

お客様各位

カタログ等資料中の旧社名の扱いについて

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

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

2010年4月1日
ルネサスエレクトロニクス株式会社

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

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

ご注意書き

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

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

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

SH7730 グループ

キャッシュ キャッシュメモリの設定例

要旨

この資料は SH7730 のキャッシュ機能の設定例・動作例を掲載しています。

動作確認デバイス

SH7730

目次

1. はじめに.....	2
2. 応用例の説明.....	3
3. 参考プログラム例.....	16
4. 参考ドキュメント.....	22

1. はじめに

1.1 仕様

- 本参考プログラムでは、キャッシュを命令およびオペランドキャッシュ有効、コピーバックモードに設定します。
- 本参考プログラムでは、一つの参考例としてコピーバックモードで動作した場合、オペランドキャッシュと外部メモリの内容が一致しない場合を示します。

1.2 使用機能

- 命令およびオペランドキャッシュ

1.3 適用条件

評価ボード:	アルファプロジェクト製 SH-4A ボード 型番 AP-SH4A-1A
外付けメモリ (エリア 0):	NOR 型フラッシュメモリ 4 M バイト Spansion 製 S29AL032D70TFI04
(エリア 3):	SDR-SDRAM 32 M バイト (16 M バイト × 2 個) Samsung 製 K4S281632F-UC75
マイコン:	SH7730 (R8A77301)
動作周波数	CPU クロック: 266.66 MHz SuperHyway バスクロック: 133.33 MHz バスクロック: 66.66 MHz 周辺クロック: 33.33 MHz
エリア 0 バス幅:	16 ビット固定 (MD3 端子 = Low レベル)
クロック動作モード:	モード 2 (MD0 端子 = Low レベル, MD1 端子 = High レベル)
エンディアン:	ビッグエンディアン (MD5 端子 = Low レベル)
ツールチェーン:	ルネサス テクノロジ製 SuperH RISC engine Standard Toolchain Ver.9.1.1.0
コンパイルオプション:	High-performance Embedded Workshop でのデフォルト設定 (-cpu=sh4a -debug -optimize=0 -noinline -gbr=auto -macsave=0 -save_cont_reg=0 -chgincpath -errorpath -global_volatile=0 -opt_range=all -infinite_loop=0 -del_vacant_loop=0 -struct_alloc=1 -nologo)

1.4 関連アプリケーションノート

本資料の参考プログラムは、「SH7730 グループ アプリケーションノート SH7730 初期設定例 (RJJ06B0864)」の設定条件で動作確認しています。そちらも合わせてご参照ください。

2. 応用例の説明

パワーオンリセット/マニュアルリセット直後,キャッシュは無効になっています。本資料では,キャッシュの設定方法について説明します。

2.1 SH7730 のキャッシュメモリ概要

表1 SH7730 のキャッシュメモリ概要

項目	命令キャッシュ	オペランドキャッシュ
容量	32 Kバイトキャッシュ	32 Kバイトキャッシュ
方式	4 ウェイセットアソシアティブ, 仮想アドレスインデックス/物理アドレスタグ	4 ウェイセットアソシアティブ, 仮想アドレスインデックス/物理アドレスタグ
ラインサイズ	32 バイト	32 バイト
エントリ数	256 エントリ/ウェイ	256 エントリ/ウェイ
書き込み方式	—	コピーバック/ライトスルー選択可能
置換方式	LRU (Least Recently Used) アルゴリズム	LRU (Least Recently Used) アルゴリズム

【注】 キャッシュメモリの詳細については,「SH7730 グループ ハードウェアマニュアル (RJJ09B0339)」の「キャッシュ」の章を参照してください。

2.2 オペランドキャッシュの書き込み方式について

SH7730 のオペランドキャッシュの書き込み方式は、コピーバックモードとライトスルーモードの 2 種類があります。本節では、オペランドキャッシュが有効でキャッシング可能な領域に対してデータを書き込むときのオペランドキャッシュの動作について説明します。また、オペランドキャッシュを使用する際に、ソフトウェア側での注意点についても説明します。詳細は、「SH7730 グループ ハードウェアマニュアル (RJJ09B0339)」の「キャッシュ 書き込み動作」を参照ください。また、読み出し動作につきましては、「SH7730 グループ ハードウェアマニュアル (RJJ09B0339)」の「キャッシュ 読み出し動作」を、プリフェッチ動作につきましては、「SH7730 グループ ハードウェアマニュアル (RJJ09B0339)」の「キャッシュ プリフェッチ動作」を参照ください。

2.2.1 ライトスルーモードの動作

仮想アドレスのビット [12:5] でインデックスされる各ウェイのキャッシュラインから、タグ、V ビット、U ビットおよび LRU ビットを読み出します。その後、仮想アドレスを MMU により変換した物理アドレスのビット [28:10] と、各ウェイから読み出したタグを比較します。

- タグが一致かつ V ビットが 1 の場合→(1) キャッシュヒット時へ
- タグが一致かつ V ビットが 0 の場合→(2) キャッシュミス時へ
- タグが不一致かつ V ビットが 0 の場合→(2) キャッシュミス時へ
- タグが不一致かつ V ビットが 1 かつ U ビットが 0 の場合→(2) キャッシュミス時へ
- タグが不一致かつ V ビットが 1 かつ U ビットが 1 の場合→(2) キャッシュミス時へ

1. キャッシュヒット時

キャッシュと外部メモリに同じ内容を同時に書き込みます。

- (1) ヒットしたウェイのデータ部の、仮想アドレスのビット [4:0] でインデックスされるデータ位置に対し、CPU がアクセスサイズに応じてデータを書き込みます。
- (2) ライトスルーバッファに書き込みデータを書き込みます。これにより CPU はライトスルーバッファへの書き込みが完了すると、外部メモリへの書き込みの完了を待たずに次の動作へ移ります。
- (3) 仮想アドレスに対応する外部メモリに対してもデータの書き込みを行います。
- (4) ヒットしたウェイが最新となるように LRU ビットを更新します。
- (5) U ビットは更新されません。

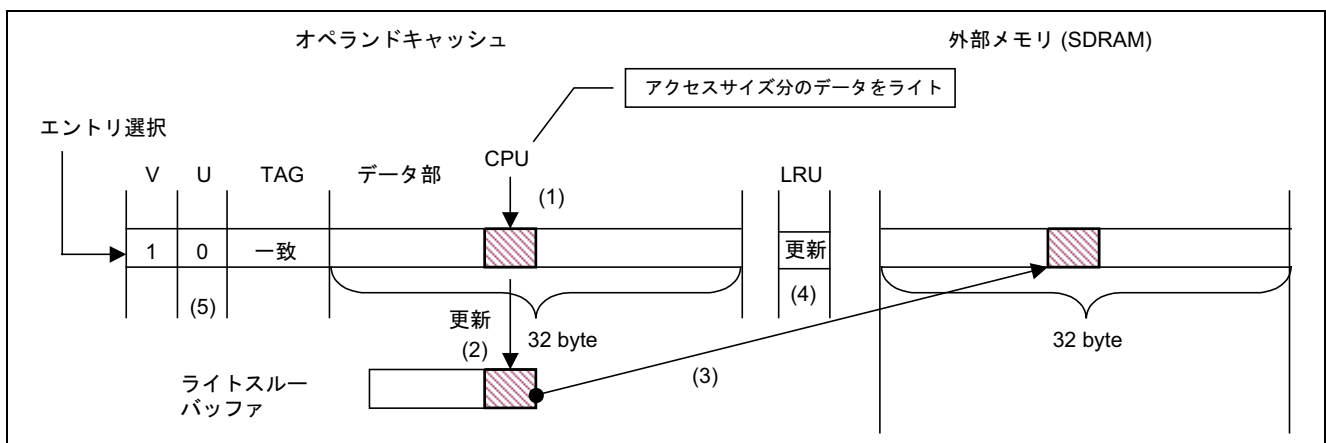


図1 ライトスルーモード (キャッシュヒット時)

2. キャッシュミス時

キャッシュへの書き込みは行いません。

- (1) ライトスルーバッファに書き込みデータを書き込みます。これにより CPU はライトスルーバッファへの書き込みが完了すると、外部メモリへの書き込みの完了を待たずに次の動作へ移ります。
- (2) 仮想アドレスに対応した外部メモリへ、指定されたアクセスサイズでデータの書き込みを行います。
- (3) タグ、V ビット、U ビット、LRU ビットも更新されません。

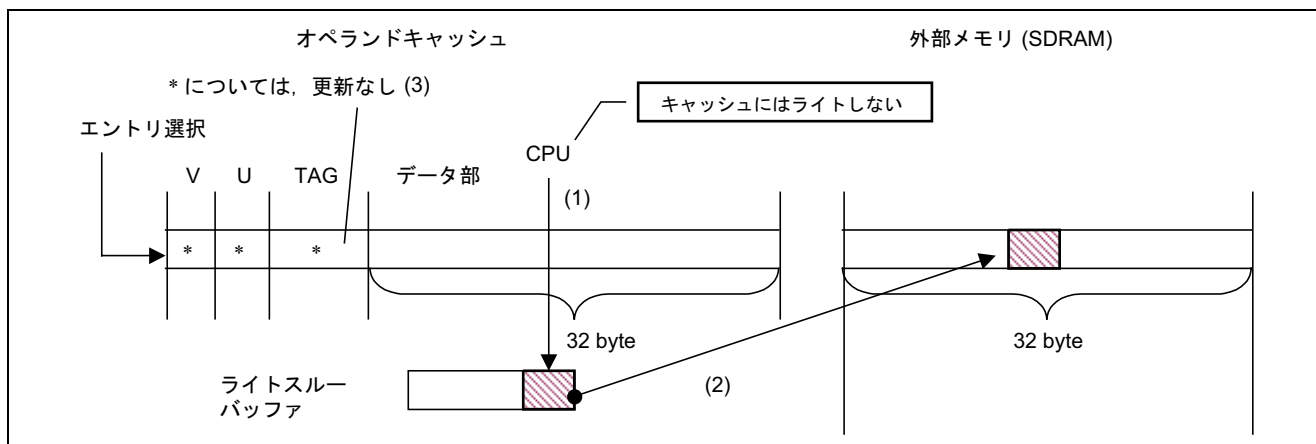


図2 ライトスルーモード (キャッシュミス時)

2.2.2 コピーバックモードの動作

仮想アドレスを MMU により変換した物理アドレスのビット [28:10] と、各ウェイから読み出したタグを比較します。

- タグが一致かつ V ビットが 1 の場合→(1) キャッシュヒット時へ
- タグが一致かつ V ビットが 0 の場合→(2) キャッシュミス時 (書き戻しなし) へ
- タグが不一致かつ V ビットが 0 の場合→(2) キャッシュミス時 (書き戻しなし) へ
- タグが不一致かつ V ビットが 1 かつ U ビットが 0 の場合→(2) キャッシュミス時 (書き戻しなし) へ
- タグが不一致かつ V ビットが 1 かつ U ビットが 1 の場合→(3) キャッシュミス時 (書き戻しあり) へ

1. キャッシュヒット時

- (1) ヒットしたウェイのデータ部の、仮想アドレスのビット [4:0] でインデックスされるデータ位置に対し、CPU がアクセスサイズに応じてデータを書き込みます。
- (2) U ビットに 1 を書き込みます。
- (3) ヒットしたウェイが最新となるように LRU ビットを更新します。

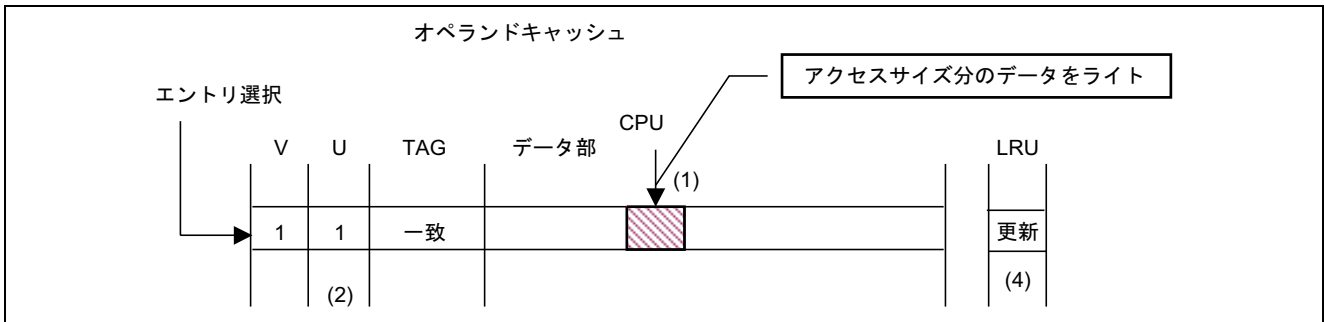


図 3 コピーバックモード (キャッシュヒット時)

上記のようにキャッシュのみ書き込みを行い、外部メモリには書き込みを行いません。そのため、オペランドキャッシュと外部メモリの内容が一致しないことがあります。キャッシュしている領域を外部デバイスが参照する場合 (DMAC 等) は、ソフトウェアにて書き戻し処理を行い、オペランドキャッシュと外部メモリのコヒーレンスを保証しないと、想定通りの動作を行わない場合がありますのでご注意ください。オペランドキャッシュのコヒーレンシ制御については、「SH7730 グループ アプリケーションノート キャッシュ オペランドキャッシュの書き戻し例 (RJJ06B0923)」を参照してください。

2. キャッシュミス時 (書き戻しなし)

- (1) 書き込み対象ウェイのデータ部の、仮想アドレスのビット [4:0] でインデックスされるデータ位置に対し、CPU がアクセスサイズに応じてデータを書き込みます。
- (2) 仮想アドレスに対応する物理アドレス空間から、書き込み対象ウェイのキャッシュラインへデータを読み込みます (ただし、すでに書き込み済みのキャッシュミスしたデータを除く)。キャッシュ 1 ライン分のデータが読み込まれている間、CPU は次の処理を実行することができます。
- (3) キャッシュに 1 ライン分のデータの読み込みが完了した時点で、物理アドレスによるタグを登録し、V ビットに 1 を、U ビットに 1 を書き込みます。
- (4) 書き込みを行ったウェイが最新となるように LRU ビットを更新します。

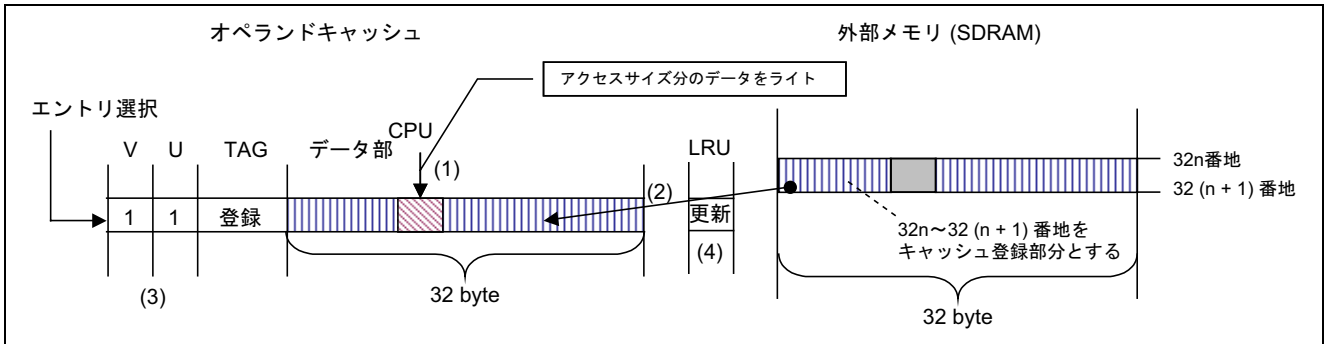


図4 コピーバックモード (キャッシュミス, 書き戻しなし時)

3. キャッシュミス時 (書き戻しあり)

- (1) LRU アルゴリズムにより、最も古くアクセスされた置換対象ウェイのキャッシュラインを決定します。その後、置換対象ウェイのキャッシュラインのタグとデータ部をライトバックバッファへ退避します。
- (2) 置換対象ウェイのデータ部の、仮想アドレスのビット [4:0] でインデックスされるデータ位置に対し、CPU がアクセスサイズに応じてデータを書き込みます。
- (3) 仮想アドレスに対応する物理アドレス空間から、置換対象ウェイのキャッシュラインへデータを読み込みます (ただし、すでに書き込み済みのキャッシュミスしたデータを除く)。キャッシュ 1 ライン分のデータが読み込まれている間、CPU は次の処理を実行することができます。
- (4) キャッシュに 1 ライン分のデータの読み込みが完了した時点で、物理アドレスによるタグを登録し、V ビットに 1 を、U ビットに 1 を書き込みます。
- (5) 置換したウェイが最新となるように LRU ビットを更新します。
- (6) ライトバックバッファのデータを外部メモリへ書き戻します。

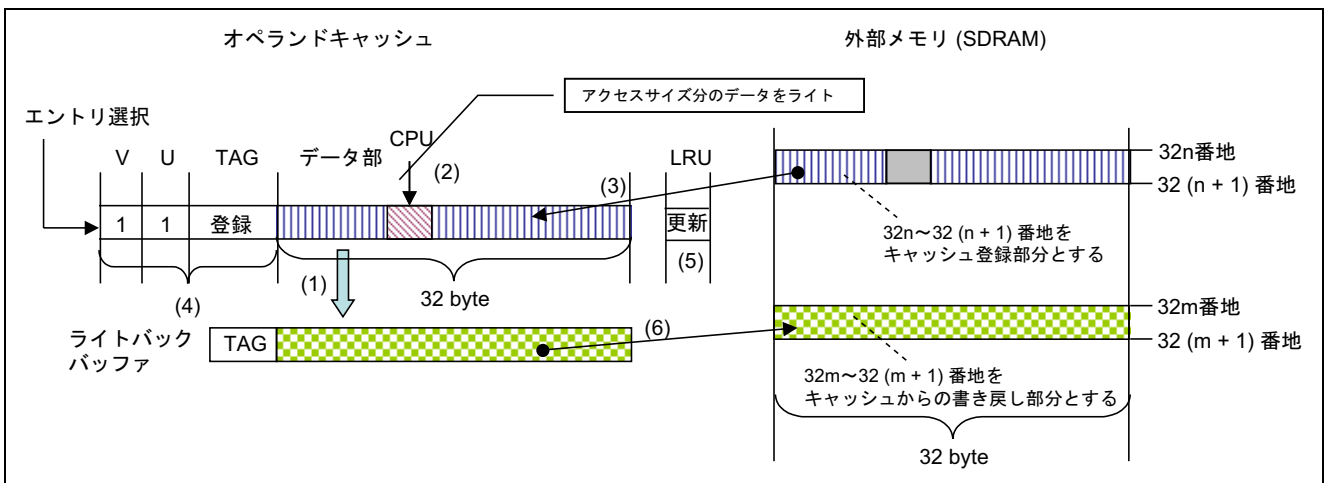


図5 コピーバックモード (キャッシュミス, 書き戻しあり時)

2.3 ライトスルーとコピーバックのメリット/デメリット

表2 ライトスルーとコピーバックのメリット/デメリット

	メリット	デメリット
ライトスルー	CPU からのアクセスに関しては、オペランドキャッシュと外部メモリの内容が常に一致します。	書き込み頻度が上がると、外部バスの使用頻度が上がってしまうため、動作速度はコピーバック方式よりも低速となります。
コピーバック	キャッシュヒット時はキャッシュにのみデータを書き込んでいるので外部バスを有効活用することにより高速動作が可能となります。	CPU からのアクセスに関しても、オペランドキャッシュと外部メモリの内容が異なる場合があるため、ソフト側でコヒーレンスを保証する必要があります。

2.4 キャッシュ機能の設定

2.4.1 キャッシュの設定処理

ここでは、キャッシュの設定手順について説明します。キャッシュモードの設定は、キャッシュ制御レジスタ (CCR) で行います。CCR レジスタ設定後、キャッシング可能領域へのアクセスを行う前の処理として本設定例では ICBI 命令を発行しています。また、キャッシュ制御レジスタを操作するプログラムは、キャッシング不可の P2 領域のプログラムまたは IL メモリ上に配置しなければなりません。本設定例では、キャッシング不可の P2 領域に配置しています。

本応用例では、キャッシュモード更新中にキャッシュ有効空間をアクセスする割り込み処理を受け付けないように、ステータスレジスタ (SR) の例外/割り込みブロックビット (BL) を 1 に設定します。

図 6 に、命令キャッシュおよびオペランドキャッシュを有効にする場合の設定フロー例を示します。

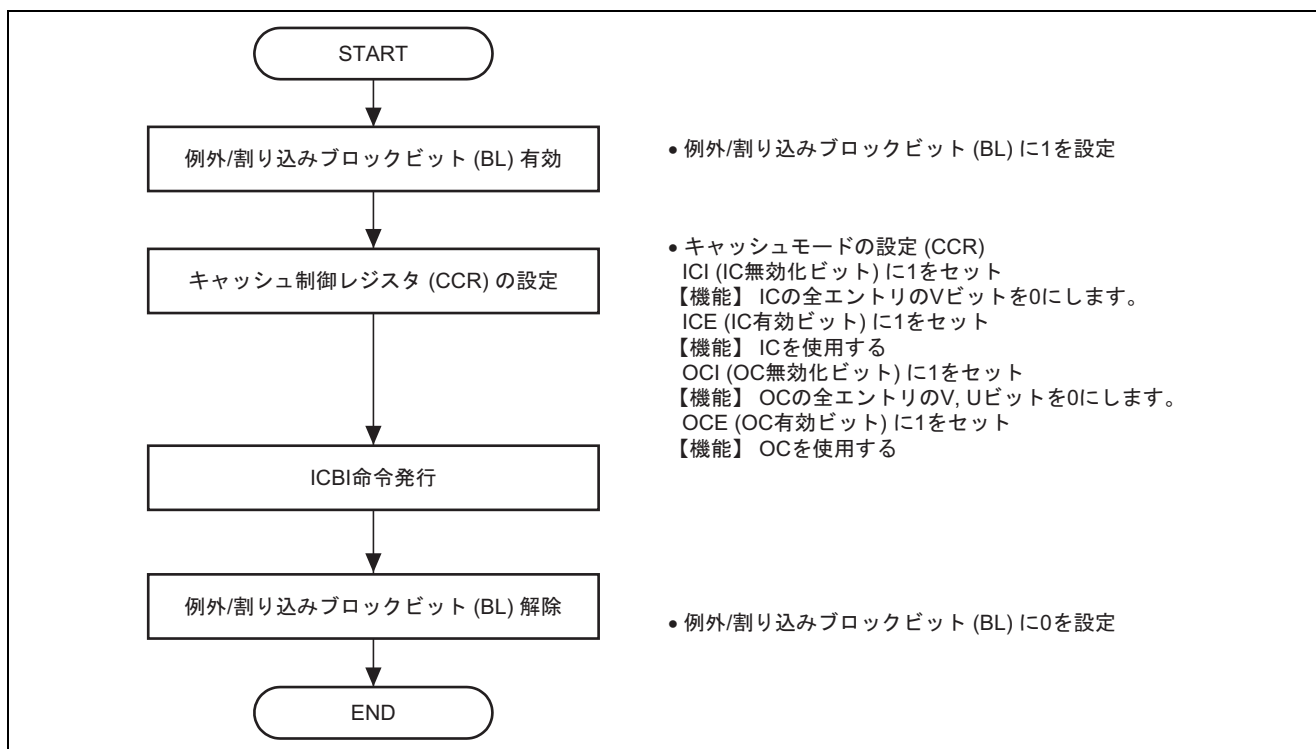


図 6 キャッシュ設定フロー例

- 【注】 1. 本サンプルプログラムは、特権モードで実行してください。
 2. 本サンプルプログラムでは、MMU は初期値のまま (MMU 無効) にしています。

2.4.2 キャッシュ制御レジスタ (CCR) への設定値

表3 CCR レジスタ ビット構成

ビット:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ICI	—	—	ICE	—	—	—	—	OCI	CB	WT	OCE
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

表4 CCR レジスタ ビット説明

ビット	ビット名	初期値	R/W	説明
11	ICI	0	R/W	IC 無効化ビット このビットに 1 を書き込むと IC の全エントリの V ビットを 0 にします。読み出すと常に 0 が読み出されます。
8	ICE	0	R/W	IC 有効ビット IC の使用を選択します。ただしアドレス変換が行われる場合は、ページ管理情報の C ビットも 1 でなければ IC を使用できません。 0: IC を使用しない 1: IC を使用する
3	OCI	0	R/W	OC 無効化ビット このビットに 1 を書き込むと OC の全エントリの V, U ビットを 0 にします。読み出すと常に 0 が読み出されます。
2	CB	0	R/W	コピーバックビット P1 領域のキャッシュへの書き込みモードを示します。 0: ライトスルーモード 1: コピーバックモード
1	WT	0	R/W	ライトスルーモード P0, U0, P3 領域のキャッシュへの書き込みモードを示します。ただし、アドレス変換が行われる場合は、ページ管理情報の WT ビットの値を優先します。 0: コピーバックモード 1: ライトスルーモード
0	OCE	0	R/W	OC 有効ビット OC の使用を選択します。ただしアドレス変換が行われる場合は、ページ管理情報の C ビットも 1 でなければ OC を使用できません。 0: OC を使用しない 1: OC を使用する

表 5 に参考プログラムでのキャッシュ設定を示します。

表 5 キャッシュの設定

レジスタ名	アドレス	設定値	機能
キャッシュ制御 レジスタ (CCR)	H'FF00 001C	H'0000 0909	ICI = 1: IC 無効化ビット ICE = 1: IC 有効ビット OCI = 1: OC 無効化ビット CB = 0: ライトスルーモード WT = 0: コピーバックモード OCE = 1: OC 有効ビット 【注】 ICI, OCI は読み出し時 0 です。

2.5 参考プログラムの動作

本参考プログラムでは、コピーバックモードで動作した場合、オペランドキャッシュと外部メモリの内容が一致しない場合を示します。

2.5.1 に参考プログラムの処理フローを示します。また、2.5.2 に参考プログラム動作時のキャッシュ動作を示します。

2.5.1 参考プログラムの処理フロー

参考プログラムの処理フローについて説明します。図7 と合わせて参照ください。

- (1) 命令キャッシュおよびオペランドキャッシュが無効の状態（CCR.OC = 0）で、H'0C00 0000 から 1 キャッシュラインサイズ分（32 バイト）を H'00 でフィルします。オペランドキャッシュが無効であるため、オペランドキャッシュヘータはライトされず、外部メモリにデータがライトされます。
- (2) 命令キャッシュおよびオペランドキャッシュをコピーバックモードで有効にします。
- (3) H'0C00 0000 から 32 バイト分を H'55 でフィルします。このとき、コピーバックモードに設定されているので、外部メモリヘータはライトされず、オペランドキャッシュにデータがライトされます。
- (4) H'AC00 0000 から 32 バイト分のデータ（外部メモリ上のデータ）と、H'0C00 0000 から 32 バイト分のデータ（キャッシュにヒットするので、オペランドキャッシュ上のデータ）を比較します。外部メモリ上のデータは H'00、オペランドキャッシュ上のデータは H'55 であるので、データは一致しません。

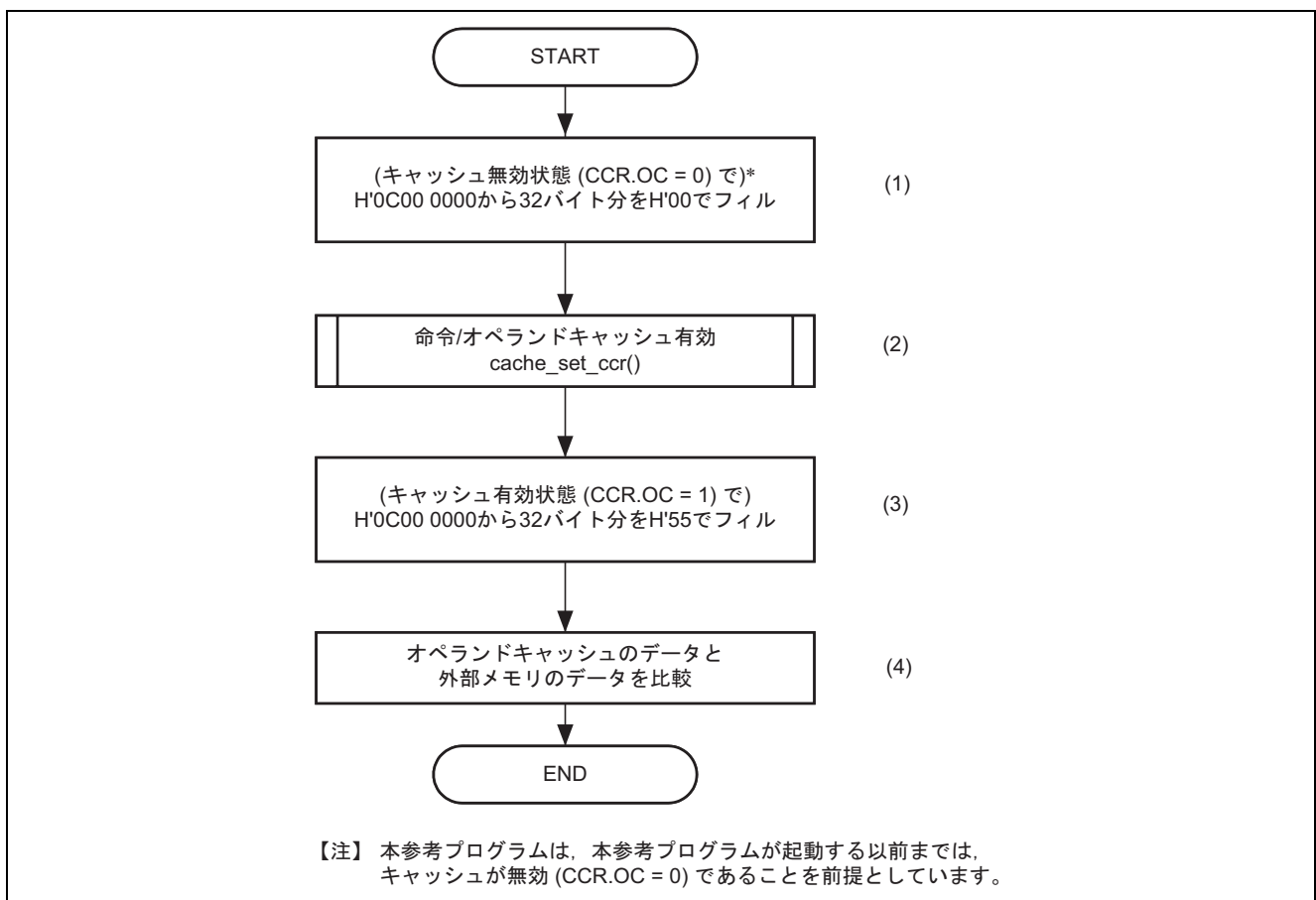


図7 参考プログラムの処理フロー

2.5.2 参考プログラム動作時のキャッシュ動作

オペラントキャッシュ無効状態からオペラントキャッシュ有効状態にして、初回の H'0C00 0000 アドレスへ H'55 データを書き込む場合のキャッシュミス（書き戻しなし）動作と、2 回目の H'0C00 0001 アドレスへ H'55 データを書き込む場合のキャッシュヒット動作について記載し、参考プログラム動作時のキャッシュ動作について説明します。

1. 初回書き込み時

オペラントキャッシュ有効後、初回の H'0C00 0000 アドレスへ H'55 データを書き込む場合のキャッシュミス（書き戻しなし）動作について説明します。図 8 と合わせて参照ください。

- (1) H'0C00 0000 アドレスにデータを書き込むのでエントリはビット [12:5] より 0 が選択されます。このエントリ 0 の各ウェイのキャッシュラインから、タグ、V ビット、U ビットおよび LRU ビットを読み出します。データを書き込む処理の直前で、CCR レジスタの OC 無効化ビット (OCI) の無効化によりオペラントキャッシュの全エントリの V、U ビットは 0 になるため、読み出した V ビット、U ビットは、0 になります。また、読み出したタグはパワーオンリセット、マニュアルリセットで初期化されませんので、不定値となります。
- (2) H'0C00 0000 アドレスのビット [28:10] と、各ウェイから読み出したタグの比較を行い、V ビット、U ビットの値を参照します。(1) よりタグは不定値で、V ビット、U ビットは 0 であるため、キャッシュ動作は、キャッシュミス（書き戻しなし）となります。
- (3) オペラントキャッシュへのデータ H'55 の書き込み先は、置換対象ウェイのエントリ 0 の、H'0C00 0000 アドレスのビット [4:0] でインデックスされるデータ位置に対し、データ H'55 を書き込みます。
- (4) H'0C00 0000 ~ H'0C00 001F アドレスに相当する外部メモリから、置換対象ウェイのエントリ 0 のキャッシュラインのデータ部へデータを読み込みます（ただし、すでに書き込み済みのキャッシュミスしたデータを除く）。
- (5) エントリ 0 のキャッシュラインのデータ部へ 1 ライン分のデータの読み込み (4) が完了した時点で、置換対象ウェイのエントリ 0 のタグに、H'0C00 0000 アドレスのビット [28:10] を登録し、V ビットに 1 を、U ビットに 1 を書き込みます。
- (6) また置換したウェイが最新となるように LRU ビットを更新します。

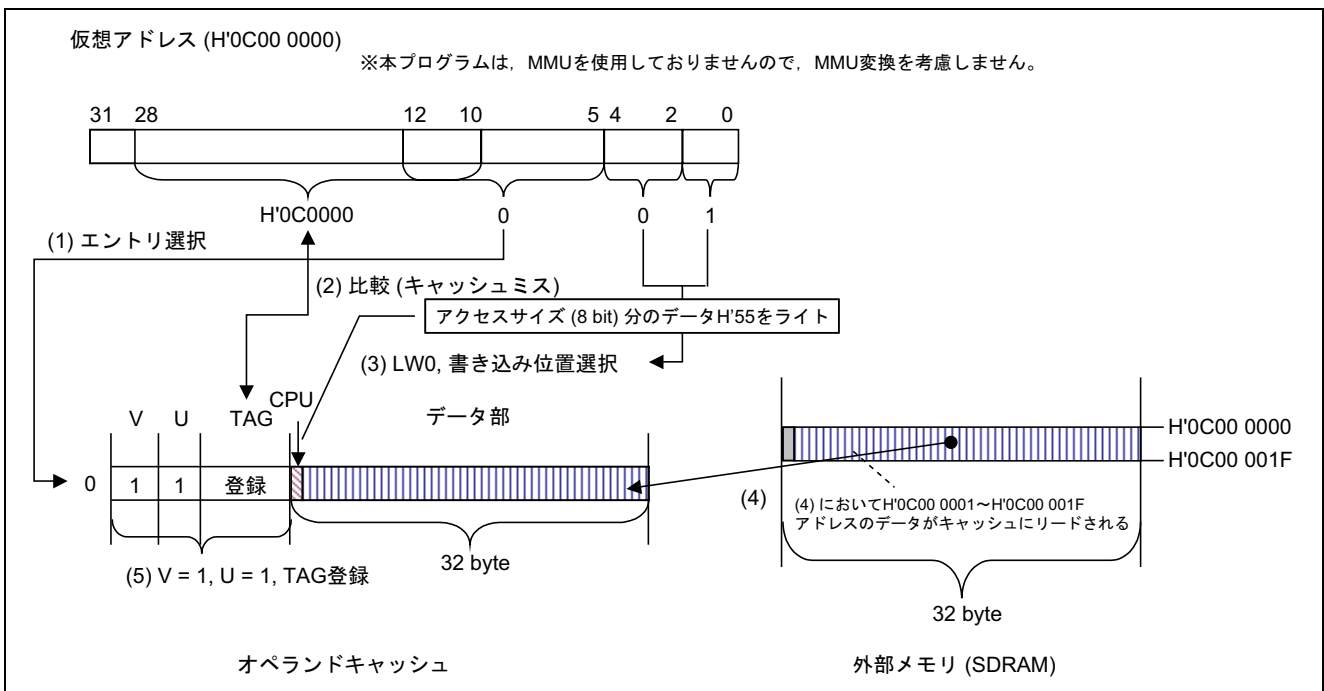


図 8 参考プログラムのキャッシュ動作 (初回書き込み時)

2. 2 回目書き込み時

初回の H'0C00 0000 アドレスへ H'55 データを書き込み処理後 2 回目の H'0C00 0001 アドレスへ H'55 データを書き込む場合のキャッシュヒット動作について説明します。図 9 と合わせて参照ください。

- (1) H'0C00 0001 アドレスにデータを書き込むのでエントリはビット [12:5] より 0 が選択されます。このエントリ 0 の各ウェイのキャッシュラインから、タグ、V ビット、U ビットおよび LRU ビットを読み出します。1 回目の書き込み処理より、タグに H'0C0000 が、V ビットに 1、U ビットに 1 が格納されているウェイが存在します。
- (2) H'0C00 0001 アドレスのビット [28:10] と、各ウェイから読み出したタグの比較を行い、V ビット、U ビットの値を参照します。(1)よりタグが H'0C0000 で一致し、V ビットが 1 のエントリを含むウェイが存在するため、キャッシュヒットとなります。
- (3) オペランドキャッシュへのデータ H'55 の書き込み先は、ヒットしたウェイのエントリ 0 の、H'0C00 0001 アドレスのビット [4:0] でインデックスされるデータ位置に対し、データ H'55 を書き込みます。
- (4) U ビットに 1 を書き込みます。
- (5) ヒットしたウェイが最新となるように LRU ビットを更新します。

以降 H'0C00 001F までの書き込み処理において、すべてキャッシュヒットとなり、キャッシュにのみデータが設定されます。また、キャッシュの書き戻しが発生しません。そのため、キャッシュと外部メモリとでデータが異なります。

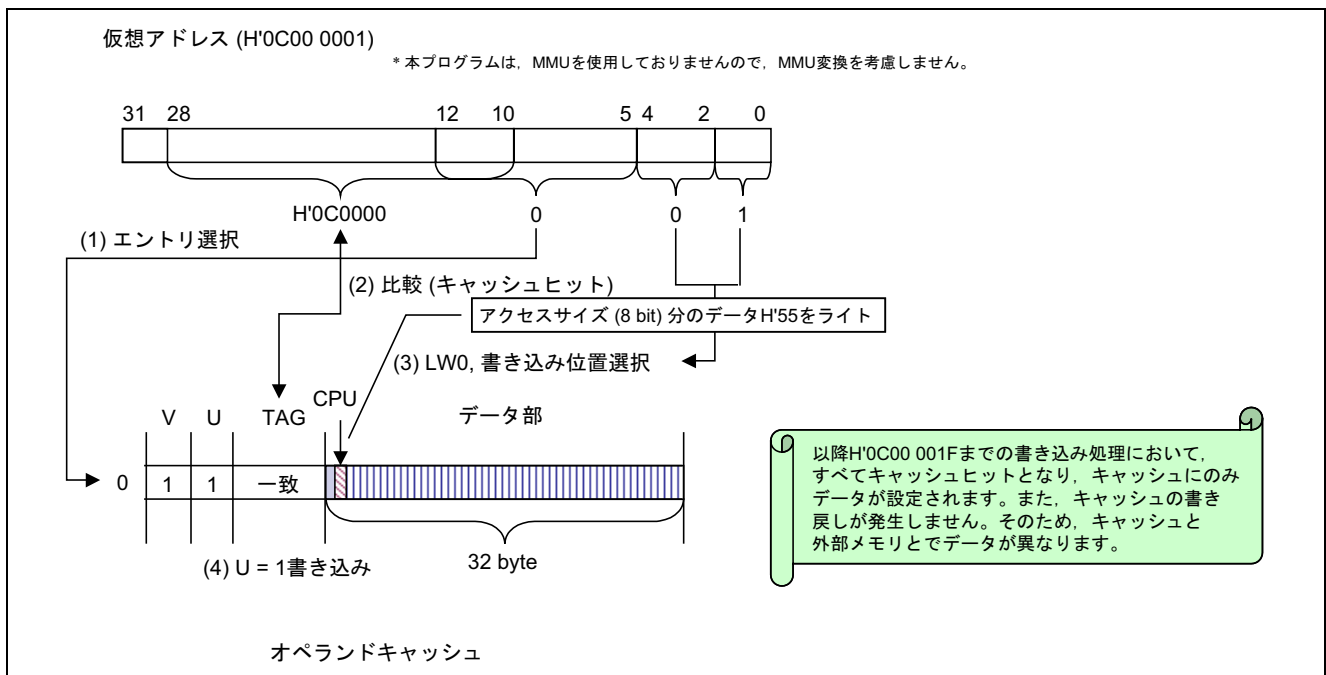


図 9 参考プログラムのキャッシュ動作 (2 回目書き込み時)

2.6 参考プログラムでのセクション配置

コンパイラ拡張機能 `#pragma section` を使用してキャッシュ制御レジスタ操作を行う関数のセクション名を切り替えます。参考プログラムでは、`cache_set_ccr()` 関数のプログラム領域を PnonCache セクションに切り替えます。この PnonCache セクションを SH7730 のキャッシング不可の P2 領域に配置します。セクション配置 (アドレス指定) はリンカのオプションで指定します。

表 6 に参考プログラムにおけるセクション情報を示します。

表 6 参考プログラムのセクション情報

セクション名	セクション用途	配置アドレス (仮想アドレス)	
P	プログラム領域 (指定なしの場合)	0x0000 3000	P0 領域 (キャッシング可能, MMU アドレス変換可能)
C	定数領域		
C\$BSEC	未初期化データ領域用アドレス構造体		
C\$DSEC	初期化データ領域用アドレス構造体		
D	初期化データ (初期値)		
B	未初期化データ領域	0x0C00 0000	
R	初期化データ領域		
S	スタック領域	0x0FFF F9F0	
INTHandler	例外/割り込みハンドラ	0x8000 0800	P1 領域 (キャッシング可能, MMU アドレス変換不可)
VECTTBL	リセットベクタテーブル 割り込みベクタテーブル		
INTTBL	割り込みマスクテーブル		
PIntPRG	割り込み関数		
SP_S	TLBmiss ハンドラ用スタック領域		
RSTHandler	リセットハンドラ	0xA000 0000	P2 領域 (キャッシング不可, MMU アドレス変換不可)
PResetPRG	リセットプログラム		
PnonCache	プログラム領域 (キャッシュ無効アクセス)		

3. 参考プログラム例

(1) サンプルプログラムリスト"sh7730.c"

```

1  /*"FILE COMMENT"***** Technical reference data *****
2  * System Name   : SH7730 Sample Program
3  * File Name     : sh7730.c
4  * Abstract      : SH7730 キャッシュ設定例 Sample Program
5  * Version       : Ver 1.00
6  * Device        : SH7730
7  * Tool-Chain    : SuperH RISC engine Standard Toolchain Ver.9.1.1.0
8  * OS            : None
9  * H/W Platform : アルファプロジェクト製 SH-4A ボード 型番 AP-SH4A-1A
10 * Description   : SH7730 キャッシュ設定例のサンプルプログラムです。
11 *              :
12 * Operation     :
13 * Disclaimer    :
14 *              :
15 * Copyright (C) 2008. Renesas Technology Corp., All Rights Reserved.
16 *              :
17 *****
18 * History       : 27.May.2008 Ver. 1.00 First Release
19 *"FILE COMMENT END"*****
20 #include <machine.h>
21 #include "cache.h"          /* キャッシュ機能 追加 */
22
23 // #include "typedefine.h"
24 #ifdef __cplusplus
25 // #include <ios>           // Remove the comment when you use ios
26 // _SINT ios_base::Init::init_cnt; // Remove the comment when you use ios
27 #endif
28
29 void main(void);
30 #ifdef __cplusplus
31 extern "C" {
32 void abort(void);
33 }
34 #endif
35
36 /* ---- SDRAM 領域アドレス ---- */
37 #define D_SDRAM_ADDR1 (unsigned char *) (0x0c000000) /* キャッシング可能な P0 領域 */
38 #define D_SDRAM_ADDR2 (unsigned char *) (0xac000000) /* キャッシング不可の P2 領域 */
39
40 /* ==== プロトタイプ宣言 ==== */
41 void main(void);
42
43 /*"FUNC COMMENT"*****
44 * ID           :
45 * Outline      : サンプルプログラムメイン(キャッシュメモリ使用例)
46 * Include     :
47 * Declaration  : void main(void)
48 * Description  : コピーバックモードで動作した場合,
49 *              : オペランドキャッシュと外部メモリの内容が
50 *              : 一致しない例を示すサンプルプログラムです。
51 *              :
52 *              :   手順
53 *              :   1.
54 *              :   オペランドキャッシュが無効の状態,
55 *              :   0x0C000000 から 1 キャッシュラインサイズ分
56 *              :   (32 バイト)を 0x00 でフィルします。
57 *              :   外部メモリを 0x00 でフィルします。
58 *              :   2.
59 *              :   オペランドキャッシュを有効にし,
60 *              :   0x0C000000 から 32 バイト分を 0x55 でフィルします。
61 *              :   オペランドキャッシュを 0x55 でフィルします。
    
```

```

62 *          : 3 .
63 *          : 0x0C000000 から 32 バイト分のデータ
64 *          : (オペランドキャッシュ)と 0xAC000000 から
65 *          : 32 バイト分のデータ(外部メモリ)と比較し、
66 *          : 外部メモリ上のデータとオペランドキャッシュ
67 *          : 上のデータが一致していないことを示します。
68 *          :
69 * Disclaimer      : 本サンプルプログラムは、
70 *          : 命令/オペランドキャッシュが無効の状態
71 *          : コールされることを想定しています。
72 *          : 初期設定例では、下記 main()関数がコール
73 *          : される以前の段階で、命令/オペランドキャッシュ
74 *          : を有効に設定しています。
75 *          : そのため、初期設定例のコードから
76 *          : 命令/オペランドキャッシュを有効にする処理
77 *          : を削除して本サンプルプログラムを
78 *          : コールしてください。
79 *          :
80 * Argument        : none
81 * Return Value    : none
82 * Calling Functions :
83 * "FUNC COMMENT END"*****
84 void main(void)
85 {
86     int i;
87     unsigned char *ptr1,*ptr2;
88
89     /* ==== 【注】 ==== */
90     /* ==== 命令/オペランドキャッシュが無効の状態
91     /* ==== での処理が行なわれる ==== */
92     /* ==== ことを前提とします。 ==== */
93
94     /* ==== オペランドキャッシュが無効な状態
95     /* ==== でキャッシング可能な P0 領域からライト ==== */
96     /* ==== 外部メモリを 0x00 でフィルしている ==== */
97     ptr1 = D_SDRAM_ADDR1;
98     for(i=0; i<32 ; i++){
99         *ptr1++ = 0x00;
100     }
101
102     /* ==== 命令/オペランドキャッシュ有効 ==== */
103     cache_set_ccr(D_CACHE_I_ON | D_CACHE_O_ON | D_CACHE_I_INVALID | D_CACHE_O_INVALID);
104
105     /* ==== オペランドキャッシュが有効な状態
106     /* ==== でキャッシング可能な P0 領域からライト ==== */
107     /* ==== オペランドキャッシュを 0x55 でフィルしている ==== */
108     ptr1 = D_SDRAM_ADDR1;
109     for(i=0; i<32 ; i++){
110         *ptr1++ = 0x55;
111     }
112
113     /* ==== 外部メモリ上のデータとオペランドキャッシュ上のデータを比較 ==== */
114     ptr1 = D_SDRAM_ADDR1; /* キヤシング可能な P0 領域 */
115     ptr2 = D_SDRAM_ADDR2; /* キヤシング不可の P2 領域 */
116
117     for(i=0; i<32; i++){
118         if(*ptr1++ == *ptr2++){
119             while(1){
120                 /* 外部メモリ上のデータとオペランドキャッシュ上のデータが */
121                 /* 一致している(意図しないケース) */
122             }
123         }
124     }
125
126     /* ==== 上記無限ループに入らないことにより、データが一致していないことを示す ==== */
127     while(1){
128         /* Program end */

```

```
128     }  
129  
130     }  
131
```

(2) サンプルプログラムリスト"cache.c"

```

1  /*"FILE COMMENT"***** Technical reference data *****
2  * System Name   : SH7730 Sample Program
3  * File Name     : cache.c
4  * Abstract      : SH7730 キャッシュ設定例 Sample Program
5  * Version       : Ver 1.00
6  * Device        : SH7730
7  * Tool-Chain    : SuperH RISC engine Standard Toolchain Ver.9.1.1.0
8  * OS            : None
9  * H/W Platform : アルファプロジェクト製 SH-4A ボード 型番 AP-SH4A-1A
10 * Description    : SH7730 キャッシュ設定例のサンプルプログラムです。
11 *               :
12 * Operation     :
13 * Disclaimer     :
14 *               :
15 * Copyright (C) 2008. Renesas Technology Corp., All Rights Reserved.
16 *
17 *****
18 * History       : 27.May.2008 Ver. 1.00 First Release
19 *"FILE COMMENT END"*****/
20 #include <machine.h>
21 #include "iodefine.h"
22 #include "cache.h"
23
24 #pragma section nonCACHE /* CS0 キャッシュ無効空間に配置します*/
25 /*"FUNC COMMENT"*****
26 * ID            :
27 * Outline       : キャッシュの設定
28 * Include       :
29 * Declaration   : void cache_set_ccr(unsigned int i_mode)
30 * Description   : キャッシュレジスタの設定を行います。
31 *               :
32 * Argument      : unsigned int i_mode
33 *               : 以下のモードを論理和で設定します。
34 *               : D_CACHE_I_INVALID   : IC 無効化
35 *               : D_CACHE_I_ON       : IC 有効
36 *               : D_CACHE_O_INVALID  : OC 無効化
37 *               : D_CACHE_O_ON       : OC 有効
38 *               : D_CACHE_IO_ON      : IC・OC 有効
39 *               : D_CACHE_O_WT       : ライトスルーモード
40 *               : D_CACHE_OFF        : IC・OC 無効化
41 * Return Value  : none
42 * Calling Functions :
43 *"FUNC COMMENT END"*****/
44 void cache_set_ccr(unsigned int i_mode)
45 {
46     /* ==== 例外/割り込みブロックビット(BL)設定 ==== */
47     set_cr(get_cr() | 0x10000000);
48
49     /* ==== キャッシュレジスタの設定 ==== */
50     CACHE.CCR.LONG = i_mode;
51
52     /* ==== キャッシュを有効にするため ICBI 発行 ==== */
53     icbi(0);
54
55     /* ==== 例外/割り込みブロックビット(BL)解除 ==== */
56     set_cr(get_cr() & ~(0x10000000));
57
58 }
    
```

(3) サンプルプログラムリスト"cache.h"

```

1  /*"FILE COMMENT"***** Technical reference data *****
2  * System Name   : SH7730 Sample Program
3  * File Name    : cache.h
4  * Abstract     : SH7730 キャッシュ設定例 Sample Program
5  * Version      : Ver 1.00
6  * Device       : SH7730
7  * Tool-Chain   : SuperH RISC engine Standard Toolchain Ver.9.1.1.0
8  * OS           : None
9  * H/W Platform : アルファプロジェクト製 SH-4A ボード 型番 AP-SH4A-1A
10 * Description  : SH7730 キャッシュ設定例のサンプルプログラムです。
11 *              :
12 * Operation    :
13 * Disclaimer    :
14 *              :
15 * Copyright (C) 2008. Renesas Technology Corp., All Rights Reserved.
16 *
17 *****
18 * History       : 27.May.2008 Ver. 1.00 First Release
19 *"FILE COMMENT END"*****/
20
21 #ifndef __CACHE_DEF_H__
22 #define __CACHE_DEF_H__
23
24 /* ==== マクロ定義 ==== */
25 /* ---- キャッシュ設定 ---- */
26 #define D_CACHE_OFF                0x0000u
27 #define D_CACHE_I_INVALID          0x0800u
28 #define D_CACHE_I_ON                0x0100u
29 #define D_CACHE_O_INVALID          0x0008u
30 #define D_CACHE_O_ON                0x0001u
31 #define D_CACHE_IO_ON              (CACHE_I_ON | CACHE_O_ON)
32 #define D_CACHE_O_WT                0x0002u
33
34 /* ---- ページ対象領域 ---- */
35 #define D_CACHE_PURGE_SDRAM_START  *(volatile unsigned long*)(0x0C000000)
36 #define D_CACHE_PURGE_SDRAM_END   *(volatile unsigned long*)(0x0FFFFFFF)
37
38 /* ==== 関数宣言 ==== */
39 void cache_set_ccr(unsigned int i_mode);
40 void cache_Purge_OCBP(unsigned long *i_start, unsigned long *i_end);
41
42
43 #endif /* __CACHE_DEF_H__ */
    
```

3.1 実行結果

サンプルプログラム実行後の外部メモリとキャッシュの内容を、以下に示します。

キャッシング不可の P2 領域からリードした場合、外部メモリからデータをリードします。(図 10)

キャッシング可能の P0 領域からリードした場合、かつ、キャッシュメモリ上にリード要求したデータが存在する場合は、外部メモリからはデータをリードせず、キャッシュメモリからデータを読み出します。(図 11)

図 10, 図 11 より、キャッシュメモリのライト方式をコピーバックモードに設定した場合は、キャッシュメモリの内容と外部メモリの値が異なる場合があることが分かります。

このように、コピーバックモードに設定にした場合は、キャッシュメモリと外部メモリの同期が取れていない場合があります。キャッシュしている領域を外部デバイスが参照する場合 (DMAC 等) は、ソフトウェアにて書き戻し処理を行う必要があります。この方法については、「SH7730 グループ アプリケーションノート キャッシュ オペランドキャッシュの書き戻し例 (RJJ06B0923)」を参照してください。

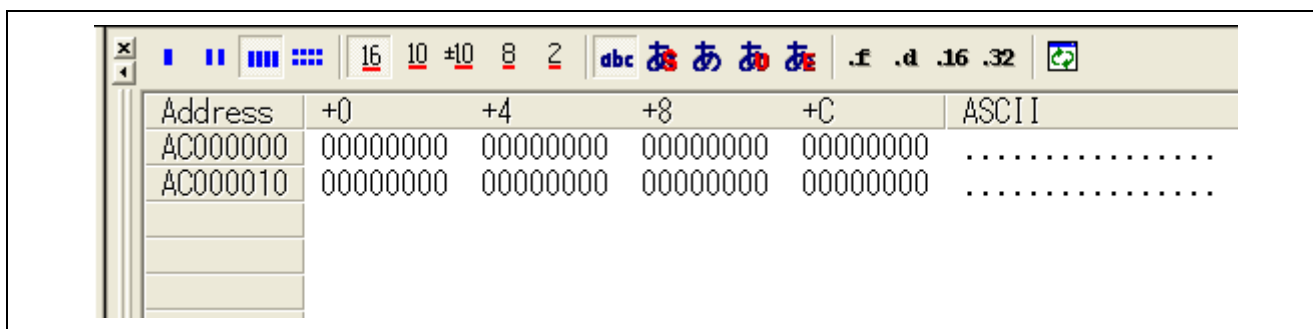


図 10 キャッシング不可の P2 領域をアクセスした場合のメモリダンプ

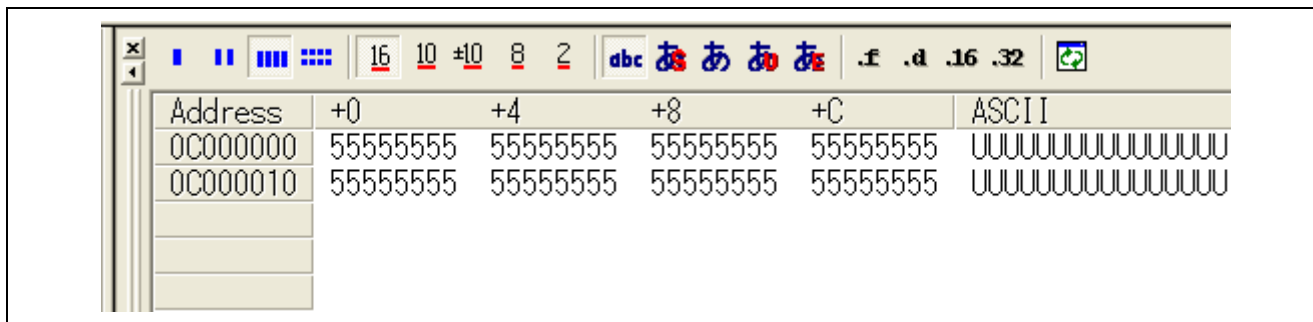


図 11 キャッシング可能の P0 領域をアクセスした場合のメモリダンプ

4. 参考ドキュメント

- ソフトウェアマニュアル
SH-4A ソフトウェアマニュアル (RJJ09B0090)
(最新版をルネサス テクノロジホームページから入手してください。)
- ハードウェアマニュアル
SH7730 グループ ハードウェアマニュアル (RJJ09B0339)
(最新版をルネサス テクノロジホームページから入手してください。)

ホームページとサポート窓口

ルネサス テクノロジホームページ

<http://japan.renesas.com/>

お問合せ先

<http://japan.renesas.com/inquiry>

csc@renesas.com

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2009.02.09	—	初版発行

すべての商標および登録商標は、それぞれの所有者に帰属します。

本資料ご利用に際しての留意事項

1. 本資料は、お客様に用途に応じた適切な弊社製品をご購入いただくための参考資料であり、本資料中に記載の技術情報について弊社または第三者の知的財産権その他の権利の実施、使用を許諾または保証するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例など全ての情報の使用に起因する損害、第三者の知的財産権その他の権利に対する侵害に関し、弊社は責任を負いません。
3. 本資料に記載の製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的、あるいはその他軍事用途の目的で使用しないでください。また、輸出に際しては、「外国為替および外国貿易法」その他輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
4. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例などの全ての情報は本資料発行時点のものであり、弊社は本資料に記載した製品または仕様等を予告なしに変更することがあります。弊社の半導体製品のご購入およびご使用に当たりますは、事前に弊社営業窓口で最新の情報をご確認いただきますとともに、弊社ホームページ (<http://www.renesas.com>) などを通じて公開される情報に常にご注意ください。
5. 本資料に記載した情報は、正確を期すため慎重に制作したものです。万一本資料の記述の誤りに起因する損害がお客様に生じた場合においても、弊社はその責任を負いません。
6. 本資料に記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を流用する場合は、流用する情報を単独で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。弊社は、適用可否に対する責任を負いません。
7. 本資料に記載された製品は、各種安全装置や運輸・交通用、医療用、燃焼制御用、航空宇宙用、原子力、海底中継用の機器・システムなど、その故障や誤動作が直接人命を脅かしあるいは人体に危害を及ぼすおそれのあるような機器・システムや特に高度な品質・信頼性が要求される機器・システムでの使用を意図して設計、製造されたものではありません（弊社が自動車用と指定する製品を自動車に使用する場合を除きます）。これらの用途に利用されることをご検討の際には、必ず事前に弊社営業窓口へご照会ください。なお、上記用途に使用されたことにより発生した損害等については弊社はその責任を負いかねますのでご了承願います。
8. 第7項にかかわらず、本資料に記載された製品は、下記の用途には使用しないでください。これらの用途に使用されたことにより発生した損害等につきましては、弊社は一切の責任を負いません。
 - 1) 生命維持装置。
 - 2) 人体に埋め込み使用するもの。
 - 3) 治療行為（患部切り出し、薬剤投与等）を行うもの。
 - 4) その他、直接人命に影響を与えるもの。
9. 本資料に記載された製品のご使用につき、特に最大定格、動作電源電圧範囲、放熱特性、実装条件およびその他諸条件につきましては、弊社保証範囲内でご使用ください。弊社保証値を越えて製品をご使用された場合の故障および事故につきましては、弊社はその責任を負いません。
10. 弊社は製品の品質および信頼性の向上に努めておりますが、特に半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。弊社製品の故障または誤動作が生じた場合も人身事故、火災事故、社会的損害などを生じさせないよう、お客様の責任において冗長設計、延焼対策設計、誤動作防止設計などの安全設計（含むハードウェアおよびソフトウェア）およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特にマイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
11. 本資料に記載の製品は、これを搭載した製品から剥がれた場合、幼児が口に入れて誤飲する等の事故の危険性があります。お客様の製品への実装後に容易に本製品が剥がれることがなきよう、お客様の責任において十分な安全設計をお願いします。お客様の製品から剥がれた場合の事故につきましては、弊社はその責任を負いません。
12. 本資料の全部または一部を弊社の文書による事前の承諾なしに転載または複製することを固くお断りいたします。
13. 本資料に関する詳細についてのお問い合わせ、その他お気付きの点等がございましたら弊社営業窓口までご照会ください。

D039444