Honeywell UDC Ethernet Driver

Table of Contents

Honeywell UDC Ethernet Driver

Help version 1.026

CONTENTS

Overview

The Honeywell UDC Ethernet Driver provides a reliable way to connect Honeywell UDC Ethernet controllers to OPC client applications; including HMI, SCADA, Historian, MES, ERP, and countless custom applications. It is intended for use with Honeywell UDC controllers.

Setup

Supported Devices

UDC 2500
UDC 3200
UDC 3500

Communication Protocol

Modbus TCP/IP using Winsock V1.1 or higher

Connect Timeout

This parameter specifies the time that the driver will wait for a connection to be made with a device. Depending on network load the connect time may vary with each connection attempt. The default setting is 3 seconds. The valid range is 1 to 30 seconds.

Request Timeout

This parameter specifies the time that the driver will wait for a response from the device before giving up and going on to the next request. Longer timeouts only affect performance if a device is not responding. The default setting is 1000 milliseconds. The valid range is 100 to 30000 milliseconds.

Fail After *x* Successive Timeouts

This parameter specifies the number of times that the driver will retry a message before giving up and going on to the next message. The default setting is 3 retries. The valid range is 1 to 10.

Inter-Request Delay

The inter-request delay's default is 50 milliseconds. For more information on the inter-request delay setting, refer to the OPC Server Help documentation.

Note: The manufacturer of Honeywell UDC devices requires that the Inter-request Delay be set to 200 milliseconds or higher.

Channel and Device Limits

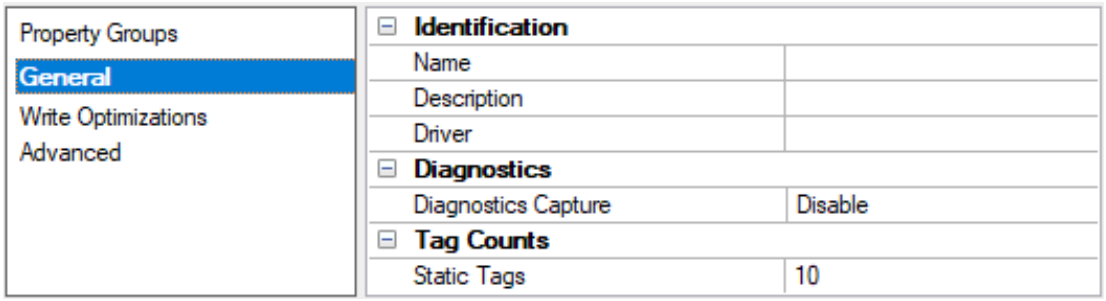
The maximum number of channels supported by this driver is 100. The maximum number of devices supported by this driver is 256 per channel.

Device ID (PLC Network Address)

The Device ID is used to specify the device IP in standard YYY.YYY.YYY.YYY format.

Channel Properties — General

This server supports the use of multiple simultaneous communications drivers. Each protocol or driver used in a server project is called a channel. A server project may consist of many channels with the same communications driver or with unique communications drivers. A channel acts as the basic building block of an OPC link. This group is used to specify general channel properties, such as the identification attributes and operating mode.



Identification

Name: Specify the user-defined identity of this channel. In each server project, each channel name must be unique. Although names can be up to 256 characters, some client applications have a limited display window when browsing the OPC server's tag space. The channel name is part of the OPC browser information. The property is required for creating a channel.

For information on reserved characters, refer to "How To... Properly Name a Channel, Device, Tag, and Tag Group" in the server help.

Description: Specify user-defined information about this channel.

Many of these properties, including Description, have an associated system tag.

Driver: Specify the protocol / driver for this channel. Specify the device driver that was selected during channel creation. It is a disabled setting in the channel properties. The property is required for creating a channel.

Note: With the server's online full-time operation, these properties can be changed at any time. This includes changing the channel name to prevent clients from registering data with the server. If a client has already acquired an item from the server before the channel name is changed, the items are unaffected. If, after the channel name has been changed, the client application releases the item and attempts to re-acquire using the old channel name, the item is not accepted. Changes to the properties should not be made once a large client application has been developed. Utilize proper user role and privilege management to prevent operators from changing properties or accessing server features.

Diagnostics

Diagnostics Capture: When enabled, this option makes the channel's diagnostic information available to OPC applications allows the usage of statistics tags that provide feedback to client applications regarding the operation of the channel. Because the server's diagnostic features require a minimal amount of overhead processing, it is recommended that they be utilized when needed and disabled when not. The default is disabled.

Note: This property is not available if the driver does not support diagnostics.

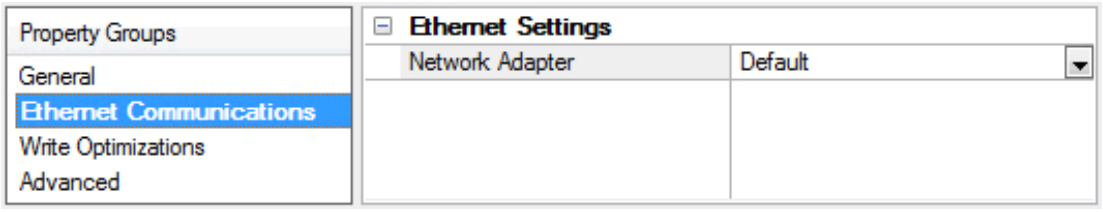
For more information, refer to "Communication Diagnostics" and "Statistics Tags" in the server help.

Tag Counts

Static Tags: Provides the total number of defined static tags at this level (device or channel). This information can be helpful in troubleshooting and load balancing.

Channel Properties — Ethernet Communications

Ethernet Communication can be used to communicate with devices.

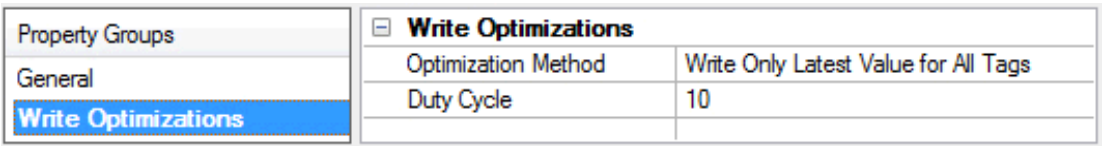


Ethernet Settings

Network Adapter: Specify the network adapter to bind. When left blank or Default is selected, the operating system selects the default adapter.

Channel Properties — Write Optimizations

The server must ensure that the data written from the client application gets to the device on time. Given this goal, the server provides optimization properties to meet specific needs or improve application responsiveness.



Write Optimizations

Optimization Method: Controls how write data is passed to the underlying communications driver. The options are:

- **Write All Values for All Tags**: This option forces the server to attempt to write every value to the controller. In this mode, the server continues to gather write requests and add them to the server's internal write queue. The server processes the write queue and attempts to empty it by writing data to the device as quickly as possible. This mode ensures that everything written from the client applications is sent to the target device. This mode should be selected if the write operation order or the write item's content must uniquely be seen at the target device.
- **Write Only Latest Value for Non-Boolean Tags**: Many consecutive writes to the same value can accumulate in the write queue due to the time required to actually send the data to the device. If the server updates a write value that has already been placed in the write queue, far fewer writes are needed to reach the same final output value. In this way, no extra writes accumulate in the server's queue. When the user stops moving the slide switch, the value in the device is at the correct value at virtually the same time. As the mode states, any value that is not a Boolean value is updated in the server's internal write queue and sent to the device at the next possible opportunity. This can greatly improve the application performance.
  Note: This option does not attempt to optimize writes to Boolean values. It allows users to optimize the operation of HMI data without causing problems with Boolean operations, such as a momentary push button.
- **Write Only Latest Value for All Tags**: This option takes the theory behind the second optimization mode and applies it to all tags. It is especially useful if the application only needs to send the latest

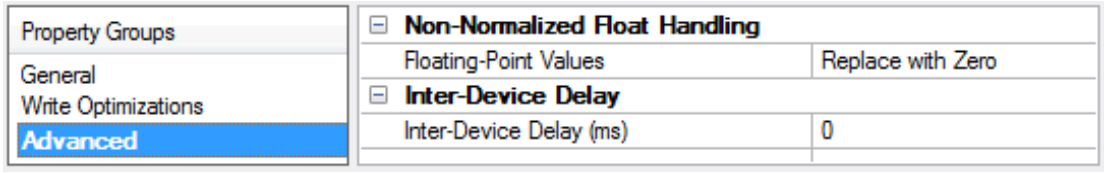
value to the device. This mode optimizes all writes by updating the tags currently in the write queue before they are sent. This is the default mode.

Duty Cycle: is used to control the ratio of write to read operations. The ratio is always based on one read for every one to ten writes. The duty cycle is set to ten by default, meaning that ten writes occur for each read operation. Although the application is performing a large number of continuous writes, it must be ensured that read data is still given time to process. A setting of one results in one read operation for every write operation. If there are no write operations to perform, reads are processed continuously. This allows optimization for applications with continuous writes versus a more balanced back and forth data flow.

Note: It is recommended that the application be characterized for compatibility with the write optimization enhancements before being used in a production environment.

Channel Properties — Advanced

This group is used to specify advanced channel properties. Not all drivers support all properties; so the Advanced group does not appear for those devices.



Non-Normalized Float Handling: A non-normalized value is defined as Infinity, Not-a-Number (NaN), or as a Denormalized Number. The default is Replace with Zero. Drivers that have native float handling may default to Unmodified. Non-normalized float handling allows users to specify how a driver handles non-normalized IEEE-754 floating point data. Descriptions of the options are as follows:

- **Replace with Zero**: This option allows a driver to replace non-normalized IEEE-754 floating point values with zero before being transferred to clients.
- **Unmodified**: This option allows a driver to transfer IEEE-754 denormalized, normalized, non-number, and infinity values to clients without any conversion or changes.

Note: This property is disabled if the driver does not support floating-point values or if it only supports the option that is displayed. According to the channel's float normalization setting, only real-time driver tags (such as values and arrays) are subject to float normalization. For example, EFM data is not affected by this setting.

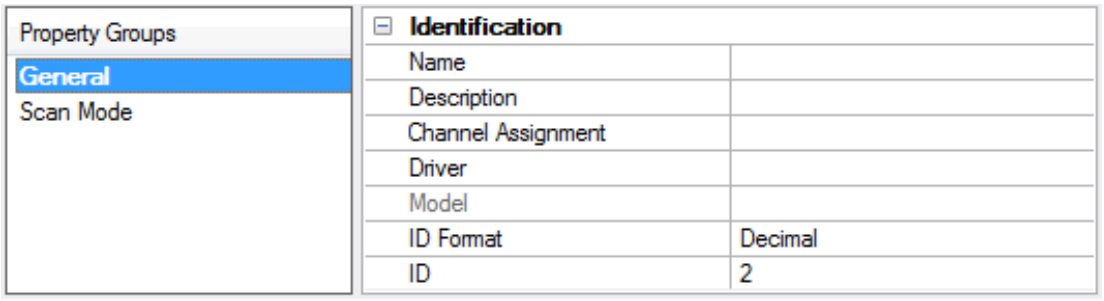
For more information on the floating-point values, refer to "How To ... Work with Non-Normalized Floating-Point Values" in the server help.

Inter-Device Delay: Specify the amount of time the communications channel waits to send new requests to the next device after data is received from the current device on the same channel. Zero (0) disables the delay.

Note: This property is not available for all drivers, models, and dependent settings.

Device Properties — General

A device represents a single target on a communications channel. If the driver supports multiple controllers, users must enter a device ID for each controller.



Identification

Name: Specify the name of the device. It is a logical user-defined name that can be up to 256 characters long and may be used on multiple channels.

Note: Although descriptive names are generally a good idea, some OPC client applications may have a limited display window when browsing the OPC server's tag space. The device name and channel name become part of the browse tree information as well. Within an OPC client, the combination of channel name and device name would appear as "ChannelName.DeviceName".
For more information, refer to "How To... Properly Name a Channel, Device, Tag, and Tag Group" in server help.

Description: Specify the user-defined information about this device.
Many of these properties, including Description, have an associated system tag.

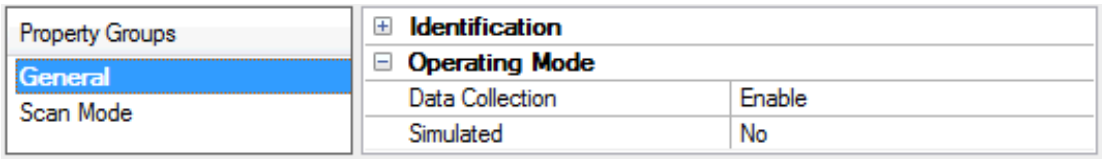
Channel Assignment: Specify the user-defined name of the channel to which this device currently belongs.

Driver: Selected protocol driver for this device.

Model: Specify the type of device that is associated with this ID. The contents of the drop-down menu depend on the type of communications driver being used. Models that are not supported by a driver are disabled. If the communications driver supports multiple device models, the model selection can only be changed when there are no client applications connected to the device.
Note: If the communication driver supports multiple models, users should try to match the model selection to the physical device. If the device is not represented in the drop-down menu, select a model that conforms closest to the target device. Some drivers support a model selection called "Open," which allows users to communicate without knowing the specific details of the target device. *For more information, refer to the driver documentation.*

ID: Specify the device's driver-specific station or node. The type of ID entered depends on the communications driver being used. For many communication drivers, the ID is a numeric value. Drivers that support a Numeric ID provide users with the option to enter a numeric value whose format can be changed to suit the needs of the application or the characteristics of the selected communications driver. The format is set by the driver by default. Options include Decimal, Octal, and Hexadecimal.
Note: If the driver is Ethernet-based or supports an unconventional station or node name, the device's TCP/IP address may be used as the device ID. TCP/IP addresses consist of four values that are separated by periods, with each value in the range of 0 to 255. Some device IDs are string based. There may be additional properties to configure within the ID field, depending on the driver.

Operating Mode



Data Collection: This property controls the device's active state. Although device communications are enabled by default, this property can be used to disable a physical device. Communications are not attempted when a device is disabled. From a client standpoint, the data is marked as invalid and write operations are not accepted. This property can be changed at any time through this property or the device system tags.

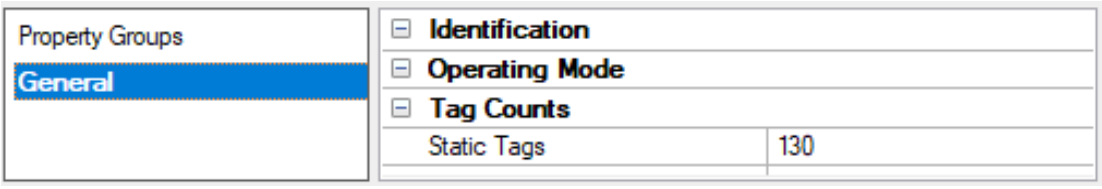
Simulated: Place the device into or out of Simulation Mode. In this mode, the driver does not attempt to communicate with the physical device, but the server continues to return valid OPC data. Simulated stops physical communications with the device, but allows OPC data to be returned to the OPC client as valid data. While in Simulation Mode, the server treats all device data as reflective: whatever is written to the simulated device is read back and each OPC item is treated individually. The data is not saved if the server removes the item (such as when the server is reinitialized). The default is No.

Notes:

1. This System tag (_Simulated) is read only and cannot be written to for runtime protection. The System tag allows this property to be monitored from the client.
2. When a device is simulated, updates may not appear faster than one (1) second in the client.

Simulation Mode is for test and simulation purposes only. It should never be used in a production environment.

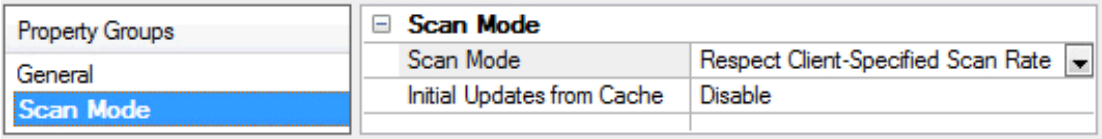
Tag Counts



Static Tags: Provides the total number of defined static tags at this level (device or channel). This information can be helpful in troubleshooting and load balancing.

Device Properties — Scan Mode

The Scan Mode specifies the subscribed-client requested scan rate for tags that require device communications. Synchronous and asynchronous device reads and writes are processed as soon as possible; unaffected by the Scan Mode properties.



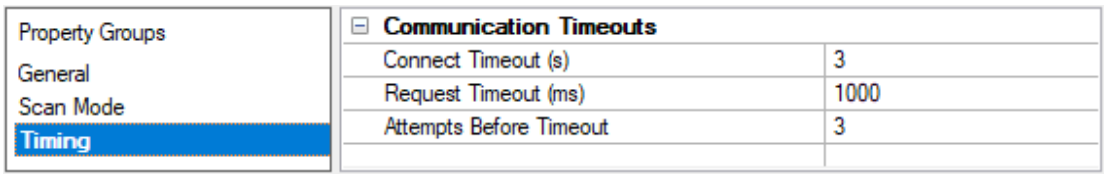
Scan Mode: Specify how tags in the device are scanned for updates sent to subscribing clients. Descriptions of the options are:

- **Respect Client-Specified Scan Rate**: This mode uses the scan rate requested by the client.
- **Request Data No Faster than Scan Rate**: This mode specifies the value set as the maximum scan rate. The valid range is 10 to 99999990 milliseconds. The default is 1000 milliseconds.
  Note: When the server has an active client and items for the device and the scan rate value is increased, the changes take effect immediately. When the scan rate value is decreased, the changes do not take effect until all client applications have been disconnected.
- **Request All Data at Scan Rate**: This mode forces tags to be scanned at the specified rate for subscribed clients. The valid range is 10 to 99999990 milliseconds. The default is 1000 milliseconds.
- **Do Not Scan, Demand Poll Only**: This mode does not periodically poll tags that belong to the device nor perform a read to get an item's initial value once it becomes active. It is the OPC client's responsibility to poll for updates, either by writing to the _DemandPoll tag or by issuing explicit device reads for individual items. *For more information, refer to "Device Demand Poll" in server help*.
- **Respect Tag-Specified Scan Rate**: This mode forces static tags to be scanned at the rate specified in their static configuration tag properties. Dynamic tags are scanned at the client-specified scan rate.

Initial Updates from Cache: When enabled, this option allows the server to provide the first updates for newly activated tag references from stored (cached) data. Cache updates can only be provided when the new item reference shares the same address, scan rate, data type, client access, and scaling properties. A device read is used for the initial update for the first client reference only. The default is disabled; any time a client activates a tag reference the server attempts to read the initial value from the device.

Device Properties — Timing

The device Timing properties allow the driver's response to error conditions to be tailored to fit the application's needs. In many cases, the environment requires changes to these properties for optimum performance. Factors such as electrically generated noise, modem delays, and poor physical connections can influence how many errors or timeouts a communications driver encounters. Timing properties are specific to each configured device.



Communications Timeouts

Connect Timeout: This property (which is used primarily by Ethernet based drivers) controls the amount of time required to establish a socket connection to a remote device. The device's connection time often takes longer than normal communications requests to that same device. The valid range is 1 to 30 seconds. The default is typically 3 seconds, but can vary depending on the driver's specific nature. If this setting is not supported by the driver, it is disabled.
Note: Due to the nature of UDP connections, the connection timeout setting is not applicable when communicating via UDP.

Request Timeout: Specify an interval used by all drivers to determine how long the driver waits for a response from the target device to complete. The valid range is 50 to 9999999 milliseconds (167 minutes). The default is usually 1000 milliseconds, but can vary depending on the driver. The default timeout for most

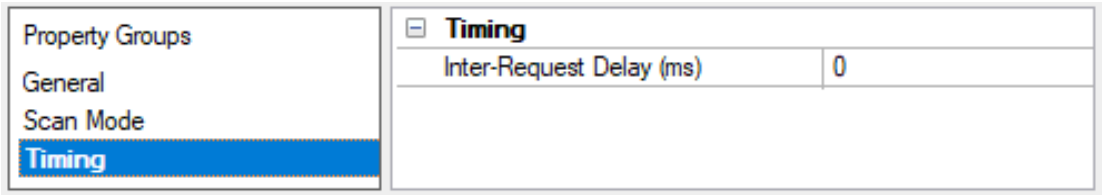
serial drivers is based on a baud rate of 9600 baud or better. When using a driver at lower baud rates, increase the timeout to compensate for the increased time required to acquire data.

Attempts Before Timeout: Specify how many times the driver issues a communications request before considering the request to have failed and the device to be in error. The valid range is 1 to 10. The default is typically 3, but can vary depending on the driver's specific nature. The number of attempts configured for an application depends largely on the communications environment. This property applies to both connection attempts and request attempts.

Timing

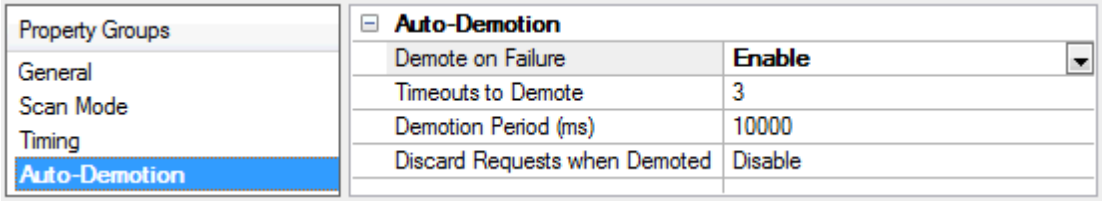
Inter-Request Delay: Specify how long the driver waits before sending the next request to the target device. It overrides the normal polling frequency of tags associated with the device, as well as one-time reads and writes. This delay can be useful when dealing with devices with slow turnaround times and in cases where network load is a concern. Configuring a delay for a device affects communications with all other devices on the channel. It is recommended that users separate any device that requires an inter-request delay to a separate channel if possible. Other communications properties (such as communication serialization) can extend this delay. The valid range is 0 to 300,000 milliseconds; however, some drivers may limit the maximum value due to a function of their particular design. The default is 0, which indicates no delay between requests with the target device.

Note: Not all drivers support Inter-Request Delay. This setting does not appear if it is not available.



Device Properties — Auto-Demotion

The Auto-Demotion properties can temporarily place a device off-scan in the event that a device is not responding. By placing a non-responsive device offline for a specific time period, the driver can continue to optimize its communications with other devices on the same channel. After the time period has been reached, the driver re-attempts to communicate with the non-responsive device. If the device is responsive, the device is placed on-scan; otherwise, it restarts its off-scan time period.



Demote on Failure: When enabled, the device is automatically taken off-scan until it is responding again.

Tip: Determine when a device is off-scan by monitoring its demoted state using the _AutoDemoted system tag.

Timeouts to Demote: Specify how many successive cycles of request timeouts and retries occur before the device is placed off-scan. The valid range is 1 to 30 successive failures. The default is 3.

Demotion Period: Indicate how long the device should be placed off-scan when the timeouts value is reached. During this period, no read requests are sent to the device and all data associated with the read

requests are set to bad quality. When this period expires, the driver places the device on-scan and allows for another attempt at communications. The valid range is 100 to 3600000 milliseconds. The default is 10000 milliseconds.

Discard Requests when Demoted: Select whether or not write requests should be attempted during the off-scan period. Disable to always send write requests regardless of the demotion period. Enable to discard writes; the server automatically fails any write request received from a client and does not post a message to the Event Log.

Device Properties — Tag Generation

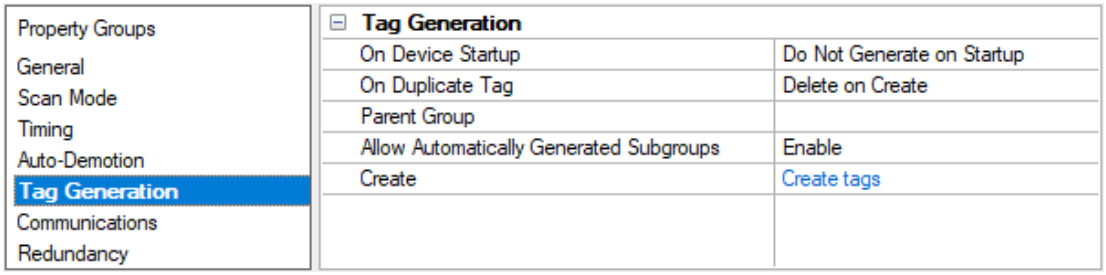
The automatic tag database generation features make setting up an application a plug-and-play operation. Select communications drivers can be configured to automatically build a list of tags that correspond to device-specific data. These automatically generated tags (which depend on the nature of the supporting driver) can be browsed from the clients.

Not all devices and drivers support full automatic tag database generation and not all support the same data types. Consult the data types descriptions or the supported data type lists for each driver for specifics.

If the target device supports its own local tag database, the driver reads the device's tag information and uses the data to generate tags within the server. If the device does not natively support named tags, the driver creates a list of tags based on driver-specific information. An example of these two conditions is as follows:

1. If a data acquisition system supports its own local tag database, the communications driver uses the tag names found in the device to build the server's tags.
2. If an Ethernet I/O system supports detection of its own available I/O module types, the communications driver automatically generates tags in the server that are based on the types of I/O modules plugged into the Ethernet I/O rack.

Note: Automatic tag database generation's mode of operation is completely configurable. *For more information, refer to the property descriptions below.*



On Property Change: If the device supports automatic tag generation when certain properties change, the **On Property Change** option is shown. It is set to **Yes** by default, but it can be set to **No** to control over when tag generation is performed. In this case, the **Create tags** action must be manually invoked to perform tag generation. To invoke via the Configuration API service, access /config/v1/project/channels/{name}/devices/{name}/services/TagGeneration.

On Device Startup: Specify when OPC tags are automatically generated. Descriptions of the options are as follows:

- **Do Not Generate on Startup**: This option prevents the driver from adding any OPC tags to the tag space of the server. This is the default setting.
- **Always Generate on Startup**: This option causes the driver to evaluate the device for tag information. It also adds tags to the tag space of the server every time the server is launched.
- **Generate on First Startup**: This option causes the driver to evaluate the target device for tag information the first time the project is run. It also adds any OPC tags to the server tag space as needed.

Note: When the option to automatically generate OPC tags is selected, any tags that are added to the server's tag space must be saved with the project. Users can configure the project to automatically save from the **Tools** | **Options** menu.

On Duplicate Tag: When automatic tag database generation is enabled, the server needs to know what to do with the tags that it may have previously added or with tags that have been added or modified after the communications driver since their original creation. This setting controls how the server handles OPC tags that were automatically generated and currently exist in the project. It also prevents automatically generated tags from accumulating in the server.

For example, if a user changes the I/O modules in the rack with the server configured to **Always Generate on Startup**, new tags would be added to the server every time the communications driver detected a new I/O module. If the old tags were not removed, many unused tags could accumulate in the server's tag space. The options are:

- **Delete on Create**: This option deletes any tags that were previously added to the tag space before any new tags are added. This is the default setting.
- **Overwrite as Necessary**: This option instructs the server to only remove the tags that the communications driver is replacing with new tags. Any tags that are not being overwritten remain in the server's tag space.
- **Do not Overwrite**: This option prevents the server from removing any tags that were previously generated or already existed in the server. The communications driver can only add tags that are completely new.
- **Do not Overwrite, Log Error**: This option has the same effect as the prior option, and also posts an error message to the server's Event Log when a tag overwrite would have occurred.

Note: Removing OPC tags affects tags that have been automatically generated by the communications driver as well as any tags that have been added using names that match generated tags. Users should avoid adding tags to the server using names that may match tags that are automatically generated by the driver.

Parent Group: This property keeps automatically generated tags from mixing with tags that have been entered manually by specifying a group to be used for automatically generated tags. The name of the group can be up to 256 characters. This parent group provides a root branch to which all automatically generated tags are added.

Allow Automatically Generated Subgroups: This property controls whether the server automatically creates subgroups for the automatically generated tags. This is the default setting. If disabled, the server generates the device's tags in a flat list without any grouping. In the server project, the resulting tags are named with the address value. For example, the tag names are not retained during the generation process.
Note: If, as the server is generating tags, a tag is assigned the same name as an existing tag, the system automatically increments to the next highest number so that the tag name is not duplicated. For example, if the generation process creates a tag named "AI22" that already exists, it creates the tag as "AI23" instead.

Create: Initiates the creation of automatically generated OPC tags. If the device's configuration has been modified, **Create tags** forces the driver to reevaluate the device for possible tag changes. Its ability to be accessed from the System tags allows a client application to initiate tag database creation.
Note: **Create tags** is disabled if the Configuration edits a project offline.

Automatic Tag Database Generation

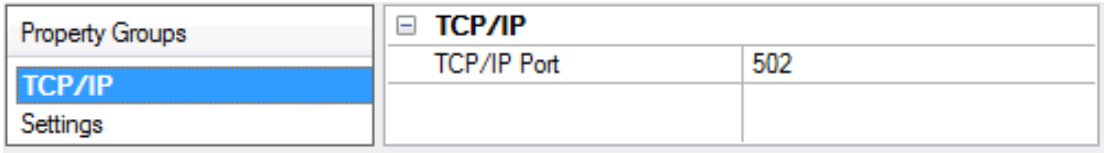
The Honeywell UDC Ethernet Driver can automatically create most of the tags needed for the application. To do so, double-click on the device for which tags will be generated, and then select the **Tag Generation** property group and click **Create Tags**. For a complete list of generated tags, see the addressing page for the specific device model.

UDC 2500 Generated Tags
UDC 3200 Generated Tags
UDC 3500 Generated Tags

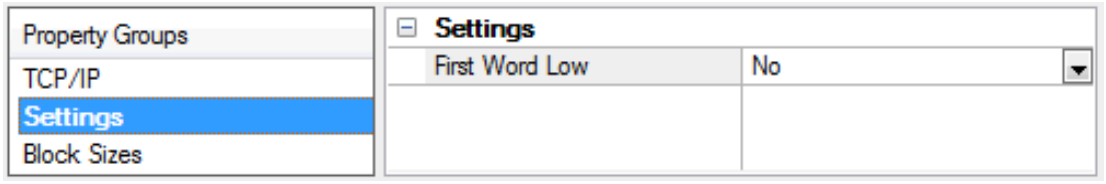
Note: Click **Apply** or **OK** before attempting to generate tags after making changes to the device model selection.

Device Properties — TCP/IP

This parameter specifies the TCP/IP port number that the remote device is configured to use. The default port number is 502.



Device Properties — Settings



First Word Low: Two consecutive register addresses are used for 32 bit data types such as floats. Users can specify whether the driver should treat the contents of the first register as the low or high word in 32 bit values.

Note: The UDC units can be configured to use a number of Double Register Formats.

Double Register Format

Format	Description	Byte Order	Notes
FP B	Floating Point Big Endian	4, 3, 2, 1	Honeywell default
FP BB	Floating Point Big Endian with byte-swap	3, 4, 1, 2	
FP L	Floating Point Little Endian	1, 2, 3, 4	
FP LB	Floating Point Little Endian with byte-swap	2, 1, 4, 3	Modbus standard

Examples of Data in "FP B" Format

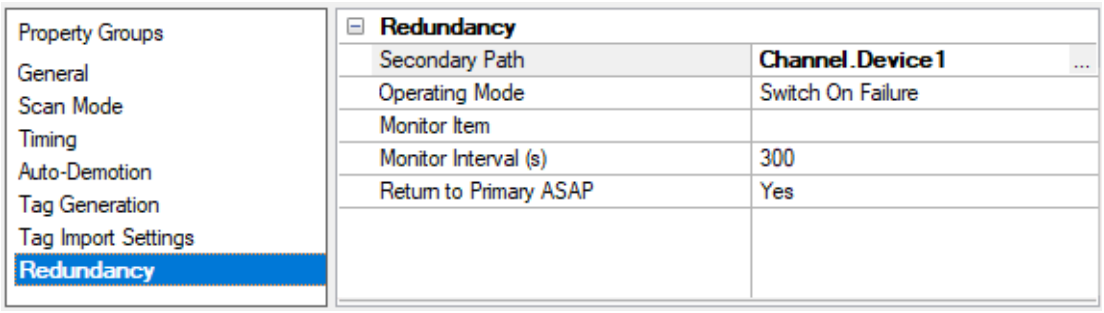
Value (decimal)	Value (hex)	Register N		Register N+1	
		high	low	high	low
100.0	0x42C80000	0x42	0xC8	0x00	0x00
55.32	0x425D47AE	0x42	0x5D	0x47	0xAE
2.0	0x40000000	0x40	0x00	0x00	0x00
1.0	0x3F800000	0x3F	0x80	0x00	0x00
-1.0	0xBF800000	0xBF	0x80	0x00	0x00

The driver will use the Honeywell default "FP B" if this device property is set to No. If set to Yes, the "FP LB" format will be used. The driver does not currently support the Honeywell "FP BB" and "FP L" double register formats.

Device Properties — Block Sizes

Registers: Registers can be read from 1 to 22 locations at a time. Given the overhead involved in sending data via TCP/IP, it is generally advantageous to keep the block size large. However, if data will be read from non-contiguous locations within the device, reducing the block size may increase performance.

Device Properties — Redundancy



Redundancy is available with the Media-Level Redundancy Plug-In.

*Consult the website, a sales representative, or the **user manual** for more information.*

Data Types Description

Data Type	Description
Boolean	Single bit
Word	Unsigned 16-bit value bit 0 is the low bit bit 15 is the high bit
Short	Signed 16-bit value bit 0 is the low bit bit 14 is the high bit bit 15 is the sign bit
DWord	Unsigned 32-bit value bit 0 is the low bit bit 31 is the high bit
Long	Signed 32-bit value bit 0 is the low bit bit 30 is the high bit bit 31 is the sign bit
Float	32-bit floating point value. The driver interprets two consecutive registers as a floating-point value by making the second register the high word and the first register the low word.

Address Descriptions

Address specifications vary depending on the model in use. Select a link from the following list to obtain specific address information for the model of interest.

UDC 2500
UDC 3200
UDC 3500

UDC 2500 Addressing

The default data types for dynamically defined tags are shown in **bold** where appropriate.

Modbus Addressing Decimal Format

Memory Type	Range	Data Type	Access
Output Coils	000001-000008	**Boolean**	Read/Write
Input Coils	100001-100008	**Boolean**	Read Only
Internal Registers	300001-365536 300001-365535 3xxxxx.00-3xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read Only
Holding Registers	400001-465536 400001-465535 4xxxxx.00-4xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read/Write

Modbus Addressing Hexadecimal Format

Modbus Addressing Decimal Format

Memory Type	Range	Data Type	Access
Output Coils	000001-000008	**Boolean**	Read/Write
Input Coils	100001-100008	**Boolean**	Read Only
Internal Registers	300001-310000 300001-30FFFF 3xxxxx.00-3xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read Only
Holding Registers	400001-410000 400001-40FFFF 4xxxxx.00-4xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read/Write

Notes:

- Not all input coil and holding register addresses are writable in the UDC2500. See tables below and device user's manual for complete parameter mapping and access permission.
- Internal registers and Holding registers are mapped to the same memory range in the UDC2500.
- Addresses 307693, 307694, 407693, and 407694 are write only. Client applications will always read 0 for these tags.

Parameter Mapping

The following tables describe the most important parameters and their Modbus addresses. These are the tags that will be automatically generated by this driver.

Digital Inputs

Name	Address	Type	Access	Description
Input1	100001	Boolean	Read Only	Digital Input 1
Input2	100002	Boolean	Read Only	Digital Input 2

Digital Outputs

Name	Address	Type	Access	Description
Output1	000001	Boolean	Read Only	Digital Output 1
Output2	000002	Boolean	Read Only	Digital Output 2
Alarm2	000003	Boolean	Read Only	Alarm Relay 2
Alarm1	000004	Boolean	Read Only	Alarm Relay 1

Integer Registers

Name	Address	Type	Access	Description
PV	400001	Short	Read Only	Present Value
RSPSP2	400002	Short	Read Only	RSP SP2
WSP	400003	Short	Read Only	Working Setpoint (LSP1, LSP2, or RSP)
Output	400004	Short	Read/Write	Output
Input1	400005	Short	Read Only	Input 1
Input2	400006	Short	Read Only	Input 2
Gain1PropBand1	400007	Short	Read/Write	Gain 1 Prop Band 1
Direction	400008	Short	Read Only	Direction (0=Direct, 1=Reverse)
Reset1	400009	Short	Read/Write	Reset 1
Rate1	400010	Short	Read/Write	Rate 1
CycleTime	400011	Short	Read/Write	Cycle Time
PVLowRange	400012	Short	Read Only	PV Low Range
PVHighRange	400013	Short	Read Only	PV High Range
Alarm1SP1	400014	Short	Read/Write	Alarm1 SP1
Alarm1SP2	400015	Short	Read/Write	Alarm1 SP2
Alarm1Action	400016	Short	Read Only	Alarm 1 Action
Gain2ProBand2	400017	Short	Read/Write	Gain 2 Prop Band 2
2PosStepDeadband	400018	Short	Read/Write	3 Pos Step Deadband (-5 to 25)
Reset2	400019	Short	Read/Write	Reset 2
Rate2	400020	Short	Read/Write	Rate 2
CycleTime2	400021	Short	Read/Write	Cycle Time 2
LSP1	400022	Short	Read/Write	Local Set Point 1
LSP2	400023	Short	Read/Write	Local Set Point 2
Alarm2SP1	400024	Short	Read/Write	Alarm 2 SP 1
Alarm2SP2	400025	Short	Read/Write	Alarm 2 SP 2
Alarm2Ev	400026	Short	Read/Write	Alarm 2 Events

Name	Address	Type	Access	Description
SPLowLimit	400027	Short	Read/Write	SP Low Limit
SPHighLimit	400028	Short	Read/Write	SP High Limit
SP	400029	Short	Read/Write	SP Working Value
OutputLowLimit	400030	Short	Read/Write	Output Low Limit
OutputHighLimit	400031	Short	Read/Write	Output High Limit
OutputWorkingValue	400032	Short	Read/Write	Output Working Value
PVOverride	400033	Short	Read/Write	PV Override Value
SPOverride	400034	Short	Read/Write	SP Override Value
OutputOverride	400035	Short	Read/Write	Output Override Value
CSPRatio	400036	Short	Read/Write	CSP Ratio
CSPBias	400037	Short	Read Only	CSP Bias
Deviation	400038	Short	Read Only	Deviation
AutoManState	400251	Word	Read/Write	0=Manual, 1=Auto
LSPSelectState	400252	Word	Read/Write	0=LSP1, 1=LSP2
RemLocSPState	400253	Word	Read/Write	0=LSP, 1=RSP
TuneSetState	400254	Word	Read/Write	0=Tune Set 1, 1= Tune Set 2
LoopStatus_Register	400255	Word	Read Only	Loop Status Register
LoopStatus_Mode	400255.00	Boolean	Read Only	Loop Status Mode Bit (0=Manual, 1=Auto)
LoopStatus_SP	400255.01	Boolean	Read Only	Loop Status SP Bit (0=SP1, 1=SP2)
LoopStatus_RemLoc	400255.02	Boolean	Read Only	Loop Status Remote/Local Bit (0=LSP, 1=RSP)
LoopStatus_TuneSet	400255.03	Boolean	Read Only	Loop Status Tune Set Bit (0=Set1, 1=Set2)
LoopStatus_CSP	400255.06	Boolean	Read Only	Loop Status CSP In Use Bit
AlarmStatus_Register	407153	Word	Read Only	Alarm Status Register
AlarmStatus_Alarm1	407153.00	Boolean	Read Only	Alarm Status Alarm 1 Bit
AlarmStatus_Alarm2	407153.01	Boolean	Read Only	Alarm Status Alarm 2 Bit

Float Registers

Name	Address	Type	Access	Description
PV	400065	Float	Read Only	Present Value
RSPSP2	400067	Float	Read Only	RSP SP2
WSP	400069	Float	Read Only	Working Setpoint
Output	400071	Float	Read/Write	Output
Input1	400073	Float	Read Only	Input 1
Gain1PropBand1	400077	Float	Read/Write	Gain 1 Prop Band 1
Direction	400079	Float	Read Only	Direction (0=Direct, 1=Reverse)
Reset1	400081	Float	Read/Write	Reset 1
Rate1	400083	Float	Read/Write	Rate 1
CycleTime	400085	Float	Read/Write	Cycle Time

Name	Address	Type	Access	Description
PVLowRange	400087	Float	Read Only	PV Low Range
PVHighRange	400089	Float	Read Only	PV High Range
Alarm1SP1	400091	Float	Read/Write	Alarm1 SP1
Alarm1SP2	400093	Float	Read/Write	Alarm1 SP2
Gain2ProBand2	400097	Float	Read/Write	Gain 2 Prop Band 2
3PosStepDeadband	400099	Float	Read/Write	3 Pos Step Deadband (-5 to 25)
Reset2	400101	Float	Read/Write	Reset 2
Rate2	400103	Float	Read/Write	Rate 2
CycleTime2	400105	Float	Read/Write	Cycle Time 2
LSP1	400107	Float	Read/Write	Local Set Point 1
LSP2	400109	Float	Read/Write	Local Set Point 2
Alarm2SP1	400111	Float	Read/Write	Alarm 2 SP 1
Alarm2SP2	400113	Float	Read/Write	Alarm 2 SP 2
SPLowLimit	400117	Float	Read/Write	SP Low Limit
SPHighLimit	400119	Float	Read/Write	SP High Limit
WSP	400121	Float	Read/Write	Working Setpoint (LSP1, LSP2, or RSP)
OutputLowLimit	400123	Float	Read/Write	Output Low Limit
OutputHighLimit	400125	Float	Read/Write	Output High Limit
OutputWorkingValue	400127	Float	Read/Write	Output Working Value
PVOverride	400129	Float	Read/Write	PV Override Value
SPOverride	400131	Float	Read/Write	SP Override Value
OutputOverride	400133	Float	Read/Write	Output Override Value
CSPRatio	400135	Float	Read/Write	CSP Ratio
CSPBias	400137	Float	Read Only	CSP Bias
Deviation	400139	Float	Read/Write	Deviation
AuxOutput	400163	Float	Read Only	Auxilary Output
SPRampTime	400165	Float	Read/Write	Setpoint Ramp Time
SetpointRampSP	400167	Float	Read/Write	Setpoint Ramp SP
In1Ratio	400169	Float	Read/Write	Input1 Ratio
In1Bias	400171	Float	Read/Write	Input1 Bias
In2Ratio	400173	Float	Read/Write	Input2 Ratio
In2Bias	400175	Float	Read/Write	Input2 Bias
SPSwitchValue	400177	Float	Read/Write	SP Switch Value
AnalogInp1	406145	Float	Read Only	Analog Input 1
AnalogInp2	406147	Float	Read Only	Analog Input 2
Alarm1SpVal1	407169	Float	Read/Write	Alarm 1 Setpoint 1
Alarm1SpVal2	407171	Float	Read/Write	Alarm 1 Setpoint 2
Alarm2SpVal1	407173	Float	Read/Write	Alarm 2 Setpoint 1
Alarm2SpVal2	407175	Float	Read/Write	Alarm 2 Setpoint 2

Set Point Programming

Name	Address	Type	Access	Description
ProgramOutput	407681	Float	Read Only	Program Output
SegmentNum	407683	Float	Read Only	Segment Number
SegTimeRemain	407689	Float	Read Only	Segment Time Remain
Status_Register	407692	Word	Read Only	Status Register
Status_Ready	407692.00	Boolean	Read Only	Status Ready Bit
Status_Run	407692.01	Boolean	Read Only	Status Run Bit
Status_Hold	407692.02	Boolean	Read Only	Status Hold Bit
Status_End	407692.03	Boolean	Read Only	Status End Bit
Run	407693	Short	Write Only	Write 1 to Run
Hold	407694	Short	Write Only	Write 0 to Hold
TimeUnits	407995	Word	Read/Write	Time Units
RampUnits	407996	Word	Read/Write	Ramp Units
ProgEndSeg_ Register	407998	Word	Read/Write	Program End Segment Register
ProgEndSeg_02	407998.00	Boolean	Read/Write	Program End Segment 2 Bit
ProgEndSeg_04	407998.01	Boolean	Read/Write	Program End Segment 4 Bit
ProgEndSeg_06	407998.02	Boolean	Read/Write	Program End Segment 6 Bit
ProgEndSeg_08	407998.03	Boolean	Read/Write	Program End Segment 8 Bit
ProgEndSeg_10	407998.04	Boolean	Read/Write	Program End Segment 10 Bit
ProgEndSeg_12	407998.05	Boolean	Read/Write	Program End Segment 12 Bit
PrgTermState	407999	Word	Read/Write	Program Termination State (0=Last SP, 1=FailSafe)
PrgStateEnd	408000	Word	Read/Write	Program state at Program End (0=Disable, 1=Hold)
EURampUnits_ Register	408001	Word	Read/Write	Engineering Units for Ramp Segments Register
EURampUnits_ HrMin	408001.00	Boolean	Read/Write	Engineering Units for Ramp Segments Hours & Minutes Bit
EURampUnits_ DegMin	408001.01	Boolean	Read/Write	Engineering Units for Ramp Segments Degrees/Minute Bit
EURampUnits_ DegHr	408001.02	Boolean	Read/Write	Engineering Units for Ramp Segments Degrees/Hour Bit
PrgStartSeg_ Register	408002	Word	Read/Write	Program Start Segment Register
PrgStartSeg_01	408002.00	Boolean	Read/Write	Program Start Segment 1 Bit
PrgStartSeg_02	408002.01	Boolean	Read/Write	Program Start Segment 2 Bit
PrgStartSeg_03	408002.02	Boolean	Read/Write	Program Start Segment 3 Bit
PrgStartSeg_04	408002.03	Boolean	Read/Write	Program Start Segment 4 Bit
PrgStartSeg_05	408002.04	Boolean	Read/Write	Program Start Segment 5 Bit
PrgStartSeg_06	408002.05	Boolean	Read/Write	Program Start Segment 6 Bit

Name	Address	Type	Access	Description
PrgStartSeg_07	408002.06	Boolean	Read/Write	Program Start Segment 7 Bit
PrgStartSeg_08	408002.07	Boolean	Read/Write	Program Start Segment 8 Bit
PrgStartSeg_09	408002.08	Boolean	Read/Write	Program Start Segment 9 Bit
PrgStartSeg_10	408002.09	Boolean	Read/Write	Program Start Segment 10 Bit
PrgStartSeg_11	408002.10	Boolean	Read/Write	Program Start Segment 11 Bit
PrgStartSeg_12	408002.11	Boolean	Read/Write	Program Start Segment 12 Bit
PrgRcycl	408003	Word	Read/Write	Program Recycle
Seg01Ramp	410241	Word	Read Only	Segment 1 Ramp
Seg01TimeRate	410243	Float	Read/Write	Segment 1 Time Ramp
Seg02Soak	410249	Word	Read Only	Segment 2 Soak
Seg02SoakTime	410251	Float	Read/Write	Segment 2 Soak Time
Seg02SoakSpValue	410253	Float	Read/Write	Segment 2 Soak SP Value
Seg03Ramp	410257	Word	Read Only	Segment 3 Ramp
Seg03TimeRate	410259	Float	Read/Write	Segment 3 Time Ramp
Seg04Soak	410265	Word	Read Only	Segment 4 Soak
Seg04SoakTime	410267	Float	Read/Write	Segment 4 Soak Time
Seg04SoakSpValue	410269	Float	Read/Write	Segment 4 Soak SP Value
Seg05Ramp	410273	Word	Read Only	Segment 5 Ramp
Seg05TimeRate	410275	Float	Read/Write	Segment 5 Time Ramp
Seg06Soak	410281	Word	Read Only	Segment 6 Soak
Seg06SoakTime	410283	Float	Read/Write	Segment 6 Soak Time
Seg06SoakSpValue	410285	Float	Read/Write	Segment 6 Soak SP Value
Seg07Ramp	410289	Word	Read Only	Segment 7 Ramp
Seg07TimeRate	410291	Float	Read/Write	Segment 7 Time Ramp
Seg08Soak	410297	Word	Read Only	Segment 8 Soak
Seg08SoakTime	410299	Float	Read/Write	Segment 8 Soak Time
Seg08SoakSpValue	410301	Float	Read/Write	Segment 8 Soak SP Value
Seg09Ramp	410305	Word	Read Only	Segement 9 Ramp
Seg09TimeRate	410307	Float	Read/Write	Segment 9 Time Ramp
Seg10Soak	410313	Word	Read Only	Segment 10 Soak
Seg10SoakTime	410315	Float	Read/Write	Segment 10 Soak Time
Seg10SoakSpValue	410317	Float	Read/Write	Segment 10 Soak SP Value
Seg11Ramp	410321	Word	Read Only	Segment 11 Ramp
Seg11TimeRate	410323	Float	Read/Write	Segment 11 Time Ramp
Seg12Soak	410329	Word	Read Only	Segment 12 Soak
Seg12SoakTime	410331	Float	Read/Write	Segment 12 Soak Time
Seg12SoakSpValue	410333	Float	Read/Write	Segment 12 SP Value

UDC 3200 Addressing

The default data types for dynamically defined tags are shown in **bold** where appropriate.

Modbus Addressing Decimal Format

Memory Type	Range	Data Type	Access
Output Coils	000001-000008	**Boolean**	Read/Write
Input Coils	100001-100008	**Boolean**	Read Only
Internal Registers	300001-365536 300001-365535 3xxxxx.00-3xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read Only
Holding Registers	400001-465536 400001-465535 4xxxxx.00-4xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read/Write

Modbus Addressing Hexadecimal Format

Memory Type	Range	Data Type	Access
Output Coils	000001-000008	**Boolean**	Read/Write
Input Coils	100001-100008	**Boolean**	Read Only
Internal Registers	300001-310000 300001-30FFFF 3xxxxx.00-3xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read Only
Holding Registers	400001-410000 400001-40FFFF 4xxxxx.00-4xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read/Write

Notes

- Not all input coil and holding register addresses are writable in the UDC3200. See tables below and device user's manual for complete parameter mapping and access permission.
- Internal registers and Holding registers are mapped to the same memory range in the UDC3200.
- Addresses 307693, 307694, 407693, and 407694 are write only. Client applications will always read 0 for these tags.

Parameter Mapping

The following tables describe the most important parameters and their Modbus addresses. These are the tags that will be automatically generated by this driver.

Digital Inputs

Name	Address	Type	Access	Description
Input1	100001	Boolean	Read Only	Digital Input 1
Input2	100002	Boolean	Read Only	Digital Input 2

Digital Outputs

Name	Address	Type	Access	Description
Output1	000001	Boolean	Read Only	Digital Output 1
Output2	000002	Boolean	Read Only	Digital Output 2
Alarm2	000003	Boolean	Read Only	Alarm Relay 2
Alarm1	000004	Boolean	Read Only	Alarm Relay 1

Integer Registers

Name	Address	Type	Access	Description
PV	400001	Short	Read Only	Present Value
RSPSP2	400002	Short	Read Only	RSP SP2
WSP	400003	Short	Read Only	Working Setpoint (LSP1, LSP2, or RSP)
Output	400004	Short	Read/Write	Output
Input1	400005	Short	Read Only	Input 1
Input2	400006	Short	Read Only	Input 2
Gain1PropBand1	400007	Short	Read/Write	Gain 1 Prop Band 1
Direction	400008	Short	Read Only	Direction (0=Direct, 1=Reverse)
Reset1	400009	Short	Read/Write	Reset 1
Rate1	400010	Short	Read/Write	Rate 1
CycleTime	400011	Short	Read/Write	Cycle Time
PVLowRange	400012	Short	Read Only	PV Low Range
PVHighRange	400013	Short	Read Only	PV High Range
Alarm1SP1	400014	Short	Read/Write	Alarm1 SP1
Alarm1SP2	400015	Short	Read/Write	Alarm1 SP2
Alarm1Action_ Register	400016	Word	Read Only	Alarm1 ActionRegister
Alarm1Action_AL11EV	400016.00	Boolean	Read Only	Alarm1 Actin AL11EV Bit
Alarm1Action_AL12EV	400016.01	Boolean	Read Only	Alarm1 Action AL12EV Bit
Gain2ProBand2	400017	Short	Read/Write	Gain 2 Prop Band 2
2PosStepDeadband	400018	Short	Read/Write	3 Pos Step Deadband (-5 to 25)
Reset2	400019	Short	Read/Write	Reset 2
Rate2	400020	Short	Read/Write	Rate 2
CycleTime2	400021	Short	Read/Write	Cycle Time 2
LSP1	400022	Short	Read/Write	Local Set Point 1
LSP2	400023	Short	Read/Write	Local Set Point 2
Alarm2SP1	400024	Short	Read/Write	Alarm 2 SP 1
Alarm2SP2	400025	Short	Read/Write	Alarm 2 SP 2
Alarm2Ev_Register	400026	Word	Read/Write	Alarm 2 Events Register
Akarn2Ev_AL11EV	400026.00	Boolean	Read/Write	Alarm 2 Events AL11EV Bit
Alarm2Ev_AL12EV	400026.01	Boolean	Read/Write	Alarm 2 Events AL12EV Bit
SPLowLimit	400027	Short	Read/Write	SP Low Limit

Name	Address	Type	Access	Description
SPHighLimit	400028	Short	Read/Write	SP High Limit
SP	400029	Short	Read/Write	SP Working Value
OutputLowLimit	400030	Short	Read/Write	Output Low Limit
OutputHighLimit	400031	Short	Read/Write	Output High Limit
OutputWorkingValue	400032	Short	Read/Write	Output Working Value
PVOverride	400033	Short	Read/Write	PV Override Value
SPOverride	400034	Short	Read/Write	SP Override Value
OutputOverride	400035	Short	Read/Write	Output Override Value
CSPRatio	400036	Short	Read/Write	CSP Ratio
CSPBias	400037	Short	Read Only	CSP Bias
Deviation	400038	Short	Read Only	Deviation
LSP3	400039	Word	Read/Write	LSP #3
PerCO	400040	Word	Read/Write	Percent CO
DecimalPoint	400041	Short	Read/Write	Decimal Point Location
Alg1Bias	400042	Word	Read Only	Algorithm 1 Bias (prescale dependent on DP)
Fuzzy	400056	Short	Read/Write	Fuzzy Enable
ShedEnable	400057	Short	Read/Write	Shed Enable
AutoManState	400059	Short	Read/Write	0=Manual, 1=Auto
LSPSelectState	400060	Short	Read/Write	0=LSP1, 1=LSP2, 2=LSP3
RemLocSPState	400061	Short	Read/Write	0=LSP, 1=RSP
TuneSetState	400062	Short	Read/Write	0=Tune Set 1, 1=Tune Set 2
LoopStatus_Register	400063	Word	Read Only	Loop Status Register
LoopStatus_Mode	400063.00	Boolean	Read Only	Loop Status Mode Bit (0=Manual, 1=Auto)
LoopStatus_SP	400063.01	Boolean	Read Only	Loop Status SP Bit (0=SP1, 1=SP2)
LoopStatus_RemLoc	400063.02	Boolean	Read Only	Loop Status Remote/Local Bit (0=LSP, 1=RSP)
LoopStatus_TuneSet	400063.03	Boolean	Read Only	Loop Status Tune Set Bit (0=Set1, 1=Set2)
LoopStatus_LSP3	400063.04	Boolean	Read Only	Loop Status LSP3 In Use Bit
LoopStatus_CSP	400063.06	Boolean	Read Only	Loop Status CSP In Use Bit
AlarmStatus_Register	407153	Word	Read Only	Alarm Status Register
AlarmStatus_Alarm1	407153.00	Boolean	Read Only	Alarm Status Alarm 1 Bit
AlarmStatus_Alarm2	407153.01	Boolean	Read Only	Alarm Status Alarm 2 Bit

Float Registers

Name	Address	Type	Access	Description
PV	400065	Float	Read Only	Present Value
RSPSP2	400067	Float	Read Only	RSP SP2
WSP	400069	Float	Read Only	Working Setpoint
Output	400071	Float	Read/Write	Output

Name	Address	Type	Access	Description
Input1	400073	Float	Read Only	Input 1
Gain1PropBand1	400077	Float	Read/Write	Gain 1 Prop Band 1
Direction	400079	Float	Read Only	Direction (0=Direct, 1=Reverse)
Reset1	400081	Float	Read/Write	Reset 1
Rate1	400083	Float	Read/Write	Rate 1
CycleTime	400085	Float	Read/Write	Cycle Time
PVLowRange	400087	Float	Read Only	PV Low Range
PVHighRange	400089	Float	Read Only	PV High Range
Alarm1SP1	400091	Float	Read/Write	Alarm1 SP1
Alarm1SP2	400093	Float	Read/Write	Alarm1 SP2
Gain2ProBand2	400097	Float	Read/Write	Gain 2 Prop Band 2
3PosStepDeadband	400099	Float	Read/Write	3 Pos Step Deadband (-5 to 25)
Reset2	400101	Float	Read/Write	Reset 2
Rate2	400103	Float	Read/Write	Rate 2
CycleTime2	400105	Float	Read/Write	Cycle Time 2
LSP1	400107	Float	Read/Write	Local Set Point 1
LSP2	400109	Float	Read/Write	Local Set Point 2
Alarm2SP1	400111	Float	Read/Write	Alarm 2 SP 1
Alarm2SP2	400113	Float	Read/Write	Alarm 2 SP 2
SPLowLimit	400117	Float	Read/Write	SP Low Limit
SPHighLimit	400119	Float	Read/Write	SP High Limit
WSP	400121	Float	Read/Write	"Working Setpoint (LSP1, LSP2, or RSP)
OutputLowLimit	400123	Float	Read/Write	Output Low Limit
OutputHighLimit	400125	Float	Read/Write	Output High Limit
OutputWorkingValue	400127	Float	Read/Write	Output Working Value
PVOverride	400129	Float	Read/Write	PV Override Value
SPOverride	400131	Float	Read/Write	SP Override Value
OutputOverride	400133	Float	Read/Write	Output Override Value
CSPRatio	400135	Float	Read/Write	CSP Ratio
CSPBias	400137	Float	Read Only	CSP Bias
Deviation	400139	Float	Read/Write	Deviation
LSP3	400141	Float	Read/Write	LSP #3
Alg1Bias	400159	Float	Read Only	Algorithm 1 Bias (prescale dependent on DP)
AuxOutput	400163	Float	Read Only	Auxilary Output
SPRampTime	400165	Float	Read/Write	Setpoint Ramp Time
SetpointRampSP	400167	Float	Read/Write	Setpoint Ramp SP
In1Ratio	400169	Float	Read/Write	Input1 Ratio
In1Bias	400171	Float	Read/Write	Input1 Bias
In2Ratio	400173	Float	Read/Write	Input2 Ratio

Name	Address	Type	Access	Description
In2Bias	400175	Float	Read/Write	Input2 Bias
SPSwitchValue	400177	Float	Read/Write	SP Switch Value
AnalogInp1	406145	Float	Read Only	Analog Input 1
AnalogInp2	406147	Float	Read Only	Analog Input 2
Alarm1SpVal1	407169	Float	Read/Write	Alarm 1 Setpoint 1
Alarm1SpVal2	407171	Float	Read/Write	Alarm 1 Setpoint 2
Alarm2SpVal1	407173	Float	Read/Write	Alarm 2 Setpoint 1
Alarm2SpVal2	407175	Float	Read/Write	Alarm 2 Setpoint 2

Set Point Programming

Name	Address	Type	Access	Description
ProgramOutput	407681	Float	Read Only	Program Output
SegmentNum	407683	Float	Read Only	Segment Number
SegTimeRemain	407689	Float	Read Only	Segment Time Remain
Status_Register	407692	Word	Read Only	Status Register
Status_Ready	407692.00	Boolean	Read Only	Status Ready Bit
Status_Run	407692.01	Boolean	Read Only	Status Run Bit
Status_Hold	407692.02	Boolean	Read Only	Status Hold Bit
Status_End	407692.03	Boolean	Read Only	Status End Bit
Run	407693	Short	Write Only	Write 1 to Run
Hold	407694	Short	Write Only	Write 0 to Hold
TimeUnits	407995	Word	Read/Write	Time Units
RampUnits	407996	Word	Read/Write	Ramp Units
ProgEndSeg_ Register	407998	Word	Read/Write	Program End Segment Register
ProgEndSeg_02	407998.00	Boolean	Read/Write	Program End Segment 2 Bit
ProgEndSeg_04	407998.01	Boolean	Read/Write	Program End Segment 4 Bit
ProgEndSeg_06	407998.02	Boolean	Read/Write	Program End Segment 6 Bit
ProgEndSeg_08	407998.03	Boolean	Read/Write	Program End Segment 8 Bit
ProgEndSeg_10	407998.04	Boolean	Read/Write	Program End Segment 10 Bit
ProgEndSeg_12	407998.05	Boolean	Read/Write	Program End Segment 12 Bit
PrgTermState	407999	Word	Read/Write	Program Termination State (0=Last SP, 1=FailSafe)
PrgStateEnd	408000	Word	Read/Write	Program state at Program End (0=Disable, 1=Hold)
EURampUnits_ Register	408001	Word	Read/Write	Engineering Units for Ramp Segments Register
EURampUnits_ HrMin	408001.00	Boolean	Read/Write	Engineering Units for Ramp Segments Hours & Minutes Bit
EURampUnits_ DegMin	408001.01	Boolean	Read/Write	Engineering Units for Ramp Segments Degrees/Minute Bit

Name	Address	Type	Access	Description
EURampUnits_ DegHr	408001.02	Boolean	Read/Write	Engineering Units for Ramp Segments Degrees/Hour Bit
PrgStartSeg_ Register	408002	Word	Read/Write	Program Start Segment Register
PrgStartSeg_01	408002.00	Boolean	Read/Write	Program Start Segment 1 Bit
PrgStartSeg_02	408002.01	Boolean	Read/Write	Program Start Segment 2 Bit
PrgStartSeg_03	408002.02	Boolean	Read/Write	Program Start Segment 3 Bit
PrgStartSeg_04	408002.03	Boolean	Read/Write	Program Start Segment 4 Bit
PrgStartSeg_05	408002.04	Boolean	Read/Write	Program Start Segment 5 Bit
PrgStartSeg_06	408002.05	Boolean	Read/Write	Program Start Segment 6 Bit
PrgStartSeg_07	408002.06	Boolean	Read/Write	Program Start Segment 7 Bit
PrgStartSeg_08	408002.07	Boolean	Read/Write	Program Start Segment 8 Bit
PrgStartSeg_09	408002.08	Boolean	Read/Write	Program Start Segment 9 Bit
PrgStartSeg_10	408002.09	Boolean	Read/Write	Program Start Segment 10 Bit
PrgStartSeg_11	408002.10	Boolean	Read/Write	Program Start Segment 11 Bit
PrgStartSeg_12	408002.11	Boolean	Read/Write	Program Start Segment 12 Bit
PrgRcycl	408003	Word	Read/Write	Program Recycle
Seg01Ramp	410241	Word	Read Only	Segment 1 Ramp
Seg01TimeRate	410243	Float	Read/Write	Segment 1 Time Ramp
Seg02Soak	410249	Word	Read Only	Segment 2 Soak
Seg02SoakTime	410251	Float	Read/Write	Segment 2 Soak Time
Seg02SoakSpValue	410253	Float	Read/Write	Segment 2 Soak SP Value
Seg03Ramp	410257	Word	Read Only	Segment 3 Ramp
Seg03TimeRate	410259	Float	Read/Write	Segment 3 Time Ramp
Seg 04Soak	410265	Word	Read Only	Segment 4 Soak
Seg04SoakTime	410267	Float	Read/Write	Segment 4 Soak Time
Seg04SoakSpValue	410269	Float	Read/Write	Segment 4 Soak SP Value
Seg05Ramp	410273	Word	Read Only	Segment 5 Ramp
Seg05TimeRate	410275	Float	Read/Write	Segment 5 Time Ramp
Seg6Soak	410281	Word	Read Only	Segment 6 Soak
Seg06SoakTime	410283	Float	Read/Write	Segment 6 Soak Time
Seg06SoakSpValue	410285	Float	Read/Write	Segment 6 Soak SP Value
Seg07Ramp	410289	Word	Read Only	Segment 7 Ramp
Seg07TimeRate	410291	Float	Read/Write	Segment 7 Time Ramp
Seg8Soak	410297	Word	Read Only	Segment 8 Soak
Seg08SoakTime	410299	Float	Read/Write	Segment 8 Soak Time
Seg08SoakSpValue	410301	Float	Read/Write	Segment 8 Soak SP Value
Seg09Ramp	410305	Word	Read Only	Segment 9 Ramp
Seg09TimeRate	410307	Float	Read/Write	Segment 9 Time Ramp
Seg10Soak	410313	Word	Read Only	Segment 10 Soak

Name	Address	Type	Access	Description
Seg10SoakTime	410315	Float	Read/Write	Segment 10 Soak Time
Seg10SoakSpValue	410317	Float	Read/Write	Segment 10 Soak SP Value
Seg11Ramp	410321	Word	Read Only	Segment 11 Ramp
Seg11TimeRate	410323	Float	Read/Write	Segment 11 Time Ramp
Seg12Soak	410329	Word	Read Only	Segment 12 Soak
Seg12SoakTime	410331	Float	Read/Write	Segment 12 Soak Time
Seg12SoakSpValue	410333	Float	Read/Write	Segment 12 SP Value

UDC 3500 Addressing

The default data types for dynamically defined tags are shown in **bold** where appropriate.

Modbus Addressing Decimal Format

Memory Type	Range	Data Type	Access
Output Coils	000001-000008	**Boolean**	Read/Write
Input Coils	100001-100008	**Boolean**	Read Only
Internal Registers	300001-365536 300001-365535 3xxxxx.00-3xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read Only
Holding Registers	400001-465536 400001-465535 4xxxxx.00-4xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read/Write

Modbus Addressing Hexadecimal Format

Memory Type	Range	Data Type	Access
Output Coils	000001-000008	**Boolean**	Read/Write
Input Coils	100001-100008	**Boolean**	Read Only
Internal Registers	300001-310000 300001-30FFFF 3xxxxx.00-3xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read Only
Holding Registers	400001-410000 400001-40FFFF 4xxxxx.00-4xxxxx.15	**Word**, Short DWord, Long, Float Boolean	Read/Write

Notes:

- Not all input coil and holding register addresses are writable in the UDC3500. See tables below and device user's manual for complete parameter mapping and access permission.
- Internal registers and Holding registers are mapped to the same memory range in the UDC3500.
- Addresses 307693, 307694, 407693, and 407694 are write only. Client applications will always read 0 for these tags.

Parameter Mapping

The following tables describe the most important parameters and their Modbus addresses. These are the tags that will be automatically generated by this driver.

Digital Inputs

Name	Address	Type	Access	Description
Input1	100001	Boolean	Read Only	Digital Input 1
Input2	100002	Boolean	Read Only	Digital Input 2

Digital Outputs

Name	Address	Type	Access	Description
Output1	000001	Boolean	Read Only	Digital Output 1
Output2	000002	Boolean	Read Only	Digital Output 2
Alarm2	000003	Boolean	Read Only	Alarm Relay 2
Alarm1	000004	Boolean	Read Only	Alarm Relay 1

Integer Registers

Name	Address	Type	Access	Description
PV	400001	Short	Read Only	Present Value
RSPSP2	400002	Short	Read Only	RSP SP2
WSP	400003	Short	Read Only	Working Setpoint (LSP1, LSP2, or RSP)
Output	400004	Short	Read/Write	Output
Input1	400005	Short	Read Only	Input 1
Input2	400006	Short	Read Only	Input 2
Gain1PropBand1	400007	Short	Read/Write	Gain 1 Prop Band 1
Direction	400008	Short	Read Only	Direction (0=Direct, 1=Reverse)
Reset1	400009	Short	Read/Write	Reset 1
Rate1	400010	Short	Read/Write	Rate 1
CycleTime	400011	Short	Read/Write	Cycle Time
PVLowRange	400012	Short	Read Only	PV Low Range
PVHighRange	400013	Short	Read Only	PV High Range
Alarm1SP1	400014	Short	Read/Write	Alarm1 SP1
Alarm1SP2	400015	Short	Read/Write	Alarm1 SP2
Alarm1Action_ Register	400016	Word	Read Only	Alarm1 ActionRegister
Alarm1Action_AL11EV	400016.00	Boolean	Read Only	Alarm1 Actin AL11EV Bit
Alarm1Action_AL12EV	400016.01	Boolean	Read Only	Alarm1 Action AL12EV Bit
Gain2ProBand2	400017	Short	Read/Write	Gain 2 Prop Band 2
2PosStepDeadband	400018	Short	Read/Write	3 Pos Step Deadband (-5 to 25)
Reset2	400019	Short	Read/Write	Reset 2
Rate2	400020	Short	Read/Write	Rate 2
CycleTime2	400021	Short	Read/Write	Cycle Time 2

Name	Address	Type	Access	Description
LSP1	400022	Short	Read/Write	Local Set Point 1
LSP2	400023	Short	Read/Write	Local Set Point 2
Alarm2SP1	400024	Short	Read/Write	Alarm 2 SP 1
Alarm2SP2	400025	Short	Read/Write	Alarm 2 SP 2
Alarm2Ev_Register	400026	Word	Read/Write	Alarm 2 Events Register
Akarn2Ev_AL11EV	400026.00	Boolean	Read/Write	Alarm 2 Events AL11EV Bit
Alarm2Ev_AL12EV	400026.01	Boolean	Read/Write	Alarm 2 Events AL12EV Bit
SPLowLimit	400027	Short	Read/Write	SP Low Limit
SPHighLimit	400028	Short	Read/Write	SP High Limit
SP	400029	Short	Read/Write	SP Working Value
OutputLowLimit	400030	Short	Read/Write	Output Low Limit
OutputHighLimit	400031	Short	Read/Write	Output High Limit
OutputWorkingValue	400032	Short	Read/Write	Output Working Value
PVOverride	400033	Short	Read/Write	PV Override Value
SPOverride	400034	Short	Read/Write	SP Override Value
OutputOverride	400035	Short	Read/Write	Output Override Value
CSPRatio	400036	Short	Read/Write	CSP Ratio
CSPBias	400037	Short	Read Only	CSP Bias
Deviation	400038	Short	Read Only	Deviation
LSP3	400039	Word	Read/Write	LSP #3
PerCO	400040	Word	Read/Write	Percent CO
DecimalPoint	400041	Short	Read/Write	Decimal Point Location
Alg1Bias	400042	Word	Read Only	Algorithm 1 Bias (prescale dependent on DP)
Fuzzy	400056	Short	Read/Write	Fuzzy Enable
ShedEnable	400057	Short	Read/Write	Shed Enable
AutoManState	400059	Short	Read/Write	0=Manual, 1=Auto
LSPSelectState	400060	Short	Read/Write	0=LSP1, 1=LSP2, 2=LSP3
RemLocSPState	400061	Short	Read/Write	0=LSP, 1=RSP
TuneSetState	400062	Short	Read/Write	0=Tune Set 1, 1=Tune Set 2
LoopStatus_Register	400063	Word	Read Only	Loop Status Register
LoopStatus_Mode	400063.00	Boolean	Read Only	Loop Status Mode Bit (0=Manual, 1=Auto)
LoopStatus_SP	400063.01	Boolean	Read Only	Loop Status SP Bit (0=SP1, 1=SP2)
LoopStatus_RemLoc	400063.02	Boolean	Read Only	Loop Status Remote/Local Bit (0=LSP, 1=RSP)
LoopStatus_TuneSet	400063.03	Boolean	Read Only	Loop Status Tune Set Bit (0=Set1, 1=Set2)
LoopStatus_LSP3	400063.04	Boolean	Read Only	Loop Status LSP3 In Use Bit
LoopStatus_CSP	400063.06	Boolean	Read Only	Loop Status CSP In Use Bit
AlarmStatus_Register	407153	Word	Read Only	Alarm Status Register
AlarmStatus_Alarm1	407153.00	Boolean	Read Only	Alarm Status Alarm 1 Bit

Name	Address	Type	Access	Description
AlarmStatus_Alarm2	407153.01	Boolean	Read Only	Alarm Status Alarm 2 Bit

Float Registers

Name	Address	Type	Access	Description
PV	400065	Float	Read Only	Present Value
RSPSP2	400067	Float	Read Only	RSP SP2
WSP	400069	Float	Read Only	Working Setpoint
Output	400071	Float	Read/Write	Output
Input1	400073	Float	Read Only	Input 1
Gain1PropBand1	400077	Float	Read/Write	Gain 1 Prop Band 1
Direction	400079	Float	Read Only	Direction (0=Direct, 1=Reverse)
Reset1	400081	Float	Read/Write	Reset 1
Rate1	400083	Float	Read/Write	Rate 1
CycleTime	400085	Float	Read/Write	Cycle Time
PVLowRange	400087	Float	Read Only	PV Low Range
PVHighRange	400089	Float	Read Only	PV High Range
Alarm1SP1	400091	Float	Read/Write	Alarm1 SP1
Alarm1SP2	400093	Float	Read/Write	Alarm1 SP2
Gain2ProBand2	400097	Float	Read/Write	Gain 2 Prop Band 2
3PosStepDeadband	400099	Float	Read/Write	3 Pos Step Deadband (-5 to 25)
Reset2	400101	Float	Read/Write	Reset 2
Rate2	400103	Float	Read/Write	Rate 2
CycleTime2	400105	Float	Read/Write	Cycle Time 2
LSP1	400107	Float	Read/Write	Local Set Point 1
LSP2	400109	Float	Read/Write	Local Set Point 2
Alarm2SP1	400111	Float	Read/Write	Alarm 2 SP 1
Alarm2SP2	400113	Float	Read/Write	Alarm 2 SP 2
SPLowLimit	400117	Float	Read/Write	SP Low Limit
SPHighLimit	400119	Float	Read/Write	SP High Limit
WSP	400121	Float	Read/Write	"Working Setpoint (LSP1, LSP2, or RSP)
OutputLowLimit	400123	Float	Read/Write	Output Low Limit
OutputHighLimit	400125	Float	Read/Write	Output High Limit
OutputWorkingValue	400127	Float	Read/Write	Output Working Value
PVOverride	400129	Float	Read/Write	PV Override Value
SPOverride	400131	Float	Read/Write	SP Override Value
OutputOverride	400133	Float	Read/Write	Output Override Value
CSPRatio	400135	Float	Read/Write	CSP Ratio
CSPBias	400137	Float	Read Only	CSP Bias
Deviation	400139	Float	Read/Write	Deviation

Name	Address	Type	Access	Description
LSP3	400141	Float	Read/Write	LSP #3
Alg1Bias	400159	Float	Read Only	Algorithm 1 Bias (prescale dependent on DP)
AuxOutput	400163	Float	Read Only	Auxilary Output
SPRampTime	400165	Float	Read/Write	Setpoint Ramp Time
SetpointRampSP	400167	Float	Read/Write	Setpoint Ramp SP
In1Ratio	400169	Float	Read/Write	Input1 Ratio
In1Bias	400171	Float	Read/Write	Input1 Bias
In2Ratio	400173	Float	Read/Write	Input2 Ratio
In2Bias	400175	Float	Read/Write	Input2 Bias
SPSwitchValue	400177	Float	Read/Write	SP Switch Value
AnalogInp1	406145	Float	Read Only	Analog Input 1
AnalogInp2	406147	Float	Read Only	Analog Input 2
Alarm1SpVal1	407169	Float	Read/Write	Alarm 1 Setpoint 1
Alarm1SpVal2	407171	Float	Read/Write	Alarm 1 Setpoint 2
Alarm2SpVal1	407173	Float	Read/Write	Alarm 2 Setpoint 1
Alarm2SpVal2	407175	Float	Read/Write	Alarm 2 Setpoint 2

Set Point Programming

Name	Address	Type	Access	Description
ProgramOutput	407681	Float	Read Only	Program Output
SegmentNum	407683	Float	Read Only	Segment Number
SegTimeRemain	407689	Float	Read Only	Segment Time Remain
Status_Register	407692	Word	Read Only	Status Register
Status_Ready	407692.00	Boolean	Read Only	Status Ready Bit
Status_Run	407692.01	Boolean	Read Only	Status Run Bit
Status_Hold	407692.02	Boolean	Read Only	Status Hold Bit
Status_End	407692.03	Boolean	Read Only	Status End Bit
Run	407693	Short	Write Only	Write 1 to Run
Hold	407694	Short	Write Only	Write 0 to Hold
TimeUnits	407995	Word	Read/Write	Time Units
RampUnits	407996	Word	Read/Write	Ramp Units
ProgEndSeg_ Register	407998	Word	Read/Write	Program End Segment Register
ProgEndSeg_02	407998.00	Boolean	Read/Write	Program End Segment 2 Bit
ProgEndSeg_04	407998.01	Boolean	Read/Write	Program End Segment 4 Bit
ProgEndSeg_06	407998.02	Boolean	Read/Write	Program End Segment 6 Bit
ProgEndSeg_08	407998.03	Boolean	Read/Write	Program End Segment 8 Bit
ProgEndSeg_10	407998.04	Boolean	Read/Write	Program End Segment 10 Bit
ProgEndSeg_12	407998.05	Boolean	Read/Write	Program End Segment 12 Bit
PrgTermState	407999	Word	Read/Write	Program Termination State (0=Last SP,

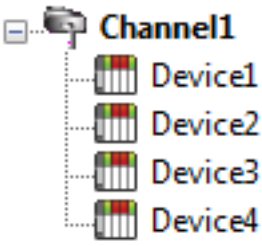
Name	Address	Type	Access	Description
				1=FailSafe)
PrgStateEnd	408000	Word	Read/Write	Program state at Program End (0=Disable, 1=Hold)
EURampUnits_ Register	408001	Word	Read/Write	Engineering Units for Ramp Segments Register
EURampUnits_ HrMin	408001.00	Boolean	Read/Write	Engineering Units for Ramp Segments Hours & Minutes Bit
EURampUnits_ DegMin	408001.01	Boolean	Read/Write	Engineering Units for Ramp Segments Degrees/Minute Bit
EURampUnits_ DegHr	408001.02	Boolean	Read/Write	Engineering Units for Ramp Segments Degrees/Hour Bit
PrgStartSeg_ Register	408002	Word	Read/Write	Program Start Segment Register
PrgStartSeg_01	408002.00	Boolean	Read/Write	Program Start Segment 1 Bit
PrgStartSeg_02	408002.01	Boolean	Read/Write	Program Start Segment 2 Bit
PrgStartSeg_03	408002.02	Boolean	Read/Write	Program Start Segment 3 Bit
PrgStartSeg_04	408002.03	Boolean	Read/Write	Program Start Segment 4 Bit
PrgStartSeg_05	408002.04	Boolean	Read/Write	Program Start Segment 5 Bit
PrgStartSeg_06	408002.05	Boolean	Read/Write	Program Start Segment 6 Bit
PrgStartSeg_07	408002.06	Boolean	Read/Write	Program Start Segment 7 Bit
PrgStartSeg_08	408002.07	Boolean	Read/Write	Program Start Segment 8 Bit
PrgStartSeg_09	408002.08	Boolean	Read/Write	Program Start Segment 9 Bit
PrgStartSeg_10	408002.09	Boolean	Read/Write	Program Start Segment 10 Bit
PrgStartSeg_11	408002.10	Boolean	Read/Write	Program Start Segment 11 Bit
PrgStartSeg_12	408002.11	Boolean	Read/Write	Program Start Segment 12 Bit
PrgRcycl	408003	Word	Read/Write	Program Recycle
Seg01Ramp	410241	Word	Read Only	Segment 1 Ramp
Seg01TimeRate	410243	Float	Read/Write	Segment 1 Time Ramp
Seg2Soak	410249	Word	Read Only	Segment 2 Soak
Seg02SoakTime	410251	Float	Read/Write	Segment 2 Soak Time
Seg02SoakSpValue	410253	Float	Read/Write	Segment 2 Soak SP Value
Seg3Ramp	410257	Word	Read Only	Segment 3 Ramp
Seg03TimeRate	410259	Float	Read/Write	Segment 3 Time Ramp
Seg04Soak	410265	Word	Read Only	Segment 4 Soak
Seg04SoakTime	410267	Float	Read/Write	Segment 4 Soak Time
Seg04SoakSpValue	410269	Float	Read/Write	Segment 4 Soak SP Value
Seg05Ramp	410273	Word	Read Only	Segment 5 Ramp
Seg05TimeRate	410275	Float	Read/Write	Segment 5 Time Ramp
Seg06Soak	410281	Word	Read Only	Segment 6 Soak
Seg06SoakTime	410283	Float	Read/Write	Segment 6 Soak Time

Name	Address	Type	Access	Description
Seg06SoakSpValue	410285	Float	Read/Write	Segment 6 Soak SP Value
Seg07Ramp	410289	Word	Read Only	Segment 7 Ramp
Seg07TimeRate	410291	Float	Read/Write	Segment 7 Time Ramp
Seg08Soak	410297	Word	Read Only	Segment 8 Soak
Seg08SoakTime	410299	Float	Read/Write	Segment 8 Soak Time
Seg08SoakSpValue	410301	Float	Read/Write	Segment 8 Soak SP Value
Seg9Ramp	410305	Word	Read Only	Segment 9 Ramp
Seg09TimeRate	410307	Float	Read/Write	Segment 9 Time Ramp
Seg10Soak	410313	Word	Read Only	Segment 10 Soak
Seg10SoakTime	410315	Float	Read/Write	Segment 10 Soak Time
Seg10SoakSpValue	410317	Float	Read/Write	Segment 10 Soak SP Value
Seg11Ramp	410321	Word	Read Only	Segment 11 Ramp
Seg11TimeRate	410323	Float	Read/Write	Segment 11 Time Ramp
Seg12Soak	410329	Word	Read Only	Segment 12 Soak
Seg12SoakTime	410331	Float	Read/Write	Segment 12 Soak Time
Seg12SoakSpValue	410333	Float	Read/Write	Segment 12 SP Value

Optimizing Communications

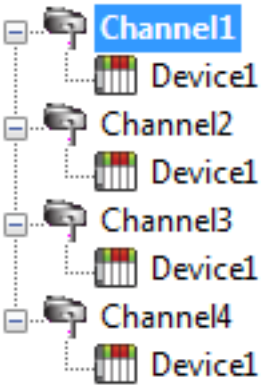
The Honeywell UDC Ethernet Driver has been designed to provide the best performance with the least amount of impact on the system's overall performance. While the Honeywell UDC Ethernet Driver is fast, there are a couple of guidelines that can be used to control and optimize the application and gain maximum performance.

Our server refers to communications protocols like Honeywell UDC Ethernet as a channel. Each channel defined in the application represents a separate path of execution in the server. Once a channel has been defined, a series of devices must then be defined under that channel. Each of these devices represents a single Honeywell UDC Ethernet from which data will be collected. While this approach to defining the application will provide a high level of performance, it won't take full advantage of the Honeywell UDC Ethernet Driver or the network. An example of how the application may appear when configured using a single channel is shown below.



Each device appears under a single Honeywell UDC Ethernet channel. In this configuration, the driver must move from one device to the next as quickly as possible to gather information at an effective rate. As more devices are added or more information is requested from a single device, the overall update rate begins to suffer.

If the Honeywell UDC Ethernet Driver could only define one single channel, then the example shown above would be the only option available; however, the Honeywell UDC Ethernet Driver can define up to 100 channels. Using multiple channels distributes the data collection workload by simultaneously issuing multiple requests to the network. An example of how the same application may appear when configured using multiple channels to improve performance is shown below.



Each device can be defined under its own channel. In this configuration, a single path of execution is dedicated to the task of gathering data from each device. If the application has fewer devices, it can be optimized exactly how it is shown here.

The performance will improve even if the application has more devices. While fewer devices may be ideal, the application will still benefit from additional channels. Although by spreading the device load across all channels will cause the server to move from device to device again, it can now do so with far less devices to process on a single channel.

Block Size, which is available on each defined device, can also affect the Honeywell UDC Ethernet Driver performance. Block Size refers to the number of bytes that may be requested from a device at one time. To refine the performance of this driver, configure Block Size to 1 to 22 registers.

Error Descriptions

The following error/warning messages may be generated. Click on the link for a description of the message. The errors are listed in alphabetical order.

See Also:
Modbus Exception Codes

Address '<address>' is out of range for the specified device or register

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically references a location that is beyond the range of supported locations for the device.

Solution:
Verify that the address is correct; if it is not, re-enter it in the client application.

Array support is not available for the specified address: '<address>'

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically contains an array reference for an address type that doesn't support arrays.

Solution:
Re-enter the address in the client application to remove the array reference or correct the address type.

Bad address in block [x to y] on device '<device name>'

Error Type:
Fatal addresses falling in this block.

Possible Cause:
This error is reported when the driver attempts to read a location in a PLC that does not exist. For example in a PLC that only has holding registers 40001 to 41400, requesting address 41405 would generate this error. Once this error is generated, the driver will not request the specified block of data from the PLC again. Any other addresses being requested that are in this same block will also go invalid.

Solution:
The client application should be modified to ask for addresses within the range of the device.

Bad received length [x to y] on device '<device name>'

Error Type:
Fatal addresses falling in this block.

Possible Cause:
The driver attempted to read a block of memory in the PLC. The PLC responded with no error, but did not provide the driver with the requested block size of data.

Solution:
Ensure that the range of memory exists for the PLC.

Data Type '<type>' is not valid for device address '<address>'

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically has been assigned an invalid data type.

Solution:
Modify the requested data type in the client application.

Device '<device name>' block request [x to y] responded with exception <code>

Error Type:
Fatal addresses failing in this block

Possible Cause:
This error is reported when the driver attempts to read a location in a PLC that does not exist. For example, in a PLC that only has holding registers 40001 to 41400, requesting address 41405 would generate this error. Once this error is generated, the driver will not request the specified block of data from the PLC again. Any other addresses being requested that are in this same block will also go invalid.

Solution:
The client application should be modified to ask for addresses within the range of the device.

Device '<device name>' is not responding

Error Type:
Serious

Possible Cause:

1. The connection between the device and the host PC is broken.
2. The communication parameters for the connection are incorrect.
3. The named device may have been assigned an incorrect Network ID.
4. The response from the device took longer to receive than the amount of time specified in the "Request Timeout" device setting.

Solution:

1. Verify the cabling between the PC and the device.
2. Verify that the specified communication parameters match those of the device.
3. Verify that the Network ID given to the named device matches that of the actual device.
4. Increase the Request Timeout setting so that the entire response can be handled.

Device address '<address>' contains a syntax error

Error Type:
Warning

Possible Cause:
An invalid tag address has been specified in a static request.

Solution:
Re-enter the address in the server.

Device address '<address>' is not supported by model '<model name>'

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically references a location that is valid for the communications protocol but not supported by the target device.

Solution:
Verify that the address is correct; if it is not, re-enter it in the client application. Also verify that the selected model name for the device is correct.

Device address '<address>' is Read Only

Error Type:

Warning

Possible Cause:
A tag address that has been specified statically has a requested access mode that is not compatible with what the device supports for that address.

Solution:
Change the access mode in the client application.

Failure to initiate 'winsock.dll'

Error Type:
Fatal

Possible Cause:
Could not negotiate with the operating system's winsock 1.1 functionality.

Solution:
Verify that the winsock.dll is properly installed on the system.

Missing address

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically has no length.

Solution:
Re-enter the address in the client application.

Unable to write to address '<address>' on device '<device>': Device responded with exception code '<code>'

Error Type:
Warning

Possible Cause:
See **Modbus Exception Codes** for a description of the exception code.

Solution:
See **Modbus Exception Codes**.

Unable to write to '<address>' on device '<device name>'

Error Type:
Serious

Possible Cause:

1. The named device may not be connected to the network.
2. The named device may have been assigned an incorrect Network ID.
3. The named device is not responding to write requests.
4. The address does not exist in the PLC.

Solution:

1. Check the PLC network connections.
2. Verify the Network ID given to the named device matches that of the actual device.

Modbus Exception Codes

The following data is from Modbus Application Protocol Specifications documentation.

Code Dec/Hex	Name	Meaning
01/0x01	ILLEGAL FUNCTION	The function code received in the query is not an allowable action for the server. This may be because the function code is only applicable to newer devices, and was not implemented in the unit selected. It could also indicate that the server is in the wrong state to process a request of this type, for example, because it is unconfigured and is being asked to return register values.
02/0x02	ILLEGAL DATA ADDRESS	The data address received in the query is not an allowable address for the server. More specifically, the combination of reference number and transfer length is invalid. For a controller with 100 registers, a request with offset 96 and length 4 would succeed. A request with offset 96 and length 5 generates exception 02.
03/0x03	ILLEGAL DATA VALUE	A value contained in the query data field is not an allowable value for server. This indicates a fault in the structure of the remainder of a complex request, such as that the implied length is incorrect. It specifically does not mean that a data item submitted for storage in a register has a value outside the expectation of the application program, since the Modbus protocol is unaware of the significance of any particular value of any particular register.
04/0x04	SERVER DEVICE FAILURE	An unrecoverable error occurred while the server was attempting to perform the requested action.
05/0x05	ACKNOWLEDGE	The server has accepted the request and is processing it, but a long duration of time is required to do so. This response is returned to prevent a timeout error from occurring in the client. The client can next issue a Poll Program Complete message to determine if processing is completed.
06/0x06	SERVER DEVICE BUSY	The server is engaged in processing a long-duration program command. The client should retransmit the message later when the server is free.
07/0x07	NEGATIVE ACKNOWLEDGE	The server cannot perform the program function received in the query. This code is returned for an unsuccessful programming request using function code 13 or 14 decimal. The client should request diagnostic or error information from the server.
08/0x08	MEMORY PARITY ERROR	The server attempted to read extended memory, but detected a parity error in the memory. The client can retry the request, but service may be required on the server device.
10/0x0A	GATEWAY PATH UNAVAILABLE	Specialized use in conjunction with gateways indicates that the gateway was unable to allocate an internal communication path from the input port to the output port for processing the request. This usually means that the gateway is misconfigured or overloaded.
11/0x0B	GATEWAY TARGET DEVICE FAILED TO RESPOND	Specialized use in conjunction with gateways indicates that no response was obtained from the target device. This usually means that the device is not present on the network.

Note: For this driver, the terms server and unsolicited are used interchangeably.

Index

E

F

G

I

L

M

N

O

P

R

S

T

U

W