



IFM/PTM シリーズ
プリンタードライバー
取扱説明書

セイコーインスツル株式会社

Copyright © セイコーインスツル株式会社 2009-2021

無断転載を禁じます。

本書の内容は、断りなく変更することがあります。

本書及び本書に記載された製品の利用によって発生した損害
及びその回復に要する費用に対し、当社は一切の責任を負いかねます。

SII ● は、セイコーインスツル株式会社の登録商標です。

Microsoft®及びWindows®は、Microsoft Corporationの
米国及びその他の国における登録商標です。

目次

1. はじめに	1
1.1 概要	1
1.2 操作説明について	1
1.3 オペレーティングシステム略称について	1
1.4 プリンターフォルダーの表記について	1
1.5 その他の表記について	2
1.6 対象機種とドライバーモデル	2
2. 動作環境	3
2.1 システム環境	3
2.2 機能設定	4
2.3 注意事項	4
3. インストール	5
3.1 概要	5
3.2 新規インストール方法	6
3.2.1 インストール共通手順	6
3.2.2 USB 接続でのインストール	7
3.2.3 シリアル接続でのインストール	9
3.2.4 共有プリンターのインストール	10
4. 印刷設定	12
4.1 概要	12
4.2 印刷設定画面	12
4.3 詳細設定項目	12
4.3.1 用紙サイズ	12
4.3.2 部数	12
4.3.3 ハーフトーン	12
4.3.4 印刷濃度	12
4.3.5 用紙排出	13
4.3.6 カッタ制御	13
4.3.7 印刷モード	13
4.3.8 機能設定	13
5. プロパティ設定	14
5.1 概要	14
5.2 プロパティ画面	14
5.3 ポート シート	14
5.4 デバイスの設定シート	15
5.5 バージョンシート	15
6. 用紙設定	16
6.1 概要	16
6.2 用語	16
6.3 用紙サイズ	17
6.4 余白	18
6.5 カスタム用紙の登録方法	19
7. 通信ライブラリ	20
7.1 概要	20
7.2 機能	20

IFM/PTM プリンタードライバー

7.3	ライブラリファイル	21
7.4	関数	22
7.5	通信ライブラリ関数詳細	23
7.6	応答データ形式	38
7.7	ASB 応答に関する注意事項	39
7.8	関数戻り値(エラーコード)	40
7.9	サンプルプログラム	40
8.	免責	41

1. はじめに

1.1 概要

本書は、セイコーインスツル株式会社（以下 SIIと呼びます）がSII製サーマルプリンター用に提供するMicrosoft Windows向けソフトウェアの仕様、機能、操作方法について説明します。

1.2 操作説明について

本書で説明される操作方法及び画面表示等につきましては、オペレーティングシステムがインストール時の標準設定であることを前提で説明しています。

オペレーティングシステムの設定を変更した場合、操作説明や画面表示が異なる場合があります。

1.3 オペレーティングシステム略称について

本書で説明されるオペレーティングシステムの名称はそれぞれ

◆ Microsoft® Windows®の全般	⇒ Windows
◆ Microsoft® Windows® 11	⇒ Windows 11
◆ Microsoft® Windows® 10	⇒ Windows 10
◆ Microsoft® Windows Server® 2019	⇒ Windows 10(Windows Server 2019)
◆ Microsoft® Windows Server® 2016	⇒ Windows 10(Windows Server 2016)
◆ Microsoft® Windows® 8.1	⇒ Windows 8.1
◆ Microsoft® Windows Server® 2012	⇒ Windows Server 2012

として略称しています。

個別で説明する必要がある場合は、略称の後ろに記載した括弧書きの内容で説明します。

1.4 プリンターフォルダーの表記について

本書での説明される[プリンターフォルダー]とは下記の操作により表示されるフォルダーを意味します。

[コントロールパネル]⇒[ハードウェアとサウンド]⇒[デバイスとプリンター]の順にクリックして表示される[デバイスとプリンター]フォルダー。

1.5 その他の表記について

本書で使用する下記の用語について、記載します。

用 語	内 容
プリンタードライバー	提供するソフトウェアに含まれるプリンタードライバー
通信ライブラリ	提供するソフトウェアに含まれる、プリンターと通信するためのアプリケーションに組込むダイナミックリンクライブラリ
製品技術説明書	各対象製品のインタフェースボード技術説明書またはチップセット技術説明書
機能設定	製品技術説明書に記載がある[機能設定]の機能
ASB	プリンターのコマンドである[自動ステータス送信の有効・無効]コマンドにより取得できるプリンターステータスの応答

1.6 対象機種とドライバーモデル

プリンタードライバーがサポートする製品(インタフェースまたはCPU)の一覧を説明します。

製品名	メカニズムモデル	ドライバーモデル
IFM201-01UK-E IFM201-01SK-E (インタフェース)	CAPM347	SII IFM20x
PTM20P01-E (CPU)	CAPM347	SII PTM20

※CPUタイプを利用の場合、印刷モードはスタンダードモードでのみ動作します。

([詳細オプション]での選択項目は表示されません。)

それ以外の機能及び仕様はドライバーモデルに関係なく共通です。

2. 動作環境

2.1 システム環境

プリンタードライバーが対応するシステム環境を記載します。

項 目	仕 様
オペレーティングシステム	■ Windows 11 (64ビット) ■ Windows 10 (32ビット及び64ビット) ■ Windows Server 2019 (64ビット) ■ Windows Server 2016 (64ビット) ■ Windows 8.1 (32ビット及び64ビット) ■ Windows Server 2012 (64ビット) Modern UIはサポートしていません。
通信方式	■ USB通信 ■ シリアル通信(RS-232C)

2.2 機能設定

機能設定の利用条件を記載します。

- ◆ プリンタードライバーを利用する場合、機能設定の下記の項目については次の設定であることが利用条件となります。
- ◆ プリンタードライバーの機能により機能設定を下記条件以外の設定にすることも可能ですが、その場合プリンタードライバーが正常に動作できなくなる恐れがあります。
- ◆ 機能設定の変更方法については、"5.4 デバイスの設定シート"を参照してください。

DIP番号	項目名	設定条件
DIP5 - 1	自動ステータス応答機能選択	0:有効 ※1
DIP5 - 2	イニシャライズ完了応答選択	0:有効 ※2
DIP5 - 3	エラー時データ破棄選択	1:無効
DIP40 - 1、2	シリアルボーレート選択	11:115200
DIP40 - 5、6	シリアル制御方式選択	01:Busy ※1
DIP40 - 7	ホストビジー制御選択	0:CTS
DIP40 - 8	ブ레이크信号選択	1:RxD ※1

※1 :プリンタードライバーを利用する上で必須設定となります。

※2 :GetSiiPrinterDataで[イニシャライズ完了応答]の受信履歴を取得する場合の設定条件となります。

2.3 注意事項

- ※ プリンターが実装しているフォントをプリンタードライバーから印刷することはできません。
- ※ 印刷機能についてはプリンターの共有機能によるネットワーク接続でも利用できます。
(通信ライブラリを利用した機能はローカル接続のみの利用となります。)
- ※ 「ログオフ」機能である「ユーザの切り替え機能」を利用したマルチユーザ環境でのサポートは対応していません。
- ※ プリンターの用紙幅設定とプリンタードライバーの用紙幅が合致しない場合、印刷内容が崩れる場合があります。
- ※ Windows の標準機能のスリープモードはサポートしません。

3. インストール

3.1 概要

本章ではプリンタードライバーのインストールについて説明します。

インストール方法には以下の種類があります。

- [プリンターフォルダー]の[プリンターの追加]からのインストール
- 専用インストーラーからのインストール

本章では専用インストーラーからのインストール手順のみ説明します。

注意

- ◆ インストール作業にはコンピューターの管理者権限が必要です。
- ◆ インストール中に[Windowsロゴテスト]についての警告が表示される場合がありますが、そのままインストールを続行してください。
- ◆ WOW64環境で通信ライブラリを利用する場合、及び通信ライブラリの.NET APIを利用する場合は専用インストーラーからインストールを行ってください。

参考

- プリンタードライバーが不要となった場合は、コントロールパネルの[プログラムと機能]から[Printer Driver for SII IFM Series]を選択し、アンインストールを行ってください。

3.2 新規インストール方法

以下に新規インストール方法について説明します。

本ソフトウェアは、セットアッププログラム「SetupPrinter(64).exe」からインストールしてください。

注意

- ◆ 同一の通信ポートに複数のドライバーをインストールしないでください。

3.2.1 インストール共通手順

以下に各接続で共通するインストール開始手順について説明します。

1. [インストーラー起動]画面が表示されますので[次へ(N) >]ボタンをクリックします。

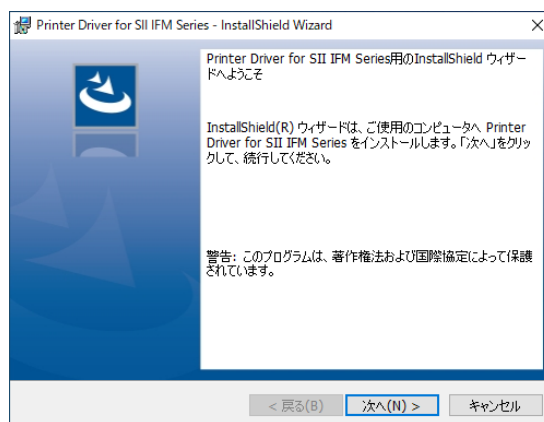


図 3-1 [インストーラー起動]画面

2. [ソフトウェア使用許諾契約書]が表示されますので、内容を確認して「使用許諾契約の条項に同意します (A)」を選択し、[次へ(N) >]ボタンをクリックします。

3. [インストール確認]画面が表示されますので、[インストール(I)]ボタンをクリックします。

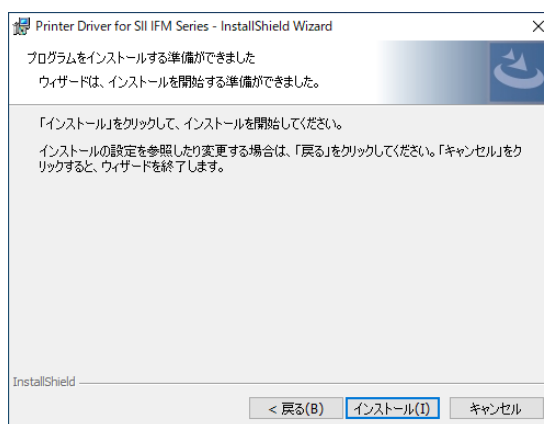


図 3-2 [インストール確認]画面

以降の手順についてはプリンターとの接続方式により異なります。接続方式がUSB接続の場合『3.2.2 USB接続でのインストール』、シリアル接続の場合『3.2.3 シリアル接続でのインストール』、共有プリンターを利用する場合は『3.2.4 共有プリンターのインストール』へ進んでください。

3.2.2 USB接続でのインストール

以下にUSB接続でのインストール手順について説明します。

注意

- ◆ USB接続の場合、本書の指示があるまではプリンターの電源は切っておいてください。

4. [インストール形式の選択]画面が表示されますので、「プラグアンドプレイインストール」を選択し、[Install]ボタンをクリックします。

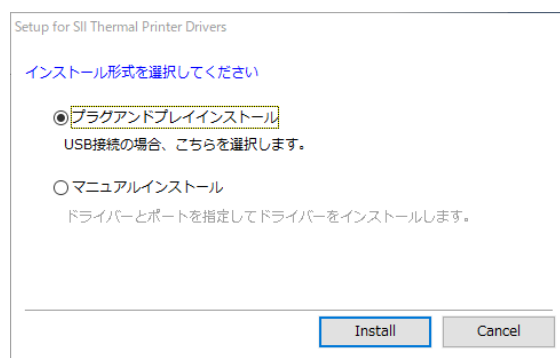


図 3-3 [インストール形式の選択]画面(USB接続)

5. [完了]画面が表示されたら、プリンターとコンピューターをUSBケーブルで接続し、プリンターの電源を入れてください。

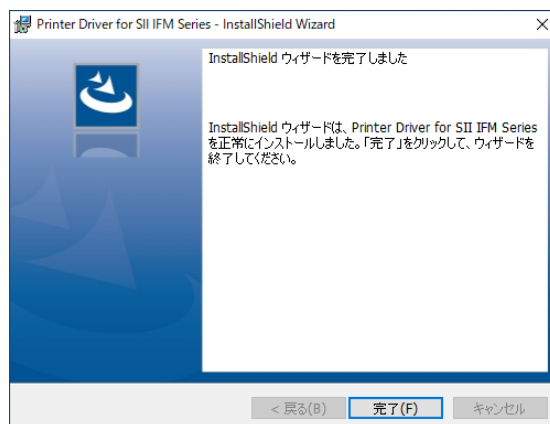


図 3-4 [完了]画面(USB 接続)

IFM/PTM プリンタードライバー

6. プリンターがコンピューターに認識されると、プラグアンドプレイによりプリンタードライバーが自動的にインストールされます。

注意

- ◆ プリンタードライバーのインストール完了後に取り外したUSBケーブルを再度接続する場合には、インストールした時と同一のUSBポートに接続してください。USBポートの位置を変更すると、コンピューター側で別のプリンターを接続したと認識し、再びプリンタードライバーのインストール要求が発生します。

3.2.3 シリアル接続でのインストール

以下にシリアル接続でのインストール手順について説明します。

4. [インストール形式の選択]画面が表示されますので、「マニュアルインストール」を選択し、インストールするドライバーと接続するポートを選択して[Install]ボタンをクリックしてインストールを開始します。

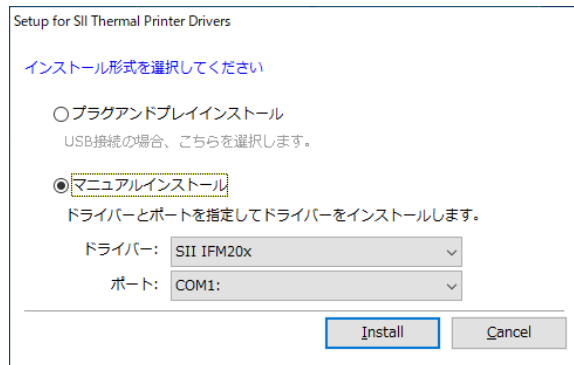


図 3-5 [インストール形式の選択]画面(シリアル接続)

5. インストールが終了すると、[完了]画面が表示されます。

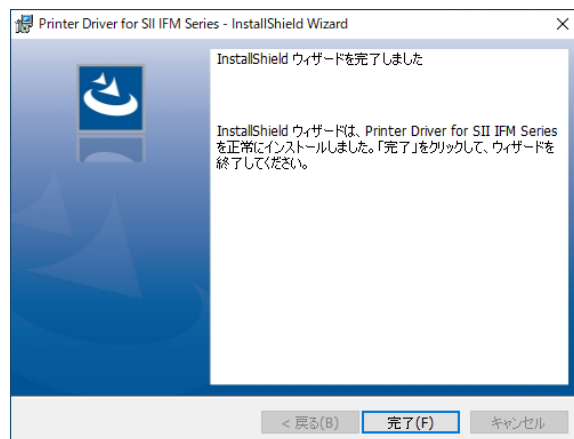


図 3-6 [完了]画面(シリアル接続)

3.2.4 共有プリンターのインストール

以下に共有プリンターのインストール手順について説明します。

4. [インストール形式の選択]画面が表示されますので、「マニュアルインストール」を選択し、「新しいポートの追加...」を選択します。

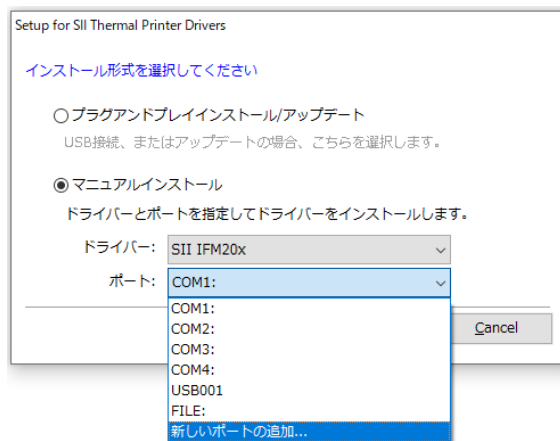


図 3-7 [インストール形式の選択]画面(共有プリンター)

5. [新しいポートの作成]画面が表示されますので、「ネットワークプリンターポート」を選択し、[プリンター]を設定して[OK]ボタンをクリックすると、[インストール形式の選択]画面のポートリストにポートを追加されます。

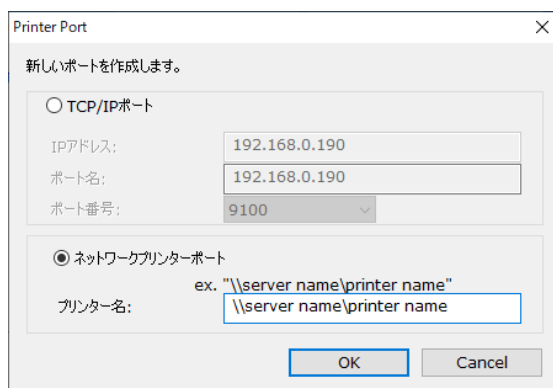


図 3-8 [新しいポートの作成]画面

6. インストール形式の選択]画面に戻りますので、インストールするドライバーと接続するポートを選択し、[Install]ボタンをクリックしてインストールを開始します。

IFM/PTM プリンタードライバー

7. インストールが終了すると、[完了]画面が表示されます。

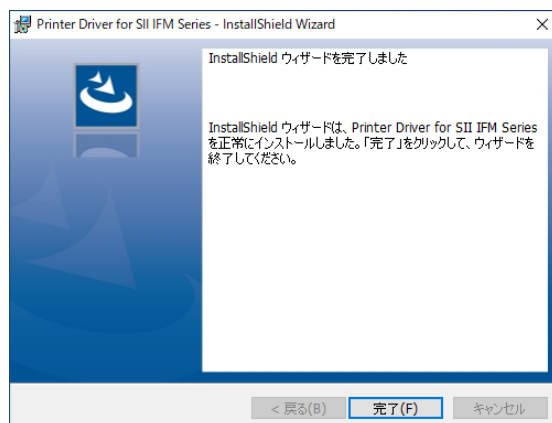


図 3-9 [完了]画面(共有プリンター)

4. 印刷設定

4.1 概要

用紙や印刷制御の設定については[印刷設定]画面より行います。

Modern UI designから印刷を行う場合は、[印刷設定]画面で設定可能な項目の一部しか表示されません。

4.2 印刷設定画面

[印刷設定]画面は以下の手順で表示できます。

1. プリンターフォルダー内に表示されているプリンターアイコンを右クリックし、サブメニューを表示します。
2. 表示したサブメニューの中から[印刷設定(G)]をクリックします。

4.3 詳細設定項目

詳細な印刷設定は[詳細オプション]画面から変更できます。

[詳細オプション]画面は以下の手順で表示できます。

1. "4.2 印刷設定画面"で説明した[印刷設定]画面を表示します。
2. [印刷設定]画面右下の[詳細設定(V)...]をクリックして[詳細オプション]を表示します。

4.3.1 用紙サイズ

用紙サイズを選択します。

新規に用紙を登録する方法については、"6.5 カスタム用紙の登録方法"を参照してください。

4.3.2 部数

印刷する部数を指定します。

4.3.3 ハーフトーン

印刷イメージの二値化方法を選択します。

4.3.4 印刷濃度

プリンター側で制御する印刷濃度設定を選択します。

4.3.5 用紙排出

印刷後の用紙排出動作を選択します。

- ◆ 有効 : 選択された用紙長まで用紙排出を行います。
- ◆ 無効 : 選択された用紙長に関係なく、イメージの終端まで用紙排出を行います。
但し、用紙排出動作は選択された用紙サイズを上限とします。

※マーク紙を利用する場合、必ず[無効]を選択してください。

※カット前後の紙送りについては[

※カッタ制御]に依存します。

※各用紙排出選択での用紙排出の動作有無を下記表に記載します。

用紙排出	下余白の紙送り
有効	有り
無効	無し

4.3.6 カッタ制御

印刷後の切断動作を選択します。

- ◆ フルカット : フルカットを行います。
- ◆ パーシャルカット : パーシャルカットを行います。
- ◆ カットなし : 切断動作を無効にします。

※[フルカット]または[パーシャルカット] 選択時には、カット前にカッターヘッド間の紙送りを行います。

※マーク紙を利用する場合、カッターヘッド間の紙送りはマーク検出コマンド処理前に実行されます。

※カット後にカット位置から順方向もしくは逆方向への紙送り動作を行います。この用紙送り動作は紙詰まりまたは印字詰まり防止のために行います。各カッタ制御選択での紙送り量は以下の通りです。

カッタ制御選択	カット後紙送り量
フルカット	(-) 6 mm
パーシャルカット	7 mm

(-)は逆方向の紙送りを示します。

4.3.7 印刷モード

印刷の処理モードを選択します。

- ◆ ページモード優先 : ページモードで印刷可能な印刷内容の場合はページモードで印刷を行い、それ以外はスタンダードモードで印刷を行います。
- ◆ スタンダードモード : スタンダードモードで印刷を行います。

※CPUタイプのモデルの場合、全て「スタンダードモード」で印刷します。

[印刷モード]の項目は表示されません。

※ページモード優先時では、下余白を除いた(上余白を含めた)全体のイメージサイズがページモードに展開可能なサイズである場合にページモードで印刷されます。

4.3.8 機能設定

機能設定ユーティリティを起動し、機能設定の状態表示及び変更を行います。

機能設定ユーティリティの具体的な説明については"5.4 デバイスの設定シート"の機能設定項目を参照してください。

5. プロパティ設定

5.1 概要

プリンタードライバーの設定は、[プロパティ]画面より行います。

5.2 プロパティ画面

[プロパティ]画面は以下の手順で表示できます。

1. プリンターフォルダー内に表示されているプリンターアイコンを右クリックでサブメニューを表示します。
2. 表示したサブメニューの中から[プリンターのプロパティ]をクリックします。

※[プロパティ]画面上の[全般]シートにある[テストページの印刷(T)]による印刷は、小型プリンター向け用紙幅に対しては正しく印刷できないため印刷内容が崩れてしまいますが、不具合ではありません。

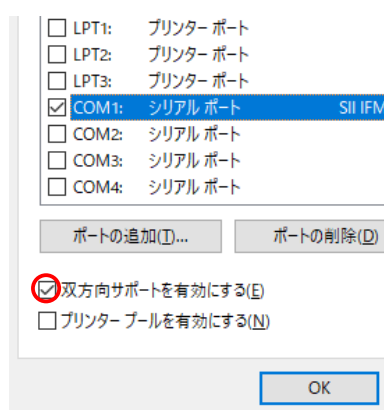
5.3 ポート シート

印刷するポートの設定を行います。ポートの設定をプリンター側の通信設定に合わせてください。

※[双方向サポートの機能を有効にする(E)]の設定はチェックしてください。

仮想ポート等を利用して正常動作できない場合、「双方向サポートの機能を有効にする(E)」のチェックは外してください。その際、双方向通信を利用した機能は利用できなくなります。

[プロパティ]-[ポート]シートの画面下側



シリアル通信設定の変更方法

- ◆ [COMx : シリアルポート]を選択の上、[ポートの構成(C)...]をクリックして表示される通信設定画面からプリンター側と同一の通信設定に合わせてください。
- ◆ 通信設定について、以下に注意事項を記載します。
 - ※ 設定にはコンピューターの管理者権限が必要です。
 - ※ [ビット/秒(B)](通信速度)を 115200bps でご利用ください。




IFM/PTM プリンタードライバー

- ※ 通信ライブラリを利用する場合、[フロー制御(F)]はハードウェアでご利用ください。
- ※ [データビット(D)]を8ビットでご利用ください。
- ※ Windows Server 2019、2016及び2012の場合は、[ポートの構成(C)]ボタンでは通信設定を行えません。コンピュータの「デバイスマネージャー」の「ポート(COMとLPT)」で通信設定を行ってください。

5.4 デバイスの設定シート

ユーティリティを起動し、プリンター設定を行います。

機能設定

- ◆ 機能設定の状態表示及び変更を行います。
機能設定の内容については、製品技術説明書を参照してください。
- ◆ 現在表示されている設定状態をファイルに保存することが可能です。
また保存されたデータは再度ユーティリティで読み込み可能となります。
- ◆ 機能設定の操作には管理者権限が必要です。
管理者権限のないユーザが機能設定の操作を行う場合、管理者権限にて以下の設定が必要です。
 1. [プロパティ]の[セキュリティ]シートを表示します。
 2. [グループ名またはユーザ名(G)]で[Everyone]を選択します。
 3. [Everyoneのアクセス許可(P)]で[このプリンターの管理]の[許可]にチェックを入れます。
- ◆ 機能設定の変更方法の一例を記載します。
 1. [機能設定]の[セットアップ...]をクリックすると現在の機能設定状態を表示します。
 2. 表示されている項目のうち変更したい項目をダブルクリックすると編集モードとなります。
 3. 設定の変更が完了した後、[適用(A)]をクリックして設定を書き込みます。
 4. リスト上に表示されている設定内容をファイルに保存する場合、 をクリックします。
 5. ファイルに保存した設定を読み出す場合、 をクリックします。
 6. 終了する場合、 ボタンまたは[終了(X)]をクリックします。

※機能設定の操作はプリンターが印刷可能状態でのみ利用可能となります。

5.5 バージョンシート

ドライバーのバージョンに関する情報を表示します。

6. 用紙設定

6.1 概要

プリンタードライバーで利用可能な用紙サイズについて説明します。

利用可能な用紙サイズには、予めインストール時に登録される用紙サイズとユーザが新規に登録可能な用紙サイズがあります。

ユーザが新規に用紙を登録する方法については、"6.5 カスタム用紙の登録方法"を参照してください。

プリンタードライバーで利用するドライバー用紙幅とは実際利用する用紙の幅ではなく、印字(ヘッド)幅を用紙幅として指定することになりますのでご注意ください。用紙幅については"6.3 用紙サイズ"も参照してください。

以降の説明では用紙排出方向を用紙長方向であることを前提として説明しています。

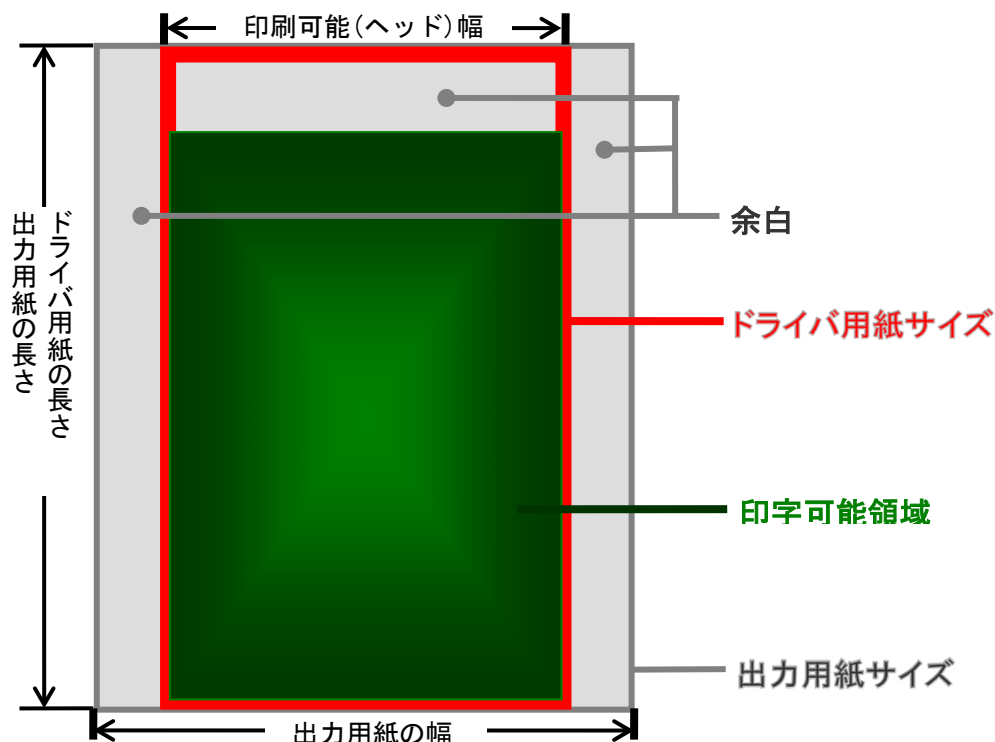
6.2 用語

用紙の説明で使用する用語について、記載します。

用 語	内 容
出力用紙(サイズ)	実際に出力される用紙(サイズ)
ドライバー用紙(サイズ)	コンピューター(プリンタードライバー)上で利用される用紙(サイズ) この用紙の幅は印刷可能(ヘッド)幅、及び用紙長は出力用紙(サイズ)となります。
余白	出力用紙に対して印刷ができない範囲
標準用紙	ドライバーインストール時に予め登録されるドライバー用紙サイズ
カスタム用紙	ユーザが新規に登録可能なドライバー用紙サイズ

6.3 用紙サイズ

出力用紙サイズとドライバー用紙サイズの関係を示したイメージ図を記載します。



各出力用紙幅における利用可能な用紙サイズの一覧を記載します。

出力用紙幅	標準用紙サイズ	カスタム用紙サイズ(範囲)	
58 mm	54 × 297 mm	幅	25.6 mm ~ 54 mm
		長	30 mm ~ 3276 mm
60 mm	54 × 297 mm 56 × 297 mm	幅	25.6 mm ~ 56 mm
		長	30 mm ~ 3276 mm
80 mm	54 × 297 mm 56 × 297 mm 72 × 297 mm	幅	25.6 mm ~ 72 mm
		長	30 mm ~ 3276 mm
82.55 mm	54 × 297 mm 56 × 297 mm 72 × 297 mm 80 × 297 mm A4 Letter	幅	25.6 mm ~ 80 mm
		長	30 mm ~ 3276 mm

※ 標準用紙として登録されている[A4]や[Letter]などプリンターに実装できる印刷用紙より大きいドライバー用紙幅も存在します。その様な標準用紙を選択した場合、コンピューター上では右側に大きく余白が発生します。

※ 用紙サイズの内容に対し用紙長についてはプリンターの機構上、若干の誤差が生じる可能性があります。

※ マーク紙を利用の場合、利用可能な用紙長は下記の通りである必要があります。

* 用紙長 < [マーク間の距離]-10mm

6.4 余白

用紙幅方向

用紙幅方向における余白サイズについて説明します。

用紙幅方向の余白サイズについては出力用紙幅に依存します。

出力用紙幅	左右余白合計	ドライバー用紙幅
58 mm	4 mm	54 mm
60 mm	4 mm	56 mm
80 mm	8 mm	72 mm
83 mm	3 mm	80 mm

※ 用紙長について「出力用紙」と「ドライバー用紙」とでは共通となります。

※ ここで説明する余白とは出力用紙に対して印刷ができない範囲であり、ドライバー用紙サイズからの領域ではありません。

※ 余白についての説明はA4またはLetterの場合を除きます。

用紙長方向

用紙長方向における余白サイズについて説明します。

用紙長方向の余白サイズについては[

カット制御]の選択に依存します。

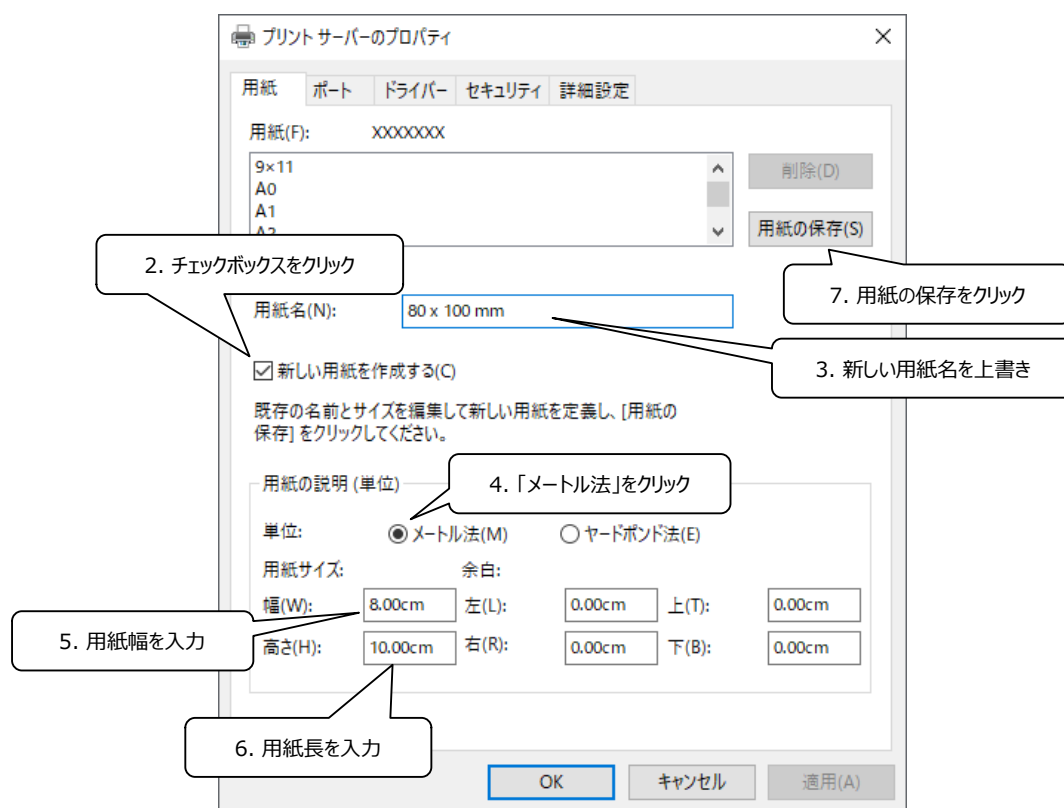
カット制御選択	上側余白	下側余白
フルカット	8.2 mm	0.5 mm
パーシャルカット	21.2 mm	0.5 mm
カット無し	2.5 mm	0 mm

※ マーク紙を利用の場合、プリンター側の上下余白サイズは上記表の値と異なります。

6.5 カスタム用紙の登録方法

カスタム用紙を登録する方法について説明します。

1. [プリントサーバーのプロパティ]画面上にある[用紙]シートを表示します。
2. [新しい用紙を作成する(C)]をクリックします。
3. 用紙名を[用紙名(N)]に上書きします。
4. [メートル法(M)]をクリックします。
5. [用紙の説明(単位)]の[幅(W)]にカスタム用紙の用紙幅を入力します。
6. [用紙の説明(単位)]の[高さ(H)]にカスタム用紙の用紙長を入力します。
7. 最後に[用紙の保存(S)]をクリックして登録は完了です。



- ※ カスタム用紙は、「6.3 用紙サイズ」の[カスタム用紙サイズ(範囲)]を考慮して登録してください。
- ※ 用紙排出方向の印刷領域長さとは用紙サイズ長さは異なります。
- ※ 余白のサイズ 0 で入力した場合、既定の余白サイズが適用になります。
- ※ カスタム用紙の登録にはコンピュータの管理者権限が必要です。
- ※ [用紙]シート上に表示されている用紙の中で「6.3 用紙サイズ」の[カスタム用紙サイズ(範囲)]に記載された内容に該当する用紙のみプリンタードライバーで利用できます。
- ※ [サーバーのプロパティ]は、以下の手順で表示できます。
 1. プリンターフォルダーの任意のプリンターアイコンをクリックします。
 2. プリンターフォルダー上部のツールバーに表示された[プリントサーバー プロパティ]をクリックします。

7. 通信ライブラリ

本章の仕様でバージョン1.xxから仕様変更された内容には(★)マークを記載しています。
通信ライブラリを利用する場合、機能設定の一部を指定する設定値にする必要があります。
詳しくは"2.2 機能設定"を参照してください。

7.1 概要

プリンタードライバーにはプリンターを直接制御できる通信ライブラリを開発者向けに提供しています。
通信ライブラリはプリンタードライバーと共にインストールされ、プリンタードライバーを利用して動作します。
通信ライブラリを利用することでアプリケーション開発においてプリンターを直接制御することが可能となり、ポートの種類に依存しない設計が可能となります。
関数の具体的な利用方法については付属するサンプルプログラム (Microsoft Visual C++ version 2005版) も参照してください。

7.2 機能

通信ライブラリは、開発されるアプリケーションに対して次の機能を提供します。

- ◆ プリンターへの任意バイナリデータの送信
- ◆ プリンターからの受信データの取得
- ◆ プリンターのステータスデータの取得
- ◆ プリンターのリセット

7.3 ライブラリファイル

通信ライブラリのファイル名は次の通りです。

SII_IFM_API.DLL

通信ライブラリのファイル保管場所は、Windowsのシステムフォルダーです。

特殊な場合を除き、通信ライブラリが保存されているフォルダーへパスを通す必要はありませんので、フォルダーから移動せずにそのまま使用してください。

通信ライブラリファイルを移動するとプリンタードライバーのバージョンアップの際に通信ライブラリが正しく更新されません。

7.4 関数

通信ライブラリ関数一覧

通信ライブラリに実装された関数の一覧を示します。

関数名	機能概略
OpenSiiPrinterA OpenSiiPrinterW	プリンターオブジェクトを作成し、識別番号を取得します
CloseSiiPrinter	プリンターオブジェクトを破棄し、識別番号を無効にします
SetSiiPrinterData	プリンターへ任意のバイナリデータを出力します
SetSiiPrinterTimeout	SetSiiPrinterData 実行時のタイムアウト時間を設定します
GetSiiPrinterASB	ASBを取得します
SetSiiPrinterCallbackASB	ASBの変化で呼出されるコールバック関数を登録します
GetSiiPrinterCounter	メンテナンスカウンタの情報を取得します
GetSiiPrinterError	エラー履歴を取得します
GetSiiPrinterStatus	プリンターステータス情報を取得します
GetSiiPrinterId	プリンターIDを取得します
GetSiiPrinterFontId	フォントID情報を取得します
GetSiiPrinterDataA GetSiiPrinterDataW	指定の応答データを取得します
SetSiiPrinterReset	プリンターをハードウェアリセットします

引数に文字列を伴う関数名について

一部の関数名の最後にある'W'または'A'の表記は、文字列で設定する引数をMBCS(マルチバイトコード)またはUNICODE(ユニコード)のいずれかを利用するかにより、呼出すべき関数名が異なることを意味します。

例えば**OpenSiiPrinter**関数の場合に呼出すべき関数名は、MBCSの場合、**OpenSiiPrinterA**、UNICODEの場合、**OpenSiiPrinterW**で呼出することになります。

以降の説明においては末尾の'W'または'A'を省略した表記での説明となります。

7.5 通信ライブラリ関数詳細

各関数の詳細について説明します。

全関数共通仕様

戻り値

関数失敗時≠0(エラーコード)、成功時=0を応答します。

解説

- ◆ エラーコードについては、"7.8 関数戻り値(エラーコード)"を参照してください。
- ◆ 通信ライブラリはプリンターの共有機能によるネットワーク接続では利用できません。
- ◆ Windows Platform SDKで定義されるPRINTER_INFO_X構造体のStatusメンバーなどのWindowsが提供する機能によるプリンター情報のサポートは対応しておりません。
- ◆ シリアル通信設定のフロー制御は[ハードウェア]でご利用ください。
設定方法は"5.3 ポート シート"を参照してください。
- ◆ 何れかの手段によりASBを無効にするコマンドを含んだデータを出力した場合、以降のASBが正常に取得できない可能性があります。
- ◆ 通信ライブラリの全ての機能は双方向サポートの機能が有効になっていることが利用条件となります。
設定方法は"5.3 ポート シート"を参照してください。
- ◆ データ出力を途中で中断するなどプリンターが正しくコマンドを読取れなくなった状態からドライバーの機能を利用できるように復帰させるにはプリンターのリセット(**SetSiiPrinterReset**関数呼出)が必要となります。
- ◆ 応答サイズを取得する機能においてプリンターが未接続時には固定サイズの応答コマンドに限り取得できません。

OpenSiiPrinter

プリンターオブジェクトを作成し、識別番号を取得します。

```
DWORD OpenSiiPrinter(
    LPCTSTR pszName,
    LPDWORD pdwSessionId)
```

パラメーター

pszName

プリンターのフレンドリ名を表す、NULLで終わる文字列へのポインタ

pdwSessionId

プリンターの識別番号の変数へのポインタ

解説

- ◆ *pszName* に入力すべきプリンターのフレンドリ名とは、プリンターフォルダーに表示された指定するプリンターの名前を指します。
- ◆ 本関数で取得した識別番号を利用しなくなった場合、必ず**CloseSiiPrinter**関数で無効にする必要があります。
- ◆ プリンターとコンピューターとの接続状態に関係なく本関数は成功します。
- ◆ プリンタードライバーの接続先がサポート対象外のポートになっている場合、本関数は失敗します。
- ◆ 同時にオープンできるオブジェクトは最大8個となります。
- ◆ プリンターが印字可能状態でない場合、本関数の応答に時間がかかる場合があります。

CloseSiiPrinter

プリンターオブジェクトを破棄し、識別番号を無効にします。

```
DWORD CloseSiiPrinter(
    DWORD dwSessionId)
```

パラメーター

dwSessionId

OpenSiiPrinter 関数で取得した識別番号

解説

- ◆ プリンターオブジェクトの識別番号を無効にします。
- ◆ **SetSiiPrinterCallbackASB**関数によるASBの監視を停止します。
- ◆ 本関数以外の呼出中の関数の処理が完了するまで、本関数の応答は戻りません。

SetSiiPrinterData

プリンターにデータを書込みます。

```
DWORD SetSiiPrinterData(
    DWORD dwSessionId,
    LPBYTE pCmd,
    DWORD cbCmd,
    LPDWORD pcWritten)
```

パラメーター

dwSessionId

OpenSiiPrinter 関数で取得した識別番号

pCmd

プリンターデータが格納されている配列へのポインタ

cbCmd

pCmd が指すバッファのバイト単位のサイズ

pcWritten

プリンターに書込まれたデータのバイト数を受取る変数へのポインタ

不要な場合にはNULLを指定します

解説

- ◆ 任意のバイナリデータをプリンターに送信することが可能です。
- ◆ データ送信が完了するか、またはタイムアウトになるまで制御は戻りません。
- ◆ プリンタードライバーのジョブが存在する場合、本関数は失敗します。
- ◆ 1ドキュメント分のデータを複数回の呼出しに分けて出力した場合、プリンタードライバーからの印刷データが割り込む可能性があります。
- ◆ 本関数で処理されるデータ送信は、プリンタードライバーのジョブには含まれません。
- ◆ タイムアウト時間については**SetSiiPrinterTimeout**関数を利用して設定が可能です。
タイムアウト時間についての詳細は
"**SetSiiPrinterTimeout**SetSiiPrinterTimeoutSetSiiPrinterTimeoutSetSiiPrinterTimeout"
を参照してください。
- ◆ 出力されるバイナリデータにASBを無効にするコマンドが含まれる場合、その後のASBを正常に取得できなくなる場合があります。
- ◆ 複数のプロセスから通信ライブラリが利用され、かつ本関数を複数回に分割してデータ送信を行った場合、途中で別プロセスからの意図しない出力が割り込む可能性があります。特にイメージデータなど途中で他のデータの割り込みが許されないコマンドやデータを出力する場合、全てのデータを1度の呼出しにより出力するようにしてください。
- ◆ 実際に出力されたデータサイズが*cbCmd*のサイズに満たない場合、次の動作となります。
 - *pcWritten*にNULLを指定した場合、本関数は失敗します。
 - *pcWritten*にNULL以外を指定した場合、送信済みデータ数を*pcWritten*で示される変数に格納して本関数は成功します。
- ◆ 本関数は**SetSiiPrinterReset**によって中断することが出来ます。(★)

SetSiiPrinterTimeout

SetSiiPrinterData関数で有効なタイムアウト時間を設定します。

DWORD SetSiiPrinterTimeout(

DWORD *dwSessionId*,

DWORD *dwTimeout*)

パラメーター

dwSessionId

OpenSiiPrinter関数で取得した識別番号

dwTimeout

出力時のタイムアウト時間をミリ秒単位で指定した値の変数

解説

- ◆ **SetSiiPrinterData**関数で有効なタイムアウト時間を設定します。
SetSiiPrinterData関数以外のタイムアウト時間には影響しません。
- ◆ 設定可能時間は3～90秒の範囲となります。範囲外の時間を指定した場合には本関数は失敗します。
- ◆ 実際のタイムアウト時間は設定された時間以上になる場合があります。
- ◆ *dwTimeout*で設定した値は、指定し直したまたは、**CloseSiiPrinter**が行われるまで有効です。
- ◆ 本関数によりタイムアウト時間が設定されていない場合、システム(LPTポート)のタイムアウト時間が有効となります。システム(LPTポート)のタイムアウト時間は以下の2通りの手順で設定できます。

プロパティシートでの編集

1. [プロパティ]画面のポートシートにおいていずれかの[プリンタポート(LPTx)]を選択します。
2. [ポートの構成]ボタンをクリックするとタイムアウトの時間が表示されますので、任意のタイムアウト時間に秒単位で入力します。

※ [プロパティ]画面については"5. プロパティ設定"を参照してください。

※ タイムアウト時間の変更の際、プリンタードライバーの出力先(ポート)が変更されない様にご注意ください。

レジストリを直接編集

1. レジストリエディタ(regedit.exe)を利用してのタイムアウト時間の設定を行う場合、
HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥WindowsNT¥CurrentVersion¥Windows
フォルダーにある"TransmissionRetryTimeout"の値を任意のタイムアウト時間に秒単位で入力します。

※ レジストリ編集の操作を誤りますとPCが起動できなくなる恐れがありますので十分にご注意ください。

- ◆ システム(LPTポート)のタイムアウト時間は3秒未満に設定できません。
- ◆ システム(LPTポート)のタイムアウトを有効にするには通信ライブラリの再ロードが必要です。(★)

GetSiiPrinterASB

最新のASBを取得します。

DWORD GetSiiPrinterASB(
 DWORD dwSessionId,
 LPDWORD pdwStatus)

パラメーター

dwSessionId

OpenSiiPrinter関数で取得した識別番号

pdwStatus

ASBを格納する変数へのポインタ

解説

- ◆ 最新のASBを取得します。
- ◆ プリンターとの切断を検出した場合、ASBの値を0として応答します。
- ◆ 応答データはASB形式で取得します。詳しくは"7.6 応答データ形式"を参照してください。
- ◆ ASBの応答内容については製品技術説明書を参照してください。
- ◆ ASBの取得に関する注意事項は"7.7 ASB応答に関する注意事項"を参照してください。
- ◆ **SetSiiPrinterCallbackASB**でコールバック関数を登録している場合には、本関数の呼出しで現在のコールバック関数の登録は無効となります。

SetSiiPrinterCallbackASB

ASBの変化により呼出されるコールバック関数を登録します。

DWORD SetSiiPrinterCallbackASB(
DWORD *dwSessionId*,
INT (*CALLBACK EXPORT *lpfnCallBackStatus*)(*DWORD dwStatus*))

パラメーター

dwSessionId

OpenSiiPrinter関数で取得した識別番号

lpfnCallBackStatus

コールバック関数の関数ポインタ

dwStatus

コールバック関数でASBを受信するための変数

解説

- ◆ ASBの変化を検出すると呼出されるコールバック関数を本関数で登録できます。
- ◆ ASBの取得に関する注意事項は"7.7 ASB応答に関する注意事項"を参照してください。
- ◆ ASBの応答内容については製品技術説明書を参照してください。
- ◆ 本関数によりコールバック関数を登録すると、登録直後に1回だけ現在のASBでコールバック関数を呼出します。
- ◆ コールバック関数で取得できる応答データはASB形式で取得します。
詳しくは"7.6 応答データ形式"を参照してください。
- ◆ プリンターからASBを受信した場合でも、直前に受信したASBと比べ変化がない場合にはコールバック関数は呼出されません。
- ◆ プリンターとの接続状態に変化があった場合にコールバック関数が呼出されます。
- ◆ 再接続を検出した場合、最後に受信しているASBを応答します。
- ◆ ASBの受信とコールバック関数の呼出しタイミングの同期性については、保証されません。
- ◆ 既にコールバック関数を登録した状態で再度本関数を呼出した場合、登録されている関数は無効となり、新しいコールバック関数が登録されます。
- ◆ *lpfnCallBackStatus*にNULLを指定した場合、ASBの監視を中断します。
- ◆ **CloseSiiPrinter**関数、または**GetSiiPrinterASB**関数の呼出しにより登録済みのコールバック関数を無効にして、ASBの監視を中断させることができます。
- ◆ コールバック関数の戻り値は無視されます。
- ◆ 既に登録済みで有効なコールバック関数を再度指定して本関数を呼出した場合も、直後のASB応答は行われます。(★)
- ◆ 登録されたコールバック関数内から通信ライブラリの関数を呼出すことは出来ません。(★)

GetSiiPrinterCounter

メンテナンスカウンタ応答を取得します。

```
DWORD GetSiiPrinterCounter(
    DWORD dwSessionId,
    WORD wIndex,
    LPBYTE pData,
    DWORD cbData,
    LPDWORD pcbNeeded)
```

パラメーター

dwSessionId

OpenSiiPrinter関数で取得した識別番号を指定します

wIndex

取得するメンテナンス番号の変数

pData

応答データを受け取るバイト配列へのポインタ

cbData

pData が指すバッファのバイト単位のサイズ

pcbNeeded

必要なバッファサイズまたは実際に受信したバイト数を受け取る変数へのポインタ

解説

- ◆ 製品技術説明書に記載の[メンテナンスカウンタの送信]コマンドの応答を取得します。
- ◆ 応答データはHEXコード形式で取得します。詳しくは"7.6 応答データ形式"を参照してください。
- ◆ [メンテナンスカウンタの送信]コマンド、及びその応答内容については製品技術説明書を参照してください。
- ◆ *wIndex*には、[メンテナンスカウンタの送信]コマンドのパラメーター「メンテナンス番号(n)」を2バイトで指定してください。(ex. n=14Hの場合、*wIndex*=0014H)
- ◆ *wIndex*に無効な値を設定した場合、本関数はデータ取得に失敗します。
- ◆ 応答データのサイズを取得したい場合、*cbData*に0をセットして本関数を呼出すことにより本関数は **ERROR_INSUFFICIENT_BUFFER**のエラーコードを応答し、応答データサイズが*pcbNeeded*のポインタが示す変数に格納されます。

GetSiiPrinterError

エラー履歴応答を取得します。

DWORD GetSiiPrinterError(

DWORD *dwSessionId*,

BYTE *byIndex*,

LPBYTE *pData*,

DWORD *cbData*,

LPDWORD *pcbNeeded*)

パラメーター

dwSessionId

OpenSiiPrinter関数で取得した識別番号

byIndex

取得する応答の種類を示す値の変数

pData

応答データを受け取るバイト配列へのポインタ

cbData

pData が指すバッファのバイト単位のサイズ

pcbNeeded

必要なバッファサイズまたは実際に受信したバイト数を受け取る変数へのポインタ

解説

- ◆ 製品技術説明書に記載の[エラー履歴応答]コマンドの応答を取得します。
- ◆ *byIndex*には、[エラー履歴応答]コマンドのパラメーター(n)を指定してください。
- ◆ 応答データはHEXコード形式で取得します。詳しくは"7.6 応答データ形式"を参照してください。
- ◆ [エラー履歴応答]コマンド、及びその応答内容については製品技術説明書を参照してください。
- ◆ *byIndex*に無効な値を設定した場合、本関数はデータ取得に失敗します。
- ◆ 応答データのサイズを取得したい場合、*cbData*に0をセットして本関数を呼出すことにより本関数はERROR_INSUFFICIENT_BUFFERのエラーコードを応答し、応答データサイズが*pcbNeeded*のポインタが示す変数に格納されます。

GetSiiPrinterStatus

プリンターのステータス応答を取得します。

```
DWORD GetSiiPrinterStatus(  
    DWORD dwSessionId,  
    BYTE byIndex,  
    LPBYTE pData)
```

パラメーター

dwSessionId

OpenSiiPrinter関数で取得した識別番号

byIndex

取得するステータスの種類を示す値の変数

pData

応答データを受け取るバイト配列へのポインタ

解説

- ◆ 製品技術説明書に記載の[ステータスの送信]コマンドの応答を取得します。
- ◆ 応答データはステータス応答形式で取得します。詳しくは"7.6 応答データ形式"を参照してください。
- ◆ [ステータスの送信]コマンド、及びその応答内容については製品技術説明書を参照してください。
- ◆ *byIndex*には、[ステータスの送信]コマンドのパラメーター(n)を指定してください。
- ◆ *byIndex*に無効な値を設定した場合、本関数はデータ取得に失敗します。

GetSiiPrinterId

プリンターID応答を取得します。

```
DWORD GetSiiPrinterId(
    DWORD dwSessionId,
    BYTE byIndex,
    LPBYTE pData,
    DWORD cbData,
    LPDWORD pcbNeeded)
```

パラメーター

dwSessionId

OpenSiiPrinter関数で取得した識別番号

byIndex

取得するプリンターIDの種類を示す値の変数

pData

応答データを受け取るバイト配列へのポインタ

cbData

*pData*が指すバッファのバイト単位のサイズ

pcbNeeded

必要なバッファサイズまたは実際に受信したバイト数を受け取る変数へのポインタ

解説

- ◆ 製品技術説明書に記載の[プリンタIDの送信]コマンドの応答を取得します。
- ◆ 応答データはASCII文字列またはHEXコード形式で取得します。詳しくは"7.6 応答データ形式"を参照してください。
- ◆ [プリンタIDの送信]コマンド、及びその応答内容については製品技術説明書を参照してください。
- ◆ *byIndex*には、[プリンタIDの送信]コマンドのパラメーター(n)を指定してください。
- ◆ *byIndex*に無効な値を設定した場合、本関数はデータ取得に失敗します。
- ◆ 応答データのサイズを取得したい場合、*cbData*に0をセットして本関数を呼出すことにより、本関数は **ERROR_INSUFFICIENT_BUFFER**のエラーコードを応答し、応答データサイズが*pcbNeeded*のポインタが示す変数に格納されます。

GetSiiPrinterFontId

プリンターフォントID応答を取得します。

DWORD GetSiiPrinterFontId(

DWORD dwSessionId,

BYTE byType,

BYTE byParam,

BYTE byFunc,

LPBYTE pData,

DWORD cbData,

LPDWORD pcbNeeded)

パラメーター

dwSessionId

OpenSiiPrinter関数で取得した識別番号

byType

取得するプリンターフォントIDのフォント種類を示すパラメーターの変数

byParam

取得するプリンターフォントIDの種類を示すパラメーターの変数

byFunc

取得するプリンターフォントIDの機能を示すパラメーターの変数

pData

応答データを受け取るバイト配列へのポインタ

cbData

pData が指すバッファのバイト単位のサイズ

pcbNeeded

必要なバッファサイズまたは実際に受信したバイト数を受け取る変数へのポインタ

解説

- ◆ 製品技術説明書に記載の[半角フォントIDの送信]、または[全角フォントIDの送信]コマンドの応答を取得します。
- ◆ 応答データはASCII文字列またはHEXコード形式で取得します。
詳しくは"7.6 応答データ形式"を参照してください。
- ◆ [半角フォントIDの送信]、または[全角フォントIDの送信]コマンド、及びその応答内容については製品技術説明書を参照してください。
- ◆ *byType*には、取得するフォントIDの種類を示す値を下記表から選び入力してください。

<i>byType</i>の値		フォントIDの種類	
16進数	10進数	全角／半角	フォントサイズ
00H	0	全角	24
01H	1	↑	16
02H	2	半角	24
03H	3	↑	16

- ◆ *byParam*には、半角文字と全角文字の場合で入力する値が異なります。
半角文字の場合
 [半角フォントIDの送信]コマンドの「ページ番号」を示すパラメーター(b)の値を入力してください。
全角文字の場合
 [全角フォントIDの送信]コマンドの「書体」を示すパラメーター(b)の値を入力してください。
- ◆ *byFunc*には、製品技術説明書の[全角フォントIDの送信]または[半角フォントIDの送信]コマンドの説明にある「機能」を示すパラメーター(c)の値を入力してください。

IFM/PTM プリンタードライバー

- ◆ 応答データのサイズを取得したい場合、*cbData*に0をセットして本関数を呼出すことにより本関数はERROR_INSUFFICIENT_BUFFERのエラーコードを応答し、応答データサイズが*pcbNeeded*のポインタが示す変数に格納されます。
- ◆ いずれかのパラメーターに無効な値を設定した場合、本関数はデータ取得に失敗します。

GetSiiPrinterData

プリンターからの応答データを取得します。

```
DWORD GetSiiPrinterData(
    DWORD dwSessionId,
    LPTSTR pValueName,
    LPBYTE pData,
    DWORD cbData,
    LPDWORD pcbNeeded)
```

パラメーター

dwSessionId

OpenSiiPrinter関数で取得した識別番号

pValueName

取得するデータを識別する応答コマンド文字列

pData

応答データを受け取るバイト配列へのポインタ

cbData

pData が指すバッファのバイト単位のサイズ

pcbNeeded

必要なバッファサイズまたは実際に受信したバイト数を受け取る変数へのポインタ

解説

- ◆ *pValueName*に"コマンド文字列一覧"で紹介しているコマンド文字列を指定すると、任意のプリンター応答を取得できます。
- ◆ 応答のデータ形式については*pValueName*の内容により異なります。
各応答形式については"7.6 応答データ形式"を参照してください。
- ◆ 各プリンターコマンド及びその応答内容については製品技術説明書を参照してください。
- ◆ 応答データのサイズを取得したい場合、*cbData*に0をセットして本関数を呼出すことにより本関数は **ERROR_INSUFFICIENT_BUFFER**のエラーコードを応答し、応答データサイズが*pcbNeeded*のポインタが示す変数に格納されます。応答すべきデータが無い場合には0が格納されます。

IFM/PTM プリンタードライバー

- ◆ *pValueName*に指定できるコマンド文字列の一覧を記載します。

コマンド文字列一覧

対応コマンド文字列 (<i>pValueName</i>)	対応プリンターコマンド(応答)
GET_RAW_DATA_ASB_RESP	[ASB]の受信履歴
GET_RAW_DATA_EXEC_RESP	[実行応答]の受信履歴
GET_RAW_DATA_INIT_RESP	[イニシャライズ完了応答]の受信履歴
GET_RAW_DATA_PROG_RESP	[進捗応答]の受信履歴
AUTO_STATUS_BACK	[ASB]
FUNCTION_SET_RESP	機能設定応答
REMAIN_USER_MEMORY_CAP	ユーザ領域の残り容量応答
EXT_RAM_MEMORY_CAP	拡張RAM容量応答
REMAIN_USER_MEMORY_CAP_DEFRAG	ユーザ領域の整理後の残り容量応答
NV_MEMORY_CAP	NVグラフィックスメモリ容量の送信
REMAIN_NV_MEMORY_CAP	NVグラフィックスメモリの残容量の送信
REG_NV_IMAGE_KEYCODE_RESP	定義されているNVグラフィックスのキーコード一覧の送信

- ◆ 受信履歴は、*pValueName*で指定された応答データを、それぞれ最大256バイト分保持します。
- ◆ 受信履歴では、**OpenSiiPrinter**関数の呼出し後に受信した生データ(未変換データ)を記録します。
- ◆ 受信履歴は別プロセスからの操作により応答された受信データも含まれます。
また同一の操作や関数の呼出しにより同等の応答結果を取得できるとは限りません。
- ◆ プリンターを再接続された場合、プリンターに蓄積されていた過去のデータをまとめて受信する場合があります。
- ◆ *pValueName*に"GET_RAW_DATA_ASB_RESP"を指定した場合、ASBの受信履歴を取得します。
尚、前回と同一の内容のASBが受信された場合、その応答は破棄されます。
1回の応答(8バイト)分のASBのデータが全て'0'である場合、未接続を検出したことを意味します。
- ◆ *pValueName*に"GET_RAW_DATA_EXEC_RESP"を指定した場合、[実行応答]の受信履歴を取得します。
[実行応答]については製品技術説明書を参照してください。
- ◆ *pValueName*に"GET_RAW_DATA_INIT_RESP"を指定した場合、[イニシャライズ完了応答]の受信履歴を取得します。
[イニシャライズ完了応答]については製品技術説明書を参照してください。
- ◆ *pValueName*に"GET_RAW_DATA_PROG_RESP"を指定した場合、[進捗応答]の受信履歴を取得します。
[進捗応答]については製品技術説明書を参照してください。
- ◆ "コマンド文字列一覧"で紹介されているコマンド文字列はシステム標準関数(Win32API)である**GetPrinterData**関数では利用できません。
- ◆ 関数が失敗した(関数の戻り値が'0'でない)場合、全ての引数の内容は不定になります。
- ◆ *pValueName*に設定したコマンドによっては、関数が失敗する場合があります。

SetSiiPrinterReset

プリンターをハードウェアリセットします。

**DWORD SetSiiPrinterReset(
 DWORD dwSessionId)**

パラメーター

dwSessionId

OpenSiiPrinter関数で取得した識別番号

解説

- ◆ 通信プロトコルを利用した(プリンターコマンドを利用しない)ハードウェアリセットを行います。
ただし、仮想ポートではハードウェアリセットは対応できません。(★)
- ◆ ハードウェアリセットが成功すると一定の待ち時間が発生します。(★)
- ◆ **SetSiiPrinterData**中に本関数を呼出した場合、**SetSiiPrinterData**を中断できます。(★)
- ※ 機能設定において通信プロトコルによるリセット機能が有効である必要があります。
機能設定については 製品技術説明書を参照してください。
- ※ 本関数を利用する場合、プリンターのリセット処理が実行されるまで、1～2秒おいてからデータ出力などの処理を行ってください。
時間をおかずにデータ出力を行うとデータ抜けの原因となりますのでご注意ください。
また、フラッシュメモリを書き換える処理などにより、プリンターがすぐにリセット処理を実行できない状態の場合は、さらに時間が必要となる場合があります。

7.6 応答データ形式

概要

プリンターからの受信データは、開始コード/終了コードまたは識別コードが付加されております。

通信ライブラリでは一部の機能を除き、この受信データから開始コード/終了コードまたは識別コードを取り除いた形式にして応答します。以下に各関数で応答する受信データの形式について記載します。

各形式の(例)で説明しているプリンター出力とはプリンターからの生データで、関数応答とは通信ライブラリの関数が応答するデータの形式を指します。

応答形式一覧

通信ライブラリ関数	応答データの識別コード	応答形式
GetSiiPrinterASB SetSiiPrinterCallbackASB	CxH / DxH	ASB応答形式
GetSiiPrinterStatus	AxH	ステータス応答形式
GetSiiPrinterCounter GetSiiPrinterError GetSiiPrinterId	2xH ~ 7xH	ASCII文字列応答形式
GetSiiPrinterFontId GetSiiPrinterData ※1	ExH / FxH	HEXコード応答形式
GetSiiPrinterData ※2	8xH ~ DxH	RAWデータ応答形式

※1: pValueNameの値が"GET_RAW_DATA_XXX"以外の場合

※2: pValueNameの値が"GET_RAW_DATA_XXX"の場合

ASB応答形式

識別コード(CxH)で始まるASB応答データ(8バイト)の上位4ビット(識別コード(CxH/DxH))を除いた下位4ビットだけを1バイト目から順にLSBから詰めた形式で32ビット(4バイト)として応答します。

また未接続状態を検出した場合は全ビットを0として応答します。

ASB応答については"7.7 ASB応答に関する注意事項"を参照してください。

(例) プリンター出力: **C1H、D2H、D3H、D4H、D5H、D6H、D7H、D8H** ⇒ 関数応答: **0x87654321**

ステータス応答形式

識別コード(AxH)で始まるステータス応答データ(1バイト)を、識別コードを含めたそのままの形式で応答します。

ASCII文字列応答形式

開始コード(02H)で始まる文字列形式応答データを開始コードのみ除いた形式で応答します。

(例) プリンター出力: **02H、53H、49H、49H、00H** ⇒ 関数応答: **53H、49H、49H、00H**

指定したパラメーターに対応するプリンター情報がない場合、関数応答は**00H**を応答します。

HEXコード形式

開始コード(0EH)で始まるHEXコード形式応答データを開始コード及び終了コード(00H)/識別コード(ExH / FxH)を除くデータ形式で応答します。

(例) プリンター出力: **0EH、E2H、F1H、E4H、F3H、E6H、F5H、00H** ⇒ 関数応答: **12H、34H、56H**

指定したパラメーターに対応するプリンター情報がない場合、関数の戻り値はエラーコード30となります。

但し、0レングス応答の場合、関数は成功となります。(★)

RAWデータ形式

受信データを開始コード/終了コードまたは識別コードを含めた生データを受信した順に応答します。

7.7 ASB 応答に関する注意事項

- ※ ASBは**GetSiiPrinterASB**関数、**SetSiiPrinterCallbackASB**関数及び**GetSiiPrinterData**関数で取得することが可能です。
詳しくは各関数の説明を参照してください。
- ※ 未接続状態の検出は識別子を含め全ビットを0としたASBで表現されます。
- ※ ASBの機能を無効にするコマンドを**SetSiiPrinterData**関数より出力した場合、正常にASBを取得できなくなる恐れがあります。
- ※ プリンターと通信不能、または通信エラー発生の場合でも[未接続状態]になる場合があります。
- ※ シリアル接続のフロー制御設定がハードウェアの場合、DSR端子をMARK側に固定する利用方法はサポート対象外となります。
- ※ シリアルケーブルは最低限TxD,RxD, DTR及びDSRの端子がコンピューター側とプリンター側で接続されていないと接続状態が検出できないため、正常に動作できなくなる恐れがあります。
- ※ 切断中のASB変化は取得できない場合があります。
- ※ 接続状態に関わらず、ASBを取得できていない場合、ASB応答を伴う関数は失敗します。
- ※ 再接続を検出した場合、その時点で最後に受信しているASBを応答します。
よって必ずしも接続後のステータスを保証するものではありません。
- ※ ASBの内容については、製品技術説明書の[自動ステータス送信の有効・無効]コマンドを参照してください。

7.8 関数戻り値(エラーコード)

- ◆ 関数の戻り値として成功時はERROR_SUCCESS(=0)、失敗時はエラーコード(≠0)となります。
エラーコードは、Windowsのシステムエラーコードに準拠したコードとなります。
- ◆ WindowsのシステムエラーコードにつきましてはMicrosoftのドキュメント(SDK System Error Codes等)を参照してください。
- ◆ Windowsのシステムエラーコードを利用しているためエラー内容が曖昧な表現場合があります。
その補足として主なエラー名(エラーコード)と想定される原因を示します。

主なWindowsシステムエラーコードと想定される原因

エラー名 (エラーコード)	想定されるエラー原因
ERROR_INVALID_HANDLE(6)	入力した識別番号が不正である
ERROR_READ_FAULT(30) ERROR_TIMEOUT(1460)	<ul style="list-style-type: none"> ◆ 指定したパラメーターのプリンター情報が存在しない ◆ 不正なパラメーターで応答が取得できない ◆ プリンターが処理中、または異常状態で応答が取得できない ◆ プリンターのバッファに他のデータが存在している
ERROR_BUSY(170)	プリンターがビジー(印刷中)状態である
ERROR_DEVICE_NOT_CONNECTED(1167)	<ul style="list-style-type: none"> ◆ ケーブルが未接続である ◆ プリンターの電源が入っていない
ERROR_UNKNOWN_PORT(1796)	FILEなどサポート外のポート名が設定されている
ERROR_UNKNOWN_PRINTER_DRIVER(1797)	サポート外のプリンターのフレンドリ名を指定している
ERROR_INVALID_PRINTER_STATE(1906)	双方向通信サポートが[無効]になっている
ERROR_DEVICE_NOT_AVAILABLE(4319)	通信不能になっている
ERROR_WRITE_FAULT(29)	データの書き込みができない
ERROR_PRINTER_HAS_JOBS_QUEUED(3009)	プリンターに印刷待ちジョブが存在する
ERROR_ACTIVE_CONNECTIONS(2402)	別のスレッド、または別プロセスでDLLを利用中である
ERROR_CONNECTION_COUNT_LIMIT(1238)	オブジェクト数が上限を超過した
ERROR_BAD_ENVIRONMENT(10) ERROR_ACCESS_DENIED(5) ERROR_INVALID_ACCESS(12) ERROR_UNEXP_NET_ERR(59) ERROR_DEV_NOT_EXIST(55)	双方向モジュールが起動できない。または、双方向モジュールでエラーが発生している

7.9 サンプルプログラム

通信ライブラリを利用したサンプルプログラム(Microsoft Visual C++ version 2005版)を用意しています。

- ※ サンプルプログラムの動作保証また、サンプルプログラム及びMicrosoft Visual C++ version 2005版の技術サポートは致しておりません。
- ※ サンプルプログラムは予告なしに変更される場合があります。

8. 免責

本ソフトウェアの開発ではトラブルが発生しないよう十分留意していますが、万一、本製品によりお客様に損害を及ぼした場合でも、一切の責を負いかねますのでご了承ください。