

Програмно-апаратний комплекс для управління рухомим об'єктом

Павло Гуменюк¹, Лариса Гуменюк²,
Володимир Лотиш²

1. Кафедра автомобілів, Національний транспортний університет, УКРАЇНА, м. Київ, вул. Суворова 1, E-mail: nacasom@gmail.com

2. Кафедра автоматизованого управління виробничими процесами, Луцький національний технічний університет, УКРАЇНА, м. Луцьк, вул. Потебні, 56, E-mail: lorapost@gmail.com

Abstract – The article substantiates the need to develop a model of the object movement to study its behavior under various conditions and under the influence of various external factors. The software to simulate the movement of two-linked wheeled object with reference connection between the units was developed. Powered algorithm of the basic mechanisms of the facility to ensure the sustainability movement was constructed. Software development environment for the AVR ATmega328P microcontroller board in the Arduino Uno, which is placed on board of the object model was selected. The connection to the virtual COM-port system from a higher level was described. The model allows to perform any maneuvers of two-linked wheeled object, including pre- registered at 90o turning maneuvers and lane changes for the study of factors influencing the stability of the object without costly and dangerous experiments was created.

Ключові слова – модель, рухомий об'єкт, Arduino, програмне забезпечення.

I. Вступ

Розробка алгоритмів для створення автоматизованої системи керування рухомим об'єктом потребує інформації про динаміку руху та її залежності від зовнішніх факторів. Використання реальних об'єктів для проведення експериментальних досліджень є досить затратним та небезпечним. Одним з методів вирішення цієї проблеми є моделювання. Розробка імітаційної моделі руху об'єкта вирішує проблему дослідження управління цим об'єктом. Застосування програмних пакетів для віртуального моделювання дозволяє отримати інформацію про об'єкт та його поведінку в різних умовах та під впливом тих чи інших зовнішніх факторів. Використання такої моделі реалізує можливість дослідження факторів впливу на стабільність руху даного об'єкта без проведення затратних та небезпечних натурних експериментів.

II. Опис моделі

Метою даної роботи є створення та дослідження моделі процесу керування дволанковим колісним об'єктом з опорним навантаженням. Створення моделі включає в себе розробку алгоритмів керування рухом об'єкта та відповідного програмного забезпечення. Робота цих алгоритмів перевірена на фізичній моделі. Сам процес управління здійснюється на основі програмно-апаратної системи.

Перевагою фізичної моделі є те, що для неї не обов'язкова модель математична, отже спрощуються розрахунки, а дані можна отримувати прямо з об'єкта дослідження, використовуючи датчики, записуючі елементи і т.п., а рух задавати виконавчими пристроями. Разом з тим, точність результатів залежить від точності копіювання оригіналу.

У створеній фізичній моделі керівні команди задаються мікроконтролером AVR ATmega328P у складі плати Arduino Uno, яка розміщена на борту моделі об'єкта. Також використовується плата розширення Arduino Motor Shield на основі мікросхеми L293, для керування двигунами постійного струму і сервоприводами.

III. Програмне забезпечення моделі

Електронний обчислювальний пристрій (мікропроцесор, чіп, контролер, т.п.) виконує наперед вказані дії залежно від записаних у нього інструкцій. Для кожної дорожньої ситуації набір подібних інструкцій може відрізнятись, а їхня кількість сягає тисяч рядків програмного коду. У випадку збою або некоректної обробки даних певною функцією можна отримати небажаний результат роботи системи в цілому, навіть при успішному спрацюванні всіх фізичних виконавчих органів. Написання якісного програмного коду є не менш важливим завданням, ніж вибір апаратної частини системи.

Модель рухомого об'єкта керується за допомогою плати Arduino, тому середовищем розробки є багатоплатформовий Java-додаток, заснований на мові Processing. Синтаксис даного середовища подібний до C++, але використовує деякі додаткові бібліотеки. Програми, створені для роботи з платою Arduino, називаються скетчами. Після успішної компіляції скетч передається у процесор засобами віртуального COM-порту. Вбудований у середовище розробки монітор дозволяє реалізувати зворотній зв'язок із платою у процесі виконання програми (Рис. 1).

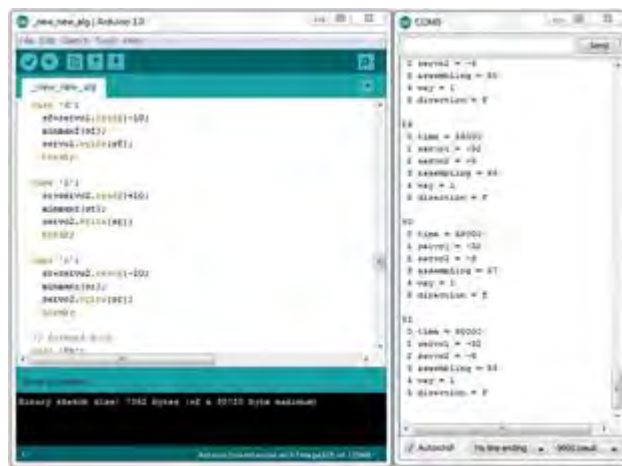


Рис. 1. Середовище розробки скетчів Arduino (зліва) і серійний монітор (справа)

Можлива передача команд процесору, а також зчитування і відображення даних. Разом з тим, серій-

ний монітор не має графічної оболонки чи механізму збереження відображуваних даних. Тому, для полегшення роботи з моделлю і нарощування її функціональності засобами мови Delphi було написано Windows-сумісний додаток (Рис. 2). У додатку керування моделлю здійснюється через панель управління. Достатньо натискати на відповідні кнопки замість введення команди із клавіатури, що зручно і економить час користувача.

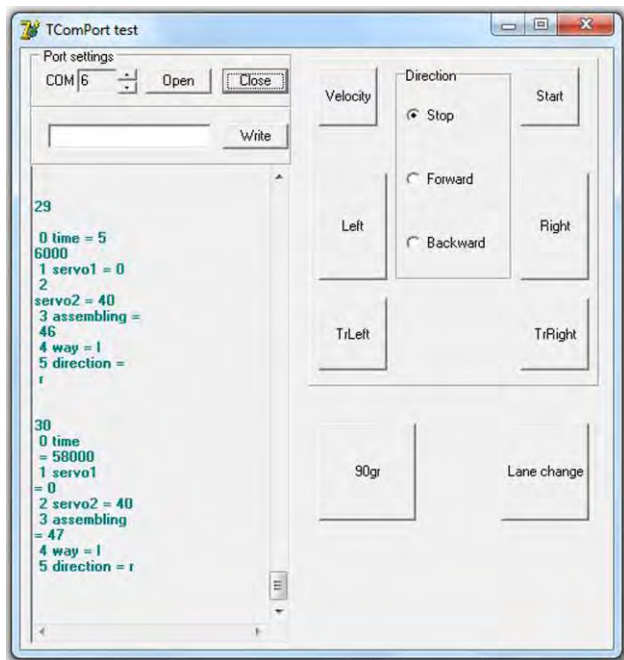


Рис. 2. Windows-додаток для роботи з моделлю

Оскільки COM-порт працює в середовищі Win32, то і підключення відбувається з використанням специфічних функцій WinAPI подібно до роботи зі звичайними файлами. Будь-яка робота з портом починається з його відкриття, тому після запуску додатка в першу чергу потрібно вказати номер віртуального COM-порта, до якого підключена плата, і відкрити його. Обмін даними з портом відбувається у асинхронному режимі, тобто операції зчитування і запису в порт виконуються паралельно з різних потоків. Мінімальна одиниця інформації, що передається у цьому режимі, – один байт, тобто один символ. Передача кожного байта починається із старт-біта, що сигналізує приймачу про початок послідовності, за яким ідуть біти даних і, можливо, біт паритету (парності). Завершує послідовність стоп-біт, що гарантує паузу між послідовностями. Старт-біт наступного байта посилається у будь-який момент після стоп-біта, тобто між передачами можливі паузи довільної тривалості. Старт-біт, що має завжди строго певне значення (логічний 0), забезпечує простий механізм синхронізації приймача по сигналу від передавача, тобто приймач і передавач працюють на одній швидкості обміну. Даний режим, на відміну від синхронного, більше імпонує ідеї багатозадачності Windows.

Після вибору номера віртуального COM-порта і натиснення на кнопку Open, якщо до вказаного порта

дійсно підключено обладнання, відразу ж почнеться обмін даними із процесором. Команди на плату Arduino можна передавати як за допомогою текстового рядка, так і натискаючи на потрібні кнопки панелі управління.

У додатку передбачено два маневри повороту на 90° і зміни смуги руху – кнопки 90gr та Lane change відповідно. Контролер автоматично встановить потрібні значення кутів повороту і буде їх змінювати так, щоб траєкторії характерних точок всіх ланок об'єкта збігались. Дані, отримані від процесора відображаються у вікні зліва, під рядком вводу команд і автоматично записуються у текстовий файл для можливості подальшої роботи з ними (Рис. 3).

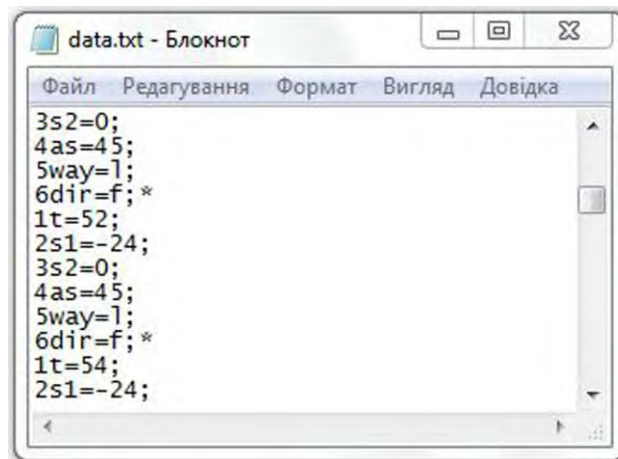


Рис. 3. Дані, записані додатком.

Висновок

Створена модель дволанкового колісного об'єкта з опорним навантаженням дозволяє перевіряти керівні алгоритми асистивних систем для автопоїздів. Розроблене програмне забезпечення містить дві складові: систему управління низького рівня для роботи із електромоторами і сервоприводами та систему управління вищого рівня для здійснення загального керування, задання наперед прописаних маневрів, протоколювання результатів роботи моделі у режимі реального часу та забезпечення збереження результатів експериментів

References

- [1] P. J. McNaull, Modelling and validation of heavy truck moder with electronic stability control: Thesis. Ohio: The Ohio State University, 2009
- [2] Sakhno V.P., Lotysh V.V., Gumeniuk P.O., "The improvement of vehicle stability by means of ESC system", Project Management, System Analysis and Logistics: Scientific Journal. Kyiv: National Transport University, 2011, no. 8.
- [3] Gumeniuk P.O., "The development of scaled road tractor model", in Proceedings of the XV International scientific conference "Road transport: problems and prospects": September 10-17, 2012, Sevastopol, Ukraine.