```
/*
 * Watchdog Driver Test Program
 */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/ioctl.h>
#include <linux/types.h>
#include <linux/watchdog.h>

int fd;

/*
 * This function simply sends an IOCTL to the driver, which in turn ticks
 * the PC Watchdog card to reset its internal timer so it doesn't trigger
 * a computer reset.
 */
static void keep_alive(void)
{
    int dummy;

    ioctl(fd, WDIOC_KEEPALIVE, &dummy);
}

/*
 * The main program.  Run the program with "-d" to disable the card,
 * or "-e" to enable the card.
 */
int main(int argc, char *argv[])
{
    int flags;

    fd = open("/dev/watchdog", O_WRONLY);

    if (fd == -1) {
    fprintf(stderr, "Watchdog device not enabled.\n");
    fflush(stderr);
    exit(-1);
    }

    if (argc > 1) {
    if (!strncasecmp(argv[1], "-d", 2)) {
        flags = WDIOS_DISABLECARD;
        ioctl(fd, WDIOC_SETOPTIONS, &flags);
        fprintf(stderr, "Watchdog card disabled.\n");
        fflush(stderr);
        exit(0);
    } else if (!strncasecmp(argv[1], "-e", 2)) {
        flags = WDIOS_ENABLECARD;
        ioctl(fd, WDIOC_SETOPTIONS, &flags);
        fprintf(stderr, "Watchdog card enabled.\n");
        fflush(stderr);
        exit(0);
    } else {
        fprintf(stderr, "-d to disable, -e to enable.\n");
        fprintf(stderr, "run by itself to tick the card.\n");
        fflush(stderr);
        exit(0);
    }
    } else {
    fprintf(stderr, "Watchdog Ticking Away!\n");
    fflush(stderr);
    }
```

```
    while(1) {
    keep_alive();
    sleep(1);
    }
}
```