



Welcome

PLCnext Control Node-REDのEIPノード を用いたEtherNet/IP スキャナ通信検証

概要

- 本手順の概要は以下のようになります。

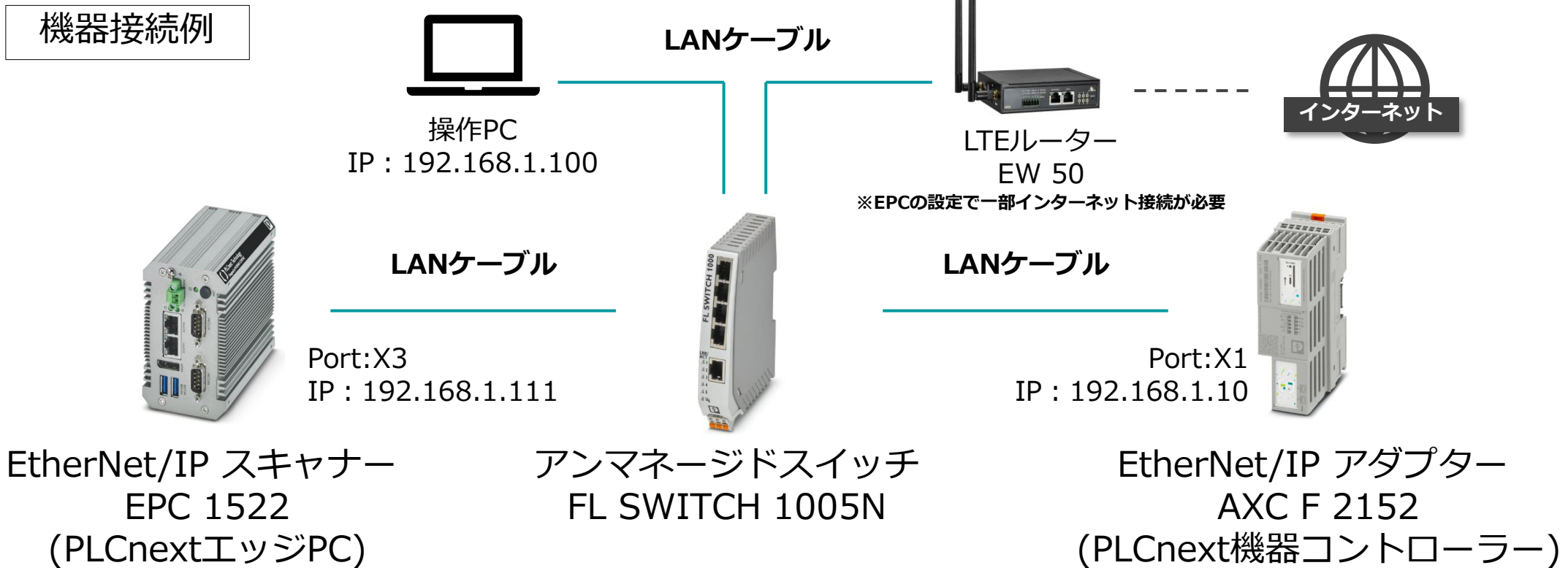


※podmanはdocker互換のコンテナエンジンです。

※Node-REDにて専用ノードをインストールする際、EPCがインターネットに接続されている必要があります。

各機器の接続

- 各機器を以下の例のように接続します。



Node-REDアプリデータのダウンロード

Node-REDアプリデータをダウンロード

- PLCnextStoreよりNode-REDのアプリデータを検索します。

PLCnextStoreへアクセスし、「node」と検索。

「Node-RED for PLCnext x86」を選択します。

[PLCnext Store](#)

The screenshot shows the PLCnext Store interface. On the left, there are filters for 'Type', 'Hardware', and 'Function'. The 'Function' filter is expanded, showing a list of categories with checkboxes: Artificial Intelligence, Cloud Connector, Communication, Database, Human Machine Interface, IO Module, Functions, and Redundancy. On the right, there is a search bar with 'node' entered, a 'Sort by' dropdown, and a 'Name: node-red' filter. Below the search bar, three app cards are displayed. The first card, 'Node-RED for PLCnext x86', is highlighted with a red box. It is a free app by Phoenix Contact GmbH & Co. KG. The second card is 'Node-RED for PLCnext ARM', also free. The third card is 'Node-RED - coffee machine demo', also free. Each card includes a description of the app's functionality.

Type ▼

Hardware ▼

Function ▲

node

Sort by: ▼

Type ▼ Developed By: ▼ Only free Apps

Name: node-red

App
Node-RED for PLCnext x86
Phoenix Contact GmbH & Co. KG
Free

The Node-RED for PLCnext app installs Node-RED directly on the PLCnext device. Now, you can use all benefits of the low-code,

App
Node-RED for PLCnext ARM
Phoenix Contact GmbH & Co. KG
Free

The Node-RED for PLCnext app installs Node-RED directly on the AXC F 2152. Now, you can use all benefits of the low-code,

App
Node-RED - coffee machine demo
Phoenix Contact GmbH & Co. KG
Free

THIS APP IS FOR TRAINING AND DEMO PURPOSES ONLY! The app runs Node-RED version 3.9.1. as a

Node-REDアプリデータのダウンロード

Node-REDアプリデータをダウンロード

- PLCnextStoreよりNode-REDのアプリデータをダウンロードします

DOWNLOAD ボタンよりダウンロードします。

※ダウンロードにはPLCnextStoreへの
ログインが必要です。

■ APP ■ NODE-RED FOR PLCNEXT X86



Node-RED for PLCnext x86

Phoenix Contact GmbH & Co. KG

✉ Contact Developer 🔗 Share App 🔔 Get Notifications

Version: 4.0.2.1

  ⓘ Help

EPCへNode-REDアプリのインストール

- WebブラウザよりEPCのWBM画面にアクセスします

webブラウザよりEPCのIPアドレスを入力します

ログイン画面が表示されるためログインします。

【ログイン情報】

username : admin

password : 機器本体に記載

Please login with your username and password.

Username

admin

Password

.....

Login

Hinweis:

Dieses Gerät darf nur von autorisierten Benutzern für autorisierte Zwecke verwendet werden. Ihre Anmeldeinformationen und alle Benutzeraktionen auf diesem Gerät können überwacht, aufgezeichnet, kopiert und auditiert werden.

Durch die weitere Verwendung dieses Geräts erklären Sie sich mit diesen Bedingungen einverstanden.

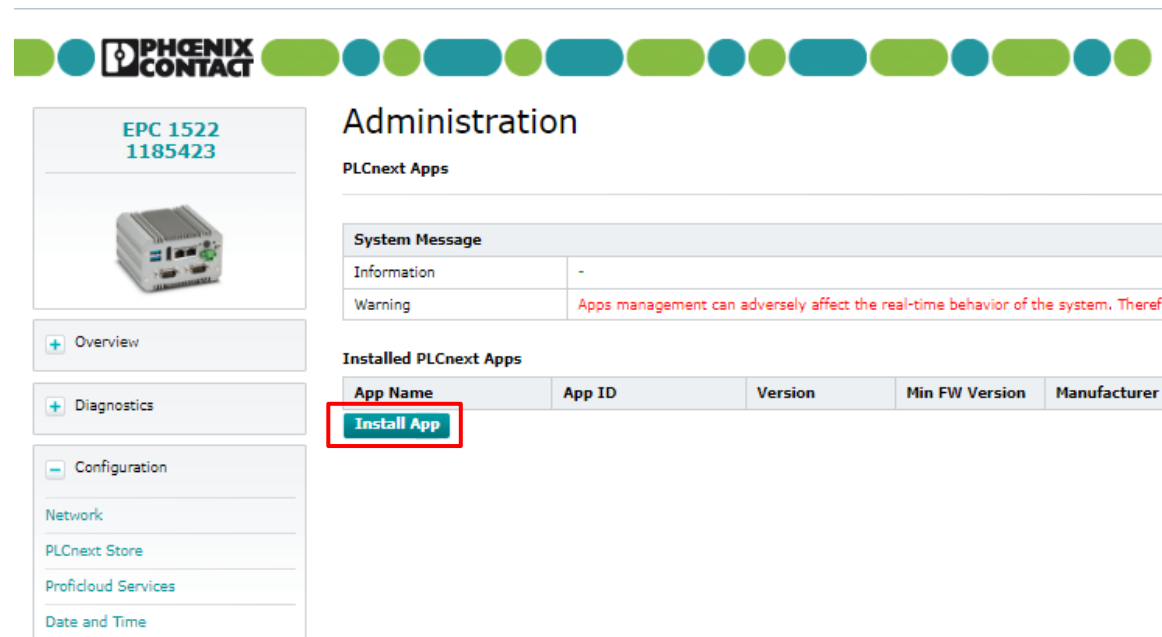
EPCへNode-REDアプリのインストール

- Node-REDアプリデータをアップロードします

Install App をクリックします

ファイル選択画面になるので
先ほどダウンロードしたNode-REDアプリデータ
を選択します。

ファイルを選択すると
インストールが開始されます。



The screenshot shows the Phoenix Contact PLCnext Control web interface. The top navigation bar includes the Phoenix Contact logo and a series of colored circles. The main content area is titled "Administration" and "PLCnext Apps". On the left, there is a sidebar with a menu containing "Overview", "Diagnostics", and "Configuration". The "Configuration" section is expanded, showing sub-items: "Network", "PLCnext Store", "Proficloud Services", and "Date and Time". The main content area displays a "System Message" section with a warning message: "Apps management can adversely affect the real-time behavior of the system. Therefore, it is recommended to perform updates during planned maintenance times." Below this, there is a table titled "Installed PLCnext Apps" with columns: "App Name", "App ID", "Version", "Min FW Version", and "Manufacturer". The "Install App" button is highlighted with a red box.

App Name	App ID	Version	Min FW Version	Manufacturer
Install App				

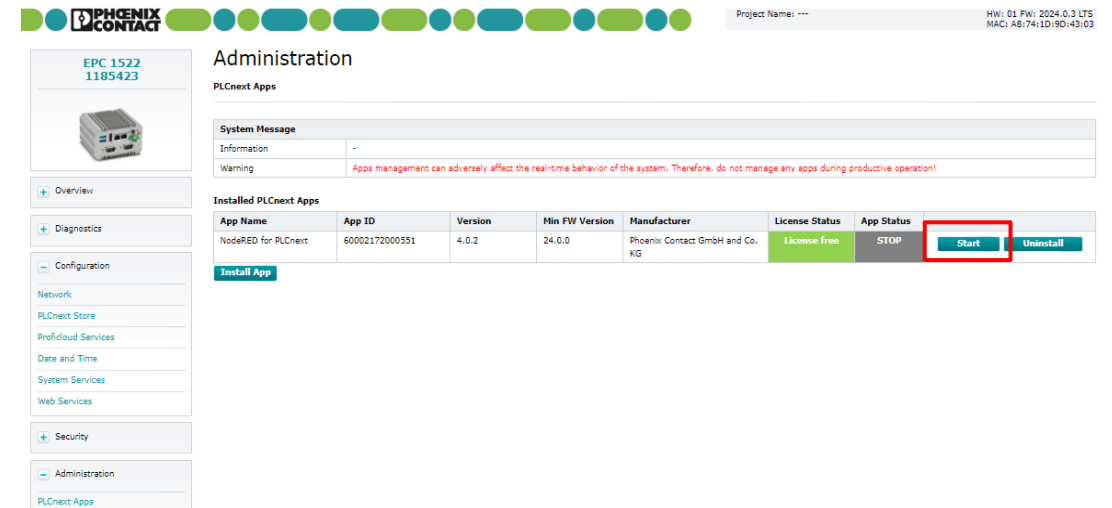
EPCへNode-REDアプリのインストール

- Node-REDのアプリをスタートします

インストールが完了したら
Startをクリックします。

アプリがスタートするまで画面が
読み込み中の表示になります。
その間は機器の電源を落とさないでください。
(アプリのスタートには数分かかります)

App Status が RUN になっていればOKです。



EPC_Node-RED_専用ノードのインストール

EPCへSSHアクセス

- EPCの操作を行うためSSHアクセスを行います。

EPCにSSHアクセスを行います。

【ログイン情報】

username : admin

password : PLC前面に記載

```
Last login: Thu Sep 19 00:31:44 2024 from 192.168.1.100  
admin@epc1522:~$
```

rootユーザー作成・切り替え

- rootユーザーにアクセスするためrootユーザーパスワードを設定します

以下のコマンドを入力します。

```
$ sudo passwd root
```

最初にログインパスワードを求められるので
ログインパスワードを入力します。

その後にrootユーザーのパスワードを求められる
ので任意パスワードを入力します。
(確認で2回入力します)

以下のコマンドを入力しrootユーザーに切り替えます

```
$ su
```

(パスワードは設定したパスワードです)

```
admin@axcf2152:~$ sudo passwd root

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

For security reasons, the password you type will not be visible.

Password:
New password:
Retype new password:
passwd: password updated successfully
admin@axcf2152:~$ su
Password:
root@axcf2152:/opt/plcnext/#
```

Node-REDのEtherNet/IP専用ノードインストール

- Node-REDを実行しているコンテナ内に入ります。

Node-REDのコンテナIDを調べるために
Podmanの実行プロセスを表示します。

```
# podman ps
```

※rootユーザーにて実行してください

コンテナIDが表示されるのでそちらを使って
コンテナ内に入ります。

```
# podman exec -it [コンテナID] /bin/bash
```

```
root@epc1522:/opt/plcnext/# podman ps
CONTAINER ID  IMAGE                COMMAND              CREATED
70a2e519fdb  localhost/nodered:4.0.2  12 minutes ago
root@epc1522:/opt/plcnext/# podman exec -it 70a2e519fdb /bin/bash
70a2e519fdb:~$
```

Node-REDのEtherNet/IP専用ノードインストール

- Node-REDのコンテナ内で専用ノードのインストールを実施します。

コンテナ内で対象のディレクトリに移動します。

```
$ cd /data
```

専用ノードインストールコマンドを実行します。

```
$ npm install @serafintech/node-red-contrib-eip-io
```

```
70a2e519fdbe:~$ cd /data
70a2e519fdbe:/data$ npm install @serafintech/node-red-contrib-eip-io
added 29 packages, and audited 30 packages in 15s

17 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

専用ノードインストール後、EPCを再起動させます。

※専用ノードのインストールを行うには

EPCがインターネットにつながっている必要があります。

※こちらの専用ノードはNode-REDバージョン2.0.0以上でないと使えません。

EPC_Node-RED_専用ノードのインストール

Node-REDのEtherNet/IP専用ノードインストール

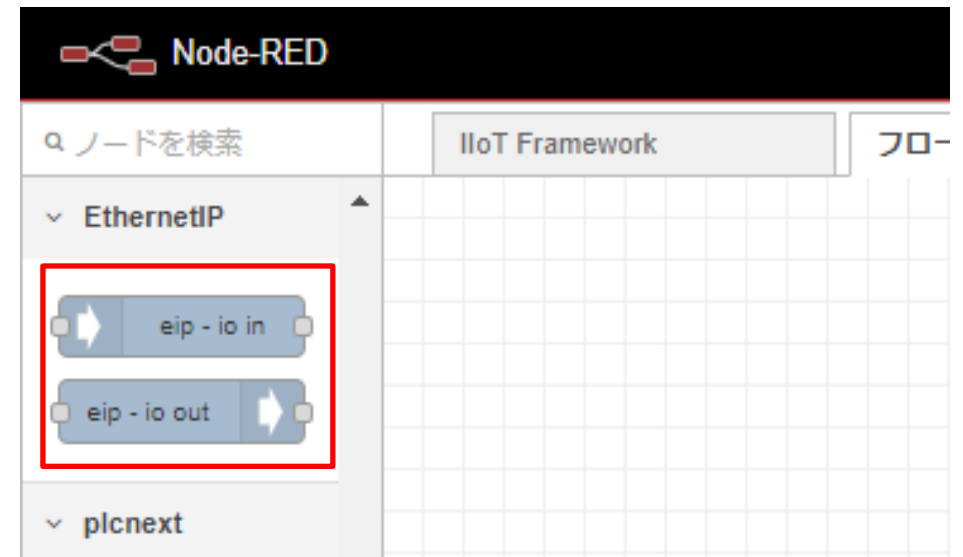
- 専用ノードが正常にインストールされているか確認します。

WEBブラウザにてEPCのNode-REDの
web画面にアクセスします

IPアドレス:51880

(例) 192.168.1.111:51880

画面よりEtherNet/IPの専用ノードがあるか確認します。



Node-REDコンテナのネットワーク設定変更

- Node-redコンテナのymlファイルのバックアップファイルを作成します。

EPCにsshアクセスしrootユーザーに切り替え
対象ディレクトリに移動します。

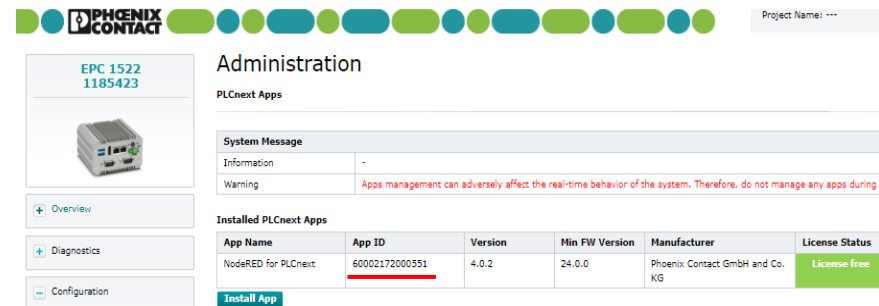
```
# cd /opt/plcnext/appshome/data/[Node-REDのID]/
```

ymlファイルのバックアップファイルを作成します。

```
# cp docker-compose.yml docker-compose.yml.bak
```

```
root@epc1522:/opt/plcnext/# cd /opt/plcnext/appshome/data/60002172000551/
root@epc1522:/opt/plcnext/appshome/data/60002172000551# ls -la
total 44
drwxrwxrwx 5 plcnext_firmware plcnext 4096 Sep 19 00:54 .
drwxrwxr-x 3 plcnext_firmware plcnext 4096 Sep 19 00:44 ..
-rwxrwxrwx 1 root root 97 Sep 19 00:52 .env
-rwxrwxrwx 1 root root 0 Sep 19 00:52 NodeRED
-rwxrwxrwx 1 root root 3693 Sep 19 01:56 'NodeRED for PLCnext.log'
-rwxrwxrwx 1 root root 653 Sep 19 00:52 app_info.json
drwxrwxrwx 2 root root 4096 Sep 19 00:52 bin
-rwxrwxrwx 1 root root 199 Sep 19 00:52 docker-compose.yml
-rwxrwxrwx 1 root root 0 Sep 19 00:54 dockerapp_install
drwxrwxrwx 2 root root 4096 Sep 19 00:52 images
-rwxrwxrwx 1 root root 5807 Sep 19 00:52 initscript.sh
drwxrwxrwx 3 root root 4096 Sep 19 00:52 volumes
root@epc1522:/opt/plcnext/appshome/data/60002172000551# cp docker-compose.yml docker-compose.yml.bak
root@epc1522:/opt/plcnext/appshome/data/60002172000551#
```

※Node-REDのIDは
EPCのWBMアプリ画面より確認できます。



The screenshot shows the 'Administration' page of the PLCnext Apps interface. On the left, there is a sidebar with a navigation menu containing 'Overview', 'Diagnostics', and 'Configuration'. The main content area displays 'System Message' with a warning about app management affecting real-time behavior. Below this, a table titled 'Installed PLCnext Apps' lists the installed applications. The table has columns for App Name, App ID, Version, Min FW Version, Manufacturer, and License Status. One application, 'NodeRED for PLCnext', is listed with App ID 60002172000551, Version 4.0.2, and a 'License free' status. An 'Install App' button is visible at the bottom of the table.

App Name	App ID	Version	Min FW Version	Manufacturer	License Status
NodeRED for PLCnext	60002172000551	4.0.2	24.0.0	Phoenix Contact GmbH and Co. KG	License free

Node-REDコンテナのネットワーク設定変更

- ymlファイルの内容を変更します。

nanoエディタにてymlファイルを開きます。

nano docker-compose.yml

以下の設定を ports に追加します

- 2222:2222/udp

nanoエディターより上書き・終了操作を行います。

上書き : [Ctrlキー] + O

終了 : [Ctrl] + X

nanoエディター終了後、EPCを再起動させてください。

```
version: "3.7"
services:
  node-red:
    image: ${IMAGE_NAME}:${IMAGE_TAG}
    ports:
      - 51880:1880
      - 2222:2222/udp
    user: ${USER_ID}
    volumes:
      - ./volumes/node-red:/data
    restart: unless-stopped
```

Node-REDのプログラム作成

- EPCのNode-redのWEB画面にアクセスします。

WEBブラウザにてEPCの
Node-REDのweb画面にアクセスします

IPアドレス:1880
(例)192.168.1.111:1880

※yamlファイル内容変更によりポート番号が1880となっています。

Node-RED画面より新しいフローを追加し
「IIoT Framework」フローを削除します。

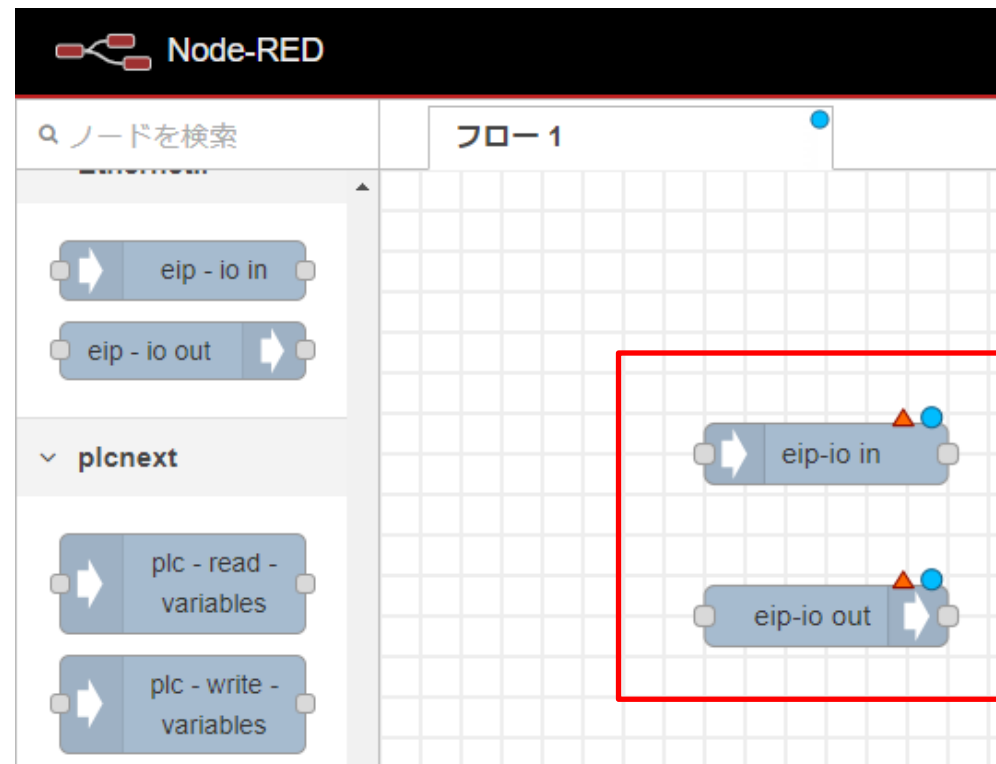


Node-REDのプログラム作成

- EtherNet/IPの専用ノードを設置します。

以下2つのノードを設置します。

- eip-io in
- eip-io out



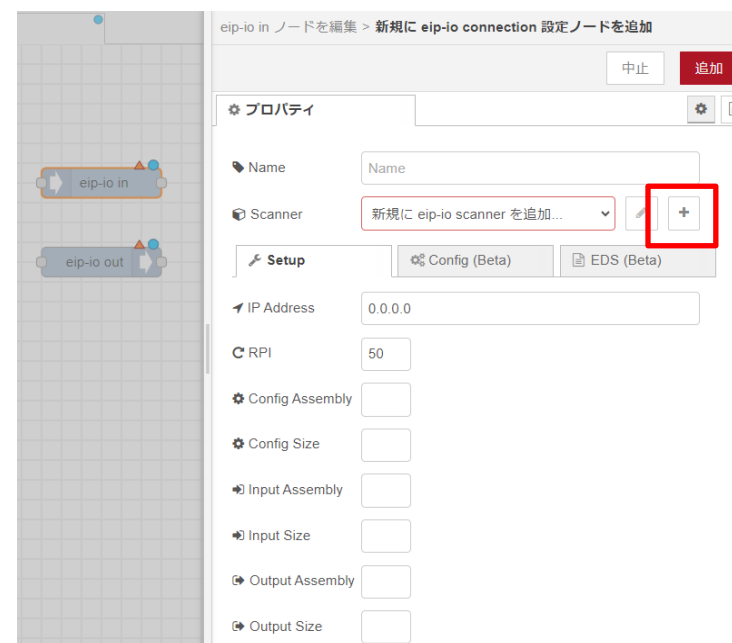
EPC_Node-RED_動作テストプログラム作成

Node-REDのプログラム作成

- EtherNet/IPの専用ノードの設定を行います。



ノードをダブルクリックし編集画面を開きます。
Connectionの+をクリックします。



Scannerの+をクリックします。

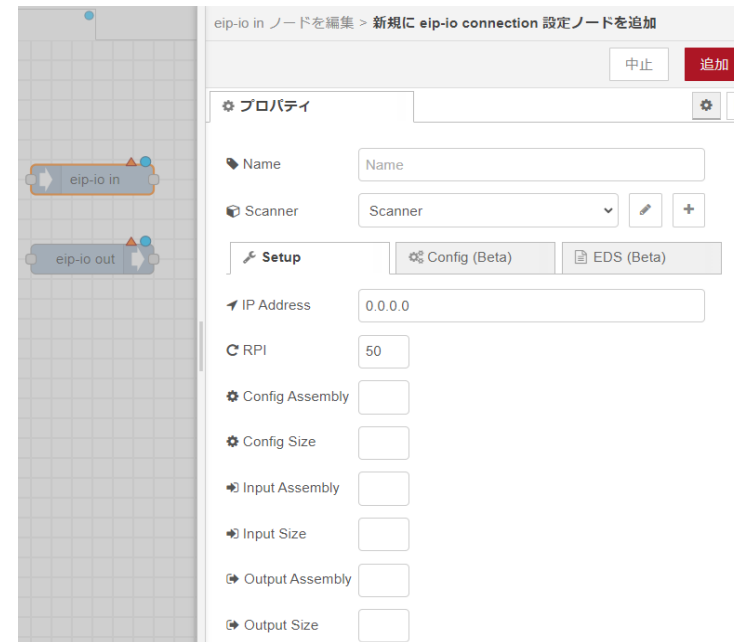
EPC_Node-RED_動作テストプログラム作成

Node-REDのプログラム作成

- EtherNet/IPの専用ノードの設定を行います。



設定はそのまま追加をクリックします。
(クリック後アダプターのプロパティ画面に遷移します)



アダプターのプロパティ画面は
アダプター機器の設定を行います。

Node-REDのプログラム作成

- アダプターのEDSファイルより設定の確認を行う。

対象アダプターのEDSファイルより
各種設定の確認をおこないます。

アダプターとなるAXC F 2152のEDSファイルを
製品ページよりダウンロードします。

※[ダウンロード] - [機器の説明]にedsファイルがあります。

EDSファイルをテキストエディタで開き
RPIやinput Assembly, output Assemblyの
確認を行います。

```
[Params]
Param1 =
    0,                $ reserved, shall equal 0
    ,,               $ Link Path Size, Link Path
    0x0000,          $ Descriptor
    0xC8,            $ Data Type
    4,               $ Data Size in bytes
    "RPI",           $ name
    "",              $ units
    "",              $ help string
    10000,1000000,50000, $ min, max, default data values
```

```
Assem110 =
    "Input Data",     $ Assembly name
    "",              $ "20 04 24 65 30 03", $ Path
    ,                $ size
    0x0001,           $ descriptor, 7=variable size, adapter included
    ,,               $ reserved
```

```
Assem100 =
    "Output Data",   $ Assembly name
    "",              $ "20 04 24 64 30 03", $ Path
    ,                $ size
    0x0001,           $ descriptor, 7=variable size, adapter included
    ,,               $ reserved
```

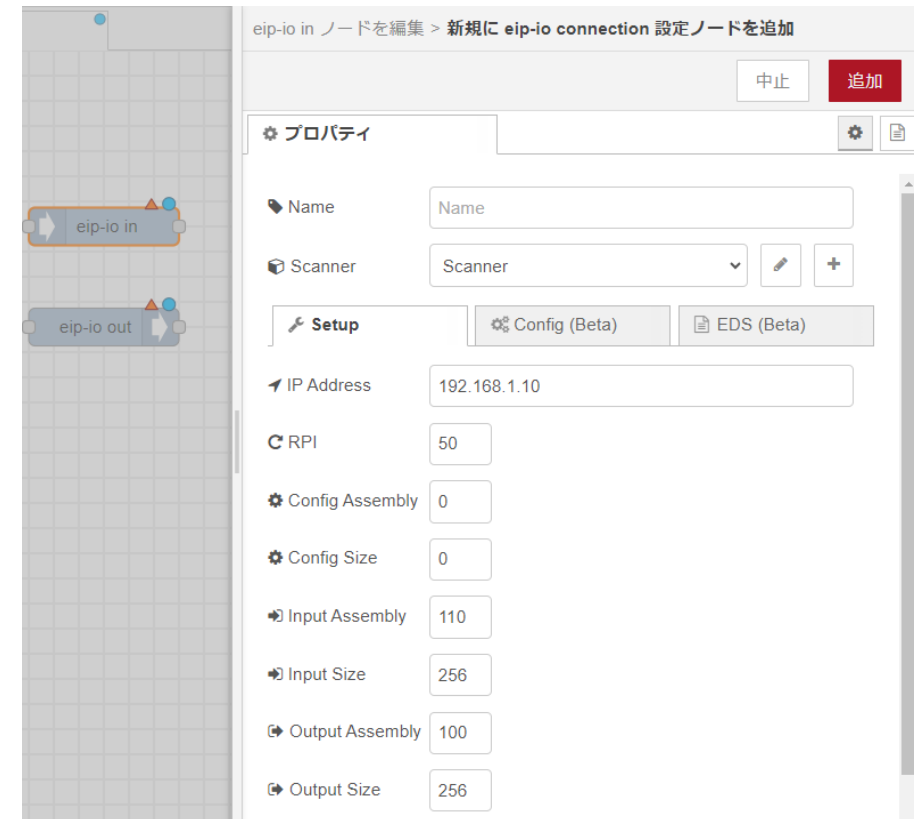
AXC F 2152 EDSファイル

Node-REDのプログラム作成

- EtherNet/IPの専用ノードの設定を行います。

EDSファイルにて確認した各設定内容を記入します。

記入後に追加をクリックします。
(クリック後、ノードの設定画面に遷移します)



Node-REDのプログラム作成

- EtherNet/IPの専用ノードの設定を行います。

ノードの設定画面ではノード本体の設定を行います。

設定後に完了をクリックします。

これらの設定を

- ・ eip-io in
- ・ eip-io out

の両方に適応します。

(一度アダプターなどの設定をすると、プルダウンにて選択可能です)

eip-io in ノードを編集

削除 中止 完了

プロパティ

Name

Connection 192.168.1.10

Byte Offset 0

Bit Offset 0

Bit Size 8

Data Type UInteger8 (USINT)

Big Endian

Decimal Points 0

Update Rate 50

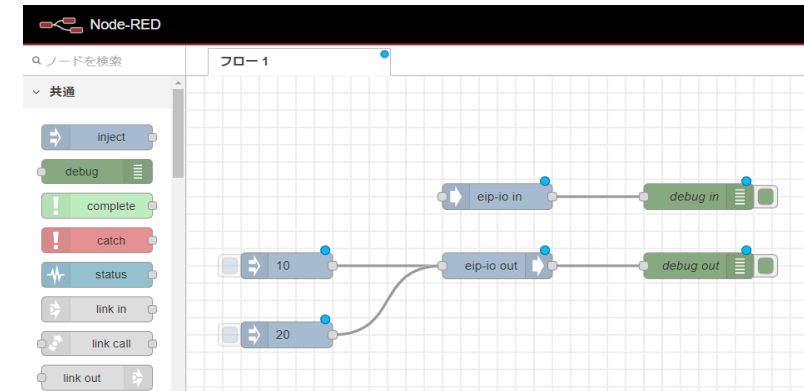
Node-REDのプログラム作成

- 通信確認用のノードを追加します。

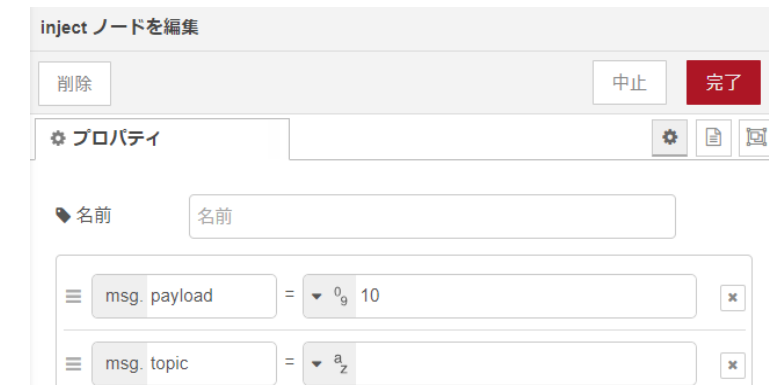
eip-io in と eip-io out のノードの出力を表示するため debugノードをそれぞれのノードの出力と繋がめます。

eip-io out のノードにはテスト用の出力データとして injectノードを eip-io out ノードの入力に2つ繋がめます。

injectノードでは任意の値を手動で eip-ip out ノードに入力するように設定します。
(例として10,20にて設定しています)



ノード配置



injectノード設定

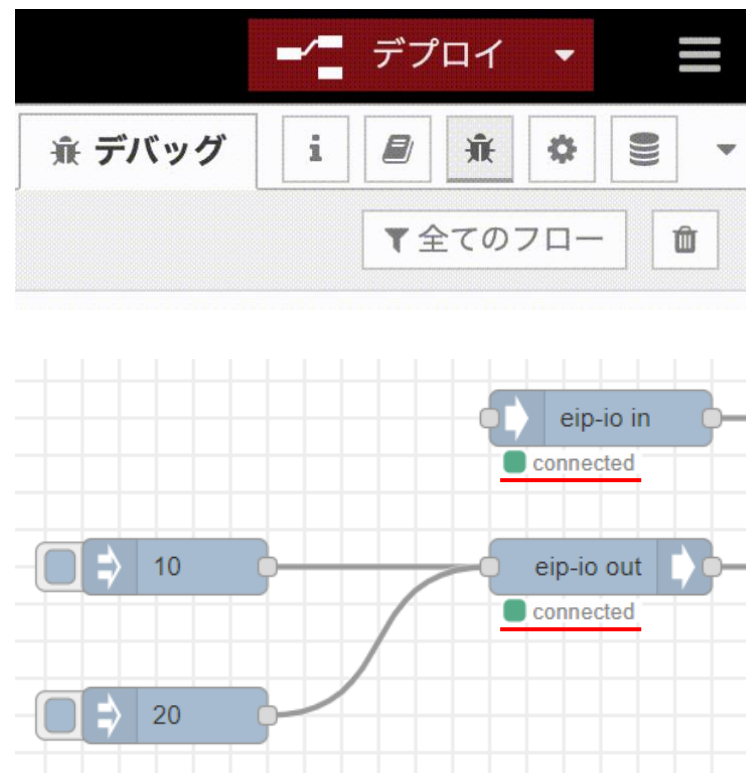
Node-REDのプログラム作成

- プログラムをデプロイしてアダプターとの接続を確認する。

画面右上のデプロイを実行して
アダプターと接続できるか確認します。

- ・ eip-io in
- ・ eip-io out

ノードの下に connected と表示されれば
アダプターと接続ができています。

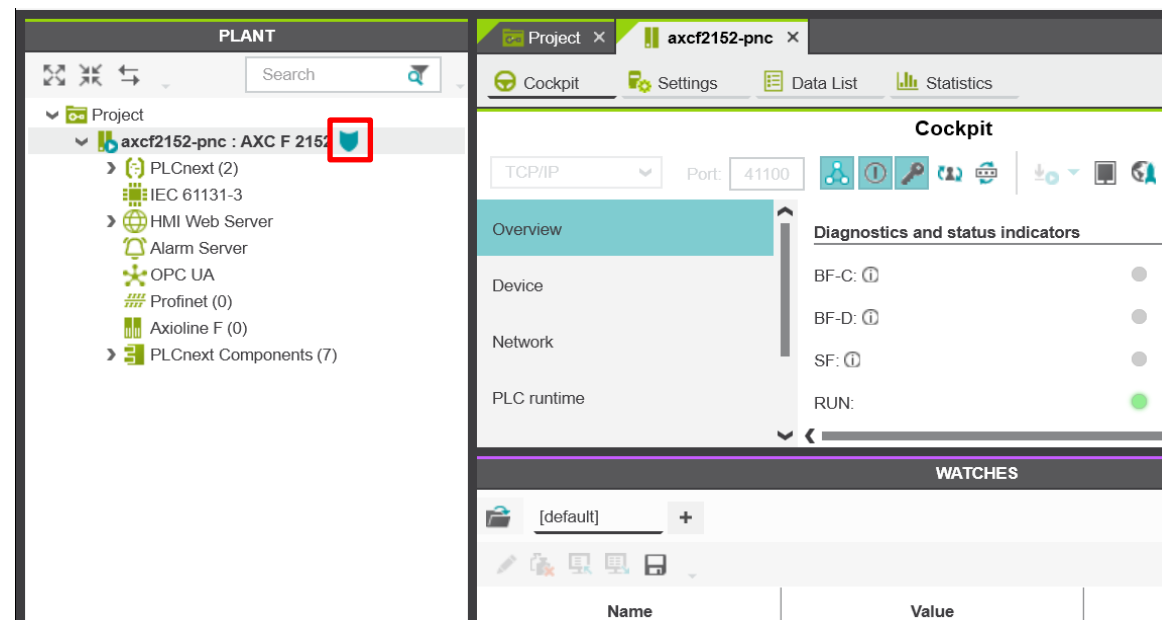


アダプター側(AXC F 2152)の準備

- PLCnext Engineerを立ち上げてアダプター(AXC F 2152)と接続します。

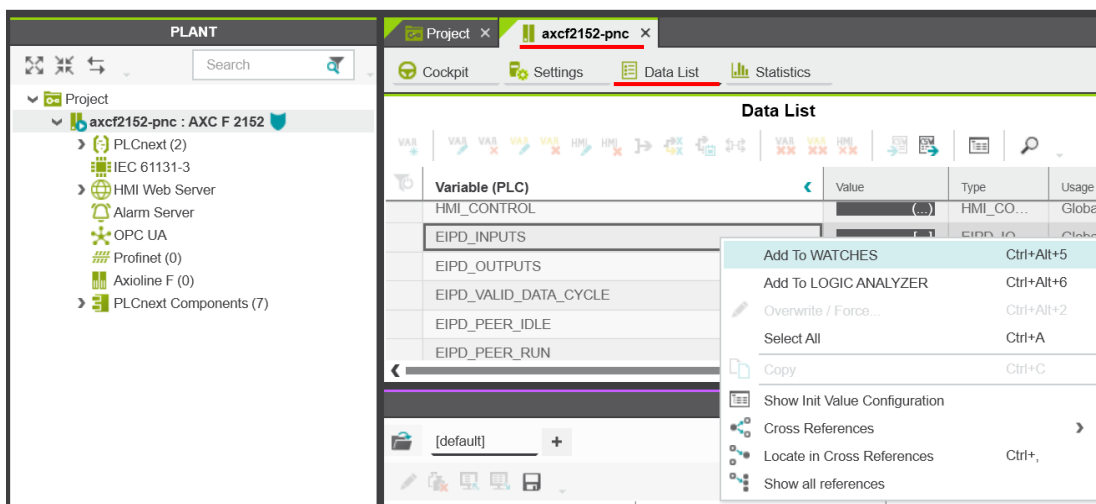
PLCnext Engineerソフトを立ち上げて
AXC F 2152とconnectしている状態にしてください。
(プログラムを作成する必要ありません)

※本手順書ではPLCnextEngineerの
基本的な操作については割愛いたします。

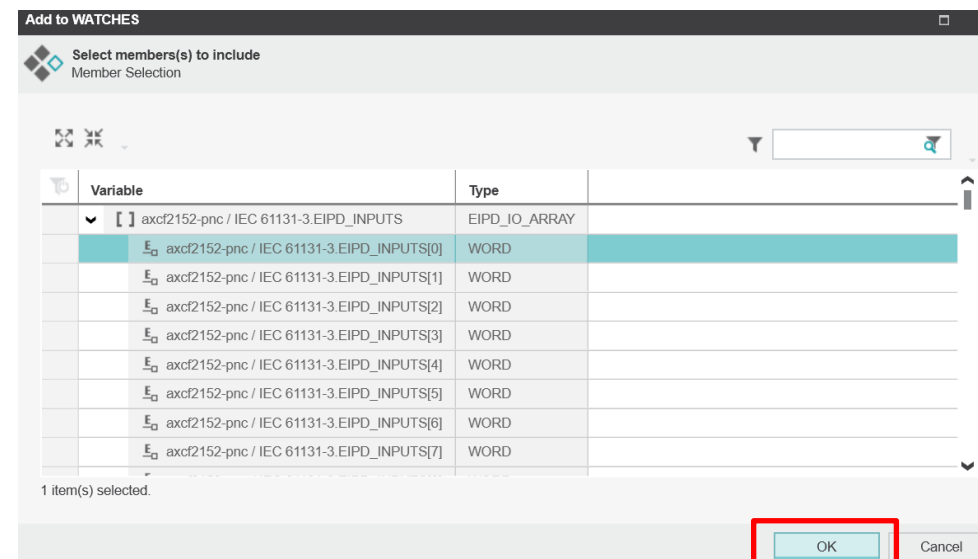


アダプター側(AXC F 2152)の準備

- EtherNet/IPデータのモニタリング設定を行います。



[Axcf2152-pnc]-[Data List]より
EIPD_INPUTS と EIPD_OUTPUTS を右クリックし
Add To WATCHES をクリックします。



変数が表示されるので
~EIPD_INPUTS(OUTPUTS)[0]を選択して
OKをクリックします。

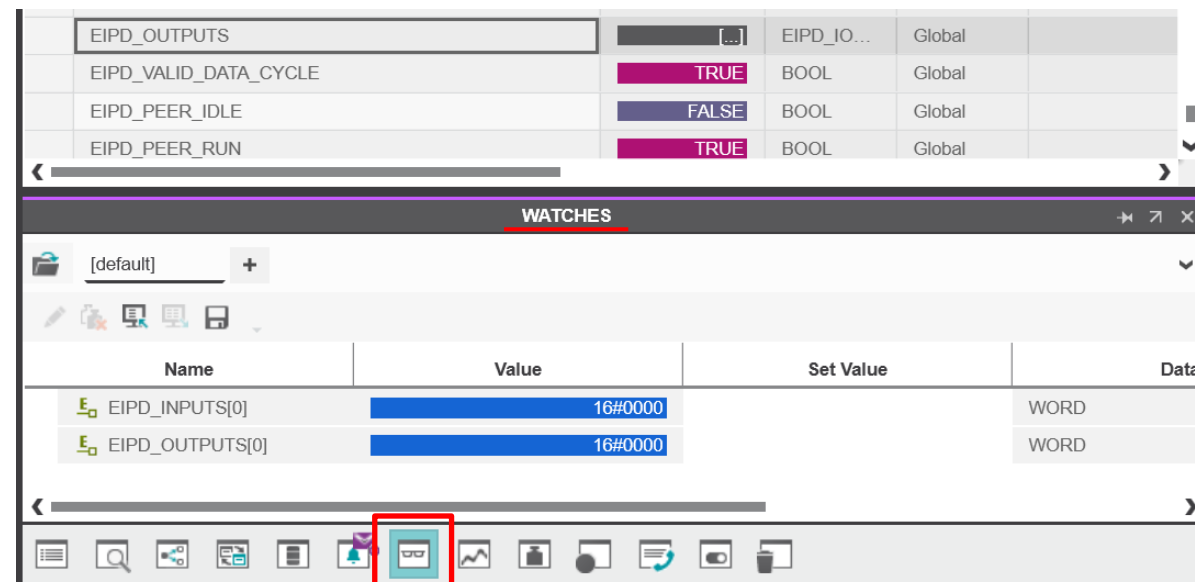
アダプター側(AXC F 2152)の準備

- PLCnext Engineerを立ち上げてアダプター(AXC F 2152)と接続します。

画面下部の  アイコンをクリックして
WATCHES画面を表示します。

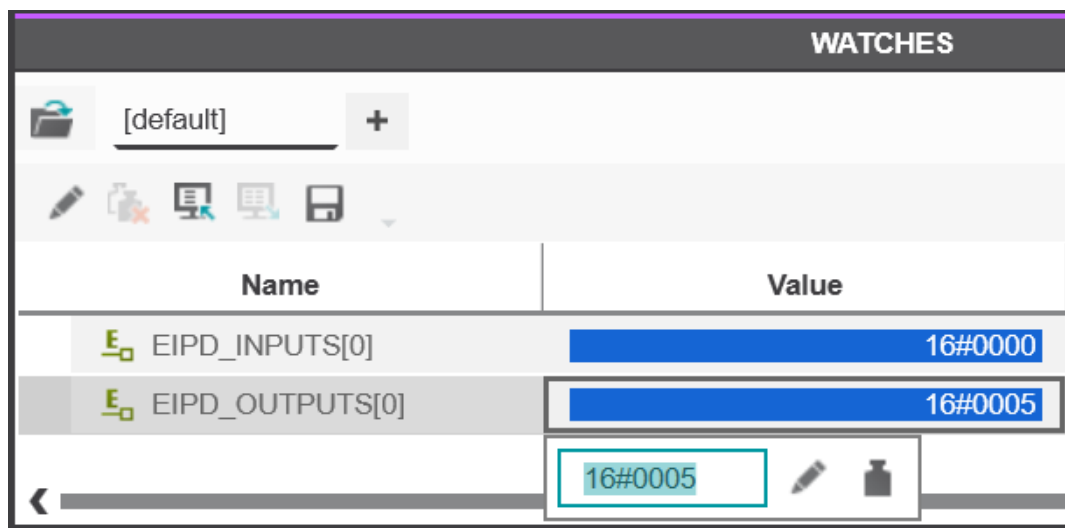
こちらに先ほどの設定した
EIPD_INPUTS[0]
EIPD_OUTPUTS[0]
が表示されます。

これでアダプター側(AXC F 2152)
の準備は完了です。



動作テスト_アダプターからスキャナヘデータ出力

- アダプターからスキャナーヘデータを出力します。



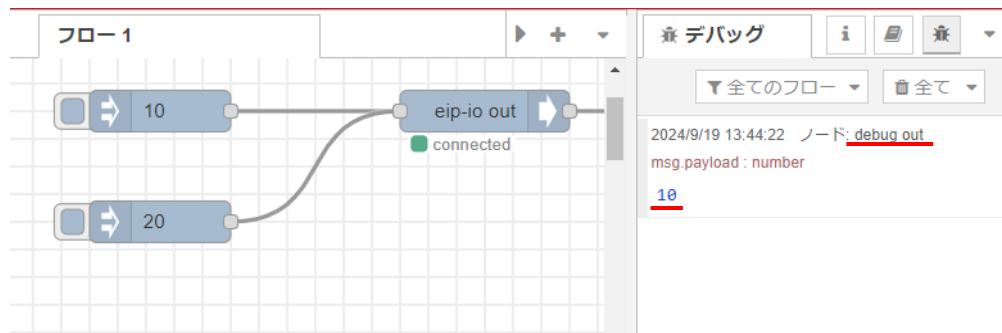
PLCnextEngineerにて
EIPD_OUTPUTS[0]のデータを
16#0005 に設定します。



Node-RED画面にて画面右上の  をクリックし
デバッグ画面を表示します。
デバッグ画面にアダプターからのデータが表示されます。

動作テスト_スキャナからアダプターへデータ出力

- スキャナからアダプターへデータを出力します。



Node-RED画面にて任意のinjectノードのボタンをクリックします。
デバッグ画面に出力された数値が表示されます。
(例として10のinjectノードのボタンをクリック)

The screenshot shows the 'WATCHES' window in the PLCnextEngine. It displays two variables: EIPD_INPUTS[0] with a value of 16#000A and EIPD_OUTPUTS[0] with a value of 16#0005. The values are shown in hexadecimal format.

Name	Value
EIPD_INPUTS[0]	16#000A
EIPD_OUTPUTS[0]	16#0005

PLCnextEngineにて
EIPD_INPUTS[0]にスキャナからの
データが入力されます。
(16進数表記)

Thank you