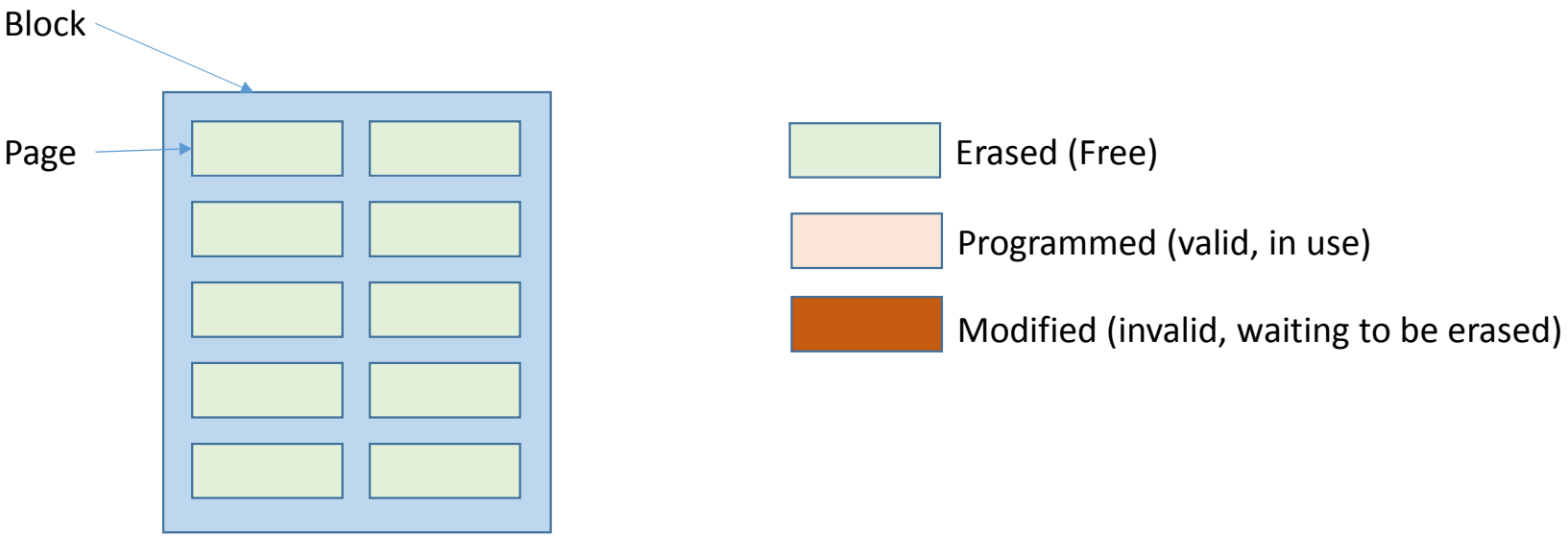
Flash Memory

- Increasingly used as a substitute for hard drive (SSD vs HDD)
- Physically similar to RAM, but with important differences
- In RAM, reads/writes can apply to any individual byte
- In flash, there are only two operations: program and erase
- Unit of program is the page
- Unit of erase is the block
- Each block has a finite lifetime of PE (program-erase cycles)
- Somewhere (firmware or OS) PE cycles for each block should be maintained
- Some effort to distribute PE cycles across the whole device is recommended
- This process is called wear leveling

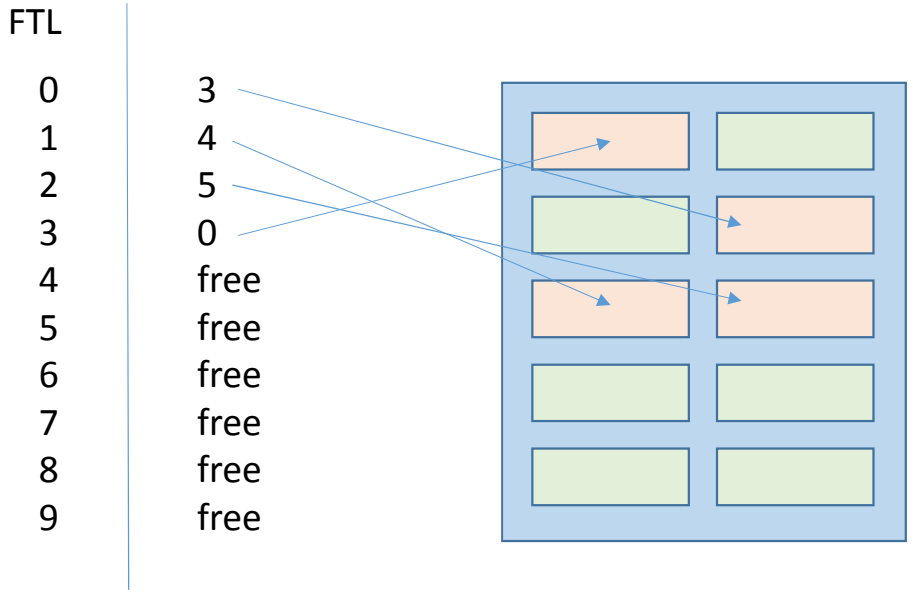


Facts about using pages and blocks

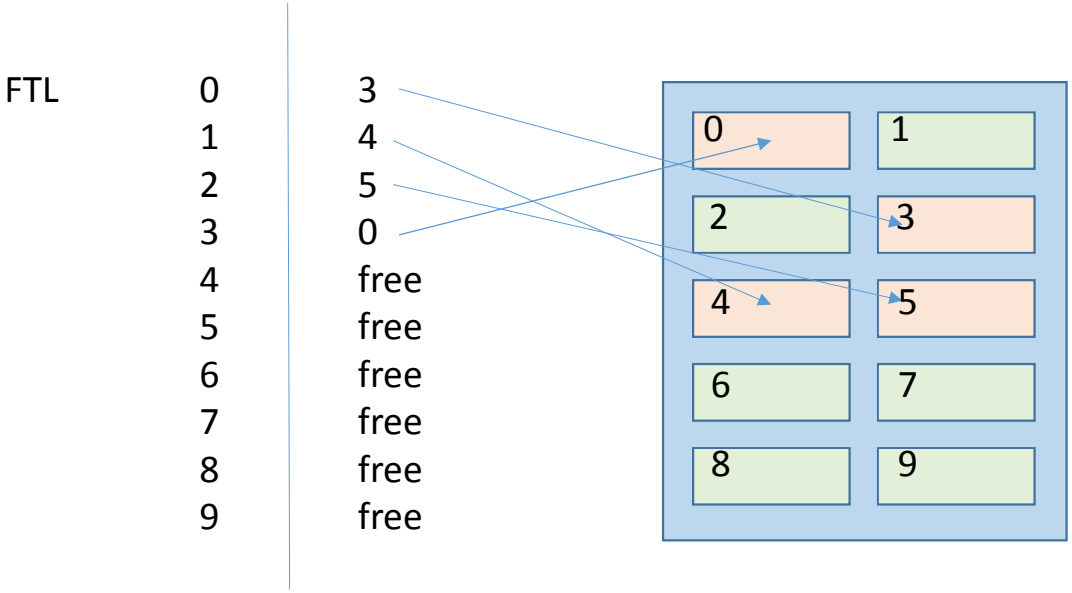
-- **Free (erased) pages** can be **programmed (written)**
-- Modifying a programmed page means programming a different free page and marking the original as **modified (invalid)**
-- Modified pages must be **erased** before being **programmed again**
-- Individual pages cannot be erased, **only complete blocks can be erased**

Example Flash Page Mapping

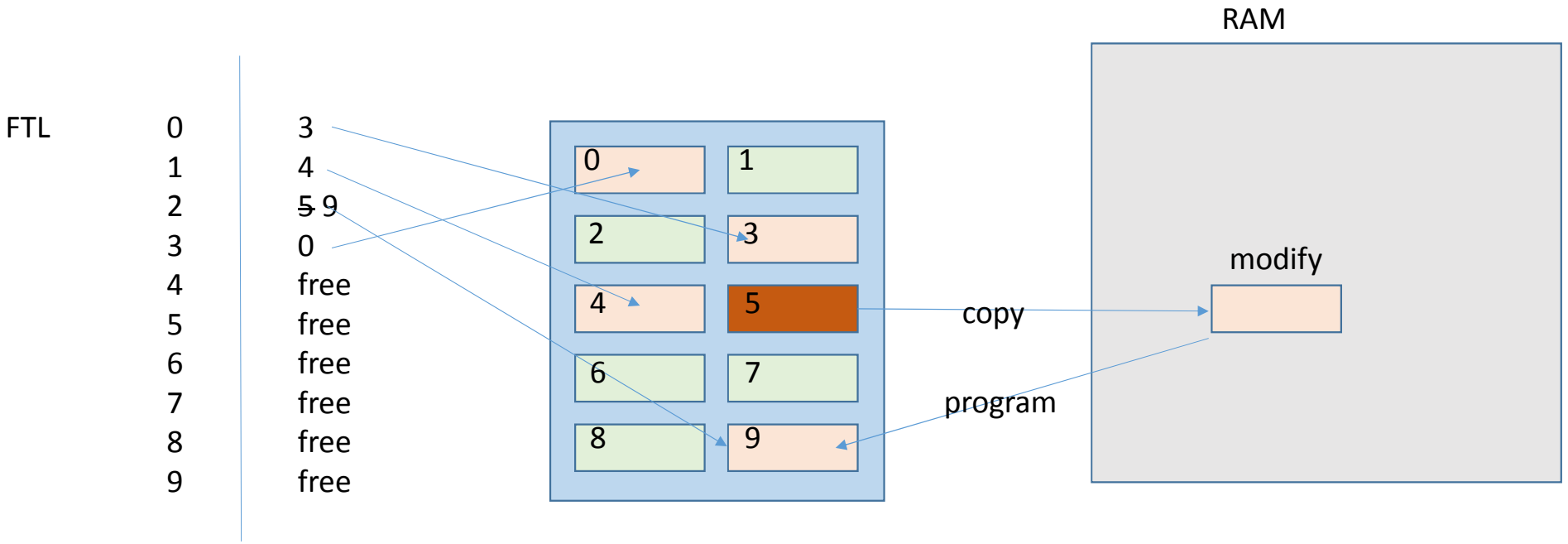
FTL (Flash Translation Layer) maps logical pages to physical pages
Since updates require pages to be relocated, FTL makes it easy to remap



Example: Modify logical page 2, currently stored in physical page 5



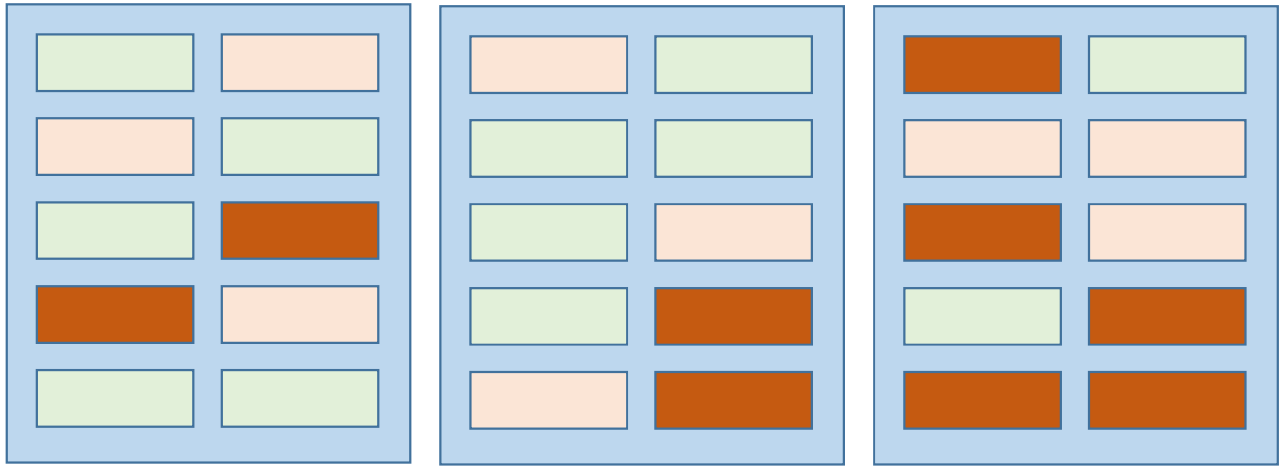
- Logical page 2 = physical page 5, copied to RAM
- Content modified in RAM
- Modified physical page 5 newly programmed to free page 9
- Physical page 5 marked invalid, can't be reused until entire block is erased
- FTL updated so logical page 2 now points to physical page 9 not 5

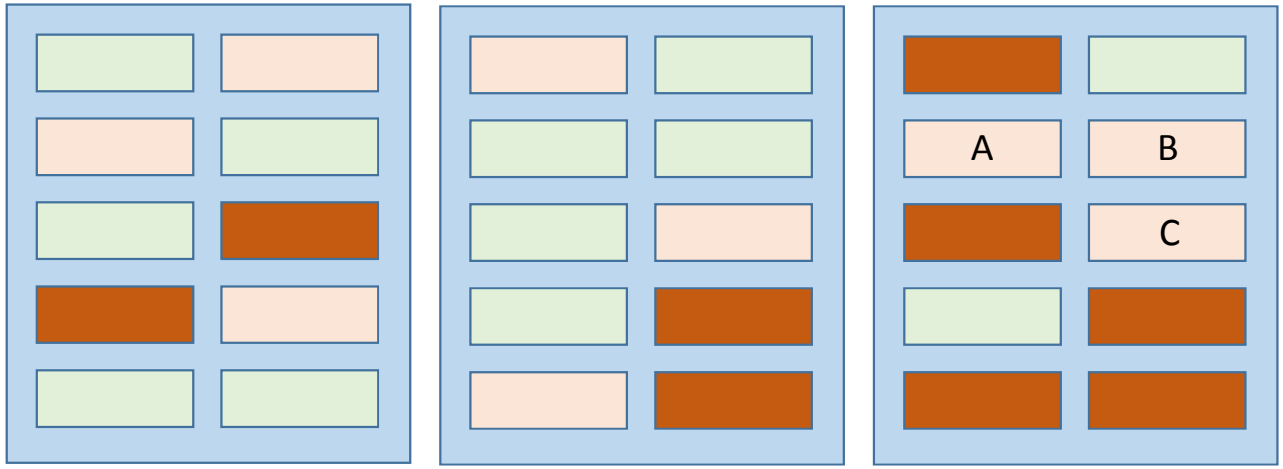


Garbage collection

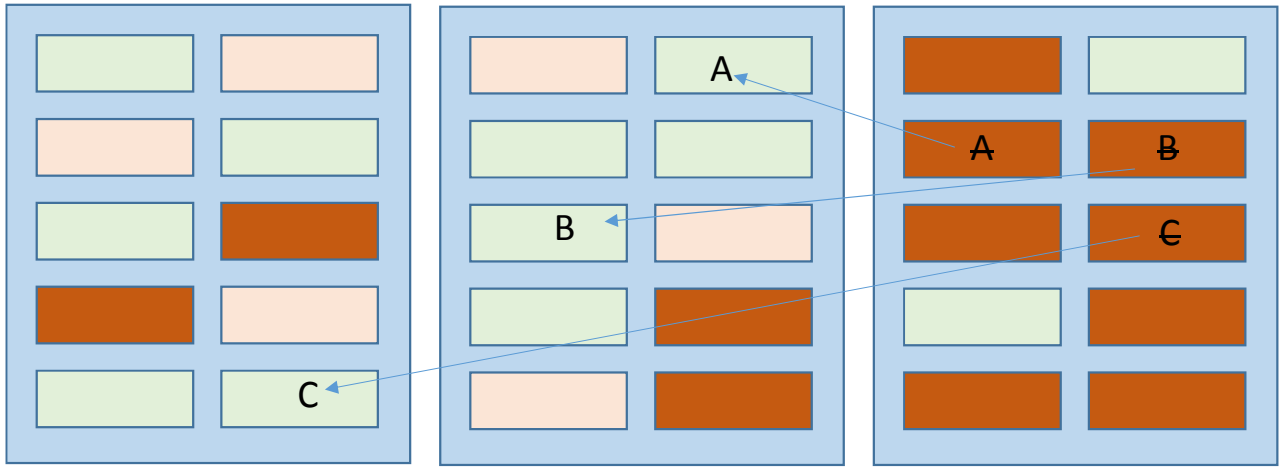
If invalid pages accumulate, memory gradually becomes unusable
Modified pages can't be reused (reprogrammed) until the entire block is erased

Garbage collection is occasionally executed to:
-- find a block with a lot of invalid pages
-- move any remaining valid pages in that block to a different block (and update FTL mapping)
-- erase the entire block with the invalid pages
-- try to move valid pages to a block with a low erase count
-- increment the erase count for the block

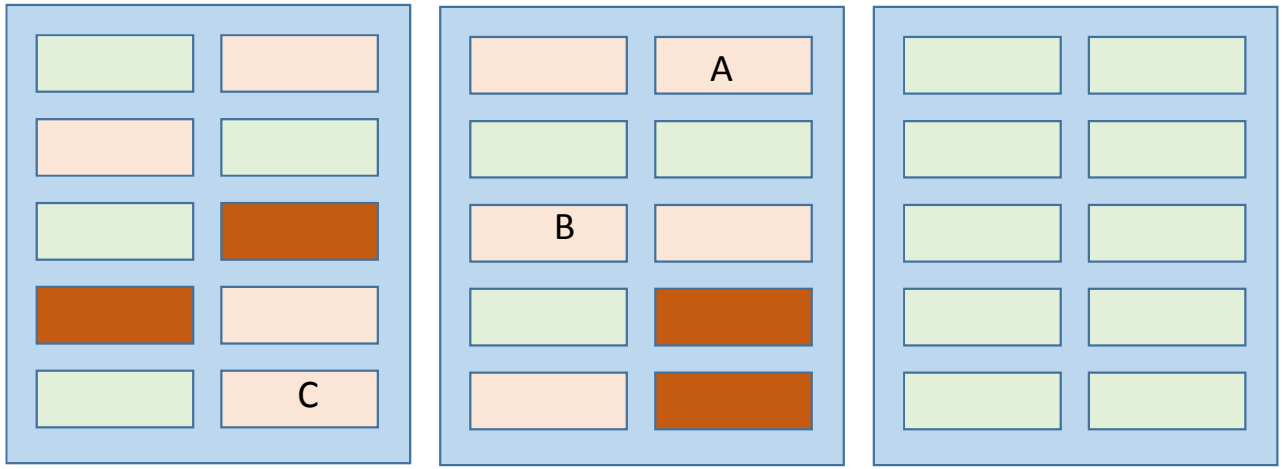




3rd block is almost ready to be erased
But blocks A, B, C are valid
Move A, B, C to free pages in a different block
Remap STL entries to new locations for A, B, C



3rd block can now be erased
Also want to minimize erasing a block with too many free pages, this wastes PE cycle on a page that doesn't need to be erased again.



Erase Count: 0

Erase Count: 0

Erase Count: ~~0~~ 1

Flash Drives (USB, SSD)

- Uses flash memory at the physical layer
- Provides integrated controller that manages reading/writing blocks/pages
- So the device can be formatted as FAT32, NTFS, ext4, or other.
- But the file system format still communicates with the flash controller
- What about flash-specific file systems like YAFFS?
- These are low-level file systems that duplicate the job of the flash controller
- If your device already has a flash controller, formatting to YAFFS will interfere with the existing controller
- YAFFS is designed for a raw flash device without a controller.