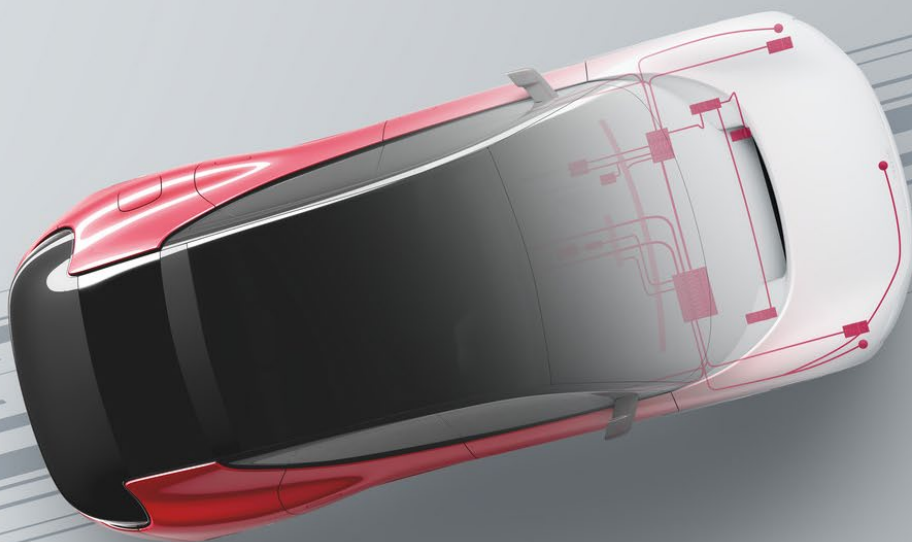


はじめての CANoe.Ethernet

ビギナー向け資料「はじめてシリーズ」



目次

はじめに.....	4
1. CANoe.Ethernet とは？	6
2. CANoe.Ethernet でできること	9
2.1 Ethernet フレームやシグナル、各種プロトコルの解析.....	10
2.2 Ethernet ノードのシミュレーション.....	11
2.3 Ethernet ノードのテスト	12
3. CANoe.Ethernet の機能	13
3.1 解析関連.....	14
3.1.1 トレース Window (CANoe/CANalyzer 共通機能).....	14
3.1.2 プロトコルモニター (CANoe/CANalyzer 共通機能).....	15
3.1.3 統計 Window (CANoe/CANalyzer 共通機能).....	15
3.1.4 フィルターブロック (CANoe/CANalyzer 共通機能)	16
3.2 シミュレーション関連	17
3.2.1 Ethernet Packet Builder (CANoe/CANalyzer 共通機能)	17
3.2.2 TCP/IP スタック (CANoe 専用機能)	18
3.2.3 CAPL (CANoe/CANalyzer 共通機能)	19
3.2.4 インタラクショナルレイヤー (CANoe 専用機能)	20
3.2.5 モデル生成ウィザード (CANoe 専用機能)	22
3.3 テスト関連 (CANoe 専用機能)	23
4. Ethernet 通信用の ネットワークインターフェイス	24
4.1 TAP.....	25
4.2 レイヤー2 スイッチ	26
4.3 メディアコンバーター	26
4.4 ハードウェアバースト送信	27
4.5 エラーフレーム送信	27
4.6 ハードウェアフィルター	27
5. ソフトウェアとハードウェアを使用するための事前準備.....	28
5.1 ソフトウェアのインストール.....	28
5.2 ハードウェアの準備	29
5.2.1 VN5000 インターフェイスの起動と接続.....	29
5.2.2 Ethernet ケーブル	34
5.3 CANoe の動作確認	36

5.3.1 サンプルコンフィギュレーションの起動	36
5.3.2 サンプルコンフィギュレーションの実行	37
6. 実際に使ってみよう.....	38
6.1 必要なもの	38
6.2 作成する環境	38
6.3 インターフェイスの設定	39
6.4 CANoe のコンフィギュレーションの作成	46
6.4.1 新規コンフィギュレーションの作成	46
6.4.2 Ethernet チャンネルの設定	47
6.4.3 ネットワークの作成	48
6.4.4 仮想ノードの作成	52
6.4.5 ポート設定	66
6.4.6 CANoe の開始	67
6.4.7 仮想ノードからパケット送信	67
あとがき	68
7. お問い合わせ窓口	69

発行元:ベクター・ジャパン株式会社

ベクター・ジャパン株式会社の書面による許可なしに、本書の内容を転載、複製、複写することを禁じます。

記述されている内容や製品画面の表示名称は予告なく変更されることがあります。

はじめに

本稿は、ベクターが提供する ECU および車載ネットワークの開発支援ツール「CANoe.Ethernet (読み方: キャヌー・オプション・イーサネット)」について、ECU および車載 Ethernet ネットワークの開発を検討されている方や、CANoe.Ethernet がどのようなツールかを詳しくご存じない方など、CANoe.Ethernet のご使用経験のない方を対象に作成した資料です。

「CANoe.Ethernet を使うと、どのようなことができるのか?」、「どのように使用するのか?」などについて、やさしく解説していきます。

CANoe.Ethernet が持つ機能の概要全般を説明しますので、ご使用経験のない方や一部の機能を使用しているユーザー様においては、使ったことがない機能などを知る機会となります。是非お読みいただければ幸いです。

また、Ethernet 用のネットワークインターフェイスを使用して実際の ECU と通信する方法もご紹介しますので、CANoe.Ethernet のスタートアップ資料としてご使用いただけます。

上述のとおり、本稿は CANoe.Ethernet ができることの概要について説明する資料となります。そのため、詳細な設定方法、操作方法などは解説していません。

CANoe.Ethernet や Ethernet 用のネットワークインターフェイスを実際にお使いになる場合には、下記を参考にしてください。

【設定方法や操作方法のドキュメント】

> CANoe.Ethernet に関するアドバンスマニュアル

アドバンスマニュアルは、ベクター・ジャパン Web サイト「製品機能活用マニュアル」に掲載しています。詳細は下記リンク先をご覧ください。

<https://www.vector.com/jp/ja/support-downloads/vj-products-service/>

【はじめてシリーズ】

また、CANoe 本体や Ethernet プロトコルについて詳しくご存じない方は、以下を参考にしてください。その他のツールや車載システムの基礎知識に関する資料もございます。ご興味があればご参照ください。詳細は下記リンク先をご覧ください。

<https://www.vector.com/jp/ja/know-how/beginners/>

> はじめての CANoe

> はじめての車載 Ethernet

**【トレーニング・セミナー】**

それ以外にも、ベクター・ジャパンでは、ベクターのソフトウェアツールに関するトレーニングやセミナーを開催していますので、是非ご活用ください。詳細は下記リンク先をご覧ください。

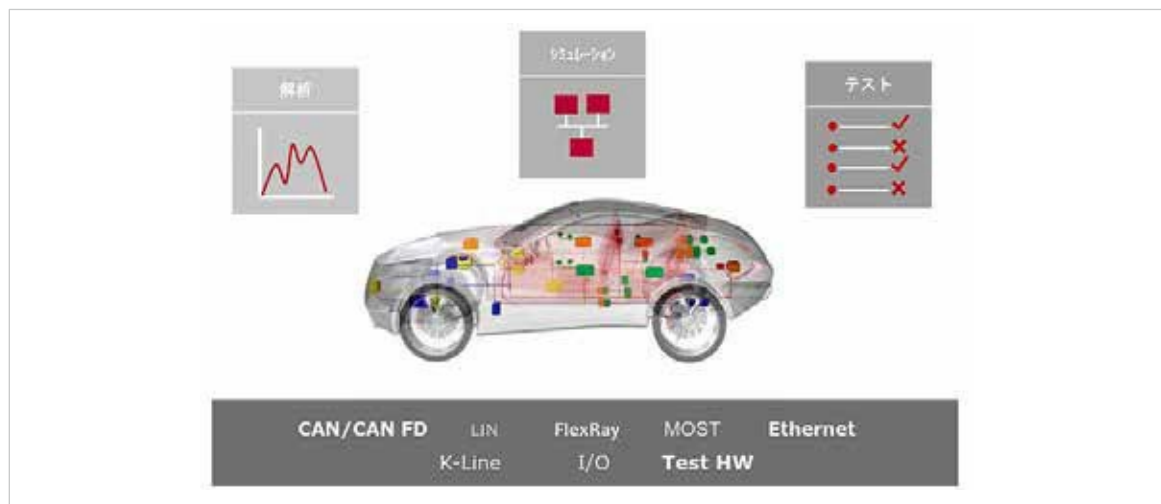
<https://academy.vector.com/jp/ja/>

それでは、CANoe.Ethernet の製品概要を見ていきましょう。

1. CANoe.Ethernet とは？

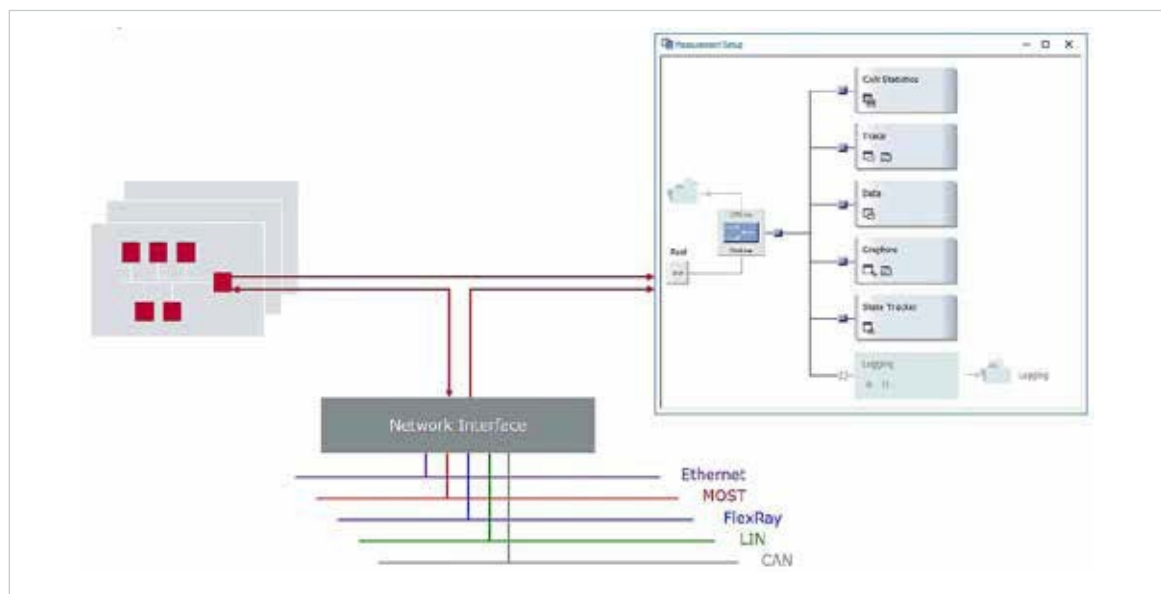
CANoe.Ethernet の概要を説明する前に、CANoe について簡単におさらいしましょう。

CANoe は、ECU および車載ネットワークの開発・テスト・解析用の総合開発ツールです。CANoe は CAN/LIN/Ethernet などの車載ネットワークに対し、「解析」、「シミュレーション」、「テスト」の3つの機能を持っています。



CANoe の解析機能では、各種解析用の Window を使用して送受信データを表示できます。

また、ネットワークインターフェイスを使用して、実機の ECU と仮想 ECU 間の通信や実機の ECU 間の通信状況を解析することが可能です。



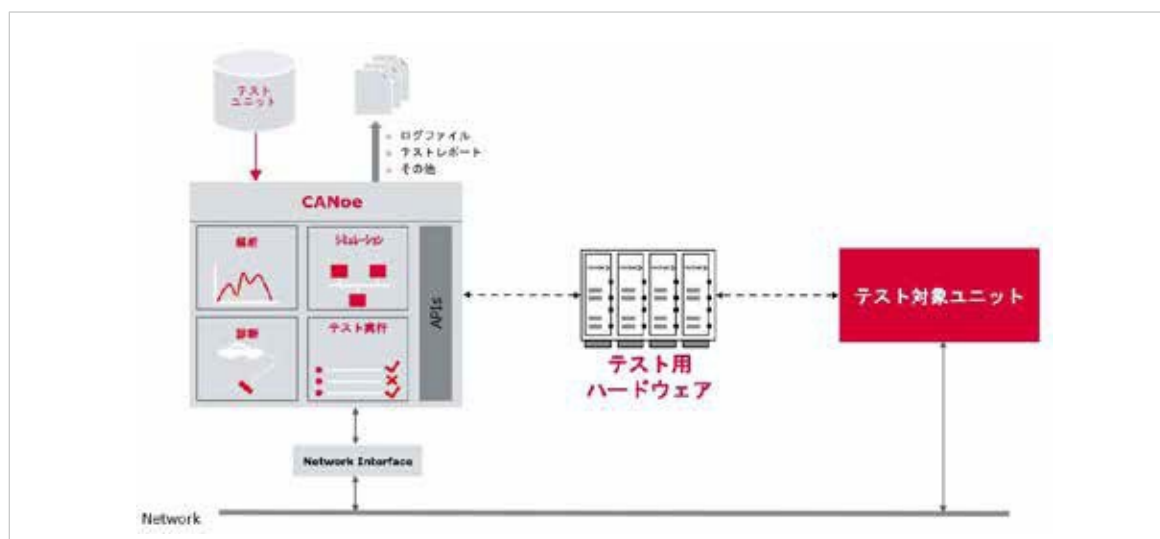
CANoe のシミュレーション機能では、複数の仮想 ECU を PC 上に設定してシミュレーションが実行できるため、実機がない状態でもネットワーク全体もシミュレーションが実行できます。

そのため、実機の完成を待たずにネットワーク設計や評価を行うことが可能です。

実機が完成した後は、実機と仮想 ECU のシミュレーションを行うことで実機の検証を行うことができます。



CANoe のテスト機能では、テストシナリオをインポートして、そのテストシナリオを自動実行することができます。仮想 ECU や実機の ECU に対してテストを実行でき、テスト結果のレポートを自動生成する機能を備えています。なお、テストシナリオは専用のテスト作成ツールである vTESTstudio を使用して作成します。vTESTstudio の基本的な機能については、「はじめての vTESTstudio」をご覧ください。



CANoe.Ethernet は、CANoe の基本機能を利用して車載 Ethernet に対応するよう拡張されたオプション製品です。

CANoe.Ethernet 専用の機能をいくつか備えており、車載 Ethernet の解析・シミュレーション・テストを効率よく実施することが可能です。



2. CANoe.Ethernet でできること

本章では CANoe.Ethernet でできることについて簡単に紹介します。

CANoe.Ethernet は CANoe の基本機能を使って車載 Ethernet の解析・シミュレーション・テストを実施するためのツールです。

CANoe の基本的な機能や CANoe.Ethernet 専用の機能を利用して次の内容を実現します。

- > Ethernet フレームやシグナル、各種プロトコルの解析
- > Ethernet ノードのシミュレーション
- > Ethernet ノードのテスト

CANoe.Ethernet でできることは上記以外にも数多くありますが、本稿では代表的なものを抜粋して掲載しています。

なお、CANalyzer.Ethernet は CANoe.Ethernet の解析機能と一部のシミュレーション機能のみに限定したツールです。

下表のように、CANalyzer では単体の仮想ノードのシミュレーションのみ対応しています。また、シミュレーション機能によっては CANoe でしか対応していない機能がありますので、後述の CANoe.Ethernet の機能紹介の中で CANalyzer の対応有無についても表記しています。

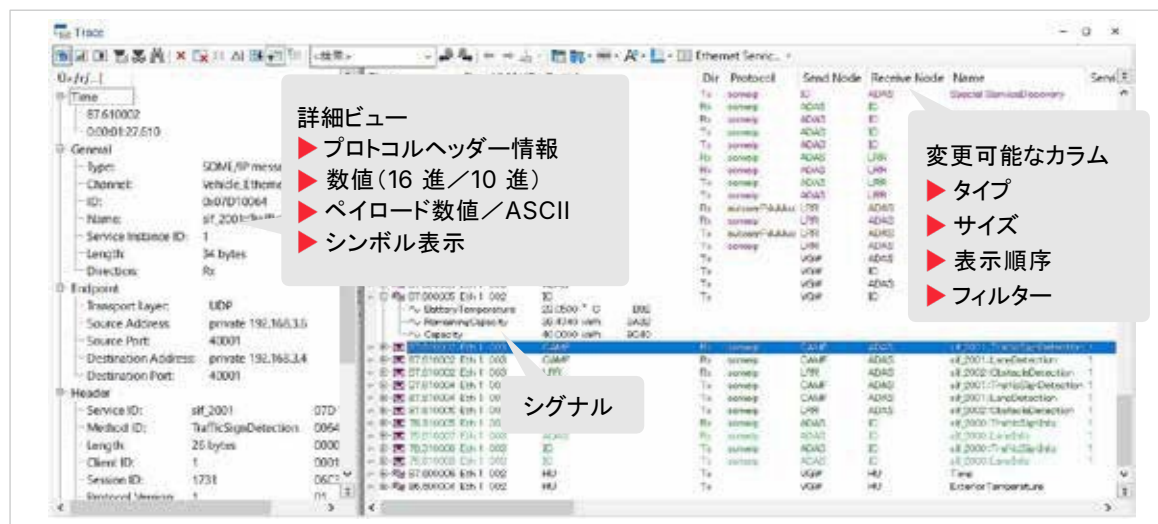
	解 析	シミュレーション		テ ス ト
		単体の仮想ノード	複数の仮想ノード	
CANalyzer	✓	✓	—	—
CANoe	✓	✓	✓	✓

それでは CANoe.Ethernet でできることについて順番に見ていきましょう。

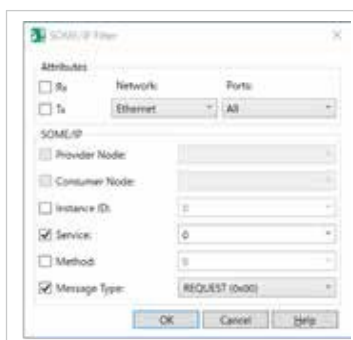
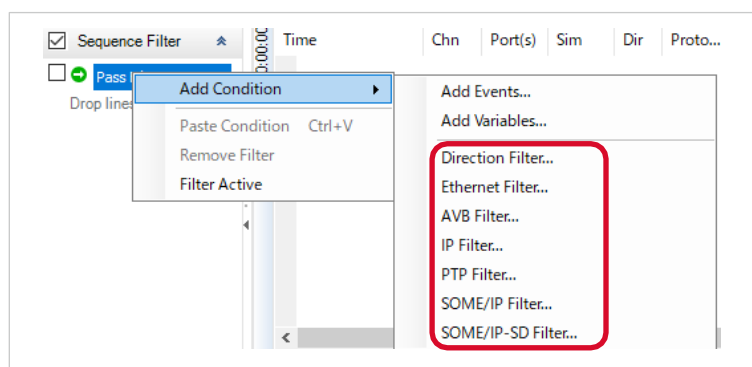
2.1 Ethernet フレームやシグナル、各種プロトコルの解析

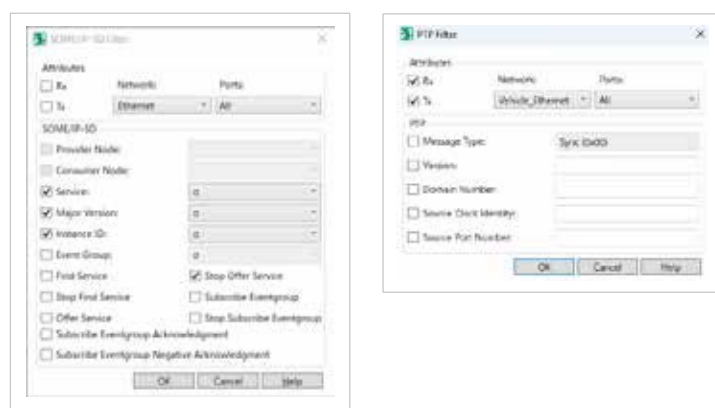
CANoe が持つ基本的な解析機能を使って Ethernet の解析を行うことができます。たとえば、トレース Window は送受信している Ethernet パケットのプロトコルを自動的に解釈して表示します。

トレース Window の詳細 Window には Ethernet パケットの詳細なヘッダー情報を確認することができます。



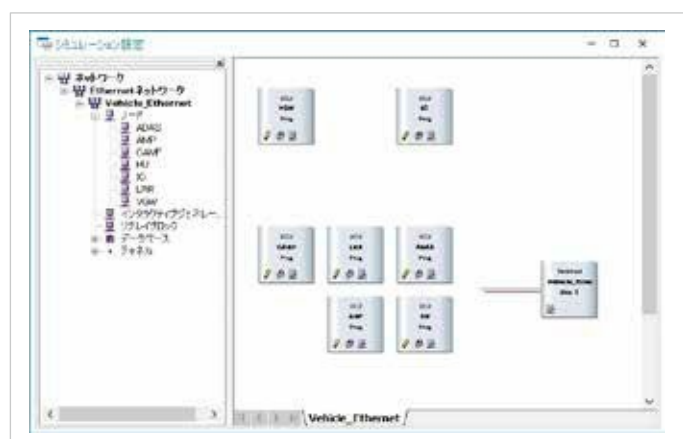
トレース Window のフィルター機能を使って必要な情報のみを表示することが可能ですので、効率的な解析を行うことができます。



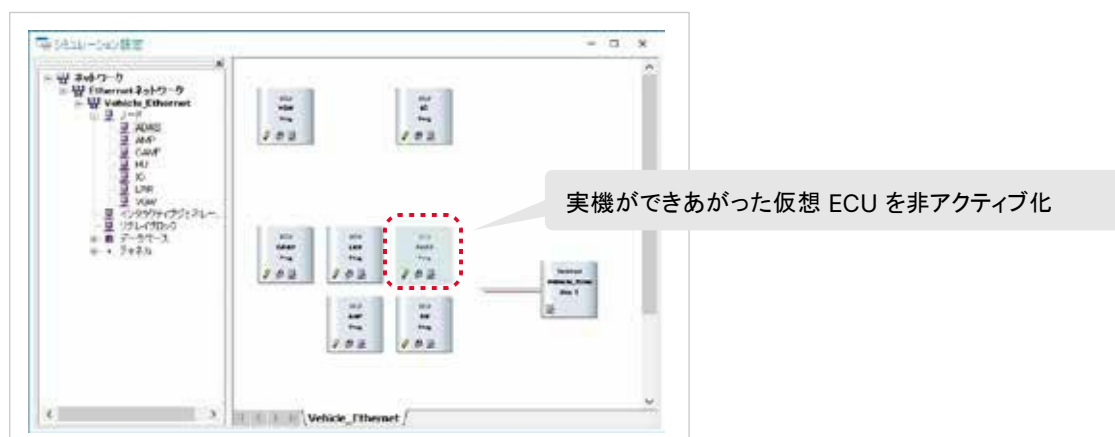


2.2 Ethernet ノードのシミュレーション

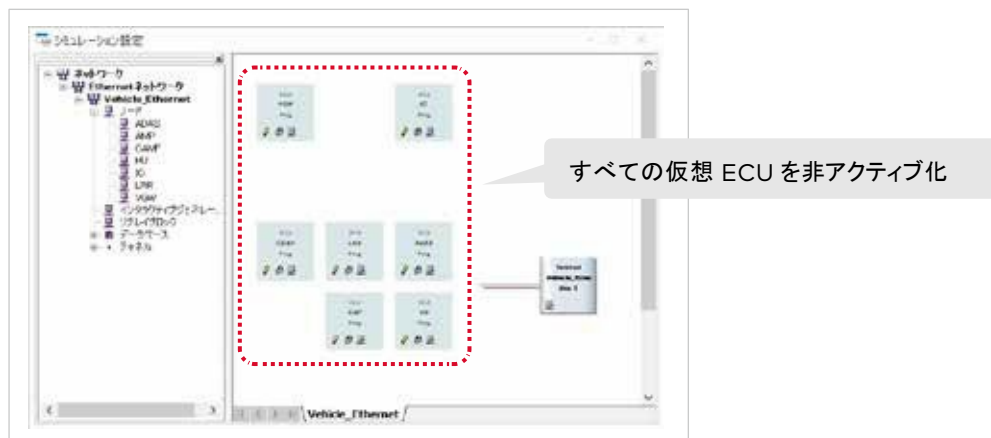
CANoe のシミュレーション機能は、複数の仮想 ECU を CANoe 上に設置して通信を行うことができます。開発初期では、ネットワーク上のすべての ECU を仮想 ECU として設置してシミュレーションを行います。



開発中期では、できあがった実機の ECU と仮想 ECU を組み合わせてシミュレーションします。



開発後期では、すべての ECU を実機で通信させ、CANoe は解析用途として使用できます。



また、CANoe には CAPL(読み方:キャプル)という専用のプログラミング言語があります。

CAPL を使用して各仮想ノードの振る舞いを実装することができます。

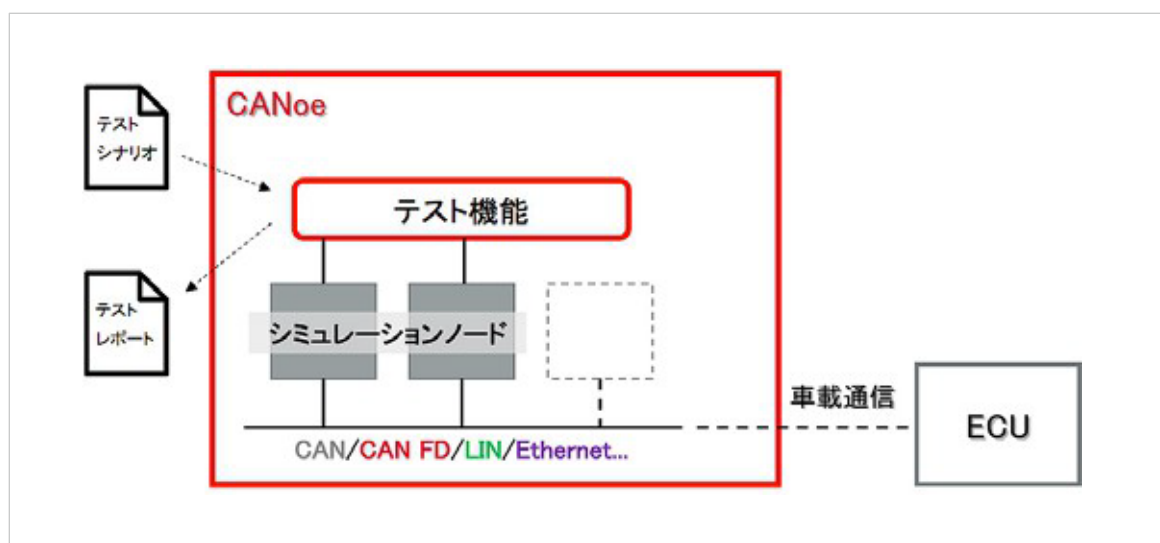
CAPL では Ethernet パケットを自由に作成・送信することができたり、さまざまな受信処理や送信処理を実行できます。

仮想 ECU の振る舞いを記述した CAPL ファイルを作成して、仮想 ECU に割り当てることで仮想ノードをシミュレーションすることができます。

2.3 Ethernet ノードのテスト

CANoe のテスト機能は、仮想 ECU や実機の ECU に対してテストを実行できます。CANoe にテストシナリオをインポートし、テストシナリオを基に仮想 ECU や実機の ECU にテストを実行します。

すべてのテストが実行された後、テスト結果をテストレポートとして出力します。



3. CANoe.Ethernet の機能

CANoe.Ethernet には、Ethernet の解析やシミュレーション、テストを行うための CANoe.Ethernet 専用の機能やサンプルがいくつか搭載されています。

- 解析関連
 - トレース Window
 - プロトコルモニター
 - 統計 Window
 - フィルターブロック
- シミュレーション関連
 - Ethernet Packet Builder
 - TCP/IP スタック
 - CAPL
 - インタクションレイヤー
 - モデル生成ウィザード
- テスト関連
 - TC8 サンプルコンフィギュレーション

本章では、上記機能を簡単に紹介します。それでは順番に見ていきましょう。

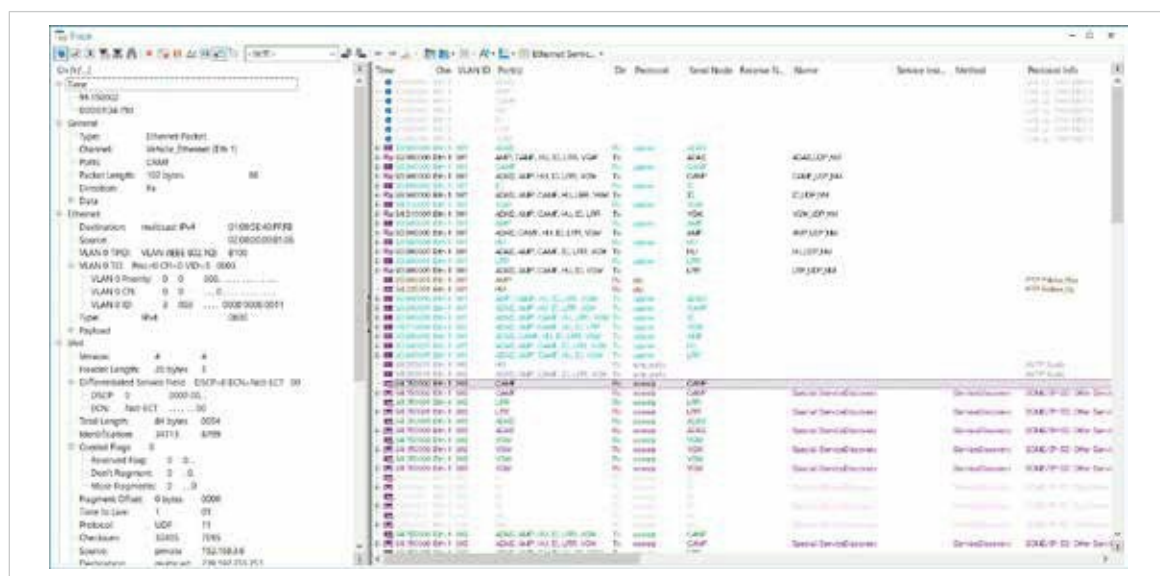
3.1 解析関連

3.1.1 トレース Window (CANoe/CANalyzer 共通機能)

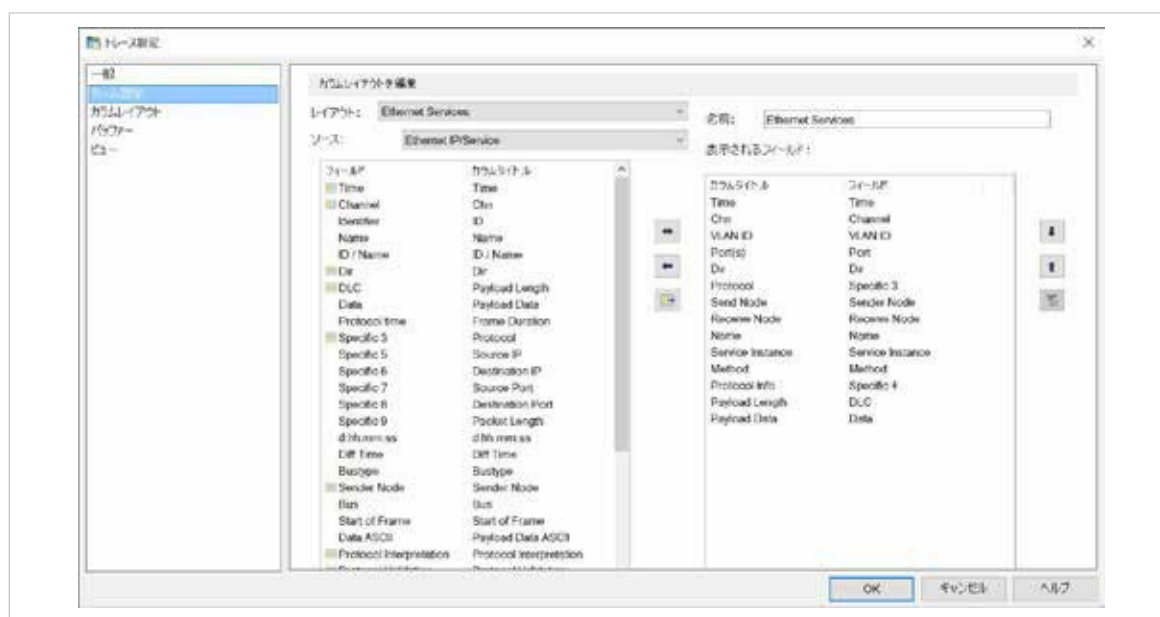
トレース Window は送受信しているメッセージを Window 上にテキスト行として表示します。

送受信しているメッセージのプロトコルを自動で解釈して、トレース Window に表示します。トレース Window に表示されたメッセージを選択すると、詳細ビューにメッセージのヘッダーやペイロードの情報が確認できます。

送受信しているメッセージ以外にも、Ethernet リンク情報や、TC10 のスリープ、ウェイクアップイベントなども表示します。



また、カラムレイアウトを自由に変更できるため、解析したいプロトコルに合わせてレイアウトを選択できます。

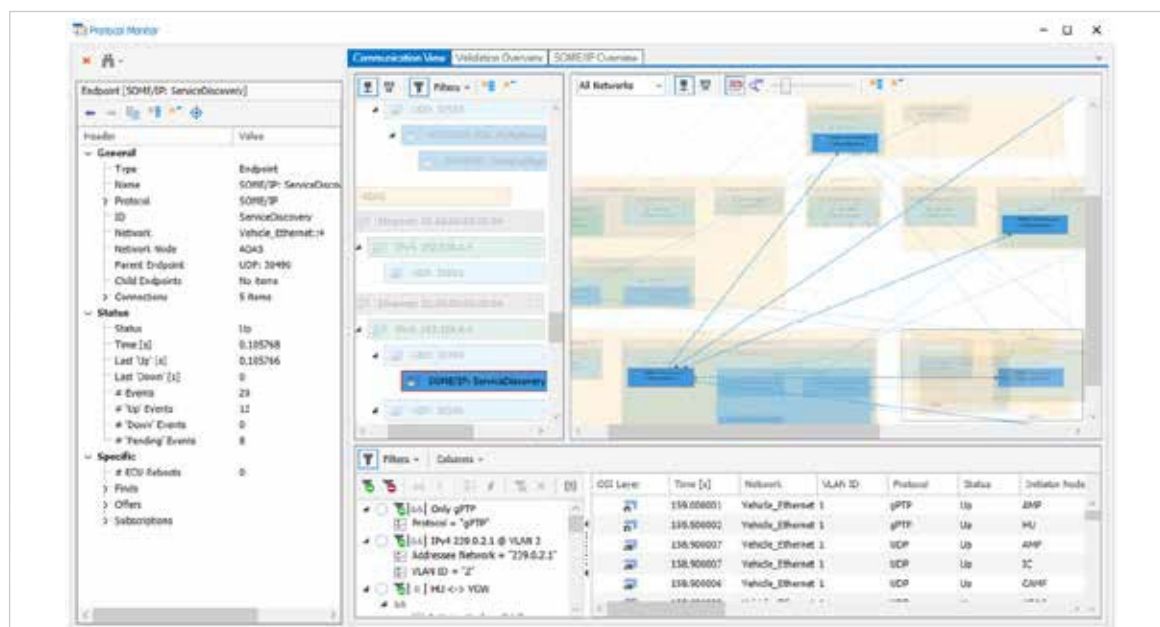


3.1.2 プロトコルモニター(CANoe/CANalyzer 共通機能)

プロトコルモニターは ECU 間の通信をグラフィカルに表示します。

CANoe でサポートしている Ethernet 関連の複数のプロトコルに対応しているため、下位レイヤーから上位レイヤーの通信状況を確認することができます。

フィルター機能を搭載しているため、必要なレイヤーの情報だけを表示させることができます。



3.1.3 統計 Window(CANoe/CANalyzer 共通機能)

バス統計 Window はさまざまなバスプロトコルに対応していますが、Ethernet でも同様に対応しています。

Ethernet 用のバス統計の項目が用意されており、各仮想 ECU のバス負荷状況の詳細を確認することができます。

Bus statistic	AMP (Eth1)	CANF (Eth1)	HJ (Eth1)	IC (Eth1)	LRR (Eth1)	VGW (Eth1)
Link speed [Mbit/s]	1000.0	1000.0	1000.0	1000.0	1000.0	1000.0
Rx Bus load [%]	0.00	0.00	0.25	0.00	0.00	0.00
Tx Bus load [%]	0.25	0.25	0.00	0.26	0.25	0.25
Rx Peak load [%]	0.00	0.00	0.26	0.00	0.00	0.01
Tx Peak load [%]	0.26	0.26	0.01	0.26	0.26	0.26
Rx Packets total [pkt]	31	28	45786	36	28	398
Tx Packets total [pkt]	45891	45882	352	46031	45882	45701
Rx Packet rate [pkt/s]	10	10	4030	10	10	30
Tx Packet rate [pkt/s]	4060	4060	60	4070	4060	4050
Rx Error packets [total]	0	0	0	0	0	0
Tx Error packets [total]	0	0	0	0	0	0
Connection status	Link up	Link up	Link up	Link up	Link up	Link up
Rx Bit rate [Mbit/s]	0.006	0.006	2.582	0.006	0.006	0.021
Tx Bit rate [Mbit/s]	2.598	2.598	0.008	2.606	2.598	2.592
Rx Peak rate [Mbit/s]	0.027	0.029	2.602	0.070	0.029	0.109
Tx Peak rate [Mbit/s]	2.650	2.640	0.197	2.660	2.640	2.652
Rx Byte rate [byte/s]	800	800	322760	800	800	2580
Tx Byte rate [byte/s]	324800	324800	4800	325780	324800	324000
Rx Bytes total [byte]	2624	2618	3666786	3802	2618	36927
Tx Bytes total [byte]	3673436	3672338	30963	3687976	3672338	3658208
BroadR-Reach SQI	-	-	-	-	-	-

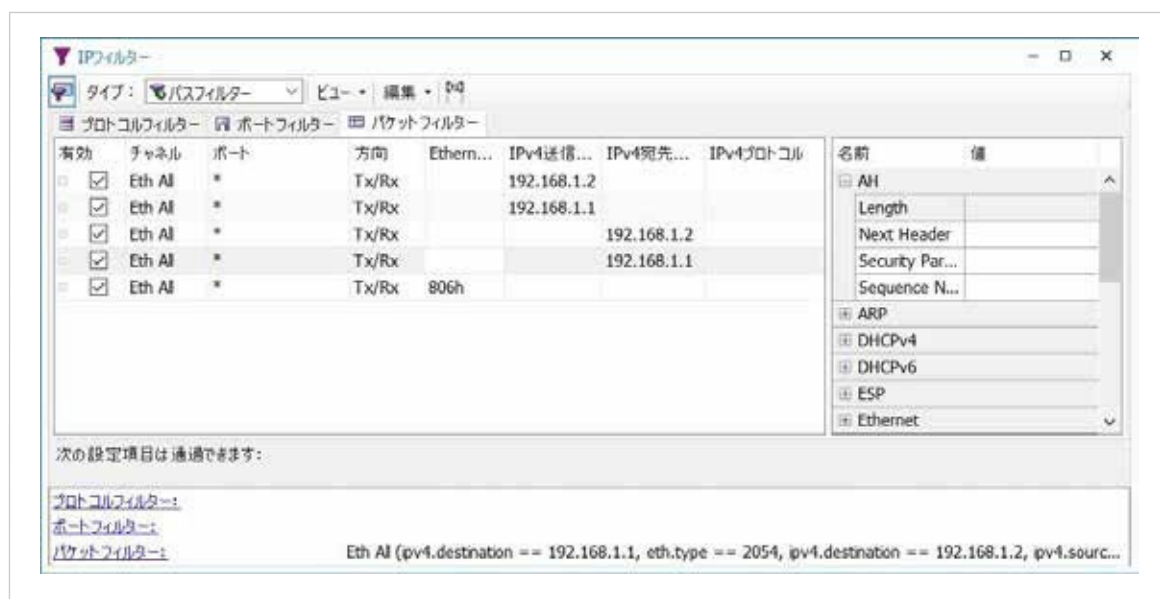
3.1.4 フィルターブロック(CANoe/CANalyzer 共通機能)

フィルターブロックは、測定設定 Window の各種解析機能(トレース Window やログブロック)に対し、それぞれフィルターを設定することができます。

たとえば、複数のトレース Window を用意して、各トレース Window に異なる条件のフィルターを行うことができます。

これにより、各トレース Window に必要な情報のみ表示させることができ、効率的に解析できます。

フィルター設定は以下のような GUI で簡単に設定できます。



3.2 シミュレーション関連

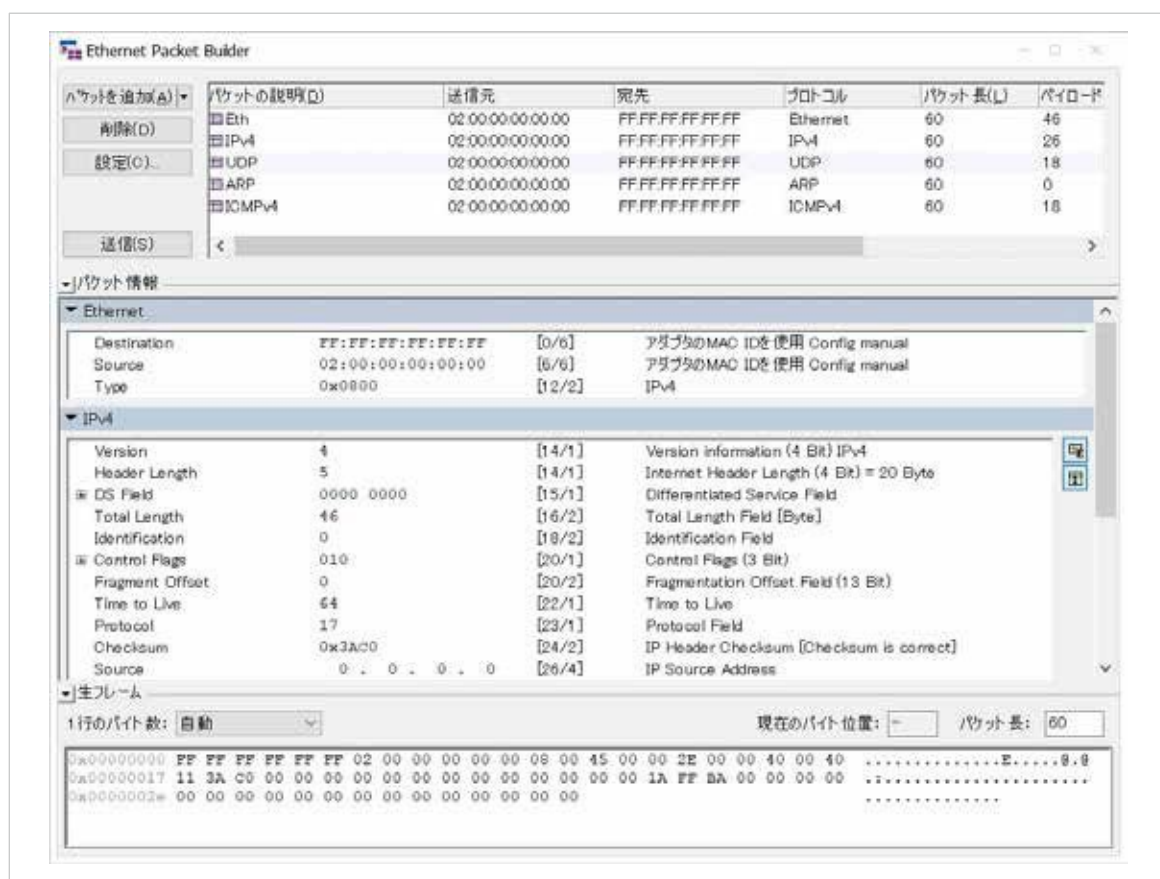
3.2.1 Ethernet Packet Builder (CANoe/CANalyzer 共通機能)

Ethernet Packet Builder は、専用の GUI から任意の Ethernet Packet を設定することができます。

Ethernet プロトコルやヘッダー情報、ペイロードを自由に設定できます。

チェックサムやパケット長の自動計算の有効／無効化ができるため、エラーパケットを設定することもできます。

設定した Ethernet Packet は送信ボタンを押すことで簡単に送信できます。



3.2.2 TCP/IP スタック(CANoe 専用機能)

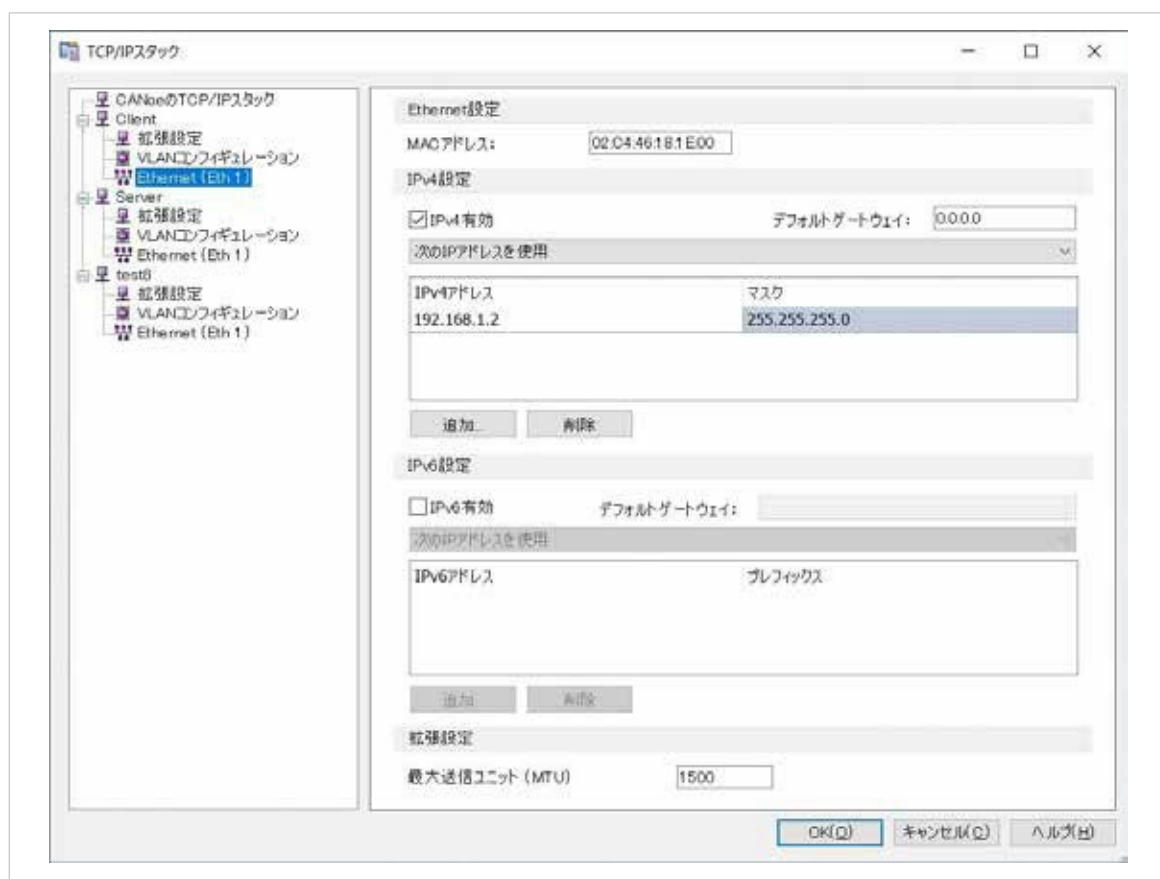
CANoe ではシミュレーションノードが通信に使用する TCP/IP スタックを次の 3 項目から選択できます。

- シミュレーションノード個別の TCP/IP スタック
- CANoe 共通の TCP/IP スタック
- OS のネットワークスタック

OS のネットワークスタック以外を選択した場合、TCP/IP スタックを任意に設定できます。IP アドレスや MAC アドレス、VLAN などを設定できます。

なお、OS のネットワークスタックを選択した場合は、Windows のネットワーク経由で通信を行います。

そのため、スタックの設定は PC のネットワークアダプタから設定する必要があります。



3.2.3 CAPL (CANoe/CANalyzer 共通機能)

CANoe には CAPL (読み方: キャプル) という専用のプログラミング言語があり、この言語を用いて仮想 ECU の振る舞いをプログラミングできます。

CAPL を使用することでさまざまな受信処理や送信処理を実行できます。

仮想 ECU の振る舞いが記述された CAPL ファイルを仮想 ECU に割り当ててシミュレーションします。

CANoe.Ethernet では、Ethernet 専用の CAPL 関数を用意しています。

その関数を利用することで Ethernet パケットを自由に作成・送信することができます。

以下は CAPL の記述例です。

Ethernet パケットを送受信した際に呼び出される CAPL 関数 "on ethernetPacket" を定義して、その関数内でさまざまな処理を記述することができます。

```
on ethernetPacket *
{
    if ((this.msgChannel > 0) && (this.msgChannel <= kPortCount))
    {
        if (this.dir == TX)
        {
            gPort[this.msgChannel - 1].waitForTxAck = 0;

            OutputPendingPackets();
        }
        else
        {
            int bufferIndex;

```

Ethernet パケットを送受信した時に呼び出される関数

任意の処理

また、TCP や UDP、SOME/IP などの Ethernet パケットを設定して送信することができます。

たとえば以下のように、UDP パケットのヘッダーやペイロードのデータを記述し、最後に output 関数で UDP パケットを送信することができます。

```
ethernetPacket packet;

packet.udp.Init();

packet.source = EthGetMacAddressAsNumber("20:00:00:00:00:01");
packet.destination = EthGetMacAddressAsNumber("FF:FF:FF:FF:FF:FF");
packet.ipv4.source = ipGetAddressAsNumber("192.168.1.1");
packet.ipv4.destination = ipGetAddressAsNumber("192.168.1.2");
packet.udp.source = 3;
packet.udp.destination = 4;
packet.udp.SetData("Hello world", 11);

packet.CompletePacket();
output(packet);
```

パケットの作成

パケットの初期化

ヘッダーパラメータの設定

ペイロードデータの設定

パケットのチェックサムとパケット長の自動計算

UDP パケットの送信

3.2.4 インタクションレイヤー (CANoe 専用機能)

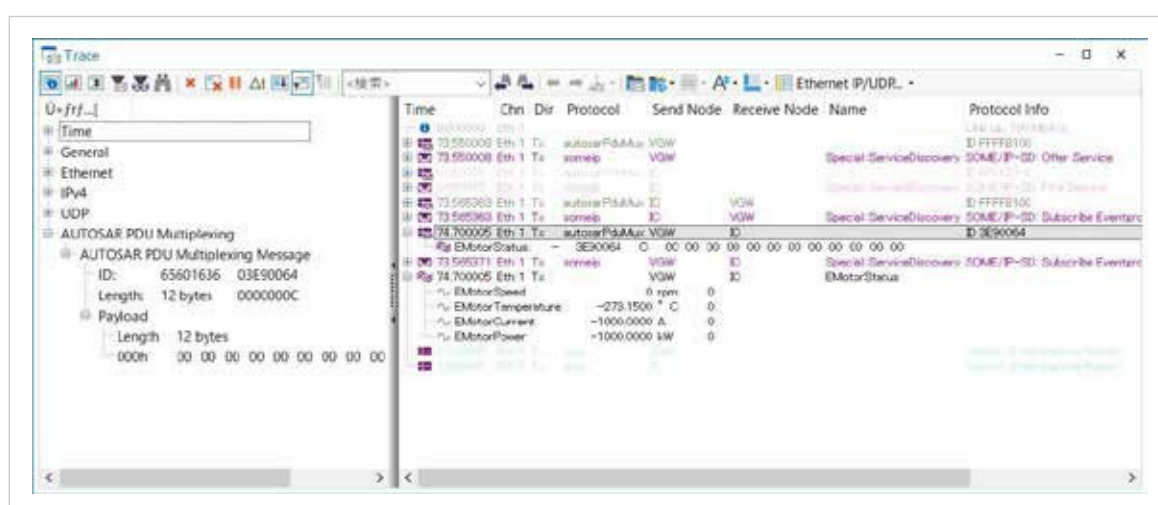
CANoe に仮想 ECU のシミュレーションをするためには、仮想 ECU の振る舞いを模擬する必要があります。CAPL を用いて仮想 ECU のすべての振る舞いを実装するには大きな工数がかかります。

その課題に対応するため、さまざまな専用関数や、仮想 ECU の振る舞いを模擬する機能を提供しています。それを総称してインタクションレイヤーと呼んでいます。

Ethernet 関連のインタクションレイヤーは以下の 3 種類です。

- **AUTOSAR Ethernet インタクションレイヤー**

Ethernet 経由で AUTOSAR PDU の送信をシミュレーションするための関数などを提供しています。

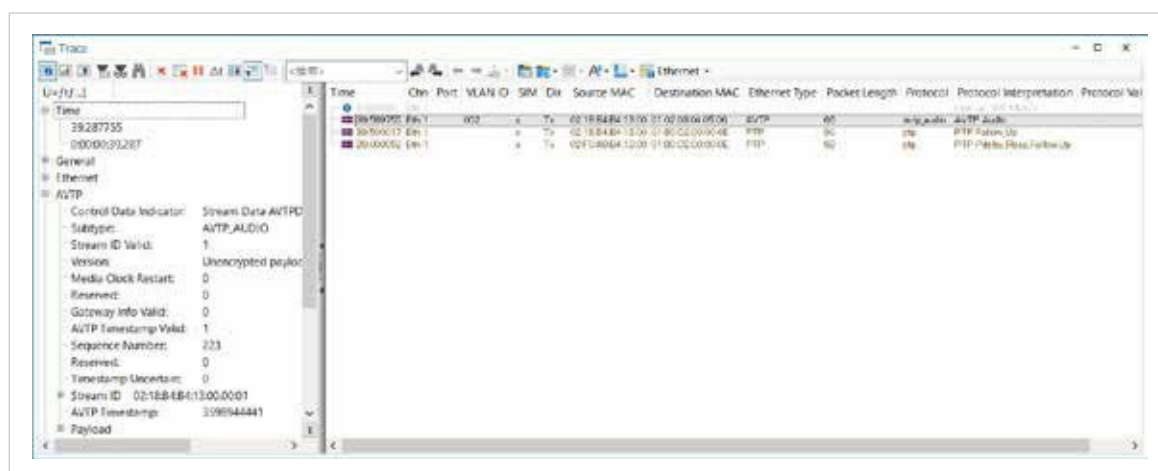


- **AVB インタクションレイヤー**

Ethernet AVB のトーカーとリスナーをシミュレーションするための関数などを提供しています。

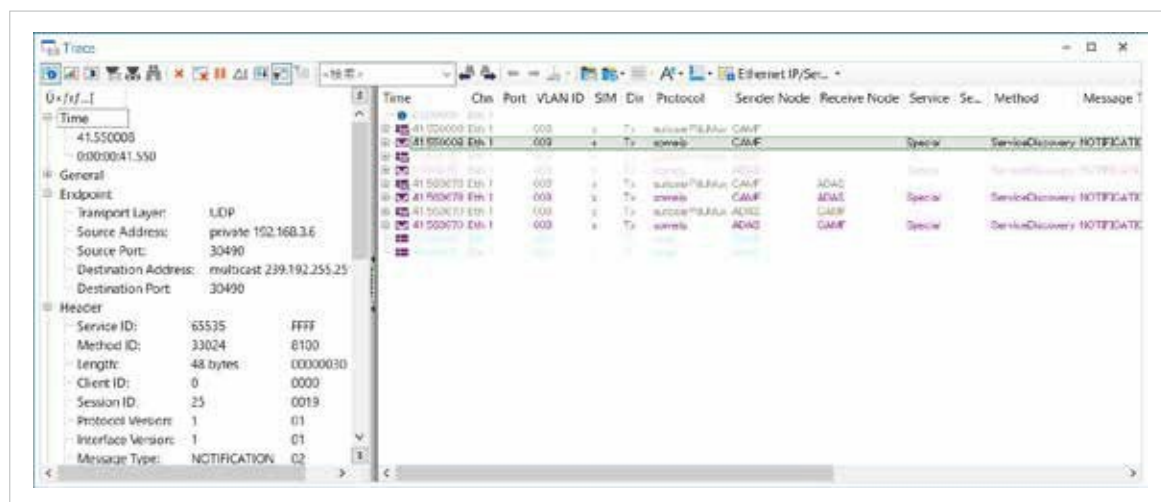
音声や動画の転送や再生に対応しています。

メディアファイルやマイク、スピーカーを操作する関数も提供しています。



- **SOME/IP インタラクションレイヤー**

SOME/IP、サービスディスカバリーに対応した仮想ノードを構築するための関数などを提供しています。



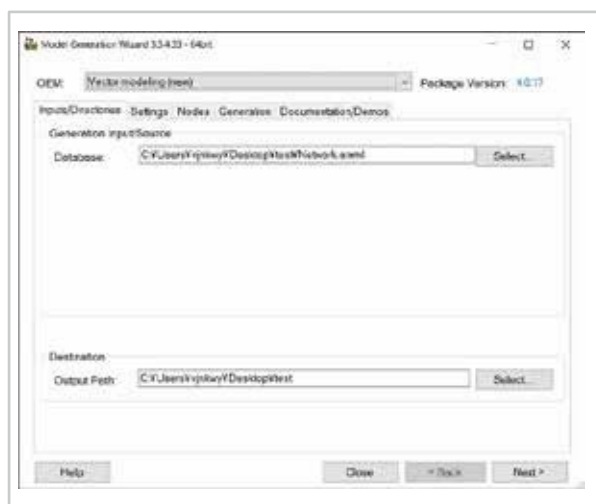
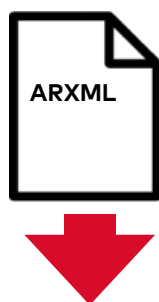
3.2.5 モデル生成ウィザード(CANoe 専用機能)

モデル生成ウィザードはシミュレーション環境を自動で構築する機能です。

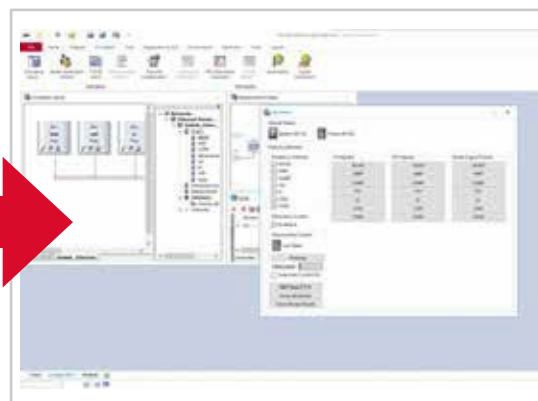
データベースファイル(ARXML)の情報を読み取り、CANoe のコンフィギュレーションファイルを自動生成します。

自動生成されたコンフィギュレーションファイルを編集し、任意のシミュレーション環境にカスタマイズ可能です。

なお、モデル生成ウィザードでサポートしているネットワークは CAN、Ethernet および FlexRay です。



モデル生成ウィザード



CANoe コンフィギュレーション

また、自動車メーカー固有の通信仕様に対応したコンフィギュレーションを生成する

「OEM パッケージ」も用意しています。

CANoe 標準機能には含まれないため、データベースと併せて使用する必要があります。

3.3 テスト関連(CANoe 専用機能)

車載 Ethernet ECU のテスト仕様「Automotive Ethernet ECU Test Specification (TC8 ECU Test)」が OPEN Alliance から発行されています。

なお、OPEN Alliance とは、車載 Ethernet 技術の仕様を統一するために作られた非営利組織です。

OPEN Alliance 内ではいくつかのグループに分かれて、車載 Ethernet 技術の開発、規格化、仕様策定を行っています。

各グループは"Technical Committee、TC#"と呼ばれています。車載 Ethernet ECU のテスト仕様は TC8 が仕様策定しています。

CANoe は TC8 が策定したテスト仕様を基にサンプルコンフィギュレーションを提供しています。TC8 で策定されたテストの大部分を実行することができます。



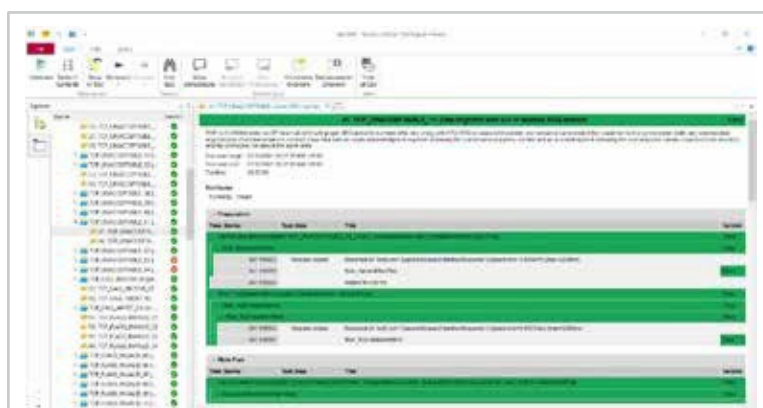
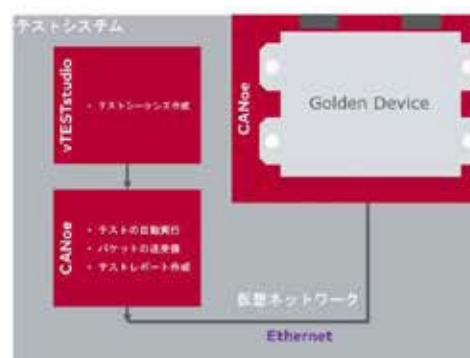
サンプルコンフィギュレーションのテストシナリオは、テスト編集ツールである vTESTstudio で実装されているため、vTESTstudio を使用することでテストシナリオを簡単に編集することができます。

サンプルコンフィギュレーションにはテスト対象の ECU (Golden Device) を仮想 ECU として実装しているため、シミュレーション環境下でテストが実行でき、その結果をリファレンスとして確認することができます。

CANoe でテストを実行すると、テストレポートファイルが自動生成されます。

テストの判定結果や詳細な情報が記録されています。

テストレポートファイルは専用のビューワーが用意されていますので、テストレポートの閲覧・解析を容易に行うことができます。



4. Ethernet 通信の ネットワークインターフェイス

Ethernet 用ネットワークインターフェイス(以後、"VN5000 インターフェイス"と総称)は Ethernet ネットワークの解析、シミュレーション、テストや妥当性検証に使用できる高機能なインターフェイスです。

用途に応じて小型・軽量タイプのものから多機能タイプのものまでさまざまなラインナップを用意しています。

なお、一部の VN5000 インターフェイスは本章で紹介するいくつかの機能に対応していません。

VN5000 インターフェイスの詳細な仕様につきましては、弊社 Web サイトの「見積依頼」もしくは弊社営業担当までお問い合わせください。



【特長】

VN5000 インターフェイスには、IEEE 規格 100BASE-T 1、1000BASE-T 1 や 10BASE-T/100BASE-TX/1000BASE-T に対応する物理層を搭載しています。

その他にも 10GBASE-T などの高速な Ethernet の物理層を搭載するインターフェイスや、CAN チャネル (CAN FD 対応) やアナログ/デジタル入出力の IO チャネルを搭載しているインターフェイスもあります。

VN5000 インターフェイスは USB または Ethernet を使用して PC に接続することができます。

また、2 つ以上のインターフェイスを使用してシミュレーションや解析・テストを行う場合は、同期ケーブルを使ってインターフェイス同士を同期させることができます。

これにより、インターフェイス間のタイムスタンプが高精度で同期できます。

VN5000 インターフェイスは以下のようなさまざまな機能を搭載しています。

- TAP
- レイヤー2 スイッチ
- メディアコンバーター
- ハードウェアバースト送信
- エラーフレーム送信
- ハードウェアフィルタ

もちろん、上記の機能以外にも多くの機能を搭載していますが、ここでは上記機能について簡単に紹介します。

4.1 TAP

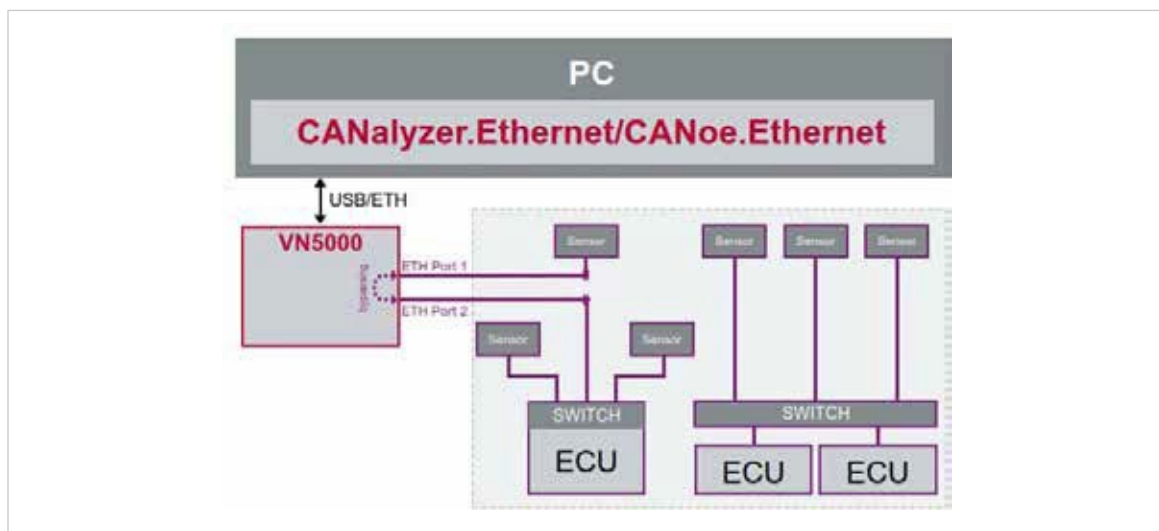
VN5000 インターフェイスは ECU 間の Ethernet 通信をモニターできます。

図のようにモニターしたいネットワークの間にインターフェイスを接続してモニターします。

Ethernet データの転送は非常に小さい遅延時間のため、Ethernet ネットワークへ大きな影響を与えません。

インターフェイスが持つ 2 つのポートを使用して 1 つの Ethernet ネットワークをモニターします。

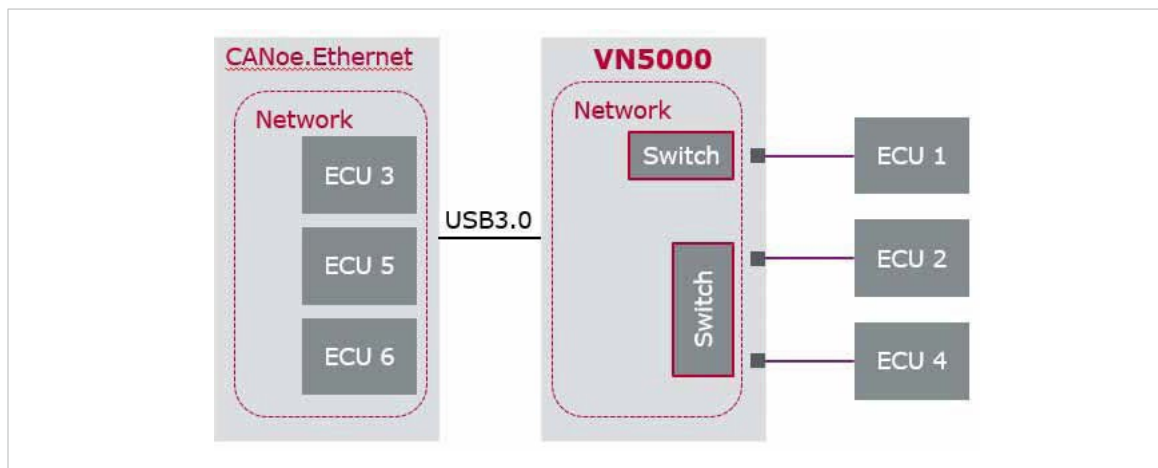
たとえば、インターフェイスが 4 つの Ethernet ポートを使用できる場合、一度にモニターできる最大のネットワーク数は 2 つまでとなります。



4.2 レイヤー2 スイッチ

VN5000 インターフェイスは単体でレイヤー2 のスイッチとして動作します。

VN5000 のスイッチを介して CANoe の仮想 ECU と実 ECU をシミュレーションすることができます。

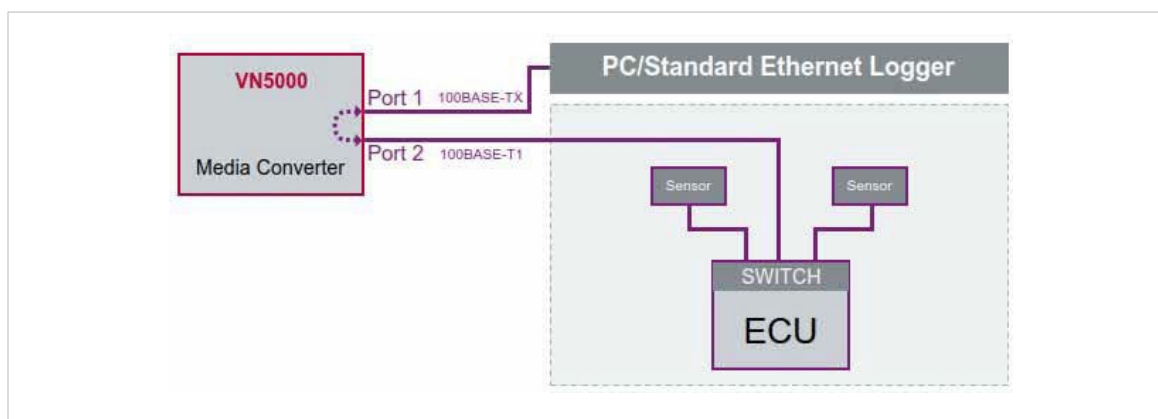


4.3 メディアコンバーター

VN5000 インターフェイスに搭載された 100BASE-T1/1000BASE-T1 ポートと 100BASE-TX/1000BASE-T ポートを使用して、物理層を自動的に変換できます。

メディアコンバーター機能に各ポートを割り当てた後、電源を入れるだけで動作します。メディアコンバートできる数は 100BASE-TX/1000BASE-T が搭載されているポート数に比例します。

そのため、100BASE-TX/1000BASE-T が搭載されていないインターフェイスは対象外です。



4.4 ハードウェアバースト送信

VN5000 インターフェイスから Ethernet フレームを任意のフレームレートでバースト送信できます。

この機能により PC の負荷を軽減しながら高負荷送信を実現できます。

1 つの Ethernet ポートから、フレーム長やデータが固定の 1 種類の Ethernet フレームをバースト送信できます。

それぞれの Ethernet ポートにバースト送信する Ethernet フレームを設定できます。

1 つの Ethernet ポートから複数の Ethernet フレームを一度にバースト送信したい場合は、本機能は使用できないため、CAPL を利用したバースト送信が必要です。

4.5 エラーフレーム送信

VN5000 インターフェイスから Ethernet エラーフレームを送信できます。テスト用途でさまざまな Ethernet フレームを送信できます。

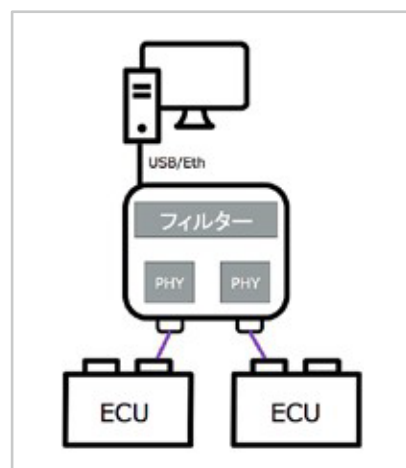
たとえば、ECU と競合する MAC アドレスの Ethernet フレームを設定することや、Ethernet プロトコルに定められた最小サイズよりも小さな Ethernet フレームを設定できます。

4.6 ハードウェアフィルター

送受信する Ethernet データを VN5000 インターフェイス上でフィルタリングできます。本機能により、不要なデータが PC に転送されないため、PC の処理負荷が軽減できます。解析データが低減され、効率的な解析が可能です。

フィルタリングできる条件は以下の項目です。

- MAC アドレス
- Ethernet タイプ
- VLAN ID
- IP アドレス
- TCP/UDP ポート番号
- 送受信方向



5. ソフトウェアとハードウェアを使用するための事前準備

ここからは実際に CANoe.Ethernet と VN5000 インターフェイスファミリーを使用するために必要な準備を説明します。

以下の内容を実行してソフトウェアとハードウェアが使用可能な状態にします。

最後にサンプルコンフィギュレーションを実行して CANoe が正常に動作することを確認します。

それでは早速準備をはじめましょう。

- ソフトウェアのインストール
- ハードウェアの準備
 - VN5000 インターフェイスファミリー
 - Ethernet ケーブル
- サンプルコンフィギュレーションの起動

5.1 ソフトウェアのインストール

CANoe を購入されたお客様には、ダウンロード方式によるご納品となります。

主に以下の内容を実施します。詳細は納品物に含まれるドキュメントをご参照ください。

- CANoe.Ethernet のインストール
- 最新のサービスパックのインストール
- ライセンスアクティベーション
- 日本語化キットのインストール
- Vector Driver Setup のインストール

5.2 ハードウェアの準備

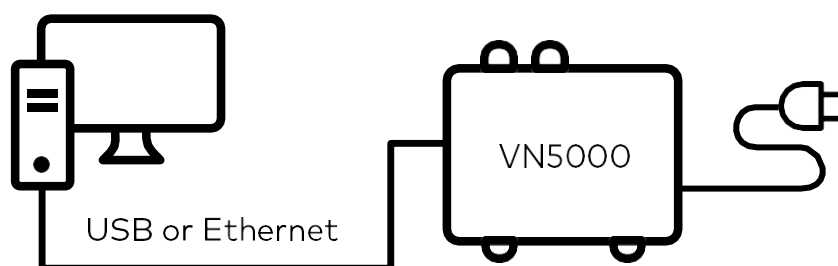
次にハードウェアを準備します。

5.2.1 VN5000 インターフェイスの起動と接続

VN5000 インターフェイスを用意します。

はじめに VN5000 インターフェイスを USB または Ethernet で PC と接続します。

USB 給電できないインターフェイスには電源も接続します。



次に VN5000 インターフェイスと PC を USB または Ethernet で接続します。

Windows のスタートメニューから Vector Hardware Manager を起動します。なお、Vector Hardware Manager はバージョンに応じて操作方法が異なる可能性

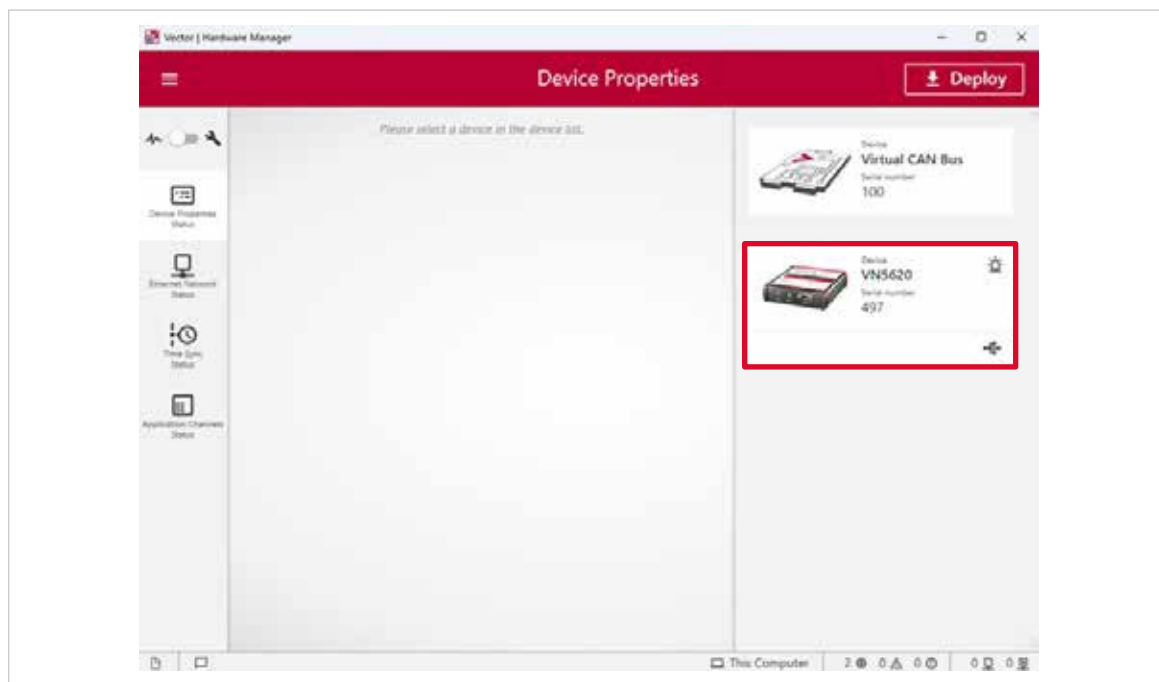
がありますのでご注意ください。なお、本書においては、バージョン 24.20.16 での操作方法をもとに説明しております。



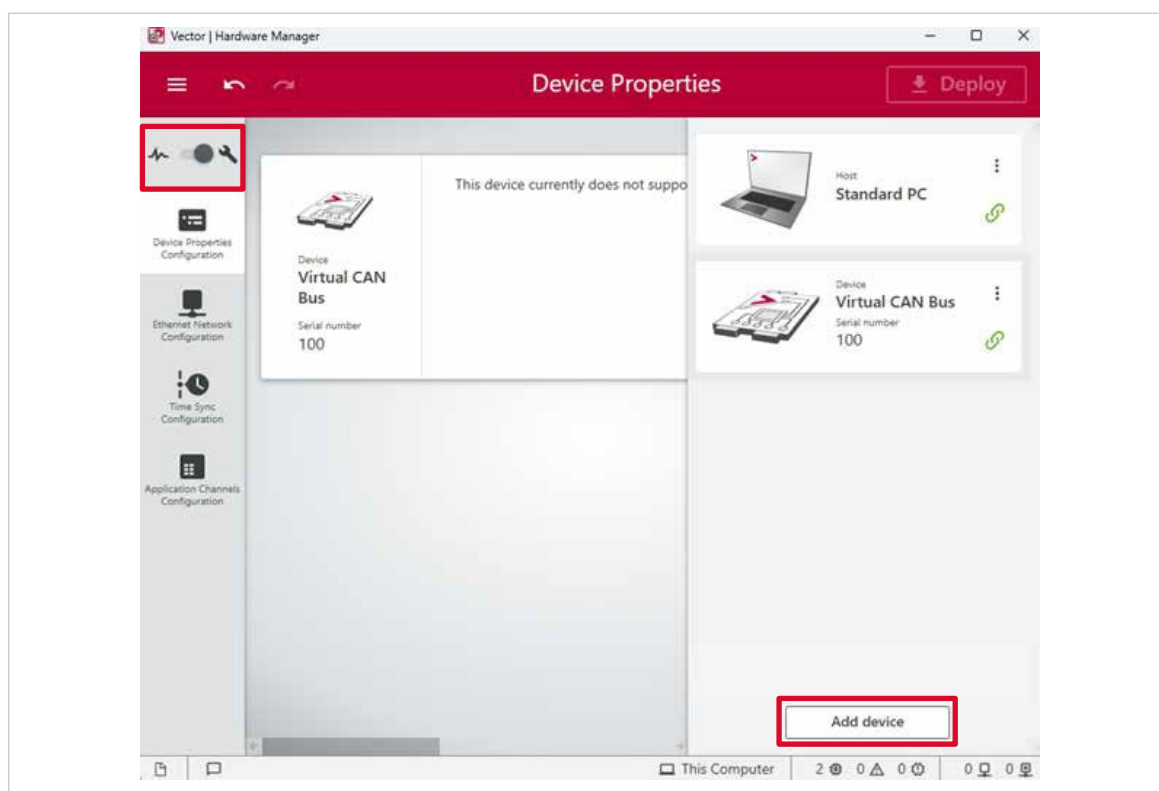
はじめに、USB 接続できるインターフェイスの場合の手順を記載します。

Vector Hardware Manager の起動時、USB 接続のインターフェイスを認識すると、起動時の画面右側のメニューにインターフェイスのアイコンが下図のように表示されます

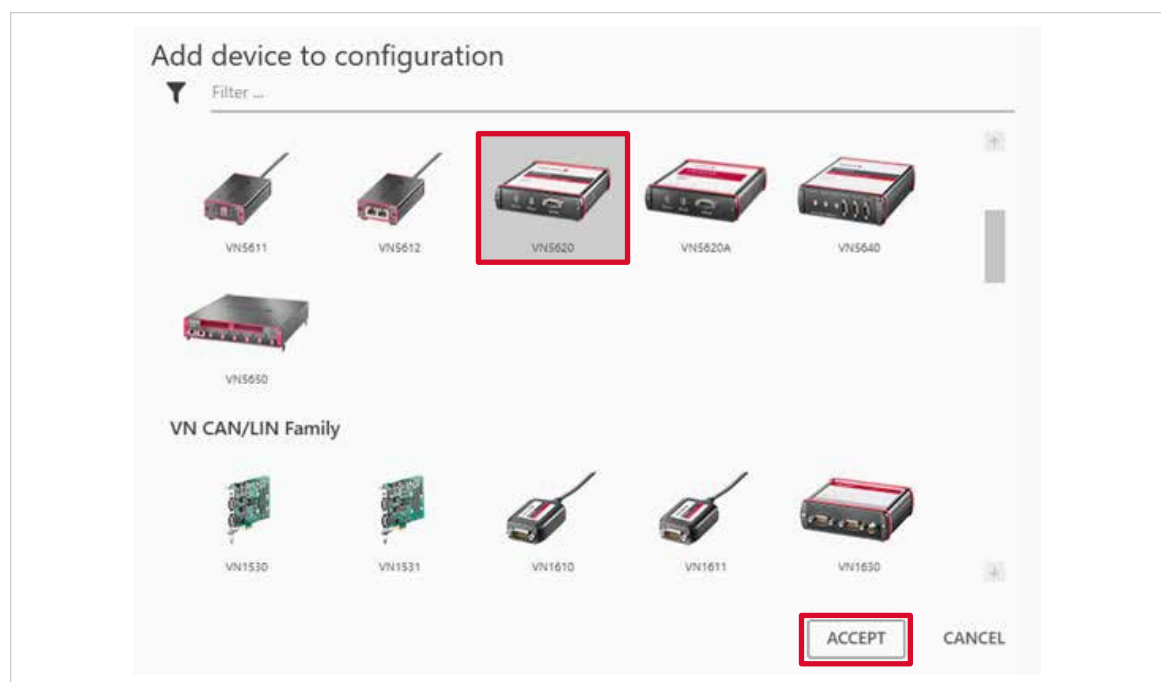
(下図は VN5620 の例です)。



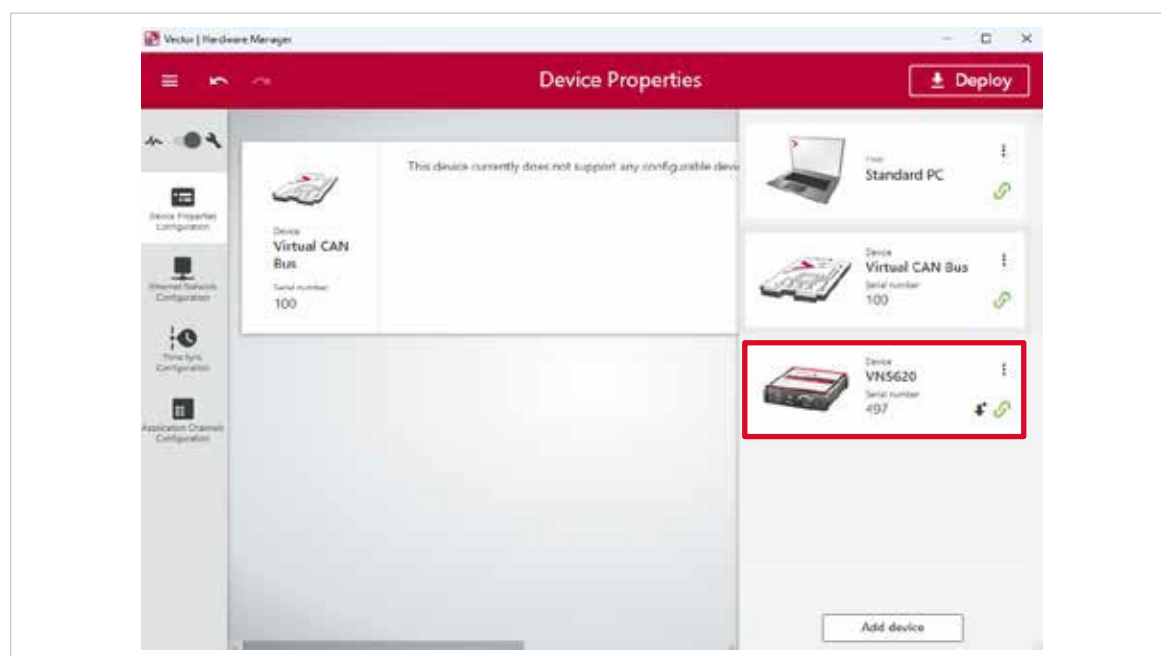
対象のインターフェイスが表示されていない場合は、画面左上のスイッチバーを右側(スパナのアイコン側)に設定し、画面右下の "Add device" を選択します。



インターフェイスの選択ウィンドウが表示されるので、追加したいインターフェイスを選択して"ACCEPT"をクリックします。



対象のインターフェイスが接続されていることを確認します。

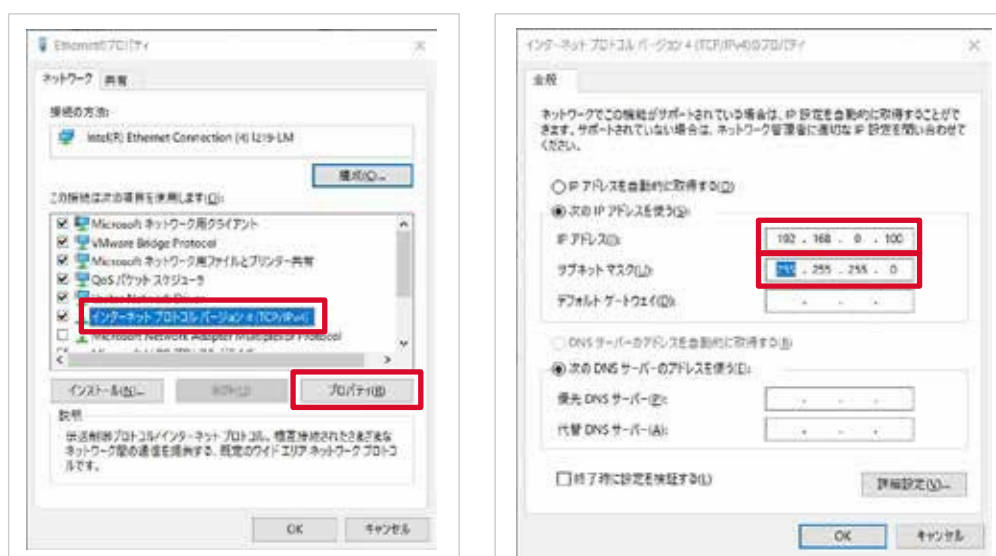


続いて、Ethernet で PC と接続する場合の手順を記載します。

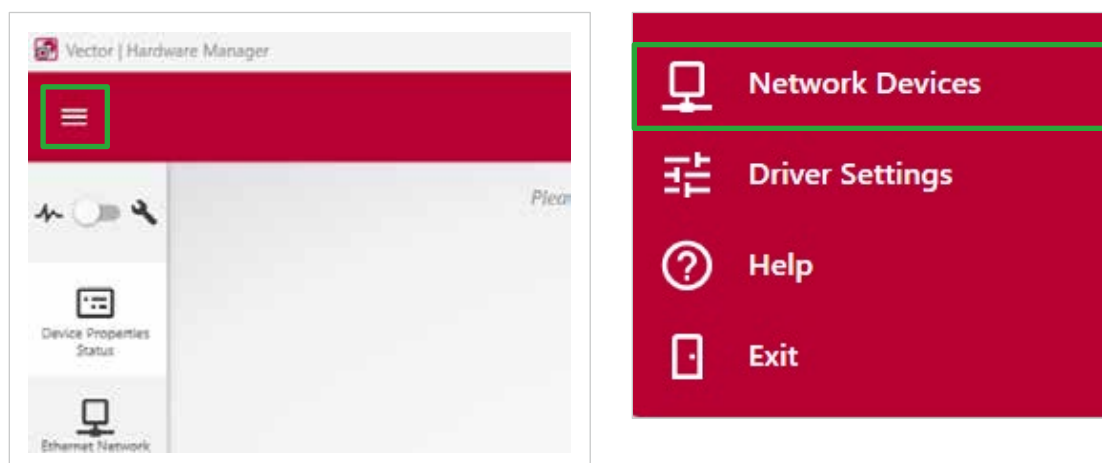
PC とインターフェイスが Ethernet で接続されている場合、ネットワークで認識できるように IP アドレスを変更する必要があります。

インターフェイスの IP アドレスはデフォルトでは以下のようになっています。デフォルト IP アドレス:192.168.0.1

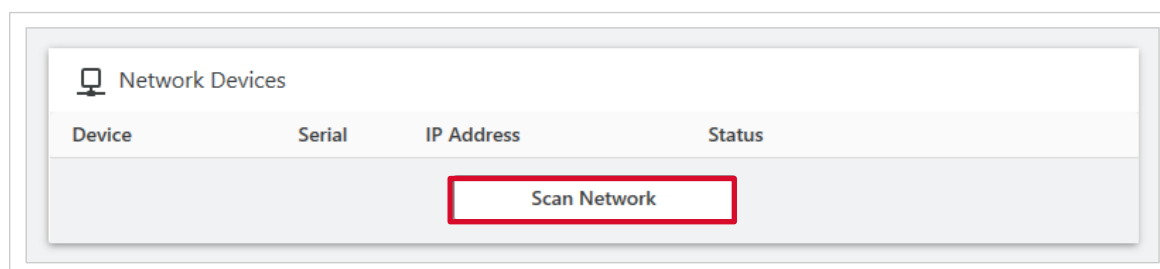
そのため、PC のネットワークアダプタの IP アドレスをインターフェイスの IP アドレスと同じネットワークにします。ネットワークアダプタの設定画面から IPv4 のプロパティを開き、IP アドレスを変更します。



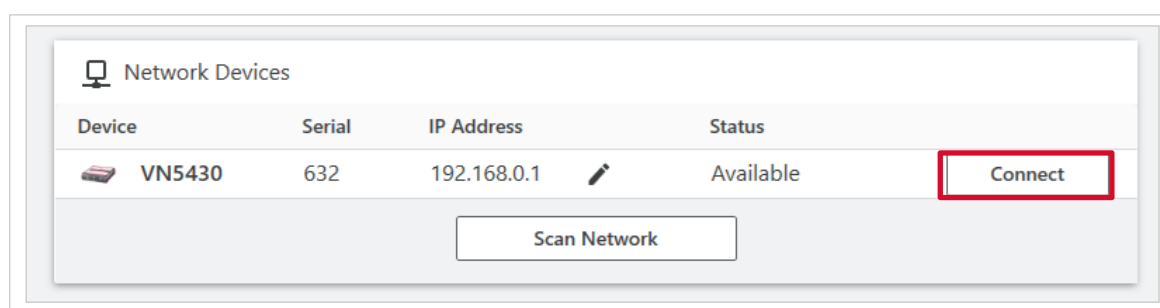
Vector Hardware Manager 起動時に表示される画面より、左上の"≡"をクリックし、Network Devices を選択します。



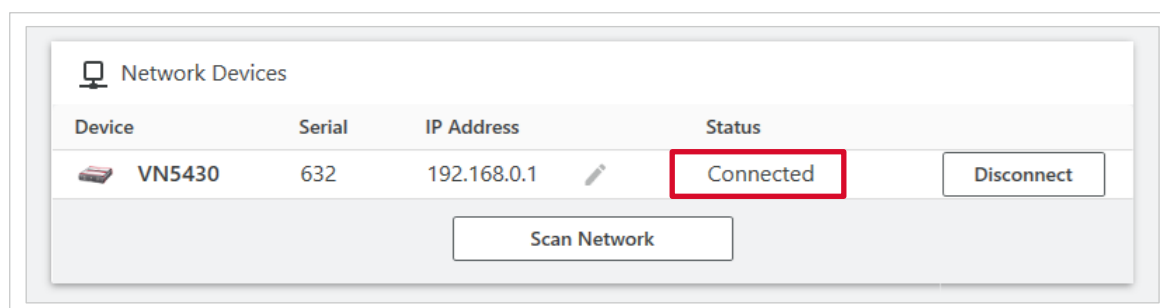
"Scan network"をクリックします。



PC に認識されているインターフェイスが表示されます。対象のインターフェイスが認識されていることを確認し、"Connect"をクリックします(下図は VN5430 の例です)。



インターフェイスが接続され、Status が"Connected"に変わります。



以上で VN5000 インターフェイスの起動と接続が確認できます。

5.2.2 Ethernet ケーブル

次に Ethernet ケーブルを用意します。

VN5000 インターフェイスは標準 Ethernet(100BASE-TX/1000BASE-T など)のポートと、車載 Ethernet (100BASE-T1/1000BASE-T1 など)のポートが搭載されています。

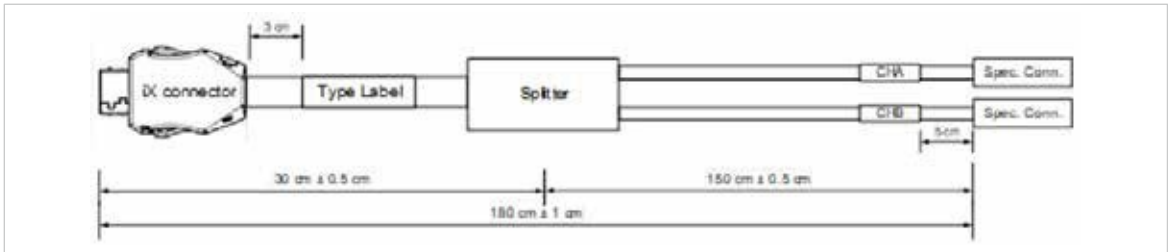
標準 Ethernet ポートは RJ45 コネクタのケーブルを使用して接続します。



車載 Ethernet ポートのケーブルは弊社から提供しています。



VN5000 インターフェイス側は iX-connector で接続し、2 つに分岐して接続相手にそれぞれ接続します。これは VN5000 インターフェイスの車載 Ethernet の各コネクタには 2 つのポートが割り当てられているためです。



iX-connector のピンアサインは以下のようにになっています。1 つのコネクタに 2 つの Ethernet ポートが存在しています。

PIN	Assignment	Color
1	CHBP	white/orange
2	CHBP	orange
3	Not connected	—
4	Not connected	blue
5	Not connected	white/blue



接続相手側のコネクタは以下の各種形状から選択できます。

なお、D -SUB コネクタは Male タイプを標準として提供しているため、D -SUB の Female タイプの提供については応相談となります。



H-MTD Z Jack



H-MTD Z Plug



HSD Jack



HSD Plug



MATenet Z Jack



MATenet Z Plug



D-SUB9 male



RJ45 Jack



RJ45 Plug

それぞれのコネクタのピンアサインは以下のようになっています。

Connector Type	Ethernet CH P	Ethernet CH N
H-MTD Z	Pin 1	Pin 2
HSD Z	Pin 2	Pin 3
MATenet Z	Pin 1	Pin 2
D-SUB9	Pin 4	Pin 5
RJ45	Pin 1	Pin 2

各コネクタの接続相手のピンアサインを確認し、正しく接続されることを確認します。

5.3 CANoe の動作確認

サンプルコンフィギュレーションを使って CANoe を実行してみましょう。

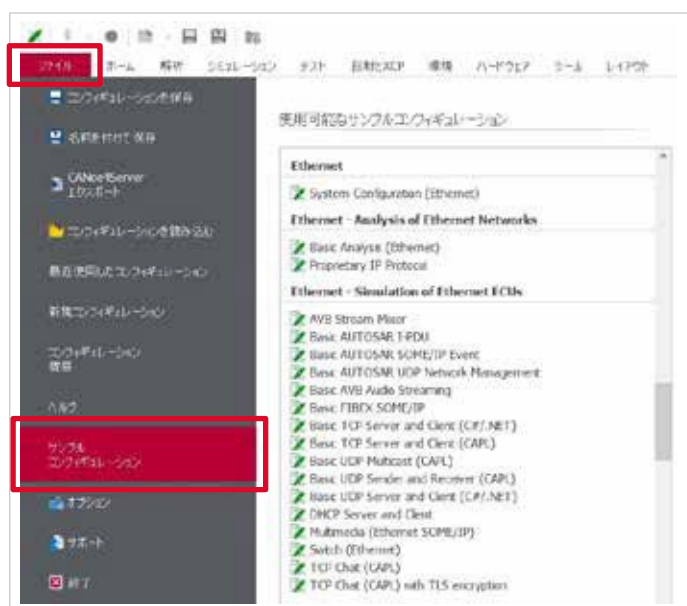
5.3.1 サンプルコンフィギュレーションの起動

はじめにサンプルコンフィギュレーションを起動してみましょう。

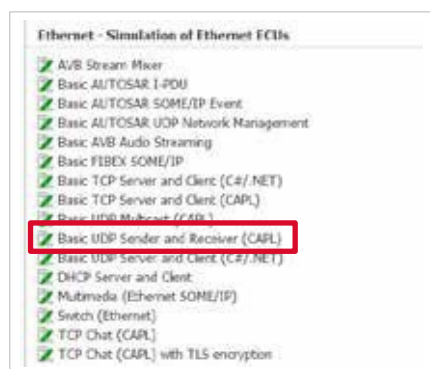
CANoe ではさまざまな用途に応じて Ethernet 関連のサンプルコンフィギュレーションを提供しています。

CANoe のサンプルコンフィギュレーションは以下のように CANoe から起動することができます。

1. CANoe を起動します。
2. CANoe のリボンメニューの"ファイル"から、"サンプルコンフィギュレーション"をクリックします。



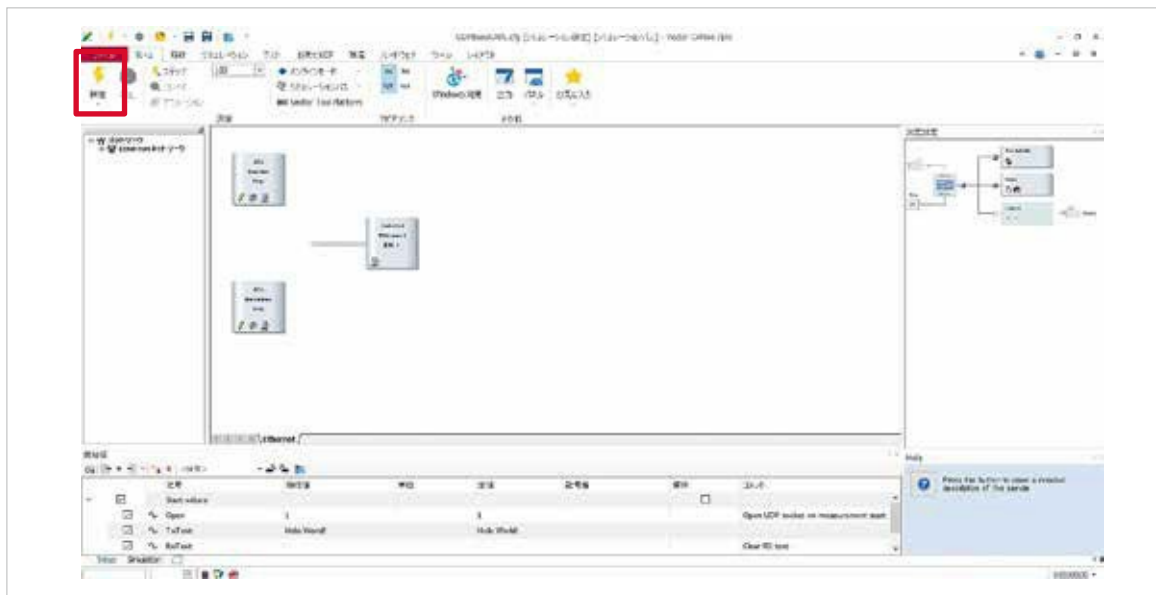
3. "Ethernet — Simulation of Ethernet ECUs"の"Basic UDP Server and Receiver (CAPL)"をダブルクリックしてサンプルコンフィギュレーションを開きます。



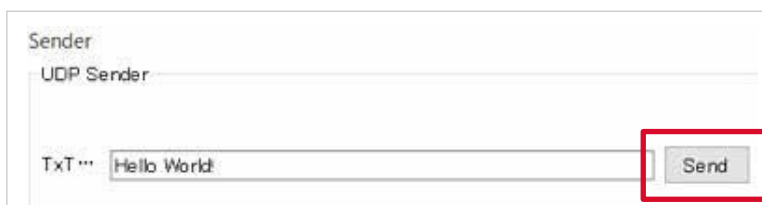
5.3.2 サンプルコンフィギュレーションの実行

サンプルコンフィギュレーションを起動した後は実行してみましょう。

リボンメニューの測定開始ボタン  から、CANoe を開始します。

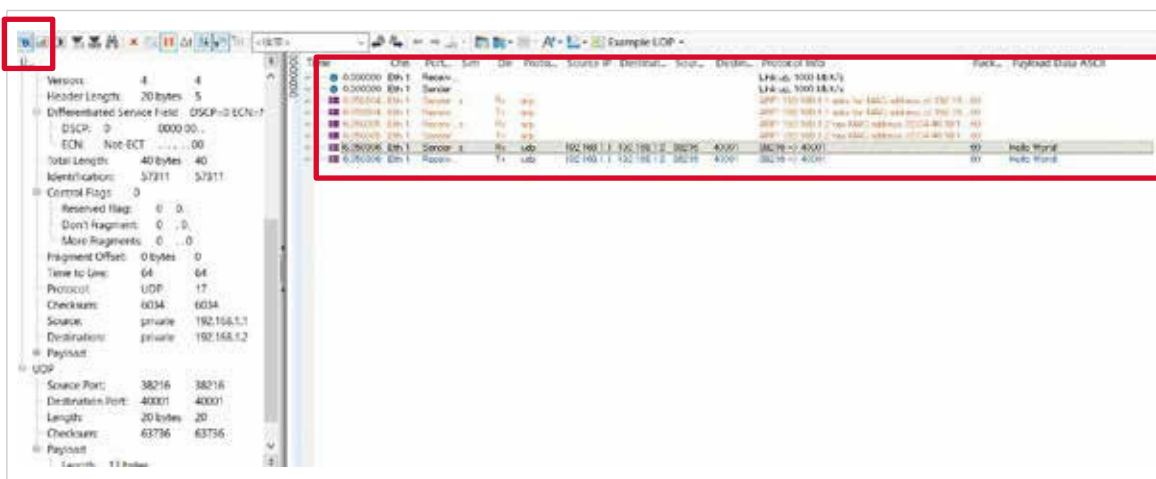


Sender パネルの"Send"ボタンを押すと、メッセージが送信されます。



送信されたパケットはトレース Window で確認できます。

詳細ビューアイコンをクリックして、パケットのヘッダー情報などを確認することができます。



6. 実際に使ってみよう

前章で CANoe と VN5000 インターフェイスドライバーのインストール状態に問題がないことが確認できました。

それでは CANoe.Ethernet と VN5000 インターフェイスを使って、実バスで Ethernet のシミュレーションを行う環境を作ってみましょう。

6.1 必要なもの

前章で確認した機材一式を用意します。

- CANoe がインストールされた PC
- Ethernet インターフェイス
- Ethernet ケーブル

6.2 作成する環境

今回は実バスで Ethernet シミュレーションする環境として、UDP パケットを送受信する環境を作ります。

CANoe の 1 つの仮想 ECU から、もう 1 つの仮想 ECU に対して UDP パケットを送信するような通信環境です。

はじめに接続構成を考えましょう。

【接続構成】

UDP パケットを送受信する環境を作るため、仮想 ECU が 2 つ必要です。どちらも CANoe 上で作成できます。

VN5000 インターフェイスは CANoe の仮想 ECU に対応する物理ポートが 2 つ必要です。

Ethernet ケーブルを使用して VN5000 インターフェイスの物理ポートを接続します。

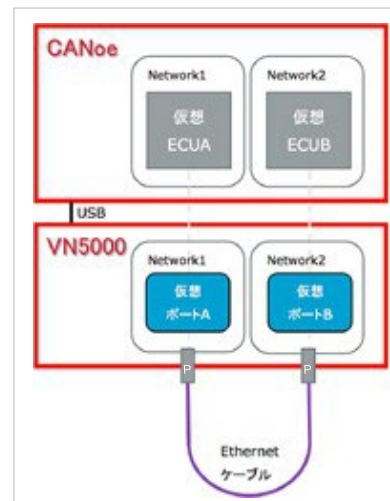
これで接続構成は完成しました。

【設定概要】

次に CANoe と VN5000 インターフェイスの設定を考えましょう。

CANoe に 2 つのネットワークを作成し、その中にそれぞれ 1 つの仮想 ECU を配置して、UDP パケットの送信側と受信側の仮想 ECU を用意します。

仮想 ECU A が UDP パケットを送信し、仮想 ECU B がそれを受信します。



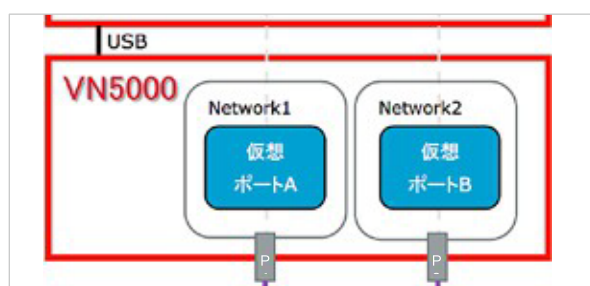
Ethernet インターフェイスは CANoe の仮想 ECU と物理ポートの接続先を 1 対 1 で通信するように設定します。2 つのネットワークを作成し、その中にそれぞれ仮想 ECU A、B に対応する仮想ポート A、B と物理ポート P1、P2 を配置します。

これで作成する環境の検討ができました。

6.3 インターフェイスの設定

それでは、さきほど検討した環境に従って Ethernet インターフェイスの設定を行いましょう。

VN5000 インターフェイスは CANoe の仮想 ECU と物理ポートの接続先を 1 対 1 で通信する設定です。2 つのネットワークの中にそれぞれ仮想ノード A、B に対応する仮想ポート A、B と物理ポート P1、P2 を配置します。

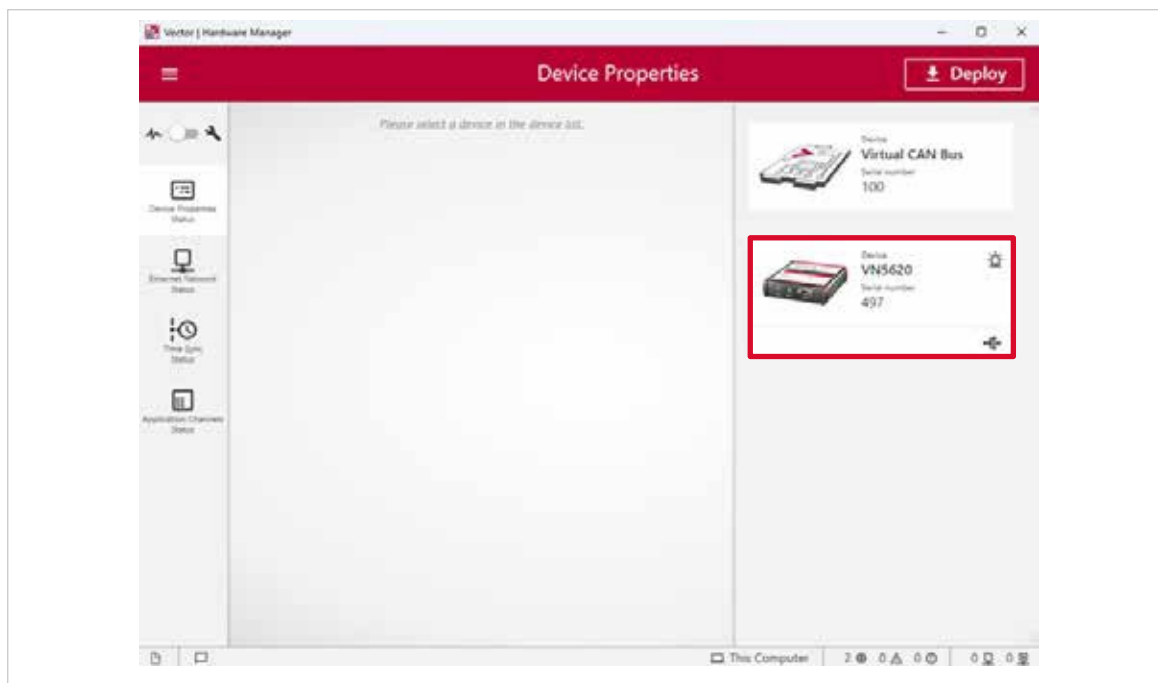


【作成手順】

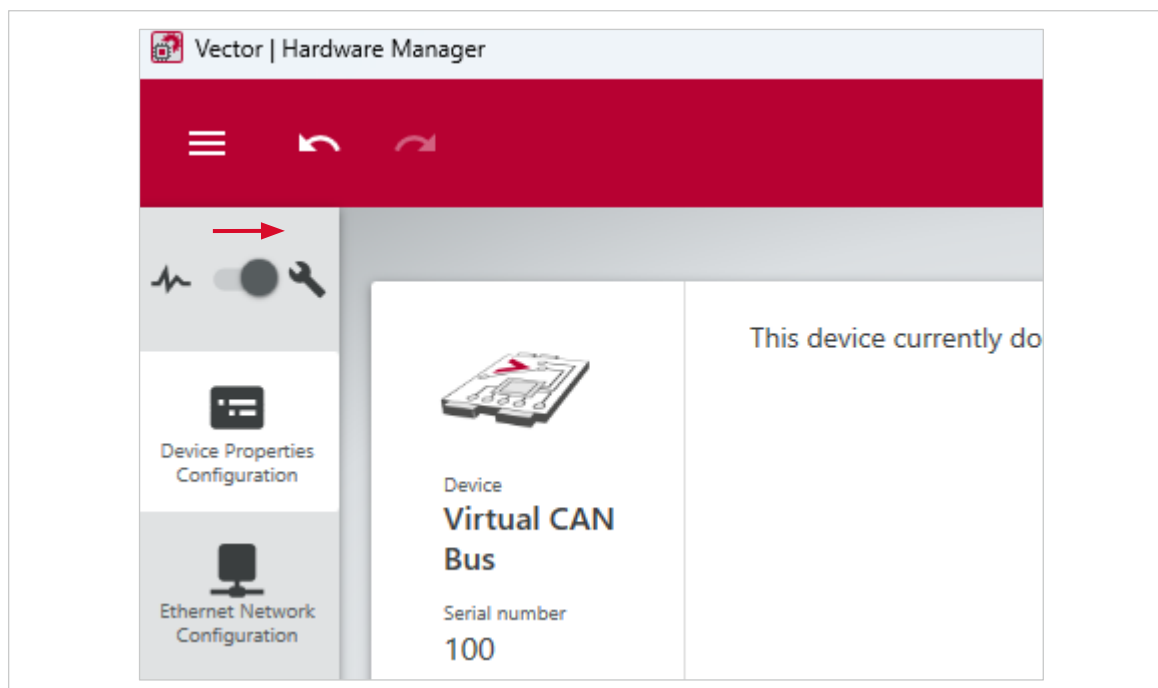
Windows のスタートメニューから Vector Hardware Manager を起動します。



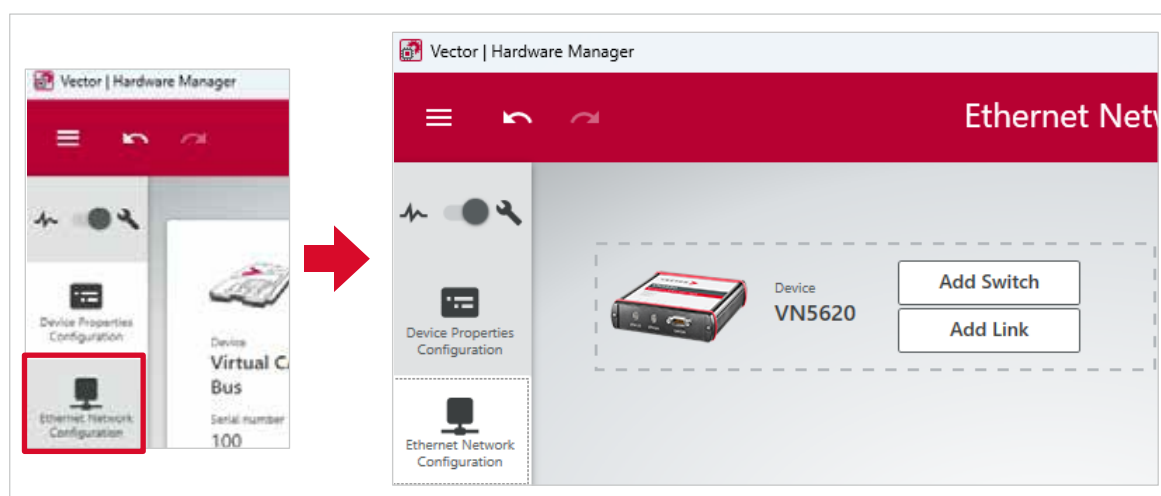
Vector Hardware Manager の起動右側のメニューに Vector Hardware Manager が認識しているインターフェイスが表示されるので、対象のインターフェイスが PC に接続されていることを確認します。接続されていない場合は、5.2.1 章の手順で接続を行います。



インターフェイスとの接続が確認できたら、画面左上のスイッチバーを設定ビュー（スパナのアイコン）に合わせます。

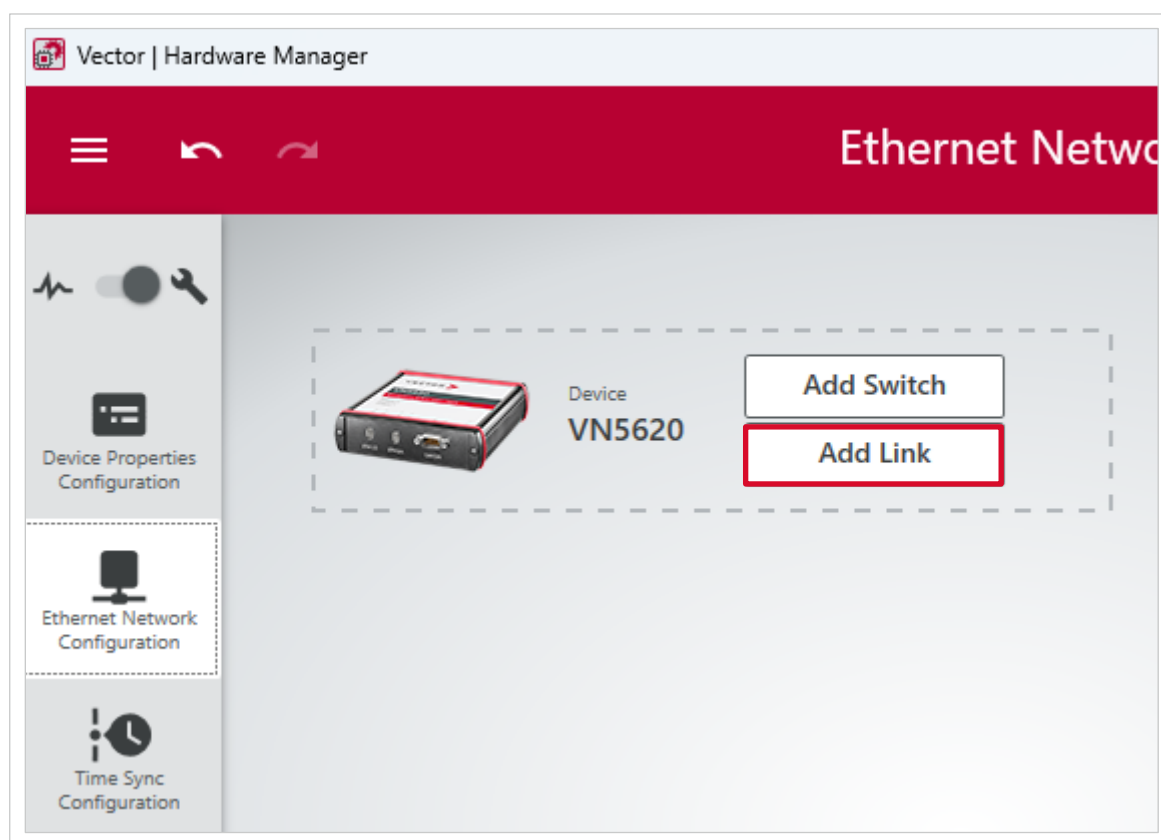


画面左側のメニューより、"Ethernet Network Configuration"を選択します。VN5000 インターフェイスのネットワーク設定はこの画面上で行います。



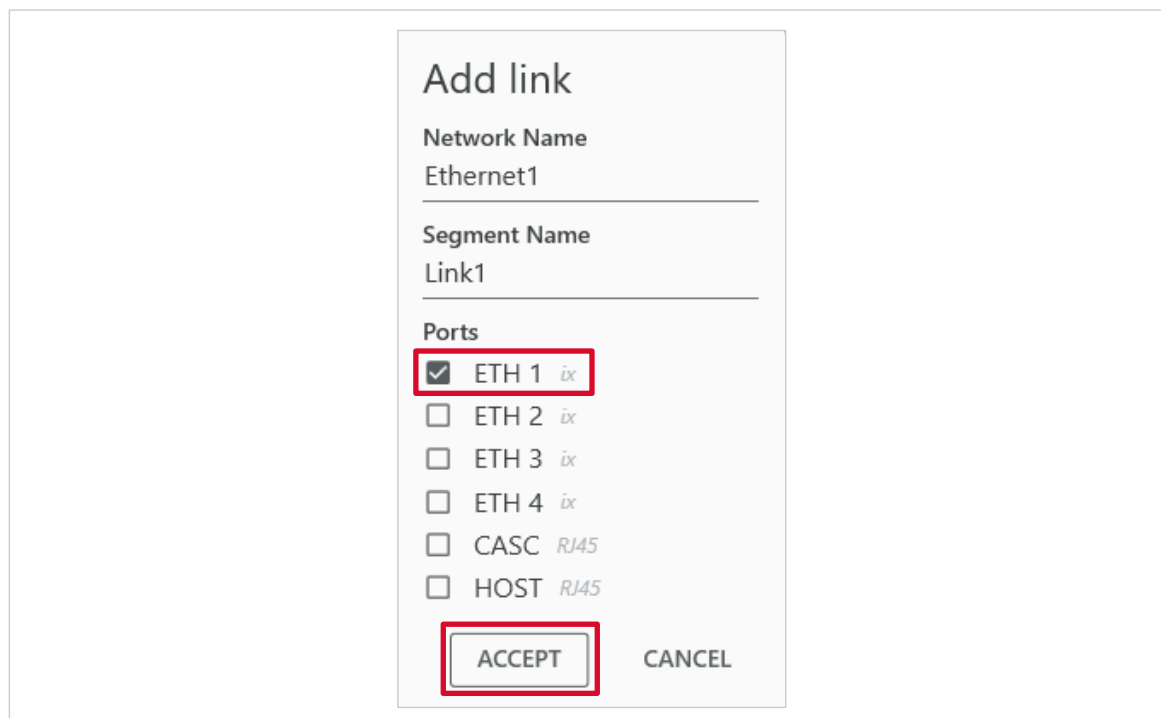
それでは2つのネットワークに、仮想 ECU と物理ポートの接続先を1対1で通信する機能と、物理ポート P1、P2、仮想ポート A、B を配置していきます。

はじめに、"Add Link"ボタンをクリックします。



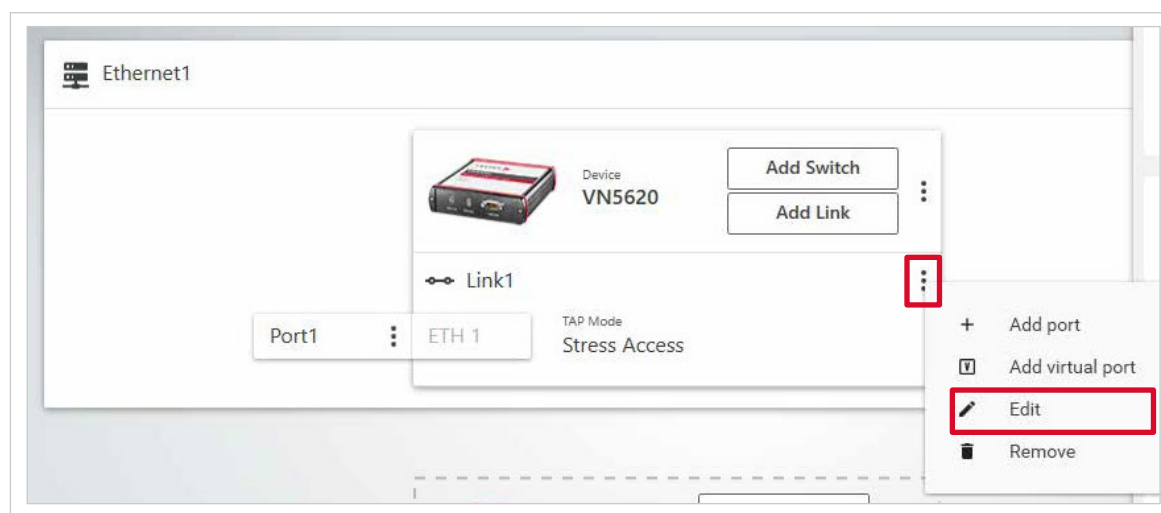
表示された window にて、ネットワーク名称が"Ethernet1"であることを確認し、任意の車載 Ethernet ポート 1 つを選択し、"ACCEPT"をクリックします。

なお、Ethernet1 以外のネットワーク名称でも設定はできますが、その場合は後述の手順において CANoe へのネットワークの割当を手動で行う必要があります。

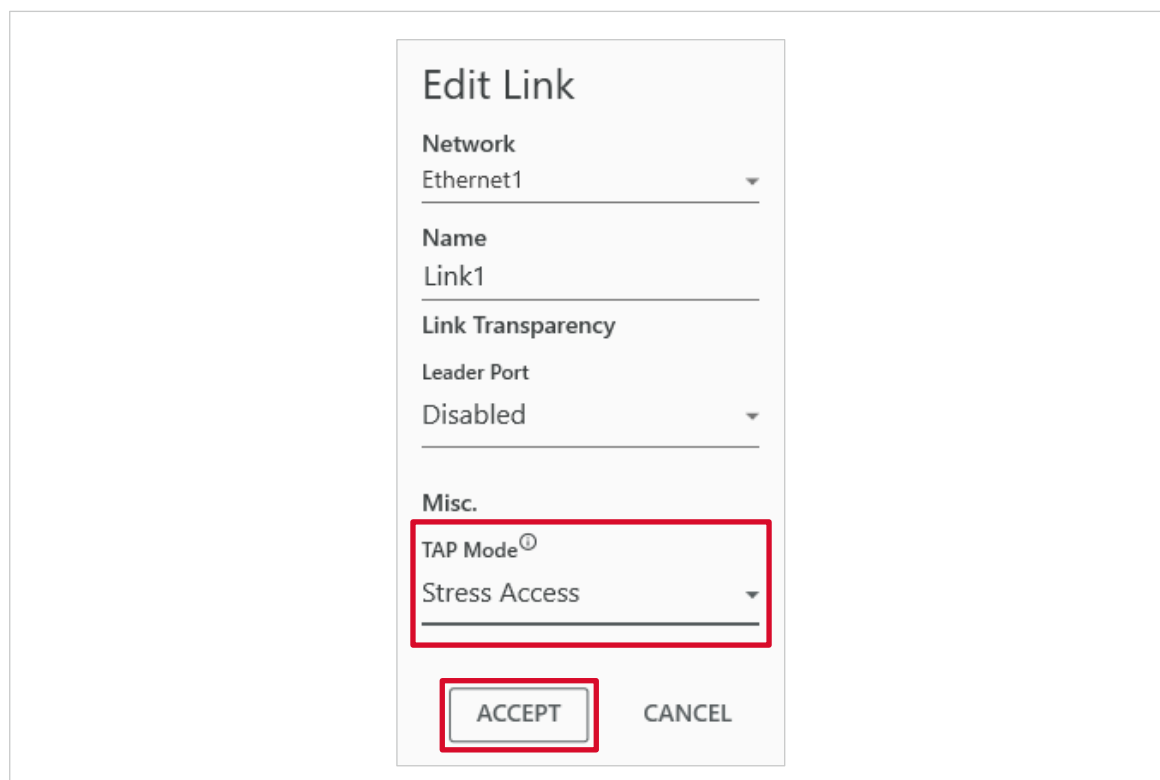


これで、Link セグメント"Link1"および物理ポート"Port1"をもつネットワーク"Ethernet1"が設定されます。

続いて Link セグメントを編集するため、Link1 の "⋮" アイコンをクリックし、"Edit"を選択します。



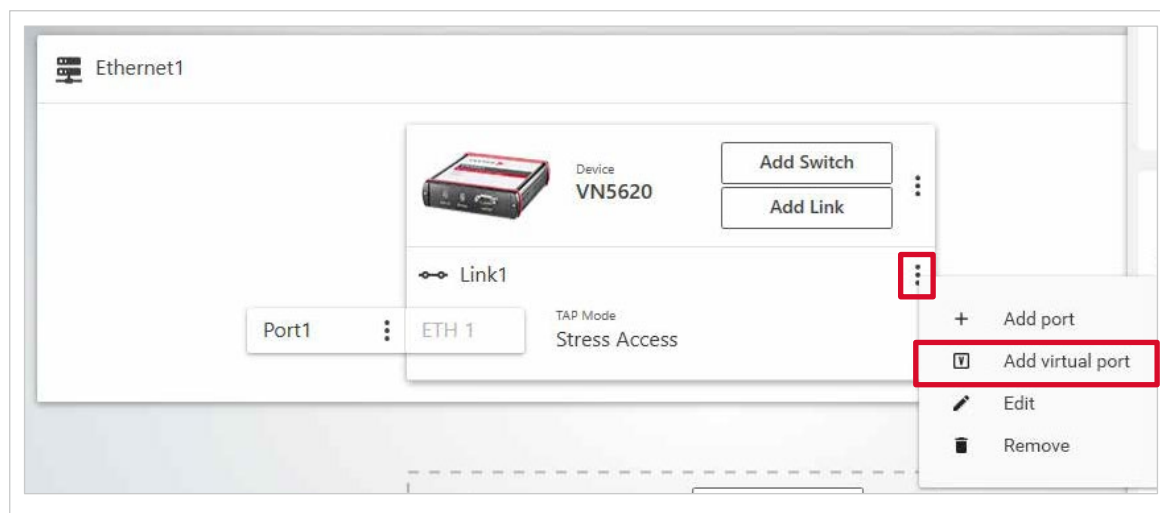
表示された window にて、TAP Mode を"Stress Access"に設定して、"ACCEPT"をクリックします。



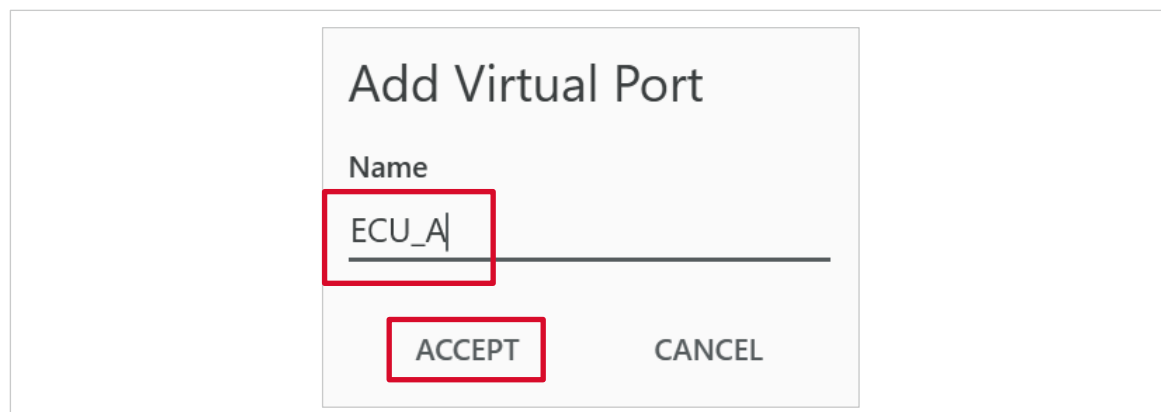
これで、Link セグメントの設定が完了しました。

続いて、Link セグメントに仮想ポートを配置します。Link1 の "⋮"アイコンから、"Add virtual port"をクリックします。

ただし、仮想ポートは後述「6.4.4 章 仮想ノードの作成」にて動的に設定されるため、必ずしも設定を行う必要はありません。意図的に作成したい場合のみ設定することを推奨します。



表示された window にて、仮想ポート名を"ECU_A"に設定し、"ACCEPT"をクリックします。



これで、1つのネットワークに1対1で通信する機能が配置できました。

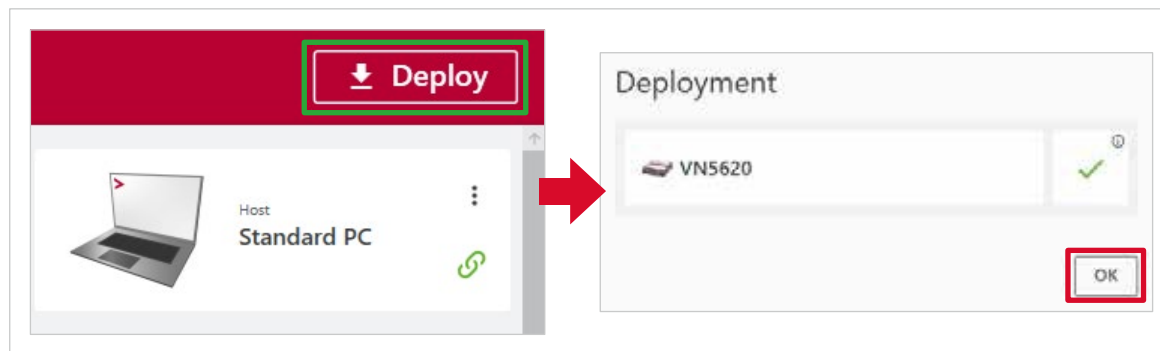


続いて同様の要領で、2 つ目のネットワーク、Link セグメント、物理ポート、仮想ポート (ECU_B) を追加します。



設定が完了したら、画面右上の"Deploy"ボタンをクリックし、下図のダイアログが表示されることを確認し"OK"をクリックします。

これで Ethernet インターフェイスの設定は完了です。



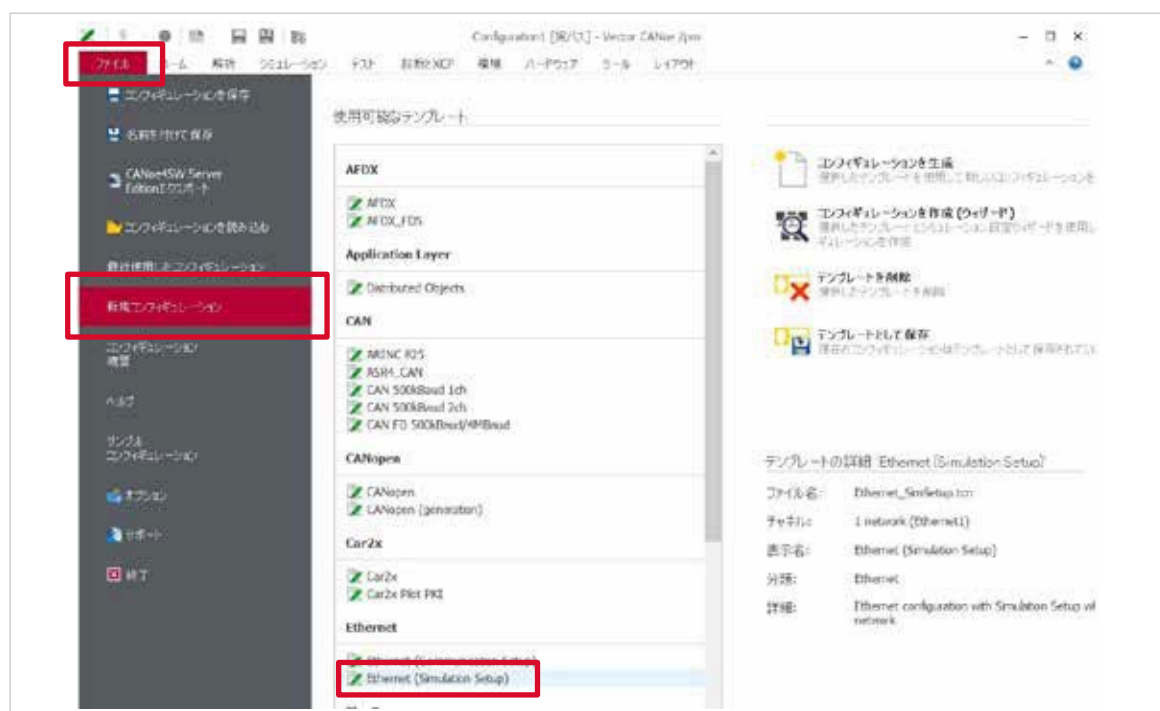
6.4 CANoe のコンフィギュレーションの作成

続いて CANoe のコンフィギュレーションを作成します。

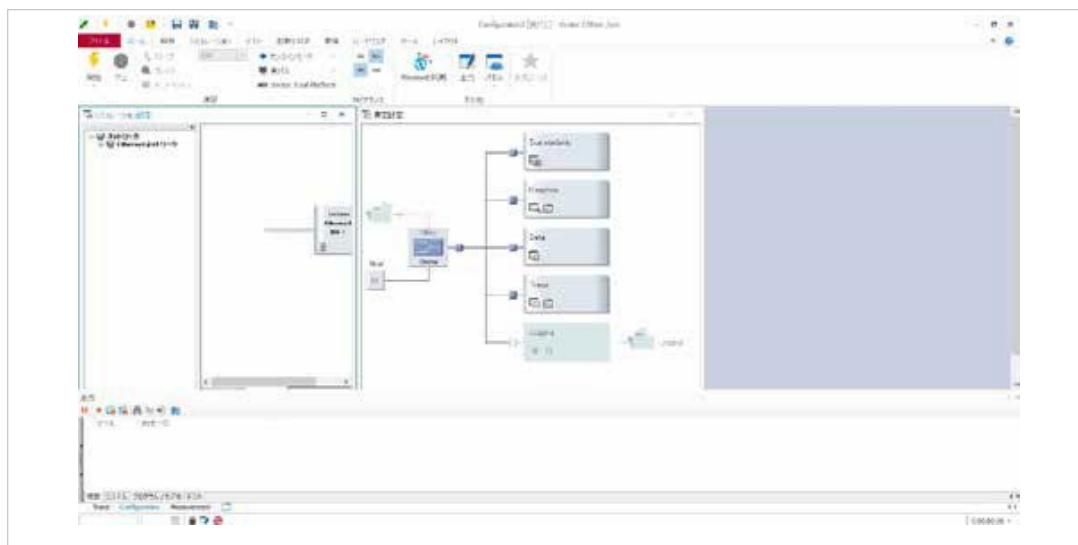
6.4.1 新規コンフィギュレーションの作成

はじめに CANoe の新規コンフィギュレーションを作成します。

ファイルタブの"新規コンフィギュレーション"をクリックし、使用可能なテンプレートから"Ethernet (Simulation Setup)"をダブルクリックします。



Ethernet 用の新規コンフィギュレーションが表示されます。

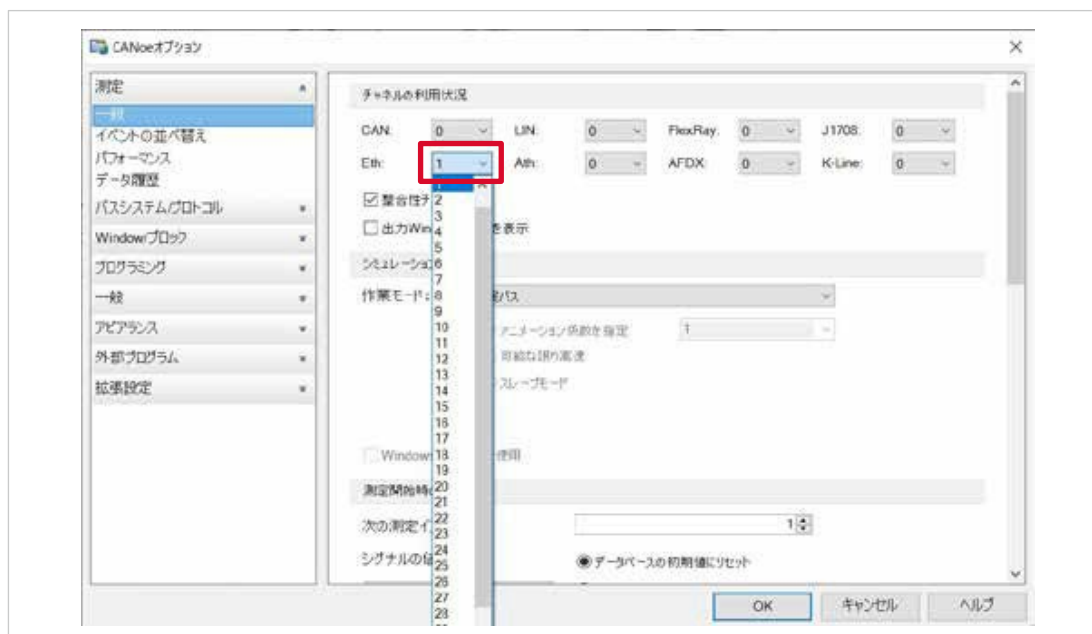


6.4.2 Ethernet チャンネルの設定

CANoe のオプションアイコンをクリックしてオプションメニューを表示します。

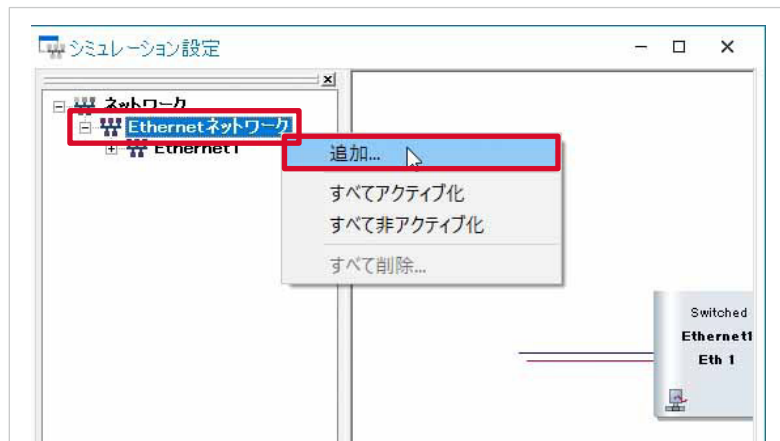


左ツリーの「測定」-「一般」から、チャンネルの利用状況が表示されますので、Eth チャンネルの数を 1 から 2 に変更します。

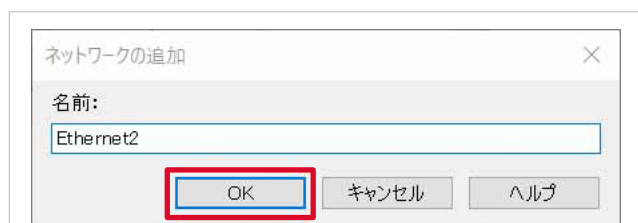


6.4.3 ネットワークの作成

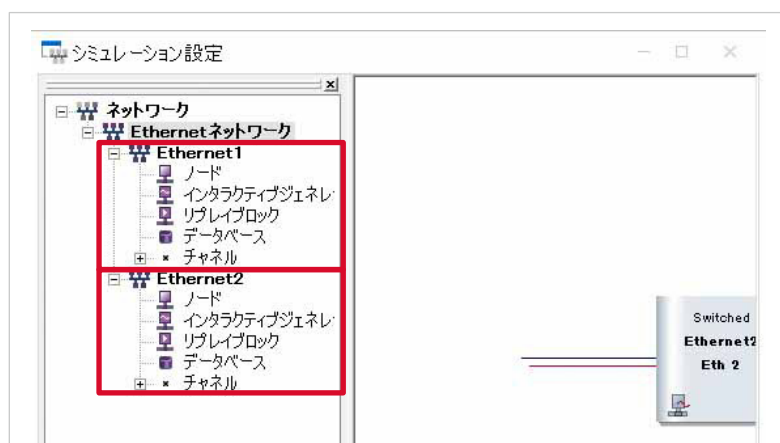
シミュレーション設定 Window の左ツリーの "Ethernet ネットワーク" を右クリックして、コンテキストメニューの "追加" をクリックします。



追加するネットワーク名を Ethernet2 にして、"OK" ボタンを押します。



これで CANoe のコンフィギュレーションに2つの Ethernet ネットワークが作成されました。

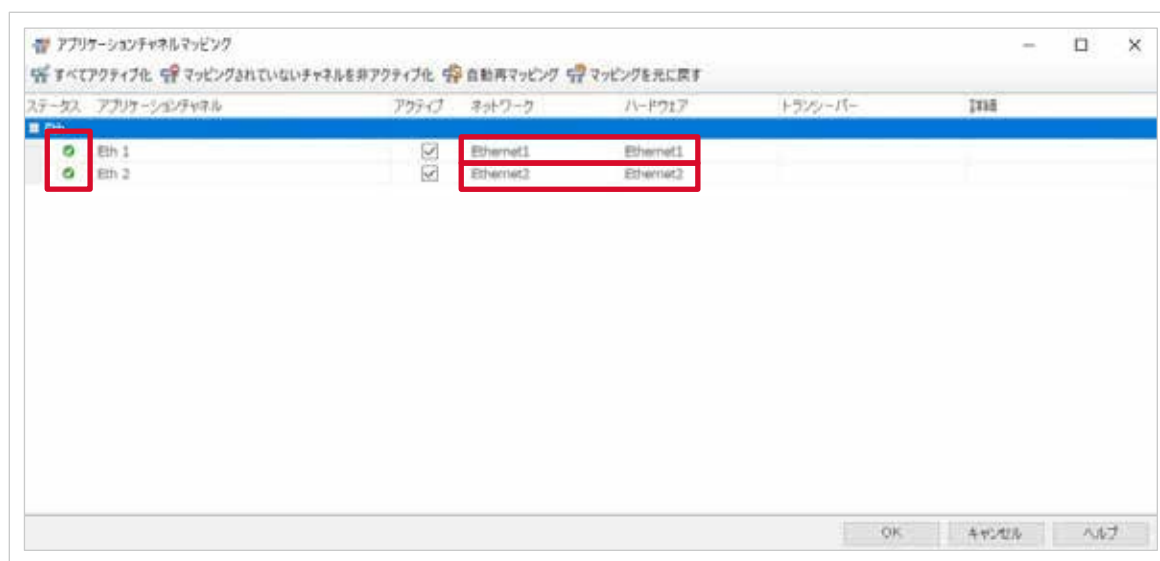


続いて、作成されたネットワークに、Vector Hardware Manager で設定したネットワークを割り当てます。
ハードウェアタブより"チャンネルマッピング"をクリックします。



チャンネルマッピングウィンドウが表示されます。6. 3 章の Vector Hardware Manager にて設定したインターフェイスの 2 つのネットワーク名称が "Ethernet1"、"Ethernet2" である場合 (CANoe で設定したネットワーク名称と同一である場合)、CANoe で設定したネットワークに対し、インターフェイスで設定したネットワークインターフェイスが自動的に割り当てられます。

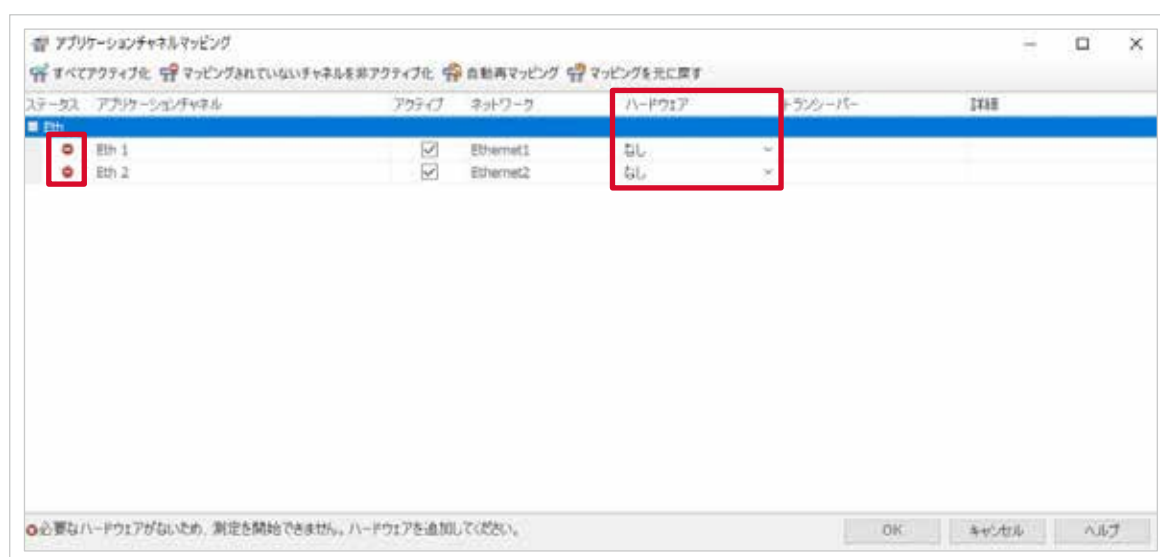
正しく割り当てがされている場合、ステータス欄が緑色のチェックアイコンにて表示されます。



一方で、CANoe で設定したネットワーク名称とインターフェイスのネットワーク名称が不一致である場合は自動的に割り当てが行われず、手動で割り当てを行う必要があります。以下は、Vector Hardware Manager にて、インターフェイスのネットワーク名称を"Network1"、"Network2"に設定した際の例です。

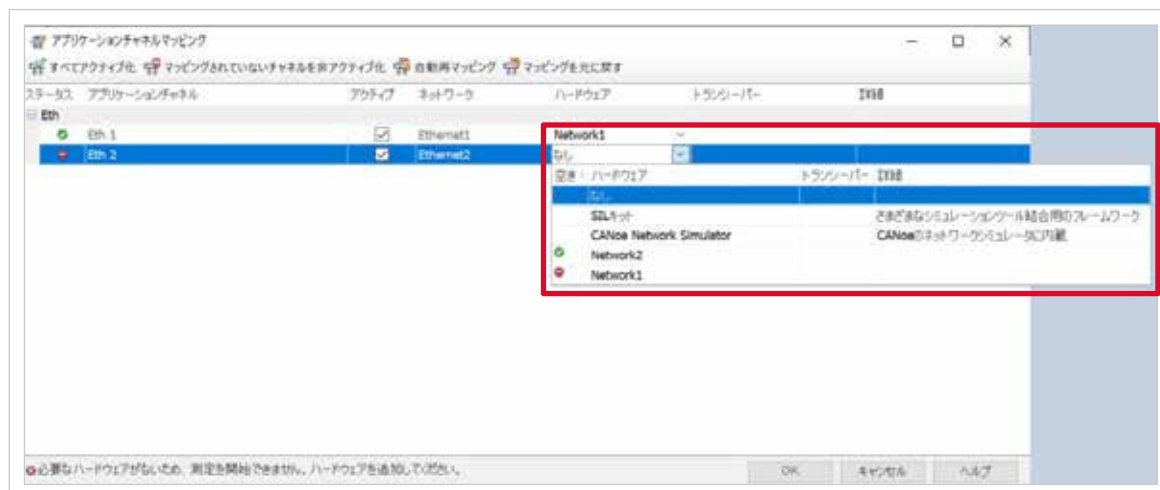


インターフェイスがこのように設定された状態で CANoe のチャンネルマッピングを開くと、下図のようにネットワークハードウェアの割り当てが"なし"となり、ステータスも赤色表示となります。



この状態から各ネットワークハードウェアのプルダウンメニューより設定したネットワーク

(下図の例では"Network1"、"Network2")を選択し"OK"をクリックすることで、ネットワークの割り当てが完了します。



6.4.4 仮想ノードの作成

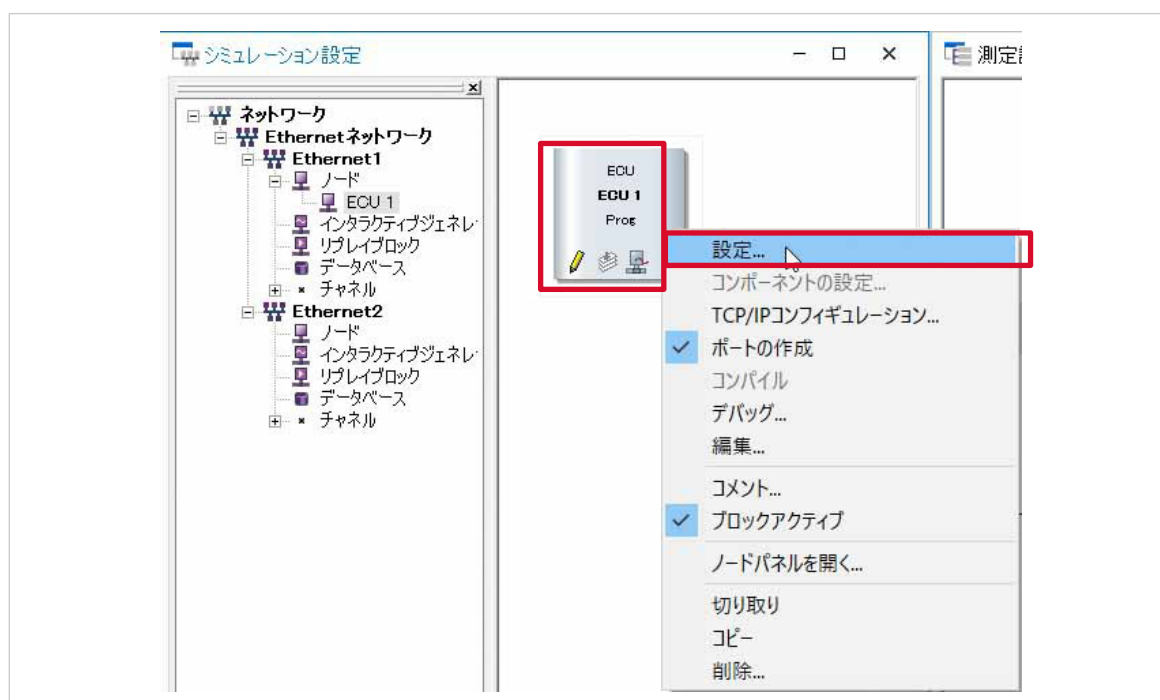
次に 2 つのネットワークにそれぞれ仮想 ECU を作成します。

シミュレーション設定 Window の左ツリーの Ethernet1 をクリックして、右 Window に表示されている青と赤の通信ライン上で右クリックをします。

コンテキストメニューの"ネットワークノードを挿入"をクリックして仮想 ECU を作成します。



作成した仮想 ECU 上で右クリックをして、コンテキストメニューの"設定"をクリックします。



仮想ノードの名前を「ECU_A」に変更して、「OK」ボタンを押して画面を閉じます。



Ethernet2 ネットワークも同様に仮想 ECU を作成します。仮想 ECU の名前は「ECU_B」に設定します。



これで仮想 ECU を使用する準備ができました。

6.4.4.1 TCP/IP スタックの設定

CANoe では仮想 ECU に TCP/IP スタックを使用できます。

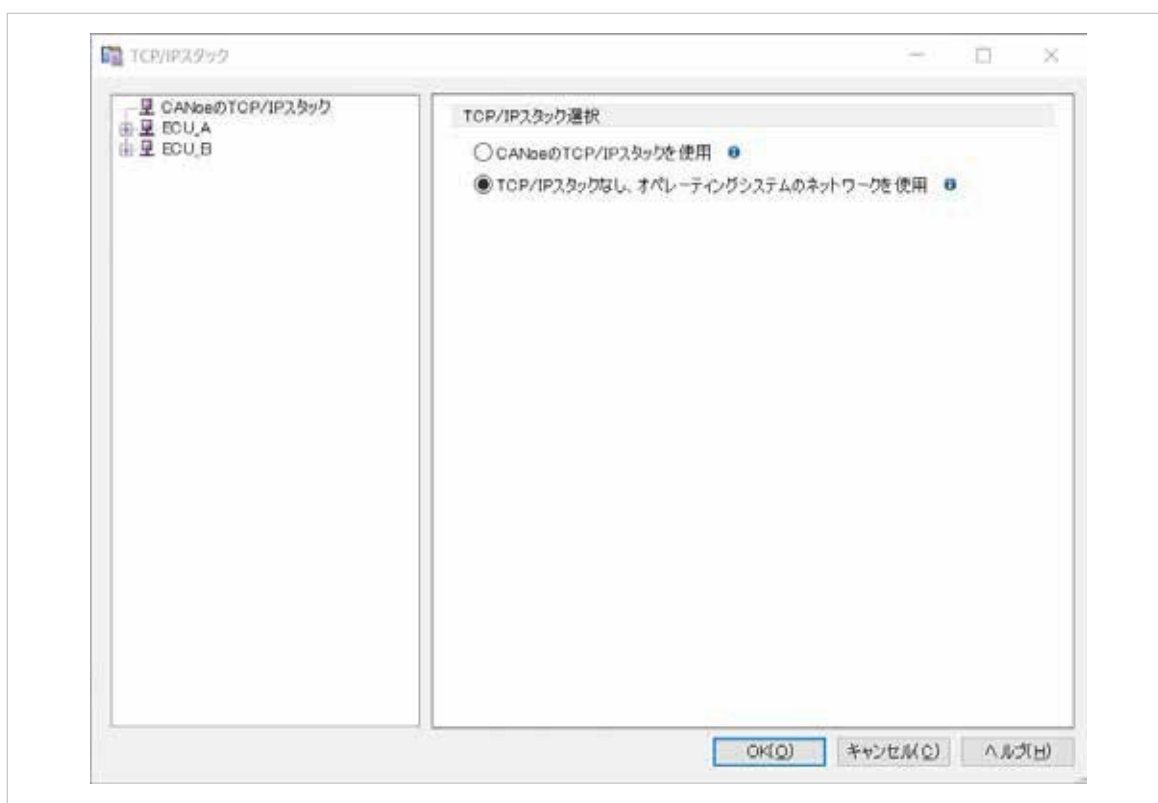
各仮想ノードに MAC アドレスや IP アドレス、VLAN などを設定できます。

ここでは UDP パケットを送受信する仮想 ECU A、B に IP アドレスを設定してみましょう。TCP/IP スタックは次のように設定します。

CANoe のリボンメニューの"TCP/IP スタック"をクリックします。

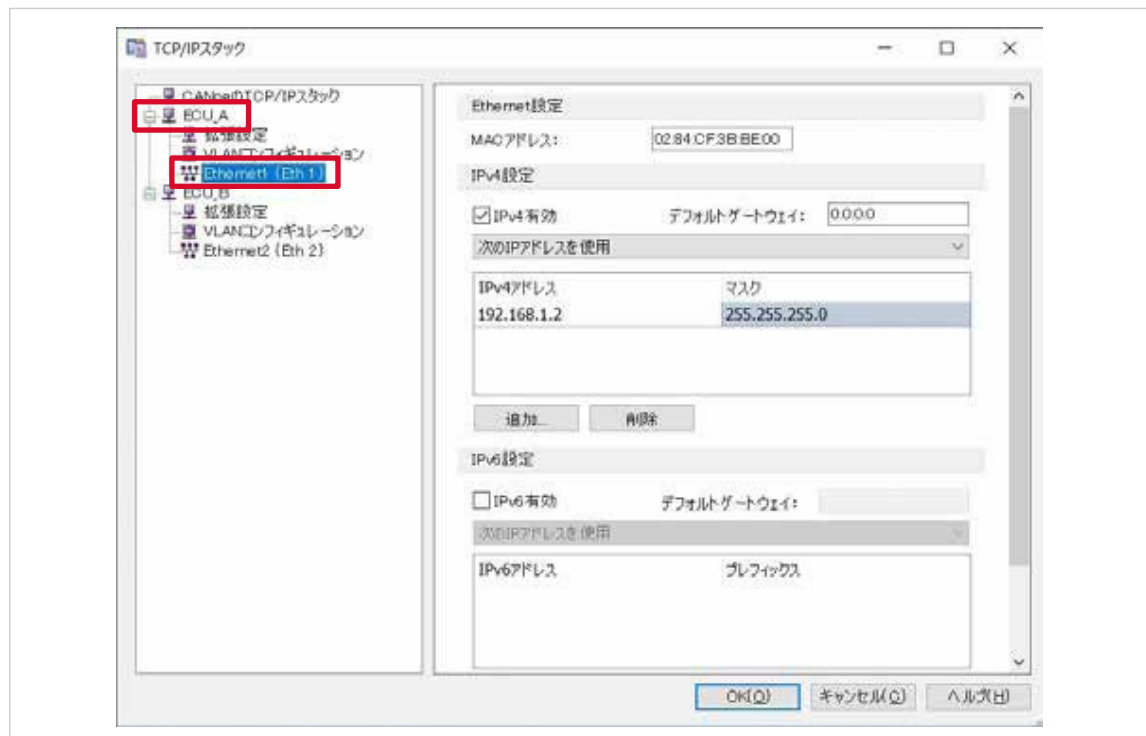


TCP/IP スタック Window が表示されます。



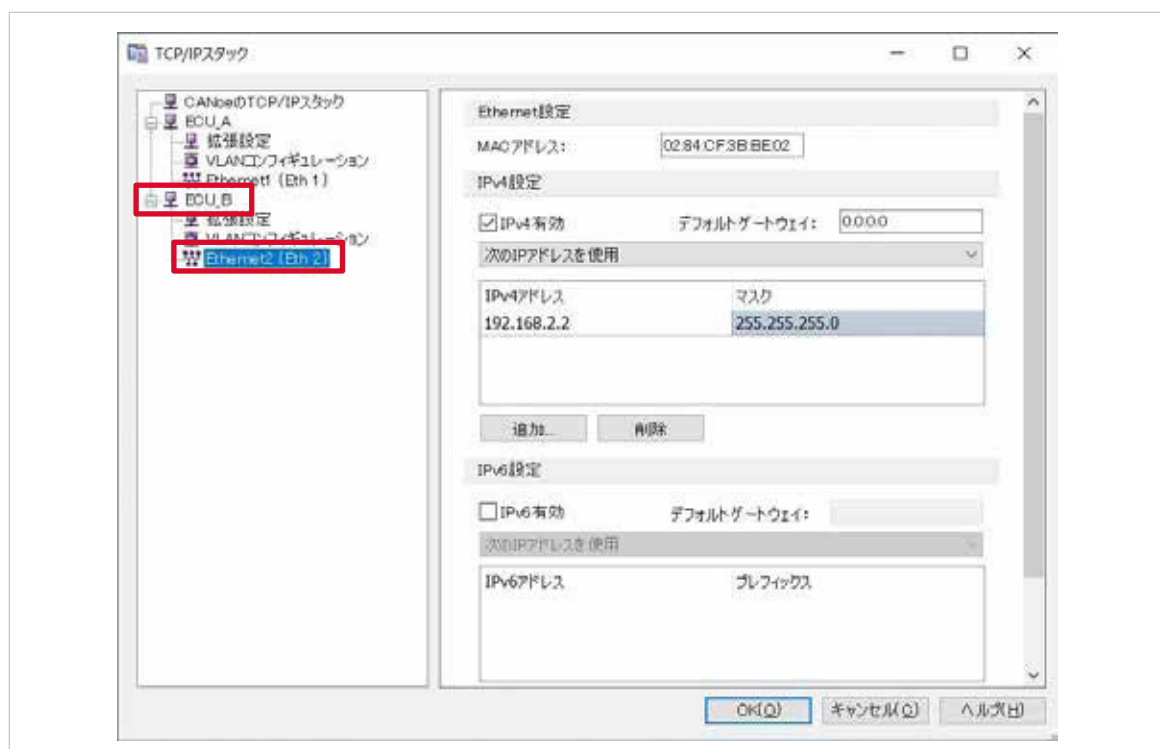
左側ツリーの ECU_A、ECU_B の「+」マークをクリックして展開します。

ECU_A の Ethernet1(Eth1)をクリックすると、右側 Window に設定画面が表示されます。ECU_A の IP アドレスは 192.168.1.2 となっています。今回は変更せずそのままとします。

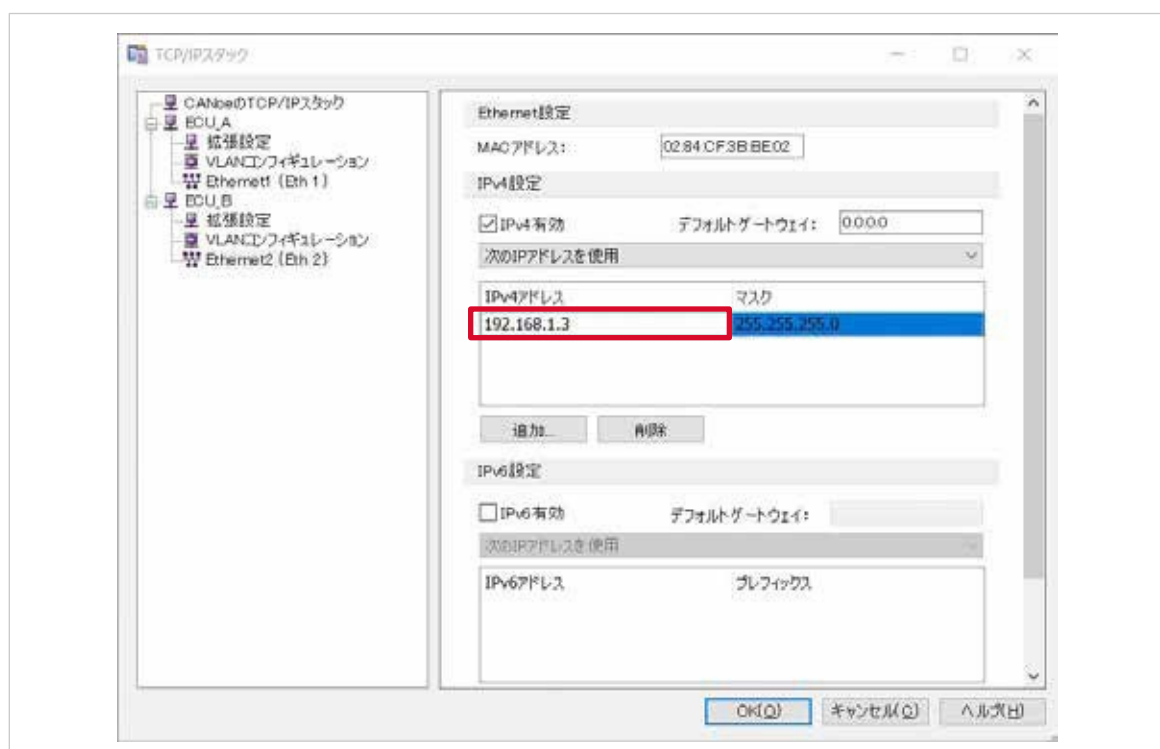


続いて ECU_B を設定します。

左ツリーの ECU_B の Ethernet2(Eth2)をクリックします。



ECU_B の IPv4 アドレスを以下のように 192.168.1.3 に変更してみましょう。



変更が終わったら"OK"ボタンを押して画面を閉じます。

6.4.4.2 仮想ノードの CAPL プログラミング

それでは仮想 ECU の振る舞いを模擬する CAPL プログラミングを記述してみましょう。

CAPL は CANoe 専用のプログラミング言語で C 言語によく似た言語です。そのため、C 言語をご存じの方は、あまり違和感なく使用可能かと思います。

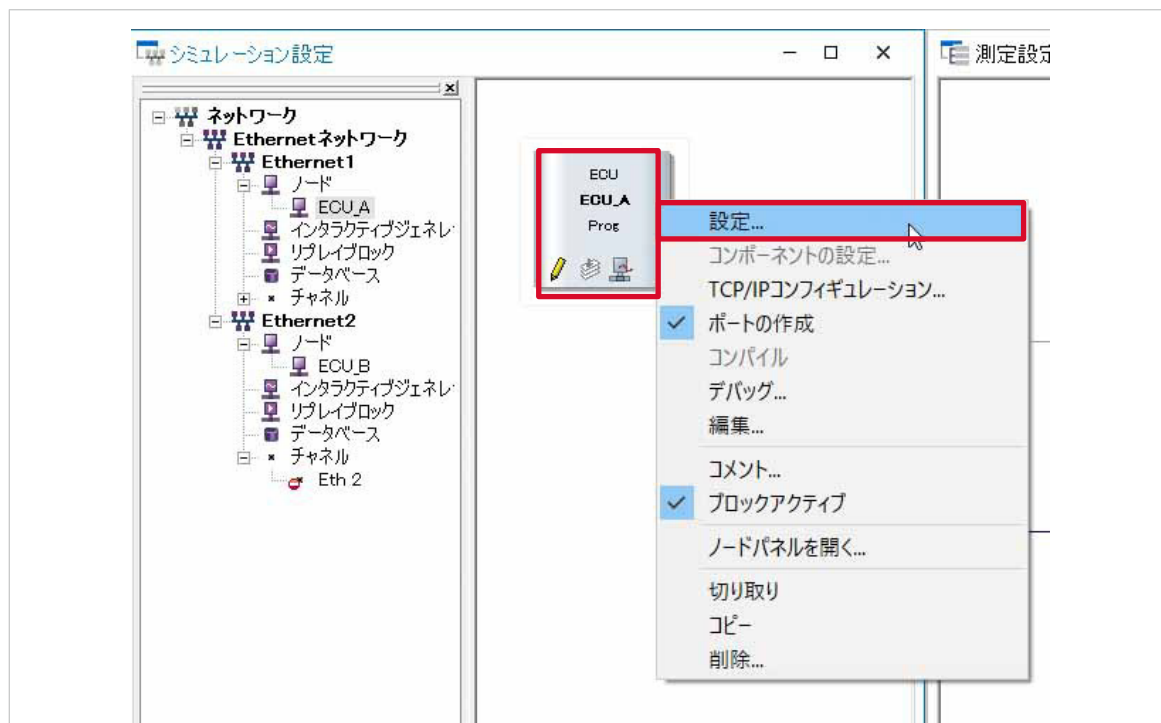
今回は 2 つのパターンで UDP パケットを送信するプログラムを作成します。1 つは TCP/ IP スタックを使ってソケット通信を行う方法、もう 1 つは Ethernet パケットを直接作成する方法です。

前者は CANoe の TCP/IP スタック機能を使って送受信しますが、後者は直接的に Ethernet インターフェイスからパケットを出力します。

それでは、はじめに仮想 ECU に割り当てる CAPL プログラムファイルを作成します。

6.4.4.2.1 CAPL プログラムファイルの作成

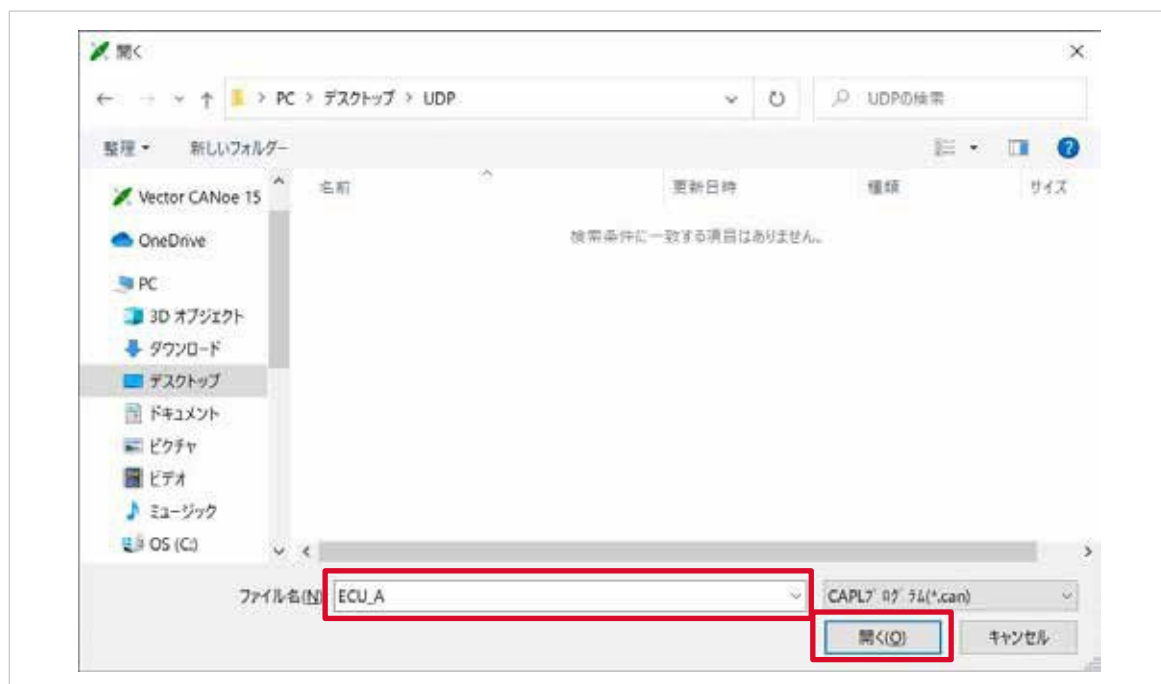
シミュレーション設定 Window の仮想ノードを右クリックして、コンテキストメニューから"設定"を選択します。



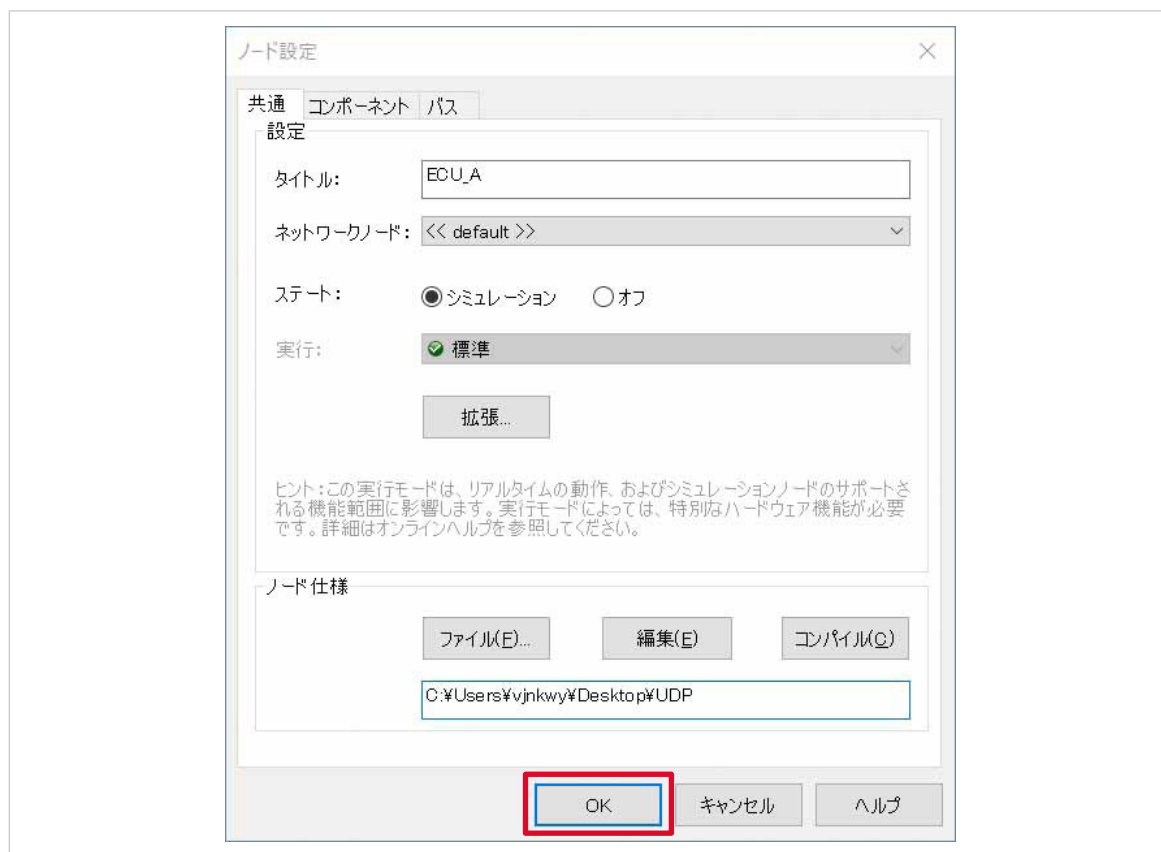
ノード設定 Window の下部の"ファイル"ボタンをクリックします。



ECU_A の CAPL プログラムファイル名を入力し、"開く"をクリックします。



"OK"ボタンを押して画面を閉じます。

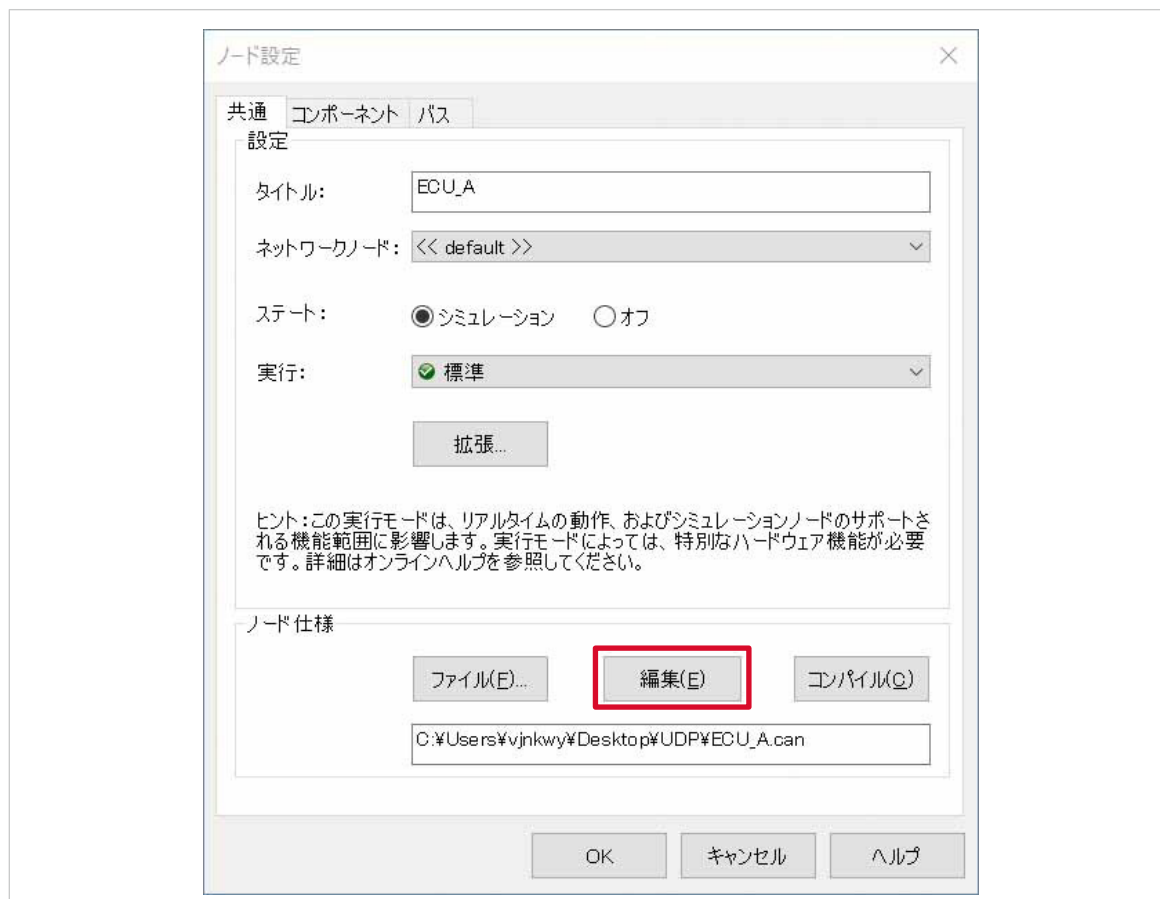


ECU_B も同様に CAPL ファイルを作成します。

6.4.4.2.2 CAPL プログラムファイルの編集

CAPL プログラムファイルを編集するには以下のどちらかで編集できます。

1. 仮想ノードを右クリックし、コンテキストメニューの"設定"からノード設定 Window を開きます。
下部の"編集"ボタンをクリックして、CAPL ブラウザを起動します。



2. 仮想 ECU 上にある編集アイコンをクリックして、CAPL ブラウザを起動します。



6.4.4.2.3 UDP API で UDP パケットを作成しよう

それでは CAPL ブラウザを起動して CAPL プログラムを作成しましょう。はじめに UDP API を使ってソケット通信プログラムを記述します。

ECU_A の送信プログラムを以下の手順で記述します。

1. ECU_A に UDP ソケットを生成します。
2. 生成した UDP ソケットに宛先 (ECU_B) の IP アドレスとポート番号を設定します。
3. キーボード"a"を押した時に、UDP パケットを送信します。

実装方法は以下のとおりです。

```
variables
{
  UdpSocket gSocket; //UDP ソケットの定義
}

on start
{
  gSocket = UdpSocket::Open( IP_Endpoint(0.0.0.0:40001) ); //UDP ソケットの生成
}

on Key 'a'
{
  gSocket.SendTo( IP_Endpoint(192.168.1.3:40002), "Hello", 5 );
}

on preStop
{
  gSocket.Close(); //UDP ソケットのクローズ
}
```

次に ECU_B の受信プログラムを以下の手順で記述します。ECU_B の受信プログラムを以下の手順で記述します。1. ECU_B に UDP ソケットを生成します。

UDP データを受信した場合に呼び出されるイベントを準備します。

UDP データを受信した時に呼び出されるイベントの処理を記述します。

実装方法は以下のとおりです。

```
variables
{
    UdpSocket gSocket; // UDP ソケットの定義
    char gRxBuffer[1472]; // 受信バッファ用変数
}

on start
{
    gSocket = UdpSocket::Open( IP_Endpoint(0.0.0.0:40002) ); //UDP ソケットの生成
    gSocket.ReceiveFrom( gRxBuffer, elcount( gRxBuffer) ); //受信する UDP データを取得
}

on preStop
{
    gSocket.Close(); // UDP ソケットのクローズ
}

OnUdpReceiveFrom( UdpSocket socket, long result, IP_Endpoint remoteEndpoint, char
buffer[], dword size)
{
    if (result == 0)
    {
        write( "Received: %s", buffer );
        gSocket.ReceiveFrom( gRxBuffer, elcount( gRxBuffer) );//次回受信する UDP データを取得
    }
}
```

6.4.4.2.4 Ethernet Packet オブジェクトで UDP パケットを作ってみよう

UDP パケットを送信する 2 つめの方法として、Ethernet Packet オブジェクトを使用して UDP パケットを生成します。

ECU_A の送信プログラムを以下の手順で記述します。

1. キーボード"b"を押した時に、2~8 の処理を実行します。
2. EthernetPacket オブジェクトを用いて、Ethernet フレームオブジェクトを生成します。3. 生成した Ethernet オブジェクトを UDP パケットに初期化します。
3. 初期化した UDP パケットに必要な情報を設定していきます。
4. Ethernet インターフェイスのポートや、MAC アドレス、IP アドレス、ポート番号などを設定します。
5. ペイロードに値を設定します。
6. 最後にチェックサムとパケット長の自動計算処理を記述します。
7. UDP パケットを送信します。

実装方法は以下のとおりです。

```
on key 'b'
{
ethernetPacket txPacket; //Ethernet フレームオブジェクト
int i;

txPacket.udp.Init(); //UDP パケットの初期化
txPacket.hwPort = ethernetport::Ethernet1::ECU_A; //Ethernet ポート
txPacket.source = EthGetMacAddressAsNumber( "20:00:00:00:00:01" ); //送信元 MAC アドレス
txPacket.destination = EthGetMacAddressAsNumber( "FF:FF:FF:FF:FF:FF" ); //宛先 MAC アドレス
txPacket.ipv4.source.ParseAddress( "192.168.1.2" ); //送信元 IP アドレス
txPacket.ipv4.destination.ParseAddress( "192.168.1.3" ); //宛先 IP アドレス
txPacket.udp.source = 40001; //送信元ポート番号
txPacket.udp.destination = 40002; //宛先ポート番号
txPacket.udp.ResizeData( 10 ); //UDP のペイロードサイズの設定
for( i = 0; i < 10; i++ )
{
txPacket.udp.Byte(i) = i; //UDP ペイロードに値を代入
}

txPacket.CompletePacket(); //チェックサム計算処理
output( txPacket ); //フレームの送信
}
```

次に ECU_B の受信プログラムを以下の手順で記述します。

ECU_B の受信プログラムを以下の手順で記述します。

1. Ethernet パケットを送受信した時に呼び出されるイベントを記述します。
2. 送受信した Ethernet パケットに、UDP パケットが含まれているかどうかを判定します。
3. UDP パケットが含まれている場合、this キーワード、プロトコルセクターを使用してアクセスします。

実装方法は以下のとおりです。

```
on ethernetPacket * //Ethernet パケット受信イベントの定義
{
  if (this.udp.IsAvailable())//UDP パケットが含まれているかの判定
  {
    write( "Packet has UDP protocol with port %d to %d!", this.udp.source, this.udp.destination );
  }
}
```

6.4.4.2.5 コンパイル

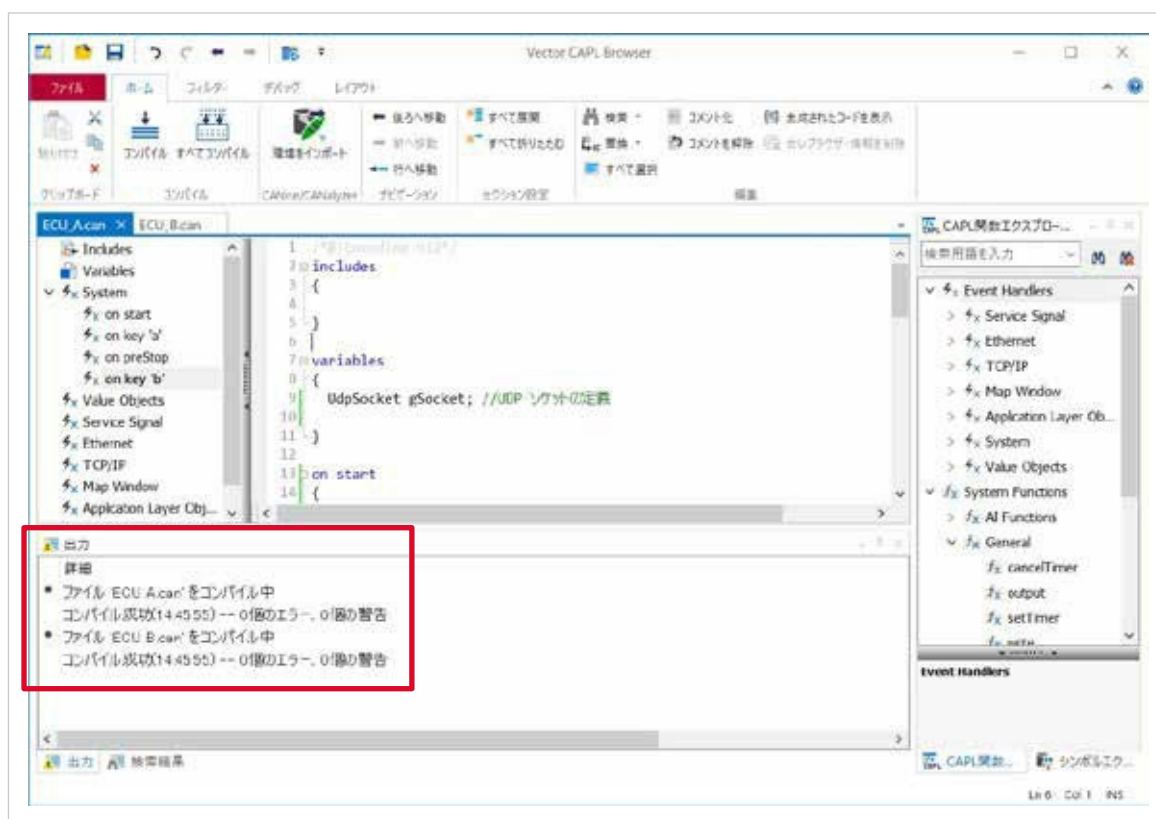
CAPL プログラムの記述が終わった後はコンパイルします。

CAPL ブラウザのリボンメニューのホームタブにある"すべてコンパイル" アイコンをクリックします。



コンパイルが終わると出力 Window に結果が表示されます。エラーがないことを確認できたら CAPL ブラウザを閉じます。

エラーが表示された場合は、該当の CAPL プログラムを見直しましょう。



6.4.5 ポート設定

最後に測定するポートを設定します。

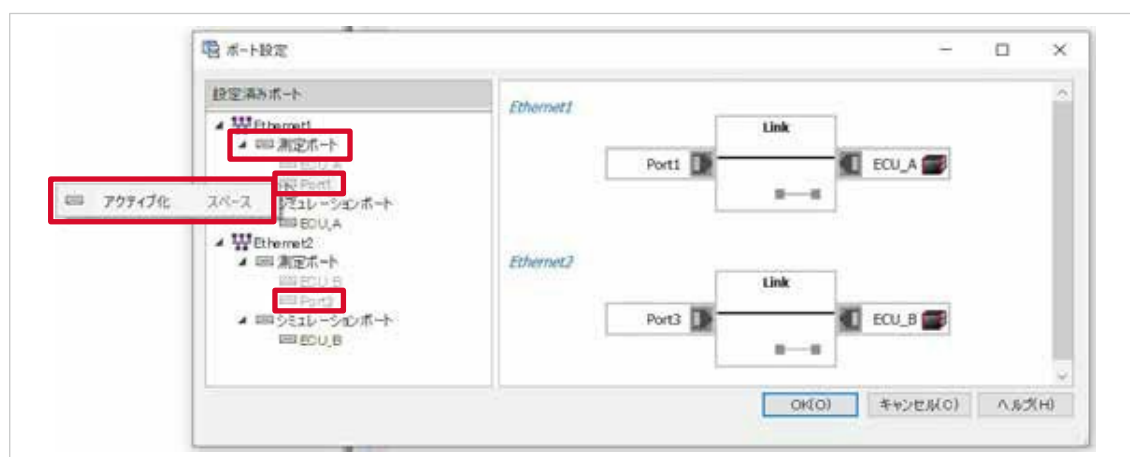
トレース Window などに表示するには測定ポートを設定する必要があります。

CANoe のリボンメニューのハードウェアから"ポート設定"をクリックします。

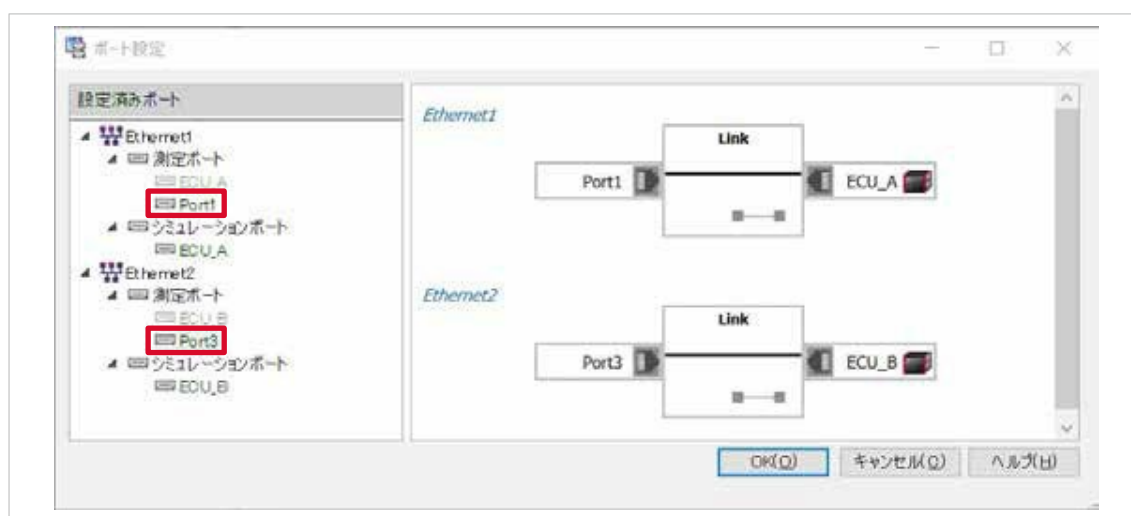


ポート設定 Window の左側の各 Ethernet ネットワークの測定ポートを設定します。今回は物理ポートである Port1、3 を測定ポートとして有効にします。

測定ポートの Port1 と Port3 上で右クリックして、コンテキストメニューからアクティブ化します。



アクティブ化された後は、ポート名が黒字で表示されます。



設定が完了したら"OK"ボタンを押して画面を閉じます。

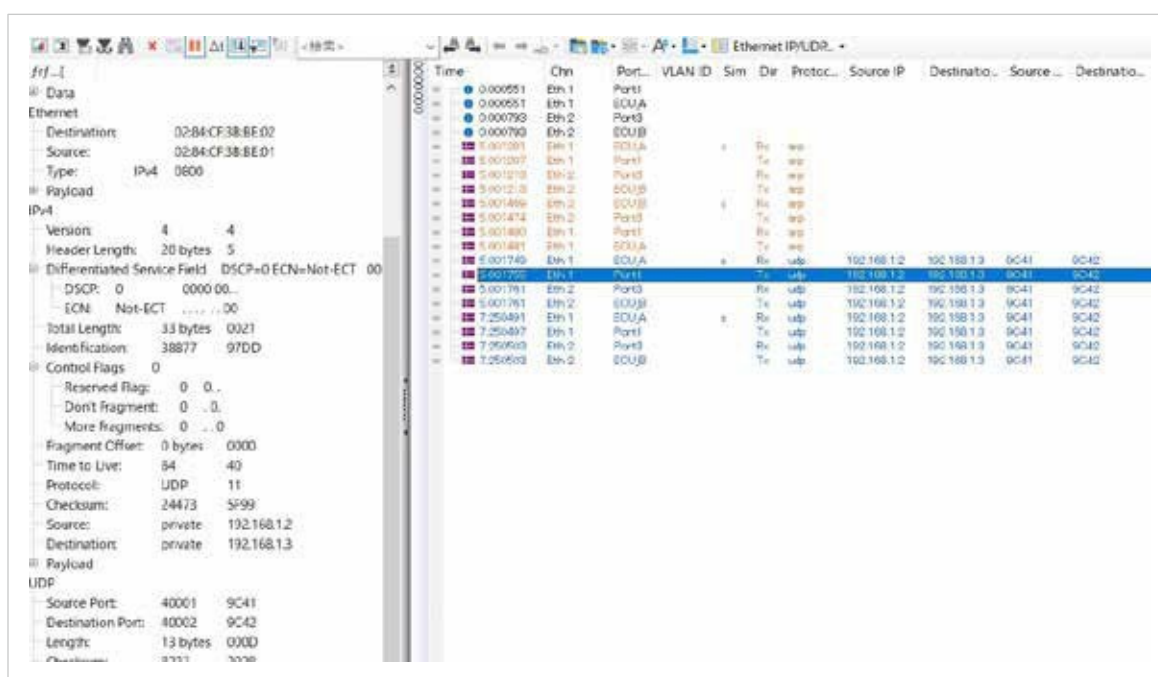
6.4.6 CANoe の開始

これで CANoe と Ethernet インターフェイスの準備ができました。それでは CANoe で測定開始してみましょう。
CANoe のリボンメニューのホームから"開始"アイコンをクリックします。



6.4.7 仮想ノードからパケット送信

次にキーボードの"a"、"b"を押して UDP パケットを送信してみましょう。正しく送信できれば、以下のようにトレース Window に表示されます。



実バスで Ethernet シミュレーションを行う環境の作成例は以上です。

あとがき

以上、CANoe.Ethernet で実現できることや、CANoe.Ethernet の使用例について解説しましたが、いかがでしたでしょうか。

本稿が CANoe.Ethernet のすべての機能を解説しているわけではありませんが、CANoe.Ethernet の基本的な機能はご理解いただけたと思います。

ベクターでは、本稿以外にも CANoe.Ethernet に関するマニュアルを各種取りそろえておりますので、是非そちらもご覧ください。

本稿で解説した基礎知識が、車載 Ethernet 開発に携わる、または今後携わっていくエンジニアの方々の一助となれば幸いです。

7. お問い合わせ窓口

技術関連のお問い合わせは、ベクターサポートまでお寄せください。

- ✓ サポートリクエスト(Web サイト経由のサポート問い合わせフォーム)
www.vector.com/jp/ja/support-downloads/support/

【お問い合わせ時のお願い】

お問い合わせの際、サポート開始前に製品のバージョン(例:CANape バージョン xx.x など)、シリアルナンバーなどをお伺いいたします。お手数ではございますが、事前にご確認のうえ、お問い合わせくださいますようお願い申し上げます。

【サポート対応について】

夜間／休日に頂いたお問い合わせは翌営業日の対応となります。また、内容によりご回答までにお時間を頂く場合がございます。なお、不具合が発生した際にご使用されていた関連プログラム一式のコピーを、検証のためご提供いただくようお願いする場合がございます。何卒ご理解ご協力のほど、よろしくお願い申し上げます。

- ✓ お見積もり／保守契約関連のお問い合わせ

営業部 TEL:(東京) 03-4586-1808 E-mail:sales@jp.vector.com

- ✓ トレーニングに関するお問い合わせ

トレーニング部 TEL:(東京) 03-4586-1816 E-mail:training@jp.vector.com

【ベクターのトレーニング】

ベクターでは、車載およびオープンネットワークに関連する技術や、その開発ツールの使用法など、エンジニアの皆さまに向けた専門トレーニングをご用意しています。トレーニングは、東京本社および名古屋支社にて行っております。受講のお申し込みについては下記 URL をご覧ください。定員になり次第締め切らせていただきますので、お早めにお申し込みください。

また、規模やご要望に応じて、お客様ご指定の場所でのオンサイト・トレーニングも実施しております。詳細は、ベクターのトレーニング部までお気軽にお問い合わせください。

<https://academy.vector.com/jp/ja/>



MORE INFORMATION

Visit our website for:

- > News
- > Products
- > Demo software
- > Support
- > Training and workshops
- > Contact addresses

vector.com