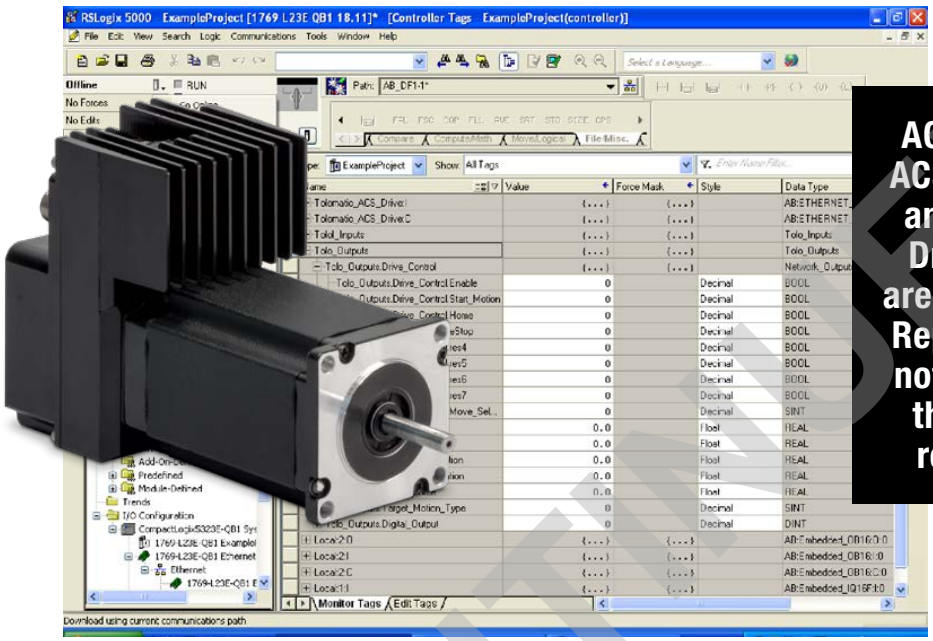
Tolomatic™
EXCELLENCE IN MOTION

Technical Note

ACSI Motor/Drive/Controller with Allen Bradley RSLogix 5000 EtherNet/IP Setup Guide



ACS Servo Drive, ACS Stepper Drive and ACSI Motor/ Drive/Controller are DISCONTINUED. Replacements are not available. Use this manual for reference only.

Contents

1. System Requirements

HARDWARE	MINIMUM REQUIREMENTS
Processor	Intel Pentium II 450 MHz Min Intel Pentium III 733 MHz (or better) recommended
Software Requirements	Windows XP with Service Pack 2 (or above)
RAM	128 MB Minimum 256 MB Recommended
Disk Space	3 GB Free HDD space
PLC	Allen Bradley ControlLogix L3X or above
RSLogix	5000 version 18
Cables	USB CAT 5e

NOTE: Tolomatic's Motion Interface is dependent on .NET 4. Reference the TMI User Guide 3600-4167 for minimum requirements.

2. Cabling

Setup and establish communication between RSLogix and the controller. Connect an Ethernet cable to the PLC controller and to the ACS Drive.

3. Setting up the Tolomatic ACS Drive IP Address

Using the Tolomatic Motion Interface (TMI) program with the ACS, go to Tools -> Ethernet menu selection or press the Configure Ethernet button on the toolstrip. Assign the IP address of the drive (Figure 1). By default the ACS will be set for DHCP mode. A static IP address can be set using the "Ethernet Setup Tool" in TMI or the TCP/IP object (OXF5) in EtherNet/IP. For additional instruction, refer to the "Ethernet Setup Tool" section of the TMI User Guide 3600-4167

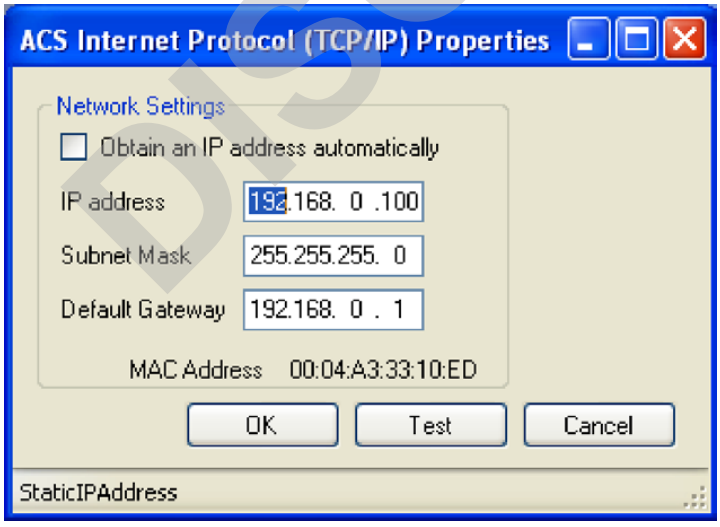


Figure 1: Assigning ACS Drive an IP Address.

4. Using Add-on Profiles & Add-On Instructions

4.1 Overview

These instructions assume that the user is very familiar with Studio 5000 Logix Designer / RSLogix 5000 programming and EtherNet/IP communications. This document references the EtherNet/IP Users Guide 3600-4168 which defines the EtherNet/IP interface to the ACS drive and controller.

4.2 Step 1: Import Add-On Instructions

The first step to using the ACS Add-On Instructions is to import them into your Studio 5000 Logix Designer / RSLogix 5000 program.

1. In your Controller Organizer tree view, right click on Add-On Instructions and select Import Add-On Instruction.
2. Browse to the directory which contains the ACS Add-On Instructions
3. Select the instruction you wish to import
4. The default configuration should be ok, click 'OK'
5. Repeat for all instructions you wish to use.

NOTE: IF YOU ARE PLANNING ON USING THE MOTION ADD-ON INSTRUCTIONS, YOU WILL NEED THE START MOTION, AND CLEAR START MOTION INSTRUCTIONS.

4.3 Step 2: Add ACS Drives to Project

You will now want to add the ACS Drive(s) to your project.

Tolomatic has created EDS based Add-On Profiles which work hand-in-hand with the Add-On Instructions. The EDS files are included in the Add-On Instructions download. EDS file 36043187 is for ACS servo and stepper drives. EDS file 36043188 is for ACSI integrated motor/drive/controller.

1. Click on Tools->EDS Hardware Installation Tool
2. Follow the RSLogix EDS wizard to Register an EDS file
3. When prompted, browse to the EDS file (36043187_ACS_DRIVE.eds) or (36043188_ACSI_DRIVE.eds)
4. Once completed, Right click on the Ethernet Port Controller in your controller organizer window
5. Select 'New Module'
6. In the Catalog, search 'Tolomatic'
7. Assign a Name and Description for each drive
8. Assign an IP Address (default Drives ship in DHCP mode)

4.4 Step 3: Create Enable Rung

Many of the Add-on Instructions reference the same command word on the ACS drive. Because of this, "race conditions" on this command word can exist in programs if proper care is not followed. The program should only send the enable command if the drive is not enabled. See below:

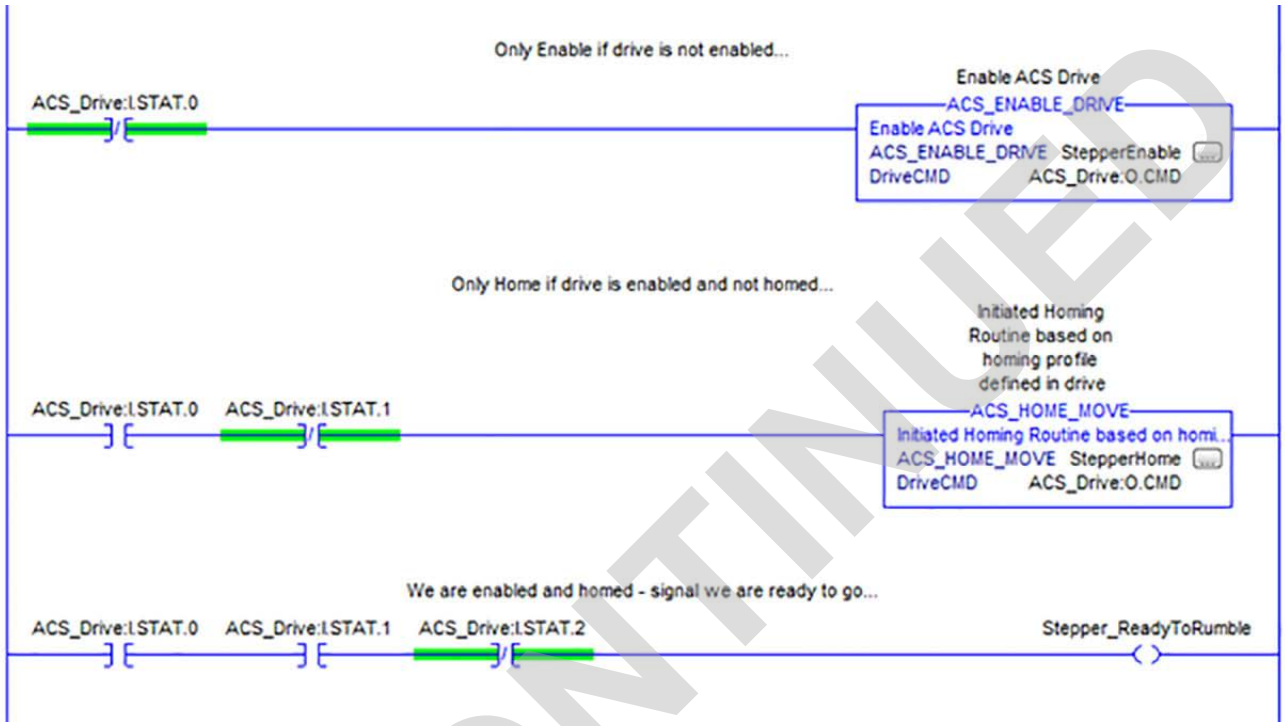


Figure 2 - Create Enable Rung

As soon as the drive is enabled, the enable command is no longer active. If this run was left on, the command word would overwrite other commands such as Home and Start Motion.

> A note about faults:
>
> A common implementation for the ACS Drive is to cut motor power when an ESTOP switch is activated, keeping the drive powered by an auxiliary power supply connected to Keep Alive input on the drive. This causes an under voltage fault on the drive. The drive disables the motor in an attempt to protect itself. To clear most faults, the drive must be disabled and enabled by the program. This is true even if the drive has already disabled itself as a result of a fault. The drive uses the cycling of the enable bit from the PLC as an active acknowledgement that the program has detected and processed the fault. Once the drive detects this transition, the fault will clear if the fault condition is no longer present.

4.5 Step 4: Create Start Motion Timer Rungs

According to the EtherNet/IP users guide, in order to continuously make motion, you must strobe the Start Motion bit in the Output Assembly. The drive detects the rising edge of the bit, so if you want to make a move after you have already made a move, you must bring the bit low, before you again assert it to start motion. To accomplish this, you use a timer to force down a Clear Start Motion command after a Start Motion Command has been sent. To enforce this logic, we do the following for each drive:

1. In the Program Tags, add a StartMotionTimerEnable BOOL tag
2. In the Program Tags, add a StartMotionTimer TIMER tag

Now create the following rungs in your program:

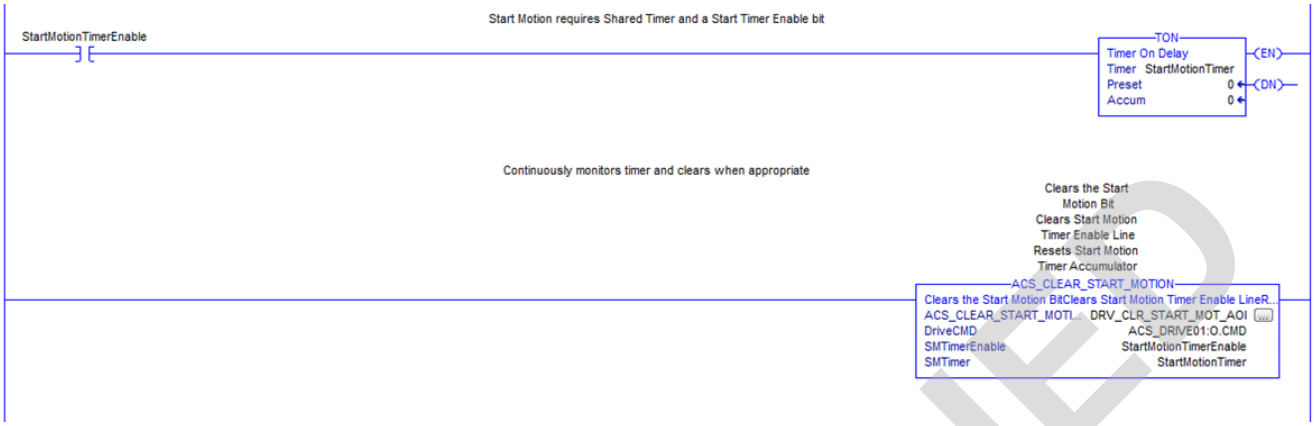


Figure 3 - Start Motion

You may want different logic in your application, so the add-on instructions give you this control. The basic thing to remember is that the drive must detect the start motion bit LOW before the PLC sets it high again. This means that the Start Motion Timer must be longer than the RPI, and you should not command another move before the Clear ACS Start Motion command has a chance to be sent to the drive.

4.6 Motion Example

The following is an example of a Force Move (ACS Servo and ACSI Servo Only), but Absolute and Incremental Moves have the same inputs.

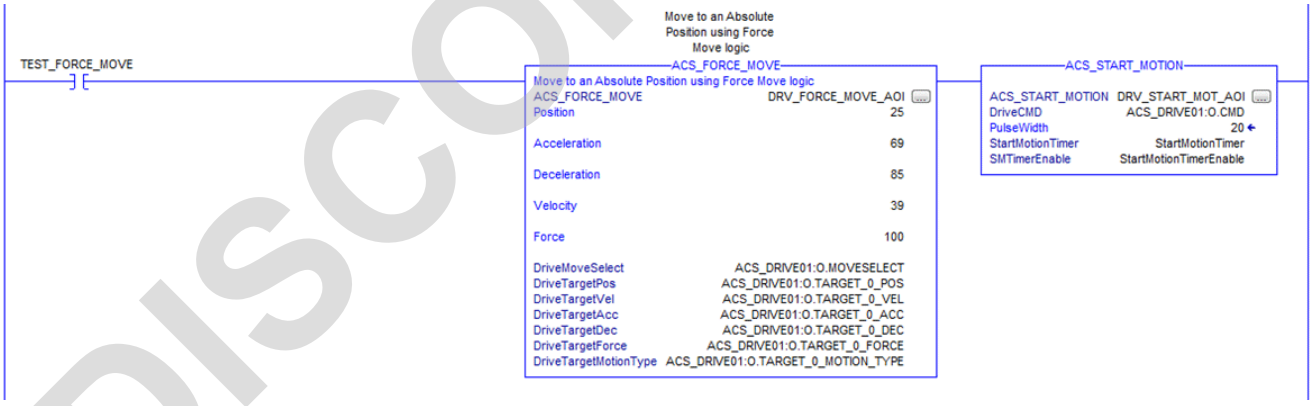


Figure 4 - Force Move Example

In this application, the move is immediately initiated when the TEST_FORCE_MOVE bit is set high. The move is set up and the Drive Target Variables are mapped to the specific variables in the Controller Tags for a given drive.

Once the move is set up, the ACS Start Motion command is sent. We reference the Start Motion Timer program tag, the Start Motion Timer Enable semaphore program tag, and specify the pulse width of the Enable line. For this to be work correctly, the pulse width should be approximately 2x the RPI of the drive. In this instance, it is 20ms.

Add-On Instruction Variable	Controller Tag Reference	Description
DriveCMD	[ACS Drive]:O.CMD	Command Register of Drive[1, 2]
DriveMoveSelect	[ACS Drive]:O.MOVESELECT	Index Move to Execute – 0 uses Target_0 settings, otherwise the indexed positions are configured in the Tolomatic Motion Interface
DriveTargetPos	[ACS Drive].O.TARGET_0_POS	Index 0 Target Position[4]
DriveTargetVel	[ACS Drive].O.TARGET_0_VEL	Index 0 Target Velocity[4]
DriveTargetAcc	[ACS Drive].O.TARGET_0_ACC	Index 0 Target Acceleration[4]
DriveTargetDec	[ACS Drive].O.TARGET_0_DEC	Index 0 Target Deceleration[4]
DriveTargetForce	[ACS Drive].O.TARGET_0_ FORCE	Index 0 Force %
DriveTargetMotionType	[ACS Drive].O.TARGET_0_ MOTION_TYPE	Motion Type Register[3]
DriveDigitalOutput	[ACS Drive].O.DIGITAL_OUTPUT	Set Digital Output Pins register mask
[Unused]	[ACS Drive].I.CPOS	Current Position[4]
[Unused]	[ACS Drive].I.STAT	Drive Status Register[5]
DriveFaults	[ACS Drive].I.FAULTS	Drive Faults Register[5]
[Unused]	[ACS Drive].I.INPUT_BITS	Read Digital Inputs
[Unused]	[ACS Drive].I.OUTPUT_BITS	Read Digital Outputs
[Unused]	[ACS Drive].I.AIN	Read Analog Input
[Unused]	[ACS Drive].I.AOUT	Read Analog Output

[1]The individual bits in this register should not be changed independently of each other. There are some combinations that are invalid, and the drive will not recognize these commands. The following are accepted: 0x0 – Disable Drive; 0x1 – Enable Drive; 0x3 – Start Motion; 0x5 – Home; 0x8 – Software Stop (Estop); 0x11 – Stop Motion

[2]To clear start motion manually, simply send it another valid command. Typically the Enable Drive Command. If the drive is already enabled, the drive will stay enabled.

[3]Valid motion types for network controlled ACS Drives: 0x0 – Absolute; 0x1 – Increment Position; 0x2 – Decrement Position; 0x9 – Force Move (Servo Only) 0xB – Increment Position (Rotary); 0xC – Decrement Position (Rotary); 0xD – Velocity Forward (Rotary); 0xE – Velocity Reverse (Rotary)

[4]All distance is in drive default millimeters; Speed is in mm/s; Acceleration is mm/s^2 this can be changed in TMI for rotary actuators to Revs or Degrees.

[5]Drive Status and Fault Register Masks are defined in the ACS Drive Ethernet/IP Programmers Guide, Section 3.2 Input Assembly (3600-4168).

5. Setting Up the Allen Bradley PLC Using the RSLogix 5000 Software Using Generic Module (no EDS file)

This instruction will walk through how to add an Ethernet module to a PLC controller, create data types, and download configurations and instructions to the controller.

5.1 Adding an Ethernet Module

Open RS Logix and select a 'New Project.'

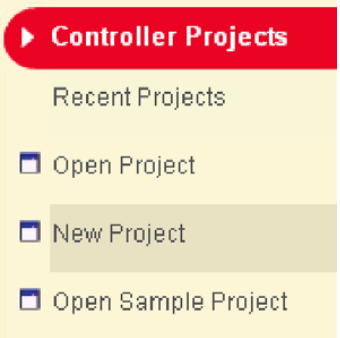


Figure 5: Select New Project

In the 'New Controller' window, select controller, give project a name, and choose a directory to save the project.

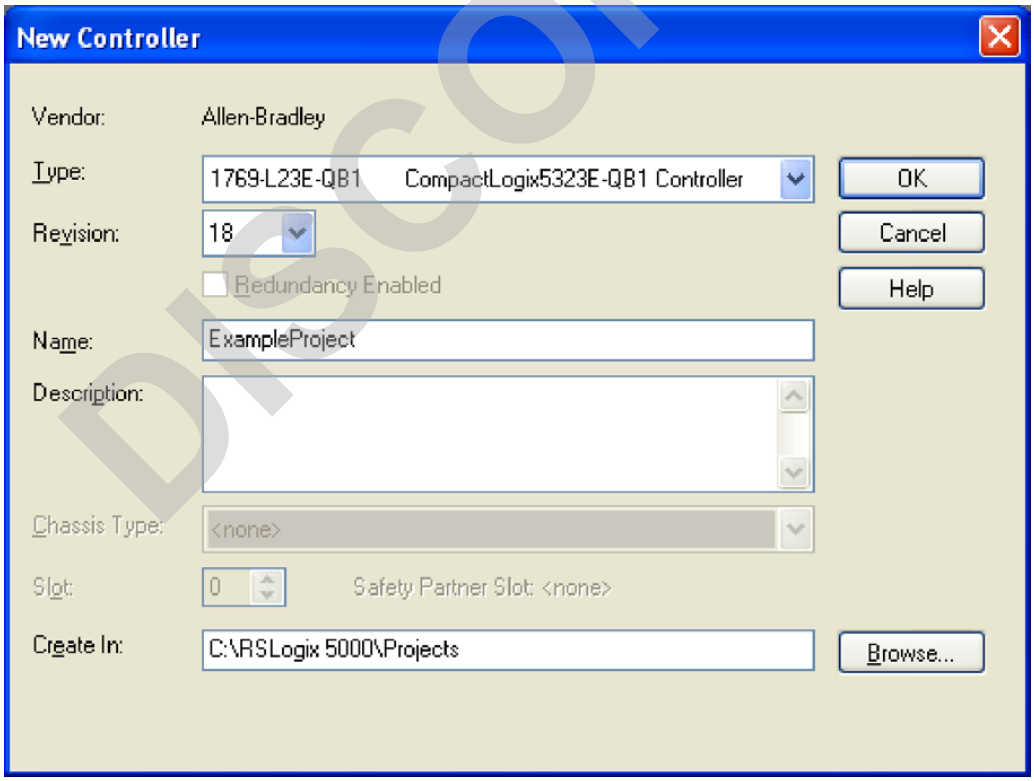


Figure 6: RSLogix New Project Window

Next, add a generic EtherNet/IP module; right click on 'Ethernet' and select 'New Module'.

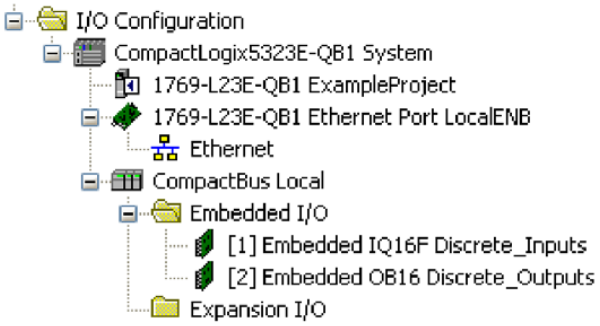


Figure 7: Adding a Generic EtherNet/IP Module

The 'Select Module' window will open. Choose the 'Generic Ethernet Module' and click 'OK". For details on how to use an Add on Profile & Instructions see section 6 "Using Add-On Instructions".

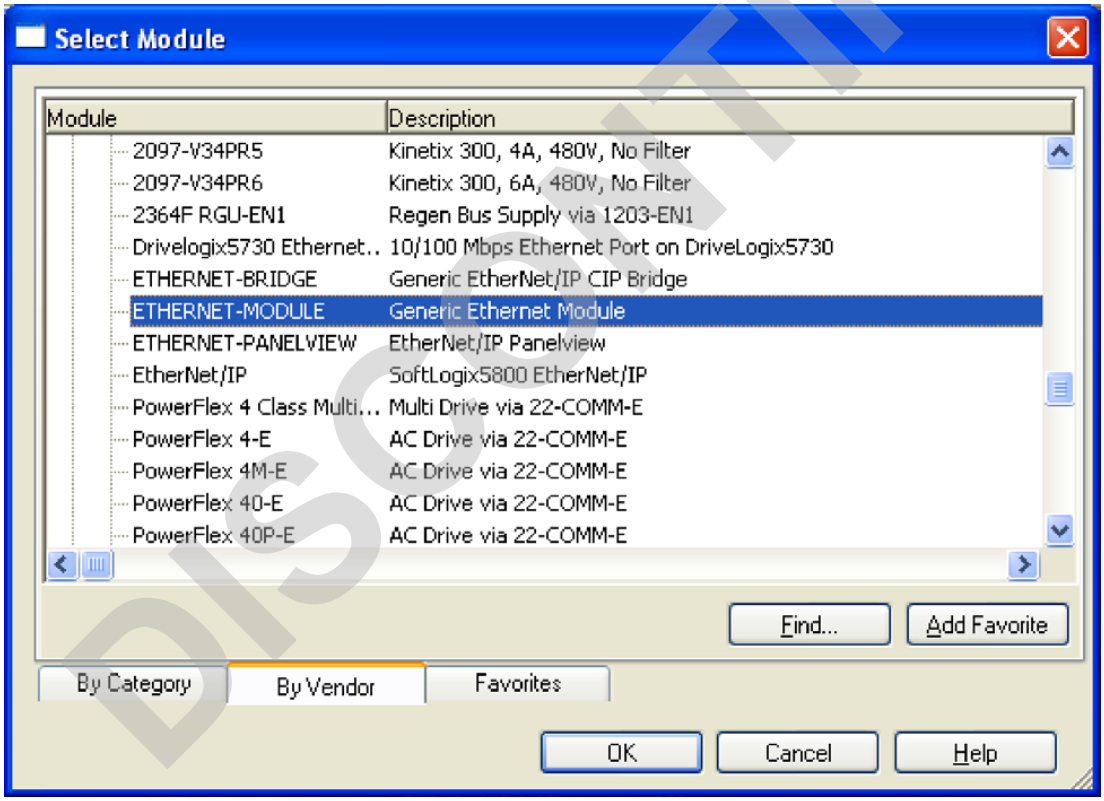


Figure 8: Selecting a Generic Ethernet Module

NOTE: An IP address can be obtained via DHCP by configuring the PLC. For more information on this procedure, please reference the the software/hardware manual for the PLC in use.

The new module properties window should have opened. Enter a name for the module, an IP address for the drive, and the assembly object parameters and click 'OK'. Set up Assembly Instances for the ACS drive as shown in Figure 6. The IP address must match the address configured in the Tolomatic Motion Interface Software (see section 3 of the TMI User Guide 3600-4167.

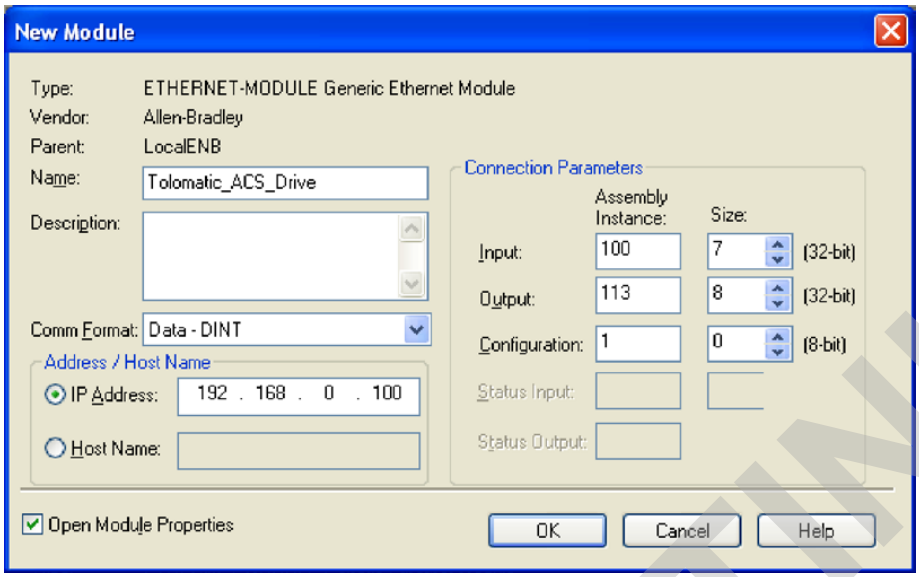


Figure 9: New Module Properties Window

In the connection tab, select these settings or other Requested Packet Interval (RPI). This value determines the interval the controller will use to send/receive data. To conserve bandwidth, use higher values. Click 'OK' when finished. **NOTE:** Setting the packet interval to low may result in erratic motion. Recommended RPI is 12.0 ms. Fastest RPI is 10.0 ms.

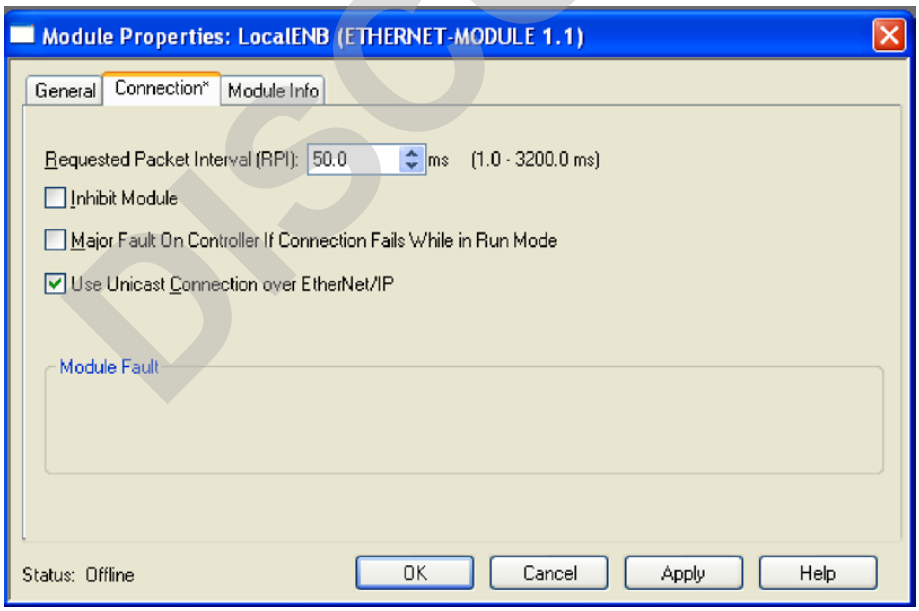


Figure 10: Module Properties Connection Tab

Now the module should have been automatically added in the organizer window.

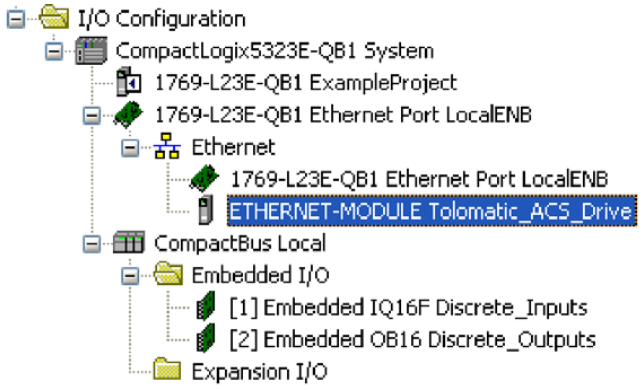


Figure 11: Ethernet Module Added to Organizer Window

A new node named Tolomatic_ACS_Drive now exists under I/O Configuration. Module-defined data types have also been created. These tags allow access to the Input and Output data of the ACS drive using the controller's ladder logic.

5.2 Download Configuration to Controller

Download the previous configurations to the controller and save the project.

5.3 Controller Tags

In the organizer window, expand 'Module-Defined' under 'Data Types'. Make sure Ethernet module data types are:

AB:ETHERNET_MODULE_DINT_28Bytes:I:0

AB:ETHERNET_MODULE_DINT_32Bytes:O:0

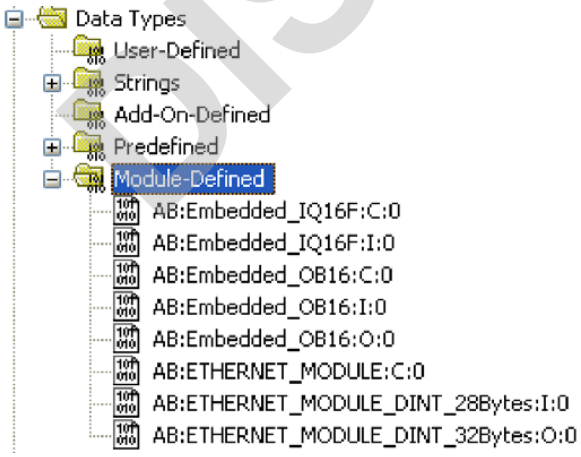


Figure 12: Verify Ethernet Module Data Types in Organizer Window

Next, double-click on 'Controller Tags' in the organizer window.

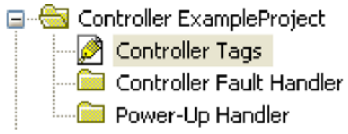


Figure 13: Controller Tags in Organizer Window

A window opens showing all of the existing controller tags. With the new Ethernet module, the controller tags were also created.

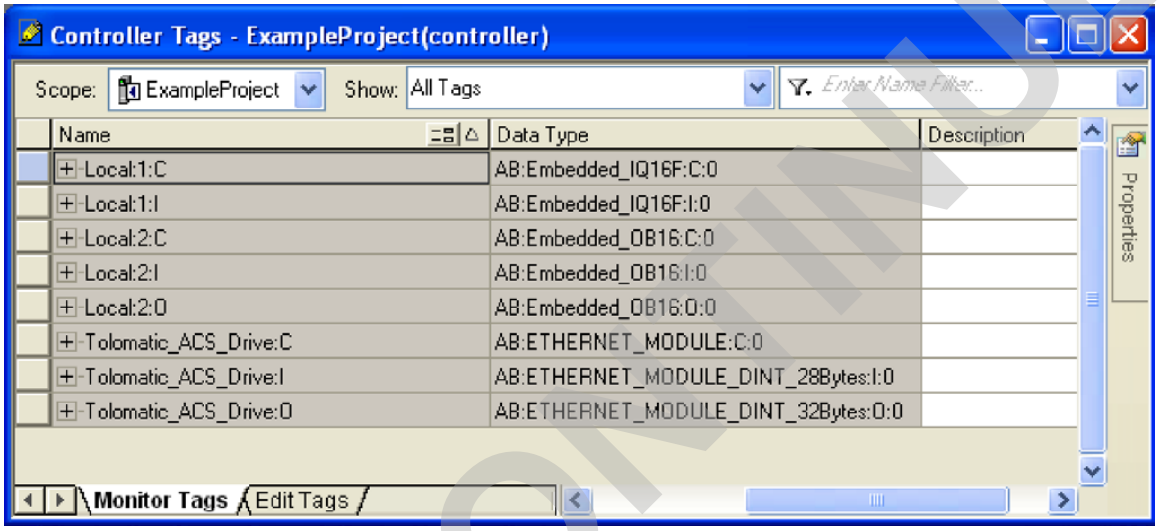


Figure 14: Controller Tags Window

This example uses the new Ethernet module: Tolomatic_ACS_Drive:C, Tolomatic_ACS_Drive:I, and Tolomatic_ACS_Drive:O.

Click the '+' next to each tag to expand the bytes. The data tags are listed numerically.

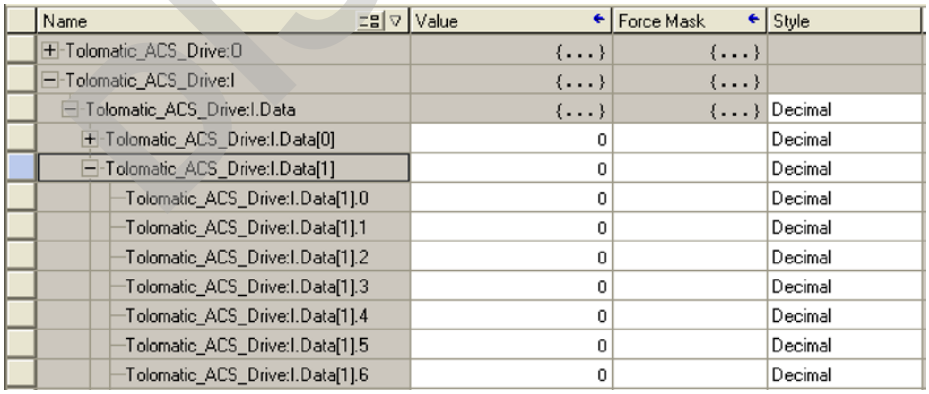


Figure 15: Expanded Controller Tags

Create user defined data types that reflect better naming conventions.

5.4 Creating Program Tags

Create two user defined data types called Tolo_Inputs, and Tolo_Outputs. To do this, Right click on 'User Defined' and select 'New Data Type'. Use the assembly object table as a reference to map the new program tags to each controller tag. Use naming conventions that will be easy to understand in the ladder logic. The following figures show each data type created.

Each bit in the 'Drive_Faults' register (bytes 8-11 of instance 100) represents a particular fault. The 'Drive_Faults' data type is created so each fault can be easily referenced by name instead of by bit number when a ladder logic program is defined.

Give each member a name, data type, and style to display a formatted number.

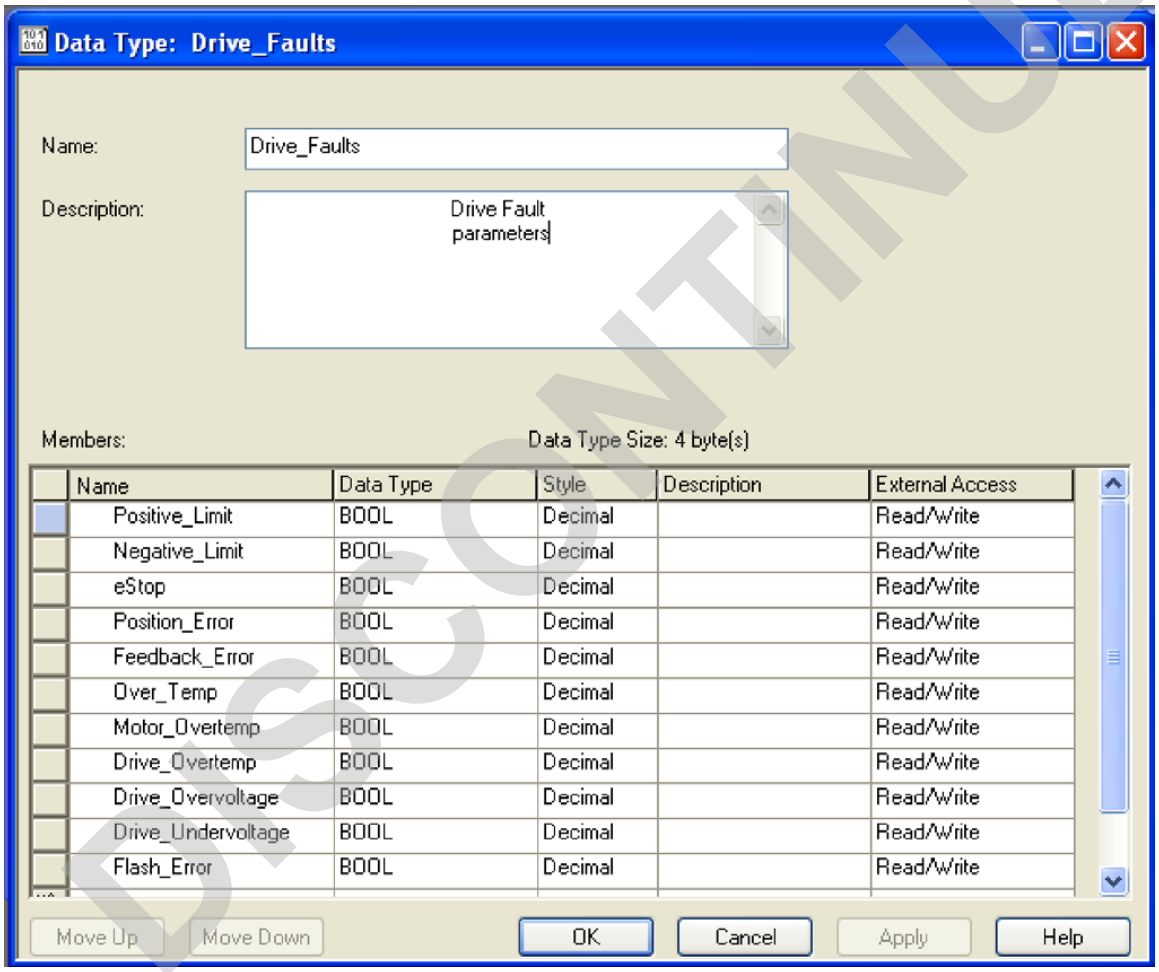


Figure 16: Creating a Drive Faults Data Type

Follow the same procedure for the drive 'Drive_Status' register. Some bits of the Drive_Status data type are not used in the ACS drive. Those are marked as reserved.

Members: Data Type Size: 4 byte(s)

Name	Data Type	Style	Description	External Access
Drive_Enable	BOOL	Decimal	Drive Enable Input	Read/Write
Drive_Homed	BOOL	Decimal	Drive Home Input	Read/Write
Drive_InMotion	BOOL	Decimal	Drive In Motion Input	Read/Write
Drive_EStop	BOOL	Decimal	Drive E-Stop Input	Read/Write
rsved_bit00	BOOL	Decimal		Read/Write
rsved_bit01	BOOL	Decimal		Read/Write
rsved_bit02	BOOL	Decimal		Read/Write
rsved_bit03	BOOL	Decimal		Read/Write
rsved_bit04	BOOL	Decimal		Read/Write
rsved_bit05	BOOL	Decimal		Read/Write
rsved_bit06	BOOL	Decimal		Read/Write
rsved_bit07	BOOL	Decimal		Read/Write
rsved_bit08	BOOL	Decimal		Read/Write
Drive_EthAvail	BOOL	Decimal	Drive Ethernet Online	Read/Write
rsved_bit09	BOOL	Decimal		Read/Write
rsved_bit10	BOOL	Decimal		Read/Write
rsved_bit11	BOOL	Decimal		Read/Write
rsved_bit12	BOOL	Decimal		Read/Write
rsved_bit13	BOOL	Decimal		Read/Write
rsved_bit14	BOOL	Decimal		Read/Write
Drive_BrakeOff	BOOL	Decimal	Brake Not Active	Read/Write
rsved_bit15	BOOL	Decimal		Read/Write
rsved_bit16	BOOL	Decimal		Read/Write

Figure 17: Drive Status Data Type

The previously created data types 'Drive_Faults' and 'Drive_Status' will now be used within our next data type. Create a data type called 'Tolo_Inputs'. Add these faults and status members using user defined data types. The other members can be created using the statndard data types DINT and REAL.

Data Type: Tolo_Inputs

Name: Tolo_Inputs

Description: Tolomatic ACS Drive Inputs

Members: Data Type Size: 28 byte(s)

Name	Data Type	Style	Description	External Access
Current_Position	REAL	Float		Read/Write
Drive_Status	Drive_Status			Read/Write
Drive_Faults	Drive_Faults			Read/Write
Digital_Input	DINT	Binary	8 bits used of 32	Read/Write
Digital_Output	DINT	Binary	4 bits used of 32	Read/Write
Analog_Input	REAL	Float		Read/Write
Analog_Output	REAL	Float		Read/Write

Move Up | Move Down | OK | Cancel | Apply | Help

Figure 18: Creating a Tolo_Input Data Type

Next, create another data type called 'Network_Outputs' that will be used as a data type within the 'Tolo_Outputs' data type.

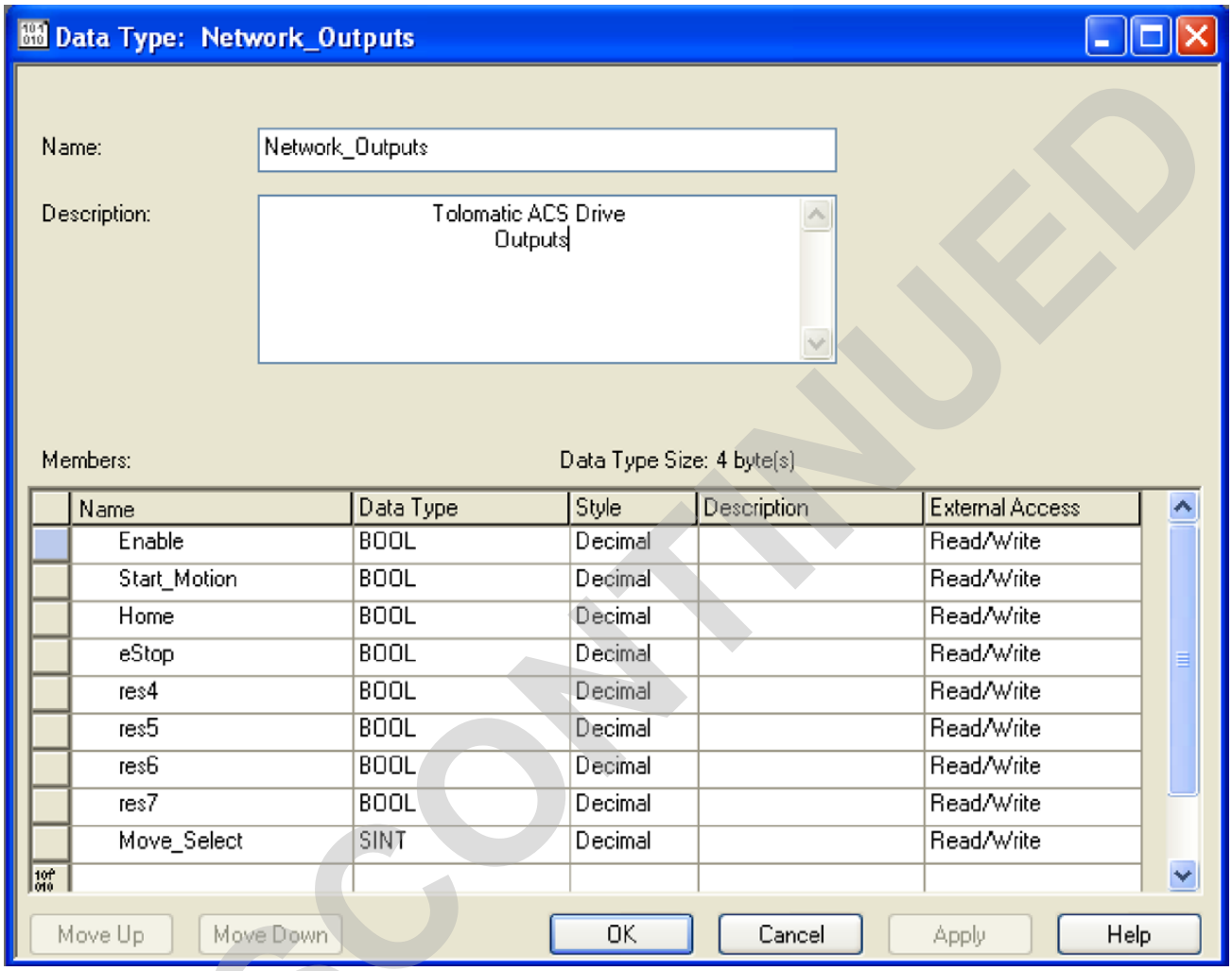


Figure 19: Creating a Network_Output Data Type

Next, create the 'Tolo_Outputs' data type as shown.

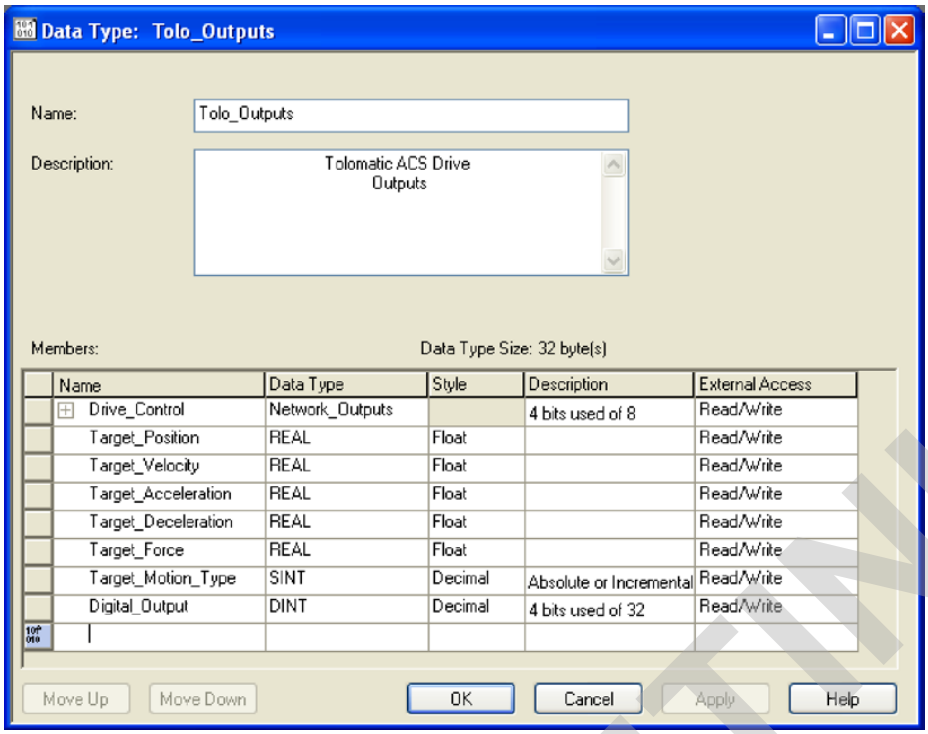


Figure 20: Create a Tolo_Outputs Data Type

Open the controller tags and click on the 'Edit' tab. Add Tolo_Inputs and Tolo_Outputs as the new data types just created.

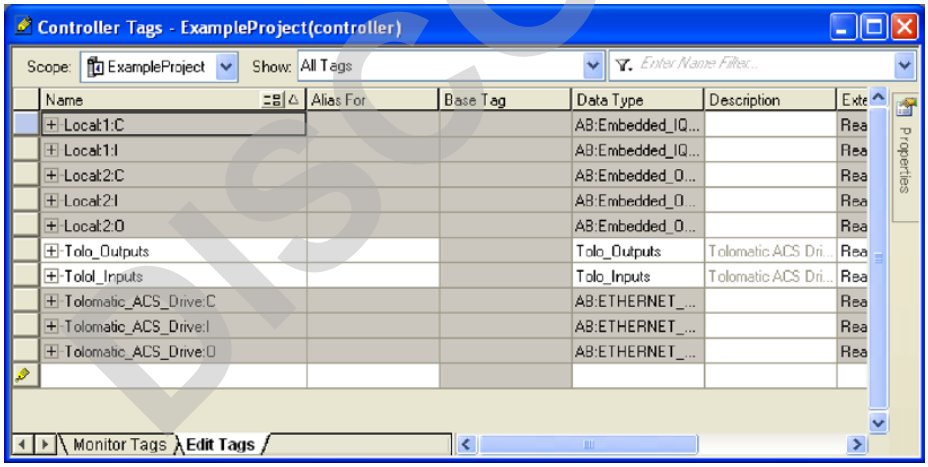


Figure 21: Add Tolo_Inputs and Tolo_Outputs as New Data Types

This completes setup of program tags with recognizable naming conventions.

5.5 Ladder Logic Instructions

Double-click on 'Main Routine' to enter the ladder logic program window.

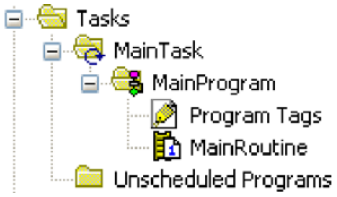


Figure 22: Main Routine in the Organizer Window

Insert 'Synchronous Copy File' instructions to copy the program tags to the controller tags.

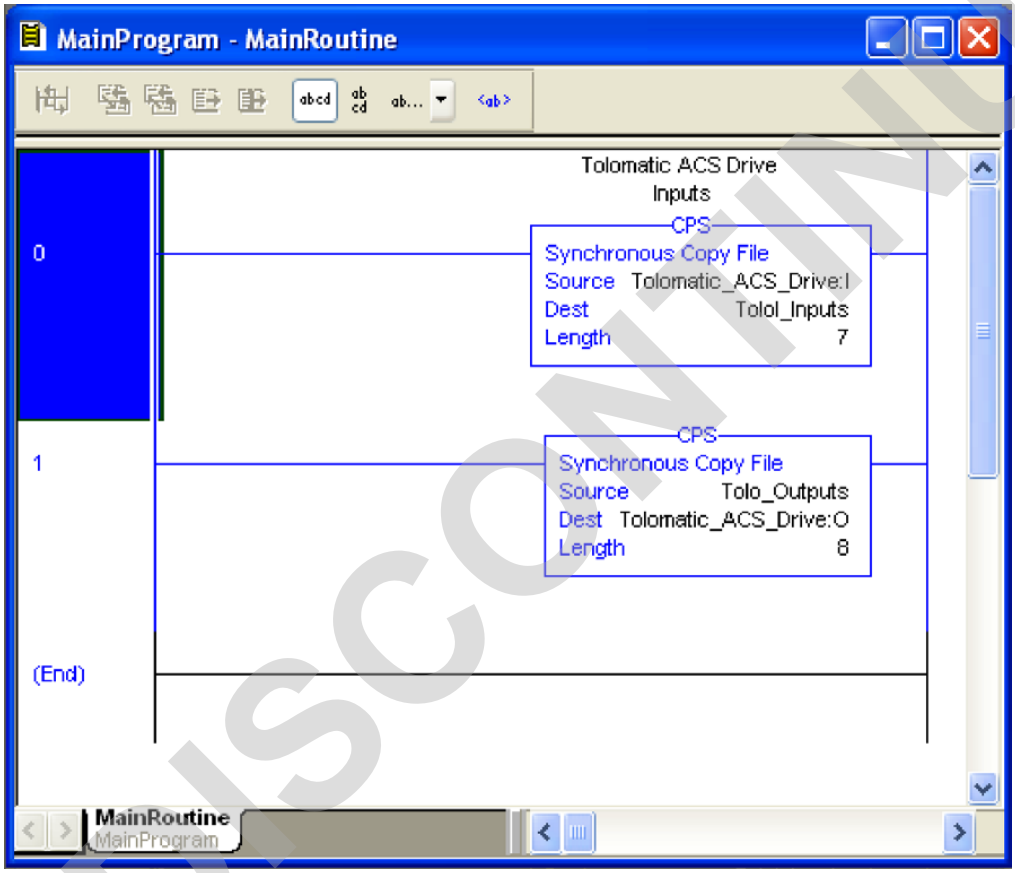


Figure 23: Inserting the Synchronous Copy File Instructions

Download the instructions to the controller and run the program. Turn the key on the PLC to RUN and the program should be online and running.

6. Make Motion

When the PLC is in RUN mode and RSLogix is online, manipulate the data tags in the controller tags window. If the proper bits are written, the drive will react as instructed.

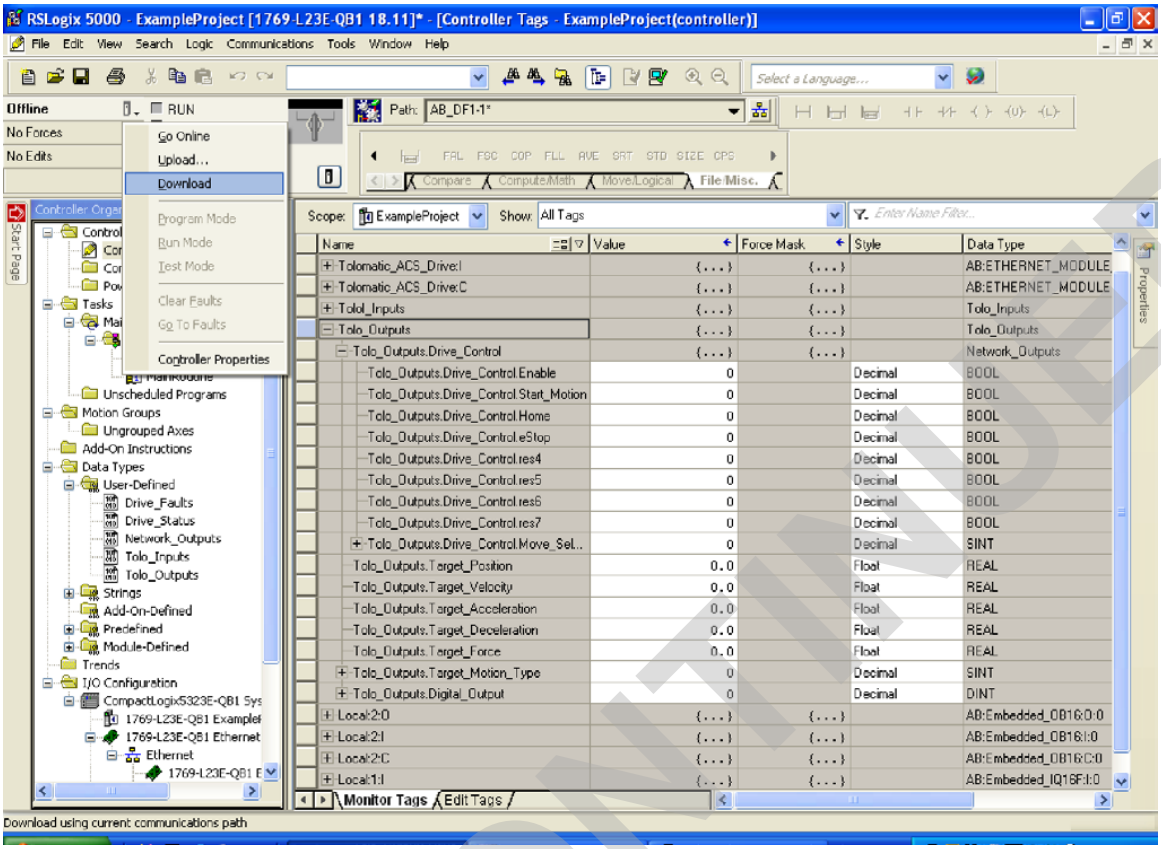


Figure 24: Manipulating Data Tags in Controller Tags Window

6.1 Home Move

The first step when bringing a drive online from power-up, is usually to have it find the actuator's home location. To do this, write a '1' to Enable and Home bits (send command 0x5) of Tolo_Outputs.Drive_Control. The drive should immediate begin the homing motion profile that was setup previously using TMI. When homing has successfully completed the tag Tolo_Inputs.Drive_Status.Drive_Home will turn to '1'.

Tolo_Outputs	{...}
Tolo_Outputs.Drive_Control	{...}
Tolo_Outputs.Drive_Control.Enable	1
Tolo_Outputs.Drive_Control.Start_Motion	0
Tolo_Outputs.Drive_Control.Home	1
Tolo_Outputs.Drive_Control.eStop	0

Figure 25: Finding the Home Postion

6.2 Absolute Move

To do an absolute move, set the Position, Velocity, Acceleration, and Force parameters of Tolo_Outputs.Drive_Control to desired values. Set Move_Select to '0' and Motion_Type to '0'. Then raise Start_Motion from '0' to '1'. Start_

Motion begins the move only when its value cycles from ‘0’ to ‘1’ and Enable is also ‘1’ (send command 0x3). If the position has not changed since the last move there will be no motion because the actuator is already at the desired position.

Tolo_Outputs	{...}
Tolo_Outputs.Drive_Control	{...}
Tolo_Outputs.Drive_Control.Enable	1
Tolo_Outputs.Drive_Control.Start_Motion	1
Tolo_Outputs.Drive_Control.Home	0
Tolo_Outputs.Drive_Control.eStop	0
Tolo_Outputs.Drive_Control.res4	0
Tolo_Outputs.Drive_Control.res5	0
Tolo_Outputs.Drive_Control.res6	0
Tolo_Outputs.Drive_Control.res7	0
Tolo_Outputs.Drive_Control.Move_Select	0
Tolo_Outputs.Target_Position	55.2
Tolo_Outputs.Target_Velocity	25.0
Tolo_Outputs.Target_Acceleration	200.0
Tolo_Outputs.Target_Deceleration	200.0
Tolo_Outputs.Target_Force	100.0
Tolo_Outputs.Target_Motion_Type	0
Tolo_Outputs.Digital_Output	0

Figure 26: Making an Absolute Move

6.3 Increment Move

Set the same motion profile parameters as in the Absolute Move example in section 5.2. Change Target_Motion_Type to ‘1’ and keep Move_Select at ‘0’. Set the position to the desired increment distance; in this example it is 10mm. Make sure Enable is ‘1’ (command 0x1). Now toggle Start_Motion from ‘0’ to ‘1’ to initiate the move (command 0x3).

To perform an Incremental Negative Move, repeat the same procedure using a '2' for Target_Motion_Type.

Tolo_Outputs	{...}
Tolo_Outputs.Drive_Control	{...}
Tolo_Outputs.Drive_Control.Enable	1
Tolo_Outputs.Drive_Control.Start_Motion	1
Tolo_Outputs.Drive_Control.Home	0
Tolo_Outputs.Drive_Control.eStop	0
Tolo_Outputs.Drive_Control.res4	0
Tolo_Outputs.Drive_Control.res5	0
Tolo_Outputs.Drive_Control.res6	0
Tolo_Outputs.Drive_Control.res7	0
Tolo_Outputs.Drive_Control.Move_Select	0
Tolo_Outputs.Target_Position	10.0
Tolo_Outputs.Target_Velocity	25.0
Tolo_Outputs.Target_Acceleration	200.0
Tolo_Outputs.Target_Deceleration	200.0
Tolo_Outputs.Target_Force	100.0
Tolo_Outputs.Target_Motion_Type	1
Tolo_Outputs.Digital_Output	0

Figure 27: Increment Positive Move

6.4 Index Move

An Index Move uses the setting from the move definitions table put into the drive at setup. These move definitions can only be changed using TMI. It is not necessary to set any of the motion parameters in Drive_Control. These

parameters are ignored the the drive uses the parameters from the move definition table. Set Enable to ‘1’, (command 0x4) and set Move_Select to any value between ‘1’ and 16’. In this example index ‘4’ is selected. Now toggle Start_Motion from ‘0’ to ‘1’ (command 0x3) to initiate the move.

Tolo_Outputs	{...}
Tolo_Outputs.Drive_Control	{...}
Tolo_Outputs.Drive_Control.Enable	1
Tolo_Outputs.Drive_Control.Start_Motion	1
Tolo_Outputs.Drive_Control.Home	0
Tolo_Outputs.Drive_Control.eStop	0
Tolo_Outputs.Drive_Control.res4	0
Tolo_Outputs.Drive_Control.res5	0
Tolo_Outputs.Drive_Control.res6	0
Tolo_Outputs.Drive_Control.res7	0
Tolo_Outputs.Drive_Control.Move_Select	4
Tolo_Outputs.Target_Position	0.0
Tolo_Outputs.Target_Velocity	0.0
Tolo_Outputs.Target_Acceleration	0.0
Tolo_Outputs.Target_Deceleration	0.0
Tolo_Outputs.Target_Force	0.0
Tolo_Outputs.Target_Motion_Type	0
Tolo_Outputs.Digital_Output	0

Figure 28: Index Move

6.5 Other Supported Moves

Force_Move: Motion Type = '9'

Increment Move Positive Rotary: Motion Type = '11'

Increment Move Negative Rotary: Motion Type = '12'

Velocity Forward Rotary: Motion Type = '13'

Velocity Reverse Rotary: Motion Type = '14'

7. Setting ACS drive IP address automatically with Rockwell BOOTP/DHCP Server

7.0 System Requirements

Tolomatic's ACS Drive uses the Tolomatic Motion Interface (TMI) software which is dependent on .NET4. Please reference the TMI User Guide 3600-4164 for minimum requirements.

Hardware

Tolomatic ACS Stepper Drive: P/N 36049666
Tolomatic ACS Servo Drive: P/N 36049662

Tolomatic ACSI - EIP motors

Software

Tolomatic Motion Interface (TMI), version 2.4 or higher.

BOOTP/DHCP Server v2.3 by Rockwell

Cabling

Tolomatic ACS Stepper Drive: USB Type B cable

Tolomatic ACS Servo Drive: USB Type B cable

Network connection

Tolomatic ACSI Motor/Drive/Controller: USB Type Micro B

7.1 Overview

This section describes how to use the Rockwell BOOTP/DHCP server application included with the RSLogix5000 software with the Tolomatic ACS drive. This application is used to assign IP addresses to EtherNet/IP™ devices via DHCP. Many IP addresses can be assigned to a network of ACS drives that are uniquely identified by their MAC address.

NOTE: Windows Firewall may interfere with the procedure. It is recommended to turn the Windows Firewall OFF before proceeding with the procedure.

NOTE: ACSI should be set to Infrastructure Mode and configured for DHCP to use this mode.

7.2 ACS / TMI Setup

Run Tolomatic Motion Interface and connect to the ACS drive. Configure the actuator and motor, set the ACS Drive for EtherNet/IP on the Mode Setup tab.

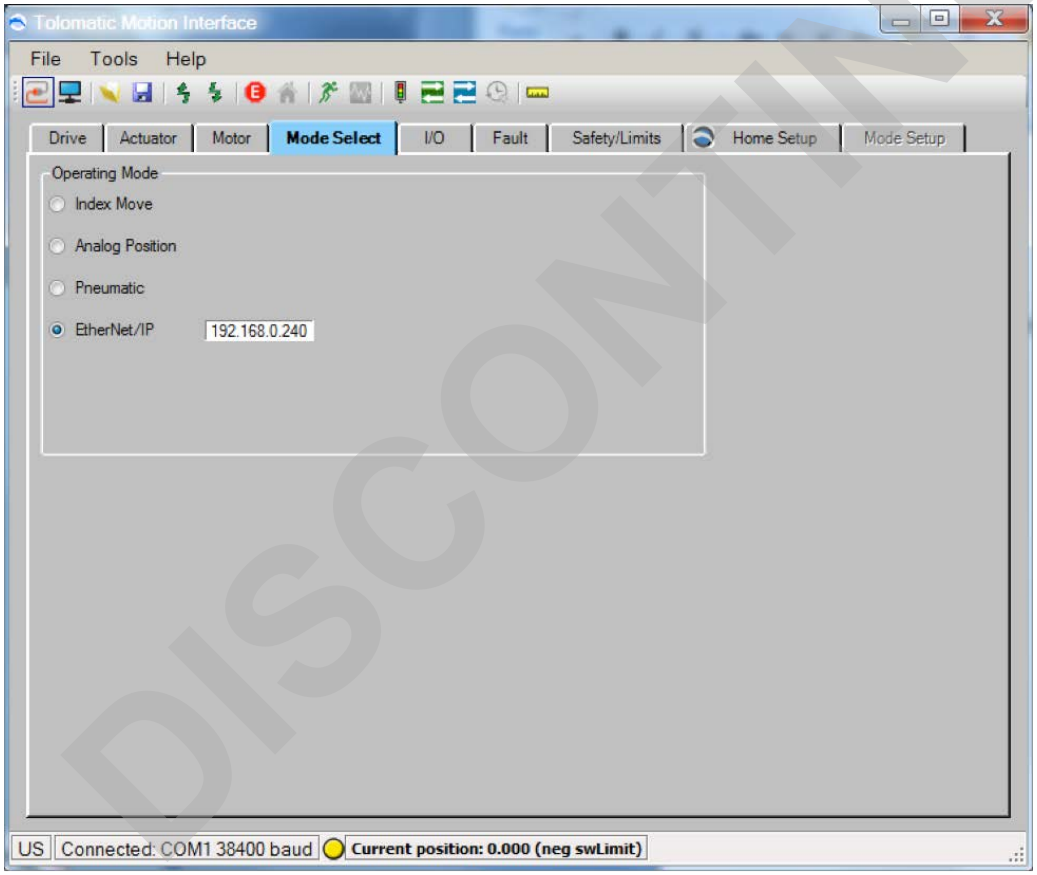


Figure 29; Tolomatic Motion Interface Dialog

Open the 'ACS Internet Protocol Properties' window (menu option Tools- > EtherNet/IP setup). In this window you will see the default values for IP address, Subnet Mask, and Default Gateway parameters.

Check the box for 'Obtain an IP Address automatically'. Set the Subnet Mask and Default Gateway to match your network.

For ACSI Motor/Drive/Controllers Infrastructure Mode automatically has DHCP enabled. The ACSI can also be configured using the TCP/IP EtherNet/IP Object (0xF5).

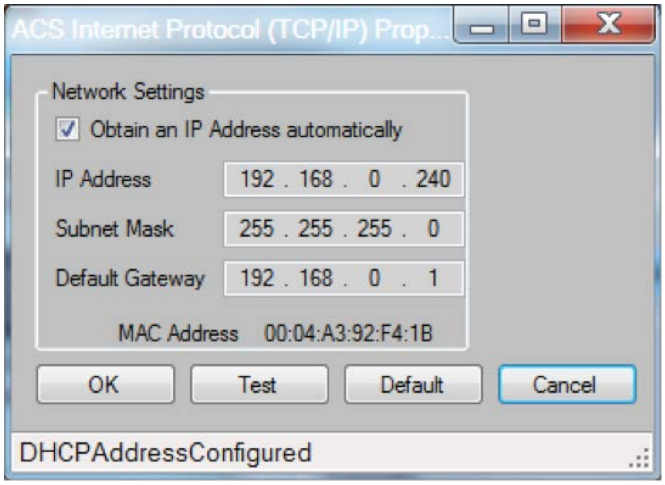


Figure 30; IP Address Display

Click 'OK' and write settings to Flash: File–>Write Current Settings to Drive Flash

Obtain MAC Address (00:04:A3:92:F4:1B in this example) from the Drive Info area on the Drive tab, or from a label on the cover of the ACS drive.

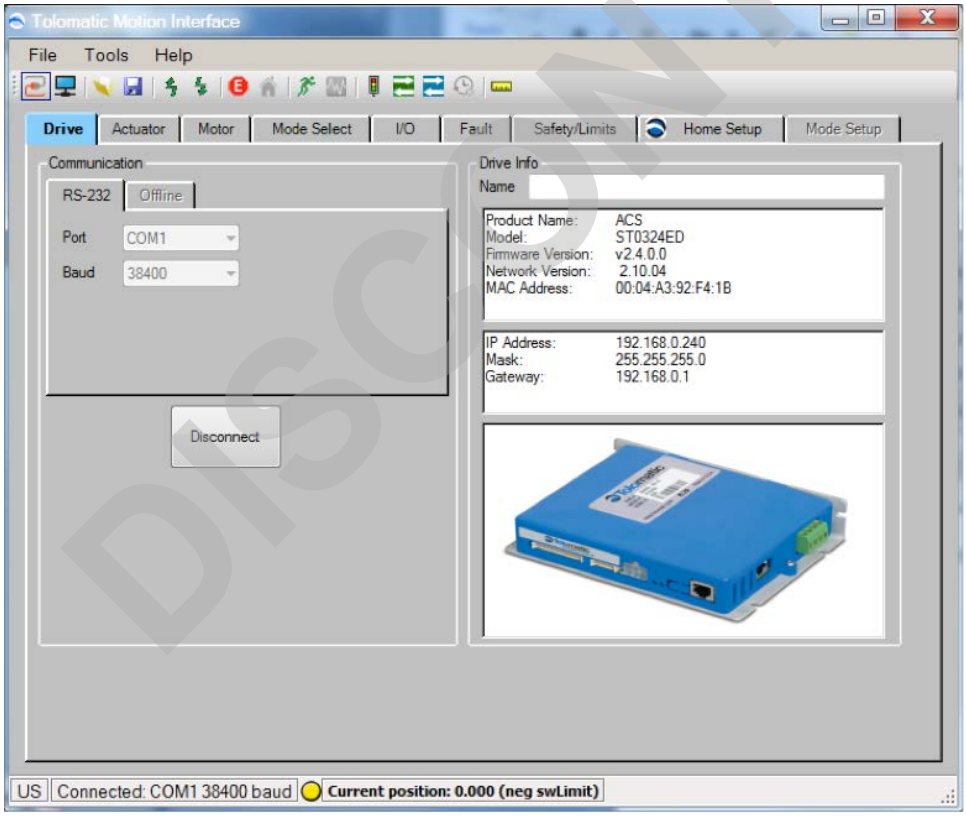


Figure 31; MAC Address Window

Disconnect TMI saving settings to flash.

Cycle Power on the ACS drive, it will now boot with DHCP enabled.

7.3 Rockwell BOOTP/DHCP SERVER Setup

Open BOOTP/DHCP Software

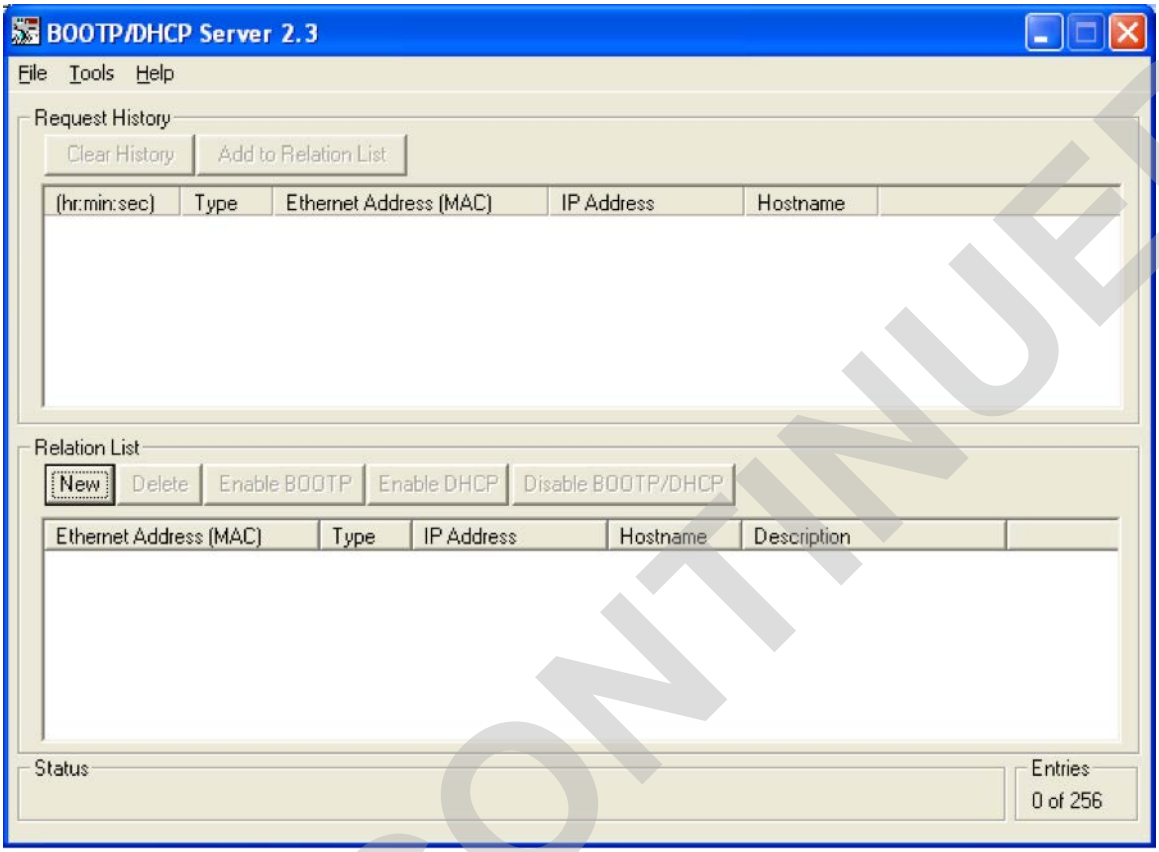


Figure 32; BOOTP/DHCP Server Configuration

Under the Relations list click 'NEW'

Enter the MAC address and IP address desired. Then click OK.

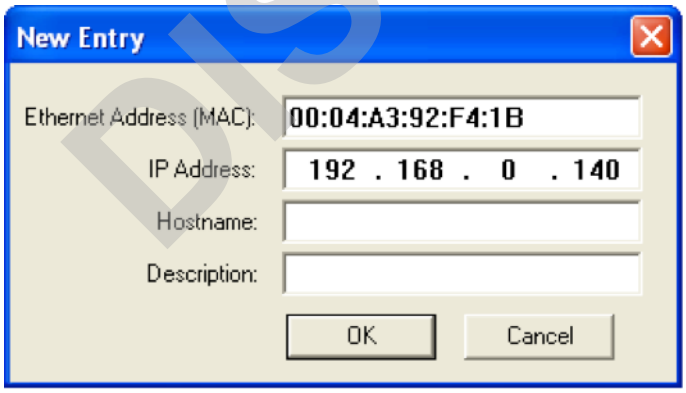


Figure 33; IP Address Entry Dialog

Confirm that the Relation List includes the newly added MAC address with desired IP address.

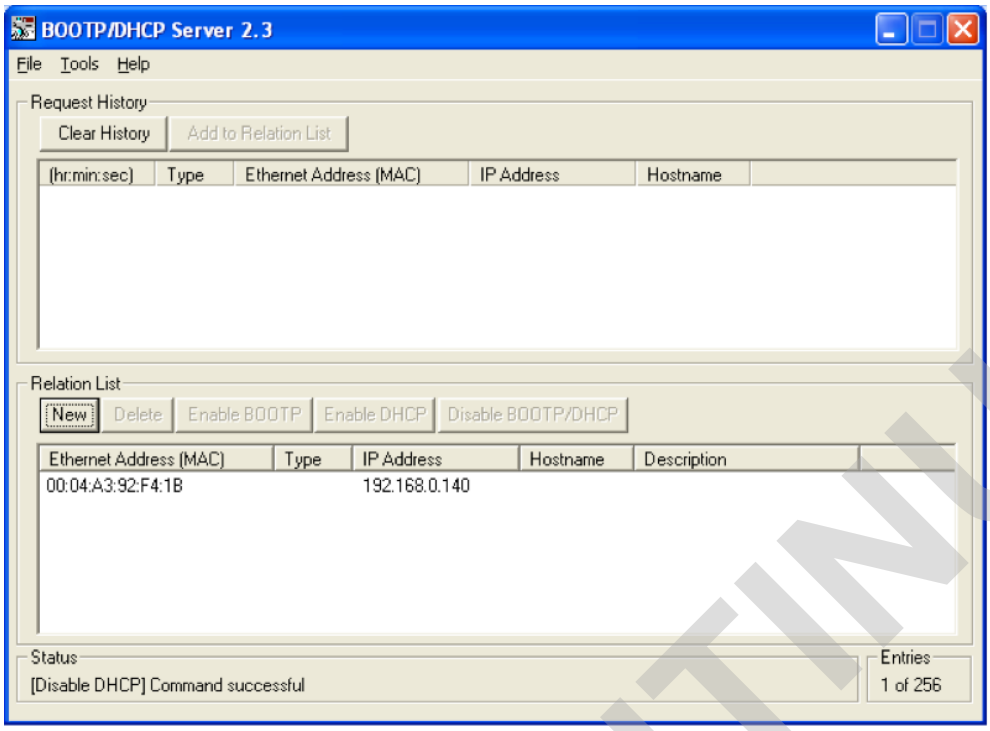


Figure 34; BOOTP/DHCP Server Configuration

Once the server software sends the IP address to the drive, a 'Request History' message should appear displaying the new IP address for the ACS drive.

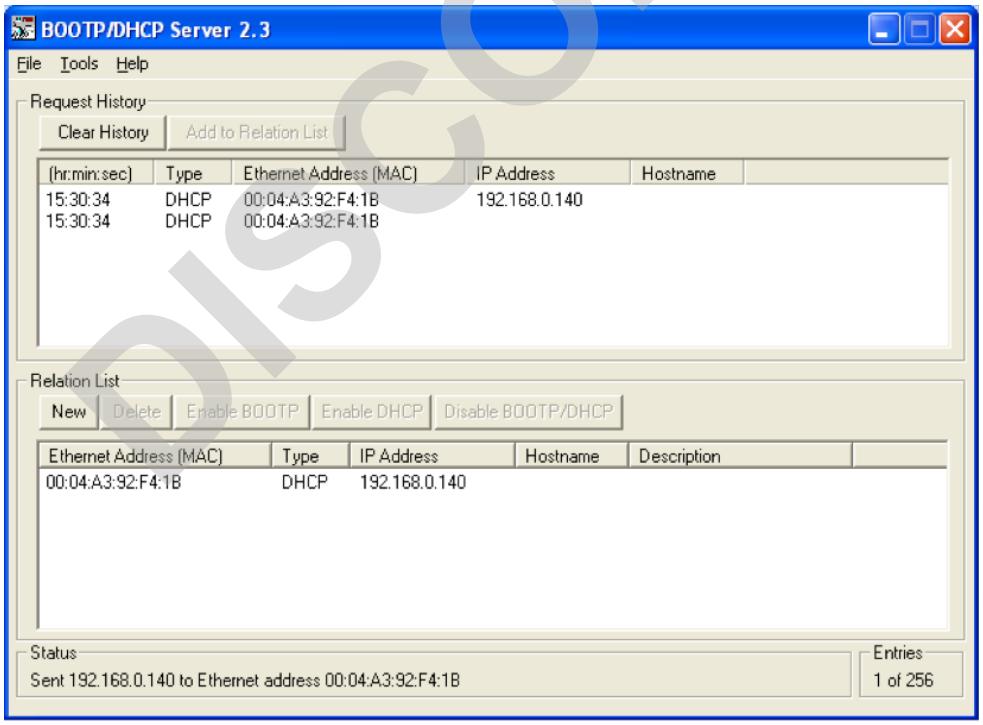


Figure 35; BOOTP/DHCP Server Configuration

7.4 Confirm IP Address

To verify that the IP address assignment was successful, connect the ACS drive with TMI and validate the updated IP address.

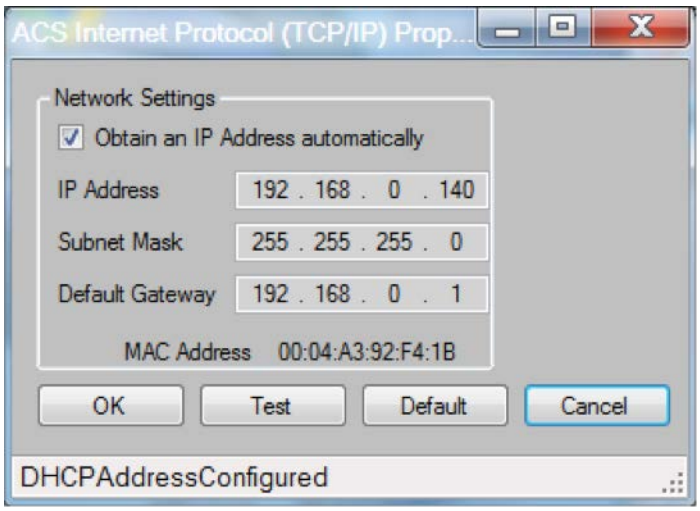


Figure 36; IP Address Dialog

On the DHCP Server software select the ACS drive MAC address and click Disable BOOTP/DHCP.

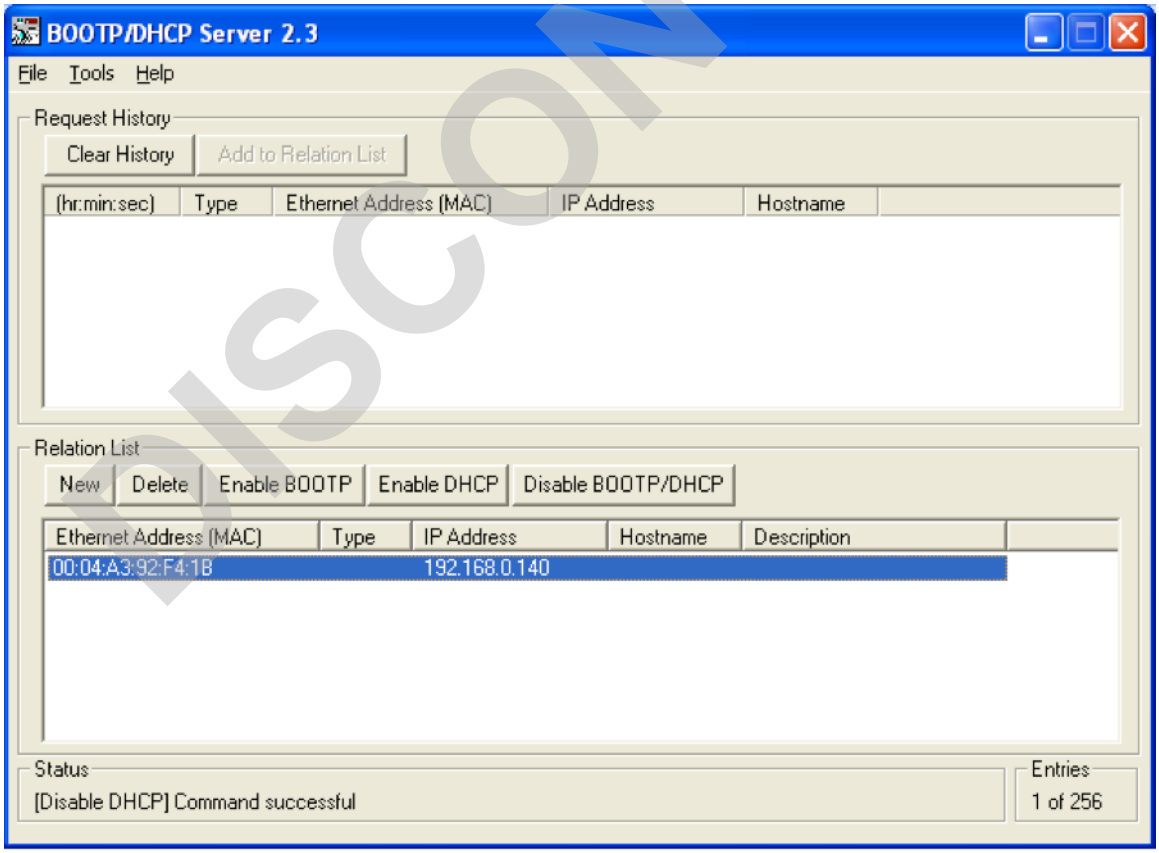


Figure 37; BOOTP/DHCP Server Configuration

Confirm the 'Obtain an IP Address automatically' check box is unchecked. This will require the window to be reopened.

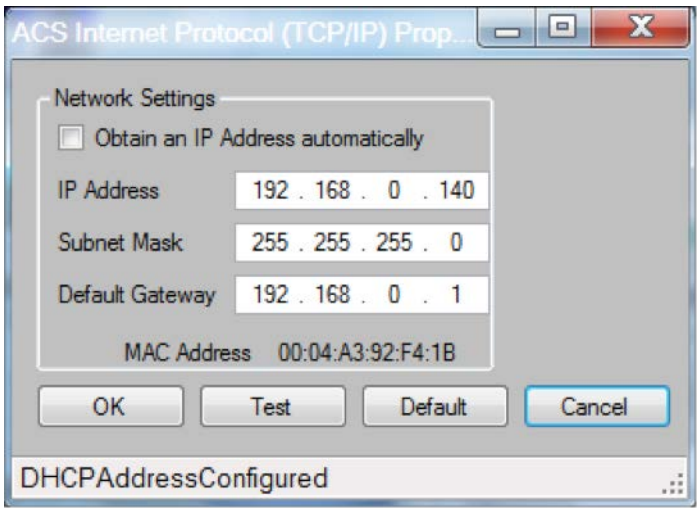


Figure 38; IP Address Dialog

The process is now complete.

7.5 Conclusion

This tutorial went through the process of using Rockwell's BOOTP/DHCP Server to automatically assign IP addresses to Tolomatic ACS drives. The second ACS drive connection to TMI was only to show the ACS drive's behavior using DHCP. Normally only the initial TMI connection and configuration is necessary. Using DHCP, many IP addresses can be assigned to a network of ACS drives that are uniquely identified by their MAC addresses.

MADE IN U.S.A.

Tolomatic EXCELLENCE IN MOTION

COMPANY WITH QUALITY SYSTEM CERTIFIED BY DNV = ISO 9001 = Certified site: Hamel, MN

USA - Headquarters
Tolomatic Inc.
3800 County Road 116
Hamel, MN 55340, USA
Phone: (763) 478-8000
Toll-Free: **1-800-328-2174**
sales@tolomatic.com
www.tolomatic.com

MEXICO
Centro de Servicio
Parque Tecnológico Innovación
Int. 23, Lateral Estatal 431,
Santiago de Querétaro,
El Marqués, México, C.P. 76246
Phone: +1 (763) 478-8000
help@tolomatic.com

EUROPE
Tolomatic Europe GmbH
Elisabethenstr. 20
65428 Rüsselsheim
Germany
Phone: +49 6142 17604-0
help@tolomatic.eu
www.tolomatic.com/de-de

CHINA
Tolomatic Automation Products (Suzhou) Co. Ltd.
No. 60 Chuangye Street, Building 2
Huqiu District, SND Suzhou
Jiangsu 215011 - P.R. China
Phone: +86 (512) 6750-8506
TolomaticChina@tolomatic.com