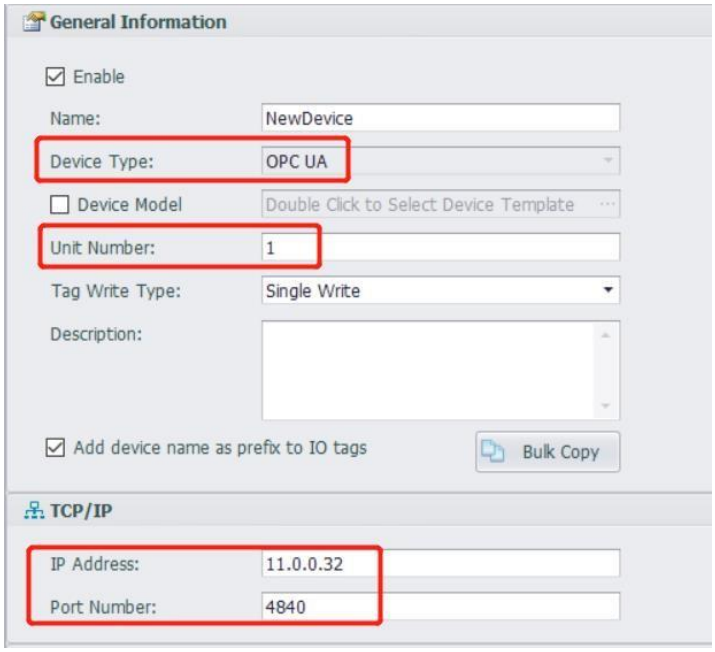
1. Use Uaexpert to connet to OPCUA Server to check if the OPCUA server is running.

2. Edgelink OPCUA master configuration.

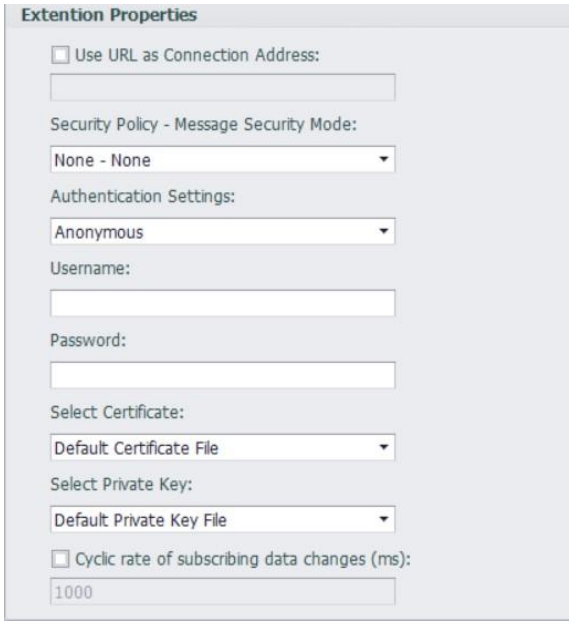
   1) Add a OPCUA device



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.



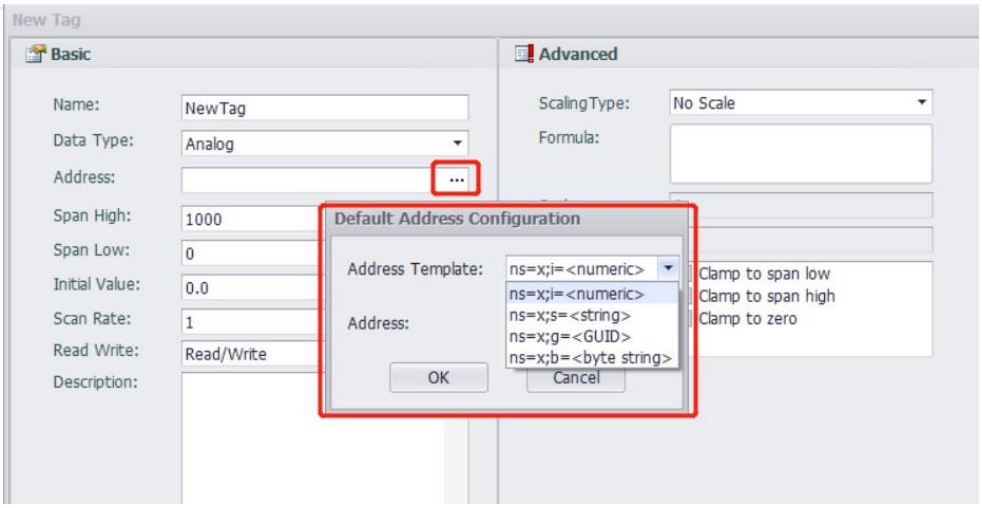
Use URL as Connection Address: If OPCUA server must be connected by its URL, you can choose this.

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 opcua 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 grammer is : ns=<Namespace Index>;<type>=<value>

Examples

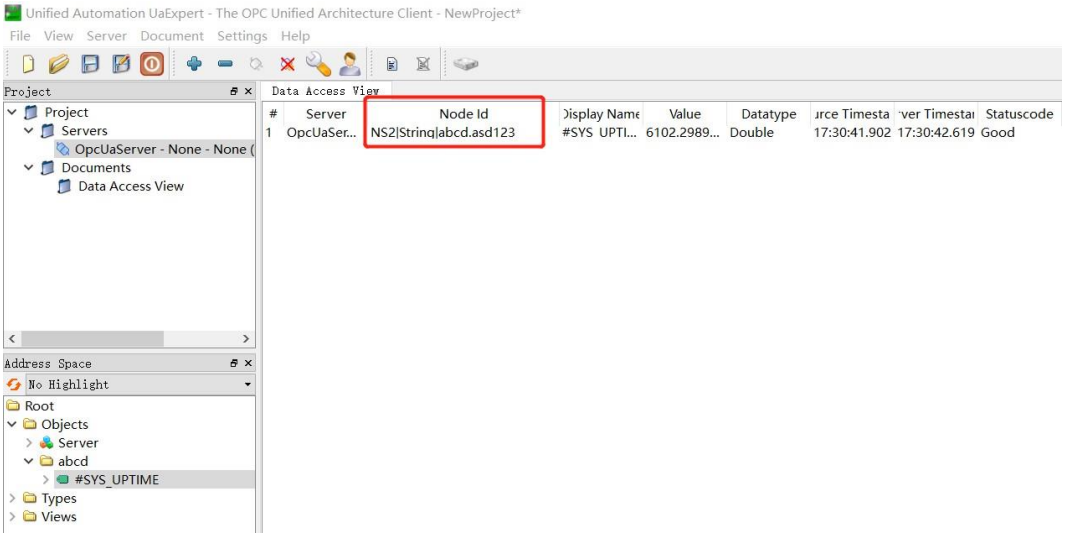
Address Type	**Namespace Index**	**Example**
Numeric	2	ns=2;i=1234
String	2	ns=2;s=Device.System_Tag.#SYS_TIME_SECOND
GUID	0	ns=0;g={8ACE8827-ECC3-4c9a-8032-CA1E9957A8E8}
Opaque	2	ns=2;b=M/RbKBsRVkePCePcx24oRA==

The data type in OPCUA we have supported:

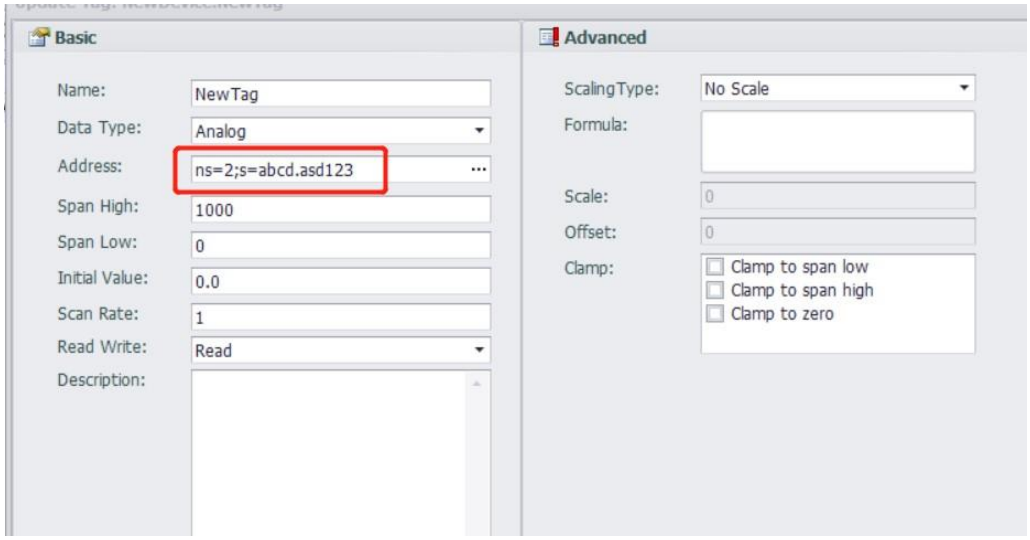
Data Type	**Description**
Boolean	Single bit
Byte	Unsigned 8 bit value
SByte	Signed 8 bit value
UInt16	Unsigned 16 bit value
Int16	Signed 16 bit value

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 edgelink should be “ns=2;s=abcd.asd123”



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