

Лабораторна робота №1

Програмування і дослідження роботи контролера Durus.

Мета: ознайомитися з будовою ПЛК Durus, придбати навички програмування і роботи з контролером.

Лабораторна установка, рис.1.1, призначена для програмування і дослідження роботи контролера Durus виробництва GE Fanuc.

Контролер Durus забезпечує трипозиційне регулювання температури в печі, значення якої вимірюють два позиційні датчики ТК-3. При досягненні температурою 30°C спрацьовує перший датчик, сигнал з якого передається на дискретний вхід контролера. Відповідно до програми користувача контролер вмикає вентилятор. Якщо температура продовжує підвищуватися і досягає значення 35°C спрацьовує другий датчик, по сигналу з якого контролер вмикає додатковий другий вентилятор. Одночасно з цим починає працювати програма керування послідовним вмиканням чотирьох ламп із затримкою 6с.



1- кнопка подачі живлення на лабораторну установку, 2- кнопка подачі напруги на нагрівальну піч, 3- кнопка для зміни потужності печі, 4-світлодіоди біжучого вогню, 5- джерело живлення, 6- датчики температури, 7 контролер, 8- вентилятори.

Рисунок 1.1 – Лабораторна установка

Контролер DURUS

DURUS - контролер виробництва GE Fanuc, який розрахований на нескладні завдання логічного управління, рис.1.2, має розміри 72x90x59,6мм і встановлюється на DIN-рейку. Робочий температурний діапазон – в межах 0-55 градусів за Цельсієм.

DURUS є представником мікроконтролерів з вбудованим операторським інтерфейсом, комунікаційними опціями і різними комбінаціями каналів вводу-виводу. Залежно від вибору моделі кількість входів-виходів може бути 10, 12, 20, 24. Є можливість підключення 3-х модулів розширення (доступні 3 моделі дискретного вводу-виводу і 1 аналогового вводу). З урахуванням модулів розширення загальна кількість зовнішніх сигналів може досягати 44-х.



Рисунок 1.2 – Контролер DURUS

Дискретні входи розраховані на 12 або 24В постійного, а також на діапазон 100-240В змінного струму. Аналогові входи призначені для вводу сигналів напруги 0-10В і постійного струму 4-20 мА (модулі розширення). Залежно від моделі використовуються релейні виходи до 8А, транзисторні виходи 24В/1А, лічильники імпульсів до 1кГц і виходи ШІМ.

Напруга живлення контролера залежить від вибраної моделі:

- моделі на постійний струм мають напругу 12В, від 20,4 до 28,8 або від 85 до 265 VDC;
- моделі на змінний струм - 24В або від 85 до 265 VAC.

У лабораторній роботі використовується малогабаритний контролер DURUS модифікації IC210MDR024, в якому є 12 дискретних входів на 24В постійного струму, 4 аналогових входа напругою 0-10В постійного струму та 8 релейних виходів на струм до 8А.

Контролер має операторський інтерфейс – LCD-дисплей на 4 рядки по 12 символів і 8 клавіш для програмування.

Пам'ять контролера здатна зберігати до 17 призначених для користувача екранів. Є вбудований годинник реального часу. Контролер здатний інтегруватися в розподілені системи за рахунок використання комунікаційних модулів розширення Modbus RTU Slave, Profibus_DP або DeviceNet.

Програмується пристрій інженерними мовами релейноконтактних схем (Ladder) і функціональних блоків (Function Blocks). Спільний розмір призначеної для користувача програми у вбудованій флеш-пам'яті контролера може досягати 200 сегментів LD або 99 блоків FBD, а швидкість її виконання складає 10мс на цикл.

Максимальна кількість таймерів, лічильників і компараторів у програмі обмежена 15 шт. Діапазон підрахунку часу від 0,01с і до 9999хв. Програмування може здійснюватися як на комп'ютері, за допомогою програмного середовища Durus Development Software, так і за допомогою LCD дисплея. У контекстному меню програми Durus Development Software присутня кнопка допомоги, що значно спрощує роботу. Після вибору контролера у верхній частці контекстного меню з'являються кнопки. Серед них, окрім стандартних кнопок (Run, Stop, Quit, Save, Print і тому подібне), є кнопки симуляції і кнопка "Key Pad".

Перша дозволяє налагодити програму, що створена інженерними мовами, навидь не маючи самого контролера, а друга при цьому повністю емулює поведінку LCD дисплея контролера, у тому числі при маніпуляціях з кнопками управління. При роботі з мовою Ladder доступні не лише стандартні булеві операції, але і виклик функціональних блоків, які в тексті програми показуються у вигляді "котушок".

Бібліотека містить 7 різних типів таймерів (з окремим входом скидання і без, із затримкою по вмиканню і вимиканню та ін.), 8 типів лічильників (інкрементний, декрементний, із збереженням та ін.), 5 типів компараторів (більше, менше, діапазон та ін.). Функції інформаційного обміну дозволяють з'єднуватися з 8-ма іншими контролерами і пересилати їм дані по мережі RS-485.

Програмування Digrus з лицевої панелі контролера

Увімкнути напругу на лабораторному стенді. Для входу в головне меню, що на цифровому дисплеї контролера, натиснути кнопку ESC. Користуючись кнопками ↑ або ↓, вибрати INITIAL і натиснути OK. У підменю, що відкриється, вибрати мову LADDER і підтвердити кнопкою OK. Очистити пам'ять контролера. Для цього в головному меню перевести курсор на CLEAR PROG., натиснути послідовно OK і YES.

Щоб створити перший сегмент програми, який за сигналами з позиційних терморегуляторів вмикає в роботу вентилятор, необхідно увійти у меню LADDER, встановити курсор у крайнє ліве положення і натиснути кнопку SEL. З'явиться символічне ім'я I1 прямого контакту. Натиснути OK, змінна I1 зафіксується, а курсор зміститься праворуч. Натиснути SEL, на дисплеї з'явиться другий прямий контакт з таким самим символічним ім'ям I1. Щоб з прямого отримати інвертований контакт з символічним ім'ям i2, необхідно залишити курсор на другому контакті I1 і двічі натиснути кнопку SEL. Велика літера «I» заміниться на прописну «i». Перевести курсор на індекс «1» і кнопкою ↑ збільшити до 2, натиснути OK. Тут стан змінних I1 та i2 визначається вихідними сигналами терморегуляторів.

Щоб створити лінію сполучення, тричі натиснути кнопку SEL. Перевести курсор у крайнє праве положення і натиснути кнопку SEL. З'явиться пряма котушка Q1. Оскільки контролер здійснює керування вентилятором з другого дискретного виходу, адресу прямої котушки треба змінити на Q2. Для цього курсор перевести на індекс біля літери Q і кнопкою ↑ збільшити його до 2.

Перевести курсор на нову строку і за попередньою аналогією створити послідовність контактів I2-i1. Щоб з'єднати паралельно перший і другий сегмент програми, поставити курсор за контактом i1 і натиснути кнопку SEL.

Оскільки лампи біжучого вогню спочатку роботи контролера мають бути вимкнуті, необхідно перевести курсор у праве крайнє положення і натиснути кнопку SEL, з'явиться пряма котушка Q1. Щоб

надати їй статус Reset-котушки необхідно курсор перевести на дужку, що перед літерою Q, і вибрати кнопкою ↑ або ↓ ідентифікатор Reset-котушки – стрілку ↓. Для надання цій котушці адреси виходу контролера, що керує роботою першої лампи, перевести курсор на індекс біля літери Q і кнопкою ↑ збільшити його до 5. Для паралельного підключення Reset-котушки Q5 до котушки Q2 курсор установити попереду Set-котушки Q5 і натиснути кнопку SEL. Так само запрограмувати Reset-котушки Q6-Q8, які керують роботою решти ламп.

Програма, яка забезпечує автоматичне регулювання температури в печі та керування послідовним вмиканням ламп має наступний вигляд:



Користуючись наведеною програмою і навичками отриманими при створенні перших сегментів, завершити самостійно створення програмного коду.

Для програмування таймера затримки вмикання лампи, курсор перевести у крайнє праве положення і натиснути кнопку SEL. З'явиться пряма котушка Q1. Встановити курсор на літеру Q і кнопкою ↑ або ↓ вибрати літеру T, натиснути ОК. З'явиться меню настройки таймера.

Для створення затримки при вмиканні лампи на 6с, курсор перевести на друге знаомісце і натиснути кнопку SEL. Кнопкою ↑ встановити час затримки 6 і натиснути ОК. Для виходу з меню таймера, натиснути ESC.

Щоб запустити контролер, кнопкою ESC увійти в головне меню, вибрати альтернативу RUN і натиснути ОК. У підменю RUN PROG курсор перевести на YES і підтвердити вибір кнопкою ОК. Контролер почне працювати відповідно до створеної програми.

Для припинення роботи контролера у головному меню скористатися альтернативою STOP.

Програмування контролера за допомогою інструментальної системи програмування Durus Development Software

Запустити програму, користуючись її ярликом на моніторі EOM. Коли з'явиться вікно системи програмування, увійти в меню File→New→NewLAD, щоб створити проект LAD-мовою. З'явиться екран LAD.Version:1.0 з панеллю інструментів. Лівою клавішею миші натиснути на кнопку New і у вікні Selekt Model Type, що з'явиться, вибрати тип контролера IC210MDR024, натиснути ОК. Відкриється робоче поле для створення програмного коду. Поміж двома вертикальними шинами живлення знаходиться редакційна сітка, а у нижній частині екрану - панель інструментів, за допомогою якої у клітках редакційної сітки встановлюються мовні елементи:

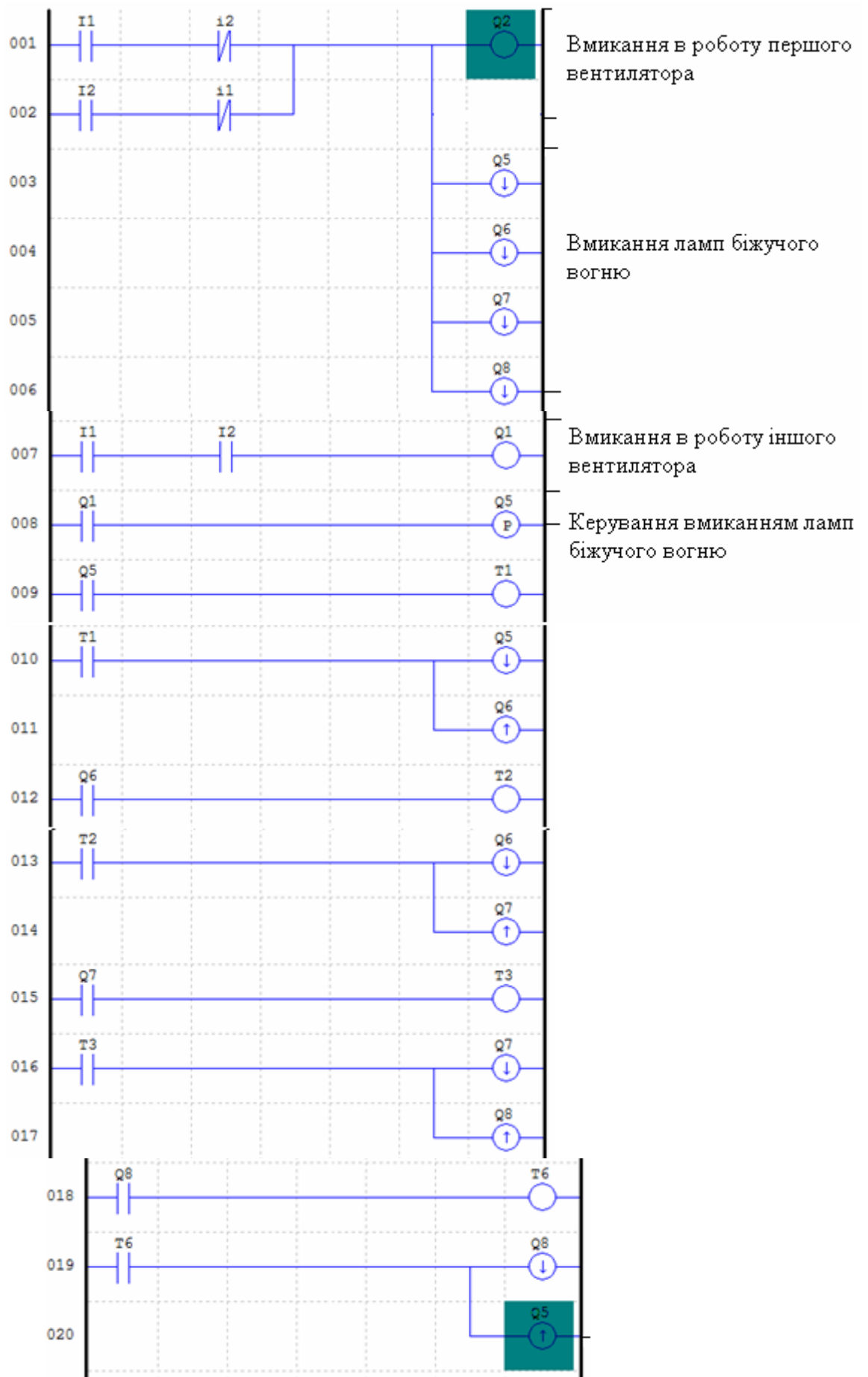


Для створення першого сегменту програми, який забезпечить вмикання першого вентилятора, в панелі інструментів вибрати кнопку I-контакта  і натиснути ліву клавішу миші. Перевести курсор у ліву крайню клітинку першого сегменту програми і повторно клацнути. У вікні Edit Contact, що з'явиться, призначити контакту перший номер і вибрати тип STR (прямий контакт):

Натиснути ОК і контакт I1 з'явиться у першому сегменті, а зелений квадрат зміститься праворуч. Щоб послідовно за прямим контактом установити інверсний, необхідно клацнути лівою клавішею миші по зеленому квадрату. У вікні Edit Contact призначити змінній індекс № 2, вибрати тип контакту STR NOT (інвертований контакт) і натиснути ОК. За допомогою кнопки  у панелі інструментів продовжити лінію сполучення до передостанньої клітинки редакційної сітки, а в останню встановити пряму котушку Q, позначивши у вікні Edit Contact/Coil індекс змінної -2 і її тип ”-(”:

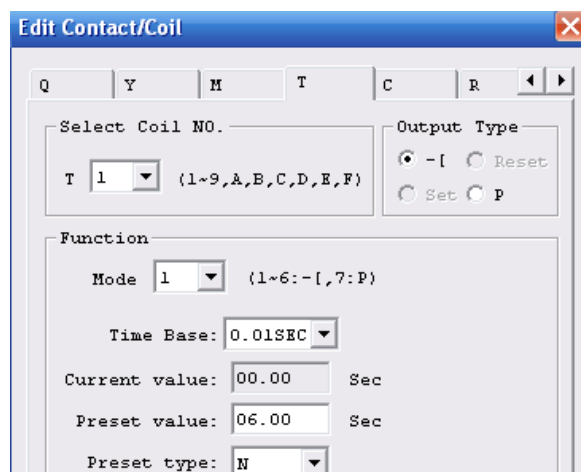
Створити таку саму послідовність контактів у другому сегменті програми, тільки прямому контакту призначити символічне ім'я I2, а інвертованому – i1. За допомогою кнопки  у панелі інструментів приєднати праву сполучну лінію інвертованого контакту i1 до першого сегменту програми, забезпечивши таким чином паралельне сполучення контактів першого і другого сегментів. У сегментах програми 3-6 паралельно котушці Q2 встановити Reset-котушки Q5, Q6, Q7, Q8, які вимикають лампи.

Завершити створення програми, користуючись наведеним кодом:



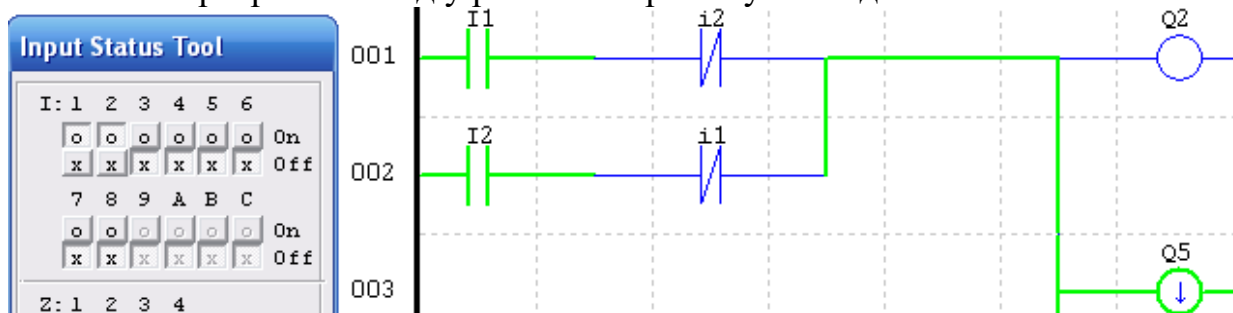
При установленні таймера скористатися в панелі інструментів кнопкою . Відкриється діалогове вікно Edit Contact/Coil, в якому

призначити відповідний індекс для таймера T і час затримки Preset value: 06.00sec:



Решту параметрів залишити не змінними.

Щоб перевірити роботу програми, увійти в меню проекту Operation і активізувати Simulator. В панелі інструментів натиснути лівою клавішею миші кнопку . З'явиться симулятор входних каналів Input Status Tool і програмний код у різнокольоровому вигляді:



Перемикаючи лівою клавішею миші в емуляторі стан змінних I1 та I2 з положення Off у положення On, можна переглядати роботу програми. При цьому елементи програми, які набувають стан TRUE, фарбуються у зелений колір.

Коли програма відкоригована, необхідно зв'язатися з контролером. Відкрити меню Operation, активізувати альтернативу Stop і натиснути ліву клавішу миші на Link Com Port. У вікні даного пункту вибрати COM1 PORT і натиснути клавішу Link. Для завантаження програми у контролер, увійти в меню Operation → Write, а щоб запустити її необхідно натиснути лівою клавішею миші кнопку  в панелі інструментів.

Завдання для самостійної роботи

Модифікувати код програми регулювання температури печі так, щоб кількість спрацьовувань датчиків температури підраховував лічильник.

Зміст звіту

Звіт має містити назву роботи, короткі відомості про контролер, програму регулювання температури печі, результати самостійної роботи і висновки.

Контрольні запитання

1. Яка конструкція контролера?
2. Якими мовами програмується контролер DURUS?
3. Яке призначення мають клавіші на лицевій панелі DURUS?
4. Як здійснюється програмування контролера з EOM?
5. Що містить бібліотека контролера?
6. Які можливості має пакет програмування Durus Development Software?
7. Для чого призначений годинник реального часу?
8. Який розмір може мати програма?
9. Як програмується таймер?
10. Як здійснюється завантаження програми у контролер?