

Лабораторна робота №2 Створення програмного коду LD- мовою

Знайомство з редактором LD- мови

Для знайомства з редактором LD-мови необхідно за методикою, що приведена у лабораторній роботі №1, створити новий проект і, двічі клацнувши лівою клавішею миші по іконі Main* теки Logical POU's(Логічні POU's), відкрити робочий листок для створення тіла коду LD-мовою.

Створений новий проект Main з'явиться у вікні проектного дерева, рис.2.1, у піддеревах Logical POU's і Physical Hardware.

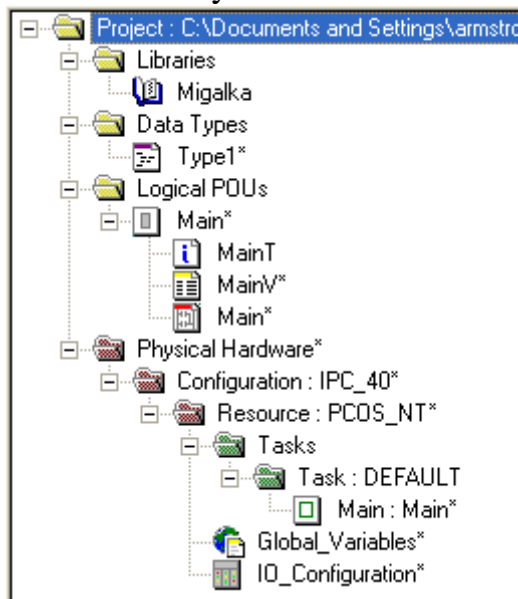


Рис. 2.1. Дерево проекту

Для створення LD- мережі необхідно:

- у зручному місці робочого листка тіла коду клацнути лівою клавішею миші, щоб позначити вихідну точку програмування;
- в панелі інструментів клацнути по іконі Contact network (Контактна мережа) і в робочому листку тіла коду з'явиться перша LD- мережа з одним контактом, одною котушкою і двома шинами живлення:



LD- мережу можна завжди розширити за рахунок додаткового контакту вставленого зліва або справа від існуючого контакту, а також котушки, яка вставляється тільки справа від контакту.

Щоб вставити в LD- мережу інший стандартний об'єкт:

- позначте в ній існуючий контакт або котушку;
- в панелі інструментів клацніть по іконі Contact right (Контакт праворуч) і контакт з'явиться праворуч або по іконі Contact left (Контакт ліворуч), щоб контакт з'явився ліворуч;

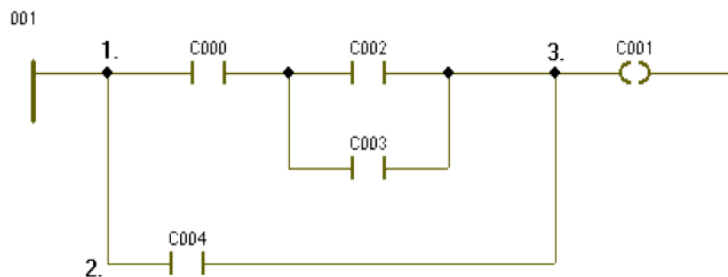
Якщо в панелі інструментів клацнути по іконі Coil right (Котушка праворуч), праворуч з'явиться котушка.

Щоб вставити елементи LD-мови паралельно існуючим контактам і котушкам необхідно:

- позначити лівою клавiшею миші контакт чи котушку;
- в панелі інструментів клацнути по іконі Parallel (З'єднати паралельно), щоб з'явився новий елемент нижче вибраного, або по іконі Add contact/coil above (Додати контакт/ котушку зверху), аби новий елемент з'явився вище вибраного.

Щоб вставити в LD-мережу паралельні гілки треба:

- в панелі інструментів клацнути лівою клавiшею миші по іконі Branch (Вставити гілку), щоб активізувати режим редагування LD-гілок.
- клацнути лівою клавiшею миші по лінії сполучення між двома об'єктами, де бажано почати паралельну гілку (крок 1);



- просуваючи мишу вперед або назад, нарисувати нову сполучну лінію у вільну зону (крок 2);
- клацнути лівою клавiшею миші на рівні майбутнього розташування нового мовного об'єкта;
- перемістити мишею курсор до бажаного кінця паралельної гілки (крок 3);
- клацнути лівою клавiшею миші, щоб встановити зв'язок ліній.

Для вставлення у LD-мережу шин живлення, якщо їх немає:

- у LD- мережі позначити елемент, біля якого бажано зробити вставку;
- в панелі інструментів клацнути по іконі Left powerrail (Ліва шина живлення), щоб вставити ліву шину живлення, або по іконі Right powerrail (Права шина живлення), щоб вставити праву шину живлення.

Для видалення шини живлення позначте її лівою клавiшею миші і натисніть "Delete" на клавіатурі комп'ютера.

Окремі LD- мережі можна з'єднати між собою. Ліві шини живлення сполучаються тільки з лівими, а праві - тільки з правими.

Для з'єднання LD- мереж:

- в панелі інструментів клацнути лівою клавiшею по іконі Connect (З'єднати);
- позначити курсором першу шину живлення;
- клацнути послідовно по інших шинах живлення - і вони з'єднуються автоматично.

Для створення декількох паралельних мереж однакового розміру і вже з'єднаних одна з одною необхідно нижче лівої шини живлення попередньої мережі встановити мітку і в панелі інструментів клацнути по іконі Network (Мережа). Зробити це потрібно стільки разів, скільки додаткових мереж необхідно вставити.

Розмір LD-мережі можна змінити, для цього необхідно:

- у підменю Layout (Формат) вибрати пункт Contact Width (Ширина контакту). З'явиться діалог Contact Width (Ширина контакту);

- ввести нове значення ширини мережі і натиснути ОК. Після коригування розміру всі нові LD-мережі мають нову ширину.

Можна вирівняти LD-мережі, якщо їх праві шини живлення по різному розташовані. Редактор забезпечує дві можливості вирівнювання правих шин живлення:

1. Щоб вирівняти шини живлення за самою правою шиною треба:

- установити курсор на саму праву шину живлення і клацнути лівою клавішею миші;

- відкрити підменю Edit (Редагування), вибрати і активізувати пункт Stretch/ Compress > Arrange Powerrails (Розтягнення/ Стиснення > Видалення шин живлення). Усі шини живлення змістяться до рівня позначеної шини.

2. Щоб вирівняти шини живлення за міткою у робочому листку (вирівнювання здійснюється тільки зміщенням шин праворуч) треба:

- клацнути лівою клавішею миші у необхідній точці робочого листка;

- відкрити підменю Edit (Редагування), вибрати і активізувати пункт меню Stretch/ Compress > Arrange Powerrails (Розтягнення/стиснення > Видалення шин живлення). Усі шини змістяться на рівень встановленої мітки.

В LD- мережу можна викликати функції та функціональні блоки, які є елементами FBD- мови, тобто в тілі коду LD- програми змішуються елементи LD- і FBD- мови.

Функцію або функціональний блок можна вставити в будь-якому місці робочого листа, а потім з'єднати їх з іншими елементами. Можна також вставити функцію або функціональний блок безпосередньо в існуючу LD-мережу.

Найзручніший шлях встановлення FBD функцій або функціональних блоків в LD- програму - це використання Edit Wizard (Майстра редагування).

- Позначте лівою клавішею миші місце на LD- мережі, де бажаєте вставити функцію або функціональний блок;

- в панелі інструментів натисніть ікону Edit Wizard (Майстра редагування);

- у вікні, що з'явиться, виберіть зі списку Group потрібну функцію або функціональний блок і двічі клацніть по ньому лівою клавішею миші, щоб вставити в позначене місце.

В LD- мережі можна змінити властивості контактів і котушок.

Для цього необхідно:

- подвійно клацнути по контакту або котушці, що змінюється;

- у діалоговому вікні Contact/ Coil Properties (Властивості контакт/ котушка на сторінці Contact (Контакт) змінити тип контакту або котушки, використовуючи перемикачі Contact (Контакт) і Coil (Котушка), а також можливості вікна Type (Тип).

Для оголошення змінних контактів або котушок існує дві можливості:

- використання змінних, які вже декларовані в робочому листку змінних;

- використання змінних, які ще не декларовані в робочому листку змінних.

Щоб змінити типове ім'я контакту або котушки на вже існуюче треба:

- подвійно клацнути по контакту/котушці, для якого/якої бажано оголосити вже декларовану змінну. З'явиться діалогове вікно Contact/Coil Properties (Властивості Контакт/ котушка).

На сторінці Contact (Контакт) список змінних у полі Name (Ім'я) містить усі локальні або глобальні змінні, які були раніше декларовані у відповідних локальних або глобальних робочих листках змінних;

- у вікні списку змінних Name (Ім'я) клацнути по імені бажаної змінної, і позначене ім'я увійде у перший рядок текстового поля змінних;
- натиснути ОК, і вибране ім'я змінної з'явиться у робочому листку тіла коду, як ім'я позначеного контакту/ котушки.

Щоб оголосити нову змінну для контакту або котушки:

- подвійно клацнути по контакту/котушці, для якого/якої бажано декларувати нову змінну. З'явиться діалогове вікно Contact/Coil Properties (Властивості контакт/ котушка);

- у полі Name (Ім'я) ввести нове ім'я;

- вибрати для змінній контакт/котушка відповідні установки Usage(Вжити), Data Type(Тип Даних), I/O address(Адреса В/В);

- у полях Local Variable Groups (Групи локальних змінних) і Global Variable Groups (Групи глобальних змінних) вибрати групу змінних Default (За замовчуванням), в яку бажано вставити опис нової змінної, і клацнути лівою клавішею миші.

- Після вибору компетенції клацнути ОК. Змінна з'явиться у робочому листку тіла коду, як ім'я контакту/ котушки, а її декларація автоматично буде вставлена у відібрану групу робочого листка сітки змінних.

У разі глобальної змінної декларація вставляється у сітку локальних змінних, використовуючи ключове слово VAR_EXTERNAL і в таблицю глобальних змінних робочого листа, використовуючи VAR_GLOBAL.

Щоб вставити і декларувати нову змінну, яка не є контактом/ котушкою необхідно:

- у робочому листку тіла коду лівою клавішею миші позначити місце вставки нової змінної;

- в панелі інструментів клацнути по іконі Variable (Змінна) і у діалоговому вікні Variable Properties (Властивості змінної), що з'явиться, зробити все те, що робилося при оголошенні змінної контакту або котушки.

Після закриття діалогового вікна змінна з'явиться у робочому листку тіла коду, а її декларація автоматично вставиться у сітку відповідних змінних.

Біля лівої шини живлення LD- мережі можна вставити коментарі.

Для цього треба:

- подвійно клацнути по лівій шині живлення, з'явиться діалог Comment (Коментар);

- ввести необхідний коментар і натиснути ОК.

Розроблення коду програми

Розробити LD-мовою програму управління роботою двигуна, що вмикається триразовим натисканням стартової кнопки, а вимикається автоматично за 20с роботи.

Покроково нагадаємо, як створити першу LD-мережу, як оголосити властивості об'єктів, що автоматично з'являться з першою LD-мережею, як вставити і з'єднати функціональний блок у робочому листку тіла LD- коду за допомогою Edit Wizard (Майстер редагування), як вставити і з'єднати контакт у робочому листку коду програми, як оголосити властивості окремих контактів і котушок, як вставити другу LD-мережу і надрукувати коментар мережі.

Вставка першої LD- мережі:

- клацнути лівою клавiшею миші у зручному місці робочого листка для установлення мітки першої вставки;

- в панелі інструментів клацнути по іконі Network (Мережа) і в установленому місці з'явиться LD- мережа 001 певної ширини з контактом і котушкою.



Оголошення властивостей змінних, які автоматично з'являються з першою LD- мережею:

- двічі клацнути лівою клавiшею миші по контакту C000, щоб оголосити змінну, яка має запустити двигун. У діалоговому вікні Contact/Coil Properties (Властивості Контакт/Котушка), що з'явиться, замінити типове ім'я змінної C000 на Motor_Start;

- у рядку Data Type (Тип даних) вибрати тип BOOL, а зі списку Usage (Вжити) – VAR.

Таким чином, змінна оголошується як локальна, а тому вона може використовуватися тільки в ROU проекту.

Щоб призначити зовнішній змінній Motor_Start фізичну адресу симулятора модуля вводу/виводу ПЛК:

- у рядку I/O address (Адреса вводу/виводу) ввести %IX0.0, тут 0.0 – позначає перший модуль вводу і його перший канал; I – фізичний вхід; X – одинбітовий розмір змінної;

- зі списку Local Variable Groups: вибрати Default;

- клацнути ОК, підтверджуючи діалог Contact/Coil Properties (Властивості Контакт/Котушка). Після цього змінна є оголошеною і вставлена у відповідний список, а на робочому листку проекту замість C000 з'явиться ім'я Motor_Start.

Двічі клацніть по котушці C001, з'явиться діалогове вікно Contact/Coil Properties (Властивості Контакт/Котушка);

- у діалоговому вікні Contact/Coil Properties (Властивості Контакт/Котушка), що з'явиться, замінити типове ім'я змінної C001 на Motor;

- у рядку Data Type (Тип даних) вибрати тип BOOL, а зі списку Usage (Вжити) – VAR.

Для призначення зовнішній змінній Motor фізичної адреси симулятора модуля вводу/виводу I/O PLC у рядку I/O Address введіть %QX0.0,

тут Q – позначає фізичний вихід; 0.0 – перший вихідний модуль і його перший канал; X – одинбітовий розмір змінної;

- оскільки двигун має працювати деякий час безперервно після його увімкнення, у рядку Type (Тип) виберіть котушку –(S)– (увімкнути вихід і зафіксувати);

- зі списку Local Variable Groups: вибрати Default і клацнути ОК, щоб завершити оголошення змінної і вставити її над котушкою і у відповідному списку змінних.

Вставка в LD- мережу лічильника натисків стартової кнопки:

Оскільки двигун має увімкнутися після триразового натискання стартової кнопки вставимо в LD- мережу лічильник.

- Клацніть по змінній Motor_Start, щоб установити лічильник одразу за контактом;

- в панелі інструментів лівою клав'яшею миші активізуйте ікону Edit Wizard (Майстер редагування) і у таблиці Group (Група), що з'явиться на екрані, двічі клацніть по блоку STU;

- у діалоговому вікні Variable properties (Властивості змінної) змініть типове ім'я STU-1 на Motor_Count і натисніть ОК. На позначеному місці LD-мережі з'явиться лічильник;

- в панелі інструментів клацніть по іконі Edit Wizard (Майстер редагування), щоб сховати бібліотеку Майстра редагування.

Для визначення параметрів лічильника двічі клацніть по блакитній точці PV-входу, з'явиться діалогове вікно Variable properties (Властивості змінної);

- у рядку Name (Ім'я) введіть константу INT#3, щоб двигун (Motor) спрацював після триразового натискання стартової кнопки, тут INT – означає ціле число; # - означає константу; 3 – фактичне значення константи;

- клацніть ОК і на PV-вході лічильника з'явиться INT#3;

Двічі клацніть по зеленій точці CV-виходу лічильника, з'явиться діалогове вікно Variable properties (Властивості змінної);

- в рядку Name (Ім'я) введіть Pressed - це змінна, яка запам'ятовує поточне значення лічильника;

- у рядку Date type (Тип даних) виберіть INT, оскільки змінною є число, а у рядку Usage (Вжити) – VAR;

- клацніть ОК і у робочому листку поряд з CV-виходом з'явиться Pressed.

Вставка контакту скидання лічильника:

- клацніть на функціональному блоці STU по синій точці Reset-входу;

- в панелі інструментів клацніть по іконі Contact left (Контакт зліва) і на вході лічильника з'явиться контакт C002;

- активізуйте в панелі інструментів кнопку Connect (З'єднання);

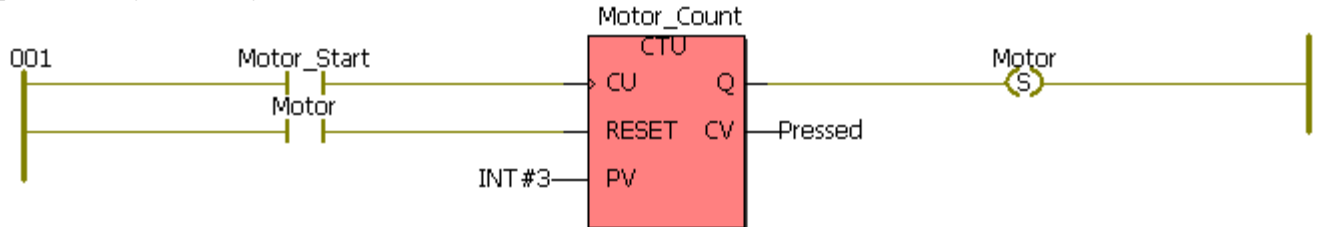
- клацніть по синій точці лінії контакту C002, перемістіть її курсором ліворуч до шини живлення 001 і повторно натисніть ліву клав'яшу миші, щоб з'єднати лінії між собою;

- в панелі інструментів клацніть по іконі Mark (Позначка), щоб деактивізувати кнопку Connect (З'єднання);

Оголошення властивостей контакту скидання лічильника.

Двічі клацніть по контакту C002 і у діалоговому вікні Contact/Coil Properties (Властивості Контакт/котушка) зі списку вже оголошених змінних, виберіть Motor.

Оскільки змінна була визначена раніше, немає потреби конкретизувати її властивості. У рядку Type (Тип) виберіть прямий контакт і клацніть ОК. На робочому листку з'явиться змінна Motor замість C002:



Вставка другої LD- мережі

Клацніть лівою клавішею миші нижче першої LD- мережі, на робочому листку з'явиться мітка установлення нової мережі;

- в панелі інструментів клацніть по іконі Network (Мережа) і на екрані з'явиться друга LD- мережа;

- двічі клацніть по контакту C003 для оголошення його властивостей;

- у діалозі Contact/Coil Properties (Властивості Контакт/Котушка) із списку вже оголошених змінних виберіть Motor і клацніть ОК для зміни у LD- мережі імені контакту C003 на Motor .

Для вставлення таймера, що керує тривалістю роботи двигуна:

- клацнути лівою клавішею миші по контакту Motor;

- в панелі інструментів натиснути на ікону Edit Wizard (Майстер редагування), щоб на екрані з'явилося вікно Group (Група);

- у вікні Group (Група) вибрати Function blocks (Функціональні боки);

- зі списку блоків курсором вибрати таймер TON і двічі клацнути лівою клавішею миші;

- у рядку Name (Ім'я) діалогова вікна Variable properties (Властивості змінної) ввести M_Time і натиснути кнопку ОК;

- клацнути по іконі Edit Wizard (Майстер редагування), щоб приховати вікно Group (Група);

- для визначення часу роботи таймера лівою клавішею миші двічі клацнути по блакитній точці його RT-входу;

- у діалоговому вікні Variable properties (Властивості змінної), що з'явиться, у рядку Name (Ім'я) ввести T#20s і натиснути ОК.

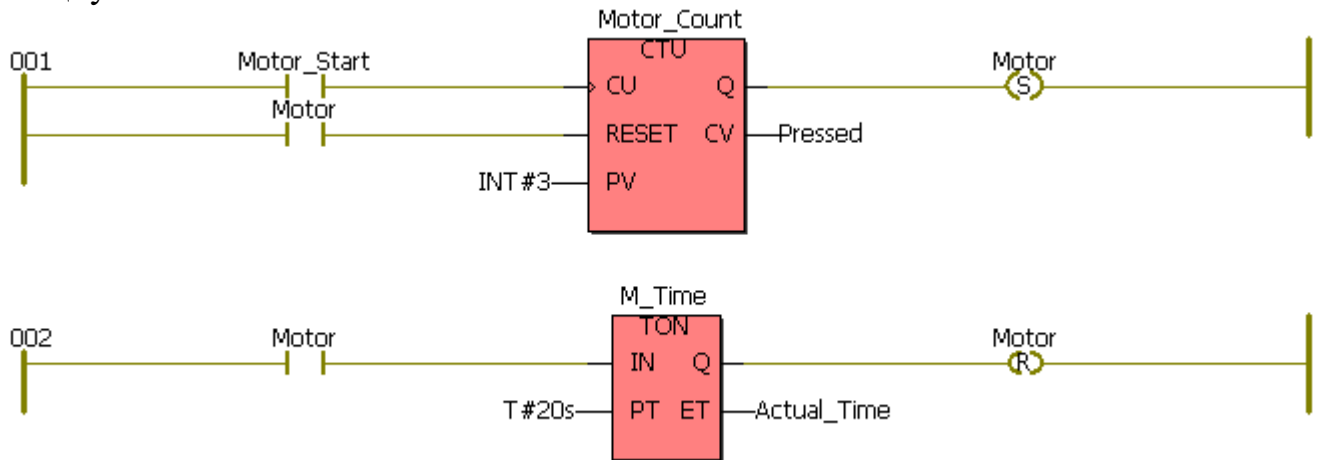
Тут T – позначає префікс часу; # - означає константу; 20s – час роботи двигуна (20с);

Для визначення фактичного часу роботи таймера двічі клацнути по зеленій точці ET-виходу таймера;

- у діалоговому вікні Variable properties (Властивості змінної), що з'явиться, у рядку Name (Ім'я) ввести Actual_Time (Фактичний час), як ім'я локальної змінної;

- зі списку Data_Type (Тип даних) вибрати Time (Час), клацнути ОК і на ET-виході таймера з'явиться Actual_Time;

- двічі клацнути по котушці C004, відкриється діалогове вікно Contact/Coil Properties (Властивості Контакт/Котушка);
- для зупинення двигуна вибрати курсором зі списку змінних ім'я Motor, у вікні Type (Тип) - котушку –(R)– (вимкнути вихід і зафіксувати) і клацнути ОК.



Присвоєння імені розробленої мережі:

- двічі клацнути по лівій шині живлення першої LD- мережі, з'явиться діалогове вікно Comment (Коментар);
- у діалоговому вікні Comment (Коментар) надрукувати Circuit (Схема) і клацніть Font>> (Шрифт);
- для зміни властивостей шрифту у вікні, що з'явиться, вибрати блакитний колір і розмір 20;
- клацнути ОК у вікні Font (Шрифт) і у вікні Comment (Коментар). Над розробленою мережею з'явиться ім'я Circuit.

Компілювання проекту

Тепер, коли редагування закінчено, необхідно зробити компілювання проекту. Протягом компілювання вміст робочих листків перетворюється у спеціальний код, який може бути виконаний ПЛК.

Оскільки у цьому прикладі немає ПЛК, треба активізувати симулятор. Для цього:

- у дереві проекту клацнути правою клавішею миші по теці Resource (Ресурс) і в контекстному меню Settings (Установки) у вікні Resource settings for IRC_40 (Установки ресурсу для IRC-0), що з'явиться, активізувати Simulation 1 (Симуляція 1), якщо це необхідно, і закрити діалог, клацнувши ОК;
- в панелі інструментів клацнути по іконі Make (Утворювати);
- у вікні Build (Конструкція) внизу екрана відобразяться помилки (Error) і попередження (Warning), якщо будуть знайдені протягом компіляції.

Помилки синтаксичного або структурного характеру перешкоджають завершенню процесу компіляції.

Попередження вказують на потенційні проблеми, подібно змінній, якої немає, а вона використовується. Попередження не перешкоджають завершенню процесу компілювання. Попередженнями можна нехтувати, а помилки потрібно виправити.

Для виводу на екран списку знайдених помилок у вікні Build (Конструкція) клацнути по Error (Помилки), а для виводу на екран списку попереджень у вікні Build (Конструкція) клацнути по Warning (Попередження).

Щоб виправити усі помилки:

- двічі клацнути по вказаній помилці або попередженню, відкриється відповідний робочий листок;

- виправити усі помилки і попередження і повторно компілювати проект, клацнувши в панелі інструментів по іконі Make (Утворювати);

Тільки після цього можна завантажувати програму до симулятора.

Завантаження проекту до симулятора

Зв'язок з ПЛК або симулятором здійснюється завдяки діалогу Resource (Ресурс), який керує контролером. Щоб здійснити зв'язок необхідно:

- клацнути по іконі Project Control Dialog (Діалогове управління проектом), з'явиться діалогове вікно Resource (Ресурс) для управління ПЛК або симулятором;

- натиснути кнопку Download (Завантаження), з'явиться відповідне діалогове вікно;

- продублювати у лівій частині вікна натискання на кнопку Download і проект завантажиться у пам'ять симулятора;

- успішний процес завантаження проекту супроводжується бігом синьої стрічки внизу екрана.

Налагодження проекту

Робочі листки проекту можуть перемикатися з режиму редагування на режим налагодження і навпаки, за допомогою ікони Debug on/off (Налагодження В/В) в панелі інструментів. Коли режим налагодження активізовано, стан і поточні значення змінних на робочому листку проекту показані різними кольорами: блакитний – хибний, червоний – істинний.

Режим налагодження використовується для знаходження помилок програмування і переконання, що програма ПЛК працює правильно. Для запуску програми у діалозі управління Resource (Ресурс) натиснути кнопку Cold (Холодний).

- Для виводу на екран симулятора модулів I/O (Вводу/виводу) клацнути по Demoio - Driver, що видно внизу монітора;

- розмістити симулятор таким чином, щоб проект на робочому листку не був прихований;

- тричі клацнути по зеленому віртуальному світлодіоду з адресою IX0.0 і спостерігати реакцію схеми в робочому листку проекту.

Motor (Двигун) почне працювати після того, як з'явиться цифра три біля Pressed, тому що поточне значення CV- виходу на блоці Motor_Count досягає встановленого значення на PV- вході того ж блока.

Коли котушка Motor у мережі 001 вмикає двигун, вмикається контакт Motor у мережі 002 і таймер M_Time починає відлік часу протягом 20с. Тривалість руху двигуна продовжується доки час, що фіксує Actual_Time (Фактичний час) біля ET-виходу таймера, не співпаде з уставкою біля PT-

входу (T#20s). По завершенню роботи таймера котушка –(R)– у мережі 002 вимикає двигун (Motor), що у мережі 001.

Оперативне редагування

Оперативне редагування можливе без зупинки програми, яка реалізується на ПЛК або симуляторі. Цей режим викликається, клацаючи в панелі інструментів по іконі Patch POU (Вставка POU) .

Коли використовується режим Patch POU (Вставка POU), зміни, що зроблені в кодї програми автоматично завантажуються до ПЛК або симулятора. Протягом всього часу роботи в режимі Patch POU (Вставка POU) виконання коду програми не припиняється.

Як приклад роботи в режимі Patch POU (Вставка POU), вставимо у програму змінну Emergency_Stop (Непередбачена зупинка). Активізація Emergency_Stop (Непередбачена зупинка) негайно зупинить двигун.

- Клацніть по іконі Debug on/off (Налагодження В/В), щоб перевести робочий листок коду програми в режим редагування. Проте, ресурс так само, як і реальний контролер, працює;

- установіть курсор на робочому листку коду програми нижче шини живлення 002 і клацніть лівою клавішею;

- в панелі інструментів клацніть по іконі Contact Network (Мережа контактів), і на робочому листку з'явиться нова LD- мережа 003;

- двічі клацніть по контакту C005, відкриється діалог Contact/Coil Properties (Властивості Контакт/Котушка). У рядку Name (Ім'я) замініть типове ім'я C005 на Emergency_Stop (Непередбачена зупинка);

- клацніть ОК, автоматично відкриється сторінка діалогу Common (Загальний);

- з вікна списку Usage (Вжити) виберіть VAR_EXTERNAL, декларуючи таким чином Emergency_Stop (Непередбачена зупинка), як глобальну змінну.

Для непередбаченої зупинки двигуна використаємо другий канал симулятора I/O:

- у рядку I/O Address (Адреса В/В) введіть % IX0.1. Клацніть ОК, у мережі 003 з'явиться змінна Emergency_Stop замість C005;

- двічі клацніть по контакту C006, з'явиться діалог Contact/Coil Properties (Властивості Контакт/Котушка);

- з вікна списку Type (Тип) виберіть –(R)–, а з вікна списку Variable (Змінна) – Motor. Клацніть ОК, і на робочому листку з'явиться третя відкоригована LD- мережа.

Тепер, коли код програми змінено, використаємо режим Patch POU (Вставка POU), щоб компілювати змінні і завантажити їх до симулятора без його зупинки.

В панелі інструментів клацніть по іконі Patch POU (Вставка POU), змінений код проекту компілюється і завантажиться у симулятор I/O.

Після того, як режим Patch POU (Вставка POU) успішно завершиться, робочий листок буде автоматично переведений до оперативного режиму.

Клацніть по DemoIO-Driver, щоб відкрити симулятор I/O;

- тричі подвійним клацанням по нульовому віртуальному світлодіоду нульового модуля I/O змінити стан біту;

- використовуючи новий контакт Emergency_Stop (Непередбачена зупинка), негайно зупиніть двигун Motor, клацаючи по другому світлодіоду 1-го нульового модуля вводу.

Протягом оперативного редагування можна використовувати можливості вікна спостереження.

Вікно спостереження змінних – це могутній інструмент, що дозволяє легко вставити різні змінні у список і спостерігати динаміку їх поведінки. Як тільки змінна додана у вікно спостереження, не потрібно відкривати відповідний робочий листок для контролю за нею. Таким чином, можна зосередитися лише на змінних, які потрібні для аналізу роботи програми.

Клацніть по іконі Debug on/off (Налагодження В/В), щоб перевести робочий листок в оперативний режим;

- в панелі інструментів клацніть по іконі Watch Window (Вікно спостереження), унизу екрана з'явиться вікно спостереження;

- установіть курсор на змінну Motor_Start і клацніть правою клавішею миші;

- у контекстному меню, що з'явиться, виберіть Add to Watch Window (Додати до вікна спостереження) і клацніть лівою клавішею миші, змінна Motor_Start буде внесена до списку;

- продублюйте такі самі дії зі змінними Pressed (Натиснути) і Actual_Time (Фактичний час). У вікні спостереження з'явиться список з трьох змінних;

- натисніть Demoio-Driver для виклику симулятора I/O;

- тричі подвійно клацніть по нульовому світлодіоду нульового модуля I/O, щоб запустити програму, і спостерігайте стан змінних одночасно на робочому листку і у вікні спостереження.

Завдання для самостійної роботи

1. Розробити програму керування роботою двох двигунів, які після запуску одного з двигунів безперервно по черзі вмикаються і вимикаються. Тривалість роботи двигунів відповідно 5 і 10 секунд. Початковий запуск першого двигуна здійснюється одноразовим натисканням пускової кнопки, а початковий запуск другого двигуна – дворазовим. При цьому двигуни можна у будь-яку мить зупинити, а кожне вмикання двигунів підраховується лічильником. Коли кількість вмикань двигунів досягає десяти, їх робота автоматично зупиняється.

Налагодження програми здійснити, використовуючи вікна спостереження змінних і перехресне довідкове вікно.

2. Розробити програму нескінченного миготіння лампи, в якій довжина імпульсу у двічі більше паузи.

Контрольні запитання

1. Як розробляється код програми?
2. У чому різниця між прямим і інвертованим контактами?
3. Коли асоційована змінна 'Set' котушки приймає істинне значення?
4. Що трапляється з асоційованою змінною "Reset" котушки, коли стан лівого сполучення дорівнює TRUE?

1. Як оголошуються змінні при створенні додатку користувача?
2. Як оголосити функціональний блок у LD-програмі?
3. Як здійснюється компіляція і завантаження проекту в симулятор?
4. Як відбувається налагодження роботи програми?