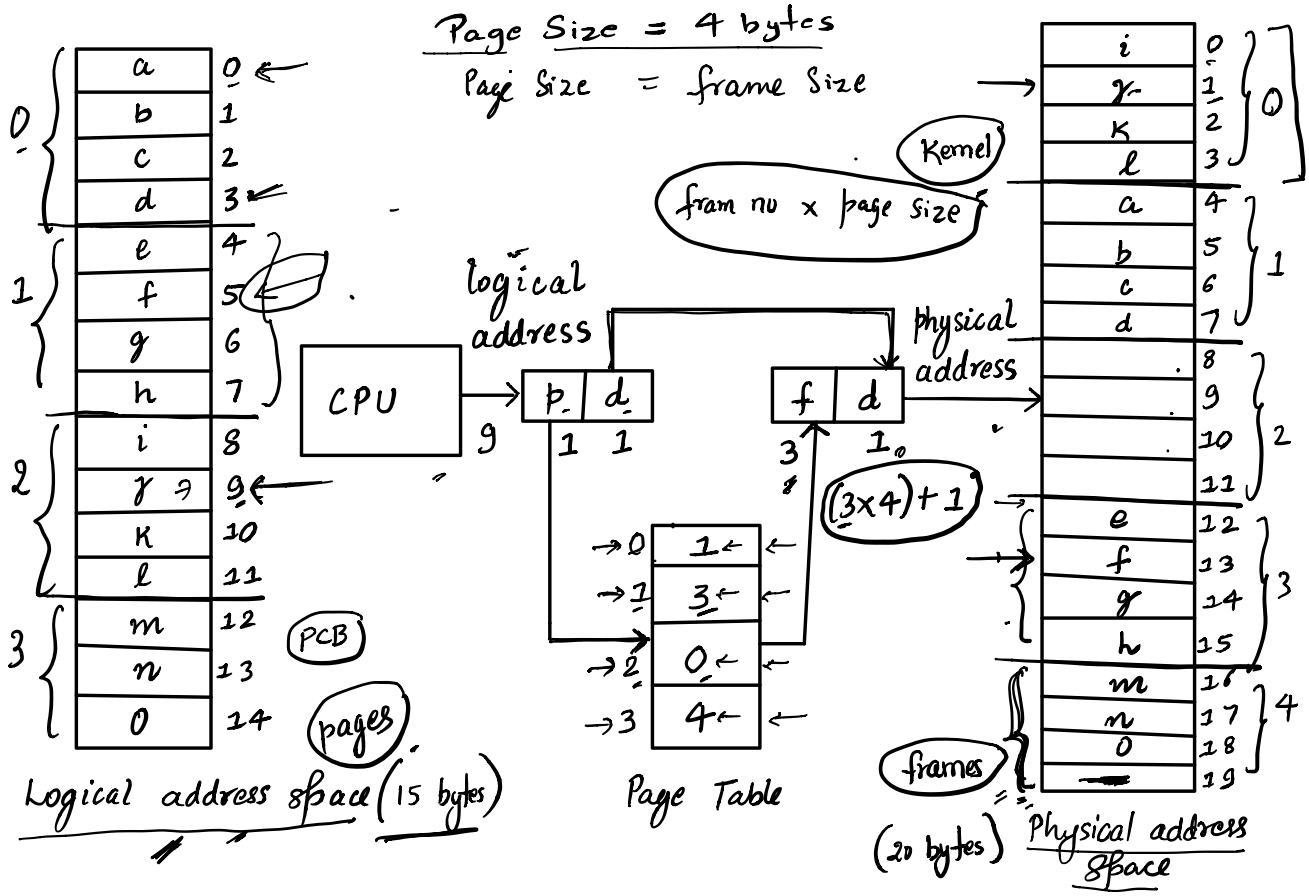
PAGING TECHNIQUE FOR NONCONTIGUOUS MEMORY ALLOCATION

- Memory management technique that allows the physical address space of a process to be noncontiguous
- Avoids external fragmentation and minimises internal fragmentation. How?
- Assume byte addressable memory in all examples
- Byte addressable means every byte has its own unique memory address.
- Word addressable memory also exists - 4 bits, 8 bits, 16 bits etc

How Paging is implemented?

physical → frames
logical → pages

- Break down physical memory into fixed sized blocks called frames
- Break down logical memory of a process into blocks of same size as frames called pages
- Page size(frame size) is defined by the hardware - 8KB or 16KB
- When executing a process, load the pages of the process into frames which are free
- If the process requires N pages, at least N free frames must be available in main memory. Demand Paging avoids this limitation.
- Page table is used for storing the mapping between pages and frames for a process
- Logical address is divided into two parts - page number and offset/displacement within page
- MMU combines the page table mappings with the page offset to arrive at correct physical address



- page number = logical address / page size
- page offset = logical address % page size

2.9 → 2

- These computations can be simplified if the page size is a power of 2. We will cover this in next video
- number of pages = logical memory size / page size
- number of frames = physical memory size / page size
- Avoids external fragmentation
- May have some internal fragmentation
- Using Paging is similar to using a table of base/relocation registers