

МІКРОПРОЦЕСОРНА СИСТЕМА ВИМІРЮВАННЯ ТА ВІЗУАЛІЗАЦІЇ ЧАСТОТИ ОБЕРТАННЯ І КУТА ПОВОРОТУ ВАЛУ ЕЛЕКТРОДВИГУНА

У роботі приведено опис розробленої мікропроцесорної системи вимірювання та відображення на текстовому LCD дисплеї частоти обертання і кута повороту валу електродвигуна. Передбачається її використання, як складової більш складної системи векторного частотного управління.

In this article the developed microprocessor system of measuring and displaying on the text LCD display speed and rotation angle of the motor shaft is described. Expected to use it as a component of a more complex system of vector frequency control.

1. ФОРМУЛЮВАННЯ ПРОБЛЕМИ ТА АНАЛІЗ РОБІТ ЇЇ ПРИСВЯЧЕНИХ

При побудові замкнених цифрових систем керування швидкістю електродвигунів виникає потреба у вимірюванні кутових переміщень та частоти обертання їх валів. Точність вимірювання при цьому грає вирішальну роль у забезпеченні високої якості процесу управління (точності підтримки заданої швидкості, її стабільності при зміні величини чи характеру навантажень) [2].

Розвиток і вдосконалення давачів швидкості електродвигунів завжди був спрямований на підвищення точності і надійності з одночасним зменшенням габаритів та зниженням вартості. Тахогенератори (асинхронні, синхронні, постійного струму), що у минулому були основними засобами вимірювання швидкості, витісняються сучасними імпульсними сенсорами швидкості. Пов'язано це, насамперед, із переходом від аналогових до цифрових систем керування та індикації, для яких ЧМ і ШМ сигнали давачів є більш «привабливими» ніж аналоговий сигнал, який повинен бути обов'язково перетворений за допомогою АЦП у цифрову форму. Проте, за умов сучасного стану розвитку мікропроцесорної техніки, достатньо точну цифрову інтерпретацію аналогового сигналу здійснити не так складно, тому що практично будь-який мікроконтролер містить хоча б один вбудований багатоканальний блок АЦП з роздільною здатністю від 8 до 12 біт. Не менш важливими проблемами тахогенераторних давачів є нелінійність вихі-

¹ Українська академія друкарства

дної характеристики (у асинхронних і безконтактних постійного струму) та нестабільність параметрів, пов'язана із змінами температури і перехідного опору контактної пари (у колекторних тахогенераторах постійного струму).

Крім тахогенераторів існують ще давачі швидкості побудовані на основі ефекту Холла, ротаційні енкодери та резольвери. Сенсори частоти обертання на основі ефекту Холла не володіють високою роздільною здатністю, тому в основному використовуються у системах індикації частоти обертання чи управління вентилятором. Ротаційні енкодери бувають двох типів: абсолютні і інкрементальні. Відмінності та особливості кожного з цих двох типів енкодерів приведені у [3]. Резольвер ще називають енкодером з синусоїдальним виходом, хоча між ними існує багато відмінностей. Основою резольвера є обертовий трансформатор. За один оберт він генерує на своїх виводах певну, залежну від конструкції, кількість періодів синусоїди і косинусоїди. Ці аналогові сигнали зазвичай перетворюються у цифрові інтерфейсною платою резольвера.

На даний час найчастіше використовують два основних способи вимірювання кутових переміщень та швидкостей:

- 1) за допомогою імпульсного тахогенератора;
- 2) з використанням інкрементального ротаційного енкодера.

Перший спосіб є дешевшим у реалізації, проте має ряд суттєвих недоліків. Зокрема: нижча точність у порівнянні з другим способом, недостовірність виміряних значень при низьких швидкостях, складність визначення кута повороту, потреба у додаткових елементах для захисту системи управління від перенапруг та фільтрації. Саме тому, коли необхідна висока точність вимірювання у широкому діапазоні швидкостей використовують ротаційний енкодер [1].

Застосування імпульсних давачів швидкості автоматично не забезпечує високої точності регулювання швидкості, оскільки вимагає відповідного програмного вирішення для перетворення отриманої інформації у форму, зручну для подальшої обробки та реєстрації.

2. ФУНКЦІОНАЛЬНА СХЕМА СИСТЕМИ ВИМІРЮВАННЯ ТА ВІЗУАЛІЗАЦІЇ ЧАСТОТИ ОБЕРТАННЯ І КУТА ПОВОРОТУ

Враховуючи вище сказане, розроблено схему підключення ротаційного енкодера *Autonics E40S8-5000-6-L-5* до мікроконтролера *STM32F103RB*, який опрацьовуватиме його сигнали (рис. 1).

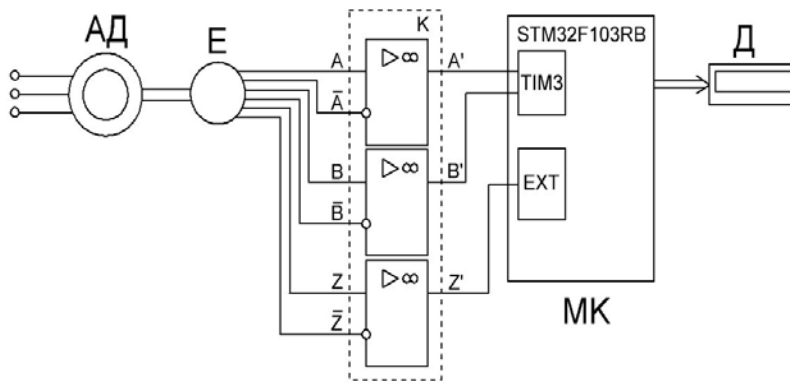


Рис. 1. Схема підключення інкрементального ротаційного енкодера Autonics E40S8-5000-6-L-5 до мікроконтролера STM32F103RB

Енкодер, закріплений на валу асинхронного двигуна, генерує імпульси на своїх шести виводах (їх ще називають фазами), частота яких прямопропорційна частоті обертання валу. Виводи \bar{A} , \bar{B} і \bar{Z} є диференційними для A , B і Z відповідно. В однакові проміжки часу логічні рівні на диференційних виводах протилежні. Прямі і диференційні сигнали енкодера попарно поступають на входи компараторів K , на виходах яких одержуємо три сигнали A' , B' і Z' (рис. 2). У такий спосіб значно підвищується завадостійкість системи. Сигнали A' і B' є основними, поступають на входи таймера-лічильника $TIM3$ контролера МК, який виконує функції фіксації та підрахунку кількості імпульсів. Сигнал у фазі B' зміщений відносно сигналу фази A' на чверть періоду (рис. 2). Завдяки цьому, опрацьовуючи одночасно сигнали цих двох фаз, можна безпомилково визначити напрям обертання валу енкодера. Якщо вал енкодера обертається у напрямку за годинниковою стрілкою, то при фіксації у фазі A' фронту імпульсу, логічний рівень у фазі B' буде низьким, а при обертанні проти годинникової стрілки – високим. Програмна реалізація такої перевірки є досить складною задачею, через високу частоту імпульсів основних сигналів A' і B' , тому в обраному контролері передбачена можливість відповідного налаштування таймера для виконання апаратної перевірки і встановлення біта напрямку обертання (старший біт лічильного регістра). Оскільки лічильний регістр у даному контролері є 16-ти бітним, то числове значення в ньому лежить у діапазоні $[-32768; 32767]$. У фазі Z' імпульс з'являється лише раз за один оберт валу енкодера (індексний імпульс). Таким чином, під'єднавши Z' до входу блоку фіксації зміни логічного рівня EXT , можна обчислювати кількість здійснених повних обертів.

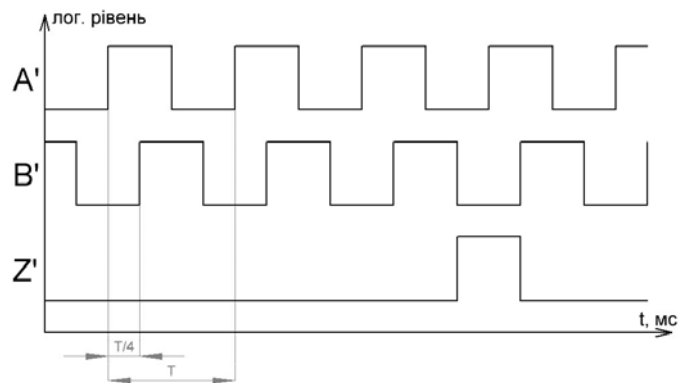


Рис. 2. Форма та взаємовідношення сигналів ротаційного енкодера

Для відображення вимірних значень частоти та кута повороту валу енкодера використовується текстовий LCD дисплей (Д) *Winstar WH1602* на базі контролера *HD44780*. Обмін інформацією між МК і дисплеєм здійснюється по 4 лініях даних та 3 лініях управління.

3. АЛГОРИТМ ФУНКЦІОНУВАННЯ МІКРОПРОЦЕСОРНОЇ СИСТЕМИ ТА ПРОГРАМА УПРАВЛІННЯ КОНТРОЛЕРОМ

Для визначення кута повороту та частоти обертання вала ротаційного енкодера було розроблено алгоритм, що зображений на рис. 3. Блоки 1-4 відображають послідовність виконання процесів налаштування необхідної периферії контролера МК. Налаштування *RCC* включає в себе конфігурування робочих частот усіх блоків МК, які будуть використовуватись програмою та частоти, на якій виконуватимуться обчислення. Також на цьому етапі відбувається конфігурація системного таймера *SysTick*, який зазвичай використовується операційними системами реального часу. У даному випадку застосування операційної системи не передбачається, тому він призначений для створення часових затримок, які зручно використовувати при написанні програми. Наприклад, для розподілення часу на виконання не пов'язаних між собою операцій: обчислення значень швидкості та кута повороту не обов'язково супроводжується виведенням значень на дисплей. *SysTick* сконфігурований таким чином, щоб переривання генерувались кожні 0,5 мс, тому усі часові інтервали у програмі будуть кратними цьому значенню.

Налаштування входів/виходів (блок 2) відповідає за вибір режимів роботи усіх задіяних виводів МК. На цьому ж етапі відбувається налаштування комунікації з текстовим LCD дисплеєм, за алгоритмом, що

описаний у документації на даний дисплей [3]. Блок 3 алгоритму (налаштування таймера *TIM3*) передбачає налаштування і дозвіл переривання при переповненні лічильного регістра таймера, конфігурування регістрів *TIM3* для роботи у режимі розпізнавання сигналів енкодера. Зокрема інкрементування/декрементування лічильного регістра *CNT* таймера буде відбуватись не лише за фронтами імпульсів фаз *A'* і *B'*, але й за зрізами, тобто за один оберт валу енкодера число в лічильному регістрі буде у 4 рази більшим від роздільної здатності самого енкодера ($4*N=20000$ імп./об.). Відповідно підвищиться і точність вимірювань. Наступний блок алгоритму демонструє оголошення констант та ініціалізацію змінних, котрі будуть у ньому використовуватись: *Fs* – частота, з якою здійснюється обчислення значень швидкості та кута повороту ($Fs=100\text{Гц}$); *N* – роздільна здатність енкодера ($N=5000$ імп./об.); *STB* та *DTB* – числа, що вказують на співвідношення між частотою переривань від таймера *SysTick* та частотами вимірювання і виведення інформації на дисплей відповідно. Якщо виконуються умови 5 (відбулося переривання від таймера *SysTick*) та 6 (змінна *Speed_TB* стала рівною 0), опрацьовується група блоків 8-17. Вони відповідають за: визначення кількості імпульсів, яку зафіксував таймер *TIM3* у лічильному регістрі, між двома «сусідніми» вимірюваннями (тобто за 10 мс) – блок 9; обчислення за виразами значень швидкості (блок 10) та кута повороту (блок 11); обмеження величини вимірюваного значення швидкості на рівні *MAX_SPEED* – вимірювання вище цього значення вважається недостовірним або недопустимим (блоки 13-15); обчислення середнього арифметичного значення швидкості (блоки 16-17).

Якщо виконуються умови блоків 5 та 18 (змінна *Display_TB* стала рівною 0), то опрацьовуються блоки 20-22, що відповідають за відображення вимірних значень на дисплеї. Ці умови будуть виконуватись кожні 0,2 с, і відповідно через ці проміжки часу буде оновлюватись зображення на дисплеї.

Умови, представлені у блоках 5, 6, 18, перевіряються циклічно, аж до моменту вимкнення живлення мікроконтролера – у цьому випадку виконається умова 24 і виконання програми завершиться.

На основі алгоритму поданого на рис. 3 розроблено програму управління контролером *STM32F103RB* для опрацювання сигналів енкодера та виведення вимірних значень на дисплей. Програма написана на мові «Сі» у середовищі розробки програм CoIDE для мікроконтролерів на базі ядра ARM. Для перетворення коду програми у машинний код використовувався компілятор GCC. Для запису та відлагодження програми застосовувався програматор-відладчик *ST-LINK*, що є складовою макетної плати *STM32VLDISCOVERY*.

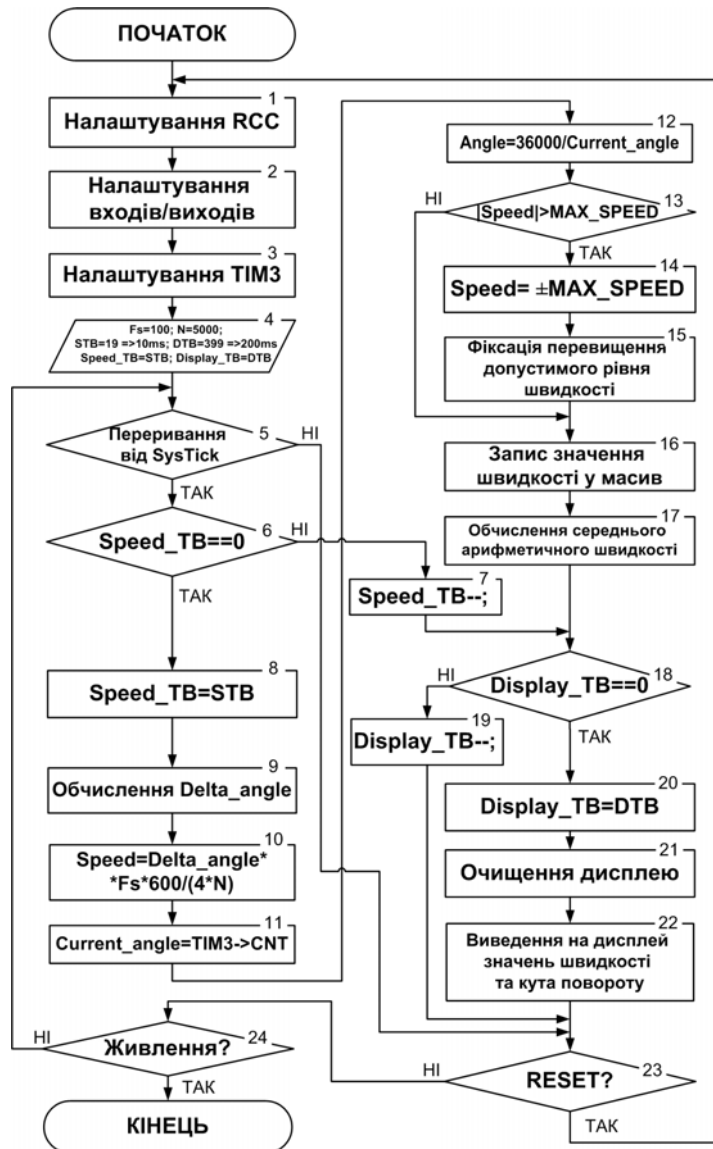


Рис. 3. Алгоритм визначення кута повороту та частоти обертання валу ротаційного енкодера

4. ВИСНОВКИ

1. Запропоновано функціональну схему підключення ротаційного енкодера, що виступає давачем швидкості ротора електродвигуна, до мікроконтролера.
2. Розроблено алгоритм опрацювання сигналів енкодера, визначення частоти обертання та кута повороту валу енкодера і виведення отриманих значень на LCD дисплей.
3. На основі алгоритму складено програму на мові «Сі» для вимірювання та відображення швидкості та кута повороту вала двигуна на базі контролера *STM32F103RB*.
4. Виготовлено експериментальний зразок розробленої мікропроцесорної системи вимірювання і реєстрації параметрів швидкості та кутового переміщення давача.
5. Розроблена мікропроцесорна система дозволяє з високою точністю визначати частоту обертання і кут повороту валу енкодера, що дає можливість застосовувати її у замкнених системах управління швидкістю електродвигунів та системах позиціонування.

1. Аристов Е.В., Хузин Р.А. Исследование датчиков скорости – Пермь: Изд-во Перм. Гос. Техн. Ун-та, 2007. – 13 с. 2. Евтихий Н.Н., Купершмидт Я.А., Папуловский В.Ф., Скугоров В.Н. Измерение электрических и неэлектрических величин: Учеб. пособие для вузов. – М.: Энергоатомиздат, 1990. – 352 с. 3. Смирнов Б. Промышленные энкодеры, ООО «Ракурс» [Электронный ресурс]. – Режим доступа: www.rakurs.ru/files/Prom_encoders.pdf, 5 с. 4. Winstar Display Co. specification for WH1602A No.163 Chung Ching RD., Taichung, Taiwan, R.O.C – специфікація на дисплей, 21 с. 5. Парк Д., Маккей С., Райт Э. Передача данных в системах контроля и управления. – М.: ООО «Группа ИДТ», 2007. – 480 с. 6. STM32F10x ReferenceManual Rev2 – технічна документація для сімейства мікроконтролерів STM32F10. – 501 с.