

X p o r t を応用した遠隔制御及び 遠隔監視の基本システムの構築

金野茂男

1. はじめに

インターネットを利用しようとするれば、普通はパソコン（最近のパソコンでは標準装備としてインターネット通信用 I / O デバイスが内部に組み込まれている。かつてはオプションとして LAN カードを購入し、自分でパソコン本体に装着していたものである）を購入し、インターネット通信のネット管理業者であるプロバイダと契約し、自分のパソコンに割り当てられた、いわゆる電話番号に相当する IP アドレスを獲得してからインターネットに接続する。あるいは機関内ならば、購入したパソコンを機関内に張り巡らされている LAN 端末に接続すれば、簡単にインターネットの利用にはいることができる。

このように、現在ではパソコン間のインターネット通信が主であるが、パソコン以外の機器でもネットに接続でき、その機器を遠隔で制御あるいは監視ができるシステムは有益であることは疑いがない。そのようなシステムは既に存在している。パソコンの I / O ポートに必要な I / O ボードを装着し、そのボードと機器を接続する方法である。機器の例としては家庭内のクーラーがある。このシステムを利用すれば、勤務中の職場からインターネットで自宅のパソコンに接続し、そのパソコンを経由して機器を制御することができるようになる。帰宅前にエアコンの電源を入れておくことができる。その他、様々な応用が提案できよう。

インターネットはパソコンで利用できるものであるという固定概念を覆す機器が現れた。このインターネット分野に応用できる「小型ネットワークアダプタ」とも呼べるユニットである⁽¹⁾。ラントロニクス社の商品名「X p o r t」である。詳細は文献(1)が詳しいのでそれに譲る。このユニットはイーサネットの接続端子 RJ - 4 5 ソケットの形状をしており、大きさは親指の半分ほどの長さである。外見からは単なるインターネット接続用の接続ソケットとしかみれない。が、実のところこのソケットにはイーサネット通信に必要な通常の接続端子以外に、この内部には、CPU, RAM, ROM が内蔵されている。機能デバイスのマクロ化の進歩には目を見張るものがある事例であろう。外見はソケットながら、これ単体でインターネットの「サーバ」としての動作も保証されている小粒ながらの優れものである。このユニットには「イーサネット信号 シリアル信号」変換機能、汎用 I / O ポートも装備されている。

本研究室では昨年度、文献(1)なども参考にしながら、卒業研究課題として「イーサネット・シリアル変換器を応用した家庭内サーバの構築」に取り組んだ⁽²⁾。この課題は本研究室の卒業研究生(5年生)の課題であった。著者自身も X p o r t を利用した遠隔制御・遠隔監視システムの構築に興味を持っていた。

学生の卒業論文を参考にして、システムの再現及び構築に取りかかった。取り組んでいる最中に、厄介な暗礁に乗り上げた。参考文献(1)、(2)両方で使用されているインターネット通信用ソフトに使用している j a v a 言語である。インターネット分野では j a b a 言語が使われていることはわかっていた。今回、初歩からこの言語を勉強し始めた

が、年のせいでもなかろうが、取り扱いに手間暇がかかる上に、非常にわかりにくいのである。その上に、論理構造、単語使用が支離滅裂、複雑怪奇にも思えてきた。j a v a 言語はサンマイクロシステムズ社が無償で提供しているフリーソフトである。学生が好むはずである。

ところで、著者はマイクロソフト社の有償のアプリケーション開発ソフトであるV i s u a l B a s i cは一応使い慣れている。多くの学生に聞いたところではV i s u a l B a s i cはインターネットには使えないであろう、とのことであった。インターネットアプリケーションソフトの開発にはj a v aかC ++です、とのことであった。しかし、V i s u a l B a s i cにインターネットアプリケーションソフトの開発機能が備わっていなければ、現時点では時代遅れの言語となっているはずである。古いバージョンには備わっていなかった、そのような機能はヴァージョンが上げれば機能追加がされているはずと考え、調べてみた。その結果、V i s u a l B a s i c 6 . 0 の「P r o f e s s i o n a l E d i t i o n」に確かに、インターネット開発環境の機能が備わっていることがわかった。

V i s u a l B a s i c 5 . 0 を、かつて使用していたことがあった。「S t a n d a r t E d i t i o n」ではなく、「P r o f e s s i o n a l E d i t i o n」である。外部機器との間で、データの送受信をするためにシリアルポート(R S - 2 3 2 C規格) R Sを使用する必要がある。このシリアルポートの利用機能は「S t a n d a r t E d i t i o n」ではなく、「P r o f e s s i o n a l E d i t i o n」以上のスペックに備わっていた。従って、購入ではS t a n d a r t版ではなく、P r o f e s s i o n a l版としたわけであった。なを、当然なのかもしれないが、S t a n d a r t版の方が販売価格は安い。マニュアル本もほとんどがS t a n d a r t版対応である。

思うに、確かめてはいないが、V 6 . 0でも「S t a n d a r t E d i t i o n」にはV 5 . 0と同じようにコミュニケーション機能は備わっていないと思われる。書店などで、書棚に並んでいる「V i s u a l B a s i c」解説本のほとんどは「S t a n d a r t E d i t i o n」のものであることにも気がついた。学生のいった言葉の理解がいった。学生たちはS t a n d a r d版での知識に基づいていたのである。

以上のことから、X p o r t用のインターネットアプリケーションソフトの開発言語としては、j a v aではなく、「V i s u a l B a s i c 6 . 0 P r o f e s s i o n a l E d i t i o n」で行うこととした。やはり、書店ではほとんど「V i s u a l B a s i c S t a n d a r t版」の解説本しか見られなかったが、注意深く書店を探し回り、P r o f e s s i o n a l版も取り扱っている解説本を探した。適当なものが見つかった⁽³⁾。これを参考にして、ソフトの作成に取りかかった。

本システムの開発環境は以下の通りである。

| | |
|-------------|-----------------------------------|
| パソコンOS | マイクロソフト社W i n d o s 2 0 0 0 |
| X p o r t | ラントロニクス社 |
| ライター | マイクロチップ社純正P I C S t a r t P l u s |
| ライターソフト | マイクロチップ社純正M P L A B V . 5 . 7 0 |
| 使用ワンチップマイコン | マイクロチップ社P I C 1 8 F 2 5 2 |

なを、マイクロチップ社のMPLAB最新版であるV.7.20では、OSがWindows 2000下では、PIC18F252, 452のソースプログラムをアセンブルすると、今までに出てこなかったコンフィギュレーションビットに関する「ワーニングメッセージ」が出力表示されるようになり、更に悪いのは、ライター使用時、PICへの書き込みエラーとなり、書き込み不能となるのである。原因は全くわからない。OSがWindows ME下では、何らの問題もなかったのであるが。一応報告しておこう。従って、上記の開発環境のとおり、MPLABのスペックを落とし、V.5.70として、本システムの開発環境としている。これで、正常に仕事が行えている。

写真1に完成した「Xport - ハード部」の外観を示している。左上のRJ-45端子がXopr tである。外観はソケットそのものである。イーサネットケーブルが差し込まれている。中央にワンチップマイコンPIC18F252、右上には電源部がある。ACアダプタ(+6VDC)からPIC用+5.0V, とXport用+3.3Vを3端子レギュレータで回路に供給している。右下に制御用デモ端子である4個のLED、状態監視デモ用端子である4連のスイッチがある。2個のLEDが点灯し、スイッチの3個がオン、1個がオフとなっている。実際の動作状態を示している。詳細は後述している。



写真1 本システムの「Xport - ハード部」

左端にある2連14ピン端子はC-LCD用のものである。開発中にはハード部の動作状態をモニターするには便利な素子であるが、正常動作状態では特に必要としないので、写真では取り外している。

写真から見てとれようが、C-LCDが無いと基板には大幅な余裕がある。本システム

の再現の場合には、より小さくまとめることもできようし、他の部品も配置できよう。

なを、本システムは現時点では、学内LAN中の更に局所的なネットワーク内（研究室近傍内）での接続動作を確認している。小山高専の学内LANはいくつかのサブネットワークから構築されている。学内LAN全体の間、あるいは学内外の間での接続動作はサーバ管理者と相談し、本Xportに固定IPアドレスなどを割り振ってもらえれば容易に実現できるものと考えている。

2. 制作

Xportの応用として図1のようなシステムの構築を目指した。参考文献（2）とほぼ同じであるが、ハード面だけではなくソフト面でもより簡易にシステム全体を構築することを心がけた。文献（2）では、文献（1）と同じように、Xport内部のROMにHTMLファイルを書き込み、それを利用している。が本システムではそのような機能の使用は特に必要としていない。利用しようとするときだけシステムの使用方法を複雑にするような気がする。イーサネット・シリアル変換機能がXportの一番の特徴であると思う。今回はこれだけを使用することにし、Xportに備わっている他の機能はそのうちに必要に応じて利用していきたい。すなわち、Xportを単に、イーサネット信号とシリアル信号の変換器としてのみ使用する。

ラントロニクス社のXportには、これをイーサネットケーブルに接続（当然Xportに電源が供給されている必要があるが）すると、自動的にIPアドレスを獲得する機能が備わっているため、IPアドレスを自動割り当てとなっている機関内LAN等があるところでは、これは便利な機能である。

パソコン側から信号を送出し、イーサネットを経由して、Xportに受信させ、これをシリアルデータに変換した後、ワンチップマイコンで受け取る。ワンチップマイコンでこのデータを解析し、データに前もって予約している処理をデバイス部で行わせる。また、パソコン側から出されるデバイス側の状態報告要請により、デバイス側から、デバイスの状態情報をパソコン側に送り出し、パソコンでそれを受信し、ソフト実行画面上に、デバイスの状態を描写させることとする。つまり、双方向通信状態のシステムとする。

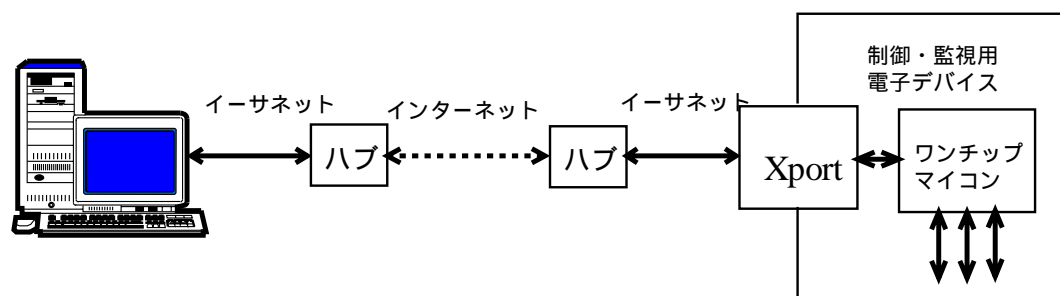


図1 予定システムの外観図

図2がX p o r tを搭載したハード部の回路図である。X p o r tの電源電圧は+ 3 . 3 V , P I Cの電源電圧は+ 5 . 0 Vなので、X p o r tのD a t a o u t端子とP I CのR X端子の間、X p o r tのD a t a i n端子とP I CのT X端子の間に論理レベル変換回路が必要となっている。

パソコン側から制御するX p o r t側の制御される部分として、4個のL E D（出力モニター）を準備する。パソコン側からの点灯、消灯命令信号により、これらのL E Dを任意に点滅させる。パソコン側からX p o r t側のデバイスを制御する基本動作試験に供するものとしている。

また、X p o r t側におけるデバイスの状態を、パソコン側から監視する基本的な監視動作試験回路として、4連のスイッチを準備する。パソコン側からのこれらスイッチの状態の問い合わせに対して、X p o r t側からその状態信号を送出し、パソコンの方で得られた状態情報をパソコンの画面上に表示し、各スイッチの状態を視認できるようにする。

P I CのBポート等に接続しているC - L C D（キャラクタ液晶表示器）はシステムの開発段階では、あった方が便利なデバイスである。がシステムが完成すると特に必要となしない。現システムでは、X p o r t側は無人状態を想定している。従って、実働時にはC - L C Dの存在は意味がない。デバイスが正常動作するようになったならば、取り外して結構である。図3にその場合の回路図を示しておく。回路はだいぶ簡単になる。その上、この方の回路にすると、P I CのI / Oポートにだいぶ余裕ができるので、多様な仕様に十分対応もできよう。

P I Cに外付けのクロックオシレータは1 0 M H zセラロックを使用している。が、P I C内部のクロック周波数は1 0 M H zではなく、4 0 M H zである。P I C 1 8シリーズにはP L L回路が内蔵され、4倍のクロック周波数を指定することができる。このP L L使用はコンフィグレーションビットの指定で行える。

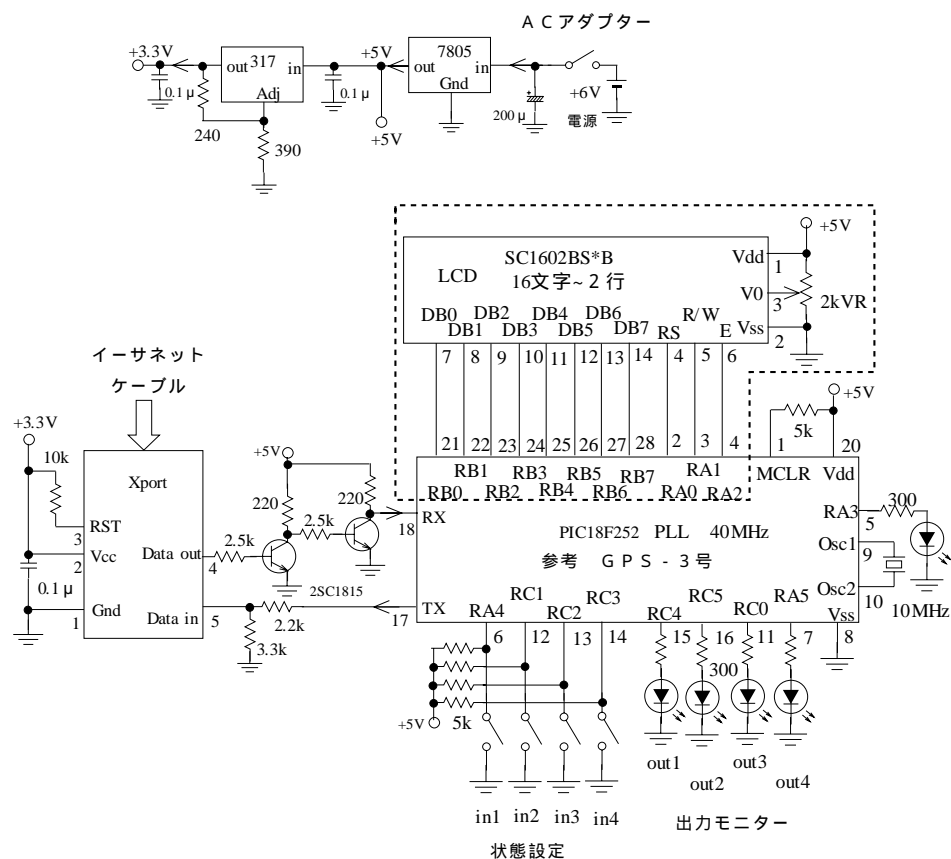


図2 Xport - ハード部の回路図 その1

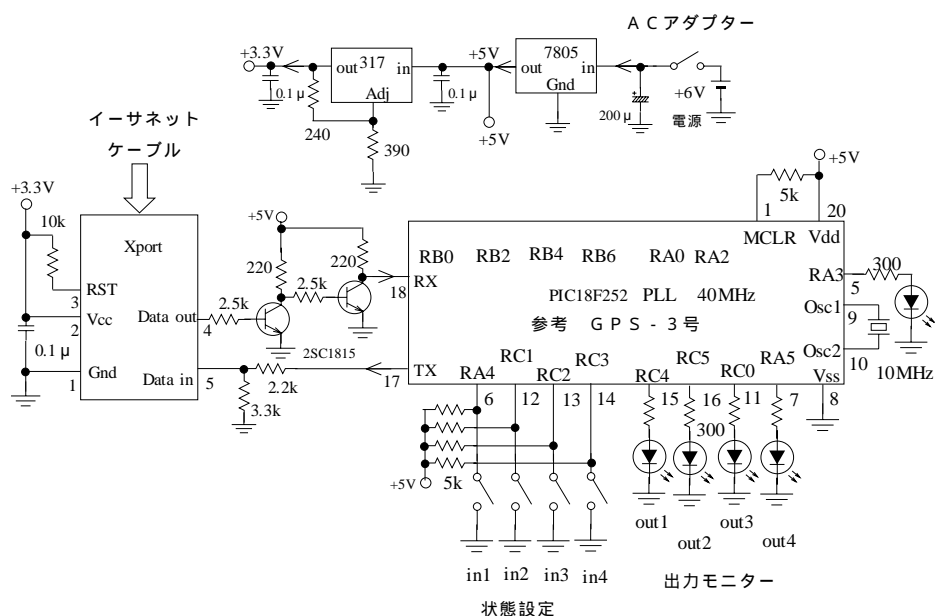


図3 Xport - ハード部の回路図 その2 C - L C Dは使用しない場合

ハード回路を制御及び監視するためのパソコン側で動かすソフトは、Visual Basic V.6 Professional Edition で行った。ソースプログラムはあまり長くはないので、表1として掲載しておく。プログラム作成では参考文献(3)にお世話になった。

PICに書き込んでいるアセンブラプログラムは少し長いので後半に資料として添付しておく。

プログラムファイル名 kinno-2

Dim string_data As String

Private Sub connect_click() 接続実行

With Winsock1

If .State = sockClosed Then

.RemoteHost = Text1.Text

.RemotePort = Text2.Text

.connect

End If

End With

End Sub

Private Sub Winsock1_Connect()

With Winsock1

'コネクション確立

Text4.Enabled = True 接続が実現したので、状況を知らせる

Text4.Text = "接続！" & _
.RemoteHostIP & " " & _
.RemotePort

End With

Frame5.Enabled = True ネット接続したので、制御盤を使用可とする

Text1.Enabled = False IPアドレス窓を使用不可とする

Text2.Enabled = False ポート番号窓を使用不可とする

End Sub

Private Sub Command2_Click() ジョブの終了

End

End Sub

Private Sub Option1_Click() スイッチ1中のボタン 当面Xport側のLED1を点灯させる

string_data = "11" 状態選択として文字1を代用する

Text3.Text = string_data 試験表示

With Winsock1

.SendData string_data

End With

End Sub

Private Sub Option2_Click() スイッチ1中のボタン 当面Xport側のLED1を消灯させる

```

string_data = "12"      状態選択として文字2を代用する
Text3.Text = string_data 試験表示

With Winsock1
    .SendData string_data
End With

End Sub

Private Sub Option3_Click() 'スイッチ2中のボタン 当面Xport側のLED2を点灯させる
    string_data = "21"      状態選択として文字1を代用する
    Text3.Text = string_data 試験表示

    With Winsock1
        .SendData string_data
    End With

End Sub

Private Sub Option4_Click() 'スイッチ2中のボタン 当面Xport側のLED2を消灯させる
    string_data = "22"      状態選択として文字2を代用する
    Text3.Text = string_data 試験表示

    With Winsock1
        .SendData string_data
    End With

End Sub

Private Sub Option5_Click() 'スイッチ3中のボタン 当面Xport側のLED3を点灯させる
    string_data = "31"      状態選択として文字1を代用する
    Text3.Text = string_data 試験表示

    With Winsock1
        .SendData string_data
    End With

End Sub

Private Sub Option6_Click() 'スイッチ3中のボタン 当面Xport側のLED13消灯させる
    string_data = "32"      状態選択として文字2を代用する
    Text3.Text = string_data 試験表示

    With Winsock1
        .SendData string_data
    End With

End Sub

Private Sub Option7_Click() 'スイッチ4中のボタン 当面Xport側のLED4を点灯させる
    string_data = "41"      状態選択として文字1を代用する
    Text3.Text = string_data 試験表示

    With Winsock1
        .SendData string_data
    End With

End Sub

Private Sub Option8_Click() 'スイッチ4のボタン 当面Xport側のLED4を消灯させる

```



```

string_data = "42"      状態選択として文字2を代用する
Text3.Text = string_data 試験表示

With Winsock1
    .SendData string_data
End With

End Sub

Private Sub Command1_Click() このコマンドボタンを押すことで、Xport側から状態を報告させる

    string_data = "re"      'report の re を代用する
    Text3.Text = string_data 試験表示

    With Winsock1
        .SendData string_data データの送信
    End With

End Sub

Private Sub Winsock1_DataArrival(ByVal bytesTotal As Long)
    'Xportから送られてくるデータを受け取る

    Dim strReceiveData As String 受信データ

    文字列データの受信
    Winsock1.GetData strReceiveData, vbString, bytesTotal
    Text6.Text = strReceiveData 文字列を表示する

End Sub

```

表1 Visual Basic のプログラム

3．システムの動作

上述した「Xport - ハード回路」が既に作成できたものとする。以下に本システムの使用手順を紹介する。

(1)「Xport - ハード回路」とハブの間を一般のイーサネットケーブルで接続する。

(2)「Xport - ハード回路」の電源を入れる。

(3) Xportを利用しているので、パソコン側にXport用ドライブソフトが既にインストールされているものとする。

Windowsの初期画面で、プログラムを選択 Xport Installerを選択 Xport Installerを実行する。

(4) XportInstaller画面で、searchアイコンをクリックする。ソフトはイントラネット内に接続されているXportを探しに行く。ネットに接続されているXportを探し出すと、それに関する情報が割り当てられたIPアドレスとともにテキスト欄に表示される。「Xport - ハード回路」にIPアドレスが割り当てられていることを確認することができる。「online」の表示があれば、Xportにアクセ

ス可能である。このあたりの説明は参考文献（１）に詳しい。

ここでのＩＰアドレスの割り当ては機関内ＬＡＮでの自動取得によっている。そのような機能のない場合は、プロバイダーあるいはシステム管理者などに問い合わせる必要がある。

（５）表１に掲示しているＶｉｓｕａｌ　Ｂａｓｉｃのプログラムを実行する。図４の画面が表示される。

画面中にはＩＰアドレスとポート番号には著者の開発時の値が表示されている。が、このテキスト内容を各自の正しいＩＰアドレスとポート番号に設定した後、接続実行のコンボボタンをクリックする。ポート番号は１０００１とする。これに関しては参考文献、あるいはＸｐｏｒｔに付属してくるマニュアルを参照すること。

ＩＰ番号、ポート番号の設定は表示のテキスト欄中で設定できるし、がプログラムの本文中で前もって設定することもできる。本学のＬＡＮでは、ＸｐｏｒｔにＩＰアドレスを割り当ててもらおうと、再接続時には同じＩＰアドレスが割り当てられるようである。



図４ プログラム実行の初期画面

（６）接続が成功すれば、テキスト欄に「接続！ 」のメッセージ等が表示される。接続に時間がかかる場合もある、そのような場合には、一度終了ボタンを押して、処理を終了する。時間をおいて再度処理を開始するようにすること。テキスト欄に「接続！ 」のメッセージが表示されない限り、接続がなされていない。そのような状態になったら、どこかに設定などの不調がある。問題を解決して再トライをすること。

(7)テキスト欄に「接続！」が表示されれば、ソフトは「X p o r t - ハード回路」を認識した。相互通信が可能となった。制御盤フレームが使用可となる。



図 5 ネット接続成功

(8)制御盤フレーム中のスイッチ 1 ～スイッチ 4 のどれでも、オプションボタンのオン、オフをクリックすれば、この設定に対応して、「X p o r t - ハード回路」では対応 L E D の点灯、消灯をリアルタイムで制御できる。

(9)「状況報告要請」コマンドボタンをクリックすると、クリックした時点で、「X p o r t - ハード回路」でスイッチ 1 からスイッチ 4 の設定状態(オン状態か、オフ状態か)がこのコマンドボタンの右隣のテキスト窓に表示される。制御・管理のフォーム画面が図 6 の場合の実際の「X p o r t - ハード回路」の作動中の様子が写真 1 に示されている。

(1 0)後はエンドレスである。

(1 1)制御・管理プログラムの実行を終了したければ、終了コマンドボタンをクリックする。



図6 制御・管理実行中

Visual Basic プログラムでは、スイッチ1に文字"1"、... スイッチ4に文字"4"を対応させ、オンに文字"1"、オフに文字"2"を対応させている。例えば、スイッチ3のオンをクリックすると、プログラムは文字列"31"を出力し、イーサネットを経由して、Xportに入力し、シリアルデータとなって、PICがこのデータを取り込む。アセンブラプログラム中で、RXポートから入力したこのデータ列をPICが解読し、文字列が"31"と判断すれば、ハード回路中の3番目のLEDを点灯させることになる。

状態報告要請のコマンドボタンをクリックすると、プログラムから文字列"re"が送信される。Xport側でこの文字列を受信すると、PICは4連スイッチのオン、オフ状態を調べに行き、その状態を文字列として送出する。例えば4連スイッチの状態が、1番目のスイッチはオン、2番目がオフ、3番目がオン、4番目がオンならば、PICは文字列"11223141"を返送してくる。このデータをプログラムで解読し、要請ボタンの右に位置しているテキスト欄に状態報告をする。図6にそれが示されている。

4. おわりに

XportとPIC、それにVisual Basic 6.0 Professional Editionを用いて、イーサネットを利用した基本的な遠隔制御・管理システムを構築することができた。java言語を用いた場合より、コントロールプログラム

は容易に書き上げることができたようである。また、V i s u a l B a s i cの汎用性から、基本的な本システムを土台として、様々なシステムへと拡張することも容易と考えている。

この基礎研究をもとに、今後とも応用性のあるシステムの構築を学生とともに考えていきたい。その1つとして、X p o r t側に、簡易なカメラまたは簡易なビデオカメラを装着し、パソコン画面でリアルタイムでX p o r t側の状況を視認できるようなシステムを考えている。このようなシステムならば、費用は格安ながら、遠隔制御・遠隔管理システムをして、一応申し分はないであろう。このようなシステムは多方面での応用に富んでもいよう。

参考文献

- (1) 「シリアル - イーサネット変換器X p o r tの試用レポート 前編、後編」小野康正、トランジスタ技術、2 0 0 3年9月号、1 0月号、C Q出版社。
- (2) 「イーサネット・シリアル変換器を用いた家庭用サーバの構築」村田 晃一、小山高専電子制御工学科金野研究室2 0 0 5年度卒業論文。2 0 0 6年3月。
- (3) 「V e r . 6 . 0対応 V i s u a l B a s i c 逆引き大全 5 0 0の極意」、池谷京子他、株式会社秀和システム。

2 0 0 6年6月1 4日