

HT series

HT1040

---

ユーザーズマニュアル



# 目次

---

---

<b>1</b>	<b>はじめに</b> .....	<b>1</b>
<b>2</b>	<b>注意事項</b> .....	<b>2</b>
2.1	安全に関する注意事項 .....	2
2.2	取り扱い上の注意事項 .....	2
2.3	ソフトウェア使用に関しての注意事項 .....	3
<b>3</b>	<b>資料・参考文献</b> .....	<b>4</b>
<b>4</b>	<b>ガイドツアー</b> .....	<b>5</b>
4.1	必要なハードウェア・ソフトウェア .....	5
4.2	ライタープログラムのインストール .....	6
4.3	準備 .....	6
4.4	電源の接続 .....	7
4.5	S ファイルローダのコマンド .....	9
4.6	S フォーマットファイルを転送・実行する .....	9
4.7	FLASH メモリの内容書き換え .....	10
4.8	プログラム開発手順のまとめ .....	12
<b>5</b>	<b>仕様</b> .....	<b>13</b>
<b>6</b>	<b>ハードウェア機能</b> .....	<b>14</b>
6.1	ブロック図 .....	14
6.2	コネクタ .....	15
6.2.1	コネクタピン配列 .....	15
6.2.2	CN1 信号機能 .....	17
6.2.3	CN2 信号機能 .....	18
6.2.4	CN2A 信号機能 .....	20
6.2.5	CN3 信号機能 .....	20
6.2.6	CN4 信号機能 .....	20
6.2.7	CN5 信号機能 .....	21
6.2.8	CN6/CN6A 信号機能 .....	21
6.2.9	CN7/CN7A/CN7B 信号機能 .....	21
6.2.10	CN8 .....	22
6.2.11	コネクタ型式 .....	22
6.3	ジャンパ設定 .....	22
6.3.1	動作モード(JP1,2) .....	23

6.3.2	ブートモード・ユーザモード選択(JP3)	23
6.3.3	バスパワー(JP4)	24
6.3.4	A/D コンバータ(JP5,6,7)	24
6.3.5	オンボード SRAM 有効・無効(JP8)	24
<b>6.4</b>	<b>モードスイッチ(SW1)</b>	<b>25</b>
<b>6.5</b>	<b>TP</b>	<b>25</b>
<b>6.6</b>	<b>メモリ,I/O マップ</b>	<b>26</b>
6.6.1	512KB SRAM	26
6.6.2	H8/3069 内蔵 SRAM	26
6.6.3	H8/3069 内蔵 FLASH メモリ	27
6.6.4	CN1 拡張バス空間	27
6.6.5	ポート V	27
6.6.6	W5100	27
<b>6.7</b>	<b>H8/3069 内蔵 I/O レジスタ</b>	<b>27</b>
6.7.1	ASTCR	28
6.7.2	BCR	28
6.7.3	DIVCR	29
6.7.4	IER	30
6.7.5	IPRA	30
6.7.6	ISCR	31
6.7.7	ISR	31
6.7.8	P1DDR	32
6.7.9	P2DDR	32
6.7.10	P5DDR	32
6.7.11	P8DDR	33
6.7.12	PBDDR	33
6.7.13	SYSCR	34
6.7.14	WCRL	35
<b>6.8</b>	<b>外部割り込み</b>	<b>35</b>
<b>6.9</b>	<b>カレンダー時計</b>	<b>36</b>
6.9.1	H8/3069 との接続回路	36
6.9.2	内部レジスタマップ	37
<b>6.10</b>	<b>W5100</b>	<b>37</b>
6.10.1	コモンレジスタ	38
6.10.2	モードレジスタ(MR)	39
6.10.3	ゲートウェイアドレスレジスタ(GAR)	39
6.10.4	サブネットマスクレジスタ(SUBR)	39
6.10.5	ソースハードウェアアドレスレジスタ(SHAR)	40
6.10.6	ソース IP アドレスレジスタ(SIPR)	40
6.10.7	インタラプトレジスタ(IR)	40
6.10.8	インタラプトマスクレジスタ(IMR)	41
6.10.9	リトライタイムアウトレジスタ(RTR)	42
6.10.10	リトライカウントレジスタ(RCR)	42
6.10.11	Rx メモリサイズレジスタ(RMSR)	42
6.10.12	Tx メモリサイズレジスタ(TMSR)	43
6.10.13	PPPoE 認証タイプレジスタ(PATR)	43
6.10.14	PPP LCP リクエストタイマ(PTIMER)	43
6.10.15	PPPLCP マジックナンバーレジスタ(PMAGIC)	43

6.10.16	アンリーチャブルIP アドレスレジスタ(UIPR) .....	44
6.10.17	アンリーチャブルポートレジスタ(UPORT) .....	44
6.10.18	ソケットレジスタ .....	44
6.10.19	ソケット N モードレジスタ(SN_MR) .....	48
6.10.20	ソケット N コマンドレジスタ(SN_CR) .....	49
6.10.21	ソケット N インタラプトレジスタ(SN_IR) .....	50
6.10.22	ソケット N ステータスレジスタ(SN_SR) .....	50
6.10.23	ソケット N ソースポートレジスタ(SN_PORT) .....	51
6.10.24	ソケット N デスティネーションハードウェアアドレスレジスタ(SN_DHAR) .....	52
6.10.25	ソケット N デスティネーション IP アドレスレジスタ(SN_DIPR) .....	52
6.10.26	ソケット N デスティネーションポートレジスタ(SN_DPORT) .....	52
6.10.27	ソケット N 最大セグメントサイズレジスタ(SN_MSS) .....	53
6.10.28	ソケット N IP プロトコルレジスタ(SN_PROTO) .....	53
6.10.29	ソケット N TOS レジスタ(SN_TOS) .....	53
6.10.30	ソケット N TTL レジスタ(SN_TTL) .....	53
6.10.31	ソケット N TX フリーサイズレジスタ(SN_TX_FSR) .....	53
6.10.32	ソケット N TX リードポイントレジスタ(SN_TX_RR) .....	54
6.10.33	ソケット N TX ライトポイントレジスタ(SN_TX_WR) .....	54
6.10.34	ソケット N RX 受信サイズレジスタ(SN_RX_RSR) .....	54
6.10.35	ソケット N RX リードポイントレジスタ(SN_RX_RD) .....	54
<b>6.11</b>	<b>VNC1L .....</b>	<b>54</b>
6.11.1	ファームウェア .....	54
6.11.2	H8/3069 との接続回路 .....	55
6.11.3	VNC1L /RESET .....	56
6.11.4	/RI .....	56
6.11.5	/RTS, /CTS .....	56
6.11.6	/DATAREQ, /DATAACK .....	56
6.11.7	汎用ポート .....	57
6.11.8	LED .....	57
6.11.9	MAX1607 スイッチ .....	58
6.11.10	ファームウェアの更新 .....	58
<b>6.12</b>	<b>ポート V .....</b>	<b>58</b>
<b>6.13</b>	<b>ブートモード .....</b>	<b>59</b>
<b>6.14</b>	<b>バスタイミング .....</b>	<b>60</b>
<b>6.15</b>	<b>バックアップ .....</b>	<b>61</b>
<b>6.16</b>	<b>拡張バス接続 .....</b>	<b>61</b>
<b>6.17</b>	<b>RM1 ~ 4 .....</b>	<b>61</b>
<b>7</b>	<b>ソフトウェア .....</b>	<b>62</b>
<b>7.1</b>	<b>H8/3069 レジスタ設定 .....</b>	<b>62</b>
7.1.1	システムクロックの設定 .....	62
7.1.2	アドレス出力 .....	62
7.1.3	アクセスステート数とウェイト数 .....	63
7.1.4	/CS 出力設定 .....	63
7.1.5	IOCHRDY 有効・無効の設定 .....	63
7.1.6	SRAM(512KB)を使用するための設定 .....	64
7.1.7	CN1 拡張バスを使用するための設定 .....	64

7.1.8	W5100 を使用するための設定 .....	64
7.2	MAC アドレスの取得 .....	65
7.3	YELLOW IDE/YELLOW SCOPE を使用する .....	65
7.3.1	ローダプログラム .....	65
7.3.2	スタートアッププログラム .....	67
7.4	VNC1L .....	68
7.4.1	アプリケーションで使用するための設定 .....	68
7.4.2	TIPS .....	68
7.4.3	ファームウェアの初期設定内容 .....	68
7.4.4	コマンドモード・データモード .....	69
7.4.5	モニタコマンドセット .....	69
7.4.6	数値モード .....	69
7.4.7	ファイル名 .....	70
7.4.8	コマンド応答 .....	71
7.4.9	イベント通知 .....	72
7.4.10	コマンド .....	72
<b>8</b>	<b>ライブラリ .....</b>	<b>76</b>
8.1	ライブラリの使用方法 .....	76
8.2	時刻取得・設定 .....	76
8.2.1	RTC_GETTIME .....	76
8.2.2	RTC_SETTIME .....	77
8.3	TCP/IP .....	77
8.3.1	GETMAC_FROM_FLASH .....	77
8.3.2	SOCKET .....	77
8.3.3	SCLOSE .....	78
8.3.4	CONNECT .....	78
8.3.5	DISCONNECT .....	78
8.3.6	LISTEN .....	78
8.3.7	SEND .....	79
8.3.8	RECV .....	79
8.3.9	SENDTO .....	79
8.3.10	RECVFROM .....	80
8.3.11	RECEIVED_LENGTH .....	80
8.3.12	PACKET_AVAILABLE .....	80
8.3.13	RECEIVED_FIN .....	81
8.3.14	IINCHIP_INIT .....	81
8.3.15	SYSINIT .....	81
8.3.16	GETISR .....	81
8.3.17	PUTISR .....	82
8.3.18	GETIINCHIP_RxMAX .....	82
8.3.19	GETIINCHIP_TxMAX .....	82
8.3.20	GETIINCHIP_RxMASK .....	82
8.3.21	GETIINCHIP_TxMASK .....	83
8.3.22	GETIINCHIP_RxBASE .....	83
8.3.23	GETIINCHIP_TxBASE .....	83
8.3.24	SETGAR .....	83

8.3.25	SETSUBR.....	84
8.3.26	SETSHAR .....	84
8.3.27	SETSIPR .....	84
8.3.28	SETRTR .....	84
8.3.29	SETRTR .....	84
8.3.30	SETIMR .....	85
8.3.31	GETGAR .....	85
8.3.32	GETSUBR.....	85
8.3.33	GETSHAR.....	85
8.3.34	GETSIPR.....	86
8.3.35	GETIR .....	86
8.3.36	SETSN_MSS.....	86
8.3.37	SETSN_PROTO.....	86
8.3.38	GETSN_IR.....	87
8.3.39	GETSN_SR .....	87
8.3.40	GETSN_TX_FSR .....	87
8.3.41	GETSN_RX_FSR .....	87
8.3.42	SETSN_DHAR.....	87
8.3.43	SETSN_DIPR .....	88
8.3.44	SETSN_DPORT.....	88
8.3.45	SETSN_TTL .....	88
8.3.46	SETMR.....	88
<b>8.4</b>	<b>USB コントローラ(ファイル操作) .....</b>	<b>89</b>
8.4.1	USB_INIT .....	89
8.4.2	USB_SEND .....	89
8.4.3	USB_OPEN .....	89
8.4.4	USB_CLOSE.....	89
8.4.5	USB_LSEEK.....	90
8.4.6	USB_FPRINTF .....	90
8.4.7	USB_READ .....	90
8.4.8	USB_BREAD .....	91
8.4.9	USB_WRITE.....	91
8.4.10	USB_FGETC .....	91
8.4.11	USB_FGETS.....	91
8.4.12	USB_FPUTC .....	92
8.4.13	USB_FILELENGTH.....	92
8.4.14	USB_MKDIR .....	92
8.4.15	USB_CHDIR.....	92
8.4.16	USB_REMOVE .....	93
8.4.17	VNC1LCMD_EXEC .....	93
8.4.18	USB_SYNC .....	93
8.4.19	GET_REPLY.....	94
8.4.20	WAIT_REPLY .....	94
8.4.21	GOTO_DATAMODE.....	94
8.4.22	GOTO_CMDMODE.....	94
8.4.23	USB_WAIT1MS .....	95

<b>9</b>	<b><u>サンプルプログラム</u></b> .....	<b>96</b>
9.1	ファイル作成 .....	96
9.2	時刻設定・取得 .....	96
9.3	HTTP アクセス.....	96
9.4	TCP クライアント.....	97
9.5	USB マウス .....	97
9.6	FT232.....	97
9.7	USB 接続プリンタ .....	97
9.8	USB キーボード.....	98
9.9	キャラクタ LCD 表示器 .....	98
<b>10</b>	<b><u>内蔵 FLASH メモリ書き込み</u></b> .....	<b>99</b>
10.1	モードスイッチ・ジャンパ設定.....	99
10.2	書き込みソフトウェア.....	99
10.3	出荷時の FLASH メモリ内容 .....	99
10.4	書き込みに使用するポート.....	100
10.5	FLASH メモリ書き換え時の注意 .....	100
<b>11</b>	<b><u>ハードウェア回路増設例</u></b> .....	<b>101</b>
11.1	82C55 増設例.....	101
11.2	データバスバッファリング例 .....	102
<b>12</b>	<b><u>ICE 接続モデル</u></b> .....	<b>103</b>
12.1	HT1040-SKA.....	103
12.2	HT1040-SKB.....	103
<b>13</b>	<b><u>ユーティリティリファレンス</u></b> .....	<b>104</b>
13.1	<b>S ファイルローダ</b> .....	<b>104</b>
13.1.1	D(メモリ内容表示).....	106
13.1.2	E(メモリ内容変更).....	106
13.1.3	I(ポート入力).....	107
13.1.4	O(ポート出力).....	108
13.1.5	G(実行) .....	108
13.1.6	L(S フォーマットファイルロード).....	108
13.1.7	DISK(USB メモリコマンド).....	109
13.1.7.1	DIR .....	109
13.1.7.2	CD .....	109
13.1.7.3	MD.....	110
13.1.7.4	RD .....	110
13.1.7.5	DEL .....	110
13.1.7.6	DUMP .....	110
13.1.7.7	TYPE .....	110

13.1.7.8	L .....	111
13.1.8	NET(TCP/IP コマンド) .....	111
13.1.8.1	INFO .....	111
13.1.8.2	IP .....	111
13.1.8.3	MASK .....	112
13.1.8.4	GW .....	112
13.1.8.5	DHCP .....	112
13.1.8.6	FTP .....	112
13.1.8.7	FTPD .....	112
13.1.9	RTC(時刻表示・設定) .....	113
13.1.10	USB(VNC1L テスト) .....	113
13.1.11	VT(ベクタテーブルモード) .....	113
13.1.12	割り込みプログラムの実行 .....	114
13.2	HT1040WR .....	115
14	<u>外形寸法図</u> .....	<u>116</u>

## 目次

図 4-1 接続ケーブルコネクタピン配置 .....	7
図 4-2 ソフトウェア開発手順 .....	12
図 6-1 HT1040 ブロック図 .....	14
図 6-2 HT1030-03 パソコン接続ケーブル結線 .....	20
図 6-3 JP5 ~ 8 位置 .....	23
図 6-4 メモリマップ .....	26
図 6-5 アクセスステートコントロールレジスタ(ASTCR)の構成 .....	28
図 6-6 バスコントロールレジスタ(BCR)の構成 .....	29
図 6-7 分周比コントロールレジスタ(DIVCR)の構成 .....	29
図 6-8 IRQ イネーブルレジスタ(IER)の構成 .....	30
図 6-9 インタラプトプライオリティレジスタ(IPRA)の構成 .....	30
図 6-10 IRQ センスコントロールレジスタ(ISCR)の構成 .....	31
図 6-11 IRQ ステータスレジスタ(ISR)の構成 .....	31
図 6-12 ポート 1 データディレクションレジスタ(P1DDR)の構成 .....	32
図 6-13 ポート 2 データディレクションレジスタ(P2DDR)の構成 .....	32
図 6-14 ポート 5 データディレクションレジスタ(P5DDR)の構成 .....	33
図 6-15 ポート 8 データディレクションレジスタ(P8DDR)の構成 .....	33
図 6-16 ポート B データディレクションレジスタ(PBDDR)の構成 .....	34
図 6-17 システムコントロールレジスタ(SYSCR)の構成 .....	34
図 6-18 ウェイトコントロールレジスタ(WCRL)の構成 .....	35
図 6-19 RX-4045SA 周辺回路 .....	36
図 6-20 W5100 レジスタ・バッファメモリマップ .....	38
図 6-21 モードレジスタ(MR)の構成 .....	39
図 6-22 ゲートウェイアドレスレジスタ(GAR)の構成 .....	39
図 6-23 サブネットマスクレジスタ(SUBR)の設定例 .....	39
図 6-24 ソースハードウェアアドレスレジスタ(SHAR)の設定例 .....	40
図 6-25 ソース IP アドレスレジスタ(SIPR)の設定例 .....	40
図 6-26 インタラプトレジスタ(IR)の構成 .....	40
図 6-27 インタラプトレジスタ(IMR)の構成 .....	42
図 6-28 リトライタイムアウトレジスタ(RTR)の設定例 .....	42
図 6-29 RX メモリサイズレジスタ(RMSR)の構成 .....	42
図 6-30 アンリーチャブル IP アドレスレジスタ(UIPR)の内容例 .....	44
図 6-31 アンリーチャブルポートレジスタ(UPORT)の内容例 .....	44
図 6-32 ソケット N モードレジスタ(SN_MR)の構成 .....	48
図 6-33 ソケット N モードレジスタ(SN_IR)の構成 .....	50
図 6-34 ソケット N ソースポートレジスタ(SN_PORT)の内容例 .....	52
図 6-35 デスティネーションハードウェアアドレスレジスタ(SN_DHAR)の設定例 .....	52
図 6-36 ソケット N デスティネーション IP アドレスレジスタ(SN_DIPR)の内容例 .....	52
図 6-37 ソケット N デスティネーションポートレジスタ(SN_DPORT)の内容例 .....	53
図 6-38 ソケット N 最大セグメントサイズレジスタ(SN_MSS)の内容例 .....	53
図 6-39 ソケット N Tx フリーサイズレジスタ(SN_TX_FSR)の内容例 .....	53
図 6-40 VNC1L 周辺回路 .....	55
図 6-41 LED 接続回路例 .....	57
図 6-42 I/O、メモリアクセスタイミング .....	60
図 9-1 キャラクタ LCD 表示器接続例 .....	98
図 11-1 82C55 増設回路例 .....	101
図 11-2 データバスバッファリング例 .....	102

图 14-1 外形寸法图.....	116
-------------------	-----

## 表目次

表 5-1 HT1040 仕様	13
表 6-1 CN1 信号配列	15
表 6-2 CN2 信号配列	15
表 6-3 CN3 信号配列	16
表 6-4 CN4 信号配列	16
表 6-5 CN2A 信号配列	16
表 6-6 CN6 信号配列	16
表 6-7 CN7 信号配列	16
表 6-8 CN8 信号配列	16
表 6-9 CN1 信号機能	17
表 6-10 CN2 信号機能	18
表 6-11 CN3 信号機能	20
表 6-12 CN4 信号機能	20
表 6-13 CN5 LED 機能	21
表 6-14 CN6 信号機能	21
表 6-15 CN7 信号機能	21
表 6-16 CN8 信号機能	22
表 6-17 コネクタ型式	22
表 6-18 H8/3069 動作モード設定	23
表 6-19 オプションメモリタイプのジャンパ設定	23
表 6-20 バスパワー選択	24
表 6-21 A/D コンバータ電源周辺ジャンパ設定	24
表 6-22 SRAM 有効・無効	24
表 6-23 モードスイッチ設定	25
表 6-24 TP 機能	25
表 6-25 HT1040 初期設定に関連する内蔵 I/O アドレス	28
表 6-26 ASTCR のビット機能	28
表 6-27 BCR のビット機能	29
表 6-28 DIVCR のビット機能	29
表 6-29 IER のビット機能	30
表 6-30 IPRA のビット機能	30
表 6-31 ISCR のビット機能	31
表 6-32 ISR のビット機能	31
表 6-33 P1DDR のビット機能	32
表 6-34 P2DDR のビット機能	32
表 6-35 P5DDR のビット機能	33
表 6-36 P8DDR のビット機能	33
表 6-37 PBDDR のビット機能	34
表 6-38 SYSCR のビット機能	34
表 6-39 WCRL のビット機能	35
表 6-40 デバイスアクセスタイム	35
表 6-41 H8/3069 の割込み要因とベクタナンバー	35
表 6-42 PBDR のビット機能(PB0=H の場合)	37
表 6-43 RTC 内レジスタアドレス	37
表 6-44 W5100 コモンレジスタ	38
表 6-45 MR のビット機能	39
表 6-46 IR のビット機能	41

表 6-47 IMR のビット機能 .....	42
表 6-48 RMSR のビット機能.....	43
表 6-49 PATR 設定値と認証方式.....	43
表 6-50 W5100 ソケット 0 レジスタ .....	45
表 6-51 W5100 ソケット 1 レジスタ .....	46
表 6-52 W5100 ソケット 2 レジスタ .....	47
表 6-53 W5100 ソケット 3 レジスタ .....	48
表 6-54 SN_MR のビット機能 .....	49
表 6-55 SN_CR に書き込みするコマンドと動作.....	49
表 6-56 SN_IR のビット機能 .....	50
表 6-57 SN_SR のステータス .....	51
表 6-58 ファームウェア機能.....	55
表 6-59 PBDR のビット機能(PB0=L の場合) .....	55
表 6-60 VNC1L 汎用ポート.....	57
表 6-61 LED 機能 .....	57
表 6-62 アクセスタイミング規定.....	60
表 6-63 $T_{w1}, T_{w2}$ .....	61
表 6-64 RM とブルアップ・ブルダウンされるポート .....	61
表 7-1 VNC1L コマンド応答 .....	71
表 7-2 イベント通知メッセージ.....	72
表 7-3 モニタコンフィグレーションコマンド.....	72
表 7-4 ディスクコマンド.....	73
表 7-5 パワーマネージメント .....	73
表 7-6 未使用 I/O ピンコマンド .....	74
表 7-7 プリンタコマンド.....	74
表 7-8 USB デバイスコマンド.....	74
表 7-9 FT232/FT2345/FT2232 用コマンド .....	74
表 7-10 デバッグコマンド.....	74
表 12-1 HT1040-SKA 適合インサーキットエミュレータ .....	103
表 12-2 HT1040-SKB 適合インサーキットエミュレータ .....	103
表 13-1 S ファイルローダのコマンド .....	104
表 13-2 DISK サブコマンド .....	109
表 13-3 NET サブコマンド.....	111



# 1 はじめに

---

このたびは HT1040 をお求めいただき、ありがとうございます。

HT1040 は Renesas H8/3069(HD64F3069R)を採用したシングルボードコンピュータです。ボード上にはハードウェア TCP/IP デバイス Wiznet W5100 や、FAT ファイルシステムを内蔵した USB ホストコントローラ FTDI VNC1L のほか、カレンダー時計や SRAM512KB、RS232C シリアル 2 チャンネルなど制御用途に必要とされる周辺回路も搭載しています。

CPU の H8/3069 は 16 ビットの汎用レジスタを 16 本もち、メモリ空間も最大 16MB をリニアに扱うことができるほか、シリアルインターフェース、パラレルインターフェース、タイマや A/D コンバータ等が集積されており、制御用途に最適なマイクロプロセッサです。

W5100 は 100baseTX/10baseT に対応し TCP/IP をハードウェアで処理するため、アプリケーションではプロトコルスタックを用意する必要がありません。

VNC1L は FAT ファイルシステムを内蔵した USB コントローラで、CPU には非同期シリアルポートで接続されており、単純なコマンドでファイルへのデータ書き込み、読み出しが可能です。

拡張バスは 8 ビット PC/104 に準拠した信号配列になっており、弊社 HT シリーズ等の拡張モジュールをスタッキング接続して使用することができます。拡張モジュールにはオプトアイソレート I/O や GPIB、パルスモータコントローラなどが用意されています。

H8/3069 内蔵フラッシュメモリの書き込みには特別なライターが不要で、パソコンからシリアルポートを通じて転送して書き込みすることができます。

本マニュアルは、HT1040 のハードウェア・ソフトウェアの仕様や使用方法について書かれたものです。HT1040 の機能を最大限引き出すために、ご活用いただければ幸いです。

## 2 注意事項

---

### 2.1 安全に関する注意事項

HT1040 を安全にご使用いただくために、特に以下の点にご注意くださいますようお願いいたします。



本製品には一般電子機器用（OA機器・通信機器・計測機器・工作機械等）に製造された半導体部品を使用しておりますので、その誤作動や故障が直接生命を脅かしたり、身体・財産等に危害を及ぼす恐れのある装置（医療機器・交通機器・燃焼制御・安全装置等）に組み込んで使用しないでください。

また、半導体部品を使用した製品は、外来ノイズやサージにより誤作動したり故障したりする可能性がありますので、ご使用になる場合は万一誤作動、故障した場合においても生命・身体・財産等が侵害されることのないよう、装置としての安全設計（リミットスイッチやヒューズ・ブレーカ等の保護回路の設置、装置の多重化等）に万全を期されますようお願い申し上げます。

### 2.2 取り扱い上の注意事項

HT1040 に恒久的なダメージをあたえないよう、取り扱い時には以下のような点にご注意ください。

- 電源の投入  
HT1040 や周辺回路に電源が供給されている状態では絶対に本ボードの着脱を行わないでください。
- 静電気  
HT1040 には CMOS デバイスを使用しておりますので、ご使用になるまでは帯電防止対策されている出荷時のパッケージ等にて保管してください。
- ラッチアップ  
電源および入出力からの過大なノイズやサージ、電源電圧の急激な変動等で使用している CMOS デバイスがラッチアップを起こす可能性があります。いったんラッチアップ状態となると、電源を切断しないかぎりこの状態が維持されるため、デバイスの破損につながる可能性があります。ノイズの影響を受けやすい入出力ラインには保護回路を入れることや、ノイズ源となる装置と共通の電源を使用しない等の対策をとることをお勧めします。

## 2.3 ソフトウェア使用に関する注意事項

本製品のために弊社から提供するソフトウェア（ドキュメント・サンプル等も含む）は、現状のまま（AS IS）提供されるものであり、特定の目的に適合することや、その信頼性、正確性を保証するものではありません。また、本製品の使用による結果についてもなんら保証するものではありません。

## 3 資料・参考文献

---

本マニュアル記載の内容を補完する資料・参考文献を以下に示します。

- Renesas H8/3069R F-ZTAT TM ハードウェアマニュアル (資料番号 RJJ09B0165-0600)
- Renesas H8/300H シリーズ プログラミングマニュアル (資料番号 RJJ09B0141-0400)  
ハードウェアマニュアルには H8/3069 の詳細な機能が、プログラミングマニュアルは H8/3069 の CPU 命令詳細が記載されています。株式会社ルネサステクノロジの WEB サイトから PDF ファイル形式でダウンロードすることができます。(http://japan.renesas.com/)
- エプソントヨコム Real Time Clock Module RX-4045SA/NB アプリケーションマニュアル(資料番号 ETM01J-02)  
搭載されているカレンダー時計の仕様・使用方法について説明されています。エプソントヨコム株式会社の WEB サイトから PDF ファイル形式でダウンロードすることができます。(http://www.epsontoyocom.co.jp/)
- W5100 Datasheet (version1.1.8)  
詳細なレジスタの説明や、プログラミング方法が説明されています。Wiznet の WEB サイトから PDF ファイル形式でダウンロードすることができます。(http://www.wiznet.co.kr/)
- Vinculum VNC1L Data Sheet version 2.00
- Vinculum firmware User Manual Version 2.05  
VNC1L のデータシートおよびファームウェアの仕様や使用方法が説明されています。FTDI の Vinculum WEB サイトから PDF ファイル形式でダウンロードすることができます。(http://www.vinculum.com/)

# 4 ガイドツアー

---

この章では HT1040 を初めて使用することを想定し、最低限の動作確認とフラッシュメモリの書き換え方法を簡単な例を交えて説明します。HT1040 と4.1節で説明するソフトウェア・ハードウェアを準備し、説明に従って実際に HT1040 を動作させてみることをお勧めします。

## 4.1 必要なハードウェア・ソフトウェア

- PC 接続ケーブル  
HT1040 にパソコンおよび電源を接続するためのケーブルが必要です。このケーブルは製品には付属せず別売ですが、コネクタ結線は図 6-2に示されています。HT1040-03 の PC 接続コネクタは Dsub9 ピンタイプですので、必要に応じて変換コネクタをご使用ください。
- 電源  
供給電源電圧は 5V です。HT1040 に周辺回路の接続が何もない場合は、300 mA 程度が供給可能な電源をご用意ください。周辺にその他のボードを接続する場合は、各ボードの消費電流に応じた容量の電源をご用意ください。



HT1040 には電源電圧検出しリセット回路がありますので、電源電圧が規定よりも低くなるとボードがリセットされてしまいます。動作が不安定とならないよう、電源には安定化した 5V を使用してください。

- 通信ソフトウェア  
お使いのパソコンで使用できる通信ソフトウェア（ターミナルソフトウェアとも呼ばれます）をご用意ください。パソコンから S フォーマット形式のファイルを HT1040 に転送しますので、通信ソフトウェアはテキストファイル送信機能をサポートしている必要があります。  
Windows XP にはハイパーターミナルという通信ソフトウェアが標準で添付されており、テキストファイル送信がサポートされています。（スタートからプログラム アクセサリ 通信にフォルダが見つからない場合は、アプリケーションの追加と削除を使ってインストールする必要があります。）Windows Vista には通信ソフトウェアは含まれていませんので、TeraTerm 等のフリーソフトウェアをご利用ください。

## 4.2 ライタプログラムのインストール

Windows 上で動作する FLASH メモリ書き込みプログラムは、マニュアルディスクに添付されています。

弊社 WEB サイト(<http://www.umezawa.co.jp/>)からも入手することができます。

¥flashwr ディレクトリ

HT1040WR.EXE

FLASH メモリライタープログラム

¥loader ディレクトリ

SLDR1040.BIN

S ファイルローダ(出荷時のフラッシュメモリ内容)

KEYCODE.MOT

S ファイル転送実行テスト用プログラム

FKKEYCODE.MOT

FLASH メモリ書き込みテスト用プログラム

FLASH メモリ書き込みプログラムにはインストーラやセットアッププログラムがありませんので、必要に応じてスタートメニューへの登録や、ショートカット作成等を行ってください。FLASH メモリ書き込みプログラムは HT1040WR.EXE をダブルクリックする等の方法で実行することができます。



アンインストールするには...

HT1040WR.EXE を削除してください。Windows のシステムディレクトリに組み込まれた DLL 等はありません。

## 4.3 準備

パソコンと HT1040 を接続して動作させる前に、次の手順で通信ソフトウェアの設定をして、通信ソフトウェアが正しく機能していることを確認してください。

### A. 通信ソフトウェアの設定

お使いの通信ソフトウェアのヘルプ・マニュアル等を参照して、通信に必要なパラメータを設定します。主な設定項目は次の通りです。

通信に使用するポート	使用するシリアルポート番号
通信速度	38400
データビット長	8
ストップビット長	1 または 2
パリティビット	なし
ローカルエコー	なし
リターンキー入力	CR(0DH)のみを送信
バックスペース入力	BS(08H)を送信

B. パソコンへのケーブル接続

パソコン接続ケーブルの Dsub コネクタ側をパソコンに接続します。この段階では、ケーブルの逆端にある黒い 5 ピンコネクタは HT1040 に接続しないでください。

C. ループバック接続

細い電線か部品のリード切り屑等を使って、HT1040 に接続する側の黒い 5 ピンコネクタの 2 ピン、3 ピン間をショートさせます。コネクタのピン配置は下図をご参照ください。

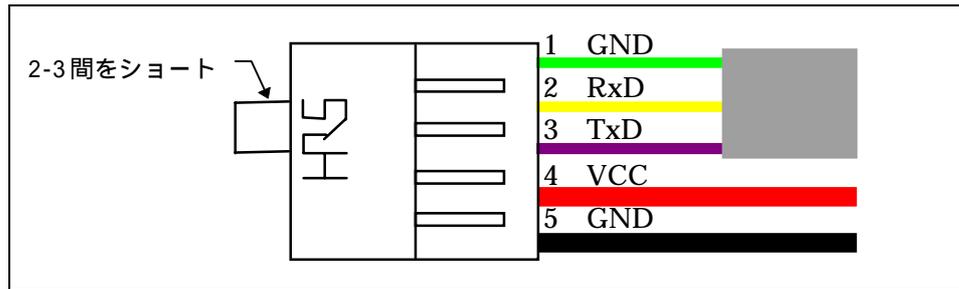


図 4-1 接続ケーブルコネクタピン配置

D. 通信ソフトウェアのテスト

キーボードの文字・数字キーを押すと、押したキーに相当する文字が画面に表示されれば正常です。

キーボードから文字を送っても画面に何も表示されない場合は....



- 選択した通信ポートの設定  
通信用に指定したポートは有効になっていますか？
- ケーブルの異常  
ケーブルの断線や、ループバックの配線不良がないか、テスターで導通を確認します。コネクタの結線は6.2.5項をご参照ください。

E. ケーブルの GND 接続テスト

ループバックテストだけではパソコン接続ケーブルの GND 断線が検出できませんので、以上の手順でキーを押すと正常に表示ができることを確認したら、最後に現在ショートしている 2-3 ピンを、さらに 1 ピンとショートしてからキーを押し、今度は画面に表示がでないことを確認してください。万一この接続を加えても表示ができる場合は接続ケーブルの GND が断線している可能性がありますのでテスターで導通を確認してください。

F. HT1040 モードスイッチ、ジャンパソケットの設定

スイッチ SW1 のレバーを下向き(基板側向き)に設定してください。出荷時のジャンパを変更している場合は、JP1:2-3 JP2:1-2 JP3:1-2 に設定してください。

## 4.4 電源の接続

通信ソフトウェアの動作に問題がなければ、パソコンは通信ソフトウェアを立ちあげたままにして、HT1040 接続用の 5 極コネクタの 1-2-3 間ショートをはずし、HT1040

に接続します。コネクタ上の HRS のロゴマークがジャンパ設定表のある面になるよう接続します。このコネクタには電源接続用の電線が配線されていますので、極性を誤らないようにして電源を接続し、電源を投入します。(4ピンに接続されている赤色電線が+5V、5ピンに接続されている黒色電線がGNDです。)



開発中は、実験用安定化電源装置等の電流制限がかけられるものを使用し、適切なりミットを設定して使用することをお勧めします。また万一極性を誤った場合や、5Vを超える電圧が加えられた場合に備えて、ボード外部にツェナーダイオードやヒューズ等で構成した保護回路を接続しておくのも効果的です。

正常であれば、画面に以下のようなメッセージが表示されます。(表示されるリビジョンや、日付は以下に示すものとは異なる場合があります。)

```
HT1040 S Record File Loader ver.1.00(May 21, 2009)
Copyright (c) Umezawa Musen Denki Co.Ltd. 2009
-
```

斜体文字部分は  
キーボードから  
の入力を表しま  
す。

キーボードから、?と入力し Enter キーを押すと、コマンド一覧が表示されます。

```
-?
Available commands
Dump                D/DB/DW/DL
Examine             E/EB/EW/EL
Load S-format file  L
Port                I/IB/IW/IL/O/OB/OW/OL
Excution           G
RTC                 RTC
VNC1L command mode USB
TCP/IP commands    NET
USB Disk commands  DISK
Vector table mode  VTY/VTE
-
```



画面に何も表示があらわれない場合は....

- 電源電圧・極性の確認  
電源電圧と、極性が正しいことをテスター等で確認してください。
- コネクタ挿入極性の確認  
ジャンパ設定表のある面を上にしたときに、PC 接続用 5 極コネクタ上にかかれた HRS のロゴが見えます。逆接続されていた場合は正しい向きに入れ直してください。
- ボードの不良  
テスターで消費電流を測定してみてください。正常であれば HT1040 単独で使用している場合、250 ~ 260mA 程度になります。極端に少ない、あるいは多い場合はボード不良の可能性がります。
- FLASH メモリ内容の問題  
FLASH メモリには、出荷時に S ファイルローダが書きこまれています。この内容をアプリケーションプログラムで上書きした場合は、S ファイルローダが動作しません。この場合は、8章の説明にしたがって、FLASH メモリの内容を S ファイルローダプログラムで更新してください。
- モードスイッチの設定  
モードスイッチが USER 側(下側)になっていることを確認してください。

コンソールには何か出力されているが、正しい文字になっていない場合

- 通信速度の設定

通信ソフトウェアのボーレート設定が 38400bps 以外になっている可能性があります。4.3節のループバックテストではボーレートの設定が違っていることはチェックできませんので、再度通信ソフトウェアの設定を確認してください。

## 4.5 S ファイルローダのコマンド

前項までの動作に問題がなければ、コマンドのいくつかを試してみてください。(コマンドの詳細は13.1節をご参照ください。)

たとえば D コマンドでは、メモリ内容を表示させることができます。

```
-D 400000
400000 5E 40 01 3A 5E 40 00 20 0C 80 A0 61 4C 04 0C 08 ^@.:^@. .. aL...
400010 54 70 A0 7A 4F 04 0C 08 54 70 80 E0 0C 08 54 70 Tp z0...Tp...Tp
400020 6D F6 5E 40 00 C0 0C 8E A8 FF 47 F6 0C E8 55 D8 m.^@.タ..イ.G...Uリ
400030 5E 40 01 18 40 EC 54 70 01 00 6D F6 0F 86 40 06 ^@..@.Tp..m...@.
400040 6C 68 5C 00 00 D2 68 68 46 F6 01 00 6D 76 54 70 lh¥..メhhF...mvTp
400050 01 00 6D F6 7E BC 73 60 47 60 29 BD 0C 98 A8 11 ..m.~シs`G`)ス..イ.
400060 47 08 A8 13 46 0E F8 01 40 02 18 88 6A A8 00 40 G.イ.F...@...j イ.@
400070 02 08 40 42 7A 06 00 40 02 02 01 00 69 60 0B 70 ..@Bz..@....i`.p
-
```

E コマンドを使うと、メモリ内容を更新することができます。

```
-E 400000
400000 5E 12
400001 40 34
400002 01 56
400003 3A 78
400004 5E .
-D 400000 L 10
400000 12 34 56 78 5E 40 00 20 0C 80 A0 61 4C 04 0C 08 .4Vx^@. .. aL...
-
```

以降の節では、簡単なプログラムを PC から転送して HT1040 上で実行したり、FLASH メモリの内容を書き換えする手順について説明します。

## 4.6 S フォーマットファイルを転送・実行する

ここでは、例としてライタープログラム解凍時に作成された¥loader ディレクトリにある、keycode.mot を転送・実行する手順を説明します。

1. PC で通信ソフトウェアを起動し、L を入力して ENTER キーを押します。  
このとき、画面にはプロンプトが表示されません。
2. 通信ソフトウェアのテキストファイル送信機能を使用して、keycode.mot ファイルを送信します。(PC の通信ソフトウェアとしてハイパーターミナルを使用している場合は、転送 テキストファイルの送信の順にメニューを選択します。)

3. ファイル送信が完了すると、S フォーマットファイルで指定されているプログラムスタートアドレスが表示され、コマンドプロンプトに戻ります。

```
-L  
Start Address:400000  
-
```

4. G を入力し ENTER キーを押してください。プログラムは押されたキーコードを表示します。ESC キーを押すとプログラムは終了し、コマンドプロンプトに戻ります。

```
-G  
Key code dump program  
54 45 53 54 0D 1B  
Returned from G command  
-
```

## 4.7 FLASH メモリの内容書き換え

ここでは、例としてライタープログラム解凍時に作成された¥loader ディレクトリにある、fkeycode.mot を FLASH メモリに転送・書き込みする手順を説明します。プログラムの動作は前節でテストしたものと同じです。(ただしプログラムは終了できません。)

1. 通信ソフトウェアを実行中の場合は、通信ソフトウェアをいったん終了します。(FLASH メモリ書き込みプログラムが同じ通信ポートを使用するため、競合するためです。)
2. HT1040WR.EXE プログラムを起動します。
3. HT1040 を接続しているシリアルポートを選択します。
4. モードスイッチ SW1 を BOOT 側(上側)にあげてブートモードを設定します。設定が済んだら HT1040 に電源を供給します。
5. File Send で書き込みする MOT ファイルを選択します。この例では fkeycode.mot を選んで開くを押してください。
6. 書き込み中のメッセージが表示されたのち、正常に終了すると書き込み終了のメッセージが表示されます。
7. HT1040 の電源を切断し、モードスイッチを USER 側(基板方向)に倒します。ジャンパ JP1,JP2,JP3 が次のように設定されていることを確認してください。  
JP1:2-3 JP2:1-2 JP3:1-2
8. FLASH メモリ書き込みプログラムを終了し、通信ソフトウェアを起動します。
9. HT1040 に電源を供給すると、前節と同じプログラムが動作します。



COMx Open Error!と表示される

- 指定された COM ポートが使用できません。存在しないポート番号を指定しているか、他のソフトウェアと競合している可能性があります。他の通信ソフトウェアを立ち上げている場合は、いったん終了してください。

Handshake Error あるいは Handshake Timeout Error と表示される

- HT1040 との通信に失敗しました。FLASH メモリ書き込みモードにするためのジャンパ設定が正しく設定されているか、HT1040 に正しく電源が接続されているかどうかを確認してください。

Flash Programming Error と表示される

- FLASH メモリ書き込みに失敗しました。HT1040 の電源を再投入して書き込みをやり直してみてください。

使用したい COM ポートが一覧に表示されない

- 指定された COM ポートを他のプログラムが使用中の場合、一覧には表示されません。他のプログラム(通信ソフトウェア等)をいったん終了してください。

## 4.8 プログラム開発手順のまとめ

本章の最後に、プログラム開発の流れをまとめておきます。

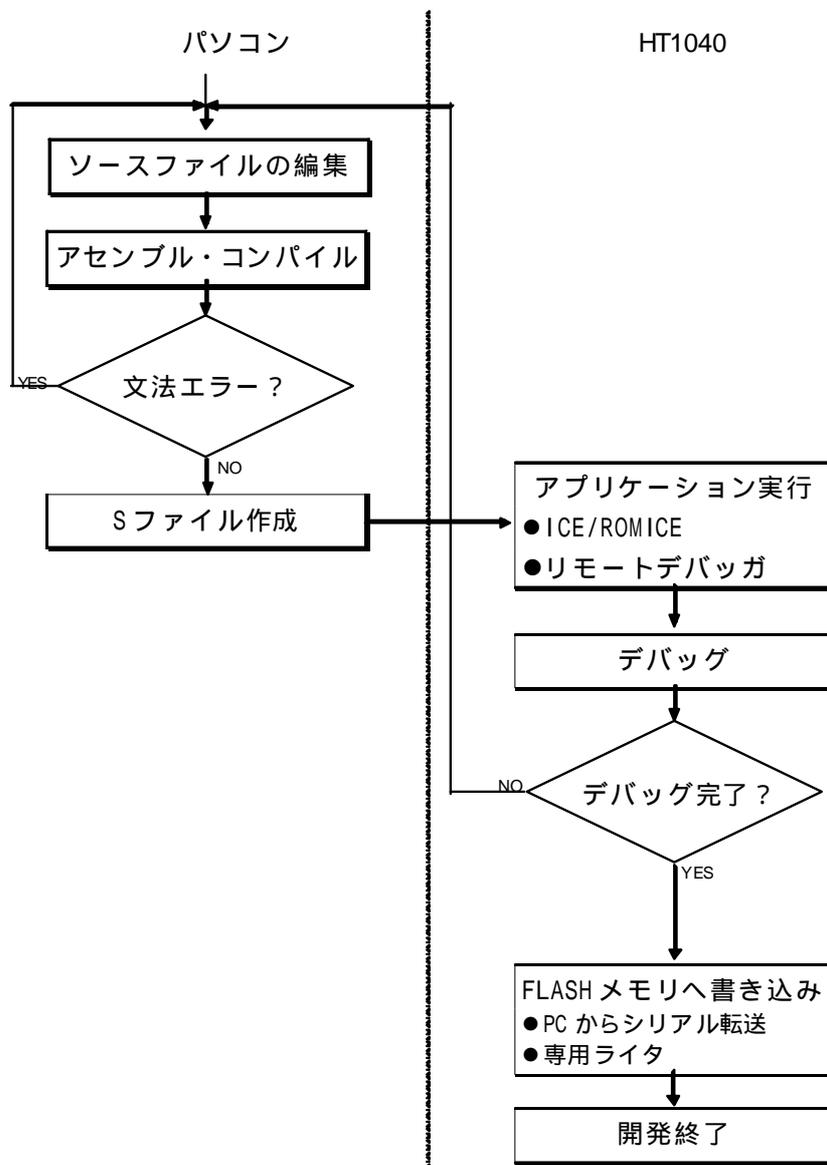


図 4-2 ソフトウェア開発手順

## 5 仕様

本ボードの主な仕様を表 5-1に示します。

表 5-1 HT1040 仕様

プロセッサ	日立 HD64F3069RF25V(H8/3069R)
システムクロック	16MHz(原発振、分周可能)
メモリ	SRAM 16KB(CPU 内蔵,バックアップ無) SRAM 512KB(バックアップ付) FLASH 512KB(CPU 内蔵、書き換え回数 100 回 min.)
システムリセット	電圧検出リセット部:S80842(セイコー電子) 検出電圧 4.099V(Min.) 4.200V(Typ.) 4.301V(Max.) 外部リセット入力
カレンダー時計	エプソントヨコム RX-4045SA(バックアップ付)
シリアルポート	2 チャンネル チャンネル 0,1 RS232C (チャンネル 1 は FLASH メモリ書き換えに使用)
ボーレートジェネレータ	各チャンネル独立、CPU 内蔵
汎用入出力	35 ビット(一部端子は他の I/O 機能と重複)
A/D コンバータ	CPU 内蔵 10 ビット 8 チャンネル(変換時間 3.5 $\mu$ s)
D/A コンバータ	CPU 内蔵 8 ビット 2 チャンネル(変換時間 10 $\mu$ s)
タイマ	16 ビット 3 チャンネル 8 ビット 4 チャンネル
ウォチドッグタイマ	CPU 内蔵
バックアップ	0.1F 電気二重層コンデンサ(電池併用可能) 512KB SRAM 最低保持電圧 2.0V カレンダー時計 計時最低電圧 1.15V バックアップ電流 1.45 $\mu$ A Typ. (@3.0V)
基板サイズ	90.2 $\times$ 95.9 $\times$ 15.2mm ( 突出部を含まず )
電源電圧	5V $\pm$ 5%
消費電流	動作時 250mA(Typ.)
動作温度範囲	0 ~ 60

# 6 ハードウェア機能

この章では、HT1040 のハードウェア機能に関連する事項について説明します。  
H8/3069 の機能詳細については、ルネサステクノロジ発行の H8/3069R ハードウェア  
マニュアルをご参照ください。

## 6.1 ブロック図

図 6-1に HT1040 のブロック図を示します。

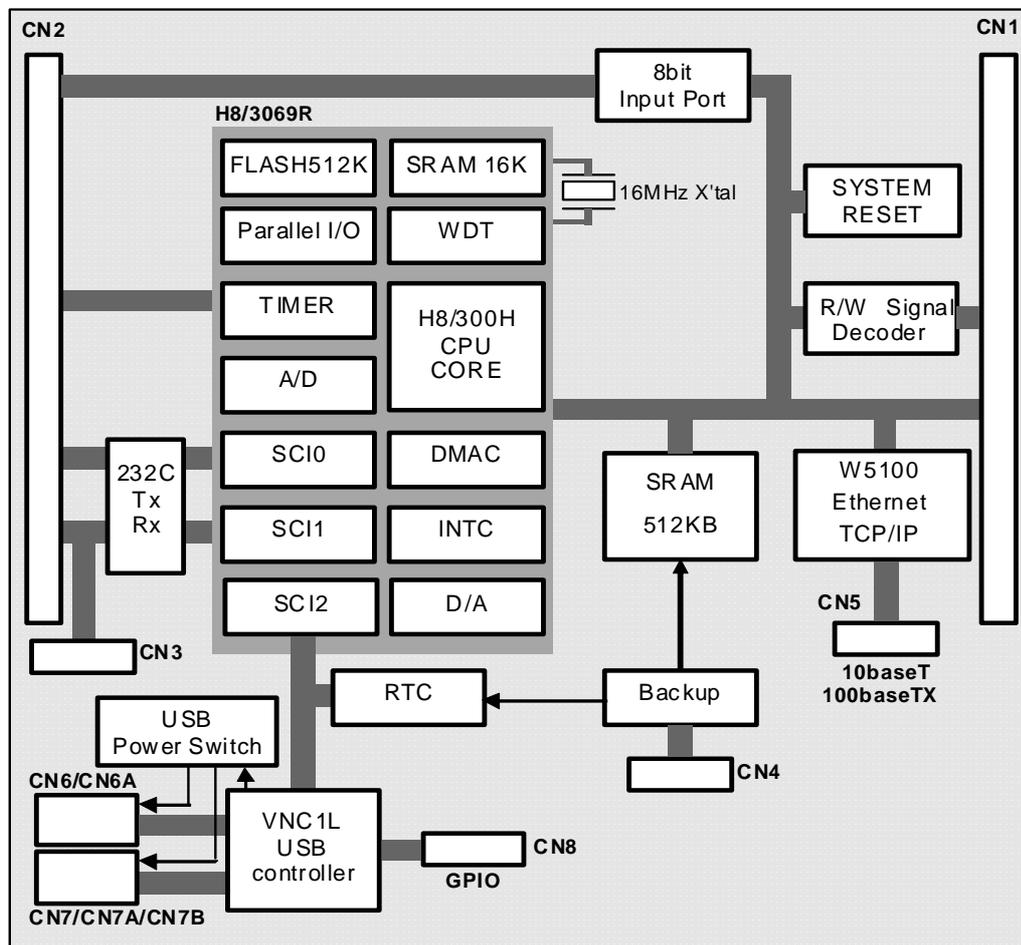


図 6-1 HT1040 ブロック図

## 6.2 コネクタ

### 6.2.1 コネクタピン配列

表 6-1から表 6-8にコネクタの信号配列を示します。表中、/印の付いた信号名はその信号が負論理であることを示します。-印の端子は未使用です。また、/TXD0、/RXD0、/TXD1、/RXD1 は RS232C レベルです。

表 6-1 CN1 信号配列

A1	/IOCHCHK	B1	GND
A2	SD7	B2	RESETDRV
A3	SD6	B3	+5V
A4	SD5	B4	-
A5	SD4	B5	-
A6	SD3	B6	-
A7	SD2	B7	-
A8	SD1	B8	-
A9	SD0	B9	-
A10	IOCHRDY	B10	GND
A11	AEN	B11	/SMEMW
A12	SA19	B12	/SMEMR
A13	SA18	B13	/IOW
A14	SA17	B14	/IOR
A15	SA16	B15	(+5VPullUp)
A16	SA15	B16	-
A17	SA14	B17	(+5VPullUp)
A18	SA13	B18	-
A19	SA12	B19	(+5VPullUp)
A20	SA11	B20	SYSCLK
A21	SA10	B21	IRQ7
A22	SA9	B22	-
A23	SA8	B23	IRQ5
A24	SA7	B24	IRQ4
A25	SA6	B25	-
A26	SA5	B26	(+5VPullUp)
A27	SA4	B27	(+5VPullUp)
A28	SA3	B28	(+5VPullUp)
A29	SA2	B29	+5V
A30	SA1	B30	-
A31	SA0	B31	GND
A32	GND	B32	GND

表 6-2 CN2 信号配列

E1	/TXD1	F1	/RXD1
E2	/TXD0	F2	/RXD0
E3	GND	F3	SYSCLK(P67)
E4	/LWR(P66)	F4	/WR(P65)
E5	/RD(P64)	F5	/AS(P63)
E6	P62(/BACK)	F6	P61(/BREQ)
E7	/IOCHRDY(P60)	F7	/CS2(P82,IRQ2)
E8	/CS0(P84)	F8	AVCC
E9	VREF	F9	AN0
E10	AN1	F10	AN2
E11	AN3	F11	AN4
E12	AN5	F12	AN6
E13	AN7	F13	AVSS
E14	PA0	F14	PA1
E15	PA2	F15	PA3
E16	PA4	F16	PA5
E17	PA6	F17	PA7
E18	/VPROG	F18	VCC
E19	PV0	F19	PV1
E20	PV2	F20	PV3
E21	PV4	F21	PV5
E22	PV6	F22	PV7
E23	TXD0	F23	RXD0
E24	FWEX	F24	/SHDN232
E25	/IORESET	F25	/RTCINT
E26	SCK0	F26	SCK1
E27	P40	F27	P41
E28	P42	F28	P43
E29	P44	F29	P45
E30	P46	F30	P47
E31	VCC	F31	GND
E32	BACKUP	F32	/EXRESET

表 6-3 CN3 信号配列

1	GND
2	/RxD1
3	/TxD1
4	+5V
5	GND

表 6-4 CN4 信号配列

1	+VBAT
2	GND

表 6-5 CN2A 信号配列

1	VCC	2	GND
3	/TXD1	4	/RXD1
5	/TXD0	6	/RXD0
7	AN0	8	AN1
9	AN2	10	AN3
11	AN4	12	AN5
13	AN6	14	AN7
15	PA0	16	PA1
17	PA2	18	PA3
19	PA4	20	PA5
21	PA6	22	PA7
23	PV0	24	PV1
25	PV2	26	PV3
27	PV4	28	PV5
29	PV6	30	PV7
31	P40	32	P41
33	P42	34	P43
35	P44	36	P45
37	P46	38	P47
39	SCK0	40	SCK1

表 6-7 CN7 信号配列

1	USBPOWER
2	USB1P
3	USB1M
4	GND
5	VCC
6	POWERSENS

表 6-8 CN8 信号配列

1	BD0
2	BD1
3	BD2
4	BD3
5	BD4
6	BD5
7	BD6
8	BD7
9	3.3V
10	GND

表 6-6 CN6 信号配列

1	USBPOWER
2	USB2P
3	USB2M
4	GND

## 6.2.2 CN1 信号機能

CN1 には、CPU バス関連信号を配置しています。信号配置は PC/104 に準拠していますが、バスドライブ能力やタイミング等完全には互換ではありませんのでご注意ください。

表 6-9 CN1 信号機能

信号名	機能
SYSCLK	システムクロック出力です。HT1040 では H8/3069 の CLK 出力がそのまま接続されていますので、 <b>デフォルトでは 16MHz が出力されます。</b> バスに接続される拡張モジュールで、SYSCLK に 8MHz を必要とするものがある場合は、H8/3069 のクロック分周器を設定してこの端子に 8MHz が供給されるようにしてください。設定方法は7.1.1節をご参照ください。H8/3069 のスタンバイ機能を使用することで、この出力を停止することができます。
SA[19:0]	アドレス出力です。この端子は H8/3069 のポート 1,2,5 と兼用となっており、 <b>H8/3069 の動作モード 5,7 のときデフォルトではアドレス出力となります。</b> ポート 1,2,5 のディレクションレジスタを設定してアドレス出力としてください。設定方法は7.1.2節をご参照ください。
AEN	この端子はプルダウンで L に固定されています。PC/104 バスに接続される拡張モジュールは、通常 I/O アドレスデコードにこの信号が L であることを使用しています。
SD[7:0]	データ入出力バスです。
/SMEMR	PC/104 バスに拡張されるモジュールに対する、メモリリード信号です。H8/3069 のアドレス空間 280000-2FFFFFF が PC/104 バスのメモリ空間 80000-FFFFFF に対応します。 <b>デフォルトではこの機能は無効となっています。</b> 7.1.7節をご参照ください。
/SMEMW	PC/104 バスに拡張されるモジュールに対する、メモリライト信号です。H8/3069 のアドレス空間 280000-2FFFFFF が PC/104 バスのメモリ空間 80000-FFFFFF に対応します。 <b>デフォルトではこの機能は無効となっています。</b> 7.1.7節をご参照ください。
/IOR	PC/104 バスに拡張されるモジュールに対する、I/O リード信号です。H8/3069 のアドレス空間 200000-20FFFF が PC/104 バスの I/O 空間 0000-FFFF に対応します。 <b>デフォルトではこの機能は無効となっています。</b> 7.1.7節をご参照ください。
/IOW	PC/104 バスに拡張されるモジュールに対する、I/O ライト信号です。H8/3069 のアドレス空間 200000-20FFFF が PC/104 バスの I/O 空間 0000-FFFF に対応します。 <b>デフォルトではこの機能は無効となっています。</b> 7.1.7節をご参照ください。
IOCHRDY	H8/3069 の/WAIT 端子に接続されており、この信号が H になるまでバスサイクルが延長されます。 <b>デフォルトではこの機能は無効となっています。</b> バスに接続するモジュールがこの機能を使用する場合は、バスコントロールレジスタを設定して、この機能を有効にしてください。詳細は7.1.5節をご参照ください。
RESETDRV	リセット出力です。 <b>Hアクティブですのでご注意ください。</b> 外部リセット入力 は/EXRESET です。RESETDRV にスイッチ等を接続しないでください。
IRQ7	外部割込み入力です。H8/3069 の IRQ0 端子に割り当てられています。
IRQ4	外部割込み入力です。H8/3069 の IRQ4 端子に割り当てられています。
IRQ5	外部割込み入力です。H8/3069 の IRQ5 端子に割り当てられています。
/IOCHCHK	H8/3069 のノンマスクブル割り込み入力/NMI に接続されています。

## 6.2.3 CN2 信号機能

CN2 には主に H8/3069 の I/O 信号端子が配置されています。

表 6-10 CN2 信号機能

信号名	機能
PA[7:0]	ポート A 入出力信号です。デフォルトでは入力となっていますが、 <b>H8/3069 の動作モード 3 では PA7 が強制的に A20 出力となりますのでご注意ください。</b> RM2 を実装してプルアップまたはプルダウンすることができます。なお、これらの端子は H8/3069 内蔵の他の I/O 機能(タイマ等)と兼用です。
PV[7:0]	増設入力ポート V 入力信号です。 RM3 を実装してプルアップまたはプルダウンすることができます。ポート V は H8/3069 のアドレス 220000 でアクセスします。
P4[7:0]	ポート 4 入出力信号です。デフォルトでは入力となっています。RM1 を実装してプルアップまたはプルダウンすることができます。
CLK/(P67)	CN1 の SYSCLK と同一の信号が接続されています。
/LWR(P66)	H8/3069 の同名端子に接続されています。H8/3069 動作モード 6,7 以外では常に /LWR 出力となります。
/WR(P65)	H8/3069 の P65/HWR 端子に接続されています。H8/3069 動作モード 6,7 以外では常に /HWR 出力となります。
/RD(P64)	H8/3069 の P64/RD 端子に接続されています。H8/3069 動作モード 6,7 以外では常に /RD 出力となります。
/AS(P63)	H8/3069 の P63/AS 端子に接続されています。H8/3069 動作モード 6,7 以外では常に /AS 出力となります。
P62/(BACK)	H8/3069 の P62/BACK 端子に接続されています。デフォルトでは P62 入力となっており、出力として使用する場合は、/BACK として使用する場合は設定が必要です。
P61/(BREQ)	H8/3069 の P61/BREQ 端子に接続されています。デフォルトでは P61 入力となっており、出力として使用する場合は、/BREQ として使用する場合は設定が必要です。
/IOCHRDY (P60)	H8/3069 の P60/WAIT 端子に接続されています。デフォルトでは P60 入力となっており、出力として使用する場合は、/WAIT として使用する場合は設定が必要です。なお、この信号は CN1 の /IOCHRDY にも配置されていますので、出力として使用する場合は CN1 経由で接続される信号と競合のないようご注意ください。
P82 (/CS2,/IRQ2)	H8/3069 の P82/CS2/IRQ2 端子に接続されています。デフォルトでは P82 入力となっていますが、通常この信号はオンボード 512KBSRAM の選択信号として使用するよう設定して使用します。SRAM を使用しない場合は P82 入出力または IRQ2 として使用することができます。
P84(/CS0)	H8/3069 の P84/CS0 端子に接続されています。デフォルトでは P84 入力となっています。
SCK0 (P94,/IRQ4)	H8/3069 の P94/SCK0/IRQ4 信号が接続されています。デフォルトではこの信号は P94(汎用入出力)の入力ポートとなっています。SCK0 入出力として使用する場合は設定が必要です。この端子には、CN1 の IRQ4 反転信号が接続されています。
SCK1 (P95,/IRQ5)	H8/3069 の P95/SCK1/IRQ5 信号が接続されています。デフォルトではこの信号は P95(汎用入出力)の入力ポートとなっています。SCK1 入出力として使用する場合は設定が必要です。この端子には、CN1 の IRQ5 反転信号が接続されています。

信号名	機能
/TxD0	RS232C レベルのチャンネル 0 送信データ出力です。H8/3069 の P90/TxD0 信号がレベル変換回路を通して接続されています。デフォルトではこの信号は P90(汎用入出力)の入力ポートとなっていますので、 <b>シリアル通信に使用する場合はモードを変更してください。</b>
/RxD0	RS232C レベルのチャンネル 0 受信データ入力です。レベル変換回路を通して H8/3069 の P92/RxD0 に接続されています。デフォルトではこの信号は P92(汎用入出力)の入力ポートとなっていますので、 <b>シリアル通信に使用する場合はモードを変更してください。</b>
/TxD1	RS232C レベルのチャンネル 1 送信データ出力です。H8/3069 の P91/TxD1 信号がレベル変換回路を通して接続されています。この信号は CN3 にも接続されています。デフォルトではこの信号は P91 機能(汎用入出力)の入力ポートとなっていますので、 <b>シリアル通信に使用する場合はモードを変更してください。</b>
/RxD1	RS232C レベルのチャンネル 1 受信データ入力です。レベル変換回路を通して H8/3069 の P93/RxD1 に接続されています。この信号は CN3 にも接続されています。デフォルトではこの信号は P93 機能(汎用入出力)の入力ポートとなっていますので、 <b>シリアル通信に使用する場合はモードを変更してください。</b>
TxD0(P90)	H8/3069 の内蔵シリアルポートチャンネル 0 の送信出力端子です。この端子は P90(汎用入出力)と兼用です。デフォルトでは P90(汎用入出力)の入力ポートとなっていますので、使用用途に応じてモードを変更してください。
RxD0(P92)	この端子には、チャンネル 0 の RS232C レベル変換回路を通った受信データ出力が接続されています。この端子は H8/3069 では P92(汎用入出力)と兼用です。デフォルトでは P90(汎用入出力)の入力ポートとなっていますので、使用用途に応じてモードを変更してください。
/EXRESET	外部リセット入力端子です。 GND との間にスイッチを接続することができます。
BACKUP	メインの電源が切断された場合でもバックアップを必要とするデバイスの電源を接続します。HT1040 では 512KBSRAM と RTC の電源が接続されています。この端子には電気二重層コンデンサが接続されています。 <b>注意：外部電池はこの端子ではなく、CN4 から接続してください。</b>
AVREF	H8/3069 内蔵 A/D コンバータの基準電圧入力です。デフォルトではボード上で VCC に接続されていますが、JP7 をカットし他の電圧(ただし 0 AVREF VCC)を入力することができます。
AN[7:0]	H8/3069 の P7/AN[7:0]/DA[1:0]に接続されています。RM4 を実装してプルアップまたはプルダウンすることができます。
AVCC	H8/3069 内蔵 A/D 変換回路部分の電源端子です。ボード上で他の回路用の VCC に接続されていますが、JP5 をカットし外部から供給することができます。
AVSS	H8/3069 内蔵 A/D 変換回路部分の GND 端子です。ボード上で他の回路用の GND に接続されていますが、JP6 をカットし外部から供給することができます。
FWEX	この端子の入力信号が H8/3069 の FWE 入力に接続されます。(FWE は H8/3069 の内蔵 FLASH メモリの書き込みを許可する信号です。)
/VPROG	この端子は 10k でプルアップされており、L にすると VNC1L はファームウェア書き込みモードとなります。
/SHDN232	通常はこの端子をオープンで使用します。この端子を L にすると、ボード上の 232C ドライバがシャットダウンモードとなります。(ただしこのとき送信はできなくなります。)
/IORESET	この端子に L にすると、CN1 の RESETDRV がアクティブとなり、PC/104 バスに接続された拡張ボードをリセットすることができます。この入力により CPU はリセットされませんので、プログラムにより拡張ボードのリセットを行うことが可能です。(たとえば汎用出力ポートを使って、H8/3069 内蔵ウォッチドッグタイマ動作時に拡張ボードのリセットを出力するような場合に使用できます。)
/RTCINT	カレンダー時計からの割り込み出力端子で、オープンドレインです。

## 6.2.4 CN2A 信号機能

CN2A には、CN2 に接続されている主な信号が 40 極にまとめられており、CN2 の信号をピンヘッダ等で引き出す場合に使用することができます。信号機能については、表 6-10 の同名信号機能説明をご参照ください。

## 6.2.5 CN3 信号機能

CN3 はシリアルチャンネル 1 入出力および電源接続端子です。PC と接続してフラッシュメモリの書き換えをする場合は、この端子を使用します。

表 6-11 CN3 信号機能

信号名	機能
+5V	ボードの電源を供給します。CN1 から電源を供給する場合は接続不要です。
GND	電源およびコンソールの GND に接続します。
TxD1*	H8/3069 内蔵シリアルポートのチャンネル 1 送信出力が RS232C レベル変換回路を通して接続されています。この信号は CN2 の E1 にも接続されています。
RxD1*	RS232C レベル変換回路を通して H8/3069 内蔵シリアルポートのチャンネル 1 受信入力に接続されています。この信号は CN2 の F1 にも接続されています。

別売の PC 接続用ケーブル(HT1030-03)はこのコネクタに接続します。接続は次の通りです。

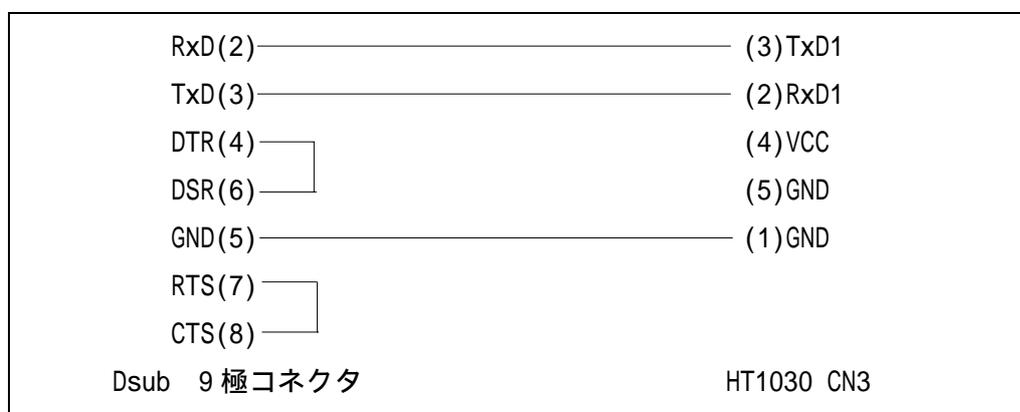


図 6-2 HT1030-03 パソコン接続ケーブル結線

## 6.2.6 CN4 信号機能

CN4 はバックアップ用の電源接続用端子です。バックアップ機能については 6.15 節をご参照ください。CN4 にはコネクタが実装されていません。

表 6-12 CN4 信号機能

信号名	機能
+VBAT	SRAM・RTC のバックアップ用に電池を接続する端子（プラス側）です。
GND	バックアップ用電池のマイナス側を接続します。

## 6.2.7 CN5 信号機能

CN5 は、10baseT/100baseTX 用パルストランス内蔵のモジュラジャックです。LED が内蔵されており、W5100 の LED 出力に接続されています。

表 6-13 CN5 LED 機能

LED	機能
黄	リンク確立で点灯
緑・オレンジ	緑点灯時は送信、オレンジ点灯時は受信

## 6.2.8 CN6/CN6A 信号機能

CN6 には VNC1L の USB ポート 2 の信号が接続されています。CN6A には USB の A タイプコネクタが実装されています。CN6 にはコネクタが実装されていません。

表 6-14 CN6 信号機能

信号名	機能
USBPOWER	MAX1607 でスイッチされた USB 電源出力です。この端子に電源を供給するためには VNC1L に設定が必要です。をご参照ください。この信号は USB ポート 1/2 共通に接続されています。
USB2M	USB ポート 2 の D-信号端子です。
USB2P	USB ポート 2 の D+信号端子です。
GND	GND

なお CN6 と CN6A を同時に使用することはできません。CN6 に搭載可能なコネクタの型式例は表 6-17をご参照ください。

## 6.2.9 CN7/CN7A/CN7B 信号機能

CN7 には VNC1L の USB ポート 1 の信号が接続されています。USB ポート 1 は、VNC1L のファームウェアによりホストまたはデバイスとして使用することができます。(出荷時に書き込まれているファームウェアはホスト用です。)ホストとして使用する場合は CN7A に A タイプ USB コネクタを、デバイスとして使用する場合は CN7B に B タイプ USB コネクタを実装することができます。CN7/CN7A/CN7B にはコネクタは実装されていません。搭載可能なコネクタの型式例は表 6-17をご参照ください。

表 6-15 CN7 信号機能

信号名	機能
USBPOWER	MAX1607 でスイッチされた USB 電源出力です。この端子に電源を供給するためには VNC1L に設定が必要です。をご参照ください。この信号は USB ポート 1/2 共通に接続されています。
USB1M	USB ポート 1 の D-信号端子です。
USB1P	USB ポート 1 の D+信号端子です。
POWERSENS	USB ポート 1 をデバイスとして使用する場合のホスト接続検出信号です。
VCC	+5V
GND	GND

なお CN7/CN7A/CN7B は同時に使用することはできませんので、使用する場合はいずれかにコネクタを実装してください。CN7B を使用する場合は、HT1040 の電源を PC から供給することも可能です。この場合は JP4 にジャンパポストを実装し、1-2 間にジャンパソケットを取り付けてください。

CN7B に B タイプの USB コネクタを実装した場合は、PC に FTDI のドライバをインストールすることで HT1040 のフラッシュメモリ書き換えを USB 経由で行うことができます。内蔵フラッシュメモリ書きこみについては10章をご参照ください。

## 6.2.10 CN8

CN8 は VNC1L の汎用入出力ポート信号が接続されています。このポートの入出力レベルは 3.3V ですのでご注意ください。(入力は 5V までの電圧が入力可能です。) CN8 にはコネクタが実装されていません。

表 6-16 CN8 信号機能

信号名	機能
BD[7:0]	VNC1L の汎用入出力信号です。デフォルトでは BD7 は入力、BD[6:0]は出力となっています。出力の初期値は L です。
+3.3V	ボードの 3.3V 電源出力が接続されています。
GND	GND

## 6.2.11 コネクタ型式

コネクタ型式等を表 6-17に示します。CN1、CN2、CN3、CN5 および CN6A は実装済みです。その他のコネクタは実装可能なコネクタの型式例です。

表 6-17 コネクタ型式

コネクタ	メーカー	型式	備考
CN1	ASTRON	25-0206-232-1G-R	PC/104 J1 スタックスルー
CN2	ASTRON	25-0201-232-1G-R	
CN3	ヒロセ電機	A2-5PA-2.54DS(71)	適合ハウジング A2-5S-2.54C 圧着ピン A1-2226SCC
CN4	JST	S2B-XH-A	
CN5	PULSE	J0G-0007NL	
CN6	JST	S4B-XH-A	
CN6A	OMRON	XM7A-0442-A	
CN7	JST	S7B-XH-A	
CN7A	OMRON	XM7B-0442	
CN7B	LUMBERG	2411 03	
CN8	ヒロセ電機	HIF3FC-10PA-2.54DS(71)	

## 6.3 ジャンパ設定

ここでは機能別にジャンパの設定について説明します。なお、3 極ポスト型のジャンパはシルクで 印がある端子から 1、2、3 番ピンとなっています。0 チップ抵抗でショートしているジャンパは基板に表示がありませんので、図 6-3をご参照ください。

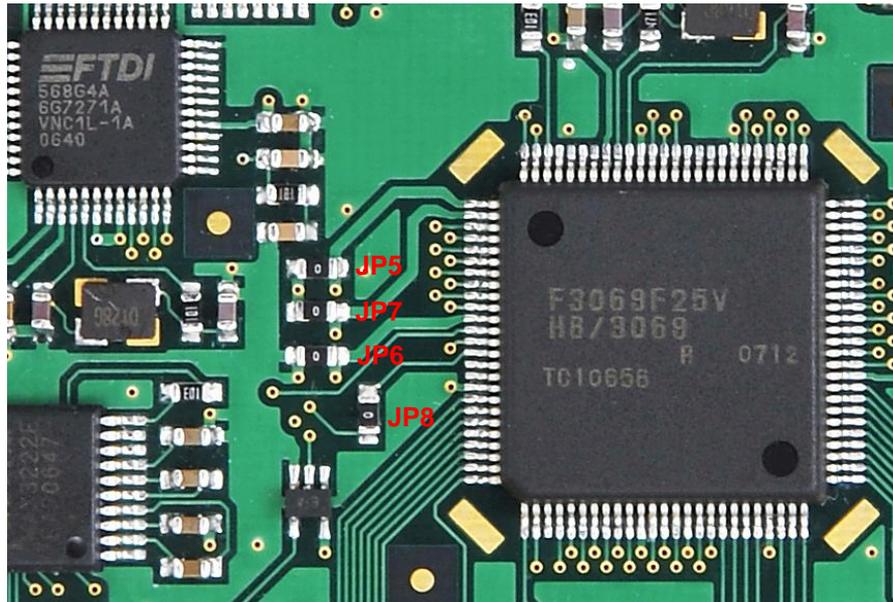


図 6-3 JP5～8 位置

### 6.3.1 動作モード(JP1,2)

JP1,2 は H8/3069 の動作モードを選択するジャンパです。H8/3069 にはモード 1 から 7 までの動作モードがありますが、HT1040 はモード 5 でご使用ください。JP1 ,JP2 の切替でモード 3 に設定することも可能ですが、モード 3 では CPU 内蔵のフラッシュメモリが無効となるため、HT1040 の外部にメモリ増設が必要です。モード 3 は ROM インサートキット形のデバuggaを接続するために ROM ソケットを拡張バスに接続して使用するような場合にご使用ください。JP1,2 は出荷時動作モードがモード 5 となるように設定されています。ジャンパは表 6-18 に示された組み合わせ以外には設定しないでください。

表 6-18 H8/3069 動作モード設定

JP1	JP2	動作モード
1-2	2-3	モード 3(内蔵フラッシュ禁止、外部バス有効)
2-3	1-2	モード 5(内蔵フラッシュ有効・外部バス有効、出荷時設定)

### 6.3.2 ブートモード・ユーザモード選択(JP3)

JP3 は、リセット時にブートモード(フラッシュメモリ書換)とするか、ユーザモード(フラッシュメモリに書き込まれているプログラム実行)とするかを設定するジャンパです。出荷時 JP3 は 1-2 に設定されており、SW1 でモード選択します。

表 6-19 オプションメモリタイプのジャンパ設定

JP3	機能
1-2	ブートモードかユーザモードかは SW1 の状態で設定されます。(出荷時設定)
2-3	ブートモード
なし	ユーザモード(ブートモード禁止)

### 6.3.3 バスパワー(JP4)

JP4 は、CN7B にタイプ B の USB コネクタを実装した場合に、この USB コネクタから HT1040 の電源を供給するかどうかを設定するジャンパです。出荷時には部品が取り付けられていませんので、USB コネクタから電源の供給を受ける場合は JP4 を実装し、1-2 にソケットを取り付けてください。

表 6-20 バスパワー選択

JP4	機能
1-2	HT1040 の電源を CN7B から供給します。
2-3	HT1040 の電源を CN7B 以外から供給します。



#### 《注意》

JP4 を 1-2 に設定した場合は、HT1040 の電源を USB コネクタ CN7B 以外から供給しないでください。外部電源と USB コネクタから供給された 5V とが競合し、接続した PC に損傷を与える可能性があります。

USB デバイスはデフォルトで最大消費電流が 100mA 以下と規定されていますが、HT1040 の消費電流は 250mA(Typ.)のため、バスパワーで接続する場合は USB 規格には適合しませんのでご注意ください。

### 6.3.4 A/D コンバータ(JP5,6,7)

JP5 から 7 は H8/3069 内蔵 A/D コンバータ部の電源と基準電圧の供給をボード内から行うか、外部から行うかの切り替えジャンパです。出荷時は全てのジャンパが 0 オームのチップ抵抗で接続されていますので、切り替えが必要な場合はこれらの抵抗を取り外してください。

表 6-21 A/D コンバータ電源周辺ジャンパ設定

ジャンパ	機能
JP5	AVCC(A/D コンバータ部の VCC)をシステムの VCC に接続しています。ジャンパをはずした場合は CN2-F8 から A/D コンバータ部の VCC を供給してください。
JP6	AVSS(A/D コンバータ部の GND)をシステムの GND に接続しています。ジャンパをはずした場合は CN2-F13 から A/D コンバータ部の GND を接続してください。
JP7	VREF(A/D コンバータの基準電圧)をシステムの VCC に接続しています。ジャンパをはずした場合は CN2-E9 から A/D コンバータの基準電圧を供給してください。

### 6.3.5 オンボード SRAM 有効・無効(JP8)

JP8 は、ボード上の SRAM を有効・無効を設定するジャンパです。出荷時にはジャンパが 0 オームのチップ抵抗で接続されており、SRAM は有効になっています。ボード上の SRAM を無効にする場合は、抵抗を取り外してください。

ボード上の SRAM を無効にした場合、CN2 に接続されている CPU の /CS2 を使用して外部にさらに大容量のメモリを増設することができます。

表 6-22 SRAM 有効・無効

JP8	機能
Short	ボード上の SRAM が使用できます。
Open	ボード上の SRAM は使用できなくなります。

## 6.4 モードスイッチ(SW1)

モードスイッチはブートモードとユーザモードを切替するスイッチです。

表 6-23 モードスイッチ設定

スイッチ位置	動作モード
USER	JP1,2 で設定された動作モード
BOOT	ブートモード

なお JP3 のジャンパソケットを取り外すと、SW1 によるブートモード設定が禁止されますので、SW1 の誤切替でブートモードに設定されることを防止できます。

## 6.5 TP

HT1040 には、TP1 から TP3 までの 3 つのテスト用端子が用意されています。部品は取り付けられていませんので、必要に応じてチェック端子等を取り付けてご使用ください。

表 6-24 TP 機能

TP	機能
TP1	GND に接続されています。
TP2	/EXRESET に接続されています。
TP3	L レベルを入力すると、ブートモードで CPU の/NMI 端子を H にします。

TP3 については6.13節をご参照ください。

## 6.6 メモリ,I/O マップ

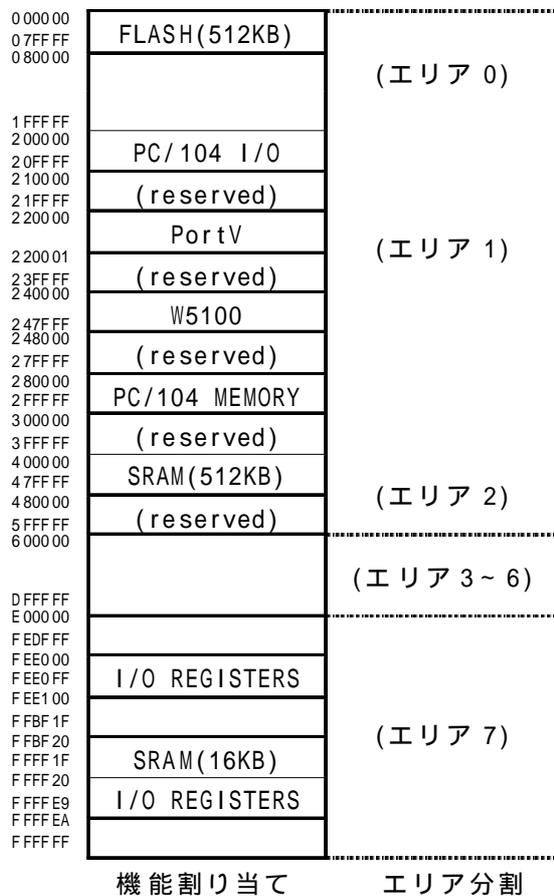


図 6-4 メモリマップ

H8/3069 のメモリ選択信号出力は、バスコントロールレジスタ(BCR)のRDEA ビットが 1 の場合、全メモリ空間が 2MB ずつ 8 つのエリアに分割され、それぞれのエリアに対応した/CS0 ~ /CS7 で接続するデバイスを選択することができます。HT1040 では、/CS1 と/CS2 のエリアを使用しています。

図 6-4に出荷時のジャンパ設定(モード 5)に対応した HT1040 のメモリマップを示します。モード 3 では先頭部分のフラッシュメモリが無効となる以外、メモリマップは同一です。H8/3069 では、I/O もメモリ空間に割り当てられます。拡張バスでサポートされる PC/104 モジュールは通常 I/O アドレス空間に割り当てられているため、HT1040 では CPU からみた特定のアドレス範囲が PC/104 バスの I/O アドレス空間となるようにマッピングされています。詳細は6.6.4項をご参照ください。

メモリマップで空白の領域は未使用です。Reserved の領域はデコードの簡略化によるイメージが出現するため、使用することができません。

### 6.6.1 512KB SRAM

出荷時のジャンパ設定(JP8 がショートの状態)で、512KB SRAM は 400000 から 47FFFFF までを占有します。ただし、SRAM を選択する/CS2 出力は電源投入直後無効となっていますので、H8/3069 の関連するレジスタを設定する必要があります。詳しくは7.1.6節をご参照ください。このメモリは電気二重層コンデンサでバックアップされています。

### 6.6.2 H8/3069 内蔵 SRAM

FFBF20 から FFFF1F までの 16KB は H8/3069 内蔵の SRAM 領域です。この SRAM は電源投入直後から有効となっていますが、システムコントロールレジスタ(SYSCR)の設定により無効にすることができます。設定については6.7.13項をご参照ください。なお、このメモリはバックアップ対象とはなりませんのでご注意ください。

## 6.6.3 H8/3069 内蔵 FLASH メモリ

000000 から 07FFFF までの 512KB は H8/3069 内蔵 FLASH メモリが占有しています。(H8/3069 動作モード 3 ではこの FLASH メモリは無効となります。)このメモリは通常作成したアプリケーションを保存するために使用されます。占有アドレスは変更することができません。FLASH メモリはブートモードによる書き換えのほか、アプリケーションプログラムからの書き換えも可能です。

アプリケーションから書き換えを行う場合は、CN2 の FWEX 端子を正しく制御する必要があります。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 18 章をご参照ください。

## 6.6.4 CN1 拡張バス空間

200000-20FFFF までは、CN1 を通して拡張する増設ボードの I/O 空間に割り当てられています。CN1 の信号配置は PC/104 規格に準拠しているため、弊社 HT シリーズの I/O モジュールをスタッキング接続することができます。PC/104 規格が想定している CPU は x86 系のバスのため、I/O およびメモリがそれぞれ別のアドレス空間をもっています。一方 H8/3069 では I/O アドレス空間の概念はなく、I/O もメモリも同一の空間にマッピングされるため、HT1040 では CN1 で拡張される I/O 空間 0000 ~ FFFF を上記のように 200000 ~ 20FFFF にマッピングしています。

280000-2FFFFFFF までは、CN1 を通して拡張する増設ボードのメモリ空間に割り当てられています。CN1 を通して増設される PC/104 バスのメモリ空間のアドレス 80000 ~ FFFFFF までとなります。(通常、80000 より下位のメモリ空間を占有する PC/104 拡張モジュールはありません。)

## 6.6.5 ポート V

ポート V は入力専用の 8 ビットポートで、アドレス 220000 番地を占有します。ポート V についてはをご参照ください。アドレスデコード簡略化のため実際には 220001 ~ 23FFFFF までイメージが存在しています。

## 6.6.6 W5100

W5100 は 240000 ~ 247FFF を占有します。

W5100 のレジスタについては 6.10 節をご参照ください。

## 6.7 H8/3069 内蔵 I/O レジスタ

H8 は I/O とメモリの区別がなくどちらもメモリ空間にマッピングされており、I/O に関連するレジスタは FEE000 ~ FEE0FF、FFFF20 ~ FFFF E9 までを占有しています。I/O 機能詳細については、H8/3069 R F-ZTAT ハードウェアマニュアルをご参照ください。

表 6-25 HT1040 初期設定に関連する内蔵 I/O アドレス

名称	R/W	アドレス	機能	リセット時
ASTCR	R/W	FEE021	アクセスステートコントロールレジスタ	FF
BCR	R/W	FEE024	バスコントロールレジスタ	C6
DIVCR	R/W	FEE01B	分周比コントロールレジスタ	FC
IER	R/W	FEE015	IRQ イネーブルレジスタ	00
IPRA	R/W	FEE018	インタラプトプライオリティレジスタ A	00
ISCR	R/W	FEE014	IRQ センسコントロールレジスタ	00
ISR	R/W	FEE016	IRQ ステータスレジスタ	00
P1DDR	W	FEE000	ポート 1 データディレクションレジスタ	FF/00
P2DDR	W	FEE001	ポート 2 データディレクションレジスタ	FF/00
P5DDR	W	FEE004	ポート 5 データディレクションレジスタ	FF/F0
P8DDR	W	FEE007	ポート 8 データディレクションレジスタ	F0/E0
PBDDR	W	FEE00A	ポート B データディレクションレジスタ	00
SYSCR	R/W	FEE012	システムコントロールレジスタ	09
WCRL	R/W	FEE023	ウェイトコントロールレジスタ L	FF

以下の項では、H8/3069 内蔵レジスタのうち HT1040 を使用するために最低限設定しなければならないレジスタを中心に簡単に説明します。

## 6.7.1 ASTCR

アクセスステートコントロールレジスタは、H8/3069 の 8 つに分割されたメモリ空間のアクセスサイクルを 2 ステートあるいは 3 ステートに設定します。(アクセスステートについては、H8/3069R F-ZTAT ハードウェアマニュアル 6.4.5 項をご参照ください。)図 6-5 に ASTCR の構成を、表 6-26 に各ビットの機能を示します。詳細は H8/3069 R F-ZTAT ハードウェアマニュアル 6.2.2 項をご参照ください。

bit7	Bit6	bit5	bit4	bit3	bit2	Bit1	bit0
AST7	AST6	AST5	AST4	AST3	AST2	AST1	AST0

図 6-5 アクセスステートコントロールレジスタ(ASTCR)の構成

表 6-26 ASTCR のビット機能

AST[7:0]	機能
0	2 ステートアクセス空間に設定します。
1	3 ステートアクセス空間に設定します。

初期値は 0xFF で全エリア 3 ステートアクセスに設定されます。HT1040 を 16MHz システムクロックで使用する場合は、3 ステートアクセスでご使用ください。8MHz で使用する場合は、2 ステートアクセスも設定可能です。

## 6.7.2 BCR

バスコントロールレジスタはアイドルサイクルの挿入許可/禁止、エリア分割単位の選択、メモリマップの選択、/WAIT 入力の許可禁止を設定するレジスタです。アイドルサイクルについては H8/3069R F-ZTAT ハードウェアマニュアル 6.9 章をご参照ください。図 6-6 に BCR の構成を、表 6-27 に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 6.2.5 項をご参照ください。

bit7	Bit6	Bit5	bit4	bit3	bit2	Bit1	bit0
ICIS1	ICIS0	BROME	BRSTS1	BRSTS0	EMC	RDEA	WAITE

図 6-6 バスコントロールレジスタ(BCR)の構成

表 6-27 BCR のビット機能

ビット	機能
ICIS1	異なるエリアの外部バスサイクル間にアイドルサイクルを挿入するかしないかを選択します。0:挿入しない 1:挿入する(初期値)
ICIS0	外部リードサイクルと外部ライトサイクルが連続する場合、バスサイクル間にアイドルサイクルを挿入するかしないかを選択します。0:挿入しない 1:挿入する(初期値)
BROME	エリア0をバーストROMインターフェースとするかどうかを選択します。0:基本バスインターフェース(初期値) 1:バーストROMインターフェース
BRSTS1	バーストROMインターフェースのバーストサイクル数 0:2 ステート(初期値) 1:3 ステート
BRSTS0	バーストアクセスワード数を選択します。0:4 ワード(初期値) 1:8 ワード
EMC	メモリマップを変更します。初期値は1です。(H8/3069R F-ZTAT ハードウェアマニュアル 3.6 節をご参照ください。)
RDEA	エリア分割単位を選択します。 0: エリア分割不均一(H8/3069R F-ZTAT ハードウェアマニュアル 6.3.1 項をご参照ください。) 1:エリア分割は2MBとなります。
WAITE	/WAIT 端子によるウェイト入力の許可禁止を選択します。 0:禁止(初期値) 1:許可

初期値は0xC6ですが、/WAIT入力(CN1 IOCHRDY入力)を使用する場合は、0xC7を設定してください。外部接続回路およびオプションメモリソケットを使用しない場合は、ICIS0/ICIS1を0に設定してアクセスサイクルを早めることが可能です。バスを使用して増設される各デバイスのデータバス出力イネーブル・ディスエーブル時間を考慮してアイドルサイクルが不要かどうか検討することができますが、通常はデフォルトのまま1を設定することをお勧めします。

## 6.7.3 DIVCR

分周比コントロールレジスタはシステムクロック用分周器の分周比を選択するレジスタです。分周器についてはH8/3069R F-ZTAT ハードウェアマニュアル 19.5 節をご参照ください。図 6-7にDIVCRの構成を、表 6-28に各ビットの機能を示します。

Bit7	Bit6	Bit5	bit4	bit3	bit2	Bit1	Bit0
-	-	-	-	-	-	DIV1	DIV0

図 6-7 分周比コントロールレジスタ(DIVCR)の構成

表 6-28 DIVCR のビット機能

ビット	ビット	機能
0	0	1/1(初期値)
0	1	1/2
1	0	1/4
1	1	1/8

初期値は0xFCで、HT1040では16MHzシステムクロックとなっています。CN1に接続する拡張ボードがこのシステムクロックに対応していない場合や、システムクロックを低くして消費電流を低く抑えたい場合に、設定比をかえることができます。

## 6.7.4 IER

IRQ イネーブルレジスタは、IRQ5 ~ IRQ0 の割り込み許可 / 禁止を制御するレジスタです。図 6-8 に IER の構成を、表 6-29 に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 5.2.4 項をご参照ください。

bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
-	-	IRQ5E	IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E

図 6-8 IRQ イネーブルレジスタ(IEE)の構成

表 6-29 IER のビット機能

IRQxE	機能
0	対応する IRQ の割り込みを禁止します。(初期値)
1	対応する IRQ の割り込みを許可します。

このレジスタの初期値は 0x00 で、すべての割り込みは禁止されています。  
 なお HT1040 では H8/3069 の IRQ0, IRQ4, IRQ5 を CN1 の外部割り込み入力に、IRQ1 を W5100 の割り込みに使用しています。また IRQ2 および IRQ3 に対応するピンの機能を /CS2, /CS1 として使用しているため、HT1040 ではこれらの外部割り込みは使用できません。

## 6.7.5 IPRA

インタラプトプライオリティレジスタ A は、各割り込み要求のプライオリティレベルを設定するレジスタです。図 6-9 に IPRA の構成を、表 6-30 に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 5.2.2 項をご参照ください。

bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
IPRA7	IPRA6	IPRA5	IPRA4	IPRA3	IPRA2	IPRA1	IPRA0

図 6-9 インタラプトプライオリティレジスタ(IPRA)の構成

表 6-30 IPRA のビット機能

IPRAx	機能
IPRA7	0: IRQ0 割り込み要求プライオリティレベルを 0 にします。(初期値) 1: IRQ0 割り込み要求プライオリティレベルを 1 にします。
IPRA6	0: IRQ1 割り込み要求プライオリティレベルを 0 にします。(初期値) 1: IRQ1 割り込み要求プライオリティレベルを 1 にします。
IPRA5	0: IRQ2/3 割り込み要求プライオリティレベルを 0 にします。(初期値) 1: IRQ2/3 割り込み要求プライオリティレベルを 1 にします。
IPRA4	0: IRQ4/5 割り込み要求プライオリティレベルを 0 にします。(初期値) 1: IRQ4/5 割り込み要求プライオリティレベルを 1 にします。
IPRA3	0: WDT, DRAM I/F, A/D 割り込み要求プライオリティレベルを 0 にします。(初期値) 1: WDT, DRAM I/F, A/D 割り込み要求プライオリティレベルを 1 にします。
IPRA2	0: 16bit タイマ CH0 割り込み要求プライオリティレベルを 0 にします。(初期値) 1: 16bit タイマ CH0 割り込み要求プライオリティレベルを 1 にします。
IPRA1	0: 16bit タイマ CH1 割り込み要求プライオリティレベルを 0 にします。(初期値) 1: 16bit タイマ CH1 割り込み要求プライオリティレベルを 1 にします。
IPRA0	0: 16bit タイマ CH2 割り込み要求プライオリティレベルを 0 にします。(初期値) 1: 16bit タイマ CH2 割り込み要求プライオリティレベルを 1 にします。

このレジスタの初期値は 0x00 で、すべての割り込みのプライオリティレベルは 0 になっています。

## 6.7.6 ISCR

IRQ センスコントロールレジスタは、IRQ5 ~ IRQ0 の割り込み入力レベルセンスまたは立下りエッジを選択するレジスタです。図 6-10に ISCR の構成を、表 6-31に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 5.2.5 項をご参照ください。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
-	-	IRQ5SC	IRQ4SC	IRQ3SC	IRQ2SC	IRQ1SC	IRQ0SC

図 6-10 IRQ センスコントロールレジスタ(ISCR)の構成

表 6-31 ISCR のビット機能

IRQXSC	機能
0	対応する IRQ の Low レベルで割り込み要求を発生します。(初期値)
1	対応する IRQ の立下りエッジで割り込み要求を発生します。

このレジスタの初期値は 0x00 で、すべての割り込みは Low レベル割り込みに設定されています。



HT1040 の外部バス割り込み入力インバータを通して H8/3069 に接続されているため、このレジスタの対応するビットに 0 が設定されている場合は High レベルで、1 が設定されている場合は立ち上がりエッジで割り込み要求が発生しますのでご注意ください。

## 6.7.7 ISR

IRQ ステータスレジスタは、IRQ5 ~ IRQ0 の割り込み要求ステータスを読み出すレジスタです。このレジスタへの書き込みにより、ステータスを 0 クリアすることができます。図 6-11に ISR の構成を、表 6-32に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 5.2.3 項をご参照ください。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
-	-	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F

図 6-11 IRQ ステータスレジスタ(ISR)の構成

表 6-32 ISR のビット機能

IRQxF	機能
0	IRQnF=1 の状態でこのレジスタを読み出し、IRQnF に 0 を書き込みすると該当するビットは 0 クリアされます。 レベル割り込みが設定されているとき、対応する割り込み入力 High レベルで割り込み例外処理を実行したとき 0 クリアされます。 エッジ割り込みが設定されているとき、対応する割り込み例外処理を実行すると 0 クリアされます。
1	外部割り込み発生条件が成立したとき 1 にセットされます。

このレジスタの初期値は 0x00 です。

## 6.7.8 P1DDR

ポート 1 データディレクションレジスタは、ポート 1 各端子の入出力をビット単位に指定するレジスタです。図 6-12に P1DDR の構成を、表 6-33に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 8.2.2 項をご参照ください。ポート 1[7:0]はアドレス A[7:0]と兼用されており、このレジスタで機能を選択します。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
P1 <sub>7</sub> DDR	P1 <sub>6</sub> DDR	P1 <sub>5</sub> DDR	P1 <sub>4</sub> DDR	P1 <sub>3</sub> DDR	P1 <sub>2</sub> DDR	P1 <sub>1</sub> DDR	P1 <sub>0</sub> DDR

図 6-12 ポート 1 データディレクションレジスタ(P1DDR)の構成

表 6-33 P1DDR のビット機能

P1xDDR	機能
0	対応するポート 1 の端子は入力ポートとなります。
1	モード 5 では、対応する端子をアドレス出力に設定します。モード 7 では出力ポートとなります。

H8/3069 動作モード 3 では初期値は 0xFF で、ポート 1 はアドレス出力として機能します。モード 3 ではこのレジスタ設定値を変更できません。モード 5 では初期値が 0x00 ですので、(外部バスが使用できるよう)0xFF を設定してください。

## 6.7.9 P2DDR

ポート 2 データディレクションレジスタは、ポート 2 各端子の入出力をビット単位に指定するレジスタです。図 6-13に P2DDR の構成を、表 6-34に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 8.3.2 項をご参照ください。ポート 2[7:0]はアドレス A[15:8]と兼用されており、このレジスタで機能を選択します。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
P2 <sub>7</sub> DDR	P2 <sub>6</sub> DDR	P2 <sub>5</sub> DDR	P2 <sub>4</sub> DDR	P2 <sub>3</sub> DDR	P2 <sub>2</sub> DDR	P2 <sub>1</sub> DDR	P2 <sub>0</sub> DDR

図 6-13 ポート 2 データディレクションレジスタ(P2DDR)の構成

表 6-34 P2DDR のビット機能

P2xDDR	機能
0	対応するポート 2 の端子は入力になります。
1	モード 5 では、対応する端子をアドレス出力に設定します。モード 7 では出力ポートとなります。

H8/3069 動作モード 3 では初期値は 0xFF で、ポート 2 はアドレス出力として機能します。モード 3 ではこのレジスタ設定値を変更できません。モード 5 では初期値が 0x00 ですので、(外部バスが使用できるよう)0xFF を設定してください。

## 6.7.10 P5DDR

ポート 5 データディレクションレジスタは、ポート 5 各端子の入出力をビット単位に指定するレジスタです。図 6-14に P5DDR の構成を、表 6-35に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 8.6.2 項をご参照ください。

い。ポート 5[3:0]はアドレス A[19:16]と兼用されており、このレジスタで機能を選択します。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
-	-	-	-	P5 <sub>3</sub> DDR	P5 <sub>2</sub> DDR	P5 <sub>1</sub> DDR	P5 <sub>0</sub> DDR

図 6-14 ポート 5 データディレクションレジスタ(P5DDR)の構成

表 6-35 P5DDR のビット機能

P5xDDR	機能
0	対応するポート 5 の端子は入力になります。
1	モード 5 では、対応する端子をアドレス出力に設定します。モード 7 では出力ポートとなります。

H8/3069 動作モード 3 では初期値は 0xFF で、ポート 5 はアドレス出力として機能します。モード 3 ではこのレジスタ設定値を変更できません。モード 5 では初期値が 0xF0 ですので、(外部バスが使用できるよう)0xFF を設定してください。

## 6.7.11 P8DDR

ポート 8 データディレクションレジスタは、ポート 8 各端子の入出力をビット単位に指定するレジスタです。図 6-15 に P8DDR の構成を、表 6-36 に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 8.9.2 項をご参照ください。ポート 8[4:1]は/CS[0:3]と兼用されており、このレジスタで機能を選択します。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
-	-	-	P8 <sub>4</sub> DDR	P8 <sub>3</sub> DDR	P8 <sub>2</sub> DDR	P8 <sub>1</sub> DDR	P8 <sub>0</sub> DDR

図 6-15 ポート 8 データディレクションレジスタ(P8DDR)の構成

表 6-36 P8DDR のビット機能

P8xDDR	機能
0	対応するポート 8 の端子は入力ポートとなります。
1	モード 3/5 では、対応する端子を/CS 出力に設定します。モード 7 では出力ポートとなります。 ポートと/CS の対応 P8 <sub>4</sub> /CS <sub>0</sub> P8 <sub>3</sub> /CS <sub>1</sub> P8 <sub>2</sub> /CS <sub>2</sub> P8 <sub>1</sub> /CS <sub>3</sub>

H8/3069 動作モード 3 での初期値は 0xF0(/CS<sub>0</sub>のみ有効)、モード 5 では初期値が 0xE0 ですので、HT1040 では 0xEC を設定して/CS<sub>1</sub> と/CS<sub>2</sub> を有効にしてください。

## 6.7.12 PBDDR

ポート B データディレクションレジスタは、ポート B 各端子の入出力をビット単位に指定するレジスタです。図 6-13 に PBDDR の構成を、表 6-34 に各ビットの機能を示します。ポート B[7:0]はリアルタイムクロック RX-4045SA および USB コントローラ VNC1L の制御に使用されています。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
PB7DDR	PB6DDR	PB5DDR	PB4DDR	PB3DDR	PB2DDR	PB1DDR	PB0DDR

図 6-16 ポート B データディレクションレジスタ(PBDDR)の構成

表 6-37 PBDDR のビット機能

PBxDDR	機能
0	対応するポート B の端子は入力になります。
1	対応するポート B の端子は出力になります。

H8/3069 動作モード 3/5 では初期値は 0x00 で、入力ポートとなっています。ポート B は H8/3069 の SCI 等他の I/O 機能と兼用端子のため、端子機能は PBDDR のほか、SCR2 や SMCR2 等の他のレジスタ設定によって決まります。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 8.12.2 項をご参照ください。HT1040 では RTC や USB コントローラのアクセスに SCI2 を使用するため、PBDDR には 0x33 を設定してください。

## 6.7.13 SYSCR

システムコントロールレジスタは、ソフトウェアスタンバイモード制御、CCR の UI ビット機能、NMI エッジ選択、H8/3069 内蔵 SRAM の有効/無効選択を行うレジスタです。図 6-17 に SYSCR の構成を、表 6-38 に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 3.3 節をご参照ください。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
SSBY	STS2	STS1	STS0	UE	NMIEG	SSOE	RAME

図 6-17 システムコントロールレジスタ(SYSCR)の構成

表 6-38 SYSCR のビット機能

ビット	機能
SSBY	0:SLEEP 命令実行でスリープモードに遷移(初期値) 1: SLEEP 命令実行でソフトウェアスタンバイモードへ遷移
STS[2:0]	スタンバイ解除の際の、内部クロック安定までの待機時間を選択します。 STS2 STS1 STS0 待機時間 0 0 0 8192 ステート(初期値) 0 0 1 16384 ステート 0 1 0 32768 ステート 0 1 1 65536 ステート 1 0 0 131072 ステート 1 0 1 262144 ステート 1 1 0 1024 ステート
UE	0:CCR の UI ビットを割り込みマスクビットとして使用 1:CCR の UI ビットをユーザービットとして使用(初期値)
NMIEG	0:NMI 端子の立下りエッジで割り込み要求を発生(初期値) 1:NMI 端子の立上がりエッジで割り込み要求を発生
SSOE	0:ソフトウェアスタンバイ時アドレス、バスコントロール信号は Hi-Z(初期値) 1:ソフトウェアスタンバイ時アドレス保持、バスコントロール信号は H
RAME	0:内蔵 RAM 無効 1:内蔵 RAM 有効(初期値)

SYSCR の初期値は 0x09 です。

## 6.7.14 WCRL

ウェイトコントロールレジスタは、各エリアのプログラムウェイト数を設定するレジスタです。図 6-18に WCRL の構成を、表 6-39に各ビットの機能を示します。詳細は H8/3069R F-ZTAT ハードウェアマニュアル 6.2.3 項をご参照ください。

bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
W31	W30	W21	W20	W11	W10	W01	W00

図 6-18 ウェイトコントロールレジスタ(WCRL)の構成

表 6-39 WCRL のビット機能

Wx1	Wx0	機能
0	0	ウェイトを挿入しない
0	1	1 ステートウェイト
1	0	2 ステートウェイト
1	1	3 ステートウェイト(初期値)

W31,W30 はエリア 3、W21,W20 はエリア 2、W11,W10 はエリア 1、W01,W00 はエリア 0 のウェイト数を設定します。ウェイト挿入機能はアクセスステート数が 3 ステートの場合(ASTCR の該当ビットが 1 の場合)にのみ有効です。

初期値は 0xFF で、全エリア 3 ウェイトの挿入となりますので、必要に応じて設定を変更してください。この設定は H8/3069 に外部で接続されるデバイスにのみ有効で、H8/3069 内蔵の I/O アクセスサイクル数は常に固定です。外部バスに増設するデバイスのアクセスタイムは、このレジスタおよび ASTCR、DIVCR の設定によって変わります。詳細は表 6-40をご参照ください。

表 6-40 デバイスアクセスタイム

SYSCLK	2 state access	3 state access			
		No wait	1 wait	2 wait	3 wait
8MHz	150ns	250ns	400ns	540ns	670ns
16MHz	40ns	100ns	150ns	200ns	250ns

## 6.8 外部割り込み

H8/3069 には IRQ0~5 と、NMI が外部割り込みとして用意されていますが、HT1040 では、IRQ0,IRQ4,IRQ5 と NMI が使用可能で、CN1 の IRQ4、IRQ5、IRQ7 の反転信号と/IOCHCHK がそれぞれ H8/3069 の IRQ4、IRQ5、IRQ0、NMI に接続されています。また、W5100 の割り込み出力は H8/3069 の IRQ1 にボード上で接続されています。

表 6-41に H8/3069 の割り込みとベクタ番号、割り込み要因等の関係を示します。

表 6-41 H8/3069 の割り込み要因とベクタナンバー

H8/3069	割り込み要因	ベクタ番号	ベクタアドレス	IPR	優先順位
NMI	CN1 /IOCHCHK	7	001C ~ 001F	-	高
IRQ0	CN1 IRQ7	12	0030 ~ 0033	IPRA7	
IRQ1	W5100 /INT	13	0034 ~ 0037	IPRA6	
IRQ4	CN1 IRQ4	16	0040 ~ 0043	IPRA4	
IRQ5	CN1 IRQ5	17	0044 ~ 0047	IPRA4	低

## 6.9 カレンダー時計

HT1040 にはカレンダー時計 (RTC) として EPSONTOYOCOM の RX-4045SA が搭載されています。デバイスの使用方法およびプログラミングの詳細については、RX-4045SA/NB アプリケーションマニュアルをご参照ください。カレンダー時計は電気二重層コンデンサで計時が 2 日程度は維持されますが、長時間の計時バックアップには電池を併用することができます。6.15 節をご参照ください。

### 6.9.1 H8/3069 との接続回路

RX-4045SA は H8/3069 の SIO チャンネル 2 およびポート B を通して接続されています。図 6-19 に RX-4045SA 周辺の回路を示します。

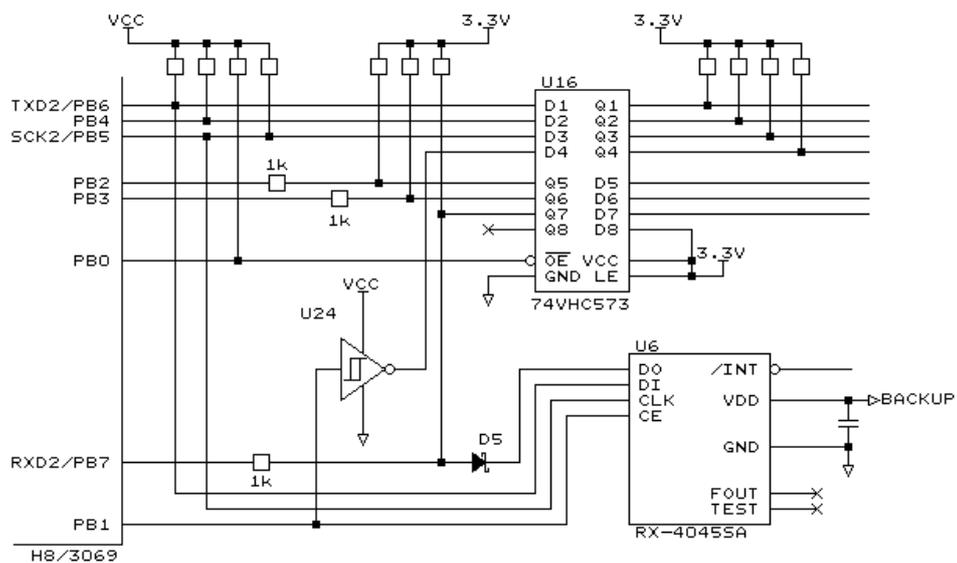


図 6-19 RX-4045SA 周辺回路

RTC は 4 線式シリアルインタフェースのため SCI チャンネル 2 の TXD2、RXD2、SCK2 およびポート B のビット 1(PB1)で接続されています。



《重要》SCI チャンネル 2 の送受信信号は USB コントローラと兼用のため、RTC をアクセスする際はポート B のビット 0(PB0)を H レベルとして USB コントローラとの信号接続を遮断してください。一方、USB コントローラをアクセスする場合は、PB0 を L レベルにし、RTC が同時にアクセスされないよう、ポート B のビット 1(PB1)を L レベルにしてください。

ポート B の各ビットの用途を表 6-42 に示します。この表では PB0 が H で RTC をアクセスする場合の機能を説明しています。PB0 が L の場合については表 6-59 をご参照ください。

表 6-42 PBDR のビット機能(PB0=H の場合)

PB	入出力	機能
PB0	出力	VNC1L と RX-4045SA のどちらを制御するか選択します。 0: VNC1L を制御 1:RX-4045SA を制御
PB1	出力	RX-4045SA の CE を制御します。 0:計時動作状態 1:RTC 内部レジスタアクセス許可状態
PB2	入力	1 が読み出されます。
PB3	入力	1 が読み出されます。
PB4	出力	出力値は RX-4045SA や VNC1L の動作に影響ありません。
PB5	出力	RX-4045SA の CLK 入力に接続されています。(SCI2 の SCK 出力)
PB6	出力	RX-4045SA の DI 入力に接続されています。(SCI2 の TXD 出力)
PB7	入力	RX-4045SA の DO 出力に接続されています。(SCI2 の RXD 入力)

## 6.9.2 内部レジスタマップ

表 6-43に RTC のレジスタ一覧を示します。各レジスタの詳細は RX-4045SA/NB アプリケーションマニュアルをご参照ください。

表 6-43 RTC 内レジスタアドレス

名称	R/W	アドレス	機能
Seconds	R/W	0	秒カウンタ
Minutes	R/W	1	分カウンタ
Hours	R/W	2	時カウンタ
Weekdays	R/W	3	曜日カウンタ
Days	R/W	4	日カウンタ
Months	R/W	5	月カウンタ
Years	R/W	6	年カウンタ
DigitalOffset	W	7	時計精度調整レジスタ
Alarm_W;Minute	W	8	アラーム W 分桁設定レジスタ
Alarm_W;Hour	W	9	アラーム W 時桁設定レジスタ
Alarm_W;Weekday	W	A	アラーム W 曜日設定レジスタ
Alarm_D;Minute	R/W	B	アラーム D 分桁設定レジスタ
Alarm_D;Hour	R/W	C	アラーム D 時桁設定レジスタ
Reserved	R/W	D	予約
Control1	R/W	E	制御レジスタ 1
Control2	R/W	F	制御レジスタ 2

## 6.10 W5100

W5100 は、ハードウェアで TCP/IP 処理を実現したデバイスです。HT1040 ではアドレス 240000 ~ 247FFF を占有します。W5100 の内部レジスタ・バッファメモリのアドレスマップを図 6-20に示します。

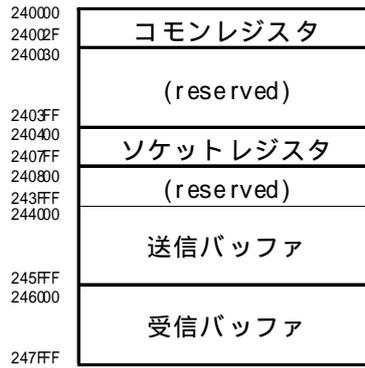


図 6-20 W5100 レジスタ・バッファメモリマップ

## 6.10.1 コモンレジスタ

表 6-44にコモンレジスタのアドレスとレジスタ名を示します。

表 6-44 W5100 コモンレジスタ

名称	R/W	アドレス	機能	リセット時
MR	R/W	240000	モードレジスタ	00
GAR0	R/W	240001	ゲートウェイ IP アドレスレジスタ	00
GAR1	R/W	240002	例 192.168.0.1	00
GAR2	R/W	240003	GAR0 GAR1 GAR2 GAR3	00
GAR3	R/W	240004	192(0xC0) 168(0xA8) 0(0x00) 1(0x01)	00
SUBR0	R/W	240005	サブネットマスクレジスタ	00
SUBR1	R/W	240006	例 255.255.255.0	00
SUBR2	R/W	240007	SUBR0 SUBR1 SUBR2 SUBR3	00
SUBR3	R/W	240008	255(0xFF) 255(0xFF) 255(0xFF) 0(0x00)	00
SHAR0	R/W	240009	ソースハードウェアアドレスレジスタ	00
SHAR1	R/W	24000A	例 00:04:14:10:12:34	00
SHAR2	R/W	24000B	SHAR 0 SHAR 1 SHAR 2 SHAR 3 SHAR 4 SHAR 5	00
SHAR3	R/W	24000C	0 1 2 3 4 5	00
SHAR4	R/W	24000D	0x00 0x04 0x14 0x10 0x12 0x34	00
SHAR5	R/W	24000E		00
SIPRO	R/W	24000F	ソース IP アドレスレジスタ	00
SIPR1	R/W	240010	例 192.168.0.11	00
SIPR2	R/W	240011	SIPR0 SIPR1 SIPR2 SIPR3	00
SIPR3	R/W	240012	192(0xC0) 168(0xA8) 0(0x00) 11(0x0B)	00
IR	R/W	240015	インタラプトレジスタ	00
IMR	R/W	240016	インタラプトマスクレジスタ	00
RTR0	R/W	240017	リトライタイムレジスタ	07
RTR1	R/W	240018		D0
RCR	R/W	240019	リトライカウントレジスタ	08
RMSR	R/W	24001A	Rx メモリサイズレジスタ	55
TMSR	R/W	24001B	Tx メモリサイズレジスタ	55
PATRO	R	24001C	PPPoE 認証タイプレジスタ	00
PATR1	R	24001D		00
PTIMER	R/W	240028	PPP LCP リクエストタイムレジスタ	28
PMAGIC	R/W	240029	PPP LCP マジックナンバーレジスタ	00
UIPRO	R	24002A	アンリーチャブル IP アドレスレジスタ	00
UIPR1	R	24002B	例 192.168.0.20	00
UIPR2	R	24002C	UIPR0 UIPR1 UIPR2 UIPR3	00
UIPR3	R	24002D	192(0xC0) 168(0xA8) 0(0x00) 20(0x14)	00
UPORT0	R	24002E	アンリーチャブルポートレジスタ	00
UPORT1	R	24002F		00

## 6.10.2 モードレジスタ(MR)

図 6-21に MR の構成を、表 6-45に各ビットの機能を示します。リセット後 MR の設定内容は 0x00 となります。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
RST	-	-	PB	PPPoE	-	AI	IND

図 6-21 モードレジスタ(MR)の構成

表 6-45 MR のビット機能

ビット	機能
RST	<b>ソフトウェアリセット</b> このビットに 1 を書き込みすると、内部レジスタが初期化されます。その後このビットは自動的に 0 になります。
PB	<b>Ping ブロックモード</b> 0:Ping に応答します。(初期値) 1:Ping に応答しません。
PPPoE	<b>PPPoE モード</b> 0:非 PPPoE モード(初期値) 1:PPPoE モード(ルータを使用せず ADSL モデムで接続する場合(初期値))
AI	<b>アドレスオートインクリメント(インダイレクトバス I/F モード)</b> 0:オートインクリメント禁止(初期値) 1:オートインクリメント有効 HT1040 では W5100 をダイレクトバスインタフェースモードで接続しているため、本ビットの設定内容は動作に影響しません。
IND	<b>バスインタフェースモード</b> 0:ダイレクトバスインタフェースモード(初期値) 1:インダイレクトバスインタフェースモード HT1040 では W5100 をダイレクトバスインタフェースモードで接続しているため、本ビットは 0 のまま使用してください。

## 6.10.3 ゲートウェイアドレスレジスタ(GAR)

このレジスタにはゲートウェイアドレスを設定します。GWR は 4 バイト長で、例えばゲートウェイアドレスが 192.168.0.1 の場合、図 6-22のように設定します。

GAR0	GAR1	GAR2	GAR3
192(0xC0)	168(0xA8)	0(0x00)	1(0x01)

図 6-22 ゲートウェイアドレスレジスタ(GAR)の構成

## 6.10.4 サブネットマスクレジスタ(SUBR)

このレジスタにはサブネットマスクを設定します。SUBR は 4 バイト長で、例えばサブネットマスクが 255.255.255.0 の場合、図 6-23のように設定します。

SUBR0	SUBR1	SUBR2	SUBR3
255(0xFF)	255(0xFF)	255(0xFF)	0(0x00)

図 6-23 サブネットマスクレジスタ(SUBR)の設定例

## 6.10.5 ソースハードウェアアドレスレジスタ(SHAR)

このレジスタにはソースハードウェアアドレス(MAC アドレス)を設定します。SHAR は 6 バイト長で、例えばソースハードウェアアドレスが 00:04:14:10:12:34 の場合、

SHAR0	SHAR1	SHAR2	SHAR3	SHAR3	SHAR3
0x00	0x04	0x14	0x10	0x12	0x34

図 6-24 ソースハードウェアアドレスレジスタ(SHAR)の設定例

のように設定します。HT1040 ではユーザブートマットに書き込まれている MAC アドレスをこのレジスタに設定してください。MAC アドレスの読み出し方法は7.2節をご参照ください。

## 6.10.6 ソース IP アドレスレジスタ(SIPR)

このレジスタにはソース IP アドレスを設定します。SIPR は 4 バイト長で、例えばソース IP アドレスが 192.168.0.11 の場合、

SIPR0	SIPR1	SIPR2	SIPR3
192(0xC0)	168(0xA8)	0(0x00)	11(0x0B)

図 6-25 ソース IP アドレスレジスタ(SIPR)の設定例

のように設定します。

## 6.10.7 インタラプトレジスタ(IR)

このレジスタは CPU が W5100 の発生した割り込み要因を調べるために使用します。割り込みはインタラプトマスクレジスタ(IMR)でマスクすることができます。W5100 の割り込み出力はこのレジスタの各ビットへの 0 書き込みでクリアするまでアクティブとなります。図 6-26 に IR の構成を、表 6-46 に各ビットの機能を示します。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
CONFLICT	UNREACH	PPPoE	-	S3_INT	S2_INT	S1_INT	S0_INT

図 6-26 インタラプトレジスタ(IR)の構成

表 6-46 IR のビット機能

ビット	機能
CONFLICT	<b>IP コンフリクト</b> W5100 に設定されているソース IP アドレスと同じ IP アドレスの ARP リクエストを検出した場合に 1 がセットされます。この割り込みをクリアするためには、このビットに 1 を書き込みしてください。
UNREACH	<b>到達不能</b> W5100 が ICMP(到達不能)パケットを受信するとこのビットに 1 がセットされます。このときの IP アドレスとポート番号がそれぞれ UIPR レジスタと UPORT レジスタに保存されます。この割り込みをクリアするためには、このビットに 1 を書き込みしてください。
PPPoE	<b>PPPoE コネクションクローズ</b> PPPoE モードでコネクションがクローズされるとこのビットに 1 がセットされます。この割り込みをクリアするためには、このビットに 1 を書き込みしてください。
S3_INT	<b>ソケット 3 割り込みの発生</b> このビットが 1 にセットされた場合は、ソケット 3 に関連する割り込みが発生したことを示します。S3_IR レジスタでソケット割り込みの詳細を確認することができます。この割り込みをクリアするためには、このビットではなく S3_IR レジスタの割り込み要因ビットに 1 を書き込みして S3_IR レジスタをクリアしてください。
S2_INT	<b>ソケット 2 割り込みの発生</b> このビットが 1 にセットされた場合は、ソケット 2 に関連する割り込みが発生したことを示します。S2_IR レジスタでソケット割り込みの詳細を確認することができます。この割り込みをクリアするためには、このビットではなく S2_IR レジスタの割り込み要因ビットに 1 を書き込みして S2_IR レジスタをクリアしてください。
S1_INT	<b>ソケット 1 割り込みの発生</b> このビットが 1 にセットされた場合は、ソケット 1 に関連する割り込みが発生したことを示します。S1_IR レジスタでソケット割り込みの詳細を確認することができます。この割り込みをクリアするためには、このビットではなく S1_IR レジスタの割り込み要因ビットに 1 を書き込みして S1_IR レジスタをクリアしてください。
S0_INT	<b>ソケット 0 割り込みの発生</b> このビットが 1 にセットされた場合は、ソケット 0 に関連する割り込みが発生したことを示します。S0_IR レジスタでソケット割り込みの詳細を確認することができます。この割り込みをクリアするためには、このビットではなく S0_IR レジスタの割り込み要因ビットに 1 を書き込みして S0_IR レジスタをクリアしてください。



《注意》

W5100 に設定した IP と同じ IP アドレスの存在を確認するために ARP を送信すると IP コンフリクト割り込みが発生しますが、この場合実際には IP のコンフリクトは存在しません。

## 6.10.8 インタラプトマスクレジスタ(IMR)

このレジスタは W5100 が発生する割り込み要因をマスクするために使用します。インタラプトマスクレジスタ(IMR)の各ビットはインタラプトレジスタ(IR)の各ビットに対応しています。0 を設定したビットの割り込みは発生しません。1 を設定したビットの割り込みは有効となり条件が成立すると割り込みが発生します。図 6-27に

IMR の構成を、表 6-47に各ビットの機能を示します。リセット後 IMR は 0x00 となり、全ての割り込みは禁止されます。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
IM_IR7	IM_IR6	IM_IR5	Reserved	IM_IR3	IM_IR2	IM_IR1	IM_IR0

図 6-27 インタラプトレジスタ(IMR)の構成

表 6-47 IMR のビット機能

ビット	機能	
IMR_7	IP コンフリクト割り込み	0:禁止 1:許可
IMR_6	到達不能割り込み	0:禁止 1:許可
IMR_5	PPPoE コネクションクローズ割り込み	0:禁止 1:許可
Reserved	このビットには 0 を設定してください。	
IMR_3	ソケット 3 割り込み	0:禁止 1:許可
IMR_2	ソケット 2 割り込み	0:禁止 1:許可
IMR_1	ソケット 1 割り込み	0:禁止 1:許可
IMR_0	ソケット 0 割り込み	0:禁止 1:許可

## 6.10.9 リトライタイムアウトレジスタ(RTR)

RTR にはリトライのタイムアウト時間を設定します。設定値に 100uS をかけた値が実際のタイムアウト時間となります。例えば

RTR0	RTR1
0x07	0xD0

図 6-28 リトライタイムアウトレジスタ(RTR)の設定例

の場合、 $0x07D0=2000$  ですので  $2000 \times 100\mu s=200ms$  がタイムアウト時間となります。このレジスタはリセット時 0x07D0 に設定されます。

## 6.10.10 リトライカウントレジスタ(RCR)

RCR には再送回数を設定します。再送回数がこのレジスタの設定値を超えた場合はタイムアウト割り込みが発生します。(Sn\_IR レジスタのタイムアウト割り込みビットが 1 となります。)

## 6.10.11 Rx メモリサイズレジスタ(RMSR)

RMSR には各ソケットに割り当てる受信バッファのメモリサイズを設定します。

図 6-29に RMSR の構成を、表 6-48に各ビットの機能を示します。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
S3_MS1	S3_MS0	S2_MS1	S2_MS0	S1_MS1	S1_MS0	S0_MS1	S0_MS0

図 6-29 Rx メモリサイズレジスタ(RMSR)の構成

表 6-48 RMSR のビット機能

Sn_MS1	Sn_MS0	メモリサイズ
0	0	1KB
0	1	2KB
1	0	4KB
1	1	8KB

RMSR はリセット時に 0x55 が設定され、各ソケットには 2KB が割り当てられます。

《注意》

W5100 には 8KB の受信バッファメモリが用意されています。RMSR に対し、受信バッファメモリの合計が 8KB 以上となるような設定も可能ですが、受信メモリはソケット 0 から順に 8KB まで割り当てられ、それ以上のメモリが必要なソケットは使用できませんのでご注意ください。



## 6.10.12 Tx メモリサイズレジスタ(TMSR)

TMSR には各ソケットに割り当てる送信バッファのメモリサイズを設定します。設定方法は RMSR と同じです。TMSR はリセット時に 0x55 が設定され、各ソケットには 2KB が割り当てられます。

## 6.10.13 PPPoE 認証タイプレジスタ(PATR)

PATR には PPPoE サーバとの接続で使用する認証方式を設定します。設定可能な値と認証方式は表 6-49をご参照ください。

表 6-49 PATR 設定値と認証方式

PATR0	PATR1	認証方式
0xC0	0x23	PAP
0xC2	0x23	CHAP

## 6.10.14 PPP LCP リクエストタイマ(PTIMER)

PTIMER には LCP エコー要求を送信する間隔を指定します。設定値に 25ms をかけた値が実際の時間間隔となります。例えば 0x28 を設定した場合は、 $0x28=40$  ですので  $40 \times 25ms=1s$  が時間間隔となります。このレジスタはリセット時 0x28 が設定されます。

## 6.10.15 PPP LCP マジックナンバーレジスタ(PMAGIC)

PMAGIC には LCP ネゴシエーションのマジックナンバー設定に使用されます。詳細は Wiznet 発行のアプリケーションノート "How to connect ADSL" をご参照ください。

## 6.10.16 アンリーチャブルIP アドレスレジスタ(UIPR)

ICMP メッセージでデスティネーションアンリーチャブル応答を受信すると、UIPR レジスタにその IP アドレスを保存します。UIPR は 4 バイト長で、例えば IP アドレスが 192.168.0.20 の場合、

UIPR0	UIPR1	UIPR2	UIPR3
192(0xC0)	168(0xA8)	0(0x00)	20(0x14)

図 6-30 アンリーチャブル IP アドレスレジスタ(UIPR)の内容例

のように保存されます。

## 6.10.17 アンリーチャブルポートレジスタ(UPORT)

ICMP メッセージでデスティネーションアンリーチャブル応答を受信すると、UPOINT レジスタにそのポート番号を保存します。UPOINT は 2 バイト長で、例えばポート番号が 123(=0x7B)の場合、

UPOINT0	UPOINT1
0x00	0x7B

図 6-31 アンリーチャブルポートレジスタ(UPOINT)の内容例

のように保存されます。

## 6.10.18 ソケットレジスタ

表 6-50からにソケットレジスタのアドレスとレジスタ名、機能等を示します。使用可能なソケットは 0 から 3 まで 4 つあり、それぞれのソケットに独立のレジスタが用意されています。以下の説明では、ソケット 0 から 3 までの同機能のレジスタをまとめて Sn\_xx のように表記します。(例えばソケット 0 から 3 のモードレジスタは Sn\_MR と表記します。)

表 6-50 W5100 ソケット 0 レジスタ

名称	R/W	アドレス	機能	リセット時
SO_MR	R/W	240400	ソケット 0 モードレジスタ	00
SO_CR	R/W	240401	ソケット 0 コマンドレジスタ	00
SO_IR	R/W	240402	ソケット 0 インタラプトレジスタ	00
SO_SR	R	240403	ソケット 0 ステータスレジスタ	00
SO_PORT0	R/W	240404	ソケット 0 ソースポートレジスタ	00
SO_PORT1	R/W	240405		00
SO_DHAR0	R/W	240406	ソケット 0 デスティネーションハードウェアアドレスレジスタ 例 00:04:14:10:12:34	00
SO_DHAR1	R/W	240407		00
SO_DHAR2	R/W	240408	DHAR 0	00
SO_DHAR3	R/W	240409	DHAR 1	00
SO_DHAR4	R/W	24040A	DHAR 2	00
SO_DHAR5	R/W	24040B	DHAR 3	00
SO_DIPO0	R/W	24040C	DHAR 4	00
SO_DIPO1	R/W	24040D	DHAR 5	00
SO_DIPO2	R/W	24040E	0x00	00
SO_DIPO3	R/W	24040F	0x04	00
SO_DPORT0	R/W	240410	0x14	00
SO_DPORT1	R/W	240411	0x10	00
SO_MSSR0	R/W	240412	0x12	00
SO_MSSR1	R/W	240413	0x34	00
SO_PROTO	R/W	240414	ソケット 0 デスティネーション IP アドレスレジスタ 例 192.168.0.11	00
SO_TOS	R/W	240415	DIPR0	00
SO_TTL	R/W	240416	DIPR1	00
SO_TX_FSR0	R/W	240420	DIPR2	00
SO_TX_FSR1	R/W	240421	DIPR3	00
SO_TX_RD0	R/W	240422	192(0xC0)	07
SO_TX_RD1	R/W	240423	168(0xA8)	D0
SO_TX_WR0	R/W	240424	0(0x00)	07
SO_TX_WR1	R/W	240425	11(0x0B)	D0
SO_RX_RSR0	R/W	240426	ソケット 0 Tx リードポインタレジスタ	07
SO_RX_RSR1	R/W	240427	ソケット 0 Tx ライトポインタレジスタ	D0
SO_RX_RD0	R	240428	ソケット 0 Rx 受信サイズレジスタ	07
SO_RX_RD1	R	240429	ソケット 0 Rx リードポインタレジスタ	D0
				00

表 6-51 W5100 ソケット 1 レジスタ

名称	R/W	アドレス	機能	リセット時
S1_MR	R/W	240500	ソケット 1 モードレジスタ	00
S1_CR	R/W	240501	ソケット 1 コマンドレジスタ	00
S1_IR	R/W	240502	ソケット 1 インタラプトレジスタ	00
S1_SR	R	240503	ソケット 1 ステータスレジスタ	00
S1_PORT0	R/W	240504	ソケット 1 ソースポートレジスタ	00
S1_PORT1	R/W	240505		00
S1_DHAR0	R/W	240506	ソケット1 デスティネーションハードウェアアドレスレジスタ 例 00:04:14:10:12:34	00
S1_DHAR1	R/W	240507		00
S1_DHAR2	R/W	240508	DHAR 0	00
S1_DHAR3	R/W	240509	DHAR 1	00
S1_DHAR4	R/W	24050A	DHAR 2	00
S1_DHAR5	R/W	24050B	DHAR 3	00
S1_DIPR0	R/W	24050C	DHAR 4	00
S1_DIPR1	R/W	24050D	DHAR 5	00
S1_DIPR2	R/W	24050E	0x00	00
S1_DIPR3	R/W	24050F	0x04	00
S1_DPORT0	R/W	240510	0x14	00
S1_DPORT1	R/W	240511	0x10	00
S1_MSSR0	R/W	240512	0x12	00
S1_MSSR1	R/W	240513	0x34	00
S1_PROTO	R/W	240514	ソケット 0 デスティネーション IP アドレスレジスタ 例 192.168.0.11	00
S1_TOS	R/W	240515	DIPR0	00
S1_TTL	R/W	240516	DIPR1	00
S1_TX_FSR0	R/W	240520	DIPR2	00
S1_TX_FSR1	R/W	240521	DIPR3	00
S1_TX_RD0	R/W	240522	192(0xC0)	00
S1_TX_RD1	R/W	240523	168(0xA8)	00
S1_TX_WR0	R/W	240524	0(0x00)	00
S1_TX_WR1	R/W	240525	11(0x0B)	00
S1_RX_RSRO	R/W	240526	ソケット 1 デスティネーションポートレジスタ	00
S1_RX_RSR1	R/W	240527	ソケット 1 最大セグメントサイズレジスタ	00
S1_RX_RDO	R	240528	ソケット 1 プロトコルレジスタ	00
S1_RX_RD1	R	240529	ソケット 1 TOS レジスタ	00
			ソケット 1 TTL レジスタ	00
			ソケット 1 Tx フリーサイズレジスタ	07
			ソケット 1 Tx リードポインタレジスタ	D0
			ソケット 1 Tx ライトポインタレジスタ	07
			ソケット 1 Rx 受信サイズレジスタ	D0
			ソケット 1 Rx リードポインタレジスタ	07
			ソケット 1 Rx ライトポインタレジスタ	D0

表 6-52 W5100 ソケット 2 レジスタ

名称	R/W	アドレス	機能	リセット時
S2_MR	R/W	240600	ソケット 2 モードレジスタ	00
S2_CR	R/W	240601	ソケット 2 コマンドレジスタ	00
S2_IR	R/W	240602	ソケット 2 インタラプトレジスタ	00
S2_SR	R	240603	ソケット 2 ステータスレジスタ	00
S2_PORT0	R/W	240604	ソケット 2 ソースポートレジスタ	00
S2_PORT1	R/W	240605		00
S2_DHAR0	R/W	240606	ソケット 2 デスティネーションハードウェアアドレスレジスタ 例 00:04:14:10:12:34	00
S2_DHAR1	R/W	240607		00
S2_DHAR2	R/W	240608	DHAR 0	00
S2_DHAR3	R/W	240609	DHAR 1	00
S2_DHAR4	R/W	24060A	DHAR 2	00
S2_DHAR5	R/W	24060B	DHAR 3	00
S2_DHAR6	R/W	24060C	DHAR 4	00
S2_DHAR7	R/W	24060D	DHAR 5	00
S2_DIPR0	R/W	24060E	ソケット 2 デスティネーション IP アドレスレジスタ 例 192.168.0.11	00
S2_DIPR1	R/W	24060F		00
S2_DIPR2	R/W	240610	DIPR0	00
S2_DIPR3	R/W	240611	DIPR1	00
S2_DIPR4	R/W	240612	DIPR2	00
S2_DIPR5	R/W	240613	DIPR3	00
S2_DPORT0	R/W	240614	ソケット 2 デスティネーションポートレジスタ	00
S2_DPORT1	R/W	240615		00
S2_MSSR0	R/W	240616	ソケット 2 最大セグメントサイズレジスタ	00
S2_MSSR1	R/W	240617		00
S2_PROTO	R/W	240618	ソケット 2 プロトコルレジスタ	00
S2_TOS	R/W	240619	ソケット 2 TOS レジスタ	00
S2_TTL	R/W	24061A	ソケット 2 TTL レジスタ	00
S2_TX_FSR0	R/W	240620	ソケット 2 Tx フリーサイズレジスタ	07
S2_TX_FSR1	R/W	240621		D0
S2_TX_RD0	R/W	240622	ソケット 2 Tx リードポインタレジスタ	07
S2_TX_RD1	R/W	240623		D0
S2_TX_WR0	R/W	240624	ソケット 2 Tx ライトポインタレジスタ	07
S2_TX_WR1	R/W	240625		D0
S2_RX_RSR0	R/W	240626	ソケット 2 Rx 受信サイズレジスタ	07
S2_RX_RSR1	R/W	240627		D0
S2_RX_RD0	R	240628	ソケット 2 Rx リードポインタレジスタ	00
S2_RX_RD1	R	240629		00

表 6-53 W5100 ソケット 3 レジスタ

名称	R/W	アドレス	機能	リセット時
S3_MR	R/W	240700	ソケット 3 モードレジスタ	00
S3_CR	R/W	240701	ソケット 3 コマンドレジスタ	00
S3_IR	R/W	240702	ソケット 3 インタラプトレジスタ	00
S3_SR	R	240703	ソケット 3 ステータスレジスタ	00
S3_PORT0	R/W	240704	ソケット 3 ソースポートレジスタ	00
S3_PORT1	R/W	240705		00
S3_DHAR0	R/W	240706	ソケット 3 デスティネーションハードウェアアドレスレジスタ 例 00:04:14:10:12:34	00
S3_DHAR1	R/W	240707		00
S3_DHAR2	R/W	240708	DHAR 0	00
S3_DHAR3	R/W	240709	DHAR 1	00
S3_DHAR4	R/W	24070A	DHAR 2	00
S3_DHAR5	R/W	24070B	DHAR 3	00
S3_DIPRO	R/W	24070C	ソケット 3 デスティネーション IP アドレスレジスタ 例 192.168.0.11	00
S3_DIPR1	R/W	24070D		00
S3_DIPR2	R/W	24070E	DIPR0	00
S3_DIPR3	R/W	24070F	DIPR1	00
S3_DPORT0	R/W	240710	ソケット 3 デスティネーションポートレジスタ	00
S3_DPORT1	R/W	240711		00
S3_MSSR0	R/W	240712	ソケット 3 最大セグメントサイズレジスタ	00
S3_MSSR1	R/W	240713		00
S3_PROT0	R/W	240714	ソケット 3 プロトコルレジスタ	00
S3_TOS	R/W	240715	ソケット 3 TOS レジスタ	00
S3_TTL	R/W	240716	ソケット 3 TTL レジスタ	00
S3_TX_FSR0	R/W	240720	ソケット 3 Tx フリーサイズレジスタ	07
S3_TX_FSR1	R/W	240721		D0
S3_TX_RD0	R/W	240722	ソケット 3 Tx リードポインタレジスタ	07
S3_TX_RD1	R/W	240723		D0
S3_TX_WR0	R/W	240724	ソケット 3 Tx ライトポインタレジスタ	07
S3_TX_WR1	R/W	240725		D0
S3_RX_RSR0	R/W	240726	ソケット 3 Rx 受信サイズレジスタ	07
S3_RX_RSR1	R/W	240727		D0
S3_RX_RD0	R	240728	ソケット 3 Rx リードポインタレジスタ	00
S3_RX_RD1	R	240729		00

## 6.10.19 ソケット n モードレジスタ(Sn\_MR)

図 6-32に Sn\_MR の構成を、表 6-54に各ビットの機能を示します。リセット後 Sn\_MR の設定内容は 0x00 となります。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
MULTI	-	ND/MC	-	P3	P2	P1	P0

図 6-32 ソケット n モードレジスタ(Sn\_MR)の構成

表 6-54 Sn\_MR のビット機能

ビット	機能				
MULTI	<b>マルチキャスト</b> 0:禁止(初期値) 1:有効 UDP の場合このビットに 1 を書き込みすると、マルチキャストが有効になります。OPEN コマンドを実行する前に、マルチキャストのグループアドレスを Sn_DIPR に、グループポート番号を Sn_DPORT に設定してください。				
ND/MC	MULTI=0、TCP の場合 <b>遅延 ACK</b> 0:遅延 ACK 有効(初期値) 1:遅延 ACK 無効	MULTI=1 の場合 <b>マルチキャスト</b> 0: IGMP version2(初期値) 1: IGMP version1			
P[3:0]	<b>Protocol</b> ソケットを使用するモードを設定します。				
	P3	P2	P1	P0	モード
	0	0	0	0	クローズ
	0	0	0	1	TCP
	0	0	1	0	UDP
	0	0	1	1	IPRAW
	0	1	0	0	MACRAW(ソケット 0 のみ)
	0	1	0	1	PPPoE(ソケット 0 のみ)

## 6.10.20 ソケット n コマンドレジスタ(Sn\_CR)

Sn\_CR にはコマンドを書き込み、ソケットの動作を制御します。コマンド実行後 Sn\_CR の内容は自動的に 0x00 となります。

表 6-55 Sn\_CR に書き込みするコマンドと動作

書き込み値	コマンド	動作内容
0x01	OPEN	ソケットを初期化します。Sn_MR に設定されたプロトコルモードによって、ソケットステータスは SOCK_INIT, SOCK_UDP, SOCK_IPRAW, SOCK_MACRAW に変化します。
0x02	LISTEN	TCP プロトコルモードの場合にのみ使用します。ソケットステータスを SOCK_LISTEN にし、TCP クライアントからの接続状態待ち状態とします。
0x04	CONNECT	TCP プロトコルモードの場合にのみ使用します。TCP サーバに対して接続要求を送信します。接続に失敗した場合は、タイムアウト割り込みが発生します。
0x08	DISCON	TCP プロトコルモードの場合にのみ使用します。切断要求を送信します。切断に失敗した場合は、タイムアウト割り込みが発生します。
0x10	CLOSE	ソケットをクローズします。ソケットステータスは SOCK_CLOSED になります。このコマンドはコネクションの切断は行いませんのでご注意ください。
0x20	SEND	送信済みの Tx バッファの次のアドレスから、現在の Tx 書き込みポインタ-1 のアドレスまでのデータを送信します。

0x21	SEND_MAC	UDP プロトコルモードで使用します。送信済みの Tx バッファの次のアドレスから、現在の Tx 書き込みポインタ-1 のアドレスまでのデータを送信します。このコマンドの場合、送信先のハードウェアアドレスは ARP プロトコルで取得せず、Sn_DHAR レジスタに指定されている値を使用します。
0x22	SEND_KEEP	TCP プロトコルモードの場合にのみ使用します。コネクション状態をチェックするため、1 バイトのデータを送信します。コネクションが切断されていたり、応答がない場合はタイムアウト割り込みが発生します。
0x40	RECV	Rx リードポインタレジスタで受信します。

## 6.10.21 ソケット n インタラプトレジスタ(Sn\_IR)

図 6-33に Sn\_IR の構成を、表 6-56に各ビットの機能を示します。リセット後 Sn\_IR は 0x00 となります。各ビットは 1 を書き込みしてクリアするまでその内容が保持されます。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
-	-	-	SEND_OK	TIMEOUT	RECV	DISCON	CON

図 6-33 ソケット n モードレジスタ(Sn\_IR)の構成

表 6-56 Sn\_IR のビット機能

ビット	機能
SEND_OK	SEND 動作が完了した場合に 1 にセットされます。
TIMEOUT	コネクション確立・切断中およびデータ送信中にタイムアウトが発生すると 1 にセットされます。
RECV	データ受信時、および RECV コマンド実行後受信データがある場合、1 にセットされます。
DISCON	コネクション終了要求または終了した場合 1 にセットされます。
CON	コネクションが確立すると 1 にセットされます。

## 6.10.22 ソケット n ステータスレジスタ(Sn\_SR)

ソケット n のステータスは Sn\_SR を読み出して確認することができます。表 6-57にステータス一覧を示します。

表 6-57 Sn\_SR のステータス

読み出し値	記号	説明
0x00	SOCK_CLOSED	CLOSE コマンドが実行され、タイムアウト割り込みが発生したか、コネクションが切断されたことを示します。この状態では、コネクションのリソースはすべて開放された状態となります。
0x13	SOCK_INIT	TCP プロトコルモードで OPEN コマンドが実行された状態です。このステータスはソケットの TCP 接続確立の最初のステップです。動作タイプが TCP サーバモードかクライアントモードかはコマンドが LISTEN か CONNECT かで決まります。
0x14	SOCK_LISTEN	SOCK_INIT 状態で LISTEN コマンドを実行した状態です。ソケットは TCP サーバモードで動作し、コネクション要求を受けると ESTABLISHED 状態に移行します。
0x17	SOCK_ESTABLISHED	コネクションが確立したことを示します。TCP データの送受信ができる状態です。
0x1C	SOCK_CLOSE_WAIT	相手側から接続終了要求を受信したことを示します。この状態では相手側から ACK を受信していますがまだ切断とはなっていません。コネクションは DISCON か CLOSE コマンドの実行でクローズされます。
0x22	SOCK_UDP	UDP プロトコルモードで OPEN コマンドが実行されたことを示します。UDP ではピアとの接続は不要で、データは直接送受信します。
0x32	SOCK_IPRAW	IPRAW プロトコルモードで OPEN コマンドが実行されたことを示します。IPRAW モードでは IP ヘッダは処理されません。
0x42	SOCK_MACRAW	MACRAW プロトコルモードで OPEN コマンドが実行されたことを示します。MACRAW モードではプロトコル処理は行いません。
0x5F	SOCK_PPPOE	PPPOE プロトコルモードで OPEN コマンドが実行されたことを示します。
0x15	SOCK_SYNSENT	SOCK_INIT 状態で CONNECT コマンドが実行された状態です。接続が確立すると SOCK_ESTABLISHED 状態へ移行します。
0x16	SOCK_SYNRECV	ピアから接続要求を受信したことを示します。通常リクエストに応答し、SOCK_ESTABLISHED 状態へ移行します。
0x18	SOCK_FIN_WAIT	コネクションの終了処理中であることを示します。
0x1A	SOCK_CLOSING	切断が正常に処理されるか、タイムアウト割り込みが発生した場合は SOCK_CLOSED 状態へ移行します。
0x1B	SOCK_TIME_WAIT	
0x1D	SOCK_LAST_ACK	
0x01	SOCK_ARP	リモートピアのハードウェアアドレス取得のために ARP 要求が送られたことを示します。ARP 応答を受信すると SOCK_SYNSENT, SOCK_UDP, SOCK_IPRAW に移行します。

## 6.10.23 ソケット n ソースポートレジスタ(Sn\_PORT)

Sn\_PORT は TCP/UDP プロトコルモードで各ソケットのソースポート番号を設定するレジスタです。OPEN コマンドを実行する前に設定してください。

Sn\_PORT は 2 バイト長で、例えばポート番号が 5000(=0x1388)の場合、

Sn_PORT0	Sn_PORT1
0x13	0x88

図 6-34 ソケット n ソースポートレジスタ(Sn\_PORT)の内容例のように設定します。

## 6.10.24 ソケットnデスティネーションハードウェアアドレスレジスタ(Sn\_DHAR)

UDP または IPRAW プロトコルモードで SEND\_MAC コマンドを使用する場合、このレジスタには各ソケットの宛先ハードウェアアドレス(MAC アドレス)を設定します。Sn\_DHAR は 6 バイト長で、例えばハードウェアアドレスが 00:04:14:10:12:34 の場合、

Sn_DHAR0	Sn_DHAR1	Sn_DHAR2	Sn_DHAR3	Sn_DHAR4	Sn_DHAR5
0x00	0x04	0x14	0x10	0x12	0x34

図 6-35 デスティネーションハードウェアアドレスレジスタ(Sn\_DHAR)の設定例のように設定します。

## 6.10.25 ソケットnデスティネーションIPアドレスレジスタ(Sn\_DIPR)

Sn\_DIPR は TCP プロトコルモードで各ソケットの宛先 IP アドレスを設定するレジスタです。クライアントモードでは CONNECT コマンドを実行する前に設定してください。サーバモードでは接続時に設定されます。

UDP プロトコルモードでは、SEND や SENDMAC コマンドを実行する前に設定してください。

Sn\_DIPR は 4 バイト長で、例えば IP アドレスが 192.168.0.20 の場合、

Sn_DIPR0	Sn_DIPR1	Sn_DIPR2	Sn_DIPR3
192(0xC0)	168(0xA8)	0(0x00)	20(0x14)

図 6-36 ソケット n デスティネーション IP アドレスレジスタ(Sn\_DIPR)の内容例のように設定します。

## 6.10.26 ソケットnデスティネーションポートレジスタ(Sn\_DPORT)

Sn\_DPORT は TCP プロトコルモードで各ソケットの宛先ポート番号を設定するレジスタです。クライアントモードでは CONNECT コマンドを実行する前に設定してください。サーバモードでは接続時に設定されます。

UDP プロトコルモードでは、SEND や SENDMAC コマンドを実行する前に設定してください。

Sn\_PORT は 2 バイト長で、例えばポート番号が 5000(=0x1388)の場合、

Sn_DPORT0	Sn_DPORT1
0x13	0x88

図 6-37 ソケット n デスティネーションポートレジスタ(Sn\_DPORT)の内容例のように設定します。

## 6.10.27 ソケット n 最大セグメントサイズレジスタ(Sn\_MSS)

Sn\_MSS は TCP の最大セグメントサイズを設定します。サーバモードでは通信相手側によって設定されます。

Sn\_MSS は 2 バイト長で、例えば 1460(=0x05B4)の場合、

Sn_MSS0	Sn_MSS1
0x05	0xB4

図 6-38 ソケット n 最大セグメントサイズレジスタ(Sn\_MSS)の内容例のように設定します。

## 6.10.28 ソケット n IP プロトコルレジスタ(Sn\_PROTO)

Sn\_PROTO は IPRAW プロトコルモードで IP ヘッダのプロトコル番号を設定するレジスタです。例えば ICMP の場合プロトコル番号は 0x01 です。

## 6.10.29 ソケット n TOS レジスタ(Sn\_TOS)

Sn\_TOS は IP ヘッダのタイプオブサービスフィールドの値を設定するレジスタです。リセット後 0x00 となります。

## 6.10.30 ソケット n TTL レジスタ(Sn\_TTL)

Sn\_TTL は IP ヘッダの TTL フィールドの値を設定するレジスタです。リセット後 128(0x80)となります。

## 6.10.31 ソケット n TX フリーサイズレジスタ(Sn\_TX\_FSR)

Sn\_TX\_FSR は送信可能なデータサイズを読み出すことができます。Sn\_TX\_FSR は 2 バイト長ですが、読み出しする場合は上位バイト Sn\_TX\_FSR0、下位バイト Sn\_TX\_FSR1 の順に読み出ししてください。

例えば 2048(=0x0800)の場合、

Sn_TX_FSR0	Sn_TX_FSR1
0x08	0x00

図 6-39 ソケット n Tx フリーサイズレジスタ(Sn\_TX\_FSR)の内容例のように読み出されます。

## 6.10.32 ソケットnTX リードポインタレジスタ(Sn\_TX\_RR)

Sn\_TX\_RR は送信バッファメモリの送信が終了したアドレスを示します。SEND コマンドで現在の Sn\_TX\_RR から Sn\_TX\_WR までのデータが送信され、送信終了で Sn\_TX\_RR の値は Sn\_TX\_WR の値で更新されます。Sn\_TX\_RR は 2 バイト長ですが、読み出しする場合は上位バイト Sn\_TX\_RR0、下位バイト Sn\_TX\_RR1 の順に読み出ししてください。

## 6.10.33 ソケットnTx ライトポインタレジスタ(Sn\_TX\_WR)

Sn\_TX\_WR は送信バッファメモリの送信データ書き込みアドレスを示します。SEND コマンドで送信されるデータサイズは、現在の Sn\_TX\_RR から Sn\_TX\_WR までのデータとなりますので、送信データを送信バッファメモリに書き込み後、Sn\_TX\_WR を送信データサイズ分進めてください。

なお Sn\_TX\_WR の示すアドレスから実際の送信バッファメモリのオフセットアドレスを取得する方法はをご参照ください。Sn\_TX\_WR は 2 バイト長ですが、読み出しする場合は上位バイト Sn\_TX\_WR0、下位バイト Sn\_TX\_WR1 の順で読み出ししてください。

## 6.10.34 ソケットnRx 受信サイズレジスタ(Sn\_RX\_RSR)

Sn\_RX\_RSR は受信データサイズを読み出すことができます。Sn\_RX\_RSR は 2 バイト長ですが、読み出しする場合は上位バイト Sn\_RX\_RSR0、下位バイト Sn\_RX\_RSR1 の順に読み出ししてください。

## 6.10.35 ソケットnRX リードポインタレジスタ(Sn\_RX\_RD)

Sn\_RX\_RD は受信バッファメモリの受信データ読み出しアドレスを示します。Sn\_RX\_RD が示すアドレスから実際の受信バッファメモリのオフセットアドレスを取得する方法はをご参照ください。Sn\_RX\_RD は 2 バイト長ですが、読み出しする場合は上位バイト Sn\_RX\_RD0、下位バイト Sn\_RX\_RD1 の順で読み出ししてください。

# 6.11 VNC1L

VNC1L は FAT ファイルシステムが内蔵された USB ホストコントローラです。

## 6.11.1 ファームウェア

VNC1L には内部のフラッシュメモリに書き込みする数種類のファームウェアが用意されており、HT1040 では VDAP と VDPS をサポートしています。表 6-58 にファームウェアと、対応している機能を示します。

HT1040 出荷時の VNC1L には VDPS ファームウェアが書き込まれています。

表 6-58 ファームウェア機能

ファームウェア	ポート	機能
VDAP	1	USBディスク、FTDIスレーブペリフェラル、キーボード、マウス、プリンタ
	2	USBディスク、FTDIスレーブペリフェラル、キーボード、マウス、プリンタ
VDPS	1	FT232スレーブとしてUSBホストに接続
	2	USBディスク、FTDIスレーブペリフェラル、キーボード、マウス、プリンタ

## 6.11.2 H8/3069 との接続回路

VNC1LはH8/3069のSIOチャンネル2およびポートBを通して接続されています。図 6-40にVNC1L周辺の回路を示します。

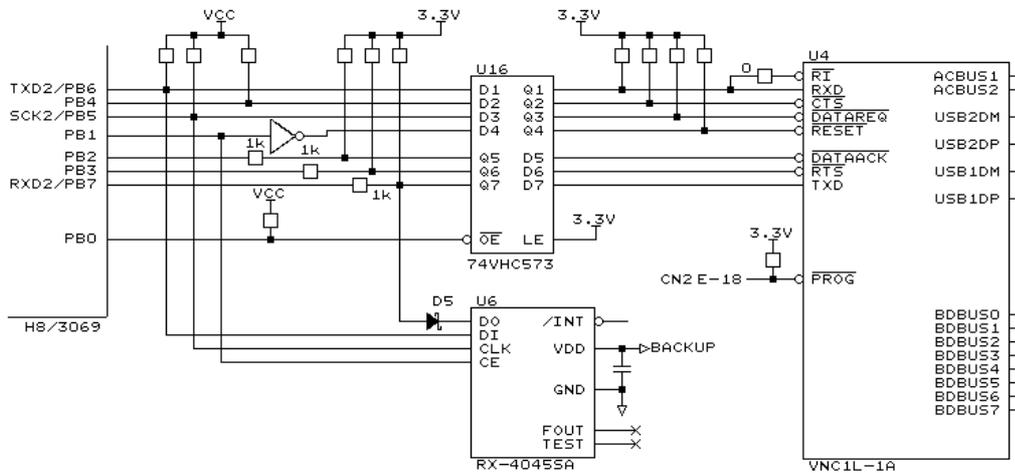


図 6-40 VNC1L 周辺回路

VNC1L と H8/3069 は非同期シリアルインタフェース SCI チャンネル 2 の TXD2、RXD2 を通して接続されています。その他の制御信号は H8/3069 のポート B を通して接続されています。



《重要》SCI チャンネル 2 の送受信信号は RTC と兼用のため、VNC1L をアクセスする際はポート B のビット 1(PB1)を L レベルとして RTC が選択されないようにしてください。一方、RTC をアクセスする場合は、PB0 を H レベルにし、VNC1L が同時にアクセスされないよう、ポート B のビット 0(PB0)を H レベルにしてください。ポート B の各ビットの用途を表 6-59に示します。この表では PB0 が L で VNC1L をアクセスする場合の機能を説明しています。PB0 が H の場合については表 6-42をご参照ください。

表 6-59 PBDR のビット機能(PB0=L の場合)

PB	入出力	機能
PB0	出力	VNC1L と RX-4045SA のどちらを制御するか選択します。 0: VNC1L を制御 1:RX-4045SA を制御
PB1	出力	VNC1L の/RESET 入力を制御します。 0:/RESET=H 1:/RESET=L
PB2	入力	VNC1L の/DATAACK 端子状態を読み出します。 0: /DATAACK=L 1: /DATAACK=H

PB3	入力	VNC1L の/RTS 端子状態を読み出します。 0: /RTS=L 1: /RTS=H
PB4	出力	VNC1L の/CTS 入力を制御します。 0: /CTS=L 1: /CTS=H
PB5	出力	VNC1L の/DATAREQ 入力を制御します。 0: /DATAREQ=L 1: /DATAREQ=H
PB6	出力	VNC1L の RXD 入りに接続されています。(SCI2 の TXD 出力)
PB7	入力	VNC1L の TXD 出力に接続されています。(SCI2 の RXD 入力)

### 6.11.3 VNC1L /RESET

VNC1L を初期化する入力です。2ms 程度 L レベルとし、通常は H レベルとしてください。なおこの端子をドライブするポート B のビット 1(PB1)は RTC の CE 端子と兼用になっており、PB1 に 1 を設定すると VNC1L の/RESET が L になると同時に RTC の CE が H になります。RTC は CE 端子が 1 秒以上 H になると計時桁上げ機能に影響がでますので、1 秒以上 PB1 の出力を 1 に設定しないようご注意ください。

### 6.11.4 /RI

VNC1L は SUM コマンドで動作を停止して低消費電力状態にすることができ、/RI 入力はこの状態からのウェークアップに使用されます。/RI は RXD と接続されていますので、CPU から VNC1L に何らかのデータを送信してウェークアップさせることができます。

### 6.11.5 /RTS, /CTS

HT1040 出荷時に書き込みされている VNC1L のファームウェアは送受信のハンドシェイクに/RTS と/CTS を使用しています。CPU が VNC1L からの受信データを受け取る場合は PB4 を使って/CTS を L にしてください。また CPU から VNC1L に送信する場合は、PB3 を通して/RTS が L であることを確認して送信してください。

### 6.11.6 /DATAREQ, /DATAACK

VNC1L にはコマンドモードとデータモードがあり、/DATAREQ 入力でコマンドモードとデータモードを切り替えます。/DATAREQ 入力が H の場合はコマンドモード、L の場合はデータモードとなります。/DATAACK は/DATAREQ で指定されたモードが使用できる状態となったときに/DATAREQ 入力と同じレベルが出力されます。



#### 《参考》

#### コマンドモード

USB ディスクやキーボード、マウス等のデバイス操作にはコマンドモードを使用します。コマンドモードでは CPU は SCI2 から VNC1L にコマンドを送り、VNC1L はコマンドに応じた処理結果を報告します。

#### データモード

スレーブに FTDI の FT232 等の UART デバイスを使用する場合や、ポート 1 を USB ホスト(パソコン)に接続して使用する場合、データモードではホスト側との送受信データがそのまま VNC1L の TXD,RXD 端子に得られます。

## 6.11.7 汎用ポート

HT1040 では VNC1L の汎用入出力ポート BDBUS[7:0]が CN8 に接続されています。また ACBUS1 は MAX1607 の /OC 出力に、ACBUS2 は MAX1607 の /EN 入力に接続されています。VNC1L の電源電圧が 3.3V のため、出力の H レベル電圧は 3.3V ですが、入力は 5V までの電圧が入力可能です。入出力切替や出力状態変更は IOW コマンド、状態入力は IOR コマンドで行います。

表 6-60 に汎用ポート初期状態と、その用途を示します。

表 6-60 VNC1L 汎用ポート

ビット	初期状態	初期値	用途
ACBUS1	出力	H	MAX1607 /OC 入力、入力に設定変更が必要
ACBUS2	出力	H	MAX1607 /EN 出力
BDBUS0	出力	H	
BDBUS1	出力	H	
BDBUS2	出力	L	
BDBUS3	出力	H	
BDBUS4	出力	H	
BDBUS5	出力	H/L	LED1 出力
BDBUS6	出力	H/L	LED2 出力
BDBUS7	出力(VDAP) 入力(VDPS)	H -	VDPS の場合はホスト接続の検出用入力

## 6.11.8 LED

BDBUS5 は LED1、BDBUS6 は LED2 接続用に予約されており、それぞれ表 6-61 のように動作します。

表 6-61 LED 機能

状態	LED1	LED2
電源オン	点滅	点滅
USB ディスク準備中	オン	オフ
USB ディスクレディ	オフ	オン
USB ディスク取り外し	オフ	オフ
コマンド実行	オフ	点滅
コマンド実行(USB ディスク無)	オフ	オフ

電源オン時の点滅は、VNC1L の /CTS 入力が L にドライブされ VNC1L がデータを送信できる状態になるまで続きます。LED 接続回路例を図 6-41 に示します。

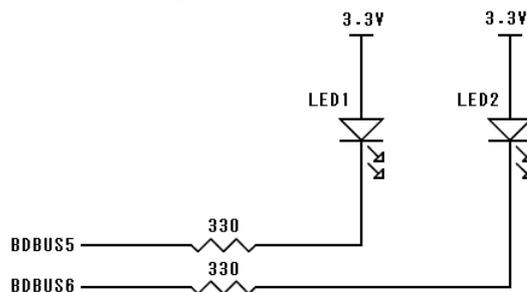


図 6-41 LED 接続回路例

## 6.11.9 MAX1607 スイッチ

USB コネクタ(CN6、CN6A、CN7、CN7A)に接続される USB デバイスに供給できる電流は MAX1607 によって合計最大 1A に制約されています。MAX1607 の/EN 入力(L でスイッチオン、H でオフ)は VNC1L の ACBUS2 端子に、/OC 出力(過電流やサーマルシャットダウン時 L)は ACBUS1 端子に接続されています。USB ポートに電源を供給する場合は、ACBUS2 を通して/EN を L に設定してください。



### 《注意》

ACBUS2 は VNC1L の初期化時に出力となっていますので、/OC 端子の状態を取得する場合は IOW コマンドを使って ACBUS 1 が入力となるように設定してください。

## 6.11.10 ファームウェアの更新

VNC1L のファームウェアは、以下の 2 つの方法で更新することができます。詳しくは Vinctum Firmware User Manual 3.1.3 項、6.10.3 項をご参照ください。

### 1. ファームウェアを USB メモリに用意し、自動更新する方法

FTD 形式のファームウェアイメージファイルを "FTRFB.FTD" として USB メモリのルートディレクトリに用意します。HT1040 で S ファイルローダを起動し、USB メモリを CN6A コネクタに差し込みます。VNC1L は USB メモリが挿入されると自動的にイメージファイルを認識し、ファームウェアを更新します。

### 2. ファームウェアを USB メモリに用意し、コマンドで書き込みする方法

FTD 形式のファームウェアイメージファイルを任意の名称で USB メモリに保存し、HT1040 の CN6A コネクタに差し込みます。HT1040 の S ファイルローダで USB コマンドを使い、VNC1L に "FWU ファイル名" としてイメージファイルを指定してアップデートを実行します。イメージファイルはルートディレクトリ以外に置くことができますが、FWU コマンド実行前に DIR コマンドでカレントディレクトリをイメージファイルのあるディレクトリに移動しておく必要があります。



### 《注意》

Vinctum の WEB サイトからダウンロードしたファームウェアは、**HT1040 に書き込みする前に必ずファームウェアカスタマイズツールで設定を変更してください。**そのまま HT1040 に書き込みすると 8 章で説明するライブラリが使用できなくなる可能性があります。設定変更については 7.4.3 項をご参照ください。

## 6.12 ポート V

ポート V は入力専用の 8 ビットポートで、アドレス 220000 番地を占有します。入力は TTL レベルです。RM3 にシングルインライン 8 素子抵抗モジュールを取り付けてプルアップまたはプルダウンすることができます。6.17 節をご参照ください。

## 6.13 ブートモード

HT1040 では、SW1 や JP3 で選択されるブートモードによるフラッシュメモリ書き込みには H8/3069 のブートモードではなく、ユーザブートモードを使用しています。(H8/3069 のブートモードを使用してフラッシュメモリ書き換えを行うと、MAC アドレスを保存しているユーザブートマットのフラッシュメモリが消去されてしまうためです。)

H8/3069 のブートモードを使用してフラッシュメモリの書き換えを行う場合は、TP3 を L レベルにしてリセットまたは電源投入を行ってください。



《重要》TP3 を L レベルにしてリセットまたは電源投入し、H8/3069 のブートモードでフラッシュメモリを書き込みする場合は、ユーザマットに保存されている MAC アドレスが消去されます。MAC アドレスを再設定する場合は HT1040WR.EXE を使用してください。(再設定時にはユーザマット内容は消去されず。)



表 6-63  $t_{w1}, t_{w2}$ 

アクセスステート・ウェイト	$t_{w1}$	$t_{w2}$
2 アクセスステート	0	0.5
3 アクセスステート、0 ウェイト	1	1
3 アクセスステート、1 ウェイト	2	2
3 アクセスステート、2 ウェイト	3	3
3 アクセスステート、3 ウェイト	4	4

## 6.15 バックアップ

HT1040 に搭載されている 512KBSRAM とカレンダー時計(RX-4045SA)は、ボード上の 0.1F 電気二重層コンデンサでバックアップされています。常温で約 2 日程度バックアップすることができますが、さらに長期間のバックアップが必要な場合は、ボード上にタブ端子付きのコイン型リチウム電池を実装するか、外部電池を CN4 から接続してください。コイン型リチウム電池を搭載する場合は、極性に注意してください。ボード上に正極側が + 印で示されています。

バックアップ電池には通常のマンガン・アルカリ乾電池やリチウム 1 次電池、ニッケルカドミウム電池等が使用できます。ニッケルカドミウム電池の場合は、基板上にトリクル充電用の抵抗を実装するスペース(R4)が用意されていますので、電池の仕様にあわせた適切な抵抗を取り付けることができます。

なお、HT1040 に搭載されている SRAM のデータ保持最低電圧は 2.0V(min.)、カレンダー時計の計時最低電圧は 1.15V(Min.)です。

## 6.16 拡張バス接続

HT1040 に HT シリーズの I/O モジュールをスタック接続して拡張することが可能です。ドライブできる負荷容量の観点から、4 枚程度まで接続可能ですが、それ以上の拡張が必要な場合は、HT9070 バスバッファボードの併用をご検討ください。HT9070 を使用して 10 枚程度まで I/O モジュールを増設することが可能となります。

## 6.17 RM1 ~ 4

RM1 ~ 4 は、ポート端子のプルアップ・プルダウンを HT1040 ボード上で行うためのシングルインライン 8 素子抵抗モジュールを取り付けするスペースです。各スペースは 10 極となっており、そのうちの 9 極を使用して抵抗を取り付けると、プルアップまたはプルダウンすることができます。印のある 1 ピン側が VCC ですので、1 ピン側に集合抵抗のコモンをあわせて取り付ければプルアップ、10 ピン側は GND ですから、このピンに集合抵抗のコモンをあわせて取り付ければプルダウンとなります。

表 6-64 RM とプルアップ・プルダウンされるポート

抵抗	ポート
RM1	P4
RM2	PA
RM3	PV
RM4	P7

# 7 ソフトウェア

---

本章では、ソフトウェアに関連する事項を説明します。CPU や W5100 のレジスタ機能等については、本マニュアル 6 章の説明や各デバイスのハードウェアマニュアル・データシートをご参照ください。

## 7.1 H8/3069 レジスタ設定

### 7.1.1 システムクロックの設定

初期状態では、システムクロックは 16MHz となります。次のような場合には、システムクロックを 1/2, 1/4, 1/8 に設定して使用することができます。

- オプションメモリソケットに増設するメモリのアクセスタイムが遅い場合
- 消費電流を低くするため、システムクロックを下げたい場合
- CN1 から増設する回路の SYSCCLK が 16MHz では動作しない場合

#### 設定レジスタ

- DIVCR (システムクロック分周比設定)

#### 設定例

システムクロックを 8MHz に設定する場合

```
DIVCR      .EQU      #H'FEE01B
            MOV.B    #H'01,R0L
            MOV.B    R0L,@DIVCR
```

### 7.1.2 アドレス出力

H8/3069 のモード 5 では、アドレスバス出力ピンが初期状態でポート入力となるため、ボード上の SRAM や CN1 拡張バス、W5100 を使用するため出力となるように設定する必要があります。

#### 設定レジスタ

- P1DDR (アドレス A7 ~ A0 を出力にする設定)
- P2DDR (アドレス A15 ~ A8 を出力にする設定)
- P5DDR (アドレス A19 ~ A16 を出力にする設定)

#### 設定例

A19 ~ A0 を有効にする場合

```
P1DDR      .EQU      #H'FEE000
P2DDR      .EQU      #H'FEE001
P5DDR      .EQU      #H'FEE004
            MOV.B    #H'FF,R0L
            MOV.B    R0L,@P1DDR
            MOV.B    R0L,@P2DDR
            MOV.B    R0L,@P5DDR
```

## 7.1.3 アクセスステート数とウェイト数

H8/3069 のバス接続で拡張されるデバイスは、アドレス範囲によって 8 つのエリアに分かれており、各エリアに対して アクセスステート数とウェイト数を設定することができます。初期値は 3 ステートアクセス・3 ウェイトに設定されていますので、そのまま使用しても支障ありませんが、必要に応じてアクセスサイクルを短くすることが可能です。デバイス毎に設定可能な範囲がありますので、詳しくは7.1.5項以降の説明をご参照ください。なお、アクセスステートを 2 ステートに設定したエリアでは、WCRL による追加のウェイト数設定は無視され、ウェイトは挿入されませんのでご注意ください。

### 設定レジスタ

- WCRL (エリア 0~3 のウェイト数を設定する)
- ASTCR (アクセスステート数の設定)

### 設定例 1

エリア 2 を 3 ステートアクセスとし、ウェイトをいれない場合

```
ASTCR    .EQU    #H'FEE021
WCRL     .EQU    #H'FEE023
          MOV.B   #B,11111111,R0L ;全エリア 3 ステートアクセス
          MOV.B   R0L,@ASTCR
          MOV.B   #B'11001111,R0L
          MOV.B   R0L,@WCRL
```

### 設定例 2

エリア 3 のアクセスを 2 ステートアクセスにする場合

```
ASTCR    .EQU    #H'FEE021
          MOV.B   #B'11111011,R0L
          MOV.B   R0L,@ASTCR
```

## 7.1.4 /CS 出力設定

HT1040 に搭載されているデバイスの選択や、CN1 に出力される制御信号の発生には H8/3069 の/CS 出力機能を使用しています。H8/3069 の初期値のままでは、/CS1、/CS2 端子がポート入力となっているため、/CS 出力となるように設定する必要があります。

### 設定レジスタ

- P8DDR (/CS1、/CS2 を有効にする設定)

### 設定例

/CS1、/CS2 を有効にする場合

```
P8DDR    .EQU    #H'FEE007
          MOV.B   #B'11101100,R0L
          MOV.B   R0L,@P8DDR
```

## 7.1.5 IOCHRDY 有効・無効の設定

IOCHRDY は、バスサイクルの延長入力(H8/3069 の/WAIT)です。H8/3069 の初期値はこの機能が無効となっていますので、必要に応じて有効に設定してください。

### 設定レジスタ

- BCR (/WAIT 入力の有効・無効設定)

### 設定例

IOCHRDY を有効にする場合

```
BCR      .EQU    #H'FEE024
          MOV.B   #B'11000111,R0L
          MOV.B   R0L,@BCR
```

## 7.1.6 SRAM(512KB)を使用するための設定

ボード上の 512KB SRAM は、/CS2 で選択されています。/CS2 端子は初期状態で P82 ポート入力となるため、/CS2 出力となるよう P8DDR を設定する必要があります。SRAM は、システムクロックが 16MHz の場合 3 ステートアクセス、ノーウェイトか、それよりも遅いタイミングでご使用ください。システムクロックが 8MHz の場合は、2 ステートアクセスでのアクセスが可能です。

設定が必要な関連するレジスタは次の通りです。

- P1DDR (アドレス A7 ~ A0 を出力にする設定)
- P2DDR (アドレス A15 ~ A8 を出力にする設定)
- P5DDR (アドレス A19 ~ A16 を出力にする設定)
- P8DDR (/CS を有効にする)
- ASTCR(アクセスステート数の設定)
- DIVCR (システムクロックを 8MHz にする必要がある場合)

## 7.1.7 CN1 拡張バスを使用するための設定

CN1 拡張バスを使用する場合は、以下の H8/3069 内蔵レジスタを設定する必要があります。

- P1DDR (アドレス A7 ~ A0 を出力にする設定)
- P2DDR (アドレス A15 ~ A8 を出力にする設定)
- P5DDR (アドレス A19 ~ A16 を出力にする設定)
- P8DDR (/CS1 を出力にする設定、/IOR,/IOW,/MEMR,/MEMW 出力に必要)
- DIVCR (システムクロックを 8MHz にする必要がある場合)
- WCRL (ウェイトサイクルの挿入)
- ASTCR (アクセスステート数の選択)

PC/104 バスの拡張モジュールにはシステムクロックとして 8MHz が必要なものがありますので、接続するボードの仕様を充分ご確認ください。また、PC/104 の I/O アクセスタイミング規定に一番近いタイミングとするためには、システムクロックを 8MHz としたうえで、エリア 1 について 3 ステートアクセス、3 ウェイト挿入を設定する必要があります。なお、W5100 も同じエリア 1 にマッピングされるため、レジスタ設定内容は W5100 のアクセスにも適用されます。

## 7.1.8 W5100 を使用するための設定

W5100 を使用する場合は、以下の H8/3069 内蔵レジスタを設定する必要があります。

- P1DDR (アドレス A7 ~ A0 を出力にする設定)
- P2DDR (アドレス A15 ~ A8 を出力にする設定)
- P5DDR (アドレス A19 ~ A16 を出力にする設定)
- P8DDR (/CS1 を出力にする設定、/IOR,/IOW 出力に必要)
- DIVCR (システムクロックを 8MHz にする必要がある場合)
- WCRL (ウェイトサイクルの挿入)
- ASTCR (アクセスステート数の選択)

W5100 は、CN1 を通して接続される拡張モジュールの I/O・メモリと同じエリア 1 にマッピングされるため、CN1 に接続する拡張モジュールがある場合は、レジスタへの設定値はこれらの拡張モジュールの動作タイミングを考慮して決定する必要があります。

ります。CN1 に拡張モジュールを接続しない場合は、2 ステートアクセス、ノーウェイトの設定が可能です。

## 7.2 MAC アドレスの取得

HT1040 はイーサネットで使用する MAC アドレスを H8/3069 のユーザブートマット (0x000004 番地から 6 バイト) に保存しています。ユーザブートマットは H8/3069 のフラッシュメモリ更新プログラムが書き込まれている領域ですが、アプリケーションプログラムから以下の手順のプログラムを H8/3069 の内蔵 SRAM 上で実行し、内容をアクセスすることができます。(マスク不能な割り込み/IOCHCHK には信号が加わらないようにしてください。)

1. H8/3069 の CCR でマスクブル割り込みを禁止
2. FMATS レジスタ(0xFEE0B5 番地)に 0xAA を書きこみ(ユーザブートマット選択)
3. 0x000004 番地からの 6 バイトを RAM に読み出しする
4. FMATS レジスタに 0x00 を書き込みし、ユーザマットに切替する

なお HT1040 用ライブラリに MAC アドレス取得関数 `getmac_from_flash` を用意しています。8.3.1項をご参照ください。

HT1040 では通常ユーザブートマットに用意している専用の書き込みプログラムを使用してフラッシュメモリの更新を行っていますが、H8/3069 のブートモードを使用してフラッシュメモリの更新をする場合は、ユーザブートマットの内容が消去され、MAC アドレスも失われます。ユーティリティの HT1040WR.EXE を使って MAC アドレスを再設定することができます。CN5 上に MAC アドレスが貼り付けてありますので、このアドレスを再設定してください。(MAC アドレス再設定時にユーザマットの内容は消去されますのでご注意ください。)

## 7.3 Yellow IDE/Yellow Scope を使用する

有限会社イエローソフト(<http://www.yellowsoft.com/>)では H8 のソフトウェア開発に便利な C コンパイラ(Yellow C)やリモートデバッガ(Yellow Scope)を販売しています。これらのツールを使用する場合は、イエローソフトから提供されている H8/3067 用ローダおよびスタートアップルーチンを HT1040 用に修正する必要があります。

なお、出荷時にフラッシュメモリに書き込まれている S ファイルローダの L コマンドを使って、YellowIDE で RAM ヘダウンロード(S)を選択して作成したプログラムを実行することができます。RAM の先頭は 0x400000 を指定してください。

### 7.3.1 ローダプログラム

YellowIDE からプログラムを RAM に転送して実行するためのローダプログラムは、デフォルトで `c:\¥YellowIDE6¥Loader¥梅澤無線¥HT1030` に用意されています。HT1040 でローダを使用する場合は、このフォルダ内の C1.asm に以下の追加・修正を行ってください。(変更が必要な箇所は下記網がけ部分です。)

```

=====
;
;          !!!!!!!場合によって修正必須!!!!!!
;
;      イエローソフトの CPU ボード以外を使用される場合は修正
;      が必要です。
;
;
;      外部メモリの設定
;      下記の例はエリア 1 にバス幅 8 ビットの RAM が接続されている場合の例です
=====
;アドレスバス   ポート
P1DDR           EQU       H'FEE000
P2DDR           EQU       H'FEE001
P5DDR           EQU       H'FEE004
P8DDR           EQU       H'FEE007
ASTCR           EQU       H'FEE021
;バスコントローラ
ABWCR           EQU       H'FEE020
ASTCR           EQU       H'FEE021
WCRH            EQU       H'FEE022
WCRL            EQU       H'FEE023
BCR             EQU       H'FEE024

IF ((DEFD __YIDE_ROM__) || (DEFD __YIDE_ROM_DEBUG__)) ;<- ROM 化と ROM デバッグ時のみ
必要
;<<変更  >>
;外部メモリの設定
;ポート 1、 2、 5 を出力に設定することにより
;アドレスバスとして機能させる
;外部メモリを使用しない場合はコメントにして下さい。
MOV.B   #H'FF,R0L
MOV.B   R0L,@P1DDR
MOV.B   R0L,@P2DDR
MOV.B   R0L,@P5DDR

;CS1/CS2 端子をチップセレクト端子として有効にする
MOV.B   #H'EC,R0L
MOV.B   R0L,@P8DDR

MOV.B   #B'11111011,R0L
MOV.B   R0L,@ASTCR           ;2 ステートアクセスの場合、コメントははず

;MOV.B   @ABWCR,R0L
;AND.B   #B'11111101,R0L
;MOV.B   R0L,@ABWCR           ;16 ビットバス幅の場合、コメントははず。
MOV.B   #B'11000111,R0L
MOV.B   R0L,@BCR
ENDIF

;----- ユーザが変更するのはここまでです -----

```

## 7.3.2 スタートアッププログラム

YellowScope に対応したスタートアッププログラムは、デフォルトで c:\¥YelloIDE6¥STARTUP¥梅澤無線¥HT1030 に Cs3067.asm としてインストールされています。HT1040 で使用する場合は、以下の追加・修正を行ってください。(変更が必要な箇所は下記網がけ部分です。)

```
=====
;
;          !!!!!!!場合によって修正必須!!!!!!
;
;   イエローソフトの CPU ボード以外を使用される場合は修正
;   が必要です。
;
;
;   外部メモリの設定
;   下記の例はエリア 1 にバス幅 8 ビットの RAM が接続されている場合の例です
;
=====
;アドレスバス   ポート
P1DDR           EQU       H'FEE000
P2DDR           EQU       H'FEE001
P5DDR           EQU       H'FEE004
P8DDR           EQU       H'FEE007
ASTCR           EQU       H'FEE021
;バスコントローラ
ABWCR           EQU       H'FEE020
ASTCR           EQU       H'FEE021
WCRH            EQU       H'FEE022
WCRL            EQU       H'FEE023
BCR             EQU       H'FEE024

IF ((DEFD __YIDE_ROM__) || (DEFD __YIDE_ROM_DEBUG__)) ;<- ROM 化と ROM デバッグ時のみ
必要
;<<変更 >>
;外部メモリの設定
;ポート 1、 2、 5 を出力に設定することにより
;アドレスバスとして機能させる
;外部メモリを使用しない場合はコメントにして下さい。
MOV.B   #H'FF,R0L
MOV.B   R0L,@P1DDR
MOV.B   R0L,@P2DDR
MOV.B   R0L,@P5DDR

;/CS1,/CS2 端子をチップセレクト端子として有効にする
MOV.B   #H'EC,R0L
MOV.B   R0L,@P8DDR

MOV.B   #B'11111011,R0L
MOV.B   R0L,@ASTCR           ;:2 ステータアクセスの場合、コメントははずす
;MOV.B   @ABWCR,R0L
;AND.B   #B'11111101,R0L
;MOV.B   R0L,@ABWCR         ;:16 ビットバス幅の場合、コメントははずす。
mov.b   #B'11000111,R0L
mov.b   R0L,@BCR
ENDIF
```

## 7.4 VNC1L

### 7.4.1 アプリケーションで使用するための設定

VNC1L は H8/3069 の SCI2 に接続されています。SCI2 には RTC も接続されているため、切替回路を制御し VNC1L と SCI2 が接続されるよう以下のような手順でレジスタを設定してください。ライブラリの usb\_init 関数ソースもご参照ください。

```
PBDR に 0xfc、PBDDR に 0x33 を設定      /* 切替回路の制御 */
/* SCI2 を非同期シリアル 8bit パリティ無し、ストップビット 1 に設定 */
SCI2_SCMR に 0、SCI2_SMCR に 0、SC_BRR に 12 を設定
/* SCI2 を割り込みで受信する場合 */
SCI2_SCR に=0x70 を設定      /*Rx interrupt ,TxRx enable*/
PBDR の bit0 を 1 に設定し 2ms 待って 0 に戻す      /* VNC1L のリセット*/
1 秒待つ
SCI2 から VNC1L にコマンド"IOW $010400%x00d"を送信 /* MAX1607 を ON する */
```

### 7.4.2 Tips

VNC1L のデータシートやマニュアルにはタイミングの記載がありませんが、VNC1L からのコマンド応答メッセージを受信した後すぐに次のコマンドを送信すると、コマンドを無視したり、正しく認識されないことがあるようです。メッセージを受信してから 10ms 程度間隔をあけて次のコマンドを送信した場合はこの現象は発生しないようですので、メッセージ受信から 10ms 程度待ってから次のコマンドを送信することをお勧めします。

### 7.4.3 ファームウェアの初期設定内容

HT1040 出荷時の VNC1L には VDPS ファームウェアが書き込まれていますが、フロー制御や通信速度の初期設定はメーカー提供のファームウェアカスタマイズツール(VncFwMod.EXE)で変更することができます。HT1040 出荷時のファームウェア設定は下記の通りです。(ファームウェアカスタマイズツールで表示される内容のコピーです。)ファームウェアの設定を出荷時の状態から変更した場合は、次章で説明するライブラリが使用できない可能性があります。メーカー WEB サイトからダウンロードできるファームウェアの初期設定内容も以下の設定とは異なりますので、次章のライブラリを使用する場合は、ファームウェアカスタマイズツールで設定を変更してからファームウェアのアップデートを行ってください。

Default Monitor Settings: 38400 baud, 8 bits, 1 stop bit, no parity, CTS/RTS  
Initial Mode: ECS, IPA, LED Flash at Power-on  
Options: Allow Firmware Upgrades, Allow Data Mode,  
Show Prompt, No Device Connects/Removals, No Version Report at Startup.

なおマニュアルディスクの vnc1lfirm ディレクトリに、HT1040 のライブラリに適合するように設定変更した VDPS、VDAP ファームウェアを収録していません。



《注意》通信速度を 38400bps 以外に設定されているファームウェアを VNC1L に書き込みした場合、S ファイルローダは USB コントローラの初期設定ができないため USB ポートに電源が供給されず、USB ポートが使用できなくなりますので充分ご注意ください。

## 7.4.4 コマンドモード・データモード

VNC1L は H8/3069 の SCI2 に非同期シリアルで接続されています。

VNC1L にはコマンドモードとデータモードがあり、リセット後はコマンドモードとなっています。コマンドモードでは VNC1L がコマンドを受信して結果を返します。データモードでは、CPU から送られたデータはそのまま USB ポート 1 または 2 のデバイスに送られ、USB ポートに接続されたデバイスからのデータは直接 CPU に送られます。

コマンドモードで使用する場合は、VNC1L の/DATAREQ 信号を H レベルにします。データモードで使用する場合は/DATAREQ 信号を L レベルとし、/DATAACK が L になるのを待ってください。

データモードにする前にコマンドモードで SC コマンドを使い、CPU からの送受信データをどちらの USB ポートに接続するかを選択する必要があります。

## 7.4.5 モニタコマンドセット

コマンドモニタには Extended と Short の 2 つのコマンドセットがあり、コマンドや応答形式が異なります。リセット後どちらのコマンドセットを使用するか、ファームウェア変更ツールで変更できます。HT1040 出荷時は Extended コマンドセットとなっています。またそれぞれのコマンドセットを移行するコマンドも用意されています。(SCS、ECS コマンド)

## 7.4.6 数値モード

コマンドセットの選択とは独立に、ASCII モードと Binary モードの 2 つの数値モードが使用できます。

ASCII モードでは、コマンドの数値パラメータは ASCII 文字で入力します。数値の先頭に \$ または 0x を付加すると 16 進数として扱われます。ASCII モードでは 10 進数は 8 桁まで有効です。

例えば SBD コマンドの数値パラメータが 0x384100 の場合、

SBD \$384100  
SBD 0x384100  
SBD 3686656  
SBD 03686656

はいずれも同じ結果となります。

バイナリモードではコマンドによって決められた固定長の値をバイト単位で送信します。上記同じ SBD コマンドの例では、コマンドの後のパラメータはバイトデータを 0x38、0x41、0x00 の順に 3 バイト送信します。

コマンドの応答として返される数値は、ASCII モードではバイトごとに \$ をつけたバイト単位の文字列となります。例えばコマンドの応答が 0x000310aa を返す場合、

\$AA \$10 \$03 \$00

となります。バイト順が逆になっていること、最後のバイトと改行の間にスペースが 1 つあることに注意してください。

数値モードが ASCII でも、USB デバイスから返されたそのままのデータの場合は ASCII コードではなくバイナリで応答が返されます。(例えば DRD コマンドの応答等)

リセット後どちらのコマンドセットを使用するか、ファームウェア変更ツールで変更できます。HT1040 出荷時は ASCII コマンドセットとなっています。またそれぞれのコマンドセットを移行するコマンドも用意されています。(IPA、IPH コマンド)

## 7.4.7 ファイル名

VNC1L が USB メモリに作成するファイル名は ASCII 文字(大文字)、数字および下記記号のみが有効です。

\$%'-\_@~`!(){ }^#&

ASCII 小文字が使用された場合は大文字に変換されます。

ファイル名は 8.3 形式(ファイル名メイン部分 8 文字 + 拡張子 3 文字)で構成されます。ロングファイルネームには対応していません。

コマンドの引数として与えられたファイル名は次のルールが適用されます。

- ・ ファイル名に拡張子がなく、8 文字以上の場合は 9 文字目以降は拡張子として扱われます。
- ・ ファイル名が拡張子を含み、ファイル名メイン部分が 8 文字以上の場合はファイル名メイン部分は 8 文字で切り詰められます。
- ・ 拡張子が 3 文字より長い場合は拡張子は 3 文字に切り詰められます。

ファイル名は DIR コマンドで表示されます。表示されるファイル名はファイル名メイン部分+'.'(ドット)+拡張子の形式となります。ファイル名が 8 文字より短い場合や拡張子が 3 文字より短い場合に、スペースが追加されることはありません。

ディレクトリ名の場合は、ディレクトリ名+' '(スペース)+拡張子の形式となります。ディレクトリ名にもファイル名と同じルールが適用されます。

## 7.4.8 コマンド応答

コマンドが完了すると、VNC1L はプロンプト(正常終了)かエラーを返します。

表 7-1にコマンド応答の一覧を示します。

コマンド応答の最後には、必ず改行(0x0D)コードが送信されます。

(なお以降の説明では改行コードを' '記号で表しています。)

表 7-1 VNC1L コマンド応答

Extended コマンド	Short コマンド	内容
D:>	> (3E 0D)	コマンド正常終了
Bad Command	BC (42 43 0D)	コマンドが不正
Command Failed	CF (43 46 0D)	ファイル/ディレクトリ名が存在しない
Disk Full	DF (44 46 0D)	ディスクに空きがない
Invalid	FI (46 49 0D)	ディレクトリをオープンしようとした ディレクトリをファイルに変更しようとした
Read Only	RO (52 4F 0D)	リードオンリファイルに書き込みしようとした
File Open	FO (46 4F 0D)	ライトオープンされているファイルがあり コマンドが実行できない
Dir Not Empty	NE (4E 45 0D)	空でないディレクトリを削除しようとした
Filename Invalid	FN (46 4E 0D)	ファイル名が不正
No Upgrade	NU (4E 55 0D)	ファームウェアのアップグレードファイル が存在しない
No Disk	ND (4E 44 0D)	ディスクが存在しない

## 7.4.9 イベント通知

VNC1LはUSBポートのデバイス挿入や取り外しをメッセージで通知する機能があります。通知されるメッセージを表 7-2に示します。HT1040 出荷時のファームウェアでは、このメッセージを出力しない設定となっています。

表 7-2 イベント通知メッセージ

Extended コマンド	Short コマンド	内容
Device Detected P1	DD1 (44 44 31 0D)	USB ポート 1 にデバイスが接続された
Device Removed P1	DR1 (44 52 31 0D)	USB ポート 1 のデバイスが取り外しされた
Device Detected P2	DD2 (44 44 32 0D)	USB ポート 2 にデバイスが接続された
Device Removed P2	DR2 (44 52 32 0D)	USB ポート 2 のデバイスが取り外しされた
Slave Enabled	SDA (53 44 41 0D)	USB ホストに接続された
Slave Disabled	SDD (53 44 44 0D)	USB ホストから切断された

USB メモリがポート 2 に接続された場合は、'FTRFB.FTD'の名称でファームウェアアップデートファイルがルートディレクトリに存在しないかをチェックします。このため、このファイルが存在しない場合の通知メッセージは

```
Device Detected P2
No Upgrade
D:>
となります。
```

## 7.4.10 コマンド

VNC1L のコマンドを表 7-3から表 7-10に示します。各コマンドの詳細については VinculumFirmware User Manual をご参照ください。

表 7-3 モニタコンフィグレーションコマンド

Extended コマンド	Short コマンド	内容
SCS	10 0D	Short コマンドセットを指定
ECS	11 0D	Extended コマンドセットを指定
IPA	90 0D	ASCII モード数値を指定
IPH	91 0D	Binary モード数値を指定
SBD divisor	14 20 divisor 0D	ボーレートを変更
FWV	13 0D	ファームウェアバージョンを表示
E	45 0D	エコーコマンド(E をエコーバック)
e	65 0D	エコーコマンド(e をエコーバック)

表 7-4 ディスクコマンド

Extended コマンド	Short コマンド	内容
DIR	01 0D	カレントディレクトリのリスト
DIR file	01 20 file 0D	指定されたファイルのサイズを表示
CD file	02 20 file 0D	カレントディレクトリを変更
CD ..	02 20 2E 2E 0D	1つ上のディレクトリ階層へ移動
RD file	04 20 file 0D	指定されたファイル全てを読み出し
DLD file	05 20 file 0D	指定されたサブディレクトリを削除
MKD file	06 20 file 0D	指定されたサブディレクトリを作成
MKD file datetime	06 20 file 20 datetime 0D	指定された日時でサブディレクトリを作成
DLF file	07 20 file 0D	指定されたファイルを削除
WRF dword data	08 20 dword 0D data	指定されたバイト数を現在オープンしているファイルにデータ書き込み
OPW file	09 20 file 0D	既存のファイルを書き込みオープンまたは新規作成
OPW file datetime	09 20 file 20 datetime 0D	日時を指定して既存のファイルを書き込みオープンまたは新規作成
CLF file	0A 20 file 0D	現在オープンしているファイルをクローズ
RDF dword	0B 20 dword 0D	指定されたバイト数をオープンされているファイルから読み出し
REN file file	0C 20 file 20 file 0D	ファイルまたはディレクトリをリネーム
OPR file	0E 20 file 0D	ファイルを読み出し用にオープン
OPR file date	0E 20 file 20 date 0D	ファイルを読み出し用にオープンし、ファイルアクセス日を設定
SEK dword	28 20 dword 0D	現在オープンしているファイルで指定された位置へシーク
FS	12 0D	ディスクの空き容量を報告(4GB 未満の空き)
FSE	93 0D	ディスクの空き容量を報告
IDD	0F 0D	4GB 未満のディスクのディスク情報
IDDE	94 0D	ディスク情報
DSN	2D 0D	ディスクシリアル表示
DVL	2E 0D	ディスクボリュームラベル表示
DIRT file	2F 20 file 0D	ファイル作成・修正・アクセス日付表示

表 7-5 パワーマネージメント

Extended コマンド	Short コマンド	内容
SUD	15 0D	ディスクのサスペンド
WKD	16 0D	ディスクのウェーク
SUM	17 0D	モニタのサスペンド

表 7-6 未使用 I/O ピンコマンド

Extended コマンド	Short コマンド	内容
IOR byte	29 20 byte 0D	指定された I/O ポートのリード
IOW byte+byte+byte	2A 20 byte byte byte 0D	指定された I/O ポートの入出力設定および出力値設定

表 7-7 プリンタコマンド

Extended コマンド	Short コマンド	内容
PGS	81 0D	プリンタステータスの取得
PSR	82 0D	プリンタのソフトリセット

表 7-8 USB デバイスコマンド

Extended コマンド	Short コマンド	内容
QP1	2B 0D	ポート 1 問い合わせ
QP2	2C 0D	ポート 2 問い合わせ
QD byte	85 20 byte 0D	デバイス問い合わせ
SC byte	86 29 byte 0D	カレントデバイスを指定
DSD byte data	83 20 byte 0D data	USB デバイスにデータを送信
DRD	84 0D	USB デバイスからデータを受信
SSU qword (data)	9A 20 qword 0D (data)	USB デバイスのコントロールエンドポイントにセットアップコマンドを送信
SF byte	87 20 byte 0D	デバイスを FTDI のデバイスとみなす
QSS	98 0D	スレーブステータス問い合わせ

表 7-9 FT232/FT2345/FT2232 用コマンド

Extended コマンド	Short コマンド	内容
FBD divisor	18 20 divisor 0D	ビットレート設定
FMC word	19 20 word 0D	モデムコントロール設定
FSD word	1A 20 word 0D	データフォーマット設定
FFC byte	1B 20 byte 0D	フロー制御設定
FGM	1C 0D	モデムステータス取得
FSL byte	22 20 byte 0D	レイテンシータイム設定
FSB word	23 20 word 0D	ビットモード設定
FGB	24 0D	ビットモードデータ取得

表 7-10 デバッグコマンド

Extended コマンド	Short コマンド	内容
SD dwork	03 20 dword 0D	指定セクタからデータ読み出し
SW dword data	92 20 dword 0D data	指定セクタにデータ書き込み
FWU file	95 20 file 0D	指定ファイルでファームウェアをアップデート

divisor	3 バイトのボーレート設定値
byte	1byte(8bit)値
word	2byte(16bit)値
dword	4byte(32bit)値
qword	8byte(64bit)値
file	ファイル(ディレクトリ)名文字列
date	2byte(16bit)のファイル日付データ
datetime	4byte(32bit)のファイル日付・時刻データ
data	USB デバイスと送受信する生データ (数値モードが ASCII であってもバイナリで送受信)

# 8 ライブラリ

---

HT1040 の RTC と USB コントローラ、TCP/IP コントローラをアクセスするための関数を用意しています。この章ではそれぞれの関数の仕様を説明します。なお W5100 については Wiznet が提供しているライブラリを HT1040 用に移植したものです。

## 8.1 ライブラリの使用方法

YellowIDE6 で HT1040 用ライブラリを使用する場合は、マニュアルディスクの ht1040lib¥YellowIDE6 ディレクトリにある HT1040.LIB をプロジェクトに追加してください。アプリケーションからライブラリの関数を利用する場合は、使用する関数のヘッダファイルをアプリケーションプログラムのソースでインクルードしてください。

USB コントローラを使用する関数(USB メモリのアクセス等)を使用する場合は、YellowIDE のプロジェクトの設定で以下の 2 つの割り込みベクタを設定する必要があります。

60	sci2error
61	sci2rx

HEW4 で HT1040 用ライブラリを使用する場合は、マニュアルディスクの ht1040lib¥HEW4 ディレクトリにある HT1040.LIB をプロジェクトに追加してください。アプリケーションからライブラリの関数を利用する場合は、使用する関数のヘッダファイルをアプリケーションプログラムのソースでインクルードしてください。

その他の処理系の場合は ht1040lib¥src ディレクトリにソースファイルが用意されていますので、処理系にあわせた修正を行ってご利用ください。

## 8.2 時刻取得・設定

### 8.2.1 rtc\_gettime

#include "rtc.h"	
フォーマット	void rtc_gettime(struct tm *time);
入力	struct tm *time <time.h>で定義される tm 構造体へのポインタ
関数の戻り値	なし
動作	tm 構造体ポインタで指定された領域に日付時刻を取得します。 なお、tm.yday には 1 月 1 日からの日数ではなく、RTC のレジスタ E(下位バイト)、レジスタ F(上位バイト)の値が入ります。

## 8.2.2 rtc\_settime

```
#include "rtc.h"
フォーマット void rtc_settime(struct tm *time);
入力 struct tm *time
<time.h>で定義される tm 構造体へのポインタ
関数の戻り値 なし
動作 入力で指定された tm 構造体ポインタで指定された日付時刻を
RTC に設定します。
なお、tm.yday には RTC のレジスタ E(下位バイト)、レジスタ F(上
位バイト)へ書き込む値を設定してください。
```

## 8.3 TCP/IP

### 8.3.1 getmac\_from\_flash

```
#include "getmac.h"
フォーマット unsigned char *getmac_from_flash(unsigned char *mac);
入力 unsigned char *mac MAC アドレスを取得する配列
関数の戻り値 unsigned char * MAC アドレス取得配列
動作 指定された配列に MAC アドレスを取得します。
```

### 8.3.2 socket

```
#include "socket.h"
フォーマット uint8 socket(SOCKET s, uint8 protocol, uint16 port, uint8 flag);
入力 SOCKET s 使用するソケット番号(0 から 3)を指定
uint8 protocol プロトコルを指定
指定可能なプロトコル(w5100.h で定義)
Sn_MR_TCP TCP/IP
Sn_MR_UDP UDP
Sn_MR_IPRAW IP LAYER RAW SOCK
Sn_MR_MACRAW MAC RAW
Sn_MR_PPPOE PPPoE
uint16 port ポート番号を指定
uint8 flag ソケットのオプションを指定
指定可能なオプション (w5100.h で定義)
Sn_MR_ND 遅延 Ack を使用しない
Sn_MR_MULTI マルチキャスト
関数の戻り値 uint8 成功した場合は 1、失敗した場合は 0
動作 指定された引数でソケットを初期化します。
```

### 8.3.3 sclose

```
#include "socket.h"
フォーマット void sclose(SOCKET s);
入力 SOCKET s ソケット番号(0 から 3)を指定
関数の戻り値 なし
動作 指定されたソケットを強制的に切断します。この関数を使用した場合は、相手側に FIN メッセージは送信されません。
```

### 8.3.4 connect

```
#include "socket.h"
フォーマット uint8 connect(SOCKET s, uint8 * addr, uint16 port);
入力 SOCKET s 使用するソケット番号(0 から 3)を指定
uint8 *addr 接続先のサーバの IP アドレスを指定
例えば 192.168.11.8 が接続先の場合、
配列 uint8 destip[]={192,168,11,8};を用意し destip を
引数として指定します。
uint16 port 接続先のポート番号を指定
関数の戻り値 uint8 成功した場合は 1、失敗した場合は 0
動作 TCP クライアントとして指定されたサーバに接続します。
```

### 8.3.5 disconnect

```
#include "socket.h"
フォーマット void disconnect(SOCKET s);
入力 SOCKET s ソケット番号(0 から 3)を指定
関数の戻り値 なし
動作 相手側に FIN メッセージを送信して指定されたソケット切断します。
```

### 8.3.6 listen

```
#include "socket.h"
フォーマット uint8 listen(SOCKET s);
入力 SOCKET s ソケット番号(0 から 3)を指定
関数の戻り値 uint8 成功した場合は 1、失敗した場合は 0
動作 指定されたソケットを TCP サーバとして接続待ち状態とします。
```

## 8.3.7 send

```
#include "socket.h"
uint16 send(SOCKET s, const uint8 * buf, uint16 len);
入力      SOCKET s      ソケット番号(0 から 3)を指定
          const uint8 *buf  送信データバッファポインタ
          uint16 len      送信データ長
関数の戻り値 uint16  成功した場合は送信したデータ長、失敗した場合は 0
動作      TCP モードで指定されたソケットにデータを送信します。
```

## 8.3.8 recv

```
#include "socket.h"
uint16 recv(SOCKET s, uint8 * buf, uint16 len);
入力      SOCKET s      ソケット番号(0 から 3)を指定
          uint8 *buf      受信データバッファポインタ
          uint16 len      受信データ長
関数の戻り値 uint16  受信したデータ長
動作      TCP モードで指定されたソケットからデータを受信します。
          この関数は受信データ待ちをしませんので、あらかじめ受信データの存在とデータ長を確認してからこの関数を使用してください。
```

## 8.3.9 sendto

```
#include "socket.h"
uint16 sendto(SOCKET s, const uint8 * buf, uint16 len,
              uint8 * addr, uint16 port);
入力      SOCKET s      ソケット番号(0 から 3)を指定
          const uint8 *buf  送信データバッファポインタ
          uint16 len      送信データ長
          uint8 *addr      送信先 IP アドレスを指定
                          例えば 192.168.11.8 が接続先の場合、
                          配列 uint8 destip[]={192,168,11,8};を用意し destip を
                          引数として指定します。
          uint16 port      送信先のポート番号を指定
関数の戻り値 uint16  成功した場合は送信データ長、失敗した場合は 0
動作      UDP または IPRAW モードで指定されたあて先にデータを送信
          します。
```





## 8.3.17 putISR

```
#include "w5100.h"
```

フォーマット	void putISR(uint8 s, uint8 val);
入力	uint8 s ソケット番号(0 から 3)を指定 uint8 val I_STATUS[s]への設定値
関数の戻り値	なし
動作	指定されたソケット番号 s の I_STATUS[s]変数に val を設定しま す。

## 8.3.18 getIINCHIP\_RxMAX

```
#include "w5100.h"
```

フォーマット	uint16 getIINCHIP_RxMAX(uint8 s);
入力	uint8 s ソケット番号(0 から 3)を指定
関数の戻り値	uint16 指定されたソケット番号の RSIZE[s]変数の値
動作	指定されたソケット番号 s の RSIZE[s]変数の値を返します。この 変数にはソケットが使用できる受信バッファサイズが保持されて います。(RSIZE[s]の値は sysinit 関数で設定されます。)

## 8.3.19 getIINCHIP\_TxMAX

```
#include "w5100.h"
```

フォーマット	uint16 getIINCHIP_TxMAX(uint8 s);
入力	uint8 s ソケット番号(0 から 3)を指定
関数の戻り値	uint16 指定されたソケット番号の SSIZE[s]変数の値
動作	指定されたソケット番号 s の SSIZE[s]変数の値を返します。この 変数にはソケットが使用できる送信バッファサイズが保持されて います。(SSIZE[s]の値は sysinit 関数で設定されます。)

## 8.3.20 getIINCHIP\_RxMASK

```
#include "w5100.h"
```

フォーマット	uint16 getIINCHIP_RxMASK(uint8 s);
入力	uint8 s ソケット番号(0 から 3)を指定
関数の戻り値	uint16 指定されたソケット番号の RMASK[s]変数の値
動作	指定されたソケット番号 s の RMASK[s]変数の値を返します。こ の変数にはソケットが使用する受信バッファのロール処理用マス クサイズが保持されています。(RMASK[s]の値は sysinit 関数で設 定されます。)

## 8.3.21 getIINCHIP\_TxMASK

```
#include "w5100.h"
```

フォーマット	uint16 getIINCHIP_TxMASK(uint8 s);
入力	uint8 s ソケット番号(0 から 3)を指定
関数の戻り値	uint16 指定されたソケット番号の SMASK[s]変数の値
動作	指定されたソケット番号 s の SMASK[s]変数の値を返します。この変数にはソケットが使用する送信バッファのロール処理用マスクサイズが保持されています。(SMASK[s]の値は sysinit 関数で設定されます。)

## 8.3.22 getIINCHIP\_RxBASE

```
#include "w5100.h"
```

フォーマット	uint16 getIINCHIP_RxBASE(uint8 s);
入力	uint8 s ソケット番号(0 から 3)を指定
関数の戻り値	uint16 指定されたソケット番号の RBUFBASEADDRESS[s]変数の値
動作	指定されたソケット番号 s の RBUFBASEADDRESS[s]変数の値を返します。この変数にはソケットが使用する受信バッファのベースアドレスが保持されています。(RBUFBASEADDRESS [s]の値は sysinit 関数で設定されます。)

## 8.3.23 getIINCHIP\_TxBASE

```
#include "w5100.h"
```

フォーマット	uint16 getIINCHIP_TxBASE(uint8 s);
入力	uint8 s ソケット番号(0 から 3)を指定
関数の戻り値	uint16 指定されたソケット番号の SBUFBASEADDRESS[s]変数の値
動作	指定されたソケット番号 s の SBUFBASEADDRESS[s]変数の値を返します。この変数にはソケットが使用する送信バッファのベースアドレスが保持されています。(SBUFBASEADDRESS [s]の値は sysinit 関数で設定されます。)

## 8.3.24 setGAR

```
#include "w5100.h"
```

フォーマット	void setGAR(uint8 * addr);
入力	uint8 *addr ゲートウェイアドレスを保持した配列を指定
関数の戻り値	なし
動作	引数で指定されたゲートウェイアドレスを W5100 の GAR レジスタに設定します。

## 8.3.25 setSUBR

```
#include "w5100.h"
フォーマット void setSUBR(uint8 * addr);
入力 uint8 *addr サブネットマスクを保持した配列を指定
関数の戻り値 なし
動作 引数で指定されたサブネットマスクを W5100 の SUBR レジスタ
に設定します。
```

## 8.3.26 setSHAR

```
#include "w5100.h"
フォーマット void setSHAR(uint8 * addr);
入力 uint8 *addr MAC アドレスを保持した配列を指定
関数の戻り値 なし
動作 引数で指定された MAC アドレスを W5100 の SHAR レジスタ
に設定します。
```

## 8.3.27 setSIPR

```
#include "w5100.h"
フォーマット void setSIPR(uint8 * addr);
入力 uint8 *addr ソース IP アドレスを保持した配列を指定
関数の戻り値 なし
動作 引数で指定された IP アドレスを W5100 の SIPR レジスタに
設定します。
```

## 8.3.28 setRTR

```
#include "w5100.h"
フォーマット void setRTR(uint16 timeout);
入力 uint16 timeout リトライタイム設定値
関数の戻り値 なし
動作 引数で指定されたタイムアウト設定値を RTR レジスタに設定しま
す。設定値については6.10.9項を参照してください。
```

## 8.3.29 setRCR

```
#include "w5100.h"
フォーマット void setRCR(uint8 retry);
入力 uint8 retry リトライ回数
関数の戻り値 なし
```

動作 引数で指定されたリトライ回数を RTC レジスタに設定します。  
リトライ回数がこの設定値を超えた場合タイムアウト割り込みが発生します。

### 8.3.30 setIMR

```
#include "w5100.h"
フォーマット void setIMR(uint8 mask);
入力 uint8 mask 割り込みマスクレジスタ設定値
関数の戻り値 なし
動作 引数で指定された値を IMR レジスタに設定します。0 を設定したビットに対応する要因の割り込みはマスクされます。IMR については6.10.8項を参照してください。
```

### 8.3.31 getGAR

```
#include "w5100.h"
フォーマット void getGAR(uint8 * addr);
入力 uint8 *addr ゲートウェイアドレスを取得する配列を指定
関数の戻り値 なし
動作 引数で指定された配列に W5100 の GAR レジスタからゲートウェイアドレスを取得します。
```

### 8.3.32 getSUBR

```
#include "w5100.h"
フォーマット void getSUBR(uint8 * addr);
入力 uint8 *addr サブネットマスクを取得する配列を指定
関数の戻り値 なし
動作 引数で指定された配列に W5100 の SUBR レジスタからサブネットマスクを取得します。
```

### 8.3.33 getSHAR

```
#include "w5100.h"
フォーマット void getSHAR(uint8 * addr);
入力 uint8 *addr MAC アドレスを取得する配列を指定
関数の戻り値 なし
動作 引数で指定された配列に W5100 の SHAR レジスタから MAC アドレスを取得します。
```

### 8.3.34 getSIPR

```
#include "w5100.h"
void getSIPR(uint8 * addr);
入力      uint8 *addr      ソース IP アドレスを取得する配列を指定
関数の戻り値  なし
動作      引数で指定された配列に W5100 の SIPR レジスタから IP アドレスを取得します。
```

### 8.3.35 getIR

```
#include "w5100.h"
void getIR(void);
入力      なし
関数の戻り値  uint8   IR レジスタの内容
動作      W5100 の IR レジスタの値を返します。
```

### 8.3.36 setSn\_MSS

```
#include "w5100.h"
void setSn_MSS(SOCKET s, uint16 Sn_MSS);
入力      SOCKET s      ソケット番号(0 から 3)を指定
          uint16Sn_MSS  Sn_MSS 設定値
関数の戻り値  なし
動作      指定されたソケットの Sn_MSS レジスタに値を設定します。
          SnMSS レジスタは最大セグメントサイズを設定します。
```

### 8.3.37 setSn\_PROTO

```
#include "w5100.h"
void setSn_PROTO(SOCKET s, uint8 proto);
入力      SOCKET s      ソケット番号(0 から 3)を指定
          uint8 proto    プロトコル番号設定値
関数の戻り値  なし
動作      指定されたソケットの Sn_PROTO レジスタに値を設定します。
          SnPROTO レジスタは IPRAW モードで IP ヘッダのプロトコル番号フィールドを設定します。
```

### 8.3.38 getSn\_IR

```
#include "w5100.h"
```

フォーマット	uint8	setSn_IR(SOCKET s);
入力	SOCKET s	ソケット番号(0 から 3)を指定
関数の戻り値	uint8	Sn_IR レジスタ内容
動作		指定されたソケットの Sn_IR レジスタの値を返します。

### 8.3.39 getSn\_SR

```
#include "w5100.h"
```

フォーマット	uint8	setSn_SR(SOCKET s);
入力	SOCKET s	ソケット番号(0 から 3)を指定
関数の戻り値	uint8	Sn_SR レジスタ内容
動作		指定されたソケットの Sn_SR レジスタの値を返します。

### 8.3.40 getSn\_TX\_FSR

```
#include "w5100.h"
```

フォーマット	uint8	setSn_TX_FSR (SOCKET s);
入力	SOCKET s	ソケット番号(0 から 3)を指定
関数の戻り値	uint8	Sn_TX_FSR レジスタ内容
動作		指定されたソケットの Sn_TX_FSR レジスタの値を返します。

### 8.3.41 getSn\_RX\_FSR

```
#include "w5100.h"
```

フォーマット	uint8	setSn_RX_FSR (SOCKET s);
入力	SOCKET s	ソケット番号(0 から 3)を指定
関数の戻り値	uint8	Sn_RX_FSR レジスタ内容
動作		指定されたソケットの Sn_RX_FSR レジスタの値を返します。

### 8.3.42 setSn\_DHAR

```
#include "w5100.h"
```

フォーマット	void	setSn_DHAR (SOCKET s, , uint8 * addr);
入力	SOCKET s	ソケット番号(0 から 3)を指定
	uint8 *addr	あて先 MAC アドレスを保持した配列を指定
関数の戻り値	なし	
動作		引数で指定された MAC アドレスを指定されたソケットの Sn_DHAR レジスタに設定します。

### 8.3.43 setSn\_DIPR

```
#include "w5100.h"
```

フォーマット	void setSn_DIPR(SOCKET s, uint8 * addr);
入力	SOCKET s           ソケット番号(0 から 3)を指定 uint8 *addr        あて先 IP アドレスを保持した配列を指定
関数の戻り値	なし
動作	引数で指定された IP アドレスを指定されたソケットの Sn_DIPR レジスタに設定します。

### 8.3.44 setSn\_DPORT

```
#include "w5100.h"
```

フォーマット	void setSn_DPORT(SOCKET s, uint8 * addr);
入力	SOCKET s           ソケット番号(0 から 3)を指定 uint8 *addr        あて先ポート番号を保持した配列を指定
関数の戻り値	なし
動作	引数で指定されたポート番号を指定されたソケットの Sn_DPORT レジスタに設定します。

### 8.3.45 setSn\_TTL

```
#include "w5100.h"
```

フォーマット	void setSn_TTL(SOCKET s, uint8 ttl);
入力	SOCKET s           ソケット番号(0 から 3)を指定 uint8 ttl           TTL 設定値
関数の戻り値	なし
動作	TTL 値を指定されたソケットの Sn_TTL レジスタに設定します。

### 8.3.46 setMR

```
#include "w5100.h"
```

フォーマット	void setMR(uint8 val);
入力	uint8 val           MR レジスタ設定値
関数の戻り値	なし
動作	MR レジスタに val を設定します。

## 8.4 USB コントローラ(ファイル操作)

### 8.4.1 usb\_init

```
#include "usb.h"
フォーマット int usb_init(void);
入力 なし
関数の戻り値 int 成功した場合は 0、失敗した場合は-1
動作 USB コントローラ関連のハードウェアを初期化します。
```

### 8.4.2 usb\_send

```
#include "usb.h" (#include "sci2io.h")
フォーマット void usb_send(char *s);
入力 char *s 送信する文字列
関数の戻り値 なし
動作 USB コントローラに指定された文字列を送信します。
```

### 8.4.3 usb\_open

```
#include "usb.h"
フォーマット int usb_open(char *filename, int oflag);
入力 char *filename オープンするファイル名
int oflag リード(O_RDONLY)またはライト(O_WRONLY)を指定
関数の戻り値 int 成功した場合はハンドル番号、失敗した場合は-1
動作 USB メモリで指定されたファイル名をリードまたはライト用にオープンします。
```

### 8.4.4 usb\_close

```
#include "usb.h"
フォーマット int usb_close(int handle);
入力 int handle オープンされているファイルのハンドル
int oflag リード(O_RDONLY)またはライト(O_WRONLY)を指定
関数の戻り値 int 成功した場合は 0、失敗した場合は-1
動作 指定されたハンドルのファイルをクローズします。
```

## 8.4.5 usb\_lseek

#include "usb.h"

フォーマット	long	usb_lseek(int handle, long offset, int origin);
入力	int handle	オープンされているファイルのハンドル
	long offset	origin からのオフセット
	int origin	基準点 以下の定数が使用可能
		SEEK_SET      ファイル先頭
		SEEK_CUR      現在の位置
		SEEK_END      ファイル末尾
関数の戻り値	long	成功した場合は新しいファイル読み出し位置、失敗した場合は-1 を返します。
動作		指定されたハンドルのファイルポインタを origin で指定された基準点から offset で指定された位置に移動します。

## 8.4.6 usb\_fprintf

#include "usb.h"

フォーマット	int	usb_fprintf(int handle, const char *format [,arg]...);
入力	int handle	オープンされているファイルのハンドル
	const char *format	フォーマット制御
	arg	オプションの引数(フォーマット制御に依存)
関数の戻り値	int	成功した場合は出力した文字数を、失敗した場合は-1 を返します。
動作		ハンドルで指定されたファイルに、フォーマット制御にしたがって引数の値を書き込みます。

## 8.4.7 usb\_read

#include "usb.h"

フォーマット	int	usb_read(int handle, void *buffer, size_t count);
入力	int handle	オープンされているファイルのハンドル
	void *buffer	データ取得バッファ
	size_t count	読み出しバイト数
関数の戻り値	int	成功した場合は取得したデータのバイト数、失敗した場合は-1 を返します。
動作		ハンドルで指定されたファイルから、count で指定されたバイト数のデータを読み出します。

## 8.4.8 usb\_bread

```
#include "usb.h"
int usb_bread(int handle, void *buffer, size_t count);
```

フォーマット  
入力

int handle	オープンされているファイルのハンドル
void *buffer	データ取得バッファ
size_t count	読み出しバイト数

関数の戻り値

int	成功した場合は取得したデータのバイト数、失敗した場合は-1を返します。
-----	-------------------------------------

動作

ハンドルで指定されたファイルから、countで指定されたバイト数のデータを読み出します。usb\_breadは関数内部に読み出しバッファ512バイトを持っている点が異なります。usb\_fgetsやusb\_fgetc関数は下位の関数としてusb\_breadを使用しています。

## 8.4.9 usb\_write

```
#include "usb.h"
int usb_write(int handle, void *buffer, size_t count);
```

フォーマット  
入力

int handle	オープンされているファイルのハンドル
void *buffer	データ取得バッファ
size_t count	読み出しバイト数

関数の戻り値

int	成功した場合は取得したデータのバイト数、失敗した場合は-1を返します。
-----	-------------------------------------

動作

ハンドルで指定されたファイルから、countで指定されたバイト数のデータを読み出します。

## 8.4.10 usb\_fgetc

```
#include "usb.h"
int usb_fgetc(int handle);
```

フォーマット  
入力

int handle	オープンされているファイルのハンドル
------------	--------------------

関数の戻り値

int	成功した場合は取得した文字、失敗した場合は-1を返します。
-----	-------------------------------

動作

ハンドルで指定されたファイルから、1文字を読み出します。

## 8.4.11 usb\_fgets

```
#include "usb.h"
char *usb_fgets(char *s, int n, int handle);
```

フォーマット  
入力

char *s	データ取得バッファポインタ
int n	最大の読み出しバイト数
int handle	オープンされているファイルのハンドル

関数の戻り値

char *	成功した場合は取得したバッファへのポインタ、失敗した場合はNULLを返します。
--------	---

動作 ハンドルで指定されたファイルから、改行文字に達するか、ファイル末尾に達するか、n で指定された最大バイト数になるまでバッファにデータを読み出します。読み出したデータには¥x0a が含まれ、最後に ¥0 が付加されます。

## 8.4.12 usb\_fputc

```
#include "usb.h"
フォーマット int usb_fputc(char c, int handle);
入力 char c 出力文字
int handle オープンされているファイルのハンドル
関数の戻り値 int 成功した場合は出力した文字、失敗した場合は
EOF を返します。
動作 ハンドルで指定されたファイルにから、文字 c を出力します。
```

## 8.4.13 usb\_filelength

```
#include "usb.h"
フォーマット unsigned long usb_filelength(int handle);
入力 int handle オープンされているファイルのハンドル
関数の戻り値 unsigned long 成功した場合はファイルサイズ、失敗した場合は
0xFFFFFFFF を返します。
動作 ハンドルで指定されたファイルのサイズを取得します。
```

## 8.4.14 usb\_mkdir

```
#include "usb.h"
フォーマット int usb_mkdir(char *dirname);
入力 char *dirname 作成するディレクトリ名の文字列
関数の戻り値 int 成功した場合は 0、失敗した場合は -1 を返します。
動作 dirname で指定されたサブディレクトリを作成します。
```

## 8.4.15 usb\_chdir

```
#include "usb.h"
フォーマット int usb_chdir(char *dirname);
入力 char *dirname 移動するサブディレクトリ名の文字列
関数の戻り値 int 成功した場合は 0、失敗した場合は -1 を返します。
動作 dirname で指定されたサブディレクトリに移動します。移動できるディレクトリは現在のディレクトリ内にあるサブディレクトリが、1 つ上の階層(ディレクトリ名 ".." で指定)に限られます。
```

## 8.4.16 usb\_remove

```
#include "usb.h"
```

フォーマット     int usb\_remove(char \*filename);  
入力             char \*filename     削除するファイル名の文字列  
関数の戻り値     int                成功した場合は0、失敗した場合は-1 を返します。  
動作             filename で指定されたファイルを作成します。

## 8.4.17 vnc1lcmd\_exec

```
#include "usb.h"
```

フォーマット     int vnc1lcmd\_exec(unsigned char \*cmd, unsigned char \*param);  
入力             unsigned char \*cmd  実行するコマンド文字列  
                  unsigned char \*param コマンド引数文字列  
関数の戻り値     int                成功した場合は0、失敗した場合は-1 を返します。  
動作             cmd で指定されコマンドと param で指定された引数を uec コント  
                  ローラに送信し、結果文字列を取得して成功または失敗を報告しま  
                  す。成功した場合は usberra 変数に 0 が、失敗した場合は 0 以外  
                  の値がセットされます。usberra で返される値は以下の通りです。

UNOERR	0	成功
UENOENT	3	コマンド実行に失敗
UEBADF	4	ファイルがオープンされて いて実行できないコマンド
UENOSPC	5	ディスクに空きがない
UEINVAL	6	ファイル名が不正
UEBADCMD	7	コマンドが不正
UENODSK	8	ディスクが存在しない
UEUNKWN	9	不明なエラー

## 8.4.18 usb\_sync

```
#include "usb.h"
```

フォーマット     void usb\_sync(void);  
入力             なし  
関数の戻り値     なし  
動作             VNC1L からの受信バッファを空にします。VNC1L にコマンドを  
                  送信する前に、受信バッファに読み出ししていない文字がある場合、送信したコマン  
                  ドに対する応答かどうかの判断ができなくなりますので、このコマンドで未読み出し  
                  のデータを破棄します。

## 8.4.19 get\_reply

```
#include "usb.h"
```

フォーマット int get\_reply(char \*s, unsigned timeout);

入力 char \*s 応答文字列取得バッファポインタ

unsigned timeout タイムアウト秒数

関数の戻り値 int 応答受信に成功した場合は受信文字列長、タイムアウトの場合は-1

動作 VNC1L から¥0d で区切られるコマンド応答を受信します。何らかの理由で受信応答がない場合、指定された秒数でタイムアウトします。

## 8.4.20 wait\_reply

```
#include "usb.h"
```

フォーマット int wait\_reply(unsigned timeout);

入力 unsigned timeout タイムアウト秒数

関数の戻り値 int 正常にプロンプトを受信した場合は 0、それ以外の場合は-1

動作 VNC1L からコマンド応答を受信し、正常なプロンプト応答の場合は 0、それ以外の場合は-1 を返します。エラーの場合は usberra に 0 以外の値がセットされます。usberra の値については8.4.17項をご参照ください。

## 8.4.21 goto\_datamode

```
#include "usb.h"
```

フォーマット int goto\_datamode(void);

入力 なし

関数の戻り値 int VNC1L がデータモードに移行した場合は 1、タイムアウトの場合は 0

動作 VNC1L をコマンドモードからデータモードに変更します。2 秒以内にデータモードに移行できない場合はタイムアウトします。

## 8.4.22 goto\_cmdmode

```
#include "usb.h"
```

フォーマット int goto\_cmdmode(void);

入力 なし

関数の戻り値 int VNC1L がコマンドモードに移行した場合は 1、タイムアウトの場合は 0

動作 VNC1L をデータモードからコマンドモードに変更します。2 秒以内にコマンドモードに移行できない場合はタイムアウトします。

## 8.4.23 usb\_wait1ms

#include "usb.h"

フォーマット      void usb\_wait1ms(int m);

入力                int m      ウェイトする時間(ms)

関数の戻り値      なし

動作                m で指定された時間待ちをします。

## 9 サンプルプログラム

---

マニュアルディスクの ht1040sample フォルダには、ライブラリを使用した簡単なサンプルプログラムを YellowIDE6 で使用できるプロジェクト形式と HEW4 のワークスペース形式で用意しています。YellowIDE6 のプロジェクトは RAM にロードして実行する形式になっています。S ファイルローダから実行する場合は、拡張子が.S のオブジェクトファイルを L コマンドで読み込み、G コマンドで 0x400000 番地から実行してください。HEW4 のワークスペースでは Debug 用のビルドで作成された拡張子.mot のオブジェクトが S ファイルローダで読み込み、実行できます。HEW4 で作成された USB を使用するサンプルプログラムを実行する場合は、G コマンドの前に VTE コマンドを実行してください。

Flash メモリに書き込みする場合、YellowIDE6 ではプロジェクトウィンドウで ROM 化を選択してプロジェクトの再構築したオブジェクトを、HEW4 では Release でビルドしたオブジェクトを使用してください。

### 9.1 ファイル作成

フォルダ:fileop

USB メモリにファイルを作成するサンプルです。プログラムを実行すると、接続されている USB メモリに sample.txt という名称のファイルを書き込みモードでオープンし、0 から 9 までの値をテキストで書き込みクローズします。その後このファイルを読み出し、コンソールに表示します。数種類の読み出し方法をテストしています。

### 9.2 時刻設定・取得

フォルダ:rtcrw

RTC の読み出し、時刻の設定サンプルです。プログラムを実行すると、RTC から日付と時刻を読み出してコンソールに表示します。その後日付と時刻を入力し、RTC に設定したあと再度 RTC から日付と時刻を読み出し表示します。

### 9.3 HTTP アクセス

フォルダ:portrw

ブラウザからのアクセスにตอบสนองする TCP/IP 通信のサンプルです。プログラムを実行すると HT1040 の IP アドレス、ネットマスク等を設定しポート 80 をオープンしてブラウザからの接続待ち状態となります。HT1040 のポート 4 は出力に、ポート A は入力に設定されます。

ブラウザから HT1040 に割り当てたアドレスを指定すると、ポート 4、ポート A の状態を表示します。ポート 4 は値を変更しボタンを押すと出力状態を変更することができます。

IP アドレス等はプログラム中に固定値が書き込まれていますので、使用環境にあわせて変更してください。

## 9.4 TCP クライアント

フォルダ:tcpclient

ソケットを使ってサーバにデータを送信する TCP クライアントのサンプルです。

接続先の IP アドレスとポートを設定すると、15 秒に一度サーバに日付時刻文字列を送信します。ソース IP はプログラム中に固定値で設定してありますので使用環境に合わせて変更してください。

サーバプログラムとして TCPServ.exe ファイルをプロジェクトフォルダに用意しています。ポートを指定して OPEN ボタンを押すと、サーバソケットをオープンして接続待ち状態となります。クライアントから接続すると、接続 IP アドレスと送信されたテキスト文字列をウィンドウに表示します。

なおこのサーバプログラムを PC で実行する場合は、Windows のファイアウォールや、インターネットセキュリティソフトウェアを設定して使用するポートへの外部からの接続ブロックを解除する必要があります。

## 9.5 USB マウス

フォルダ:usbmouse

USB マウスからデータを取得するサンプルです。プログラムを実行すると、マウスの移動やボタンの状態に応じたデータを取得しコンソールに表示します。データの内容については USB2.0 の HID デバイスの規格([www.usb.org](http://www.usb.org) から入手可能)等をご参照ください。

## 9.6 FT232

フォルダ:usbft232

FTDI の USB 変換デバイス(FT232 等)と HT1040 を接続した場合は、USB を意識せず接続先の機器のシリアルポートと通信することができます。このプログラムを実行すると接続されている FTDI のデバイスを 8bit、パリティ無し、1STOP ビットで指定のボーレートで通信するように設定します。HT1040 のコンソールからの入力は接続先の TxD へ、接続先の RxD に入力されたデータはコンソールに表示されます。FT232 のフロー制御は CTS-RTS を仮定しています。

## 9.7 USB 接続プリンタ

フォルダ:usbprn

USB 接続可能なプリンタで、プリント制御コマンド仕様が公開されているものは HT1040 から制御することが可能です。(多くのターミナルプリンタは ESC/P をサポートしています。)このプログラムでは入力された文字列を USB 接続されたプリンタに送信し、最後に CR/LF を送信します。プログラムは"."入力で終了します。

## 9.8 USB キーボード

フォルダ:usbkey

USB 接続可能なキーボードを接続してキーコードを取得することができます。このプログラムは USB キーボードから 8 バイトの入力レポートを取得し、表示します。また取得したキーが CAPSLOCK+SHIFT の場合は CapsLock の LED を点灯・消灯、NUMLOCK キーの場合は Numlock の LED を点灯・消灯します。

キーコードについては USB HID Usage Tables([www.usb.org](http://www.usb.org) から入手可能)をご参照ください。

## 9.9 キャラクタ LCD 表示器

フォルダ:lcd44780

16(20)桁 2 行のキャラクタ LCD 表示器に文字表示するサンプルです。接続には HT1040 の汎用 I/O ポート P4[0:5]を使用しています。接続回路例は図 9-1をご参照ください。(LCD モジュールの端子番号はメーカーにより異なる場合があります。)プログラムを実行すると、表示器に起動メッセージと RTC から読み出した日付を表示します。その後はコンソールから入力された文字をスクロールしながら表示します。

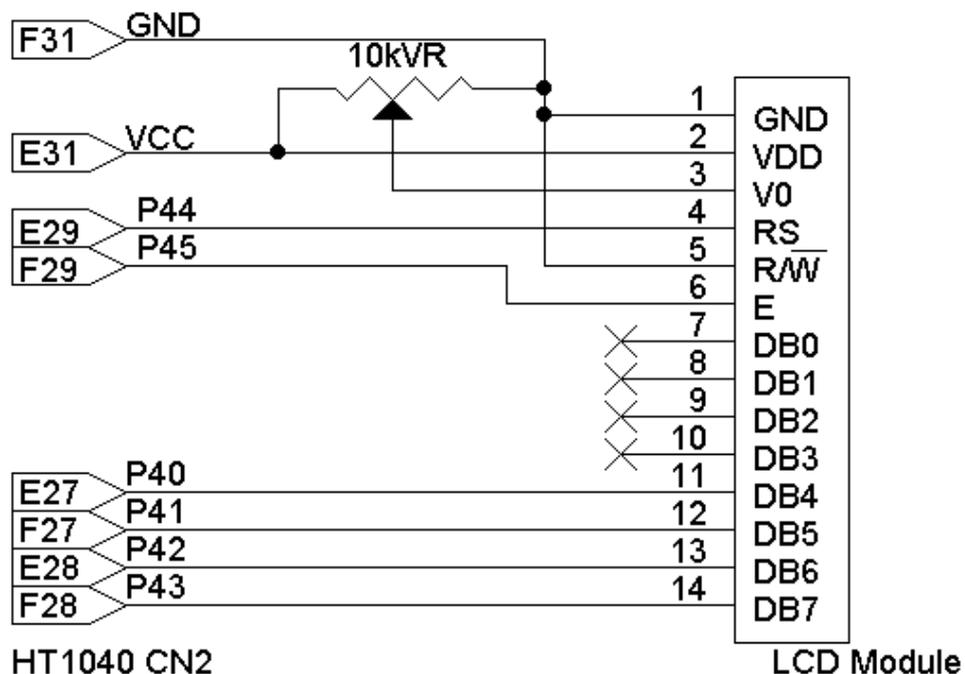


図 9-1 キャラクタ LCD 表示器接続例

# 10 内蔵 FLASH メモリ書き込み

---

この章では、H8/3069 内蔵 FLASH メモリ書き込みに関連する事項をまとめてあります。なおフラッシュメモリ書き込み操作の概略は4.7節をご参照ください。

## 10.1 モードスイッチ・ジャンパ設定

モードスイッチ(SW1)を BOOT 側に倒します。JP4 の設定は任意ですが、どちらかにジャンパソケットが挿入されている必要があります。この状態で電源を投入すると、H8/3069 はブートモード(モード 5)で起動します。

## 10.2 書き込みソフトウェア

FLASH メモリ書き込みプログラムは Windows95/98/Me/2000/Xp 用のアプリケーションです。配布されている HT1040WR.EXE にはインストーラやセットアッププログラムはありませんので、必要に応じてスタートメニューへの登録や、ショートカット作成を行ってください。HT1040WR.EXE はダブルクリックする等の方法で実行することができます。

プログラムを起動すると、HT1040 との通信に使用する COM ポート番号を指定するウィンドウが開きますので COM ポートを選択してください。次に File-Send メニューから書き込みするファイルを選択して Send ボタンを押してください。なお Send ボタンを押す前に HT1040 の電源を投入しておく必要があります。

拡張子が MOT および S のファイルは、内容を S フォーマットのデータとして書き込みます。また拡張子が BIN のファイルは、ファイル内容をバイナリイメージとしてそのまま書き込みます。

書き込みソフトウェアは、弊社提供のもののほか、H8/3069 のブートモードに対応している書き込みソフトウェアも使用できます。イエローソフト FWRITE2、Renesas FDT4.05、Renesas HTerm については動作を確認しています。

## 10.3 出荷時の FLASH メモリ内容

HT1040 出荷時の H8/3069 内蔵フラッシュメモリには、S ファイルローダプログラム (S フォーマットのファイルを PC から HT1040 の RAM に転送するツール)が書き込まれています。このプログラムはフラッシュメモリの内容を書き換えると失われますが、マニュアルディスクの LOADER ディレクトリに S ファイルローダのバイナリイメージファイル SLDR1040.BIN が用意されていますので、必要な場合はこのファイルをフラッシュメモリに書き込みしてください。

## 10.4 書き込みに使用するポート

HT1040 では SCI1 に接続されているコンソールのほか、CN7B の USB ポート経由でもフラッシュメモリの書き換えができます。PC には FTDI の VCP ドライバをインストールし、割り当てた仮想 COM ポートをフラッシュメモリ書き込みソフトで指定してください。なお COM ポート番号が大きいと書き込みプログラムが認識しない場合がありますので、COM ポート番号は 8 以下にしてください。

## 10.5 FLASH メモリ書き換え時の注意

FLASH メモリ内容書き換え中、/IOCHCHK(NMI)入力が完全に禁止されていませんので、この入力が与えられると書き込みプログラムが暴走してしまいます。FLASH メモリ書き換え中には HT1040 に /IOCHCHK 入力が加えられないようご注意ください。

# 11 ハードウェア回路増設例

## 11.1 82C55 増設例

図 11-1に 82C55 (uPD71055) を HT1040 に接続する例を示します。

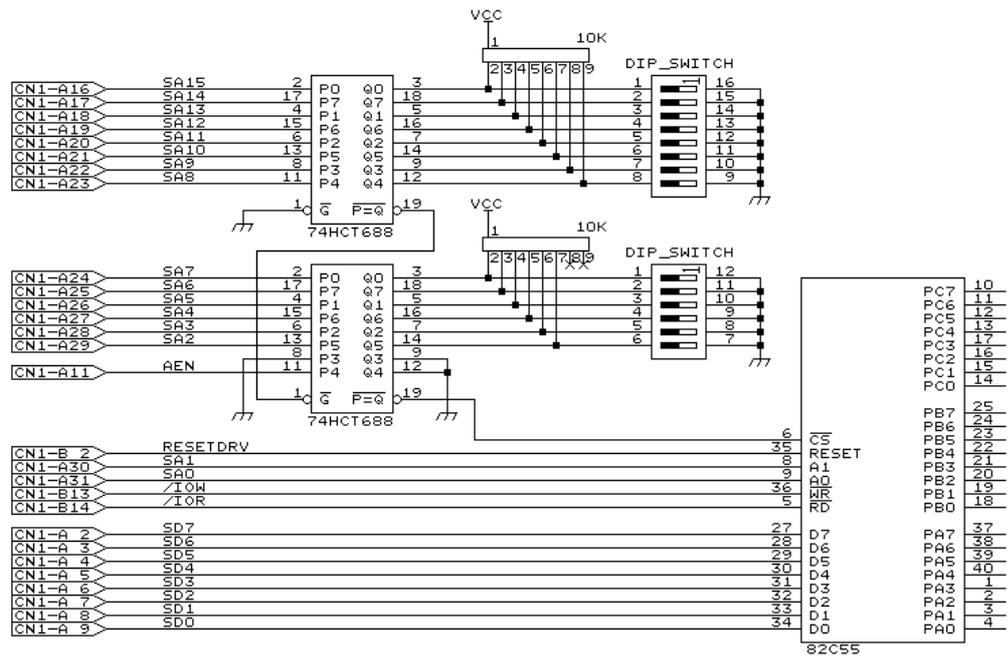


図 11-1 82C55 増設回路例

## 11.2 データバスバッファリング例

図 11-2にデータバスバッファの使用例を示します。本回路は 82C54 を接続する場合の参考例を兼ねています。

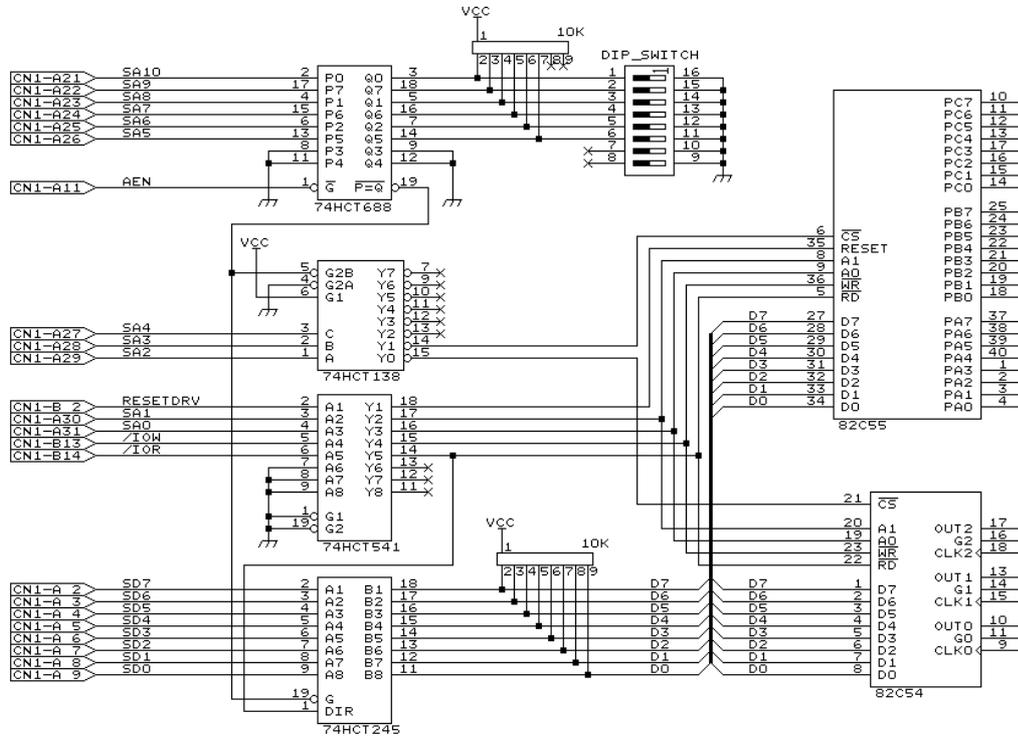


図 11-2 データバスバッファリング例

# 12 ICE 接続モデル

HT1040-U00 には H8 / 3069CPU がはんだ付けされているため、ICE 接続用に 100 ピン QFP ソケットを搭載したモデルを用意しています。なおこれらのモデルに CPU は付属しませんのでご注意ください。ICE メーカーによって採用しているプローブの適合ソケット製造メーカーが異なるため、2 タイプの ICE 接続用モデルがあります。ご使用になる ICE のプローブ仕様をご確認ください。

## 12.1 HT1040-SKA

ソケットに山一電機株式会社製 IC149-100-154-B5 を使用しています。このソケットに IC を搭載するためのカバーやネジは添付されています。HT1040-SKA に適合する ICE の例を表 12-1 に示します。なお各社製品仕様等が変更となっている場合もありますので、ご購入前に必ずマニュアルやメーカーにお問い合わせのうえプローブオプションや適合ソケットについて充分ご確認ください。

表 12-1 HT1040-SKA 適合インサーキットエミュレータ

メーカー	型式	プローブ
(株)ルネサステクノロジ	E6000 エミュレータ H8/300H	HS3064BECH61H
(株)日立超 LSI システムズ	MY-ICE H8/300H	MS3069UB5Q100
(株)コンピューテックス	Mr ICE2 H8/300H	EADP3052-100PB

## 12.2 HT1040-SKB

ソケットに東京エレテック株式会社製 NQPACK100SD を使用しています。このソケットに IC を搭載する場合は、HQPACK100SD が別途必要となりますのでご注意ください。なお製品仕様等が変更となっている場合もありますので、ご購入前に必ずマニュアルやメーカーにお問い合わせのうえプローブオプションや適合ソケットについて充分ご確認ください。

表 12-2 HT1040-SKB 適合インサーキットエミュレータ

メーカー	型式	プローブ
(株) ソフィアシステムズ	HyperSTAC for H8/300H	YQPACK100SD

# 13 ユーティリティリファレンス

この章では、HT1040 に付属するユーティリティの使用方法を説明します。

## 13.1 S ファイルローダ

### 【説明】

ホストパソコンで作成した S フォーマットのオブジェクトファイルを RAM へ転送するためのツールです。簡単なメモリ、I/O のテストをするためのコマンド等が用意されています。

このプログラムは、バイナリイメージで提供されていますので、必要に応じて内蔵 FLASH メモリに書き込みご使用ください。(出荷時の内蔵 FLASH メモリには書き込まれています。)

FLASH メモリ上の割り込みベクタテーブルを RAM 上にあるテーブルで置き換える機能がありますので、割り込みプログラムを RAM 上でテストすることもできます。

### 【使用メモリ】

ワークエリアとして、FFBF20 ~ FFEFFF までを使用します。また、スタックポインタは FFDFFF に初期化しています。

### 【シリアル通信条件】

38400bps, 8bit, No Parity, 1 stop bit, X 制御あり

ターミナルソフトウェアではローカルエコーなしに設定してください。

表 13-1 S ファイルローダのコマンド

コマンド名	機能
D/DB/DW/DL	メモリ内容をダンプ表示します。
E/EB/EW/EL	メモリ内容を変更します。
L	S フォーマットファイルをロードします。
I/IB/IW/IL	CN1 I/O 空間のポートを読み出し表示します。
O/OB/OW/OL	CN1 I/O 空間のポートヘデータ書き込みします。
G	指定アドレスへ実行制御を移します。
NET	TCP/IP 関連コマンドモードへ移行します。
DISK	USB メモリ制御コマンドモードへ移行します。
RTC	日付時刻表示、設定を行います。
USB	USB コントローラのコマンドモードへ移行します。
VTE/VTY	割り込みベクタテーブルのモードを変更します。
?	コマンド一覧表示

### 【数値データタイプ】

各コマンドの引数あるいは設定値として入力する数値には、次の 3 種類の表現を用いることができます。

・ 16 進数

通常、入力された数値は全て 16 進数として解釈されます。  
また、数値の前に \$ をつけた場合も 16 進数として扱われます。

【例】 3FFC  
\$2F

・ 10 進数

数値の前に # をつけた場合は、それに続く数値は 10 進数として扱われます。数値は正数のみ有効で、± 符号はつけることができません。

【例】 #25  
#3500

・ 2 進数

数値の前に @ をつけた場合は、それに続く数値は 2 進数として扱われます。

【例】 @10101111  
@0100

【コマンド入力方法】

コマンド待ち状態になるとマイナス(-)をプロンプト記号として出力します。投入可能なコマンド名は表 13-1 をご参照ください。引数を与える場合、コマンド名と引数、引数どうしは 1 つ以上のカンマ(,)あるいは空白で区切ります。これ以外のもの(セミコロンやタブ等)は区切り符号として使用できません。ただし、コマンド名と第一引数は区切らずに入力可能ですが、コマンド名が優先的に検出されるため注意が必要です。

送出文字は、バックスペースを送信することで 1 字ずつ訂正することができます。また、途中でその行をすべて取り消したい場合にはエスケープキーによって 1 行キャンセルすることができます。

コマンド入力および引数は大文字・小文字どちらでも受け付けられます。ただし、ダブルクォートで囲まれた文字列中では区別されています。

《注意》

複数個の引数が必要な場合、途中の引数を省略して後の引数のみを与えることはできませんのでご注意ください。

【コマンド記述に用いた表現】

以下の項には各コマンドについてその機能、引数等が説明されていますが、ここではその入力フォーマット説明中に用いている記法について説明します。

<引数>

不等号記号で囲まれた項目は、それがコマンドの引数であることを示します。実際にコマンドを投入する場合には<>をつける必要はありません。引数が数値である場合は、上述のデータタイプに従って解釈されます。

/オプション

スラッシュ記号の後に続けて(区切り記号をいれずに)アルファベット 1 文字からなるオプション(B,C,D,Hのいずれか)を与えます。これは、すべて省略可能となっています。

[省略可能な引数またはオプション]

大括弧で囲まれた中にある引き数またはオプションはそれが省略可能であることを示します。コマンド投入時には[ ]をつける必要はありません。

項目 1 | 項目 2

項目 1 または項目 2(さらに項目が並列に記述されている場合もあります)のどちらかを用いることを意味します。

## 13.1.1 D(メモリ内容表示)

### 【フォーマット】

- a) D | DB | DW | DL [ /H | /B ]
- b) D | DB | DW | DL <開始アドレス> [ /H | /B ]
- c) D | DB | DW | DL L <表示バイト数> [ /H | /B ]
- d) D | DB | DW | DL [ <開始アドレス> <終了アドレス> ] [ /H | /B ]
- e) D | DB | DW | DL [ <開始アドレス> L <表示バイト数> ] [ /H | /B ]

### 【機能】

メモリーの内容を様々な形式で表示します。

コマンド名の後に付加される文字(B/W/L)によって、表示データ単位が決まります。コマンド名 D のみの場合は、直前に実行されたコマンドで使用された形式で表示されます。デフォルトはバイト型となっています。

a)のように表示範囲の指定を行なわなければ、前回の表示終了アドレスから、以前に指定された<表示バイト数>分表示します。デフォルトの<表示バイト数>は 128 バイトです。

b)のように<開始アドレス>のみ指定した場合は、以前に指定された<表示バイト数>分表示されます。

c)のように<開始アドレス>なしに L <表示バイト数>を与えると、前回の表示終了アドレスから<表示バイト数>分表示します。

表示範囲の指定方法には 2 種類あり、d)のように<開始アドレス>と<終了アドレス>を指定する方式の他、e)のように<開始アドレス>と<表示バイト数>を与える方法があります。/B は 2 進数表示、/H は 16 進数での表示を指定します。リセット後のデフォルトは 16 進表示となります。

表示中にエスケープキーを押すと表示を中止してコマンド入力に戻ります。それ以外のキーが押されると一行表示して休止し、再度何かのキーが押されると表示を再開します。

## 13.1.2 E(メモリ内容変更)

### 【フォーマット】

- a) E | EB | EW | EL [ /H | /B ]
- b) E | EB | EW | EL <開始アドレス> [ /H | /B ]
- c) E | EB | EW | EL <開始アドレス> <データリスト>

### 【機能】

メモリーの内容を変更するコマンドです。

コマンド名の後に付加される文字(E/W/L)によって、データを表示・変更する単位が決まります。コマンド名(E)のみの場合は、直前に実行されたコマンドで使用された型になります。デフォルトはバイト型です。

/B は 2 進数表示、/H は 16 進数での表示を指定します。デフォルトは 16 進表示です。

a)のようにアドレスを指定しない場合は、前回の E(EB/EW/EL)コマンドの最終アドレスが<開始アドレス>となります。

b)のように変更<開始アドレス>を指定すると、指定アドレスのデータを表示してデータ入力待ちとなります。ここで新たなデータを入力するとメモリ内容が更新され、次のアドレスについてデータの入力待ちとなります。

データの入力待ち状態で、複数個のデータをスペースまたはカンマで区切って一度に与えることができます。これらのデータは、現在のアドレスから連続してメモリに書き込まれます。

バイト型データが指定されている場合には、データ入力待ち状態でダブルクォート( ")で囲んだ文字列を指定することができます。

データを変更せずに次のアドレスへ進むためには、改行のみを入力します。アドレスを 1 データ分前に戻すためには、マイナス(-)を入力して改行します。ピリオド(.)を入力して改行すると、コマンドを終了します。

c)のように開始アドレスの後にデータをカンマまたはスペースで区切って与えると、データの表示を行わずに直接メモリに書き込みを行いません。バイト型が指定されている場合はデータリスト中にダブルクォートで囲まれた文字列を与えることもできます。

#### 《注意》

1. データリスト中に不正なデータがあった場合、それ以前のデータはメモリに書き込まれますが、不正なデータ以降のデータは書き込みが中止されます。
2. メモリのない空間や FLASH メモリ上への書き込みはチェックしていません。
3. 指定アドレスがワークエリア内かどうかチェックしていませんので、ワークエリアを破壊しないようご注意ください。

## 13.1.3 I(ポート入力)

### 【フォーマット】

I | IB | IW | IL <ポートアドレス> [/C]

### 【機能】

入力ポートからデータを読み込み、表示します。/C オプションがつけられている場合は、何かキーが押されるまで繰り返しポートからの読み込み・表示を行いません。この機能は、ハードウェアのデバッグ時に便利です。

IW の場合は連続する 2 バイト、IL の場合は 4 バイトを読み出し表示します。H8/3069 では I/O はすべてメモリ空間に割り当てられています。このコマンドで指定するアドレスは 16 ビットで、CN1 に増設される PC/104 拡張モジュールの I/O アドレスです。実際に読み出し・表示されるアドレスは、H8/3069 の対応するメモリアドレスとなっています。

## 13.1.4 O(ポート出力)

### 【フォーマット】

O | OB | OW | OL <ポートアドレス> <出力データリスト> [/C]

### 【機能】

<ポートアドレス>で指定された出力ポートに、<出力データリスト>で与えられたデータを書き込みます。/C オプションがつけられている場合は、何かキーが押されるまで繰り返し同一ポートへの出力を行いません。この機能は、ハードウェアのデバッグ時(特に増設した I/O デコード回路のテスト等)に便利です。

<出力データリスト>中には、複数のデータをスペースまたはカンマで区切って一度に与えることができます。/C オプションが指定されている場合は、これらのデータが繰り返し使用されます。

OW の場合は連続する 2 バイト、OL の場合は 4 バイトに書き込みされます。

H8/3069 では I/O はすべてメモリ空間に割り当てられています。このコマンドで指定するアドレスは 16 ビットで、CN1 に増設される PC/104 拡張ボードの I/O アドレスです。実際に書き込みされるアドレスは、H8/3069 の対応するメモリアドレスに変換されています。

## 13.1.5 G(実行)

### 【フォーマット】

G [<実行開始アドレス>]

### 【機能】

実行制御をユーザープログラムに移します。

<実行開始アドレス>が指定された場合はそのアドレスから、指定されないときは S フォーマットファイルロード時のエンドレコードで指定されたスタートアドレスに制御を移します。制御はサブルーチンコールの形で移されますので、プログラムから RTS によって S ファイルローダのコマンドプロンプトに制御を戻すことができます。S ファイルローダに制御を戻したい場合は、実行するプログラムで ER6,ER7 を保存してください。

## 13.1.6 L(S フォーマットファイルロード)

### 【フォーマット】

L

### 【機能】

このコマンドを投入すると、これ以降入力されるデータをモトローラ S フォーマットであるものと解釈してそのデータをメモリ上に展開します。このコマンドを投入後、ホストマシンのテキストファイル送信機能を使用して、ロードしたいオブジェクトファイルを送信します。

エンドレコードを受信するか、空行(リターンのみ)を受信するとコマンド入力待ち状態に復帰します。エンドレコード中のアドレスフィールドで値が指定されている場合には、その値を G コマンドのスタートアドレスとして保持します。

エンドレコードの指定アドレスが 0 の場合は、RAM 先頭番地(0x400000)をスタートアドレスに設定します。

《注意》

- 1.ファイルの受信中はエコーバックしません。
- 2.ファイルの最後にエンドレコードがない場合、あるいはエラーなどでエンドレコードが正常に検出されなかった場合等ではLコマンドが終了しないため、ファイル送信終了後にキーボードから空行(リターンのみ)を送信し、コマンドプロンプトが表示されてから次のコマンドを投入してください。

## 13.1.7 DISK(USB メモリコマンド)

【フォーマット】

DISK

【機能】

USB メモリディスクの管理コマンドモードに移行します。このコマンドを実行すると、プロンプトが-[DISK]と表示されます。このモードで利用できるサブコマンドを表 13-2に示します。プロンプトで.(ピリオド)を入力すると DISK コマンドを終了します。

表 13-2 DISK サブコマンド

コマンド名	機能
DIR	ディレクトリ内容を表示します。
CD	ディレクトリを変更します。
MD	ディレクトリを作成します。
RD	ディレクトリを削除します。
DEL	ファイルを削除します。
DUMP	ファイル内容を 16 進数でコンソールに表示します。
TYPE	ファイル内容をコンソールに表示します。
L	S フォーマットファイルをロードします。
?	サブコマンド一覧表示

### 13.1.7.1 DIR

【フォーマット】

DIR [<ファイル名>]

【機能】

カレントディレクトリの内容を表示します。ファイル名を指定しない場合は全ファイルが、ファイル名を指定した場合は指定されたファイルがリスト表示されます。ファイル名には?や\*のワイルドカードが使用できます。

### 13.1.7.2 CD

【フォーマット】

CD <ディレクトリ名>

【機能】

カレントディレクトリを変更します。ディレクトリ名を指定しない場合はエラーとなります。指定可能なディレクトリ名はカレントディレクトリの

直接のサブディレクトリに限定されています。上位ディレクトリに移動する場合はディレクトリ名に..(ピリオド2つ)を指定してください。

### 13.1.7.3 MD

【フォーマット】

MD <ディレクトリ名>

【機能】

ディレクトリを作成します。

### 13.1.7.4 RD

【フォーマット】

RD <ディレクトリ名>

【機能】

指定されたディレクトリを削除します。

### 13.1.7.5 DEL

【フォーマット】

DEL <ファイル名>

【機能】

指定されたファイルを削除します。ファイル名にはワイルドカード(?や\*)は使用できません。

### 13.1.7.6 DUMP

【フォーマット】

DUMP <ファイル名>

【機能】

指定されたファイル内容を16進ダンプ表示します。

表示中にエスケープキーを押すと表示を中止してコマンド入力に戻ります。それ以外のキーが押されると一行表示して休止し、再度何かのキーが押されると表示を再開します。

### 13.1.7.7 TYPE

【フォーマット】

TYPE <ファイル名>

【機能】

指定されたファイル内容を表示します。

表示中にエスケープキーを押すと表示を中止してコマンド入力に戻ります。それ以外のキーが押されると一行表示して休止し、再度何かのキーが押されると表示を再開します。

### 13.1.7.8 L

【フォーマット】

L <ファイル名>

【機能】

指定されたファイルをもとローラ S フォーマットとしてメモリ上にロードします。エンドレコード中のアドレスフィールドで値が指定されている場合には、その値を G コマンドのスタートアドレスとして保持します。エンドレコードの指定アドレスが 0 の場合は、RAM 先頭番地(0x400000)をスタートアドレスに設定します。

## 13.1.8 NET(TCP/IP コマンド)

【フォーマット】

NET

TCP/IP コマンドモードに移行します。このコマンドを実行すると、プロンプトが-[NET]と表示されます。このモードで利用できるサブコマンドを表 13-3に示します。プロンプトで.(ピリオド)を入力すると NET コマンドを終了します。

表 13-3 NET サブコマンド

コマンド名	機能
INFO	MAC/IP/MASK/GW 設定値を表示します。
IP	IP アドレスを設定します。
MASK	ネットマスクを設定します。
GW	ゲートウェイアドレスを設定します。
DHCP	IP/MASK/GW を DHCP により設定します。
FTP	FTP サーバを指定して接続します。
FTPD	FTP サーバとして動作します。
?	サブコマンド一覧表示

### 13.1.8.1 INFO

【フォーマット】

INFO

【機能】

MAC アドレス、IP アドレス、ネットマスク、ゲートウェイ IP アドレスを表示します。

### 13.1.8.2 IP

【フォーマット】

IP <IP アドレス>

【機能】

IP アドレスを指定します。アドレスを指定しない場合は現在の設定内容を表示します。

### 13.1.8.3 MASK

【フォーマット】

MASK <ネットマスク値>

【機能】

ネットマスクを設定します。マスクアドレスを指定しない場合は現在の設定内容を表示します。

### 13.1.8.4 GW

【フォーマット】

GW <ゲートウェイ IP アドレス>

【機能】

ゲートウェイ IP アドレスを設定します。アドレスを指定しない場合は現在の設定内容を表示します。

### 13.1.8.5 DHCP

【フォーマット】

DHCP

【機能】

DHCP を使用して IP アドレス、ネットマスク、ゲートウェイ IP アドレスを設定します。

### 13.1.8.6 FTP

【フォーマット】

FTP <接続先 IP アドレス>

【機能】

指定された IP アドレスに FTP 接続します。接続先から GET コマンドで取得したファイルは USB メモリに保存されます。また USB メモリに保存されているファイルを PUT コマンドで接続先の FTP サーバに送信することができます。

### 13.1.8.7 FTPD

【フォーマット】

FTPD

【機能】

このコマンドを実行すると、FTP サーバとして動作します。キー入力でのモードを終了します。

FTP クライアントからの LS, CD, MKDIR, DELETE, PUT, GET, BINARY コマンドに対応しています。

接続時にユーザ名とパスワードを要求しますが、入力内容にかかわらず接続可能です。

## 13.1.9 RTC(時刻表示・設定)

### 【フォーマット】

RTC [<年 4 桁> <月> <日> <時> <分> <秒>]

### 【機能】

引数なしでコマンドを実行すると、RTC から日付と時刻を読み出してコンソールに表示します。パラメータをつけて実行した場合は、与えられた日付・時刻を RTC に設定します。RTC コマンドのパラメータは、全て 10 進数で与えます。年(4 桁)、月、日、時、分、秒をスペースで区切って入力してください。時刻は 24 時間制で設定します。日付のみ、時刻のみの指定はできません。また設定値の正当性はチェックしていませんので、不正な値を設定しないでください。曜日は年月日から自動設定されます。

## 13.1.10 USB(VNC1L テスト)

### 【フォーマット】

USB

### 【機能】

USB コントローラ VNC1L を初期化し、コンソールの入出力をそのまま USB コントローラに接続します。USB コントローラに直接コンソールからコマンドを送り、応答を確認することができます。ESC キーを押すとこのモードを終了し、プロンプトに戻ります。

VNC1L のコマンドモードでデータをバイナリで送信したり、バイナリで受信する場合、CTRL+¥を押すと 16 進データ入出力モードに切り替わります。(CTRL-¥のコード 0x1C は送信されません。)このモードでは、入力された 2 桁の 16 進数をバイナリデータとして VNC1L に送信します。また、受信したデータは全て 16 進数で表示します。16 進データ入出力モードで CTRL+¥を押すと、16 進データ入出力モードを終了します。16 進データモードに切り替わるときコンソールには[HEXStart]、終了すると[HEXEnd]と表示されます。

## 13.1.11 VT(ベクタテーブルモード)

### 【フォーマット】

VTE | VTY

(リセット時は VTY モード)

### 【機能】

割り込みを使用したプログラムを RAM 上で実行する場合に、ベクタテーブルのモードを変更します。リセット時のデフォルトは VTY モードです。YellowIDE6 で作成した RAM 上で実行可能なプログラムは、VTY モードで実行することができます。VTY モードでは、RAM の先頭(0x400000 番地)から 512 バイトがジャンプテーブルとなっています。

VTE モードの場合は、H8/3069 の RAM エミュレーション機能を使用し、G コマンドで指定されたアドレスに制御を移す前にフラッシュメモリのベクタテーブル部分を RAM で置き換えます。

VTE モードを使用するプログラムでは、リンカの指定で割り込みベクタテーブルを 0xFFE000 番地からに配置してください。

### 13.1.12 割り込みプログラムの実行

H8/3069 では割り込みベクタアドレスは固定(000000 ~ 0000FF)で、S ファイルローダ動作中は FLASH メモリ上に存在するため、RAM 上にロードされたプログラムからこのベクタテーブルの内容を変更することはできません。

このため、RAM にロードされるアプリケーションから割り込みが利用できるよう、S ファイルローダでは 2 種類の方法で対応しています。

VTE モードでは、H8/3069 の RAM エミュレーション機能を使用し、G コマンドで RAM にロードしたプログラムに制御を移す直前に 0xFFE000 ~ 0xFFEFFF(H8/3069 内蔵 SRAM 上)を 0 番地からに割り当てます。作成するプログラムでは通常 0x000000 番地から配置するベクタテーブルをリンカへの指示を変更し 0xFFE000 番地からに配置してください。

VTY モードでは、RAM の先頭(0x400000)からベクタ番号に対応したジャンプテーブル(ベクタではないことに注意)が配置されています。作成するプログラムでは、通常の割り込みベクタテーブルのかわりに、割り込み処理ルーチンへの JMP 命令を RAM の先頭からベクタ番号×4 のアドレスに配置してください。YellowIDE6 は RAM 上で実行するプログラムにこのジャンプテーブルを自動的に作成します。

## 13.2 HT1040WR

H8/3069 内蔵フラッシュメモリ書き換えのユーティリティです。

### 【説明】

メニュー構成は次の通りです。

#### FILE – SEND

ファイルを選択して、SEND ボタンを押すと、書き込みを開始します。

#### FILE – MAC

MAC アドレスを指定してユーザブートマットに書き込みします。

この機能を使用する場合は、TP3 を L にしてブートモードで電源を投入してください。MAC 書き込みを行う場合、ユーザブートマットの内容およびユーザマットの内容は失われます。再設定する MAC アドレスは必ず CN5 上に表示されている値を使用してください。異なる値を設定した場合は MAC アドレスが他の弊社製品と重複する可能性があります。

#### FILE – QUIT

プログラムを終了します。

#### SETTINGS – PORT

HT1040 との通信を行うポートを選択します。

#### ABOUT

バージョンを表示します。

### 【エラー】

- COMx Open Error!  
指定された COM ポートが使用できません。存在しないポート番号を指定しているか、他のソフトウェアと競合している可能性があります。通信ソフトウェアを立ち上げている場合は、いったん終了してください。
- Handshake Error
- Handshake Timeout Error  
HT1040 との通信に失敗しました。FLASH メモリ書き込みモードにするためのジャンパ設定が正しく設定されているか、HT1040 に正しく電源が接続されているかどうかを確認してください。
- Flash Programming Error  
何らかの原因で FLASH メモリ書き込みに失敗しました。HT1040 の電源を再投入してやり直してみてください。なお、H8/3069R F-ZTAT ハードウェアマニュアルによると、FLASH メモリ書き換え回数は 100 回となっていますので、これを大幅に越える場合は書き込みができない可能性があります。
- 使用する COM ポートがリストに存在しない  
他のアプリケーションが使用中の COM ポートはリストに含まれません。HT1040WR をいったん終了し、COM ポートを使用している他のアプリケーションを終了して COM ポートを開放してから HT1040WR を再起動してください。

# 14 外形寸法图

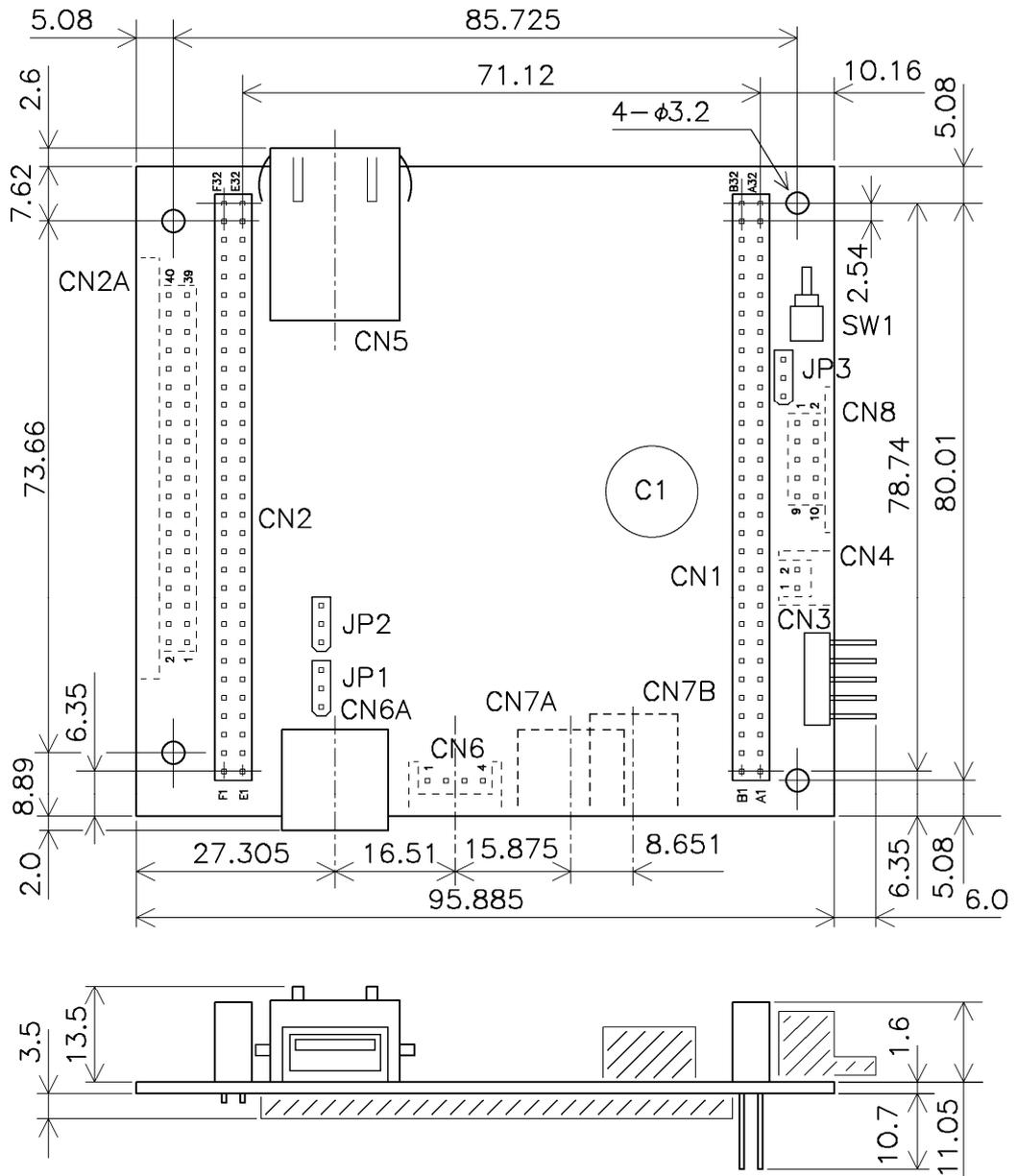


图 14-1 外形寸法图



HT1040 ユーザーズマニュアル 2011年4月10日 rev.1.2

---

梅澤無線電機株式会社

東京営業部

101-0044 東京都千代田区鍛冶町 2-3-14

TEL03-3256-4491 FAX03-3256-4494

仙台営業所

982-0012 仙台市太白区長町南 4 丁目 25-5

TEL022-304-3880 FAX022-304-3882

札幌営業所

060-0062 札幌市中央区南 2 条西 7 丁目

TEL011-251-2992 FAX011-281-2515

本製品・資料についての技術的なお問い合わせは



(TEL/FAX)0120-024768

<http://www.umezawa.co.jp/>

---