

1. Use Uaexpert to connect to OPCUA Server to check if the OPCUA server is running.
2. Edgeline OPCUA master configuration.
 - 1) Add a OPCUA device

The screenshot displays the configuration interface for adding an OPCUA device. It is divided into two main sections: 'General Information' and 'TCP/IP'.

General Information:

- Enable
- Name: NewDevice
- Device Type: OPC UA
- Device Model: Double Click to Select Device Template
- Unit Number: 1
- Tag Write Type: Single Write
- Description: (empty text area)
- Add device name as prefix to IO tags
- Bulk Copy button

TCP/IP:

- IP Address: 11.0.0.32
- Port Number: 4840

Device Type: OPCUA

Unit Number: Different from other devices in one interface is ok.

IP Address and Port Number: OPCUA server's IP and Port.

The screenshot shows the 'Extention Properties' configuration window, which includes the following settings:

- Use URL as Connection Address: (empty text field)
- Security Policy - Message Security Mode: None - None
- Authentication Settings: Anonymous
- Username: (empty text field)
- Password: (empty text field)
- Select Certificate: Default Certificate File
- Select Private Key: Default Private Key File
- Cyclic rate of subscribing data changes (ms): 1000

Use URL as Connection Address: If OPCUA server must be connected by its URL, you can choose this.

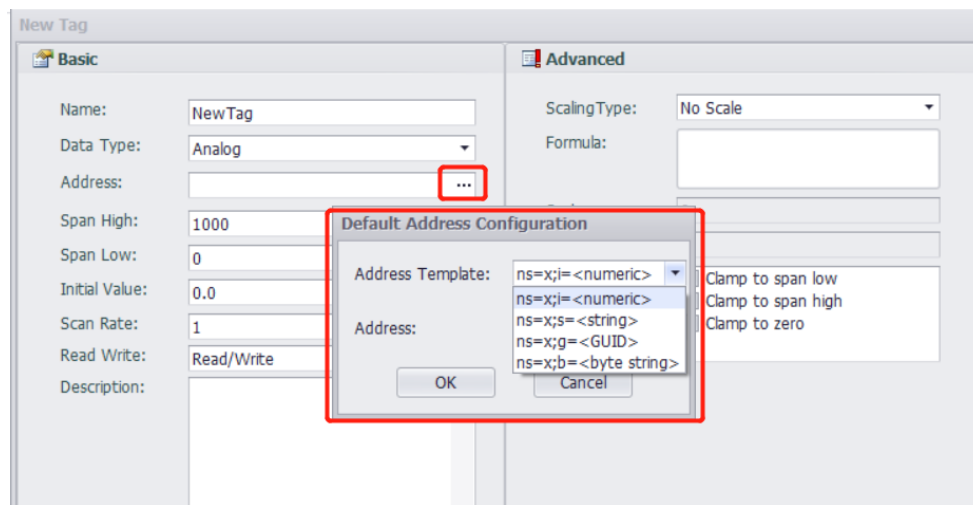
Security Policy and Authentication Setting: Follow the OPCUA server.

Certificate and Private Key: We have self authentication files by default. If customer needs to use the special file, they can choose in here.

Cyclic rate of subscribing data changes: By default, the mode of our opcu master is

Asking-Answer. If choosing this, the mode we use will be: We send a message to OPCUA server to tell it to check itself by cycle if there is some data changed. When some data changing, server will tell us, we don't need to ask server any more.

2) Add tags.



The address grammar is : ns=<Namespace Index>;<type>=<value>

Examples

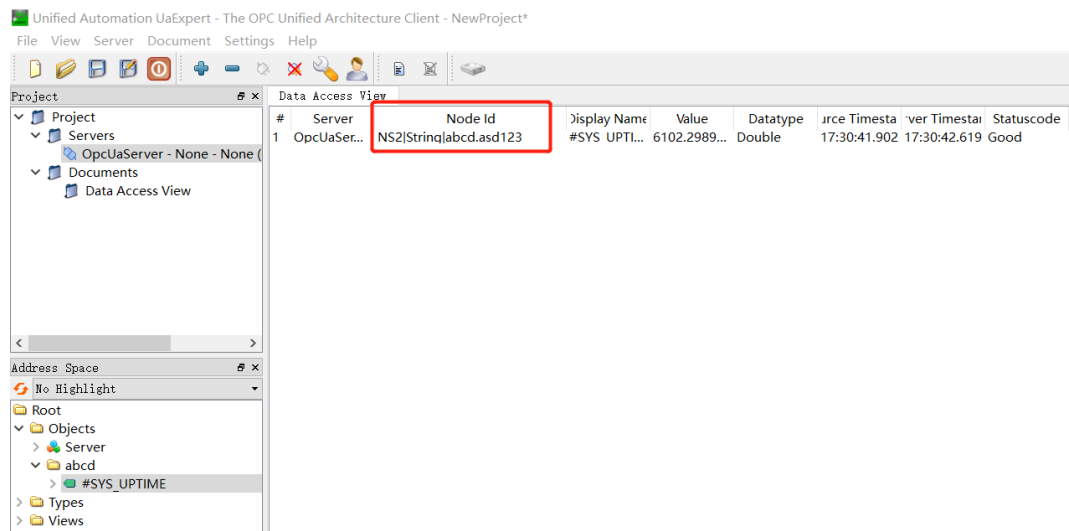
Address Type	Namespace Index	Example
Numeric	2	ns=2;i=1234
String	2	ns=2;s=Device.System_Tag.#SYS_TIME_SECOND
GUID	0	ns=0;g={8ACE8827-ECC3-4c9a-8032-CA1E9957A8E8}
Opaque	2	ns=2;b=M/RbKBsRVkePCePcx24oRA==

The data type in OPCUA we have supported:

Data Type	Description
Boolean	Single bit
Byte	Unsigned 8 bit value
SByte	Signed 8 bit value
UInt16	Unsigned 16 bit value
Int16	Signed 16 bit value

UInt32	Unsigned 32 bit value
Int32	Signed 32 bit value
Float	
Double	
Enumeration	
StatusCode	StatusCode is a 32-bit unsigned integer

We can see the address by Uaexpert.



In this example, we can see “ns=2”, type is string, value is “abcd.asd123”. So the address in edgelinek should be “ns=2;s=abcd.asd123”

Basic

Name:

Data Type:

Address:

Span High:

Span Low:

Initial Value:

Scan Rate:

Read Write:

Description:

Advanced

ScalingType:

Formula:

Scale:

Offset:

Clamp: Clamp to span low
 Clamp to span high
 Clamp to zero

3) Download the project. We can see the result in online monitor.

ECU-1251-R10

Tags

System Information

Configuration

System Tag **IO Tag** User Tag Calculation Tag

IO Tag

Tag Name Tag Value

Tag	Value	Quality	Timestamp
NewDevice:NewTag	6425.75	Good	2019-11-27T09:36:06 (UTC)

Showing 1 to 1 of 1 rows