Quantum with Unity Pro

140 HLI 340 00 High Speed I/O Module with Interrupt Function User Manual

10/2013

35012591.04

www.schneider-electric.com

Schneider Electric

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

Table of Contents

Safety Information

Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a Danger safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates an imminently hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a potentially hazardous situation which, if not avoided, **can result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

PLEASE NOTE

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

About the Book

At a Glance

Document Scope

This documention presents the Quantum Automation Series 140 HLI 340 00 High Speed I/O Module with Interrupt Function. The topics include the Quantum System and Panel Software Support, Related Communication Sources, an Overview, and the Theory of Operation. In a second chapter on Hardware Overview and System Specifications, the document presents the140 HLI 340 00 High Speed Latch and Interrupt Module Hardware Specifications, Specifications and Wiring Diagram, and the Quantum Automation Series System Specifications.

Validity Note

This documentation is valid for Unity Pro v8.0.

Part I
Functional Overview

Introduction

This part describes the functional and physical characteristics of the Quantum Automation Series 140 HLI 340 00 High Speed Interrupt I/O Module

What Is in This Part?

This part contains the following chapters:

Chapter 1
Introduction

Overview

Quantum High Speed Module

The 140 HLI 340 00 is the Quantum high speed, latch, and interrupt input module. Each of the module' s 16 independently configured 24 Vdc inputs can be used with sink or source field devices. Primarily designed for use in time critical I/O applications, the module is currently supported for use only in local backplane configurations. This restriction applies regardless of the intended operating mode and point configuration of the module. When used in conjunction with I/O events (interrupt handling), inputs can be updated asynchronously to the normal I/O scanning mechanism. Inputs are updated once per scan for normal logic solve processing.

The following sections include descriptions, along with features and restrictions of the operating modes.

NOTE: It's not recommended to use this module with an input frequecy higger than 2.5 Khz (risk of loss of event)

High Speed Input Mode

The default configuration of the 140 HLI 340 00 module is high-speed mode.

Features

- This mode is the default configuration of the module
- Response times as fast as 30 μs turn on and 130 μs turn off
- Normal I/O service routine is utilized
- Compatibility with two or three wire input devices
- Used with sink or source field devices

Latched High Speed Input Mode

In this mode, the module is read by the controller at the end of the scan, similar to other input modules in the Quantum product line. This function captures very high speed signals which is read by the controller. It does not, however, require the processor to be interrupted.

Features

- Latching of high speed signals with pulse widths greater than 30 μs turn on and 130 μs turn off
- Automatic unlatching of the point after the controller has read the point
- Can be configured to latch by the input turning on or off
- Compatibility with two or three wire input devices
- Used with sink or source field devices

Restrictions

- This mode is only available in a local configuration, including an expansion rack.

Interrupt Mode

The interrupt function is only available over local I/O. The interrupt module provides a physical interrupt to the controllers, which are serviced by the Unity event processing.

Features

- The interrupt can be configured to trigger by the input turning on, off, or both
- Input No. 1 has a higher input priority than input No. 16 for point priority
- Slot position No. 1 has a higher priority than slot position No. 16 for module priority
- Compatibility with two or three wire input devices
- Used with sink or source field devices
- Interrupt inputs look like high speed inputs at I/O time
- The interrupt is immediately read, acknowledged by the controller, and then cleared automatically

Restrictions

- This mode is only available in a local configuration

Split Mode

All inputs of the module can be configured independently, so any mixture of High Speed Inputs, Latched Inputs and Interupt Inputs is possible.

NOTE: If the module is configured in a split mode, where there are interrupts and latched inputs, note that the latched input data is read and cleared at the interrupt service time and may not be valid at the end of the scan.

Chapter 2
Interrupt Handling with Unity Pro

Interrupt Processing with Unity Pro

Introduction

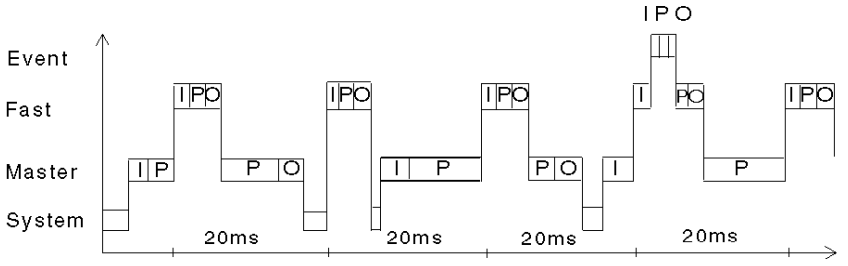
Unity Pro uses a multi tasking structure in the PLC. An application may consist of any of the following tasks:

- Master task
- Fast task
- Auxiliary task
- Event task

The task used for interrupt processing is the Event task which will be described here in more details. For a complete overview on the multitasking software structure refer to the Unity Pro Reference Manual.

Task Sequence

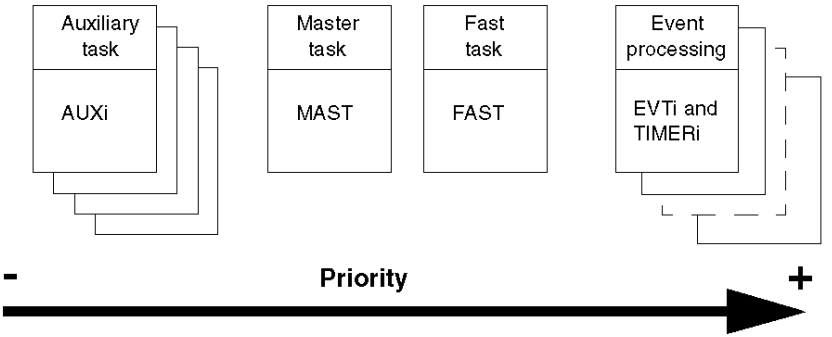
The following diagram illustrates the task sequence of multitasking processing with a cyclic master task, a fast task with a 20ms period and event processing:



I: Acquisition of Inputs
P: Program processing
O: Updating of Outputs

Task Priorities

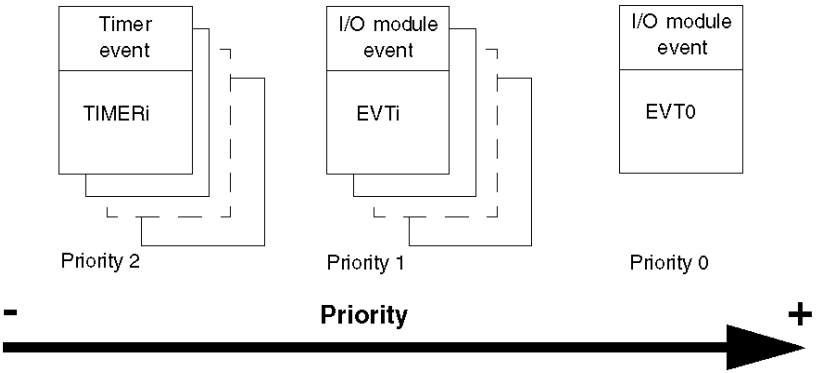
The following diagram shows the positioning of the Event tasks in the priority hierarchy of a multitasking structure:



Event Processing

Event processes can either be triggered by the event timer or through an input signal from the HLI module. Within the Event tasks there is again a hierarchy of priorities.

The following diagram shows the positioning of the I/O events in the priority hierarchy of the Event tasks:



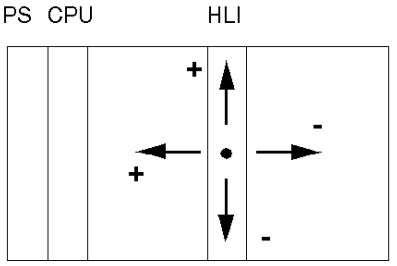
Rules for I/O Event Priorities

The following list shows which basic rules apply to Event process priorities.

- EVT0 event is the highest priority (priority 0). It can itself interrupt other types of events.
- Only one I/O signal can be attached to EVT0.
- EVTi has priority 1. It can interrupt TIMERi events.
- The priority of an input signal attached to a priority 1 event is determined by:
  - the position of the HLI module in the rack
  - the position of the channel within the module

  The Input signal with the lowest position number has the highest priority.

The following illustration shows the interdependence between slot/channel number and the priority of an input signal:



Priority of a Channnel

Control of Event Processing

The various types of event processing can be globally validated or inhibited by the application program, by using the system bit %S38. If one or more events occur while they are inhibited, the associated processing is lost.

Two elementary functions of the language, `MASKEVT()` and `UNMASKEVT()`, used in the application program can also be used to mask or unmask event processing.

If one or more events occur while they are masked, they are stored by the system and the associated processing is carried out after unmasking.

Chapter 3
Operating Modes

Introduction

This chapter describes the operating modes of the 140 HLI 340 00 module.

The 140 HLI 340 00 module is point-configurable to any of three operating modes:

- interrupt handling mode
- automatic latch/unlatch mode
- high-speed input mode

What Is in This Chapter?

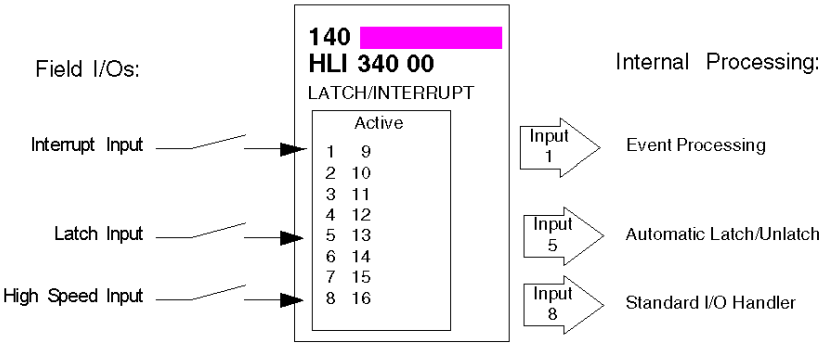
This chapter contains the following topics:

Overview

General

The 140 HLI 340 00 module is a multi-purpose high performance input, latch input, and interrupt module combined, configured through the programming software. The module is point configured to one of three modes, as is shown in the diagram below.

The following illustration shows the unit's configuration modes.



Interrupt Mode

Overview

This mode allows the main application program to be interrupted by a physical real world interrupt signal. This signal stops the Quantum CPU from solving the main logic program, and forces the Quantum CPU to start the event process configured for the respective input. Interrupt input data coming to the Quantum CPU is serviced as it arrives and is read by the Quantum CPU by utilizing hardware handshaking on the backplane system.

Interrupt Logic within the Quantum Processor

Both hardware and timer interrupts are handled within the Quantum CPU in a similar fashion, both systems use Events for the interrupt handler. Within the interrupt handler, the user application program determines what steps need to occur to handle the interrupt. In all cases, within the interrupt handler, if additional inputs need to be read, or outputs need to be written, this is done using the Immediate I/O function blocks IMIO_IN and IMIO_OUT (for detailed information see: *Direct I/O Function Block IMIO_IN, page 55* and *Direct I/O Function Block IMIO_OUT, page 58*). The IMIO functionblocks can read or write I/O information from the local backplane. For example, if an interrupt occurred and the logic needs to know the current position residing in the high speed counter module, the IMIO_IN function block would be activated, reading the position asynchronous to scan. This information could then be used in the logic section of the interrupt handler to make some logical decision based on position, and at the end of the routine update a local output module using the IMIO_OUT function block, ending the interrupt handler routine.

Interrupt Impact on Scan

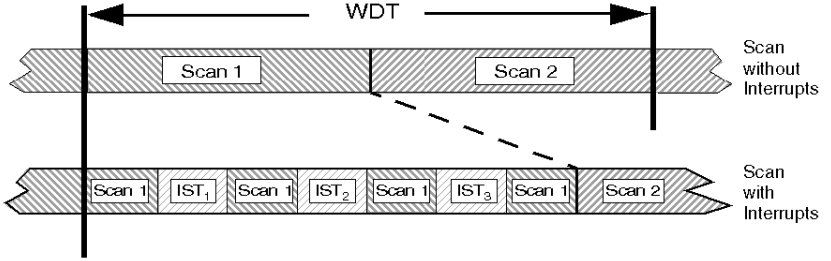
Interrupts may be triggered many times per scan, as is shown in the following diagram, with minimal impact on overall scan for most applications. There are applications however where the scan time is relatively small (10 ms) and the application requires interrupt service every 1 ms. The use of interrupt processing will allow critical sections to be serviced faster than the rest of the overall application, however be aware that you could be asking the controller to service interrupts more than the controller is capable. Users are recommended to create a timing diagram to ensure that the use of interrupts does not consume more than 40% of the total processing time of the controller. Greater than 40% could mean that the process becomes totally interrupt driven, never being able to service the rest of the application program. For those applications, it is recommended to breakup the application across several processors. Also critical to knowing the impact on scan is the duty cycle of interrupts, which is the amount of time the same interrupt will be asking for service from the controller.

The following illustration depicts the interrupt impact on a scan.

$$\text{Scan Time} + IST_1 + IST_2 + \ldots IST_n < WDT$$

Definitions:

WDT:	Watch Dog Timer setting
TFL:	Firmware Latency
TUL:	User Logic Time
IST_n:	Interupt Service Time for Interrupt n (TFL + 100µs + TUL)

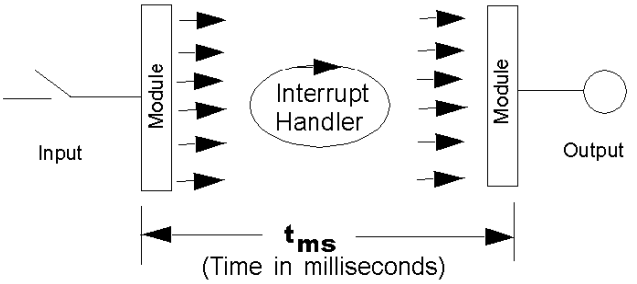


The impact on the scan depends on the size of the subroutine segment section which needs to be solved. The amount of time it takes to solve the interrupt handler subroutine logic can be calculated by adding up the instruction execution times used within the given subroutine.

General Performance Capabilities

In the area of interrupt processing, performance is measured from the time the input signal arrives at the input module, to the time an output point change state, and everything in between. This takes into consideration I/O module latency times, controller overhead for servicing interrupts and the size of the interrupt handler, as is depicted in the diagram below.

This illustration shows the general performance capabilities of the unit.



The overall performance range of the Quantum system, measured in milliseconds, is shown in the table below:

Processor	Interrupt Type	1 ms Throughput	2 ms Throughput	Logic within the Interrupt Handler	I/O Module
CPU	Hardware	2 Interrupts	4 Interrupts	Increment Counter	In:HLI340000 Out:DDO35300

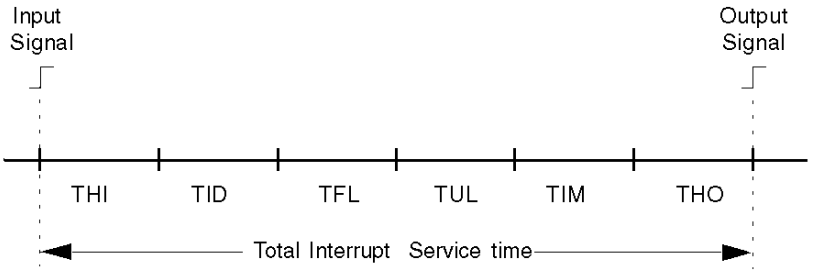
Basics of Interrupt Processing

Overview

To estimate the performance of an interrupt system it is necessary to be aware of all processes that have to be performed from the change of a signal at an input till the change of an output signal triggered by the users interrupt program.

Interrupt Process Flow

The diagram shows the process involved in an interrupt sequence:



Time	Name	Description
THI	Hardware Input Latency	THI time is defined as the time from the input field connector to the interrupt into theCPU going active.
TID	Interrupt Delay	TID time is defined as the time that firmware is unable to service interrupts. This is done because the CPU maybe communicating with other Quantum modules (I/O,NOM, RIO, etc.).
TFL	Firmware Latency	TFL time is the time that is required to identify source of the interrupt signal.
TUL	User Logic	TUL time is based strictly on the users interrupt program.
TIM	Immediate I/O	TIM time is the time from the output command of the interrupt program to the output module.
THO	Hardware Output Latency	THO time is based on the time from the output of the ASIC to the field connector.

THI

Hardware Input Latency time for the module is

- 30 microseconds turn on
- 130 microseconds turn off

TID

In order to determine the worst case delay time, the module types in the local rack must be known. The worst case time is only based on the module or operation with the longest delay time. The minimum delay is a simple module. See the table below for a listing of Quantum module TID times (worst case).

Quantum module TID times:

Operation	TID (worst case)
Hot swapping an option module	208 µs
Simple module with 4 byte rd/wr transfer*	148 µs
DPM module with 6 byte transfer**	171 µs
User logic editing/power flow display	3-100 µs
Modbus communication	243 µs
CPU Modbus Plus communication	180 µs
NOM communication with Modbus message	208 µs
NOM-like modules***	208 µs

Exceptions to operations—as noted by asterisks—in the table above which lists the Quantum module TID times (worst case).

*	140DDI35300, 140DDO35300, 140DAI54300, 140DAO84210, 140DRC83000, 140HLI34000
**	140ACO02000, 140ACI03000, 140AVI03000, 140AVO02000
***	140CHS11000, 140CRA21X00, 140MMS4250X, 140NOA61100, 140NOE2X100

TFL

The firmware latency time is based on the slot position and how many points are generating interrupts. The following calculation computes the (TFL) for that specific I/O Point.

TFL = 95 µs + (**S** * 6) µs + (**P** * 4) µs + (**X** - 1) µs
with:
S = Slot position of the highest priority interrupting module (1 — 16)
P = Highest priority interrupting point (1 — 16)
X = Number of input points interrupting

TUL

TUL time is based strictly on the users interrupt program. Listed below is an example case:

Increment counter (on interrupt) = 13 µs

TIM

The time consumed by the immediate I/O handling is basically the execution time of the IMIO_OUT function block. Should the user interrupt routine require additional input signals to be read, the execution time of IMIO_IN. IMIO_IN is not required if the additional inputs are located on the HLI module which generated the interrupt, as all channels of this module are read at itnerrupt time (see also *Split Mode, page 28*).

THO

THO time is based on the time from the output of the ASIC to the field connector.The analysis is based on a Quantum 140DDO35300.

THO time for the 140DDO35300 is:

- 100 µs turn on
- 500 µs turn off

Latch Mode

Overview

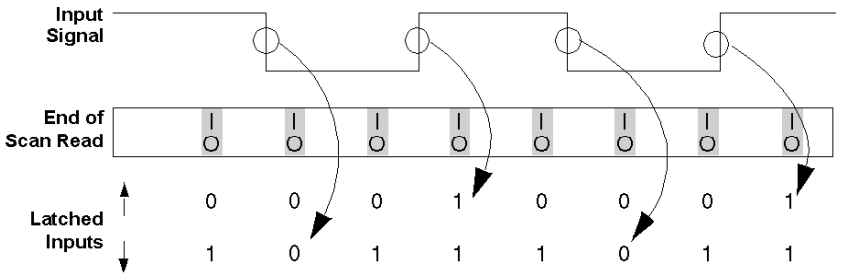
This mode latches input signals, but does not cause an interrupt to be generated within the Quantum controller. A latched signal is guaranteed to be read by the module. Immediately after the reading process, the controller automatically unlatches the input state. This is normally used in applications where the input signal on-time is less than the controllers module service time, meaning the controller could miss the input signal. Data from latch inputs is serviced using the normal input update methods, at the end of the scan (no user intervention is required).

Automatic Latch/Unlatch Method

When an input point is configured for latch mode falling edge, the corresponding input data will be latched on the transition of 24 Vdc to 0. The data will be presented to the controller as logical zero. The corresponding input data for this point will be cleared to a one' s state at the end of that respective scan (see below and following illustration).

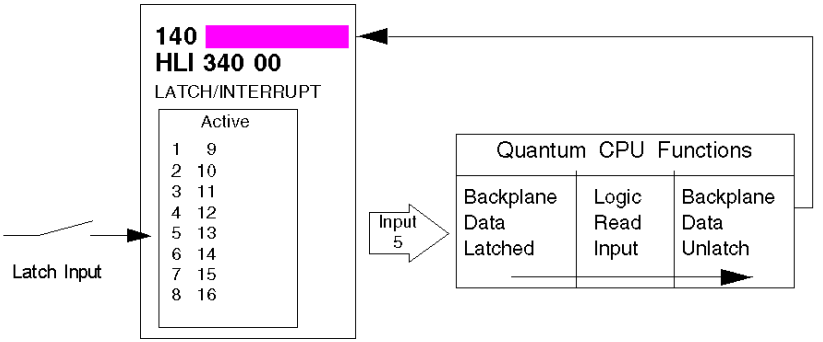
When an input point is configured for latch mode rising edge, the corresponding input data will be latched on the transition of 0 to 24 Vdc. The data will be presented to the controller as logical one. The corresponding input data for this point will be cleared to a zero' s state at the end of that respective scan (see below and following illustration).

The following illustration shows the relationship between the input and the end of scan read. The input signal is connected to two inputs configured for Latch mode, one for rising edge and one for falling edge.



NOTE: If the module is configured in a split mode, where there are interrupts and latched inputs, note that the latched input data is read and cleared at the interrupt service time and may not be valid at the end of the scan.

The following illustration shows details of the automatic latch/unlatch method.



High-Speed Input Mode

Overview

These are normal input signals which do not latch nor do they create an interrupt within the controller. Data from high speed inputs is serviced using the normal input update methods, at the end of the scan.

Features

The inputs configured as High Speed Input differ from standard I/O module inputs in their fast response time:

- 30 µs Off - On
- 130 µs On - Off

Usage of High Speed Inputs

As the scan time of an average application program is in the ms range, the normal I/O service at the end of the scan will not benefit from those fast inputs. However together with interrupt processing where all inputs on an HLI module are read immediately after the interrupt input has been triggered, it is essential to use this inputs for all time critical signals associated with the interrupt routine.

Split Mode

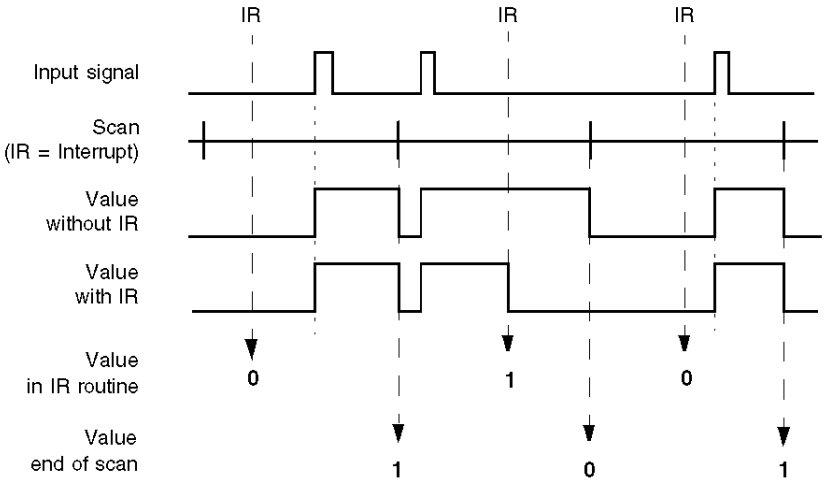
Overview

As each channel of the HLI module can be configured independently, any mix of the three modes High Speed Input, Latch Input and Interrupt may exist.

Possible Conflict

If the module is configured in a split mode, where there are interrupts and latched inputs, note that the latched input data is read and cleared at the interrupt service time and may not be valid at the end of the scan. This results in different values of the input signal within the interrupt routine and in the rest of the program. The following diagram illustrates this situation.

Value of a latched input signal as read by the interrupt routine and at the end of the scan during the normal I/O routine:



Part II
Module Description

Chapter 4
Hardware Overview and System Specifications

Purpose

This section presents an overview of the hardware and the system specifications for the 140 HLI 340 00 High Speed Latch and Interrupt Module.

What Is in This Chapter?

This chapter contains the following topics:

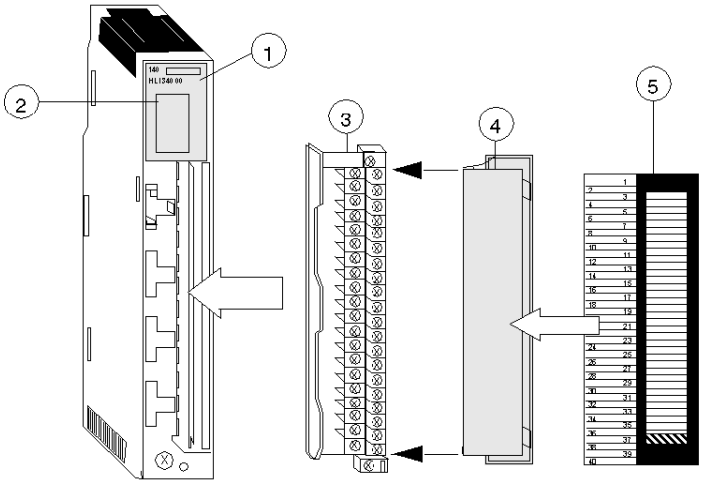
Presentation

Function

The High Speed Latch and Interrupt 24 VDC 16x1 Sink / Source input module accepts 24 VDC inputs and is for use with 24 VDC sink/source input devices.

Illustration

The following figure shows the 140 HLI 340 00 module and its components.



1 Model Number, Module Description, Color Code
2 LED Display
3 Field Wiring Terminal Strip
4 Removable Door
5 Customer Identification Label (Fold label and place it inside door)

NOTE: The field wiring terminal strip (Modicon #140 XTS 002 00) must be ordered separately. (The terminal strip includes the removable door and label.)

NOTE: The tightening torque must be between 0.5 Nm and 0.8 Nm.

NOTICE

DESTRUCTION OF ADAPTER

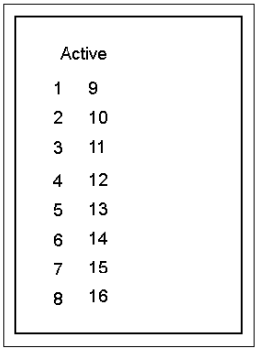
- Before tightening the locknut to the torque 0.50...0.80 Nm, be sure to properly position the right-angle F adapter connector.
- During tightening, be sure to maintain the connector securely.
- Do not tighten the right-angle F adapter beyond the specified torque.

Failure to follow these instructions can result in equipment damage.

Indicators

Illustration

The following table shows the LED indicators for the 140 HLI 340 00 module.



Descriptions

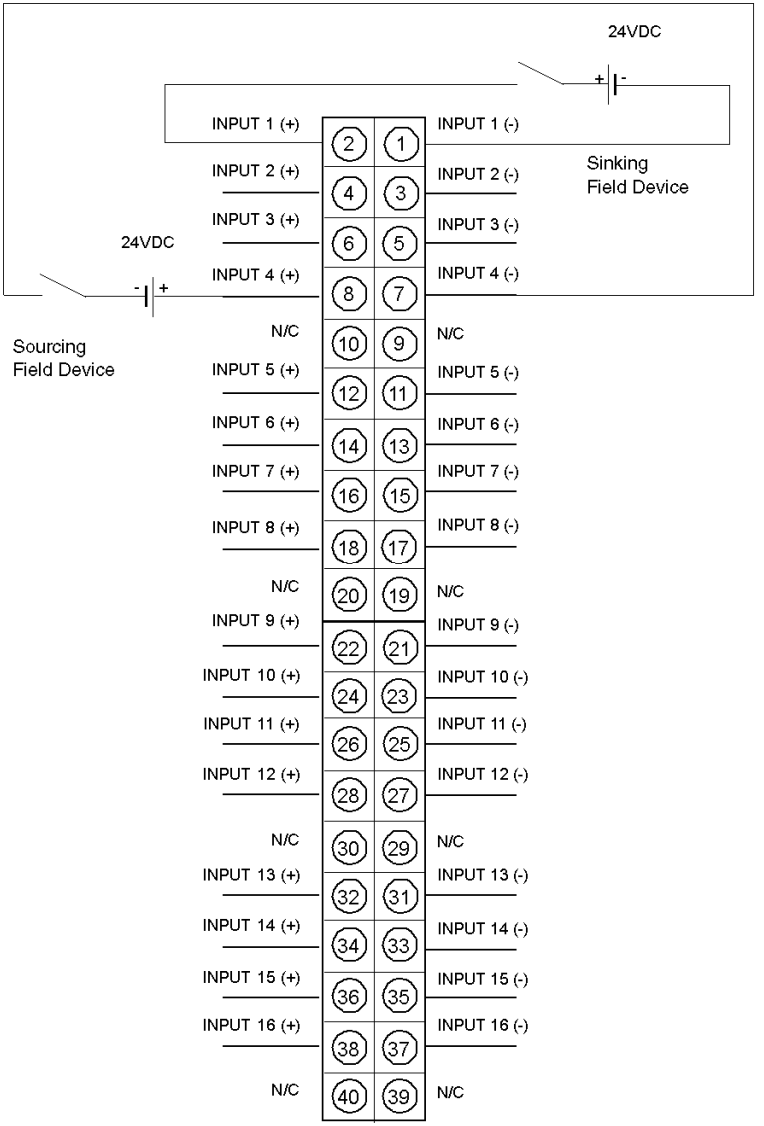
The following table shows the LED descriptions for the 140 HLI 340 00 module.

LEDs	Color	Indication when ON
Active	Green	Bus communication is present.
1 ... 16	Green	The indicated point or channel is turned ON.

NOTE: Due to the speed of the module, LED indications do not represent the state of the input signal, when the input signal is a short duration pulse.

Wiring Diagram

Illustration



NOTE: Either shielded or unshielded signal cables may be used (the user should consider using shielded wire in a noisy environment). Shielded types should have a shield tied to earth ground near the signal ground.

N / C = Not Connected.

NOTE: The tightening torque must be between 0.5 Nm and 0.8 Nm.

NOTICE

DESTRUCTION OF ADAPTER

- Before tightening the locknut to the torque 0.50...0.80 Nm, be sure to properly position the right-angle F adapter connector.
- During tightening, be sure to maintain the connector securely.
- Do not tighten the right-angle F adapter beyond the specified torque.

Failure to follow these instructions can result in equipment damage.

Specifications

General Specifications

General Specifications

Module Type	16 IN individually isolated
External Power	Not required for this module
Power Dissipation	2.0 W/0.30 W x the number of points ON
Bus Current required	400 mA
I/O map	1 input word
Fault Detection	None

Indicators

Indicators

LEDs	Active 1...16 (green)-indicates point status

Operating Voltage and Input Current

Operating Voltage and Input Current

ON (voltage)	15...30 Vdc
OFF (voltage)	-3...+5 Vdc
ON (current)	2.0...8.0 mA
OFF (current)	0...0.5 mA

Maximum Input

Absolute Maximum Input Voltage

Continuous	30 Vdc

Input Protection

Input Protection

30 Vdc reverse polarity (diode protection)

Isolation

Isolation

Point to Point	500 Vac rms for 1 minute
Point to Bus	1780 Vac rms for 1 minute

Response

Response

OFF - ON	30 µs (max)
ON - OFF	130 µs (max)

Fuses

Fuses

Internal	None
External	User discretion

Module Keying

Field Wiring Terminal Strip/Module Keying

Module Coding	ABE
Terminal Strip Coding	CDF

Quantum Automation Series System Specifications

Overview

This section presents the specifications for the Quantum Automation Series.

Specification Table

All Quantum Automation Series modules are designed to the following system specifications. Also included in this section are agency approvals.

Mechanical

Mechanical

Weight	2 lbs (1 kg) max
Dimensions (H x D x W)	9.84 in. x 4.09 in. x 1.59 in (250 mm x 103.85 mm x 40.34 mm)
Wire Size	1—14 AWG or 2—16 AWG max; 20 AWG min
Material (Enclosures and Bezels)	Lexan
Space Requirements	1 backplane slot

Electrical

Electrical

RFI Immunity (IEC 801-3)	27...500 MHz, 10 V/m
Electrostatic discharge (IEC 801-2)	8 kV air/4 kV contact

Modules with Operating Voltages Less than 24 Vac or Vdc

Function I/O Modules with Operating Voltages Less than 24 Vac or Vdc

Fast Transients (IEC 801-4)	0.5 kV common mode
Damped Oscillatory Transients	1 kV common mode; 0.5 kV differential mode
Surge Withstand Capability (Transients) (IEC 801-5)	1 kV common mode; 0.5 kV differential mode

Operating Conditions

Operating Conditions

Temperature	0...60° C (32...140° F)
Humidity	0..95% RH noncondensing @ 60° C
Chemical Interactions	Enclosures and bezels are made of Lexan, a polycarbonate that can be damaged by strong alkaline solutions
Altitude	2000 meters full operation
Vibration	10...57 Hz @ 0.075 mm d.a.; 57...150 Hz @ 1 g
Shock	15 g peak, 11 ms, half-sine wave

Storage Conditions

Storage Conditions

Temperature	-40...85° C (-40...185° F)
Humidity	0...95% RH noncondensing @ 60° C
Free Fall	3 ft (1 m)

Agency Approvals

Agency Approvals

UL 508
CSA 22.2-142
Factory Mutual Class I, Div. 2
European Directive on EMC 89/336/EEC

Part III
Configuration

Introduction

The following part provides information of the 140 HLI 340 00 configuration and some aspects of interrupt programming and direct I/O under Unity Pro.

What Is in This Part?

This part contains the following chapters:

Chapter 5
Quantum Addressing Modes

Overview

In the functional description of this expert module, the %IW/%MW (3x/4x) register addressing mode established in the Quantum world is widely used. This chapter describes the different modes used in Unity Pro to address the data from a Quantum module.

NOTE: Topological addresses overlapping (%IWr.m.c) is not supported by Quantum application, use flat addressing (%IWx) when memory overlapping control is needed.

What Is in This Chapter?

This chapter contains the following topics:

Flat Addressing—800 Series I/O Modules

Introduction

800 series I/O modules follow a system of flat address mapping in Unity Pro. To work properly. each module requires a determinate number of bits and/or words. The IEC addressing system is equivalent to the 984LL register addressing. Use the following assignments:

- 0x is now `%Mx`
- 1x is now `%Ix`
- 3x is now `%IWx`
- 4x is now `%MWx`

The following table shows the relationship between 984LL notation and IEC notation.

Outputs and Inputs	984LL Notation Register Addresses	IEC Notation		
		System Bits and Words	Memory Addresses	I/O Addresses
output	0x	System Bit	%Mx	%Qx
input	1x	System Bit	%Ix	%Ix
input	3x	System Word	%IWx	%IWx
output	4x	System Word	%MWx	%QWx

To access the I/O data of a module,

Step	Action
1	Enter the address range in the configuration screen.

Examples

The following examples show the relationship between 984LL register addressing and IEC addressing:

000001 is now `%M1`

100101 is now `%I101`

301024 is now `%IW1024`

400010 is now `%MW10`

Topological Addressing—800 Series I/O Modules with Unity

Accessing I/O Data Values

Use topological addressing to access I/O data items. Identify the topological location of the module within an 800 series I/O module with Unity Pro using the following notation:

```
%<Exchangetype><Objecttype>[\b.e\]r.m.c[.rank]
```

where:

- **b** = bus
- **e** = equipment (drop)
- **r** = rack
- **m** = module slot
- **c** = channel

NOTE: When addressing,

1. The [\b.e\] defaults to \1.1\ in a local rack and does not need to be specified.
2. The rank is an index used to identify different properties of an object with the same data type (value, warning level, error level).
3. The rank numbering is zero-based, and if the rank is zero, omit the entry.

For detailed information on I/O variables, please refer to the *Unity Pro Reference Manual*.

Reading Values: An Example

To read	Action
input value (rank = 0) from channel 7 of an analog module located in slot 6 of a local rack:	Enter `%IW1.6.7[.0]`
input value (rank = 0) from channel 7 of an analog module located in drop 3 of RIO bus 2:	Enter `%IW\2.3\1.6.7[.0]`
'out of range' value (rank = 1) from channel 7 of an analog module located in slot 6 of a local rack:	Enter `%I1.6.7.1[.0]`

Addressing Example

Comparing the 3 Addressing Modes

The following example compares the 3 possible addressing modes. An 8-channel thermocouple 140 ATI 030 00 module with the following configuration data is used:

- mounted in slot 5 of the CPU rack (local rack)
- starting input address is 201 (input word `%IW201`)
- end input address is 210 (input word `%IW210`)

To access the I/O data from the module you can use the following syntax:

Module data	Flat Addressing	Topological Addressing	IODDT Addressing	Concept Addressing
Channel 3 temperature	%IW203	%IW1.5.3	My_Temp.VALUE	300203
Channel 3 out of range	%IW209.5	%I1.5.3.1	My_Temp.ERROR	300209 Bit 5 to be extracted by user logic
Channel 3 range warning	%IW209.13	%I1.5.3.2	My_Temp.WARNING	300209 Bit 13 to be extracted by user logic
Module internal temperature	%IW210	%IW1.5.10	not accessible through IODDT	300210

NOTE: For the IODDT the data type `T_ANA_IN_VWE` is used and the variable `My_Temp` with the address `%CH1.5.10` was defined.

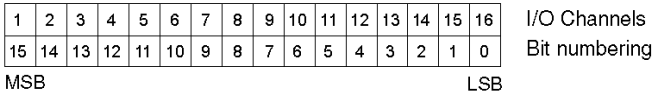
For comparison, the register addressing as used with Concept is added in the last column. As Concept does not support direct addressing of a bit in a word, the bit extraction has to be performed in the user program.

Discrete I/O Bit Numbering

Introduction

The numbering of channels of an I/O module usually starts with 1 and counts up to the maximum number of supported channels. The software however starts numbering with a 0 for the least significant bit in a word (LSB). The Quantum I/O modules have their lowest channel mapped to the most significant bit (MSB).

The following figure shows the mapping of I/O channels related to the bits in a word:.



Word Addressing Versus Bit Addressing

Mainly discrete I/O modules can be configured to deliver their I/O data either in word format or in bit format. This can be selected during configuration by selecting either `%IW` (`%MW`) or `%I` (`%M`). If you need to access a single bit from an I/O module configured to use an I/O word, you can use the syntax `%word.bit`. The following table gives you the connection between I/O point number and the associated I/O address in bit and word addressing.

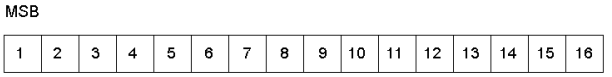
The table shows a 32-point input module in the main rack, slot 4 configured with starting address `%I1` or `%IW1`:

I/O channel	Bit address (flat addressing)	Bit address (topological addressing)	Bit address extracted from word (flat addressing)	Bit address extracted from word (topological addressing)
1	%I1	%I1.4.1[.0]	%IW1.15	%IW1.4.1.1.15
2	%I2	%I1.4.2[.0]	%IW1.14	%IW1.4.1.1.14
3	%I3	%I1.4.3[.0]	%IW1.13	%IW1.4.1.1.13
• • •				
15	%I15	%I1.4.15[.0]	%IW1.1	%IW1.4.1.1.1
16	%I16	%I1.4.16[.0]	%IW1.0	%IW1.4.1.1.0
17	%I17	%I1.4.17[.0]	%IW2.15	%IW1.4.1.2.15
18	%I18	%I1.4.18[.0]	%IW2.14	%IW1.4.1.2.14
• • •				
31	%I31	%I1.4.31[.0]	%IW2.1	%IW1.4.1.2.1
32	%I32	%I1.4.32[.0]	%IW2.0	%IW1.4.1.2.0

Addressing

Flat Addressing

This module requires either 16 contiguous %I references or one %IW word. For a description of how to access the input points, please refer to *Discrete I/O Bit Numbering, page 47*.



Topological Addressing

The following tables show the topological addresses for the 140HLI34000 module.

Topological addresses in Bit Mapping format:

Point	I/O Object	Comment
Input 1	%I[\b.e\]r.m.1	Value
Input 2	%I[\b.e\]r.m.2	Value
• • •		
Input 15	%I[\b.e\]r.m.15	Value
Input 16	%I[\b.e\]r.m.16	Value

Topological addresses in Word Mapping format:

Point	I/O Object	Comment
Inputword 1	%IW[\b.e\]r.m.1.1	Value

Used abbreviations: **b** = bus, **e** = equipment (drop), **r** = rack, **m** = module slot.

I/O Map Status Byte

There is no I/O map status byte associated with this module.

Chapter 6
Configuration of the 140 HLI 340 00 Module

Purpose

The following chapter provides information of the configuration of 140 HLI 340 00.

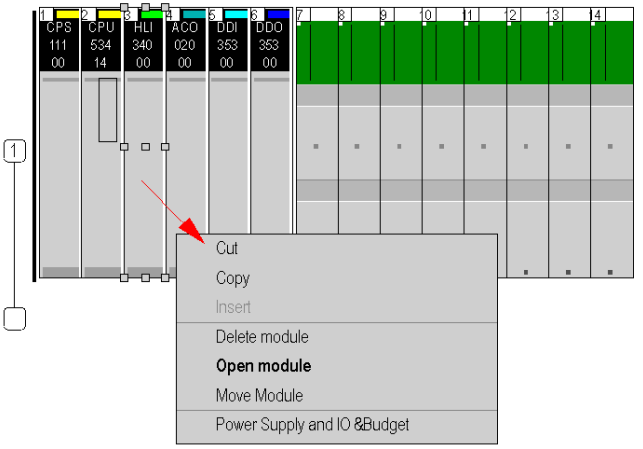
What Is in This Chapter?

This chapter contains the following topics:

Open the Parameter Configuration

Procedure

This table shows you how to open the parameter configuration.

Step	Action
1	Call Bus Editor
2	Select the module
3	Click the right mouse button **Result:** A shortcut menu is opened 
4	Select **Open module** **Result:** The module is opened with a window for parameter configuration

Parameter Configuration

Module Placement

The 140 HLI 340 00 High Speed Interrupt module's functionality depends on its location.

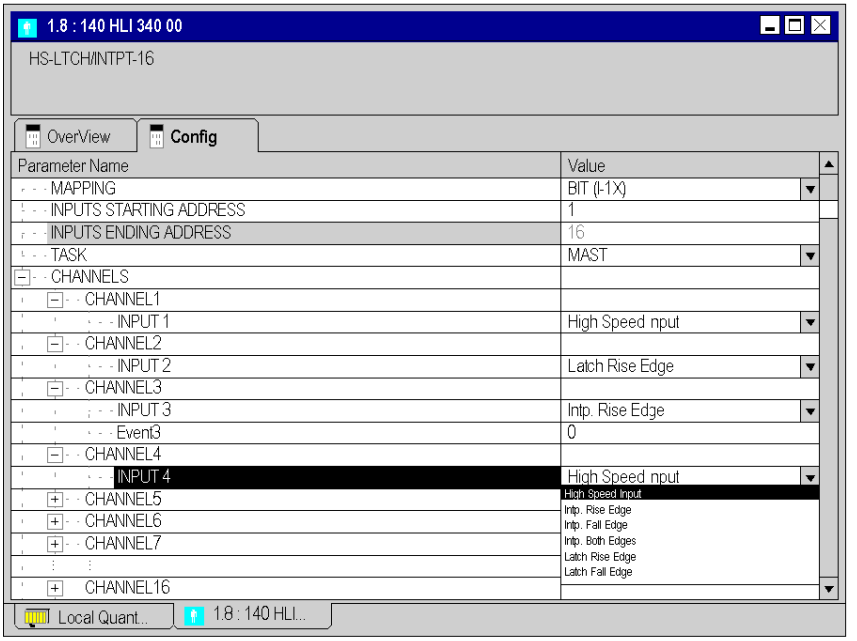
The following list shows the available modes depending on the location:

- **Local Rack:** High Speed Inputs, Latch Inputs, Interrupts,
- **Remote Rack:** High Speed Inputs.

NOTE: The 140 HLI 340 00 module cannot be connected to a secondary rack.

Parameters and Default Values

Parameter Configuration Window



Parameter Description

Name	Default Value	Options	Description
Mapping	BIT (%I-1x)	WORD (%IW-3X)	
Inputs Starting Address	1		
Inputs Ending Address	16		
Task	MAST	FAST AUX0 AUX1 AUX2 AUX3	
Channels			
Input n	High Speed Input	Intp. Rise Edge Intp. Fall Edge Intp. Both Edges Latch Rise Edge Latch Fall Edge	Interrupt modes only available in local rack, latch only in local- and expansion rack
Event n	0	1 - 127	

Chapter 7
Event- and Direct I/O Configuration

Introduction

This chapter provides basic information on how to configure I/O events and use direct I/O under Unity Pro.

What Is in This Chapter?

This chapter contains the following topics:

Event Configuration

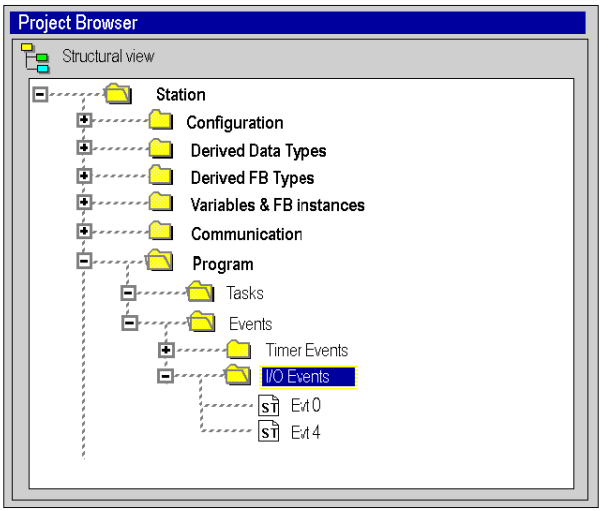
Overview

To make use of the interrupt capabilities of the HLI module you have to define I/O events in Unity. For an overview on the task and event management refer to the chapter *Interrupt Handling with Unity Pro, page 13*.

Defining an Event

To define an event to be triggered by the HLI module select from the Project browser the **I/O Events** node and open a new event section (Evt 0 ... Evt 127).

The I/O Events in the Project Browser:



In the dialog you have to enter the number of the event and the programming language you want to use for your interrupt routine.

Programming of Events

Once you have defined an event you can open the event section and write your interrupt routine in the same way as any other section in your project.

Link an Event to an Input

The link to your hardware input is entered in the configuration dialog for the HLI module (see: *Parameters and Default Values, page 51*).

Direct I/O Function Block IMIO_IN

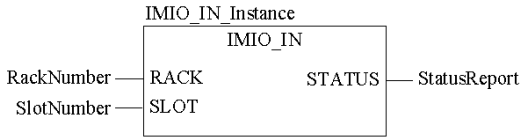
Function description

This function block reads I/O module signals immediately during processing. The input module must be in the local rack of the PLC.

`EN` and `ENO` can be configured as additional parameters.

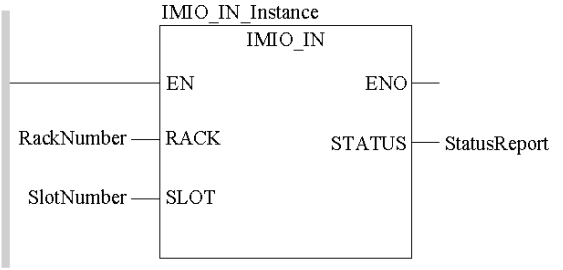
Representation in FBD

Representation:



Representation in LD

Representation:



Representation in IL

Representation:

```
CAL IMIO_IN_Instance (RACK:=RackNumber, SLOT:=SlotNumber,
    STATUS=>StatusReport)
```

Representation in ST

Representation:

```
IMIO_IN_Instance (RACK:=RackNumber, SLOT:=SlotNumber,
    STATUS=>StatusReport) ;
```

Parameter description

Description of the input parameters:

Parameter	Data type	Meaning
RACK	INT	Rack number (Quantum: 1)
SLOT	INT	Slot number (Quantum: 1...16)

Description of the output parameters:

Parameter	Data type	Meaning
STATUS	WORD	Status report

Runtime error

The ENO parameter can be used for error display:

ENO	Meaning
1	Operation OK (STATUS equals "0")
0	Operation faulty (STATUS not equal to "0")

Detailed description

Detailed description

The input of signals takes place directly during block processing as well as during normal I/O processing at the end of a cycle.

The input module must be in the local rack of the PLC. It must also be entered into the I/O map of its configuration. The I/O module is addressed using rack number and slot number.

Parameter description

The STATUS parameter may contain the following messages:

Status	Meaning
0000	Operation OK
2001	invalid operation type (e.g. the I/O module addressed is not an input module)
2002	Invalid rack or slot number (I/O map in the configurator contains no module entry for this slot)
2003	invalid slot number
F001	Module not OK

Direct I/O Function Block IMIO_OUT

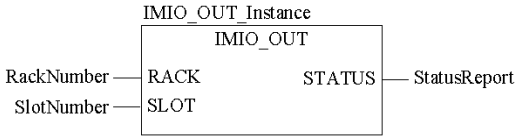
Function description

This function block supplies the I/O module signals immediately during processing. The output module must be in the local rack of the PLC.

`EN` and `ENO` can be configured as additional parameters.

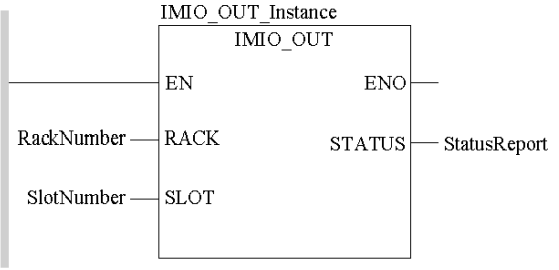
Representation in FBD

Representation:



Representation in LD

Representation:



Representation in IL

Representation:

```
CAL IMIO_OUT_Instance (RACK:=RackNumber, SLOT:=SlotNumber,
    STATUS=>StatusReport)
```

Representation in ST

Representation:

```
IMIO_OUT_Instance (RACK:=RackNumber, SLOT:=SlotNumber,
    STATUS=>StatusReport) ;
```

Parameter description

Description of the input parameters:

Parameter	Data type	Meaning
RACK	INT	Rack number (Quantum: 1)
SLOT	INT	Slot number (Quantum: 1...16)

Description of the output parameters:

Parameter	Data type	Meaning
STATUS	WORD	Status report

Runtime error

The ENO parameter can be used for error display:

ENO	Meaning
1	Operation OK (STATUS equals "0")
0	Operation faulty (STATUS equals "0")

Detailed description

Detailed description

The output of signals takes place immediately during block processing as well as during normal I/O processing at the end of a cycle.

The output module must be in the local rack of the PLC. It must also be entered into the I/O map of its configuration. The I/O module is addressed using rack number and slot number.

Parameter description

Status report STATUS

The `STATUS` parameter may contain the following messages:

Status	Meaning
0000	Operation OK
2001	invalid operation type (e.g. the I/O module addressed is not an input module)
2002	Invalid rack or slot number (I/O map in the configurator contains no module entry for this slot)
2003	invalid slot number
F001	Module not OK

Index