ジョイントモジュール Webアプリケーション

∨1.10 取扱説明書

2005年7月22日



渡辺電機工業株式会社

	Page
第1章 製品が届きましたら	1
第2章 動作概要	2
第3章 操作手順	3
第4章 操作	4
4-1 HTMLを作成する	4
4-2 HTMLをジョイントモジュールヘアップロードする	6
4-3 ジョイントモジュール内のHTMLを確認する	9
4-4 HTMLをジョイントモジュールからダウンロード	10
4-5 HTMLをジョイントモジュールから削除する	11
4-6 Webを表示する	11
第5章 SSIコマンド	1 2
5-1 デジタルデータの表示指定	1 2
5-2 SSI処理	1 2
5-3 表示データを取得する	13
5-4 取得したデータを表示	15
5-5 制御データを取り込む	17
5-6 制御する	1 9
第6章 サンプルプログラム	2 2 2
6-1 サンプルプログラム概要	2 2 2
6-2 サンプルプログラムの解説	2 3

·PC/ATは米国IBM社の登録商標です。

·Pentiumは米国Intel社の登録商標です。

・MS-Windowsは米国Microsoft社の登録商標です。

※この説明書の内容は、改善のため予告なしに一部変更することがありますのでご了承下さい。

目 次

第1章 製品が届きましたら

このたびは「**ジョイントモジュール**」(WJM□-□-□TNN 以下ジョイントモジュール) をお買い求めいただき誠にありがとうございます。

この取扱説明書には、ジョイントモジュールでWebを利用する際の操作について書かれています。

製品を正しく末永くご使用いただくために、ご使用前に必ずお読み下さい。

【製品の確認】

次のものがお手元に揃っているかご確認下さい。 なお、この取扱説明書は大切に保管して下さい。

- ・ 取扱説明書(本書)
 ...1部
- 注意!本書に記載されているユニットNo、チャンネルNo、サブユニットNo
 については「RS232C・UDP/IP通信仕様書」の
 バージョン3.50以降を参照してください。

第2章 動作概要

SSI機能によりジョイントモジュールの機能ユニット(WJF 以下機能ユニット)データ 及びリアルリンクのモジュールデータをWebブラウザで表示することが出来ます。



制限事項:設定可能データは12モジュールまたは12ユニットまで となります。

クライアント数は最大5台となります。

第3章 操作手順

ジョイントモジュールWeb機能を使用するための手順をご確認下さい。

【操作手順】



第4章 操作

4-1 HTMLを作成する

通常のHTMLのソースコードに第5章で説明するSSIコマンドを記述してHTMLを 作成します。

コマンド毎の詳細の説明は第5章を参照してください。

サンプルプログラムは第6章で説明しています、参照してください。

HTMLの標準的なコマンドの説明につきましては本書では省略させていただきます。

(1) モジュールのデータを表示する

注意! 下記①~③の順で記述する必要があります。

①デジタルデータの表示方法の指定

HTMLヘッダーの後に記述するようにします。

<!--#exec cmd_argument='DigitalDispType(_0_)'-->

- 0:0の時OFF、1の時ONと表示します

1:1の時ON、1の時OFFと表示します

詳しくは第5章5-1を参照してください。

②データの取得

モジュールのデータを通信で取得するコマンドを記述します。

SSI処理のコマンドの後にデータを取得するコマンドを記述します。

<!--#exec cmd_argument='SsiOutput("", "")'-->

<!--#exec cmd_argument='<u>RequestRD_Analog</u>(001,001,0,0)'-->

SSI処理コマンド アナログモジュールから全チャンネル分のデータを

詳しくは第5章5-2 取得するコマンドです。

を参照してください カッコ内は取得するモジュールの情報を指定しています。 この他にデジタルデータの取得 RequestRD_Digital など があります。 詳しくは第5章5-3を参照してください。

③データの表示

②で取得したデータを表示するコードを記述します。

HTMLのテキストボックスなどの"内容"にあたる部分に記述します。

<INPUT TYPE="text" NAME="003CH101" value=</pre>

"<!--#exec cmd_argument='ReadData(001,001,0,0,%s,Analog(1),0,0,0)'-->"
SIZE=22 MAXLENGTH=16 readonly>

②で取得したデータのアナログデータ
 デジタルデータの時:Digital(1)
 1チャンネル目を指定しています。

 ដ算データの時:Count(1)

 ដしくは第5章5-4を参照してください。

(2) モジュールのデータを制御する

注意! 下記の①、②の順で記述する必要があります。 制御をするには制御する値を取り込む処理を行った後、実際制御する処理をします。 ①データの取込

HTMLのテキスト名を指定して読込み用のコードを記述します。 <!--#exec cmd_argument='WriteData(001,001,2,0,030UT1,%s,Digital(1),0,0,0)'-->

● 030UT1という名のテキストボックスまたは
 ● アナログデータの時: Analog(1)
 ● クロ・ジュールの1チャンネル目
 ● として取込む指定をしています。
 詳しくは第5章5-5を参照してください

②データの制御

①で取込したデータを使用してモジュールに制御をかけます。
 SSI処理のコマンドの後に制御するコマンドを記述します。
 <!--#exec cmd_argument='SsiOutput("", "")'-->
 <!--#exec cmd_argument='RequestPD_Digital(001,001,2,0,0)'-->
 ✓
 SSI処理コマンド
 ①で取得した 001,001,2 のモジュールの
 詳しくは第5章5-2
 ごの他にアナログデータの制御 RequestPD_Analog など
 があります。
 詳しくは第5章5-6を参照してください。

③制御の解除

<!--#exec cmd_argument=' RequestPD_Cancel(001,001,2,0,0)'-->

Ψ
 001,001,2のモジュールに対しての
 制御を解除します。
 詳しくは第5章5-4を参照してください。

4-2 HTMLをジョイントモジュールへアップロードする

ジョイントモジュールへFTPを使用してアップロードします。 作成したHTMLをC:¥のルートパスに保存した場合の例を説明します。 拡張子は.HTMにしてください。 Windowsのコマンドプロンプトからコマンドを下記の手順で入力してください。

- ① コマンドプロンプトを実行します
- ② C:¥> と表示されている状態から ftp 192.168.1.10 と入力します。
 IP アドレスは工場出荷時の場合を表しています。



③ User の入力を要求されますので watanabe と入力します



④ Passwordを要求されますので rial と入力します。
 パスワードですので画面には入力した文字は表示されません。

⑤ パスワードが正しく認識されるとプロンプトが ftp> となります。



⑥ put XXXXXX.htm と入力しますとアップロードが開始します。
 ※XXXXXX.htm は作成したHTMLのファイル名になります。

⑦ Transfer complete と表示されましたらアップロードは完了です。

INVESTIGATE Complete
 INDER STREET
 INDER STREET

⑧ bye と入力しますとFTPが終了します。

```
コマンドプロンプト
    Microsoft Windows XP [Version 5.1.2600]
    (C) Copyright 1985-2001 Microsoft Corp.
    c:¥>ftp 192.168.1.10
    Connected to 192.168.1.10.
    220 NET+ARM FTP Server 1.0 ready.
User (192.168.1.10:(none)): watanabe
    331 User watanabe OK, send password.
    Password:
    230 Password OK.
    ftp> put watanabe.htm
    200 PORT command Ok.
    150 About to open data connection.
    226 Transfer complete
    ftp: 11319 bytes sent in 0.00Seconds 11319000.00Kbytes/sec.
→ ftp> bve
    221 Goodbye.
     ::¥>_
```

4-3 ジョイントモジュール内のHTMLを確認する

ジョイントモジュール内のHTMLを確認するには4-2の⑥で 1s と入力します。 Transfer complete と表示されたら確認完了です。 内部に存在するファイル名が表示されます。

注意! RiALLink.cfg は削除/変更しないでください。 ジョイントモジュールが正しく動かなくなります。



図の例では RiALLink. cfg と watanabe. htm が存在しています。

4-4 HTMLをジョイントモジュールからダウンロードする

HTMLをジョイントモジュールから取り出すには4-2の⑥で get XXXXXX.htm と 入力します。

Transfer complete と表示されたら取り出しは完了です。

C:¥のルートパスに XXXXXX.htm が作成されます。

注意! ファイル名は大文字小文字を正しく入力してください。

	🛤 コマンド プロンプト - ftp 192.168.1.10
	Microsoft Windows XP [Version 5.1.2600]
	(C) Copyright 1985-2001 Microsoft Corp.
	V.FL- 102 100 1 10
	C:#/TTP 102.100.1.10 Composted to 102 160 1 10
	220 NET+ADW ETR Samuer 1 0 maadu
	220 NETTARM FIF Server I.U ready.
	User (192.168.1.10:(none)): watanabe
	331 User watanabe UK, send password.
	Password:
	230 Password OK.
	ftp> get watanabe.htm
	200 PORT command Ok.
	150 About to open data connection.
≻	226 Transfer complete.
	ftp: 11319 bytes received in 0.52Seconds 21.98Kbytes/sec.
	ftp>

4-5 HTMLをジョイントモジュールから削除する

HTMLを削除したいときは4-2の⑥で delete XXXXX.htmと入力します。 DELE command successful と表示されたら削除成功です。 ジョイントモジュールから XXXXXX.htm が削除されます。 注意! ファイル名は大文字小文字を正しく入力してください。

注意! RiALLink.cfg は削除/変更しないでください。 ジョイントモジュールが正しく動かなくなります。



4-6 Webを表示する

インターネットブラウザを使ってURLを指定します。 URLは http://<u>192.168.1.10/XXXXXX.htm</u> ↓ アップロードした HTML の名前

ジョイントモジュールのIPアドレス となります。

第5章 SSIコマンド

5-1 デジタルデータの表示指定

・コマンド形式

<!--#exec cmd_argument='DigitalDispType(Type)'-->

・パラメータ

Type:表示タイプ (0,1)

0を指定した時: 0=0FF、1=0N になります
 1を指定した時: 0=0N、 1=0FF になります
 ※このコマンドを指定しない場合は0, 1表示になります。

5-2 SSI処理

・コマンド形式

<!-=#exec cmd_argument='SsiOutput(" $\bigcirc \bigcirc$ ", " $\triangle \triangle$ ")'-->

・パラメータ

○○:実行結果が正常の場合出力する文字を指定します
 △△:実行結果が異常の場合出力する文字を指定します
 ※出力なしの場合は""と指定します。
 最大 126 文字まで指定可能です。

※このコマンドは5-3で説明する表示データ取得コマンド

RequestRD_Analog

RequestGD_Analog

RequestRD_Digital

RequestGD_Digital

5-6で説明する制御コマンド

RequestPD_Analog

RequestPN_Analog

RequestPD_Digital

RequestPT_Digital

を記述する行と同じ行の直前に記述してください。

記述をしない場合、その行における上記コマンドは実行結果が正常でも異常でも 文字の表示はなしになります。

例: (表示の関係で折り返して2行になっています。実際には1行で記述してください) 1行目は実行結果が成功で OK、異常で NG と文字表示されます。

2行目は実行結果が成功でも異常でも文字が表示されません。

1 行目: <!--#exec cmd_argument='SsiOutput("OK","NG")'-->

<!--#exec cmd_argument='RequestRD_Analog(000,000,0,0)'-->

2 行目: <!--#exec cmd_argument='RequestRD_Analog(000,000,1,0)'-->

5-3 表示データを取得する

表示するためにまずデータをLON通信で取得します。 ○アナログデータ取得

- ・コマンド形式
 - 下記パラメータで指定したモジュールの全チャンネルのデータをLON通信で 取得する
 - <!--#exec cmd_argument='RequestRD_Analog(Subnet, Node, Unit, Circ)'-->
 - 下記パラメータで指定したモジュールのデータをLON通信で取得する <!--#exec cmd_argument='RequestGD_Analog(Subnet, Node, Unit, Ch, Circ)'-->
- ・パラメータ

Subnet: サブネットナンバー

データを取得するモジュールのサブネットNoを指定します。 モジュール登録ツールにて登録したサブネットNoを指定します。

Node:モジュールナンバー

データを取得するモジュールのモジュールNoを指定します。 モジュール登録ツールにて登録したモジュールNoを指定します。

Unit: ユニットナンバー(0~C) データを取得するモジュールのユニット No を16進数で指定します。 ユニットAを指定する場合0とします。ユニットHの場合は7となります。 ユニットNoが存在しないモジュールの場合Cを指定します。

 $Ch: \mathcal{F} + \mathcal{V} \times \mathcal{V} + \mathcal{V} \times \mathcal{V} = (0 \sim 1 F)$

データを取得するモジュールのチャンネルNoを16進数で指定します。
1 chのデータを取得する場合は0となります。
1 6 chのデータを取得する場合はFとなります。

Circ:サブユニットナンバー(0~4) ジョイントモジュールの機能ユニットのサブユニットナンバーを 0~3まで指定します。 ジョイントモジュール以外を指定する場合は4を指定します。 ○デジタルデータ取得

- ・コマンド形式
 - 下記パラメータで指定したモジュールの全チャンネルのデータをLON通信で 取得する

<!--#exec cmd_argument='RequestRD_ Digital (Subnet, Node, Unit, Circ)'-->

- 下記パラメータで指定したモジュールのデータをLON通信で取得する
 - <!--#exec cmd_argument='RequestGD_Digital(Subnet, Node, Unit, Ch, Circ)'-->
- ・パラメータ
 - Subnet:サブネットナンバー

データを取得するモジュールのサブネットNoを指定します。 モジュール登録ツールにて登録したサブネットNoを指定します。

Node:モジュールナンバー

データを取得するモジュールのモジュールNoを指定します。 モジュール登録ツールにて登録したモジュールNoを指定します。

データを取得するモジュールのユニット No を16進数で指定します。 ユニットAを指定する場合0とします。ユニットHの場合は7となります。 ユニットNoが存在しないモジュールの場合Cを指定します。

 $Ch: \mathcal{F} + \mathcal{V} \times \mathcal{V} + \mathcal{V} \times \mathcal{V} = (0 - 1 F)$

データを取得するモジュールのチャンネルNoを16進数で指定します。
1 chのデータを取得する場合は0となります。
1 6 chのデータを取得する場合はFとなります。

Circ:サブユニットナンバー(0~4) ジョイントモジュールの機能ユニットのサブユニットナンバーを 0~3まで指定します。 ジョイントモジュール以外を指定する場合は4を指定します。

5-4 取得したデータを表示する

5-3で取得したデータを表示します。

・コマンド形式

<!--#exec cmd_argument='ReadData (Subnet, Node, Unit, Circ, 書式, OFFSET,

Scale, arg1, arg2) '-->

・パラメータ

Subnet: サブネットナンバー

5-3で取得したデータのサブネットNoを指定します。

モジュール登録ツールにて登録したサブネットNoを指定します。

Node:モジュールナンバー

5-3で取得したデータのモジュールNoを指定します。 モジュール登録ツールにて登録したモジュールNoを指定します。

Unit: ユニットナンバー(0~C) 5-3で取得したデータのユニットNoを16進数で指定します。 ユニットAを指定する場合0とします。ユニットHの場合は7となります。

ユニットNoが存在しないモジュールの場合Cを指定します。

Circ:サブユニットナンバー(0~4) ジョイントモジュールの機能ユニットのサブユニットナンバーを 0~3まで指定します。 ジョイントモジュール以外を指定する場合は4を指定します。

書式:%s 固定になります。

OFFSET: Analog(No) アナログデータの取得の場合指定します
 Digital(No) デジタルデータの取得の場合指定します
 Count(No) 積算データの取得の場合指定します
 Noには5-3で取得したデータのチャンネルを10進数で指定します
 (1~21)
 注意! 1の時1チャンネル目、21の時21チャンネル目となります。
 「RS232C・UDP/IP通信仕様書」と異なります。

Scale: スケール変換パターン番号 (0~2) 0:スケールなし 1:スケーリングあり 5-3で取得したデータ × (arg1 - arg2/100 + arg2) 2:係数を掛ける 5-3取得したデータ × arg1 arg1,2:スケール値/係数

Scale=1の時 age1=スケールH I age2=スケールLOW

Scale=2の時 age1=係数

※整数部最大10桁まで指定できます。

小数点以下は最大2桁まで指定できます。

5-5 制御データを取り込む

制御するためにまず制御するデータをWeb画面から取り込みます。

・コマンド形式

<!--#exec cmd_argument='WriteData (Subnet, Node, Unit, Circ, ジンボッル, 書式, OFFSET,

Scale, arg1, arg2) '-->

・パラメータ

Subnet: サブネットナンバー

制御するモジュールのサブネットNoを指定します。

モジュール登録ツールにて登録したサブネットNoを指定します。

Node:モジュールナンバー

制御するモジュールのモジュールNoを指定します。 モジュール登録ツールにて登録したモジュールNoを指定します。

Unit: ユニットナンバー(0~C) 制御するモジュールのユニットNoを16進数で指定します。 ユニットAを指定する場合0とします。ユニットHの場合は7となります。 ユニットNoが存在しないモジュールの場合Cを指定します。

Circ: サブユニットナンバー(0~4) ジョイントモジュールの場合は0を指定します。 ジョイントモジュール以外を指定する場合は4を指定します。

シンボル:テキストボックスなどに指定したシンボル名を指定します。

書式:%s 固定になります。

OFFSET: Analog(No) アナログデータの取得の場合指定します
 Digital(No) デジタルデータの取得の場合指定します
 Count(No) 積算データの取得の場合指定します
 Noにはモジュールのチャンネルを10進数で指定します(1~16)
 注意! 1の時1チャンネル目、16の時16チャンネル目となります。
 「RS232C・UDP/IP通信仕様書」と異なります。

Scale: スケール変換パターン番号 (0~2)

0:スケールなし

- 1:スケーリングあり
 取り込む制御データがスケーリングしてある場合に指定します
 制御データ ÷ (arg1 arg2/100 + arg2)
- 2:係数 取り込む制御データが係数を掛けてある場合に指定します 制御データ ÷ arg1

arg1,2:スケール値/係数

Scale=1の時 age1=スケールH I age2=スケールLOW

Scale=2の時 age1=係数

※整数部最大10桁まで指定できます。

小数点以下は最大2桁まで指定できます。

5-6 制御する

5-5で取り込んだデータで制御します。

- ○アナログ制御
 - ・コマンド形式

下記パラメータで指定したモジュールの全データをLON通信で制御する <!--#exec cmd_argument='RequestPN_Analog(Subnet, Node, Unit, Circ, ChCnt)'-->

下記パラメータで指定したモジュールのデータをLON通信で制御する <!--#exec cmd_argument='RequestPD_Analog(Subnet, Node, Unit, Ch, Circ)'-->

・パラメータ

Subnet:サブネットナンバー

5-5で取り込んだデータのサブネットNoを指定します。 モジュール登録ツールにて登録したサブネットNoを指定します。

Node:モジュールナンバー

5-5で取り込んだデータのモジュールNoを指定します。 モジュール登録ツールにて登録したモジュールNoを指定します。

Unit: ユニットナンバー(0~C) 5-5で取り込んだデータのユニットNoを16進数で指定します。 ユニットAを指定する場合0とします。ユニットHの場合は7となります。 ユニットNoが存在しないモジュールの場合Cを指定します。

 $Ch: \mathcal{F} + \mathcal{V} \times \mathcal{V} + \mathcal{V} \times \mathcal{V} = (0 \sim 1 F)$

5-5で取り込んだデータのチャンネルNoを16進数で指定します。
1 chのデータを制御する場合は0となります。
1 6 chのデータを制御する場合はFとなります。

Circ: サブユニットナンバー(0~4) ジョイントモジュールの場合は0を指定します。 ジョイントモジュール以外を指定する場合は4を指定します。

ChCnt: チャンネル数(1~16)

複数設定時の設定データチャンネル数を設定します。 例:8チャンネルを指定すると9チャンネル以降は制御されません。 ○デジタル制御

・コマンド形式

下記パラメータで指定したモジュールの全データをLON通信で制御する <!--#exec cmd_argument='RequestPT_Digital(Subnet, Node, Unit, Circ, ChCnt)'-->

下記パラメータで指定したモジュールのデータをLON通信で制御する <!--#exec cmd_argument='RequestPD_Digital(Subnet, Node, Unit, Ch, Circ)'-->

・パラメータ

Subnet:サブネットナンバー

5-5で取り込んだデータのサブネットNoを指定します。 モジュール登録ツールにて登録したサブネットNoを指定します。

Node:モジュールナンバー

5-5で取り込んだデータのモジュールNoを指定します。 モジュール登録ツールにて登録したモジュールNoを指定します。

Unit: ユニットナンバー(0~C) 5-5で取り込んだデータのユニットNoを16進数で指定します。 ユニットAを指定する場合0とします。ユニットHの場合は7となります。 ユニットNoが存在しないモジュールの場合Cを指定します。

Ch: チャンネルナンバー(0~1F)
 5-5で取り込んだデータのチャンネルNoを16進数で指定します。
 1 chのデータを制御する場合は0となります。
 1 6 chのデータを制御する場合はFとなります。

Circ:サブユニットナンバー(0~4) ジョイントモジュールの場合は0を指定します。 ジョイントモジュール以外を指定する場合は4を指定します。

ChCnt: チャンネル数(1~16)

複数設定時の設定データチャンネル数を設定します。 例:8チャンネルを指定すると9チャンネル以降は制御されません。 ○制御の解除

・コマンド形式

下記パラメータで指定したモジュールのデータをLON通信で制御解除するコマンド <!--#exec cmd_argument='RequestPD_Cancel(Subnet, Node, Unit, Ch, Circ)'-->

・パラメータ

Subnet: サブネットナンバー

制御解除するモジュールのサブネットNoを指定します。 モジュール登録ツールにて登録したサブネットNoを指定します。

Node: モジュールナンバー 制御解除するモジュールのモジュールNoを指定します。 モジュール登録ツールにて登録したモジュールNoを指定します。

Unit: ユニットナンバー(0~C) 制御解除するユニットNoを16進数で指定します。 ユニットAを指定する場合0とします。ユニットHの場合は7となります。 ユニットNoが存在しないモジュールの場合Cを指定します。

Ch: チャンネルナンバー(0~1F)
 制御解除するチャンネルNoを16進数で指定します。
 1 chのデータを制御解除する場合は0となります。
 1 6 chのデータを制御解除する場合はFとなります。

Circ:サブユニットナンバー(0~4)

ジョイントモジュールの場合は0を指定します。 ジョイントモジュール以外を指定する場合は4を指定します。

ジョイント WEB サンプル 自動リフレッシュ10秒毎)ユニット ディスクリート出力16ch Aユニット アナログ入力8ch Bユニット 電力ユニット(三相3線) Cユニット ディスクリート入力16ch 0.00 1ch -0.00 R相電流 デジタル1ch OFF カウント1ch 📃 2ch -0.00 デジタル2ch OFF S相電流 0.00 カウント2ch 🕛 3ch -0.00 デジタル3ch OFF 7相電流 0.00 カウント3ch 🛛 🔍 4ch -0.00 デジタル4ch OFF R-S電圧 0.00 カウント4ch 🛛 🔍 5ch -0.00 S-T電圧 0.00 時間積算1ch 0 デジタル5ch OFF 6ch -0.00 T-R電圧 0.00 時間積算2ch 0 デジタル6ch OFF 時間積算3ch 0 デジタル7ch OFF 7ch -0.00 有効電力 0.00 8ch -0.00 無効電力 0.00 時間積算4ch 0 デジタル8ch OFF 上記状態で制御する 力率 0.00 デジタル1ch OFF 周波数 0.00 デジタル2ch OFF デジタル3ch OFF 有効電力量 0.00 無効電力量 0.00 デジタル4ch OFF

6-1 サンプルプログラム概要

第6章 サンプルプログラム

サンプルのHTMLは工場出荷時にジョイントモジュールにアップロードしてあります。 ジョイントモジュールからダウンロードし、修正してお使いいただけます。

○構成

ジョイントモジュールに機能ユニットA~Dまで4つを接続した時のHTMLになります。 機能ユニットA:アナログ入力モジュール(WJF-AI16) 機能ユニットB:電力モジュール(三相3線)(WJF-PA2) 機能ユニットC:デジタル入力モジュール(WJF-DI1624) 機能ユニットD:デジタル出力モジュール(WJF-D016□)

○動作

・10秒ごとに自動で更新してデータを表示します。

 「上記状態で制御する」ボタンで機能ユニットDのデジタル出力モジュールを制御 できるようになっています。

6-2 サンプルプログラムの解説

①10行目

DigitalDiuspType にてデジタル表示をON/OFFに指定します。

②11行目~16行目

SsiOutput と Request RD_Analog、Request RD_Digital を使用して モジュールからデータを取得する指示をします。

③27行目~34行目

機能ユニットのアナログ入力モジュール(WJF-AI16)のデータを表示する部分。 テキストボックスの value の部分に ReadData を記述して②で取得したデータを 表示する指示をします。

④39行目~50行目

機能ユニットの電力モジュール(WJF-PA2)のデータを表示する部分。 テキストボックスの value の部分に ReadData を記述して②で取得したデータを 表示する指示をします。

⑤55行目~66行目

機能ユニットのデジタル入力モジュール(WJF-DI1624)のデータを表示する部分。 テキストボックスの value の部分に ReadData を記述して②で取得したデータを 表示する指示をします。

⑥69行目~77行目

機能ユニットのデジタル出力モジュール(WJF-D016□)のデータを表示する部分。 テキストボックスの value の部分に ReadData を記述して②で取得したデータを 表示する指示をします。

⑦80行目~87行目

テキストボックスに書き込まれている値を取得するように指示します。

⑧90行目~96行目

SsiOutput と RequestPD_Digital を使用して④で取得したデータで制御するよう に指示します。

※その他の部分につきましてはHTML標準の記述になりますので本書では 省略させていただきます。

1	<pre>(html xmlns:v="urn:schemas-microsoft-com:vml" xmlns:o="urn:schemas-microsoft-com:office:office" xmlns="http://www.w3.org/TR/REC-html40"></pre>					
2						
3	<head></head>					
4	<pre><meta content="ja" http='equiv="Content=Language"'/></pre>					
5	<meta content="text/html; charset=utf-8" http-equiv="Content-Type"/>					
6	<meta content="10" http='equiv="refresh"'/>					
7	〈title〉ジョイント WEB サンプル\/title〉					
8						
9						
10	#exec cmd_argument=`DigitalDispType(0)`					
11	#exec cmd_argument='SsiOutput("", "")'X!#exec cmd_argument='RequestRD_Analog(000,000,0,0)'					
12	#exec cmd_argument='SsiOutput("", "")'X!#exec cmd_argument='RequestRD_Analog(000,000,1,0)'					
13	#exec cmd_argument='SsiOutput("","")'×!#exec cmd_argument='RequestRD_Analog(000,000,2,0)'					
14	#exec cmd_argument='SsiOutput("","")' #exec cmd_argument='RequestRD_Analog(000,000,2,1)'					
15	#exec cmd_argument='SsiOutput("","")' #exec cmd_argument='RequestRD_Digital(000,000,2,2)'					
16	#exec cmd_argument='SsiOutput("","")' #exec cmd_argument='RequestRD_Digital(000,000,3,0)'					
17						
18	<body bgcolor="#3399FF" style="text-align: left"></body>					
19						
20	<form action="watanabe.htm" method="POST"></form>					
21						
22	<pre><pre>align="center"×i>b>font size="6">ジョイント WEB サンプル/p></pre></pre>					
23	自動リフレッシュ 10 秒毎					
24						
25	<pre><div id="layer2" style="position: absolute; width: 203px; height: 531px; z-index: 1; left: 10px; top: 141px; visibility: visible; border-style: ridge; border-width: 3px"></div></pre>					
26	Aユニット アナログ入力8ch					
27	1 c h <input maxlength="16" name="003CH101" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,0,0,%s,Analog(1),0,0,0)'->"/>					
28	2 c h <input maxlength="16" name="003CH102" readonly="" size="22" type="text" value="<!#exec cmd_argument=' ReadData(000,000,0,0,%s, Analog(2),0,0,0)' ->"/>					
29	3 c h <input maxlength="16" name="003CH103" readonly="" size="22" type="text" value="<!#exec cmd_argument=' ReadData(000,000,0,0,%s, Analog(3),0,0,0)'->"/>					
30	4 c h <input maxlength="16" name="003CH104" readonly="" size="22" type="text" value="<!#exec cmd_argument=' ReadData(000, 000, 0, 0, %s, Analog(4), 0, 0, 0)' ->"/>					
31	5 c h <input maxlength="16" name="003CH105" readonly="" size="22" type="text" value="<!#exec cmd_argument=' ReadData(000,000,0,0,%s, Analog(5),0,0,0)'->"/>					
32	6 c h <input maxlength="16" name="003CH106" readonly="" size="22" type="text" value="<!#exec cmd_argument=' ReadData(000,000,0,0,%s, Analog(6),0,0,0)'->"/>					
33	7 c h <input maxlength="16" name="003CH107" readonly="" size="22" type="text" value="<!#exec cmd_argument=' ReadData(000,000,0,0,%s, Analog(7),0,0,0)'>"/>					
34	Sch <input maxlength="16" name="003CH108" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,0,0,%s,Analog(8),0,0,0)'>"/>					
35						
36						

<div style="position: absolute; width: 270px; height: 532px; z-index: 2; left: 214px; top: 141px; visibility: visible; border-style: ridge" id="layer3">

Bユニット	電力ユニット	(三相3線)
-------	--------	--------

	R相電流	<pre><input maxlength="16" name="003CH09" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(1),0,0,0)' ->"/></pre>
	S相電流	<input maxlength="16" name="003CH10" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(2),0,0,0)'->"/>
	T 相電流	 <input maxlength="16" name="003CH11" p="" readonly×="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(3),0,0,0)'->"/>
	R-S 電圧	<input maxlength="16" name="003CH12" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(4),0,0,0)'->"/>
	S-T 電圧	<input maxlength="16" name="003CH13" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(5),0,0,0)'->"/>
	T-R 電圧	<input maxlength="16" name="003CH14" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(6),0,0,0)'->"/>
	有効電力	 <input maxlength="16" name="003CH15" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(7),0,0,0)'>"/>
	無効電力	 <input maxlength="16" name="003CH16" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(8),0,0,0)'>"/>
	力率	 <input maxlength="16" name="003CH17" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(9),0,0,0)' ->"/>
	周波数	 <pre><input maxlength="16" name="003CH18" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(10),0,0,0)'->"/></pre>
	有効電力量	<pre><input maxlength="16" name="003CH19" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(11),0,0,0)' ->"/></pre>
	無効電力量	<pre><input maxlength="16" name="003CH20" readonly="" size="22" type="text" value="<!#exec cmd_argument='ReadData(000,000,1,0,%s,Analog(12),0,0,0)' ->"/></pre>

</ti><div style="position: absolute; width: 284px; height: 532px; z-index: 3; left: 484px; top: 141px; visibility: visible; border-style: ridge" id="layer4">

Cユニット ディスクリート入力16ch

	カウント1 c h	 <input <="" name="003CH21" th="" type="text"/> <th>value="<!--#exec cmd_argument='ReadData(000,000,2,0,%s,Count(1),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly></th>	value=" #exec cmd_argument='ReadData(000,000,2,0,%s,Count(1),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly>
	カウント2ch	 <input <="" name="003CH109" td="" type="text"/> <td>' value="<!--#exec cmd_argument='ReadData(000,000,2,0,%s,Count(2),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly></td>	' value=" #exec cmd_argument='ReadData(000,000,2,0,%s,Count(2),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly>
	カウント3ch	 <input <="" name="003CH23" td="" type="text"/> <td>value="<!--#exec cmd_argument='ReadData(000,000,2,0,%s,Count(3),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly></td>	value=" #exec cmd_argument='ReadData(000,000,2,0,%s,Count(3),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly>
	カウント4ch	 <input <="" name="003CH24" td="" type="text"/> <td>value="<!--#exec cmd_argument='ReadData(000,000,2,0,%s,Count(4),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly></td>	value=" #exec cmd_argument='ReadData(000,000,2,0,%s,Count(4),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly>
	時間積算1ch	<input <="" name="003CH37" td="" type="text"/> <td>value="<!--#exec cmd_argument='ReadData(000,000,2,1,%s,Count(1),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly></td>	value=" #exec cmd_argument='ReadData(000,000,2,1,%s,Count(1),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly>
	時間積算2ch	<input <="" name="003CH38" td="" type="text"/> <td>value="<!--#exec cmd_argument='ReadData(000,000,2,1,%s,Count(2),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly></td>	value=" #exec cmd_argument='ReadData(000,000,2,1,%s,Count(2),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly>
	時間積算3ch	<input <="" name="003CH39" td="" type="text"/> <td>value="<!--#exec cmd_argument='ReadData(000,000,2,1,%s,Count(3),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly></td>	value=" #exec cmd_argument='ReadData(000,000,2,1,%s,Count(3),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly>
	時間積算4ch	<input <="" name="003CH40" td="" type="text"/> <td>value="<!--#exec cmd_argument='ReadData(000,000,2,1,%s,Count(4),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly></td>	value=" #exec cmd_argument='ReadData(000,000,2,1,%s,Count(4),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly>
	デジタル1ch	 <input <="" name="003CH53" td="" type="text"/> <td>value="<!--#exec cmd_argument='ReadData(000,000,2,2,%s,Digital(1),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly×//p></td>	value=" #exec cmd_argument='ReadData(000,000,2,2,%s,Digital(1),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly×//p>
	デジタル2ch	 <input <="" name="003CH54" td="" type="text"/> <td>value="<!--#exec cmd_argument='ReadData(000,000,2,2,%s,Digital(2),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly×//p></td>	value=" #exec cmd_argument='ReadData(000,000,2,2,%s,Digital(2),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly×//p>
	デジタル3ch	 <input <="" name="003CH55" td="" type="text"/> <td>value="<!--#exec cmd_argument='ReadData(000,000,2,2,%s,Digital(3),0,0,0)'--->" SIZE=22 MAXLENGTH=16 readonly×//p></td>	value=" #exec cmd_argument='ReadData(000,000,2,2,%s,Digital(3),0,0,0)'- " SIZE=22 MAXLENGTH=16 readonly×//p>
	デジタル4ch	 <input <="" name="003CH57" td="" type="text"/> <td><math display="block">value= "<!---= \#exec cmd_argument='ReadData(000, 000, 2, 2, \%s, Digital(4), 0, 0, 0)' \rightarrow "SIZE= 22 MAXLENGTH= 16 readonly \times /p --> 10 models and the second sec</math></td>	$value= " 10 models and the second sec$

68 〈div style="position: absolute; width: 274px; height: 533px; z-index: 4; left: 769px; top: 140px; visibility: visible; border-style: ridge" id="layer5"〉 69 Dユニット ディスクリート出力16ch

70		デジタル1 c h <pre></pre> <pre></pre> <pre></pre> <pre>(INPUT TYPE="text" NAME="0030UT1" value="<!--#exec cmd_argument='ReadData(000,000,3,0,%s,Digital(1),0,0,0)'--->" SIZE=10 MAXLENGTH=16></pre>
71		デジタル2 c h
72		デジタル3 ch 〈INPUT TYPE="text" NAME="0030UT3" value=" #exec cmd_argument='ReadData(000,000,3,0,%s, Digital(3),0,0,0)'- " SIZE=10 MAXLENGTH=16>
73		デジタル4 c h
74		デジタル5 c h <pre><input maxlength="16" name="0030UT5" size="10" type="text" value="<!-#exec cmd_argument='ReadData(000,000,3,0,%s, Digital(5),0,0,0)'->"/></pre>
75		デジタル6ch 〈INPUT TYPE="text" NAME="0030UT6" value=" #exec cmd_argument='ReadData(000,000,3,0,%s,Digital(6),0,0,0)'- " SIZE=10 MAXLENGTH=16>
76		デジタル7ch 〈INPUT TYPE="text" NAME="0030UT7" value=" #exec cmd_argument='ReadData(000,000,3,0,%s,Digital(7),0,0,0)'- " SIZE=10 MAXLENGTH=16>
77		デジタル8 c h
78		
79	<input< td=""><td>TYPE="submit" VALUE="上記状態で制御する"></td></input<>	TYPE="submit" VALUE="上記状態で制御する">
80	-</td <td>-#exec cmd_argument='WriteData(000,000,3,0,0030UT1,%s,Digital(1),0,0,0)'></td>	-#exec cmd_argument='WriteData(000,000,3,0,0030UT1,%s,Digital(1),0,0,0)'>
81	-</td <td>-#exec cmd_argument='WriteData(000,000,3,0,0030UT2,%s,Digital(2),0,0,0)'></td>	-#exec cmd_argument='WriteData(000,000,3,0,0030UT2,%s,Digital(2),0,0,0)'>
82	-</td <td>#exec cmd_argument='WriteData(000,000,3,0,00030UT3,%s,Digital(3),0,0,0)'></td>	#exec cmd_argument='WriteData(000,000,3,0,00030UT3,%s,Digital(3),0,0,0)'>
83	-</td <td>-#exec cmd_argument='WriteData(000,000,3,0,0030UT4,%s,Digital(4),0,0,0)'></td>	-#exec cmd_argument='WriteData(000,000,3,0,0030UT4,%s,Digital(4),0,0,0)'>
84	-</td <td>-#exec cmd_argument='WriteData(000,000,3,0,0030UT5,%s,Digital(5),0,0,0)'></td>	-#exec cmd_argument='WriteData(000,000,3,0,0030UT5,%s,Digital(5),0,0,0)'>
85	-</td <td>-#exec cmd_argument='WriteData(000,000,3,0,0030UT6,%s,Digital(6),0,0,0)'></td>	-#exec cmd_argument='WriteData(000,000,3,0,0030UT6,%s,Digital(6),0,0,0)'>
86	-</td <td>-#exec cmd_argument='WriteData(000,000,3,0,0030UT7,%s,Digital(7),0,0,0)'></td>	-#exec cmd_argument='WriteData(000,000,3,0,0030UT7,%s,Digital(7),0,0,0)'>
87	-</td <td>-#exec cmd_argument='WriteData(000,000,3,0,0030UT8,%s,Digital(8),0,0,0)'></td>	-#exec cmd_argument='WriteData(000,000,3,0,0030UT8,%s,Digital(8),0,0,0)'>
88		
89	-</td <td>-#exec cmd_argument='SsiOutput("", "")'-><!--#exec cmd_argument='RequestPD_Digital(000,000,3,0,0)'--></td>	-#exec cmd_argument='SsiOutput("", "")'-> #exec cmd_argument='RequestPD_Digital(000,000,3,0,0)'
90	-</td <td>-#exec cmd_argument='SsiOutput("", "")'-><!--#exec cmd_argument='RequestPD_Digital(000,000,3,1,0)'--></td>	-#exec cmd_argument='SsiOutput("", "")'-> #exec cmd_argument='RequestPD_Digital(000,000,3,1,0)'
91	-</td <td>-#exec cmd_argument='SsiOutput("", "")'-><!--#exec cmd_argument='RequestPD_Digital(000,000,3,3,0)'--></td>	-#exec cmd_argument='SsiOutput("", "")'-> #exec cmd_argument='RequestPD_Digital(000,000,3,3,0)'
92	-</td <td>-#exec cmd_argument='SsiOutput("", "")'-><!--#exec cmd_argument='RequestPD_Digital(000,000,3,2,0)'--></td>	-#exec cmd_argument='SsiOutput("", "")'-> #exec cmd_argument='RequestPD_Digital(000,000,3,2,0)'
93	-</td <td>-#exec cmd_argument='SsiOutput("", "")'-><!--#exec cmd_argument='RequestPD_Digital(000,000,3,4,0)'--></td>	-#exec cmd_argument='SsiOutput("", "")'-> #exec cmd_argument='RequestPD_Digital(000,000,3,4,0)'
94	-</td <td>-#exec cmd_argument='SsiOutput("","")'-><!--#exec cmd_argument='RequestPD_Digital(000,000,3,5,0)'--></td>	-#exec cmd_argument='SsiOutput("","")'-> #exec cmd_argument='RequestPD_Digital(000,000,3,5,0)'
95	-</td <td>-#exec cmd_argument='SsiOutput("","")'-><!--#exec cmd_argument='RequestPD_Digital(000,000,3,6,0)'--></td>	-#exec cmd_argument='SsiOutput("","")'-> #exec cmd_argument='RequestPD_Digital(000,000,3,6,0)'
96	-</td <td>-#exec cmd_argument='SsiOutput("", "")'-><!--#exec cmd_argument='RequestPD_Digital(000,000,3,7,0)'---></td>	-#exec cmd_argument='SsiOutput("", "")'-> #exec cmd_argument='RequestPD_Digital(000,000,3,7,0)'-
97		
98		
99		
100		

101 </html>

渡辺電機工業株式会社

〒150-0001 東京都渋谷区神宮前6丁目16番地19号
 電話 03(3400)6141(代表) FAX 03(3409)3156
 ホ-ムペ→ジ http://www.watanabe-electric.co.jp
 (JR原宿駅/地下鉄千代田線明治神宮前駅下車)