



**FIGURE e5.11.2 Small write update on RAID 4.** This optimization for small writes reduces the number of disk accesses as well as the number of disks occupied. This figure assumes we have four blocks of data and one block of parity. The naive RAID 4 parity calculation in the left of the figure reads blocks D1, D2, and D3 before adding block D0' to calculate the new parity P'. (In case you were wondering, the new data D0' comes directly from the CPU, so disks are not involved in reading it.) The RAID 4 shortcut on the right reads the old value D0 and compares it to the new value D0' to see which bits will change. You then read the old parity P and then change the corresponding bits to form P'. The logical function exclusive OR does exactly what we want. This example replaces three disk reads (D1, D2, D3) and two disk writes (D0', P') involving all the disks for two disk reads (D0, P) and two disk writes (D0', P'), which involve just two disks. Increasing the size of the parity group increases the savings of the shortcut. RAID 5 uses the same shortcut.