

業務システム向け FAXサーバーソフト スターファクス サーバー エスディーケー

STARFAX[®] ***Server SDK***

VB 開発向け OCX プログラミング

はじめに

本書は、STARFAX Server SDK を利用した ユーザープログラム の作成方法をご説明しています。
なお、本書は、Visual Basic 6.0 を開発ツールとして、STARFAX Server SDK OCX コントロールを使用してプログラムを作成する方を対象としています。そして、STARFAX Server SDK とユーザープログラムとのインターフェイスは、STARFAX Server SDK OCX のコントロールが処理します。
本書をお読みになる前に、「STARFAX Server SDK セットアップマニュアル」をお読みいただき、STARFAX Server SDK の動作をご理解していただくようお願いいたします。

■ご注意

本書に登場するシステム名・製品名は、一般に開発メーカーの登録商標です。

本書の構成について

本書は、次のような内容で構成されています。

- 第Ⅰ章 ユーザープログラム の開発について
- 第Ⅱ章 サンプルプログラム
- 第Ⅲ章 クライアントプログラムの開発について
- 第Ⅳ章 FAX 送信命令フォルダを共有して FAX 送信
- 第Ⅴ章 [メール de FAX]で FAX 送信

まずは、第Ⅰ章に開発の手順等をご説明していますので、第Ⅰ章をご覧になってから第Ⅱ章にお進み下さい。

目次

はじめに	1
本書の構成について	1

第Ⅰ章 ユーザープログラムの開発について

1.1 開発の手順	6
1.2 ユーザープログラムの基本的な動き	7

第Ⅱ章 サンプルプログラム

2.1 FAX 送信	9
2.1.1 FAX を送信する	11
2.2 FAX 情報の参照、削除	18
2.2.1 送信情報を参照する	19
2.2.2 受信情報を参照する	29
2.2.3 未送信情報を参照する	39
2.2.4 送信情報を削除する	49
2.2.5 受信情報を削除する	52
2.2.6 未送信情報を削除する	55
2.3 FAX 情報の追加、削除状況の取得	58
2.3.1 送信情報が追加された事を知る	59
2.3.2 受信情報が追加された事を知る	67
2.3.3 送信情報が削除された事を知る	75
2.3.4 受信情報が削除された事を知る	77
2.4 ファイル印刷	79
2.4.1 ファイルを印刷する	80
2.5 メール送信	83
2.5.1 メールを送信する	85

2.6 動作情報の参照 90
2.6.1 動作情報を参照する 91

第Ⅲ章 クライアントプログラムの開発について

3.1 開発の手順 99
-----------	----------

第Ⅳ章 FAX 送信命令フォルダを共有して FAX 送信

4.1 FAX 送信命令フォルダを共有して FAX 送信 101
4.2 サンプルプログラム 102
4.2.1 FAX 送信する ... (SendFAX.exe) 103
4.2.2 印刷結果の FAX 送信 ... (PrtCli.exe) 107
4.2.3 TIFF ファイルの作成と FAX 送信 ... (PrtCli2.exe) 111

第Ⅴ章 [メール de FAX] で FAX 送信

5.1 [メール de FAX] で FAX 送信 116
5.2 [メール de FAX] の依頼メールの仕様 117
5.3 サンプルプログラム 123
5.3.1 FAX 送信する ... (SendFAX.exe) 124
5.3.2 印刷結果の FAX 送信 ... (PrtCli.exe) 129
5.3.3 TIFF ファイルの作成と FAX 送信 ... (PrtCli2.exe) 134

第 I 章

ユーザープログラムの開発について

ユーザープログラムの開発の手順や概要についてご説明しています。

- 1.1 開発の手順
- 1.2 ユーザープログラムの基本的な動き

1.1 開発の手順

STARFAX Server SDK を操作するユーザープログラムは、STARFAX Server SDK が動作していることが前提です。したがって、ユーザープログラム 開発を行う前に STARFAX Server SDK 本体のインストールを行い、その操作を簡単に理解しておく必要があります。

それらを考慮して、以下の手順でユーザープログラム開発を行うことをお勧めします。

- ① 「STARFAX Server SDK セットアップマニュアル」をお読みください。
 - STARFAX Server SDK 本体のインストールを行ってください。
 - 基本的な操作を理解してください。
- ② 1.2 ユーザープログラムの基本的な動きをお読みください。
 - STARFAX Server SDK に対して、ユーザープログラムから操作できることを理解してください。
- ③ STARFAX Server SDK で実現したいことを、第Ⅱ章 サンプルプログラム よりお選びください。
 - サンプルプログラムを学習してください。
- ④ ユーザープログラム を作成してください。
 - ①～③ を踏まえて、ユーザープログラムの作成、テストを行ってください。

なお、STARFAX Server SDK プログラミング(OCX 利用版)は、記述されている STARFAX Server SDK OCX のメソッド や、各種項目の内容についての詳細上方は、「STARFAX Server SDK VB 開発向け OCX リファレンスマニュアル」、「STARFAX Server SDK ファイル de FAX」にてご説明しておりますので、本マニュアルと併せてご利用ください。

1.2 ユーザープログラムの基本的な動き

STARFAX Server SDK のプログラムインターフェイスには、

- STARFAX Server SDK への動作命令
- STARFAX Server SDK の情報の参照

があります。

そして、それらの STARFAX Server SDK に対しての処理は、STARFAX Server SDK OCX の各種コントロールが行います。

■ STARFAX Server SDK への動作命令

- FAX 送信
FAX 送信を行います。
- FAX 情報の削除
任意の FAX 送受信情報を削除します。
- ファイル印刷
ファイルを印刷します。
- メール送信
メールを送信します。

■ STARFAX Server SDK の情報の参照

- FAX 情報
FAX の送受信情報 を参照できます。
- FAX 情報の追加、削除状況
STARFAX Server SDK によって、追加、削除、または、 ユーザープログラム によって削除された FAX 送受信情報 (以前との差分) を知ることができます。
- 動作情報
STARFAX Server SDK の起動状況、バージョン等を参照できます。

第Ⅱ章

サンプルプログラム

STARFAX Server SDK を操作するサンプルプログラムです。

- 2.1 FAX 送信
- 2.2 FAX 情報の参照、削除
- 2.3 FAX 情報の追加、削除状況の取得
- 2.4 ファイル印刷
- 2.5 動作情報の取得

2.1 FAX 送信

FAX 送信プログラム【SendFax.exe】は、FAX の送信を行うサンプルプログラムです。本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥SendFax.exe ... FAX 送信プログラム
¥サンプル¥VB6 SP6¥SendFax¥ ... FAX 送信プログラム 開発プロジェクト

主な仕様、および操作方法是以下の通りです。

- ① STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
- ② FAX 送信プログラム【SendFax.exe】を起動します。
(起動時の作業(カレント)フォルダの指定は特にありません)
- ③ 相手先(S)ボタンを押して、相手先を指定します。
 - ・ 最大4件まで指定できます。(これは、このサンプルプログラムの仕様です)
 - ・ 最低限、FAX 番号(A)の入力が必要です。送付状への差込が必要な場合や、送信情報に情報を残したい場合は、会社名(B)以降の項目も入力します。
- ④ 原稿(D)ボタンを押して、原稿を指定します。
 - ・ 最大4つのファイルを指定でき、送信時に連結して送信されます。
(これは、このサンプルプログラムの仕様です)
 - ・ 最低限、1つの原稿ファイルを指定する必要があります。ただし、送付状が指定されている場合は、原稿の指定がなくても、送付状のみ送信されます。
 - ・ 指定できるファイル形式は以下の通りです。
 - ・ TIFF 形式 圧縮なし、修正 CCITT MH 圧縮、CCITT G3 MH 圧縮、CCITT G3 MR 圧縮
PackBits 圧縮、Class F 圧縮、G4 圧縮、JPEG 圧縮
 - ・ BMP ファイル
 - ・ PCX ファイル
 - ・ DCX ファイル
 - ・ JPEG ファイル
 - ・ テキストファイル
 - ・ FAX ファイル
 - ・ LNK ファイル

- ⑤ 必要であれば、送付状(C)ボタンを押して、送付状を指定します。
(指定できる送付状の仕様に関しては「STARFAX Server SDK ファイル de FAX」の 4.1 送信命令ファイル をご覧下さい)
 - ⑥ 必要であれば、発信元(U)ボタンを押して、発信元情報を指定します。
(指定できる発信元情報の仕様に関しては「STARFAX Server SDK ファイル de FAX」の 4.1 送信命令ファイル をご覧下さい)
 - ⑦ 送信(G)ボタンを押して、FAX 送信を行います。
この後、FAX 送信が正常に動作していない場合は、STARFAX Server SDK ログ管理プログラムでイベントの内容を確認して下さい。
-

2.1 FAX 送信

2.1.1 FAX を送信する

FAX 送信するためのプログラミング例を、FAX 送信プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 送信プログラム【SendFax.exe】の仕様、および操作方法については、*2.1 FAX 送信* をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥SendFax¥ ... FAX 送信プログラム 開発プロジェクト

■ FAX 送信

STARFAX Server SDK への FAX 送信命令は、送信命令フォルダに送信命令ファイルを置くことで行います。

そして、送信命令ファイルを作成は、STARFAX OCX の FAX 送信コントロール [SFCSSendFax] が行います。

作成したユーザープログラムで、FAX 送信が正常に動作していない場合は、**STARFAX Server SDK ログ管理プログラム** でイベントの内容を確認して下さい。そして、その内容を参考にしてプログラムを見直してみてください。

Form1. frm :

```
'*****  
' 送信ボタン: クリック  
'*****
```

```
Private Sub CommandOK_Click()
```

```
    Dim strCover As String  
    Dim strUserInfo As String  
    Dim strUserID As String  
    Dim SendDataTEMP As SendFaxListData  
    Dim strDocTEMP As String
```

```
    Dim sLine As Integer  
    Dim sPriority As Integer  
    Dim strTime As String
```

```
    Dim lRet As Long
```

```
    ' _____  
    ' 画面より値を取得
```

```
    ' 現在表示されている相手先
```

```
    With SendDataTEMP
```

```
        .strFax = TextFAX.Text  
        .strCompany = TextCOMPANY.Text  
        .strDivision = TextDIVISION.Text  
        .strPosition = TextPOSITION.Text  
        .strName = TextNAME.Text  
        .strTitle = ComboTITLE.Text  
        .strTelephone = TextTELEPHONE.Text  
        .strZipCode = TextZIPCODE.Text  
        .strAddress1 = TextADDRESS1.Text  
        .strAddress2 = TextADDRESS2.Text  
        .strFCode = TextFCODE.Text
```

```
    End With
```

```
    Select Case s_strCurSENDNUM
```

```
        Case 1
```

```
            s_SendData0 = SendDataTEMP
```

```
        Case 2
```

```
            s_SendData1 = SendDataTEMP
```

```
        Case 3
```

```

        s_SendData2 = SendDataTEMP
    Case 4
        s_SendData3 = SendDataTEMP
End Select

' 送付状
strCover = TextCOVER.Text

' 発信元情報
strUserInfo = CStr(ComboUSERPOS.ListIndex) + "," + ComboUSERINFO.Text
strUserID = TextUSERID.Text

' その他指定

' 回線指定
sLine = SFCSENDFAX_LINE_NO
Select Case Combo0_LINE.ListIndex
Case 1
    sLine = SFCSENDFAX_LINE_1
Case 2
    sLine = SFCSENDFAX_LINE_2
Case 3
    sLine = SFCSENDFAX_LINE_3
Case 4
    sLine = SFCSENDFAX_LINE_4
End Select

' 優先順位
sPriority = SFCSENDFAX_PRIORITY_NO
Select Case Combo0_PRI.ListIndex
Case 1
    sPriority = SFCSENDFAX_PRIORITY_0
Case 2
    sPriority = SFCSENDFAX_PRIORITY_1
Case 3
    sPriority = SFCSENDFAX_PRIORITY_2
Case 4
    sPriority = SFCSENDFAX_PRIORITY_3
Case 5
    sPriority = SFCSENDFAX_PRIORITY_4
Case 6
    sPriority = SFCSENDFAX_PRIORITY_5
Case 7
    sPriority = SFCSENDFAX_PRIORITY_6
Case 8

```

```

        sPriority = SFGSENDFA_PRIORITY_7
Case 9
        sPriority = SFGSENDFA_PRIORITY_8
Case 10
        sPriority = SFGSENDFA_PRIORITY_9
Case 11
        sPriority = SFGSENDFA_PRIORITY_10
Case 12
        sPriority = SFGSENDFA_PRIORITY_11
Case 13
        sPriority = SFGSENDFA_PRIORITY_12
Case 14
        sPriority = SFGSENDFA_PRIORITY_13
Case 15
        sPriority = SFGSENDFA_PRIORITY_14
Case 16
        sPriority = SFGSENDFA_PRIORITY_15
End Select

```

```

' 時刻指定
strTime = ""

```

```

If CheckO_CHOICETIME.Value = 1 Then
    strTime = Format$(Val(TextO_C_SETYEAR.Text), "0000") + _
        Format$(Val(TextO_C_SETMONTH.Text), "00") + _
        Format$(Val(TextO_C_SETDAY.Text), "00") + _
        Format$(Val(TextO_C_SETHOUR.Text), "00") + _
        Format$(Val(TextO_C_SETMINUTE.Text), "00") + _
        "00"
End If

```

```

' //////////////////////////////////////
' // (1) 相手先設定
' //
' // SFGSendFax コントロール 《 相手先情報リセット 》
' // Sub ResetSendInfo()
' //
' // SFGSendFax コントロール 《 相手先情報リスト追加設定 》
' // Function AddSendInfoEx(Fax As String, Company As String,
' // Division As String, Position As String,
' // Name As String, Title As String,
' // Telephone As String, ZipCode As String,
' // Address1 As String, Address2 As String,
' // FCode As String, FreeArea As String,
' // Speed As Integer, Comp As Integer,

```

```
'//          Ecm As Integer, Line As Integer,
'//          Priority As Integer, Time As String)as Long
'//
```

SFCSSendFax1.ResetSendInfo

```
For i = 1 To 4
    Select Case i
        Case 1
            SendDataTEMP = s_SendData0
        Case 2
            SendDataTEMP = s_SendData1
        Case 3
            SendDataTEMP = s_SendData2
        Case 4
            SendDataTEMP = s_SendData3
    End Select

    If SendDataTEMP.strFax <> "" Then

        IRet = SFCSSendFax1. AddSendInfoEx (
            SendDataTEMP.strFax, SendDataTEMP.strCompany, SendDataTEMP.strDivision, SendDataTEMP.strPosition,
            SendDataTEMP.strName, SendDataTEMP.strTitle, SendDataTEMP.strTelephone, SendDataTEMP.strZipCode,
            SendDataTEMP.strAddress1, SendDataTEMP.strAddress2, SendDataTEMP.strFCCode, "", SFCSSENDFAX_SPEED_AUTO,
            SFCSSENDFAX_COMP_AUTO, SFCSSENDFAX_ECM_AUTO, sLine, sPriority, strTime)
        If IRet <> SFCSSENDFAX_SUCCESS Then
            MsgBox "メモリエラーが発生しました", vbCritical, s_cstTITLE
            Exit Sub
        End If

    End If

Next i

' ///////////////////////////////////////////////////////////////////
' // (2) 送信原稿設定
' //
' // SFCSSendFax コントロール 《送信原稿リスト リセット》
' // Sub ResetDocInfo()
' //
' // SFCSSendFax コントロール 《送信原稿リスト 追加設定》
' // Function AddDocInfo(DocName As String)as Long
' //
```

SFCSSendFax1.ResetDocInfo

```
For i = 1 To 4
    Select Case i
        Case 1
            strDocTEMP = TextDOC1.Text
        Case 2
            strDocTEMP = TextDOC2.Text
        Case 3
            strDocTEMP = TextDOC3.Text
        Case 4
            strDocTEMP = TextDOC4.Text
    End Select

    If strDocTEMP <> "" Then

        IRet = SFCSSendFax1.AddDocInfo(strDocTEMP)
        If IRet <> SFCSENDFAX_SUCCESS Then
            MsgBox "メモリエラーが発生しました", vbCritical, s_cstTITLE
            Exit Sub
        End If

    End If

Next i

' //////////////////////////////////////
' // (3) 送信命令ファイル作成
' //
' // SFCSSendFax コントロール 《 FAX 送信命令ファイル作成 》
' // Function MakeSendMission(CoverName As String,
' //                           UserInfo As String,
' //                           UserID As String,
' //                           Account As String) as Long
' //

IRet = SFCSSendFax1.MakeSendMission(strCover, strUserInfo, strUserID, "")

If IRet = SFCSENDFAX_SUCCESS Then
    MsgBox "送信命令ファイルを作成しました。", vbOKOnly, s_cstTITLE
Else
    If IRet = SFCSENDFAX_ERR_NoMisFolder Then
        MsgBox "送信命令フォルダが存在しません", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDFAX_ERR_GetMisFolder Then
        MsgBox "送信命令フォルダの取得に失敗しました", vbExclamation, s_cstTITLE
    End If
End If
```



```

End If
If IRet = SFCSEND FAX_ERR_GetTempFolder Then
    MsgBox "一時フォルダの取得に失敗しました", vbExclamation, s_cstTITLE
End If
If IRet = SFCSEND FAX_ERR_GetTempFile Then
    MsgBox "一時ファイルの取得に失敗しました", vbExclamation, s_cstTITLE
End If
If IRet = SFCSEND FAX_ERR_MakeMisName Then
    MsgBox "送信命令ファイル名の作成に失敗しました", vbExclamation, s_cstTITLE
End If
If IRet = SFCSEND FAX_ERR_PARAM_INFO Then
    MsgBox "パラメータエラー: 送信命令ファイル作成情報が指定されていません。", vbExclamation, s_cstTITLE
End If
If IRet = SFCSEND FAX_ERR_PARAM_SENDCOUNT Then
    MsgBox "パラメータエラー: 相手先数の指定が 0 です。", vbExclamation, s_cstTITLE
End If
If IRet = SFCSEND FAX_ERR_PARAM_SENDINFO Then
    MsgBox "パラメータエラー: 相手先情報が指定されていません。", vbExclamation, s_cstTITLE
End If
If IRet = SFCSEND FAX_ERR_PARAM_FAX Then
    MsgBox "パラメータエラー: FAX 番号が指定されていません。", vbExclamation, s_cstTITLE
End If
If IRet = SFCSEND FAX_ERR_PARAM_DOCorCOVER Then
    MsgBox "パラメータエラー: 送信原稿ファイル、送付状ファイルが、ともに指定されていません。", vbExclamation, _
        s_cstTITLE
End If
If IRet = SFCSEND FAX_ERR_PARAM_DOCNAME Then
    MsgBox "パラメータエラー: 送信原稿ファイル名が指定されていません。", vbExclamation, s_cstTITLE
End If
End If

End Sub

```

2.2 FAX 情報の参照、削除

FAX 情報表示プログラム【MonLog.exe】は、FAX 情報の参照、削除を行うサンプルプログラムです。
本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonLog.exe ... FAX 情報表示プログラム
¥サンプル¥VB6 SP6¥MonLog¥ ... FAX 情報表示プログラム 開発プロジェクト

主な仕様、および操作方法は以下の通りです。

- ① STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
 - ① FAX 情報表示プログラム【MonLog.exe】を起動します。
(起動時の作業(カレント)フォルダの指定は特にありません)
 - ① ツリービューの「未送信」「送信」「受信」の各項目をクリックすることにより、リストビューにそれぞれの情報が表示されます。
 - ① リストビューに表示される「未送信」「送信」「受信」の情報は、それぞれ操作(A)-最新の情報に更新(F)メニュー、または F5 キーで最新の情報に更新されます。
 - ① リストビューに「未送信」「送信」または「受信」の情報が表示されている状態で、レコードを選択状態にして、編集(E)-削除(D)メニューで選択されたレコードを削除することができます。
但し、この段階では STARFAX Server SDK に対して削除命令を通知しただけです。
F5 キーでリストビューを更新していると、実際に削除された状態に表示が更新されます。
送信・受信・削除等の状態については、インデックスファイルに反映されますが、サービスからの通知はありません。そのため、ユーザープログラムからの動的な情報取得が必要になります。
削除が正常に動作していない場合は、STARFAX Server SDK ログ管理プログラム でイベントの内容を確認して下さい。なお、現在通信中の未送信情報を指定した場合はエラーとなり、削除できません。エラー内容は、STARFAX Server SDK ログ管理プログラム で参照できます。
 - ① リストビューに、「未送信」「送信」または、「受信」の情報が表示されている状態で、レコードを選択状態にして、ツール(T)-ビューア(V)メニューで、選択されたレコードを STARFAX Server SDK ビューアプログラムで表示することができます。
-

2.2 FAX 情報の参照、削除

2.2.1 送信情報を参照する

送信情報を参照するためのプログラミング例を、FAX 情報表示プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 情報表示プログラム【MonLog.exe】の仕様、および操作方法については、2.2 FAX 情報の参照、削除 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonLog¥ ... FAX 情報表示プログラム 開発プロジェクト

■ 送信情報の参照

送信情報の参照は、送信情報インデックスファイルを参照することにより行います。そして、送信情報インデックスファイルの読み込みは、STARFAX OCX の 送信情報の参照、削除コントロール [SFCSLogSned] が行います。

● 送信情報インデックスファイルの読み込み

frmMain.frm : Private Sub Form_Load()

~

```
' //////////////////////////////////////  
' // 各種ログの読み込み  
' //  
' // SFCSLogQue コントロール 《 未送信ログの読み込み 》  
' //   Function Load(Dlg As Integer)as Long  
' //  
' // SFCSLogSend コントロール 《 送信ログの読み込み 》  
' //   Function Load(Dlg As Integer)as Long  
' //  
' // SFCSLogRecv コントロール 《 受信ログの読み込み 》  
' //   Function Load(Dlg As Integer)as Long  
' //
```

```
SFCSLogQue1.Load (SFCSLOG_Dialog_TRUE)  
SFCSLogSend1.Load (SFCSLOG_Dialog_TRUE)  
SFCSLogRecv1.Load (SFCSLOG_Dialog_TRUE)
```

~

● 送信情報インデックスファイルの参照

frmMain.frm :

' 送信ログの表示

Private Sub ViewListViewSend()

' ステータスバー表示

sbStatusBar.SimpleText = "ログ件数:" & Str(SFCSLogSend1.GetRecCount())

' //////////////////////////////////////

' // 送信ログのリスト表示

' //

' // SFCSLogSend コントロール 《 送信ログのレコード数 取得 》

' // Function GetRecCount() as Long

' //

For i = 0 To (SFCSLogSend1.GetRecCount() - 1)

Set obItem = lvListView.ListItems.Add()

Dim strWork As String

' //////////////////////////////////////

' // 各項目の取得と表示

' //

' // SFCSLogSend コントロール 《 送信ログのレコード項目 文字列取得 》

' // Function GetRecItemString (Rec As Long, Item As Integer,

' // Get As String) as Long

' //

' 通信結果

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Return, strWork) = SFCSLOG_SUCCESS Then

If strWork = "0" Then

obItem.SmallIcon = 2

obItem.Text = "0"

Else

obItem.SmallIcon = 3

obItem.Text = strWork

End If

End If

’ 日付/時刻

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Date, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(1) = strWork
```

End If

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Time, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(2) = strWork
```

End If

’ 回線番号

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Line, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(3) = strWork
```

End If

’ アカウント

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Acount, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(4) = strWork
```

End If

’ 相手先

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_SendName, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(5) = strWork
```

End If

’ FAX

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_FaxNumber, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(6) = strWork
```

End If

’ ファイル名

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_FileName, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(7) = strWork
```

End If

’ ページ数

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Page, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(8) = strWork
```

End If

’ 通信速度

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_TermSpeed, strWork) = SFCSLOG_SUCCESS Then  
    Select Case strWork  
        Case "2"  
            strWork = "2400bps"  
        Case "4"  
            strWork = "4800bps"
```

```

Case "7"
    strWork = "7200bps (V. 29)"
Case "9"
    strWork = "9600bps (V. 29)"
Case "A"
    strWork = "7200bps (V. 17)"
Case "B"
    strWork = "9600bps (V. 17)"
Case "C"
    strWork = "12000bps"
Case "E"
    strWork = "14400bps"
Case Else
    strWork = ""
End Select
obItem.SubItems(9) = strWork
End If

```

' 通信時間

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_TotalTime, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(10) = strWork
End If

```

' 画品質

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_TermReso, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = "ノーマル"
        Case "F"
            strWork = "ファイン"
        Case "S"
            strWork = "スーパーファイン"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(11) = strWork
End If

```

' 符号化

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_TermComp, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "M"
            strWork = "MMR"
        Case "R"
            strWork = "MR"
    End Select
End If

```

```

        Case "H"
            strWork = "MH"
        Case Else
            strWork = ""
        End Select
        obItem.SubItems(12) = strWork
    End If

```

' 用紙サイズ

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_TermSize, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "A"
            strWork = "A4"
        Case "B"
            strWork = "B4"
        Case 3
            strWork = "A3"
        Case Else
            strWork = ""
        End Select
        obItem.SubItems(13) = strWork
    End If

```

' エラー訂正

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_TermECM, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "D"
            strWork = "なし"
        Case "E"
            strWork = "ECM"
        Case "S"
            strWork = "ECM"
        Case Else
            strWork = ""
        End Select
        obItem.SubItems(14) = strWork
    End If

```

' 回線指定

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEMEX_LineDes, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = "なし"
        Case "O"
            strWork = "回線1"
    End Select

```



```

        Case "1"
            strWork = "回線2"
        Case "2"
            strWork = "回線3"
        Case "3"
            strWork = "回線4"
        Case Else
            strWork = ""
        End Select
        obItem.SubItems(15) = strWork
    End If

```

’ 優先順位

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEMEX_Priority, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = ""
    End Select
    obItem.SubItems(16) = strWork
End If

```

’ リトライ回数

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Retry, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(17) = strWork
End If

```

’ Fコード番号

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_FCode, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(18) = strWork
End If

```

’ パスワード

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Password, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(19) = strWork
End If

```

’ 会社名

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Company, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(20) = strWork
End If

```

’ 所属名

```

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Division, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(21) = strWork
End If

```

' 役職名

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Position, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(22) = strWork  
End If
```

' 氏名

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Name, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(23) = strWork  
End If
```

' 郵便番号

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_ZipCode, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(24) = strWork  
End If
```

' 住所1

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Address1, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(25) = strWork  
End If
```

' 住所2

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Address2, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(26) = strWork  
End If
```

' 電話番号

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Telephone, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(27) = strWork  
End If
```

' FAX 番号

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Fax, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(28) = strWork  
End If
```

' 敬称

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_Title, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(29) = strWork  
End If
```

' ユーザ使用

```
If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_FreeArea, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(30) = strWork
```

End If

' 記録位置

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_UserInfo, strWork) = SFCSLOG_SUCCESS Then

 obItem.SubItems(31) = strWork

End If

' 自局電話番号

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_UserID, strWork) = SFCSLOG_SUCCESS Then

 obItem.SubItems(32) = strWork

End If

' ジョブ ID

If SFCSLogSend1.GetRecItemString(i, SFCSLOG_ITEM_JobID, strWork) = SFCSLOG_SUCCESS Then

 obItem.SubItems(33) = strWork

End If

Next

End Sub

● プログラムの終了処理

frmMain.frm : Private Sub Form_Unload(Cancel As Integer)
--

～

```
' ///////////////////////////////////////////  
' // 各種ログの読み込み 終了処理  
' //  
' // SFCSLogQue コントロール 《 未送信ログの読み込み 終了処理 》  
' // Sub Release()  
' //  
' // SFCSLogSend コントロール 《 送信ログの読み込み 終了処理 》  
' // Sub Release()  
' //  
' // SFCSLogRecv コントロール 《 受信ログの読み込み 終了処理 》  
' // Sub Release()  
' //
```

```
SFCSLogQue1.Release  
SFCSLogSend1.Release  
SFCSLogRecv1.Release
```

～

2.2 FAX 情報の参照、削除

2.2.2 受信情報を参照する

受信情報を参照する為のプログラミング例を、FAX 情報表示プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 情報表示プログラム【MonLog.exe】の仕様、および操作方法については、2.2 FAX 情報の参照、削除 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonLog¥ ... FAX 情報表示プログラム 開発プロジェクト

■ 受信情報の参照

受信情報の参照は、受信情報インデックスファイルを参照することにより行います。そして、受信情報インデックスファイルの読み込みは、STARFAX OCX の 受信情報の参照、削除コントロール [SFCSLogRecv] が行います。

● 受信情報インデックスファイルの読み込み

frmMain.frm : Private Sub Form_Load()

~

```
' ///////////////////////////////////////////  
' // 各種ログの読み込み  
' //  
' // SFCSLogQue コントロール 《 未送信ログの読み込み 》  
' //   Function Load(Dlg As Integer)as Long  
' //  
' // SFCSLogSend コントロール 《 送信ログの読み込み 》  
' //   Function Load(Dlg As Integer)as Long  
' //  
' // SFCSLogRecv コントロール 《 受信ログの読み込み 》  
' //   Function Load(Dlg As Integer)as Long  
' //
```

```
SFCSLogQue1.Load (SFCSLOG_Dialog_TRUE)  
SFCSLogSend1.Load (SFCSLOG_Dialog_TRUE)  
SFCSLogRecv1.Load (SFCSLOG_Dialog_TRUE)
```

~

● 受信情報インデックスファイルの参照

frmMain.frm :

' 受信ログの表示

Private Sub ViewListViewRecv()

' ステータスバー表示

sbStatusBar.SimpleText = "ログ件数:" & Str(SFCSLogRecv1.GetRecCount())

' //////////////////////////////////////

' // 受信ログのリスト表示

' //

' // SFCSLogRecv コントロール 《 受信ログのレコード数 取得 》

' // Function GetRecCount() as Long

' //

For i = 0 To (SFCSLogRecv1.GetRecCount() - 1)

Set obItem = lvListView.ListItems.Add()

Dim strWork As String

' //////////////////////////////////////

' // 各項目の取得と表示

' //

' // SFCSLogRecv コントロール 《 受信ログのレコード項目 文字列取得 》

' // Function GetRecItemString(Rec As Long, Item As Integer,

' // Get As String) as Long

' //

' 通信結果

If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Return, strWork) = SFCSLOG_SUCCESS Then

If strWork = "0" Then

obItem.SmallIcon = 2

obItem.Text = "0"

Else

obItem.SmallIcon = 3

obItem.Text = strWork

End If

End If

' 日付/時刻

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Date, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(1) = strWork
```

End If

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Time, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(2) = strWork
```

End If

' 回線番号

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Line, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(3) = strWork
```

End If

' アカウント

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Acount, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(4) = strWork
```

End If

' 相手先

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_SendName, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(5) = strWork
```

End If

' FAX

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_FaxNumber, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(6) = strWork
```

End If

' ファイル名

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_FileName, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(7) = strWork
```

End If

' ページ数

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Page, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(8) = strWork
```

End If

' 通信速度

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_TermSpeed, strWork) = SFCSLOG_SUCCESS Then  
    Select Case strWork  
        Case "2"  
            strWork = "2400bps"  
        Case "4"  
            strWork = "4800bps"
```



```

Case "7"
    strWork = "7200bps (V. 29)"
Case "9"
    strWork = "9600bps (V. 29)"
Case "A"
    strWork = "7200bps (V. 17)"
Case "B"
    strWork = "9600bps (V. 17)"
Case "C"
    strWork = "12000bps"
Case "E"
    strWork = "14400bps"
Case Else
    strWork = ""
End Select
obItem.SubItems(9) = strWork
End If

```

' 通信時間

```

If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_TotalTime, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(10) = strWork
End If

```

' 画品質

```

If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_TermReso, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = "ノーマル"
        Case "F"
            strWork = "ファイン"
        Case "S"
            strWork = "スーパーファイン"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(11) = strWork
End If

```

' 符号化

```

If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_TermComp, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "M"
            strWork = "MMR"
        Case "R"
            strWork = "MR"
    End Select
End If

```

```

        Case "H"
            strWork = "MH"
        Case Else
            strWork = ""
        End Select
        obItem.SubItems(12) = strWork
    End If

```

' 用紙サイズ

```

If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_TermSize, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "A"
            strWork = "A4"
        Case "B"
            strWork = "B4"
        Case 3
            strWork = "A3"
        Case Else
            strWork = ""
        End Select
        obItem.SubItems(13) = strWork
    End If

```

' エラー訂正

```

If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_TermECM, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "D"
            strWork = "なし"
        Case "E"
            strWork = "ECM"
        Case "S"
            strWork = "ECM"
        Case Else
            strWork = ""
        End Select
        obItem.SubItems(14) = strWork
    End If

```

’ 回線指定

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEMEX_LineDes, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = "なし"
        Case "0"
            strWork = "回線 1"
        Case "1"
            strWork = "回線 2"
        Case "2"
            strWork = "回線 3"
        Case "3"
            strWork = "回線 4"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(15) = strWork
End If
```

’ 優先順位

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEMEX_Priority, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = ""
    End Select
    obItem.SubItems(16) = strWork
End If
```

’ リトライ回数

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Retry, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(17) = strWork
End If
```

’ フコード番号

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_FCode, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(18) = strWork
End If
```

’ パスワード

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Password, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(19) = strWork
End If
```

’ 会社名

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Company, strWork) = SFCSLOG_SUCCESS Then
```

```
        obItem.SubItems(20) = strWork
```

```
End If
```

```
' 所属名
```

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Division, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(21) = strWork
```

```
End If
```

```
' 役職名
```

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Position, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(22) = strWork
```

```
End If
```

```
' 氏名
```

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Name, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(23) = strWork
```

```
End If
```

```
' 郵便番号
```

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_ZipCode, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(24) = strWork
```

```
End If
```

```
' 住所1
```

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Address1, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(25) = strWork
```

```
End If
```

```
' 住所2
```

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Address2, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(26) = strWork
```

```
End If
```

```
' 電話番号
```

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Telephone, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(27) = strWork
```

```
End If
```

```
' FAX 番号
```

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Fax, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(28) = strWork
```

```
End If
```

```
' 敬称
```

```
If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_Title, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(29) = strWork
```

End If

' ユーザ使用

If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_FreeArea, strWork) = SFCSLOG_SUCCESS Then
 obItem.SubItems(30) = strWork

End If

' 記録位置

If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_UserInfo, strWork) = SFCSLOG_SUCCESS Then
 obItem.SubItems(31) = strWork

End If

' 自局電話番号

If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_UserID, strWork) = SFCSLOG_SUCCESS Then
 obItem.SubItems(32) = strWork

End If

' ジョブ ID

If SFCSLogRecv1.GetRecItemString(i, SFCSLOG_ITEM_JobID, strWork) = SFCSLOG_SUCCESS Then
 obItem.SubItems(33) = strWork

End If

Next

End Sub

● プログラムの終了処理

frmMain.frm : Private Sub Form_Unload(Cancel As Integer)
--

～

```
' ///////////////////////////////////////////  
' // 各種ログの読み込み 終了処理  
' //  
' // SFCSLogQue コントロール 《 未送信ログの読み込み 終了処理 》  
' // Sub Release()  
' //  
' // SFCSLogSend コントロール 《 送信ログの読み込み 終了処理 》  
' // Sub Release()  
' //  
' // SFCSLogRecv コントロール 《 受信ログの読み込み 終了処理 》  
' // Sub Release()  
' //
```

```
SFCSLogQue1.Release  
SFCSLogSend1.Release  
SFCSLogRecv1.Release
```

～

2.2 FAX 情報の参照、削除

2.2.3 未送信情報を参照する

未送信情報を参照するためのプログラミング例を、FAX 情報表示プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 情報表示プログラム【MonLog.exe】の仕様、および操作方法については、2.2 FAX 情報の参照、削除 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonLog¥ ... FAX 情報表示プログラム 開発プロジェクト

■ 未送信情報の参照

未送信情報の参照は、未送信情報インデックスファイルを参照することにより行います。そして、未送信情報インデックスファイルの読み込みは、STARFAX OCX の 未送信情報の参照、削除コントロール [SFCSLogQue] が行います。

● 未送信情報インデックスファイルの読み込み

```
frmMain.frm : Private Sub Form_Load()
```

~

```
' //////////////////////////////////////  
' // 各種ログの読み込み  
' //  
' // SFCLogQue コントロール 《 未送信ログの読み込み 》  
' //   Function Load(Dlg As Integer)as Long  
' //  
' // SFCLogSend コントロール 《 送信ログの読み込み 》  
' //   Function Load(Dlg As Integer)as Long  
' //  
' // SFCLogRecv コントロール 《 受信ログの読み込み 》  
' //   Function Load(Dlg As Integer)as Long  
' //
```

```
SFCLogQue1.Load (SFCLOG_Dialog_TRUE)  
SFCLogSend1.Load (SFCLOG_Dialog_TRUE)  
SFCLogRecv1.Load (SFCLOG_Dialog_TRUE)
```

~

● 未送信情報インデックスファイルの参照

frmMain.frm :

' 未送信ログの表示

Private Sub ViewListViewQue()

' ステータスバー表示

sbStatusBar.SimpleText = "ログ件数:" & Str(SFCSLogQue1.GetRecCount())

' //////////////////////////////////////

' // 未送信ログのリスト表示

' //

' // SFCSLogQue コントロール 《 未送信ログのレコード数 取得 》

' // Function GetRecCount() as Long

' //

For i = 0 To (SFCSLogQue1.GetRecCount() - 1)

Set obItem = lvListView.ListItems.Add()

Dim strWork As String

' //////////////////////////////////////

' // 各項目の取得と表示

' //

' // SFCSLogQue コントロール 《 未送信ログのレコード項目 文字列取得 》

' // Function GetRecItemString(Rec As Long, Item As Integer,

' // Get As String) as Long

' //

' 通信結果

If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Return, strWork) = SFCSLOG_SUCCESS Then

If strWork = "0" Then

obItem.SmallIcon = 2

obItem.Text = "0"

Else

obItem.SmallIcon = 3

obItem.Text = strWork

End If

End If

’ 日付/時刻

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Date, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(1) = strWork
```

End If

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Time, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(2) = strWork
```

End If

’ 回線番号

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Line, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(3) = strWork
```

End If

’ アカウント

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Acount, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(4) = strWork
```

End If

’ 相手先

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_SendName, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(5) = strWork
```

End If

’ FAX

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_FaxNumber, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(6) = strWork
```

End If

’ ファイル名

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_FileName, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(7) = strWork
```

End If

’ ページ数

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Page, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(8) = strWork
```

End If

’ 通信速度

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_TermSpeed, strWork) = SFCSLOG_SUCCESS Then  
    Select Case strWork  
        Case "2"  
            strWork = "2400bps"  
        Case "4"  
            strWork = "4800bps"
```

```

Case "7"
    strWork = "7200bps (V. 29)"
Case "9"
    strWork = "9600bps (V. 29)"
Case "A"
    strWork = "7200bps (V. 17)"
Case "B"
    strWork = "9600bps (V. 17)"
Case "C"
    strWork = "12000bps"
Case "E"
    strWork = "14400bps"
Case Else
    strWork = ""
End Select
obItem.SubItems(9) = strWork
End If

```

' 通信時間

```

If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_TotalTime, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(10) = strWork
End If

```

' 画品質

```

If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_TermReso, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = "ノーマル"
        Case "F"
            strWork = "ファイン"
        Case "S"
            strWork = "スーパーファイン"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(11) = strWork
End If

```

’ 符号化

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_TermComp, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "M"
            strWork = "MMR"
        Case "R"
            strWork = "MR"
        Case "H"
            strWork = "MH"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(12) = strWork
End If
```

’ 用紙サイズ

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_TermSize, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "A"
            strWork = "A4"
        Case "B"
            strWork = "B4"
        Case 3
            strWork = "A3"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(13) = strWork
End If
```

’ エラー訂正

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_TermECM, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "D"
            strWork = "なし"
        Case "E"
            strWork = "ECM"
        Case "S"
            strWork = "ECM"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(14) = strWork
End If
```

’ 回線指定

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEMEX_LineDes, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = "なし"
        Case "0"
            strWork = "回線 1"
        Case "1"
            strWork = "回線 2"
        Case "2"
            strWork = "回線 3"
        Case "3"
            strWork = "回線 4"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(15) = strWork
End If
```

’ 優先順位

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEMEX_Priority, strWork) = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = ""
    End Select
    obItem.SubItems(16) = strWork
End If
```

’ リトライ回数

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Retry, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(17) = strWork
End If
```

’ フコード番号

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_FCode, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(18) = strWork
End If
```

’ パスワード

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Password, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(19) = strWork
End If
```

’ 会社名

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Company, strWork) = SFCSLOG_SUCCESS Then
```

```
        obItem.SubItems(20) = strWork
```

```
End If
```

```
    ' 所属名
```

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Division, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(21) = strWork
```

```
End If
```

```
    ' 役職名
```

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Position, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(22) = strWork
```

```
End If
```

```
    ' 氏名
```

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Name, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(23) = strWork
```

```
End If
```

```
    ' 郵便番号
```

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_ZipCode, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(24) = strWork
```

```
End If
```

```
    ' 住所1
```

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Address1, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(25) = strWork
```

```
End If
```

```
    ' 住所2
```

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Address2, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(26) = strWork
```

```
End If
```

```
    ' 電話番号
```

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Telephone, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(27) = strWork
```

```
End If
```

```
    ' FAX 番号
```

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Fax, strWork) = SFCSLOG_SUCCESS Then
```

```
    obItem.SubItems(28) = strWork
```

```
End If
```

```
    ' 敬称
```

```
If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_Title, strWork) = SFCSLOG_SUCCESS Then
```

```

        obItem.SubItems(29) = strWork
    End If

    ' ユーザ使用
    If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_FreeArea, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(30) = strWork
    End If

    ' 記録位置
    If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_UserInfo, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(31) = strWork
    End If

    ' 自局電話番号
    If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_UserID, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(32) = strWork
    End If

    ' ジョブID
    If SFCSLogQue1.GetRecItemString(i, SFCSLOG_ITEM_JobID, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(33) = strWork
    End If

Next
End Sub

```

● プログラムの終了処理

frmMain.frm : Private Sub Form_Unload(Cancel As Integer)
--

～

```
' ///////////////////////////////////////////  
' // 各種ログの読み込み 終了処理  
' //  
' // SFCSLogQue コントロール 《 未送信ログの読み込み 終了処理 》  
' // Sub Release()  
' //  
' // SFCSLogSend コントロール 《 送信ログの読み込み 終了処理 》  
' // Sub Release()  
' //  
' // SFCSLogRecv コントロール 《 受信ログの読み込み 終了処理 》  
' // Sub Release()  
' //
```

SFCSLogQue1.Release

SFCSLogSend1.Release

SFCSLogRecv1.Release

～

2.2 FAX 情報の参照、削除

2.2.4 送信情報を削除する

送信情報を削除するためのプログラミング例を、FAX 情報表示プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 情報表示プログラム【MonLog.exe】の仕様、および操作方法については、2.2 FAX 情報の参照、削除 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonLog¥ ... FAX 情報表示プログラム 開発プロジェクト

■ 送信情報の削除

送信情報の削除は、送信情報削除命令フォルダに送信情報削除命令ファイルを置いて、STARFAX Server SDK に対して命令することで行います。そして、送信情報削除命令ファイルの作成は STARFAX Server SDK OCX の送信情報の参照、削除コントロール [SFCSLogSned] が行います。

作成したユーザープログラムで、送信情報の削除が正常に動作していない場合は、STARFAX Server SDK ログ管理プログラム でイベントの内容を確認して下さい。そして、その内容を参考にし、プログラムを見直してみてください。

frmMain. frm :

送信情報削除命令ファイル作成

Private Sub DeleteSend()

Dim strDeletItem As String

If lvListView.ListItems.Count > 0 Then

```
' //////////////////////////////////
' // 削除情報の初期化
' //
' // SFCSLogSend コントロール 《 削除ジョブ ID リスト リセット 》
' // Sub ResetDelInfo()
' //
```

SFCSLogSend1.ResetDelInfo

For i = 1 To lvListView.ListItems.Count

If lvListView.ListItems.Item(i).Selected = True Then

strDeletItem = lvListView.ListItems.Item(i).ListSubItems(33)

```
' //////////////////////////////////
' // 削除情報の追加
' //
' // SFCSLogSend コントロール 《 削除ジョブ ID リスト 追加設定 》
' // Function AddDelInfo(JobID As String) as Long
' //
```

Call SFCSLogSend1.AddDelInfo(strDeletItem)

End If

Next

```
' //////////////////////////////////
' // 削除命令ファイル作成
' //
' // SFCSLogSend コントロール 《 送信情報削除命令ファイル作成 》
' // Function MakeDelMission() as Long
' //
```

Call **SFCSLogSend1.MakeDeIMission**

End If

End Sub

2.2 FAX 情報の参照、削除

2.2.5 受信情報を削除する

受信情報を削除するのプログラミング例を、FAX 情報表示プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 情報表示プログラム【MonLog.exe】の仕様、および操作方法については、2.2 FAX 情報の参照、削除 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonLog¥ ... FAX 情報表示プログラム 開発プロジェクト

■ 受信情報の削除

受信情報の削除は、受信情報削除命令フォルダに受信情報削除命令ファイルを置いて、STARFAX Server SDK に対して命令することで行います。そして、受信情報削除命令ファイルの作成は、STARFAX Server SDK OCX の受信情報の参照、削除コントロール [SFCSLogRecv] が行います。

作成したユーザープログラムで、受信情報の削除が正常に動作していない場合は、STARFAX Server SDK ログ管理プログラム でイベントの内容を確認して下さい。そして、その内容を参考にし、プログラムを見直してみてください。

frmMain.frm :

' 受信情報削除命令ファイル作成

Private Sub DeleteRecv()

Dim strDeletItem As String

If lvListView.ListItems.Count < 0 Then

' ///////////////////////////////////
' // 削除情報の初期化
' //
' // SFCSLogRecv コントロール 《 削除ジョブ ID リスト リセット 》
' // Sub ResetDocInfo()
' //

SFCSLogRecv1.ResetDelInfo

For i = 1 To lvListView.ListItems.Count

If lvListView.ListItems.Item(i).Selected = True Then

strDeletItem = lvListView.ListItems.Item(i).ListSubItems(33)

' ///////////////////////////////////
' // 削除情報の追加
' //
' // SFCSLogRecv コントロール 《 削除ジョブ ID リスト 追加設定 》
' // Function AddDelInfo(JobID As String) as Long
' //

Call **SFCSLogRecv1.AddDelInfo**(strDeletItem)

End If

Next

' ///////////////////////////////////
' // 削除命令ファイル作成
' //
' // SFCSLogRecv コントロール 《 受信情報削除命令ファイル作成 》
' // Function MakeDelMission() as Long
' //

Call **SFCSLogRecv1.MakeDelMission**

End If

End Sub

2.2 FAX 情報の参照、削除

2.2.6 未送信情報を削除する

未送信情報を削除するためのプログラミング例を、FAX 情報表示プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 情報表示プログラム【MonLog.exe】の仕様、および操作方法については、2.2 FAX 情報の参照、削除 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonLog¥ ... FAX 情報表示プログラム 開発プロジェクト

■ 未送信情報の削除

未送信情報の削除は、未送信情報削除命令フォルダに未送信情報削除命令ファイルを置いて、STARFAX Server SDK に対して命令することで行います。そして、未送信情報削除命令ファイルの作成は、STARFAX OCX の 未送信情報の参照、削除コントロール [SFCSLogQue] が行います。

作成したユーザープログラムで、未送信情報の削除が正常に動作していない場合は、STARFAX Server SDK ログ管理プログラム でイベントの内容を確認して下さい。そして、その内容を参考にし、プログラムを見直してみてください。

frmMain.frm :

' 未送信情報削除命令ファイル作成

Private Sub DeleteQue()

Dim strDeletItem As String

If lvListView.ListItems.Count > 0 Then

' ///////////////////////////////////
' // 削除情報の初期化
' //
' // SFCSLogQue コントロール 《 削除ジョブ ID リスト リセット 》
' // Sub ResetDelInfo()
' //

SFCSLogQue1.ResetDelInfo

For i = 1 To lvListView.ListItems.Count

If lvListView.ListItems.Item(i).Selected = True Then

strDeletItem = lvListView.ListItems.Item(i).ListSubItems(33)

' ///////////////////////////////////
' // 削除情報の追加
' //
' // SFCSLogQue コントロール 《 削除ジョブ ID リスト 追加設定 》
' // Function AddDelInfo(JobID As String) as Long
' //

Call **SFCSLogQue1.AddDelInfo**(strDeletItem)

End If

Next

' ///////////////////////////////////
' // 削除命令ファイル作成
' //
' // SFCSLogQue コントロール 《 未送信情報削除命令ファイル作成 》
' // Function MakeDelMission() as Long
' //

Call **SFCSLogQue1.MakeDeIMission**

End If

End Sub

2.3 FAX 情報の追加、削除状況の取得

FAX 情報の追加、削除状況の取得プログラム【MonDiff.exe】は、FAX 情報の追加、削除状況の表示を行うサンプルプログラムです。このプログラムは、コンソールプログラムです。本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonDiff.exe ... FAX 情報の追加、削除状況の取得プログラム

¥サンプル¥VB6 SP6¥MonDiff¥ ... FAX 情報の追加、削除状況の取得プログラム 開発プロジェクト

主な仕様、および操作方法是以下の通りです。

- ① STARFAX Server SDK 環境設定プログラム-オプション-通知ファイル で通知したい項目を設定します。
- ② STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
- ③ FAX 情報の追加、削除状況の取得プログラム【MonDiff.exe】を起動します。
(起動時の作業(カレント)フォルダの指定は特にありません)
- ④ 約5秒間隔で以下の通知の存在をチェックして、各リストに内容が出力されます。
 - ・ 送信情報追加済み通知
 - ・ 受信情報追加済み通知
 - ・ 送信情報削除済み通知
 - ・ 受信情報削除済み通知

但し、①で設定されていない項目は通知されません。

2.3 FAX 情報の追加、削除状況の取得

2.3.1 送信情報が追加された事を知る

送信情報が追加された事を知るためのプログラミング例を、FAX 情報の追加、削除状況の取得プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 情報の追加、削除状況の取得プログラム【MonDiff.exe】の仕様、および操作方法については、
2.3 FAX 情報の追加、削除状況の取得 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

サンプル¥VB6 SP6¥MonDiff¥ ... FAX 情報の追加、削除状況の取得プログラム 開発プロジェクト

■ 送信情報が追加された事を知る

送信情報が追加された事を知る方法は、送信情報追加済み通知ファイルを参照することにより行います。
そして、送信情報追加済み通知ファイルの読み込みは、STARFAX OCX の FAX 情報の追加、削除状況の取得コントロール [SFCSLogDiff] が行います。

Form1. frm :

' 送信情報追加済み通知処理

Private Sub SendAddNotiProc()

Dim bGet As Boolean

bGet = True

Do While (bGet = True)

' //////////////////////////////////////

' // 送信情報追加済み通知 の取得

' //

' // SFCSLogDiff コントロール 《 送信情報追加済み通知 取得 》

' // Function GetSendAddNoti() as Long

' //

If SFCSLogDiff1.GetSendAddNoti() <> SFCSLOGDIFF_GS_NoNoti Then

Set obItem = ListViewSENDADD.ListItems.Add(1)

Dim strWork As String

' //////////////////////////////////////

' // 各項目の取得と表示

' //

' // SFCSLogDiff コントロール 《 送信情報追加済み通知のレコード項目 文字列取得 》

' // Function GetSendRecItemString(Item As Integer,

' // Get As String) as Long

' //

' 通信結果

If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Return, strWork) = SFCSLOG_SUCCESS Then

If strWork = "0" Then

obItem.Text = "0"

Else

obItem.Text = strWork

End If

End If

' 日付/時刻

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Date, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(1) = strWork
```

```
End If
```

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Time, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(2) = strWork
```

```
End If
```

```
' 回線番号
```

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Line, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(3) = strWork
```

```
End If
```

```
' アカウント
```

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Acount, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(4) = strWork
```

```
End If
```

```
' 相手先
```

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_SendName, strWork) _  
    = SFCSLOG_SUCCESS Then  
    obItem.SubItems(5) = strWork
```

```
End If
```

```
' FAX
```

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_FaxNumber, strWork) _  
    = SFCSLOG_SUCCESS Then  
    obItem.SubItems(6) = strWork
```

```
End If
```

```
' ファイル名
```

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_FileName, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(7) = strWork
```

```
End If
```

```
' ページ数
```

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Page, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(8) = strWork
```

```
End If
```

```
' 通信速度
```

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_TermSpeed, strWork) _  
    = SFCSLOG_SUCCESS Then  
    Select Case strWork  
        Case "2"  
            strWork = "2400bps"
```

```

        Case "4"
            strWork = "4800bps"
        Case "7"
            strWork = "7200bps (V. 29)"
        Case "9"
            strWork = "9600bps (V. 29)"
        Case "A"
            strWork = "7200bps (V. 17)"
        Case "B"
            strWork = "9600bps (V. 17)"
        Case "C"
            strWork = "12000bps"
        Case "E"
            strWork = "14400bps"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(9) = strWork
End If

```

' 通信時間

```

If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_TotalTime, strWork) _
    = SFCSLOG_SUCCESS Then
    obItem.SubItems(10) = strWork
End If

```

' 画品質

```

If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_TermReso, strWork) _
    = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = "ノーマル"
        Case "F"
            strWork = "ファイン"
        Case "S"
            strWork = "スーパーファイン"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(11) = strWork
End If

```

' 符号化

```

If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_TermComp, strWork) _
    = SFCSLOG_SUCCESS Then

```

```

Select Case strWork
    Case "M"
        strWork = "MMR"
    Case "R"
        strWork = "MR"
    Case "H"
        strWork = "MH"
    Case Else
        strWork = ""
End Select
obItem.SubItems(12) = strWork
End If

```

' 用紙サイズ

```

If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_TermSize, strWork) _
    = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "A"
            strWork = "A4"
        Case "B"
            strWork = "B4"
        Case 3
            strWork = "A3"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(13) = strWork
End If

```

' エラー訂正

```

If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_TermECM, strWork) _
    = SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "D"
            strWork = "なし"
        Case "E"
            strWork = "ECM"
        Case "S"
            strWork = "ECM"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(14) = strWork
End If

```

’ 回線指定

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEMEX_LineDes, strWork) _  
    = SFCSLOG_SUCCESS Then  
    Select Case strWork  
        Case "N"  
            strWork = "なし"  
        Case "0"  
            strWork = "回線 1"  
        Case "1"  
            strWork = "回線 2"  
        Case "2"  
            strWork = "回線 3"  
        Case "3"  
            strWork = "回線 4"  
        Case Else  
            strWork = ""  
    End Select  
    obItem.SubItems(15) = strWork  
End If
```

’ 優先順位

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEMEX_Priority, strWork) _  
    = SFCSLOG_SUCCESS Then  
    Select Case strWork  
        Case "N"  
            strWork = ""  
    End Select  
    obItem.SubItems(16) = strWork  
End If
```

’ リトライ回数

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Retry, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(17) = strWork  
End If
```

’ Fコード番号

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_FCode, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(18) = strWork  
End If
```

’ パスワード

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Password, strWork) _  
    = SFCSLOG_SUCCESS Then  
    obItem.SubItems(19) = strWork  
End If
```


’ 会社名

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Company, strWork) _  
    = SFCSLOG_SUCCESS Then  
    obItem.SubItems(20) = strWork  
End If
```

’ 所属名

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Division, strWork) _  
    = SFCSLOG_SUCCESS Then  
    obItem.SubItems(21) = strWork  
End If
```

’ 役職名

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Position, strWork) _  
    = SFCSLOG_SUCCESS Then  
    obItem.SubItems(22) = strWork  
End If
```

’ 氏名

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Name, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(23) = strWork  
End If
```

’ 郵便番号

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_ZipCode, strWork) _  
    = SFCSLOG_SUCCESS Then  
    obItem.SubItems(24) = strWork  
End If
```

’ 住所1

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Address1, strWork) _  
    = SFCSLOG_SUCCESS Then  
    obItem.SubItems(25) = strWork  
End If
```

’ 住所2

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Address2, strWork) _  
    = SFCSLOG_SUCCESS Then  
    obItem.SubItems(26) = strWork  
End If
```

’ 電話番号

```
If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Telephone, strWork) _  
    = SFCSLOG_SUCCESS Then
```

```

        obItem.SubItems(27) = strWork
    End If

    ' FAX 番号
    If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Fax, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(28) = strWork
    End If

    ' 敬称
    If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_Title, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(29) = strWork
    End If

    ' ユーザ使用
    If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_FreeArea, strWork) _
        = SFCSLOG_SUCCESS Then
        obItem.SubItems(30) = strWork
    End If

    ' 記録位置
    If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_UserInfo, strWork) _
        = SFCSLOG_SUCCESS Then
        obItem.SubItems(31) = strWork
    End If

    ' 自局電話番号
    If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_UserID, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(32) = strWork
    End If

    ' ジョブ ID
    If SFCSLogDiff1.GetSendRecItemString(SFCSLOG_ITEM_JobID, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(33) = strWork
    End If

Else
    bGet = False
End If

Loop

End Sub

```

2.3 FAX 情報の追加、削除状況の取得

2.3.2 受信情報が追加された事を知る

受信情報が追加された事を知るためのプログラミング例を、FAX 情報の追加、削除状況の取得プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 情報の追加、削除状況の取得プログラム【MonDiff.exe】の仕様、および操作方法については、*2.3 FAX 情報の追加、削除状況の取得* をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥¥MonDiff¥ ... FAX 情報の追加、削除状況の取得プログラム 開発プロジェクト

■ 受信情報が追加された事を知る

受信情報が追加された事を知る方法は、受信情報追加済み通知ファイルを参照することにより行います。そして、受信情報追加済み通知ファイルの読み込みは、STARFAX OCX の FAX 情報の追加、削除状況の取得コントロール [SFCSLogDiff] が行います。

Form1.frm :

受信情報追加済み通知処理

Private Sub RecvAddNotiProc()

Dim bGet As Boolean

bGet = True

Do While (bGet = True)

```
' //////////////////////////////////
' // 受信情報追加済み通知 の取得
' //
' // SFCSLogDiff コントロール 《 受信情報追加済み通知 取得 》
' // Function GetRecvAddNoti () as Long
' //
```

If SFCSLogDiff1.GetRecvAddNoti () <> SFCSLOGDIFF_GS_NoNoti Then

Set obItem = ListViewRECVADD.ListItems.Add(1)
Dim strWork As String

```
' //////////////////////////////////
' // 各項目の取得と表示
' //
' // SFCSLogDiff コントロール 《 受信情報追加済み通知のレコード項目 文字列取得 》
' // Function GetRecvRecItemString(Item As Integer,
' //                               Get As String) as Long
' //
```

通信結果

If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Return, strWork) = SFCSLOG_SUCCESS Then

If strWork = "0" Then
obItem.Text = "0"

Else
obItem.Text = strWork

End If

End If

日付/時刻

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Date, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(1) = strWork
```

```
End If
```

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Time, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(2) = strWork
```

```
End If
```

```
' 回線番号
```

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Line, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(3) = strWork
```

```
End If
```

```
' アカウント
```

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Account, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(4) = strWork
```

```
End If
```

```
' 相手先
```

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_SendName, strWork) = _  
    SFCSLOG_SUCCESS Then  
    obItem.SubItems(5) = strWork
```

```
End If
```

```
' FAX
```

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_FaxNumber, strWork) = _  
    SFCSLOG_SUCCESS Then  
    obItem.SubItems(6) = strWork
```

```
End If
```

```
' ファイル名
```

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_FileName, strWork) = _  
    SFCSLOG_SUCCESS Then  
    obItem.SubItems(7) = strWork
```

```
End If
```

```
' ページ数
```

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Page, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(8) = strWork
```

```
End If
```

```
' 通信速度
```

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_TermSpeed, strWork) = _  
    SFCSLOG_SUCCESS Then  
    Select Case strWork  
        Case "2"
```

```

        strWork = "2400bps"
    Case "4"
        strWork = "4800bps"
    Case "7"
        strWork = "7200bps (V. 29)"
    Case "9"
        strWork = "9600bps (V. 29)"
    Case "A"
        strWork = "7200bps (V. 17)"
    Case "B"
        strWork = "9600bps (V. 17)"
    Case "C"
        strWork = "12000bps"
    Case "E"
        strWork = "14400bps"
    Case Else
        strWork = ""
    End Select
    obItem.SubItems(9) = strWork
End If

```

' 通信時間

```

If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_TotalTime, strWork) = _
    SFCSLOG_SUCCESS Then
    obItem.SubItems(10) = strWork
End If

```

' 画品質

```

If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_TermReso, strWork) = _
    SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = "ノーマル"
        Case "F"
            strWork = "ファイン"
        Case "S"
            strWork = "スーパーファイン"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(11) = strWork
End If

```

' 符号化

```

If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_TermComp, strWork) = _

```

```

SFCLOG_SUCCESS Then
Select Case strWork
    Case "M"
        strWork = "MMR"
    Case "R"
        strWork = "MR"
    Case "H"
        strWork = "MH"
    Case Else
        strWork = ""
End Select
obItem.SubItems(12) = strWork
End If

```

' 用紙サイズ

```

If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_TermSize, strWork) = _
SFCLOG_SUCCESS Then
Select Case strWork
    Case "A"
        strWork = "A4"
    Case "B"
        strWork = "B4"
    Case 3
        strWork = "A3"
    Case Else
        strWork = ""
End Select
obItem.SubItems(13) = strWork
End If

```

' エラー訂正

```

If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_TermECM, strWork) = _
SFCLOG_SUCCESS Then
Select Case strWork
    Case "D"
        strWork = "なし"
    Case "E"
        strWork = "ECM"
    Case "S"
        strWork = "ECM"
    Case Else
        strWork = ""
End Select
obItem.SubItems(14) = strWork
End If

```

’ 回線指定

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEMEX_LineDes, strWork) = _
    SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = "なし"
        Case "0"
            strWork = "回線 1"
        Case "1"
            strWork = "回線 2"
        Case "2"
            strWork = "回線 3"
        Case "3"
            strWork = "回線 4"
        Case Else
            strWork = ""
    End Select
    obItem.SubItems(15) = strWork
End If
```

’ 優先順位

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEMEX_Priority, strWork) = _
    SFCSLOG_SUCCESS Then
    Select Case strWork
        Case "N"
            strWork = ""
    End Select
    obItem.SubItems(16) = strWork
End If
```

’ リトライ回数

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Retry, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(17) = strWork
End If
```

’ Fコード番号

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_FCode, strWork) = SFCSLOG_SUCCESS Then
    obItem.SubItems(18) = strWork
End If
```

’ パスワード

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Password, strWork) = _
    SFCSLOG_SUCCESS Then
    obItem.SubItems(19) = strWork
End If
```


’ 会社名

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Company, strWork) = _  
    SFCSLOG_SUCCESS Then  
    obItem.SubItems(20) = strWork  
End If
```

’ 所属名

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Division, strWork) = _  
    SFCSLOG_SUCCESS Then  
    obItem.SubItems(21) = strWork  
End If
```

’ 役職名

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Position, strWork) = _  
    SFCSLOG_SUCCESS Then  
    obItem.SubItems(22) = strWork  
End If
```

’ 氏名

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Name, strWork) = SFCSLOG_SUCCESS Then  
    obItem.SubItems(23) = strWork  
End If
```

’ 郵便番号

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_ZipCode, strWork) = _  
    SFCSLOG_SUCCESS Then  
    obItem.SubItems(24) = strWork  
End If
```

’ 住所1

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Address1, strWork) = _  
    SFCSLOG_SUCCESS Then  
    obItem.SubItems(25) = strWork  
End If
```

’ 住所2

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Address2, strWork) = _  
    SFCSLOG_SUCCESS Then  
    obItem.SubItems(26) = strWork  
End If
```

’ 電話番号

```
If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Telephone, strWork) = _  
    SFCSLOG_SUCCESS Then
```

```

        obItem.SubItems(27) = strWork
    End If

    ' FAX 番号
    If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Fax, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(28) = strWork
    End If

    ' 敬称
    If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_Title, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(29) = strWork
    End If

    ' ユーザ使用
    If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_FreeArea, strWork) = _
        SFCSLOG_SUCCESS Then
        obItem.SubItems(30) = strWork
    End If

    ' 記録位置
    If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_UserInfo, strWork) = _
        SFCSLOG_SUCCESS Then
        obItem.SubItems(31) = strWork
    End If

    ' 自局電話番号
    If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_UserID, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(32) = strWork
    End If

    ' ジョブ ID
    If SFCSLogDiff1.GetRecvRecItemString(SFCSLOG_ITEM_JobID, strWork) = SFCSLOG_SUCCESS Then
        obItem.SubItems(33) = strWork
    End If

Else
    bGet = False
End If

Loop

End Sub

```

2.3 FAX 情報の追加、削除状況の取得

2.3.3 送信情報が削除された事を知る

送信情報が削除された事を知るためのプログラミング例を、FAX 情報の追加、削除状況の取得プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 情報の追加、削除状況の取得プログラム【MonDiff.exe】の仕様、および操作方法については、
2.3 FAX 情報の追加、削除状況の取得 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonDiff¥ ... FAX 情報の追加、削除状況の取得プログラム 開発プロジェクト

■ 送信情報が削除された事を知る

送信情報が削除された事を知る方法は、送信情報削除済み通知ファイルを参照することにより行います。そして、送信情報削除済み通知ファイルの読み込みは、STARFAX OCX の FAX 情報の追加、削除状況の取得コントロール [SFCSLogDiff] が行います。

Form1.frm :

' 送信情報削除済み通知処理

Private Sub SendDelNotiProc()

Dim strJobID As String

Dim bGet As Boolean

bGet = True

Do While (bGet = True)

' //////////////////////////////////////
' // 送信情報削除済み通知 の取得
' //
' // SFCSLogDiff コントロール 《 送信情報削除済み通知 取得 》
' // Function GetSendDelNoti (GetJobID As String) as Long
' //
' // 注意事項
' // 戻り値 で 【SFCSLOGDIFF_GS_GetAndNext (2) : 通知を読み
' // 込みました。そして次の通知もあります】 が 返された場合、
' // GetSendDelNoti メソッド を 再度呼び出して、次の
' // 【削除済みジョブ ID】を取得しないと、その情報は失われて
' // しまいます。
' //

If SFCSLogDiff1.GetSendDelNoti (strJobID) <> SFCSLOGDIFF_GS_NoNoti Then

Set obItem = ListViewSENDEL.ListItems.Add(1)

obItem.Text = strJobID

Else

bGet = False

End If

Loop

End Sub

2.3 FAX 情報の追加、削除状況の取得

2.3.4 受信情報が削除された事を知る

受信情報が削除された事を知るためのプログラミング例を、FAX 情報の追加、削除状況の取得プログラム 開発プロジェクトのソースファイルを元にご説明します。

FAX 情報の追加、削除状況の取得プログラム【MonDiff.exe】の仕様、および操作方法については、

2.3 FAX 情報の追加、削除状況の取得 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonDiff¥ ... FAX 情報の追加、削除状況の取得プログラム 開発プロジェクト

■ 受信情報が削除された事を知る

受信情報が削除された事を知る方法は、受信情報削除済み通知ファイルを参照することにより行います。そして、受信情報削除済み通知ファイルの読み込みは、STARFAX OCX の FAX 情報の追加、削除状況の取得コントロール [SFCSLogDiff] が行います。

Form1.frm :

' 受信情報削除済み通知処理

Private Sub RecvDelNotiProc()

Dim strJobID As String

Dim bGet As Boolean

bGet = True

Do While (bGet = True)

' //////////////////////////////////////

' // 受信情報削除済み通知 の取得

' //

' // SFCSLogDiff コントロール 《 受信情報削除済み通知 取得 》

' // Function GetRecvDelNoti (GetJobID As String) as Long

' //

' // 注意事項

' // 戻り値 で 【SFCSLOGDIFF_GS_GetAndNext (2) : 通知を読み

' // 込みました。そして次の通知もあります】 が 返された場合、

' // GetRecvDelNoti メソッド を 再度呼び出して、次の

' // 【削除済みジョブ ID】を取得しないと、その情報は失われて

' // しまいます。

' //

If SFCSLogDiff1.GetRecvDelNoti (strJobID) <> SFCSLOGDIFF_GS_NoNoti Then

Set obItem = ListViewRECVDL.ListItems.Add(1)

obItem.Text = strJobID

Else

bGet = False

End If

Loop

End Sub

2.4 ファイル印刷

ファイル印刷プログラム【PrtFile.exe】は、ファイルの印刷を行うサンプルプログラムです。本CD-ROMの以下の位置に入っています。

¥サンプル¥VB6 SP6¥PrtFile.exe ... ファイル印刷プログラム
¥サンプル¥VB6 SP6¥PrtFile¥ ... ファイル印刷プログラム 開発プロジェクト

主な仕様、および操作方法是以下の通りです。

- ① STARFAX Server SDK 環境設定プログラム-オプション-カレントプリンタ で印刷したいプリンタを設定します。
 - ② STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
 - ③ ファイル印刷プログラム【PrtFile.exe】を起動します。
(起動時の作業(カレント)フォルダの指定は特にありません)
 - ④ 原稿を指定します。
 - ・ 最大4つのファイルを指定でき、印刷時に連結して印刷されます。
(これは、このサンプルプログラムの仕様です)
 - ・ 最低限、1つの原稿ファイルを指定する必要があります。
 - ・ 指定できるファイル形式は以下の通りです。
 - ・ TIFF 形式 圧縮なし、修正 CCITT MH 圧縮、CCITT G3 MH 圧縮、CCITT G3 MR 圧縮
PackBits 圧縮、Class F 圧縮、G4 圧縮、JPEG 圧縮
 - ・ BMP ファイル
 - ・ PCX ファイル
 - ・ DCX ファイル
 - ・ JPEG ファイル
 - ・ テキストファイル
 - ・ FAX ファイル
 - ・ LNK ファイル
 - ⑤ 印刷(P)ボタンを押して、ファイル印刷を行います。
この後、ファイル印刷が正常に動作していない場合は、STARFAX Server SDK ログ管理プログラム でイベントの内容を確認して下さい。
-

2.4 ファイル印刷

2.4.1 ファイルを印刷する

ファイルを印刷するためのプログラミング例を、ファイル印刷プログラム 開発プロジェクトのソースファイルを元にご説明します。

ファイル印刷プログラム【PrtFile.exe】の仕様、および操作方法については、2.4 ファイル印刷をご覧ください。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥PrtFile¥ ... ファイル印刷プログラム 開発プロジェクト

■ ファイル印刷

STARFAX Server SDK へのファイル印刷命令は、印刷命令フォルダに印刷命令ファイルを置くことで行います。そして、印刷命令ファイルの作成は、STARFAX OCX の ファイル印刷コントロール [SFCSPrtFile] が行います。

作成したユーザープログラムで、ファイル印刷が正常に動作していない場合は、STARFAX Server SDK ログ管理プログラム でイベントの内容を確認して下さい。そして、その内容を参考にしてプログラムを見直してみてください。

Form1.frm :

```
'*****  
' ファイル印刷ボタン: クリック  
'*****
```

```
Private Sub CommandOK_Click()
```

```
    Dim strDocTEMP As String
```

```
    Dim lRet As Long
```

```
    ' ///////////////////////////////////  
' // (1) 印刷原稿設定  
' //  
' // SFCSPrtFile コントロール 《印刷原稿リセット》  
' // Sub ResetDocInfo()  
' //  
' // SFCSPrtFile コントロール 《印刷原稿リスト追加設定》  
' // Function AddDocInfo(DocName As String) as Long  
' //
```

SFCSPrtFile1.ResetDocInfo

```
For i = 1 To 4
```

```
    Select Case i
```

```
        Case 1
```

```
            strDocTEMP = TextDOC1.Text
```

```
        Case 2
```

```
            strDocTEMP = TextDOC2.Text
```

```
        Case 3
```

```
            strDocTEMP = TextDOC3.Text
```

```
        Case 4
```

```
            strDocTEMP = TextDOC4.Text
```

```
    End Select
```

```
    If strDocTEMP <> "" Then
```

```
        lRet = SFCSPrtFile1.AddDocInfo(strDocTEMP)
```

```
        If lRet <> SFCSPRTFILE_SUCCESS Then
```

```
            MsgBox "メモリエラーが発生しました", vbCritical, s_cstTITLE
```

```
            Exit Sub
```

```
        End If
```

```
    End If
```

Next i

```
' //////////////////////////////////////  
' // (2) 印刷命令ファイル作成  
' //  
' // SFCSPrtFile コントロール 《印刷命令ファイル作成》  
' // Function MakePrintMission() as Long  
' //
```

IRet = **SFCSPrtFile1.MakePrintMission()**

```
If IRet = SFCSPRTFILE_SUCCESS Then  
    MsgBox "印刷命令ファイルを作成しました。", vbOKOnly, s_cstTITLE  
Else  
    If IRet = SFCSPRTFILE_ERR_NoMisFolder Then  
        MsgBox "印刷命令フォルダが存在しません", vbExclamation, s_cstTITLE  
    End If  
    If IRet = SFCSPRTFILE_ERR_GetMisFolder Then  
        MsgBox "印刷命令フォルダの取得に失敗しました", vbExclamation, s_cstTITLE  
    End If  
    If IRet = SFCSPRTFILE_ERR_GetTempFolder Then  
        MsgBox "一時フォルダの取得に失敗しました", vbExclamation, s_cstTITLE  
    End If  
    If IRet = SFCSPRTFILE_ERR_GetTempFile Then  
        MsgBox "一時ファイルの取得に失敗しました", vbExclamation, s_cstTITLE  
    End If  
    If IRet = SFCSPRTFILE_ERR_MakeMisName Then  
        MsgBox "印刷命令ファイル名の作成に失敗しました", vbExclamation, s_cstTITLE  
    End If  
    If IRet = SFCSPRTFILE_ERR_PARAM_INFO Then  
        MsgBox "パラメータエラー: 印刷命令ファイル作成情報が指定されていません。", vbExclamation, s_cstTITLE  
    End If  
    If IRet = SFCSPRTFILE_ERR_PARAM_DOC Then  
        MsgBox "パラメータエラー: 印刷原稿ファイルが指定されていません。", vbExclamation, s_cstTITLE  
    End If  
    If IRet = SFCSPRTFILE_ERR_PARAM_DOCNAME Then  
        MsgBox "パラメータエラー: 印刷原稿ファイル名が指定されていません。", vbExclamation, s_cstTITLE  
    End If  
End If  
  
End Sub
```

2.5 メール送信

メール送信プログラム【SendMail.exe】は、メールの送信を行うサンプルプログラムです。本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥SendMail.exe ... メール送信プログラム
¥サンプル¥VB6 SP6¥SendMail¥ ... メール送信プログラム 開発プロジェクト

主な仕様、および操作方法是以下の通りです。

- ① STARFAX Server SDK 環境設定プログラム-メールサーバー でメール送信で使用するメールサーバー環境を設定します。
- ② STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
- ③ メール送信プログラム【SendMail.exe】を起動します。
(起動時の作業(カレント)フォルダの指定は特にありません)
- ④ 宛先を指定します。
 - ・ 複数の宛先を指定する場合は、カンマ区切りで指定します。
 - ・ 最低限、1 件の宛先を指定する必要があります。
- ⑤ 必要であれば、CC を指定します。
 - ・ 複数の CC を指定する場合は、カンマ区切りで指定します。
- ⑥ 必要であれば、件名を指定します。
- ⑦ 必要であれば、本文を指定します。
- ⑧ 必要であれば、添付ファイルを指定します。
 - ・ 最大2つの添付ファイルを指定できます。
(これは、このサンプルプログラムの仕様です)
 - ・ 指定できるファイル形式に特に制限はありません。

⑨ 送信(P)ボタンを押して、メール送信を行います。

この後、メール送信が正常に動作していない場合は、STARFAX Server SDK ログ管理プログラム でイベントの内容を確認して下さい。

2.5 メール送信

2.5.1 メールを送信する

メールを送信するためのプログラミング例を、メール送信プログラム 開発プロジェクトのソースファイルを元にご説明します。

メール送信プログラム【SnedMail.exe】の仕様、および操作方法については、2.5 メール送信 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥SendMail¥ ... メール送信プログラム 開発プロジェクト

■ メール送信

STARFAX Server SDK へのメール送信命令は、メール送信命令フォルダにメール送信命令ファイルを置くことで行います。そして、メール送信命令ファイルの作成は、STARFAX Server SDK OCX のメール送信コントロール [SFCSSendMail] が行います。

作成したユーザープログラムで、メール送信が正常に動作していない場合は、STARFAX Server SDK ログ管理プログラム でイベントの内容を確認して下さい。そして、その内容を参考にしてプログラムを見直してみてください。

Form1.frm :

メール送信ボタン: クリック

Private Sub CommandOK_Click()

Dim strSubject As String

```
Dim strTheBody As String
```

Dim strAttachTEMP As String

Dim lRet As Long

， 件名

```
strSubject = TextSUBJECT.Text
```

， 本文

```
strTheBody = TextTHEBODY.Text
```

////////////////////////////////////

'// (1) 宛先設定

' //

' // SFCSSendMail コントロール 《 宛先リセット 》

```
'// Sub ResetToInfo()
```

' //

'// SFCSSendMail コントロール 《宛先リスト追加設定》

```
'// Function AddToInfo(ToAddress As String)as Long
```

'//

SFCSSendMail1.ResetToInfo

If TextT0.Text \diamond "" Then

```
lRet = SFCSSendMail1.AddToInfo(TextT0.Text)
```

```
If IRet <> SFCSSENDMAIL_SUCCESS Then
```

MsgBox "メモリエラーが発生しました", vbCritical, s_cstTITLE

Exit Sub

End If

End If

////////////////////////////////////

' // (2) CC 設定

'//

```
'// SFCSSendMail コントロール 《 CC リストリセット 》
'// Sub ResetCcInfo()
'//
'// SFCSSendMail コントロール 《 CC リスト追加設定 》
'// Function AddCcInfo(CcAddress As String) as Long
'//
```

SFCSSendMail1.ResetCcInfo

```
If TextCC.Text <> "" Then
```

```
    IRet = SFCSSendMail1.AddCcInfo(TextCC.Text)
    If IRet <> SFCSENDMAIL_SUCCESS Then
        MsgBox "メモリエラーが発生しました", vbCritical, s_cstTITLE
        Exit Sub
    End If
```

```
End If
```

```
'////////////////////
'// (3) 添付ファイル設定
'//
'// SFCSSendMail コントロール 《 添付ファイルリストリセット 》
'// Sub ResetAttachInfo()
'//
'// SFCSSendMail コントロール 《 添付ファイルリスト追加設定 》
'// Function AddAttachInfo(AttachName As String) as Long
'//
```

SFCSSendMail1.ResetAttachInfo

```
For i = 1 To 2
```

```
    Select Case i
        Case 1
            strAttachTEMP = TextATTACH1.Text
        Case 2
            strAttachTEMP = TextATTACH2.Text
    End Select
```

```
If strAttachTEMP <> "" Then
```

```
    IRet = SFCSSendMail1.AddAttachInfo(strAttachTEMP)
    If IRet <> SFCSENDMAIL_SUCCESS Then
        MsgBox "メモリエラーが発生しました", vbCritical, s_cstTITLE
        Exit Sub
    End If
```

```

End If

End If

Next i

' ///////////////////////////////////////////////////////////////////
' // (4) メール送信命令ファイル作成
' //
' // SFCSSendMail コントロール 《 メール送信命令ファイル作成 》
' // Function MakeEmailMission (Subject As String,
' //                               TheBody As String) as Long
' //

IRet = SFCSSendMail1.MakeEmailMission(strSubject, strTheBody)

If IRet = SFCSENDMAIL_SUCCESS Then
    MsgBox "メール送信命令ファイルを作成しました。", vbOKOnly, s_cstTITLE
Else
    If IRet = SFCSENDMAIL_ERR_NoMisFolder Then
        MsgBox "メール送信命令フォルダが存在しません", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDMAIL_ERR_GetMisFolder Then
        MsgBox "メール送信命令フォルダの取得に失敗しました", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDMAIL_ERR_GetTempFolder Then
        MsgBox "一時フォルダの取得に失敗しました", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDMAIL_ERR_GetTempFile Then
        MsgBox "一時ファイルの取得に失敗しました", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDMAIL_ERR_MakeMisName Then
        MsgBox "メール送信命令ファイル名の作成に失敗しました", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDMAIL_ERR_PARAM_INFO Then
        MsgBox "パラメータエラー: メール送信命令ファイル作成情報が指定されていません。", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDMAIL_ERR_PARAM_TO Then
        MsgBox "パラメータエラー: 宛先が指定されていません。", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDMAIL_ERR_PARAM_TOADR Then
        MsgBox "パラメータエラー: 宛先アドレスが指定されていません。", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDMAIL_ERR_PARAM_CCADR Then
        MsgBox "パラメータエラー: CC アドレスが指定されていません。", vbExclamation, s_cstTITLE
    End If

```



```
If IRet = SFCSENDMAIL_ERR_PARAM_ATTACHNAME Then
    MsgBox "パラメータエラー：添付ファイルパスが指定されていません。", vbExclamation, s_cstTITLE
End If
End If

End Sub
```

2.6 動作情報の参照

動作情報プログラム【MonEnv.exe】は、STARFAX Server SDK の起動状況、および動作状況の表示を行うサンプルプログラムです。本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonEnv.exe ... 動作情報プログラム

¥サンプル¥VB6 SP6¥MonEnv¥ ... 動作情報プログラム 開発プロジェクト

主な仕様、および操作方法是以下の通りです。

- ① STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
 - ② 動作情報プログラム【MonEnv.exe】を起動します。
(起動時の作業(カレント)フォルダの指定は特にありません)
 - ③ ダイアログに各種動作状況が表示されます。
 - ④ 更新(U) ボタンを押すと、表示内容が最新の状態に更新されます。
-

2.6 動作情報の参照

2.6.1 動作情報を参照する

STARFAX Server SDK の起動状況、および動作状況の参照する為のプログラミング例を、動作情報プログラム 開発プロジェクトのソースファイルを元にご説明します。

動作情報プログラム【MonEnv.exe】の仕様、および操作方法については、2.6 動作情報の参照 をご覧下さい。

開発プロジェクトは、本 CD-ROM の以下の位置に入っています。

¥サンプル¥VB6 SP6¥MonEnv¥ ... 動作情報プログラム 開発プロジェクト

■ STARFAX Server SDK の起動状況、および動作状況の参照

STARFAX Server SDK の起動状況、および動作状況の参照は、起動情報ファイルを参照することにより行います。そして、起動情報ファイルの参照は、STARFAX Server SDK OCX の動作情報の参照コントロール [SFCSEnv] が行います。

Form1.frm :

動作情報の表示

Private Sub UpdateConorols()

Dim strWork As String

Dim strWarning As String

Dim lRet As String

' //////////////////////////////////////

' // (1) STARFAX Engine サービス状態取得

' //

' // SFCSEnv コントロール 《 STARFAX Engine サービス状態取得 》

' // Function CheckServiceStatus() as Long

' //

If SFCSEnv1.CheckServiceStatus() <> SFCSENV_SERVICE_STOP Then

s_bRun = True

TextRUN.Text = "動作中"

Else

s_bRun = False

TextRUN.Text = "停止中"

Exit Sub

End If

' //////////////////////////////////////

' // (2) その他情報を取得

' //

' // SFCSEnv コントロール 《 起動情報文字列取得 》

' // Function GetRunInfoString(Section As String, Key As String,

' // Default As String, Get As String)as Long

' //

' // SFCSEnv コントロール 《 起動情報数値取得 》

' // Function GetRunInfoInt(Section As String, Key As String,

' // Default As Long)as Long

' //

' // ■ 起動情報ファイル

' // [制御関連インターフェイスフォルダ\SfCsRun.inf

' // ● セクション名: [Version] ... バージョン 情報

' // ・ Product ... 製品種類 ("SfCs" 固定)

```

'//      ▪ Major      ... メジャーバージョン
'//      ▪ Minor      ... マイナーバージョン
'//      ▪ Revision   ... バージョン毎修正回数
'//      ● セクション名: [Line0] ... 回線0情報
'//      ● セクション名: [Line1] ... 回線1情報
'//      ● セクション名: [Line2] ... 回線2情報
'//      ● セクション名: [Line3] ... 回線3情報
'//      ▪ Run        ... 回線起動状況 ("0":動作していない, "1":動作している)
'//      ▪ Send        ... 送信状況 ("0":送信不可, "1":送信可)
'//      ▪ Receive     ... 受信状況 ("0":受信不可, "1":受信可)
'//      ▪ Modem       ... モデム名
'//      ● セクション名: [AutoDel] ... 自動削除情報
'//      ▪ AutoTx       ... 送信情報自動削除 (1:ON, 0:OFF)
'//      ▪ AutoTxTime   ... 送信情報自動削除対象 経過日数 (日)
'//      ▪ AutoRx       ... 受信情報自動削除 (1:ON, 0:OFF)
'//      ▪ AutoRxTime   ... 受信情報自動削除対象 経過日数 (日)
'//      ● セクション名: [Warning] ... 警告
'//      ▪ DiskFreeInstall ... インストールフォルダのディスク空き容量
'//                                     (1:残り 150M 以下, 0:正常)
'//      ▪ DiskFreeCtrl  ... 制御関連インターフェイスフォルダのディスク空き容量
'//                                     (1:残り 150M 以下, 0:正常)
'//      ▪ DiskFreeData  ... データフォルダのディスク空き容量
'//                                     (1:残り 150M 以下, 0:正常)
'//      ▪ SendIndexNum  ... 送信情報インデックスファイル件数
'//                                     (1:8 万件以上, 0:正常)
'//      ▪ RecvIndexNum  ... 受信情報インデックスファイル件数
'//                                     (1:8 万件以上, 0:正常)
'//      ▪ QueIndexNum   ... 未送信情報インデックスファイル件数
'//                                     (1:8 万件以上, 0:正常)
'//      ▪ DustSendIndexNum ... ごみ箱 送信情報インデックスファイル件数
'//                                     (1:8 万件以上, 0:正常)
'//      ▪ DustRecvIndexNum ... ごみ箱 受信情報インデックスファイル件数
'//                                     (1:8 万件以上, 0:正常)
'//
'//

```

```

If SFCSEnv1.GetRunInfoString(SFCSENV_SEC_RUNVERSION, SFCSENV_KEY_RUNMAJOR, "", strWork) _
    = SFCSENV_SUCCESS Then
    TextMAJOR.Text = strWork
End If
If SFCSEnv1.GetRunInfoString(SFCSENV_SEC_RUNVERSION, SFCSENV_KEY_RUNMINOR, "", strWork) _
    = SFCSENV_SUCCESS Then
    TextMINOR.Text = strWork
End If
If SFCSEnv1.GetRunInfoString(SFCSENV_SEC_RUNVERSION, SFCSENV_KEY_RUNREVISION, "", strWork) _
    = SFCSENV_SUCCESS Then

```

```

TextREVISION.Text = strWork
End If

strWork = “しない”
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_AUTODEL, SFCSENV_KEY_AUTOTX, 0)
If IRet < 0 Then
    IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_AUTODEL, SFCSENV_KEY_AUTOTXTIME, 0)
    strWork = Str(IRet) & “ 日”
End If
TextAUTOTX.Text = strWork
strWork = “しない”
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_AUTODEL, SFCSENV_KEY_AUTORX, 0)
If IRet < 0 Then
    IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_AUTODEL, SFCSENV_KEY_AUTORXTIME, 0)
    strWork = Str(IRet) & “ 日”
End If
TextAUTORX.Text = strWork

IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE0, SFCSENV_KEY_LINERUN, 0)
If IRet < 0 Then
    TextLINE0_RUN.Text = “動作中”
    IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE0, SFCSENV_KEY_LINESEND, 0)
    If IRet < 0 Then
        TextLINE0_SEND.Text = “送信可”
    Else
        TextLINE0_SEND.Text = “送信不可”
    End If
    IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE0, SFCSENV_KEY_LINERECV, 0)
    If IRet < 0 Then
        TextLINE0_RECEIVE.Text = “受信可”
    Else
        TextLINE0_RECEIVE.Text = “受信不可”
    End If
    If SFCSEnv1.GetRunInfoString(SFCSENV_SEC_LINE0, SFCSENV_KEY_LINEMDM, 0, strWork) = _
        SFCSENV_SUCCESS Then
        TextLINE0_MODEM.Text = strWork
    End If
End If

IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE1, SFCSENV_KEY_LINERUN, 0)
If IRet < 0 Then
    TextLINE1_RUN.Text = “動作中”
    IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE1, SFCSENV_KEY_LINESEND, 0)
    If IRet < 0 Then
        TextLINE1_SEND.Text = “送信可”

```

```

Else
    TextLINE1_SEND.Text = “送信不可”
End If
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE1, SFCSENV_KEY_LINERECV, 0)
If IRet < 0 Then
    TextLINE1_RECEIVE.Text = “受信可”
Else
    TextLINE1_RECEIVE.Text = “受信不可”
End If
If SFCSEnv1.GetRunInfoString(SFCSENV_SEC_LINE1, SFCSENV_KEY_LINEMDM, 0, strWork) = _
    SFCSENV_SUCCESS Then
    TextLINE1_MODEM.Text = strWork
End If
End If

IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE2, SFCSENV_KEY_LINERUN, 0)
If IRet < 0 Then
    TextLINE2_RUN.Text = “動作中”
    IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE2, SFCSENV_KEY_LINESEND, 0)
    If IRet < 0 Then
        TextLINE2_SEND.Text = “送信可”
    Else
        TextLINE2_SEND.Text = “送信不可”
    End If
    IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE2, SFCSENV_KEY_LINERECV, 0)
    If IRet < 0 Then
        TextLINE2_RECEIVE.Text = “受信可”
    Else
        TextLINE2_RECEIVE.Text = “受信不可”
    End If
    If SFCSEnv1.GetRunInfoString(SFCSENV_SEC_LINE2, SFCSENV_KEY_LINEMDM, 0, strWork) = _
        SFCSENV_SUCCESS Then
        TextLINE2_MODEM.Text = strWork
    End If
End If
End If

IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE3, SFCSENV_KEY_LINERUN, 0)
If IRet < 0 Then
    TextLINE3_RUN.Text = “動作中”
    IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE3, SFCSENV_KEY_LINESEND, 0)
    If IRet < 0 Then
        TextLINE3_SEND.Text = “送信可”
    Else
        TextLINE3_SEND.Text = “送信不可”
    End If
End If

```

```

IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_LINE3, SFCSENV_KEY_LINERECV, 0)
If IRet < 0 Then
    TextLINE3_RECEIVE.Text = “受信可”
Else
    TextLINE3_RECEIVE.Text = “受信不可”
End If
If SFCSEnv1.GetRunInfoString(SFCSENV_SEC_LINE3, SFCSENV_KEY_LINEMDM, 0, strWork) = _
    SFCSENV_SUCCESS Then
    TextLINE3_MODEM.Text = strWork
End If
End If

strWarning = ""
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_WARNING, SFCSENV_KEY_DISKFREEINST, 0)
If IRet < 0 Then
    strWarning = strWarning & _
        “STARFAX Engine がインストールされているフォルダのディスク空き容量が 150 メガバイト以下 になりました” & vbCrLf
End If
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_WARNING, SFCSENV_KEY_DISKFREECTRL, 0)
If IRet < 0 Then
    strWarning = strWarning & _
        “制御関連情報のインターフェイスフォルダのディスク空き容量が 150 メガバイト以下 になりました” & vbCrLf
End If
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_WARNING, SFCSENV_KEY_DISKFREEDATA, 0)
If IRet < 0 Then
    strWarning = strWarning & _
        “データ情報が設定されるフォルダのディスク空き容量が 150 メガバイト以下 になりました” & vbCrLf
End If
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_WARNING, SFCSENV_KEY_SENDIDXNUM, 0)
If IRet < 0 Then
    strWarning = strWarning & “送信情報インデックスファイルの件数が 8 万件以上 になりました” & vbCrLf
End If
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_WARNING, SFCSENV_KEY_RECVIDXNUM, 0)
If IRet < 0 Then
    strWarning = strWarning & “受信情報インデックスファイルの件数が 8 万件以上 になりました” & vbCrLf
End If
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_WARNING, SFCSENV_KEY_QUEIDXNUM, 0)
If IRet < 0 Then
    strWarning = strWarning & “未送信情報インデックスファイルの件数が 8 万件以上 になりました” & vbCrLf
End If
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_WARNING, SFCSENV_KEY_DUSENDIDXNUM, 0)
If IRet < 0 Then
    strWarning = strWarning & “ごみ箱 送信情報インデックスファイルの件数が 8 万件以上 になりました” & vbCrLf
End If
IRet = SFCSEnv1.GetRunInfoInt(SFCSENV_SEC_WARNING, SFCSENV_KEY_DURECVIDXNUM, 0)

```



```
If IRet <> 0 Then
    strWarning = strWarning & "ごみ箱 受信情報インデックスファイルの件数が 8 万件以上 になりました" & vbCrLf
End If
TextWARNING.Text = strWarning

End Sub
```

第Ⅲ章

クライアントプログラムの開発について

ユーザープログラムの開発の手順についてご説明しています。

3.1 開発の手順

3.1 開発の手順

STARFAX Server SDK をセットアップしていない別のコンピュータから FAX 送信を行う際には、1 台のコンピュータに STARFAX Server SDK 本体のセットアップを行い、もう 1 台のコンピュータにクライアント環境のセットアップを行う必要があります。そして、ユーザープログラム開発を行う前に、STARFAX Server SDK 本体の操作を簡単に理解しておく必要があります。

それらを考慮して、以下の手順でユーザープログラム開発を行うことをお奨めします。

① 「STARFAX Server SDK セットアップマニュアル」をお読みください。

- [STARFAX Server SDK のセットアップ(本体)] を行ってください。
 - STARFAX Server SDK の基本的な操作を理解してください。
 - もう 1 台のコンピュータに
 - [クライアント送信のサンプルと O C X のセットアップ]
 - [プリンタドライバ のセットアップ] (※)
 - [ビューア のセットアップ] (※)を行ってください。
- (※ サンプルプログラムによって必要となります)

② [STARFAX Server SDK 本体をセットアップしているコンピュータ] と [クライアント送信を行うコンピュータ] の連絡を、共有フォルダを利用して行う場合は、

第IV章 FAX 送信命令フォルダを共有して FAX 送信

を、ご覧ください。

メール送信を利用して連絡を行う場合は、

第V章 [メール de FAX]で FAX 送信

を、ご覧ください。

そして、クライアントコンピュータから FAX 送信する仕組み、及び、各種サンプルプログラムを学習してください。

③ ユーザープログラム を作成してください。

- ①～② を踏まえて、ユーザープログラムの作成・テストを行って下さい。



第Ⅳ章

FAX 送信命令フォルダを共有して FAX 送信

FAX 送信命令フォルダを共有して、STARFAX Server SDK 本体をセットアップしていないコンピュータから FAX 送信する方法についてご説明します。

- 2.1 FAX 送信命令フォルダを共有して FAX 送信
- 2.2 サンプルプログラム

4.1 FAX 送信命令フォルダを共有して FAX 送信

FAX 送信命令フォルダを共有して、STARFAX Server SDK 本体をセットアップしていないコンピュータから FAX 送信する方法は、以下の通りです。

以降、STARFAX Server SDK 本体をセットアップしているコンピュータを [サーバー側]、STARFAX Server SDK 本体をセットアップしていないコンピュータを [クライアント側] とします。

- ① [サーバー側] で STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「STARFAX Server SDK セットアップマニュアル」をお読みください)
- ② [サーバー側] の 送信命令フォルダ[制御関連インターフェイスフォルダ\SENDMIS] を共有設定して、他のコンピュータからアクセス(読み書き)できるようにします。
(※フォルダの共有設定方法は、OSにより異なります)
- ③ [クライアント側] から STARFAX Server SDK OCX のメソッドを使用して、共有している送信命令フォルダに、送信命令ファイルを作成します。

(FAX 送信コントロール [SFCSSendFax] - MakeSendMissionToSharedFolder 《 共有フォルダ形式の送信命令ファイル作成 》 FAX 送信の詳細は、「STARFAX Server SDK VB 開発向け OCX リファレンスマニュアル」の 2.1 FAX 送信コントロール [SFCSSendFax] をご覧ください)

4.2 サンプルプログラム

サンプルプログラムは、3種類のプログラムを用意しています。それぞれ、元となるサンプルプログラムがあり、FAX 送信に関する部分を共有フォルダに出力する仕組みに変更して、クライアント動作するようにしています。

共有フォルダ以外の処理については、元の各種マニュアルをご覧ください。

- FAX 送信する (SendFAX.exe)

元のプログラムは、「STARFAX Server SDK プログラミング (OCX 利用版)」の
2.1 FAX 送信 をご覧ください。

- 印刷結果の FAX 送信 (PrtCli.exe)

元のプログラムは、「STARFAX Server SDK VB 開発向けプリンタドライバとビューア」の
2.1 印刷結果の FAX 送信 をご覧ください。

- TIFF ファイルの作成と FAX 送信 (PrtCli2.exe)

元のプログラムは、「STARFAX Server SDK VB 開発向けプリンタドライバとビューア」の
2.2 TIFF ファイルの作成と FAX 送信 をご覧ください。

4.2.1 FAX 送信

FAX 送信プログラム【SendFax.exe】は、STARFAX Server SDK 本体をセットアップしていないコンピュータから FAX の送信を行うサンプルプログラムです。本 CD-ROM の以下の位置に入っています。

¥サンプル¥応用サンプル¥クライアント¥共有フォルダ¥VB6¥SendFax.exe ... FAX 送信プログラム
¥サンプル¥応用サンプル¥クライアント¥共有フォルダ¥VB6¥SendFax¥
... FAX 送信プログラム 開発プロジェクト

なお、このサンプルプログラムには、元となるサンプルプログラムがあり、FAX 送信に関する部分を共有フォルダに出力する仕組みに変更して、クライアント動作するようにしています。

共有フォルダ以外の処理については、「STARFAX Server SDK プログラミング(OCX 利用版)」の 2.1 FAX 送信をご覧ください。

主な仕様、および操作方法是以下の通りです。

(以降、STARFAX Server SDK 本体をセットアップしているコンピュータを [サーバー側]、STARFAX Server SDK 本体をセットアップしていないコンピュータを [クライアント側] とします)

- ① [サーバー側] で STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
- ② [サーバー側] の 送信命令フォルダ[制御関連インターフェイスフォルダ¥SENDMIS] を共有設定して、他のコンピュータからアクセス(読み書き)できるようにします。
(※フォルダの共有設定方法は、OSにより異なります)
- ③ [クライアント側] で FAX 送信プログラム【SendFax.exe】を起動します。
(起動時の作業(カレント)フォルダの指定は特にありません)
- ④ [共有設定した FAX 送信フォルダ(X)] を指定します。
- ⑤ 各種送信内容を指定します。
 - [相手先(S)] ボタンを押して、相手先を指定します。
 - [原稿(D)] ボタンを押して、原稿を指定します。
 - 必要であれば、[送付状(C)] ボタンを押して、送付状を指定します。
 - 必要であれば、[発信元(U)] ボタンを押して、発信元情報を指定します。

⑥ [送信 (G)] ボタンを押して、FAX 送信を行います。

この後、FAX 送信が正常に動作していない場合は、[サーバー側] の STARFAX Server SDK ログ管理プログラム で [イベント] の内容を確認して下さい。

Form1.frm :

```
'*****  
' 送信ボタン: クリック  
'*****
```

```
Private Sub CommandOK_Click()
```

～

```
'/////////////////////////////////////////  
'// (3) 共有フォルダ形式の送信命令ファイル作成  
'//  
'// SFCSSendFax コントロール 《 共有フォルダ形式の送信命令ファイル作成 》  
'// Function MakeSendMissionToSharedFolder(SharedFolder As String,  
'// CoverName As String,  
'// FontName As String,  
'// FontSize As Integer  
'// UserInfo As String,  
'// UserID As String,  
'// Account As String)as Long  
'//  
  
IRet = SFCSSendFax1.MakeSendMissionToSharedFolder(strShrFolder, strCover, _  
strCoverFontName, sCoverFontSize, strUserInfo, strUserID, "")  
  
If IRet = SFCSENDFAX_SUCCESS Then  
MsgBox "送信命令ファイルを作成しました。", vbOKOnly, s_cstTITLE  
Else  
If IRet = SFCSENDFAX_ERR_GetTempFolder Then  
MsgBox "一時フォルダの取得に失敗しました", vbExclamation, s_cstTITLE  
End If  
If IRet = SFCSENDFAX_ERR_GetTempFile Then  
MsgBox "一時ファイルの取得に失敗しました", vbExclamation, s_cstTITLE  
End If  
If IRet = SFCSENDFAX_ERR_MakeMisName Then  
MsgBox "送信命令ファイル名の作成に失敗しました", vbExclamation, s_cstTITLE  
End If  
If IRet = SFCSENDFAX_ERR_NoShareFolder Then  
MsgBox "共有設定した F A X 送信命令フォルダが存在しません", _  
vbExclamation, s_cstTITLE  
End If  
If IRet = SFCSENDFAX_ERR_MakeShareDocFolder Then
```

```

        MsgBox "共有フォルダに送信原稿フォルダを作成できませんでした", _
            vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDFAX_ERR_MakeShareDocFile Then
        MsgBox "共有フォルダに送信原稿をコピーできませんでした", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDFAX_ERR_MakeShareCvrFile Then
        MsgBox "共有フォルダに送付状をコピーできませんでした", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDFAX_ERR_PARAM_INFO Then
        MsgBox "パラメータエラー: 送信命令ファイル作成情報が指定されていません。", _
            vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDFAX_ERR_PARAM_SENDDNUM Then
        MsgBox "パラメータエラー: 相手先数の指定が 0 です。", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDFAX_ERR_PARAM_SENDINFO Then
        MsgBox "パラメータエラー: 相手先情報が指定されていません。", _
            vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDFAX_ERR_PARAM_FAX Then
        MsgBox "パラメータエラー: FAX 番号が指定されていません。", vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDFAX_ERR_PARAM_DOCorCOVER Then
        MsgBox "パラメータエラー: 原稿ファイル、送付状ファイルが、指定されていません。", _
            vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDFAX_ERR_PARAM_DOCNAME Then
        MsgBox "パラメータエラー: 送信原稿ファイル名が指定されていません。", _
            vbExclamation, s_cstTITLE
    End If
    If IRet = SFCSENDFAX_ERR_PARAM_SHAREFOLDER Then
        MsgBox "パラメータエラー: 共有設定 FAX 送信命令フォルダが指定されていません。", _
            vbExclamation, s_cstTITLE
    End If
End If
End Sub

```

4.2.2 印刷結果のFAX送信

印刷結果のFAX送信プログラム【PrtCli.exe】は、STARFAX Server SDK 本体をセットアップしていないコンピュータで印刷結果の表示とFAX送信を行うサンプルプログラムです。本CD-ROMの以下の位置に入っています。ワード・エクセル等のアプリケーションから手動で印刷後、プリンタドライバからユーザープログラムが起動されます。

¥サンプル¥応用¥サンプル¥クライアント¥共有フォルダ¥VB6¥PrtCli.exe
... 印刷結果のFAX送信プログラム
¥サンプル¥応用¥サンプル¥クライアント¥共有フォルダ¥VB6¥PrtCli¥
... 印刷結果のFAXプログラム 開発プロジェクト

なお、このサンプルプログラムには、元となるサンプルプログラムがあり、FAX送信に関する部分を共有フォルダに出力する仕組みに変更して、クライアント動作するようにしています。共有フォルダ以外の処理については、「STARFAX Server SDK VB 開発向けプリンタドライバとビューア」の2.1 印刷結果のFAX送信をご覧ください。

主な仕様、および操作方法是以下の通りです。

(以降、STARFAX Server SDK 本体をセットアップしているコンピュータを [サーバー側]、STARFAX Server SDK 本体をセットアップしていないコンピュータを [クライアント側] とします)

- ① [サーバー側] で STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
- ② [サーバー側] の 送信命令フォルダ [制御関連インターフェイスフォルダ¥SENDMIS] を共有設定して、他のコンピュータからアクセス(読み書き)できるようにします。
(※フォルダの共有設定方法は、OSにより異なります)
- ③ [クライアント側] で、以下の STARFAX Server SDK プリンタドライバの動作に関するレジストリを指定します。
 - HKEY_LOCAL_MACHINE¥Software¥MEGASOFT¥SfCs¥OutFolder ... ファイル出力フォルダ
文字列項目で、任意の作業フォルダを作成して指定します。
(例: "C:¥Program Files¥SfCs¥Temp")
 - HKEY_LOCAL_MACHINE¥Software¥MEGASOFT¥SfCs¥DocName ... ドキュメント名
文字列項目で、このサンプルプログラムの場合、任意の文字列を指定します。
(例: "SFCSPRN")

- HKEY_LOCAL_MACHINE\Software\MEGASOFT\SfCs\ExecFlag ...プログラム実行フラグ
DWORD 項目で、1 を指定します。
 - HKEY_LOCAL_MACHINE\Software\MEGASOFT\SfCs\ExecPath ...プログラムパス
文字列項目で、PrtCli.exe をフルパスで指定します。
(例: "C:\Program Files\SfCs\PrtCli.exe")
 - HKEY_LOCAL_MACHINE\Software\MEGASOFT\SfCs\ExecParam ...追加パラメータ
文字列項目で、何も指定していない状態("")を設定します。
- ④ 印刷可能な適当なアプリケーション(ワード等)から プリンタ名 "MEGASOFT STARFAX Server SDK"
に対して印刷を行うと 印刷結果のFAX送信プログラム【PrtCli.exe】 が起動して、印刷結果リ
スト に 印刷結果が 登録された状態になります。
- ⑤ [共有設定したFAX 送信フォルダ(X)]を指定します。
- ⑥ [表示(V)]ボタンを押して、印刷結果の内容を確認します。
- ⑦ [FAX送信(S)]ボタンを押して、FAX 送信を行います。
この後、FAX 送信が正常に動作していない場合は、[サーバー側] の STARFAX Server SDK ログ
管理プログラム で [イベント] の内容を確認して下さい。
-

Form1.frm :

```
'*****  
' F A X送信ボタン: クリック  
'*****
```

```
Private Sub CommandOK_Click()
```

```
    ~
```

```
    ' _____  
    ' 共有フォルダ形式の送信命令ファイル作成  
    ' _____
```

```
    IRet = SFCSSendFax1.MakeSendMissionToSharedFolder(strShrFolder, "", "", 0, "", "", "")
```

```
    If IRet = SFCSENDFAX_SUCCESS Then
```

```
        MsgBox "送信命令ファイルを作成しました。", vbOKOnly, s_cstTITLE
```

```
    Else
```

```
        If IRet = SFCSENDFAX_ERR_GetTempFolder Then
```

```
            MsgBox "一時フォルダの取得に失敗しました", vbExclamation, s_cstTITLE
```

```
        End If
```

```
        If IRet = SFCSENDFAX_ERR_GetTempFile Then
```

```
            MsgBox "一時ファイルの取得に失敗しました", vbExclamation, s_cstTITLE
```

```
        End If
```

```
        If IRet = SFCSENDFAX_ERR_MakeMisName Then
```

```
            MsgBox "送信命令ファイル名の作成に失敗しました", vbExclamation, s_cstTITLE
```

```
        End If
```

```
        If IRet = SFCSENDFAX_ERR_NoShareFolder Then
```

```
            MsgBox "共有設定した F A X送信命令フォルダが存在しません", _  
                vbExclamation, s_cstTITLE
```

```
        End If
```

```
        If IRet = SFCSENDFAX_ERR_MakeShareDocFolder Then
```

```
            MsgBox "共有フォルダに送信原稿フォルダを作成できませんでした", _  
                vbExclamation, s_cstTITLE
```

```
        End If
```

```
        If IRet = SFCSENDFAX_ERR_MakeShareDocFile Then
```

```
            MsgBox "共有フォルダに送信原稿をコピーできませんでした", _  
                vbExclamation, s_cstTITLE
```

```
        End If
```

```
        If IRet = SFCSENDFAX_ERR_MakeShareCvrFile Then
```

```
            MsgBox "共有フォルダに送付状をコピーできませんでした", vbExclamation, s_cstTITLE
```

```
        End If
```

```

If IRet = SFCSENDFAX_ERR_PARAM_INFO Then
    MsgBox "パラメータエラー: 送信命令ファイル作成情報が指定されていません。", _
        vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_SENDFAX Then
    MsgBox "パラメータエラー: 相手先数の指定が 0 です。", vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_SENDFAXINFO Then
    MsgBox "パラメータエラー: 相手先情報が指定されていません。", _
        vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_FAX Then
    MsgBox "パラメータエラー: FAX 番号が指定されていません。", vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_DOCorCOVER Then
    MsgBox "パラメータエラー: 原稿ファイル、送付状ファイルが、指定されていません。", _
        vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_DOCNAME Then
    MsgBox "パラメータエラー: 送信原稿ファイル名が指定されていません。", _
        vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_SHAREFOLDER Then
    MsgBox "パラメータエラー: 共有設定 FAX 送信命令フォルダが指定されていません。", _
        vbExclamation, s_cstTITLE
End If
End If
End If

~

End

End Sub

```

4.2.3 TIFF ファイルの作成と F A X 送信

TIFF ファイルの F A X 送信プログラム【PrtCli2.exe】は、STARFAX Server SDK 本体をセットアップしていないコンピュータで TIFF ファイルの作成 と FAX 送信を行うサンプルプログラムです。本 CD-ROM の以下の位置に入っています。ユーザープログラムがプリンタドライバを制御して印刷結果 (TIFF ファイル) を取得します。

¥サンプル¥応用サンプル¥クライアント¥共有フォルダ¥VB6¥PrtCli2.exe

... TIFF ファイルの F A X 送信プログラム

¥サンプル¥応用サンプル¥クライアント¥共有フォルダ¥VB6¥PrtCli2¥

... TIFF ファイルの F A X 送信プログラム 開発プロジェクト

なお、このサンプルプログラムには、元となるサンプルプログラムがあり、FAX 送信に関する部分を共有フォルダに出力する仕組みに変更して、クライアント動作するようにしています。共有フォルダ以外の処理については、「STARFAX Server SDK VB 開発向けプリンタドライバとビューア」の 2.2 *TIFF ファイルの作成と FAX 送信* をご覧ください。

主な仕様、および操作方法は以下の通りです。

(以降、STARFAX Server SDK 本体をセットアップしているコンピュータを [サーバー側]、STARFAX Server SDK 本体をセットアップしていないコンピュータを [クライアント側] とします)

- ① [サーバー側] で STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
- ② [サーバー側] の 送信命令フォルダ [制御関連インターフェイスフォルダ¥SENDMIS] を共有設定して、他のコンピュータからアクセス (読み書き) できるようにします。
(※フォルダの共有設定方法は、OS により異なります)
- ③ [クライアント側] で PrtCli2.exe を起動します。
- ④ [共有設定した FAX 送信フォルダ (X)] を指定します。
- ⑤ 「操作 1」の下欄に FAX 原稿に表示させる文字を入力します。
- ⑥ 「①FAX 原稿の作成」ボタンをクリックします。
(作成するファイル名を任意指定したい場合は、[指定] ラジオボタンをクリックして、ファイル名を入力してください)

- ⑦ 「②FAX 原稿の表示」ボタンをクリックすると、作成されたFAX 原稿が STARFAX Server SDK ビューアで表示されます。
 - ⑧ 「操作3」の下欄に送信先のFAX 番号を入力します。
 - ⑨ 「FAX 送信」ボタンをクリックします。
この後、FAX 送信が正常に動作していない場合は、[サーバー側] の STARFAX Server SDK ログ管理プログラム で [イベント] の内容を確認して下さい。
-

Form1.frm :

```
'*****  
' [FAX送信スタート]ボタン: クリック  
'*****
```

```
Private Sub CommandOK_Click()
```

～

```
' _____  
' 共有フォルダ形式の送信命令ファイル作成  
' _____
```

```
IRet = SFCSSendFax1.MakeSendMissionToSharedFolder(strShrFolder, "", "", 0, "", "", "")
```

```
If IRet = SFCSENDFAX_SUCCESS Then
```

```
    MsgBox "送信命令ファイルを作成しました。", vbOKOnly, s_cstTITLE
```

～

```
Else
```

```
    If IRet = SFCSENDFAX_ERR_GetTempFolder Then
```

```
        MsgBox "一時フォルダの取得に失敗しました", vbExclamation, s_cstTITLE
```

```
    End If
```

```
    If IRet = SFCSENDFAX_ERR_GetTempFile Then
```

```
        MsgBox "一時ファイルの取得に失敗しました", vbExclamation, s_cstTITLE
```

```
    End If
```

```
    If IRet = SFCSENDFAX_ERR_MakeMisName Then
```

```
        MsgBox "送信命令ファイル名の作成に失敗しました", vbExclamation, s_cstTITLE
```

```
    End If
```

```
    If IRet = SFCSENDFAX_ERR_NoShareFolder Then
```

```
        MsgBox "共有設定したFAX送信命令フォルダが存在しません", _  
            vbExclamation, s_cstTITLE
```

```
    End If
```

```
    If IRet = SFCSENDFAX_ERR_MakeShareDocFolder Then
```

```
        MsgBox "共有フォルダに送信原稿フォルダを作成できませんでした", _  
            vbExclamation, s_cstTITLE
```

```
    End If
```

```
    If IRet = SFCSENDFAX_ERR_MakeShareDocFile Then
```

```
        MsgBox "共有フォルダに送信原稿をコピーできませんでした", vbExclamation, s_cstTITLE
```

```

End If
If IRet = SFCSENDFAX_ERR_MakeShareCvrFile Then
    MsgBox "共有フォルダに送付状をコピーできませんでした", vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_INFO Then
    MsgBox "パラメータエラー: 送信命令ファイル作成情報が指定されていません。", _
        vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_SENDDNUM Then
    MsgBox "パラメータエラー: 相手先数の指定が 0 です。", vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_SENDINFO Then
    MsgBox "パラメータエラー: 相手先情報が指定されていません。", _
        vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_FAX Then
    MsgBox "パラメータエラー: FAX 番号が指定されていません。", vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_DOCorCOVER Then
    MsgBox "パラメータエラー: 送信ファイル、送付状ファイルが、指定されていません。", _
        vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_DOCNAME Then
    MsgBox "パラメータエラー: 送信原稿ファイル名が指定されていません。", _
        vbExclamation, s_cstTITLE
End If
If IRet = SFCSENDFAX_ERR_PARAM_SHAREFOLDER Then
    MsgBox "パラメータエラー: 共有設定 FAX 送信命令フォルダが指定されていません。", _
        vbExclamation, s_cstTITLE
End If
End If
End Sub

```

第 V 章

[メール de FAX] で FAX 送信

[メール de FAX] 機能で、STARFAX Server SDK 本体をセットアップしていないコンピュータから FAX 送信する方法についてご説明します。

- 5.1 [メール de FAX] で FAX 送信
- 5.2 [メール de FAX] の依頼メールの仕様
- 5.3 サンプルプログラム

5.1 [メール de FAX]でFAX 送信

[メール de FAX]機能で、STARFAX Server SDK 本体をセットアップしていないコンピュータから FAX 送信する方法は、以下の通りです。

以降、STARFAX Server SDK 本体をセットアップしているコンピュータを [サーバー側]、STARFAX Server SDK 本体をセットアップしていないコンピュータを [クライアント側] とします。

① [サーバー側]で [メール de FAX] の設定を有効にします。

(1) タスクトレイの STARFAX Server SDK のアイコンを右クリックして表示されるメニューから [メール de FAX] を実行します。

(2) 最低限、以下の項目を設定します。

- [メール de FAX を有効にする(Y)] をチェックします。
- [メールサーバーの設定(S)] を行います。

(3) 必要に応じて、以下の項目を設定します。

- [件名に含まれる文字で識別(I)]
- [メール受信間隔(N)]
- [対象外のメールを EML ファイルに保存(P)]、及び、関連項目
- [FAX 送信結果をメール通知する(N)]、及び、関連項目

② [サーバー側]で STARFAX Server SDK を起動します。

(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)

③ [クライアント側]から[メール de FAX]の依頼メールを[サーバー側]で設定した [POP メールアドレス] に送信します。

([メール de FAX]の依頼メールの詳細は 3.2 [メール de FAX]の依頼メールの仕様をご覧ください)

5.2 [メール de FAX]の依頼メールの仕様

STARFAX Server SDK の [メール de FAX]機能が動作している場合、設定された [POP メールアドレス]に対しての依頼メールを送信すると、FAX 送信を行います。

この仕組みには、以下の特徴があります。

- 相手先を複数指定することができます。(同報送信)
- 送信原稿ファイルを複数指定することができます。
- 送付状を指定することができます。
(送付状ファイルは、テキストファイルで、差込内容を %? で指定できます)
- 発信元情報を指定することができます。

依頼メールの仕様についてご説明します。

■ 依頼メールの構成

● 宛先

[メール de FAX 設定] の [メールサーバー設定(S)]で設定されている [POP メールアドレス] を指定します。

● 件名

件名には、[メール de FAX 設定] の [件名に含まれる文字で識別(I)] で設定されている文字列を含んでいる必要があります。含んでいないと依頼メールとして扱われません。

例: “ Mail to FAX”

● 本文

[メール de FAX] では扱われません。
(設定されていても FAX 送信されることはありません)

● 添付ファイル

添付ファイルは以下のファイルを指定します。

- ・ 送信命令ファイル
- ・ 送信原稿ファイル
- ・ 送付状ファイル

詳細については後述をご覧ください。

■ 送信命令ファイル

送信命令ファイルは、INI ファイル形式のファイルです。

ファイル名は、必ず、“Trans. txt” である必要があります。

全ての項目を指定する必要はありません。

最低限、相手先の FAX 番号を指定すれば FAX 送信できます。(※)

(※: 添付ファイルで送信原稿ファイルが設定されている必要があります)

なお、STARFAX OCX に 送信命令ファイル作成メソッド をご用意しています。

(FAX 送信コントロール [SFCSSendFax] -

MakeTransFileForMailToFax 《 [メール de FAX] の送信命令ファイル作成 》)

```
[SendInfo]
Num=1
[Send1]
Fax=0663868894
```

その他の情報は、送付状への差込、発信元情報への差込、ログに情報を残したい、等の必要に応じて設定します。

● セクション名: [SendInfo] ... 送信のための相手先情報

- ・ Num ... 送信相手先数

● セクション名: [Send%id] ... 送信のための相手先内容 (1~)

- Fax ... FAX 番号 (最大 128 バイト)
- Company ... 会社名 (最大 128 バイト)
- Division ... 所属名 (最大 128 バイト)
- Position ... 役職名 (最大 128 バイト)
- Name ... 氏名 (最大 128 バイト)
- Title ... 敬称 (最大 128 バイト)
- Telephone ... 電話番号 (最大 128 バイト)
- ZipCode ... 郵便番号 (最大 128 バイト)
- Address1 ... 住所 1 (最大 128 バイト)
- Address2 ... 住所 2 (最大 128 バイト)
- Speed ... 通信速度 ("0": 自動, "1": 高速, "2": 中速, "3": 低速)
- Comp ... 圧縮方式 ("0": 自動, "1": MH, "2": MR, "3": MMR)
- Ecm ... エラー訂正 ("0": 自動, "1": 利用する, "2": 利用しない)
- FCode ... Fコード番号 (最大 20 バイト)
- FreeArea ... ユーザが自由に利用できるエリア (最大 128 バイト)
- Line ... 送信回線指定 (0~)
(指定がない場合は、空いている回線から送信されます)
- Priority ... 優先順位 (0~15)
(指定がない場合の優先順位は 8 です)
- Time ... 送信時刻 ("YYYYMMDDHHMMSS")

● セクション名: [Cover] ... 送付状

- Name ... 送付状ファイルパス (.txt)
(さらに、添付ファイルで設定する必要があります)
- FontName ... 送付状フォント名
(指定がない場合は "MS ゴシック" です)
- FontSize ... 送付状フォントサイズ (8 ~ 72 (ポイント))
(指定がない場合は 10 ポイント です)

● セクション名: [User] ... 発信元情報

- UserInfo ... 発信元情報記録 ("記録位置, 記録情報")
("0": 記録しない, "1": 原稿の内側に記録, "2": 原稿の外側に記録)
- UserID ... 自局電話番号 (FAXID として、相手ファクシミリに通知されます)
(最大 20 バイトで、半角数字、スペース、+を設定してください)

【 発信元情報の仕様 】

発信元情報とは、FAX 送信時に、各ページ毎、送信原稿の先頭に付加する情報です。

発信元情報は、[User]-UserInfo に、“記録位置, 記録情報” の形式で指定します。

● 記録位置

- 0: 記録しません。
- 1: 原稿の内側に記録します。原稿の内容によっては、原稿の先頭が少し消えてしまう可能性があります。
- 2: 原稿の外側に記録します。

「1」または「2」を指定すると、記録情報を指定していなくも日付とページ数は記録されます。

[User]

User Info=1,

2001 12/ 1 10:12 Page 01

● 記録情報

自由に文字を指定できます。

以下の差込も使用できます。

- ・ %S ... 会社名 ([Send%d] セクション-Company が差し込まれます)
- ・ %N ... 氏名 ([Send%d] セクション-Name と
[Send%d] セクション-Title が差し込まれます)
- ・ %T ... FAX 番号 ([Send%d] セクション-Fax が差し込まれます)

[Send1]

Company=山田サークル

Name=山田

[User]

User Info=1, メガ太郎 → %S %N

メガ太郎 → 山田サークル 山田様 2001 12/ 1 10:12 Page 01

■ 送信原稿ファイル

送信原稿ファイルは、以下の形式のファイルを複数指定可能です。

- ファイル形式：
- ・ TIFF 形式
 - 圧縮なし
 - 修正 CCITT MH 圧縮
 - CCITT G3 MH 圧縮
 - CCITT G3 MR 圧縮
 - PackBits 圧縮
 - Class F 圧縮
 - G4 圧縮
 - JPEG 圧縮
 - ・ BMP ファイル
 - ・ PCX ファイル
 - ・ DCX ファイル
 - ・ JPEG ファイル
 - ・ テキストファイル
 - ・ FAX ファイル
 - ・ LNK ファイル

■ 送付状ファイルの仕様

テキストファイル(*.txt)で、自由に作成したファイルを指定できます。
文字フォントは、“MS ゴシック”、大きさは10ポイントとして扱われ、FAX 送信されます。

以下の差込が有効です。

- %S ... 会社名 ([Send%d] セクション-Company が差し込まれます)
- %D ... 所属名 ([Send%d] セクション-Division が差し込まれます)
- %Y ... 役職名 ([Send%d] セクション-Position が差し込まれます)
- %N ... 氏名 ([Send%d] セクション-Name が差し込まれます)
- %Z ... 郵便番号 ([Send%d] セクション-ZipCode が差し込まれます)
- %A ... 住所1 ([Send%d] セクション-Address1 が差し込まれます)
- %B ... 住所2 ([Send%d] セクション-Address2 が差し込まれます)
- %H ... 電話番号 ([Send%d] セクション-Telephone が差し込まれます)
- %T ... FAX 番号 ([Send%d] セクション-Fax が差し込まれます)
- %K ... 敬称 ([Send%d] セクション-Title が差し込まれます)
- %P ... ページ数 (自動的に計算され、差し込まれます)
- %x ... xxxx ([Send%d] セクション- が差し込まれます)

ファクシミリ送付のご案内

%S
%D
%Y
%N %K

メガソフト株式会社
メガ太郎

毎度格別のお引き立てにあずかり、まことにありがとうございます。
下記の書類を拝送しますので、よろしくご査収下さいますようお願い申し上げます。

記

以上

5.3 サンプルプログラム

サンプルプログラムは、3種類のプログラムを用意しています。それぞれ、元となるサンプルプログラムがあり、FAX 送信に関する部分を[メール de FAX]の仕組みに変更して、クライアント動作するようにしています。

[メール de FAX]以外の処理については、元の各種マニュアルをご覧ください。

- FAX 送信する (SendFAX.exe)

元のプログラムは、「STARFAX Server SDK プログラミング(OCX 利用版)」の
2.1 FAX 送信 をご覧ください。

- 印刷結果の FAX 送信 (PrtCli.exe)

元のプログラムは、「STARFAX Server SDK VB 開発向けプリンタドライバとビューア」の
2.1 印刷結果の FAX 送信 をご覧ください。

- TIFF ファイルの作成と FAX 送信 (PrtCli2.exe)

元のプログラムは、「STARFAX Server SDK VB 開発向けプリンタドライバとビューア」の
2.2 TIFF ファイルの作成と FAX 送信 をご覧ください。

5.3.1 FAX 送信

FAX 送信プログラム【SendFax.exe】は、STARFAX Server SDK 本体をセットアップしていないコンピュータから FAX の送信を行うサンプルプログラムです。本 CD-ROM の以下の位置に入っています。

¥サンプル¥応用サンプル¥クライアント¥メール de F A X¥VB6¥SendFax.exe
... FAX 送信プログラム
¥サンプル¥応用サンプル¥クライアント¥メール de F A X¥VB6¥SendFax¥
... FAX 送信プログラム 開発プロジェクト

なお、このサンプルプログラムには、元となるサンプルプログラムがあり、FAX 送信に関する部分を[メール de F A X]の仕組みに変更して、クライアント動作するようにしています。
[メール de F A X]以外の処理については、「STARFAX Server SDK プログラミング(OCX 利用版)」の 2.1 FAX 送信 をご覧ください。

主な仕様、および操作方法は以下の通りです。

(以降、STARFAX Server SDK 本体をセットアップしているコンピュータを [サーバー側]、STARFAX Server SDK 本体をセットアップしていないコンピュータを [クライアント側] とします)

① [サーバー側]で [メール de F A X] の設定を有効にします。

(1) タスクトレイの STARFAX Server SDK のアイコンを右クリックして表示されるメニューから [メール de F A X] を実行します。

(2) 最低限、以下の項目を設定します。

- [メール de F A X を有効にする(Y)] をチェックします。
- [メールサーバーの設定(S)] を行います。

(3) 必要に応じて、以下の項目を設定します。

- [件名に含まれる文字で識別(I)]
- [メール受信間隔(N)]
- [対象外のメールを EML ファイルに保存(P)]、及び、関連項目
- [FAX 送信結果をメール通知する(N)]、及び、関連項目

② [サーバー側]で STARFAX Server SDK を起動します。

(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)

③ [クライアント側] で FAX 送信プログラム【SendFax.exe】を起動します。
(起動時の作業(カレント)フォルダの指定は特にありません)

④ 各種送信内容を指定します。

- [相手先(S)] ボタンを押して、相手先を指定します。
- [原稿(D)] ボタンを押して、原稿を指定します。
- 必要であれば、[送付状(C)] ボタンを押して、送付状を指定します。
- 必要であれば、[発信元(U)] ボタンを押して、発信元情報を指定します。

⑤ [メール送信(G)] ボタンを押して、メール送信を行います。

この後、FAX 送信が正常に動作していない場合は、[サーバー側] の以下の表示をご確認下さい。

- STARFAX Server SDK ログ管理プログラム の [イベント]
(主に、[コントロール] と [メール] をご確認ください)
 - [メール de FAX 設定] の システムメニュー の [イベントログ]
-

Form1.frm :

```
'*****  
' メール送信ボタン: クリック  
'*****
```

```
Private Sub CommandOK_Click()
```

~

```
'/////////////////////////////////////////  
'// (2) [ メール de FAX ]の送信命令ファイル作成  
'//  
'// SFCSSendFax コントロール 《 [ メール de FAX ]の送信命令ファイル作成 》  
'// Function MakeTransFileForMailToFax(TransFile As String,  
'// CoverName As String,  
'// FontName As String,  
'// FontSize As Integer  
'// UserInfo As String,  
'// UserID As String)as Long  
'//
```

```
IRet = SFCSSendFax1.MakeTransFileForMailToFax(strTrans, strCover, strCoverFontName,  
sCoverFontSize, strUserInfo, strUserID)  
If IRet = SFCSENDFAX_SUCCESS Then
```

```
MsgBox "このサンプルプログラムは MAPI を利用してメール送信を行います。" &  
vbCrLf & "宛先 に STARFAX Engine の [ メール de FAX ] で設定した" & vbCrLf & "POP  
メールアドレス を指定して送信してください。" & vbCrLf & vbCrLf & "なお、送信後、OUTLOOK  
等メーラーを起動して送信を行わないと" & vbCrLf & "実際に送信されないことがあります。  
", vbInformation, s_cstTITLE
```

```
'/////////////////////////////////////////  
'// メール送信  
'/////////////////////////////////////////
```

```
MAPISession1.SignOn
```

```
With MAPIMessages1
```

```
' セッション確立
```

```

.SessionID = MAPISession1.SessionID
.Compose

'// 件名、本文
.MsgSubject = "Mail to FAX"
.MsgNoteText = "Mail to FAX"

' 送信命令ファイル
.AttachmentIndex = .AttachmentCount
.AttachmentPathName = strTrans
.AttachmentPosition = .AttachmentCount

' 送信原稿
If TextDOC1.Text <> "" Then
    .AttachmentIndex = .AttachmentCount
    .AttachmentPathName = TextDOC1.Text
    .AttachmentPosition = .AttachmentCount
End If
If TextDOC2.Text <> "" Then
    .AttachmentIndex = .AttachmentCount
    .AttachmentPathName = TextDOC2.Text
    .AttachmentPosition = .AttachmentCount
End If
If TextDOC3.Text <> "" Then
    .AttachmentIndex = .AttachmentCount
    .AttachmentPathName = TextDOC3.Text
    .AttachmentPosition = .AttachmentCount
End If
If TextDOC4.Text <> "" Then
    .AttachmentIndex = .AttachmentCount
    .AttachmentPathName = TextDOC4.Text
    .AttachmentPosition = .AttachmentCount
End If

' 送付状
If strCover <> "" Then
    .AttachmentIndex = .AttachmentCount
    .AttachmentPathName = strCover
    .AttachmentPosition = .AttachmentCount
End If

'// メール送信
.Send True

```

End With

'セッション開放

MAPISession1.SignOff

Else

If IRet = SFCSENDFAX_ERR_GetTempFolder Then

MsgBox "一時フォルダの取得に失敗しました", vbExclamation, s_cstTITLE

End If

If IRet = SFCSENDFAX_ERR_GetTempFile Then

MsgBox "一時ファイルの取得に失敗しました", vbExclamation, s_cstTITLE

End If

If IRet = SFCSENDFAX_ERR_CreateTransFile Then

MsgBox "送信命令ファイルの作成に失敗しました", vbExclamation, s_cstTITLE

End If

If IRet = SFCSENDFAX_ERR_PARAM_INFO Then

MsgBox "パラメータエラー: 送信命令ファイル作成情報が指定されていません。", _
vbExclamation, s_cstTITLE

End If

If IRet = SFCSENDFAX_ERR_PARAM_SENDDNUM Then

MsgBox "パラメータエラー: 相手先数の指定が 0 です。", vbExclamation, s_cstTITLE

End If

If IRet = SFCSENDFAX_ERR_PARAM_SENDINFO Then

MsgBox "パラメータエラー: 相手先情報が指定されていません。", _
vbExclamation, s_cstTITLE

End If

If IRet = SFCSENDFAX_ERR_PARAM_FAX Then

MsgBox "パラメータエラー: FAX 番号が指定されていません。", vbExclamation, s_cstTITLE

End If

If IRet = SFCSENDFAX_ERR_PARAM_TRANSFILE Then

MsgBox "パラメータエラー: 送信命令ファイルが指定されていません。", _
vbExclamation, s_cstTITLE

End If

End If

~

End Sub

5.3.2 印刷結果のFAX送信

印刷結果のFAX送信プログラム【PrtCli.exe】は、STARFAX Server SDK 本体をセットアップしていないコンピュータで印刷結果の表示とFAX送信を行うサンプルプログラムです。本CD-ROMの以下の位置に入っています。ワード・エクセル等のアプリケーションから手動で印刷後、プリンタドライバからユーザープログラムが起動されます。

¥サンプル¥応用サンプル¥クライアント¥メール de FAX¥VB6¥PrtCli.exe
... 印刷結果のFAX送信プログラム
¥サンプル¥応用サンプル¥クライアント¥メール de FAX¥VB6¥PrtCli¥
... 印刷結果のFAXプログラム 開発プロジェクト

なお、このサンプルプログラムには、元となるサンプルプログラムがあり、FAX送信に関する部分を[メール de FAX]の仕組みに変更して、クライアント動作するようにしています。[メール de FAX]以外の処理については、「STARFAX Server SDK VB 開発向けプリンタドライバとビューア」の2.1 印刷結果のFAX送信をご覧ください。

主な仕様、および操作方法是以下の通りです。

(以降、STARFAX Server SDK 本体をセットアップしているコンピュータを[サーバー側]、STARFAX Server SDK 本体をセットアップしていないコンピュータを[クライアント側]とします)

① [サーバー側]で[メール de FAX]の設定を有効にします。

(1) タスクトレイのSTARFAX Server SDKのアイコンを右クリックして表示されるメニューから[メール de FAX]を実行します。

(2) 最低限、以下の項目を設定します。

- [メール de FAX を有効にする(Y)] をチェックします。
- [メールサーバーの設定(S)] を行います。

(3) 必要に応じて、以下の項目を設定します。

- [件名に含まれる文字で識別(I)]
- [メール受信間隔(N)]
- [対象外のメールをEMLファイルに保存(P)]、及び、関連項目
- [FAX送信結果をメール通知する(N)]、及び、関連項目

- ② [サーバー側]で STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
- ③ [クライアント側]で、以下の STARFAX Server SDK プリンタドライバの動作に関するレジストリを指定します。
- HKEY_LOCAL_MACHINE¥Software¥MEGASOFT¥SfCs¥OutFolder ... ファイル出力フォルダ
文字列項目で、任意の作業フォルダを作成して指定します。
(例: "C:¥Program Files¥SfCs¥Temp")
 - HKEY_LOCAL_MACHINE¥Software¥MEGASOFT¥SfCs¥DocName ... ドキュメント名
文字列項目で、このサンプルプログラムの場合、任意の文字列を指定します。
(例: "SFCSPRN")
 - HKEY_LOCAL_MACHINE¥Software¥MEGASOFT¥SfCs¥ExecFlag ... プログラム実行フラグ
DWORD 項目で、1 を指定します。
 - HKEY_LOCAL_MACHINE¥Software¥MEGASOFT¥SfCs¥ExecPath ... プログラムパス
文字列項目で、PrtCli.exe をフルパスで指定します。
(例: "C:¥Program Files¥SfCs¥PrtCli.exe")
 - HKEY_LOCAL_MACHINE¥Software¥MEGASOFT¥SfCs¥ExecParam ... 追加パラメータ
文字列項目で、何も指定していない状態("")を設定します。
- ④ 印刷可能な適当なアプリケーション(ワード等)から プリンタ名 "MEGASOFT STARFAX Server SDK" に対して印刷を行うと 印刷結果の F A X 送信プログラム【PrtCli.exe】が起動して、印刷結果リスト に 印刷結果が 登録された状態になります。
- ⑤ [表示(V)]ボタンを押して、印刷結果の内容を確認します。
- ⑥ [メール送信(S)]ボタンを押して、メール送信を行います。
この後、FAX 送信が正常に動作していない場合は、[サーバー側] の以下の表示をご確認下さい。
- STARFAX Server SDK ログ管理プログラム の [イベント]
(主に、[コントロール] と [メール] をご確認ください)
 - [メール de F A X 設定] の システムメニュー の [イベントログ]
-

Form1.frm :

```
'*****  
' メール送信ボタン: クリック  
'*****
```

```
Private Sub CommandOK_Click()
```

~

```
' _____  
' [ メール de F A X ] の送信命令ファイル作成  
' _____
```

```
IRet = SFCSSendFax1.MakeTransFileForMailToFax(strTrans, "", "", 0, "", "")  
If IRet = SFCSENDFAX_SUCCESS Then
```

```
MsgBox
```

```
"このサンプルプログラムは MAPI を利用してメール送信を行います。" & vbCrLf & "宛先 に  
STARFAX Engine の [ メール de F A X ] で設定した" & vbCrLf & "POP メールアドレス を  
指定して送信してください。" & vbCrLf & vbCrLf & "なお、送信後、OUTLOOK 等メーラーを  
起動して送信を行わないと" & vbCrLf & "実際に送信されないことがあります。", vbInformation,  
s_cstTITLE
```

```
' ///////////////////////////////////  
' // メール送信  
' ///////////////////////////////////
```

```
MAPISession1.SignOn
```

```
With MAPIMessages1
```

```
    ' セッション確立  
    .SessionID = MAPISession1.SessionID  
    .Compose
```

```
    ' // 件名、本文  
    .MsgSubject = "Mail to FAX"  
    .MsgNoteText = "Mail to FAX"
```

```
    ' 送信命令ファイル  
    .AttachmentIndex = .AttachmentCount  
    .AttachmentPathName = strTrans
```

```
.AttachmentPosition = .AttachmentCount
```

```
' 送信原稿
```

```
.AttachmentIndex = .AttachmentCount
```

```
.AttachmentPathName = s_strFileName
```

```
.AttachmentPosition = .AttachmentCount
```

```
' // メール送信
```

```
.Send True
```

```
End With
```

```
' セッション開放
```

```
MAPISession1.SignOff
```

```
~
```

```
Else
```

```
  If IRet = SFCSENDFAX_ERR_GetTempFolder Then
```

```
    MsgBox "一時フォルダの取得に失敗しました", vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_GetTempFile Then
```

```
    MsgBox "一時ファイルの取得に失敗しました", vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_CreateTransFile Then
```

```
    MsgBox "送信命令ファイルの作成に失敗しました", vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_PARAM_INFO Then
```

```
    MsgBox "パラメータエラー: 送信命令ファイル作成情報が指定されていません。", _  
      vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_PARAM_SENDNUM Then
```

```
    MsgBox "パラメータエラー: 相手先数の指定が 0 です。", vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_PARAM_SENDINFO Then
```

```
    MsgBox "パラメータエラー: 相手先情報が指定されていません。", _  
      vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_PARAM_FAX Then
```

```
    MsgBox "パラメータエラー: FAX 番号が指定されていません。", vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_PARAM_TRANSFILE Then
```

```
    MsgBox "パラメータエラー: 送信命令ファイルが指定されていません。", _  
      vbExclamation, s_cstTITLE
```

End If
End If

~

End

End Sub

5.3.3 TIFF ファイルの作成と F A X 送信

TIFF ファイルの F A X 送信プログラム【PrtCli2.exe】は、STARFAX Server SDK 本体をセットアップしていないコンピュータで TIFF ファイルの作成 と FAX 送信を行うサンプルプログラムです。本 CD-ROM の以下の位置に入っています。ユーザープログラムがプリンタドライバを制御して印刷結果 (TIFF ファイル) を取得します。

¥サンプル¥応用サンプル¥クライアント¥メール de F A X¥VB6¥PrtCli2.exe
... TIFF ファイルの F A X 送信プログラム
¥サンプル¥応用サンプル¥クライアント¥メール de F A X¥VB6¥PrtCli2¥
... TIFF ファイルの F A X 送信プログラム 開発プロジェクト

なお、このサンプルプログラムには、元となるサンプルプログラムがあり、FAX 送信に関する部分を [メール de F A X] の仕組みに変更して、クライアント動作するようにしています。
[メール to F X] 以外の処理については、「STARFAX Server SDK VB 開発向けプリンタドライバとビューア」の 2.2 TIFF ファイルの作成と FAX 送信 をご覧ください。

主な仕様、および操作方法は以下の通りです。

(以降、STARFAX Server SDK 本体をセットアップしているコンピュータを [サーバー側]、STARFAX Server SDK 本体をセットアップしていないコンピュータを [クライアント側] とします)

① [サーバー側] で [メール de F A X] の設定を有効にします。

(1) タスクトレイの STARFAX Server SDK のアイコンを右クリックして表示されるメニューから [メール de F A X] を実行します。

(2) 最低限、以下の項目を設定します。

- [メール de F A X を有効にする (Y)] をチェックします。
- [メールサーバーの設定 (S)] を行います。

(3) 必要に応じて、以下の項目を設定します。

- [件名に含まれる文字で識別 (I)]
- [メール受信間隔 (N)]
- [対象外のメールを EML ファイルに保存 (P)]、及び、関連項目
- [FAX 送信結果をメール通知する (N)]、及び、関連項目

- ② [サーバー側]で STARFAX Server SDK を起動します。
(STARFAX Server SDK の起動に関しては、「セットアップマニュアル」P21 参照)
- ③ [クライアント側]で PrtCli2.exe を起動します。
- ④ 「操作1」の下欄にFAX 原稿に表示させる文字を入力します。
- ⑤ 「①FAX 原稿の作成」ボタンをクリックします。
(作成するファイル名を任意指定したい場合は、[指定]ラジオボタンをクリックして、ファイル名を入力してください)
- ⑥ 「②FAX 原稿の表示」ボタンをクリックすると、作成されたFAX 原稿がSTARFAX Server SDK ビューアで表示されます。
- ⑦ 「操作3」の下欄に送信先のFAX 番号を入力します。
- ⑧ 「メール送信」ボタンをクリックします。
この後、FAX 送信が正常に動作していない場合は、[サーバー側] の以下の表示をご確認下さい。
- STARFAX Server SDK ログ管理プログラム の [イベント]
(主に、[コントロール] と [メール] をご確認ください)
 - [メール de FAX 設定] の システムメニュー の [イベントログ]
-

Form1.frm :

```
'*****  
' [メール送信]ボタン: クリック  
'*****
```

Private Sub CommandOK_Click()

~

```
' _____  
' [ メール de F A X ]の送信命令ファイル作成  
' _____
```

lRet = SFGSSendFax1.**MakeTransFileForMailToFax**(strTrans, "", "", 0, "", "")

If lRet = SFGSENDFAX_SUCCESS Then

MsgBox "このサンプルプログラムは MAPI を利用してメール送信を行います。" &
vbCrLf & "宛先 に STARFAX Engine の [メール de F A X] で設定した" & vbCrLf & "POP
メールアドレス を指定して送信してください。" & vbCrLf & vbCrLf & "なお、送信後、OUTLOOK
等メーラーを起動して送信を行わないと" & vbCrLf & "実際に送信されないことがあります。
", vbInformation, s_cstTITLE

```
' ///////////////////////////////////  
' // メール送信  
' ///////////////////////////////////
```

MAPISession1.SignOn

With MAPIMessages1

 ' セッション確立

 .SessionID = MAPISession1.SessionID
 .Compose

 '// 件名、本文

 .MsgSubject = "Mail to FAX"
 .MsgNoteText = "Mail to FAX"

 ' 送信命令ファイル

 .AttachmentIndex = .AttachmentCount
 .AttachmentPathName = strTrans


```
.AttachmentPosition = .AttachmentCount
```

```
' 送信原稿
```

```
.AttachmentIndex = .AttachmentCount
```

```
.AttachmentPathName = s_strTIFFFILE
```

```
.AttachmentPosition = .AttachmentCount
```

```
' // メール送信
```

```
.Send True
```

```
End With
```

```
' セッション開放
```

```
MAPISession1.SignOff
```

～

```
Else
```

```
  If IRet = SFCSENDFAX_ERR_GetTempFolder Then
```

```
    MsgBox "一時フォルダの取得に失敗しました", vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_GetTempFile Then
```

```
    MsgBox "一時ファイルの取得に失敗しました", vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_CreateTransFile Then
```

```
    MsgBox "送信命令ファイルの作成に失敗しました", vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_PARAM_INFO Then
```

```
    MsgBox "パラメータエラー: 送信命令ファイル作成情報が指定されていません。", _  
      vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_PARAM_SENDDNUM Then
```

```
    MsgBox "パラメータエラー: 相手先数の指定が 0 です。", vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_PARAM_SENDINFO Then
```

```
    MsgBox "パラメータエラー: 相手先情報が指定されていません。", _  
      vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_PARAM_FAX Then
```

```
    MsgBox "パラメータエラー: FAX 番号が指定されていません。", vbExclamation, s_cstTITLE
```

```
  End If
```

```
  If IRet = SFCSENDFAX_ERR_PARAM_TRANSFILE Then
```

```
    MsgBox "パラメータエラー: 送信命令ファイルが指定されていません。", _  
      vbExclamation, s_cstTITLE
```

End If
End If

~

End Sub