MLC Series PLC Programming Manual & Installation Guide

Your Industrial Control Solutions Source

_____

www.maplesystems.com

For use with the following:

- MLC Series PLCs

COPYRIGHT NOTICE

This manual is a publication of Maple Systems, Inc., and is provided for use by its customers only. The contents of the manual are copyrighted by Maple Systems, Inc.; reproduction in whole or in part, for use other than in support of Maple Systems equipment is prohibited without the specific written permission of Maple Systems.

WARRANTY

Warranty Statements are included with each unit at the time of purchase and are available at www.maplesystems.com.

TECHNICAL SUPPORT

This manual is designed to provide the necessary information for trouble-free installation and operation of Maple Systems products. However, if you need assistance, please contact Maple Systems:

- Phone: 425-745-3229
- Email: support@maplesystems.com
- Web: www.maplesystems.com

Intended Audience

A **Qualified Person** is one that has the skills and knowledge relating to the construction, installation, operation, and maintenance of the electrical equipment and has received safety training on the hazards involved. **Qualified Personnel** shall:

- Have carefully read the entire operation manual.
- Be trained and authorized to safely energize, de-energize, ground, lockout and tag circuits and equipment, and clear faults in accordance with established safety practices.
- Be trained in the proper care and use of protective equipment such as safety shoes, rubber gloves, hard hats, safety glasses, face shields, flash clothing, etc., in accordance with established safety practices.
- Be trained in rendering first aid.

Purpose and scope of this manual

Thank you for purchasing the MLC Series Programmable Logic Controller (PLC) from Maple Systems. These products are configured using the MAPware-7000 software (purchased separately).

This manual provides information on how to safely install, program, operate, and maintain your product.

Read the manual completely before installing, operating, or performing maintenance on this equipment.

This manual and the accompanying drawings should be considered a permanent part of the equipment and should be readily available for reference and review.

Maple Systems reserves the right, without prior notice, to update information, make product changes, or to discontinue any product or service identified in this publication.

General Precautions

The user must operate the product according to the performance specifications described in the operational manual.

- The MLC Series models are general-purpose products. It is a system component and is used in conjunction with other items of industrial equipment such as PLCs, Loop Controllers, Adjustable Speed Drives (ASD), etc.
- A detailed system analysis and job safety analysis should be performed by the systems designer or systems integrator before including the unit in any new or existing system. Contact Maple Systems for options availability and for application-specific system integration information if required.
- The product may be used to control an adjustable speed drive connected to high voltage sources and rotating machinery that is inherently dangerous if not operated safely. Interlock all energy sources, hazardous locations, and guards in order to restrict the exposure of personnel to hazards. The adjustable speed drive may start the motor without warning.
- Signs at the equipment installation must be posted to this effect. A familiarity with Auto restart settings is a requirement when controlling adjustable speed drives. Failure of external or ancillary components may cause intermittent system operation, i.e., the system may start the motor without warning or may not stop on command. Improperly designed or improperly installed system interlocks and permissives may render a motor unable to start or stop on command.
- Control through serial communications can fail or can also override local controls, which can create an unsafe condition. System safety features should be employed and designed into the integrated system in a manner such that system operation, even in the event of system failure, will not cause harm or result in personnel injury or system damage. Use of the built-in system protective features and interlocks of the equipment being controlled is highly recommended (i.e., emergency-off, overload protection, etc.).
- Never use the MLC Series units to perform emergency stops. Separate switches outside the product, the PLC, and the ASD should be used for emergency stops.
- Changes or modifications to the MAPware-7000 program should not be made without the approval of the system designer or systems integrator. Minor changes or modifications could cause the defeat of safety interlocks and permissives. Any changes or modifications should be noted and included with the system documentation.

WARNING It is extremely important that the unit and other peripherals be used for the specified purpose and under the specified conditions, especially in applications that can directly or indirectly affect human beings.

WARNING Do not use input functions as PT (Plunger-Type) touch switches for applications where danger to human life or serious damage is possible, or for emergency switch applications.

Safety Precautions

Please observe the following precautions when installing the unit. Failure to comply with these restrictions could result in loss of life, serious personal injury, or equipment damage.

WARNING Do not operate the unit in areas subject to explosion due to flammable gases, vapors, or dusts.

WARNING Do not connect the unit to an AC power source. You will cause permanent damage to the unit.

WARNING Do not attempt to use a DC power supply that does not meet unit power requirements. You may cause malfunction or permanent damage to unit.

WARNING Do not power the unit with a DC power supply used for inductive loads or for input circuitry to the programmable logic controller. Severe voltage spikes caused by these devices may damage the unit.

CAUTION Upon receipt of the equipment inspect the packaging and equipment for shipping damage.

- Carefully unpack the equipment and check for parts that were damaged from shipping, missing parts, or concealed damage. If any discrepancies are discovered, it should be noted with the carrier prior to accepting the shipment, if possible. File a claim with the carrier if necessary and immediately notify your Maple Systems representative.
- **DO NOT** install or energize equipment that has been damaged. Damaged equipment may fail during operation resulting in further equipment damage or personal injury.
- Check to see that the model number specified on the nameplate conforms to the order specifications.
- Modification of this equipment is dangerous and must not be performed except by factory trained representatives. When modifications are required contact your Maple Systems representative.
- Inspections may be required before and after moving installed equipment.
- Keep the equipment in an upright position as indicated on the shipping carton.
- Contact your Maple Systems representative for assistance if required.

Handling and Storage

It is recommended to NOT remove the enclosure of the product. If removed, circuitry inside may be susceptible to possible damage by electrostatic discharge. Minimize the possibility of electrostatic discharge by:

- Dissipating static electricity prior to handling
- Handle product only at a static-free grounded location
- Connect frame ground connector of product to a clean earth ground
- Store in an antistatic bag during transport

Store in a well-ventilated covered location and preferably in the original carton if the equipment will not be used upon receipt.

Safety Precautions for Storage

- Store in a cool, clean, and dry location. Avoid storage locations with extreme temperatures, rapid temperature changes, high humidity, moisture, dust, corrosive gases, or metal particles.
- Do not store the unit in places that are exposed to outside weather conditions (i.e., wind, rain, snow, etc.).

Disposal

Never dispose of electrical components via incineration. Contact your state environmental agency for details on disposal of electrical components and packaging in your area.

Installation Precautions

Location and Ambient Requirements

a) Adequate personnel working space and adequate illumination must be provided for adjustment, inspection, and maintenance of the equipment.
b) This product has been designed and manufactured for use in an industrial environment. However, it is not intended to be used for systems which may endanger human life. Consult Maple Systems if you intend to use this product for a special application, such as transportation machines, medical apparatus, aviation and space systems, nuclear controls, submarine systems, etc.
c) Due to the reliance on electrical connections, this product should not be considered as part of any fail-safe system.
d) Avoid installation in areas where vibration, heat, high humidity, rapid temperature variations, dust, fibers, steel particles, explosive/corrosive mists or gases, or sources of electrical noise are present.
e) The installation location shall not be exposed to direct sunlight.
f) Allow proper clearance spaces for installation. Do not obstruct the ventilation openings. Refer to the recommended minimum installation dimensions as shown on the enclosure outline drawings.
g) The MLC Series is intended primarily for indoor installations and may not be suitable for certain outdoor applications. The ambient operating temperature shall be between 0° and 55° C (32° and 131° F).

Mounting Requirements

a) Only **Qualified Personnel** should install this equipment.
b) Install the unit in a secure upright position in a well-ventilated area.
c) A noncombustible insulating floor or mat should be provided in the area immediately surrounding the electrical system at the place where maintenance operations are to be performed.
d) Turn off power before installing or removing any units, modules, racks or terminal blocks. Failure to do so can cause electrical shock or damage to this product and related equipment.

Conductor Routing and Grounding

a) Use separate metal conduits for routing the input power, and control circuits.
b) A separate ground cable should be run inside the conduit with the input power, and control circuits.
c) **DO NOT** connect the control terminal strip return marked COM to earth ground.
d) Always ground the unit to prevent electrical shock and to help reduce electrical noise.

The Metal Of Conduit Is Not An Acceptable Ground.

Table of Contents

Chapter 1 – Introduction

Welcome

Welcome to the Maple Systems' MLC Series Programmable Logic Controllers (PLCs). The MLC Series is a low cost micro PLC that supports ladder logic or IEC 61131-3 programming. The MLC Series products are industrial grade equipment. They are certified to meet the European CE noise immunity/emission standards and they are UL listed for Class I Division 2 environments.

The MAPware-7000 configuration software is used to program both MLC (Maple PLCs) and HMC (Maple HMI + PLCs) products. This manual focuses on the MLC line. For more information on programming HMC devices, see the general *MAPware-7000 Programming Manual*.

Please read this operation manual and visit www.maplesystems.com for the latest information on these products.

MLC Series Basics

Maple Systems MLC Series PLCs are economical programmable logic controllers. The MLC Series is an ideal solution for control systems that require an inexpensive PLC with a lot of functionality. The MLC Series offers flexibility as well. If your control system needs a PLC with just a few I/O, you can select from several models with only fixed I/O – to minimize cost, or models that support expansion slots so that you can customize your I/O configuration. In addition, Maple Systems offers a wide assortment of I/O (input/output) modules including combination digital input/output modules, digital input only, digital output only, analog modules, and high-speed counter modules.

Programmable Logic Controllers (PLCs) provide much more versatility than traditional mechanical control panels. An MLC Series PLC allows a plant floor operator to monitor current conditions of a control system and, if necessary, initiate a change in the operation of the system. Each MLC model has a CPU to run either a ladder logic or an IEC 61131-3 program that reads/writes directly to digital and analog I/O (Inputs/Outputs). The MLC Series PLC can also connect to external 3rd party programmable logic controllers (PLCs), typically through the PLC's serial communications port. The MLC Series can communicate to graphic HMIs that provide the plant floor operator the means to monitor and change parameters in a control system. Maple Systems provides many options for an HMI that will work for your application.

The MLC Series has a power LED to indicate when the unit is on. The MLC Series is powered by +24VDC. Serial ports are located on the front or top of the MLC for connection to HMIs and/or PLCs. The power input is also located on the bottom of each unit. Each model has a USB port for project upload/download. Push-type terminal connectors are used for I/O and power connectors for easy connection to various input/output devices. Finally, the back of the enclosure of the MLC Series allows mounting to a standard 35mm DIN rail.

The MLC Series is designed for industrial environments and conforms to CE compliance for noise immunity and emissions. It is also UL listed and certified for Class I Division 2 installations.

MLC Models Supported

Maple Systems offers several models of the MLC Series. FX-Series MLCs have built-in I/O with no support for expansion modules. EX-Series MLCs support built-in I/O as well as the option for additional I/O modules. All models have an RJ45 RS232 serial communications port and most models offer an RS485 two wire serial port.

Below is a brief description of each currently available FX-Series MLC with built-in I/O but no expansion slots:

- MLC1-F0604P – A PLC with built-in digital I/O (6 inputs, 5 PNP outputs) and no expansion slots.
- MLC1-F0604N – A PLC with built-in digital I/O (6 inputs, 5 NPN outputs) and no expansion slots.
- MLC1-F0808P – A PLC with built-in digital I/O (8 inputs, 8 PNP outputs) and no expansion slots.
- MLC1-F0808N – A PLC with built-in digital I/O (8 inputs, 8 NPN outputs) and no expansion slots.
- MLC1-F0808Y – A PLC with built-in digital I/O (8 inputs, 2PNP/6 Relay outputs) and no expansion slots.
- MLC1-F0808P0201 – A PLC with built-in digital I/O (8 inputs, 8 PNP outputs), analog I/O (2 voltage/current inputs, 1 current output) and no expansion slots.
- MLC1-F0808N0201 – A PLC with built-in digital I/O (8 inputs, 8 NPN outputs), analog I/O (2 voltage/current inputs, 1 current output) and no expansion slots.
- MLC1-F0808Y0201 – A PLC with built-in digital I/O (8 inputs, 2PNP/6 Relay outputs), analog I/O (2 voltage/current inputs, 1 current output) and no expansion slots.
- MLC1-F1616P0201 – A PLC with built-in digital I/O (16 inputs, 16 PNP outputs), analog I/O (2 voltage/current inputs, 1 voltage/current output) and no expansion slots.

Below is a brief I/O description of each EX-Series MLC with expansion capabilities offering the option of adding additional I/O expansion modules:

- MLC1-E1616P – A PLC with built-in digital I/O (16 inputs, 16 PNP outputs).
- MLC1-E1616Y – A PLC with built-in digital I/O (6 inputs, 2 PNP/14 Relay outputs).
- MLC1-E0808Y0402T – A PLC with built-in digital I/O (8 inputs, 2 PNP/6 Relay outputs) and analog I/O (4 universal inputs, 2 voltage/current outputs).
- MLC1-E1616P0201 – A PLC with built-in digital I/O (16 inputs, 16 PNP outputs) and analog I/O (2 voltage/current inputs, 1 voltage/current output).
- MLC1-E1616N0201 – A PLC with built-in digital I/O (16 inputs, 16 NPN outputs) and analog I/O (2 voltage/current inputs, 1 voltage/current output).
- MLC1-E1616Y0201 – A PLC with built-in digital I/O (16 inputs, 2 PNP/14 Relay outputs) and analog I/O (2 voltage/current inputs, 1 voltage/current output).
- MLC2-E0404P0802T – A PLC with built-in digital I/O (4 inputs, 4 PNP outputs) and analog I/O (4 universal inputs, 4 current only inputs, 2 voltage/current outputs).
- MLC3-E – A PLC with no built-in I/O.

Below is a brief description of each I/O Expansion module that can be added to the EX-Series MLCs:

- MLE-D1600 – An expansion module with 16 digital inputs.
- MLE-D0016P – An expansion module with 16 PNP digital outputs.
- MLE-D0016N – An expansion module with 16 NPN digital outputs.

- MLE-D0016R – An expansion module with 16 relay digital outputs.
- MLE-D0808P – An expansion module with 8 digital inputs and 8 PNP digital outputs.
- MLE-D0808N – An expansion module with 8 digital inputs and 8 NPN digital outputs.
- MLE-D0808R – An expansion module with 8 digital inputs and 8 relay digital outputs.
- MLE-D0808PH – An expansion module with 8 digital inputs (including 4 high speed inputs for single phase or quadrature counter) and 8 PNP digital outputs (including 2 PWM).
- MLE-D0808NH – An expansion module with 8 digital inputs (including 4 high speed inputs for single phase or quadrature counter) and 8 NPN digital outputs (including 2 PWM).
- MLE-A0800 – An expansion module with 8 analog inputs (voltage/current).
- MLE-A0402T – An expansion module with 4 analog inputs that support voltage, current, RTD, and thermocouple), and 2 analog outputs for voltage/current.
- MLE-A0004 – An expansion module with 4 voltage/current analog outputs.

For more specific information, please see *The MLC Series Quick Start Guide* included with each MLC product. For more information regarding installing the I/O expansion modules or configuring for High Speed Counters digital inputs, Pulse Width Modulated digital outputs, or the various analog input/output modes, consult the *I/O Module Guide for the MLC PLC Series*.

New models are already planned for the future. For the latest list of MLC Series models supported by the MAPware-7000 software, please visit our website at: www.maplesystems.com.

External HMI or 3rd Party PLCs Supported

The MLC Series support serial ports that allow you to connect an external HMI or a third party PLC (such as Allen Bradley MicroLogix™, GE Fanuc VersaMax™, Schneider Electric Modicon™, etc.) to the unit. Maple Systems supports a wide range of HMIs and industrial PCs that will interface with the MLC Series PLCs. For more information, please visit the *Controller Information Sheets* section of our website in the *Support Center*.

Projects

The MLC Series has two basic segments of internal memory. The code memory (firmware) contains the information required by the MLC that controls how it operates such as the features supported. The MLC programmer does not have the ability to change this code memory. Project memory pertains to all the ladder logic created, and any other features that the MLC programmer can create using the MAPware-7000 configuration software. Therefore, the term project designates the file sent to the MLC Series from the MAPware-7000 software.

Backing Up Projects

The MAPware-7000 configuration software does not perform automatic saving of the open project. We recommend that you frequently save your project as you are working on it to make sure that no work is needlessly lost in the event of a power failure or computer error. When you have completed a project, we strongly urge that you archive it to another folder, an external network drive, or storage media for safekeeping. Although Maple Systems does provide repair support on all of our products, we cannot guarantee that we will be able to restore a project on a damaged unit.

About Your Documentation

Maple Systems provides many resources to allow you to get the most out of your MLC Series controllers.

- This MLC Series PLC Programming Manual (shipped with MAPware-7000 software as a PDF file) describes the general operation and features of the MLC Series using the MAPware-7000 configuration software. The IEC61131 Programming Guide, Ladder Logic Guide and I/O Module Guide are also included.
- Controller Information Sheets – Important information specific to each supported protocol when connecting to an external third party PLC.
- MAPware-7000 Online Help – Covers the operation of the MAPware-7000 programming software. It is always available by clicking Help Topics from the Help menu in MAPware-7000.
- MLC Series Quick Start Guide – Describes general specifications and installation of the MLC Series. It is included with every MLC Series PLC and is available for download from our website.

For more information, please visit the Support Center section on our website at: http://www.maplesystems.com/support.htm. The Support Center provides links to manuals, FAQs, technical notes, training videos, sample projects, controller information sheets, and controller cable diagrams.

Conventions

When using MAPware-7000, there are usually several ways to perform a task. For example, if you want to copy an object, you can:

- click the Copy command on the Edit menu
- click the Copy button on the Standard toolbar
- press the CTRL+C keys on your computer

In most cases, we will describe each method when the task is introduced. The menu method is then used whenever the task is used in later procedures. Other conventions used in this manual are listed below.

Convention	Meaning
Bold	Characters that you must type exactly as they appear. For example, if you are directed to type **a:\setup**, you should type all the bold characters exactly as they are printed.
Italic	Placeholders for information you must provide. For example, if you are directed to type *filename*, you should type the actual name for a file instead of the word shown in italic type. Italics are also used to indicate a glossary term.
ALL CAPITALS	Directory names, files names, key names, and acronyms.
KEY1+KEY2	A plus sign (+) between key names means to press and hold down the first key while you press the second key.
Click	Refers to clicking the primary mouse button (usually the left mouse button) once.
Double-click	Refers to quickly clicking the primary mouse button (usually the left mouse button) twice.
Right-click	Refers to clicking the secondary mouse button (usually the right mouse button) once. Right-clicking usually opens shortcut menus.
Icon	**Meaning**
📑	Indicates a reference to additional information.
📝	Indicates an important note.

About This Manual

This manual is designed as a training manual as well as a reference guide. In this effort, the manual tries to minimize the amount of time required to become familiar with the MLC Series product line and how to configure these products using the MAPware-7000 programming software. Therefore, we strongly recommend that you read each chapter of this manual in the order that it is presented. To minimize redundancy, concepts explained in a chapter are not repeated in later chapters.

Through experience and many years of customer feedback, we have found that the quickest path to finish your project is to skim through the entire manual. By doing this, you will become familiar with the layout of the manual as well as learn basic concepts and operation of the product.

Next, install the MAPware-7000 software onto your computer and try creating a simple project. *Chapter 2 – A Quick Start* provides you with a step-by-step example of creating a project for an MLC1-F0808Y model. Even if you do not intend to use that particular model, we recommend you spend the time to go through this chapter, as it will be time well spent.

Do not forget to take full advantage of the Help files in MAPware-7000. These are a good source of information when you quickly want to research some particular feature.

Finally, our technical support staff is available by phone or by email to assist you if you run into any problems not covered by this manual. Visit our website at www.maplesystems.com for contact information.

What You Need

The following items are needed to configure and operate your MLC Series.

Configuration Software	MAPware-7000
I/O Expansion Modules	All expansion modules for the MLC Series begin with a Maple P/N of MLE; visit our website for a complete listing of all available modules
Configuration Cable	USB download cable or Serial download cable For MLC1 and MLC2 models, USB Type A to micro Type B (Maple P/N 7431-0119) For MLC3 models, USB Type A to mini Type B (Maple P/N 7431-0115)
Personal Computer	User Provided[1]
Power Cable	2-conductor 18 AWG, shielded, no connectors (Maple P/N 6030-0009)
24VDC Power Supply	User Provided (or available from Maple Systems)
HMI (optional)	Maple Systems provides a wide selection of HMIs. Please call and ask to speak to one of our sales representatives for more information.
Controller Information Sheets	Maple Systems provides Controller Information Sheets specific to communicating to third party PLCs. Please locate the sheet that corresponds to your PLC on our website.
Communication Cable	Refer to our website (www.maplesystems.com) for a list of available cables.

[1] Computer requirements include a Pentium 800 MHz or higher processor, 256 MB RAM, 800 MB available Hard Disk Space, 1024x768 resolution color monitor, Keyboard and Mouse, USB 2.0 port for downloading project, one RS232 port for online simulation, Windows 7/10 operating system.

Chapter 2 – A Quick Start

Introduction

In this chapter, we will guide you through the creation of a simple PLC project. When finished, we will show you how to simulate this project and then download the project into a unit.

For this example, we will use the following tools:

- MLC1-F0808Y
- MAPware-7000 software
- USB Download cable

This chapter assumes that you have already installed the MAPware-7000 software (as well as the USB download driver) and have some familiarity with the MLC hardware. If you need more information on the MLC Series hardware, please read *Chapter 3 – Hardware Overview*. If you need help installing the MAPware-7000 software, please read *Chapter 4 – MAPware-7000 Software Tour* before you begin.

Setting Up

The MLC1-F0808Y is a CPU module with 8 digital inputs and 8 digital outputs (6 relay, 2 PNP) built-in. This particular model is a stand-alone PLC designed for low cost applications where only a small number of inputs/outputs are required. Therefore, this model does not support expansion I/O modules. The MLC1-F0808Y has one RS232 port that can be used to communicate to an HMI or a 3rd party PLC. Another serial port on the MLC1 supports RS485 2W connections. The micro USB port is used to upload/download projects with the MAPware-7000 programming software.

Connect the MLC1 to a 24VDC power. Connect a USB download cable between your programming computer and the MLC Series USB Slave Port.

Apply power to the unit – the power LED should indicate the MLC is on. Start the MAPware-7000 programming software.

Beginning a New Project

Starting a new project in MAPware-7000 is straightforward. After MAPware-7000 has started, it will display recent projects, as well as the option to create a **New** project or **Open** an existing project.

Select the **New** project option, which will bring up the **Select Product** dialog box. Select the following and click **OK**:

- Product Series: MLC
- Product: MLC1
- Model: MLC1-F0808Y
- Programming Language: Native Ladder
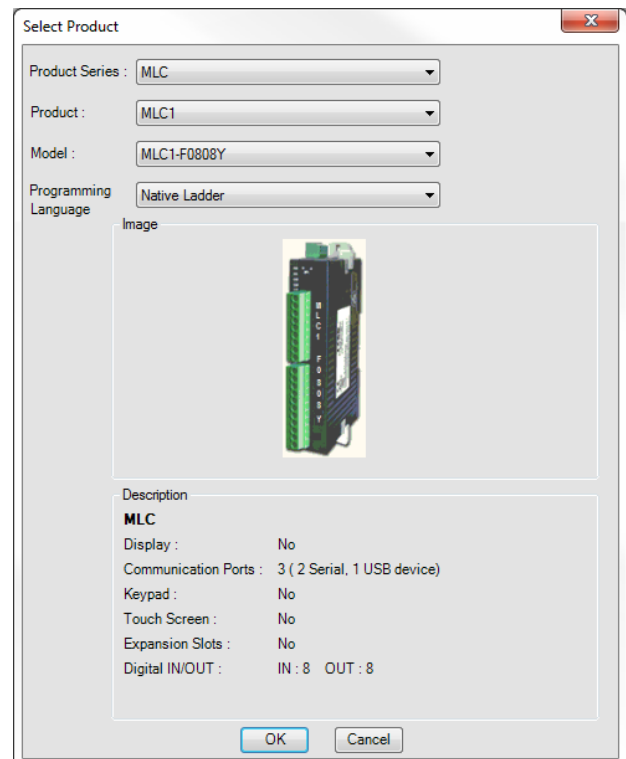
Native Ladder vs. IEC 61131-3 Logic Editors

MAPware-7000 projects use one of two programming language modes to create PLC Logic; Native Ladder mode, or IEC 61131-3 mode. Native Ladder projects contain a Ladder Logic editor which will be familiar to PLC programmers who have used this type of editor in other PLC software packages. IEC 61131-3 mode projects contain five different logic editors which are defined according to the IEC 61131-3 standard:

1. Ladder Diagram (LD)
2. Function Block Diagram (FBD)
3. Structured Text (ST)
4. Instruction List (IL)
5. Sequential Function Chart (SFC)

🗐 Information on the Logic Block editing is in Chapter 6 – Logic Blocks. Detailed lists of instructions supported in Native Ladder and IEC61131-3 can be found in the *MAPware-7000 Ladder Logic Guide* and the *MAPware-7000 IEC61131 Programming Guide* respectively. Both manuals are included with MAPware-7000 configuration software or can be downloaded from the Support Center of our website. Both programming modes are also described in detail in the help file.

> ✒ **Note:** Once a mode is selected, the project **cannot** be converted to the other mode.

Native Ladder mode was selected in the **Select Product** dialog and will be used in this chapter. The main screen area of MAPware-7000 appears:
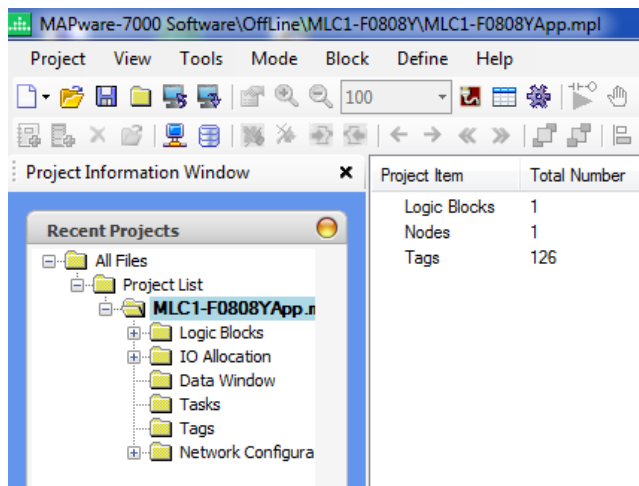
Sub-folders appear underneath the project name in the **Project Information Window**:

* **Logic Blocks** – This section is used to create the programming logic (either Native or IEC61131-3).
* **IO Allocation** – This section defines the base CPU module as well as any expansion modules to be used. In this example, the MLC1-F0808Y does not support I/O expansion modules, so the option is not available.
* **Data Window** – The Data Window feature is used to monitor the current values in selected registers when online mode is activated.

- **Tasks** – Open this folder to configure tasks that execute once during power-up or in the background while the MLC1 is running.
- **Tags** – Displays the Tag Database. Tags are used to access memory locations in the MLC1, any expansion I/O modules, and the memory areas of external HMIs or PLCs.
- **Network Configuration** – This folder assigns network node addresses to the serial ports of the MLC and is used to select and configure the communications driver that the MLC uses to connect to a 3rd party PLC or HMI.

When a new project is started, the sequence of steps is typically:

1. Select and configure the I/O Expansion modules (if any are used) in the **IO Allocation** folder.
2. Select and configure the serial port communications drivers (if any are used) in the **Network Configuration** folder.
3. Open the **Tag Database** to create the tags that will be used in your program.
4. Use the **Logic Blocks** section to create the routines that run when the MLC is in Run mode.
5. Compile and save the project.
6. Download the project to the target device.

For this example, Steps 1 and 2 are not necessary, so let's create some tags that we will use in our ladder logic.

Adding Tags to the Tag Database

1. Click the Tags folder in the **Project Information Window**:

Tag No	Tag Name	Tag	Attribute	Tag Address	Port	Node	Node Name
29	CPU error	bit	Read Only	M00016	-	0	MLC1-F0808Y
30	Program error	bit	Read Only	M00018	-	0	MLC1-F0808Y
31	Clock/Calendar illegal value warning	bit	Read Only	M00021	-	0	MLC1-F0808Y
32	Retentive data invalid warning	bit	Read Only	M00022	-	0	MLC1-F0808Y
33	Watchdog timer error	bit	Read Only	M00027	-	0	MLC1-F0808Y
34	Ladder scan time error	bit	Read Only	M00033	-	0	MLC1-F0808Y
50	HSC Enable Bit (CH1)	bit	Read Write	M00240	-	0	MLC1-F0808Y
51	HSC Reset Bit(CH1)	bit	Read Write	M00241	-	0	MLC1-F0808Y

2. Notice that we already have several tags listed. These are system tags. System tags are predefined tags that perform a specific function. For example, Tag No.68-75 indicate the state of output coils Y0-Y7 on the MLC1-F0808Y. Tag No. 35-42 are System Timer coils which cycle ON/OFF during the time interval stated. We want to add a new tag. There are two ways to do this:
   a. Click the **Add Tag** button located above the **Tag Database**:

Tag No	Tag Name	Tag	Attribute	Tag Address	Port	Node	Node Name
33	Watchdog timer error	bit	Read Only	M00027	-	0	MLC1-F0808
34	Ladder scan time error	bit	Read Only	M00033	-	0	MLC1-F0808

b. Or right-click anywhere on the **Tag Database** or on the Tags folder in the **Project Information Window** and select **Add**:
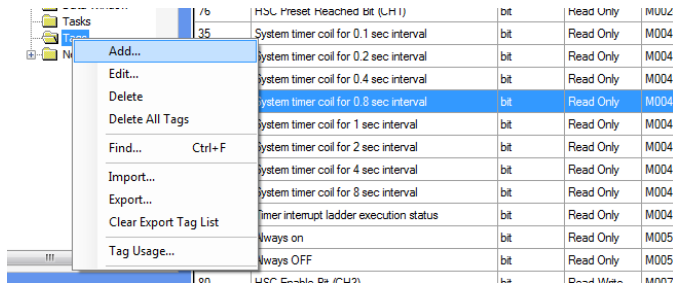
3. In the **Add Tag** dialog window, enter the following, then click the **Add** button:

4. Notice that the **Add Tag** dialog window remains open so that you may create additional tags. Let's create one more tag:

5. Click the **Add** button.
6. You can continue to add other tags but this is all we need, so click the **Close** button.

Using the Tasks Feature

Tasks are not always necessary in PLC projects but they do offer an easier way to perform a simple task that might otherwise have to be coded into the PLC logic program. In this example, we will create a power-on task that initializes a tag with a value.

1. Click the **Tasks** folder in the **Project Information Window**:
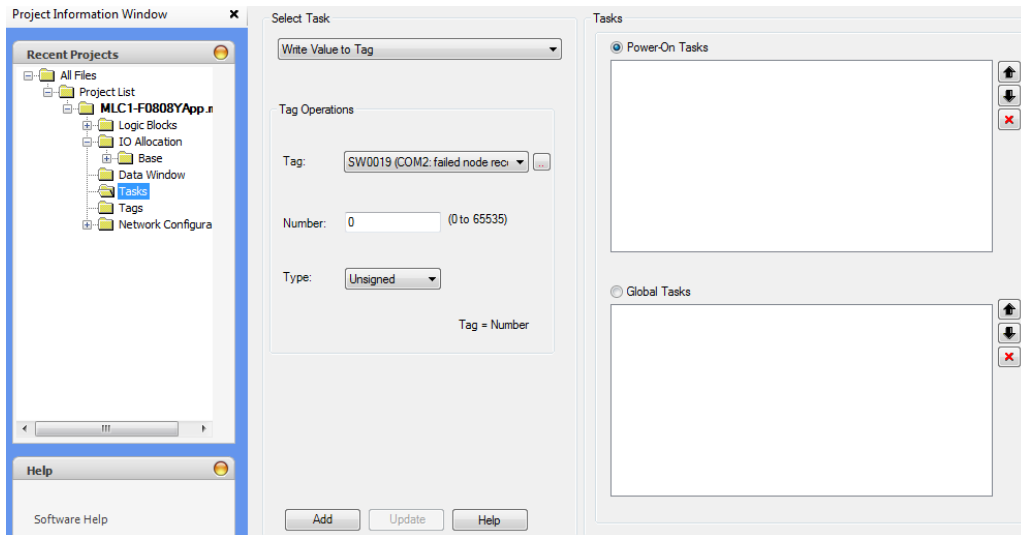
2. Click the **Power-On Tasks** radio button.
3. Under the **Select Task** field, select the **Write Value to Tag** task from the dropdown box.
4. Select the D00000 (*MyDataTag*) tag from the **Tag** dropdown box.
5. For **Number** enter 10.
6. Click the **Add** button. You should now see *Write 10 to Tag MyDataTag* in the **Power-On Tasks** window. This task executes once at start-up of the MLC.

Finally, we need to create a ladder logic program. For our example, we will write code that pulses output Y2 once per second. The program starts pulsing Y2 when power is applied to the MLC and continues to run for 10 seconds, then stops.

Create a Simple Logic Block Program

Logic blocks are another formidable feature in the MLC Series. As with a typical PLC, logic blocks interpret information from various inputs (both physical as well as internal), and then activate or control outputs (using built-in outputs, optional I/O expansion modules or controlling an external PLC) based upon that information. You can create extensive ladder logic programs and/or multiple programs that run continuously in the background or are called only when needed. This section will cover creating a simple logic block in the Native Ladder editor.

 Logic blocks can be created in the Native Ladder editor or in any of the IEC 61131-3 editors. (For more information on Native Ladder Logic Block programming, see *the MAPware-7000 Ladder Logic Guide. For more information on IEC 61131-3 Logic Block programming, see the MAPware-7000 IEC61131 Programming Guide*).

1. Click the **Logic Blocks** folder in the **Project Information Window** to expand it.

2. Click the **Block 1** subfolder under the **Main** folder to open the default block created in every new project.
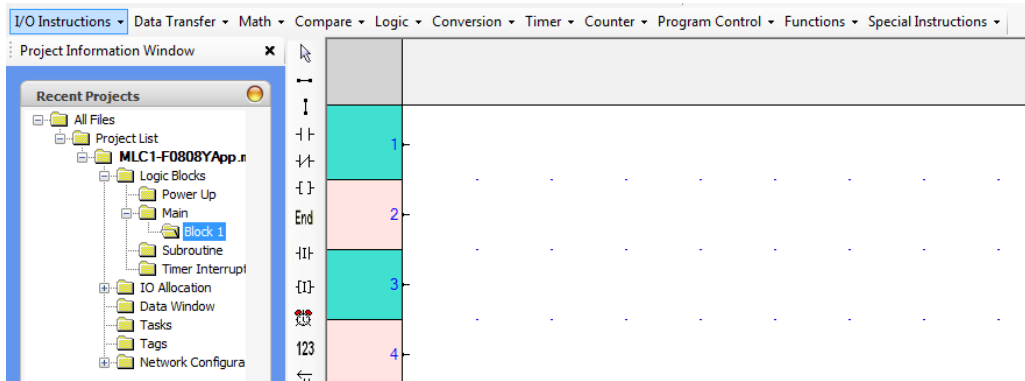
3. Notice that the numbered column on the left side is used to indicate rungs. We want to add two rungs to this program. To do this, click anywhere on Rung#1 to highlight the rung.
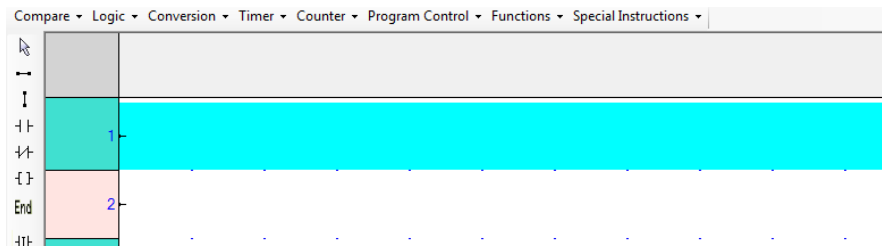
4. First, we wish to add a simple contact that is normally closed (NC) . This is found in the Quick Entry toolbar (or click the **I/O Instructions** menu, then click **NC Contact**). Click this object, then click to the right of Rung#1 to place the contact.
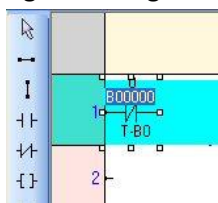
5. When the instruction is placed on the rung, a popup dialog box appears allowing you to enter the tag address required for **Operand A**. Click **OK** to accept the default B0 tag.
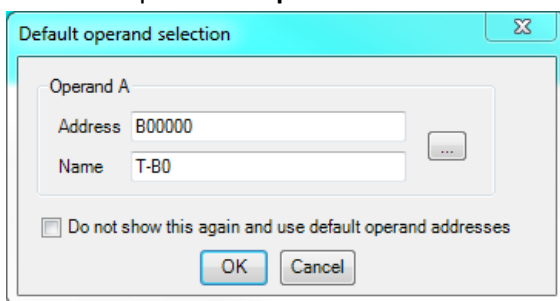
6. Now click a normally open **NO Contact** instruction to add to Rung#1. Enter *M484* for the tag address.
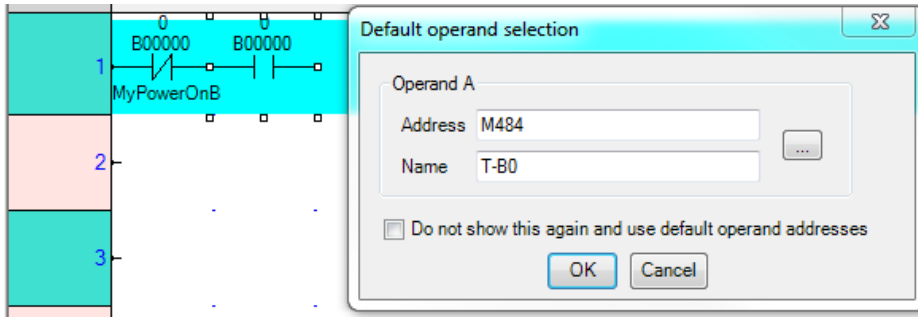
7. In the **Instruction Properties** box (off to the right), the properties for the **NO Contact** are listed.
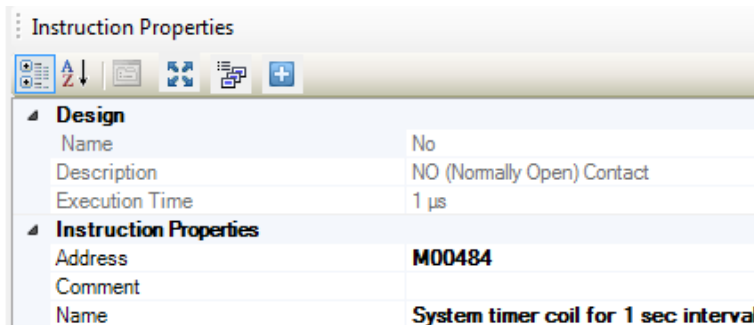
In addition to providing information, the **Instruction Properties** box can be used to enter or modify the properties of the selected instruction. Notice that the M00484 tag is a system tag that cycles between the ON and OFF states every second.

8. In the **I/O Instructions** toolbar, click the **Rising Edge** instruction and it place on Rung#1.
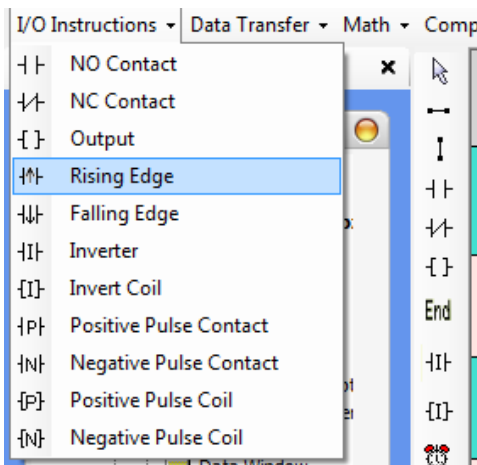
The **Rising Edge** instruction causes the following instructions to execute only once when the logic to the left of it transitions from false to true.

9. In the **Math** instructions toolbar, click the **Decrement** instruction and place it on Rung#1.
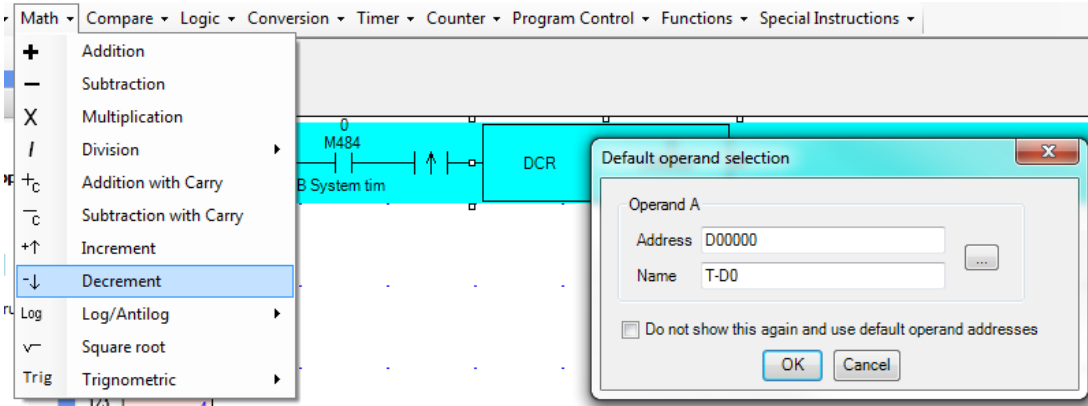
Click **OK** to accept default tag address D0. This will decrement *MyDataTag (D0)* once per second.

10. In the **Compare** instructions toolbar, click the **Equal** instruction and place it on Rung#1.

11. Leave **Operand A** assigned to tag address D0. In the **Instruction Properties** box, click the **Type** pull-down box for **Operand B** and change it from **Register** to **Constant**. Enter the value *0* in the **Constant** property field.
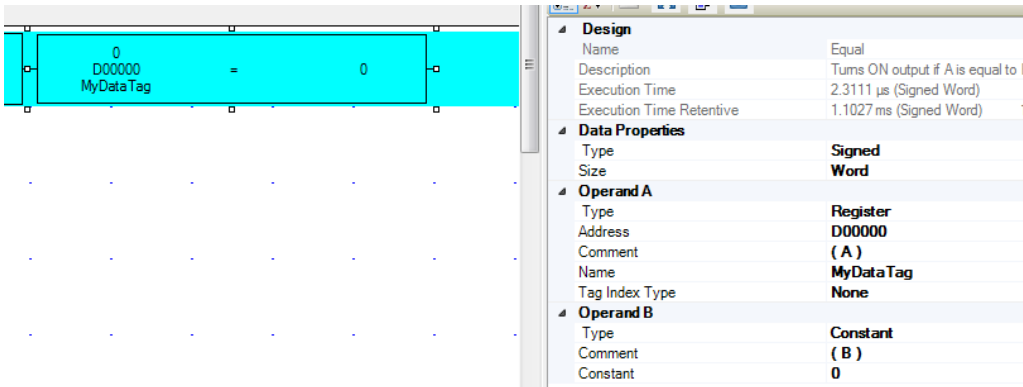
This will check to see if ten seconds have passed since power on. Remember that we set *MyDataTag (D0)* to 10 in the **Power-On Tasks** area.

12. In the **Special Instructions** toolbar, select **Data Processing** > **Device Set**. Place this instruction on Rung#1. The **Device Set** instruction sets the specified tag to true (ON or 1).

13. Change **Operand A** to tag address B0.
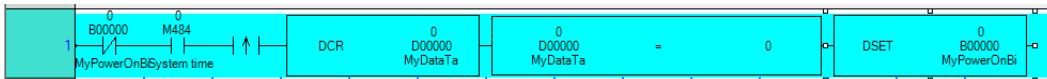
This will set the *MyPowerOnBit (B0)* tag to true once ten seconds have passed, which will disable the entire rung, since the first **NC Contact** on the rung checks for *MyPowerOnBit* to be false.

14. This completes the entries for Rung#1. For Rung #2, place the following:

   a. **NC Contact**: B0

   b. **NO Contact**: M482

   c. **Output**: Y2
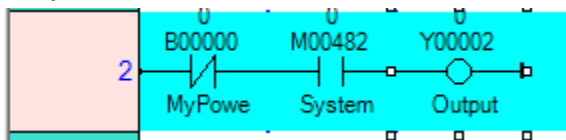
15. Finally, place the **End** instruction on Rung#3. This is found in the Quick Entry toolbar and is used to indicate the end of logic for **Block 1**.
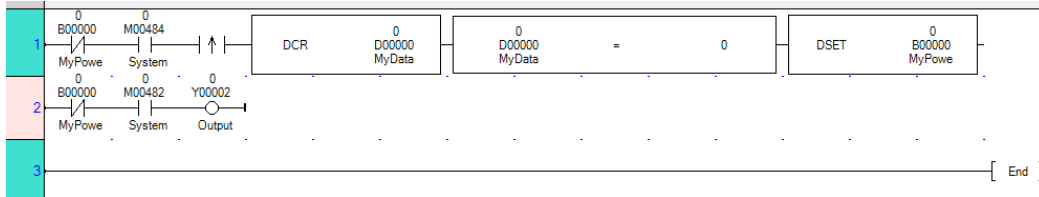
16. Each rung has a comment line that you can use to enter notes about the ladder logic or the logic block. Right-click anywhere on Rung#1 to display the context menu, then select **Show rung comment**.
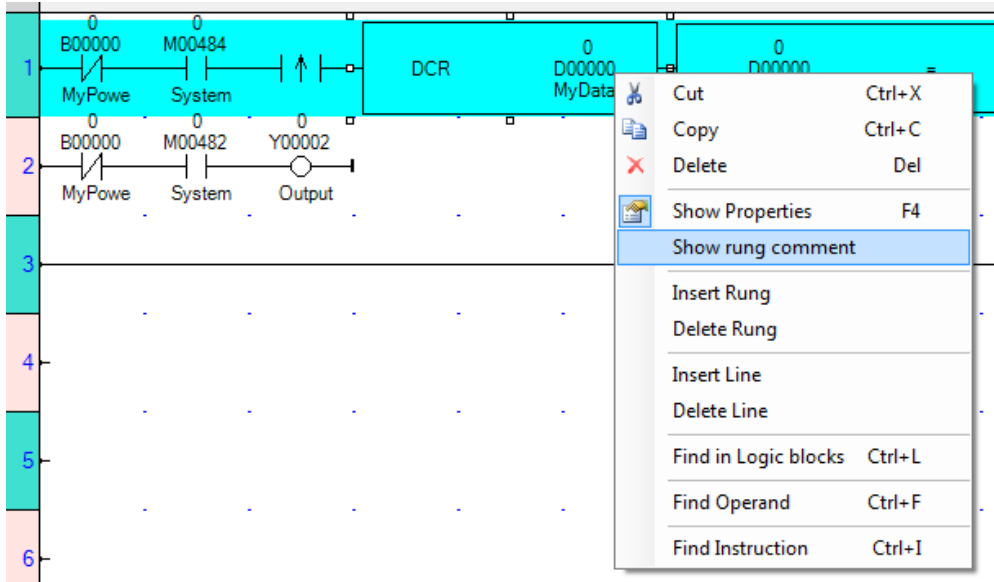
17. You can use this to enter a comment about what this logic does.
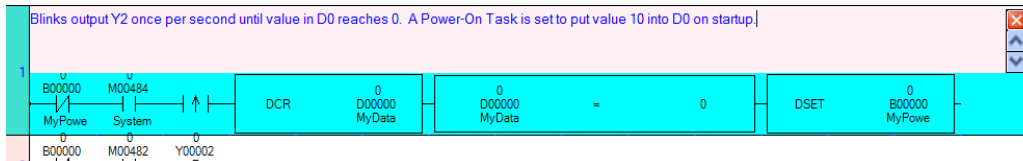
18. Finally, we need to save and compile our project. Click **Project > Save** from the menu bar. Give the project a unique name.

19. Next, select **Project > Compile** to compile the project. If the compiler finds errors, review the steps above and use the sample project called 'Quick Start for MLC' included with the MAPware-7000 software as a reference.

20. This completes your first ladder logic program for the MLC Series using MAPware-7000.

Download Your First Project

This step assumes that you are using a USB download cable and that you have already installed the USB driver onto your computer so that it can communicate with your MLC. If not, see *Chapter 4 – MAPware-7000 Software Tour*.

1. Select **Project > Transfer > Download** from the menu bar to display the **Download to device** dialog box.
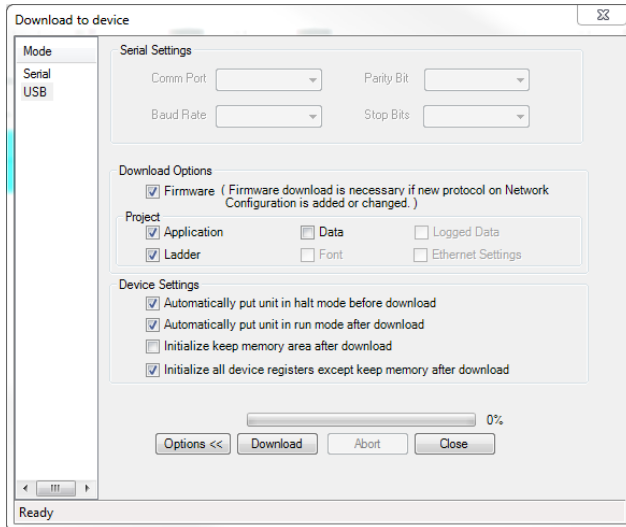
2. Under **Download Options**, check **Firmware**.
3. Under **Project**, check **Application** and **Ladder**.
4. Under **Device Settings**, check **Automatically put unit in halt mode** and **Automatically put unit in run mode**.
5. Click the Download button.
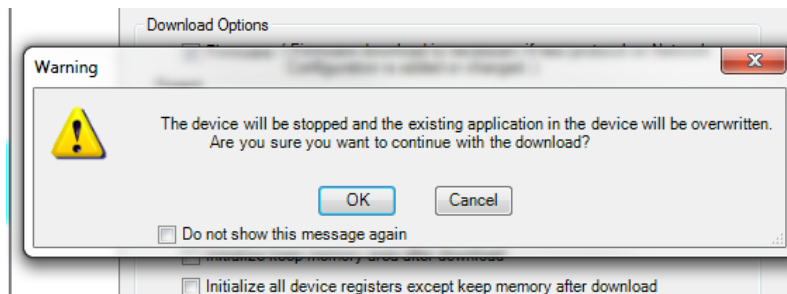
6. Click **OK**. The file will begin downloading
   📝 **Note:** Checking the **Do not show this message again** box will cause the warning box to not appear for future downloads during this session.
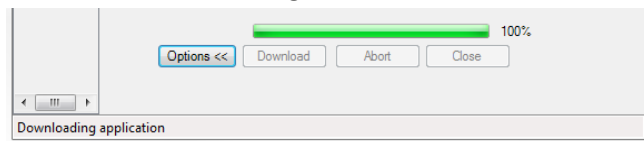
When complete, the MLC will reinitialize and run the program. You should see the LED for output Y2 blink rapidly on/off for about 10 seconds.

You can also test this project using Online Mode.

Using Online Mode to Monitor Your Project

Online mode enables you to upload the project that resides in a MLC and monitor the operation of the ladder logic that is executing in the MLC. You can also use the **Data Window** feature to view the current values in selected tag registers.

When in Online mode when using the Native Ladder editor, you are not allowed to edit the **Data Window**, so let's set up the **Data Window** before we begin.

1. With the project that we created above still open in MAPware-7000, click the **Data Window** folder in the **Project Information Window**.
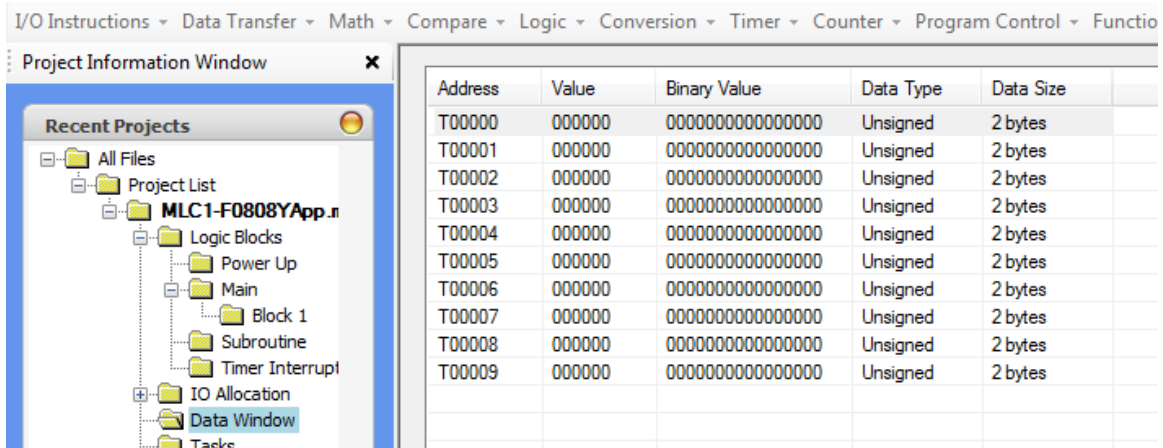
2. By default, the **Data Window** already has tag addresses T0-T9 listed. We wish to monitor all of the tag addresses used in the ladder logic program we just created:
   - B00000 (*MyPowerOnBit*)
   - M00482 (System timer coil for 0.4 sec interval)
   - M00484 (System timer coil for 1 sec interval)
   - D00000 (*MyDataTag*)
   - Y00002 (Output Coil_0002)

3. Right-click anywhere in the **Data Window** to display the context menu:
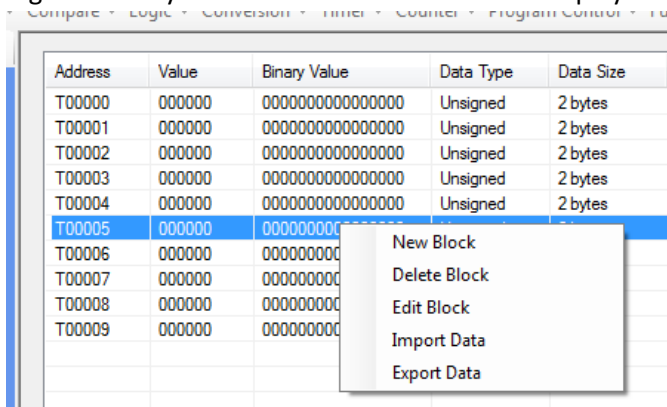
4. Select **New Block** and enter the B00000 tag address, then click the **OK** button.
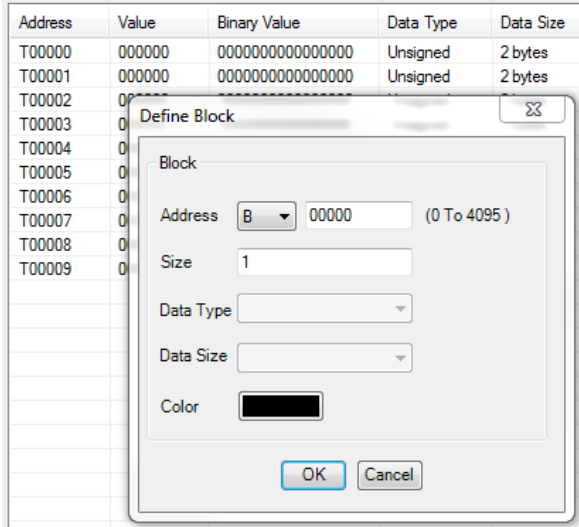
5. Enter the remaining tag addresses we want to monitor. You may also delete the T0-T9 tag addresses.
    When done, you should see the following:
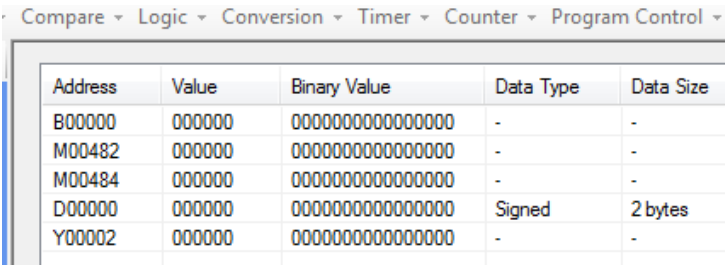
6. Now we can go online with the MLC. Select **Logic Block > Main > Block1** to view the ladder logic we
    created. Download the project to the MLC again to store the **Data Window** tag listing in the MLC.

7. Select **Mode > Go Online > With Upload** from the MAPware-7000 menu bar. The **Upload from Device**
    dialog window appears and the MAPware-7000 application attempts to upload the project that is in the
    MLC.
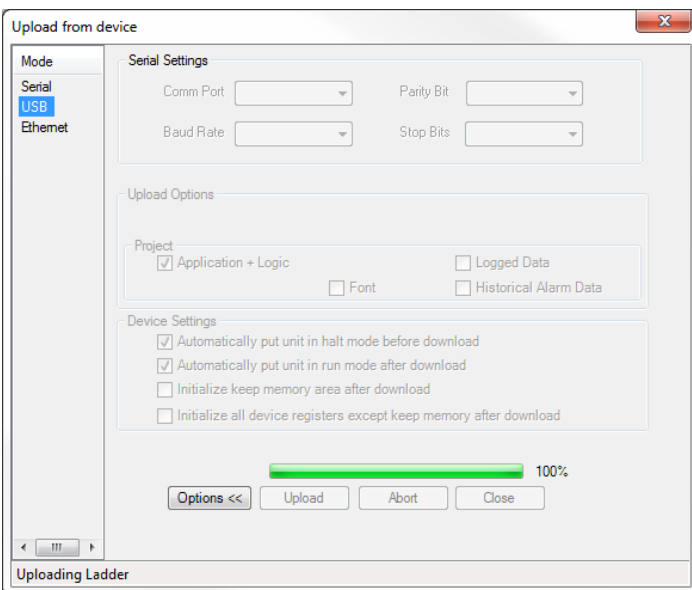
    📝 **Note:** If this is unsuccessful, select **Tools > .Preferences > Online Communication Mode** from the

menu bar. Make sure that the correct connection mode is selected, (in our example, we are using USB).
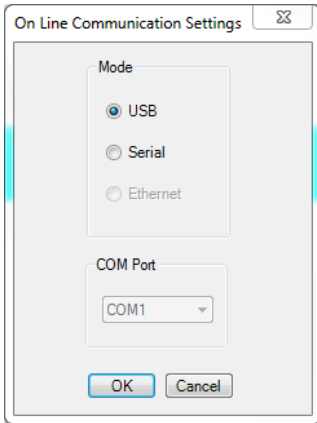
8. After the project has been uploaded from the MLC to your computer, MAPware-7000 automatically displays **Block 1** of the Main logic block.
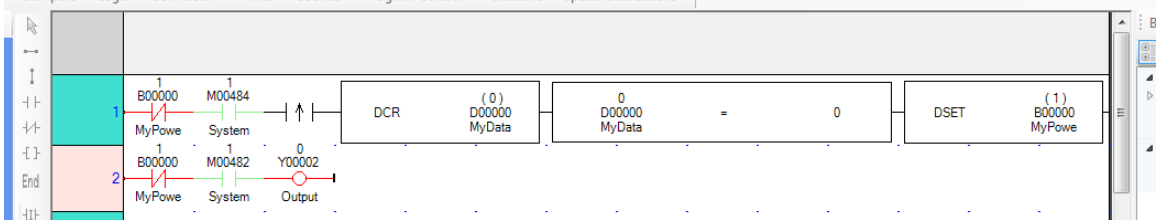
9. We want to see the **Data Window** as well. From the **View** menu, select **Data Monitor Window**:
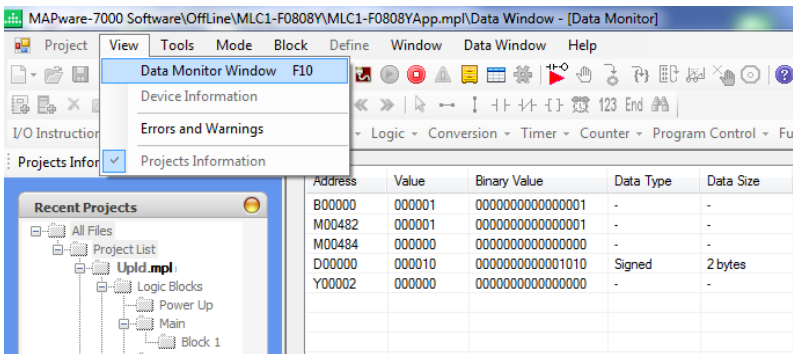
10. Now select **Window > Tile Horizontally** to display the **Data Window** and **Block 1**:
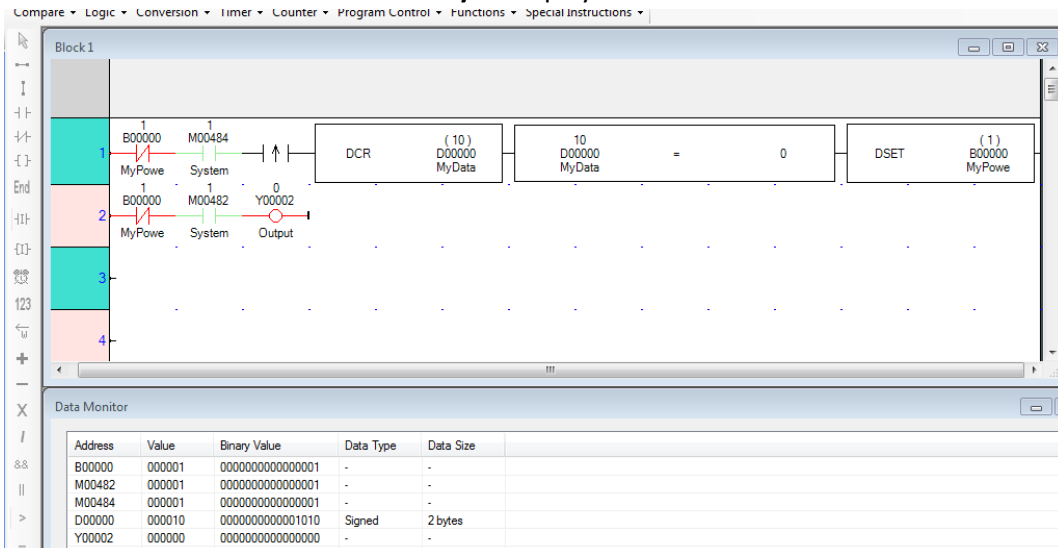
11. The Status toolbar of MAPware-7000 displays the operating condition of the MLC (run or halt):
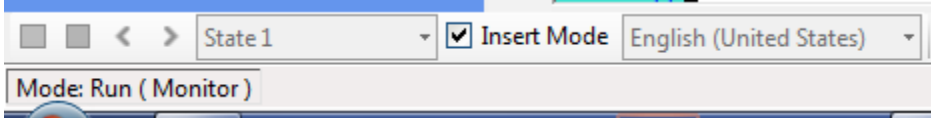
12. Notice that since you are now operating in Online Mode, you cannot edit the ladder logic. All of the ladder logic instructions and the Quick Entry toolbar are disabled (greyed out). Also note that the ladder logic displayed in Main **Block 1** now has either red or green color on each of the contacts and output coils:
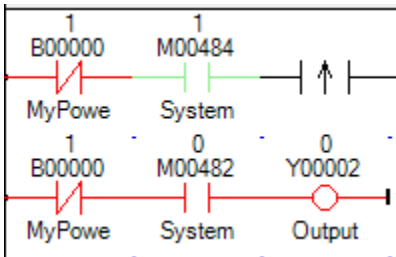
- Red: Indicates no power is flowing through that contact or coil.
- Green: The contact or coil has power flowing through it.

13. Each contact/coil also displays the current state above the tag address. Note that tags used in each instruction (i.e. Decrement, Compare [Equal], Device Set) display the current value as well.

Our program was designed to blink the Y2 output LED for approximately 10 seconds then stop. This is done by loading tag register D0 with 10 and resetting the B0 contact to 0. When the value of D0 reaches 0, the blinking of Y2 stops.

We could repeat the process by cycling power to the MLC unit but, using Online mode, we could also do this by simply placing the value 10 back into D0 and forcing the B0 contact (which is currently set) back to 0.

14. Double-click the DCR or Compare [Equal] instruction (or in the **Data Monitor** window). This causes the **Operand A** (D0) to display:
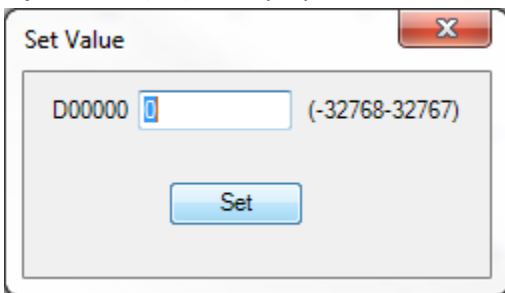
15. Although the MLC is currently running, you can force a new value into tag register D0. Enter the value '10', then press the **Set** button.

16. To get output coil Y2 to begin blinking, we must also force contact B0 to 0, since we are using B0 with a Normally Closed contact. To do this, simply double-click the B0 contact in Rung#1 or Rung#2:
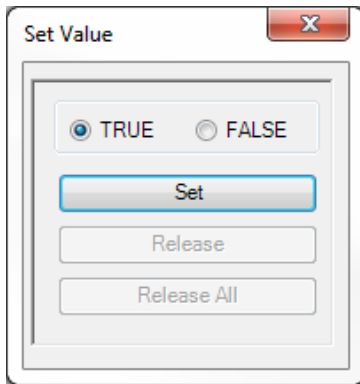
17. The Set Value dialog box opens displaying the current value of the selected contact. Click **FALSE**, then press the **Set** button.
18. Notice that the value in D0 immediately starts decrementing back to 0. If you are fast enough, you can also look at the MLC and see that the Y2 output is blinking again.
19. There are more features when using Online Mode (like Debug mode) but for now, click **Mode > Offline** to disconnect from the MLC and to allow project editing mode.

Review

Before we leave this chapter, let us review what we have learned. Although we have not covered all of the features available in MAPware-7000 or the MLC Series, we have taken our first steps in using this software and becoming familiar with the rudimentary operation. Consider what we have learned:

- We now know how to start a project, select and configure an MLC Series model.
- The **Tag Database** was used to create new tags using internal MLC memory.
- We used the **Tasks** feature to create a **Power-On Task**.
- This project showed us how to create a simple ladder logic program using the Main Logic Block.
- We know how to save, compile, and download a project into the MLC
- Finally, we know how to use the Online Mode feature (and **Data Monitor**) to view/edit tag registers that affect the operation of the ladder logic program running in the MLC unit.

The rest of this manual will go into greater detail on the many features available in the MLC Series, beginning with the next chapter which provides detailed information on the hardware of the MLC models and expansion I/O modules. Chapter 3 also provides installation advice as well as general guidelines when connecting an external HMI or 3rd party PLC to the serial port.

Chapter 3 – Hardware Overview and Installation

MLC Series PLCs are powerful programmable logic controllers. They are intended to be used in small applications that require basic controls using simple ladder logic commands or IEC61131-3 program logic. These models provide compact modules that are available in a wide variety of I/O configurations. The FX-Series models have fixed I/O built-in to the compact CPU module. They are non-expandable and are ideal in applications that require a low cost PLC with a defined number of I/O points. The MLC1-E and MLC2-E EX-Series models also have I/O built-in to the main CPU module but these models are expandable to allow the possibility of adding more I/O points as system requirements grow. The MLC3-E EX-Series unit does not have built-in I/O, but like the MLC1-E and MLC2-E supports adding MLE expansion modules. Up to 16 expansion I/O modules are supported for the MLC-E models.
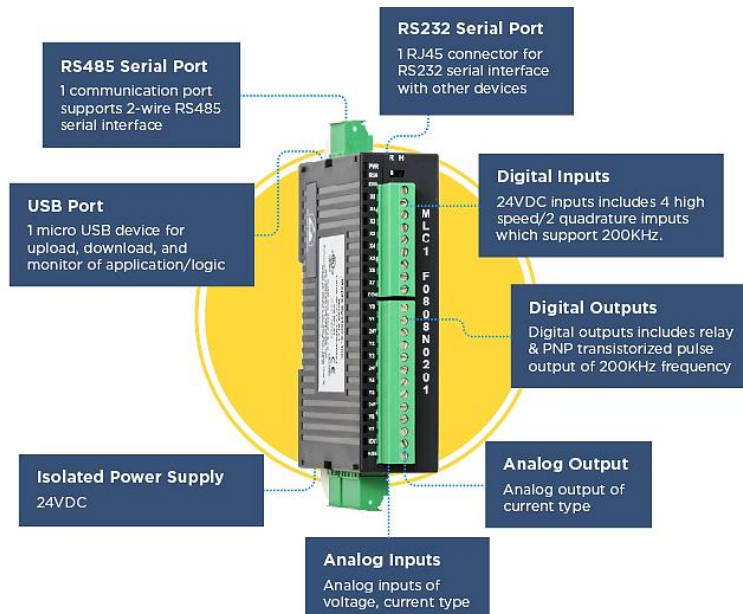
Figure 1 FX-Series Non-expandable MLC1-F model
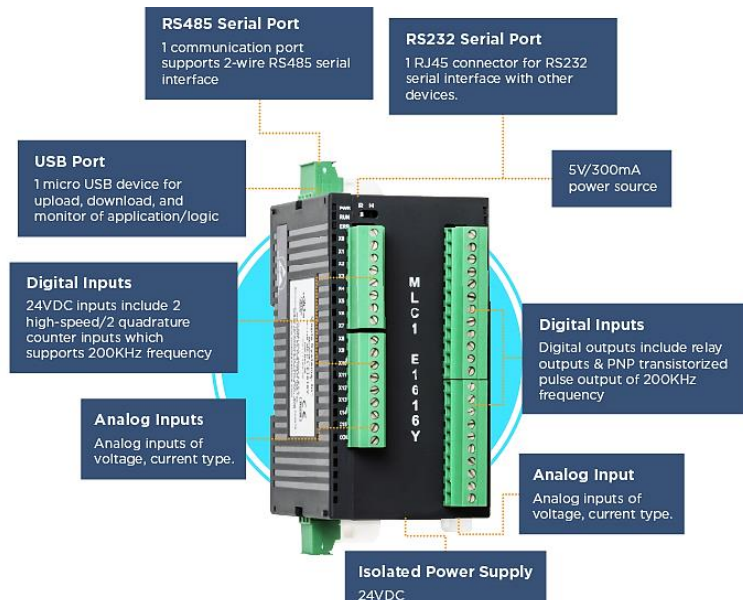
Figure 2 EX-Series Expandable MLC1-E model

Each MLC CPU base unit is composed of a microprocessor and memory to retain and run a ladder logic or IEC61131-3 program downloaded from the MAPware-7000 software. The modules require a 24VDC power source. Each MLC1 and MLC2 base unit comes with built-in I/O contacts that are wired to external devices via the pluggable green terminal block connectors. These connectors have a 3.81mm pitch (distance between each contact) with screw clamp wire connection for 30-16 AWG gauge wire. Each contact is labeled and has a red LED indicator.

Status LEDs are used to indicate power to the module (PWR), operation (RUN), and error (ERR) condition. A small toggle switch is used to change the operation of the MLC between Run and Halt mode.

Digital and analog input/output modules are available. High speed digital inputs can monitor signals configured as counters or quadrature encoders. Digital outputs are available as sink/source or relay outputs and can be configured for Pulse Width Modulation (PWM). Analog inputs can be configured to monitor voltage, current, RTD or Thermocouple measurements. Analog outputs can control voltage or current devices. Not all options are available on every model.

MLC1 and MLC2 models are programmed using the micro USB or RS232 serial port connected to a PC with MAPware-7000 configuration software. The MLC3 is programmed using a mini USB or RS232 serial cable. The MLC2 and MLC 3 also offer Ethernet. The RS232 and RS485 serial ports on the MLC Series can be used to connect to an HMI or a 3rd party PLC. DIN rail mountable, the MLC Series is CE certified and UL approved for Class 1 Division 2 applications.

Comparison of FX-Series CPU Units – Non expandable

Model	Digital Inputs	Digital Outputs	Analog Inputs	Analog Outputs
MLC1-F0604P	6 sink/source (4 high-speed)	5 PNP (1 PWM)	NA	NA
MLC1-F0604N	6 sink/source (4 high-speed)	5 NPN (1 PWM)	NA	NA
MLC1-F0808P	8 sink/source (4 high-speed)	8 PNP (2 PWM)	NA	NA
MLC1-F0808N	8 sink/source (4 high-speed)	8 NPN (2 PWM)	NA	NA
MLC1-F0808Y	8 sink/source (4 high-speed)	8 6 Relay, 2 PNP (PWM)	NA	NA
MLC1-F0808P0201	8 sink/source (4 high-speed)	8 PNP (2 PWM)	2 (voltage, current)	1 (current)
MLC1-F0808N0201	8 sink/source (4 high-speed)	8 NPN (2 PWM)	2 (voltage, current)	1 (current)
MLC1-F0808Y0201	8 sink/source (4 high-speed)	8 6 Relay, 2 PNP (PWM)	2 (voltage, current)	1 (current)
MLC1-F1616P0201	16 sink/source (4 high-speed)	16 PNP (2 PWM)	2 (voltage, current)	1 (voltage, current)

Comparison of EX-Series CPU Units with Built-in I/O – Expandable

Model	Digital Inputs	Digital Outputs	Analog Inputs	Analog Outputs
MLC1-E0808Y0402T	8 sink/source (4 high-speed)	8 6 Relay, 2 PNP (PWM)	4 (V, I, RTD, or Thermocouple)	2 (voltage, current)
MLC1-E1616Y0201	16 sink/source (4 high-speed)	16 14 Relay, 2 PNP (PWM)	2 (voltage, current)	1 (voltage, current)
MLC1-E1616P0201	16 sink/source (4 high-speed)	16 PNP (2 PWM)	2 (voltage, current)	1 (voltage, current)
MLC1-E1616N0201	16 sink/source (4 high-speed)	16 NPN (2 PWM)	2 (voltage, current)	1 (voltage, current)
MLC1-E1616P	16 sink/source (4 high-speed)	16 PNP (2 PWM)	NA	NA
MLC1-E1616Y	16 sink/source (4 high-speed)	16 14 Relay, 2 PNP (PWM)	NA	NA
MLC2-E0404P0802T	4 sink/source (4 high-speed)	4 4 PNP (2 PWM)	4 (V, I, RTD, or Thermocouple) & 4 current only inputs	2 (voltage, current)

Comparison of EX-Series MLE Expansion I/O Modules

Model	Digital Inputs	Digital Outputs	Analog Inputs	Analog Outputs
MLE-D1600	16 sink/source	NA	NA	NA
MLE-D0016P	NA	16 PNP	NA	NA
MLE-D0016N	NA	16 NPN	NA	NA
MLE-D0016R	NA	16 Relay	NA	NA
MLE-D0808P	8 sink/source	8 PNP	NA	NA
MLE-D0808N	8 sink/source	8 NPN	NA	NA
MLE-D0808R	8 sink/source	8 Relay	NA	NA
MLE-A0800	NA	NA	8 (voltage, current)	NA
MLE-A0004	NA	NA	NA	4 (voltage, current)
MLE-A0402T	NA	NA	4 (V, I, RTD, Thermocouple)	2 (voltage, current)
MLE-D0808NH	8 sink/source (4 high-speed)	8 NPN (2 PWM)	NA	NA
MLE-D0808PH	8 sink/source (4 high-speed)	8 PNP (2 PWM)	NA	NA

Accessories and Cables

Maple Systems provides 40W and 60W power supplies that are compatible with the MLC Series PLCs. The 24VDC power supplies are DIN rail mountable with current overload protection and rated for UL.

A USB cable is required to connect your PC to the MLC to download projects. Maple Systems also offers pre-made communications cables to connect your MLC to a 3rd party PLC. Visit our website for specific part numbers and pricing.

Finally, programming the MLC requires MAPware-7000 v2.34 or later.

Hardware Installation

This section will show you how to properly install your MLC in an enclosure and connect it to your control system.

Mounting the MLC in an Enclosure

When installing your MLC in an industrial enclosure, locate your controller as far as possible from power lines, load lines, and other sources of electrical noise such as hard-contact switches, relays, and AC motor drives. For more information on proper grounding techniques, see our general technical note *Ground Wiring and Electrical Noise Reduction: TN1027* available for download from the Support Center: Technical Notes section of our website.

Make sure that the unit is installed correctly and that the operating limits are adhered to (see product specifications). Do not operate this product in areas subject to explosion hazards due to flammable gases, vapors or dusts. This product should not be installed where fast temperature fluctuations are present. High humidity should also be avoided (causes water condensation on unit).

Care should be taken when locating equipment behind the product to ensure that all high voltage wires, PLC output modules, contactors, starters, relays and any other source of electrical interference are located away from the product. Particular care should be taken to locate variable speed drives and switching power supplies away from the product.

If you are installing the product into a Class I, Division 2 (Groups A, B, C, D) location, please adhere to the following guidelines:

- Do not replace or disconnect equipment unless power has been switched off.
- All wiring must comply with N.E.C. article 501.4(b).
- Circuits installed for safety reasons, like over-travel limit switches, stop push buttons, and interlocks, should always be hard-wired directly to the master control relay. These devices must be wired in series so that when any one device opens, the master control relay is de-energized, thereby removing power to the machine. Never alter these circuits to defeat their function. Serious injury or machine damage could result.

Below are the dimensions (in mm) of typical MLC/MLE units:

Figure 3 MLC1-F Dimensions (mm)   Figure 4 MLC-E Dimensions (mm)   Figure 5 MLE Dimensions (mm)
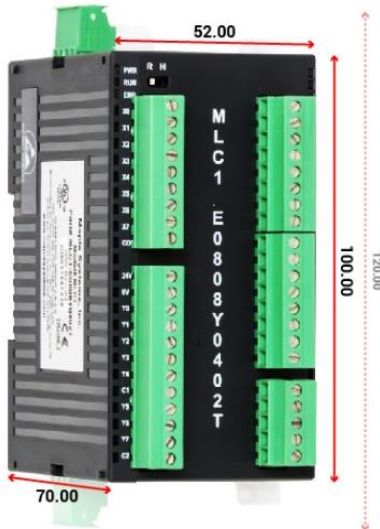
DIN Rail Mounting

MLC Series PLCs should be mounted onto a 35mm DIN rail plate. They are shipped with a DIN rail slider and locking connector attached to the unit (DIN rail not supplied).
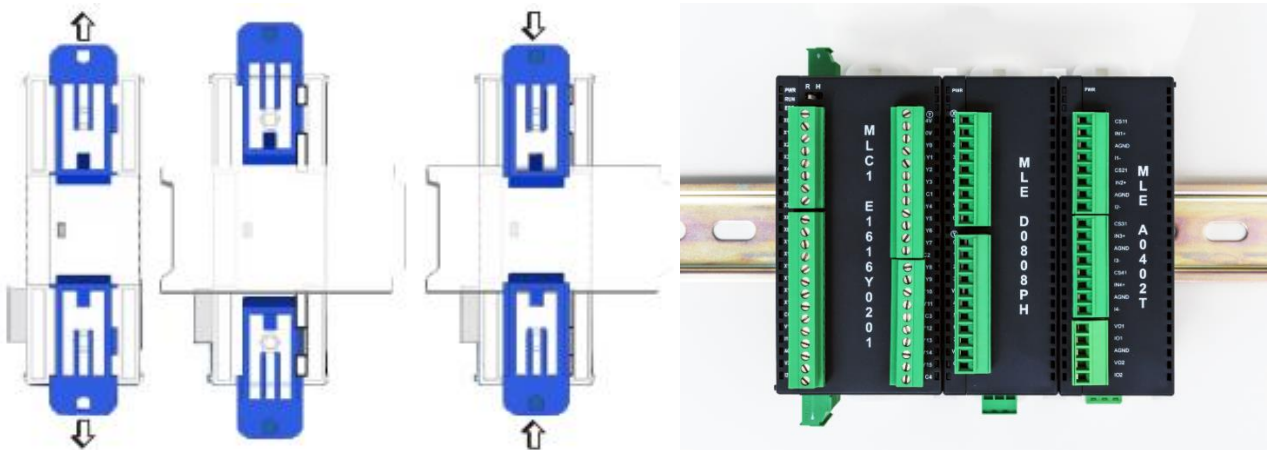
Steps to install onto DIN rail

- Pull up the sliders provided with the unit towards outward direction
- Rest the unit on the DIN rail plate
- Push down the slider again so that the unit attaches to the DIN rail.

Panel Mounting

The MLC Series can also be mounted using two screws (not included) to attach the DIN rail sliders directly to a panel.

Steps to install mounted onto a panel

- Use M4 or #8 machine screws to mount the MLC product to a panel that does not have a DIN rail.
- Make sure the DIN rail sliders on the MLC are in the open (extended) position.
- If necessary, temporarily remove any green terminal blocks that are obstructing the path for the screws.
- Locate the MLC product where you wish to install onto the panel, then mark with a pen/pencil the two places where the screws are to be inserted thru the holes of the DIN sliders.
- Pre-drill holes into the panel for the screws.
- Attach the MLC product to the panel using two screws. If necessary, reattach the green terminal blocks.

Power Supply Selection

The power supply used to power the MLC should provide an output of +24VDC $\pm$20% measured at the MLC power connector. A 24VDC regulated power supply dedicated to the MLC is recommended. Use a power supply with adequate current rating based upon your particular model (visit the Support Center Specifications page on our website).

The power supply of the MLC product should not be used to provide power to switching relays, solenoids, or other active devices. Use appropriate surge suppression devices in areas subjected to lightning or power surges. Keep all high voltage wiring, AC, or rapidly switching DC wiring separate from the MLC power and signal wires.

Connecting high voltages or AC power mains to the DC inputs will damage the product and may create an electrical shock hazard to personnel. Such a failure or shock could result in serious personal injury, loss of life and/or equipment damage.

DC voltage sources should provide proper isolation from main AC power and similar hazards.

A power line filter installed at the AC input to the MLC power supply is highly recommended as a safeguard against conducted RF noise, which is often present on factory power lines. The wires connecting the output of the power line filter to the power supply should be kept as short as possible to minimize any additional noise pickup. The case of the power line filter should be connected to a quiet earth ground. The power line filter should have a current rating of at least 3 Amps with common mode and differential mode attenuation. In applications that may have high frequency noise present, we also recommend using a resistor (~1MΩ) and capacitor (~4700pF) in parallel to clean earth ground on the DC output of the power supply.
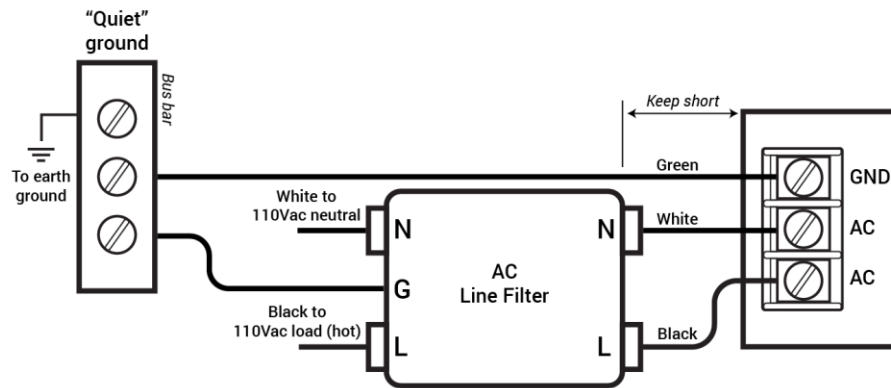
Figure 6 Power Line Filter Connection

Cable Routing and Noise Immunity

Follow these guidelines when routing cables to the MLC:

- Always route the MLC I/O wiring, communication cables, and the power cable away from any AC voltage or rapidly switching DC control lines.
- Never bundle the MLC cables together with 120VAC power wires or with relay wiring.
- Try to keep at least 8 inches (20 cm) of separation between the MLC cables and other power wiring. If voltages greater than 120VAC are used in the system, greater separation is required.
- If the MLC cables must come near AC wiring, make sure they cross at 90 degrees.
- Run AC power wires in a separate grounded conduit to reduce electrical noise interference.
- Keep the cable lengths for the MLC as short as possible. Do not coil excess cable and place it next to AC powered equipment.
- Cover any equipment used in the enclosure that operates at high frequency or high current levels with a grounded metal shield.

Connect the MLC to Power

Use the separate 3-position terminal block supplied to provide power to the MLC.

The power cable for the MLC should be 18AWG, 2-conductor wire with a shield drain wire and protective shield (foil/braid). You may buy cable P/N 6030-0009 by the foot from Maple Systems to make your own power cable.

Always run the DC ground wire directly back to the signal return of the power supply. *Do not use the chassis ground wire as your signal return.*

To connect the MLC to power:

1. Connect the power cable to the MLC.
   a. Strip cable shield to expose 2" of the black/red wires.
   b. Strip ¼" of insulation from the black/red wires.
   c. Connect red wire to the 24V DC positive (+) input.
   d. Connect black wire to the 24V DC negative (–) input.
   e. Connect power shield wire to chassis ground input.

2. Route the power cable to the MLC power supply. The power cable should not be any longer than necessary.

3. Install the power supply wires as follows (with colors shown for Maple Systems cable P/N 6030-0009):

Color	Power Supply	MLC
Red	+Output/+24VDC	+24V
Black	-Output/+24VDC return	GND
Shield	Case ground	FG or Earth

HMI or PLC Communications

Most MLC Series PLC CPU base modules have two serial ports which can be used to connect to either a Human Machine Interface (HMI) or a 3rd party Programmable Logic Controller (PLC).

COM1 is an RS232 port with a RJ45 connector. COM2 is a four pin removable terminal connector for RS485 2W (two-wire) connections:
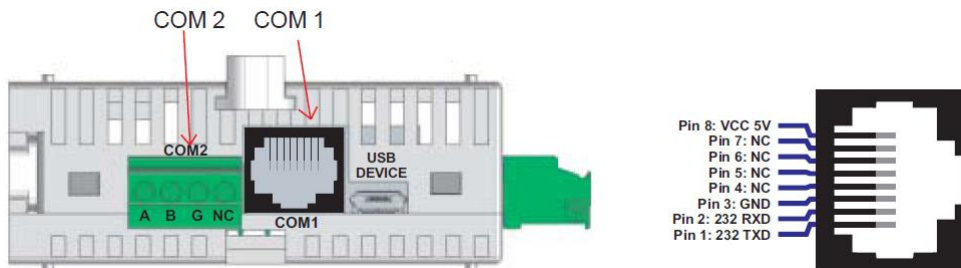
Each port can be configured to communicate to a separate device (HMI or PLC) using one of the communications drivers available in MAPware-7000.

📝 **Note:** The MLC1-F0604 models omit COM2. The MLC3-E also supports RS422/485 on COM1.

COM1 port

The COM1 port supports RS232 connection with an RJ45 jack. This port can be used to upload/download a MAPware-7000 project from a PC, connect to a HMI, or connect to a 3rd party PLC.

Pin Out Diagram (COM1) for MLC1 and MLC2 models:

Pin Number	Name	Description
1	TXD	RS232 transmit
2	RXD	RS232 receive
3	GND	Ground
4-7	NC	No connection
8	Vcc +5v	+5v Power

☞ **Note:** In MLC1 and MLC2 models, the +5V volt source pin 8 on the RJ45 connector can be used to provide power to a connected HMI that draws no more than 100mA of current.

COM1 also supports RS422/RS485 2W & 4W connections on the MLC3-E. Pin Out Diagram (COM1) for the MLC3:
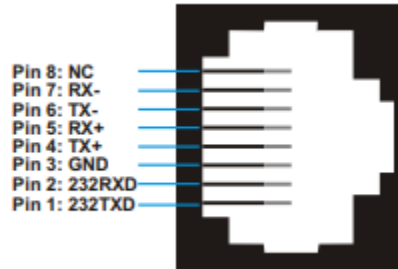
Pin Number	Name	Description
1	TXD	RS232 transmit
2	RXD	RS232 receive
3	GND	Ground
4	TX+	RS485 TX+
5	RX+	RS485 RX+
6	TX-	RS485 TX-
7	TX+	RS485 RX-
8	NC	No connection

COM2 port

The COM2 port supports RS485 2 wire connection with a pluggable terminal block. This port can be used to connect to a HMI, or connect to a 3rd party PLC.

Pin Out Diagram (COM2) for all models:

Pin Number	Name	Description
1	A (-)	RS485 Inverted
2	B (+)	RS485 Signal
3	GND	Ground
4	NC	No connection

☞ **Note:** RS422/485 4 wire connections are not supported for the MLC1 and MLC2 Series PLCs.

Connecting to an HMI

If you need an HMI for local monitoring and control of your automation system, Maple Systems provides several HMIs that are compatible with the MLC Series PLC. Then you must configure the communications driver that the MLC unit uses to connect to the HMI. This is accomplished with the MAPware-7000 configuration software, and assigning either the HMI/PLC Slave Driver (for the HMC Series) or the MODBUS RTU Slave driver (for the HMI5000 Series).

Connect to an HMC

All of the HMC products can connect to any of the MLC products using an RS232 connection:
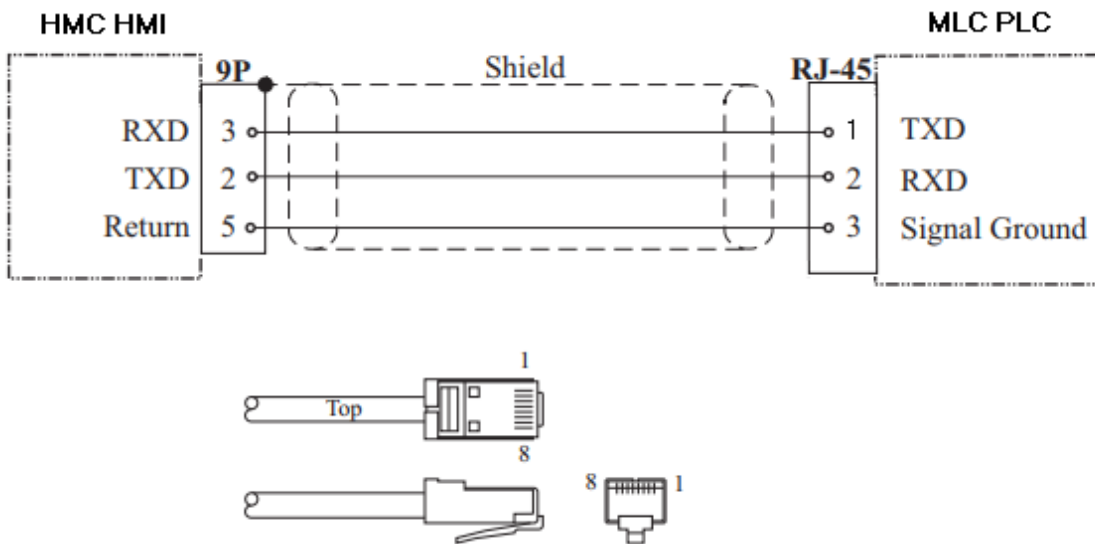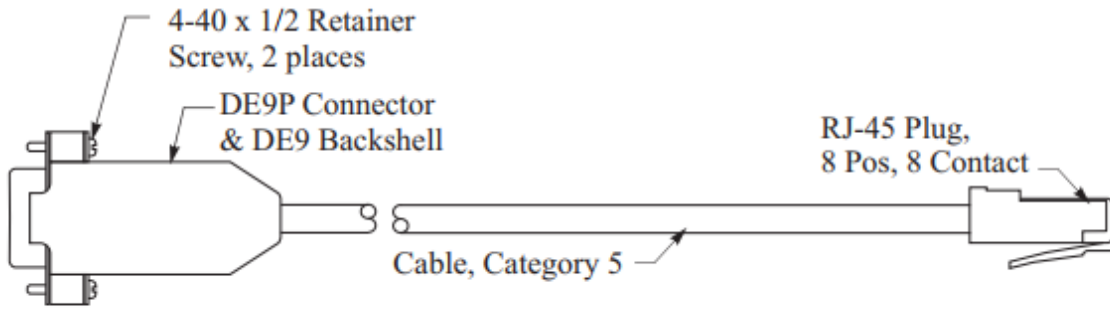
MAPware-7000 has a serial communications driver called HMC/PLC Driver which is specifically designed for communications between HMC/MLC products. To use this driver, both products must select Native Ladder as the programming language mode when creating the project. If using IEC mode, Modbus is used to communicate between the MLC and HMC.

Using the HMC/PLC communications driver

To configure the RS232 port of the MLC Series using MAPware-7000, follow these steps:

1. Open your MLC project in MAPware-7000, then right-click the **Network Configuration** folder in the **Project Information Window**.
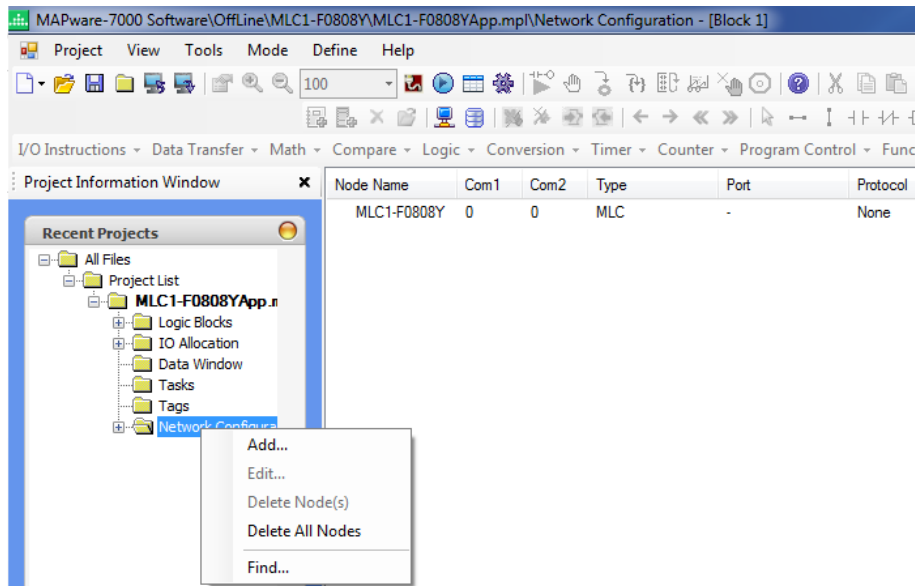
2. Click **Add** to display the **Node Information** dialog box. Configure per the picture:
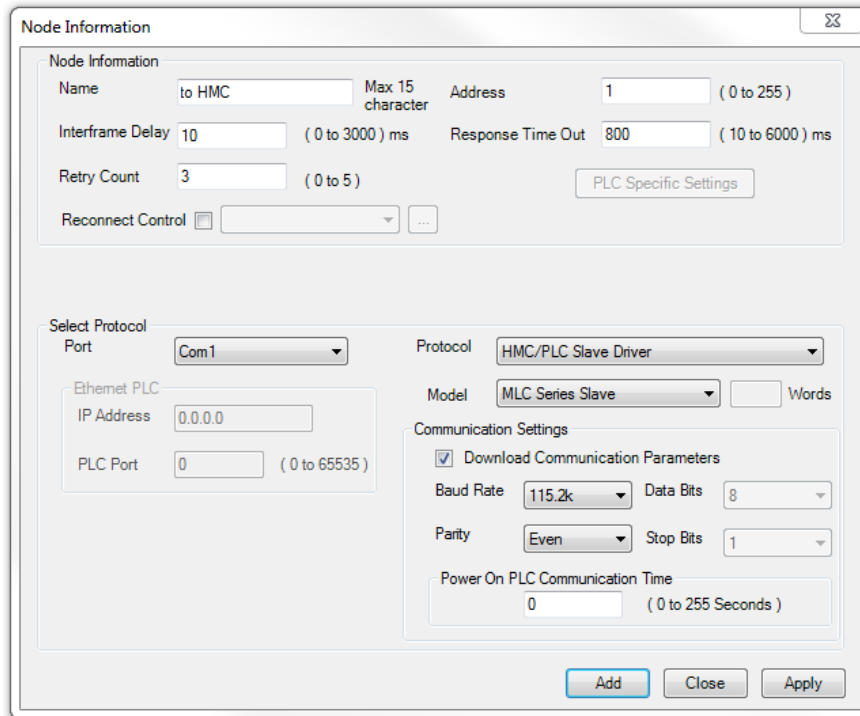
3. Click **Add** button. Then click the **Close** button.

   📝 **Note**: *If you have selected IEC 61131-3 programming language for the MLC unit, then the HMC/PLC Slave Driver is not available. Therefore, you must select the MODBUS RTU Slave driver. For more information, consult the next section.*

To configure the RS232 port of the HMC Series using MAPware-7000, follow these steps:

1. Open your HMC project in MAPware-7000, then right-click the **Network Configuration** folder in the **Project Information Window**:
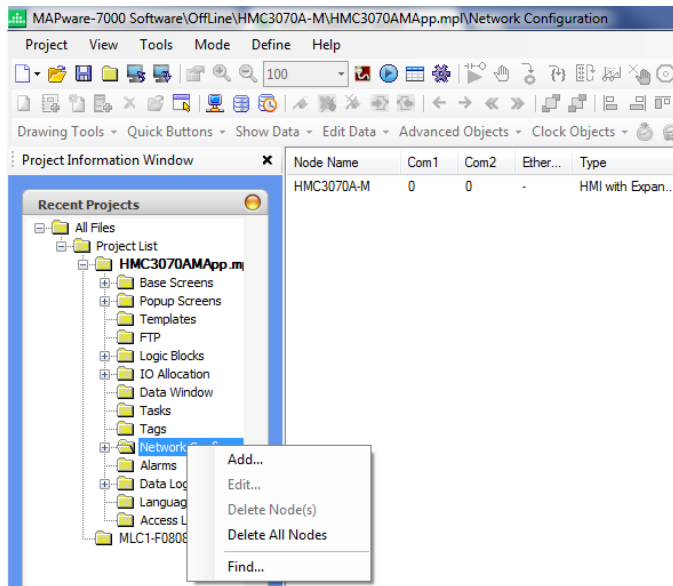
2. Click **Add** to display the **Node Information** dialog box. Configure per the picture:
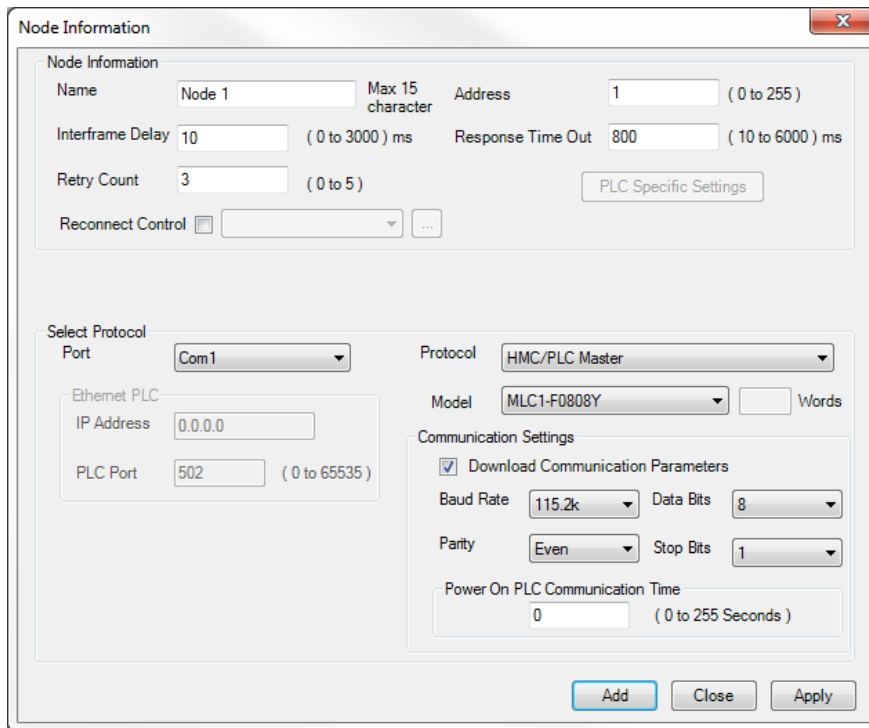
3. Click **Add** button. Then click the **Close** button.

In this configuration, the HMC communicates to the MLC in a Master-Slave serial configuration with the HMC acting as the Master. Therefore, to exchange data between the two devices, you would create tags in the HMC project that target the various memory areas of the MLC (see Chapter 5 – Tags for more information on how to use the **Tag Database**).

If either the HMC or MLC unit is configured to use the IEC61131-3 programming language, the HMC/PLC Driver mentioned above is not available. Therefore, when using the IEC61131-3 programming language, the best option for communications between the HMC and MLC products is to use the Modbus RTU communications driver with the HMC configured as the Master unit and the MLC configured as the Slave unit.

Using the Modbus RTU communications driver
To configure the RS232 port of the MLC Series using MAPware-7000, follow these steps:

1. As above, open your MLC project in MAPware-7000, then right-click the **Network Configuration** folder in the **Project Information Window**. Click **Add** to display the **Node Information** dialog box. Configure per the picture:

2. Click **Add** button. Then click the **Close** button.

To configure the RS232 port of the HMC Series using MAPware-7000, follow these steps:

1. Open your HMC project in MAPware-7000, then right-click the **Network Configuration** folder in the **Project Information Window**. Click **Add** to display the **Node Information** dialog box.

Configure per the picture:

2. Click **Add** button. Then click the **Close** button.

In this configuration, the HMC communicates to the MLC using a MODBUS RTU Master-Slave serial configuration with the HMC acting as the Master. Therefore, to exchange data between the two devices, you would create tags in the HMC project that target the various memory areas of the MLC, as though they were MODBUS registers.

Modbus RTU Slave Address Mapping with Native Ladder
If the MLC project is set to use Native Ladder Logic programming, the MLC internal registers were automatically mapped to corresponding Modbus Addresses when you selected the MODBUS RTU Slave driver. The MODBUS mapping is given in the table below:

MLC Memory Address	Range	Modbus Memory Address	Tag Length
Timer Register (T)	0-255	400001-400256	1 word
Counter Register (C)	0-255	410001-410256	1 word
System Register (SW)	0-255	420001-420256	1 word
Retentive Register (R)	0-1399	430001-431400	1 word
Input Register (XW)	0-399	440001-440400	1 word
Output Register (YW)	0-399	441001-441400	1 word
Internal Register (BW)	0-255	442001-442256	1 word
Index Registers (I, J, K)	0-2	443001-443003	1 word
I/O Configuration Register (MW)	0-1599	460001-461600	1 word
Data Register (D)	0-4095	450001-454096	1 word
System Coil (S)	0-99	020001-020100	1 bit
Internal Coil (B)	0-4095	030001-034096	1 bit
Timer Coil (T.)	0-255	021001-021256	1 bit
Counter Coil (C.)	0-255	022001-022256	1 bit
I/O Configuration Coil (M)	0-25599	035001-060600	1 bit

Simply reference the correct MODBUS address in the HMC project that corresponds to the internal memory address of the MLC unit. For example, to access Data Register 10 (D00010), the table above shows that you would create a tag in the *HMC* project that is assigned to Holding Register 450011.

Modbus RTU Slave Address Mapping with IEC61131-3

If the MLC project is set to use IEC61131-3 programming, each MLC tag register to be accessed (read or write) by the HMC must be assigned a unique MODBUS address. *In the MLC project*, follow these steps:

1. After selecting the Modbus RTU (Unit as Slave) communications driver in the **Network Configuration** folder, click the **Tags** folder.

Tag No	Tag Name	Data Type	Attribute	Tag Address	Port	Node	Com 1	Node Name	Tag Category	Export Tag
1	COM_1_status	BOOL	Read Only	SW0003_14	-	0		MLC1-F0808Y	Default Tag	☐
2	COM_2_status	BOOL	Read Only	SW0003_15	-	0		MLC1-F0808Y	Default Tag	☐
3	RTC_day_of_month	WORD	Read Only	SW0010	-	0		MLC1-F0808Y	Default Tag	☐
4	RTC_month	WORD	Read Only	SW0011	-	0		MLC1-F0808Y	Default Tag	☐
5	RTC_year	WORD	Read Only	SW0012	-	0		MLC1-F0808Y	Default Tag	☐
6	RTC_hour	WORD	Read Only	SW0013	-	0		MLC1-F0808Y	Default Tag	☐
7	RTC_min	WORD	Read Only	SW0014	-	0		MLC1-F0808Y	Default Tag	☐
8	RTC_sec	WORD	Read Only	SW0015	-	0		MLC1-F0808Y	Default Tag	☐
9	RTC_day_of_week	WORD	Read Only	SW0016	-	0		MLC1-F0808Y	Default Tag	☐
10	Scan_time_register	WORD	Read Only	SW0017	-	0		MLC1-F0808Y	Default Tag	☐
11	Carry_bit	BOOL	Read Write	S00000	-	0		MLC1-F0808Y	Default Tag	☐
12	Minute_change_status	BOOL	Read Only	S00003	-	0		MLC1-F0808Y	Default Tag	☐
13	Hour_change_status	BOOL	Read Only	S00004	-	0		MLC1-F0808Y	Default Tag	☐
14	Date_change_status	BOOL	Read Only	S00005	-	0		MLC1-F0808Y	Default Tag	☐
15	Month_change_status	BOOL	Read Only	S00006	-	0		MLC1-F0808Y	Default Tag	☐
16	Year_change_status	BOOL	Read Only	S00007	-	0		MLC1-F0808Y	Default Tag	☐
17	COM2_failed_node_reconnect_control	BOOL	Read Write	S00022	-	0		MLC1-F0808Y	Default Tag	☐
18	Driver_scan_time_register_for_port2	WORD	Read Only	SW0021	-	0		MLC1-F0808Y	Default Tag	☐
19	COM1_failed_node_reconnect_control	BOOL	Read Write	S00021	-	0		MLC1-F0808Y	Default Tag	☐
20	Driver_scan_time_register_for_port1	WORD	Read Only	SW0020	-	0		MLC1-F0808Y	Default Tag	☐
21	COM2_failed_node_reconnect_time_sec	WORD	Read Write	SW0019	-	0		MLC1-F0808Y	Default Tag	☐
22	Invalid_RTC_date_entry	BOOL	Read Only	S00019	-	0		MLC1-F0808Y	Default Tag	☐
23	COM1_failed_node_reconnect_time_sec	WORD	Read Write	SW0018	-	0		MLC1-F0808Y	Default Tag	☐
24	RTC_battery_status	BOOL	Read Only	S00010	-	0		MLC1-F0808Y	Default Tag	☐

2. Add a new tag that will be accessed by the HMI using a MODBUS address. When creating a new data tag, keep the following rules in mind:
   a. Only MODBUS 4x registers and 0x coils can be mapped to IEC data types. 3x registers and 1x coils are not supported.
   b. To map to a MODBUS 4x register, the IEC data type must be DINT, DWORD, INT, REAL, Retentive Register, UDINT, UINT, or WORD.

      c.   To map to a MODBUS 0x register, the IEC data type must be BOOL.

      d.   Each MODBUS address must be mapped to only one IEC data tag.

3. For this example, create a tag of type UINT called My4xRegister and another tag of type BOOL called My0xCoil:

Tag No	Tag Name	Data Type	Attribute	Tag Address	Port	Node	Com 1	Node Name	Tag Category
127	My0xCoil	BOOL	Read Write	-	-	0		MLC1-F0808Y	UserDefined Tag
126	My4xRegister	UINT	Read Write	-	-	0		MLC1-F0808Y	UserDefined Tag

4. Now we wish to assign MODBUS addresses to these two tags so that the HMC HMI can read/write to them. To assign a 4x register to My4xRegister, highlight the tag by clicking once. Then click the Com 1 field and enter a 4x address.

Tag No	Tag Name	Data Type	Attribute	Tag Address	Port	Node	Com 1	No
127	My0xCoil	BOOL	Read Write	-	-	0		MLC
126	My4xRegister	UINT	Read Write	-	-	0		MLC

Note: To enter a 4x address, the number must be between 400001-465535.

Tag No	Tag Name	Data Type	Attribute	Tag Address	Port	Node	Com 1
127	My0xCoil	BOOL	Read Write	-	-	0	
126	My4xRegister	UINT	Read Write	-	-	0	400005

5. Now assign a 0x coil to My0xCoil:

Tag No	Tag Name	Data Type	Attribute	Tag Address	Port	Node	Com 1	Node Na
127	My0xCoil	BOOL	Read Write	-	-	0	000003	MLC1-F0
126	My4xRegister	UINT	Read Write			0		MLC1-F0

Note: To enter a 0x address, the number must be between 000001-065535.

6. Compile and download the project to the target MLC unit. To read/write to the above MODBUS addresses using the HMC, you would simply create two tags in the HMC project that target these addresses.

Tag No	Tag Name	Data Type	Attribute	Tag Address	Port	Node	Ethernet	Node Name	
190	Tank1Temp	Register	Read Write	400005	Com1	1		to HMC	U
1	Language							HMC3070A-M	D
2	Logger_memory_							HMC3070A-M	D
3	Logger_memory_							HMC3070A-M	D
4	Logger_memory_							HMC3070A-M	D
5	RTC_status							HMC3070A-M	D
6	COM_1_status							HMC3070A-M	D
7	COM_2_status							HMC3070A-M	D
8	Historical_alarm_							HMC3070A-M	D
9	Screen_trigger_re							HMC3070A-M	D
10	Screen_saver_tin							HMC3070A-M	D
11	RTC_day_of_mo							HMC3070A-M	D
12	RTC_month							HMC3070A-M	D
13	RTC_year							HMC3070A-M	D

Add Tag dialog:
- Node Name: [to HMC] Modbus RTU (Unit as Master) (16 words)
- Tag - Name: Tank1Start   Max 255 chars
- Register/Coil Type: Coils   Read Write
- Tag-Type: ○ Register   ◉ Coil or Bit addressed Register
- Size : 1 bit   [00001-65536]
- Coil: 000003   3
- ☐ Auto Add   Number of Tags: 1
- ☐ Show Error Report

Note: The **Tag Name** assigned in the HMC project does not have to match the **Tag Name** given in the MLC project (what matters is that the MODBUS registers match).

Connect to a HMI5000 or cMT Product

All of the HMI5000 or cMT products can connect to any of the MLC products using an RS232 connection:
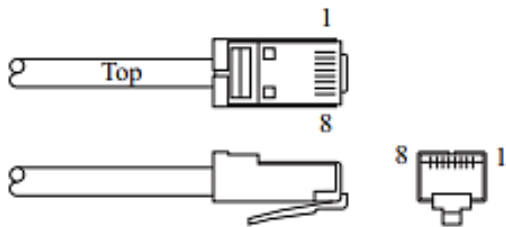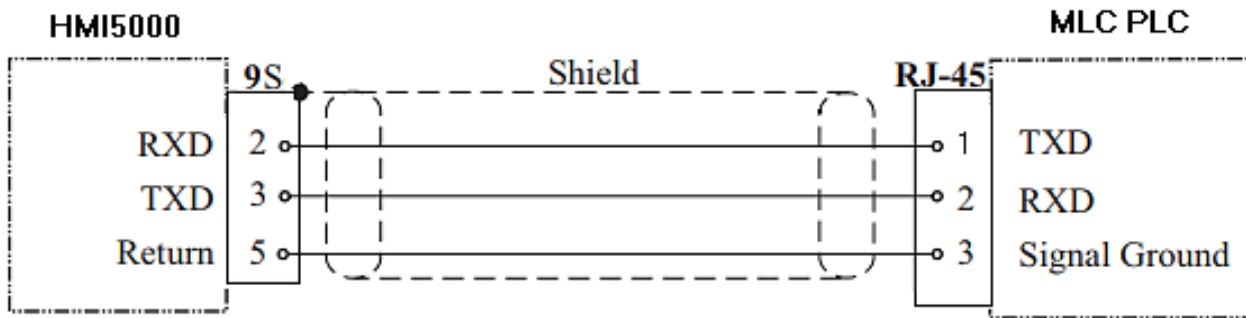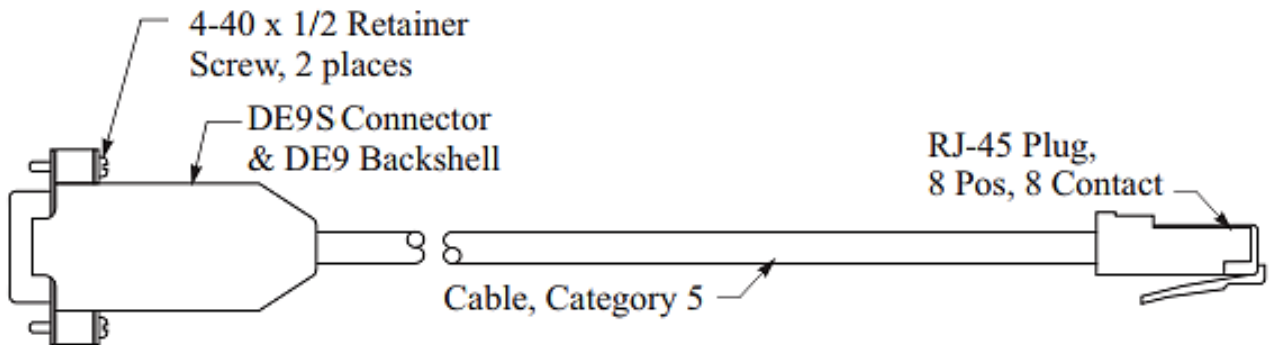
Note: The RS232 pinout for the HMI5000 above is typical but does not represent all models in the HMI5000 Series. Consult your HMI5000 Installation Guide to verify correct pinouts.

To configure the RS232 port of the MLC Series using MAPware-7000, follow these steps:

1. Open your MLC project in MAPware-7000, then right-click the **Network Configuration** folder in the **Project Information Window**:
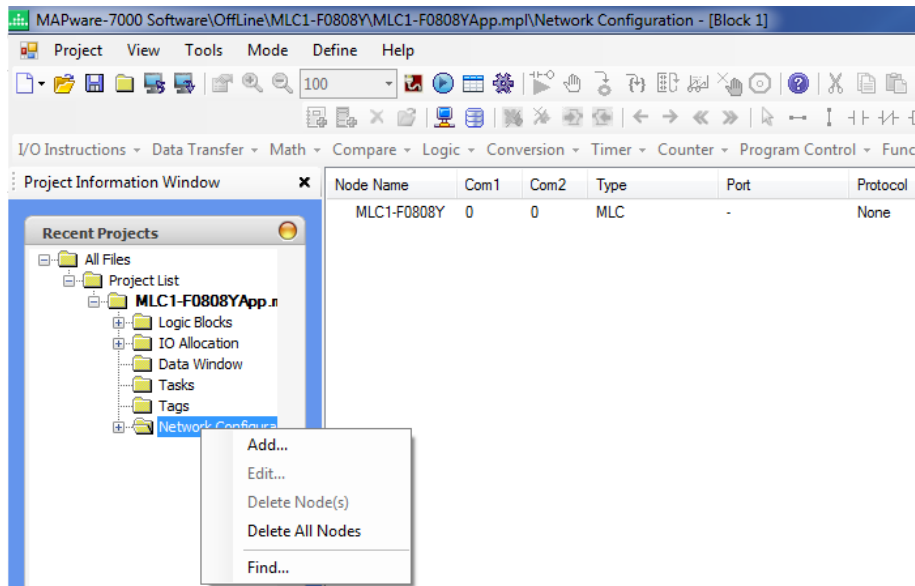
2. Click **Add** to display the **Node Information** dialog box. Configure per the picture:
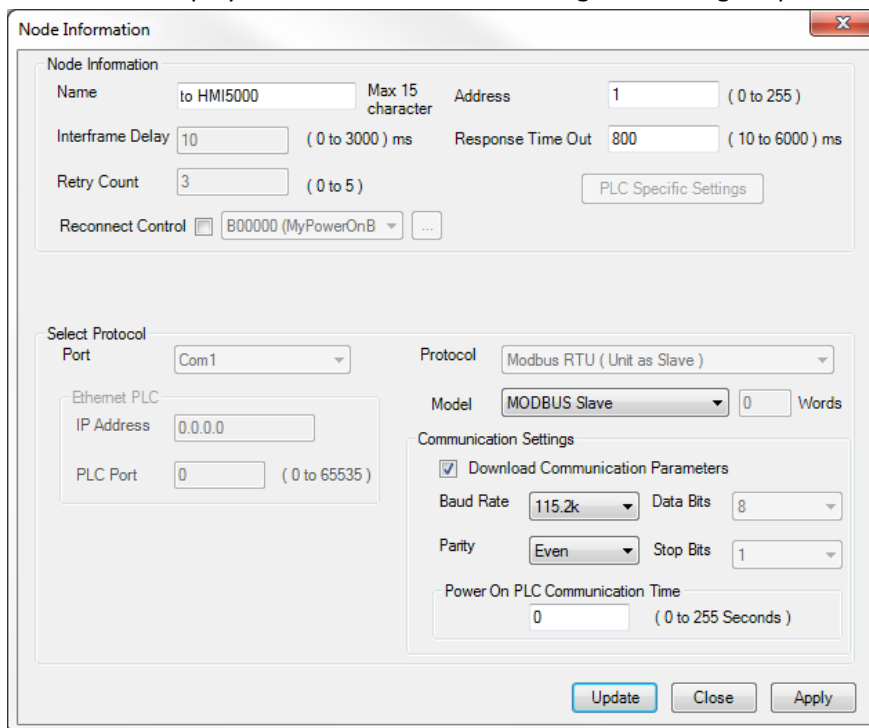
3. Click Add button. Then click the Close button.

To configure the RS232 port of the HMI5000 Series using EZwarePlus, follow these steps:

1. Open your HMI5000 project in EZwarePlus or EBPro, then click Edit > System Parameters to display the System Parameter Settings dialog box:
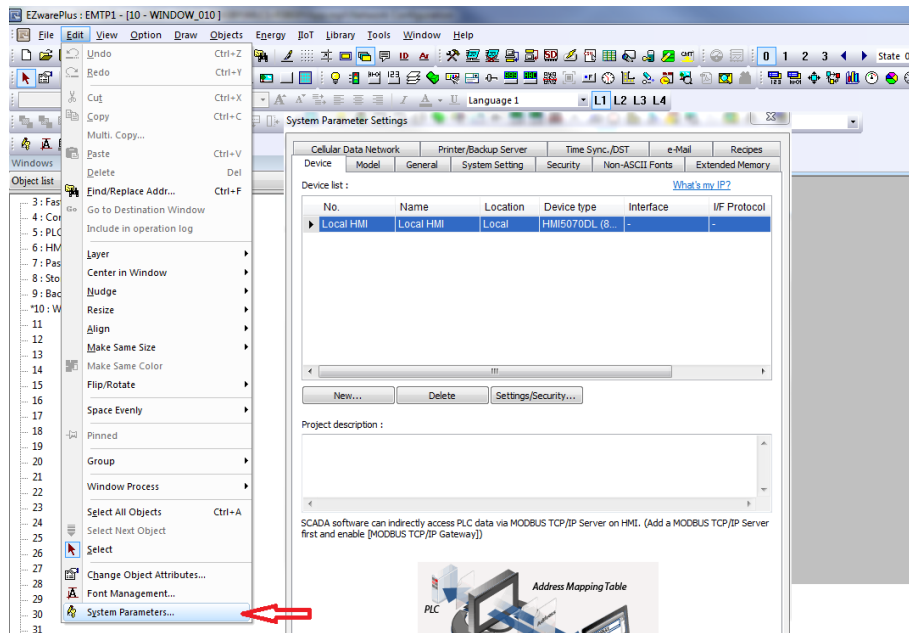
2. Click New to display the Device Properties dialog box. For PLC type select Modbus RTU Master.
3. Click the Settings button to change the RS232 baud rate to 187.5K. Configure per the picture:
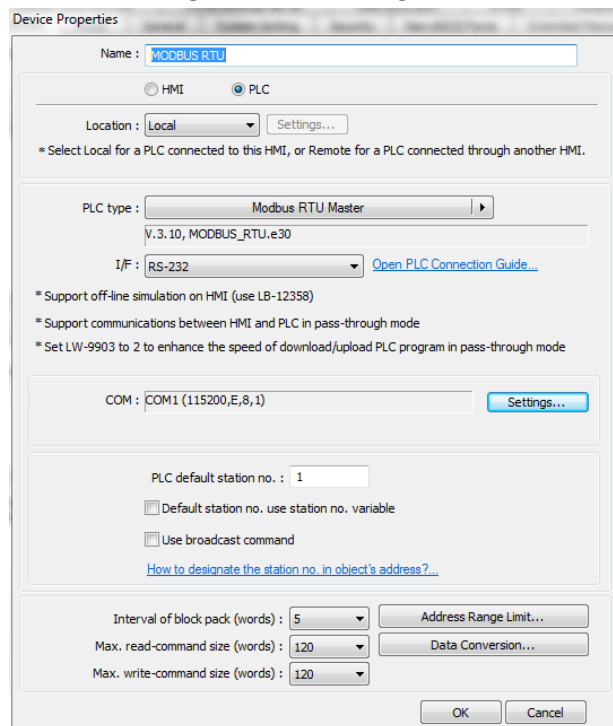
4. Click OK button. Then click the OK button again to exit the System Parameters.

In this configuration, the HMI5000 communicates to the MLC using a Modbus RTU Master-Slave serial communication with the HMI acting as the Master. Therefore, to exchange data between the two devices, you would create tags in the HMI5000 project that target the various memory areas of the MLC (consult the EZwarePlus Programming Manual for more information on how to use the **Tag Database** in the HMI5000).

Since the Modbus RTU driver in EZwarePlus does not have direct information on the MLC memory areas, you must use a Modbus RTU – MLC Memory conversion table to read/write to the memory area of the MLC unit. The conversion table varies according to what type of programming logic you are using in the MLC unit (native ladder logic or IEC61131-3):

For Native Ladder Logic:

MLC Memory Address	Range	Modbus Memory Address	Tag Length
Timer Register (T)	0-255	400001-400256	1 word
Counter Register (C)	0-255	410001-410256	1 word
System Register (SW)	0-255	420001-420256	1 word
Retentive Register (R)	0-1399	430001-431400	1 word
Input Register (XW)	0-399	440001-440400	1 word
Output Register (YW)	0-399	441001-441400	1 word
Internal Register (BW)	0-255	442001-442256	1 word
Index Registers (I, J, K)	0-2	443001-443003	1 word
I/O Configuration Register (MW)	0-1599	460001-461600	1 word
Data Register (D)	0-4095	450001-454096	1 word
System Coil (S)	0-99	020001-020100	1 bit
Internal Coil (B)	0-4095	030001-034096	1 bit
Timer Coil (T.)	0-255	021001-021256	1 bit
Counter Coil (C.)	0-255	022001-022256	1 bit
I/O Configuration Coil (M)	0-25599	035001-060600	1 bit

Simply reference the correct MODBUS address in the HMI5000 project that corresponds to the internal memory address of the MLC unit. For example, to access Data Register 10 (D00010), the table above shows that you would create a tag in the *EZwarePlus* project that is assigned to Holding Register 450011:
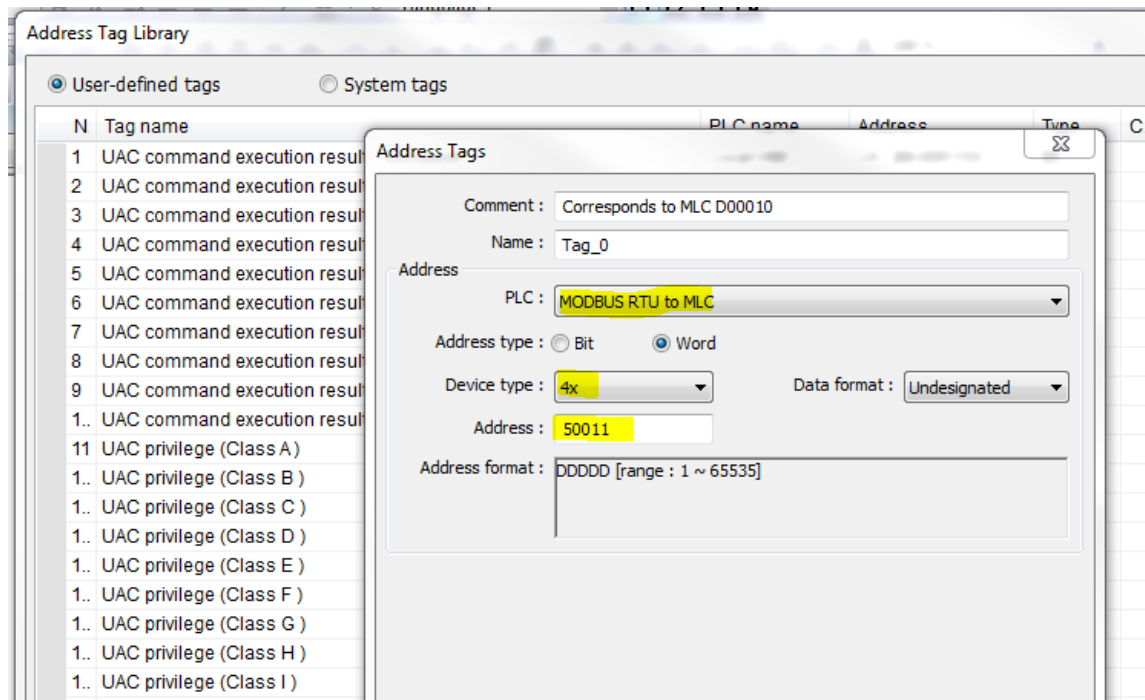
For IEC 61131-3 Logic:

If the MLC project is set to use IEC61131-3 programming, each MLC tag register to be accessed (read or write) by the HMI5000 must be assigned a unique MODBUS address. Refer to the section "Modbus RTU Slave Address Mapping with IEC61131-3" above for detailed steps on how to do this.
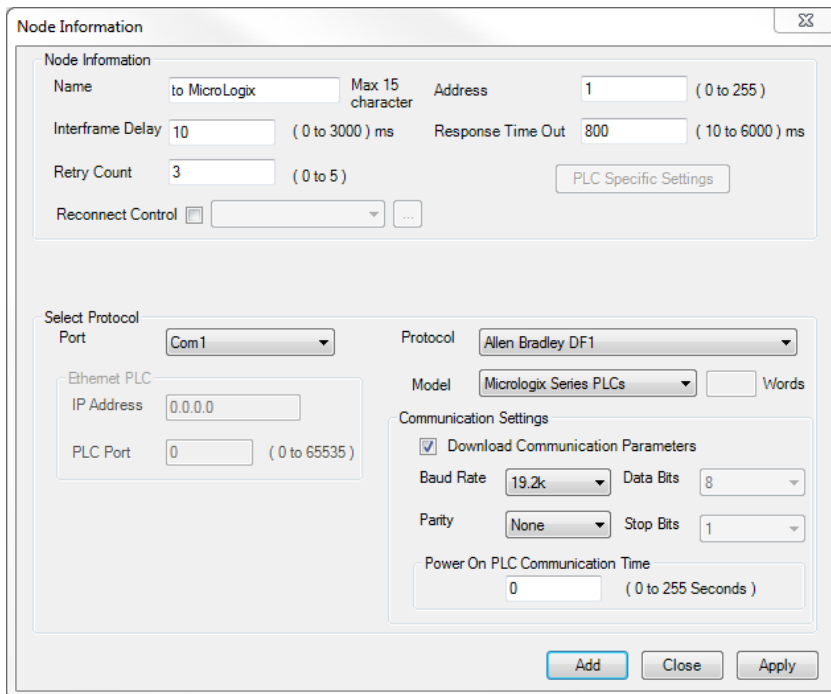
Connecting to a 3rd Party PLC

An external PLC can be connected to the MLC Series via a serial port using RS232 or RS485.
        📝 **Note:** *RS422 4-wire connection is not supported on MLC1 or MLC2 models.*

In addition to making the physical connection to the external PLC, you must also configure the communications parameters (i.e. communications driver, baud rate, etc.) using the MAPware-7000 software:

1. Open a new project in MAPware-7000 and select which MLC product you wish to use.
2. In the **Project Information Window**, right-click the **Network Configuration** tab to display the **Node Information** dialog box.

    In the **Select Protocol** section, select which MLC serial port you will use. Then choose from the **Protocol** list box the appropriate protocol driver for your PLC. Click the **Model** list box to select the particular PLC model that most closely matches your PLC.
3. In the **Communication Settings** section, enter the communications parameters that the MLC will use to attempt communications with your external PLC. For more specific information about each external PLC supported, please visit our Support Center-Controller Info Sheets at our website: http://www.maplesystems.com.
        📝 **Note:** *Although a serial printer driver is listed as one of the options in the Protocol drop-down list, this driver is not supported for any of the MLC Series*

After assigning an external PLC to one of the COM ports on the MLC, you can read/write to the internal memory of the attached PLC. To do this, you must first assign tags to the PLC memory using the **Tag Database**:
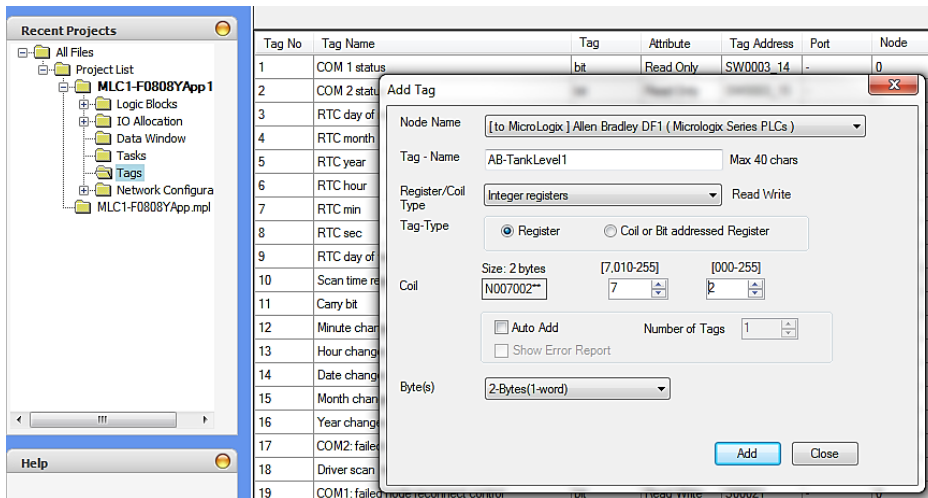
Once PLC memory addresses have been assigned to the **Tag Database**, you can then read/write to the PLC in your project.

USB Client port

This port is used solely for downloading/uploading projects into the MLC unit. The port complies with the USB 2.0 specification, and has either a Type B micro or Type B mini female connector.

EX-Series MLE I/O Modules

Twelve different types of input/output modules are available that can be added to the MLC-E base CPU modules. The I/O modules have various combinations of digital and analog connections depending upon the I/O module selected. Below are pictures of one of the digital I/O modules (MLE-D0808P) and one of the analog I/O modules (MLE-A0402T):
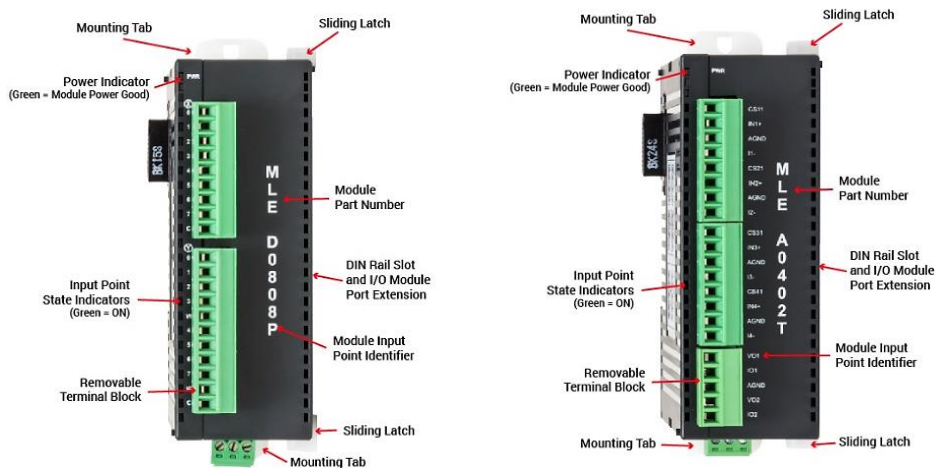
Each MLC-E base CPU module can connect to a maximum of sixteen optional MLE I/O modules.

For more information on MLE expansion modules, please see the *MLC Series I/O Module Guide (P/N 1010-1055)*.

Chapter 4 – MAPware-7000 Software Tour

Introduction

MAPware-7000 is used to program the MLC Series PLCs with optional I/O expansion modules. The software is easy to use and requires a minimal amount of time to become familiar with operation. With the MAPware-7000 software, you can quickly create projects using the various features and tools that come with the software.

> ⧉ **Note:** MAPware-7000 is also used to program Maple Systems HMC (HMI + PLC) products. This manual focuses on features available for the MLC line. For more information on features only available in HMC products, such as screens and graphic objects, see the general *MAPware-7000 Programming Manual*.

System Requirements

The following basic computer hardware is required to install and use the MAPware-7000 software

- 800 MHz Pentium-based processor or equivalent
- 256 MB of RAM (more memory improves performance)
- 800 MB (including 200 MB for the .NET Framework Redistributable) available hard disk space
- Monitor set for 16-bit color 1024x768 pixel mode
- Microsoft mouse or compatible pointing device and keyboard
- USB 2.0 port (to upload/download projects)
- Serial port (to upload/download projects – optional)
- Microsoft Windows 7 or Windows 10

Installing MAPware-7000

Use the following procedure to install the MAPware-7000 software from the CD-ROM:

1. Turn on power to your development computer.
2. Make sure no other application programs are currently running.
3. Insert the MAPware-7000 software CD into the computer's CD-ROM drive.
   Note: If 'Auto Play' mode is not set for your CD-ROM drive, click setup.exe on the root directory of the CD drive using Windows Explorer.

4. The Welcome to MAPware-7000 software screen appears.

5. Click **Next** to continue.

6. Select the preferred directory location on your computer to install the MAPware-7000 software (default is C:\Maple Systems\MAPware-7000\).

7. Then click **Next**.

8. Confirm Installation.

9. Click **Next** to begin.

10. Once installation is complete, click **Close**.

Starting the MAPware-7000 Software

Open the MAPware-7000 software by clicking the Start button in the Windows Task bar, then the All Programs button:

The MAPware-7000 Task menu provides the following:

- MAPware-7000 icon to start the software
- Help icon to access the Help files directly
- The MLC Series Programming Manual
- Uninstall icon

Click the MAPware-7000 icon to begin execution of this program. The opening screen appears:
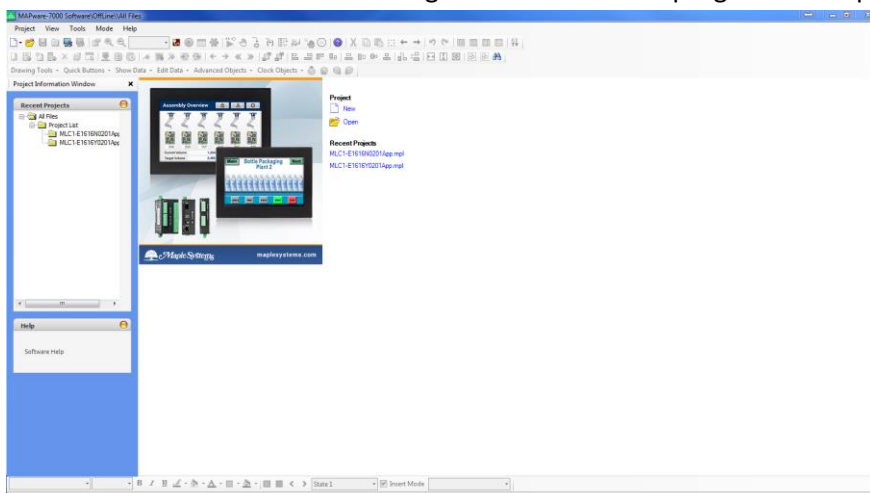
 Note: The software requires that you set your monitor resolution to 1024x768 pixels or higher.

If you have prior experience programming HMIs, you will notice that the MAPware-7000 software is intentionally designed with the same look and feel as most HMI configuration software in the marketplace today. This is done to make your transition to using the MLC Series products as trouble-free as possible. If you are new to using the MAPware-7000 configuration software, this chapter will give you a brief rundown of how to take maximum advantage of the ease-of-use this software provides.

Opening Screen

As mentioned above, the initial screen displayed when you start the MAPware-7000 software displays the overall workspace environment that you will be creating your projects in:
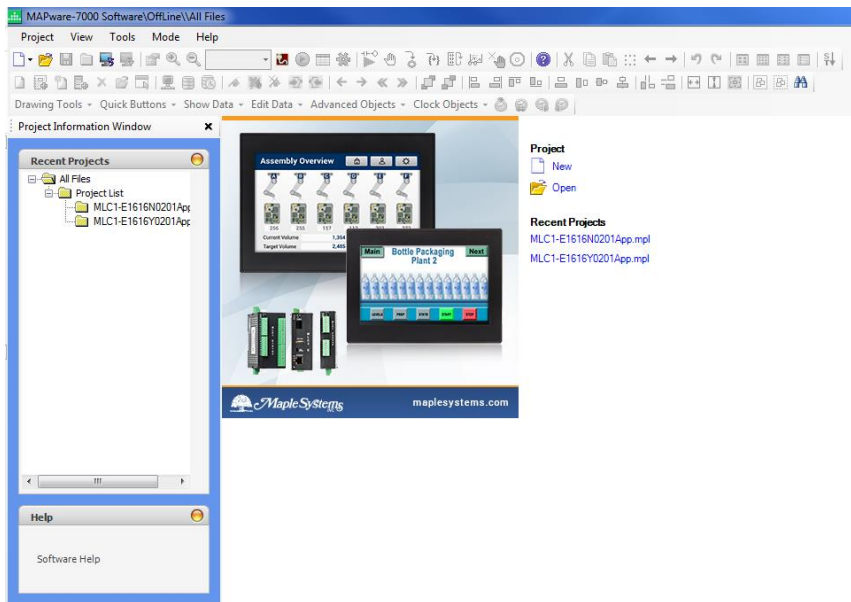
Perform the following:

- Click **New** to start a new project.
- The **Select Product** dialog box appears. Select any product.
- Select **Native Ladder** in the **Programming Language** drop down list. Click **OK**.

- Now, you should see the following:
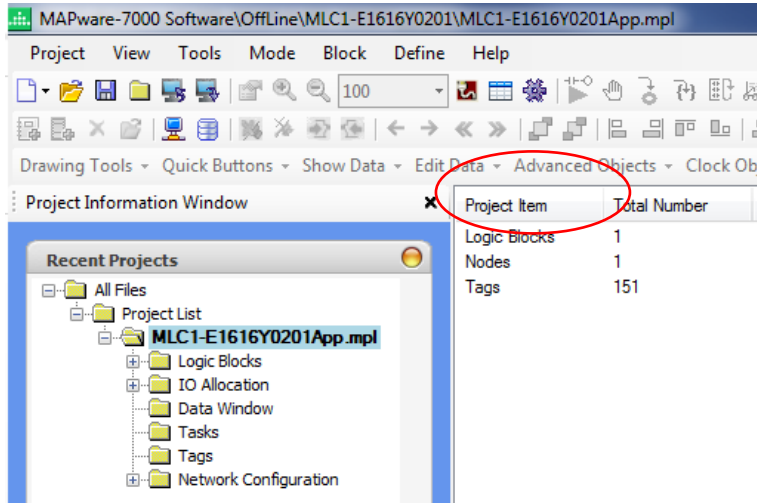
- Double-click Project Item – **Logic Blocks**. This displays a listing of all current ladder logic blocks that have been created (since this is a new project, only **Block 1** is listed):
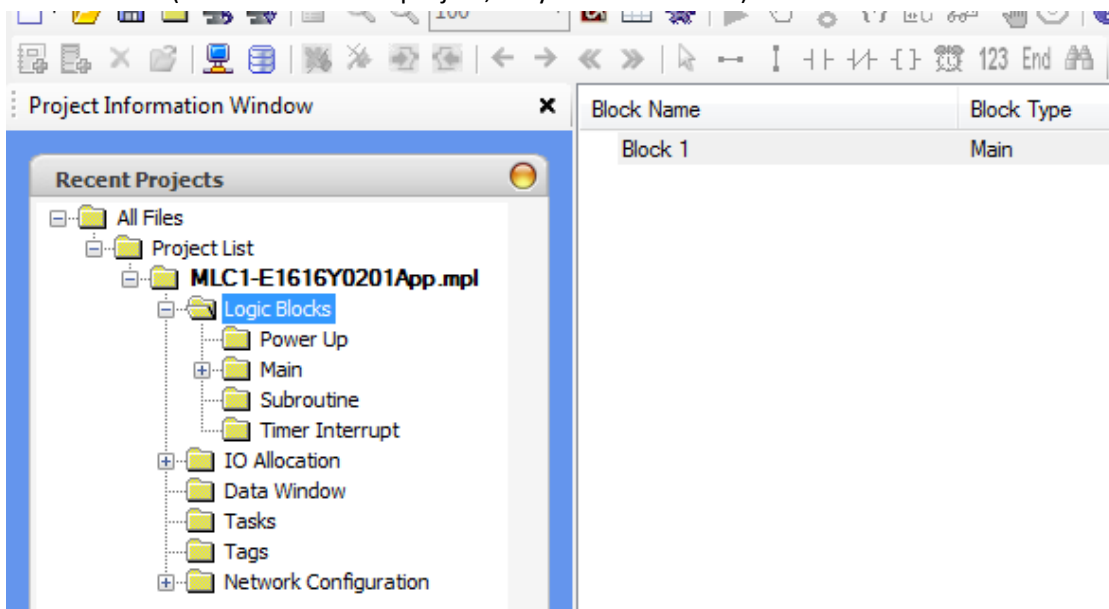
- Double-click **Block 1**. This displays a blank ladder logic diagram, along with appropriate toolbars:
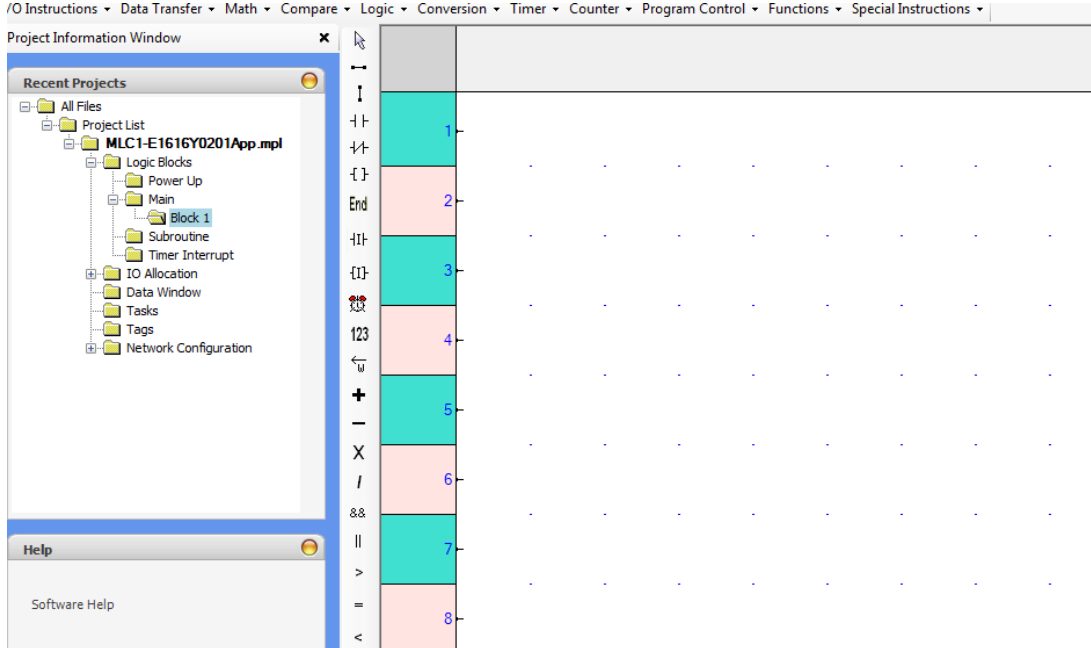
Main Editor Window

This is the primary window used to configure and edit the MAPware-7000 project. Let us describe each section of the software in detail:
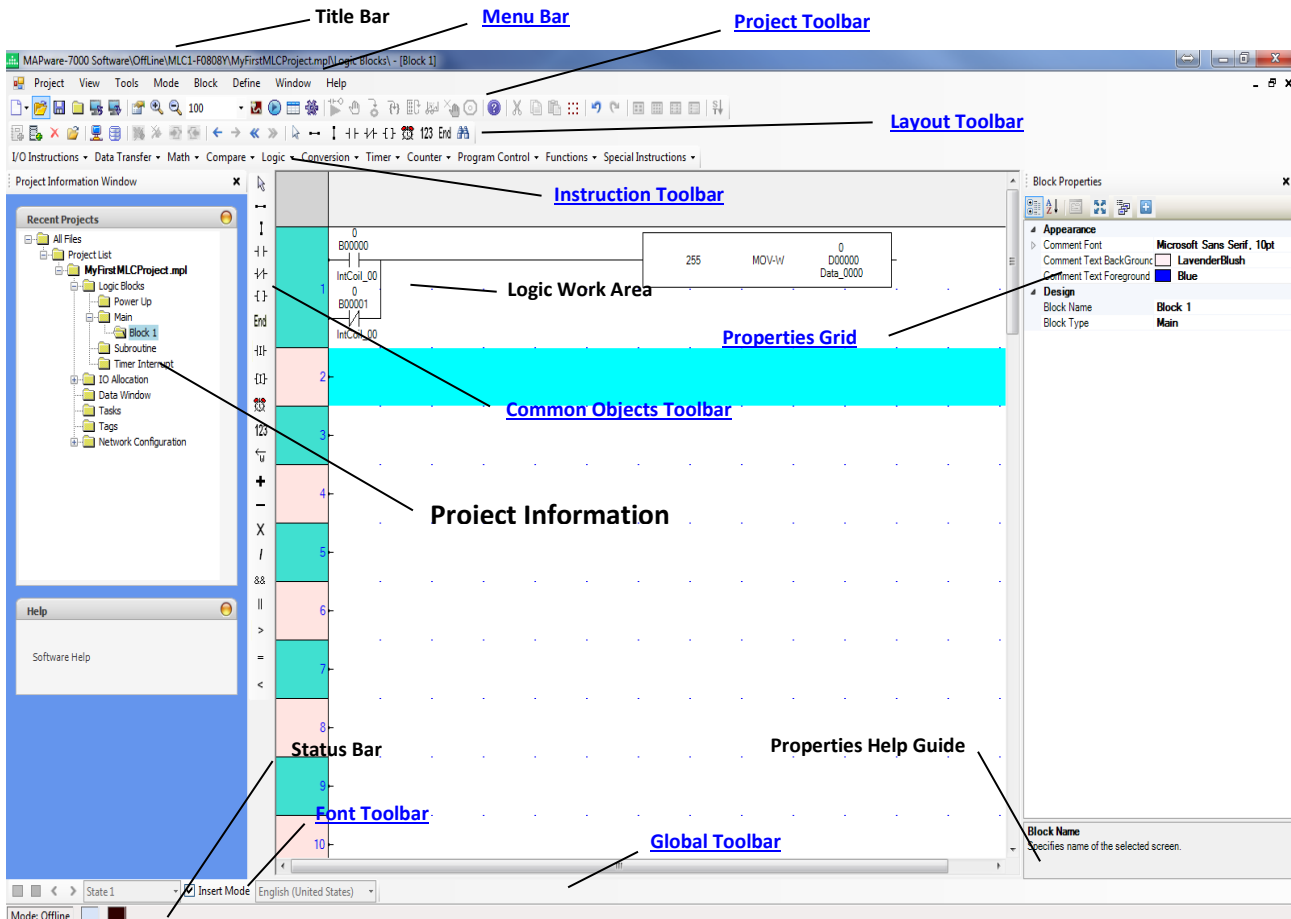
- Title Bar – Indicates the current open project and directory location.
- Menu Bar – These are the standard menu options for the MAPware-7000 software.
- Toolbars – Shortcuts of features and commands that you may use with regularity.
- Project Toolbar – Commands used on a project level (i.e. Save, Open, Cut, etc.).
- Layout Toolbar – Commands used when working with MLCs (i.e. New block, Tag Database, etc.).
- Instruction Toolbar – Access to all instructions that can be placed in a ladder logic block.
- Common Objects Toolbar – Access to most commonly used objects.
- Font Toolbar – Controls for text objects.
- Global Toolbar – Controls to view the various states of a highlighted object.
- Status Bar – Provides information about current project (i.e. offline, X Y coordinates).
- Project Information Window – Folder-style depiction of the current project.
- Screen Work Area – Shows all objects and attributes of current open screen.
- Properties Grid – This is the area where attributes of the highlighted object are adjusted.
- Properties Help Guide – Quick help on each attribute highlighted for an object.

Menu Bar Options

⌨ **Note:** Some of the menus listed below are only available when an HMC model is selected.

Project menu
Settings/controls used on the entire project

- New – Creates a new project
- Open – Opens an existing project
- Close – Closes the project that is currently open
- Save – Saves the currently opened project
- Save As – To save the currently opened project under a new file name
- Properties – This displays the Project Configuration dialog box
- Import Text Objects – Import a CSV file that contains text for all labels in the project
- Export Text Objects – Exports a CSV formatted file containing the text used in all labels of the project
- Transfer
  - o Download – Sends the currently opened project to a MLC Series
  - o Upload – Receives a project from a MLC Series
- Print – Prints a hardcopy of the opened project
- Compile – Compiles the opened project and checks for errors
- Run – Runs the opened project in a simulation mode
- Exit – Exits the MAPware-7000 application

View menu
Provides options for viewing screens and other sections of the opened project

- Data Monitor window – Opens the data monitor window (used to monitor the current values in tags during simulation)
- Device Information – Provides information about the MLC Series that is currently connected to your computer
- Zoom to – Enlarge or reduce the size of screen currently displayed in the workspace
- Screen Information – Opens a window at the bottom of the workspace that shows detailed information about the currently displayed screen
- Errors and Warnings – Opens a window at the bottom of the workspace that shows detailed information about any errors or warnings that may occur when the project compiles. This window helps you to locate potential problems with your project.
- Refresh – Repaints the workspace area [SDS]
- Screen/Object Properties – Displays or closes the screen/object properties box that is displayed on the right hand side of the workspace area
- Project Information – Displays or closes the **Project Information Window** located on the left hand side of the workspace area

Layout menu

These are tools used to edit simple shapes or to quickly adjust the size of objects on a web screen.

- Pattern – Displays the Pattern Dialog box for any simple shape that has the ability to set a pattern
- Line Color – Displays the Color Palette dialog box so that you can select a color for the line
- Fill Color – Displays the Color Palette dialog box so that you can change color of the area inside one of the simple shapes (i.e. rectangle, ellipse, etc).
- Text Color – Allows you to quickly change the color of the text in a text or multilingual text object
- Pattern Color – Use this to change the color of the pattern selected
- Bring to Front – If two or more objects are overlaid, this causes the selected object to be on top
- Send to Back – If two or more objects are overlaid, this will place the selected object on the bottom of the stack
- Align to – When two or more objects are highlighted, this will align the objects according to the left, right, top, or bottom sides of the objects
- Center Along – When two or more objects are highlighted, this feature moves the objects to the same location on the screen
  - Screen's Horizontal – Aligns along the horizontal center of the screen
  - Screen's Vertical – Aligns along the vertical center of the screen
  - Object's Horizontal – Aligns with respect to the horizontal center of the two objects
  - Object's Vertical – Aligns with respect to the vertical center of the two objects
- Place Equidistant – Takes three or more highlighted objects and places them equidistant from each other along
  - Along Horizontal – The horizontal plane
  - Along Vertical – The vertical plane
- Make Equal – Takes two or more highlighted objects and makes them all the same size.
  - **Note:** The height/width of the last selected object is used for the adjustment.
  - Height – Affects only the height of each highlighted object
  - Width – Affects only the width of each highlighted object
  - Size – Both width and height are adjusted

Screen menu

Options for creating/deleting screens in the project.
  - **Note:** This menu only appears if a current web screen is shown.

- New Base Screen – Creates a new base (full-sized) screen
- New Popup Screen – Creates a new popup (partially sized) screen
- New Template Screen – Create a new template screen
- Open Screen – Displays the Open Screen dialog box from which you can select which screen you would like displayed
- Open Template Screen – Displays the Open Template Screen dialog box from which you can select which template screen you would like displayed
- Import Screen – Import a screen from an existing project with the same screen size

- Apply Template Screens – Displays the Template Screen Setting dialog box which is used to assign any of the template screens to each base screen
- Close – Closes the displayed screen
- Delete – Deletes the displayed screen
- Duplicate –Duplicates the currently displayed base screen or template
- First – Displays the first screen (Screen 1) or first template (Template 1)
- Next – Displays the next screen in sequence
- Previous – Displays the prior screen in sequence
- Last – Displays the last (highest numbered) screen

Define menu

Use this menu to configure many of the background tasks of the MLC Series

- Data Logger – Displays the Data Logger screen. The Data Logger captures values from predefined tags. Capturing is based upon a time interval or a triggered event. Capture values are represented on the MLC Series screen using the Historical Trend Object or values are uploaded to a computer.
  - **Note:** Same as clicking the Data Logger folder in the **Project Information Window**.
- Global Keys – Displays the Global Keys configuration screen. Global Keys determine how the function keys on the right side of the display function when pressed. These keys can also be configured locally per screen using the Keys folder under each Screen folder in the **Project Information Window**.
  - **Note:** Same as clicking the Keys folder in the **Project Information Window**.
- Nodes – Use to assign a network node address to each serial com port on the MLC Series. You can also modify the settings for each serial port (i.e. baud rate, communications driver, etc.) as well.
  - **Note:** Same as clicking the **Network Configuration** folder in the **Project Information Window**.
- Tag Database – Displays the **Tag Database**. Use this to configure and assign tags to each internal memory address or PLC memory address you wish to access.
  - **Note:** Same as clicking the Tags folder in the **Project Information Window**.
- Alarm Database – Displays the Alarm Configuration screen. Use this screen to assign and create alarm messages.
  - **Note:** Same as clicking Alarms folder in the **Project Information Window**.
- Languages – Displays the Languages Configuration screen. Use this screen to select additional languages (up to 9) in text and labels for objects.
  - **Note:** Same as clicking Languages in the **Project Information Window**.
- System Parameters – Displays the Systems Parameters dialog box to configure the Timer Interrupt and Keep Memory Area.
- Ethernet Settings – Displays the Download Ethernet Settings dialog box for the MLC Series.
  - **Note:** Only available for models that have an Ethernet port.

Mode menu

Determines the operating mode of the MAPware-7000 software

- Edit – When selected, allows editing of the opened project (default mode)

- Online/Offline – When selected, the MAPware-7000 software connects to the MLC Series to read/write to internal memory addresses of the MLC Series. Data is edited and monitored using the Data Monitor window.
  - With upload – Current project residing in MLC Series uploads into the MAPware-7000 before going online.
  - Without upload – MAPware-7000 simply connects to MLC Series without uploading the existing project that is in the MLC.
- PLC Control – Controls how the ladder logic blocks in the MLC Series operate while in Online mode.
  - Run – Starts execution of ladder logic in the MLC
  - Halt – Temporarily stops execution of ladder logic
  - Run Force – Very similar to Run mode except in this mode, the ladder logic continues to run even if the physical expansion I/O modules are not connected to the MLC Series. The registers assigned to the I/O modules are updated. This mode is useful to check operation of the program.
  - Hold – Stops execution of the ladder logic but continues to update all inputs and outputs. This mode is primarily useful to troubleshoot external I/O signals connected to the expansion modules.
  - Hold Cancel – Cancels the Hold mode and returns to normal (Run) operation.
  - Error Reset – The MLC Series has a self-diagnostics routine that continuously runs in the background during normal operation. If a problem is detected during normal operation, the MLC Series may go into shutdown mode. In this mode, execution of the ladder logic stops and all outputs are assigned to the Error State Output Condition. Once the cause of the error is corrected, normal operation resumes if this command is sent to the MLC Series.

Tools menu
These miscellaneous settings and tools are available with the MAPware-7000 software.

- Application Memory Status – Display the Memory Status dialog box. This provides detailed information on memory required by the MLC Series for the currently opened project.
- Display Logged Data – Displays logged data (in CSV format) that has been uploaded from the MLC Series. Logged data can be captured by the MLC Series using the Data Logger feature.
- Display Historical Alarm Data – Displays data collected by the MLC Series on alarms
- Convert Application – Allow programmer to convert a project using one MLC Series to a different model of the MLC Series
- Preferences – Miscellaneous settings when using MAPware-7000
  - Colors – Displays the Color Settings dialog box. These are color options when using the Logic Block feature
  - On line communication – Determines which port on the MLC Series is used during an online connection – USB, serial, or Ethernet
  - Others – Set the following:
    - Ladder instruction tag length – Determines maximum number of characters shown for each tag address in the ladder logic

- ▪ Show default tag name definition screen – If enabled, a popup dialog box will appear when entering instructions in a logic block It gives programmer ability to change the name of the tag easily.
- ▪ Show overlapping message screen – During compile, if MAPware-7000 detects that two adjacent objects are overlapping, a message is displayed

Data Window menu

This menu is for the **Data Window** feature.

- New block – Displays the Define Block dialog box. Use to define a block of data viewed in the Data Monitor.
- Delete block – Displays the Delete Block dialog box. Select which block of data you wish to remove from the Data Monitor.
- Edit block – Displays the Edit Block dialog box. Use to modify an existing block (i.e. change address, size, color, etc.).
- Import Data – Import a Data Monitor file (in *.CSV format) that was created in a different project. Useful if you wish to monitor the same addresses
- Export Data – Export a Data Monitor file (in *.CSV format) so that you do not have to recreate the Data Monitor list in another project

Window menu

MAPware-7000 provides many options for displaying open screens

- Cascade – Cascades (overlapping windows) all open screens
- Tile Horizontally – Displays all open screens horizontally (top to bottom of monitor)
- Tile Vertically – Displays all open screens vertically (left to right of monitor)
- Close All – Closes all open screens
- List of Open Screens – Displays a popup dialog box of all open screens

Help menu

MAPware-7000 includes extensive help files that cover the operation of each feature.

- Contents – Displays the Help files, Contents tab
- Index – Displays the Help files, Index tab
- Search – Display the Help files, Search tab
- About – Provides information on Software revision

Draw menu

Use to place objects onto each screen. Also configures the grid function for screens. A limited amount of these functions are available on web screens when using an MLC3-E.

- Graphical Objects – Objects used to create simple shapes and text
    - o Line – Draws a line on screen
    - o Ellipse – Draws an ellipse or circle
    - o Rectangle – Draws a rectangle or square

- o Rounded Rectangle – Draws a rectangle or square with beveled edges
- o Multilingual Text – Use to create a static text box. This object differs from the simple text box in that this object allows use of Windows TrueType fonts.
- Buttons – Objects that require input from the HMC Series touchscreen
  - o Navigation Buttons – Buttons used to change or navigate through screens
    - Go to Screen – Creates a button that, when pressed, displays a base screen
    - Go to Next Screen – When pressed, goes to next screen in numerical sequence
    - Go to Previous Screen – When pressed, goes to prior screen in numerical sequence
    - Open Popup Screen – Creates a button that, when pressed, displays a popup (partially sized) screen which overlays on top of the base screen already showing
  - o Bit Action Buttons – Buttons used to control a coil or bit
    - Set Bit – Creates a button that, when pressed sets (turns On) a bit
    - Reset Bit – When pressed, clears (turns Off) a bit
    - Toggle Bit – Alternates setting/clearing a bit each time button is pressed
  - o Word Action Buttons – Buttons that write values to registers. Data can be 8-bit, 16-bit, or 32-bit formats depending upon how the registers are defined in the **Tag Database**.
    - Write Value to Tag – Writes a predefined constant value to the register
    - Add Value to Tag – Adds a predefined constant value (addend) to the number already existing in the target register and places the resulting value (sum) in that register
    - Subtract Value from Tag – Subtracts a predefined constant value (subtrahend) from the existing number (minuend) in the target register and places the resulting value (difference) in that register
    - Add Tag B to Tag A – When pressed, this button will
      1. Read the current value in Tag B
      2. Read the current value in Tag A
      3. Add the Tag B value to the Tag A value
      4. Write the resulting sum to Tag A
    - Subtract Tag B from Tag A – When pressed, this button will
      1. Read the current value in Tag B
      2. Read the current value in Tag A
      3. Subtract the Tag B value from the Tag A value
      4. Write the resulting difference to Tag A
    - Alarm Action Buttons – Buttons used to control active alarms
    - Alarm Acknowledge – Creates a button that, when pressed 'acknowledges' or indicates that an active alarm has been recognized by the plant floor operator
    - Acknowledge All – When pressed, this button will acknowledge all active alarms
    - Next Alarm – When pressed, displays the next active alarm (according to chronological order) in sequence
    - Previous Alarm – Displays the prior active alarm (according to chronological order)
  - o Recipe – These are buttons that control recipes
    - Transfer Recipe – When pressed, this button copies a block of internal memory registers in the HMC Series to a set of PLC registers or internal memory registers

- ▪ Upload Recipe – When pressed, this button copies a predefined block of registers in the PLC to a set of internal memory registers in the HMC Series
  - o Multi-task Single-state Button – This button is very similar to the Bit Action Buttons described above except that instead of controlling a single bit/coil, the Multi-task Single-state button performs one or more tasks. This button performs the same set of tasks every time the button is pressed and/or released.
  - o Multi-task Multi-state Button – This button is similar to the Word Action buttons above, except that instead of writing to a single register, the Multi-task Multi-state button performs one or more tasks. In addition, unlike the Multi-task Single-state Button, this button can be configured for several 'states', meaning that every time the button is pressed, a different set of tasks can be initiated.
- Input Objects – Keypads and data entry objects
  - o Data Entry – Objects that display a keypad to change the value of a bit or register
    - ▪ Bit Entry – This object is very similar to a Toggle Bit object but instead of writing immediately to the targeted bit whenever the button is pressed, this object displays a popup keypad. The popup keypad allows you to select a '1' or '0', then press the Enter key to write the new value to the bit.
    - ▪ Numeric Entry – This object displays the current value in a register. When pressed, a popup keypad appears that allows the operator to write a new value to the register.
  - o Keypad – Displays a ten-key keypad on the screen
  - o ASCII Keypad – Displays an alphanumeric keypad on the screen
  - o Custom Keypad – Displays a keypad on the screen but without any defined keys. Programmer defines each key as to operation.
- Display Objects – These are objects that display data stored in a bit or register as a number or as some graphic representation (i.e. lamp, bar graph, etc.).
  - o Text – This simple text object displays static text using one of four predefined font sizes.
  - o Date – Displays the date using the internal clock
  - o Time – Display time
  - o Data Display – These objects display the data in bits or registers as numbers or characters.
    - ▪ Numeric Display – Displays the current value in the register
    - ▪ Message Display – Displays predefined text strings based upon the value in the register. Each text string represents a predefined range of values.
    - ▪ Bit Display – Displays two predefined text strings according to the state (On or Off) of the bit.
  - o Bit Lamp – Represents the state of a targeted bit using a predefined shape or picture (BMP, JPG or GIF)
  - o Word Lamp – Same as Bit Lamp except a Word Lamp represents registers.
  - o Bar Graph – Displays a single bar graph that represents the current value in a register.
  - o Multiple Bar Graphs – Displays up to four separate bar graphs
  - o Real Time Trend – Displays a trend graph that represents up to four registers. The graph polls the registers on a timed basis and displays the values as plots on a time graph.

- o   Historical Trend – Same as above except instead of real-time, the values are read from the Data Logger.
- o   Analog Meter – Represents a data register as an analog meter.
- o   Alarm List – Displays a list of alarm messages
- Grid – Turn on or off a grid displayed on the screens. Use the grid to help align objects.
- Snap to Grid – If enabled, objects will align on screen according to the location of the grid points.
- Configure Grid – Displays the Configure Grid dialog box, which allows the programmer to set the size (in pixels) of the grid boxes

Block menu

This menu is for configuring the ladder logic in a Logic Block.

> 🖅 **Note:** This menu is hidden unless you click a Block in one of the Logic Block subfolders in the **Project Information Window**.

- New – Creates a new ladder logic block in one of the six types of logic block menus:
  - o   Power Up – This is a logic block that executes once when power is initially applied to the MLC Series
  - o   Main – Logic blocks created under the Main menu are executed continuously during normal operation
  - o   Subroutine – These logic blocks are only executed when called from another logic block
  - o   Timer Interrupt – Logic blocks created under the timer interrupt folder are given the highest priority of all logic blocks. They are executed continuously according to the time interval programmed.
  - o   IO Interrupt 1 – This logic block is executed immediately when called (unless a timer interrupt is in process) and is called by the high speed counter function, when the count value reaches the preset value set.
  - o   IO Interrupt 2 – Similar to IO Interrupt 1 except that, when called, it takes lower priority than the timer interrupt and IO Interrupt 1
- Duplicate – Copies the highlighted logic block and creates a new one identical to it.
- Open – Displays the Open Block dialog box with a list of all logic blocks. Highlight a logic block to open and display it.
- Import – Bring a logic block from another project into the existing project. Click this option to display the Open Project dialog box. Select the project that has the logic block you wish to import. An Import Block dialog box displays. Highlight all logic blocks for import, and then click OK.
  🖅 **Note:** In Native Ladder programming mode the blocks are imported directly from an existing project. In IEC 61131-3 programming mode, logic blocks are exported (see below) to .xml files which can then be imported using this menu option.
- Export – (IEC 61131-3 programming mode only) Export a logic block as an .xml file for use in another project.
- Delete – Removes the currently viewed logic block from the project
- Close – Removes the currently viewed logic block from the workspace

The Toolbars

There are several toolbars available in MAPware-7000 to make it easier to access objects and features that you need. Below is a list of the toolbars and a brief description of each shortcut icon in the toolbar (for a more detailed description of the feature represented by the icon, check the Standard menu above or elsewhere in this manual).

⎙ **Note:** Some of the toolbars and options listed below appear only for certain models (HMC or MLC).

Project Toolbar
Provides shortcuts to common project tasks

Icon	Name	Description
	New Project	Create a new project
	Open Project	Opens an existing project
	Save Project	Saves the current project to the hard drive
	Close Project	Closes current project without exiting MAPware-7000 software
	Upload Project	Uploads a project from a MLC Series
	Download Project	Downloads a project to a MLC Series
	Screen/Object Properties	Opens/closes the Screen/Objects Properties box
	Zoom In Screen	Magnifies size of screens shown in the workspace
	Zoom Out Screen	Reduces size of screens shown in the workspace
	Zoom Factor	Adjust the magnification size of the zoom
	Go Online	This button causes MAPware-7000 to go online with the MLC without uploading the application.
	Run	Simulate the project on your computer monitor
	Data Window	Displays the data window; used to monitor values
	Compile	Compiles the project into a format that can be interpreted by the MLC Series and checks for errors
	Help	Displays the Help files
	Cut	Use to delete objects from displayed screen
	Copy	Copies the selected object(s) for pasting
	Paste	Pastes a copied object(s) onto the screen
	Select All	Highlights all objects currently on displayed screen
	Previous Object	Moves focus cursor (highlight) to prior object on screen
	Next Object	Moves focus cursor to next object on screen
	Undo	Undoes last executed action in MAPware-7000

Icon	Name	Description
	Redo	If the Undo action was performed by mistake, Redo will bring back the recorded action last performed in MAPware-7000
	Large Icons	Shows open screens in workspace as large icons
	Small Icons	Shows open screens in workspace as small icons
	List	Shows open screens in workspace as a brief list
	Details	Shows open screens in workspace as detailed list
	Hide System Tags	Toggles display of the system tags listed in the **Tag Database**

Layout Toolbar

Provides shortcuts for screen-related tasks

Icon	Name	Description
	New Base Screen	Creates a new base screen
	New Template Screen	Creates a new template screen
	New Popup Screen	Creates a new popup screen
	Delete Screen	Deletes the current screen shown in the workspace
	Open Screen	A popup dialog box lists all screens, click one to open
	Show Screen	Similar to Open Screen but you can only use this action if the highlight cursor is on Screens in the Project Information Window
	Nodes	Displays a list of all connections (serial, Ethernet) assigned. **Note:** Same as clicking Network Configuration folder in the **Project Information Window**
	Tag Database	Displays the tag list (including predefined tags). **Note:** Same as clicking Tags folder in the **Project Information Window**
	Alarm Database	Displays the alarm database. **Note:** Same as clicking Alarms folder in the **Project Information Window**
	Add Bookmark	Places a bookmark on the screen in the workspace
	Clear Bookmark	Removes a bookmark assigned to the screen in the workspace
	Clear All Bookmarks	Removes all bookmarks from screens
	Next Bookmark	Displays the next bookmarked screen (in numerical order)
	Previous Bookmark	Displays the prior bookmarked screen (in numerical order)
	Previous	Displays the prior screen on the screen list (in numerical order)
	Next	Displays the next screen on the screen list (in numerical order)
	First	Displays Screen 1
	Last	Displays the last configured screen (typically Screen 65004 ASCII Keypad)

Icon	Name	Description
	Bring to Front	If two or more objects are overlaid on screen, the selected object is brought to the forefront
	Send to Back	If two or more objects are overlaid on screen, the selected object is sent to the background
	Align Left	If two or more objects are highlighted, this action aligns the objects along the left side of the leftmost highlighted object on screen
	Align Right	If two or more objects are highlighted, this action aligns the objects along the right side of the rightmost highlighted object on screen
	Align Top	If two or more objects are highlighted, this action aligns the objects along the top side of the topmost highlighted object on screen
	Align Bottom	If two or more objects are highlighted, this action aligns the objects along the bottom side of the lowest highlighted object on screen
	Center Along Screen's Horizontal	If one or more objects are highlighted, this action aligns the object(s) along the horizontal center of the screen.
	Center Along Screen's Vertical	If one or more objects are highlighted, this action aligns the object(s) along the vertical center of the screen.
	Center Along Object's Horizontal	If two or more objects are highlighted, this action aligns the objects along the calculated midpoint of the total vertical distance between the objects.
	Center Along Object's Vertical	If two or more objects are highlighted, this action aligns the objects along the calculated midpoint of the total horizontal distance between the objects.
	Place Equidistant Along Horizontal	If three or more objects are highlighted, this action places the objects equidistant from each other along the horizontal axis.
	Place Equidistant Along Vertical	If three or more objects are highlighted, this action places the objects equidistant from each other along the vertical axis.
	Make Equal Width	If two or more objects are highlighted, this action resizes the width of each object to the same width of the last selected object.
	Make Equal Height	If two or more objects are highlighted, this action resizes the height of each object to the same height of the last selected object.
	Make Equal Size	If two or more objects are highlighted, this action resizes each object to the same size of the last selected object.
	Group Objects	If two or more objects are highlighted, this action combines the objects into a 'group'. Once this is done, you can then move or resize the grouped object rather than perform these actions separately on each member of the group.
	Ungroup Objects	If a grouped object is highlighted, this action separates each object into the group so that you can now resize or move each object separately.
	Search Instruction	Used only in the **Logic Blocks**, this allows you to quickly locate a specific instruction of tag address that is configured in one of the logic blocks.

Draw Toolbar

Provides shortcuts to all objects that can be placed on screens

Refer to the *MAPWare-7000 Programming Manual,* for a list of objects available from this toolbar, and for detailed instructions on using each object.

Common Objects Toolbar

This vertical toolbar is directly to the left of the screen workspace. It contains commonly used objects that are placed onto screens.

Refer to the *MAPWare-7000 Programming Manual*, for a list of objects on this toolbar and detailed instructions on using each of the objects.

Font Toolbar

This toolbar is located along the bottom of the MAPware-7000 software to adjust or modify text object boxes and simple drawn objects such as lines, rectangles and circles.

Icon	Name	Description
Arial	Font Name	Change the font used in a text object. On other objects, this box changes to a font size box for simple fonts.
10	Font Size	Change the size of the font used
B	Bold	Use to bold characters
I	Italics	Use to italicize characters
U	Underline	Use to place an underline on characters
	Line Color	Change the color of the perimeter line of a simple object (i.e. line, rectangle, ellipse) or the border of the Multilingual Text object
	Fill Color	Change the color of the interior of a simple object (i.e. line, rectangle, ellipse) or the interior of data objects (i.e. Multilingual Text, Simple Text, Bit Entry, Message Display, Numeric Display, etc)
A	Font Color	Change the color of the text characters of data objects (i.e. Multilingual Text, Simple Text, Bit Entry, Message Display, Numeric Display, etc)
	Pattern	Use to create a pattern of two colors for the fill area of a rectangle
	Pattern Color	Use to select the secondary color for the pattern (primary color based on Fill Color)

Global Toolbar

This toolbar is located along the bottom of the MAPware-7000 software and is used to view the different states of a highlight object.

Icon	Name	Description
	On State	Displays the 'on' state of all two state buttons shown on screen.
	Off State	Displays the 'off' state of all two state buttons shown on screen.
	Previous State	Displays the 'previous' state on multi-state objects shown on screen.
	Next State	Displays the 'next' state on multi-state objects shown on screen.
State 1	State	Use the pull-down list box to jump directly to a particular state on a multi-state object.
English (United States)	Language	When using the Multi-Language feature, this allows you to see all of the text in each language state.

Status Toolbar

Provides general information about the project currently loaded.

Icon	Name	Description
Mode: Offline	Mode	Shows what operational mode the project is currently in: Offline, Online, Run, Halt, etc.
Scratchpad area	Scratchpad area	This simply indicates that the area of the workspace that is not the screen is light blue
Screen area	Screen Area	This indicates the background color of the screen that is currently being modified
Text Wizard selected.	Object selected	Indicates which object on screen currently has the focus.
Top Left X : 32 Top Left Y : 6	Object Position	Indicates the top left corner coordinates of the object selected.
Width : 67 Height : 16	Object Size	Indicates the size of the object selected in pixels.

Project Information Window

This window displays a listing of all projects created in the past and opened with the MAPware-7000 software. It also displays a detailed feature list of the currently opened project in folder format. Use this window to easily maneuver and display various aspects of the opened project.

Properties Grid

The Properties Grid is a dialog box that shows the attributes for the screen/object currently highlighted. To make any changes to an attribute, simply click the attribute and edit the setting. More information about objects and the attributes available for each object are found in *Chapter 6 – Logic Blocks*. On the bottom of the Properties Grid is a local help guide. This is a brief description of the attribute highlighted.

Starting a New Project

To start a new project in MAPware-7000, first start the application by clicking the MAPware-7000 desktop icon

or the icon located in the Maple Systems folder:
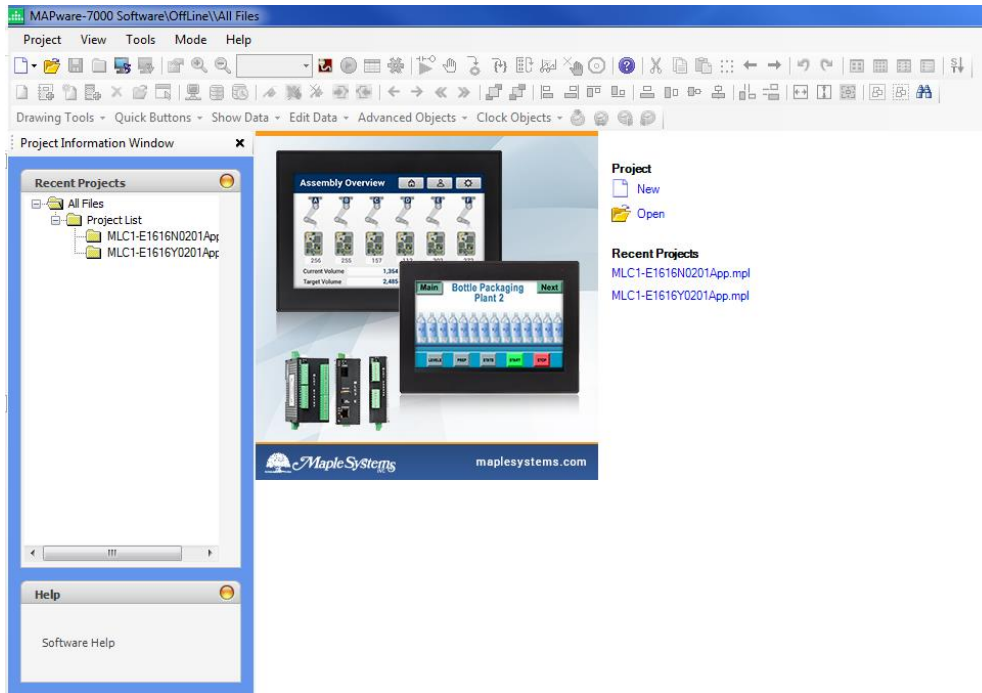
In the Project menu, click **New**, or simply click **Project > New** in the main work area.

Select a Product

When beginning a new project, you must first select which particular HMC or MLC model that you will be programming in the **Select Product** window:
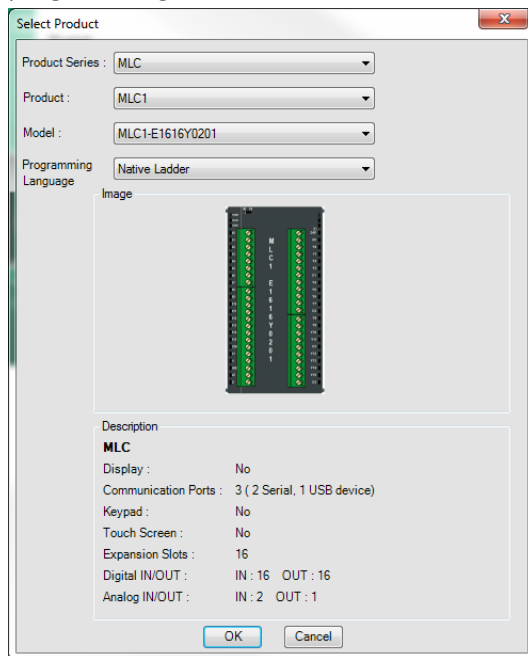
- **Product Series:** Select either a HMC (Human Machine Controller) HMI or a MLC (Maple Logic Controller) PLC.
- **Product:** Select the product family
- **Model:** Select the particular model.
- **Programming Language:** Select from Native Ladder or IEC 61131-3.
- **Image:** An image of the selected product.
- **Description:** A brief description of the selected product and its hardware features.

Click **OK** to start your project:
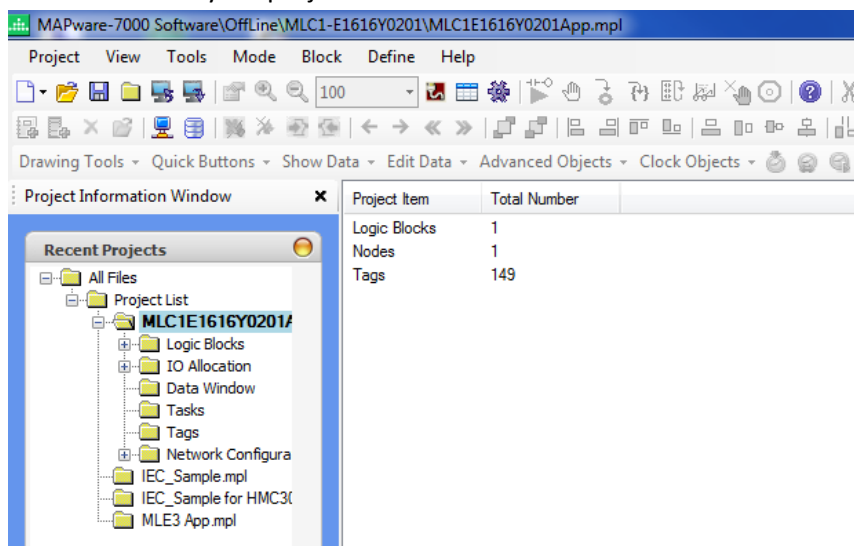
In the **Project Information Window**, you will see a **Project List**. The project folder titled *MLC1E1616Y0201App.mpl* is the project that we just opened in this example. Below that folder is a list of all other MAPware-7000 projects that have been recently opened in the software.

Underneath the *MLC1E1616Y0201App.mpl* folder is a collection of sub-folders which will be used to further configure the unit.

Configure the I/O modules

If you plan to use any I/O expansion modules with your MLC CPU module, you must assign them to slots in the MAPware-7000 software and then configure them. To do this, click the **IO Allocation** sub-folder:
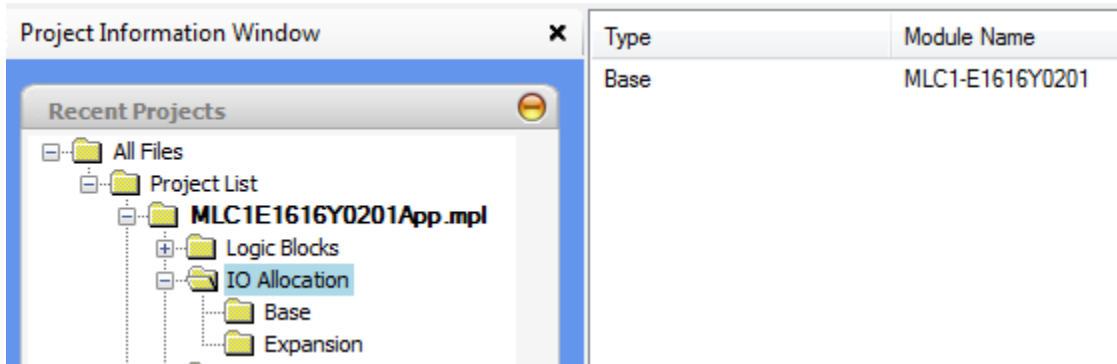
The **Base** folder shows the CPU module that was selected. The **Expansion** folder provides a listing of all I/O expansion modules added to the CPU module.

To configure the I/O located on the CPU base module or to add an I/O expansion module, click the **Base** folder or the **Expansion** folder, then double-click the CPU/Slot name to display the **IO Allocation** window.

     Note: You can also right-click to display the **Edit** option.
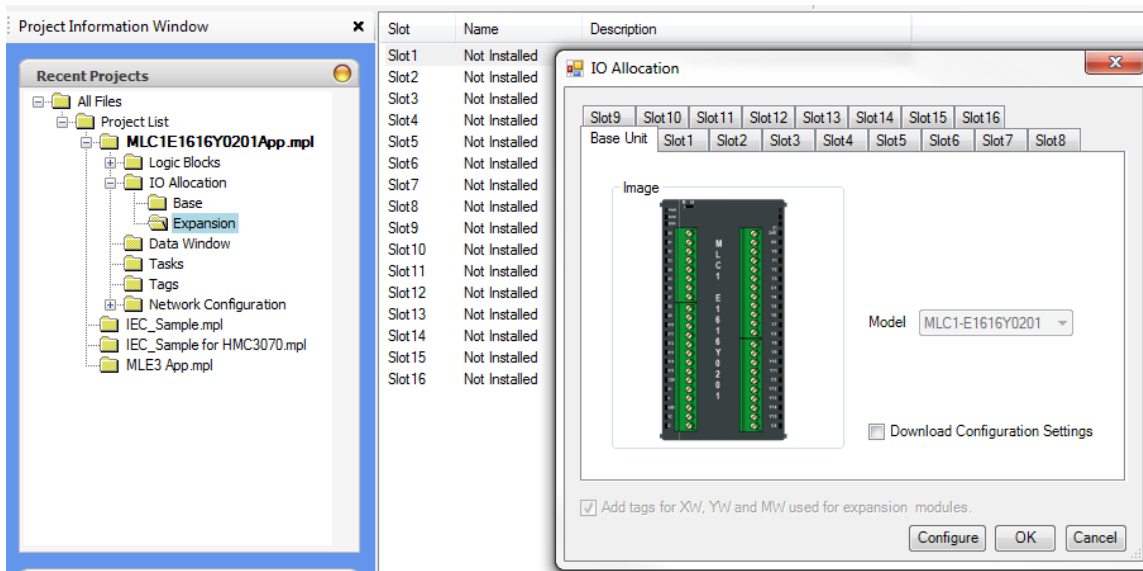
Click the **Base Unit** tab to configure the Inputs/Outputs built-in to the CPU module. To configure a High Speed Counter (HSC), Pulse Width Modulated output (PWM), or the analog inputs/outputs, click the **Configure** button:

After changing the settings for each digital/analog channel, click **Confirm** to save. Click the **Close** button when done.

Check the **Download Configuration Settings** option to store these settings in the MLC during a download of the project.

Configuring any I/O expansion modules is very similar. Select the Slot that the I/O Expansion module will be attached, then select from the **Model** pull-down box the appropriate I/O module.
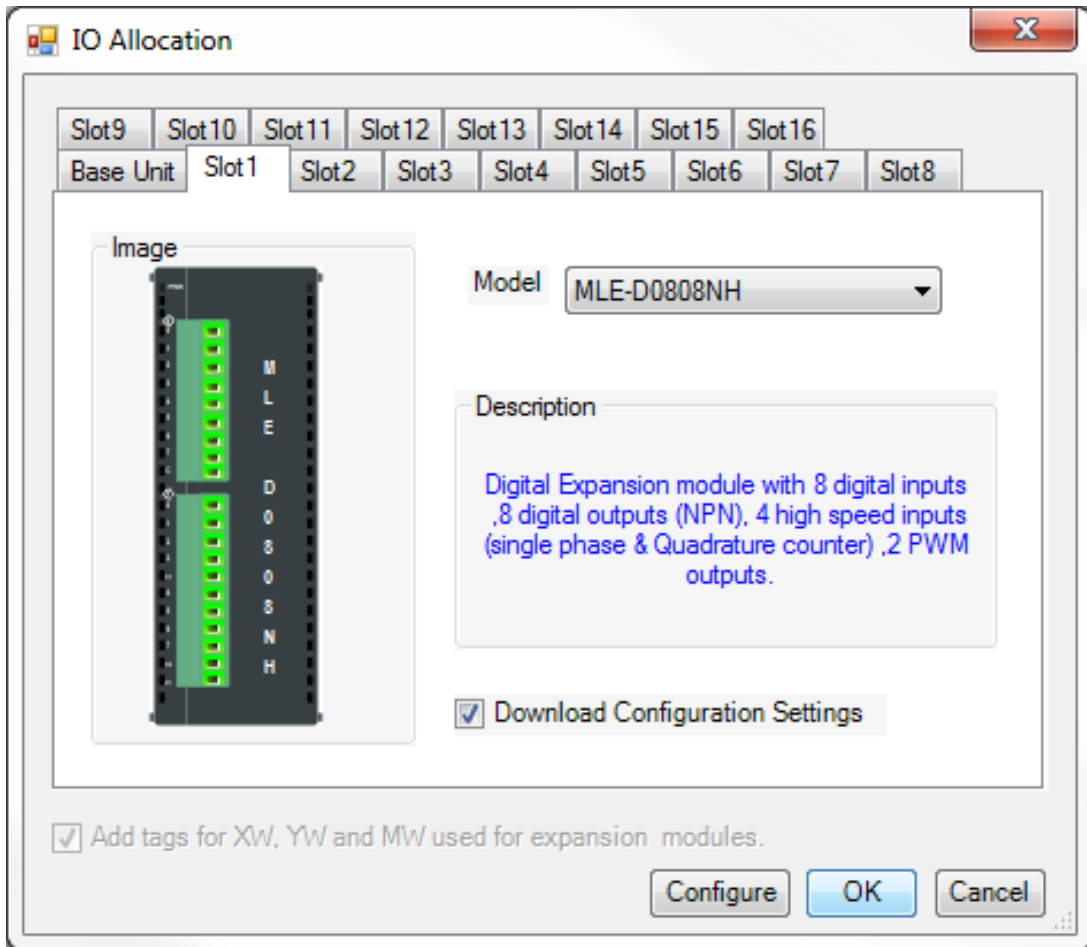
When done, you will see your I/O modules listed according to Slot location:

Configure the Communications (COM) Ports

The **Network Configuration** folder in the **Project Information Window** is used to configure the communications ports on the MLC unit. The communications ports can be used to connect to a separate HMI or PLC. Most of the MLC models support COM1 (RS232) and COM2 (RS485 2W), and a few models, like the MLC 2 and MLC3 also support Ethernet.

To configure one of the COM ports, highlight it, then right-click to display the context menu:
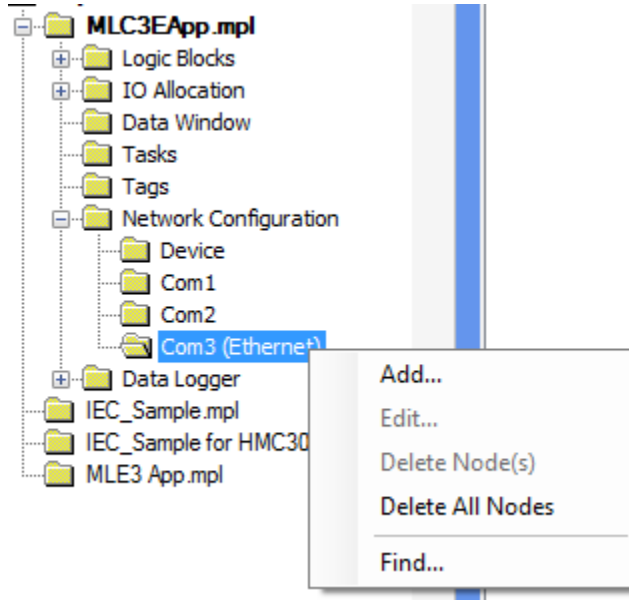
Click the **Add** option to display the **Node Information** window:

Node Information section:

- **Name** – This is the name that appears when assigning tags to the PLC memory in the **Tag Database**.
- **Address** – This is the network node address assigned to the serial/Ethernet port of the external HMI/PLC.

- **Interframe Delay** – Amount of time that the HMC/MLC Series waits before sending another command packet. Default is 10 msec.
- **Response Time Out** – Amount of time that the HMC/MLC Series waits for a response from the external PLC before it attempts to resend a command. Default is 800 msec.
- **Retry Count** – The total number of attempts that the HMC/MLC Series performs to try to reestablish a lost communication link to the PLC. If the retry count is exceeded, the HMC/MLC Series will signal a fault.
- **PLC Specific Settings** – These are settings that may be unique to a particular PLC protocol selected.

Select Protocol section:

- **Port** – Select Com1, Com2 or Ethernet.
  ⌨ **Note:** Options vary according to HMC/MLC Series model selected.
- **Protocol** – Select the PLC communications driver to assign to the serial/Ethernet port from the drop-down list.
- **Model** – Select the particular PLC model.
- **Communication Settings** – These are the serial communications settings (i.e. baud rate, parity, etc.) that must match the settings of the external PLC.
- **Power On PLC Communication Time** – This is the amount of time that passes after the HMC/MLC Series powers up before it attempts to communicate to the external PLC. Default is no delay.
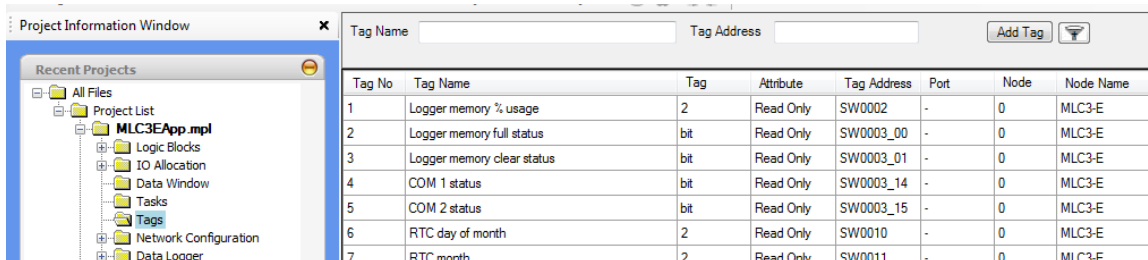
When done, click **Add** or **Apply** to record the settings.

More information about configuring the communications ports is available in the MAPware-7000 help files (search for Network Configuration).

Tag Database

The Tags folder is used to create the **Tag Database** for the project. Tags are created and linked to internal memory or external memory (i.e. HMI/PLC). A tag is a register, coil, or an individual bit of a register. Some tags are predefined or reserved for special purposes. The internal memory that a tag can be assigned to differ according to the programming language (native ladder or IEC 61131-3) selected.

The existing **Tag Database** for a project can be displayed by clicking the **Tags** subfolder of the currently open project.

To add a new tag, click the **Add Tag** button. For more information on using tags, see *Chapter 5 – Tags* or the MAPware-7000 help files.

Create PLC Ladder Logic

All ladder logic for the MLC is created in the **Logic Blocks** sub-folder shown in the **Project Information Window**:
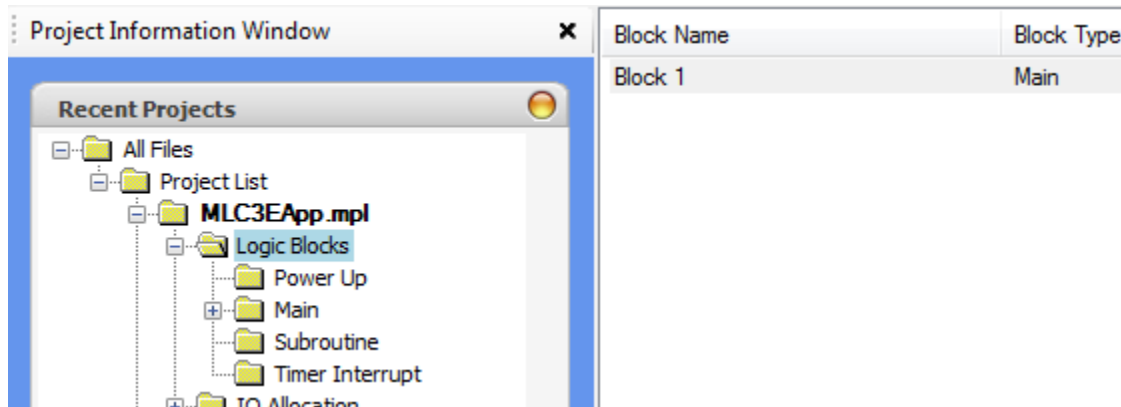
Consult Chapter 6 – Logic Blocks for more information on how to start programming the MLC.

Compiling the Project

After creating your ladder logic code, you must compile the project before downloading to the MLC. MAPware-7000 will check your project for any structural errors (logic errors cannot be determined by the compiler) and report them in the Errors and Warnings window.
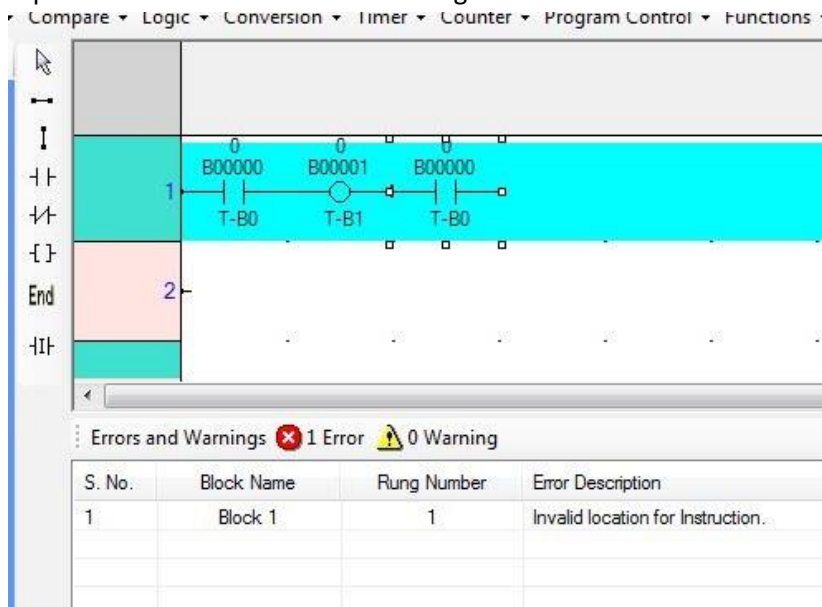
To compile the project, click the **Compile** icon or from the **Project** menu, click **Compile**.

> **Note:** You must compile the project before downloading it to the MLC.

Saving the Project

To save the opened project, select **Project > Save** or click the Save Application icon.

Save the project with a specific name and the .mpl extension.

> ⧉ **Note:** When saving a project, there are two parts – the *.mpl project file and a folder with the same name. Both parts are required when transferring the project to another computer.

Installing the USB Download Driver

Projects created in MAPware-7000 are downloaded into the MLC Series via the RS232 Com port on your computer, by connecting to a USB port or through Ethernet on models with an Ethernet port. If you intend to use the USB connection, you must first install a USB driver onto your computer. This allows MAPware-7000 to communicate with your MLC Series PLC via a USB connection.

1. To install the USB driver, connect an MLC Series PLC to your computer using a USB cable.
2. Apply power to the MLC. After a moment, your computer will detect the USB connection to the MLC and notify you that a new driver is required.

3. Check the 'No, not this time' option and then click **Next**.
4. Check 'Install from a list or specific location (Advanced)'.

5. Click **Next** to continue.

6. Check 'Search for the best driver in these locations' and check 'Include this location in the search'.
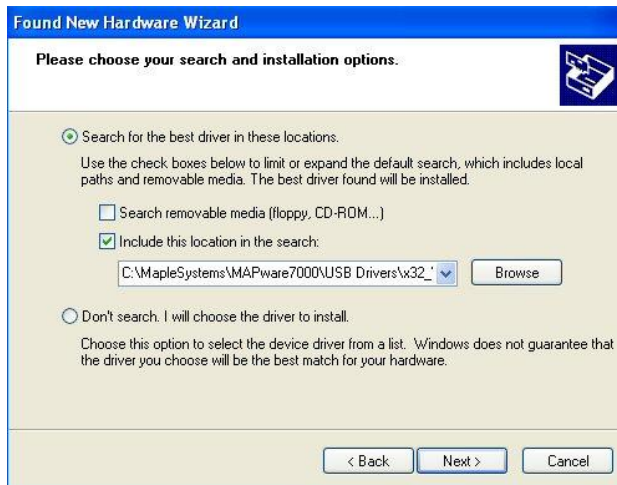
7. Click the **Browse** button.

8. Go to: '**C:\MapleSystems\MAPware7000\USB Drivers\ x32_Win7**'.

📑 **Note:** Select 'x64_Win77' if your computer has a 64-bit operating system.

9. Click **OK**. When you get back to the screen above, click **Next**.

10. If this dialog box appears, click **Continue Anyway**.

11. The driver will now install onto your computer. This takes a few moments but eventually the final screen below appears.

12. Click **Finish**.

You can now open the MAPware-7000 configuration software and upload/download projects to the MLC Series products using a USB connection.
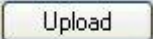
Downloading/Uploading Projects

Chapter 2 – A Quick Start gives you a step-by-step introduction into creating and downloading a new project. This section guides you through the specific steps and options of uploading/downloading a project into your MLC Series unit.

Uploading a project

1. Start the MAPware-7000 configuration software. It is not necessary to open a project.
2. Connect to the MLC Series using either a USB, serial RS232, or Ethernet cable, if using a model with an Ethernet port.
3. Apply power to the unit.
4. Select **Project > Transfer > Upload** from the standard menu.
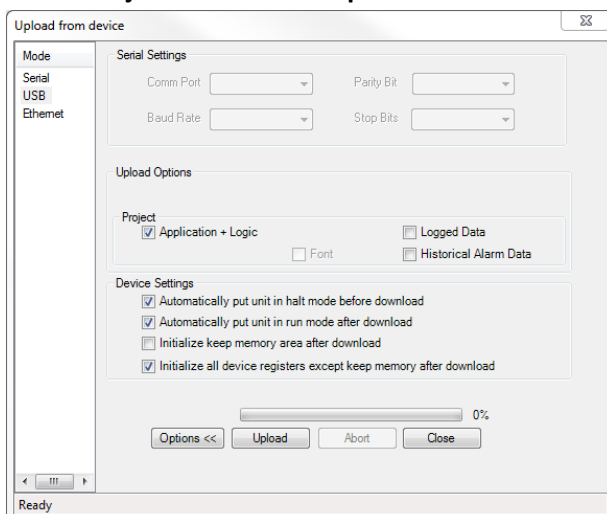
5. Under **Mode**, select Serial, USB, or Ethernet if available.
6. Under **Upload Options** (click the Options button [Options >>] if you do not see this section), select what data you wish to upload:
   a. **Application + Logic** – Uploads the application and the Native Ladder or IEC61131-3 program logic. The ladder logic is any ladder instructions created using the **Logic Blocks**.
   b. **Logged Data** – Uploads any data collected using the Data Logger feature (MLC2 and MLC3 models only, see *Chapter 8 – Miscellaneous Features* "Data Logger" section for more details). This option is not available when using Ethernet in Upload Mode.
   c. **Historical Alarm Data** – Uploads any alarm messages collected when using the Historical Alarm feature (not supported for the MLC Series). This option is not available using Ethernet.
7. Click the **Upload** [Upload] button:
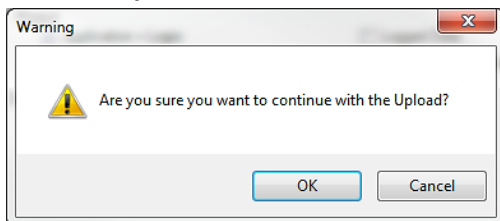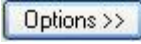
8. Click **OK**.
9. When the upload is finished, click **Close**. The project is automatically stored under the Projects folder in MAPware-7000 under the name Untitled1.mpl with a folder by the same name "Untitled1". To save the project under a different name, click **Project > Save As**.

Downloading a project

1. Start the MAPware-7000 configuration software and open the project you wish to download to the unit.
2. Connect to the MLC Series using either a USB, serial RS232, or Ethernet download cable.
3. Apply power to the unit.
4. Click **Project > Transfer > Download** from the standard menu or click [icon] from the Standard toolbar.
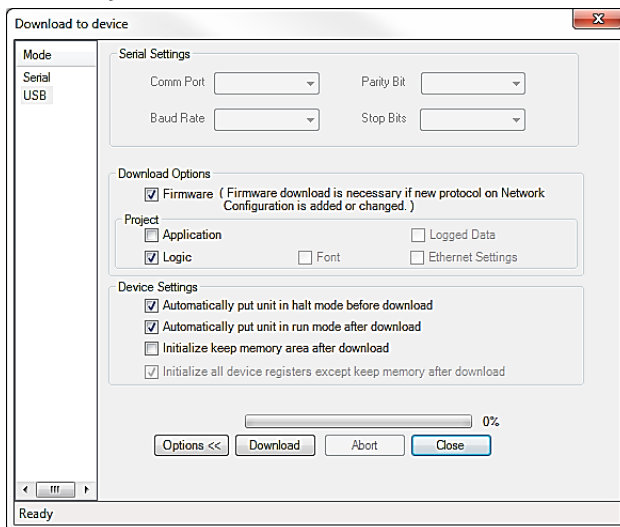
5. Under **Mode**, select Serial, USB, or Ethernet if available.

6. Under **Download Options** (click the Options button [Options >>] if you do not see this section), select what data you wish to download:
    a) **Firmware** – The internal program inside the MLC Series that makes it run. If you are downloading:
        i. To a new unit
        ii. Upgrading the firmware of the unit to a newer version
        iii. A project that was created in an older version of MAPware-7000
        iv. A project in which you have made any changes to the PLC communications drivers assigned to the Com ports on the MLC Series, then check this option. For subsequent downloads of the same project to a unit, it is not necessary to continue to download the firmware.
    b) **Application** – Downloads application only (this is the project that you create using MAPware-7000)
    c) **Ladder** – Downloads the ladder logic. The ladder logic is any ladder instructions created using the **Logic Blocks**.
    d) **Fonts** – Option not available for MLC Series.
    e) **Logged Data** – Option not available for MLC Series.
    f) **Ethernet Settings** – Downloads Ethernet settings configured in the Project Configuration window. Only available on models with an Ethernet port.
7. **Device Settings** options:
    a) **Automatically put unit in halt mode before download** – Check this option to stop operation of the project in the MLC Series while downloading a new project.
    b) **Automatically put unit in run mode after download** – Checking this option enables MAPware-7000 to command the MLC Series to begin running the project after the download is finished.
    c) **Initialize keep memory area before download** – If you are using the Keep Memory area (see *Chapter 8 – Miscellaneous Features*), checking this option will initialize all keep memory to 0.
    d) **Initialize all device registers except keep memory after download** – This option will initialize all internal memory of the MLC Series (except for the keep memory area as outlined in Chapter 8) to zero.

   ⌨ **Note:** Whenever power is cycled on the MLC, only the Keep Memory Area is retained. Other device registers are cleared (initialized to zero). The Download options listed above only apply when downloading a program.

8. Click the **Download** [Download] button:

   ⌨ **Note:** If you check the 'Do not show' option, the next time you download, this dialog box will not be shown.
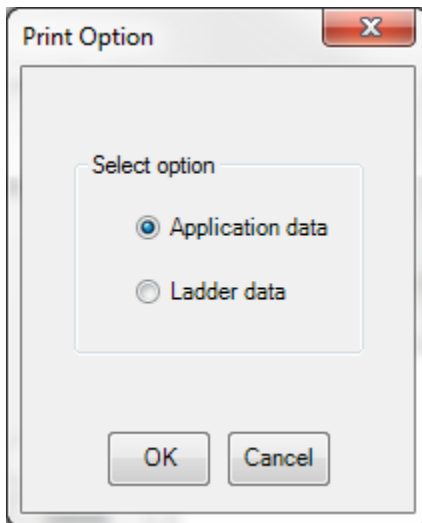
9. Click **OK**.
10. The following may appear:
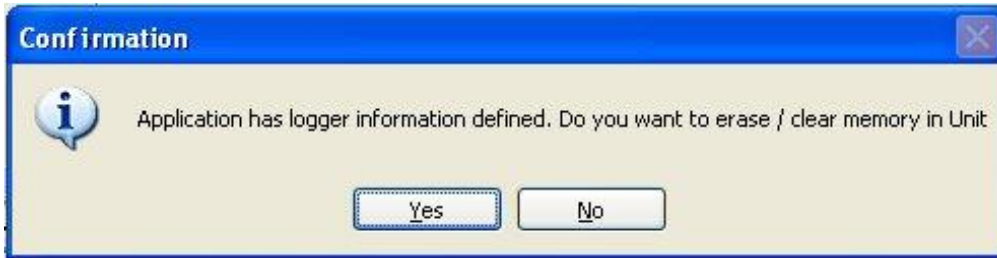
This means you have used the Data Logger feature in your project, so MAPware-7000 is asking if you want to clear the memory in the MLC Series that is used to store data log information.

11. When the download is finished, click **Close**.

 Note: Clicking **Abort** during a download may require that you perform a download of the original project as the existing project in the unit may have already been erased.
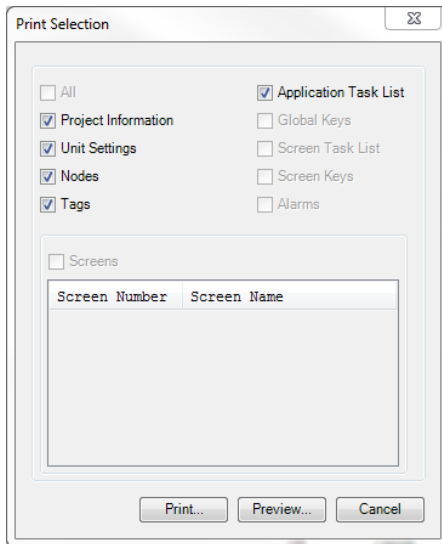
Printing a hardcopy of projects

The MAPware-7000 software allows printing a hardcopy of the project to a printer. Open the project you wish to print in MAPware-7000, then:

1. Click **Project > Print** to display the **Print Option** box.

2. Select **Application data** to print module settings, tags, tasks, etc.. Click **OK** to display the **Print Selection** dialog box.

   a. **All** – Prints all screen data (not available for MLC Series).
   b. **Project Information** – This is general information about the project such as title, author, date and time last edited.
   c. **Unit Settings** – Serial printer port settings.
   d. **Nodes** – Assigned PLC drivers to the COM ports.
   e. **Tags** – The **Tag Database**.
   f. **Application Task List** – Displays all global tasks.
   g. **Global Keys** – Displays settings for all global function keys (not available for MLC Series).
   h. **Screen Task List** – Displays all tasks created at the screen level (not available for MLC Series).
   i. **Screen Keys** – Displays configuration of all screen-based function keys (not available for MLC Series).
   j. **Alarm** – Displays the alarm database (not available for MLC Series).
   k. **Screens** – Shows the screens created, two per page (not available for MLC Series).

3. Select **Ladder Data** to display the **Print Ladder data** dialog box.
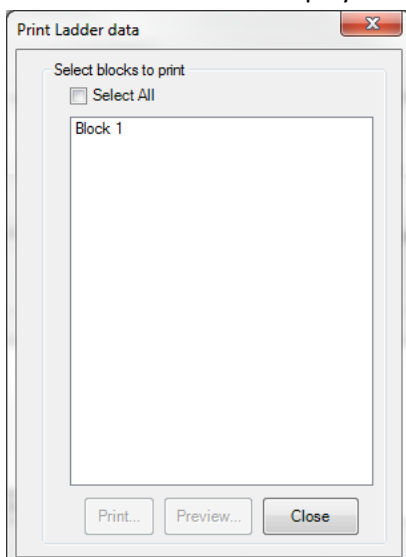
4.   Highlight (click) the blocks that you wish to print, and then click **Preview** or **Print**.

📰 **Note:** The information below is a brief summary of steps required to create a MAPware-7000 project. For a more detailed look, consult the *MAPware-7000 Programming Manual, Maple P/N: 1010-1040*.

Ethernet Settings

The Ethernet port available on the MLC2 and MLC3 models can be used to upload / download projects, monitor PLC logic execution in Online Mode, communicate with an external HMI or PLC, send data logs via FTP (MLC2 only), or to act as a server to display Web screens (MLC3 only). To configure Ethernet settings open the **Project Configuration** window by clicking **Project > Properties > Ethernet**:

Default values for the IP Address, Subnet Mask, Download Port, Default Gateway and Monitoring Port are:

Parameter	Default Setting
MLC IP address	192.168.0.254
Subnet Mask	255.255.255.0
Download Port	5000
Default Gateway	0.0.0.0
Monitoring Port	1100

PC Ethernet Settings

In order to communicate with the MLC, the PC running MAPware-7000 must be configured with an IP address that is on the same Subnet as the MLC. This means that the portion of the IP address masked by the Subnet Mask must be the same on the PC and the MLC and the portion that is not masked must be unique for each device. For example, with the settings above, the PC would have to be configured with an IP address of 192.168.0.XXX where XXX is any number from 0 to 253.

If the PC is connected directly to the MLC Ethernet port, a crossover cable should be used. If the MLC is connected through an Ethernet switch, a straight through Ethernet cable can be used.

DHCP

If a DHCP server is present on the network, the **DHCP** checkbox can be checked to allow the MLC to receive an IP address from the server. The IP address assigned is available in system registers beginning at SW-28 (see **Error! Reference source not found.** for details).

Download Port

The download port is the TCP port the MLC uses to upload and download projects. This must match the port number selected in the Upload / Download window.

Monitoring Port

The Monitoring Port is the TCP port that MAPware-7000 uses to monitor logic executing in an MLC in Online Mode. This port must match the port selected in the **On Line Communication Settings** window when Ethernet is selected as the **Mode** (**Tools > Preferences > Online Communication**).
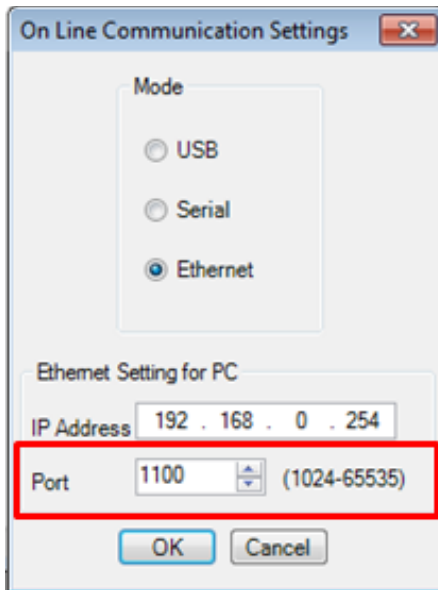
Download Ethernet Settings to an MLC

The settings configured in the **Project Configuration** window above need to be loaded into the MLC in order to take effect. The settings can be downloaded using USB or serial or, if Ethernet communication is already established, through the Ethernet port.

> ⬛ **Note:** If Ethernet is used, communication with the MLC will be lost until PC settings are updated to reflect the changes.

To download the settings, open the download window by selecting **Project > Transfer >Download**, or click the download icon 🔃 . In the **Download to device** window, check the **Ethernet Settings** checkbox in the **Project** section and then click the **Download** button.

Using Online Mode

This mode of operation allows the MAPware-7000 software to connect to the MLC as you would when uploading/downloading a project. Online mode provides the opportunity to use the **Data Window** feature to monitor address tag values and to use the **PLC Control** feature, **Force Enable** feature and **Debug** feature to troubleshoot the ladder logic in the MLC.

To use the Online Mode feature, perform the following steps:

1. Connect your computer to the MLC so that you can upload the project from the device. From **Tools > Preferences** menu, select **Online Communication Mode.**
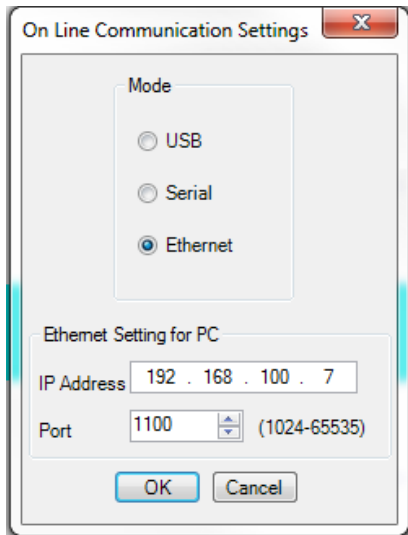
2. Select your preferred connection to the MLC, then click **OK**.
3. To go online, from the **Mode > Online** menu, select one of the following:
   a. **With Upload** – Select this option to upload the project that is in the MLC before going online. Or click the Go Online button. The **Upload from Device** window appears and the project in the MLC is uploaded into MAPware-7000.
   b. **Without Upload** – Select this option to go online without first uploading the project that is in the MLC. If this option is selected, the project that is currently opened in MAPware-7000 will be displayed while online. Normally, it is best to upload the current project that is in the MLC to ensure that what you see in MAPware-7000 matches the ladder logic that is in the device. If you know that the currently opened project in MAPware-7000 is the same as the project that is in the MLC, then selecting this option reduces the time required to get online. Also, if you want to simply use the **Data Window** feature to monitor what the current values are in the MLC then it is not necessary to upload the project from the device.
4. The current state of the MLC (i.e. Run/Halt mode) is also captured. After going online, MAPware-7000 will automatically display **Block 1** ladder logic in the Main program section.
   📑 **Note**: While online, you cannot make edits to your project or any changes to the project that resides in the MLC. If you wish to make any edits, you must go offline by clicking the Offline button or from the Mode menu, click Offline. When online, you can do the following:
   a. Use the **Data Window** feature – The **Data Window** is simply a table that you can configure to monitor any tag addresses in the MLC when online. To display the **Data Window** while online, click the **Data Window** button, press the F10 hot key, or from the View menu, click Data Monitor Window.
   b. Display **Logic Blocks** – You can also view any of the logic blocks of the project while online. From the Block menu, click Open. Or from the Window menu, click the Block that you wish to see:
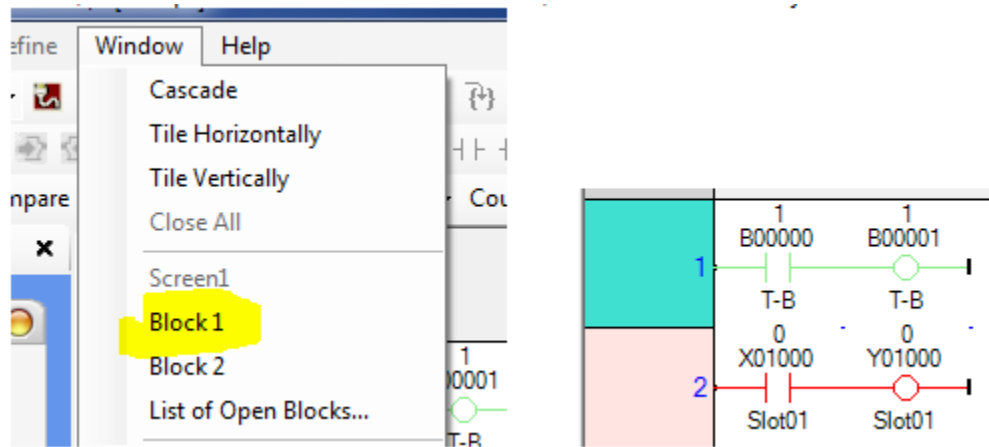
Notice that when using Online mode, the color red is used to indicate that the power is not flowing through a rung. The color green indicates the rung is active. You can change the current state of input coils or input values of logic instructions by double-clicking the input coil or instruction:
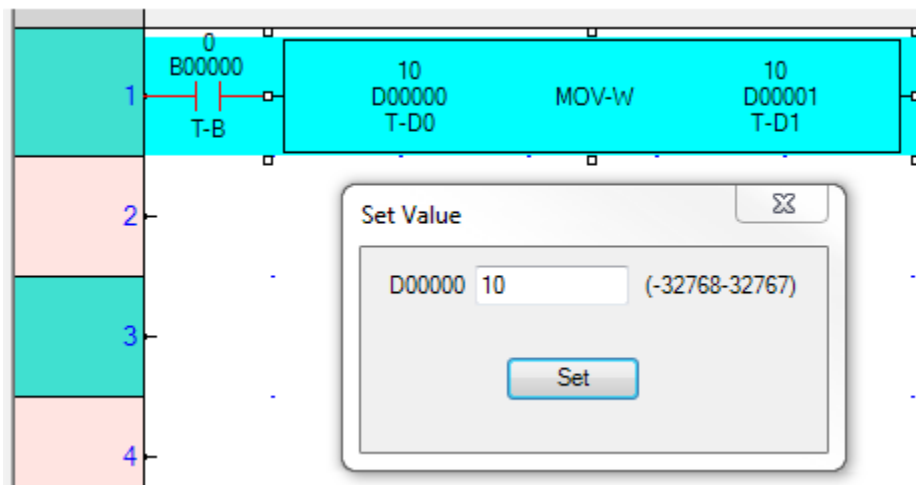

c. Force input (X) coils and output (Y) coils – Use the Enable Force feature to override any input/output coils connected to physical I/O modules.

d. Change operating mode of the MLC – The MLC operating mode can be changed (i.e. Run, Halt, Error Reset, etc.) when online using the PLC Control feature.

e. Debug the MLC ladder program – Use the tools available in the Debug feature to troubleshoot the ladder logic in the MLC.

Things you can't do while online:

- Online mode will not allow you to change/edit the project that is currently in the MLC. To do this, you must go offline, use Edit mode to make changes to the project that has been uploaded from the MLC, then download the project to the MLC.

- Make changes to what is being viewed in the Data Monitor Window when in Native Ladder mode. Although you can view the **Data Window**, you must go offline to make any add/delete any tag registers from the window, then go back online. When is IEC mode, the **Data Window** can be edited while online.

To exit the Online Mode feature, click the Offline button or from the **Mode** menu, click **Offline**.

> 🖳 **Note:** Clicking the **Halt** button does not take you offline – it will put the MLC in stop mode but you will still be online.

> 🖳 **Note:** Clicking the **Run** button does not put you in online mode. When online, clicking this button will change the operating mode of the MLC to run mode. If you click this button while offline, you will activate the off-line simulation mode.

Using Debug Mode

The **Debug** feature provides various functions that can be helpful when troubleshooting the ladder logic in your MLC. To use the **Debug** options, you must be online. **Debug** mode is currently available only when Native Ladder Logic is selected.
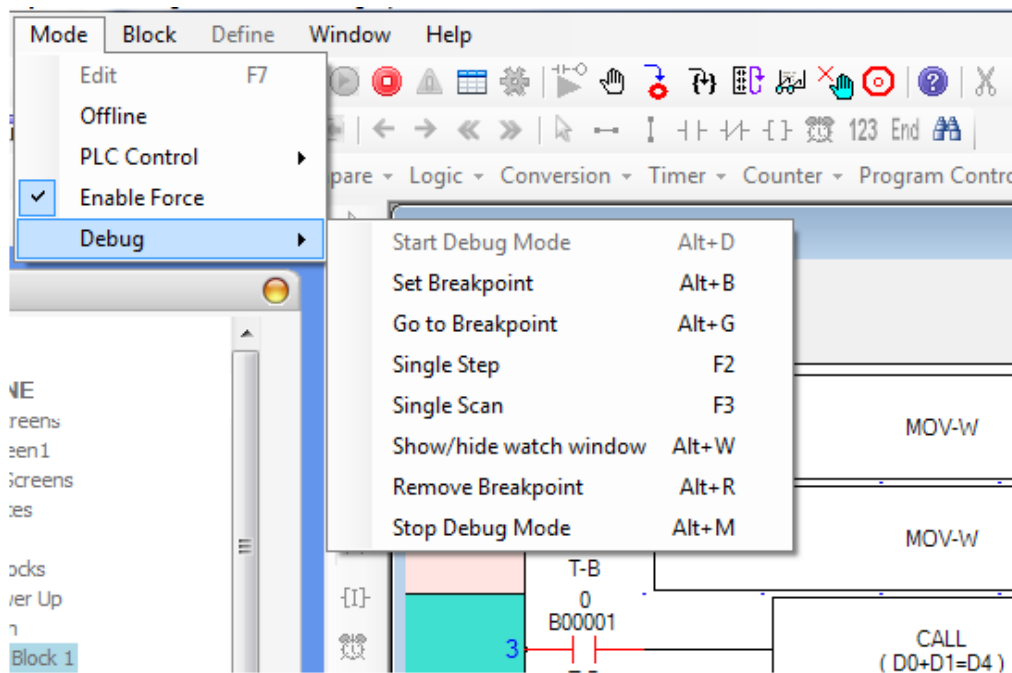
Debug has the following options:

- **Start Debug Mode** – Starts debug mode. In MAPware-7000, the Status Window depicts that the MLC is in Debug mode:
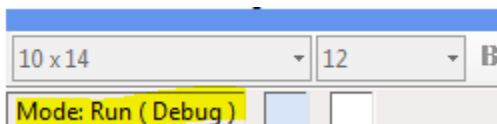
  For the MLC products, the Run LED on the device flashes to indicate Debug mode. The other options below become active once debug mode is started.

- **Set Breakpoint** – Set a breakpoint with a logic block:
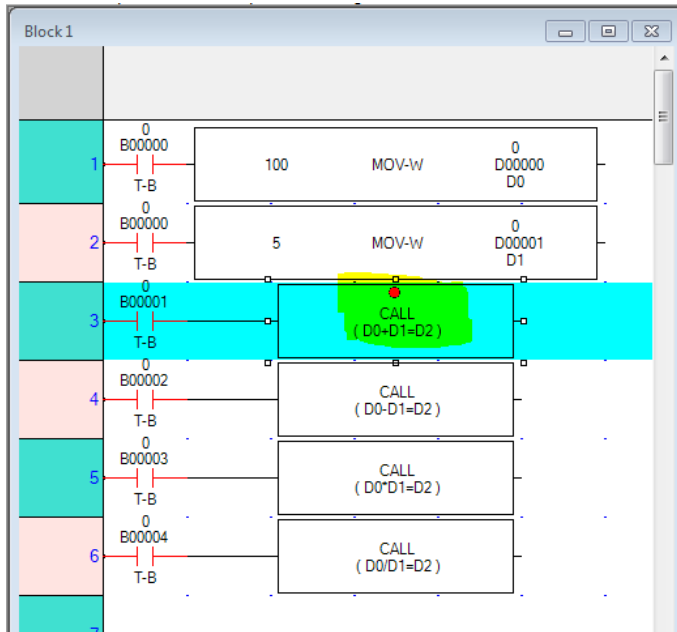
To set a breakpoint, you must first highlight an instruction (ex. CALL subroutine). A red dot appears where the breakpoint is set. Multiple breakpoints can be created. When enabled (see below), a breakpoint is a location where ladder logic execution is temporarily suspended until a command is given to continue. Note that setting a breakpoint does not cause a halt in the execution of the ladder logic. To do that, you must use the Go to Breakpoint command.

- **Go to Breakpoint** – After a breakpoint is created, this function causes the ladder logic to execute until the breakpoint is reached. At that point, execution is temporarily suspended. For multiple breakpoints, the next breakpoint on the list is executed. The instruction where the last breakpoint caused a temporary halt in execution will be highlighted in yellow:
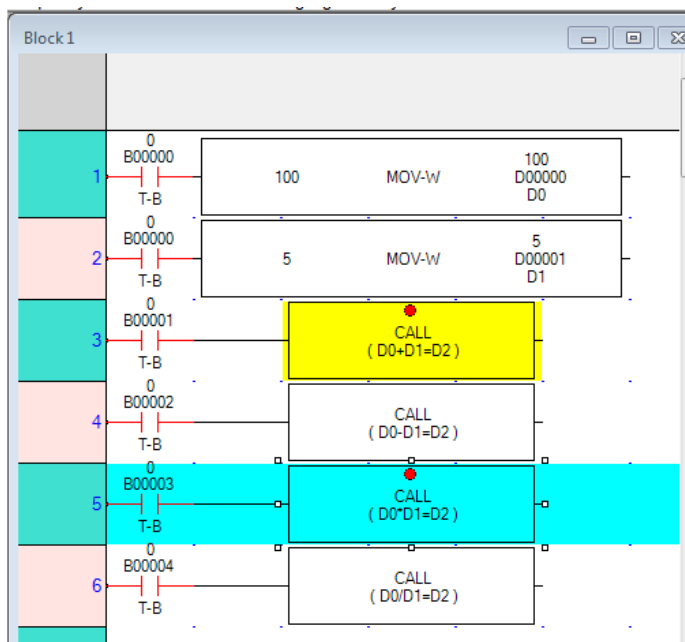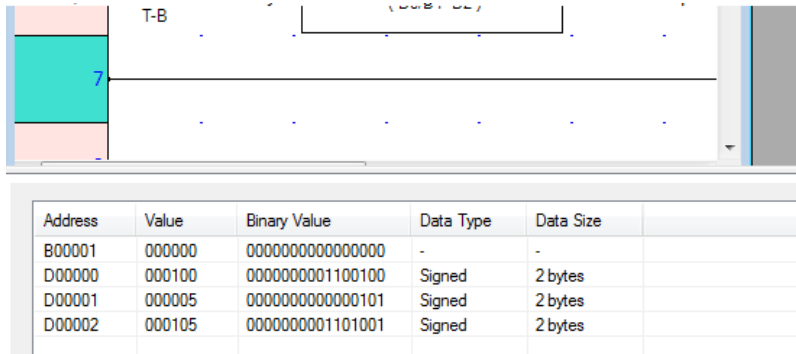
To get the ladder logic to run again after a breakpoint has tripped, you must either press the Go to Breakpoint command again, or you must press the Single Step command.

- **Single Step** – Executes ladder logic one instruction at a time. After an instruction is executed, the debug moves to the next instruction. The instruction that was last executed will be highlighted in yellow.
- **Single Scan** – Forces the ladder logic to perform a single scan, then stops at the current active (highlighted) step instruction.
  🖅 **Note:** The Single Scan command cannot be used when breakpoints are set.
- **Show/hide watch window** – This option displays a watch window. Similar to the Data Monitor Window, a watch window is only available in Debug mode and is used to watch specified address tags.

Address	Value	Binary Value	Data Type	Data Size
B00001	000000	0000000000000000	-	-
D00000	000100	0000000001100100	Signed	2 bytes
D00001	000005	0000000000000101	Signed	2 bytes
D00002	000105	0000000001101001	Signed	2 bytes

- **Remove Breakpoint** – Displays a list of all breakpoints. Select the breakpoint that you wish to remove.
- **Stop Debug Mode** – To exit debug mode. A dialog window will ask if you want to leave the MLC in run mode or halt mode.

🖅 **Notes:**

- Internal System Registers SW00150~SW00171 are used by the MLC during debug mode. Therefore, do not use or modify these registers.
- Debugging is supported for the Main logic blocks only. It is currently not supported for subroutines, I/O Interrupt routines, or Timer Interrupt routines.
- Debugging has no effect on execution of the PLC communications drivers, task execution, screen execution, etc. It only affects execution of the ladder logic.
- When you are done Debugging, don't forget to go Offline to make any changes/edit to your project.

Data Monitor

The Data Monitor feature is a window available in the MAPware-7000 software that displays data values in MLC internal memory. The entire internal memory of the MLC Series is accessible using the Data Monitor but you select what areas of memory you wish to monitor. In addition to reading values, you can modify data online or create a table of preset values to download to the registers. With this feature, you can quickly view data in table format or to initialize areas of memory in the MLC.

To access the Data Monitor window, select **View > Data Monitor Window** from the standard menu, or click the Data Monitor icon ▦ in the standard toolbar, or click the **Data Monitor** folder in the **Project Information**

Window.

By default, the monitor shows a block of ten Timer (T) memory registers. To create or edit additional blocks of memory, right-click anywhere on the **Data Monitor** window to display the context menu.

Create a New Block

By clicking the **New Block** option in the dialog box above, the following is displayed:

- **Address**: Use the Address pull-down box to select the area of memory (see *Appendix A – Internal Memory Address* for more information on internal memory). In the numeric entry box, enter the starting address of the memory selected.
- **Size**: This number represents the block of memory (registers) monitored. For example, if D0 is entered into the Address section and 16 is entered in Size, the Data Monitor will show a range of addresses from D0 to D15.
- **Data Type**: This option varies according to the memory area selected. When available, select how the data is interpreted and displayed in the Data Monitor window. Options are Signed, Unsigned, and Hexadecimal.
- **Data Size**: Each register monitored can be a bit, (i.e. B memory), 2 bytes, or 4 bytes.
- **Color**: You can display the block of memory using a particular color.

Click **OK** to add the block.

Delete a Block

Click **Delete Block** to display the following:

The **Delete Block** dialog box displays all of the blocks currently configured, along with the size of each block. To delete a particular block, simply highlight (click) the block you wish to delete and click the **OK** button.

Editing a Block

To modify a block listed in the Data Monitor window, right-click any of the addresses listed within the block to display the context menu. Then click the **Edit Block** option:
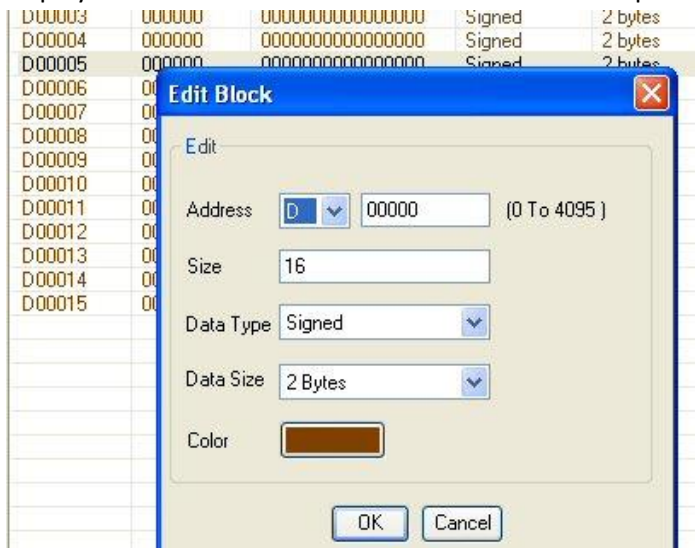
See 'Creating a Block' for information on what each field does.

Importing/Exporting Data

The **Data Monitor** table can be exported as a CSV (comma-separated value) file to be used by another project or edited outside of the MAPware-7000 software. Then import the file into an existing project.

To Export

Right-click anywhere in the **Data Monitor** window to display the context menu. Click **Export Data** from the context menu.

In the **Save As** box, enter the name you wish to give to the CSV file. Then click **OK**.

To Import

Right-click anywhere in the **Data Monitor** window to display the context menu. Click **Import Data**.

In the **Open** box, locate the CSV file that you wish to import, and then click to highlight it.

Click **OK**. A popup dialog box appears asking if you wish to overwrite any of the values in existing memory registers. When editing the CSV file, you can change the address, value in each address, data format and/or byte length. By clicking **Yes**, changes are made to memory addresses that already exist. Clicking **No** prevents the imported data from altering any addresses that already exist in the file – only new addresses are added.
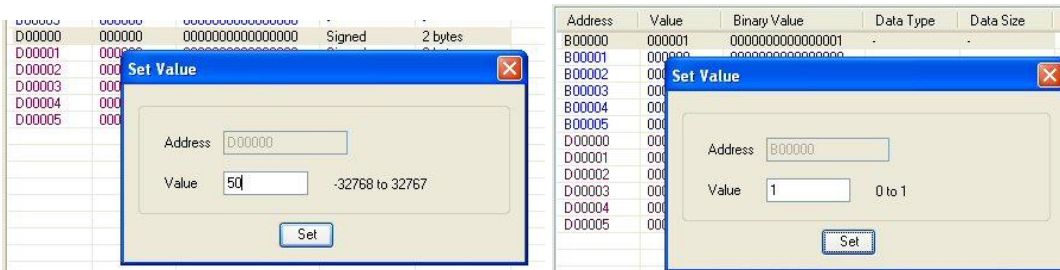
Using Data Monitor to Download Preset Values

After you have created the data registers that you wish to monitor, you may wish to preload these registers with values. This is easily done by configuring a value for each register, then downloading the values into the MLC unit.

1. Open the project you wish to download to the MLC.
2. Configure the registers in the **Data Monitor** window (see steps above).

3. Double-click each register to set a value.

   Note that for bit registers (i.e. B memory), although the data is represented under the Binary Value column with 16 bits, it is really a single bit register. Therefore, when entering a value, you have two options: 0 or 1.

4. Continue clicking each register until you have entered all of your preset values.

5. Next, save and compile your project, then click the Download icon to display the **Download to device** screen.
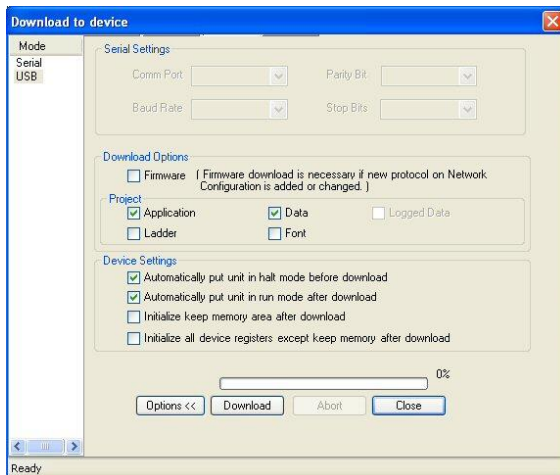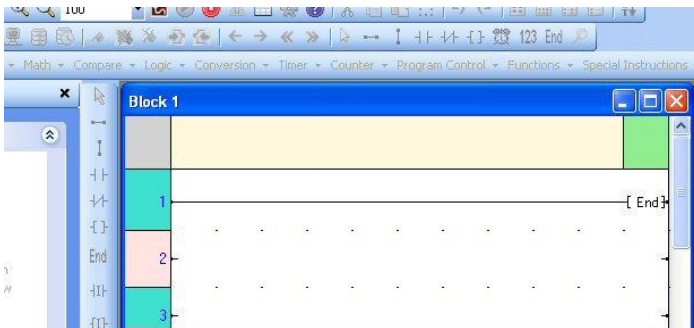
6. Under **Project**, make sure that **Data** is checked. This will download the preset values in the **Data Monitor** window to the MLC. Click **Download** to begin.

Monitoring Data using the Data Monitor

Once you have created your **Data Monitor** window and downloaded your preset values (this is an optional step), you are ready to monitor the MLC Series.

1. Start the MAPware-7000 (it is not necessary to open a project).
2. Connect your computer to the MLC you wish to monitor.
3. Click the Go Online ![icon] button in the standard toolbar. MAPware-7000 will go online with the PLC using the currently open application.
4. MAPware-7000 displays a message stating that it is loading data from the MLC (this takes a few minutes). When finished, the MAPware-7000 software displays the main Logic block.

5. Click the Data Monitor ![icon] icon or press F10 on your keyboard to display the **Data Monitor** window.

6. You can change values while online with the MLC simply by double-clicking a register.

7. The new value is sent to the MLC immediately after you press the **Set** button.
8. You can also place the MLC in **Halt** mode. In **Halt** mode, the MLC discontinues running the project .**Halt** mode is useful if you wish to change values in the **Data Monitor** window but not while the MLC is running its project. However, the MLC will clear all internal memory registers (except the retentive memory) when you place the MLC in **Run** mode again.

   To place the unit in Halt mode, click the Halt icon in the standard toolbar. To resume, click the Run icon.

9. When you are finished using the **Data Monitor** mode, click the Go Online button (now Go Offline) again to go offline.

Chapter 5 – Tags

Tags are names assigned to internal memory registers of the MLC, contacts of an expansion module, and any PLC data registers/coils of an external PLC. Some system tags are predefined when you first begin a project (see Predefined System Tags below). Other tags are created by the programmer. For example, you must create and assign a tag to every PLC memory address that you wish to read/write to. When using the optional I/O Expansion modules, you must also assign tags in order to use them. The **Tag Database** collects and stores all tags for review and editing.

Once a tag is assigned, you can reference the tag in the logic blocks. Tags have several advantages:

- Tags provide an organized method of tracking all memory addresses used in a project.
- Tags are much more descriptive of the function than the name of the memory address.
- Tags are more easily edited, should a change be required.
- Tags can be exported and imported into other MAPware-7000 projects, regardless of which HMC/MLC Series unit is selected.

Tag Types and Addresses (Native Ladder vs. IEC 61131-3)

Tags created by the user are classified into different types which are stored in different memory locations on the MLC. How these tags appear to the user differs somewhat depending on whether the project is created in Native Ladder mode or in IEC 61131-3 mode.

In Native Ladder projects, tags are defined with a specific address in the MLC's memory. These addresses are grouped into different address spaces according to how a tag defined in each space is intended to be used. Examples are; D registers (general purpose data registers for 16 or 32 bit data), B registers (single bit internal registers), and SW (S) registers (System tags that control how the MLC hardware functions). For a complete list and information on each of the available address spaces refer to *Appendix A – Internal Memory*.

In IEC61131-3 projects, tags do not necessarily have a specific address. Instead they are assigned a type which determines how they are intended to be used. Examples are BOOL (one bit), INT (a 16-bit signed number), and REAL (a 32-bit floating point number). For a complete list and information on each of the types available in IEC 61131-3 projects refer to *Appendix A – Internal Memory*. In some circumstances, IEC 61131-3 tags are assigned specific addresses. For example, retentive registers are assigned addresses and tags that are intended to be accessed from an external PLC using the MLC as a Modbus slave can be assigned a Modbus address.

Predefined System Tags

In addition to user-defined tags, the **Tag Database** contains predefined tags assigned to special internal memory coils (bits) and registers of the MLC Series. These internal memory registers have predefined functions (e.g. beeper control, screen saver timer, RTC settings, etc.) which may be useful in your project. For more information on these predefined system tags, see *Appendix B – Predefined Systems Tags*.

Creating and Modifying a Tag

To create a new tag and assign it to an internal memory address or PLC address, you must first open the **Tag Database**.

- Click **Define > Tag Database** from the standard menus or
- Click the **Tag Database** icon [icon] located in the Layout Toolbar or
- Click the **Tags** folder in the **Project Information Window**

The **Tag Database** displays:

Each assigned tag in the database has attributes arranged and presented as columns:

- **Tag No.** – This is the tag number assigned to the tag when created. Tags are assigned in numerical order.
- **Tag Name** – When you create a tag, you must provide a name for the tag.
- **Data Type** (IEC 61131-3 projects only) – Defines the type (BOOL, INT, DINT, etc.) of the tag.
- **Tag** (Native Ladder projects only) – Specifies the size of the tag in bytes (1, 2, 4, or bit). For some system tags, the data may occupy either the high or low byte of a two byte register. If that is the case High or Low will be specified in this column.
- **Attribute** – Indicates if tag has Read Only or Read/Write access.

- **Tag Address** – This is the specific internal memory address or HMI/PLC memory address assigned to the tag.
- **Port** – If the address assigned to the tag comes from one of the external ports (i.e. Com1, Com2, Ethernet), then the port is shown.
- **Node Name** – Where the tag memory address is located (i.e. internal memory or external connection to a HMI/PLC).
- **Tag Category** – Either Default Tag (predefined tag) or User Defined Tag (programmer defined).
- **Export Tag** – Check this box when exporting only selected tags (see Export Tag feature).

To modify the **Tag Database**, use the buttons displayed in the **Tag Database** popup window (see above) or right-click the **Tags** folder in the **Project Information Window** to display the context menu.
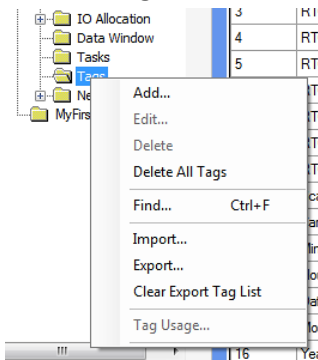
Adding a New Tag

In the **Tag Database**, click the **Add Tag** button to display the **Add Tag** dialog box.
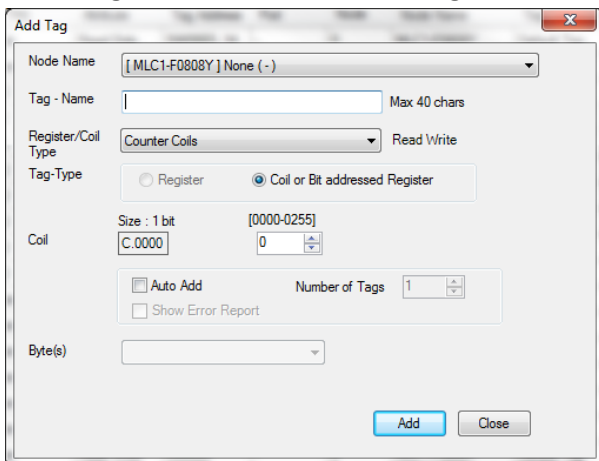
- **Node Name**: Click the dropdown list to select the node (i.e. internal memory or HMI/PLC communications driver attached to the COM port) where the target address is located.
- **Tag Name**: Enter a name that you wish to assign to the tag address (max 40 characters).
- **Register/Coil Type**: Click the list to select the proper memory address or tag type. Each listing corresponds to a specific memory type within the MLC Series or an external HMI/PLC (for more information on HMI/PLC memory types, download the applicable Controller Information Sheet from our website). For information on the internal data types refer to *Appendix A*. The Read/Write (Read Only) descriptor indicates if selected memory area can be written to by the MLC Series.

- **Tag Type**: Options are Register and Coil/Bit Addressed Register. If both options are available then the memory area selected is interpreted by the MLC Series as either a data register or a bit.
- **Coil/Register**: The first field (Size: 2 bytes, 1 bit, etc.) indicates how much memory is allocated to each address in this memory range. The enclosed address underneath represents the particular address targeted after entries are made in the 'address input' fields located to the right. Below are some examples:
  - MLC Series Internal Address

  - GE Micro (SNP-X) Address

  - Allen Bradley MicroLogix Address

- **Auto-add**: By checking this option, you can quickly add a series of tags based upon a consecutive set of registers. When used, the **Tag Name** will have a five digit number appended (ex. 00000, 00001, etc.)

  - **Number of Tags**: Enter the total number of consecutive tags you wish to create when using the Auto-add feature.

- o **Show Error Report**: If checked, then the MAPware-7000 software displays a Text Only file (ErrorTag_AutoAddTag.txt) that shows any errors that may have occurred when using the Auto-add feature.

- **Byte(s)**: Some memory registers can be configured as a single byte, word, or double-word register:
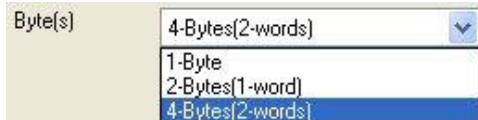
- **Scope** (IEC 61131-3 mode only) – There are three radio buttons that can be selected to define a tag's scope.
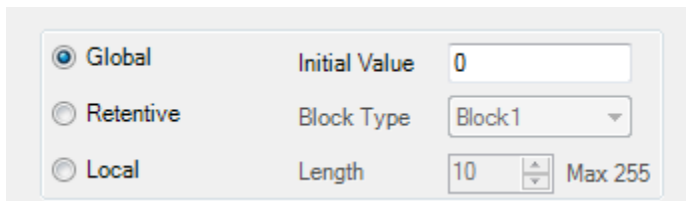
  - o **Global** – The tag can be used throughout the project. Tags created as Global can be changed to Retentive during project development.
  - o **Retentive** – Same as Global but the value is retained through a power cycle. Tags created as Retentive can be changed to Global during project development. For more information on this option refer to Chapter 8 – Miscellaneous Features – Keep Memory Area and Retentive Tags.
  - o **Local** – The tag can only be used in a specific logic block. The Block is selected in the **Block Type** dropdown list. Local tags can still be used in application objects such as screens and tasks.
- **Initial Value** (IEC 61131-3 projects only) – Specify the default value the tag will have when the project starts. This option is not available for Strings or Retentive Registers or if *Retentive* is selected for the scope (see item 9b above).
- **Block Type** (IEC 61131-3 projects only) – Specify the block where a Local tag can be used when the Local option is chosen as the Scope (see item 9c above).
- **Length** (IEC 61131-3 projects only) – Specify the length for a string. Available only when the Register / Coil Type is set to string.

Click the **Add** button to add the new tag to the **Tag Database**. The **Add Tag** dialog box remains on screen so that you can create other tags. When you have finished, click the **Close** button.

Deleting a Tag

To delete a tag that is in the **Tag Database**, simply:

1. Open the **Tag Database**.
2. Highlight (click) the tag.

3. Click the **Delete** button (in the Tag Information window) or right-click to display the context menu.
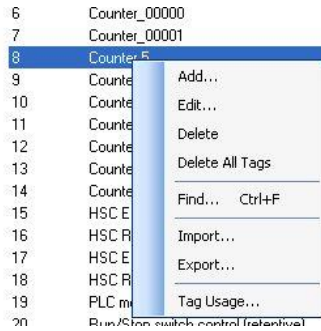
4. A popup dialog box appears asking if you are sure that you want to delete the tag. Click **Yes**. Before you can delete a tag, you must remove any references to the tag in logic blocks.

📝 **Note:** All tags defined as **Default Tags** are system tags and cannot be deleted.

You can delete multiple tags simultaneously by highlighting one tag in the **Tag Database**, and then holding down the CTRL (or SHIFT) key and selecting other tags.

You also have the option of deleting all user-defined tags in the **Tag Database** by clicking **Delete All Tags**.

Editing a Tag

To edit a tag:

1. Open the **Tag Database**.
2. Highlight the tag.
3. Click the **Edit** button (in the Tag Information window) or right-click to display the context menu, and then click **Edit.**
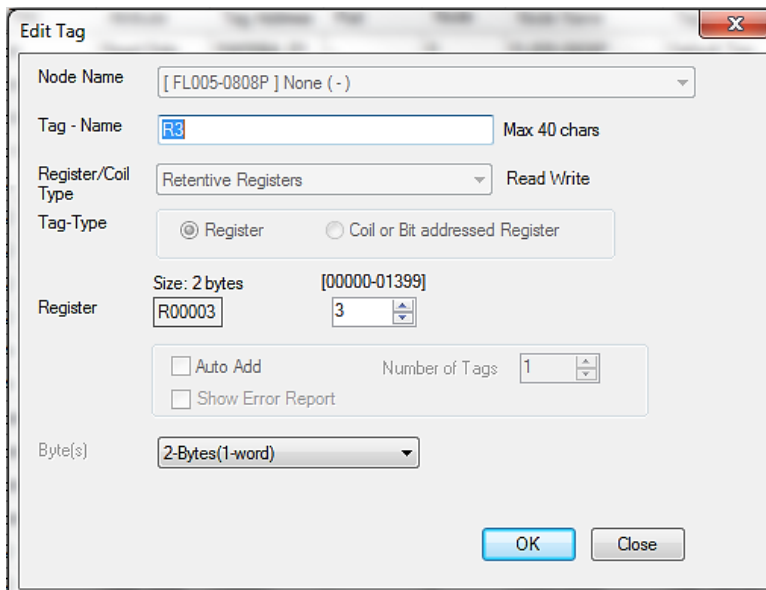
In **Edit** mode, you can only change the name of the tag, the target address and bytes allocated. Any other changes require that you delete the tag entirely, and then reenter.

Finding Tags

If you have created many tags and have a large database, it may be difficult to locate a particular tag. To help you locate a tag quickly, right-click anywhere in the **Tag Database** to display the context menu.
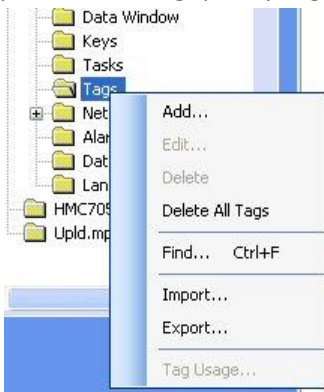
Click the **Find** option to display the **Find** dialog box.
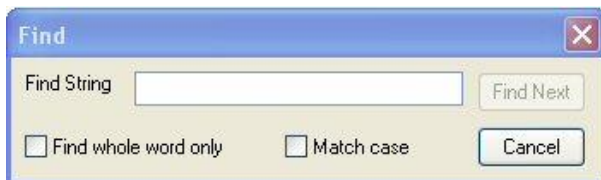
Enter the name of the tag or tag address. MAPware-7000 will then search through the database and highlight the tag.

You can also reorder the **Tag Database** by clicking any of the column headings. The tags reorder alphabetically according to the entries in that column.

Tag Usage

This feature allows you to select a tag in the **Tag Database** and view where used in your project.
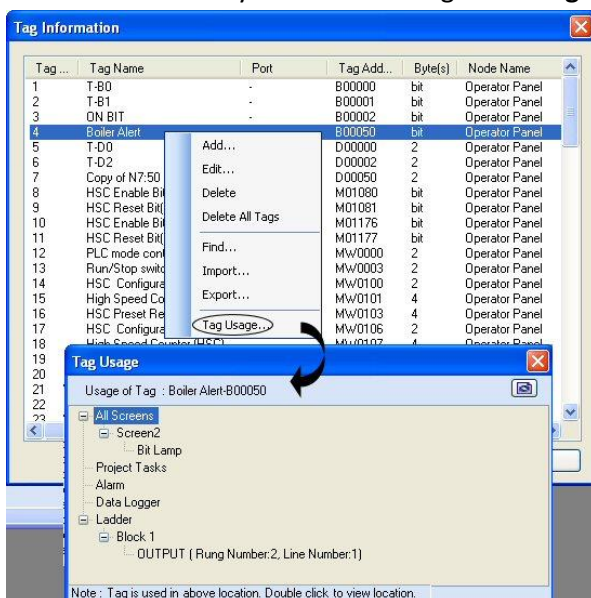
Importing and Exporting Tags

The **Tag Database** created in MAPware-7000 exports as a CSV (comma-separated value) file. You can edit this file then import back into MAPware-7000. This provides the following advantages:

- Tag databases can be moved from one project to another without copying the entire project.
- If you wish to modify many tags, this is quickly done by editing a CSV file.
- In some cases, you may be able to import the PLC addresses you use in your external PLC ladder logic program, thus saving time.

Export Tags

Open the **Tag Database** and right-click to display the context menu.
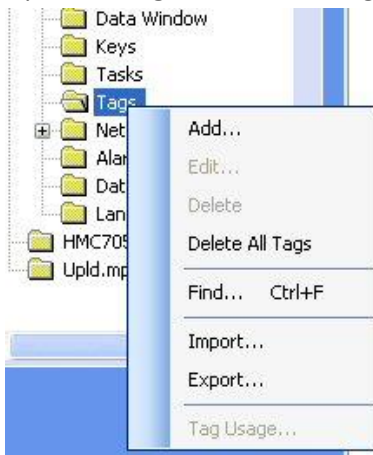
Select **Export** to bring up the **Export Tags** dialog.
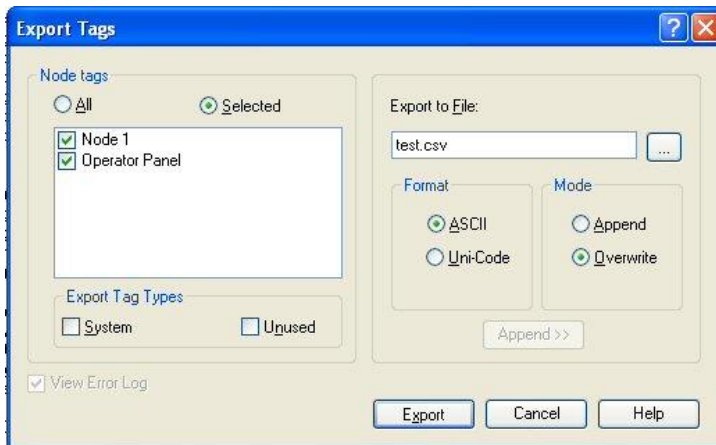
- **Node tags** – Select All to export all nodes (internal memory and tags assigned to all Com ports). Check Selected and use the checkboxes to filter which nodes you wish to export.
- **Export Tag Types** – Check System to export internal system (S and SW register) memory tags in the **Tag Database**. Check Unused to export <u>only</u> tags that are not used in the project.
- **Export to File** – For **Format**, select ASCII or Unicode. For **Mode**, select Append (add to existing file) or Overwrite.

Click the **Export** button. The file is created.

Sample Exported Tag Database File in Excel

Import Tags

Open the **Tag Database** and right-click to display the context menu:
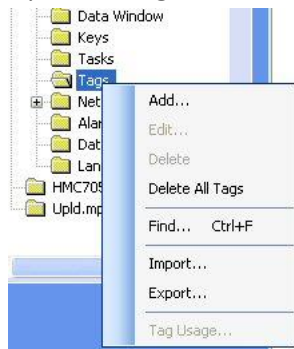
Select **Import.**
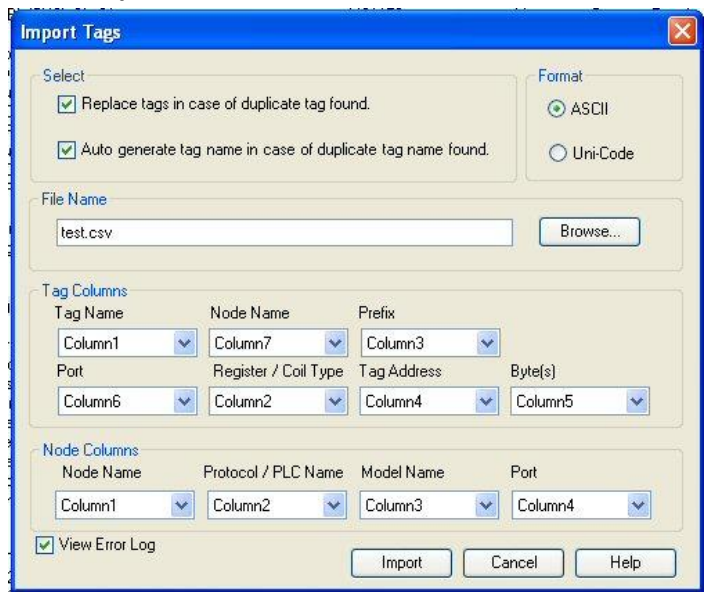
- **Replace tags in case of duplicate tag found** – Check this box if you wish MAPware-7000 to replace a tag name in the **Tag Database** with one of the same name in the imported file.
- **Auto generate tag name in case of duplicate tag name found** – Checking this box will make MAPware-7000 append a number to the duplicate tag name.
- **Tag Columns** – In order to import a CSV file, you must specify where each tag column (i.e. Tag Name, Tag Address, Node Name, etc.) is located in the CSV file. By default, the columns arrange in the same format used when exporting the **Tag Database** as a CSV file.
- **View error log** – If checked, this will display a text file after MAPware-1000 has finished importing that shows any errors that have occurred.

Chapter 6 – Logic Blocks

In MAPware-7000, the logic blocks are used to create the program logic of the PLC. A logic block is a collection of instructions (either ladder logic or IEC61131-3) and variables that executes when the MLC runs the logic block. You can create multiple logic blocks in each MLC project. With several logic blocks, you can split a long program into segments that are easier to read. Up to 256 blocks can be created.

The blocks are executed according to the type of logic block (i.e. main, subroutine, power-up) and the order in which they are listed (Block Number) in the **Project Information > Logic Blocks** folder.
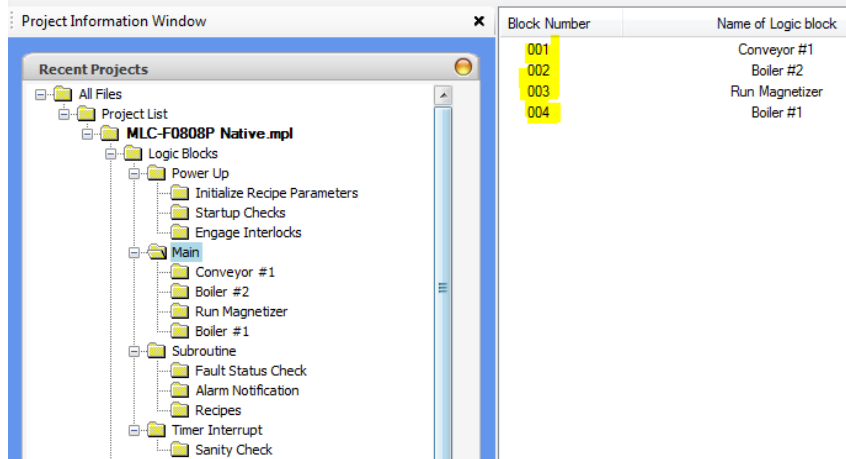
During operation, the MLC unit scans (executes) the logic code in a continuous repeatable sequence in this order:

1st (initial) scan

Hardware Initialization	Power up the CPU, checks memory, performs diagnostic test, initializes external ports and all I/O modules.
Process all Hardware Inputs	Reads all physical inputs of the I/O modules to determine the current state.
Mode	Determines mode of operation (i.e. Run or Halt)
I/O Interrupts	Executes any hardware interrupts.
Timer Interrupts	Executes any timer interrupts.
Power-up	Executes any power-up logic blocks.
Main	Executes all Main logic blocks.
Process all Hardware Outputs	Writes all physical outputs of I/O modules.
Service Peripherals	Service the external ports (USB, serial)

2nd scan (and each scan that follows)

Process all Hardware Inputs	Reads all physical inputs of the I/O modules to determine the current state.
Mode	Determines mode of operation (i.e Run or Halt)
I/O Interrupts	Executes any hardware interrupts.
Timer Interrupts	Executes any timer interrupts.
Main	Executes all Main logic blocks.
Process all Hardware Outputs	Writes all physical outputs of I/O modules.
Service Peripherals	Service the external ports (USB, serial)

Types of Logic Blocks

There are several types of logic blocks that have specific functions when used.

Power Up

Power-up logic blocks are executed once when the MLC is started. They are run before the logic blocks in the Main folder are executed. Therefore, they are often used to initialize values or place the MLC in a particular operations mode.

Main

The Main logic blocks are the core of the program running in the MLC unit. All logic blocks in the Main folder are executed once during each scan.

Subroutine

A subroutine is a logic block called by another logic block (main, subroutine, or interrupt). A maximum of 256 subroutines can be created. A subroutine is called using the CALL instruction along with the number assigned to the subroutine. Each subroutine must end with a RET instruction. When the RET instruction in a subroutine is reached, program execution returns to the calling logic block at the next rung after the CALL instruction.
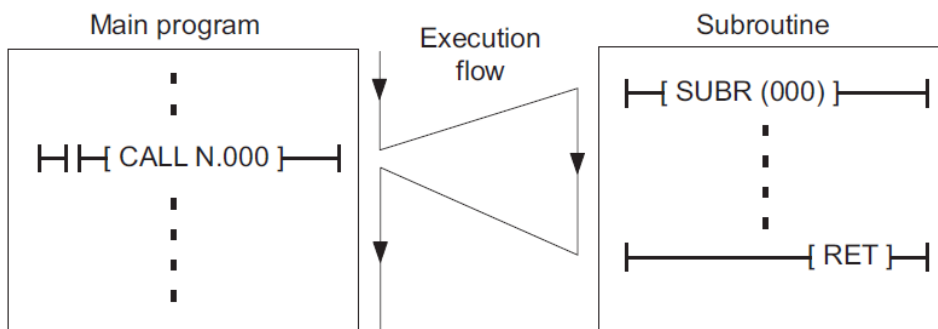
Timer Interrupt

This is a logic block that is called based upon a time interval set in the **Define > System Parameters** menu.
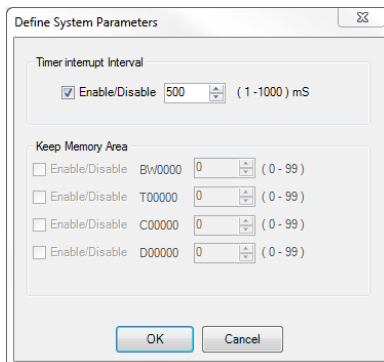
Only one timer interrupt logic block can be created per project. The timer interrupt has the highest priority of all logic blocks – any logic block that is currently executing will halt when the timer interrupt is called (see the Enable Interrupt {EI} and Disable Interrupt {DI} Instructions for information on how to get around this). The

number of instructions and logic in the Timer Interrupt should be kept to a minimum in order to not degrade performance for the other logic blocks. When using native ladder logic, use the IRET instruction to end the Timer Interrupt. Program execution will go back to the logic block that was halted when the Timer Interrupt occurred.

Hardware Interrupts

Hardware interrupt logic blocks are also high priority logic blocks and thus are executed when the designated input channel is triggered, according to the following:

- Any logic block other than a Timer Interrupt or Hardware Interrupt is halted while the triggered hardware interrupt runs.
- If a Timer Interrupt is in progress, the hardware interrupt is processed after the Timer Interrupt.
- If a hardware interrupt and a timer interrupt trigger at the same time, the Timer Interrupt is executed first.
- If more than one hardware interrupt triggers at the same time, the hardware interrupt that is set with the higher priority is executed first.

Only digital inputs on the MLC base module can trigger a hardware interrupt –expansion modules cannot be configured to trigger hardware interrupts. When using native ladder logic, use the IRET instruction to end the Hardware Interrupt. For example, the two hardware interrupt logic blocks are created for an MLC1-F0808P by right-clicking the **Hardware Interrupt** folder in the **Project Information Window** and select **New Logic Block**.
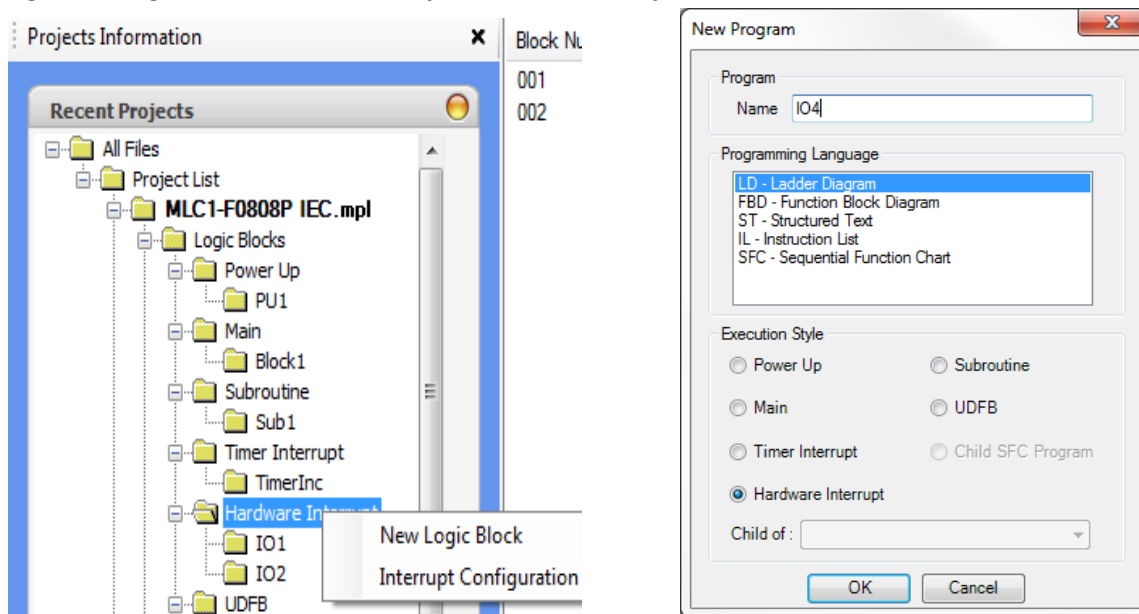
After creating the new logic blocks, they must be assigned to particular digital inputs. Select **Interrupt Configuration.**
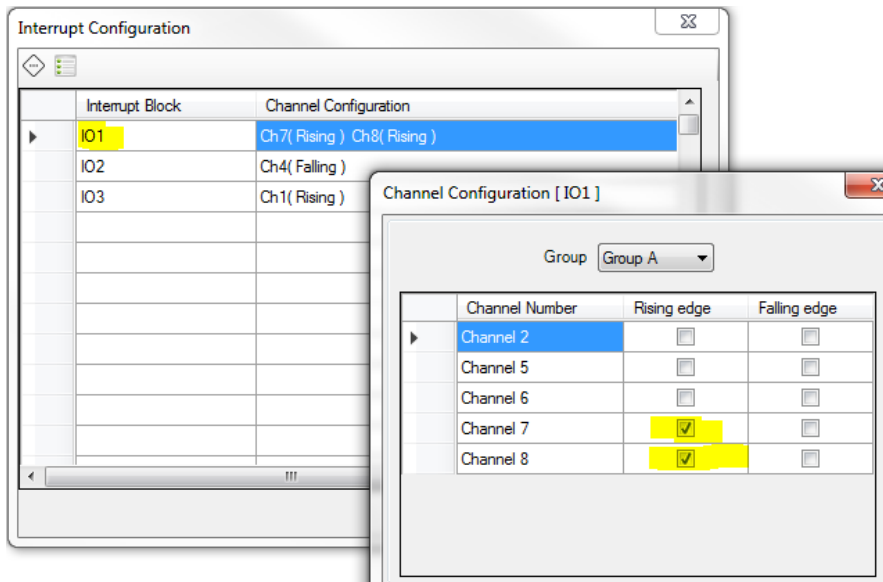
Highlight the Interrupt Block that you want to assign, then click the Channel Configuration [icon] icon. The Input channels fall within two priority groups (A or B). Select the correct group, then select which channel will trigger the hardware interrupt. Activating the trigger can occur on the **Rising edge** (OFF to ON), **Falling edge** (ON to OFF), or both (any change of state). Note that more than one channel can be selected to trigger the event. Also note that Channel 1 corresponds to Input X0, Channel 2 to Input X1, and so on. Finally, click the Interrupt Priority [icon] icon to configure which set of input channels has higher priority.

UDFB (IEC only)

A user-defined function block (UDFB) is a logic block with a unique set of parameters and instance (member variables) data. A UDFB can contain constants, variables, pre-defined instructions and other UDFBs. The primary advantage of UDFBs is that you can create custom instructions (similar to the pre-defined instructions such as TON, Addition, any_to_real, etc.) that can be placed in a logic block. As with pre-defined instructions, a UDFB can be repeatedly used in the logic blocks.

Once defined, multiple instances of the UDFB can be created and executed, (just as is done with a pre-defined instruction), each with its own set of parameters (variables). Each instance has its unique copy of UDFB data, including internal member variables and input/output parameters. When a UDFB instance is called, the logic operates only on that instance's copy of data. Values of the instance are retained from one execution of the UDFB to the next.

Available only when IEC61131-3 is selected, UDFBs require using the LD, FBD, or ST programming languages. The following steps illustrate the concept of using UDFBs and how to create one:

1. Start a project that uses IEC61131-3 as the programming language. In the **Project Information Window**, right-click the **UDFB** folder and select **New Logic Block**.
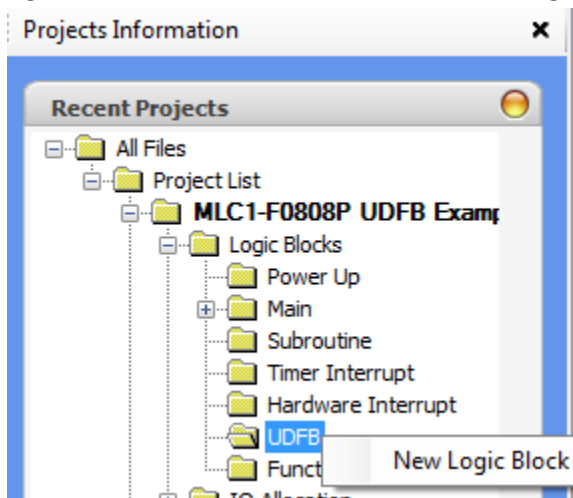
2. In the **New Program** dialog box, select a name for your UDFB. In this example, we want to create a UDFB that converts a temperature value in Fahrenheit (F) to Celsius (C).

3. Make sure that UDFB is selected as the Execution Style, then select which **Programming Language** to use (LD, FBD, or ST).
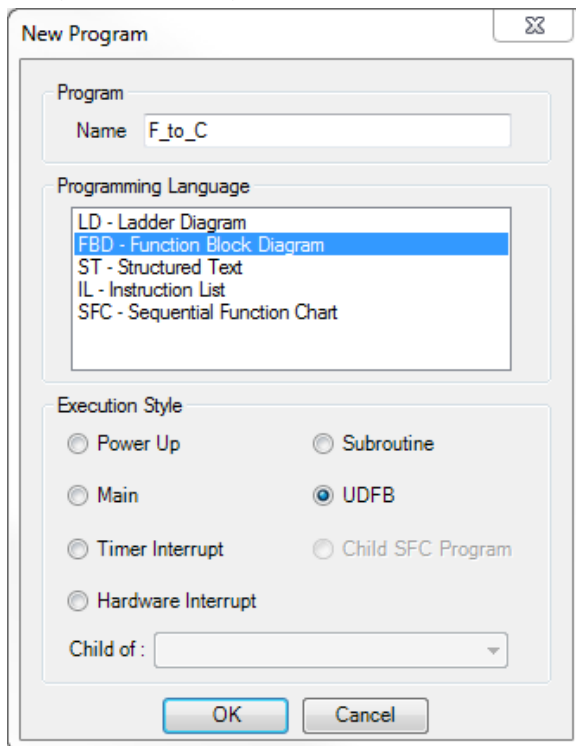
4. Click **OK**.

5. A new folder is created under the **UDFB** folder in the **Project Information Window**. Also note that the UDFB appears under the **Project** folder in the **Instruction List** box.

6. Create the logic shown below:
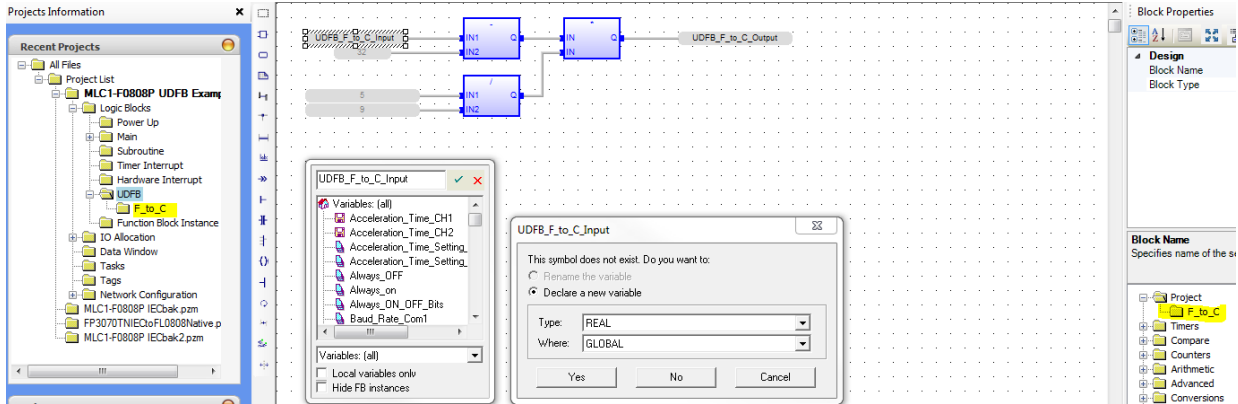
Create two variables (*UDFB_F_to_C_Input* and *UDFB_F_to_C_Output*) with Data Type **REAL** as shown.

✍ **Note:** Be sure to select **Global** for access level in the **Where** field. If you make them Local by selecting *F_to_C*, then these variables will only be available inside the UDFB.

7. There are two variables that we want to be able to access when using the *F_to_C* UDFB in any logic block, *UDFB_F_to_C_Input* and *UDFB_F_to_C_Output*. To do this, we need to add these to the UDFB's **Program Parameter** table. Right-click the *F_to_C* folder and select **Edit Parameters** to bring up the **Program Parameters** dialog.
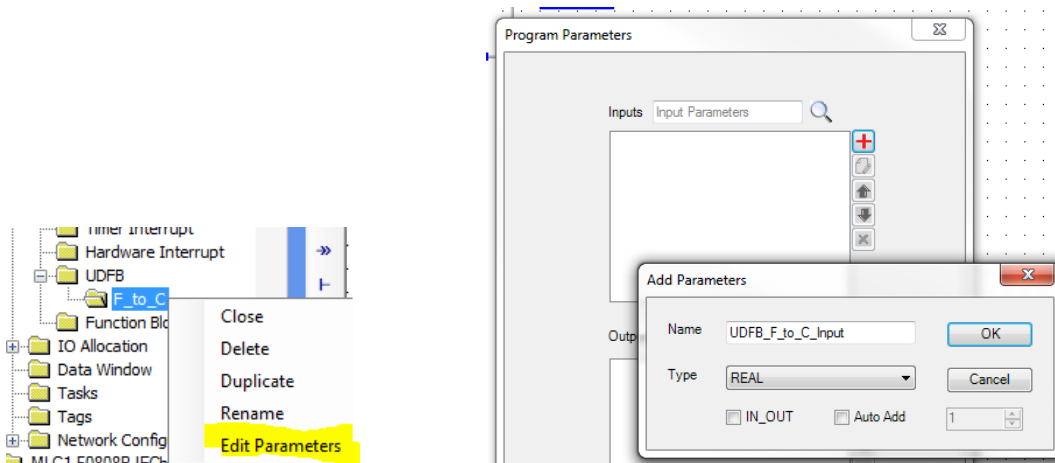
a. **Inputs** Section – These are the input parameters of the UDFB. Enter the input name (*UDFB_F_to_C_Input*) exactly as was done when creating the input variable in the logic, otherwise that value will not be passed into the UDFB.

 i. **IN_OUT** – When checked, this indicates that the parameter is to be used as both an input and output parameter – no need to add the parameter to the Outputs section.

 ii. **Auto Add** – Use this feature when creating similar parameters (ex. IN0, IN1, etc.). MAPware-7000 automatically adds the number of variables based upon the value entered.

b. **Outputs** Section – These are the output parameters of the UDFB. As with Inputs, enter the output name (*UDFB_F_to_C_Output*) exactly as was done when creating the output variable in the logic, otherwise that value cannot be passed out of the UDFB.

8. Once a UDFB is created, it can be used in any of the logic blocks (Power up, Main, Subroutine, Timer Interrupt, or Hardware Interrupt). Using the new UDFB created above, let's add this routine to **Block1** in

the Main logic block. Click **Block1** under the **Main** logic block folder in the **Project Information Window**. On the right side of the MAPware-7000 work area, the **Block Properties** section is shown. Below is a listing of all of the instructions available. Any UDFBs created are listed in the **Project** folder.
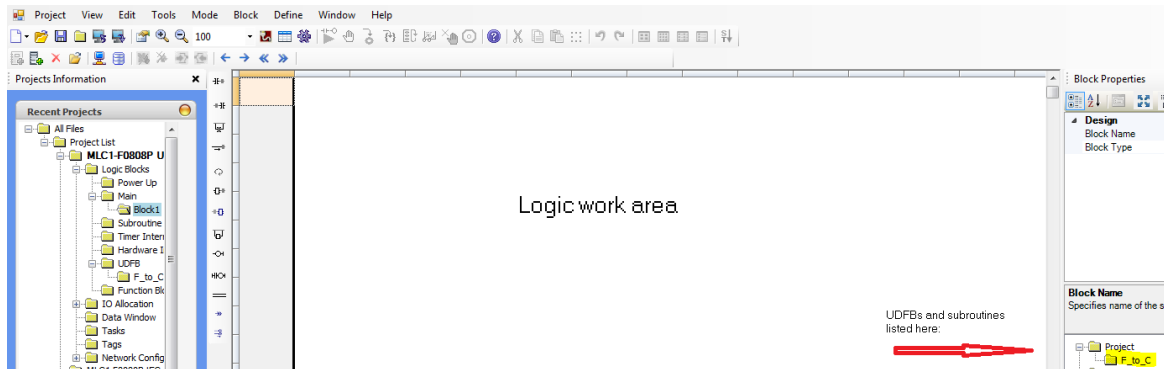
9. Click the *F_to_C* UDFB in the Instruction List, then drag to place an instance in the logic work area:
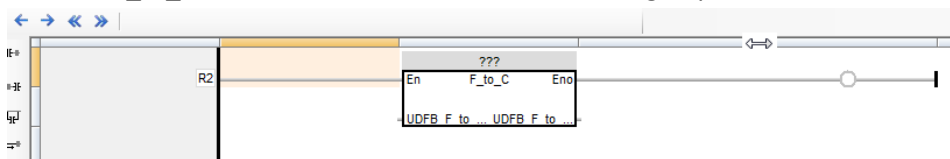
📄 **Note:** You can adjust the size of objects shown in the work area by moving the mouse cursor to the top of the work area where the columns dividers are located. When the mouse cursor is held over one of the dividers (as shown in the picture), it will change to a double-arrow. Then click and drag to adjust the width.

10. Notice the **???** symbol above the *F_to_C* UDFB. This indicates that this is an instance of the UDFB. Before you compile, you must assign a name to the instance. Double-click the **???** symbol to display the **New Tag** popup dialog box. Enter a new name for the UDFB instance:.
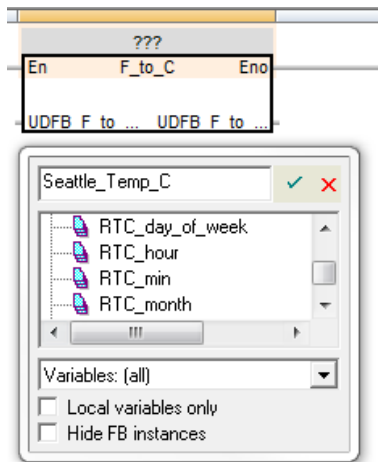
11. Click the Accept ✔ icon. A popup window will state that this tag does not exist yet. Make sure that **Type** *F_to_C* is selected.
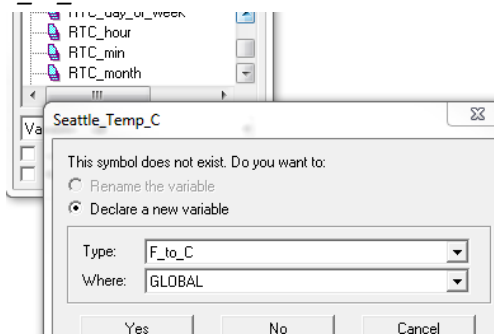
12. Finally, as is done with any instruction, assign variables to the input and output.
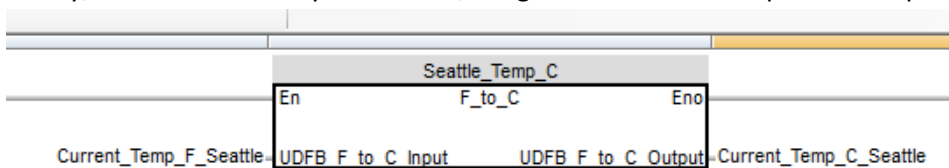
Function Block Instance (IEC only)

The **Function Block Instance** folder is a listing of all instances created in your project. These can be instances of UDFBs that you created or instances of pre-defined instructions (such as a TON). Every time you create a new instance in one of your logic blocks, the instance is automatically added to this list. If you created the UDFB in the section above and used it in the **Main-Block1**, you can click the **Function Block Instance** folder and see it listed:
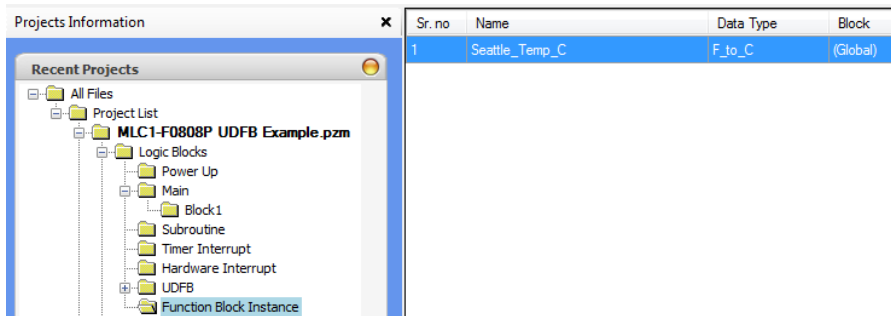
Right-click the instance to display the context menu.
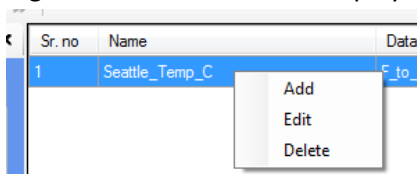
- **Add** – Creates a new instance. This option is handy if you want to create new instances but do not want to place them into your logic blocks just yet.
- **Edit** – Highlight an instance, then click Edit to rename the instance. If the instance is used in any logic block, you must edit the instance in the logic block to reassign to a valid name.

- **Delete** – Deletes the instance.

  ⧉ **Note:** You must delete the instance in all logic blocks that reference the instance before you can delete it.

Editing a Logic Block

As mentioned in the beginning of this chapter, logic blocks are the segments of logic that comprise the program that runs in an MLC PLC. Separating the program logic into blocks helps to organize the code and make it easier to read. This section describes how to perform various functions, such as creating, deleting, renaming, and duplicating logic blocks.

Creating a New Logic Block

Native Ladder

To create a new logic block, right-click the folder (i.e. **Power Up**, **Main**, **Subroutine**, etc.) in the **Project Information Window** of the type of logic block that you wish to create.
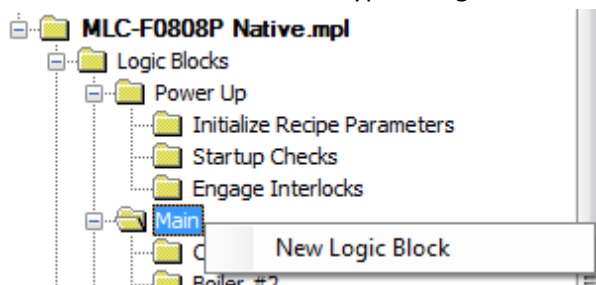
Or with the logic block type folder (i.e. **Power Up**, **Main**, **Subroutine**, etc.) highlighted, simply click the New Block ⧉ icon or right-click anywhere in the work area and select **Add New**.
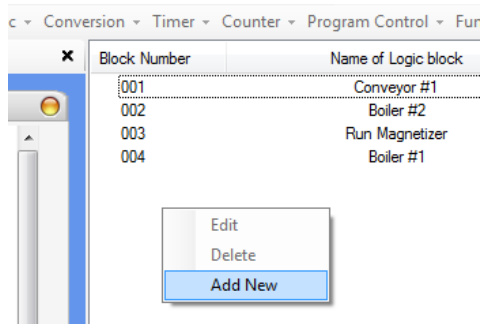
When a new logic block is created in Native Ladder, the work area displays a column of numbers that represent the rungs.
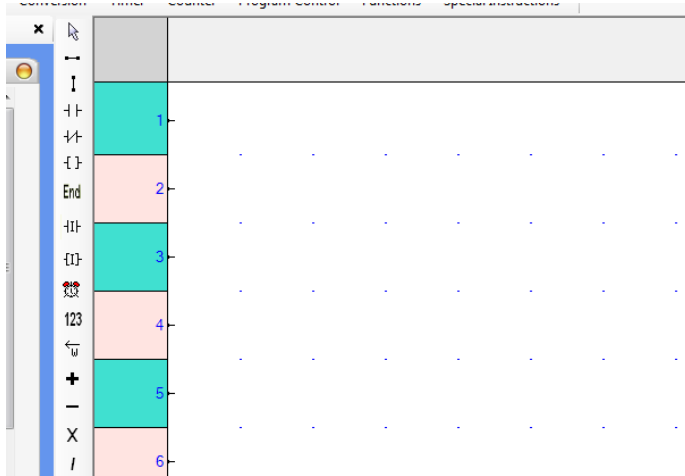
IEC61131-3

To create a new logic block in IEC61131-3 mode, the steps are the same, with the addition of choosing the **Programming Language** when the **New Program** dialog box appears.
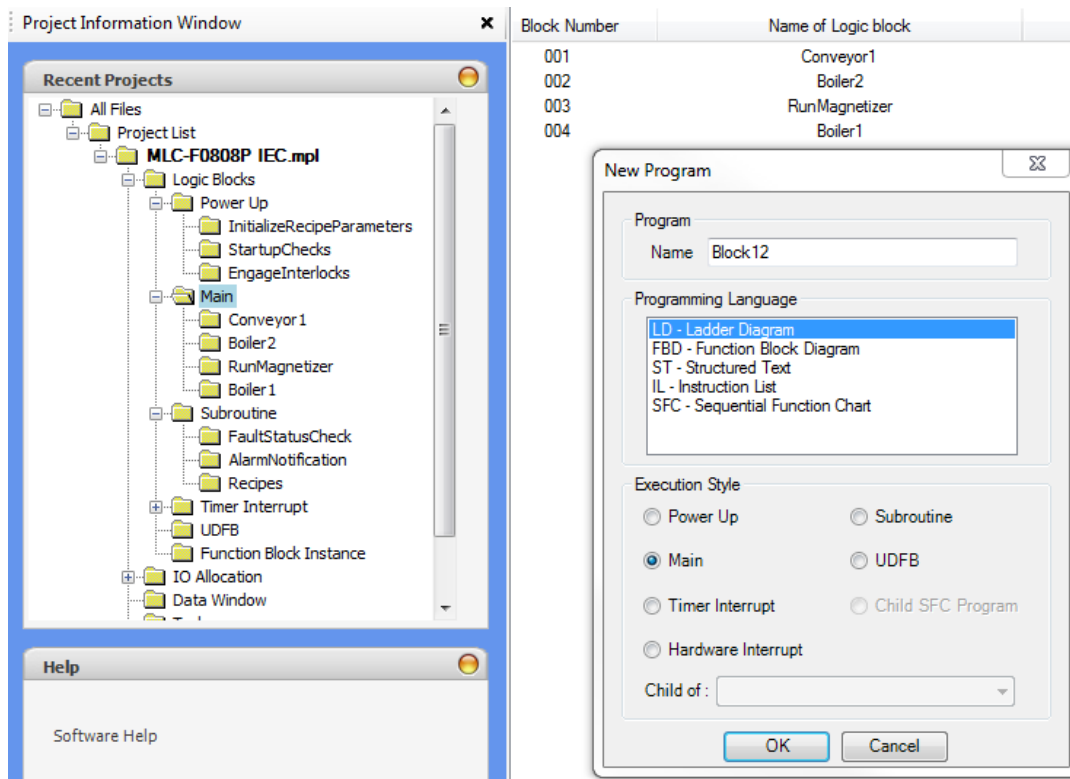
- **Name** – Enter the name of the new logic block.
  Note: With Native Ladder Logic selected, the name of a new logic block can include spaces and special characters such as the pound sign #. IEC mode limits the name to alphanumeric characters and the underscore character only.

- **Programming Language** – Select the type of programming language you wish to use for this logic block.
  ⭢ **Note:** See the *IEC 61131 Programming Guide for MAPware-7000* for more details on the different IEC programming languages and how to use them.
- **Execution Style** – Select which type of logic block is created. Note that the logic block type is already set according to which folder (Power Up, Main, etc.) was highlighted.
  ⭢ **Note:** For more information on the **Child SFC Program** option, see the **Sequential Function Chart** section in the *IEC61131 Programming Guide for MAPware-7000*.

When the **OK** button is clicked, a blank work area for the new logic block is displayed according to which programming language was selected.

Deleting a Logic Block

To delete a logic block, right-click the block that you wish to delete in the **Project Information Window**.
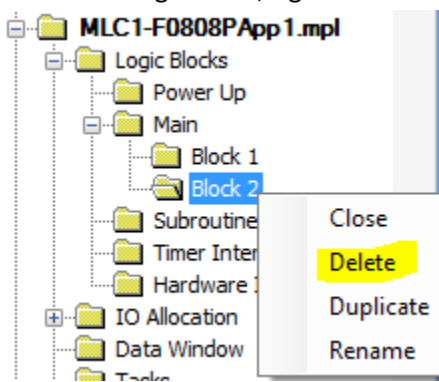
Or with the Logic Block highlighted (i.e. Block 2), select **Delete** from the **Block** menu. Another option is to simply click the Delete Block ❌ icon or right-click anywhere in the work area to display the context menu,
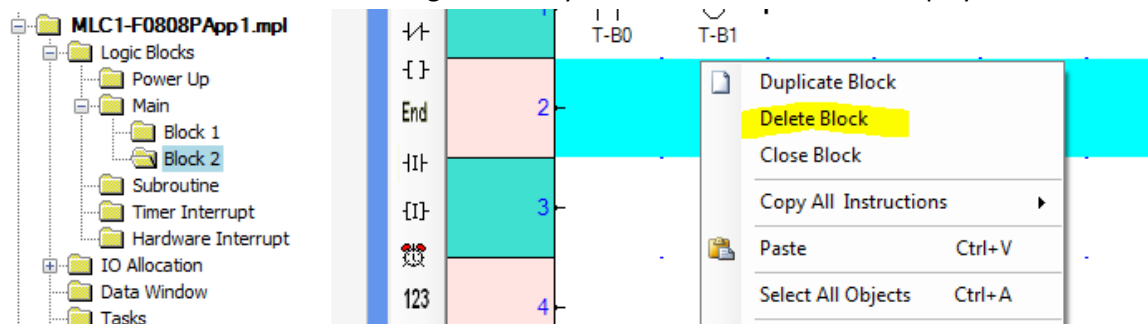
Renaming a Logic Block

To rename a logic block, right-click the block that you wish to rename in the **Project Information Window**. Select **Rename**.
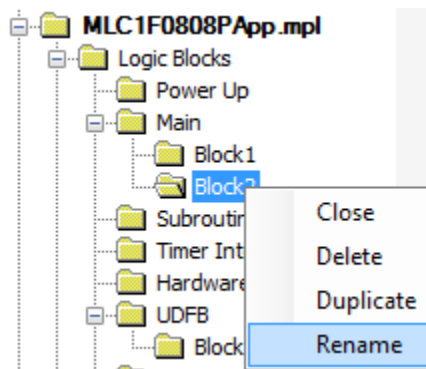
> ⬚ **Note:** All logic blocks must have a unique name.

Closing a Logic Block

Closing a logic block will remove it from the list of open windows for the work area. To close a logic block, right-click the block that you wish to close in the **Project Information Window**.
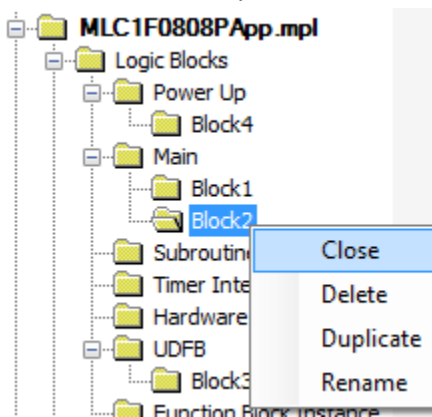
Or with the Logic Block highlighted (i.e. **Block2**), click **Close** from the **Block** menu.

Duplicating a Logic Block

You may have a logic block that you wish to copy so that you don't have to recreate the code.

To duplicate a logic block, right-click the block that you wish to duplicate in the **Project Information Window** and select **Duplicate**.
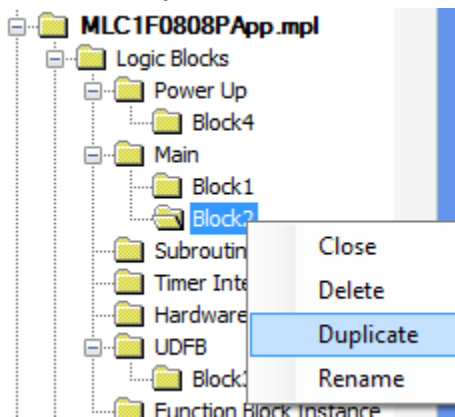
Or with the Logic Block highlighted (i.e. **Block2**), select **Duplicate** from the **Block** menu.

Creating a Ladder Logic Program

> ⬚ **Note:** The information below is just a brief introduction into creating a native ladder logic program using MAPware-7000. For a more detailed look, consult the *MAPware-7000 Ladder Logic Guide, Maple P/N: 1010-1041*.

Using Ladder Rungs

In native ladder logic programming, a ladder rung is simply a network of interconnected elements (i.e. coils, contacts, instructions) between a left and a right power rail. The left power rail represents active power and the right power rail is used to represent completing the electrical circuit to ground. Thus, native ladder logic is a pictorial representation of a physical electrical layout. There is no limit to the number of rungs that you can place into each logic block, except for the available ladder memory for the selected MLC model.

Inserting a Rung

To add elements to a rung in a logic block, simply highlight the rung you want to edit in the work area, then click one of the elements listed in the Quick Access menu or from any of the PLC Instruction menus (i.e. **I/O Instructions**, **Data Transfer**, **Math**, etc.).
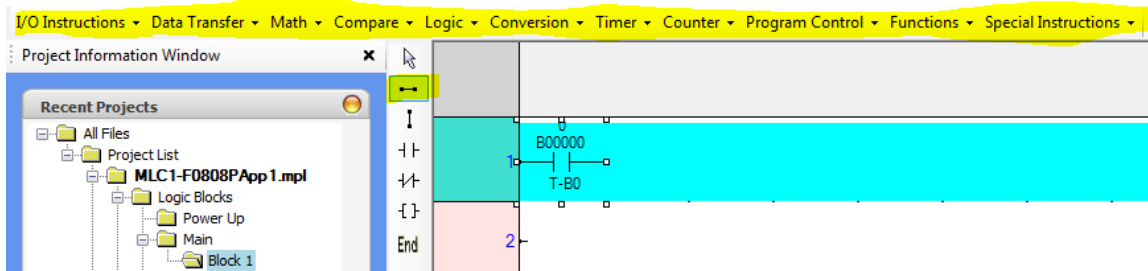
Move your mouse cursor (changes to a red cross-hair) to the location on the rung that you want to place the element, then click to place.
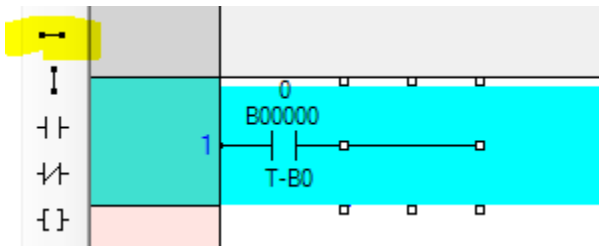
If you want to add or insert a new rung above an existing rung, right-click to display the context menu.
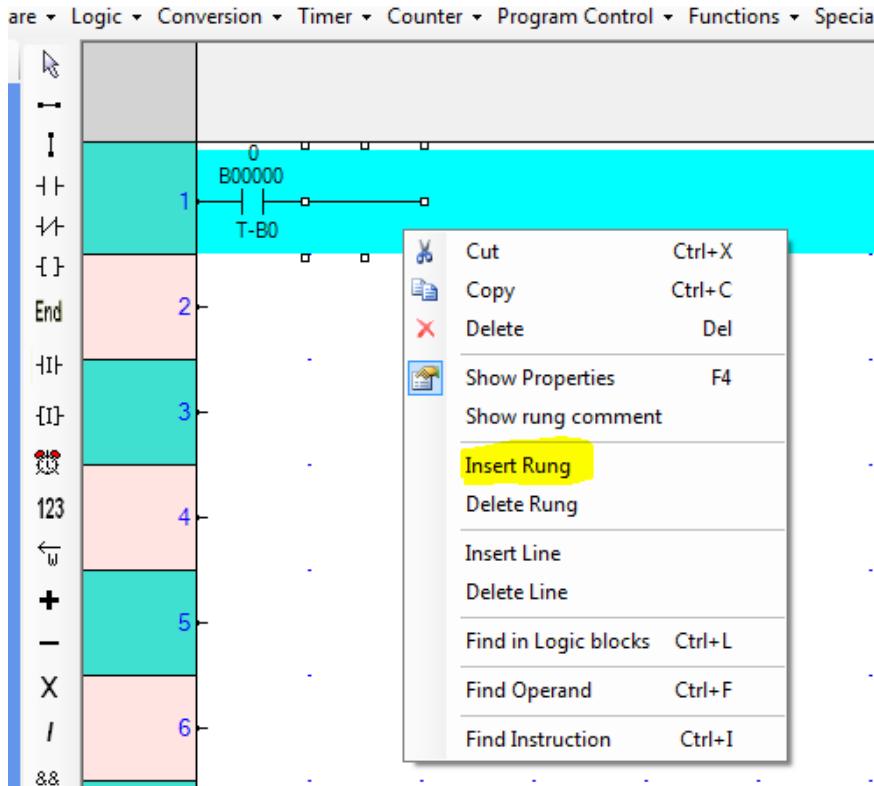
Click **Insert Rung** to create a new rung above the currently highlighted rung.
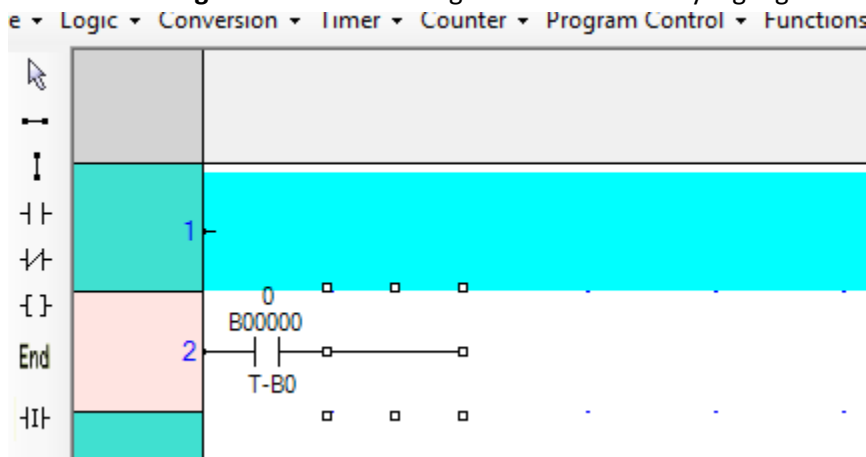
Deleting a Rung

To delete any elements of a rung in a logic block, simply highlight the element you want to delete in the work area, then click the [Delete] button on your keyboard or right-click to display the context menu.
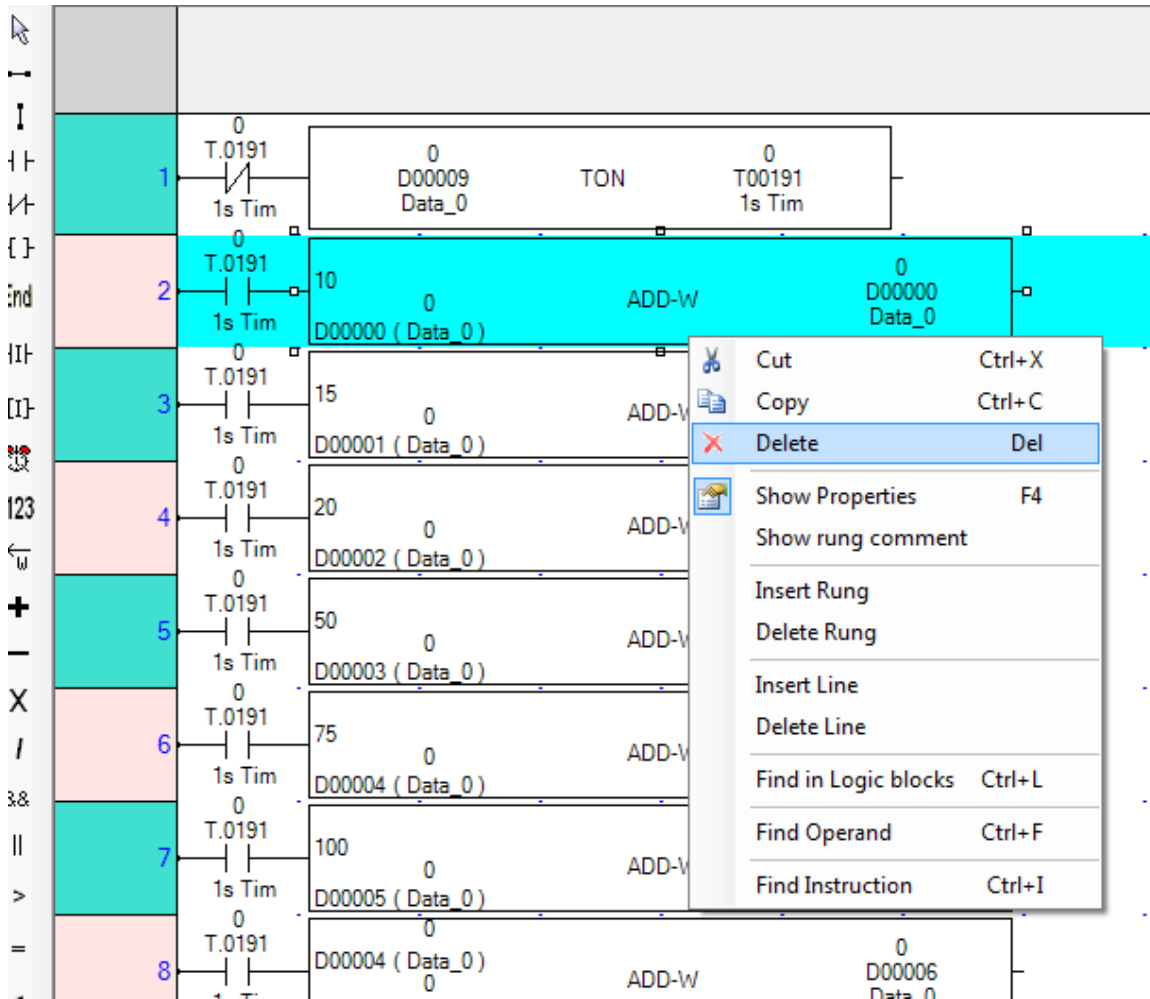
> 📝 **Note:** You can delete any element, including one that is connecting other elements of a ladder rung. If deleting an element creates a problem in your ladder logic, an error will be displayed in the **Errors and Warnings** window when the project is compiled.

If you want to remove an entire rung from a logic block, highlight the rung, then right-click (as above) to display the context menu and select **Delete Rung**.

Show Rung Comment

MAPware-7000 allows you to add notes or comments to each rung that you create to help what the logic is doing. To write a comment, you must first display the **Edit Rung Comment** field.
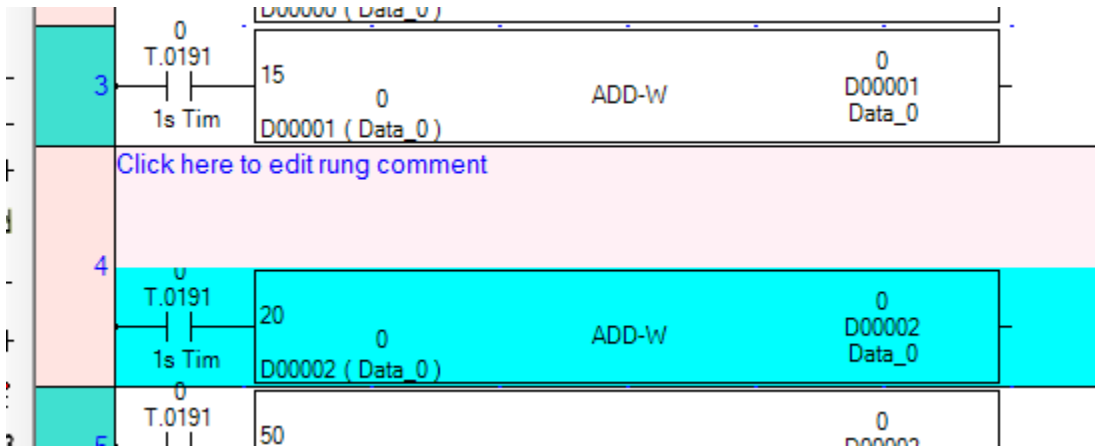
This is accomplished by right-clicking anywhere on the selected rung to display the context menu.
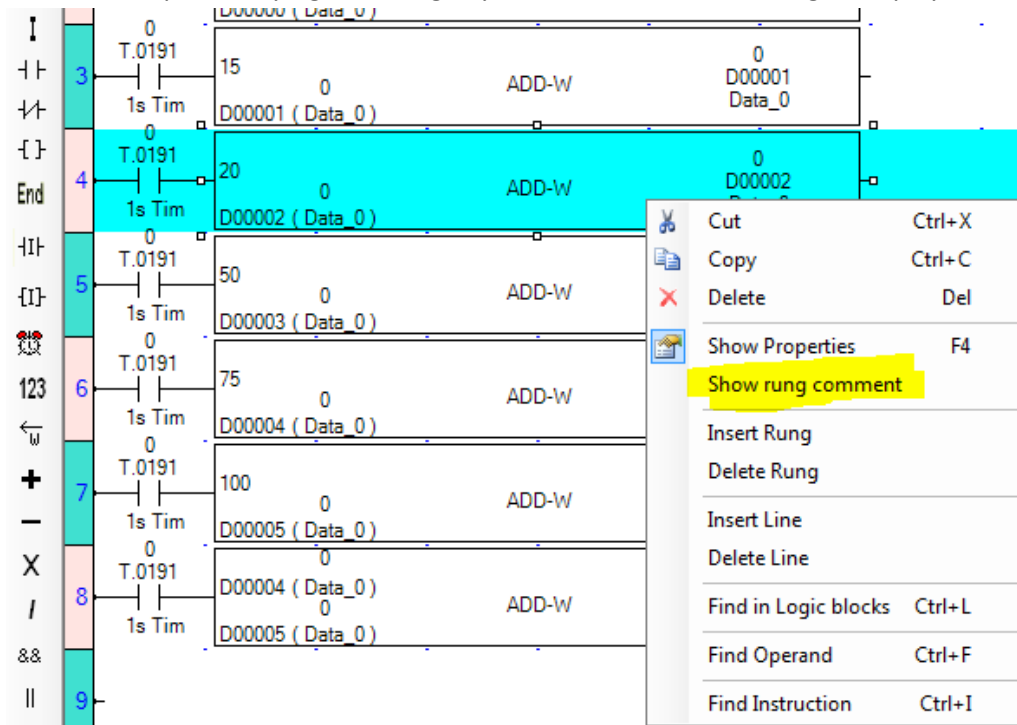
Select the **Show Rung Comment** option. Once displayed, you can add your comment by highlighting the default text in the comment field and replacing it with your own comment.
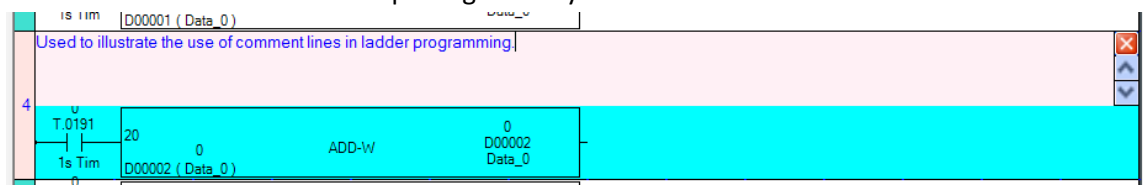
At any time, you can hide the comment line by simply clicking the Close ❌ button located on the right side of the work area (if you don't see this, move the horizontal scroll bar located on the bottom of the work area). Note that closing the Comment field does not delete or remove the comment – you can redisplay it at any time.

Lines vs. Rungs

A rung may have parallel connections or instructions that require more than one line of space:
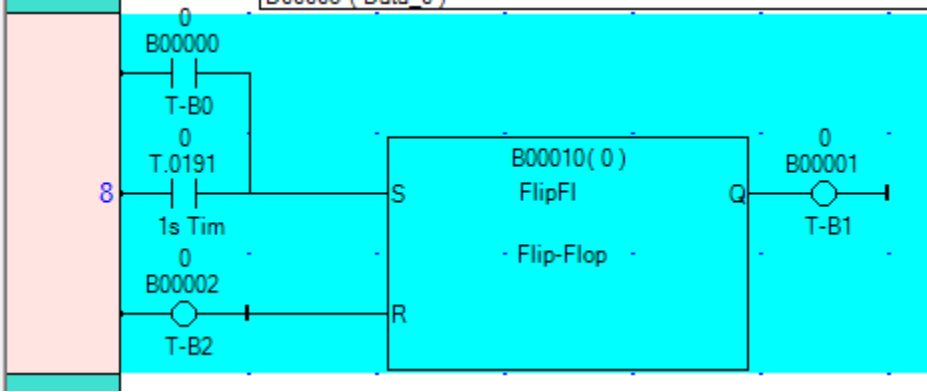
When inserting an element that requires more space, a new line is usually created to accommodate the new element. However, you can manually insert a new line to a rung.

Insert Line

To insert a new line to an existing rung, right-click to display the context menu.
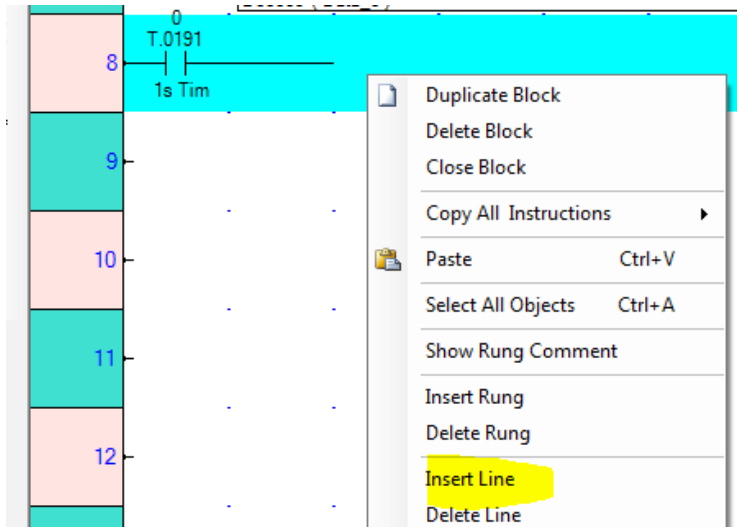
Select **Insert Line** to create a new line above the currently highlighted line of the rung.
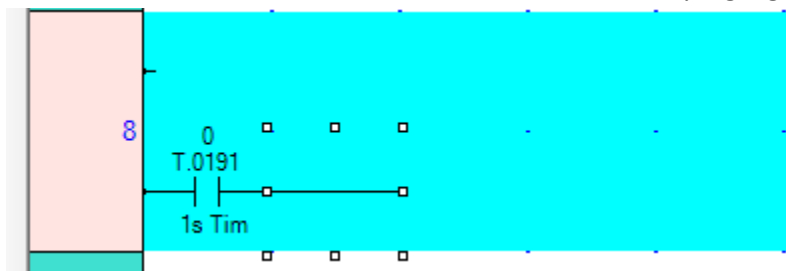
⊞ **Note**: You can add up to 50 lines per rung.

Delete Line

Similar to inserting a new line, deleting a line of a rung is accomplished by first highlighting the line, then right-clicking to display the context menu.
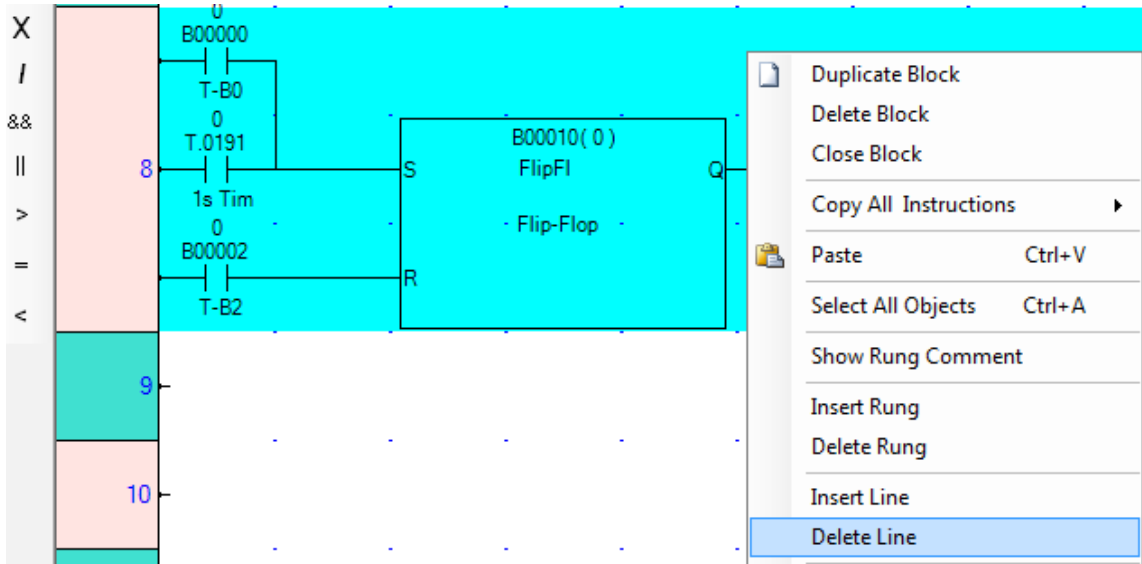
The selected line and all elements on that line are deleted.
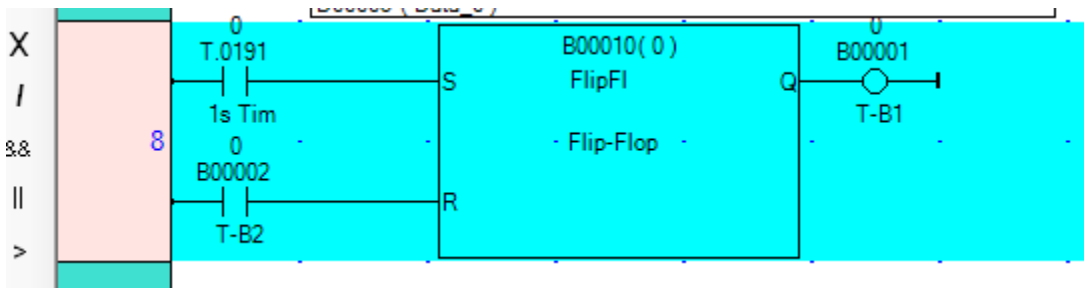
Using Instructions

The following section covers the basics on selecting and editing ladder logic instructions. For a complete list of instructions, consult the _MAPware-7000 Ladder Logic Guide, Maple P/N 1010-1041_ manual or the MAPware-7000 Help files.

What are Instructions?

In MAPware-7000, instructions refer to the various elements (i.e. contacts, coils, functions, etc.) that can be added to a rung of logic. Some instructions are fairly simple, requiring only one input or output, such as a normally open contact. Other instructions are more complex, performing some function (i.e. add, multiply, converting from ASCII to Hex, running a timer, etc.) that may require several inputs/outputs.

Inserting an instruction

To insert an instruction into a ladder rung, highlight the rung you want to edit in the work area, then click one of the elements listed in the Quick Access menu or from any of the PLC Instruction menus (i.e. **I/O Instructions**,

Data Transfer, **Math**, etc.).
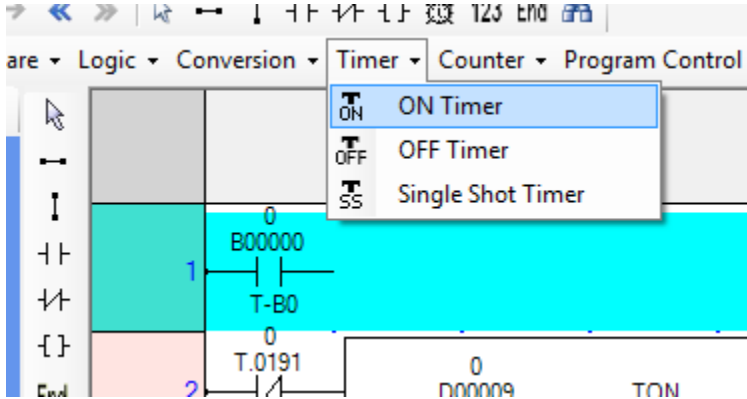
Move your mouse cursor (changes to a red cross-hair) to the location on the rung that you want to place the element and click to place.
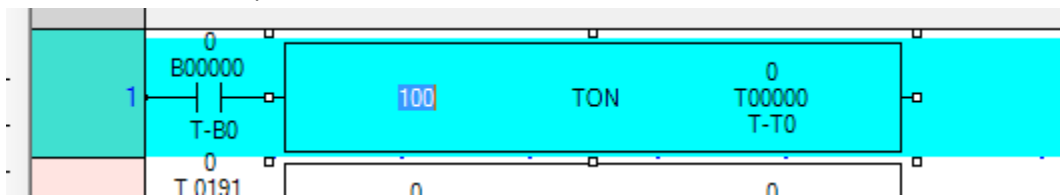
Deleting an instruction

To delete any instructions of a rung in a logic block, simply highlight the instruction you want to delete in the work area, then click the [Delete] button on your keyboard or right-click to display these options.
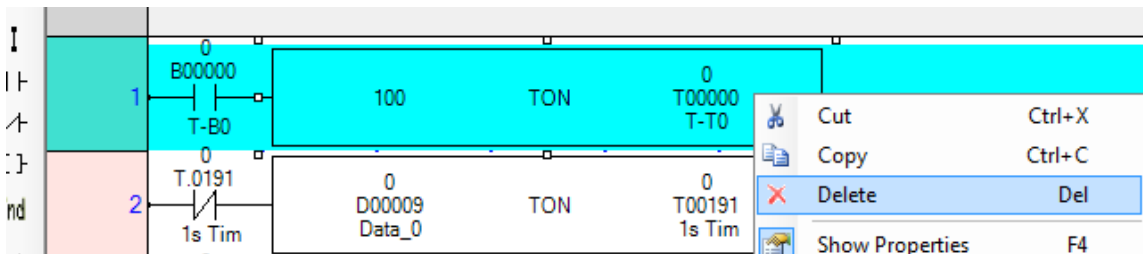
> ✒ **Note:** You can delete any instruction, including one that is connecting other instructions of a ladder rung. If deleting an instruction creates a problem in your ladder logic, an error will be displayed in the Errors and Warning window when the project is compiled.

Copying instructions

To copy any instructions of a rung in a logic block, highlight the instruction you want to copy in the work area, then right-click to display the context menu.
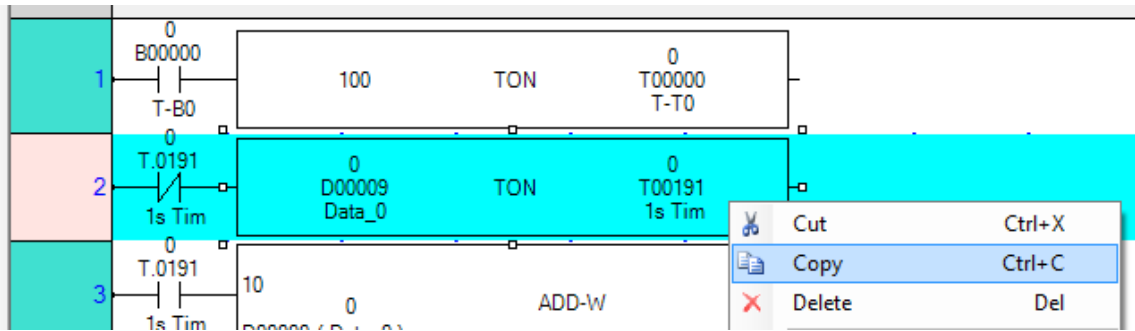
After copying the instruction, click the rung that you want to paste the instruction to, then right-click again, and select **Paste**.
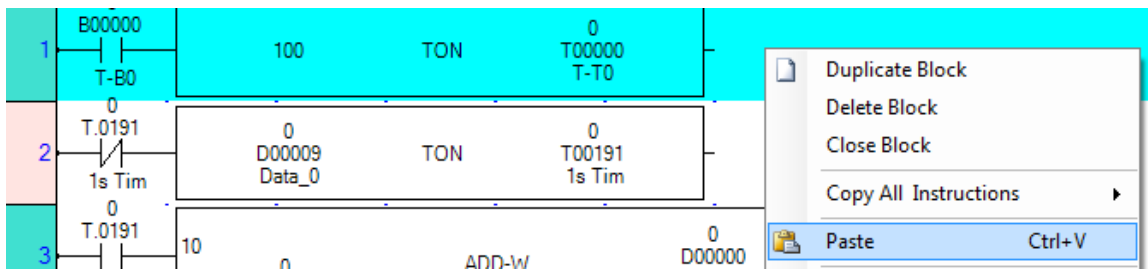
> 🖅 **Note:** Although both the **Copy** and **Paste** commands have shortcut keys, they will not work when copying/pasting instructions.

Moving Instructions

There may be an occasion in which you wish to move an existing instruction (to connect to the ladder rung).
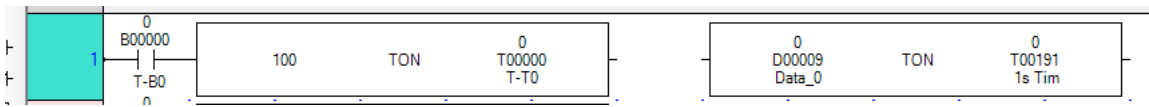
To move the instruction to a new location, click the instruction once. A perimeter of small white boxes will appear around the instruction (to indicate focus) and the mouse cursor changes to a crosshair ✥ symbol. Click and drag to move the instruction to a new location.

Editing Instructions

Most instructions require some editing to configure the parameters that are used for the instruction. This is done using the **Instruction Properties** window that displays on the right side of the work area.

Instruction Properties

The **Instruction Properties** window displays pertinent information about the highlighted instruction. In addition to the name of the instruction, a brief description is included, as well as execution times for the particular model selected.

If any input or output parameters are required for the selected instruction, you can edit these by clicking the parameters for each operand. In this example of using the **AND (Word)** instruction, there are two inputs (**Operand A** and **B**) and one output (**Operand C**) required. The particular parameters required depend upon the instruction used. In this example, the three operands use the following parameters:

- **Type**: Option to select a tag (**Register**) or a number (**Constant**)
- **Address**: If **Register** is selected as the **Type**, this field is used to select the specific tag register by referencing the tag address.
- **Comment**: Can be used by the programmer to enter text that better describes the operand.
- **Name**: Another method of entering the specific tag register. Instead of selecting the tag address, you can select the name of the tag.
- **Tag Index Type**: Option to use one of the indexes (I, J, or K) as a pointer to another tag address.

Finding Instructions

MAPware-7000 includes three search features that may help you to locate a segment of code. These are particularly helpful when multiple logic blocks with many rungs of code have been developed.

To use these search features, right-click anywhere on the work area.
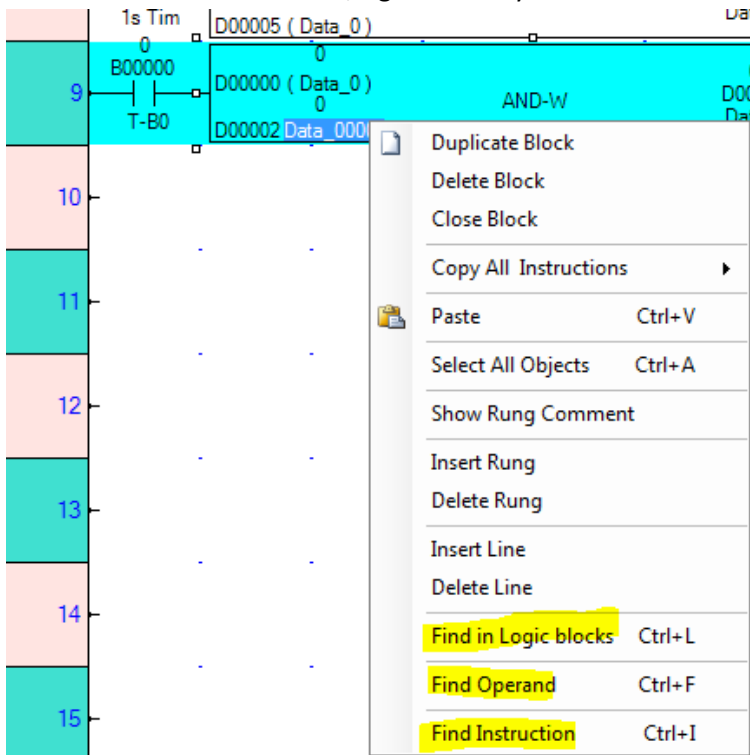
Find in Logic blocks

Use this feature to search all logic blocks for a particular string of characters. The string of characters could be a tag address or a tag name.

Find Result

Block 1:Addition:(Rung=6, Line=1)
Block 1:Addition:(Rung=8, Line=1)
Block 1:AND:(Rung=9, Line=1)
Block 2:Addition:(Rung=1, Line=1)
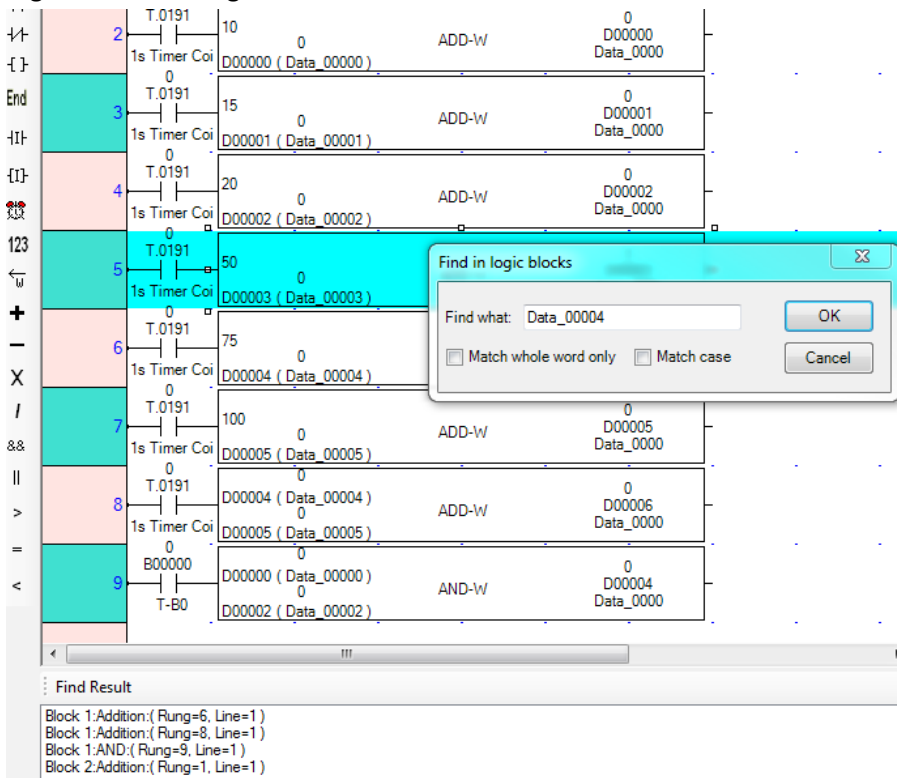
Find Operand

Use this feature to find a particular tag address from a list of tag addresses that are used in the selected logic block. You also have the option to replace the selected tag address with a new one.
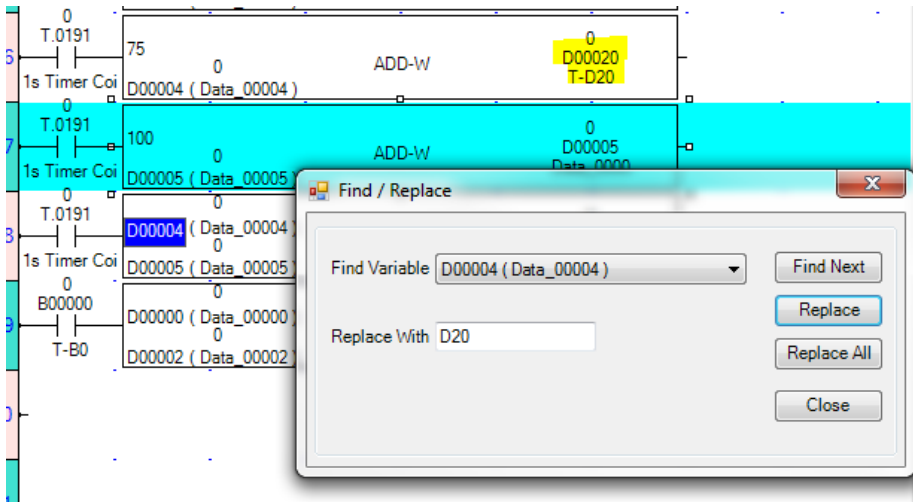
Find Instruction

The **Find Instruction** feature is available if you know the Instruction that you want to find in the selected logic block and the particular tag address used in that instruction.
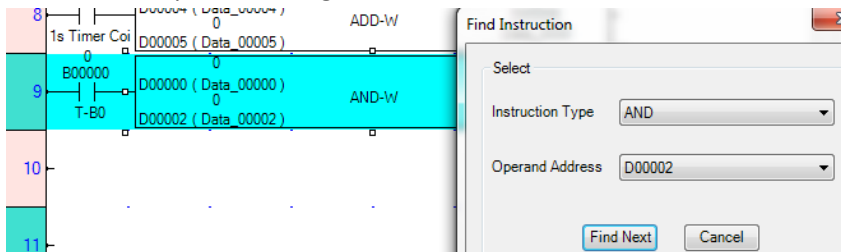
Creating an IEC 61131-3 program

> ⌨ **Note:** The information below is just a brief introduction into creating an IEC 61131-3 logic program using MAPware-7000. For a more detailed look, consult the _IEC 61131-3 Programming Guide for MAPware-7000, Maple P/N: 1010-1045_.

IEC 61131-3 Defined

IEC 61131-3 is a section of an International Electro-Technical Committee (IEC) standard that provides a definition for implementing PLC programming software. The standard was first introduced in 1993 as the result of an effort to standardize the myriad PLC logic editors in the automation market place. The goal of the standard is to give automation professionals a familiar environment and set of tools to create PLC programs across vendor platforms. MAPware-7000 has editors implemented for all five programming languages defined by the standard:

- Ladder Diagram (LD) – Graphical language that simulates an electrical circuit; program instructions are attached as discrete elements in the circuit and are executed when "energized".
- Function Block Diagram (FBD) – Graphical language based on logic diagrams. Functions are represented by blocks; complex operations can be built by interconnecting function blocks.

- Structured Text (ST) – Text based programming language. Programs are built using keywords, operators and function calls.
- Instruction List (IL) – Text based procedural programming language.
- Sequential Function Chart (SFC) – Graphic programming language in which program execution is modeled as a flow chart. Programs are developed by adding blocks to the flow chart.

The Editor Window

Let's take a quick tour of some of the aspects of MAPware-7000 that are unique to an IEC 61131-3 project.

Projects created using IEC 61131-3 programming mode contain a **Project Information Window** that allows easy navigation to various parts of the project. When a new project is created it contains a default logic block named **Block1**.
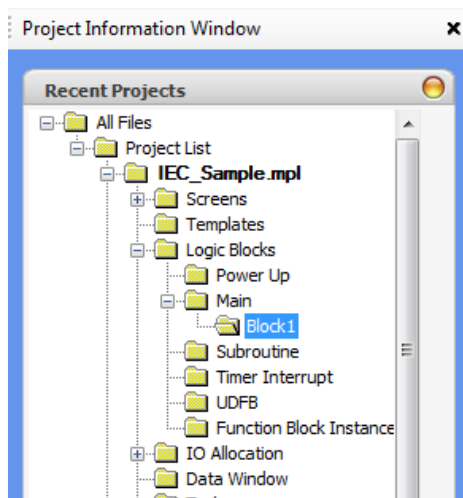
Clicking the folder for **Block1** in the **Project Information Window** displays the editor for that block. The default logic block uses the Ladder Diagram editor, so the Ladder Diagram editor is displayed when the **Block1** folder is

selected in the Project Tree:
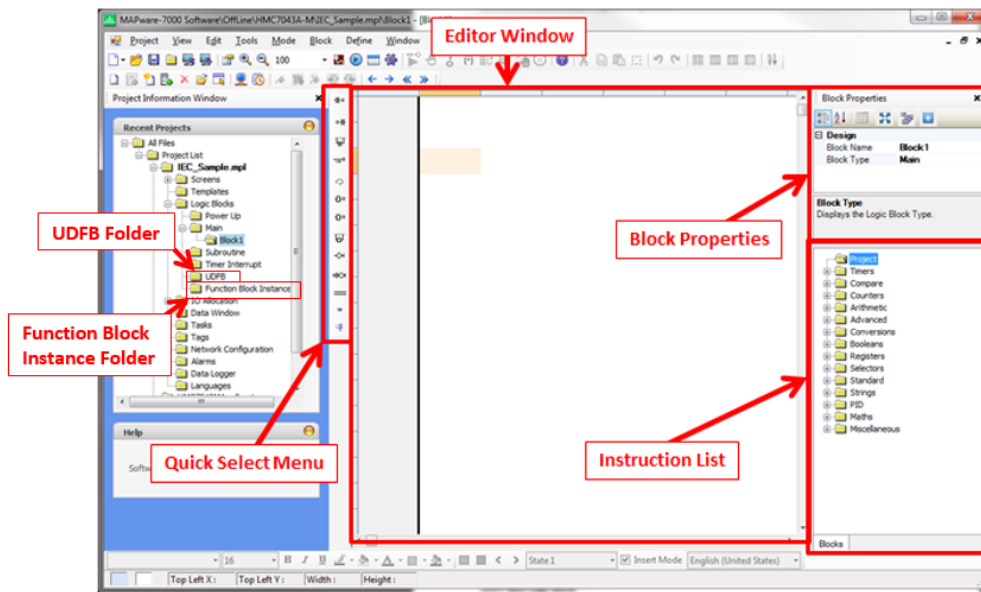
Elements that are common to all of the IEC languages are:

- **Block Properties** – This is where you can change the Name or the Execution Type of the block.
- **Instruction List** – Lists all of the available instructions for the project. Instructions are categorized according to functionality. Expand the node for a given category to see individual instructions. To select a particular instruction simply click and drag it into the editor window. Subroutines and UDFBs (Function Blocks that users create) will appear under the **Project** folder of the instruction list once they are defined. Information on how to configure and use a given function block is available in the help file. To access the help file entry for any of the blocks in the instruction list simply double-click the instruction.
- **Editor Window** – This is where the logic is defined. It displays the graphical or text representation of the logic program.
- **Quick Select Menu** – The options available on this menu depend on the editor in use but provide quick access to common program elements. Click the location in the editor where you want to place the element then click the element in the quick select menu to place it in the editor.
- **UDFB Folder** – This folder contains the definitions for User Defined Function Blocks. To add or edit a UDFB, click the block name in this folder. To use a UDFB in another logic block, select it from the **Project** folder of the **Instruction List**.

Creating Tags for IEC logic

All of the logic in the various logic blocks operates on tags. Tags are linked to Graphic Objects displayed on the HMC, where the operator can interact with them using the touchscreen, or to the HMC's IO modules, or some other IO mechanism. Tags are defined and listed in the **Tag Database** which can be opened by clicking the **Tags** folder of the project tree:

Tags are handled somewhat differently in IEC 61131-3 projects vs. Native Ladder projects. In IEC mode projects tags are not assigned an explicit address. Instead they are given name and a type, MAPware-7000 is responsible for allocating and tracking a memory address for the tag. For more information on creating and using tags refer to the MAPware-7000 programming manual. The following table lists the data types available in IEC 61131-3 programming mode:

Data Type	Description	Bits	Value Range
BOOL	Boolean value can have one of two values: 1 (ON / True) or 0 (OFF / False)	1	0 to 1
BYTE	Unsigned Short (same as USINT)	8	0 to 255
DINT	Signed Double Integer	32	-2,147,483,648 to 2,147,483,647
DWORD	Unsigned Double Integer (same as UDINT)	32	0 to 4,294,967,295
INT	Signed Integer	16	-32,768 to 32,767
REAL	IEEE 754 format, single precision floating point number	32	N/A
SINT	Signed Short Integer	8	-128 to 127
STRING	Variable Length string of ASCII characters	1 to 255 bytes	N/A
TIME	Time of Day	32	N/A
UDINT	Unsigned Double Integer (same as DWORD)	32	0 to 4,294,967,295
UINT	Unsigned Integer	16	0 to 65,535
USINT	Unsigned Short Integer (Same as BYTE)	8	0 to 255
WORD	Unsigned Integer (Same as UINT)	16	0 to 65,535

Chapter 7 – Tasks

An integral part of the MAPware-7000 software is the ability to create tasks. Tasks are specifically predefined actions taken by the controller such as writing a value to a tag register, turning a bit on/off, etc.

The number of tasks is limited only by the total amount of memory available in the controller. Each task has two fundamental components: the action taken when the task activates, and the triggering mechanism that starts the action.

Tasks are assigned in the **Tasks** folder in the **Project Information Window**.
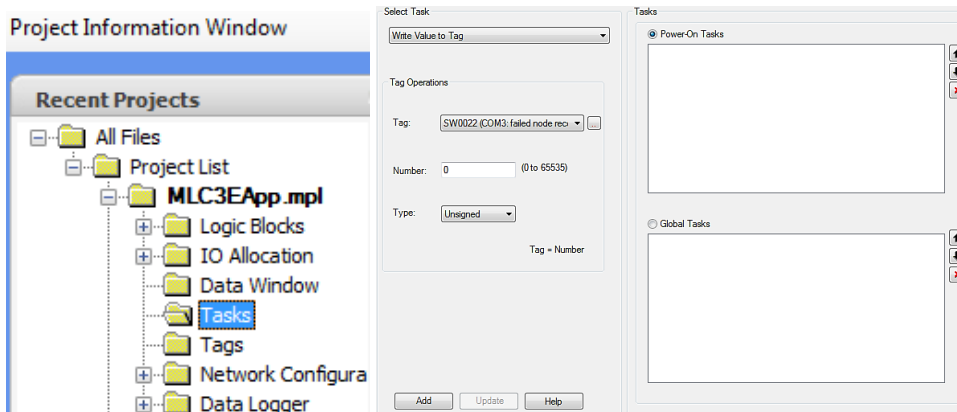
Two execution styles are available:

- **Power-on Tasks**: Tasks that execute once during initial power-up of the MLC.
- **Global Tasks**: These tasks execute continuously while MLC Series is on.

The table below provides a list of all tasks available in MLC projects and a brief description.

Task Action	Description
Add a Constant Value to a Tag	Add a constant value to the current value residing in an internal tag or PLC tag
Add Tag B to Tag A	Add a value residing in the Tag B register to the current value residing in the Tag A register
Copy PLC Block to External Device Block	Copies the values in a block of continuous registers beginning with the Tag B register to the block of registers designated with the Tag A register
Copy External Device Block to PLC Block	Copies the values in a block of continuous registers beginning with the Tag B register to the block of registers designated with the Tag A register
Copy Tag B to Tag A	Copies a value residing in the Tag B register to the target Tag A register
Swap Tag A and Tag B	Swaps (exchanges) the values residing in the Tag A register and the Tag B register
Subtract a Constant Value from a Tag	Subtract a constant value from the current value residing in an internal tag or PLC tag
Subtract Tag B from Tag A	Subtract a value residing in the Tag B register to the current value residing in the Tag A register
Toggle Bit	Toggles (turn off/on) an internal tag or PLC tag coil/bit
Turn Bit ON	Set (turn on) an internal tag or PLC tag coil/bit
Turn Bit OFF	Clear (turn off) an internal tag or PLC tag coil/bit
Write Value to Tag	Write a constant value to an internal tag or PLC tag

Add a Constant Value to a Tag

Description: Add a constant value to the current value residing in an internal tag or PLC tag.

Parameters:

- Tag – Target address
- Number – Constant value added to the target tag when the task executes.
- Type – Format (Unsigned, Signed, Hexadecimal, BCD, Binary, or Float) of data.

Add Tag B to Tag A

Description: Add a value residing in the Tag B register to the current value residing in the Tag A register.

Parameters:

- Tag A: Destination address
- Tag B: Source address

Copy PLC Block to External Device Block

Description: Copies the values in a block of continuous registers beginning with the Tag B register to the block of registers designated with the Tag A register.

Parameters:

- Tag A: Destination location of values. Can be either internal memory or a block of PLC registers
- Tag B: Source location of values. Must be internal memory
- Number of words: This is the total number of consecutive registers copied.
- Enable Control: Allows tags to control this task.
  Note: This option is only available if configuring a **Global** task. Only PLC tags are used.
    o   Enable: If ON, allows task to be executed
    o   Done: MLC Series sets this coil after completion of task
    o   Error: MLC Series sets this coil if error performing task

Copy External Device Block to PLC Block

Description: Copies the values in a block of continuous registers beginning with the Tag B register to the block of registers designated with the Tag A register.

Parameters:

- Tag A: Destination location of values. Must be internal memory
- Tag B: Source location of values. Can be either internal memory or a block of PLC registers
- Number of words: This is the total number of consecutive registers copied.
- Enable Control: Allows tags to control this task.
  Note: This option is only available if configuring a **Global** task. Only PLC tags are used.

- o   Enable: If ON, allows task to be executed
- o   Done: MLC Series sets this coil after completion of task
- o   Error: MLC Series sets this coil if error performing task

Copy Tag B to Tag A

Description: Copies a value residing in the Tag B register to the target Tag A register.

Parameters:

- Tag A: This is the destination address
- Tag B: This is the source address
- Enable Control: Allows tags to control this task.
  - **Note:** This option is only available if configuring a **Global** task. Only PLC tags are used.
    - o   Enable: If ON, allows task to be executed
    - o   Done: MLC Series sets this coil after completion of task
    - o   Error: MLC Series sets this coil if error performing task

Swap Tag A and Tag B

Description: Swaps (exchanges) the values residing in the Tag A register and the Tag B register.

Parameters:

- Tag A: Target location for Tag B value
- Tag B: Target location for Tag A value

Subtract a Constant Value from a Tag

Description: Subtract a constant value from the current value residing in an internal tag or PLC tag.

Parameters:

- Tag: This is the target address
- Number: Constant value to subtract from the target tag when the task is executed.
- Type: Format (Unsigned, Signed, Hexadecimal, BCD, Binary, or Float) of data

Subtract Tag B from Tag A

Description: Subtract a value residing in the Tag B register to the current value residing in the Tag A register.

Parameters:

- Tag A: This is the destination address
- Tag B: This is the source address

Toggle Bit

Description: Toggles (turn off/on) an internal tag or PLC tag coil/bit.

Parameters:

- Tag: This is the target address

Turn Bit ON

Description: Set (turn on) an internal tag or PLC tag coil/bit.

Parameters:

- Tag: This is the target address
- Enable Control: Allows tags to control this task.
  - **Note:** This option is only available if configuring a **Global** task. Only PLC tags are used.
    - o Enable: If ON, allows task to be executed
    - o Done: MLC Series sets this coil after completion of task
    - o Error: MLC Series sets this coil if error performing task

Turn Bit OFF

Description: Clear (turn off) an internal tag or PLC tag coil/bit.

Parameters:

- Tag: This is the target address
- Enable Control: Allows tags to control this task.
  - **Note:** This option is only available if configuring a **Global** task. Only PLC tags are used.
    - o Enable: If ON, allows task to be executed
    - o Done: MLC Series sets this coil after completion of task
    - o Error: MLC Series sets this coil if error performing task

Write Value to Tag

Description: Write a constant value to an internal tag or PLC tag.

Parameters:

- Tag: This is the target address
- Number: Constant value placed into the target tag when the task executes.
- Type: Format (Unsigned, Signed, Hexadecimal, BCD, Binary, or Float) of data
- Enable Control: Allows tags to control this task.
  - **Note:** This option is only available if configuring a **Global** task. Only PLC tags are used.
    - o Enable: If ON, allows task to be executed
    - o Done: MLC Series sets this coil after completion of task
    - o Error: MLC Series sets this coil if error performing task

Chapter 8 – Miscellaneous Features

This chapter reviews some features available in the MAPware-7000 software and the MLC Series that are important but do not easily fit in other chapters of this manual.

The features described include:

- Data Logger
- FTP
- Application Memory Status
- Keep Memory Area
- Real Time Clock
- Internal Battery

Data Logger [MLC2-E and MLC3-E Only]

The Data Logger is used to capture values in specified data registers. The data is stored in the internal memory of the MLC3 PLC and external memory for the MLC2 PLC (a separate micro SD card must be installed in the MLC2 for data logging to function). Once data is collected, it can be transferred from the MLC to a computer for analysis. Although the Data Logger feature is available for all HMC models, only the MLC2 and MLC3 from the MLC line support data logging – if you are using a MLC1 model, the Data Logger feature is not available.

To configure the Data Logger, in the **Project Information Window**, click the **Data Logger** folder, then for the MLC3 click the **Internal Memory** folder, or for the MLC2 click the **External Memory** folder.
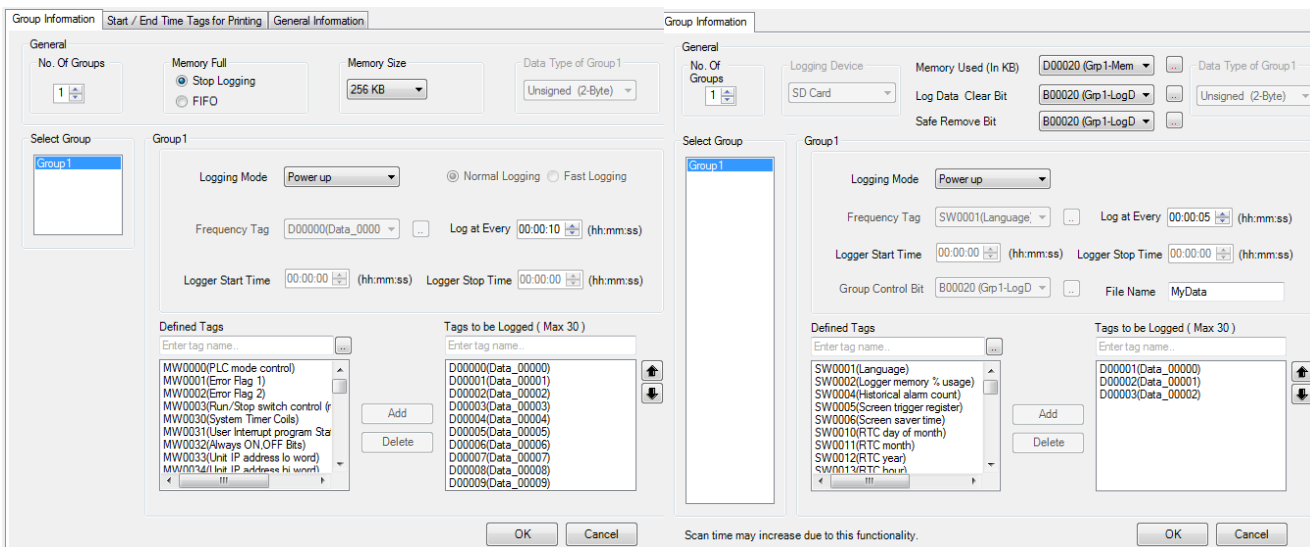
MLC3 Internal Memory Data Logging   MLC2 External Memory Data Logging

Group Information Tab

- **No. Of Groups** – Data Logging is configured according to 'groups'. A group is a collection of registers that the MLC Series polls to record data according to the 'logging mode' selected. Each group can have a maximum of 30 registers. Up to four groups can be created for Internal Memory Data Logging, and up to

10 groups can be created for External Memory Data Logging. Use this box to enter how many data groups you need.

⌨ **Note:** You must enter at least one group in this field to begin creating a data log.

- **Memory Full (Internal) –** Determines how the MLC captures data once the memory storage area is full.
  - ○ **Stop Logging –** Data captures stop once the memory reserved has been used. You must clear the memory by toggling the *Logged Data Clear Bit (S43)*. Or you can clear the memory by enabling the **Erase Data Logger Memory** option under the **Settings** tab in the **Project Configuration** dialog box (see **Project > Properties**) and downloading a project into the MLC Series.
  - ○ **FIFO –** Data logging continues but the oldest data capture is deleted from memory when a new data capture is added.
- **Memory Size (Internal) –** Options are 256KB to 20MB. This is the amount of memory allocated for all of the groups combined. Select the most appropriate setting for your needs but remember that any memory allocated to the data logger reduces the amount of memory available for the rest of your project (click **Tools > Application Memory Status** for a breakdown of memory usage).
- **Logging Device (External) –** This is the physical location of the external storage device attached to the MLC. For now, the only option is SD card.
- **Memory Used (in KB) (External) –** Select a tag register that will be used by the Data Logger to record the total amount of memory used. Note that the tag register must be a 4-byte register. Click the directory button to display the **Select Tag** window.
- **Log Data Clear Bit (External) –** Select a tag that will be used by the Data Logger to clear the Data Log memory, whenever the bit is set. Click the directory button to display the **Select Tag** window.
- **Safe Remove Bit (External) –** Select a tag that will be used by the Data Logger to indicate when it is safe to remove the external storage device. Indicates that the logger is not actively writing to the SD card. Click the directory button to display the **Select Tag** window.
- **Data Type –** Options are Unsigned (2-Byte), Unsigned (4-Byte), Signed (2-Byte), Signed (4-Byte), and Float (4-Byte). This option applies to each individual data group created and determines how the data logger will interpret the data collected. Only tags that are configured with the appropriate byte length (i.e. 2 bytes {1 word} for Signed/unsigned {2-Byte}, and 4 bytes {2 words} for signed/unsigned/float {4-Byte}) in the **Tag Database** will show in the Defined Tags box for the selected group.

⌨ **Note 1 –** Settings above apply to all groups

⌨ **Note 2 –** Any change to the settings above requires that all previously logged data in the MLC Series be erased.

Configuring Each Data Group

Select Group – Once you have selected how many data groups you need in the No. of Groups box, the data groups will be listed (ex. Group1, Group2, etc). Click each Group to edit that group's attributes.

Logging Mode – This setting determines how and when the MLC collects data. Each group can have a different logging mode. There are four options for the logging mode:
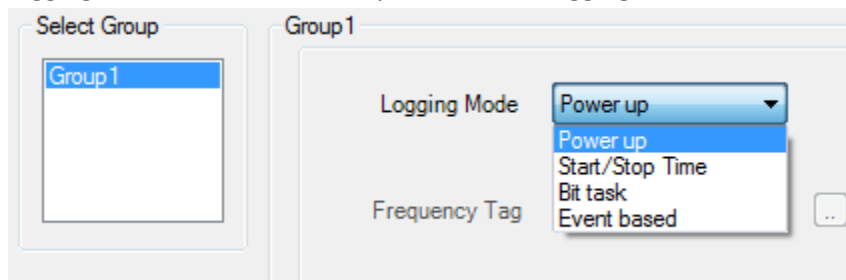
- **Power up** – Collects data immediately after the MLC has initialized, according to the time value entered into **Log at Every** field. Therefore, if the value entered into this field is 00:00:10, then the MLC records the data every ten seconds after initial power-up.
- **Start/Stop Time** – This option is used if you wish to collect data during a specific time period every day. The starting time is entered in the **Logger Start Time** field and end time in the **Logger Stop Time** field. Between these time periods, the MLC collects data continuously according to the frequency specified in the **Log at Every** field.
- **Bit Task (Internal)** – The Start and Stop time is controlled by one of four predefined system bits (S0028-S0031) located in the **Tag Database**. The frequency of data collection is specified using the **Log at Every** field.
- **Bit Task (External)** – The Start and Stop time is controlled by the **Group Control Bit**. The frequency of data collection is specified using the **Log at Every** field.
- **Event Based** – Data logging is controlled using a **Logging bit**, which is any MLC internal memory bit (external PLC bits not allowed) configured in the tag database. The data is captured based upon the logging bit changing state according to one of the following three options:
  - **Positive Edge** – Transition from 0 to 1
  - **Negative Edge** – Transition from 1 to 0
  - **Both Edges**

  Event Based also has an option of selecting **Normal Logging** or **Fast Logging** mode.
  - **Normal Logging** –Data is captured and stored as HH:MM:SS in a CSV file.
  - **Fast Logging** – The time in milliseconds (HH:MM:SS:mS) is also captured.

File Name (External) – This is the file name assigned to the *.csv file generated when logging data to an external SD card.
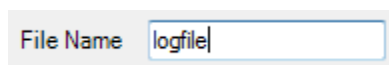
Frequency Tag (Logging with Run Time Frequency) – This option is not available in MLC models.

Log at Every – This is the time interval or frequency of data capture used when **Power Up**, **Start/Stop Time**, or **Bit Task** is selected as the **Logging Mode**. For Internal Memory logging, max speed is 1 second. For External Memory logging, max speed is 5 seconds.
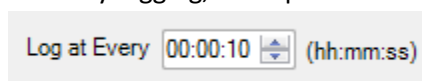
Logger Start Time – This is the set time at which the Data Logger starts capturing data when **Start/Stop mode** is selected as the **Logging Mode**. The frequency is based upon the time interval entered for the **Log at Every** setting.

Logger Start Time   00:00:00 ⇕   (hh:mm:ss)

Logger Stop Time – Same as **Start Time** above, except this is when the Data Logger stops capturing the data.

Logger Stop Time 00:00:00 ⇕ (hh:mm:ss)

Defined Tags

This list displays all of the tags available in the **Tag Database** which are the same size (ex 2 Byte Int) as the **Data Type** selected in the **General** section. Scroll through the tags until you find the one(s) you wish to use. Click the tag to highlight it, then click the **Add** button to move the selected tag over to the **Tags to be Logged** list.

The **Add** and **Delete** buttons can add/remove highlighted tags from the selected group. A maximum 30 tags per group is allowed.

After you have finished selecting which tags to be logged, you must click the **OK** button to save tags to a group.

Start/End Time Tags for Printing Tab (Internal Only)

This tab is used to configure the times at which collected data is sent to a serial printer that is connected to an HMC Series HMI + PLC through the serial port. This feature is not available for the MLC Series.

General Information Tab (Internal Only)

This tab provides quick information and rules that must be adhered to when using the Data Logger feature:

1. Any recorded data that is stored inside the MLC Series as a result of using the Data Logger feature will be erased if:
    a. A change is made to a logged tag (i.e. deleted or added) in a group or the total number of groups used for data logging has changed.
    b. A change has been made to the general parameters (i.e. memory size, data type) of the data logger configuration.
2. The 'Firmware' Download Option must be checked when downloading the project to the MLC Series if:
    a. The Data Logger feature has been deleted.

      b.   Any Data Logger settings are changed.

3.   The Event-Based Logging Mode has two options for logging data:

      a.   Normal Logging: The time when data is captured is recorded in HH:MM:SS format in a CSV file.

      b.   Fast Logging: The time is recorded using HH:MM:SS:mS format (so milliseconds is captured).

Using System (S, SW) Memory Tags-

The following internal system memory tags are useful when using the Data Logger feature:

Register/Coil	Tag Name	Bytes	R/W	Description
S00028 thru S00031	DataLog Group # Log Control	Bit	Read/Write	Use these bits to Start/Stop data logging for the specified group when using Logging Mode: Bit Task. 0= Stop Logging, 1= Start Logging
S00043	Logged Data Clear Bit	Bit	Read/Write	When set, the data captured and stored in internal MLC memory is cleared.
SW0002	Logger Memory % Usage	16-bit Register	Read Only	This register shows a percentage of the allocated Data Log memory that has already been used by the Data Logger
SW0003_00	Logger Memory Full Status	Bit 0 of 16-bit Register	Read Only	This bit indicates when the Data Logger memory is full and has stopped logging.
SW0003_01	Logger Memory Clear Status	Bit 1 of 16-bit Register	Read Only	This bit indicates when the MLC is clearing the Data Logger memory. When this bit is 1, clearing memory is in process. When this bit is 0, process is completed.

Retrieving Data Log files from the MLC3

With the MLC3 Series PLC, you must connect your PC running MAPware-7000 to the unit to upload the Data Log file.

1. Start MAPware-7000, then click **Project > Transfer > Upload** or click the Upload ![icon] icon in the standard toolbar to bring up the **Upload from device** dialog box.
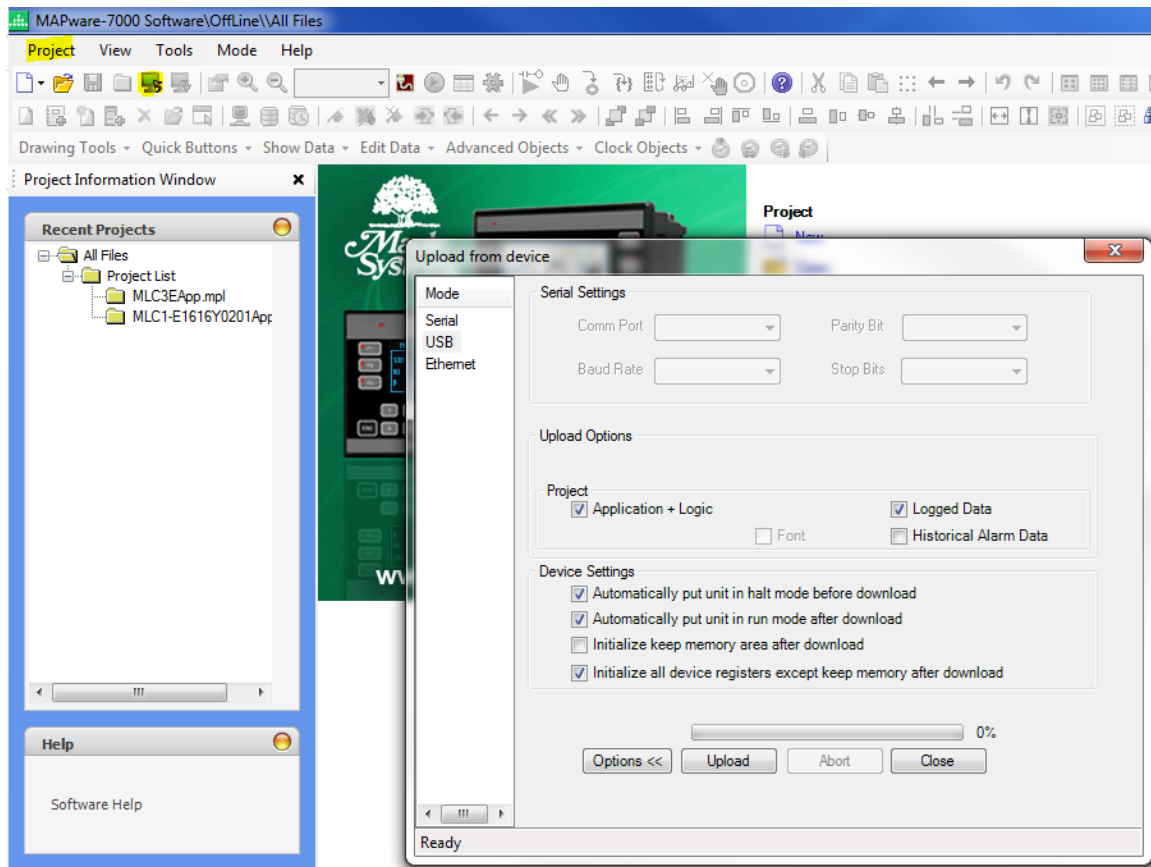
2. Select the **Application + Logic** and **Logged Data** options.
3. Click **Upload** to upload the project and logged data from the MLC.
4. When done, click **Close**.

5. From the **Tools** menu, select **Display Logged Data**.
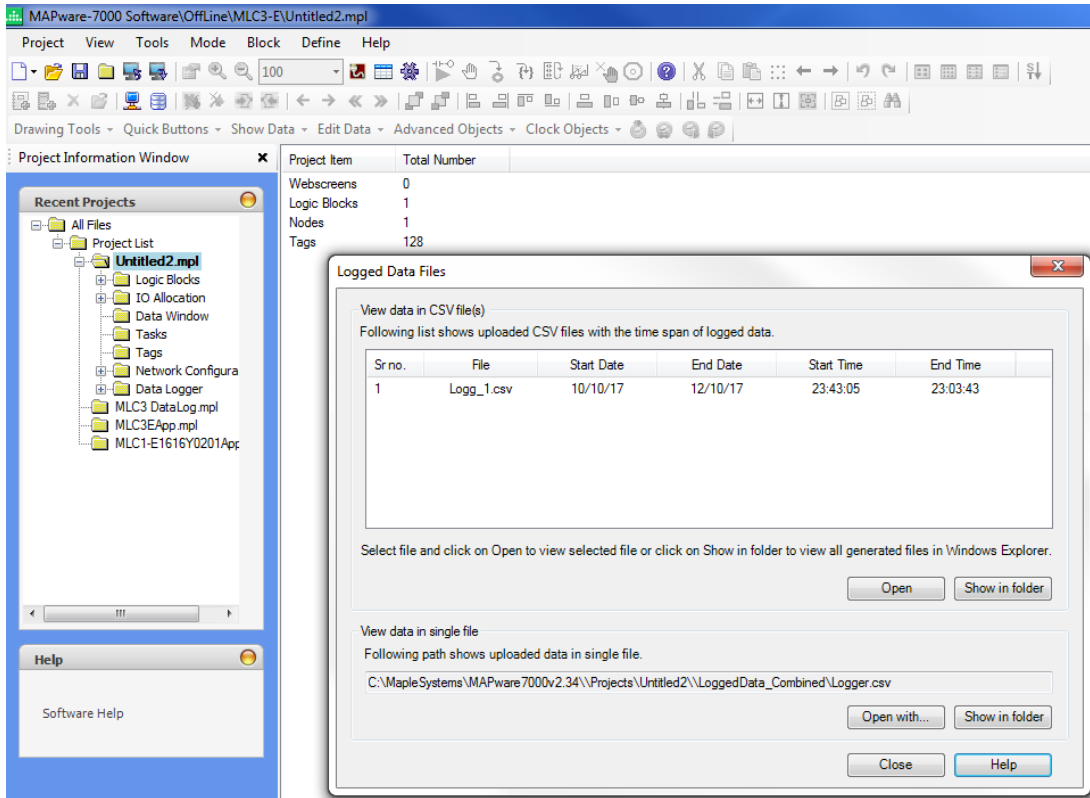
6. The top half of the **Logged Data Files** window allows you to view a CSV file for each **Data Log** group created separately.

7. Click **Open** to display the selected file in Excel. Click the **Show in folder** button to easily locate the file to copy or save it to another location.

8. The bottom half allows you to view all Data Logs (1-4) in a combined CSV file.
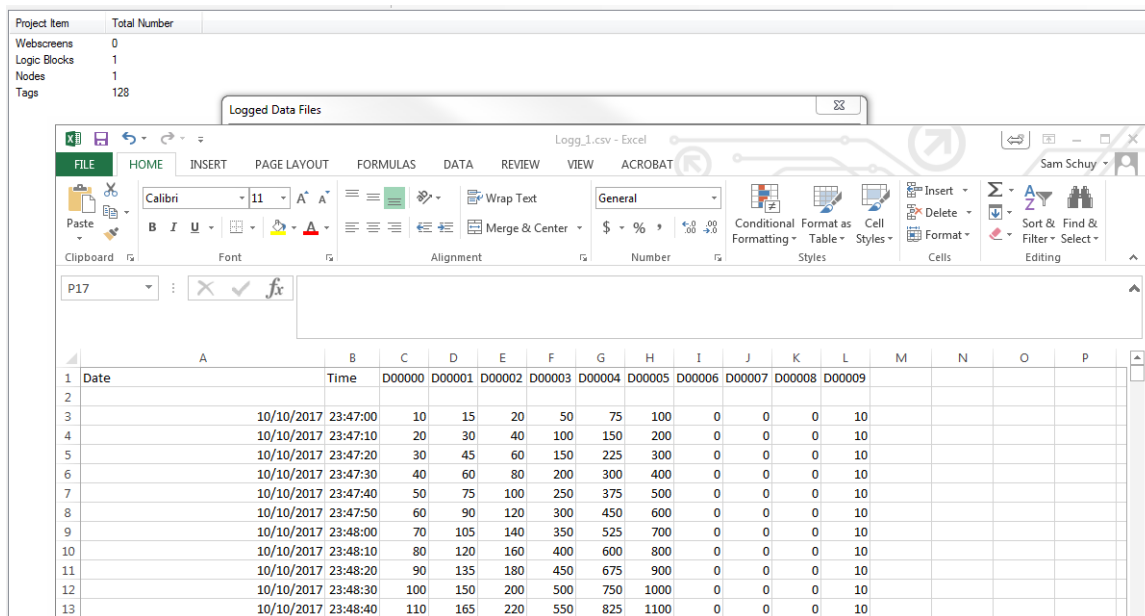
 Note: Date is in DD/MM/YYYY format. Time is HH:MM:SS format.

Retrieving Data Log files from the MLC2

The external data log is not uploaded using the upload option (**Project > Tools > Upload**) in MAPware-7000. To view the data, you must remove the SD Card from the MLC2 and load it into a PC. The SD card contains two folders; FTP, and LOG.
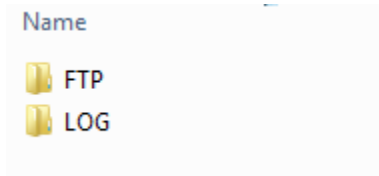
The FTP folder contains a list of files that were sent to the FTP server for each group. The LOG folder contains the *.csv files with data log data for each group. The file name corresponds to the File Name configured for the group.
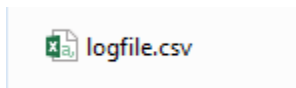
Simply open the file using a text editor or spreadsheet application capable of displaying csv data. The format of the data is a date and time stamp column followed by a column for each tag in the group. The Date stamp is in DD/MM/YY format. The time stamp is in HH:MM:SS format. The first line of the file is a header that specifies the column as the date, time or tag name being logged.

| DATE,TIME,ScaledInput, |
| 17/10/17,22:24:58,.860685110 |
| 17/10/17,22:25:03,.860685110 |
| 17/10/17,22:25:08,.860456228 |
| 17/10/17,22:25:13,.860456228 |
| 17/10/17,22:25:18,.860685110 |
| 17/10/17,22:25:23,.860456228 |
| 17/10/17,22:25:28,.860685110 |

	A	B	C
1	DATE	TIME	ScaledInput
2	17/10/17	22:24:58	0.86068511
3	17/10/17	22:25:03	0.86068511
4	17/10/17	22:25:08	0.860456228
5	17/10/17	22:25:13	0.860456228
6	17/10/17	22:25:18	0.86068511
7	17/10/17	22:25:23	0.860456228
8	17/10/17	22:25:28	0.86068511
9	17/10/17	22:25:33	0.86068511
10	17/10/17	22:25:38	0.860837697
11	17/10/17	22:25:43	0.860837697

Text Editor   Spreadsheet

Alternatively, data logs can be sent via FTP (File Transfer Protocol) from the SD card to an external FTP server. This is detailed below.

More information on using the **Data Logger** feature is available in the MAPware-7000 help files.

FTP (File Transfer Protocol) [MLC2-E Only]

The **FTP** (File Transfer Protocol) feature is used to transfer a *.csv file created using the **Data Logger** via Ethernet or serial connection from the MLC2 to an FTP server.

     ✒ **Note:** The FTP feature is only available for the MLC2 model.

Configuring the FTP Client

To configure the FTP client click on the *FTP* folder in the Project Information Window:
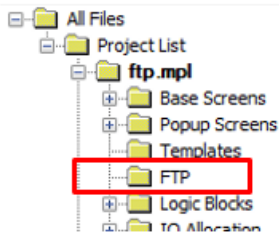
When at least one Group is configured in the external data log, the **Configure FTP Client** option is enabled. Check this box to enable the fields below:

Next, you must create several tags that will be used to control the FTP client:

- **FTP Enable Bit**: When the bit is ON files will be sent/uploaded to the FTP server at set time intervals as determined by the **Send File at Every** setting. When OFF, nothing will be sent.
- **File Resend Bit**: When the bit is ON, files are resent in the event that the server address or credentials are changed.
- **Source Media**: This is the device that contains the data log that will be sent to the FTP server. Currently the only option is SC Card.
- **Group Number***: This corresponds to the group configured in the data logger.
- **Destination Media***: This is the port on which the file is to be sent. Select Ethernet (3). Serial com ports are not supported at this time.

- **APN***: The Access Point Name is only required when using a COM port as the destination media and is not currently supported. This field is not applicable to Ethernet connections.
- **User Name*:** This is the user name used to log into the FTP server
- **Password*:** Password used to log into the FTP server
- **Server Address*:** The IPv4 address of the FTP server
- **Destination Path*:** Path to the folder on the FTP server where the data log is to be saved.
- **Send at Every*:** Interval at which data will be sent to the FTP server. The MLC will create a new log file every time it writes to the FTP server. Each file contains only the data for the current interval. The date and time for the interval is appended to the file name that is written to the server. See using the FTP feature section below for more details.
- **Status Registers:** The FTP feature provides the status registers detailed below to aid in trouble shooting any issues logging data to the FTP server. All status registers are 16-bit unsigned integers.
  - o **Media:** This status register contains error codes for issues related to connection with the SD Card where the data is saved.

Code	Meaning
0	No Error
1	Data Log in Progress
2	Media OK
41	SD Card Not Connected
42	USB Not Connected
43	File Systems Not Mounted
44	FTP Status File Does Not Exist
45	Upload File Does Not Exist
46	Unable to Open File
47	FTP Driver Not Defined
48	FTP Destination Media Not Supported
49	Source Media is Not Supported

  - o **Net Connection:** This status register contains error codes associated with network connection issues.

Code	Meaning
0	No Error
1	Connecting to Network or Link Down (Ethernet)
2	Connected to Network or Link Up (Ethernet)
ls	Network Connection Error
56	APN Error

  - o **FTP:** This status register contains error codes associated with connection to the FTP server.

Code	Meaning
0	No Error
1	Authentication in Progress
2	Authentication Complete
59	User Name Length Error

Code	Meaning
60	Password Length Error
61	Server Length Error
62	Group Number Error
63	Connection Error
64	Timeout Error
65	Server Error
66	Operation Not Allowed
71	User Name Error
72	User or Password Error
77	Operation Error
78	Upload Error
79	Error Creating File
80	Unknown FTP Error

o **FTP Send:** This register indicates the progress in sending the log file to the FTP server. Progress is indicated as a percent of total file size; i.e., a number from 0 to 100.

o **Communication:** This status register contains errors related to connection status.

Code	Meaning
0	No Error
1	FTP Busy
2	Healthy Communication
81	Communication Lost
82	Communication Error

* Asterisk indicates that a tag is not necessary for this option if the predetermined or fixed value setting is selected.

Using the FTP Feature

To use the FTP feature, there must be an FTP server accessible at the configured IP address. The FTP server must be configured with the user and password specified and must permit write access for that user. The SD card must be present and contain a log file to send.

Provided the above conditions are met, the MLC will periodically upload a *.csv file to the FTP server. A new file is created each time the MLC transmits data logger data to the FTP server. Each file will contain only the data recorded since the last file was sent.

The name of each *.csv file consists of the file name specified for the group number, appended with the date and time information for when the file was sent. The period contained in each log file is determined by the **Send File at Every** field in the **FTP Client Configuration** window.

For example, consider an external data logger group (**Data Logger > External memory**) configured with:

- File Name set to *logfile*:

File Name	logfile|

- Logging Mode set to *Power Up:*

Logging Mode	Power up ▼

- Log at Every set to 5 seconds:

  Log at Every 00:00:05 ⇅ (hh:mm:ss)

- Logging a tag called *ScaledInput:*

  Tags to be Logged (Max 30)
  Enter tag name..
  D00012(ScaledInput)   ⬆
                         ⬇

- While the FTP Client is configured with **Send File at Every** set to 1 minute:

  Send File at Every
  ◉ Fixed Value   ◯ Tag Based
  00:01:00 ⇅ (hh:mm:ss)   D00400 (FTP_Interval) ▼ ...

If the FTP client began sending files at 10:25:58 PM on October 17, 2017 the first file would be named:
logfile_17_10_17_22_24_58.csv

⬆ Up to higher level directory	
Name	**Size**
📄 logfile_17_10_17_22_24_58.csv	1 KB

This file will contain 12 data samples, one taken every five seconds, between 22:24:58 and 22:25:58.

	A	B	C
1	DATE	TIME	ScaledInput
2	17/10/17	22:24:58	0.86068511
3	17/10/17	22:25:03	0.86068511
4	17/10/17	22:25:08	0.860456228
5	17/10/17	22:25:13	0.860456228
6	17/10/17	22:25:18	0.86068511
7	17/10/17	22:25:23	0.860456228
8	17/10/17	22:25:28	0.86068511
9	17/10/17	22:25:33	0.86068511
10	17/10/17	22:25:38	0.860837697
11	17/10/17	22:25:43	0.860837697
12	17/10/17	22:25:48	0.860837697
13	17/10/17	22:25:53	0.860837697

After 10 minutes there will be ten *.csv files, each one containing 12 data samples.

Name	Size
⬆ Up to higher level directory	
📄 logfile_17_10_17_22_24_58.csv	1 KB
📄 logfile_17_10_17_22_25_58.csv	1 KB
📄 logfile_17_10_17_22_26_58.csv	1 KB
📄 logfile_17_10_17_22_27_58.csv	1 KB
📄 logfile_17_10_17_22_28_58.csv	1 KB
📄 logfile_17_10_17_22_29_58.csv	1 KB
📄 logfile_17_10_17_22_30_58.csv	1 KB
📄 logfile_17_10_17_22_31_58.csv	1 KB
📄 logfile_17_10_17_22_32_58.csv	1 KB
📄 logfile_17_10_17_22_33_58.csv	1 KB

Web Server/Web Screens [MLC3-E Only]
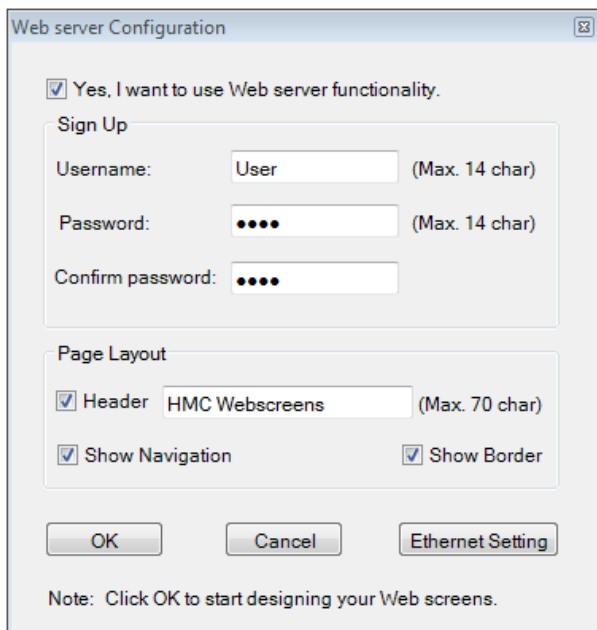
With the Web Server enabled on an MLC3-E, web screens can be created and then viewed through a web browser connected to the MLC3's IP address, allowing access to MLC data from a remote computer on the Ethernet network or, if exposed to the internet, from anywhere in the world. Web screens allow the remote user to both view and modify data in the MLC3. Up to 10 remote users can connect to the web server at one time. The objects used on web screens are restricted to text based data display / input objects, static images and navigation buttons. The following objects are available for use on web screens:

Icon	Command	Description
	Bit Display	Represent the state of a bit in a text format (1 for On, 0 for Off).
	Numeric Display	Displays a number stored in a data register.
	Bit Entry	Use this object to change the state of a bit (1 for On, 0 for Off).
	Set Bit	Use this object to latch a bit on
	Reset Bit	Use this object to latch a bit off
	Momentary Bit	Use this object to set a bit when the object is actuated and reset the bit when the object is released
	Toggle Bit	Use this object to toggle the status of a bit (set when pressed once; reset when pressed again)
	Write Value to Tag	Use this object to write a constant value to a tag
	Add Value to Tag	Use this object to add a constant value to a tag
	Subtract Value from Tag	Use this object to subtract a constant value from a tag
	Add Tag B to Tag A	Use this object to add the value of Tag B to Tag A
	Subtract Tag B from Tag A	Use this object to subtract the value of Tag B from Tag A
	Numeric Entry	Allows numeric or ASCII character entry to a targeted data register.
	Multilingual Text	Create a text box using Windows TrueType fonts. **Note**: the browser must be able to display the selected font, otherwise a default font is displayed.

	Picture Object	Display a JPG, BMP, or GIF picture using either predefined libraries or your own custom image.
	Go To Screen	Allows the remote user to navigate to other web screens.
	Go To Next Screen	Allows the remote user to navigate to the sequentially next web screen
	Go to Previous Screen	Allows the remote user to navigate to the previous web screen

Enabling the Web Server

Before a web screen can be created, the web server must be enabled and configured. To do this, select **Tools > Web Server Configuration** from the main menu and check the enable the web server box:

Once enabled, several fields are activated allowing the web server to be configured:

- Username / Password – A username and password are required. The remote operator will be prompted to enter these credentials before they can view the web screens in their browser. Both Username and Password can be up to 14 characters and can contain alpha numeric characters as well as; '*', '-', '_' and '.' characters.
- Header – If enabled, the text entered here will be displayed in the browser window as banner with white text on a blue background.
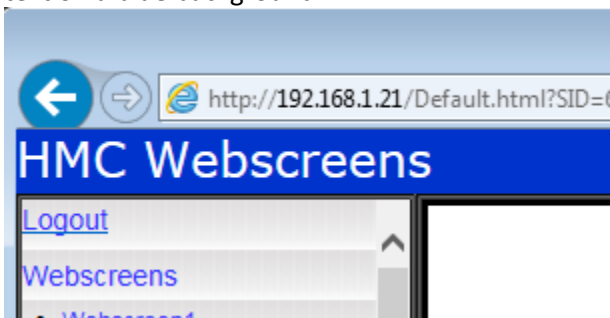
- Show Navigation – If enabled, a navigation bar will appear in the browser to the left of the active web screen with links to all of the web screens in the project. The operator can click on any of the screens

listed to go directly to that screen.
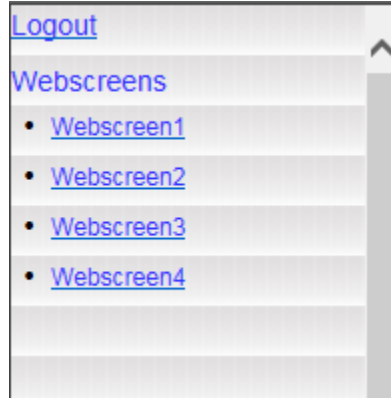
- Show Border – If enabled, a border will appear around the web screens.
- Ethernet Settings – This button is a shortcut to the Ethernet Settings tab on the **Project Configuration** window. The IP address entered here is the address the remote user will point their browser to in order to view the screens.

Creating Web screens

Once the web server is enabled, as described above, a **Webscreens** folder will appear in the **Project Information Window**.
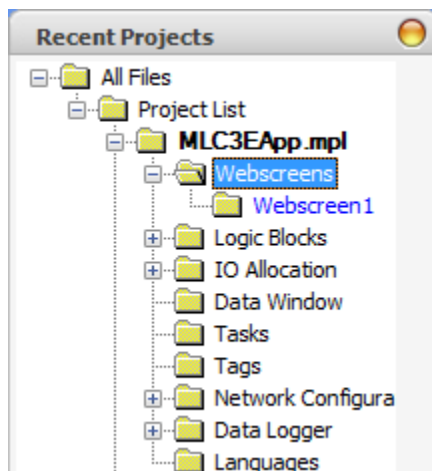
In this folder a default web screen will be created. The default screen name will appear in blue indicating that this is the first screen displayed when the user connects the web server. To create additional web screens, simply right click the **Webscreens** folder and select **New Web Screen** from the context menu.
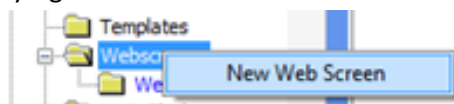
Once there are multiple web screens, a different web screen can be designated as the default screen by right clicking the screen name in the **Project Information Window** and selecting **Set as Default** screen from the context menu.

The process for adding objects to a web screen is the same as adding objects to a base screen. See the general *MAPware-7000 Programming Manual* for more information on how screen graphics objects work. Only objects

available for use on web screens will be selectable. There are some differences in how the objects behave on a web screen as opposed to on a base screen:

- Input objects used on web screens will not have a keypad associated with them. Instead, the browser will have an input method that can be used to enter data. For example, on a desktop, the remote user can click on a numeric input object with their mouse then enter data with a keyboard, pressing enter on the keyboard writes the data to the MLC3.
- Not all properties are available on web screens
- A font selected for an object on a web screen must be present on the browser; otherwise the browser will use a default font to display the object.
- GoTo buttons can only be used to go to another web screen. Popup screens are not allowed.
- External PLC tags cannot be used on a web screen.
- A maximum of 100 tags can be used per web screen.
- A maximum of 100 web screens can be configured per project.

Once the web screens are created, download the application to the MLC3. If the Ethernet settings have been changed, don't forget to check the Ethernet Settings checkbox in the download window.

Connecting to Web Screens with an Internet Browser

Once the screens have been created and downloaded to the MLC3, the next step is to connect to the web screens with an internet browser. If the computer running the browser is on the same Local Area Network (LAN) as the MLC3, simply enter the IP address assigned to the MLC3 in the web browser's address bar:
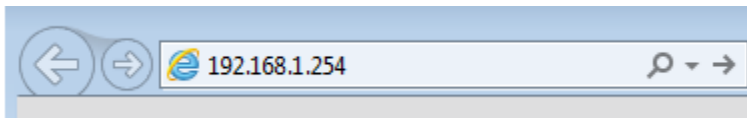
A prompt will appear to enter the Username and password configured for the web server:

Once that is entered click the Sign in button to display the default web screen:
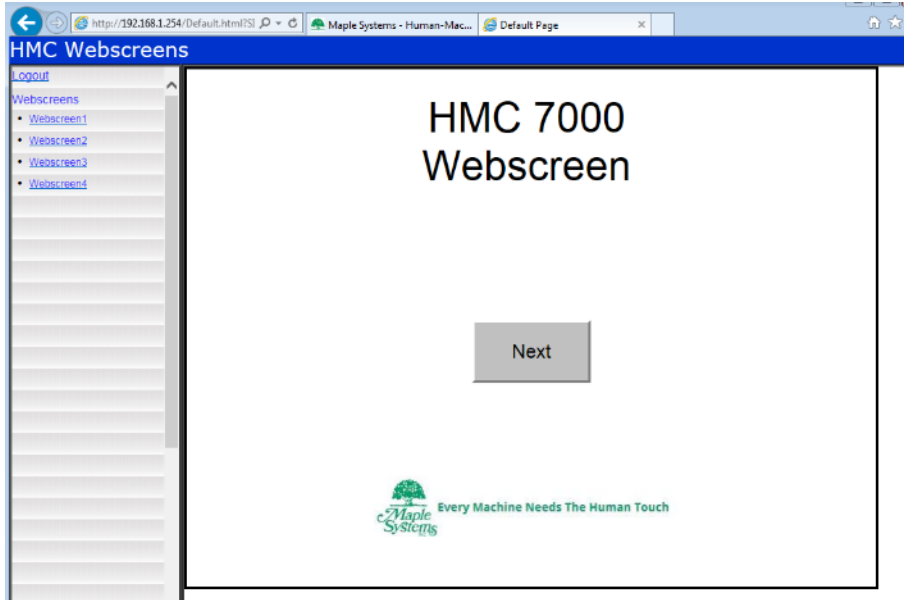
The following notes apply to web screens:

- A maximum of 10 users can connect to a single MLC3's web server.
- TCP port 80 is used to serve the web screens.
- Web screens have a refresh time of one second.

Connecting from the Internet

If the browser used to connect to the MLC3 is not on the same Local Area Network as the MLC3 it will be necessary to expose the MLC3's web server to the Internet. Typically this is done by opening port 80 in the router or by forwarding port 80 to the HMC's IP address by adding an entry in the port forwarding table in the router. On the Ethernet Settings tab of the **Project Configuration** window set the default gateway to the router's private IP address. Once the router is configured, connect to the MLC3 web server by entering the router's public IP address in the web browser address bar.

> **Note**: Network set up issues are beyond the scope of this manual. Contact a networking specialist for further information.

Project Password/Intellectual Property Protection

The ability to password protect your project is available in the **Project Configuration** window, **Project > Properties**.
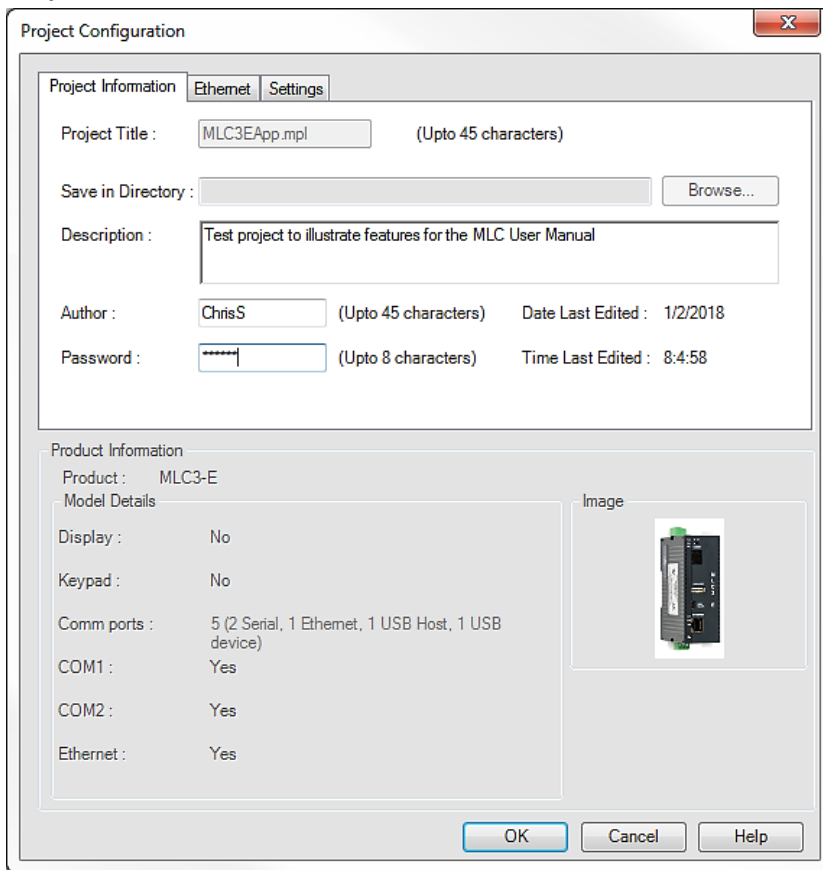
Creating an optional project password is a way to protect your project (and intellectual property) from unauthorized modification. The password can be up to 8 alphanumeric characters in length. If a password is entered in the **Project Information** tab of the **Project Configuration** window, the programmer will be required to enter the password whenever project is opened or uploaded from the MLC. Care must be taken to remember the project password, as it is not recoverable if forgotten. ***Maple Systems does not have a backdoor into password protected projects.*** If the password is not available, the project must be recreated from scratch.
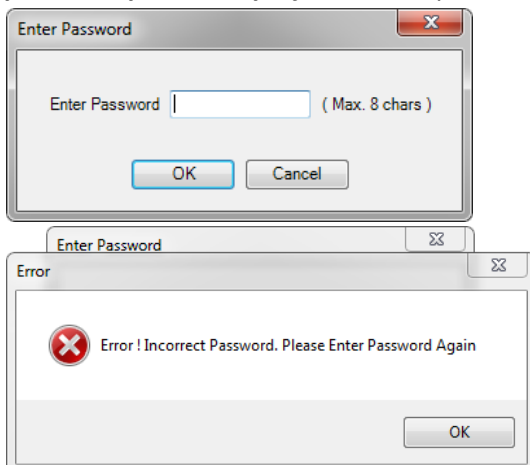
Application Memory Status

The Application Memory Status feature is a tool available in the MAPware-7000 software that is used to display memory used by the current project. This tool can be used to determine how much memory is still available for new screens, ladder logic, etc.
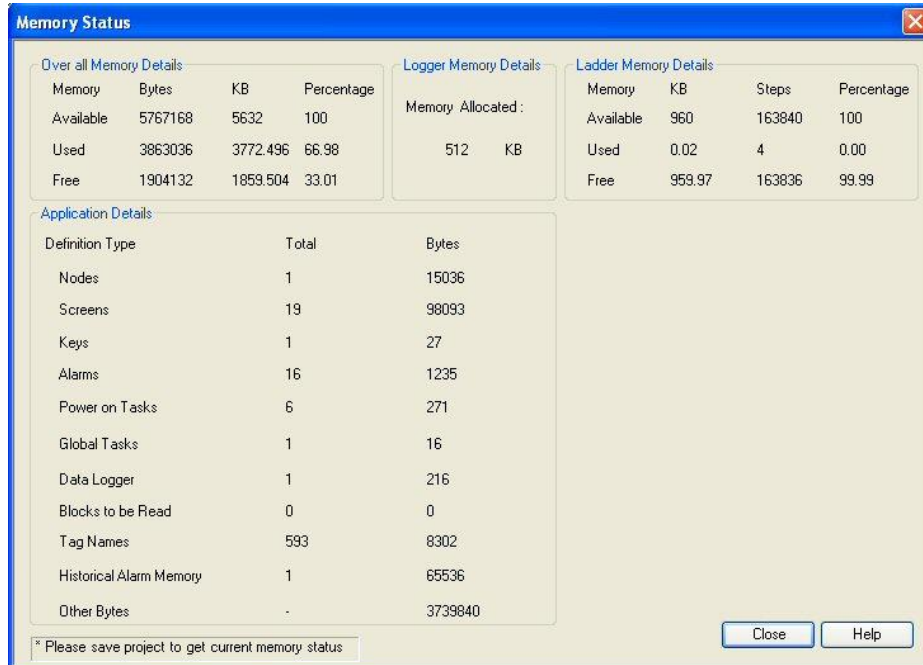
It also provides details on how the screen memory is currently allocated. The memory is divided into three main areas:

- Application Memory – This is the memory used for web screens, tasks, and tags.
- Data Logger Memory – Indicates how much memory has been allocated for the data logger feature.
- Ladder Logic Memory – Shows how much memory has been consumed for the ladder logic diagrams.

This tool is accessible is by clicking **Tools > Application Memory Status** from the menu bar.

Keep Memory Area and Retentive Tags

For projects created in the Native Ladder programming mode, the **Keep Memory Area** feature provides the ability to retain values in designated internal memory after removing power from the MLC. Click **Define >**

System Parameters from the menu bar.
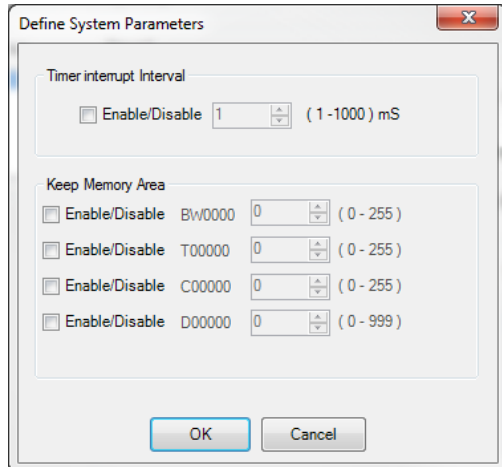
To enable, click the **Enable/Disable** checkbox next to the memory that you wish to use (BW, T, C, or D). Enter the number of registers (starting with 0) that you want as retentive memory. For BW, T and C registers, up to 256 registers (0 to 255) can be used as keep memory. Up to 1000 D registers (0 to 999) can be used as keep memory. The total number of keep memory registers cannot exceed 1000.

≣ **Note:** To use these registers, you must initialize them in the **Tag Database**.

For projects created in IEC 61131-3 programming mode, tags do not have specified addresses so the option to specify a range of addresses as keep memory described above is disabled. Instead there is an option to designate a tag as **Retentive** when the tag is created in the **Tag Database**:
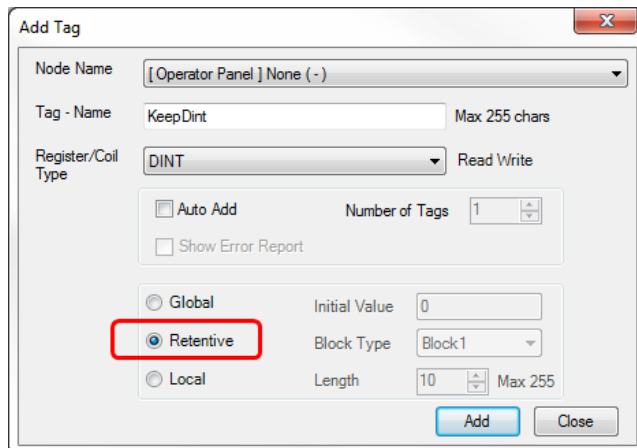
A tag created with this option will behave exactly as a tag created in the Keep Memory area of a Native Ladder project. A maximum of 2,000 bytes can be used to define **Retentive** tags (i.e., 500 DINTs, 1,000 INTS, 16,000 BOOLs etc., or some combination thereof). Tags created as **Global** can be changed to **Retentive** during project development and vice versa.

When you download your project into the MLC Series unit, there are two options listed in the **Download to Device** window of MAPware-7000 in the **Device Settings** section:
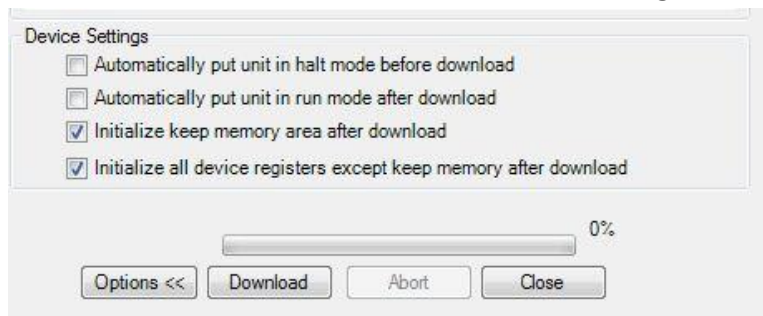
Initialize keep memory area after download – If checked, the designated **Keep Memory Areas** are set to 0 after download is completed. If not checked, the values in the **Keep Memory Areas** are retained. Similarly, in IEC 61131-3 mode projects **Retentive** tag values are set to 0 if this box is checked and retained through the download if it is unchecked.

Initialize all device registers except keep memory after download – If checked, the entire internal tag registers (i.e. D, X, XW, C, T, etc.) memory of the MLC Series are set to 0 after download *except for any registers designated as Keep Memory*.

> ⌨ **Note:** This memory area is automatically set to 0 if power is removed from the MLC unit. If not checked, values in these areas are retained, unless power is removed. This option is not available in IEC 61131-1 projects; instead non-Retentive tags are set to the value specified in the initial value field in the **Add Tag** or **Edit Tag** windows in the **Tag Database**.

Real Time Clock

> ⌨ **Note:** All MLC CPU models have a built-in RTC except for the -0604P/N models.

The internal battery powers the real-time clock (RTC) when the MLC is not powered by external 24VDC. Reserved registers pertaining to the clock:

Tag Address	Tag Name	Description	Access
SW0010	RTC day of month	Ex: 1-31, Unsigned format	Read Only
SW0011	RTC month	Ex: 1-12, Unsigned format	Read Only
SW0012	RTC year	Ex: 0-99, Unsigned format	Read Only
SW0013	RTC hour	Ex: 0-23, Unsigned format	Read Only
SW0014	RTC min	Ex: 0-59, Unsigned format	Read Only
SW0015	RTC sec	Ex: 0-59, Unsigned format	Read Only
SW0016	RTC day of week	Ex: 1-7, Unsigned format with 1=Sun…7=Sat	Read Only
S00003	Minute Change Status	Momentarily sets whenever the minute value in SW14 changes	Read Only
S00004	Hour Change Status	Momentarily sets when the hour value in SW13 changes	Read Only

S00005	Date Change Status	Momentarily sets when the date value in SW10 changes	Read Only
S00006	Month Change Status	Momentarily sets when the month value in SW11 changes	Read Only
S00007	Year Change Status	Momentarily sets when the year value in SW12 changes	Read Only
S00010	RTC battery status	Sets when battery is dead	Read Only
S00019	Invalid RTC date entry	Sets while an invalid value is in one of the RTC registers	Read Only
SW0003_02	RTC Status	Sets if the RTC is not working	Read Only

Writing the Date/Time

You cannot directly write to the clock using the SW10-16 internal system registers. You are able to set or change the internal RTC by using the Set Calendar (CLND) instruction in the ladder logic commands (see *The MAPware-7000 Ladder Logic Guide*)

Internal Battery

The internal battery is used to power the real-time clock (RTC) when the MLC is not powered by external 24VDC.

Appendix A – Internal Memory

Each MLC Series unit has internal memory which can be used for operations. The amount of memory available depends upon which MLC model you have selected. Part of this memory is reserved for specific uses (see *Appendix B – System Tag Memory*) while the rest of the memory is openly available for your application.

Native Ladder Memory Addresses

In Native Ladder mode, tags created in the **Tag Database** are given explicit addresses which are grouped together into various address types according data format and function. The address type is selected in the **Register/Coil Type** dropdown box in the **Add Tag** window (for directions on adding tags to a project refer to *Chapter 5 – Tags*).

Below is a table that defines the internal memory addresses available in Native Ladder projects. More detail on the function of each type is given in the sections that follow.

Type	Name	Range	Type	Access	Description
Index Register (I)	I00000	0	Register (2 bytes)	Read/Write	Index Register
Index Register (J)	J00000	0	Register (2 bytes)	Read/Write	Index Register
Index Register (K)	K00000	0	Register (2 bytes)	Read/Write	Index Register
Counter Coils	C.0000	0-255	Bit	Read/Write	Counter Coils
Counter Registers	C0000	0-255	Register (2 bytes)	Read/Write	Counter Registers
Timer Coils	T.0000	0-255	Bit	Read/Write	Timer Coils
Timer Registers	T00000	0-255	Register (2 bytes)	Read/Write	Timer Registers
Input Coils	X**00**000[1]	0-31015	Bit	Read Only	Physical Input Coils
Input Registers	XW**00**00[1]	0-3100	Register (2 bytes)	Read Only	Physical Input Registers **Note:** Input Coils and Registers share same memory
Output Coils	Y**00**000[1]	0-31015	Bit	Read/Write	Physical Output Coils
Output Registers	YW**00**00[1]	0-3100	Register (2 bytes)	Read/Write	Physical Output Registers **Note:** Output Coils and Registers share same memory
I/O Configuration Coils	M**00**000[1]	0-31015	Bit	Read/Write	IO coils for base expansion modules
I/O Configuration Registers	MW**00**00 or MW**00**00_00[1]	0-3100, 0-15	Bit or Register (2 bytes)	Read/Write	IO registers for base expansion modules **Note:** IO Coils and IO Registers share same memory
System Coils	S00000	0-99	Bit	Read/Write	Status and control bits for MLC
System Registers	SW0000 or SW0000_00	0-255, 0-15	Bit or Register (2 bytes)	Read/Write	Status and control registers for MLC

Internal Coils	B00000	0-4095	Bit	Read/Write	General Purpose Internal Coils
Internal Registers	BW0000	0-255	Register (2 bytes)	Read/Write	General Purpose Internal Registers **Note:** Internal Coils and Registers share same memory
Data Registers	D00000 or D00000_00	0-4095, 0-15	Bit or Register (1, 2 or 4 bytes)	Read/Write	General Purpose Registers
Retentive Registers	R00000	0-1399	Register (1, 2 or 4 bytes)	Read/Write	Non-volatile memory registers

1 – The first two digits of X, XW, Y, YW, M, and MW I/O addresses refer to the slot number of the I/O module. Built in I/O will have an address of 00, while slot 1 would have a prefix of 01.

⯈ **Note:** Memory areas are accessible in the **Tag Database** under Node Name: [MLC Model] None (-)

Generally speaking, the entire memory area of the MLC can be used for any purpose. However, the memory is split into several classifications so that they can be used for special purposes.

Index Registers (I, J, K)

The MLC Series has three 16-bit index registers (I, J, and K) that perform a specific purpose. As the name implies, these registers are used to 'index' or act as pointers to multiple registers. This method is called 'indirect addressing' because the MLC Series does not read/write to the specified address directly. Instead, the target address is determined indirectly by referencing the base address and the value read from the index register.

For example, if the MOV WORD instruction is used in a Ladder Logic Block, you can assign an index register to one of the operands:
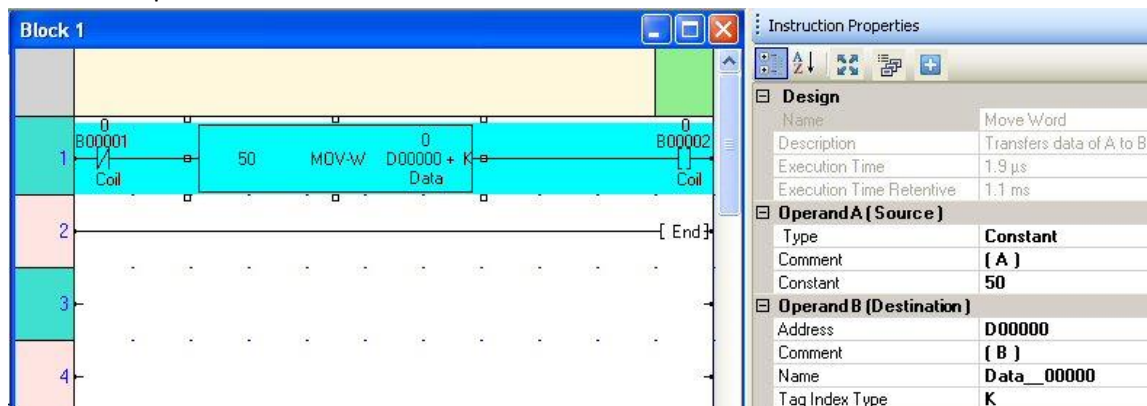
In this example, we use the K index register along with an internal register (D0). When this ladder rung is executed, the MLC Series reads the value in the K register. It then adds this value to the D0 register (as an offset) to determine the true register that the constant value 50 is written to. If the value in the K register is 1, then the constant value 50 is written to internal memory register D1. If the K value is 2, then 50 is written to internal memory register D2, and so on.

⯈ **Note:** Care should be taken that the value in the index register does not cause the MLC Series to read/write to a register that does not exist (the address range is not checked by the MLC).

Many of the ladder logic instructions have tag indexing capability. To find out which instructions support this feature, see the MAPware-7000 help files. In each section that defines a particular instruction, you will see an Operand table with a column marked Index. This column is checked for each operand that supports indexing.

Counter and Timer Registers/Coils (C, C., T, T.)

The MLC Series has four memory registers which are specifically used with the Counter and Timer instructions:

- Counter (C) registers
- Counter (C.) coils
- Timer (T) registers
- Timer (T.) coils

Each Timer or Counter register is 'connected' to a corresponding Timer or Counter coil. In a Timer or Counter instruction, when the Timer or Counter reaches its set value, the corresponding Timer or Counter coil is set.

For example, the picture below illustrates using the ON Timer:
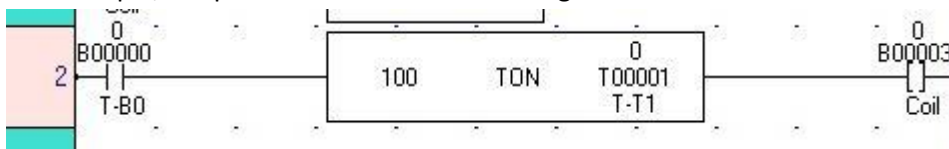
The timer T00001 register is used to count up to 1 sec (10msec timer). When the preset count (100) has been reached, timer coil T.0001 is set.

See *The MAPware-7000 Ladder Logic Programming Guide* for instructions that use counters and timers.

Input Registers/Coils (X, XW)

These registers and coils are designed for use by the physical inputs (digital or analog) from an attached or built-in input expansion module. These are the only internal memory addresses of the MLC which are read only. Therefore, the value read from these registers/coils always represents the current state of the physical input of the expansion module assigned to it. When you start a new project and assign I/O expansion modules to each expansion slot of the MLC, the MAPware-7000 will automatically assign the appropriate memory addresses according to slot location. For example, the sixteen built-in digital inputs of the MLC1-E1616P are assigned to coils X**00**000-X**00**015 and register XW**00**000. For a MLC Series unit that has eight digital inputs in expansion slot #1, the MAPware-7000 software automatically assigns coil addresses X**01**000-X**01**007 and register XW**01**00 to the inputs. If sixteen digital inputs are located in slot#4, the software assigns coil addresses X**04**000-X**04**015 and register XW**04**00 to the inputs. If an analog module with four analog inputs is installed into slot#2, then no coil addresses would be assigned (since these are analog inputs) and registers XW**02**00-XW**02**03 are assigned.

The X and XW registers are mapped to the same memory area in the MLC with sixteen X coils per XW register. To determine the register and bit location of an X coil memory area use the following equation: Xnnn = XW(nnn/16 quotient), bit (nnn/16 remainder).

For example, X00161 corresponds to the second least significant bit (bit 1) of XW0010 (161/16 = 10r1).

Output Registers/Coils (Y, YW)

These registers and coils are designed for use by the physical outputs (digital or analog) from an attached or built-in output expansion module. The value read from these internal registers/coils is used to determine and set the current state of the physical output of the expansion module assigned to it. When you start a new project and assign I/O expansion modules to each expansion slot of the MLC, the MAPware-7000 will automatically assign the appropriate memory addresses according to slot location. For example, the eight built-in digital outputs of the MLC1-F0808N0201 are assigned to coils Y**00**000-Y**00**007 and register YW**00**00. For a MLC Series unit that has eight digital outputs in expansion slot #1, the MAPware-7000 software automatically assigns coil addresses Y**01**000-Y**01**007 and register YW**01**00 to the outputs. If twelve digital outputs are located in slot#3, the software assigns coil addresses Y**03**000-Y**03**011 and register YW**03**00 to the outputs. If an analog module with two analog outputs is installed into slot#5, then no coil addresses would be assigned (since these are analog outputs) and registers YW**05**00-YW**05**01 are assigned.

The Y and YW registers are mapped to the same memory area in the MLC with sixteen Y coils per YW register. To determine the register and bit location of an Y coil memory area use the following equation: Ynnn = YW(nnn/16 quotient), bit (nnn/16 remainder).

For example, Y00161 corresponds to the second least significant bit (bit 1) of YW0010, (161/16 = 10r1).

IO Configuration Registers/Coils (M, MW)

These registers and coils are reserved to perform special functions when the MLC Series has built-in or attached expansion modules. For example, the MLC1-F0604N has the following I/O Register/Coil memory allocated for the six digital inputs and five digital outputs built into the unit:

- M00240   HSC Enable Bit (CH1)
- M00241   HSC Reset Bit (CH1)
- M00400   HSC Enable Bit (CH2)
- M00401   HSC Reset Bit (CH2)
- MW0010   HSC Configuration Register (CH1)
- MW0011   HSC Current Register (CH1)
- MW0013   HSC Preset Register (CH1)
- MW0020   HSC Configuration Register (CH2)
- MW0021   HSC Current Register (CH2)
- MW0023   HSC Preset Register (CH2)

These registers are used when the built-in I/O of the MLC1-F0604N are configured as high speed counters.

The MAPware-7000 automatically allocates the necessary IO registers to each expansion module according to slot location and type of module. For example, a MLE high-speed I/O expansion module that has eight digital inputs and eight digital outputs in expansion slot #1 has coil addresses M**01**080, M**01**081, M**01**176, and M**01**177 and registers MW**01**00-MW**01**09 assigned for two high-speed counters. If slot#2 contains MLE-A0402T (4 analog inputs, 2 analog outputs) module, the software assigns no coil addresses and registers MW**02**06, MS**02**10,

MS**02**14, etc. to read/write to the analog registers. When slot#4 contains a MLE-A0004 (4 analog outputs) module, MAPware-7000 assigns the MW**04**04, MW**04**08, etc. registers for the analog outputs.

For more information on which M/MW registers are allocated per MLC product, consult the Help files in MAPware-7000 (Configuration (M) or (MW) Registers), or the M Memory section at the end of this chapter.

The M and MW registers are mapped to the same memory area in the MLC with sixteen M coils per MW register. To determine the register and bit location of a M coil memory area use the following equation: Mnnn = MW(nnn/16 quotient), bit (nnn/16 remainder).

For example, M00161 corresponds to the second least significant bit (bit 1) of MW0010, (161/16 = 10r1). Individual bits of the MW registers can be accessed with using the MW0000_00 notation, so the previous example would be written as MW0010_01.

System Registers/Coils (S, SW)

These registers and coils are reserved by the MLC for specific purposes. A total of 100 system coils and 256 system registers are reserved. Although undefined system registers and coils could be used by the MLC programmer, this area of memory may be used in the future for predefined uses. Therefore, use of this area of memory should be limited to the predefined coils and registers. For a list of these memory areas, see *Appendix B –System Tag Memory*.

Internal Registers/Coils (B, BW)

These are general purpose registers and coils which can be used at the discretion of the MLC programmer. This area of memory is non-retentive so any data stored in these registers is lost once power has been removed from the MLC.

The B coil memory and the BW register memory actually occupy the same area of memory in the MLC Series. Therefore, the least significant bit in register BW0 corresponds to the B0 coil. Likewise, the B00018 coil shares the same memory area as bit 2 of BW0001. To determine the register and bit location of a B coil memory area use the following equation: Bnnnn = BW(nnnn/16 quotient), bit (nnnn/16 remainder).

Therefore B00050 coil corresponds to BW0003, bit 2 (50/16 = 3r2).

Data Registers (D)

These are general purpose registers which can be used at the discretion of the MLC programmer. This area of memory is non-retentive so any data stored in these registers is lost once power has been removed from the MLC.

Notice that these registers can be configured as bits as well.

Retentive Registers (R)

These are general registers which can be used at the discretion of the MLC programmer. This area of memory is retentive so any data stored in these registers remains once power has been removed from the MLC.

⌨ **Note:** that reading/writing to these registers requires more processing time than does using any of the other registers in memory due to their retentive nature. Therefore, be aware that using these registers in ladder logic blocks requires more processing time.

IEC 61131-3 Tags

In projects created using the IEC 61131-3 programming mode, tags are not, necessarily, assigned an explicit memory address. Instead they are created with a data type that determines the tags format and function. The data type is selected in the *Register/Coil Type* dropdown list in the Add Tag window of the **Tag Database**. For detailed instructions on creating tags refer to *Chapter 5 – Tags*.

Below is a table displaying the tag data types available in IEC 61131-3 projects.

Data Type	Description	Bits	Value Range
BOOL	Boolean value can have one of two values: 1 (ON / True) or 0 (OFF / False)	1	0 to 1
BYTE	Unsigned Short (same as USINT)	8	0 to 255
DINT	Signed Double Integer	32	-2,147,483,648 to 2,147,483,647
DWORD	Unsigned Double Integer (same as UDINT)	32	0 to 65,535
INT	Signed Integer	16	-32,768 to 32,767
REAL	IEEE 754 format, single precision floating point	32	$\pm 3.403 \times 10^{38}$
SINT	Signed Short Integer	8	-128 to 127
STRING	Variable length string of ASCII characters	1 to 255 bytes	N/A
TIME	Time of day	32	N/A
UDINT	Unsigned Double Integer (same as DWORD)	32	0 to 4,294,967,295
UINT	Unsigned Integer	16	0 to 65,535
USINT	Unsigned Short Integer (Same as BYTE)	8	0 to 255
WORD	Unsigned Integer (Same as UINT)	16	0 to 65,535

In addition to these data types, IEC 61131-3 projects share some of the data types available in Native Ladder projects. In these cases the tags are given an explicit address which is visible in the **Tag Database**. These registers are; Retentive Registers (R), System Coils / Registers (S, SW), Input Registers (X, XW), Output Registers (Y, YW), IO Configuration Coils / Registers (M, MW). Refer to the previous section of this Appendix for information on available addresses and using these registers.

Appendix B – System Tag Memory

The following tables list all of the internal memory of the MLC Series which is reserved for specific purposes.

S memory (System bits)

Tag Address	Tag Name	Type	Access	Description
S00000	Carry bit	Coil (1 bit)	Read/Write	Overflow indication in math operations of ladder and also used in rotate with carry instruction.
S00003	Minute Change status	Coil (1 bit)	Read Only	1 for every change in minute for one scan cycle
S00004	Hour Change status	Coil (1 bit)	Read Only	1 for every change in hour for one scan cycle
S00005	Date Change status	Coil (1 bit)	Read Only	1 for every change in date for one scan cycle
S00006	Month Change status	Coil (1 bit)	Read Only	1 for every change in month for one scan cycle
S00007	Year Change status	Coil (1 bit)	Read Only	1 for every change in year for one scan cycle
S00010	RTC Battery status	Coil (1 bit)	Read Only	0: Battery voltage is OK (i.e. above 2.2 V); 1: Low battery (i.e. below 2.2 V)
S00011	Default Coil	Coil (1 bit)	Read/Write	New objects added to the project will default to this address. Does not perform any function.
S00019	Invalid RTC date entry	Coil (1 bit)	Read Only	0=valid date; 1= Invalid date
S00021	COM1 failed node reconnect control	Coil (1 bit)	Read/Write	If this bit is set communication with the failed nodes is detected after scan time SW0018 for port1. By default : ON
S00022	COM2 failed node reconnect control	Coil (1 bit)	Read/Write	If this bit is set communication with the failed nodes is detected after scan time SW0019 for port2. By default : ON
S00034	Ladder Instruction Error Status	Coil (1 bit)	Read/Write	Set if Division by zero operation is performed in the ladder instruction and for invalid conditions or operands in case of conversion instructions.
S00092	Initialize Port COM1	Coil (1 bit)	Read/Write	
S00093	Initialize Port COM1	Coil (1 bit)	Read/Write	

SW memory (System registers)

Tag Address	Tag Name	Type	Access	Description
SW0000	DefaultReg	Register (2 bytes)	Read/ Write	Newly added objects will default to using this register, does not perform any function
SW0003_14	COM1 Status	Bit	Read Only	0 = Communication Error; 1= Communicating with PLC
SW0003_15	COM2 Status	Bit	Read Only	0 = Communication Error; 1= Communicating with PLC
SW0010	RTC Day of Month	Register (2 bytes)	Read Only	RTC day in integer format (00-31)
SW0011	RTC Month	Register (2 bytes)	Read Only	RTC month in integer format (01-12)
SW0012	RTC Year	Register (2 bytes)	Read Only	RTC year in integer format (00-99)
SW0013	RTC Hour	Register (2 bytes)	Read Only	RTC hour in integer format (00-23)
SW0014	RTC Min	Register (2 bytes)	Read Only	RTC minute in integer format (00-59)
SW0015	RTC Sec	Register (2 bytes)	Read Only	RTC sec in integer format (00-59)
SW0016	RTC Day of Week	Register (2 bytes)	Read Only	RTC day of week in integer format (00-06) with 0=Sunday, …, Saturday=6
SW0017	Scan time register	Register (2 bytes)	Read Only	Shows time required to execute reading inputs, executing tasks, executing ladder logic, update outputs, etc. in 0.1 milliseconds. Use ####.# format for display
SW0018	COM1 failed node reconnect time (Sec)	Register (2 bytes)	Read/Write	Shows time in sec to recover the communication with failed nodes for port1. The default value is 60Sec
SW0019	COM2 failed node reconnect time (Sec)	Register (2 bytes)	Read/Write	Shows time in sec to recover the communication with failed nodes for port1. The default value is 60Sec
SW0020	Driver scan time register for port 1	Register (2 bytes)	Read Only	
SW0021	Driver scan time register for port 2	Register (2 bytes)	Read Only	
SW0046	Ladder Scan Time	Register (2 bytes)	Read Only	Value is multiple of 0.1 mSec
SW0047	Timer Interrupt Ladder Execution Time	Register (2 bytes)	Read Only	Value is multiple of 0.1 mSec
SW0064_01	Status tag: COM1 Node Address 1	Coil (1 bit per node)	Read Only	Status of the node 1: Node Present, 0: Not Present.

Tag Address	Tag Name	Type	Access	Description
SW0080_00	Status tag: COM2 Node Address 0	Coil (1 bit per node)	Read Only	Status of the node 0: Node Present, 0: Not Present.
SW0080_01	Status tag: COM2 Node Address 1	Coil (1 bit per node)	Read Only	Status of the node 1: Node Present, 0: Not Present.
SW0239	Baud Rate COM 1	Register (2 bytes)	Read/Write	
SW0240	Parity COM 1	Register (2 bytes)	Read/Write	
SW0241	Data Bits COM 1	Register (2 bytes)	Read/Write	
SW0242	Stop Bits COM 1	Register (2 bytes)	Read/Write	
SW0243	Node Address COM 1	Register (2 bytes)	Read/Write	
SW0244	Baud Rate COM 2	Register (2 bytes)	Read/Write	
SW0245	Parity COM 2	Register (2 bytes)	Read/Write	
SW0246	Data Bits COM 1	Register (2 bytes)	Read/Write	
SW0247	Stop Bits COM 2	Register (2 bytes)	Read/Write	
SW0248	Node Address COM 2	Register (2 bytes)	Read/Write	

M memory (I/O configuration Coils)

In addition to the tags listed here, M addressed tags are assigned when an IO module is added to the project and are used to control the IO module. For a list of MW tags available for each module and detailed information on how to use them, refer to the *MLC Series IO Module Guide* or consult the Help files in MAPware-7000.

Tag Address	Tag Name	Type	Access	Description
M00016	CPU error	Coil (1 bit)	Read Only	On at error state
M00018	Program error	Coil (1 bit)	Read Only	On at error state. **Note**: Includes the Ladder Scan Time.
M00021	Clock/calendar illegal value warning	Coil (1 bit)	Read Only	ON when clock/calendar data is illegal
M00022	Retentive data invalid warning	Coil (1 bit)	Read Only	ON when retentive data in RAM is invalid
M00027	Watchdog timer error	Coil (1 bit)	Read Only	On at error state

Tag Address	Tag Name	Type	Access	Description
M00033	Ladder scan time error	Coil (1 bit)	Read Only	On when scan time exceeds 200 msec. (default)
M00240	HSC Enable Bit (CH1)	Coil (1 bit)	Read/ Write	Start Counting for High Speed Counter (CH1)
M00241	HSC Reset Bit(CH1)	Coil (1 bit)	Read/ Write	Reset High Speed Counter (CH1)
M00242	HSC Preset Reached Bit(CH1)	Coil (1 bit)	Read Only	
M00480	System timer coil for 0.1 sec interval	Coil (1 bit)	Read Only	Toggle at 50% duty cycle
M00481	System timer coil for 0.2 sec interval	Coil (1 bit)	Read Only	Toggle at 50% duty cycle
M00482	System timer coil for 0.4 sec interval	Coil (1 bit)	Read Only	Toggle at 50% duty cycle
M00483	System timer coil for 0.8 sec interval	Coil (1 bit)	Read Only	Toggle at 50% duty cycle
M00484	System timer coil for 1 sec interval	Coil (1 bit)	Read Only	Toggle at 50% duty cycle
M00485	System timer coil for 2 sec interval	Coil (1 bit)	Read Only	Toggle at 50% duty cycle
M00486	System timer coil for 4 sec interval	Coil (1 bit)	Read Only	Toggle at 50% duty cycle
M00487	System timer coil for 8 sec interval	Coil (1 bit)	Read Only	Toggle at 50% duty cycle
M00496	Timer interrupt ladder execution status	Coil (1 bit)	Read Only	ON when timer program is executing
M00512	Always On	Coil (1 bit)	Read Only	The coil is always on.
M00513	Always Off	Coil (1 bit)	Read Only	The coil is always off.
M00720	HSC Enable Bit (CH3)	Coil (1 bit)	Read/ Write	
M00721	HSC Reset Bit(CH3)	Coil (1 bit)	Read/ Write	Reset High Speed Counter (CH1)
M00722	HSC Preset Reached Bit(CH3)	Coil (1 bit)	Read Only	
M00816	Pulse Enable flag (CH1)	Coil (1 bit)	Read Only	
M00818	ON duty setting error flag (CH1)	Coil (1 bit)	Read Only	
M00819	Frequency Setting error flag (CH1)	Coil (1 bit)	Read Only	
M00820	Acceleration Time Setting error flag (CH1)	Coil (1 bit)	Read Only	
M00821	Deceleration Time Setting error flag (CH1)	Coil (1 bit)	Read Only	

Tag Address	Tag Name	Type	Access	Description
M00822	No of total pulses setting error flag (CH1)	Coil (1 bit)	Read Only	
M00823	End of Total Pulses Flag (CH1)	Coil (1 bit)	Read Only	
M00832	Pulse Enable flag (CH2)	Coil (1 bit)	Read Only	
M00834	ON duty setting error flag (CH2)	Coil (1 bit)	Read Only	
M00835	Frequency Setting error flag (CH2)	Coil (1 bit)	Read Only	
M00836	Acceleration Time Setting error flag (CH2)	Coil (1 bit)	Read Only	
M00837	Deceleration Time Setting error flag (CH2)	Coil (1 bit)	Read Only	
M00838	No of total pulses setting error flag (CH2)	Coil (1 bit)	Read Only	
M00839	End of Total Pulses Flag (CH2)	Coil (1 bit)	Read Only	

MW memory (I/O configuration registers)

In addition to the tags listed here, MW addressed tags are assigned when an IO module is added to the project and are used to control the IO module. For a list of MW tags available for each module and detailed information on how to use them, refer to the *MLC Series I/O Module Guide* or consult the MAPware-7000 help files.

Tag Address	Tag Name	Type	Access	Description
MW00	PLC Mode Control (bits 0-3)	Register (2 bytes)	Read/Write	0 = Initialization 1 = Halt mode 2 = Run mode 3 = Run in Forced mode 4 = Hold mode 6 = Fault mode **Note:** If an invalid number is entered into this register, then previous value will be retained.
MW01	Error Flag 1	Register (2 bytes)	Read/Write	Each bit represents status of M16-M31 errors (ex. Bit 0 = M16, Bit 1 = M17, etc.)
MW02	Error Flag 2	Register (2 bytes)	Read/Write	Each bit represents status of M32-M47 errors (ex. Bit 0 = M32, Bit 1 = M33, etc.)
MW03	Run/Stop Switch Control	Register (2 bytes)	Read/Write	0= Run 1 = Halt **Note:** This value is stored in retentive memory

Tag Address	Tag Name	Type	Access	Description
MW05	Variable input filter constant (0 to 15 msec)	Register (2 bytes)	Read/Write	Enabled when MW10 bit 15 is ON. When OFF, digital filter is set to 10 msec. Used by PLC to determine if state of any digital input is valid. Ex. By default, state must be steady for 10 msec. to be considered valid.
MW10	Configuration register for digital inputs	Register (2 bytes)	Read/Write	Use to set the special input settings when configuring Channel 1 or 2 as: • Normal mode • Interrupt mode • High Speed counter mode • Quadrature mode • High Speed counter with rate
MW11	Configuration register for setting PWM or Pulse mode for Output Y0	Register (4 bytes)	Read Only	Use to configure mode of digital output when in pulse mode or pulse width modulation (PWM) mode
MW12-15	High-speed counter (HSC) Preset values	Register (4 bytes)	Read/Write	Use to set the preset value for HSC
MW16-19	High-speed counter (HSC) Current values	Register (4 bytes)	Read/Write	Current value for HSC
MW20	High-speed counter (HSC) Control flags	Register (2 bytes)	Read/Write	Use to configure the control flags for HSC
MW21	High-speed counter (HSC) special output control flags	Register (2 bytes)	Read/Write	Displays the output control flags for HSC
MW23	High-speed counter (HSC) Preset value (for Channel 2)	Register (2 bytes)	Read/Write	Use to set the preset value for HSC Channel 2
MW30	System timer coils	Register (2 bytes)	Read Only	
MW31	User Interrupt program Status Flags	Register (2 bytes)	Read/Write	Status or IO interrupt routines 0 = Running 1 = Not Running
MW32	Always ON.OFF bits	Register (2 bytes)	Read/Write	
MW40	HSC Configuration register (CH3)	Register (2 bytes)	Read/Write	
MW41	HSC register (CH3)	Register (4 bytes)	Read Only	

Tag Address	Tag Name	Type	Access	Description
MW43	HSC preset register (CH3)	Register (4 bytes)	Read/Write	
MW54	Trapezoidal Min pulse Count register (CH1)	Register (4 bytes)	Read Only	
MW56	Trapezoidal Min pulse Count Register (CH2)	Register (4 bytes)	Read Only	
MW59	PWM configure register (CH1)	Register (2 bytes)	Read/Write	
MW60	Minimum Frequency Setting (CH1)	Register (4 bytes)	Read/Write	
MW62	On duty Max Freq. Setting Register (CH1)	Register (4 bytes)	Read/Write	
MW64	Acceleration Time (CH1)	Register (2 bytes)	Read/Write	
MW65	Deceleration Time (CH1)	Register (2 bytes)	Read/Write	
MW66	Total Pulses (CH1)	Register (4 bytes)	Read/Write	
MW68	Elapsed Value (CH1)	Register (4 bytes)	Read/Write	
MW75	PWM configure register (CH2)	Register (2 bytes)	Read/Write	
MW76	Minimum Frequency Setting (CH2)	Register (4 bytes)	Read/Write	
MW78	On duty Max Freq. Setting Register (CH2)	Register (4 bytes)	Read/Write	
MW80	Acceleration Time (CH2)	Register (2 bytes)	Read/Write	
MW81	Deceleration Time (CH2)	Register (2 bytes)	Read/Write	
MW82	Total Pulses (CH2)	Register (4 bytes)	Read/Write	
MW84	Elapsed Value (CH2)	Register (4 bytes)	Read/Write	

Your Industrial Control Solutions Source

_____

www.maplesystems.com

1010-1054 Rev. 01