Fundusze Europejskie
Wiedza Edukacja Rozwój

Rzeczpospolita Polska

Unia Europejska
Europejski Fundusz Społeczny

Politechnika Śląska jako Centrum Nowoczesnego Kształcenia
opartego o badania i innowacje

POWR.03.05.00-IP.08-00-PZ1/17

Projekt współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego

Microprocessor and Embedded Systems

Faculty of Automatic Control, Electronics and Computer Science, Informatics, Bachelor Degree

Lecture 5

80486
Virtual memory

Bartłomiej Zieliński, PhD, DSc

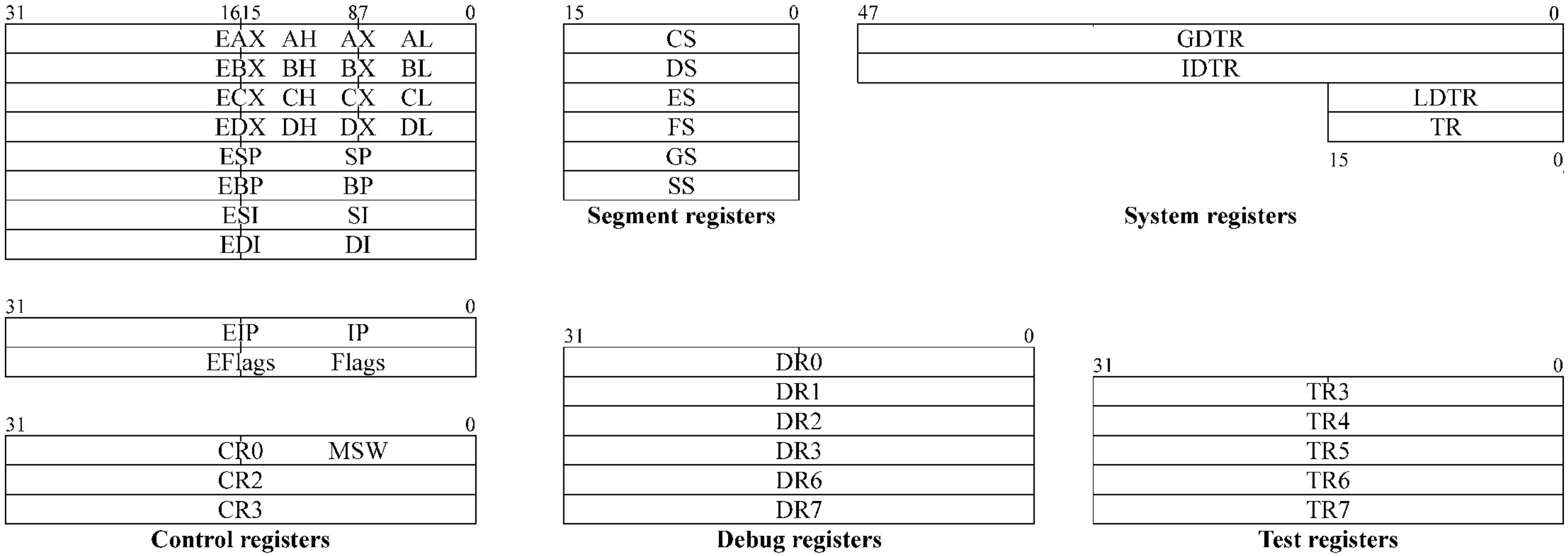
486 virtual memory

Program:

- 486 registers
- Address translation in a protected mode
- Memory segmentation
- Memory paging

486 virtual memory

- 486 registers



31 16 15 8 7 0

EAX AH AX AL

EBX BH BX BL

ECX CH CX CL

EDX DH DX DL

ESP SP

EBP BP

ESI SI

EDI DI

15 0

CS

DS

ES

FS

GS

SS

Segment registers

47 0

GDTR

IDTR

LDTR

TR

15 0

System registers

31 0

EIP IP

EFlags Flags

31 0

CR0 MSW

CR2

CR3

Control registers

31 0

DR0

DR1

DR2

DR3

DR6

DR7

Debug registers

31 0

TR3

TR4

TR5

TR6

TR7

Test registers

+ 8×80-b FPU registers, tag, control, status, …

486 virtual memory

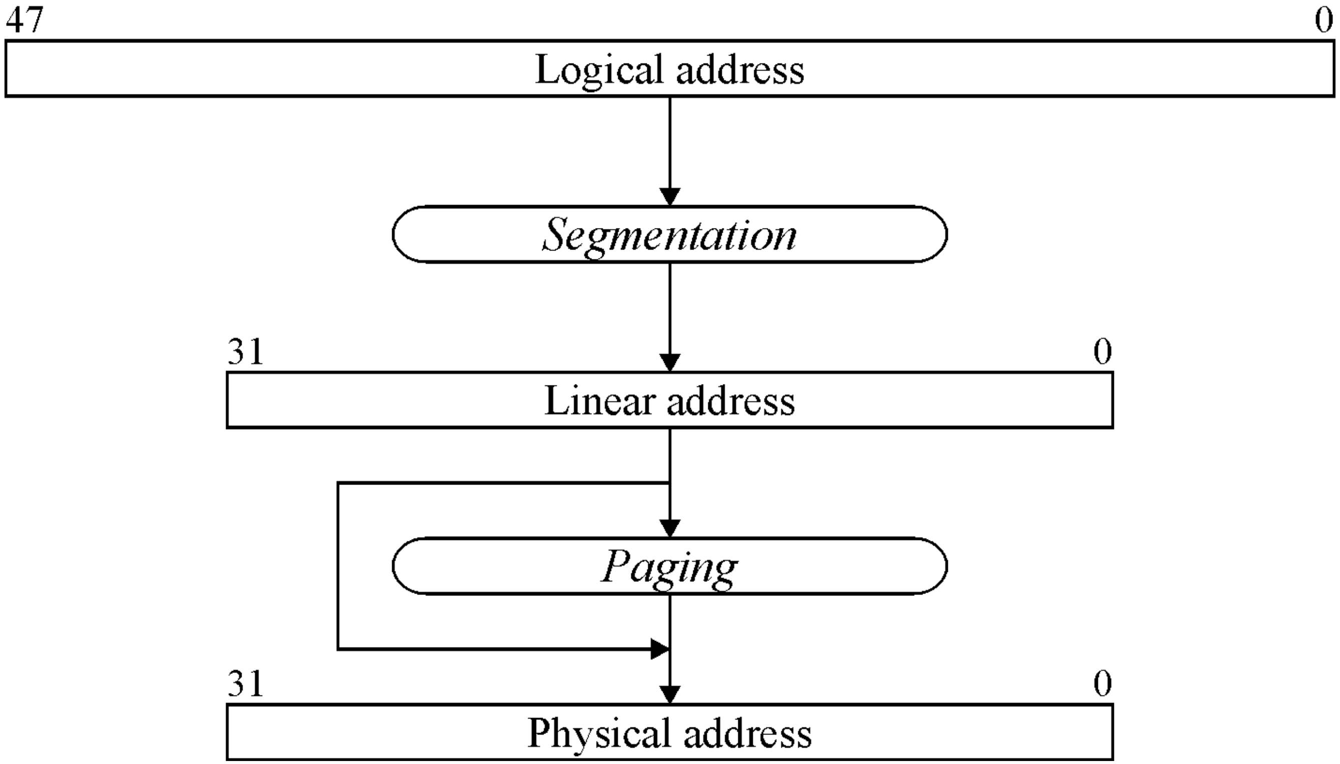
- 486 registers
  - Flags
    - AC – alignment check
    - VM – virtual mode 8086
    - RF – resume flag
    - NT – nested task
    - IOPL – I/O priviledge level
    - OF – overflow
    - DF – direction
    - IF – interrupt
    - TF – trap
    - SF – sign
    - ZF – zero
    - AF – aux carry
    - PF – parity
    - CF – carry

486 virtual memory

- 486 registers
  - CR0 control register
    - PG – paging
    - CD – cache disable
    - NW – non write-through
    - AM – alignment mask
    - WP – write protect
    - NE – numeric error
    - TS – task switch
    - EM – coproc. emulation
    - MP – coproc. monitor
    - PE – protected Mode

486 virtual memory

- Address translation in protected mode



486 virtual memory

- Segmentation
  - Segment registers → segment selectors
    - Point to segment descriptors
  - Descriptor = base address, size, attributes
    - Collected in descriptor tables
    - 1×GDT, *n*×LDT, 1×IDT
    - Address & size: 32-b each

486 virtual memory

- Segmentation
  - Selector & descriptor structure

15	3	2	1	0
Descriptor number		TI	RPL	

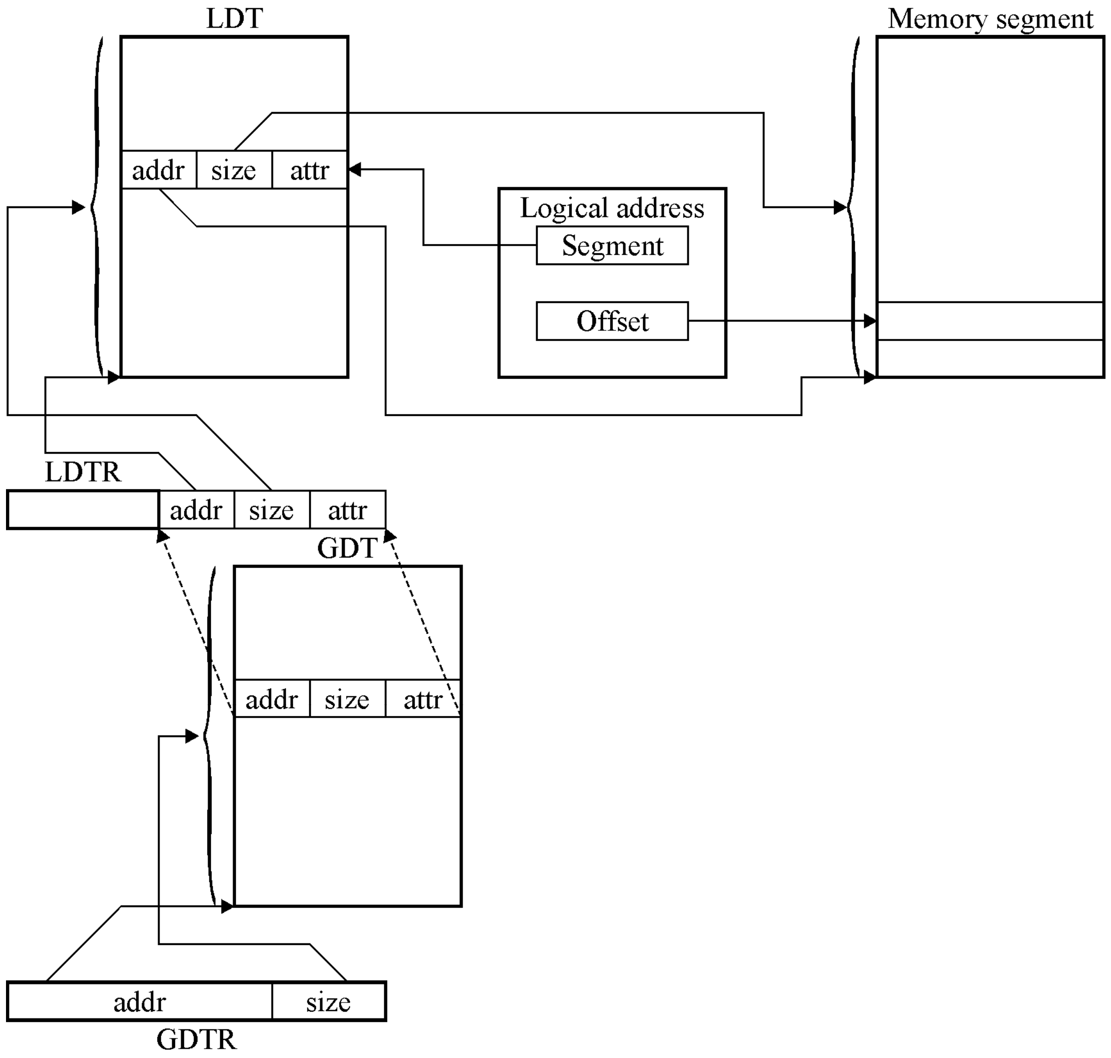
<table>
<tr><td>15</td><td colspan="14"></td><td>0</td></tr>
<tr><td colspan="8">Address [31..24]</td><td>G</td><td>D</td><td>0</td><td>AVL</td><td colspan="4">Size [19..16]</td></tr>
<tr><td>P</td><td colspan="2">DPL</td><td>S</td><td colspan="3">type</td><td>A</td><td colspan="8">Address [23..16]</td></tr>
<tr><td colspan="16">Address [15..0]</td></tr>
<tr><td colspan="16">Size [15..0]</td></tr>
</table>

486 virtual memory

- Segmentation
  - Segment types
    - Data (Rd or Rd/Wr)
    - Data expandable down (Rd or Rd/Wr)
      - E.g., stack in 80x86 family
      - Reverse Size field interpretation
    - Code (Ex or Ex/Rd)
    - Conforming code (Ex or Ex/Rd)
      - No priviledge level change after call to such a segment

486 virtual memory

- Segmentation



486 virtual memory

- Segmentation
  - Approximate (false) calculation
    - An application can use:
      - 8192 segments described in GDT
      - 8192 segments described in LDT
    - Σ=16K segments × 4 GB each

      = 64 TB virtual addressing space for each application
    - But:
      - LDT – system segment described in GDT
      - GDT contains descriptors of other system segments
      - Etc.

486 virtual memory

- Segmentation
  - Disadvantages
    - Variable segment length
      - Is it really a disadvantage?
      - Makes memory management harder
      - Memory fragmentation
    - No „segment written” information

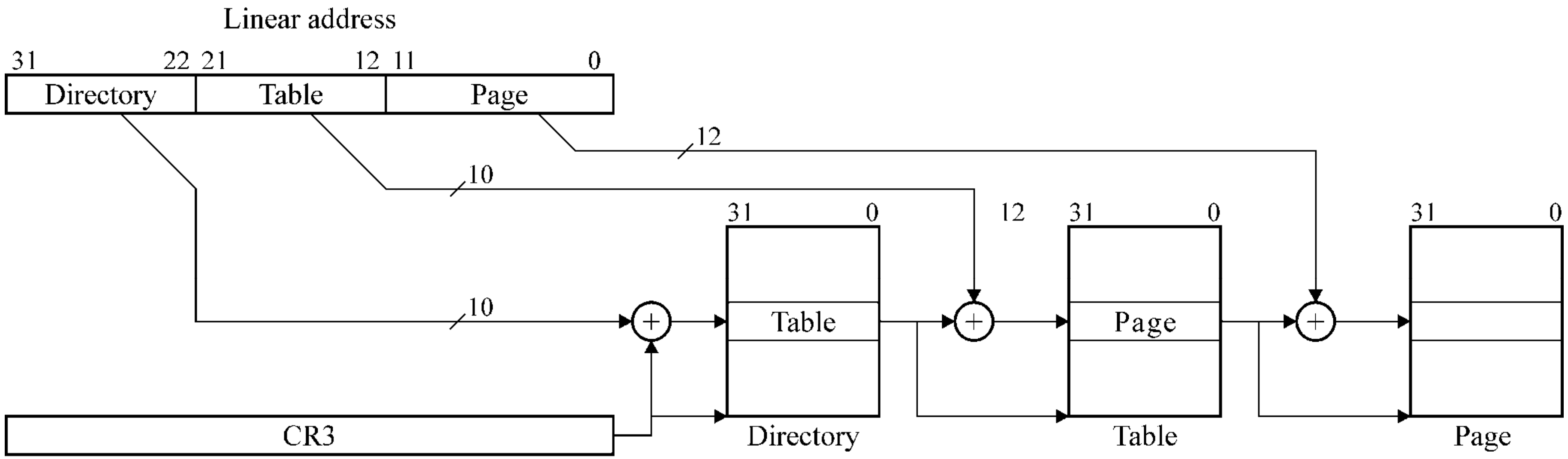
    - Both solved by paging

486 virtual memory

- Segmentation
  - Splits memory into variable-size fragments
  - With direct regard to logical program structure
- Paging
  - Splits memory into uniform-size fragments
  - With no regard to logical program structure
  - Page size = 4 KB

486 virtual memory

- Paging



31 … 12	11 … 9	8 7	6	5	4	3	2	1	0
Address [12..31]	AVL	00	D	A	PWT	PCD	R/W	U/S	P

486 virtual memory

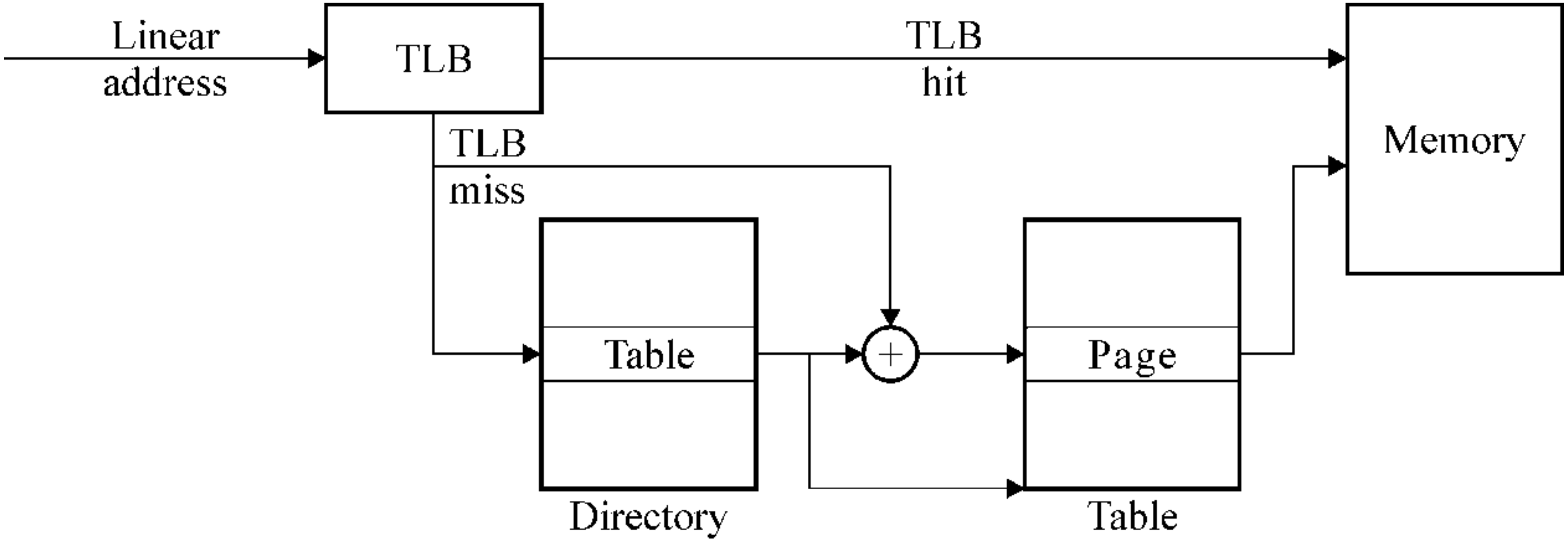
- Paging
  - Page access rights

$U/\overline{S}$	$W/\overline{R}$	WP	User	Supervisor
0	0	0		Rd/Wr/Ex
0	1	0		Rd/Wr/Ex
1	0	0	Rd/Ex	Rd/Wr/Ex
1	1	0	Rd/Wr/Ex	Rd/Wr/Ex
0	0	1		Rd/Ex
0	1	1		Rd/Wr/Ex
1	0	1	Rd/Ex	Rd/Ex
1	1	1	Rd/Wr/Ex	Rd/Wr/Ex

  - Cache enabled: PCD=CD=$\overline{\text{KEN}}$=0

486 virtual memory

- Paging
  - TLB (Transaction Look-aside Buffer)



486 virtual memory

- Segmentation vs. paging – summary

	Page	Segment
Size	4 KB	1 B..4 GB
Priviledge levels	2	4
Base address	4 KB multiple	Any
„Dirty” bit	Yes	No
Access bit	Yes	Yes
Present bit	Yes	Yes
R/W protection	Yes	Yes