

# TCP-VSP 連続 Open/Close の不具合状況

2版 2008年4月21日

## 1. 概要

### 1. 1 概要

TCP-VSP により作成した仮想 COM ポートを連続で Open/Close を行った場合、COM の状態と TCP/IP の状態が異なる状況が発生します。

本アプリケーションノートでは、不具合内容と対応策を説明します。

### 1. 2 対象バージョン

- ・TCP-VSP は Ver1.14 以前のすべてのバージョン
- ・TCP-VSP for ezTCP は Ver1.04 以前のすべてのバージョン

※本不具合は TCP-VSP Ver1.20 及び TCP-VSP for ezTCP Ver1.10 以降のバージョンで改善されています。

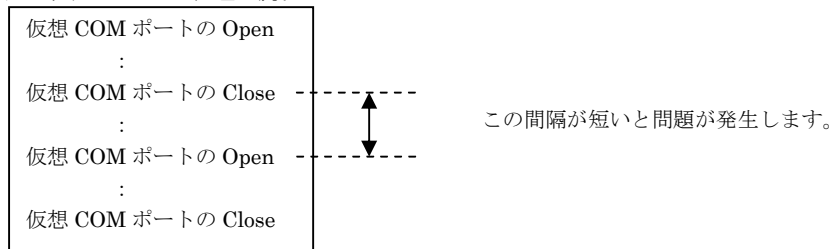
## 2. 症状

### 2.1 発生条件

仮想 COM ポートを連続で Open/Close を行うような場合、Close と Open の間隔が短いと、COM の状態と TCP/IP の状態が異なる不具合が発生します。

なお、タイミングによる症状のため、発生する時としない時があります。

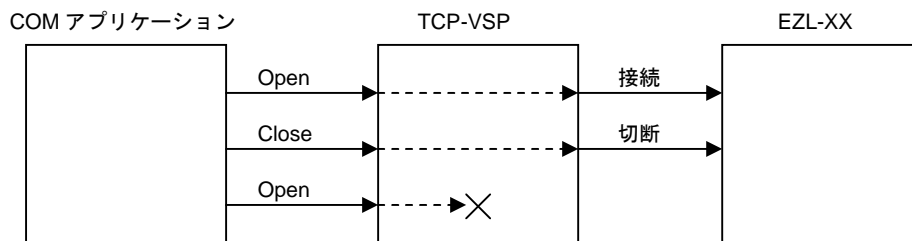
アプリケーションの処理の流れ



### 2.2 症状

仮想 COM ポートの Open/Close を連続で繰り返すような処理を行う時に、Close から Open を短い間隔で実行すると、TCP-VSP の内部処理で Open を取りこぼしてしまい、TCP/IP の接続処理が行われない状況になります。

その時に、COM アプリケーション側では、Open 状態でも、TCP/IP 側では切断状態となり、異なる状態になってしまいます。



### 3. 対策

#### 3. 1 COM アプリケーションの変更

この不具合を回避するためには、COM アプリケーション側で対策する必要があります。  
対策としては、連続して Open/Close を行う場合には、Close 処理が終わった後に次の Open 処理までに、少し間を空けます。  
以下に C 言語による対策例を記述します。

##### ●不具合が起こるパターンのコード例

```
// COMM ポートのオープン //
m_hComm = (HANDLE)::CreateFile("COM3", GENERIC_READ | GENERIC_WRITE, 0, NULL, OPEN_EXISTING,
                               FILE_ATTRIBUTE_NORMAL | FILE_FLAG_OVERLAPPED, NULL);
if(m_hComm == INVALID_HANDLE_VALUE)
    return;
// COMM ポートのクローズ //
if(::CloseHandle(m_hComm) == FALSE)
    return;

// COMM ポートのオープン //
m_hComm = (HANDLE)::CreateFile("COM3", GENERIC_READ | GENERIC_WRITE, 0, NULL, OPEN_EXISTING,
                               FILE_ATTRIBUTE_NORMAL | FILE_FLAG_OVERLAPPED, NULL);
if(m_hComm == INVALID_HANDLE_VALUE)
    return;
// COMM ポートのクローズ //
if(::CloseHandle(m_hComm) == FALSE)
    return;
```

##### ●対策例

```
// COMM ポートのオープン //
m_hComm = (HANDLE)::CreateFile("COM3", GENERIC_READ | GENERIC_WRITE, 0, NULL, OPEN_EXISTING,
                               FILE_ATTRIBUTE_NORMAL | FILE_FLAG_OVERLAPPED, NULL);
if(m_hComm == INVALID_HANDLE_VALUE)
    return;
// COMM ポートのクローズ //
if(::CloseHandle(m_hComm) == FALSE)
    return;

Sleep(1);

// COMM ポートのオープン //
m_hComm = (HANDLE)::CreateFile("COM3", GENERIC_READ | GENERIC_WRITE, 0, NULL, OPEN_EXISTING,
                               FILE_ATTRIBUTE_NORMAL | FILE_FLAG_OVERLAPPED, NULL);
if(m_hComm == INVALID_HANDLE_VALUE)
    return;
// COMM ポートのクローズ //
if(::CloseHandle(m_hComm) == FALSE)
    return;
```

※赤字のような処理を追加して次の Open 処理の開始を遅くします。上記の例の場合、約 1 ミリ秒スリープする処理となります。  
なお、動作環境によってタイミングが異なってきますので、上記間隔でも問題あるようでしたら、もう少し長めに間隔を設定してください。

※コード中の変数 m\_hComm は、送信を行う通信デバイスのハンドルを表します。

## ご注意

- ・本文書の著作権は（株）アルファプロジェクトが保有します。
- ・本文書の内容を無断で転載することは一切禁止します。
- ・本文書に記載された回路図およびサンプルプログラム等の著作権は（株）アルファプロジェクトが保有しますが、お客様のアプリケーションで使用される場合には、ご自由にご利用いただけます。
- ・本文書の内容は、将来予告なしに変更されることがあります。
- ・本文書に記載されている内容およびサンプルプログラムについての質問等のサポートは一切受け付けておりませんのでご了承ください。
- ・本文書の内容については、万全を期して作成いたしました。万が一不審な点、誤りなどお気づきの点がありましたら弊社までご連絡下さい。
- ・本文書の内容およびサンプルプログラムに基づき、アプリケーションを運用した結果、万一損害が発生しても、弊社では一切責任を負いませんのでご了承下さい。

- ・Windows®の正式名称は Microsoft®Windows®Operating System です。  
Microsoft、Windows、Windows NT は、米国 Microsoft Corporation.の米国およびその他の国における商標または登録商標です。  
Windows®Vista、Windows®XP、Windows®2000 は、米国 Microsoft Corporation.の商品名称です。  
本文書では下記のように省略して記載している場合がございます。ご了承ください。  
Windows®Vista は Windows Vista もしくは WinVista  
Windows®XP は Windows XP もしくは WinXP  
Windows®2000 は Windows 2000 もしくは Win2000
- ・その他の会社名、製品名は、各社の登録商標または商標です。



株式会社アルファプロジェクト  
〒431-3114  
静岡県浜松市東区積志町 834  
<http://www.apnet.co.jp>  
E-MAIL : [query@apnet.co.jp](mailto:query@apnet.co.jp)