

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
КОНОТОПСЬКИЙ ІНСТИТУТ

Кафедра електронних  
приладів і автоматики

Кваліфікаційна робота бакалавра  
**МІКРОКОНТРОЛЕРНІ СИСТЕМИ КЕРУВАННЯ  
ЕЛЕКТРОПРИВОДАМИ ПОСТІЙНОГО СТРУМУ**

Студентка гр. ЕП-61к

Д.О.Глухих

Науковий керівник  
к.ф.-м.н., ст. викладач

І.П.Бурик

Нормоконтроль,  
ст. викладач, к.т.н.

О.Д. Динник

## РЕФЕРАТ

Об'єктом дослідження кваліфікаційної роботи є мікроконтролери та мікроконтролерні системи керування електроприводами постійного струму.

Мета роботи полягає у порівнянні типових варіантів реалізації мікроконтролерної системи керування електроприводами постійного струму, розробці схеми електричної принципової, лістингу програми мікроконтролера, та в огляді сфери застосування таких систем.

При виконанні роботи було проаналізовано види, модифікації, переваги та недоліки мікроконтролерних систем керування електроприводами постійного струму, а також розроблено таку систему та написано лістинг програми для роботи такої системи.

Робота складається із вступу, трьох розділів основної частини та висновків. В першому розділі наведено огляд сучасних мікроконтролерних систем для електроприводу постійного струму. В другому розділі дається методика розробки та налагодження програмного забезпечення. В третьому розділі розглянуто особливості застосування мікроконтролерів ATtiny13 для електроприводу постійного струму.

Робота викладена на 34 сторінках, у тому числі включає 13 рисунків, 2 таблиць, список цитованої літератури із 28 джерел.

**КЛЮЧОВІ СЛОВА:** МІКРОКОНТРОЛЕР, ЕЛЕКТРОПРИВІД, ДВИГУН ПОСТІЙНОГО СТРУМУ, МІКРОКОНТРОЛЕРНА СИСТЕМА

## ЗМІСТ

<b>ВСТУП .....</b>	<b>4</b>
<b>РОЗДІЛ 1 ОГЛЯД СУЧАСНИХ МІКРОКОНТРОЛЕРНИХ СИСТЕМ..</b>	<b>5</b>
1.1 Мікроконтролери Texas Instruments .....	5
1.2 Архітектура 32-розрядних мікроконтролерів C2000 Piccolo і Delfino	6
1.3 Електропривідні системи на мікроконтролерах Motorola.....	10
<b>РОЗДІЛ 2 МЕТОДИКА І ТЕХНІКА ЕКСПЕРИМЕНТУ .....</b>	<b>15</b>
2.1 Методика розробки програми для мікроконтролера ATtiny13... ..	15
2.2 Налагодження програмного забезпечення..... ..	21
<b>РОЗДІЛ 3 РЕЗУЛЬТАТИ ТА ЇХ ОБГОВОРЕННЯ.....</b>	<b>24</b>
3.1 Керування двигунами постійного струму .....	24
3.2 Мікроконтролерна система керування електроприводом постійного струму на основі Atmel AVR ATtiny13.....	29
<b>ВИСНОВКИ.....</b>	<b>31</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....</b>	<b>32</b>
<b>ДОДОТОК А.....</b>	<b>34</b>

Сьогодні приводи з регулюванням швидкості потрібні не тільки в високопрофесійних і потужних промислових застосуваннях, таких як обробні машини або підйомні крани, але все більше і більше в побутовій техніці, наприклад, в пральних машинах, компресорах, невеликих насосах, кондиціонерах повітря і т.п. [1-10]

З'явилися відносно недавно мікроконтролери з ядрами Cortex-M3 / R4F / M4F, представлені сімейством LM3S, TMS570 і LM4F, дозволяють створювати системи зі зниженою споживаною потужністю і багатим набором комунікаційних можливостей, підвищеною надійністю і можливістю обрахунку складних алгоритмів керування приводом [1].

З появою сімейства двоядерних мікроконтролерів Concerto (Cortex-M3 + C28xx) можливості розробників ще більше розширились за рахунок поділу алгоритмів обміну з керуючою системою і управління рухом на різні ядра.

Мікроконтролерна система керування електроприводом постійного струму буде реалізована на мікроконтролері ATtiny13 що є найпростішою і найдешевшою з усього сімейства 8 розрядних мікроконтролерів Atmel AVR.

Проте, в своєму арсеналі вона має набір найрізноманітніших сучасних периферійних пристроїв. У тому числі два 8-розрядних таймера з можливістю роботи в режимі джерела сигналу з ШІМ (PWM). А також 4 каналний 10-розрядний АЦП.

## РОЗДІЛ 1

### ОГЛЯД СУЧАСНИХ МІКРОКОНТРОЛЕРНИХ СИСТЕМ

#### 1.1. Мікроконтролери Texas Instruments

У всьому світі в області цифрового керування електроприводом величезною популярністю користуються так звані DSC (Digital Signal Controllers) сімейства C2000 компанії Texas Instruments (TI). Таку зацікавленість пов'язано з високою продуктивністю пристроїв, багатим набором периферійних модулів і швидкою реакцією на зовнішні події [1].

В даний час можна виділити шість базових типів електродвигунів:

- колекторні (традиційні);
- безколекторні постійного струму (BLDC), засновані на трифазному синхронному двигуні змінного струму;
- крокові - спеціальні BLDC-двигуни з зубчастим ротором / статором (зуби статора формують «кроки», до яких притягується обмотка або відповідні зуби ротора);
- вентильно-індукторні (комбінація BLDC і уніполярного крокового двигуна);
- трифазні індукційні і синхронні змінного струму;
- однофазні змінного струму.

Для кожного з перерахованих типів електромоторів існують різні алгоритми управління.

Основну їх частину можна розділити на частотні (коли частота обертання валу є функцією частоти яке живить двигун напруги) і широтно-імпульсні (ШІМ, коли частота обертання є функцією ширини імпульсу сигналу з постійною частотою). Частотна схема управління в даний час застосовується все рідше за рахунок складності і низького ККД пристроїв управління. На зміну їй приходить ШІМ - управління, яка крім регулювання частоти обертання, дозволяє легко здійснювати і регулювання обертового моменту.

Велика кількість референс-дизайнів, що випускаються TI, дозволяє досконало вивчити як алгоритми управління, так і схемотехнічні особливості конструювання систем управління для кожного з типів двигунів. Для управління електроприводами можна використовувати мікроконтролери чотирьох сімейств, що випускаються TI. Основні переваги їх застосування наведені в таблиці 1.1.

*Таблиця 1.1*

**Основні переваги мікроконтролерів TI в системах управління електроприводами [1]**

Сімейство	Переваги використання в системах управління електродвигунами
C2000	Максимальна продуктивність електродвигунів; промислові і автомобільні Q100, середній рівень безпеки, широкий асортимент.
Hercules TMS570/RM48L	Максимальний рівень безпеки (ASIL-D, SIL-3); Q100 (TMS570), ARM, найбільший розмір flash-пам'яті.
Stellaris LM3S/LM4F	Дешеві ARM-мікроконтролери, велика кількість інтерфейсів обміну даними, простота додавання O / S, початковий рівень безпеки (в обладнанні).
MSP430	Мінімальна вартість, високий рівень монтування

Корпорація TI випускає два основних сімейства мікроконтролерів, забезпечених відповідними периферійними модулями: мікроконтролери реального часу C2000 і стандартні Stellaris з ядром Cortex-M3 або Cortex-M4F.

## **1.2. Архітектура 32-розрядних мікроконтролерів C2000 Piccolo і Delfino**

Платформа C2000 мікроконтролерів реального часу є галузевим лідером у створенні цифрових систем управління електродвигунами з початку випуску покоління TMS320F24x в 1996 р. У 2002 р була представлена серія F281x, заснована на DSP-ядрі C28x, яка стала першою серією з 32-розрядної

архітектурою, спеціально створеної для високопродуктивних систем управління за допомогою алгоритмів, що вимагають інтенсивних математичних обчислень.

З метою задоволення вимогам прикладних задач щодо продуктивності, ціни, призначення висновків і наявності периферійних модулів, на основі ядра C28x було створено сімейство сумісних за кодом мікросхем. В даний час популярні два сімейства - Piccolo і Delfino, що відрізняються ціною і обчислювальними можливостями.

Мікросхеми сімейства Concerto були довгоочікуваними для розробників вбудованих систем пристроями, оскільки дозволяють розділити покладені на ядро C28 завдання, пов'язані з об'єктом управління, і покладені на ядро M3 завдання, пов'язані з обміном даними з централізованими системами управління та моніторингу. Мікроконтролери Concerto розділені на три групи, що відрізняються продуктивністю (рис. 1.1). Коротка характеристика мікроконтролерів Concerto приведена в таблиці 1.2.

Таблиця 1.2

### Мікроконтролери Concerto [2]

Серія мікросхем	Продуктивність C28x/Cortex-M3, МГц	Пам'ять	Інтерфейси обміну даними
F28M35Nx	150/75 або 100/100	До 1 Мбайт flash-пам'яті, 132 кбайт ОЗУ	Ethernet, USB (OTG), SPI, SCI, CAN, I <sup>2</sup> C, McBSP
F28M35Mx	75/75	До 1 Мбайт flash-пам'яті, 132 кбайт ОЗУ	Ethernet, USB (OTG), SPI, SCI, CAN, I <sup>2</sup> C, McBSP
F28M35Ex	60/60	До 1 Мбайт flash-пам'яті, 132 кбайт ОЗУ	Ethernet, USB (OTG), SPI, SCI, CAN, I <sup>2</sup> C, McBSP

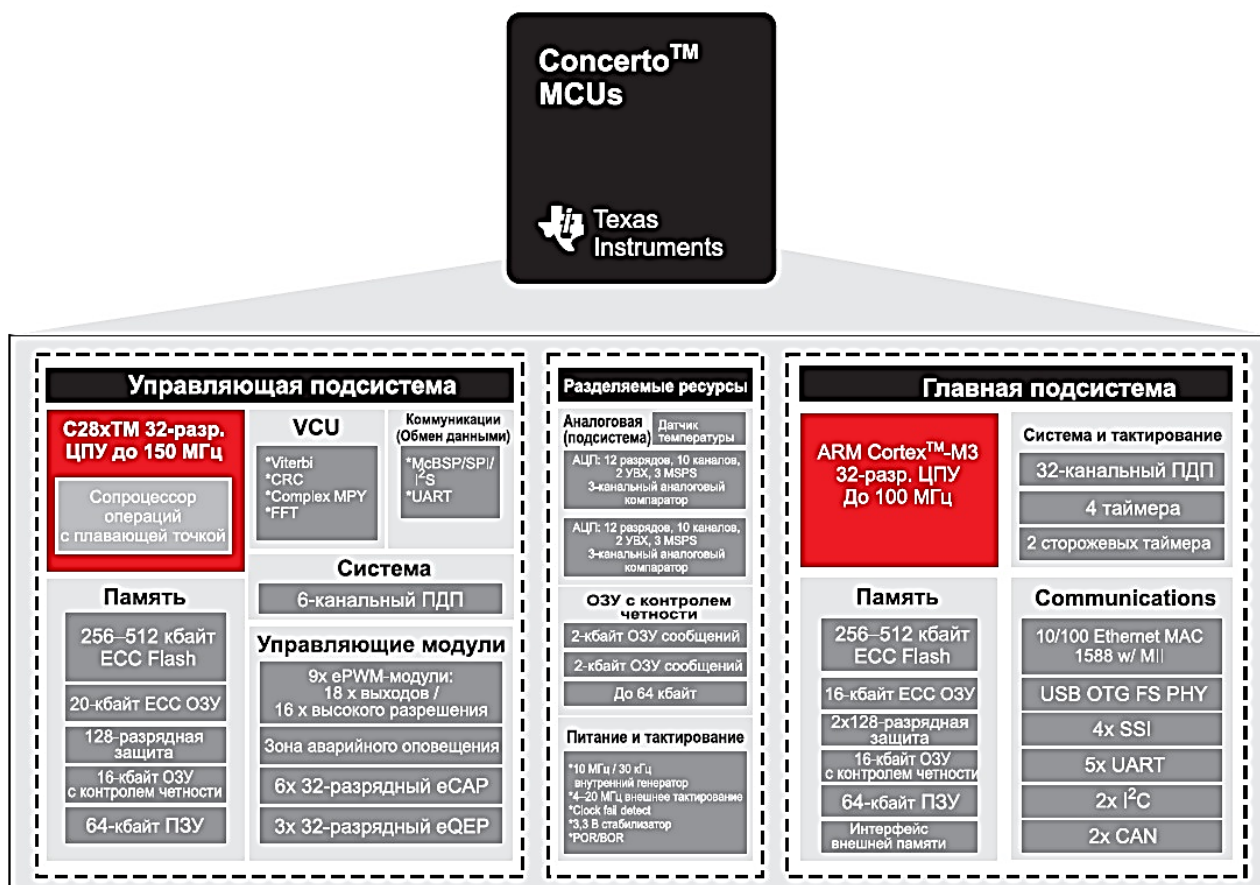


Рис 1.1. Микроконтролеры Concerto [2]

Мікроконтролери з ядром Cortex-M3 де-факто є на сьогодні стандартом для використання у вбудованих додатках. Вони отримали широке застосування за рахунок 32-розрядної архітектури, детермінованою реакції на переривання, невеликим споживанням потужності, наявності режимів зниженого енергоспоживання.

Зростанню популярності мікроконтролерів з ядром Cortex-M корпорації TI, що входять в сімейство Stellaris (рис. 1.2), сприяє випуску великої кількості мікроконтролерів з різним об'ємом вбудованої пам'яті (в тому числі вбудованого ПЗУ з бібліотекою для роботи з периферійними модулями) і різним набором периферійних модулів, що дозволяє вибрати мікроконтролери конкретно під задачу.

Подальшому зростанню популярності сімейства сприяє наявність великої кількості потужних засобів розробки і налагодження, таких як Keil, IAR і,



звичайно, TI CCS 4.0 і 5.0, а також велика кількість оціночних і налагоджувальних модулів і референс-дизайнів готових виробів [3].

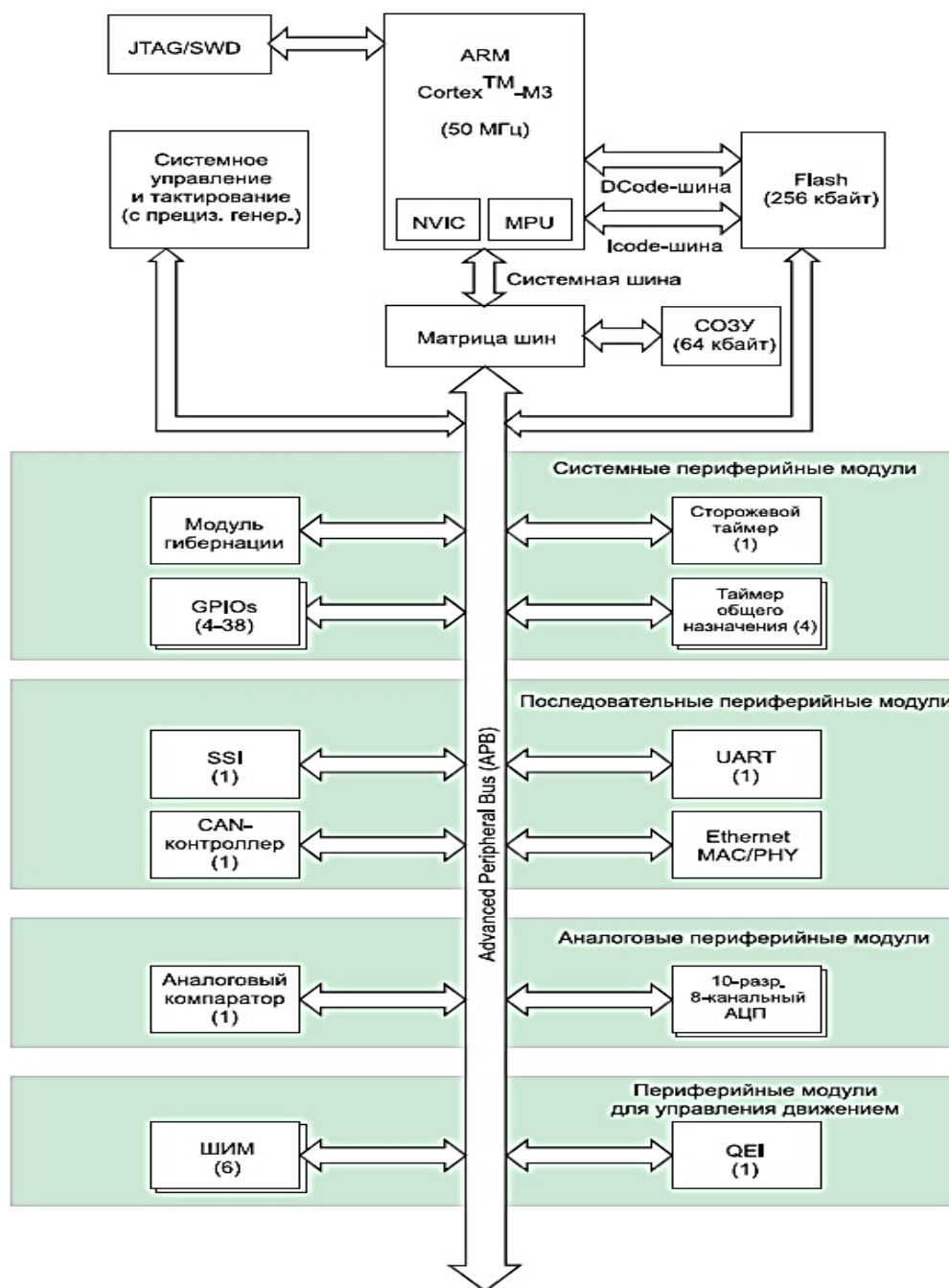


Рис.1. 2. Архітектура мікроконтролера Stellaris LM3S8971 [3]

У 2011 році до мікроконтролерів Stellaris LM3S з ядром Cortex-M3 додалися мікроконтролери LM4F з ядром Cortex-M4F, які відрізняються підвищеною робочою частотою і продуктивністю, розширеним набором команд, що включає команди цифрової обробки сигналів, і наявність співпроцесора

арифметики з плаваючою комою. Все це дозволяє реалізовувати на мікроконтролерах LM4F більш складні алгоритми управління.

Мінімальний набір що включають в себе периферійні модулі мікроконтролера для керування електродвигунами повинен включати наступні вузли:

- трьохканальний синхронний ШІМ з комплементарними виходами, можливістю регулювання «мертвого часу» і входом аварійного відключення висновків;
- кілька каналів АЦП для спостереження за різними параметрами в кожній з силових фаз управління електродвигуном;
- квадратура енкодер для установки вала в заданий початкове положення і спостереження за його обертанням;
- один зі стандартних для промислового застосування керуючих портів - UART (для RS232 / RS485), CAN або Ethernet-контролер.

### **1.3. Електропривідні системи на мікроконтролерах Motorola**

Збереження електричної енергії стає важливою частиною загальної тенденції щодо захисту навколишнього середовища. Електродвигуни, що призводять в дію системи в побуті і на виробництві, споживають значну частину виробленої енергії. Більшість цих двигунів працюють в нерегульованому режимі і, отже, з низькою ефективністю. Недавній прогрес в напівпровідниковій індустрії, особливо в силовій електроніці і мікроконтролерах, зробили приводи з регулюванням швидкості більш практичними і значно менш дорогими.

Приводи, керовані по розвиненим алгоритмам з допомогою мікроконтролерів, мають ряд переваг:

- збільшення енергетичної ефективності системи - регулювання швидкості знижує втрати потужності в двигунах;

- удосконалення функціонування - цифрове управління може додати такі властивості, як інтелектуальні замкнуті контури, зміна частотних властивостей, діапазону контрольованих несправностей і здатність до взаємодії з іншими системами;

- спрощення електромеханічного перетворення енергії - регульовані приводи дозволяють усунути необхідність в трансмісіях, коробках передач, редукторах;

- простота оновлення програмного забезпечення - системи на базі мікроконтролерів з флеш-пам'яттю можуть швидко змінювати при необхідності свій алгоритм і регульовані змінні.

З виконанням приводів регульованими складність системи часто збільшується. Основною умовою їх використання є збереження загальної вартості системи в обґрунтованих межах. Для ряду систем, особливо в побуті, загальна вартість повинна бути еквівалентна вартості нерегульованого варіанту.

Ці всі вимоги можуть задовольнити мікроконтролери MC68HC908MR24 і MC68HC (7) 05JJ6 / 7, розроблені фірмою MOTOROLA для дешевих застосувань в електроприводі. Наведено також практичні рішення для дешевих приводів з трифазним асинхронним двигуном, трифазним безколекторним двигуном постійного струму (БДПС) і універсальним колекторним двигуном змінного струму.

Мікроконтролер MC68HC908MR24 - мікроконтролер сімейства HC08, розроблений для застосувань в одно- або трифазних електроприводах. Він включають в себе 24 Кбайт FLASH, 768 байт RAM, два 16-бітових таймера, SPI, SCI (UART), 13 висновків I / O загального призначення і модуль LVR. Конструктивне виконання - QFP корпус з 64 висновками.

MR24 має спеціальні властивості, які орієнтують його на застосування в асинхронних електроприводах: 6-канальний 12-бітовий модуль PWM (ШИМ), порт виводу з високою здатністю навантаження по струму і 10-канальний, 10-бітовий АЦП. До основних рис 6-канального PWM-модуля входить:

- режими формування зосереджених або вирівняних по фронту ШІМ-сигналів;
- режим, в якому шість виходів конфігуруються в три комплементарні пари з когерентним оновленням інформації;
- реєстр генерації "мертвого часу" для запобігання наскрізних струмів в інверторі;
- входи для розпізнавання полярності струмів в фазах двигуна з метою корекції спотворень від наявності "мертвого часу" при перемиканні силових ключів інвертора;
- входи детектування несправності для швидкого відключення виходів PWM [4].

Наявні в PWM-модулі апаратні засоби усувають необхідність використання декількох зовнішніх компонентів (таких як логіка розпізнавання струмів, генерація "мертвого часу" і обробка сигналів відмов). На рис 1.3 показано блок-схему мікроконтролера.

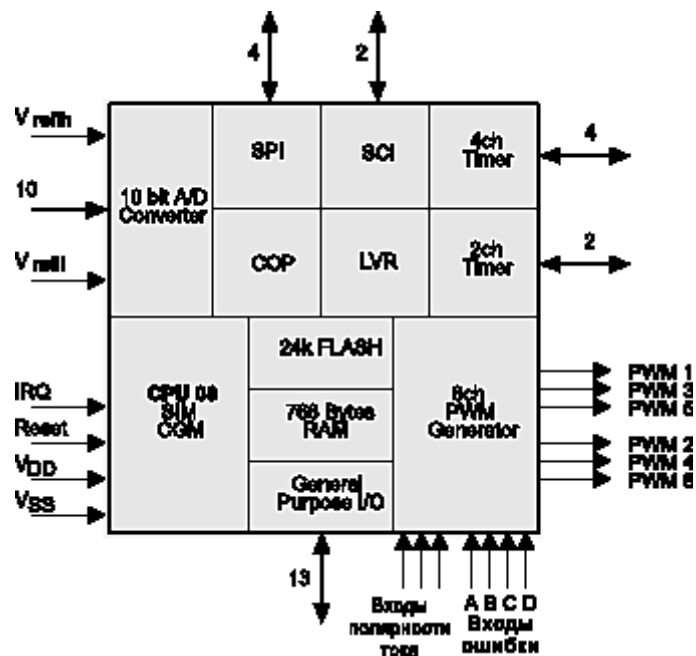


Рис 1.3. Блок-схему мікроконтролера MC68HC908MR24 [6]

Мікроконтролер MC68HC05JJ6 - мікроконтролер сімейства HC05, розроблений для простих застосувань з низькою вартістю. Основні властивості MR24 включають в себе:

- 6,1 Кбайт ROM, 224 байта RAM;
- 16-бітовий таймер, який має властивості захоплення по входу і порівняння по виходу;
- 14 висновків I / O загального призначення і простий порт послідовного введення / виводу (SIOP);
- корпус SOIC або DIP з 20 висновками.

На додаток JJ6 має спеціальні елементи, включаючи два компаратора, які можуть бути скомбіновані з 16-бітовим програмованим таймером для створення 4-х каналного single slope АЦП, а також порт введення / виводу з високим навантаженням по току і вбудований в кристал діод для вимірювання температури. Висока здатність навантаження по струму дуже важлива для простих застосувань, так як дозволяє скоротити загальне число елементів схеми. JJ6 має 6 висновків з навантаженням 10 мА. На рис 1.4 показано блок-схему мікроконтролера.

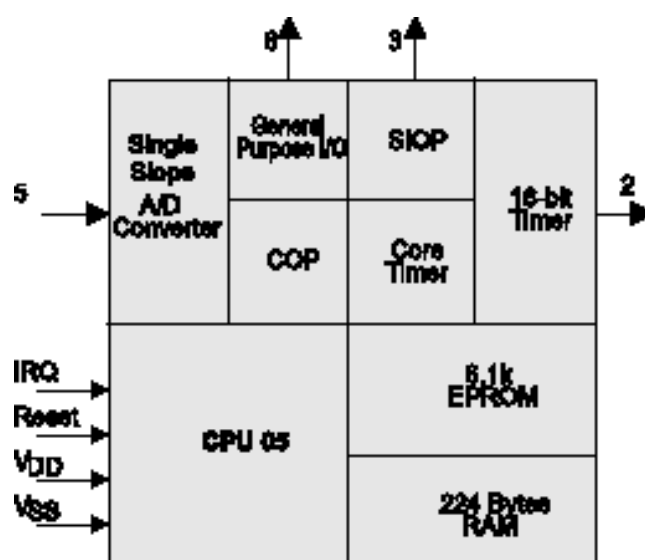


Рис 1.4. Блок-схему мікроконтролера MC68HC705JJ7 [6]

Дешевий високоефективний привід що немає датчика для бесколлекторного двигуна постійного струму на базі MC68HC908MR24

Трифазні безколекторні двигуни постійного струму (БДПС) - це хороші кандидати для дешевих високоефективних приводів через їх здатності до високої ефективності і забезпечення якості легкої керованості.

Недоліком цього типу двигунів є той факт, що їх комутація реалізується у функції положення ротора. Якщо для детектування положення ротора використовуються будь-які датчики, то для передачі отриманої інформації в пристрій управління потрібні додаткові приєднання до двигуна [7] .

Це може бути неприйнятним для деяких типів застосувань. Існує принаймні дві причини, за якими можливе усунування датчика положення:

- реальна неможливість здійснити додаткові з'єднання між датчиками положення і пристроєм управління;
- вартість датчиків положення і з'єднання.

Перша проблема може бути вирішена інтеграцією системи управління в корпус двигуна. Проте, все ще залишається достатня кількість застосувань, що вимагають рішення що немає датчика.

Розроблена технологія що немає датчика положення ротора показує точки, в яких наводиться в обмотках двигуна протидії ЕРС що переходить через нуль.

Точки переходу через нуль фазної протидії ЕРС реєструються, коли одна з трьох фаз обмотки не отримує живлення.

Отримана інформація обробляється для управління напругою фаз з використанням широтно-імпульсної модуляції.

## РОЗДІЛ 2

### МЕТОДИКА І ТЕХНІКА ЕКСПЕРИМЕНТУ

#### 2.1. Методика розробки програми для мікроконтролера ATtiny13

При створенні програмного забезпечення вирішено використовувати мову програмування «асемблер» замість «си», оскільки потрібно максимально ефективно використати обсяг програмної пам'яті в 1К [22-27].

Перша група операцій програми виробляє установку вершини стека. Стек використовує пам'ять SRAM мікроконтролера. Встановлюємо покажчик стека на кінець цієї пам'яті (пам'ять має 64 байта). Системна змінна RAMEND завжди містить адресу останньої клітинки пам'яті. Тобто в нашому випадку RAMEND = 64 [22]

```
ldi R24, RAMEND; Установка покажчика стека
out SPL, R24; SP = RAMEND
```

Наступна група операцій встановлює режим роботи порту PORTB. Режими кожної лінії порту встановлюються так, як показано на рис.2.1.

DDRB	Logical	Направл	Вив	Название	Функция
PB0	1	Выход	5	PB0/OC0A	PWM Вых (цифр)
PB1	1	Выход	6	PB1	LED Вых (цифр)
PB2	0	Вход	7	PB2/ADC1	ADC Вход (Аналог)
PB3	0	Вход	2	PB3	Switch Вход (Цифр)

Рис. 2.1. Ініціалізація Аналого-цифрового перетворювача [22]

Наступна група команд виконує ініціалізацію вбудованого АЦП мікроконтролера ATtiny13. Для того, щоб контролер зміг використати свій вбудований АЦП необхідно слідувати наступним вимогам.

Вибрати коефіцієнт попереднього розподілу та активувати АЦП, попередній дільник використовується для розподілу тактової частоти. Тактова

частота використовується в схемі послідовних наближень АЦП для перетворення аналогового сигналу [23].

Щоб отримати максимальний дозвіл потрібно вибрати частоту в межах від 50 кГц до 200 кГц. У даній конструкції використовується розподіл на 64, тому тактова частота складатиме приблизно 150 кГц (9600000 Гц / 64).

Завдяки тому, що не використовується автозапуск перетворення (ADATE) і переривання від АЦП (ADIF і ADIE), то можна встановити ці біти таким чином щоб, встановити коефіцієнт ділення 64 (ADPS2, ADPS1 і ADPS0) і включити АЦП (ADEN). Запис значення в регістр стану АЦП проводиться таким чином:

```
ldi R16,0b10000110
```

```
out ADCSRA, R16; Включити АЦП, з коефіцієнтом пред ділення 64
```

```
Вибрати джерело запуску процесу перетворення АЦП
```

Використовується автономний режим запуску, значить, вбудований АЦП буде сам періодично запускати процес перетворення, як тільки програма дозволить виконувати цей процес. [23]

Таким чином, записавши в усі три біта управління режимами запуску АЦП (ADTS2, ADTS1 і ADTS0) рівень логічного нуля, переводиться АЦП в автономний режим роботи:

```
ldi R16,0b00000000
```

```
out ADCSRB, R16; Автономний режим роботи
```

Для вибору каналу АЦП повинні вказати, який з каналів АЦП будемо використовувати, вибравши значення бітів управління мультиплексором АЦП (MUX1 і MUX0) в регістрі управління мультиплексором (ADMUX).

Вибераємо значення внутрішньої опорної напруги, що дорівнює 1,1 вольт, записавши у відповідний біт (REFS0) значення логічної "1":

Коли вбудований АЦП закінчить процес перетворення, результат вимірювання запишеться в пару регістрів ADCH і ADCL. Однак в цьому проекті використовується 8-розрядне значення.



Тому вибираємо режим зсуву вліво результату вимірювання. Для цього встановлюємо біт ADLAR в регістрі ADMUX в одиницю і будемо зчитувати результат тільки з реєстру ADCH.

```
ldi R16,0b01100001
```

```
out ADMUX, R16; Reference 1.1 Volt, Left Adjust, Channel: PB2
```

Відключення функції цифрового входу для аналогового каналу. Кожен розряд порту введення-виведення в звичайному режимі працює як вхід для цифрового сигналу [23].

Якщо даний вихід мікросхеми використовуємо як вхід АЦП, то рекомендується відключити функцію цифрового входу. Це дозволяє зменшити струм споживання і збільшити точність перетворення.

Записавши в біт ADC1D значення логічної "1", відключаємо функцію цифрового входу для лінії порту PB2:

```
ldi R16,0b00000100; Відключення цифрового входу для PB2
```

```
out DIDR0, R16
```

Ініціалізація режиму широтно-імпульсної модуляції

Мікроконтролер ATtiny13 має два незалежні канали ШІМ. Вихід каналу, який використовуємо в цьому проєкті, називається OC0A (виходить на PB0). А інший канал називається OC0B (виходить на PB1).

Для включення вбудованого ШІМ мікроконтролера ATtiny13 використовуємо наступні кроки [23].

Вибір режиму генерації сигналу

Мікроконтролер ATtiny13 підтримує режиму ШІМ. Перший з них називається "Fast PWM" (швидкий ШІМ). Він забезпечує найвищу частоту вихідного сигналу. Другий режим називається "Phase correct PWM" (ШІМ з коректною фазою).

Цей режим забезпечує симетричну зміну, як переднього, так і заднього фронтів вихідного сигналу. Цей режим працює на більш низьких частотах у порівнянні з "Fast PWM". В рамках цього проєкту будемо використовувати "Fast PWM".

Встановивши  $SOM0A1 = 1$  і  $SOM0A0 = 0$ , вибираємо режим скидання  $OS0A$  при збігу, і установки  $OS0A$  при досягненні кінця. Це означає, що в той момент, коли значення рахункового регістра  $TCNT0$  таймера виявиться рівним значенню регістра  $OCR0A$ , вихід  $OS0A$  ( $PB0$ ) буде скинутий в "0".

Коли ж таймер дорахував до свого максимального значення ( $TCNT0 = 255$ ), на виході  $OS0A$  встановиться логічна "1". Змінюючи значення  $OS0A$ , можна змінювати ширину вихідного імпульсу. Режим "Fast PWM" можна вибрати, встановивши в регістрі  $TCCR0B$  біт  $WGM2 = 0$ , а в регістрі  $TCCR0A$  біти  $WGM1 = 1$  і  $WGM2 = 1$ .

```
ldi R16,0b10000011
```

```
out TCCR0A, R16; Установка режиму Fast PWM, Скидання OS0A
```

Вибір тактової частоти на вході таймера [23]

Серце вбудованого ШІМ - 8-розрядний таймер / лічильник. Для формування вихідного сигналу існує спеціальна схема порівняння. вона безперервно порівнює значення рахункового регістра таймера  $TCNT0$  зі значенням регістра  $OCR0A$  і використовує результат порівняння для формування вихідного сигналу.

Для того щоб вся ця схема запрацювала, ми повинні подати на вхід таймера тактовий сигнал. Як тактового сигналу вибирається один з виходів зі спеціального вбудованого дільника тактової частоти. Для вибору одного з цих сигналів служить спеціальний регістр  $TCCR0B$ .

У цьому проекті використовуються сигнал з виходу дільника з коефіцієнтом ділення 256.

Лістинг програми наведено нижче.

```
*****
```

```
; Program: t13pwm.asm
```

```
; Description: Tiny13 Fast PWM and ADC Fan Controller
```

```
; Last Updated: 10.05.20
```

```
; Author: RWB
```

```

; IDE / Compiler: Atmel AVR Studio 4.14
; Programmer: AvrOspII v5.47 from Mike Henning
; : AVRJazz Tiny2313 Board
;*****
.include "tn13def.inc"
; Тактова частота мікросхеми Tiny13 за замовчуванням
.equ F_CPU = 9600000
.cseg
; Починаємо з нульової адреси пам'яті програм
.org 0
main: ldi R24, RAMEND; Установка показника стека
      out SPL, R24; SP = RAMEND
; Ініціалізація портів введення виведення
      ldi R16,0b00000011; Виставляємо PB0 = вихід, PB1 = вихід, PB2 = вхід,
PB3 = вхід, PB4 = вихід
      out DDRB, R16; DDRB = 0x03
; ініціалізація АЦП
      ldi R16,0b10000110
      out ADCSRA, R16; Включаємо АЦП, з коефіцієнтом ділення 64
      ldi R16,0b00000000
      out ADCSRB, R16; Автономний режим роботи
      ldi R16,0b01100001
      out ADMUX, R16; Внутрішнє джерело опорної напруги 1.1 В, режим зсуву
результату вліво, канал: PB2 (ADC1)
      ldi R16,0b00000100; Вимикаємо функцію цифрового входу для PB2
      out DIDR0, R16
; Ініціалізація ШІМ (PWM)
      ldi R16,0b10000011
      out TCCR0A, R16; Режим Fast PWM, скидання OC0A
      ldi R16,0b00000100

```

```

out TCCR0B, R16; Використовуємо коефіцієнт Дилен fclk / 256
; Ініціалізація прапора кнопки і регістра PORTB
ldi R17,0; Ініціалізація прапора кнопки
out PORTB, R17
lb_00: sbic PINB, PB3; перевіряємо умову (PB3 == 0)
rjmp lb_20; якщо немає, переходимо до lb_20
ldi R19,5; використовуємо підпрограму затримки
rcall delay_func
sbic PINB, PB3; якщо (PB3 == 0), читаємо знову
rjmp lb_20; інакше переходимо до lb_20
cpi R17,1; Обробка натискання кнопки
brne lb_10; якщо (R17! = 1) переходимо до lb_10
ldi R17,0; інакше R17 = 0
ldi R16,0; Вимикаємо генератор ШИМ
out TCCR0A, R16; TCCR0A = 0
out TCCR0B, R16; TCCR0B = 0
cbi PORTB, PB0; вимикаємо мотор
cbi PORTB, PB1; вимикаємо світлодіод
rjmp lb_20
lb_10: ldi R17,1; R17 = 1
ldi R16,0b10000011
out TCCR0A, R16; Режим Fast PWM, скидання OCR0A
ldi R16,0b00000100
out TCCR0B, R16; Використовуємо дільник fclk / 256
sbi PORTB, PB1; включаємо світлодіод
lb_20: cpi R17,1; якщо (R17! = 1)
brne lb_40; переходимо до lb_50;
sbi ADCSRA, ADSC; Старт АЦП перетворення
lb_30: sbic ADCSRA, ADSC; для (ADCSRA & (1 << ADSC))
rjmp lb_30

```

```

in R16, ADCH; Читаємо результат без урахування 2-х останніх біт з ADCL
out OCR0A, R16; OCR0A = R16
cbi PORTB, PB1; вимикаємо світлодіод
mov R19, R16
rcall delay_func; Викликаємо функцію затримки (тривалість в R19)
sbi PORTB, PB1; включаємо світлодіод
lb_40: mov R19, R16
      rcall delay_func; Викликаємо функцію затримки (тривалість в R19)
      rjmp lb_00
; Проста функція затримки
delay_func:
delay0: ldi R20,25; R20 = 25
delay1: ldi R21,255; R21 = 255
delay2: dec R21; Зменшуємо R21 на одиницю
      brne delay2; якщо (R20! = 0) переходимо до мітці delay2
      dec R20; Зменшуємо R20 на одиницю
      brne delay1; якщо (R20! = 0) переходимо до мітці delay1
      dec R19; Зменшуємо R19 на одиницю
      brne delay0; якщо (R19! = 0) переходимо до мітці delay0
      ret; Вихід їх підпрограми
.exit

```

### 3.2. Налаштування програмного забезпечення

Після компіляції та налаштування повинні отримати такий результат що зображено на рис. 2.2.

```

Build
-----
ATtiny13 memory use summary [bytes]:
Segment  Begin    End      Code  Data  Used   Size  Use%
-----
[.cseg]  0x000000 0x000080  128   0    128   1024  12.5%
[.dseg]  0x000060 0x000060   0    0     0     64   0.0%
[.eseg]  0x000000 0x000000   0    0     0     64   0.0%

Assembly complete, 0 errors. 0 warnings
-----
Build Find in Files Breakpoints and Tracepoints

```

Рис. 2.2. Результат налагодження мікроконтролера в програмному середовищі Build [27]

Власне кодом зайнято близько 128 байт із загального (1024 байт) обсягу програмної пам'яті. Цей дуже хороший результат, так як залишається ще місце на удосконалення програми на майбутнє.

Як видно з рис.2.2 1К програмної пам'яті мікроконтролера ATtiny13 цілком достатньо для нескладного пристрою використовує ШІМ і АЦП.

Тепер можна зашивати код в мікроконтролер. Але перш ніж зробити це необхідно перевірити мікросхему ATtiny13 чи правильно встановлені біти захисту та інші FUSE перемикачі [28]

Для правильної роботи описуваного пристрою в програмі OvrOspII біти FUSE перемикачів повинні бути встановлені таким чином як зображено на рис.2.3.

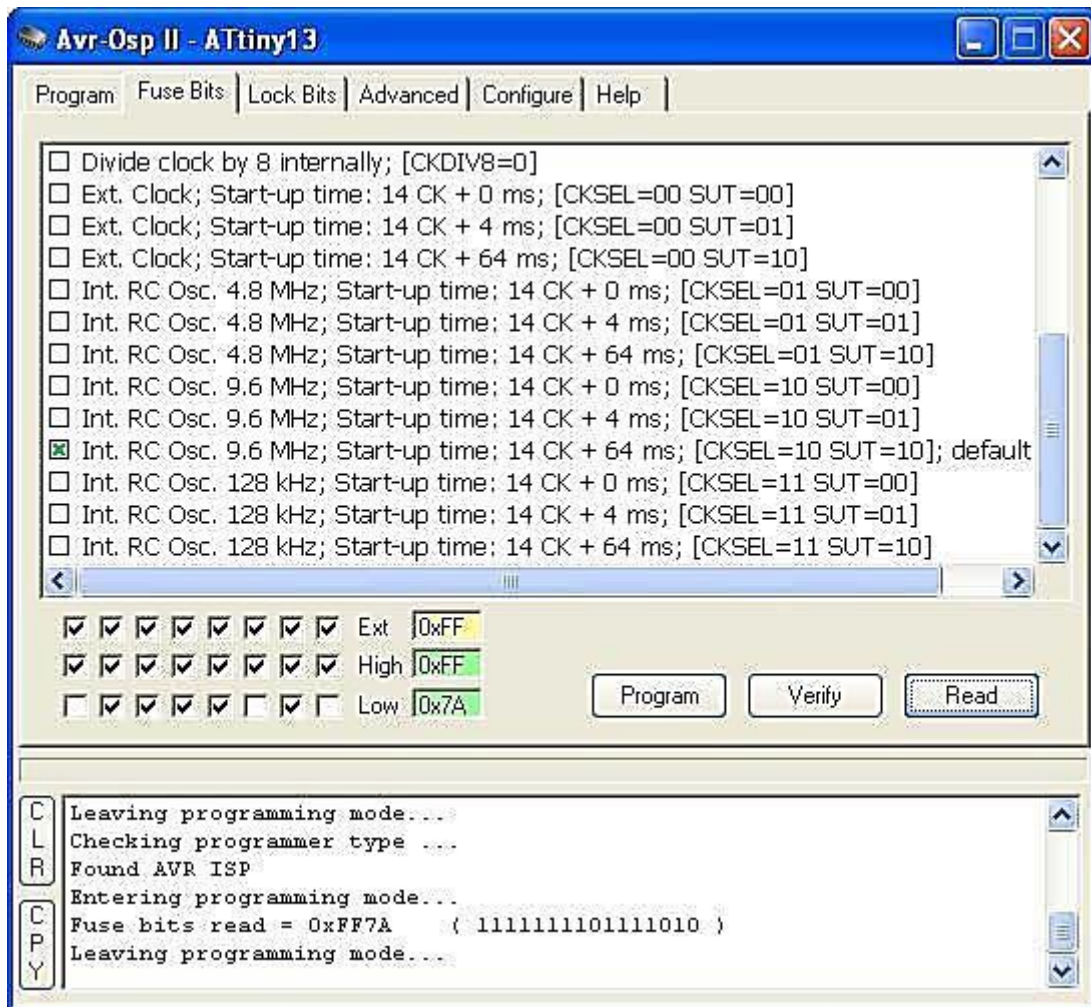


Рис. 2.3. Програмне середовище OvrOspII де встановлені біти FUSE перемикачів [27]

Необхідно вибрали режим синхронізації від внутрішнього RC генератора, частота 9.6МГц; час запуску: 14 СК + 64ms.

## РОЗДІЛ 3

### РЕЗУЛЬТАТИ ТА ЇХ ОБГОВОРЕННЯ

#### 3.1. Керування двигунами постійного струму

Двигун постійного струму має дві основні частини - ротор і статор. У колекторному двигуні статор - нерухома частина, складається з постійних магнітів (або в більш потужних двигунах електромагнітів). Ротор (якір) - обертається, суміщений з валом двигуна і складається з багатьох котушок (як мінімум трьох) [8].

Колектор (щітково-колекторний вузол) відповідає за перемикання виводів котушок ротора. Струм в такому двигуні підводиться до котушок ротора через ковзаючі контакти (або щітки). В один момент часу підключена тільки одна котушка, вона і створює момент обертання двигуна за рахунок проходить струму.

З точки зору базових елементів схемотехніки будь-який двигун можна представити у вигляді такої еквівалентної схеми яка зображена на рис.2.1:

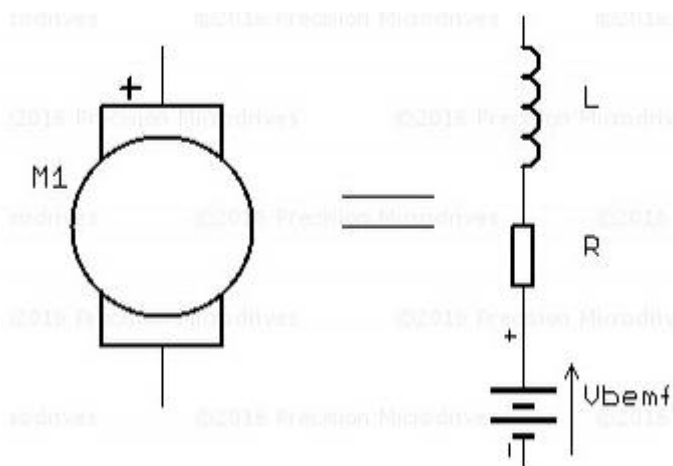


Рис 3.1. Схематичне зображення двигуна постійного струму та його схемотехніка [9]

Коли мотор підключений до джерела постійного струму і ще не почав обертатися, то він являє собою звичайне опір. Тобто через нього тече струм



відповідно до закону Ома і опору його обмотки. Переважає компонента  $R$ . Індуктивність починає впливати коли напруга не постійна, наприклад, якщо мотор живиться від ШІМ (PWM) сигналу.

Опір ротора і індуктивність, як правило, дуже малі. Його можна поміряти звичайним мультиметром.

Невеликі модельні мотори мають опір 1-10 Ом. Тому, при старті двигуна (коли він ще не почав обертатися), струм сильно перевищує робочий струм двигуна і якщо мотор довго буде нерухомий (його заклинило), то такий високий струм може привести до перегріву двигуна і виходу з ладу.

Індуктивність котушок ротора намагається підтримувати струм що протікає через обмотки. Її вплив помітно тільки коли напруга змінюється.

Коли мотор починає обертатися, то колектор починає перемикає котушки ротора, що викликає зміна напруги. Індуктивність намагається в ці моменти підтримувати струм протікає через мотор на постійному рівні за рахунок напруги.

Для того щоб колекторний мотор постійного струму почав обертатися, досить подати на нього певну напругу. Полярність даної напруги буде визначати напрямок його обертання, а величина напруги - швидкість обертання.

Напруга не може змінюватись безмежно. Кожен мотор розрахований на певний діапазон напруг [10] .

При підвищенні напруги струм через мотор буде рости, і він почне перегріватися і може згоріти. На діаграмі що зображено на рис.3.2 добре видно взаємозв'язок його основних показників.

Максимальної потужності мотор досягає при максимальному струмі. І залежність струму і моменту мають лінійну залежність. Максимальної швидкості двигун досягає при відсутності навантаження (при холостому ході), при збільшенні навантаження швидкість обертання падає. Номінальна робоча напруга вказано в паспорті на двигуні.

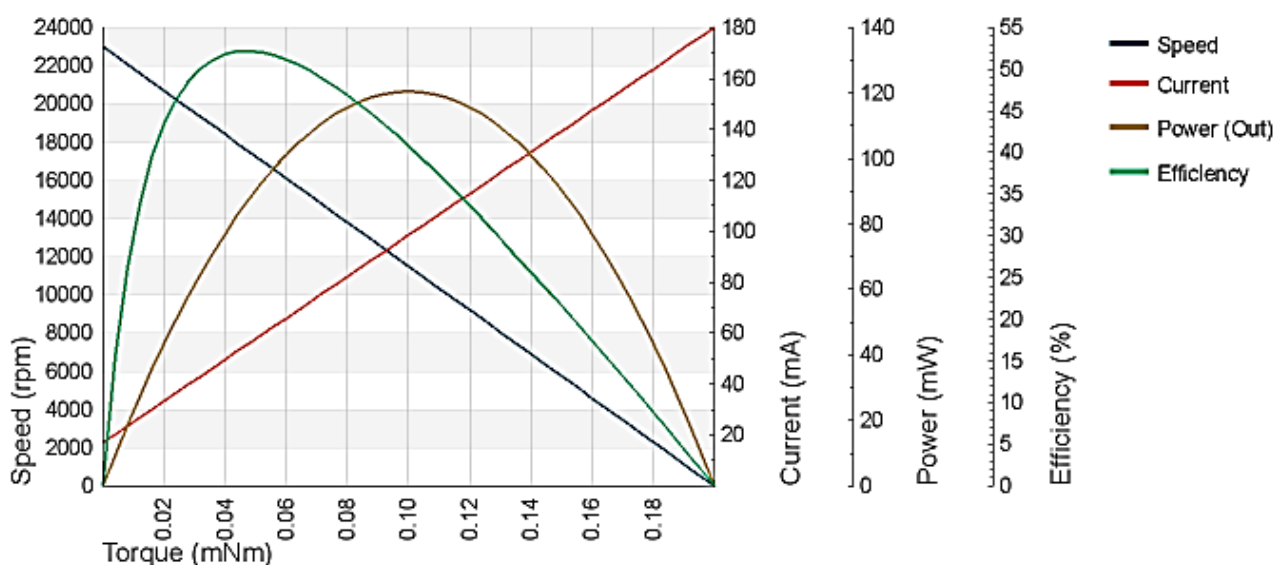


Рис. 3.2. Діаграма взаємозв'язку основних показників двигуна постійного струму [11]

При підключенні мікроконтролеру до двигуна необхідно керуватись наступними правилами. Порт мікроконтролера не може видати струм більше 50mA, і напруга живлення 12В для нього занадто велика. Для управління двигуном знадобиться електронний ключ - біполярний транзистор NPN, підключається він за схемою що наведено на рис. 3.3

Для того щоб мотор почав обертатися, на базу транзистора необхідно подати невеликий струм, далі транзистор відкриється і зможе пропустити через себе набагато більший струм і напруга - мотор буде обертатися. Варто зазначити що, якщо зберемо таку схему, то транзистор дуже скоро, якщо не відразу, вийде з ладу. Щоб цього не сталося, його необхідно захистити.

Для того щоб вимкнути мотор, необхідно закрити транзистор і опір на ньому різко збільшиться і він перестане пропускати через себе струм.

Однак індуктивність буде чинити опір цьому, і для того, щоб утримати струм на належному рівні, згідно із законом Ома, напруга на колекторі транзистора почне різко підвищуватися (може досягти навіть 1000В, правда дуже малий час) і транзистор згорить. Щоб цього не сталося необхідно паралельно обмоток мотора поставити діод, який відкриє шлях для зворотної напруги і замкне його на обмотку двигуна, тим самим захистити транзистор.

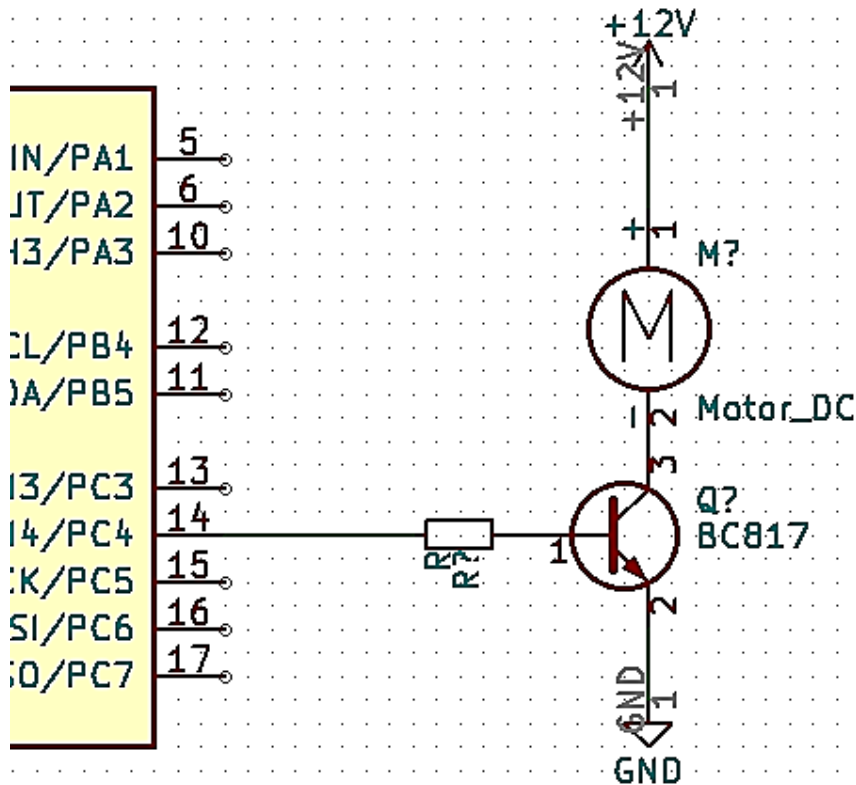


Рис.3.3. Схема підключення транзистора до мікроконтролера [12]

Також, всі двигуни постійного струму мають ще одну негативну характеристику - при обертанні механічний контакт в колекторі не ідеальний, щітки іскрять в процесі роботи, створюючи перешкоди, що може привести до збою мікроконтролера. [13]

Щоб знизити ці перешкоди, необхідно використовувати конденсатори невеликої ємності, підключений паралельно виводів двигуна (як можна ближче до самого мотору). На рис 3.4 зображено остаточно правильну схему (діод може бути не обов'язково Шотткі, але він кращий).

Біполярні транзистори у відкритому стані вони поведуться як діоди (на них падає близько 0.7 В). А це, в свою чергу, викликає їх великий нагрів на великих струмах і знижує ККД схеми управління мотором. Тому краще управляти моторами за допомогою польових (MOSFET) транзисторів. В даний час вони досить поширені і мають невисоку ціну. Їх низький опір у відкритому стані дозволяє комутувати дуже високі струми з мінімальними втратами. Однак і у них є свої недоліки. Так як MOSFET транзистори управляються напругою, а не струмом, то потрібно або вибрати спеціальні логічні MOSFET, які можуть

управлятися низькою напругою – 1.8 .. 2.5В або використовувати спеціальні схеми накачування напруги (драйвери польових транзисторів) [15].

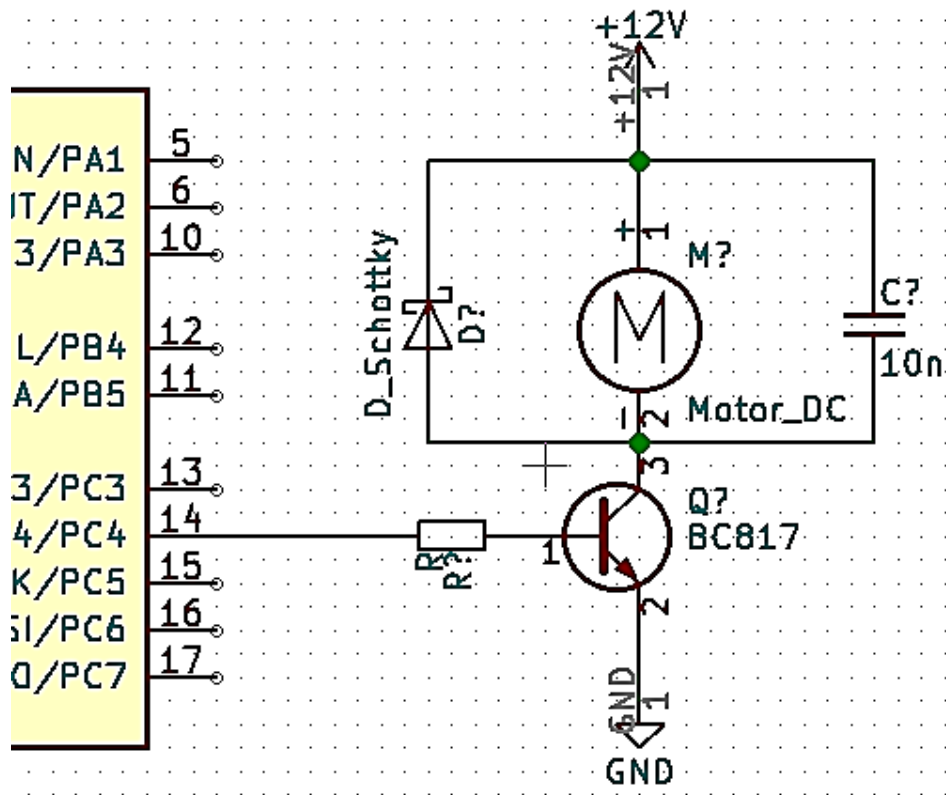


Рис. 3.4. Схема підключення транзистора до мікроконтролера з захистом від перенавантаження [14]

Тепер, подаючи на вихід мікроконтролера логічну одиницю, ми змусимо мотор обертатися, а логічний нуль – зупиниться. Однак обертатися він буде з постійною швидкістю і тільки в одну сторону. Розглянемо, як цього можна досягти за допомогою мікроконтролера.

### 3.2. Мікроконтролерна система керування електроприводом постійного струму на основі Atmel AVR ATtiny13

Мікросхема ATtiny13 є найпростішою і найдешевшою з усього сімейства 8 розрядних мікроконтролерів Atmel AVR. Проте, в своєму арсеналі вона має набір найрізноманітніших сучасних периферійних пристроїв. У тому числі два 8-розрядних таймера з можливістю роботи в режимі джерела сигналу з ШІМ

(PWM). А також 4 каналний 10-розрядний АЦП. Не дивлячись на те, що обсяг пам'яті у даної мікросхеми не великий (1К пам'яті програм і всього 64 байта SRAM і 64 байта EEPROM), проте вона прекрасно підходить для більшості завдань вимагають застосування генератора ШІМ і АЦП. На рис.3.5 зображено зовнішній вигляд мікроконтролера АТіny13.

Нижче наведено рисунок, де зображено зовнішній вигляд мікроконтролера АТіny13 та розпінування виводів.

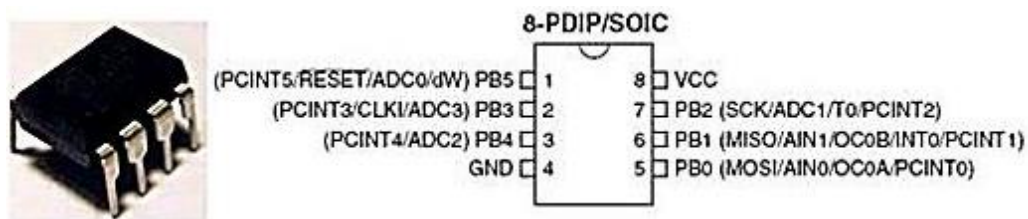


Рис. 3.5. Зовнішній вигляд мікроконтролера АТіny13 та розпінування виводів [24]

Для того щоб продемонструвати можливості мікроконтролера АТіny13, використовуючи ШІМ та АЦП, вирішено використовувати його для управління швидкістю двигуна.

Підключивши двигун постійного струму через ключовий каскад на транзисторі TIP120 до виходу OC0A мікроконтролера АТіny13, можливо легко управляти швидкістю двигуна, використовуючи ШІМ. На рис.3.6 наводиться повна схема для цього проекту.

Змінний резистор R4 (10K) використовується в якості джерела керуючого напруги, що подається на вхід АЦП (вхід ADC1 АТіny13 - PIN 7). Перемикач (S1), служить для ввімкнення та вимкнення вентилятора і світлодіодного індикатора (LED1), який своїм миготінням відображає роботу програми. Частота миготіння світлодіода відповідає змінам сигналу ШІМ. Якщо двигун обертається швидше індикатор блимає повільно. Коли двигун обертається повільніше частота миготіння індикатора збільшується. Швидкість обертання вентилятора регулюється змінним резистором R4 який включений як дільник

напруги. Переміщаючи движок цього резистора можна змінювати рівень напруги на вході АЦП мікроконтролера [25].

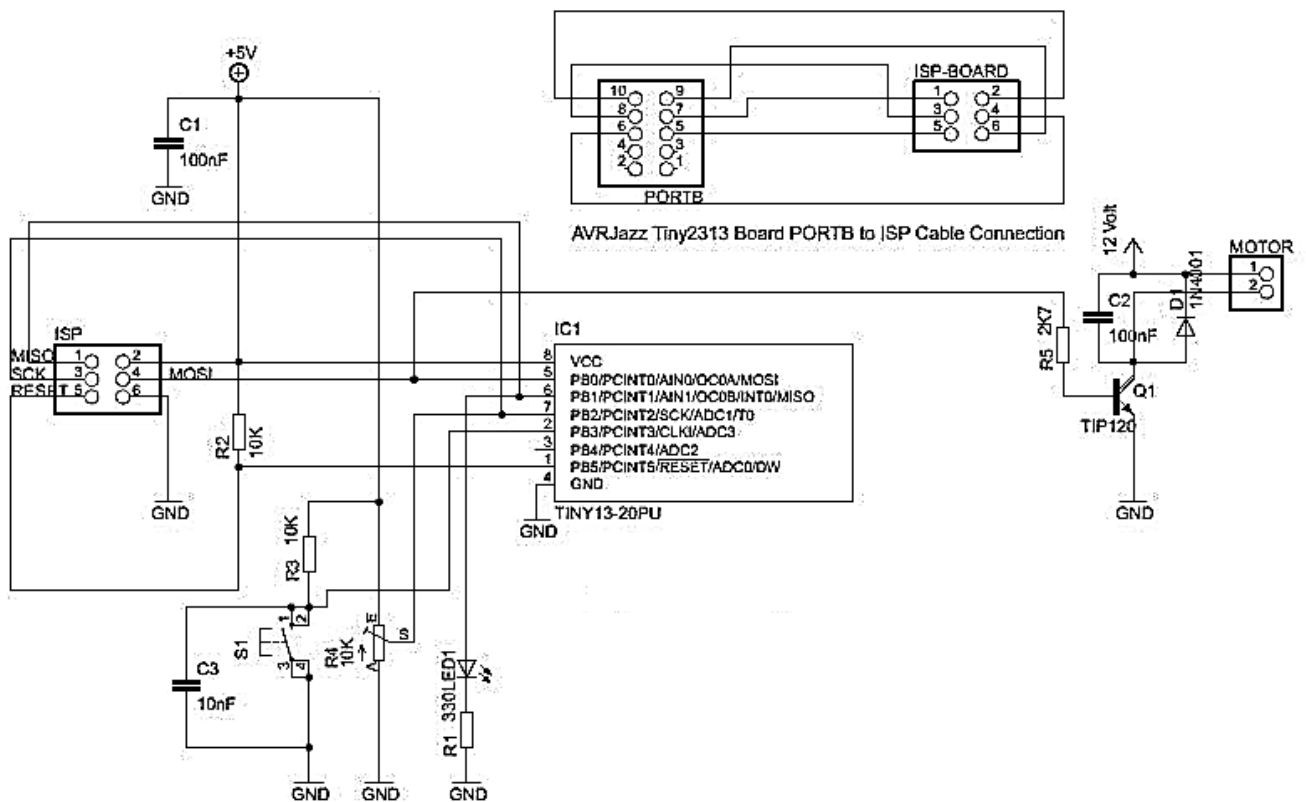


Рис. 3.6. Система керування електроприводом постійного струму на мікроконтролері ATtiny13 [25]

В цій системі керування використовувалися наступні радіоелементи, обладнання та програм: 4 резистора: 2,7К, 10К (2 шт), 300 Ом, 1 змінний резистор: 10К, 2 Конденсатора 100нФ, 1 конденсатор: 10нФ, 1 діод: 1N4001, 1 світлодіод, мікроконтролер ATtiny13, транзистор: пара Дарлінгтона TIP120, джерело постійного струму 12В, налагоджувальна плата AVRJazz, програматор OvrOspII розробки Міккі Хеннінга, Atmel AVR Studio 4 для введення, компіляції та відлагодження програми.

## ВИСНОВКИ

1. Було розглянуто архітектури мікроконтролерних систем Texas Instruments, Motorola та Piccolo і Delfino та їх застосування для керування електроприводами постійного струму

2. Здійснено опис принципу роботи електроприводу постійного струму та його реакції на навантаження, робота в холостому ході та досліджена діаграма взаємозв'язку основних показників двигуна постійного струму.

3. Створено лістинг програми роботи мікроконтролерної системи керування електроприводами постійного струму за допомогою програмного середовища Atmel AVR Studio 4 та показано етапи завантаження та налагодження програми.

3. Розроблено схемотехнічні рішення з варіаціями підключення мікроконтролера до електроприводу постійного струму, зокрема мікроконтролерна система керування електроприводом постійного струму була реалізована на мікроконтролері ATtiny13, налагоджування мікроконтролера здійснено за допомогою налагоджувальної плати AVRJazz, за для програмування мікроконтролера використовувався програматор OvrOspII розробки Міккі Хеннінга.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Основные тенденции развития встроенных систем управления двигателями и требования к контроллерам / В. Козаченко // СИИР NEWS. – 2006. – №1. – С.2-9.
2. Control Integrated Circuit Designers<sup>4</sup> Manual. International Rectifier, 2010. Стр 155 – 161
3. Power Semiconductors. Product Digest 47th Edition, 2015. Стр. 158 - 164
4. PIC16/17 MICROCONTROLLER DATA BOOK, 2017. Стр. 166 – 174
5. Однокристалльные микроконтроллеры Microchip: PIC16C8X. / Пер. с англ. Под ред. А.Н. Владимирова. – Рига.: ORMIX, 2003. –120 с.
6. Системы управления электродвигателями на микроконтроллерах фирмы MOTOROLA / R. Visinka, L. Chalupa, I. Skalka // СИИР NEWS. – 2008. – №1. – Р.14-16.
7. Chalupa L. Low Cost High Efficiency Sensorless Drive For Brushless DC Motor Using MC68HC(7)05MC4 / L. Chalupa // Motorola Semiconductor Application Note, AN1627, 1998. Стр 288 - 297
8. Skalka I. Low Cost Universal Motor Choper Drive System / I. Skalka // Motorola Semiconductor Application Note, AN1661, 1998. Стр. 255 - 264
9. Skalka I. Low Cost Universal Motor Phase Angle Drive System / I. Skalka // Motorola Semiconductor Application Note, AN1662, 1998. Стр 166 – 168
10. Skalka I., Low Cost Universal Motor Sensorless Phase Angle Drive System / I. Skalka // Motorola Semiconductor Application Note, AN1663, 1998. Стр. 99 - 103
11. Skalka I., Chalupa, L., Visinka, R., High Voltage medium Power Board for Three Phase Motors. Motorola Semiconductor Application Note, AN1590, 1998. Стр. 149 - 153
12. Visinka R. Low Cost 3-phase AC Motor Control System Based On MC68HC908MR24 / R. Visinka // Motorola Semiconductor Application Note, AN1664, 1998. Стр. 55 - 64



13. Wilson D. "Get Your Motor Running" with the MC68HC908MP16 / D.Wilson // Motorola Semiconductor Application Note, AN1712, 1997. Стр. 88 - 92
14. Wilson D. Making Low-Distortion Waveforms with the MC68HC908MP16/ D. Wilson // Motorola Semiconductor Application Note, AN1728, 1997. Стр. 69 - 76
15. MC68HC908MR24 General Release Specifications, Motorola Inc., April 1, 1998. Стр. 168 - 174
16. MC68HC05JJ6 General Release Specifications, Motorola Inc., September 6, 1996 Стр. 89 - 95
17. Stellaris ARM Cortex-M-based Microcontrollers. [http://www.ti.com/lscds/ti/microcontroller/arm\\_stellaris/overview.page](http://www.ti.com/lscds/ti/microcontroller/arm_stellaris/overview.page)
18. Stellaris ARM Cortex™-M Microcontrollers Brochure. <http://www.ti.com/lit/pdf/slab054>
19. Бородин И.Ф. Автоматизация технологических процессов / И.Ф. Бородин, Н.М. Недилько. – М.: Агропромиздат, 1986. Стр. 134 - 140
20. Бородин И.Ф. Автоматизация технологических процессов / И.Ф. Бородин, Ю.А. Судник. – М.: Колос, 2004. – 344 с.
21. ATMEL / Офіційний сайт: [http// www.atmel.com](http://www.atmel.com)
22. Гончар В.Ф., Тищенко Л.П. Електрообладнання і автоматизація с.-г. агрегатів і установок. – К.: Вища школа, 1989. – 343 с.
23. Шпак Ю.А Программирование на языке С для AVR и PIC микроконтроллеров. – К.: МК-Пресс, 2006. – 400 с.
24. Иванов Ю.И. Микропроцессорные устройства систем управления / Ю.И. Иванов, В.Л. Югай. – Таганрог: Изд-во ТРТУ, 2005. – 133 с.
25. Баранов В.Н. Применение микроконтроллеров AVR: схемы, алгоритмы, программы / В.Н. Баранов. – М.: Додэка-XXI, 2004. – 288 с.
26. Евстифеев А.В. Микроконтроллеры AVR семейства Classic фирмы Atmel / А.В. Евстифеев. – М.: Издательский дом "Додэка-XXI", 2002. – 288 с.
27. Голубцов М.С. Микроконтроллеры AVR: от простого к сложному / М.С. Голубцов. – М.: СОЛОН-Пресс, 2004. – 304 с.

28. Астраханцев А.В. Микропроцессоры и микроконтроллеры. Архитектура и периферийные устройства микроконтроллеров семейства AVR / А.В.Астраханцев, Ю.И.Мошкович, А.А.Тестоедов.– Севастополь: СевНТУ, 2005. – 40с.

## **ДОДАТОК А**