УДК 681.5.01

Л.А. Савицька
А.В. Процюк

СИСТЕМА КЕРУВАННЯ ПЕРИФЕРІЙНИМИ ПРИСТРОЯМИ З ВИКОРИСТАННЯМ ВБУДОВАНИХ ІНТЕРФЕЙСІВ

Вінницький національний технічний університет

Анотація

У роботі розроблено та досліджено систему керування периферійними пристроями з використанням вбудованих інтерфейсів мікроконтролера на базі апаратної платформи STM32 Nucleo-F401RE (ARM Cortex-M4). Система забезпечує збір інформації від сенсорів, обробку отриманих даних, формування керуючих сигналів для виконавчих пристроїв та передавання діагностичної інформації на персональний комп'ютер. Реалізовано взаємодію основних периферійних модулів мікроконтролера — GPIO, USART і DMA, а також обробку зовнішніх подій за допомогою контролера переривань EXTI.

Програмну частину виконано мовою C у двох варіантах реалізації: з автоматичною генерацією ініціалізаційного коду в середовищі STM32CubeIDE та з ручним налаштуванням регістрів у середовищі Keil µVision5. Передача даних організована за схемою «пам'ять → периферія» через контролер DMA, що дозволяє зменшити навантаження на центральний процесор і підвищити швидкодію обміну. Працездатність системи підтверджено сценарним тестуванням на апаратному стенді з PIR-датчиком руху, реле та світлодіодним навантаженням; результати візуалізовано у термінальній програмі та засобами Python.

Ключові слова: вбудовані системи, ARM-мікроконтролер, STM32, периферійні пристрої, USART, GPIO, DMA, STM32CubeIDE, Keil µVision5, мова C.

Abstract

This work presents the design and study of a control system for peripheral devices using the embedded interfaces of a microcontroller, based on the STM32 Nucleo-F401RE platform (ARM Cortex-M4). The system acquires sensor data, processes it, generates control signals for actuators, and transmits diagnostic information to a personal computer. The interaction of the main peripheral modules — GPIO, USART and DMA — is implemented, together with external event handling via the EXTI interrupt controller.

The software part is written in C in two implementation variants: with automatic initialization-code generation in STM32CubeIDE and with manual register configuration in Keil µVision5. Data transfer is organized as a memory-to-peripheral transaction through the DMA controller, which reduces the CPU load and improves the throughput of the exchange. The system was validated by scenario testing on a hardware stand with a PIR motion sensor, relays and an LED load; the results were visualized in a terminal program and with Python tools.

Keywords: embedded systems, ARM microcontroller, STM32, peripheral devices, USART, GPIO, DMA, STM32CubeIDE, Keil µVision5, C language.

Вступ

Стрімкий розвиток мікропроцесорної техніки зумовлює розширення сфери застосування вбудованих систем у промисловій автоматизації, транспорті, медичному обладнанні, системах безпеки та побутовій електроніці. Однією з ключових складових будь-якої вбудованої системи є периферійні пристрої, через які здійснюється взаємодія із зовнішнім середовищем, а ефективність роботи системи значною мірою залежить від правильного використання вбудованих інтерфейсів мікроконтролера — GPIO, USART та DMA.

Актуальність теми обумовлена широким застосуванням мікроконтролерних систем у різних галузях техніки та необхідністю створення ефективних засобів взаємодії з периферійними пристроями, що забезпечують надійний обмін даними та підвищують продуктивність системи при низькому навантаженні на центральний процесор.

Мета роботи — розробити систему керування периферійними пристроями з використанням вбудованих інтерфейсів на базі ARM-мікроконтролера сімейства STM32, що забезпечує збір даних від сенсорів, їх обробку, керування виконавчими пристроями та передавання інформації на ПК.

Для досягнення мети вирішено такі задачі:

– проаналізувати сучасний стан розвитку вбудованих систем та критерії вибору мікроконтролерної платформи;
– обґрунтувати вибір апаратної платформи STM32 Nucleo-F401RE та середовищ розробки;
– вивчити принципи роботи інтерфейсів GPIO, USART і технології прямого доступу до пам’яті DMA;
– реалізувати програмні засоби взаємодії мікроконтролера з периферією у двох варіантах — з генерацією коду та без неї;
– виконати налаштування та тестування системи керування периферією;
– провести аналіз і візуалізацію результатів роботи системи.

Загальні відомості та обґрунтування вибору технологій

Вибір мікроконтролерної платформи є багатокритеріальною задачею, що враховує енергоефективність, обсяг і швидкість пам’яті, умови експлуатації, наявність периферійних інтерфейсів, вартість та габарити. Для систем збору даних і керування периферією визначальними є різноманітність вбудованих інтерфейсів та повний доступ до ресурсів мікроконтролера.

Для реалізації обрано:
– STM32 Nucleo-F401RE (ядро ARM Cortex-M4, FPU, 512 КБ Flash, 96 КБ RAM, до 84 МГц) — завдяки відкритій архітектурі, підтримці стандартів розширення Arduino Uno Rev3 і ST morpho та вбудованому програматору ST-LINK/V2-1;
– USART — як основний канал послідовного обміну даними з ПК (асинхронний режим, 115200 бод, 8 біт даних, без парності, 1 стоп-біт);
– DMA — для організації передачі «пам’ять → периферія» без участі CPU;
– GPIO та контролер зовнішніх переривань EXTI — для зчитування сигналів сенсорів і кнопок та керування виконавчими елементами;
– STM32CubeIDE (графічний конфігуратор CubeMX, GCC, GDB) — для швидкої розробки з автоматичною генерацією коду;
– Keil µVision5 (CMSIS, ручне налаштування регістрів) — для низькорівневого контролю та оптимізації.

Архітектура системи

Архітектура системи включає три логічні рівні:
– Рівень периферії: PIR-датчик руху HC-SR501, тактові кнопки керування, електромагнітні реле, транзисторні ключі та світлодіодне навантаження, що забезпечують взаємодію із зовнішнім середовищем.
– Рівень мікроконтролера: STM32F401RE з ядром Cortex-M4, системними шинами AHB/APB, контролером переривань NVIC/EXTI, модулями GPIO, USART2 та DMA1, які реалізують збір, обробку й передавання даних.
– Рівень візуалізації: персональний комп’ютер, що приймає дані через USART (термінальна програма або застосунок на Python) і відображає стани системи у часі.

Обмін між рівнем мікроконтролера та ПК відбувається за послідовним протоколом USART, при цьому основний потік даних проходить від оперативної пам’яті через контролер DMA до інтерфейсу USART, що мінімізує навантаження на центральний процесор.

РІВЕНЬ ПЕРИФЕРІЇ
PIR-датчик HC-SR501 | кнопки | реле | світлодіодне навантаження
↕ сигнали GPIO / EXTI ↕
РІВЕНЬ МІКРОКОНТРОЛЕРА (STM32F401RE, ARM Cortex-M4)
GPIO → EXTI → логіка керування → DMA1 → USART2
↕ послідовний обмін USART ↕
РІВЕНЬ ВІЗУАЛІЗАЦІЇ
ПК: термінал / застосунок Python — відображення станів системи

Рисунок 1 – Структурна схема системи

Функціональні модулі:
– Ініціалізація периферії: налаштування тактування, GPIO, USART та DMA (через CubeMX або вручну через регістри).

– Модуль GPIO: зчитування станів датчиків і кнопок, керування виконавчими пристроями через регістри ODR/BSRR.
– Модуль USART: асинхронний обмін діагностичними повідомленнями та командами з ПК.
– Модуль DMA: передача даних «пам’ять → периферія» з інкрементом пам’яті та перериванням завершення передачі.
– Обробка переривань EXTI: реакція на зовнішні події (рух, натискання кнопок) у реальному часі.
– Модуль візуалізації: приймання даних на ПК і відображення станів системи (on_simple, on_pir, off).

Результати роботи системи

У результаті розробки створено працюючу програмно-апаратну систему, яка:
– забезпечує повний цикл взаємодії з периферією — від зчитування сигналів сенсорів до керування виконавчими пристроями;
– реалізує обмін даними з ПК через USART з апаратним прискоренням засобами DMA;
– обробляє зовнішні події в реальному часі через контролер переривань EXTI;
– підтримує два режими керування освітленням — ручний та автоматичний (за сигналом PIR-датчика);
– реалізована у двох варіантах — з генерацією коду (STM32CubeIDE, HAL) і без неї (Keil µVision5, рівень регістрів) — для порівняння підходів.

Перевірку працездатності виконано шляхом сценарного тестування на апаратному стенді: натискання кнопок керування, спрацювання інфрачервоного датчика та перемикання режимів роботи. Візуалізація підтвердила чітку зміну станів у часі, стабільну роботу системи та відсутність критичних затримок у передачі даних.

Висновки

Розроблено та досліджено систему керування периферійними пристроями з використанням вбудованих інтерфейсів на базі STM32 Nucleo-F401RE. Реалізовано конфігурацію та взаємодію модулів GPIO, USART і DMA, а також обробку зовнішніх переривань EXTI. Організація обміну даними через USART із застосуванням DMA дозволила зменшити навантаження на центральний процесор і підвищити ефективність передачі інформації.

Порівняння реалізацій у STM32CubeIDE та Keil µVision5 показало, що високорівневий підхід забезпечує швидке прототипування завдяки автоматичній генерації коду, тоді як низькорівневий надає повний контроль над апаратними ресурсами та можливість оптимізації. Отримана функціональна модель може застосовуватися в навчальних лабораторіях, системах автоматизації, моніторингу та пристроях збору даних, а також як основа для подальшого розширення функціоналу.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. STM32 Nucleo-64 development board with STM32F401RE MCU. STMicroelectronics. URL: https://www.st.com/en/evaluation-tools/nucleo-f401re.html
2. STM32F401xE Reference Manual (RM0368). STMicroelectronics, 2021. URL: https://www.st.com/
3. STM32CubeIDE: Integrated Development Environment. STMicroelectronics. URL: https://www.st.com/en/development-tools/stm32cubeide.html
4. Keil µVision IDE. URL: https://www2.keil.com/mdk5/uvision/
5. Yiu J. The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors. 3rd ed. Newnes, 2014. 864 p.

Савицька Людмила Андріївна — доцент кафедри обчислювальної техніки, факультет інформаційних технологій та комп’ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: savytska.liudmyla@vntu.edu.ua.

Процюк Андрій Васильович — студент, факультет інформаційних технологій та комп’ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: andriyprotsiuk@gmail.com.

Колесник Ірина Сергіївна — кандидат технічних наук, доцент, Вінницький національний технічний університет, м. Вінниця.

Savytska Liudmyla Andriivna — Associate Professor, Department of Computer Engineering, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia.

Protsyuk Andriy Vasylovych — student, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia.

Kolesnyk Iryna Serhiivna — Candidate of Technical Sciences, Associate Professor, Vinnytsia National Technical University, Vinnytsia.