```
struct csr { /* Template for printer CSRs */
    int csr_power; /* Is printer powered on? */
    int csr_load; /* Load a sheet of paper */
    int csr_addr; /* Specify address of data to print */
    int csr_getdata; /* Upload data from memory */
    int csr_spray; /* Start inkjet spraying */
    int csr_advance; /* Advance paper to next band */
    int csr_dev_busy; /* Nonzero => device busy */
    int csr_levels; /* CMYK Ink levels in 4 bytes */
}
struct csr *p; /* Pointer to the device address area */

p = (struct csr *)0x110000; /* Set p to device address */
if (p->csr_power == 0); /* Test if printer is on */
    error("printer not on");
p->csr_load = 1; /* Start loading paper */
while (p->csr_dev_busy) /* Poll to wait for the load to complete */
    ;
p->csr_addr = &mydata /* Specify the location of data in memory */
p->csr_getdata = 1; /* Cause printer to pick up data */
while (p->csr_dev_busy) /* Poll to wait for printer to complete loading data */
    ;
p->csr_spray = 1; /* Start the inkjet spraying */
while (p->csr_dev_busy) /* Poll to wait for the inkjet to finish */
    ;
p->csr_ = 1; /* Advance the paper to the next band */
while (p->csr_dev_busy) /* Poll to wait for the paper advance to complete*/
    ;
```

Figure 16.4 The code from Figure 16.3 rewritten to use a C struct.