

PlcDWordAddress Members

Namespace: IPS7Lnk.Advanced

Assemblies: IPS7LnkNet.Advanced.dll, IPS7LnkNet.Advanced.dll

The [PlcDWordAddress](#) type exposes the following members.

Constructors

PlcDWordAddress(PlcOperand, Int32)

Initializes a new instance of the [PlcDWordAddress](#) class using the specified **operand** and **byteNumber**.

C#

```
public PlcDWordAddress(PlcOperand operand, int byteNumber)
```

Parameters

operand [PlcOperand](#)

The [PlcOperand](#) to that the new [PlcDWordAddress](#) refers.

byteNumber [Int32](#)

The number of the byte to that the new [PlcDWordAddress](#) refers.

Exceptions

[ArgumentException](#)

The type of **operand** specified is not supported.

[ArgumentNullException](#)

The **operand** is a null reference (Nothing in Visual Basic).

[ArgumentOutOfRangeException](#)

The **byteNumber** is out of the bounds defined by [MinByteNumber](#) or [MaxByteNumber](#).

PlcDWordAddress(PlcOperandType, Int32)

Initializes a new instance of the [PlcDWordAddress](#) class using the specified **operandType** and **byteNumber**.

C#

```
public PlcDWordAddress(PlcOperandType operandType, int byteNumber)
```

Parameters

operandType [PlcOperandType](#)

The `PlcOperandType` to that the new `PlcDWordAddress` refers.

`byteNumber` `Int32`

The number of the byte to that the new `PlcDWordAddress` refers.

Exceptions

`ArgumentException`

The `operandType` specified requires an operand number.

`ArgumentOutOfRangeException`

The `byteNumber` is out of the bounds defined by `MinByteNumber` or `MaxByteNumber`.

PlcDWordAddress(PlcOperandType, Int32, Int32)

Initializes a new instance of the `PlcDWordAddress` class using the specified `operandType`, `operandNumber` and `byteNumber`.

C#

```
public PlcDWordAddress(PlcOperandType operandType, int operandNumber, int byteNumber)
```

Parameters

`operandType` `PlcOperandType`

The `PlcOperandType` to that the new `PlcDWordAddress` refers.

`operandNumber` `Int32`

The number of the operand to that the new `PlcDWordAddress` refers.

`byteNumber` `Int32`

The number of the byte to that the new `PlcDWordAddress` refers.

Exceptions

`ArgumentException`

The `operandType` specified is not supported or the `operandType` specified does not support an operand number.

`ArgumentOutOfRangeException`

The `operandNumber` or `byteNumber` is out of the bounds defined by `MinOperandNumber`, `MaxOperandNumber`, `MinByteNumber` or `MaxByteNumber`.

PlcDWordAddress(SerializationInfo, StreamingContext)

Initializes a new instance of the `PlcDWordAddress` class with serialized data.

C#

```
protected PlcDWordAddress(SerializationInfo info, StreamingContext context)
```

Parameters

info [Serialization.SerializationInfo](#)

The [Serialization.SerializationInfo](#) that holds the serialized object data about the exception being thrown.

context [Serialization.StreamingContext](#)

The [Serialization.StreamingContext](#) that contains contextual information about the source or destination.

Exceptions

[ArgumentNullException](#)

The info parameter is null reference (Nothing in Visual Basic).

[Serialization.SerializationException](#)

The class name is null reference (Nothing in Visual Basic) or [HResult](#) is zero (0).

Methods

CompareTo(Object)

Compares the current [PlcDWordAddress](#) with the **other**.

C#

```
public override int CompareTo(object other)
```

Parameters

other [Object](#)

The [PlcDWordAddress](#) to compare with this [PlcDWordAddress](#).

Returns

[Int32](#)

A 32-bit signed integer that indicates the relative order of the objects being compared ([CompareTo\(Object\)](#)).

CompareTo(PlcDWordAddress)

Compares the current [PlcDWordAddress](#) with another [PlcDWordAddress](#).

C#

```
public int CompareTo(PlcDWordAddress other)
```

Parameters

other [PlcDWordAddress](#)

The [PlcDWordAddress](#) to compare with this [PlcDWordAddress](#).

Returns

[Int32](#)

A 32-bit signed integer that indicates the relative order of the objects being compared ([CompareTo\(\)](#)).

Equals(Object)

Determines whether the specified **other** is equal to this [PlcDWordAddress](#).

C#

```
public override bool Equals(object other)
```

Parameters

other [Object](#)

The [PlcDWordAddress](#) to compare to the current [PlcDWordAddress](#).

Returns

[Boolean](#)

The value true if the specified [PlcDWordAddress](#) is equal to the current [PlcDWordAddress](#); otherwise the value false.

Equals(PlcDWordAddress)

Determines whether the specified **other** is equal to this [PlcDWordAddress](#).

C#

```
public bool Equals(PlcDWordAddress other)
```

Parameters

other [PlcDWordAddress](#)

The [PlcDWordAddress](#) to compare to the current [PlcDWordAddress](#).

Returns

Boolean

The value true if the specified [PlcDWordAddress](#) is equal to the current [PlcDWordAddress](#); otherwise the value false.

GetHashCode()

Retrieves a hash code for this [PlcDWordAddress](#).

C#

```
public override int GetHashCode()
```

Returns

Int32

An [Int32](#) that contains the hash code for the [PlcDWordAddress](#).

Operators

Equality(PlcDWordAddress, PlcDWordAddress)

Returns a value indicating whether two instance of [PlcDWordAddress](#) are equal.

C#

```
public static bool operator ==(PlcDWordAddress left, PlcDWordAddress right)
```

GreaterThan(PlcDWordAddress, PlcDWordAddress)

Determines whether the first specified [PlcDWordAddress](#) object is greater than the second specified [PlcDWordAddress](#) object.

C#

```
public static bool operator >(PlcDWordAddress left, PlcDWordAddress right)
```

GreaterThanOrEqual(PlcDWordAddress, PlcDWordAddress)

Determines whether the first specified [PlcDWordAddress](#) object is greater than or equal to the second specified [PlcDWordAddress](#) object.

C#

```
public static bool operator >=(PlcDWordAddress left, PlcDWordAddress right)
```

Implicit(String to PlcDWordAddress)

Converts a string formatted as address to an [PlcDWordAddress](#) object.

C#

```
public static implicit operator PlcDWordAddress(string value)
```

Exceptions

[FormatException](#)

The **value** is not a valid PLC word address.

Inequality(PlcDWordAddress, PlcDWordAddress)

Returns a value indicating whether two instances of [PlcDWordAddress](#) are not equal.

C#

```
public static bool operator !=(PlcDWordAddress left, PlcDWordAddress right)
```

LessThan(PlcDWordAddress, PlcDWordAddress)

Determines whether the first specified [PlcDWordAddress](#) object is less than the second specified [PlcDWordAddress](#) object.

C#

```
public static bool operator <(PlcDWordAddress left, PlcDWordAddress right)
```

Exceptions

[ArgumentNullException](#)

The **left** is a null reference (Nothing in Visual Basic).

LessThanOrEqual(PlcDWordAddress, PlcDWordAddress)

Determines whether the first specified [PlcDWordAddress](#) object is less than or equal to the second [PlcDWordAddress](#) object.

C#

```
public static bool operator <=(PlcDWordAddress left, PlcDWordAddress right)
```

Exceptions

[ArgumentNullException](#)

The **left** is a null reference (Nothing in Visual Basic).

Table of Contents

Constructors	1
PlcDWordAddress(PlcOperand, Int32)	1
PlcDWordAddress(PlcOperandType, Int32)	1
PlcDWordAddress(PlcOperandType, Int32, Int32)	2
PlcDWordAddress(SerializationInfo, StreamingContext)	2
Methods	3
CompareTo(Object)	3
CompareTo(PlcDWordAddress)	3
Equals(Object)	4
Equals(PlcDWordAddress)	4
GetHashCode()	5
Operators	5
Equality(PlcDWordAddress, PlcDWordAddress)	5
GreaterThan(PlcDWordAddress, PlcDWordAddress)	5
GreaterThanOrEqual(PlcDWordAddress, PlcDWordAddress)	5
Implicit(String to PlcDWordAddress)	6
Inequality(PlcDWordAddress, PlcDWordAddress)	6
LessThan(PlcDWordAddress, PlcDWordAddress)	6
LessThanOrEqual(PlcDWordAddress, PlcDWordAddress)	6

