

IPS7WrMulti Members

Namespace: IPS7Lnk

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

The [IPS7WrMulti](#) type exposes the following members.

Constructors

IPS7WrMulti()

Constructor

C#

```
public IPS7WrMulti()
```

Properties

Cnt

Anzahl der Elemente des Types [DataType](#)

C#

```
public int Cnt { get; set; }
```

Property Value

[Int32](#)

DataArea

Datenbereich in der SPS, siehe: [DA](#)

C#

```
public int DataArea { get; set; }
```

Property Value

[Int32](#)

DataType

Datentyp in der SPS, , siehe: [S7DataTypes](#)

C#

```
public int DataType { get; set; }
```

Property Value

Int32

DBNr

Nummer des Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

C#

```
public int DBNr { get; set; }
```

Property Value

Int32

ErrMemSize

MemSize Fehler, -20, wenn user zu kleines Array übergibt/>

C#

```
public int ErrMemSize { get; set; }
```

Property Value

Int32

PcDataType

Hier wird der Datentyp im PC abgelegt.

C#

```
public int PcDataType { get; set; }
```

Property Value

Int32

Result

Nach erfolgreichem Aufruf, wird hier das Ergebnis für diesen Eintrag abgelegt siehe: [Result](#)

C#

```
public int Result { get; set; }
```

Property Value

Int32

Start

Nummer des ersten Elements **DataType** in der SPS

C#

```
public int Start { get; set; }
```

Property Value

Int32

StartBit

Nummer des ersten Bits in der SPS (wird nur bei Bitfunktionen ausgewertet)

C#

```
public int StartBit { get; set; }
```

Property Value

Int32

UserData0

UserData0 und **Userdata1** sind Elemente, die der Aufrufer frei verwenden kann, um z.B. Eigenschaften für diesen Eintrag zu kennzeichnen.

C#

```
public int UserData0 { get; set; }
```

Property Value

Int32

UserData1

UserData0 und **Userdata1** sind Elemente, die der Aufrufer frei verwenden kann, um z.B. Eigenschaften für diesen Eintrag zu kennzeichnen.

C#

```
public int UserData1 { get; set; }
```

Property Value

Int32

Methods

Bit(Char, Int32, Int32, Int32, Int32, Byte)

Bit-Zugriff: Datentyp der Variable im PC: ref byte

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, byte Buf)
```

Parameters

DataArea Char

Datenbereich, siehe: DA

DBNr Int32

Nummer des gewünschten Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

Start Int32

Nummer des Bytes, ab welchem der Lesevorgang gestartet werden soll.

StartBit Int32

Nummer des ersten Bits, ab welchem der Lesevorgang gestartet werden soll.

Cnt Int32

Anzahl der zu lesenden Elemente.

Buf Byte

Pointer auf den Speicher der Daten im Anwenderprogramm.

Returns

Int32

Auftrag wurde eingetragen, IPS7.Result.E_PC_BUFSIZE (-20) Bei Array, das übergebene Array ist zu klein.

Bit(Char, Int32, Int32, Int32, Int32, Byte[])

Bit-Zugriff, Datentyp der Variable im PC: byte[] Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, byte[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Byte[]

Returns

Int32

Remarks

Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

Bit(Char, Int32, Int32, Int32, Int32, Char)

Bit-Zugriff, Datentyp der Variable im PC: ref char Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, char Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Char

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, Char[])

Bit-Zugriff, Datentyp der Variable im PC: sbyte[] Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, char[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Char[]

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, Double)

Bit-Zugriff, Datentyp der Variable im PC: ref double Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, double Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Double

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, Double[])

Bit-Zugriff, Datentyp der Variable im PC: double[] Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, double[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Double[]

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, Int16)

Bit-Zugriff, Datentyp der Variable im PC: ref short Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, short Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Int16

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, Int16[])

Bit-Zugriff, Datentyp der Variable im PC: short[] Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, short[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Int16[]

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, Int32)

Bit-Zugriff, Datentyp der Variable im PC: ref int Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, int Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Int32

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, Int32[])

Bit-Zugriff, Datentyp der Variable im PC: int[] Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, int[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Int32[]

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, SByte)

Bit-Zugriff, Datentyp der Variable im PC: ref sbyte Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, sbyte Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf SByte

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, SByte[])

Bit-Zugriff, Datentyp der Variable im PC: sbyte[] Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, sbyte[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf SByte[]

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, Single)

Bit-Zugriff, Datentyp der Variable im PC: ref float Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, float Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Single

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, Single[])

Bit-Zugriff, Datentyp der Variable im PC: float[] Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, float[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf Single[]

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, UInt16)

Bit-Zugriff, Datentyp der Variable im PC: ref ushort Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, ushort Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf UInt16

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, UInt16[])

Bit-Zugriff, Datentyp der Variable im PC: ushort[] Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, ushort[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf UInt16[]

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, UInt32)

Bit-Zugriff, Datentyp der Variable im PC: ref uint Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, uint Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf UInt32

Returns

Int32

Bit(Char, Int32, Int32, Int32, Int32, UInt32[])

Bit-Zugriff, Datentyp der Variable im PC: uint[] Beschreibung: Bit(Char, Int32, Int32, Int32, Int32, Byte%40)

C#

```
public int Bit(char DataArea, int DBNr, int Start, int StartBit, int Cnt, uint[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

StartBit Int32

Cnt Int32

Buf UInt32[]

Returns

Int32

Byte(Char, Int32, Int32, Int32, Byte)

Byte-Zugriff: Datentyp der Variable im PC: ref byte

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, byte Buf)
```

Parameters

DataArea Char

Datenbereich, siehe: DA

DBNr Int32

Nummer des gewünschten Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

Start Int32

Nummer des Bytes, ab welchem der Lesevorgang gestartet werden soll.

Cnt Int32

Anzahl der zu lesenden Elemente.

Buf Byte

Pointer auf den Speicher der Daten im Anwenderprogramm.

Returns

Int32

Auftrag wurde eingetragen, IPS7.Result.E_PC_BUFSIZE (-20) Bei Array, das übergebene Array ist zu klein.

Byte(Char, Int32, Int32, Int32, Byte[])

Byte-Zugriff, Datentyp der Variable im PC: byte[] Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, byte[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Byte[]

Returns

Int32

Byte(Char, Int32, Int32, Int32, Char)

Byte-Zugriff, Datentyp der Variable im PC: char Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, char Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Char

Returns

Int32

Byte(Char, Int32, Int32, Int32, Char[])

Byte-Zugriff, Datentyp der Variable im PC: char[] Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, char[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Char[]

Returns

Int32

Byte(Char, Int32, Int32, Int32, Double)

Byte-Zugriff, Datentyp der Variable im PC: double Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, double Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Double

Returns

Int32

Byte(Char, Int32, Int32, Int32, Double[])

Byte-Zugriff, Datentyp der Variable im PC: double[] Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, double[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Double[]

Returns

Int32

Byte(Char, Int32, Int32, Int32, Int16)

Byte-Zugriff, Datentyp der Variable im PC: short Beschreibung: `Byte(Char, Int32, Int32, Int32, Byte%40)`

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, short Buf)
```

Parameters

`DataArea` Char

`DBNr` Int32

`Start` Int32

`Cnt` Int32

`Buf` Int16

Returns

Int32

Byte(Char, Int32, Int32, Int32, Int16[])

Byte-Zugriff, Datentyp der Variable im PC: short[] Beschreibung: `Byte(Char, Int32, Int32, Int32, Byte%40)`

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, short[] Buf)
```

Parameters

`DataArea` Char

`DBNr` Int32

`Start` Int32

`Cnt` Int32

Buf Int16[]

Returns

Int32

Byte(Char, Int32, Int32, Int32, Int32)

Byte-Zugriff, Datentyp der Variable im PC: int Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, int Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int32

Returns

Int32

Byte(Char, Int32, Int32, Int32, Int32[])

Byte-Zugriff, Datentyp der Variable im PC: int[] Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, int[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int32[]

Returns

Int32

Byte(Char, Int32, Int32, Int32, SByte)

Byte-Zugriff, Datentyp der Variable im PC: sbyte Beschreibung: `Byte(Char, Int32, Int32, Int32, Byte%40)`

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, sbyte Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf SByte

Returns

Int32

Byte(Char, Int32, Int32, Int32, SByte[])

Byte-Zugriff, Datentyp der Variable im PC: sbyte[] Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, sbyte[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf SByte[]

Returns

Int32

Byte(Char, Int32, Int32, Int32, Single)

Byte-Zugriff, Datentyp der Variable im PC: float Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, float Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Single

Returns

Int32

Byte(Char, Int32, Int32, Int32, Single[])

Byte-Zugriff, Datentyp der Variable im PC: float[] Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, float[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Single[]

Returns

Int32

Byte(Char, Int32, Int32, Int32, UInt16)

Byte-Zugriff, Datentyp der Variable im PC: ushort Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, ushort Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt16

Returns

Int32

Byte(Char, Int32, Int32, Int32, UInt16[])

Byte-Zugriff, Datentyp der Variable im PC: ushort[] Beschreibung: Byte(Char, Int32, Int32, Int32, Byte%40)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, ushort[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt16[]

Returns

Int32

Byte(Char, Int32, Int32, Int32, UInt32)

Byte-Zugriff, Datentyp der Variable im PC: uint Beschreibung: [Byte\(Char, Int32, Int32, Int32, Byte%40\)](#)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, uint Buf)
```

Parameters

[DataArea](#) [Char](#)

[DBNr](#) [Int32](#)

[Start](#) [Int32](#)

[Cnt](#) [Int32](#)

[Buf](#) [UInt32](#)

Returns

[Int32](#)

Byte(Char, Int32, Int32, Int32, UInt32[])

Byte-Zugriff, Datentyp der Variable im PC: uint[] Beschreibung: [Byte\(Char, Int32, Int32, Int32, Byte%40\)](#)

C#

```
public int Byte(char DataArea, int DBNr, int Start, int Cnt, uint[] Buf)
```

Parameters

[DataArea](#) [Char](#)

[DBNr](#) [Int32](#)

[Start](#) [Int32](#)

[Cnt](#) [Int32](#)

Buf UInt32[]

Returns

Int32

Counter(Char, Int32, Int32, Int32, Char[])

Zähler/ Counter Datentyp der Variable im PC: char [] Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, char[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Char[]

Returns

Int32

Counter(Char, Int32, Int32, Int32, Double[])

Zähler/ Counter Datentyp der Variable im PC: double [] Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, double[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Double[]

Returns

Int32

Counter(Char, Int32, Int32, Int32, Int16[])

Zähler/ Counter Datentyp der Variable im PC: short [] Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, short[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int16[]

Returns

Int32

Counter(Char, Int32, Int32, Int32, Int32[])

Zähler/ Counter Datentyp der Variable im PC: int [] Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, int[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int32[]

Returns

Int32

Counter(Char, Int32, Int32, Int32, ref Char)

Zähler/ Counter Datentyp der Variable im PC: char Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, ref char Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Char

Returns

Int32

Counter(Char, Int32, Int32, Int32, ref Double)

Zähler/ Counter Datentyp der Variable im PC: double Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, ref double Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Double

Returns

Int32

Counter(Char, Int32, Int32, Int32, ref Int16)

Zähler/Counter Datentyp der Variable im PC: short

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, ref short Buf)
```

Parameters

DataArea Char

Datenbereich, siehe: [DA](#)

DBNr Int32

Nummer des gewünschten Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

Start Int32

Nummer des Bytes, ab welchem der Lesevorgang gestartet werden soll.

Cnt Int32

Anzahl der zu lesenden Elemente.

Buf Int16

Pointer auf den Speicher der Daten im Anwenderprogramm.

Returns

Int32

Auftrag wurde eingetragen, IPS7.Result.E_PC_BUFSIZE (-20) Bei Array, das übergebene Array ist zu klein.

Counter(Char, Int32, Int32, Int32, ref Int32)

Zähler/ Counter Datentyp der Variable im PC: int Beschreibung: [Counter\(Char, Int32, Int32, Int32, Int16%40\)](#)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, ref int Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int32

Returns

Int32

Counter(Char, Int32, Int32, Int32, ref Single)

Zähler/ Counter Datentyp der Variable im PC: float Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, ref float Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Single

Returns

Int32

Counter(Char, Int32, Int32, Int32, ref UInt16)

Zähler/ Counter Datentyp der Variable im PC: ushort [] Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, ref ushort Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt16

Returns

Int32

Counter(Char, Int32, Int32, Int32, ref UInt32)

Zähler/ Counter Datentyp der Variable im PC: uint Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, ref uint Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt32

Returns

Int32

Counter(Char, Int32, Int32, Int32, Single[])

Zähler/ Counter Datentyp der Variable im PC: float [] Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, float[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Single[]

Returns

Int32

Counter(Char, Int32, Int32, Int32, UInt16[])

Zähler/ Counter Datentyp der Variable im PC: ushort [] Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, ushort[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt16[]

Returns

Int32

Counter(Char, Int32, Int32, Int32, UInt32[])

Zähler/ Counter Datentyp der Variable im PC: uint [] Beschreibung: Counter(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Counter(char DataArea, int DBNr, int Start, int Cnt, uint[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt32[]

Returns

Int32

DWord(Char, Int32, Int32, Int32, Int32)

DWord (32 Bit)-Zugriff: Datentyp der Variable im PC: int

C#

```
public int DWord(char DataArea, int DBNr, int Start, int Cnt, int Buf)
```

Parameters

DataArea Char

Datenbereich, siehe: [DA](#)

DBNr Int32

Nummer des gewünschten Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

Start Int32

Nummer des Bytes, ab welchem der Lesevorgang gestartet werden soll.

Cnt Int32

Anzahl der zu lesenden Elemente.

Buf Int32

Pointer auf den Speicher der Daten im Anwenderprogramm.

Returns

Int32

Auftrag wurde eingetragen, IPS7.Result.E_PC_BUFSIZE (-20) Bei Array, das übergebene Array ist zu klein.

DWord(Char, Int32, Int32, Int32, Int32[])

DWord (32 Bit)-Zugriff: Datentyp der Variable im PC: int[] Beschreibung: [DWord\(Char, Int32, Int32, Int32, Int32%40\)](#)

C#

```
public int DWord(char DataArea, int DBNr, int Start, int Cnt, int[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int32[]

Returns

Int32

DWord(Char, Int32, Int32, Int32, UInt32)

DWord (32 Bit)-Zugriff: Datentyp der Variable im PC: uint Beschreibung: DWord(Char, Int32, Int32, Int32, Int32%40)

C#

```
public int DWord(char DataArea, int DBNr, int Start, int Cnt, uint Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt32

Returns

Int32

DWord(Char, Int32, Int32, Int32, UInt32[])

DWord (32 Bit)-Zugriff: Datentyp der Variable im PC: uint[] Beschreibung: DWord(Char, Int32, Int32, Int32, Int32%40)

C#

```
public int DWord(char DataArea, int DBNr, int Start, int Cnt, uint[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt32[]

Returns

Int32

Int16(Char, Int32, Int32, Int32, Int16)

short (16 Bit signed)-Zugriff: Datentyp der Variable im PC: short

C#

```
public int Int16(char DataArea, int DBNr, int Start, int Cnt, short Buf)
```

Parameters

DataArea Char

Datenbereich, siehe: DA

DBNr Int32

Nummer des gewünschten Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

Start Int32

Nummer des Bytes, ab welchem der Lesevorgang gestartet werden soll.

Cnt Int32

Anzahl der zu lesenden Elemente.

Buf Int16

Pointer auf den Speicher der Daten im Anwenderprogramm.

Returns

Int32

Auftrag wurde eingetragen, IPS7.Result.E_PC_BUFSIZE (-20) Bei Array, das übergebene Array ist zu klein.

Int16(Char, Int32, Int32, Int32, Int16[])

short (16 Bit signed)-Zugriff: Datentyp der Variable im PC: short[] Beschreibung: [Int16\(Char, Int32, Int32, Int32, Int16%40\)](#)

C#

```
public int Int16(char DataArea, int DBNr, int Start, int Cnt, short[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int16[]

Returns

Int32

Int16(Char, Int32, Int32, Int32, UInt16)

short (16 Bit signed)-Zugriff: Datentyp der Variable im PC: ushort Beschreibung: [Int16\(Char, Int32, Int32, Int32, Int16%40\)](#)

C#

```
public int Int16(char DataArea, int DBNr, int Start, int Cnt, ushort Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt16

Returns

Int32

Int16(Char, Int32, Int32, Int32, UInt16[])

short (16 Bit signed)-Zugriff: Datentyp der Variable im PC: ushort[] Beschreibung: Int16(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Int16(char DataArea, int DBNr, int Start, int Cnt, ushort[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt16[]

Returns

Int32

Int32(Char, Int32, Int32, Int32, Int32)

int (32-Bit signed)-Zugriff: Datentyp der Variable im PC: int

C#

```
public int Int32(char DataArea, int DBNr, int Start, int Cnt, int Buf)
```


Parameters

DataArea Char

Datenbereich, siehe: [DA](#)

DBNr Int32

Nummer des gewünschten Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

Start Int32

Nummer des Bytes, ab welchem der Lesevorgang gestartet werden soll.

Cnt Int32

Anzahl der zu lesenden Elemente.

Buf Int32

Pointer auf den Speicher der Daten im Anwenderprogramm.

Returns

Int32

Auftrag wurde eingetragen, IPS7.Result.E_PC_BUFSIZE (-20) Bei Array, das übergebene Array ist zu klein.

Int32(Char, Int32, Int32, Int32, Int32[])

int (32-Bit signed)-Zugriff: Datentyp der Variable im PC: int[] Beschreibung: [Int32\(Char, Int32, Int32, Int32, Int32%40\)](#)

C#

```
public int Int32(char DataArea, int DBNr, int Start, int Cnt, int[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int32[]

Returns

Int32

Int32(Char, Int32, Int32, Int32, UInt32)

int (32-Bit signed)-Zugriff: Datentyp der Variable im PC: uint Beschreibung: [Int32\(Char, Int32, Int32, Int32, Int32%40\)](#)

C#

```
public int Int32(char DataArea, int DBNr, int Start, int Cnt, uint Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt32

Returns

Int32

Int32(Char, Int32, Int32, Int32, UInt32[])

int (32-Bit signed)-Zugriff: Datentyp der Variable im PC: uint[] Beschreibung: [Int32\(Char, Int32, Int32, Int32, Int32%40\)](#)

C#

```
public int Int32(char DataArea, int DBNr, int Start, int Cnt, uint[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt32[]

Returns

Int32

Int64(Char, Int32, Int32, Int32, Int64)

Lint (64-Bit signed)-Zugriff: Datentyp der Variable im PC: long

C#

```
public int Int64(char DataArea, int DBNr, int Start, int Cnt, long Buf)
```

Parameters

DataArea Char

Datenbereich, siehe: DA

DBNr Int32

Nummer des gewünschten Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

Start Int32

Nummer des Bytes, ab welchem der Lesevorgang gestartet werden soll.

Cnt Int32

Anzahl der zu lesenden Elemente.

Buf Int64

Pointer auf den Speicher der Daten im Anwenderprogramm.

Returns

Int32

Auftrag wurde eingetragen, IPS7.Result.E_PC_BUFSIZE (-20) Bei Array, das übergebene Array ist zu klein.

Int64(Char, Int32, Int32, Int32, Int64[])

LInt (64-Bit signed)-Zugriff: Datentyp der Variable im PC: long[]

C#

```
public int Int64(char DataArea, int DBNr, int Start, int Cnt, long[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int64[]

Returns

Int32

LReal(Char, Int32, Int32, Int32, Double)

LReal (Fließpunktzahl 64-Bit)-Zugriff: Datentyp der Variable im PC: double Beschreibung: DWord(Char, Int32, Int32, Int32, Int32%40)

C#

```
public int LReal(char DataArea, int DBNr, int Start, int Cnt, double Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Double

Returns

Int32

LReal(Char, Int32, Int32, Int32, Double[])

LReal (Fließpunktzahl 64-Bit)-Zugriff: Datentyp der Variable im PC: double[] Beschreibung: DWord(Char, Int32, Int32, Int32, Int32%40)

C#

```
public int LReal(char DataArea, int DBNr, int Start, int Cnt, double[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Double[]

Returns

Int32

Real(Char, Int32, Int32, Int32, Double)

Real (Fließpunktzahl 32-Bit)-Zugriff: Datentyp der Variable im PC: double Beschreibung: DWord(Char, Int32, Int32, Int32, Int32%40)

C#

```
public int Real(char DataArea, int DBNr, int Start, int Cnt, double Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Double

Returns

Int32

Real(Char, Int32, Int32, Int32, Double[])

Real (Fließpunktzahl 32-Bit)-Zugriff: Datentyp der Variable im PC: double[] Beschreibung: DWord(Char, Int32, Int32, Int32, Int32%40)

C#

```
public int Real(char DataArea, int DBNr, int Start, int Cnt, double[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Double[]

Returns

Int32

Real(Char, Int32, Int32, Int32, Single)

Real (Fließpunktzahl 32-Bit)-Zugriff: Datentyp der Variable im PC: float

C#

```
public int Real(char DataArea, int DBNr, int Start, int Cnt, float Buf)
```

Parameters

DataArea Char

Datenbereich, siehe: [DA](#)

DBNr Int32

Nummer des gewünschten Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

Start Int32

Nummer des Bytes, ab welchem der Lesevorgang gestartet werden soll.

Cnt Int32

Anzahl der zu lesenden Elemente.

Buf Single

Pointer auf den Speicher der Daten im Anwenderprogramm.

Returns

Int32

Auftrag wurde eingetragen, IPS7.Result.E_PC_BUFSIZE (-20) Bei Array, das übergebene Array ist zu klein.

Real(Char, Int32, Int32, Int32, Single[])

Real (Fließpunktzahl 32-Bit)-Zugriff: Datentyp der Variable im PC: float[] Beschreibung: [DWord\(Char, Int32, Int32, Int32, Int32%40\)](#)

C#

```
public int Real(char DataArea, int DBNr, int Start, int Cnt, float[] Buf)
```

Parameters

DataArea Char**DBNr** Int32

Start Int32

Cnt Int32

Buf Single[]

Returns

Int32

SetData(Char, Int32, Int32, Int32, Int32, Int32,
PCDataType, Object)

C#

```
public void SetData(char DataArea, int DBNr, int Cnt, int Start, int StartBit, int  
S7DataType, PCDataType PCDataType, object DataObj)
```

Parameters

DataArea Char

DBNr Int32

Cnt Int32

Start Int32

StartBit Int32

S7DataType Int32

PCDataType PCDataType

DataObj Object

String(Char, Int32, Int32, String)

C#

```
public int String(char DataArea, int DBNr, int Start, string Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Buf String

Returns

Int32

Timer(Char, Int32, Int32, Int32, Double)

Timer Datentyp der Variable im PC: double Beschreibung: [Timer\(Char, Int32, Int32, Int32, Int32%40\)](#)

C#

```
public int Timer(char DataArea, int DBNr, int Start, int Cnt, double Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Double

Returns

Int32

Timer(Char, Int32, Int32, Int32, Double[])

Timer Datentyp der Variable im PC: double [] Beschreibung: [Timer\(Char, Int32, Int32, Int32, Int32%40\)](#)

C#

```
public int Timer(char DataArea, int DBNr, int Start, int Cnt, double[] Buf)
```

Parameters

[DataArea](#) Char

[DBNr](#) Int32

[Start](#) Int32

[Cnt](#) Int32

[Buf](#) Double[]

Returns

Int32

Timer(Char, Int32, Int32, Int32, Int32)

Timer Datentyp der Variable im PC: int

C#

```
public int Timer(char DataArea, int DBNr, int Start, int Cnt, int Buf)
```

Parameters

[DataArea](#) Char

Datenbereich, siehe: [DA](#)

[DBNr](#) Int32

Nummer des gewünschten Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

Start Int32

Nummer des Bytes, ab welchem der Lesevorgang gestartet werden soll.

Cnt Int32

Anzahl der zu lesenden Elemente.

Buf Int32

Pointer auf den Speicher der Daten im Anwenderprogramm.

Returns

Int32

Auftrag wurde eingetragen, IPS7.Result.E_PC_BUFSIZE (-20) Bei Array, das übergebene Array ist zu klein.

Timer(Char, Int32, Int32, Int32, Int32[])

Timer Datentyp der Variable im PC: int[] Beschreibung: Timer(Char, Int32, Int32, Int32, Int32%40)

C#

```
public int Timer(char DataArea, int DBNr, int Start, int Cnt, int[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int32[]

Returns

Int32

Timer(Char, Int32, Int32, Int32, Single)

Timer Datentyp der Variable im PC: float Beschreibung: [Timer\(Char, Int32, Int32, Int32, Int32%40\)](#)

C#

```
public int Timer(char DataArea, int DBNr, int Start, int Cnt, float Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Single

Returns

Int32

Timer(Char, Int32, Int32, Int32, Single[])

Timer Datentyp der Variable im PC: float[] Beschreibung: [Timer\(Char, Int32, Int32, Int32, Int32%40\)](#)

C#

```
public int Timer(char DataArea, int DBNr, int Start, int Cnt, float[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Single[]

Returns

Int32

Timer(Char, Int32, Int32, Int32, UInt32)

Timer Datentyp der Variable im PC: uint Beschreibung: Timer(Char, Int32, Int32, Int32, Int32%40)

C#

```
public int Timer(char DataArea, int DBNr, int Start, int Cnt, uint Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt32

Returns

Int32

Timer(Char, Int32, Int32, Int32, UInt32[])

Timer Datentyp der Variable im PC: uint[] Beschreibung: Timer(Char, Int32, Int32, Int32, Int32%40)

C#

```
public int Timer(char DataArea, int DBNr, int Start, int Cnt, uint[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt32[]

Returns

Int32

UInt64(Char, Int32, Int32, Int32, UInt64)

ULIN (64-Bit unsigned)-Zugriff: Datentyp der Variable im PC: ulong

C#

```
public int UInt64(char DataArea, int DBNr, int Start, int Cnt, ulong Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt64

Returns

Int32

UInt64(Char, Int32, Int32, Int32, UInt64[])

LUINT (64-Bit signed)-Zugriff: Datentyp der Variable im PC: ulong[]

C#

```
public int UInt64(char DataArea, int DBNr, int Start, int Cnt, ulong[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt64[]

Returns

Int32

Word(Char, Int32, Int32, Int32, Int16)

Word (16 Bit)-Zugriff: Datentyp der Variable im PC: short

C#

```
public int Word(char DataArea, int DBNr, int Start, int Cnt, short Buf)
```

Parameters

DataArea Char

Datenbereich, siehe: [DA](#)

DBNr Int32

Nummer des gewünschten Datenbausteins. Wird nur bei Zugriff auf DB ausgewertet.

Start Int32

Nummer des Bytes, ab welchem der Lesevorgang gestartet werden soll.

Cnt Int32

Anzahl der zu lesenden Elemente.

Buf Int16

Pointer auf den Speicher der Daten im Anwenderprogramm.

Returns

Int32

Auftrag wurde eingetragen, IPS7.Result.E_PC_BUFSIZE (-20) Bei Array, das übergebene Array ist zu klein.

Word(Char, Int32, Int32, Int32, Int16[])

Word (16 Bit)-Zugriff: Datentyp der Variable im PC: short[] Beschreibung: Word(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Word(char DataArea, int DBNr, int Start, int Cnt, short[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf Int16[]

Returns

Int32

Word(Char, Int32, Int32, Int32, UInt16)

Word (16 Bit)-Zugriff: Datentyp der Variable im PC: ushort Beschreibung: Word(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Word(char DataArea, int DBNr, int Start, int Cnt, ushort Buf)
```


Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt16

Returns

Int32

Word(Char, Int32, Int32, Int32, UInt16[])

Word (16 Bit)-Zugriff: Datentyp der Variable im PC: ushort[] Beschreibung: Word(Char, Int32, Int32, Int32, Int16%40)

C#

```
public int Word(char DataArea, int DBNr, int Start, int Cnt, ushort[] Buf)
```

Parameters

DataArea Char

DBNr Int32

Start Int32

Cnt Int32

Buf UInt16[]

Returns

Int32

Table of Contents

Constructors	1
IPS7WrMulti()	1
Properties	1
Cnt	1
DataArea	1
DataType	1
DBNr	2
ErrMemSize	2
PcDataType	2
Result	2
Start	3
StartBit	3
UserData0	3
UserData1	3
Methods	4
Bit(Char, Int32, Int32, Int32, Int32, Byte)	4
Bit(Char, Int32, Int32, Int32, Int32, Byte[])	4
Bit(Char, Int32, Int32, Int32, Int32, Char)	5
Bit(Char, Int32, Int32, Int32, Int32, Char[])	6
Bit(Char, Int32, Int32, Int32, Int32, Double)	7
Bit(Char, Int32, Int32, Int32, Int32, Double[])	7
Bit(Char, Int32, Int32, Int32, Int32, Int16)	8
Bit(Char, Int32, Int32, Int32, Int32, Int16[])	9
Bit(Char, Int32, Int32, Int32, Int32, Int32)	9
Bit(Char, Int32, Int32, Int32, Int32, Int32[])	10
Bit(Char, Int32, Int32, Int32, Int32, SByte)	11
Bit(Char, Int32, Int32, Int32, Int32, SByte[])	11
Bit(Char, Int32, Int32, Int32, Int32, Single)	12
Bit(Char, Int32, Int32, Int32, Int32, Single[])	13
Bit(Char, Int32, Int32, Int32, Int32, UInt16)	13
Bit(Char, Int32, Int32, Int32, Int32, UInt16[])	14
Bit(Char, Int32, Int32, Int32, Int32, UInt32)	15
Bit(Char, Int32, Int32, Int32, Int32, UInt32[])	15
Byte(Char, Int32, Int32, Int32, Byte)	16
Byte(Char, Int32, Int32, Int32, Byte[])	17
Byte(Char, Int32, Int32, Int32, Char)	17
Byte(Char, Int32, Int32, Int32, Char[])	18
Byte(Char, Int32, Int32, Int32, Double)	18
Byte(Char, Int32, Int32, Int32, Double[])	19
Byte(Char, Int32, Int32, Int32, Int16)	20
Byte(Char, Int32, Int32, Int32, Int16[])	20
Byte(Char, Int32, Int32, Int32, Int32)	21
Byte(Char, Int32, Int32, Int32, Int32[])	21
Byte(Char, Int32, Int32, Int32, SByte)	22
Byte(Char, Int32, Int32, Int32, SByte[])	23
Byte(Char, Int32, Int32, Int32, Single)	23
Byte(Char, Int32, Int32, Int32, Single[])	24
Byte(Char, Int32, Int32, Int32, UInt16)	24
Byte(Char, Int32, Int32, Int32, UInt16[])	25
Byte(Char, Int32, Int32, Int32, UInt32)	26
Byte(Char, Int32, Int32, Int32, UInt32[])	26

Counter(Char, Int32, Int32, Int32, Char[])	27
Counter(Char, Int32, Int32, Int32, Double[])	27
Counter(Char, Int32, Int32, Int32, Int16[])	28
Counter(Char, Int32, Int32, Int32, Int32[])	29
Counter(Char, Int32, Int32, Int32, ref Char)	29
Counter(Char, Int32, Int32, Int32, ref Double)	30
Counter(Char, Int32, Int32, Int32, ref Int16)	30
Counter(Char, Int32, Int32, Int32, ref Int32)	31
Counter(Char, Int32, Int32, Int32, ref Single)	32
Counter(Char, Int32, Int32, Int32, ref UInt16)	32
Counter(Char, Int32, Int32, Int32, ref UInt32)	33
Counter(Char, Int32, Int32, Int32, Single[])	34
Counter(Char, Int32, Int32, Int32, UInt16[])	34
Counter(Char, Int32, Int32, Int32, UInt32[])	35
DWord(Char, Int32, Int32, Int32, Int32)	35
DWord(Char, Int32, Int32, Int32, Int32[])	36
DWord(Char, Int32, Int32, Int32, UInt32)	37
DWord(Char, Int32, Int32, Int32, UInt32[])	37
Int16(Char, Int32, Int32, Int32, Int16)	38
Int16(Char, Int32, Int32, Int32, Int16[])	39
Int16(Char, Int32, Int32, Int32, UInt16)	39
Int16(Char, Int32, Int32, Int32, UInt16[])	40
Int32(Char, Int32, Int32, Int32, Int32)	40
Int32(Char, Int32, Int32, Int32, Int32[])	41
Int32(Char, Int32, Int32, Int32, UInt32)	42
Int32(Char, Int32, Int32, Int32, UInt32[])	42
Int64(Char, Int32, Int32, Int32, Int64)	43
Int64(Char, Int32, Int32, Int32, Int64[])	44
LReal(Char, Int32, Int32, Int32, Double)	44
LReal(Char, Int32, Int32, Int32, Double[])	45
Real(Char, Int32, Int32, Int32, Double)	45
Real(Char, Int32, Int32, Int32, Double[])	46
Real(Char, Int32, Int32, Int32, Single)	47
Real(Char, Int32, Int32, Int32, Single[])	47
SetData(Char, Int32, Int32, Int32, Int32, Int32, PCDataType, Object)	48
String(Char, Int32, Int32, String)	49
Timer(Char, Int32, Int32, Int32, Double)	49
Timer(Char, Int32, Int32, Int32, Double[])	50
Timer(Char, Int32, Int32, Int32, Int32)	50
Timer(Char, Int32, Int32, Int32, Int32[])	51
Timer(Char, Int32, Int32, Int32, Single)	52
Timer(Char, Int32, Int32, Int32, Single[])	52
Timer(Char, Int32, Int32, Int32, UInt32)	53
Timer(Char, Int32, Int32, Int32, UInt32[])	53
UInt64(Char, Int32, Int32, Int32, UInt64)	54
UInt64(Char, Int32, Int32, Int32, UInt64[])	55
Word(Char, Int32, Int32, Int32, Int16)	55
Word(Char, Int32, Int32, Int32, Int16[])	56
Word(Char, Int32, Int32, Int32, UInt16)	56
Word(Char, Int32, Int32, Int32, UInt16[])	57