

Одеський національний політехнічний університет

Кафедра радіотехнічних систем

КУРСОВИЙ ПРОЕКТ

з дисципліни «Цифрові пристрої»

на тему: «Пристрій керування радіомоделлю

на мікроконтролері АТМega8 »

Студента 4 курсу групи **РТ-091**

напряму підготовки **6.050901 - радіотехніка**

Марченко Іллі Валентиновича _____

Керівник к.т.н., доц. Цевух І. В. _____

Національна шкала _____

Кількість балів _____

Оцінка ECTS _____

Члени комісії

(підпис)

Зав. каф. РТС Баранов П.Ю.

(підпис)

доц. Цевух І.В.

(підпис)

доц. Мелешкевич О. М.

РЕФЕРАТ

У даному курсовому проекті розроблено модель автомобілю з дистанційним керуванням. Модель може приймає шість різних команд: вперед, назад, стоп, вліво, вправо та прямо. Передача керуючих сигналів здійснюється за технологією Bluetooth. Керуючим блоком є апаратна обчислювальна платформа Arduino на основі мікроконтролера ATmega8.

ЗМІСТ

| | |
|---|----|
| ВСТУП..... | 4 |
| 1 РОЗРОБКА ФУНКЦІОНАЛЬНОЇ СХЕМИ МІКРОПРОЦЕСОРНОЇ СИСТЕМИ..... | 5 |
| 2 РОЗРОБКА ПРИНЦИПОВОЇ СХЕМИ МІКРОПРОЦЕСОРНОЇ СИСТЕМИ..... | 7 |
| 2.1 Апаратна обчислювальна платформа Arduino..... | 7 |
| 2.1.1 Апаратна частина..... | 7 |
| 2.1.2 Програмне забезпечення..... | 9 |
| 2.2 Бездротовий Bluetooth модуль HC-04..... | 10 |
| 2.2.1 Апаратна частина..... | 11 |
| 2.2.2 Програмне забезпечення..... | 12 |
| 2.3 Драйвер двигуна L293DD..... | 12 |
| 2.3.1 Принцип роботи..... | 12 |
| 2.3.2 Характеристики мікросхеми L293DD..... | 15 |
| 2.4 Розробка принципової схеми..... | 16 |
| 3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ..... | 17 |
| 3.1 Розробка словесного алгоритму..... | 17 |
| 3.2 Розробка блок-схеми..... | 17 |
| 3.3 Розробка коду програми..... | 18 |
| ВИСНОВКИ..... | 20 |
| ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ..... | 21 |
| ДОДАТОК А..... | 22 |
| ДОДАТОК Б..... | 24 |

| | | | | | | | |
|-----------|------|-----------------|--------|------|---|-------------------|------|
| | | | | | РТ91.000000.011ПЗ | | |
| Змн. | Арк. | № докум. | Підпис | Дата | | | |
| Розроб. | | Марченко І. В. | | | Радіокерована модель. Пояснювальна записка | Літ. | Арк. |
| Перевір. | | Цевух І.В. | | | | | 3 |
| Реценз. | | | | | | Акрушів | 28 |
| Н. Контр. | | Мелешкевич О.М. | | | | ОНПУ, ІРТ, РТ-091 | |
| Затверд. | | | | | | | |

ВСТУП

Основна мета курсового проекту – отримання практичних навичок в розробці апаратних та програмних засобів мікропроцесорної системи.

Радіокерована модель – це модель автомобіля з радіо керуванням. Застосування такої моделі може бути доволі різним – від іграшки і до використання в саперній техніці. В даний час розробити таку модель достатньо просто, маючи базові знання у програмуванні мікроконтролерів і схемотехніці.

У даному курсовому проекті потрібно розробити модель автомобілю з дистанційним керуванням. Передача керуючих сигналів має здійснюватися за технологією Bluetooth. Керуючим блоком має бути апаратна обчислювальна платформа Arduino.

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | РТ91.000000.011ПЗ | Лист |
| | | | | | | 4 |
| Изм. | Лист | № документа | Подпись | Дата | | |

1 РОЗРОБКА ФУНКЦІОНАЛЬНОЇ СХЕМИ МІКРОПРОЦЕСОРНОЇ СИСТЕМИ

Функціональна схема радіокерованої моделі зображена на рисунку 1.1:

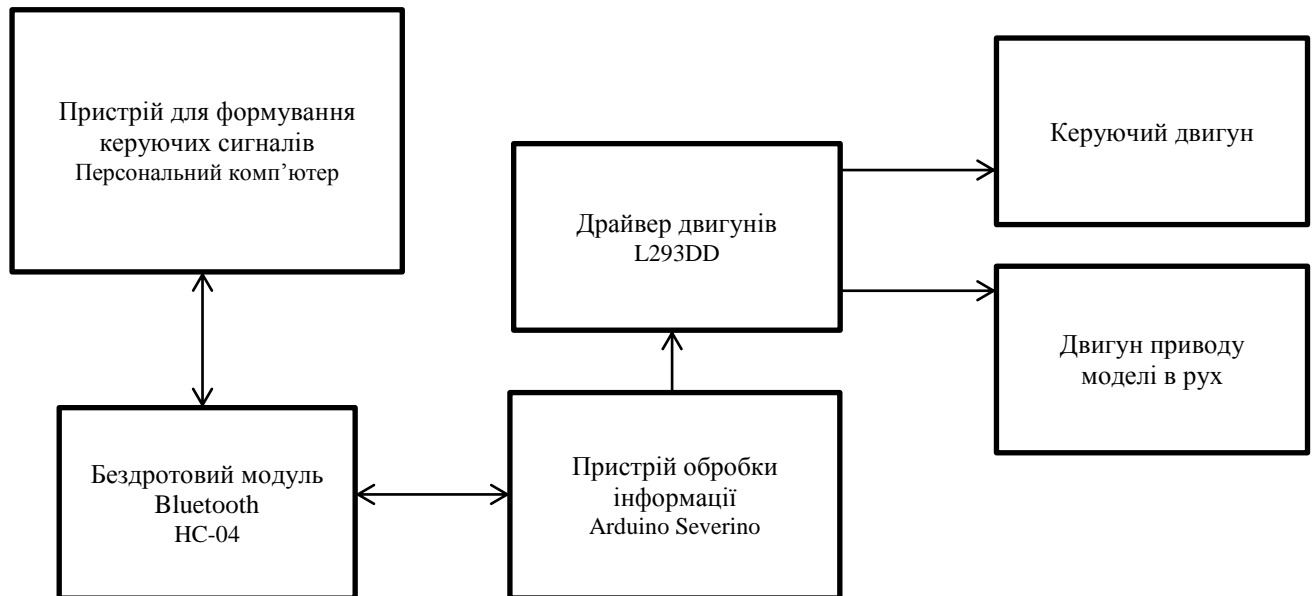


Рисунок 1.1 – Функціональна схема радіокерованої моделі

Дана функціональна схема включає в себе:

1. Апаратна обчислювальна платформа Severino фірми Arduino з мікроконтролером ATmega8 фірми Atmel використовується для обробки керуючих команд та генерування відповідних цифрових сигналів на вході блоку керування двигунами.
2. Бездротовий модуль прийому та передачі даних у форматі RS232 за технологією Bluetooth HC-04 фірми CSR використовується для прийому керуючих команд та передачі підтвердження отримання команди.
3. Драйвер двигуна L293DD фірми STMicroelectronics використовується для керування двигунами радіокерованої моделі.

4. Два двигуни для приводу в рух радіокерованої моделі. Один двигун керує поворотом передніх рульових коліс, другий двигун керує рухом моделі вперед та назад.

5. Пристрій для формування керуючих сигналів. Наприклад ПК з Bluetooth адаптером.

При передачі однієї з керуючих команд радіокерована модель приймає і виконує цю команду. Після прийому керуючим блоком команд «вперед» або «назад» задня пара коліс обертається відповідно вперед або назад до тих пір, поки керуючий блок не отримає команду «стоп». Після прийому керуючим блоком команд «вліво» або «вправо» відбувається поворот передньої пари коліс відповідно вліво або вправо. Після прийому команди «центр» відбувається вирівнювання передньої пари коліс.

Після прийому кожної команди модель передає підтвердження прийому команди на керуючий пристрій.

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | РТ91.000000.011ПЗ | Лист |
| | | | | | | 6 |
| Изм. | Лист | № документа | Подпись | Дата | | |

2 РОЗРОБКА ПРИНЦИПОВОЇ СХЕМИ МІКРОПРОЦЕСОРНОЇ СИСТЕМИ

2.1 Апаратна обчислювальна платформа Arduino

Arduino — апаратна обчислювальна платформа, основними компонентами якої є проста плата введення / виводу і середовище розробки на мові Processing / Wiring. Arduino може використовуватися як для створення автономних інтерактивних об'єктів, так і підключатися до програмного забезпечення, виконуваному на комп'ютері (наприклад, Adobe Flash, Processing, Max / MSP, Pure Data, SuperCollider). Інформація про пристрій плати (малюнок друкованої плати) знаходиться у відкритому доступі і може бути використана тими, хто воліє збирати плати самостійно.

2.1.1 Апаратна частина

Плата Arduino складається з мікроконтролера ATmega8, а також елементів обв'язки для програмування та інтеграції з іншими схемами. На платах присутній лінійний стабілізатор напруги +5 В. Тактування здійснюється на частоті 16 МГц кварцовим резонатором. У мікроконтролер попередньо прошивається завантажувач BootLoader, тому зовнішній програматор не потрібен.

На концептуальному рівні всі плати програмуються через RS-232 (послідовне з'єднання), але реалізація цього способу відрізняється від версії до версії. Плата Arduino Severino, яка використовується в даному курсовому проекті, містить просту інвертуючу схему для конвертації рівнів сигналів RS-232 в рівні TTL, і навпаки. У продажу наявні плати, наприклад, Diecimila, що програмуються через USB, що здійснюється завдяки мікросхемі конвертера USB-to-Serial FTDI FT232R. У версії платформи Arduino Uno в якості конвертера використовується мікроконтролер Atmega8 в SMD-корпусі. Дане рішення дозволяє програмувати

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | PT91.000000.011ПЗ | Лист |
| Изм. | Лист | № документа | Подпись | Дата | | 7 |

конвертер так, щоб платформа відразу визначалася як миша, джойстик або інший пристрій на розсуд розробника з усіма необхідними додатковими сигналами управління. У деяких варіантах, таких як Arduino Mini або неофіційної Boarduino, для програмування потрібне підключення окремої плати USB-to-Serial або кабелю.

Плати Arduino дозволяють використовувати більшу частину I / O виводів мікроконтролера в зовнішніх схемах. Наприклад, в платі Severino доступно 14 цифрових входів / виходів, 3 з яких можуть видавати ШИМ сигнал, і 6 аналогових входів. Ці сигнали доступні на платі через штирові роз'єми. Також доступні декілька видів зовнішніх плат розширення, званих «shields» («щити»), які приєднуються до плати Arduino через штирові роз'єми.

Зовнішній вигляд плати Arduino та розташування роз'ємів наведений на рисунку 2.1.

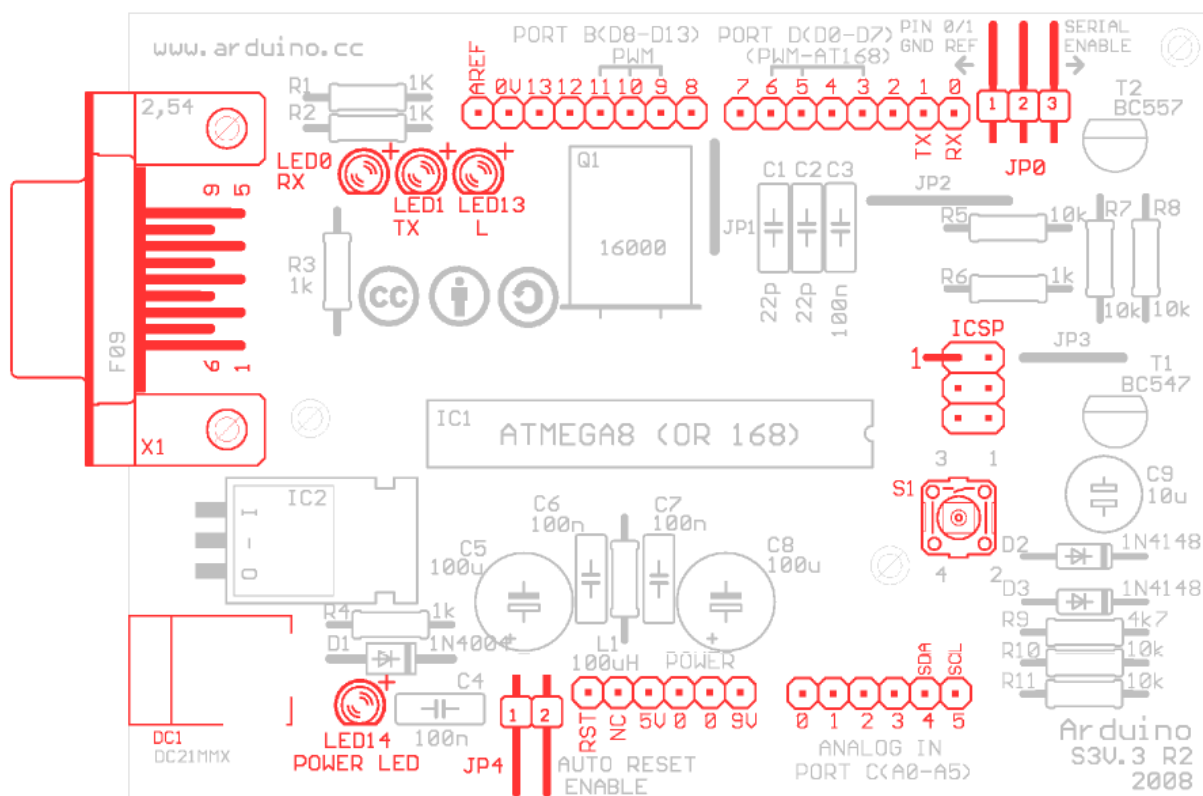


Рисунок 2.1 – Розташування роз'ємів на платі Arduino

Для підключення комп'ютера (або іншого пристрою) за допомогою RS - 232 стандарту використовується DE-9 послідовний роз'єм X1. Для підключення зовнішнього джерела живлення використовується роз'єм живлення DC1.

Для програмування завантажувача в Atmega8 використовується штировий роз'єм ICSP. Для скидання Atmega8 використовується тактильна кнопка S1.

2.1.2 Програмне забезпечення

Інтегроване середовище розробки Arduino це кроссплатформений додаток на Java, що включає в себе редактор коду (рис.2.2), компілятор і модуль передачі прошивки в плату.

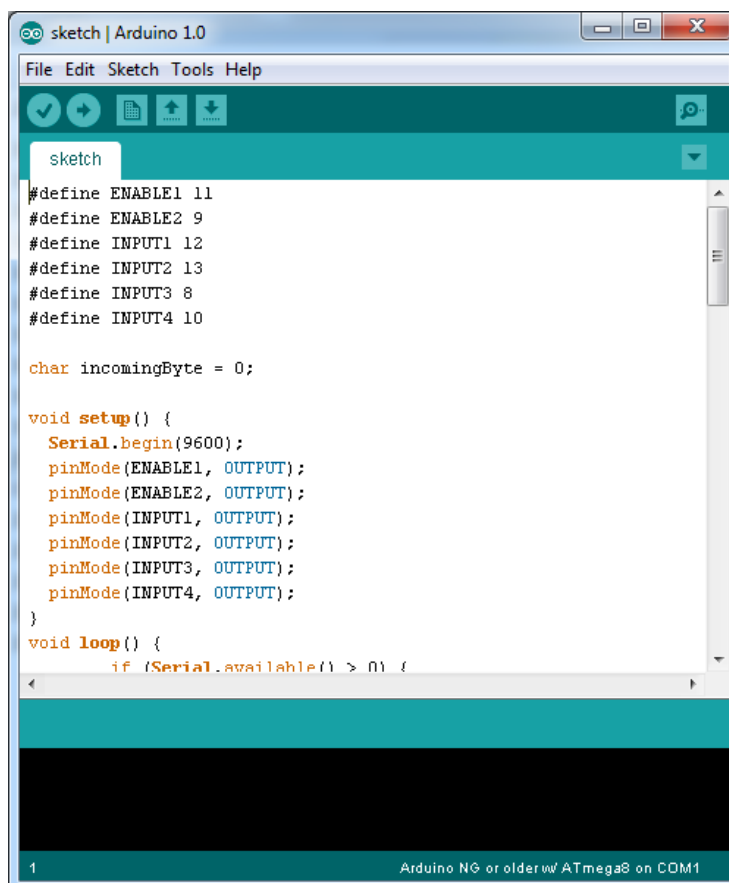


Рисунок 2.2 – Редактор коду Arduino 1.0

Середовище розробки базується на мові програмування Processing та спроектована для програмування новачками, не знайомими близько з розробкою програмного забезпечення. Мова програмування, м'яко кажучи, це C++, доповнений деякими бібліотеками. Програми обробляються за допомогою препроцесора, а потім компілюються за допомогою AVR-GCC.

2.2 Бездротовий Bluetooth модуль HC-04

Бездротовий Bluetooth модуль HC-04 виконаний у вигляді готової плати з винесеними 34 контактами та контактом для підключення зовнішньої антени. Зовнішній вигляд пристрою приведений на рисунку 2.3:

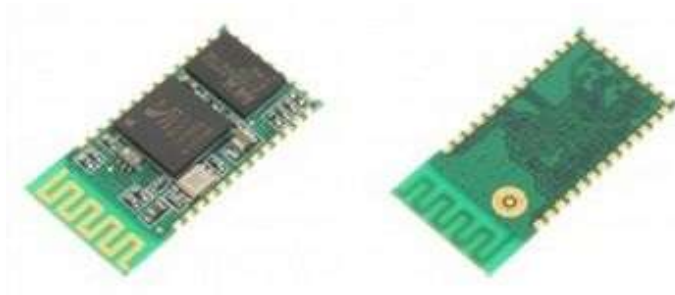


Рисунок 2.3 – Зовнішній вигляд Bluetooth модуля HC-04

За замовчуванням в пам'ять FLASH плати HC-04 записано ПЗ, котре дозволяє зв'язати по радіо Bluetooth любим пристроєм (телефон, ноутбук и т. п.) з вбудованої системою на основі мікроконтролера (робот, плата Arduino, будь-який пристрій на мікроконтролері, що має TTL-порт UART RS-232). За допомогою пакета CSR CASIRA BLUELAB SDK (в якому є робочі приклади програм Bluetooth) можна самому перепрограмувати модуль HC-04 та створювати свої власні пристрої Bluetooth. Програматор і повноцінний апаратний відладчик для модуля можна легко зробити самому, підключається до комп'ютера він через порт

LPT. Завдяки своїм малим розмірам та низькій ціні модуль HC-04 вже цікавий як зручний готовий пристрій для бездротового зв'язку.

2.2.1 Апаратна частина

Модуль має чіп пам'яті на 1 мегабайт. Там записано керуюче програмне забезпечення та всі налаштування. На зовнішні 34 контакти модуля винесені:

1. апаратний UART, сигнали TXD, RXD, CTS и RTS;
2. послідовний порт PCM (для цифрового вводу/виводу звуку) ;
3. два аналогових входів/виводів АІО;
4. пін скидання RESET;
5. вхід напруги живлення +3.3 В, максимальний струм 35 мА;
6. інтерфейс USB;
7. інтерфейс SPI, через який завантажується програмне;
8. 12 цифрових портів вводу/виводу РІО.

Розташування пінів Bluetooth модуля HC-04 зображена на рисунку 2.4:

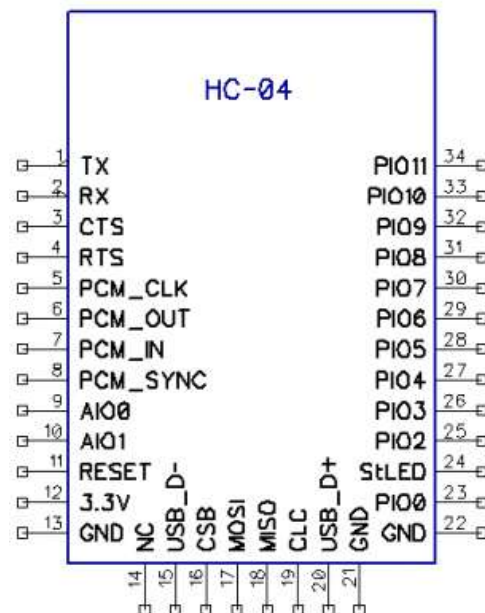


Рисунок 2.4 – Розташування пінів Bluetooth модуля HC-04

2.2.2 Програмне забезпечення

Після подачі живлення на модуль його можна знайти як бездротове Bluetooth-пристрій з профілем послідовного COM-порту, через який можна безпосередньо обмінюватися даними через TTL-сигнали TX і RX стандартного порту RS-232. Програмне забезпечення модулю HC-04 дозволяє AT-командами міняти швидкість передачі даних в широких межах (від 1200 до 1382400 бод), назву для ідентифікації та пароль для підключення. Зміни налаштування енергонезалежні, тобто зберігаються між виключеннями живлення.

2.3 Драйвер двигуна L293DD

Для керування двигунами необхіден пристрій, який би перетворював керуючі сигнали малої потужності в струми, достатні для управління двигунами. Такий пристрій називають драйвером двигунів.

Існує досить багато самих різних схем для керування електродвигунами. Вони розрізняються як потужністю, так і елементною базою, на основі якої вони виконані.

Ми зупинимося на самому простому драйвері управління двигунами, виконаному у вигляді повністю готової до роботи мікросхеми. Ця мікросхема називається L293D і є однією з найпоширеніших мікросхем, призначених для цієї мети.

2.3.1 Принцип роботи

Драйвер двигуна L293DD містить відразу два драйвера для управління електродвигунами невеликої потужності (чотири незалежні канали, об'єднаних в дві пари). Має дві пари входів для керуючих сигналів і дві пари виходів для підключення електромоторів. Крім того, у Драйвер двигуна L293DD є два входи

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | РТ91.000000.011ПЗ | Лист |
| Изм. | Лист | № документа | Подпись | Дата | | 12 |

для включення кожного із драйверів. Ці входи використовуються для управління швидкістю обертання електромоторів за допомогою широтно модульованого сигналу (ШІМ).

Драйвер двигуна L293DD забезпечує поділ електроживлення для мікросхеми і для керованих нею двигунів, що дозволяє підключити електродвигуни з більшою напругою живлення, ніж у мікросхеми. Поділ електроживлення мікросхем і електродвигунів може бути також потрібен для зменшення перешкод, викликаних кидками напруги, пов'язаними з роботою моторів.

Принцип роботи кожного із драйверів, що входять до складу мікросхеми, ідентичний, тому розглянемо принцип роботи одного з них.

До виходів OUTPUT1 і OUTPUT2 підключимо електромотор MOTOR1 (рис. 2.5).

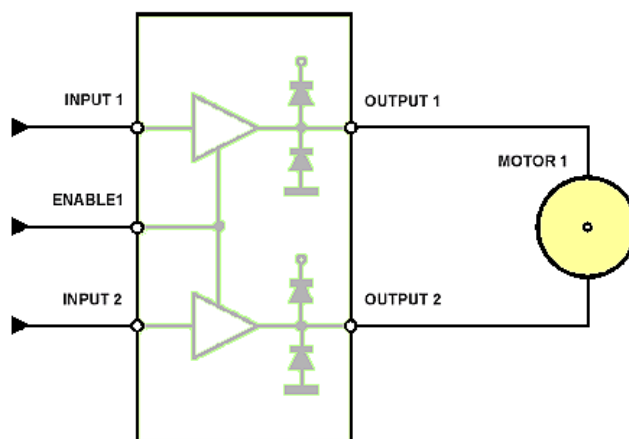


Рисунок 2.5 – Підключення електромотору до драйвера двигуна L293D

На вхід ENABLE1, що включає драйвер, подамо сигнал (з'єднаємо з позитивним полюсом джерела живлення +5 В). Якщо при цьому на входи INPUT1 і INPUT2 не подаються сигнали, то мотор обертатися не буде.

Якщо вхід INPUT1 з'єднати з позитивним полюсом джерела живлення, а вхід INPUT2 - з негативним, то мотор почне обертатися.

Тепер спробуємо з'єднати вхід INPUT1 з негативним полюсом джерела живлення, а вхід INPUT2 - з позитивним. Мотор почне обертатися в інший бік.

Спробуємо подати сигнали одного рівня відразу на обидва керуючих входу INPUT1 і INPUT2 (з'єднати обидва входи з позитивним полюсом джерела живлення або з негативним) - мотор обертатися не буде.

Якщо ми приберемо сигнал з входу ENABLE1, то при будь-яких варіантах наявності сигналів на входах INPUT1 і INPUT2 мотор обертатися не буде.

Уявити краще принцип роботи драйвера двигуна можна, розглянувши наступну таблицю 2.1:

Таблиця 2.1 – Таблиця станів входів та виходів мікросхеми L293DD

| ENABLE1 | INPUT1 | INPUT2 | OUTPUT1 | OUTPUT2 |
|---------|--------|--------|---------|---------|
| 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Тепер розглянемо призначення виводів мікросхеми L293DD (рис 2.6).

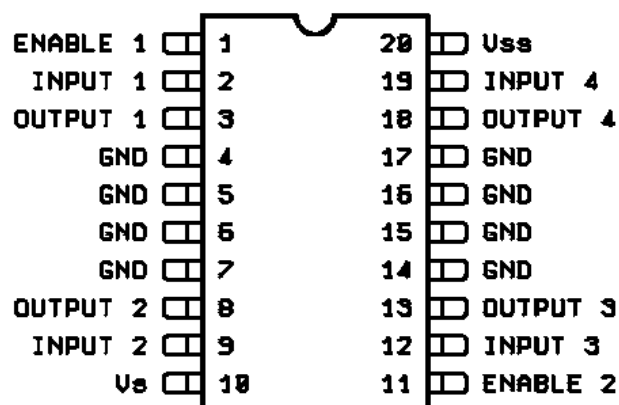


Рисунок 2.6 – Розташування виводів мікросхеми L293DD

Призначення виводів мікросхеми:

1. Входи ENABLE1 і ENABLE2 відповідають за включення кожного із драйверів, що входять до складу мікросхеми.
2. Входи INPUT1 і INPUT2 управляють двигуном, підключеним до виходів OUTPUT1 і OUTPUT2.
3. Входи INPUT3 і INPUT4 управляють двигуном, підключеним до виходів OUTPUT3 і OUTPUT4.
4. Контакт Vs з'єднують з позитивним полюсом джерела електроживлення двигунів або просто з позитивним полюсом живлення, коли живлення схеми і двигунів єдине. Простіше кажучи, цей контакт відповідає за живлення електродвигунів.
5. Контакт Vss з'єднують з позитивним полюсом джерела живлення. Цей контакт забезпечує живлення самої мікросхеми.
6. Чотири контакти GND з'єднують з "землею" (загальним проводом або негативним полюсом джерела живлення). Крім того, за допомогою цих контактів зазвичай забезпечують тепловідвід від мікросхеми, тому їх найкраще розпаювати на досить широку контактну площадку.

2.3.2 Характеристики мікросхеми L293DD

Мікросхема L293DD має наступні технічні характеристики:

1. напруга живлення двигунів (Vs) - 4,5 ... 36В;
2. напруга живлення мікросхеми (Vss) – 5В;
3. допустимий струм навантаження – 600мА (на кожен канал);
4. піковий (максимальний) струм на виході - 1,2 А (на кожен канал);
5. логічний "0" вхідної напруги - до 1,5 В;
6. логічна "1" вхідної напруги - 2,3 ... 7В;
7. швидкість перемикачів до 5 кГц;
8. захист від перегріву.

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | РТ91.000000.011ПЗ | Лист |
| Изм. | Лист | № документа | Подпись | Дата | | 15 |

2.4 Розробка принципової схеми

З основними елементами радіокерованої моделі ми визначилися у попередньому розділі.

Як видно зі схеми на рисунку (PT91.000000.011E3) мікроконтролер ATmega8, позначений U1, є керуючим блоком, у який завантажена програма керування радіокерованою моделлю. До 20-го піну підводиться живлення 5 В.

До пінів XTAL1 та XTAL2 підключаємо кварцевий резонатор X1 на 16 МГц для стабілізації частоти.

Конденсатори C1 та C2 використовуються для виділення гармонічної складової коливання кварцового резонатору.

Мікросхема U2 є драйвером двигунів L293DD і використовуються для керування двигунами. На пін Vss подається стабілізована напруга 5В. На пін Vs подається напруга 12В.

Світлодіоди D1, D2, D3 та D4 використовуються для світлової індикації полярності напруги, поданої на двигуни. Резистори R1 та R2 використовуються для обмеження струму світлодіодів.

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | PT91.000000.011ПЗ | Лист |
| | | | | | | 16 |
| Изм. | Лист | № документа | Подпись | Дата | | |

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Розробка словесного алгоритму

Після подачі живлення модель очікує на команду. При передачі однієї з керуючих команд радіокерована модель приймає і виконує цю команду. Після прийому керуючим блоком команд «вперед» або «назад» задня пара коліс обертається відповідно вперед або назад до тих пір, поки модель не отримає команду «стоп». Після прийому керуючим блоком команд «вліво» або «вправо» відбувається поворот передньої пари коліс відповідно вліво або вправо. Після прийому команди «центр» відбувається вирівнювання передньої пари коліс.

Після прийому кожної команди модель передає підтвердження прийому команди на керуючий пристрій.

3.2 Розробка блок-схеми

На основі сформованого словесного алгоритму розробимо блок-схему програми (РТ91.000000.011Д1). Розроблена блок-схема потребує детального пояснення.

Перший блок «Початок» вказує на початок програми.

В блоці «Ініціалізація портів та УАПП» виконується налаштування вказаних пінів Arduino як виходів та налаштування УАПП для прийому та передачі даних зі швидкістю 9600 бод/с.

В блоці «Отримати команду по УАПП» виконується прийом байту по УАПП та занесення його до пам'яті.

Далі за допомогою умовних блоків виконується вибір та виконання відповідної прийнятому байту команди.

Після цього виконується перехід на початок програми.

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | РТ91.000000.011ПЗ | Лист |
| Изм. | Лист | № документа | Подпись | Дата | | 17 |

3.3 Розробка коду програми

На основі сформованої розробленої блок-схеми, пишемо програму (додаток А), що забезпечує функціонування автомобіля с дистанційним керуванням. Розроблений код програми потребує детального пояснення.

Код програми починається з визначення констант: ENABLE1, ENABLE2, INPUT1, INPUT2, INPUT3, INPUT4. Данні константи означають відповідність цифрових виходів Arduino та цифрових входів драйверу L293DD. Отже якщо підключити один з входів драйверу двигунів до іншого виходу Arduino, достатньо змінити значення відповідної константи.

Код програми, що залишився складається з восьми процедур. Вони потребують детального пояснення.

Процедура «setup» виконує ініціалізацію Arduino: налаштовує УАПП для прийому та передачі даних зі швидкістю 9600 бод/с, налаштовує піни, визначені константами, на вихід. Дана процедура викликається один раз перед стартом основної програми.

Процедура «loop» виконує основну програму. За допомогою умовного оператора if перевіряється наявність даних для прийому через УАПП. Якщо данні наявні, то виконується прийом байту і занесення його у змінну «incomingByte» і відправлення даного байту для підтвердження. Наступним кроком за допомогою оператора вибору «switch-case» виконується перехід на відповідну прийнятому байту процедуру. Якщо байт не є відомою командою то через УАПП відправляється строчка « error ».

Наступні процедури виставляють відповідні прийнятій команді логічні рівні на вході драйвера двигунів, а саме:

1. Процедура «forward» викликається, коли потрібно щоб модель рухалась вперед.
2. Процедура «backward» викликається, коли потрібно щоб модель рухалась назад.

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | PT91.000000.011ПЗ | Лист |
| Изм. | Лист | № документа | Подпись | Дата | | 18 |

3. Процедура «neutrally» викликається для зупинки руху моделі.
4. Процедура «left» викликається для повороту моделі вліво.
5. Процедура «right» викликається для повороту моделі вправо.
6. Процедура «center» викликається для вирівнювання передніх коліс моделі по центру.

| | | | | | | |
|------|------|-------------|---------|------|--------------------------|------|
| | | | | | <i>PT91.000000.011ПЗ</i> | Лист |
| | | | | | | 19 |
| Изм. | Лист | № документа | Подпись | Дата | | |

ВИСНОВКИ

В результаті виконання курсового проекту була розроблена радіокерована модель автомобіля.

В розділі 1 була розроблена функціональна схема і були сформульовані основні вимоги для реалізованого пристрою.

В розділі 2 була розроблена принципова схема мікропроцесорної системи і були визначені необхідні для реалізації компоненти.

В розділі 3 було розроблено програмне забезпечення для керування пристроєм. Програмний код був написаний на мові програмування Arduino.

Пристрій функціонує без ускладнень і помилок, і може бути використаний у побуті.

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | РТ91.000000.011ПЗ | Лист |
| | | | | | | 20 |
| Изм. | Лист | № документа | Подпись | Дата | | |

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мортон Дж. Микроконтроллеры AVR. Вводный курс. / Пер. с англ.–М.: Издательский дом «Додэка- XXI», 2006. – 272 с.: ил. (Серия «Мировая электроника»).
2. Евстифеев А. В. Микроконтроллеры AVR семейства Mega. Руководство пользователя. – М.: Издательский дом «Додэка-XXI», 2007. – 592 с.: ил. (Серия «Программируемые системы»).
3. Arduino - ArduinoBoardSerialSingleSided3 [Электронный ресурс] – Режим доступа: <http://www.arduino.cc/en/Main/ArduinoBoardSerialSingleSided3>
4. Arduino - Language Reference [Электронный ресурс] – Режим доступа: <http://arduino.cc/en/Reference/HomePage>
5. Brian W. Evans Arduino блокнот программиста – 2007. [Электронный ресурс] – Режим доступа: http://mkpochtoi.narod.ru/arduino_book_rus.pdf

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | РТ91.000000.011ПЗ | Лист |
| Изм. | Лист | № документа | Подпись | Дата | | 21 |

ДОДАТОК А

```
/*
 * Програма для дистанційного керування моделі автомобіля
 * @author Марченко Ілля
 */

// наступні константи задають відповідність між пінами Arduino та пінами L293DD
#define ENABLE1 11
#define ENABLE2 9
#define INPUT1 12
#define INPUT2 13
#define INPUT3 8
#define INPUT4 10
// змінна буде зберігати прийнятий по УАПП байт
char incomingByte = 0;
// данна функція викликається після подачі живлення або скидання програми
void setup() {
    Serial.begin(9600); // ініціалізація УАПП
    pinMode(ENABLE1, OUTPUT); // налаштувати піни як виходи
    pinMode(ENABLE2, OUTPUT);
    pinMode(INPUT1, OUTPUT);
    pinMode(INPUT2, OUTPUT);
    pinMode(INPUT3, OUTPUT);
    pinMode(INPUT4, OUTPUT);
}
// наступна функція виконується нескінченно по колу
void loop() {
    if (Serial.available() > 0) { // якщо є доступна команда
        incomingByte = Serial.read(); // занесемо її у змінну
        Serial.print(incomingByte); // відправимо її назад
        switch (incomingByte) { // та виконаємо її
            case 119: forward();
                break;
            case 115: backward();
                break;
            case 32: neutrally();
                break;
            case 97: left();
                break;
            case 100: right();
                break;
            case 113: center();
                break;
        }
    }
}
```

| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | РТ91.000000.011ПЗ | Лист |
| | | | | | | 22 |
| Изм. | Лист | № документа | Подпись | Дата | | |

```

        default: Serial.print(" error "); // якщо команда не опізнана
                break;
    }
}

/*
 * Наступні функції виставляють відповідні логічні рівні на пінах драйверу
 * двигунів для приводу керованої моделі в рух
 */
// ідемо вперед
void forward() {
    digitalWrite(ENABLE2, LOW);
    digitalWrite(INPUT3, HIGH);
    digitalWrite(INPUT4, LOW);
    digitalWrite(ENABLE2, HIGH);
}
// ідемо назад
void backward() {
    digitalWrite(ENABLE2, LOW);
    digitalWrite(INPUT3, LOW);
    digitalWrite(INPUT4, HIGH);
    digitalWrite(ENABLE2, HIGH);
}
// не ідемо
void neutrally() {
    digitalWrite(ENABLE2, LOW);
}
// повертаємо вліво
void left() {
    digitalWrite(ENABLE1, LOW);
    digitalWrite(INPUT1, HIGH);
    digitalWrite(INPUT2, LOW);
    digitalWrite(ENABLE1, HIGH);
}
// повертаємо вправо
void right() {
    digitalWrite(ENABLE1, LOW);
    digitalWrite(INPUT1, LOW);
    digitalWrite(INPUT2, HIGH);
    digitalWrite(ENABLE1, HIGH);
}
// не повертаємо
void center() {
    digitalWrite(ENABLE1, LOW);
}

```

| | | | | | | |
|------|------|-------------|---------|------|--------------------------|------|
| | | | | | PT91.000000.011ПЗ | Лист |
| | | | | | | 23 |
| Изм. | Лист | № документа | Подпись | Дата | | |

ДОДАТОК Б

ВІДОМІСТЬ ЕЛЕКТРОННИХ ДОКУМЕНТІВ

| Поз. | Позначення | Найменування | Номер аркуша |
|------|--------------------|-----------------------------|--------------|
| 1 | РТ91.000000.011 Е2 | Схема електрична структурна | 1 |
| 2 | РТ91.000000.011 Е3 | Схема електрична принципова | 2 |
| 3 | РТ91.000000.011 Д1 | Блок-схема програми МК | 3 |
| 4 | РТ91.000000.011 ПЗ | Текст програми МК | 4 |

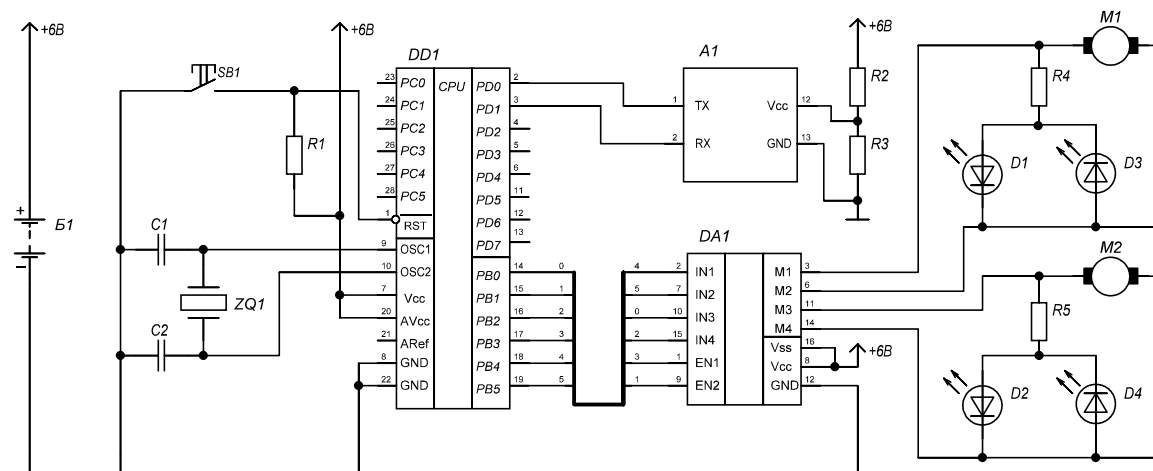
Склав _____ Марченко І. В.
(підпис)

Перевірів _____ Мелешкевич О. М.
(підпис)

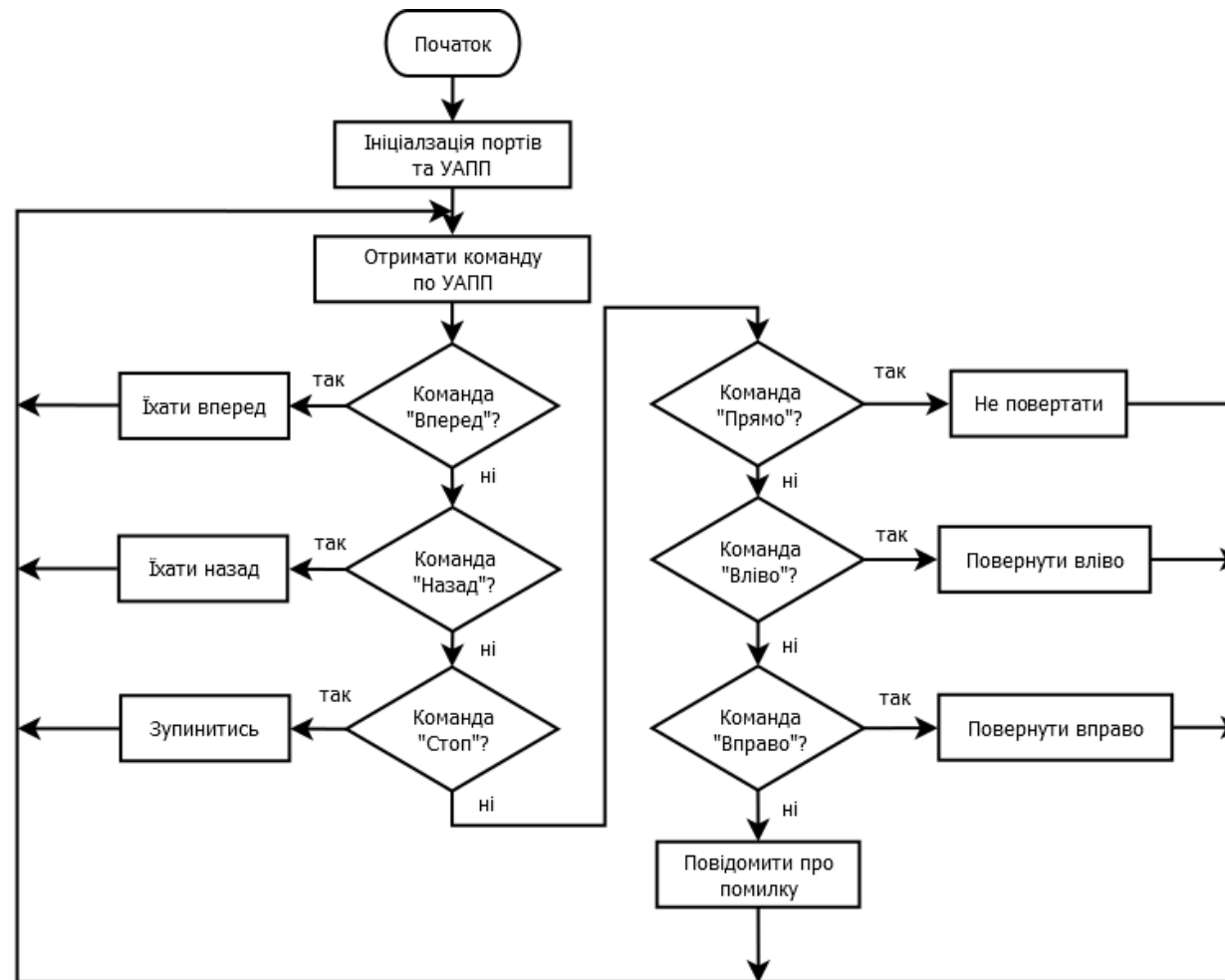
| | | | | | | |
|------|------|-------------|---------|------|-------------------|------|
| | | | | | РТ91.000000.011ПЗ | Лист |
| Изм. | Лист | № документа | Подпись | Дата | | 24 |



| | | | | | | | | | | | | | | | |
|-----------|------|-----------------|--------|------|--|-----|-------------------|--|------|--------|--|---------|--|--|--|
| | | | | | PT91.000000.011E2 | | | | | | | | | | |
| | | | | | Радіокерована модель. Структурна схема | Лім | | | Маса | | | Масштаб | | | |
| Змн. | Лист | № докум. | Підпис | Дата | | | | | | | | | | | |
| Розроб. | | Марченко І. В. | | | | | | | | | | | | | |
| Перевір. | | Цевух І.В. | | | | | | | | | | | | | |
| Т. контр. | | | | | | | | | | | | | | | |
| Реценз. | | | | | | | | | | | | | | | |
| Н. контр. | | Мелешкевич О.М. | | | | | | | | | | | | | |
| Затверд. | | | | | | | | | | | | | | | |
| | | | | | | | Лист 25 | | | Листів | | | | | |
| | | | | | | | ОНПУ, ІРТ, РТ-091 | | | | | | | | |



| | | | | | | | | | | | |
|-----------|------|-----------------|--------|------|---|-------------------|--|--------|------|---------|--|
| | | | | | PT91.000000.011E3 | | | | | | |
| | | | | | Радіокерована модель. Схема електрична принципова | Лім | | | Маса | Масштаб | |
| Змн. | Лист | № докум. | Підпис | Дата | | | | | | | |
| Розроб. | | Марченко І. В. | | | | | | | | | |
| Перевір. | | Цевух І.В. | | | | | | | | | |
| Т. контр. | | | | | | | | | | | |
| Реценз. | | | | | | | | | | | |
| Н. контр. | | Мелешкевич О.М. | | | | | | | | | |
| Затверд. | | | | | | | | | | | |
| | | | | | | Лист 26 | | Листів | | | |
| | | | | | | ОНПУ, ІРТ, РТ-091 | | | | | |



| | | | | | | | | | | |
|-----------|------|-----------------|--------|------|---|-------------------|--|--------|------|---------|
| | | | | | РТ91.000000.011Д1 | | | | | |
| | | | | | Радіокерована модель. Блок-схема | Лім | | | Маса | Масштаб |
| | | | | | | | | | | |
| Змн. | Лист | № докум. | Підпис | Дата | | | | | | |
| Розроб. | | Марченко І. В. | | | | | | | | |
| Перевір. | | Цевух І.В. | | | | | | | | |
| Т. контр. | | | | | | | | | | |
| Реценз. | | | | | | | | | | |
| Н. контр. | | Мелешкевич О.М. | | | | | | | | |
| Затверд. | | | | | | Лист 27 | | Листів | | |
| | | | | | | ОНПУ, ІРТ, РТ-091 | | | | |

[illegible]

[illegible]