|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **ПРОГРАМНОЕ ОБЕСПЕЧЕНИЕ ЭВМ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ (ИУ7)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.03 ПРОГРАММНАЯ ИНЖЕНЕРИЯ**

**Отчет**

|  |  |
| --- | --- |
| **По лабораторной работе №** | 9 |

**Дисциплина:** Операционные системы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ7-62Б |  |  | Н.А. Гарасев |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | Н.Ю. Рязанова |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2020

**Задание 1:**

* Написать загружаемый модуль ядра, в котором зарегистрировать обработчик аппаратного прерывания с флагом IRQF\_SHARED.
* Инициализировать тасклет.
* В обработчике прерывания запланировать тасклет на выполнение.
* Вывести информацию о тасклете используя, или printk(), или seq\_file interface - <linux/seq\_file.h> (Jonathan Corber: <http://lwn.net//Articales//driver-porting/>).

**Код программы.**

|  |
| --- |
| #include <linux/module.h>  #include <linux/kernel.h>  #include <linux/init.h>  #include <linux/interrupt.h>  #include <linux/timex.h>  MODULE\_LICENSE("GPL");  static int my\_irq = 1, my\_dev\_id, irq\_cnt = 0;  void tasklet\_function(unsigned long data);  char my\_tasklet\_data[] = "my\_tasklet\_function was called";  DECLARE\_TASKLET(my\_tasklet, tasklet\_function, (unsigned long)&my\_tasklet\_data);  void tasklet\_function(unsigned long data) {  printk(KERN\_INFO "++ tasklet\_function was called. state '%ld' count '%i' data '%s'\n", my\_tasklet.state, my\_tasklet.count, (char\*)data);  }  static irqreturn\_t my\_irq\_handler(int irq, void \*dev) {  if (irq == my\_irq) {  printk(KERN\_INFO "++ my\_irq\_handler was called %d time(s)\n", ++irq\_cnt);  tasklet\_schedule(&my\_tasklet);  return IRQ\_HANDLED;  }  return IRQ\_NONE;  }  static int \_\_init my\_tasklet\_init(void) {  if (request\_irq(my\_irq, my\_irq\_handler, IRQF\_SHARED, "my\_tasklet", &my\_dev\_id)) {  printk(KERN\_ERR "++ can't get assigned IRQ %i\n", my\_irq);  return 1;  }  printk(KERN\_INFO "++ Successfully loaded handler for IRQ %d\n", my\_irq);  return 0;  }  static void \_\_exit my\_tasklet\_exit(void) {  tasklet\_kill(&my\_tasklet);  synchronize\_irq(my\_irq);  free\_irq(my\_irq, &my\_dev\_id);  printk(KERN\_INFO "++ tasklet unloaded, irq\_counter = %d\n", irq\_cnt);  return;  }  module\_init(my\_tasklet\_init);  module\_exit(my\_tasklet\_exit); |

**Демонстрация работы программы.**

При каждом прерывании от клавиатуры вызывается обработчик прерывания my\_irq\_handler. В нем вызывается tasklet\_schedule(). При обработке тасклета выводится информация о нем.

Проверим, что my\_tasklet добавилось в список прерываний. Ниже представлено содержимое файла /proc/interrupts, что говорит о том, что my\_tasklet добавилось в список прерываний.

Выгрузим модуль и проверим содержимое системного журнала.

Покажем, что модуль удален. Ниже представлено содержание файла /proc/interrupts, что говорит о том, что my\_tasklet удалилось из списка прерывания после выгрузки модуля.

**Задание 2:**

* Написать загружаемый модуль ядра, в котором зарегистрировать обработчик аппаратного прерывания с флагом IRQF\_SHARED.
* Инициализировать очередь работ.
* В обработчике прерывания запланировать очередь работ на выполнение.
* Вывести информацию об очереди работ используя, или printk(), или seq\_file interface - <linux/seq\_file.h> (Jonathan Corber: <http://lwn.net//Articales//driver-porting/>).