

Московский физико-технический институт (государственный университет)
Факультет радиотехники и кибернетики
Кафедра информатики и вычислительной техники

Выпускная квалификационная работа бакалавра

Разработка драйвера программируемой кнопки на базе контроллера LTC2954

Студент: Козлов Андрей, группа 913

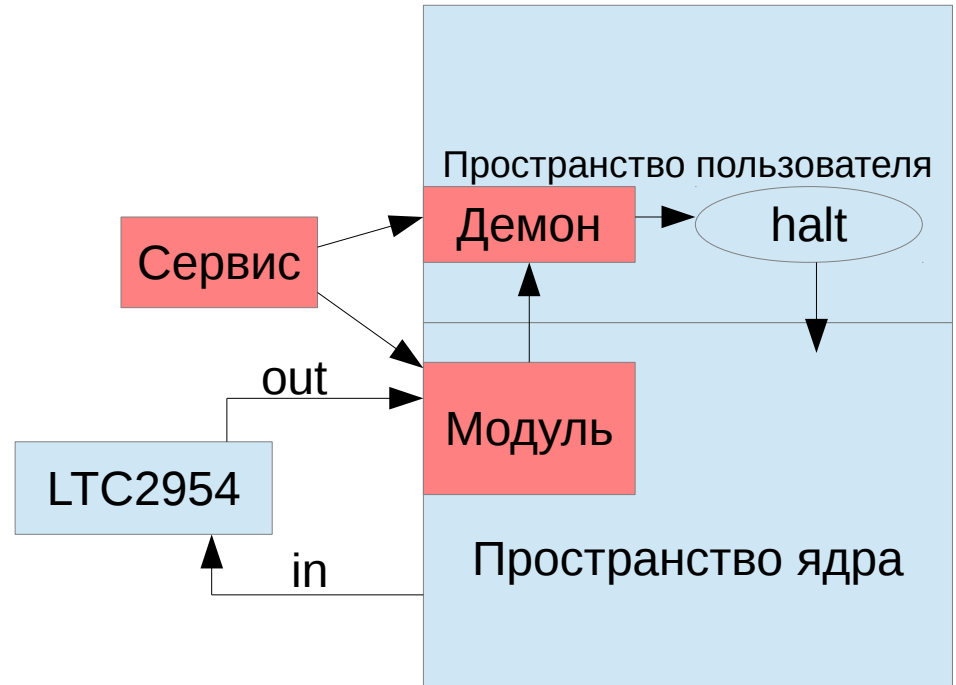
Научный руководитель: д.т.н. проф. Семенихин С.В.



Контроллер кнопки LTC2954

LTC2954 – это контроллер включения/выключения, который управляет системой питания через интерфейс кнопки.

- 1) Получаем сигнал с выхода контроллера (out) и передаем его модулю ядра.
- 2) Модуль обрабатывает сигнал и отправляет его в пространство пользователя демону.
- 3) Демон вызывает команду halt, которая отключает сначала программы пользователя, затем передается ядру.
- 4) Ядро отправляет сигнал отключения на вход контроллера (in).
- 5) Сервис подгружает модуль ядра и запускает демон на старте системы.



Постановка задачи

Разработка программной поддержки кнопки на базе контроллера LTC2954 для материнских плат ЗАО «МЦСТ».

Данная задача включает в себя написание 3-х программ:

- **Модуль ядра** — обеспечивает взаимодействие контроллера с пространством ядра.
- **Демон** для опроса устройства и запуска, заложенного в него, скрипта.
- **Сервис** для подгрузки модуля и запуска демона.

Модуль ядра

Написание программы модуля ядра было сопряжено с реализацией взаимодействия двух подсистем:

- **Подсистема GPIO** – использование конвенции `gpio-lib` для решения задачи унификации драйвера.
- **Подсистема прерываний** – применение конвенции для привязки (программной) пина прерывания контроллера к пространству ядра.



Модуль ядра

General purpose Input/Output

Особенности

Параллельный двунаправленный интерфейс общего назначения.

- пины могут быть сконфигурированы для ввода или вывода
- пин может находиться в состоянии включен/выключен
- выходные значения доступны для записи/чтения
- входные данные могут быть использованы в качестве прерываний (как правило, для событий пробуждения)



Модуль ядра

General purpose Input/Output

Конвенция `gpiolib`

Упрощает работу с `gpio`, предоставляя ряд функций решающих спектр проблем при разработке, позволяет сделать код универсальным и понятным другому разработчику.

- Получение доступа к пину - *`gpio_get_value`*
- Сопоставление прерываний и пинов - *`gpio_to_irq`*
- Занятие и освобождение пинов - `gpio_request/gpio_free`

Через функции конвенции происходит взаимодействие контроллера с пространством ядра.



Модуль ядра

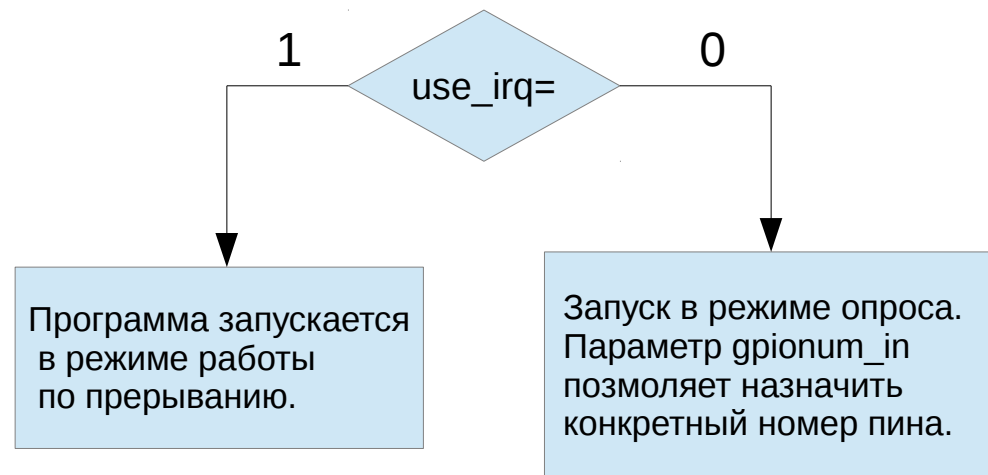
Подсистема прерываний

Режимы работы

Драйвер устройства написан с поддержкой двух режимов:

- С использованием линии прерываний (irq).
- С использованием механизма опроса (poll).

В программе реализован запуск с параметрами, что позволяет выбрать режим работы.

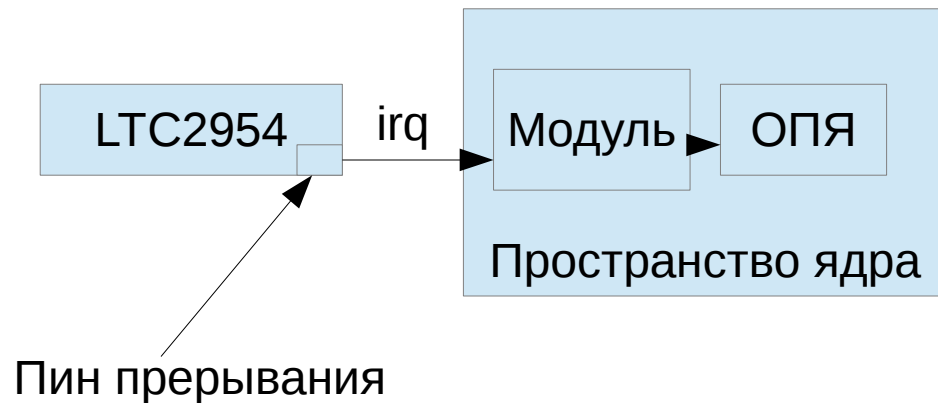


Модуль ядра

Подсистема прерываний

Режимы работы по прерыванию

Мы используем текущую линию прерывания, отлавливаем само прерывание, затем находим соответствующий ему номер `gpio` (функция `gpio_to_irq`) и передаем обработчику прерываний ядра (ОПЯ).

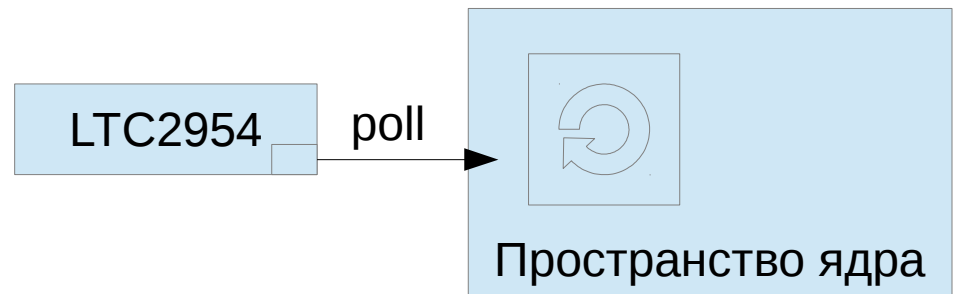


Модуль ядра

Подсистема прерываний

Режимы работы с механизмом опросов

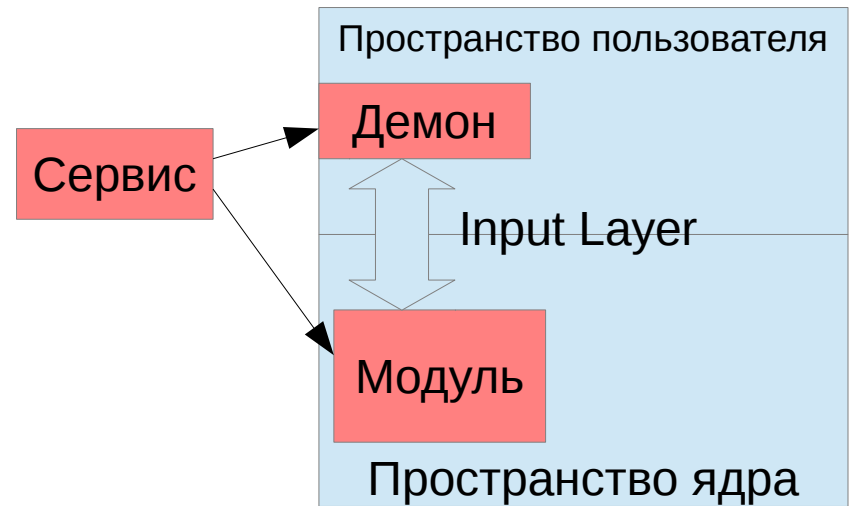
Проблема использования линии прерывания – она может быть занята. Решение проблемы – циклический опрос портов gpio.



Программы уровня пользователя

Подсистема Input Layer(IL)

Для взаимодействия пользовательского пространства с пространством ядра существует подсистема IL. Для управления модулем из пространства пользователя был написан демон, который принимает информацию из IL.



Программы уровня пользователя

Демон

- Демон - опрашивает в бесконечном цикле устройство `/dev/input/eventX`, которое занимает кнопка при подгрузке модуля.
- Демон должен запускаться из сервиса, на старте системы, поэтому он должен быть написан грамотно, то есть с использованием `daemonize`.

Программы уровня пользователя

Демон

Daemonize

- Вызов `fork()` используется для создания отдельного процесса.
- Вызов `setsid()` используется для отделения процесса от родителя.
- Изменить текущий каталог.
- Переоткрыть все `std-` файлы.

Невыполнение любого из этих шагов приведет к неправильной работе демона. Типичные ошибки:

- Запуск демона, а затем выход из системы приведет к зависанию терминала.
- Каталог, из которого запущен демон, будет заблокирован.

Программы уровня пользователя

Сервис

- Сервис - подгружает модуль.
- Находит номер устройства eventX, на котором оказалась кнопка.
- Запускает демон, передавая ему в качестве параметра путь к идентификатору устройства.



Результаты

Реализована программная поддержка кнопки на базе контроллера LTC2954, состоящая из 3-х программ:

- **Драйвер**, работающий в 2-х режимах: по прерыванию и с использованием механизма опроса.
- **Демон**, обеспечивающий взаимодействие пространства пользователя с драйвером, работающим в пространстве ядра.
- Специальная программа **сервис**, отвечающая за запуск драйвера и модуля на старте системы.
- Протестирована на ВК «Монокуб», программная поддержка вошла в поставляемый пользователям дистрибутив для ВК последних моделей.
- Благодаря универсальности решения оно было опубликовано под лицензией GPL на git.mcst.ru, то есть сделано доступным для сообщества.



Спасибо за
внимание!

