CLAIMS

1. A method comprising: scanning pages, beginning at a first selected logical page of a memory array, to find a first free page, the first free page being determined by making a determination whether a page contains data; based on a determination the page contains data, determining whether a commit marker is present in a header in a specifically allocated portion on the page; based on a determination that a commit marker is present in the header on the page, scanning a subsequent page; and based on a determination that a commit marker is not present in the header on the page, designating a page subsequent to the page as the first free page; and based on a determination that the page does not contain data; designating a page subsequent to the page as the first free page; marking the first free page as available; and writing to the page marked as available with the next write cycle.

2. The method of claim 1, further comprising: marking a current scanned page prior to the first free page as invalid; and erasing the current scanned page marked as invalid.

3. The method of claim 1, wherein scanning pages comprises scanning pages in a sequential order.

4. The method of claim 1, wherein determining whether a commit marker is present in a header on the page comprises reading the header in a spare area of the memory array that is associated with the page.

5. The method of claim 2, wherein marking as invalid comprises writing a marker into a header associated with the page.

6. The method of claim 2, wherein marking as invalid comprises writing a marker into a header associated with a block of pages containing the first free page.

7. The method of claim 1, wherein the memory array is divided into blocks, each block containing multiple pages, wherein scanning pages comprises scanning pages of a block, the method further comprising: marking a current scan page before the first free page in each block as invalid; and marking the first free page as available in each block.

8. A power loss recovery system for a memory array, the system comprising: a page scanner to scan pages of a memory array to find a first free page, the first free page being determined by making a determination whether a page contains data: based on a

determination the page contains data, the page scanner to determine whether a commit marker is present in a header in a specifically allocated portion on the page; based on a determination that a commit marker is present in the header on the page, the page scanner to scan a subsequent page; and based on a determination that a commit marker is not present in the header on the page, the page scanner to designate a page subsequent to the page as the first free page; and based on a determination that the page does not contain data; the page scanner to designate a page subsequent to the page as the first free page; a page marker to mark the first free page as available; and program circuitry to write to the page marked as available with the next write cycle.

9. The system of claim 8, wherein, the page marker is further to mark a current scanned page prior to the first free page as invalid, and wherein the program circuitry is further to erase the current scan page marked as invalid.

10. A memory device comprising: a memory cell array comprising a plurality of memory cells, the cells being grouped into a plurality of blocks; and a memory controller coupled to the memory cell array to scan pages of the memory cell array to find a first free page, the first free page being determined by making a determination whether a page contains data; based on a determination the page contains data, the memory controller to determine whether a commit marker is present in a header in a specifically allocated portion on the page; based on a determination that a commit marker is present in the header on the page, the memory controller to scan a subsequent page; and based on a determination that a commit marker is not present in the header on the page, the memory controller to designate a page subsequent to the page as the first free page; and based on a determination that the page does not contain data; the memory controller to designate a page subsequent to the page as the first free page; the memory controller further to mark the first free page as available and to write to the page marked as available.

11. The device of claim 10, wherein the memory controller scans pages in a sequential order.

12. The device of claim 10, wherein the memory cell array is a flash memory cell array.

13. The device of claim 10, wherein the memory cell array is a phase change memory cell array.

14. A tangible computer-readable storage medium not comprising transitory signals

containing instructions that, when executed by one or more processors, cause the one or more processors to perform operations comprising: scanning pages, beginning at a first selected logical page of a memory array, to find a first free page, the first free page being determined by making a determination whether a page contains data; based on a determination the page contains data, determining whether a commit marker is present in a header in a specifically allocated portion on the page; based on a determination that a commit marker is present in the header on the page, scanning a subsequent page; and based on a determination that a commit marker is not present in the header on the page, designating a page subsequent to the page as the first free page; and based on a determination that the page does not contain data; designating a page subsequent to the page as the first free page; marking the first free page as available; and writing to the page marked as available with the next write cycle.