IP S7 LINK SDK
SIMATIC connected Development

Version History

Tested? You want it?

License Model Prices Quotation Order Now

History

2.5.0.0 (released 2024-01-23)

NEW

- Improved evaluation experience.
- DnsDeviceEndPoint can be used with IP address strings and can also be used with IPv6 host names (for .NET Core 3.0 and higher) by using a dual-mode socket.

CHANGED

- Members DnsDeviceEndPoint.Address, DnsDeviceEndPoint.AddressFamily, and DnsDeviceEndPoint.DiscardResolvedAddress() will throw a NotSupportedException.

FIXED

- Issue with wrong values being returned by PlcDeviceConnection.ReadInt64() and PlcDeviceConnection.ReadUInt64(), or when using an PlcInt64 or PlcUInt64 (except for array values).

2.4.0.0 (released 2022-10-11)

NEW

- Maintenance Release.

FIXED

- Issue with ignoring channel type SimaticChannelType.ProgrammerDevice when the SimaticeDeviceType is S7300 or S7400.

2.3.2.0 (released 2022-01-14)

FIXED

- Issue with wrong values being returned by PlcDeviceConnection.ReadValues().
- Issue with NullReferenceException when running the QueryDynamicData sample.
- Issue with still incorrectly returned NoData status when reading multiple PlcValues but one of them exceeds the data block size.
- Issue that caused SimaticBlockInfo.Size to return a wrong number in some cases.
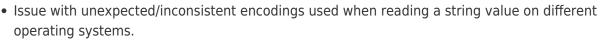
2.3.1.0 (released 2021-12-02)

NEW

- Introduced new IPDeviceEndPoint.Port and IPDeviceEndPoint.IsCustomPort properties including IPDeviceEndPoint.DefaultPort, IPDeviceEndPoint.MinPort and IPDeviceEndPoint.MaxPort fields.

FIXED

- Issue with TypeLoadException being thrown when publishing a .NET application using ReadyToRun.

2.3.0.1 (released 2021-08-03)

FIXED

- Issue with unexpected/inconsistent encodings used when reading a string value on different operating systems.
- Issue with unnecessary string allocations while comparing PlcNames.

2.3.0.0 (released 2021-05-21)

NEW

- Introduced support for LDT using PlcDateTimeLong.
- Unified PlcStatus information among ReadValue and ReadValues operations regarding their PlcStatus.Type information.
- Introduced support of PlcType related PlcStatus evaluation whenever an operation has been completed. This means PlcNotifications.EvaluateStatus is invoked for PlcType related operations as well.
- Introduced logic to correctly determine which data areas could not be read using PlcDeviceConnection.ReadValues. This means using multiple PlcValues to read data which might exceed the size of a data block now results into the correct status and value for all the data areas accessed.

CHANGED

- PlcDeviceConnection.Status does not longer reflect the most recent used PlcType. Use PlcDeviceConnection.GetStatus to determine the status of the PlcType used instead.

FIXED

- Issue with determining data block size for very small data blocks.
- Issue with high CPU usage when the PLC closes the connection with a FIN flag.

2.2.0.3 (released 2021-03-01)

FIXED

- Issue with reference to System.ValueTuple.

2.2.0.2 (released 2020-11-18)

FIXED

- Issues determining object type properties with the same property names used by PlcObject.

2.2.0.1 (released 2020-09-29)

FIXED

- Issue with wrong interpreted channel type SimaticChannelType.Other.

2.2.0.0 (released 2020-09-09)

NEW

- The Licenser class now offers different FailIf… and ThrowIf… methods to verify the used license on-demand by user code. Use FailIf… methods in your DEBUG configurations and use ThrowIf… methods in your RELEASE configurations to protect you from the unlicensed use of the SDK.
- Additional constructors in SimaticDevice to accept an addressOrHostName and the according

SimaticDeviceType.

CHANGED

- License codes used prior v2.2 do not longer work. Customers will receive new license keys working in all frameworks supported.
- All classes with the prefix "Siemens" has been renamed to "Simatic" to better express the product relation especially relating to other products which work with Siemens products.

2.1.2.0

NEW

- Preview release.

2.1.1.0

NEW

- Implemented new SiemensDevice.Options property to further adjust device communication characteristics like the use of a reduced PDU size.

2.1.0.0

NEW

- Integrated IPS7LnkNet code base directly into assembly (= lower file size).
- Prepared the framework for use under .NET Standard and .NET Core 2.0.
- Introduced the LicenseInfo property on the Licenser class to provide some information about the license conditions used.

2.0.1.5

FIXED

- Issue with reading a single value which is not addressable using ReadValues that still a connection wide exception is thrown although the status evaluation continues in case of an exception on PlcValues.

2.0.1.4

FIXED

- Issue with reading data areas reaching the limits of a data block.
- Issue with reading string values longer than 221 characters.

2.0.1.3

FIXED

- Issue with reading a PlcString which caused a general error when the read failed.

2.0.1.2

NEW

- Implemented node traversal to correctly initialize and update nodes recursively.

FIXED

- Issue with wrong handling of node operation results that lead to the wrong status information of the node.
- Issue with nine times connection status evalutions when retrieving device information.

2.0.1.1

NEW

- Corrected ToString implementation in IPDeviceEndPoint to take use of the virtual Address property.

FIXED

- Issue with resolution to IPV6 addresses which are not supported.

2.0.1.0

NEW

- Implemented GetSupportedRawTypes method on PlcType to determine the supported PlcRawTypes to address a specific PlcTypeCode.
- Implemented IsSupported method on PlcType to determine whether a specific PlcTypeCode can accessed by a specific PlcRawType.
- Implemented IsSupportedArrayType method on PlcType to determine whether a PlcTypeCode can be used to define an array type of the current type.
- Implemented MakeArrayType method on PlcType to create PlcType objects of an array type.
- Implemented GetSupportedRawTypes method on PlcOperand to determine the supported PlcRawTypes to address the PLC memory of a specific PlcOperandType.
- Implemented IsSupported method on PlcOperand to determine whether a specific PlcOperandType can be accessed by a specific PlcRawType.

FIXED

- Issue with non unique PlcType hash codes.

2.0.0.0

NEW

- Implemented new Name property on IPlcDevice and added more specific device types to SiemensDeviceType.
- Added implicit string to PlcOperand cast operator.
- Implemented support for the UDT (UserDefinedType) operands.
- Implemented new Symbols property on IPlcDevice.
- Enhanced support for Timer and Counter operands to automatically use the PlcRawType.Byte in case there is no raw type in the address string.
- Extended PlcValue and PlcArray implementation with Name property and implemented new PlcIdentity system to support different identities like addresses, operands and names.
- All PlcValue and PlcArray classes can now be used as symbols.
- The PlcType has been completely reviewed regarding towards the new PlcName and PlcIdentity base class of PlcOperand and PlcAddress.

- Updated documentation of PlcAddress class.
- Implemented further address restrictions for Counter and Timer operand types.
- Implemented new address derivates to represent explicit bit, byte, word and dword addresses.
- Reworked different operand type specific operand fields to properties.
- Implemented specialized sub classes to represent each possible operand type by its own class type.
- Implemented support for UserDefinedType operands.
- Completely reworked PlcType implementation using PlcTypeMetadata.
- Implemented the whole set of possible Of method overloads to represent the possible type metadata combinations.
- Adjusted the exception messages when creating a new PlcType.
- Implemented a whole set of sub classes of the PlcType to represent all different type maps (from PC type to PLC type) as a single unique PlcType implemented to not longer be aware of the internal type representations.
- Implemented for each PlcType sub class the according PLC member information class to define different subsequent members by name/type pairs for other types like the new PlcObjectType.
- Implemented use of the new PLC type specific sub classes.
- Implemented additional ctor's to create instances using predefined types in value/array classes.
- Introduced Types and Values on IPlcDevice.
- Implemented DnsDeviceEndPoint to access a PlcDevice using DNS information.
- Implemented IPlcEntity based read/write operations including result instances on PlcDeviceConnection.
- Implemented a Tag property on PlcObject and PlcMember.
- Reworked GetHashCode implementation in PlcAddress and PlcOperand.

CHANGED

- Reworked ToString implementations in PLC value, type and info classes.
- Renamed Read<T> to ReadObject<T>.
- Renamed Write<T>(T instance) to WriteObject<T>(T instance).
- WriteValuesCore<T>(PlcAddress, T[]) does not longer exist. Replaced by WriteValuesCore<T>(PlcValueType|PlcArrayType, value(s))

FIXED

- Issue with bit array end address information.

1.9.1.1 (released 2019-09-13)

FIXED

- Performance issue with PlcDeviceConnection.ReadValues(IEnumerable<PlcType>).

1.9.1.0 (released 2019-06-24)

CHANGED

- TSAP encoding now uses ISO-8859-1 instead of Windows-1252 for better .NET Standard support.

1.9.0.0 (released 2019-05-15)

NEW

- Prepared the framework for use under .NET Standard 2.0 and .NET Core 2.0 or higher.

1.8.2.3 (released 2019-05-13)

FIXED

- Exception when accessing the IPS7 class directly.

1.8.2.2 (released 2019-05-09)

FIXED

- Issue with connecting to S7-200.

1.8.2.1 (released 2019-04-03)

FIXED

- Issue when loading IPS7LnkNet.Advanced as embedded resource.
- Exception when accessing the LicenseInfo property on the Licenser class.

CHANGED

- In the samples, adjusted methods Read<T>() and Write<T>(T instance) to ReadObject<T>() and WriteObject<T>(T instance) and updated target framework to .NET Framework 4.0.

1.8.2.0 (released 2019-03-20)

NEW

- Support for S7-1500 to determine sizes of data blocks.
- Upgraded to .NET Framework 4.0.
- Introduced the LicenseInfo property on all Licenser class to provide some information about the license conditions used.

1.8.1.1 (released 2018-04-07)

FIXED

- Issue with Virtual device provider which returned NoData in case there is no data written yet.
- Issue with wrong assignment of pooled connection channels when there a new channel is to be created although an existing channel would match the connection setup.

1.8.1.0 (released 2018-04-06)

NEW

- Implemented connection pooling using new channel based device communication layer. This allows to create multiple connection objects using the same lower channel based communication layer in case there the end point and device setup of one or more connections matches together. To enable this feature PlcDeviceConnection.UsePool is to be set to the value true. The default value is false to keep the previous behaviour and to additionally support multiple independent connections e.g. one for read and one for write operations.

FIXED

- Issue with wrong forwarded inner exception which resulted into a ArgumentNullException instead of a expected PlcException with the according InnerException.

- Issues with sometimes not correctly updated status information, especially on PlcType based status information.

1.8.0.2 (released 2018-02-20)

FIXED

- Issue with implicitly using LReal when using "double" as the type of a member in a PlcObject.

1.8.0.1 (released 2017-12-17)

FIXED

- Issue with wrong WriteLReal.

1.8.0.0 (released 2017-12-17)

NEW

- Implemented support for LInt, LWord and LReal data types.
- Implemented support for local and remote TSAP in IPDeviceEndPoint.
- Implemented additional ctors in SiemensDevice to simplify object creation.
- Implemented support for host name based PLC addressing, see DnsDeviceEndPoint.

1.7.3.0 (released 2017-11-14)

NEW

- Implemented additional overload for Read<T>() to read data into an existing PlcObject in case there the concrete PlcObject does not provide a default constructor.

CHANGED

- Renamed methods Read<T>() and Write<T>(T instance) to ReadObject<T>() and WriteObject<T>(T instance) to provide parameterized Read overloads which do not collidate with other Read methods.

1.7.2.4 (released 2017-10-28)

FIXED

- Issue with connecting to Logo 8.x PLC.

1.7.2.3 (released 2017-09-04)

FIXED

- Issue with reading the maximum number of characters in case of using PlcString.GetValue.

1.7.2.2 (released 2017-05-30)

FIXED

- Issue with reading data areas reaching the limits of a data block.
- Issue with reading string values longer than 221 characters.

1.7.2.1 (released 2017-04-08)

FIXED

- Issue reading data using ReadValues(PlcType[]|IEnumabler<PlcType>) : object[], which led to a "shrinked" result. This "shrinked" result was produced in case there the same PlcType is used multiple times the data was only provided for one of them.

1.7.2.0 (released 2016-10-12)

NEW

- Changed sequence/logic within that IPlcValue.Changed events were raised so far. Until this version a Changed event has been raised during reading the value(s) as long the connection is in synchronized state. This mean so far that a subsequent operation using the same connection within the Changed event handler + separate thread results into a deadlock. Beginning from this version (V1.7.2.0) the Changed events are raised afterwards the synchronized state of the connection has been exited.

1.7.1.1 (released 2016-09-28)

FIXED

- Issue with missing week of day portion in PlcDateTime.

1.7.1.0 (released 2016-09-28)

NEW

- Implemented deep-value change verification in PlcArray instances to also raise the Changed event whenever at least one item in the array gets changed during a value read operation.

1.7.0.0 (released 2016-03-11)

NEW

- Implemented optimized Multi-Write (WriteValues) to longer write each value within one transaction. This does now speed up the performance when writing multipe values at once.

1.6.9.2 (released 2016-02-04)

FIXED

- Removed internal buffered read request cache to improve low level driver optimization. This does also fix issues with reading the same string values at once.

1.6.9.1 (released 2016-01-20)

NEW

- Upgraded to IPS7LnkNet v1.1.77.6

FIXED

- MultiRead using PlcString since v1.1.77.5: In case of using MultiRead to read the same PlcString multiple times the driver decremented the ength of the string. As soon the length reached a value

lower than zero the MultiRead did throw an exception.

1.6.9.0 (released 2015-12-27)

NEW

- Implemented method to retrieve PLC block information like the existence or size of a DataBlock defined in a PLC device.
- Implemented PlcOperand creation using string values.

1.6.8.1 (released 2015-12-16)

FIXED

- Issue with ReadValues using PlcString data which addresses data areas which reach the end of a datablock.

1.6.8.0 (released 2015-09-15)

FIXED

- Issue with unresolved locking situations in multi-threaded environments when an exception occured during the connection is about being used by multiple threads at the same time.

1.6.7.2 (released 2015-09-04)

NEW

- New product icon.

1.6.7.1 (released 2015-03-03)

FIXED

- InvalidCastException while reading a Bool[] using ReadBoolean.

1.6.7.0 (released 2015-02-26)

NEW

- Implemented SiemensChannelType enumeration on SiemensDevice to manage the type of channel to use on a connection created by the SiemensDevice instead to manage it via the generic PlcDeviceConnection using the PlcDeviceConnectionType. In general a channel type is only relevant for the Siemens provider. Therefore the PlcDeviceConnectionType will become obsolete in future releases.
- Implemented IsDisposed property to internal indiciate, that a PlcDeviceConnection is diposed of.
- Improved GetInfo() method on SiemensDeviceConnection to not longer produce re-occuring connection status changes and status evaluations.
- Implemented check in PlcType to disallow the use of a different PlcAddress.RawType as Bit for PlcType instances with PlcTypeCode.Bool.

FIXED

- Wrong calculation of End Address of PlcType instances representing a Bool[].

1.6.6.1 (released 2015-02-24)

NEW

- Reworked implementation to better take use of the maximum string length property (Length). Additionally care about the PlcString.MinLength and PlcString.MaxLength and do not longer pad value with whitespaces.
- Fixed issue with out dated return data from ReadValues and simplified some Read methods to take more use of the EnumerableHelper.

FIXED

- Issue with wrong type of buffer array (changed short buffer to float buffer).

1.6.6.0 (released 2015-02-23)

NEW

- Reworked PlcStatus to internally LatestXXX properties to directly get/set the current status information without the need to end initialization to get the latest status information while processing within a PLC operation context.
- In case of Multi-Read (ReadValues) and Multi-Write (WriteValues) the connection does not longer reflect data status codes. Data status codes are now stored on the connection using the PlcType used and on each IPlcValue instance specific PlcStatus.

1.6.5.0 (released 2015-02-19)

NEW

- Improved Exception and Status information.
- Improved Status and Value change handling in transactional way.

FIXED

- Multi-Threaded problem in PlcType.Of(...)

1.6.4.0 (released 2015-02-09)

NEW

- Implemented new overloads for GetValue and SetValue. Additionally implemented ValidateValue. Furthermore reworked mechanism to get and set the Value property and the implementation of GetValue and SetValue to better integrate with new solutions and provide more meaningful codeflows for deriving classes.
- Upgraded to IPS7LnkNet v1.0.74.0

FIXED

- Fixed issues with reading and writing char values.
- Connection to PLC didn't repair itself automatically.
- Using threads with connections: License expired appeard several times.

1.6.3.4 (released 2014-12-15)

FIXED

- Issue with duplicate PlcOperand instances in multi-threaded environments.

1.6.3.3 (released 2014-12-09)

FIXED

- Issue with reading reference data (array data) using ReadValues on a PlcDeviceConnection instance which lead to overridden data on reading same address areas by the same PlcType and to a reference to the internal data buffer. Furthermore user specific reference data was overridden with a reference to the internal data buffer.

1.6.3.2 (released 2014-11-28)

NEW

- Updated IPS7LnkNet (v1.73): Logo/S7-200 license managment implemented.

FIXED

- Issue with failed auto connect attempts then starting a communication while the network device is disabled at start and enabled during further connect/access attempts.

1.6.2.1 (released 2014-11-23)

FIXED

- Issue with default values when reading data using ReadValues.

1.6.2.0 (released 2014-11-07)

FIXED

- Changed base class of PlcException from ExternalException to Exception, because upon the unerlaying PLC layer does not longer belong to any External (unmanaged) API. Furthermore this change does lead to a fully support of the use in Windows Store Apps.

1.6.1.0 (released 2014-09-12)

NEW

- Added new connection state Disconnected.
- Extended global notitications with new event ConnectionDisconnected.
- Extended abstract base class PlcDeviceConnection with new event Disconnected.
- Reworked connection state handling to determine during each interaction with a connection whether a connection is either in connected or disconnected state.

1.6.0.0 (released 2014-08-28)

NEW

- Added setter to generic IPlcValue<T> interface to also support setting the Value property.
- New event ConnectionCreated on PlcNotifications which occurs always after a new PlcDeviceConnection has been created.
- New events Connecting and Connected on PlcDeviceConnection which inform about that a connection to a device is about and has been fully established.

- New event StateChanged on PlcDeviceConnection which informs about any change to the State property of the PlcDeviceConnection.
- New event ConnectionStateChanged on PlcNotifications which informs about any change to the State property of any PlcDeviceConnection.
- New property IsConnected on PlcDeviceConnection which indicates whether a connection to a device has been fully established.
- New method Connect() on PlcDeviceConnection to manually fully establish a connection to a device.
- The data status cache of a PlcDeviceConnection does not always contain the same PlcStatus instance. This allows to subscribe on status changes on PlcType specific PlcStatus objects.
- The PlcConnectionState enum does now define the new states Connecting and Connected.
- The PlcConnectionStateChangedEventArgs class is used by the new StateChanged event on PlcDeviceConnection and does provide OldState and NewState properties.
- Implemented exception handling in VirtualDeviceConnection to ensure that all PlcDevice provider implementations do behave in the same way.
- Implemented exception handling in WebDeviceConnection to ensure that all PlcDevice provider implementations do behvae in the same way. Additionally adjusted the existing exception handling to match the expected behavior.
- Implemented Exception property on PlcStatus which provides the according PlcException instance initialized with the status information provided by a PlcStatus instance.
- Implemented InnerException property on PlcStatus which provides the most recent Exception instance thrown by an internal operation. This property is additionally synchronized with the SocketCode property.
- Implemented TimeStamp property on PlcStatus which does indicate the DateTime of the most recent update of the status meta data provided by the PlcStatus.
- Implemented Type property on PlcStatus which does provide the PlcType information of the PlcType used by the most recent operation.
- All abstract Read/Write methods on PlcDeviceConnection are now called using at least one PlcType. This instance does now obsolete previous parameters likes address:PlcAddress, count:int or length:int.

CHANGED

- Removed state change to Faulted after a PlcDeviceConnection has been disposed of.
- The previous PlcDeviceConnectionEventArgs class is now a subclass of PlcNotifications. Because it is specific to the events provided by the PlcNotifications class and does therefore on one hand provide the PlcDeviceConnection affected, because there is no sender provided by the events and on the other hand it would conflict with the new PlcDeviceConnectionEventArgs which does not provide the event source connection.
- Renamed the ValidateStatusDelegate to EvaluateStatusDelegate to better match the concept behind that the callback is used. Additionally added a bool as return type to indicate that framework specific evaluation is to be performed.

1.5.8.2 (released 2014-08-12)

NEW

- Internal Changed event on PlcDeviceEndPoint which occurs on any changes made on an endpoint.
- General exception handling on each API call to the Siemens provider.

FIXED

- Reset cached connection in PlcDevice in case of any endpoint changes.

1.5.8.1 (released 2014-08-21)

FIXED

- Wrong calculation of S5TIME values (cause: high and low byte had to be reversed).

Table of Contents