```
monitor Disk_Scheduler {  ## Invariant DISK
  int position = -1, c = 0, n = 1;
  cond scan[2];   # scan[c] signaled when disk released
  procedure request(int cyl) {
    if (position == -1)  # disk is free, so return
      position = cyl;
    elseif (position != -1 && cyl > position)
      wait(scan[c],cyl);
    else
      wait(scan[n],cyl);
  }
  procedure release() {
    int temp;
    if (!empty(scan[c]))
      position = minrank(scan[c]);
    elseif (empty(scan[c]) && !empty(scan[n])) {
      temp = c; c = n; n = temp;       # swap c and n
      position = minrank(scan[c]);
      }
    else
      position = -1;
    signal(scan[c]);
  }
}
```

Figure 5.13 Separate disk scheduler monitor.