

RTE-V832-PC

ユーザーズ・マニュアル (Rev. 1.04)

Midas lab

改訂履歴

実施日	Revision	章	内容
1998年7月10日	1.00		初版
1998年8月20日	1.01	10.1	基板端との位置関係が上下逆であった誤記訂正
1998年9月10日	1.02	5.16 11.2 13.3.8	コネクタ名称の誤記訂正 サイズ値の誤記訂正 追記
1999年2月2日	1.03	11.1	RFC の初期値訂正
2000年12月26日	1.04	8.9	PIC: プリンタからの割り込みをハイレベルから立ち 上がりエッジの入力に変更(H/W Rev.2.1 ~ が対象)

目次

1. はじめに	1
1.1. マニュアル表記について	1
2. 機能	2
3. 主な特徴	2
4. 基本仕様	3
5. ボードの構成	3
5.1. リセット・スイッチ (RESET)	3
5.2. 電源ジャック (JPOWER)	3
5.3. スイッチ 1 (SW1)	4
5.4. スイッチ 2 (SW2)	5
5.5. スイッチ 3 (SW3)	5
5.6. スイッチ 4 (SW4)	6
5.7. LED	6
5.8. ROM エミュレータ用テストピン (JROM_EM)	6
5.9. クロック・ソケット (OSC1)	7
5.10. ROM ソケット	7
5.11. DMARQ0-,DMARQ1-切り離しジャンパー(JP1)	7
5.12. タイマクロックの周波数切り替えジャンパー(JP2)	7
5.13. AUDIO 入力レベルの切り替えジャンパー(JP3,JP4,JP5,JP6)	7
5.14. シリアル・コネクタ (JSIO1,JSIO2)	8
5.15. パラレル・コネクタ (JPRT)	9
5.16. AUDIO 用ミニジャック(JIN-R,JIN-L,JOUT)	10
5.17. デバッグ用コネクタ(JDCU)	10
5.18. CPU コネクタ(JCPU-1,JCPU-2)	11
5.19. 拡張バス・コネクタ (JEXT)	12
6. ホスト PC との接続	13
6.1. ボード単体で使用する場合 (RS-232C 接続)	13
6.2. PCI スロットに組み込む場合 (PCI バス接続)	13
7. ハードウェア・リファレンス	14
7.1. メモリ・I/O のマップ	14
8. IO マップ	16
8.1. IO 一覧	16
8.2. DIPSW2 読み出しポート(4500-0000H [READ ONLY])	17
8.3. DIPSW1 読み出しポート(4500-1000H [READ ONLY])	17
8.4. 7 セグメント LED 表示データ出力ポート(4500-2000H [WRITE ONLY])	17

8.5.	コマンドレジスタ(4500-5000H [READ/WRITE]).....	18
8.6.	EXT-IO 上位アドレス設定レジスタ(4500-6000H [READ/WRITE]).....	18
8.7.	UART/PRINTER (TL16PIR552) (4500-8000H ~ 4500-A03EH).....	18
8.8.	TIC (uPD71054) (4500-B000H ~ 4500-B00CH)	20
8.9.	割り込みコントローラ (PIC) (4500-D000H ~ 4500-D018H)	21
8.10.	AUDIO コントローラ (AUDCNT) (4580-0000H ~ 4580-0010H,4580-2000H).....	22
8.11.	uPD63310 レジスタ(4580-1000H ~ 4580-100FH).....	24
9.	割込みと DMA	25
9.1.	割込み	25
9.2.	NMI の使用方法.....	25
9.3.	DMA 要求	26
10.	EXT-BUS バス仕様	27
10.1.	ピン配置	27
10.2.	信号	28
10.3.	データバスの接続	30
10.3.1.	16 ビット・データバス CPU (参考)	30
10.3.2.	32 ビット・データバス CPU (V832 の場合)	31
10.4.	タイミング	32
10.5.	適合コネクタ	33
10.6.	注意事項	33
11.	ソフトウェア	34
11.1.	初期化	34
11.2.	ライブラリ	35
11.3.	タイマの使用法	36
11.4.	音声入出力	37
12.	マスクブル割り込みを使用したアプリケーションの開発	39
12.1.	割り込みベクタ	39
12.2.	内蔵命令 RAM	40
12.3.	一般的な制限事項 / 注意事項	40
12.4.	割り込み処理内におけるブレーク・ポイント使用に関する制限事項	41
13.	APPENDIX.A MULTI モニタ	42
13.1.	ボードの設置	42
13.1.1.	RTE for Win32 のインストール	42
13.1.2.	SW1 の設定	42
13.1.3.	ボードの接続	42
13.2.	MULTI モニタ	43
13.2.1.	モニタ・ワーク RAM	43
13.2.2.	割り込み	43
13.2.3.	強制ブレーク用の割り込み	43

13.2.4.	<i>_INIT_SP の設定</i>	43
13.2.5.	リモート接続	43
13.2.6.	モニタの実行領域	43
13.2.7.	特殊命令	43
13.3.	RTE コマンド	44
13.3.1.	<i>HELP(?)</i>	44
13.3.2.	<i>INIT</i>	44
13.3.3.	<i>VER</i>	44
13.3.4.	<i>INB,INH,INW</i>	44
13.3.5.	<i>OUTB,OUTH,OUTW</i>	45
13.3.6.	<i>DCTR</i> コマンド	45
13.3.7.	<i>ICTR</i> コマンド	45
13.3.8.	<i>PLLCR</i> コマンド	45
13.3.9.	<i>CMCR</i> コマンド	45
13.3.10.	<i>SFR</i> コマンド	45
14.	APPENDIX.B PARTNER モニタ	46
14.1.	ボードの設置	46
14.1.1.	<i>SW1</i> の設定	46
14.2.	PARTNER モニタ	47
14.2.1.	モニタ・ワーク RAM	47
14.2.2.	割り込み	47
14.2.3.	強制ブレーク用の割り込み	47
14.2.4.	<i>SP</i> の設定	47
14.2.5.	リモート接続	47
14.2.6.	モニタの実行領域	47
14.2.7.	特殊命令	47

1. はじめに

「RTE-V832-PC」は、NEC 製の RISC プロセッサ V832 の評価を目的とした、PCI バスのインターフェースを持った評価ボードです。DOS/V 互換機の PCI スロットに実装してご使用頂けます。

ボードは、最高 143MHz で動作する V832 とメモリ、シリアル・パラレルインターフェースやオーディオ入出力などの I/O で構成されます。メモリは、高速 SRAM と大容量の SDRAM を標準で実装しています。SDRAM の制御は、V832 が内蔵するメモリコントローラを使用して行います。

これらの機能を使用して、プロセッサの性能評価、デモ、シミュレータの実行エンジン、アプリケーション・プログラムの初期段階の開発など、幅広くご利用ください。

本製品は、開発用のソフトウェアツールとして、GHS 社の Multi と自社製の PARTNER のどちらかをソースレベルデバッガとしてご使用になります。ご使用になるデバッガによって、ROM に搭載するモニタは異なります。

ROM は、購入時にご指定頂いたモニタが搭載されています。デバッガを同時に購入されていない場合は、それぞれ別売りされていますので、別途お買い求めください。

1.1. マニュアル表記について

本書では、数字の表記については表の表記を用います。16 進数や 2 進数の表記では、桁数が多くて読みにくい場合は、4 桁ごとに“-”（ハイフン）を入れてあります。

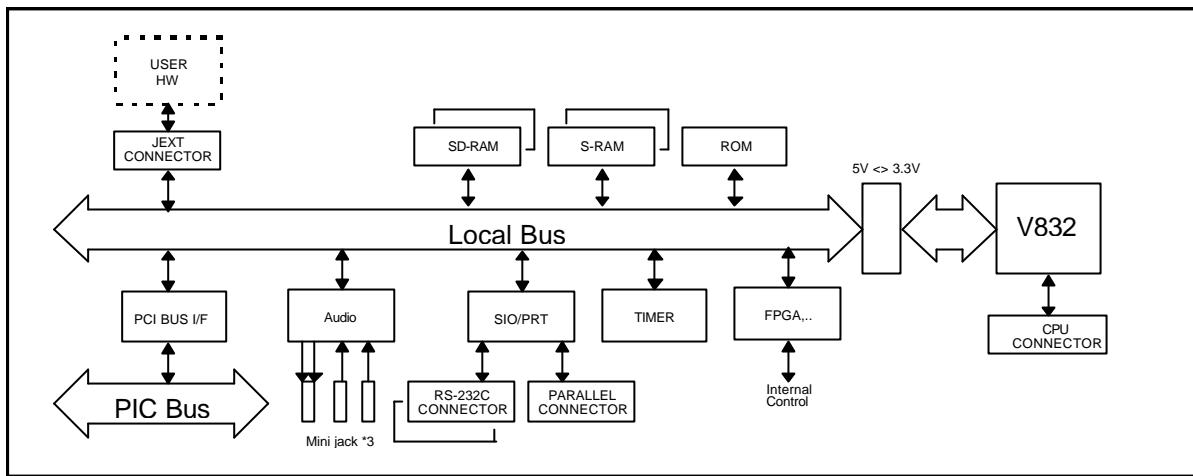
進数	表記規則	例
10 進数	数字のみを示します	“10”は 10 進数の“10”を示します
16 進数	数字の末尾に”H”を記します	“10H”は 10 進数の“16”を示します
2 進数	数字の末尾に”B”を記します	“10B”は 10 進数の“2”を示します

数字表記規則

MULTI は米国 Green Hills Software, Inc の商標です。

2. 機能

RTE-V832-PC の機能ブロックの概要を図に示します。



RTE-V832-PC ブロック図

3. 主な特徴

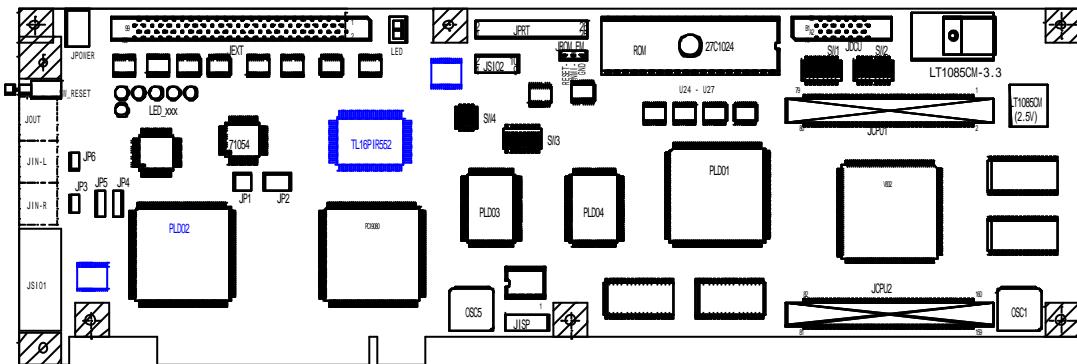
- GreenHills 社の MULTI と自社製 PARTNER 用のモニタ ROM を 2 種類用意しています。
- MULTI また PARTNER を使用した高級言語レベルでのリアルタイム実行・評価が可能です。
- ROM エミュレータが接続できます。
- 高速 SRAM を 512KB、SDRAM は 32MB を標準搭載しています。
- SRAM,DRAM は、16bit バスモードでの評価ができます。
- シリアル(2ch)&プリンタ(1ch)のインターフェースを用意しています。
- タイマー2ch を搭載しています。(モニタで、1ch 使用)
- 音声の入出力チャンネルをそれぞれ 2ch 搭載しています。

4. 基本仕様

プロセッサ	V832	
CPU クロック	142.8MHz	
バスクロック	47.6MHz	
消費電力	+5V (2A)	
メモリ		
EPROM	128KB	64K × 16bit (40pin-DIP) × 1(max.512KB)
SRAM	512KB	128K × 8bit × 4
DRAM	32MB	64M-SDRAM × 4
I/O		
シリアル(2ch)	NS16550 相当	10 ピンヘッダ , DB9 コネクタ
プリンタ	IEEE1284 互換	26 ピンヘッダ
音声入出力(2ch)	uPD63310	ミニジャック(MIC * 2,LINEOUT* 1)
タイマ	i8254 相当	分解能 500nS
IO ポート	LED(7seg)表示 / スイッチ入力	
その他		
CPU コネクタ	V832 の全機能ピンを接続したコネクタ	
32bit 標準外部拡張バス	RTE-PC 標準 32bit I/F(16MB,32bit バス,DMA 対応)	
リセット・スイッチ	Push 式	

5. ボードの構成

下図は RTE-V832-PC ボード上の主要な部品の物理的な配置です。ここでは、それぞれの部品について説明します。



RTE-V832-PC の部品配置図

5.1. リセット・スイッチ (RESET)

RESET は本ボード全体のリセット・スイッチです。このスイッチを押すと CPU を含む全ての回路がリセットされます。

5.2. 電源ジャック (JPOWER)

本ボードを PCI バス・スロットに挿さずに単体で使用する場合に、JPOWER コネクタに外部電源を接続して電源を供給します。

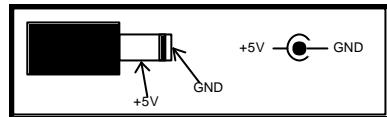
JPOWER ジャックに供給する電源は、下記の通りです。

電圧：5V

電流：最大 2.0A (ただし、JEXT コネクタへの供給分を含まず)

適合コネクタ：Type A (5.5mm)

極性：



【注意】外部から電源を給電する場合は、電源プラグの**極性に注意**してください。また、PCI バス・スロットに挿して使用する場合には、JPOWER に電源を接続しないでください。故障の原因になります。

5.3. スイッチ 1 (SW1)

SW1 は、汎用の入力ポートのスイッチです。モニタを使用する場合には、一部を除き、割り当て済みです。ポートからの読み出し時、スイッチは、OFF で 1、ON で 0 の値になります。モニタでの割り付けは、以下の各章を参照して、使用環境に合わせて設定してご使用ください。

Multi を使用する場合、13.1.2 SW1 の設定を参照ください。

PARTNER を使用する場合、14.1.1 SW1 の設定を参照ください。

5.4. スイッチ 2 (SW2)

SW2 は、本ボードの H/W の状態を設定するスイッチです。全ビット、ソフトウェアで読み出し可能です。ポートからの読み出し時、スイッチは、OFF で 1、ON で 0 の値になります。詳細は、「8.2 DIPSW2 読み出しポート(4500-0000H [Read Only])」を参照してください。

番号	機能名称	内容
1	BSIZE16	SRAM,SDRAM のバスサイズの指定 OFF:32bit (出荷時の設定) ON:16bit
2	BCLK_HI	バスクロックの周波数の指定 OFF:33MHz を越える周波数 (出荷時の設定) ON:33MHz 以下の周波数
3	CMODE	CPU の CMODE 端子に直結 OFF:8 週倍 ON:6 週倍 (出荷時の設定)
4	TEST	常に OFF でご使用ください。
5	ROM_TYPE0	使用する ROM の種類を設定
6	ROM_TYPE1	[ROM_TYPE1,ROM_TYPE0] [OFF,OFF] モニタ ROM 使用時 (出荷時の設定) [OFF,ON] 27C4096 使用時 [ON,OFF] 27C2048 使用時 [ON,ON] 27C1024 使用時
7	BNK_DIS	ROM の上下半分を分けて使用(Bank)するかどうかの設定 OFF:上下半分に分けて使用 (出荷時の設定) ON:連続した領域として使用
8	BNK_LOW	BNK_DIS=OFF の時の上下の選択 OFF:上位側の半分を選択 ON:下位側の半分を選択 (出荷時の設定)

5.5. スイッチ 3 (SW3)

SW3 は、本ボード内で使用している割込みを物理的にカットする為のスイッチです。出荷時全ての設定は、ON (接続された状態) になっています。外部で使用する場合にのみ、OFF に設定してください。但し、内部で使用している割込みが不要な場合に限ります。

番号	INT 名称	内部で使用している割り込みソース
1	INTP03	PIC-INT0 (モニタを使用する場合、OFF にしないでください)
2	INTP02	PIC-INT1
3	INTP01	EXTbus-INT0
4	INTP00	Audio
5	INTP13	EXTbus-INT3
6	INTP12	EXTbus-INT2
7	INTP11	EXTbus-INT1
8	INTP10	Printer

5.6. スイッチ 4 (SW4)

SW4 は、本ボード内で使用している DMA を物理的にカットする為のスイッチです。出荷時全ての設定は、ON (接続された状態) になっています。外部で使用する場合にのみ、OFF に設定してください。但し、内部で使用している DMA が不要な場合に限ります。

番号	DMA 名称	内部で使用している割り込みソース
1	DMARQ2-	EXTbus-DREQ0-
2	DMAAK2-	EXTbus-DACK0-
3	DMARQ3-	EXTbus-DREQ1-
4	DMAAK3-	EXTbus-DACK1-

【注意】SW 番号 1 と 2、3 と 4 は、それぞれ同じ位置に設定してください。

5.7. LED

LED は、各種ステータスを示しています。表に内容を示します。

名称	内容
POWER	ボードに電源が供給されている時に点灯
PLY	音声を出力中緑色に点灯 音声を出力中にエラーが発生した時に赤色に点灯
REC	音声を録音中緑色に点灯 音声を録音中にエラーが発生した時に赤色に点灯
PCI_PERR	PCI バスでパリティエラーが発生した時に点灯
PCI9_DEAD	PCI コントローラがデッドロックした時に点灯
TOVER	タイムオーバー発生時に点灯

LED ステータス

【注意】PCI_PERR と PCI9_DEAD が点灯した場合は、システムを立ち上げ直してください。

5.8. ROM エミュレータ用テストピン (JROM_EM)

JROM_EM は、ROM エミュレータを接続する際に使用するテストピンです。下記の制御信号が入力できます。表に信号名と機能を示します。

信号名	入出力	機能
RESET-(1)	入力	Low レベル入力により、CPU がリセットされます。 ROM エミュレータからのリセット要求信号を接続します。 1K でプルアップされています。
NMI-(2)	入力	Low レベル入力により、CPU に NMI が入ります。ただし、ソフトウェアによってマスクできますので、解除する必要があります。 (「8.9 割り込みコントローラ (PIC) (4500-D000H ~ 4500-D018H)」参照ください。) ROM エミュレータからの NMI 要求信号を接続します。 1K でプルアップされています。
GND(3)	- - -	GND。ROM エミュレータの GND と接続します。

JROM_EM 端子の機能

5.9. クロック・ソケット (OSC1)

OSC1 ソケットには、CPU に供給するクロックのためのオシレータを実装します。V832 では、システム・クロックの生成に PLL を使用しています。OSC1 ソケットへは、V832 内部の動作周波数の 6 分の 1 または、8 分の 1 の周波数のオシレータを実装してください。

オシレータは、DIP8 ピンタイプ (ハーフタイプ) のものを実装してください。

【注意】 オシレータの足を切って実装する場合、足が短かすぎるとフレーム (外装) 部分が、ソケットの端子とショートしてしまいますのでご注意願います。

5.10. ROM ソケット

ROM ソケットには、標準で 128K バイト ($64K \times 16$ ビット) の 40 ピン ROM が実装されています。変更する場合は、27C1024,27C2048,27C4096 タイプで、アクセス・タイムが 120ns 以下のものをご利用ください。使用する ROM の種類や使い方によって、SW2 の設定が必要です。詳細は、「5.4 スイッチ 2 (SW2)」を参照してください。

5.11. DMARQ0-,DMARQ1-切り離しジャンパー(JP1)

JP1 は、Audio のデータ転送に使用している、DMARQ0-と DMARQ1-を物理的に V832 から切り離す為のジャンパーです。通常は、1-23-4 をショートしてご使用ください。

【備考】 オープンにした場合、Audio のデータを DMA で転送することはできません。

5.12. タイマクロックの周波数切り替えジャンパー(JP2)

JP2 は、アプリケーションで使用可能なタイマ(CH#1,CH#2)へ供給するクロックの切り替えに使用します。

1-2	:2MHz	(出荷時の設定)
3-4	:4MHz	
5-6	:8MHz	

5.13. Audio 入力レベルの切り替えジャンパー(JP3,JP4,JP5,JP6)

Audio の入力レベルの切り替えに使用します。MIC か LINE かによって、以下の通り設定してください。出荷時の設定は、MIC 入力です。

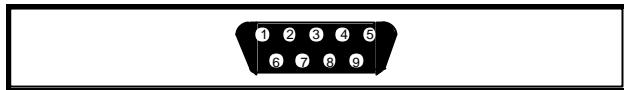
入力レベル	JP3	JP4	JP5	JP6
MIC	ショート	1-2 ショート	1-2 ショート	ショート
LINE	オープン	2-3 ショート	2-3 ショート	オープン

5.14. シリアル・コネクタ (JSIO1,JSIO2)

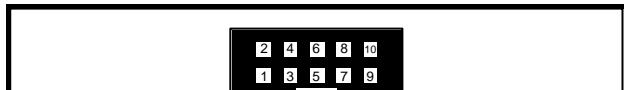
JSIO1,JSIO2 コネクタは、シリアル・コントローラ (TL16PIR552PH) によって制御される RS-232C インターフェース用のコネクタです。コネクタの形状は、JSIO1 は PC/AT で用いられる一般的な D-SUB9 ピンの RS-232C コネクタ、JSIO2 は 2.54mm ピッチのピンプラグ型のコネクタです。何れも、全ての信号は RS-232C レベルに変換されています。コネクタのピン番号と内容は図と表の通りです。

表には、ホストと接続する場合の接続信号について、ホスト側が D-SUB9 ピンの場合と D-SUB25 ピンの場合の布線をそれぞれ示してあります (一般的なクロスケーブルの布線です)。

また JSIO2 のピン配置は、リボンケーブルに対して圧接型コネクタを使用した場合、JSIO1 のピン配置と同じになるようになっています。



JSIO1 ピン配置 (オス)



JSIO2 ピン配置

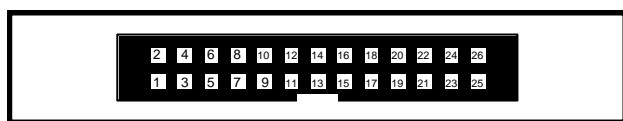
JSIO1 ピン番号	JSIO2 ピン番号	信号名	入出力	ホストの接続ピン番号	
				D-SUB9	D-SUB25
1	1	DCD	入力		
2	3	RxD(RD)	入力	3	2
3	5	TxD(SD)	出力	2	3
4	7	DTR(DR)	出力	1, 6	6, 8
5	9	GND		5	7
6	2	DSR(ER)	入力	4	20
7	4	RTS(RS)	出力	8	5
8	6	CTS(CS)	入力	7	4
9	8	RI	入力		
--	10	NC			

JSIO1,2 コネクタ信号

5.15.パラレル・コネクタ (JPRT)

JPRT コネクタは、パラレル（プリンタ）・コントローラ（TL16PIR552PH）によって制御されるパラレル通信用のコネクタです。コネクタの形状は、2.54mm ピッチのピンプラグ型のコネクタです。何れも、全ての信号は 5V レベルです。コネクタのピン番号と内容は図と表の通りです。

また JPRT のピン配置は、リボンケーブルに対して圧接型コネクタを使用した場合、PC/AT で用いられている一般的な D-SUB25 ピンのピン配置と同じになっています。



JPRT ピン配置

JPRT ピン番号	信号名	JPRT ピン番号	信号名
1	STB-	2	AUTO_FD-
3	D0	4	ERROR-
5	D1	6	INIT-
7	D2	8	SELECT_IN-
9	D3	10	GND
11	D4	12	GND
13	D5	14	GND
15	D6	16	GND
17	D7	18	GND
19	ACK-	20	GND
21	BUSY	22	GND
23	PE	24	GND
25	SELECT	26	NC

JPRT コネクタ信号

5.16. Audio 用ミニジャック(JIN-R,JIN-L,JOUT)

Audio 用のジャックとして、モノラルのマイク / ライン入力が 2ch、ステレオの出力が 1ch 用意されています。それぞれの入出力条件は以下の通りです。

JIN-R,JIN-L

電気的な入力条件

MIC 入力指定時： 140mVp-p (内部アンプ増幅 約 20db)

LINE 入力指定時： 1.4Vp-p

適合する物理的なプラグの形状

モノラル・ミニプラグ (3.5 DIA) × 2ch

JLINEOUT

電気的な出力条件

1.4Vp-p

適合する物理的なプラグの形状

ステレオ・ミニプラグ (3.5 DIA) × 1ch

【補足】MIC と LINE の切り替えは、JP3,4,5,6 の設定で行います。

5.17. デバッグ用コネクタ(JDCU)

V832 に内蔵しているデバッグ機能を利用した、デバッグ・ツールを接続するためのコネクタです。

基板側のコネクタ : KEL 社 8830E-026-170S

ピン番号	信号名	ピン番号	信号名
A1	TRCCLK	B1	GND
A2	TRCDATA0	B2	GND
A3	TRCDATA1	B3	GND
A4	TRCDATA2	B4	GND
A5	TRCDATA3	B5	GND
A6	NC.	B6	GND
A7	DDI	B7	GND
A8	DCK	B8	GND
A9	DMS	B9	GND
A10	DDO	B10	GND
A11	DRST-	B11	NC.
A12	NC.	B12	NC.
A13	NC.	B13	+3.3V

JDCU コネクタ信号

5.18. CPU コネクタ(JCPU-1,JCPU-2)

CPU コネクタの信号は、V832 と直結した信号です。多くの信号は、ボード内部で使用していますので、JCPU から信号を引き出す場合は、注意が必要です。尚、信号のレベルは、3.3V です。

ピン番号	信号名	ピン番号	信号名
1	GND	2	CS1-
3	CS0-	4	WE-
5	RAS-	6	UDQM
7	ULDQM	8	LUDQM
9	LLDQM	10	+3.3V
11	GND	12	SDCLKOUT
13	CKE	14	CAS-
15	A1	16	A2
17	A3	18	A4
19	+2.5V	20	GND
21	+3.3V	22	GND
23	A5	24	A6
25	A7	26	A8
27	A9	28	A10
29	A11	30	+3.3V
31	GND	32	A12
33	A13	34	A14
35	A15	36	A16
37	A17	38	A18
39	A19	40	+5V
41	GND	42	A20
43	A21	44	A22
45	A23	46	CLKOUT
47	BCLK2(bufferd CLKOUT)	48	MWR0-
49	MWR1-	50	MWR2-
51	MWR3-	52	NC.
53	NC.	54	NC.
55	NC.	56	NC.
57	NC.	58	GND
59	+2.5V	60	GND
61	NC.	62	BT16B(GND)
63	RESET-	64	NMI-
65	NC.	66	CMODE
67	P3	68	P4
69	P2	70	P1
71	P0	72	+5V
73	GND	74	INTP0
75	INTP12	76	INTP11
77	INTP13	78	PB0
79	PB1	80	+2.5V

JCPU-1 コネクタ信号

ピン番号	信号名	ピン番号	信号名
81	GND	82	INTP00
83	INTP01	84	INTP02
85	INTP03	86	DMARQ3-
87	DMARQ2-	88	DMARQ1-
89	DMARQ0-	90	DMAAK3-
91	DMAAK2-	92	DMAAK1-
93	DMAAK0-	94	TC-/STOPAK-
95	CS7-	96	CS6-
97	CS5-	98	CS4-
99	CS3-	100	CS2-
101	+3.3V	102	GND
103	HLDLK-	104	HLDLK-
105	R/W-	106	READY-
107	BCYST-	108	IORD-
109	IOWR-	110	UUBEN-
111	ULBEN-	112	LUBEN-
113	LLBEN-	114	MWR-
115	MRD-	116	+3.3V
117	GND	118	D0
119	D1	120	+2.5V
121	GND	122	D2
123	D3	124	D4
125	D5	126	D6
127	D7	128	+5V
129	GND	130	D8
131	D9	132	D10
133	D11	134	D12
135	D13	136	D14
137	D15	138	+5V
139	GND	140	D16
141	D17	142	D18
143	D19	144	D20
145	D21	146	D22
147	D23	148	+5V
149	GND	150	+2.5V
151	GND	152	D24
153	D25	154	D26
155	D27	156	D28
157	D29	158	D30
159	D31	160	+5V

JCPU-2 コネクタ信号

使用しているコネクタは、ヒロセ電機製の FX2-80P-1.27SV です。

5.19. 拡張バス・コネクタ (JEXT)

JEXT コネクタは、メモリや I/O などを拡張できるように用意されたコネクタです。このコネクタには、本ボード内部のローカル・バスが接続されています。詳細は、「10 EXT-BUS バス仕様」を参照ください。

6. ホスト PC との接続

6.1. ボード単体で使用する場合 (RS-232C 接続)

ホストマシンとシリアルで接続する際の手順は以下の通りです。

ホストと接続するための RS-232C ケーブルと、電源供給のための外部電源 (+5V ??A) を用意してください。電源については、電圧と容量、**コネクタの極性に注意してください**。RS-232C ケーブルは、一般的にクロスケーブルと呼ばれているタイプのものです。念のため、結線を確認した上でご使用ください。結線は「5.14 シリアル・コネクタ (JSIO1,JSIO2)」、電源については、「5.2 電源ジャック (JPOWER)」を参照してください。

ボード上の DIPSW の設定と確認を行ってください。SW1 のボーレイトの設定は、必ず実施ください。

(13.1.2 SW1 の設定、14.1.1 SW1 の設定の各項を参照ください。)

JSIO1 コネクタとホストマシンを RS-232C ケーブルで接続し、JPOWER コネクタへ電源を供給してください。ボードの POWER-LED が点灯することを確認します。LED が点灯しない場合は、**すぐに電源を切り接続を確認してください**。

ホストマシンでデバッガを起動して、RS-232C 経由でコネクトします。エラーが発生する場合には、シリアル・ケーブルやスイッチ（特にボーレート）の設定等に間違いがないかを確認してください。

【注意】

1. PCI バスに実装しない状態で通電する場合は、必ず、添付の PCI バスター・ミネータ基板を接続してください。
2. ボードは絶縁物の上に設置してください。通電中に、導電物を基板に接触させると故障の原因になります。

6.2. PCI スロットに組み込む場合 (PCI バス接続)

ホスト PC の PCI スロットに組み込んで使用する場合の手順は以下の通りです。

ボード上の DIPSW の設定と確認を行ってください。

PC の筐体を開き、PCI スロットにボードを取り付けます。しっかりと差し込まれたかを確認して、バックパネルをねじ止めしてください。

PC の電源を入れてボードの POWER-LED が点灯することを確認します。LED が点灯しない場合は、**すぐに電源を切り接続を確認してください**。また、ホストマシンが正常に起動するかも確認してください。

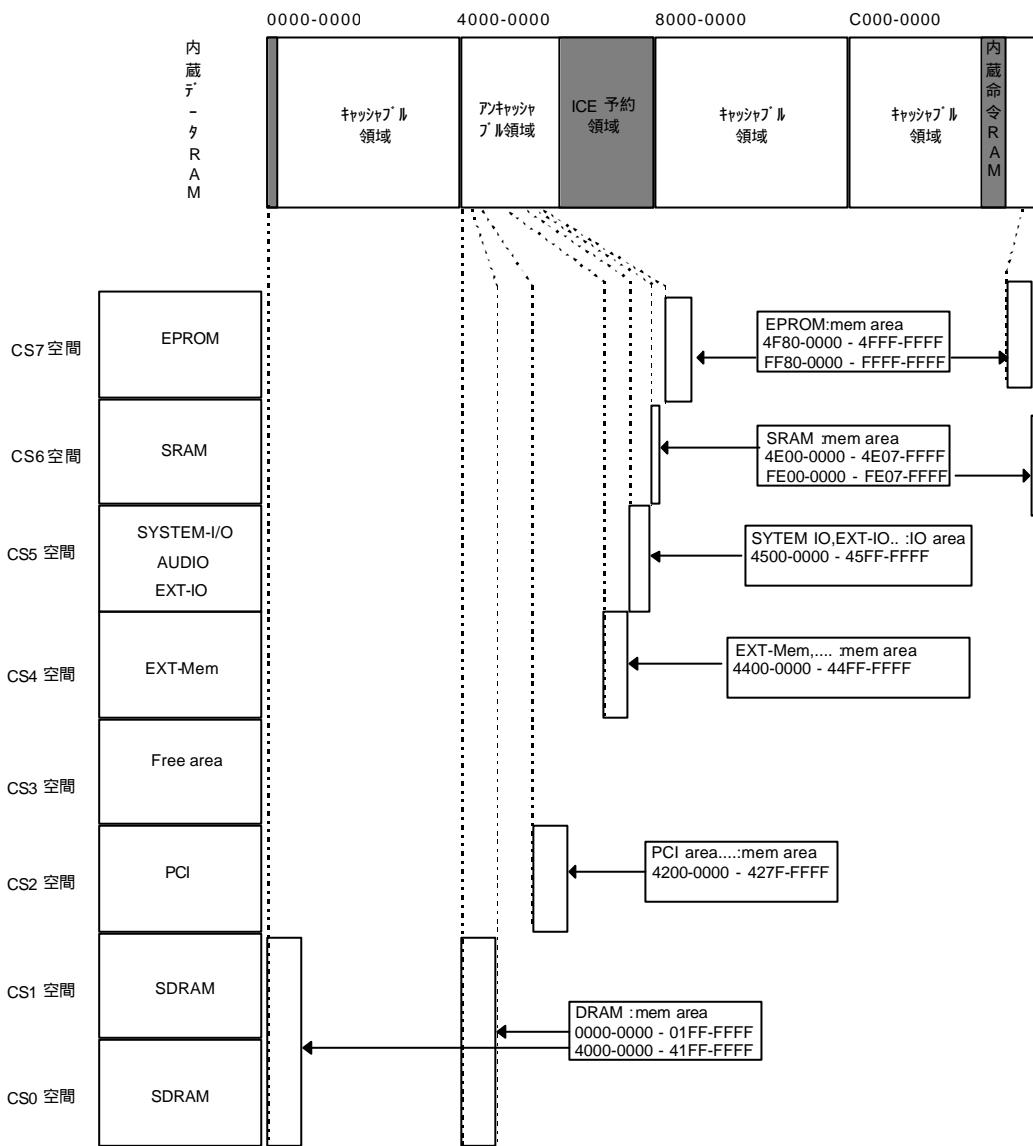
ホストマシンでデバッガを起動して、PCI バス経由でコネクトします。エラーが発生する場合には、ボードの実装やソフトウェアのインストールに間違いがないかを確認してください。

7. ハードウェア・リファレンス

ここでは、RTE-V832-PC ボードのハードウェアについて記述します。

7.1. メモリ・I/O のマップ

ボードのメモリと I/O の割り付けは、以下の通りです。



メモリ・I/O マップ

CS0 空間 (x000-000 ~ x0FF-FFFF, x800-000 ~ x8FF-FFFF)

16M バイトの SDRAM がマップされています。SW2-1 の指定により、16-bit と 32-bit、両方のデータバス幅の指定が可能です。但し、16-bit 時は、8M バイトまでに容量制限されます。

SDRAM とのインターフェースは、V832 内蔵の DRAM コントローラを使用しています。

CS1 空間 (x100-000 ~ x1FF-FFFF, x900-000 ~ x9FF-FFFF)

16M バイトの SDRAM がマップされています。SW2-1 の指定により、16-bit と 32-bit、両方のデータバス幅の指定が可能です。但し、16-bit 時は、8M バイトまでに容量制限されます。

タバス幅の指定が可能です。但し、16-bit 時は、8M バイトまでに容量制限されます。

SDRAM とのインターフェースは、V832 内蔵の DRAM コントローラを使用しています。

CS2 空間 (x200-000 ~ x27F-FFFF,xA00-000 ~ xA7F-FFFF)

PCI とブリッジする為のコントローラがマップされています。

CS3 空間 (x300-000 ~ x3FF-FFFF,xB00-000 ~ xBFF-FFFF)

未使用の領域です。

CS4 空間 (x400-000 ~ x40F-FFFF,xC00-000 ~ xC0F-FFFF)

16M バイトの EXT バス上のメモリ領域がマップされています。

CS5 空間 (x500-000 ~ x5FF-FFFF,xD00-000 ~ xDFF-FFFF)

ボードに搭載しているタイマ、Audio、シリアル、パラレルなどの I/O デバイスと EXT バス IO 領域がマップされています。フルデコードを行っていないため、各所にイメージ空間が現れます。記載された I/O アドレス以外へのアクセスは行わないでください。

ウェイト制御は外部の FPGA で行なっています。各 I/O デバイスの詳細につきましては、「8 IO マップ」を参照してください。

CS6 空間 (x600-000 ~ x6FF-FFFF,xE00-000 ~ xEFF-FFFF)

512K バイトの高速 SRAM (日本電気製の uPD431008LE-15 : 128K * 8bit, 15nS) をマップしています。SW2-1 の指定によって、16-bit と 32-bit、両方のデータバス幅の指定が可能です。16-bit 時も 512k バイトの容量を確保しています。512K バイトおきに SRAM のイメージが現れます。

ウェイト制御は CPU 内部のバスコントローラで行ないます。33MHz 以下の外部クロックであれば、0 ウェイトでアクセスできます。

CS7 空間 (x700-000 ~ x7FF-FFFF,xF00-000 ~ xFFF-FFFF)

ブート ROM 用の空間で、16 ビットバスでブートします。(BT16B 端子は、16-bit 固定です)

EPROM は 40pin DIP タイプの 27C1024, 27C2048, 27C4096 (120nS 以下) が使用できます。出荷時には、モニタが格納された 27C1024 が実装されています。

ウェイト制御は CPU 内部のバスコントローラで行ないます。ROM の領域は、120nS 以上のアクセスタイムが必要です。

【注意】 SDRAM, SRAM を 16bit 幅で使用する場合は、SW2-1:BSIZE32 を ON にした上で、メモリコントローラを設定してください。SDRAM, SRAM を片方だけ 16bit で使用することはできません。

8. IO マップ

IO としては、Audio, UART/PRINTER, TIC, PIO 等の IO デバイスと EXT-BUS の IO 空間があります。これらは CS5 の空間にマップされており、ここではそれらのマップと I/O デバイスについて説明します。

8.1. IO 一覧

IO の一覧を下表に示します。

アドレス	機能
4500-0000H	DIPSW2 参照
4500-1000H	DIPSW1 参照
4500-2000H	7 セグメント LED 表示データ設定
4500-5000H	コマンド設定 / 参照(Time-Over-Enable)
4500-6000H	EXT-BUS-High Addr 設定 / 参照(EXA23,EXA22)
4500-8000H ~ 4500-801FH	UART-CH#1(TL16PIR552)設定 / 参照
4500-9000H ~ 4500-901FH	UART-CH#2(TL16PIR552)設定 / 参照
4500-A000H ~ 4500-A01FH	PRINTER-PP(TL16PIR552)設定 / 参照
4500-B000H ~ 4500-B00FH	タイマコントローラ(uPD71054)設定 / 参照
4500-D000H ~ 4500-D01FH	割り込みコントローラ設定 / 参照
4500-E000H ~ 4500-E01FH	PRINTER-ECP(TL16PIR552)設定 / 参照
4540-0000H ~ 457F-FFFFH	EXT-BUS IO 空間(*1)
4580-0000H ~ 4580-001FH	Audio-Control レジスタ設定 / 参照
4580-1000H ~ 4580-100FH	uPD63310 レジスタ設定 / 参照
4580-2000H	Audio-FIFO データレジスタ設定 / 参照

*1:EXT-BUS IO 空間

EXT-BUS の IO 空間は、上位アドレスに EXT-IO 上位アドレス設定レジスタ(4500-6000H)を用いて、16M バイトの空間をアクセスします。その関係を以下に示します。

EXT-IO 上位アドレス設定 レジスタの設定値[A23,A22]	4540-0000H ~ 457F-FFFFH でアクセスできる EXT-BUS の IO 空間
[0.0]	00-0000H - 3F-FFFFH
[0.1]	40-0000H - 7F-FFFFH
[1.0]	80-0000H - BF-FFFFH
[1.1]	C0-0000H - FF-FFFFH

8.2. DIPSW2 読み出しポート(4500-0000H [Read Only])

DIP-SW2 の状態を読み出すためのポートです。データ・フォーマットを下表に示します。

論理アドレス	データバス								内容
	D7	D6	D5	D4	D3	D2	D1	D0	
4500-0000H 入力	SW2 -8	SW2 -7	SW2 -6	SW2 -5	SW2 -4	SW2 -3	SW2 -2	SW2 -1	0=ON 1=OFF
ハード割り付け	BNK_ MOD E1	BNK_ MOD E0	ROM_ TYPE 1	ROM_ TYPE 0	PT_ PRT EN	CPU_ CMODE	BCLK _Hi	BSIZE 32	

SW2-1 が SW2 の”1”のスイッチに、SW2-8 が SW2 の”8”のスイッチに対応しています。該当するビットのスイッチが ON で 0、OFF で 1 が読み出されます。SW2 は、主にハードウェアの環境設定に使用しているスイッチです。設定方法は、「5.4 スイッチ 2 (SW2)」を参照してください。

8.3. DIPSW1 読み出しポート(4500-1000H [Read Only])

DIP-SW1 の状態を読み出すためのポートです。データ・フォーマットを下表に示します。

論理アドレス	データバス								内容
	D7	D6	D5	D4	D3	D2	D1	D0	
4500-1000H 入力	SW1 -8	SW1 -7	SW1 -6	SW1 -5	SW1 -4	SW1 -3	SW1 -2	SW1 -1	0=ON 1=OFF

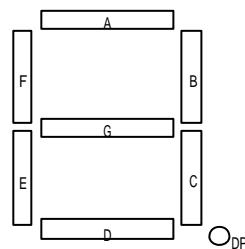
SW1-1 が SW1 の”1”のスイッチに、SW1-8 が SW1 の”8”のスイッチに対応しています。また、該当するビットのスイッチが ON で 0 が、OFF で 1 が読み出されます。SW1 は、モニタの動作設定用のスイッチとして使用しています。設定方法は、「13.1.2 SW1 の設定、14.1.1 SW1 の設定」の各項を参照してください。

8.4. 7セグメント LED 表示データ出力ポート(4500-2000H [Write Only])

7セグメント LED に表示するデータを設定します。データ・フォーマットを下表に示します。該当するビットに 0 を設定すると対応するセグメントが点灯します。

論理アドレス	データバス								内容
	D7	D6	D5	D4	D3	D2	D1	D0	
4500-2000H 出力	7SEG -DP	7SEG -G	7SEG -F	7SEG -E	7SEG -D	7SEG -C	7SEG -B	7SEG -A	0=点灯 1=消灯

7セグメント LED のビット対応は、下図の通りです。



8.5. コマンドレジスタ(4500-5000H [Read/Write])

本レジスタは、以下の機能が割り付けられています。

論理アドレス	レジスタ	データバス							
		D7	D6	D5	D4	D3	D2	D1	D0
4500-5000H	CMD#1 (初期値)	X	X	X	X	X	X	X	TOVEN (0)

TOVEN:タイムオーバー機能の使用を制御します。タイムオーバー機能は、バスサイクルが 512 バスクロック以上となった場合に、強制的に READY-を返してバスサイクルを終了させます。

TOVEN	タイムオーバー機能
0	使用しない
1	使用する

8.6. EXT-IO 上位アドレス設定レジスタ(4500-6000H [Read/Write])

本レジスタは、外部拡張バス (EXT バス) の IO 空間をアクセスする場合の上位アドレス(A23,A22)を設定するためのポートです。

論理アドレス	レジスタ	データバス							
		D7	D6	D5	D4	D3	D2	D1	D0
4500-6000H	CMD#1 (初期値)	X	X	X	X	X	X	EA23 (0)	EA22 (0)

8.7. UART/PRINTER (TL16PIR552) (4500-8000H ~ 4500-A03EH)

UART/PRINTER として TEXAS INSTRUMENTS 製の TL16PIR552(DUAL UART WITH 1284 PARALLEL PORT)LSI を使用しています。TL16PIR552 は、UART を 2 チャンネル、IEEE1284 準拠の双方向プリンタ・ポートを 1 チャンネル備えており、UART の送受信部には 16 キャラクタ分の FIFO バッファを持ち、RTS/CTS フローを自動的に制御する機能を備えているため、最小限の割り込みでオーバーラン・エラーを押さえられます。

TL16PIR552 の各レジスタは、表のように割り付けられています。各レジスタの機能については、TL16PIR552 のマニュアルを参照してください (TL16PIR552 のマニュアルは米国 TEXAS INSTRUMENTS 社のホームページ (<http://www.ti.com/>) の TI&ME のコーナーで入手可能です)

アドレス	機能	読み出し	書き込み
4500-8000H	UART-CH#0	RBR/DLL	THR/DLL
4500-8004H		IER/DLM	IER/DLM
4500-8008H		IIR	FCR
4500-800CH		LCR	LCR
4500-8010H		MCR	MCR
4500-8014H		LSR	LSR
4500-8018H		MSR	MSR
4500-801CH		SCR	SCR
4500-9000H	UART-CH#1	RBR/DLL	THR/DLL
4500-9004H		IER/DLM	IER/DLM
4500-9008H		IIR	FCR
4500-900CH		LCR	LCR
4500-9010H		MCR	MCR
4500-9014H		LSR	LSR
4500-9018H		MSR	MSR
4500-901CH		SCR	SCR
4500-A000H	PRINTER(PPCS-)	DATA	DATA/ECPAFIFO
4500-A 004H		DSR	-----
4500-A 008H		DCR	DCR
4500-A 00CH		EPPADDR	EPPADDR
4500-A 010H ~ 4500-A 01CH		EPPDATA	EPPDATA
4500-E000H	PRINTER(ECPCS-)	PPDATAFIFO/ TESTFIFO/CNFGA	PPDATAFIFO/ TESTFIFO
4500-E004H		CNFGB	-----
4500-E008H		ECR	ECR

TL16PIR552 レジスタ配置

TL16PIR552 の XIN 入力には 22.1184MHz のクロックが接続されています。

UART-CH#0、UART-CH#1、PRINTER の各割り込みは PIC を介して、CPU の割り込みに接続することができます。尚、PRINTER からの割込みは、SW3-8 を介して直接 V832 の INTP10 に接続することもできます。

割り込み発生元	接続 CPU 割り込み	割り込みエッジ
PRINTER	INTP10	立ち上がりエッジ

UART-CH#0 はボードのリアパネルにある JSIO1 コネクタに、UART-CH#1 は JSIO2 コネクタ、PRINTER は JPRT に接続しています。また、UART-CH#0 はデバッガをシリアル通信で用いる場合に使用され、その時の割り込みは、PIC を通して、INTP3 または、NMI を使用します。

TL16PIR552 は、システム・リセットによってリセットされます。

8.8. TIC (uPD71054) (4500-B000H ~ 4500-B00CH)

TIC は NEC 製の uPD71054 が実装されています。uPD71054 は Intel 製の i8254 と互換であり、3 つ のタイマ / カウンタを持っています。これらのタイマ / カウンタにより、モニタのタイマ割り込みの生成を行っています。

TIC の各レジスタは、表の通りに割り当てられています。

アドレス	読み出し	書き込み
4500-B000H	COUNTER#0	COUNTER#0
4500-B004H	COUNTER#1	COUNTER#1
4500-B008H	COUNTER#2	COUNTER#2
4500-B00CH	-----	Control Word

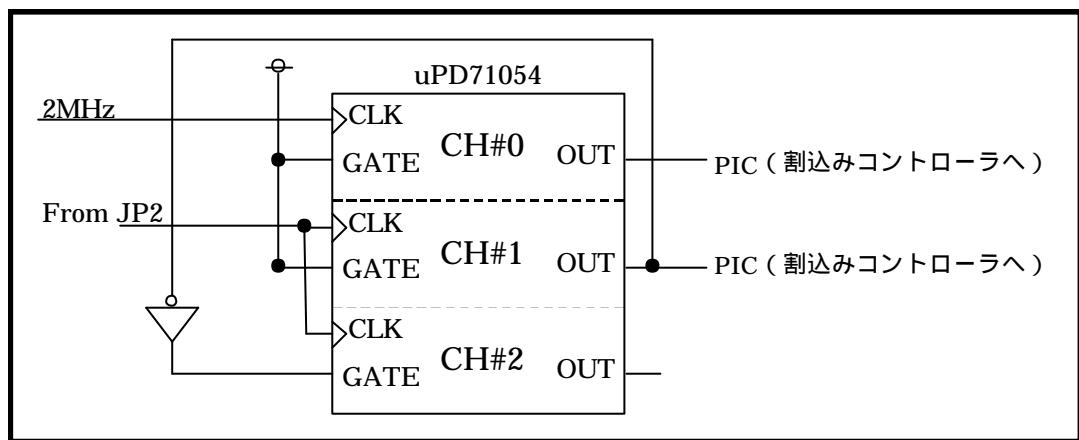
TIC のレジスタ配置

TIC の各チャンネルは下図のように接続されています。

チャンネル 0 は、モニタ・プログラムのインターバル・タイマとして使用されます。

チャンネル 1, 2 は、ユーザのプログラムで自由に使用することができます。

チャンネル 2 は、チャンネル 1 とカスケード接続されています。



使用モード例

- CH#0 : モード 2 (レートジェネレータ)
- CH#1 : モード 2 (レートジェネレータ)
- CH#2 : モード 0 (ダウンカウンタ)

8.9. 割り込みコントローラ (PIC) (4500-D000H~4500-D018H)

PIC は、主に割り込み関係の制御を行ないます。レジスタ割り付けは下表の通りです。

RTE-V832-PC では、PIC の INT0 は、NMI/INT3 の指定により、V832 の NMI か INTP03 に接続され、INT1 は INTP02 に接続されます。

論理アドレス	レジスタ	データバス							
		D7	D6	D5	D4	D3	D2	D1	D0
4500-D000H	PIC INT0M	0	IM06	IM05	IM04	IM03	IM02	IM01	IM00
4500-D008H	PIC INT1M	0	IM16	IM15	IM14	IM13	IM12	IM11	IM10
4500-D010H	PIC INTR	IR7	IR6	IR5	IR4	IR3	IR2	IR1	IR0
4500-D018H	PIC INTEN	0	0	0	NMI/ INT3-	0	0	0	NMI EN

INT0M,INT1M レジスタはそれぞれ INT0,INT1 に入力する割り込みをマスクします。IM0x, IM1x のビットが”1”の時にイネーブルとなり、複数ビットを選択した場合にはそれぞれの OR で割り込みがアクティブとなります。

【注意】 INT0M,INT1M の bit7 は、常に 0 を書き込んでください。

INTR レジスタは割り込みステータスで、割り込み要求がある場合に”1”が読み出せます。これはマスク状態に関係ありません。またエッジ割り込み要求の解除（クリア）には、このレジスタの対応ビットに”1”を書き込みます。

IM0[0..7],IM1[0..7].IR[0..7]の各ビットに割り付けられている割り込み要因は以下の通りです。

IM0,IM1,IR	割り込み要因	要求レベル
0	タイマ0（モード2）	エッジ（立ち上がり）
1	シリアル0	レベル（High）
2	ホスト（PCI通信）	レベル（Low）
3	タイムオーバー	エッジ（立ち上がり）
4	タイマ1（モード2）	エッジ（立ち上がり）
5	シリアル1	レベル（High）
6	パラレル（プリンタ）	エッジ（立ち上がり）
7	リザーブ	エッジ（立ち下がり）

INTEN レジスタは、割り込み全体のイネーブル / ディセーブルなどを制御します。

NMIEN: ノンマスカブル割り込み（NMI）をハード的に禁止することができます。この時、NMI 端子の状態は High レベルとなります。JROM_EM1 コネクタからの NMI 要求は、このビットに関係なく、CPU へ入力されます。

NMIEN	NMI
0	マスクする
1	マスクしない

NMI/INTP3-: INT0 の割り込みを NMI と INTP03 のどちらに入れるかの指定です。

NMI/INTP3-	INT0
0	INTP03
1	NMI

【注意】 INT0 (NMI/INTP03) はモニタで使用していますので、関連するレジスタを変更しないでください。INT1 は解放されていますので、自由に使用可能です。

8.10. AUDIO コントローラ (AUDCNT) (4580-0000H ~ 4580-0010H, 4580-2000H)

AUDCNT は、AUDIO チップ(uPD63310)とのデジタルデータ入出力を制御します。

論理アドレス	レジスタ	データバス							
4580-0000H	CONTROL	D15	D14	D13	D12	D11	D10	D9	D8
		RST	0	0	0	0	RM1	RM0	REC
		D7	D6	D5	D4	D3	D2	D1	D0
		0	0	0	0	0	PM1	PM0	PLY
4580-0008H	STATUS	D15	D14	D13	D12	D11	D10	D9	D8
		0	FF1	0	EF1	0	ROV	RUD	REX
		D7	D6	D5	D4	D3	D2	D1	D0
		0	FF0	0	EF0	0	POV	PUD	PEX
4580-0010H	MCLKDIV	D15	D14	D13	D12	D11	D10	D9	D8
		0	0	0	0	0	0	0	0
		D7	D6	D5	D4	D3	D2	D1	D0
		0	0	0	DIV4	DIV3	DIV2	DIV1	DIV0
4580-2000H	AUDIO DATA	D15 ~ D0							
		1回目：L チャンネル 16bits、2回目：R チャンネル 16bits							

CONTROL レジスタは、音声の録音 / 再生を制御するレジスタです。 (Read/Write)

RST	Audio リセット
0	リセット解除 (リセット値)
1	リセット

PLY	再生動作
0	停止 (リセット値)
1	起動

PM0	再生時アンダーフロー割り込み
0	ディセーブル (リセット値)
1	イネーブル

PM1	再生時オーバーフロー割り込み
0	ディセーブル (リセット値)
1	イネーブル

REC	録音動作
0	停止 (リセット値)
1	起動

RM0	録音時アンダーフロー割り込み	
0	ディセーブル	(リセット値)
1	イネーブル	

RM1	録音時オーバーフロー割り込み	
0	ディセーブル	(リセット値)
1	イネーブル	

STATUS レジスタは、リード専用レジスタで各種のステータスを示します。

PEX	再生状態	
0	停止中	
1	実行中	

POV	再生時オーバーフロー	
0	なし	
1	発生	

PUD	再生時アンダーフロー	
0	なし	
1	発生	

FF0	EF0	再生用データバッファステータス
0	1	エンプティ
1	0	フル(バッファ書き込み禁止)

REX	録音状態	
0	停止中	
1	実行中	

ROV	録音時オーバーフロー	
0	なし	
1	発生	

RUD	録音時アンダーフロー	
0	なし	
1	発生	

FF1	EF1	録音用データバッファステータス
0	1	エンプティ(バッファ読み出し禁止)
1	0	フル

MCLKDIV レジスタは、MCLK の周波数を決定します。

DIV4	DIV3	DIV2	DIV1	DIV0	MCLK 49.152 / (DIV+2)	サンプリング 周期 (MCLK / 256)	バイト / 秒 fs * 4
0	0	0	0	0	24.576 MHz		
0	0	0	0	1	16.384 MHz		
0	0	0	1	0	12.288 MHz	48.0 KHz	192.0 KB
0	0	0	1	1	9.830 MHz	38.4 KHz	153.6 KB
0	0	1	0	0	8.192 MHz	32.0 KHz	128.0 KB
0	0	1	0	1	7.022 MHz	27.5 KHz	109.7 KB
0	0	1	1	0	6.144 MHz	24.0 KHz	96.0 KB
0	0	1	1	1	5.461 MHz	21.3 KHz	85.3 KB
0	1	0	0	0	4.915 MHz	19.2 KHz	76.8 KB
0	1	0	0	1	4.468 MHz	17.5 KHz	69.8 KB
0	1	0	1	0	4.096 MHz	16.0 KHz	64.0 KB
0	1	0	1	1	3.780 MHz	14.8 KHz	59.1 KB
0	1	1	0	0	3.511 MHz	13.7 KHz	54.9 KB
0	1	1	0	1	3.277 MHz	12.8 KHz	51.2 KB
0	1	1	1	0	3.072 MHz	12.0 KHz	48.0 KB
0	1	1	1	1	2.891 MHz	11.3 KHz	45.2 KB
1	0	0	0	0	2.731 MHz	10.7 KHz	42.7 KB
1	0	0	0	1	2.587 MHz	10.1 KHz	40.4 KB
1	0	0	1	0	2.458 MHz	9.6 KHz	38.4 KB
1	0	0	1	1	2.341 MHz	9.1 KHz	36.6 KB
1	0	1	0	0	2.234 MHz	8.7 KHz	34.9 KB
1	0	1	0	1	2.137 MHz	8.3 KHz	33.4 KB
1	0	1	1	0	2.048 MHz	8.0 KHz	32.0 KB
1	0	1	1	1	1.966 MHz	7.7 KHz	30.7 KB
1	1	0	0	0	1.890 MHz	7.4 KHz	29.5 KB
1	1	0	0	1	1.820 MHz	7.1 KHz	28.4 KB
1	1	0	1	0	1.755 MHz	6.9 KHz	27.4 KB
1	1	0	1	1	1.695 MHz	6.6 KHz	26.5 KB
1	1	1	0	0	1.638 MHz	6.4 KHz	25.6 KB
1	1	1	0	1	1.586 MHz	6.2 KHz	24.8 KB
1	1	1	1	0	1.536 MHz	6.0 KHz	24.0 KB
1	1	1	1	1	1.489 MHz	5.8 KHz	23.3 KB

8.11. uPD63310 レジスタ(4580-1000H ~ 4580-100FH)

uPD63310 のレジスタは、以下の通り割り付けられます。詳細には、uPD63310 のデータシートを参照ください。

アドレス	機能	D5	D4	D3	D2	D1	D0
4580-1000H	アドレス・レジスタ						レジスタ番号
4580-1008H	データ・レジスタ						利得コントローラータ

9. 割込みと DMA

RTE-V832-PC の割込みと DMA について説明します。

9.1. 割込み

外部割込み	割込み信号	備考
NMI-	NMI	PIC の INT0(MONITOR 専用)の割り込み要求です。
INTP00	INT_AUDIO	エラーが発生した場合に割り込みを発生できます。
INTP01	EXTBUS-INT0	EXT-BUS の INT0 からの割り込み要求です。
INTP02	PIC-INT1	PIC の INT1(USER 解放)からの割り込み要求です。
INTP03	PIC-INT0	PIC の INT0(MONITOR 専用)の割り込み要求です。
INTP10	INT_PRN	プリンタ IF からの割り込み要求です。
INTP11	EXT-BUS INT1	EXT-BUS の INT1 からの割り込み要求です。
INTP12	EXT-BUS INT2	EXT-BUS の INT2 からの割り込み要求です。
INTP13	EXT-BUS INT3	EXT-BUS の INT3 からの割り込み要求です。

備考 :

1. 割込みは、NMI を除き全て正論理の信号要求です。
2. 全ての信号は、SW3 で物理的に切り離しできます。(出荷時は全て接続している状態です)
4. EXT-BUS からの割込みは全てインバートしたものが V832 に接続されています。
3. ボード上の割り込みコントローラ (「8.9 割り込みコントローラ (PIC) (4500-D000H ~ 4500-D018H)」参照) により、以下の割り込みソースから任意の割り込み要求を選択し、2種の割り込み(INT0, INT1)を生成できます。INT0 はシステム用 (モニタで使用) INT1 はユーザアプリケーション用です。

PIC で選択できる割り込み要因一覧
タイマ 0 (モード 2)
シリアル CH#0
ホスト (PCI 通信)
タイムオーバー
タイマ 1 (モード 2)
シリアル CH#1
パラレル (プリンタ)

9.2. NMI の使用方法

本ボード上にモニタ等を移植する場合に NMI を使用する方法を説明します。NMI はエッジ検出されますが、割り込みソースはレベル出力のため NMI をハード的にマスクできるようにしています。マスク方法については、「8.9 割り込みコントローラ (PIC) (4500-D000H ~ 4500-D018H)」の INTEN レジスタに関する記述を参照してください。

NMI が発生した場合は、下記の手順で処理します。

PIC の NMIEN に”0”を設定して、NMI をハード的にマスクする。

PIC の INTR を検査する。

要求元のための NMI 処理を行ない、要求をクリアする。

PIC の NMIEN に”1”を設定して、マスクを解除する。

NMI 処理から復帰する。

【注意】モニタを使用している時に、NMI, INTP03, PIC の INT0 に関連したレジスタの書き込み操作を行なうとモニタがハングアップする場合があります。

9.3. DMA 要求

DMARQ-/AK-	DMARQ 信号	備考
CH0	再生時の要求	再生時に Audio のデータバッファにデータを書き込むことを要求する DMA 要求です。間に合わなかつた場合、アンダーラン・エラーが発生します。
CH1	録音時の要求	録音時に Audio のデータバッファからデータを読み出すことを要求する DMA 要求です。間に合わなかつた場合、オーバフロー・エラーが発生します。
CH2	EXT-BUS からの DMARQ0-の要求	
CH3	EXT-BUS からの DMARQ1-の要求	

備考 :

- 1 . DMARQ-/AK-[3..0]は、共に負論理に設定します。
- 2 . CH0,1 は、JP1、CH2,3 は、SW4 で物理的に切り離しできます。(出荷時は接続している状態です)

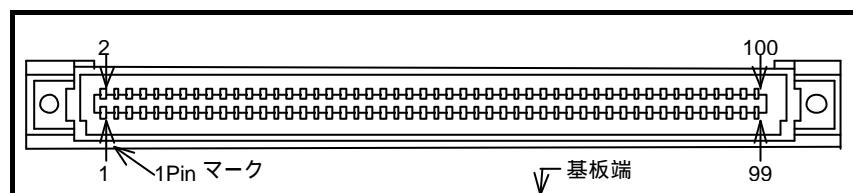
10.EXT-BUS バス仕様

JEXT コネクタは、メモリや I/O などを拡張できるように用意された EXT-BUS 用のコネクタです。
このコネクタには、本ボードの内部のローカル・バスが接続されています。

10.1. ピン配置

番号	信号名	番号	信号名	番号	信号名	番号	信号名
1	GND	2	+5V	3	D0	4	D1
5	D2	6	D3	7	GND	8	D4
9	D5	10	D6	11	D7	12	GND
13	D8	14	D9	15	D10	16	D11
17	GND	18	D12	19	D13	20	D14
21	D15	22	GND	23	D16	24	D17
25	D18	26	D19	27	GND	28	D20
29	D21	30	D22	31	D23	32	GND
33	D24	34	D25	35	D26	36	D27
37	GND	38	D28	39	D29	40	D30
41	D31	42	GND	43	+5V	44	GND
45	Reserve	46	Reserve	47	(A1)	48	A2
49	A3	50	A4	51	GND	52	A5
53	A6	54	A7	55	A8	56	A9
57	A10	58	GND	59	A11	60	A12
61	A13	62	A14	63	A15	64	A16
65	GND	66	A17	67	A18	68	A19
69	A20	70	A21	71	A22	72	A23
73	GND	74	+5V	75	MRD-	76	Reserve
77	MWR0-	78	MWR1-	79	MWR2-	80	MWR3-
81	IORD-	82	IOWR-	83	GND	84	READY
85	GND	86	INT0-	87	INT1-	88	INT2-
89	INT3-	90	DMARQ0-	91	DMARQ1-	92	DMAAK0-
93	DMAAK1-	94	RESET-	95	32/16BIT-	96	N/C
97	+5V	98	GND	99	CLK	100	GND

JEXT コネクタピン配置



JEXT のピン配置

10.2. 信号

信号名	入出力	機能
D[0..31]	入出力	データ・バス信号。CPUのデータ・バス信号をバッファして接続。ボード上で10K ブルアップ。
A[1..23]	出力	アドレス・バス信号。CPUのアドレス信号をバッファして接続。
MRD-	出力	メモリ・リード・サイクルのタイミング信号。EXT-BUS空間のアクセス時のみ、アクティブになる。
MWR-[0..3]	出力	メモリ・ライト・サイクルのタイミング信号。それぞれ、MWR0-はD[0..7]に、MWR1-はD[8..15]に、MWR2-はD[16..23]に、MWR3-はD[24..31]に対応。EXT-BUS空間のアクセス時のみ、アクティブになる。
IORD-	出力	I/Oリード・サイクルのタイミング信号。EXT-BUS空間のアクセス時、もしくはフライバイ DMA サイクル(参考)の時アクティブになる。
IOWR-	出力	I/Oライト・サイクルのタイミング信号。EXT-BUS空間のアクセス時、もしくはフライバイ DMA サイクル(参考)の時アクティブになる。
READY	入力	サイクルの終了をCPUに通知する信号。EXT-BUS空間のみで有効。確実にCPUにREADYを認識させるためには、MRD-,MWR-[0..3],IORD-,IOWR-がインアクティブになるまでREADYをアクティブに保つことが必要。ボード上で10K ブルアップ。
INT-[0..3]	入力	Lowアクティブの割り込み要求信号。インバータでバッファ後SW3を経由して、それぞれCPUのINTP01,INTP11,INTP12,INTP13端子に接続されている。ボード上で10K ブルアップ。(「9 割込みとDMA」参照)
DMARQ-[0..1]	入力	LowアクティブのDMA要求信号。バッファ後SW4を経由して、それぞれCPUのDMARQ2-,DMARQ3-端子に接続されている。ボード上で10K ブルアップ。(「9 割込みとDMA」参照)
DMAAK-[0..1]	出力	LowアクティブのDMA応答信号。CPUのDMAAK0-,DMAAK1-が、SW4を経由後、バッファされて接続されている。(「9 割込みとDMA」参照)
RESET-	出力	Lowアクティブのシステム・リセット信号。
32/16BIT-	入力	この信号をLowにすると、CPUが16Bitデータバスの場合、D[15..0]のみが使用される。また、CPUが32Bitデータバスの場合、アドレスによりD[15..0]もしくはD[31..16]のいずれかが使用される(16ビットバスモード)。Highにすると、データバスのD[31..0]が使用される(32ビットバスモード)。この信号はダイナミックに変化させてはならない。ボード上で10K ブルアップ。
CLK	出力	クロック信号。V832のCLKOUT端子がバッファ後、接続されている。
Reserve		予約信号。EXT-BUSを使用するボードは、この端子に何も接続しないこと。

JEXT コネクタ信号

注意事項 :

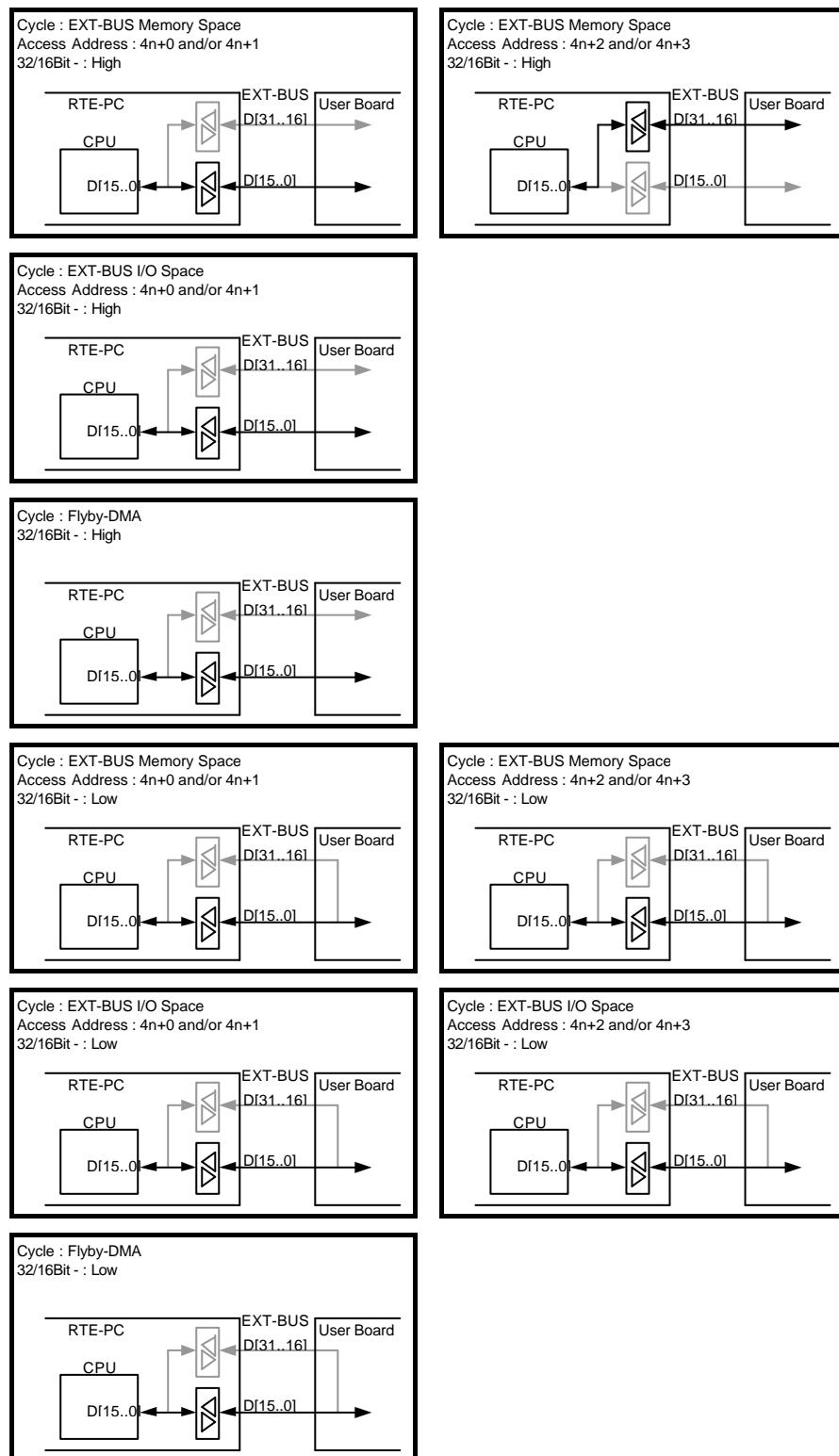
- 32/16BIT-信号は、将来のRTE-PCシリーズの全てでサポートされるとは限りません。EXT-BUSに接続するボードを将来のRTE-PCシリーズでも使用する予定の場合は、32ビットバスモードで動作するように設計してください。
32/16BIT-がLowの時は、MWR2-およびMWR3-がアサートされることはありません。代わりにMWR0-およびMWR1-がアサートされます。
また、32/16BIT-をLowにして使用する場合は、EXT-BUSに接続するボード上でD[15..0]とD[31..16]をショートしてください(『10.3 データバスの接続』参照)。
- A1は32/16BIT-信号がLowの時有効です。したがって、32/16BIT-信号がサポートされていない将来のRTE-PCシリーズでは、A1が出力されないことがあります。
- EXT-BUSバスの1回のサイクルでの最大アクセス・バス幅は、CPUのデータ・バスのバス幅に依存します。V832は32bitのデータバスですので問題にはなりませんが、16bitのCPUを搭

載した製品の場合、1回でアクセスできるデータサイズは最大 16 ビットまでに制限されます。このことは、本バスに汎用的に使用する目的のボードを設計する場合、考慮する必要があります。

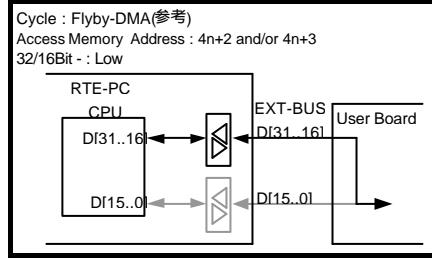
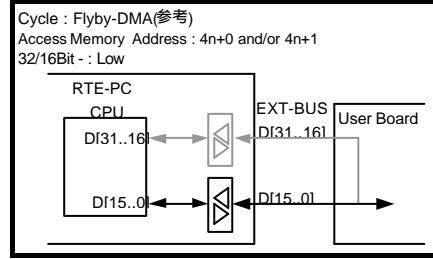
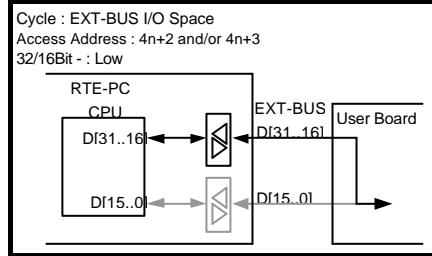
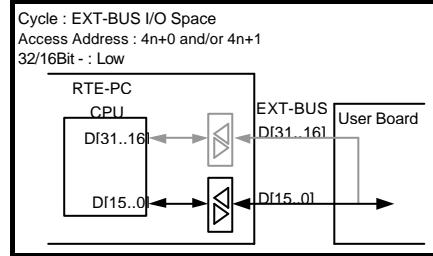
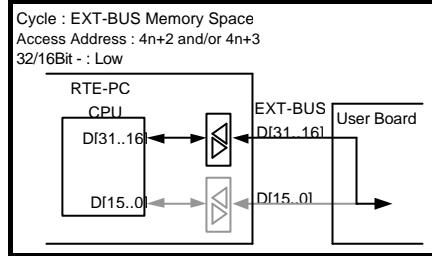
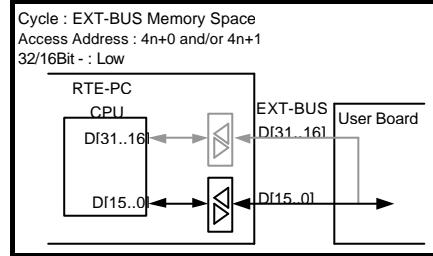
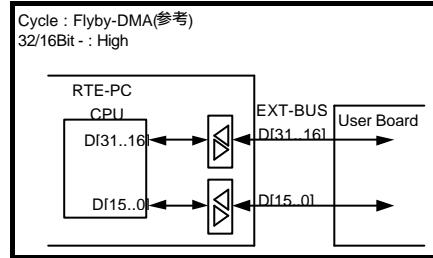
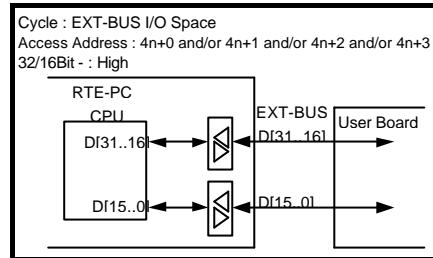
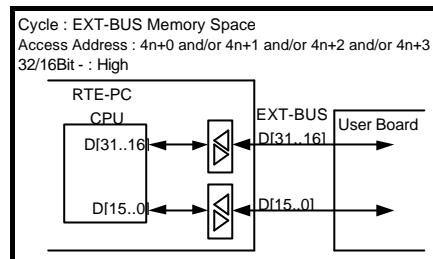
4. DMA 機能は、将来の RTE-PC シリーズの全てでサポートされるとは限りません。

10.3. データバスの接続

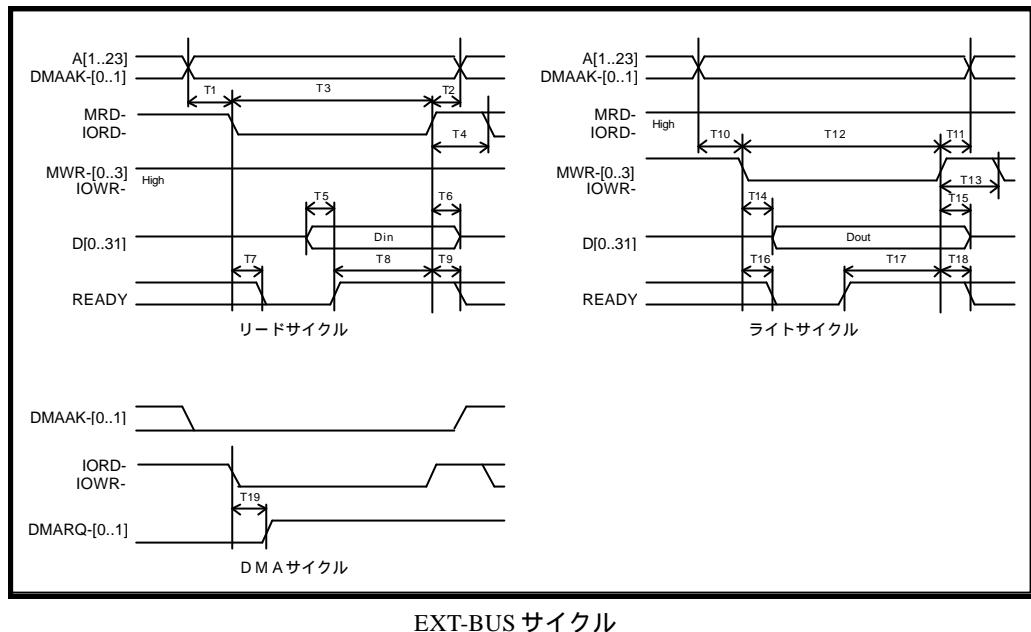
10.3.1. 16 ビット・データバス CPU (参考)



10.3.2. 32 ビット・データバス CPU (V832 の場合)



10.4. タイミング



EXT-BUS サイクル

記号	内容	MIN(ns)	MAX(ns)
T1	ADDR,DMAAK- MRD-,IORD- セットアップ時間	10	
T2	MRD-,IORD- ADDR,DMAAK- ホールド時間	10	
T3	MRD-,IORD- サイクル時間	50	
T4	MRD-,IORD- サイクル間隔	20	
T5	RD DATA RD READY セットアップ時間	0	
T6	MRD-,IORD- RD DATA ホールド時間	0	
T7	MRD-,IORD- RD READY ディレイ時間		20
T8	RD READY MRD-,IORD- ディレイ時間	15	
T9	MRD-,IORD- RD READY ホールド時間	0	
T10	ADDR,DMAAK- MWR-,IOWR- セットアップ時間	10	
T11	MWR-,IOWR- ADDR,DMAAK- ホールド時間	10	
T12	MWR-,IOWR- サイクル時間	50	
T13	MWR-,IOWR- サイクル間隔	20	
T14	MWR-,IOWR- WR DATA ディレイ時間		20
T15	MWR-,IOWR- WR DATA ホールド時間	10	
T16	MWR-,IOWR- WR READY ディレイ時間		20
T17	WR READY MWR-,IOWR- ディレイ時間	0	
T18	MWR-,IOWR- WR READY ホールド時間	0	
T19	IORD-,IOWR- DMARQ- インアクティブ遅延時間		xx

EXT-BUS AC スペック

10.5. 適合コネクタ

EXT-BUS に使用しているコネクタと、そのコネクタに勘合する適合コネクタの型番を以下に示します。
複数のボードを EXT-BUS に接続する場合は、ケーブルを使用してデージーチェーン接続を行います。

EXT-BUS 使用コネクタ	: KEL 社 8830E-100-170S
適合コネクタ (基板用)	: KEL 社 8802-100-170S
適合コネクタ (ケーブル用)	: KEL 社 8825E-100-175
対ケーブル用ライトアングル (基板用)	: KEL 社 8830E-100-170L KEL 社 8831E-100-170L

10.6. 注意事項

EXT-BUS に接続するボードを設計する上での注意事項を以下に示します。

- 1 . 複数のボードを EXT-BUS に接続する場合は、READY 信号はボードが選択されているときのみドライブするように、Hi-Z 制御を行わなければなりません。
- 2 . EXT-BUS のサイクルにウェイトを挿入するためには、T7 および T16 を満足する必要があります。その為には、ノーマルノットレディ (通常、ノットレディの状態で保持し、アクセスが発生し、スレーブデバイスが準備できた時点でレディを返す手法) で制御することを推奨します。
- 3 . DMA サイクルをシングル転送モードで行う場合、次の DMA サイクルを確実に発生させないためには、タイミング図の T19 を満足する必要があります。ただし、この T19 は使用している DMA コントローラの仕様に大きく依存するため、各製品で異なります。
- 4 . I/O 空間のアクセスは、RTE-V832-PC では EXT-BUS の 16M バイトの I/O 空間をアクセスする為の十分な空間がありませんので、上位の 2bit(A23,A22)は、EXT-IO 上位アドレス設定レジスタ #1 ポートからの出力で贿います。このポートの詳細は、「8.6 EXT-IO 上位アドレス設定レジスタ(4500-6000H [Read/Write])」を参照ください。

11.ソフトウェア

RTE-V832-PC ボードのハードウェアの初期化と周辺デバイスの使用方法について説明します。

11.1. 初期化

ブートルーチンでは外部メモリ / I/O をアクセスするために V832 の内部バスコントローラを以下の通り初期化します。

バスクロック : 33MHz 以下の場合

レジスタ	内部 I/O アドレス	設定値	サイズ	備考
BCTC	C000-0010H	22H	byte	
DBC	C000-0012H	00H(43H)	byte	() 内は、SW2-1(BSIZE32) が ON(0)の時の設定値
PWC0	C000-0014H	7000H	half-word	
PWC1	C000-0016H	4000H	half-word	
PIC0	C000-0100H	0000H	half-word	
PIC1	C000-0102H	0000H	half-word	
SDC	C000-0020H	9401H	half-word	
RFC	C000-0022H	0102H 8102H	half-word	1回目 33MHz >>11.5 μS 2回目 Refresh Enable
PAC	C000_00F4	FFH	byte	PortA:select DMACont sig.
PBC	C000_00FA	FFH	Byte	PortB:select INTPxx

バスクロック : 33MHz 越えの場合

レジスタ	内部 I/O アドレス	設定値	サイズ	備考
BCTC	C000-0010H	22H	byte	
DBC	C000-0012H	00H(43H)	byte	() 内は、SW2-1(BSIZE32-) が ON(0)の時の設定値
PWC0	C000-0014H	7200H	half-word	
PWC1	C000-0016H	7111H	half-word	
PIC0	C000-0100H	0000H	half-word	
PIC1	C000-0102H	0000H	half-word	
SDC	C000-0110H	941AH	half-word	
RFC	C000-0022H	0104H 8104H	half-word	1回目 47.6MHz >>13.4 μS 2回目 Refresh Enable
PAC	C000_00F4	FFH	byte	PortA:select DMACont sig.
PBC	C000_00FA	FFH	Byte	PortB:select INTPxx

レジスタの詳細については、V832 CPU マニュアルを参照ください。

11.2. ライブライ

Cコンパイラでプログラムする時に必要となるI/Oアクセスなどのライブラリです。ただし、これらの記述やパラメータ受け渡し方法などは、MULTI環境でのものです。他のコンパイラ等を使用する場合には、変更が必要となる場合があります。

```
/* I/O入出力ライブラリ */

/* GHS V800コンパイラ パラメータ受け渡し */
/* arg0: r6, arg1: r7, arg2: r8, return: r10 */

inb(int addr)           /* バイト(8ビット)入力 */
{
    __asm__(" in.b 0[r6], r10");
}

inh(int addr)           /* ハーフワード(16ビット)入力 */
{
    __asm__(" in.h 0[r6], r10");
}

inw(int addr)           /* ワード(32ビット)入力 */
{
    __asm__(" in.w 0[r6], r10");
}

outb(int addr, int data) /* バイト(8ビット)出力 */
{
    __asm__(" out.b r7, 0[r6]");
}

outh(int addr, int data) /* ハーフワード(8ビット)出力 */
{
    __asm__(" out.h r7, 0[r6]");
}

outw(int addr, int data) /* ワード(8ビット)出力 */
{
    __asm__(" out.w r7, 0[r6]");
}
```

11.3. タイマの使用法

ボード上の外部タイマ(8254)でカスケード接続されたタイマ1とタイマ2を使用した時間計測のサンプルを示します。タイマ1はインターバルカウンタ(モード2)、タイマ2は、ダウンカウンタ(モード0)として初期化して、時間計測するルーチンの前後でカウンタ値を求めておくことで実行時間が算出できます。ただし、タイマのカウント値はどちらもダウンカウンタとなることに注意してください。また、外部タイマの連続アクセスではコマンドリカバリ(ROM領域のダミーリード)が必要となります。

```
/* タイマによる実行時間計測サンプル */

#define TIMERCLK 2000000 /* 2MHz */
#define INTERVAL (TIMERCLK * 10 / 1000) /* 10ms (1/100) */
#define IOWAIT() (*char *)0x4FFF0000 /* I/Oコマンドリカバリ用 */

InitTimer() /* タイマ初期化 */
{
    outb(0x4500B00C, 0x74); IOWAIT(); /* タイマ1 モード2 */
    outb(0x4500B004, INTERVAL); IOWAIT(); /* タイマ1 下位カウント */
    outb(0x4500B004, INTERVAL / 256); IOWAIT(); /* タイマ1 上位カウント */
    outb(0x4500B00C, 0xB0); IOWAIT(); /* タイマ2 モード0 */
    outb(0x4500B008, 0xFF); IOWAIT(); /* タイマ2 下位カウント */
    outb(0x4500B008, 0xFF); IOWAIT(); /* タイマ2 上位カウント */
    return 0;
}

LatchTimer() /* カウントラッチ */
{
    int count1, count2, counts;

    outb(0x4500B00C, 0xDC); IOWAIT(); /* タイマ1/2 マルチブルラッチ */
    count1 = inb(0x4500B004); IOWAIT(); /* タイマ1 カウント */
    count1 += inb(0x4500B004) * 256; IOWAIT(); /* タイマ1 カウント */
    count2 = inb(0x4500B008); IOWAIT(); /* タイマ2 カウント */
    count2 += inb(0x4500B008) * 256; IOWAIT(); /* タイマ2 カウント */
    counts = INTERVAL * (0xFFFF - count2)
            + (INTERVAL - count1);
    return counts;
}

double total_time;

main()
{
    int start_count, stop_count;

    InitTimer();
    start_count = LatchTimer(); /* スタートカウント値 */
    func();
    stop_count = LatchTimer(); /* ストップカウント値 */
    total_time = (double)(stop_count - start_count)
                / (double)TIMERCLK; /* 秒数 */
    return 0;
}

#include <time.h>

func() /* 時間計測ルーチン */
{
    ....
}
```

11.4. 音声入出力

ボードに搭載されている音声入出力インターフェースを使用したプログラムサンプルです。データの入出力には、V832 の内蔵 DMA を使用しています。

```
/*
 * 音声入出力サンプル */
#define DMA0 0xC0000030 /* 内蔵 DMA ch0 (再生) */
#define DMA1 0xC0000040 /* 内蔵 DMA ch1 (録音) */
#define AUDIO_DATA 0x45802000 /* AUDIO データ (FIFO) */

static Set63310Reg(int reg, int data) /* uPD63310 レジスタ設定 */
{
    outb(0x45801000, reg); /* アドレスレジスタ ライト */
    outb(0x45801008, data); /* データレジスタ ライト */
    return 0;
}

static Get63310Reg(int reg) /* uPD63310 レジスタ獲得 */
{
    outb(0x45801000, reg); /* アドレスレジスタ ライト */
    return inb(0x45801008) & 0x3F; /* データレジスタ リード (6 ビット) */
}

AudioInit() /* Audio 初期化 */
{
    outh(0x45800000, 0x8000); /* リセット Audio */
    outh(0x45800010, 8); /* fs = 12KHz */
    inh(0x45800010); /* ダミーリード */
    outh(0x45800000, 0); /* リセット解除 */
    Set63310Reg(0, 0); /* IN1L 0db */
    Set63310Reg(1, 0); /* IN1R 0db */
    Set63310Reg(17, 0); /* OUTDACL 0db */
    Set63310Reg(18, 0); /* OUTDACR 0db */
    return 0;
}

AudioTerm() /* Audio 終了処理 */
{
    Set63310Reg(0, 0x20); /* IN1L mute */
    Set63310Reg(1, 0x20); /* IN1R mute */
    Set63310Reg(17, 0x20); /* INDACL mute */
    Set63310Reg(18, 0x20); /* INDACR mute */
    outh(0x45800000, 0); /* stop command */
    return 0;
}

AudioPlay(int addr, int size) /* 再生処理 DMAO 使用 */
{
    outh(DMA0 + 0, addr >> 16); /* DMA-DSA0H */
    outh(DMA0 + 2, addr); /* DMA-DSA0L */
    outh(DMA0 + 4, AUDIO_DATA >> 16); /* DMA-DDA0H */
    outh(DMA0 + 6, AUDIO_DATA); /* DMA-DDA0L */
    size = (size / 2 - 1) * 2; /* DMA 転送カウント数 */
    outh(DMA0 + 8, size >> 16); /* DMA-DBCOH */
    outh(DMA0 + 10, size); /* DMA-DBCOL */
    outh(DMA0 + 12, (0<<12) /* DMA-DCHC0 TTYP DMARQ */
        | (1<<10) /* TBT Mem->I/O */
        | (0<<8) /* SAD Inc */
        | (2<<6) /* DAD Fix */
        | (0<<5) /* DAL Low */
        | (0<<4) /* DRL Low */
        | (1<<3) /* TM Demand */
        | (1<<1) /* DS Half-word */
        | 1); /* Enable */
    outh(0x45800000, 0x0001); /* 再生スタート */
    while ((inh(DMA0 + 12) & 1) != 0) /* DMA 終了を待つ */
        ;
}

while ((inh(0x45800008) & 0x10) == 0)
```

```

        ;
        outh(0x45800000, 0);           /* FIFO エンプティまで待つ */
        /* 再生終了 */
        return 0;
    }

    AudioRecord(int addr, int size)           /* 録音処理 DMA1 使用 */
    {
        outh(DMA1 + 0, AUDIO_DATA >> 16); /* DMA-DSA1H */
        outh(DMA1 + 2, AUDIO_DATA);          /* DMA-DSA1L */
        outh(DMA1 + 4, addr >> 16);        /* DMA-DDA1H */
        outh(DMA1 + 6, addr);              /* DMA-DDA1L */
        size = (size / 2 - 1) * 2;          /* DMA 転送カウント数 */
        outh(DMA1 + 8, size >> 16);        /* DMA-DBC1H */
        outh(DMA1 + 10, size);              /* DMA-DBC1L */
        outh(DMA1 + 12, (0<<12)           /* DMA-DCHC1      TTYP (DMARQ) */
            | (2<<10)                   /* */
            | (2<<8)                    /* SAD fix */
            | (0<<6)                   /* DAD inc */
            | (0<<5)                   /* DAL Low */
            | (0<<4)                   /* DRL Low */
            | (1<<3)                    /* TM demand */
            | (1<<1)                   /* DS half-word */
            | 1);                      /* Enable */
        outh(0x45800000, 0x100);           /* 録音スタート */
        while ((inh(DMA1 + 12) & 1) != 0)
            ;
        outh(0x45800000, 0);           /* DMA 終了を待つ */
        /* 録音終了 */
        return 0;
    }

#define COUNT      0x10000           /* L/R データサンプリング数 */

int buffer[COUNT];           /* L/R データバッファ */

main()
{
    inb(0xC000006E);
    outb(0xC000006E, 1);           /* DMA-DC MEM=1 */
    /* 初期化 */
    AudioInit();
    AudioRecord((int)buffer, sizeof(buffer)); /* 録音 */
    AudioPlay ((int)buffer, sizeof(buffer)); /* 再生 */
    AudioTerm();                   /* 終了 */
    return 0;
}

```

音声データは、Lch(左) / Rch(右)ごとに各 16 ビットデータで、録音 / 再生ともにスタートから必ず L1, R1, L2, R2, ... の順番でデータ入出力します。

注意：録音 / 再生を行う場合は、Multi のタイマを使用しない(SW1-3,SW1-4 を共に ON)の状態に設定してください。タイマー割り込みが入った場合、音声のオーバラン / アンダーランのエラーが発生する場合があります。

12.マスカブル割り込みを使用したアプリケーションの開発

RTE-V832-PC 上でマスカブル割り込みを使用したアプリケーションの開発を行う場合の方法と、制限事項について説明します。

12.1. 割り込みベクタ

V832 の割り込みベクタ領域である FFFF-FE00H ~ FFFF-FFFFH 番地は、ROM により固定されていて書き換えることができません。そこで弊社が提供するモニタでは、SRAM 上に代替えのベクタ領域を用意し、FFFF-FE00H ~ FFFF-FFFFH 番地のベクタには、その代替えベクタ領域への分岐命令が置かれています。例えば、例外コード FE00H の割り込みが発生すると、CPU の割り込み機能により FFFF-FE00H 番地に分岐し、そこには対応する代替えベクタ番地への分岐命令があります。したがって、ユーザ・プログラムでは、この代替えベクタ領域を本来のベクタ領域と同じように書き換えれば、割り込み発生時にユーザ・プログラムの割り込み処理ルーチンに分岐するようにできます。

通常の V832 のプログラムと異なるのは、通常はベクタ領域は ROM 化の時点で固定されており、プログラムで書き換える必要はありません。しかし、モニタ上で動作するプログラムの場合、自分で代替えベクタを書き換えてから割り込みを許可する必要があります。

RTE-V832-PC のモニタでは、代替えベクタ領域を SRAM 上の FE07-0000H ~ FE07-01FFH に確保しています。したがって、例外コード FE00H の割り込みは FE07-0000H 番地、例外コード FE10H の割り込みは FE07-0010H 番地...に目的の割り込み処理に分岐する命令を書き込みます。また、V832 CPU はキャッシュメモリを内蔵していますので、ベクタを書き換えた後にキャッシュをクリアすることが必要です。クリアしないと、書き換え前の命令が実行されてしまう場合があります。

代替えベクタを書き換えるためのプログラム例を下記に示します（割り込み処理ルーチンから代替えベクタ領域への相対アドレスが 26 ビット以内の場合）。

```
#define VECT_CPU      0xffffffe00          /* CPU 割り込みベクタ先頭 */
#define VECT_RAM      0xfe070000          /* 代替え割り込みベクタ先頭 */
#define VECT(n)        ((VECT_CPU - n) + VECT_RAM)  /* 割り込みベクタ番地を求める */

main()
{
    extern void __interrupt IntEntry();          /* 割り込み処理ルーチン */

    /* CPU ベクタの 0xfffffe30 に対応する代替えベクタ番地と
     * 割り込みルーチンへの "JR dest26" 命令の作成 */
    addr = VECT(0xfffffe30);
    offs = (int)IntEntry - addr;
    inst = 0xa8000000 | (offs & 0x3fffffe);      /* 32 ビット命令 JR dest26 */

    /* ベクタの置き換え */
    di();                                         /* 割り込み禁止 __asm(" di"); */
    *((unsigned short*)(addr + 0)) = (inst >> 16) & 0xffff; /* 上位 16 ビットコード */
    *((unsigned short*)(addr + 2)) = (inst       ) & 0xffff; /* 下位 16 ビットコード */
    outw(0xFFFFFFF4, 3);                          /* キャッシュクリア */

    /* 割り込みデバイスの初期化など */
    ....
    ei();                                         /* 割り込み許可 __asm(" ei"); */
    ....
}
```

12.2. 内蔵命令 RAM

V832 は、マスカブル割り込みのベクタを内蔵命令 RAM に配置することができます（システム・レジスタ HCCW の IHA ビット）。この機能を使用した場合は、V832 の通常のプログラムと、RTE-V832-PC 上でモニタを使用したプログラムの場合で、ベクタの設定に関して異なることはありません。

内蔵命令 RAM のベクタの使用方法や、内蔵命令 RAM の内容の変更方法については、CPU のマニュアルを参照してください。

また、割り込み処理に限らずプログラムを内蔵命令 RAM に配置する場合には、コンパイルされたオブジェクトコードに注意する必要があります。特に C の switch-case 文は、命令コード中にジャンプテーブルが作成され、このテーブル参照により分岐を行うコードが生成されることがあります。このテーブル参照は LD 命令で行われますが、内蔵命令 RAM は LD 命令による参照ができないためプログラムは予期しない動作を行う場合があります。

12.3. 一般的な制限事項 / 注意事項

マスカブル割り込みを使用したアプリケーションをデバッグする上での制限事項と注意事項を下記に示します。

- 1) 代替えベクタの設定前に割り込みが発生した場合や、代替えベクタを正しく設定しないで割り込みが発生した場合には、割り込み発生時点でのプログラム位置でブレークします。これは、代替えベクタの初期値がモニタ ROM のブレーク処理ルーチンへの分岐命令になっているためです。
- 2) 代替えベクタ領域から割り込み処理ルーチンまでの相対アドレスが 26Bit を超える場合、割り込み処理ルーチンへの分岐のために、少なくとも 1 つ以上のレジスタの値を壊すか、分岐の中継点を作る必要があります。
- 3) 代替えベクタ領域は、ROM モニタの管理領域として保護されているため、プログラムのダウンロードで書き換えることはできません。 したがって、ソース・プログラム上ベクタ領域を独立したセクションとして定義し、リンク時のパラメータによりそのセクションを代替えベクタ領域に割り付けることはできません。
- 4) 代替えベクタ領域を書き換えた直後に CPU 内蔵のキャッシュ・メモリをフラッシュしてください。この操作を忘ると、代替えベクタ書き換え前のベクタが使用されてしまうことがあります。
- 5) 割り込み関係を含む全てのペリフェラルは、ボード上のリセット・スイッチによってのみ初期化されます。 したがって、一度プログラムを実行した後にプログラムを再ロードして動作させる場合、前のプログラム実行による影響がペリフェラル上に残ってしまいます。ペリフェラルを使用するプログラムを動作させた後に、再度プログラムを始めから動作させる場合は、以下の手順にしたがってください。
 - (1) RTE-V831-PC のリセット・スイッチを押しリセットします。
 - (2) プログラムをリロードして再実行します。
- 6) プログラムの先頭で DI (割り込み禁止) 状態にしてから、ペリフェラルやベクタの設定をした後、EI (割り込み許可) 状態にするようにしてください。

12.4. 割り込み処理内におけるブレーク・ポイント使用に関する制限事項

割り込み処理ルーチン内でブレークした場合の制限事項を以下に記します。

- 1) ブレーク中、全てのマスカブル割り込みは受け付けません。
- 2) シングルステップ機能は、次の命令にテンポラリ・ブレーク・ポイントを設定する方式を取っています。したがって、EI (割り込み許可) 状態のユーザ・プログラムをシングル・ステップする場合、シングルステップ中にも割り込みを受け付けます。
- 3) 割り込みルーチン中でブレーク後に、シングルステップによって割り込み処理ルーチンから抜けることはできません (具体的には、割り込みルーチンの最後の "}" でのシングルステップができません)。同様に、IRET 命令のシングルステップもできません。
- 4) デバッガの "Return" 機能で、割り込み処理ルーチンから元のルーチンへ戻ることはできません。

13.APPENDIX.A MULTI モニタ

MULTI用のモニタ ROM を使用して、ホストの MULTI デバッガと接続して使用する場合の設置方法と使用上の注意事項について説明します。

13.1. ボードの設置

13.1.1. RTE for Win32 のインストール

MULTI デバッガを使用する場合には、PC に通信用のソフトウェア (RTE for Win32) をインストールする必要があります。ソフトウェアのインストールとテストについては、添付の「 RTE for Win32 インストール・マニュアル」を参照してください。

13.1.2. SW1 の設定

SW1 は、汎用の入力ポートのスイッチですが、実装されている MULTI 用のモニタでは、以下の通り使用しています。正しく設定してご使用ください。

SW1 番号	1	2	ポート
設定	ON	ON	設定禁止
	OFF	ON	38400 baud
	ON	OFF	19200 baud
	OFF	OFF	9600 baud

ポートの設定

SW1 番号	3	4	プロファイラ周期
設定	ON	ON	タイムを使用しない
	OFF	ON	200 Hz 5 ms
	ON	OFF	100 Hz 10 ms
	OFF	OFF	60 Hz 16.67ms

プロファイラ周期の設定

SW1 番号	5	デバッグモード
設定	ON	7segLED をモニタが使用

デバッグモードの設定

SW1 番号	6	ブレーク割り込み
設定	ON	INTP3 を使用

ブレーク割り込みの設定

SW1-7,SW1-8 は、MULTI では使用していません。

13.1.3. ボードの接続

「6 ホスト PC との接続」を参照して、シリアルまたは、PCI バスのどちらかで、PC と接続してください。

13.2. Multi モニタ

13.2.1. モニタ・ワーク RAM

モニタでは、SRAM の最上位の 64KB をワーク用の RAM として使用しています。したがって、FE07-0000H ~ FE07-FFFFH はユーザ・プログラムでは使用できません。

13.2.2. 割り込み

ユーザプログラムで割り込みを使用する場合には、「12 マスカブル割り込みを使用したアプリケーションの開発」を参照してください。

13.2.3. 強制ブレーク用の割り込み

モニタが通信、強制ブレーク (MULTI デバッガでの Halt ボタン) 及び、タイマ (プロファイラ等) で使用する割り込みは、NMI と INTP03 が選択できます。選択基準としては、DMA を使用するプログラムを実行する場合 (音声入出力等) には INTP3 を使用し、そうでない場合は NMI を使用することをお薦めします。以下にそれぞれの場合の相違点を示します。

NMI を使用する場合

ユーザプログラム実行中いつでもブレークできますが、DMA が動作中にブレークした場合は、DMA も中断してしまいます (再実行時にも自動的に再開されることはありません)

INTP03 を使用した場合

ユーザプログラムで INTP03 より高い優先順位の割り込みを使用している場合や、割り込みが許可されていない状態ではブレークできません。但し、DMA はブレーク中も継続して実行されます。

13.2.4. _INIT_SP の設定

モニタで _INIT_SP (スタック・ポインタの初期値) は、FE06-FFF0H (SRAM の最上位) に設定されています (MULTI デバッガの環境で _INIT_SP を変更することもできます)

13.2.5. リモート接続

MULTI デバッガでモニタとの接続は、シリアル接続と PCI バス接続が選択できますが、一度接続した方から他方に切り替える場合には、モニタをリセット (リアパネルのリセット・スイッチを押す) してから RTE for Win32 のユーティリティ Check RTE32 を起動して変更してください。

13.2.6. モニタの実行領域

搭載している ROM には、128K バイトの前半にキャッシュ領域で実行するコードが、後半にアンキャッシュ領域で実行するコードの二つが入っています。

プログラムのダウンロードを高速に行いたい場合は、キャッシュ側を選択してください。

但し、モニタの介入 (プロファイルタイマなど) によって、ユーザプログラムで実行中に使用していたキャッシュの内容が一時的にフラッシュされてしましますので、プロファイル測定中においては獲得データの誤差が大きくなる場合があります。このような場合は、アンキャッシュ側を指定してください。

出荷時の設定では、キャッシュ (SW2-8:ON) となっています。

13.2.7. 特殊命令

モニタでは、以下の命令を、シングルステップ、ブレークポイント及びシスコール機能で使用しています。

BRKPNT 命令 (0x6cxx)

ユーザプログラム内では、ブレーク命令と解釈されるコードは使用しないでください。

13.3. RTE コマンド

サーバと接続すると TARGET ウィンドウが開かれ、ここで RTE コマンドを発行することができます。表に RTE コマンドの一覧を示します。

コマンド名	内容
HELP, ?	ヘルプ表示
INIT	イニシャライズ
VER	バージョン表示
INB, INH, INW	I/O リード
OUTB, OUTH, OUTW	I/O ライト
DCTR,ICTR,PLLCR, CMCR	内部レジスタの変更、表示
SFR	内部 I/O 表示 / 設定

RTE コマンド一覧

各コマンドには、パラメータを必要とするものがあります。アドレスやデータなど、数値のパラメータは、全て 16 進数とみなされます。以下の数値指定は誤りです。

0x1234 1234H \$1234

13.3.1. HELP(?)

<書式> HELP [コマンド名]

HELP は、RTE コマンドの一覧や書式を表示します。また、"HELP"と入力するかわりに"?"としても同様です。コマンド名を省略すると、使用できるコマンド一覧を表示します。

<例> HELP SFR

SFR コマンドのヘルプを表示します。

13.3.2. INIT

<書式> INIT

INIT は、RTE 環境の初期化を行ないます。通常、このコマンドを使用しないでください。

13.3.3. VER

<書式> VER

VER は、RTE 環境のバージョンを表示します。

13.3.4. INB,INH,INW

<書式> INB [アドレス]

INH [アドレス]

INW [アドレス]

INB,INH,INW は、I/O リードを行ないます。INB はバイト、INH はハーフ・ワード、INW はワード単位でアクセスします。アドレスを省略すると、前回のアドレスが指定されたものとみなします。

<例> INB 1000

1000H からバイトで I/O リードします。

13.3.5. OUTB,OUTH,OUTW

<書式> OUTB [[アドレス] データ]
 OUTH [[アドレス] データ]
 OUTW [[アドレス] データ]

OUTB,OUTH,OUTW は、I/O ライトを行ないます。OUTB はバイト、OUTH はハーフ・ワード、OUTW はワード単位でアクセスします。アドレスとデータを省略すると、前回の指定と同じものとみなされます。

<例> OUTH 2000 55AA
 2000H 番地にデータ 55AAH をハーフ・ワードで I/O ライトします。

13.3.6. DCTR コマンド

<書式> DCTR [ALL]

DCTR レジスタを表示します。レジスタは 256 個ありますが、バリッドビットが有効になっているレジスタのみ表示します。ただし、all を指定した場合は、全てのレジスタ値を表示します。DCTR レジスタは、I/O 空間 f2000000h-f2000ffffh にマップされています。

13.3.7. ICTR コマンド

<書式> ICTR [ALL]

ICTR レジスタを表示します。レジスタは 128 個ありますが、バリッドビットが有効になっているレジスタのみ表示します。ただし、all を指定した場合は、全てのレジスタ値を表示します。ICTR レジスタは、I/O 空間 fa000000h-fa000ffffh にマップされています。

13.3.8. PLLCR コマンド

<書式> PLLCR

PLLCR の値を表示します。

13.3.9. CMCR コマンド

<書式> CMCR [=]VALUE

CMCR (キャッシュメモリ・コントロールレジスタ) に値を設定します。

13.3.10. SFR コマンド

<書式> SFR [レジスタ名 [=データ]]

レジスタ名を指定してデータを省略した場合は、そのレジスタからリードしたデータを表示します。レジスタ名と“の後にデータを指定した場合には、そのレジスタにデータをライトします。データのサイズは、指定したレジスタの有効サイズで自動的に決定されます。内部 I/O レジスタの詳細については、V832-CPU のマニュアルを参照してください。

<例 1> SFR
 レジスター一覧を表示します。
 <例 2> SFR IMR
 レジスタ IMR の内容を表示します。
 <例 3> SFR IMR=55AA
 レジスタ IMR にデータ 55AAH をライトします。

14.APPENDIX.B PARTNER モニタ

PARTNER 用のモニタ ROM を使用して、ホストの PARTNER と接続して使用する場合の設置方法と使用上の注意事項について説明します。

14.1. ボードの設置

14.1.1. SW1 の設定

SW1 は、汎用の入力ポートのスイッチですが、実装されている PARTNER 用のモニタでは、以下の通り使用しています。正しく設定してご使用ください。

SW1 番号	1	2	ポーレート
設定	ON	ON	115200 Baud
	OFF	ON	38400 baud
	ON	OFF	19200 baud
	OFF	OFF	9600 baud

ポーレートの設定

SW1 番号	3	4	タイマ
設定	ON	ON	常時この状態でご使用ください。

SW1 番号	5	デバッグモード
設定	ON	7segLED をモニタが使用
	OFF	通常の使用状態

デバッグモードの設定

SW1 番号	6	ブレーク割り込み
設定	ON	INTP3 を使用
	OFF	NMI を使用

ブレーク割り込みの設定

SW1-7,SW1-8 は、使用していません。

14.2. PARTNER モニタ

14.2.1. モニタ・ワーク RAM

モニタでは、SRAM の最上位の 64KB をワーク用の RAM として使用しています。したがって、FE07-0000H ~ FE07-FFFFH はユーザ・プログラムでは使用できません。

14.2.2. 割り込み

ユーザプログラムで割り込みを使用する場合には、「12 マスカブル割り込みを使用したアプリケーションの開発」を参照してください。

14.2.3. 強制ブレーク用の割り込み

モニタの通信及び、強制ブレーク（ESC ボタン）で使用する割り込みは、NMI と INTP03 が選択できます。選択基準としては、DMA を使用するプログラムを実行する場合（音声入出力等）には INTP03 を使用し、そうでない場合は NMI を使用することをお薦めします。以下にそれぞれの場合の相違点を示します。

NMI を使用する場合

ユーザプログラム実行中いつでもブレークできますが、DMA が動作中にブレークした場合は、DMA も中断してしまいます（再実行時にも自動的に再開されることはありません）。

INTP03 を使用した場合

ユーザプログラムで INTP03 より高い優先順位の割り込みを使用している場合や、割り込みが許可されていない状態ではブレークできません。但し、DMA はブレーク中も継続して実行されます。

14.2.4. SP の設定

モニタのスタック・ポインタの初期値は、FE06-FFF0H（SRAM の最上位）に設定されています。また、モニタでは、ユーザ・プログラムで設定したスタック領域を 32 バイト使用します。

14.2.5. リモート接続

デバッガとの接続は、シリアル接続と PCI バス接続が選択できますが、一度接続した方から他方に切り替える場合には、モニタをリセット（リアパネルのリセット・スイッチを押す）した後、RPTSETUP.exe を起動して通信路の変更を実施してください。

14.2.6. モニタの実行領域

搭載している ROM は、128K バイトの前半にキャッシュ領域で実行するコードが、後半にアンキヤッシュ領域で実行するコードの二つが入ってます。

キャッシュ領域でモニタを動作させた場合の方が、モニタの実行が速いためプログラムのダウンロードも速くなります。通常は、キャッシュ側の設定（出荷時の状態）でご使用ください。出荷時の設定では、キャッシュ（SW2-8:ON）となっています。

14.2.7. 特殊命令

モニタでは、以下の命令を、シングルステップ、ブレークポイント及びシスコール機能で使用しています。

BRKPNT 命令 (0x6cxx)

ユーザプログラム内では、ブレーク命令と解釈されるコードは使用しないでください。

- Memo -

RTE-V832-PC ユーザーズ・マニュアル

M813MNL01

Midas lab