

Segmentation

What happens if program increase their size in their execution?

How to manage expanding and contracting tables?

How to protect only data from the program?

How to share data to other program or functions?

The general solution of these issues is to provide the machine with many completely independent address spaces, called segments.

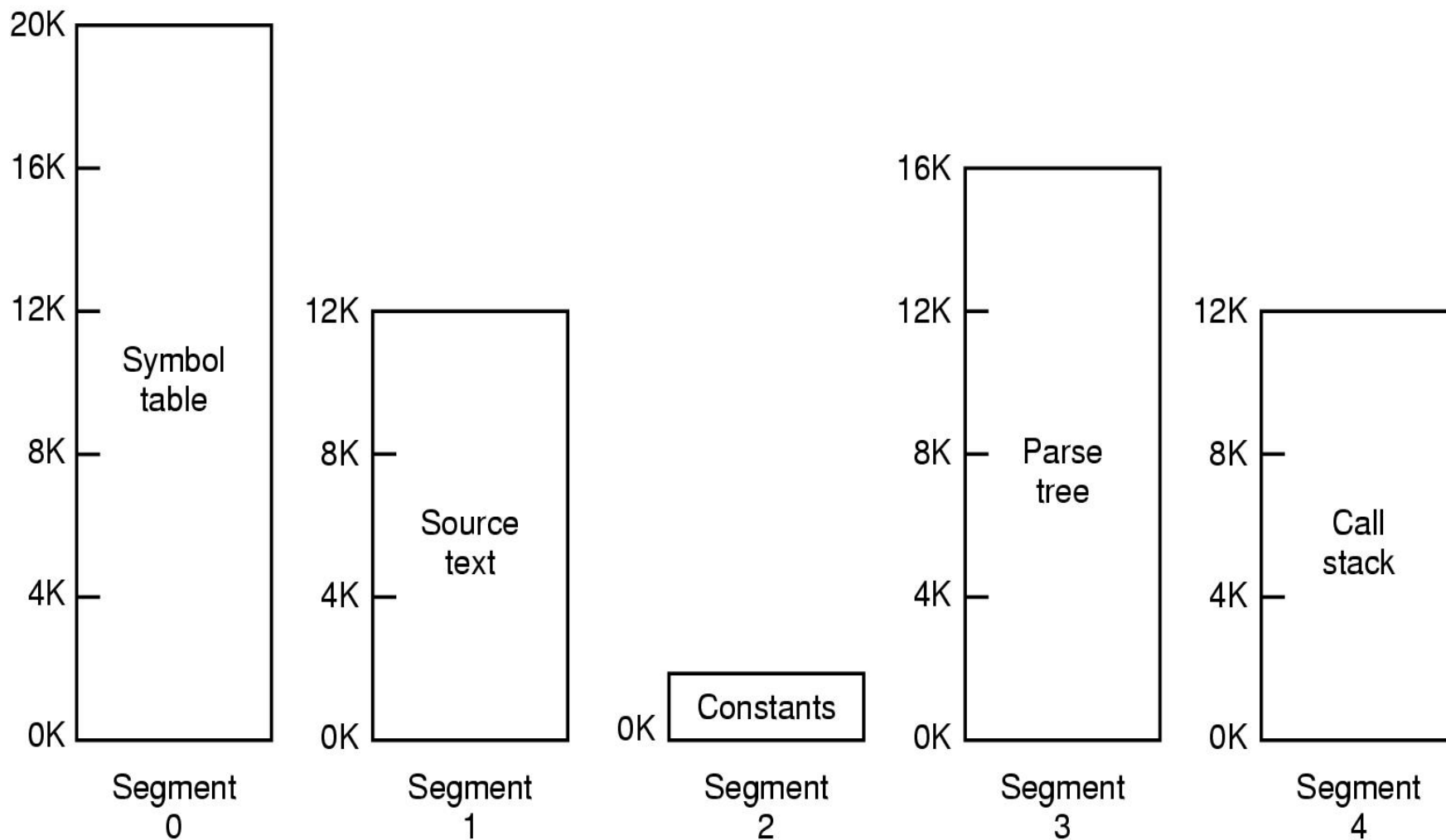
Segmentation

Memory management that support variable partitioning and mechanisms with freedom of contiguous memory requirement restriction.

The independent block of the program is a segment such as: main program, procedures, functions, methods, objects, local variables, global variables, common blocks, stacks, symbol table, arrays.

The responsibility for dividing the program into segments lies with user (or compiler).

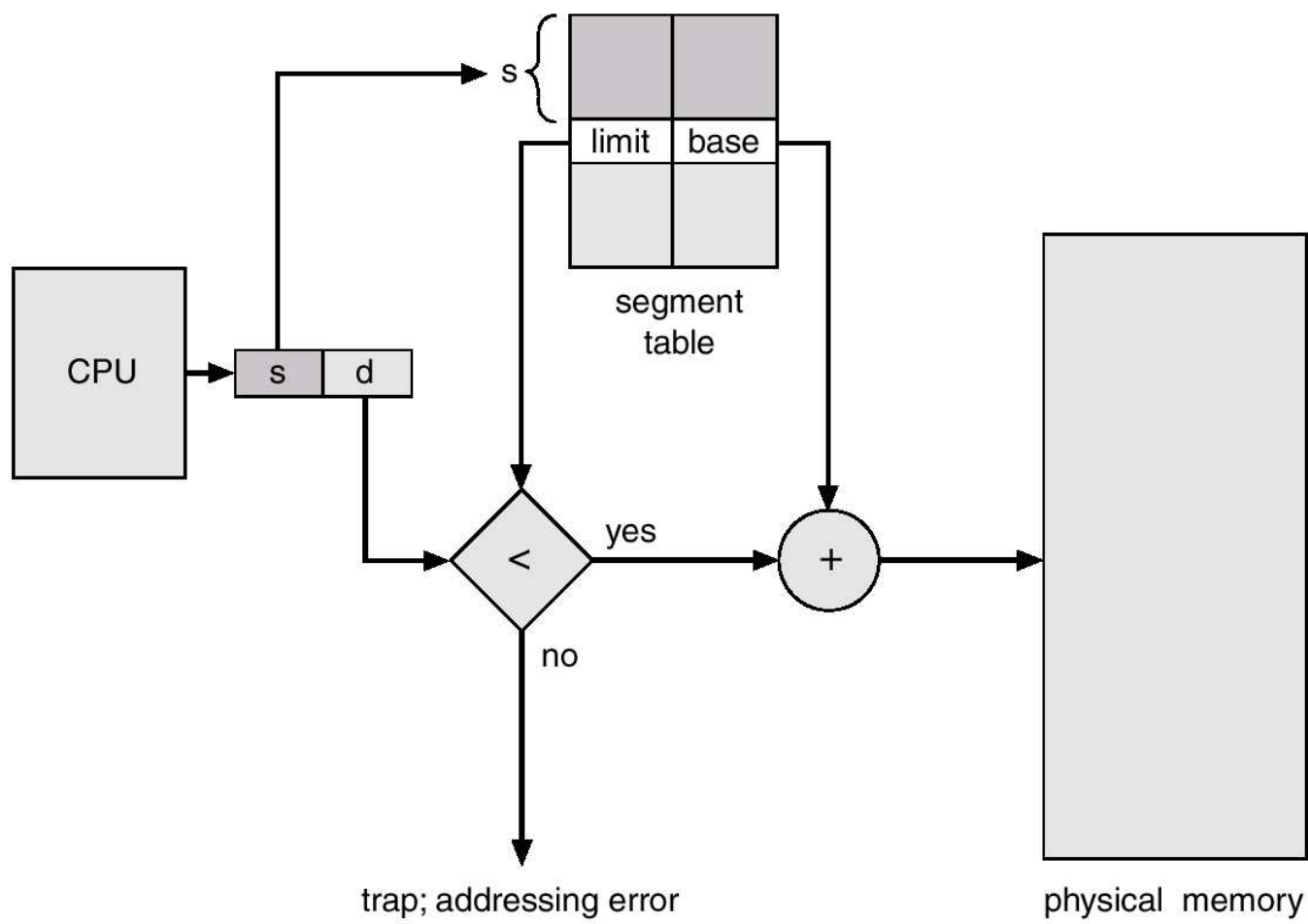
Segmentation



Segmentation

- Different segments have its own name and size.
- The different segment can grow or shrink independently, with out effecting the others; so the size of segment changed during execution.
- For the simplicity of implementation, segments are numbered and are referred to by a segment number, rather than by segment name. Thus the logical address consist: *segment number* and *offset*.
- The segment table (like page table but each entry consist limit and base register value) is used to map the logical address to physical address.

Segmentation

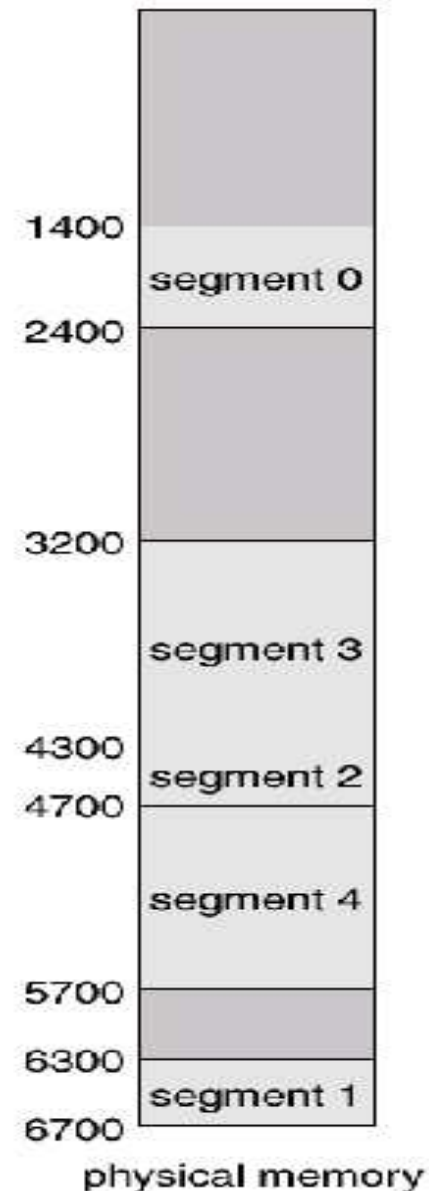


Segmentation

The segment number used as index into the segment table. The offset d of the logical address must be between 0 and the segment limit. If not, trap occur, if it is legal it is added to the segment base to produce the address in the physical memory.

| | limit | base |
|---|-------|------|
| 0 | 1000 | 1400 |
| 1 | 400 | 6300 |
| 2 | 400 | 4300 |
| 3 | 1100 | 3200 |
| 4 | 1000 | 4700 |

segment table



The segmentation scheme causes fragmentation, this can be handle by same technique of variable partition memory management.

Paging vs. Segmentation

| Consideration | Paging | Segmentation |
|--|--|--|
| Need the programmer be aware that this technique is being used? | No | Yes |
| How many linear address spaces are there? | 1 | Many |
| Can the total address space exceed the size of physical memory? | Yes | Yes |
| Can procedures and data be distinguished and separately protected? | No | Yes |
| Can tables whose size fluctuates be accommodated easily? | No | Yes |
| Is sharing of procedures between users facilitated? | No | Yes |
| Why was this technique invented? | To get a large linear address space without having to buy more physical memory | To allow programs and data to be broken up into logically independent address spaces and to aid sharing and protection |

Segmentation with Paging

What happen when segment are larger than main memory?

Segmentation can be combined with paging to provide the efficiency of paging with the protection and sharing capabilities of segmentation.

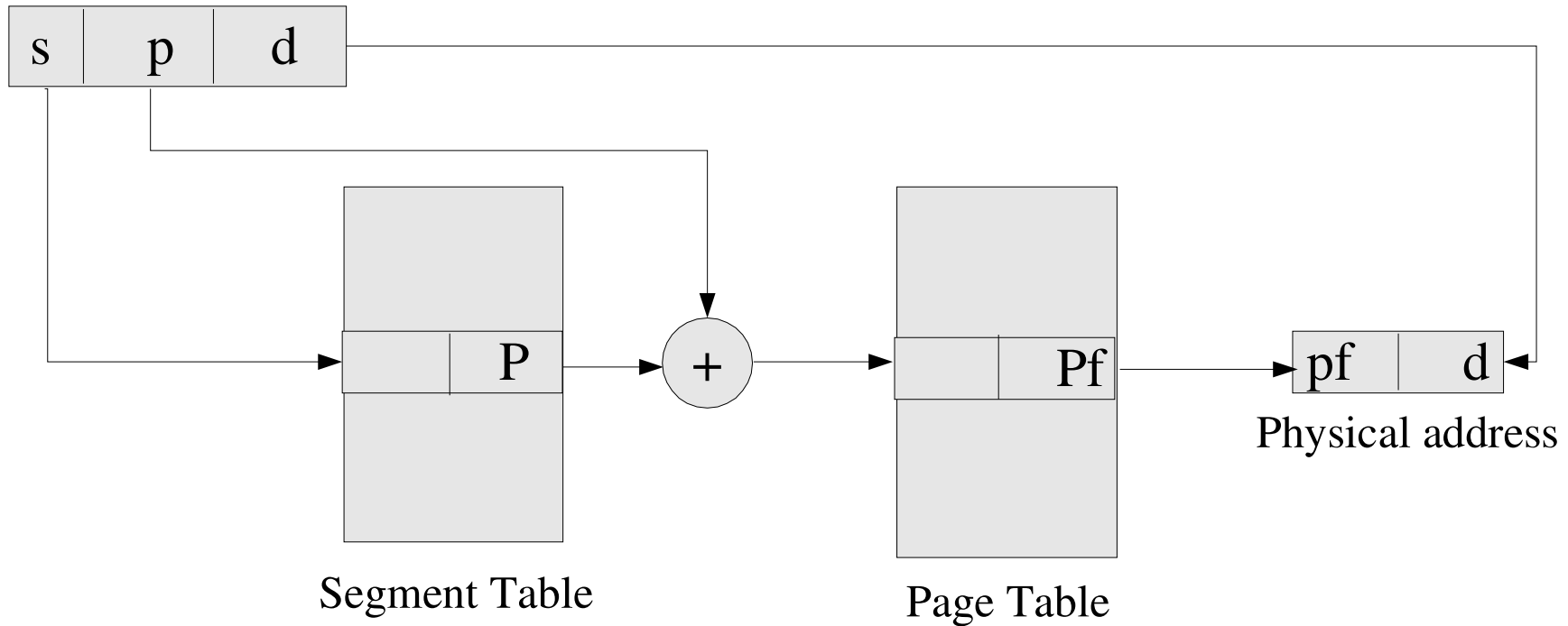
As with simple segmentation, the logical address specifies the segment number and the offset within the segment.

When paging is added, the segment offset is further divided into a page number and page offset.

The segment table entry contains the address of the segment's page table.

Segmentation with Paging

Logical address



Segmentation with Paging

Examples:

The Intel Pentium:

The Intel Pentium 80386 and later architecture uses segmentation with paging memory management.

The maximum number of segments per process is 16K, and each segment can be large as 4 GB. The page size is 4K. It use two-level paging scheme.

Multics:

It has 256K independent segments, and each up to 64K. The page size is 1K or small.

Home Works

HW#10:

1. Q. 36, & 37 from Textbook Tanenbaum.
2. Why are segmentation and paging sometimes combined into one scheme?
3. Consider the following segment table:

| <u>segment</u> | <u>base</u> | <u>size</u> |
|----------------|-------------|-------------|
| 0 | 219 | 600 |
| 1 | 2300 | 14 |
| 2 | 90 | 100 |
| 3 | 1327 | 580 |
| 4 | 1952 | 96 |

What are the physical address for the following logical address?

a) 0430 b) 110 c) 2500 d) 3400 e) 4112

4. Distinguish the paging and segmentation.