

ВИКОРИСТАННЯ ІНТЕРФЕЙСУ SPI НА КОМП'ЮТЕРАХ ПІД УПРАВЛІННЯМ ОС LINUX

Шабалов І. Г.

(Науковий керівник Тарабаров С. Б. к.т.н., доцент)

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Радіотехнічний факультет

В розробці сучасних електронних пристроїв та дослідницькій роботі часто виникає необхідність обміну даними із модулями або мікросхемами давачів або актюаторів. Такий обмін зазвичай відбувається за допомогою спеціальних інтерфейсів зв'язку. У найпростішому випадку, засоби для обміну даними вбудовані в середовище розробки, але найчастіше — необхідно розробляти окрему спеціальну програму для обміну даними. Якщо ця програма працює безпосередньо із периферією мікроконтролера — складностей виникати не повинно, але у випадку коли така програма існує як окремий модуль на базі операційної системи(ОС) — виникають складнощі, пов'язані із роботою з такими інтерфейсами на базі ОС.

В популярній серед розробників електроніки операційній системі(ОС) *Linux* за замовчуванням включено підтримку великої кількості комп'ютерної периферії, через те що ця ОС є відкритою і її розробкою та підтримкою різної периферії може зайнятися будь-який розробник. Але для починаючого розробника не завжди очевидно, як на базі такої ОС зробити програму, що буде використовувати периферійні модулі апаратної платформи.

В ОС *Linux* «будь-що є файл»[1]. І інтерфейси для обміну даними не стали виключенням. Так, в дереві файлів можна знайти пристрої із назвою */dev/spidev0.0*, де перший нуль — це порядковий номер периферійного модулю, а другий — порядковий номер лінії вибору пристрою на цій шині. Тобто якщо апаратно в платформі існують два модулі *Serial Peripheral Interface(SPI)*, кожен з яких має можливість вести обмін даними із чотирма підпорядкованими йому пристроями, тобто має чотири лінії вибору пристроя (англ. *Chip Select, CS*) — в системі можна буде побачити файли із назвами */dev/spidev[0-1].[0-3]*.

У випадку написання своєї програми для роботи із, наприклад, інтерфейсом *SPI*, у, так званому, просторі користувача ОС *Linux*, розробник має менше можливостей використовувати апаратні засоби напряду: використання прямого доступу до пам'яті (англ. *Direct Memory Access, DMA*), або апаратних переривань (англ. *Interrupts*) в такому випадку не дозволяється. Для застосування всіх апаратних можливостей платформи необхідно писати свою програму в просторі ядра ОС *Linux*, але це потребує досить глибоких знань про внутрішні механізми роботи операційних систем в цілому, та гарні знання щодо створення драйверів пристроїв (англ. *Linux Device Drivers, LDD*), які, в свою чергу, є модулями ОС (англ. *Linux Kernel Modules, LKM*). Але оскільки зазвичай

доступ до таких пристроїв з боку програм користувача є унікальним, а об'єм даних, що поширюється за їх допомогою є порівняно малим — різниця у швидкості роботи програми із використанням лише простору користувача, та програми із використанням простору ядра ОС, не є помітною. Тому в загальному випадку доцільніше використовувати перший підхід через його простоту.

Оскільки репрезентацією пристрою інтерфейсу *SPI* є системні звичайні системні файли — їх можна відкривати звичайним викликом системної функції *open()*. Запис та зчитування даних з таких файлів пристроїв можна проводити використовуючи системні функції *write()* та *read()*. Після закінчення роботи із інтерфейсом необхідно закрити файл викликом функції *close()*. Досить поширеною помилкою є нехтування закриванням системного файлу, що призводить до помилок в подальшій роботі із пристроєм через те що системні ресурси не були звільнені належним чином.

Для налаштування параметрів такого інтерфейсу необхідно скористатись нетиповою для звичайного програмування функцією *ioctl()*[2]. Використовуючи цю функцію користувач може передати їй наступні параметри для налаштування:

- *SPI_IOC_WR_MODE* — налаштування режиму роботи інтерфейсу: за яким фронтом зчитувати тактовий сигнал, та яким буде рівень на лінії тактування у стані спокою на шині. Всього існує чотири режими[3];

- *SPI_IOC_WR_BITS_PER_WORD* — кількість бітів що буде передана однією «порцією»: у типових модулях інтерфейсу *SPI* можливо два варіанти — 8 та 16 біт;

- *SPI_IOC_WR_MAX_SPEED_HZ* — вибір частоти тактування периферійного модулю і, відповідно, швидкості передачі даних по інтерфейсу. Налаштування залежить від конкретної апаратної платформи. Типові значення лежать в діапазоні від одиниць кілогерц до сотень мегагерц;

- *SPI_IOC_WR_LSB(MSB)_FIRST* — встановлення порядку проходження бітів інформації через середовище передачі даних. Існує два варіанти: *MSB(Most Significant Bit)* — коли найстарший біт відправляється першим, а також *LSB(Least Significant Bit)* — коли наймолодший біт відправляється першим.

-

Література

1. Explanation of “Everything is a File” and Types of Files in Linux. [Електронний ресурс]. Доступно за посиланням: <https://www.tecmint.com/explanation-of-everything-is-a-file-and-types-of-files-in-linux/>.
2. Ioctl, Linux Programmer's Manual. [Електронний ресурс]. Доступно за посиланням: <http://man7.org/linux/man-pages/man2/ioctl.2.html>.
3. Introduction to SPI Interface. [Електронний ресурс]. Доступно за посиланням: <https://www.analog.com/en/analog-dialogue/articles/introduction-to-spi-interface.html>.