

## 汎用 LCD コントローラ説明書

- ・ 汎用チェッカー（タイプD・E）
- ・ 汎用ボード
- ・ 汎用コントローラソフト Version4.01

第一版 2008/04/15

第二版 2008/06/17

製造元 フジ産業株式会社

〒101-0023 東京都千代田区神田松永町1-8 相沢ビル2F

TEL: 03 - 3255 - 3404

FAX: 03 - 3251 - 0769

=====	
1.概要.....	3
1.1 DタイプとEタイプチェッカーの違いについて.....	4
2.ブロック図、電氣的仕様.....	5
2.1 汎用基板タイプのブロック図.....	5
2.2 汎用チェッカーDタイプのブロック図.....	6
2.3 汎用チェッカーEタイプのブロック図.....	7
3.各種コネクタのピン配.....	8
3.1 出力コネクタ(34ピン).....	8
3.2 出力コネクタ(40ピン).....	9
3.3 シリアルポート(D-Sub 9ピン).....	10
3.4 USBポート(Bタイプコネクタ).....	10
3.5 外部測定器用コネクタ(8ピン：Dチェッカー).....	11
3.6 外部測定器用コネクタ(12ピン：Eチェッカー).....	11
3.7 外部SW用コネクタ(4ピンモジュラージャック：Eチェッカー).....	11
4.動作タイミング(汎用基板タイプ、Dチェッカータイプ).....	12
4.1 パラレルIF動作タイミング.....	12
4.2 シリアルIF動作タイミング.....	13
5.動作タイミング(Eチェッカータイプ).....	14
5.1 パラレルIF動作タイミング.....	14
5.2 シリアルIF動作タイミング.....	15
5.3 I2C動作タイミング.....	15
5.4 DigitalRGB動作タイミング.....	16
5.5 FLM方式(STN)動作タイミング.....	17
5.6 Driver方式動作タイミング.....	19
6.汎用コントローラのシーケンス動作について.....	20
6.1 メモリマップ.....	20
6.2 動作シーケンス.....	21
7.初期設定(LCDファイル)で使えるコマンド.....	22
7.1 コメント.....	22
7.2 MOTファイルの設定用コマンド.....	22
7.3 チェッカーの設定用コマンド.....	23
7.4 各入出力の設定用コマンド.....	25
7.5 LCDファイル作成例.....	28
8.ソフトウェアのインストールについて.....	29
8.1 ソフトウェアに必要なシステム.....	29
8.2 ソフトウェアのインストールについて.....	29
8.3 USBドライバーのインストールについて.....	30
8.4 ソフトウェアのアンインストールについて.....	31
9.ソフトウェアの使用法.....	32
9.1 メイン画面.....	32
9.2 設定画面.....	35
9.3 画面の登録.....	43
9.4 現在の画面.....	51
10.PCソフトウェアのVersionについて.....	52
11.チェッカーのBiosVersionについて.....	56
=====	

## 1. 概要

小・中型LCDを表示制御するには、さまざまな初期設定方法・インターフェイス方式があります。このコントローラは、初期設定・インターフェイスをユーザーが自由に換えられるようにPCから設定・変更が可能な汎用LCDコントローラです。

PCとの接続はRS232C又はUSBで通信可能、各種設定のテスト動作やデバッグが簡単に行え表示させたい画面もBMPファイル等で転送すれば簡単に表示画面を換えられます。

最終的にはコントローラ内蔵のフラッシュメモリに全データを保存できるので、デモ機としてのコントローラ・エージング用としてのコントローラなど利用が可能です。

汎用LCDコントローラは、現在3種類用意しています。

汎用基板タイプ(基板単体のコントローラでデモ機用として最適、PCとはRS232C又はUSB通信)

汎用チェッカーDタイプ(アルミケースに覆われた検査機用として最適、PCとはRS232C通信のみ)

汎用チェッカーEタイプ(アルミケースに覆われた検査機用として最適、PCとはRS232C又はUSB通信)

### 出力可能なインターフェイス

低速SPI(CLK=約300kHz)	.....対応可能
パラレル4ビット(68系、80系)	.....対応可能
パラレル8ビット(68系、80系)	.....対応可能
パラレル16ビット(68系、80系)	.....対応可能
パラレル20ビット(68系、80系)	.....対応可能
高速SPI(CLK=12・6・3・1.5MHzの4段階)	.....対応可能
高速SPI(CLK=96kHz~24MHzの256段階)	.....対応可能
I2C(CLK=96kHz~24MHzの256段階)	.....対応可能
DigitalRGB(CLK=750Hz~49MHzの65k段階)	.....対応可能
FLM方式(STN用LCD)	.....対応可能

### 登録可能な表示画面

Windows系のグラフィックソフトでBMP・JPEGファイルで保存したものが登録できます。フラッシュメモリの容量によりますが16×16dotsから2048×2048dotsまで登録可能です。但し登録可能な色数は以下の通り

モノクロ2値	.....色深度1bpp
モノクロ4階調	.....色深度2bpp
モノクロ16階調	.....色深度4bpp
16カラー	.....色深度4bpp
256カラー	.....色深度8bpp R:G:B=3:3:2bit
65kカラー	.....色深度16bpp R:G:B=5:6:5bit
フルカラー	.....色深度24bpp R:G:B=8:8:8bit

### 初期設定用のコマンド

Windows系のエディタソフト(メモ帳等)でLCDファイルとして保存したものが登録できます。最大1500コマンド登録が可能です。

VDD,VEE 電圧・電流リミッタの設定

インターフェイス方法の設定

各出力端子のHigh,Low設定が可能

各種インターフェイスによるコマンドデータ等の出力・入力

Wait時間による出力時間待ちの設定

繰り返しの設定(for文に近い使い方)

## 1.1 DタイプとEタイプチェッカーの違いについて

出力端子 (Dタイプは34ピンコネクタ、Eタイプは40ピンコネクタ)

ただし1~34まではピン配互換

出力電源の違い

- ・Dタイプは2電源出力  
VDD=1.50~5.00V、電流 max 50mA  
VEE=0.00~19.00V、電流 max 500mA  
2電源同時ON、又は同時OFFの対応のみ
- ・Eタイプは3電源出力  
VDD=1.50~5.00V、電流 max 800mA  
VEE=1.50~29.00V、電流 max 800mA  
VBB=1.50~29.00V、電流 max 800mA  
3電源同時ON/OFF、または時間差ON/OFFが可能

信号出力の違い

- ・Dタイプの場合 (波形タイミング固定又は数段階可変のみ)  
I80系又はM68系 I/F (4/8/16/20bit)  
SPI出力のみ (8/9/10/11bit)
- ・Eタイプの場合 (波形タイミングは各端子256段階可変可能)  
I80系又はM68系 I/F (4/8/16/20bit)  
SPI入出力対応 (7-16bit)  
I2C入出力対応  
DigitalRGB 出力対応  
FLM方式(STN)出力対応

Dタイプの場合、信号端子同士ショートによりバッファICが壊れることがありますが

Eタイプの場合、バッファに電流検出回路を搭載してエラー検知します

PCとの接続

- ・DタイプはRS232Cのみ(38400bps)
- ・EタイプはRS232C又はUSB(2.0規格)

内部処理について

DタイプはH8/300H、EタイプはH8S/2600を使用

周辺LSIも高速化したので、Eタイプは処理速度2~4倍早くなりました

外部SWの対応

Eタイプはモジュージャックを追加して、外部SWボックスと接続できるように対応

外部SWボックスは、Auto/Manual、Step、On/OFFのスイッチ3個とロータリエンコーダ2個搭載のSWボックスになります

BiosVersionUPについて

チェッカーのバグや機能追加により内部ソフトのVersionUPをする場合、

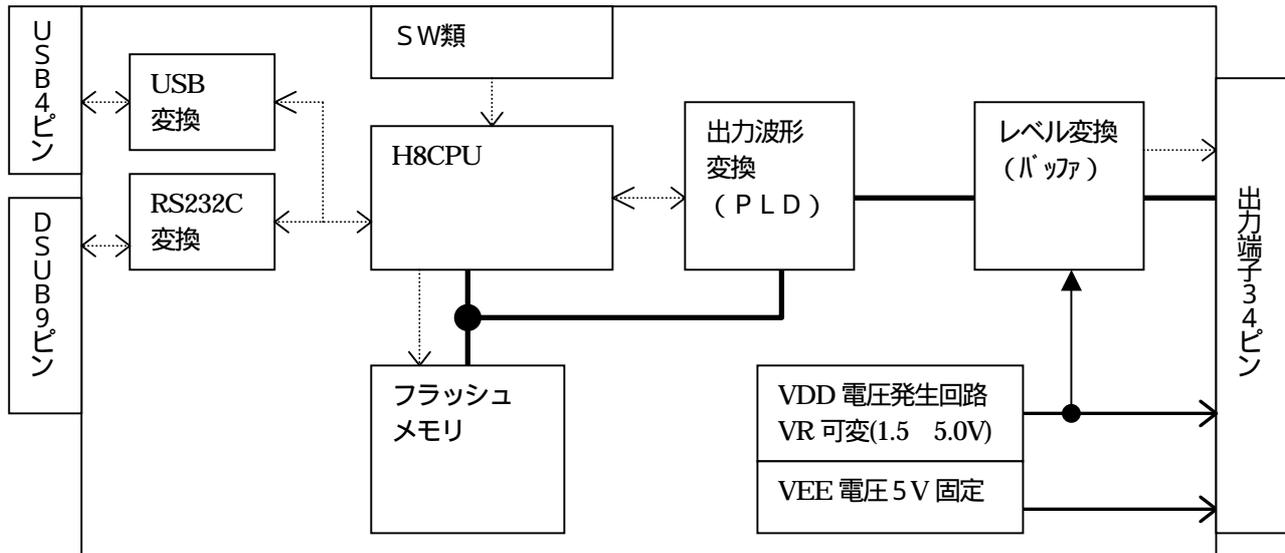
Dタイプは弊社に返品してもらってVersionUPして再出荷でしたが (専用の書込み器必要だった)

EタイプはメモリーカードにUpdateプログラム(DATファイル)を登録すれば

VersionUP可能になりました。(DATファイルをmail等で添付すれば御社でVersionUP可能)

## 2. ブロック図、電気的仕様

### 2.1 汎用基板タイプのブロック図



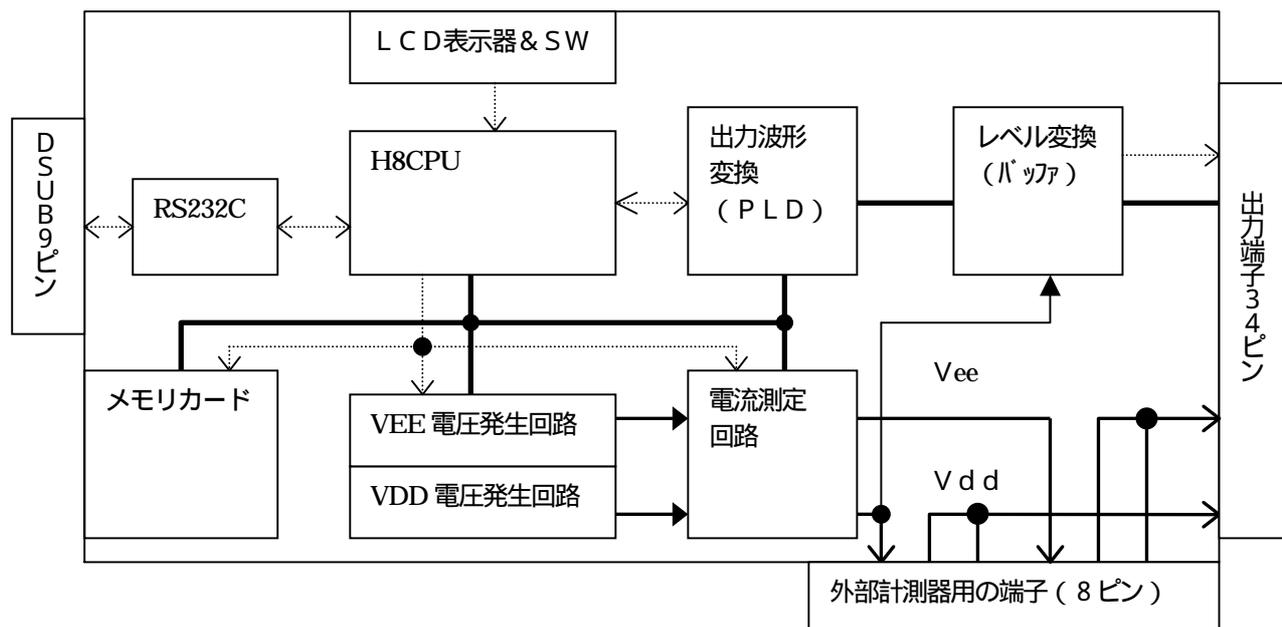
#### SW 類の機能

- ・ MAIN VR : 検査モジュールのコントラスト調整用 (ロータリエンコーダ方式)
- ・ SUB VR : 検査モジュールのコントラスト調整用 (サブ用、ロータリエンコーダ方式)
- ・ Auto/ManualSW : 表示画面切替用、自動/手動SW
- ・ STEP SW : 検査モジュールの表示画面切り替えSW
- ・ BUS SW : データバス(DB0-15)にのみ 6 MHz ぐらいのランダム波形を出力するSW
  
- ・ 出力端子 : 検査モジュールとの接続端子 (34ピン)
- ・ DSUB 9 ピン : RS232C ストレートケーブルによりパソコンと接続するコネクタ
- ・ USB 4 ピン : パソコンと接続する USB Btype コネクタ

#### 汎用基板タイプの電気的仕様

項目	記号	Min	Max	単位	
入力電源電圧	VCC	DC 5.0	DC 5.5	V	ACアダプタにより入力
入力電源電流	ICC	200.0	700.0	mA	
出力電源電圧1	VDD	1.50	5.00	V	VRにより変更
出力電源電流1	IDD	0.00	300.00	mA	
出力電源電圧2	VEE	5.0	5.0	V	5V固定出力
出力電源電流2	IEE	0.00	500.0	mA	
高レベル出力電圧	Voh	Vdd - 0.1 Vdd - 0.6		V V	Ioh = - 100 μA Ioh = - 12mA
低レベル出力電圧	Vol		0.1 0.4	V V	Iol = 100 μA Iol = 12mA
高レベル入力電圧	Vih	2.0		V	
低レベル入力電圧	Vil		0.8	V	

## 2.2 汎用チェッカーDタイプのブロック図



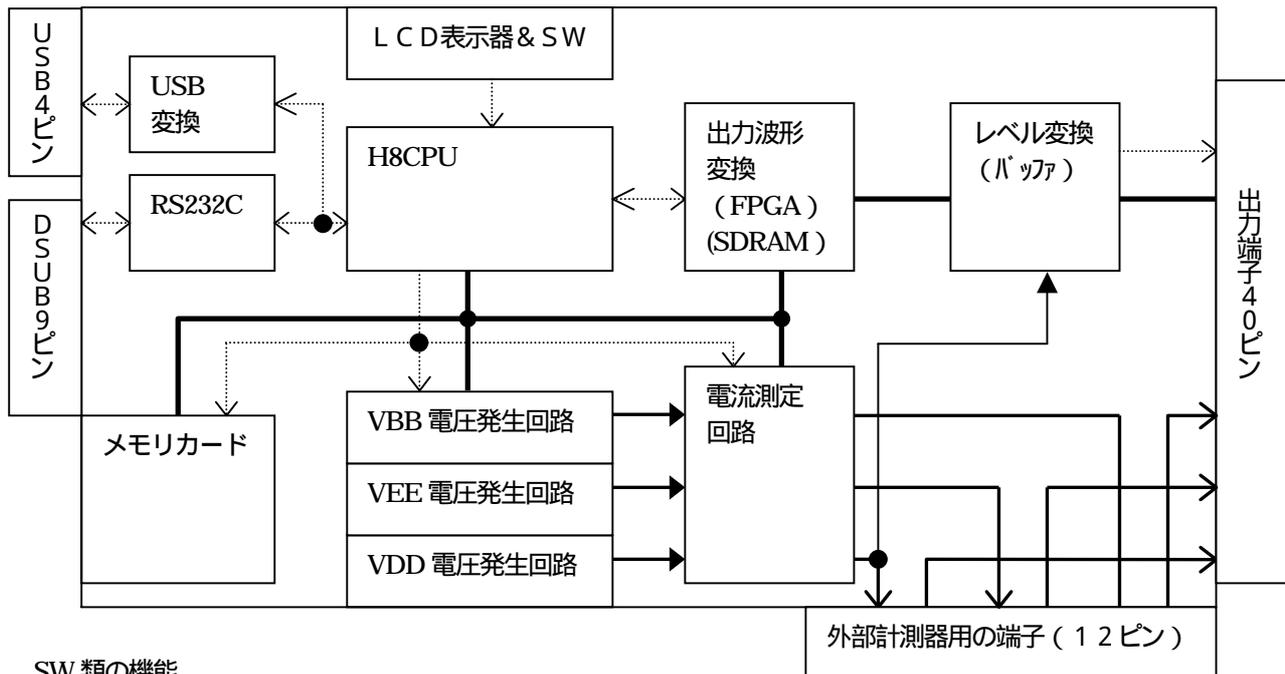
### SW 類の機能

- ・ LCD 表示器 : 20 × 4 行のキャラクターLCD、現在の動作や電流値が表示される。
- ・ MAIN VR : 検査モジュールのコントラスト調整用 (ロータリエンコーダ方式)
- ・ SUB VR : 検査モジュールのコントラスト調整用 (サブ用、ロータリエンコーダ方式)
- ・ ONOFF SW : 出力端子 34 ピンの電源出力 ON / OFF 切り替え SW (パワーダウンプロテクション機能)
- ・ BUS SW : データバス (DB0-15) にのみ 6 MHz ぐらいのランダム波形を出力する SW
- ・ STEP SW : 検査モジュールの表示画面切り替え SW
  
- ・ 出力端子 : 検査モジュールとの接続端子 (34 ピン)
- ・ DSUB9 ピン : RS232C ストレートケーブルによりパソコンと接続するコネクタ
- ・ メモリーカード : 検査モジュール用の各種設定や表示画面を保存しておく Flash Memory
- ・ 外部計測器 : VDD・VEE 電圧・各電流値を外部計測器でも測定可能な端子

### 汎用チェッカーDタイプの電氣的仕様

項目	記号	Min	Max	単位	
入力電源電圧	ACC	AC 85	AC 264	V	ワイド入力
入力電源電流	ICC		400	mA	1.5W電源
出力電源電圧 1	VDD	1.50	5.00	V	設定分解能 0.01V
出力電源電流 1	IDD	0.00	50.00	mA	電流計分解能 0.05mA
出力電源電圧 2	VEE	0.0	19.0	V	設定分解能 0.01V
出力電源電流 2	IEE	0.00	200.0	mA	電流計分解能 0.2mA
高レベル出力電圧	Voh	Vdd - 0.1 Vdd - 0.6		V	Ioh = - 100μA Ioh = - 12mA
低レベル出力電圧	Vol		0.1 0.4	V	Iol = 100μA Iol = 12mA
高レベル入力電圧	Vih	2.0		V	
低レベル入力電圧	Vil		0.8	V	

### 2.3 汎用チェッカーEタイプのブロック図



#### SW 類の機能

- ・ LCD 表示器 : 20 × 4 行のキャラクターLCD、現在の動作や電流値が表示される。
- ・ MAIN VR : 検査モジュールのコントラスト調整用 (ロータリエンコーダ方式)
- ・ SUB VR : 検査モジュールのコントラスト調整用 (サブ用、ロータリエンコーダ方式)
- ・ Auto/ManualSW : 表示画面切替用、自動 / 手動 SW
- ・ ONOFF SW : 出力端子 40 ピンの電源出力 ON / OFF 切り替え SW (パワーダウンプロテクション機能)
- ・ BUS SW : 現在未使用
- ・ STEP SW : 検査モジュールの表示画面切り替え SW
  
- ・ 出力端子 : 検査モジュールとの接続端子 (40 ピン)
- ・ DSUB 9 ピン : RS232C ストレートケーブルによりパソコンと接続するコネクタ
- ・ メモリーカード : 検査モジュール用の各種設定や表示画面を保存しておく Flash Memory
- ・ 外部計測器 : VDD・VEE・VBB 電圧・各電流値を外部計測器でも測定可能な端子
- ・ USB 4 ピン : パソコンと接続する USB Btype コネクタ

#### 汎用チェッカーEタイプの電氣的仕様

項目	記号	Min	Max	単位	
入力電源電圧	ACC	AC 85	AC 264	V	ワイド入力
入力電源電流	ICC		1200	mA	50W電源
出力電源電圧1	VDD	1.50	5.00	V	設定分解能 0.01V
出力電源電流1	IDD	0.00	800.00	mA	電流計分解能 0.012mA
出力電源電圧2	VEE	1.50	29.0	V	設定分解能 0.01V
出力電源電流2	IEE	0.00	800.0	mA	電流計分解能 0.012mA
出力電源電圧3	VBB	1.50	29.0	V	設定分解能 0.01V
出力電源電流3	IBB	0.00	800.0	mA	電流計分解能 0.012mA
高レベル出力電圧	Voh	Vdd - 0.1 Vdd - 0.6		V	Ioh = - 100 μA Ioh = - 12mA
低レベル出力電圧	Vol		0.1 0.4	V	Iol = 100 μA Iol = 12mA
高レベル入力電圧	Vih	2.0		V	
低レベル入力電圧	Vil		0.8	V	

### 3. 各種コネクタのピン配

#### 3.1 出力コネクタ(34ピン)

使用するLCDモジュールと接続するコネクタで、信号のHiレベルは全てVDD電圧です  
汎用基板タイプ、Dチェッカータイプ

ピンNo.	信号名	内容
1	VDD	電源出力(+1.5V~+5.0V)
2	GND	グラウンド
3	A0	出力ポート(シリアルモード時DOUT)
4	A1	出力ポート(シリアルモード時CK)
5	A2	出力ポート(シリアルモード時CS)
6	RESET	出力RESET端子
7	RD/E	出力READ(80系)/E(68系)
8	WR/RW	出力WRITE(80系)/RW(68系)
9	E/CLK	出力E端子(68系)/CLKOUT
10	CS	出力CS端子
11	GND	グラウンド
12	GND	グラウンド
13	DB0	双方向データバス
14	DB1	双方向データバス
15	DB2	双方向データバス
16	DB3	双方向データバス
17	DB4	双方向データバス
18	DB5	双方向データバス
19	DB6	双方向データバス
20	DB7	双方向データバス
21	DB8	双方向データバス
22	DB9	双方向データバス
23	DB10	双方向データバス
24	DB11	双方向データバス
25	DB12	双方向データバス
26	DB13	双方向データバス
27	DB14	双方向データバス
28	DB15	双方向データバス
29	GND	グラウンド
30	VEE	電源出力(0~+19.0V、但し汎用基板は5V固定)
31	DB16/SOUT	双方向データバス/高速シリアルSOUT
32	DB17/SCLK	双方向データバス/高速シリアルSCLK
33	DB18/SCS	双方向データバス/高速シリアルSCS
34	DB19/SRS	双方向データバス/高速シリアルSRS

注1) ピンNo31~34端子について

20ビットI/Fの場合、DB16~19として動作

その他は高速シリアル通信用として動作する。

CLK周波数は1.5MHz・3MHz・6MHz・12MHzの切り替えができ、転送ビットも8・9・10・11bitと4段階の転送フォーマットが設定できます。

注2) 双方向データバスについて

- ・20ビット制御の場合は、DB0(LSB) - DB19(MSB)を使用する
- ・16ビット制御の場合は、DB0(LSB) - DB15(MSB)を使用する
- ・8ビット制御の場合は、DB0(LSB) - DB7(MSB)を使用する
- ・4ビット制御の場合は、上位・下位の順の時DB4(LSB) - DB7(MSB)を使用する  
下位・上位の順の時DB0(LSB) - DB3(MSB)を使用する

### 3.2 出力コネクタ(40ピン)

使用するLCDモジュールと接続するコネクタで、信号のHiレベルは全てVDD電圧です

#### Eチェッカータイプ

ピンNo.	信号名	内容
1	VDD	電源出力(+1.5V~+5.0V)
2	GND	グラウンド
3	A0 / VSH / STV	出力ポート A0 / DigitalRGB の VSH or STV(DRV=1)
4	A1 / HSD / CPV	出力ポート A1 / DigitalRGB の HSD or CPV(DRV=1)
5	A2 / DEN / STH	出力ポート A2 / DigitalRGB の DEN or STH(DRV=1)
6	RESET	出力 RESET 端子
7	RD / E / VCOM	出力 RD(80系) / E(68系) / DigitalRGB の VCOM(DRV=1)
8	WR / RW / OE	出力 WR(80系) / RW(68系) / DigitalRGB の OE(DRV=1)
9	E / CLK	出力 E 端子(68系) / CLKOUT
10	CS / LD	出力 CS 端子 / DigitalRGB の LD(DRV=1)
11	GND	グラウンド
12	GND	グラウンド
13	DB0	双方向データバス
14	DB1	双方向データバス
15	DB2	双方向データバス
16	DB3	双方向データバス
17	DB4	双方向データバス
18	DB5	双方向データバス
19	DB6	双方向データバス
20	DB7	双方向データバス
21	DB8	双方向データバス
22	DB9	双方向データバス
23	DB10	双方向データバス
24	DB11	双方向データバス
25	DB12	双方向データバス
26	DB13	双方向データバス
27	DB14	双方向データバス
28	DB15	双方向データバス
29	GND	グラウンド
30	VEE	電源出力(+1.5V~+29.0V)
31	DB16 / SOUT	双方向データバス / SPI の SOUT
32	DB17 / SCLK / SCL	双方向データバス / SPI の CLK / I2C の SCL
33	DB18 / SCS	双方向データバス / SPI の CS
34	DB19 / SRS	双方向データバス / SPI の RS
35	Sdin/Busy/SDA/POT	入力 SPI の SIN / Pararel の Busy / I2C の SDA / DigitalRGB の POT
36	GND	グラウンド
37	VBB	電源出力(+1.5V~+29.0V)
38	VBB	電源出力(+1.5V~+29.0V)
39	VEE	電源出力(+1.5V~+29.0V)
40	VDD	電源出力(+1.5V~+5.0V)

注1) ピンNo31~34端子について

20ビットI/Fの場合、DB16~19として動作

その他は高速シリアル通信として動作する。

CLK周波数は96kHz~24MHzの256段階切り替えができ、転送ビットも

7~16bitと転送フォーマットが設定できます。

注2) 双方向データバスについて

- ・ 20ビット制御の場合は、DB0(LSB) - DB19(MSB)を使用する
- ・ 16ビット制御の場合は、DB0(LSB) - DB15(MSB)を使用する
- ・ 8ビット制御の場合は、DB0(LSB) - DB7(MSB)を使用する
- ・ 4ビット制御の場合は、上位・下位の順の時DB4(LSB) - DB7(MSB)を使用する  
下位・上位の順の時DB0(LSB) - DB3(MSB)を使用する

注3) ピンNo3～5端子について

I/Fに必要な端子を追加したいときに用意されたPort出力端子です。(A0,A1,A2端子)

例 使用LSIにRS端子があればA0をRSに割り当てて対応する等

I/FをDigitalRGBに設定した場合、VSH、HSD、DEN出力端子として機能します

更にDRVビット=1の場合、STV、CPV、STH出力端子として機能します

(DRV=1によりピンNo7, 8, 10もVCOM、OE、LD出力端子になる)

注4) ピンNo3～5端子について

I/Fの設定により機能が変わります

Sdin: SPI用のSdin入力端子として機能します

Busy: パラレル用Busy入力端子として機能します

SDA: I2C用のSDA入出力端子として機能します

POT: DigitalRGB用のPOT入力端子として機能します

### 3.3 シリアルポート(D-Sub9ピン)

パソコンと接続するコネクタで、ストレートケーブルを使用する。

汎用基板タイプ、Dチェッカータイプ、Eチェッカータイプ

ピンNo.	信号名
1	NC
2	RXD 受信データ(in)
3	TXD 送信データ(out)
4	NC
5	GND 信号グランド
6	NC
7	CTS 送信可(in)
8	RTS 送信要求(out)
9	NC

### 3.4 USBポート(Bタイプコネクタ)

パソコンとUSBで接続するコネクタです。(USB2.0仕様に対応し最大480Mbps)

汎用基板タイプ、Eチェッカータイプ

ピンNo.	信号名
1	+5V
2	-DATA
3	+DATA
4	GND

### 3.5 外部測定器用コネクタ(8ピン：Dチェッカー)

#### Dチェッカータイプ

ピンNo	内容	
1	外部VEE 電流計用 - 端子	(注意)未使用時は1, 2をショート
2	外部VEE 電流計用 + 端子	
3	外部VEE 電圧計用 - 端子	
4	外部VEE 電圧計用 + 端子	
5	外部VDD 電流計用 - 端子	(注意)未使用時は5, 6をショート
6	外部VDD 電流計用 + 端子	
7	外部VDD 電圧計用 - 端子	
8	外部VDD 電圧計用 + 端子	

外部電圧計・電流計を付けたい場合使用する端子

### 3.6 外部測定器用コネクタ(12ピン：Eチェッカー)

#### Eチェッカータイプ

ピンNo	内容	
1	外部VBB 電流計用 - 端子	(注意)未使用時は1, 2をショート
2	外部VBB 電流計用 + 端子	
3	外部VBB 電圧計用 - 端子	
4	外部VBB 電圧計用 + 端子	
5	外部VEE 電流計用 - 端子	(注意)未使用時は5, 6をショート
6	外部VEE 電流計用 + 端子	
7	外部VEE 電圧計用 - 端子	
8	外部VEE 電圧計用 + 端子	
9	外部VDD 電流計用 - 端子	(注意)未使用時は9, 10をショート
10	外部VDD 電流計用 + 端子	
11	外部VDD 電圧計用 - 端子	
12	外部VDD 電圧計用 + 端子	

外部電圧計・電流計を付けたい場合使用する端子

### 3.7 外部SW用コネクタ(4ピンモジュラージャック：Eチェッカー)

外部SWと接続するコネクタです。(ON/OFF SW、STEP SW、ロータリエンコーダ等)

#### Eチェッカータイプ

ピンNo.	信号名
1	+5V (max300mA)
2	TXD 送信データ(out)
3	RXD 受信データ(in)
4	GND

#### 4. 動作タイミング(汎用基板タイプ、Dチェッカータイプ)

##### 4.1 パラレルIF動作タイミング

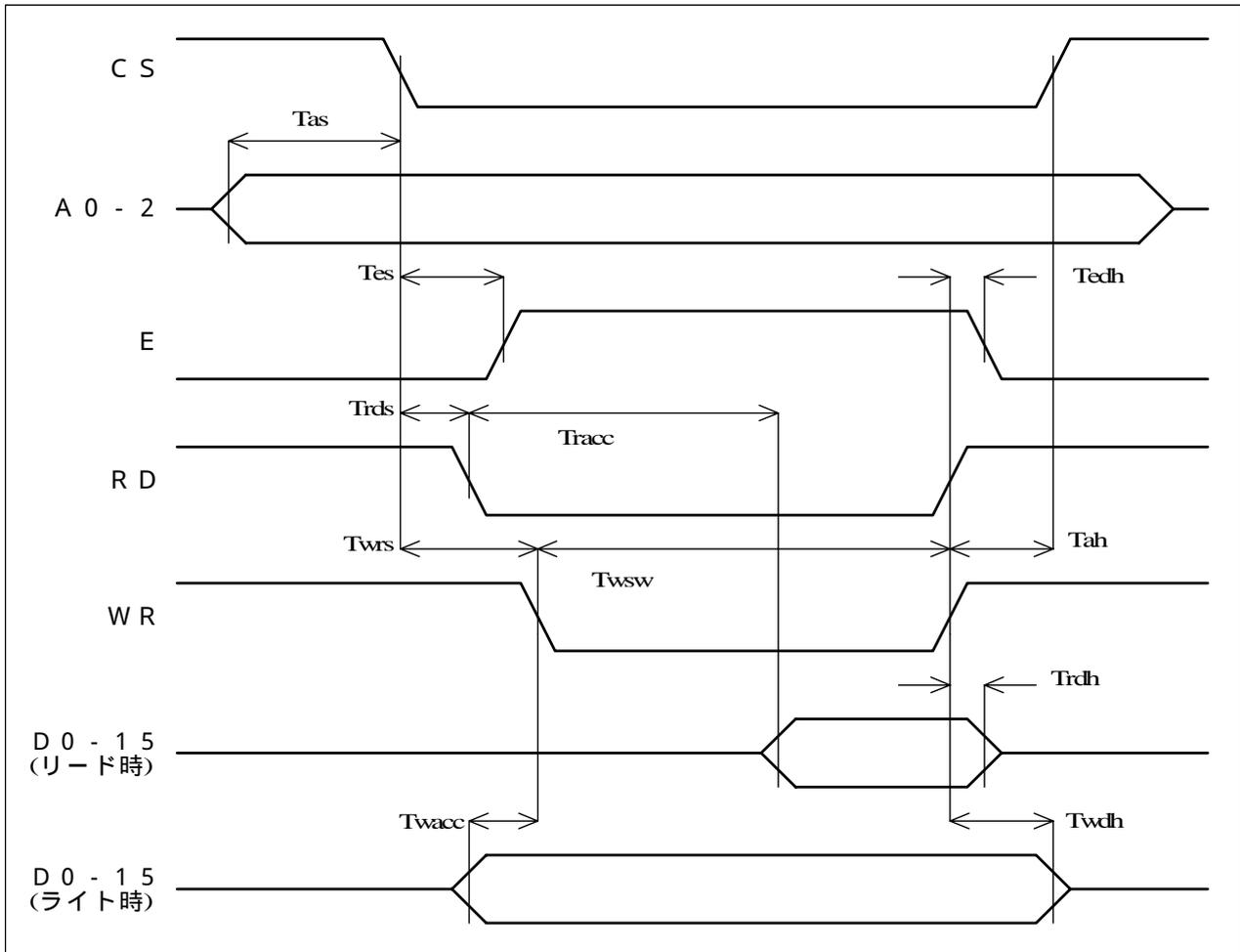


図 パラレルIF

パラレルI/F タイミング表

項目	記号	min	max	単位
Address セットアップ時間	$T_{as}$	1500	-	nS
E セットアップ時間	$T_{es}$	10	33	nS
E ホールド時間	$T_{edh}$	10	33	nS
RD セットアップ時間	$T_{rds}$	20	-	nS
リードデータアクセス時間	$T_{racc}$	$40 + (n \times A)$	-	nS
WR セットアップ時間	$T_{wrs}$	46	-	nS
ライトデータストロブパルス幅	$T_{wsw}$	$40 + (n \times A)$	-	nS
Address ホールド時間	$T_{ah}$	10	-	nS
リードデータホールド時間	$T_{rdh}$	0	-	nS
ライトデータセットアップ時間	$T_{wacc}$	25	-	nS
ライトデータホールド時間	$T_{wdh}$	60	-	nS

注)  $n = \text{DPSW}$  の値(1 - 15 まで設定可能)

$A = 40.7 \text{ nS}$

#### 4.2 シリアル I/F 動作タイミング

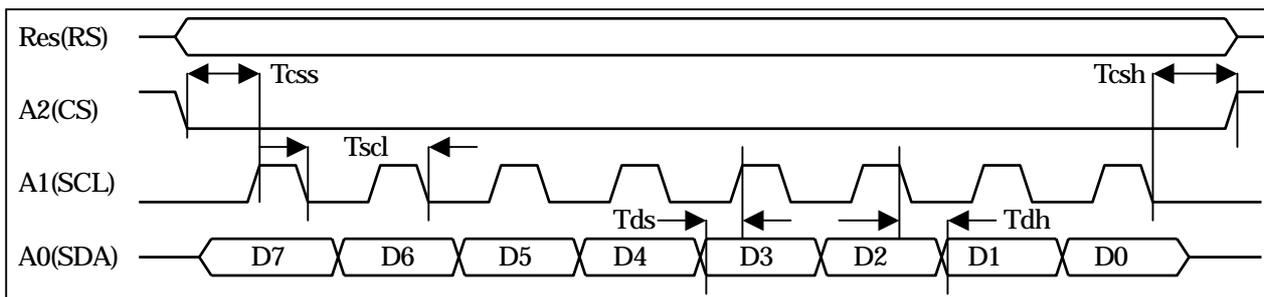


図 低速SPI (8bit 固定)

低速SPI タイミング表

項目	記号	min	max	単位
SCL 周波数	TscL	300		kHz
SCL ハイレベル時間	TscL_Lo	2.0	2.2	μS
SCL ロウレベル時間	TscL_Hi	1.2	1.4	μS
SDA セットアップ時間	Tds	0.8	1.0	μS
SDA ホールド時間	Tdh	0.8	1.0	μS
CS セットアップ時間	Tcss	1.9	2.1	μS
CS ホールド時間	Tcsh	0.8	1.0	μS

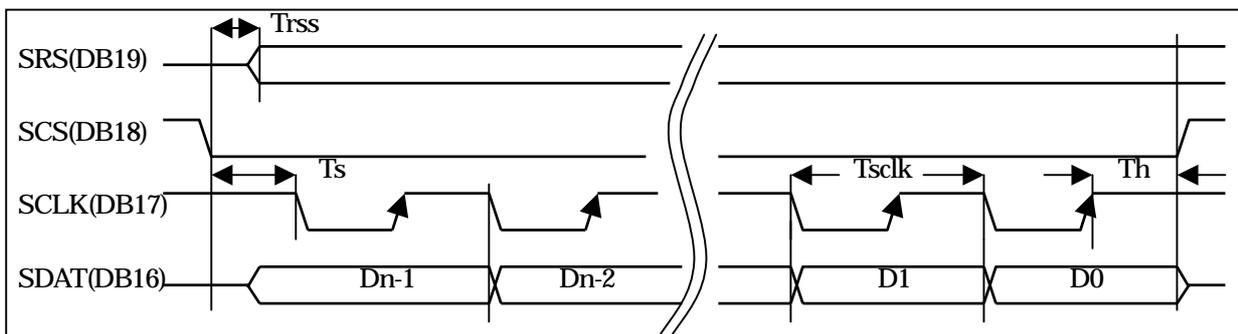


図 高速SPI (8 - 11 bit)

高速SPI タイミング表

項目	記号	min	max	単位
SCLK 周波数	TscLk	a		MHz
SCLK ハイレベル時間	TscLk_Lo	b	b + 5	nS
SCLK ロウレベル時間	TscLk_Hi	b	b + 5	nS
SCLK セットアップ時間	Ts	122	127	nS
SCLK ホールド時間	Th	b	b + 5	nS
SRS セットアップ時間	Trss	76	81	nS

注) 周波数 4 段階 (n = 0 - 3 まで 0=12MHz、1=6MHz、2=3MHz、3=1.5MHz)

EXPMODE 命令により、SCLK 論理反転も可能

$$a = 12.288\text{MHz} / 2^n \cdots 12.288\text{MHz}, 6.144\text{MHz}, 3.072\text{MHz}, 1.536\text{MHz}$$

$$b = 2^n / 24.576\text{MHz} \cdots 40\text{ns}, 81\text{ns}, 162\text{ns}, 325\text{ns}$$

## 5. 動作タイミング(E チェッカータイプ)

### 5.1 パラレル I/F 動作タイミング

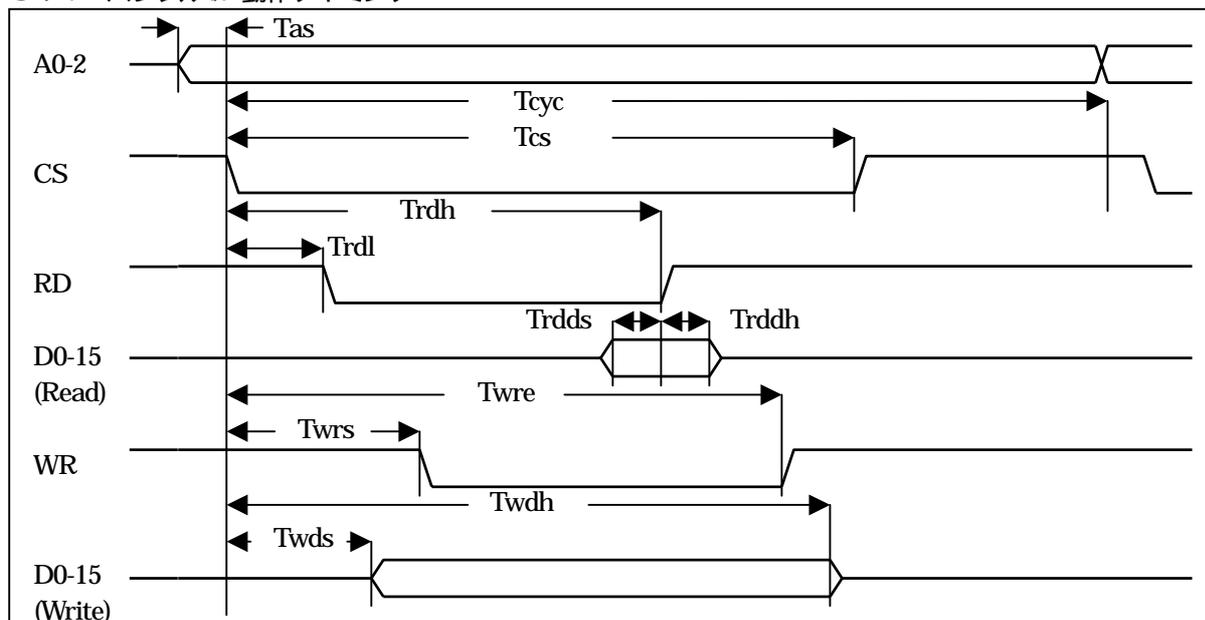


図 パラレル I/F (80 系)

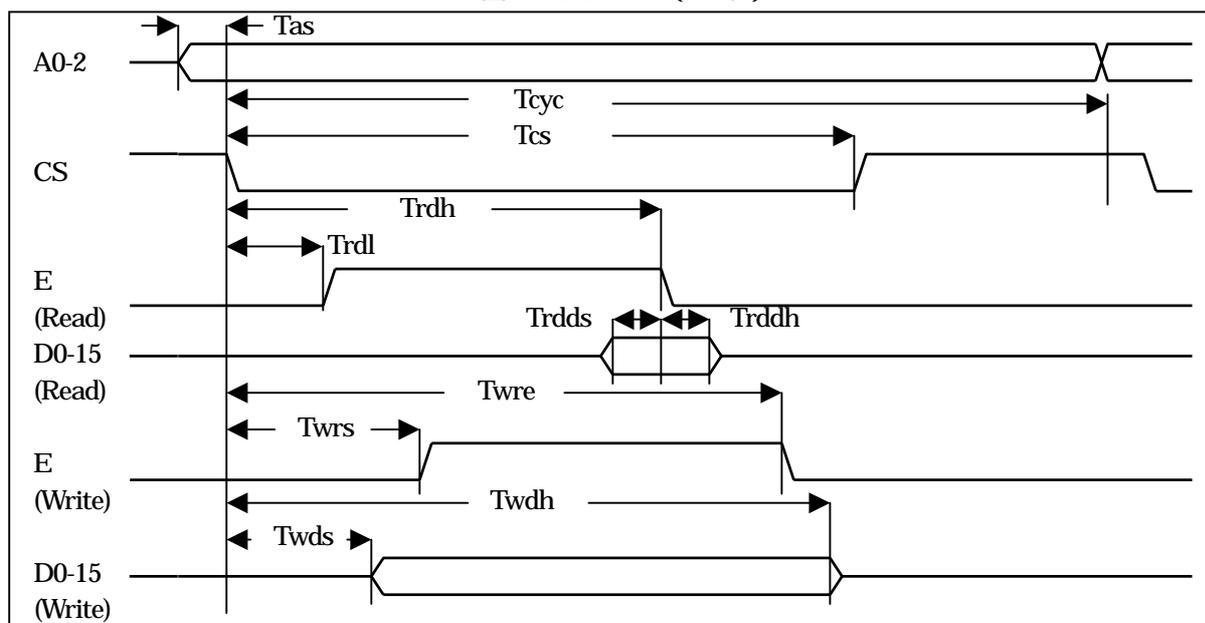


図 パラレル I/F (68 系)

パラレル I/F タイミング表

項目	記号	min	max	単位
Address セットアップ	$T_{as}$	60	80	nS
システム・サイクル	$T_{cyc}$	61	$20.3 \times (n1+3)$	nS
CS パルス幅	$T_{cs}$	20.3	$20.3 \times (n2+1)$	nS
RD セットアップ Low	$T_{rdl}$	20.3	$20.3 \times (n3+1)$	nS
RD セットアップ High	$T_{rdh}$	20.3	$20.3 \times (n4+1)$	nS
Read Data セットアップ時間	$T_{rdds}$	10	20	nS
Read Data ホールド時間	$T_{rddh}$	0	-	nS
WR セットアップ Low	$T_{wrs}$	20.3	$20.3 \times (n5+1)$	nS
WR セットアップ High	$T_{wre}$	20.3	$20.3 \times (n6+1)$	nS
Write Data セットアップ時間	$T_{wds}$	20.3	$20.3 \times (n7+1)$	nS
Write Data ホールド時間	$T_{wdh}$	20.3	$20.3 \times (n8+1)$	nS

注) n 1 ~ 8 = ソフトにて変更値(0 - 255まで設定可能)

### 5.2 シリアルI/F動作タイミング

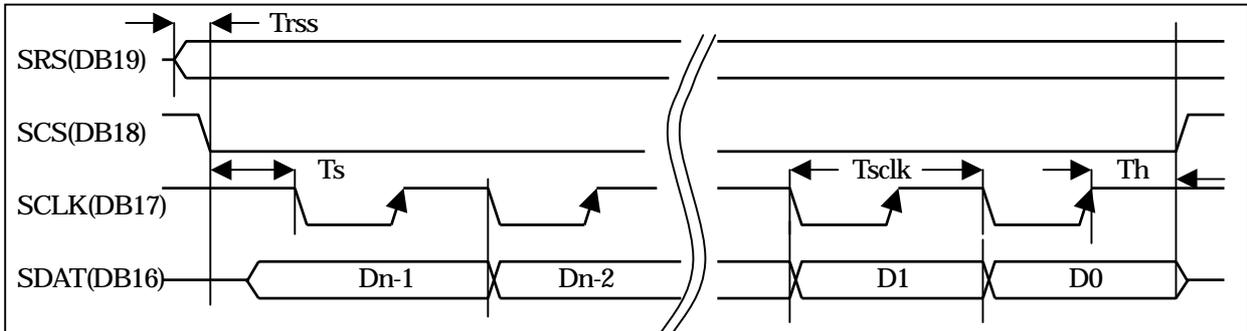


図 高速SPI(6 - 16 bit)

高速SPIタイミング表

項目	記号	min	max	単位
SCLK 周波数	Tsclk	a		MHz
SCLK ハイレベル時間	Tsclk_Lo	20	b / 2	nS
SCLK ロウレベル時間	Tsclk_Hi	20	b / 2	nS
SCLK セットアップ時間	Ts	20	b / 2	nS
SCLK ホールド時間	Th	20	b	nS
SRS セットアップ時間	Trss	80	160	nS

注) 周波数 256 段階 ( n = 0 - 255 まで 0=24MHz・…255=96kHz )

EXPMODE 命令により、SCLK 論理反転も可能

a = 24.576MHz / (n+1) ・ ・ ・ 24.576MHz、12.288MHz、…、96kHz

b = (n+1) / 24.576MHz ・ ・ ・ 40ns、81ns、…、10.4us

### 5.3 I2C動作タイミング

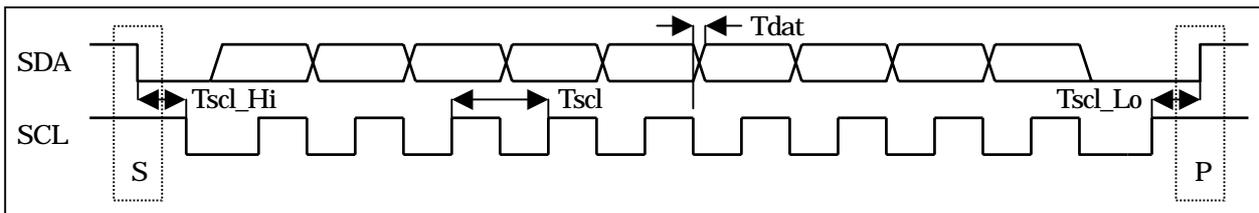


図 I2C

I2C タイミング表

項目	記号	min	max	単位
SCL 周波数	Tsl	C		MHz
SCL ハイレベル時間	Tsl_Lo	20	d / 2	nS
SCL ロウレベル時間	Tsl_Hi	20	d / 2	nS
Data セットアップ時間	Tdat	5	20	nS

注) 周波数 256 段階 ( n = 0 - 255 まで 0=24MHz・…255=96kHz )

c = 24.576MHz / (n+1) ・ ・ ・ 24.576MHz、12.288MHz、…、96kHz

d = (n+1) / 24.576MHz ・ ・ ・ 40ns、81ns、…、10.4us

## 5.4 DigitalRGB 動作タイミング

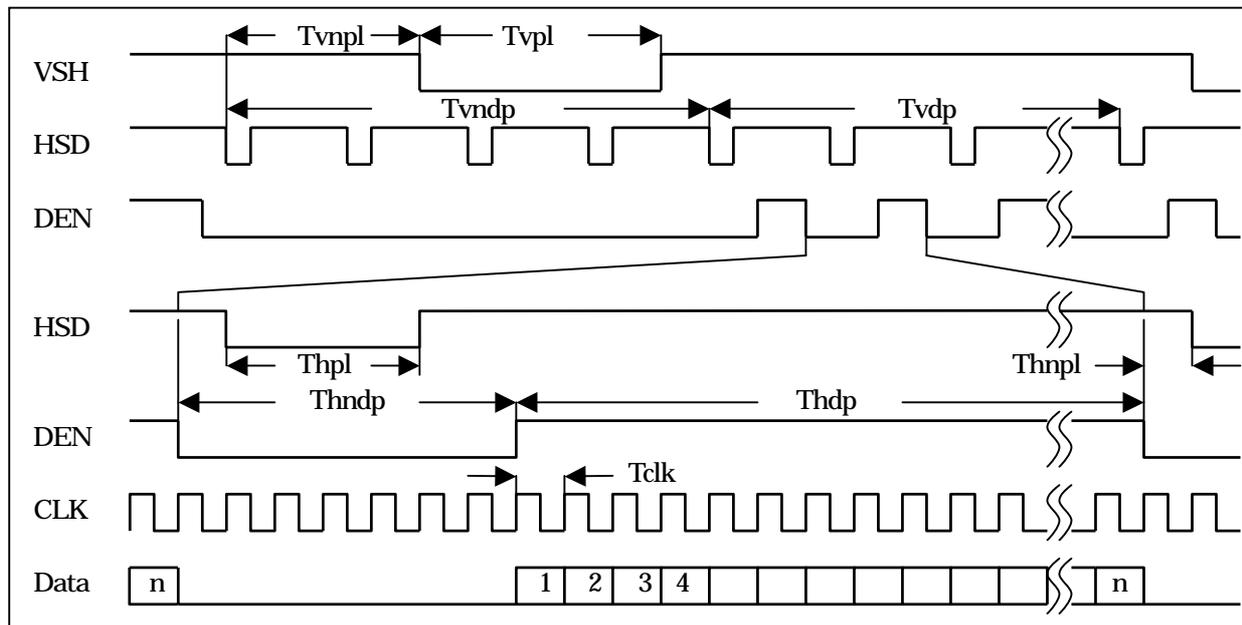


図 DigitalRGB

DigitalRGB タイミング表

項目	記号	min	max	単位
垂直表示期間	$T_{vdp}$	4	1024	Line
垂直非表示期間	$T_{vndp}$	1	256	Line
垂直開始位置	$T_{vnpl}$	1	256	Line
垂直パルス幅	$T_{vpl}$	1	256	Line
水平表示期間	$T_{hdp}$	16	2048	Clk
水平非表示期間	$T_{hndp}$	2	512	Clk
水平開始位置	$T_{hnpl}$	1	256	Clk
水平パルス幅	$T_{hpl}$	1	256	Clk
CLK周波数	$T_{clk}$	750	49.152M	Hz

注1) CLK周波数は「CLK\_OUT」又は「CLK2X\_OUT」命令で変更可能です。

CLK\_OUT 命令 : 周波数 =  $12.288\text{MHz} / (n+1)$  ...  $n = 0 \sim 16383$

CLK 2 X\_OUT 命令 : 周波数 =  $24.576\text{MHz} / (n+1)$  ...  $n = 0 \sim 32767$  (b15=0)

CLK 2 X\_OUT 命令 : 周波数 =  $49.152\text{MHz}$  ( $n=8000h$ )、 $16.384\text{MHz}$  ( $n=8001h$ )

注2) 水平又は垂直の各種設定は、

EXOUT[0,100000h] = hdp-1 ; 水平表示(0 - 2047)

EXOUT[0,100020h] = hndp-1 ; 水平非表示(1 - 511)

EXOUT[0,100040h] = (hnpl-1):(hpl-1) ; 水平開始位置(上位 0-255) / パルス幅(下位 0-255)

EXOUT[0,100060h] = vdp-1 ; 垂直表示(0 - 1024)

EXOUT[0,100080h] = vndp-1 ; 垂直非表示(1 - 255)

EXOUT[0,1000A0h] = (vnpl-1):(vpl-1) ; 垂直開始位置(上位 0-255) / パルス幅(下位 0-255)

注3) 各種波形の論理は正転・反転が可能です

### 5.5 FLM方式(STN)動作タイミング

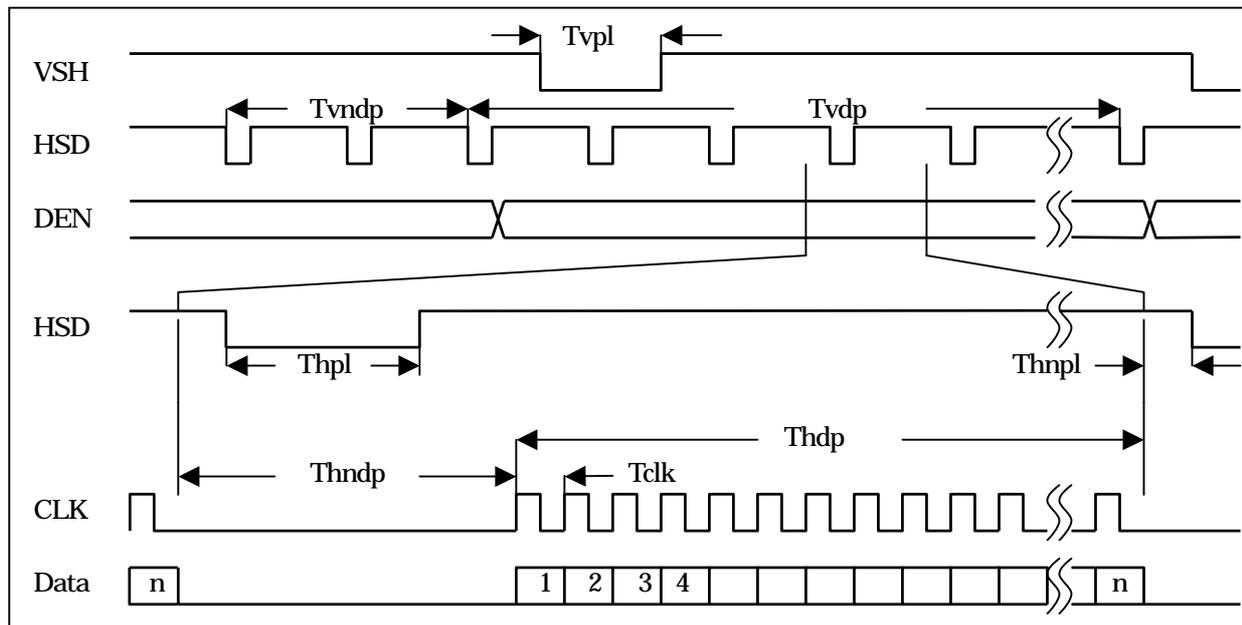


図 FLM方式(STN)波形出力

FLM方式 (STN 波形) 出力タイミング表

項目	記号	min	max	単位
垂直表示期間	Tvdp	4	1024	Line
垂直非表示期間	Tvndp	1	256	Line
垂直開始位置	Tvnpl	-	-	Line
垂直パルス幅	Tvpl	1	1	Line
水平表示期間	Thdp	16	2048	Clk
水平非表示期間	Thndp	2	512	Clk
水平開始位置	Thnpl	1	256	Clk
水平パルス幅	Thpl	1	256	Clk
CLK周波数	Tclk	750	49.152M	Hz

注1) CLK周波数は「CLK\_OUT」又は「CLK2X\_OUT」命令で変更可能です。

CLK\_OUT 命令 : 周波数 = 12.288MHz / (n+1) ... n = 0 ~ 16383

CLK 2 X\_OUT 命令 : 周波数 = 24.576MHz / (n+1) ... n = 0 ~ 32767

注2) 水平又は垂直の各種設定は、

EXOUT[0,100000h] = hdp-1 ;水平表示(0 - 2047)

EXOUT[0,100020h] = hndp-1 ;水平非表示(1 - 511)

EXOUT[0,100040h] = (hnpl-1):(hpl-1) ;水平開始位置(上位 0-255) / パルス幅(下位 0-255)

EXOUT[0,100060h] = vdp-1 ;垂直表示(0 - 1024)

EXOUT[0,100080h] = vndp-1 ;垂直非表示(1 - 255)

EXOUT[0,1000A0h] = (vnpl-1):(vpl-1) ;STN 出力の場合、無効

EXOUT[0,1000E0h] = (wave)(nLine) ;STN 出力の場合のみ、交流化信号

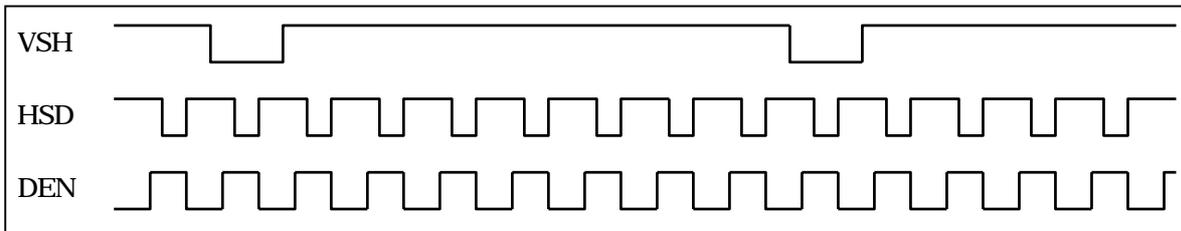
注3) 各種波形の論理は正転・反転が可能です

FLM方式 (STN) の場合 DEN 端子は交流化信号として出力します。

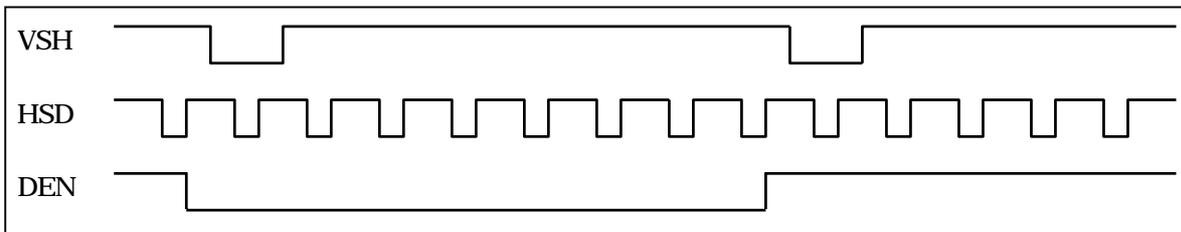
EXOUT[0,1000E0h] = (wave)(nLine-1) ;8bit データ : 交流化信号  
;wave =00:A 波形 / 01:B 波形 / 10:C 波形 / 11:C+XOR 波形  
;nLine= 1 - 6 4

各 wave 設定した時の DEN 端子の波形出力は、以下の通り

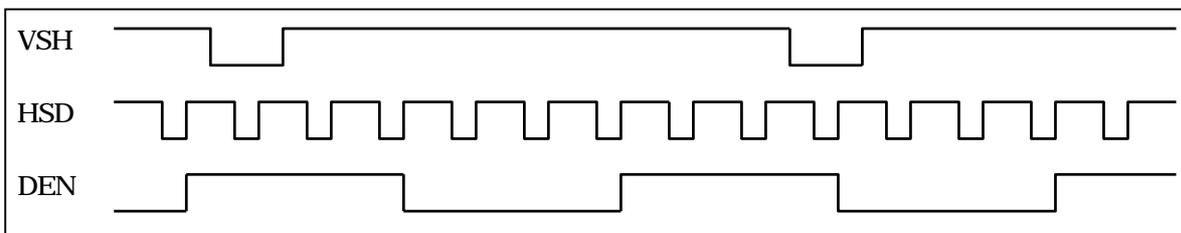
#### A 波形タイプ



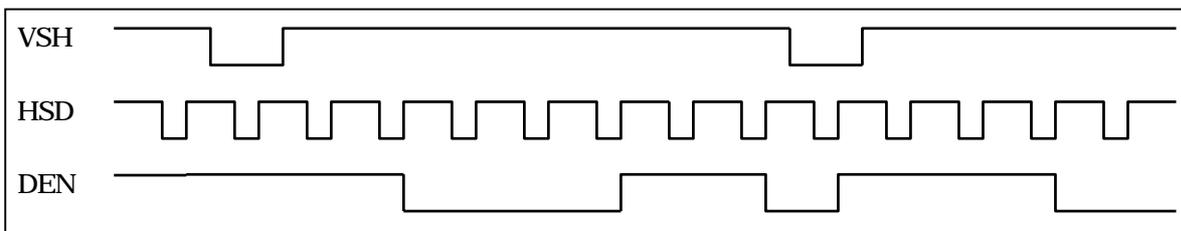
#### B 波形タイプ



#### C 波形タイプ ( nLine = 3 )



#### C+XOR 波形タイプ ( nLine = 3 )



この波形は、( B波形 ) x o r ( C 波形 ) の演算になります

## 5.6 Driver方式動作タイミング

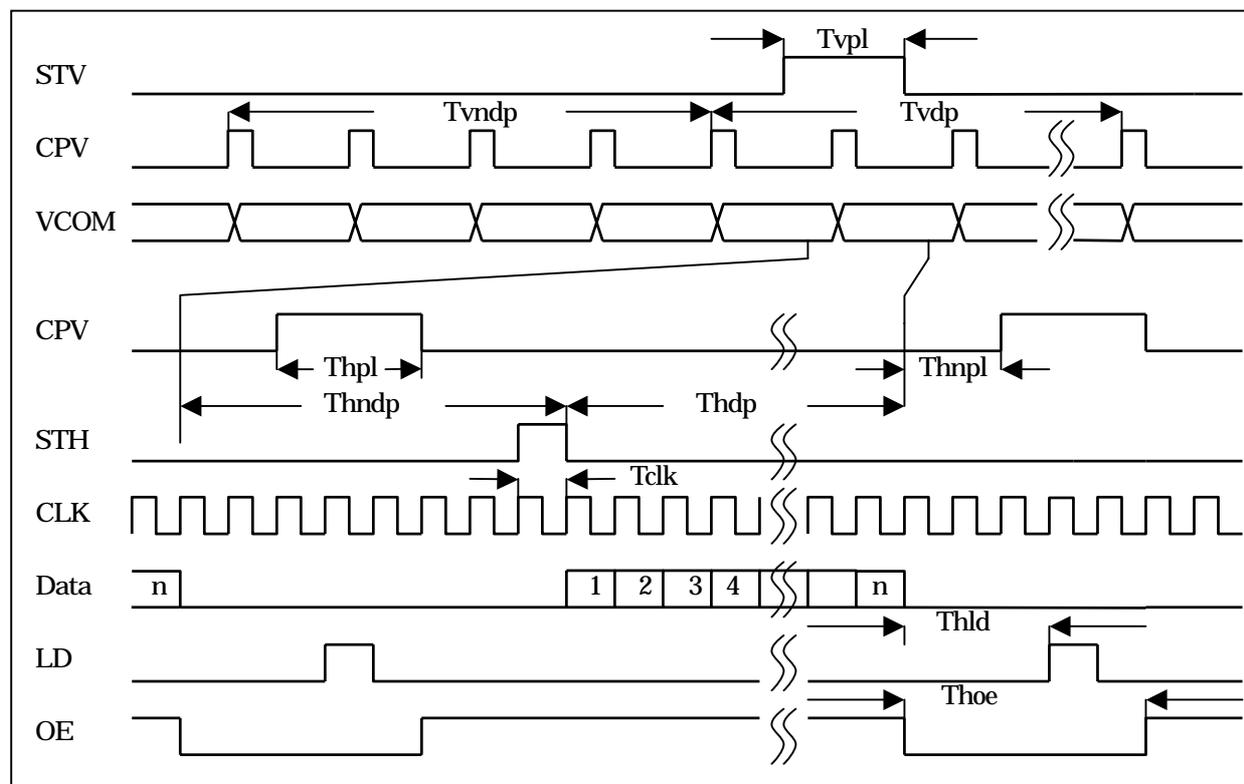


図 Driver方式

Driver方式タイミング表

項目	記号	min	max	単位
垂直表示期間	$T_{vdp}$	4	1024	Line
垂直非表示期間	$T_{vndp}$	1	256	Line
垂直開始位置	$T_{vnpl}$	-		Line
垂直パルス幅	$T_{vpl}$	1	1	Line
水平表示期間	$T_{hdp}$	16	2048	Clk
水平非表示期間	$T_{hndp}$	2	512	Clk
水平開始位置	$T_{hnpl}$	1	256	Clk
水平パルス幅	$T_{hpl}$	1	256	Clk
CLK周波数	$T_{clk}$	750	49.152M	Hz
LD開始位置	$T_{hld}$	$(T_{hnpl} + T_{hpl} / 2)$		Clk
OEパルス幅	$T_{hoe}$	$(T_{hnpl} + T_{hpl})$		Clk

## 6. 汎用コントローラのシーケンス動作について

メモリカードには、以下の情報を登録することが可能です。

- LCD ファイル・・・初期設定動作の記述ファイル(Text ファイルなのでユーザで自由に変更可能)
- BMP ファイル・・・検査モジュールに表示させる画像ファイル(ユーザで自由に変更可能)
- MOT ファイル・・・画像出力・電子 VR・パワーOFF シーケンスのプログラムファイル(ユーザで変更不可)
- DAT ファイル・・・メモリには            のデータが入りますが、その全てを1つのファイルにまとめたファイル

### 6.1 メモリマップ

	16Mb メモリマップ	8Mb メモリマップ	2Mb メモリマップ
00,0000h	InitData(256byte)	InitData(256byte)	InitData(256byte)
00,0100h	Driver(MOT)	Driver(MOT)	Driver(MOT)
00,1000h			
00,2000h	初期 Comand	初期 Comand	初期 Comand
00,4000h	Page1(BMP)	Page1(BMP)	Page1(BMP)
00,8000h	Page2(BMP)	Page2(BMP)	Page2(BMP)
00,C000h	Page3(BMP)	Page3(BMP)	Page3(BMP)
03,C000h	Page15(BMP)	Page15(BMP)	Page15(BMP)
			max03,FFFFh
0E,C000h	Page59(BMP)	Page59(BMP)	
0F,0000h	Page60(BMP)	初期 TEXT(LCD)	
		max0F,FFFFh	
1E,C000h	Page122(BMP)		
1F,0000h	初期 TEXT(LCD)		
	max1F,FFFFh		

LCD ファイルを登録すると、InitData 領域、初期 Comand、初期 TEXT 領域が上書きで登録されます。

- ・ InitData 領域には、LCD ファイル(Text データ)を checker 動作に重要な情報だけ保管する領域 (各種電圧・電流、I/F 動作、タイミング等)
- ・ 初期 Comand 領域は、LCD ファイル(Text データ)を、チェッカーが読める中間フォーマット変換の領域 (DOUT、TM\_WAIT 命令等、BASIC のインタプリタ方式に近い動作をしています)
- ・ 初期 TEXT 領域は、LCD ファイル内容を TEXT 保管するだけの領域

MOT ファイルを登録すると、Driver 領域に上書きで登録されます。

このファイルはプログラムそのものなので、ユーザで自由に変更できません。

画像出力シーケンス、電子 VR 出力シーケンス、パワーOFF シーケンスのプログラム情報が入っています。

(検査モジュールに使用されている LSI によって各種 MOT ファイルを用意しています)

BMP ファイルを登録すると、Page1 ~ n まで位置選択により上書き登録されます。

通常最大 1 2 2 画面登録(16Mb)ですが、Dot 数や色数が大きくなると 1page で 1 画面は入らなくなります。

その場合 2 page で 1 画面、Xpage で 1 画面等に登録が可能になりますが最大登録画面数が少なくなります。

## 6.2 動作シーケンス

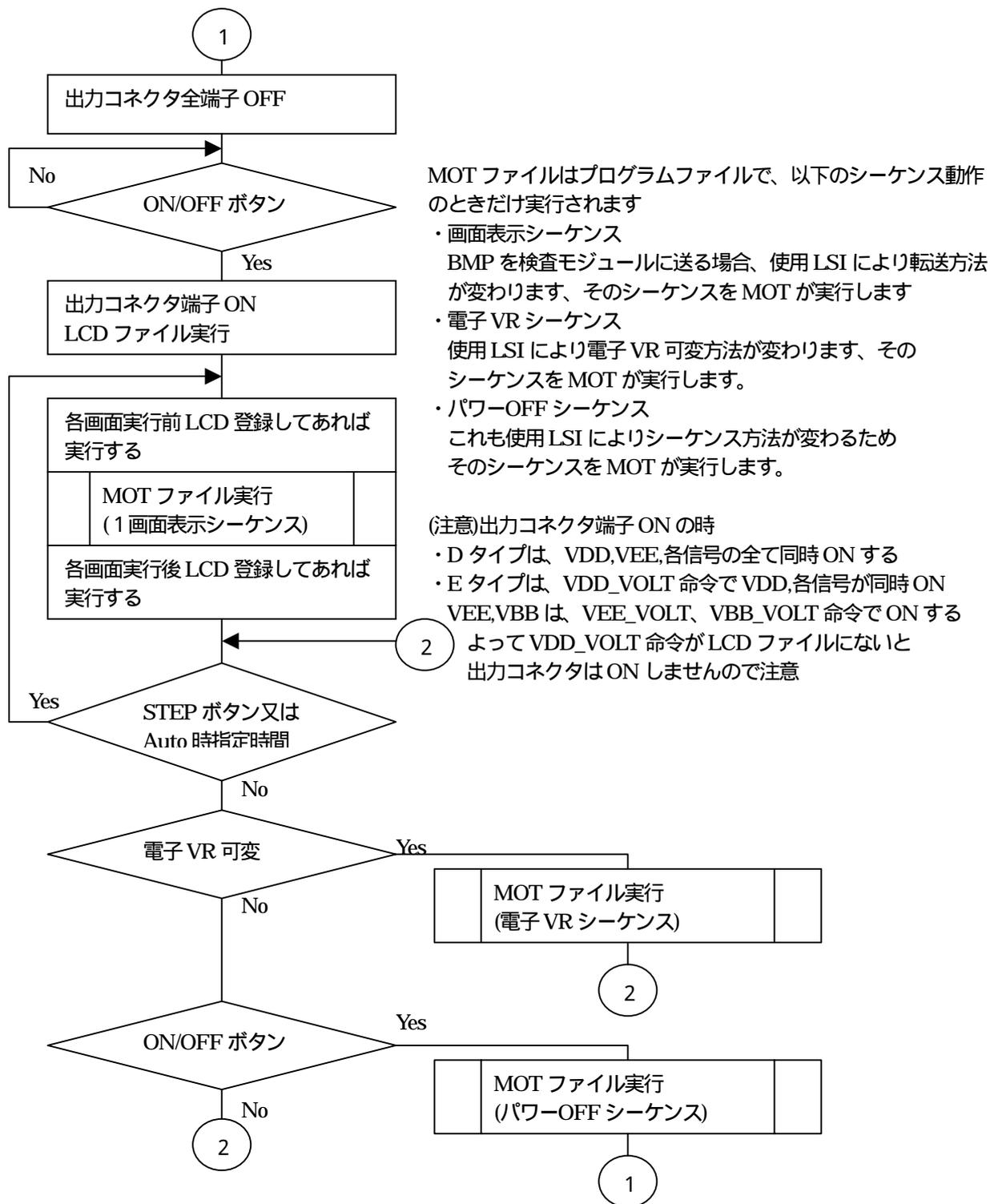
各種 SW の状態により、汎用コントローラは動作します。

ON/OFF ボタン(赤ボタン)・・・ 出力コネクタ(34,40 ピン)の ON/OFF 制御用ボタン

STEP ボタン ・・・ 表示画面(BMP)を切り替えるボタン

Auto/Manual SW ・・・ Manual 時は Step ボタンにより画面切替、Auto 時は指定時間で画面切替

ロータリエンコーダ ・・・ 電子 VR 機能があるモジュールなら、回す事によりコントラスト可変する



## 7. 初期設定(LCD ファイル)で使用できるコマンド

### 7.1 コメント

#### コメント表示(セミコロン=;) )

セミコロンの後の1行は、コメント文になります

#### PG\_NAME 命令(全タイプで使用可能)

設定プログラムの名前で、半角で48文字まで入力可能

基本的にモジュール名や Version 名を入力するために用意しました(見出し名前と考えて下さい)

D,E チェッカーでは、AC 電源投入時に表示器に48文字中20文字だけ2秒間点灯するので確認用にもなる

例) PG\_NAME = "F51002\_Version2.00"

### 7.2 MOT ファイルの設定用コマンド

#### PDRIVER 命令(全タイプで使用可能)

使用するドライバーのファイル名(MOT 名)を設定します。

使用 LSI により画像出力・電子 VR・パワー-OFF の各シーケンス方法が違うためプログラムとして用意

例) PDRIVER = "S6B0021\_V132.MOT"

#### DISTYPE 命令(全タイプで使用可能)

ディスプレイTYPEの設定(キャラクタータイプ又はグラフィックタイプの判定用)

例) DISTYPE = GRAPHIC ;グラフィックTYPE

DISTYPE = CHARACTER ;キャラクターTYPE

#### EV\_INIT 命令(全タイプで使用可能)

MOT ファイルの電子 VR 機能の初期値を設定する命令で16ビットデータ(0-65535)が入力可能です

チェッカーには2つのロータリエンコーダがありますので、メイン用=EV\_INIT 命令

サブ用=EV\_INIT2 命令と2つ用意しています

注)この値は電子 VR の初期値用データで、検査モジュールに直接送られる値ではありません

例) EV\_INIT = 127 ;Main コントラスト VR 初期値 127

EV\_INIT2 = 63 ;Sub コントラスト VR 初期値 63

#### EXPDATA 命令(全タイプで使用可能)

ドライバー(MOT ファイル)に拡張命令として16ビットデータ(0-65535)が入力可能です

EXPDATA1 ~ 8 まで8個命令を用意しています

各設定データはドライバーファイルによって値の意味が変わります。

例) EXPDATA1 = 1234H ;設定データ 1

EXPDATA2 = 128 ;設定データ 2

EXPDATA3 = 160 ;設定データ 3

EXPDATA4 = 65535 ;設定データ 4

### 7.3 チェッカーの設定用コマンド

#### I/O インターフェイス設定用の命令

汎用基板・D チェッカータイプでは、**IO\_MODE** 命令によってインターフェイスを設定します。  
パラレルの場合、I80 系・M68 系どちらかでデータビット数も4・8・16・20ビットの4種類  
シリアルの場合、SPI のみ出力可能です。

例) IO\_MODE = SERIAL ;シリアル(SPI)出力  
IO\_MODE = I80\_04 ;I80 系 4Bit 出力  
IO\_MODE = I80\_08 ;I80 系 8Bit 出力  
IO\_MODE = I80\_16 ;I80 系 16Bit 出力  
IO\_MODE = I80\_20 ;I80 系 20Bit 出力  
「I80」「M68」に変えれば M68 系出力になります

E チェッカーでは、**IO\_MODE** 命令、**EXPMODE** 命令によってインターフェイスを設定します。  
パラレルの場合、I80 系・M68 系どちらかでデータビット数も4・8・16・20ビットの4種類  
更にデータ出力 OFF 時の状態を High、Low、Hi-z の3種類設定が可能です (D チェッカーは Low のみ)

例) IO\_MODE = I80\_04 ;I80 系 4Bit 出力(データ OFF 時、Low 状態)  
IO\_MODE = I80\_08 ;I80 系 8Bit 出力(データ OFF 時、Low 状態)  
IO\_MODE = I80\_16 ;I80 系 16Bit 出力(データ OFF 時、Low 状態)  
IO\_MODE = I80\_20 ;I80 系 20Bit 出力(データ OFF 時、Low 状態)  
IO\_MODE = I80\_16\_HI ;I80 系 16Bit 出力(データ OFF 時、High 状態)  
IO\_MODE = I80\_16\_HIZ ;I80 系 16Bit 出力(データ OFF 時、Hi-z 状態)  
「I80」「M68」に変えれば M68 系出力になります

パラレル時 No.35 を Busy 入力にして外部からタイミングを遅らせることも可能です  
その場合 EXPMODE 命令で Busy 入力に設定します。

例) EXPMODE = BUSY ;No.35 を Busy 入力端子にします (Low で Wait アクティブ動作)

シリアルの場合、SPI 又は I2C 出力が可能です **IO\_MODE**、**EXPMODE** 命令で設定します

例) SPI 出力の場合  
IO\_MODE = SERIAL ;SPI 出力  
EXPMODE = SDIN ;No.35 を Sdin 入力端子にします (SPI は立ち上がりエッジ有効)  
(注意)SPI の立下りエッジ有効にすることも可能です  
EXPMODE = SDIN\_RISING ;No.35 を Sdin 入力端子(SPI は立下りエッジ有効)  
例) I2C 出力の場合  
IO\_MODE = SERIAL ;I2C 出力  
EXPMODE = IIC ;No.35 を SDA 入出力端子にします (SCL は No.32)

DigitalRGB 出力にすることも可能です

例) IO\_MODE = DIGITALRGB ;DigitalRGB 出力  
EXPMODE = POT ;No.35 を POT 入力端子にします

#### 電源設定用の命令

D チェッカーでは VDD・VEE の2電源、E チェッカーでは VDD・VEE・VBB の3電源あり電圧を設定します  
VDD 電圧は、1.50 ~ 5.00V まで 0.01V 単位で設定可能です

例) VDD\_VOLT = 3.00 ;V

VEE 電圧は、D チェッカーは 0.0 ~ 19.0V、E チェッカーは 1.5 ~ 29.0V まで設定可能です

例) VEE\_VOLT = 15.00 ;V

VBB 電圧は、E チェッカーで 1.5 ~ 29.0V まで設定可能です

例) VBB\_VOLT = 10.00 ;V

(注意) D チェッカーでは VBB\_VOLT 命令があると VEE 電圧がおかしくなりますので注意が必要

- (注意) E チェッカーでは 0 . 0 ~ 1 . 5 V までの設定では 0 V として出力動作します  
(注意) D チェッカーでは V<sub>xx</sub>\_VOLT 命令を書き忘れると、VDD=3.0V、VEE=0.0V 初期値で動作します。  
E チェッカーでは V<sub>xx</sub>\_VOLT 命令を書き忘れると、VDD=VEE=VBB=0.0V で動作します。

### 電流 max リミッタ設定の命令

各電源の max リミット値を設定できます、設定値を超えるとブザーが鳴り電源 OFF 動作します  
D チェッカーでは、0 < VDD < 50mA、0 < VEE < 200mA の範囲で設定可能です(0.05mA 単位)  
E チェッカーでは、VDD・VEE・VBB とともに 0 ~ 800mA の範囲で設定可能です(0.0125mA 単位)

例) VDD\_LIMIT = 10.00 ;mA  
VEE\_LIMIT = 100.00 ;mA  
VBB\_LIMIT = 200.00 ;mA

更に E チェッカーでは I/O 端子のショート検査等に VDD\_IFLIMIT 命令も設定可能です  
未使用時「VDD\_IFLIMIT = 30 ;mA」がデフォルト値で 0 ~ 800mA まで設定可能です(0.0125mA 単位)

例) VDD\_IFLIMIT = 125.00 ;mA

### 電流 min リミッタ設定の命令

各電源の min リミット値を設定できます、設定値以下でブザーが鳴りますが電源 OFF はしません  
D チェッカーでは、0 < VDD < 50mA、0 < VEE < 200mA の範囲で設定可能です(0.05mA 単位)  
E チェッカーでは、VDD・VEE・VBB とともに 0 ~ 800mA の範囲で設定可能です(0.0125mA 単位)

例) VDD\_LoLIMIT = 1.05 ;mA  
VEE\_LoLIMIT = 5.12 ;mA  
VBB\_LoLIMIT = 7.50 ;mA

### Power ON/OFF 命令

チェッカーの出力 3 4 ピンの電源 ON/OFF の設定を可能にした  
(通常は ON/OFF SW があるので必要は無いが、あくまでデバッグ用)

例) POWER = OFF ;電源を強制的に OFF する  
例) POWER = ON ;電源を強制的に ON する

D チェッカーの場合、VDD・VEE は同時 ON/OFF 動作しかしませんが、  
E チェッカーでは VDD・VEE・VBB の ON タイミングがずれて動作します

例) VDD\_VOLT = 3.00 ;V  
VBB\_VOLT = 5.00 ;V  
VEE\_VOLT = 7.00 ;V

この例では、VDD、VBB、VEE の順で電圧 ON します

そこで以下のようにすると VDD・VEE・VBB が同時 ON します (最初に Power 命令入れる)

例) POWER = ON  
VDD\_VOLT = 3.00 ;V  
VBB\_VOLT = 5.00 ;V  
VEE\_VOLT = 7.00 ;V

## 7.4 各入出力の設定用コマンド

### 時間待ちの設定命令

汎用基板・D チェッカータイプでは、**TM\_WAIT** 命令によって 1 ~ 32768ms まで時間待ちの設定が可能  
例) **TM\_WAIT = 1000 ;1 秒間時間待ち**

E チェッカータイプでは、**TM\_WAIT** 命令によって 1 ~ 32768ms まで時間待ちの設定が可能

例) **TM\_WAIT = 1000 ;1 秒間時間待ち**

更に **TU\_WAIT** 命令が使用でき、10us 単位で時間待ちの設定が可能

例) **TU\_WAIT = 5 ;50uS 時間待ち ( 10us × 5 )**

### 出力ポートのビット ON/OFF

汎用基板・D チェッカータイプでは、A0・A1・A2・RES・SCS 端子の 5 本のみ High/Low 設定が可能  
High にする場合は **BIT\_SET** 命令、Low にする場合は **BITRSET** 命令で設定します

例) **BIT\_SET = RES ;RES 端子 High**  
**BITRSET = RES ;RES 端子 Low**  
**BIT\_SET = A1 ;A 1 端子 High**  
**BIT\_SET = SRCS ;SCS 端子 High(BIOS Version4.00 以上)**  
**BITRSET = SRCS ;SCS 端子 Low(BIOS Version4.00 以上)**

E チェッカーでは、全端子の High/Low 設定が可能

High にする場合は **BIT\_SET** 命令、Low にする場合は **BITRSET** 命令で設定します

端子名は A0・A1・A2・RES・RDE・WRRD・ECLK・CS・DB01...DB15・SDAT・SCLK・SCS・SRS

例) **BIT\_SET = RES ;RES 端子 High**  
**BITRSET = WRRD ;WRRD 端子 Low**  
**BIT\_SET = DB02 ;DB2 端子 High**  
**BIT\_SET = SDAT ;SDAT 端子 High**  
**BIT\_SET = SRS ;SRS 端子 High**  
**BITRSET = SCS ;SCS 端子 Low ( D チェッカーと同じ SRCS でも可 )**

### DOUT 命令、EXOUT 命令

DOUT 命令は、IO\_MODE 命令で設定したインターフェイスでデータを出力する命令です

汎用基板・D チェッカータイプでは、低速 SPI、I80 系、M68 系のとき使用できます

同時に A0,A1,A2 端子も変化させることができます

例) **DOUT[0] = 128 ;A0=0,A1=0,A2=0:DATA10 進値 128 OUTPUT**  
**DOUT[3] = 1234H ;A0=1,A1=1,A2=0:DATA16 進値 1234H OUTPUT**  
**DOUT[7] = 11110000B ;A0=1,A1=1,A2=1:DATA 2 進値 F0H OUTPUT**

E チェッカーでは、I80 系、M68 系、高速 SPI、I2C のとき使用できます

・パラレル(I80,M68)時、A0,A1,A2 端子を変化させることができます

例) **DOUT[0] = 128 ;A0=0,A1=0,A2=0:DATA10 進値 128 OUTPUT**  
**DOUT[3] = 56789H ;A0=1,A1=1,A2=0:DATA16 進値 56789H OUTPUT**

・高速 SPI 時、SRS 端子を変化させることができます

例) **DOUT[0] = 128 ;SRS=0:DATA10 進値 128 OUTPUT**  
**DOUT[1] = 15H ;SRS=1:DATA16 進値 15H OUTPUT**

・I2C 時、Start 波形、Data 出力波形、Stop 波形の 3 種類の機能で以下のようになります

例) **DOUT[0] = I2C\_START ;I2C Start 波形**  
**DOUT[0] = 23h ;I2C Data 出力波形(16 進 2 3 h 出力)**  
**DOUT[0] = I2C\_STOP ;I2C Stop 波形**

EXOUT 命令は、拡張用の機能を使う場合に必要な命令で D チェッカーでは無効です  
汎用基板では拡張基板にデータを送る時に必要な命令です

例) EXOUT[0,1234h] = 5678h ;A0=0,Adres=1234h:DATA16 進値 5678h OUTPUT  
EXOUT[1,0020h] = 63 ;A0=1,Adres=0020h:DATA10 進値 63 OUTPUT

E チェッカーでは拡張機能を使うときに必要な命令です (特に DigitalRGB 出力)

例) EXOUT[0,100000h]= 319 ;(RegHDP )b11-0=>水平表示(0-2047)xHCK  
EXOUT[0,100020H]= 15 ;(RegHNDP)b 8-0=>水平非表示(2-511)xHCK  
EXOUT[0,100040H]= 0303H ;(RegHPL )b15-8=>開始位置(0-255) /b7-0=>パルス幅(0-255)  
EXOUT[0,100060H]= 239 ;(RegVDP )b 9-0=>垂直表示(0-1023)  
EXOUT[0,100080H]= 15 ;(RegVNDP)b 7-0=>垂直非表示(2-255)  
EXOUT[0,1000A0H]= 0303h ;(RegVPL )b15-8=>開始位置(0-255) /b7-0=>パルス幅(0-255)  
EXOUT[0,1000C0H]= 0080H ;(RegBIT )b10-0=> MODE[3]  
; DISP 18b STN POL VSD HSD DEN HCK

### DTIN 命令、EXPIN 命令

DTIN 命令は、IO\_MODE 命令で設定したインターフェイスでデータを入力する命令です  
汎用基板・D チェッカータイプでは、I80 系、M68 系のとき使用できます  
同時に A0,A1,A2 端子も変化させることができます

例) DTIN[0] = ;A0=0,A1=0,A2=0:Data Input  
DTIN[3] = ;A0=1,A1=1,A2=0: Data Input

E チェッカーでは、I80 系、M68 系、高速 SPI、I2C のとき使用できます

・パラレル(I80,M68)時、A0,A1,A2 端子を変化させることができます

例) DTIN[0] = ;A0=0,A1=0,A2=0: Data Input  
DTIN[3] = ;A0=1,A1=1,A2=0: Data Input

・高速 SPI 時、SRS 端子を変化させることができます

例) DTIN[0] = ;SRS=0: Data Input  
DTIN[1] = ;SRS=1: Data Input  
DOUT[0] =1234h ;SRS=0: Data Output  
DTIN[2] = ;前回の値 ( SPI 波形出力しないで前行の DOUT で Read した値を読む)

・I2C 時、Data 入力波形で使用でき 9bit 目の ACK を High/Lo することが可能です

例) DOUT[0] = I2C\_START ;I2C Start 波形  
DTIN[0] = ;I2C Data InputData ( ACK=Lo で動作する )  
DTIN[1] = ;I2C Data InputData ( ACK=High で動作する )  
DOUT[0] = I2C\_STOP ;I2C Stop 波形

EXPIN 命令は、拡張用の機能を使う場合に必要な命令で D チェッカーでは無効です  
汎用基板では拡張基板にデータ Read するのに必要な命令です

例) EXPIN[0,1234h] = ;A0=0,Adres=1234h:Data Input  
EXPIN[1,0020h] = ;A0=1,Adres=0020h:Data Input

E チェッカーでは拡張機能を使うときにデータ Read で必要な命令です

例) EXPIN[0,100000h]= ;(RegHDP )b11-0=>水平表示(0-2047)Input

### B\_IN 命令

B\_IN 命令は、IO\_MODE 命令で設定したインターフェイスでデータを入力し指定 bit が Lo/High になるまで待つ機能です (指定ビットが変化無しでも約 500ms 以上なら終了します)

(bit7 = Lo/High になるまで待つ、bit4-0 = 指定ビット 1 9 ~ 0 )

例) B\_IN[0] = 04h ;A0=1,A1=0,A2=0:DATA INPUT bit4 が 0 になるまで待つ  
B\_IN[1] = 8ch ;A0=1,A1=0,A2=0:DATA INPUT bit12 が 1 になるまで待つ

## LOOP\_BK 命令

前回のコマンドを繰り返すときに使用する命令です

例) DOUT[1] = FFH ;00FFH OUTPUT  
LOOP\_BK = 9 ;前の DOUT[1] = FFH を 10 回繰り返す

更に前 2 行を繰り返す LOOP\_BK2 命令から前 15 行を繰り返す LOOP\_BK15 命令まで用意しています

例) DOUT[1] = FF00h  
DOUT[1] = 00FFh  
LOOP\_BK2 = 15 ;DOUT 命令 2 行を 16 回繰り返し (合計 DOUT 命令 32 回)  
LOOP\_BK3 = 9 ;DOUT 命令 2 行を 16 × 10 = 160 回繰り返し (合計 DOUT 命令 320 回)

## CLK\_OUT 命令 (CLK2X\_OUT)

ECLK 端子に CLK 波形を出力する場合使用します

汎用基板・D チェッカーの場合、0 ~ 8191 まで設定できますので 12.288MHz ~ 15kHz まで変更可能

例) CLK\_OUT = 3 ;CLK = 12.288MHz/(data+1) = 3.072MHz  
CLK\_OUT = 1227 ;CLK = 12.288MHz/(data+1) = 10.00kHz

E チェッカーの場合、0 ~ 16383 まで設定できますので 12.288MHz ~ 750Hz まで変更可能

例) CLK\_OUT = 3 ;CLK = 12.288MHz/(data+1) = 3.072MHz  
CLK\_OUT = 1227 ;CLK = 12.288MHz/(data+1) = 10.00kHz

E チェッカーは CLK2X\_OUT 命令も使用が可能で 49.152MHz ~ 1.5kHz まで変更可能です

b15 ビットが 0 又は 1 かで基準周波数が変わります。

例) CLK2X\_OUT = 0 ;(b15=0)CLK = 24.576MHz/(data+1) = 24.576MHz  
CLK2X\_OUT = 490 ;(b15=0)CLK = 24.576MHz/(data+1) = 50.05kHz  
CLK2X\_OUT = 8002h ;(b15=1)CLK = 49.152MHz/(data+1) = 16.384MHz

設定値を FFFFh にすると、CLK 出力を OFF にできます

例) CLK\_OUT = FFFFh ;CLK OFF  
CLK2X\_OUT = FFFFh ;CLK OFF

DigitalRGB 出力の場合、CLK 出力端子として動作します

## SOUT 命令

汎用基板・D チェッカーの場合、高速 SPI でデータ出力する場合使用する命令です

シリアル転送用、CLK は 4 段階、転送ビット数も 4 段階ある

CLK=12MHz : HS、6MHz:FS、3MHz:MS、1.5MHz:LS

BIT = 08, 09, 10, 11

例) SOUT[0,HS08B]=7FFH ;RS=0,CLK=12MHz,8bit(データは上位 3bit が無効で 7FFH を出力)  
SOUT[1,FS09B]=7FFh ;RS=1,CLK=6MHz,9bit(データは上位 2bit が無効で 1FFH を出力)  
SOUT[0,MS10B]=7FFh ;RS=0,CLK=3MHz,10bit(データは上位 1bit が無効で 3FFH を出力)  
SOUT[1,LS11B]=7FFh ;RS=0,CLK=1.5MHz,11bit(データは 7FFH を出力)

E チェッカーでも、高速 SPI でデータ出力します (D チェッカーと互換性を保つため)

## 7.5 LCDファイル作成例

```

;-----
; Echecker DigitalRGB 設定例
; TFT320x240dots 65kColor(16bit)
;          2008.04.22 by imq
;-----
PG_NAME = "QVGA_SAMPLE_V1.00"
;***** MOT 設定 *****
PDRIVER = "DigitalCont_V500.MOT"
DISTYPE = GRAPHIC
EXPDATA1= 00h    ;02h:縦2倍転送 /20h:横2倍 /22h:縦横2倍
EXPDATA2= 0      ;0:65kColor / 1:mono 変換
EV_INIT = 0      ;未使用
;***** E Checker 設定 *****
IO_MODE = DigitalRGB
EXPMODE = POT          ;No.35 POT input
VDD_VOLT = 3.30       ;V
VDD_LIMIT= 100.00     ;mA
VEE_VOLT = 24.00      ;V
VEE_LIMIT= 50.00      ;mA
TM_WAIT = 200
BITRSET = RES         ;Reset Low
TM_WAIT = 1
BIT_SET = RES         ;Reset High
TM_WAIT = 10
;***** DigitalRGB 設定 *****
;CLK = 24.576MHz
;FIm = 24.576MHz/4/(320+80)/(240+16) = 60.00Hz
;----- DCLK
;CLK_OUT = 0          ;12.288MHz/(data+1)=12.288MHz
CLK2X_OUT= 3          ;24.576MHz/(data+1)= 6.144MHz
;----- 水平表示設定
EXOUT[0,100000h]= 319 ;(RegHDP )水平表示 (0-2047)+1
EXOUT[0,100020h]= 79  ;(RegHNDP)水平非表示(2- 511)+1
EXOUT[0,100040h]= 1F0FH ;(RegHNPL)b15-8=>開始位置(0-255)+1
; (RegHPL ) b7-0=>パルス幅(0-255)+1
;----- 垂直表示設定
EXOUT[0,100060h]= 239 ;(RegVDP )垂直表示 (0-1023)+1
EXOUT[0,100080h]= 15  ;(RegVNDP)垂直非表示(2- 255)+1
EXOUT[0,1000A0h]= 0404h ;(RegVNPL)b15-8=>開始位置(0-255)+1
; (RegVPL ) b7-0=>パルス幅(0-255)+1
EXOUT[0,1000E0h]= 7Fh ;(RegnLine)b7-6=>wave(00:A,01:B,10:C,11:C+Xor)
; b5-0=>nLine(0-63)+1
;----- 各bit 設定(MODE=000,DISP=1)
TM_WAIT = 10
EXOUT[0,1000C0h]= 083H ;(RegBIT )b10-8=> DRV OE LD SYNC x MODE[3]
; b7-0=> DISP 18b STN POL VSD HSD DEN HCK

```

## 8.ソフトウェアのインストールについて

### 8.1 ソフトウェアに必要なシステム

Windows Me/2000/Xp で正しく動作させるためには以下の条件が必要です。

Pentium 1GHz 以上のCPUを使用するPCを推奨

メモリ 256Mbyte 以上を推奨

画面表示 800×600 以上、256色以上表示可能なもの

1つ以上のシリアルポート(RS232C)又はUSB

シリアルポートで通信するために、ストレートケーブルを使用して下さい \ (DSUB9pin メス)

USB で通信するためには、USB ケーブルを用意してください \ (Btype コネクタ)

### 8.2 ソフトウェアのインストールについて

CD-ROM ドライブの SETUP.EXE を実行させるとセットアップウィザードが起動しますので、画面の指示に従って簡単にセットアップできます。



「NEXT」ボタンを押していくだけで、基本的に自動でインストールされます

インストールフォルダのみユーザで変更できる画面が表示されます

初期フォルダは「C:\Program Files\ALL\_LCD」ですが、問題あるなら変更してもかまいません。

### 8.3 USB ドライバーのインストールについて

USB でチェッカーと通信したい場合、USB ドライバーをインストール必要があります  
(RS232C のみで通信する場合は USB ドライバーは無くても動作します)  
また Version3.26 以前を使って USB 通信していた PC には旧 USB ドライバーが入っていますが  
Version4.01 以上から新 USB ドライバーを使用してもらいたい。

#### 初めて汎用ソフトをインストールした PC に USB ドライバーをインストールする場合

汎用基板またはチェッカーとは PC からはずしてください。  
(RS232C または USB との接続をしていない状態にして下さい)  
「USB Driver」のインストール  
・ CD-ROM ドライブの newUSBdriver フォルダを開く  
・ 「USBinstall.bat」ファイルをダブルクリックして USBdriver のインストール  
(DOS 窓が表示されて自動的にインストールされますので、DOS 窓が閉まるまで待ってください)  
汎用基板またはチェッカーの電源を入れ PC と USB で接続する  
「新しいハードウェアウィザード」が表示されますので  
「ソフトウェアを自動的にインストールする(推奨)」を選択して実行してください。  
自動で MultiLCDcontrol という名前でインストールされます。

通常は自動でインストールされますが、MultiLCDcontrol をもし見つけられなかった  
場合はファイルの場所を聞いてきますので、その場合は手動で CDROM の newUSBdriver フォルダ  
の「MultiLCDcontrol.inf」指定してインストールしてください。

#### Version3.26 以前の汎用コントローラが入っている PC に USB ドライバーをインストールする場合

汎用基板またはチェッカーとは PC からはずしてください。  
(RS232C または USB との接続をしていない状態にして下さい)  
「USB Driver」のインストール  
・ CD-ROM ドライブの newUSBdriver フォルダを開く  
・ 「USBinstall.bat」ファイルをダブルクリックして USBdriver のインストール  
(DOS 窓が表示されて自動的にインストールされますので、DOS 窓が閉まるまで待ってください)  
汎用基板またはチェッカーの電源を入れ PC と USB で接続する  
旧 USBdriver 「Netchip 'Transfer device' (VID\_0525&PID\_1000) using NcBulk.SYS」が  
自動選択されてしまいますので、新 USBdriver に更新する必要があります。

- ・ コントロールパネルのシステムを開く
- ・ ハードウェアタブを選んでデバイスマネージャを開く
- ・ 「Netchip 'Transfer device' (VID\_0525&PID\_1000) using NcBulk.SYS」を右クリックして  
ドライバーの更新をするとハードウェア更新ウィザードが開きます。
- ・ 「ソフトウェアを自動的にインストールする(推奨)」を選択して実行してください。  
自動で MultiLCDcontrol という名前でインストールされます。

通常は自動でインストールされますが、MultiLCDcontrol をもし見つけられなかった  
場合はファイルの場所を聞いてきますので、その場合は手動で CDROM の newUSBdriver フォルダ  
の「MultiLCDcontrol.inf」指定してインストールしてください。

正常に USB ドライバーがインストールされると、Windows デバイスマネージャには以下のように「MultiLCDcontrol」という USB ドライバーが表示されます  
(チェッカーと USB で接続された状態のとき)



#### 8.4 ソフトウェアのアンインストールについて

本ソフトが不要になった場合、削除することができます。

- . 「スタート」ボタンから「設定」を選択し「コントロールパネル」をクリック。
- . 「アプリケーションの追加と削除」アイコンをダブルクリックし「セットアップと削除」タブが前面に表示されていることを確認します。
- . 一覧の中から「汎用コントローラ」を選択し「追加と削除」ボタンをクリックします。
- . 削除を確認するメッセージが表示されたら「OK」をクリックします。  
プログラムが削除されるまで待ちます。

## 9.ソフトウェアの使用方法

### 9.1 メイン画面

本ソフトを起動すると Main 画面が表示されます。



#### 「終了」ボタン

本ソフトを終了させるとき押してください。  
このボタン以外では本ソフトを終了させることはできません。

#### 「日本語・英語」ボタン

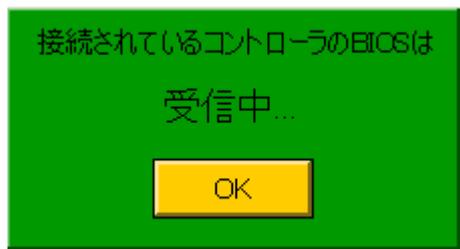
本ソフトの表示を日本語か英語に切り替えができます。

#### 「SerialPort」ボタン

汎用コントローラと通信するために接続したポート番号に合わせてください。  
・RS232C、USB 接続 : COM1 ~ COM7 又は USB で選択してください  
・RAM 接続 : RAM\_2Mb、RAM\_8Mb、RAM\_16MB に選択してください  
接続したポート番号と正しく合わせないと通信できません。  
RAM 接続の場合、PC メモリーを仮想 RAM としますのでチェッカー無しでデータ作成可能(DAT 作成用)

#### 「BIOS Version」ボタン

現在接続中の汎用コントローラの BIOS プログラムバージョンを確認できます。



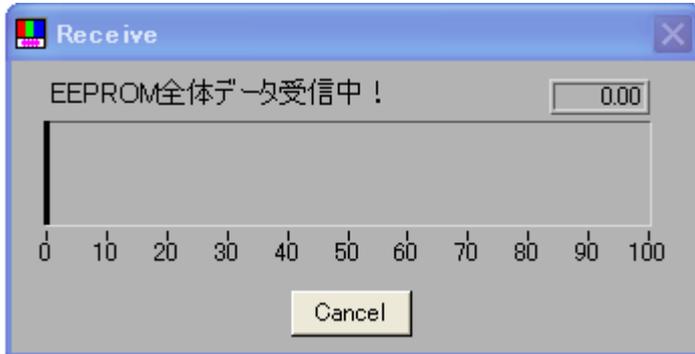
正常に通信できると「受信中」の所に接続されたコントローラの Version が表示されます。

### 「EEPROM全体保存」ボタン

汎用コントローラに登録されているデータを全てバイナリファイル(拡張子DAT)に保存します  
(初期設定ファイル、ドライバーファイル、画面ファイル全て保存します)

もう一台の汎用コントローラに同じデータをコピーやファイルで保管したいときのために用意しました。  
汎用コントローラと通信しますので、汎用コントローラ電源を投入し、シリアルケーブルで接続してください。

「EEPROM全体保存」ボタンを押すと



汎用コントローラ内のデータを全て受信しますのでバーグラフが100に達するまでしばらくお待ちください。

バイナリ保存を途中でキャンセルする場合は、「Cancel」ボタンを押すことによりキャンセルされます。  
途中まで受信した場合でも、保存することができます。



「ファイル名」の行に保存するデータのファイル名を入力してから「保存」ボタンを押してください。  
拡張子(DAT)は入力しなくても自動的に入力されて保存されます。

保存するフォルダーを変更したい場合「保存する場所」の行で変更できます。

「キャンセル」ボタンを押すとファイル保存を実行しないでメイン画面に戻ります。(32ページへ)

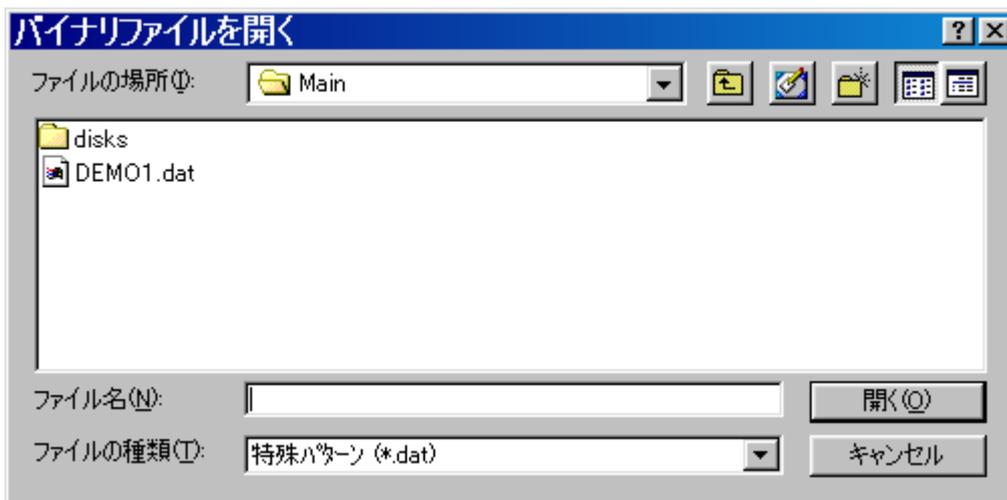
### 「EEPROM全体登録」ボタン

バイナリファイル(拡張子DAT)データを汎用コントローラに登録します。  
(初期設定ファイル、ドライバーファイル、画面ファイル全て登録します)

バイナリファイル(拡張子DAT)は前ページ「EEPROM全体保存」で保存したファイルが使用できます。  
もう一台の汎用コントローラに同じデータをコピーしたいときのために用意されました。

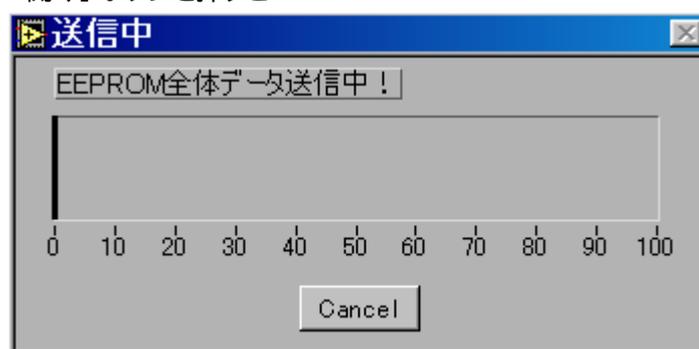
汎用コントローラには上書き登録になりますので、これを実行すると前のデータは消去されますので注意。  
汎用コントローラと通信しますので、汎用コントローラ電源を投入し、シリアルケーブルで接続してください。

「EEPROM全体登録」ボタンを押すと



「ファイル名」の行に登録するデータのファイル名を入力してから「開く」ボタンを押してください。または、表示されているファイル名をクリックして「開く」ボタンを押しても同様です。「キャンセル」ボタンを押すと登録を実行しないでメイン画面に戻ります。(32ページへ)

「開く」ボタンを押すと



汎用コントローラ内にデータを全て送信しますのでバーグラフが100に達するまでしばらくお待ちください。

「Cancel」ボタンを押すことにより送信がキャンセルされメイン画面に戻ります。(32ページへ)  
 (注：途中でキャンセルするとデータ登録も途中までしか登録しませんので動作が不安定になる可能性があります、できるだけ最後まで登録してください)

#### 「設定の登録」ボタン

初期設定ファイル(拡張子LCD)やドライバーファイル(拡張子MOT)を汎用コントローラに送受信したい場合押してください。

「設定の登録」ボタンを押すと画面が切り替わります。(35ページへ)

#### 「画面の登録」ボタン

画面ファイル(拡張子BMP)やICONデータを汎用コントローラに送受信したい場合押してください。

「画面の登録」ボタンを押すと画面が切り替わります。(43ページへ)

#### 「現在の画面」ボタン

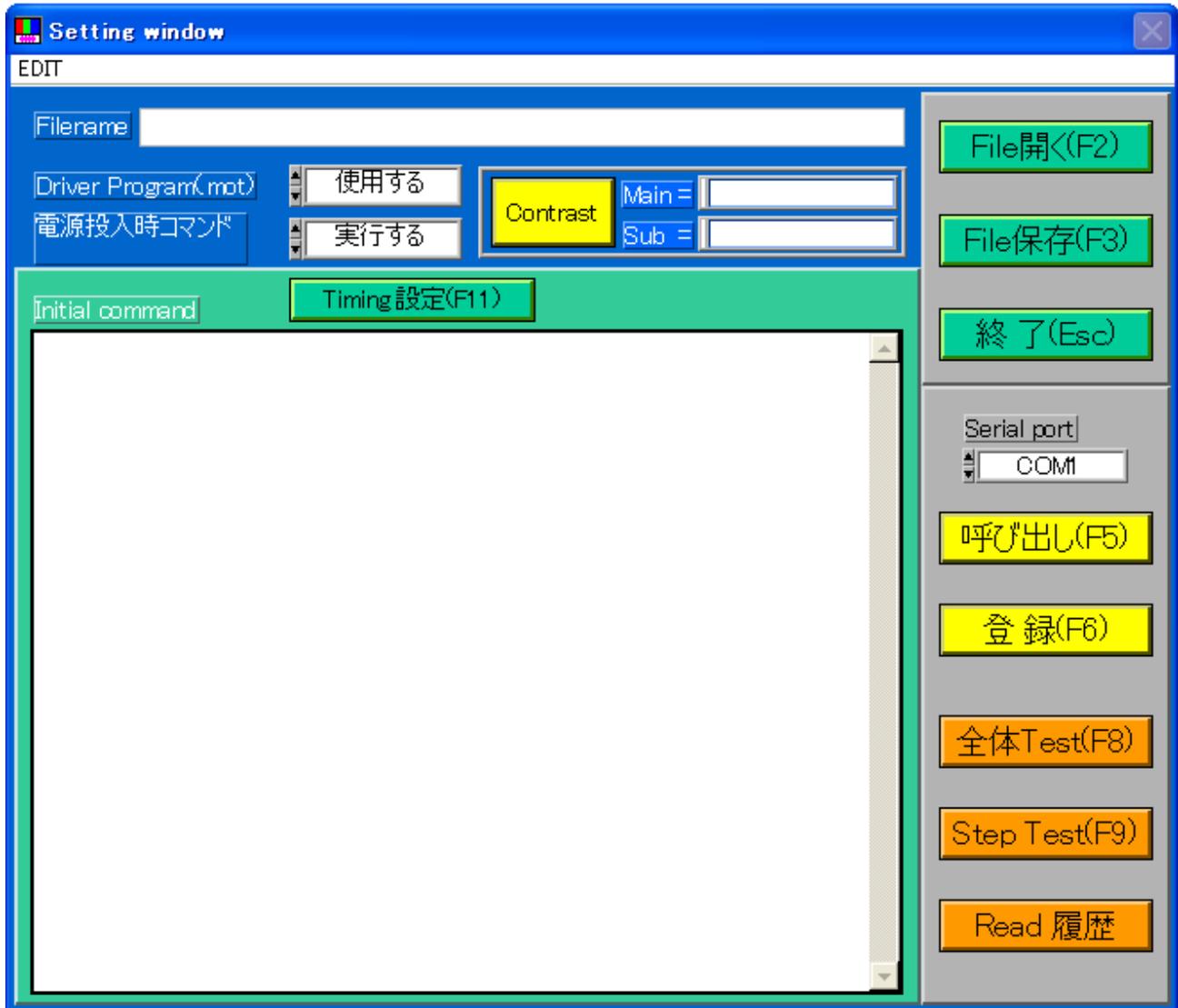
現在表示中の画面が何か知りたい時に押してください、表示画面を切り替える時も使用できます。

「現在の画面」ボタンを押すと画面が切り替わります。(51ページへ)

## 9.2 設定画面

Main 画面で「設定の登録」ボタンを押すと設定画面が表示されます。

設定画面では、LCDモジュールの初期設定ファイルの登録、編集、ドライバーファイルの登録ができます。汎用コントローラと通信しますので、汎用コントローラ電源を投入し、シリアルケーブルで接続してください。



初期設定ファイル（拡張子LCD）は、テキストファイルですのでメモ帳等ソフトでも編集できます。初期設定ファイルの作成は、7. 初期設定(LCD ファイル)で使用できるコマンドを参照してください。

### 「終了」ボタン

設定画面を終了させる時、押してください。（ESC キーでも可）

Main画面に戻ります。（32ページへ）

### 「SerialPort」ボタン

汎用コントローラと通信するために接続したポート番号に合わせてください。

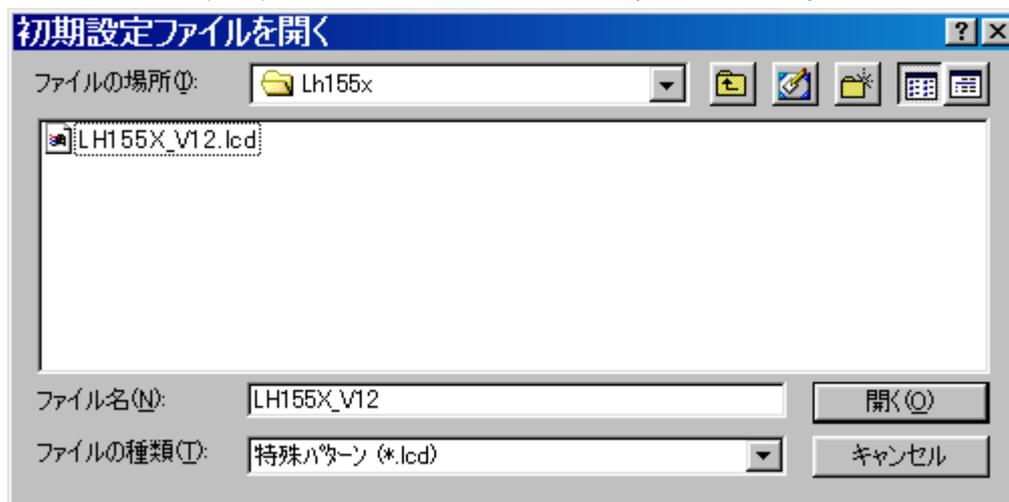
- ・RS232C、USB 接続 : COM1～COM7又はUSBで選択してください
- ・RAM 接続 : RAM\_2Mb、RAM\_8Mb、RAM\_16MBに選択してください

接続したポート番号と正しく合わせないと通信できません。

RAM 接続の場合、PCメモリーを仮想RAMとしますのでチェッカー無しでデータ作成可能(DAT作成用)

### 「File開く」ボタン

初期設定ファイル(LCD)を呼び出す時、押してください。(F2 キーでも可)

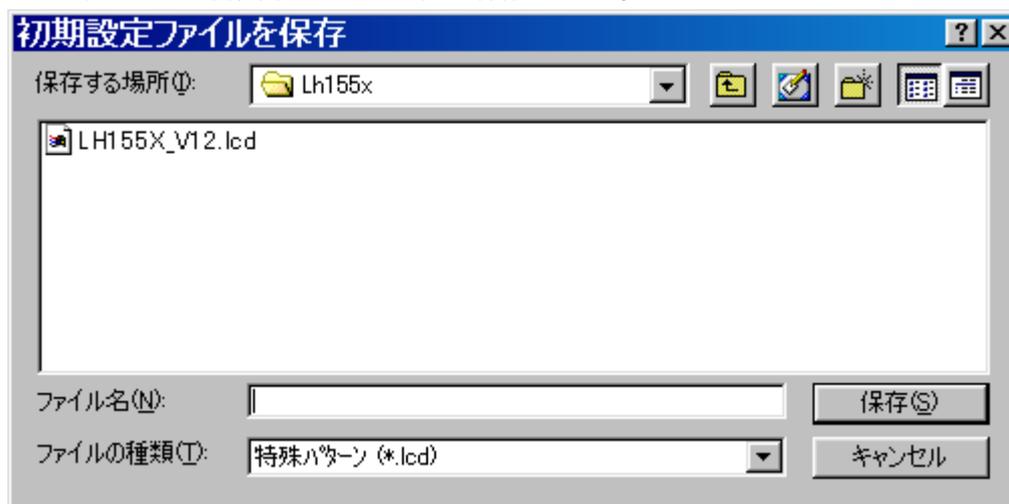


ファイルを呼び出しますと、「Initial command」行に初期設定ファイルの中身が表示されます。

### 「File保存」ボタン

「Initial command」行に表示されている内容を保存するとき、押してください。(F3 キーでも可)

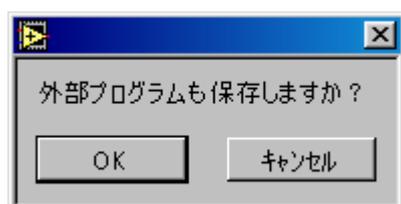
「呼び出し」ボタンで汎用コントローラに登録されている初期設定ファイルとドライバーファイルを呼び出している場合、両ファイルを別々に保存できます。



「ファイル名」の行に保存するデータのファイル名を入力してから「保存」ボタンを押してください。拡張子(LCD)は入力しなくても自動的に入力されて保存されます。

「キャンセル」ボタンを押すとファイル保存しないで戻ります。

続いてドライバーファイルも呼び出していれば、



「キャンセル」ボタンを押すと、ドライバーファイルは保存しません。

「OK」ボタンを押すと、



「ファイル名」の行に保存するデータのファイル名を入力してから「保存」ボタンを押してください。  
拡張子(MOT)は入力しなくても自動的に入力されて保存されます。  
「キャンセル」ボタンを押すとファイル保存しないで戻ります。

#### 「呼び出し」ボタン

汎用コントローラに登録されている初期設定ファイルとドライバーファイルを呼び出す場合、使用します。  
(F5 キーでも可)

「呼び出し」ボタンを押すと、まずドライバーファイルを受信します。



続いて初期設定ファイルを受信します。



正常に受信できていれば、「Initial command」行に汎用コントローラに登録されている初期設定ファイルが表示されます。

## 「登録」ボタン

汎用コントローラに初期設定ファイルとドライバーファイルを登録する場合、使用します。

(F6 キーでも可)

「登録」ボタンを押すと、



「キャンセル」ボタンを押すと、ドライバーファイルは登録しません、初期設定ファイルは登録します。すでに登録されているドライバーファイルと同じファイルを登録する場合、転送してもしなくても動作に影響はでませんので「キャンセル」できるようにしました。

転送時間を短縮するために用意ただけです。

新規にドライバーファイルを転送する場合は「OK」ボタンを押してください。

「OK」ボタンを押すと



ファイル名には、初期設定ファイルで設定したファイル名が入っています。

例) PDRIVER = "LH155X\_V12"

同ファイル名が入っているフォルダーを指定して、ファイルがあることを確認してから「開く」ボタンを押すとドライバーファイルが転送されます。

「キャンセル」ボタンを押すと、ドライバーファイルは転送しません。

続いて初期設定ファイルが転送されます。

ドライバーファイル転送をキャンセルしても初期設定ファイルの転送は実行されます。



転送するデータは、「Initial command」で表示されている行全てを登録します。  
注)「ドライバープログラム」「電源投入時実行コマンド」で指定した値も登録されます。

#### 「ドライバープログラム」の設定

汎用コントローラに登録されているドライバーファイルを使用するか、しないかの設定ができます。  
「登録」ボタンにより転送した時、有効になります。



「使用する」の場合、Auto / Manual SW、Step SWによる表示画面の表示、Up / Down SWによる電子ボリューム値の調整ができます (ドライバープログラム実行中)  
「使用しない」の場合、ドライバーファイルを登録してもドライバープログラムを実行しないようにします。  
注)基本的にドライバーファイルのデバッグ用に用意したもので通常は「使用する」で使用してください。

#### 「電源投入時実行コマンド」の設定

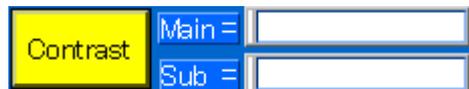
電源投入時に初期コマンドをモジュールに実行するか、しないかの設定ができます。  
「登録」ボタンにより転送した時、有効になります。



初期設定ファイルは、汎用コントローラの電源投入時に1度だけモジュールに設定データを送りますので電源投入時に初期設定ファイルをモジュールに転送を実行するか、しないかの設定です。

#### 「Contrast」ボタン

汎用コントローラのUp / Down SWは、コントラスト値を±1してモジュールに転送します。  
よって現在のコントラスト値を所得してみたい場合使用します。



コントラスト値の初期値は、初期設定ファイルのEV\_INIT コマンドで設定できます。  
例) EV\_INIT = 64 初期値を64にする  
注) Up / Down SWはドライバーファイルが登録されていないと有効になりません。

#### 「全体テスト」ボタン

「Initial command」行に表示されている全てのコマンドを実験的にモジュールに転送します。メモリーには登録しません。  
作成した初期設定ファイルが意図したとおりにLCDモジュールに初期設定できているか確認できます。

#### 「Step Test」ボタン

「Initial command」行に表示されているカーソルのコマンドだけモジュールに転送します。実行されるとカーソルが1行下に移動します。  
接続されているLCDモジュールの1命令だけを転送して変化を見る場合に有効です。

#### 「Initial command」内のカーソル

「Initial command」行では「↑」「↓」Key、カーソルが上下に移動できます。  
「Page Up」「Page Down」Keyで長い行でもページスクロールしますが、カーソルは移動しませんので注意が必要です。  
あるカーソル行位置で「Enter」Keyを押すと、1行編集が可能になります。



ここで1行の編集ができます。

「OK」ボタン又は「Enter」Keyを押すと編集行が有効になり1行が編集されます。

「Cancel」ボタン又は「ESC」Keyを押すと編集行が無効になり元の1行にもどります。

#### EDIT

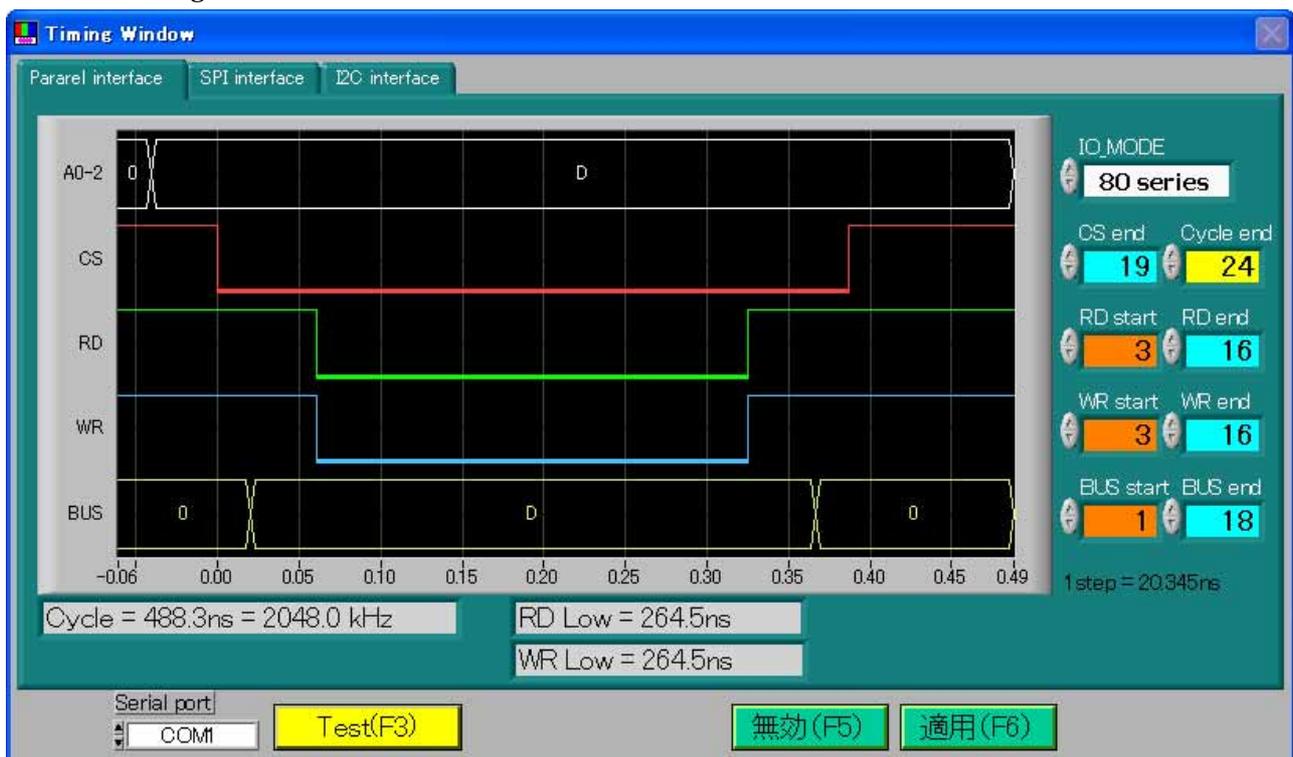
1line Cut	Ctrl+X
1line Copy	Ctrl+C
1line Paste	Ctrl+V
1line Add	Ctrl+I
1line Clear	

編集機能を使うと1行ごとのコピーや貼り付けができます。

#### 「Timing 設定」ボタン

インターフェイスのタイミングを調整する場合に使用します (F11 キーでも可) (注意 E チェッカーのみ有効)

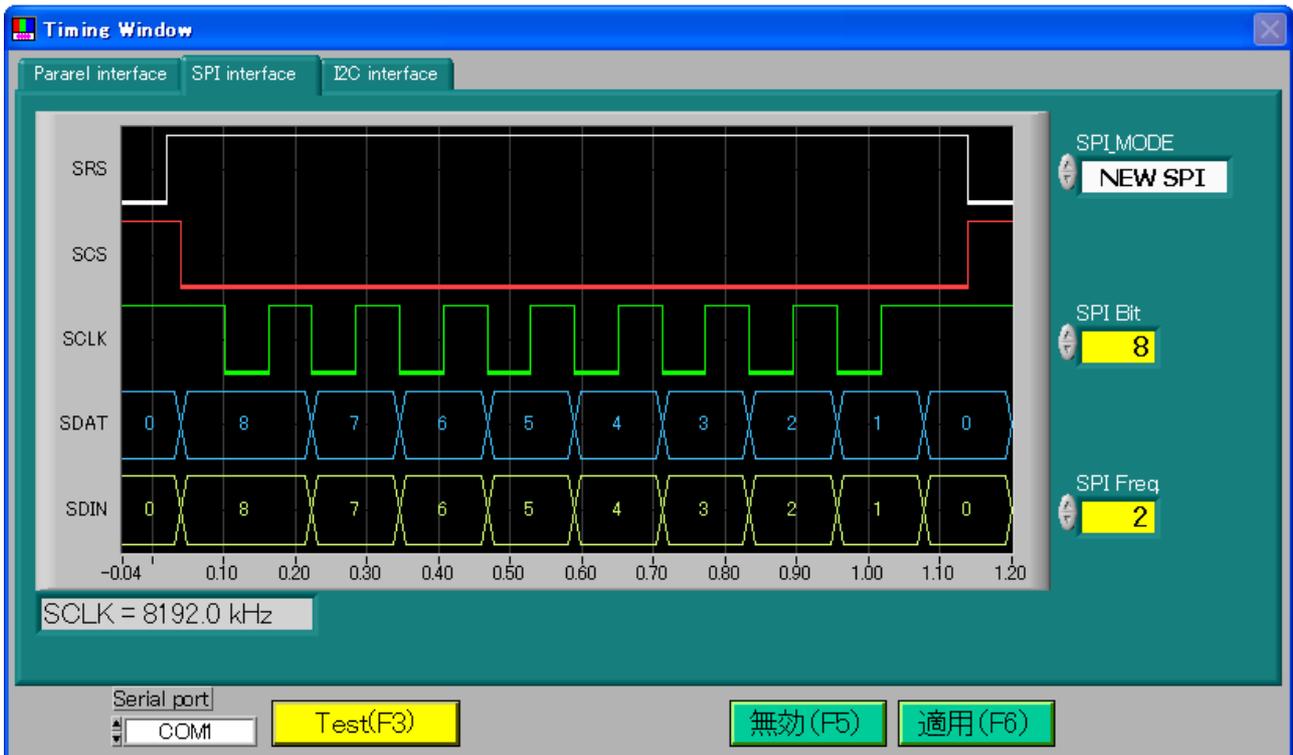
「Timing 設定」ボタンを押すと、



パラレル I/F で動作させる場合、CS 立下りを 0 秒として各波形のタイミングを変更可能です。

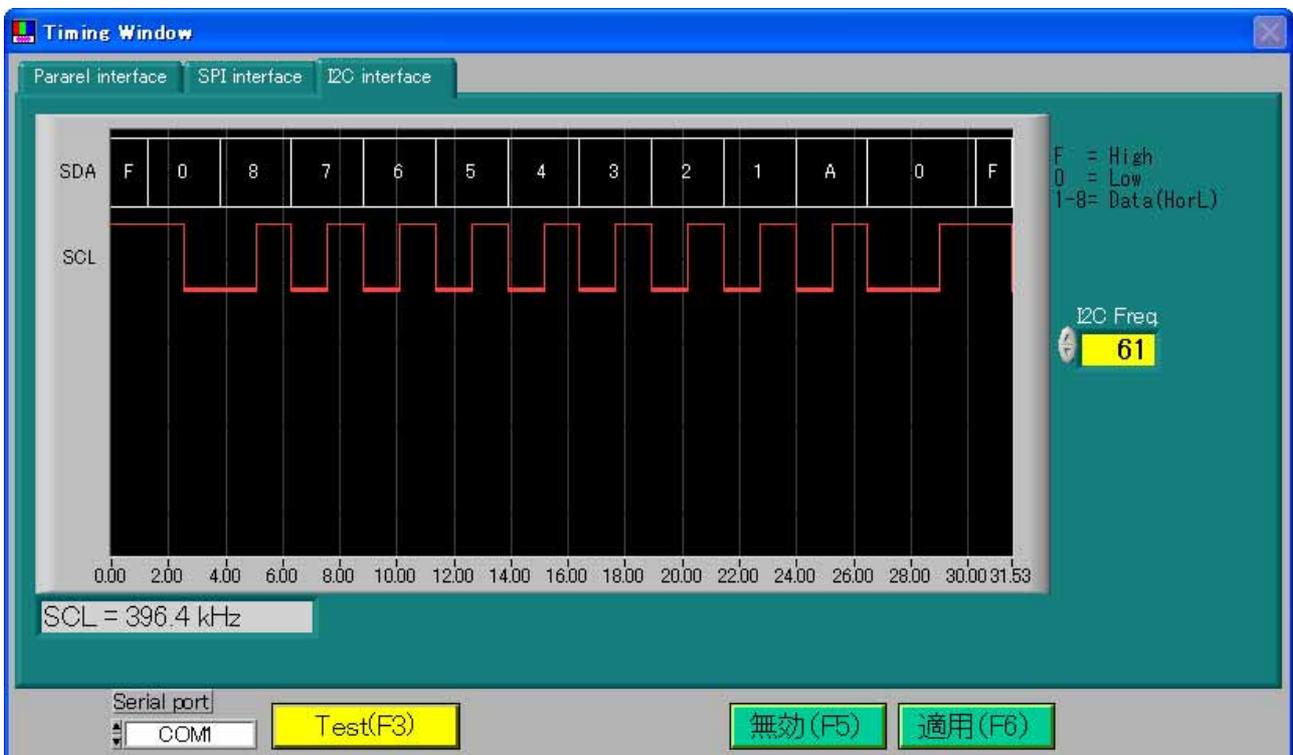
各数値 n とすると、n は 0 - 255 まで変更でき  $\text{Time} = n \times 20.345\text{ns}$  と 0~5 $\mu\text{s}$  まで変更可能です

- CS end = CS 立上り位置
- Cycle end = 1cycle 全体(他の数値より小さくしないでください)
- RD start = RD 立下り位置
- RD end = RD 立上り位置
- WR start = WR 立下り位置
- WR end = WR 立上り位置
- BUS start = DataBUS 出力初め位置
- BUS end = DataBUS 出力終了位置
- IO\_MODE = 80 系、68 系の波形グラフが変更させるだけで設定には影響しません  
(LCD ファイルの IO\_MODE 命令が I/F 種類の設定になりますので)



高速SPIでI/F出力させる場合、データビット数と周波数のタイミングを変更可能です。

- ・SPI\_MODE = OLD SPI : Dチェッカー互換モードで動作します  
NEW SPI : 変更したビット数と周波数で動作します
- ・SPI BIT = 6 ~ 16 bit データとして出力変更が可能
- ・SPI Freq =  $24.576\text{MHz} / (n + 1)$  周波数でSCLK端子が動作するように変更が可能



I2CでI/F出力させる場合、周波数のタイミングを変更可能です。

- ・I2C Freq =  $24.576\text{MHz} / (n + 1)$  周波数でSCL端子が動作するように変更が可能

#### 「TEST」ボタン

これは 40 ピンコネクタが出力状態(ON)のとき実験的にタイミングを変更させる Test 動作です。通常は初期設定動作(LCD ファイル)からタイミング変更を有効にしないとなりますが、確認用として用意しました。

(注意) D チェッカーでは使用しないでください、壊れることはありませんが出力動作がおかしくなります  
もし D チェッカーで使用してしまった場合、電源を入れ直してください。

#### 「無効」ボタン

タイミング変更した数値を無効にして「Timing Window」画面を終了し設定画面に戻ります  
各数値は「Timing Window」画面表示前に戻ります

#### 「適用」ボタン

タイミング変更した数値を有効にして「Timing Window」画面を終了し設定画面に戻ります  
各数値は保管されていますが、チェッカーにはまだ設定されていない状態です。

各数値のタイミングをチェッカーで有効にするには「適用」ボタンの後、設定画面の  
「登録」ボタンで転送させてください(LCD ファイルと同時に転送される)  
これで初めて有効になります

(注意) D チェッカーで登録しても問題ありません。登録した場合、各数値のタイミングは  
メモリーには登録されますが D チェッカーでは無視します

(E チェッカーは ON/OFF ボタンでメモリーに登録された各数値のタイミングを有効にします)

#### Serial port

汎用コントローラと通信するために接続したポート番号に合わせてください。

・RS232C、USB 接続 : COM 1 ~ COM 7 又は USB で選択してください

・RAM 接続 : RAM\_2Mb、RAM\_8Mb、RAM\_16MB に選択してください

接続したポート番号と正しく合わせないと通信できません。

RAM 接続の場合、PC メモリーを仮想 RAM としますのでチェッカー無しでデータ作成可能(DAT 作成用)

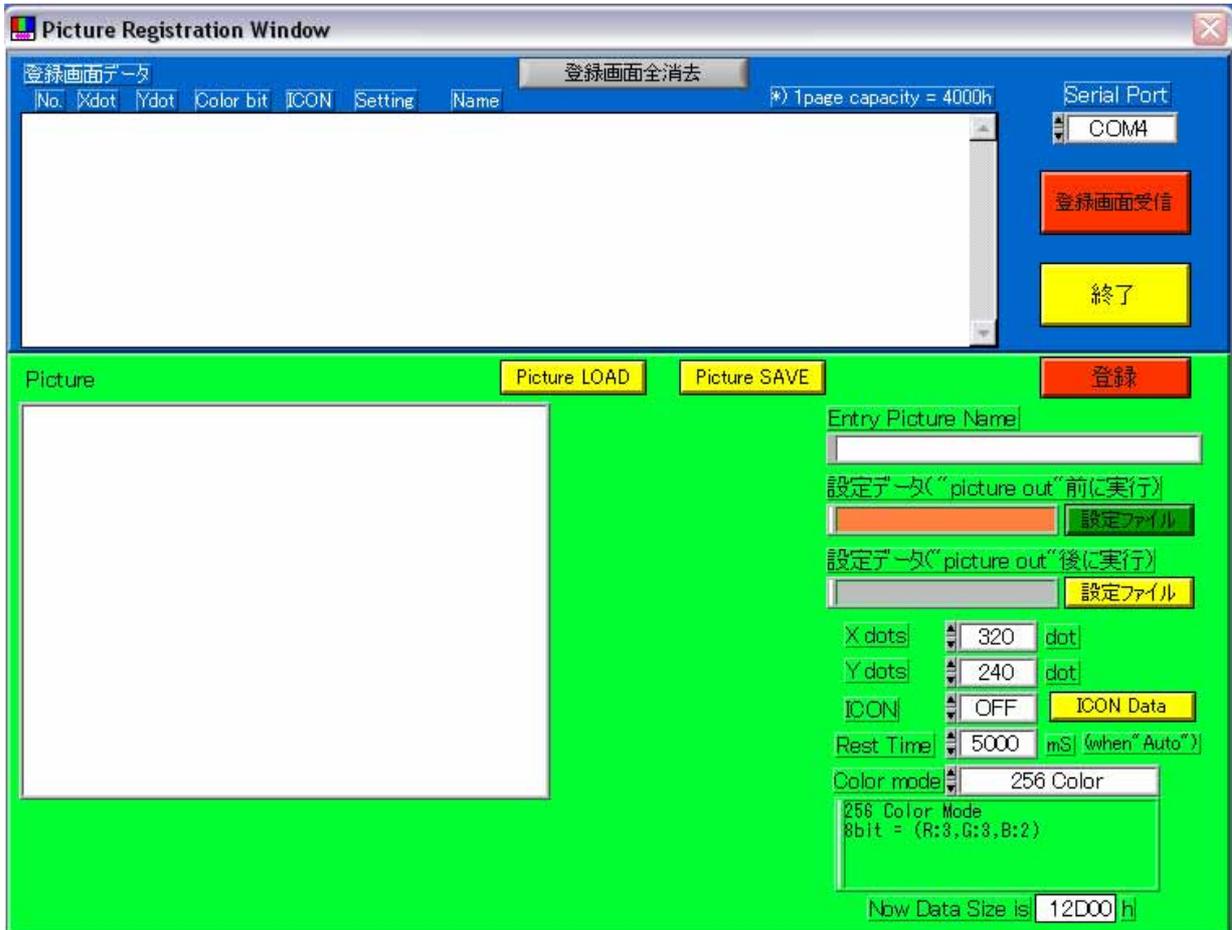
### 9.3 画面の登録

Main 画面で「画面の登録」ボタンを押すと登録画面が表示されます。

登録画面では、表示データの登録・差し替え、ICONデータの登録・編集、1画面毎のAUTO時の  
制止時間が登録できます。

汎用コントローラと通信しますので、汎用コントローラ電源を投入し、シリアルケーブルで接続してください。

#### 登録画面



#### 「終了」ボタン

登録画面を終了させる時、押してください。

Main画面に戻ります。(32ページへ)

#### 「SerialPort」ボタン

汎用コントローラと通信するために接続したポート番号に合わせてください。

- ・RS232C、USB 接続 : COM1 ~ COM7 又は USB で選択してください
- ・RAM 接続 : RAM\_2Mb、RAM\_8Mb、RAM\_16MB に選択してください

接続したポート番号と正しく合わせないと通信できません。

RAM 接続の場合、PC メモリーを仮想 RAM としますのでチェッカー無しでデータ作成可能(DAT 作成用)

#### 「登録画面受信」ボタン

汎用コントローラに登録されている画面データのリストを表示します。

画面データを登録する場合、このリストが表示されていないと登録できませんので  
初めに必ず実行しなければなりません。

正常にリストを受信すると、

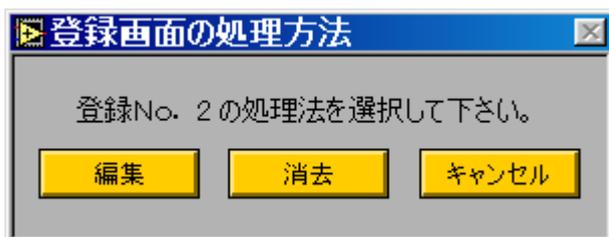
登録画面データ							
No.	Xdot	Ydot	Mode	ICON	設定	Name	注1登録画面の容量は4000hです
1	128	160	256Color	OFF	あり	文字128126.bmp	
2						文字128126.bmp	
3	120	161	256Color	OFF		51285_Picture.bmp	
4						51285_Picture.bmp	
5	120	160	256Color	OFF	あり	face_tel.bmp	
6						face_tel.bmp	
7							
8							

- Xdot・・・登録画面のX方向ドット数です。
- Ydot・・・登録画面のY方向ドット数です。
- Colorbit・・・登録画面の使用色数です。
- ICON・・・登録画面のICONデータも出力ON/OFFしているかです。
- 設定・・・登録画面といっしょに登録されている設定ファイルがあるか無いかです。
- Name・・・登録画面の名称です。(通常はBMPファイル名です。任意に変更できます)

### 「登録画面データ」領域

汎用コントローラに登録されているデータを編集・消去したい場合、「登録画面データ」領域のカーソル位置でダブルクリックすると処理を選択できます。

例えばNo 2の位置でダブルクリックすると

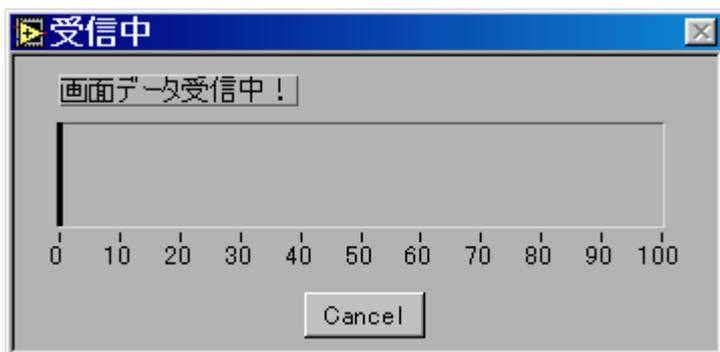


「キャンセル」ボタンで、何も処理しないで戻ります。

「消去」ボタンで、選択した登録Noの画面を消去します。

「編集」ボタンで、選択した登録Noの画面を受信してPicture領域等に表示されます。

「編集」ボタンを押したとき、



100%になるまで、しばらくお待ちください。  
受信が完了すると登録されているデータを確認できます。

### 「BMP File LOAD」ボタン

登録したい画面データを読み込みます (BMPファイル読み出し)

「BMP File LOAD」ボタンを押すと



ファイルを選択して「開く」ボタンを押すと、BMPファイルが読み込まれて Picture領域に表示されます。  
NameにBMPファイル名が入力されます。Nameは変更することができます。  
Nameは半角で64文字、全角で32文字まで入力できます。

### Picture領域について

Picture領域は、最大2048×2048ドットまでです。それ以上のデータを読み込んでも2048×2048ドット分しか読み込みません。

### 「BMP File SAVE」ボタン

Picture領域に表示されているデータをBMPファイルで保存します。  
ICONデータはファイルに保存できません。

「BMP File SAVE」ボタンを押すと



「ファイル名」の行に保存するデータのファイル名を入力してから「保存」ボタンを押してください。  
拡張子 (BMP) は入力しなくても自動的に入力されて保存されます。  
「キャンセル」ボタンを押すとファイル保存しないで戻ります。

### 「Xdots」の設定

汎用コントローラに登録する画面データの横方向のドット数を設定できます。

8dot~2048dotまで可変できます。

可変しますと「現在のDataSize」の容量が変化しますのでコントローラの空き容量の大きさに注意して調整してください

### 「Ydots」の設定

汎用コントローラに登録する画面データの縦方向のドット数を設定できます。

4dot~2048dotまで可変できます。

可変しますと「現在のDataSize」の容量が変化しますのでコントローラの空き容量の大きさに注意して調整してください

### 「ICON」の設定

ICONデータをLCDモジュールに出力するか選択できます。

ICON出力を持つLCDモジュールで有効です。

### 「静止時間」の設定

汎用コントローラに登録する画面データのAUTO時の静止時間を設定できます。

1~60000mSまで可変できます。

これにより各画面毎に静止時間を可変することができます。

### 「Color Mode」の設定

汎用コントローラのメモリーに登録する画面データの色数を設定できます。

モノクロ2値(1ドット=1ビット)

:黒=1、白=0として変換し登録します、その他の色は白として変換します。

モノクロ4階調(1ドット=2ビット)

: Color = (Red + Blue + Green)/3 と計算して

: 0 color 63...11データ(黒)

: 64 color 127...10データ

: 128 color 191...01データ

: 192 color 255...00データ(白)

: と変換して登録します。

SRCカラー(1ドット=2ビット)

: 白色...00データ

: 赤色...01データ

: 青色...10データ

: 緑色...11データ

: と変換して登録します。その他の色は白色と判断します。

モノクロ16階調(1ドット=4ビット)

: Color = (Red + Blue + Green)/3/16 と計算して

: 0000(黒)~1111(白)と変換して登録します

16色カラー(1ドット=4ビット)

: 黒                   ...0000      薄い白      ...1000

: 薄い青               ...0001      青           ...1001

: 薄い赤               ...0010      赤           ...1010

: 薄い紫               ...0011      紫           ...1011

: 薄い緑               ...0100      緑           ...1100

: 薄い水色              ...0101      水色       ...1101

: 薄い黄                ...0110      黄           ...1110

: 灰色                  ...0111      白           ...1111

: と変換して登録します。

256色カラー（1ドット=8ビット）

: 赤8階調(3BIT)、緑8階調(3BIT)、青4階調(2BIT)  
: と変換して登録します。

65536色カラー（1ドット=16ビット）

: 赤32階調(5BIT)、緑64階調(6BIT)、青32階調(5BIT)  
: と変換して登録します。

フルカラー（1ドット=24ビット）

: 赤256階調(8BIT)、緑256階調(8BIT)、青256階調(8BIT)  
: と変換して登録します。

ColorModeを可変しますと「現在のDataSize」の容量が変化しますのでコントローラの空き容量の大きさに注意して調整してください

### 「ICON詳細」ボタン

ICON出力を持つLCDモジュールの場合、ここでデータを作成できます。

データは256ドット分まで登録でき1ドットで0～3まで設定できますので4階調のICONデータまで作成できます。

モノクロICON出力モジュールの場合

: 0 or 1のデータ = 0としてLCDに出力されます。  
: 2 or 3のデータ = 1としてLCDに出力されます。

4階調ICON出力モジュールの場合

: 0 = 00としてLCDに出力されます。  
: 1 = 01としてLCDに出力されます。  
: 2 = 10としてLCDに出力されます。  
: 3 = 11としてLCDに出力されます。

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
00h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
40h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
50h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
70h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
80h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
90h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A0h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E0h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C0h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D0h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E0h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F0h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

データを変更後「適用」ボタンで有効になり I C O N 設定 W i n d o w が閉じます。  
「キャンセル」ボタンで変更する前のデータに戻して I C O N 設定 W i n d o w が閉じます。

### 「設定ファイル」ボタン

登録する画面データを表示する際に、設定ファイルのコマンドを送ることができます。  
例えば、この表示の時のみパーシャル表示モードで動作するようにするとかスクロール表示させるとか使用するモジュールの機能を使って動作の制御ができます。



設定画面では、電源投入時にだけモジュールが動作するために必要な初期設定コマンドを設定しますがここでは、モジュールが動作中に表示機能を切り換えたりする部分的なコマンドの記述だけで良いので登録できるコマンド数が最大500までになっています。  
(設定画面の初期設定ファイルでは最大1536まで登録可能)  
コマンドの記述方法は初期設定ファイルと同じです。

「FileLoad」「FileSave」ボタンで設定ファイル(拡張子LCD)の読み込み、保存ができます。  
編集は設定ファイルと同じで1行ごとの編集ができます。(設定画面の項を参照)

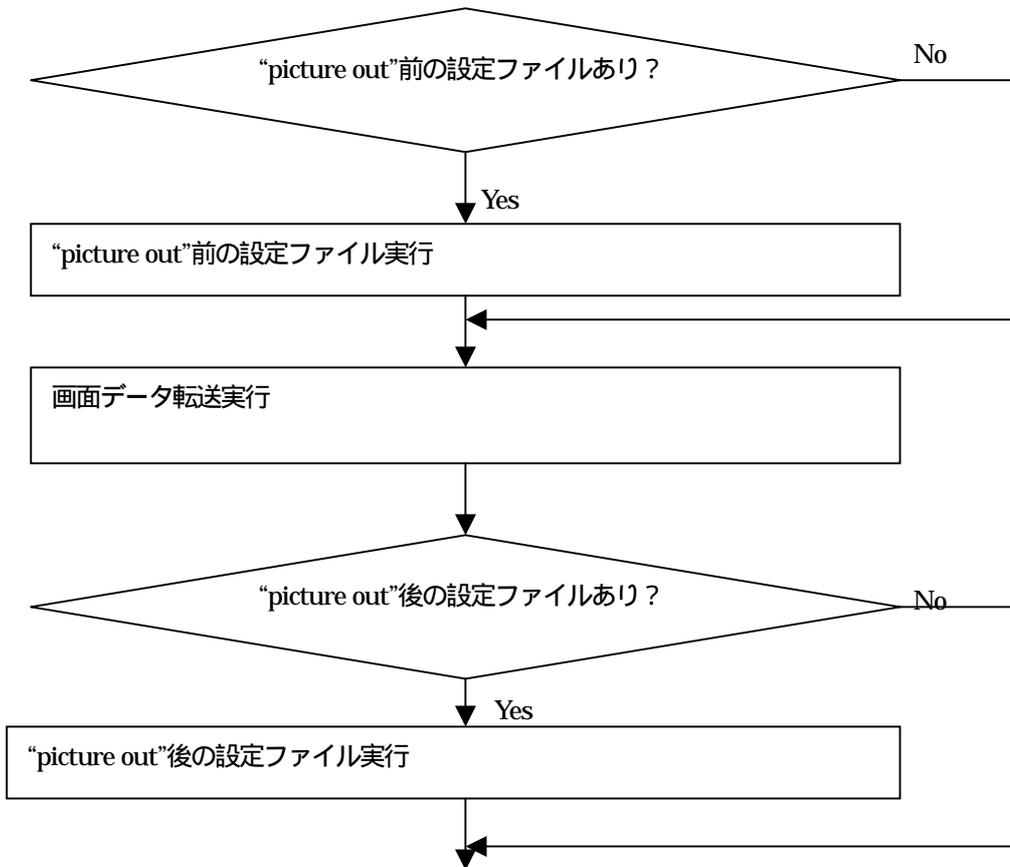
#### ・「有効」ボタン

この設定ファイルを画面データといっしょに登録したい場合、有効ボタンで終了します。  
登録画面内の実行コマンド Windows が閉じて登録画面の設定 Windows に戻ります。

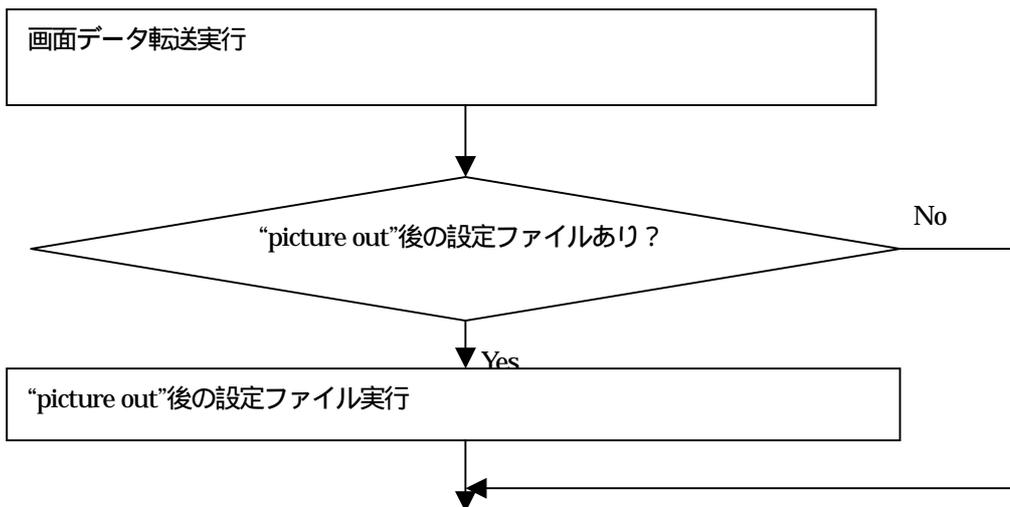
#### ・「無効」ボタン

この設定ファイルを登録するのをやめる場合、無効ボタンで終了します。  
登録画面内の実行コマンド Windows が閉じて登録画面の設定 Windows に戻ります。

登録する画面データを表示する際に、設定ファイルは2つ登録することができます。  
但し2つ設定して動作するのはタイプD チェッカー-BiosVersion4.70 以上だけです。



それ以外の古い BiosVersion では"picture out"後の設定ファイルのみ使用できます。

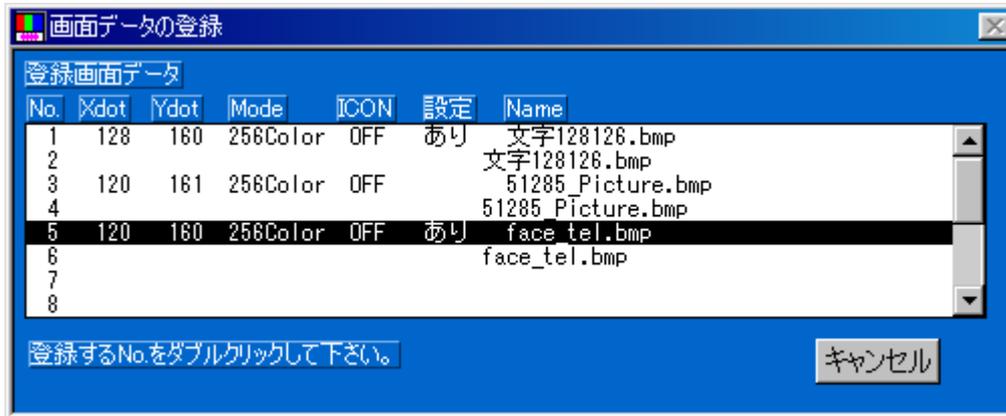


### 「登録」ボタン

設定したPicture領域のデータ、ICONデータ、ColorMode、静止時間、設定ファイルを1画面として汎用コントローラに登録します。

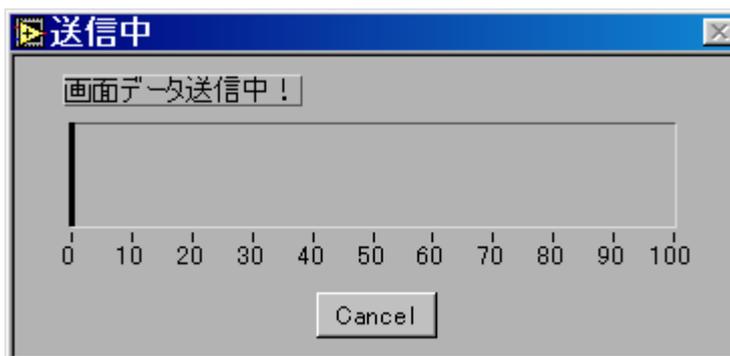
「現在のData Size」とコントローラの空き容量を確認して登録してください。

「登録」ボタンを押すと



「キャンセル」ボタンを押すと、データを転送しないで戻ります。

登録したいNo 1 ~ 15にカーソルを移動してダブルクリックするとデータが転送されます。



正常に転送されると「登録画面データ」領域のリストが更新されます。

## 9.4 現在の画面

Main 画面で「現在の画面」ボタンを押すと現在の画面が表示されます。  
現在の画面では、表示中の画面データが何画面目か知るか、逆に何画面目を強制的に表示させることができます。  
但し BiosVersion3.20 以上のコントローラでしか使用できません。



### 「終了」ボタン

終了させるとき押してください。Main 画面に戻ります。(32ページへ)

### 「SerialPort」ボタン

汎用コントローラと通信するために接続したポート番号に合わせてください。

- ・RS232C、USB 接続 : COM1 ~ COM7 又は USB で選択してください
- ・RAM 接続 : RAM\_2Mb、RAM\_8Mb、RAM\_16MB に選択してください

接続したポート番号と正しく合わせないと通信できません。

RAM 接続の場合、PC メモリーを仮想 RAM としますのでチェッカー無しでデータ作成可能(DAT 作成用)

### 「現在の画面受信」ボタン

現在表示中の画面データ名がカーソル位置になります。

カーソル位置にない場所でダブルクリックするとその画面データが強制的に表示されます。

## 1.0.PCソフトウェアのVersionについて

### Version1.00 から Version1.20 への変更点

1. Version1.00 での画面データ登録は、モノクロ・モノクロ4階調・SRC4色変換登録の3通りでしたが、16色・256色変換登録を追加して計5通りにしました。
2. 設定初期設定ファイル(拡張子LCD)で16進入力の数値がモジュールに転送されない場合があることを修正。  
例) Version1.00 の場合  
「DOUT[1]=FBH」をLCDに送る場合「DOUT[1]=00H」として送られていた。Bが含まれる値を送ると全て0で転送されていました(Bが含まれない場合は正常に転送していました)
3. 設定の登録画面でイニシャライズ実行コマンドの編集操作性の向上  
「`<code>`」でカーソル移動できるようにしました。

### Version1.20 から Version1.50 への変更点

1. 1画面の登録が最大3F00h=16128byteだったのを最大BF00h=48896byteまで登録できるようにしました。
2. 登録画面で、BMPファイル、アイコンデータ、設定ファイル(拡張子LCD)の3種類を1つとして登録できるようにしました。  
動作の順番は、  
BMPデータをLCDに表示  
アイコンデータがあり、使用ICも対応していればLCDに表示  
設定ファイルがあれば設定コマンドを転送  
この機能の追加により、各画面表示毎にパーシャルモード表示やスクロール表示機能の動作をさせることができます。
3. BIOS VERSION1.50 について
  - ・16bit出力を選択すると上位8bit 下位8bitが逆になるのを修正(この場合、ドライバファイルもVersion1.30以上を使ってください)
  - ・画面登録に含まれる設定ファイルの実行をできるようにしました。

### Version1.50 から Version1.51 への変更点

1. 初期設定ファイル内、EV\_INIT命令のデータの上限を255から65535に変更(16ビットデータを持つ電子ボリューム機能のICに対応するため)

### Version1.51 から Version2.00 への変更点

1. VDD、VEE電圧を変更できる命令を追加(VDD\_VOLT or VEE\_VOLT)
2. VDD、VEE電流リミッタできる命令を追加(VDD\_LIMIT or VEE\_LIMIT)
3. 電子ボリューム用VRのリミッタの設定命令を追加(EV\_MAX)
4. ファイルを開く、保存時にフォルダが前回と違うフォルダになることを修正
5. 設定ファイルの中身を修正できる機能を追加(カット&コピー&1行追加等)

### Version2.00 から Version2.10 への変更点

1. 画面と同時に登録する設定ファイルでEV\_INITを使えるようにしました。
2. 画面登録の簡易表示で設定ファイルがあるなし表示を追加
3. メモリ4Mb、6Mbも認識できるようにしました。

### Version2.10 から Version2.20 への変更点

1. 画面データ登録は、モノクロ・モノクロ4階調・SRC4色・16色・256色変換の5通りでしたが65536色変換を追加して計6通りにしました。
2. COM1~4の設定をCOM1~7に変更

#### Version2.20 から Version2.50 への変更点

1. USB ケーブルによる通信機能を追加

#### Version2.50 から Version2.60 への変更点

1. 電子ボリュームの Main/Sub 2 つが独立動作する設定ができるようにした。  
(サブ用「EV\_INIT2」、「EV\_MAX2」の命令追加)
2. VDD の電圧 1.80 ~ 5.00V 電流リミッタが Max50mA までは変わらずですが  
VEE の電圧 0.0 ~ 19.0V 電流リミッタは Max200mA までは設定できるように変更。

#### Version2.60 から Version3.00 への変更点

1. 折り畳み携帯電話の通信が可能(BIOS Version3.20 以上)  
(旧 Ver2.60 ではページ境目の書込みに失敗する場合があります)
2. VDD、VEE 電圧最小単位を 0.01V に対応
3. ドライバー拡張データコマンドの追加
4. ドライバー容量拡張 0F00h 1F00h(BIOS Version3.20 以上)
5. 現在の表示画面を読めるようにした(BIOS Version3.20 以上)
6. 初期設定ファイルのテキストデータも登録可能(BIOS Version3.00 以上)
7. 画面データフルカラー色変換を追加して計 7 通りにした

#### Version3.00 から Version3.01 への変更点

1. 登録済みの表示画面をパソコンから切り換えることができるようにした。
2. USB ドライバーを修正(USB 通信エラーを起こりづらくした)
3. 初期設定 TEXT の登録がうまくいかないことを修正

#### Version3.01 から Version3.02 への変更点

1. ドライバー拡張命令を送っていなかったのを修正

#### Version3.02 から Version3.03 への変更点

1. TEXT データの登録に失敗することがあるのを修正
2. 全体登録と全体保存の通信強化
3. 初期設定登録時、ドライバー登録をキャンセルしたら暴走してしまうことがあることを修正

#### Version3.03 から Version3.04 への変更点

1. 電源投入時のバックライト調整機能追加  
「MAIN\_BKLT」「SUB\_BKLT」命令追加

#### Version3.04 から Version3.05 への変更点

1. 画面データの容量計算の修正(320\*240dots 以上が計算合わなかった)
2. 画面データ 800 x 600 ドットまで登録可能

#### Version3.05 から Version3.06 への変更点

1. 新電圧設定登録データ計算間違い修正
2. EXPDATA1 ~ 4 が BIOS Version1.57 で無効になっていたことを修正

#### Version3.05 から Version3.06 への変更点

1. 初期設定の最終行が登録できなかったことを修正
2. シリアル転送用 SOUT 命令追加
3. IO\_MODE に I80\_20,M68\_20 命令追加

- 4 . BIT\_SET,BITRSET 命令に SRCS 追加
- 5 . 各種 FILE load/save エラーダイアログ表示追加
- 6 . 画面簡易受信で TEXT のサイズも判定して計算に修正

#### Version3.06 から Version3.10 への変更点

- 1 . lcd ファイル LOAD 時最終行が消えてしまうことを修正
- 2 . 「IO\_MODE = M68\_20」「IO\_MODE = I80\_20」命令追加
- 3 . 「BIT\_SET = SRCS」「BITRSET = SRCS」命令追加
- 4 . 「DOUT」「DTIN」命令 20 ビット対応(まだ USB の場合機能しない)
- 5 . 「SOUT[0,FS08B]=7ffh」命令追加
- 6 . 各種 FILE load/save エラーダイアログ表示追加
- 7 . 簡易受信で、text データのサイズもチェックするように修正

#### Version3.10 から Version3.15 への変更点

- 1 . 「DOUT」「DTIN」命令 20 ビット USB でも対応にした
- 2 . 「EXOUT[7,0000h)」命令追加
- 3 . 「EXPIN[7,0000h)」命令追加
- 4 . 呼び出し時、EXPDATA 命令が取り込めないことを修正
- 5 . 「POWER=ON、POWER=OFF」命令追加

#### Version3.15 から Version3.20 への変更点

- 1 . 日本語と英語どちらでも使用できるように切替ボタンを用意しました。
- 2 . 表示画面(BMP)表示が設定カラーに合わせて色合いを見れるようにしました。
- 3 . 画面に登録する LCD ファイルを 2 つ登録できるようにした。  
BMP 画面転送の前と後に動作する LCD 設定を登録できます。
- 4 . 呼び出し時、EXPDATA 命令が取り込めないことを修正
- 5 . 「LOOP\_BK2 ~ LOOP\_BK10」命令を追加
- 6 . 「EXPMODE」命令追加・・・data = 0:通常 SCLK 、1:SCLK
- 7 . 「CLK\_OUT」命令追加・・・E 端子に CLK 出力 Freq = 12.288MHz/(data+1)  
入力値は 0 ~ 2046 までなので計算式より 12.288MHz ~ 6kHz まで設定できます。

#### Version3.20 から Version3.21 への変更点

- 1 . USB 通信の強化 (しかしあまり変わらなかった)
- 2 . 登録画面全消去をメモリー全体消去から登録画面メモリーのみ消去するように変更

#### Version3.21 から Version3.22 への変更点

- 1 . 電流リミッタの設定を max800mA->1500mA に修正 (タイプ D2、E 対応用)
- 2 . 登録画面全消去をメモリー全体消去から登録画面メモリーのみ消去するように変更  
で C チェッカーに 8Mb メモリを使用した場合動作しないことを修正

#### Version3.22 から Version3.23 への変更点

- 1 . 「VBB\_VOLT」命令追加
- 2 . 「VBB\_LIMIT」命令追加

#### Version3.23 から Version3.25 への変更点

- 1 . 「VDD\_VOLT」「VEE\_VOLT」「VBB\_VOLT」命令を初期設定で何度も使えるようにした
- 2 . 「VDD\_LIMIT」「VEE\_LIMIT」「VBB\_LIMIT」命令も初期設定で何度も使えるようにした  
これにより電源出力のタイミングが変更可能

Version3.25 から Version3.26 への変更点

- 1 . 「VDD\_LoLIMIT」「VEE\_LoLIMIT」「VBB\_LoLIMIT」命令の追加

Version3.26 から Version4.00 への変更点

- 1 . 「IO\_MODE」変数追加(DigitalRGB,I80\_16\_Hi,I80\_16\_Hiz 等)
- 2 . 「TU\_WAIT」命令追加 ( 1=10us )
- 3 . 「EXPMODE」変数追加(Encode,POT,IIC,Busy,SDin,Falling,Rising or 0-15)
- 4 . 「CLK2X\_OUT」命令追加 ( 24.576MHz/(data+1) )
- 5 . 「EXPDATA5-8」命令追加
- 6 . 「Bit\_set」「Bitrset」変数追加(RDE,WRRD,ECLK,CS,DB01...DB15,SDAT,SCLK,SCS,SRS)
- 7 . 「VDD\_IFlimit」命令追加
- 8 . (バグ) 設定呼出で「EXIN」になる場合あったので-「EXPIN」に修正
- 9 . 「Read window」画面追加(Read 命令のみ履歴表示)
- 1 0 . 「SerialPort」にRAM\_2Mb、RAM\_8Mb、RAM16Mb を追加(DAT ファイルの編集用)
- 1 1 . 「Timing window」画面追加作成(I/F のタイミング調整可能)
- 1 2 . USB ドライバの VersionUP

Version4.00 から Version4.01 への変更点

- 1 . 画面の登録で Xdots と Ydots の最大値が 800 以下になる場合があるのを max2048 にした

## 1.1. チェッカーの BiosVersion について

### 汎用基板の BiosVersion

- BiosVersion4.00(2002/12/6)  
USB 通信機能の追加 (動画未対応)  
FlashMemory 2 Mb から 8Mb に変更  
IO\_MODE=I80\_20,M68\_20 命令対応  
SOUT[0,HS08B]命令対応(HS,FS,MS,LS:08B,09B,10B,11B)

---

- BiosVersion4.02(2003/2/11)  
動画命令を受け付けるように修正

---

- BiosVersion4.03(2003/2/14)  
WR 信号出力の立ち上がりと DATABUS 信号の出力 OFF が同時だったため  
LCD モジュールが不安定だったのを DATABUS 出力 OFF100ns 遅らせるように修正(PLD 修正)

---

- BiosVersion4.04(2003/3/4)  
POWER=ON , POWER=OFF 命令を使えるようにハード・ソフト変更  
(但し VDD OFF するだけなので放電作用は無い)

---

- BiosVersion4.05(2003/4/9)  
FlashROM 8Mb(MBM29F800BA) 16Mb(MBM29F160TE)自動判定追加  
次画面の切替検索ルーチンの最適化

---

- BiosVersion4.07(2003/7/10)  
ドライバー(mot)に連続上書き機能がある場合、対応するように追加

---

- BiosVersion4.10(2004/8/17)  
画面に登録する LCD ファイル 2 種類機能するようにした (画面転送前 LCD 実行追加)  
「LOOP\_BK2」 ... 「LOOP\_BK15」 命令追加  
「CLK\_OUT」 命令追加 : (12.228MHz)0<=data<=8191(min1.5kHz)  
Flash\_PageCopy 高速化

### D タイプチェッカーの BiosVersion

- BiosVersion4.50(2002/12/11)  
PC-CARD メモリ(旧 2Mb , 新 8Mb) 2 種類に対応  
電圧調整機能修正(VDD : 1.80V- 5.00V 最小単位 0.01V)  
(VEE : 0.00V-19.00V 最小単位 0.01V)  
出力コネクタを 30pin->34pin に変更(PinNo.1-30 までは旧と完全互換)  
増加した 4pin はシリアル 8,9,10,11bit 出力と可変可能  
CLK speed も 12MHz,6MHz,3MHz,1.5MHz と可変可能  
又は DataBus20bit としての拡張用バス

---

- BiosVersion4.51(2002/12/15)  
パソコン側ソフト Version3.06 以下で電圧設定時 LCD 表示器に 0.00V と表示されることを修正  
WR 端子の出力タイミングの変更(PLD 修正)  
電源投入時、8Mb メモリが書き込めないことがあることを修正  
データバスの波形が鈍らないように修正(PLD 修正)

- 
- BiosVersion4.52(2002/12/19)  
2 Mb メモリのある一部の場所だけ BMP 表示画面がおかしくなることを修正
- 
- BiosVersion4.53(2002/12/19)  
旧ドライバー(MOT ファイル)が動作しないことを修正(MOT ファイル Version1.30 近辺のもの)
- 
- BiosVersion4.54(2002/12/28)  
パワーOPAMP を TA8406P に変更(VCC 電流 100mA 以上使用できなかったことを修正)  
電源投入時 50ms たってから初期設定実行に変更  
電源ショートの場合、直ぐに OFF するように変更
- 
- BiosVersion4.55(2003/1/10)  
ロータリエンコーダのサンプリングを 5ms->1ms に変更
- 
- BiosVersion4.56(2003/1/22)  
ロータリエンコーダの片側が中点にある時、反対側のエンコーダが動作しないことを修正  
ロータリエンコーダ変更 (1回転12 24クリックに変更、寿命も3万 10万回転に変更)
- 
- BiosVersion4.60(2003/1/22)  
ソフトは Version4.56 と同じ(Version 値のみ変更)、基板が違うだけ  
基板変更(ALL-LCD-002D)電流値 0mA オフセット調整回路追加のみ
- 
- BiosVersion4.62(2003/2/10)  
Power = Off, Power = On 命令に対応  
20bitL/F の READ ができなかったことを修正(PLD 修正)
- 
- BiosVersion4.63(2003/2/24)  
ON/OFF SW で OFF にする時 VDD 電圧が 0 V まで放電し易いように Tr を追加した。  
ソフトは Version4.62 と全く同じ。
- 
- BiosVersion4.65(2003/7/10)  
READ CHECK 命令のあるドライバー(mot)が READ エラー判定していなかったことを修正  
メモリーカード入っていない時、CHR\_LCD の表示が1行だけ乱れていることを修正  
ドライバー(mot)に連続上書き機能がある場合、対応するように追加
- 
- BiosVersion4.70(2004/1/18)  
画面に登録する LCD ファイル2種類に対応 (画面転送前 LCD ファイル実行追加)  
「LOOP\_BK2」 ... 「LOOP\_BK10」命令追加  
「EXPMODE」命令追加 : 0=SCLK は通常 エッジ 1=SCLK は エッジ  
「CLK\_OUT」命令追加 : E 端子が CLKOUT する  
12.288MHz/(data+1) (12.228MHz)0<=data<=2046(6kHz)
- 
- BiosVersion4.71(2004/2/19)  
「CLK\_OUT」命令範囲拡大 : (12.228MHz)0<=data<=8191(1.5kHz)  
Flash\_PageCopy 高速化、データ登録時アドレスがページ範囲外になると書き込み失敗することを修正
- 
- BiosVersion4.74(2004/8/26)  
画面に登録する LCD ファイル2種類の動作修正  
1画面目 : 前LCD 実行、2画面目 : 後 LCD 実行の場合、  
2画面目の時、前LCD 後 LCD と実行されてしまうことを修正

- 
- BiosVersion4.76(2004/10/16)
    - IO\_MODE = Serial の時、Dout(A0)出力アルゴリズムの修正
    - 通信命令追加「P\_ON」「P\_OFF」「EVUP1」「EVDN1」「EVUP2」「EVDN2」  
「EV\_1」「EV\_2」の計8命令
- 
- BiosVersion4.77(2006/3/16)
    - 1 ドライバ(MOT)の最大動作時間5秒だったが、5秒以上対応できるようにした(\_WD\_reset 関数の修正)
    - 2 ドライバ(MOT)実行前に AUXDATA1 ~7 の Reset
    - 3 ドライバ(MOT)、初期設定(LCD)動作中でも電流値表示するように修正
    - 4 Read Error 表示がすぐ消えてしまうことを修正
    - 5 PLD プログラム修正
      - I/F=20bit 時、WR=Hi なのに DataBus 出力していたのを出力なしに修正
      - DataBus のタイミング修正
        - WR 立下り・・・DataBus 同時から 20ns 早くに修正
        - WR 立上り・・・DataBus40ns 遅いから 20ns 遅くに修正
- 
- BiosVersion4.78(2006/3/31)
    - 1 SRAM 高速化(6state->3srate)これで MOT プログラムが速くなる
    - 2 DMA 転送の高速化(DMA なしから DMA ありに変更)
- 
- BiosVersion4.79(2006/4/28)
    - 1 通信命令追加「STAT」  
PAT=FFh,EV1=FFFFh,EV2=FFFFh,VDD=FFFh,Idd=FFFFh,VEE=FFFh,Iee=FFFFh

#### E タイプチェッカーの BiosVersion

- BiosVersion 5.50F03(2008/2/14)
    - 1 . ハードウェアDタイプから大幅に変更(但しソフトは上位互換)
    - 2 . PC-CARD メモリ(2Mb, 8Mb, 16Mb) 3種類に対応
    - 3 . 電圧調整機能(VDD: 1.80V- 5.00V 最小単位 0.01V、電流 max800mA)  
(VEE: 1.50V-29.00V 最小単位 0.01V、電流 max800mA)  
(VBB: 1.50V-29.00V 最小単位 0.01V、電流 max800mA)
    - 4 . 出力コネクタを 3 4 pin-> 4 0 pin に変更(PinNo.1-3 4 までは日と完全互換)
    - 5 . USB 通信機能
    - 6 . I/F は I2C、DigitalRGB 出力を追加
- 
- BiosVersion 5.51F03(2008/4/30)
    - 1 . MOT から SW 制御できるように外部関数「\_ExtSWset」命令追加
    - 2 . RS232C 通信で XOUT ,X\_IN 命令追加(ソフトデバッグ用)
    - 3 . USB 通信で XOUT ,X\_IN ,STAT 命令追加(ソフトデバッグ用)
    - 4 . STNdata 出力の修正
    - 5 . I2C 時 0:Low /1:Hz に修正
- 
- BiosVersion 5.52F04(2008/5/07)
    - 1 . MOT 動作中 LCD 表示止まらないように修正  
LCD 表示 Timer 割り込みに入れたので 200ms->300ms に変更
    - 2 . USB.....bmAttributes 値 e0h->40h 変更(自己電源)  
USB.....NETCHIP\_INITIALIZE()の USBCTL0 変数 IO\_WAKEUP\_ENABLE なし
    - 3 . ReadError 時、電流 Low リミットで BZoff しないように修正
    - 4 . Iddif リミット初期値 30mA->50mA に変更

- a . DigitaRGB 表示が虫食いになる (SDRAM Write 失敗するので修正)
  - b . HDP 変更すると水平方向ずれるので修正(奇数 dot ではまだ駄目、4の倍数なら OK)
  - c . nLine 機能追加
  - d . CLK2XOUT=8000h:49.152MHz,8001h:16.384MHz,8002-ffffh(8.192MHz/(n-1))
  - e . DRV 出力用 DigitalRGB 波形追加
- 

• BiosVersion 5.53F05(2008/5/22)

- 1 . 通信が遅くなっているので修正 (LCD 表示の影響、割り込みモード 2 と最適化で対応)
- 2 . USB 不安定になることがあるため内部電源 DC5.3->5.1V に変更
- 3 . 電流リミット Error 後、すぐ再出力すると一瞬電流リミット値が表示されることを修正
  - a . DataBUS に Pulldown 入れて(FPGA 内部)ノイズを少し除去