

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願い申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日
ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無いことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

使用方法説明書 (78K0/Lx2)

注意 このソフトウェアは、あくまでも参考用のソフトウェアであり、当社がこの動作を保証するものではありません。このソフトウェアを使用する場合、お客様のセット上で十分な評価の上ご使用いただきますよう、お願いいたします。

改版履歴

版数	作成年月	記事
第 1 版	2006 年 2 月	第 1 版作成

目 次

1. 概 要	4
2. 使用リソース	4
3. ソフトウェア構成	4
4. データのフォーマット	5
4.1 コントロール・レジスタのフォーマット	5
4.2 セグメント・データのフォーマット	8
5. 定義ファイルの説明 (LcdDrv.hファイル内)	9
5.1 EXTRN関数定義	9
5.2 コントロール・レジスタ設定時に使用する定数定義	9
5.3 その他の定数定義	12
6. インタフェース関数の説明 (LcdDrv.cファイル内)	13
6.1 インタフェース関数一覧	13
6.2 インタフェース関数の説明	14
7. I ² Cクロック選択の説明 (I2C.hファイル内)	25
8. 内部関数の説明 (I2C.cファイル内)	26
8.1 内部関数一覧	26
8.2 関数の説明	26
9. 関数構成図	30
10. 内部昇圧方式選択時のインタフェース関数の呼び出し手順	31
10.1 ドライバ初期化～表示オン処理の呼び出し手順	31
10.2 表示オフ/表示オン処理の呼び出し手順	32

1. 概 要

このドキュメントは、NEC エレクトロニクス社製マイコン 78K0/Lx2 シリーズ上で動作するハードウェア内蔵 I²C を使用した LCD コントローラ/ドライバ・ソフトの C ソースでの仕様説明とその開発環境の仕様について示したものです。

備考 このサンプル・プログラムのハードウェアは、 μ PD78F0397 を対象としています。

2. 使用リソース

このソフトウェアでは、 μ PD78F0397 の次のリソースを使用します。

リソース	内容		備考
RAM	I ² C バス処理関連	0 バイト	
	LCD ドライバ処理関連	0 バイト	呼び出し元で計 24 バイトの領域が必要となります。
	ディバグ用メイン処理関連	104 バイト	仕様により増減します。
ROM	I ² C バス処理関連	178 バイト	
	LCD ドライバ処理関連	1524 バイト	コンパイル条件により多少増減します。
	ディバグ用メイン処理関連	約 3000 バイト	仕様により増減します。
I ² C バス関連ハードウェア	I ² C バス通信		
I/O ポート	ポート 0	未使用	
	ポート 1	未使用	
	ポート 2	未使用	
	ポート 3	未使用	
	ポート 7	未使用	
	ポート 12	未使用	
割り込み	ディバグ用の表示更新用基準タイマ割り込み (INTWT) にて使用。		

3. ソフトウェア構成

このソフトウェアは、次のファイルで構成されています。

ファイル名	機能	種別
LcdDrv.h	関数および定数の定義ファイル	ヘッダ・ファイル (ユーザ向け定数および関数定義)
LcdDrv.c	LCD コントローラ/ドライバ処理	ソース・ファイル (ユーザ向けインタフェース処理群)
I2C.h	I ² C 制御クロック設定レジスタ設定ファイル	ヘッダ・ファイル (ユーザ向け定数定義)
I2C.c	I ² C 制御処理	ソース・ファイル (内部処理群)

上記の 4 つのファイルはすべてコンパイルの対象としてください。ヘッダ・ファイルはソースと同じディレクトリ、またはサーチパスの通ったディレクトリに置いてください。

4. データのフォーマット

このソフトウェアはコントロール・レジスタ部およびセグメント・データ部の 2 つの区別されたデータのフォーマットにより、LCD コントローラ/ドライバを制御します。

4.1 コントロール・レジスタのフォーマット

コントロール・レジスタのフォーマットを次に示します。後述する LcdDrv.h ファイルの定数定義を使用します。

コントロール・レジスタ・アドレス

ビット

7	6	5	4	3	2	1	0
---	---	---	---	---	---	---	---

0x00	0	0	0	0	0	0	0	LCD 基準電圧生成回路																																											
LCD モード設定レジスタ LCD 基準電圧生成回路の選択																																																			
<table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 20%;">設定値</th> <th>設定内容</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>外部抵抗分割方式</td> </tr> <tr> <td>01</td> <td>内部抵抗分割方式</td> </tr> <tr> <td>10</td> <td>内部昇圧方式</td> </tr> </tbody> </table>									設定値	設定内容	00	外部抵抗分割方式	01	内部抵抗分割方式	10	内部昇圧方式																																			
設定値	設定内容																																																		
00	外部抵抗分割方式																																																		
01	内部抵抗分割方式																																																		
10	内部昇圧方式																																																		
0x01	LCD 表示の許可/禁止	セグメント/コモ端子出力の制御	昇圧回路の動作許可/禁止	0	0	LCD コントローラ/ドライバの表示モード																																													
LCD 表示モード・レジスタ LCD 表示の許可/禁止																																																			
<table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 20%;">設定値</th> <th>設定内容</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>表示オフ (セグメント出力はすべて非選択信号出力)</td> </tr> <tr> <td>1</td> <td>表示オン</td> </tr> </tbody> </table>									設定値	設定内容	0	表示オフ (セグメント出力はすべて非選択信号出力)	1	表示オン																																					
設定値	設定内容																																																		
0	表示オフ (セグメント出力はすべて非選択信号出力)																																																		
1	表示オン																																																		
セグメント/コモ端子出力の制御																																																			
<table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 20%;">設定値</th> <th>設定内容</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>セグメント/コモ端子にグラウンド・レベルを出力</td> </tr> <tr> <td>1</td> <td>セグメント端子に非選択レベル, コモ端子に LCD 波形を出力</td> </tr> </tbody> </table>									設定値	設定内容	0	セグメント/コモ端子にグラウンド・レベルを出力	1	セグメント端子に非選択レベル, コモ端子に LCD 波形を出力																																					
設定値	設定内容																																																		
0	セグメント/コモ端子にグラウンド・レベルを出力																																																		
1	セグメント端子に非選択レベル, コモ端子に LCD 波形を出力																																																		
昇圧回路の動作許可/禁止																																																			
<table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 20%;">設定値</th> <th>設定内容</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>昇圧回路の動作停止</td> </tr> <tr> <td>1</td> <td>昇圧回路の動作許可</td> </tr> </tbody> </table>									設定値	設定内容	0	昇圧回路の動作停止	1	昇圧回路の動作許可																																					
設定値	設定内容																																																		
0	昇圧回路の動作停止																																																		
1	昇圧回路の動作許可																																																		
セグメント/コモ端子出力の制御																																																			
<table border="1" style="width: 100%;"> <thead> <tr> <th rowspan="3" style="width: 20%;">設定値</th> <th colspan="4">設定内容</th> </tr> <tr> <th colspan="2">抵抗分割方式</th> <th colspan="2">昇圧方式</th> </tr> <tr> <th>時分割数</th> <th>バイアス法</th> <th>時分割数</th> <th>バイアス法</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>4</td> <td>1/3</td> <td>4</td> <td>1/3</td> </tr> <tr> <td>001</td> <td>3</td> <td>1/3</td> <td>3</td> <td>1/3</td> </tr> <tr> <td>010</td> <td>2</td> <td>1/2</td> <td>4</td> <td>1/3</td> </tr> <tr> <td>011</td> <td>3</td> <td>1/2</td> <td>3</td> <td>1/3</td> </tr> <tr> <td>100</td> <td colspan="2">スタティック</td> <td colspan="2">設定禁止</td> </tr> <tr> <td>上記以外</td> <td colspan="4">設定禁止</td> </tr> </tbody> </table>									設定値	設定内容				抵抗分割方式		昇圧方式		時分割数	バイアス法	時分割数	バイアス法	000	4	1/3	4	1/3	001	3	1/3	3	1/3	010	2	1/2	4	1/3	011	3	1/2	3	1/3	100	スタティック		設定禁止		上記以外	設定禁止			
設定値	設定内容																																																		
	抵抗分割方式		昇圧方式																																																
	時分割数	バイアス法	時分割数	バイアス法																																															
000	4	1/3	4	1/3																																															
001	3	1/3	3	1/3																																															
010	2	1/2	4	1/3																																															
011	3	1/2	3	1/3																																															
100	スタティック		設定禁止																																																
上記以外	設定禁止																																																		

コントローラ・レジスタ・アドレス

ビット

7	6	5	4	3	2	1	0
---	---	---	---	---	---	---	---

0x02	0	0	0	0	LCD ソース・クロック	LCD クロック	
LCD クロック制御レジスタ							
LCD ソース・クロック (f _{LCD}) の選択							
設定値	設定内容						
00	f _{PCL} (ソース・クロック出力生成回路で生成されるクロック)						
10	f _{PCL} /2						
11	f _{PCL} /2 ²						
備考 f _{PCL} : ソース・クロック出力選択レジスタ (CKS) で設定します。							
LCD クロックの選択							
設定値	設定内容						
00	f _{LCD} /2 ⁶						
01	f _{LCD} /2 ⁷						
10	f _{LCD} /2 ⁸						
11	f _{LCD} /2 ⁹						
0x03	コントラスト調整	0	0	0	0	0	基準電圧
LCD 昇圧制御レジスタ							
コントラスト調整 (TYP. 値)							
設定値	設定内容						
	基準電圧 = 1.5 V			基準電圧 = 1.0 V			
	V _{LC0}	V _{LC1}	V _{LC2}	V _{LC0}	V _{LC1}	V _{LC2}	
10	4.89 V	3.27 V	1.633 V	3.29 V	2.27 V	1.133 V	
11	4.71 V	3.13 V	1.567 V	3.21 V	2.13 V	1.067 V	
00	4.50 V	3.00 V	1.500 V	3.00 V	2.00 V	1.000 V	
01	4.29 V	2.87 V	1.433 V	2.79 V	1.87 V	0.933 V	
基準電圧 (V _{LC2}) レベルの選択							
設定値	設定内容						
0	1.5 V (使用 LCD パネルが 4.5 V 仕様)						
1	1.0 V (使用 LCD パネルが 3.0 V 仕様)						

4.2 セグメント・データのフォーマット

セグメント・データのフォーマットを次に示します。

備考 COM : コモン位置, S : セグメント位置

		ビット							
		7	6	5	4	3	2	1	0
セグメント・データ・アドレス	0x00	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0
		S1				S0			
	0x01	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0
		S3				S2			
	0x02	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0
		S5				S4			
	0x03	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0
		S7				S6			
	0x04	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0
		S9				S8			
	0x05	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0
		S11				S10			
	0x06	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0
		S13				S12			
	0x07	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0
		S15				S14			
0x08	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S17				S16				
0x09	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S19				S18				
0x0A	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S21				S20				
0x0B	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S23				S22				
0x0C	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S25				S24				
0x0D	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S27				S26				
0x0E	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S29				S28				
0x0F	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S31				S30				
0x10	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S33				S32				
0x11	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S35				S34				
0x12	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S37				S36				
0x13	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0	
	S39				S38				

5. 定義ファイルの説明 (LcdDrv.h ファイル内)

インタフェース関数定義, コントロール・レジスタ (LCD コントローラ/ドライバ側) およびソース・クロック出力選択レジスタ (CKS) の定数定義となります。コントロール・レジスタの定数定義は, 先述したコントロール・レジスタのフォーマットに対応した定数定義となります。各種定数の設定内容と説明を次に示します。

5.1 EXTRN 関数定義

宣言	処理概要	引数	戻り値
LcdDrvInit	LCD ドライバ初期化处理	なし	あり ^{注2}
LcdDrvOnWait	LCD ドライバ表示オン・ウェイト開始前処理 (内部昇圧方式選択時のみ使用)	なし	あり ^{注2}
LcdDrvOn	LCD ドライバ表示オン処理	なし	あり ^{注2}
LcdDrvOff	LCD ドライバ表示オフ処理	なし	あり ^{注2}
LcdDrvCtrWrite	LCD ドライバ・コントロール・レジスタ・ライト処理	あり ^{注1}	あり ^{注2}
LcdDrvSegWrite	LCD ドライバ・セグメント・データ・ライト処理	あり	あり ^{注2}
LcdDrvSegClr	LCD ドライバ・セグメント・データ・クリア処理	なし	あり ^{注2}
LcdDrvCtrRead	LCD コントロール・レジスタ・リード処理	あり ^{注1}	あり ^{注2}
LcdDrvSegRead	LCD セグメント・データ・リード処理	あり	あり ^{注2}
LcdDrvCtrWrite1Byte	LCD ドライバ・コントロール・レジスタの1バイト・ライト処理	あり	あり ^{注2}
LcdDrvSegWrite1Byte	LCD ドライバ・セグメント・データの1バイト・ライト処理	あり	あり ^{注2}

注1. 後述する LCD コントロール・レジスタのアドレス値および設定値を使用します。

2. 後述する定数定義の値を返します。

5.2 コントロール・レジスタ設定時に使用する定数定義

LcdDrv.h ファイル内で定義しているコントロール・レジスタ設定用の定数は, 次に説明する値となっています。ファイル内に定義していますので, 必要のないものは削除 (コメント・アウト) し, 仕様に対応した値を選択してください。

LCD モード設定レジスタ (アドレス : 0x00)

シンボル	値	内容	説明
CLDR_ADDR_LCDMD	0x00	LCD モード設定レジスタ・アドレス	変更不可

シンボル	値	設定内容	説明
		LCD 基準電圧生成回路の選択	ビット 1 の設定により, 抵抗分割方式(0) / 昇圧方式(1)を選択します。 抵抗分割方式を選択した場合は, ビット 0 の設定により, 外部抵抗分割方式(0) / 内部抵抗分割方式(1)を選択します。 処理スイッチを兼ねているため, ビット 1 の抵抗分割方式 (0) / 昇圧方式 (1) の選択により, LCD ドライバの初期化から表示オンまでを異なる制御で行います。
CLDR_LCDMD	0b00000000	外部抵抗分割方式	
	0b00000001	内部抵抗分割方式	
	0b00000010	内部昇圧方式	

LCD 表示モード設定レジスタ (アドレス : 0x01)

シンボル	値	内容	説明
CLDR_ADDR_LCDM	0x01	LCD モード設定レジスタ・アドレス	変更不可

シンボル	値	設定内容				説明
		LCD コントローラ/ドライバの表示モードの選択				ビット 2-0 の設定により, 時分割数 / バイアス法を選択します。 ビット 7-5 はソフトウェア内で設定します。
		抵抗分割方式選択時		昇圧方式選択時		
		時分割数	バイアス法	時分割数	バイアス法	
CLDR_LCDM	0b11100000	4	1/3	4	1/3	
	0b11100001	3	1/3	3	1/3	
	0b11100010	2	1/2	4	1/3	
	0b11100011	3	1/2	3	1/3	
	0b11100100	スタティック		設定禁止		

LCD クロック制御レジスタ (アドレス: 0x02)

シンボル	値	内容	説明
CLDR_ADDR_LCDC	0x02	LCD クロック制御レジスタ・アドレス	変更不可

シンボル	値	設定内容		説明
		LCD ソース・クロック (f _{LCD})の選択	LCD クロックの選択	ビット 3, 2 の設定により, LCD ソース・クロックを選択します。 ビット 1, 0 の設定により, LCD クロックを選択します。
CLDR_LCDC	0b00000000	f _{PCL} (ソース・クロック出力制御で生成されるクロック)	f _{LCD} /2 ⁶	
	0b00000001	f _{PCL}	f _{LCD} /2 ⁷	
	0b00000010	f _{PCL}	f _{LCD} /2 ⁸	
	0b00000011	f _{PCL}	f _{LCD} /2 ⁹	
	0b00001000	f _{PCL} /2	f _{LCD} /2 ⁶	
	0b00001001	f _{PCL} /2	f _{LCD} /2 ⁷	
	0b00001010	f _{PCL} /2	f _{LCD} /2 ⁸	
	0b00001011	f _{PCL} /2	f _{LCD} /2 ⁹	
	0b00001100	f _{PCL} /2 ²	f _{LCD} /2 ⁶	
	0b00001101	f _{PCL} /2 ²	f _{LCD} /2 ⁷	
	0b00001110	f _{PCL} /2 ²	f _{LCD} /2 ⁸	
	0b00001111	f _{PCL} /2 ²	f _{LCD} /2 ⁹	

LCD 昇圧制御レジスタ (アドレス: 0x03)

シンボル	値	内容	説明
CLDR_ADDR_VLCG0	0x03	LCD 昇圧制御レジスタ・アドレス	変更不可

シンボル	値	設定内容				説明
		基準電圧 (V _{LC2}) レベルの選択			ビット 0 の設定により, 基準電圧レベルを選択します。 基準電圧レベルは, 使用 LCD パネルが 4.5V 仕様のときには 1.5 V に, 3.0 V 仕様のときには 1.0 V になるように選択します。 ビット 7, 6 の設定により, コントラストの調整を選択します。	
		↓	コントラスト調整 (TYP.値)			
			V _{LC0}	V _{LC1}		V _{LC2}
CLDR_VLCG0	0b10000000	1.5 V	4.89 V	3.27 V		1.633 V
	0b11000000	1.5 V	4.71 V	3.13 V		1.567 V
	0b00000000	1.5 V	4.50 V	3.00 V		1.500 V
	0b01000000	1.5 V	4.29 V	2.87 V		1.433 V
	0b10000001	1.0 V	3.29 V	2.27 V		1.133 V
	0b11000001	1.0 V	3.21 V	2.13 V		1.067 V
	0b00000001	1.0 V	3.00 V	2.00 V	1.000 V	
	0b01000001	1.0 V	2.79 V	1.87 V	0.933 V	

ソース・クロック出力選択レジスタ (CKS)

シンボル	値	設定内容			説明
		ソース・クロック (f _{PCL}) の選択			ビット 3-0 の設定により、ソース・クロックを選択します。
		f _{SUB} =	f _{PRS} =	f _{PRS} =	
			32.768 kHz	10 MHz	20 MHz
CLDR_CKS	00000110B	f _{PRS} /2 ⁶	-	156.25 kHz	312.5 kHz
	00000111B	f _{PRS} /2 ⁷		78.125 kHz	156.25 kHz
	00001000B	f _{SUB}	32.768 kHz	-	
	上記以外	設定禁止			ビット 4 (LCDC/D への出力許可(1)/禁止(0)) はソフトウェア内で設定します。

5.3 その他の定数定義

インタフェース関数の戻り値を定義しています。

シンボル	値	内容	備考
CLDR_ERR_NONE	0	エラーなし	値 0~2 は I ² C 制御でのエラー状態、3 は関数の引数に誤りがある場合を示します。
CLDR_ERR_NACK	1	NACK 受信	
CLDR_ERR_BUSY	2	ビジー状態 (通信不可)	
CLDR_ERR_PARA	3	パラメータ異常 (処理の引数に誤りあり)	

6. インタフェース関数の説明 (LcdDrv.c ファイル内)

- 注意 1. インタフェース関数に使用する定数は、LcdDrv.h ファイル内の各種定数定義から選択します。
2. コントロール・レジスタの送受信には 4 バイトの RAM エリア、セグメント・データの送受信には 20 バイトの RAM エリアの確保が必要です。

6.1 インタフェース関数一覧

インタフェース関数の一覧を次に示します。

宣言	処理概要	引数	戻り値	通信時間 ^{注1}
LcdDrvInit	LCD ドライバ初期化处理	なし	あり	1530 μ s
LcdDrvOnWait	LCD ドライバ表示オン・ウェイト開始前処理 (内部昇圧方式選択時のみ使用)	なし	あり	180 μ s
LcdDrvOn	LCD ドライバ表示オン処理	なし	あり	180 μ s
LcdDrvOff	LCD ドライバ表示オフ処理	なし	あり	1440 μ s
LcdDrvCtrWrite	LCD ドライバ・コントロール・レジスタ・ライト処理	あり	あり	90 ~ 180 μ s ^{注2}
LcdDrvSegWrite	LCD ドライバ・セグメント・データ・ライト処理	あり	あり	120 ~ 1260 μ s ^{注3}
LcdDrvSegClr	LCD ドライバ・セグメント・データ・クリア処理	なし	あり	1260 μ s
LcdDrvCtrRead	LCD コントロール・レジスタ・リード処理	あり	あり	120 ~ 210 μ s ^{注2}
LcdDrvSegRead	LCD セグメント・データ・リード処理	あり	あり	150 ~ 1290 μ s ^{注3}
LcdDrvCtrWrite1Byte	LCD ドライバ・コントロール・レジスタの 1 バイト・ライト処理	あり	あり	120 μ s (リード時) 90 μ s (ライト時) ^{注4}
LcdDrvSegWrite1Byte	LCD ドライバ・セグメント・データの 1 バイト・ライト処理	あり	あり	150 μ s (リード時) 120 μ s (ライト時) ^{注4}

注 1. 通信時間は、メイン・システム・クロックを 8 MHz に選択した場合の I²C バス通信のみに要する時間を記載しています。処理自体にかかる時間は、20 μ s 前後となります。

2. 最小で 1 バイトのデータ、最大で 4 バイトのデータ送受信した場合を記載しています。
3. 最小で 1 バイトのデータ、最大で 20 バイトのデータ送受信した場合を記載しています。
4. リード指定もしくはライト指定で、通信時間は異なります。

6.2 インタフェース関数の説明

LcdDrvInit

宣言	unsigned char LcdDrvInit(void)
処理	LCD ドライバ初期化処理
引数	-
戻り値	0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態
内容	<p>LCD チップの初期化に必要な設定を行います。処理内容を次に示します。</p> <ul style="list-style-type: none"> ・I²C バス制御の初期化 ・LCDC/D のリセット状態解除 ・出力クロック設定 ・LCDC/D へのクロック出力許可 ・LCD 基準電圧生成回路の選択設定 (内部昇圧方式もしくは抵抗分割方式に設定) ・LCD 表示データエリアの初期化 (すべて消灯に設定) ・表示モードの選択設定 ・LCD クロック, LCD ソース・クロックの設定 <p>すべての処理を終えた場合は"0"を戻り値とします。 ~ で NACK 受信した場合は"1", スタート・コンディションが送信できない場合は"2"を戻り値とし, ストップ・コンディションを送信します。</p>
備考	<p>リセット後に呼び出します。</p> <p>【処理内部で使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ・"CLDR_CKS" ・"CLDR_LCDMD" ・"CLDR_LCDM" ・"CLDR_LCDC"

LcdDrvOnWait

宣言	unsigned char LcdDrvOnWait(void)
処理	LCD ドライバ表示オン・ウエイト開始前処理 (内部昇圧方式選択時のみ使用)
引数	-
戻り値	0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態
内容	<p>定義ファイル"LcdDrv.h"にて、基準電圧生成回路を内部昇圧方式を選択した場合の、表示オン・ウエイト開始前の処理となります。処理内容を次に示します。</p> <p>【内部昇圧方式選択時】</p> <ul style="list-style-type: none"> ・I²C 動作が停止している場合は動作許可に設定 ・LCDC/D へのクロック出力許可 ・LCD 基準電圧レベル、コントラストの設定 ・昇圧回路の動作許可 <p>以降、500 ms 以上のウエイトを確保し、後述するインタフェース関数"LcdDrvOn()"を呼び出します。</p> <p>すべての処理を終えた場合は"0"を戻り値とします。 ~ で NACK 受信した場合は"1" ,スタート・コンディションが送信できない場合は"2"を戻り値とし、ストップ・コンディションを送信します。</p>
備考	<ul style="list-style-type: none"> ・定義ファイル"LcdDrv.h"にて、基準電圧生成回路を内部昇圧方式に選択した場合は、表示オン処理"LcdDrvOn()"を呼び出す前に必要な処理となります。この関数の呼び出したあと、500 ms 以上のウエイトを確保し、後述する表示オン処理"LcdDrvOn()"を呼び出します。詳細は 10. 内部昇圧方式選択時のインタフェース関数の呼び出し手順を参照してください。 ・内部 / 外部抵抗分割方式を選択した場合は、必要ありません。 ・リセット後は、この関数を呼び出す前に、一度、LCD ドライバ初期化処理"LcdDrvInit"を呼び出してください。 <p>【処理内部で使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ・"CLDR_VLCG0" ・"CLDR_LCDM"

LcdDrvOn

宣言	unsigned char LcdDrvOn(void)
処理	LCD ドライバ表示オン処理
引数	-
戻り値	0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態
内容	<p>表示オンに必要な設定を行います。処理内容を次に示します。</p> <p>【内部昇圧方式の場合】</p> <p style="padding-left: 40px;">セグメント/コモン端子出力の設定 (LCD 波形出力)</p> <p style="padding-left: 40px;">表示オン設定</p> <p>すべての処理を終えた場合は"0"を戻り値とします。 ~ で NACK 受信した場合は"1", スタート・コンディションが送信できない場合は"2"を戻り値とし, ストップ・コンディションを送信します。</p> <p>【抵抗分割方式の場合】</p> <p style="padding-left: 40px;">I²C 動作が停止している場合は動作許可に設定</p> <p style="padding-left: 40px;">LCDC/D へのクロック出力許可</p> <p style="padding-left: 40px;">セグメント/コモン端子出力の設定 (LCD 波形出力)</p> <p style="padding-left: 40px;">表示オン設定</p> <p>すべての処理を終えた場合は"0"を戻り値とします。 ~ で NACK 受信した場合は"1", スタート・コンディションが送信できない場合は"2"を戻り値とし, ストップ・コンディションを送信します。</p>
備考	<p>・定義ファイル"LcdDrv.h"にて, 基準電圧生成回路を内部昇圧方式に選択した場合は, この関数の呼び出し前に, 必ず, 表示オン・ウェイト開始前処理"LcdDrvOnWait()"を呼び出し, 500 ms 以上のウェイトを確保してください。</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;">LcdDrvOnWait(): 表示オン・ウェイト開始前処理</div> <p style="text-align: center;">↓</p> <p style="text-align: center;">500 ms 以上のウェイトを確保したあと, "LcdDrvOn()"を呼び出します。 詳細は 10. 内部昇圧方式選択時のインタフェース関数の呼び出し手順を参照してください。</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;">LcdDrvOn() : 表示オン開始前処理</div> <p>・リセット後は, この関数を呼び出す前に, 一度, LCD ドライバ初期化処理"LcdDrvInit()"を呼び出してください。</p> <p>【処理内部で使用する定義ファイルの定数】</p> <p>"CLDR_LCDM"</p>

LcdDrvOff

宣言	unsigned char LcdDrvOff(void)
処理	LCD ドライバ表示オフ処理
引数	-
戻り値	0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態
内容	<p>表示オン状態から表示オフ状態への制御を行います。処理内容を次に示します。</p> <ul style="list-style-type: none">LCD 表示データ・エリアの初期化 (すべて消灯に設定)表示オフ設定セグメント/コモン端子の出力禁止設定昇圧回路の動作禁止設定 【内部昇圧方式選択時のみ】LCDC/D へのクロック出力禁止I²C 動作停止設定 <p>すべての処理を終えた場合は"0"を戻り値とします。 ~ で NACK 受信した場合は"1", スタート・コンディションが送信できない場合は"2"を戻り値とし, ストップ・コンディションを送信します。</p>
備考	<p>定義ファイル"LcdDrv.h"にて選択した基準電圧生成回路 (内部昇圧方式 / 抵抗分割方式) により, 処理が異なります。</p> <p>【処理内部で使用する定義ファイルの定数】</p> <p>"CLDR_LCDM"</p>

LcdDrvCtrWrite

宣言	unsigned char LcdDrvCtrWrite(unsigned char *src, unsigned char addr, unsigned char size)
処理	LCD ドライバ・コントロール・レジスタ・ライト
引数	<ul style="list-style-type: none"> ・ unsigned char *src : ライトするコントロール・レジスタのデータ格納アドレス ・ unsigned char addr : ライト開始するコントロール・レジスタ・アドレス値 ・ unsigned char size : 送信バイト数 (最大 4 バイト)
戻り値	0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態, 3 = パラメータ・エラー
内容	<p>上記の引数を使用し、I²C 動作を一括して行います。処理内容を次に示します。</p> <ul style="list-style-type: none"> I²C 動作が停止している場合は動作許可に設定 LCDC/D へのクロック出力許可 パラメータ・チェック (引数のチェック) スタート・コンディション送信 スレーブ ID 送信 (処理内で使用) コントロール・レジスタ・アドレス送信 コントロール・レジスタ・データ (送信バイト数分) 送信 ストップ・コンディション送信 <p>すべての処理を終えた場合は"0"を戻り値とします。 で NACK 受信した場合は"1", でスタート・コンディションが送信できない場合は"2"を戻り値とし、ストップ・コンディションを送信します。 で引数に誤りがある場合は"3"を戻り値とし、以降に続く処理は行われません。</p>
備考	<ul style="list-style-type: none"> ・ コントロール・レジスタの送受信には 4 バイトの RAM エリアの確保が必要です。 ・ パラメータのチェックはライト開始するコントロール・レジスタ・アドレス値および送信バイト数が適正かのチェックのみ行います。送信するデータのチェックは行いません。 ・ 2 バイト以上のデータ送信時では、アドレスを自動的にインクリメントし、連続送信を行います。 ・ ライトするコントロール・レジスタ・データは 4.1 コントロール・レジスタのフォーマットを参照してください。 <p>【引数として使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ~ コントロール・レジスタのアドレス ~ ・ "CLDR_ADDR_LCDMD" ・ "CLDR_ADDR_LCDM" ・ "CLDR_ADDR_LCDC" ・ "CLDR_ADDR_VLCG0" ~ コントロール・レジスタ設定値 ~ ・ "CLDR_LCDMD" ・ "CLDR_LCDM" ・ "CLDR_LCDC" ・ "CLDR_VLCG0" <p>【戻り値として使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ・ "CLDR_ERR_NONE" ・ "CLDR_ERR_NACK" ・ "CLDR_ERR_BUSY" ・ "CLDR_ERR_PARA"

LcdDrvSegWrite

宣言	unsigned char LcdDrvSegWrite(unsigned char *src, unsigned char addr, unsigned char size)
処理	LCD ドライバ・セグメント・データ・ライト
引数	<ul style="list-style-type: none"> ・ unsigned char *src : ライトするセグメント・データの格納アドレス ・ unsigned char addr : ライト開始するセグメント・データ・アドレス値 ・ unsigned char size : 送信バイト数 (最大 20 バイト)
戻り値	0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態, 3 = パラメータ・エラー
内容	<p>上記の引数を使用し, I²C 動作を一括して行います。処理内容を次に示します。</p> <ul style="list-style-type: none"> I²C 動作が停止している場合は動作許可に設定 LCDC/D へのクロック出力許可 パラメータ・チェック (引数のチェック) スタート・コンディション送信 スレーブ ID 送信 (処理内で使用) セグメント・データ・アドレス送信 セグメント・データ (送信バイト数分) 送信 ストップ・コンディション送信 <p>すべての処理を終えた場合は"0"を戻り値とします。 で NACK 受信した場合は"1", でスタート・コンディションが送信できない場合は"2"を戻り値とし, ストップ・コンディションを送信します。 で引数に誤りがある場合は"3"を戻り値とし, 以降に続く処理は行われません。</p>
備考	<ul style="list-style-type: none"> ・セグメント・データの送受信には, 20 バイトの RAM エリアの確保が必要です。 ・パラメータのチェックは, ライト開始するセグメント・データ・アドレス値および送信バイト数が適正かのチェックのみ行います。送信するデータのチェックは行いません。 ・2 バイト以上のデータ送信時では, アドレスを自動的にインクリメントし, 連続送信を行います。 ・ライトするセグメント・データについては 4.2 セグメント・データのフォーマットを参照してください。 <p>【戻り値として使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ・ "CLDR_ERR_NONE" ・ "CLDR_ERR_NACK" ・ "CLDR_ERR_BUSY" ・ "CLDR_ERR_PARA"

LcdDrvSegClr

宣言	unsigned char LcdDrvSegClr(void)
処理	LCD ドライバ・セグメント・データ・クリア
引数	-
戻り値	0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態
内容	<p>上記の引数を使用し, I²C 動作を一括して行います。処理内容を次に示します。</p> <ul style="list-style-type: none"> I²C 動作が停止している場合は動作許可に設定 LCDC/D へのクロック出力許可 パラメータ・チェック (引数のチェック) スタート・コンディション送信 スレーブ ID 送信 (処理内で使用) セグメント・データ・アドレス送信 セグメント・データ (20 バイト分) 送信 ストップ・コンディション送信 <p>すべての処理を終えた場合は"0"を戻り値とします。 ~ で NACK 受信した場合は"1", でスタート, コンディションが送信できない場合は"2"を戻り値とし, ストップ・コンディションを送信します。</p>
備考	<p>セグメント・データ全領域にすべて消灯となるように, 設定データを送信します。</p> <p>【戻り値として使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ・ "CLDR_ERR_NONE" ・ "CLDR_ERR_NACK" ・ "CLDR_ERR_BUSY"

LcdDrvCtrRead

宣言	unsigned char LcdDrvCtrRead(unsigned char *src, unsigned char addr, unsigned char size)
処理	LCD ドライバ・コントロール・レジスタ・リード
引数	<ul style="list-style-type: none"> ・ unsigned char *src : リードするコントロール・レジスタのデータ格納アドレス ・ unsigned char addr : リード開始するコントロール・レジスタ・アドレス値 ・ unsigned char size : 受信バイト数 (最大 4 バイト)
戻り値	0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態, 3 = パラメータ・エラー
内容	<p>上記の引数を使用し, I²C 動作を一括して行います。処理内容を次に示します。</p> <ul style="list-style-type: none"> I²C 動作が停止している場合は動作許可に設定 LCDC/D クロック出力許可 パラメータ・チェック (引数のチェック) スタート・コンディション送信 スレーブ ID 送信 (処理内で使用) コントロール・レジスタ・アドレス送信 リスタート・コンディション送信 スレーブ ID 送信 (リード要求指定) (処理内で使用) コントロール・レジスタ・データ (受信バイト数分) 受信 ストップ・コンディション送信 <p>すべての処理を終えた場合は"0"を戻り値とします。 ~ , ~ で NACK 受信した場合は"1", ~ , ~ でスタート・コンディションが送信できない場合は"2"を戻り値とし, ストップ・コンディションを送信します。 で引数に誤りがある場合は"3"を戻り値とし,以降に続く処理は行われません。</p>
備考	<ul style="list-style-type: none"> ・ コントロール・レジスタの送受信には 4 バイトの RAM エリアの確保が必要です。 ・ パラメータのチェックは, リード開始するコントロール・レジスタ・アドレス値および送信バイト数が適正かのチェックのみ行います。送信するデータのチェックは行いません。 ・ 2 バイト以上のデータ受信時では, アドレスを自動的にインクリメントし, 連続受信を行います。 ・ リードするコントロール・レジスタ・データについては, 4.1 コントロール・レジスタのフォーマットを参照してください。 <p>【引数として使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ~ コントロール・レジスタのアドレス ~ ・ "CLDR_ADDR_LCDMD" ・ "CLDR_ADDR_LCDM" ・ "CLDR_ADDR_LCDC" ・ "CLDR_ADDR_VLCG0" <p>【戻り値として使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ・ "CLDR_ERR_NONE" ・ "CLDR_ERR_NACK" ・ "CLDR_ERR_BUSY" ・ "CLDR_ERR_PARA"

LcdDrvSegRead

宣言	unsigned char LcdDrvSegRead(unsigned char *src, unsigned char addr, unsigned char size)
処理	LCD ドライバ・セグメント・データ・リード
引数	<ul style="list-style-type: none"> ・ unsigned char *src : リードするセグメント・データの格納アドレス ・ unsigned char addr : リード開始するセグメント・データ・アドレス値 ・ unsigned char size : 受信バイト数 (最大 20 バイト)
戻り値	0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態, 3 = パラメータ・エラー
内容	<p>上記の引数を使用し、I²C 動作を一括して行います。処理内容を次に示します。</p> <ul style="list-style-type: none"> I²C 動作が停止している場合は動作許可に設定 LCDC/D へのクロック出力許可 パラメータ・チェック (引数のチェック) スタート・コンディション送信 スレーブ ID 送信 (処理内で使用) セグメント・データ・アドレス送信 リスタート・コンディション送信 スレーブ ID 送信 (リード要求指定) (処理内で使用) セグメント・データ (受信バイト数分) 受信 ストップ・コンディション送信 <p>すべての処理を終えた場合は"0"を戻り値とします。 ~ , ~ で NACK 受信した場合は"1", ~ , ~ でスタート・コンディションが送信できない場合は"2"を戻り値とし、ストップ・コンディションを送信します。 ~ で引数に誤りがある場合は"3"を戻り値とし、以降に続く処理は行われません。</p>
備考	<ul style="list-style-type: none"> ・セグメント・データの送受信には 20 バイトの RAM エリアの確保が必要です。 ・パラメータのチェックは、リード開始するセグメント・データ・アドレス値および送信バイト数が適正かのチェックのみ行います。送信するデータのチェックは行いません。 ・2 バイト以上のデータ受信時では、アドレスを自動的にインクリメントし、連続受信を行います。 ・リードするセグメント・データについては A.2 セグメント・データのフォーマットを参照してください。 <p>【戻り値として使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ・ "CLDR_ERR_NONE" ・ "CLDR_ERR_NACK" ・ "CLDR_ERR_BUSY" ・ "CLDR_ERR_PARA"

LcdDrvCtrWrite1Byte

宣言	unsigned char LcdDrvCtrWrite1Byte(unsigned char addr, unsigned char data)										
処理	LCD ドライバ・コントロール・レジスタの1バイト・ライト										
引数	<ul style="list-style-type: none"> ・ unsigned char addr : ライトするコントロール・レジスタ・アドレス値 ・ unsigned char data : ライトするコントロール・レジスタ・データ 										
戻り値	<ul style="list-style-type: none"> ・ 0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態 										
内容	<p>上記の引数を使用し, I²C 動作を一括して行います。処理内容を次に示します。</p> <p>【ライト要求時】</p> <ul style="list-style-type: none"> I²C 動作が停止している場合は動作許可に設定 LCDC/D へのクロック出力許可 スタート・コンディション送信 スレーブ ID 送信 (処理内で使用) コントロール・レジスタ・アドレス送信 コントロール・レジスタ・データ送信 ストップ・コンディション送信 <p>【リード要求時】</p> <ul style="list-style-type: none"> I²C 動作が停止している場合は動作許可に設定 LCDC/D へのクロック出力許可 スタート・コンディション送信 スレーブ ID 送信 (処理内で使用) コントロール・レジスタ・アドレス送信 リスタート・コンディション送信 スレーブ ID 送信 (リード要求指定) (処理内で使用) コントロール・レジスタ・データ受信 ストップ・コンディション送信 <p>すべての処理を終えた場合は"0"を戻り値とします。 ~ で NACK 受信した場合は"1", でスタート・コンディションが送信できない場合は"2"を戻り値とし, ストップ・コンディションを送信します。</p>										
備考	<p>ライトするコントロール・レジスタ・データについては, 4.1 コントロール・レジスタのフォーマットを参照してください。</p> <p>【引数として使用する定義ファイルの定数】</p> <table border="0"> <tr> <td>~ コントロール・レジスタのアドレス ~</td> <td>~ コントロール・レジスタ設定値 ~</td> </tr> <tr> <td>・ "CLDR_ADDR_LCDMD"</td> <td>・ "CLDR_LCDMD"</td> </tr> <tr> <td>・ "CLDR_ADDR_LCDM"</td> <td>・ "CLDR_LCDM"</td> </tr> <tr> <td>・ "CLDR_ADDR_LCDC"</td> <td>・ "CLDR_LCDC"</td> </tr> <tr> <td>・ "CLDR_ADDR_VLCG0"</td> <td>・ "CLDR_VLCG0"</td> </tr> </table> <p>【戻り値として使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ・ "CLDR_ERR_NONE" ・ "CLDR_ERR_NACK" ・ "CLDR_ERR_BUSY" 	~ コントロール・レジスタのアドレス ~	~ コントロール・レジスタ設定値 ~	・ "CLDR_ADDR_LCDMD"	・ "CLDR_LCDMD"	・ "CLDR_ADDR_LCDM"	・ "CLDR_LCDM"	・ "CLDR_ADDR_LCDC"	・ "CLDR_LCDC"	・ "CLDR_ADDR_VLCG0"	・ "CLDR_VLCG0"
~ コントロール・レジスタのアドレス ~	~ コントロール・レジスタ設定値 ~										
・ "CLDR_ADDR_LCDMD"	・ "CLDR_LCDMD"										
・ "CLDR_ADDR_LCDM"	・ "CLDR_LCDM"										
・ "CLDR_ADDR_LCDC"	・ "CLDR_LCDC"										
・ "CLDR_ADDR_VLCG0"	・ "CLDR_VLCG0"										

LcdDrvSegWrite1Byte

宣言	unsigned char LcdDrvSegWrite1Byte(unsigned char addr, unsigned char data)
処理	LCD ドライバ・セグメント・データの 1 バイト・ライト
引数	<ul style="list-style-type: none"> ・ unsigned char addr : ライトするセグメント・データ・アドレス値 ・ unsigned char data : ライトするセグメント・データ
戻り値	0 = 設定成功, 1 = NACK 受信, 2 = ビジィ状態
内容	<p>上記の引数を使用し, I²C 動作を一括して行います。処理内容を次に示します。</p> <p>【ライト要求時】</p> <ul style="list-style-type: none"> I²C 動作が停止している場合は動作許可に設定 LCDC/D へのクロック出力許可 スタート・コンディション送信 スレーブ ID 送信 (処理内で使用) セグメント・データ・アドレス送信 セグメント・データ送信 ストップ・コンディション送信 <p>【リード要求時】</p> <ul style="list-style-type: none"> I²C 動作が停止している場合は動作許可に設定 LCDC/D へのクロック出力許可 スタート・コンディション送信 スレーブ ID 送信 (処理内で使用) セグメント・データ・アドレス送信 リスタート・コンディション送信 スレーブ ID 送信 (リード要求指定) (処理内で使用) セグメント・データ受信 ストップ・コンディション送信 <p>すべての処理を終えた場合は"0"を戻り値とします。 ~ で NACK 受信した場合は"1", でスタート・コンディションが送信できない場合は"2"を戻り値とし, ストップ・コンディションを送信します。</p>
備考	<p>ライトするセグメント・データについては, 4.2 セグメント・データのフォーマットを参照してください。</p> <p>【戻り値として使用する定義ファイルの定数】</p> <ul style="list-style-type: none"> ・ "CLDR_ERR_NONE" ・ "CLDR_ERR_NACK" ・ "CLDR_ERR_BUSY"

7. I²C クロック選択の説明 (I2C.h ファイル内)

I²C バス制御時の転送クロックは I2C.h ファイル内のレジスタ用定数定義を使用しています。
定数の設定内容と説明を次に示します。

シンボル	値	シンボル	値	設定内容									
				動作モード	デジタル・フィルタ	選択クロック (fw)	転送クロック (fPRS/m)	設定可能な選択クロック (fw) の範囲					
CI2C_IICX	0b00000000	CI2C_IICCL	0b00000000	標準モード	オフ	fPRS/2	fPRS/88	2.00 ~ 4.19MHz					
	0b00000000		0b00000001			fPRS/2	fPRS/172	4.19 ~ 8.38MHz					
	0b00000000		0b00000010			fPRS/4	fPRS/344						
	0b00000000	CI2C_IICCL	0b00001000	高速モード	オフ	fPRS/2	fPRS/48	4.19 ~ 8.38MHz					
	0b00000000		0b00001010			fPRS/4	fPRS/96						
	0b00000001		0b00001000			fPRS/2	fPRS/24	4.00 ~ 4.19MHz					
	0b00000001		0b00001010			fPRS/4	fPRS/48						
	0b00000000		CI2C_IICCL	0b00001100	高速モード	オン	fPRS/2	fPRS/48	4.19 ~ 8.38MHz				
	0b00000000			0b00001110			fPRS/4	fPRS/96					
	0b00000001			0b00001100			fPRS/2	fPRS/24	4.00 ~ 4.19MHz				
	0b00000001			0b00001110			fPRS/4	fPRS/48					
	上記以外						設定禁止						
	【説明】												
	上記、2つの定数を用いてレジスタの設定を行います。CI2C_IICX は IIC 機能拡張レジスタ 0 (IICX0) , CI2C_IICCL は IIC クロック選択レジスタ 0 (IICL0) の設定値となります。												
CI2C_IICCL のビット 3 の設定により、標準モード (0) / 高速モード (1) の動作モードの選択を行います。													
CI2C_IICCL のビット 2 の設定により、デジタル・フィルタのオフ (0) / オン (1) の選択を行います。デジタル・フィルタは、高速モードが選択されているときのみ使用できます。													
CI2C_IICCL のビット 3, 1, 0 の3つのビットは、CI2C_IICX のビット 0 と組み合わせて転送クロックの設定を行います。													
備考 fPRS : 周辺ハードウェア・クロック発振周波数													

8. 内部関数の説明 (I2C.c ファイル内)

I²C バス制御の処理群となります。LcdDrv.c ファイル内の処理からの呼び出しとなります。

8.1 内部関数一覧

内部関数の一覧を次に示します。

関数名	処理概要	引数	戻り値
I2cWait	I ² C バス送受信完了待ち処理	なし	なし
I2cInit	I ² C バス初期化処理	なし	なし
I2cStopCondition	I ² C バス・ストップ・コンディション送信処理	なし	なし
I2cStartCondition	I ² C バス・スタート・コンディション送信処理	なし	あり
I2cReStartCondition	I ² C バス・リスタート・コンディション送信処理	なし	なし
I2cPutAddress	I ² C バス 8 ビット・データ送信 (アドレス送信) 処理	あり	あり
I2cPutData	I ² C バス 8 ビット・データ送信 (データ送信) 処理	あり	あり
I2cGetDataAckSet	I ² C バス 8 ビット・データ受信&ACK 送信処理	なし	あり
I2cGetDataNackSet	I ² C バス 8 ビット・データ受信&NACK 送信処理	なし	あり
I2cDisable	I ² C バス動作停止処理	なし	なし
I2cEnable	I ² C バス動作許可処理	なし	なし

8.2 関数の説明

I2cWait

宣言	static void I2cWait(void)
処理	I ² C バス送受信完了待ち処理
引数	-
戻り値	-
内容	データ送信もしくは受信後の完了待ちを行う。
備考	-

I2cIinit

宣言	void I2cIinit(void)
処理	I ² C バス初期化処理
引数	-
戻り値	-
内容	次の内容で処理を行います。 I ² C 動作停止 転送クロックの設定 通信予約禁止設定 ストップ・コンディション送信処理"I2cStopCondition()"呼び出し (I ² C 動作許可)
備考	-

I2cStopCondition

宣言	void I2cStopCondition(void)
処理	I ² C バス・ストップ・コンディション送信処理
引数	-
戻り値	-
内容	ストップ・コンディションの送信を行う。
備考	-

I2cStartCondition

宣言	void I2cStartCondition(void)
処理	I ² C バス・スタート・コンディション送信処理
引数	-
戻り値	0 = 送信成功 ("RESULT_OK") , 1 = ビジィ状態 ("RESULT_NG")
内容	下記内容で処理を行います。 ビジィ状態のチェック スタート・コンディションの送信 のチェックでビジィ状態であれば、スタート・コンディションの送信を行わず、ビジィ状態通知を行います。
備考	-

I2cReStartCondition

宣言	Void I2cReStartCondition(void)
処理	I ² C バス・リスタート・コンディション送信処理
引数	-
戻り値	-
内容	下記内容で処理を行います。 ・スタート・コンディションの送信
備考	-

I2cPutAddress

宣言	Unsigned char I2cPutAddress(unsigned char data)
処理	I ² C バス 8 ビット・データ送信 (アドレス送信) 処理
引数	unsigned char data : 8 ビット・アドレス値
戻り値	0 = NACK 受信 ("RESULT_NACK") , 1 = ACK 受信 ("RESULT_ACK") , 2 = スタート・コンディション検出なし ("RESULT_BUSY")
内容	次の内容で処理を行います。 スタート・コンディション検出のチェック I ² C バス 8 ビット・データ送信処理"I2cPutData"の呼び出し のチェックでビジー状態であれば、データの送信を行わず、ビジー状態通知を行います。
備考	-

I2cPutData

宣言	unsigned char I2cPutData(unsigned char data)
処理	I ² C バス 8 ビット・データ送信 (データ送信) 処理
引数	unsigned char data : 8 ビット・データ値
戻り値	0 = NACK 受信 ("RESULT_NACK") , 1=ACK 受信 ("RESULT_ACK")
内容	次の内容で処理を行います。 データ送信 送信完了ウエイト "I2cWait() ACK 検出 データ送信後、ACK の状態を返します。
備考	-

I2cGetDataAckSet

宣言	unsigned char I2cGetDataAckSet(unsigned char *src)
処理	I ² C バス 8 ビット・データ受信&ACK 送信処理
引数	unsigned char *src : 受信データ設定エリアのアドレス
戻り値	0 = 受信データあり ("RESULT_OK") , 1 = 受信データなし ("RESULT_NG")
内容	下記内容で処理を行います。 受信データあり/なしチェック ウエイト解除&ACK 送信設定 受信完了ウエイト"I2cWait" 受信データ設定 で受信データがない場合は、以降の処理を行わず、受信データなしを返します。また、 で受信データがある場合は、最後まで処理を行い、受信データを返します。
備考	最終ではない、途中のデータを受信する場合に使用します。

I2cGetDataNackSet

宣言	unsigned char I2cGetDataNackSet(unsigned char *src)
処理	I ² C バス 8 ビット・データ受信&NACK 送信処理
引数	unsigned char *src : 受信データ設定エリアのアドレス
戻り値	0 = 受信データあり ("RESULT_OK") , 1 = 受信データなし ("RESULT_NG")
内容	<p>次の内容で処理を行います。</p> <ul style="list-style-type: none"> 受信データあり/なしチェック ウェイト解除&NACK 送信設定 受信完了ウェイト"I2cWait" 受信データ設定 <p>で受信データがない場合は、以降の処理を行わず、受信データなしを返します。また、 で受信データがある場合は、最後まで処理を行い、受信データを返します。</p>
備考	最終のデータを受信する場合に使用します。

I2cDisable

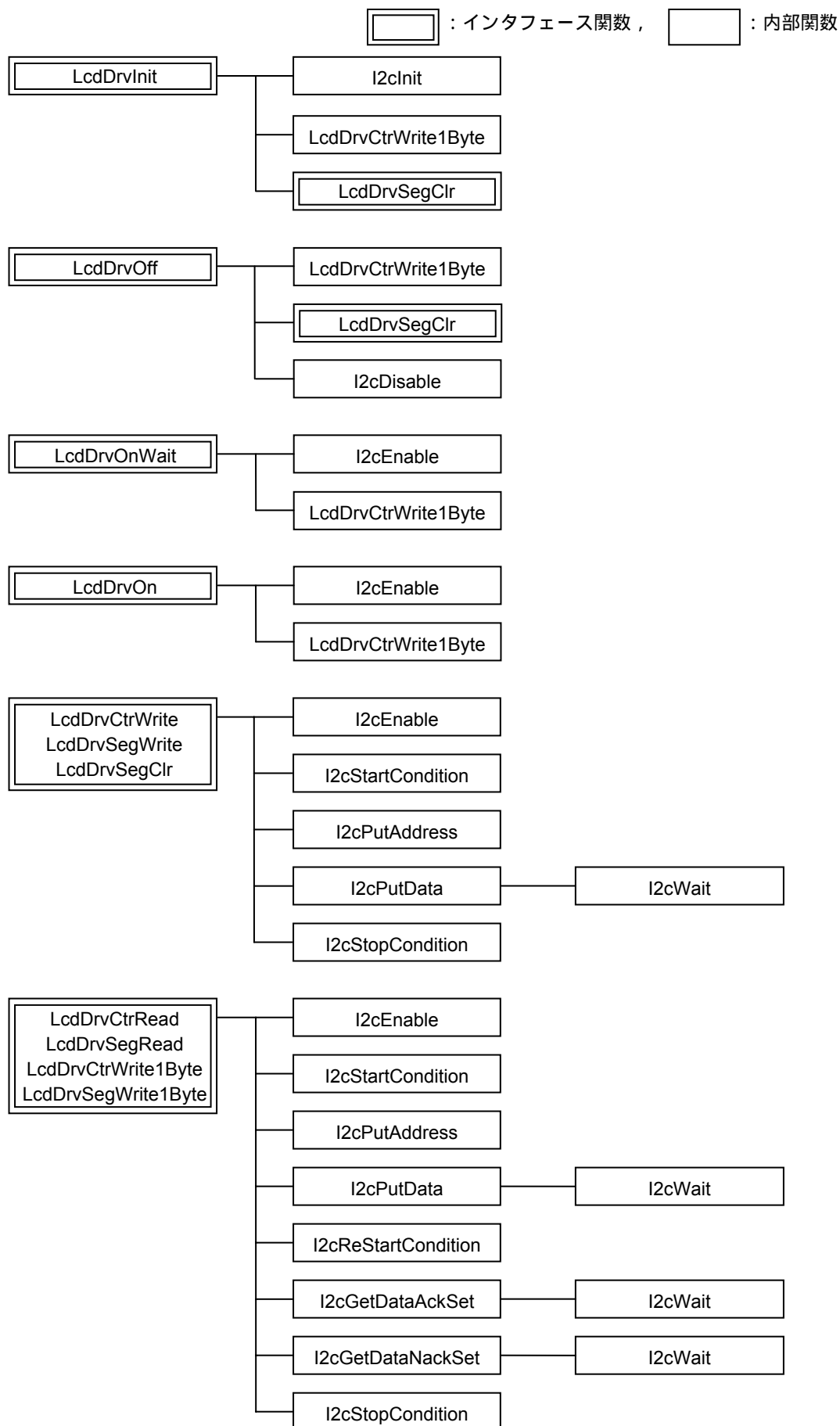
宣言	void I2cDisable(void)
処理	I ² C バス動作停止設定処理
引数	-
戻り値	-
内容	I ² C バスが動作中であれば、動作停止設定を行います。
備考	-

I2cEnable

宣言	void I2cEnable(void)
処理	I ² C バス動作許可設定処理
引数	-
戻り値	-
内容	I ² C バスが動作停止中であれば、動作許可設定を行います。
備考	-

9. 関数構成図

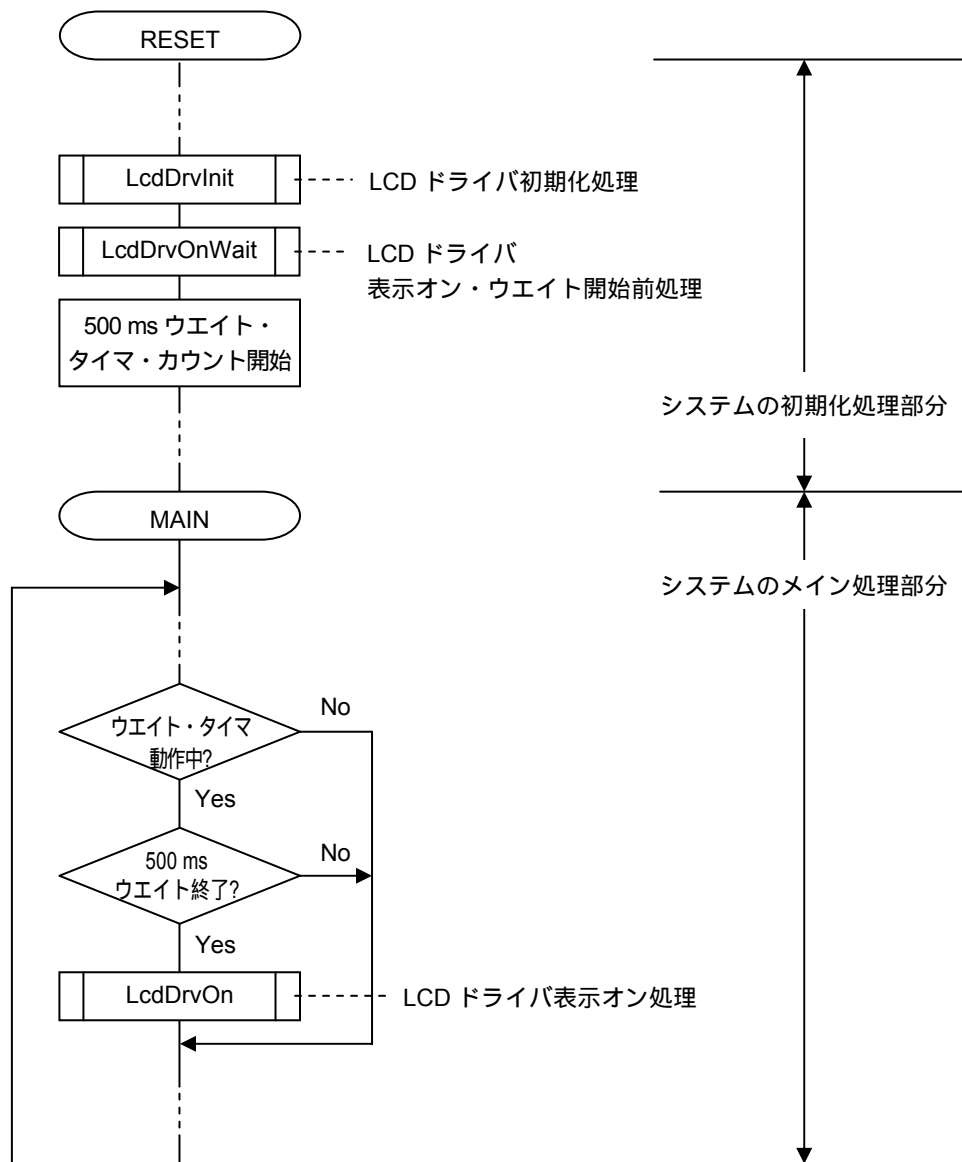
LcdDrv.c ファイル内のインタフェース関数と I2C.c ファイル内の内部関数との関係を示します。



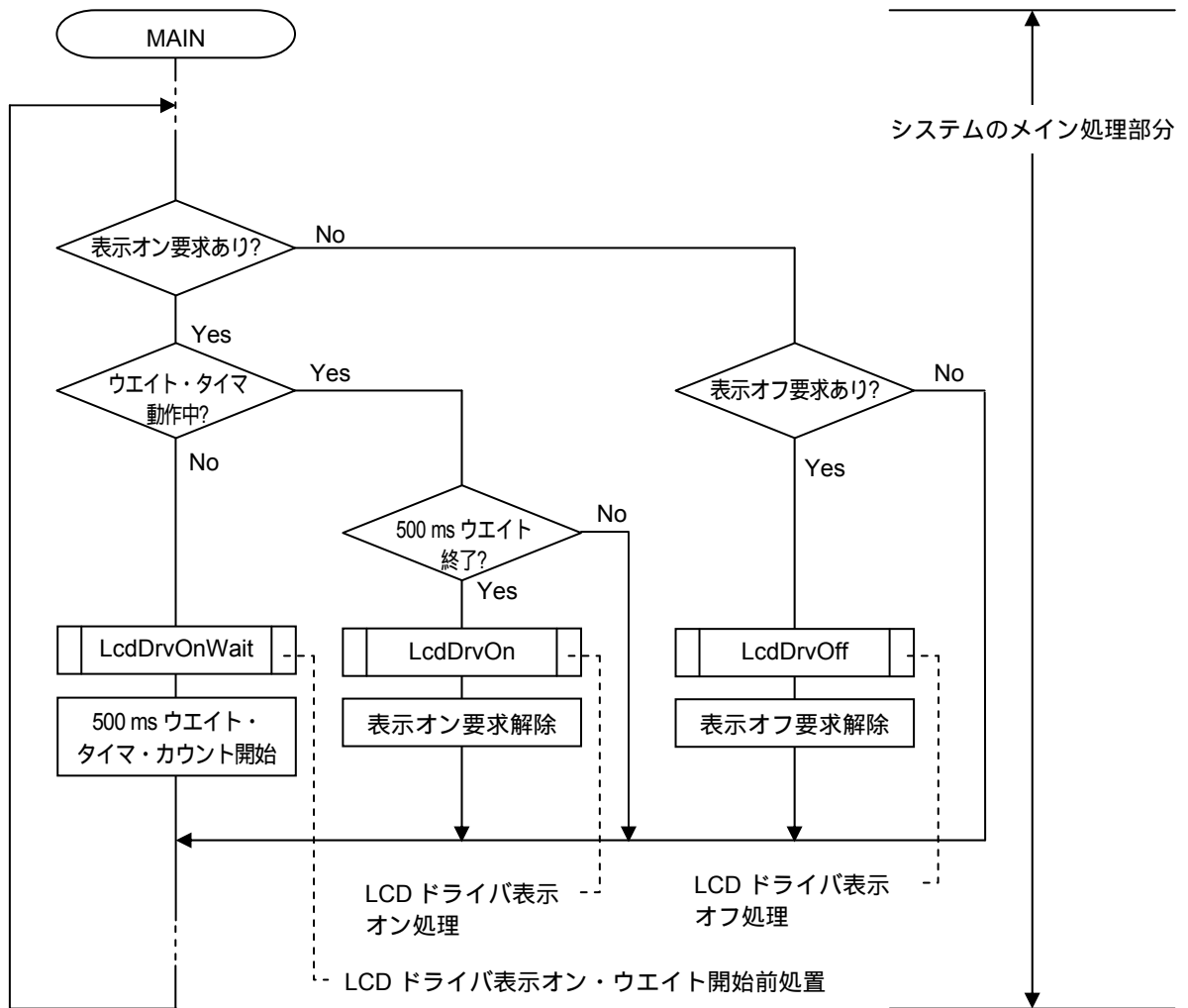
10. 内部昇圧方式選択時のインタフェース関数の呼び出し手順

定義ファイル "LcdDrv.h" にて, LCD 基準電圧生成回路を内部昇圧方式とした場合, 次に示す手順で処理を呼び出します。

10.1 ドライバ初期化～表示オン処理の呼び出し手順



10.2 表示オフ/表示オン処理の呼び出し手順



[メ モ]

[メ モ]

[メ モ]

- 本資料に記載されている内容は2006年2月現在のもので、今後、予告なく変更することがあります。量産設計の際には最新の個別データ・シート等をご参照ください。
- 文書による当社の事前の承諾なしに本資料の転載複製を禁じます。当社は、本資料の誤りに関し、一切その責を負いません。
- 当社は、本資料に記載された当社製品の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、一切その責を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
- 本資料に記載された回路、ソフトウェアおよびこれらに関する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責を負いません。
- 当社は、当社製品の品質、信頼性の向上に努めておりますが、当社製品の不具合が完全に発生しないことを保証するものではありません。当社製品の不具合により生じた生命、身体および財産に対する損害の危険を最小限度にするために、冗長設計、延焼対策設計、誤動作防止設計等安全設計を行ってください。
- 当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定していただく「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。

標準水準：コンピュータ、OA機器、通信機器、計測機器、AV機器、家電、工作機械、パーソナル機器、産業用ロボット

特別水準：輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器

特定水準：航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器、生命維持のための装置またはシステム等

当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。意図されていない用途で当社製品の使用をお客様が希望する場合には、事前に当社販売窓口までお問い合わせください。

(注)

- (1) 本事項において使用されている「当社」とは、NECエレクトロニクス株式会社およびNECエレクトロニクス株式会社がその総株主の議決権の過半数を直接または間接に保有する会社をいう。
- (2) 本事項において使用されている「当社製品」とは、(1)において定義された当社の開発、製造製品をいう。

M8E 02.11

【発行】

NECエレクトロニクス株式会社

〒211-8668 神奈川県川崎市中原区下沼部1753

電話(代表)：044(435)5111

お問い合わせ先

【ホームページ】

NECエレクトロニクスの情報がインターネットでご覧になれます。

URL(アドレス) <http://www.necel.co.jp/>

【営業関係、技術関係お問い合わせ先】

半導体ホットライン

(電話：午前 9:00～12:00、午後 1:00～5:00)

電話：044-435-9494

E-mail：info@necel.com

【資料請求先】

NECエレクトロニクスのホームページよりダウンロードいただくか、NECエレクトロニクスの販売特約店へお申し付けください。