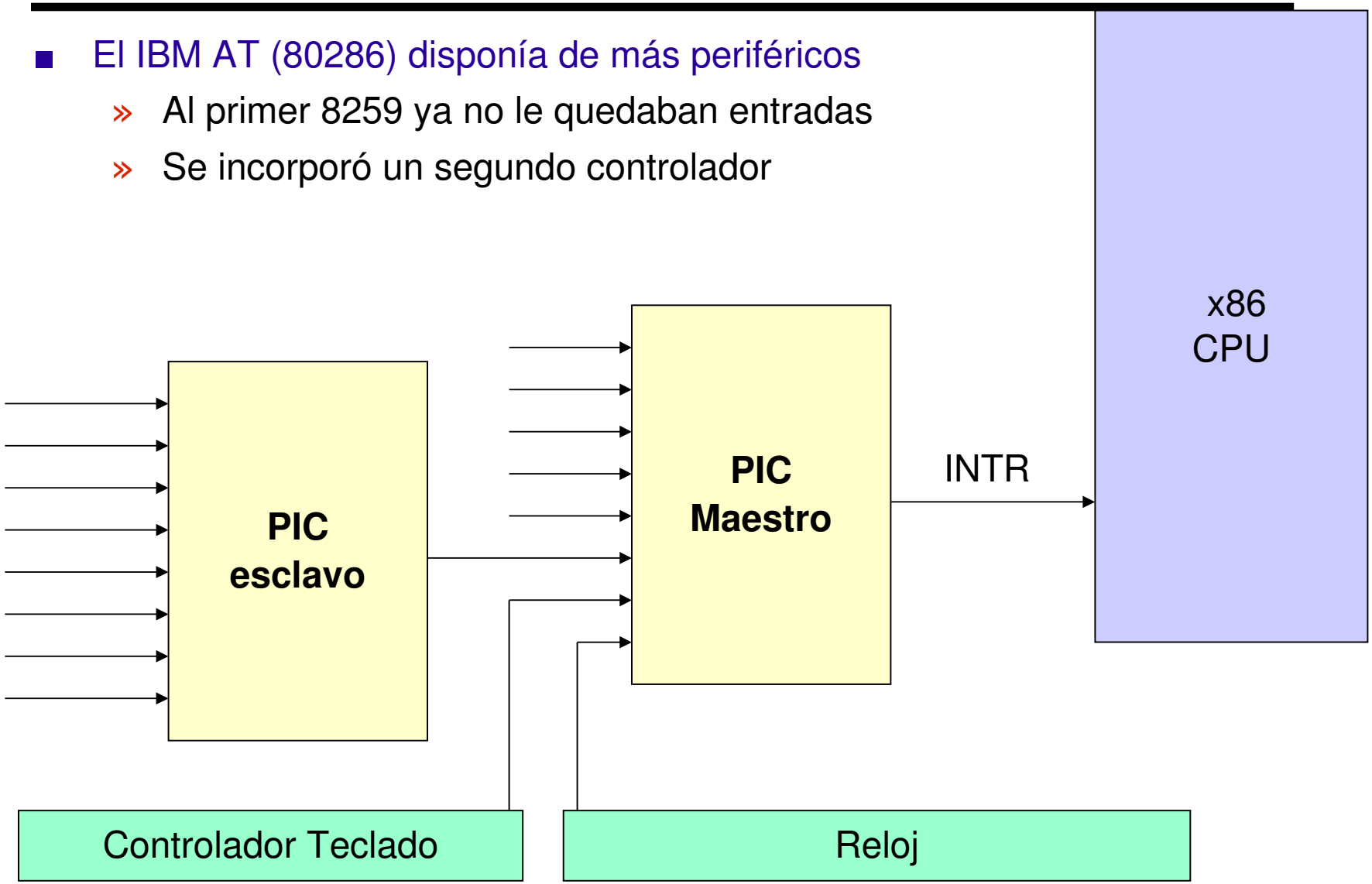


 # 8 bits (programado en la inicializacion del controlador – espacio I/O – puertos

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

El hardware de interrupción – 8259s maestro y esclavo

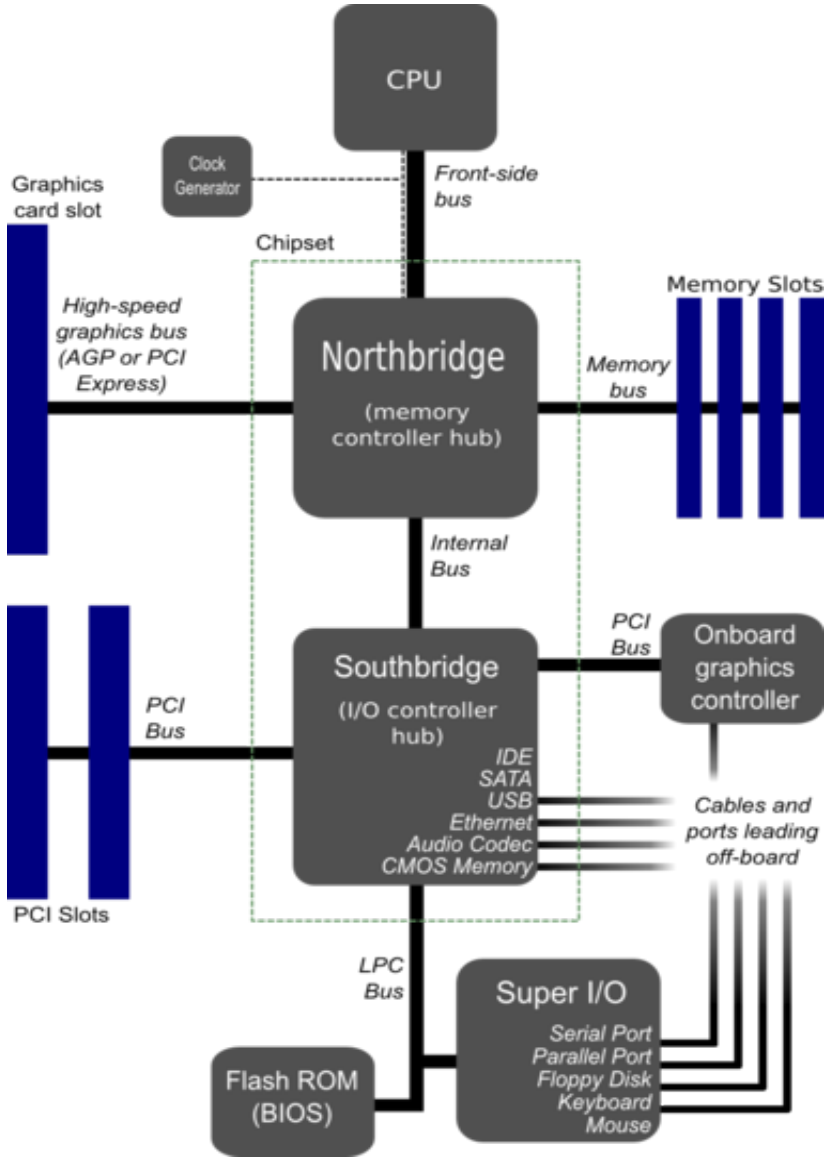
- El IBM AT (80286) disponía de más periféricos
  - » Al primer 8259 ya no le quedaban entradas
  - » Se incorporó un segundo controlador



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

El hardware de interrupción

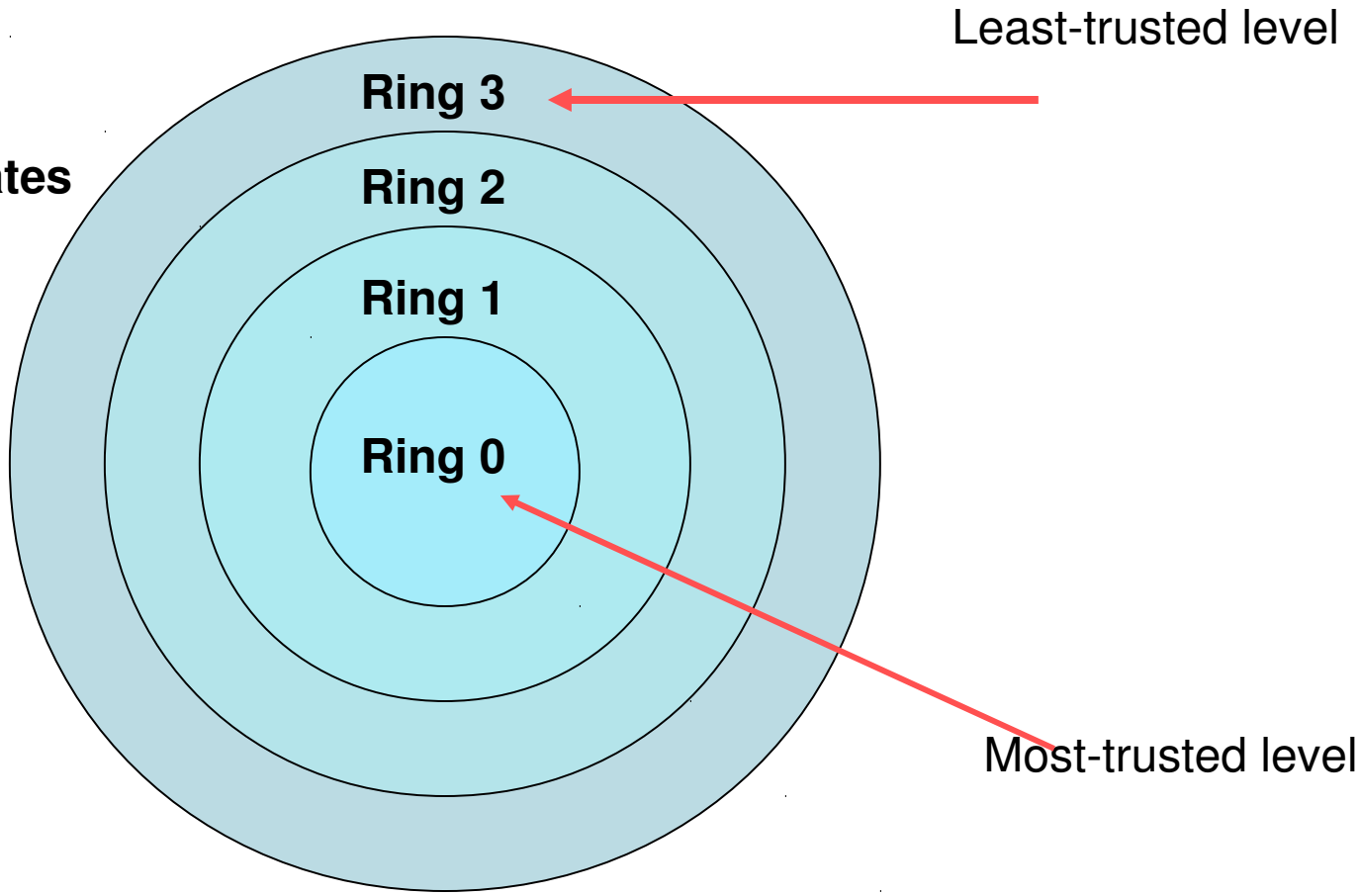
- 8259
  - » integrado en el **southbridge**
- APIC:
  - » Nuevo Controlador Interrupciones
  - » Sistemas multiprocesador



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Interrupciones modo protegido

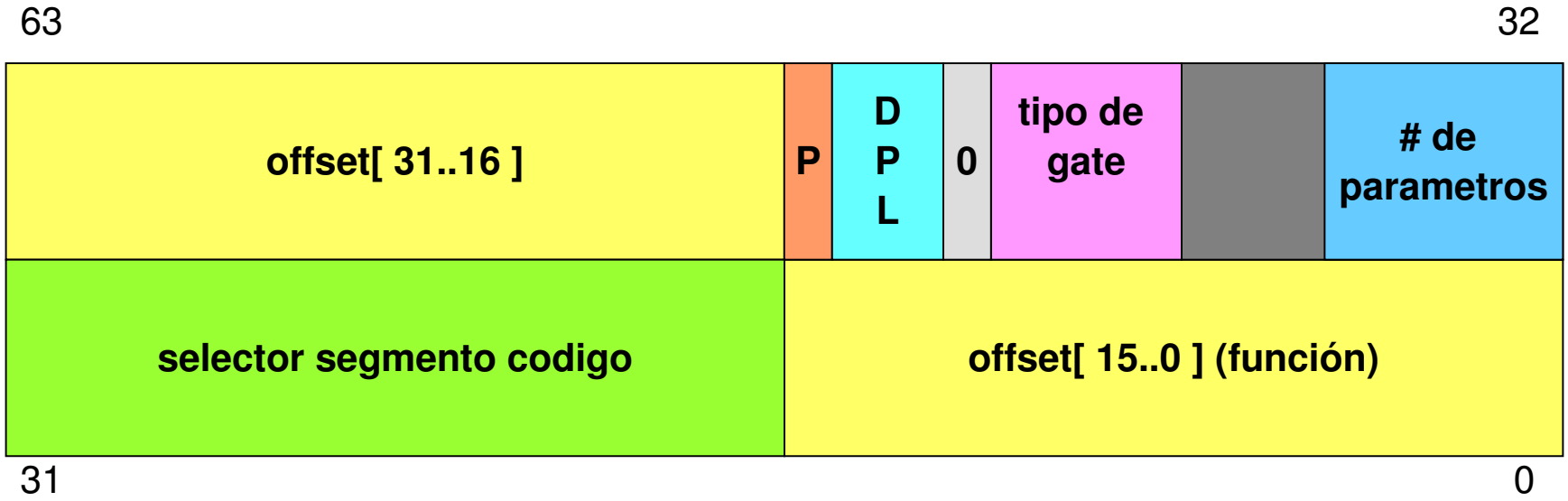
- Cuando tiene lugar una interrupción en modo protegido
  - Es necesario una transición de nivel (y cambio de pila)
- Gates: cambio de nivel
  - Call gates
  - Trap gates
  - **Interrupt gates**
  - Task gates



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Descriptores “Call-Gate”

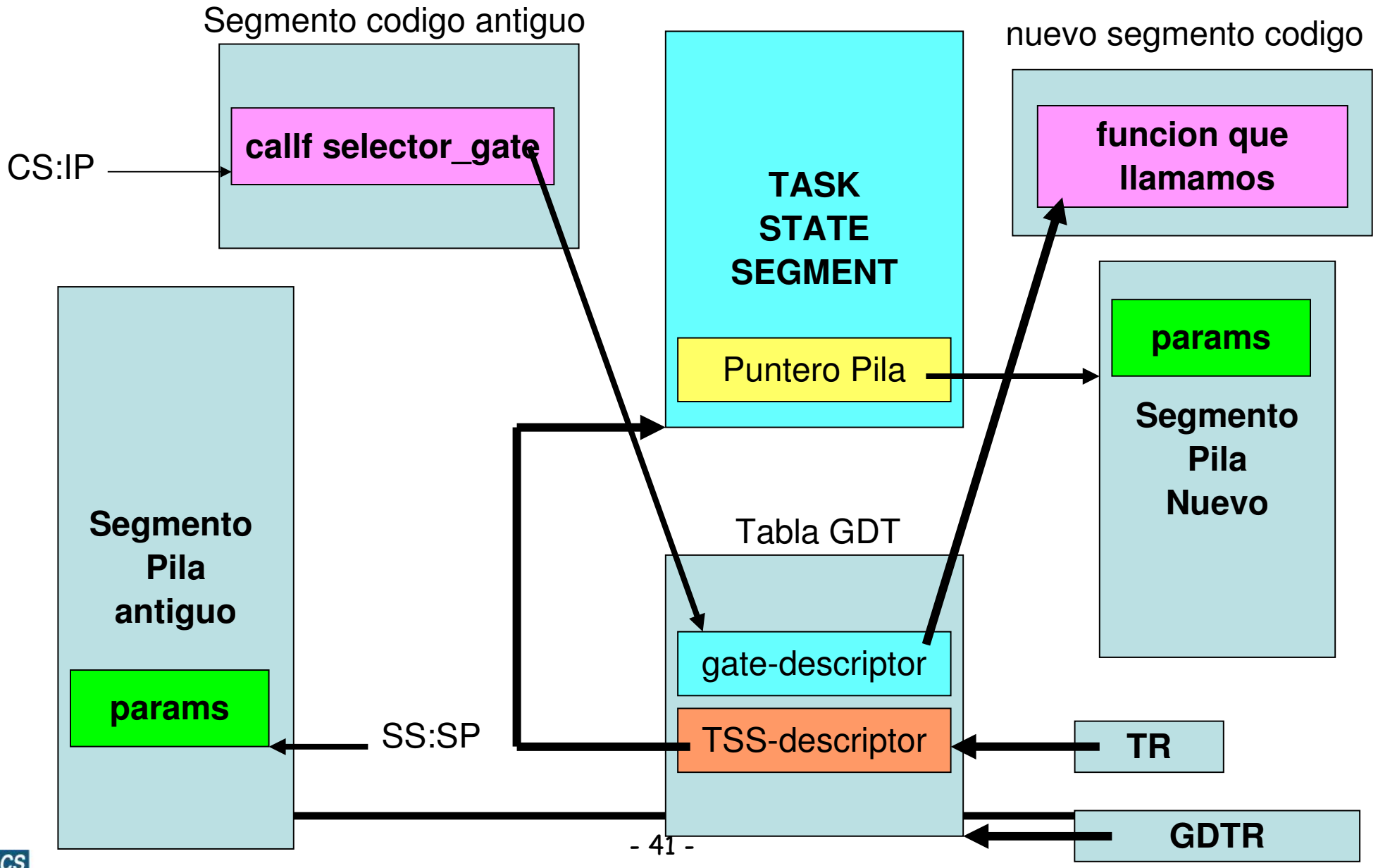
- Descriptores de 8 bytes ~ Descriptores de segmentos



- P=presencia (1=si, 0=no)
- DPL=Descriptor Prvilege Level (0,1,2,3)
- Selector de codigo (segmento que contiene el codigo de la función)
- offset (el punto de entrada a la función)
- # de parametros (parametros que se copiaran)
- tipo de gate (‘0x4’ 16-bit call-gate, ‘0xC’ 32-bit call-gate)

ArTeCS
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

far call: “llamada lejana”



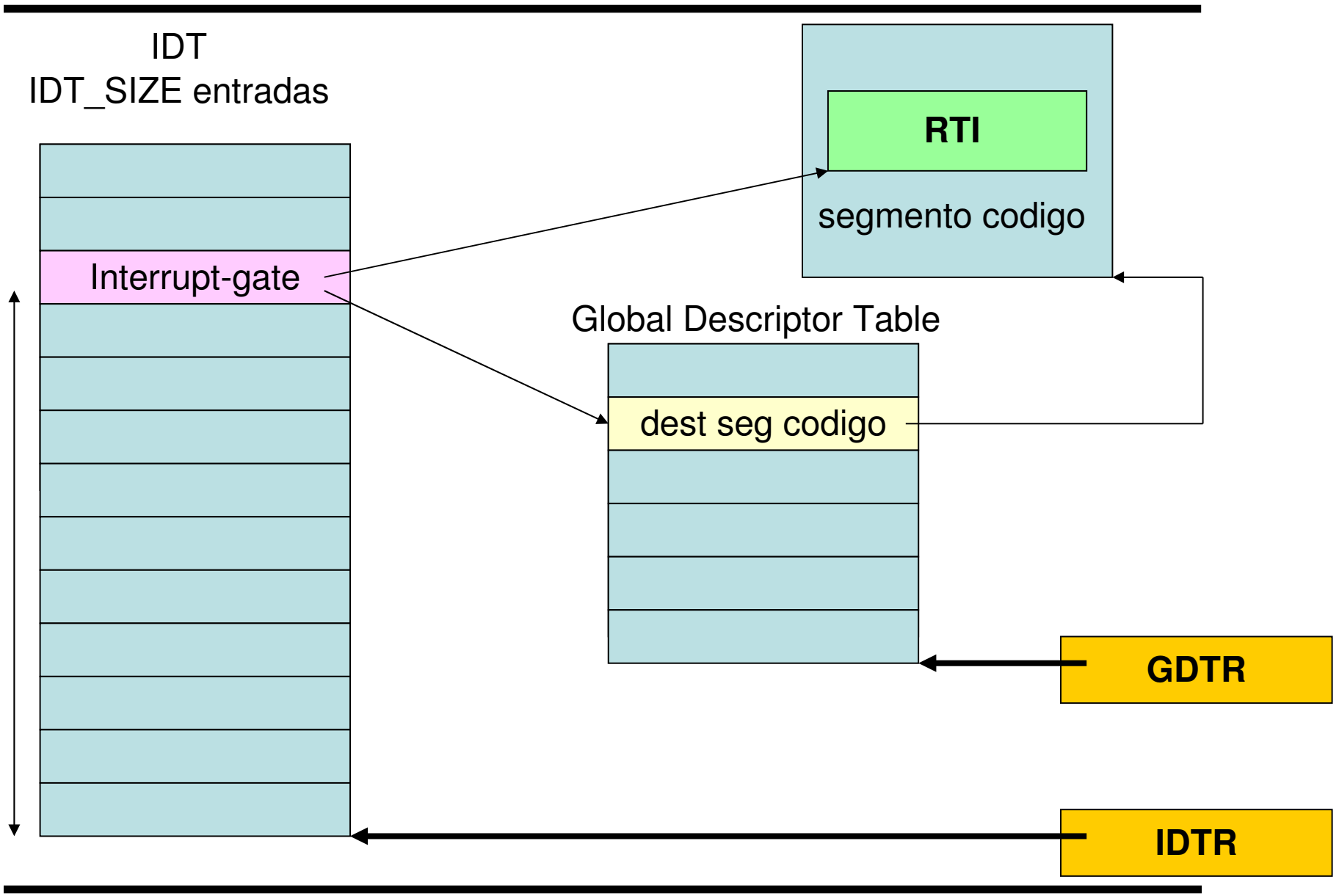
Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Descriptores “Interrupt-Gate”

- P=presencia (1=si, 0=no)
- DPL=Descriptor Prvilege Level (0,1,2,3)
- Selector de codigo (segmento que contiene la rutina tratamiento interrupcion)
- offset (el punto de entrada a la rutina de tratamiento de interrupcion)
- tipo de gate
  - » 32 ó 16 bits

Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid

Tabla de descriptores de interrupción IDT



Manuel Prieto-Matías
Arquitectura Interna Sistemas Operativos
Universidad Complutense de Madrid