

Y-Con Wxx(白黒2色 EPD)モジュール操作説明書

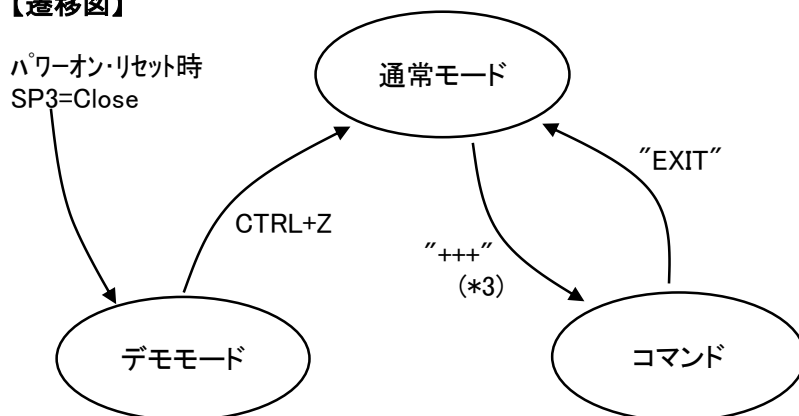
Ver 1.1

■動作モード

このモジュールには、3つの動作モードがあります。

項番	動作モード	動作
(1)	通常モード	ホストから画像データが転送されると描画します。
(2)	コマンドモード	ホストと対話形式でコマンドが実行できます。
(3)	デモモード	デモ用です。(SP3をショート時)

【遷移図】



(1)通常モード …… プロンプトは '>(3Eh)

ホストからビットマップファイル(*1)を送信することで描画が行えるモードです。
 パワーオン・リセット後、待機画面を描画します。
 待機画面の描画完了後にプロンプトを返します。
 ホストはプロンプトが返された時に画像データ(*1)を送ることができます。
 転送が完了すると、描画を開始します。
 描画が終了すると完了通知(*2)を返し、プロンプトに戻ります。
 ※プロンプトが返された時に'+++'(*3)を入力すると「コマンドモード」に移行できます。

(2)コマンドモード …… プロンプトは '!'(21h)

プロンプト('!)が表示されたらコマンド入力が行えます。
 ?コマンドで、コマンド一覧が表示されます。
 EXITコマンドで、「通常モード」に移行できます。

(3)デモモード

モジュールは、デモ用に5画面のデータを内蔵しています。
 デモモードで起動した場合は、ホストに"DEMO MODE"と返した後、
 デモ画面を逐次描画します。
 CTRL+Z(1Ah)を受信すると「通常モード」に移行します。

*1:「画像データ形式」を参照

*2: 完了通知は、正常時は"OK"、異常時は"NG"

*3: '+' (wait) '+' (wait) '+' (CR)と入力する。waitは 100-500ms の範囲内で。

■コマンドモードで使用できるコマンド

コマンドモード時は、ホストと対話形式で任意のコマンドを実行できます。

使用できるコマンドは、

コマンド文字列 + (CR) ※(CR)は、0Dh

で実行できます。

※コマンド文字列は、大文字／小文字の区別はありません。

コマンドモードのプロンプトは、' ! ' (21h)です。

プロンプトがTX出力されると、コマンド受付ができる状態です。

コマンドの実行が完了すると、プロンプトがTX出力されます。

コマンド入力を間違えた場合は、BS(バックスペース)で1文字削除できます。

入力されたコマンド文字列が適切でない場合は、“??”がTX出力されます。

LED出力が許可されている場合は、LED2(青)点灯中はコマンド入力ができる状態を表します。

■コマンド一覧

項番	コマンド文字列	機能	備考
(1)	?	コマンド一覧の表示	
(2)	W	白画面の描画	
(3)	B	黒画面の描画	
(4)	D	指定ページの描画	
(5)	R	指定ページへの画像転送と描画	
(6)	I	システム情報のTX出力	
(7)	DEMO	デモ実行	
(8)	INTERVAL	デモのページ切り替え間隔時間を設定	
(9)	LASWAIT	デモの最終ページの表示時間を設定	
(10)	STANDBY	パワーオン・リセット時の待機画面描画の許可／禁止	
(11)	EXIT	通常モードに遷移	
(12)	RESET	MCUのリセット	
(13)	YSLAB	弊社の案内をTX出力	
(14)	BORDER	画面の枠の色変更	

■(1)コマンド: ?

コマンド一覧をTX出力します。

書式: ? (CR)

```
!?  
[Usage]  
?: Command list(this menu)      ?  
W: White screen drawing         W  
B: Black screen drawing         B  
D: Draw Image Page              D [page]  
R: Regist Image Page            R [page]  
I: System Information           I  
DEMO: Run Demo                  DEMO  
INTERVAL: Demo Interval         INTERVAL [ms]  
LASTWAIT: Demo Last page wait  LASTWAIT [ms]  
STANDBY: Enable Standby screen  STANDBY [0/1]  
EXIT: Exit Command mode        EXIT  
RESET: Reset MCU                RESET  
YSLAB: Yslab Information        YSLAB  
BORDER: Border (B/W)           BORDER [0/1]
```

■(2)コマンド: W

白画面を描画します。

書式: W(CR)

コマンド実行時

!W

コマンド終了時

OK
!

画面クリアなどに使用できます。

実行後



※画像は4.2'実行時

■(3)コマンド: B

黒画面を描画します。

書式: B(CR)

コマンド実行時

!B

コマンド終了時

OK
!

画面クリアなどに使用できます。

実行後



※画像は4.2'実行時

■(4)コマンド: D

指定されたイメージ格納ページの画像を描画します。

書式: D [page] (CR) ※参照”■イメージ格納領域”

page の範囲は、0～5。
省略時は 0 と見なします。

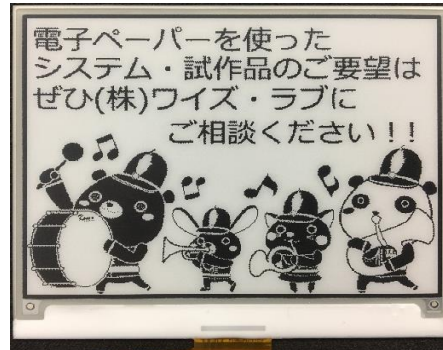
例.
コマンド実行時

```
!D 1
```

コマンド終了時

```
OK  
!
```

実行後



※画像は4.2'実行時

■(5)コマンド: R

指定されたアーカイブ領域のページに、任意の画像データを格納します。
※参照”■フラッシュの内部構造と利用され方”

書式: R [page] (CR)

page の範囲は0～5。0は待機画面で、省略時は 0 と見なします。

ホストから送出する画像データは、640x384ピクセルの16ビットマップファイルです。

コマンド実行時

```
!R 1  
Please send the BMP data(binary).
```



TeraTermで画像データを送信
してください。※下記参照

コマンド終了時

```
Registered the page 1  
!
```

【重要】

画像転送は、必ずバイナリで
行ってください。

備考: Dコマンドで同じページを描画すれば画像データが確認できます。

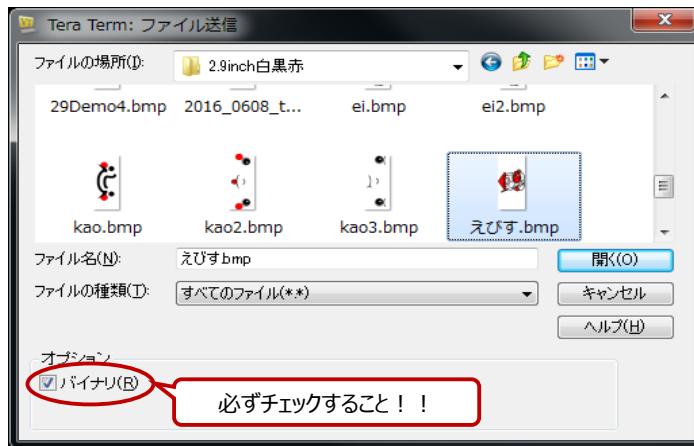
■参考:Tera TermでBmpデータを送信するには

Rコマンドを実行し、
“Please send the BMP data(binary).”と表示されたら以下を行います。

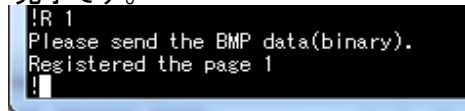
(1)「ファイル」メニューの「ファイル送信(S)」を選択 (右図)



(2)「Tera Term: ファイル送信」ダイアログで
オプションの「バイナリ(B)」をチェックする。
送信するBmpファイルを選択する。
※下図では“えびす.bmp”としています。
「開く(O)」ボタンを押す。



(4)下図のように“Registered the page #”と表示され、プロンプト(!)が表示されたら
完了です。



■(6) I

システム情報をTX出力します。

書式: I (CR)

```
!i
/**** System Information ****/
UID: 69813130080A060AC10826003800F8FE
H/W: AIMO-YCON-WEIFENG-EPD
S/W: YCON_W42_041
VCC: 3.306
Temperature: 22.8
Elapsed time: 0:26:27
EPD Drew Count : 2
EPD Name: WF0420T8PCZ15230HU76
EPD Resolution: 400 x 300
SP: SP1 (Open), SP2 (Open), SP3 (Open), SP4 (Open)
```

UID: モジュールの固有ID
H/W: 制御基板の名称
S/W: ソフトウェアのバージョン
VCC: 現在の電源電圧[Volt]
Temperature: 現在の温度[°C]
Elapsed time: モジュールの動作時間 (=通電時間)
EPD Drew Count: 描画回数(通算)
EPD Name: 電子ペーパーの名称
EPD Resolution: 電子ペーパーの画素数
SP: ショートピンの状態
UART: 通信設定

(7) DEMO

デモを実行します。

書式: **DEMO (CR)**

デモは、デモ用画像(5ページ分)の描画を繰り返し行います。

ホストからCTRL+Z(1Ah)を送出すると終了できます。
描画中は、CTRL+Zが効かないので注意してください。
デモを終了すると、プロンプト(!)がTX出力され、コマンド受付状態になります。
デモモードとの違いは、終了後もコマンドモードのままという点です。

DEMOコマンド実行時

```
!demo  
Quit CTRL+Z
```

CTRL+Zで終了すると

```
!
```

備考:

ページ間のインターバルはINTREVALコマンドで設定できます。
最終の5ページ目については、LASTWAITコマンドで更にウエイトさせることができます。
※最終ページのみインターバル+ウエイト時間になります。
これらのウエイトは、描画完了から次の描画開始までの時間です。

■(8) INTERVAL

デモのページ切り替え間隔時間を設定します。

書式: **INTERVAL [ms] (CR)**

引数無し: デモのページ切り替え間隔時間を表示します。

引数あり: デモのページ切り替え間隔時間を設定します。

引数無し: [表示]

```
!INTERVAL
Page Interval 3000 ms
!
```

引数あり: [設定] msは時間[ミリ秒]

```
!INTERVAL 1234
Page Interval 1234 ms
!
```

※最大 60000 ms です。

■(9) LASTWAIT

デモの最終ページ表示後のウエイト時間について表示/設定が行えます。

書式1: **LASTWAIT (CR)** [表示]

```
!LASTWAIT
Last Page Wait 0 ms
!
```

書式2: **LASTWAIT ##### (CR)** [設定] #####は時間[ミリ秒]

```
!LASTWAIT 1234
Last Page Wait 1234 ms
!
```

※最大 60000 ms です。

■(10) STANDBY

パワーオン・リセット時の待機画面描画の有効/無効を設定します。

書式: STANDBY [0/1] (CR)

引数無し: パワーオンリセット時に待機画面を表示する設定になっているかどうかを確認。

引数あり: パワーオン・リセット時に待機画面を描画するかどうかを設定。0は無効、1は有効。

引数無し: 待機画面表示が有効なとき [表示]

```
!STANDBY
Standby Screen Enable
!
```

引数あり: 待機画面表示を無効にする場合 [設定]

```
!STANDBY 0
Standby Screen Disable
!
```

■(11) EXIT

「コマンドモード」を終了し、「通常モード」に遷移します。

書式: EXIT (CR)

```
!EXIT
>
```

「通常モード」への遷移が完了すると、プロンプト('>')がTX出力されます。

■(12) RESET

制御基板のマイコンをハードウェアリセットします。

書式: RESET (CR)

```
!RESET
>
```

■(13) YSLAB

弊社への連絡先をTX出力します。

書式: YSLAB (CR)

```
!yslab
/=====/
YYY  YYYYYY      YSLAB INFORMATION (^-^)/
YYY  YYYY      =====
YY   YYYY      COMPANY: Y's Lab INC.
YYY  YYY  SSSSS  URL: http://www.yslab.co.jp/
YYYYY  SS   S   EMAIL: info@yslab.co.jp
YYYY   SSSS  TEL: 072-254-5109  FAX: 072-254-5209
YYYY  SS   SSS  3079-13 Nagasone-cho, Kita-ku, Sakai
YYYYY   SSSSS  Osaka, Japan      Zip code(591-8025)
```

■(14) BORDER

画面の枠(ボーダー)の色を設定します。

書式: BORDER [0/1] (CR)

引数無し: 現在の枠色の設定を表示します。

引数あり: 0 = 黒
1 = 白

引数無し: [表示]

```
!BORDER
Border Color (1)White
```

引数あり: [設定]

```
!BORDER 0
Border Color (0)Black
```

BORDER 0 時に白画面(Wコマンド)実行



BORDER 1 時に黒画面(Bコマンド)実行



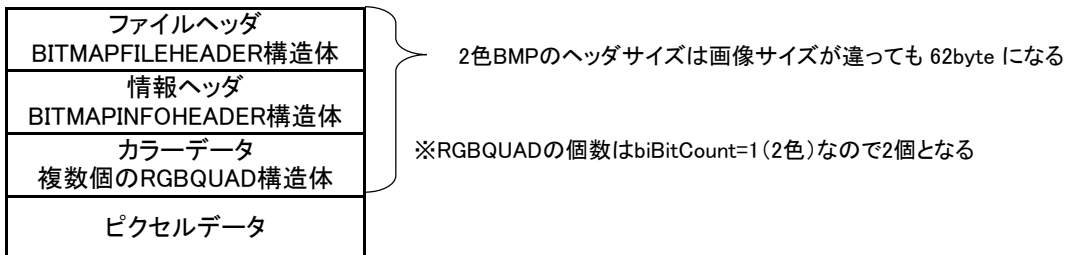
※画像は4.2'実行時

■画像データ形式

各パネルのBMPデータ

パネル種別	幅(pixel)	高さ(pixel)	ビットの深さ	BMP画像サイズ(byte)
4.2"	400	300	1	15,662
7.5"	640	384	1	30,782

■BMP形式



```
typedef struct tagBITMAPFILEHEADER {
    WORD    bfType;        // 常に"BM"
    DWORD   bfSize;       // ファイルサイズ
    WORD    bfReserved1;  // 0に設定
    WORD    bfReserved2;  // 0に設定
    DWORD   bfOffBits;    // ファイル先頭からピクセルデータ先頭までのオフセット
} BITMAPFILEHEADER;
```

```
typedef struct tagBITMAPINFOHEADER {
    DWORD   biSize;       // この構造体のサイズ
    DWORD   biWidth;     // 幅 (ピクセル単位)
    DWORD   biHeight;    // 高さ (ピクセル単位)
    WORD    biPlanes;    // 常に1
    WORD    biBitCount;   // 1ピクセルあたりのカラービットの数
    DWORD   biCompression; // BI_RGB, BI_RLE8, BI_RLE4のいずれか
    DWORD   biSizeImage; // イメージの全バイト数
    DWORD   biXPelsPerMeter; // 0または水平解像度
    DWORD   biYPelsPerMeter; // 0または垂直解像度
    DWORD   biClrUsed;    // 通常は0、biBitCount以下のカラー数に設定可能
    DWORD   biClrImportant; // 通常は0
} BITMAPINFOHEADER;
```

```
typedef struct tagRGBQUAD {
    BYTE    rgbBlue;     // 青の輝度(0~255)
    BYTE    rgbGreen;    // 緑の輝度(0~255)
    BYTE    rgbRed;     // 赤の輝度(0~255)
    BYTE    rgbReserved; // 予約値。0にする
} RGBQUAD
```

■ 適応するBMP形式の詳細例(2.9")

offset	00	01	02	03	04	05			
0000:	42	4D					※fType	"BM"	
0002:	2E	3D	00	00			※fSize	1 5 6 6 2 バイト固定	
0006:	00	00					※fReserved1	0 固定	
0008:	00	00					※fReserved2	0 固定	
000A:	3E	00	00	00			※f0ffBits	6 2 バイト目固定	
000E:	28	00	00	00			※iSize	4 0 バイト固定	
0012:	90	01	00	00			※iWidth	4 0 0 画素固定	
0016:	2C	01	00	00			※iHeight	3 0 0 ライン固定	
001A:	01	00					※iPlanes	1 固定	
001C:	01	00					※iBitCount	1 (2 値) 固定	
001E:	00	00	00	00			※iCompression	0 (BI_RGB : 無圧縮) 固定	
0022:	F0	3C	00	00			※iSizeImage	1 5 6 0 0 バイト固定	
0026:	00	00	00	00			※iXPelsPerMeter	0 固定	
002A:	00	00	00	00			※iYPelsPerMeter	0 固定	
002E:	00	00	00	00			※iClrUsed	0 固定	
0032:	00	00	00	00			※iClrImportant	0 固定	
0036:	00						※gbBlue	0 固定 or 2 5 5 固定	
0037:	00						※gbGreen	0 固定 or 2 5 5 固定	
0038:	00						※gbRed	0 固定 or 2 5 5 固定	
0039:	00						※gbReserved	0 固定	
003A:	FF						※gbBlue	2 5 5 固定 or 0 固定	
003B:	FF						※gbGreen	2 5 5 固定 or 0 固定	
003C:	FF						※gbRed	2 5 5 固定 or 0 固定	
003D:	00						※gbReserved	0 固定	
003E:	3 0 0 ライン目 (5 0 バイト+2 バイト)						00	00	00
005A:	2 9 9 ライン目 (5 0 バイト+2 バイト)						00	00	00
0076:	2 9 8 ライン目 (5 0 バイト+2 バイト)						00	00	00
	⋮								
3C92:	3 ライン目 (5 0 バイト+2 バイト)						00	00	00
3C06:	2 ライン目 (5 0 バイト+2 バイト)						00	00	00
3CFA:	1 ライン目 (5 0 バイト+2 バイト)						00	00	00

- 部は、全て固定値になるはずですが。
- 部は、ピクセルデータなので画像により変わります。