



Power Meter Monitor

Business and Mission-

Critical Solutions Provider

(12 – 60) x Digital Input 24VDC
Stand Alone – Mini PLC

Data Sheet



Model: PMM0620-024

Document: Data Sheet

Document version: 1.1

Date: January 2023



COPYRIGHT NOTICE

The information in this document is subject to change without prior notice to improve reliability, design, and function and does not represent a commitment on the part of the manufacturer.

In no event will the manufacturer be liable for direct, indirect, special, incidental, or consequential damage arising out of the use or inability to use the product or documentation, even if advised of the possibility of such damages.

This document contains proprietary information protected by copyright. All rights are reserved. No part of this manual may be reproduced by any mechanical, electronic, or other means in any form without prior written permission of the manufacturer.

TRADEMARKS

All registered trademarks and product names mentioned herein are used for identification purposes only and may be trademarks and/or registered trademarks of their respective owners.

AMI is a trademark of American Megatrends Inc.

Intel & Atom are trademarks of Intel Corporation

IBM, PC/AT, PS/2&VGA are trademarks of International Business Machines Corporation
Microsoft Windows is a trademark of Microsoft Corp. RTL is the trademark of Realtek Semiconductor Co., Ltd.

DECLARATION OF CONFORMITY

This restriction is subject to protect the operational process of the system in the business environment, which will produce, use, and transmit radiofrequency energy. Harmful interference to radio communication could result if instructions to the correct installation and usage were not applied. The interference prevention cannot be guaranteed even with proper installation according to the manual. If the device causes a bad effect on the radio / TV signal. The user could preclude that by turning the device on/off.

When this device produces some harmful interference, the user can use the following measure to solve the interference problem:

1-Setting the receiving antenna's direction or location to increase the distance between this device and receiver.

2-Plug in the device's power connector into different circuits of the power outlet with the receiver.

3-If any technical support is needed, the dealer or experienced radio/TV technical personnel must be informed.

TECHNICAL SUPPORT AND SERVICE

Visit Pmm-usa.us to browse FAQs and get further details.

User should collect the following information before submitting technical support and service requests:

- Product name, model and serial number.
- Installed software (operating system, OS version, installed applications and so on).
- Full description of the problem
- Detailed information about every error.

SAFETY INSTRUCTIONS

- Only trained and qualified personnel can install, operate, or maintain the device.
- Before starting the installation, all safety precautions must be read, and warning labels affixed to the device must be observed. Doing so protects the device from damage and ensures your safety.
- Safety precautions provided in this document may not cover all safety aspects, note to always remain mindful of safety.
- PMM is not liable for any consequence that results from violation of regulations pertaining to safe operations or safety codes pertaining to design, production, and equipment usage.
- DO NOT use liquids or decontamination spray to clean the device surface and assure that it is totally disconnected while cleaning.
- Take all measures to prevent device drop before or during installation.
- Prior to connecting the device to power source, ensure the source and device voltage and power are 100% matched.
- Keep the cables in a suitable covered place.
- If the device is not used for a long time, shut off the power to avoid the damages by transient overvoltage.
- DO NOT allow any liquid flow into the device; to avoid fire or short circuit.
- The recommended storage temperature range should NOT be less than 30°C OR higher than 85°C.

Warning:

- Read the power source and device inlet carefully.
- Handle device with both hands.
- Clean and maintain the device using recommended, safe and suitable methods.

Caution:

If any unauthorized changes of settings or repairs are done without PMM approval; then user's rights of controlling this device will be canceled.

Contents

- DESCRIPTION6
- KEY FEATURES.....6
- TECHNICAL SPECIFICATIONS.....7
- PINS ASSIGNMENTS7
- HARDWARE CONFIGURATION8
- HARDWARE CONNECTIONS.....9
 - 1. Connecting Power.....9
 - 2. Connecting Serial Device9
 - 3. Connecting to a Host or the Network.....9
 - 4. Connecting Digital Input10
- BOARD EXTENSIONS11
 - 1. 0620-024 X (16 Digital Input).....11
 - Description.....11
 - Key Features11
- PINS ASSIGNMENTS12
- HARDWARE CONFIGURATION12
- HARDWARE CONNECTIONS.....13
 - 1. Connecting Power.....13
 - 2. Connecting Serial Device14
 - 5. Connecting Digital Input14
 - 2. 0620-024 XP (24 Digital Input).....15
 - Description.....15
 - Key Features15
- PINS ASSIGNMENTS16
- HARDWARE CONNECTIONS.....16
 - Connecting Digital Input16
- ORDERING INFORMATION.....17
- Appendices18
 - Appendix 1:.....18
 - PMM USB Configuration.....18
 - 1.1. Description.....18
 - 1.2. Key Features18
 - 1.3. Set Up Instructions18
- Ethernet Settings23
- Modbus Settings23
- Serial Settings24



Appendix 2:.....	25
PMM06 Integration with Arduino IDE	25
2.1 Description.....	25
2.2 Integration Guidelines	25
Appendix 3:.....	28
PMM06 Integration with Visuino	28
3.1 Description.....	28
3.2 INTEGRATION GUIDELINES	28
3.2.1 Visuino Installation Guidelines	28
3.2.2 Integration Guidelines	31
Appendix 4:.....	34
PMM06 Integration with Node-RED.....	34
4.1 Description.....	34
4.2 INTEGRATION GUIDELINES	34
4.2.1 Node-RED Installation Guidelines.....	34
4.2.2 Integration Guidelines	37
Appendix 5:.....	43
PMM06 Integration with OpenPLC.....	43
1.1 Description.....	43
5.2 INTEGRATION GUIDELINES	43
5.2.1 OpenPLC Installation Guidelines	43
5.2.2 Integration Guidelines	45
Appendix 6:.....	50
PMM06 Integration with Scilab.....	50
7.1 Description.....	50
6.2 INTEGRATION GUIDELINES	50
6.2.1 Scilab Installation Guidelines.....	50
6.2.2 Integration Guidelines	54
Appendix 7:.....	60
PMM06 Integration with PlatformIO	60
7.2 Description.....	60
7.2 INTEGRATION GUIDELINES	60
7.2.1 PlatformIO Installation Guidelines	60
7.2.2 Integration Guidelines	64
Appendix 8:.....	67
PMM06 Integration with Visual Studio	67
8.1 Description.....	67
8.2 INTEGRATION GUIDELINES	67
8.2.1 Visual Studio Installation Guidelines	67

8.2.2	Integration Guidelines	69
	Appendix 9:.....	72
	PMM06 Integration with MICROCHIP Studio	72
9.1	Description.....	72
9.2	INTEGRATION GUIDELINES	72
9.2.1	MICROCHIP Studio Installation Guidelines	72
9.2.2	Integration Guidelines	81
	MECHANICAL CHARACTERISTICS	84

KEY FEATURES

- Simple to set up and cost effective
- (12-60) x Isolated Digital Input 24VDC
- Isolation is provided on all 12x input channels with opto-couplers
- 32-bit ARM core CPU with 48 MHz clock
- High speed data processing and rapid communications turnaround times
- Hot swap redundancy exchange and fail safe ready
- Stand-alone programming or Modular operation modes
- Optional RTC module upon order.
- Issuing alarm for the CPU in case of power losses under 22VDC
- 1x Ethernet 10/100 port
- 1x Serial RS485 port with power surge $\pm 2.5\text{kV}$ isolation protection
- Overvoltage and reverse polarity protected
- Onboard Termination resistor via dip switch
- Wide range of power supply options (10-60V DC or 10-40 V AC)
- 5000V system/field isolation
- EMI, EMS, EMC and shock protected
- -40 to 80 °C Operating temperature
- 5% to 90% Non-condensing Relative Humidity
- Standard 35mm din rail mounting bracket

DESCRIPTION

PMM0620-024 is a reliable, and simple to set up digital Input module that has 12 isolated channels basically and can be extended to 60 channels using PMM0620 extensions.

This opto-coupled architecture makes each input channel rather rugged, capable of isolating the CPU from transient voltage “spikes” and other electrical phenomena capable of causing damage.

In addition, the module can keep the saved data for almost a month in case of power disconnection as it remains in a sleep mode.

The module receives digital signals from sensors and field devices of 24V range. The LED indicators indicate the instantaneous status of the field devices whether ON or OFF. PMM0620-024 is widely used in signal interface switching of PLC, single chip or other industrial control board.

Moreover, PMM0620-024 operates under three operational modes:

- **Modular operation mode:** the module is connected to a PLC by RS485 and implement specific function assigned by the PLC.
- **Fail Safe mode:** the module should be pre-programmed in case of lost connection with the PLC to carry on its function effectively.
- **Stand-alone:** the module can be programmed to work as PLC and control the field devices.

In addition, PMM0620-024 industrial grade IO module can function under harsh environmental conditions of any site. It is reliable, easy to configure and has an extreme accuracy.

TECHNICAL SPECIFICATIONS

Input Specifications

Signal type	Voltage
Voltage signal type	24V DC
Input characteristic	High-Side switching
Input current per channel	10mA
Isolation	5000V system/field
Degree of protection	IP45

Interfaces

Serial	1x RS485 port
Ethernet	1x 10/100
DI/DO	(12 – 60) digital inputs 24VDC
USB	1x USB2.0 Type micro-B
LED Indicators	2x LED for communication over RS485 port indication 12x LED for input signal indication

Power Parameters

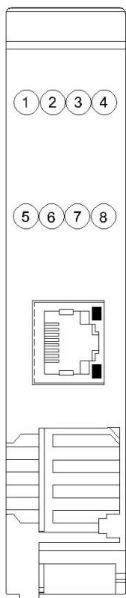
Input Power Supply Options	10-60 VDC (10-48 VAC)
Power Connector	Phoenix Contact 4 pins 3.5mm

Physical Characteristics

Housing	Polyamide (Nylon 6.66)/PA
Dimensions	3.91x4.37x0.99 inch (99.4x111.2x22.6 mm)
Mounting Options	DIN Rail

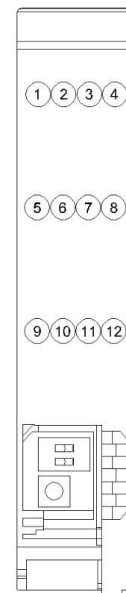
PINS ASSIGNMENTS

TOP VIEW



1. D+ (RS485)
2. D- (RS485)
3. Common
4. Common
5. V DC+ (10-60)
6. V DC- (10-60)
7. Earth
8. Earth

BOTTOM VIEW

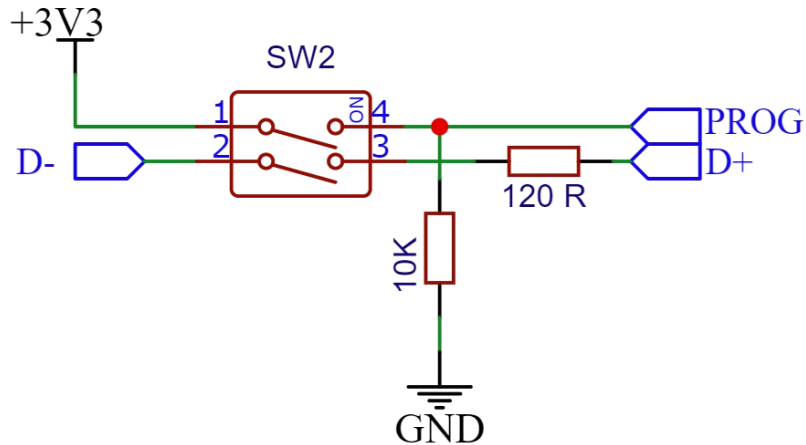


1. Digital Input 01
2. Digital Input 02
3. Digital Input 03
4. Digital Input 04
5. Digital Input 05
6. Digital Input 06
7. Digital Input 07
8. Digital Input 08
9. Digital Input 09
10. Digital Input 10
11. Digital Input 11
12. Digital Input 12

HARDWARE CONFIGURATION

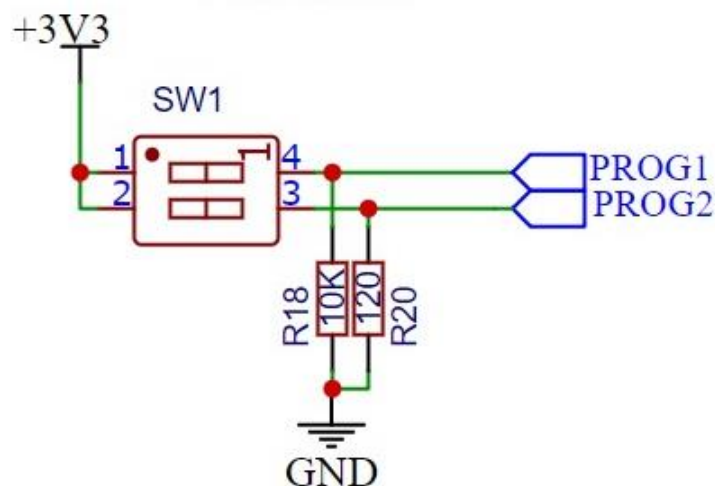
Termination resistor is needed to avoid data-corrupting reflections and expand the network between D+ and D- lines for the RS485.

- Turn on the dip switch (closed circuit) to have a termination resistor of 120Ω between D+ and D- lines.
- The other dip switch is for PROG mode, it is used to switch between two customized modes. As shown in the figure below.



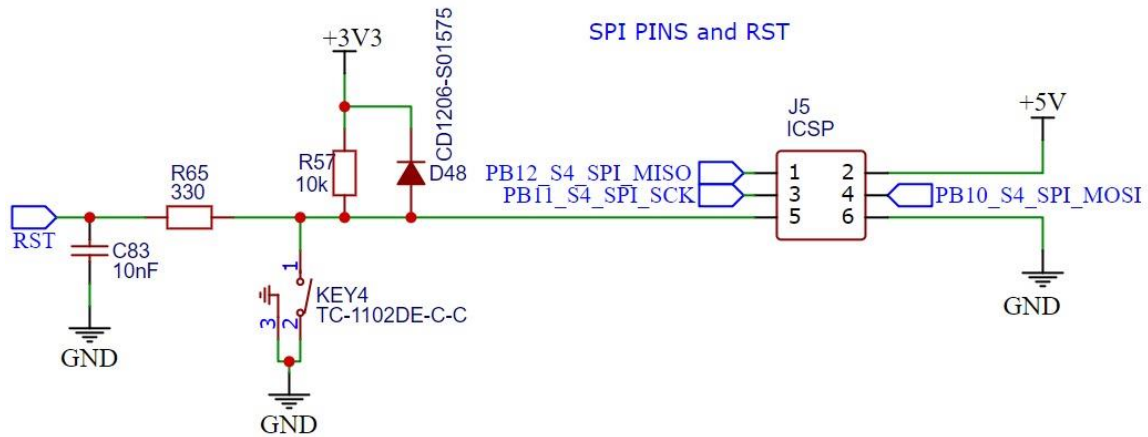
PMM0620-024 can be alternatively switched between two preprogrammed modes through the dip switch shown below.

- Turn on the dip switch (closed circuit) on mode number one to enable the device to run in the first functional programmed mode.
- Turn on the dip switch (closed circuit) on mode number two to enable the device to run in the second functional programmed mode.
- In case of not choosing any modes by the user the device will run in the general/default mode.



In addition, there is a reset button enabling the user to reset the device to the default settings.

- In order to reset the device; push the reset button.
- The device will reset automatically to the default settings.



HARDWARE CONNECTIONS

1. Connecting Power

PMM0620-024 has two power supply options 10-60V DC or 10-40 V AC, the user has to connect the positive power line (+) to pin no.5 in the top view and the negative line (-) to pin no.6 as illustrated in the pin's assignments.

Note: the power is protected against overvoltage and reverse polarity in case of wrong connection.

2. Connecting Serial Device

The unit's serial port is located on the top panel. If you are connecting an RS485 multidrop network with multiple devices, note the following:

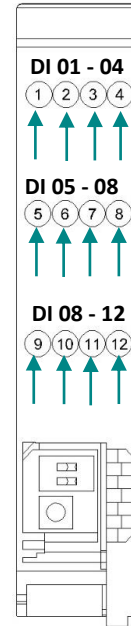
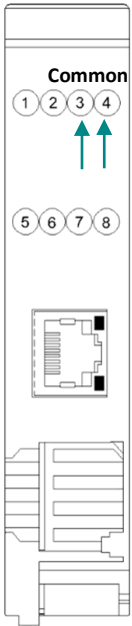
1. All devices that are connected to a single serial port must use the same protocol (i.e., either Modbus RTU or Modbus ASCII).
2. Connect the D+ with pin no.1 and D- with pin no.2 and Earth with pin no.4 as illustrated in the pin's assignments to complete the connection successfully.
3. Turn on the dip switch to have 120 Ω termination resistor between the D+ and D- lines. Refer hardware configuration section.

3. Connecting to a Host or the Network

There is a 10/100 Ethernet port at the module's top panel. This port is used to connect the module with a host or Ethernet network.

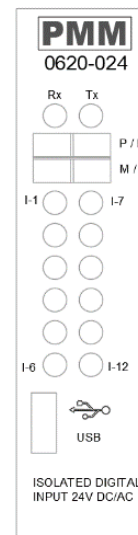
4. Connecting Digital Input

Connect the signal line with one of the twelve digital input pins on the bottom view (01-12) and the common line with pin no.3 or 4.



There are 14x LED indicators at the front panel. 2x LED are for communication indication through RS485 port and 12x LED for indicating the inputs status.

LED No.	Indication
Rx, Tx	Indicating the communication through RS485 port OFF: No Data is being transmitted or received through the port Steady-Green: Data is being transmitted or received through the port
Ix-I12	Indicating the status of Input x OFF: Input x is off Steady-Green: Input x is on



BOARD EXTENSIONS

PMM0620-024 has a fixed amount of digital input channels capability built into the unit (12 DI), while the extension boards are used to provide customized digital input capability up to 60 isolated channels.

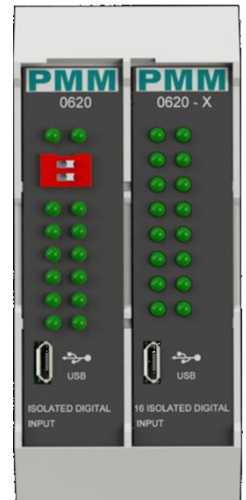
1. 0620-024 X (16 Digital Input)

Description

PMM0620-024X is a reliable, and simple to set up extension for PMM0620-024 digital Input module that adds 16 additional digital isolated input channels basically. This opto-coupled architecture makes each input channel rather rugged, capable of isolating the CPU from transient voltage “spikes” and other electrical phenomena capable of causing damage.

In addition, the module can keep the saved data for almost a month in case of power disconnection as it remains in a sleep mode.

PMM0620-024X is widely used in signal interface switching of PLC, single chip or other industrial control board.



Key Features

- Simple to set up and cost effective
- 16x additional Isolated Digital Input 24VDC
- Isolation is provided on all 16x input channels with opto-couplers
- Optional RTC module upon order.
- 1x Serial RS485 port with power surge $\pm 2.5\text{kV}$ isolation protection
- Overvoltage and reverse polarity protected
- Onboard Termination resistor via dip switch
- Wide range of power supply options (10-60V DC or 10-40 V AC)
- 5000V system/field isolation
- EMI, EMS, EMC and shock protected
- -40 to 80 °C Operating temperature
- 5% to 90% Non-condensing Relative Humidity



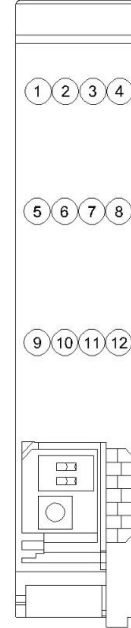
PINS ASSIGNMENTS

TOP VIEW



- 1. Digital Input 13
- 2. Digital Input 14
- 3. Digital Input 15
- 4. Digital Input 16
- 5. COM 1
- 6. COM 2
- 7. COM 3
- 8. COM4
- 9. V DC+ (10-60)
- 10. V DC- (10-60)
- 11. D+ (RS485)
- 12. D- (RS485)

BOTTOM VIEW

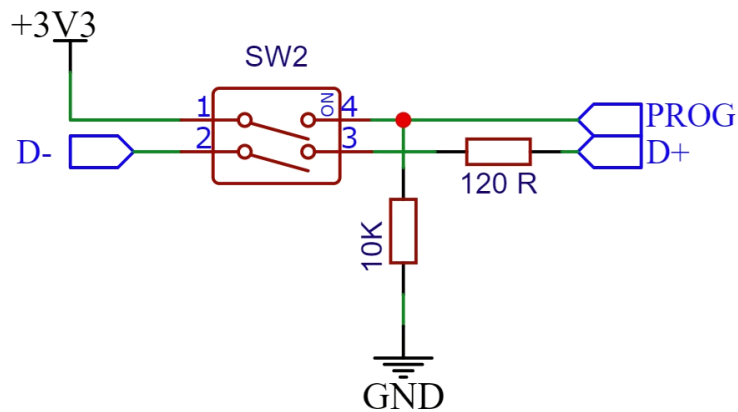


- 1. Digital Input 01
- 2. Digital Input 02
- 3. Digital Input 03
- 4. Digital Input 04
- 5. Digital Input 05
- 6. Digital Input 06
- 7. Digital Input 07
- 8. Digital Input 08
- 9. Digital Input 09
- 10. Digital Input 10
- 11. Digital Input 11
- 12. Digital Input 12

HARDWARE CONFIGURATION

Termination resistor is needed to avoid data-corrupting reflections and expand the network between D+ and D- lines for the RS485.

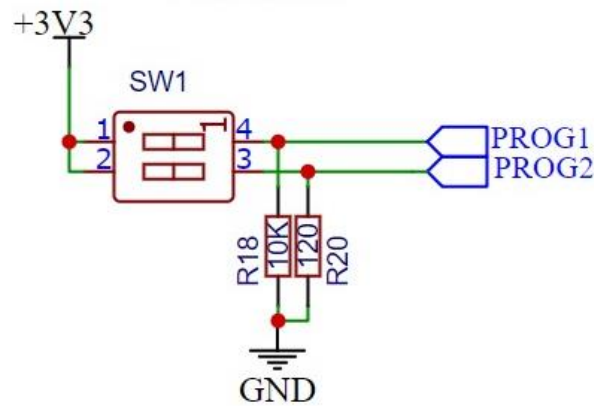
- Turn on the dip switch (closed circuit) to have a termination resistor of 120Ω between D+ and D- lines.
- The other dip switch is for PROG mode, it is used to switch between two customized modes. As shown in the figure below.



PMM0620-024X can be alternatively switched between two preprogrammed modes through the dip switch shown below.

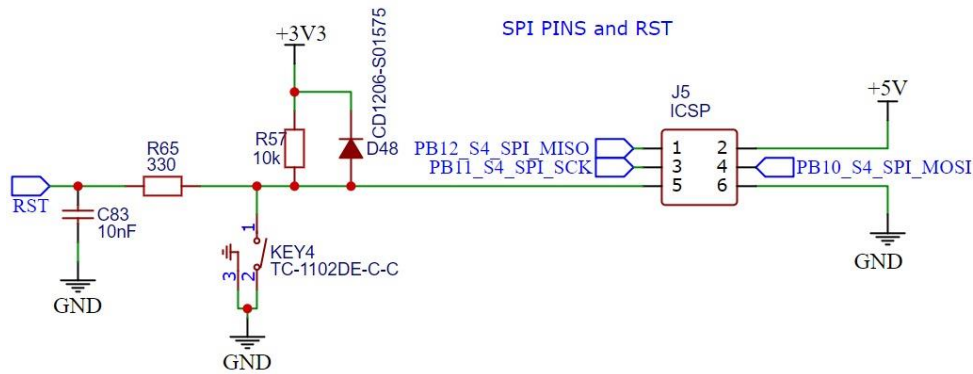
- Turn on the dip switch (closed circuit) on mode number one to enable the device to run in the first functional programmed mode.

- Turn on the dip switch (closed circuit) on mode number two to enable the device to run in the second functional programmed mode.
- In case of not choosing any modes by the user the device will run in the general/default mode.



In addition, there is a reset button enabling the user to reset the device to the default settings.

- In order to reset the device; push the reset button.
- The device will reset automatically to the default settings.



HARDWARE CONNECTIONS

1. Connecting Power

PMM0620-024X has two power supply options 10-60V DC or 10-40 V AC, the user has to connect the positive power line (+) to pin no.9 in the top view and the negative line (-) to pin no.10 as illustrated in the pin's assignments.

Note: the power is protected against overvoltage and reverse polarity in case of wrong connection.

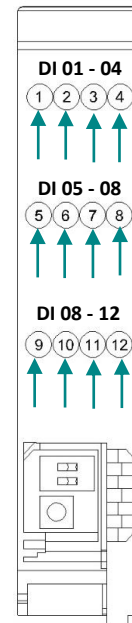
2. Connecting Serial Device

The unit's serial port is located on the top panel. If you are connecting an RS485 multidrop network with multiple devices, note the following:

1. All devices that are connected to a single serial port must use the same protocol (i.e., either Modbus RTU or Modbus ASCII).
2. Connect the D+ with pin no.11 and D- with pin no.12 as illustrated in the pin's assignments to complete the connection successfully.
3. Turn on the dip switch to have 120 Ω termination resistor between the D+ and D- lines. Refer hardware configuration section.

5. Connecting Digital Input

Connect the signal line with one of the twelve digital input pins on the bottom view (01-16) and the common line with pin no.5, no.6, no.7 or 8.



There are 16x LED indicators at the front panel for indicating the inputs status.

LED No.	Indication
Ix-I16	Indicating the status of Input x OFF: Input x is off Steady-Green: Input x is on

2. 0620-024 XP (24 Digital Input)

Description

PMM0620-024XP is a reliable, and simple to set up extension for PMM0620-024 digital Input module that adds 24 additional digital isolated input channels basically. This opto-coupled architecture makes each input channel rather rugged, capable of isolating the CPU from transient voltage “spikes” and other electrical phenomena capable of causing damage.

In addition, the module can keep the saved data for almost a month in case of power disconnection as it remains in a sleep mode.

PMM0620-024XP is widely used in signal interface switching of PLC, single chip or other industrial control board.



Key Features

- Simple to set up and cost effective
- 24x additional Isolated Digital Input 24VDC
- Isolation is provided on all 24x input channels with opto-couplers
- Overvoltage and reverse polarity protected
- Onboard Termination resistor via dip switch
- Powered internally, hence there is no need for external power supply
- 5000V system/field isolation
- EMI, EMS, EMC and shock protected
- -40 to 80 °C Operating temperature
- 5% to 90% Non-condensing Relative Humidity



PINS ASSIGNMENTS

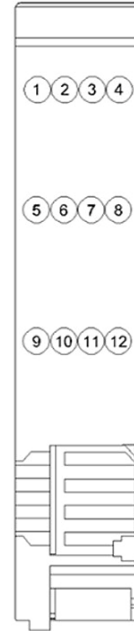
TOP VIEW



- 1. Digital Input 13
- 2. Digital Input 14
- 3. Digital Input 15
- 4. Digital Input 16
- 5. Digital Input 17
- 6. Digital Input 18
- 7. Digital Input 19
- 8. Digital Input 20
- 9. Digital Input 21
- 10. Digital Input 22
- 11. Digital Input 23
- 12. Digital Input 24

- 1. COM1
- 2. COM2
- 3. COM3
- 4. COM4

BOTTOM VIEW

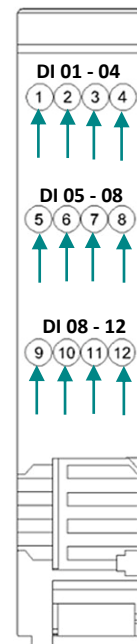
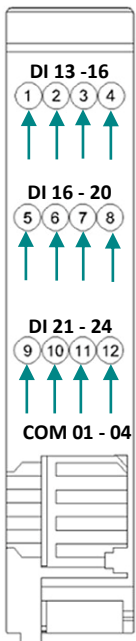


- 1. Digital Input 01
- 2. Digital Input 02
- 3. Digital Input 03
- 4. Digital Input 04
- 5. Digital Input 05
- 6. Digital Input 06
- 7. Digital Input 07
- 8. Digital Input 08
- 9. Digital Input 09
- 10. Digital Input 10
- 11. Digital Input 11
- 12. Digital Input 12

HARDWARE CONNECTIONS

Connecting Digital Input

Connect the signal line with one of the twelve digital input pins on the bottom view (01-16) and the common line with pin no.5, no.6, no.7 or 8.



There are 24x LED indicators at the front panel for indicating the inputs status.

LED No.	Indication
Ix-I24	Indicating the status of Input x OFF: Input x is off Steady-Green: Input x is on

ORDERING INFORMATION

PMM0620-024	-x	-09xx	Extension -20X, X
Power supply			
10-56 VDC	-1		
COM Port 1			
RS485		-0912	
Extensions			
No Extension			
16x Additional isolated digital input			-20X,16
24x Additional isolated digital input			-20XP,24
32x Additional isolated digital input			-20X,32
48x Additional isolated digital input			-20XP,48
40x Additional isolated digital input			-20X,20XP,40

CONTACT INFORMATION:

For direct inquiries or any customized orders,
contact us on sales@Pmm-usa.us

Appendices

Appendix 1:

PMM USB Configuration

1.1. Description

The user can use USB configuration utility which is only compatible with Microsoft Windows 10 operating system for the configuring the device's parameters for the first time such as Ethernet, Modbus and serial settings for PMM devices instead of using the web server configuration for more control and advanced features. In addition to upgrading the firmware and factory reset.

1.2. Key Features

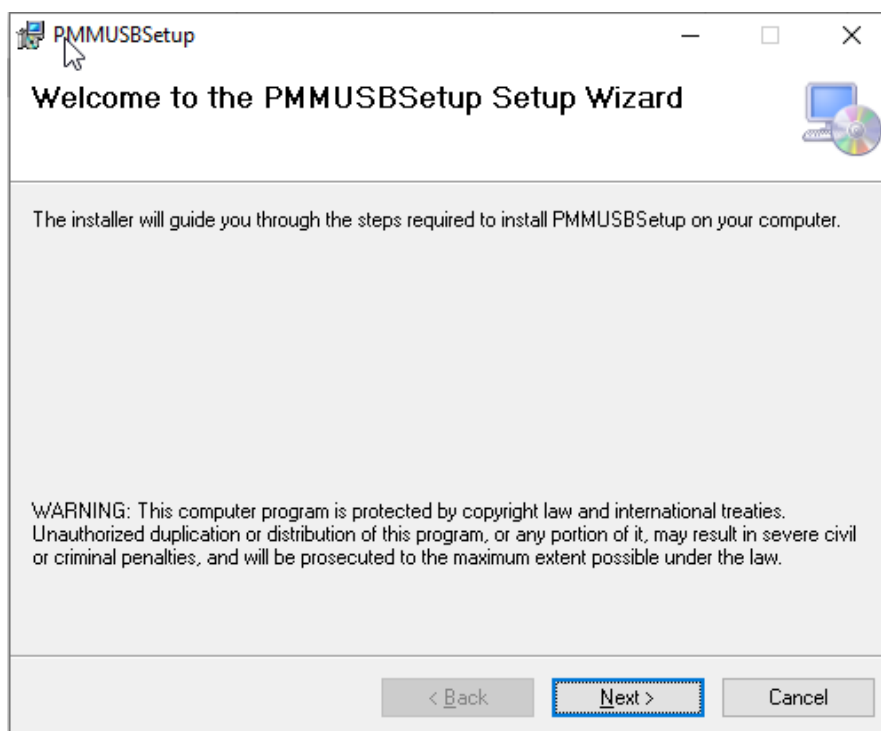
- Configure PMM Products with USB Port
- Restore and Backup user Setting
- Upload/download online Setting for process tuning
- Firmware downloading/updating
- Factory reset and more ...

1.3. Set Up Instructions

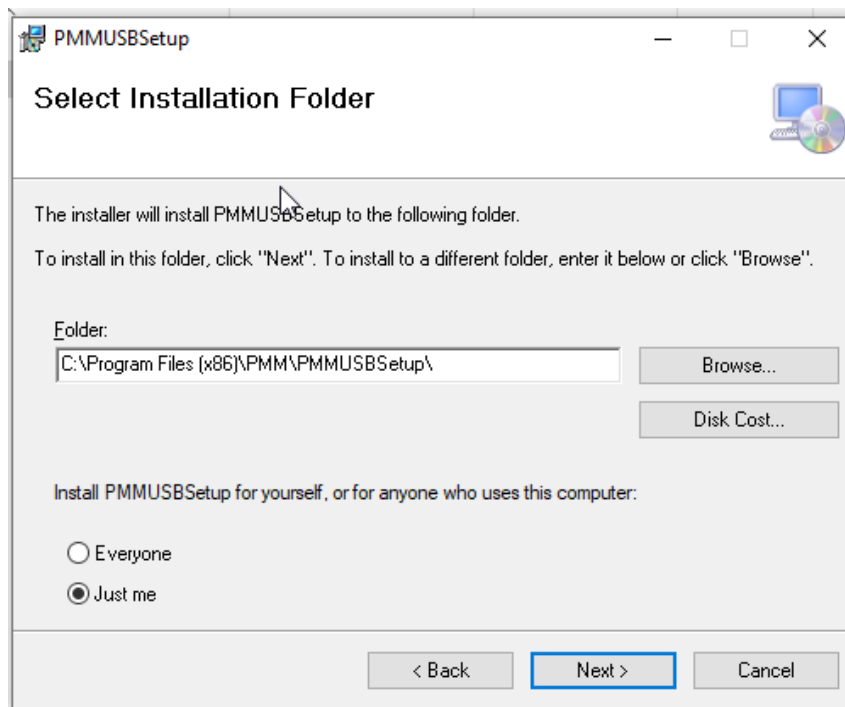
- Before installing the USB configurator, the user must install .NET 5.0 SDK (v5.0.408) run time. [Click here](#) to start downloading.
- Click on "[PMM USB Configurator](#)" then click on **download**.

Follow the downloading instructions to complete installing the application successfully.

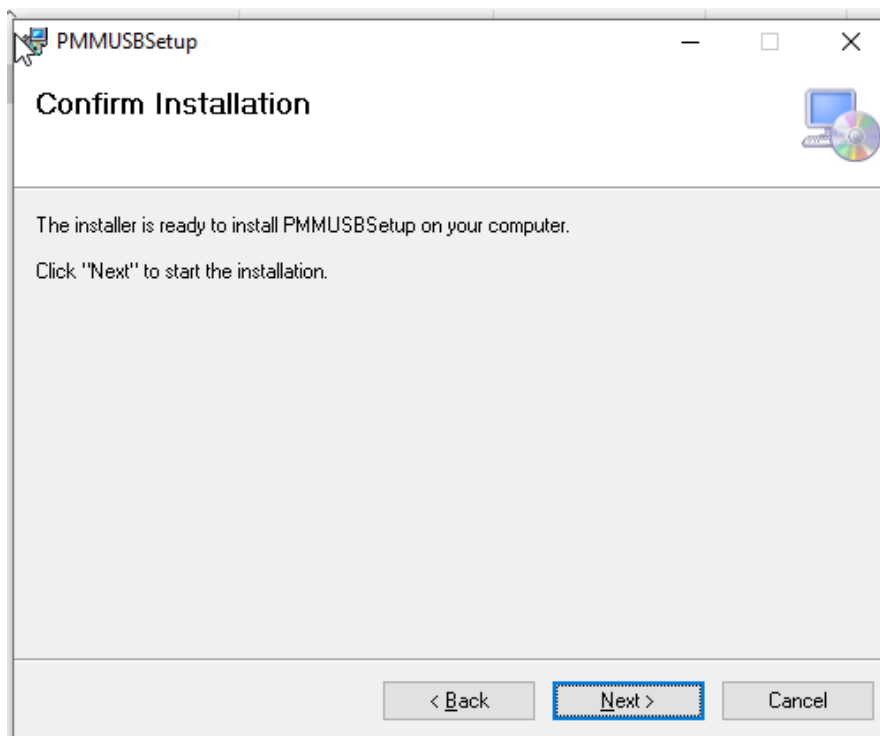
- Once the user clicks on "Setup" the following window appears.
- Click on "Next"



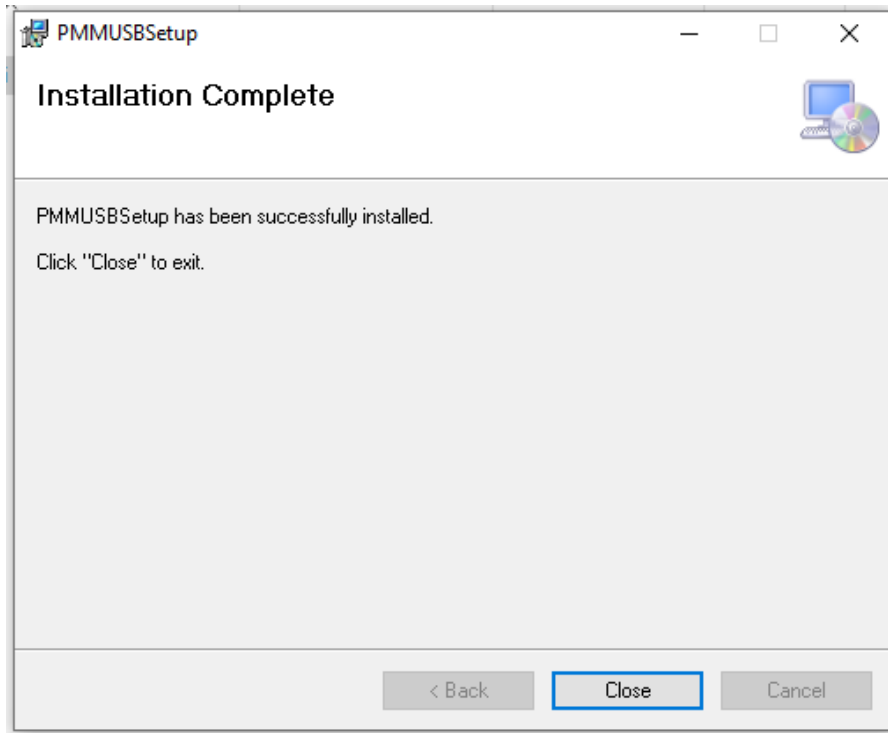
- Browse for the desired installation folder that is saved in your PC.
- Choose whether the setup is for yourself or everyone uses the PC by clicking on the desired choice.
- Click on “Next”.



- Confirm the installation by clicking on “Next”



- When the installation is completed, the following window appears for confirmation.
- Click on “Close”.



1.4. USB CONFIGURATION UTILITY

The following Instructions aim to let the user able to complete device configuration successfully:

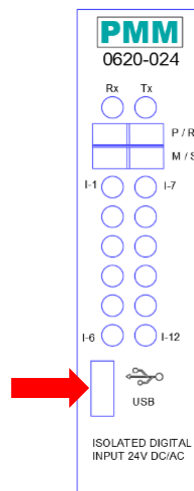
1. Table of requirements:

No.	Requirements
1.	USB type: Micro-USB
2.	Personal computer with Windows 10 operating system.

2. USB connecting instructions:

- Connect the Micro-USB type to the device through the USB port which is located in the front panel of PMM0620.

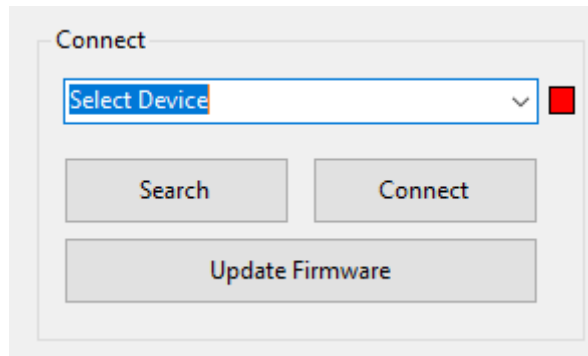
See the figure below for more clarification:



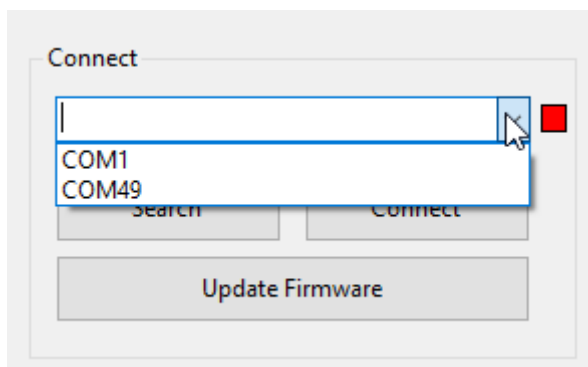
- Connect the other side with personal computer (PC).

Once the USB is connected between the device and the PC, the user can start the configuration as shown in the figures and instructions below:

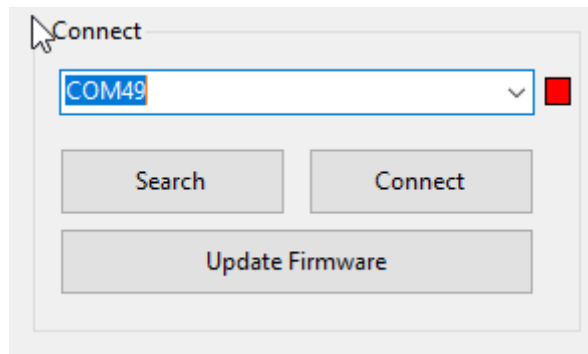
- Click on “Select device” drop list as shown below.



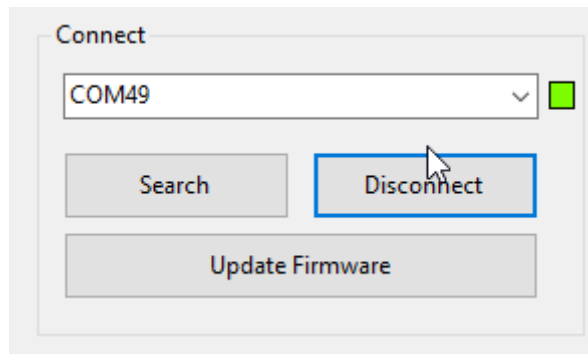
- All the connected devices will be displayed based on which of COMs are connected.



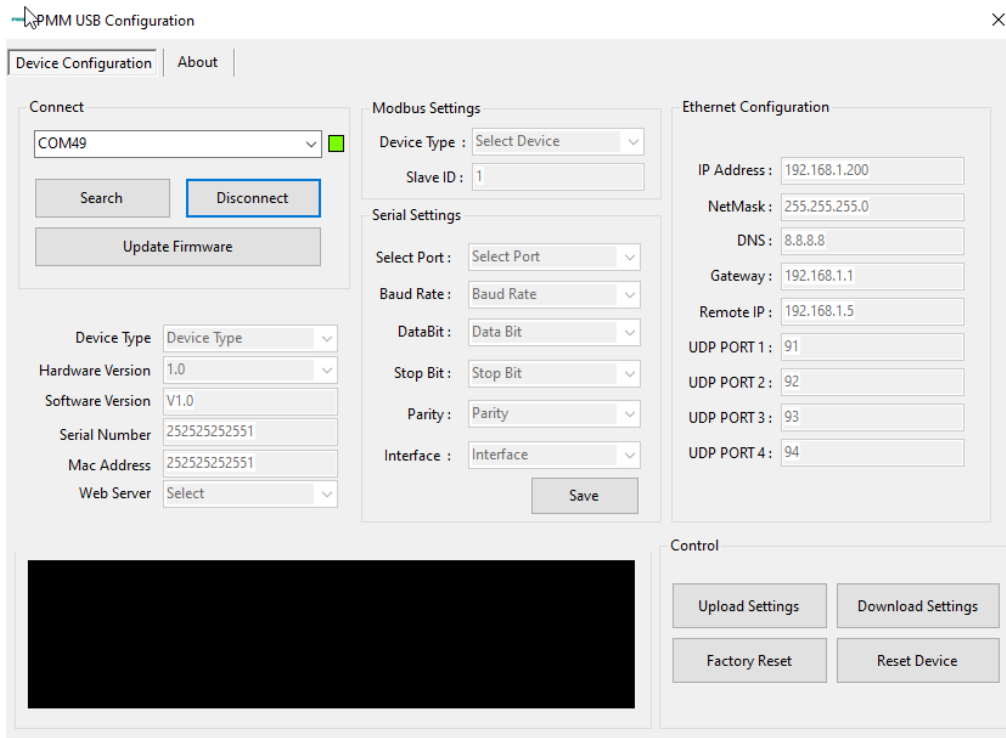
- Click on the connected PMM0620 device that needs settings configuration.



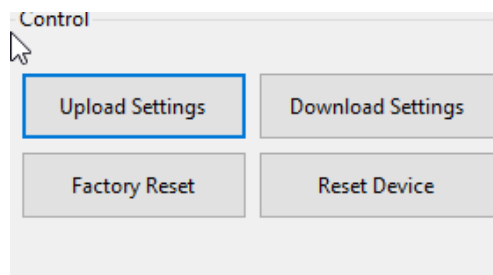
- Click on “Connect”, the light indicator will be turned into **green**.



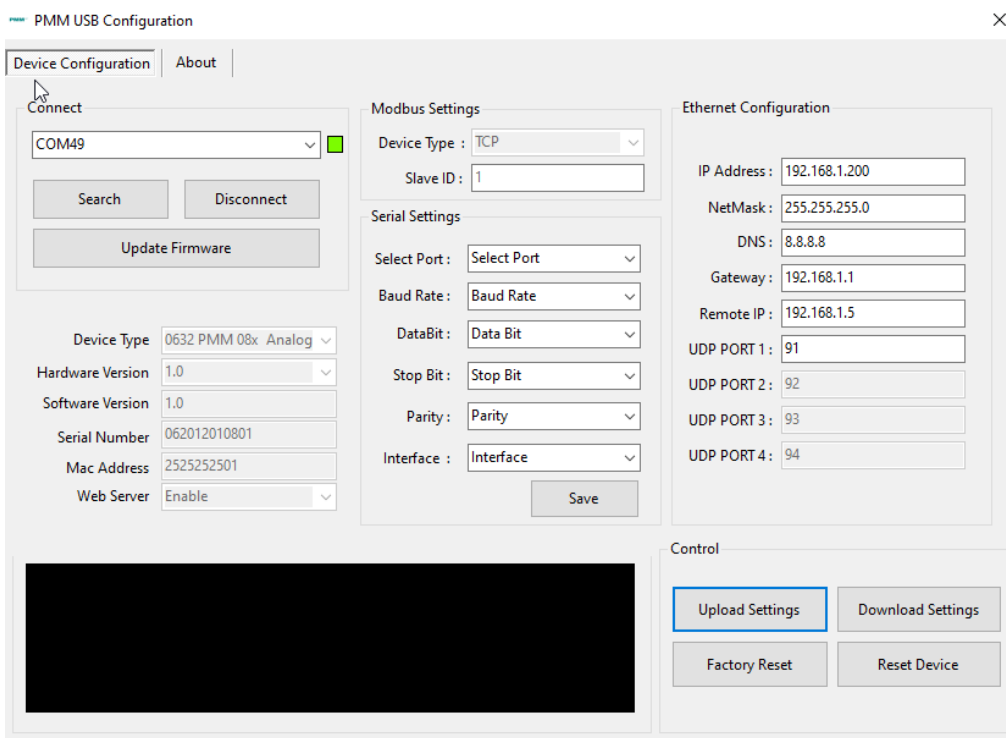
- Once the device is connected, the settings by default are not editable as shown below.



- Click on “Upload settings” to enable editing the default settings, and upload them on the device.



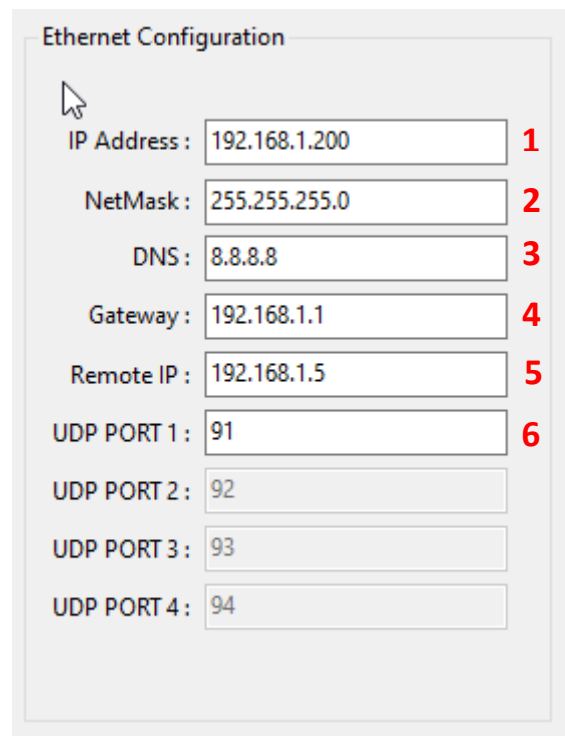
- The Modbus, serial and Ethernet settings are now editable as shown below.



Ethernet Settings

Parameters which are required when configuring Ethernet port are: IP address, Net Mask, Gateway, an DNS Server.

- Type the desired IP address in the specified space (1) which uniquely identifies each computer on the network and the computers use it to communicate with each other.
- Type the desired NetMask in the specified space (2) which identifies an octet in the IP address which is part of the subnet address (network address). Using Net Mask, we know which part of IP address the network address and which part is the host address.
- Type the desired DNS in the specified space (3)
- Set the gateway and remote IP address in the specified spaces (4,5) to be compatible with the network.
- Set the UDP port in the specified space (6).

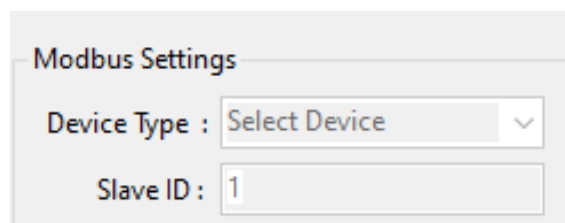


The screenshot shows a configuration window titled "Ethernet Configuration" with several input fields. A mouse cursor is positioned over the IP Address field. Red numbers 1 through 6 are placed to the right of each field to indicate the order of configuration steps.

Field	Value	Step Number
IP Address :	192.168.1.200	1
NetMask :	255.255.255.0	2
DNS :	8.8.8.8	3
Gateway :	192.168.1.1	4
Remote IP :	192.168.1.5	5
UDP PORT 1 :	91	6
UDP PORT 2 :	92	
UDP PORT 3 :	93	
UDP PORT 4 :	94	

Modbus Settings

- Select the device type: Modbus TCP or RTU from the "Device Type" drop list.
- Type the device slave ID; the ID must be unique. The slave ID is set by default to 1.

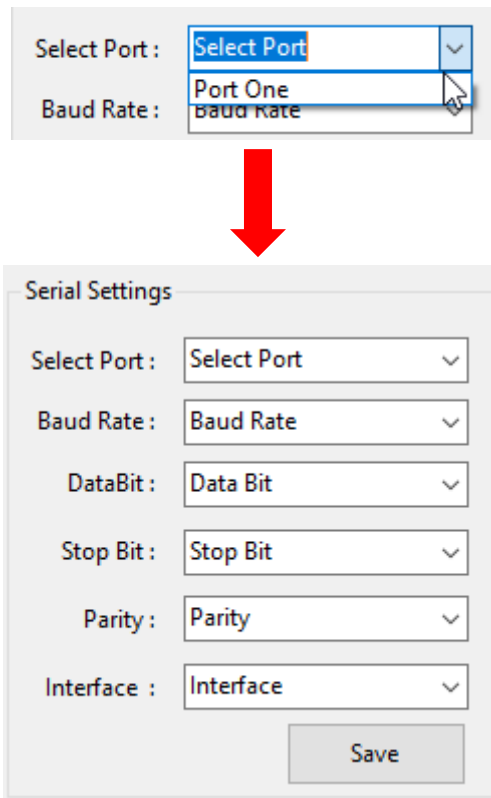


The screenshot shows a configuration window titled "Modbus Settings" with two input fields.

Field	Value
Device Type :	Select Device
Slave ID :	1

Serial Settings

Each PMM device has different number of serial ports; select the port to edit the settings for from “Select Port” drop list.

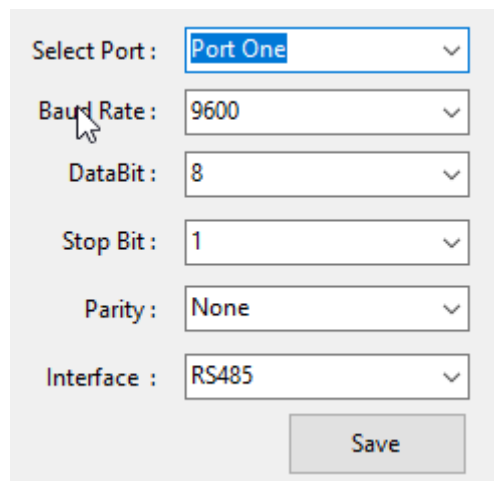


The image shows two screenshots of a software interface. The top screenshot shows a 'Select Port' dropdown menu with 'Select Port' selected, and a 'Baud Rate' dropdown menu with 'Port One' and 'Baud Rate' as options. A red arrow points down to the bottom screenshot, which shows the full 'Serial Settings' window. This window contains several dropdown menus: 'Select Port' (Set to 'Select Port'), 'Baud Rate' (Set to 'Baud Rate'), 'DataBit' (Set to 'Data Bit'), 'Stop Bit' (Set to 'Stop Bit'), 'Parity' (Set to 'Parity'), and 'Interface' (Set to 'Interface'). A 'Save' button is located at the bottom right of the window.

Parameters (that can be configured) used during serial communication include:

- Select from the drop list the **baud rate**; baud rate is the communication speed, which is the number of times per second a serial communication signal changes states; a state being either a voltage level, a frequency, or a frequency phase angle.
- Select from the drop list the **data Bit**: Number of bits to represent one character of data 8 (default)
- Select from the drop list the **stop Bit**: The stop bits are used in asynchronous communication as a means of timing or synchronizing the data characters being transmitted.
- Select from the drop list the **parity**: The parity bit is to determine if the data character being transmitted is correctly received by the remote device.
- Select from the drop list the **Interface**.

Example of Serial settings configuration for port one:



The image shows a screenshot of the 'Serial Settings' window with the following configuration: 'Select Port' is set to 'Port One', 'Baud Rate' is set to '9600', 'DataBit' is set to '8', 'Stop Bit' is set to '1', 'Parity' is set to 'None', and 'Interface' is set to 'RS485'. A 'Save' button is located at the bottom right of the window.

- Click on “Save” to save the new settings of the serial port.
- Click on “Download settings” to download the new settings on the device.
- Click on “Reset” to restore the default settings.

Appendix 2:

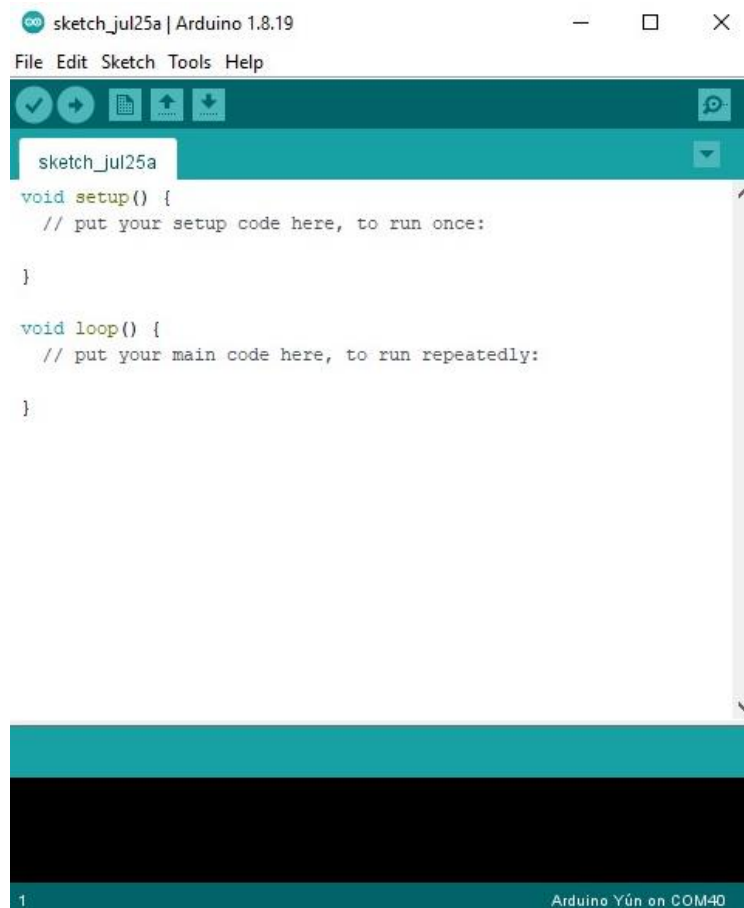
PMM06 Integration with Arduino IDE

2.1 Description

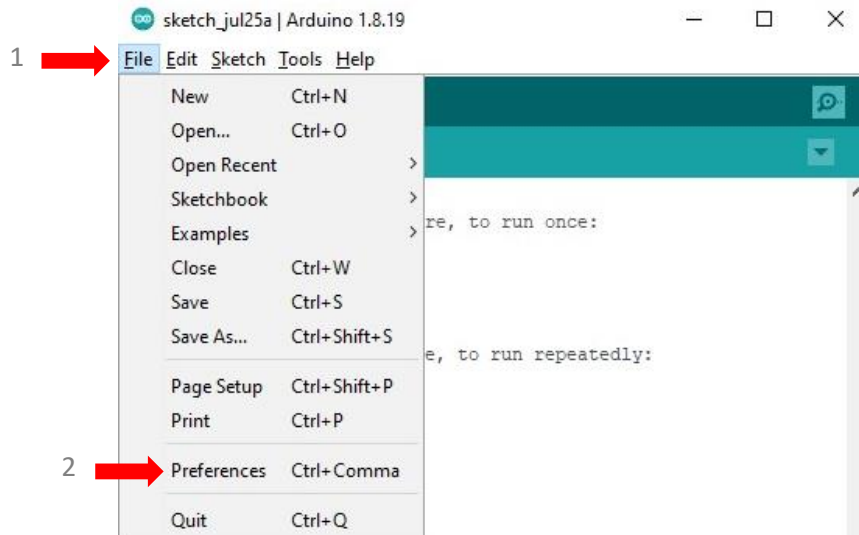
The open-source Arduino Software (IDE) is the most used IDE for Arduino and makes it easy to write code and upload it to PMM06 series. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. In order to work on Arduino (IDE) Make sure you install the libraries first as stated in the instruction manual.

2.2 Integration Guidelines

1. Open open-source Arduino Software (IDE), the following main layout will be displayed as shown below.



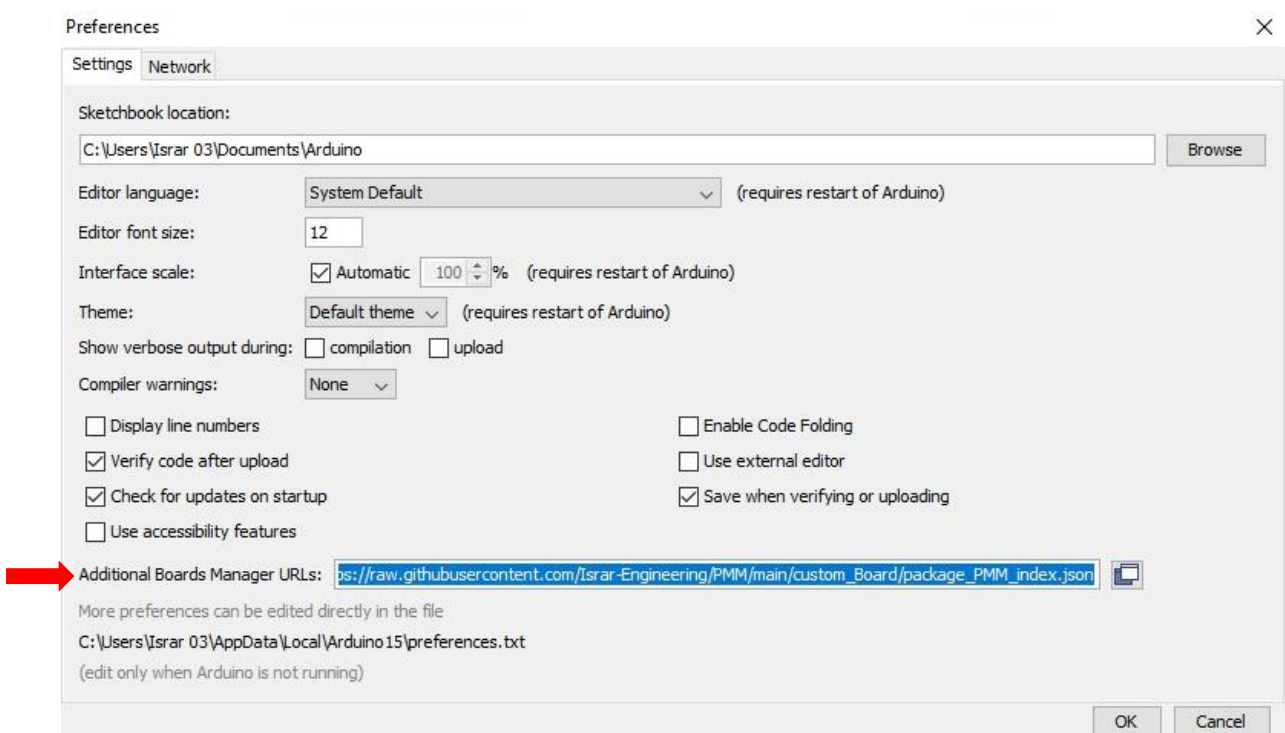
2. Click on “File” in the main tabs and choose “Preferences” or press on the keyboard “Ctrl+Comma”.



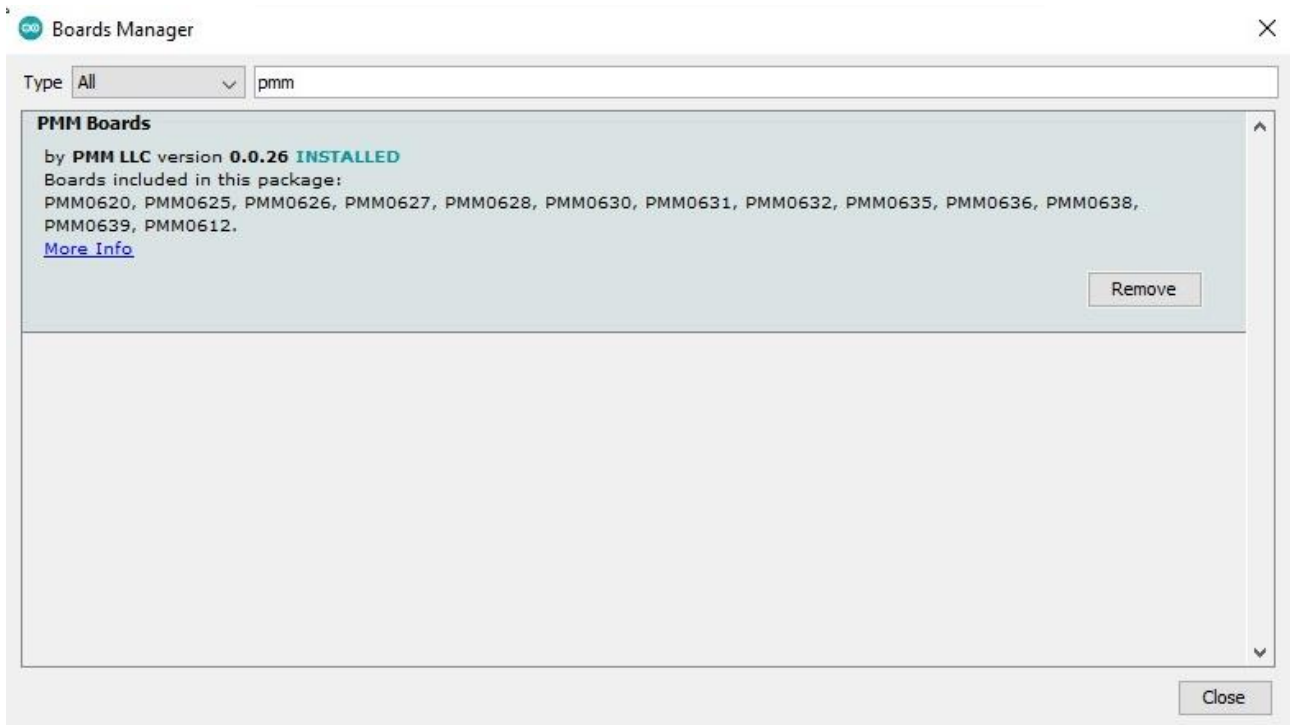
3. The preferences window will be displayed as shown below, the user can edit the settings as needed.
4. At the “Additional Boards Manager URLs” place the following link in order to download the PMM boards.

PMM Boards Definition on Arduino Link: https://raw.githubusercontent.com/Israr-Engineering/PMM/main/custom_Board/package_PMM_index.json

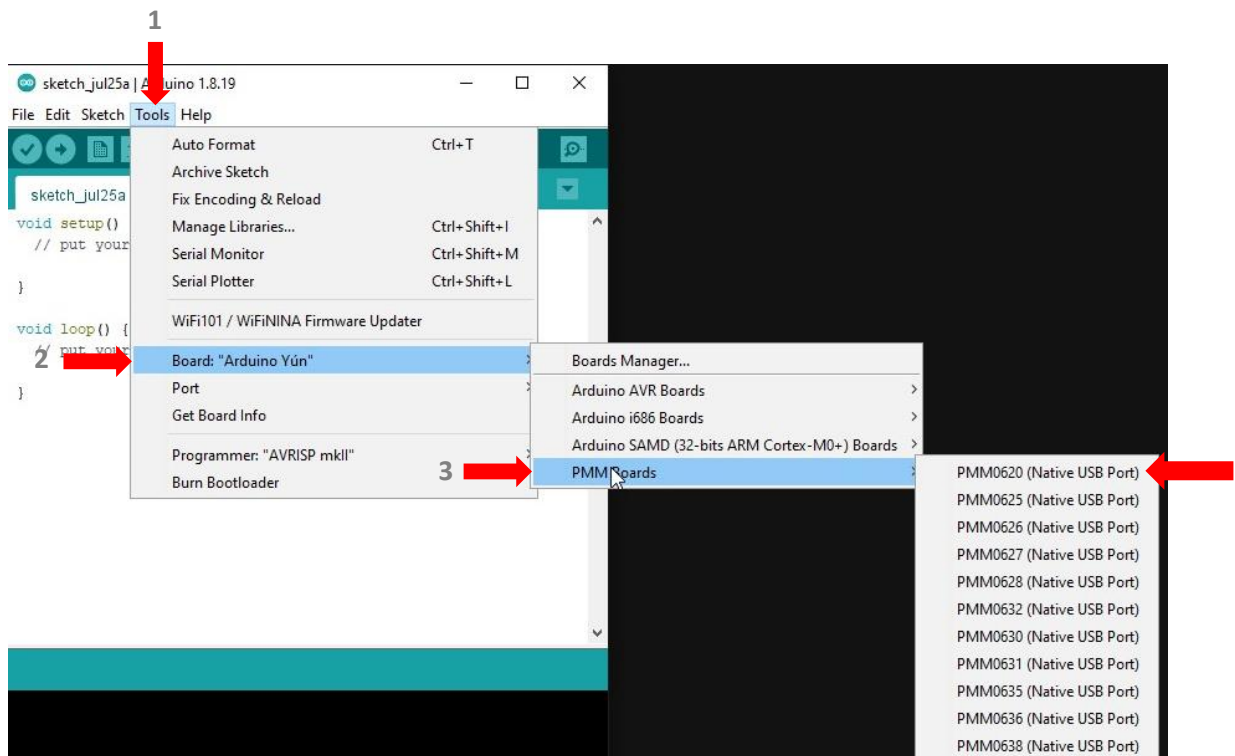
5. Click on “OK”.



- The “Boards Manager” displays all the PMM board that are included in the downloaded package.



- To choose PMM060 board to update its code; click on “Tools” in the main tab.
- Click on “Board”.
- Click on “PMM Boards”.
- All the downloaded PMM board will be displayed in the list, choose PMM0620.



Appendix 3: PMM06 Integration with Visuino

3.1 Description

PMM PLC Systems are built to be Arduino compatible programming environment, where PMM’s PLCs Range is not just compatible with Arduino IDE, but with lots of other Arduino-compatible programming software such as Visuino.

Visuino is a **graphical integrated programming environment** which helps user program Microcontrollers and Microprocessors with the help of easy-to-use visual interface. It is paid software but you can download a free version which is enough at a beginner level

3.2 INTEGRATION GUIDELINES

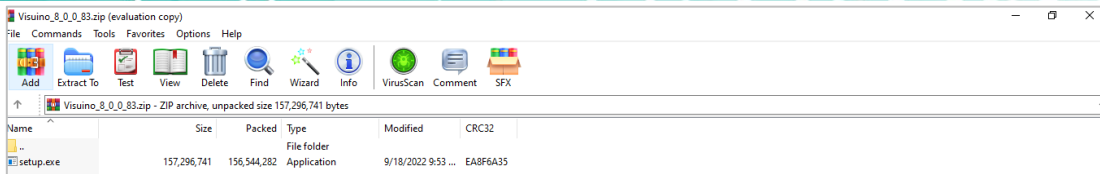
3.2.1 Visuino Installation Guidelines

1. [Click Here](#) to get to the installation page.
2. Once the main page is opened, click on “click here to download”.

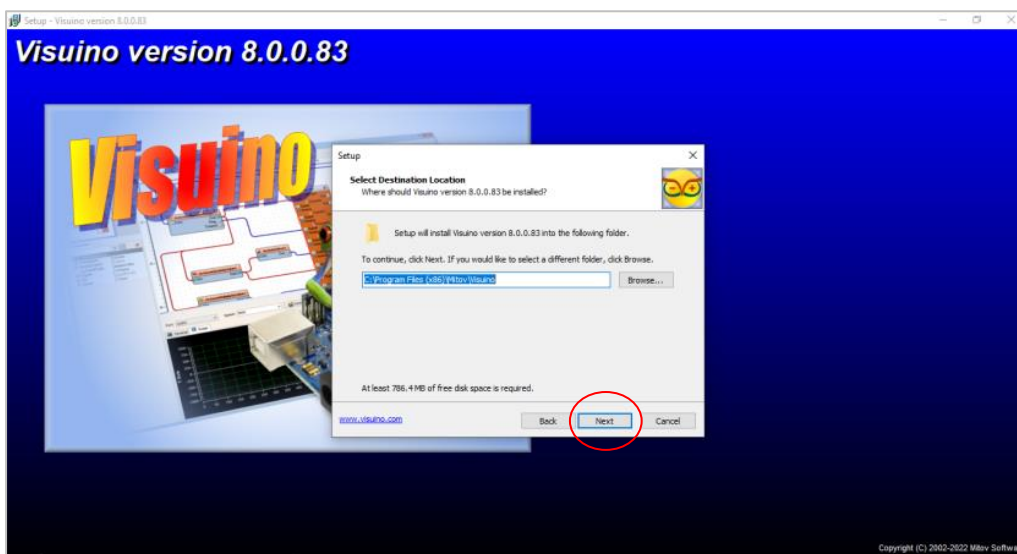
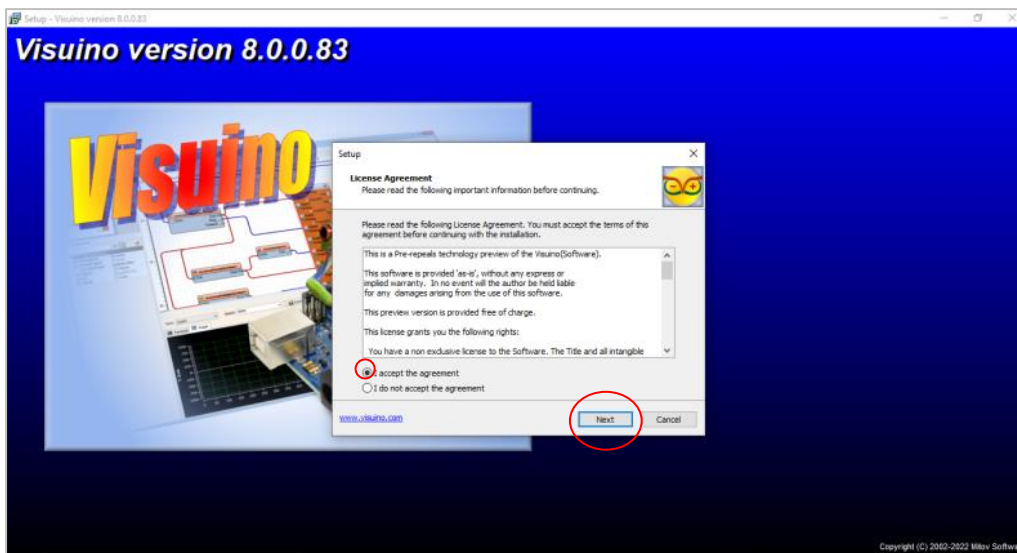


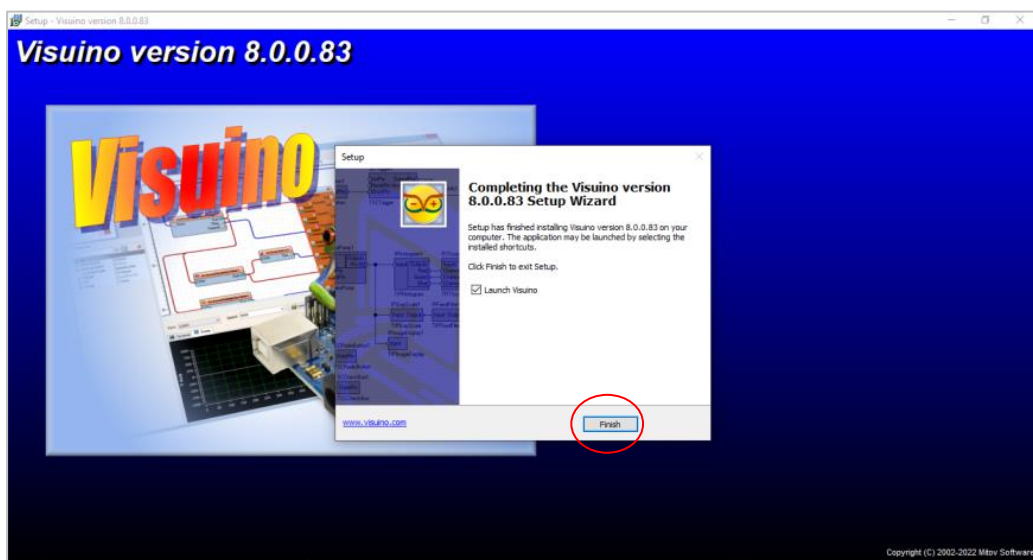
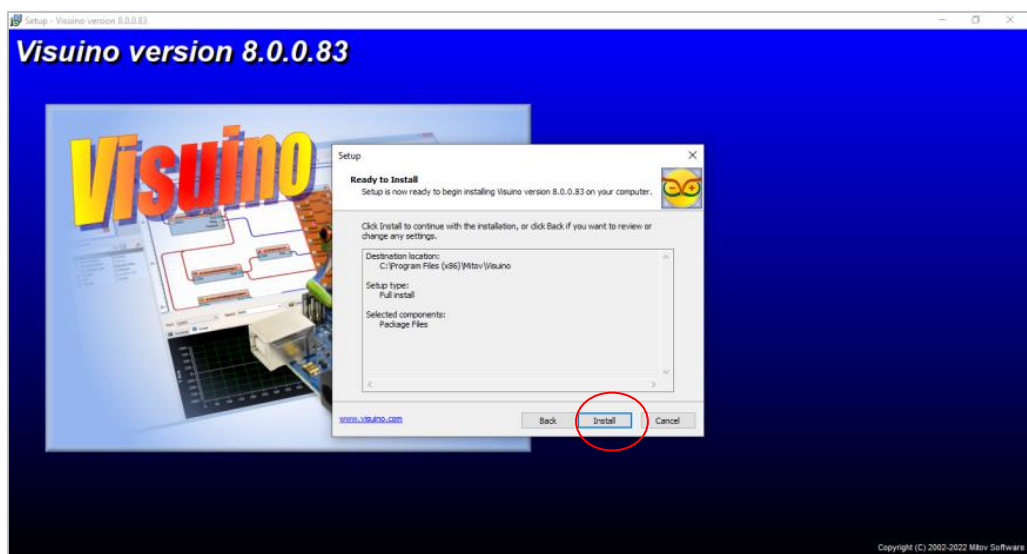
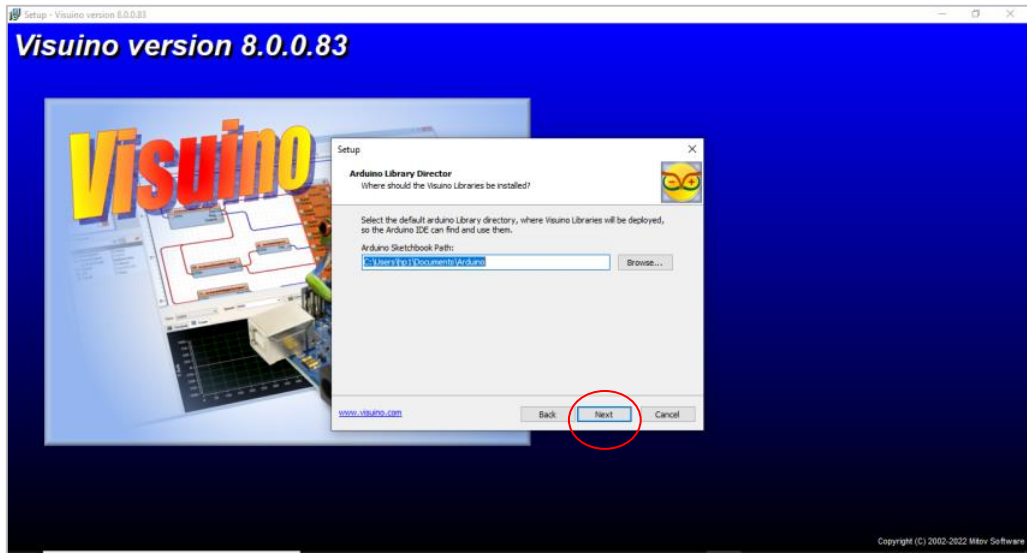
3. Click on “open file” to proceed to installation.



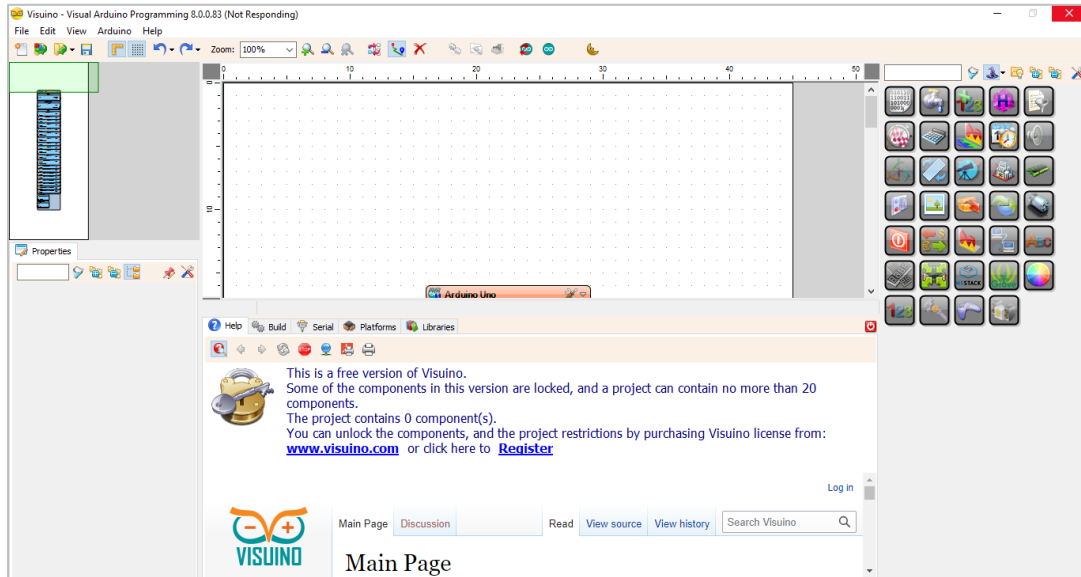


4. Visuino setup process will start, click on “Next” to continue the setup to the completion as shown in the figures below.





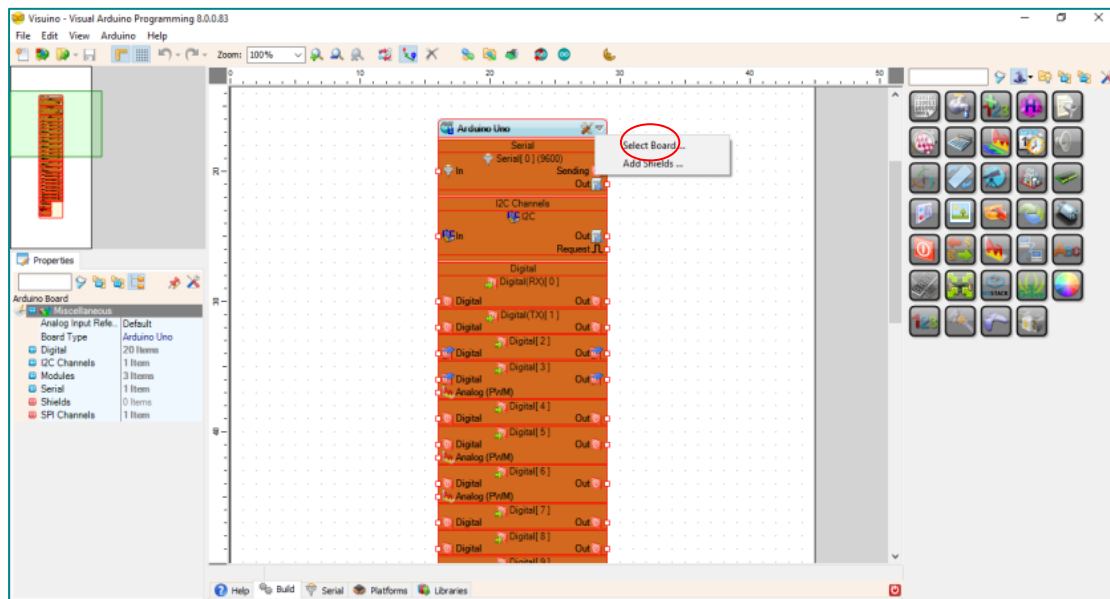
5. Now everything is setup and the user can start with the device.



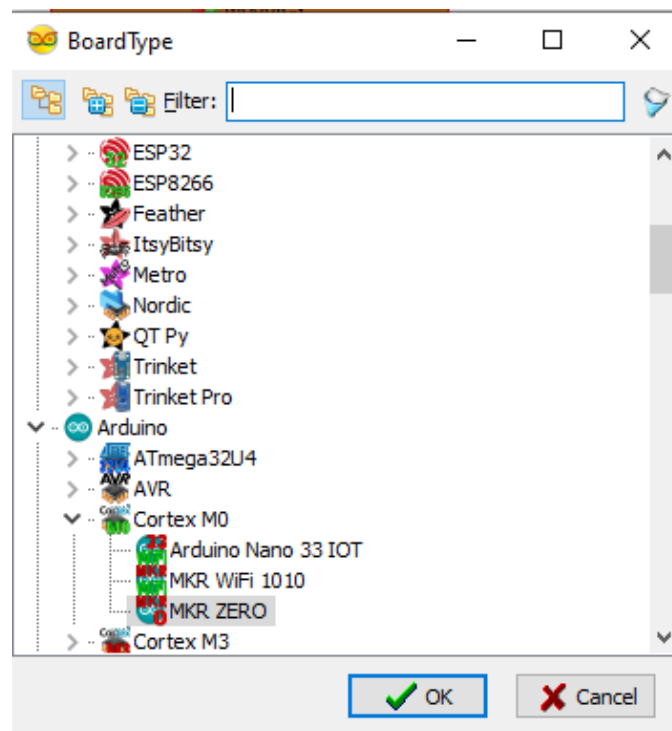
3.2.2 Integration Guidelines

Connect the USB to the device through the USB port in the front panel (Micro-USB type), and connect the other side with personal computer (PC). Once the USB is connected correctly between the device and PC, the user can start the integration as explained in the Integration Guidelines:

1. To select our board, click on "select board".



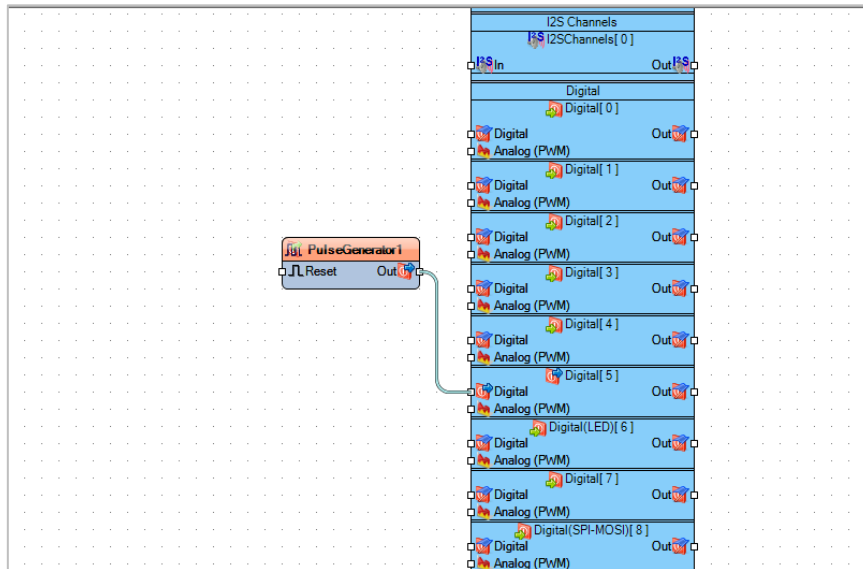
2. Choose the Arduino zero board



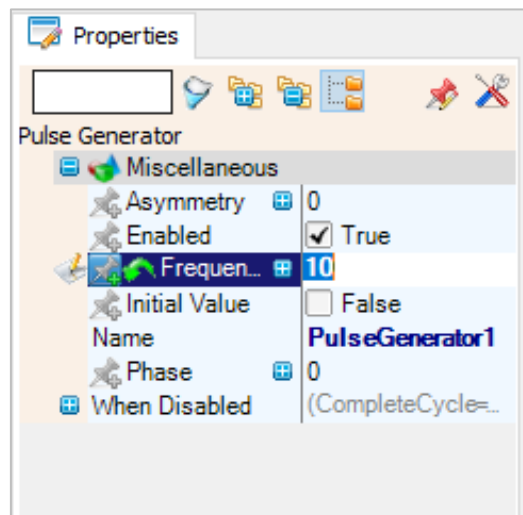
3. Search for you desired function



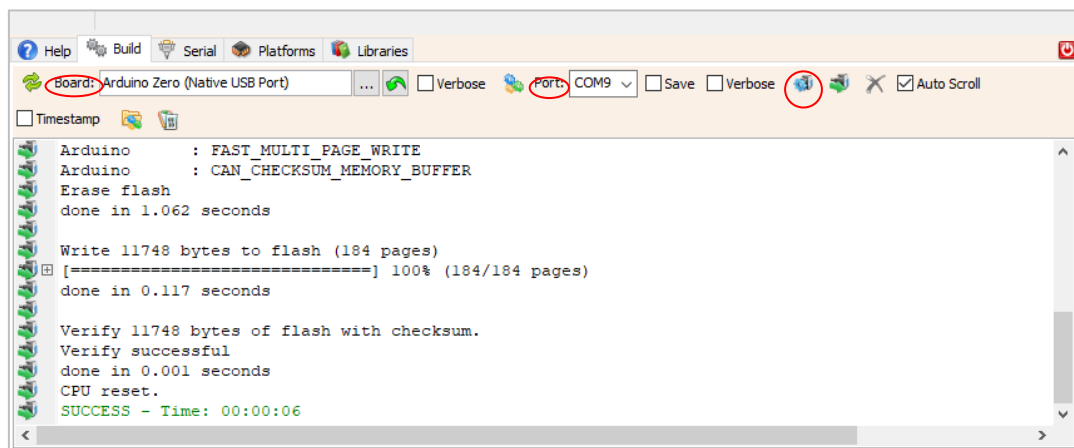
4. Drag and drop the selected function to the left of the desired pin and connect it by clicking on the out pin and dragging the wire to the other side
Note: the example aims to define Pin No.5 as an input and monitor its status between ON/OFF when a power is connected to it, as shown in the figure below



5. Press on the function to adjust values to desiring
Note: here we adjusted the frequency



6. From the bottom of the screen reselect the board and the port then press on the compile button.



7. Now your program should be running.

Appendix 4:

PMM06 Integration with Node-RED

4.1 Description

PMM PLC Systems are built to be Arduino compatible programming environment, where our PLC Range is not just compatible with Arduino IDE, but with lots of other Arduino-compatible programming software such as **Node-RED**.

Node-RED is a flow-based programming tool, ideal for Internet of Things applications. It wires together hardware devices, APIs and online services. It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its run time. - 100% Free software license - Ready for Industrial and home IoT systems based on Node.js - Very active community.

4.2 INTEGRATION GUIDELINES

4.2.1 Node-RED Installation Guidelines

1. [Click Here](#) to get to the installation page.
2. Once the main page is opened, go to “Quick Start” section and click on “[Nodes.js home page](#)” as shown in the figure below.

Quick Start

1. Install Node.js

Download the latest 14.x LTS version of Node.js from the official [Node.js home page](#). It will offer you the best version for your system.

Run the downloaded MSI file. Installing Node.js requires local administrator rights; if you are not a local administrator, you will be prompted for an administrator password on install. Accept the defaults when installing. After installation completes, close any open command prompts and re-open to ensure new environment variables are picked up.

Once installed, open a command prompt and run the following command to ensure Node.js and npm are installed correctly.

7. Choose which version to download. Most users download Version 16.17.0 LTS which is recommended for most users as shown below.

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

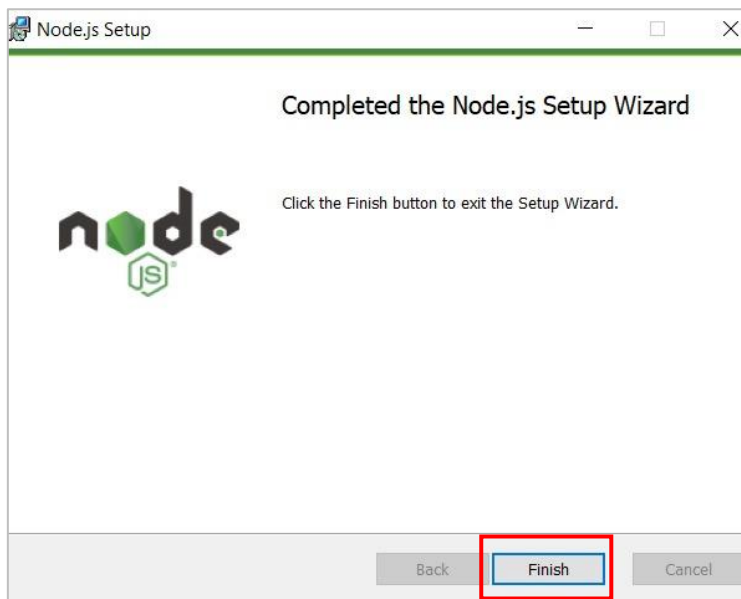
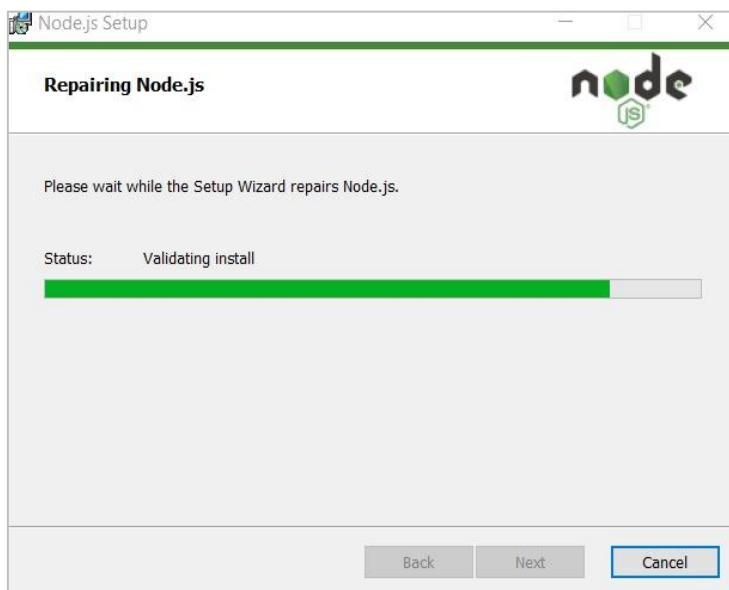
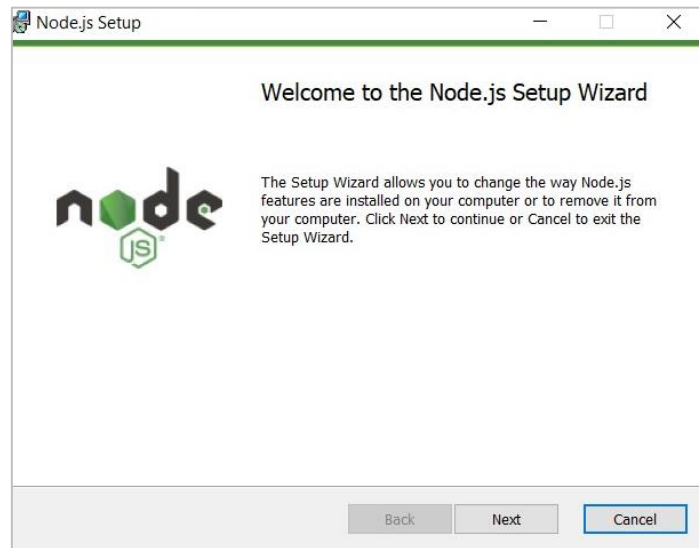
Download for Windows (x64)

16.17.0 LTS Recommended For Most Users	18.7.0 Current Latest Features
--	--

[Other Downloads](#) | [Changelog](#) | [API Docs](#) [Other Downloads](#) | [Changelog](#) | [API Docs](#)

Or have a look at the [Long Term Support \(LTS\) schedule](#)

8. Node.js setup process will start, click on “Next” to continue the setup to the completion as shown in the figures below.



9. Once installed, open a command prompt and run the following command to ensure Node.js is installed correctly.

10. Type the command “node -v” to view the version of the installed node red. You should receive back output that looks similar to:

```
v14.17.2.0  
6.14.13
```

11. Installing Node-RED as a global module adds the command node-red to your system path. Execute the following at the command prompt:

```
npm install -g --unsafe-perm node-red
```

12. Now close the command prompt, and reopen it.
13. To enable npm to compile binaries on the Windows platform, install the windows-build-tools module using the command prompt as an Administrator:

```
npm install --global --production windows-build-tools
```

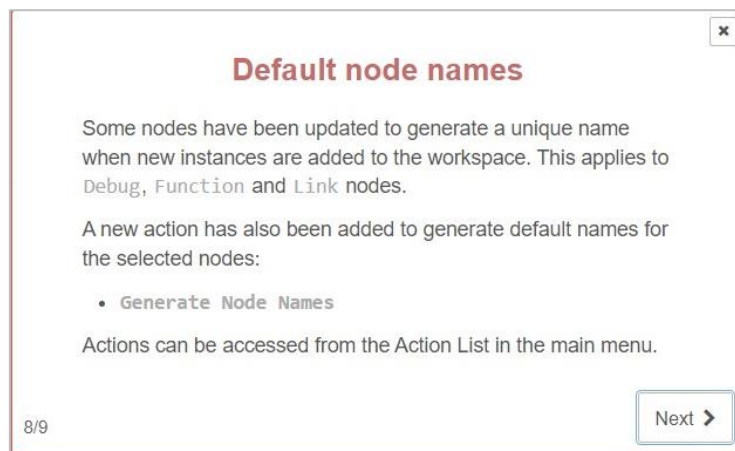
14. Type the command “node-red”, a server link will be displayed as shown below. Copy and paste the link to any browser and press enter as shown in the figures below.

```
Your flow credentials file is encrypted using a system-generated key.  
  
If the system-generated key is lost for any reason, your credentials  
file will not be recoverable, you will have to delete it and re-enter  
your credentials.  
  
You should set your own key using the 'credentialSecret' option in  
your settings file. Node-RED will then re-encrypt your credentials  
file using your chosen key the next time you deploy a change.  
-----  
23 Aug 13:19:05 - [warn] Encrypted credentials not found  
23 Aug 13:19:05 - [info] Server now running at http://127.0.0.1:1880/  
23 Aug 13:19:05 - [info] Starting flows  
23 Aug 13:19:05 - [info] Started flows  
23 Aug 13:19:05 - [info] [serialconfig:68aca14478b92a23] serial port COM22 opened at 9600 baud 8N1  
23 Aug 13:50:39 - [error] [serialconfig:68aca14478b92a23] serial port COM22 closed unexpectedly  
23 Aug 13:50:54 - [info] [serialconfig:68aca14478b92a23] serial port COM22 opened at 9600 baud 8N1
```

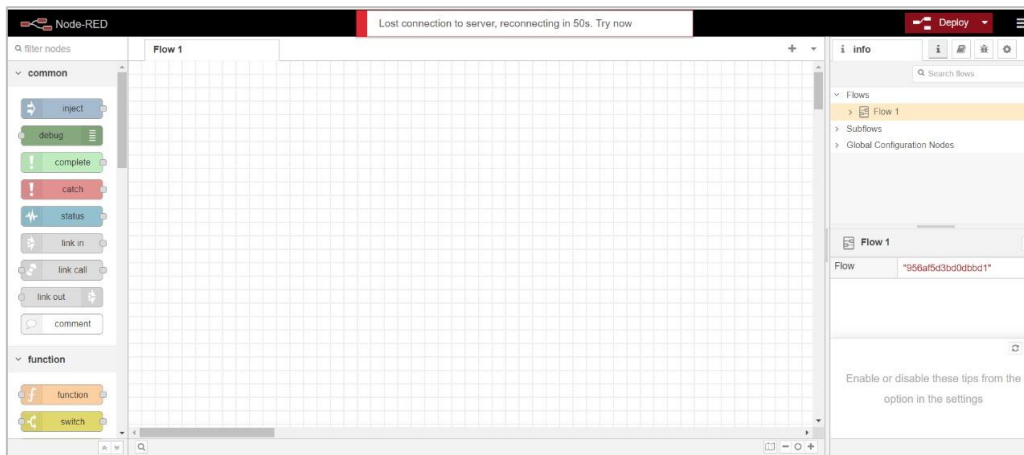


http://127.0.0.1:1880/

15. The default node names window will be displayed, click on “Next” as shown in the figure below.



16. Now everything is setup and the user can start with the device.



4.2.2 Integration Guidelines

Connect the USB to the device through the USB port in the front panel (Micro-USB type), and connect the other side with personal computer (PC). Once the USB is connected correctly between the device and PC, the user can start the integration as explained in the Integration Guidelines:

1. Upload the desired code on the device by using the Arduino IDE or Visual Studio or any other platform.
Click here for more detailed guidelines [PLC Software](#).

Note: the following code example aims to define Pin No.4 as an input and monitor its status between ON/OFF when an input device is connected to it, as shown in the figure below.

```
sketch_aug03a | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help

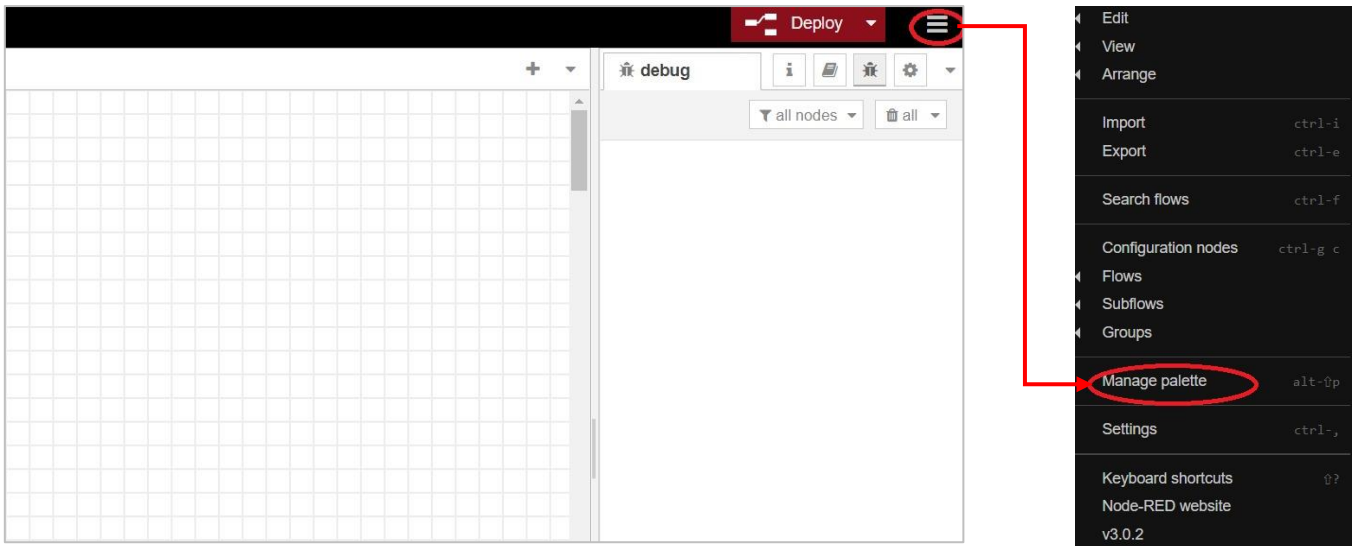
sketch_aug03a §
#include <Arduino.h>

int ledPin = 4; // LED connected to digital pin 4

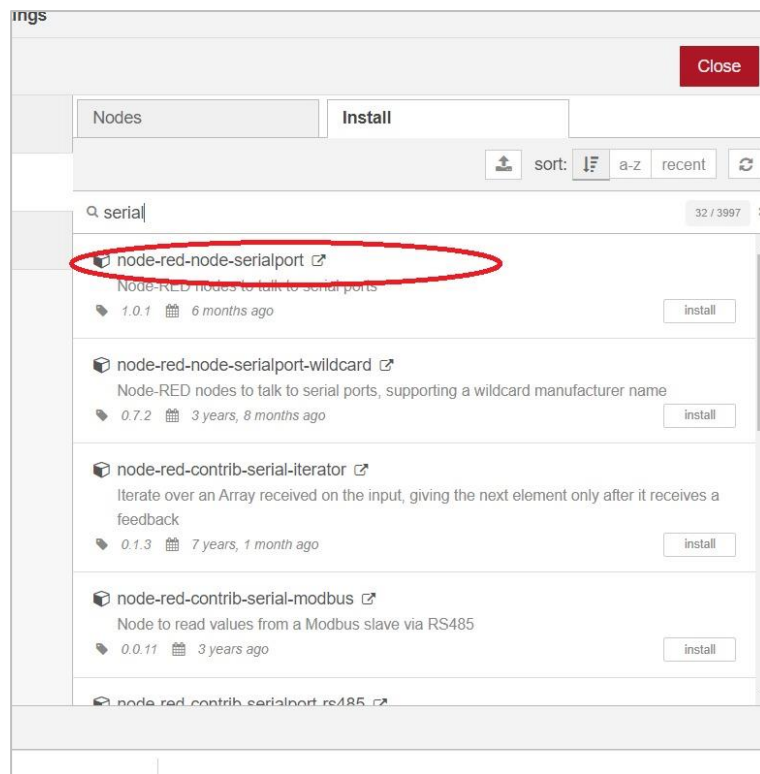
void setup()
{
  Serial.begin(9600);
  SerialUSB.begin(9600);
  pinMode(ledPin, INPUT_PULLUP); // sets the digital pin 4 as INPUT
  delay(2000);
  SerialUSB.println("hello");
}

void loop()
{
  int val = digitalRead(ledPin); // read the input pin
  if (val == HIGH)
  {
    SerialUSB.println(0);
  }
  else
  {
    SerialUSB.println(1);
  }
}
```

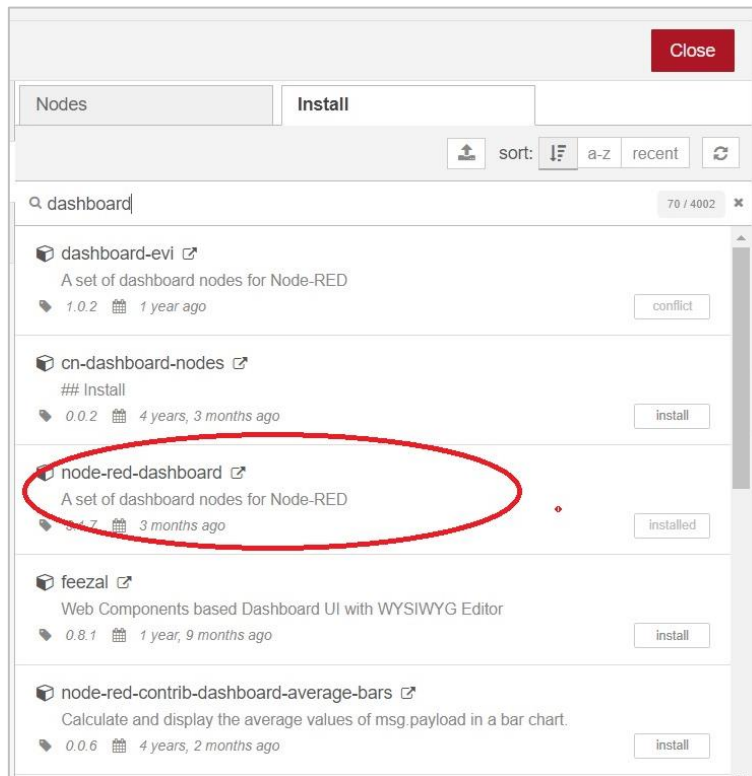
2. Once the code is successfully uploaded on PMM0620, close Arduino IDE or any other used platform.
Note: It is not possible to have two different applications sharing the same serial port connection.
3. Open Node-RED platform, and click on “Manage Palette” to add the needed modules as shown below.



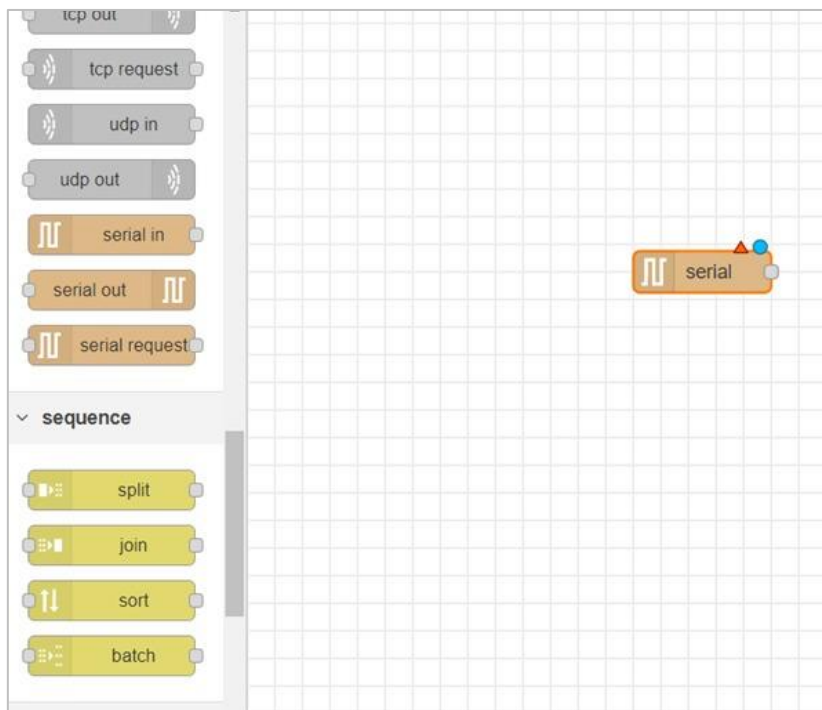
4. Search for “Serial” options then choose “node-red-node-serialport” as shown below.
5. Click on “install”.



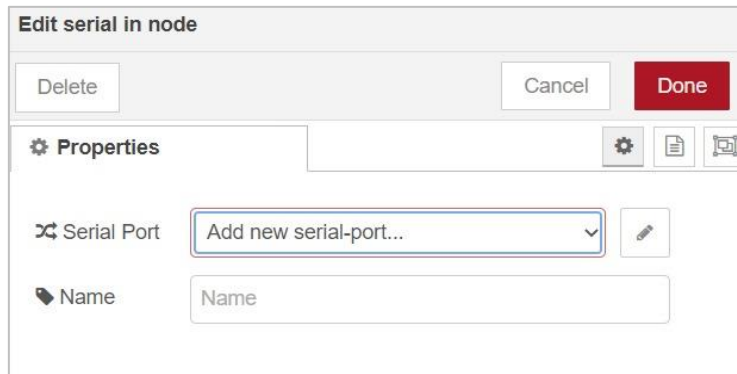
6. Search for “dashboard” module that opens new blocks which display the input value (0 or 1), then choose “node-red-dashboard”.
7. Click on “install”.



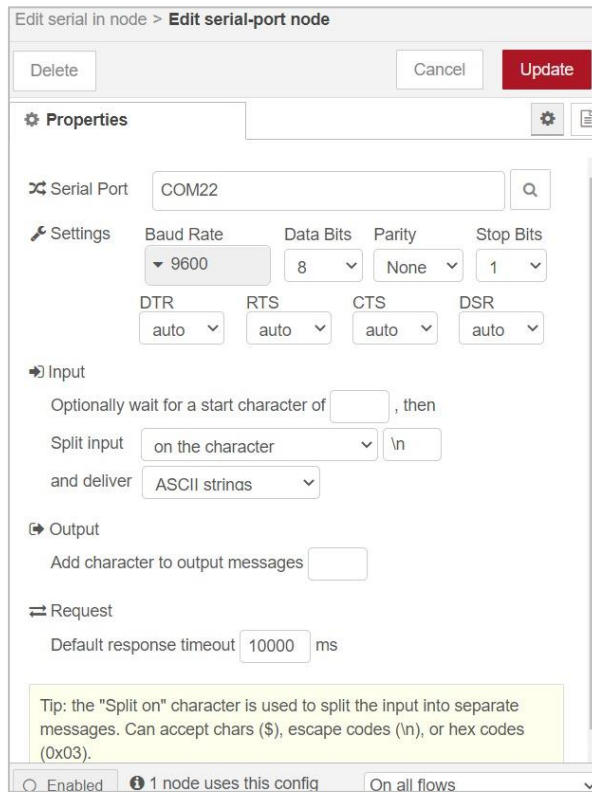
8. The installed modules will be found in the workspace.
9. Drag the serial port block and drop it where it is needed.



10. Double click on the serial module to edit the properties.



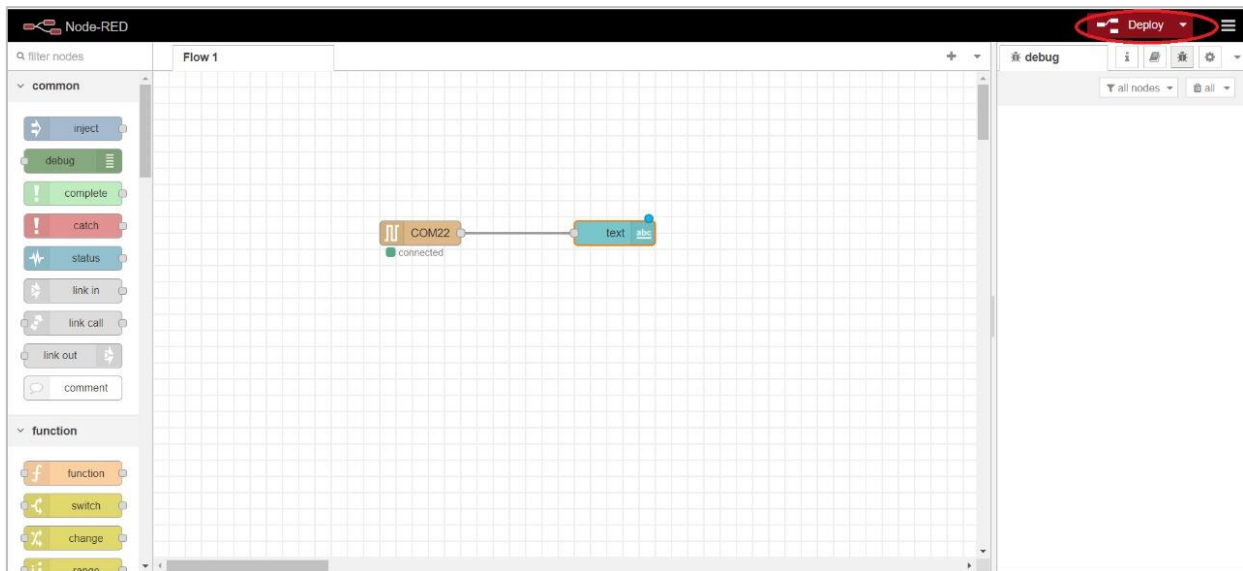
11. Type the desired name in the specified space.
12. Click on the “pin icon” to edit the properties. The following window will be displayed.



13. Choose from the drop lists the desired settings based on the code such as baud rate, data bits, parity and stop bits.
14. Once the port configuration is completed click on “update”
15. Drag the dashboard block and drop it where it is needed.
16. To edit the “Dashboard” block name and settings double click on the block.
17. The following window will be displayed.

18. Click on the “pen icon” the following window will be displayed.


19. Make sure that “Display group name” and “allow group to be collapsed” are both enabled.
20. After completing the settings click on “Done”.
21. Connect the two blocks together in the workspace.
22. And click on “Deploy”.



23. Open the user interface page to monitor the status of the input.
24. Copy the same http address that has been used.

 http://127.0.0.1:1880/

25. Type “ui” after the / and press enter.

 http://127.0.0.1:1880/ui

26. Connect a digital input with PMM0620 to pin no.4 to monitor its status. If the input is on the text value will be **1** if it is off the text value will be **0**.



Appendix 5:

PMM06 Integration with OpenPLC

1.1 Description

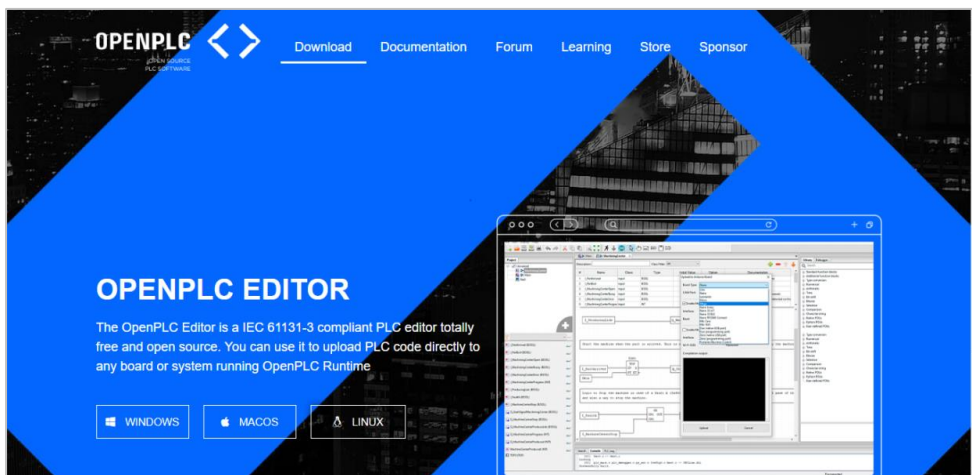
PMM PLC Systems are built to be Arduino compatible programming environment, where PMM’s PLCs Range is not just compatible with Arduino IDE, but with lots of other Arduino-compatible programming software such as openPLC.

OpenPLC is an independent organization providing efficiency in industrial automation based on the needs of users. OpenPLC members have concentrated on technical specifications around IEC 61131-3, creating specifications and implementations in order to reduce cost in industrial engineering. The outcome for example is standardized libraries for different application fields, harmonized language conformity levels and engineering interfaces for exchange. Experts of the OpenPLC members are organized in technical committees and together with end users define such open standards.

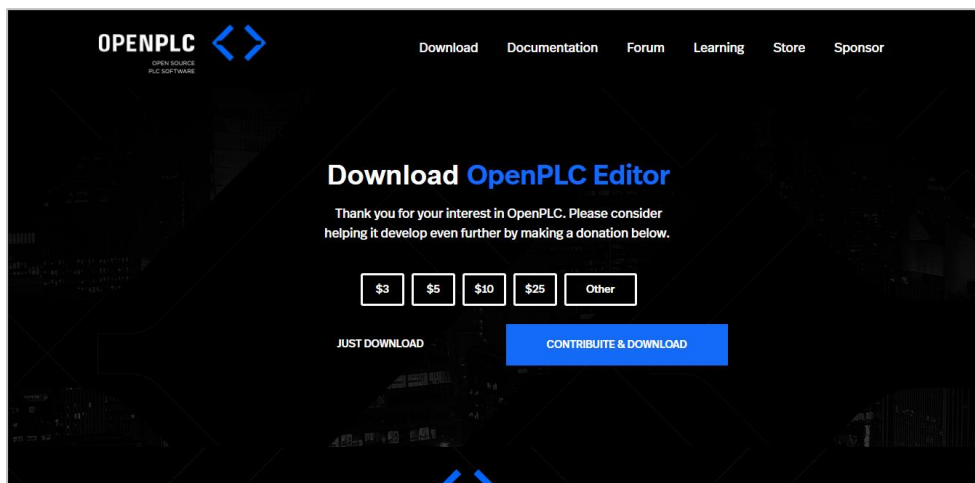
5.2 INTEGRATION GUIDELINES

5.2.1 OpenPLC Installation Guidelines

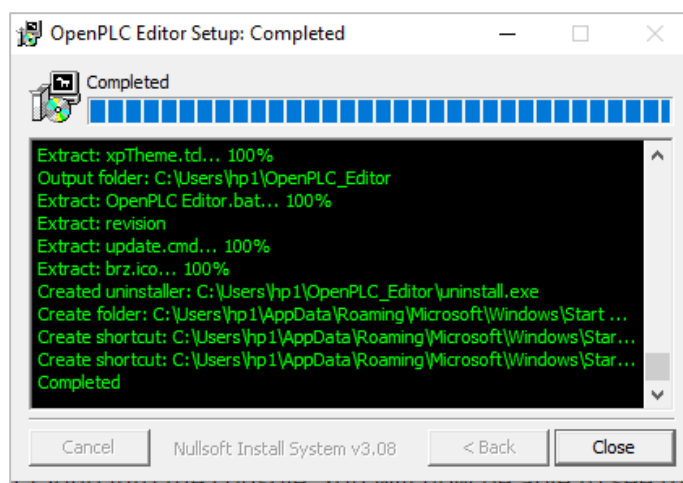
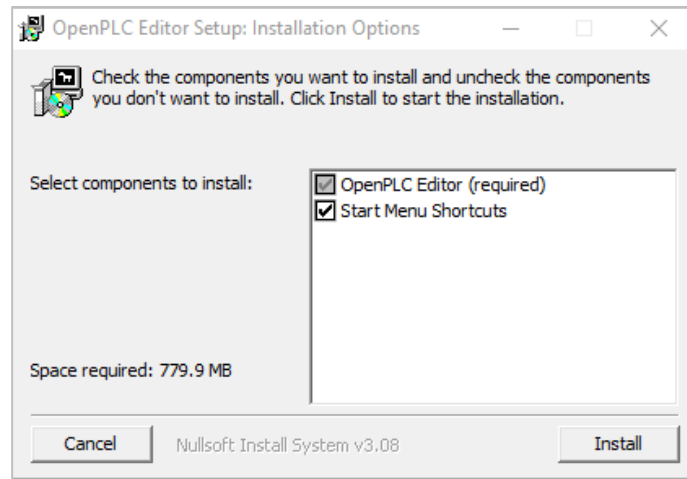
1. [Click Here](#) to get to the installation page.
2. Once the main page is opened, click on the compatible link with your device.



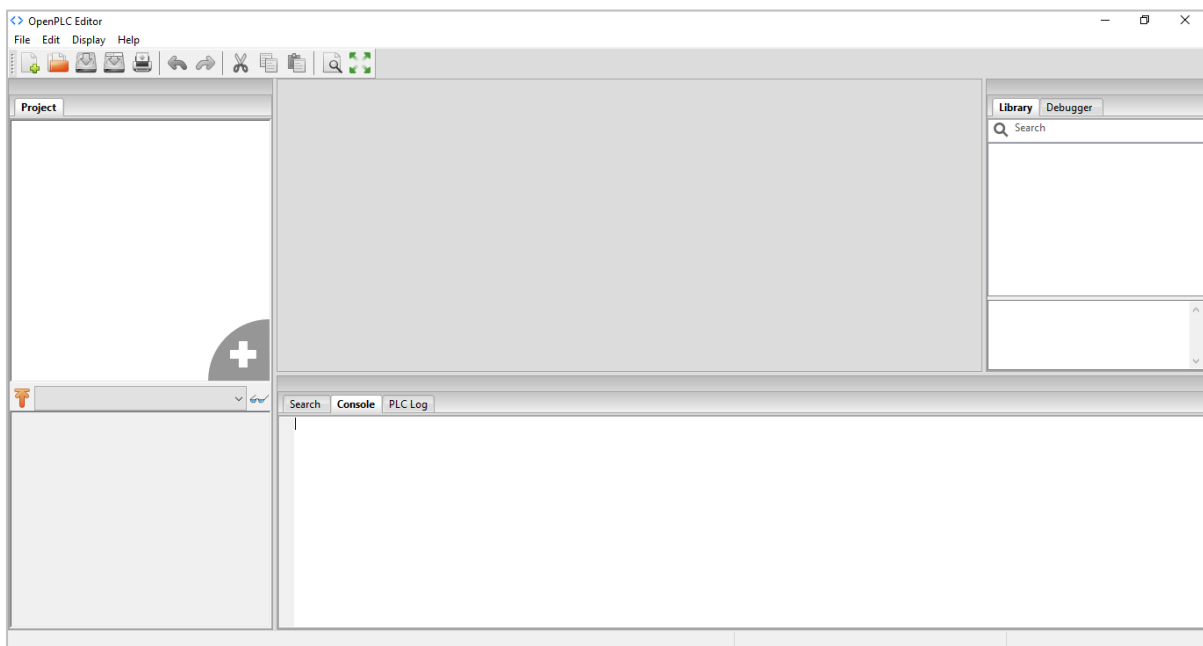
4. Click on “open file” to proceed to installation.



5. OpenPLC setup process will start, click on “Install” to continue the setup to the completion as shown in the figures below.



6. Now everything is setup and the user can start using the program.

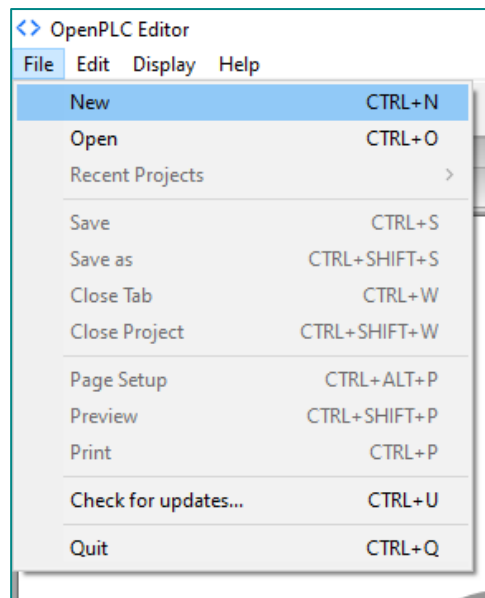


5.2.2 Integration Guidelines

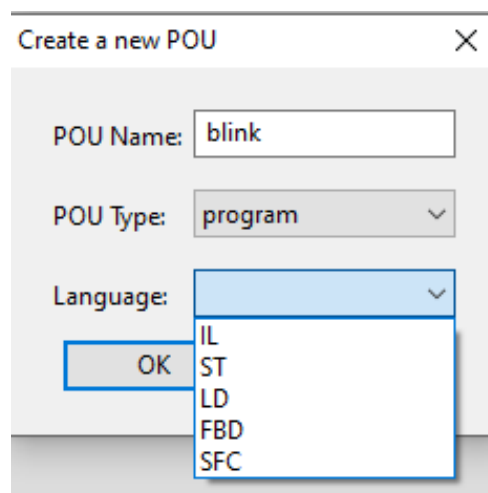
Connect the USB to the device through the USB port in the front panel (Micro-USB type), and connect the other side with personal computer (PC). Once the USB is connected correctly between the device and PC, the user can start the integration as explained in the Integration Guidelines:

1. To open a new project: select “New”, then select the destination file.

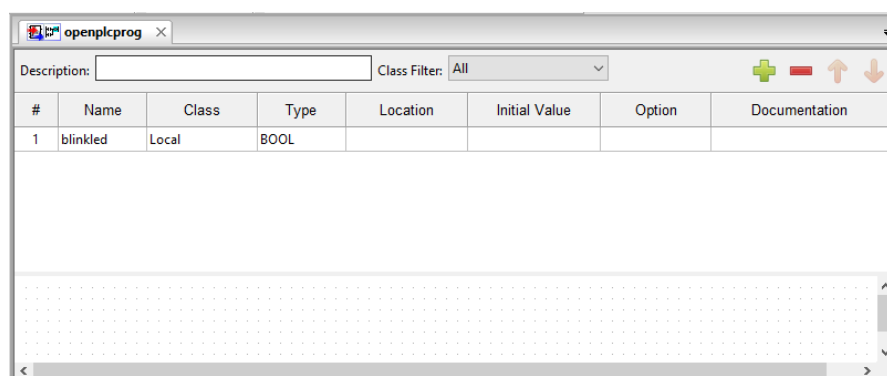
Note: the folder of destination should be empty.



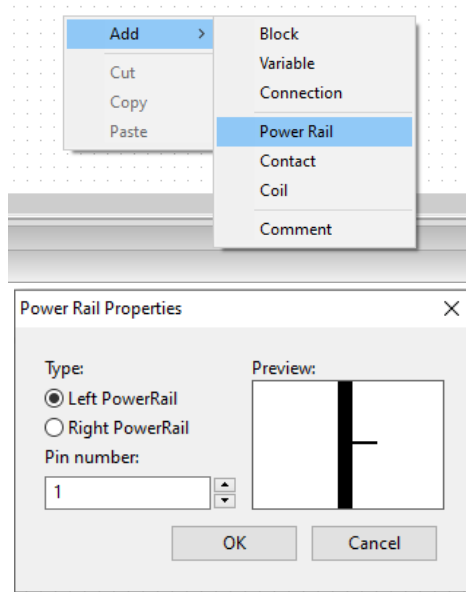
2. Name your file then choose the desired programming language.



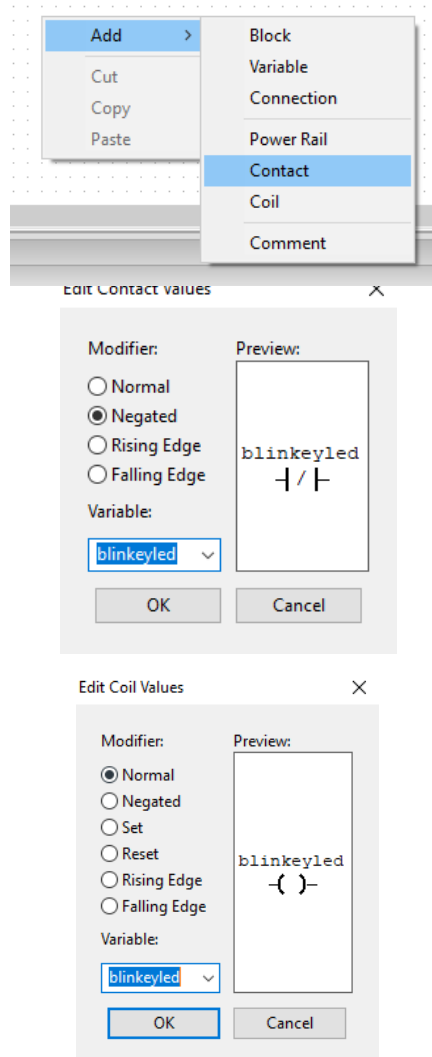
3. Now press on the green plus icon to create a variable, choose the name and the desired type.

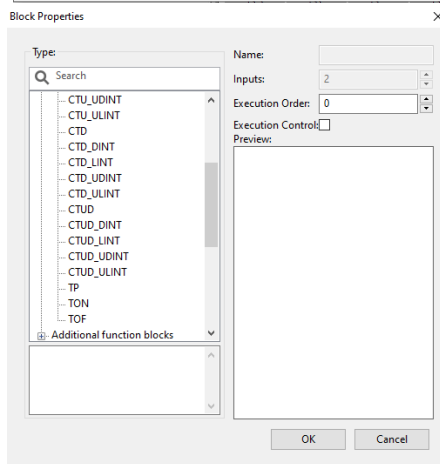
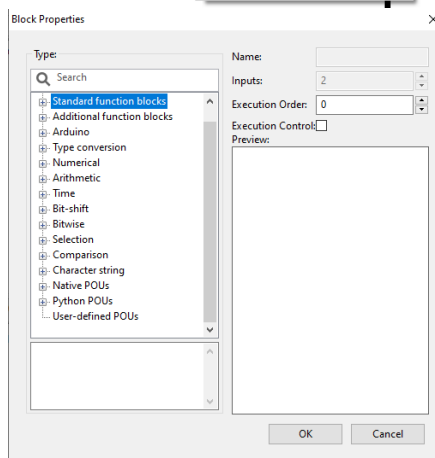
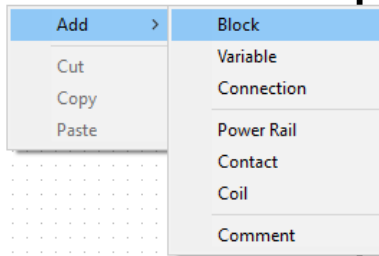
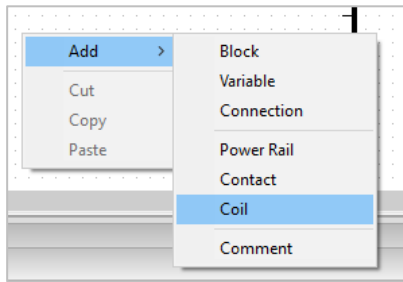


4. Now add the power rails in both left and right, and select the pin number.



5. You can add the contact and coil by right clicking and pressing on the needed one.





- To know the location of the pins, go to the website >documentation >openPLC runtime >physical addressing >zero >digitaloutput>

OPENPLC OPEN SOURCE PLC SOFTWARE


Download Documentation Forum Learning Store Sponsor

1. Get Started ^ > 1. Get Started > **1.1 OpenPLC Overview**

1.1 OPENPLC OVERVIEW

OpenPLC is an open-source **Programmable Logic Controller** that is based on an easy to use software. It is the first fully functional standardized open source PLC, both in software and in hardware. The OpenPLC project was created in accordance with the IEC 61131-3 standard, which defines the basic software architecture and programming languages for PLCs.

OpenPLC is mainly used on industrial and home automation, internet of things and SCADA research. You can check OpenPLC in action on the short video below:



1. Get Started ^ > 2. OpenPLC Runtime > **2.4 Physical Addressing**

2.4 PHYSICAL ADDRESSING

OpenPLC Runtime is compatible with several different hardware platforms with different I/O module configurations. Internally, all I/O variables are associated with a PLC Address, as explained on **2.3 Input, Output and Memory Addressing**. The hardware layer is the component responsible for translating internal PLC Address variables into physical hardware locations. Each platform OpenPLC supports must have a different hardware layer. Below you can find the pinout description for each platform, extracted from the platform's hardware layer file:

Microcontroller-based boards

Controllino Maxi

Digital In	A4, A5, A6, A7, A8, A9, I00, I01	%IX0.0 - %IX0.7
Digital out	D4, D5, D6, D7, D8, D9, D10, D11 R0, R1, R2, R3, R4, R5, R6, R7 R8, R9	%Q0.0 - %Q0.7 %Q1.0 - %Q1.7 %Q2.0 - %Q2.1
Analog In	A0, A1, A2, A3	%IW0 - %IW3
Analog out	D0, D1, D2, D3	%QW0 - %QW3

Uno, Leonardo, Nano, Micro, Zero

Digital In	2, 3, 4, 5, 6	%IX0.0 - %IX0.4
Digital Out	7, 8, 12, 13	%QX0.0 - %QX0.3
Analog In	A0, A1, A2, A3, A4, A5	%IW0 - %IW5
Analog Out	9, 10, 11	%QW0 - %QW2

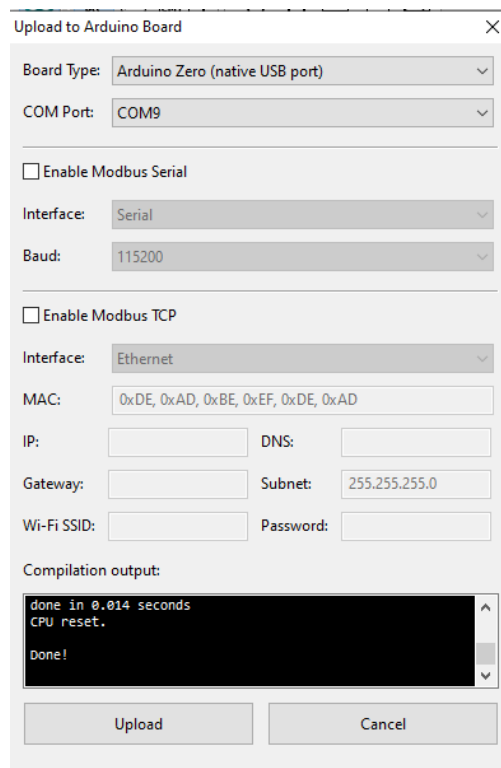
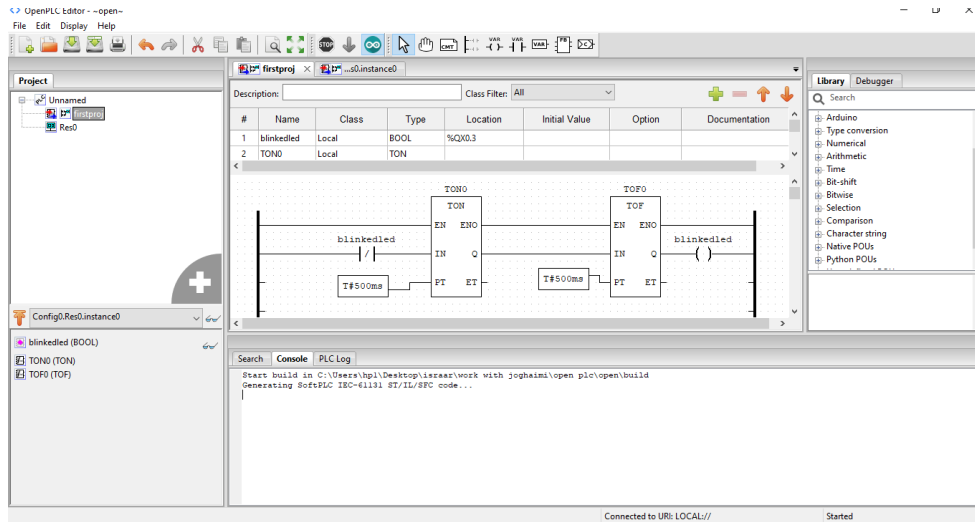
7. Write down the location in the box

firstproj x ...s0.instance0

Description: Class Filter: All

#	Name	Class	Type	Location	Initial Value	Option	Documentation
1	blinkled	Local	BOOL	%QX0.3			
2	TON0	Local	TON				

8. After assembling your project press on the Arduino sign to upload the program to your device



9. Now the program should be running on your device.

Appendix 6:

PMM06 Integration with Scilab

7.1 Description

PMM PLC Systems are built to be Arduino compatible programming environment, where PMM’s PLCs Range is not just compatible with Arduino IDE, but with lots of other Arduino-compatible programming software such as Scilab.

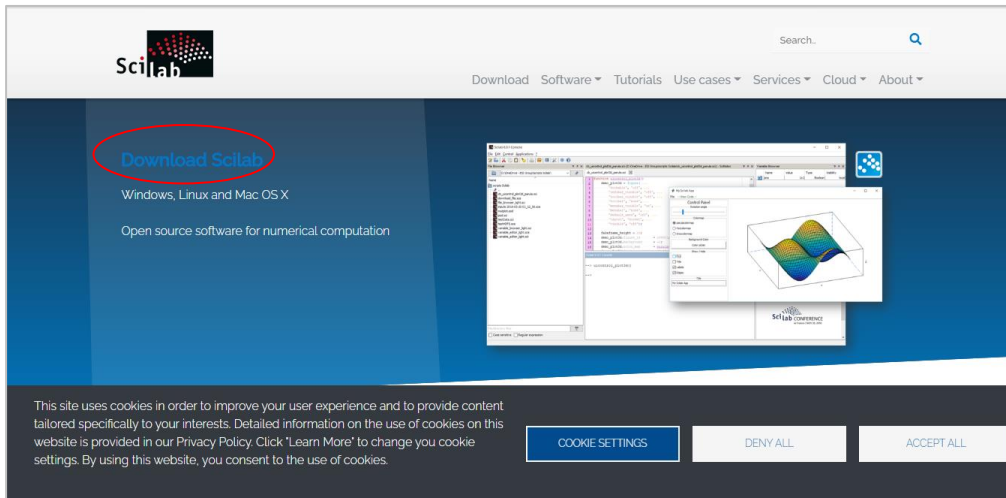
Scilab is a free and open-source, cross-platform numerical computational package and a high-level, numerically oriented programming language. It can be used for signal processing, statistical analysis, image enhancement, fluid dynamics simulations, numerical optimization, and modeling, simulation of explicit and implicit dynamical systems and (if the corresponding toolbox is installed) symbolic manipulations.

Scilab is one of the two major open-source alternatives to MATLAB, the other one being GNU Octave. Scilab puts less emphasis on syntactic compatibility with MATLAB than Octave does, but it is similar enough that some authors suggest that it is easy to transfer skills between the two systems.

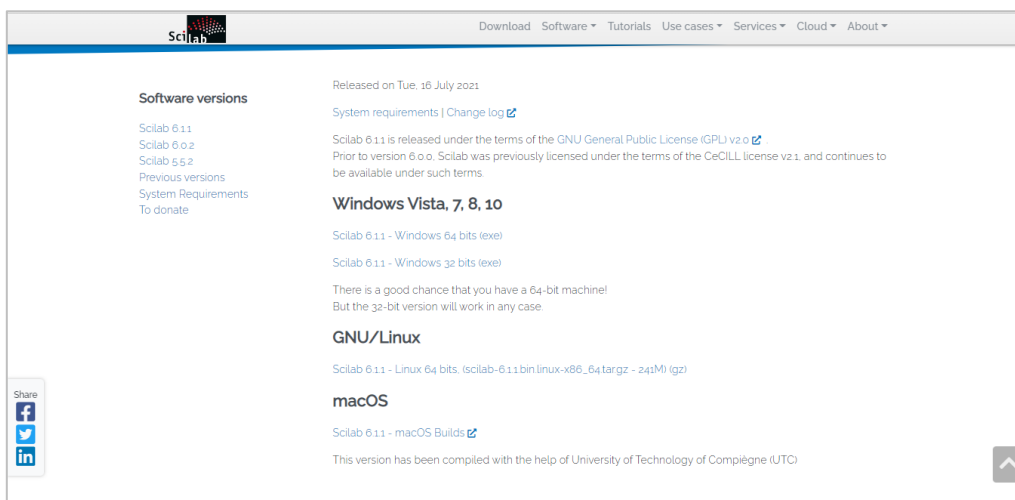
6.2 INTEGRATION GUIDELINES

6.2.1 Scilab Installation Guidelines

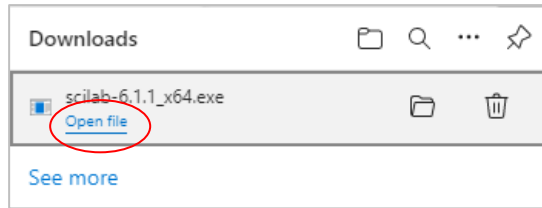
1. [Click Here](#) to get to the installation page.
2. Once the main page is opened, click on “Download Scilab”.



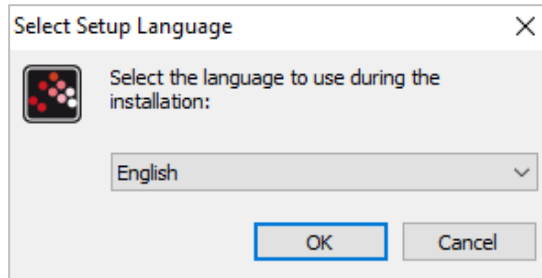
7. Choose the compatible version for your device.



8. Click on “open file” to proceed to installation.



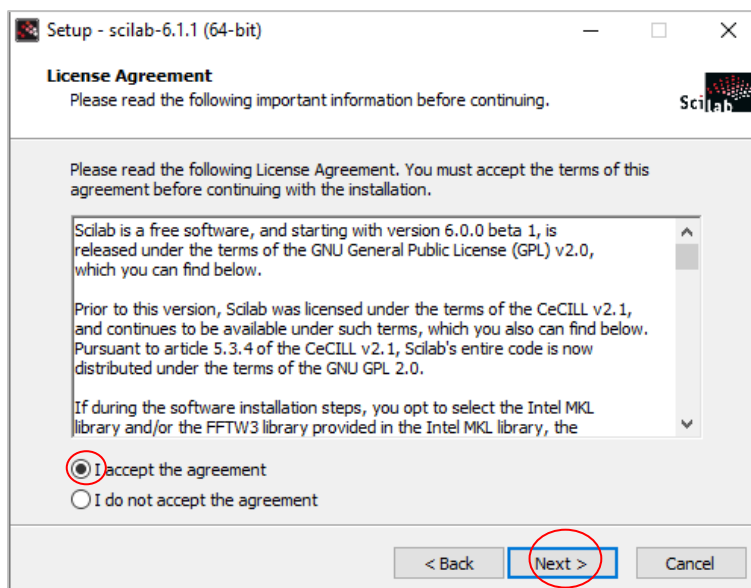
9. Select the desired language during installation> ok.



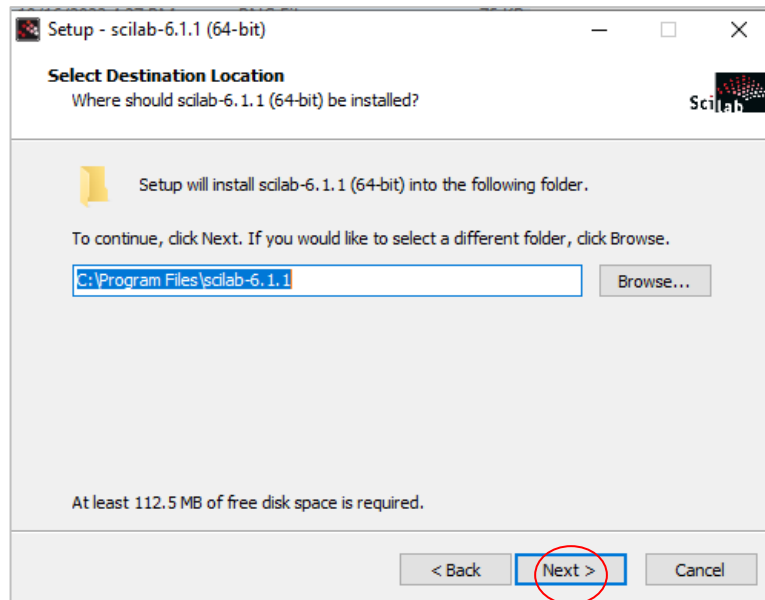
10. Scilab setup process will start, click on “Next” to continue the setup to the completion as shown in the figures below.



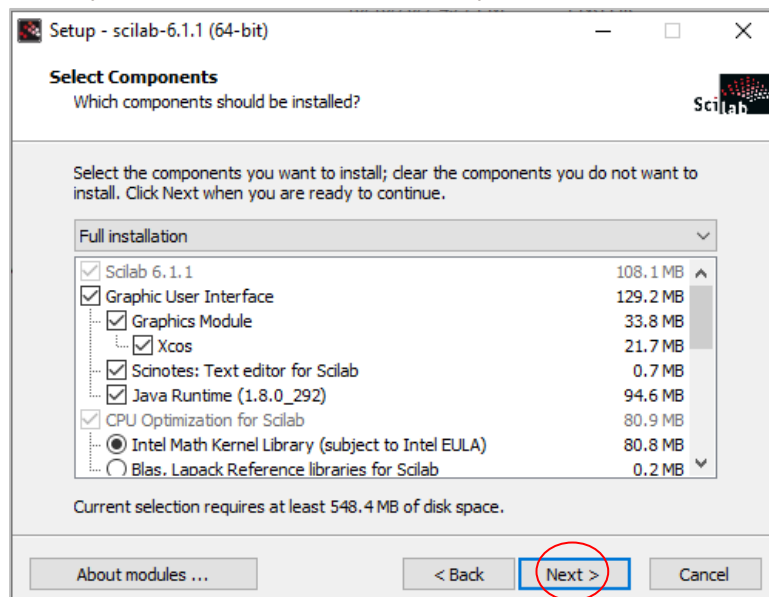
11. Choose” I accept the agreement” and hit next.



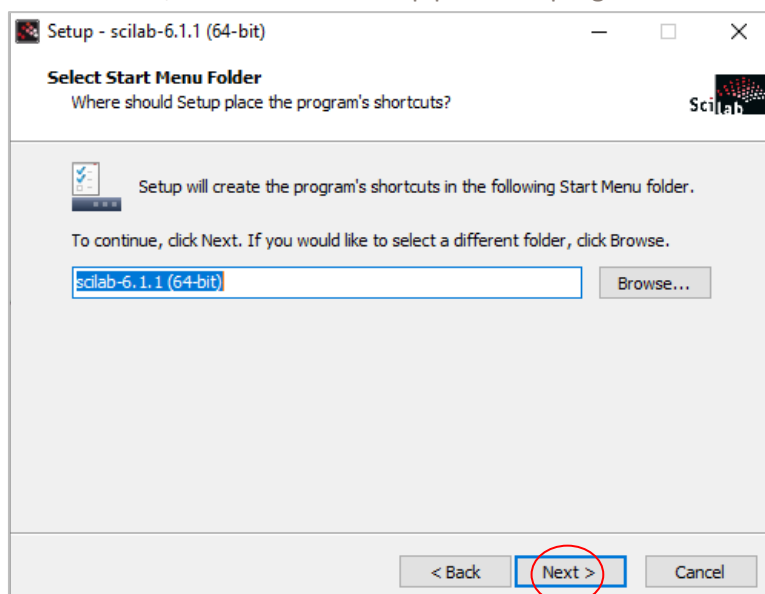
12. Select the desired destination file location and press on “Next”.



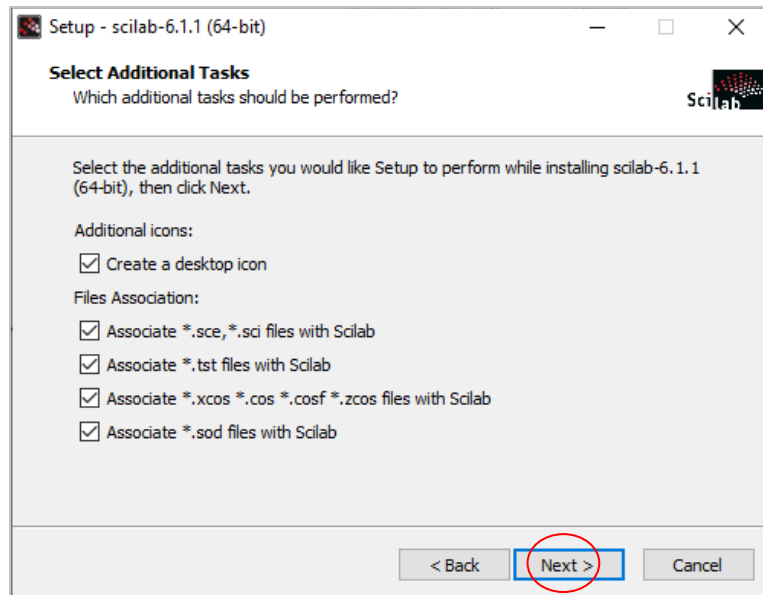
13. Select which components should be installed and press on “Next”.



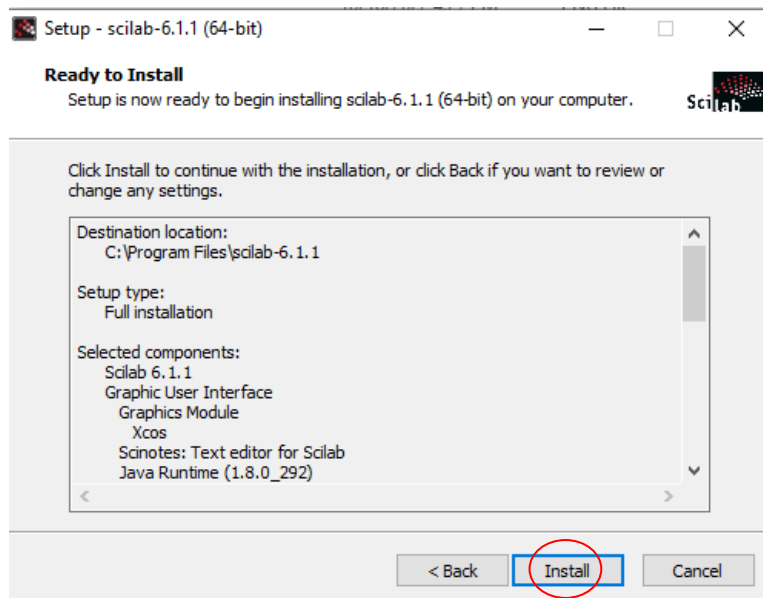
14. Select start menu folder, “where should setup place the program shortcuts?” and press on “Next”.



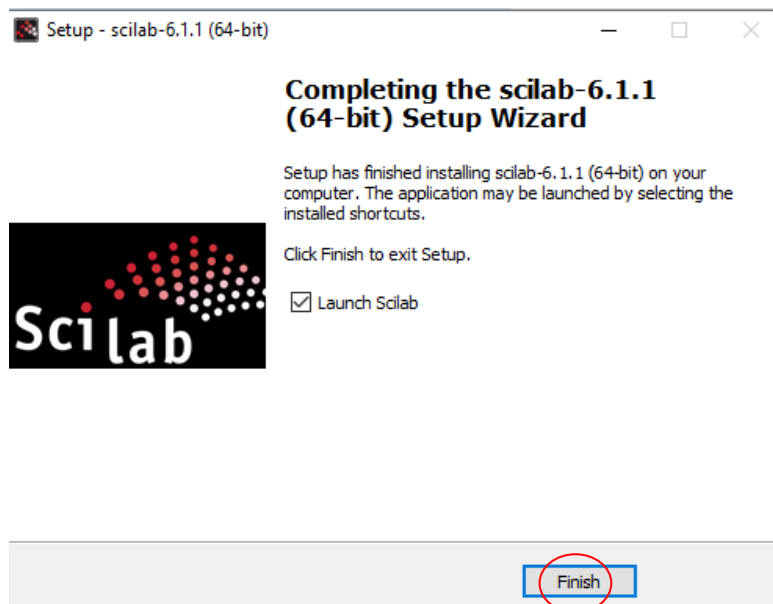
15. Select additional tasks you wish to be performed and press “Next”.



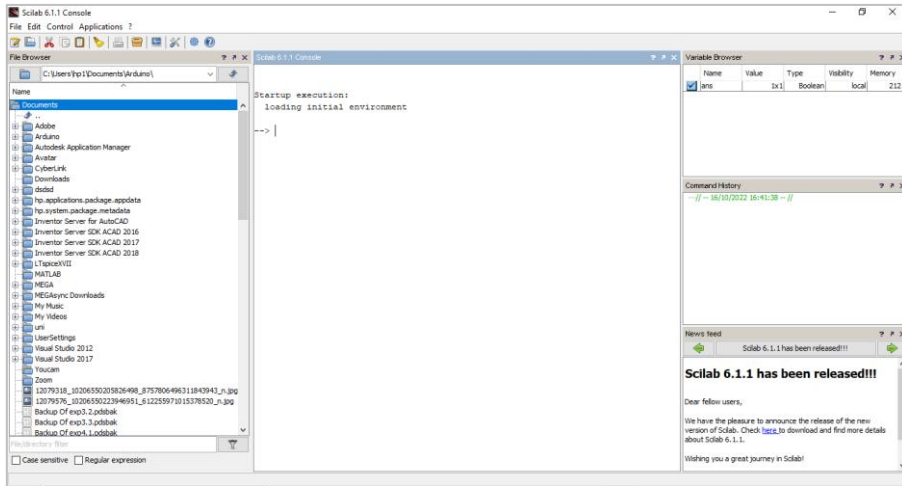
16. Now the program is ready to install, click on the “install” button.



17. Installation is done press on “Finish”.



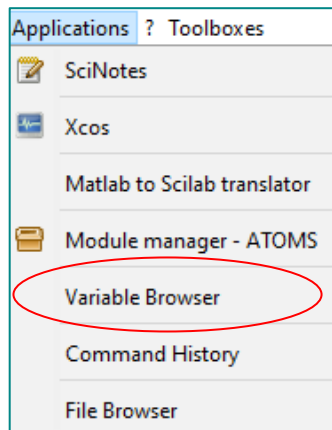
18. Now everything is setup and the user can start using the program.



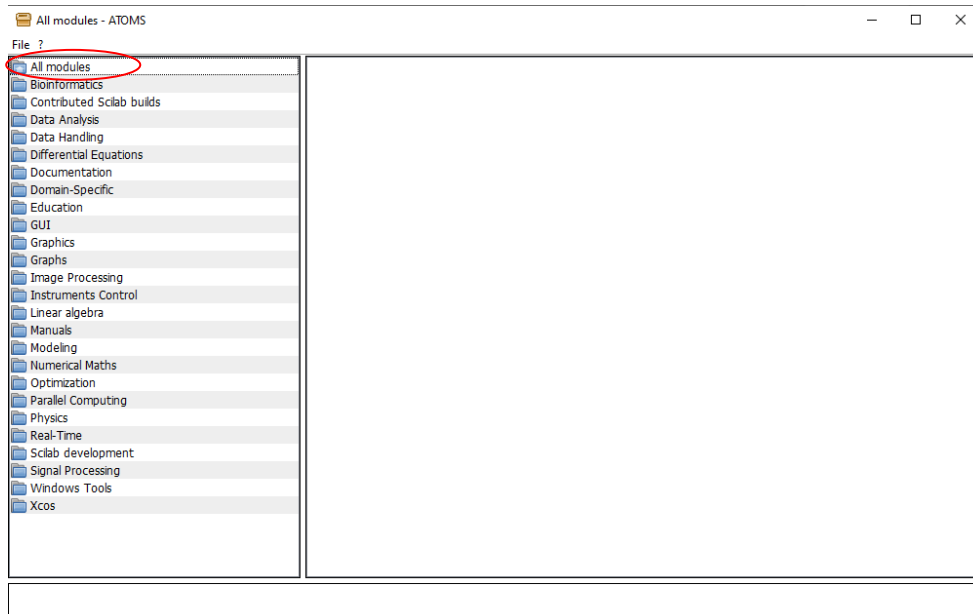
6.2.2 Integration Guidelines

Connect the USB to the device through the USB port in the front panel (Micro-USB type), and connect the other side with personal computer (PC). Once the USB is connected correctly between the device and PC, the user can start the integration as explained in the Integration Guidelines:

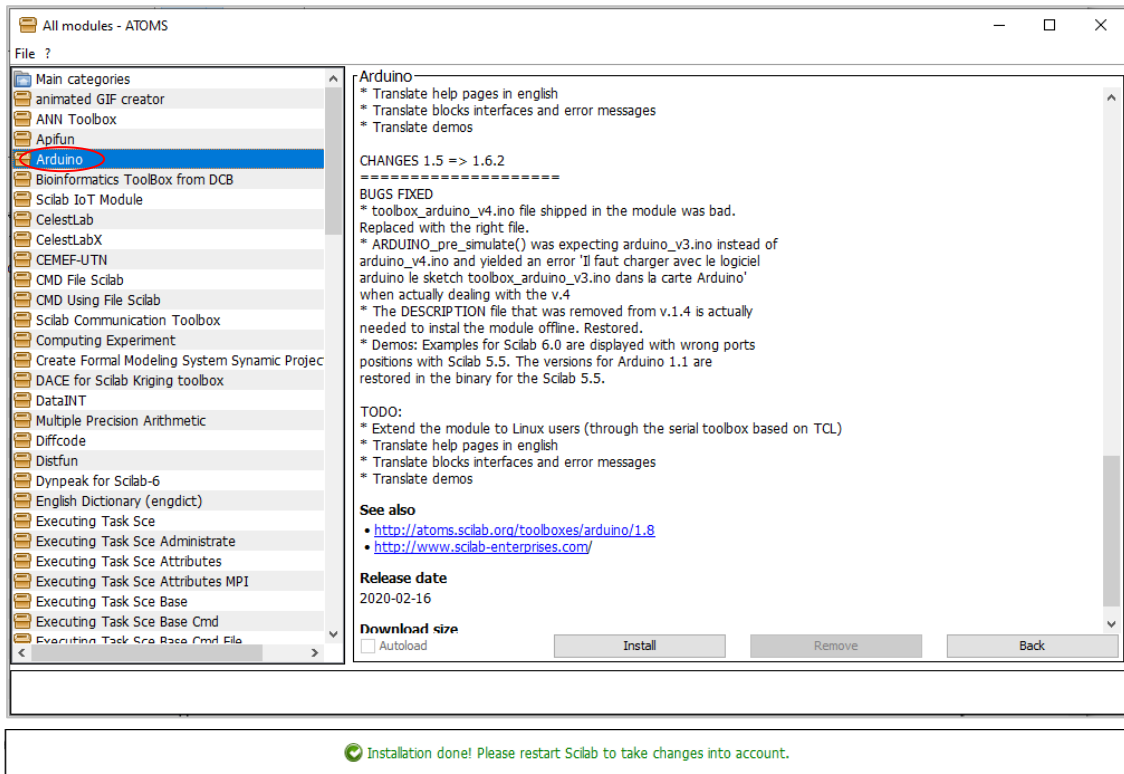
8. To connect our board with Scilab from the tool bar select “applications” then “Module manager - ATOMS”.



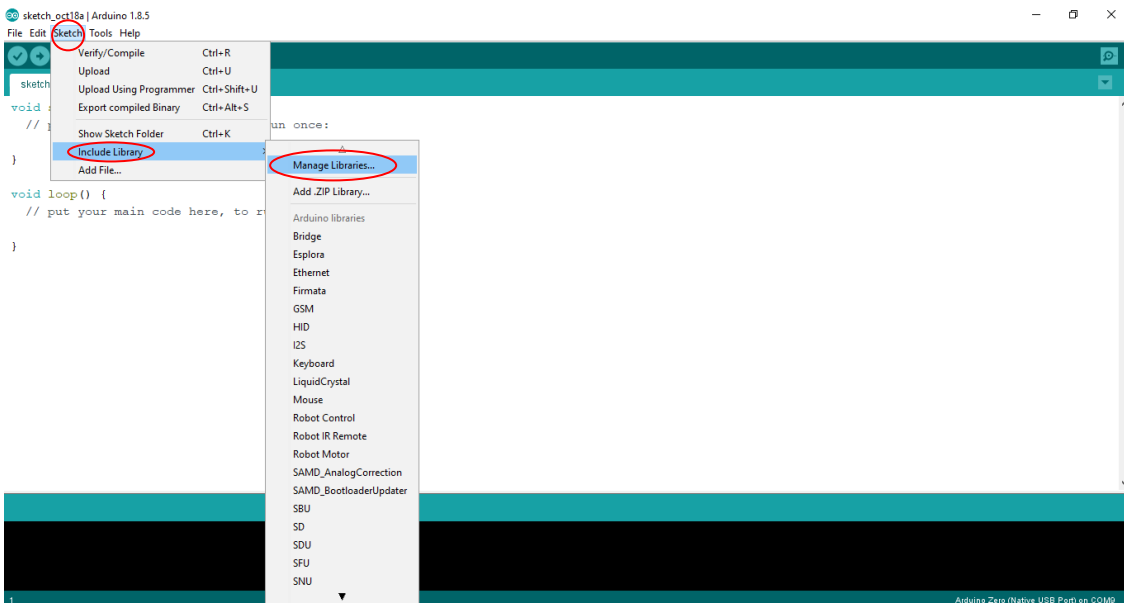
9. Double click on” All modules”.



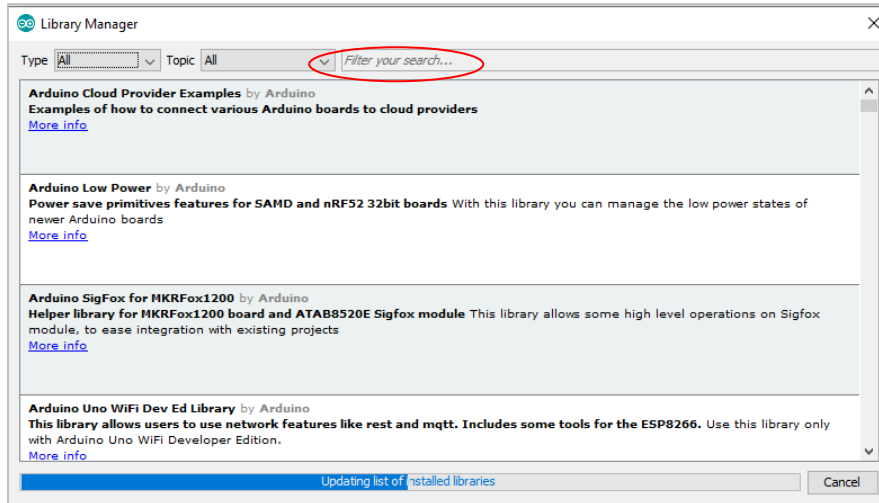
10. Select Arduino and press on " Install".



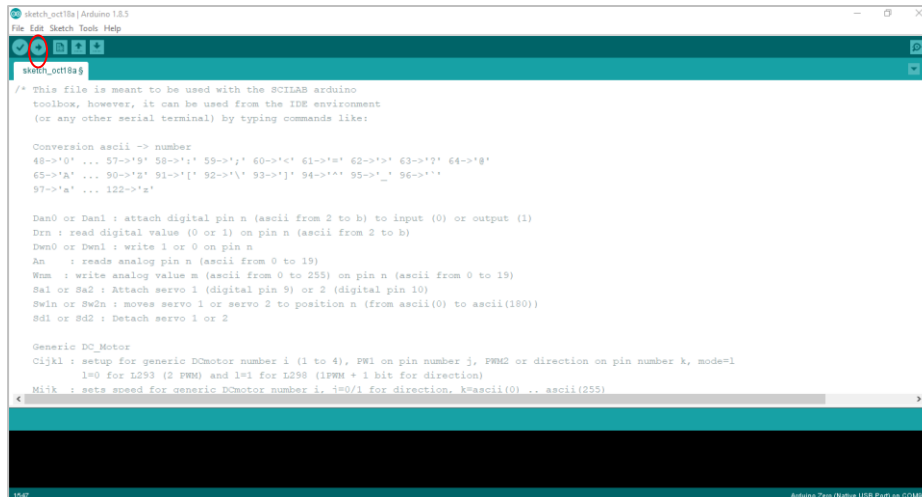
11. Now open the Arduino app from the tool bar go to "sketch>include library >manage libraries".



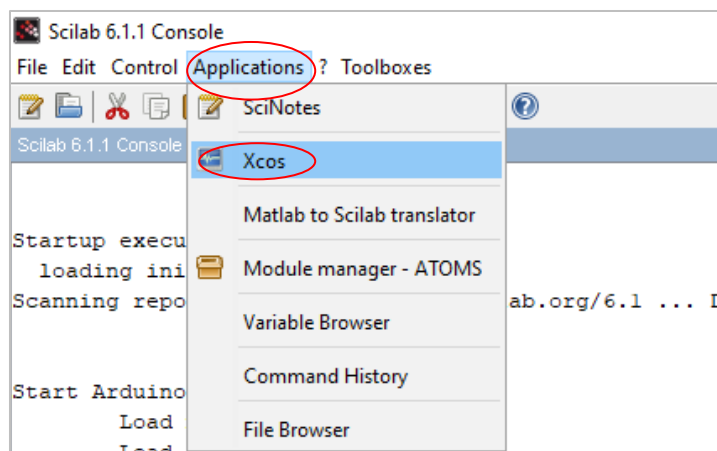
12. Now search for the following libraries (servo.h and mpu6050) and install them.



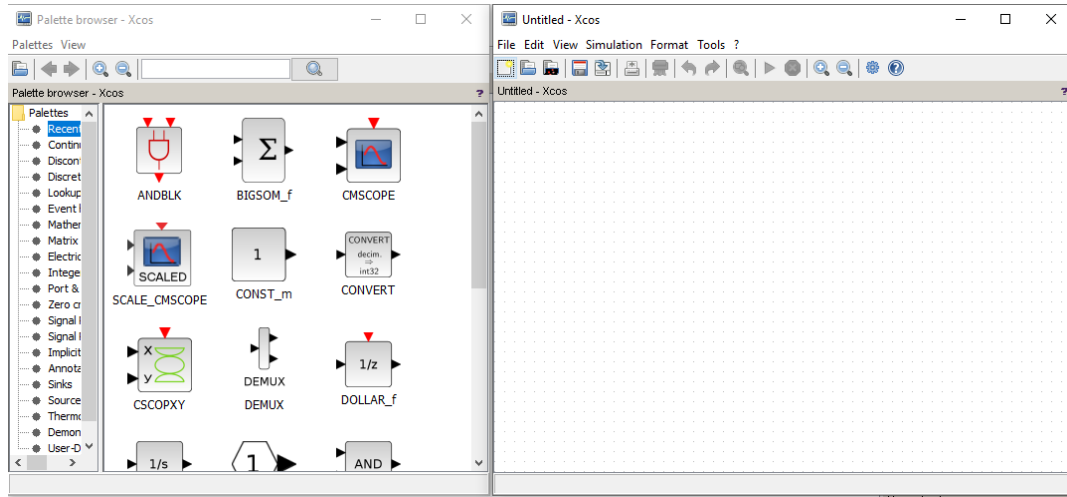
13. Once they are successfully installed, [click here](#) then copy the code and compile it on the Arduino app to your device.



14. Now that the compiling is done go back to Scilab and from the tool box > applications>Xcos.

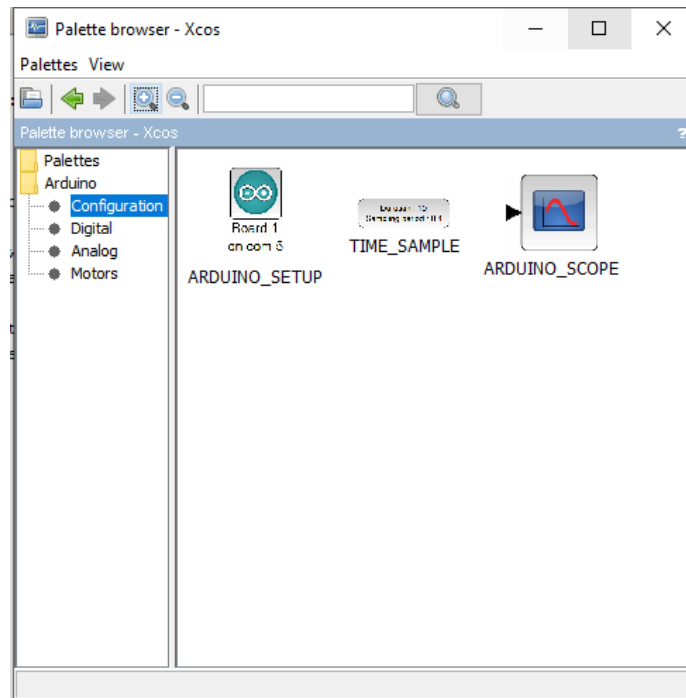


15. The following pages should be on your screen, to add your circuit.



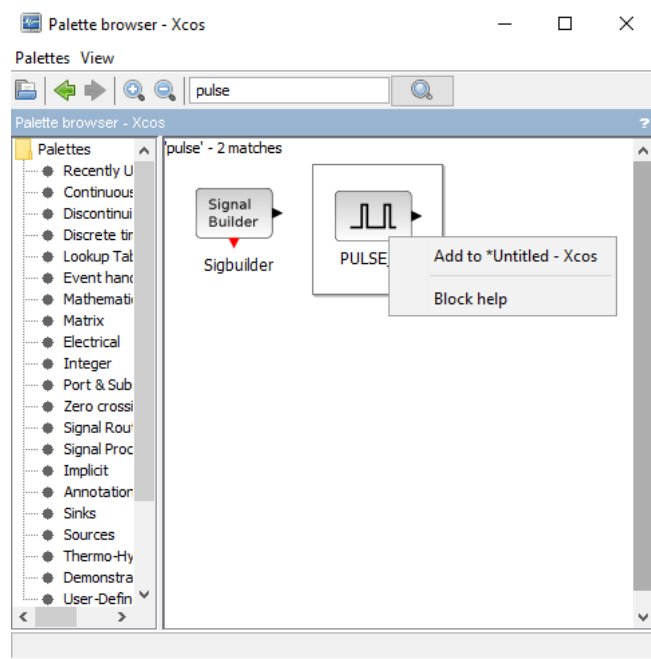
16. To select your desired functions >select Arduino > Arduino setup.

17. to add you board right click on the desired item > “add to untitled-xcos”.

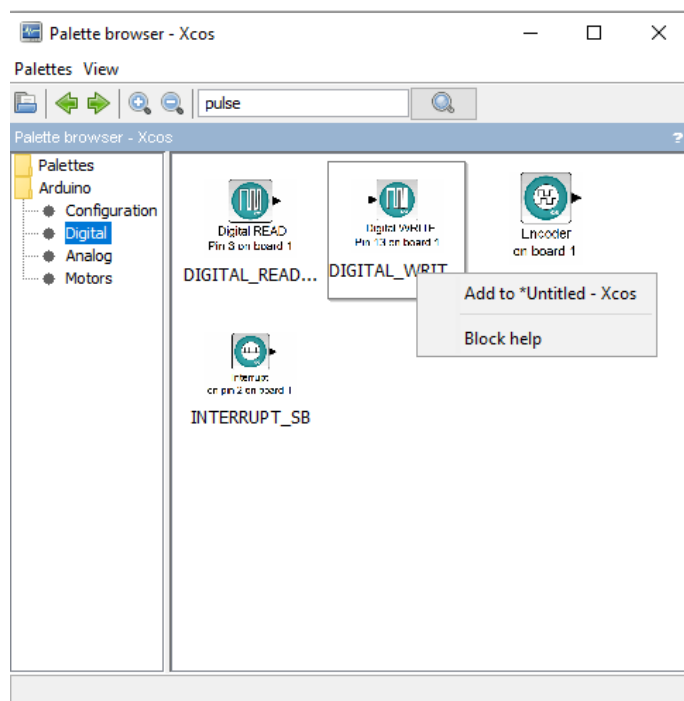


18. From the search option type your desired function.

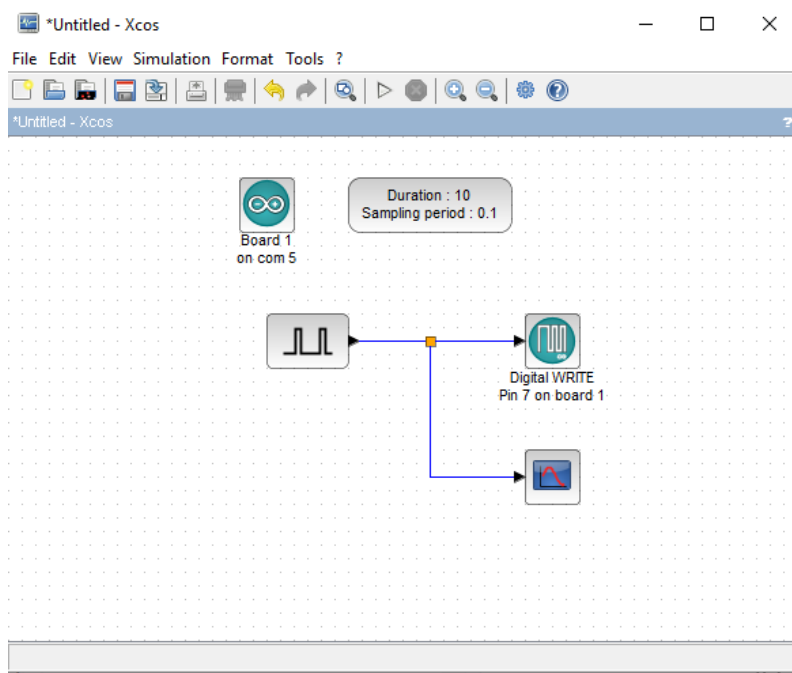
Note: in this example we made a led blink.



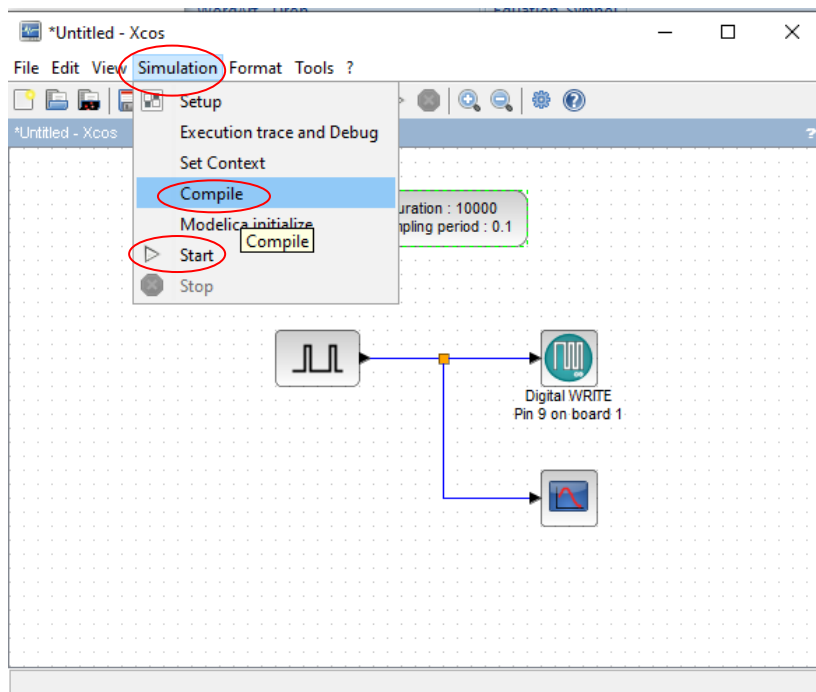
19. Do the same for pin options.



20. After adding your functions, double click on each one to edit and connect them by pressing on each item and dragging a line to the other.



21. From simulation > compile > start
And the program should be running



Appendix 7:

PMM06 Integration with PlatformIO

7.2 Description

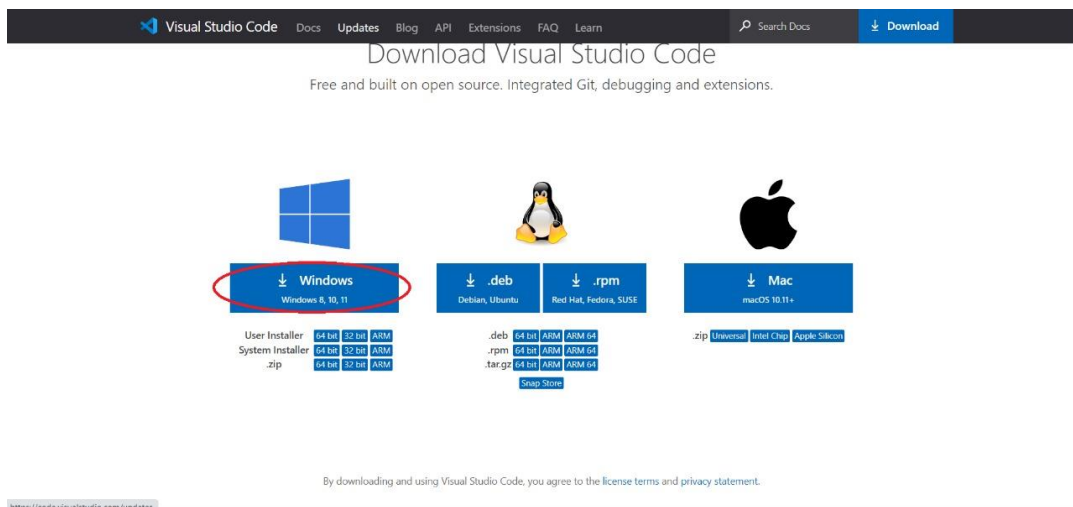
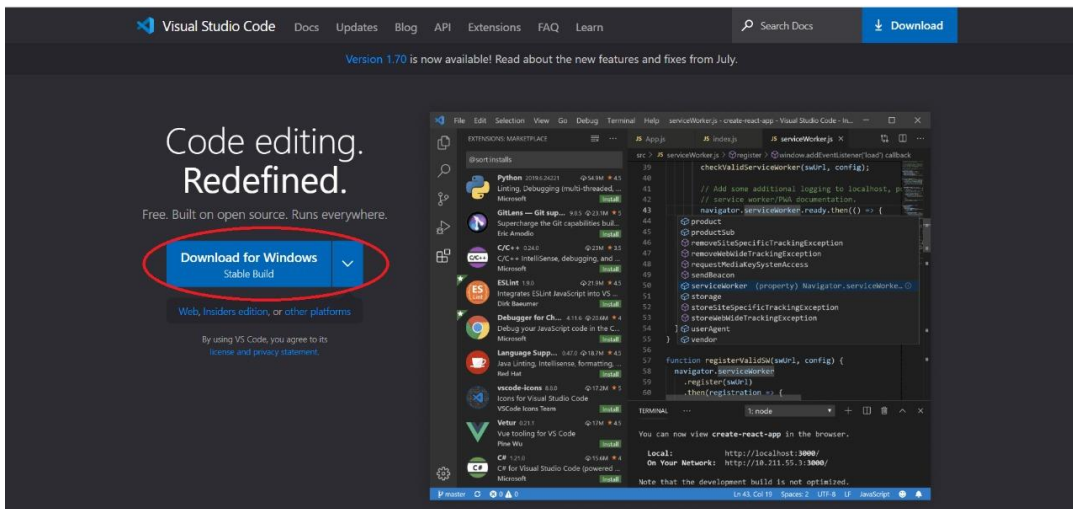
PMM PLC Systems are built to be Arduino compatible programming environment, where our PLC Range is not just compatible with Arduino IDE, but with lots of other Arduino-compatible programming software such as Visual Studio Code + Visual Micro.

PlatformIO is an ultimate customizable white-label DSP that enables transparent, resource-efficient, and profitable advertising operations for digital ad agencies, networks, and brands. Manage your campaigns effectively in real time, eliminate disruptions, and minimize time-to-value with low startup costs and full customizability.

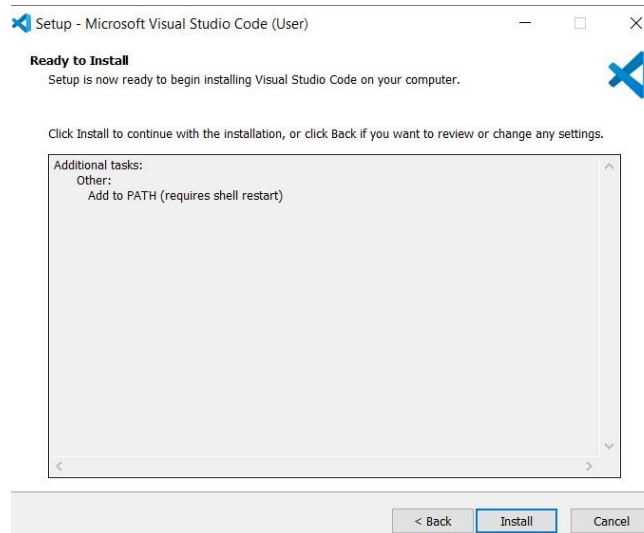
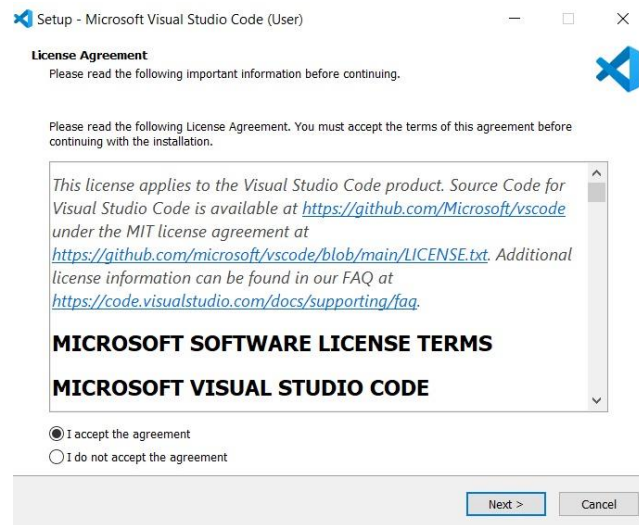
7.2 INTEGRATION GUIDELINES

7.2.1 PlatformIO Installation Guidelines

1. [Click Here](#) to get to the installation page.
2. Once the main page is opened, click on “Download for windows”.



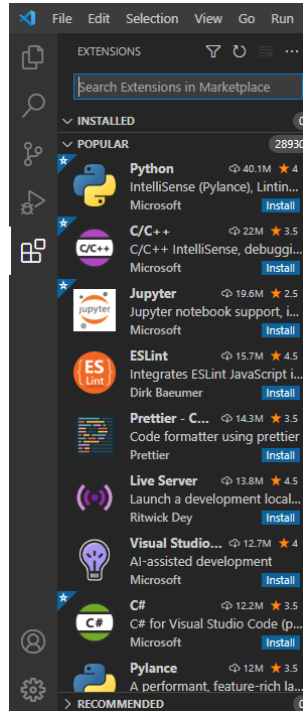
- Continue the setup process. Click on “Next” and when the application is ready for installation click on “Install”.



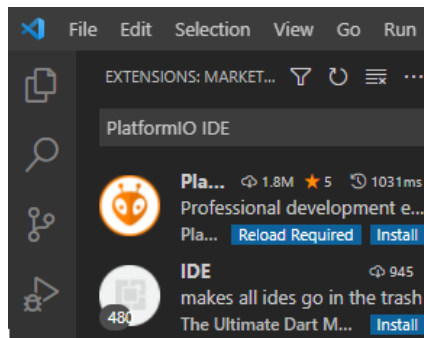
- Enable the option “Launch Visual Studio Code” then click on “Finish”.



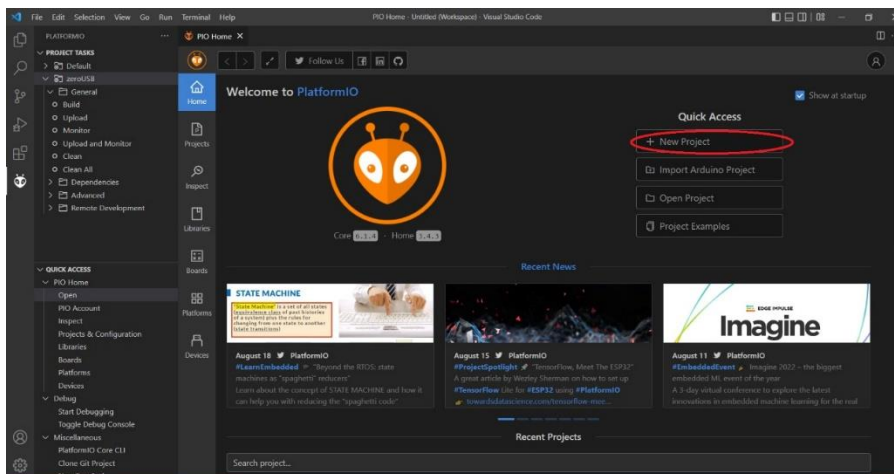
- Click on “Extensions” as shown below.



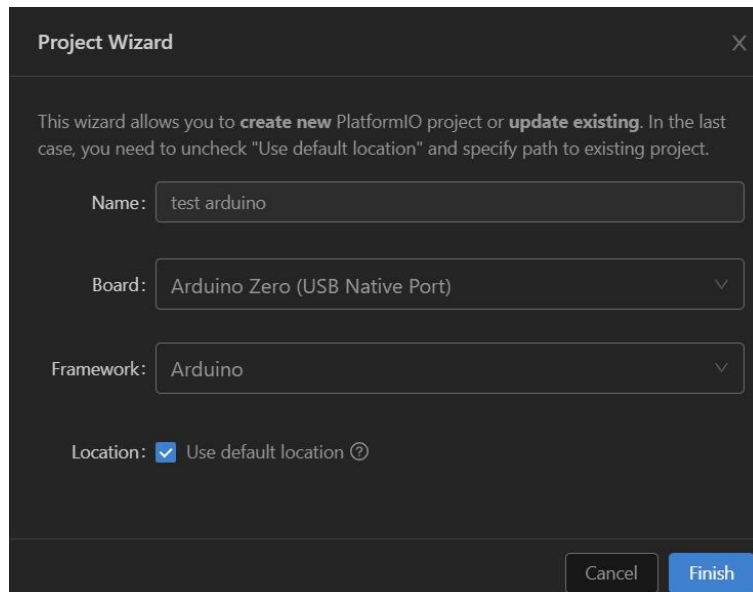
- Type in the search space “PlatformIO IDE”
- Click on “Install” as shown below.



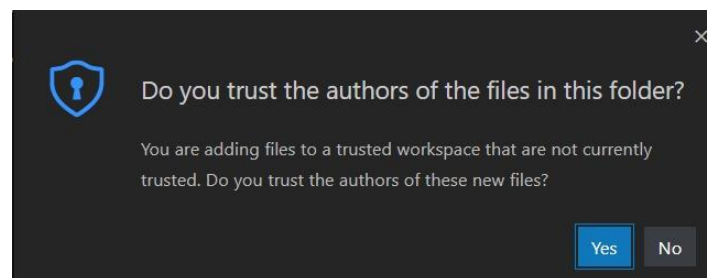
- To start writing the code, click on “new project” to open a new one.



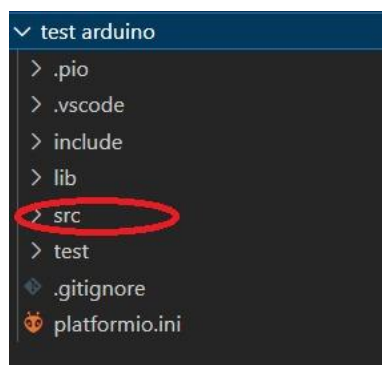
9. The project wizard window will be displayed.
10. Type the project name in the “Name” field.
11. Choose “Arduino Zero (USB Native Board)”
12. Framework is “Arduino”
13. Click on “Finish”



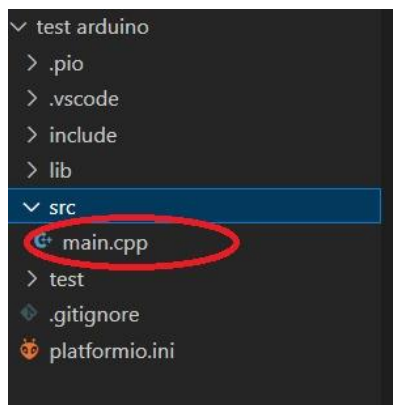
14. A security confirmation message will pop up, click on “Yes”.



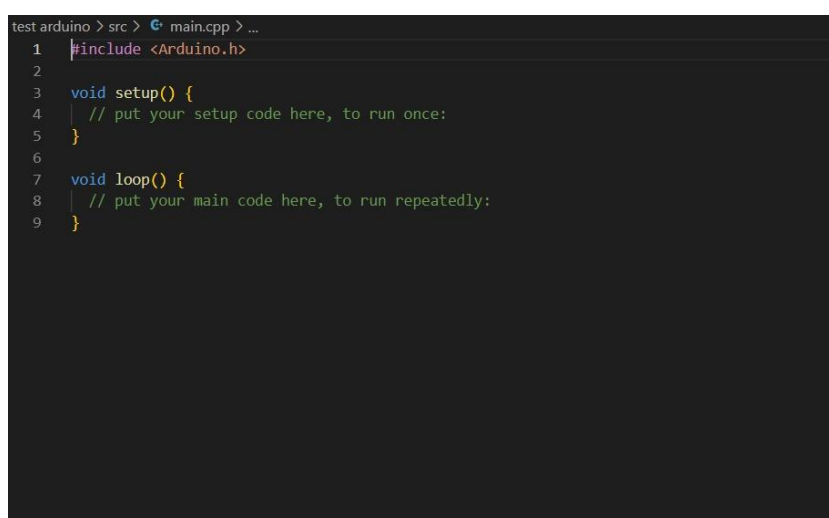
15. After completing the previous instructions, your project is created now as shown below.



16. To start writing your code click on “src” then click on “main.cpp” as shown below.



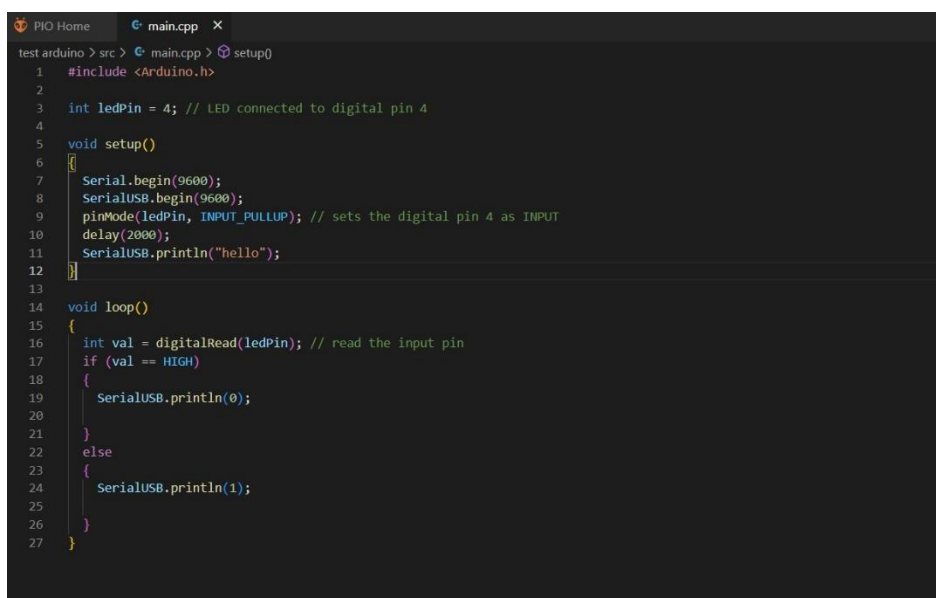
17. The coding page will be opened as shown below.



7.2.2 Integration Guidelines

Connect the USB to the device through the USB port in the front panel (Micro-USB type), and connect the other side with personal computer (PC). Once the USB is connected correctly between the device and PC, the user can start the integration as explained in the Integration Guidelines:

1. Write the desired code in the specified space.



- Click on “✓ icon; PlatformIO: build” to auto check the code to ensure nothing is wrong in the syntax.



- After checking the code, if the code is correct then “SUCCESS” is displayed.

```
#include <Arduino.h>
2
3 int ledPin = 4; // LED connected to digital pin 4
4
5 void setup()
6 {
7   Serial.begin(9600);
8   SerialUSB.begin(9600);
9   pinMode(ledPin, INPUT_PULLUP); // sets the digital pin 4 as INPUT
10  delay(2000);
11  SerialUSB.println("hello");
12 }
13
14 void loop()
15 {
16   int val = digitalRead(ledPin); // read the input pin
17   if (val == HIGH)
18   {
19     SerialUSB.println(0);
20   }
21   else
22   {
23     SerialUSB.println(1);
24   }
25 }
```

```
Compiling .pio/build/zerotest8/PlatformIO/FrameworkArduino/wiring_private.c.o
Compiling .pio/build/zerotest8/PlatformIO/FrameworkArduino/wiring_shift.c.o
Archiving .pio/build/zerotest8/lib/FrameworkArduino.a
Linking .pio/build/zerotest8/firmware.elf
Checking size .pio/build/zerotest8/firmware.elf
Advanced Memory Usage is available via "PlatformIO Home > Project Inspect"
RAM: [ ] 9.1% (used 2296 bytes from 22768 bytes)
Flash: [ ] 4.6% (used 12016 bytes from 262144 bytes)
Building .pio/build/zerotest8/firmware.bin
===== [SUCCESS] Took 7.89 seconds =====
Terminal will be reused by tasks, press any key to close it.
```

- Click on “→ icon ” that Connects the board to the USB ports to enable the user to upload the code.



- The uploading process will take a minute, the uploading is completed successfully when “SUCCESS” is displayed.


```
test arduino > src > @ main.cpp @ setup()
1 #include <Arduino.h>
2
3 int ledPin = 4; // LED connected to digital pin 4
4
5 void setup()
6 {
7   Serial.begin(9600);
8   SerialUSB.begin(9600);
9   pinMode(ledPin, INPUT_PULLUP); // sets the digital pin 4 as INPUT
10  delay(2000);
11  SerialUSB.println("hello");
12 }
13
14 void loop()
15 {
16   int val = digitalRead(ledPin); // read the input pin
17   if (val == HIGH)
18   {
19     SerialUSB.println(0);
20   }
21   else
22   {
23     SerialUSB.println(1);
24   }
25 }
```

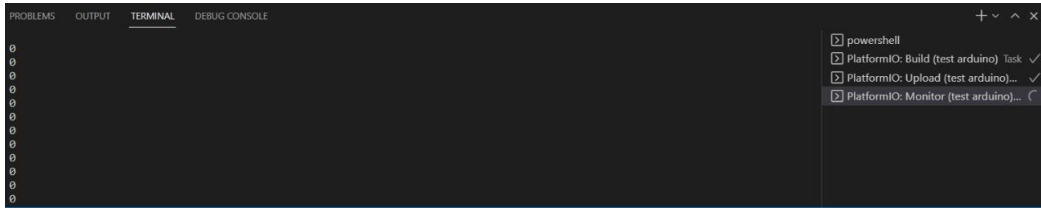
```
[=====] 34% (64/188 pages)
[=====] 68% (128/188 pages)
[=====] 100% (188/188 pages)
done in 0.093 seconds

Verify 12016 bytes of flash with checksum.
Verify successful
done in 0.009 seconds
CPU reset.
===== [SUCCESS] Took 4.73 seconds =====
Terminal will be reused by tasks, press any key to close it.
```

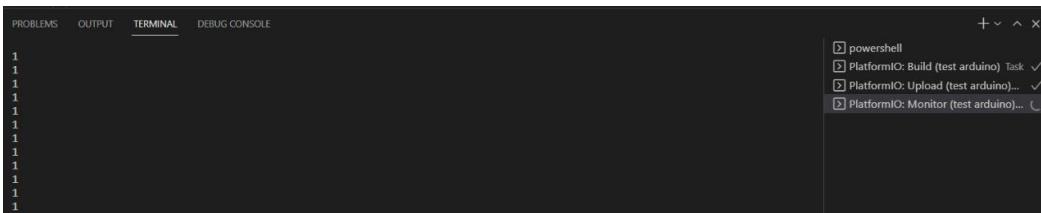
Note: the previous code example aims to define Pin No.4 as an input and monitor its status between ON/OFF when a power is connected to it, as shown in the figure below

6. In order to verify that the code is uploaded on the device and working properly:

- Click on “ serial monitor”.
- When the input status is **OFF** “0” will be displayed.



- When the input status is changed to **ON** the reading will change to “1” automatically.



Appendix 8:

PMM06 Integration with Visual Studio

8.1 Description

PMM PLC Systems are built to be Arduino compatible programming environment, where PMM’s PLCs Range is not just compatible with Arduino IDE, but with lots of other Arduino-compatible programming software such as visual studio.

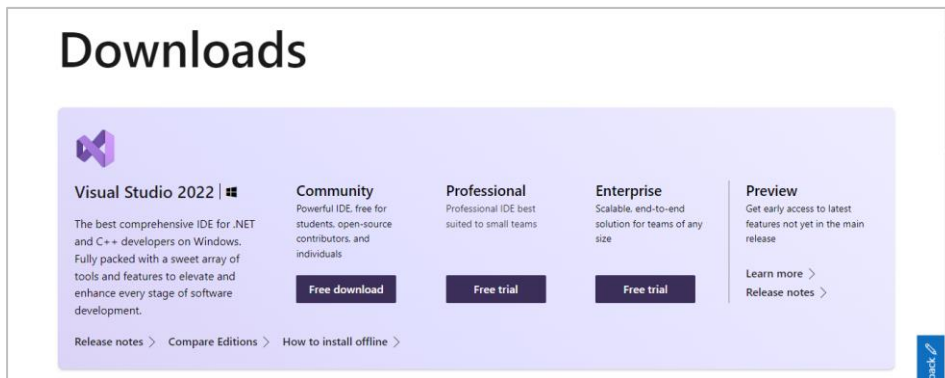
Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps.

Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

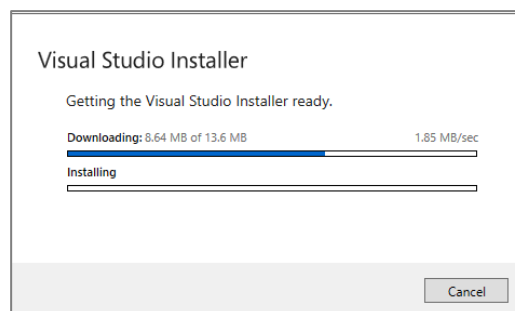
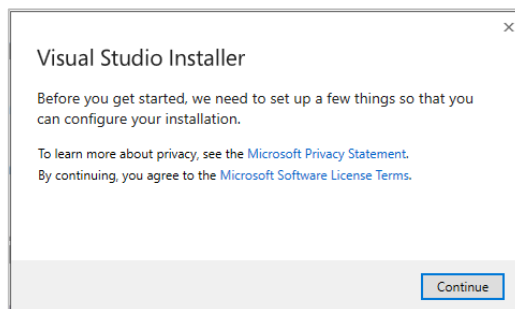
8.2 INTEGRATION GUIDELINES

8.2.1 Visual Studio Installation Guidelines

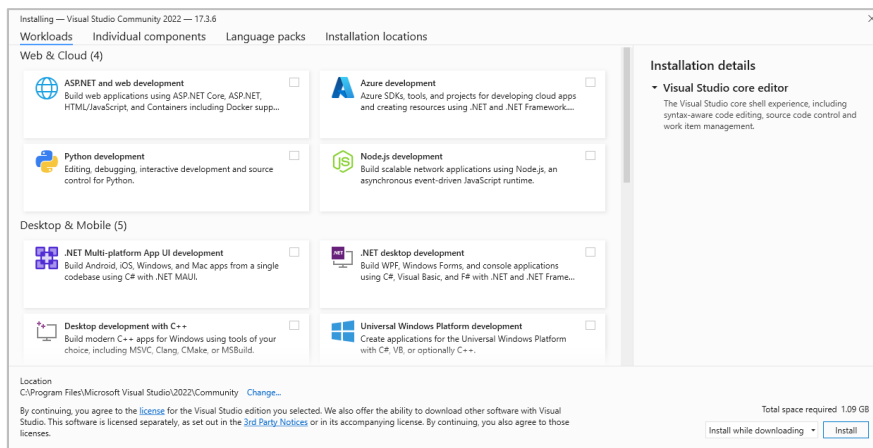
1. [Click Here](#) to get to the installation page.
2. Once the main page is opened, click on “professional”.



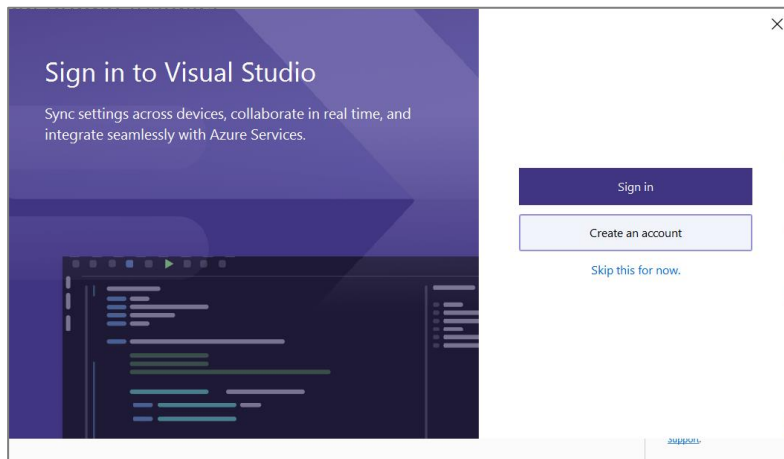
3. Click on “continue” to install the visual studio installer.



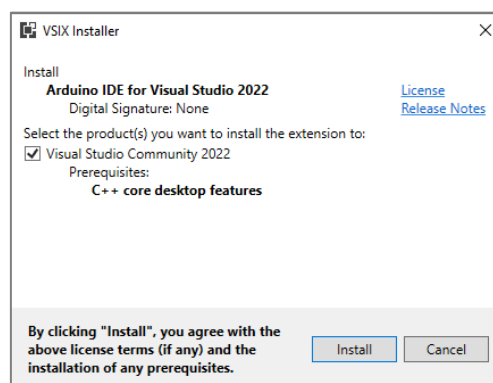
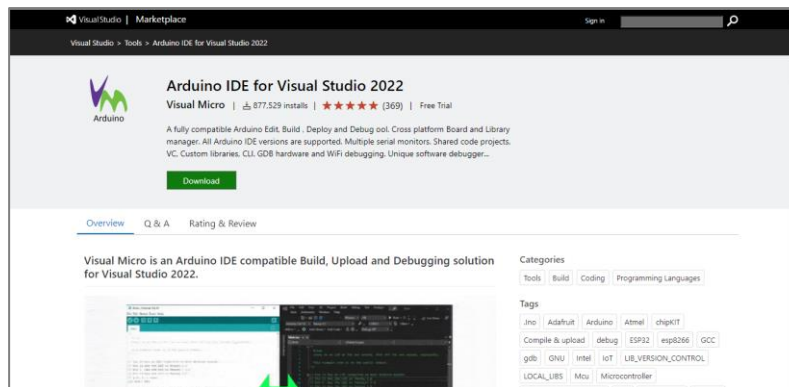
4. Visual studio installation process will start, click on “install” to continue after choosing the workloads you need.



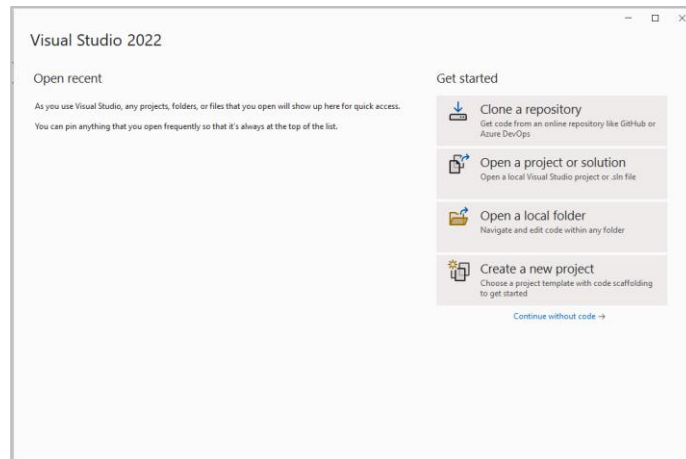
5. Sign in to your account if you have one or create a new account. You can choose to skip this step.



6. In order to work with Arduino you have to [Click here](#) to download Arduino IDE for visual studio.



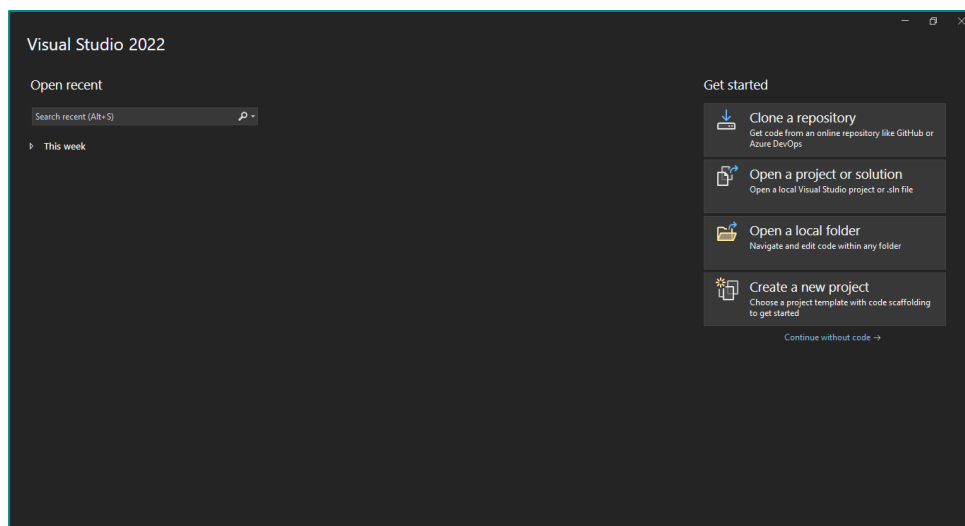
7. Now everything is setup and the user can start with the device.



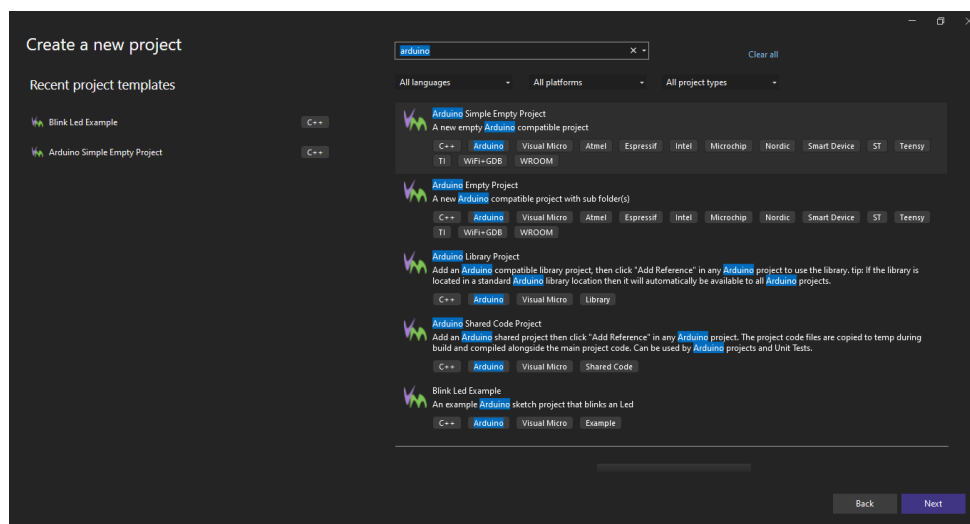
8.2.2 Integration Guidelines

Connect the USB to the device through the USB port in the front panel (Micro-USB type), and connect the other side with personal computer (PC). Once the USB is connected correctly between the device and PC, the user can start the integration as explained in the Integration Guidelines:

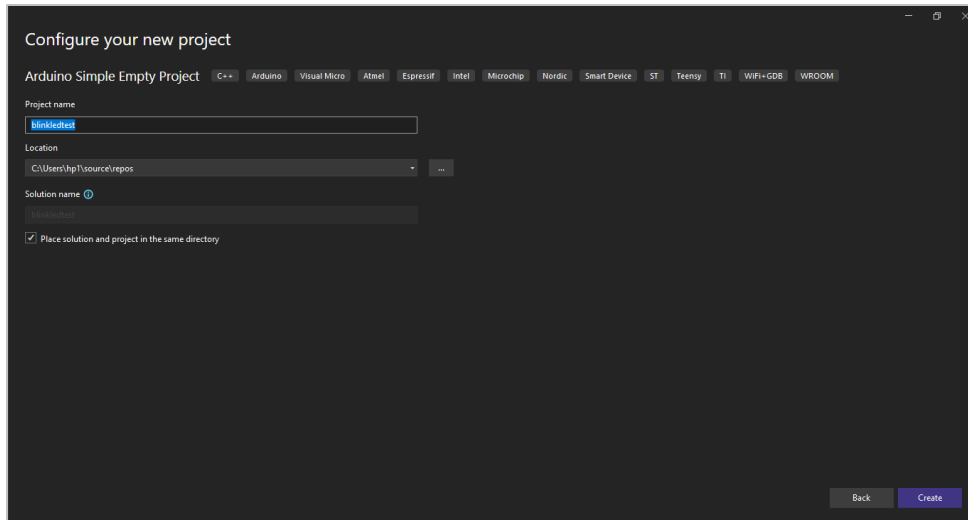
1. To start working click on “create a new project”.



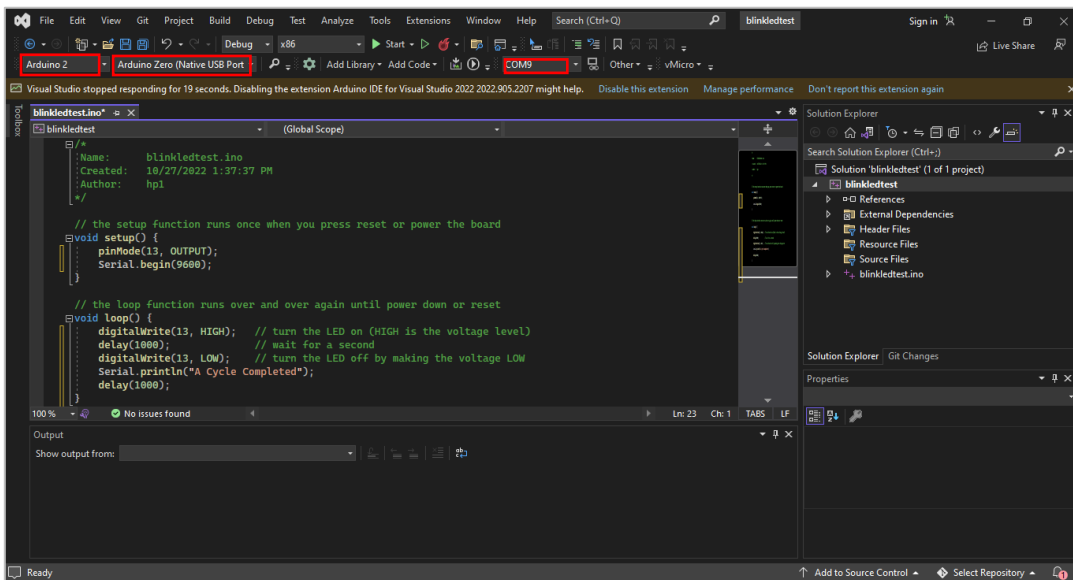
2. Choose the Arduino simple empty project.



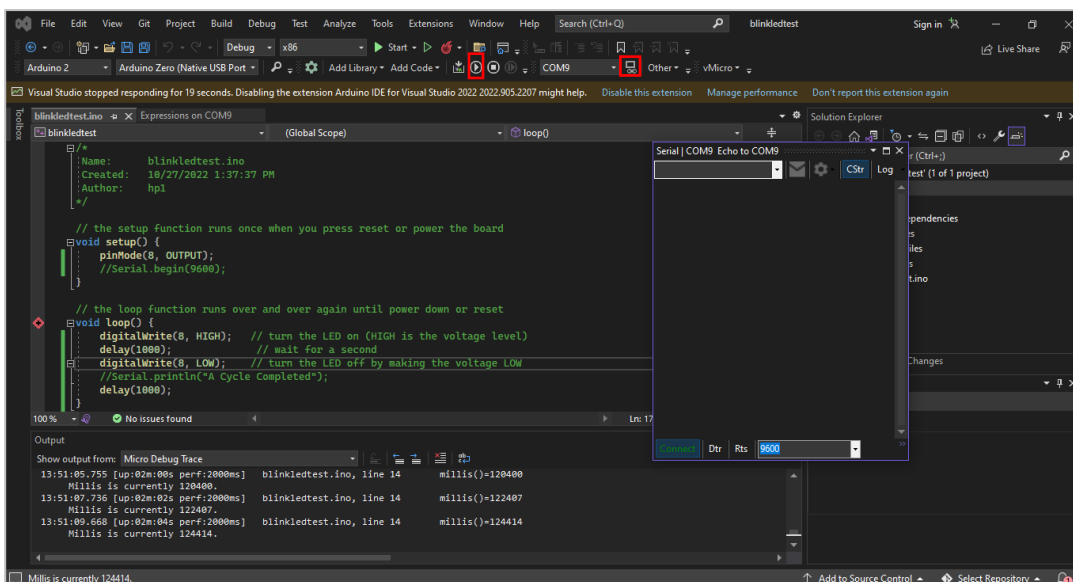
3. Name your project and select desired location.



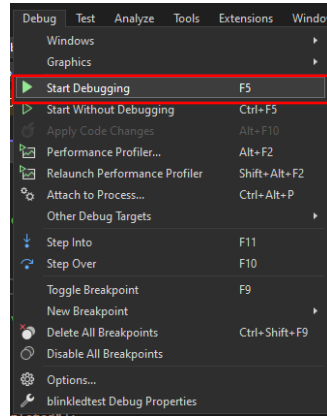
4. Choose the Arduino zero board, the Arduino IDE configure location and the port.



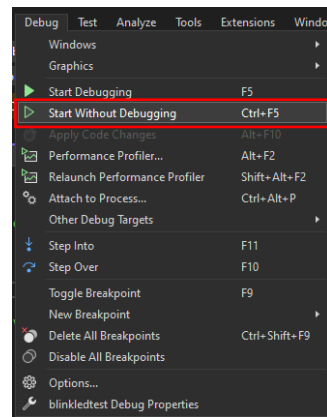
5. After writing your program, to build press on the play button and open the serial monitor by pressing on the small monitor next to the port.



6. To debug and start go to the toolbar>debug> start debugging.



7. If you wish to start without debugging go to the toolbar >debug> start without debugging



8. Now your program should be running.

Appendix 9:

PMM06 Integration with MICROCHIP Studio

9.1 Description

PMM PLC Systems are built to be Arduino compatible programming environment, where PMM's PLCs Range is not just compatible with Arduino IDE, but with lots of other Arduino-compatible programming software such as visual studio.

MICROCHIP Studio is an Integrated Development Environment (IDE) for writing and debugging AVR® /ARM® applications in Windows® XP/Windows Vista® / Windows 7/8 environments.

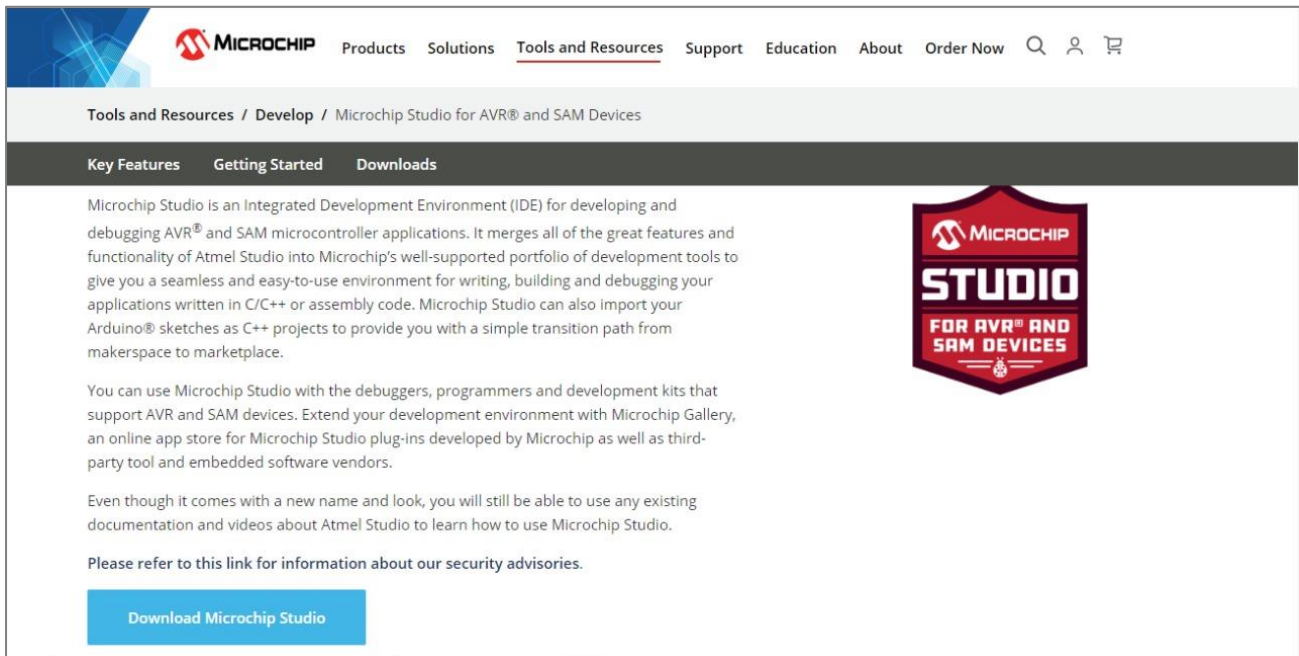
MICROCHIP Studio provides a project management tool, source file editor, simulator, assembler, and front-end for C/C++, programming, and on-chip debugging. MICROCHIP Studio supports the complete range of Atmel AVR tools. Each new release contains the latest updates for the tools as well as support for new AVR/ARM devices.

MICROCHIP Studio has a modular architecture, which allows interaction with 3rd party software vendors. GUI plugins and other modules can be written and hooked to the system. Contact Atmel for more information.

9.2 INTEGRATION GUIDELINES

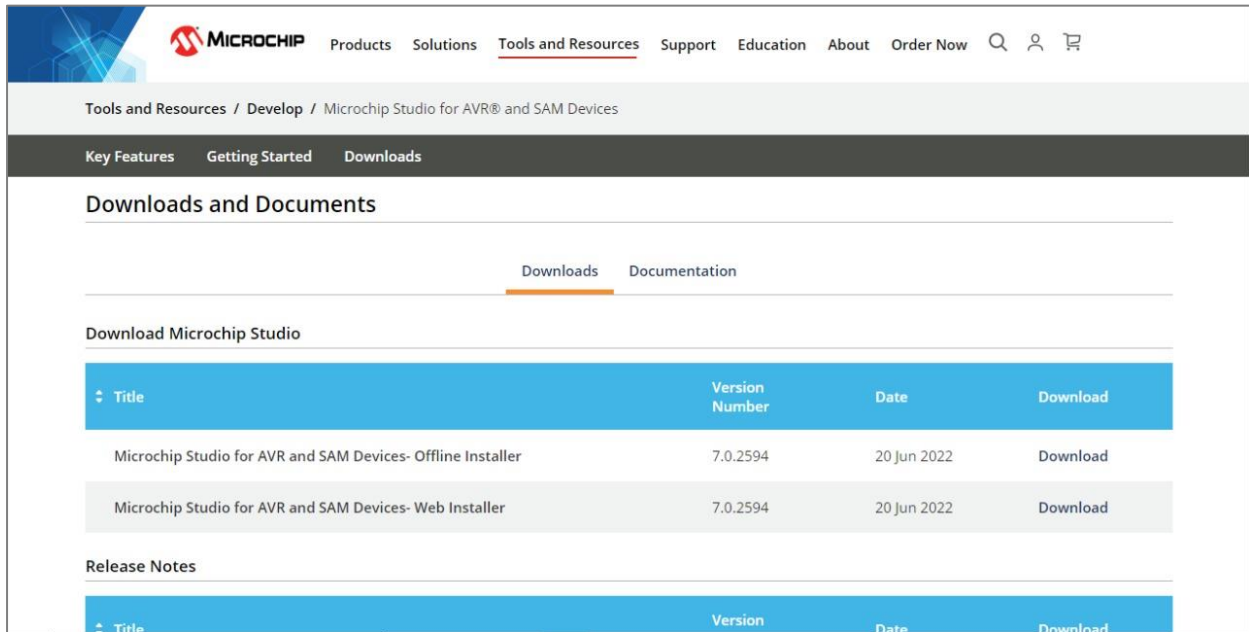
9.2.1 MICROCHIP Studio Installation Guidelines

1. [Click Here](#) to get to the installation page.
2. Once the main page is opened, click on "Download Microchip Studio".

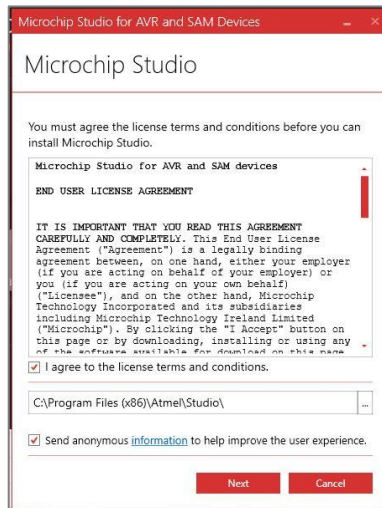


The screenshot shows the Microchip Studio website page. The navigation bar includes the Microchip logo and links for Products, Solutions, Tools and Resources (underlined), Support, Education, About, and Order Now. There are also search, user, and shopping cart icons. The breadcrumb trail reads: Tools and Resources / Develop / Microchip Studio for AVR® and SAM Devices. Below the breadcrumb, there are tabs for Key Features, Getting Started, and Downloads. The main content area contains text describing Microchip Studio as an IDE for AVR and SAM microcontroller applications, highlighting its features and ease of use. A large red shield-shaped logo for 'MICROCHIP STUDIO FOR AVR® AND SAM DEVICES' is positioned on the right. At the bottom, there is a blue button labeled 'Download Microchip Studio'.

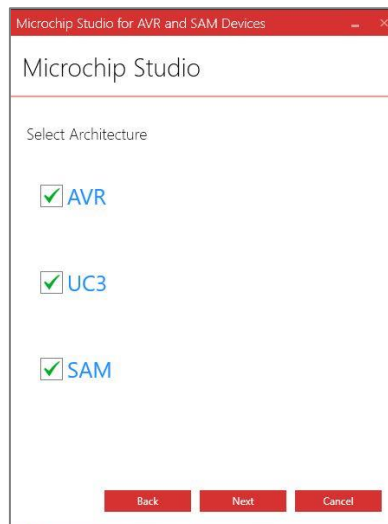
- Click on “Download” (offline Installer) to install the Microchip studio installer.



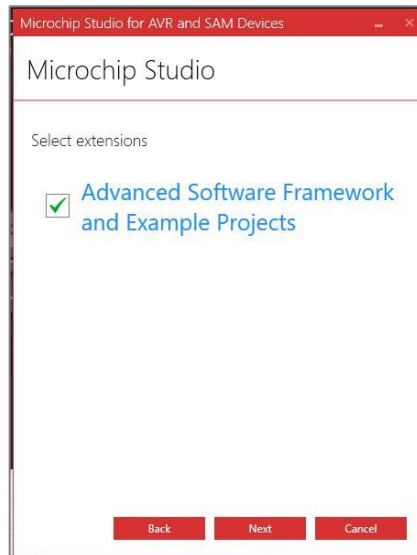
- Microchip installation process will start, click on “I agree” then “Next” to continue.



- Click on “Next” to continue.



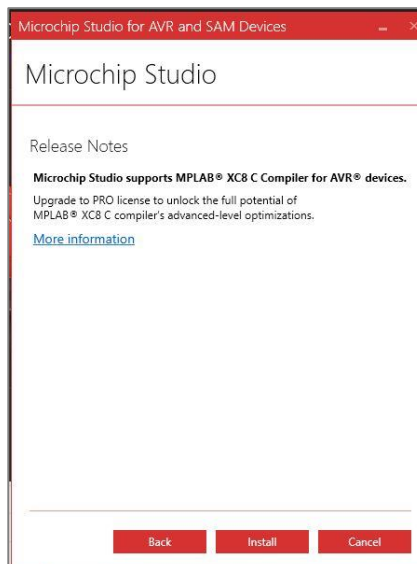
6. Click on “Next” to continue.



7. Click on “Next” to continue.



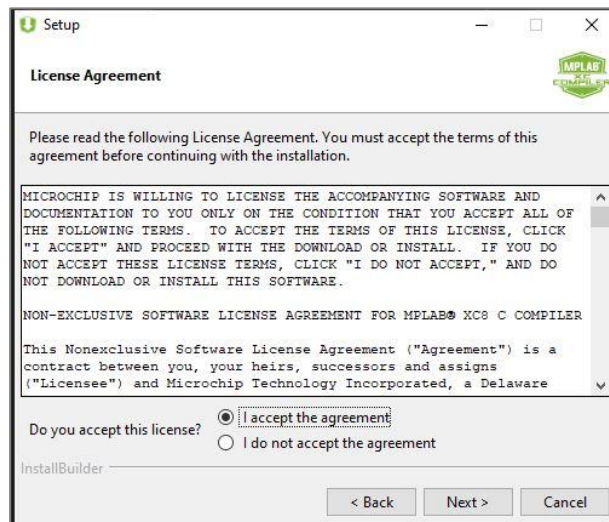
8. Press on “Install” to start the installation.



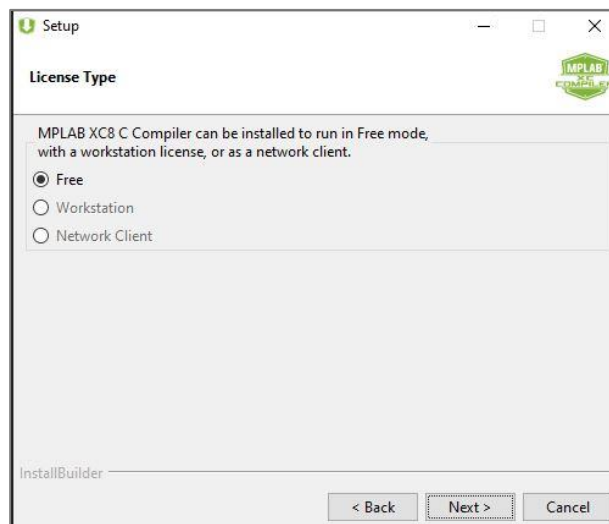
9. The MPLAB compiler will pop up, press on “Next” to proceed.



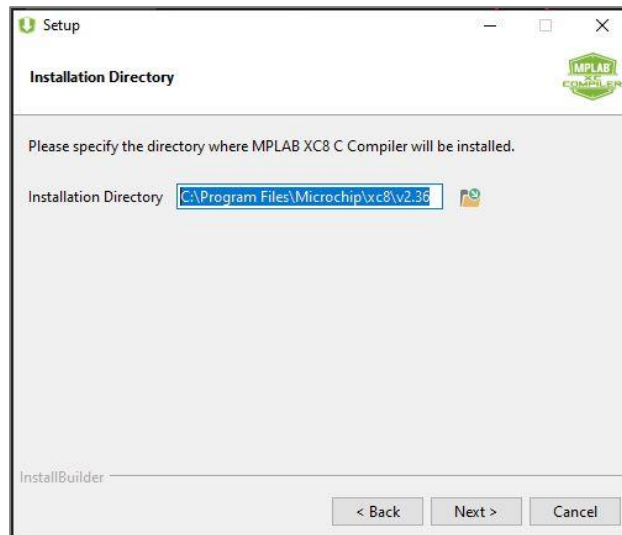
10. Choose “I accept the agreement” then click on “Next”.



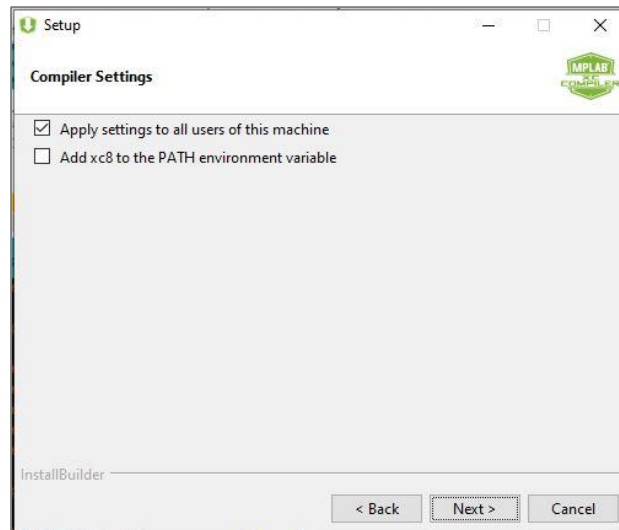
11. Choose your license type and click on “Next”.



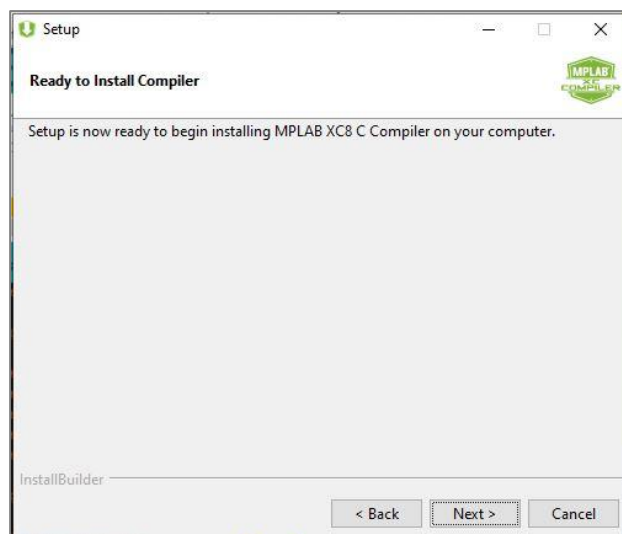
12. Choose where you want your compiler to be installed, then click on “Next”.



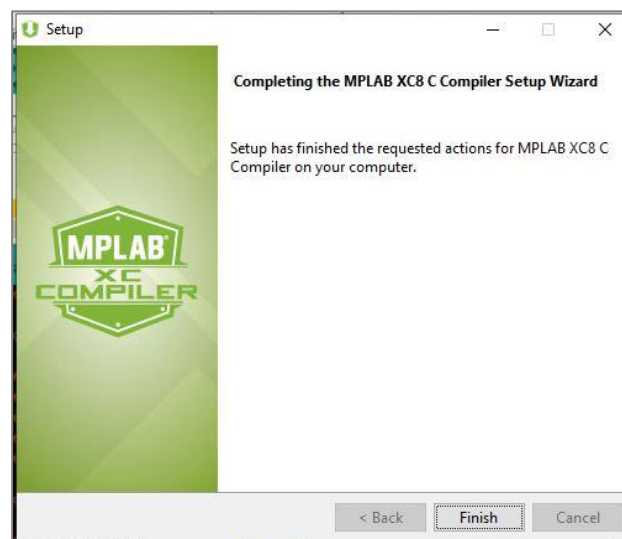
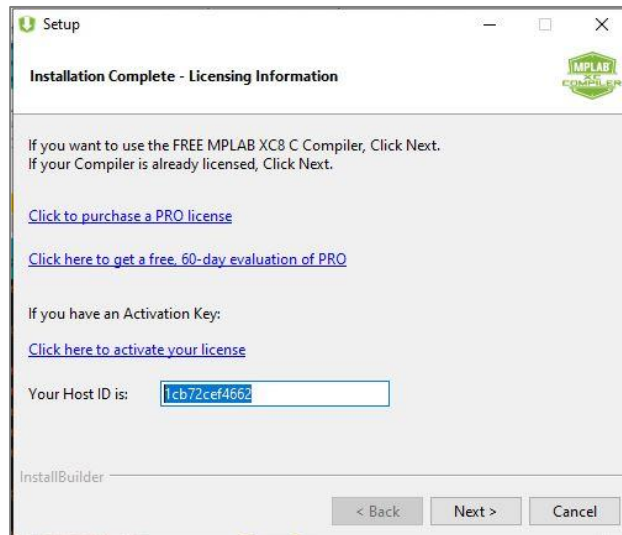
13. Choose the compiler's settings, then click on "Next".



14. Now it is ready to be installed click on "Next" to proceed.



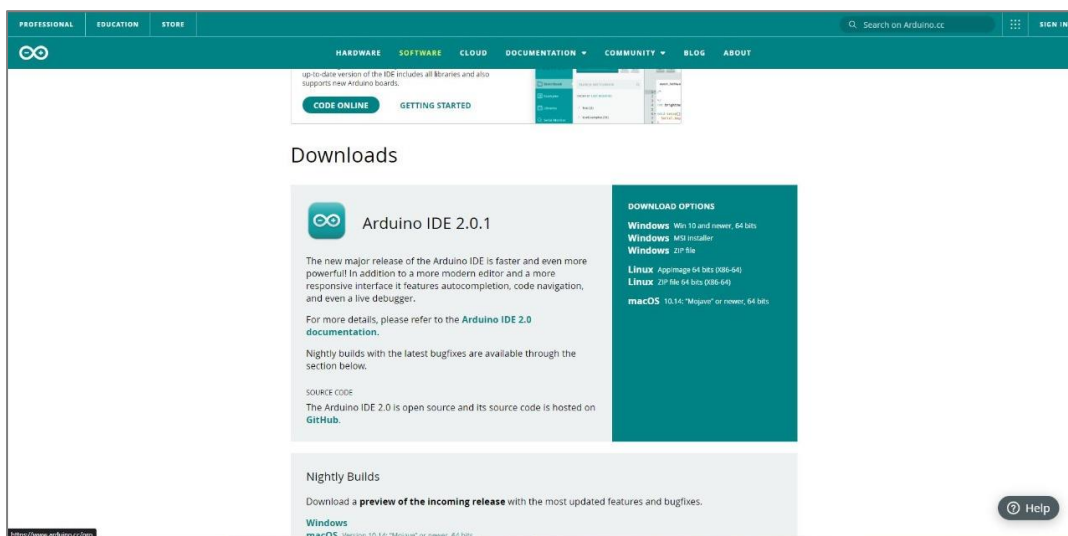
15. Click on "Next" and set up should be complete.



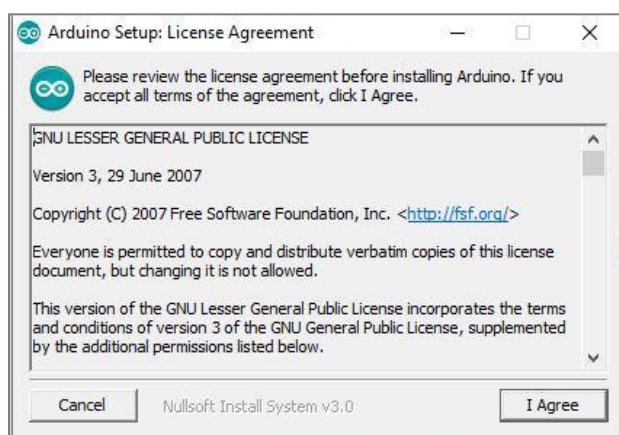
16. Now everything is set up and microchip studio installation is completed.



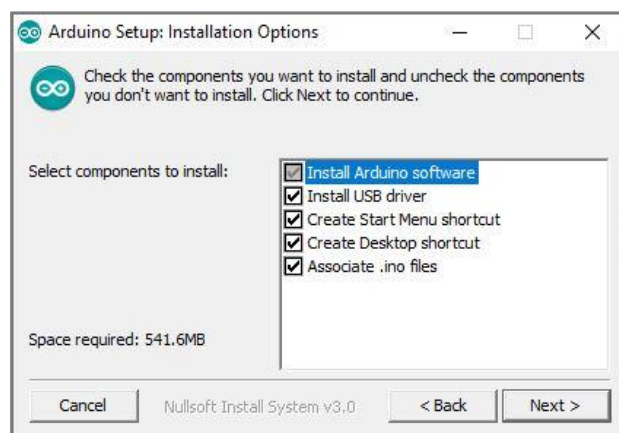
17. You will also need to install Arduino IDE. [click here](#) to go to download link, and select the suitable download option.



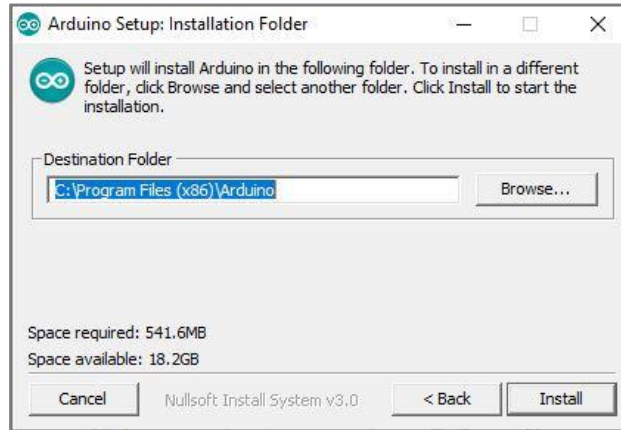
18. Press on “I Agree” to setup.



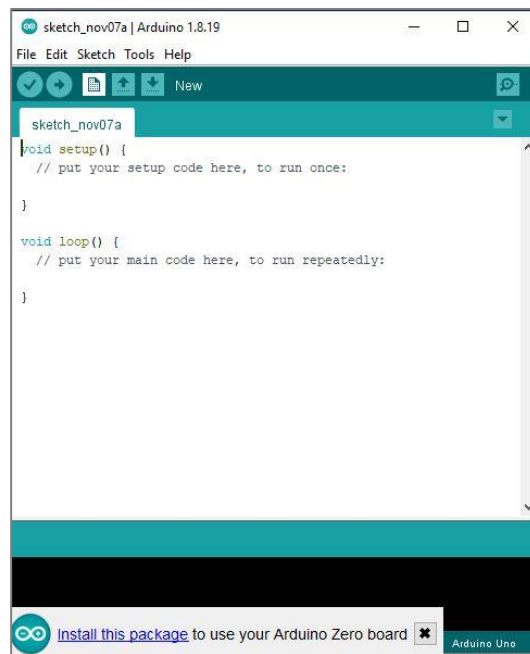
19. Choose the installation option and click on “Next”.



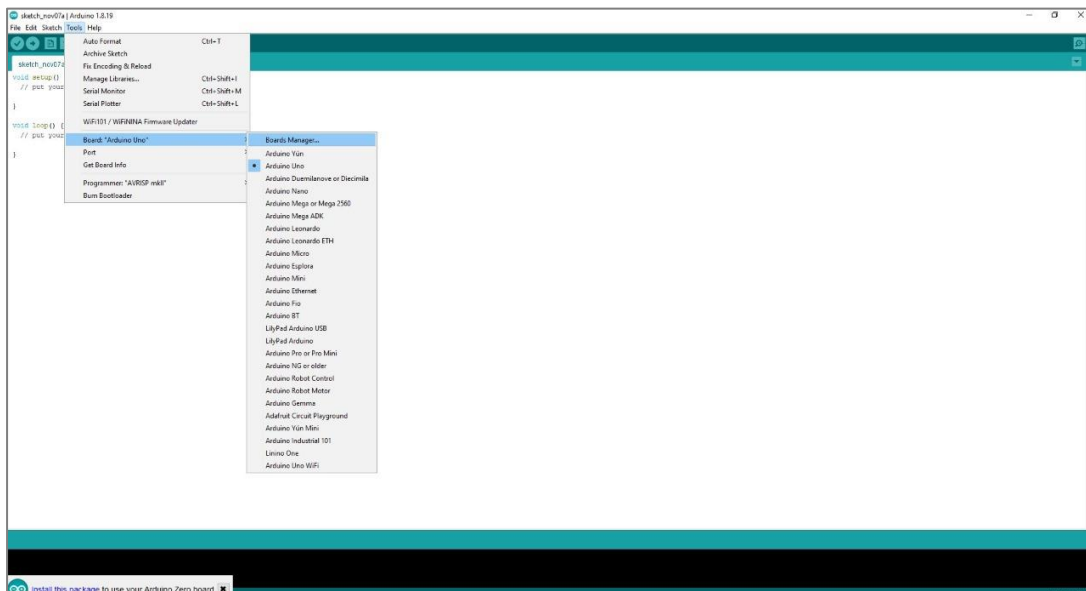
20. Choose installation folder and click on “Next”.



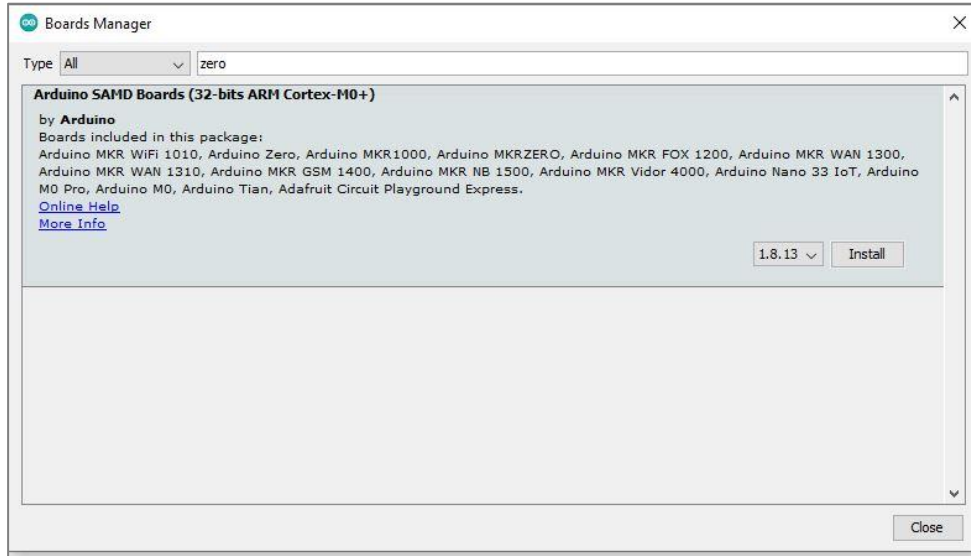
21. Now that the app is installed the following page will open.



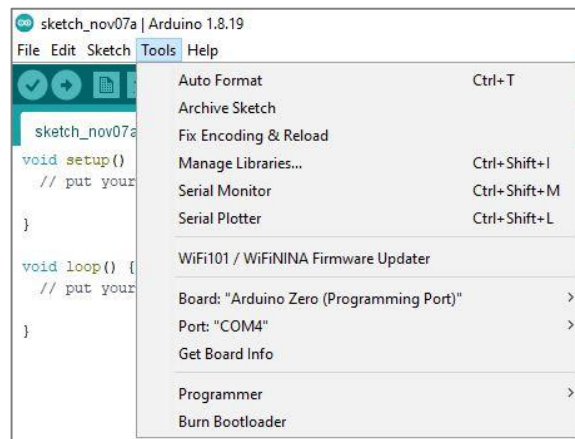
22. To install the boards, go to tools>board> board manager.



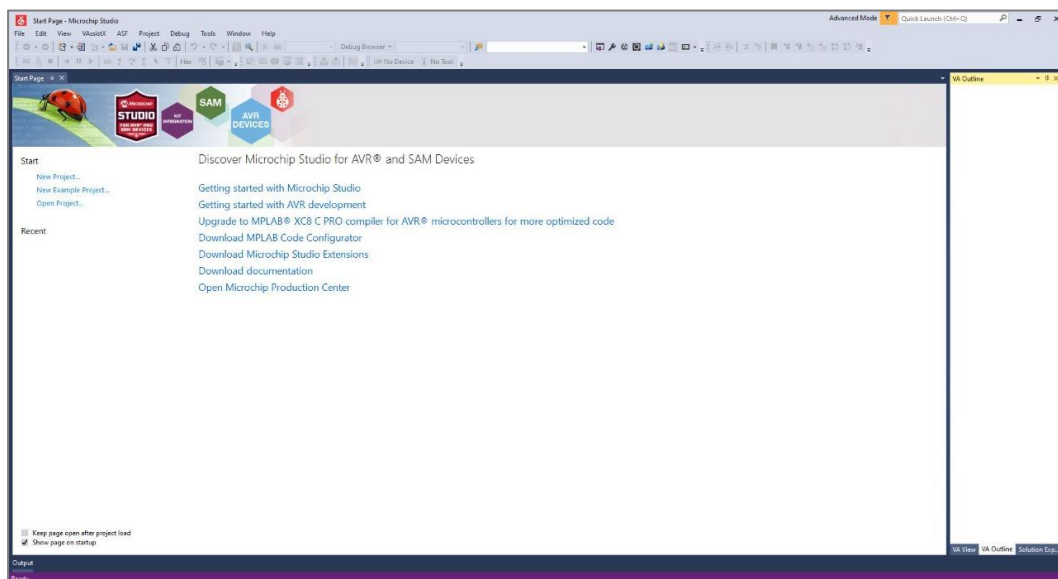
23. The board manager will open, search for the Arduino SAMD boards and install it.



24. You will be able to choose the board type and the port from the menu >tools.



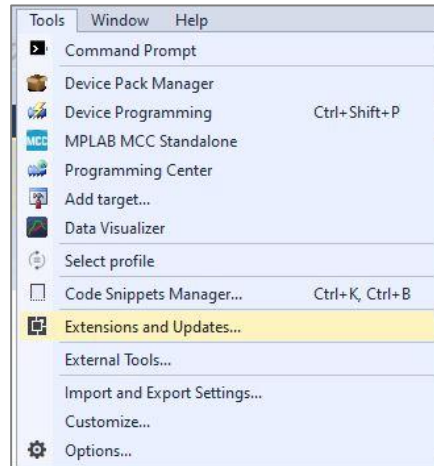
25. Now that all is set up, the start page will open and you can proceed with your project.



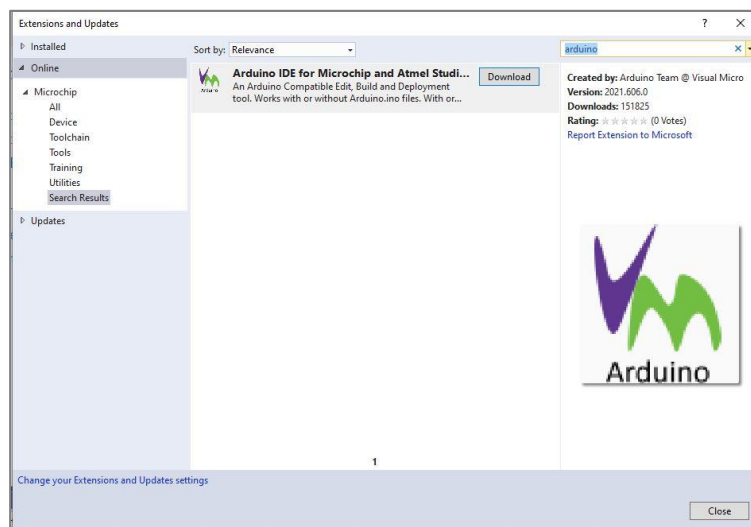
9.2.2 Integration Guidelines

Connect the USB to the device through the USB port in the front panel (Micro-USB type), and connect the other side with personal computer (PC). Once the USB is connected correctly between the device and PC, the user can start the integration as explained in the Integration Guidelines:

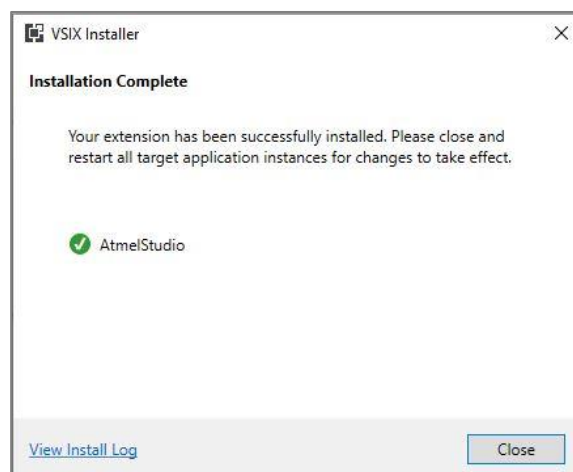
1. First you will have to install the Arduino IDE for Microchip and Atmel Studio from the menu>tools>extensions and updates.



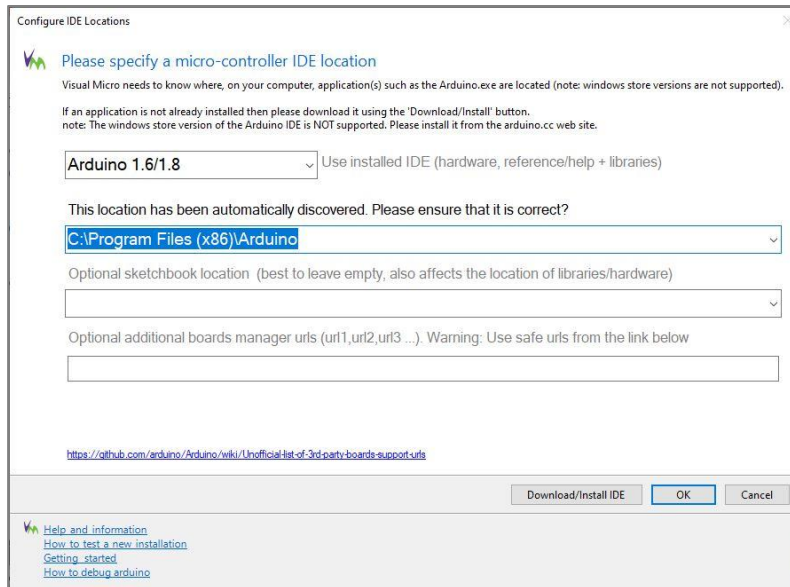
2. Download the Arduino IDE for Microchip and Atmel Studio.



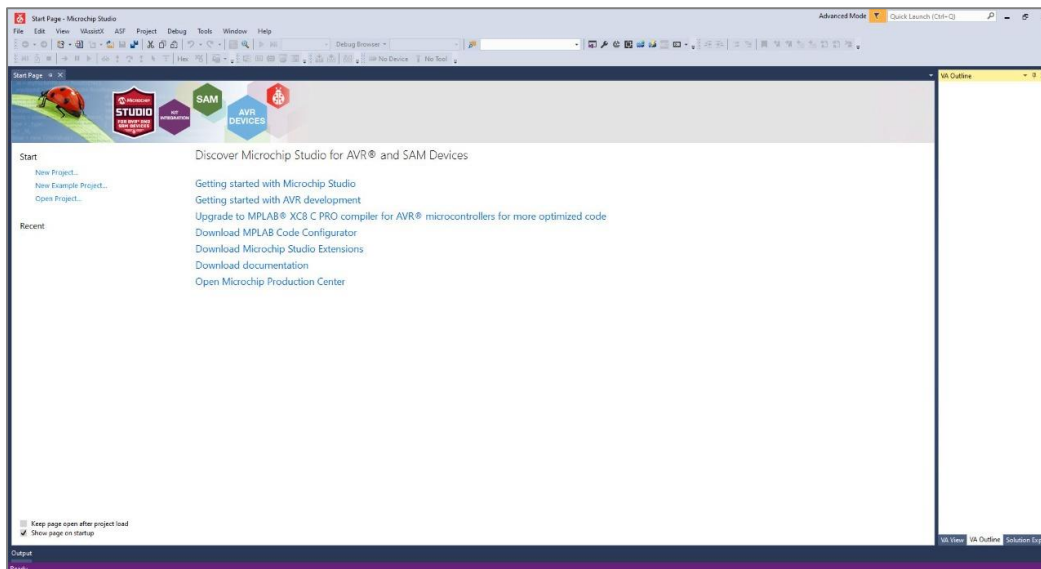
3. Name your project and select desired location.



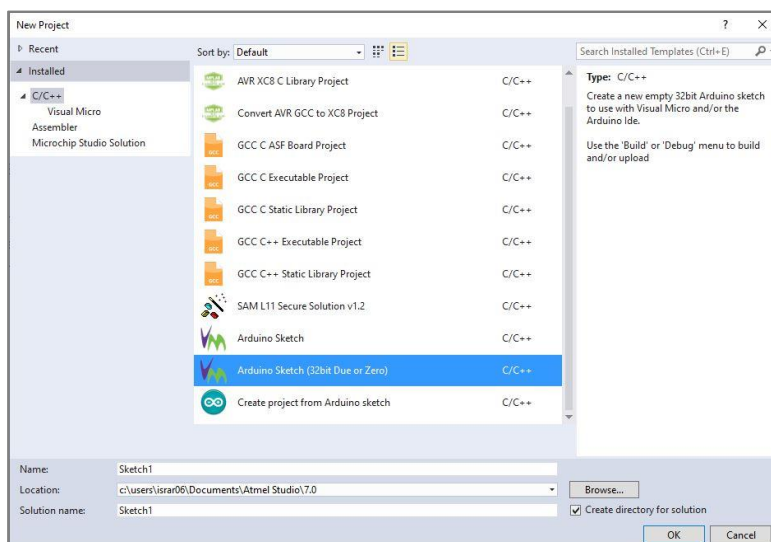
4. Specify a micro controller IDE location.



5. To create a new project, start>new project.



6. Choose the Arduino sketch zero



MECHANICAL CHARACTERISTICS

