

**К.т.н., доцент Плахотний М.В.,
студент Биков С.П., студент Пелих О.М.**

**Національний технічний університет України
"Київський політехнічний інститут"**

ВИКОРИСТАННЯ КОНТРОЛЕРІВ FREESCALE В НАВЧАЛЬНОМУ ПРОЦЕСІ

Вступ

На даний час контролери *Freescale* мало використовуються в навчальному процесі та не широко розповсюджені в різноманітних технічних засобах, а саме в системах керування, контролю та діагностики. Задачею є практичне застосування однієї з електронних систем компанії *Freescale* в навчальній дисципліні „Розробка вбудованих мікропроцесорних систем”.

Постановка задачі

За мету було поставлено розробку системи керування рухомих об'єктом з передаванням даних та керуючих команд за допомогою контролера від комп'ютера до об'єкта по бездротовому каналу.

Керованим об'єктом обрано чотириколісне шасі, обладнане електричним двигуном на задній осі, що живиться від елемента живлення на 12 В, та механізмом повороту передніх коліс.

Задача звелась до встановлення на рухомий об'єкт апаратури з можливістю аналізу даних, підтримки двонапрямого зв'язку з комп'ютером та керування рухомих об'єктом, а також створення програмного забезпечення, яке б задовольняло потреби передачі та аналізу даних для керування.

Для максимального прискорення розробки такого комплексу прийнято рішення використовувати досить розповсюджені пристрої відомих виробників, зокрема компанії *Freescale*, які б мали спільні інтерфейси та мали такі компоненти як:

– ЦАП для виведення керуючих сигналів на схему керування двигуном та, додатково, АЦП для можливості підключення пристроїв зняття інформації про навколишнє середовище;

- мікроконтролер для аналізу і підготовки даних з інтерфейсами зв'язку, сумісними з пристроєм бездротового зв'язку;
- бездротовий зв'язок з базовою станцією керування.

Виробником мікроконтролера вибрано *Freescale* через широкий набір модифікацій та прозору і зрозумілу систему підбору компонентів за допомогою спеціальної утиліти, розміщеної на веб-порталі компанії. При виборі технології бездротового зв'язку між *Wi-Fi* та *Bluetooth* від технології *Wi-Fi* довелося відмовитись через порівняно недешеві рішення з можливістю підключення до мікроконтролера через *COM*-порт та через завеликі потреби щодо живлення апаратури, а також проблеми з придбанням такого обладнання. В результаті пошуків обрано *Bluetooth* адаптер для *COM*-порту виробництва компанії *SENA* [1].

Вибір конкретної схеми мікроконтролера від *Freescale* став би реальною проблемою через надвелику їх кількість та складність з визначенням мінімально-допустимих параметрів та вимог, якби не існування утиліти від *Freescale*. Вона дозволяє відфільтровувати моделі, які мають лише ті можливості, які потрібні користувачу [2]. Для реалізації поставленої задачі обрано комплект від *Freescale* "*16-Bit HCS12C32 Student Learning Kit*", що базується на мікроконтролері *Freescale HCS12C32*.

Комплект "*16-Bit HCS12C32 Student Learning Kit*" [3] виробництва компанії *Freescale* складається з двох основних частин:

1. *HCS12* – мікроконтролер з такими основними параметрами:
 - 32 Кбайти флеш-пам'яті,
 - 2 Кбайти оперативної пам'яті,
 - порт передачі даних *RS-232*,
 - вибір рівня логічної "1" від +3,3 В до +5 В,
 - напруга живлення від +6 В до +16 В;
2. *PBMCUSLK* – базова станція для розробки і налагодження мікроконтролерів.

Після вибору всіх пристроїв для проекту вирішено на рухомий об'єкт встановлювати тільки мікроконтролер без базової станції, що теж можливо. Це значно полегшило розміщення компонентів на об'єкті.

Для живлення двигуна та поворотного механізму створено акумуляторний блок з восьми нікель-метал-гідридних акумуляторів по 1,5 В кожний, а для контролера – послідовно з'єднано дві батарейки по 4,5 В. Такі рішення найбільш вигідні з точки зору мобільності та ціни.

Програмне забезпечення

Мікроконтролер запрограмовано на мові програмування *C++* в середовищі розробки *CodeWarrior Development Studio*.

Розроблено також програмне забезпечення для комп'ютера на мові C#, за допомогою якого передаватимуться інструкції з керування об'єктом по каналу *Bluetooth*. Мікроконтролер отримуватиме ці інструкції за допомогою *Bluetooth* адаптера *SENA*.

Розробка пристрою перетворення сигналів мікроконтролера

Вихідні сигнали мікроконтролера на управління рухом об'єкта вперед, назад, поворот коліс вліво та вправо, та вибір однієї з двох можливих швидкостей руху об'єкта, треба перетворити на керуючі електричним двигуном та поворотним механізмом напруги. Дані функції реалізовані в пристрої перетворення сигналів мікроконтролера (рис.1).

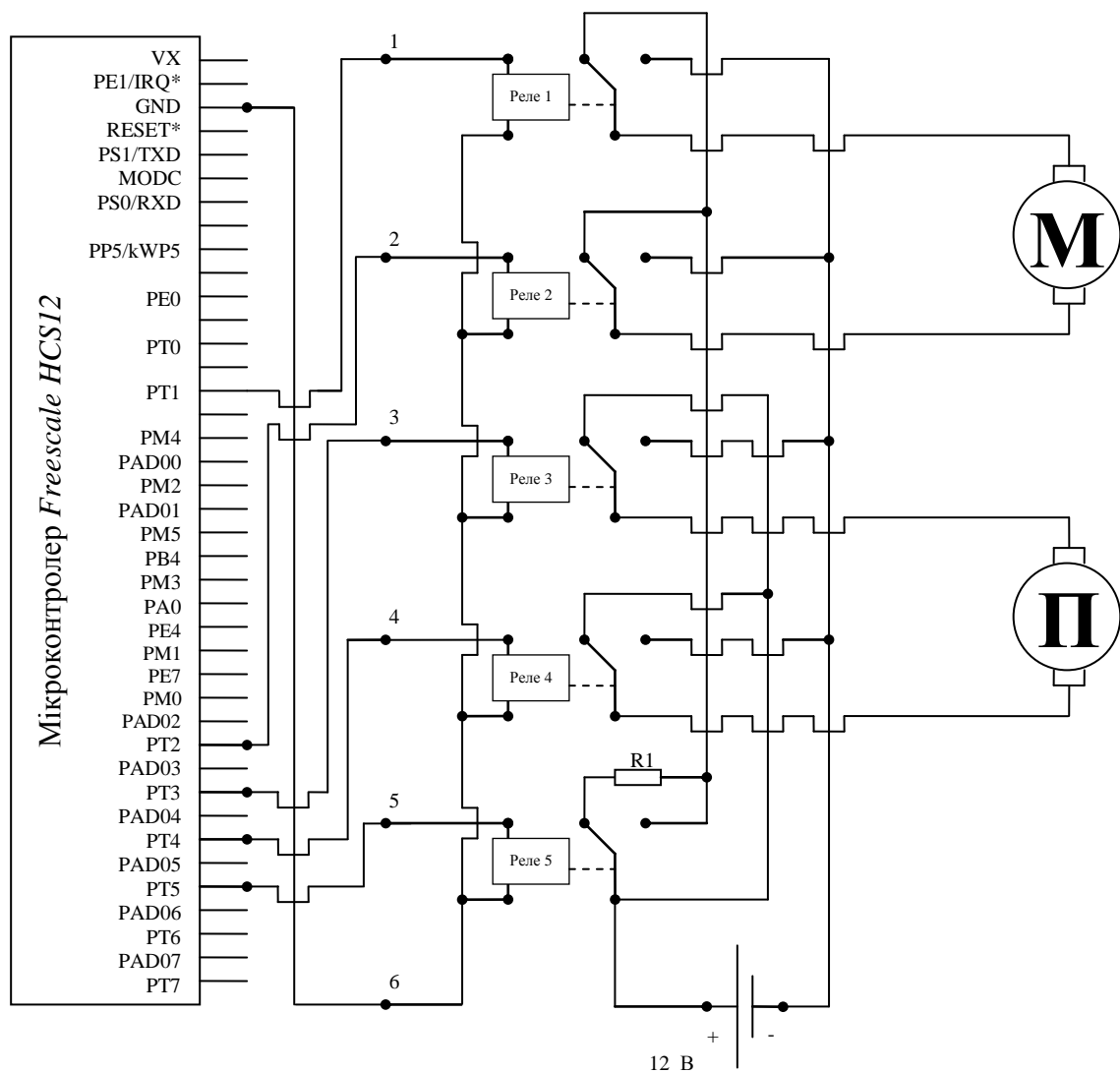


Рис.1. Електрична схема перетворення сигналів мікроконтролера, де М – електричний двигун, П – пристрій повороту коліс.

На рис.1 зображено вхідні контакти пристрою перетворення сигналів мікроконтролера з 1-го по 6-ий. Контакт номер 1 означає команду руху вперед і замикає Реле1, 2 – у зворотному напрямі (Реле2), 3 – поворот коліс вліво (Реле3), 4 – вправо (Реле4), а 5 – зміну швидкості на більшу (Реле5).

Важливою особливістю цієї схеми є те, що двигун та пристрій повороту коліс вмикаються в різних напрямках обертання, в залежності від напрямку поданого на них струму. Швидкість обертання двигуна змінюється зі зміною напруги, яка подається на двигун. Це реалізовано за допомогою резистора $R1$ (4,7 кОм, 5Вт).

В результаті розробки комплексу отримано систему передачі даних, організовану за схемою на рис.2.

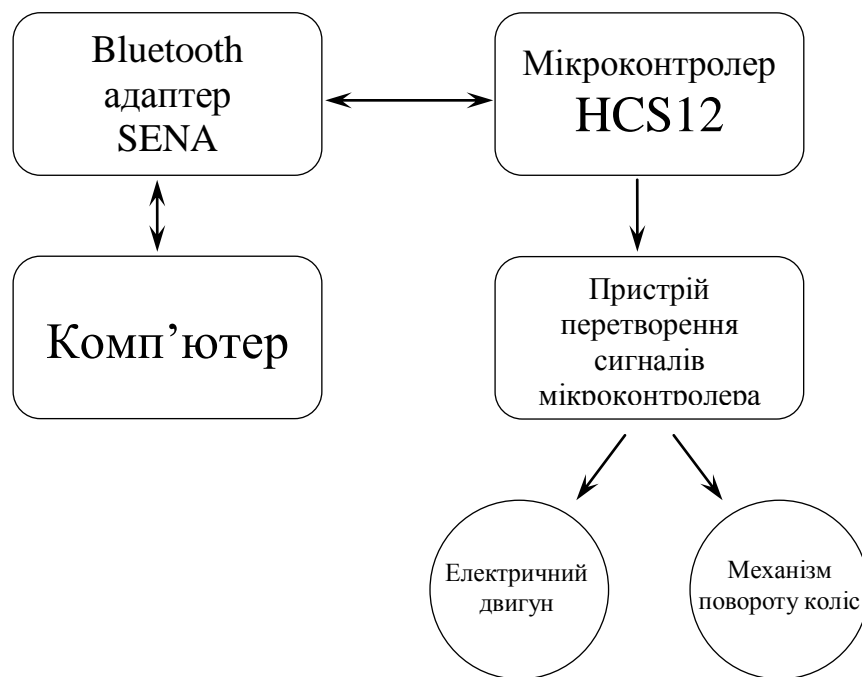


Рис.2. Схема системи передачі сигналів керування

Висновки

Розроблено систему, яка демонструє практичне використання контролерів *Freescale*. Цей приклад дає можливість студентам побачити результат проектів по курсу „Розробка вбудованих мікропроцесорних систем”.

Надалі планується продовження робіт в напрямку підключення до контролера, розташованого на рухомому об'єкті, пристроїв зняття інформації про навколишнє середовище, датчиків руху та пристроїв замірювання відстані до перешкод.

Література

1. http://www.sena.com/products/industrial_bluetooth/sd.php
2. <http://www.freescale.com/webapp/search/MainSERP.jsp?SelectedAsset=Product%20Pages>
3. http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=PBMCUSLK