

Virtual Serial Ports ヘルプ

翻訳、製作 エレクトロデザイン株式会社

General description

VSPE の目的

仮想シリアルポートエミュレータ (VSPE) は、シリアルポートなどのようなものに対処するためのものです。今日では、シリアルポートはもはや広く使われていません。理由は簡単です：それはかなり古いからです。最新のハードウェアデバイスは、通常、高速 USB ポート経由でコンピュータに接続し、時には独自の UsbToCom ドライバをインストールします。シリアルポートは、ほとんどの新しいノートブックにはインストールされていません。

- RS232 または RS422 互換シリアルポートが必要な特定のデバイスがたくさんあります。VSPE は、開発者が実際のハードウェアをインストールしなくてもアプリケーションの作成とデバッグを支援します。
- 古いアプリケーションとの互換性。VSPE は、(TCP 接続、USB ポートなどの) 任意のソースからのデータストリームを、仮想シリアルポート経由でどのプログラムにも見えるようにするのに役立ちます。
- 同時に動作する多数の仮想シリアルポートを作成する。シリアルポート拡張ボードをインストールせずにそれを行うことができます。
- 多くのアプリケーションでシリアルポートを共有する。

VSPE 制限事項

- ボーレートエミュレーションは非常に粗いです
- シリアルポートの出力バッファがパススルー (実際にはエミュレートされていない)
- シリアルポートの出力タイムアウトは無視されます
- DataFlow コントロールはあまり高速ではありません (受信側の入力バッファサイズよりも実際に大きいパケットを送信しないようにしてください)

現段階では、主な目標は次のとおりです。

- •アプリケーションまたはデバイス間のバイナリデータ交換を提供する。
- •独自のソリューションを作成する。コネクタと DataSplitter デバイスは、そのようなソリューションの明白な例です。一度に COM ポートを作成し開くことができます。
- •簡単なユーザーインターフェイスを提供する。

主要な機能:

- 仮想デバイス：コネクタ（カーネルモード）
- 仮想デバイス：データスプリッタ（カーネルモード）
- 仮想デバイス：ペア（カーネルモード）
- カーネルモードデバイス：Mapper
- ユーザーモードデバイス：TcpServer
- ユーザーモードデバイス：TcpClient
- ユーザーモードデバイス：シリアルリダイレクタ
- ユーザーモードデバイス：UDP マネージャ
- ユーザーモードデバイス：ブリッジ
- データの監視と記録
- 組み込み HTTP サーバー
- x86 および x86_64 プロセッサアーキテクチャのサポート
- 組み込み Lua スクリプト言語
- ネイティブ言語開発者のための VSPE API（C/C++ヘッダと静的ライブラリ）
- VSPE API Python 開発者のための Python バインディング
- VSPE API COM インターフェイス

Product overview

VSPE デバイス

VSPE はすべてデバイスに関するものです。デバイスは、何らかの作業を行うエンティティです。たとえば、Tcp 接続を介してリモートコンピュータからのデータをリダイレクトする必要がある場合、そのような機能を提供する TcpClient デバイスを作成できます。VSPE のデバイスは、カーネルモードとユーザーモードの 2 つのカテゴリに分類されます。あなたは好きなようにそれらを組み合わせることができます。

カーネルモードデバイス

これらのデバイスは、VSPE カーネルドライバによって実装されています。VSPE の主な目標は仮想ポートを作成することでした。通常は仮想 COM ポートを作成しますが、その一部（Mapper など）は既存の COM ポートのリンクを別のリンクで置き換えるだけです（COM9 は COM1 として開くことができます）。コネクタ、スプリッタ、ペアなどの最も一般的なデバイスは、カーネルモードのデバイスです。VSPE が閉じられると、カーネルモードのデバイスは引き続き動作します（この動作を変更する場合は、設定ページを参照してください）。

ユーザモードデバイス

これらのデバイスは仮想ポートを作成しません。彼らはカーネルドライバとは関係がなく、ユーザーモードで実装されているので、開発/デバッグやデバッグが容易です。VSPE が閉じられると、すべてのユーザモードデバイスが破棄されます。ユーザーモードデバイスは、さまざまなソース (Tcp、UDP など) からのデータストリームをリダイレクトする必要がある場合や、別の VSPE デバイスなどでスパイする必要がある場合に非常に便利です。これらはカーネルモードデバイスと併用することができます。

スパイデバイス

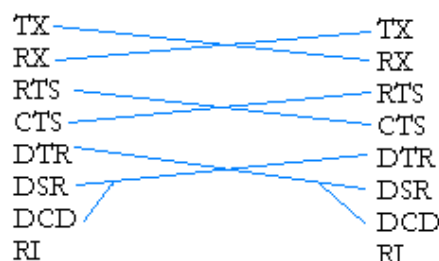
このデバイスは特別です。いくつかの VSPE デバイスをスパイして、傍受したデータを一部のデータストリームに送信することができます。これはソフトウェア開発者にとって非常に興味深い機能です。詳細については、Spy デバイスを参照してください。

VSPE とソフトウェア開発者

VSPE の主な目的は、ソフトウェア開発者の生活を楽にすることでした。あなたの仕事が COM ポートに関連している場合、あなたはアプリケーションをデバッグするのがいかに難しいかを知っています。入力データをエミュレートする必要があるため、ヌルモデムケーブルを使用するか、別のメカニズムを使用して入力データをエミュレートする必要があります。VSPE は、何か特別なことをしたい、あるいは VSPE GUI アプリケーションをまったく必要としない開発者にとって有用な API を公開しています。VSPE デバイスを作成/破壊する機能だけです。詳細は VSPE API を見てください。

Virtual serial ports pinout

Virtual serial ports have standard pinout



(painted in PaintBrush)

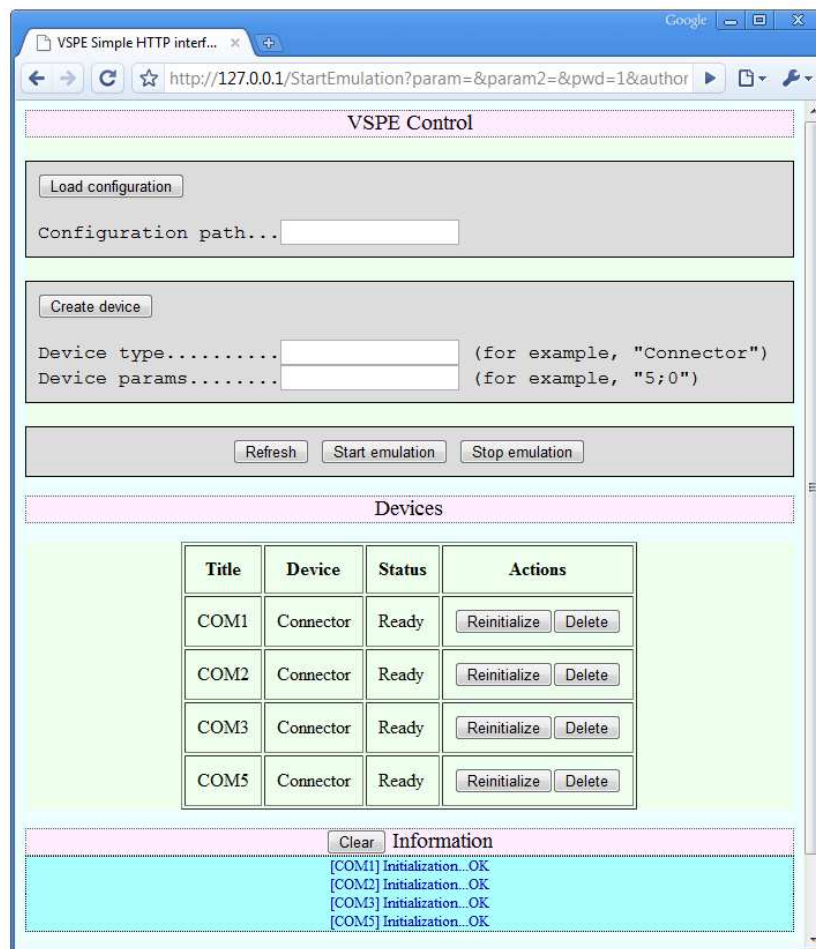
HTTP server

目的

Http サーバーは、VSPE 構成を制御し、リモートコンピュータ上のデバイスの状態を監視することを目的としています。

設定

http サーバーを設定するには、[HTTP サーバー]タブを参照してください。

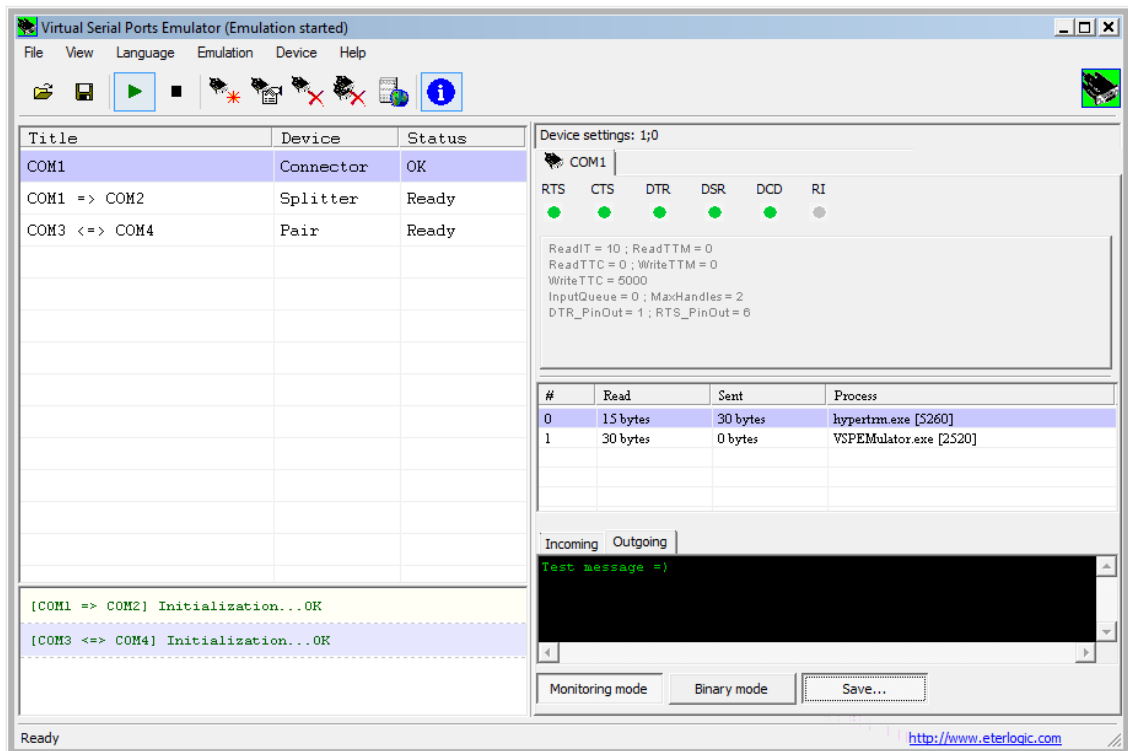


Http server in action

データのモニタ

目的

アプリケーションがどのデータを送受信しているかを知る必要がある状況はたくさんあります。上級ユーザー/開発者を支援するため、データ監視機能が VSPE に追加されました。

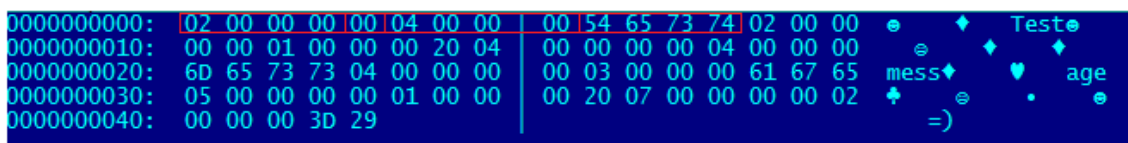


現時点では、データ監視はスプリッタデバイス、コネクタデバイス、ペアデバイスなどのカーネルモードデバイスでのみ機能します。

バイナリ形式で保存する

この機能は、ソフトウェア開発者にとって役立ちます。保存をオンにするには、[保存...]ボタンをクリックします。バイナリファイルはレコードチャンクで構成されています。各チャンクのフォーマットは次のとおりです。

- SecondsFromBeginningOfTheRecord (DWORD - 4 バイト、リトルエンディアン形式)
- DataDirection (CHAR - 1 バイト)。受信データが 1 の場合は 0、送信データの場合は 0
- DataSizeInBytes (DWORD - 4 サイズ、リトルエンディアン形式)
- RawData (データ、サイズは DataSizeInBytes フィールドで指定)



バイナリーファイルの例

単純なバイナリファイルの例です。最初のレコードチャンクは赤い矩形で選択されます。次の値があります。


- SecondsFromBeginningOfTheRecord = 2 (2 秒)
- DataDirection = 0 (送信中)
- DataSizeInBytes = 4 (4 バイト)
- RawData = テスト

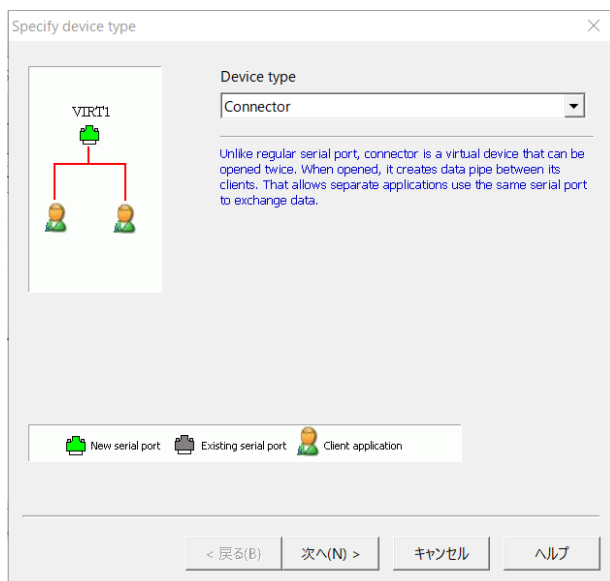
スパイデバイス

スパイのデバイスを見てください。同様の機能を提供しますが、別の方法で動作します。

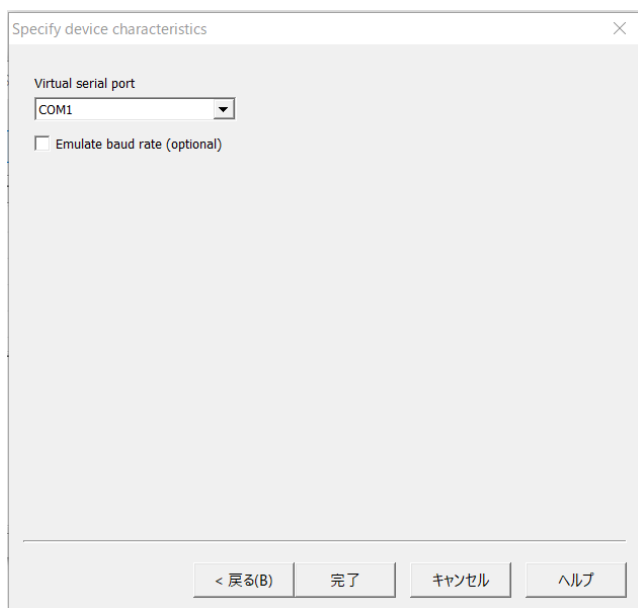
それぞれのデバイスの機能と設定

Connector (コネクタ)

コネクタは コネクタは仮想 COM ポートで Device=>Create 又は  アイコンのクリックで作成できる。



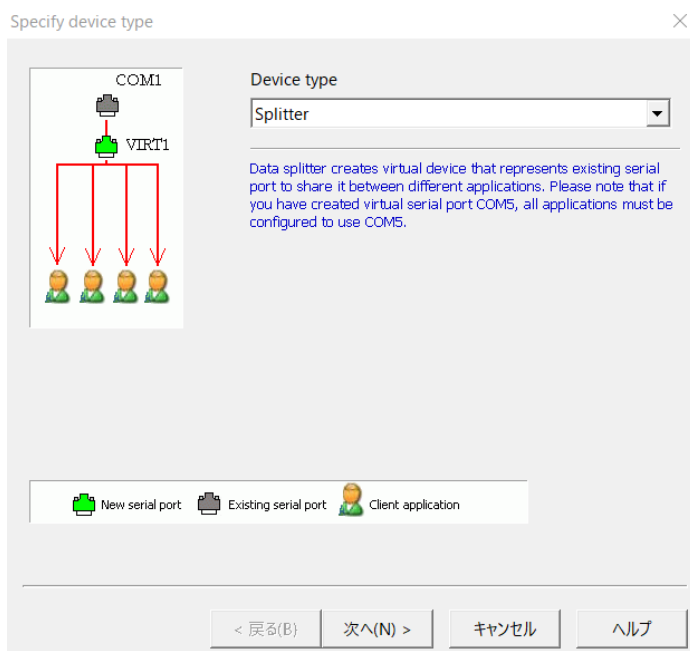
次へをクリック



希望する COM ポートをドロップダウンリストから選択する。既存のポートは選択できない。

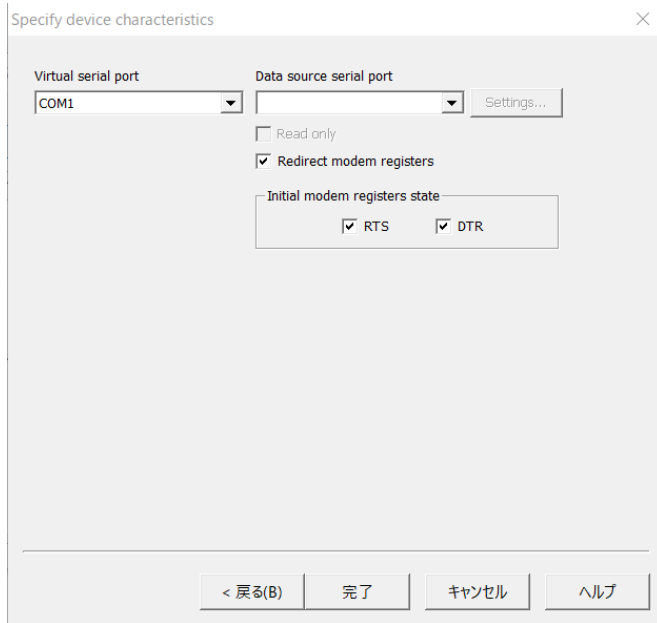
Splitter スプリッタ

スプリッタは既存の COM ポートの通信を、仮想 COM ポートを経由して別の複数のソフト通信する接続です。



実在の COM ポートと仮想 COM ポートを 1:1 で定義します。仮想 COM ポート側は異なるソフトで共有できます。

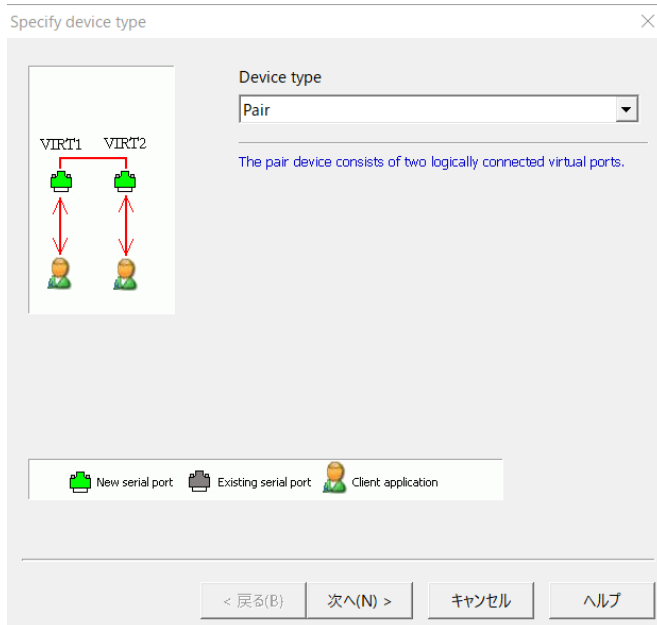
次へ をクリックし次の場面に進みます。

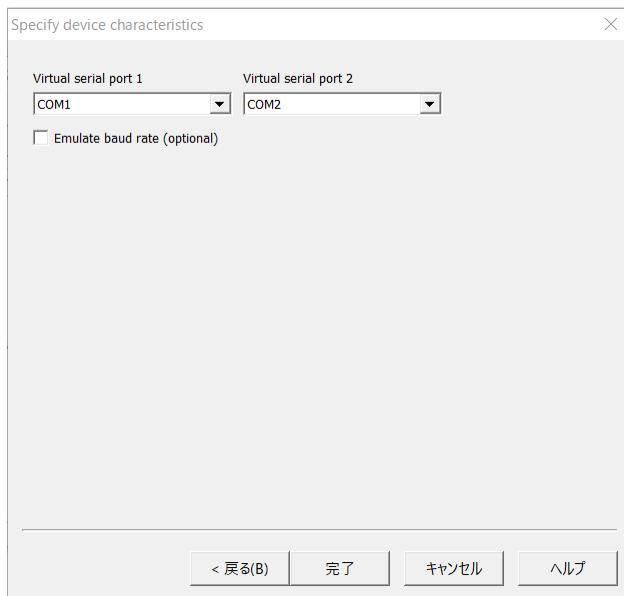


Virtual serial port に仮想 COM ポートをドロップダウンリストから選択して設定します。
Data source serial port には実在の COM ポートの中から希望のポートを選択します。

Pair ペア

Pair は 2 つの仮想 COM ポートを接続します。





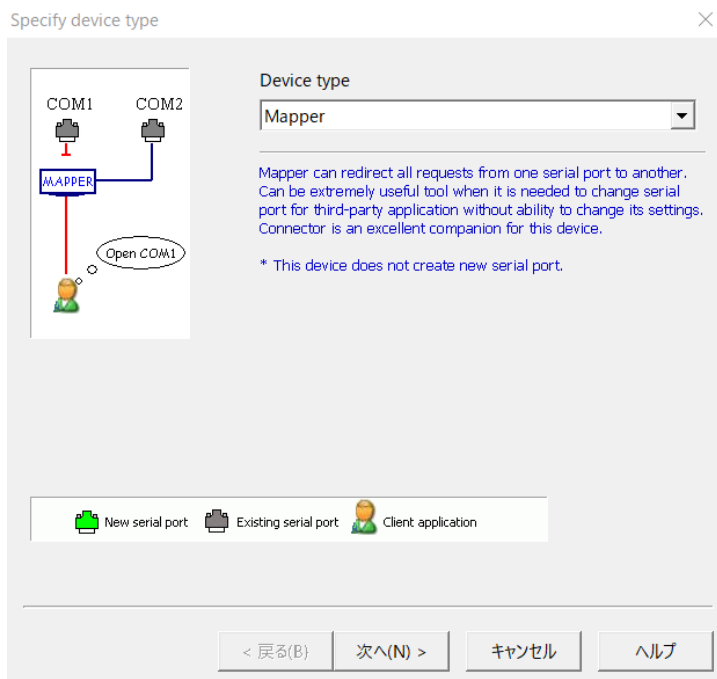
対応させたい2つのポートを設定します。

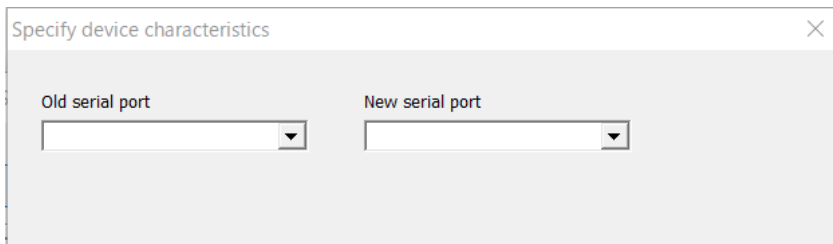
Mapper マッパー

マッパーはある COM ポートに対する動作を別の COM ポートに振り替える事ができます。

この設定では、新しいポートは作られません。

ユーザーのソフトでは選定を変えずに別の COM ポートにアクセスできます。



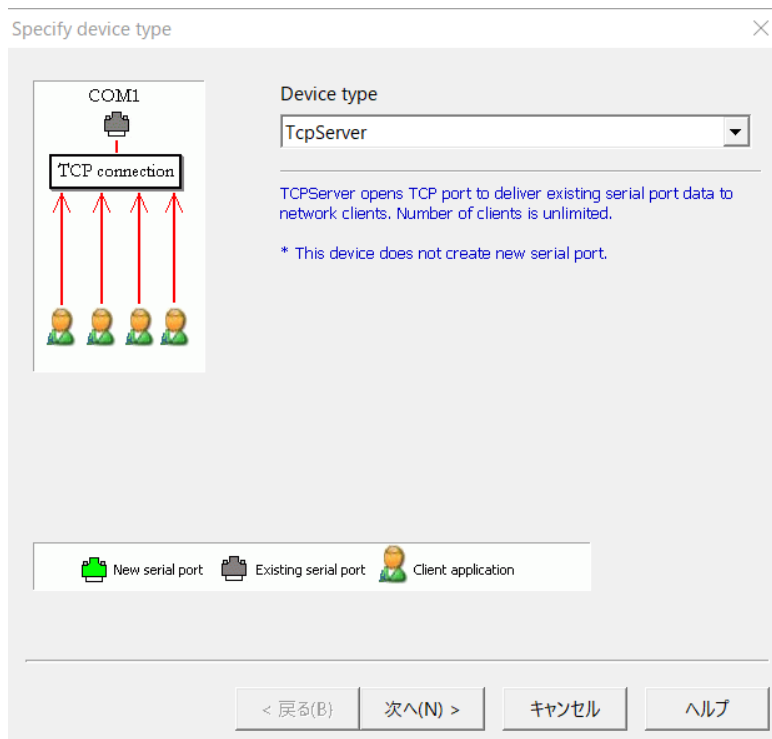


前の COM ポートと新しい COM ポートを設定します。

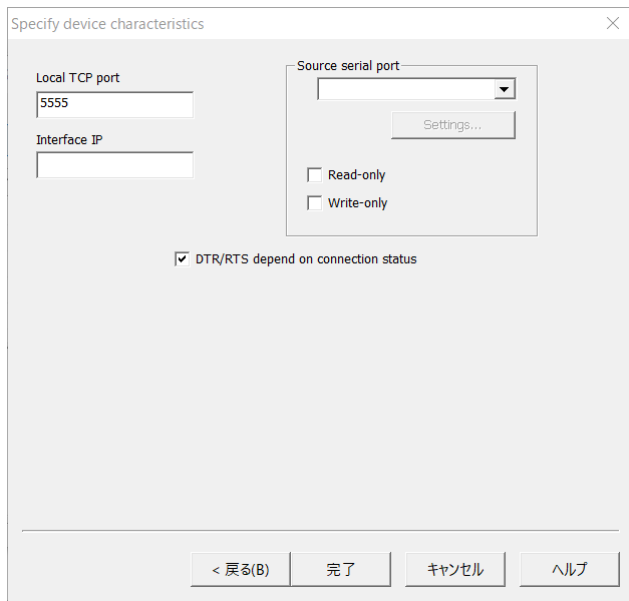
TcpServer

現存する COM ポートに送るデータをインターネット経由で TcpServer に送ります。

この設定では新しい COM ポートは定義されません。

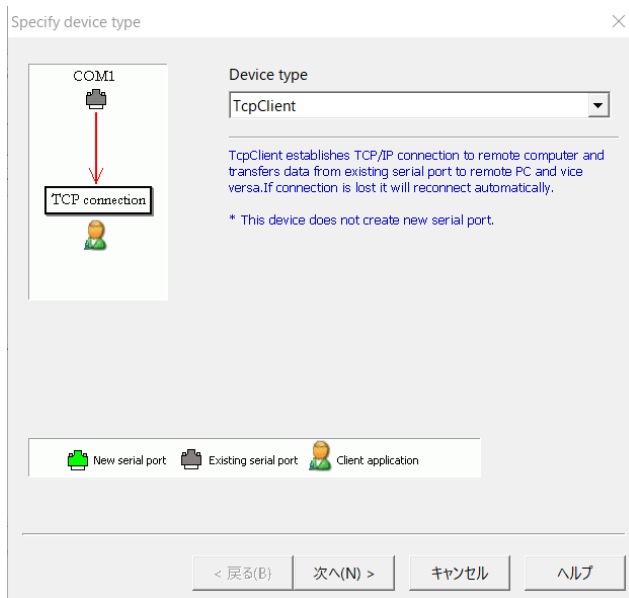


設定



接続先の IP アドレスとポートを指定し、こちら側の COM ポートも設定します。

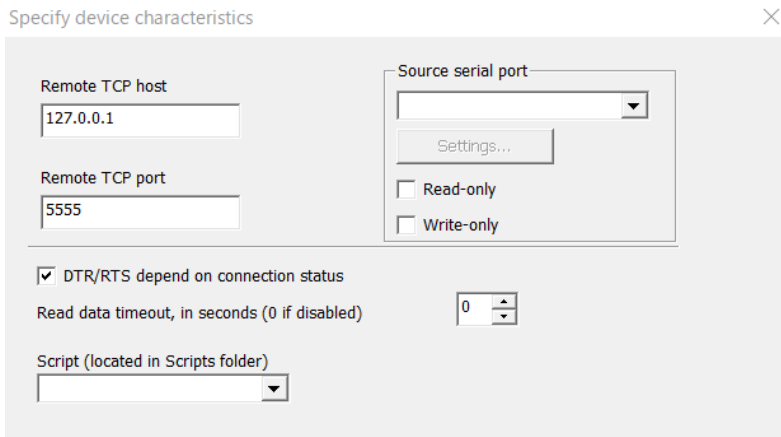
TcpClient



インターネット経由で接続するクライアント側の COM ポートを設定します。

接続が切断された場合に、自動再接続します。

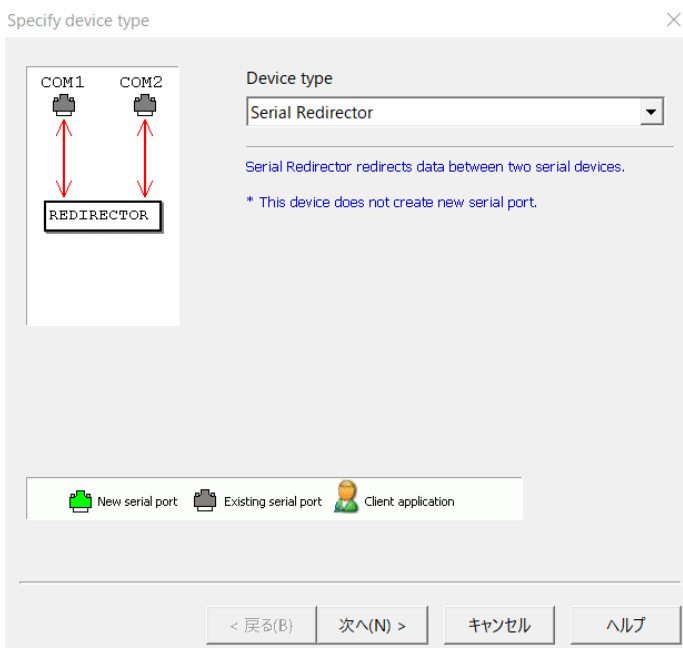
この設定では、COM ポートは作られません。

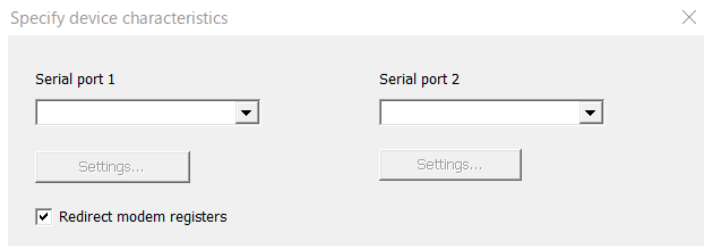


リモート側の IP アドレス、とポート番号を設定します。
リモート側の COM ポートを設定します。

Serial Redirector シリアル・リダイレクタ

2 つのポート間でデータを共有します。
この設定で COM ポートは作られません。

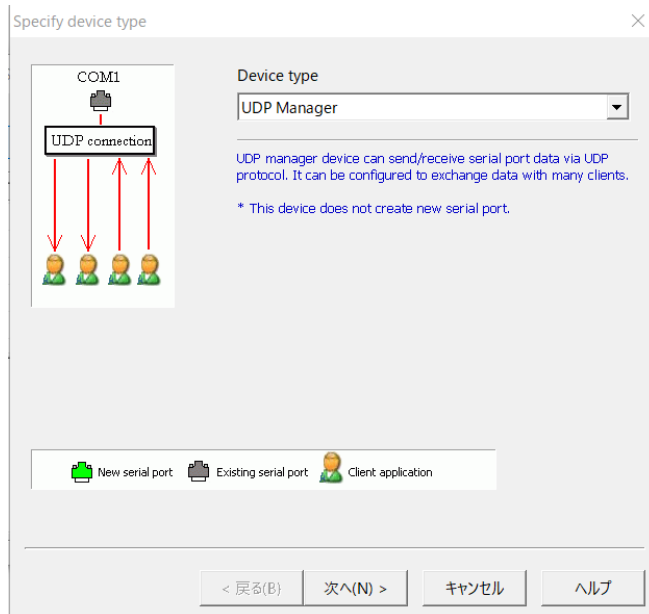


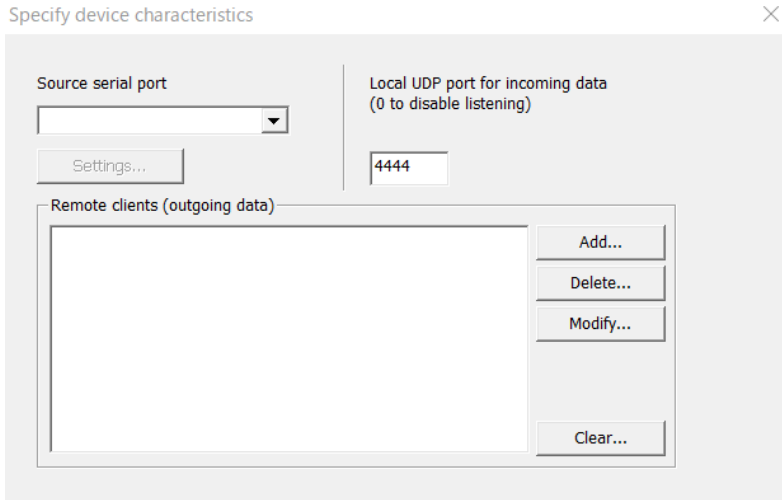


共有する 2 つのポートを指定します。

UDP Manager

インターネットの UDP プロトコルを使って COM の通信を接続します。
この設定では、COM ポートは作られません。



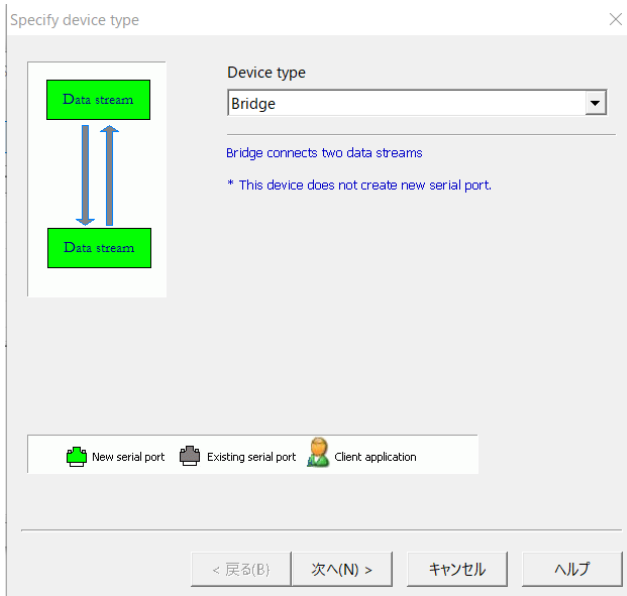


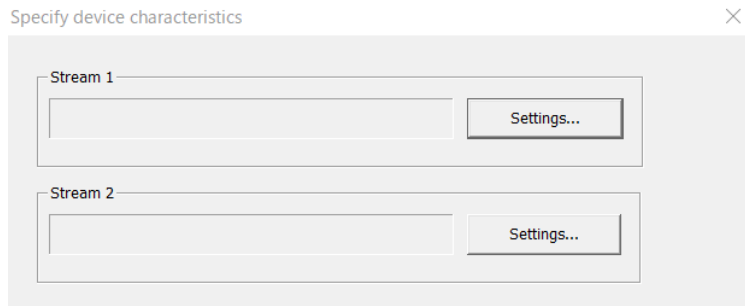
元になる COM ポートと UDP ポートを設定します。
クライアントの登録も行います。

Bridge

2つの通信データを橋渡しします。

この設定では、COM ポートは作られません。

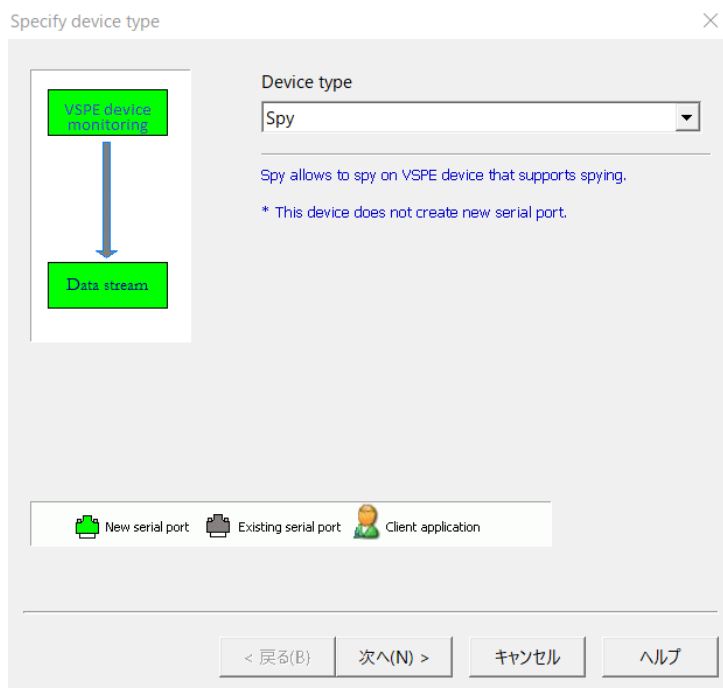




2つの通信ストリームを設定します。

Spy

指定の COM ポートの情報をデータにします。





COM ポートとデータの設定をします。

使用上のヒントとシナリオ

ここでは、最も一般的な VSPE の使用シナリオについて説明します。

仮想 VSPE デバイスを組み合わせることができます。たとえば、スプリッタデバイスをコネクタデバイスに接続すると同時に、コネクタデバイスを TcpServer デバイスを使用してローカルネットワークに公開することができます。 目標を達成するために、VSPE デバイスのチェーンと組み合わせを作成することができます。

リダイレクト **Redirect from file to virtual COM port**

(ファイルから COM ポートに送信)

あなたはソフトウェア開発者の場合。

あなたは、いくつかのデータを解析するソフトウェア（例えば、GPS 受信機からのデータ）を書きます。 あなたはあなたのプログラムを書いたが、あなたは GPS を持っていません。問題を解決するには、以下の手順に従います。

GPS データの例を見つけます（google.com などの情報源を使用してください）。 データをファイルに保存します（c : ¥ gps.txt）。

コネクタデバイスの作成（COM5）

ブリッジデバイスを作成します（Stream1 = COM5、Stream2 = File;入力パス= c : ¥ gps.txt;入力速度= 1024、出力パスは空のままにすることができます）。

COM5 ポートを開き、記録された GPS データを解析します。

Autorun: command line shortcut

Windows の起動時に仮想デバイスの設定を自動的に読み込みたい場合は、以下の手順に従ってください。

ステップ 1。

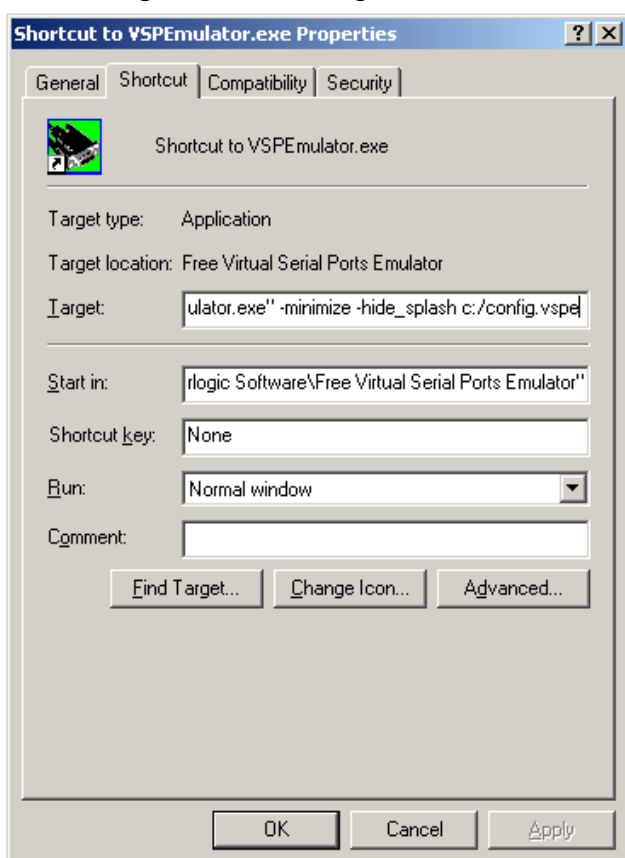
設定を作成し、"File => Save as ..."コマンドを使用して設定を保存します。設定ファイルの場所を覚えている（たとえば、c : /config.vspe）

ステップ 2。

ウィンドウのショートカットを作成します。デフォルトの場所（"C : ¥Program Files¥Eterlogic.com¥Virtual Serial Ports Emulator"）に VSPE をインストールした場合、ショートカットパラメータは次のようにする必要があります。

ターゲット : "C : ¥Program Files¥Eterlogic.com¥仮想シリアルポートエミュレータ ¥VSPEmulator.exe" - minimize -hide_splash c : /config.vspe

「C : ¥ Program Files ¥ Eterlogic.com ¥仮想シリアルポートエミュレータ」で起動します。



ステップ 3。

作成したショートカットを "スタートメニュー¥プログラム¥スタートアップ"フォルダに移動します。

参照：

コマンドライン

Autorun: run as service

VSPE は、App-to-Service のようなプログラムの 1 つを使用してサービスとして開始できます。

優れた NSSM ツールを使用して正常にチェックしました。それはフリーでオープンソースです。 <http://iain.cx/src/nssm> からダウンロードできます。

ステップ 1。

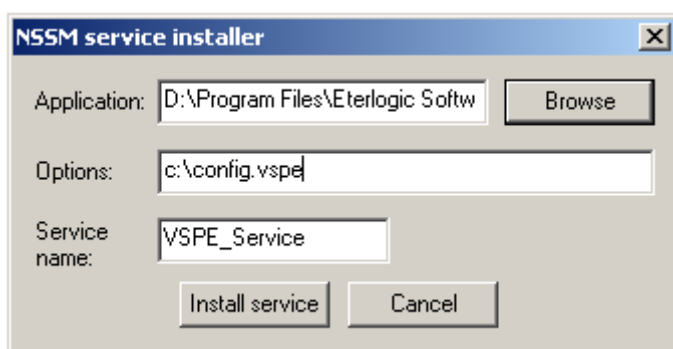
NSSM をインストールする

ステップ 2。

ファイルに VSPE 設定を作成して保存します (たとえば、`c:\config.vspe`)。

ステップ 3。

インストールサービス：NSSM フォルダに移動し、"`nssm install VSPE_Service`"を実行します。VSPEmulator.exe と設定ファイルへのパスを指定し、"Install service"ボタンをクリックしてください。



手順 4. 「`net start VSPE_Service`」 コマンドを使用してサービスを実行するか、コンピュータを再起動します。

それで全部です。

サービスをアンインストールするには、NSSM フォルダに移動し、 "nssm remove VSPE_Service"を実行します。

スプリッタ **Splitter: Share existing physical serial port between several applications**

(同じ COM ポートを別のソフトで共有する)

あなたのコンピュータに既に物理シリアルポート COM1 に接続されている GPS 受信機があるとしましょう。

COM1 を開いて GPS データを受信しているプログラムがあります。同時に、GPS データを取得するために別のプログラムを実行するまでは、すべて問題ありませんでした。しかし、それを実行しようとすると、 "シリアルポートを開くことができません" というような奇妙なメッセージが表示されます。何が起こった ? 答えは : シリアルポートは 1 つのアプリケーションだけで開くことができます。シリアルポートは排他アクセスデバイスタイプです。

このような状況に対処するには、いくつかの選択肢があります。

- 最初のプログラムを閉じて、2 番目のプログラムを実行してください。そして、2 番目を閉じて最初に実行してください...実際には、オプションではありません。
- シリアルポート拡張ボードを購入してください。はい、このオプションは機能します。しかし、時間とお金を費やし、空き PCI スロットを探して拡張ボードをインストールしたいのですか？
- シリアルポートが 2 つある場合は、GPS から各シリアルポートにデータを分割する特別なケーブルを購入することができます。しかし、物理的なシリアルポートが 1 つしかない場合や、2 つ以上のアプリケーションを実行する必要がある場合は、それを忘れることができます...
- 最後に、VSPE などのソフトウェアソリューションを使用できます。それは最善の選択肢です。

ステップ 1。

シリアルポートを開いたプログラムを閉じます。

ステップ 2。

VSPE を実行し、スプリッタデバイスを作成します。「データソースシリアルポート」コンボボックスで COM1 を選択し、「仮想シリアルポート」で COM2 を選択します。

ステップ 3。

COM2 から GPS データを受信するようにすべてのプログラムを設定します。それができない場合は、Mapper デバイスと Connector and Mapper のリンクを参照してください。

COM2 は最大 8 種類のアプリケーションで使用できます。シンプルではないですか？

Connector and Mapper link

状況 1

あなたが持っているとは仮定しよう：

- サードパーティアプリケーション。常に COM1 シリアルポートを開いて動作します。COM1 をアプリケーション設定から別のポートに変更する方法はありません。
- COM1 は物理的にこのコンピュータにインストールされています。

アプリケーションと COM1 の間でデータストリームにアクセスして変更する必要があります。アプリケーション設定を変更できる場合は、コネクタ (COM2) を作成し、アプリケーション設定を COM2 に変更し、COM1 と COM2 を接続する独自のアプリケーションブリッジを実行して、オンザフライでデータストリームを変更できます。しかし、アプリケーションの設定を変更することはできません！この問題を解決するには、Mapper デバイスを使用できます。

- コネクタデバイスの追加 (COM2)
- マッパーデバイスを追加する (COM1→COM2)

それで全部です！これで、サードパーティのアプリケーションが COM2 と連携し、独自のアプリケーションが意図したとおりに動作するようになります。

状況 1

「スプリッタ：複数のアプリケーション間で既存の物理シリアルポートを共有する」で説明しているのと同じ状況です。唯一の違いは、一部のアプリケーションでは設定を変更できないため、スプリッタデバイスが役に立たないことです。この場合、新しい VSPE デバイス Mapper (COM1 -> COM2) を追加するだけです。問題を解決します。Splitter の後に Mapper を作成する必要があります。これは、逆にスプリッタが正常に動作しないためです。

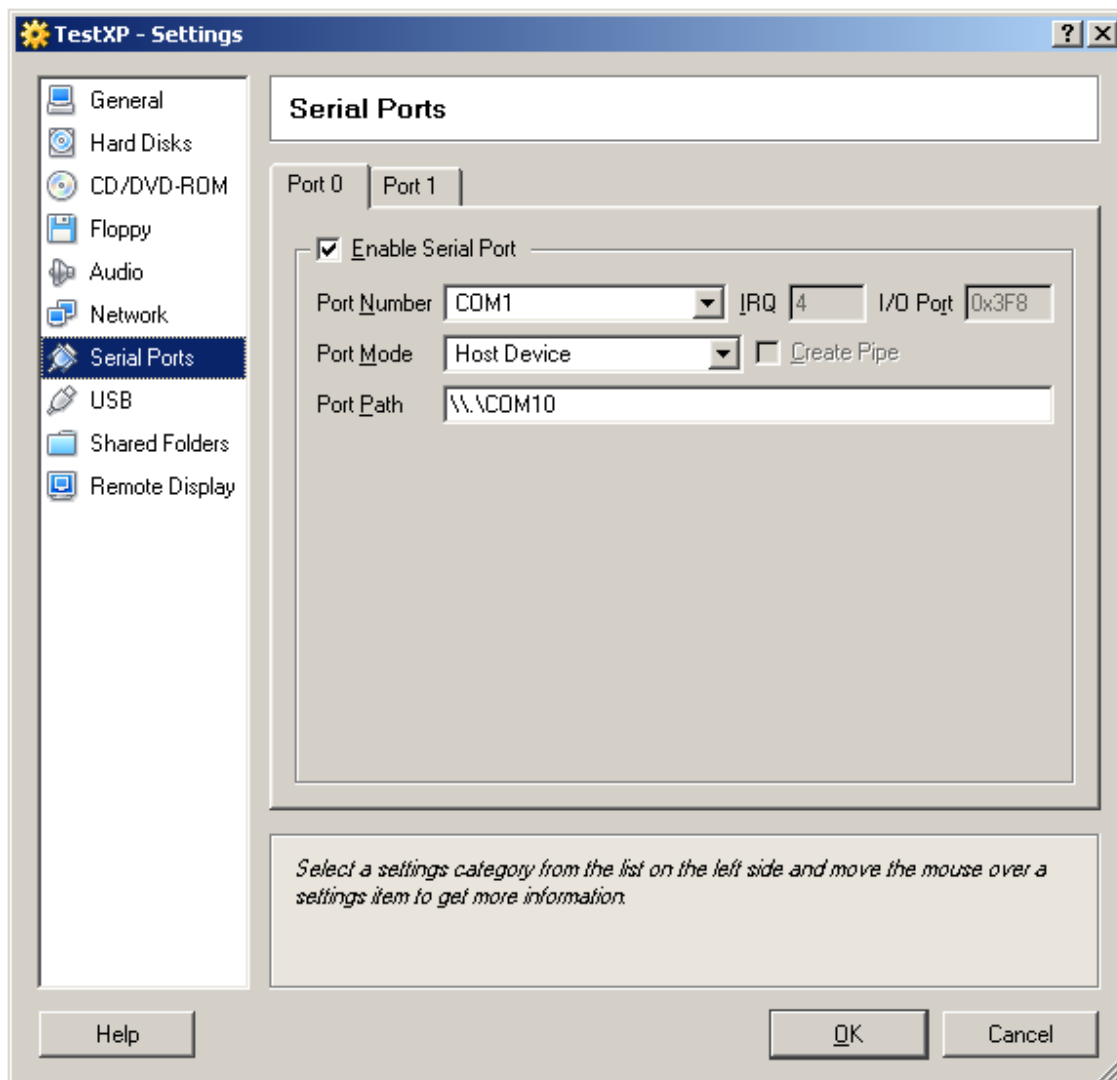
Virtual machine integration

仮想マシン (VirtualBox、VirtualPC、VmWare など) を使用する場合は、仮想シリアルポート (COM10 など) を作成して仮想マシンのデータソースとして指定できます。

これは、デバッグ/開発目的には極端に役立つかもしれませんが。

ここに VirtualBox の設定例があります (シリアルポート名は「¥¥。¥COM10」です)。 詳

細は、よくある質問のセクションを参照してください。



Network bridge

2つのコンピュータをシリアルポート経由で接続する場合は、シリアルケーブルが必要です。しかし、多くの場合、ケーブルが短すぎたり、コンピュータがさまざまな都市や国に配置されたりする可能性があるため、不便な場合があります。

VSPEはそれを解決できます：それを可能にする「ネットワークブリッジ」設定を作成するだけです。「ネットワークブリッジ」は、TcpServer、TcpClient、およびコネクタ（またはペア）デバイスの組み合わせです。

以下は、ネットワーク接続を使用して仮想シリアルポート COM5 を介して 2 台のコンピュータを接続するような設定の例です。

コンピュータ 1

- コネクタデバイスの作成 (COM5)
- TcpServer デバイスを作成する (ローカル Tcp ポート= 5555、ソースシリアルポート= COM5)

コンピュータ 2

- コネクタデバイスの作成 (COM5)
- TcpClient デバイスを作成する (リモート TCP ホスト=コンピュータ 1 の IP アドレスまたはネットワーク名、リモート TCP ポート= 5555、ソースシリアルポート= COM5)

それで、両方のコンピュータが仮想シリアルポート COM5 で接続されるようになりました。

One-to-Many data redirection

COM1、COM2、COM3 があるとします。

これらのポートは、実際のポートまたは仮想シリアルポート (コネクタやペアなど) にすることができます。データ交換を作成する場合は、COM1 <=> COM2 と COM1 <=> COM3 のいずれかを使用できます。

スプリッタとシリアルリダイレクタの組み合わせ

- スプリッタデバイスの作成 (データソースシリアルポート=COM1、仮想シリアルポート= COM5)
- シリアルリダイレクタを作成する (シリアルポート 1 = COM5、シリアルポート 2 = COM2)
- シリアルリダイレクタを作成する (シリアルポート 1 = COM5、シリアルポート 2 = COM3)

TcpServer と TcpClient の組み合わせ

- TcpServer デバイスを作成する (ソースシリアルポート= COM1、ローカル TCP ポート= 5555)
- TcpClient デバイスを作成します (ソースシリアルポート= COM2、リモート TCP ホスト= 127.0.0.1、リモート TCP ポート= 5555)
- TcpClient デバイスを作成します (ソースシリアルポート= COM3、リモート TCP ホスト= 127.0.0.1、リモート TCP ポート= 5555)

Monitoring bidirectional traffic for debugging purposes

あなたが必要なもの

あなたがソフトウェア開発者であると仮定しましょう。あなたは 2 台のコンピュータを持っています。最初のコンピュータ（開発者用マシン）には、実際の COM ポートはありません。2 台目のコンピュータ（テストマシン）には実際の COM1 ポートがあり、そこにいくつかのハードウェアが接続されています。あなたが欲しい：

- 開発者マシンで COM5 を開き、いくつかのコマンドを送信します。テストマシン上のハードウェアはこれらのコマンドを受け取り、それらに応答する必要があります。もちろん、開発者マシンの COM5 ポートからの応答が必要です。
- 後で分析するために、すべてのデータをファイルに保存する必要があります。

どうやってするの

まあ、主なアイデアは、開発者のコンピュータに仮想ポート COM5 を作成し、テストソケットを介してテストコンピュータに接続し、トラフィックをファイルに記録することです。これを行うには、次の構成を作成する必要があります。

開発者マシンで

- コネクタデバイスの作成（COM5）
- TcpClient デバイスを作成します（ソースポート COM5、リモート TCP ホスト - テストマシン IP、例としてポート 5555）。
- スパイデバイスの作成：VSPE デバイスインデックス= 1（これは作成したばかりの TcpClient、宛先ストリーム=ファイル、出力パス=c : ¥ output.txt）

試験機で

- TcpServer デバイスを作成する（COM1、ポート 5555）

これで、開発者マシンの COM5 を使用してリモート COM ポートからデータを送受信できるようになりました。すべてのデータは c : ¥ output.txt ファイルに記録されます。

参照：

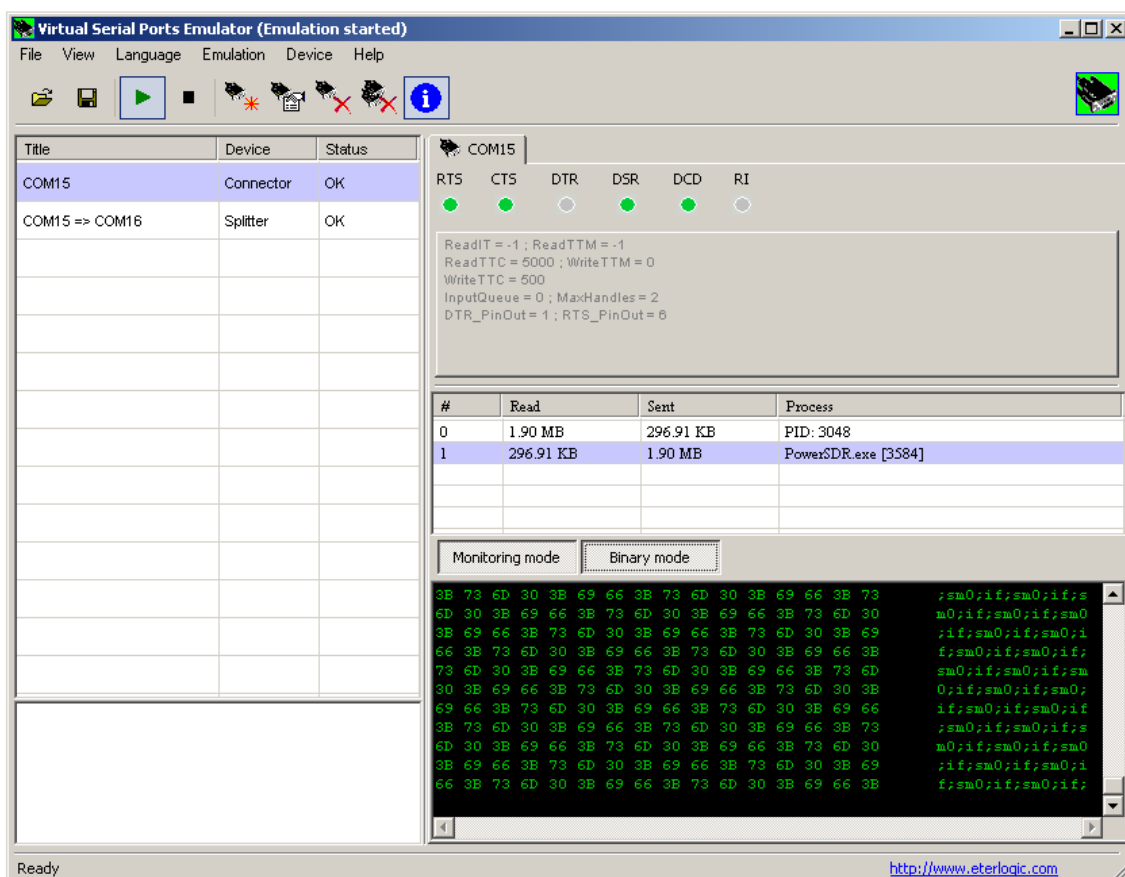
スパイデバイス

User interface

VSPE のユーザーインターフェイスは非常に簡単です。

メインウィンドウとさまざまなオプションダイアログがあります。メインウィンドウには、作成された仮想シリアルデバイスのリストが表示されます。

メインウィンドウ



アプリケーションメインウィンドウには、作成された仮想デバイスに関する情報が表示されます。

メニューまたはツールバーのコマンドを使用して設定を作成します。

作成したデバイスを編集するには、まずエミュレーションを停止する必要があります。

"Emulation => Stop"コマンドをクリックし、右クリックします。

ボタンを押します。次に、"Properties ..."メニュー項目をクリックします。

VSPE Preferences

環境設定ダイアログを表示するには、「表示=>環境設定」メニューコマンドをクリックします。

すべての設定は、現在のユーザーの Windows レジストリに格納されます。

参照：

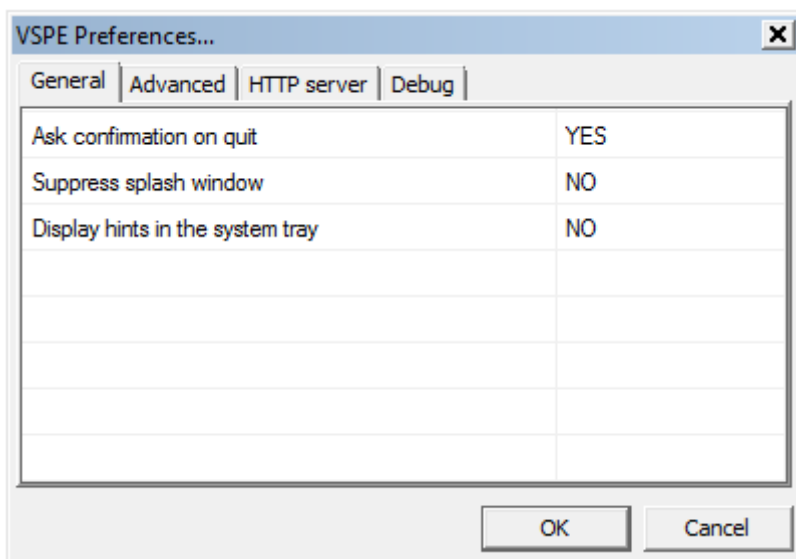
[全般]タブ

詳細設定タブ

HTTP サーバータブ

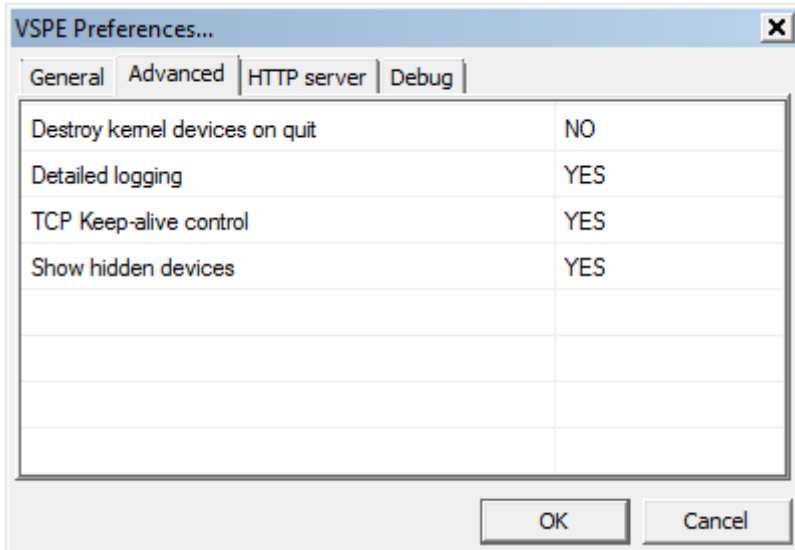
デバッグタブ

General tab

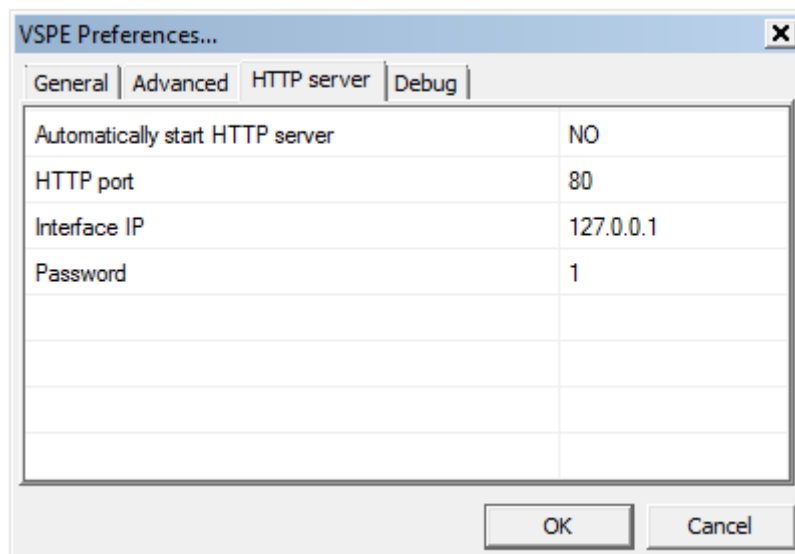


オプションの説明を読むには、VSPE 環境設定ダイアログを開いて、希望の項目にマウスを移動し、ツールチップを読んでください。

Advanced tab

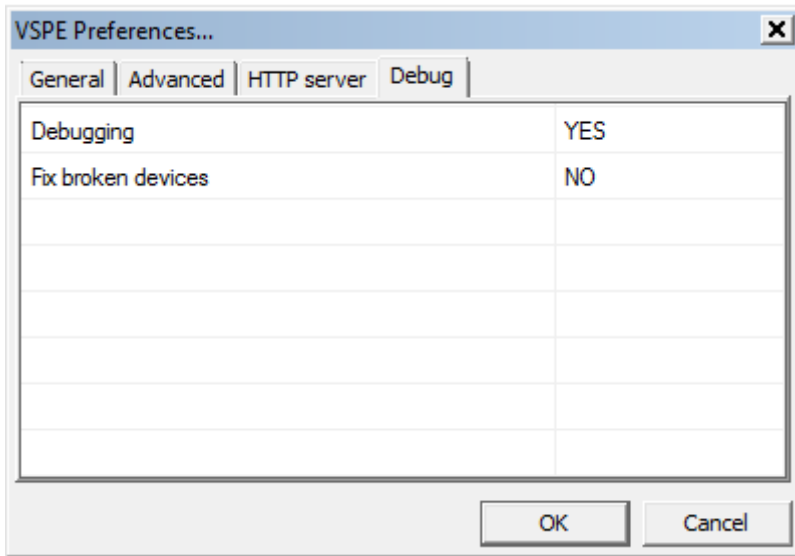


HTTP server tab



オプションの説明を読むには、VSPE 環境設定ダイアログを開いて、希望の項目にマウスを移動し、ツールチップを読んでください。

Debug tab



Create device wizard

このウィザードは、目的のデバイスを作成するのに役立ちます。デバイスの作成プロセスは、デバイスのタイプと特性を指定する 2 つの簡単なステップから構成されます。

任意のデバイスの組み合わせを作成できます。たとえば、TcpServer デバイスを使用してコネクタを作成し、ネットワークに公開することができます。

Serial port settings dialog

シリアルポート設定（速度、ストップビットなど）を調整します。

