

豊富な受託実績を開発者向けソフトウェアに

オークのソフトウェア部品

OUK

<http://www.ouk.jp/>

オークのソフトウェア部品

豊富な受託実績を開発者向けソフトウェアに

弊社では長年にわたり受託開発に携わって参りました。
その経験・蓄積を活かす形で、自社商品としてソフトウェア部品を開発致しました。
すべての商品において、そのベースになっているのは、受託開発で豊富な採用実績を持つソースコードです。
パソコン向けデバイスドライバから組み込み用途を想定した商品まで、豊富なラインナップで開発者の皆様を支援致します。
「オークのソフトウェア部品」をベースとした受託開発から、開発ノウハウの提供、お客様の手による開発のサポートまで、多様な開発形態に対応可能です。

→ カスタマイズも可能

デバイスドライバ向け商品につきましては、弊社が設定する標準機能をお客様のハードウェアで実現するための作業費用が料金に含まれます。

機能追加についても承っておりますので、お気軽にご相談下さい。

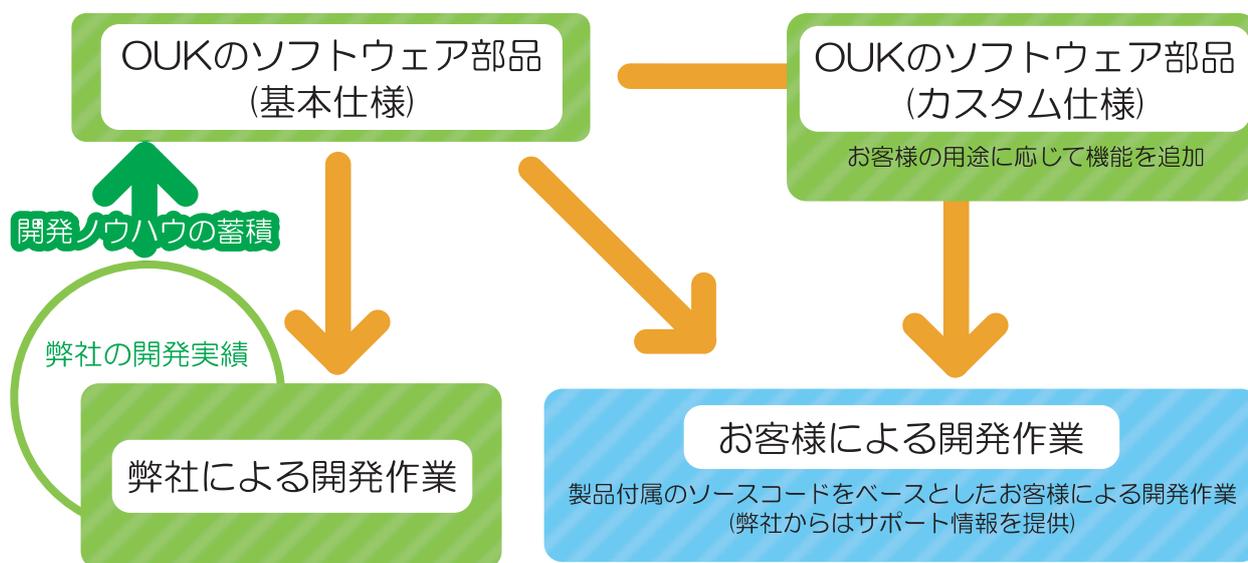
組み込み向け商品はeCos、Linux(POSIX)、 μ ITRON、VxWorksで動作実績がございます。

この他の動作プラットフォームについてはお気軽にお問い合わせ下さい。

→ 低コスト・短期での開発を実現

採用実績のあるソースコードをベースとしておりますので、基本部分は安定した性能を実現しております。
従いまして、ゼロから開発するのと比較して、低コスト・短期間で開発することが可能となります。

→ お客様の方針に合わせた柔軟な開発作業



お客様の開発方針に合わせ、3つの選択肢から開発方法をお選びいただけます。

主な商品ラインナップ

オークのソフトウェア部品には、すべて、ソースコードが付属します。お客様ご自身の手による開発も可能です。

また、コピーライセンスを必要としない契約形態（無制限コピーライセンス契約）を基本としております。

パソコン用各種USBデバイスドライバ

汎用クライアントドライバ

ストレージデバイスドライバ

パソコン用スキャナドライバ

TWAIN 準拠ドライバ(Windows,Macintosh)

Microsoft WIA 準拠ドライバ(Windows ME, Windows XP)

マルチメディア・ストリーミング関連商品

ビデオキャプチャドライバ

音声コーデックドライバ(ACM ドライバコア：新規コーデック開発に)

Windowsパソコン用プリンタドライバ

モノクロプリンタドライバ(本体)

カラー対応プリンタドライバ(本体)

ESC/POS 制御コード対応プリンタドライバ(本体)

プリントプロセッサ（印刷イメージ加工）

ステータスマニタ

マルチプラットフォーム対応ライブラリ

OXlib GUI 開発ツール

OXmail メール送信ライブラリ、OEM 開発用メールソフト

組み込み向け

ストレージミドルウェア(FAT ファイルシステム、SCSI3/ SBP2 プロトコルライブラリ)

軽量ネットワークサーバ(Web サーバ, FTPサーバ, Telnet サーバ他)

時刻補正NTP クライアント

機器監視(SNMP)：機器組込用SNMP エージェント+ MIB構築ユーティリティ

組み込み向けGUI 開発ツールOXlib

OUK

株式会社オーク
〒540-0029

大阪市中央区本町橋2-23 第7松屋ビル8F
TEL.06 (6945) 4704/FAX.06 (6945) 4705

WEB.<http://www.ouk.jp/>

技術本部営業課 滝井<takii@ouk.jp>

御用命は弊社販売代理店まで

OXlib

プラットフォーム共通GUI開発ツール

→ シンプルな設計による軽量GUI・本格的な日本語対応

OXlib は標準で日本語対応したシンプルなX-Window対応GUI 開発ツールです。日本語対応機能は文字コード変換から縦書き表示にいたるまで、充実した機能を装備しております。

開発に必要な基本的なAPI 数はわずか8つ、短期間で習得可能です。

また、X-Window 対応アプリケーション開発において通常必要な、各種ツールキットやXlibに関する知識は一切必要ありません。

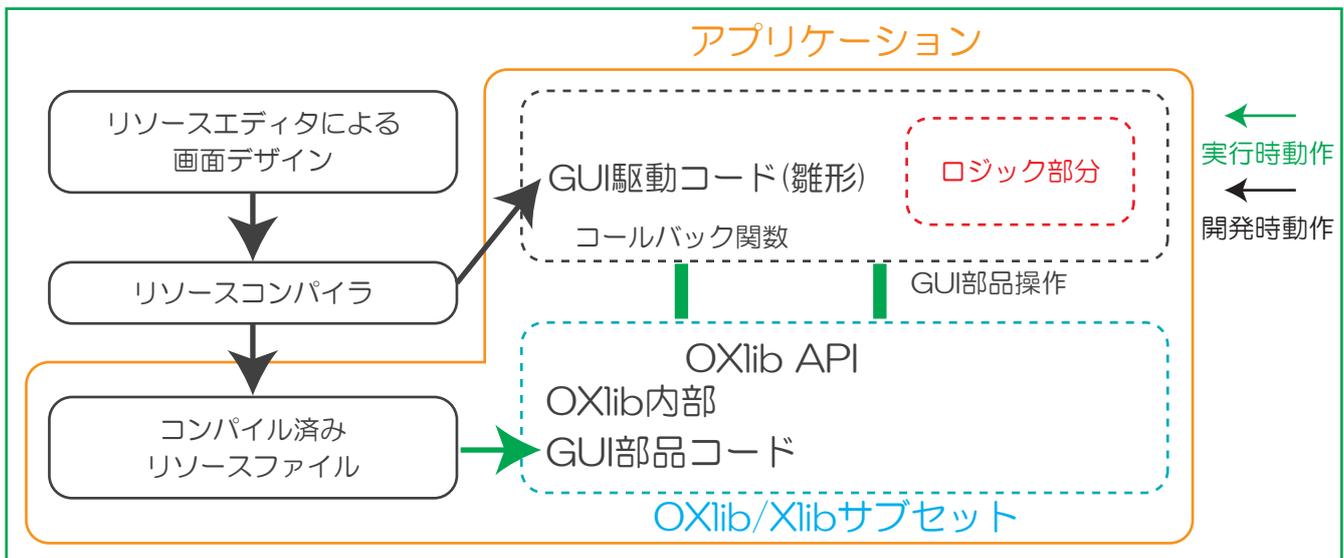
充実したサポート体制でお客様の疑問・要望にお応えします。

また、カスタム用途の部品作成などのご相談にも応じております。



→ 画面設計とロジック実装の分離 ~専用GUIデザインツール~

OXlib 最大の特徴は、アプリケーション固有のロジック部分を、画面デザインとは分離して開発できる点にあります。専用の画面設計ツール(リソースエディタ)が用意されており、まず、このリソースエディタを使用してGUIをデザインします。



● 簡単な操作

リソースエディタでの操作は、予め用意された部品を選んで画面上に配置するだけの簡単なものです。また、新たな部品を追加することも可能です。

● その場で確認

レイアウトが決定した後は、ボタン1つで、実際の画面を表示して確認することができます。ごく簡単な部品が使用されている画面であれば、プログラムを記述することなく、実際に動作させることも可能です。

GUI 駆動コードとアプリケーション固有のロジック部分を連携動作させるために、各種コールバック関数が用意されています。マウスやキーボードからの操作は、このコールバック関数を通じてアプリケーションに通知されます。GUI 部品が保持している情報の取得や動作設定用関数を併用することで、OXlib をコントロールすることができます。

→ 動作環境

OXlibはX-Windowが動作する各種プラットフォームをサポートしており、アプリケーションの開発ならびに実行が可能です。

Linux、BSD をルーツに持つOS、各社商用UNIX[®]で動作・採用実績がございます。

また、Windows[®]向けのX サーバを使用することで、Microsoft Windows[®]での画面表示も可能となります。

※弊社開発、Windows[®]、DirectX対応のリーズナブルなXサーバをご検討ください。

また、姉妹品として、OXlib のC++版であるOXlib++も御用意いたしております。

オークのソフトウェア部品には、全ての製品に、製品ソースコードが付属します。

組み込み用途に対応するため、自社開発の組み込み用X サーバとともに動作する組み込み版OXlib もご用意致しております。組み込み用X サーバは一般的なXサーバが要求する資源の数分の1で動作します。

なお、組み込みOSで使用する場合には、ハードウェア依存部分として、グラフィック・デバイス、ポインティング・デバイス対応のコードを移植する必要があります。

→ APIとGUI部品一覧

組み込み用OXlib では次の8つのAPIを使用することで、全ての基本操作が実行可能です。

OXinitial() OXlib の初期化

OXmakeBox() リソースからGUI 部品データを作成

OXdestroyBox() GUI 部品データの破棄

Oxlib では、原則として、リソースエディタが作成した静的な設計情報に従ってGUI 画面を再現します。

OXmapBox() GUI 部品を表示

OXunmapBox() GUI 部品を消去

OXsetResource() GUI 部品のリソース値の変更

OXgetResource() GUI 部品のリソース値の読み出し

各GUI 部品は大きさや位置などの情報を保持しています。

こうした情報(リソース値)を取得・変更することができます。

OXmainLoop() OXlib のメインループ

各種初期化処理を行った後に使用して、イベント駆動による動作に移行します。

GUI 部品として、以下のような部品を作成することができます。

既存の部品の組み合わせ、付属のソースコードの拡張により新たな部品を作成することも可能です。

テキスト(縦書き、横書き、点滅)

エディット・ボックス

リスト・ボックス

チェック・ボックス

ボタン(ビットマップ、テキスト)

ラジオ・ボタン(複数候補から1つを選択)

スクロール・バー(スクロール・フレーム、矢印ボタン、バー)

各種フレーム(ドロー用、イメージ用、可変サイズ、グループ化、etc)

メニュー

メッセージ・ウィンドウ

ファイル選択ダイアログ

各種グラフ描画(円グラフ、棒グラフ、折れ線グラフ、レーダーチャート、グラフ用目盛り)

※ UNIX[®]、Motif[®]は米国The OpenGroupが管理する米国、及び各国における登録商標です。※ Windows[®]は米国Microsoft Corp.の米国、及び各国における登録商標です。※ eCos[®]は米国RedHat Inc.の米国、及び各国における登録商標です。※ VxWorks[®]は米国WindRiverSystems, Inc.の米国、及び各国における登録商標です。※ Linux はLinus Torvalds 氏の米国、及び各国における登録商標です。

Windows®対応 X-Windowサーバ

Windowsで動作するXサーバソフトウェア

Xサーバソフトを利用することにより、TCP/IP ネットワークを介してLinux、BSD、商用UNIX 環境にアクセス、Windows®からの操作が可能となります。

→ 全画面モードと単一ウィンドウモードをサポート

キー入力によりWindows®デスクトップと全画面表示のXサーバとを切り替える全画面モードと、ひとつのウィンドウに複数のX クライアントを表示する単一ウィンドウモードをサポートしています。

DirectX 技術の採用により高速な表示が可能となりました。

※個別ウィンドウモードにも順次対応予定

弊社開発のX プロトコルベースGUI 構築ライブラリOXlib と併用してご利用いただけます。高速な全画面モードは、Windows®をクライアントマシンとする、OXlib を使用したプログラムの動作デモや専用システムのGUI 等に最適です。

OXlibはWindows®からLinux、BSD、各社商用UNIX、組み込み用途までを開発ターゲットとして、共通の操作性、クロス開発を実現するGUI ライブラリ&ツールです。

また、日本語の利用を前提に設計・開発されておりますので、安心して御利用いただけます。

→ 使い慣れた日本語入力環境を使用可能

Windows®側には「IME サーバ」機能を実装しており、Windows®で動作する日本語入力システムが使用可能です。X サーバ側ではX-Window 環境で使われている日本語入力プロトコルである、kinput2、XIM、Ximp の各プロトコルをサポートしています。これらのプロトコルをサポートするX-Window 対応アプリケーション(X クライアント)に対して、Windows®の日本語入力システムとの橋渡し機能を提供します。

→ その他

| | |
|---------------|---|
| 動作環境 | Windows®98/ME、Windows®NT4.0、Windows®2000、Windows®XP |
| 複数起動 | DirectX(DirectX 7)および、TCP/IP ネットワークが動作する環境 設定ファイルにより複数の設定に対応。複数同時起動も可能です。 ネットワークドライブからの起動も可能です。 |
| 表示 | マルチディスプレイ対応。カラーはWindows®の設定/制限に準じます。 |
| キーボード・マウスの動作 | Windows®の操作設定に準じます。 |
| フォント | フォントサーバまたは固定フォルダのフォントを使用。 |
| リモートホストへのアクセス | XDMCP によるログイン、REXEC によるコマンド起動。 |
| セキュリティ | ホスト名によるアクセス制御 |

※仕様は開発途中のものであり、予告なしに変更される可能性があります。

※ UNIXは米国The Open Group が管理する米国、及び各国における登録商標です。

※ Windows®、WindowsNT®、は米国Microsoft Corp.の米国、及び各国における登録商標です。

OXlib

プラットフォーム共通GUI開発ツール

→ シンプルな設計による軽量GUI

OXlib は標準で日本語対応したシンプルなX-Window対応GUI 開発ツールです。日本語対応機能は文字コード変換から縦書き表示にいたるまで、充実した機能を装備しております。

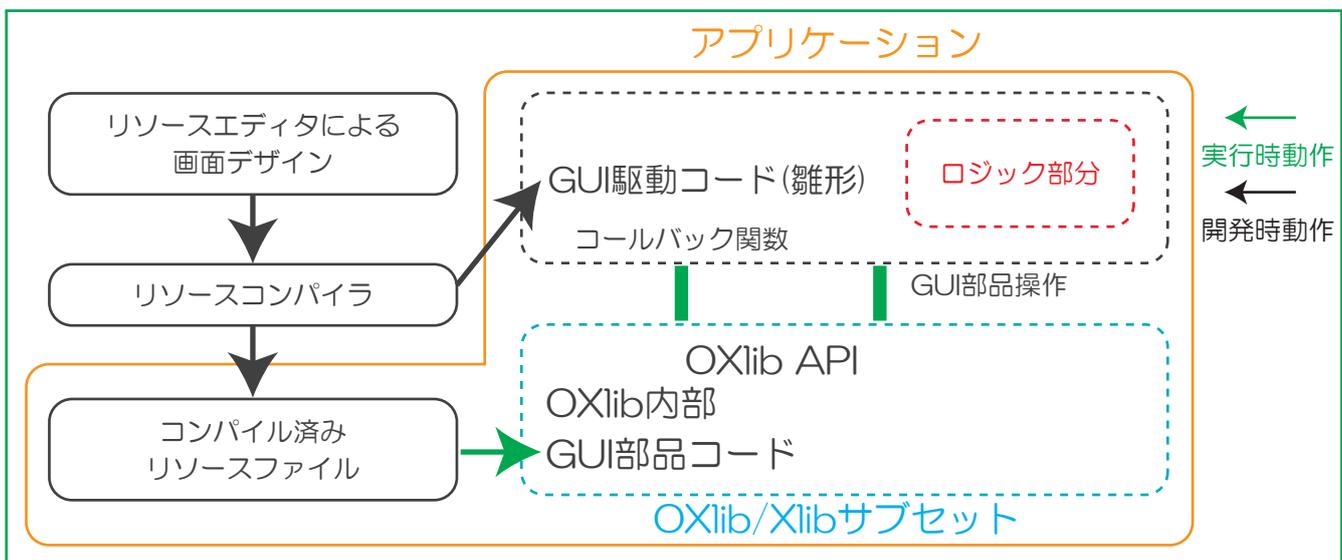
開発に必要な基本的なAPI 数はわずか8 つ、短期間で習得可能です。また、X-Window 対応アプリケーション開発において通常必要な、各種ツールキットやXlibに関する知識は一切必要ありません。

充実したサポート体制でお客様の疑問・要望にお応えします。また、カスタム用途の部品作成などのご相談にも応じております。

『組み込み用OXlib』はOXlib の特徴をそのまま継承した組み込み向けGUI 開発ツールで、専用X サーバを含めても、作成したアプリケーションは非常にコンパクトなものになります。

→ 画面設計とロジック実装の分離 ～専用GUIデザインツール～

OXlib 最大の特徴は、アプリケーション固有のロジック部分を、画面デザインとは分離して開発できる点にあります。専用の画面設計ツール(リソースエディタ)が用意されており、まず、このリソースエディタを使用してGUIをデザインします。この作業はPC、UNIX搭載EWS で行っていただけます。



● 簡単な操作

リソースエディタでの操作は、予め用意された部品を選んで画面上に配置するだけの簡単なものです。また、新たな部品を追加することも可能です。

● その場で確認

レイアウトが決定した後は、ボタン1 つで、実際の画面を表示して確認することができます。ごく簡単な部品が使用されている画面であれば、プログラムを記述することなく、実際に動作させることも可能です。

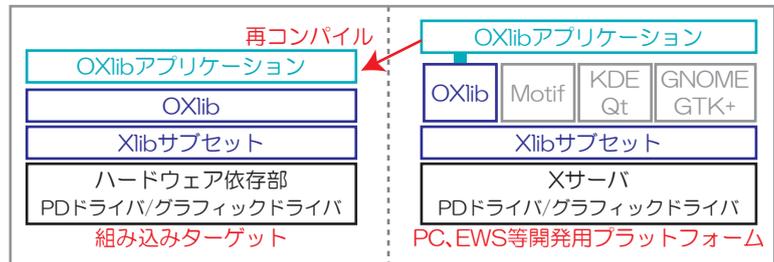
GUI 駆動コードとアプリケーション固有のロジック部分を連携動作させるために、各種コールバック関数が用意されています。マウスやキーボードからの操作は、このコールバック関数を通じてアプリケーションに通知されます。GUI 部品が保持している情報の取得や動作設定用関数を併用することで、OXlib をコントロールすることができます。

→ 動作環境

GUI 設計情報(リソース)、及びGUI 部分のソースコードは、各種プラットフォームで共通のものとなっており、わずかな修正で移植可能です。サブセットのXlib ライブラリが使用されるという違いはありますが、組み込み向けのアプリケーションでも、同様の開発手法をとることができます。

本製品を組み込む場合、ハードウェア依存部分として、グラフィック・デバイス、ポインティング・デバイス(PD)対応のコードを移植する必要があります。

オークのソフトウェア部品には、全ての製品に、製品ソースコードが付属します。



→ APIとGUI部品一覧

組み込み用OXlib では次の8つのAPIを使用することで、全ての基本操作が実行可能です。

OXinitial() OXlib の初期化

OXmakeBox() リソースからGUI 部品データを作成

OXdestroyBox() GUI 部品データの破棄

Oxlib では、原則として、リソースエディタが作成した静的な設計情報に従ってGUI 画面を再現します。

OXmapBox() GUI 部品を表示

OXunmapBox() GUI 部品を消去

OXsetResource() GUI 部品のリソース値の変更

OXgetResource() GUI 部品のリソース値の読み出し

各GUI 部品は大きさや位置などの情報を保持しています。

こうした情報(リソース値)を取得・変更することができます。

OXmainLoop() OXlib のメインループ

各種初期化処理を行った後に使用して、イベント駆動による動作に移行します。

GUI 部品として、以下のような部品を作成することができます。

付属のソースコードを拡張して、新たな部品を作成することも可能です。

テキスト(縦書き、横書き、点滅)

エディット・ボックス

リスト・ボックス

チェック・ボックス

ボタン(ビットマップ、テキスト)

ラジオ・ボタン(複数候補から1つを選択)

スクロール・バー(スクロール・フレーム、矢印ボタン、バー)

各種フレーム(ドロー用、イメージ用、可変サイズ、グループ化、etc)

メニュー

メッセージ・ウィンドウ

ファイル選択ダイアログ

各種グラフ描画(円グラフ、棒グラフ、折れ線グラフ、レーダーチャート、グラフ用目盛り)

※ UNIX®、Motif®は米国The OpenGroupが管理する米国、及び各国における登録商標です。※ Windows®は米国Microsoft Corp.の米国、及び各国における登録商標です。※ eCos®は米国RedHat Inc.の米国、及び各国における登録商標です。※ VxWorks®は米国WindRiverSystems,Inc.の米国、及び各国における登録商標です。※ Linux®はLinus Torvalds 氏の米国、及び各国における登録商標です。

アプリケーション/メールクライアント

メールソフト用ソフトウェア部品

OXmail

サポートOS Windows® 95/98 Windows® Me WindowsNT® 4.0 Windows® 2000 Windows® XP
Windows® CE2.11/3.0 UNIX X-Window

→ メールソフトの中核ソフトウェア部品

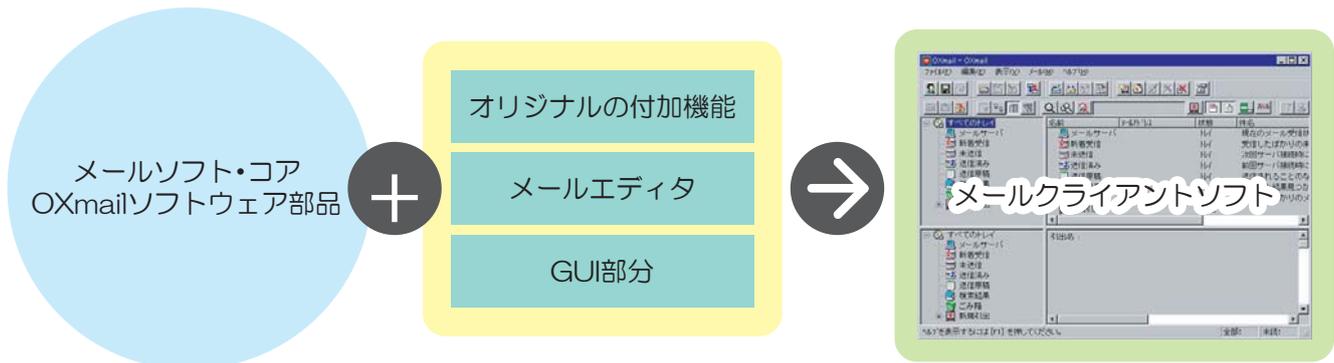
インターネットが普及した現在、メールは標準的なコミュニケーション手段として広く定着するようになりました。また、多くのメール関連ソフトウェアが登場、メール自体の機能も進化を続けています。

それゆえ、メールソフトでは高い互換性、安定性を備えることが必須条件となっております。

オークのソフトウェア部品は、様々なメールソフト、メール・サーバー、プロバイダとの接続がチェックされ、市販のメールソフトに利用されている、実績のあるコードです。

ソフトウェアの「顔」であるオリジナルのユーザーインターフェイス部分、エディタを実装していただくことで、メールソフトが完成します。

組み込みライセンスはフリー。ソースコードが付属しますから、機能追加も可能です。



→ 主な機能

主要なメールソフトに搭載されている機能は、既の実装されています。

詳細については、裏面を参照下さい。

オークのオリジナル機能

● リンク (ショートカット) 機能

振り分け先フォルダにはメールの本体を、「新着受信」フォルダには振り分けたメールのショートカットを残すことで、どちらのフォルダからもメールの内容を参照することができます。

「振り分け機能は便利、でも、振り分けてしまうと新着メールの確認が困難」といった状況を解消します。また、メールの検索結果の保存にも利用可能です。

● 機能集約、強化した「フォルダ」機能

振り分け設定、アドレス帳、テンプレート機能など、一般的なメールソフトでは独立している様々な機能を、「フォルダ」に集約、連携強化しています。

● 自然なメール振り分け設定の追及

多くのメールソフトがサポートしている、条件式(特定の単語を含むなど)による振り分けはもちろんのこと、実際の利用シーンの中から生まれたアイデア満載です。

→条件式による振り分け、キーワードによる振り分け、送信グループによる振り分け

● メールデータ同期、Windowsアドレス帳(WAB)、Windowsネットワーク共有に対応

→ 詳細情報

弊社Webサイト(<http://www.ouk.jp/product/oxmail/>)では、評価用に作成されたメールソフトをご用意しております。ソフトウェア部品の全機能が使用可能ですから、お気軽にお試し下さい。また、弊社開発のデータ暗号化コンポーネントであるODLock との連携も可能です。

搭載されている基本機能

- ・メールの送受信機能(対応プロトコルSMTP, POP3)
 - ※メールの送受信のみに機能を限定したサブセット版もご用意致しております。
- ・自由に変更可能なキー割り当て
- ・階層化フォルダをサポートしたメールデータファイル
 - フォルダ作成、削除、メール一括削除、メールインポート
- ・3ペイン構成のウィンドウ用クラス
 - Windowsの標準的なユーザーインターフェイスでもあるエクスプローラーと同様の操作性を実現しております。
- ・メール検索

オーク独自の着眼点

メールソフトは通常、メールを仕分け整理するために「フォルダ」機能を持っています。オークはこの「フォルダ」に注目し、「フォルダ」を中心に機能を強化しました。例えば、ビジネスであれば、プロジェクトや顧客ごとにフォルダを作成することが多いはず。そうであれば、プロジェクト名などのキーワードがタイトルに含まれているメールや、プロジェクトに属するメンバーからのメールを振り分けるのに、メールソフトのユーザーが細かな条件式を設定する必要はありません。また、フォルダを起点とすれば、送信する相手も自然と限定されてくるという発想のもと、アドレス帳機能(送信グループ)も統合しています。目的のアドレスに効率良くアクセスできます。

利用用途

● メールソフトの新規開発に

ノウハウの蓄積が必要なメール送受信部分は既に完成していますから、**短期間での開発を実現、本来の開発目的・付加機能**に注力していただけます。

OXmailの強力なフォルダ機能を活用すれば、通常のメールソフトから一歩進んで、メール配信サービスのメッセージ管理プログラムなどを開発することも可能です。

● メールを利用した様々なプログラムに

メールはデータ交換手段としても利用されてきました。送信するデータのフォーマットに対する制限が少なく、定期的な動作レポートや文面として埋め込んだコマンド文字列による制御など、様々な応用が可能です。

※ Windows®、WindowsNT®は米国Microsoft Corp.の米国、及び各国における登録商標です。



株式会社オーク
〒540-0029
大阪市中央区本町橋2-23 第7松屋ビル8F
TEL.06 (6945) 4704/FAX.06 (6945) 4705
WEB.<http://www.ouk.jp/>
技術本部営業課 滝井<takii@ouk.jp>

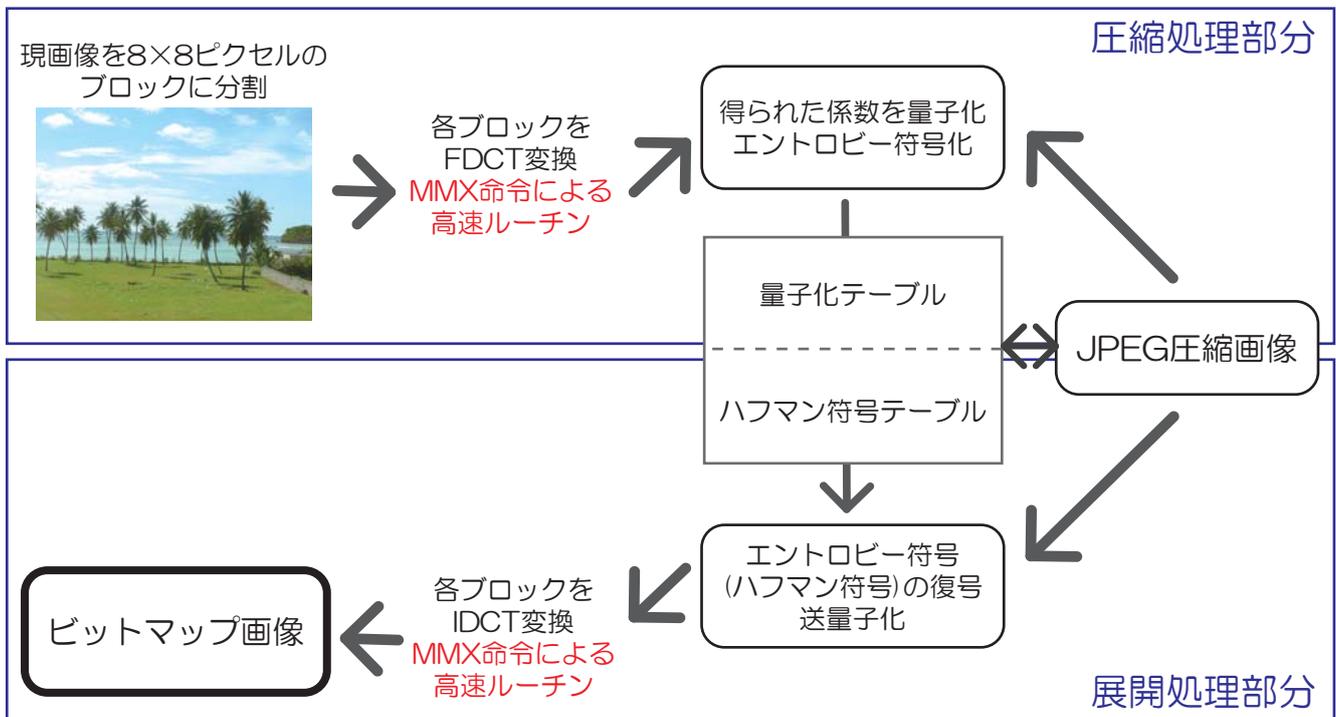
御用命は弊社販売代理店まで

JPEG応用動画形式向け JPEG高速展開ライブラリ

JPEG - 高画質・高圧縮 画像フォーマット

JPEG フォーマットは、人間の視覚的特性を利用して、データサイズの削減と高画質を実現している画像フォーマットです。Webページにおいて、多く使用されているフォーマットの1つです。また、JPEG 圧縮はDV(Motion JPEG)を採用したデジタルビデオカメラや、QuickTime でサポートされている動画記録フォーマットにも応用され、動画撮影対応のデジタルカメラで録画データの保存形式として採用されています。

難点として、圧縮/展開に必要な離散コサイン変換(DCT)の計算量が非常に多いことが挙げられます。



→ Intel MMX™ 命令による高速化

本ライブラリでは、JPEG フォーマットで保存された画像をビットマップ画像へ高速変換、表示を可能とします。動画再生にも対応できる処理性能を実現するために開発されました。

高速化の第一段階として、通常、離散コサイン変換には浮動小数点演算を使用しますが、これを整数演算で近似演算しています。

さらに、Intel製CPU に搭載されているマルチメディア用拡張命令、MMX を使用することで高速化を実現しています。

一般的な実装と比較して10% - 20%程度高速となります。(パソコンの総合的な処理性能に依ります。)

MMX 命令では、単一の命令で複数の整数値データに対して同時に処理をすることが可能であり、画像処理などを得意としています。現在、市場で入手可能なWindows®向けパソコンに搭載されているCPUでは、MMX 命令がサポートされています。

→ パフォーマンス

ターゲット・ハードウェアへの移植作業

本ライブラリを、DirectShow®のオーバーレイ機能を利用するMotion JPEG 表示プログラムに組み込んだ場合、Pentium®III 600MHz、画面サイズ640x480 の環境で30フレーム/秒(日本におけるテレビ放送などと同等のフレームレート)を達成しています。

なお、Motion JPEG 規格は静止画像フォーマットとしてのJPEGを、動画に応用した動画フォーマットです。

→ 製品詳細

対応フォーマット

本ライブラリ製品では、JPEG 規格で規定された基本機能(展開処理部分)をサポートしています。一部アプリケーション固有のJPEG フォーマットには対応しておりません。

順次圧縮(baseline、extended-sequential)されたJPEG 画像の展開、プログレッシブJPEG の展開に対応しています。

符号化処理についてはハフマン符号を使用しており、拡張仕様にあたる算術符号には対応しておりません。

以下の形式にはついては、2002年3月現在、対応しておりません。

縮小画像から差分情報を使用して段階的に元のサイズまで復元する階層(HM)方式、医療用などモノクロ画像を対象に非可逆圧縮(画像劣化なしでの圧縮)を可能としたJPEG-LS(Loss Less)、従来の変換アルゴリズム(DCT)とは異なるWavelet 変換を使用するJPEG2000 フォーマットを扱うことはできません。

細部の仕様や最新の対応状況については、お問い合わせ下さい。

入力形式

メモリ上に展開されたJFIF 形式(JPEG FileInterchange Format :一般的なJPEG 画像の保存形式)のデータを、ライブラリ関数の引数として指定。

出力形式

圧縮された画像イメージを展開し、ビットマップとして出力します。

展開処理完了をコールバック関数により通知。

動作プラットフォーム

本ライブラリは、Intel マルチメディア拡張(MMX)命令をサポートするCPUを搭載するWindows®プラットフォームでご利用いただけます。

なお、オークのソフトウェア部品には、全て、製品ソースコードが付属します。

※ Intel®、Pentium®、MMX®は米国Intel Corp.の米国、及び各国における登録商標、または商標です。

※ Windows®、DirectShow®は米国MicrosoftCorp.の米国、及び各国における登録商標です。

※ QuickTime®、Macintosh®は米国Apple ComputerInc.の米国、及び各国における登録商標です。

キーボード・ドライバ

サポートOS Windows® 95/98/Me WindowsNT® 4.0 Windows® 2000/XP/Vista Macintosh®

→ キーボード・ポート・ドライバ

本デバイスドライバはキーボード接続ポートと通信して、キー入力情報を取得するポート・ドライバとして動作します。

各OS で用意されたキーボード・ドライバは、ポート・ドライバからキー入力情報(キー・スキャンコード)を受け取り、最終的にキー入力イベントとしてアプリケーションに通知されます。

シリアルポートなど、一般的にキーボードの接続に使用されているPS/2やUSB とは、異なるインターフェイスへの対応も可能です。特殊なキーボードを使用する場合、サイズなどの制約からPC 本体に搭載されるインターフェイスが限定されている場合等に御利用いただけます。

ポート・ドライバとしての標準動作の他に、次のような追加機能が用意されております。

→ 拡張可能なキーボード・ドライバとして

本ドライバはデバイスI/O コントロールをサポートしており、プログラムからデバイスドライバの挙動を制御することが可能です。

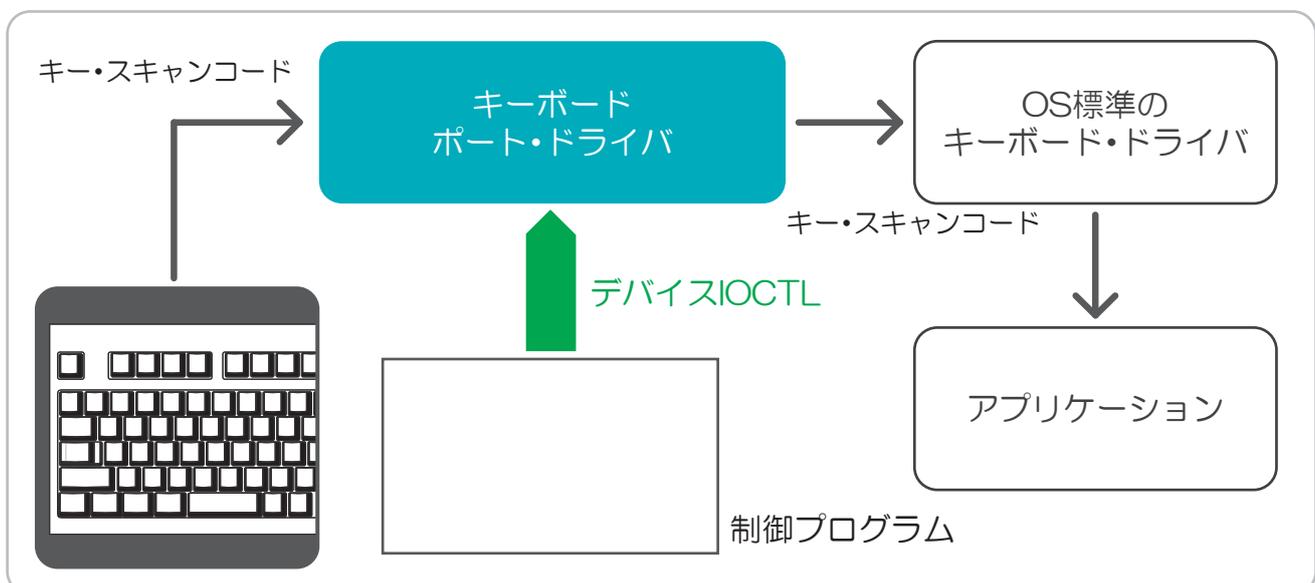
標準で用意されているのは、キー入力(スキャンコード送出)をエミュレーション機能です。

プログラムからデバイスI/O コントロールを発行することにより、ドライバは指定されたスキャンコードを、通常のキーボードからのキー入力と同じようにキーボード・ドライバに送ります。

→ 利用実績

プログラマブル・テンキー用のデバイスドライバ

POS 端末のメンテナンス用シリアル接続キーボードに対応したデバイスドライバ



→ 製品詳細

基本性能

接続インターフェイスを介したキーボードとの通信。特殊なものにも対応可能。
デバイスI/Oコントロールによるキー入力イベントの発生(スキャンコード送出)。

プラットフォーム別の実装状況

Windows® 95/98/Me

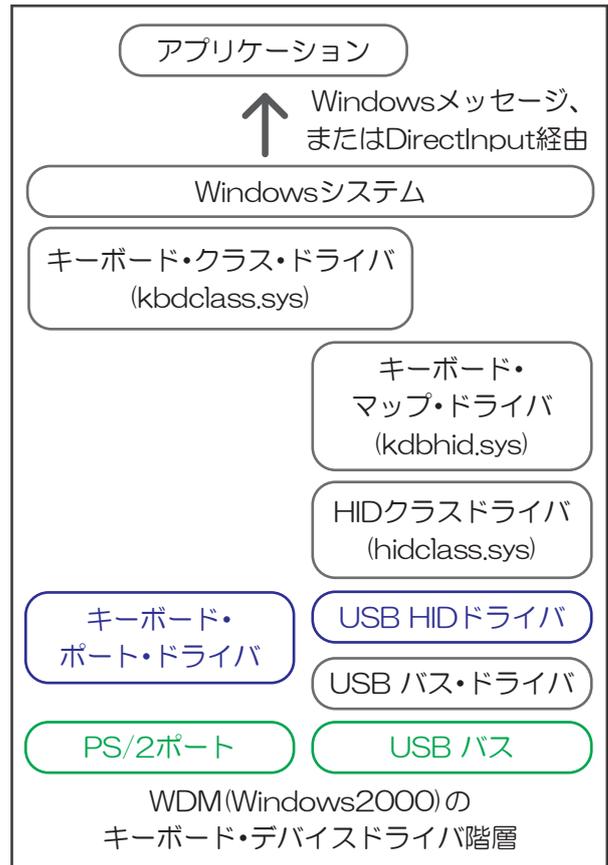
PS/2ポート用のキーボード・ドライバは、VxD(Virtual Device Driver)として実装されます。
ポート・ドライバを実装し、VKD(VirtualKeyboardDriver)に対してキー・スキャンコード送出します。
Windows 98やWindows MillenniumEditionなどWDMをサポートするプラットフォームではWDMに対応したドライバを、USBをサポートするプラットフォームではUSBに対応したドライバを選択いただくことも可能です。

WindowsNT®, Windows® 2000/XP/Vista

WindowsNT,、Windows 2000、Windows XP 向けのキーボード・ドライバはWDMのドライバ階層構造のうち、ポート・ドライバに相当するドライバとなります。
キーボード・クラス・ドライバの下位(ハードウェア寄り)に位置します。
HID(Human Input Device)準拠タイプ、または、従来の非HIDタイプ(PS/2等の場合)の実装から選択していただけます。

Macintosh®

機能拡張文書(INITタイプのファイル)として実装されます。
接続インターフェイスはUSBとなります。



接続インターフェイス

PS/2、またはUSB接続のキーボードに対応します。
シリアル等のインターフェイスにも対応可能です。
ただし、接続に使用するハードウェアに、システムが対応している必要があります。

オークのソフトウェア部品には、すべて、製品ソースコードが付属します。
部品化されているコードは、いずれも、実際にデバイスドライバとして使用実績のあるコードを使用しております。
短期間で信頼性のあるソフトウェアを開発することができます。

※ Windows®, WindowsNT®, は米国Microsoft Corp.の米国、及び各国における登録商標です。
※ Apple®, Macintosh®は米国Apple Computer Inc.の米国、及び各国における登録商標です。

マウス・ドライバ 動作設定ソフトウェア

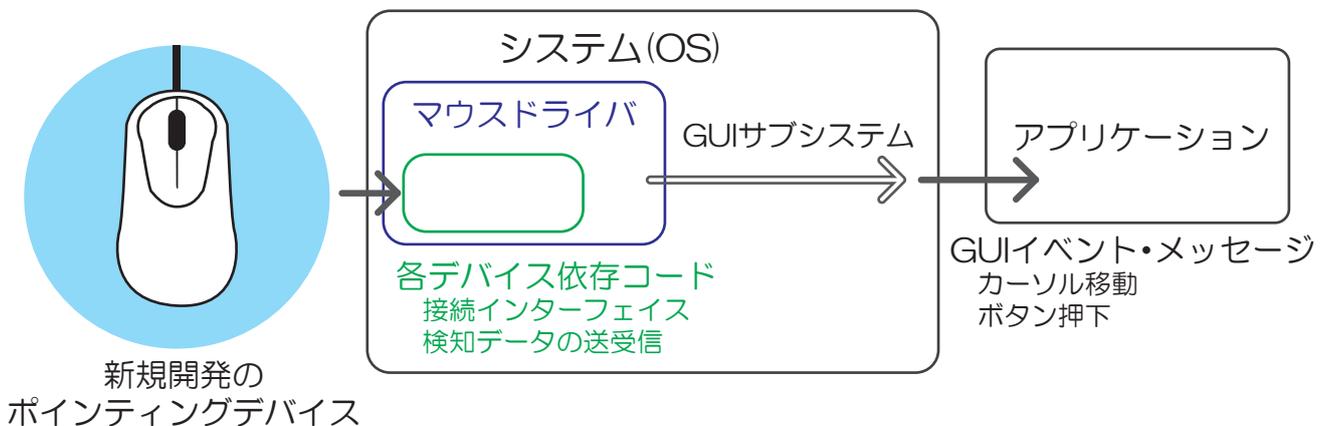
サポートOS Windows® 95/98/Me WindowsNT® 4.0 Windows® 2000/XP/Vista Macintosh®

新たなポインティング・デバイス開発に ～雛型マウスドライバを提供～

本ソフトウェア部品は、マウスをはじめとする相対座標を扱うポインティング・デバイス開発を想定した、ベーシックな構成のマウスドライバです。デバイスドライバ開発において、雛型として御利用いただけます。

マウスドライバとしての基本機能を有する本製品に対して、接続インターフェイスを選択、対象デバイスとの通信部分を追加することによりデバイスドライバが完成します。システムやアプリケーションからは、一般的なマウス互換デバイスとして認識されるようになります。

多様なOS、接続ハードウェア・インターフェイスをサポートしており、対象デバイスを制御／情報を取得することが可能です。



→ コントロールパネル用ファイルも提供

マウスのボタンに特殊な機能(スクロールやズーム、アプリケーションの起動等)を割り当てる場合など、設定画面を用意する必要があります。

Windows では各種動作設定をコントロールパネルから行います。通常、デバイスに応じて設定プログラム(.CPL ファイル)を用意して、コントロールパネルから利用できるようにする必要があります。

Macintosh(MacOS 8/9)では、コントロールパネル・フォルダ内のcdev タイプのファイルが使用されます。各対応OS 提供する標準的な設定項目に加えて、ターゲットのポインティング・デバイスに合わせた項目を追加していくことになります。

なお、追加する動作・設定項目に関して、弊社でコーディングする場合には別途見積もりが必要となります。

基本性能

デバイスドライバ

各対応OS でマウスドライバとして要求される基本機能

標準では2 ボタンマウスをエミュレーションする実装が組み込まれています。

3 つ以上のボタン等多機能デバイス用のドライバとして使用する場合には、追加実装が必要となります。

動作設定用コントロールパネル関連ファイル

標準で用意されているのは、各対応OS に付属するドライバの動作設定と同じ物になります。

ソースコード付属

オークのソフトウェア部品には、すべて、製品ソースコードが付属します。

部品化されているコードは、いずれも、実際にデバイスドライバとして使用実績のあるコードを使用しております。

短期間で、信頼性あるソフトウェア開発が実現できます。

対応OS

Windows®95/98/Meでは仮想デバイスドライバ(VxD)として、WindowsNT®/Windows® 2000/XP/VistaではWDM デバイスドライバとして、Macintosh®(MacOS 8/9)では機能拡張文書(INITタイプのファイル)として、実装されています。

対応ハードウェアインターフェイス

接続インターフェイスとしては、PS/2, シリアル, USB から選択していただけます。

ただし、動作対象となっているシステム(OS)で、各インターフェイスがサポートされている必要があります。

利用実績

スクロールボタン付きマウスドライバと、その動作設定ユーティリティ
画像センサで検知したジェスチャーによるポインティング・デバイス(マウス) 他

開発の流れ (機能追加を弊社に依頼される場合)

最初に、ターゲットとなるOS、接続インターフェイスを選択、デバイスの仕様を提示していただきます。
本マウスドライバをベースに機能を追加して、完成したデバイスドライバ、ならびにソースコードをお渡しすることになります。

※ Windows®, WindowsNT®, は米国Microsoft Corp.の米国、及び各国における登録商標です。
※ Apple®, Macintosh®は米国Apple Computer Inc.の米国、及び各国における登録商標です。

マウス・ドライバ 動作設定ソフトウェア

サポートOS Windows® 95/98/Me WindowsNT® 4.0 Windows® 2000/XP/Vista Macintosh®

多彩なアプリケーションで高い操作互換性を実現 ～標準ドライバに対するフィルタドライバを提供～

オペレーティングシステム(OS)で用意された標準ドライバを使用する場合、アプリケーションの構造等の関係から、スクロール機能などマウスデバイスが持つ機能を100%活かさない場合がございます。

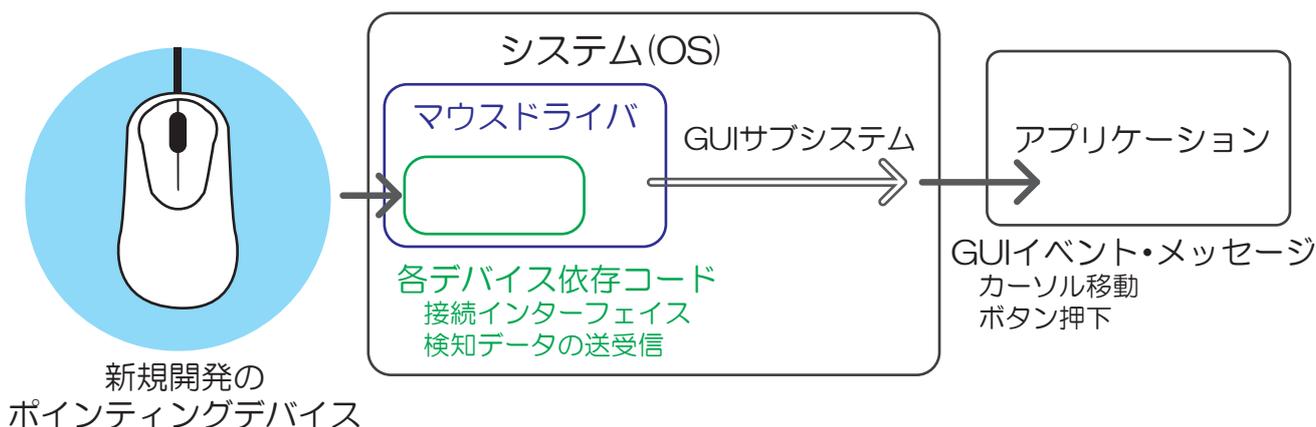
長年の開発実績を持つ弊社のマウス用フィルタドライバを採用いただければ、標準ドライバと様々なアプリケーションを組み合わせた場合でも、エンドユーザに対して共通の操作性を提供することが可能となります。

国内で開発されたものも含めて、多種多様なアプリケーションをサポートしております。

新たなポインティング・デバイス開発に ～雛型マウスドライバと機能設定プログラムを提供～

マウスをはじめとする相対座標を扱うポインティング・デバイス開発を想定した、ベーシックな構成のマウスドライバもご用意いたしております。

デバイスドライバ開発において、雛型として御利用いただけます。



マウスドライバとしての基本機能を有する本製品に対して、接続インターフェイスを選択、対象デバイスとの通信部分を追加することによりデバイスドライバが完成します。システムやアプリケーションからは、一般的なマウス互換デバイスとして認識されるようになります。

多様なOS, 接続ハードウェア・インターフェイスで動作実績を持ち、対象デバイスを制御/情報を取得することが可能です。

基本性能

デバイスドライバ

各対応OS でマウスドライバとして要求される基本機能
 フィルタドライバ:システム(OS)付属の標準マウスドライバに機能付加するドライバ
 のいずれかをご提供。

動作設定用コントロールパネル関連ファイル

マウスのボタンに特殊な機能(スクロールやズーム、アプリケーションの起動等)を割り当てる場合など、
 設定画面を用意する必要があります。
 なお、追加する動作・設定項目に関して、弊社でコーディングする場合には別途見積もりが必要となりま
 す。

ソースコード付属

オークのソフトウェア部品には、すべて、製品ソースコードが付属します。
 部品化されているコードは、いずれも、実際にデバイスドライバとして使用実績のあるコードを使用し
 ています。
 短期間で、信頼性あるソフトウェア開発が実現できます。

対応OS

Windows[®]95/98/Meでは仮想デバイスドライバ(VxD)として、WindowsNT[®]/Windows[®]2000/XP/Vista
 ではWDM デバイスドライバとして、Macintosh[®](MacOS 8/9)では機能拡張文書(INITタイプのファイル)と
 して、実装されています。

対応ハードウェアインターフェイス

接続インターフェイスとしては、PS/2, シリアル, USB から選択していただけます。
 ただし、動作対象となっているシステム(OS)で、各インターフェイスがサポートされている必要があります。

利用実績

スクロールボタン付きマウスドライバと、その動作設定ユーティリティ
 画像センサで検知したジェスチャーによるポインティング・デバイス(マウス) 他

開発の流れ (機能追加を弊社に依頼される場合)

最初に、ターゲットとなるOS、接続インターフェイスを選択、デバイスの仕様を提示していただきます。
 本マウスドライバをベースに機能を追加して、完成したデバイスドライバ、ならびにソースコードをお渡し
 することになります。

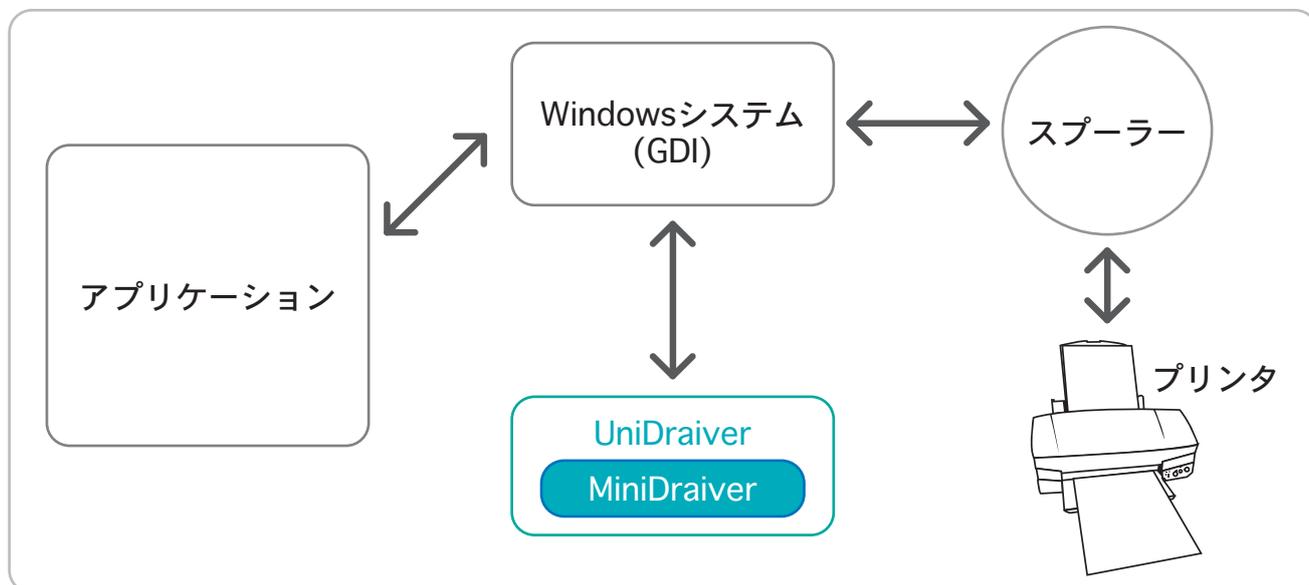
※ Windows[®], WindowsNT[®], は米国Microsoft Corp.の米国、及び各国における登録商標です。
 ※ Apple[®], Macintosh[®]は米国Apple Computer Inc.の米国、及び各国における登録商標です。

UniTool版プリンタドライバ

サポートOS Windows® 3.1 Windows® 95/98/Me WindowsNT® 4.0 Windows® 2000/XP/Vista

UniTool 版プリンタドライバ(Mini Driver)は低コスト、短納期を実現したドライバです(ソースコード提供)

WindowsO がOS 標準で用意している、一般的なプリンタで動作するように設計されたドライバ(UniDriver)と連携動作します。UniDriver に対して、機種ごとの異なる制御コマンドや印字能力などの情報を提供します。プリンタ動作設定用のダイアログは標準で用意されたものを使用しますが、お客様オリジナルのものをデザインいただくことも可能です。



→ 印刷機能

プリンタドライバで実装される各種機能は以下の通りです。DOS/Vマシン(およびその互換機)を対象としたもので、UniDriver の機能に準じます。詳しくは裏面の表を参照してください。

1. 解像度選択
2. 用紙の向き(縦/横)
3. 給紙トレイ選択
4. 両面印刷モードの指定(なし/上下綴じ/左右綴じ)
5. コピー枚数の指定
6. 用紙サイズ選択
7. ディザリングの指定
8. 出力トレイの指定
9. 印刷品質の指定

各プリンタに最適化されたものではないため、画質や印刷速度の面で制限が出る場合がございます。プリンタの能力を最大限活かすことを優先するならば、各プリンタ独自のフル規格ドライバを選択いただくこととなります。

UniTool版ドライバの能力/限界の詳細については、お問い合わせ下さい。

→ 製品詳細

製品内容

製品CDROMには、
 ドライバのソースコード一式
 プリンタドライバのインストールに使用するINF ファイル
 UniTool の設定に関するドキュメント
 が含まれます。

印刷機能

| | |
|---------------|---|
| 解像度選択 | 上限600DPI まで ※ Windows3.1では300DPI まで |
| 用紙の向き設定 | 縦・ポートレート/横・ランドスケープ |
| 給紙トレイ選択 | 自動/トレイ1～ 5/手差し |
| 両面印刷モードの指定 | なし/上下綴じ/左右綴じ |
| コピー枚数の指定 | 最大999枚まで |
| 印刷用紙の選択 | Executive/Letter/Legal/Ledger/A3/A4/A5/ 封筒Monarch/Com-10/封筒DL/封筒C5/B5/ JIS B4/JIS B5/カスタム設定 |
| ディザリングの指定 | ディザなしNone/Coarse/Fine/Line Art/Error Diffusion ※ Windows3.1はError Diffusionをサポートしません。 |
| 出力トレイ(Bin)の指定 | Default Bin/Bin1～Bin10 ※ Windows3.1 では未サポート |
| 印刷品質の指定 | Plain/Transparency (OHP)/Envelope/ Thick Paper/Letter-Head ※ Windows3.1 では未サポート |

開発環境と製品ソースコード受け渡しまでの流れ

本ソフトウェア部品を利用する場合には、別途開発環境が必要となります。

Microsoft WindowsO SDK/DDK(各種OSに対応したもの)
 DDK をサポートする開発環境(Visual C++O)

お客様から制御コマンド、データ形式などをご提示いただき、弊社において、まず標準ダイアログ仕様で実装します。

カスタム設計のダイアログを使用する場合には、標準仕様のドライバが動作することを確認した後の実装することになります。

OS 付属のアプリケーションを使用して動作検証を行い、その後納品となります。

現在市販のアプリケーションによる動作検証も別途承っております。

詳しくは弊社営業までご確認ください。

納期については、比較的コマンド数の少ないもので、開発に0.5人月程度、デバッグ/検証作業に1 人月、合計1.5 人月が目安となります。

仕様によっては、これより長い時間をいただく場合もございます。

※ Windows®、WindowsNT®、VisualC++®は米国Microsoft Corp.の米国、及び各国における登録商標です。



株式会社オーク
 〒540-0029
 大阪市中央区本町橋2-23 第7松屋ビル8 F
 TEL.06 (6945) 4704/FAX.06 (6945) 4705
 WEB.http://www.ouk.jp/
 技術本部営業課 滝井<takii@ouk.jp>

御用命は弊社販売代理店まで

TWAIN ドライバ

サポートOS Windows®3.1 Windows® 95/98/Me WindowsNT® 4.0 Windows® 2000/XP/Vista Macintosh®

→ 基本動作は実装済み

TWAIN規格書で規定されたドライバの基本動作は、既に実装されています。接続インターフェイスを使用したデータ転送、スキャナ制御(コマンド)などデバイス依存部分を追加するだけで、TWAINドライバを完成することができます。

→ オリジナルのダイアログ

スキャナ/デジタルカメラの能力に応じて、お客様オリジナルのダイアログ画面をデザインしていただけます。

→ 実際に動作するサンプルコードが付属

製品には、製品のソースコード、及び、シンプルなダイアログとSCSI接続のESC/I制御スキャナに対応したサンプル・ドライバのソースコードが付属します。

● TWAINドライバとは

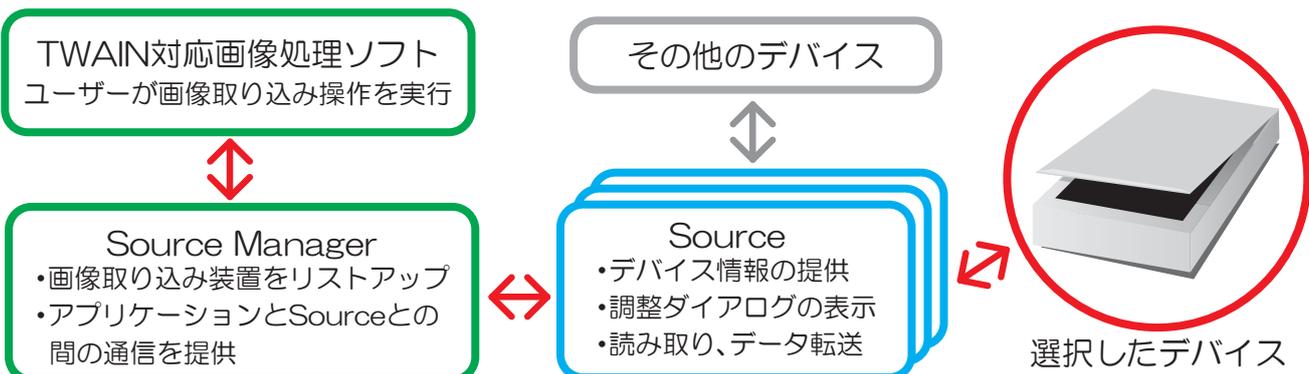
TWAINドライバはスキャナ・ハードウェアと共に製品に同梱されるものです。個々のハードウェアの違い(機能や制御コマンド)を吸収するためのダイナミックリンク・ライブラリで、異なるメーカーのスキャナであっても、TWAINドライバに対応した画像処理ソフトからは同じように操作することができます。

現在ではスキャナだけでなく、デジタルカメラなどの幅広い画像入力装置でも利用されています。

TWAIN規格ではソフトウェアを3つに分割、ハードウェアを含めて4つ階層を規定しています。

TWAIN規格に対応した市販の画像処理ソフト、TWAIN Working Groupが配布しているSource Manager(DLL形式)、個別のハードウェアに対応したSourceドライバ(拡張子.DSのDLL形式)です。

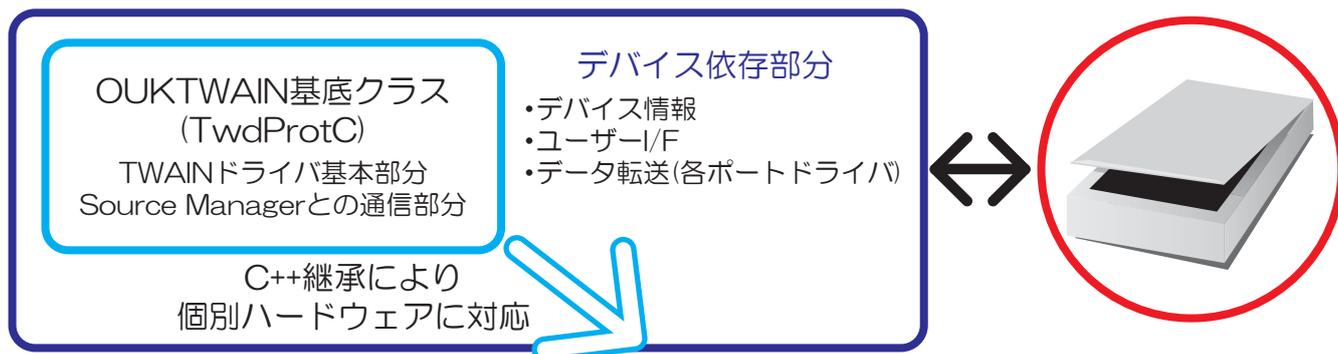
オークTWAINドライバキットはSourceドライバを開発するために設計されました。



→ オークTWINドライバの特徴

TWAIN規格で規定されているAPIなどデバイス非依存部分は既の実装されており、
 スキャナ制御部分(初期化、読み取り開始など)
 スキャナの接続方式(USB、パラレル、SCSI)に応じたデータ転送部分
 ユーザー操作設定画面
 を追加して完成となります。

[Source]TWAINドライバ(派生クラス)



共通部分を実装した基底クラスから、C++の継承機能を使用して、お客様の仕様にカスタマイズしたSourceドライバを作成していただくことになります。

接続に使用するインターフェイスによっては、専用のデバイスドライバが別途必要になる場合がございます(USB、IEEE1394等を使用する場合)。

製品にはサンプルコードとして、シンプルな構成の設定画面とセイコーエプソン社製スキャナ(SCSI接続)の制御コードを使用するドライバが付属しています。サンプルコードでは、SCSIをプログラムから制御するためのインターフェイスであるASPIを使用しています。

また、オークのソフトウェア部品には製品のソースコードが付属します。

→ 製品詳細

Sourceドライバキット対応OSは、Windows® 3.1(16bit DLL)、Windows®95、Windows®98、WindowsNT®、Windows® 2000、Windows®XP、Macintosh®(共有ライブラリ)となります。

それぞれのOSに対応した開発環境が別途必要になります。

Source Managerについては、Windows®2000ではOSに標準添付され、その他のOSではTWAIN Working Groupが開発したものがWebサイト(<http://www.twain.org/>)で配布されています。32bit用としてWAIN_32.DLLが、16bit環境向けにTWAIN.DLLと補助プログラムTWUNKER_16.EXE、TWUNKER_32.EXEが用意されています。

TWAIN規格で定義されているデータ転送のうち、Native転送に加えて、大きなイメージにも対応可能なファイル転送モードをサポートしております。データの形式はWindows DIB形式(デフォルト)を使用し、バッファ転送モードでのJPEGデータ圧縮には対応しておりません。

※最新の規格対応状況、仕様追加に関してお気軽にお問い合わせください。

※ Windows®、WindowsNT®、は米国Microsoft Corp.の米国、及び各国における登録商標です。
 ※ Apple®、Macintosh®は米国Apple Computer Inc.の米国、及び各国における登録商標です。

PAPI互換ライブラリ

サポートOS Windows® 98 Windows® 2000/XP/Vista 組み込み向け

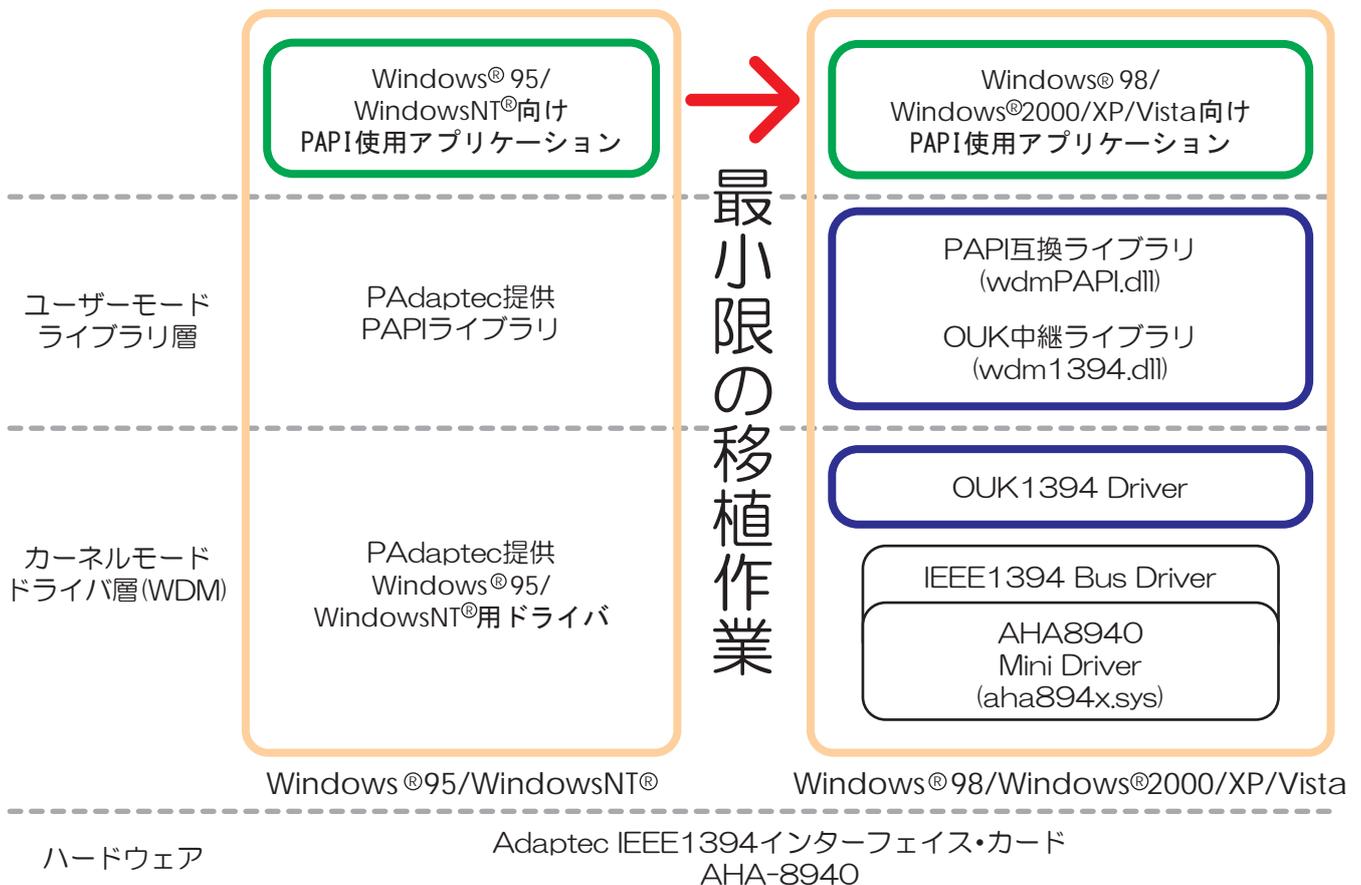
PAPIライブラリ

PAPI(Adaptec 1394 Application Programming Interface)ライブラリはAdaptec社がIEEE1394用に策定したAPIで、同社のインターフェイス・カードAHA-8940等に同梱されています。オリジナルのPAPIライブラリはWindows95/WindowsNTで動作しますが、OS標準でIEEE1394向けAPIが提供されたことにより、Windows98/Windows2000ではサポートされていません。

→ PAPI使用アプリケーションの移植を支援

オークが開発したPAPI互換ライブラリは、Windows98/Windows2000で動作する独自開発のWDM版ドライバ上に同等の機能を構築したものです。

PAPI互換ライブラリはWindows95/WindowsNT上で開発されたソフトウェアの移植作業を支援します。ソースコードを大幅に変更することなく、Windows98/Windows2000が標準で提供しているIEEE1394サポート機能による動作を実現します。また、IEEE1394バス・ドライバのみに依存しますので、AHA-8940以外のハードウェアでも動作するようになります。



中間に位置する中継ライブラリ(wdm1394.dll)と中継ドライバ(OUK1394.sys)は、オークのIEEE1394関連製品で使用されているものです。移植性を確保するとともに、アプリケーションが必要とする情報や機能をカーネルモードから提供することで性能向上を実現しています。

→ 主な関数

PAPI の関数仕様は、SCSI 用制御インターフェイスであるASPIと類似したものとなっています。
弊社のPAPI 互換ライブラリには、Adaptec社製PAPI ライブラリと同様のインターフェイスを用意されております。詳しいAPI 仕様についてはAdaptec 社製品付属のドキュメントを参照下さい。

- GetPAPISupportInfo 1394 下位ドライバの初期化
- SendPAPICommand 下位ドライバに対する処理要求(コマンド送信)
- BusConfig 下位ドライバに対する状態変更、各種設定要求

→ 制限事項

一部機能については、完全に互換性が保たれているわけではありません。
PAPI オリジナル・ライブラリでは、AHA-8940 専用ドライバを使用して、ハードウェアを直接制御することが可能であるのに対して、互換ライブラリではIEEE1394 バス・ドライバの機能を利用します。
IEEE1394 バス・ドライバには、ハードウェア依存部分をミニポート・ドライバとして分離させているという特徴があります。
従いまして、ドライバの構造や機能が制限されたり、ハードウェアから得られる情報が限定されることになります。

→ SendPAPICommand関数での未実装機能

- P_ABORT_1394_CMD P_EXEC_1394_CMD で要求したI/O コマンドの取り消し
- P_GET_CAP_VERSIONS ホストアダプタの情報を取得
搭載チップ・バージョン、プロダクト・バージョン等の情報を取得します。
- P_GET_HA_INFO の一部機能
 - Isoch_Mgr_ID アイソクロナスリソースマネージャーノード(IRM)のノードID を取得
 - Gap_Count 非同期通信におけるパケットの間隔
 - pPower_Map 電源利用状況
 - HA_Identifier ホストアダプタの識別文字列
バス構成/ハードウェア情報のうち、上記4つの情報が取得できません。
- P_NOTIFY_ON_ACCESS(領域アクセスの通知)の一部機能
アクセス元のノード、及びアクセスされたメモリ領域を特定できません。

→ BusConfig関数での未実装機能

- P_LOCAL_HA_READ、P_LOCAL_HA_WRITE
アドレス直接指定による、ホストアダプタに対する読み込み/書き込み
- P_GET_SPEED_BTWN_NODES
ノード間通信速度の取得
- P_MAP_PHYSICAL_1394_ADDR
- P_UNMAP_PHYSICAL_1394_ADDR
物理アドレスをIEEE1394用のメモリ領域として割り当て/割り当て解除
- P_MAP_1394_ADDRESS の一部機能
IEEE1394 の通信用バッファとして、複数の領域を割り当てることができません。
アプリケーションのメモリ領域を設定する場合は、連続する単一の領域を使用してください。

※ AHA®は米国Adaptec Inc. の米国、または各国における登録商標です。
※ Windows®は米国Microsoft Corp.の米国、及び各国における登録商標です。

Serial Bus Protocol-2 (SBP2) / SCSI3 プロトコルライブラリ

サポートOS Windows® 98/Me Windows® 2000/XP/Vista 組み込み向け

SCSI後継としてのIEEE1394

IEEE1394 はデジタル・ビデオカメラを始めとした家電製品を相互に接続、制御することを視野に入れた高速シリアル・インターフェイスです。

IEEE1394 規格の上に位置づけられるSBP2(Serial Bus Protocol-2)を利用することで、従来SCSI規格で動作していた周辺機器を、IEEE1394 規格に移行させることができます。

本ライブラリはIEEE1394プロトコル・スタック上に実装したSBP2、およびSCSI3 ライブラリです。

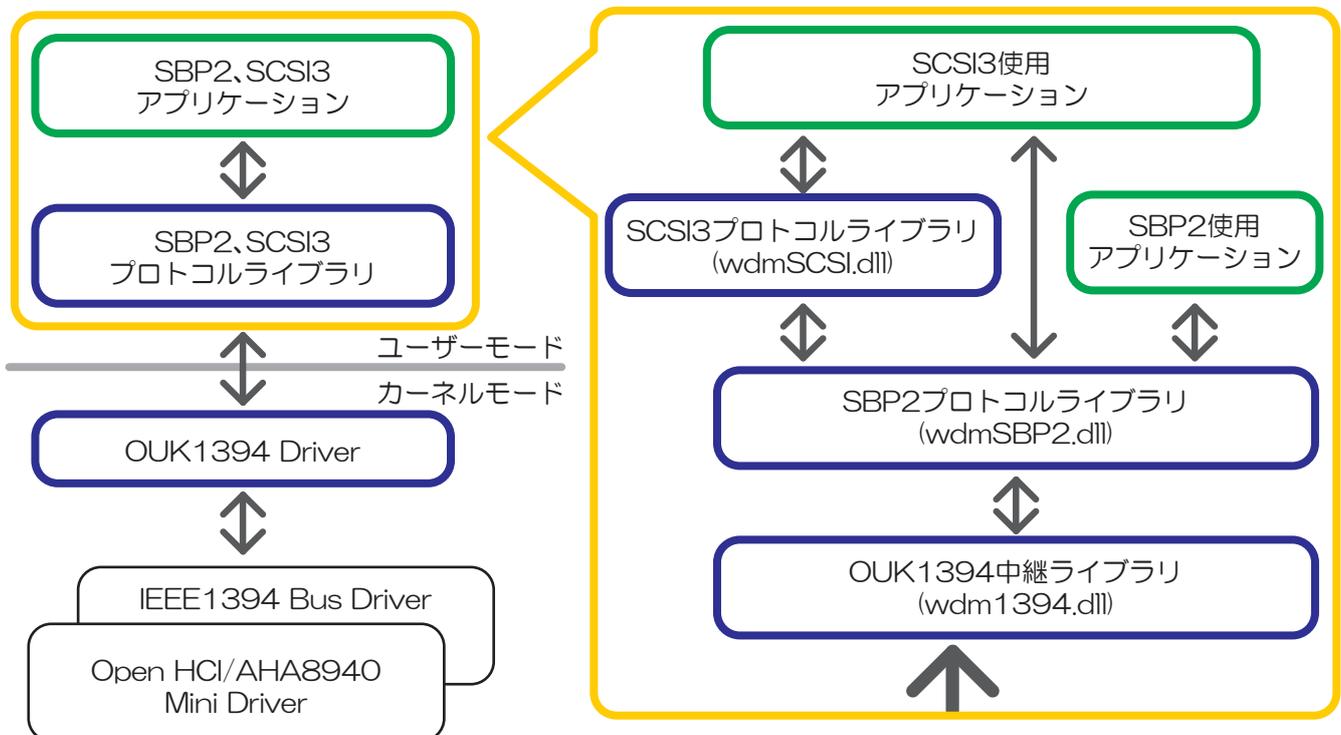
→ SBP2とは

SBP2 プロトコルはIEEE1394上にSCSI コマンドを載せるために策定されたもので、SCSI 制御のプリンタやスキャナ、ストレージデバイスなどが対象となっています。

現在では、多彩なプロトコルを扱うことが可能となっています。

SBP2 ライブラリ上で動作するSCSI3 プロトコル・ライブラリを使用することで、大容量記憶装置の制御アプリケーションの開発が可能となります。

現在、ASPI を使用してSCSI 機器を制御しているアプリケーションを、IEEE1394 に移植する際に御利用いただけます。



→ 周辺機器側(組み込み向け)ライブラリも用意

本ライブラリにはイニシエータ側(PC側)だけでなく、周辺機器側に搭載するファームウェアを対象にターゲット側ライブラリも用意しております。

→ 製品詳細

ターゲット・ハードウェアへの移植作業

個別のターゲット・ハードウェア(周辺機器)への対応は、本ライブラリ製品には含まれておりません。

移植作業を容易にするため、ライブラリからハードウェア依存部を切り離し、IEEE1394層(Windows においてはカーネルモード・ドライバ)と本プロトコル・ライブラリの間の中継ライブラリ(wdm1394.dll)を挟む形になっております。

移植に際しては、この中継ライブラリ相当の機能をターゲット側ファームウェアに用意していただくこととなります。

中継ドライバ(オーク1394 ドライバ:OUK1394.SYS)は、アプリケーションが必要とする情報や機能をカーネルモードから提供することで、性能向上を実現します。

SBP2ライブラリターゲット側関数一覧

ターゲット初期化

ログイン受付(ログインID 設定)

イベントハンドラ登録(:コマンドORB受信時/マネージメントORB 受信時)

コマンドORB(Operation Request Block)からの各種情報取得

リクエストに対するデータ/ステータスの書き込み

取り消し(アポート・タスク)要求の対象となったORBのアドレス取得

使用済みORBの解放

SBP2 ライブラリイニシエーター側関数一覧

イニシエーター初期化

各種イベントハンドラ設定

バスリセット時、コマンド処理完了時、ターゲット側ステータス受信時

ログイン・パラメータの設定/ターゲットに対するログイン/ターゲットからのログアウト

コマンドORB の発行(送信保留/ 自動送信)、保留されているコマンドORB の送信

処理完了ORB の解放

処理完了ORB の情報取得(ステータスブロック、コマンドブロック、データの各アドレス)

ログインID の列挙(クエリー・ログイン)

リクエストの取り消し(アポート・タスク)

リセット・リクエスト(ターゲット全体/Logical Unit 指定)送信

各マネージメントORB を送信します。

SCSI3 ライブラリ関数一覧

イニシエーター側リクエスト送信(リード,ライト,ライト・ベリファイ,シンクロナイズ・キャッシュ)

ターゲット側リード/ライト/ステータス送信

デバイス情報の取得(モードセンス) / 設定(モードセレクト)

ライブラリ内に保持するLogical Blockのサイズ取得、設定

CDB(Command DescriptorBlock:SCSIリクエストのデータ構造)からの各種情報取得

※ Windows®は米国Microsoft Corp.の米国、及び各国における登録商標です。



株式会社オーク

〒540-0029

大阪市中央区本町橋2-23 第7松屋ビル8F

TEL.06 (6945) 4704/FAX.06 (6945) 4705

WEB.http://www.ouk.jp/

技術本部営業課 滝井<takii@ouk.jp>

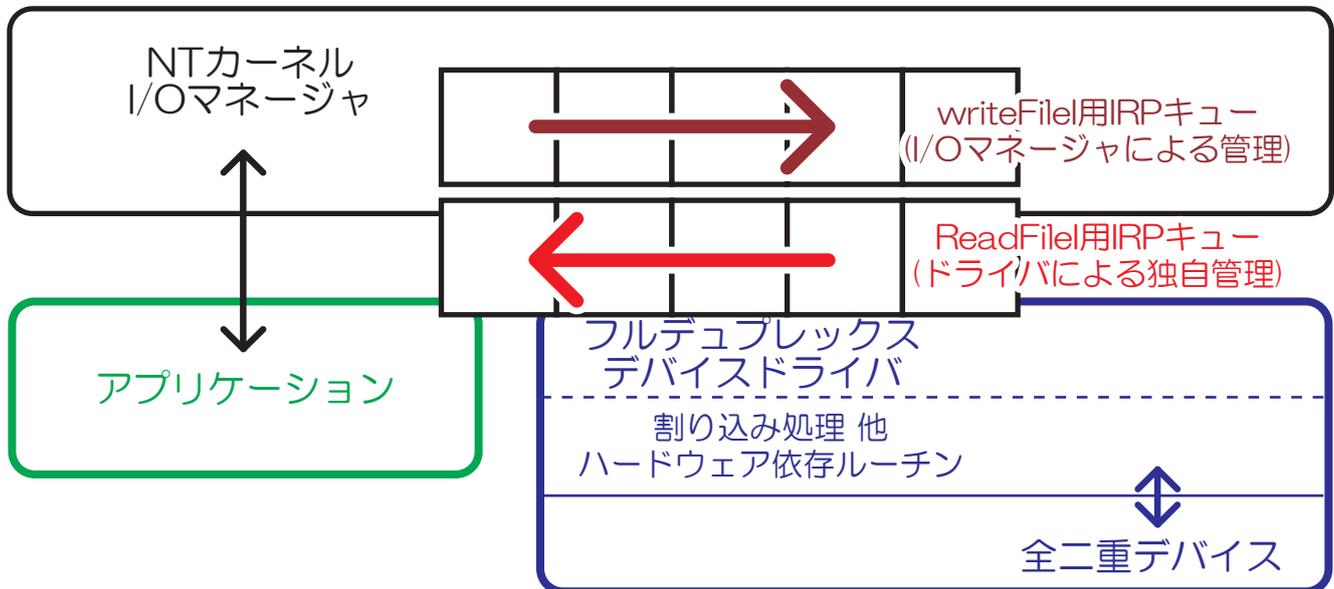
御用命は弊社販売代理店まで

フルデュプレックス・ドライバ

サポートOS WindowsNT® 4.0 Windows® 2000

→ 全二重デバイス用ドライバ

WindowsNT®のI/O マネージャーが提供する標準的なデバイスドライバの枠組みでは、一度に一つのリクエストしか処理することができません。
I/O マネージャーは送信側、受信側のどちらか先に到着したリクエストの処理完了まで、次のリクエストをドライバに渡すことはありません。結果として、データを取りこぼしたり、ハードウェアの性能を引き出せないということになります。



全二重(フルデュプレックス)通信をサポートするハードウェアでは、専用に設計したデバイスドライバ(フルデュプレックスデバイスドライバ)が必要となります。シリアルデバイスなどは、その典型的な例です。
フルデュプレックスデバイスドライバでは、リクエスト管理ルーチンをもう1セット追加し、新たなリクエストの処理ルートを用意します。I/Oマネージャーの機能と合わせて2つのルートを使って、送信側と受信側のリクエストを振り分けて処理する形になります。

→ 短期間での開発を実現

製品では、リクエスト管理ルーチンと代表的なドライバルーチンを予め実装しており、ソースコードとしてご提供いたします。

これらの実装済みルーチンにより、双方向のIRP(I/O RequestPacket)処理を実現しています。

従いまして、本製品をベースに、ハードウェアに依存する割り込み処理などを追加していただくことで、フルデュプレックスドライバが完成することになります。

いずれのルーチンも、弊社開発のデバイスドライバで使用された実績を持っております。

開発工程/テスト行程を短縮することができます。

※用途に応じて、多少の修正が必要となる場合もございます。

※本製品にはデバイスドライバ開発環境(開発言語、ドライバ開発キットDDK)は付属しません。

別途、各OS に対応した開発環境を御用意いただく必要があります。

→ 機能詳細

本ソフトウェア部品では、I/O マネージャーの内部ルーチンであるIoStartPacket()やIoStartNextPacket()と同等の機能を持つ関数を備えています。

これらの関数とI/Oマネージャーの機能を併用して、2つのリクエスト処理ルートを用意できます。

I/O リクエストは、まずディスパッチルーチンに渡されます。ディスパッチルーチンは必要な処理を行った後に、リクエストの種類によってI/O マネージャのルートを使用するか、フルデュプレックスドライバで用意した独自ルートで処理するか振り分けます。

独自のStartIoルーチン登録

独自のリクエスト処理ルート用のStartIo ルーチンです。一般的なStartIo ルーチンと異なり、独自に用意したStartPacket()関数等を使用するように変更されています。

このルーチンに、各デバイスドライバ固有の処理を追加します。

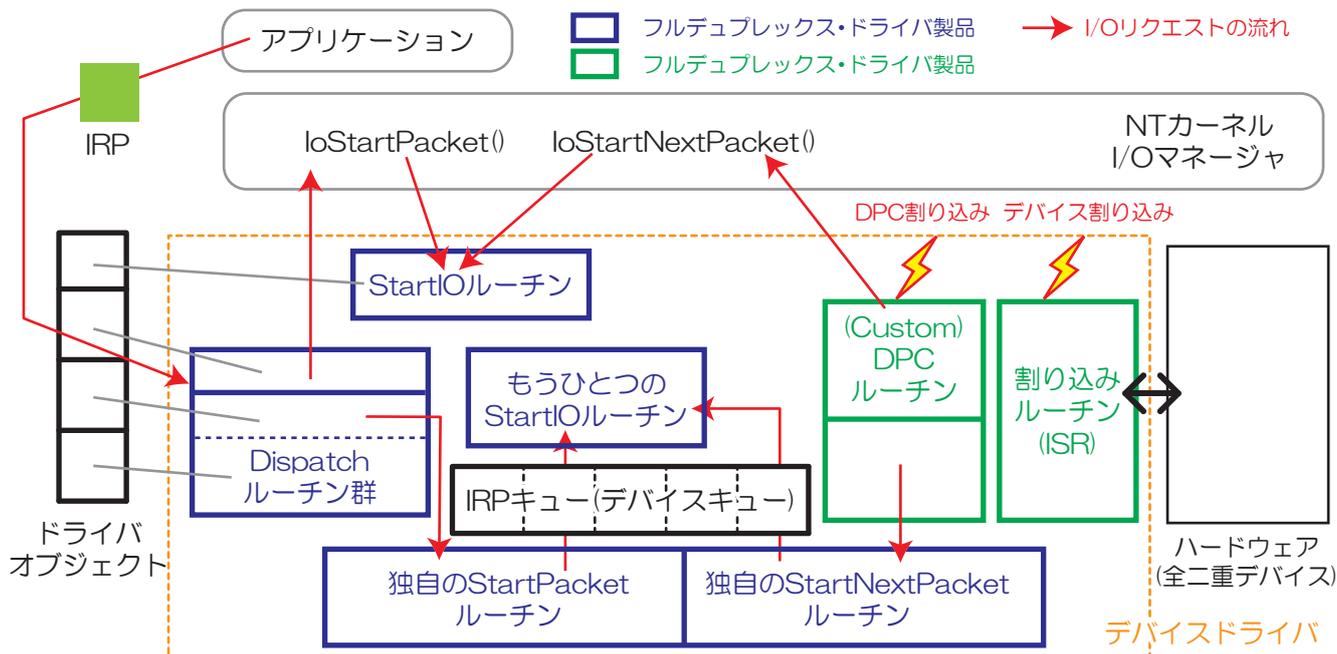
独自StartPacket()

ディスパッチルーチンから渡されたIRP(I/O RequestPacket)を、独自のリクエスト処理ルート用StartIo ルーチンに渡します。

既にリクエスト処理中であれば、IRP をデバイスキューに保存します。

独自StartNextPacket()

通常、DPC(Deferred Procedure Call:遅延呼び出し)ルーチンから呼び出されるもので、キューに保存された次のIRP 処理へ進めます。



弊社では、NT カーネル用のデバイスドライバ向けソフトウェア部品を、ラインナップとしてご用意いたしております。

ハードウェア割り込み処理に使用可能な「割り込み処理ライブラリ」についても、ぜひ御検討下さい。

※ Win32®, Windows®, WindowsNT®, は米国Microsoft Corp.の米国、及び各国における登録商標です。

ドライバ割り込みルーチン

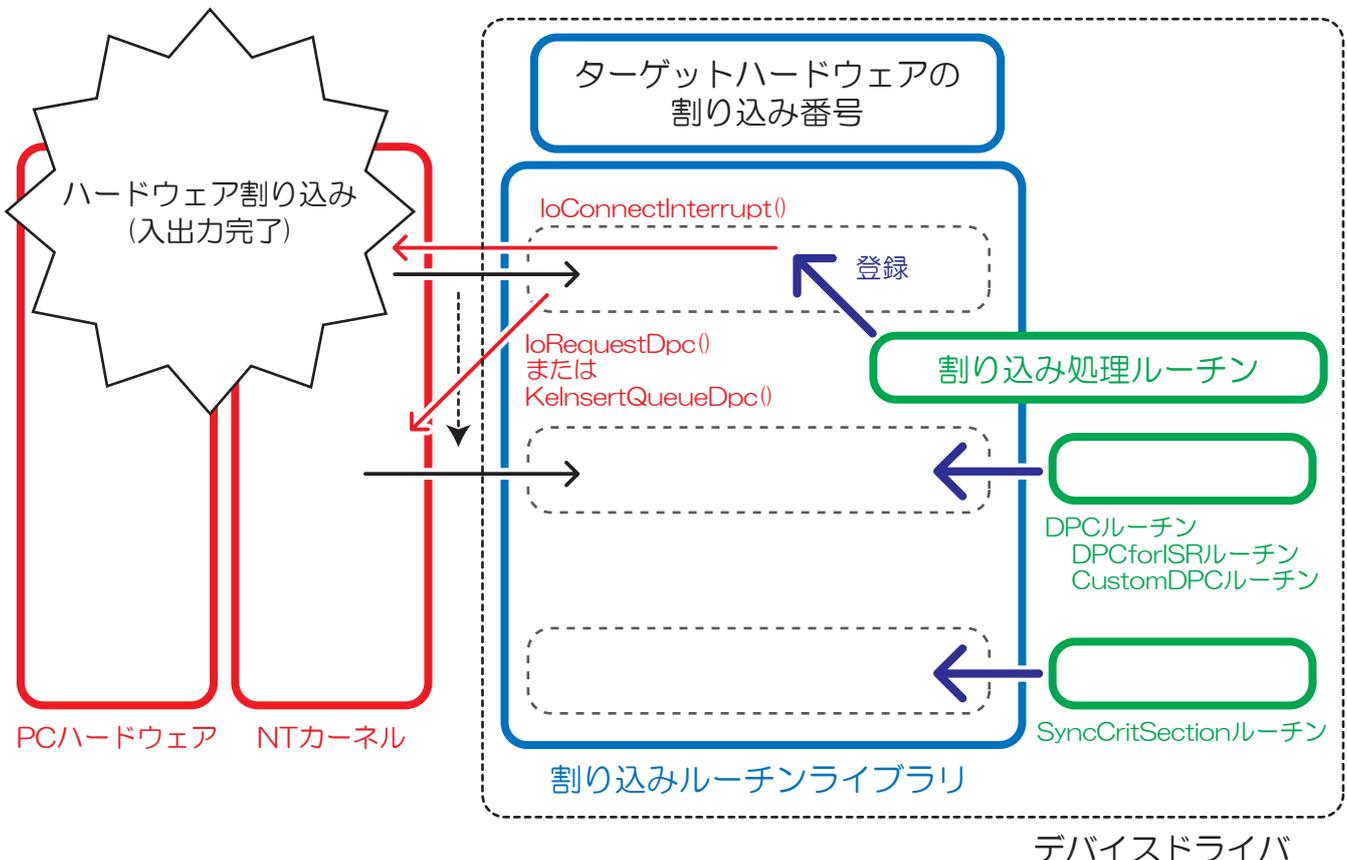
サポートOS WindowsNT® 4.0 Windows® 2000

→ デバイスドライバの割り込み処理

低レベルな物理デバイスドライバの実装に必要な、ハードウェア割り込み処理の定型部分を分離、製品化いたしました。割り込みルーチンの登録など、必要とされる一連の動作を実現しています。いずれも、実際にデバイスドライバに使用された実績のコードをベースに開発されています。信頼性の確保と短期間での開発を実現するソフトウェア部品です。
※なお、オークのソフトウェア部品には、すべて、製品ソースコードが付属します。

→ 製品概要

NT カーネルとのやりとりは既実装されており、個別のハードウェア制御に関する部分の設計・実装に専念することが可能です。また、テスト済みのコードを利用することで、動作テストに必要な期間も短縮することができます。本製品で扱えるのは、割り込み処理ルーチン(ISR - Interrupt Service Routine), SyncCriticalSectionルーチン、DPC(Deferred Procedure Call) for ISR ルーチン、Custom DPC ルーチンで、WindowsNT®, および後継OSで動作します。



※ 本製品にはデバイスドライバ開発環境(開発言語, ドライバ開発キットDDK)は付属しません。別途、各OS に対応した開発環境を御用意いただく必要があります。

→ 機能詳細

本ライブラリには、以下のような機能が搭載されています。

割り込み番号の取得

ターゲット・デバイスに対して割り当てられた、割り込み番号(割り込みベクタ)を取得します。

割り込み番号は、割り込み処理ルーチンを登録する際に必要となる情報です。

割り込みハンドラ(ISR)の登録

割り込み発生時の処理を担当する関数を登録します。

ISR(Interrupt Service Routine)はライブラリで実装され、デバイスエントリに設定されたルーチンが登録されたユーザー定義の関数を呼び出します。

クリティカルセクションの登録

デバイス・ハードウェアの操作など、排他環境下で実行しなければならない処理を担当するルーチン(SyncCriticalSectionルーチン)を登録します。

ISR ルーチン以外のデバイスドライバ中のコードが、KeSynchronizeExecutionを通じてクリティカルセクション・ルーチンを実行することで、ドライバルーチンの共有リソース(ハードウェア、状態情報、メモリ)を保護します。

DPCルーチンの登録

DPC(deferred procedurecall)ルーチンは、割り込み処理ルーチンの後を受ける形で呼び出されるルーチンです。割り込み処理ルーチンは、短時間で処理を終えなければならないという制限を持ちますが、DPC ルーチンはこの制限を緩和する役割を持っています。

応答時間の観点から、割り込み処理ルーチンでは対応できない処理を担当します。

個数が1 つに限定されているDPC forISRルーチン、複数のルーチンを状況に応じて使い分けることが可能なCustomDPC ルーチンの登録が可能です。

標準で提供される各ルーチンの実装について

その他、標準で実装されている各ドライバルーチンは修正が必要となります。

DriverEntry ルーチンの実装について

標準で実装された各ルーチンを登録します。

レジストリからのパラメタ取得、デバイスオブジェクトの作成等、各種デバイスドライバの仕様に合わせた動作のカスタマイズが必要となります。

その他のルーチンの実装について

低レベルな入出力をサポートするドライバルーチンとして、DMA 制御に使用するAdaptorControl / ControllerControl、タイムアウトを扱うためのIoTimer / CustomTimerDPC がありますが、これらのルーチンは標準では提供されておられません。

用途に応じて実装が必要になる場合がございます。

本ドライバは汎用設計になっておりますが、付属するソースコードをベースとして個別のハードウェアに対応した、より高機能な専用ドライバを開発することも可能です。

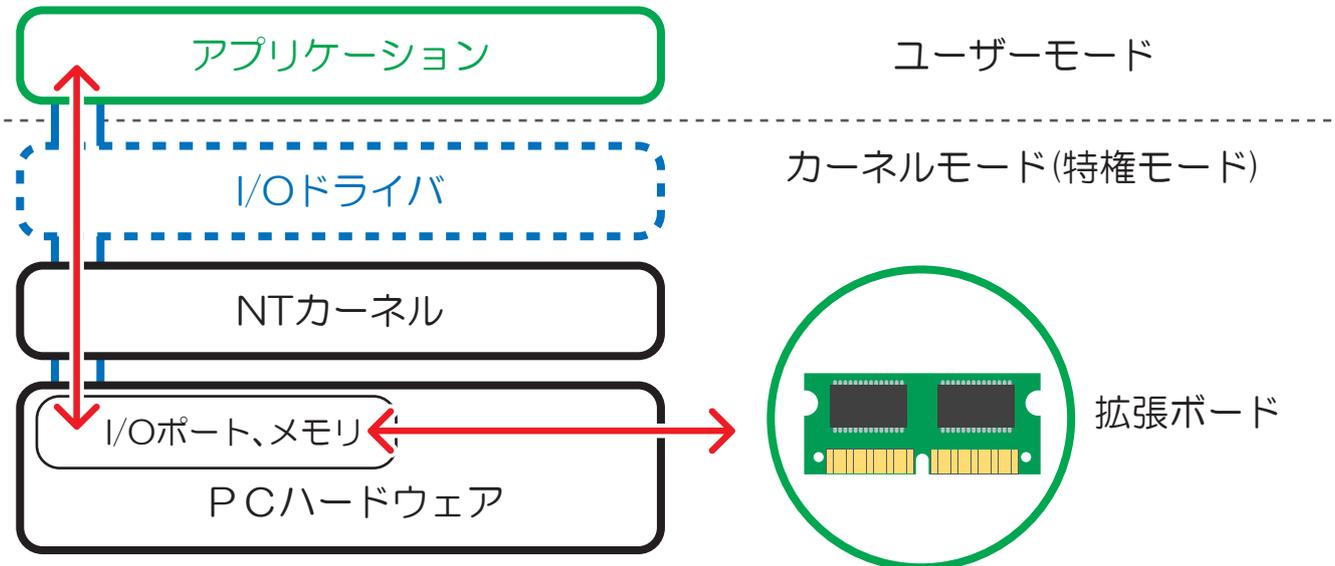
※ Windows®、WindowsNT®、は米国Microsoft Corp.の米国、及び各国における登録商標です。

カーネルモード・I/Oドライバ

サポートOS WindowsNT® 4.0 Windows® 2000

→ アプリケーションからI/Oポートにアクセス

I/Oポートなどカーネルが管理するハードウェアを使用するには、一般的なアプリケーションが動作するユーザーモードではなく、特権を持つ保護レベル(カーネルモード)で動作させる必要があります。本製品は、ハードウェア操作する部分のコードをデバイスドライバ(完成品)として提供、デバイスに対してI/Oコントロール・リクエストを発行することで、アプリケーションからのI/Oポートやメモリを使用した拡張ボードへのアクセスを可能にするものです。



ハードウェア/デバイスドライバ開発時の診断用ツールの一部として、拡張ボード制御用のアプリケーションと共に配布して、ご利用いただけます。

オークのソフトウェア部品には、すべて、製品ソースコードが付属します。

→ 機能概要

本ドライバを利用するアプリケーションは「I/Oドライバ」をオープンし、アクセスしたいハードウェア・リソースを指定して、デバイスI/Oコントロールを発行します。

●ハードウェア・リソース(I/Oポートアドレス、メモリアドレス)指定

デバイスが使用するハードウェア・リソースは固定で、ドライバインストール時に決定されます。ドライバ初期化時に、レジストリに保存された設定値を読み込みます。

アプリケーションからは相対的なポート番号/レジスタアドレスを指定します。

●データ幅

バイト(8bit - UCHAR)、ワード(16bit - USHORT)、ロング(32bit - ULONG)、各サイズのデータをそれぞれ読み込み、書き込み可能。

メモリアクセス(メモリマップトI/O)に関しては、更に、使用リソースとして予め宣言した範囲を上限として、任意のバイト数のデータを扱うことが可能です。

→ 製品詳細

製品には、I/Oポートドライバ(ドライバ完成品)、ドライバソースコード、アプリケーションのサンプルコード(コンソールツールによる使用例)が付属します。

本ドライバではDriverEntry、Dispatch、Unload ルーチンを実装しております。

DriverEntryルーチン

デバイス・オブジェクトの作成の他に、レジストリからデバイスドライバに割り当てられたハードウェア・リソースを読み取り、使用リソースの申告(IoReportResourceUsage)、論理アドレスへの変換(HalTranslateBusAddress)、I/O 空間のマッピング(MmMapIoSpace)等を実行します。

Dispatchルーチン

デバイスI/O コントロールのリクエスト(IRP)を判別し、リクエストを実行する適切な関数を呼び出します。最終的にIRPの処理完了をカーネル(I/Oマネージャー)に通知し、ステータスを返します。

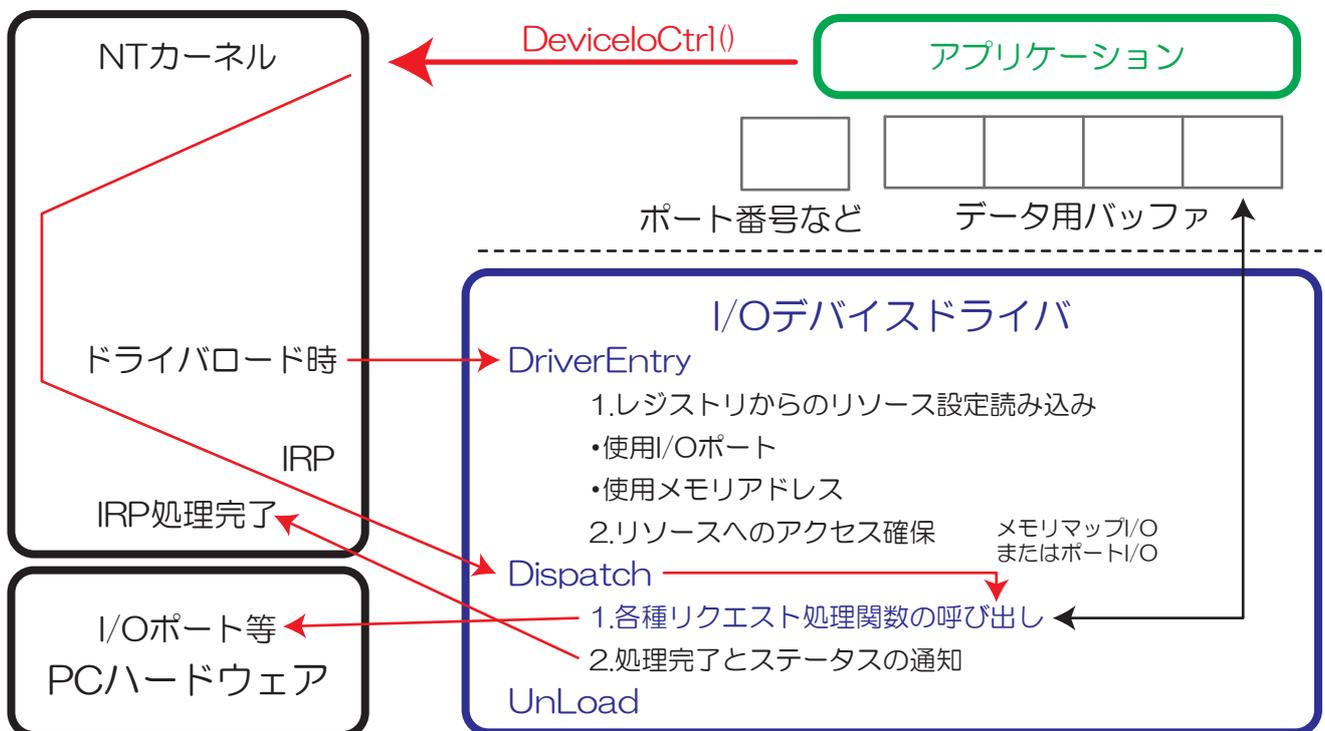
リクエスト処理関数

パラメータチェックなどを経て、実際にリクエストを処理する関数がI/O コントロールの種類ごとに用意されています。それぞれDispatch ルーチンから呼び出され、HAL(Hardware Abstraction Layer)ルーチンを使用してハードウェアにアクセスします。

Unloadルーチン

マッピング解除(MmUnmapIoSpace)、使用していたリソースの返却(IoReportResourceUsage)など、後始末を担当します。

本ドライバは汎用設計になっておりますが、付属するソースコードをベースとして個別のハードウェアに対応した、より高機能な専用ドライバを開発することも可能です。



※ Windows®, WindowsNT®, は米国Microsoft Corp.の米国、及び各国における登録商標です。

USBクライアント・ドライバ 専用制御ライブラリ

サポートOS Windows® 98/Me Windows® 2000/XP Macintosh®

USB機器の制御をアプリケーションから

本製品はUSB 機器を制御のためのドライバ・キットです。ドライバはカーネルモードで動作し、専用のユーザーモード・ライブラリが付属します。

専用ライブラリの最大の特徴は、**USB 機器を制御するための各種コマンドを、通常のアプリケーションから発行**できる点にあります。制約のあるカーネルモード・プログラミングは不要です。なお、ライブラリを他のデバイスドライバから使用することもできます。

- ベーシックな構成のUSB 機器のデバイスドライバとして
- USB 機器を直接制御するようなアプリケーションの開発に
- ハードウェアの動作検証等に

以上のような用途に、ご利用いただけます。

メモリカード・リーダーやCDROM ドライブなど、ストレージ用途には弊社の「USB ストレージ・ドライバ」をご検討下さい。

→ ドライバ/ライブラリ仕様

1 つのコンフィギュレーション、1 つのインターフェイスを持つUSB機器をサポート

USB 機器には複数のコンフィギュレーション、複数のインターフェイスを持つものもありますが、本ドライバ及びユーザーモード・ライブラリを使って制御できるのは、単一のコンフィギュレーション、単一のインターフェイスのみです。

より複雑なUSB機器の制御には、カスタム設計のドライバをご検討下さい。

複数のパイプをサポート

サポートする転送方式

- バルク転送
- インタラプト転送
- コントロール転送
- ベンダー・リクエスト
- クラス・リクエスト

Win32® のReadFile/WriteFile システムコールを経由して実現するリクエスト

- パイプに対するバルク転送
- インタラプト転送

Win32® のDeviceIoControl システムコールを経由して実現するリクエスト

- デバイスのリセット
- パイプのリセット
- 各コントロール転送

→ ユーザーモード・ライブラリ

オークのUSB クライアント・ドライバには、専用ライブラリが用意されています。
このライブラリを使用すれば、USBデバイス制御に必要なデバイス名の取得、各種コントロール/データ転送を容易に行うことができます。

ポート番号からデバイス名DeviceNameを取得

本デバイスドライバでは固定のデバイス名を持ちません。そのため、単一のドライバで複数のUSB機器を制御することができます。

デバイスドライバと専用ライブラリが連携動作するにはデバイス名が必要ですから、USB 機器のベンダーIDとプロダクトIDを指定して、デバイス名を取得することになります。

必要なら接続しているハブとポート番号を指定することもできます。

このような設計になっているのは、USB 機器を複数接続して、専用ライブラリを使用したプログラムによりハードウェアのテストへの利用を可能にするためです。

Windows®98、Windows®2000 において標準で動作します。

Windows®98 Second Edition、Windows®Millennium Edition については、OS の仕様変更により追加的なプログラムが必要となります。

使用目的に応じて補助プログラムの設計を変更する必要があるため、個別対応とさせていただきます。お気軽に御相談下さい。

デバイスのオープン (デフォルト・パイプ使用 / パイプ指定あり)

一連のUSB 機器制御の開始手続きとして、USB 機器のデフォルト・パイプ、あるいは指定したパイプをオープンします。

データの読み書き

バルク転送、インタラプト転送のリクエストを発行します。

デバイス/パイプのリセット

停止(Halt)、あるいはエラー状態にあるUSB機器、あるいは特定のパイプをリセットします。

ベンダー・リクエスト、クラス・リクエストの発行。Clear Featureリクエストの発行。

USB 機器の製造元が独自に設定できるベンダー・リクエスト、USB 機器の種類(Device Class)に応じて規格で定められているクラス・リクエストを発行します。

また、USB 機器の機能/状態(Feature)を変更するClear Feature リクエストを発行することもできます。

デバイス情報の取得

USB 機器の固有情報であるデバイス・ディスクリプタ他を取得できます。

デバイスの機能等の情報を格納したデバイス・ディスクリプタ

デバイス構成や消費電力に関する情報であるコンフィギュレーション・ディスクリプタ

インターフェイスに関する情報であるインターフェイス・ディスクリプタ

各エンドポイントの転送能力等を記述したエンドポイント・ディスクリプタ

を取得することができます。

※ Windows®、Win32®は米国Microsoft Corp.の米国、及び各国における登録商標です。

軽量HTTPサーバ

依存API：BSD TCP/IPソケット ファイルシステム(PDOS)

組み込み機器におけるWEBサーバの活用

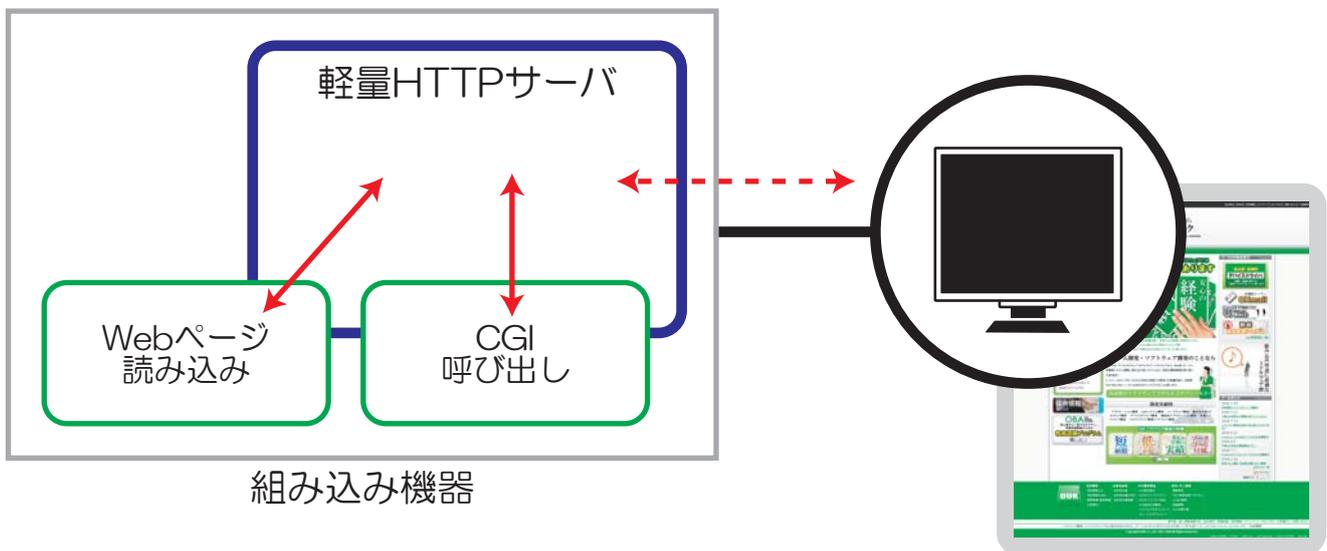
現在、パソコンに限らず、様々な機器にWeb ブラウザが搭載され、その共通した操作方法が浸透しつつあります。情報発信の窓口として

Webサーバ(HTTP サーバ)を組み込むことによって、例えば、機器内部の状態やデータをブラウザにグラフィカルに表示することができます。

また、Web ページを表示側は、TCP/IP インターフェイスとWeb ブラウザを搭載するものであれば、直接接続して、ネットワーク越しに遠隔地から、と利用形態を問わないという利点もあります。

設定・制御インターフェイスとして

対象機器にWeb サーバを組み込めば、機器の動作設定などの用途で専用のプログラムを構築する代わりに、Web ページのインターフェイスを利用することもできます。



→ 主な特徴

基本的に、BSD TCP/IP ソケット・インターフェイスが利用できれば、動作・移植することは可能です。また、ファイルを読み書きするため、通常の用途ではファイルシステムがサポートされている必要があります。

※標準では、弊社開発のFAT ファイルシステムPDOS を使用しています。

現在、サポートしているOS(RTOS)としてはeCos[®]、μ ITRON があります。

その他のOS への対応については、弊社までお気軽にお問い合わせください。

なお、オークのソフトウェア部品には、全て、製品ソースコードが付属します。

※ eCos[®]は米国RedHat、Inc の米国及び、各国における登録商標です。

→ 製品詳細

GET、POST、HEAD リクエストに対応

最低限必要な機能を搭載した、非常にコンパクトなWeb サーバとなっています。

CGI 対応

CGI開発ではC言語(ネイティブコード)、弊社開発のシェル・インターフェイス用のスクリプト言語を利用することができます。

アクセス制御:基本認証をサポート

ディレクトリごとに、各ユーザーに対してパスワードを設定することができます。

パスワードが設定されたディレクトリにあるページにアクセスすると、ブラウザはユーザー認証のためのダイアログを表示します。パスワードが一致しなければ、Web サーバはエラーを返し、リクエストしたページは表示されません。

→ サーバ制御用インターフェイス

Webサーバの制御には、C言語、あるいは弊社開発のシェル・インターフェイスを使用します。

C 言語用のインターフェイスとしては、Web サーバの起動/停止、C 言語で記述されたCGIを登録するための関数が用意されています。なお、シェル・スクリプトで記述されたCGIは、シェルを起動してURLで指定されたファイルを解釈/実行します。

→ 関連製品

本製品は、標準では、次のソフトウェア部品と連携動作する設計になっております。

「軽量HTTP サーバ」と併せてご検討ください。

FAT ファイルシステムPDOS

UNIXO システムのシステムコールに似たAPIを持つFAT ファイルシステムです。

シェル・インターフェイス

BSD csh と似たスクリプト実行機能を持つコマンドライン・インターフェイスです。

電源投入時の初期設定、Web サーバ制御、CGI スクリプト開発等に御利用いただけます。

FTP サーバ

「軽量HTTP サーバ」はPUT リクエストをサポートしておりませんので、ネットワーク・インターフェイスを使用してファイル(Web ページ用のデータ)を転送する場合には、FTP サーバ経由でアップロードを行うことになります。

各製品は独立した製品となっております。

詳細については、お気軽にお問い合わせください。

※ UNIXは米国The Open Group が管理する米国、及び各国における登録商標です。

軽量TELNETサーバ

依存API: BSD TCP/IPソケット シェル

→ TELNETサーバ

TELNET を利用すると、ネットワーク経由で文字データの交換を実現することができます。

ネットワーク経由で遠隔制御

遠隔地からマシンを操作するインターフェイスとして、UNIXなどのOSで利用されています。

TELNET プロトコルに対応したソフトを使用して、サーバが稼働しているマシンに接続、コマンドを打ち込んで実行させます。



柔軟で高度な設定インターフェイスとして

TELNET サーバを利用した場合、表示可能な情報は文字だけに限られますが、別途用意したコマンドによる細かな設定が可能となります。

一般にブラウザによる設定画面では、想定した以外の利用形態には対応が難しいという欠点があります。

→ 動作環境

基本的に、BSD TCP/IP ソケット・インターフェイスが利用できれば、動作・移植することは可能です。

また、入力されたコマンド文字列を解釈、処理するソフトウェア(=シェル)が別途必要となります。シェルの機能や認証情報を供給するパスワード・ファイルのために、ファイルシステムが必要になる場合もあります。

現在、サポートしているOS としてはeCos、μITRON があります。その他のOS への対応については、弊社までお気軽にお問い合わせください。

→ サーバ制御用インターフェイス

コマンドライン・インターフェイス(※1)とプログラミング・インターフェイスの両方から、TELNETサーバを制御することが可能です。

以下のようなサーバ制御用インターフェイスが用意されています。

- サーバの起動/停止
- ユーザーの追加/削除
- パスワードの変更
- ユーザー認証
- ホーム・ディレクトリ取得
- サービス・アクセス制御(TELNET の利用許可)

通常、ユーザー・アクセス制御に関する情報は、パスワード・ファイルによって供給されます。

TELNET サーバは、ネットワーク経由でから送信されたコマンド文字列をコマンド解析ルーチンに渡し、実行結果をTELNET クライアントに送り返します。標準では、コマンド解析ルーチンとして、弊社開発の「高機能シェル・インターフェイス」を使用しています。

コマンド解析ルーチンは次のような処理を順番に実行します。

- 1.ログイン・プロンプトの表示
- 2.入力されたパスワードを使用したユーザー認証
- 3.ユーザーごとに用意されたホーム・ディレクトリに移動
- 4.プロンプトを表示して、ユーザーからのコマンド入力待ち
- 5.入力された文字列を解析し、コマンドを実行
- 6.TELNET サーバに対して実行結果を出力
- 7.ユーザーがログアウトするまで、4~6 を繰り返す

※1:ソフトウェア部品「高機能シェル・インターフェイス」を使用

→ 関連製品

本製品は、標準で、次のソフトウェア部品と連携動作する設計になっております。

「軽量TELNET サーバ」と併せてご検討ください。

高機能シェル・インターフェイス

BSD csh と似たスクリプト実行機能を持つコマンドライン・インターフェイスです。

ファイル操作関係のコマンドを標準搭載し、コマンド追加して機能を拡張することも可能です。

TELNET のコマンドライン・インターフェイス(シェル)、電源投入時の初期設定、サーバ起動制御等に御利用いただけます。

FAT ファイルシステムPDOS

UNIXシステムのシステムコールに似たAPIを持つFAT ファイルシステムです。

各製品は独立した製品となっております。詳細については、お気軽にお問い合わせください。

なお、オークのソフトウェア部品には、全て、製品ソースコードが付属します。

※ eCos®は米国RedHat, Inc の米国及び、各国における登録商標です。

※ UNIXは米国The Open Group が管理する米国、及び各国における登録商標です。

高機能コマンドライン・インターフェイス

依存API: ファイルシステム(PDOS) 専用移植レイヤー(PBIOS)

高機能コマンドライン・インターフェイスを提供

本製品はBSDのCシェルに似た、コマンドライン・インターフェイスを実現します。
ネットワーク・インターフェイス経由のTELNET、あるいはシリアル・インターフェイスを使用して、コマンドプロンプトを表示、コマンド入力、実行結果の表示機能を提供します。

設定インターフェイスとして

デバックコンソールに組み込んで

等に、ご利用いただけます。

→ 主な機能

ファイル操作コマンドを標準搭載

ファイルの一覧表示(ls)など代表的なファイル操作コマンド、ファイル名のワイルドカード展開機能、出力ダイレクト機能が標準で用意されています。

また、新たにコマンドを追加することも可能です。

スクリプト機能

シェル・インターフェイスにはスクリプト機能が用意されています。例えば、電源投入時の初期設定をスクリプトファイルから実行することができます。

スクリプトでは、文字列変数として、システムに共通のグローバル変数、スクリプトファイル内のみで有効なローカル変数を区別して使用することができます。

制御構文としては、条件分岐(if-else)、while ループ(条件判定)、for ループ(数値範囲指定)、foreach ループ(候補リスト指定)が用意されています。

→ 動作環境

本製品では、システム固有のコードとして文字入出力、ファイルシステム部分を独立させています。該当する機能を移植/無効化すれば、任意のプラットフォームで動作するようになります。

現在、サポートしているOSとしてはeCos[®]、 μ ITRONがあります。

その他のOSへの対応については、弊社までお気軽にお問い合わせください。

なお、オークのソフトウェア部品には、全て、製品ソースコードが付属します。

→ 搭載コマンド一覧

PDOS にはプログラミング・インターフェイスとして、以下の関数が用意されています。

標準搭載コマンド

指定スクリプト起動run

指定時間の待ちsleep

変数設定set

文字列出力echo

数式評価expr

ラインエディタedit

ファイルシステムmount、format、df、ls

ディレクトリ操作pwd、cd、mkdir、rmdir

ファイル操作cp、mv、rm、cmp、cat、more、touch、chmod

システム日付操作date

この他にも、弊社開発の組み込み向けソフトウェア部品を制御するために、各種コマンドが用意されています。

拡張コマンド

標準搭載のもの以外でも、自由にコマンドを追加することも可能となっています。

引数処理、専用の入出力ライブラリを使用してのコマンド開発が可能です。

→ 関連製品

本製品は、標準で、次のソフトウェア部品と連携可能な設計になっております。
「高機能コマンド・インターフェイス」と併せてご検討ください。

ファイルシステムPDOS

UNIX システムのシステムコールに似たAPIを持つFAT ファイルシステムです。

FTP サーバ

TELNET サーバ

コマンドラインからのユーザ・アカウント管理、サーバの起動制御が可能です。

HTTP サーバ

本製品搭載のスクリプト機能によるCGI 開発、コマンドラインからのサーバの起動制御が可能です。

各製品は独立した製品となっております。詳細については、お気軽にお問い合わせください。

※ UNIX®は米国The Open Group が管理している米国、及び各国における登録商標です。
※ eCos®は米国RedHat, Inc の米国、及び各国における登録商標です。

PDOS-FATファイルシステム

依存API: 専用移植レイヤー(PBIOS)

組み込み機器向けFATファイルシステム

PDOS はMS-DOS[®]FAT12/16 ファイルシステムをサポートするソフトウェア部品です。

API としては、UNIX[®]システムのopen、create、read システムコール等と同様のインターフェイスを採用しています。

FAT ファイルシステムのサポートがないRTOSを対象として、実績のある高品質なコードを低価格でご提供いたします。

組み込み機器用のファイルシステムとして

専用のファイルシステムを設計・開発するのと比較して、コスト削減を達成できます。

PCとのデータ交換用として

PC やメモ리카ードの普及により、メモ리카ード・スロットを搭載してデータ交換することが可能となりました。

FAT ファイルシステムは、自身がパソコン用のOSで使用され、多くのPC向けOSや機器でサポートされているファイルシステムです。

※ 32bit クラスタ番号を管理することは可能ですが、現在、正式にはFAT32/VFATに対応しておりません。詳細については、弊社までお問い合わせください。

→ 動作環境

現在、動作実績のあるOS(RTOS)としては、eCos[®]、Linux(POSIX)、 μ ITRON3、VxWorks[®](tornado)があります。

その他のOS への対応については、弊社までお気軽にお問い合わせください。

なお、オークのソフトウェア部品には、全て、製品ソースコードが付属します。

→ 移植側インターフェイスについて

PDOS システムは下位に位置する、PBIOS ルーチンを使用してコーディングされています。

- ・デバイスに対するセクタ単位の読み書き
- ・FAT 領域とデータ領域の一貫性を保つための排他制御
- ・スレッド(タスク)単位のメモリ確保

本製品を使用するには、システムに依存する、これらのPBIOS ルーチンを移植する必要があります。

→ API一覧

PDOS にはプログラミング・インターフェイスとして、以下の関数が用意されています。

パス指定について

PDOS を利用するプログラムからは、パス文字列によりファイル/ディレクトリを指定します。

区切り文字はバックスラッシュ("/")を使用します。

通常PCでは、ドライブの識別にアルファベット1文字(C:など)を使用しますが、PDOS にはドライブ・レターではなく、パス文字列のトップレベルの名前により識別します。

例えば、メモ리카ード・スロットが2 台であれば、/MCO/FILENAME.TXT、/MC1/FILENAME.TXT といったパス指定で区別します。

このメモ리카ードをPC で参照すると、ルートディレクトリにFILENAME.TXT というファイルが見える形になります。

デバイス初期化(PDinitial)

デバイス、PDOS 内部を初期化して、FAT ファイルシステムを使用できるようにします。

マウント(PDmount)、デバイス・マウント情報の取得(PDfstab、PDfstab2)

PDOS からデバイスを利用できるようにするには、デバイスとトップレベルのパス名を結び付ける作業(マウント)が必要です。

また、マウント情報を取得して、利用可能なデバイスをリストアップすることもできます。

空き容量取得(PDvolume)

バイト単位で空きディスク容量を取得します。

ファイル操作

オープン/クローズ: PDopen、PDcreate、PDclose、PDcloseAll

読み書き: PDread、PDwrite

ファイルポインタの移動(PDlseek)、現在位置取得(PDtell)、ファイル終端検査(PDeof)

ファイルに関する各種情報を取得(PDfstat)

属性変更(PDchmod)、最終更新時刻の変更(PDutime)

名前変更(PDrename)、ファイル削除(PDunlink)

ディレクトリ操作

現在のディレクトリ取得(PDgetcwd)、変更(PDchdir)

ディレクトリの作成(PDmkdir)、削除(PDrmdir)

ファイル検索

ディレクトリのオープン(PDopendir)、クローズ(PDclose)

オープンしたディレクトリに指定したファイルがあるか検索(PDreaddir)

※ MS-DOS®は米国MicrosoftCorp.の米国、及び各国における登録商標です。
 ※ UNIX®は米国The Open Group が管理している米国、及び各国における登録商標です。
 ※ eCos®は米国RedHat, Inc の米国、及び各国における登録商標です。
 ※ VxWorks®は米国Wind River Systems, Inc.の米国、及び各国における登録商標です。
 ※ Linux®はLinus Torvalds 氏の米国、及び各国における登録商標です。

ファイル転送プロトコル (FTP) サーバ

依存API: BSD TCP/IPソケット ファイルシステム(PDOS)

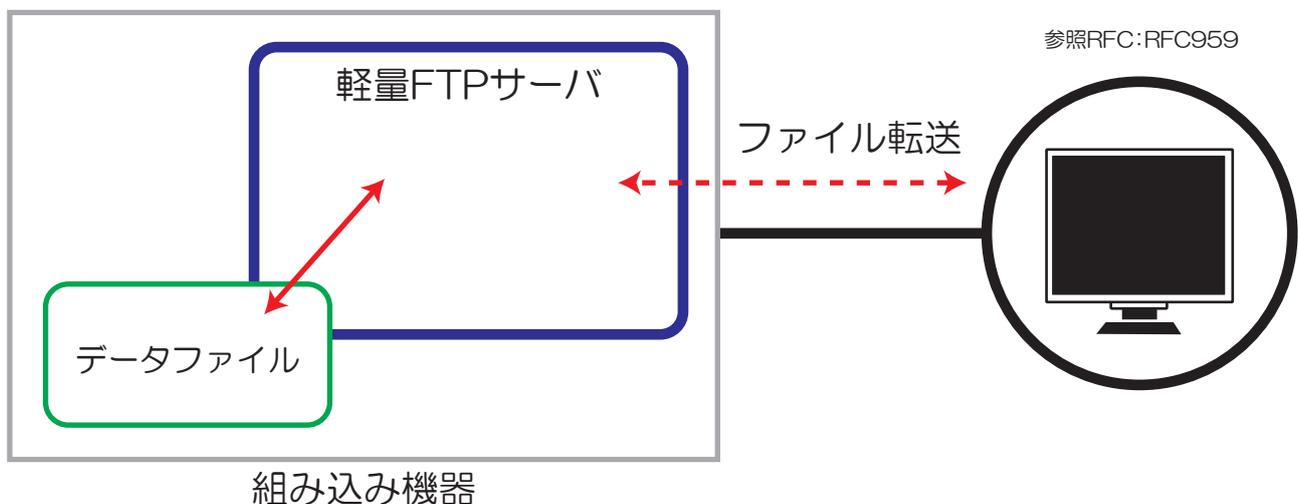
→ FTP - ファイル転送の標準プロトコル

TCP/IP ネットワークで、ファイル転送用として標準的に使用されているのがFTP(FileTransfer Protocol)です。FTP に対応したクライアント・ソフトを使用して、FTP サーバからファイルをダウンロード、組み込み機器へファイルをアップロードすることが可能です。

ネットワーク・インターフェイスを持つ機器に本製品を組み込むことにより、

- ファイルとして保存したデータの取り出しや書き込み
- 新しいファームウェアの転送

等に利用することができます。



→ 動作環境

基本的に、BSD TCP/IP ソケット・インターフェイスが利用できれば、動作・移植することは可能です。また、ファイルを読み書きするため、通常用途ではファイルシステムがサポートされている必要があります。

※標準では、弊社開発のFAT ファイルシステムPDOS を使用しています。

現在、サポートしているOS(RTOS)としては、eCos[®]、μITRON があります。

その他のOS への対応については、弊社までお気軽にお問い合わせください。

なお、オークのソフトウェア部品には、全て、製品ソースコードが付属します。

※ eCos[®]は米国RedHat, Inc の米国、及び各国における登録商標です。

→ サーバ制御用インターフェイス

FTP サーバ制御のためのインターフェイスとして、コマンドライン・インターフェイス(※1)とプログラミング・インターフェイスの2種類が用意されています。

シリアル接続やネットワーク(TELNET)接続のコンソールからコマンドを入力するか、C言語などを使用して専用の管理プログラムを開発して、FTP サーバを制御することが可能です。

以下のようなサーバ制御用インターフェイスが用意されています。

- サーバの起動/停止
- ユーザーの追加/削除
- パスワードの変更
- ホーム・ディレクトリ取得
- サービス・アクセス制御(FTP の利用許可)

通常、ユーザー・アクセス制御に関する情報は、パスワードファイルによって供給されます。

※1:ソフトウェア部品「高機能シェル・インターフェイス」を使用

→ 関連製品

本製品は、標準では、次のソフトウェア部品と連携動作する設計になっております。

「軽量FTP サーバ」と併せてご検討ください。

●FAT ファイルシステムPDOS

UNIXシステムのシステムコールに似たAPIを持つFAT ファイルシステムです。

●高機能シェル・インターフェイス

BSD csh と似たスクリプト実行機能を持つコマンドライン・インターフェイスです。

電源投入時の初期設定、FTP サーバ制御等に御利用いただけます。

●軽量TELNET サーバ

弊社開発の「高機能シェル・インターフェイス」と併用することで、TELNET クライアントからネットワークを経由したサーバ制御が可能となります。

各製品は独立した製品となっております。詳細については、お気軽にお問い合わせください。

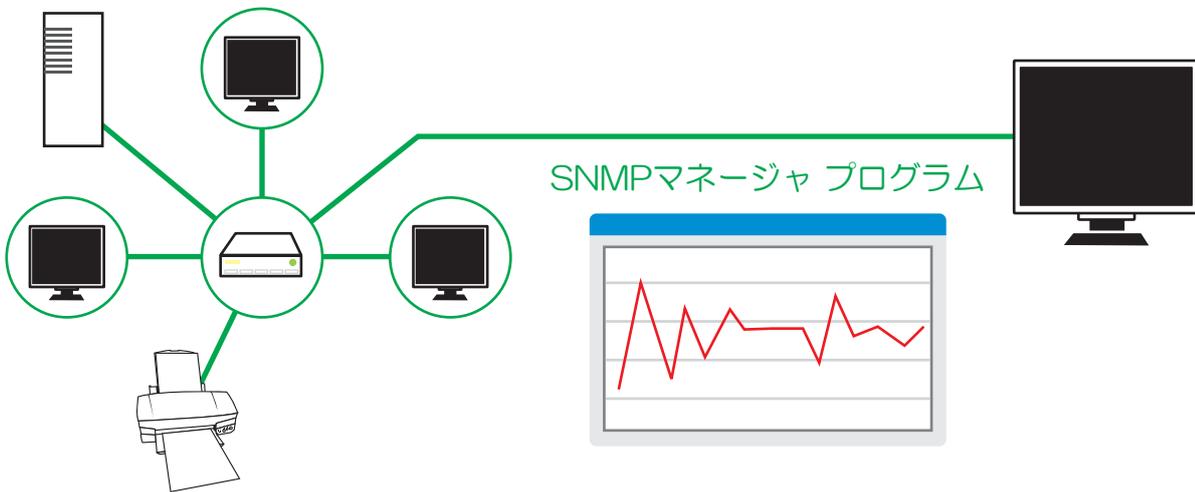
※ UNIXは米国The Open Group が管理する米国、及び各国における登録商標です。

搭載用SNMPエージェント

依存API：BSD TCP/IPソケット

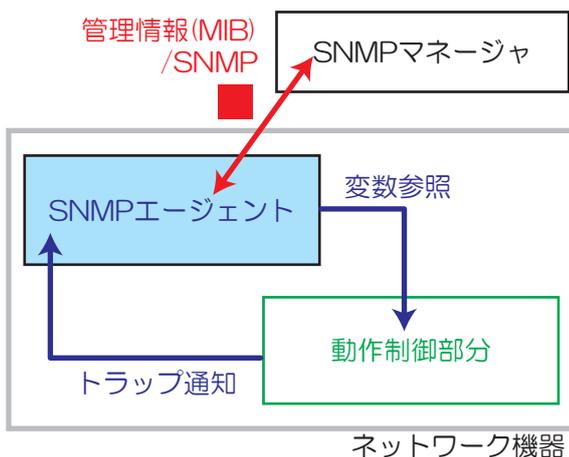
→ SNMP - ネットワーク機器管理の標準プロトコル

SNMP は、ネットワークで接続された各種機器(ルーターなど)の動作をモニタするためのプロトコルで、多くのネットワーク機器やワークステーションでサポートされています。
CPU 負荷、入出力データ量、エラーとなったデータ量などを、ネットワークを介して離れた場所から監視することができます。



ネットワーク機器 (SNMP エージェント) の情報は、管理端末 (SNMP マネージャ) に集約されます。SNMP マネージャは、ネットワーク上の SNMP エージェントからのパフォーマンス情報や動作停止など障害情報を、グラフィカルに画面表示する機能を持ちます。通知する情報 (MIB, 管理情報) については、機器の種類に応じて標準的なものが予め定義されている他、メーカー独自の 변수を定義して使用することもできます。

→ 組み込み向けSNMPエージェントプロトコルスタック



本製品は、ネットワーク機器向けに用意された、SNMP エージェントソフトウェア部品です。機器制御コードと関係動作し、SNMP 対応機能を付加します。SNMP エージェントライブラリは、SNMP マネージャのリクエストに応じて、機器のステータス情報にアクセス、レスポンスを返します。多くの SNMP マネージャは、画面の表示やメールを通じて、動作状況の異常を通知する機能を持っています。特定のイベント (トラップ) が起きた際に、エージェントライブラリを介してユーザーに通知することも可能です。

音声・映像配信プロトコル (RTP)

依存API: BSD TCP/IPソケット

→ 音声・映像配信の基盤技術

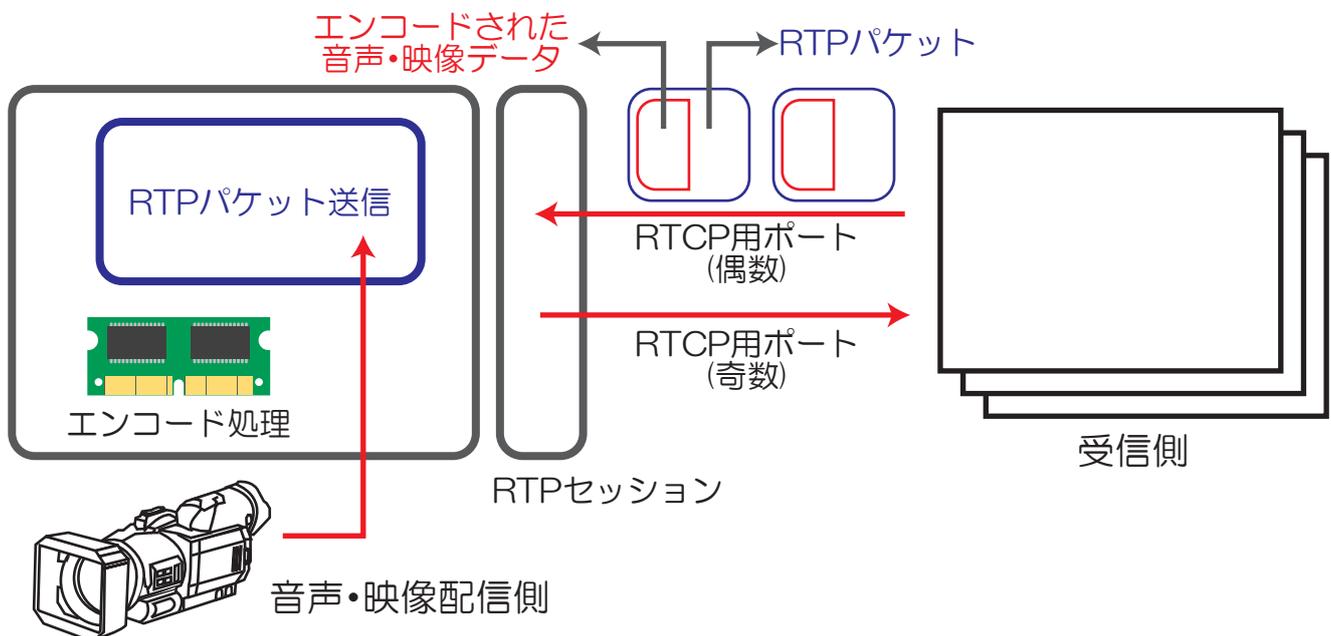
ネットワークを介した音声、映像配信(ビデオ会議やインターネット放送)において、その中心部分で使用されているのがReal-Time Transfer Protocol(RTP: RFC1889)です。

音声や映像のような連続したデータを扱う上で問題となる、ネットワーク途中経路でのデータ欠落、配送遅延への対策が盛り込まれています。

また、扱うデータの形式は問わないという特徴も備えています。RTPを使用するアプリケーションはRTPパケットのヘッダ中のペイロード・タイプで判断できます。

受信状況(配信遅延などの情報)は、RTPとは別のReal-Time Transfer Control Protocol(RTCP)によってレポートされます。

定期的に送られてくる情報をモニターしてパラメーターを変更することで、音声の途切れを低減するような配信システムを構築することも可能です。



本ソフトウェアはRTPを使用した配信システムのサーバ用(送信側:図中の青枠部分)に用意された製品です。

→ 主な特徴

基本的に、BSD TCP/IPソケット・インターフェイスが利用できれば、動作・移植することは可能です。

現在、サポートしているOSとしてはeCos[®]、μITRONがあります。

その他のOSへの対応については、弊社までお気軽にお問い合わせください。

なお、オークのソフトウェア部品には、全て、製品ソースコードが付属します。

→ 製品詳細

本製品は、RTP による音声・映像配信システムの中で、データ・パケット送信部分を担当するソフトウェア部品です。

エンコードされたデータをRTPのパケットに載せ、配信するのが主な機能です。

従いまして、以下のような処理は別途用意していただく必要があります。

●個別のフォーマットに応じた音声、映像データのエンコード処理

デバイスから入力されたデータをソフトウェア的/ハードウェア的に圧縮・エンコード処理し、本ソフトウェアに渡す必要があります。

●例えば、インターネット放送であれば、番組表の配布や受信開始手続きといった仕組み

データ配信に使用するIP アドレス、ポート番号といった情報の配布は、RTPが規定する範囲に含まれていません。ポート番号の場合、連続する2つポートを1組として、データ送信用に偶数番号を持つポート、RTCP用に奇数番号を持つポートを使用すること以外は決まっています。

●クライアント・ソフトウェア

●RTCP を利用した受信状況のモニタリング、フィードバック・システム

→ API

C 言語用のプログラミング・インターフェイスとして、以下の関数が用意されています。

●RTPセッションのオープン

宛先IP アドレス(通常はマルチキャスト・アドレス)、ポート番号、使用するデータ・タイプ(エンコード方式、RTP ペイロード・タイプ)を指定して、RTP を使用できるようにします。

●RTPデータ配送

エンコードされたデータをRTP パケットにパックして送信します。

指定されたRTP ペイロード・タイプに従って、個別のペイロードに応じたヘッダ、RTP パケットのヘッダを追加して配信します。

現在、JPEG映像:RTP ペイロード・タイプ26 RFC2345、音声にのみ対応しています。

最新の対応状況についてはお問い合わせ下さい。

●RTPセッションのクローズ

オープンしたセッションの使用終了を通知します。

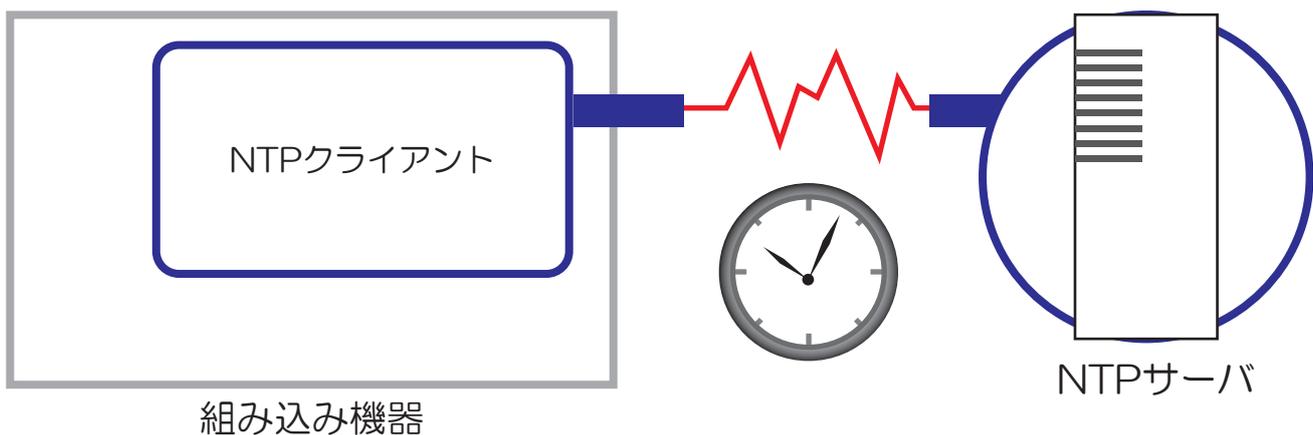
※ eCos®は米国RedHat、Inc の米国及び、各国における登録商標です。

ネットワーク時刻同期プロトコル (NTP)

依存API: BSD TCP/IPソケット

→ 時間設定を不要に

時計を内蔵している機器を導入した際に最初に行われる操作は、必ずといって良いほど時刻設定となるでしょう。NTP(Network Time Protocol)は、この時刻設定を自動的に行うためのプロトコルです。NTPクライアント・ソフトウェアは、ネットワーク上に用意されたNTPサーバと通信して、組み込み機器の内蔵時計が保持している時刻のずれを自動的に修正します。



分散システムの正常な動作のために

分散して配置された機器が、時計を利用して連携動作するようなシステムにおいて、時刻設定の正確を維持することは、システム全体の正常な動作のために不可欠な要素です。

例えば、次のような特徴を持つシステムです。

- タイムスタンプを利用して、データの新旧を判定するような処理
- 予め定められたスケジュールで、何らかのアクションを開始するシステム

つないで、すぐに使える機器のために

今後、家電製品を含めて、様々な機器がネットワークに接続するようになると予想されます。

身近な例では、どんなに便利な録画予約機能でも、時計が正確な時刻を指していなければ意図した動作はしません。

- 対象機器をネットワークに接続し、電源を入れるだけで利用可能になります。
- 停電などの原因により、内蔵時計の情報が失われる問題に対する有効な対策となります。

参照RFC: RFC958, 1059, 1119, 1305

→ 動作環境

本ソフトウェア部品の動作については、対象機器がRTC(内蔵時計)を搭載していて、ソフトウェアから日付・時刻設定が行えることが条件となります。

NTP サーバは本製品には含まれていませんので、別途利用可能にしておく必要があります。

一般に公開されているNTP サーバも存在しますが、放送局の時報音声、電波時計(通信総合研究所内 <http://jyy.crl.go.jp/>)、GPS(位置方位測定システム)の受信電波に含まれる時刻情報を利用する製品を導入するなどして、独自のNTP サーバを用意することができます。

基本的に、BSD TCP/IP ソケット・インターフェイスが利用できれば、動作・移植することは可能です。

多くの場合、日付設定API 部分の変更で対応できます。

現在、サポートしているOS としてはeCos[®]、 μ ITRON があります。

その他のOS への対応については、弊社までお気軽にお問い合わせください。

なお、オークのソフトウェア部品には、全て、製品ソースコードが付属します。

→ 制御用インターフェイス

コマンドライン・インターフェイス(※1)とプログラミング・インターフェイスの両方から、同期操作を開始することが可能です。

どちらの場合も、NTP サーバのIP アドレスを引数として起動します。設定インターフェイスを用意するか、予め用意したNTP サーバに固定して利用することになります。

対象機器が要求する精度に応じて、周期(実行頻度)や使用するNTP サーバの種類(stratum)を決定します。

実際には、タイマーなどを利用して、定期的に同期操作を実行する必要があります。

※1 : ソフトウェア部品「高機能シェル・インターフェイス」を使用

→ 関連製品

● 高機能シェル・インターフェイス

BSDcsh と似たスクリプト実行機能を持つコマンドライン・インターフェイスです。

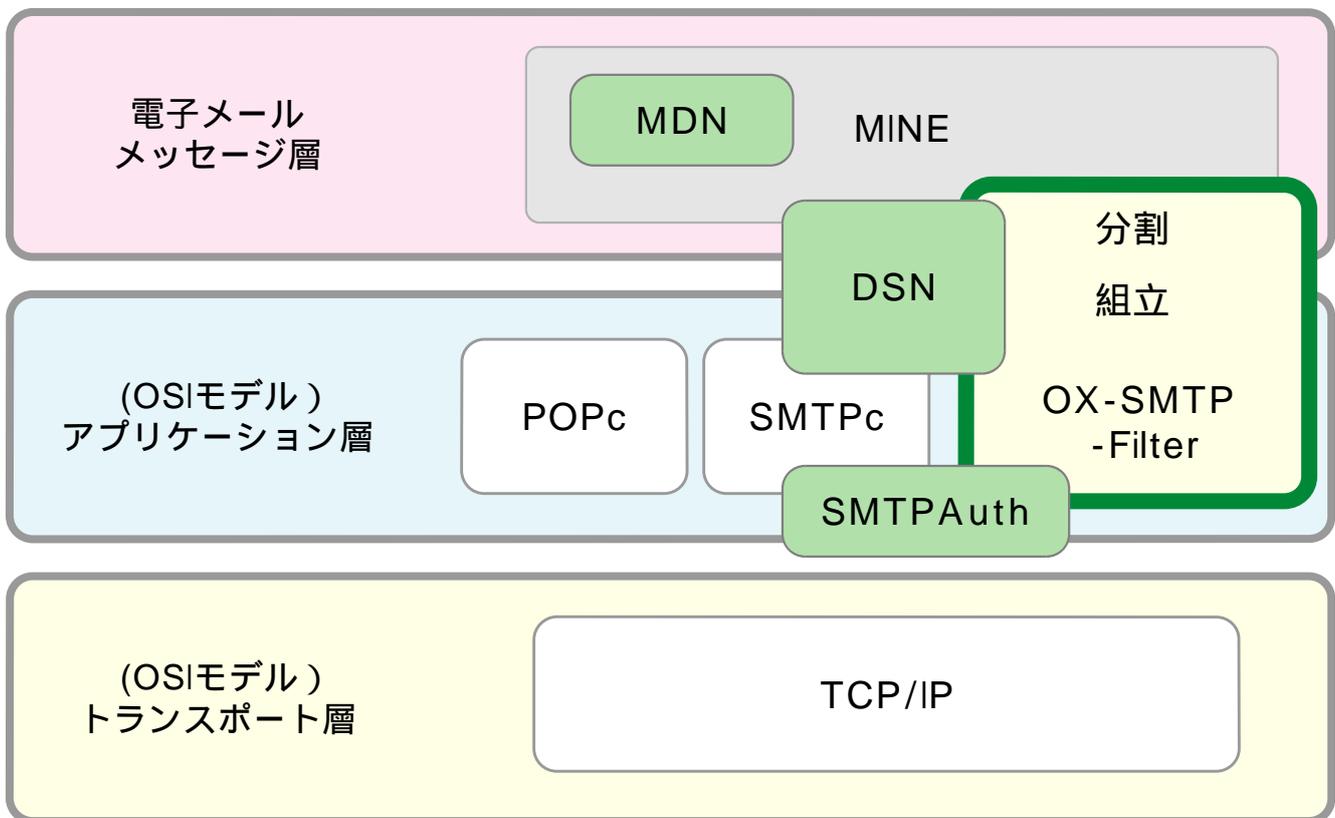
電源投入時の初期設定、組み込み用各種サーバ制御等に御利用いただけます。

各製品は独立した製品となっております。詳細については、お気軽にお問い合わせください。

※ eCos[®]は米国RedHat, Inc の米国及び、各国における登録商標です。

OX-SMTP-Filter™

ISMTTPクライアント、POPクライアント等の電子メール機能を実装するMFP(Multi Function Peripheral)に本SMTPフィルタ機能を追加実装することにより、第三者のSMTPサーバー機能を介さずに直接印刷文書を送信する機能をMFPに持たせることを可能といたします。ハードディスクは不要です。



→ MFP EMAIL Module 階層図

CPU、OS、TCP/IP非依存、100%Cソースコードでのご提供

Windows®用サンプルEXE 提供可能

以下のRFCに準拠

RFC2821 Simple Mail Transfer Protocol

RFC2822 Internet Message Format

RFC3461 SMTP Service Extension for Delivery Status Notifications (DSNs)

RFC2045 MIME Part One: Format of Internet Message Bodies

RFC2046 MIME Part Two: Media Types

RFC2047 MIME Part Three: Message Header Extensions for Non-ASCII Text

RFC2554 SMTP Service Extension for Authentication

→ OX-SMTP-Filterのご提供機能

認証機能 (SMTP認証) LOGIN、PLAIN、CRAM-MD5、DIGEST-MD5

ハードディスクは不要で、受信パケットより順次添付ファイルを抽出。

1セッションのみ処理可能。

受信パケットより順次添付ファイルをデコードしてユーザにコールバック。

(但し、Base64、Quoted-Printableのみ)

文字セット変換機能 (ISO2022JP SJIS/EUC/UNICODE / UTF8 UNICODE)

ハードディスク付きサーバ版と同時使用可能。切替関数用意。

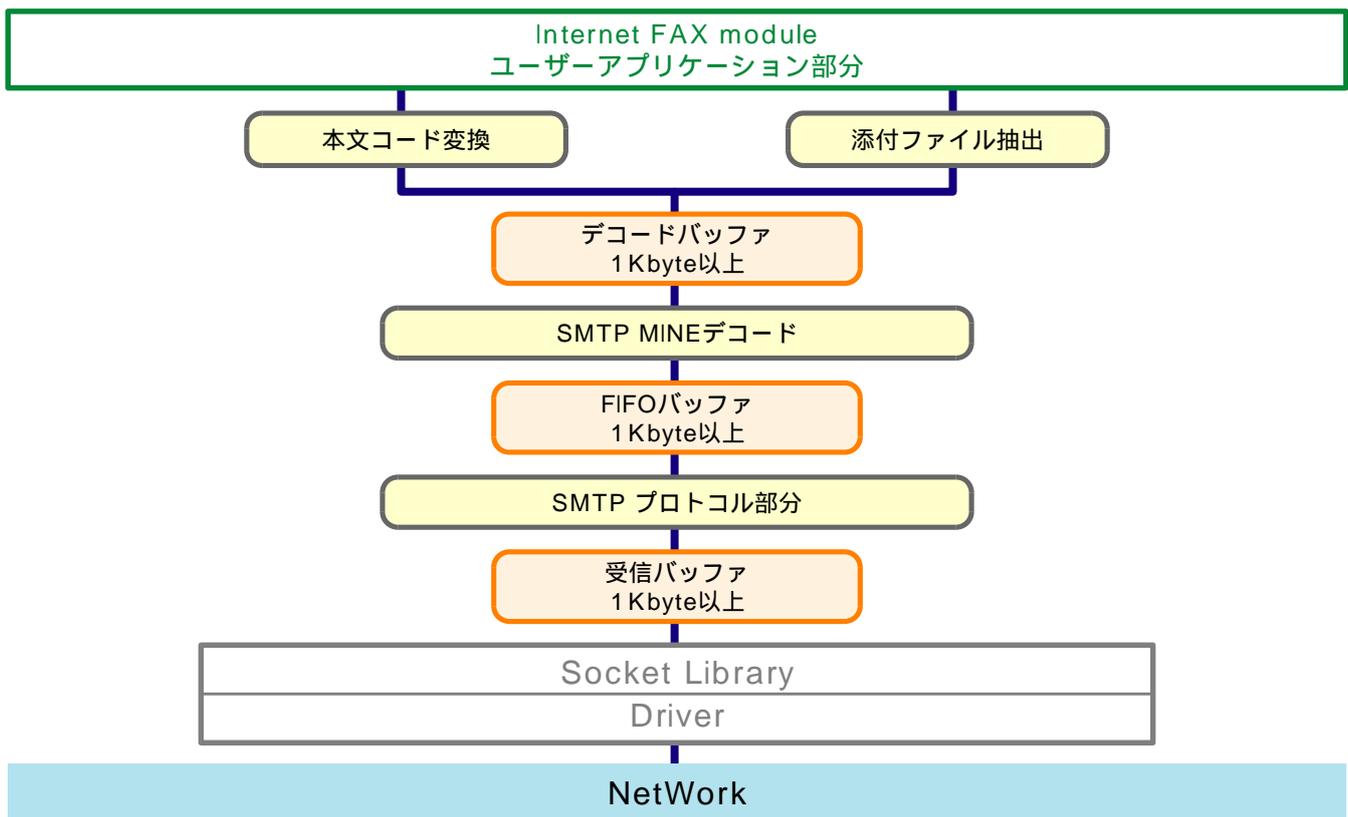
DSN応答機能

* IFAXフルモード (能力通知) を実装可能

* 上記①～⑦の機能をOX-SMTP-serverの標準機能としてご提供いたします。

* 尚、 につきましては、客先仕様への個別対応のためのカスタム対応となります。

→ SMTPフィルタ システム構成図

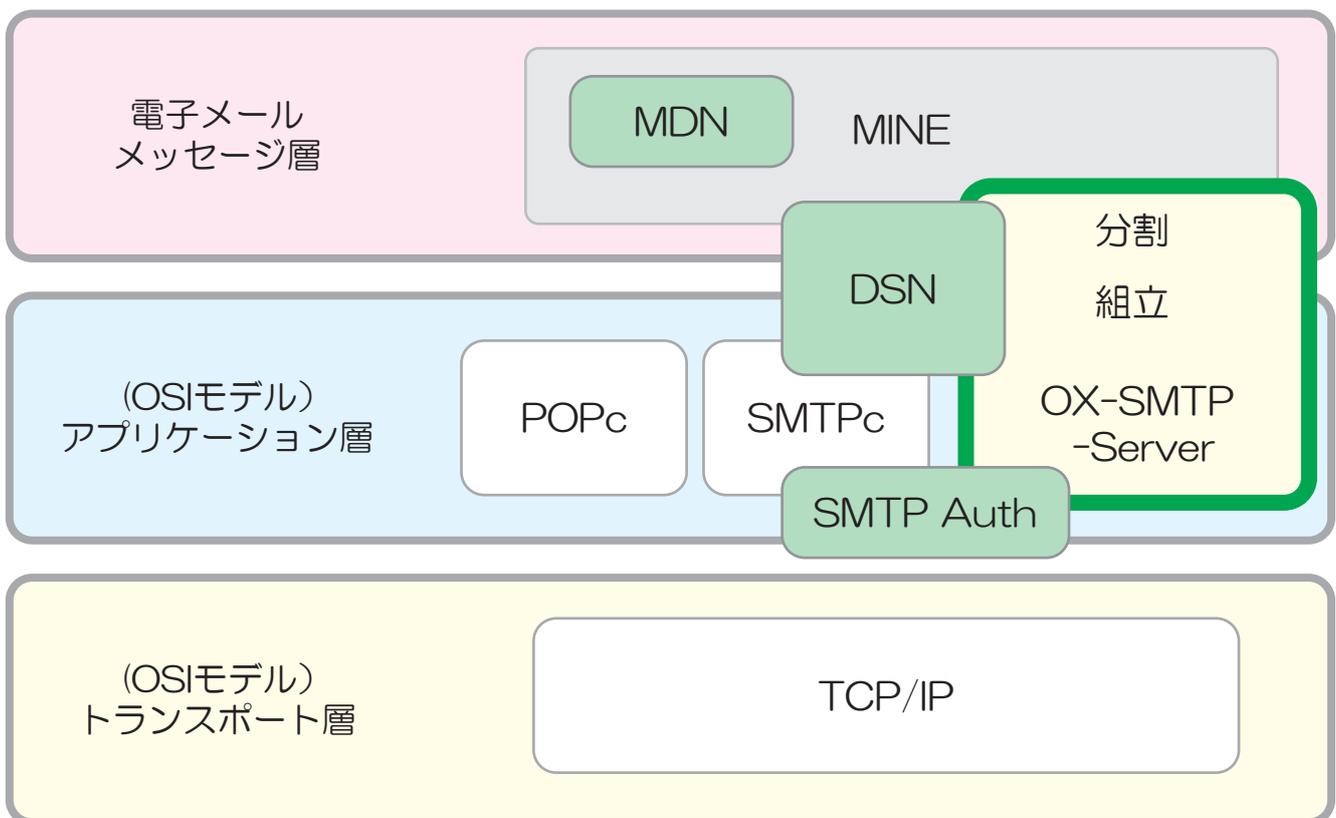


ネットワークより受信したメール本文はHDDへ保存することなく、順次アプリケーションにて処理できます。
HDDを持つ機種へ対応する『SMTPサーバ』も別途ご提供可能です。

Windows®は米国Microsoft Corp.の米国、及び各国における登録商標です。

OX-SMTP-Server™

SMTPクライアント、POPクライアント等の電子メール機能を実装するMFP(Multi Function Peripheral)に本SMTPサーバー機能を追加実装することにより、第三者のSMTPサーバー機能を介さずに直接印刷文書を送信する機能をMFPに持たせることを可能といたします。



→ MFP EMAIL Module 階層図

- CPU、OS、TCP/IP非依存、100%Cソースコードでのご提供
- Windows®用サンプルEXE 提供可能
- 以下のRFCに準拠
 - RFC2821 Simple Mail Transfer Protocol
 - RFC2822 Internet Message Format
 - RFC3461 SMTP Service Extension for Delivery Status Notifications (DSNs)
 - RFC2045 MIME Part One: Format of Internet Message Bodies
 - RFC2046 MIME Part Two: Media Types
 - RFC2047 MIME Part Three: Message Header Extensions for Non-ASCII Text
 - RFC2554 SMTP Service Extension for Authentication

→ OX-SMTP-Serverのご提供機能

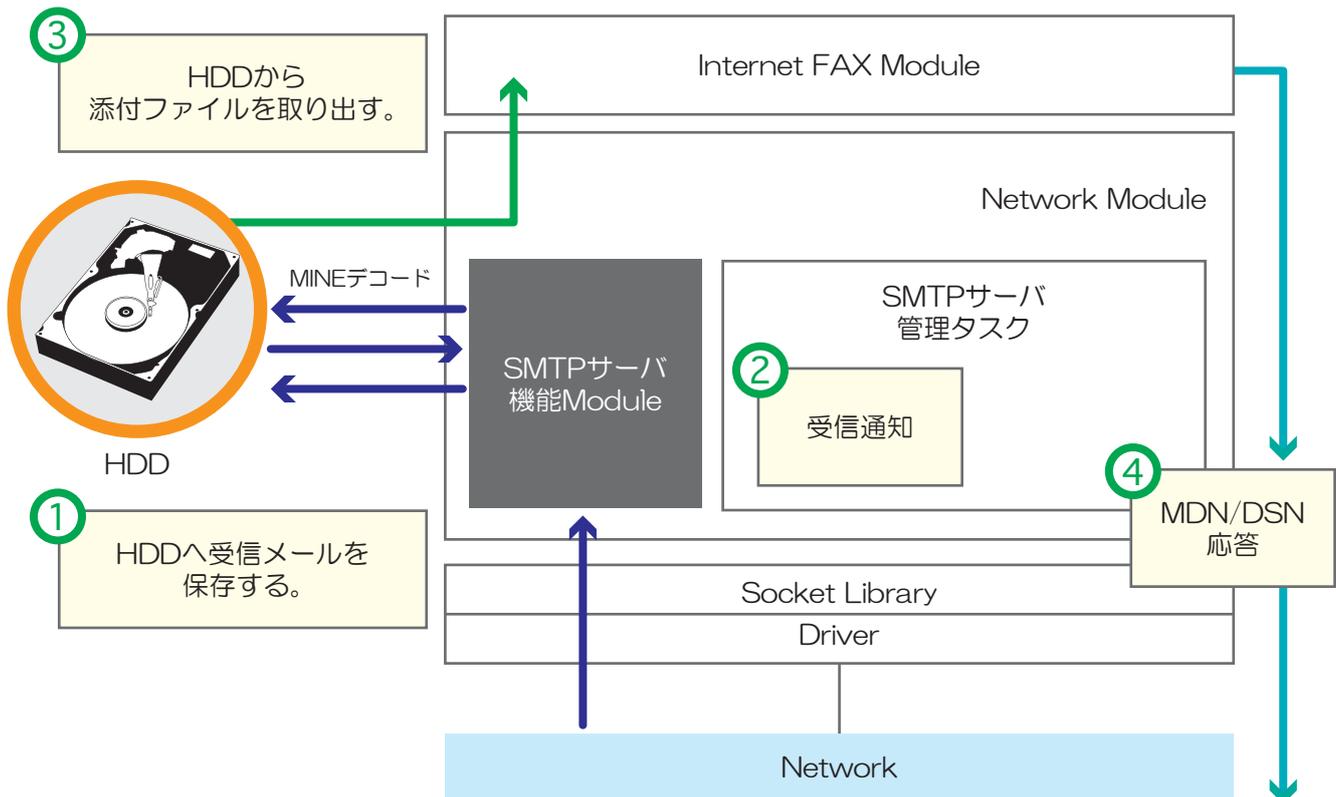
- ①認証機能(SMTP認証) LOGIN, PLAIN, CRAM-MD5, DIGEST-MD5
- ②分割メール受信機能
- ③分割メール結合機能
- ④添付ファイルデコード機能
(但し、Base64、Quoted-Printableのみ)
- ⑤文字セット変換機能(ISO2022JP → SJIS/EUC/UNICODE / UTF8 → UNICODE)
- ⑥受信メールのスプール機能(ハードディスクヘスプール)
- ⑦DSN応答機能

* IFAXフルモード(能力通知)を実装可能

* 上記①～⑦の機能をOX-SMTP-serverの標準機能としてご提供いたします。

* 尚、⑥、⑦につきましては、客先仕様への個別対応のためのカスタム対応となります。

→ SMTPサーバー システム構成図



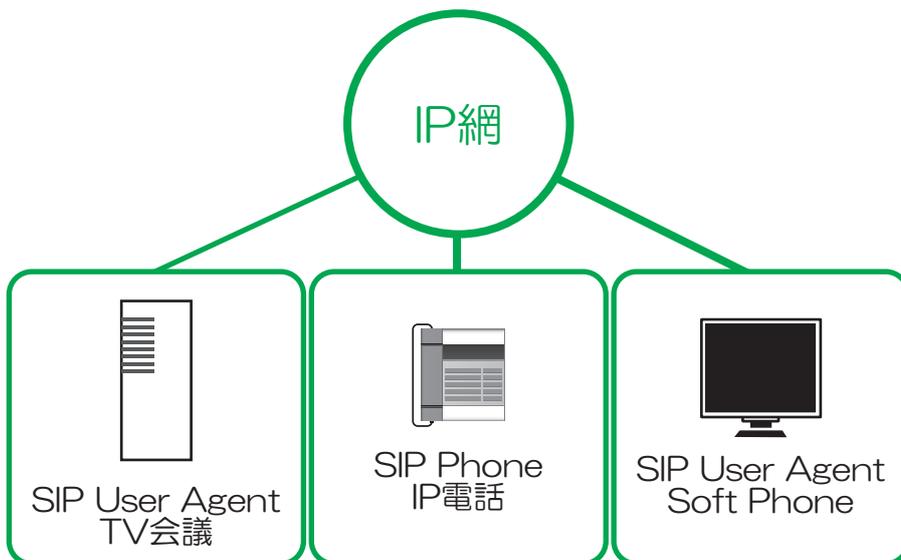
ネットワークより受信したメール本文はHDDへ一度保存する構成を取りますが、HDDを持たない機種へ対応する『SMTPフィルター』も別途ご提供可能です。

※ Windows®は米国Microsoft Corp.の米国、及び各国における登録商標です。

SIP/SDP SDK

依存API: BSD TCP/IPソケット

→ SIP - 通信セッション制御管理プロトコル



SIP の提供する主たる機能は、通信相手の解決や、接続判定、セッションパラメータの決定、セッションの開始や管理です。
 OUK SIP/SDP SDK は、ユーザの操作に近いAPIを提供することにより、VoIPクライアントアプリケーションの開発を容易にします。

→ 主な特徴

- CPU に非依存なコード。
- TCP・UDP/IP のI/F があればあらゆるOSに移植可能。
- 100KB 以下のコンパクトサイズ。
- VoIP のサンプルコード付属

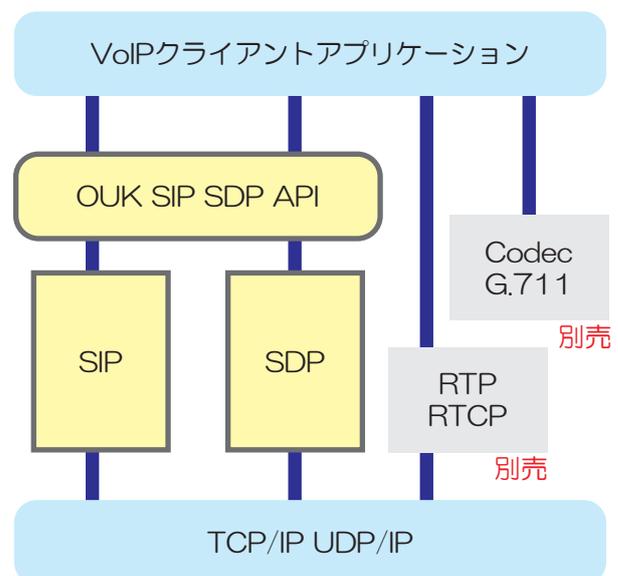
→ 応用例

- SIP 電話端末
- SIP クライアントソフト。(ソフトフォン)
- インターネットTV 会議システム
- SIP ゲートウェイ

→ 参照規格

- RFC3261: SIP: Session Initiation Protocol
- RFC2327: SDP: Session Description Protocol

VoIP アプリケーションにおけるSIP/SDP SDK の位置(黄色部分)



→ 製品詳細

● 製品内容

OUK SIP/SDP User Agent SDK ライブラリ、およびソースコード
VoIP クライアント サンプルソースコード

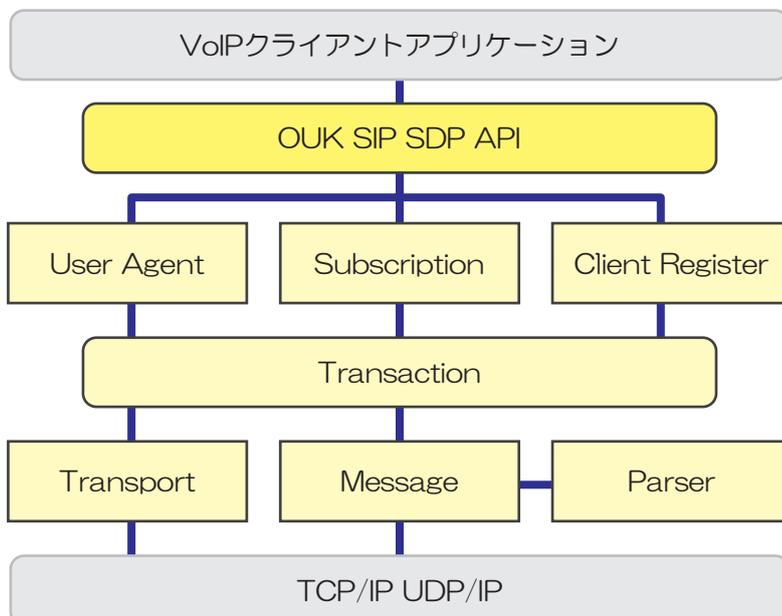
● 標準機能

INVITE セッション参加リクエスト
ACK 最終レスポンスの確認, セッション確立
CANCEL 進行中セッションの中断
BYE セッションの終了
REGISTER ユーザURI の登録

● オプション機能

OPTIONS オプション機能能力問い合わせ
INFO セッション内情報転送
PRACK 暫定的レスポンスに対する確認リクエスト
UPDATE SDPによるメディアネゴシエーション更新
REFER 別のURI への呼転送
SUBSCRIBE イベント通知要請
NOTIFY 要請されているイベント通知伝送
MESSAGE メッセージボディを使ったIM 伝送操作をSDK で記述することが可能です。

● 動作概要



OUK SIP/SDP SDK の基本APIは、発呼・応答・切断といったアプリケーションレベルの処理をI/F としていますので、SIP の複雑なトランザクションの知識を必要とせずにアプリケーションを作成することができます。

オプション機能により、VoIP だけでなくインスタントメッセージやプレゼンスの機能を実装することが可能です。

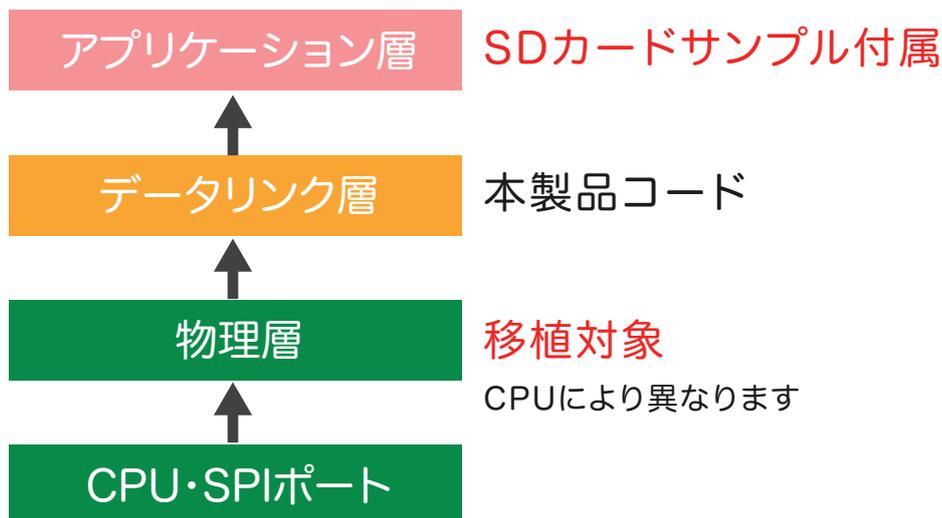
SPIドライバ

SPI - シリアル・ペリフェラル・インターフェース

組み込み用CPUが比較的低速なデバイスと接続するのに利用するシリアル通信のドライバです。

SPIを使う主なデバイスは、SDカード、EEPROM、A/Dコンバータ、D/Aコンバータ、リアルタイム・クロックなどです。

→ 階層構造



SPIデバイスのCPU側ファームウェアは大きく分けて3つの階層になり、物理層が実際のシリアル信号をハードウェアと送受信します。データリンク層はSPIマスターとスレーブの通信を司り、アプリケーション層は実際の接続デバイスに依存したコードとなります。

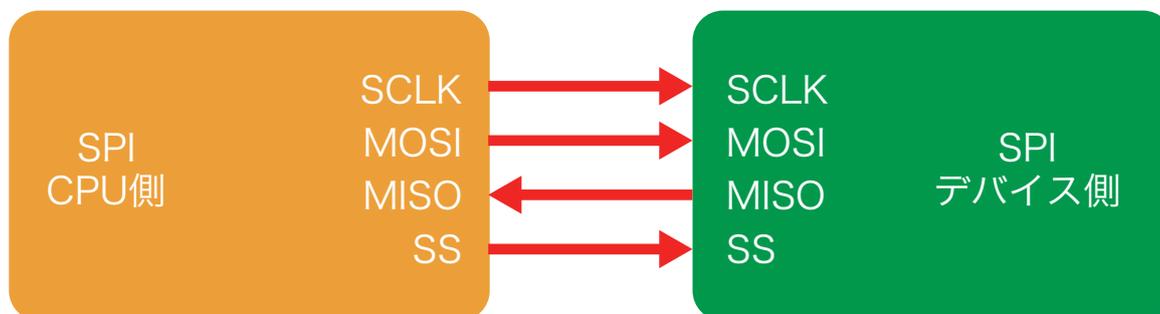
本製品では例としてSDカードのコードが付属します。

→ 主な特徴

- ・ 物理層はCPUチップに依存しますが、移植容易なコード。
- ・ 割り込み起動で動作するのでOSが無くても移植可能。
- ・ 10KB以下のコンパクトサイズ。
- ・ SDカード読み書きするサンプルコード付属

→ 製品詳細

物理層機能



SPIは上図のように、4本の信号線で通信を実施します。最近のほとんどの組み込みCPUはSPI制御ポートを備えているので、SPIの物理層は使用するCPUにより大きく異なります。この部分はCPUに合せた移植対象となります。

本製品は上図のCPU側のコードです。

データリンク層機能

CPU側からのデータ送信，受信機能を実現します。

アプリケーション層機能

実際のデバイスに合せた通信を行います。本製品は例としてSDカードの読み書きプログラムが付属します。

製品内容

- ・ SPI 物理層（移植対象）
- ・ SPI データリンク層ファームウェアソースコード
- ・ SPI アプリケーション層サンプルソースコード

OUK

株式会社オーク
〒540-0029
大阪市中央区本町橋2-23 第7松屋ビル8F
TEL.06 (6945) 4704/FAX.06 (6945) 4705
WEB.<http://www.ouk.jp/>
技術本部営業課 滝井<takii@ouk.jp>

御用命は弊社販売代理店まで

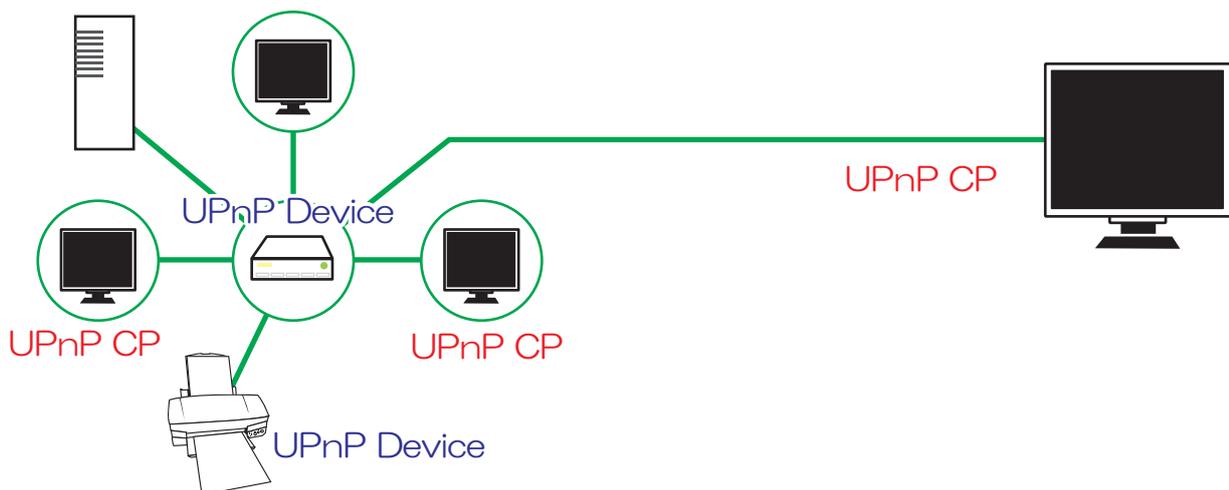
UPnP CP SDK

依存API: BSD TCP/IPソケット

→ UPnP - LAN の汎用的なプラグ&プレイ仕様

OUK UPnP Control Point SDK は、LAN で接続されたルータなどの各種UPnP デバイスの探索と制御を行うためのミドルウェアです。

UPnP IGD(Internet Gateway Device)は多くのネットワーク機器でサポートされており、プライベート・ネットワークとグローバル・ネット間の通信を可能にするNAT トラバーサル自動化や、IP アドレスの自動取得などに利用します。



→ 主な特徴

CPU に非依存なコード。

TCP/IP のI/F があればあらゆるOS に移植可能。

50KB 以下のコンパクトサイズ。

アドレッシング、ディスカバリ、ディスクリプション、コントロール、イベントング機能サポート
SSDP、GENA、SOAP、XMLDOM のソースコードを含む。

→ 応用例

プリンタやFAX のWindows 側ポートドライバ

インターネットカメラ

インターネット付きHDD レコーダやテレビ

→ 製品詳細

● 製品内容

OUK UPnP Control Point SDK ライブラリ、およびソースコード
 OUK-DOM ライブラリおよびソースコード

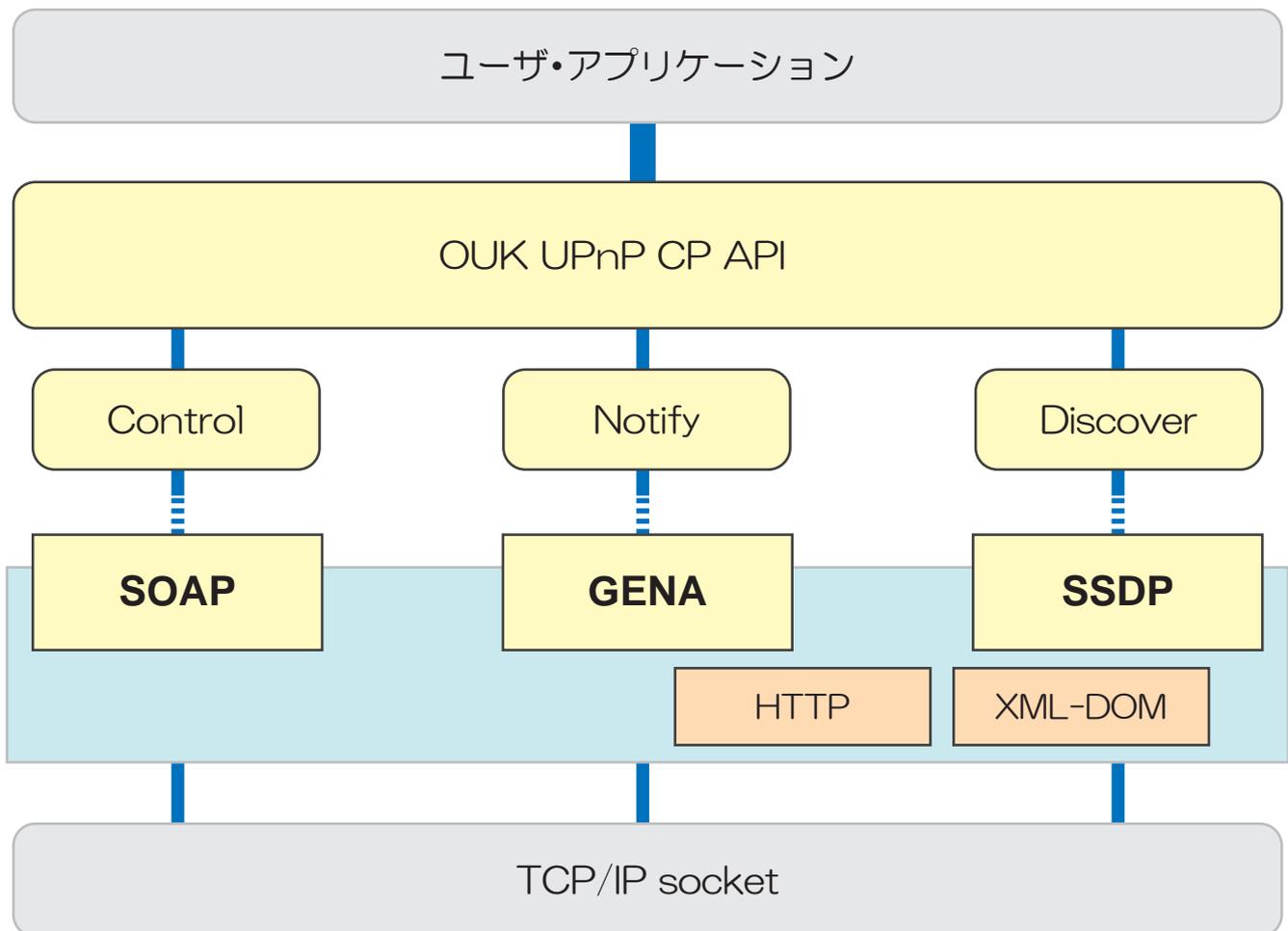
● 機能

UPnP Control Point 機能全般を実現 Device Architecture Ver1.0 準拠
 XML-DOM Level2 準拠

● 動作概要

UPnP CP API は面倒なHTTP やXML の解釈をカプセル化し、単純なAPI でControlPointの機能を実現します。送受信するXML 文書はDOMドキュメントオブジェクトで汎用的に処理するので、あらゆるデバイスの操作をSDK で記述することが可能です。

SSDP で検索したデバイスはコールバック関数により非同期に通知され、ネットワーク内にある複数のデバイスを一元管理することができます。



※ Windows®、は米国Microsoft Corp.の米国、及び各国における登録商標です。

高速プログラムへの最適化

プログラム高速化で時間短縮、作業効率大幅アップ

CPUを含むハードウェアの性能は近年、飛躍的に進歩を遂げており、それに伴い新しい技術も多く生まれています。プログラムの高速化を図るにはハードウェアの交換が効果的ですが、ハードウェアを買い換えるよりも、プログラムの高速化をした方が効果が高い場合もあります。プログラムを見直し高速化を図ることで、生産性・パフォーマンスの大幅な向上やコストの削減を実現することが可能です。

プログラムの高速化には様々なレベルがありますが、弊社ではお客様のご希望により各レベル全てのご要望にお応えすることが可能です。ご予算の都合や納期など様々な制約がある中であっても、最も効果的な方法で高速化を行います。

レベル0 調査

各種プロファイラーの利用により、プログラムのどの部分に一番時間がかかっているかを解析します。特殊なOSでプロファイラーが利用できないときでも、コードを埋め込んだりロジックトレサを使って測定いたします。

レベル1 コードの見直し

サブルーチン単位でプログラム・コードを見直します。

コンパイラの最適化

コンパイラとの整合を取り直して、高速化を計ります。場合によっては、コンパイル手法を変更するだけで10%以上の改善が見込めます。

forループの最適化

ループを整理し、最適なステップ数で実行される様書き直します。

メモリの読み書き最適化

メモリの読み書きアドレスをなるべくキャッシュに載るような形で考え直します。データキャッシュの整合性をコード側で取るにより高速化を計ります。

CPU/SSE命令の利用

Pentium III以降採用されているマルチメディア命令セットや、MMXの後継に当たる浮動小数点演算平行処理機能、3D命令セットなどを積極的に利用し、部分的に高速化を計ります。

GPUの利用

グラフィック・チップに装備されているGPU命令を積極的に使うようにコードを修正します。以下のI/Fを選ぶことが出来ます。

| | |
|-------------------|--|
| OpenCL | 主としてIntel第3世代Coreプロセッサ (IvyBridge) 以降のCPUの内蔵のグラフィックコントローラのGPUが使用でき、外部GPUボードを使わずに高速化する事が出来ます。少し冗長的ですが汎用的なI/Fもあり、GPUが変わっても使えるように記述することもできます。 |
| CUDA | NVIDIA専用の命令セットで、ベクトル・行列演算、FFTが高速で実行できます。GeForce8以上が必要です。 |
| ATI Stream | AMD専用の命令セットでCPUとGPUとの処理能力の融合により高速化が望めます。CUDAとどちらが早いかは目的の機能によります。 |

レベル2 アルゴリズムの見直し

仕様を整理し、設計レベルからの見直しを行います。

制御ループの最適化

設計変更により、制御ループをより簡単にし、高速化を計ります。

計算ロジックの見直し

計算方法そのものよりも、全体の動きを見直した結果として、計算ロジックをより高速的な手法に置き換えます。その後、GPUの利用などして、高速化を計ります。

レベル3 OSとの関係見直し

OSが存在する場合、OSとの関係を見直し、CPU動作レベルを変更します。それにより、OSとのオーバーヘッドを少なくします。例えば、入出力ルーチンをドライバ化したり、フィルタードライバを積み重ねることにタスク切替時間を最適化できます。

レベル4 ハードウェアの追加

ソフトウェアだけで満足できない場合は、ハードウェアの追加をします。

CPUの増設

並列化できる処理であれば、CPUを増やせば、CPU個数倍の速さになります。

この場合、OpenMPを利用したマルチスレッド化や、独自のマルチプレセス化を設計することもできます。

専用ハードウェアの追加

時間がかかる部分をそのままハードウェアで置き換えます。ALTERAやXilinxの試作ボードと、そのドライバソフトを全て納品することが可能です。

ハードウェアの記述言語はVHDLやAHDLやVerilog、システムCなどから選んでいただけます。

最終的にASICにすることも可能です。

プログラムをお預かりする前に秘密保持契約を結ばさせていただきますのでご安心ください。
お預かりしたプログラムは厳重に管理し、受託内容に関して秘密厳守で対応いたします。