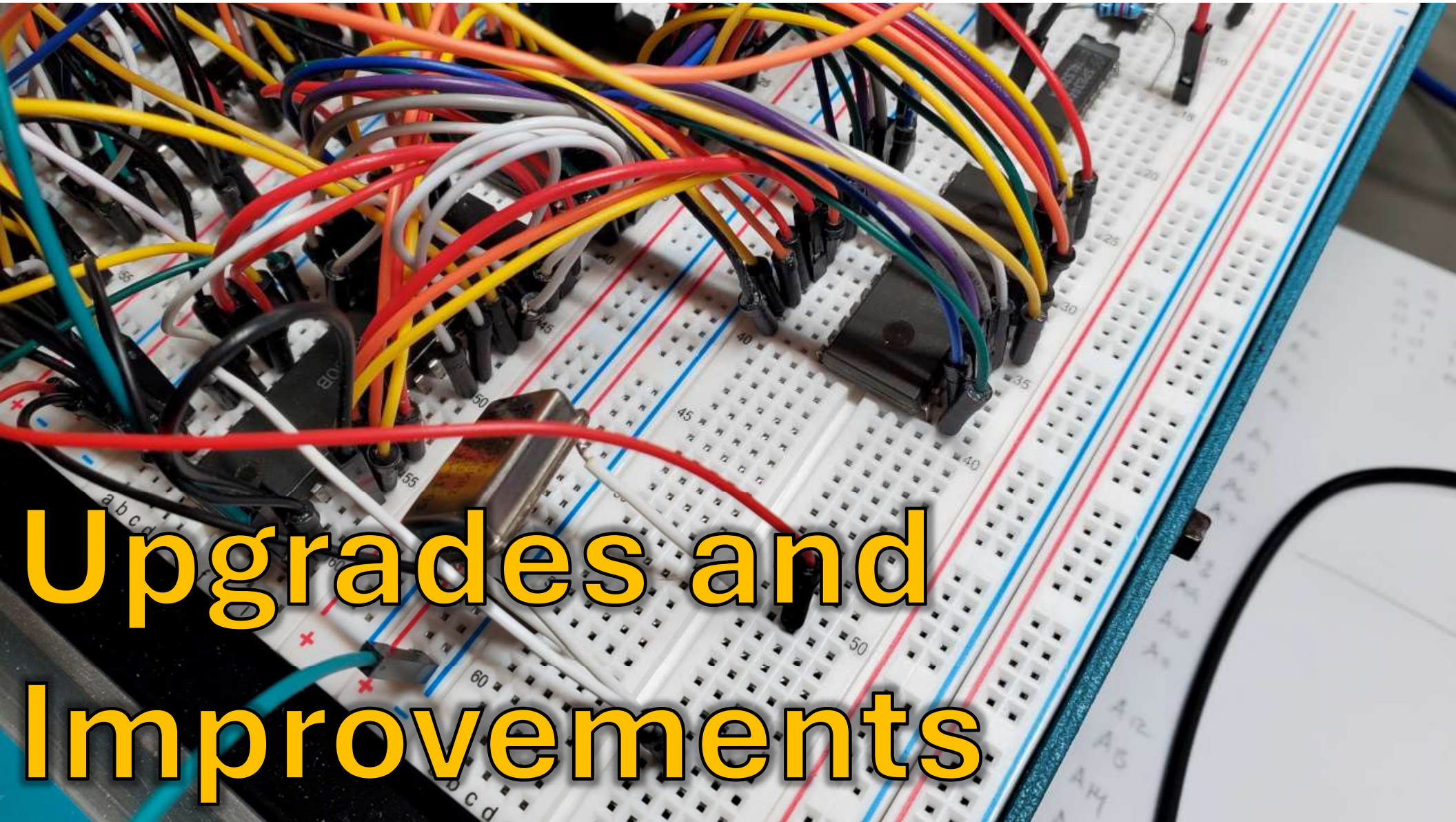
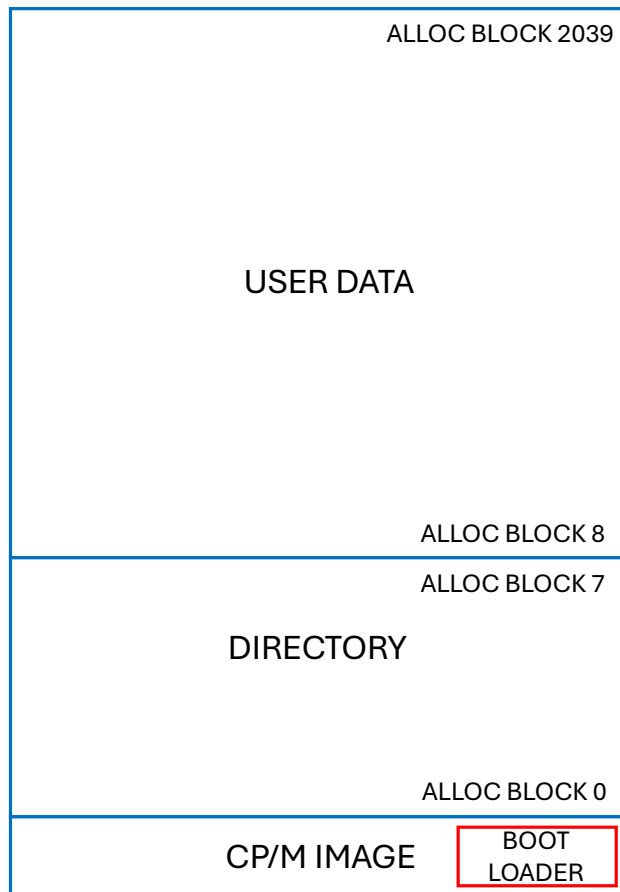




**Upgrades and
Improvements**

CP/M 2.2 File System

My 8080 Disk Layout



TRACK 255

TRACK 1

TRACK 0

Notes on my CP/M Disks

- CP/M Track and Sector are always offset 0
- CP/M Allocation blocks are offset 0
 - My block size is 4096 bytes
 - Each block holds 32 sectors
- Track 0 Sectors 0-3 are reserved for the CP/M on-disk boot loader
 - Monitor loader is drastically simplified
- Track 0 Sectors 4 through 255 are reserved for CP/M memory Image
- Track 1 allocation blocks 0 through 7 are reserved for CP/M directory
- Allocation blocks 8 through 2039 contain user files

CP/M 2.2 Boot Loader

My 8080 TRACK 0 Layout

TRACK 0
LOG SEC 57-255

UNUSED

6.8K bytes

TRACK 0
LOG SEC 4-56

CP/M IMAGE

TRACK 0
PHY SEC 0
LOG SEC 0-3

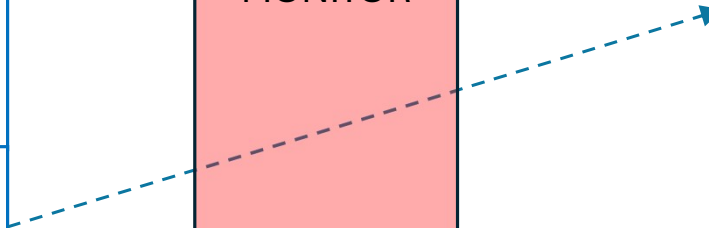
512 bytes
BOOT LOADER

MONITOR

My 8080 RAM

BOOT LOADER

\$1000



CP/M 2.2 Boot Loader

My 8080 TRACK 0 Layout

TRACK 0
LOG SEC 57-255

UNUSED

TRACK 0
LOG SEC 4-56

6.8K bytes

CP/M IMAGE

TRACK 0
PHY SEC 0
LOG SEC 0-3

512 bytes

BOOT LOADER

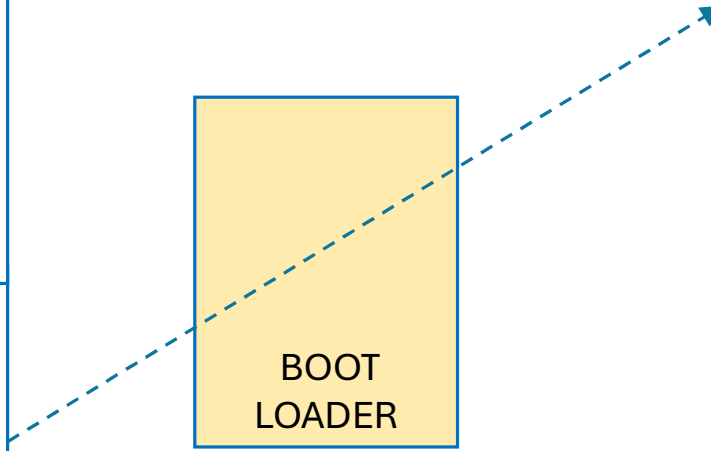
BOOT
LOADER

My 8080 RAM

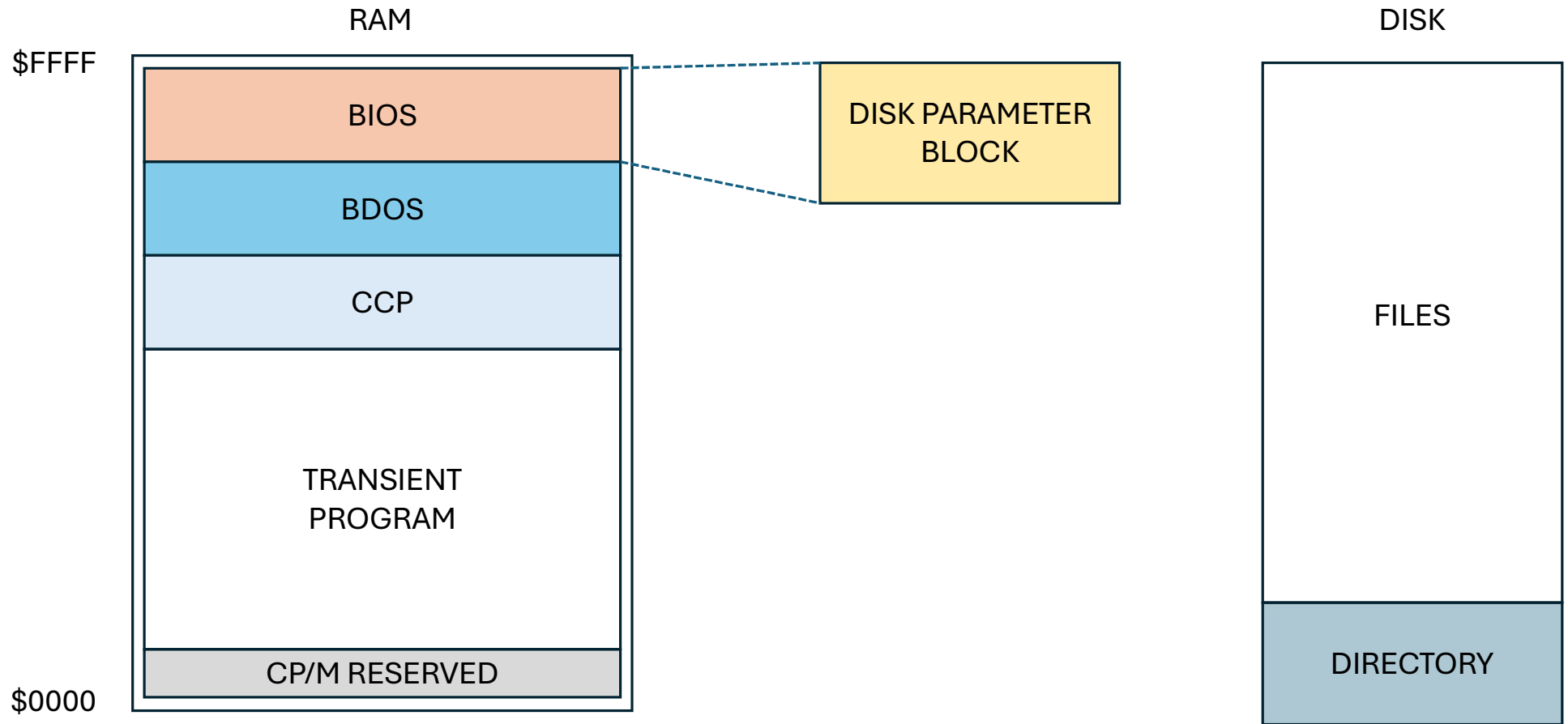
CP/M IMAGE

BOOT LOADER

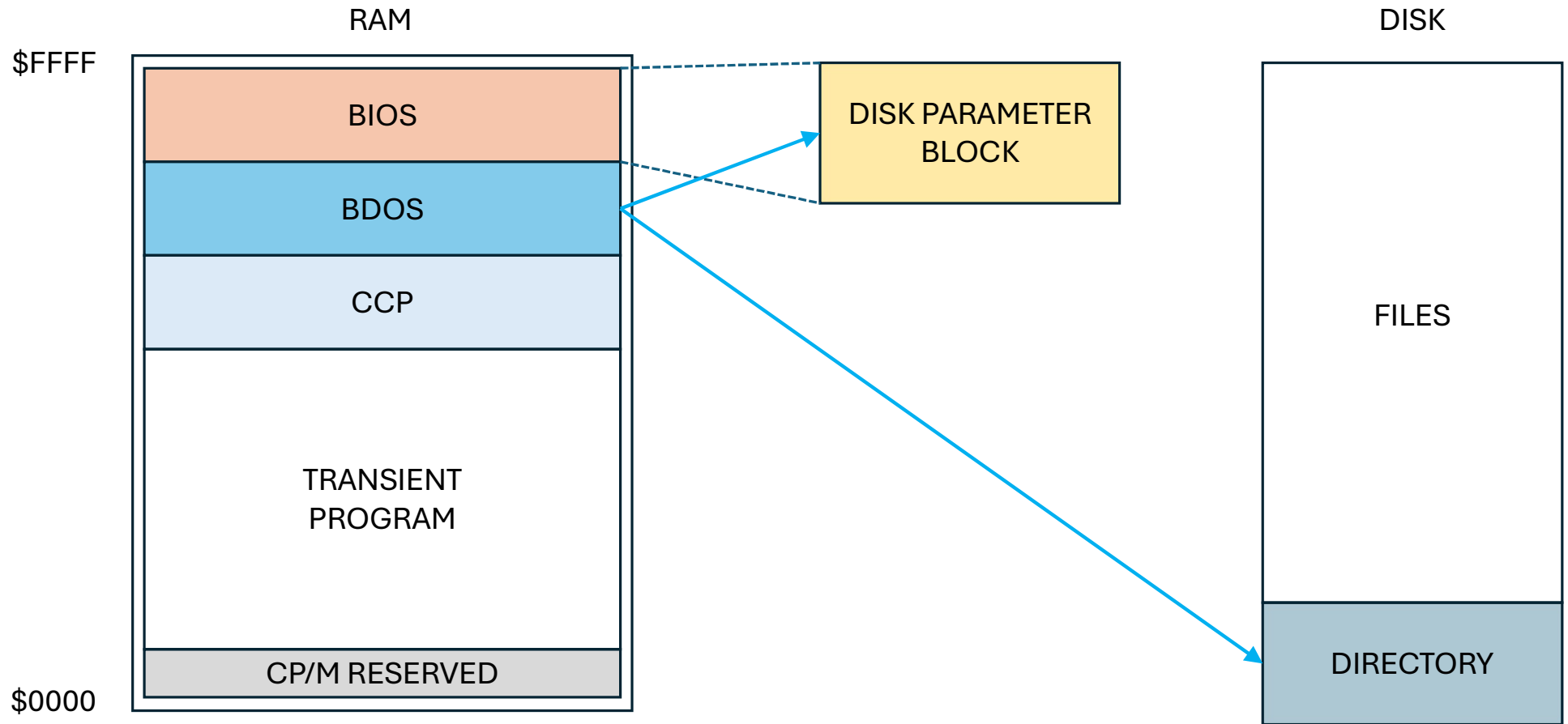
\$1000



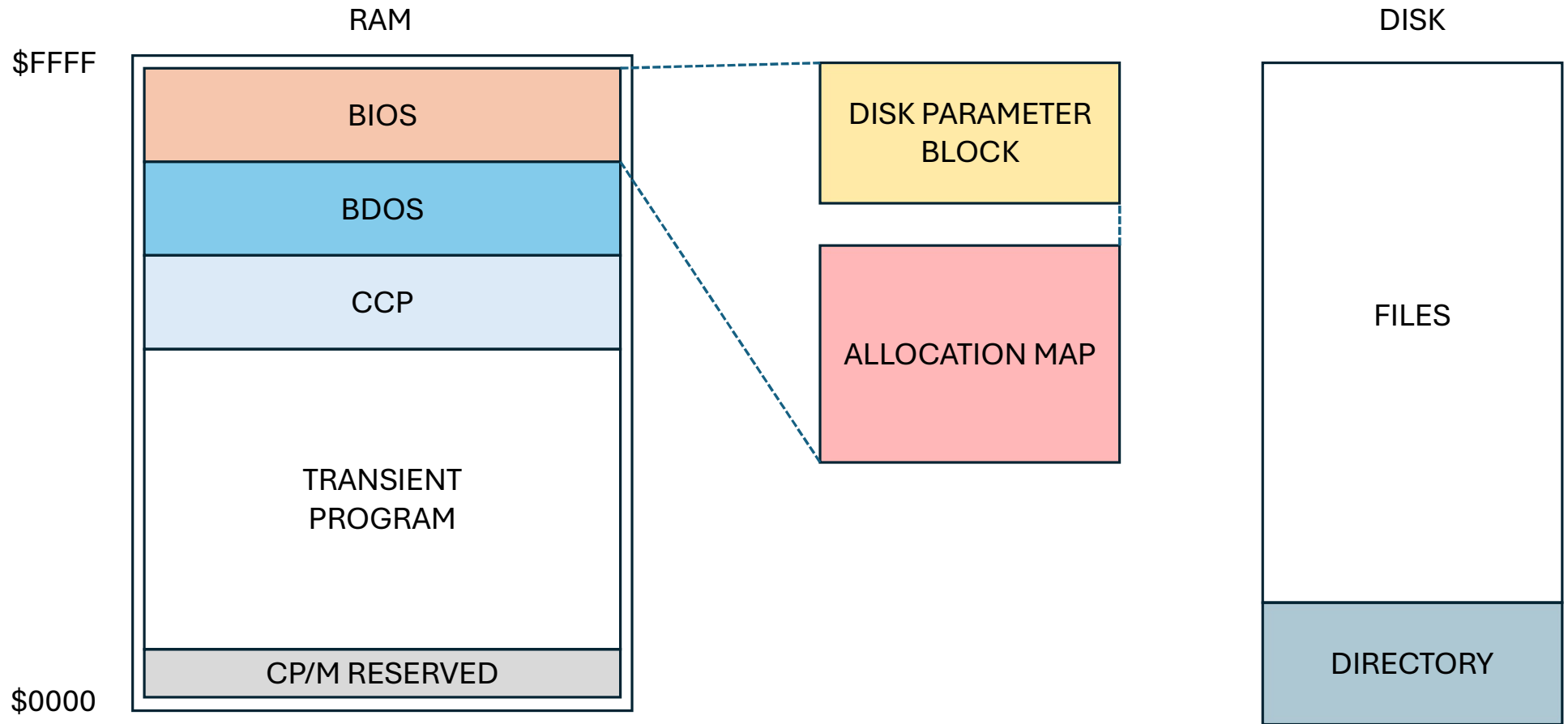
CP/M 2.2 File System



CP/M 2.2 File System



CP/M 2.2 File System



CP/M 2.2 File System

Allocation Vector - This table is created every time a disk is logged in

DISK: A TRACK: 01 START SECTOR: 00 END SECTOR: 7F BLOCK: 0000

[illegible]

Allocation Vector table is required for each drive letter available (A, B, ...)

One bit is required for each Allocation Block in the file system

My machine requires 256 bytes PER DRIVE LETTER (2040 blocks (bits) / 8 bits/byte)

CP/M 2.2 File System

This is Block 0
(First Block)

DISK: A TRACK: 01 START SECTOR: 00 END SECTOR: 7F BLOCK: 0000

[illegible]

This is block 2039
(Last Block)

CP/M 2.2 File System

1 - Indicates used block

Directory

DISK: A TRACK: 01 START SECTOR: 00 END SECTOR: 7F BLOCK: 0000

```

7654321076543210765432107654321076543210765432107654321076543210
0000 11111111111111111111111111111111000000000000000000000000000000
0008 000000000000000000000000000000000000000000000000000000000000
0010 0000000000000000000000000000000000000000000000000000000000
0018 0000000000000000000000000000000000000000000000000000000000
0020 0000000000000000000000000000000000000000000000000000000000
0028 0000000000000000000000000000000000000000000000000000000000
0030 0000000000000000000000000000000000000000000000000000000000
0038 0000000000000000000000000000000000000000000000000000000000

```

User Files

0 - Indicates unused block

CP/M 2.2 File System

Allocation Vector Memory Sizes

1024 Block Size

8" Single Sided Floppy
77 Tracks, 26 Sectors
2 Reserved Tracks
128 Bytes/Sector
243 Blocks
31 bytes per disk

Hard Disk
256 Tracks, 256 Sectors
1 Reserved Track
512 Bytes/Sector
8160 Blocks
1020 bytes per disk

2048 Block Size

8" Single Sided Floppy
77 Tracks, 26 Sectors
2 Reserved Tracks
128 Bytes/Sector
121 Blocks
16 bytes per disk

Hard Disk
256 Tracks, 256 Sectors
1 Reserved Track
512 Bytes/Sector
4080 Blocks
510 bytes per disk

4096 Block Size

8" Single Sided Floppy
77 Tracks, 26 Sectors
2 Reserved Tracks
128 Bytes/Sector
243 Blocks
8 bytes per disk

Hard Disk
256 Tracks, 256 Sectors
1 Reserved Track
512 Bytes/Sector
2040 Blocks
255 bytes per disk

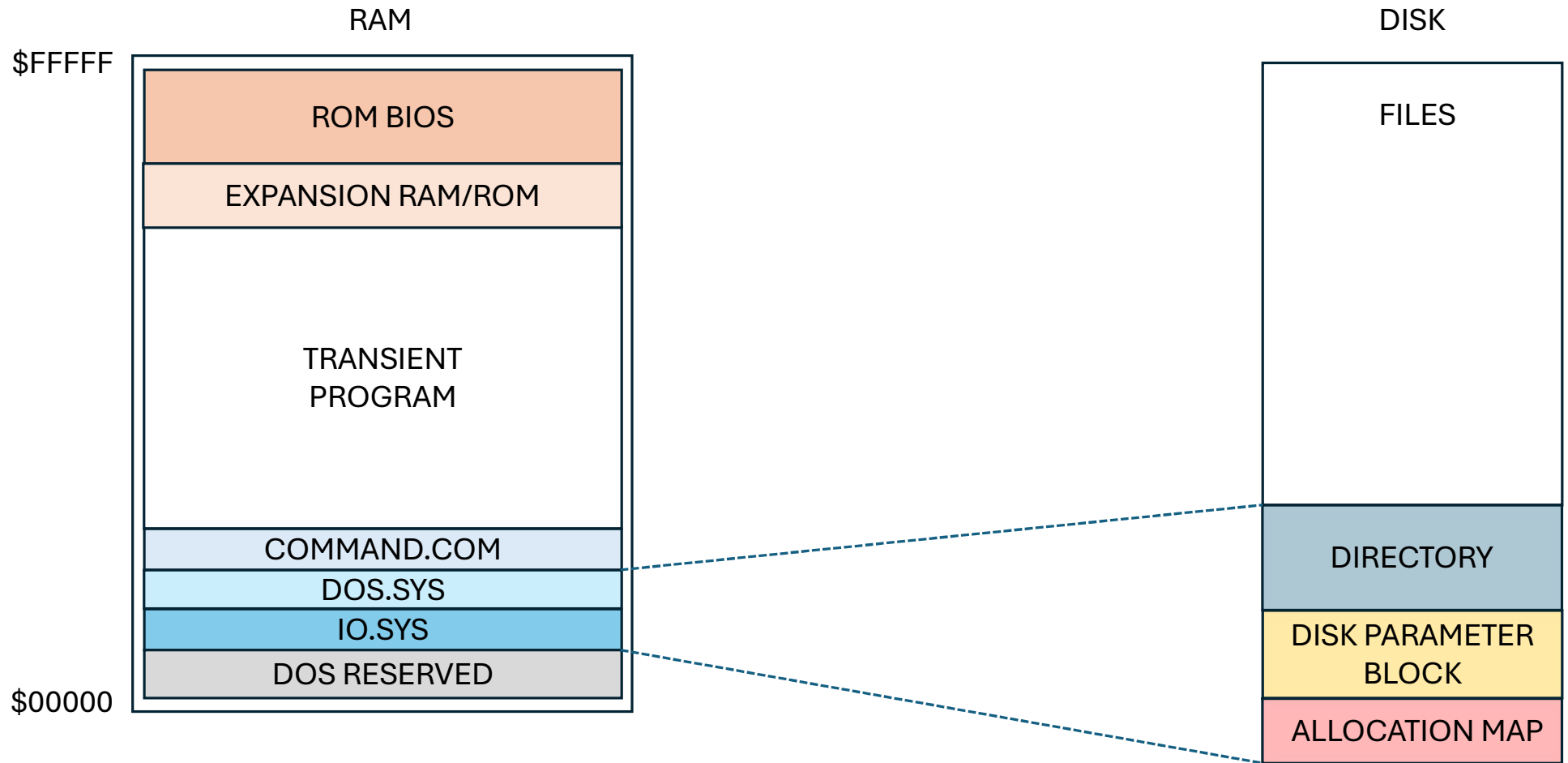
8192 Block Size

8" Single Sided Floppy
77 Tracks, 26 Sectors
2 Reserved Tracks
128 Bytes/Sector
243 Blocks
4 bytes per disk

Hard Disk
256 Tracks, 256 Sectors
1 Reserved Track
512 Bytes/Sector
1020 Blocks
128 bytes per disk

Logical Drive to Logical Volume Demo

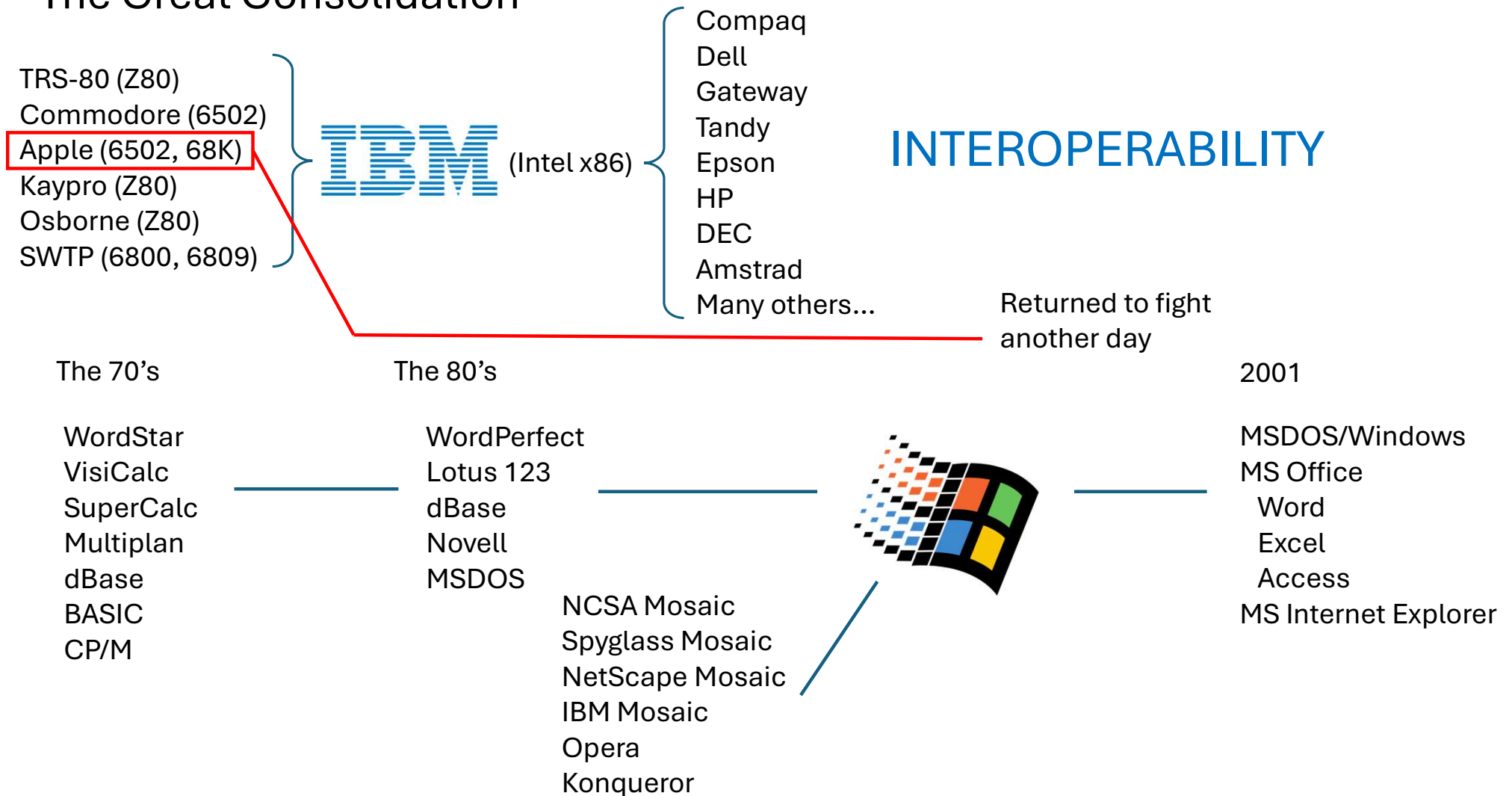
CP/M 2.2 File System vs DOS



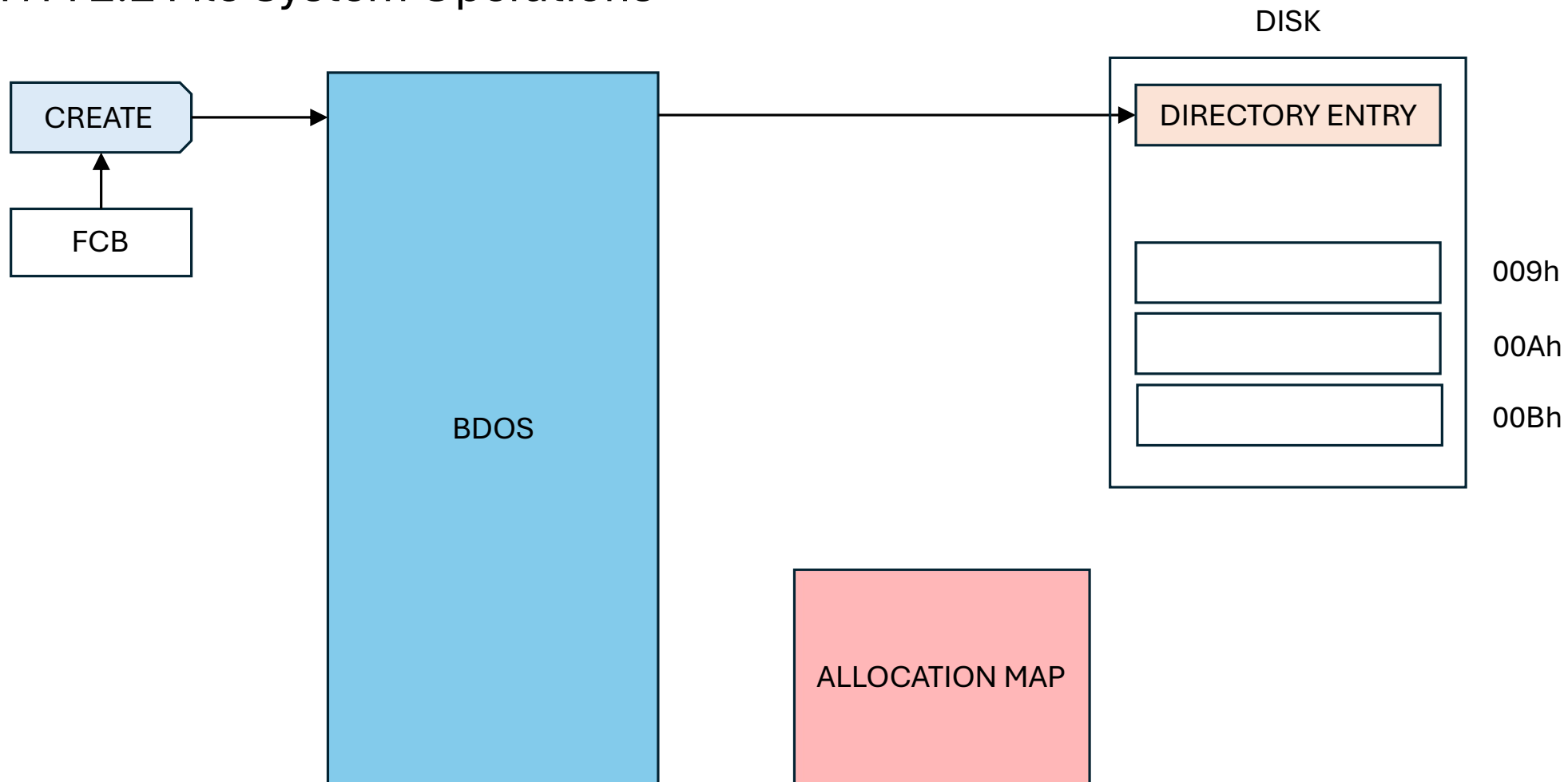
CP/M 2.2 File System vs DOS

- Easy migration from 8-Bit 8080/Z80 with CP/M to 16-Bit 8086 with DOS
 - Claimed compatibility with CP/M OS services
- Fully device independent I/O
 - No obsolete IOBYTE or devices like PUN
 - No vendor-specific CBIOS routines
- Advanced error reporting and recovery
- No “practical” limit on file and disk size
 - This claim didn’t last long
- Exact reporting of file size
- Entirely written in 8086 assembly
 - As opposed to PL/M or CP/M 80 translated to CP/M 86
- Fast efficient file structure
 - DOS FAT vs CP/M Extents and Allocation Vector
- No need to log in disks
- No overhead for non-128 byte sectors
 - Does its own deblocking
- File date/time stamping

The Great Consolidation

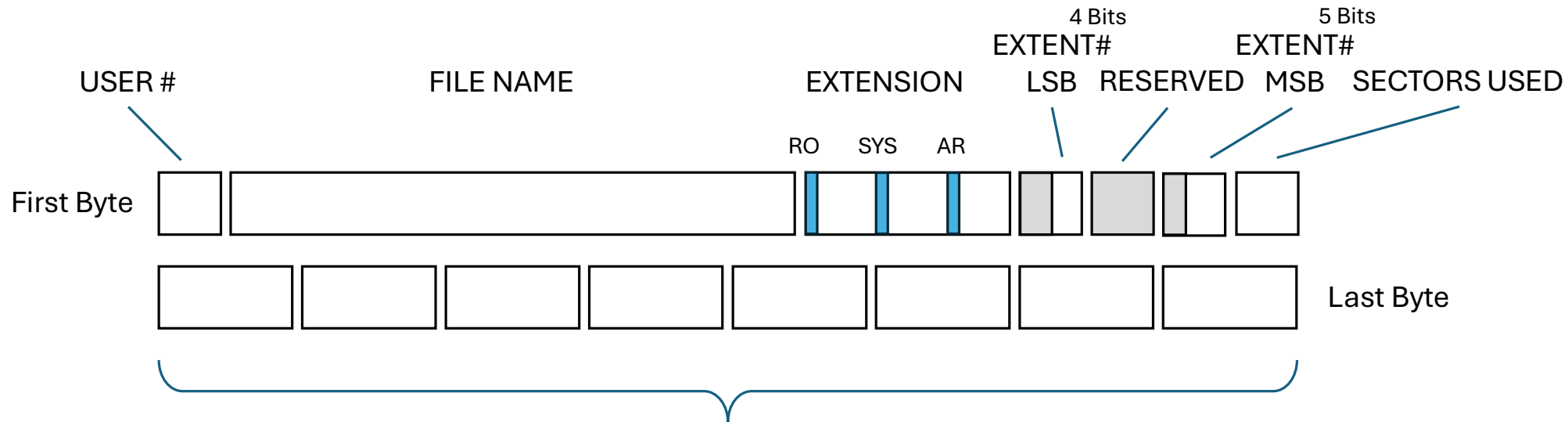


CP/M 2.2 File System Operations



CP/M 2.2 File System Operations

A directory entry (aka “extent”)



Up to 8 Allocation Blocks that can be assigned to the file

Total Blocks > 256

Maximum Sectors/Bytes per Extent (EXM=1)

BS=1024 64/8192 bytes

BS=2048 128/16384 bytes

BS=4096 256/32768 bytes Sector count rolls over once

BS=8192 256/32768 bytes Sector count rolls over once, no more than 4 blocks per extent

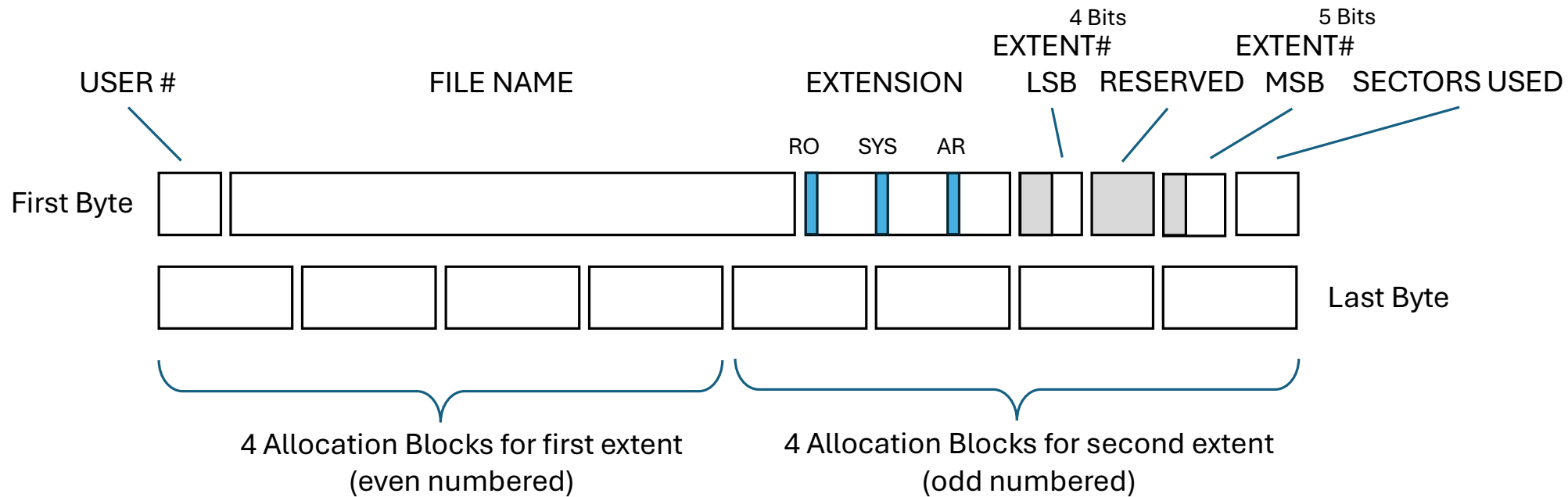
BS=16384 256/32768 bytes Sector count rolls over once, no more than 2 blocks per extent

Maximum Extents: 512

Maximum File Size: 8 Mbytes

CP/M 2.2 File System Operations

A directory entry (aka “extent”)



Maximum Sectors/Bytes per Extent (EXM=1)

BS=1024 64/8192 bytes

BS=2048 128/16384 bytes

BS=4096 256/32768 bytes Two logical extents per physical extent

BS=8192 256/32768 bytes Two logical extents per physical extent, no more than 4 blocks per extent

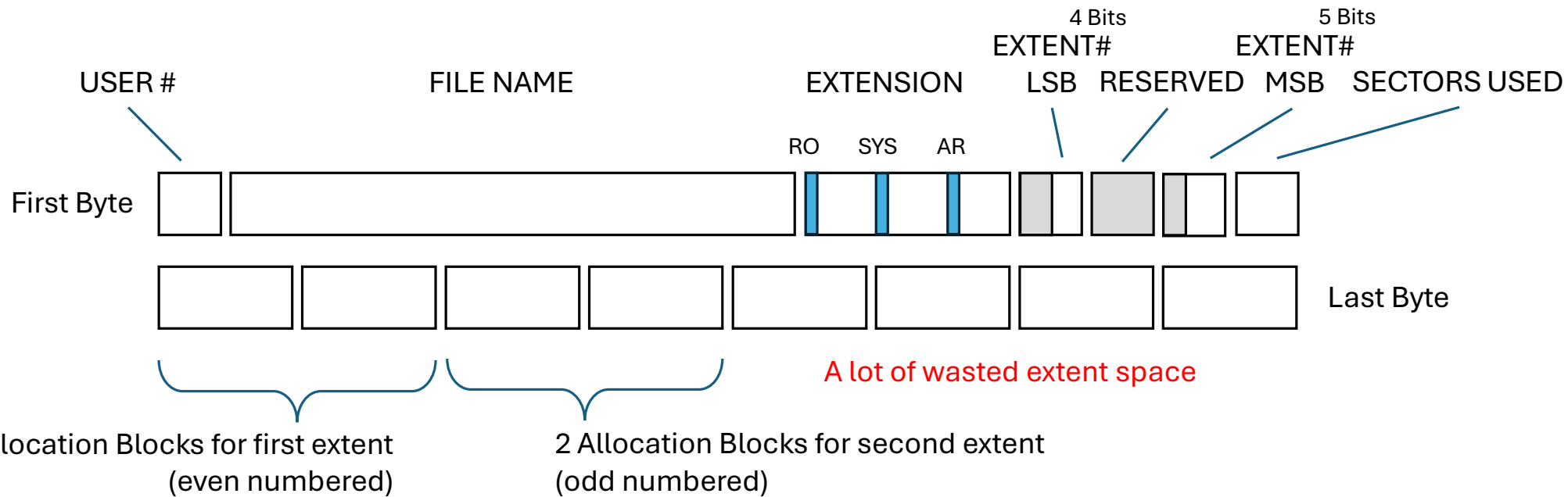
BS=16384 256/32768 bytes Two logical extents per physical extent, no more than 2 blocks per extent

Maximum Extents: 512

Maximum File Size: 8 Mbytes

CP/M 2.2 File System Operations

A directory entry (aka “extent”)



Maximum Sectors/Bytes per Extent (EXM=1)

BS=1024 64/8192 bytes

BS=2048 128/16384 bytes

BS=4096 256/32768 bytes Two logical extents per physical extent

BS=8192 256/32768 bytes Two logical extents per physical extent, no more than 4 blocks per extent

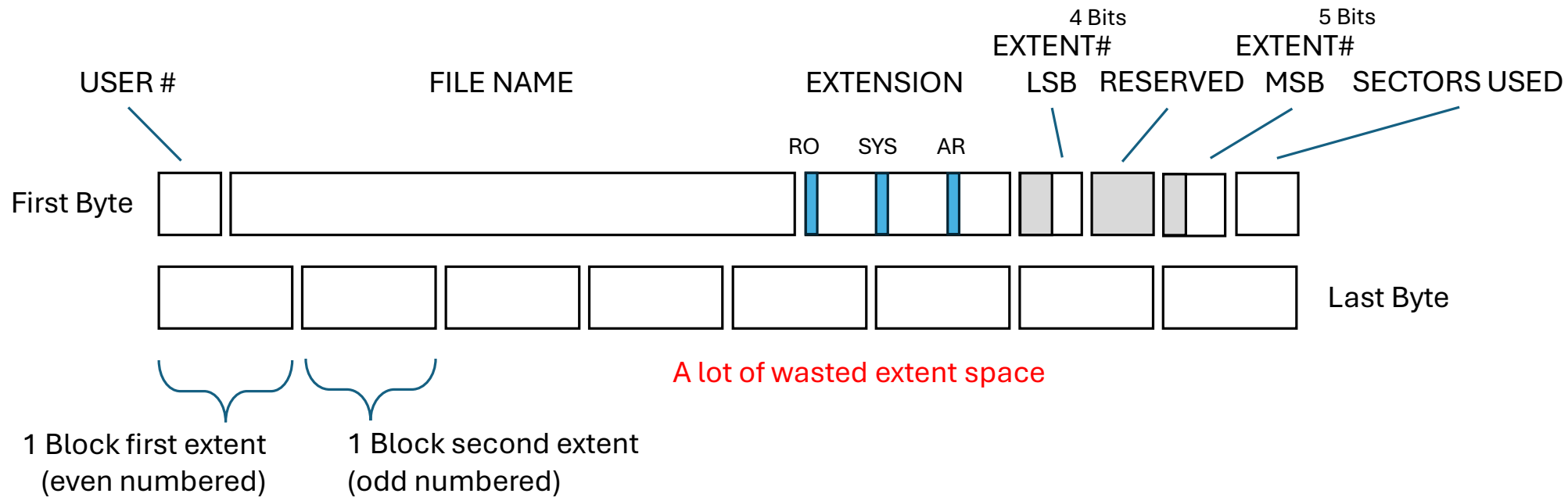
BS=16384 256/32768 bytes Two logical extents per physical extent, no more than 2 blocks per extent

Maximum Extents: 512

Maximum File Size: 8 Mbytes

CP/M 2.2 File System Operations

A directory entry (aka “extent”)



Maximum Sectors/Bytes per Extent (EXM=1)

BS=1024 64/8192 bytes

BS=2048 128/16384 bytes

BS=4096 256/32768 bytes Two logical extents per physical extent

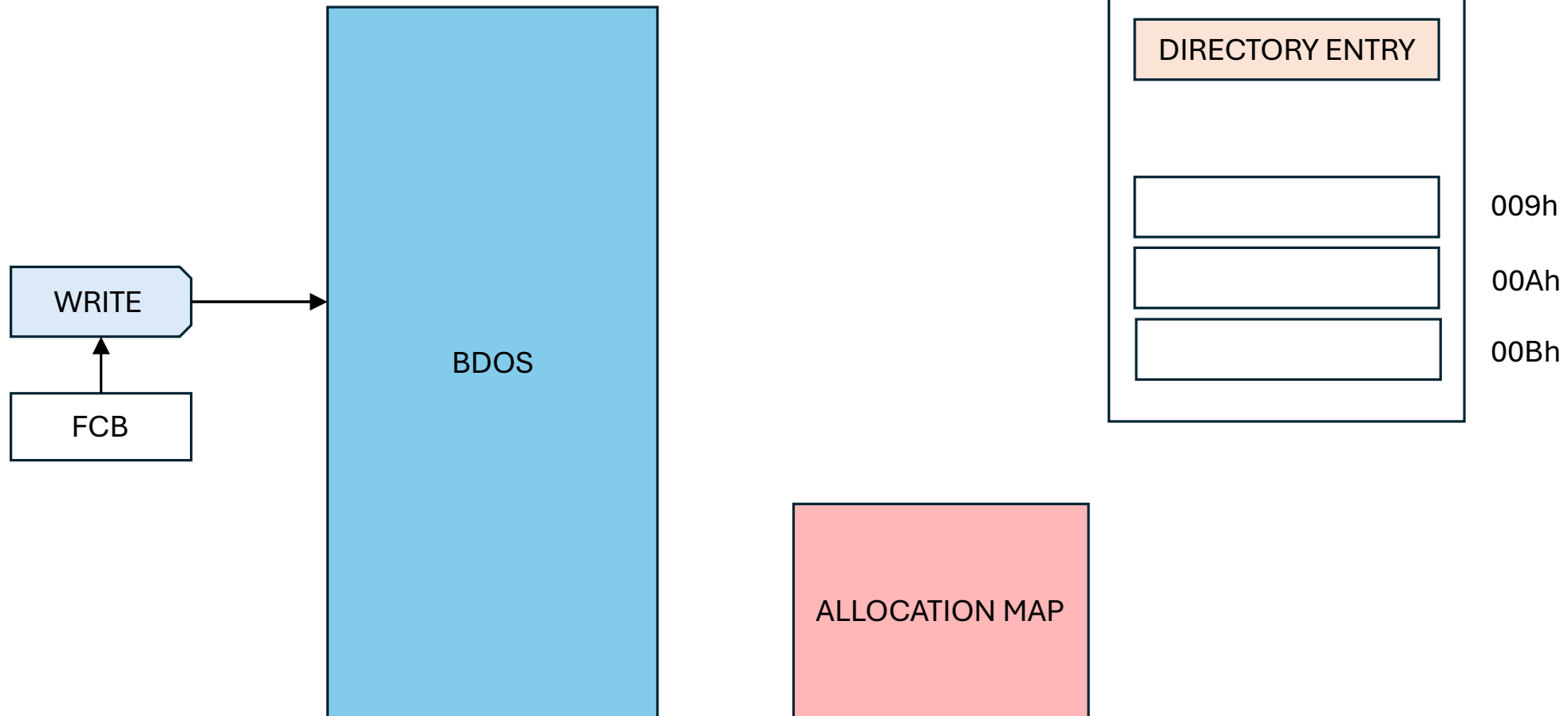
BS=8192 256/32768 bytes Two logical extents per physical extent, no more than 4 blocks per extent

BS=16384 256/32768 bytes Two logical extents per physical extent, no more than 2 blocks per extent

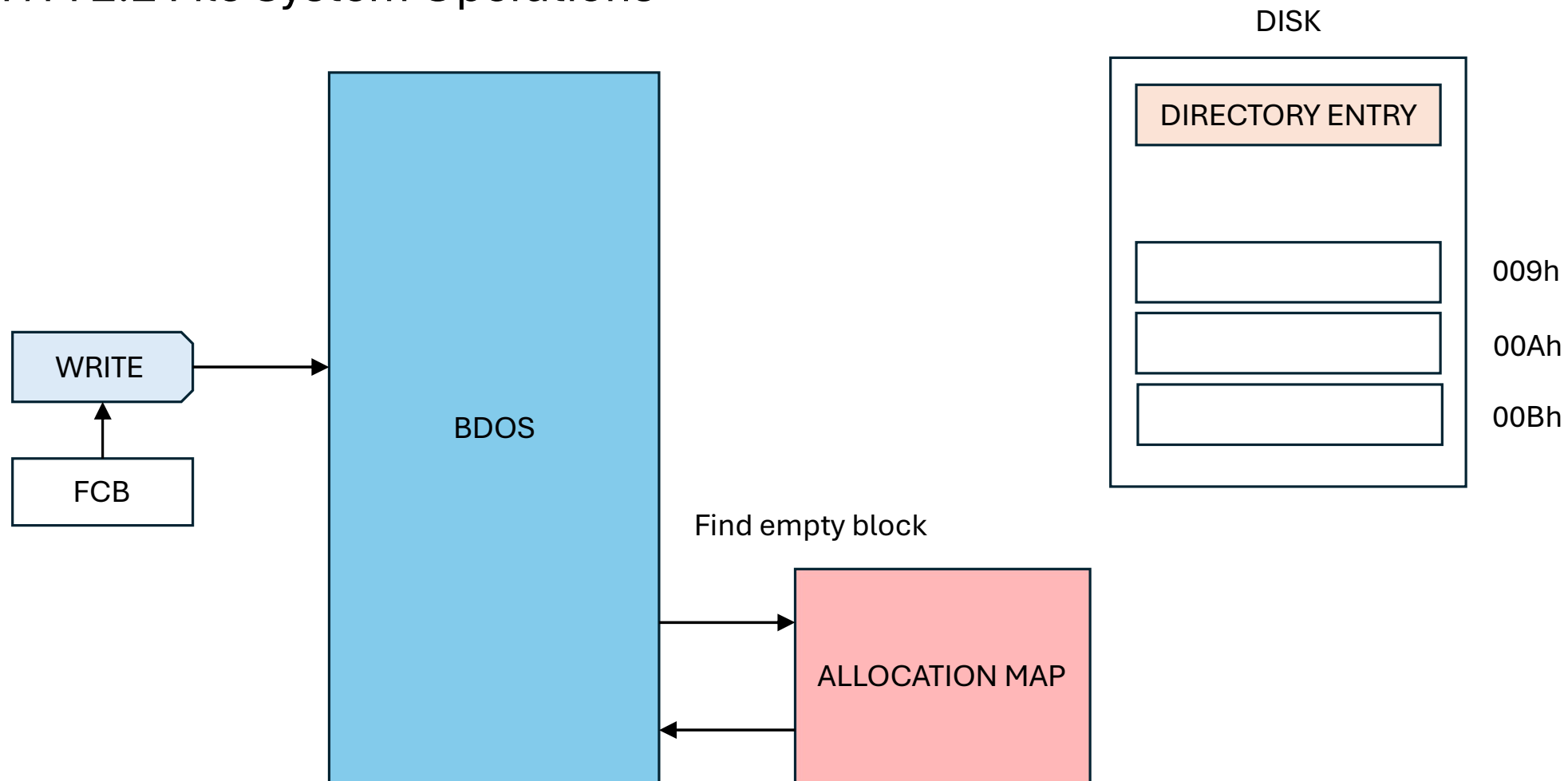
Maximum Extents: 512

Maximum File Size: 8 Mbytes

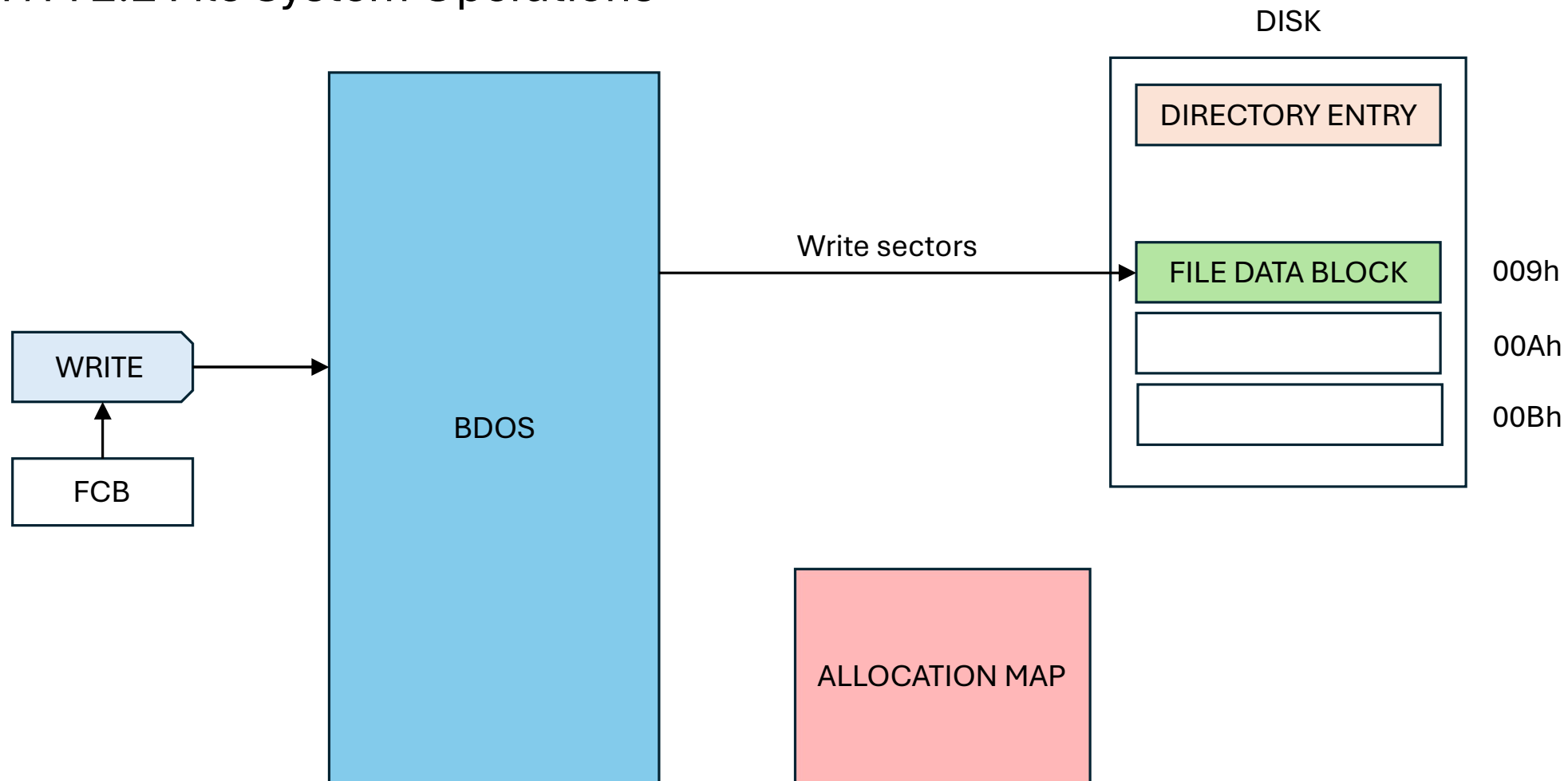
CP/M 2.2 File System Operations



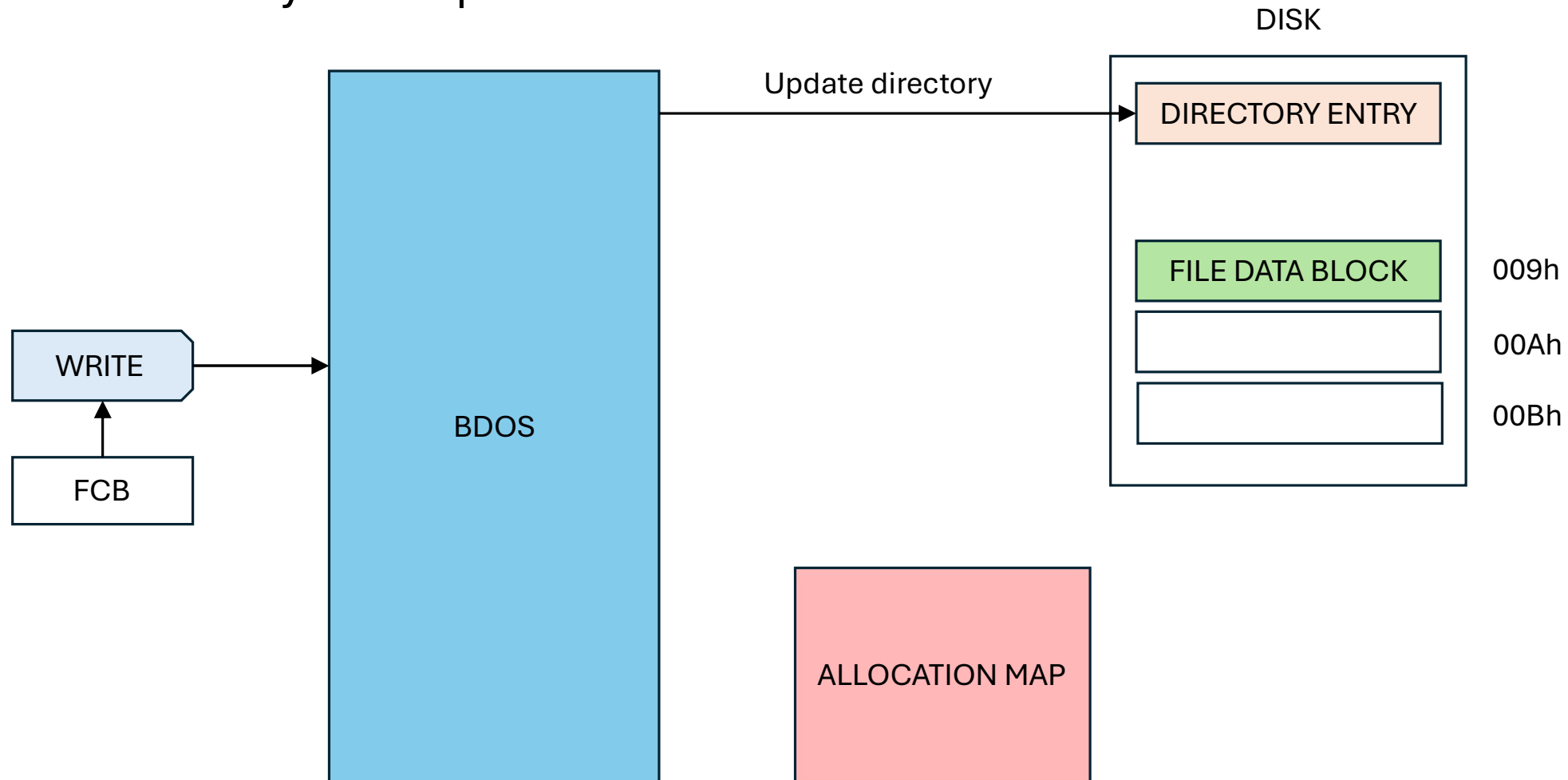
CP/M 2.2 File System Operations



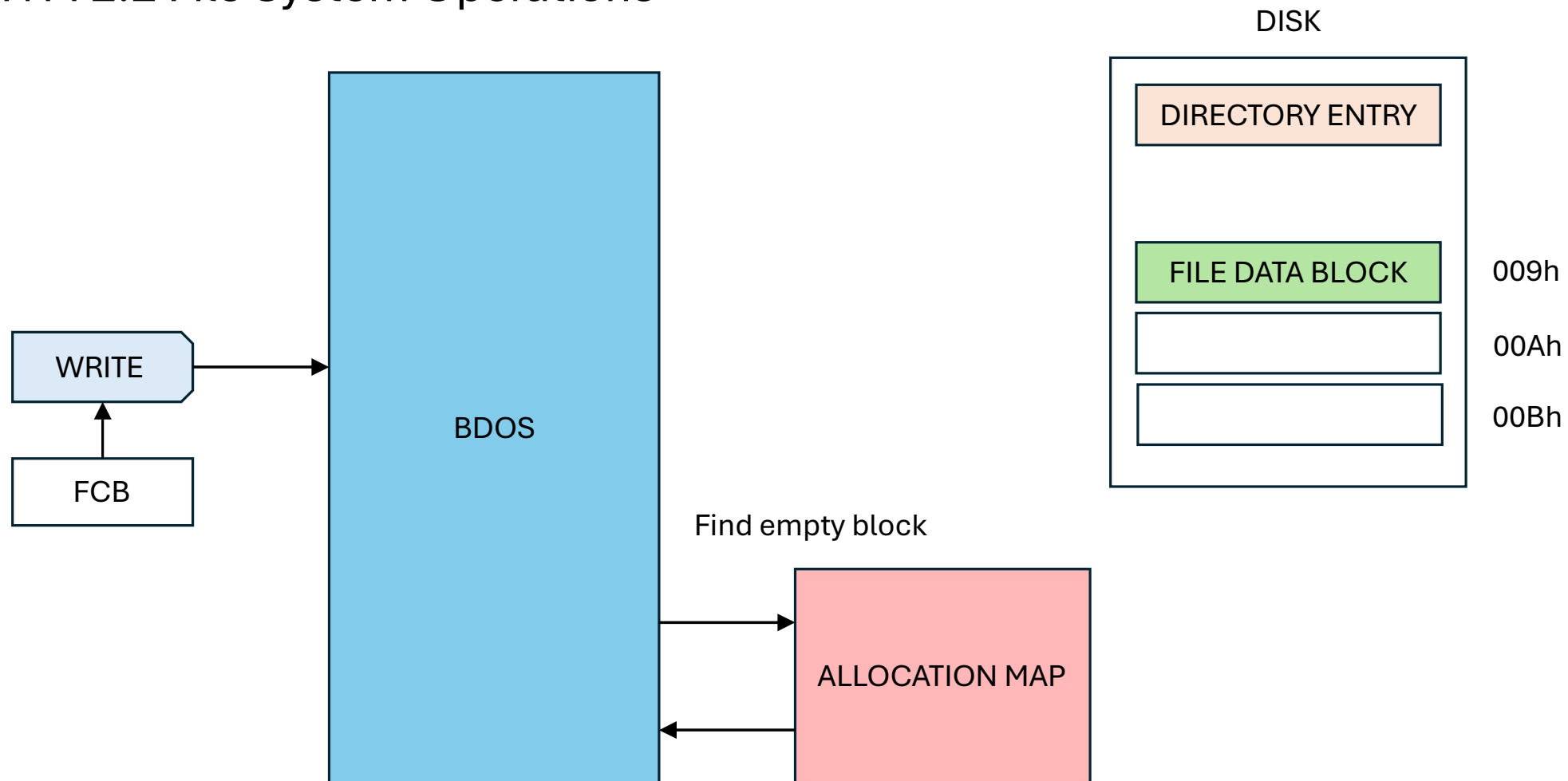
CP/M 2.2 File System Operations



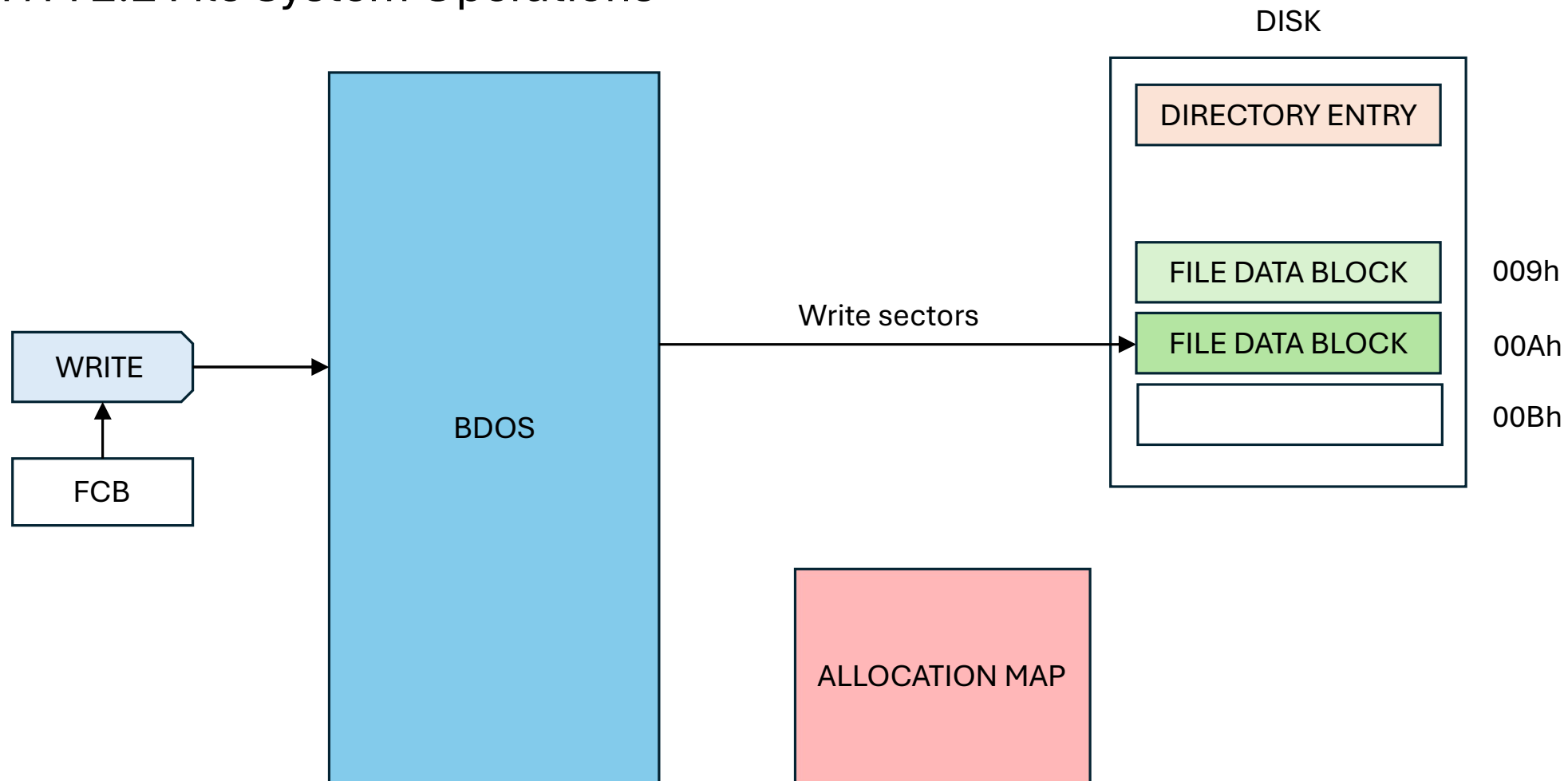
CP/M 2.2 File System Operations



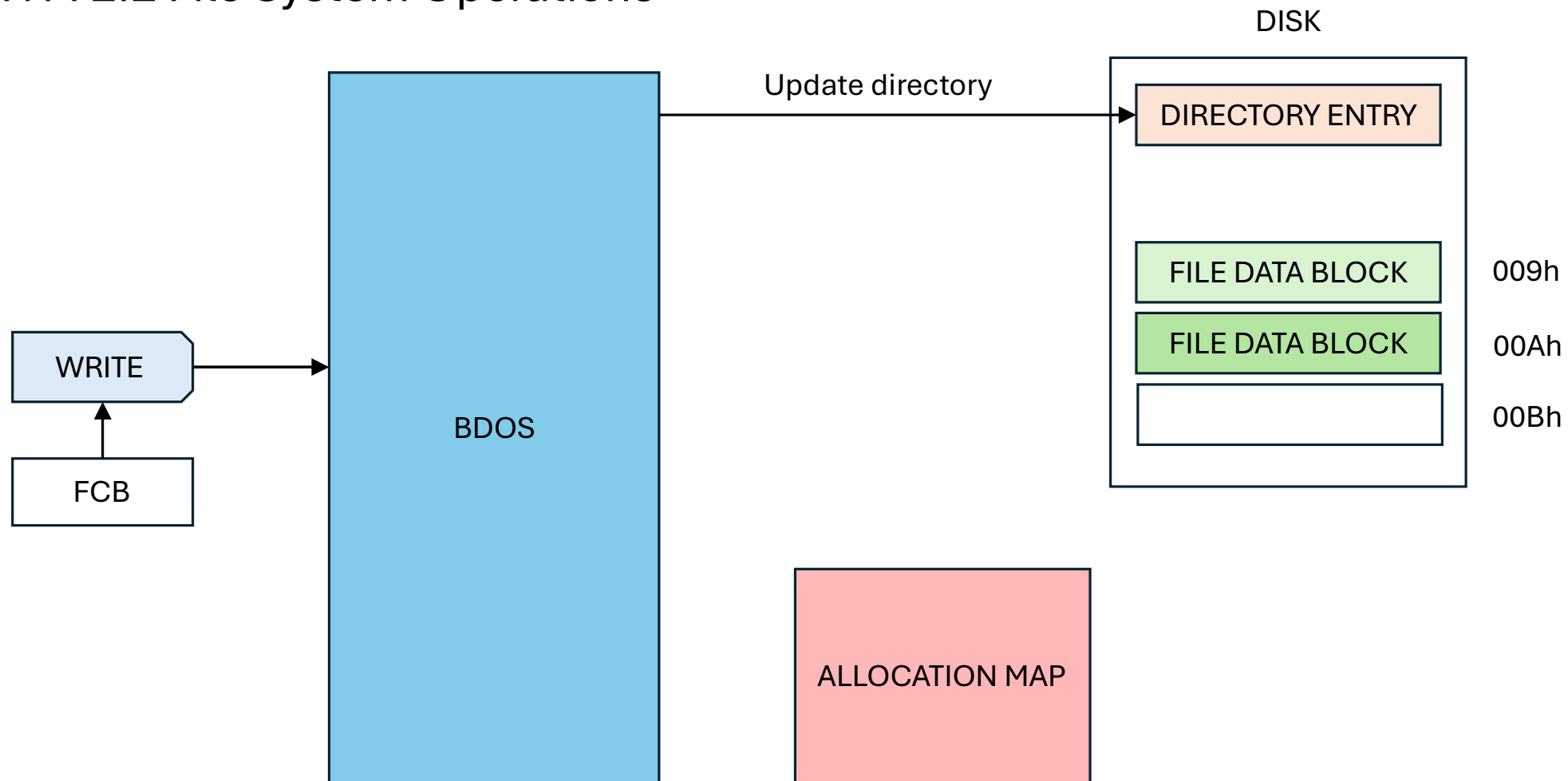
CP/M 2.2 File System Operations



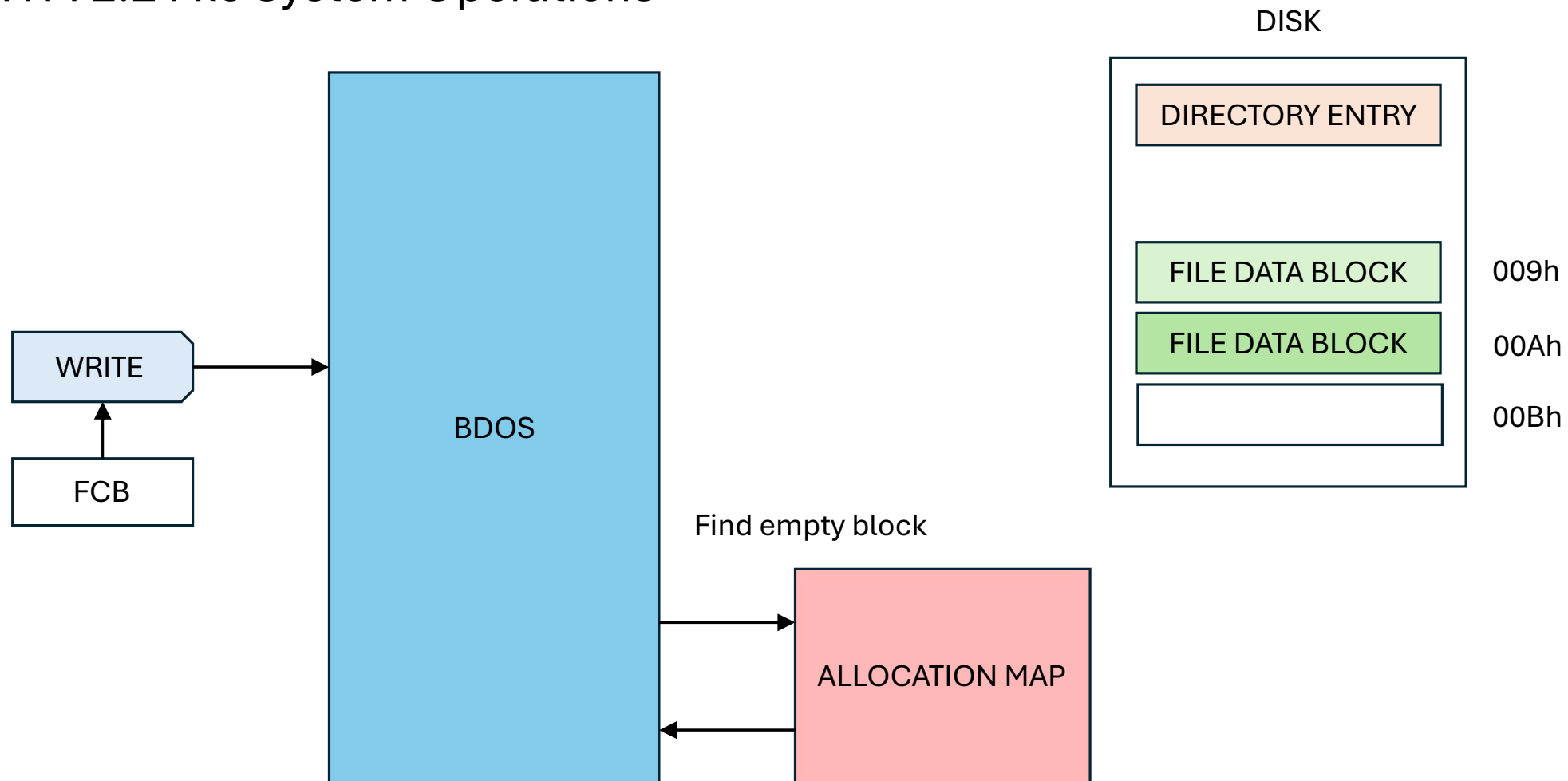
CP/M 2.2 File System Operations



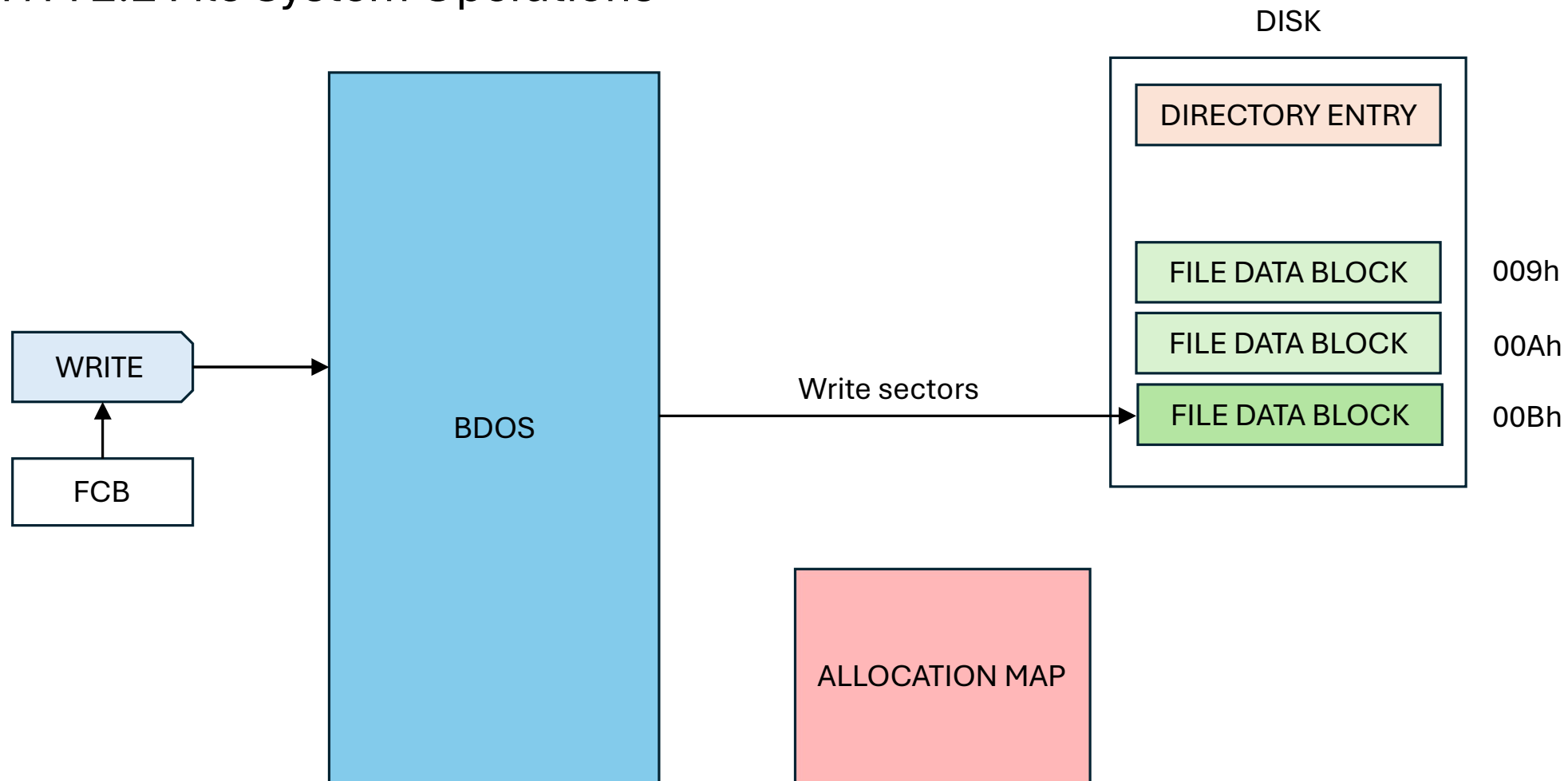
CP/M 2.2 File System Operations



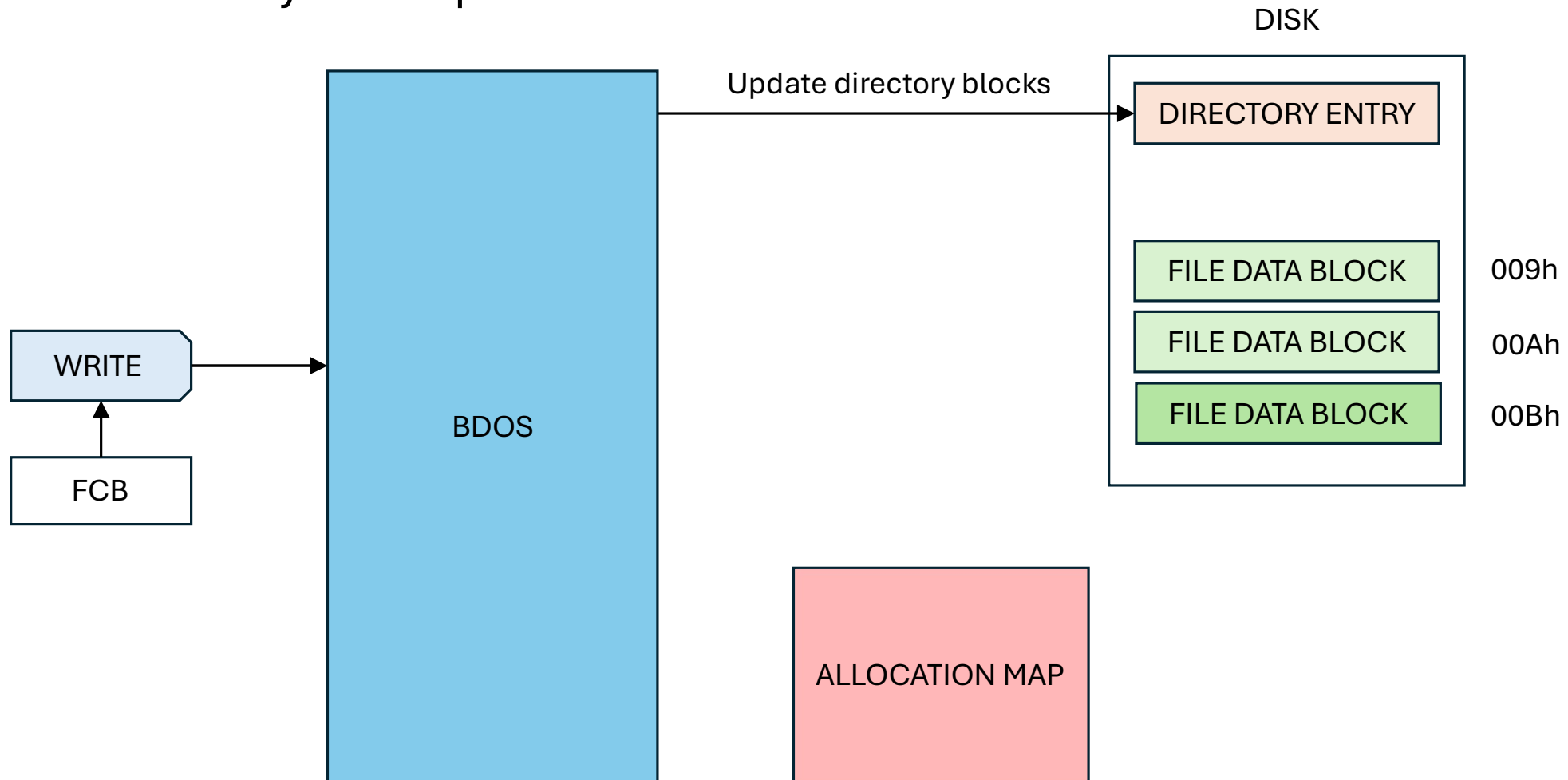
CP/M 2.2 File System Operations



CP/M 2.2 File System Operations

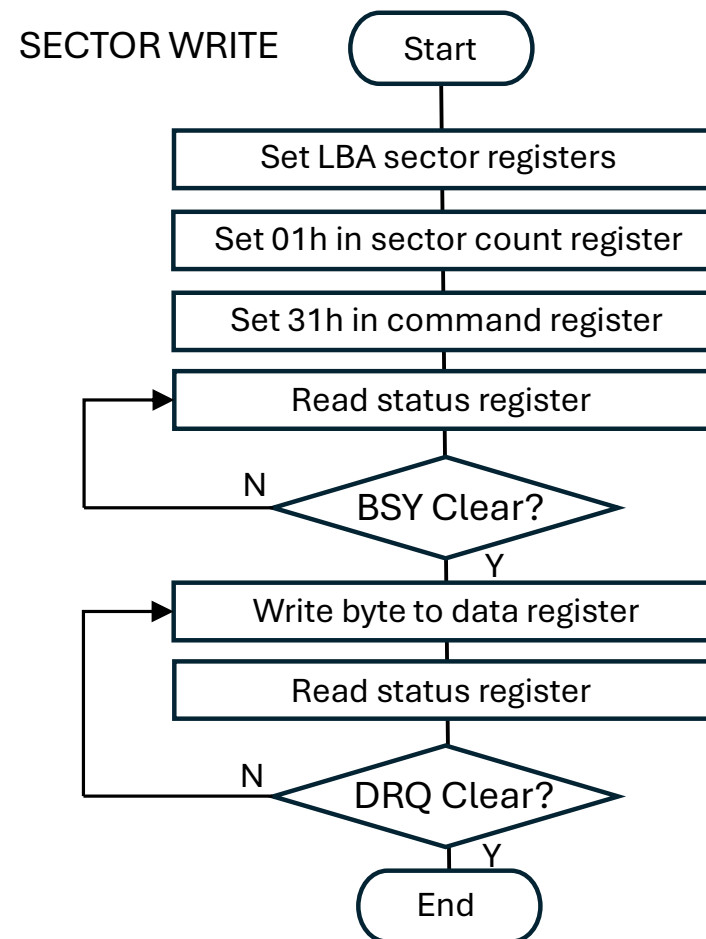
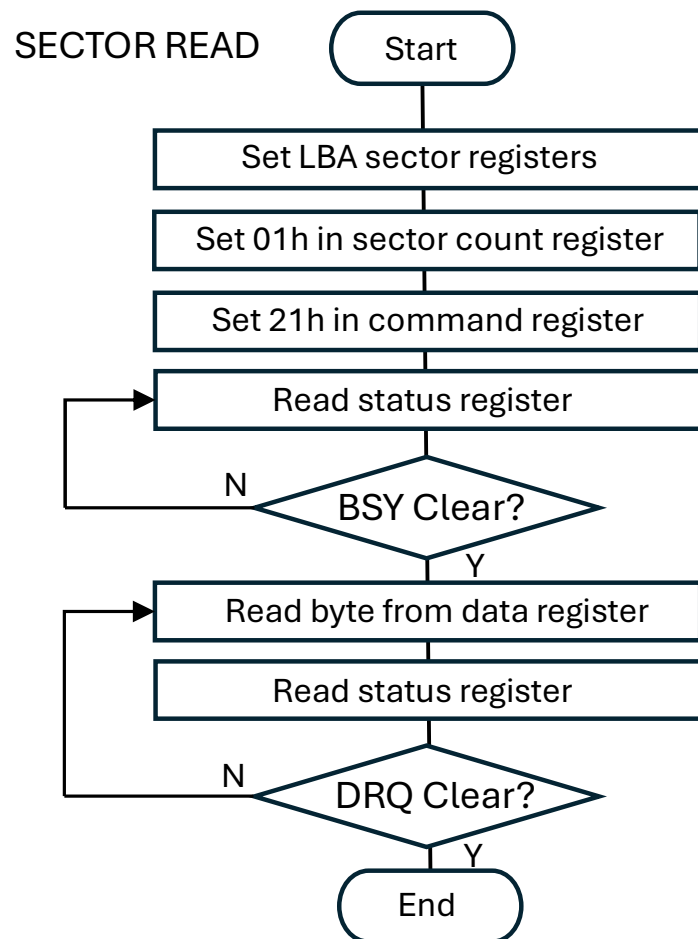


CP/M 2.2 File System Operations

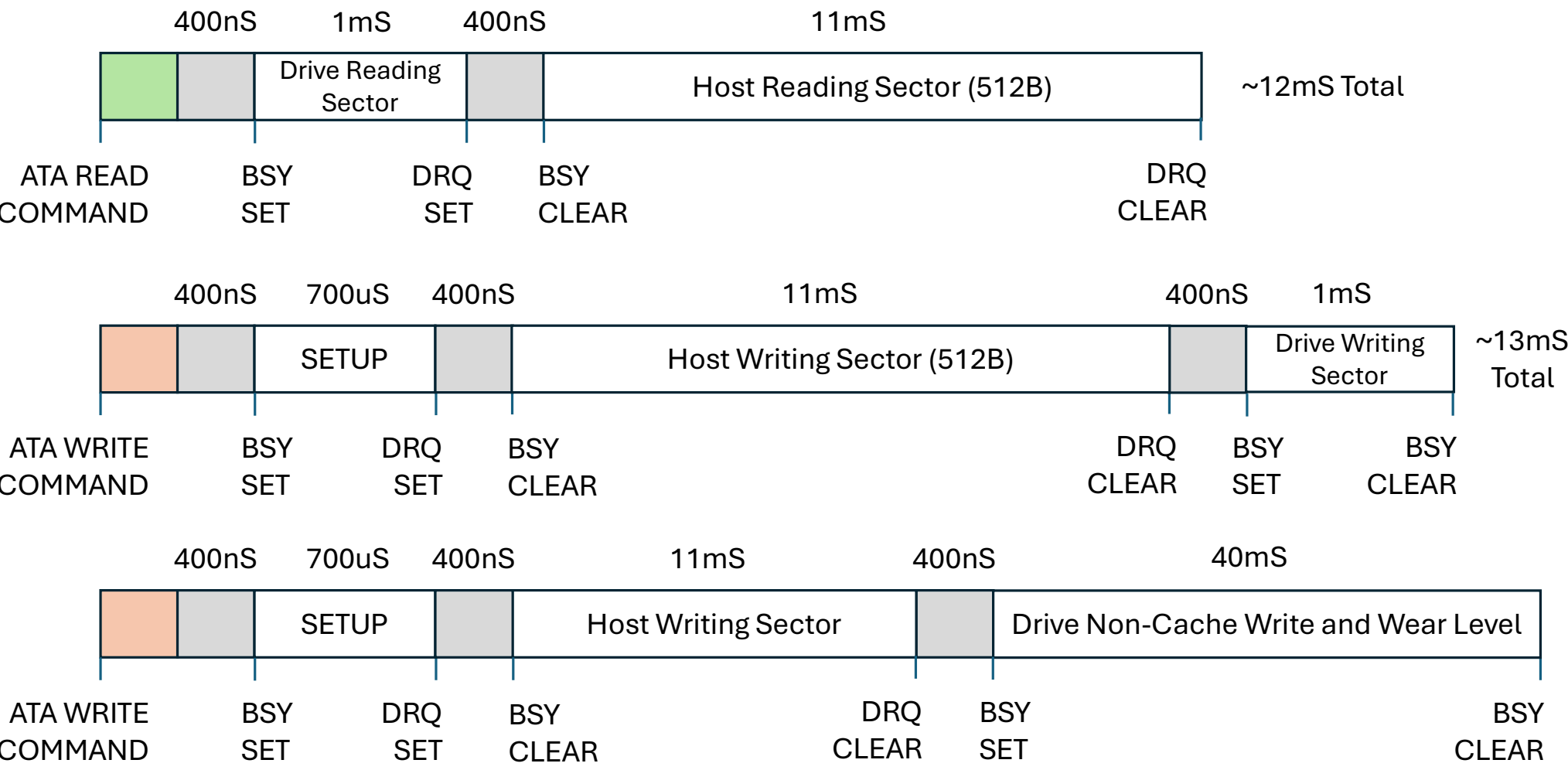


File operations demonstration

ATA Disk I/O Operations

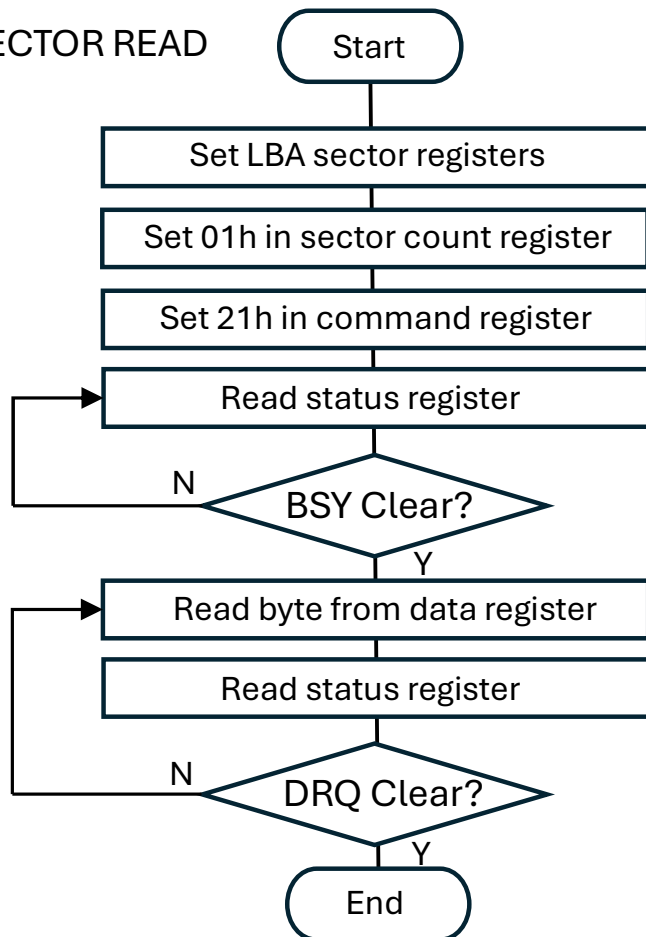


ATA Disk I/O Operations



ATA Disk I/O Operations

SECTOR READ



SECTOR WRITE

