УДК 629.113
П.О. Гуменюк, Л.О. Гуменюк, В.В.Лотиш
Луцький національний технічний університет

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ КЕРУВАННЯ ФІЗИЧНОЮ МОДЕЛЛЮ АВТОПОЇЗДА

Гуменюк П.О., Гуменюк Л.О., Лотиш В.В. Програмне забезпечення для керування фізичною моделлю автопоїзда. В роботі обґрунтовано необхідність розробки системи активної безпеки для автопоїздів. Описано програмне забезпечення, розроблене для керування фізичною моделлю автопоїзда з керованою причіпною ланкою. Наведено спосіб підключення базового ПЗ моделі до Windows-сумісного додатка для полегшення роботи з моделлю, зняття даних і запису їх для подальшої обробки.
Ключові слова: автопоїзд, модель, ПЗ

Гуменюк П.О., Гуменюк Л.О., Лотыш В.В. Программное обеспечение для управления физической моделью автопоезда. В работе обоснована необходимость разработки системы активной безопасности для автопоездов. Описано программное обеспечение, разработанное для управления физической моделью автопоезда с управляемым прицепным звеном. Приведен способ подключения базового ПО модели к Windows-совместимому приложению для облегчения работы с моделью, снятия данных и записи их для последующей обработки.
Ключевые слова: автопоезд, модель, ПО

Gumeniuk P.O., Gumeniuk L.O., Lotysh V.V. Software to control physical model of multilink vehicle. In the paper the necessity of development of active safety systems for multilink vehicles is given. The developed software to control the physical model of trailer truck with a guided attached link is described. A method to connect the basic model's software to a Windows-compatible application for data acquiring and recording it for further processing is shown.
Keywords: multilink vehicle, model, software

Постановка проблеми. Зі зростанням загальної кількості транспортних засобів на дорогах зростає необхідність використання інтелектуальних систем, які повинні допомагати водієві утримувати бажаний курс, запобігати виникненню надмірної чи недостатньої поворотності, знижувати ризик перевертання, тощо. Іншими словами, це є системи активної безпеки, які є асистивними, а у випадках, коли людської реакції недостатньо – приймають керування транспортним засобом. При розробці систем активної безпеки для транспортних засобів доцільно перевіряти їх на різного роду моделях перед тим, як впроваджувати на реальних об'єктах. У випадку некоректної роботи якогось із елементів в такому випадку можна уникнути значних фінансових збитків і не ризикувати здоров'ям водія. Зазвичай електронні системи мають наступну будову: основою є електронний обчислювальний пристрій (ЕОП), роль якого, залежно від бажаних функцій, може виконувати контролер, мікропроцесор, чіп чи інший подібний елемент. ЕОП збирає дані з датчиків (наприклад, датчик натиснення педалі чи ультразвуковий датчик, розміщений у задній площині транспортного засобу), обробляє отриману інформацію та посилає вихідний керуючий сигнал на потрібні виконавчі механізми (наприклад, головний гальмівний циліндр чи двигун). Таким чином покращується курсова стійкість, безпека при паркуванні чи інші характеристики [1]. Електронний обчислювальний пристрій 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 вцілому, навіть при успішному спрацюванні всіх фізичних виконавчих органів. Саме тому написання якісного програмного коду є не менш важливим питанням, як і вибір компонентів системи.

Мета роботи. Для розробки системи активного впливу на курсову стійкість автопоїздів було створено масштабовану фізичну модель автопоїзда з керованою причіпною ланкою (рис.1). Вона складається з тягача, на якому розміщено мікропроцесорну плату Arduino Uno і плату керування моторами, а також тривісного напівпричепа. Передня вісь тягача і задня вісь напівпричепа керовані засобами сервоприводів [2]. Вся модель приводиться у рух чотирма електромоторами, зусилля від яких передаються на другу і третю осі тягача. У точці зчеплення розміщено датчик кута повороту для визначення кута складання моделі.

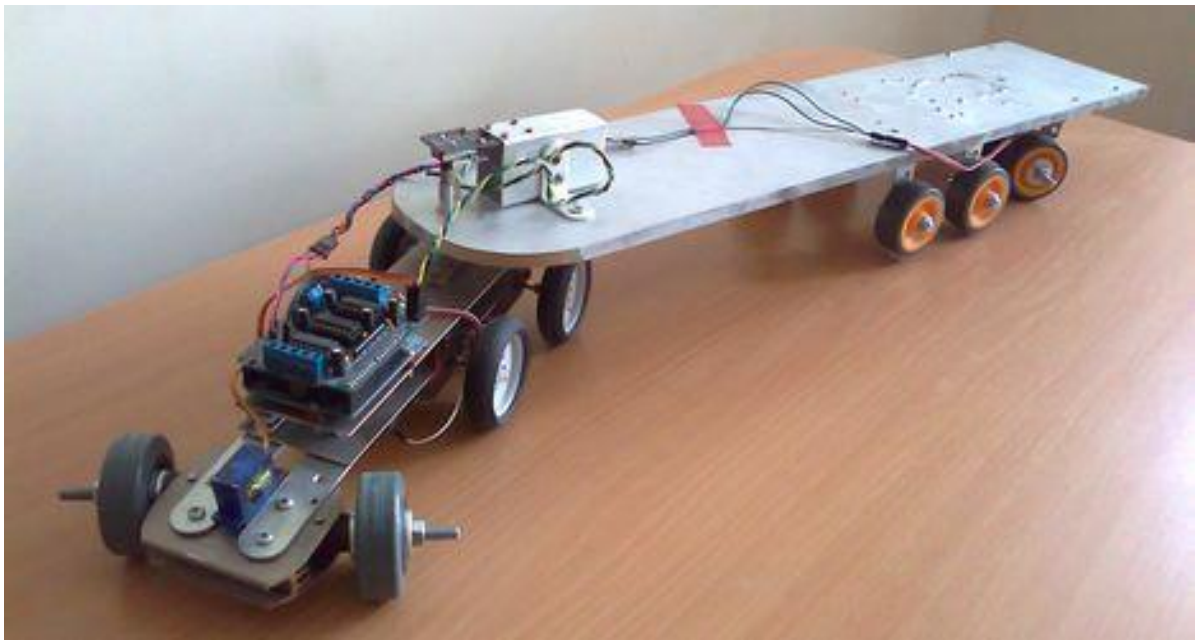

Рисунок 1. Модель автопоїзда з керованою причіпною ланкою.

Наступним завданням є написання програмного коду, який керуватиме моделлю для успішного виконання заданих маневрів, таких як поворот на 90^o, розворот, рух по колу чи зміна смуги руху [3]. Під час проходження вказаних маневрів траєкторія руху середньої точки керованої осі тягача повинна співпадати із траєкторією руху середньої точки керованої осі напівпричепа, у цьому випадку можна буде стверджувати, що поведінка моделі відповідає поведінці реального об'єкта.

Результати роботи. Дана модель керується за допомогою плати Arduino на базі мікропроцесора ATMega328P, що належить до сімейства процесорів AVR [4]. Середовищем розробки плат Arduino є багатоплатформовий Java-додаток, що заснований на мові Processing. Синтаксис даного середовища подібний до C++, але використовує деякі додаткові бібліотеки. Програми, створені для роботи з платою Arduino називаються скетчами [5]. Після успішної компіляції скетч передається у процесор засобами віртуального COM-порта. Вбудований у середовище розробки монітор дозволяє реалізувати зворотній зв'язок із платою у процесі виконання програми. Можлива передача команд процесору, а також зчитування і відображення даних (рис.1). Разом з тим, серійний монітор не має графічної оболонки чи механізму збереження відображуваних даних. Тому, для полегшення роботи з моделлю і нарощування її функціональності було розроблено Windows-сумісний додаток, написаний мовою Delphi. Застосовуючи даний додаток керувати моделлю можна натисканням на кнопку, замість введення команди із клавіатури, що зручніше і економить час користувача.

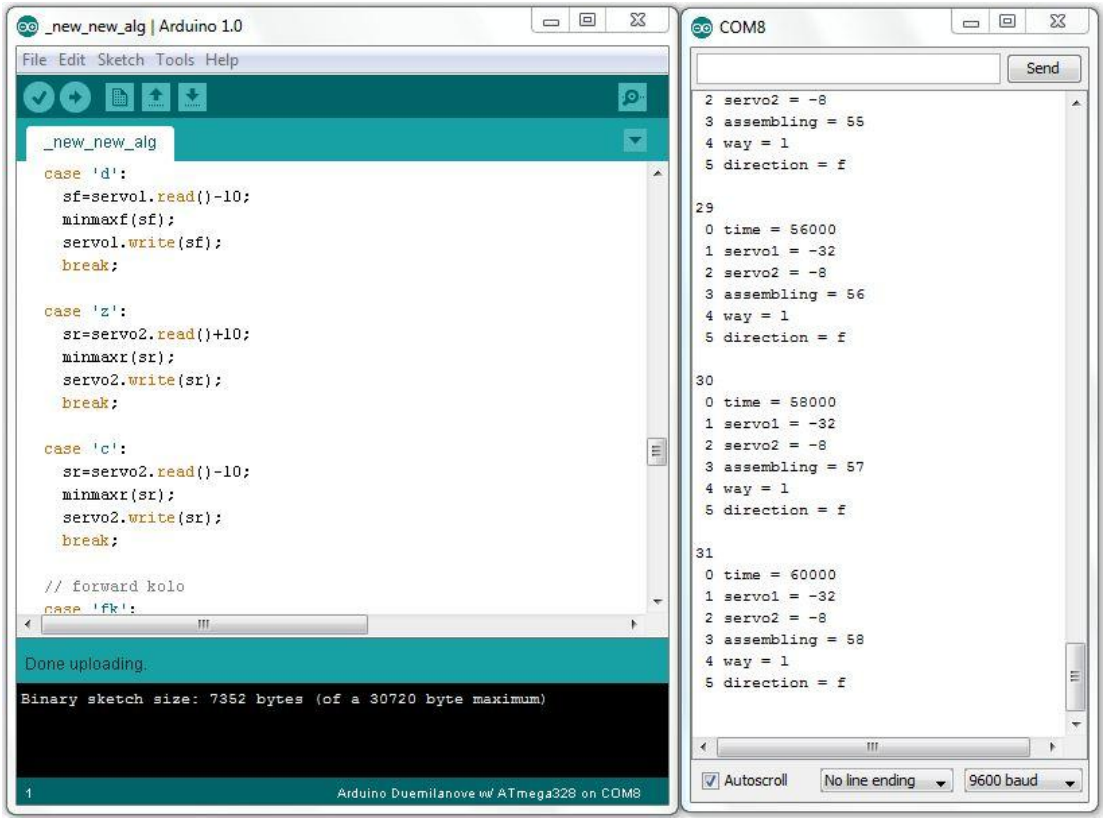

Рисунок 2. Середовище розробки скетчів Arduino (зліва) і серійний монітор (справа).

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 посилки, за яким ідуть біти даних і, можливо, біт паритету (парності). Завершує посилку стоп-біт, що гарантує паузу між посилками. 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 і натиснення на кнопку Open, якщо до вказаного порта дійсно підключено обладнання, відразу ж почнеться обмін даними із процесором. Команди на плату можна передавати як за допомогою текстового рядка, так і натискаючи на потрібні кнопки (рис.2). У даній реалізації додатку за допомогою кнопок можна визначити швидкість моделі, а також задати поворот для керованих коліс тягача і напівпричепа. Напрям руху задається групою вибору. Це зроблено по-перше, для кращої візуалізації, по-друге, через те, що активним може бути тільки один напрям: модель не може одночасно рухатись вперед і назад. Крім того, є дві кнопки для виконання наперед прописаних маневрів повороту на 90^{o} і зміни смуги руху. користувачеві потрібно тільки обрати напрям руху і натиснути на відповідну кнопку. Контролер автоматично встановить потрібні значення кутів повороту і буде їх змінювати так, щоб колії керованих коліс тягача і напівпричепа збігались.

Дані, отримані від процесора відображаються у вікні зліва, під рядком вводу команд і автоматично записуються у текстовий файл (рис.3) для можливості подальшої роботи з ними, в тому числі для обробки та аналізу. Після завершення роботи з додатком COM-порт потрібно закрити, для уникнення можливих помилок виконання програми.

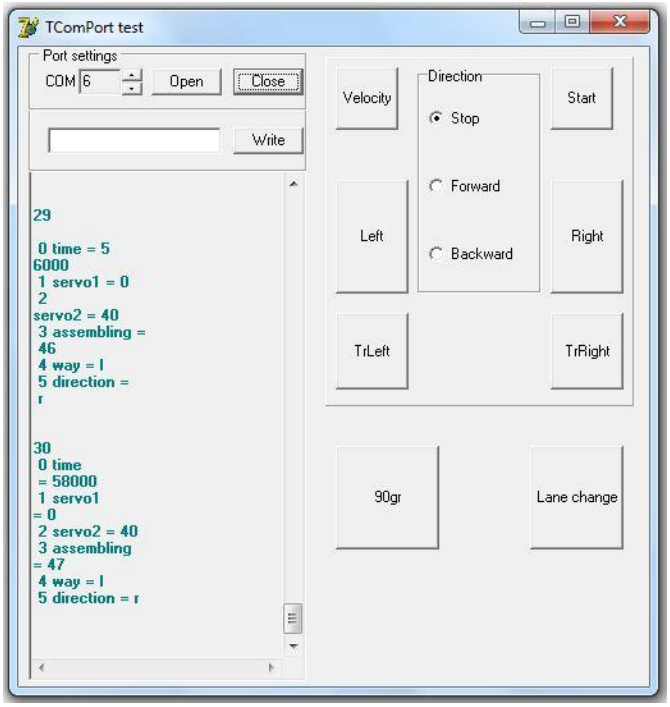

Рисунок 3. Windows-додаток для роботи з моделлю.

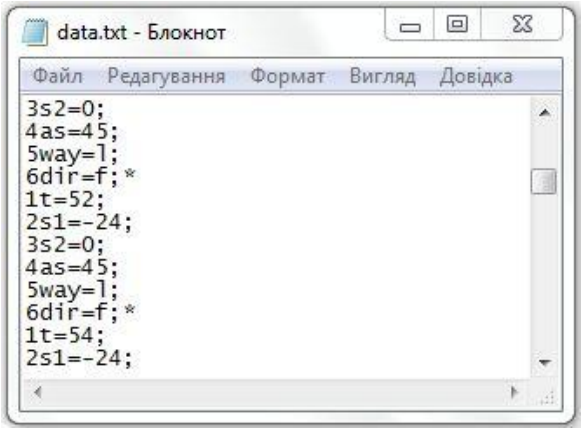

Рисунок 4. Дані, записані додатком.

Висновки. Представлене програмне забезпечення спрощує роботу з фізичною моделлю, полегшує обробку отриманих даних і дає змогу користувачеві як слідкувати за даними у ході експериментальних маневрів у режимі реального часу, так і вертатись до них пізніше.

Список використаних джерел.

1. Сахно В.П., Лотиш В.В., Гуменюк П.О. Покращення курсової стійкості автомобіля з використанням системи ESP // Управління проектами, системний аналіз і логістика: Наук. журнал. – К.: НТУ. – 2011. – Вип. 8.
2. Гуменюк П.О. Розробка масштабованої моделі автопоїзда / Гуменюк П.О. // Міжвузівський збірник «Наукові нотатки». – Луцьк, ЛНТУ, 2012. – Вип. 37. – С.68-69.
3. Стельмащук В.В., Лотиш В.В., Придюк В.М. До визначення показників маневреності автопоїзда-контейнеровоза // Матеріали міжнародної науково-технічної конференції «Автомобильный транспорт: проблемы и перспективы» Севастополь. – 2011. – Вип. 122.
4. Гуменюк П.О. Розробка масштабованої моделі сідельного тягача / Гуменюк П.О. // Матеріали XV Міжнародної науково-технічної конференції «Автомобільний транспорт: проблеми і перспективи» (м.Севастополь, 10-17 вересня 2012 року).
5. Brian W. Evans Arduino programming notebook. CreativeCommons, 1st edition. – 2007.
6. Delphi Developer's Guide. Sams Publishing. – 1999.
7. DIRECTIVE 2002/7/EC of European parliament and of the council of 18 February 2002 amending Council Directive 96/53/EC of 25 July 1996 laying down for certain road vehicles circulating within the Community the maximum authorized dimensions in national and international traffic and the maximum authorized weights in international traffic. // Official Journal of the European Communities. – 2002. – No L67/47 – 49.