

# DS80C400マイクロ コントローラによる ダイヤルアップネットワーク接続

DS80C400ネットワーク用マイクロコントローラを使うと、ネットワーク経由でデータを集め、分析や調整作業を行うことが簡単にできます。

技術の進歩に伴い大規模ネットワークが利用できるようになった結果、センサやアクチュエータをマイクロコントローラで簡単に制御・監視できるようになっています。取得したデータをネットワーク経由で中央局に集め、分析や調整作業ができるようになったのです。そのようなアプリケーションにとって、DS80C400ネットワーク用マイクロコントローラはとても便利なソリューションです。さまざまな周辺機器が利用できるだけでなく、DS80C400シリコンソフトウェアはTCP/IPスタックも実装されています<sup>1</sup>。Java™仮想マシン(JVM)を含んだTiny InterNet Interfaces(TINI®)プラットフォーム<sup>2</sup>は、IPネットワークの広範囲なサポートを提供します。DS80C400自体がイーサネットインタフェースを実装していますが、TINIランタイム環境(TRE)からもPPP(ポイントトゥポイントプロトコル)を使ったダイヤルアップ接続が行えます。PPPには、公衆通信ネットワークとIPソフトウェアインフラを活用し、モデムで2地点間の通信が行えるという大きな魅力があります。つまり、イーサネットはないけれども電話交換回線ならあるという辺境地でも、リモート組込ネットワークアプリケーションを利用できます(図1)。

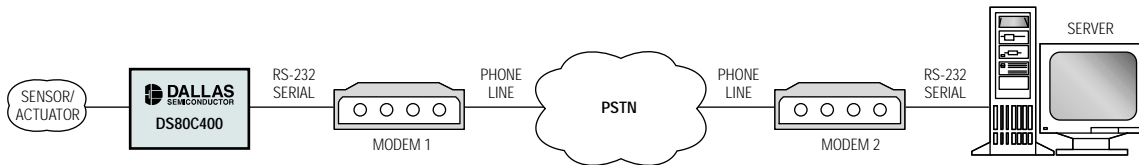


図1. TINIランタイム環境で走るリモートDS80C400がサーバにダイヤルアップし、データを転送。

## PPPの概要

PPPは汎用プロトコルで、シリアルやパラレル、イーサネット、そしてGPRS(汎用パケット無線サービス)装置のような携帯電話など、さまざまな物理媒体を通じてデータを伝送することができます。このプロトコルがダイヤルアップ接続で使われることが多い理由は、構成とセットアップが簡単だからです。物理媒体に要求されるのは、全二重の機能だけです。通信は、同期・非同期のいずれでも行うことができます。

PPPは、主に以下の3つのコンポーネントで構成されます。

- 1) 同じリンク上でマルチプロトコル・データグラムをカプセル化するメソッド。PPPでは、HDLC(ハイ・レベルデータリンク制御)フォーマットでカプセル化が行われます。利用可能な帯域幅が制限される場合には、カプセル化フィールドの一部が圧縮されます。
- 2) コネクション確立やリンクオプション設定、エラー検出、リンク切断を行うLCP(リンク制御プロトコル)。
- 3) ネットワークレイヤプロトコルを確立及び設定するNCP(ネットワーク制御プロトコル)。

## 目次

DS80C400マイクロ  
コントローラによる  
ダイヤルアップ  
ネットワーク接続.....1

セキュリティ確保に  
最適なSRAMを使った  
マイクロコントローラ.....8

DS80C400による  
ネットワーク型  
マルチメディア  
アプリケーションの  
構築.....12

差し替えるだけで  
DS5002をアップ  
グレードする  
DS5240/DS5250.....18

PPPは、シリアルや  
 パラレル、イーサネット、  
 及びGPRS(汎用パケット  
 無線サービス)デバイスと  
 しての携帯電話など、  
 さまざまな物理媒体を  
 通じてデータを伝送する  
 ことができます。

## PPP動作

インターネットの規格、RFC 1661は、PPP動作をポイント・トゥ・ポイントが構成、維持、及び終了されるにしたがって、異なるステージを経由するステートマシンとして説明されています。図2に、PPPが、デッド、確立、認証、ネットワーク、終了の5つの明確なフェーズに分かれることを示す簡易的な状態図です。PPPは、認証を除く4つのフェーズを実装しています。

**リンクデッドフェーズ**：リンク動作の最初と最後にあたるフェーズです。物理層がパケット伝送の準備ができていない状態です。物理層の準備が整うと、UPイベントが発生し、PPPはリンク確立フェーズへと移行します。

**リンク確立フェーズ**：物理層が動作状態で、LCP構成パケットの交換を通じて通信オプションのネゴシエーションが行われます。この段階で構成されるのは、ネットワークレイヤプロトコルに依存しないオプションのみです。Configure-ACKパケットが送出され、受信されると、PPPはOPENEDイベントを生成し、次のフェーズへと進みます。

**認証フェーズ**：ピアの認証を行うフェーズで、使用されない場合もあります。ダイヤルアップなどのリンクでは、ネットワークレイヤプロトコルパケットを交換する前にリンクによる認証を行うことが推奨されます。そのためには、リンク確立フェーズにおいて認証要求が送出される必要があります。

**ネットワーク層プロトコルフェーズ**：リンクが確立され、認証が行われると、サポートされているネットワーク層に特定したNCPパケットの交換によってネットワーク層プロトコルの構成を行います。ネットワーク層プロトコルごとに独自のNCPが設定されており、個別にネゴシエーションを行う必要があります。

**リンク終了フェーズ**：キャリア消失や認証失敗、リンク品質の劣化、運営管理の理由でPPPリンクが終了したとき、CLOSEイベントが発生します。ピア間では、LCP終了パケットが交換されます。ネットワーク層プロトコルにはリンク終了が通知され、各プロトコルで必要なアクションが取られます。終了ACKを受信するか、タイムアウトが発生すると、物理層はディセーブルされます。そしてDOWNイベントが発生します。PPPはリンクデッドフェーズに戻ります。

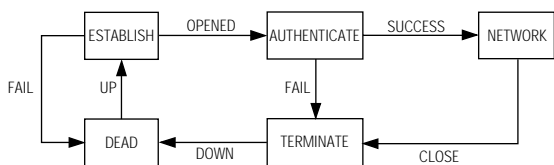


図2. RFC 1661で規定されたPPP構成の状態図

## TINI PPP

TINIでは、RFC 1661をPPP構成の枠組みとしています。PPPは、シリアルリンク経由でIPデータグラムを送るメカニズムを完全に満たします。本来のネットワーク階層では、PPPはIPモジュールよりも下位、シリアルポートドライバよりも上位に位置します。プログラムが複雑になるのを軽減するためにPPPフェーズはさらに単純化されています(図3)。

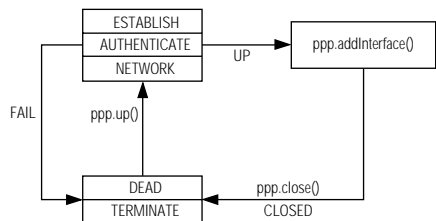


図3. TINIにおけるPPPの構成方法を示すTINI PPP状態図

com.dalsemi.tininet.pppパッケージでは、アプリケーション開発者向けに、JavaクラスとしてPPPが説明されています。PPP状態はイベント駆動型です。ppp.up()によってリンクが確立され、認証が行われ、ネットワークプロトコルが設定されます。PAP(パスワード認証プロトコル)とCHAP(チャレンジハンドシェイク認証プロトコル)もサポートされています<sup>3</sup>。一度リンクが構成されたら、UPイベントが発生し、ネットワーク階層にPPPインタフェースが追加し、ネットワーク上のトラフィックをPPPインタフェースに流せるようになります。ppp.close()を実行すると、CLOSEイベントが発生し、リンクは停止され、デッド/終了状態に戻ります。

例1は、DS80C400に構成されたPPPClientの一部です(www.ibutton.comにアクセスして「PPP」で検索すると、最新例を見ることができます)。PPPオブジェクトが生成したら、PPPオブジェクトのPPPEventListenerとしてPPPClientがインストールされます。一連のatCommandによってPPPパラメータがセットされ、リンクが開始されます。一度リンクが確立されたらppp.up()が呼ばれ、ネットワークトラフィックでPPPが使えることがネットワーク階層に通知されます。リンクを終了する時は、ppp.close()を呼びます。

## 例1. PPPClientの構成例

```
public class PPPClient extends Thread
    implements PPPEventListener, CommPortOwnershipListener{
    ...
    public void run(){
        ppp = new PPP();
        openSerialPort(portNumber);
        // Add this object as a PPP event listener
        ppp.addEventListener(this);
        // Set the local and remote IP address
        ppp.setLocalAddress(localAddress);
        ppp.setRemoteAddress(remoteAddress);
        // Set client peer type options
        // Set the ACCM to escape all octets
        ppp.setRemoteAccm(0x00000000);
        ppp.setLocalAccm(0x00000000);
        ppp.setAuthenticate(false, true);
        // Set username and password
        ppp.setUsername(username);
        ppp.setPassword(password);
        // Initialize modem
        for (int i = 0; i < dialSequence.length; ++i)
            atCommand(dialSequence[i]);
        // Set connected flag
        connected = true;
        // Issue up command to PPP FSM
        ppp.up(serialPort);
        // PPP connection is now established, we can now
        // communicate with remote host
        sendData();
        ppp.close();
        closeSerialPort();
    }
    ...
}
```

TINIランタイム環境では、細部を秘匿したユーザフレンドリなAPIを提供します。そのため、開発者はPPPをユーティリティとして利用しながら開発者自身の設計に集中できます。電話線のないところでも、モデムの代わりにGPRS無線電話を使うことにより、同じアプリケーションを走らせることが可能です。

例2は、PPPEventの処理例です。リンクの準備が整い、UPイベントを受信すると、ネットワークインタフェースにPPPが追加され、IPパケットをPPPインタフェースに転送することができます。CLOSEイベントを受信すると、ネットワーク階層はPPPインタフェースを外し、PPPによるネットワーク通信を終了します。

## 例2. PPPEventメソッド

```
/**
 * PPP event listener interface
 */
public void pppEvent(PPPEvent ev){

    switch (ev.getEventType()){
        case PPPEvent.UP:

            // PPP connection is up
            ppp.addInterface(interfaceName);
            interfaceActive = true;
            break;
        case PPPEvent.CLOSED:
            // PPP connection is closed
```

```

    if (interfaceActive){
        interfaceActive = false;
        ppp.removeInterface(interfaceName);
    }
    connected = false;
    break;
default:
    break;
}
}
}

```

## 遠隔湿度データロガーの例

ここに、経済的で小型コンピュータで提供されるネットワークの能力の利点すべてを利用した、強力で、ネットワーク接続されたアプリケーションを紹介します。この例では、TINIm400-030p、OEM Stamp+ EditionというTINIリファレンスデザインを使用します。このモジュールは低消費、豊富なI/O、部品点数の少ない組み込みコントローラ、及びデータ収集モジュールを提供します。TREの組み合わせでは、最小限のプログラミングでネットワーク対応の堅牢なデータ収集システムが完成します。Stamp+ モジュールでは、フラッシュROM、RAM、リアルタイムクロック、1-Wire<sup>®</sup> ネットワーク、パラレルI/O、非同期シリアルポートが含まれています。この特別なデザインはダイヤルアップによる遠隔アプリケーション用であるため、イーサネットインタフェースは使用しません。もちろん、イーサネットPHYと関連するマグネットを追加すれば、イーサネットを使うこともできます。

この例<sup>4</sup>は、データの取得と記録を行い、PPPによってPSTN(公衆通信交換網)経由で接続してダイヤルアップ接続を管理し、データをリモートサーバから利用可能とする、すべての機能を備えたアプリケーションです。ダイヤルアップのネットワーク接続のサポートにより、真の意味で遠隔データロガーを構築することができます。

アプリケーションノート  
702 : TINIポイント・  
トゥ・ポイントプロトコル  
(PPP)を使用し、PPPを  
使用するためのTINIを  
セットアップする方法を  
示します。シリアルリンク  
でIPパケット伝送が可能  
になります。  
[www.maxim-ic.com/ja/  
appnoteindex](http://www.maxim-ic.com/ja/appnoteindex)を参照。

### システム概要

アナログモデム経由でダイヤルアップ接続能力を実証するセットアップ例を図4に示します。電話線などが利用できない場合には、シリアルケーブルで接続してテストすることが可能です。

図4のテスト構成では、以下の機器を使用しています。

- PPPモジュール1台 - DataLoggerサーバを実行
- Windows<sup>®</sup> 2000パソコン1台 - DataLoggerServerソフトウェアを実行
- アナログモデム2台 - 1台はWin2K PCに、もう1台はPPPモジュールのシリアルポートに接続
- 湿度検知回路 - 記録する湿度データを収集する

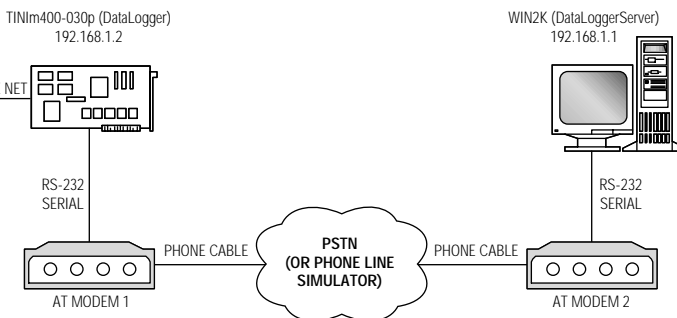


図4. 汎用モデムでデータ  
通信を行う遠隔データ  
ロギングシステムの例

Stamp+ モジュールのDS80C400には、TREがインストールされています。このTREプラットフォームでは、ASMとC、Javaプログラミングがサポートされています。組み込みファームウェアはTCP/IPスタックを実装しPPPプロトコルの枠組みを提供しています。今回の例では、TINIによってStamp+ モジュールを遠隔データロガーとして利用できるようになっていますが、「TINI」と「Stamp+ モジュール」は同義です。

PPP接続は、電話線シミュレータの両側にある2台のアナログモデムを使って確立されます。2種類の電話線が利用できる場合には、公衆電話回線を使うこともできます。PPPインタフェース

のテストを行う前に、まず、ダイヤルアップによるネットワーク接続を確立しなければなりません。接続が開始されると、以下の順番で処理が進みます。

- 1) TINIモデムからサーバ側モデムに電話をかけます。
- 2) 受信したサーバモデムが呼び出しに应答します。
- 3) PPPオプションのネゴシエーションが開始されます。
- 4) 認証情報がTINIからリモートサーバに送られます。
- 5) サーバがTINIにIPアドレスを割り当てるとともに、TINIにサーバのIPアドレスを通知します。

## ソフトウェアとハードウェアの概要

ソースコードは、[ftp://ftp.dalsemi.com/pub/tini/reference\\_designs/TINIm400-030p/DataLogger.zip](ftp://ftp.dalsemi.com/pub/tini/reference_designs/TINIm400-030p/DataLogger.zip)からダウンロードすることができます。湿度データは、図5に示すように、センサ回路によって収集されます。この例では、iButton®パッケージのDS1922H\* 1-Wire温度湿度センサを使用しました。DS80C400では、あらゆる1-Wireデバイスが利用できます。

## TINIクライアントソフトウェア

今回のTINI DataLogger例では、1-Wireネットワーク接続、シリアル通信、TCP/IPネットワーク接続という3点について説明します。まず、重要度が特に高いクラスについて、その機能の概要を紹介します。

### DataLoggerクラス

- 構成ファイル(/etc/dataLogger.properties)からパラメータを取得する
- HumidityLoggerのインスタンスを生成し、サンプルのキャプチャを行う
- PPPDaemonのインスタンスを生成し、PPP接続を管理する
- イーサネットやPPPといったネットワークインタフェースを経由し、リモートサーバに接続するアウトバウンド接続を開始する

### HumiditySensorクラス

- DS1922Hとの通信を処理するとともに、データを取得する

### HumidityLoggerクラス

- 湿度センサを起動・管理する
- 記録データをサーバに対する出力ストリームに書き出す

### PPPDaemonクラス

- ダイヤルアップクライアント
- PPPインタフェースを使ってTCP/IP接続を確立する
- 物理データリンクを管理する
- PPPイベント通知を受け取る
- 物理データリンクで発生したエラーをDataLoggerに通知する

\* 開発中の製品—入手性についてはお問い合わせください

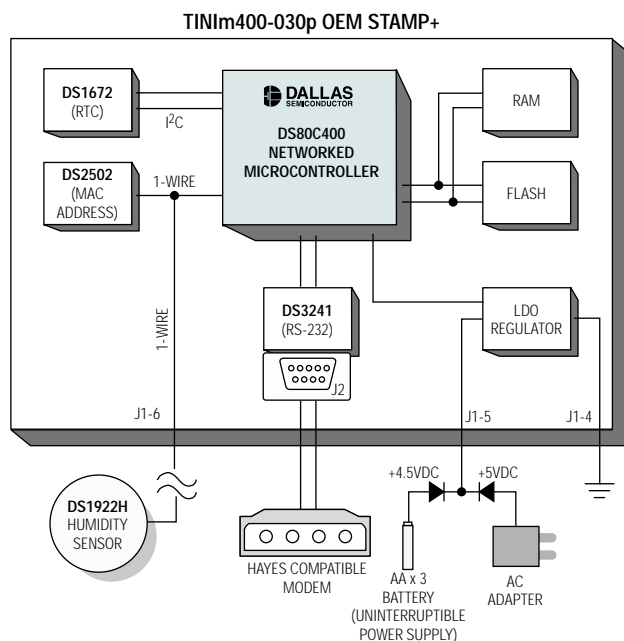


図5. TINIm400-030pデータロガーは、センサを使って湿度データを取得します。

1-Wireネットワークに関する情報は、「高信頼性1-Wireネットワークのガイドライン」(アプリケーションノート148) ([www.maxim-ic.com/ja/appnoteindex](http://www.maxim-ic.com/ja/appnoteindex))をご覧ください。

## PPPSerialLinkクラス

- PPPDataLinkインタフェースを構成する
- PPPDaemonによるデータリンクの管理を許可する
- データリンクができるようにシリアルポートを構成する

## PPPModemLinkクラス

- PPPSerialLinkのサブクラス
- モデム通信を管理する
- SerialPortEvent.CD(キャリア検出)を監視し、モデムの中断を検出する

## ModemCommandクラス

- モデムとのシリアル通信を処理する
- 意図する応答を待っているときにタイムアウトが発生した時、DataLinkExceptionを生成する

## リモートデータロギングサーバ

DataLoggerServerは、シンプルなGUIサーバアプリケーションで、TINIからの接続を受け付けるとともにTINIから最新ログをダウンロードすることができます。

## DataLoggerServerクラス

- ログデータを表示する
- acceptを受信するとブロックし、PORTにおけるソケット接続を待つ
- ログを構築するとともに、湿度と温度の経時変化をグラフにする

## アプリケーション例の実行

アプリケーションファイルは、イーサネットインタフェースを経由し、FTPプロトコルでTINIファイルシステムに転送するのが普通でした。しかしStamp+ モジュールではイーサネットインタフェースがなく、そのかわりSlushとJavaKitにymodemファイル転送プロトコルが追加されています。このymodemファイル転送プロトコルを使えば、JavaKitシリアルリンク経由でTINIファイルシステムにファイルを転送できます。Stamp+ モジュールの詳細については、"*Dial-Up Networking with the TINIm400 Stamp*"(アプリケーションノート611、英語版)をご覧ください。アプリケーションファイルとともに、以下のテキストを含むDataLogger.tini(/etc/.startup)ファイルもTINIファイルシステムに転送してください。

```
#
# Starting DataLogger application from .startup file
#
setenv FTPServer disable
setenv TelnetServer disable
setenv SerialServer disable
#
initializeNetwork
#
java /DataLogger.tini
```

このスタートアップファイルにより、シリアルサーバがディセーブルされ、データロガーアプリケーションがシリアルポートにアクセスできるようになります。アプリケーションファイルとスタートアップファイルが転送されると、TINIをリセットし、新しいスタートアップファイルを適用するとともに、データロガーアプリケーションが起動されます。

