

Trio Motion Technology Ltd.
Shannon Way, Tewkesbury,
Gloucestershire. GL20 8ND
United Kingdom
Tel: +44 (0)1684 292333
Fax: +44 (0)1684 297929

1000 Gamma Drive
Suite 206
Pittsburgh, PA 15238
United States of America
Tel: +1 412.968.9744
Fax: +1 412.968.9746

Tomson Centre
118 Zhang Yang Rd., B1701
Pudong New Area, Shanghai,
Postal code: 200122
P. R. CHINA
Tel/Fax: +86-21-58797659



Doc No.: AN-241

Version: 1.3

Date: 23 March 2012

Subject: Getting started with IEC 61131

APPLICATION NOTE

www.triomotion.com

1. Introduction

The industry standard IEC 61131 runtime is supported on some controllers¹, it is an easy to use full function PLC programming system. This enables you to program in Structured Text (ST), Ladder Diagrams (LD), Functional Block Diagram (FBD) and Structured Flow Chart (SFC)² all with integrated diagnosis. Of course you can always use TrioBASIC programs interactively with any of the other languages. The programming languages and the project structure conforms to the international standard IEC 61131-3.

The IEC 61131 runtime kernel allows multiple PLC tasks to be run and has access to all I/O and distributed I/O. The IEC 61131 programs have access to specific functions blocks containing Trio motion commands, axis parameters, VR and Table memory.

By default the IEC 61131 runtime engine will operate for 1 hour after start-up. This is so that it can be trialled free of charge. The runtime can be permanently enabled by purchasing the P877 (IEC 61131 Runtime Feature Enable Code).

Programming support for the IEC 61131 languages is included in Motion Perfect Version 3. MPv3 allows programming in the 4 supported IEC 61131 languages and has comprehensive debugging and error reporting.

This guide will provide the fundamental information to start using the Trio Motion Coordinator with the IEC61131 languages.

2. Required Components

The following minimum components are required for this install:

- Motion Perfect v3 version 3.0.1.4725 or newer
- MC464/MC405/MC403 firmware version 2.0177 or newer

Please contact your distributor or Trio Motion Technology for these if they are not available on www.triomotion.com.

¹ Please see product specification for availability

² Instruction List (IL) is not supported

3. Installing

3.1. Motion Perfect v3

3.1.1. Install

Please install Motion Perfect by running the installer; this is usually a file “MotionPerfectV3Setup.msi”. Accept any permission requests from your operating system. For full details on installing and using Motion Perfect please see the Trio Technical Reference Manual.

4. Getting started

This guide covers the initial steps in using the IEC 61131-3 features of Motion Perfect v3 with the Trio MC4 range of Motion Coordinators.

There are two methods to start a project:

1. Create a new project in Motion Perfect v3.
2. Load an example project.

4.1. Creating a new project

The easiest method of starting is to create your own new Motion Perfect v3 project.

- Launch Motion Perfect v3
- Click the connect button and select connect in Sync Mode.
- In the Resolve program differences dialogue, click the icon in the centre to make a new project.

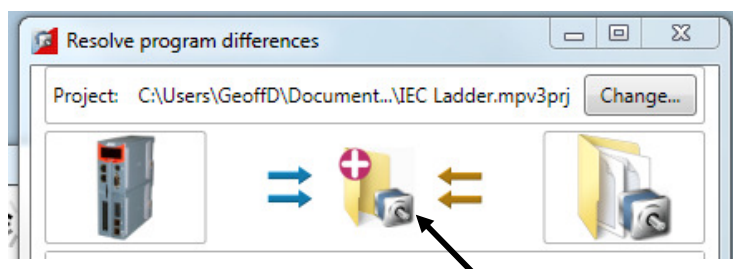


Figure 1: Resolve Project

- Give your project a name, replacing “untitled” and click the Create button

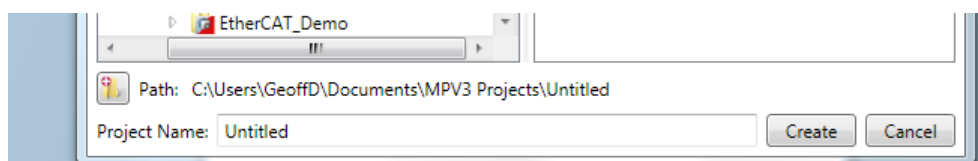


Figure 2: Name the project

- Delete any programs which may already be on the controller. Click the left-pointing arrows as shown.

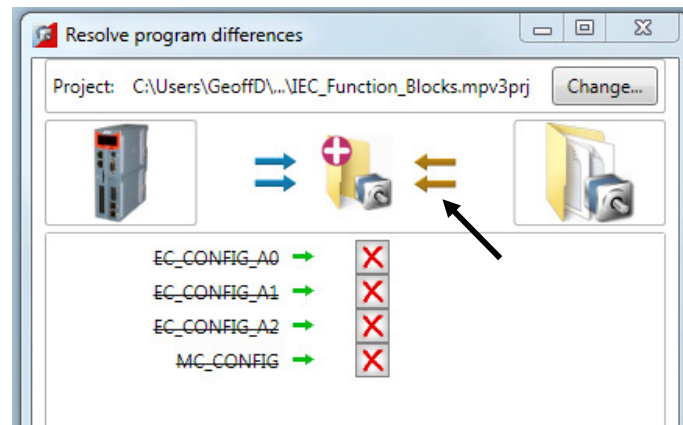


Figure 3: Start with empty controller

- Click the Synchronise Button.
- Select the Window menu above the MPv3 toolbar.



Figure 4: MPv3 menus and toolbar

- Tick the “Controller Tree view”
- Right-Click Programs and select “New”
- Select the program type required and enter a name for your program.

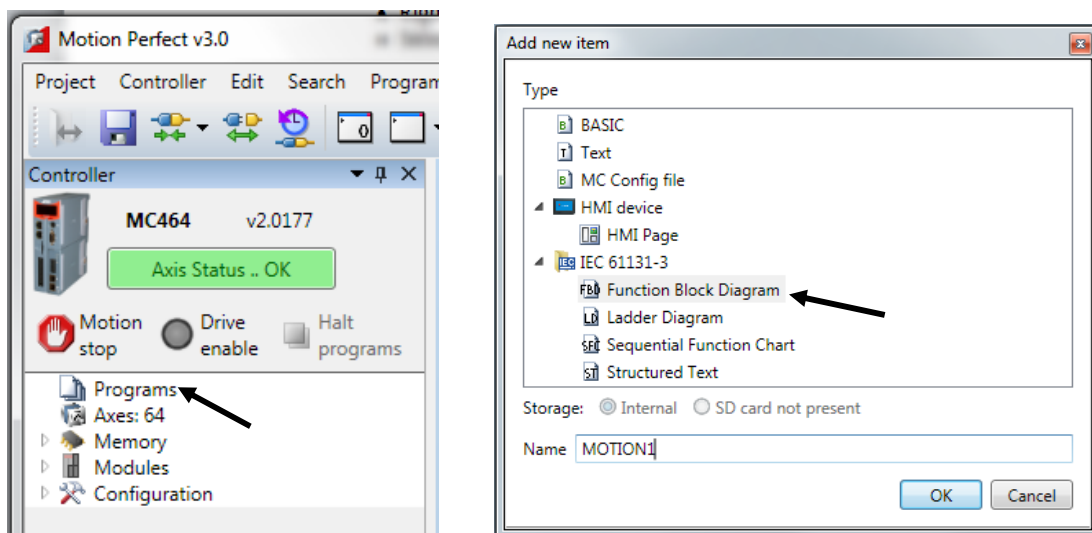


Figure 5: Create a new IEC program

- Click the OK button.
- Now assign your program to a task.
- If there are no tasks already named, click the Add New IEC 61131-3 Task button.
- Name your Task and click OK.

Note: more than one IEC program can exist in one task. When any program in that task is run, then all the other programs in that task will run also.

Select separate tasks for programs that must be run independently or at different PLC cycle times.

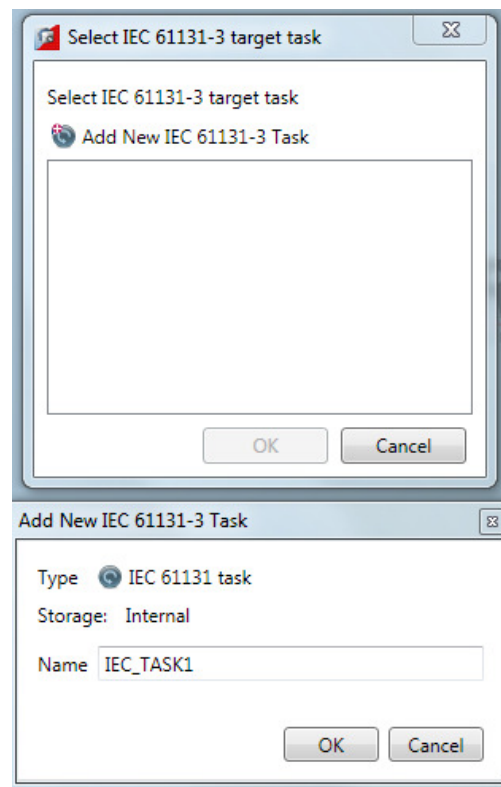


Figure 6: Adding a new task

- Now select the program type. Choose main for a program which runs stand-alone.

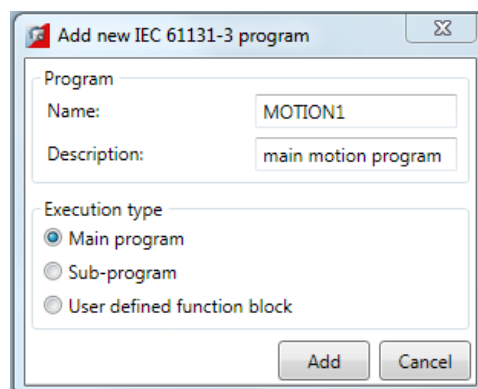


Figure 7: Select the Execution Type

Your new project is now configured and has one program. The program edit window will open automatically after the program variables dialogue box. You do not need to edit program variables at this stage. The following sections discuss how to create and run a simple program of each type.

5. Opening the Toolbox

Each programming method has a set of tools available to it. The tools for the programming method in use can be seen by clicking the Toolbox button in the editor toolbar. The Toolbox window will open initially on the right side of the MPV3 desktop.



Figure 8: Location of Toolbox button

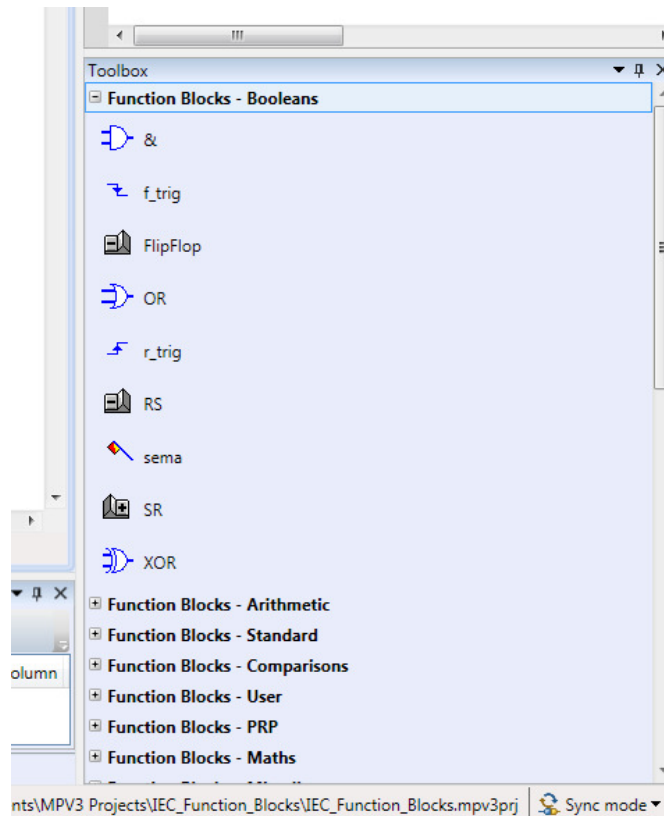


Figure 9: Toolbox for Ladder programming

The tools and functions can be selected and then added to the program with the add function button as described in each section below.

6. Using Inputs and Outputs

Your program will need to use the inputs and outputs of the controller. These must be selected and configured before you can add them to a program.

All the inputs and outputs must be assigned as Global Variables before they can be used in a program.

When an output is defined in the IEC 61131, the runtime engine controls its state. This means that any outputs defined in the Global_Variables cannot be set through TrioBASIC.

6.1. Opening the IEC61131-3 variables window

The variables definition window is opened by selecting the appropriate icon in the IEC 61131-3 edit window.



Figure 10: IEC61131-3 Variables button

The IEC61131-3 variables window will open and locate itself to the left of the edit window.

There are 5 buttons at the top of the variables window.

- Insert New Variable.
- Add New Digital Input.
- Add New Digital Output.
- Add New Analogue Input.
- Add New Analogue Output.

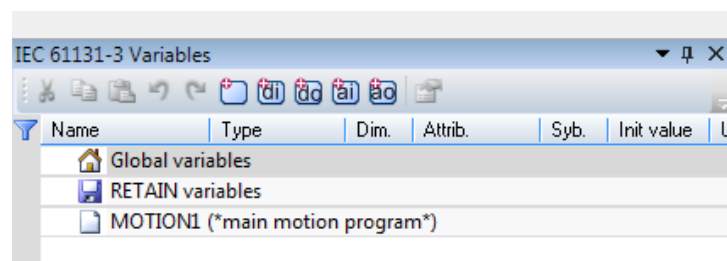


Figure 11: IEC 61131-3 variables toolbar

6.2. Assigning Digital Inputs

IN(0) to IN(15), OP(8) to OP(15) and AIN(32) & AIN(33) are already assigned. To assign additional I/O use the +di and +do buttons.

- Click the +di button.
- Double Click the value next to POINT

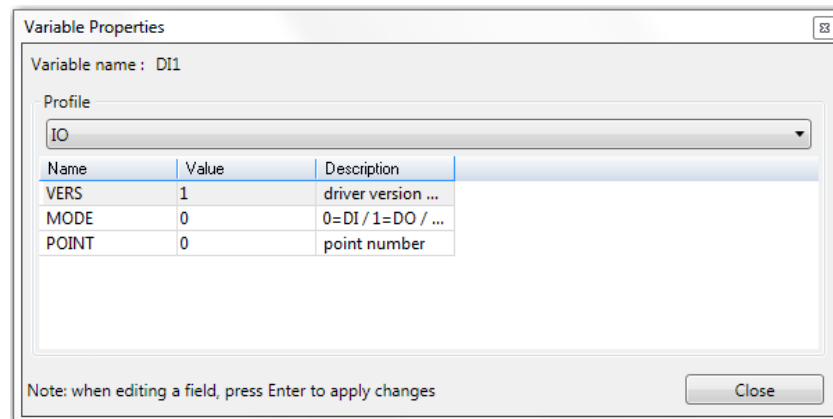


Figure 12: Digital Input Properties

- Type the input number required.
- Press the Enter Key (IMPORTANT).

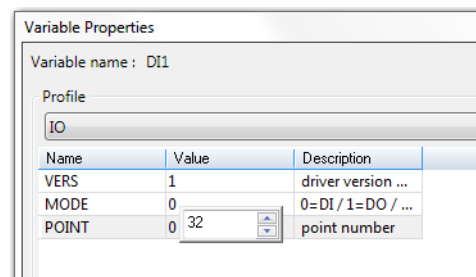


Figure 13: Enter the Input Number

- Click the Close button.
- See the new input assigned in the Global Variables list.

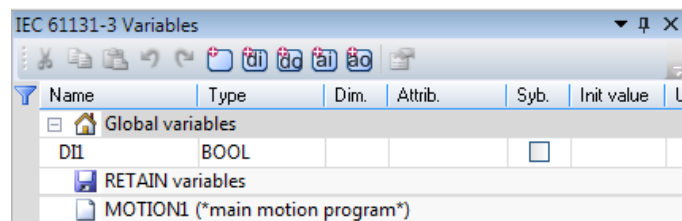


Figure 14: New DI variable added to list

- Change the variable name to one that is more meaningful by double clicking the name in the list.

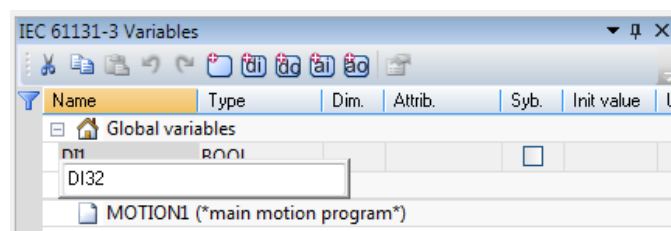


Figure 15: Re-name the variable

6.3. Assigning Digital Outputs

Click the Add New Digital Output button and follow the same process as shown in section 5.2.

6.4. Assigning Analogue Inputs

Click the Add New Analogue Input button and follow the same process as shown in section 5.2.

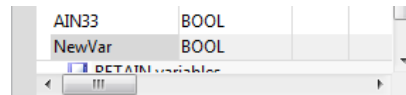
6.5. Assigning Analogue Outputs

Click the Add New Analogue Output button and follow the same process as shown in section 5.2.

6.6. Assigning Global Variables

The global VR variables in the MC464 can be added to the global or local variable list in the IEC 61131-3 program.

- Click on either the Task Variables, or the program name in the variables window.
- Click the Add Variable button at the top of the variables window.



A new variable will appear called NewVar:

- Double click the name NewVar and enter your own name for the variable.
 - Press enter.
 - Now double click the type (where it says BOOL)
 - Select the required variable type to match the type used in the program.
- Note: Any numeric type can be used. The VR value will be converted to that type automatically.*
- Double click the variable type to close the selection window.

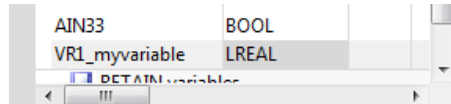


Figure 16: VR variable after name and type is set

The variable must now be assigned to a VR in the controller.

- Right Click the variable name.
- Select Properties.
- Bind Variable to VR.
- Enter the VR index number.
- Click Close.

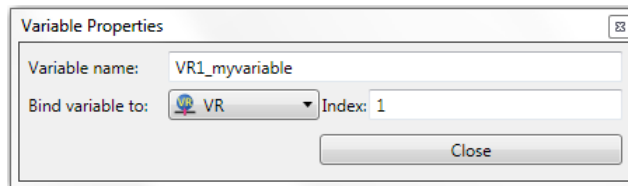


Figure 17: IEC variable is bound to a VR variable

The VR variable is ready to be used in the program.

Note: Task Variables will be assessable to any program running in that task. Local Variables will be known only to that program alone. However the VR they are bound to is still Global in the controller.

6.7. Using digital I/O in Ladder

The following instructions will add an input to a ladder diagram.

- Create a new IEC 61131-3 program of type Ladder Diagram and execution type main.
- Click into the top right corner of the edit page.
- Select Insert Rung from the toolbar.

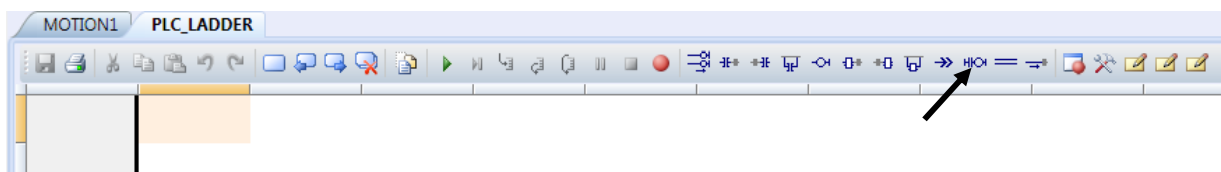


Figure 18: Insert rung

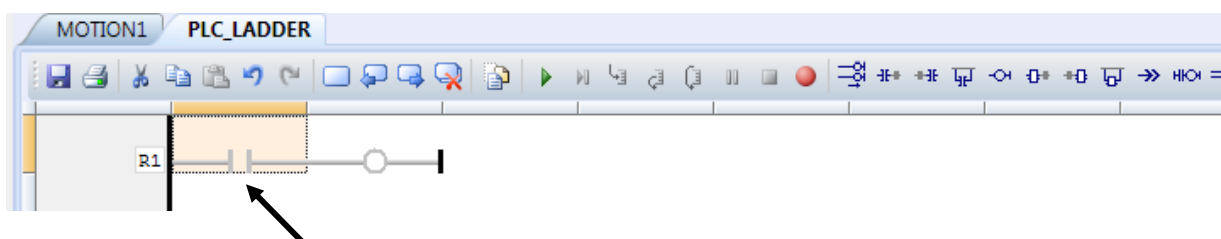


Figure 19: First rung R1

- Double click on the contact.
- Select the required digital input.
- Click the green “tick”.

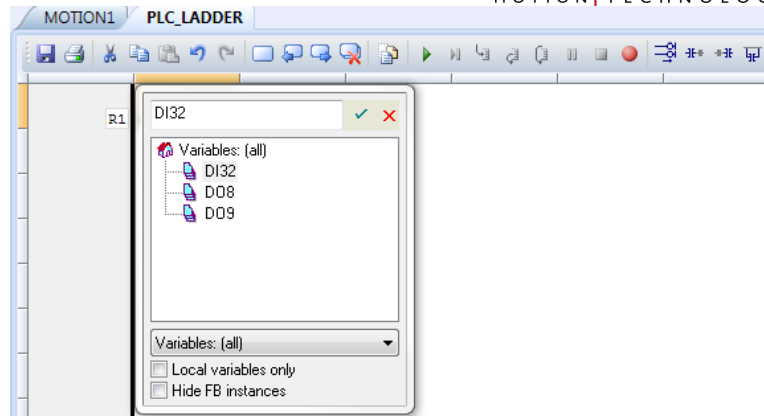


Figure 20: Set contact as digital input

- Double click on the Coil
- Select the required digital output

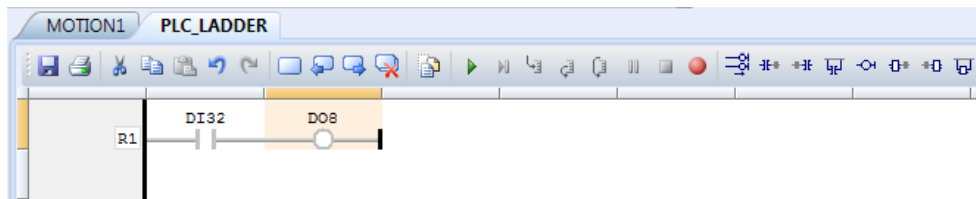


Figure 21: First rung complete

- Click on the contact or coil and use the space bar to toggle NO / NC / pulse operation.

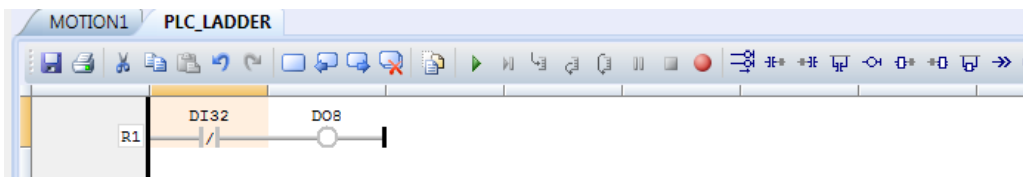


Figure 22: Normally Closed Contact

The first rung is now complete and the ladder program can be run.

6.8. Using digital IO in Function Blocks

Adding Inputs and Outputs is very straight forward. There are 2 ways to do it.

1. Drag and drop

- Open the editor for the FBD being worked on.
- Open the variables window.
- Click on the IN or OP required and while holding down the left mouse button and drag the component on to the diagram.
- Connect the component to the required place in the Function Block Diagram.

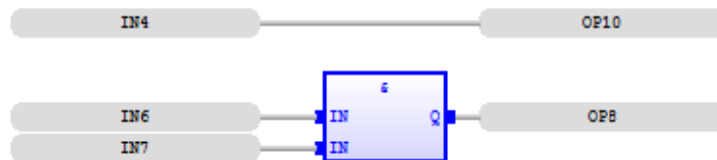


Figure 23: Input/Output on FBD

2. Place and Configure

- Click the Add Variable button in the Editor Toolbar.

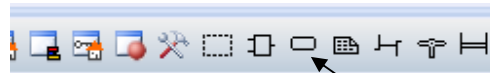


Figure 24: Add Variable Button

- Click on to the Editor page and place the new variable.
- Double Click on the new variable and select the Input or Output required.
- Click the Tick to complete.

6.9. Using digital IO in Structured Text

There are 2 ways to add digital IO to a ST program.

1. Drag and Drop

- Open the editor for the ST program being worked on.
- Open the variables window.
- Click on the IN or OP required and while holding down the left mouse button and drag the component on to the program page. Placing it on the required line.
- Add the connection using the ST programming syntax.

```
OP10 := IN4;
OP8 := IN6 & IN7;
```

2. Type in the lines of Structured Text directly.

6.10. Using digital IO in Sequential Function Chart

To be completed

- To be completed
-

7. Using VRs and Table

VRs and TABLE can be referenced as variables in the variable window or as function blocks TCR_VR, TCR_TABLE, TCW_VR and TCW_TABLE. R is for read and W is for write.

Here is an example in ST.

```
TCW_TABLE( Index:=2000, Value:=TCR_VR( Index:=10 ) );
```

Put the value from VR(10) into the location at TABLE(2000).

8. Using Functions

Functions from the Toolbox can be added to each program according to the programming method in use.

8.1. Setting WDOG in a PLC Ladder

- Find the Trio functions in the toolbox.

All the Trio functions begin with TC for “Trio Controller”.

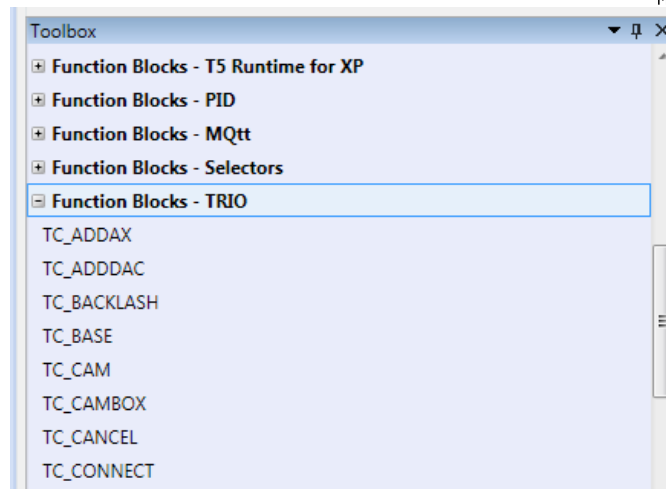


Figure 25: Function Blocks - TRIO

- Select the required function, in this example we are selecting the function block to set the WDOG relay and enable the axes.
- The function block is called TCW_WDOG.

Notice that there is a separate function for reading the state of the WDOG called TCR_WDOG. TCW for write, TCR for read.

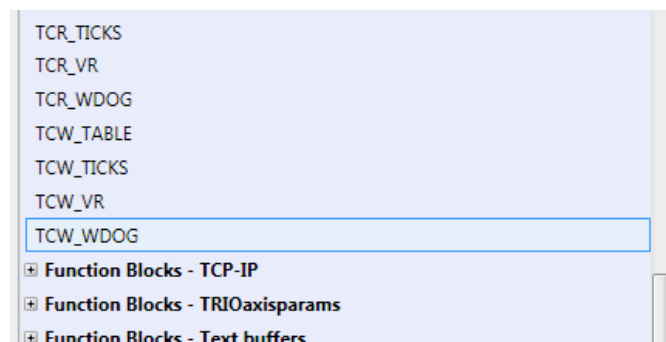


Figure 26: Select TCW_TRIO

- Next, click into the area of the ladder program where you would like this function to be added. Note that it will appear as a new rung.
- Click either the “Insert FB after” or the “Insert FB before” icon in the toolbar.

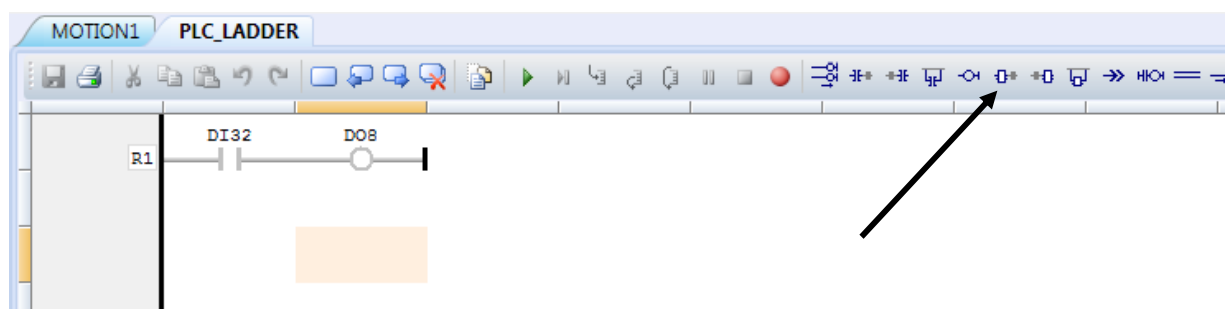


Figure 27: Inserting the Function Block

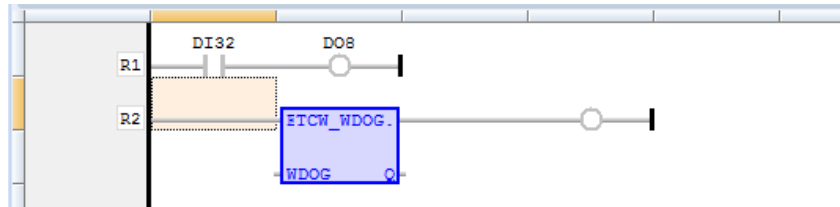


Figure 28: Function Block in place

Note that the actual rung connection has no function, it is simply a place-holder for the function block. The only active connection is the input marked WDOG.

- Make the connection to the WDOG by double-clicking the square to the left of the WDOG input.
- Select the variable from the list.

Note that it is not possible to connect a ladder contact directly to the input.

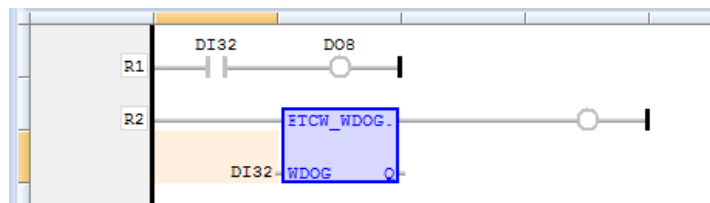


Figure 29: Function Block Input

Note: In system software V2.0165 to 2.0168, the WDOG input has the type DINT. It is therefore necessary to convert a ladder BOOL to DINT with the function block ANY-TO-DINT

- Open Function Blocks - Conversions in the Toolbox.
- Select any_to_dint.
- Select the WDOG function block in the ladder.
- Click the “Insert FB Before” icon in the toolbar.
- Double-Click to the left of the WDOG input and name a new variable.

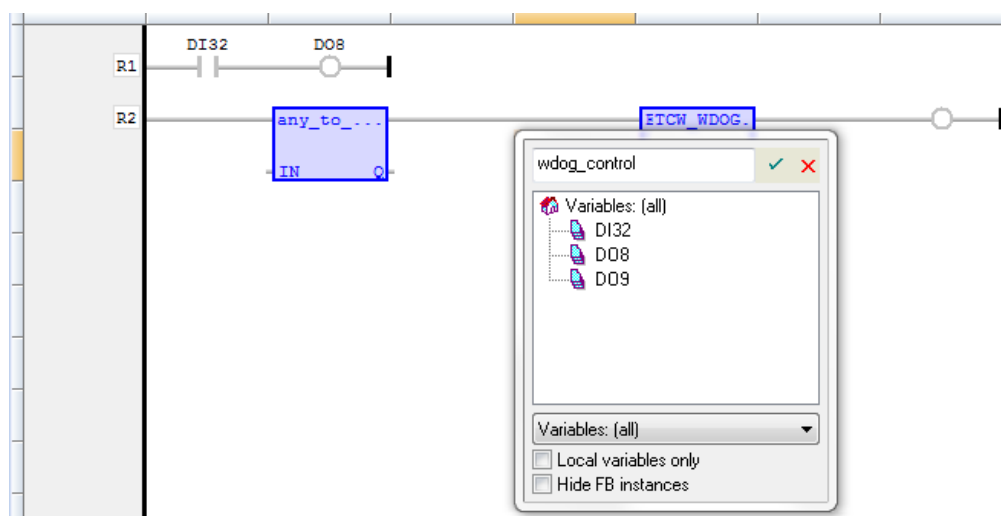


Figure 30: Create a new local variable

- Click the green Tick
- Select variable type DINT

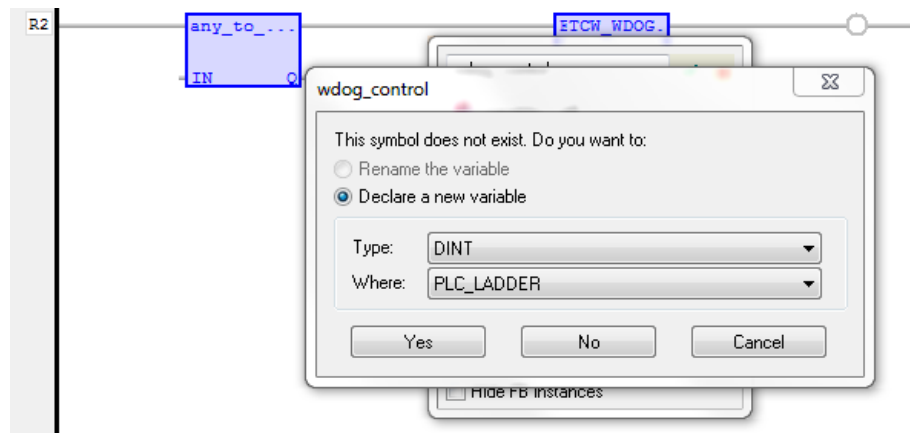


Figure 31: Set the variable type

- Click the Yes button to create the new variable and attach it to the WDOG input.
- Double-Click to the right of the Q output of the any_to_dint.
- Select the variable wdog_control which was just created.
- Click the green Tick.
- Double Click to the left of the IN on the any_to_dint function block.
- Select the Ladder variable required and click the green tick.
- Finally, tidy up the ladder by clicking the Align Coils icon.



Figure 32: Ladder with WDOG output



Figure 33: Two inputs controlling WDOG output

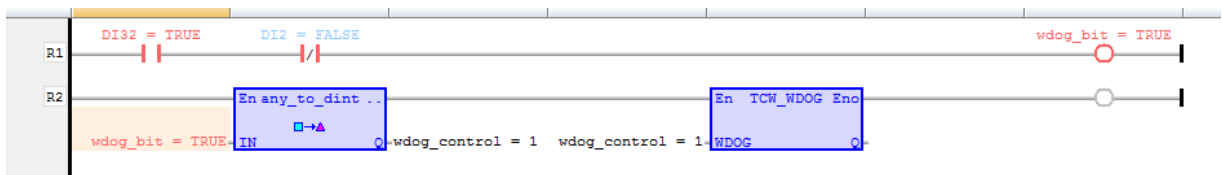


Figure 34: Ladder Running with Debug information showing

9. Appendix I - Un-Supported Functions

The MC464, MC405 and MC403 support only some of the functions which are visible in Motion Perfect Version 3. This is a list of the UN-supported functions.

Random number generation

Malloc

Files

File Management

LibSock

TXB

XML

UNICODE

Accept Missing Handler

DLL Blocks

DLL IOs

Socket V2

UDP

TCP Client

Modbus IOs

MBS TCP

MBS ReadQ

ASI

CANbus

CAN Raw

Event Server

EA

EA Hyster

FT

SP5

NOP C Switch

REG

VFS

Bigendian