Neousys Technology Inc.

Nuvo-2500 Series

Compact Intel® Bay Trail Fanless Computer

User's Manual
Rev. A1.0

Published Jul. 20th, 2015

Date	Description	Version
2015.7.15	First released	A1.0
2015.7.20	Modify fonts of contents	A1.0

Content

Figure

Table

Declaimer

This manual is intended to be used as a practical and informative guide only and is subject to change without prior notice. It does not represent commitment from Neousys Technology Inc. Neousys shall not be liable for direct, indirect, special, incidental, or consequential damages arising out of the use of the product or documentation, nor for any infringements upon the rights of third parties, which may result from such use.

Declaration of Conformity

FCC

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

CE

The product(s) described in this manual complies with all applicable European Union (CE) directives if it has a CE marking. For computer systems to remain CE compliant, only CE-compliant parts may be used. Maintaining CE compliance also requires proper cable and cabling techniques.

Copyright and Trademarks

Company/product names mentioned herein are used for identification purposes only and are trademarks and/or registered trademarks of their respective companies.

Chapter 1 Introduction

1.1 Overview

Nuvo-2500 Series is a series of industrial fanless computers equipped with Intel® Bay Trail Processors for different applications. Powered by the quad-core Bay Trail processor, Nuvo-2500 Series shows outstanding computing power and is even more power efficient compared to those with its predecessors. Nuvo-2500 Series supports dual Independent display, dual storage for isolating system and data, 2x Gigabit Ethernet ports, 4x COM ports and 4x USB ports.

Nuvo-2500 Series provides many models for different applications. Nuvo-2500 is for light-weight general embedded application which requires one PCI or PCIe slot for a 3rd party add-on card. Nuvo-2510VTC provides no PCI and PCIe slot and is quite compact. This makes Nuvo-2510VTC suitable for general embedded application where installation space is limited. Besides, Nuvo-2510VTC has a CAN Bus port and 2 built-in PoE+ ports and is also ideal for mobile surveillance applications.

Nuvis-2520at is one another model of Nuvo-2500 Series. Nuvis-2520at equips with 2 PoE+ ports, DTIO and 1 built-in constant current/voltage power supply, which directly drives a LED lighting device. DTIO is a set of MCU-controlled I/O which reacts autonomously according to a pre-defined rule. Strobe control can easily be realized by DTIO without any CPU effort. The combination of these unique features makes Nuvis-2520at proper for industrial automation and machine vision applications.

1.2 Product Specification

1.2.1 Specification of Nuvo-2500P

Table 1: Specification of Nuvo-2500P

System Core	
Processor	Intel® Celeron® Bay Trail J1900 Quad-core Processor (2.42GHz, 2MCache)
Graphic	Integrated Intel® HD graphics
Memory	1x 204-pin SO-DIMM Socket up to 8GB DDR3L-1333MHz SDRAM
Front-panel I/O Interface	
Ethernet	2x Gigabit Ethernet by Intel® Ethernet Controller I210
Video Port	1x VGA output, supporting resolution up to 2560 x 1600
Serial Port	2x BIOS-Configurable RS-232/422/485 (COM1 & COM2)
USB	1x USB3.0 and 3x USB2.0
Power Input	1x 3-pinpluggable terminal block for DC input
Back-panel I/O Interface	
Video Port	1x DVI-D output via DVI-I connector, supporting resolution up to 2560 x 1600
Serial Port	2x RS-232 (COM3 & COM4)
Audio	1x Speaker-out and 1x Mic-in
Aux I/O Port	Optional 4x DI, 8x DO, 6x PWM, 1x Quadrature Encoder Input and 2x AI via 1x DB37 connector
Storage Interface	
SATA 2.0	1x internal SATA port for 2.5" HDD/SSD installation
mSATA	1x internal half-size mPCIe socket with USB signal
Expansion Bus	
Mini PCI Express	1x full-size mini PCI Express socket with USIM holder (PCIe x1 Gen2 and USB2 signal)
	1x full-size mini PCI Express socket (USB signal)
PCI	1x PCI Slot with 33MHz/33-bit PCI, supporting max. card size up to 99.4mm x 167.6mm (with optional fan) or 99.4mm x 179.6mm (without optional fan)
Power Supply	
DC Input	8~35V DC
Mechanical	
Dimension	205 mm (W) x 145 mm (D) x 73 mm (H)
Weight	2.3 kg (including one 2.5" HDD and DDR3 SO-DIMM)
Mounting	Wall-mount (Standard) or DIN-rail mount (Optional)
Environmental	
Operating	-25°C ~ 70°C, 100% CPU loading */**

Temperature	
Storage Temperature	-40°C ~85°C
Humidity	10%~90% , non-condensing
Vibration	Operating, 5 Grms, 5-500 Hz, 3 Axes (w/ SSD, w/o add-on card, according to IEC60068-2-64)
Shock	Operating, 50 Grms, Half-sine 11 ms Duration (w/ SSD, w/o add-on card, according to IEC60068-2-27)
EMC	CE/FCC Class A, according to EN 55022 & EN 55024

1.2.2 Specification of Nuvo-2500E

Table 2: Specification of Nuvo-2500E

System Core	
Processor	Intel® Celeron® Bay Trail J1900 Quad-core Processor (2.42GHz, 2MCache)
Graphic	Integrated Intel® HD graphics
Memory	1x 204-pin SO-DIMM Socket up to 8GB DDR3L-1333MHz SDRAM
Front-panel I/O Interface	
Ethernet	2x Gigabit Ethernet by Intel® Ethernet Controller I210
Video Port	1x VGA output, supporting resolution up to 2560 x 1600
Serial Port	2x BIOS-Configurable RS-232/422/485 (COM1 & COM2)
USB	1x USB3.0 and 3x USB2.0
Power Input	1x 3-pin pluggable terminal block for DC input
Back-panel I/O Interface	
Video Port	1x DVI-D output via DVI-I connector, supporting resolution up to 2560 x 1600
Serial Port	2x RS-232 (COM3 & COM4)
Audio	1x Speaker-out and 1x Mic-in
Aux I/O Port	Optional 4x DI, 8x DO, 6x PWM, 1x Quadrature Encoder Input and 2x AI via 1x DB37 connector
Storage Interface	
SATA 2.0	1x internal SATA port for 2.5" HDD/SSD installation
mSATA	1x internal half-size mPCIe socket with USB signal
Expansion Bus	
Mini PCI Express	1x full-size mini PCI Express socket with USIM holder (PCIe x1 Gen2 and USB2 signal)
	1x full-size mini PCI Express socket (USB signal)
PCI Express	1x PCI Express x4 slot with 1-lane Gen2 PCI Express Signal, supporting max. card size up to 99.4mm x 167.6mm (with optional fan) or 99.4mm x 179.6mm (without optional fan)
Power Supply	
DC Input	8~35V DC
Mechanical	
Dimension	205 mm (W) x 145 mm (D) x 73 mm (H)
Weight	2.3 kg (including one 2.5" HDD and DDR3 SO-DIMM)
Mounting	Wall-mount (Standard) or DIN-rail mount (Optional)
Environmental	
Operating Temperature	-25°C ~ 70°C, 100% CPU loading */**
Storage	-40°C ~85°C

Temperature	
Humidity	10%~90% , non-condensing
Vibration	Operating, 5 Grms, 5-500 Hz, 3 Axes (w/ SSD, w/o add-on card, according to IEC60068-2-64)
Shock	Operating, 50 Grms, Half-sine 11 ms Duration (w/ SSD, w/o add-on card, according to IEC60068-2-27)
EMC	CE/FCC Class A, according to EN 55022 & EN 55024

1.2.3 Specification of Nuvo-2510VTC

Table 3: Specification of Nuvo-2510VTC

System Core	
Processor	Intel® Atom™ Bay Trail E3845 Quad-core Processor (1.91GHz, 2MCache)
Graphic	Integrated Intel® HD graphics
Memory	1x 204-pin SO-DIMM Socket up to 8GB DDR3L-1333MHz SDRAM
Front-panel I/O Interface	
Ethernet	2x Gigabit Ethernet by Intel® Ethernet Controller I210
PoE PSE	IEEE 802.3at (25.5W) each GbE port
Video Port	1x VGA output, supporting resolution up to 2560 x 1600
Serial Port	2x BIOS-Configurable RS-232/422/485 (COM1 & COM2)
USB	1x USB3.0 and 3x USB2.0
Power Input	1x 3-pin pluggable terminal block for DC input
Back-panel I/O Interface	
Video Port	1x DVI-D output via DVI-I connector, supporting resolution up to 2560 x 1600
Serial Port	2x RS-232 (COM3 & COM4)
Audio	1x Speaker-out and 1x Mic-in
CAN Bus	1x CAN Bus port via 1x DB-9 connector
Storage Interface	
SATA 2.0	1x internal SATA port for 2.5" HDD/SSD installation
mSATA	1x internal half-size mPCIe socket with USB signal
Expansion Bus	
Mini PCI Express	1x full-size mini PCI Express socket with USIM holder (PCIe x1 Gen2 and USB2 signal)
	1x full-size mini PCI Express socket with external push-out USIM holder (USB signal)
Power Supply	
DC Input	8~35V DC
Mechanical	
Dimension	205 mm (W) x 145 mm (D) x 44.3 mm (H)
Weight	1.9 kg (including one 2.5" HDD and DDR3 SO-DIMM)
Mounting	Patent-pending shock-absorbing wall-mounting (standard) or DIN-Rail mounting (optional)
Environmental	
Operating Temperature	-25°C ~ 70°C, 100% CPU loading */**
Storage Temperature	-40°C ~85°C
Humidity	10%~90% , non-condensing

Vibration	Operating, 5 Grms, 5-500 Hz, 3 Axes (w/ SSD, w/o add-on card, according to IEC60068-2-64)
Shock	Operating, 50 Grms, Half-sine 11 ms Duration (w/ SSD, w/o add-on card, according to IEC60068-2-27)
EMC	CE/FCC Class A, according to EN 55022 & EN 55024

1.2.4 Specification of Nuvis-2520at-P

Table 4: Specification of Nuvis-2520at-P

System Core	
Processor	Intel® Celeron® Bay Trail J1900 Quad-core Processor (2.42GHz, 2MCache)
Graphic	Integrated Intel® HD graphics
Memory	1x 204-pin SO-DIMM Socket up to 8GB DDR3L-1333MHz SDRAM
Front-panel I/O Interface	
Ethernet	2x Gigabit Ethernet by Intel® Ethernet Controller I210
PoE PSE	IEEE 802.3at (25.5W) each GbE port
Video Port	1x VGA output, supporting resolution up to 2560 x 1600
Serial Port	2x BIOS-Configurable RS-232/422/485 (COM1 & COM2)
USB	1x USB3.0 and 3x USB2.0
Power Input	1x 3-pin pluggable terminal block for DC input
Back-panel I/O Interface	
Video Port	1x DVI-D output via DVI-I connector, supporting resolution up to 2560 x 1600
Serial Port	2x RS-232 (COM3 & COM4)
Audio	1x Speaker-out and 1x Mic-in
Aux I/O Port	4x DI, 8x DO, 6x PWM, 1x Quadrature Encoder Input, 2x AI and 1x LED CC/CV output for LED driving via 1x DB37 connector
Storage Interface	
SATA 2.0	1x internal SATA port for 2.5" HDD/SSD installation
mSATA	1x internal half-size mPCIe socket with USB signal
Expansion Bus	
Mini PCI Express	1x full-size mini PCI Express socket with USIM holder (PCIe x1 Gen2 and USB2 signal)
	1x full-size mini PCI Express socket (USB signal)
PCI	1x PCI Slot with 33MHz/33-bit PCI, supporting max. card size up to 99.4mm x 167.6mm (with optional fan) or 99.4mm x 179.6mm (without optional fan)
Power Supply	
DC Input	8~35V DC
Mechanical	
Dimension	205 mm (W) x 145 mm (D) x 73 mm (H)
Weight	2.3 kg (including one 2.5" HDD and DDR3 SO-DIMM)
Mounting	Wall-mount (Standard) or DIN-rail mount (Optional)
Environmental	
Operating Temperature	-25°C ~ 70°C, 100% CPU loading */**

Storage Temperature	-40°C ~85°C
Humidity	10%~90% , non-condensing
Vibration	Operating, 5 Grms, 5-500 Hz, 3 Axes (w/ SSD, w/o add-on card, according to IEC60068-2-64)
Shock	Operating, 50 Grms, Half-sine 11 ms Duration (w/ SSD, w/o add-on card, according to IEC60068-2-27)
EMC	CE/FCC Class A, according to EN 55022 & EN 55024

1.2.5 Specification of Nuvis-2520at-E

Table 5: Specification of Nuvis-2520at-E

System Core	
Processor	Intel® Celeron® Bay Trail J1900 Quad-core Processor (2.42GHz, 2MCache)
Graphic	Integrated Intel® HD graphics
Memory	1x 204-pin SO-DIMM Socket up to 8GB DDR3L-1333MHz SDRAM
Front-panel I/O Interface	
Ethernet	2x Gigabit Ethernet by Intel® Ethernet Controller I210
PoE PSE	IEEE 802.3at (25.5W) each GbE port
Video Port	1x VGA output, supporting resolution up to 2560 x 1600
Serial Port	2x BIOS-Configurable RS-232/422/485 (COM1 & COM2)
USB	1x USB3.0 and 3x USB2.0
Power Input	1x 3-pin pluggable terminal block for DC input
Back-panel I/O Interface	
Video Port	1x DVI-D output via DVI-I connector, supporting resolution up to 2560 x 1600
Serial Port	2x RS-232 (COM3 & COM4)
Audio	1x Speaker-out and 1x Mic-in
Aux I/O Port	4x DI, 8x DO, 6x PWM, 1x Quadrature Encoder Input, 2x AI and 1x LED CC/CV output for LED driving via 1x DB37 connector
Storage Interface	
SATA 2.0	1x internal SATA port for 2.5" HDD/SSD installation
mSATA	1x internal half-size mPCIe socket with USB signal
Expansion Bus	
Mini PCI Express	1x full-size mini PCI Express socket with USIM holder (PCIe x1 Gen2 and USB2 signal)
	1x full-size mini PCI Express socket (USB signal)
PCI Express	1x PCI Express x4 slot with 1-lane Gen2 PCI Express Signal, supporting max. card size up to 99.4mm x 167.6mm (with optional fan) or 99.4mm x 179.6mm (without optional fan)
Power Supply	
DC Input	8~35V DC
Mechanical	
Dimension	205 mm (W) x 145 mm (D) x 73 mm (H)
Weight	2.3 kg (including one 2.5" HDD and DDR3 SO-DIMM)
Mounting	Wall-mount (Standard) or DIN-rail mount (Optional)
Environmental	
Operating Temperature	-25°C ~ 70°C, 100% CPU loading */**

Storage Temperature	-40°C ~85°C
Humidity	10%~90% , non-condensing
Vibration	Operating, 5 Grms, 5-500 Hz, 3 Axes (w/ SSD, w/o add-on card, according to IEC60068-2-64)
Shock	Operating, 50 Grms, Half-sine 11 ms Duration (w/ SSD, w/o add-on card, according to IEC60068-2-27)
EMC	CE/FCC Class A, according to EN 55022 & EN 55024

1.2.6 Specification of CAN Bus

CAN Bus port is only available on Nuvo-2510VTC. The other standard products of Nuvo-2500 Series don't support this functionality. The following table lists the specification of CAN Bus.

Table 6: Specification of CAN Bus

Item		Specification
CAN	Port Number	1
	Connector	DB9 Male Connector
	Version	CAN 2.0A & 2.0B
	Max. Speed	1Mbps
	Signal	CAN_H, CAN_L
	Isolation	Non-isolated

1.2.7 Specification of DI/O

DI/O is available on Nuvis-2520at. And it's also available as an option on Nuvo-2500. Nuvo-2510VTC doesn't support this functionality. The following table lists the specification of DI/O

Table 7: Specification of digital inputs and outputs

Item		Specification
DI	Port Number	4
	Wiring Type	Sink type
	Interface	Unipolar photo-coupler
	Isolation	3750 Vrms
	Rated Input Voltage	24VDC
	Max. Input Voltage	24VDC
	Logic High Voltage	5~24V
	Logic Low Voltage	0~1.5V
DO	Port Number	8
	Wiring Type	Sink type
	Interface	MOSFET, open drain
	Rated Driving Voltage	24VDC
	Max. Driving Voltage	30VDC
	Rated Driving Current	200mA
	Peak Driving Current	1A
	On Time	<2µs
	Off Time	<0.3µs

1.2.8 Specification of PWM Output

Pulse-width modulation, also known as PWM, output is available on Nuvis-2520at. And it's also available as an option on Nuvo-2500. Nuvo-2510VTC doesn't support this functionality. The following table lists the specification of PWM

Table 8: Specification of PWM Output

Item		Specification
PWM	Port Number	6
	Output Type	Voltage Output
	Output High Voltage	5V
	Output Low Voltage	0V
	Output Current	3mA
	Max. Output Frequency	500KHz
	Min. Output Frequency	20Hz
	Min. Pulse-Width	0.25μs
	Min. Pulse-Width Resolution	0.0125μs

1.2.9 Specification of Quadrature Encoder Input

Quadrature encoder input is available on Nuvis-2520at. And it's also available as an option on Nuvo-2500. Nuvo-2510VTC doesn't support this functionality. The following table lists the specification of PWM

Table 9: Specification of Quadrature Encoder Input

Item		Specification
Quadrature Encoder	Port Number	1
	Input Interface	5V pull-high with 2.2KΩ internal resistor for open-collector encoder
	Power Supply	12VDC, 50mA
	Signals	Phase A/B and Out/Dir configurable
	Max. Frequency	100KHz / single phase
	Counter Range	32-bit

1.2.10 Specification of ADC

Analog-to-digital converter, also known as ADC, is available on Nuvis-2520at. And it's also available as an option on Nuvo-2500. Nuvo-2510VTC doesn't support this functionality. The following table lists the specification of ADC

Item		Specification
ADC	Port Number	2
	Input Range	0~33VDC
	ADC Resolution	12-bit
	Effective Resolution	TBC

NOTE *The analog inputs are mainly for human interface devices with analog output signals, such as hall sensor joystick and analog stick, and are not highly calibrated. It's not suitable for any measurement applications.*

Chapter 2 Getting to Know your Nuvo-2500

2.1 Unpacking your Nuvo-2500 Series

When you receive the package of Nuvo-2500 Series, please check immediately if the package contains all the items listed in the following table. If any item is missing or damaged, please contact your local dealer or Neousys Technology Inc. for further assistance.

Nuvo-2500P and Nuvo-2500E

The following table shows the packing list of Nuvo-2500P and Nuvo-2500E. According to the configuration, accessories might have been assembled onto the systems upon you receive the system.

Table 10: Nuvo-2500 Packing List

Item	Description	Qty
1	Nuvo-2500 Intel® Celeron® Bay Trail Fanless Computer with Expansion Cassette (According to the configuration you order, your system may contain DDR3L module or SATA HDD. Please verify these items if necessary.)	1
2	Accessory box, which contains	
	• Neousys Drivers & Utilities DVD	1
	• Wall-mounting bracket	1
	• 3-pin pluggable terminal block	1
	• M4 P-Head screws for wall-mounting bracket	4
	• M3 F-Head screws (if HDD not installed)	4

Nuvo-2510VTC

The following table shows the packing list of Nuvo-2510VTC. According to the configuration, accessories might have been assembled onto the systems upon you receive the system.

Table 11: Nuvo-2510VTC Packing List

Item	Description	Qty
1	Nuvo-2510VTC Intel® Atom™ Bay Trail In-Vehicle Fanless Computer with 2x IEEE 802.3at PoE+ Ports (According to the configuration you order, your system may contain DDR3L module or SATA HDD. Please verify these items if necessary.)	1
2	Accessory box, which contains	
	• Neousys Drivers & Utilities DVD	1
	• Wall-mounting bracket	1
	• 3-pin pluggable terminal block	1
	• M4 P-Head screws for wall-mounting bracket	4
	• M3 F-Head screws (if HDD not installed)	4
	• Shock-absorbing grommet	4
	• M4 I-Type screws	8

Nuvis-2520at-P and Nuvis-2520at-E

The following table shows the packing list of Nuvis-2520at-P and Nuvis-2520at-E. According to the configuration, accessories might have been assembled onto the systems upon you receive the system.

Table 12: Nuvis-2520at Packing List

Item	Description	Qty
1	Nuvis-2520at Intel® Celeron® Bay Trail Machine Vision Fanless Computer with Expansion Cassette (According to the configuration you order, your system may contain DDR3L module or SATA HDD. Please verify these items if necessary.)	1
2	Accessory box, which contains	
	• Neousys Drivers & Utilities DVD	1
	• Wall-mounting bracket	1
	• 3-pin pluggable terminal block	1
	• M4 P-Head screws for wall-mounting bracket	4
	• M3 F-Head screws (if HDD not installed)	4

2.2 Front Panel I/O Functions

2.2.1 Power Button

Figure 1: Power Button

The power button is a non-latched switch for ATX mode on/off operation. To turn on the Nuvo-2500, press the power button and the PWR LED is lighted up. To turn off the Nuvo-2500 you can either issue a shutdown command in OS, or just simply press the power button. In case of system halts, you can press and hold the power button for 5 seconds to compulsorily shut down the system. Please note that a 5 seconds interval is kept by the system between two on/off operations (i.e. once turning off the system, you shall wait for 5 seconds to initiate another power-on operation).

2.2.2 Reset Button

Figure 2: Reset Button

The reset button is used to manually reset the system in case of any abnormal condition. To avoid unexpected operation, the reset button is hidden behind the front panel. You need to use a pin-like object to push the reset button.

2.2.3 LED Indicators

Figure 3: Nuvo-2500 and Nuvis-2520at LED Indicators

There are four LED indicators on the front panel of Nuvo-2500 Series. Nuvo-2500 / Nuvis-2520at have identical LED Indicators. The following table describes the indicators.

Table 13: Definition of Nuvo-2500 and Nuvis-2520at LED Indicators

Indicator	Color	Description
PWR	Green	Power indictor, lighted-up when system is on.
HDD	Red	Hard drive indicator, flashing when SATA HDD is active.
WDT	Amber	Watchdog timer indicator, flashing when watchdog timer is started.
UID	Green	Reserved

Figure 4: Nuvo-2510VTC LED Indicators

Nuvo-2510VTC has a dedicated indicator for ignition power control. The details are provided in the section "*Ignition Mode Switch*". *Table 14* shows the definition of indicators.

Table 14: Definition of Nuvo-2510VTC LED Indicators

Indicator	Color	Description
PWR	Green	Power indictor, lighted-up when system is on.
HDD	Red	Hard drive indicator, flashing when SATA HDD is active.
WDT	Amber	Watchdog timer indicator, flashing when watchdog timer is started.
UID	Green	Ignition signal indicator, lighted-up when IGN is high (12V/24V).

2.2.4 USB 3.0 Connectors

Figure 5: USB 3.0 Connector

There are totally one USB 3.0 port and three USB 2.0 ports on Nuvo-2500 Series. By BIOS default, one USB 3.0 port is operated in xHCI (eXtensible Host Control Interface) mode and is compatible to USB 3.0, USB 2.0, USB 1.1 and USB 1.0 devices. Legacy USB support on the port is provided only when **[xHCI Mode]** option in BIOS is configured as **[Smart Auto]**.

2.2.5 USB2.0 Connectors

Figure 6: USB 2.0 Connectors

In addition to one USB 3.0 port, Nuvo-2500 Series provides three USB 2.0 ports. By BIOS default, the three USB 2.0 ports are operated in EHCI (Enhanced Host Control Interface) mode and are compatible with USB 2.0, USB 1.1 and USB 1.0 devices. Legacy USB support is provided so you can use USB keyboard/mouse in DOS environment.

2.2.6 Gigabit Ethernet Ports

Figure 7: Nuvo-2500 Gigabit Ethernet Ports

Nuvo-2500 Series offers two Gigabit Ethernet ports using Intel® I210 GbE controller. When plugging in the Ethernet cable, you can tell the Ethernet status and speed from the LED indicators on the RJ45 connector. The following tables provide the definition of the LEDs.

Table 15: Gigabit Ethernet Active/Link LED

LED Color	Status	Description
Yellow	Off	Ethernet port is disconnected
	On	Ethernet port is connected and no data transmission
	Flashing	Ethernet port is connected and data is transmitting/receiving

Table 16: Gigabit Ethernet Speed LED

LED Color	Status	Description
Green or Orange	Off	10 Mbps
	Green	100 Mbps
	Orange	1000 Mbps

To utilize the ports in Windows, you need to install corresponding driver for Intel® I210 GbE controller. Please refer to the section "*Driver Installation*"

2.2.7 IEEE 802.3at PoE PSE

PoE, standing for Power over Ethernet, is a technology to supply electrical power along with data on a standard CAT-5/CAT-6 Ethernet cable. two Gigabit Ethernet ports on Nuvo-2510VTC and Nuvis-2520at support IEEE 802.3at PoE PSE (Power Sourcing Equipment) function. Each PoE port can deliver 25.5 W of power to a PoE PD (Powered Device), such as a PoE IP camera or a PoE WIFI AP. PoE defines a mechanism to automatically detect the device connected and determine whether to dispatch power. This makes PoE port 100% compatible with traditional Ethernet devices thus you can use these ports to connect PoE or non-PoE devices.

Power over Ethernet, also known as PoE, is available on Nuvo-2510VTC and Nuvis-2520at. Nuvo-2500 doesn't support this functionality. The following figures show PoE Port of Nuvo-2510VTC and Nuvis-2520at

Figure 8: Nuvo-2510VTC PoE PSE Port

Figure 9: Nuvis-2520at PoE PSE Port

2.2.8 Software-programmable COM Ports

Figure 10: Software-programmable COM Ports

On the front panel, Nuvo-2500 Series provides two software-selectable COM ports for communicating with external devices. Each port supports RS-232/422/485, and the operation mode of each port can be set in BIOS setup utility. COM ports are implemented using industrial-grade ITE8783 Super IO chip (-40 to 85°C) and provide up to 115200 bps baud rate. The following table describes the pin definition of two software-selectable COM ports.

Table 17: Pin Definition of COM1 and COM 2

Pin No.	COM1 / COM2		
	RS-232 Mode	RS-422 Mode	RS-485 Mode (Two-wire 485)
1	DCD		
2	RX	422 TXD+	485 TXD+/RXD+
3	TX	422 RXD+	
4	DTR	422 RXD-	
5	GND	GND	GND
6	DSR		
7	RTS		
8	CTS	422 TXD-	485 TXD-/RXD-
9	RI		

2.2.9 VGA Port

Figure 11: VGA Port

Nuvo-2500 Series has one VGA display outputs port. VGA connector is the most popular way for connecting a display. By BIOS default and hardware implementation, the VGA output is always enabled in any case. To achieve best VGA output resolution in Windows, you need to install corresponding graphics driver. Please refer to "*Driver Installation*" for information of driver installation.

2.2.10 DC Input

Figure 12: DC Input Connector

Nuvo-2500 Series features a pluggable terminal block for direct DC wiring. The 3-pin pluggable terminal block is fit for field usage where DC power is usually provided. It accepts a wide range of DC power input from 8 to 35V. And the screw clamping connection of terminal block gives a very reliable way for wiring the DC power. The following table describes the pin definition of the DC Input Connector.

Table 18: Nuvo-2500 and Nuvis-2520at DC input connector

Pin No.	Name	Description
1	V+	Positive polarity of DC power input (8 ~ 35V).
2	GND	Negative polarity of DC power input
3	-	Reserved

Table 19: Nuvo-2510VTC DC input connector

Pin No	Name	Description
1	V+	Positive polarity of DC power input (8 ~ 35V).
2	GND	Negative polarity of DC power input
3	IGN	Ignition input for ignition power control

2.3 Back Panel I/O Functions

2.3.1 DVI-D/HDMI Connector

Figure 13: DVI-D/HDMI Connector

Nuvo-2500 Series has one DVI-D/HDMI display output port. DVI/HDMI transmits graphics data in digital format and therefore can deliver better image quality at high resolution. The port can either output DVI signals or HDMI signal depending on the display device connected. You shall need a DVI to HDMI cable when connecting to a HDMI display device.

NOTE *The connector is a DVI-I connector. But the analog signal is not connected inside Nuvo-2500. This port is not compatible with a passive DVI-I to VGA adapter.*

2.3.2 COM Ports

Figure 14: RS-232 COM Ports

Nuvo-2500 Series provides two COM ports for communicating with external devices. Each port supports RS-232 only. COM ports are implemented using industrial-grade ITE8783 Super IO chip (-40 to 85°C) and provide up to 115200 bps baud rate. The following table describes the pin definition of two COM ports.

Table 20: Pin Definition of COM3 and COM4

Pin#	COM3 / COM4
1	DCD
2	RX
3	TX
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	RI

2.3.3 Speaker-out and Mic-in Jacks

Figure 15: Speaker-out and Mic-in Jacks

Nuvo-2500 Series provides audio functions using Intel® High Definition Audio and Realtek ALC262 codec. There are two audio jacks. The pink one is used for microphone input, and the green one is for speaker output. To utilize the microphone input function in Windows, you need to install corresponding drivers for both Intel® chipset and Realtek ALC262 codec. Please refer to the section "*Driver Installation*".

2.3.4 CAN Bus Port

Figure 16: CAN Bus Port on Nuvo-2510VTC

CAN Bus port is only available on Nuvo-2510VTC. The other standard products of Nuvo-2500 Series don't support this functionality. CAN Bus is a reliable industrial bus with a pair of differential signal and used in many industrial fields and also in vehicles. Nuvo-2510VTC equips with a CAN bus port for both industrial and in-vehicle applications. The CAN Bus port supports CAN2.0A and CAN2.0B up to 1Mbps. The port is a DB9 connector and the pin definition shows in the follow table.

Table 21: Pin Definition of CAN Bus Port on Nuvis-2520at

Pin No.	Definition	I/O	Description
1	Reserved	-	Reserved pin. Keep unconnected
2	Reserved	-	Reserved pin. Keep unconnected
3	CAN_H	I/O	CAN Bus High-level voltage
4	Reserved	-	Reserved pin. Keep unconnected
5	CAN_L	I/O	CAN Bus Low-level voltage
6	Reserved	-	Reserved pin. Keep unconnected
7	Reserved	-	Reserved pin. Keep unconnected
8	Reserved	-	Reserved pin. Keep unconnected
9	Reserved	-	Reserved pin. Keep unconnected

2.3.5 Auxiliary I/O Port of Nuvo-2500

Figure 17: Auxiliary I/O on Nuvo-2500

The Auxiliary I/O port on Nuvo-2500 is always mounted. However this port works only on some models of Nuvo-2500 which is shipped with specific micro-controller codes. The behavior of the Auxiliary I/O Port on a standard Nuvo-2500 is undefined. Please keep this port unconnected on a standard Nuvo-2500.

Table 22: Pin Definition of Auxiliary I/O on Nuvo-2500

Pin No.	Definition	I/O	Description
1	VDD	-	DO voltage source input for inductive load
2	DO_0	O	Digital output channel 0
3	DO_1	O	Digital output channel 1
4	DO_2	O	Digital output channel 2
5	DO_3	O	Digital output channel 3
6	DO_GND	-	Digital output GND
7	DI_0	I	Digital input channel 0
8	DI_1	I	Digital input channel 1
9	DI_GND	-	Digital input GND
10	EZ	I	Encoder Index input
11	GND	-	GND of Encoder/PWM/ADC/12V
12	EA	I	Encoder Phase A input
13	EB	I	Encoder Phase B input
14	12V	-	Non-isolated 12V power supply
15	GND	-	GND of Encoder/PWM/AI/12V
16	AI_1	I	Voltage input channel 1
17	AI_0	I	Voltage input channel 0

18	Reserved	-	Reserved pin. Keep unconnected
19	Reserved	-	Reserved pin. Keep unconnected
20	ISO_5V	-	Isolated 5V power supply
21	DO_4	O	Digital output channel 4
22	DO_5	O	Digital output channel 5
23	DO_6	O	Digital output channel 6
24	DO_7	O	Digital output channel 7
25	DO_GND	-	Digital output GND
26	DI_2	I	Digital input channel 2
27	DI_3	I	Digital input channel 3
28	DI_GND	-	Digital input GND
29	PWM_0	O	PWM output channel 0
30	PWM_1	O	PWM output channel 1
31	PWM_2	O	PWM output channel 2
32	PWM_3	O	PWM output channel 3
33	GND	-	GND of Encoder/PWM/AI/12V
34	PWM_4	O	PWM output channel 4
35	PWM_5	O	PWM output channel 5
36	Reserved	-	Reserved pin. Keep unconnected
37	Reserved	-	Reserved pin. Keep unconnected

2.3.6 Auxiliary I/O Port of Nuvis-2520at

Figure 18: Auxiliary I/O on Nuvis-2520at

The Auxiliary I/O port on Nuvis-2520at is a DB37 connector. This connector provides six different types of I/O interface, including digital inputs, digital outputs, analog inputs, pulse width modulation outputs and a quadrature encoder input. Controlled by a built-in micro-controller, these I/O can react autonomously according to a pre-defined behavior. This is titled as DTIO, stand for deterministic trigger I/O. Following is the pin definition of the Auxiliary I/O port.

Table 23: Pin Definition of Auxiliary I/O on Nuvis-2520at

Pin No.	Definition	I/O	Description
1	VDD	-	DO voltage source input for inductive load
2	DO_0	O	Digital output channel 0
3	DO_1	O	Digital output channel 1
4	DO_2	O	Digital output channel 2
5	DO_3	O	Digital output channel 3
6	DO_GND	-	Digital output GND
7	DI_0	I	Digital input channel 0
8	DI_1	I	Digital input channel 1
9	DI_GND	-	Digital input GND
10	EZ	I	Encoder Index input
11	GND	-	GND of Encoder/PWM/ADC/12V
12	EA	I	Encoder Phase A input
13	EB	I	Encoder Phase B input
14	12V	-	Non-isolated 12V power supply
15	GND	-	GND of Encoder/PWM/AI/12V

16	AI_1	I	Voltage input channel 1
17	AI_0	I	Voltage input channel 0
18	Reserved	-	Reserved pin. Keep unconnected
19	Reserved	-	Reserved pin. Keep unconnected
20	ISO_5V	-	Isolated 5V power supply
21	DO_4	O	Digital output channel 4
22	DO_5	O	Digital output channel 5
23	DO_6	O	Digital output channel 6
24	DO_7	O	Digital output channel 7
25	DO_GND	-	Digital output GND
26	DI_2	I	Digital input channel 2
27	DI_3	I	Digital input channel 3
28	DI_GND	-	Digital input GND
29	PWM_0	O	PWM output channel 0
30	PWM_1	O	PWM output channel 1
31	PWM_2	O	PWM output channel 2
32	PWM_3	O	PWM output channel 3
33	GND	-	GND of Encoder/PWM/AI/12V
34	PWM_4	O	PWM output channel 4
35	PWM_5	O	PWM output channel 5
36	LED+	-	LED power supply positive
37	LED-	-	LED power supply negative

2.4 Internal I/O Functions

2.4.1 Mini PCI Express Connector and Internal SIM Socket

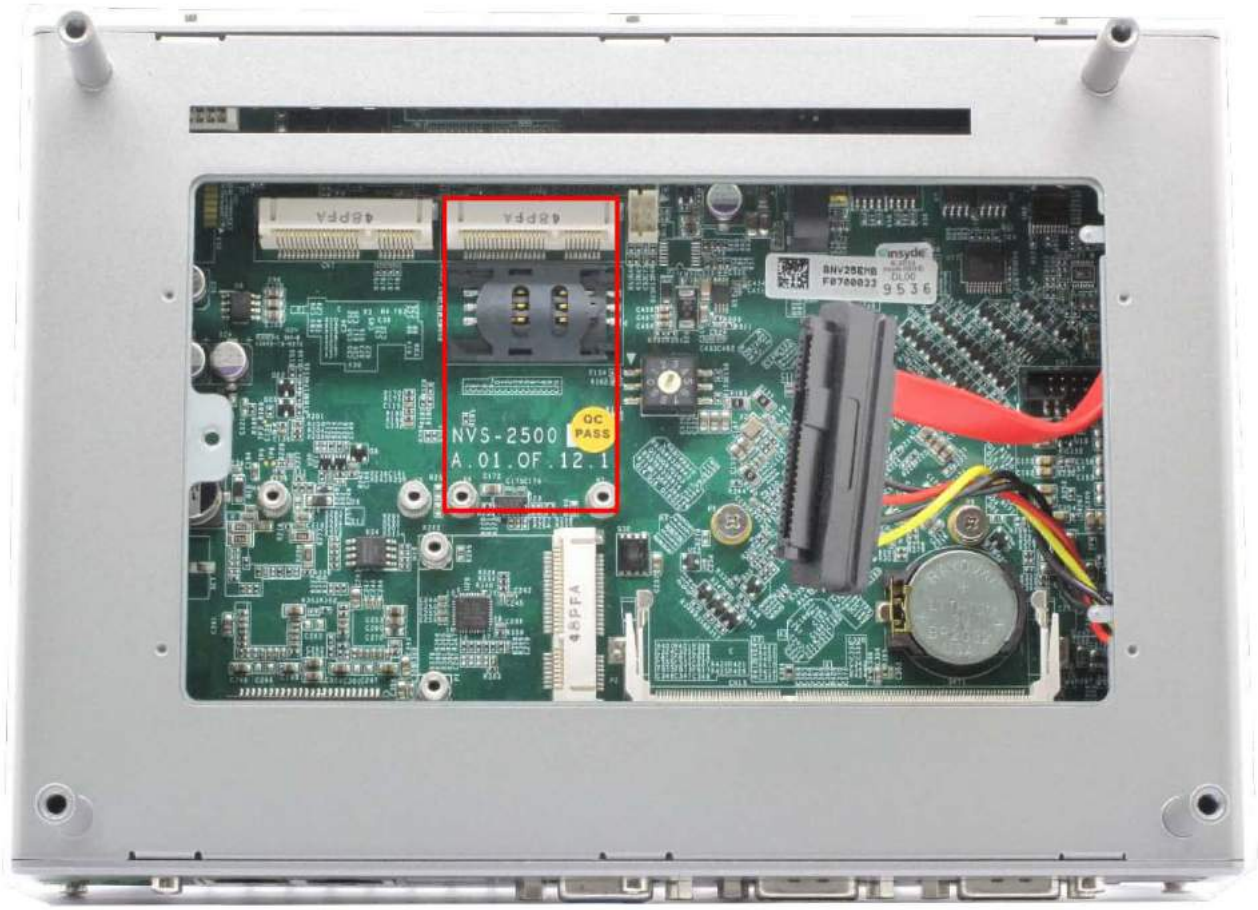


Figure 19: Mini PCI Express Connector and Internal SIM Socket

Nuvo-2500 Series provides an on-board Mini PCI Express socket with SIM card support. Supporting both PCI Express and USB signal, the socket is compatible with plenty of off-the-shelf mini-PCIe modules with versatile capabilities. By installing a mini-PCIe module, your system can have expanded features such as WIFI, 3G, GPS, RAID and etc. In addition, the SIM card support makes it possible to connect your system to Internet in wide territory through telecom operator's GPRS/3G/LTE network.

2.4.2 USB Only Mini PCI Express Connector

Figure 20: Mini PCI Express Connector with only USB Signal

Nuvo-2500 Series provides an on-board Mini PCI Express socket with only USB signal. The socket is compatible with mini-PCIe modules with only USB signal, such as WWAN and GPS cards. It's also compatible with a mPCIe-to-USB module, which make it possible to use an internal USB dongle or a wireless USB receiver.

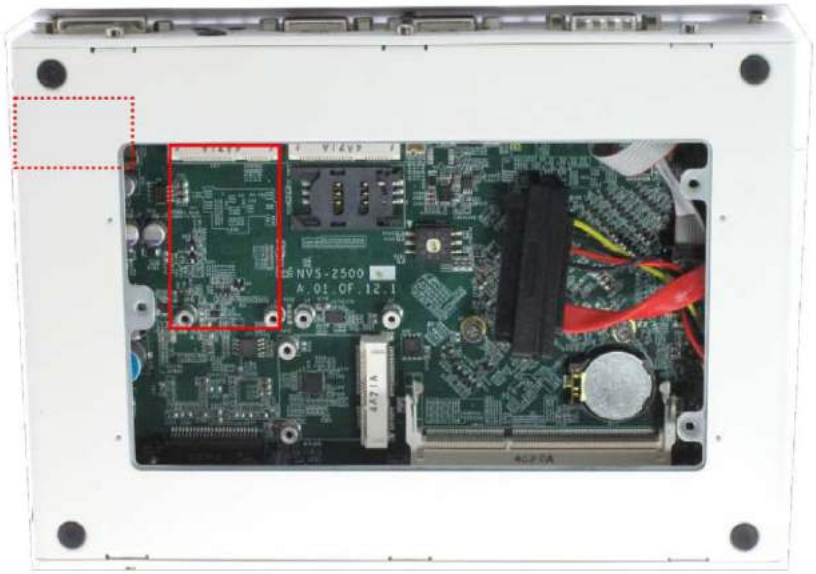


Figure 21: Mini PCI Express Connector with only USB Signal and External SIM Socket

On Nuvo-2510VTC, this USB only mini-PCIe connector also supports an external SIM socket. This makes Nuvo-2510VTC idea for applications which needs two SIM card.

2.4.3 Half-size mSATA Connector

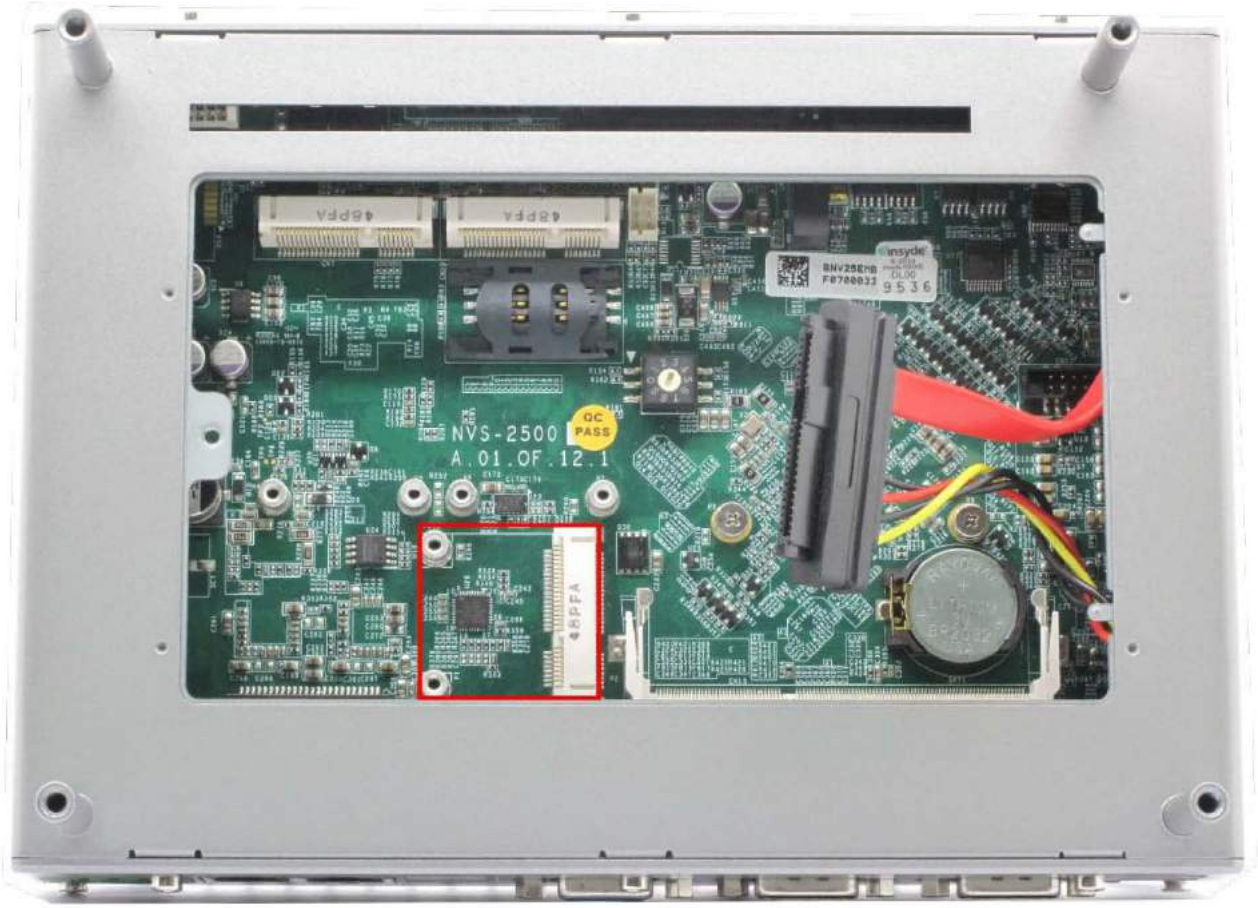


Figure 22: mSATA Connector with USB Signal

Nuvo-2500 Series provides one half-size mSATA connector. The connector comes with mSATA signal and USB signal. You can connect an mSATA module to this port and make your Nuvo-2500 Series a dual storage system. On the other hand, this connector also behaves like the USB only Mini PCI Express Connector mentioned in the section "*USB Only Mini PCI Express Connector*"

2.4.4 DDR3L SODIMM Socket

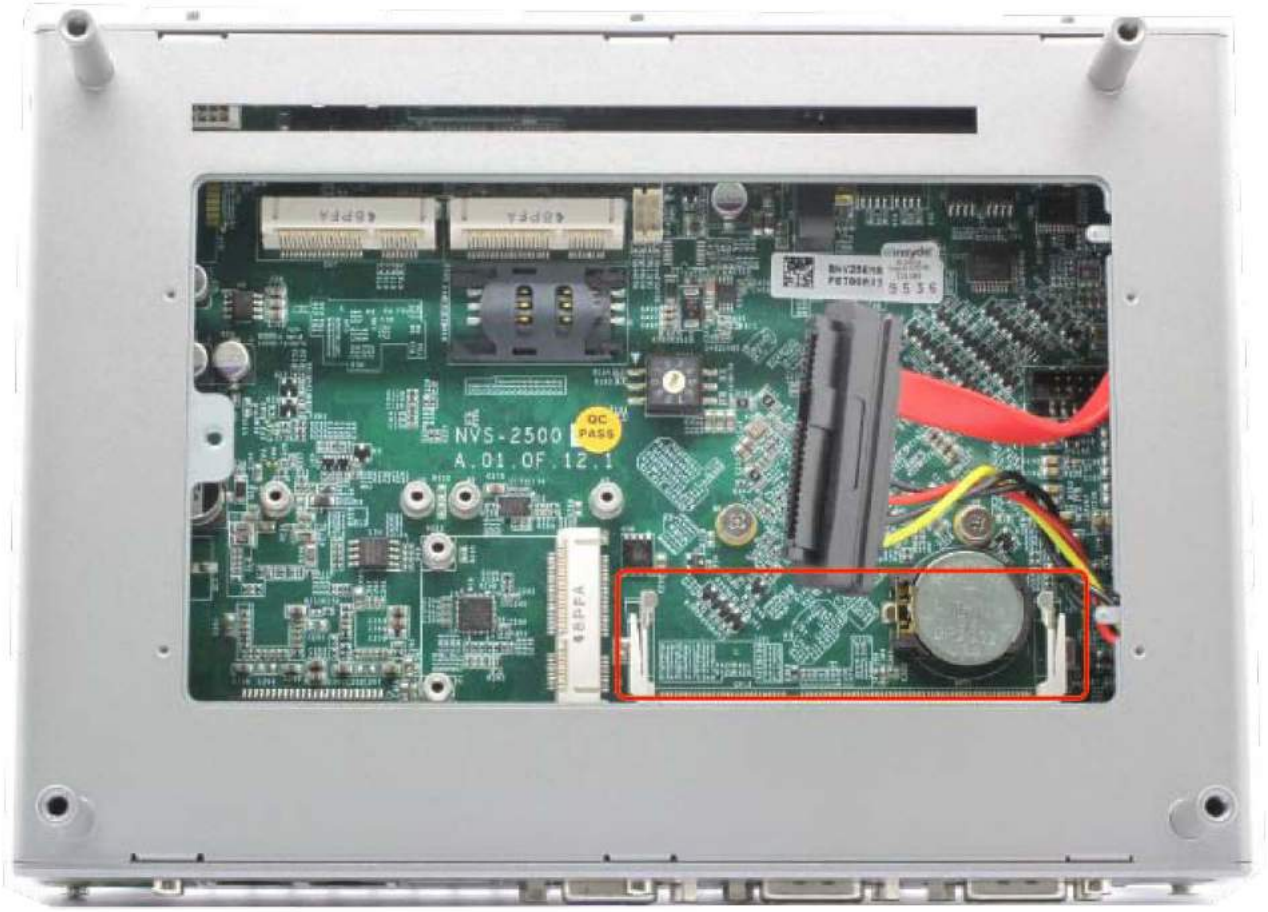


Figure 23: DDR3L SODIMM Socket

Nuvo-2500 Series provides one 204-pin SODIMM socket for memory installation. It supports a maximal 8GB capacity by installing one low-voltage 1.35V DDR3L-1333 SODIMM modules. For information of installing DDR3L memory modules, please refer to the section "Install a DDR3L SODIMM Module".

NOTE *Installing incorrect memory module might damage the system or result in system failure. Please make sure you're installing a 1.35V DDR3L SODIMM module.*

2.4.5 SATA Connector

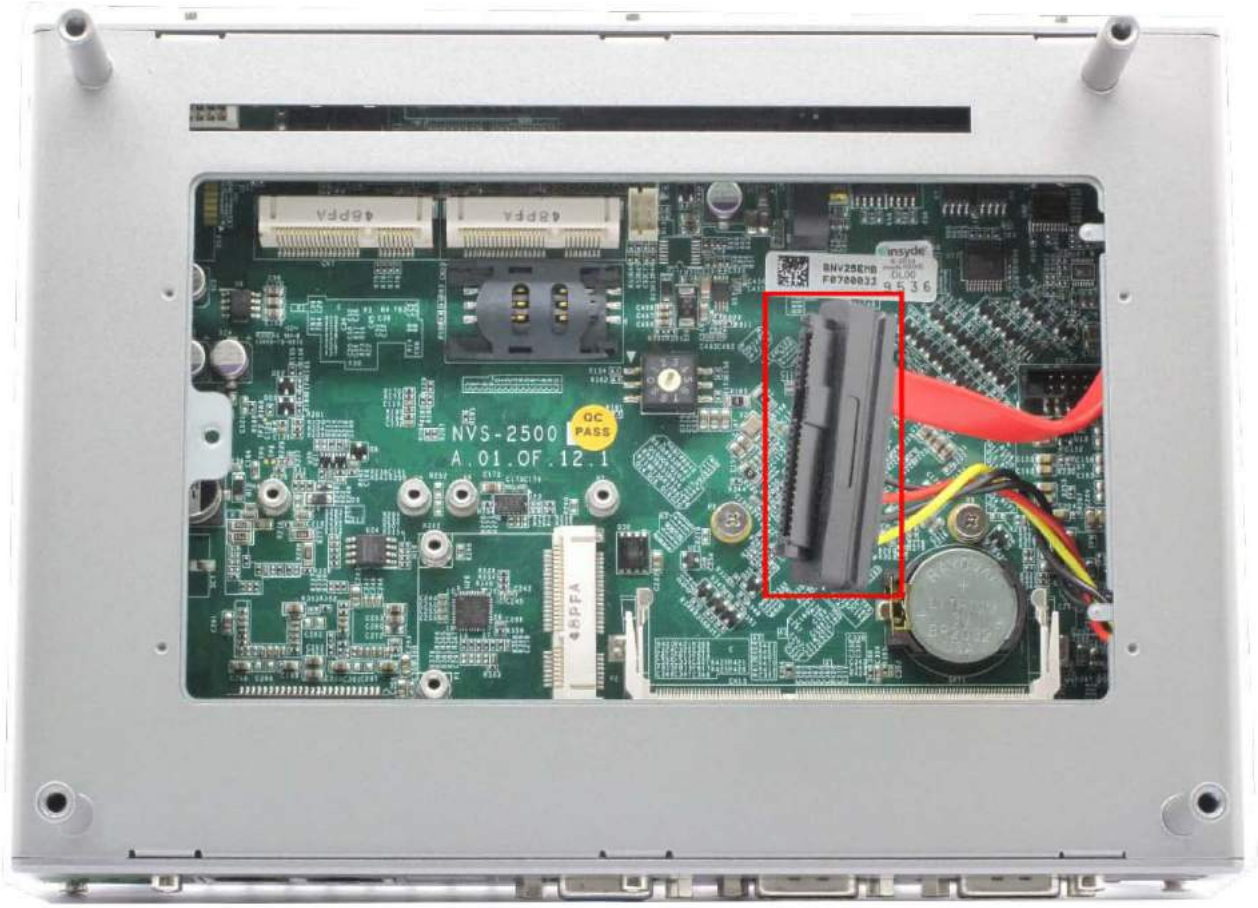


Figure 24: SATA Cable for 2.5" HDD or SSD

Nuvo-2500 Series provides one SATA port to accommodate a 2.5" SATA hard drive via a SATA cable. You can connect a 2.5" HDD or SSD to this port. Please refer to the section "*Install a 2.5" HDD/SSD*" for detail on installing an HDD/SSD to SATA port.

2.4.6 Ignition Mode Switch

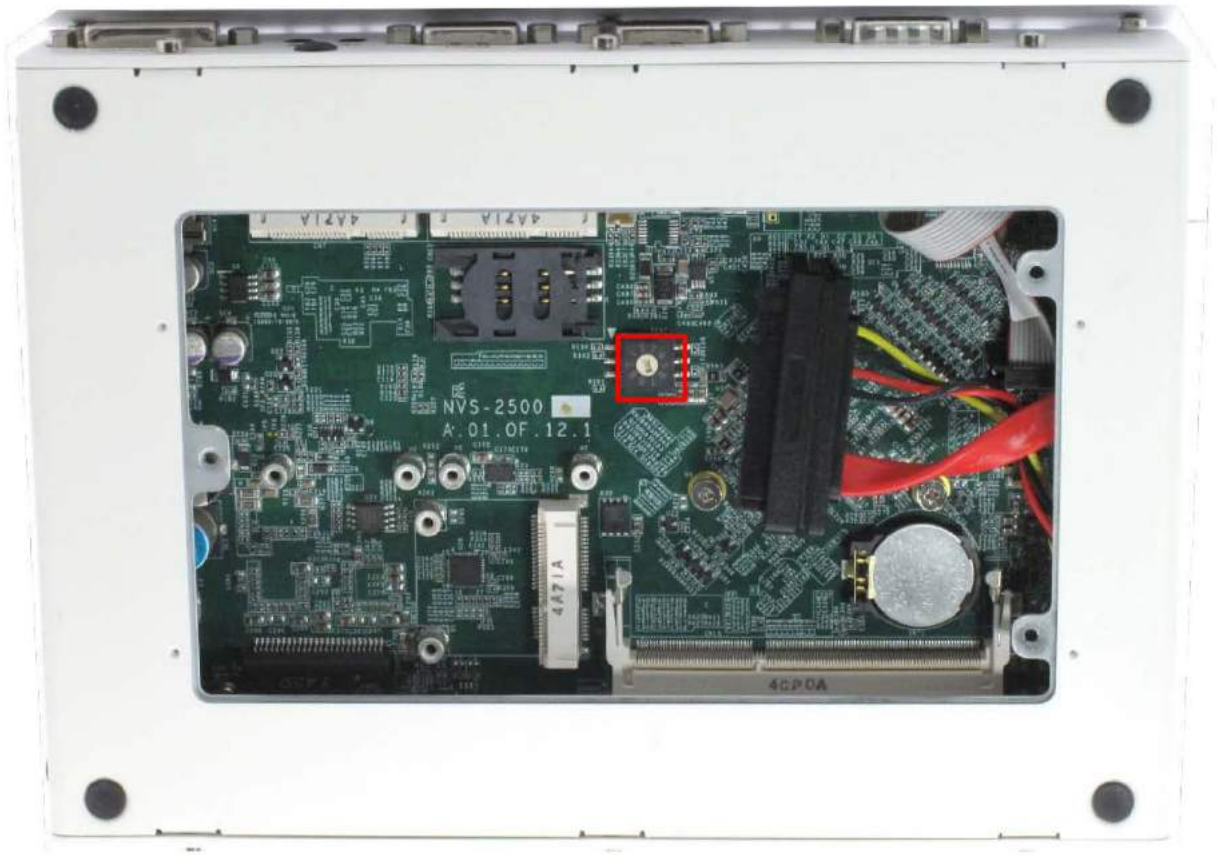


Figure 25: Ignition Switch

Nuvo-2510VTC offers ignition power control module for in-vehicle applications. There is an on-board rotary switch to configure the operation mode of ignition power control. Please refer to the section "*Ignition Power Control*" for information of using ignition power control.

2.5 Expansion Cassette

Expansion cassette is available on Nuvo-2500 and Nuvis-2520at. Nuvo-2510VTC doesn't support this functionality.

2.5.1 Support Size of Add-on Cards

There are 2 types of Expansion Cassette for PCI and PCI Express cards respectively. The maximum supported card size is 167.65mm x 108.40mm with optional fan installed, and 179.65mm x 108.40mm without optional fan installed. The following figure shows the definition of the card size. And this size applies to both PCI and PCIe cards.

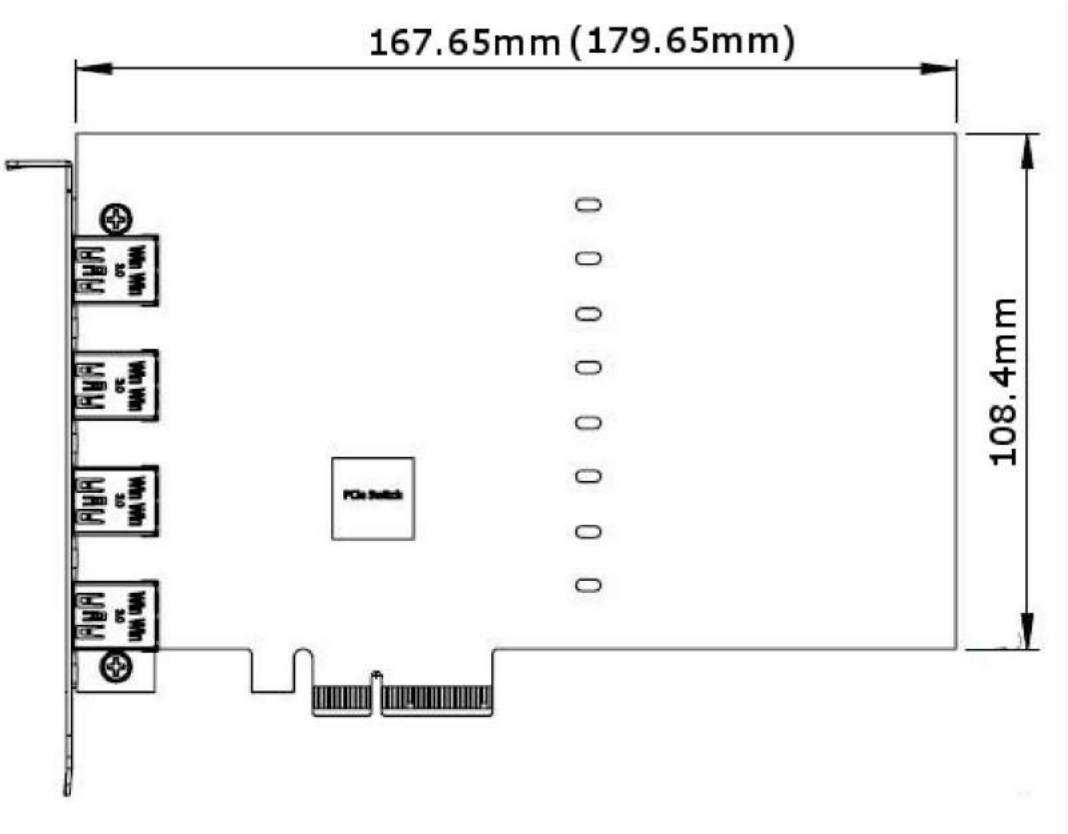


Figure 26: Maximum PCB Size of Add-on Card Supported by Expansion Cassette

NOTE *Some add-on cards have longer connects, such as BNC, on the PCI bracket. This may result in interfering while installing the card thought the PCB size is close to the limitation.*

2.5.2 Cassette of Nuvo-2500E and Nuvis-2520at-E

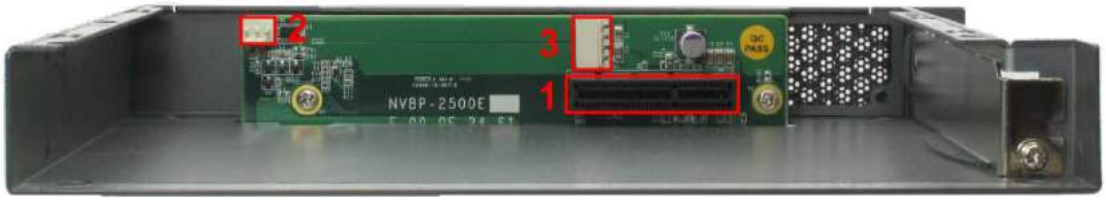
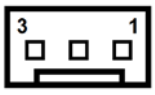


Figure 27: Cassette and PCIe Backplane

The expansion Cassette of Nuvo-2500E and Nuvis-2520at-E contains a backplane with a x4 PCI Express connector. It runs 1-lane, Gen2 PCI Express signals to provide a maximal 4GB/s bandwidth. The Expansion Cassette supports 4A@12V rated current for a PCI Express add-on card with higher power consumption. You can also get the 12VDC from the on-board power connectors if necessary. This connector shares the 12VDC with the add-on card.

Table 24: Connectors on PCIe Backplane

<table>
<tr><th>Item</th><th>Connector</th><th>Function Description</th></tr>
<tr><td>1</td><td>x4 PCI Express</td><td>x4 PCI Express connector that runs 1-lane, Gen2 signal. Compatible with x4 and x1 PCI Express add-on card.</td></tr>
<tr><td>2</td><td>12VDC Fan Power</td><td>3-pin, 2.54mm pitch power connector for supplying 12VDC to the optional fan.
 
<table><tr><th>Pin</th><th>Description</th></tr><tr><td>1</td><td>Reserved</td></tr><tr><td>2</td><td>12V</td></tr><tr><td>3</td><td>GND</td></tr></table></td></tr>
<tr><td>3</td><td>5/12VDC Power</td><td>4-pin, 2.0mm pitch wafer connector for supplying 5/12VDC
 
<table><tr><th>Pin</th><th>Description</th></tr><tr><td>1</td><td>12V</td></tr><tr><td>2</td><td>GND</td></tr><tr><td>3</td><td>GND</td></tr><tr><td>4</td><td>5V (2A rated Current)</td></tr></table></td></tr>
</table>

2.5.3 Cassette of Nuvo-2500P and Nuvis-2520at-P

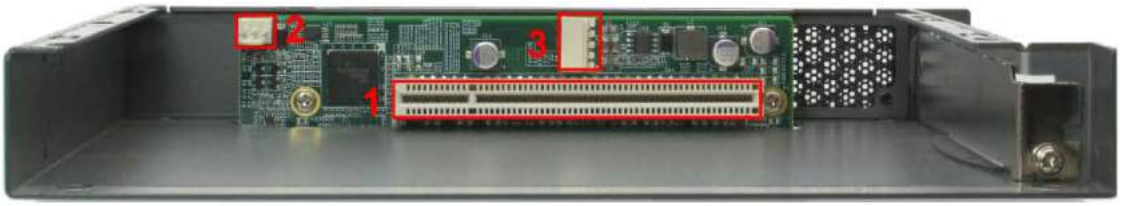


Figure 28: Cassette and PCI Backplane

The expansion Cassette of Nuvo-2500P and Nuvis-2520at-P contains a backplane with a 32-bit/33MHz PCI connector. The Expansion Cassette supports 4A@12V rated current. You can get the 12VDC from the on-board power connectors if necessary. This connector shares the 12VDC with the add-on card.

Table 25: Connectors on PCI Backplane

<table>
<tr><th>Item</th><th>Connector</th><th>Function Description</th></tr>
<tr><td>1</td><td>33MHz/32-bit PCI</td><td>33MHz/32-bit PCI bus via PLX8112 PCIe-to-PCI bridge</td></tr>
<tr><td>2</td><td>12VDC Fan Power</td><td>3-pin, 2.54mm pitch power connector for supplying 12VDC to the optional fan.
 
<table><tr><th>Pin</th><th>Description</th></tr><tr><td>1</td><td>N/A</td></tr><tr><td>2</td><td>12V</td></tr><tr><td>3</td><td>GND</td></tr></table></td></tr>
<tr><td>3</td><td>5/12VDC Power</td><td>4-pin, 2.0mm pitch wafer connector for supplying 5/12VDC
 
<table><tr><th>Pin</th><th>Description</th></tr><tr><td>1</td><td>12V</td></tr><tr><td>2</td><td>GND</td></tr><tr><td>3</td><td>GND</td></tr><tr><td>4</td><td>5V (2A rated Current)</td></tr></table></td></tr>
</table>

2.6 Mechanical Dimension

2.6.1 Nuvo-2500

Front View

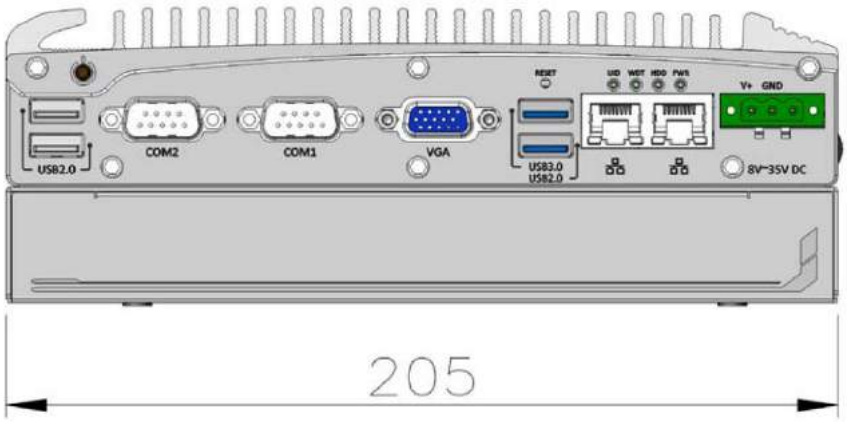


Figure 29: Front View of Nuvo-2500

Side View (Right)

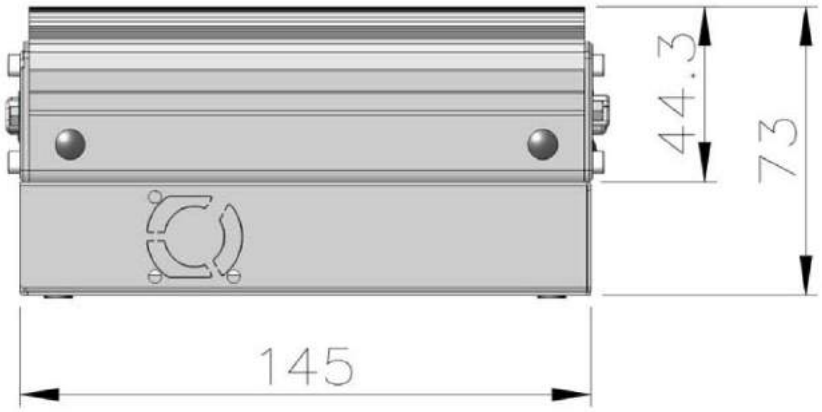


Figure 30: Side View (Right) of Nuvo-2500

Bottom View

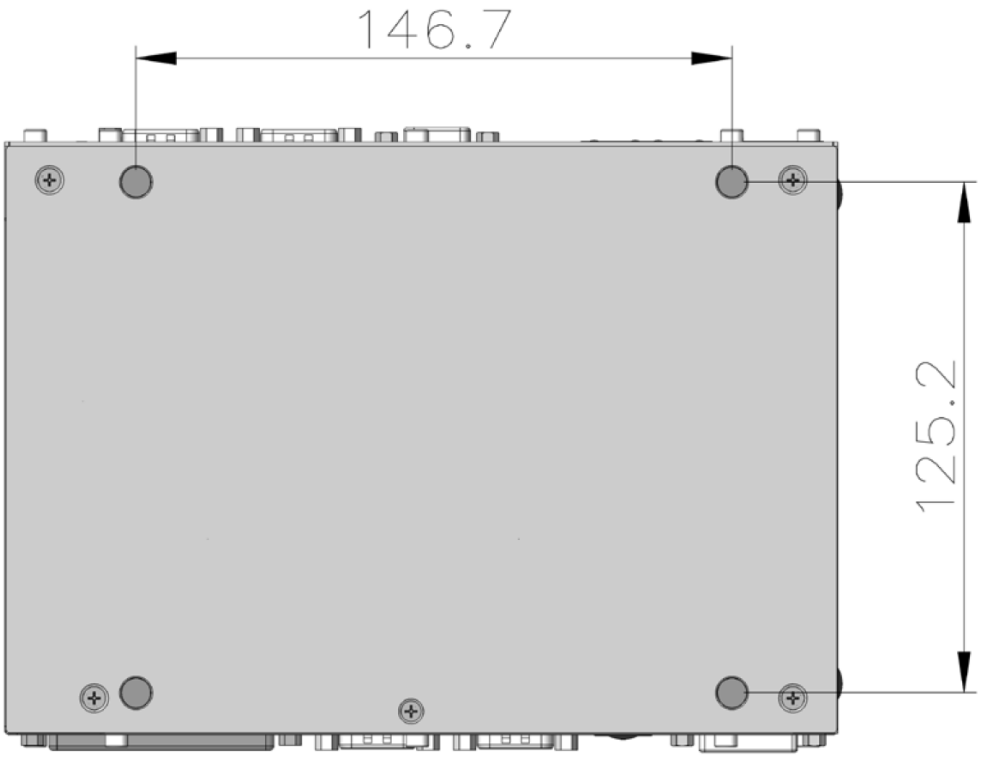


Figure 31: Bottom View of Nuvo-2500

Top View

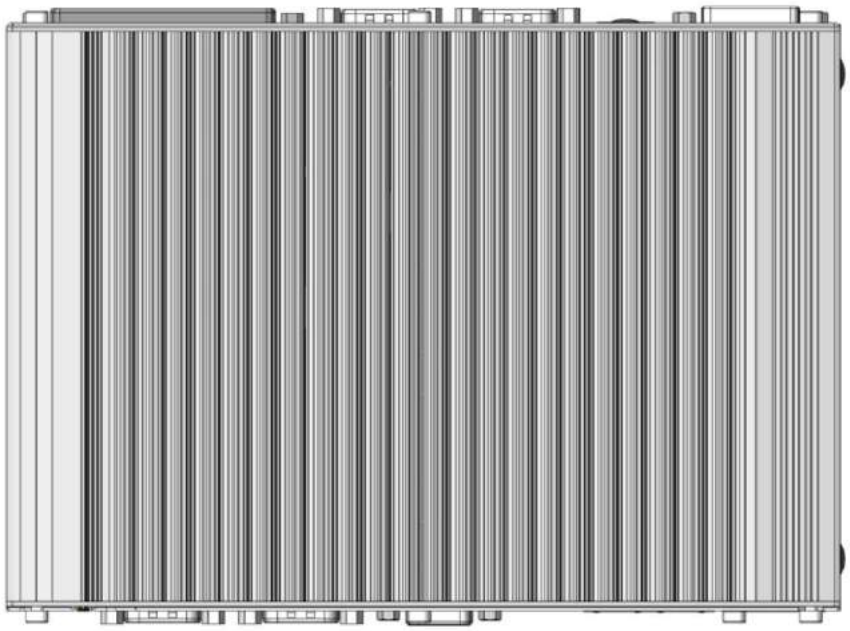


Figure 32: Top View of Nuvo-2500

2.6.2 Nuvo-2510VTC

Front View

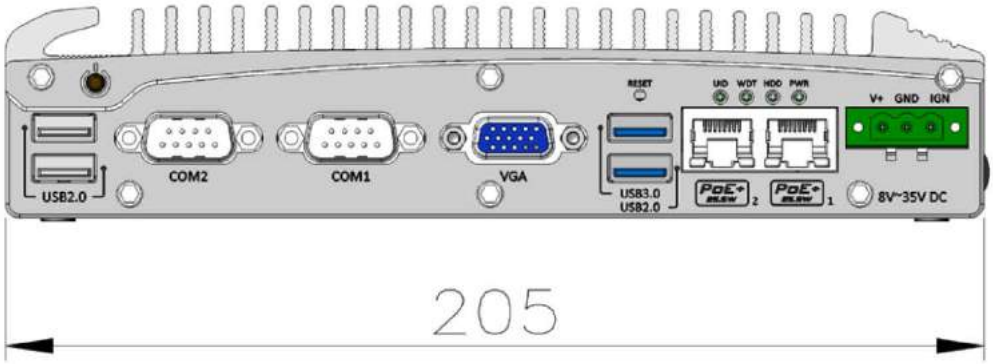


Figure 33: Front View of Nuvo-2510VTC

Side View (Right)

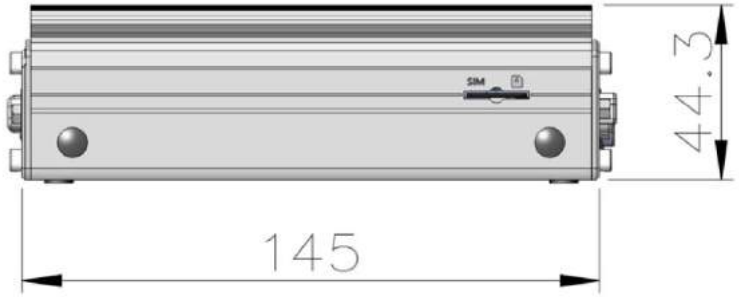


Figure 34: Side View (Right) of Nuvo-2510VTC

Bottom View

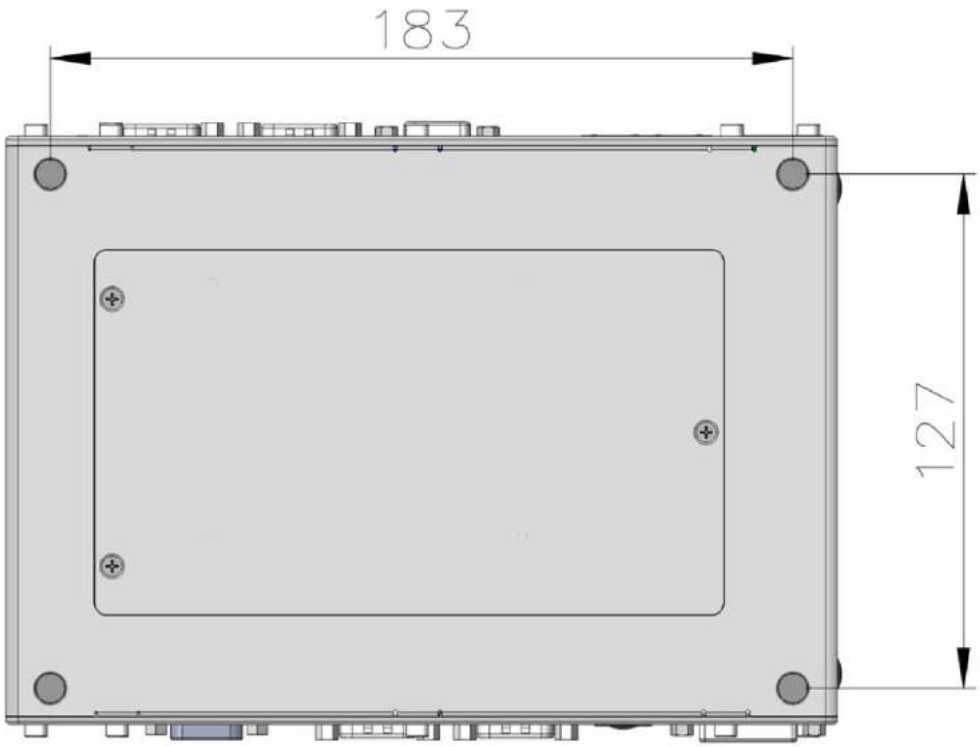


Figure 35: Bottom View of Nuvo-2510VTC

Top View

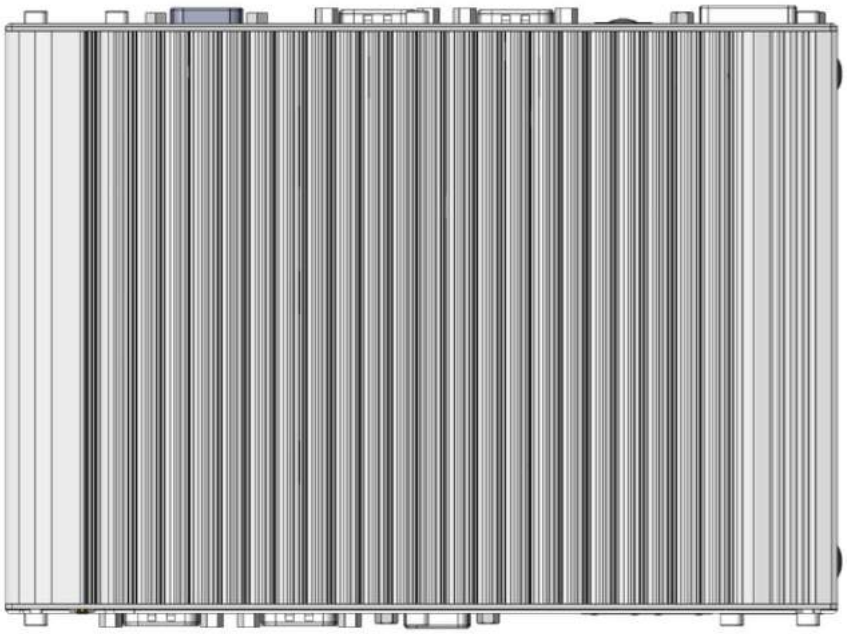


Figure 36: Top View of Nuvo-2510VTC

2.6.3 Nuvis-2520at

Front View

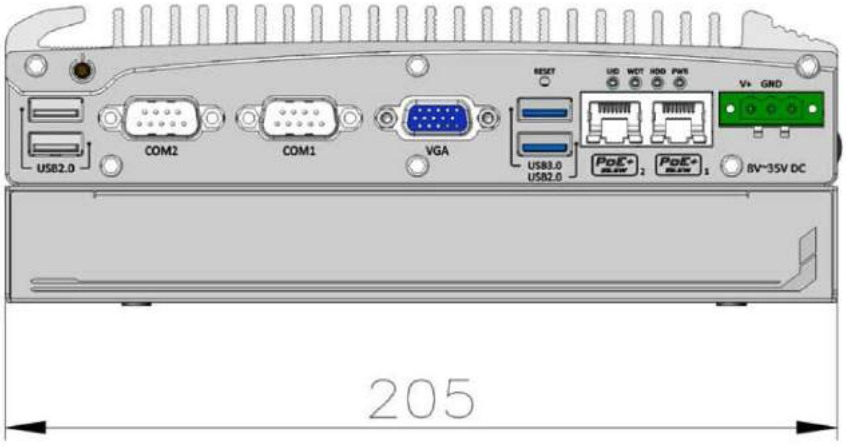


Figure 37: Front View of Nuvis-2520at

Side View (Right)

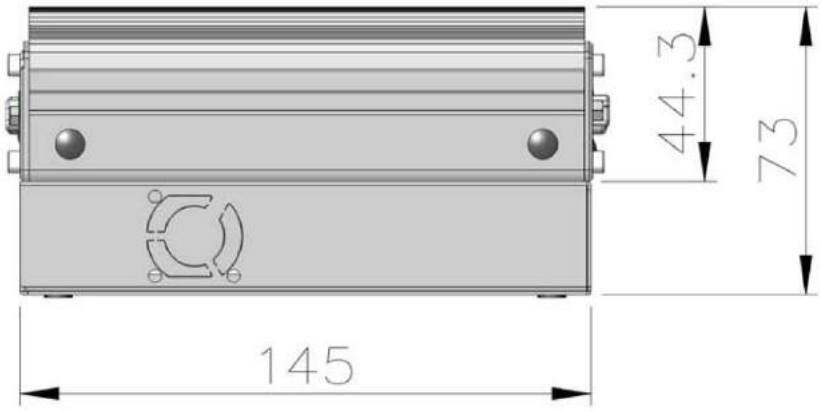


Figure 38: Side View (Right) of Nuvis-2520at

Bottom View

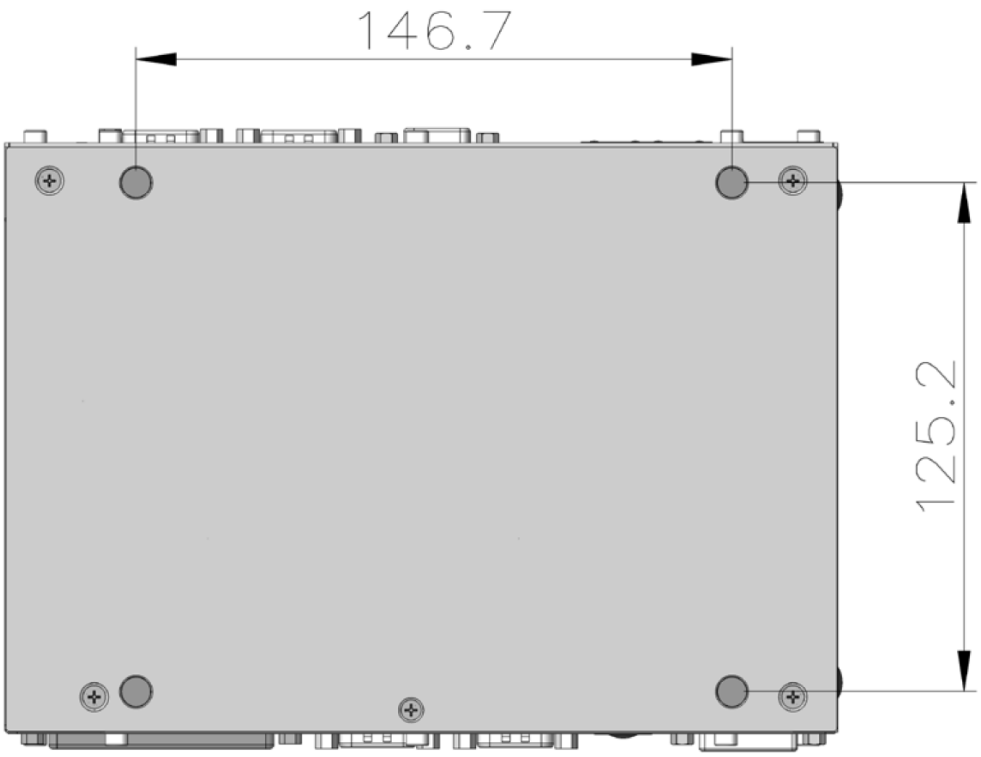


Figure 39: Bottom View of Nuvis-2520at

Top View

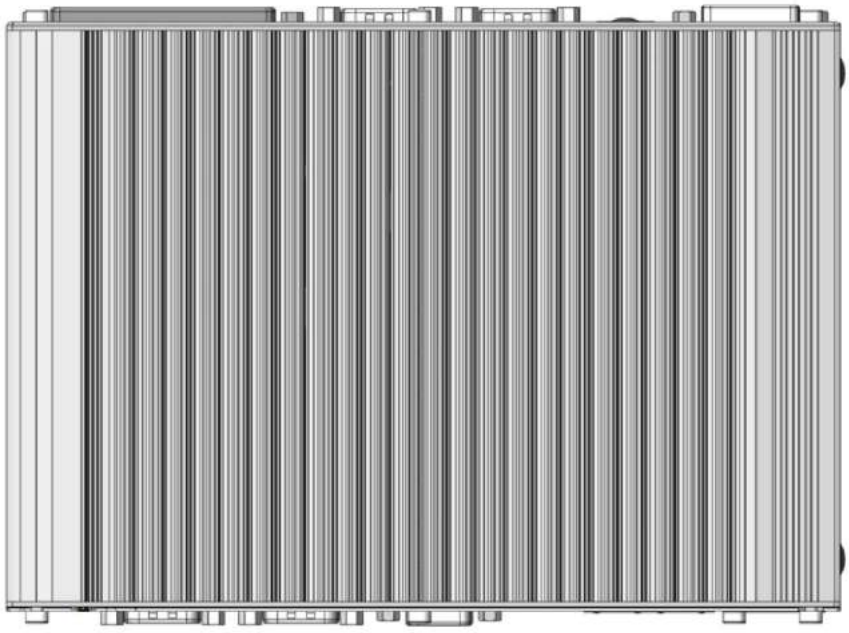


Figure 40: Top View of Nuvis-2520at

Chapter 3 Getting Start

In this chapter, we'll illustrate how to disassemble your Nuvo-2500 Series and install peripheral devices. Please note that the procedures might be varied for different models. Please follow the correct procedures to prevent any damage on your Nuvo-2500 Series

3.1 Disassemble the Cassette

Nuvo-2500 and Nuvis-2520at feature the Expansion Cassette. This patented design makes it possible to install a PCI/PCIe add-on card and lowers the thermal impact to your system resulting from the heat dissipation of the add-on card. To install an add-on card or modules, please release the four M4 screws and follow remove the cassette first as shown in the following picture.

Figure 41: Cassette Screws

3.2 Remove the Back Cover

Nuvo-2500 series has a back cover. After disassembling the cassette if any, the back cover is on the bottom of Nuvo-2500. By removing the back cover, you can see the internal I/O of Nuvo-2500. The figures show the location of back cover screws. Please release the back cover screws in prior to removing the back cover.

Figure 42: Nuvo-2500 and Nuvis-2520at Back Cover Screw

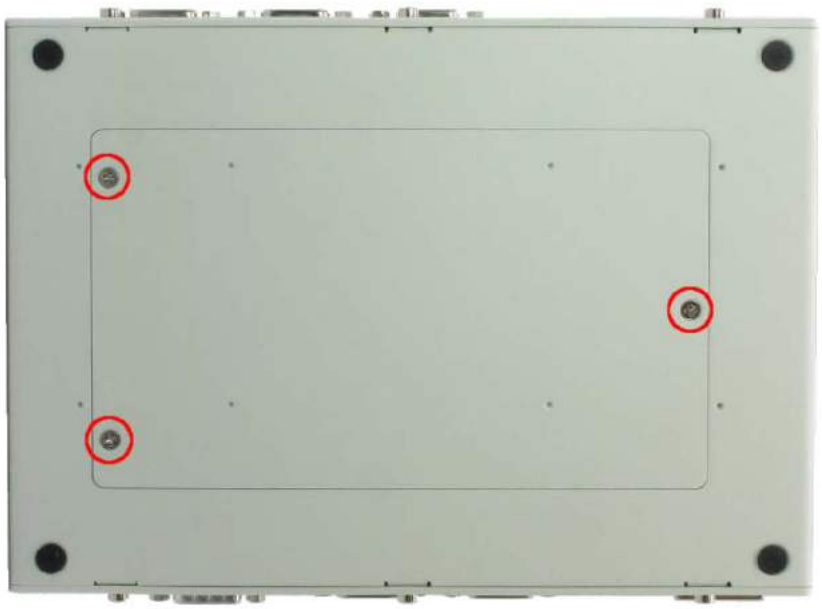


Figure 43: Nuvo-2510VTC Back Cover Screw

3.3 Install an Add-on Card into Cassette

Expansion cassette is available on Nuvo-2500 and Nuvis-2520at. Nuvo-2510VTC doesn't support this functionality. The Expansion Cassette is designed for a PCI Express or PCI add-on card. To install an add-on card into Cassette, please refer to the information listed below.

1. Refer to the section "*Disassemble the Cassette*" to remove the cassette.
2. Release the M3 screw and you can open the cassette.

Figure 44: Cassette Cover Screw

3. Remove the blank PCI Bracket installed in the Cassette by loosening the M3 screw.

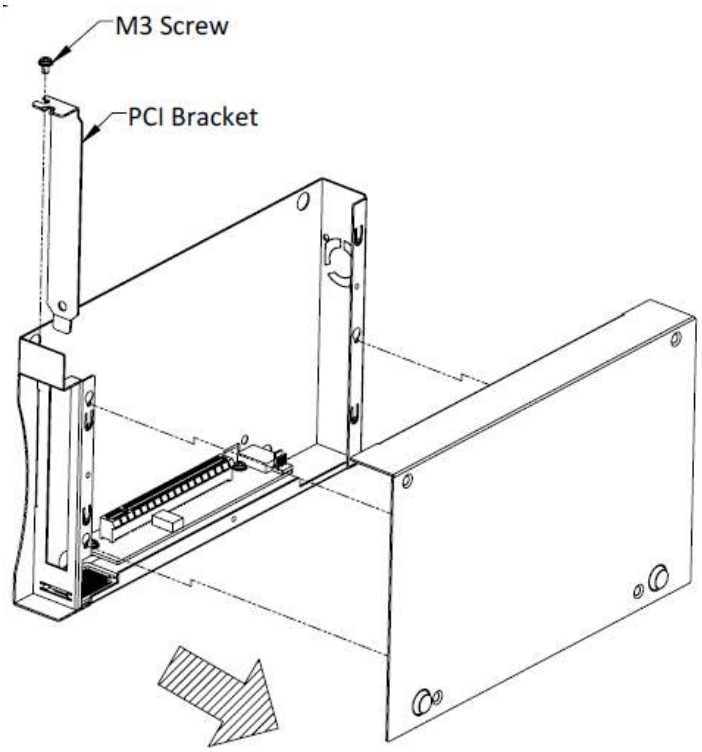


Figure 45: Remove PCI Bracket

4. Install a PCI/PCIe add-on card into the PCI/PCIe connector. Note that the bottom of PCI Bracket of the add-on card must be inserted into the mortise. Tighten the add-on card using a M3 screw.

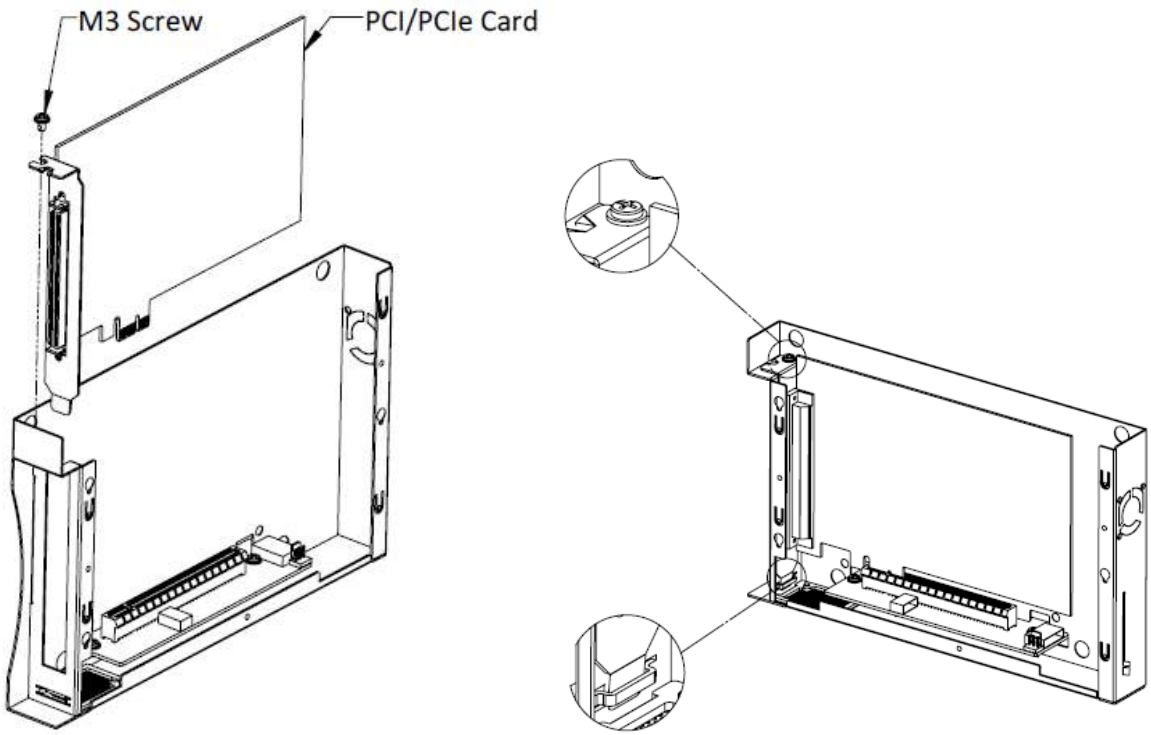


Figure 46: Install an Add-on Card

5. Recover the Cassette and assemble it to your system. Fix Cassette with the four M4 screws.

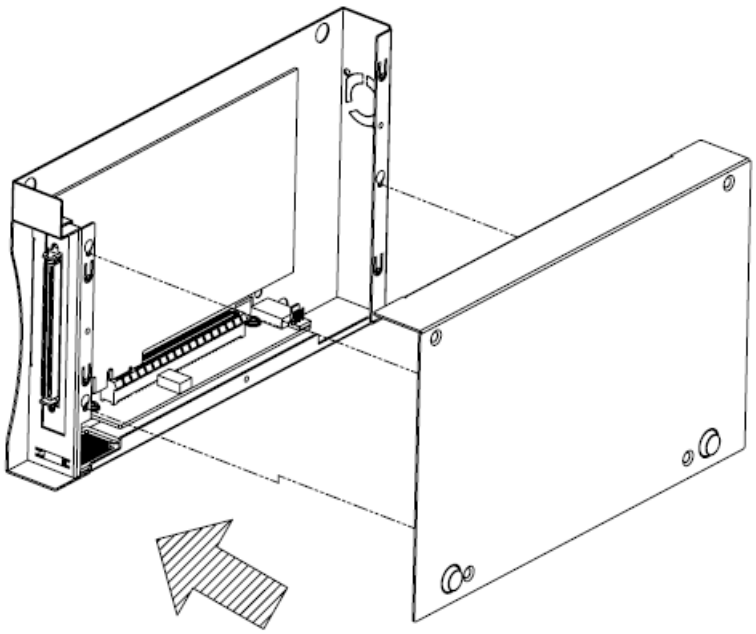


Figure 47: Close Cassette and Assemble Cassette to System

3.4 Install a DDR3L SODIMM Module

Nuvo-2500 Series provides one 204-pin, SODIMM socket for installing 1.35V DDR3L memory module. You can install/replace the memory modules according to the following the steps.

NOTE *Installing incorrect memory module might damage the system or result in system failure. Please make sure you're installing a 1.35V DDR3L SODIMM module.*

1. Refer to the section "*Disassemble the Cassette*" to remove the cassette if any.
2. Refer to the section "*Remove the Back Cover*" to remove the back cover.
3. Find the memory module socket as shown in *Figure 23*.
4. Tile the memory module and insert it to the socket. As it's firmly contacted with socket connectors, press it down till the clamps of the socket snap into the latching position of the memory module.

Figure 48: Install the Memory Module

3.5 Install a 2.5" HDD/SSD

The SATA port of Nuvo-2500 Series is provided via a 22-pin SATA connector. To install a 2.5" HDD/SSD to your Nuvo-2500 Series, please follow the steps listed below.

1. Refer to the section "*Disassemble the Cassette*" to remove the cassette if any.
2. Refer to the section "*Remove the Back Cover*" to remove the back cover. The back cover is also the HDD bracket
3. You can find M3 flat-head screws (4 pieces) and a HDD thermal pad in the accessory box. Remove the films on both sides of the thermal pad and place the pad on the center of HDD bracket

Figure 49: HDD Bracket

4. Place the HDD into the bracket and gently push it down to make it contact with the thermal pad. Use a Philips screwdriver to fix the HDD with M3 flat-head screws. Please note that the HDD must be placed in the right direction as below.

Figure 50: Fasten HDD Screws

5. Pull out the SATA cable inside the chassis and connect it to HDD

Figure 51: Connect HDD to the SATA Cable

6. Tilt the HDD assembly and insert the wedge of HDD bracket to your system. Once it's firmly wedged, push it down and fix it using a M3 flat-head screw.

Figure 52: Install the HDD Bracket with HDD mounted on it

3.6 Install a Mini-PCIe Module

Nuvo-2500 Series provides two Mini PCIe connectors and one mSATA connector. In this section, we'll demonstrate how to install a mini-PCIe WIFI module and attach an antenna to your Nuvo-2500 Series. Please note that the mini-PCIe WIFI module, cable and antenna are not part of Nuvo-2500 Series and may be different according to your system configuration.

1. Refer to the section "*Disassemble the Cassette*" to remove the cassette if any.
2. Refer to the section "*Remove the Back Cover*" to remove the back cover.
3. You can find Mini-PCIe and mSATA connectors as indicated in the figure.

Figure 53: Mini-PCIe and mSATA Connectors

Tilt the mini-PCIe module and insert it to the mini-PCIe port. And fix the module with two M2.5 P-head screws.

Figure 54: Install and Lock the Mini-PCIe Module

4. If the module needs an antenna, attach the IPEX-to-SMA cable to the module and fix the SMA connector (on the IPEX-to-SMA cable) to panel.

Figure 55: Wire the IPEX-to-SMA cable for Antenna

5. Assemble the back cover and fasten the M3 screw, and also assemble the Cassette if any.
6. Attach the WIFI antenna to the SMA connector.

Figure 56: System with an Antenna

3.7 Mount your Nuvo-2500 Series

Nuvo-2500 Series is shipped with a standard wall-mounting bracket. Neousys also offers the option of DIN-rail mounting bracket so that you can mount your system on a DIN rail. Please refer to the information listed below to mount your Nuvo-2500 Series.

NOTE *For the best efficiency of heat dissipation, please ensure your system in a correct orientation, as shown in the following figure, whenever it's vertically mounted.*

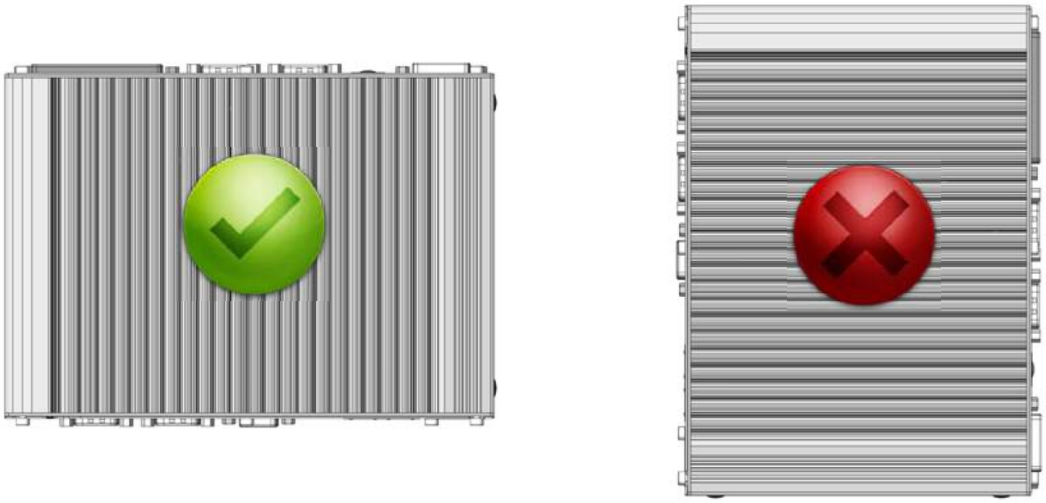


Figure 57: Correct Orientation of Mounting your System Vertically

3.7.1 Mount your Nuvo-2500 and Nuvis-2520at on the Wall

1. Take away the foot pad on the bottom of your system.

Figure 58: Footpad of Nuvo-2500 and Nuvis-2520at

2. Get the wall-mount bracket from your accessory box, and tighten the four M4 pan screws to attach the wall-mount bracket on your system.

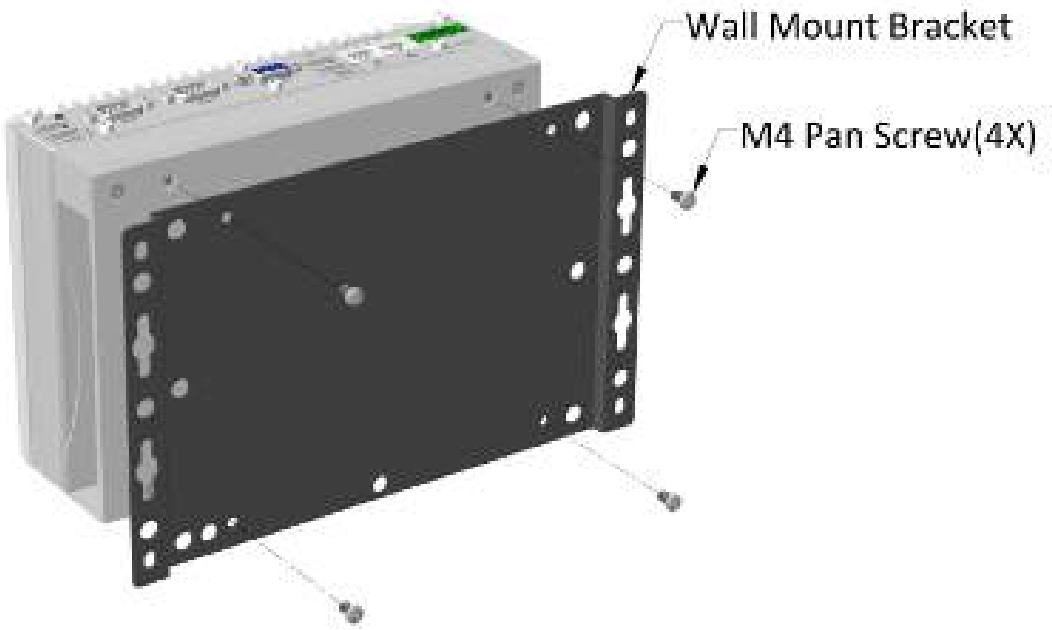


Figure 59: Assemble the Wall-mount Bracket Nuvo-2500 and Nuvis-2520at

3. You can fix the system on a flat surface through the four mount holes with M4 screws. Or alternatively, you can suspend the system on a vertical surface through the four key holes with M4 screws.

3.7.2 Mount your Nuvo-2510VTC on the wall

1. Take away the foot pad on the bottom of your system.

Figure 60: Footpad of Nuvo-2510VTC

2. Get the shock-absorbing wall-mount bracket from your accessory box, and tighten the four M4 shoulder screws to attach the bracket on your system.

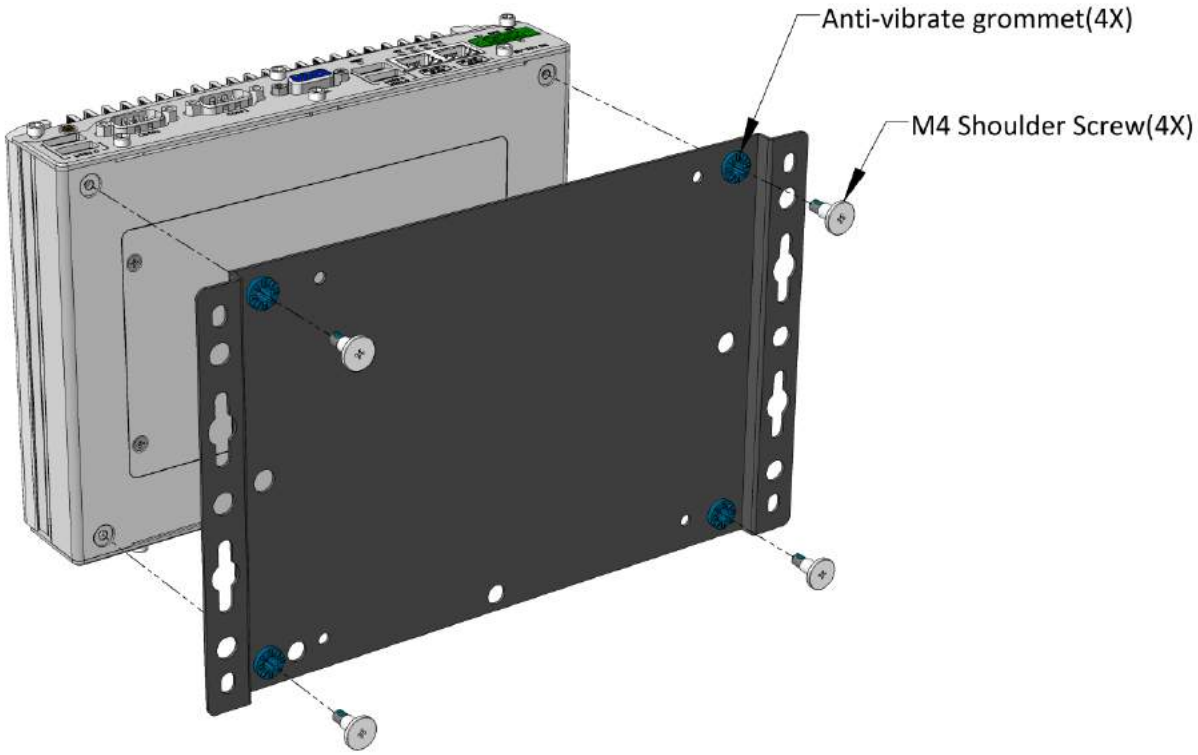


Figure 61: Assemble the Wall-mount Bracket Nuvo-2510VTC

3. You can fix the system on a flat surface through the four mount holes with M4 screws.

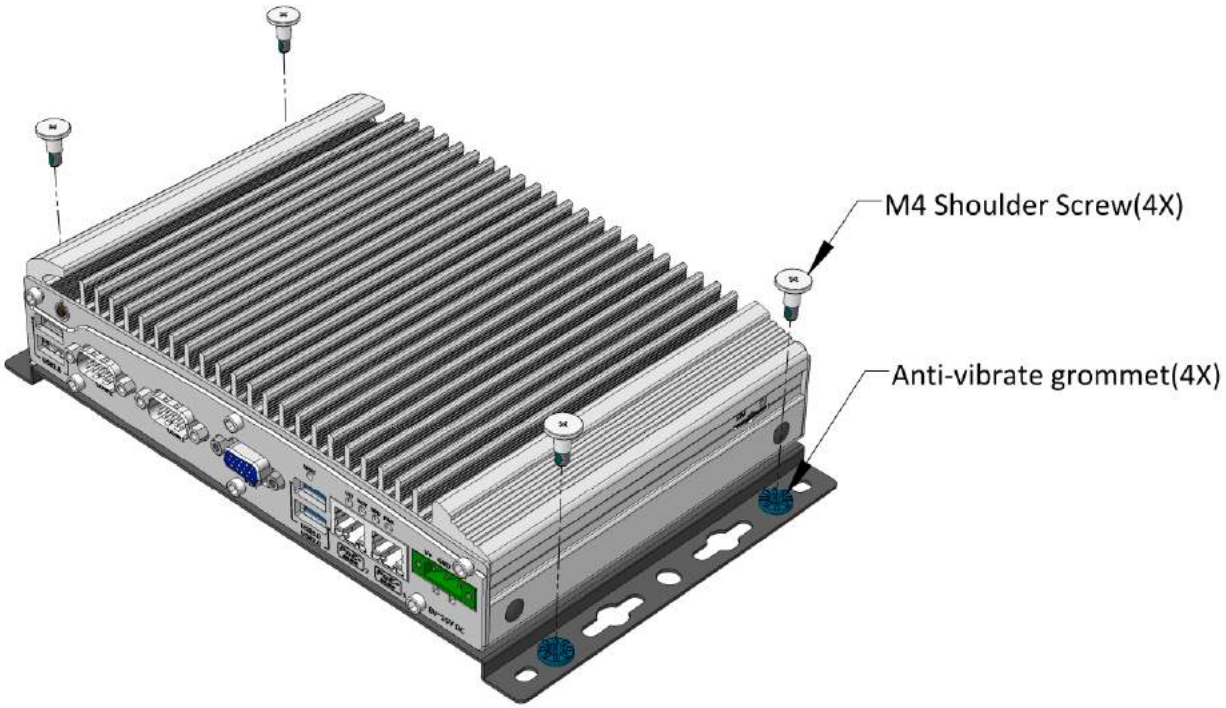


Figure 62: Fix Nuvo-2510VTC on a flat surface

4. Or alternatively, you can suspend the system on a vertical surface through the four key holes with M4 screws.

3.7.3 Mount your Nuvo-2500 and Nuvis-2520at on the DIN Rail

1. Take away the foot pad on the bottom of your system. Refer to *Figure 58*
2. Get the DIN-rail bracket from your accessory box, and tighten the four M4 pan screws to attach the DIN-rail bracket on your system.

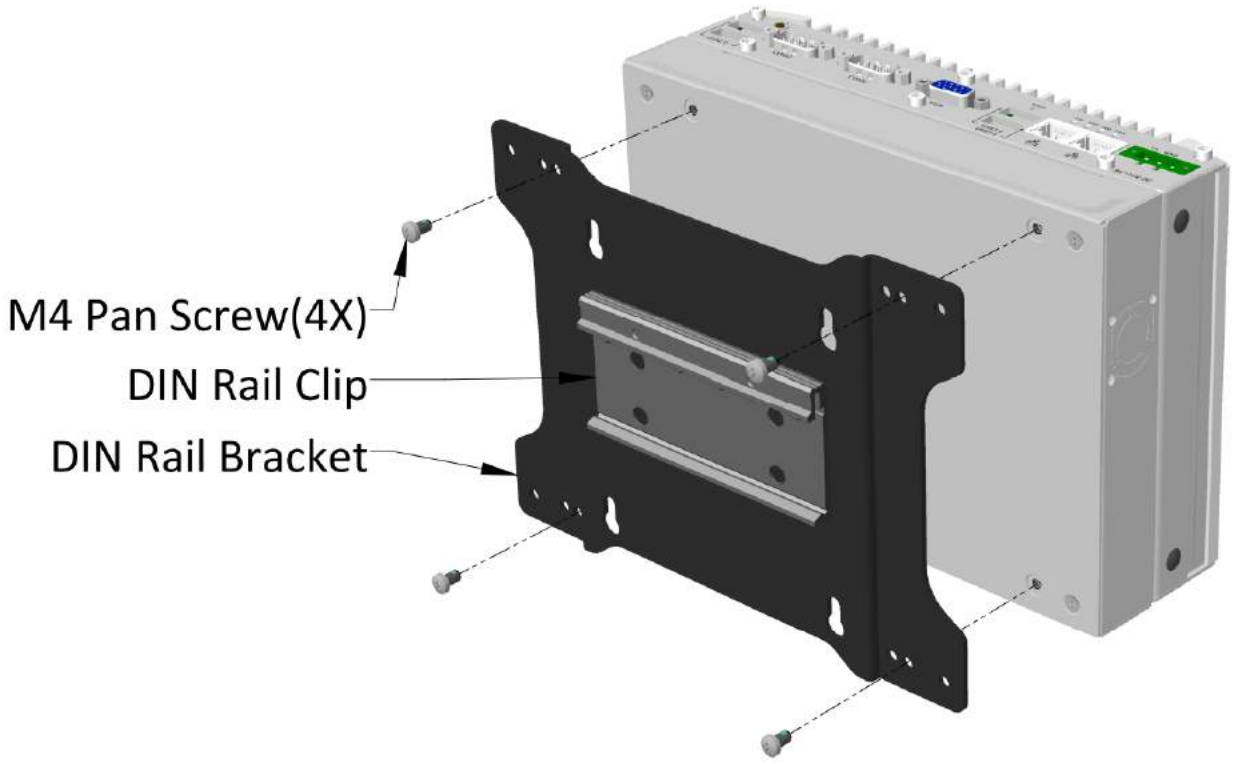


Figure 63: Assemble the DIN-rail Bracket Nuvo-2500 and Nuvis-2520at

3. You can mount the system to DIN rail with the DIN rail clip

3.7.4 Mount your Nuvo-2510VTC on the DIN Rail

1. Take away the foot pad on the bottom of your system. Refer to *Figure 60*
2. Get the DIN-rail bracket from your accessory box, and tighten the four M4 pan screws to attach the DIN-rail bracket on your system.

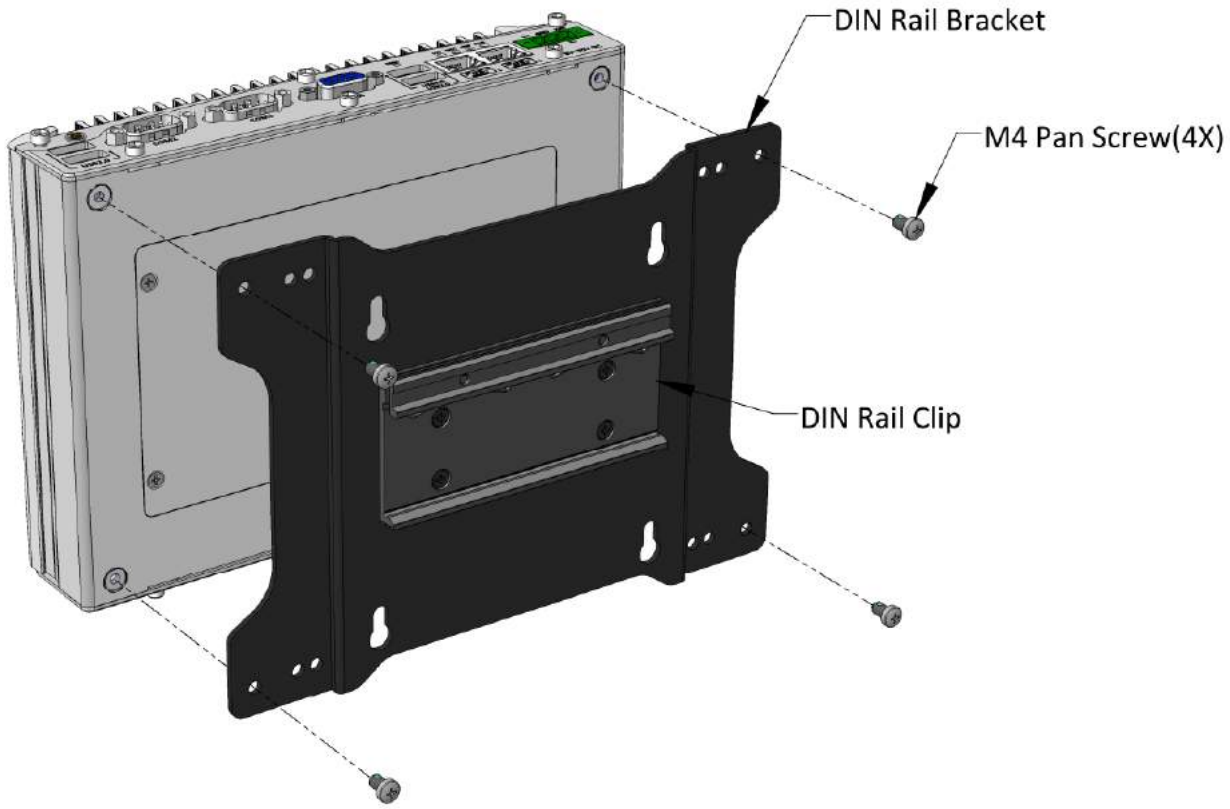


Figure 64: Assemble the DIN-rail Bracket on Nuvo-2510VTC

3. You can mount the system to DIN rail with the DIN rail clip

3.8 Connect DC power to Nuvo-2500 Series

Nuvo-2500 Series uses a 3-pin pluggable terminal block to accept from 8 to 35V power input. It provides the way for directly wiring the DC power. To connect DC power via the 3-pin pluggable terminal block, please follow the steps listed below.

1. Make sure the external DC power supply is power off or disconnected before you wire it to the pluggable terminal block.
2. Get the 3-pin pluggable terminal block from the accessory box. The terminal block fits the wires with a gauge of 12~24 AWG.
3. Carefully identify the positive and negative contacts of your DC power supply and pluggable terminal block. The polarities between DC power supply and terminal block must be positive (+) to positive (+) and negative (-) to negative (-).

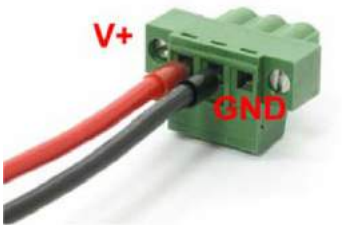


Figure 65: Wiring of DC Input Terminal Block

4. Push the terminal block to the DC input connector till it's firmly attached. Now you can supply the DC power and operate your system.

Figure 66: System with DC Input Terminal Block

3.9 Power on your Nuvo-2500 Series

3.9.1 Power on Using the Power Button

This is the simplest way to turn on your system. The power button is a non-latched switch and behaves the ATX-mode on/off control. As DC power is connected, push the power button will turn on the system as well as the PWR LED indicator. Push the button when system is on will turn off the system. If your operating system supports ATX power mode (i.e. Microsoft Windows or Linux), push the power button causes a pre-defined system behavior, such as shutdown or hibernation.

3.9.2 Power on Using Wake-on-LAN Function

Wake-on-LAN (WOL) is a mechanism to wake up a computer system from a S3 (standby), S4 (Hibernate) or S5 (system off with standby power) state via issuing Subnet Directed Broadcasts (SDB) or a magic packet. Nuvo-2500 Series implements the Wake-on-LAN function for its first GbE port. To enable WOL function and power on your system, please follow the steps listed below.

1. During booting, press F2 to enter BIOS setup utility.
2. Enter the **[Power]** menu. And configure the **[Wake On LAN]** option as **[Enabled]**. This setting enables the Wake-on-LAN function of your system. Please refer to [###] for the instruction of configuring this BIOS option.
3. In Windows 7 system, identify the Local Area Connection of corresponding Intel® I210 Gigabit Controller and click the Configure button.

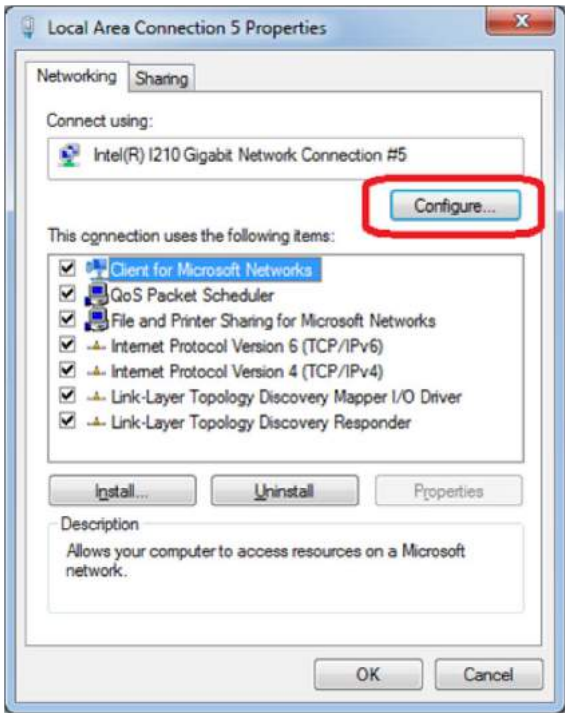


Figure 67: Configure Wake on LAN in Windows 7

4. Click the Power Management tag, and check the following two options respectively

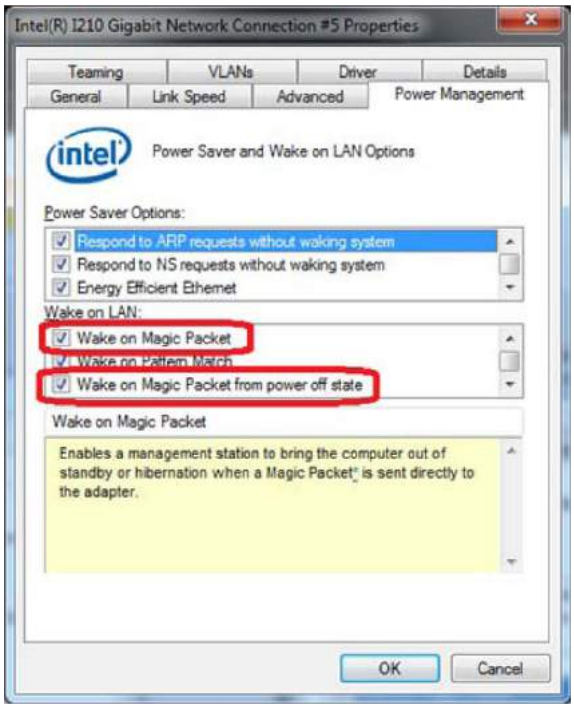


Figure 68: Enable Wake on Magic Packet in Configure Window

Wake on Magic Packet

Checking this option enables your system to wake up from S3 or S4 state upon receiving a magic packet.

Wake on Magic Packet from power off state

Checking this option enables your system to wake up from S5 (system off with standby power) state upon receiving a magic packet.

3.10 Ignition Power Control

Ignition Power Control is available only on Nuvo-2510VTC. The other standard products of Nuvo-2500 Series don't support this functionality. Ignition power control is designed for in-vehicle applications. It's a MCU-based implementation that monitors the ignition signal and reacts to turn on/off the system according to predefined on/off delay. Its built-in algorithm supports further features such as ultra-low standby power, low battery protection, system hard-off and etc. In this section, we'll illustrate the principle of ignition power control and operations modes on Nuvo-2510VTC.

3.10.1 Principle of Ignition Power Control

The basic concept of ignition power control module is to control the timing correlation between ignition signal and system power status. A typical timing correlation can be described in following diagram.

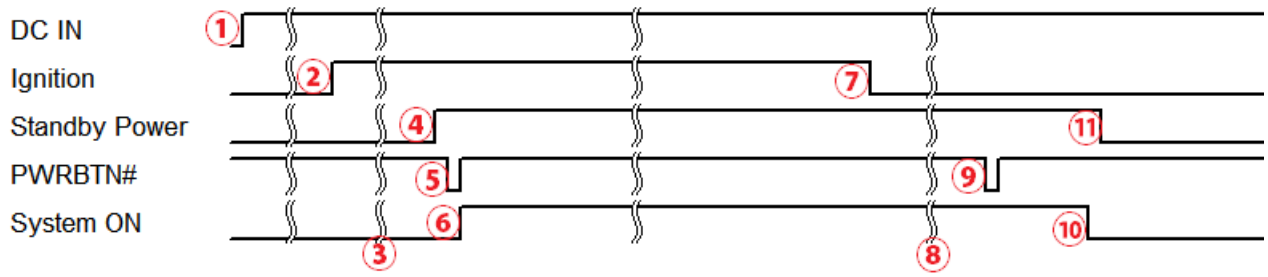


Figure 69: Timing Diagram of Ignition Power Control

1. When DC input is supplied to Nuvo-2510VTC, MCU starts to periodically detect ignition signal. Note that only MCU is working at this moment and the overall power consumption is less than 2 mW.
2. Ignition signal is active. (Both 12VDC and 24VDC ignition signals are accepted)
3. MCU starts to count a pre-defined power-on delay.
4. Once power-on delay expired, MCU turns on necessary standby power for Nuvo-2510VTC (3.3VSB & 5VSB).
5. A PWRBTN# pulse is then issued to turn on the system (a similar behavior as you press the power button on the front panel).
6. Nuvo-2510VTC is booting up and running.
7. After a period of time, the ignition signal is inactive.
8. MCU starts to count a pre-defined power-off delay.
9. Once power-off delay expired, another PWRBTN# pulse is issued to perform a soft-off

for the system (e.g. a normal shutdown process for Windows system).

10. Nuvo-2510VTC is completely shut down.
11. As MCU detects system is off, it turns off the standby power for Nuvo-2510VTC, and then operates in low power mode again (< 2mW power consumption).

In addition to the typical timing correlation, the ignition power control module offers some further features to make Nuvo-2510VTC more reliable for in-vehicle applications.

Low battery detection

The ignition power control module is capable to continuously monitor the voltage of DC input while system is running. If the detected input voltage meets the low battery voltage criteria for over 60 seconds, it will shut down the system automatically. In mode 2 to 6, the shutdown process will be initiated if the input voltage is less than 9V or keeps in the range between 16V and 18V for over 60 seconds.

Guarded power-on/power-off delay duration

If ignition signal goes inactive during the power-on delay duration, the ignition power control module will cancel the power-on delay process and go back to idle status. Likewise if ignition signal goes active during the power-off delay duration, the ignition power control module will cancel the power-off delay process and keep the system running.

System hard-off

In some cases, system may be failed to normally shutdown via a soft-off operation due to system/application halts. The ignition power control module on Nuvo-2510VTC offers a mechanism called "hard-off" to handle this unexpected condition. By detecting the system status, it can determine whether the system is normally shutdown. If not, the ignition power control module will compulsively cut off the system power 10 minutes after the power-off delay duration.

Smart off-delay

The ignition power control module on Nuvo-2510VTC offers mode 6 which have very long power-off delay duration for applications require some off-line processing after vehicle is stopped. In these two modes, the ignition power control module will automatically detect the system status during the power-off delay duration. If the system is shutdown (by the application software) in prior to power-off delay expired, it will cut off the system power immediately to prevent further consumption of battery power.

Parameters setting in BIOS

In addition to quick setup and pre-defined modes with different power on / off delay and low voltage detection, a full configuration mode is also available when the ignition rotary switch is positioned at 7. This mode allows users to set and save preferable data in BIOS. This full configurable mode provides an extremely flexible usage of ignition power control.

3.10.2 Operation Modes of Ignition Power Control

You can use the rotary switch on SBC to configure the operation mode. Nuvo-2510VTC offers 7 operation modes with different power-on/power-off delay configurations. By switching the Ignition Switch shown in *Figure 25*, you can set different mode of ignition power control.

NOTE *If you'd like to completely disable the ignition power control, please set the ignition switch to 8. In this case, you can power on / off your Nuvo-2510VTC only by pressing the power button regardless of the power ignition input.*

Mode 0

Mode 0 is the ATX mode without power-on and power-off delay. User can only use the power button on the front panel to turn on or turn off Nuvo-2510VTC.

Mode	Power-on Delay	Power-off Delay	Hard-off Timeout
0	N/A	N/A	N/A

Mode 1

If Mode 1 is specified, the system automatically turns on the system when DC power is applied. A retry mechanism is designed to repeat the power-on cycle if the system is failed to boot up.

Mode	Power-on Delay	Power-off Delay	Hard-off Timeout
1	N/A	N/A	N/A

Mode 2 / Mode 3 / Mode 4 / Mode 5

Mode 2 ~ Mode 5 are ignition power control modes with various power-on delay and power-off delay. Each mode supports a hard-off timeout of 10 minutes.

Mode	Power-on Delay	Power-off Delay	Hard-off Timeout
2	10 seconds	10 seconds	10 minutes
3	30 seconds	1 minute	10 minutes

4	30 seconds	5 minutes	10 minutes
5	30 seconds	30 minutes	10 minutes

Mode 6

Mode 6 is ignition power control modes with very long power-off delay. Mode 6 also support the feature of "smart off-delay", which automatically detect system status during power-off delay duration and cut off system power if system is off in prior to power-off delay expired.

Mode	Power-on Delay	Power-off Delay	Hard-off Timeout
6	30 seconds	2 hours	10 minutes

Mode 7

When the rotary switch is set to 7, users can configurable parameters in BIOS setup menu.

To configure ignition parameters in BIOS, please follow the steps below.

1. Make sure you have set the rotary switch to position 7.
2. When system boots up, press F2 to enter BIOS setup menu.
3. Go to **[Power]** and then **[Ignition Power Control]**.

Figure 70: BIOS Menu of Ignition Power Control

4. Configure parameters for ignition power control according to your application. The definition of parameters is described in the end of this section.
5. Press F10 to save configured parameter. The ignition control module will be reset and then operate according to parameters configured in BIOS setup menu.

IGN Operation Mode

Value	Description
ATX	ATX mode without power-on and power-off delay. Same operation as rotary switch set to 0.
AUTO-ON	Automatically turns on the system when DC power is applied. Same operation as rotary switch set to 1.
IGN	Ignition power control mode. Ignition control is executed according to the specified parameters.

Smart Off-Delay

Value	Description
Enabled	If system is manually shutdown during the power-off delay period, ignition control module will cut off system power in prior to expiration of power-off delay to save battery power.
Disabled	Ignition control module cut off system power only after power-off delay expired.

BIOS POST Check

Value	Description
Enabled	This option secures a BIOS POST operation. If the system is failed to complete the POST or failed to find a boot device (e.g. disk failure or no bootable device) within 60 seconds, ignition control module will cut off system power and retry another power on cycle.
Disabled	BIOS POST check is skipped.

Power On Delay

Specify the power-on delay duration. Once IGN signal goes active and sustains for the duration of power-on delay, ignition control module turns on system power and boot up the system.

Power Off Delay

Specify the power-off delay duration. Once IGN signal goes inactive and sustains for the duration of power-off delay, ignition control module performs system shutdown (soft-off) and then cut off system power.

Hard-off Timeout

Specify system hard-off timeout. Once system failed to normally shutdown via a soft-off operation due to system/application halts (e.g. Windows BSOD), ignition control module

can compulsively cut off system power after the given hard-off timeout.

Battery Voltage

Specify the battery voltage of the vehicle where your system is deployed. Typically it's 12 VDC for sedan and 24 VDC for bus/truck.

Low Battery Threshold

When system is running, ignition control module continuously monitors the battery voltage. Once the battery voltage is lower than the specified threshold, it performs system shutdown (soft-off) and cut off system power to prevent battery drain-out. You should specify the low battery threshold according to the given battery voltage.

Chapter 4 BIOS and Driver

4.1 BIOS Settings

Nuvo-2500 Series is shipped with factory-default BIOS settings cautiously programmed for best performance and compatibility. In this section, we'll illustrate some of BIOS settings you may need to modify. Please always make sure you understand the effect of change before you proceed with any modification.

To Enter BIOS setup:

When Nuvo-2500 is booting up, press F2 to enter BIOS setup utility. Use following keys to edit or change BIOS options.

Table 26: Function Keys in BIOS

Keys	Function
F1	Help
↑↓←→	Select Item
F5/F6	Change Values
F9	Load Setup Defaults
Esc	Exit
Enter	Select -> SubMenu
F10	Save and Exit

4.1.1 COM1 & COM2 Operation Mode

COM1 and COM2 of Nuvo-2500 Series support RS-232 (full-duplex), RS-422 (full-duplex) and RS-485 (half-duplex) mode. You can set the operating mode via BIOS settings. Another option in BIOS called **[Slew Rate]** defines how sharp the rising/falling edge is for the output signal of COM1 and COM2. For long-distance RS-422/485 transmission, you may set the **[Slew Rate]** option as **[High]** to improve signal quality. For RS-422/485 communication, the **[RS-422/485 Termination]** option determines whether to enable/disable internal termination of RS-422/485 transceiver according to your wiring configuration (e.g. with or without external termination).

To set COM1 and COM2 operating mode:

1. When Nuvo-2500 boots up, press F2 to enter BIOS setup utility.
2. Go to **[Advanced] / [Peripheral Configuration]**
3. Set the **[Set COM1 as]** option to a proper mode for COM1 of your Nuvo-2500.

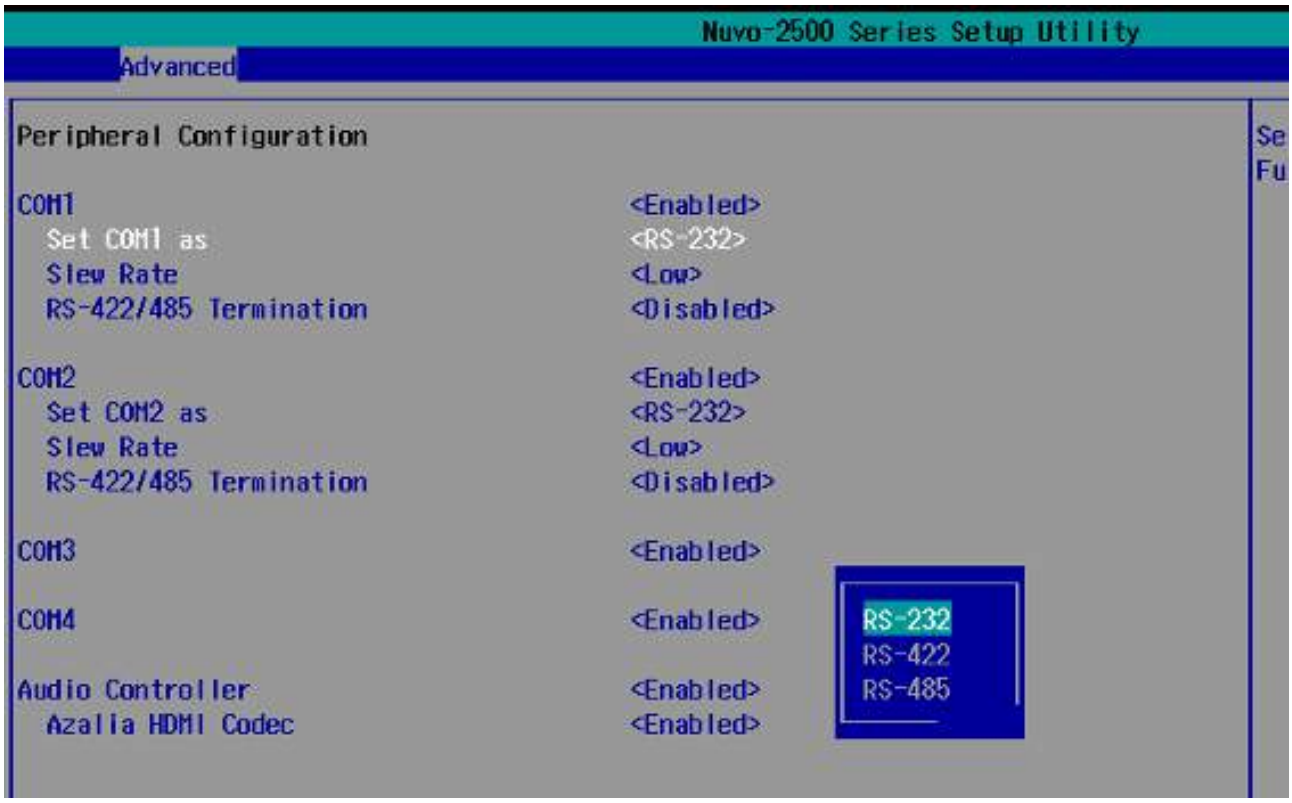


Figure 71: Setting Modes of COM Port in BIOS

4.1.2 Chipset SATA Mode

The SATA controller of Nuvo-2500 Series supports two modes of operations, IDE and AHCI mode. IDE mode configures SATA controller to access SATA interface in legacy IDE mode and is compatible with most storage devices. AHCI mode, which exposes SATA's advanced capabilities such as hot swapping and native command queuing, can deliver better performance for disk read/write. As Nuvo-2500 Series no longer supports Windows XP, we highly recommend you to set **[Chipset SATA Mode]** to **[AHCI]** for better system performance.

To set **[Chipset SATA Mode]**:

1. When Nuvo-2500 boots up, press F2 to enter BIOS setup utility.
2. Go to **[Advanced] / [SATA Configuration]**.
3. Set the **[Chipset SATA Mode]** option to a proper mode for your Nuvo-2500.

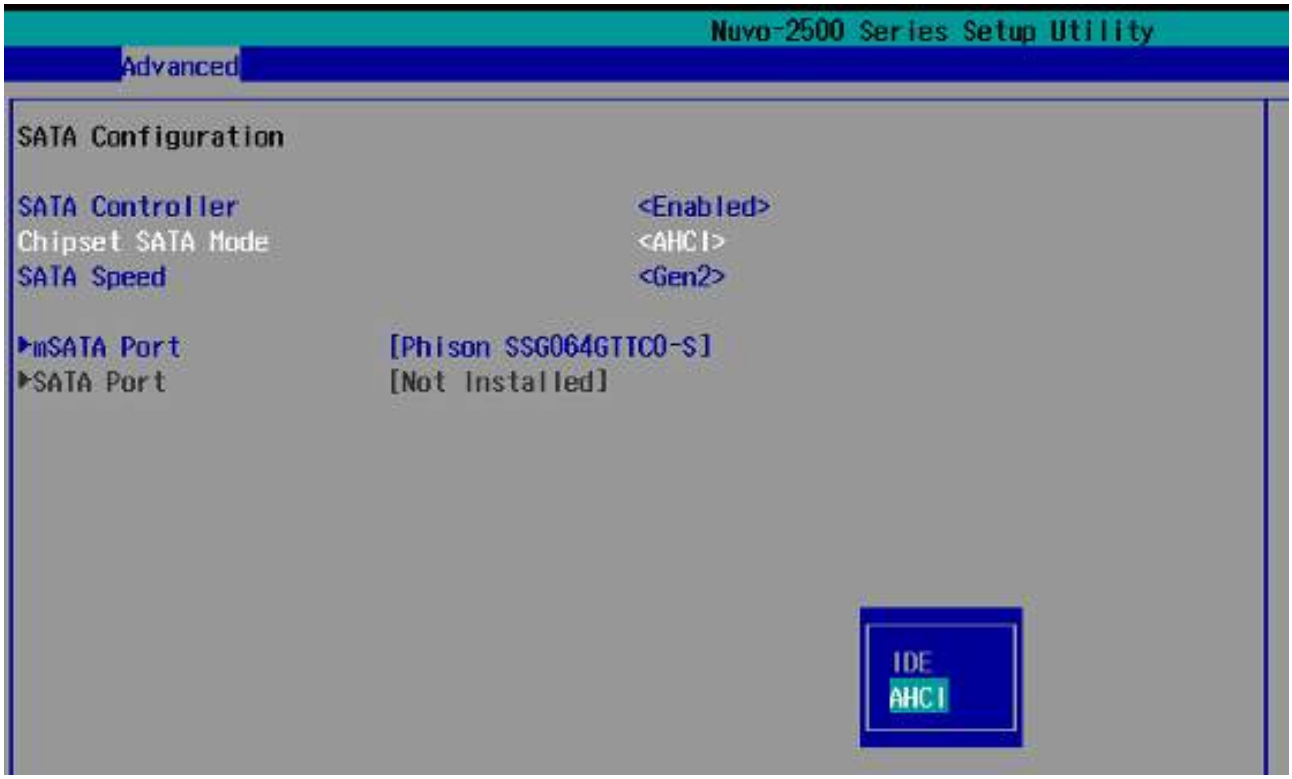


Figure 72: Setting Modes of SATA Mode in BIOS

4.1.3 C-States and Max C-States

C-States is a power-saving technique implemented in modern Intel processors. It shuts down the clock signals and power for idle logic units inside the CPU to save the energy consumed. The trade-off, however, is a longer latency for CPU to wake up and be 100% operational. Depending on your application, you can configure these options to have higher performance (disable "C-States") or lower power-consumption (enable "C-States" and set "Max C-States" to C6/C7).

To set C-States and Enhanced C-States:

1. When Nuvo-2500 boots up, press F2 to enter BIOS setup utility.
2. Go to **[Power] / [Advanced CPU Control]**.
3. Enable/disable the **[C-States]** option according to your application.
4. Configure the **[Max C-States]** option according to your application.

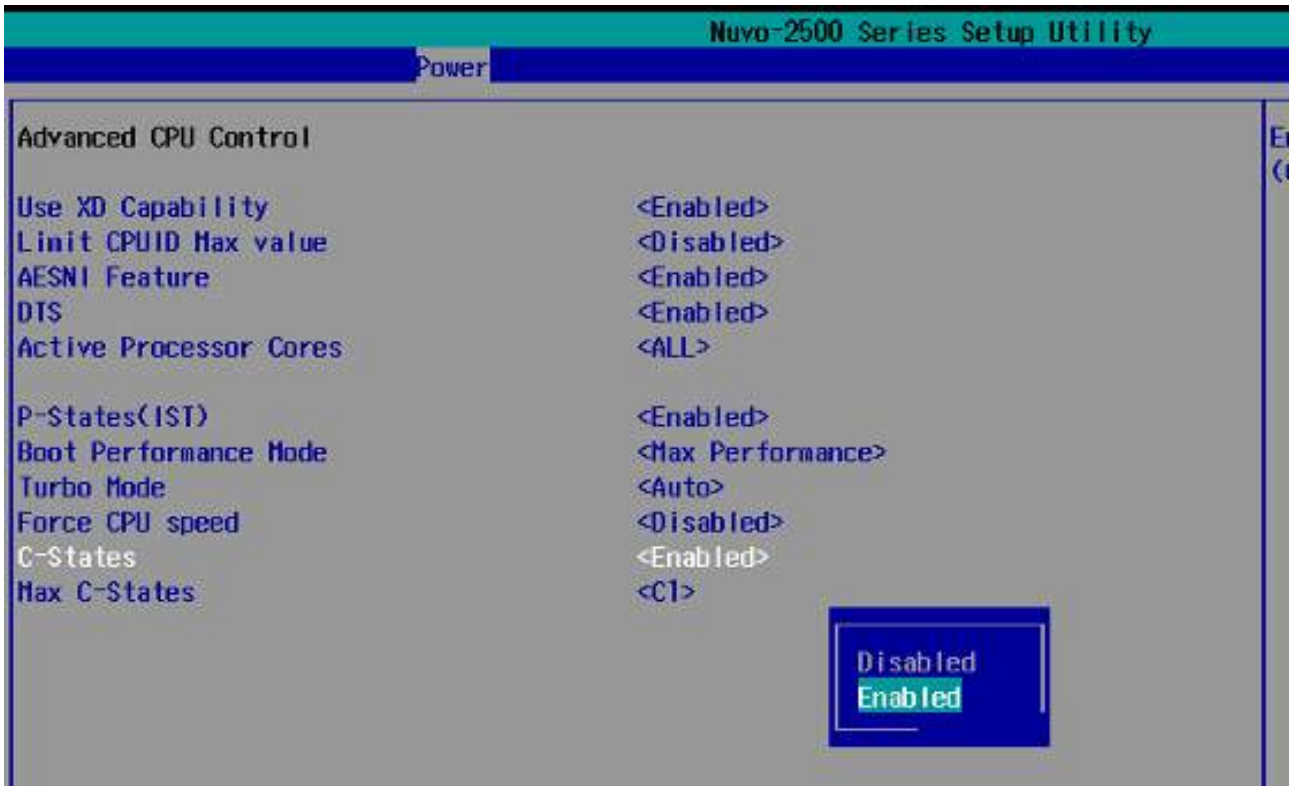


Figure 73: Configuring C-States in BIOS

4.1.4 Wake-on-LAN Option

Wake-on-LAN (WoL) is a mechanism which allows you to turn on your Nuvo-2500 via Ethernet connection. To utilize Wake-on-LAN function, you have to enable this option first in BIOS settings. Please refer to the section "*Power on Using Wake-on-LAN Function*" for instructions of using WoL function.

To enable/disable Wake on LAN option:

1. When Nuvo-2500 boots up, press F2 to enter BIOS setup utility.
2. Enable/disable the **[Wake on LAN]** option according to your application.

Figure 74: Enabling Wake on LAN in BIOS

4.1.5 Power On after Power Failure Option

This option defines the behavior of Nuvo-2500 when DC power is supplied.

To set **[Power On after Power Failure]** option:

1. When Nuvo-2500 boots up, press F2 to enter BIOS setup utility.
2. Go to **[Power]**.
3. Set the **[Power On after Power Failure]** option to a proper value for your Nuvo-2500.

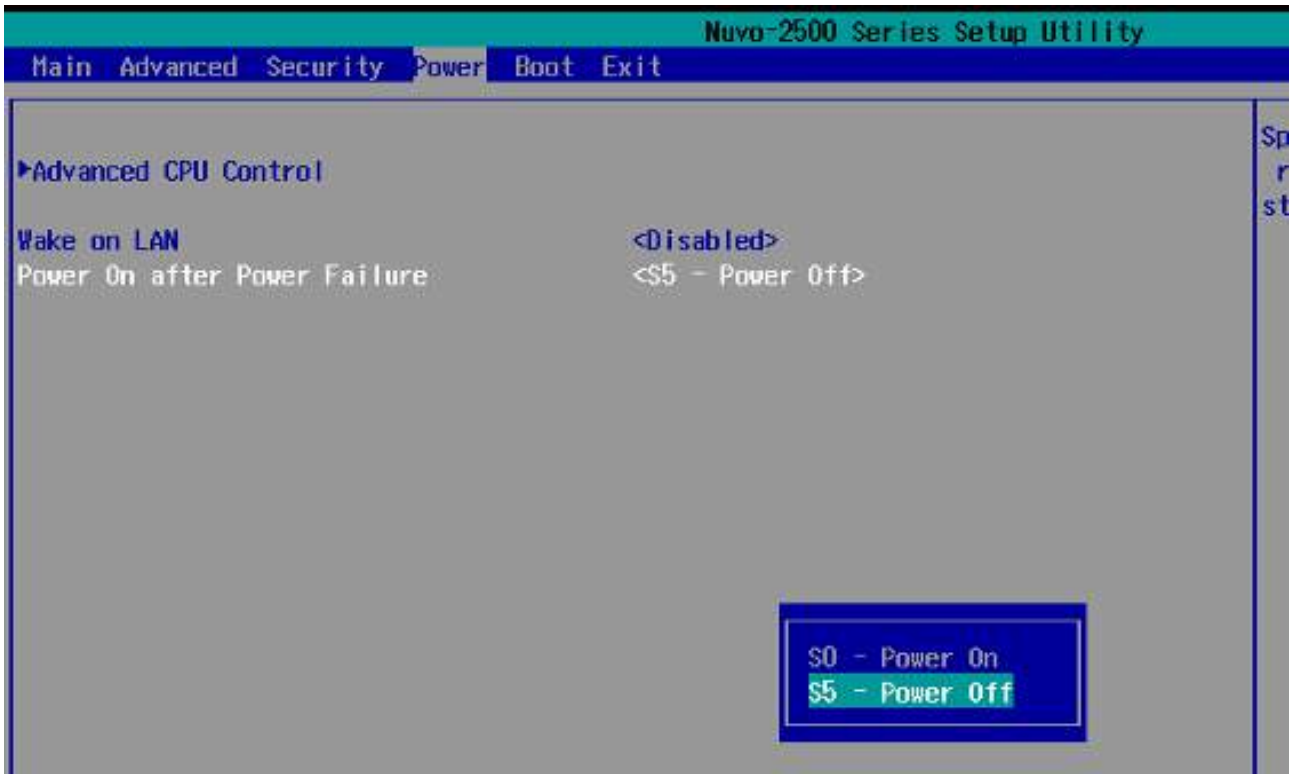


Figure 75: Configuring Power on after Power Failure in BIOS

The following table describes the options of this setting.

Table 27: Options of Power On after Power Failure

Value	Description
S0 – Power On	System is powered on when DC power is supplied.
S5 – Power Off	System is kept in off state when DC power is supplied.

4.1.6 Watchdog Timer for Booting

The BIOS of Nuvo-2500 Series has a useful feature which allows users to use the watchdog timer to secure the booting process. You can specify the timeout value for watchdog timer. Once the watchdog timer expires, the BIOS issues a reset command to initiate another booting process. You can also set the behavior of how to stop the watchdog timer. There are two options in BIOS menu, **[Automatically after POST]** and **[Manually after Entering OS]**. When **[Automatically after POST]** is selected, the BIOS automatically stop the watchdog timer after POST (Power-On Self Test) OK. When **[Manually after Entering OS]** is selected, user applications have to stop the watchdog timer after OS is activated. This guarantees the system can always boot to OS, otherwise another booting process will be initiated. For information about programming watchdog timer, please refer to the chapter "*Using Watchdog Timer and DIO*".

To set the watchdog timer for boot in BIOS:

1. When Nuvo-2500 boots up, press F2 to enter BIOS setup utility.
2. Go to **[Boot]** menu.
3. Disable or select timeout value for **[WDT for Booting]** option.
4. Once you give a timeout value, the **[WDT Stop Option]** option appears. You can select either **[Automatically after POST]** or **[Manually after Entering OS]**.

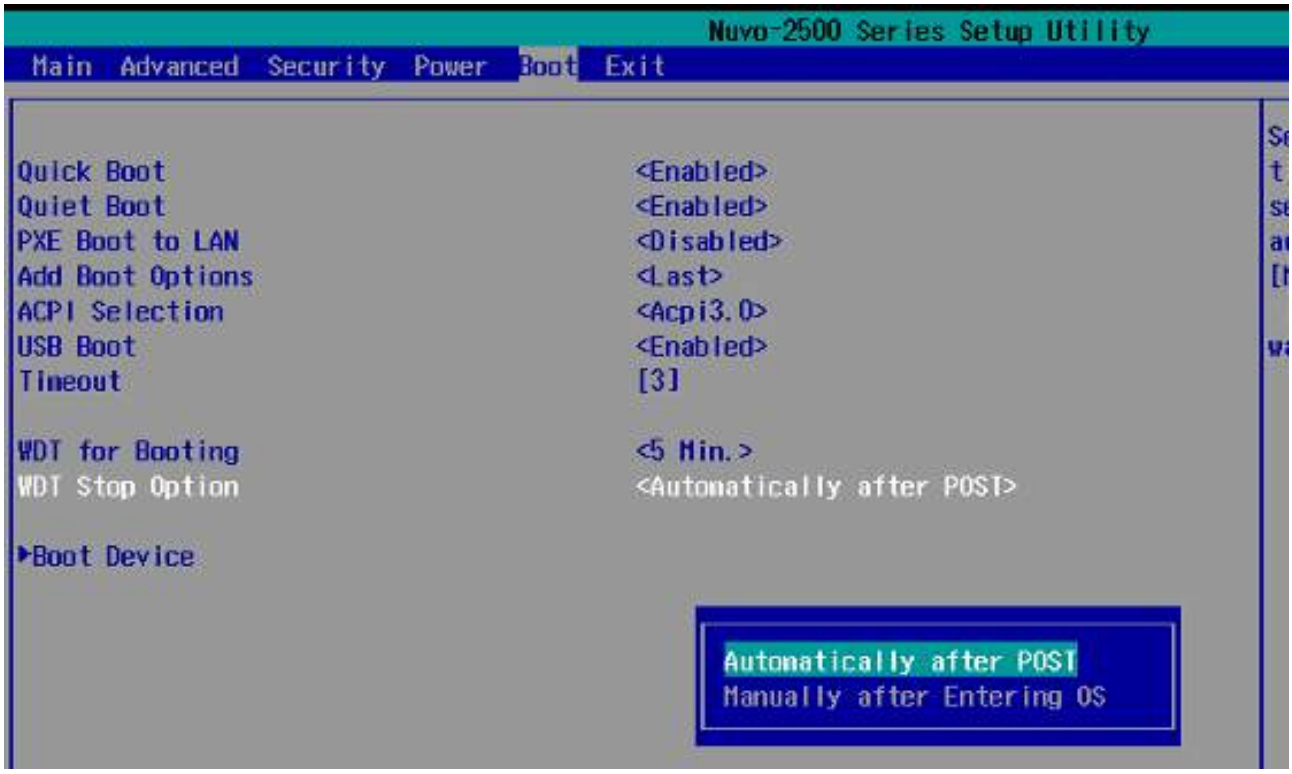


Figure 76: Configuring WDT in BIOS

4.1.7 Select a Boot Device

When you have multiple bootable devices, i.e. HDD, USB flash disk, USB DVD-drive, etc., connected to your Nuvo-2500, you may need to select one of them as the first boot device. There are two ways to select the device. You can either, press F12 when system boots up to go to Boot Manager and then select one of the devices, or select the boot device in BIOS settings.

To select a boot device in BIOS:

1. When Nuvo-2500 boots up, press F2 to enter BIOS setup utility.
2. Go to **[Boot] / [Boot Device]**.
3. The **[Boot Menu]** option determines whether to list all bootable devices connected to your Nuvo-2500 according to device or device type. The default order of boot device type is

- Hard Disk Drive
- USB
- CD/DVD-ROM Driver
- Floppy Drive
- Others

You can use F5/F6 or +/- to change the boot order of devices or device types.

Figure 77: Configuring order of booting devices in BIOS

4.2 Operating System Support

Nuvo-2500 Series supports most operating system developed for Intel® x86 architecture. The following list contains the operating systems which have been tested in Neousys Technology Inc.

- Microsoft Window 7 32-bit
- Microsoft Window 7 64-bit
- Microsoft Window 8/8.1 32-bit
- Microsoft Window 8/8.1 64-bit
- Ubuntu 14.04 or later version

NOTE *For Linux system, user may need to manually compile and install the driver for Intel I210 GbE controller if the driver is not embedded in kernel. You can visit Intel website for further information.*

Neousys will keep this list updated as we continuously test other operating systems with Nuvo-2500 Series. Please contact us for the latest OS support list.

4.3 Driver Installation

Neousys Technology Inc. provides a very convenient utility in "Drivers & Utilities DVD" to allow the "One-Click" driver installation. This utility automatically detects your Windows operating system and installs all necessary drivers to your Nuvo-2500 Series with just one mouse click.

4.3.1 Install All Drivers Using "One-Click" Driver Installation

1. Insert the "Drivers & Utilities DVD" into a USB DVD-drive attached to your Nuvo-2500. A setup utility launches and the following dialog appears.

Figure 78: Neousys Driver & Utilities DVD

2. Click on the **[Automatic Driver Installation].** The setup utility will automatically detect your Windows operating system and install all necessary drivers. According to different versions of Windows, the installation process takes about 6~8 minutes. Once driver installation is done, the setup utility reboots your Windows and your system works normally afterward.

4.3.2 Install Drivers Manually

You can also manually install each driver for Nuvo-2500 Series. Please refer to the following information about installing drivers for different operating system.

Windows 7 32-bit

The recommended driver installation sequence is

1. Chipset driver (x:\Driver_Pool\Chipset_Vlv\Win7_8_ALL\SetupChipset.exe)
2. Graphics driver (x:\ Driver_Pool\Graphics_Vlv_EMGD\Win7_32\Setup.exe)
3. Audio driver (x:\Driver_Pool\Audio_ALC262\Win7_8_ALL\Setup.exe)
4. LAN driver (x:\Driver_Pool\GbE_I210\Win7_8_32\APPS\PROSETDX\Win32\DxSetup.exe)
5. USB 3.0 driver (x:\USB3_Vlv\Win7_ALL\Setup.exe)
6. Embedded I/O (x:\Driver_Pool\IO_Vlv\Win7_32\Intel_Processor_Win7_IO_Drivers_32Bit.msi)

Windows 7 64-bit

The recommended driver installation sequence is

1. Chipset driver (x:\Driver_Pool\Chipset_Vlv\Win7_8_ALL\SetupChipset.exe)
2. Graphics driver (x:\Driver_Pool\Graphics_Vlv_EMGD\Win7_64\Setup.exe)
3. Audio driver (x:\Driver_Pool\Audio_ALC262\Win7_8_ALL\Setup.exe)
4. LAN driver (x:\Driver_Pool\GbE_I210\Win7_8_64\APPS\PROSETDX\Winx64\DxSetup.exe)
5. USB 3.0 driver (x:\Driver_Pool\USB3_Vlv\Win7_ALL\Setup.exe)
6. Embedded I/O (x:\Driver_Pool\IO_Vlv\Win7_64\Intel_Processor_Win7_IO_Drivers_64Bit.msi)

Windows 8/8.1 32-bit

The recommended driver installation sequence is

1. Chipset driver (x:\Driver_Pool\Chipset_Vlv\Win7_8_ALL\SetupChipset.exe)
2. Graphics driver (x:\Driver_Pool\Graphics_3rd_i7_Vlv\Win7_8_32\Setup.exe)
3. Audio driver (x:\Driver_Pool\Audio_ALC262\Win7_8_ALL\Setup.exe)
4. LAN driver (x:\Driver_Pool\GbE_I210\Win7_8_32\APPS\PROSETDX\Win32\DxSetup.exe)
5. TXE driver (x:\Driver_Pool\TXE_Vlv\Win8_ALL\SetupTXE.exe)
6. Embedded IO (x:\Driver_Pool\IO_Vlv\Win8_32\Setup.exe)

Windows 8/8.1 64-bit

The recommended driver installation sequence is

1. Chipset driver (x:\Driver_Pool\Chipset_Vlv\Win7_8_ALL\SetupChipset.exe)
2. Graphics driver (x:\Driver_Pool\Graphics_3rd_i7_Vlv\Win7_8_64\Setup.exe)
3. Audio driver (x:\Driver_Pool\Audio_ALC262\Win7_8_ALL\Setup.exe)
4. LAN driver (x:\Driver_Pool\GbE_I210\Win7_8_64\APPS\PROSETDX\Winx64\DxSetup.exe)
5. TXE driver (x:\Driver_Pool\TXE_Vlv\Win8_ALL\SetupTXE.exe)
6. Embedded IO (x:\Driver_Pool\IO_Vlv\Win8_64\Setup.exe)

Chapter 5 Using Watchdog Timer and DIO

5.1 Install WDT and DIO Library

The WDT_DIO function library is delivered in the form of a setup package named **WDT_DIO_Setup.exe**. In prior to program WDT and DIO on Nuvo-2500, you should execute the setup program and install the WDT and DIO library. The following procedure shows steps to complete the installation of WDT and DIO Library.

1. Execute **WDT_DIO_Setup.exe**. The following dialog appears.

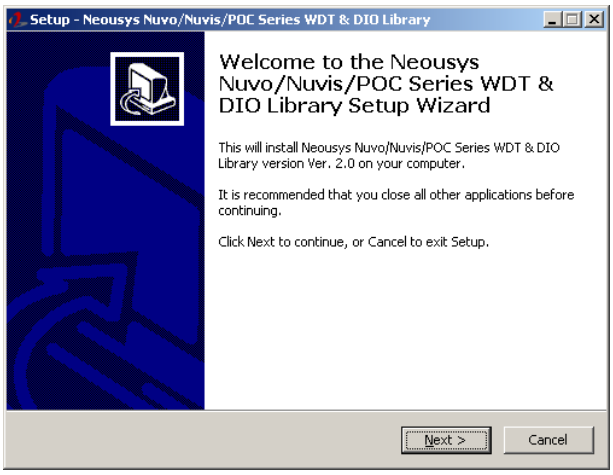


Figure 79: Welcome Window of WDT and DIO Library Install Program

2. Click "Next >" and specify the directory of installing related files. The default directory is C:\Neousys\WDT_DIO.

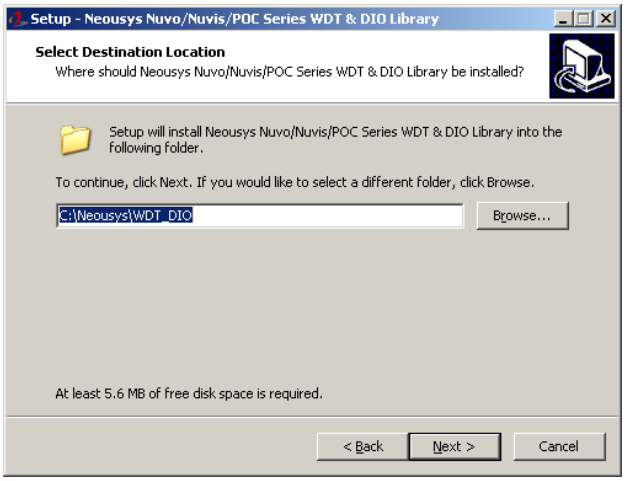


Figure 80: Set Installation Folder of WDT and DIO Library

3. Once the installation is finished, a dialog appears to prompt you to reboot the system. The WDT & DIO library will take effect after system rebooting.

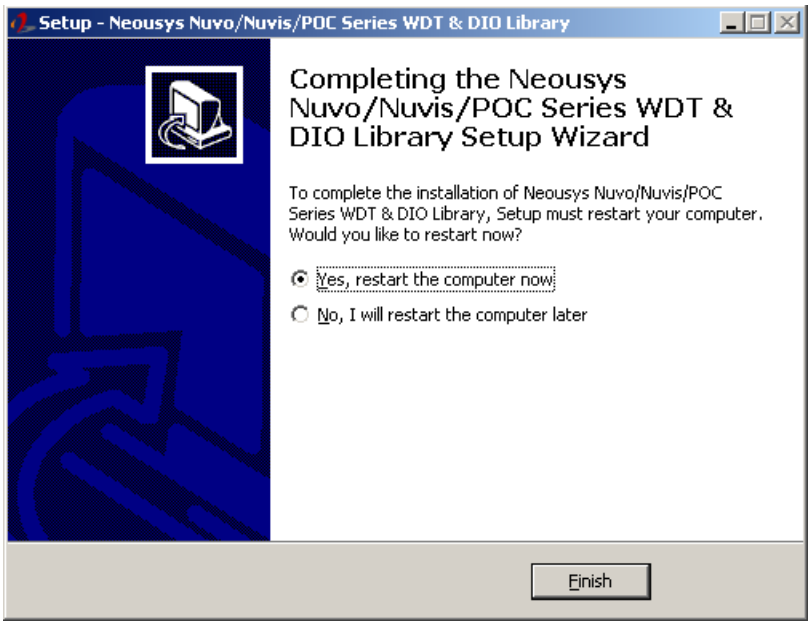


Figure 81: Complete WDT and DIO Library Installation

4. When you programming your WDT or DIO program, the related files are located in

- Header File: \Include
- Library File: \Lib
- Function Reference: \Manual
- Sample Code: \Sample\

5.2 Using WDT Function

5.2.1 WDT Function Reference

5.2.1.1 InitWDT

Description

Initialize the WDT function. You should always invoke InitWDT() before set or start watchdog timer.

Parameter

None

Return Value

Returns TRUE if initialization successes, FALSE if initialization failed.

Usage

```
BOOL bRet = InitWDT()
```

5.2.1.2 SetWDT

Description

Set timeout value and unit for watchdog timer. When InitWDT() is invoked, a default timeout value of 255 seconds is assigned.

Parameter

tick

WORD value (1 ~ 65535) to indicate timeout ticks.

unit

BYTE value (0 or 1) to indicate unit of timeout ticks.

0 : unit is minute
1 : unit is second

Return Value

If value of unit is correct (0 or 1), this function returns TRUE, otherwise FALSE.

Usage

```
WORD tick = 255;
BYTE unit = 1; //unit is second.

BOOL bRet = SetWDT(tick, unit); //timeout value is 255 seconds
```

5.2.1.3 StartWDT

Description

Start countdown of WDT. When WDT is started, the WDT LED indicator starts to blink in a frequency of 1Hz. If no ResetWDT() or StopWDT is invoked before WDT is counted to 0, the WDT expires and system resets.

Parameter

None

Return Value

If the timeout value is given in correct format, this function returns TRUE, otherwise FALSE.

Usage

```
BOOL bRet = StartWDT()
```

5.2.1.4 ResetWDT

Description

Reset the timeout value to the value given by SetWDT(). If no ResetWDT() or StopWDT is invoked before WDT is counted to 0, the WDT expires and system resets.

Parameter

None

Return Value

Always returns TRUE;

Usage

```
BOOL ret = ResetWDT()
```

5.2.1.5 StopWDT

Description

Stop the countdown of WDT. When WDT is stopped, the WDT LED indicator stops blinking.

Parameter

None

Return Value

Always returns TRUE;

Usage

```
BOOL ret = StopWDT()
```

5.3 Using DIO Function

5.3.1 Wiring for Isolated DIO

The digital input function of Nuvo series is implemented using a photo-coupler with an internally series-connected 1kΩ resistor. You need to provide a voltage to specify the logic high/low state. The input voltage for logic high is 5~24V, and the input voltage for logic low is 0~1.5V.

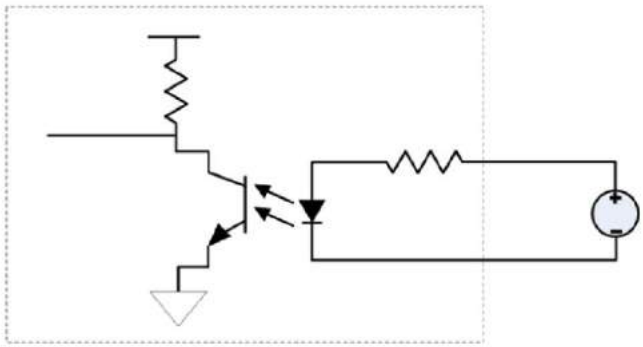


Figure 82: Wiring of isolated digital inputs

The digital output function of Nuvo series is implemented using Power MOSFET + Analog Device iCoupler® component. The DO channels are configured as NO (normally-open) configuration. When you turn on system, all DO channels have a deterministic state of logic 0 (circuit disconnected from GND return). When logic 1 is specified, MOSFET is activated and GND return path is established. The digital output function on Nuvo series supports sinking current connection. The following diagrams are the suggested wiring for DO:

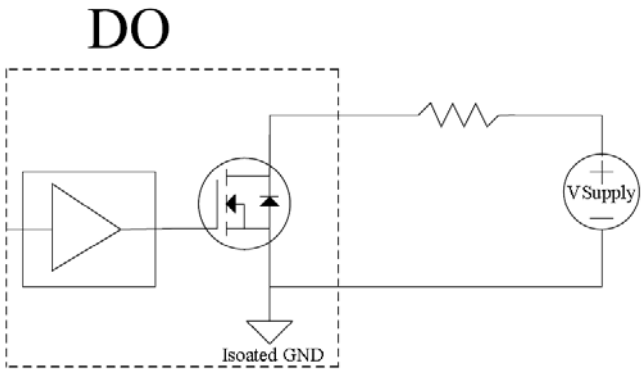


Figure 83: Wiring of isolated digital outputs

5.3.2 DIO Function Reference

5.3.2.1 InitDIO

Syntax

```
BOOL InitDIO(void);
```

Description

Initialize the DIO function. You should always invoke InitDIO() before write/read any DIO port/channel.

Parameter

None

Return Value

Returns TRUE if initialization successes, FALSE if initialization failed.

Usage

```
BOOL bRet = InitWDT()
```

5.3.2.2 DIReadLine

Syntax

```
BOOL DIReadLine(BYTE ch);
```

Description

Read a single channel of isolated digital input.

Parameter

ch

BYTE value specifies the DI channel to be read. Valid values are 0~3.

Return Value

The status (TRUE or FALSE) of the specified DI channel.

Usage

```
BYTE ch=3; //DI channel #3
BOOL DIChValue = DIReadLine(ch); //read DI channel #3
```

5.3.2.3 DIReadPort

Syntax

```
WORD DIReadPort(void);
```

Description

Read the entire isolated digital input port (8 channels).

Parameter

None

Return Value

A WORD value indicates the status of DI port. Return value are 0~255.

Usage

```
WORD DIPortValue = DIReadPort ();
```

5.3.2.4 DOWriteLine

Syntax

```
void DOWriteLine(BYTE ch, BOOL value);
```

Description

Write a single channel of isolated digital output.

Parameter

ch

BYTE value specifies the DO channel to be written. Valid value are 0~7.

value

BOOL value (TRUE or FALSE) specifies the status of DO channel.

Return Value

None

Usage

```
BYTE ch=3; //DI channel #3
BOOL DOChValue=TRUE;

DOWriteLine(ch, DOChValue); //write DO channel #3 as TRUE
```

5.3.2.5 DOWritePort

Syntax

```
void DOWritePort(WORD value);
```

Description

Write the entire isolated digital output port (8 channels).

Parameter

value

WORD value specifies the status of the DO port. Valid values are 0~255.

Return Value

None

Usage

```
WORD DOPortValue=0XFF; //11111111b
DOWritePort(DOPortValue); //write DO port as 11111111b
```

5.3.2.6 DOWriteLineChecked

Syntax

```
void DOWriteLineChecked(BYTE ch, BOOL value);
```

Description

Write a single channel of isolated digital output and read-back the value of DO register. Note that this function is not returned until the DO register is checked and identical to the written value.

Parameter

ch

BYTE value specifies the DO channel to be written. Valid values are 0~7.

value

BOOL value (TRUE or FALSE) specifies the status of DO channel.

Return Value

None

Usage

```
BYTE ch=3; //DI channel #3
BOOL DOChValue=TRUE;

DOWriteLineChecked(ch, DOChValue); //write DO channel #3 as TRUE
```

5.3.2.7 DOWritePortChecked

Syntax

```
void DOWritePortChecked(WORD value);
```

Description

Write the entire isolated digital output port (8 channels) and check it has been done. Note that this function is not returned until the write value has been checked the same with the device registry.

Parameter

value

WORD value specifies the status of the DO port. Valid values are 0~255.

Return Value

None

Usage

```
WORD DOPortValue=0XFF; //11111111b
DOWritePortChecked(DOPortValue); //write DO port as 11111111b
```

5.4 Using CAN Bus Function

CAN Bus port is only available on Nuvo-2510VTC. The other standard products of Nuvo-2500 Series don't support this functionality. Users can configure the CAN Bus, as well as get access to the bus, via the following APIs. This section provides API functions, parameters and definitions for users to create their own software applications.

5.4.1 CAN Bus Function Reference

5.4.1.1 CAN_RegisterReceived

Syntax

```
BOOL CAN_RegisterReceived(DWORD idx, void (__stdcall
*pfnHandler)(CAN_MSG *lpMsg, DWORD cbMsg));
```

Description

Registers a callback function, which is called when the CAN controller has received a message object.

Parameter

idx [in]

Specifies the index of CAN bus controllers. Currently there are only one CAN bus controller.

pfnHandler [in]

Specifies the callback function. The prototype for this function is descripted as follow.

```
void __stdcall CAN0_Received(CAN_MSG *lpMsg, DWORD cbMsg);
```

Return Value

Returns TRUE if registration successes, FALSE if registration failed.

Usage

```
void __stdcall CAN0_Received(CAN_MSG *lpMsg, DWORD cbMsg)
{
    printf( "%08x, %d, %02x %02x %02x %02x %02x %02x %02x
```

```
        %02x\r\n", lpMsg->id, lpMsg->len, lpMsg->data[0],
        lpMsg->data[1], lpMsg->data[2], lpMsg->data[3],
        lpMsg->data[4], lpMsg->data[5], lpMsg->data[6],
        lpMsg->data[7]);
}

void main(int argc, char *argv[])
{
        if ( ! CAN_RegisterReceived(0, CAN0_Received) )
        {
                printf("CAN_RegisterReceived is FAILED !\r\n");
        }
}
```

5.4.1.2 CAN_RegisterStatus

Syntax

```
BOOL CAN_RegisterStatus(DWORD idx, void (__stdcall
*pfnHandler)(DWORD status));
```

Description

Registers a callback function, which is called when the CAN controller has received a status interrupt.

Parameter

idx [in]

Specifies the index of CAN bus controllers. Currently there are only one CAN bus controller.

pfnHandler [in]

Specifies the callback function. The prototype for this function is descripted as follow.

```
void __stdcall CAN0_Status(DWORD status);
```

Return Value

Returns TRUE if registration successes, FALSE if registration failed.

Usage

```
void __stdcall CAN0_Status(DWORD status)
{
     printf("%08x\r\n", status);
}
void main(int argc, char *argv[])
{
     if ( ! CAN_RegisterStatus(0, CAN0_Status) )
     {
          printf("CAN_RegisterStatus is FAILED !\r\n");
     }
}
```

5.4.1.3 CAN_Setup

Syntax

```
BOOL CAN_Setup(DWORD idx, CAN_SETUP *lpSetup, DWORD cbSetup);
```

Description

Setup timing correlation of trigger source input (DI) and pulse target output (DO) for Deterministic Trigger Fan-Out function.

Parameter

idx [in]

Specifies the index of CAN bus controllers. Currently there are only one CAN bus controller.

lpSetup [in]

A pointer to a CAN_SETUP structure that contains the CAN-Bus configuration. This data structure contains the following variables:

Bitrate

DWORD value specifies the bit rate of the specified CAN controller.

recvConfig

DWORD value specifies the received message configuration.

recvId

DWORD value specifies the received identity number.

recvMask

DWORD value specifies the mask for received identity number.

cbSetup [in]

The length of the structure, in bytes. The caller must set this member to sizeof(CAN_SETUP).

Return Value

Returns TRUE if setup successes, FALSE if setup failed.

Usage

5.4.1.4 CAN_Start

Syntax

```
BOOL CAN_Start(DWORD idx);
```

Description

Start CAN Bus transmit and receive operation.

Parameter

idx [in]

Specifies the index of CAN bus controllers. Currently there are only one CAN bus controller.

Return Value

Returns TRUE if start procedure successes, FALSE if start procedure failed.

Usage

```
BOOL  bRet = CAN_Start(0);
```

5.4.1.5 CAN_Stop

Syntax

```
BOOL CAN_Stop(DWORD idx);
```

Description

Stop CAN Bus controller operation.

Parameter

idx [in]

Specifies the index of CAN bus controllers. Currently there are only one CAN bus controller.

Return Value

Returns TRUE if stop procedure successes, FALSE if stop procedure failed.

Usage

```
BOOL  bRet = CAN_Stop(0);
```

5.4.1.6 CAN_Send

Syntax

```
BOOL CAN_Send(DWORD idx , CAN_MSG *lpMsg, DWORD cbMsg);
```

Description

Put message object into CAN Bus controller.

Parameter

idx [in]

Specifies the index of CAN bus controllers. Currently there are only one CAN bus controller.

lpMsg [in]

A pointer to a CAN_MSG structure that contains the CAN-Bus message object. This data structure contains the following variables:.

id

DWORD value specifies the identity number of the message object.

flags

WORD value specifies the status or configuration of the message object.

extra

BYTE value specifies nothing but does reserved.

len

BYTE value specifies the length of the message object.

data

BYTE array specifies the data of the message object.

cbMsg [in]

The length of the structure, in bytes. The caller must set this member to sizeof(CAN_MSG).

Return Value

Returns TRUE if stop procedure successes, FALSE if stop procedure failed.

Usage

```
void main(int argc, char *argv[])
{
     CAN_MSG  canMsg;
     memset(&canMsg, 0, sizeof(canMag));
     if ( ! CAN_Send(0, &canMsg, sizeof(canMsg)) )
     {
          printf("CAN_Send is FAILED !\r\n");
     }
}
```

5.5 Using Ignition Control Function

Ignition Power Control is available only on Nuvo-2510VTC. The other standard products of Nuvo-2500 Series don't support this functionality. This is mainly a hardware functionality and control by the on-board micro controller. However, the following APIs enable users to check the status and configuration. Therefore, actions in users' software applications are possible according to different ignition status.

5.5.1 Ignition Control Function Reference

5.5.1.1 IGN_GetState

Syntax

```
BOOL IGN_GetState(DWORD *lpState);
```

Description

Get information about the ignition control state.

Parameter

lpState [in]

pointer to DWORD value which specifies the state of ignition control.

Return Value

Returns TRUE if stop procedure successes, FALSE if stop procedure failed.

Usage

```
void main(int argc, char *argv[])
{
     DWORD  state;
     state = 0;
     if ( ! IGN_GetState(&state) )
     {
          printf("IGN_GetState is FAILED !\r\n");
     }
```

```
}
```


5.5.1.2 IGN_GetBatteryVoltage

Syntax

```
BOOL IGN_GetBatteryVoltage(double *lpVoltage);
```

Description

Get information about the battery voltage.

Parameter

lpVoltage [in]

pointer to double value which specifies the battery voltage.

Return Value

Returns TRUE if stop procedure successes, FALSE if stop procedure failed.

Usage

```
void main(int argc, char *argv[])
{
     double  voltage;
     voltage = 0;
     if ( ! IGN_GetBatteryVoltage(&voltage) )
     {
          printf("IGN_GetBatteryVoltage is FAILED !\r\n");
     }
}
```

5.5.1.3 IGN_GetSetting

Syntax

```
BOOL IGN_GetSetting(IGN_SETTING *lpSetting, DWORD cbSetting);
```

Description

Gets information about the BIOS ignition setting.

Parameter

lpSetting [in]

A pointer to a **IGN_SETTING** structure that contains the ignition setting from BIOS. This data structure contains the following variables:

mode

BYTE value specifies the operation mode of the ignition controller. (0: atx, 1: at, 2~7: ignition mode)

batteryType

BYTE value specifies the battery type. (0: 12v, 1: 24v)

isSmartOff

BYTE value specifies the smart off function is enabled or not. (0: disabled, 1: enabled)

isPostCheck

BYTE value specifies the post check function is enabled or not. (0: disabled, 1: enabled)

onDelay

DWORD value specifies the delay count in seconds until ignition controller turn on the machine.

offDelay

DWORD value specifies the delay count in seconds until ignition controller turn off the machine.

hardOffTimeout

DWORD value specifies the limit count in seconds while ignition controller could not turn off the machine.

lowVolThreshold

DOUBLE precision floating point number specifies the low voltage limit then ignition controller will turn off the machine.

Return Value

Returns TRUE if procedure successes, FALSE if procedure failed.

Usage

```
int main(int argc, char *argv[])
{
     IGN_SETTING  setting;
     memset(&setting, 0, sizeof(setting));
     if ( ! IGN_GetSetting(&setting, sizeof(setting)) )
     {
          printf("IGN_GetSetting is FAILED !\r\n");
          return -1;
     }
     return 0;
}
```

5.6 Using PWM Function

Pulse-width modulation, also known as PWM, output is available on Nuvis-2520at. And it's also available as an option on Nuvo-2500. Nuvo-2510VTC doesn't support this functionality. The Pulse Width Modulator (PWM) API provides a set of functions for programming and operating the PWM controller. Each generator block has two PWM output signals, which can be operated independently or as a pair of signals with dead band delays inserted. The control block determines the polarity of the PWM signals and which signals are passed through to the pins.

5.6.1 PWM Function Reference

5.6.1.1 PWM_RegisterStatus

Syntax

```
BOOL PWM_RegisterStatus(void (__stdcall *pfnHandler)(DWORD
genBits, DWORD status));
```

Description

Registers a callback function, which is called when the PWM controller has a status interrupt.

Parameter

pfnHandler [in]

Specifies the callback function. The prototype for this function is descripted as follow.

```
void (__stdcall *pfnHandler)(DWORD genBits, DWORD status);
```

Return Value

Returns TRUE if registration successes, FALSE if registration failed.

Usage

```
void __stdcall PWM_Status(DWORD genBits, DWORD status)
{
    printf("%08x: %08x\n", genBits, status);
```

```
}

int main(int argc, char *argv[])
{
     if ( ! PWM_RegisterStatus(PWM_Status)
     {
          printf("PWM_RegisterStatus is FAILED !\n");
          return -1;
     }
     return 0;
}
```

5.6.1.2 PWM_ClockSet

Syntax

```
BOOL PWM_ClockSet(DWORD idx, DWORD divClock);
```

Description

Sets the PWM clock configuration.

Parameter

idx [in]

Specifies the index of PWM controllers. Currently there is only one PWM controller.

divClock [in]

Specifies the PWM clock divider as the PWM clock source. It must be one of PWM_CLK_DIV_1, PWM_CLK_DIV_2, PWM_CLK_DIV_4, PWM_CLK_DIV_8, PWM_CLK_DIV_16, PWM_CLK_DIV_32, or PWM_CLK_DIV_64.

Return Value

Returns TRUE if registration successes, FALSE if registration failed.

Usage

```
BOOL ret = PWM_ClockSet(0, PWM_CLK_DIV_1);
```

5.6.1.3 PWM_GenSetup

Syntax

```
BOOL PWM_GenSetup(DWORD genBits, PWM_GEN_SETUP *lpSetup, DWORD
cbSetup);
```

Description

Sets the PWM clock configuration.

Parameter

genBits [in]

Specifies the generator block bits of PWM controllers.

lpSetup [in]

A pointer to a **PWM_GEN_SETUP** structure that contains the PWM generator configuration. This data structure contains the following variables:

genMode

DWORD value specifies the generator mode of the specified PWM controller.

intrTriggers

DWORD value specifies the interrupts and triggers to be enabled.

deadBandRise

WORD value specifies the width of delay from the rising edge. Valid:0~4096, and 0:disable

deadBandFall

WORD value specifies the width of delay from the falling edge. Valid:0~4096, and 0:disable.

cbSetup [in]

The length of the structure, in bytes. The caller must set this member to sizeof(PWM_GEN_SETUP).

Return Value

Returns TRUE if procedure successes, FALSE if procedure failed.

Usage

```
PWM_GEN_SETUP setup;
memset(&setup, 0, sizeof(setup));
setup.genMode = PWM_GEN_MODE_UP_DOWN | PWM_GEN_MODE_NO_SYNC;
BOOL ret = PWM_GenSetup(PWM_GEN_0, &setup, sizeof(setup));
```

5.6.1.4 PWM_GenPeriod

Syntax

```
BOOL PWM_GenPeriod(DWORD genBits, DWORD period);
```

Description

Sets the period of a PWM generator.

Parameter

genBits [in]

Specifies the generator block bits of PWM controllers.

period [in]

Specifies the period of PWM generator output, measured in clock ticks.

Return Value

Returns TRUE if procedure successes, FALSE if procedure failed.

Usage

```
BOOL ret = PWM_GenPeriod(PWM_GEN_0 | PWM_GEN_1, 800);  // 100 KHz
```

5.6.1.5 PWM_PulseWidth

Syntax

```
BOOL PWM_PulseWidth(DWORD pinBits, DWORD width);
```

Description

Sets the pulse width for the specified PWM output.

Parameter

pinBits [in]

Specifies the output bits of PWM controllers.

width [in]

Specifies the width of the positive portion of the pulse.

Return Value

Returns TRUE if procedure successes, FALSE if procedure failed.

Usage

```
BOOL bRet = PWM_PulseWidth(PWM_PIN_0, 400);
```

5.6.1.6 PWM_PulseInvert

Syntax

```
BOOL PWM_PulseInvert(DWORD pinBits);
```

Description

Sets the inversion mode for PWM outputs.

Parameter

pinBits [in]

Specifies the output bits of PWM controllers.

Return Value

Returns TRUE if procedure successes, FALSE if procedure failed.

Usage

```
BOOL bRet = PWM_PulseInvert(PWM_PIN_0);
```

5.6.1.7 PWM_Start

Syntax

```
BOOL PWM_Start(DWORD pinBits);
```

Description

Starts PWM controller operation.

Parameter

pinBits [in]

Specifies the output bits of PWM controllers.

Return Value

Returns TRUE if start procedure successes, FALSE if start procedure failed.

Usage

```
BOOL bRet = PWM_Start(PWM_PIN_0 | PWM_PIN_1);
```

5.6.1.8 PWM_Stop

Syntax

```
BOOL PWM_Stop(DWORD pinBits);
```

Description

Stops PWM controller operation.

Parameter

pinBits [in]

Specifies the output bits of PWM controllers.

Return Value

Returns TRUE if stop procedure successes, FALSE if stop procedure failed.

Usage

```
BOOL bRet = PWM_Stop(PWM_PIN_0 | PWM_PIN_1);
```

5.6.1.9 PWM_SyncTimeBase

Synchronizes the counters in one or multiple PWM generator blocks.

Syntax

```
BOOL PWM_SyncTimeBase(DWORD genBits);
```

Description

For the selected PWM module, this function synchronizes the time base of the generator blocks by causing the specified generator counters to be reset to zero.

Parameter

genBits [in]

Specifies the generator bits of PWM controllers.

Return Value

Returns TRUE if procedure successes, FALSE if procedure failed.

Usage

```
BOOL bRet = PWM_SyncTimeBase(PWM_GEN_0 | PWM_GEN_1);
```

5.6.1.10 PWM_SyncUpdate

Synchronizes all pending updates.

Syntax

```
BOOL PWM_SyncUpdate(DWORD genBits);
```

Description

For the selected PWM generators, this function causes all queued updates to the period or pulse width to be applied the next time the corresponding counter becomes zero.

Parameter

genBits [in]

Specifies the generator bits of PWM controllers.

Return Value

Returns TRUE if procedure successes, FALSE if procedure failed.

Usage

```
BOOL bRet = PWM_SyncUpdate(PWM_GEN_0 | PWM_GEN_1);
```

5.7 Using ADC Function

Analog-to-digital converter, also known as ADC, is available on Nuvis-2520at. And it's also available as an option on Nuvo-2500. Nuvo-2510VTC doesn't support this functionality. The ADC API provides a set of functions for operating the ADC controller and reading the analog inputs.

NOTE *The analog inputs are mainly for human interface devices with analog output signals, such as hall sensor joystick and analog stick, and not highly calibrated. It's not suitable for any measurement applications.*

5.7.1 ADC Function Reference

5.7.1.1 ADC_Start

Syntax

```
BOOL ADC_Start(DWORD idx);
```

Description

Starts the ADC controller operation.

Parameter

idx [in]

Specifies the index of ADC controllers. Valid values are 0 and1.

Return Value

Returns TRUE if procedure successes, FALSE if procedure failed.

Usage

```
BOOL ret = ADC_Start(0);
```

5.7.1.2 ADC_Stop

Syntax

```
BOOL ADC_Stop(DWORD idx);
```

Description

Stops the ADC controller operation.

Parameter

idx [in]

Specifies the index of ADC controllers. Valid values are 0 and1.

Return Value

Returns TRUE if procedure successes, FALSE if procedure failed.

Usage

```
BOOL ret = ADC_Stop(0);
```

5.7.1.3 ADC_GetData

Syntax

```
BOOL ADC_GetData(DWORD idx, double *lpData);
```

Description

Gets the current data of the specified ADC controller.

Parameter

idx [in]

Specifies the index of ADC controllers. Valid values are 0 and1.

lpData [out]

Points to the double value which specifies the current data of the specified ADC controller.

Return Value

Returns TRUE if procedure successes, FALSE if procedure failed.

Usage

```
double data = 0.0;
BOOL ret = ADC_GetData(0, &data);
```

5.8 Using QEI Function

Quadrature encoder input is available on Nuvis-2520at. And it's also available as an option on Nuvo-2500. Nuvo-2510VTC doesn't support this functionality. The QEI API provides a set of functions for programming and operating the QEI controller.

5.8.1 QEI Function Reference

5.8.1.1 QEI_Start

Syntax

```
BOOL QEI_Start(DWORD idx);
```

Description

Starts the QEI controller operation.

Parameter

idx [in]

Specifies the index of QEI controllers. Currently there is only one QEI controller.

Return Value

Returns TRUE if operation successes, FALSE if operation failed.

Usage

```
BOOL ret = QEI_Start(0);
```

5.8.1.2 QEI_Stop

Syntax

```
BOOL QEI_Stop(DWORD idx);
```

Description

Stops the QEI controller operation.

Parameter

idx [in]

Specifies the index of QEI controllers. Currently there is only one QEI controller.

Return Value

Returns TRUE if operation successes, FALSE if operation failed.

Usage

```
BOOL ret = QEI_Stop(0);
```

5.8.1.3 QEI_Setup

Syntax

```
BOOL QEI_Setup(DWORD idx, QEI_SETUP *lpSetup, DWORD cbSetup);
```

Description

Setup some parameters used in the specified QEI controller.

Parameter

idx [in]

Specifies the index of QEI controllers. Currently there is only one QEI controller.

lpSetup [in]

A pointer to a **QEI_SETUP** structure that contains the QEI configuration. This data structure contains the following variables:

config

DWORD value specifies the configuration for the quadrature encoder.

maxPos

DWORD value specifies the maximum position value.

velPeriod

DWORD value specifies the number of clock ticks over which to measure the velocity. Set 0 value to disable velocity function.

velPreDiv

DWORD value specifies the predivider applied to the input quadrature signal before it is counted.

cbSetup [in]

The length of the structure, in bytes. The caller must set this member to sizeof(QEI_SETUP).

Return Value

Returns TRUE if setup successes, FALSE if setup failed.

Usage

```
int main(int argc, char *argv[])
{
     QEI_SETUP  setup;
     memset(&setup, 0, sizeof(setup));
     setup.config = QEI_CONFIG_CAPTURE_A_B;
     setup.maxPos = 10000;
     setup.velPeriod = 0;  // disable
     setup.velPreDiv = QEI_VEL_DIV_1;
     if ( ! QEI_Setup(0, &setup, sizeof(setup)) )
     {
          printf("QEI_Setup is FAILED !\r\n");
          return -1;
     }
     return 0;
}
```

5.8.1.4 QEI_GetDirection

Gets the current direction of rotation.

Syntax

```
BOOL QEI_GetDirection(DWORD idx, DWORD *lpDirection);
```

Description

This function returns the current direction of rotation. In this case, current means the most recently detected direction of the encoder; it may not be presently moving but this is the direction it last moved before it stopped.

Parameter

idx [in]

Specifies the index of QEI controllers. Currently there is only one QEI controller.

lpDirection [out]

Point to the DWORD value specifies the current direction of rotation, which is 1 if moving in the forward direction or -1 if moving in the reverse direction.

Return Value

Returns TRUE if operation successes, FALSE if operation failed.

Usage

```
DWORD  dir = 0;
BOOL ret = QEI_GetDirection(0, &dir);
```

5.8.1.5 QEI_GetVelocity

Gets the current encoder speed.

Syntax

```
BOOL QEI_GetVelocity(DWORD idx, DWORD *lpVelocity);
```

Description

This function gets the current position of the encoder. Depending upon the configuration of the encoder, and the incident of an index pulse, this value may or may not contain the expected data (that is, if in reset on index mode, if an index pulse has not been encountered, the position counter is not yet aligned with the index pulse).

Parameter

idx [in]

Specifies the index of QEI controllers. Currently there is only one QEI controller.

lpVelocity [out]

Points to a DWORD value specifies the number of pulses captured in the given time period.

Return Value

Returns TRUE if operation successes, FALSE if operation failed.

Usage

```
DWORD  vel = 0;
BOOL ret = QEI_GetVelocity(0, &vel);
```

5.8.1.6 QEI_GetPosition

Gets the current encoder position.

Syntax

```
BOOL QEI_GetPosition(DWORD idx, DWORD *lpPosition);
```

Description

This function gets the current position of the encoder.

Parameter

idx [in]

Specifies the index of QEI controllers. Currently there is only one QEI controller.

lpPosition [out]

Points to the DWORD value specifies current position of the encoder.

Return Value

Returns TRUE if operation successes, FALSE if operation failed.

Usage

```
DWORD  pos = 0;
BOOL ret = QEI_GetPosition(0, &pos);
```

5.8.1.7 QEI_SetPosition

Gets the current encoder position.

Syntax

```
BOOL QEI_SetPosition(DWORD idx, DWORD position);
```

Description

This function sets the current position of the encoder.

Parameter

idx [in]

Specifies the index of QEI controllers. Currently there is only one QEI controller.

position [in]

DWORD value specifies current position of the encoder.

Return Value

Returns TRUE if operation successes, FALSE if operation failed.

Usage

```
BOOL ret = QEI_setPosition(0, 0);
```

5.9 Using PoE Port Function

Power over Ethernet, also known as PoE, is available on Nuvo-2510VTC and Nuvis-2520at. Nuvo-2500 doesn't support this functionality. The PoE Port APIs enable user to implement per-port on/off in users' software applications.

5.9.1 PoE Port Function Reference

5.9.1.1 GetStatusPoEPort

Syntax

```
BYTE GetStatusPoEPort(BYTE port);
```

Description

Gets the status of the PoE Port on the platform.

Parameter

port [in]

Specifies the port number on the platform.

Return Value

Returns the status of the PoE Port on the platform (0: disabled, 1: enabled).

Usage

```
BYTE status = GetStatusPoEPort(1);
```

5.9.1.2 EnablePoEPort

Syntax

```
BOOL EnablePoEPort(BYTE port);
```

Description

Enables the PoE Port on the platform.

Parameter

port [in]

Specifies the port number on the platform.

Return Value

Returns TRUE if operation successes, FALSE if operation failed.

Usage

```
BOOL ret = EnablePoEPort(1);
```

5.9.1.3 DisablePoEPort

Syntax

```
BOOL DisablePoEPort(DWORD port);
```

Description

Disables the PoE Port on the platform.

Parameter

port [in]

Specifies the port number on the platform.

Return Value

Returns TRUE if operation successes, FALSE if operation failed.

Usage

```
BOOL ret = DisablePoEPort(1);
```

5.10 Using LED Function

LED Driving Channel is only available on Nuvis-2520at. The other standard products of Nuvo-2500 Series don't support this functionality. The LED functions are provided necessary capabilities to communicate with on-board chip.

5.10.1 LED Function Reference

5.10.1.1 LED_SetCurrentDriving

Syntax

```
BOOL LED_SetCurrentDriving(DWORD mode, DWORD current);
```

Description

Gets the status of the PoE Port on the specified PCI device.

Parameter

mode [in]

Specifies the mode for driving LED (0: disabled, 1: constant current, 2: constant voltage).

current [in]

Specifies the current number to driving LED.

Return Value

Returns TRUE if operation successes, FALSE if operation failed.

Usage

```
BOOL ret = LED_SetCurrentDriving(1, 1);
```

5.11 Using MCU Function

The Micro-Controller Unit (MCU) functions are provided necessary capabilities to communicate with on-board chip.

5.11.1 LED Function Reference

5.11.1.1 MCU_GetVersion

Syntax

```
BOOL MCU_GetVersion(DWORD *lpDateCode);
```

Description

Gets the version code of on-board MCU chip.

Parameter

lpDateCode [out]

Specifies the version code which format is 0xYYYYMMDD, e.g. 0x20150701.

Return Value

Returns TRUE if operation successes, FALSE if operation failed.

Usage

```
DWORD code = 0;
BOOL ret = MCU_GetVersion(&code);
```