

マイクロコンピュータ技術情報

技術通知 32ビットマイクロコントローラ V850E/MS2 使用制限事項の件		発行番号	ZBG-CC-07-0006号	1/2	
		発行日	2007年 3月 1日		
		発行部門	NECエレクトロニクス株式会社 第四システム事業本部 汎用マイコンシステム事業部 第二ソリューショングループ		
文書分類	○	使用制限事項	バージョン・アップ	ドキュメント誤記訂正 (正誤表)	その他
関連資料	V850E/MS2 ユーザーズ・マニュアル (ハードウェア編)				資料番号: U14985JJ
	V850E/MS1, MS2 ユーザーズ・マニュアル (アーキテクチャ編)				資料番号: U12197JJ
	μPD703130 データ・シート				資料番号: U15390JJ

CP(K),O

1. 対象製品

V850E/MS2

- ・ μPD703130 : ROMレス品

2. 新たな制限事項

今回新たに制限事項 No.8 を追加しました。

- ・ No.8 callt 命令に関する制限事項

3. 回避策

今回追加した制限事項の回避策です。詳細は別紙1を参照してください。

- ・ No.8 callt 命令を使用する場合、ベース・アドレス (CTBP レジスタ値) を内蔵メモリ領域に設定する、もしくは callt 命令を使用しないでください。

4. 改善計画

大変申し訳ございませんが、デバイスの修正は行わず、使用上の制限事項とさせていただきます。

5. 制限事項一覧

制限事項の履歴と、その詳細情報が含まれました制限事項一覧を、別紙1に記載します。

6. 発行文書履歴

V850E/MS2 使用制限事項一覧 発行文書履歴

文書番号	発行日	記事
SBG-DT-0034	2001.11.12	No.1, No.2
SBG-DT-0079	2002. 4.16	No.1(訂正), No.3
SBG-DT-0118	2002. 7.29	No.4, No.5
SBG-DT-03-0112	2003. 3.25	No.6
ZBG-CC-04-0005	2004. 5.27	No.7
ZBG-CC-07-0006 (最新版)	2007. 3. 1	No.8

以上

V 8 5 0 E / M S 2 の使用制限事項一覧

1) 製品バージョン

- ・ μ P D 7 0 3 1 3 0 : K 規格
: 規格区分は、ロット番号の左から 5 桁目のアルファベットです。

2) 製品履歴

- ・ μ P D 7 0 3 1 3 0

No.	不具合事項	規格
		K
1	s l d 命令連続実行時の制限事項	
2	s s t 命令直後の b c o n d 命令実行時の制限事項	
3	A / D コンバータに関する不具合	
4	N M I による D M A 強制中断に関する不具合	
5	内蔵 R A M でのプログラム実行と D M A 転送に関する不具合	
6	内蔵 R A M を対象とした D M A 転送と、内蔵 R A M での命令実行の競合に関する不具合	
7	E D O D R A M とリフレッシュ競合による D M A A K 不正出力の制限事項	
8	ca l l t 命令に関する制限事項	

: 該当しない、 : 今後とも制限事項、 x : 該当する

3) 使用制限事項の詳細

No.1 sld 命令連続実行時の制限事項

【内容】

外部メモリ空間からリードを行う sld 命令に対して割り込み処理 (NMI を含む) が発生すると、ロードされるべきデータがレジスタに転送されない場合があります。

(異常動作内容詳細)

高速化のために設けられた、sld 専用のアドレス計算資源が空いている場合、sld 命令は ID ステージでアドレス計算を済ませ、EX ステージを省略して ID ステージ後直ちに MEM ステージとなりバスサイクルを発行します。sld 命令の対象が外部メモリの場合、ID ステージ(a)で割り込み (NMI を含む) を受けた場合にバスサイクル発行をキャンセルできないため、MEM ステージ (リード・サイクル) を空読みサイクルとして発行した後に WB ステージを実行しない事により命令処理をキャンセルする設計としています。

しかし、命令(2)の処理をキャンセルする際に命令(1)の WB ステージが遅れた場合(b)、命令(1)の MEM ステージのラッチデータを命令(2)の空読みした MEM ステージのデータで上書きしてしまうことがあり、不具合が発生します。

外部バスサイクル

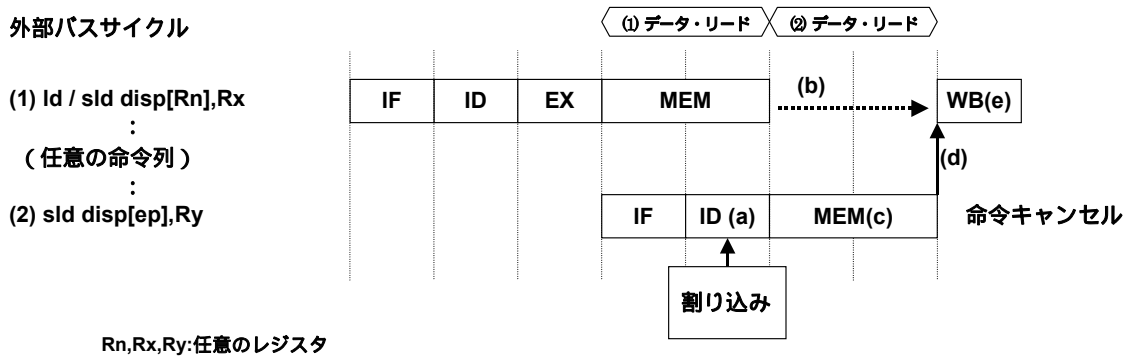


図 3. sld 不具合の発生メカニズム

sld 不具合の発生メカニズムは次の通りです。

- (a) 命令(2)の ID ステージ(a)で割り込み (NMI を含む) を受けます。
- (b) 先行する命令(1)が MEM ステージの後、WB 資源へのアクセス待ち合わせなどにより WB ステージが遅れます(b)。
- (c) sld 命令(2)の対象が外部メモリの場合、バスサイクル発行をキャンセル出来ないため、命令(2)の MEM ステージ(c)を空読みサイクルとして発行します。
- (d) 命令(2)の WB ステージを実行しないことにより命令処理をキャンセルする際、命令(1)の MEM ステージでラッチされたデータに、命令(2)で空読みしたデータを上書き(d)してしまいます。
- (e) 命令(2)で上書きされたラッチデータを命令(1)の WB ステージ(e)で使用する事により、Rx レジスタに誤書き込みが発生します。

【非該当条件】

本不具合は、以下のどちらかの場合には非該当です。

- (1) sld 命令のロード対象が内蔵メモリ (内蔵 RAM を含む) の場合
- (2) sld 命令の前後で割り込みが禁止されている、かつ NMI を使用していない場合

【回避策】

不具合回避策について以下に示します。

< アセンブラについて >

s l d 命令を l d 命令に全て変更する。

< NEC コンパイラについて >

t i d a t a セクションへの割り当てを指定しているデータをセクションへ割り当てない。または、t i d a t a セクションから s i d a t a セクション等へ割り当てを変更する。(これらにより、s l d 命令を使用しないコードを生成)

< GHS コンパイラについて >

以下の2つの対策を行うことにより、s l d 命令の連続実行を行わないため、本不具合を回避することができます。

(1) コンパイル時に「- Z 1 4 1 2」オプションを指定する。

(「- O S」オプションを使う場合は、「- Z 1 4 1 2」と「- i n l i n e _ p r o l o g u e」オプションを使用してください。)

(2) T D A (タイニ・データ領域) 機能の pragma を使用しない。

(T D A 領域を使用している場合は、T D A 領域の定義を無効にする「- n o t d a」オプションをコンパイル時に指定するか、T D A 領域の定義をソースコード上から全て削除してください。)

< O S (RX850, RX850PRO) について >

以下のどちらか一つに該当している場合には非該当です。

- ・スタック領域を内蔵 R A M 領域のみに設定している。
- ・N M I 割り込みを使用していない。

上記2つのどちらにも当てはまらない場合には、弊社営業までご相談ください。

< ミドルウェア (MH, MR, MMR, JBIG, JPEG、US ファイル、CF ドライバ) について >

本不具合は該当しません。(非該当)

No.2 sst命令直後のbcond命令実行時の制限事項

【内容】

以下のようにsst/st命令()のアクセス対象が外部メモリであり、なおかつ以降にsst命令()とbcond命令()が連続した場合、以下のいずれかの現象が発生します。

- ・ bcond命令で分岐した直後の命令コード1ワード(32ビット)が実行されず(スキップされる)かつプログラム・カウンタがずれる。
- ・ bcond命令で分岐した直後の命令コード48ビットのうち、いずれかの16ビットが0に化ける。

このため、以降のプログラムが正常に動作しない可能性があります。以下に本異常動作例を示します。なお、本異常動作は、内蔵メモリ(RAMを含む)からの命令フェッチ、及び外部メモリからの命令フェッチのいずれにおいても発生します。

sst / st 命令 (外部メモリに対するアクセス)
 : sst / st 命令以外の任意の命令列 (0個以上)
 sst 命令
 bcond (bc, be, bge, bgt, bh, bl, ble, blt, bn, bnc, bne, bnh, bnl, bnv, bnz, bp, br, bsa, bv, bz)
 命令

・ 分岐先命令の異常動作例

(1) 命令コード48ビットのうち16ビットが0に化ける場合

正常な場合		異常動作の場合	
PC値	実行される命令フロー	PC値	実行される命令フロー
0000002A	sst.b r0, 0x0[ep]	0000002A	sst.b r0, 0x0[ep]
0000002C	br 0x40	0000002C	br 0x40
00000040	movhi <u>0x1010</u> , r0, r6	00000040	movhi <u>0x0</u> , r0, r6
命令コード (40361010)		命令コード (40360000)	
00000044	movea 0x1010, r0, r6	00000044	movea 0x1010, r0, r6

(2) 命令コード1ワード(32ビット)が実行されない(スキップされる)かつプログラム・カウンタがずれる場合

正常な場合		異常動作の場合	
PC値	実行される命令フロー	PC値	実行される命令フロー
0000002A	sst.b r0, 0x0[ep]	0000002A	sst.b r0, 0x0[ep]
0000002C	br 0x4e	0000002C	br 0x4e
0000004E	movhi 0x1010, r0, r6	(movhi 0x1010, r0, r6命令が実行されない)	
00000052	movea 0x1010, r0, r6	0000004E	movea 0x1010, r0, r6
			(PC値がずれる)

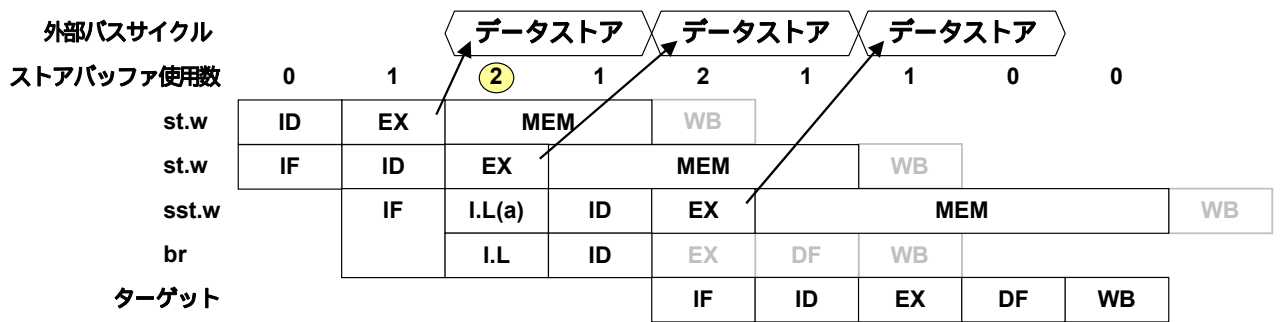
本異常動作は、下記の様なパイプラインの状況において引き起こされます。

V850E/MS1のCPUは、メモリへのライトデータを一時的に蓄える2段のストアバッファを持っており(図5)外部バス・アクセスを行っている状態でさらにsst/st命令を実行しても2段のストアバッファに空きがある限りバスサイクルの完了を待たずに後続の命令を続けて実行します。このバッファに空きが無くなると、sst/st命令はIDステージでバッファが空くまで待ちます(図4-1(a))。

sst命令とbcond命令は同時発行されるため、ストアバッファを待っているsst命令の直後にbcond命令があればこのbcond命令も一緒に待ち合わせを行わなければなりません。特定の条件下()でbcond命令が待ち合わせをせずにIDステージに移行してしまいます(図4-2(b))。この誤動作により分岐先の命令(図のターゲットの命令)フェッチ動作が異常となり、正常に動作しなくなります。

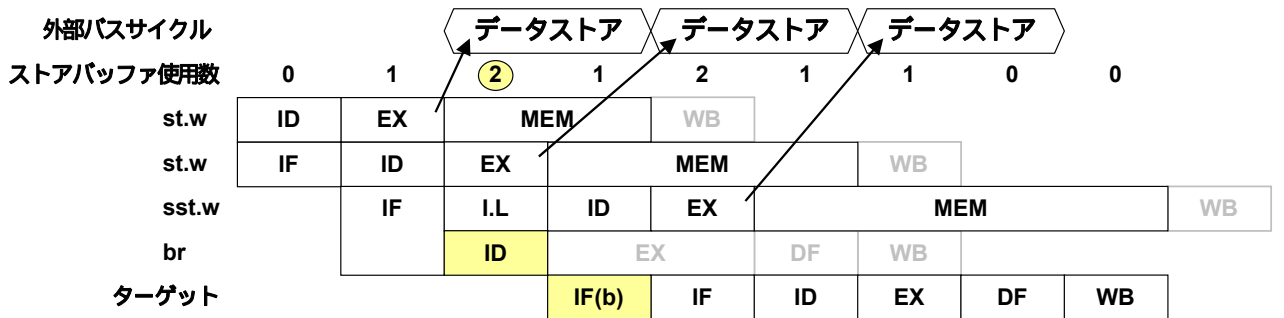
() sst命令とbcond命令の連続でも命令フェッチのバスサイクルとデータ・アクセスのバスサイクルの発行順序により不具合が発生しない場合もあります。

図4-1 本来の動作例

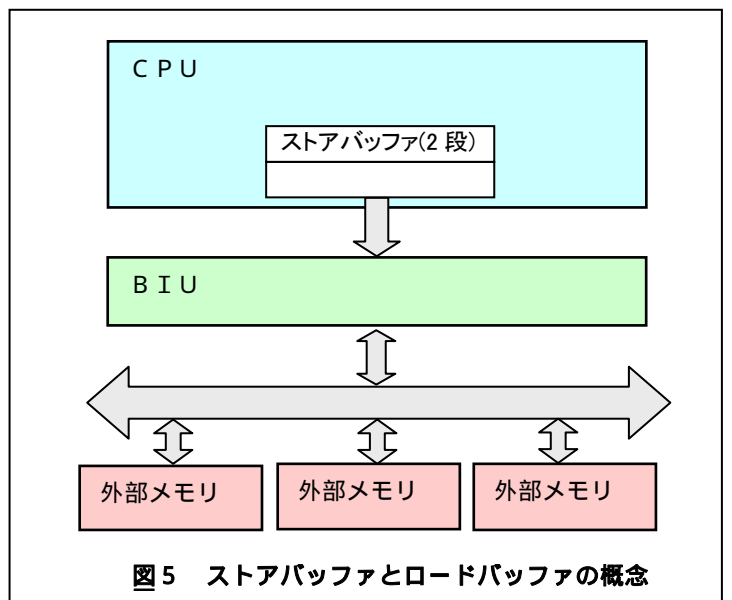


② : 誤動作の原因となったストアバッファ Full 状態

図4-2 不具合発生時



- IF : 命令フェッチ
- ID : 命令デコード
- EX : デコードした命令の実行
- MEM : 実行対象メモリへのアクセス
- WB : 実行した結果のレジスタ書き込み
- I.L : ハザードの発生により自動的に挿入されるアイドル・ステート。



【回避策】

ソフトウェアで、以下のいずれかの方法により回避可能です。

- ・ bcond 命令の直前の sst 命令を st 命令に置き換える。
- ・ bcond 命令と直前の sst 命令の間に、nop 命令を挿入する。

< NEC コンパイラについて >

2001年11月中旬に対応版をリリース予定(V2.41)です。対応版では回避オプションの指定により、本不具合を回避することができます。

コンパイル時に以下のオプションを指定

- (1) ca850の回避オプション
-Wa, -p
- (2) as850の回避オプション
-p

本日程については予告無く変更する場合がございますので、対応版のリリース日程については、別途、弊社営業までお問い合わせください。

< GHS コンパイラについて >

以下の2つの対策を行うことにより、sst 命令が出力されないため、本不具合を回避することができます。

- (1) コンパイル時に「-Z1412」オプションを指定する。
(「-OS」オプションを使う場合は、「-Z1412」と「-inline_prologue」オプションを使用してください。)
- (2) TDA (タイニ・データ領域) 機能の pragma を使用しない。
(TDA 領域を使用している場合は、TDA 領域の定義を無効にする「-notda」オプションをコンパイル時に指定するか、TDA 領域の定義をソースコード上から全て削除してください。)

GHS コンパイラでの回避方法は「No1.sld 命令連続実行時の制限事項」の不具合回避方法と同一です。

< OS (RX850, RX850PRO) について >

本不具合は該当しません。(非該当)

< ミドルウェア (MH, MR, MMR, JBIG, JPEG, US ファイル, CF ドライバ) について >

本不具合は該当しません。(非該当)

No.3 A / Dコンバータに関する不具合

【内容】

本不具合は、A / Dコンバータのタイマ1トリガ・モード、および外部トリガ・モード時に発生します（A / Dトリガ・モード、またはタイマ4トリガ・モードを使用されている場合は非該当です。）

タイマ1トリガ・モードによるA / Dコンバータの起動要因は、INTCC110のみ、また外部トリガ・モード時の起動要因は、ADTRG端子入力のみであり、以下に示す割り込み要因は無視されるべきですが、A / D変換終了の直前（図6 部分[内部システムクロック 2クロック分]）に、起動要因でない以下の割り込み要因が発生すると、A / D変換の起動要因と誤認し、A / D変換終了割り込み（INTAD）後、再度A / D変換動作を開始します。最初のA / D変換は正常に終了し、A / D変換結果はADCRnレジスタに格納されます（再変換中は読み出し可能です。）

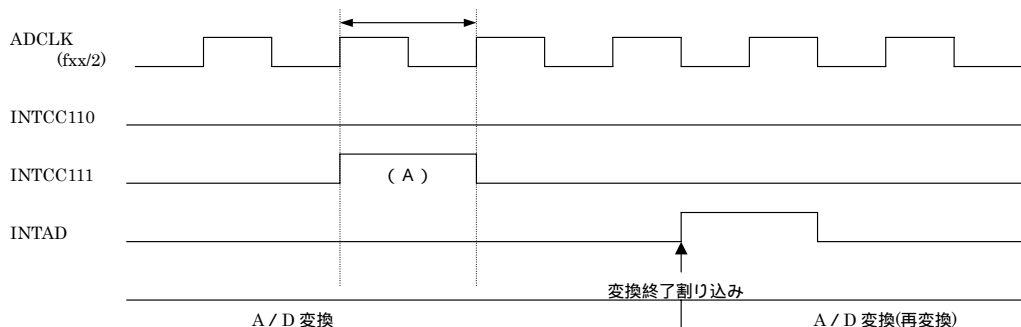
再起動したA / D変換は、正常に変換動作を行い、A / D変換終了割り込み（INTAD）を発生し終了します。A / D変換結果はADCRnレジスタに上書きされます。

【A / Dの変換を引き起こす原因となり得る割り込み要因】

タイマー一致割り込み	INTCC111
	INTCC112
	INTCC113
外部端子割り込み	INTP111
	INTP112
	INTP113

タイマ1（チャンネル1）の外部キャプチャ・トリガ入力と兼用になっている外部割り込み信号も同様に、再変換動作の要因となります。外部割り込み入力の場合は、ノイズ除去（デジタル・ノイズ除去（ $2 \times f_{xx} \sim 3 \times f_{xx} \text{ ns}$ ））回路により、実際には、下図（A）のタイミングより前に、有効エッジが入力された場合に、本現象が発生致します。

図6 . 不具合発生タイミング例



f_{xx} : 内部システムクロック

【非該当条件】

A / Dトリガ・モード、またはタイマ4トリガ・モードを使用されている場合は非該当です。タイマ1トリガ・モード、または外部トリガ・モード時の非該当条件を以下に示します。

タイマ1（チャンネル1）のコンペアー一致割り込み（INTCC111 / 112 / 113）がA / D変換中に発生せず、かつ外部割り込み信号（INTP111 / 112 / 113）がA / D変換中に入力されなければ、不具合は発生致しません。

【回避策】

不具合に該当し再変換が発生した場合でも、変換結果は正常な値を示しますので、最新の変換値を取得するような使用方法においては、影響は少ないと思われます。ただし、再変換動作が不都合となる場合には、A / Dトリガ・モードを使用し、タイマー一致割り込み、または外部端子割り込みの割り込み処理ルーチン内でADM0レジスタのCEビットを1に設定することによりA / D変換を開始してください。

No.4 NMIによるDMA強制中断に関する不具合

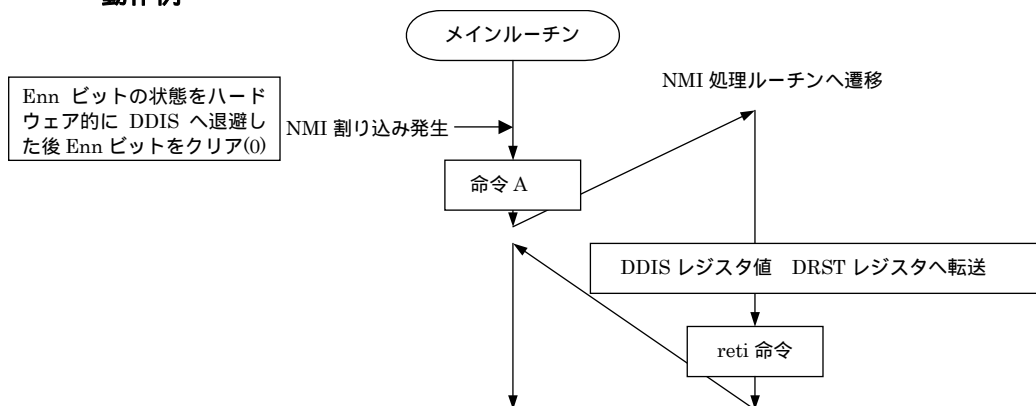
【内容】

上記製品はDMA転送中のNMI入力により、DMA転送を強制的に中断します。このときDMAコントローラは、すべてのチャンネルのEnnビット(DCHCnレジスタのビット0)の状態をDDISレジスタに退避した後、Ennビットをクリア(0)して、DMA転送禁止状態にします。また、CPUのパイプライン動作として、NMI割り込みを受け付ける際、既にフェッチが完了している命令(最大1命令)まで実行されます。

この実行された命令が、Ennビットを操作するような命令であった場合、NMI処理ルーチン内では、中断されているDMA転送を再開させるために、DDISレジスタの内容をDRSTレジスタに転送するため、NMI入力直後に実行されたEnnビットの操作が反映されません。

このため、DMA転送を禁止状態、もしくは許可状態にしたにも拘わらず、NMIで中断されたDMA転送を期待通りに状態復帰させることができません。

動作例



上記命令Aで、DCHCnレジスタのEnnビットを操作し、DMAを禁止/許可にした場合、上記とで、各DMAチャンネルの禁止/許可状態に不整合が生じるため、正常に中断されたDMA転送を正常に状態復帰できません。

また、DCHCnレジスタのソフトウェアトリガ(STGn)ビットをセット(1)している状態において、命令AでEnnビットがセット(1)された場合、NMI処理ルーチン内でもDMA転送が行われます。

【非該当条件】

NMIを使用していなければ、問題ありません。

また、NMIにより強制中断されたDMA転送を継続再開しないシステムであれば、問題となることはありません。

【回避策】

NMIにより強制中断されたDMA転送は、再度初期化して実行してください。

No.5 内蔵RAMでのプログラム実行とDMA転送に関する不具合

【内容】

内蔵RAMを対象としたDMA転送を実行しており、かつ内蔵RAM上に配置されたビット操作命令(SET1,CLR1,NOT1)もしくは、ミス・アライン・アドレスに対するデータ・アクセス命令を実行した場合、内部バスの競合動作により、CPUがデッド・ロックする可能性があります。デッド・ロック中はリセットのみ受付可能です(NMIも割り込みも受け付けられません)。

【非該当条件】

内蔵RAM上での命令実行を行っていない、もしくは、内蔵RAMを対象としたDMA転送を行っていない場合は、非該当です。

【回避策】

以下のいずれかの方法により、回避をお願い致します。

- 内蔵RAM上に配置された命令を実行する場合は、内蔵RAMを対象としたDMA転送を行わない。
- 内蔵RAMを対象とするDMAを実行する場合は、内蔵RAM上に配置された命令実行を行わない。

No.6 内蔵RAMを対象としたDMA転送と、内蔵RAMでの命令実行の競合に関する不具合

【内容】

内蔵RAM上のプログラム実行において、分岐命令または割り込みによる内蔵RAMから外部メモリへの分岐と、内蔵RAMに対するDMA転送が競合すると、外部メモリ上にある1～2命令を間違っフェッチし、誤動作が発生することがあります。

(1) 以下の2つの条件が共に該当する場合、分岐先の外部メモリ上にあるプログラムの先頭1～2命令を間違っフェッチする場合があります。

条件1：内蔵RAMでプログラムを実行しており、分岐命令（Bcond¹, JR, JARL, JMP, RETI）による外部メモリへの分岐が発生。

条件2：内蔵RAMを対象²とするDMA転送を実行。

(2) 以下の3つの条件が共に該当する場合、外部メモリ上にある割り込みハンドラの先頭1～2命令を間違っフェッチする場合があります。

条件1：ROMレス・モード、またはシングルチップ・モード1を使用している。

条件2：内蔵RAM上でプログラムを実行しており、割り込み（NMIを含む）による割り込みハンドラ（外部メモリ）への分岐が発生。

条件3：内蔵RAMを対象²とするDMA転送を実行。

1：Bcond命令

[BGT, BGE, BLT, BLE, BH, BNL, BL, BNH, BE, BNE, BV, BNV, BN, BP, BC, BNC, BZ, BNZ, BR, BSA]

2：内蔵RAMが転送先の場合、または内蔵RAMが転送元の場合

【非該当条件】

以下のいずれかの一つでも当てはまる場合は、非該当です。

(1) 内蔵RAMでプログラムを実行しない。

(2) 内蔵RAMを対象²としたDMA転送を行っていない。

(3) 内蔵RAMからの分岐先が内蔵メモリ³であり、かつシングルチップ・モード0を使用している。

2：内蔵RAMが転送先の場合、または内蔵RAMが転送元の場合

3：内蔵ROMまたは内蔵RAM

【回避策】

以下のいずれかの方法により回避することができます。

・ソフトウェアによる回避策

内蔵RAM上のプログラム実行中は割り込みを禁止した上で、以下の回避策を行ってください。

また、NMIを使用している場合は、の回避策を実施願います。

内蔵RAMから外部メモリへの分岐命令の直前に、内蔵RAMからのダミーリード+NOP命令を実行し、かつ分岐はワードアラインされた「JMP」命令を使用してください。例を示します。

(A) JMP命令で分岐する場合の例

不具合該当例	回避方法
内蔵RAM上のプログラム 任意の命令	内蔵RAM上のプログラム 任意の命令 .align 4 (*1) nop ld.b 0xFFFFE000[R0],R0 nop
jmp [Rx]	jmp [Rx] (*2)
外部メモリ上のプログラム 任意の命令	外部メモリ上のプログラム 任意の命令

*1) NOP命令以降の追加命令を32ビットアラインで配置する

*2) レジスタRxは、不具合該当例のレジスタRxと同じ

(B) RETI 命令で分岐する場合の例

不具合該当例	回避方法
内蔵 R A M 上のプログラム 任意の命令 reti	内蔵 R A M 上のプログラム 任意の命令 .align 4 (*1) nop ld.b 0xFFFFE000[R0],R0 nop reti
外部メモリ上のプログラム 任意の命令	外部メモリ上のプログラム 任意の命令

*1) NOP 命令以降の追加命令を 32 ビットアラインで配置する

(C) Bcond, JR 命令で分岐する場合の例

不具合該当例	回避方法
内蔵 R A M 上のプログラム 任意の命令 br/jr EXMEM	内蔵 R A M 上のプログラム 任意の命令 .align 4 (*1) movhi hi(EXMEM),R0,Ry (*3) movea lo(EXMEM),Ry,Ry (*3) nop ld.b 0xFFFFE000[R0],R0 nop jmp [Ry] (*3)
外部メモリ上のプログラム EXMEM: 任意の命令	外部メモリ上のプログラム EXMEM: 任意の命令

*1) MOVHI 命令以降の追加命令を 32 ビットアラインで配置する

*3) レジスタ Ry は未使用の任意のレジスタ

(D) JARL 命令で分岐する場合の例

不具合該当例	回避方法
内蔵 R A M 上のプログラム 任意の命令 jarl EXMEM,Rz	内蔵 R A M 上のプログラム 任意の命令 .align 4 (*1) movhi hi(EXMEM),R0,Rw (*4) movea lo(EXMEM),Rw,Rw (*4) jarl DUMMYLBL,Rz (*5)(*6) DUMMYLBL: add 0x0a,Rz (*7) ld.b 0xFFFFE000[R0],R0 nop jmp [Rw]
外部メモリ上のプログラム EXMEM: 任意の命令	外部メモリ上のプログラム EXMEM: 任意の命令

*1) MOVHI 命令以降の追加命令を 32 ビットアラインで配置する

*4) レジスタ Rw は、未使用の任意のレジスタ

*5) レジスタ Rz は、不具合該当例のレジスタ Rz と同じ

*6) JARL 命令のディスプレースメントは次の add 命令

*7) JMP 命令の次の命令に復帰するためのディスプレースメント加算

- ・ 不具合回避策 I が適用できない場合
以下のいずれかの方法により回避してください。
- (1) 内蔵 R A M を対象とした D M A 転送を行う場合は、内蔵 R A M でのプログラム実行を行わない。
- (2) 内蔵 R A M でプログラム実行を行う場合は、内蔵 R A M を対象とした D M A 転送を行わない。

No.7 EDO DRAM とリフレッシュ競合による DMAAK 不正出力の制限事項

【内 容】

EDO DRAM に対するアクセスがオンページ・アクセスの「 T B 」状態で始まる 2 サイクル DMA 転送において、EDO DRAM サイクルの「 T B 」状態と、リフレッシュ要求が競合したとき、DMAAKm 信号出力が以下のいずれかの状態になります。(DMA 転送は正常に実施されます)

- (1) アイドル・ステート挿入しない場合 (BCC.BCn1 ビット, BCn0 ビット = 00)
リード・サイクル終了と同時に DMAAKm 信号出力がインアクティブになる。
- (2) アイドル・ステート挿入する場合 (BCC.BCn1 ビット, BCn0 ビット = 01)
リード・サイクルとライト・サイクル間のアイドル・ステートで DMAAKm 信号出力がインアクティブになり、ライト・サイクルで再びアクティブになる。
備考 : m=0-3, n=0-7

【該当条件】

以下の 5 つの条件をすべて満たす場合、本不具合に該当します。

- 条件 1 : 2 サイクル DMA 転送 (外部 外部) である。
- 条件 2 : 転送元のメモリタイプが EDO DRAM である。
- 条件 3 : EDO DRAM に対してデータ・ウエイトを挿入していない。
(DRCp.DAC1p ビット, DAC0p ビット = 00)
- 条件 4 : EDO DRAM に対して CAS プリチャージ・ウエイトを挿入していない。
(DRCp.CPC1p ビット, CPC0p ビット = 00)
- 条件 5 : リフレッシュ機能を有効にしている。
備考 : p=0-3

【回避策】

以下のいずれかの方法により、EDO DRAM アクセス・サイクルが「 T B 」状態から始まらないようにしてください。

- (1) EDO DRAM に対してデータ・ウエイトを挿入する。
(DRCp.DAC1p ビット, DAC0p ビット = 01, 10, 11)
- (2) EDO DRAM に対して CAS プリチャージ・ウエイトを挿入する。
(DRCp.CPC1p ビット, CPC0p ビット = 01, 10, 11)
備考 : p=0-3

No.8 callt 命令に関する制限事項

【内 容】

以下のような命令シーケンスにおいて、callt 命令のベース・アドレス (C T B P レジスタ値) が外部メモリ空間に設定されており、かつその call 命令を実行中に、割り込み処理 (N M I を含む) が発生すると、先に実行されているロード命令の結果が、C T B P の値に書き換わることがあります。

該当命令シーケンス

- (1) ld/sld disp[Rn], Rx Rn, Rx: 任意のレジスタ
: (任意の命令列)
- (2) call imm16

【非該当条件】

以下のいずれかの条件を満たした場合は非該当です。

- (1) c a l l t 命令のベース・ポインタが内蔵メモリ (内蔵 R A M を含む) の場合
- (2) c a l l t 命令の前後で割り込みが禁止されている、かつ N M I を使用していない場合
- (3) c a l l t 命令を使用していない場合

【回避策】

< N E C コンパイラについて (CA850 V2.50 以降) >

以下のいずれかの方法により、回避してください。

- ・ c a l l t 命令を出力しないようにする

コンパイラ・オプションで、-Xpro_epi_runtime=off を指定してください。

- ・ c a l l t 命令のベース・アドレスを内蔵RAMに配置する

c a l l t 命令が参照するテーブルは、次のセクションになります。

```
.pro_epi_runtime
```

このセクションを内蔵RAMに配置するには、リンク・ディレクティブ・ファイルで、次の部分を内蔵RAMに配置してください。

```
TEXT      : !LOAD ?RX {
            .pro_epi_runtime = $PROGBITS      ?AX .pro_epi_runtime; <- この部分
            .text            = $PROGBITS      ?AX .text;
        };
```

(例) t e x t が外部メモリに配置されていた場合

(変更前)

```
TEXT      : !LOAD ?RX V0x100000{
            .pro_epi_runtime = $PROGBITS      ?AX .pro_epi_runtime; <- この部分
            .text            = $PROGBITS      ?AX .text;
        };
```

(変更後)

```
PETEXT    : !LOAD ?RX V0x1000{
            .pro_epi_runtime = $PROGBITS      ?AX .pro_epi_runtime;
        };

TEXT      : !LOAD ?RX V0x100000{
            .text            = $PROGBITS      ?AX .text;
        };
```

< O S (R X 8 5 0 , R X 8 5 0 P R O , R X 8 5 0 V 4) について >

c a l l t 命令を使用していないため、本不具合は該当しません。(非該当)

< C A 8 5 0 のライブラリ、g o f a s t について >

c a l l t 命令を使用していないため、本不具合は該当しません。(非該当)

< R X - N E T 8 5 0 (オプション・プロトコルを含む)、R X - F S 8 5 0 (V 2 . 0 0) について >

ライブラリについては、c a l l t 命令を使用していません。

ただし、R X - F S 8 5 0 (C A 8 5 0 版) に添付しておりますサンプル・ドライバ (S G - M A 1 用) は、コンパイラ・オプション [プロローグエピローグランタイム] をデフォルト [使用] にしているため、[未使用] に変更してからリビルドしてください。

< G H S コンパイラについて (V3.x 系 (V3.5 (R6.5.2) 以降)、 V4.x 系 (V4.0.2 (R7.0.1) 以降)) >
以下のいずれかの方法により、回避してください。

- ・ callt 命令を出力しないようにする
コンパイラ・オプションで、-no_callt を指定してください。
- ・ callt 命令のベース・アドレスを内蔵RAMに配置する
callt 命令が参照するテーブルは、次のセクションになります。
 .text
また、c a l l t 命令のテーブルは、libarch.a内のindarchl.oモジュール中に入っています。

このセクションを内蔵RAMに配置するには、リンク・ディレクティブ・ファイルで、次の部分を内蔵RAMに配置してください。

```
MEMORY {  
    <略>  
    iram_memory      : ORIGIN = 0x0, LENGTH = 0x1000  
    <略>  
}  
  
SECTION {  
    <略>  
    .newtext : {libarch.a(indarchl.o(.text))} > iram_memroy  
    <略>  
}
```

上記以外の対応につきましては、お手数をお掛けしますが、弊社販売員にご連絡ください。

以上