EZPLC Nano

Table of contents

Warnings

Programmable control devices such as the NanoPLC are not fail-safe devices and as such must not be used for stand-alone protection in any application. Unless proper safeguards are used, unwanted start-ups could result in equipment damage or personal injury. The operator must be made aware of this hazard and appropriate precautions must be taken.

In addition, consideration must be given to the use of an emergency stop function that is independent of the NanoPLC.

The diagrams and examples in this user manual are included for illustrative purposes only. The manufacturer cannot assume responsibility or liability for actual use based on the diagrams and examples.

Trademarks

Manual part number PLC-Nano-M

Designed, Built and Marketed by AVG
4140 Utica Ridge Rd. · Bettendorf, IA 52722-1327
Phone: **1-877-774-EASY** · Fax: **1-877-775-EASY** · flash.ezautomation.net

EU Information

The NanoPLC is manufactured in compliance with European Union (EU) Directives and carries the CE mark. They been tested under CE Test Standard #EN55011, and is submitted for UL Certification.

Products with CE marks perform their required functions safely and adhere to relevant standards as specified by EU directives provided they are used according to their intended purpose and that the instructions in this manual are adhered to. The protection provided by the equipment may be impaired if this equipment is not used in accordance with this manual. Only replacement parts supplied by AVG Automation or its agents should be used.

Technical Support

Consult PLC Editor Programming Software Help. You may also find answers to your questions in the operator interface section of our website @ flash.ezautomation.net. If you still need assistance, please call our technical support at 1-877-774-EASY or FAX us at 1-877-775-EASY.

SELV Circuits

All electrical circuits connected to the communications port receptacle are rated as Safety Extra Low Voltage (SELV).

Preventative and Maintenance Cleaning

No special preventative measurement is required.

Product Overview

Thank You for using AVG Automation's new series of compact programmable logic controllers - the NanoPLC. The NanoPLC is a fixed I/O PLC and offers 16 Digital DC inputs and 8 Digital Relay outputs.

This manual presents information on the installation, wiring and specifications of the NanoPLC. It also covers the troubleshooting and maintenance of an existing setup and provides understanding on how to connect the PLCs with other components in your control system.

The following table illustrates how the NanoPLC compares to other PLC units that we offer:

Feature	Standard PLC	Jr PLC	MicroPLC	NanoPLC
Input Power	24VDC or 110VAC	24VDC or 110VAC	24VDC or 110VAC	24VDC or 110VAC
I/O Type	Modular	Modular	Fixed I/O	Fixed I/O
No. of I/O	Models with 32, 48, 64, and 96 I/O	4 Mod (32/IO)	24 DC In, 8 DC Out, 8 Relay Out, 4 Analog In, 4 Analog Out	16 DC In, 8 Relay output
Com Ports	2	1	1 or 2	1 or 2
Analog I/O	Yes (Modules available)	Yes (Modules available)	Yes, fixed	None
Specialty I/O	PWM, Highspeed counter	PWM, Highspeed counter	PWM, Highspeed counter	PWM, Highspeed counter

All of our PLCs use ladder logic for programming and share the same programming environment. In addition, the ladder logic developed for one model can be used with other models by changing the target device. The ladder instructions are subject to the model's limitations.

NanoPLC Specifications

INPUT POWER

Voltage: 24 VDC nominal (20-28VDC)
Power Supply Capacity: 3.3V @ 1Amp
Max. Power Consumption: 10 Watts

MECHANICAL

External dimensions: 5.93” x 4.88” x 2.19” (150.74mm x 124.16mm x 55.55mm)
Mounting: Back Panel mount using screws

ENVIRONMENTAL

Operating Temperature: -10°C to 60°C
Storage Temperature: -20°C to 70°C
Humidity: 10-95% Non-Condensing
Atmospheric Conditions: Non-corrosive gases
Vibration: 5 to 55Hz, 2g for 2 hours in X, Y, and Z axis
Shock: 10g for under 12ms in the X, Y, and Z axis
Electrical Noise: Nema ICS 2-230 Showering arc, ANSI C37.90a SWC; Level C Chattering Relay Test

MEMORY

User Program Memory: 128 KB
Total Number of Registers: 8192 (256 registers retained on power down)
PLC Typical Scan Time: 5 ms (1K Boolean)

COMMUNICATIONS

Communication ports: 1 RS-232 port (9 pin D-Sub), 1 RS-485 port (optional)
Digital Input specifications (DC Input)

Quantity: 16
Input Voltage Range: 12-26 VDC
Input Current: 1.92 mA @12 VDC or 4mA @ 24VDC
On Voltage > 12VDC and OFF Voltage < 2VDC
Minimum ON Current: 2 mA, OFF Current: 0.2mA
Red LED Status Indicators
(See Discrete Input Section for more info)

Digital Output specifications (Relay)

Quantity: 8
Max Switching Voltage: 277VAC or 30VDC
Max Switching Power: 300W
Rated Switching Current: 10A
Red LED Status Indicators
(See Digital Output Section for more details)

Installation

Safety Considerations

Please follow all applicable local and national codes to ensure maximum safety of the equipment and personnel. The installation and operational environment must be maintained per the latest revision of these codes.

You are responsible to determine the codes to be followed and to verify the compliance of equipment, installation, and operation with the latest revision of these codes.

It is an absolute must to follow all applicable sections of:
-The National Fire Code
-The National Electrical Code (NEC)
-The National Electrical Manufacturer's Association (NEMA) codes

Safety Guidelines

Safety is the most important element of a proper system installation. Adhering to these safety considerations ensures the safety of yourself and others, as well as the condition of your equipment. We recommend reviewing the following safety guidelines:

1) Disconnecting Main Power
The main power switch should be easily accessible to the operators and maintenance personnel. It is important to make sure that all other sources of power including pneumatic and hydraulic are de-energized before starting the work on a machine or process controlled by the PLC.

2) Safety Circuits
Most of the machines are installed with safety circuits such as limit switches, emergency stop push buttons, and interlocks. These circuits should always be hardwired directly to the NanoPLC. These devices must be wired in series so that when any one device opens, the PLC is automatically de-energized. This removes power to the machine. These circuits should not be altered in any case, since this could result in serious injury or damage to the machine.

3) Fail-Safe Operation
Our products are not fault-tolerant. They are not designed or intended for use as online control equipment in hazardous environments requiring fail-safe performance, such as in operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life-support machines, weapons systems, clutch control systems on presses, in which the failure of the product could lead directly to death, personal injury or severe physical or environmental damage. External fail-safe and/or redundant components are required to make your control system fail-safe.

Installation Considerations

Our products have been designed and tested for operation in the most demanding industrial environments. Modern solid-state industrial controls are complex electronic equipment that operate at low levels of voltage and current, co-existing with components that operate at much higher levels of power. The difference in operating power characteristics between the high and low power control devices creates the possibility of unwanted signals being generated, thus causing interference. The interference, which is a by-product of electrical noise, is not present at all times. However, if it appears at random and for brief periods of time, it can cause disruptions and errors in the operation of a control system.

Enhancement of a system's noise level immunity and its tolerance to other environmental hazards can be accomplished by following proper system installation guidelines. The recommendations are of a general nature and constitute good industrial installation practice.

General Environmental Considerations

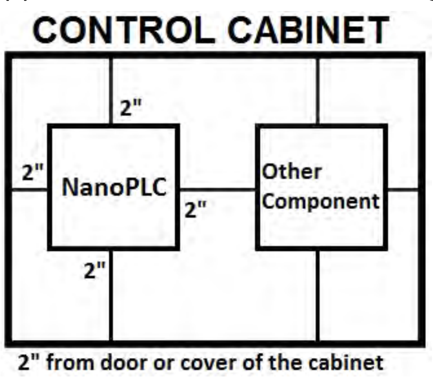
Avoid installing NanoPLC in areas where the following conditions may exist:

- Environmental temperatures above or below those specified by the NanoPLC
- Prolonged exposure to humidity and liquids which may be sprayed or splashed on the equipment
- Dusty environments where airborne particles may accumulate on equipment causing reduction of heat dissipation and reduction in effective electrical spacing between components
- Areas with excessive vibration
- Areas with high-radiated electrical noise, such as near fields of transmitting antennas and areas in close proximity of arc welding stations

Physical Layout in a Control Cabinet

When possible, cabinets housing electronic equipment should be designed with provisions for natural or forced ventilation to facilitate heat dissipation. Observe the following rules for cabinet installation:

- Heat generating equipment (power supplies and other heat inducing components) should be installed toward the top of the cabinet. The lower space in the cabinet is cooler than the top area.
- Install heat-sensitive components in the lower section.
- Provide enough space between components to allow a free flow of air for better heat dissipation.
- Provide the maximum possible physical separation between solid state and electromechanical controls. If possible, the electromechanical controls (motors, starters, solenoids, etc.) should be housed separately or at the farthest point when enclosed within the cabinet.



We recommend that the NanoPLC have a minimum clear space of 2" on all sides for adequate ventilation as shown in the image on the right.

Electrical Considerations

This section is designed to provide you with a very basic understanding of electrical noise and how to keep it away from CPUs. Industrial plants have a number of generators of electrical noise that are sometimes also referred to as Radio Frequency Interference (RFI). Anytime an inductive load like a motor, motor starter, or solenoid is turned off, it generates a burst of excess energy that has to flow back to ground, just like electrical energy from a lightning storm has to flow back to Earth. RFI is short bursts of electrical energy at very high frequencies. Other sources include RF Welders or Radio Transmitters.

Effect of RFI on Electronic Automation Equipment

Electronic controls use faster and faster CPUs today. These CPUs are also operating at 2.5V to 5VDC logic level power supply. RFI, if allowed to enter the CPU inside, is a killer of logic. A CPU under this environment loses its brain and behaves erratically. A smart industrial-grade CPU like the NanoPLC Card Engine, when faced with RFI, halts its operation instead of giving false outputs.

Types of RFI

RFI enters electronic controls in two ways: radiated RFI or conducted RFI. For most practical purposes, electronic devices, unless sitting right next to a powerful RFI transmitter, will not be affected by noise because air space severely attenuates such interference. On the other hand, conducted RFI travels over conductive surfaces such as power supply wires, electrical wiring of field devices, and worst of all; improper ground planes.

Equipment cabinets usually incorporate one or two doors and/or hinged cabinet panels. Relying on door hinges and swinging panels for a good metallic bond between hinged parts and the main body of the cabinet does not insure adequate grounding. Instead, the use of ground straps is recommended. It is vital for the reliable operation of any electronic device to have any of its metallic surfaces well grounded to Earth. This not only provides for safe operation, it will also drain out any conducted RFI to Earth, away from the CPU's signal ground.

Shielding from RFI

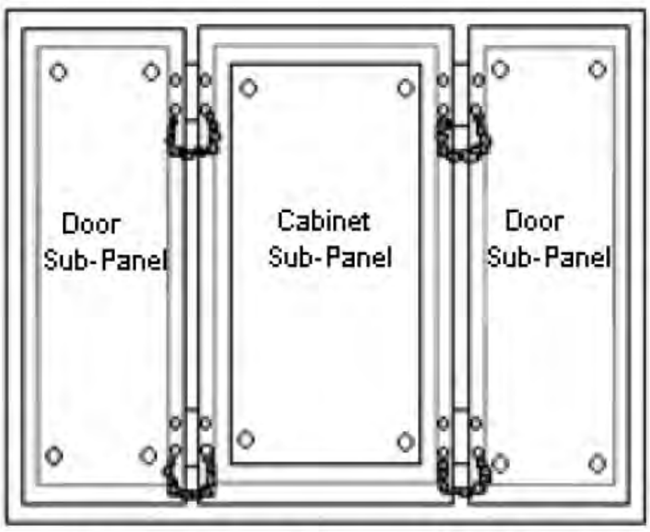
Shielded Cables

Power cables, I/O cables or wiring, and communication cables should all be separate so that they do not couple the conducted RFI on any of these wires/ cables. Another path for RFI into the PLC is through its RS232 port. Hence, the cables to this port must be shielded properly.

Equipment Cabinets

As mentioned, equipment cabinets typically incorporate one or two doors and/ or hinged cabinet panels. In addition, sub-panels may be utilized on those electronic controls and electromechanical items that are mounted. The goal is to create a medium for mounting the equipment and ensure grounding of the control's chassis to it. However, the door hinges and swinging panels by themselves are not enough to ensure adequate grounding.

Similarly, the equipment enclosures are generally either painted or anodized. Mounting of painted or anodized enclosures to like surfaces also does not ensure good metallic contact between the equipment chassis and cabinet. It is imperative that the equipment chassis are grounded such as through the use of grounding straps as illustrated below.

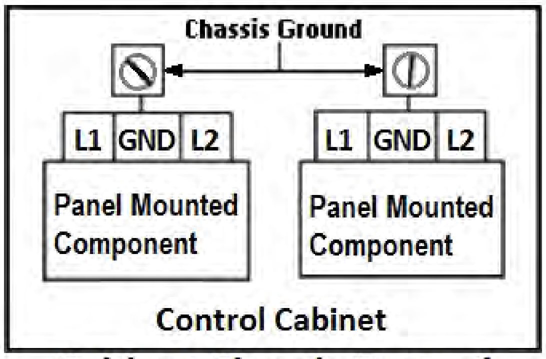


Cabinet Door Grounding Straps

Cabinet Wiring

The wiring of the NanoPLC to the “field” outside the cabinet must be by design. The wiring cannot be random in order to get the various points of the cabinet and the “field” electrically connected. Below are some general rules that apply in most situations:

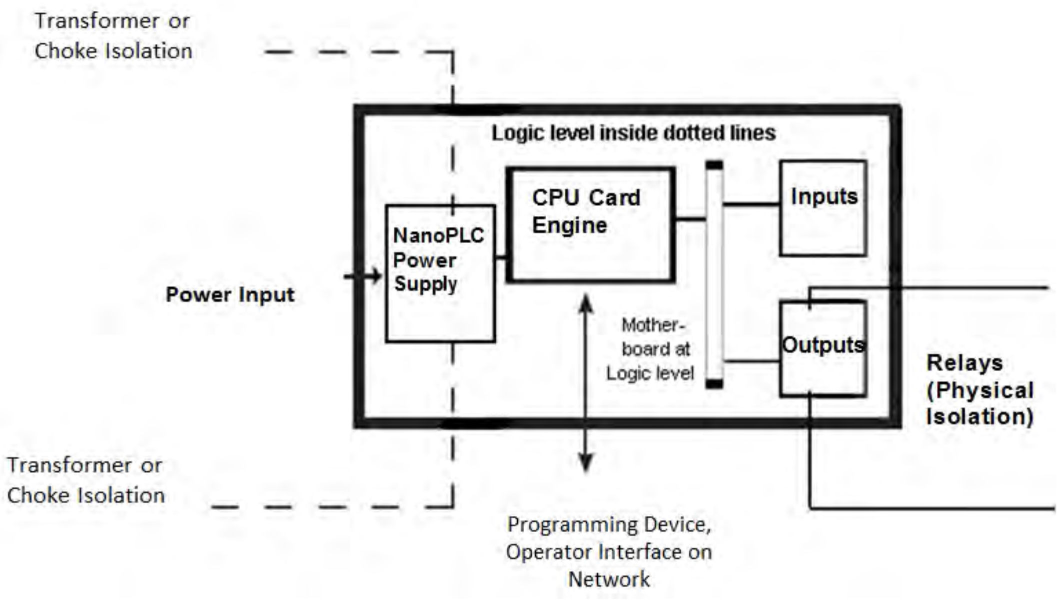
- Provide a separate power source to electronic controls and keep this power bus away from any I/O power.
- The cabinet should be wired with a safety ground (the main safety ground wire gauge is determined by the cabinet's total current consumption) and in accordance with all electrical code requirements.
- Once the cabinet doors, stationary sub-panels and swing-out sub-panels have been “strapped” to the main cabinet, it is not necessary to run safety ground wires from the equipment chassis terminals to the main safety ground connection.
- The safety ground terminal of each component can, and should be, connected with the shortest wire possible, to the cabinet or sub-panel frame.
- Plan the wiring routing. Keep all switched power in separate ducts and if there is AC and DC power being switched, keep the wiring of each branch separate from all wires and cables carrying low level signals.
- Keep all three phase power outside of the cabinet, but if it becomes necessary, keep the runs as short as possible and maintain the maximum possible distance between the three phase bus and all other wiring.
- Primary power leads to the control equipment (Base power terminals) should be made with a two wire twisted cable with approximately 12 turns per foot. The length of these cables should be kept to a minimum, and to the greatest extent possible, such cable runs should be kept separate from other wiring.



Cabinet Chassis Ground

Isolation within the PLC

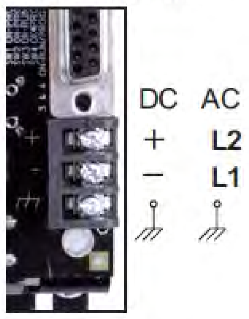
It is a common practice to isolate the sensitive CPU of the PLC from RFI by providing Transformer or Choke Isolation on the Power Supply. NanoPLC isolates the conducted RFI by both means; transformer/choke isolation as well as physical isolation for outputs as shown below.


Transformer or Choke Isolation
Logic level inside dotted lines
CPU Card Engine
Inputs
NanoPLC Power Supply
Power Input
Mother-board at Logic level
Outputs
Relays (Physical Isolation)
Transformer or Choke Isolation
Programming Device, Operator Interface on Network

Dealing with AC Line Noise

The AC power available in house outlets and at sub-stations powering industrial and commercial applications is generally generated at a power station miles away from the point of usage. While the generated power output starts its journey "clean" and free of noise, it is "polluted" by radio and TV frequencies, spikes from reactive kickbacks due to switching heavy inductive and capacitive loads in transmission lines, and from other interference.

Power connection



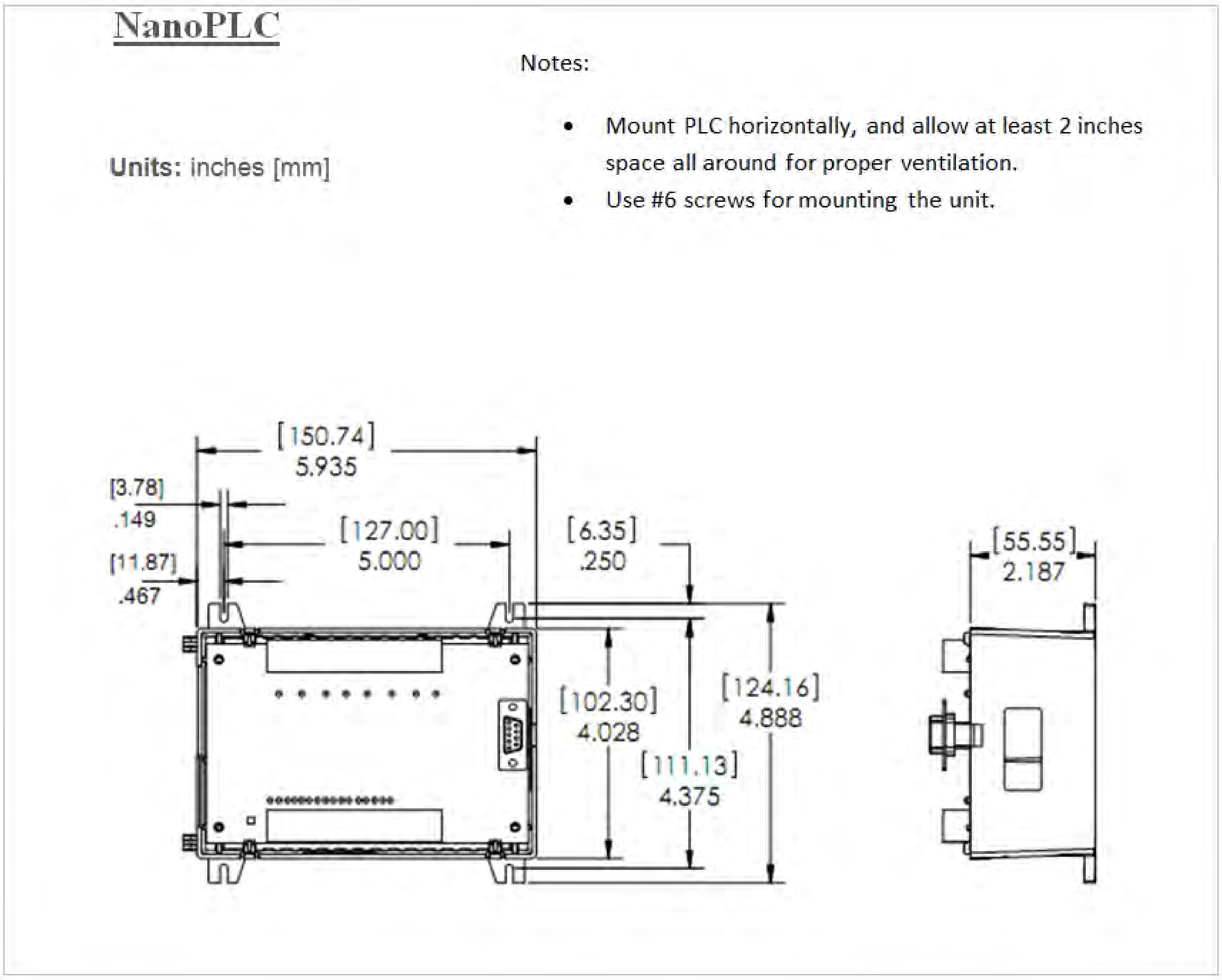
The best option to effectively eliminate or greatly reduce voltage fluctuations, spikes and line noise is through the use of isolation, constant voltage or power line conditioner transformer.

Isolation transformers are passive devices that do not have DC paths between the circuits they isolate. The transformer provides attenuation to spikes and common mode noise, but has virtually no effect on transverse mode noise and does not provide protection for voltage fluctuations.

Constant voltage transformers are static Ferro-resonant transformers that can accept fluctuating AC voltage input (within a specified range) and maintain a constant voltage output. The transformers provide good attenuation to transverse mode type noise, however, are ineffective for attenuation of common mode type signals.

Power line conditioning transformers provide good line regulation and are effective in providing attenuation to both common and transverse mode types of noise.
All of the mentioned transformer types are available by various manufacturers and they come in different varieties of operating voltages, power ratings, and frequencies.

Mounting Information



NanoPLC

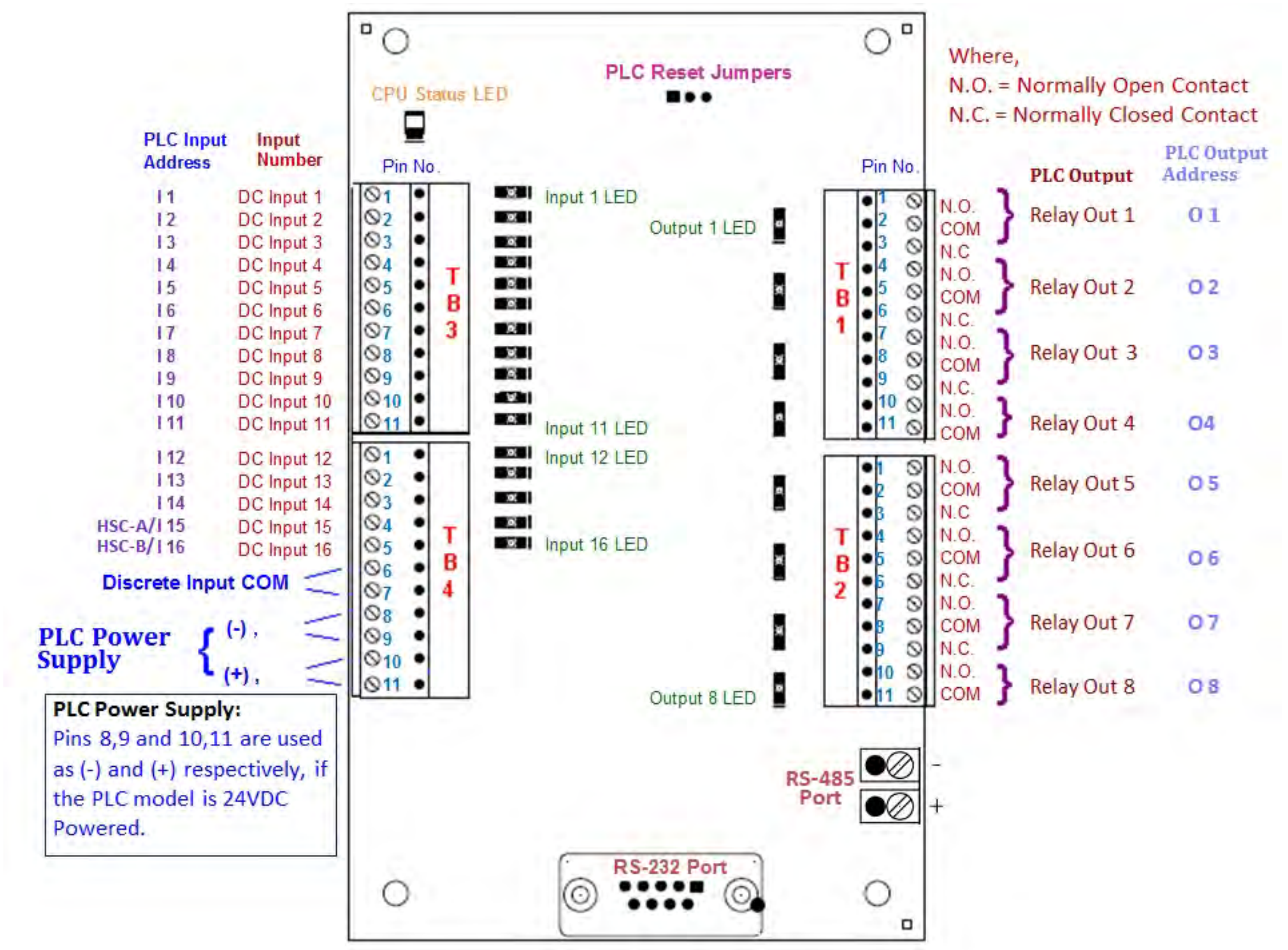
Units: inches [mm]

Notes:

- Mount PLC horizontally, and allow at least 2 inches space all around for proper ventilation.
- Use #6 screws for mounting the unit.

Wiring and Communications

Terminal Layout



PLC Power Supply:
Pins 8,9 and 10,11 are used as (-) and (+) respectively, if the PLC model is 24VDC Powered.

Terminal Pinouts

TB3 Pinout Information	
Pin No.	**Pin Function**
1	DC Input(1)
2	DC Input(2)
3	DC Input(3)
4	DC Input(4)
5	DC Input(5)
6	DC Input(6)
7	DC Input(7)
8	DC Input(8)
9	DC Input(9)
10	DC Input(10)
11	DC Input(11)

TB4 Pinout Information	
Pin No.	**Pin Function**
1	DC Input(12)
2	DC Input(13)
3	DC Input(14)
4	DC Input(15)
5	DC Input(16)
6	DC Input COM
7	DC Input COM
8	PLC Power Supply: (-)
9	PLC Power Supply: (-)
10	PLC Power Supply: (+)
11	PLC Power Supply: (+)

TB1 Pinout Information	
Pin No.	**Pin Function**
1	Output (1)_Normally open
2	Output (1)_COM
3	Output (1)_Normally closed
4	Output (2)_Normally open
5	Output (2)_COM
6	Output (2)_Normally closed
7	Output (3)_Normally open
8	Output (3)_COM
9	Output (3)_Normally closed
10	Output (4)_Normally open
11	Output (4)_COM

TB2 Pinout Information	
Pin No.	**Pin Function**
1	Output (5)_Normally open
2	Output (5)_COM
3	Output (5)_Normally closed
4	Output (6)_Normally open
5	Output (6)_COM
6	Output (6)_Normally closed
7	Output (7)_Normally open
8	Output (7)_COM
9	Output (7)_Normally closed
10	Output (8)_Normally open
11	Output (8)_COM

Note: **NanoPLC Power Supply**
DC Powered model: Pins 8, 9 and 10, 11 on terminal TB4 are used as (-) and (+) respectively

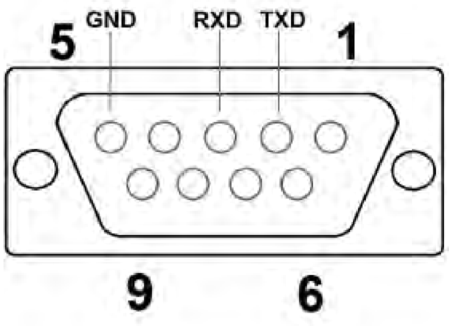
Powering the NanoPLC

Connect the power input wires into the PLC's power terminals. (Illustration provided in the Terminal Layout section.) Supply 24VDC nominal (20-28VDC) or 110 VAC (95-125 VAC) power to the system depending on the model. When power flows to the NanoPLC unit's power terminal, the PWR indicator LED on the base of the unit should turn on showing a green LED. If not, remove power from the system and check all the wiring.

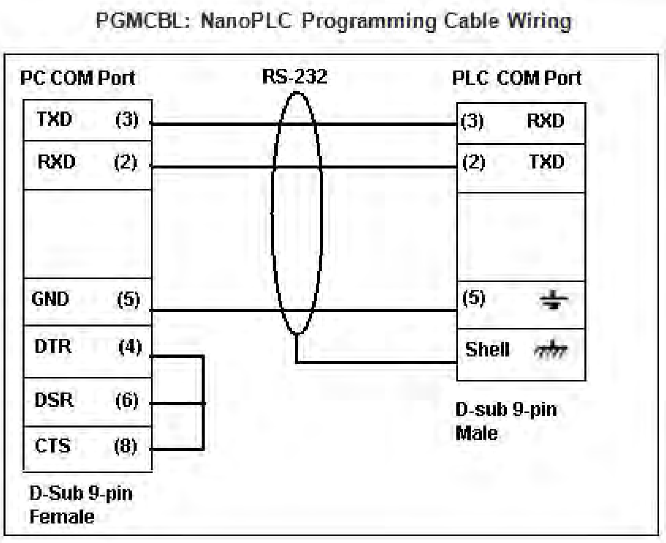
Indicator Light	Status Description	Possible Error
Blinking Red	PLC running in Boot Mode only	Missing firmware / Outdated firmware installed
Blinking Red and Green	Data abort Power cycle	Bad or corrupted program
Red	PLC locked up	Hardware trouble
Blank - No Light	No program loaded or PLC in "Stop Mode"	Power source not connected or inadequate
Green	Program loaded and running	

RS232 (COM1)

The NanoPLC model has a built-in serial port (COM1 PORT) located on the 9-pin D-Sub connector. COM1 PORT is an RS-232 port which requires an appropriate RS-232C cable (**P/N: EZ-PGMCBL**) for programming the NanoPLC through a PC. It serves as the default programming port on the NanoPLC. Since COM1 has fixed communication parameters, you can always connect the programming software to the PLC through the port without needing to make different configuration changes.

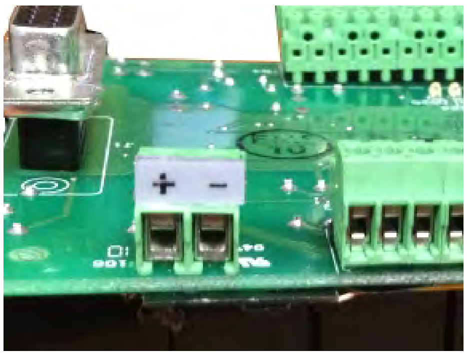


CAUTION! Keep the signal reference GND wire well protected from external noise by using shielded cable.



RS485 Port (Optional)

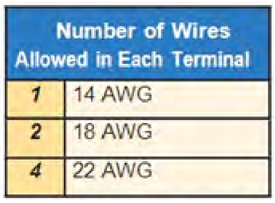
The NanoPLC model comes available with two serial communication ports: a RS232 port and a RS485 port. The optional RS485 port allows the user to utilize Communication Instructions as outlined in the Appendix.



To enable this port, you would use the Open Port command. The port only needs to be opened once. It will stay on unless the Close Port command is used or until a power cycle. More information about the functionality of this port is available in the Software Manual Help section.

Wiring I/O Connections

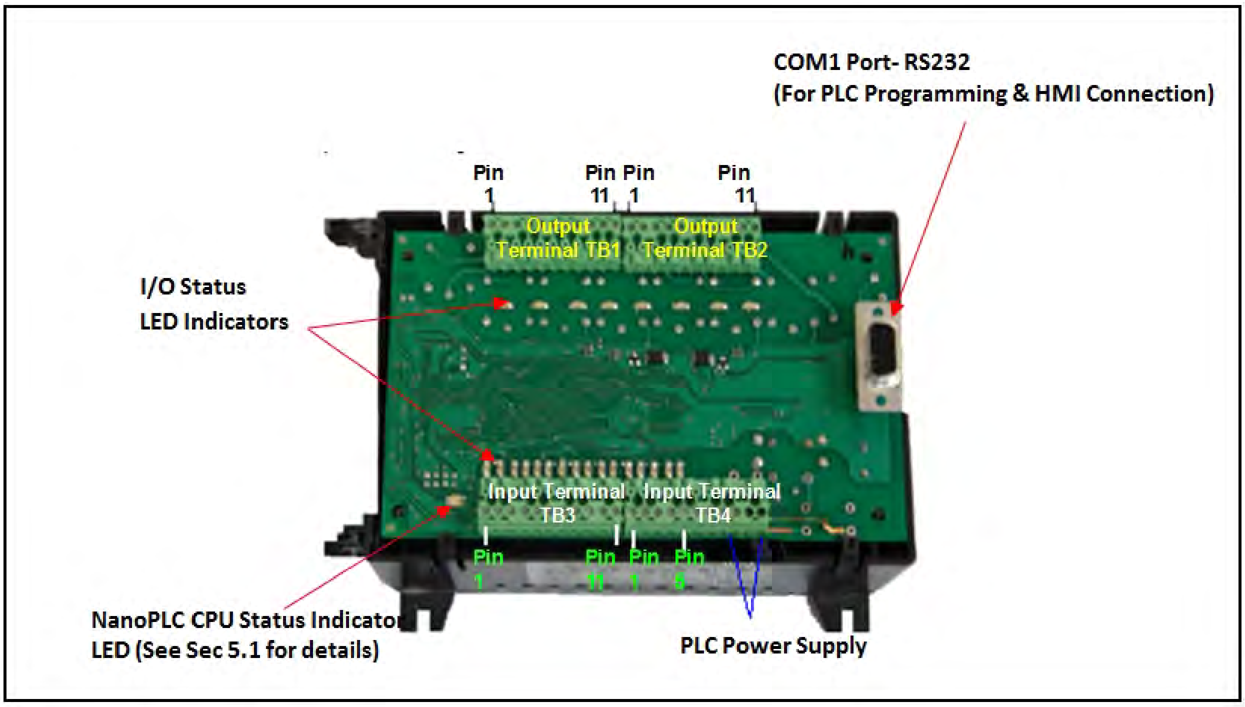
The NanoPLC comes with Easy to Wire Phoenix Terminals. As shown in the picture, simply insert the wire and screw to tighten. You can wire up to ONE 14 AWG wire, TWO 18 AWG wires, or FOUR 22 AWG wires in every terminal. You will need a 2.5mm blade screwdriver (P/N **EZIO-SCDRV**) to work with the I/O terminals and wiring.



Number of Wires Allowed in Each Terminal	
1	14 AWG
2	18 AWG
4	22 AWG

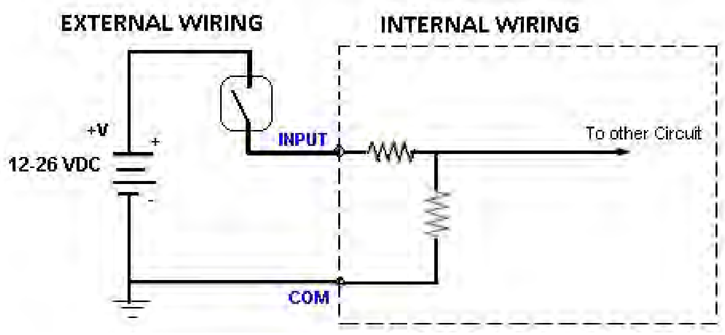
Wires Supported

UL rated at 300 volts, 10 amps 14 AWG



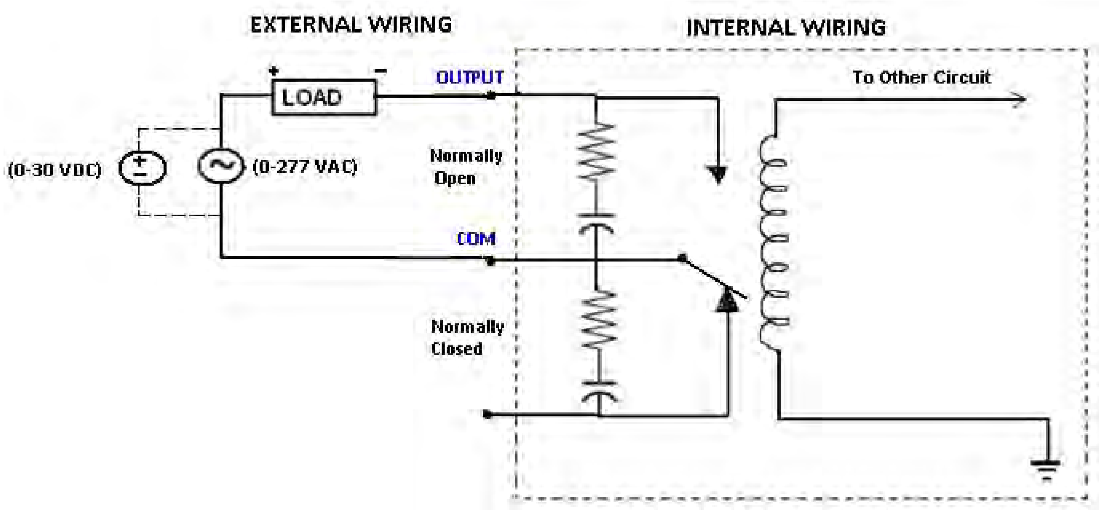
I/O Specifications

Discrete Input Specifications	
Number of Inputs	16
Input Voltage Range	12-26 VDC
Input Current	1.92 mA @12 VDC 4.0 mA @ 24VDC
Maximum Input Current	4.3 mA @ 26 VDC
Input Impedance	11.5k @ 12-26 VDC
ON Voltage Level	>12 VDC
OFF Voltage Level	<2 VDC
Min. ON Current	2mA
Min. OFF Current	0.2 mA
OFF-ON Response	2-4 ms. Typical 3 ms
ON to OFF Response	2-4 ms. Typical 3 ms
Status Indicators	Red LED for each input
Commons	2 points
Fuse	No Fuse
Wires	1 of 14 AWG, 2 of 18 AWG 4 of 22 AWG



Discrete Input Wiring

Relay Output Specifications	
Number of Outputs	8
Max Switching Voltage	277 VAC or 30 VDC
Max Switching Power	2770 VA or 300 W
Rated Switching Current	10A
Contact Resistance	100mΩ (at 1A 6VDC)
Insulation Resistance	100MΩ Min. at 500VDC
Dielectric Strength	1500VAC between Coil & Contact, for 60sec
	750VAC between Contacts for 60sec
Operate Time	Max 10ms
Release Time	Max 5ms
Status Indicators	Red LEDs



Relay Output Wiring

Using the NanoPLC

Quickstart the NanoPLC

This section outlines the steps needed to setup the NanoPLC and get it started. This is not intended to explain specific details needed to start your system. Rather, it provides a quick guide to give a broad picture of what is needed to power-up NanoPLC system.

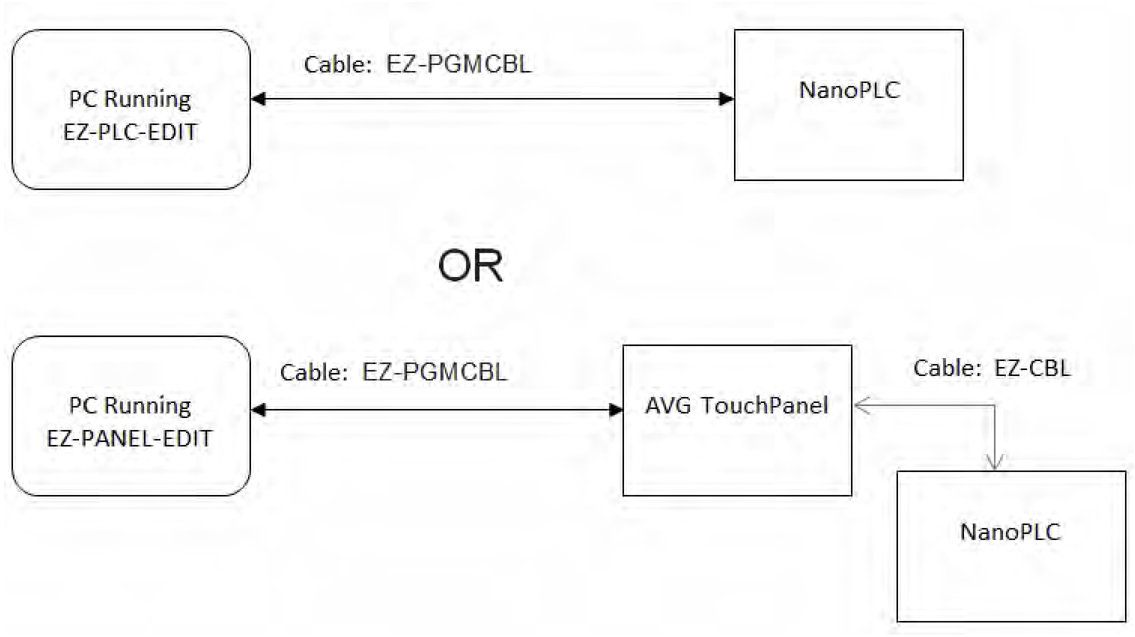
Step 1: Check all System Components
It is always recommended to make sure you have all the right parts to build your system. This is what you will need to get started:

- NanoPLC unit
- USB to serial port connector if the Programming Computer does not include a RS-232 port
- RS-232C Programming cable (P/N EZ-PGMCBL)*
- 2.5mm blade screwdriver for I/O wiring (P/N EZIO-SCDRV)*
- Programming Software
    - o EZPLC Editor (P/N EZPLC-EDIT) Programming Software*
    - o OR EZ Combined Panel/PLC Programming Software (EZ-PANEL-EDIT)*
- 24VDC Power Supply*

** These accessories have to be purchased separately*

Step 2: Install Programming Software on your PC and connect the PLC to PC port
Install the appropriate programming software (P/N **EZ-PLC-EDIT** or **EZ-PANEL-EDIT**) depending upon how you would like to program the PLC on your Personal Computer (PC).

Connect the PLC, PC and the Panel (if applicable) as shown below:

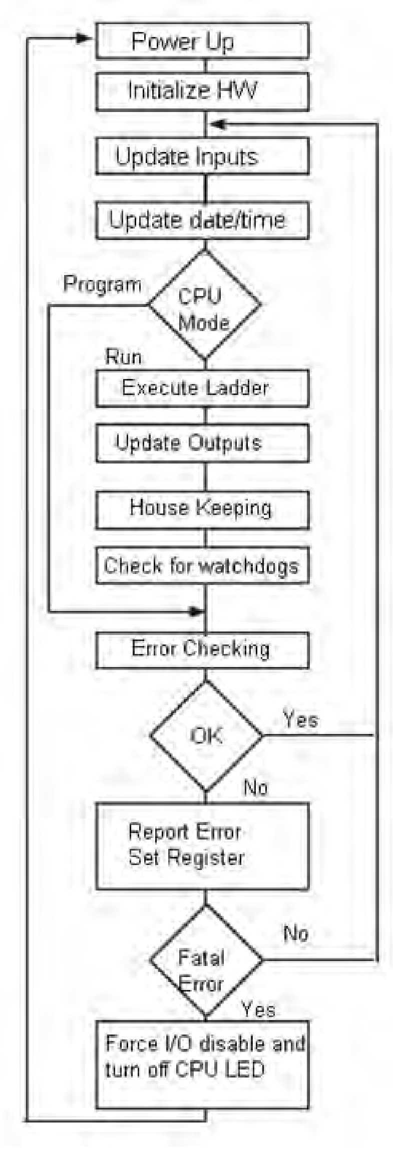


Step 3: (Optional) Wire Input / Outputs
You may wire required inputs and outputs now or later. Please refer to the wiring section for information on connecting the Input/Outputs to the NanoPLC terminals.

Step 4: Connect Power
If not already completed, please connect the power input wires to the NanoPLC's power terminals as outlined in the Powering the NanoPLC section. Supply 24VDC nominal (20-28VDC) power to the system based on the model type. Ensure the PWR indicator LED located on the PLC base is ON (blinking green LED). If not, remove power from the system and check all the wiring. More information on the CPU Status LED Indicator is also located in the Powering the NanoPLC section.

Step 5: Create and Transfer Program to PLC
Open your PLC Editor and create your PLC Ladder Logic. The EZPLC Software Manual explains the PLC instructions in detail. Use the Software manual and/or programming software help for information on programming the unit. A brief description of the process is available in the following section.

PLC Operation Sequence



A good understanding of the NanoPLC's CPU operating sequence will help you achieve the proper control for your equipment or process. The flow chart on the left shows how the CPU controls all aspects of system operation.

Power-up Initialization

On power-up, the CPU initializes the internal electronic hardware. It also checks if all the memories are intact and the system bus is operational. It sets up all the communication registers. If all registers are go, the CPU begins its cyclic scan activity as described below.

Read Inputs

The CPU reads the status of all inputs, and stores them in an image table. **Image Table** is NanoPLC's internal storage location where it stores all the values of inputs/outputs for ONE scan while it is executing ladder logic. The CPU uses this image table data when it solves the application logic program.

Execute Logic Time

This segment is also called Ladder Scan. The CPU evaluates and executes each instruction in the logic program during the ladder scan cycle. The rungs of a ladder program are made with instructions that define the relationship between system inputs and outputs. The CPU starts scanning the first rung of the ladder program, solving the instructions from left to right. It continues, rung by rung, until it solves the last rung in the Main logic. At this point, a new image table for the outputs is updated.

Write Outputs

After the CPU has solved the entire logic program, it updates the output image table. The contents of this output image table are written to the corresponding output points.

Immediate Inputs/Outputs

There is a possibility that an input changes after the CPU has read the inputs. If you have an application that cannot wait until the CPU returns for the next input scan, you can use **Immediate Instructions.** These instructions do not use the status of the input from the image table to solve the application program. The Immediate Instructions immediately read the input status directly from I/O modules and update the image table with appropriate status of input module read. Similarly, Immediate Output instructions do not wait for the CPU to complete the ladder scan. Immediate outputs are directly written to the image table and Outputs are updated accordingly.

Subroutines

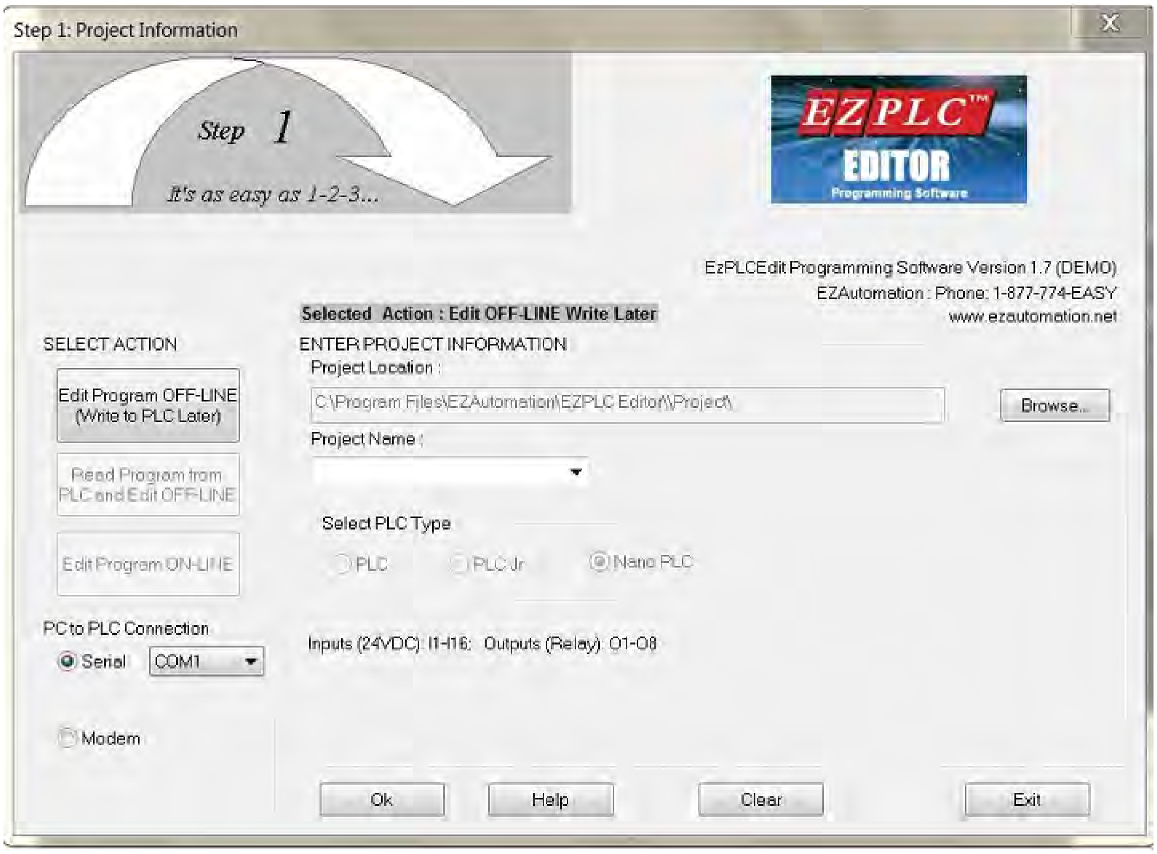
The CPU executes subroutines when called for in the ladder program. These subroutines are useful in performing the same logic operation time and time again just upon one call so you do not have to repeat the rung logic over and over again.

Programming the NanoPLC

Create a Project

This section outlines the basics of creating a project using the **EZPLC-EDIT** software. Further programming information for the NanoPLC is located in the **EZ Panel Enhanced Software Manual** and the **EZPLC Software Manual.** The NanoPLC uses a subset of the instructions described in the manual. For a complete listing of all RLL instructions supported by the NanoPLC please review the Appendix.

Launch your Programming Software and select how you would like the program to link to the NanoPLC unit. For this scenario, you can select 'Edit Program OFF-LINE.' This will enable you to create a program without having the NanoPLC unit connected through the serial port.


Step 1: Project Information
Step 1
It's as easy as 1-2-3...
EZPLC™ EDITOR Programming Software
Edit Program OFF-LINE (Write to PLC Later)
Read Program from PLC and Edit OFF-LINE
Edit Program ON-LINE
Browse...
Ok
Help
Clear
Exit

EzPLCEdit Programming Software Version 1.7 (DEMO)
EZAutomation : Phone: 1-877-774-EASY
www.ezautomation.net

Selected Action : Edit OFF-LINE Write Later

SELECT ACTION

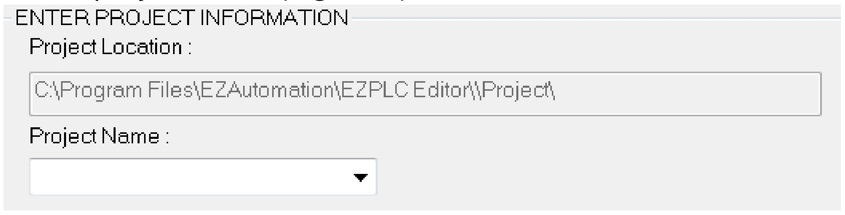
ENTER PROJECT INFORMATION
Project Location : C:\Program Files\EZAutomation\EZPLC Editor\\Project\
Project Name :

Select PLC Type
PLC   PLC Jr   [✓ Nano PLC]

PC to PLC Connection
[✓ Serial] COM1
Modem

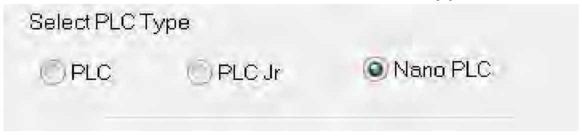
Inputs (24VDC): I1-I16; Outputs (Relay): O1-O8

1. Enter a project name (e.g. Test). Click OK.



ENTER PROJECT INFORMATION
Project Location : C:\Program Files\EZAutomation\EZPLC Editor\\Project\
Project Name :

2. Select NanoPLC from under PLC Type.

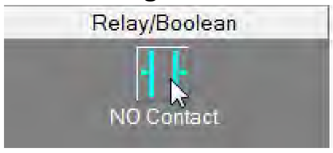


Select PLC Type
PLC   PLC Jr   [✓ Nano PLC]

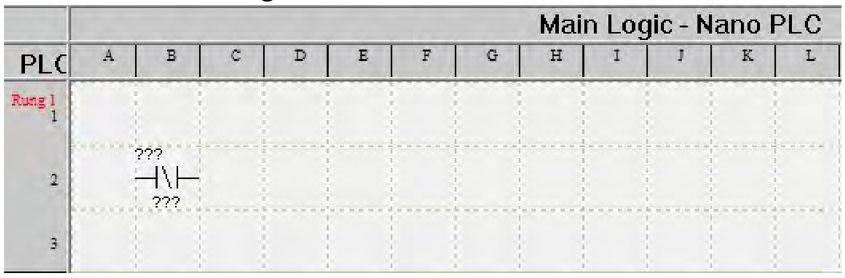
3. Click OK to launch the editing software program. The Main Project Window will then

appear. Follow the steps below to create a sample PLC Ladder Logic program.

a) Select the "Relay/Boolean" type instruction set in the instruction toolbar (located on the right side of the programming screen). Click on "NO Contact."

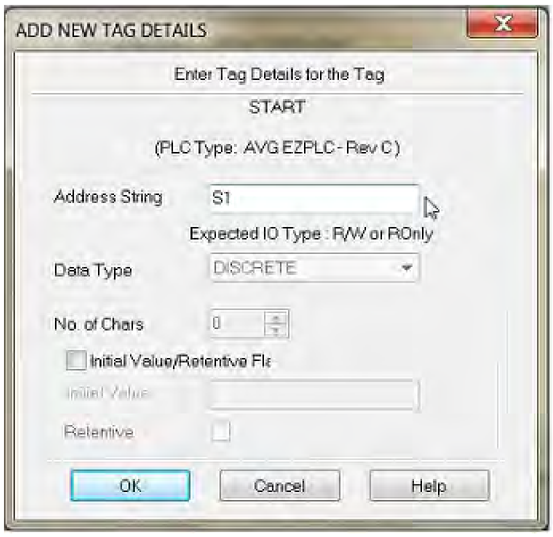


b) Click on the main ladder logic programming window to place the instruction as shown in the image below.



c) Once placed, double-click on the icon and enter the tag name as "Start". Click OK.

d) A new dialog box will appear asking for the PLC address (memory location). Enter "S1" in the field to the right of "Address String." The Data Type should be marked as DISCRETE. Click OK.



ADD NEW TAG DETAILS

Enter Tag Details for the Tag

START

(PLC Type: AVG EZPLC - Rev C)

Address String: S1

Expected IO Type : R/W or ROnly

Data Type: DISCRETE

No. of Chars: 0

Initial Value/Retentive Fl

Initial Value

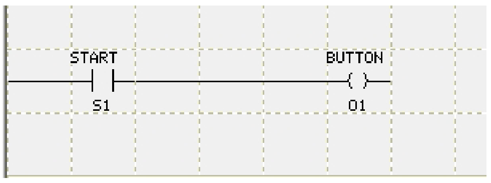
Retentive

e) Similarly, click on "NO Coil" under the Relay/Boolean instructions and place the instruction in the ladder logic programming window. Double-click the icon to select the tag name as "Button."

f) Once again, a dialog box will appear. Enter "O1" as the address string. Click OK.

g) Click on **Instructions** > **Line** to wire "NO Contact" and "NO Coil."

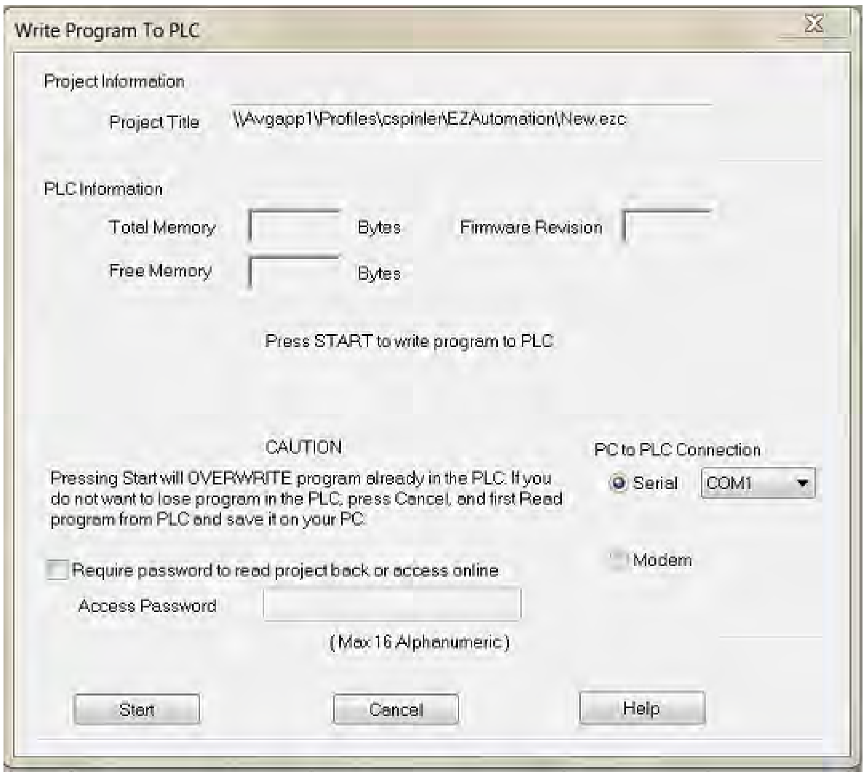
Your screen should look like this when finished:



Transfer a Project

After a project is complete, the next step is to transfer the project to the NanoPLC. When editing projects online, programming information is automatically sent to the NanoPLC unit once the project is saved. When editing in an off-line mode, the project information will need to be transferred. To transfer the project follow the steps outlined below:

From the Project drop down menu, select **File** > **Transfer to Panel**. A dialog box similar to the one below will appear.



Write Program To PLC

Project Information

Project Title \\Avgapp1\Profiles\cspinler\EZAutomation\New.ezc

PLC Information

Total Memory Bytes Firmware Revision

Free Memory Bytes

Press START to write program to PLC

CAUTION

Pressing Start will OVERWRITE program already in the PLC. If you do not want to lose program in the PLC, press Cancel, and first Read program from PLC and save it on your PC.

Require password to read project back or access online

Access Password

(Max 16 Alphanumeric)

PC to PLC Connection

● Serial COM1

Modem

1. Verify the RS-232C cable (**P/N: EZ-PGMCBL**) is connected between the unit and the PC. In the absence of an RS-232 port on the PC, a USB to RS-232 converter may be used to connect the programming cable to the PC.
2. Select *Serial (COM1)* as method of transfer under PC to PLC Connection. And then click *Start*.

When finished, a Transfer Completed message will be displayed. Click OK to continue and the project is now transferred.

High Speed Counter

Overview

The NanoPLC possess the following new High Speed Counter feature. High speed counting allows either two 32-bit up counters, or one 32-bit up-down counter, or one 32-bit quadrature encoder. These features involve use of newly added SR registers and SD bits.

The following table summarizes the availability of these features for the NanoPLC:

Feature	Nano
Program Loader	PLC Editor 1.8.6, uwin2.0.11, ezpanel 5.10.11
Firmware	c.3.75 or above
Counter function	
Up Counter	2x32 bit (Counter A Input on I15 Counter B Input on I16)
Up/Down Counter	1x32 Bit (Counter A Input on I15 Direction Input on I16)
Quadrature Encoder	1x32 bit Quadrature A on I15 Quadrature B on I16 (Increasing counts when A leads B)

Counter Input Terminals

The counter inputs terminals are fast inputs (very low filtering) allowing us to count up to 40 KHz. The inputs, if not used as counter inputs, can also be used as normal inputs. However these inputs would have lower filter compared to other inputs.

The terminal numbers are shown in the table above.

Maximum Count Frequency

The NanoPLC can count up to 40KHz.

SR Registers for high speed counter function

Following table describes the registers used for High Speed counting function:

Register	Description
SR21	Counter configuration word
SR22	Reserved (Do not use)
SR23-SR24	Lower Limit of Quadrature Range (signed 32-bit number)
SR25-SR26	(Upper Limit+1) of Quadrature Range (signed 32 bit number)
SR27-SR28	Counter A Counts (signed 32 bit number)
SR29-SR30	Counter B Counts (signed 32 bit number)
SR31-SR32	Setpoint 1 for Counter A (signed 32 bit number)
SR33-SR34	Setpoint 2 for Counter B (signed 32 bit number)

Lower and Upper Limits:
User programs these registers as per their requirements. As an example, if you would like to count from 0 to 359, set lower limit register to 0, and upper limit register to 360 (i.e. 359+1).

- When count value equals the Lower Limit, the next input to decrease the counts would rollover the count value to Upper Limit value.
- When count value equals to the Upper Limit, the next input to increase the count would roll over the count value to Lower Limit.

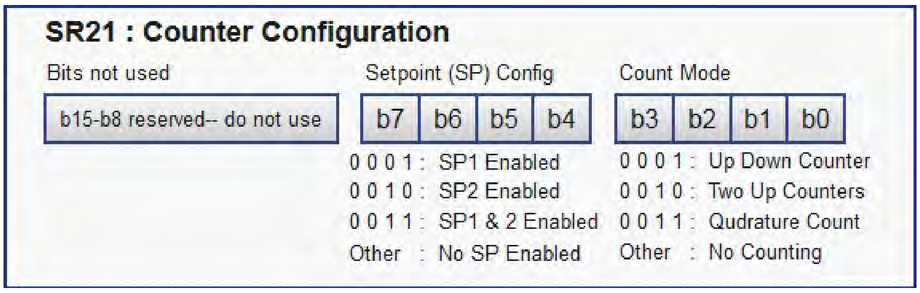
SD bits for high speed counter function

Following SD bits have been used for counter functions:

Bit	Description
SD15	Pause Counter A The Counting for Counter A is paused as long as SD15 is 1.
SD16	Pause Counter B The Counting for Counter B is paused as long as SD16 is 1.
SD17	Reset Counter A Counter A is reset to 0 when SD17 is 1.
SD18	Reset Counter B Counter B is reset to 0 when SD18 is 1.
SD19	Setpoint 1 Match Bit: The Bit is set whenever Counter A is greater than or equal to Setpoint 1. Whenever the bit goes from 0 to 1 (clear to set), **the PLC executes logic programmed in the interrupt routine.**
SD20	Setpoint 2 Match Bit: The Bit is set whenever Counter A is greater than or equal to Setpoint 2. Whenever the bit goes from 0 to 1 (clear to set), **the PLC executes logic programmed in the interrupt routine.**

Counter Configuration

The counting operation is configured using SR21 register.


SR21 : Counter Configuration

Bits not used	Setpoint (SP) Config	Count Mode
b15-b8 reserved-- do not use	b7 b6 b5 b4	b3 b2 b1 b0
	0 0 0 1 : SP1 Enabled	0 0 0 1 : Up Down Counter
	0 0 1 0 : SP2 Enabled	0 0 1 0 : Two Up Counters
	0 0 1 1 : SP1 & 2 Enabled	0 0 1 1 : Qudrature Count
	Other : No SP Enabled	Other : No Counting

Lower 4 bits (bits 0-3) are used to configure count mode as follows:

Counter configuration bits Lower 4 bits (bits 3-0) of SR21	Counter function
1 (0001 binary)	Up down counter (Counter A) For Nano: Count input on I15, Dir on I16 For Micro: Count input on I23, Dir on I24
2 (0010 binary)	Two up counters : (Counter A, B) For Nano: Counter A input on I15, Counter B input on I16

	For Micro: Counter A input on I23, Counter B input on I24.
3 (0011 binary)	One quadrature counter: (Counter A) For Nano: Quadrature A input on I15, Quadrature B input on I16 For Micro: Quadrature A input on I23, Quadrature B input on I24 (The PLC counts ALL the four edges of the quadrature signal; Count increases when Quadrature A leads B)
Other values including 0	No Counting

Next 4 bits (b7-b4) are used for setpoint configuration:

Setpoint Configuration bits Bits 7-4 of SR21	**Setpoint Configuration (Only counter A can use Setpoints)**
1 (0001 binary)	Setpoint 1 (SR31-32) enabled
2 (0010 Binary)	Setpoint 2 (SR33-34) enabled
3 (0011 binary)	Both Setpoint enabled
Other values including 0	No Setpoint Matches enabled

Example: If you want to use a quadrature counter with setpoint 1 enabled, set SR21 to 16 x1 + 3 = 19 (decimal) or 0x13 (hex). Multiplying Setpoint configuration value by 16 moves the bits to their right positions (bits 7-4) in configuration register.

Counter Upper and Lower Limits

You can put lower and upper limits on the count values. (see SR23-24, and SR25-26). Note that limits are 32 bit values.

- When count value equals the Lower Limit, the next input to decrease the counts would rollover the count value to Upper Limit value.
- When count value equals to the Upper Limit, the next input to increase the count would roll over the count value to Lower Limit.

Program these registers as per your requirements. As an example, if you would like that count value be from 0 to 359, set lower limit register to 0, and upper limit register to 360 (i.e. 359+1).

CPU Memory

Memory Types

A PLC system handles many numbers representing different types of information regarding processes/machine parameters. These processes/machine parameters may be anything from status of the input or output devices, timers/counters, or other data values. Before you start programming the NanoPLC, it would be helpful if you took a moment to familiarize yourself with how the system represents and stores the various types of data. Each PLC manufacturer has their own conventions for this process in their PLCs.

The Mapping Conventions section outlines the specific memory types used in the NanoPLC in greater detail. The NanoPLC supports the same instructions as the ones handled by the standard TouchPLC with a few minor exceptions. Refer to the Appendix section for the complete set of the available instructions. The memory types can be used to store a variety of information and can be used within various RLL instructions. See a description of each of the memory types below:

- **Discrete Memory Type**
  A discrete memory type is one bit that can be either a 1 or a 0 (On or Off). Discrete memory area is used for inputs, outputs, control relays, and timer/counter bits.
- **Word Memory Type**
  A word memory type is a 16-bit location that is normally used to store and manipulate numeric or ASCII data. A word memory location is also called a Register.

Since the NanoPLC relies on flash memory, the only values saved during a power loss are the ones associated with the registers/discretes that are retained during a power cycle. The specific registers/discretes available on a power cycle are listed in the table below. The following table also displays all the Register/ Discrete types supported by the NanoPLC along with their address range, syntax etc.

NanoPLC Memory Map					
SYNTAX: TAAAA **T - TYPE** **AAAA - Address of Memory Type in Decimal**					
MEMORY TYPE	**ADDRESS RANGE**	**I/O TYPE**	**VALUE TYPE**	**SYNTAX EXAMPLES**	**DISCRETES / REGISTERS RETAINED ON POWER CYCLE**
I- Discrete Inputs	1-128	READ_ONLY	DISCRETE	I5	**NONE**
O- Discrete Outputs	1-128	READ_WRITE	DISCRETE	O6	**O1- O32**
S- Discrete Internals	1-1024	READ_WRITE	DISCRETE	S4	**S1- S128**
IR- Input Register	1-64	READ_ONLY	WORD	1R3	**NONE**
OR- Output Register	1-64	READ_WRITE	WORD	OR2	**OR1 - OR64**
R- Register Internals	1-8192	READ_WRITE	WORD	R100	**R1 – R256**
SR- System Registers	1-64	READ_WRITE	WORD	SR1	**SR1 –SR64**
SD- System Discrete	1-64	READ_WRITE	DISCRETE	SD10	**SD1 –SD64**
XR- Index Registers	1-4	READ_WRITE	WORD	XR10	**NONE**
#R – Value Registers	1-4	READ_WRITE	WORD	#R6	**#R1 - #R4**
Note: Does not Support Access to a Bit of Word (E.g.: R100/ 0, R100/5…etc)					

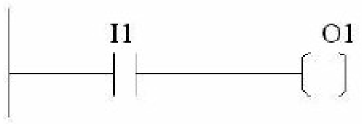
Please Note: Since the PLC Editor is a common programming platform for all the models offered by the AVG PLC family, it may allow you to include 128 Inputs (I), 128 Output (O), 64 input Registers (IR) and 64 Output Registers (OR) in the main logic. However, the NanoPLC only physically supports 16 inputs and 8 outputs; hence it is recommended that you only use I1- I16 and O1-O8 while programming the NanoPLC. The remaining O bits may be used as “Scratch bits”. Similarly, only IR1-IR4 and OR1- OR4 should be used to address the I/O Registers, while the rest of the Output Registers may be used as “Scratch Registers”. Although there are 64 System Registers (SR) and 64 System Discretes (SD) available in the programming software, some of them are preassigned a function.

Mapping Conventions

Discrete Inputs/Outputs

Discrete Inputs

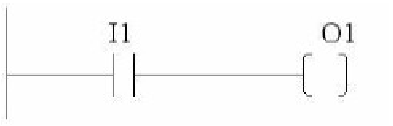
Discrete Inputs are denoted using an “I” pre-fix (e.g. I1, I4, etc.). The maximum number of physical Inputs available in an NanoPLC is 16. Hence, you may only use I1 – I16 in your main logic. Discrete inputs are Read only type.



Note: All the discrete type EZ Inputs are mapped to Discrete Input bits. In the example above, the output bit O1 will be turned on when input I1 allows power through the rung.

Discrete Outputs

Discrete Outputs are denoted using an “O” pre-fix (e.g. O1, O4, etc.). The maximum number of programmable Outputs available is 1 through 128. Although, the number of discrete physical outputs available in an NanoPLC is 8, the remaining “O” registers can be used as ‘Scratchbits' in the main logic. Discrete Outputs are Read-Write type.



Note: All the discrete type EZ Outputs are mapped to Discrete Outputs bits.

Word Inputs/Outputs

Input Register (Word)
Input Registers are denoted using an “IR” pre-fix (e.g. IR1, IR4, etc.). These are 16-bit Word data types (registers). The maximum number of Input Registers available is 1 through 64. You can only Read from an IR register.

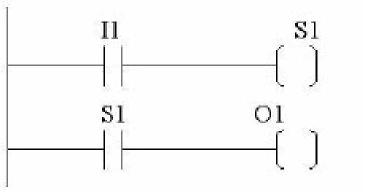
Note: All the EZ Analog Inputs (if available) are mapped to Input Registers.

Output Register (Word)
Output Registers are denoted using an “OR” pre-fix (e.g. OR1, OR4, etc.). These are 16-bit Word data types. The maximum number of Output Registers available is 1 through 64. OR are Read-Write type of Word registers.

Note: All the EZ Analog Outputs (if available) are mapped to Output Registers.

Internals

Discrete Internals (Discrete)
Discrete Internals are denoted using “S” pre-fix (e.g. S1, S4, etc.). There are 1024 Discrete Internals available in the NanoPLC Discrete Internals are Read-write type and are used as “Scratchbits”. Discrete internal bits are mainly used to control the user logic program. They do not represent a real physical device, like a switch, output coil, etc. They are only internal to the CPU. You cannot program discrete internals as discrete inputs or discrete outputs for the physical inputs or outputs.



Note: In this example, memory location S1 will be powered when input I1 turns on; you can then use a discrete internal as an input in another rung.

Register Internals (Word)
Internal Registers are denoted using an “R” pre-fix (e.g. R1, R4, etc.). These are 16-bit Word data types (registers). There are 8192 Internal Registers available in the NanoPLC R are Read-Write type of data registers.

System

System Discretes (Discrete)
System Discretes are denoted using an "SD" prefix (e.g. SD1, SD4, etc.). SDs are discrete memory locations with pre-assigned functionality. There are many different types of System Discretes. They are used to help in logic program development, provide system operating status info and more.

System Registers (Word)
System Registers are denoted using an "SR" prefix (e.g. SR1, SR4, etc.). These are 16-bit Word data types (registers). System registers are Read-Write type data points.

Index and Value Registers (Word)

The Index Register data type is represented by an "XR" prefix (e.g. XR1, XR2 etc.). There are 4 XR memory locations available in NanoPLC "XR" is a Read-Write data type and it is mainly used to point to the correct address of "R" registers. The pointed-to "R" registers data value is stored in "#R" registers.

Value Register data type is represented by a "#R" prefix (e.g. #R1, #R2 etc.). There are 4 #R memory locations available in NanoPLC "#R" is a Read-Write data type and it is mainly used to read/write value of "R" registers as pointed out by "XR" registers.

Both XR and #R registers are used in conjunction with each other and provide a convenient way of addressing R registers.

Example:

Let's assume data values: R59=9874, R8000=32

If XR1=59

Then #R1=9874 (the actual data value of R59)

If XR2=8000

Then #R2=32 (the actual data value of R8000)

XR contains the address of the operand (or specifies a register that contains the effective address), #R is used to read or write the actual operand. Indirect addressing is often combined with pre- or post-increment (or decrement) addressing. This allows the address of the operand to be increased or decreased by the specified number either before or after using it. Proper usage of XR variables often saves a lot of programming.

Maintenance and Troubleshooting

Hardware Maintenance

Routine maintenance checks should be performed on the unit to avoid any risk of hardware problems. The NanoPLC is designed to be a very rugged controller so that just a few checks periodically will help keep it up and running.

The key points to be checked include:

· Ambient operating conditions
· Wiring and connections

Maintaining the Ambient Operating Conditions

Keeping the NanoPLC's environment within specified operating conditions is the best method to minimize the maintenance.

1. Always ensure that ambient temperature inside the cabinet is within NanoPLC's temperature ratings.
2. If any other equipment inside or outside of the cabinet is producing heat, employ cooling methods like a blower fan to reduce 'hot spots' around the NanoPLC.
3. Periodically inspect and clean if there are any air filters on the cabinet. Ensure that the PLC is free from dust, humidity and corrosive gases.

Error Checking Process

The NanoPLC performs a standard diagnostic routine during each CPU scan. This is called the error-checking step. The primary task of this step is to identify various types of CPU and I/O failures. We classify these errors/failures broadly into two categories: Fatal and Non-Fatal.

Fatal Errors

These errors are the ones that lead to the system failure. During the CPU scan if a fatal error is detected, PLC is automatically switched out of Run mode and all I/O points are disabled. Some instances of fatal errors include: Wrong parity value, Programming errors, etc. The NanoPLC will not go into Run mode from Program if it detects a fatal error.

Non-Fatal Errors

These errors just need your attention and are not detrimental to PLC operation. Unlike fatal errors, the PLC will continue in Run mode despite an occurrence of non-fatal errors. When you identify such errors, you can proceed with an orderly shutdown and take the required corrective action. An example of non-fatal error is – a minor programming error.

Troubleshooting

If you encounter difficulties while using our NanoPLC device, please consult the table below. Additional assistance is also available within the **PLC Editor Programming Software Help**. Alternatively, you may also find answers to your questions in the operator interface section of our website @ flash.ezautomation.net.

Problem		Possible Cause	Suggested Action
Operation	CPU Status LED is off	Disconnected or faulty power source	Check and repair power source.
			Check the wiring for loose contacts and secure them if found.
			(for 24 VDC powered NanoPLC) Ensure that proper polarity is observed.
		Input power level is outside of NanoPLC's power rating specifications	Ensure that the power being presented to the PLC terminal is within the specified range.
	CPU LED is blinking red and green	Bad or corrupted program	Check the logic program
			Pay special attention to Program Control Instructions and make sure there is a Next or Return statement at the end of Jump and Subroutine Instructions
	CPU LED is red	Electrical Noise	Power cycle the PLC once to see if an intermittent high frequency noise has caused the failure.
			Follow instructions to avoid electrical noise.
			Consider installing an Isolation Transformer if you think the noise is making its way through the Power source.
			Check to ensure that RS232 signal GND is not connected to Earth

			ground and the shield is connected to Earth ground on both sides.
			If problem persists, call AVG Automation for assistance.
Communication	No communication with NanoPLC	Disconnected or loose cable	Check the wiring for loose contacts and secure them if found.
			Ensure you are using a correct communication cable.
	No communication with the PC (RS232 Port error)	Wrong/broken cable	Ensure the correct communication cable is being used (PGMCBL).
		Wrong communication port settings	Check and correct the COM port attributes.
			Open the PLC Editor and click on the configuration button
		Wrong COM port assignment on the computer	Check if correct Serial Port (COM1) of the computer has been selected.

Still Need Help?

Technical Support
Most of the frequently encountered problems regarding the NanoPLC unit's operation are answered in the sections above. However, if you still need answers to your questions, please call our technical support at 1-877-774-EASY.

Warranty Repairs
If your NanoPLC is under warranty, contact us at 1-877-774-EASY.

Out of Warranty Services
If your NanoPLC is out of warranty, contact EZ Automation at 1-877-774-EASY for an evaluation of repair costs. You can then decide whether it is more economical to proceed with the repairs or to upgrade your system with a new unit.

Appendix

The following table provides a quick reference to all RLL instructions supported by the NanoPLC, as well as a brief description regarding the function of each instruction.

Instruction	Description
Relay/Boolean Instructions	
NO Contact	When the corresponding memory bit is a 1 (on) it will allow power flow through this element
NC Contact	When the corresponding memory bit is a 0 (off) it will allow power flow through this element
Positive Transition	If the corresponding bit has changed from 0 (off) to 1 (on) in the current scan, power flows through this element
Negative Transition	If the corresponding bit has changed from 1 (on) to 0 (off) in the current scan, power flows through this element
NO Coil	As long as the power flows to the instruction, corresponding memory bit remains 1 (on)
NC Coil	As long as the power flows to the instruction, corresponding bit to remains 0 (off)
Set Coil	When power flows to the instructions, corresponding bit is set to 1 (on) and remains 1 (on) even if the rung condition goes to false (Use RESET COIL instruction to turn the corresponding bit Off.)
Reset Coil	When power flows to the instructions, corresponding bit is set to 0 (off) and remains 0 (off) even if the rung condition becomes false (Use SET COIL instruction to turn the corresponding bit On.)
NO Immediate Input	PLC reads the addressed bit immediately from the input module (instead of memory). The power flows through the instruction if the read bit is 1 (on). (Please note all the bits corresponding to the input module are updated with the read value.)
NC Immediate Input	PLC reads the addressed bit immediately from the input module (instead of memory). The power flows through the instruction if the read bit is 0 (off). (Please note all the bits corresponding to the input module are updated with the read value.)
NO Immediate Output	The bit status is immediately written to corresponding output module. The bit remains 1 (on) as long as the power flows to the instruction
NC Immediate Output	The bit status is immediately written to

	corresponding output module. The bit remains 0 (off) as long as the power flows to the instruction
Compare Instructions	
Equal To	Allows power flow through this element if the data value of “Opr1” register is Equal to “Opr2” register
Not Equal to	Allows power flow through this element if the data value of “Opr1” register is NOT Equal to “Opr2” register
Greater Than	Allows power flow through this element if the data value of “Opr1” register is Greater Than “Opr2” register
Less Than	Allows power flow through this element if the data value of “Opr1” register is Less Than “Opr2” register
Greater Than or Equal To	Allows power flow through this element if the data value of “Opr1” register is Greater Than or Equal to “Opr2” register
Less Than or Equal to	Allows power flow through this element if the data value of “Opr1” register is Less Than or Equal to “Opr2” register
Limit	Allows power flow through this element if the data value of “Input” register is within the data values of “High Limit” and “Low Limit” registers
Math Instructions	
Add	Adds two data values in “Opr1” and “Opr2” registers and stores the result in “Result” register
Subtract	Subtracts “Opr2” register data value from “Opr1” register data value and stores the result in “Result” register
Multiply	Multiplies two data values in “Opr1” and “Opr2” registers and stores the result in “Result” register
Divide	Divides “Opr1” register data value by “Opr2” register data value and stores the result in “Result” register
Modulo	Divides “Opr1” register data value by “Opr2” register data value and stores only the remainder in “Result” register
Absolute	Converts a negative data value from “Opr1” register to a positive value and stores it in “Result” register
X=Y Conversion	Copies the data value of “Opr” register, converts it into “Result” registers data type, and stores the data value in “Result” register
Binary Conversion	Converts the data value of “Source” register in

	Binary, BCD, or GRAY code to the data value of “Result” register in Binary, BCD or GRAY Code
Bitwise Instructions	
AND	Performs a bitwise AND operation between the data values of two registers “Opr1” and “Opr2”. The result is stored in “Result” register
OR	Performs a bitwise OR operation between the data values of two registers “Opr1” and “Opr2”. The result is stored in “Result” register
XOR	Performs a bitwise XOR operation between the data values of two registers “Opr1” and “Opr2”. The result is stored in “Result” register
NOT	Performs a bitwise NOT operation on the data value of “Source” register and stores the result in “Destination” register
Shift Left	Performs a logical Shift Left on the data value of “Opr1” register by the data value of “Opr2” register and stores the result in “Result” register
Shift Right	Performs a logical Shift Right on the data value of “Opr1” register by the data value of “Opr2” register and stores the result in “Result” register
Rotate Left	Performs a logical Rotate Left on the data value of “Opr1” register by the data value of “Opr2” register and stores the result in “Result” register
Rotate Right	Performs a logical Rotate Right on the data value of “Opr1” register by the value of “Opr2” register and stores the result in “Result” register
Move Instructions	
Move Data	Moves data value of “Source” register to “Destination” register
Bit Move	Moves either words to bits or bits to words with user-specified length for the number of words to move. Maximum of 16 words can be moved at a time
Move Block	Moves a block of memory area. “Source” register defines the starting area of memory address/ register to Move from and “Destination” register defines the starting area of memory address/ register to move to. The number of elements to move is user defined
Block Fill	Fills a block of memory area. “Source” register defines the data value to Fill with and “Destination” register defines the starting area of

	memory address/register to Fill to. The number of elements to move is user defined. The number of elements to Fill is user defined
Move Table of Constants	Loads a table of user defined constants to a consecutive memory/register locations with the starting memory address/register location defined by “Destination” register
Timer/Counter Instructions	
Timer	This instruction starts timing when called and once it reaches the preset value as defined by the data value of “Timer Preset Value” register, it will stop timing and will allow power flow through the element
Counter	This instruction starts counting either Up or Down by the increments of one until the counter reaches the data value of “Counter Preset Value” register. The Counter will then allow power flow through the element
Program Control	
Jump	Skips the rung containing Jump instruction (after execution of the rung) to a rung with the label specified in the JUMP instruction and continues executing the program thereafter
For Loop	Executes the logic between the FOR Loop and NEXT instructions by the data value of “Loop Count” register
Next Statement	Specifies the return/end point for the FOR Loop instruction
Call Subroutine	Calls a Subroutine specified by the label in CALL Subroutine instruction and is terminated by the RETURN instruction
Return	Terminates a subroutine and returns back to the main logic
String Instructions	
String Move	Moves the data value (string type) of “Source” register to “Destination” register by the number of characters specified by the user
String Compare	Allows power flow through this element if the data value (string type) of “Source1” register is Equal to “Source2” register by the number of characters specified
String Length	Computes the length of a null-terminated “String”

	register (string type) and stores the result in “Save Length in” register
Communication	
Open Port	Opens the serial port for communication using the parameters specified by the user
Send to Serial Port	Sends an ASCII string data from "Source" register to the serial port with control and character count from user defined "Control Address" and "Character Count Address" registers respectively
Receive From Serial Port	Receives an ASCII string data from serial port to "Source" register with control and character count from user defined "Control Address" and "Character Count Address" registers respectively
Close Port	Closes the serial port opened for communication
Send to Marquee	Sends an ASCII instructions for marquee communication. The message to be displayed on a marquee is selected by the data value of "Message Number" register which looks up the message number for a corresponding message from the central message database. If message number is not found in the message database, user selected action for unmatched messages is performed
Modbus Master	Sends a command to an addressed slave unit and processes the reply back from the slave. Power flows out of the instruction only after the instruction is completed, i.e. after either the reply is received (success) or the instruction times out/ generates an error message (unsuccessful)
Miscellaneous Instructions	
Drum	Time and/or Event driven drum type sequencer with up to 16 steps and 16 discrete outputs per step. The outputs are updated during each step. Counts have a specified time base (1MSec to 1 Sec) and every step has its own counter along with an event to trigger the count. After the time expires for one step, it transitions to the next step and completes up to 16 steps total. After the completion of all the steps this element allows power flow through it