

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ひ申し上げます。

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、誤りがな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。

SH7216 グループ

USB ファンクションモジュール USB マスストレージクラス

要旨

本アプリケーションノートは、SH7216 の USB ファンクションモジュールの使用方法、および USB マスストレージクラスに対応したファームウェアの作成例について掲載しています。

本アプリケーションノートの内容とソフトウェアは USB ファンクションモジュールの応用例を説明するもので、その内容を保障するものではありません。

動作確認デバイス

SH7216

目次

1. はじめに	2
2. 概要	3
3. USB マスストレージクラス (Bulk-Only Transport) の概要	5
4. 開発環境	10
5. サンプルプログラム概要	14
6. 参考ドキュメント	29

1. はじめに

1.1 仕様

本アプリケーションノートは、SH7216 の USB ファンクションモジュールの使用方法、および USB マスストレージクラスに対応したファームウェアの作成例を掲載しています。

1.2 使用機能

- 割り込みコントローラ (INTC)
- ピンファンクションコントローラ (PFC)
- USB ファンクションモジュール (USB)

1.3 適用条件

- マイコン: SH7216
- 動作周波数:
 - 内部クロック 200 MHz
 - バスクロック 50 MHz
 - 周辺クロック 50 MHz
- 統合開発環境: ルネサス テクノロジ製
High-performance Embedded Workshop Ver.4.07.00
- C コンパイラ: ルネサス テクノロジ製
SuperH RISC engine ファミリー C/C++ コンパイラパッケージ Ver.9.02 Release 00
- コンパイルオプション: High-performance Embedded Workshop でのデフォルト設定
 (-cpu=sh2afpu -pic=1 -object="\$(CONFIGDIR)\¥\$(FILELEAF).obj"
 -debug -gbr=auto -chginpath -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

- SH7216 グループ アプリケーションノート USB ファンクションモジュール HID クラス (RJJ06B1072)

2. 概要

本プログラムでは USB ファンクションモジュール (USB) を使用したコントロール転送、バルク転送、およびマスストレージクラスコマンド対応処理を行います。

SH7216 内蔵 USB ファンクションモジュールの特長を以下に示します。

- USB プロトコルを自動処理
- エンドポイント 0 に対する USB 標準コマンドを自動処理 (一部コマンドはファームウェアで処理する必要があります。)
- 転送スピード：フルスピード
- 割り込み要求：USB 送受信に必要な各種割り込み信号を生成
- クロック：USB 発振器 (48MHz) による外部入力
- 低消費電力モードを搭載
- バストランシーバを内蔵
- エンドポイント構成：表 1 に示す構成

表 1 エンドポイント構成

エンドポイント名	名称	転送タイプ	最大パケットサイズ	FIFO バッファ容量	DMA 転送
エンドポイント 0	EP0s	セットアップ	8 バイト	8 バイト	—
	EP0i	コントロールイン	16 バイト	16 バイト	—
	EP0o	コントロールアウト	16 バイト	16 バイト	—
エンドポイント 1	EP1	バルクイン	64 バイト	64×2 (128) バイト	可能
エンドポイント 2	EP2	バルクアウト	64 バイト	64×2 (128) バイト	可能
エンドポイント 3	EP3	インタラプトイン	16 バイト	16 バイト	—
エンドポイント 4	EP4	バルクイン	64 バイト	64×2 (128) バイト	可能
エンドポイント 5	EP5	バルクアウト	64 バイト	64×2 (128) バイト	可能
エンドポイント 6	EP6	インタラプトイン	16 バイト	16 バイト	—
エンドポイント 7	EP7	バルクイン	64 バイト	64 バイト	—
エンドポイント 8	EP8	バルクアウト	64 バイト	64 バイト	—
エンドポイント 9	EP9	インタラプトイン	16 バイト	16 バイト	—

システム構成例を図 1 に示します。

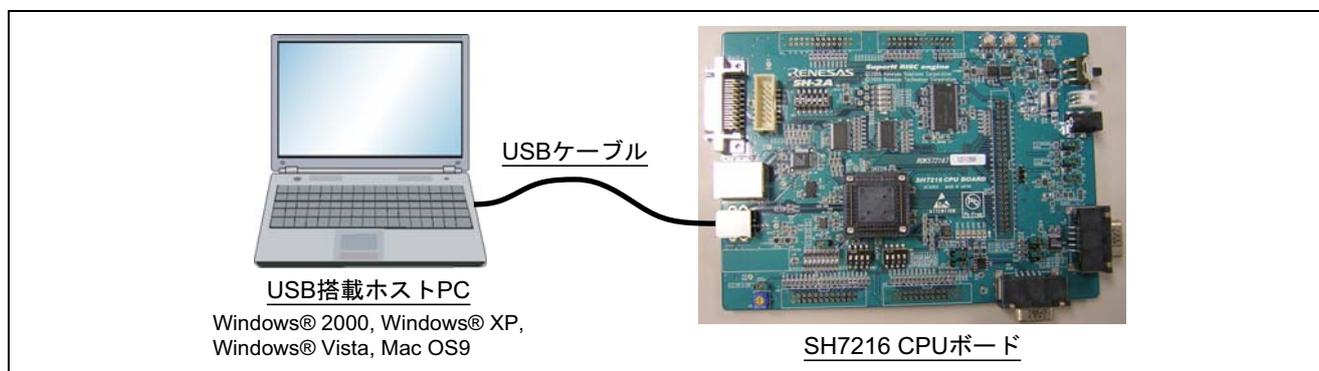


図 1 システム構成例

本システムは、SH7216 を搭載したルネサス テクノロジ製の SH7216 CPU ボード、Windows® 2000、Windows® XP、Windows® Vista、または Mac OS9 を OS として搭載する PC によって構成されています。

本システムは、ホスト PC と SH7216 CPU ボードを USB で接続し、SH7216 CPU ボード上の SDRAM を RAM Disk として動作させることにより、ホスト PC から SH7216 CPU ボードの SDRAM へのデータ書き込み、および SDRAM からの読み出しができます。

上記 OS に標準で付属している USB マスストレージクラス (Bulk-Only Transport) のデバイスドライバを使用することが可能です。

本システムの特長を以下に示します。

1. サンプルプログラムにより、SH7216 の USB モジュールを評価可能
2. サンプルプログラムは USB のコントロール転送とバルク転送をサポート
3. E10A (USB 接続型エミュレータ) によるデバッグが可能
4. プログラムを追加作成することでインタラプト転送*も対応可能

【注】 * インタラプト転送のプログラムは、お客様で作成していただく必要があります。
なお、SH7216 はアイソクロナス転送には対応しておりません。

3. USB マスストレージクラス (Bulk-Only Transport) の概要

この章では、USB マスストレージクラス (Bulk-Only Transport) について説明します。

USB のストレージ関連システムを開発する場合に参考としてください。なお、規格の詳細は、“6. 参考ドキュメント”の(3)と(4)をご参照ください。

3.1 USB マスストレージクラスについて

USB マスストレージクラスとは、大規模記憶装置をホスト PC に接続しデータの書き込み、読み出し等の動作を行う機器に適合するよう規格化されたクラスです。

ホスト PC に、このクラスのファンクションであることを伝えるためには、Interface Descriptor の bInterfaceClass フィールドに値 H'08 を記述する必要があります。また、USB マスストレージクラスでは String Descriptor を用いて Serial Number をホストへ伝える必要があります。本サンプルプログラムではユニコードで 000000000001 を返信しています。

ホスト PC とファンクション間でデータ転送をする場合、USB に規定されている 4 つの転送方法 (コントロール転送、バルク転送、インタラプト転送、アイソクロナス転送) を用いてデータの転送を行います。どの転送方法をどのように使用するかは、プロトコルコードとして定められています。

USB マスストレージクラスではデータ転送プロトコルとして次の 2 種類があります。

- USB Mass Storage Class Bulk-Only Transport
- USB Mass Storage Class Control/Bulk/Interrupt (CBI) Transport

USB Mass Storage Class Bulk-Only Transport は名前の示すとおり、バルク転送のみ使用したデータ転送プロトコルです。

USB Mass Storage Class Control/Bulk/Interrupt (CBI) Transport は、コントロール転送、バルク転送、インタラプト転送を使用したデータ転送プロトコルです。CBI Transport は、さらにインタラプト転送を使用するデータ転送プロトコル、使用しないデータ転送プロトコルの 2 種類に分かれています。

本サンプルプログラムでは、USB Mass Storage Class Bulk-Only Transport をデータ転送プロトコルとして使用します。

ホスト PC がデータのロードやセーブをするために機器を使用する場合、ホスト PC からファンクションに対して命令 (コマンド) を与えます。ファンクションは送られたコマンドを実行することによりデータのロードやセーブが行えます。ホスト PC からファンクションに対して送られるコマンドはサブクラスコードとして定められています。

3.2 サブクラスコードについて

サブクラスコードとは、ホスト PC からコマンドトランスポートでファンクションに送られるコマンドフォーマットを表す値です。コマンドフォーマットの種類としては 7 種類あり、表 2 に示すサブクラスコードが定められています。

表 2 コマンドトランスポートフォーマット

サブクラスコード	コマンドの規格
H'01	Reduced Block Commands (RBC), T10/1240-D
H'02	Attachment Packet Interface (ATAPI) for CD-ROMs. SFF-8020i, Multi-Media Command Set 2 (MMC-2)
H'03	Attachment Packet Interface (ATAPI) for Tape. QIC-157
H'04	USB Mass Storage Class UFI Command Specification
H'05	Attachment Packet Interface (ATAPI) for Floppies. SFF-8070i
H'06	SCSI Primary Commands -2 (SPC-2), Revision 3 or later

ホスト PC に、機器が対応しているコマンドフォーマットを伝えるためには、Interface Descriptor の bInterfaceSubClass フィールドにサブクラスコード値を記述する必要があります。

本サンプルプログラムでは、サブクラスコード値 H'06 の SCSI Primary Commands を使用します。

3.3 Bulk-Only Transport について

Bulk-Only Transport はバルク転送のみ使用し、ホスト PC とファンクション間でデータの転送が行われます。

バルク転送は、データを送信する向きにより 2 つに分けることができます。ホストコントローラからファンクションにデータを送信する転送をバルクアウト転送。ホストコントローラにファンクションからデータを送信する転送をバルクイン転送と言います。

Bulk-Only Transport では、バルクアウト転送とバルクイン転送をあらかじめ定めた組み合わせにすることにより、ホスト-ファンクション間のデータ転送を行います。Bulk-Only Transport は必ず図 2 に示すバルク転送の組み合わせになります。それぞれのバルク転送には異なった意味がありステージ (トランスポート) として管理します。

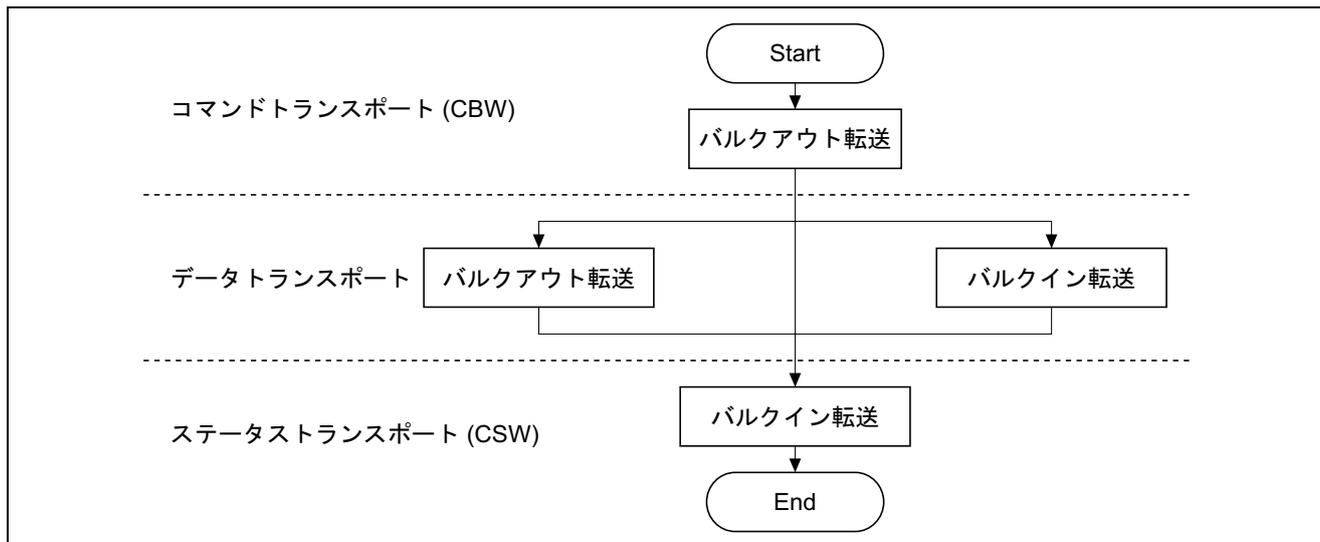


図 2 転送方法とトランスポートの関係

ホスト PC に、Bulk-Only Transport プロトコルの使用を伝えるためには、Interface Descriptor の bInterfaceProtocol フィールドに値 H'50 を記述することが必要です。

3.3.1 コマンドトランスポートについて

コマンドトランスポートはホスト PC がファンクションにバルクアウト転送を用いてコマンドを送ります。このコマンドパッケージが Command Block Wrapper (CBW) として定義されており、Bulk-Only Transport は必ず CBW から始まります。

CBW は、ホスト PC からバルクアウト転送を使用して 31 バイト長のパッケージで送られて来ます。

内容は表 3 に示すフォーマットで送られます。

表 3 コマンドトランスポートフォーマット

	7	6	5	4	3	2	1	0
H'00-H'03	dCBWSignature							
H'04-H'07	dCBWTag							
H'08-H'0B	dCBWDataTransferLength							
H'0C	bmCBWFlags							
H'0D	リザーブ(0)				bCBWLUN			
H'0E	リザーブ(0)				bCBWCBLength			
H'0F-H'1E	CBWCB							

各フィールドの内容を記します。

- dCBWSignature:
データパケットが CBW であると認知するためのフィールド。値は H'43425355 (リトルエンディアン) です。
- dCBWTag:
コマンドブロックタグ。CBW と対応する CSW を結びつけるために存在し、ホスト PC が指定します。
- dCBWDataTransferLength:
データトランスポートの予定データ長。ここが 0 の場合データトランスポートは存在しません。
- bmCBWFlags:
このフィールドのビットは、ビット 7 が 0 の場合、データトランスポートはバルクアウト転送で行われ、1 の場合、バルクイン転送で行われます。ビット 6~0 は 0 固定です。
- bCBWLUN:
コマンドブロックが送られている装置の論理ユニット番号 (Logical Unit Number)。
- bCBWCBLength:
次の CBWCB フィールドの有効バイト数を表します。
- CBWCB:
ファンクションによって実行されるコマンドブロックを格納するフィールド。このフィールドにホスト PC が実行したいコマンド (本サンプルプログラムでは SCSI コマンド) が入ります。

3.3.2 ステータストランスポートについて

ステータストランスポートはファンクションがホスト PC にバルクイン転送を用いてコマンド実行結果を送ります。

このステータスパケットが Command Status Wrapper (CSW) として定義されており、Bulk-Only Transport は必ず CSW で終わります。

CSW は、ホスト PC へバルクイン転送を使用して 13 バイト長のパケットで送ります。

内容は表 4 に示すフォーマットで送られます。

表 4 ステータストランスポートフォーマット

	7	6	5	4	3	2	1	0
H'0-H'3	dCSWSignature							
H'4-H'7	dCSWTag							
H'8-H'B	dCSWDataResidue							
H'C	bCSWStatus							

各フィールドの内容を記します。

- dCSWSignature:
データパケットが CSW であると認知するためのフィールド。値は H'53425355 (リトルエンディアン) です。
- dCSWTag:
コマンドブロックタグ。CBW に CSW を結びつけるために存在し、CBW の dCBWTag フィールドと同じ値が入ります。
- dCSWDataResidue:
CBW の dCBWDataTransferLength 値と実際にファンクションが処理したデータ量の相違を報告します。
- bCSWStatus:
コマンドの成功あるいは失敗を示します。コマンドが正常に完了した場合、ファンクションはこのフィールドを H'00 にセットします。ゼロ以外の値は次の通りとし、コマンド実行時の不具合を示します。コマンドフェイルは H'01、フェーズエラーは H'02。

3.3.3 データトランスポートについて

データトランスポートは、ホスト PC とファンクション間のデータ転送を行うトランスポートです。例えば、Read/Write コマンドでは、データトランスポートにてストレージ各セクタの実データを送信します。

データトランスポートは複数のバストランザクションで構成されます。

データトランスポートで行われるデータ転送はバルクアウト転送かバルクイン転送のどちらか一方です。どちらになるかは CBW データの bmCBWFlags フィールドで決定されます。

(1) データトランスポート (バルクアウト転送) について

データトランスポートがバルクアウト転送の場合について説明します。

この状態になるのは、CBW データの bmCBWFlags フィールドのビット 7 が 0 であり、CBW データの dCBWDataTransferLength フィールドが 0 ではない場合です。

ここでは CBW データの dCBWDataTransferLength フィールドで予定した長さのデータをファンクションが受信します。転送されるデータは、CBW データの CBWCB フィールドで指定された SCSI コマンドを実行する際に必要なデータです。

(2) データトランスポート (バルクイン転送) について

データトランスポートがバルクイン転送の場合について説明します。

この状態になるのは、CBW データの bmCBWFlags フィールドのビット 7 が 1 であり、CBW データの dCBWDataTransferLength フィールドが 0 ではない場合です。

ここでは CBW データの dCBWDataTransferLength フィールドで予定した長さのデータをファンクションがホスト PC に送信します。転送されるデータは、CBW データの CBWCB フィールドで指定された SCSI コマンドを実行した結果のデータです。

3.4 クラスコマンドについて

クラスコマンドとは、USB のクラス定義ごとに定められているコマンドです。クラスコマンドはコントロール転送を使用します。

USB Mass Storage Class Bulk-Only Transport をデータ転送プロトコルとして使用する場合にサポートしなければならないコマンドは 2 種類あります。表 5 にクラスコマンドを示します。

表 5 クラスコマンド一覧

bRequest フィールド値	コマンド	コマンドの意味
255 (H'FF)	Bulk-Only Mass Storage Reset	インタフェースをリセットする
254 (H'FE)	Get Max LUN	サポートする LUN の数を調べる

Bulk-Only Mass Storage Reset コマンドを受信した場合、ファンクションは USB Mass Storage Class Bulk-Only Transport で使用するすべてのインタフェースをリセットします。

Get Max LUN コマンドを受信した場合、ファンクションは使用できる最大の論理ユニット番号を返答します。当サンプルシステムの場合、論理ユニットは 1 つなので返答値は 0 をホストに返答します。

3.5 サブクラスコード (SCSI transparent command set) について

ファンクションはホスト PC より送信される CBW 内サブクラスコマンドに対応し、各コマンドを処理する必要があります。

本サンプルプログラムでは、SCSI コマンドの中から表 6 に示す 11 個のコマンドをサポートしています。また、未サポートのコマンドについては、ホスト PC に対し CSW を使用し「コマンドフェイルである」と報告しています。

表 6 サポートコマンド一覧

Operation Code	コマンド名	コマンドの動作
H'00	TEST UNIT READY	メディアが使用可能か否かを調べる
H'03	REQUEST SENSE	前のコマンドでエラーが発生したときどのようなエラーが発生したかをホストに伝える
H'12	INQUIRY	ドライブに関する情報をホストに伝える
H'1A	MODE SENSE (6)	ドライブの状態をホストに伝える
H'1B	STOP/START UNIT	メディアの着脱を制御します
H'1E	PREVENT ALLOW MEDIUM REMOVAL	メディアの着脱を禁止 / 許可します
H'23	READ FORMAT CAPACITY	メディアのフォーマット情報をホストに伝える
H'25	READ CAPACITY	メディアのセクタに関する情報をホストに伝える
H'28	READ (10)	指定された読み出しセクタから、指定セクタ量のデータを読み出す
H'2A	WRITE (10)	指定された書き込みセクタから、指定セクタ量のデータを書き込む
H'2F	VERIFY (10)	メディア上のデータにアクセス可能を確かめる

4. 開発環境

この章では、本システムの開発に使用した開発環境について説明します。本システムの開発は、以下のデバイス (ツール) を使用します。

- SH7216 CPU ボード (型名 R0K572167) ルネサス テクノロジ製
- E10A-USB Emulator ルネサス テクノロジ製
- E10A PC (Windows® 2000, Windows® XP)
- USB ホスト PC (Windows® 2000, Windows® XP, Windows® Vista)
- USB ケーブル
- High-performance Embedded Workshop 4 (以下 HEW4) ルネサス テクノロジ製

4.1 ハードウェア環境

図 3 に各デバイスの接続形態を示します。

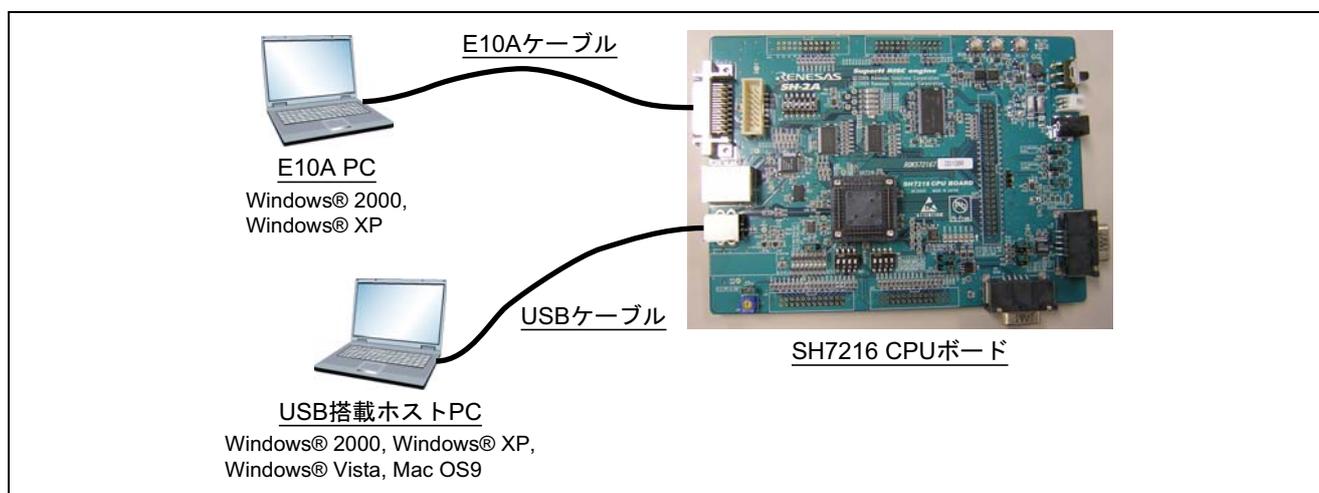


図 3 デバイスの接続形態

(1) SH7216 CPU ボード

本システムでは内蔵ROMとSDRAMを使用するため、SH7216 CPU ボードをMCU 拡張モード2 (内蔵ROMとSDRAM有効) で動作させる必要があります。そこで、SH7216 CPU ボードのディップスイッチのSW1を出荷時の設定から表 7 に示すように変更する必要があります。電源を投入する前に、これらの設定をよくご確認ください。その他のディップスイッチを変更する必要はありません。

表 7 ディップスイッチの設定

出荷時 (モード 6)	変更後 (モード 2)	ディップスイッチの機能
SW1-1 (FWE) OFF	SW1-1 (FWE) ON	内蔵フラッシュメモリの書き込み/消去プロテクト
SW1-2 (MD1) OFF	SW1-2 (MD1) OFF	MD1 端子状態
SW1-3 (MD0) ON	SW1-3 (MD0) ON	MD0 端子状態

(2) USB ホスト PC

USB ポート搭載の Windows® 2000, Windows® XP, Windows® Vista または Mac OS9 をインストールしたパソコンを USB ホスト PC として使用します。本システムでは、上記 OS に標準で搭載されている USB マスストレージクラス (Bulk-Only Transport) のデバイスドライバを使用しますので、新たにドライバをインストールする必要はありません。

(3) E10A PC

USB ポート搭載の Windows® 2000, Windows® XP をインストールしたパソコンを E10A PC として使用します。E10A-USB 用 PC の USB コネクタに E10A-USB エミュレータを接続し、接続用のケーブルを介して E10A-USB と CPU ボードを接続してください。接続後、HEW4 を起動してエミュレーションを行います。

4.2 ソフトウェア環境

ソースコードのコンパイル、リンク、およびデバッグは HEW4 で行ってください。HEW4 は本フォルダ内 MSC.hws をダブルクリックすることで起動します。

4.2.1 サンプルプログラム

サンプルプログラムとして必要なファイルは、すべて MSC フォルダ内に収められています。HEW4 がインストールされたパソコンに、このフォルダごと移動して頂くと、すぐにサンプルプログラムを使用することができます。

フォルダに含まれるファイルを図 4 に示します。

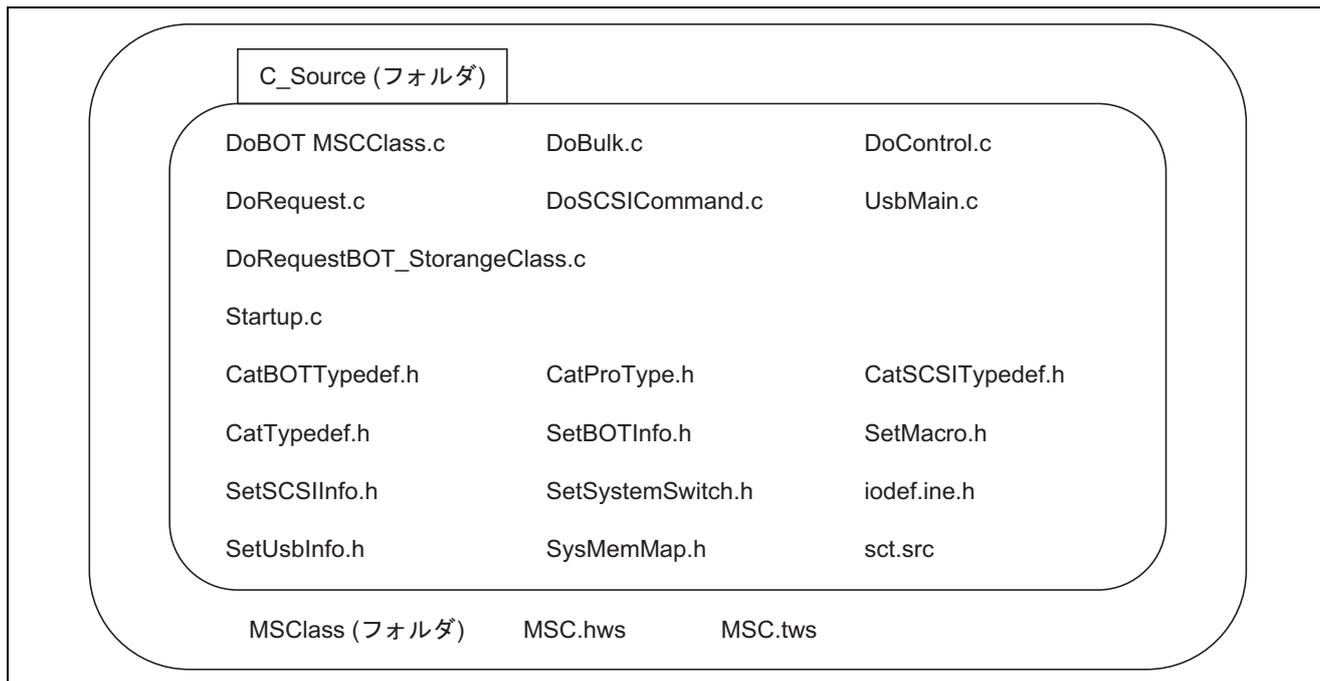


図 4 フォルダに含まれるファイル

4.2.2 コンパイルおよびリンク

ソースコードのコンパイルは HEW4 で行ってください。

4.3 プログラムのロードと実行方法

プログラムのロードと実行方法について説明します。

4.3.1 プログラムのロード

SH7216 CPU ボードにサンプルプログラムをロードするには、以下のような手順で行います。

- HEW4 をインストールした E10A-USB 用 PC に E10A-USB を接続してください。
- ユーザケーブルで E10A-USB と SH7216 CPU ボードを接続してください。
- SH7216 CPU ボードの電源を投入してください。
- sh7216_usb_msc フォルダ内の MSC.hws を実行してください。
- 「デバッグ > 接続」を選択してください。
- エミュレータのモードの選択を要求されるので、「SH7216 (R5F72167A)」, 「E10A-USB Emulator」を選択してください。
- SH7216 CPU ボードのリセットスイッチを押し、「OK」ボタンを押してください。
- 動作周波数の入力を求められるので、実装している水晶発振子 (12.50MHz) の周波数を入力してください。
- ID Code の入力を求められるので、「E10A」を入力してください。
- 「デバッグ > ダウンロード > All Download Modules」を選択することでプログラムのダウンロードが行われます。

4.3.2 プログラムの実行

「デバッグ > リセット後実行」を選択することでプログラムが実行されます。

4.4 RAM Disk の使用方法

Windows® XP を用いた場合を例に以下に説明します。

プログラムを実行した状態で、USB ケーブルのシリーズ B コネクタを SH7216 CPU ボードに挿入し、反対側のシリーズ A コネクタを USB ホスト PC に接続します。

コントロール転送およびバルク転送を用いたエミュレーション終了後、デバイスマネージャの USB コントローラの下に USB 大容量記憶装置デバイスが表示され、ディスクドライブの下に RENESAS EX RAM Disk USB Device が表示されます。その結果、ホスト PC は SH7216 CPU ボードを記憶デバイスとして認識し、マイコンピュータの中にローカルディスクがマウントされます。

次にローカルディスクをフォーマットします。

ローカルディスクを選択し、マウスの右ボタンをクリックし、フローティングメニュー内のフォーマットを選択します。ドライブのフォーマット選択ウインドが開かれるので、フォーマットの設定を行います。ファイルシステム選択項目が FAT であることを確認し、開始ボタンをクリックしてください。

フォーマットの実行確認画面が出力されるので、OK ボタンをクリックしてください。

フォーマットが完了するとフォーマット完了のメッセージウインドウが出力されるので、OK ボタンをクリックしてください。

ドライブのフォーマット選択ウインドに戻るので、閉じるボタンをクリックしてウインドを閉じてください。

以上で SH7216 CPU ボードを USB 接続の RAM Disk として使用できます。

4.5 RAM Disk の設定変更について

本サンプルプログラムで使用する RAM Disk の設定の変更方法について説明します。

4.5.1 リムーバブル・固定ディスク

本サンプルプログラムでは、RAM Disk をリムーバブルディスクとして使用しています。

SetSystemSwitch.h 内の「#define REMOVABLE_DISK」をコメント化し、コメント化されている「#undef REMOVABLE_DISK」を有効にすることにより、固定ディスクとして使用することができます。

4.5.2 RAM Disk の容量変更

本サンプルプログラムでは、16M バイト分の SDRAM を RAM Disk として使用しています。RAM Disk の容量を変更は SysMemMap.h の内容を変更する必要があります。まず、RAM Disk として使用する全体のバイト数 1 を、DISK_ALL_BYTE で指定します。次に RAM Disk として使用する領域の 始まりと終わりを RAM_DISK_S、RAM_DISK_E 2 で指定します。

- 【注】
1. 1.5M バイト以上の値を指定してください。FAT 情報などで領域を消費するため、PC から見える容量は若干減少します。本サンプルプログラムでは、約 16M バイトまでを FAT12、約 2G バイトまでを FAT16 として FAT 情報を構成します。その他の FAT システムの FAT 情報はお客様で用意していただく必要があります。
 2. RAM_DISK_S から RAM_DISK_E で指定する領域は DISK_ALL_BYTE で指定するサイズ以上必要です。

5. サンプルプログラム概要

この章ではサンプルプログラムの特長やその構成について説明します。本サンプルプログラムは SH7216 CPU ボード上で動作し、また、SH7216 CPU ボードが RAM Disk として動作します。USB 転送は USB ファンクションモジュールからの割り込みによって開始します。SH7216 内蔵モジュールの割り込みのうち、USB ファンクションモジュールに関連する割り込みは、USIO, USI1, USBRXIO, USBTXIO, USBRXI1, USBTXI1 の 6 種類ですが、本サンプルでは USIO, USI1 のみ使用しています。

5.1 状態遷移図

図 5 に、本サンプルプログラムの状態遷移図を示します。本サンプルプログラムは、図 5 のように 3 つの状態に遷移します。

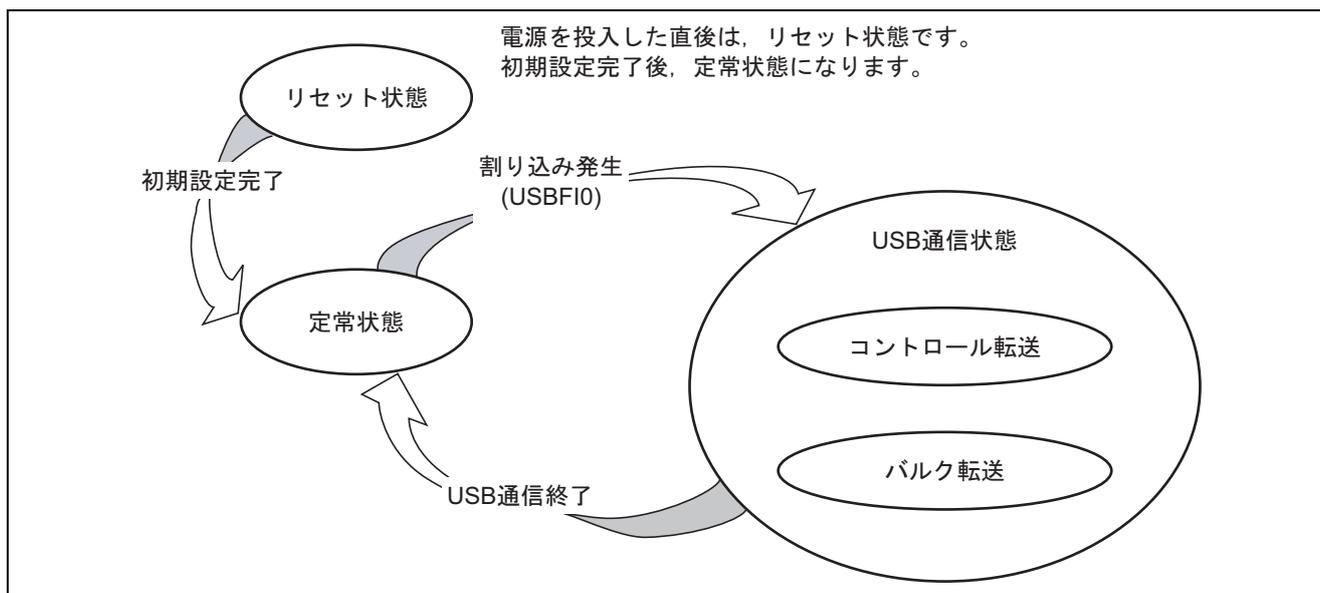


図 5 状態遷移図

- リセット状態
パワーオンリセット・マニュアルリセットの際には、この状態になります。リセット状態では、主に SH7216 の初期設定を行います。
- 定常状態
初期設定が完了すると、メインループで定常状態となります。
- USB 通信状態
定常状態において、USB モジュールから割り込みが発生するとこの状態に遷移します。USB 通信状態では、割り込みの種類に応じた転送方式によるデータ転送を行います。本サンプルプログラムで使用する割り込みは割り込みフラグレジスタ 0, 1, 2, 3, 4 (USBIFR0, 1, 2, 3, 4) によって示されます。割り込み要因が発生すると、USBIFR0, 1, 2, 3, 4 の対応するビットに 1 がセットされます。

5.2 USB 通信状態

USB 通信状態は、転送方式ごとに2つの状態に分類することができます(図6参照)。割り込みが発生すると、まず USB 通信状態へと遷移し、さらに割り込みの種類に応じて各転送状態へ分岐します。

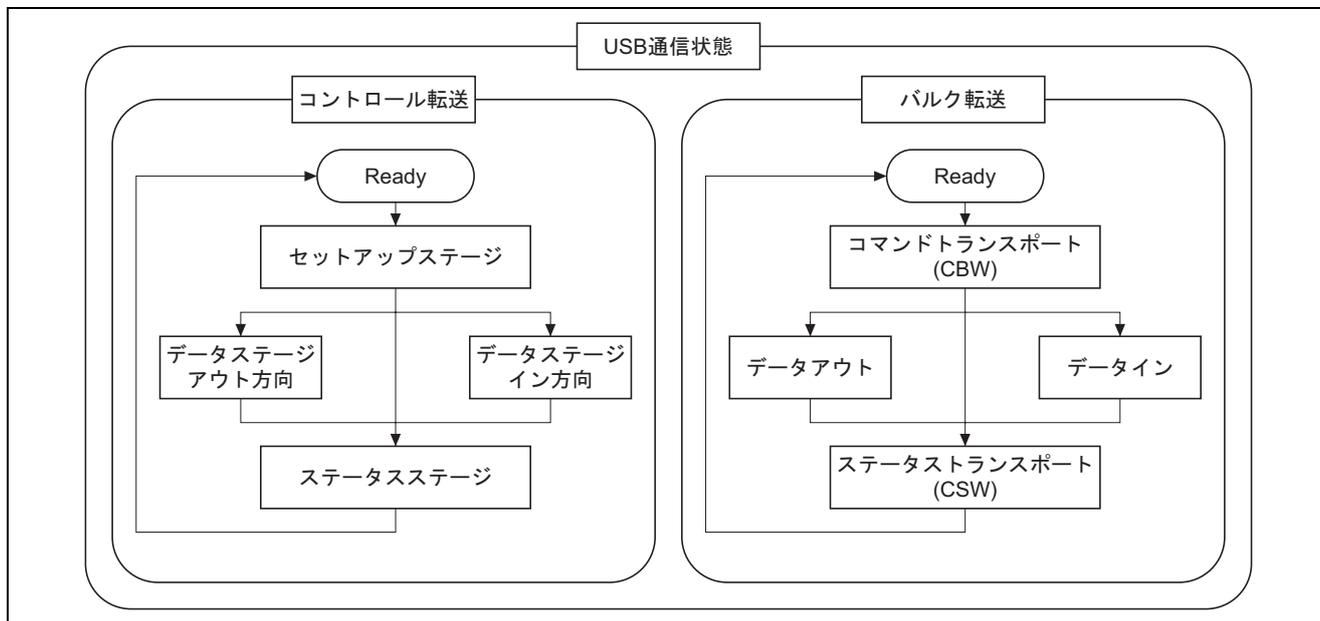


図6 USB 通信状態

5.2.1 コントロール転送について

コントロール転送は主に、デバイス情報の取得、デバイスの動作状態を設定する際などに使用されます。そのため、ホスト PC にファンクションを接続した際、最初に行われる転送でもあります。

コントロール転送の一連の転送処理は、2 または 3 つのステージから構成されます。コントロール転送のステージは、「セットアップステージ」「データステージ」「ステータスステージ」に分類することができます。

5.2.2 バルク転送について

バルク転送は時間的制約がない大量のデータを、エラーなく転送する場合に使用します。データの転送速度は保証されませんが、データの内容は保証されます。USB マスストレージクラス (Bulk-Only Transport) ではバルク転送を使用し、ホスト PC とファンクション間でストレージデータを転送します。

USB マスストレージクラス (Bulk-Only Transport) の一連の転送処理 (リードやライトなど) は、2 または 3 つのステージから構成されます。USB マスストレージクラス (Bulk-Only Transport) のステージは「コマンドトランスポート (CBW)」「データトランスポート」「ステータストランスポート (CSW)」に分類することができます。

5.3 ファイル構成

本サンプルプログラムは、9個のソースファイルと11個のヘッダファイルで構成されています。全構成ファイルを表8に示します。各関数は、転送方式または機能ごとに1つのファイルにまとめてあります。

表8 ファイル構成

ファイル名	主な機能
Startup.c	USB ファンクションの初期設定
UsbMain.c	割り込み要因の判定 パケットの送受信
DoRequest.c	ホスト PC が発行するセットアップコマンドの処理
DoRequestBOT_StorageClass.c	USB マスストレージクラス (Bulk-Only Transport) クラスコマンドの処理
DoControl.c	コントロール転送を実行
DoBulk.c	バルク転送を実行
DoBOTMSClass.c	USB マスストレージクラス (Bulk-Only Transport) のトランスポートを実行
DoSCSICommand.c	SCSI コマンドの解析および処理
sct.src	変数の初期値等の ROM から RAM への転送
CatBOTTypedef.h	Bulk-Only Transport 用構造体定義
CatProType.h	プロトタイプ宣言
CatSCSITypedef.h	SCSI 用構造体定義, FAT 情報構成のためのマクロ定義
CatTypedef.h	USB ファームウェアで使用する基本の構造体定義
SetBOTInfo.h	Bulk-Only Transport 対応に必要な変数の初期設定
SetMacro.h	マクロ定義
SetSCSIInfo.h	SCSI コマンド対応に必要な変数の初期設定
SetSystemSwitch.h	システムの動作設定
SetUsbInfo.h	USB ファームウェアで使用する変数の初期設定
SysMemMap.h	メモリマップのアドレス定義
iodef.h	SH7216 のレジスタ定義

5.4 関数の機能

各ファイルに含まれる関数とその機能を表9～表16に示します。

- Startup.c

パワーオンリセット, またはマニュアルリセットの際には, Startup.c の SetPowerOnSection が呼び出されます。ここでは SH7216 の初期設定や, バルク転送に使用する RAM 領域のクリアを行います。

表9 Startup.c

格納ファイル	関数名	機能
Startup.c	SetPowerOnSection	モジュールおよびメモリの初期化を行い, メインループへ移行
	InitSDRAM	SH7216 CPU ボードに搭載されている SDRAM の初期設定を行う
	_INIT_SCT	初期値がある変数を, RAM のワークエリアにコピーする
	InitMemory	バルク通信で使用する RAM 領域をクリアする
	InitSystem	USB バスのプルアップ制御
	Set_EPInfoR	エンドポイント情報の書き込みを行う

- UsbMain.c

UsbMain.c では、主に USB 割り込みフラグレジスタによって割り込み要因を判定し、割り込みの種類に応じた関数の呼び出しを行います。また、ホストコントローラとファンクションモジュール間におけるパケットの送受信を行います。

表 10 UsbMain.c

格納ファイル	関数名	機能
UsbMain.c	BranchOfInt0	バスリセットおよびエンドポイント 0 の割り込み要因の判定と、割り込みに応じた関数を呼び出す
	BranchOfInt1	エンドポイント 1 から 9 までの割り込み要因の判定と、割り込みに応じた関数を呼び出す
	GetPacket	ホストコントローラから転送されたデータを、RAM に書き込む
	GetPacket4	ホストコントローラから転送されたデータを、ロングワードサイズで RAM に書き込む (リングバッファ対応版) (USB マスストレージクラスでは使用しません)
	GetPacket4S	ホストコントローラから転送されたデータを、ロングワードサイズで RAM に書き込む (リングバッファ非対応, 高速版)
	PutPacket	ホストコントローラに転送するデータを USB モジュールに書き込む
	PutPacket4	ホストコントローラに転送するデータを、ロングワードサイズで USB モジュールに書き込む (リングバッファ対応版) (USB マスストレージクラスでは使用しません)
	PutPacket4S	ホストコントローラに転送するデータを、ロングワードサイズで USB モジュールに書き込む (リングバッファ非対応, 高速版)
	SetControlOutContents	ホストから送られたデータに書き換える
	SetUsbModule	USB モジュールの初期設定
	ActBusReset	バスリセット受信時に FIFO のクリア等を行う
	ActBusVcc	USB ケーブル接続割り込み動作を行う (本サンプルアプリケーションでは使用しません)
	ConvRealn	指定した番地から指定バイト長のデータを読み出す
	ConvReflexn	指定した番地から指定バイト長のデータを逆順に読み出す

- DoRequest.c

コントロール転送時に、ホストコントローラから送られてくるコマンドをデコードし、コマンドに応じた処理を行います。本サンプルプログラムでは、ベンダ ID の値に H'045B (ベンダ: ルネサス) を使用しています。お客様にて製品を開発される場合は「USB Implementers Forum」にてお客様のベンダ ID を取得願います。また、ベンダコマンドは使用していないため、DecVenderCommands では何も行っていません。ベンダコマンドを使用する際には、お客様でプログラムを作成願います。

表 11 DoRequest.c

格納ファイル	関数名	機能
DoRequest.c	DecStandardCommands	ホストコントローラが発行したコマンドをデコードし、そのうち標準コマンドの対応を行う
	DecVenderCommands	ベンダコマンドの対応を行う

- DoRequestBOT_StorageClass.c

USB マスストレージクラス (Bulk-Only Transport) コマンド (Bulk-Only Mass Storage Reset と Get Max LUN) に応じた処理を行います。

Bulk-Only Mass Storage Reset コマンドは Bulk-Only Transport で使用しているすべてのインタフェースをリセットします。

Get Max LUN コマンドは周辺装置が使用する最大の論理ユニット番号を返答します。当サンプルシステムの場合、論理ユニットは 1 つなので返答値は 0 をホストに返答します。

表 12 DoRequestBOT_StorageClass.c

格納ファイル	関数名	機能
DoRequestBOT_StorageClass.c	DecBOTClassCommands	USB マスストレージクラス (Bulk-Only Transport) コマンドの対応を行う

- DoControl.c

コントロール転送の割り込み (SETUP TS) が入ると、ActControl がコマンドを取得し、

DecStandardCommands でデコードを行いコマンドの転送方向を判別します。その後、コントロール転送の割り込み (EP0o TS, EP0i TR, EP0i TS) が発生すると ActControlInOut がコマンドの転送方向により、ActControlIn または ActControlOut を呼び出しデータステージとステータスステージを行います。

表 13 DoControl.c

格納ファイル	関数名	機能
DoControl.c	ActControl	コントロール転送のセットアップステージの制御を行う
	ActControlIn	コントロールイン転送 (データステージがイン方向の転送) のデータステージとステータスステージの制御を行う
	ActControlOut	コントロールアウト転送 (データステージがアウト方向の転送) のデータステージとステータスステージの制御を行う
	ActControlInOut	コントロール転送のデータステージとステータスステージを ActControlIn と ActControlOut に振り分ける

- DoBulk.c

バルク転送に関する処理を行います。USB マスストレージクラス (Bulk-Only Transport) では ActBulkInReady を使用しません。

表 14 DoBulk.c

格納ファイル	関数名	機能
DoBulk.c	ActBulkOut	バルクアウト転送を行う
	ActBulkIn	バルクイン転送を行う
	ActBulkInReady	バルクイン転送の準備を行う

- DoBOTMSClass.c

DoBOTMSClass.c では、USB マスストレージクラス (Bulk-Only Transport) の 2 ないし 3 つのステージ制御と仕様に従った動作を行います。

表 15 DoBOTMSClass.c

格納ファイル	関数名	機能
DoBOTMSClass.c	ActBulkOnly	Bulk-Only Transport のステージ別に振り分けを行う
	ActBulkOnlyCommand	Bulk-Only Transport の CBW の制御を行う
	ActBulkOnlyIn	(データステージがイン方向の転送) のデータトランスポートとステータストランスポートの制御を行う
	ActBulkOnlyOut	(データステージがアウト方向の転送) のデータトランスポートとステータストランスポートの制御を行う

- DoSCSICommand.c

ホスト PC から送られてきた SCSI コマンドを解析し、次のデータトランスポートまたはステータストランスポートの準備を行います。

表 16 DoSCSICommand.c

格納ファイル	関数名	機能
DoSCSICommand.c	DecBotCmd	ホストから Bulk-Only Transport で送られる SCSI コマンドの対応を行う
	SetBotCmdErr	SCSI コマンドのエラー時の処理を行う

5.5 RAM Disk について

本サンプルプログラムでは SH7216 CPU ボード上の SDRAM を Disk 装置に見立て、ホスト PC に対し SH7216 CPU ボード (ファンクション) は Disk であると報告しています。

ファンクションの Disk 装置には図 7 に示すようにマスタブートブロックと、パーティションブートブロックが存在しています。システム立ち上げ時に初期化ルーチンを用いて SDRAM 上の RAM Disk 領域にマスタブートブロックと、パーティションブートブロックを書き込みます。

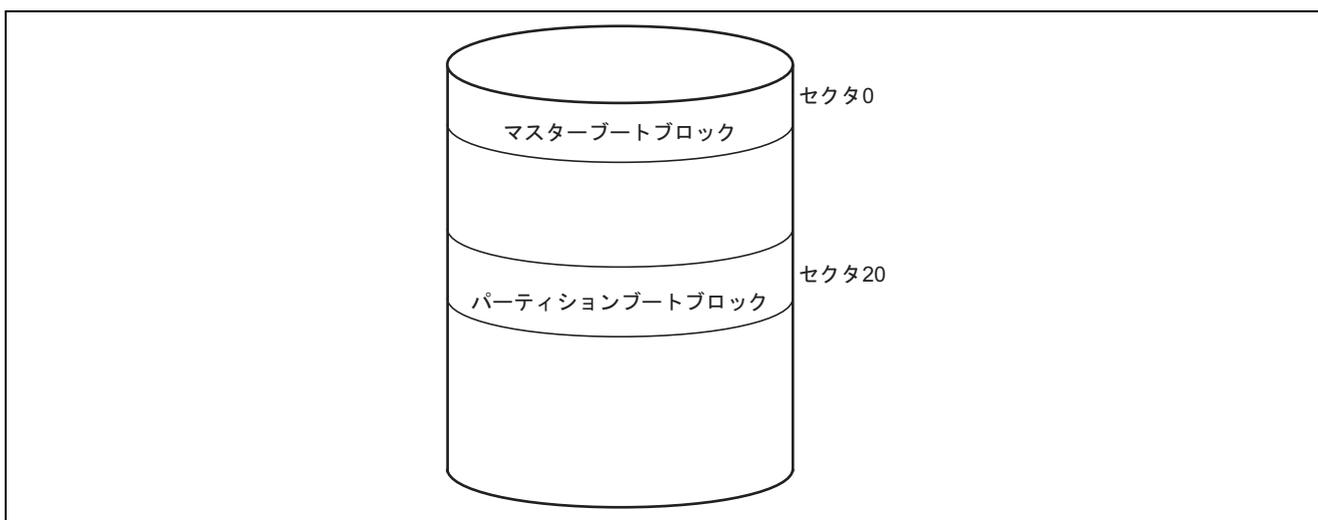


図 7 Disk の構造

ホスト PC からファンクションに対するアクセス (データの保存, 読み出し) は SCSI コマンドを使用します。SCSI コマンドの動作を行う場合、図 7 の構造を理解し動作を書く必要があります。

5.6 サポートする SCSI コマンドの動作について

本サンプルプログラムがサポートする SCSI コマンドの動作を表 17 に示します。

表 17 SCSI コマンド動作表

コマンド名	トランスポート名	動作内容
INQUIRY	CBW	コマンドをデコードし、INQUIRY コマンドである事を認識後、ROM に格納してある INQUIRY 情報 (96 バイト) の送信準備を行います。
	データ	ホスト PC に対し INQUIRY 情報をバルクイン転送にて送信します。
	CSW	ホスト PC に対しコマンド実行結果を送信します。送信データが 96 バイト以下であれば正常終了を送信します。
READ CAPACITY	CBW	コマンドをデコードし、READ CAPACITY コマンドである事を認識後、SDRAM 上に展開してある Disk 装置の 1 セクタ当りのバイト数とディスクの総セクタ数からなる READ CAPACITY 情報 (8 バイト) の送信準備を行います。メディアがアクセス不能状態の場合 (unit_state[0]の最下位ビットが 1) ファンクションはデータ転送なしとして扱い、"5.7 エラー時の処理(4)"に従います。また、REQUEST SENSE で返信する値を NOT READY (用意ができていない) に設定します。
	データ	ホスト PC に対し READ CAPACITY 情報をバルクイン転送にて送信します。メディアがアクセス不能状態の場合、ホストが要求したデータと同量のデータ (H'00) 返信します。
	CSW	ホスト PC に対しコマンド実行結果を送信します。メディアがアクセス不能状態の場合、コマンドフェイル (CSW ステータス H'01) を返信します。
READ (10)	CBW	コマンドをデコードし、READ (10)コマンドである事を認識後、SDRAM 上に展開してある Disk 装置内の指定された読み出しセクタから、指定セクタ量のデータ送信準備を行います。メディアがアクセス不能状態の場合 (unit_state[0]の最下位ビットが 1) ファンクションはデータ転送なしとして扱い、"5.7 エラー時の処理(4)"に従います。また、REQUEST SENSE で返信する値を NOT READY (用意ができていない) に設定します。
	データ	ホスト PC に対し読み出しセクタのデータをバルクイン転送にて送信します。メディアがアクセス不能状態の場合、ホストが要求したデータと同量のデータ (H'00) 返信します。
	CSW	ホスト PC に対し READ (10)コマンド実行結果を送信します。メディアがアクセス不能状態の場合、コマンドフェイル (CSW ステータス H'01) を返信します。
WRITE (10)	CBW	コマンドをデコードし、WRITE (10)コマンドである事を認識後、SDRAM 上に展開してある Disk 装置内の指定された書き込みセクタから、指定セクタ量のデータ受信準備を行います。メディアがアクセス不能状態の場合 (unit_state[0]の最下位ビットが 1) ファンクションはデータ転送なしとして扱い"5.7 エラー時の処理(9)"に従います。また、REQUEST SENSE で返信する値を NOT READY (用意ができていない) に設定します。
	データ	ホスト PC から書き込みセクタのデータをバルクアウト転送にて受信します。メディアがアクセス不能状態の場合、ホストから送られたデータを空読みします。
	CSW	ホスト PC に対し正常終了を送信します。メディアがアクセス不能状態の場合、コマンドフェイル (CSW ステータス H'01) を返信します。

コマンド名	トランスポート名	動作内容
REQUEST SENSE	CBW	コマンドをデコードし、REQUEST SENSE コマンドである事を認識後、返答値 (直前の SCSI コマンドを実行した結果) の送信準備を行います。
	データ	ホスト PC に対し返答値をバルクイン転送にて送信します。
	CSW	ホスト PC に対し本コマンド実行結果を送信します。送信データが 18 バイト数以下であれば正常終了を送信します。
PREVENT ALLOW MEDIUM REMOVAL	CBW	コマンドをデコードし、PREVENT ALLOW MEDIUM REMOVAL コマンドである事を認識後、ホスト PC に対し正常終了の送信準備を行います。メディアがアクセス不能状態の場合 (unit_state[0]の最下位ビットが 1) コマンドをフェイルに設定し、REQUEST SENSE で返信する値を NOT READY (用意ができていない) に設定します。
	データ	本コマンドにデータトランスポートは存在しません。
	CSW	ホスト PC に対し正常終了を送信します。メディアがアクセス不能状態の場合、コマンドフェイル (CSW ステータス H'01) を返信します。
TEST UNITREADY	CBW	コマンドをデコードし、TEST UNIT READY コマンドである事を認識後、ホスト PC に対し正常終了の送信準備を行います。メディアがアクセス不能状態の場合 (unit_state[0]の最下位ビットが 1) コマンドをフェイルに設定し、REQUEST SENSE で返信する値を NOT READY (用意ができていない) に設定します。
	データ	本コマンドにデータトランスポートは存在しません。
	CSW	ホスト PC に対し正常終了を送信します。メディアがアクセス不能状態の場合、コマンドフェイル (CSW ステータス H'01) を返信します。
VERIFY (10)	CBW	コマンドをデコードし、VERIFY (10)コマンドである事を認識後、ホスト PC に対し正常終了の送信準備を行います。メディアがアクセス不能状態の場合 (unit_state[0]の最下位ビットが 1) コマンドをフェイルに設定し、REQUEST SENSE で返信する値を NOT READY (用意ができていない) に設定します。
	データ	本コマンドにデータトランスポートは存在しません。
	CSW	ホスト PC に対し正常終了を送信します。メディアがアクセス不能状態の場合、コマンドフェイル (CSW ステータス H'01) を返信します。
STOP/START UNIT	CBW	コマンドをデコードし、STOP/START UNIT であることを認識後、コマンドがメディアの取り出し、もしくは停止を指定していた時にはグローバル変数 unit_state[0]の最下位ビットを 1 にセットします。その他の場合にはグローバル変数 unit_state[0]の最下位ビットを 0 にセットします。ユーザがアクセス不能状態から復帰させたい場合には unit_state[0]の最下位ビットを 0 にしてください。
	データ	本コマンドにデータトランスポートは存在しません。
	CSW	ホスト PC に対し正常終了を送信します。
MODESENSE (6)	CBW	コマンドをデコードし、MODE SENSE (6)コマンドである事を認識後、要求された MODE SENSE 情報の送信準備を行います。
	データ	ホスト PC に対し MODE SENSE 情報をバルクイン転送にて送信します。
	CSW	ホスト PC に対しコマンド実行結果を送信します。

コマンド名	トランスポート名	動作内容
READ FORMAT CAPACITY	CBW	コマンドをデコードし、READ FORMAT CAPACITY コマンドであることを認識後、SDRAM 上に展開してある Disk 装置がフォーマット済みであるものとして 1 セクタ当りのバイト数やディスクの総セクタ数などからなる READ FORMAT CAPACITY 情報 (20 バイト) の送信準備を行います。メディアがアクセス不能状態の場合 (unit_state[0]の最下位ビットが 1) ファンクションはデータ転送なしとして扱い、"5.7 エラー時の処理 (4)"に従います。また、REQUEST SENSE で返信する値を NOT READY (用意ができていない) に設定します。
	データ	ホスト PC に対し READ FORMAT CAPACITY 情報をバルクイン転送にて送信します。メディアがアクセス不能状態の場合、ホストが要求したデータと同量のデータ (H'00) 返信します。
	CSW	ホスト PC に対しコマンド実行結果を送信します。メディアがアクセス不能状態の場合、コマンドフェイル (CSW ステータス H'01) を返信します。
未サポートコマンド	CBW	コマンドをデコードし、未サポートコマンドであれば、REQUEST SENSE の返答値に INVALID FIELD IN CDB を設定後、データトランスポートの準備を行います。
	データ	ホスト PC がバルクイン転送にてデータを要求した場合、ホストが要求した量と同量のデータ (H'00) を送信します。ホスト PC がバルクアウト転送にてデータを送信した場合、受信バイト数のカウントを行います。データトランスポートが無い場合、何も動作は行いません。
	CSW	ホスト PC に対しコマンドフェイル (CSW ステータス H'01) を返信します。

5.7 エラー時の処理について

USB マスストレージクラス (Bulk-Only Transport) の転送を行う際、ホスト PC とファンクション間で発生するエラーとエラー時のファンクション側の対応動作を示します。

Bulk-Only Transport の規格では次にあげるエラーケースが規定されています。

- CBW が有効でない場合。
- ホストの期待とファンクションが意図する動作 (SCSI コマンドで指定された動作) の相違 (10 ケース)。

以上の 2 種類があります。これ以外の状態については規格書には定められていません。

ホスト-ファンクション間のデータ転送については表 18 と表 19 に示す 13 種類の状態が存在します。このうち CASE (1) (6) (12) は正常な転送状態です。

表 18 ホスト-ファンクション間のデータ転送状態

		ホストは		
		データ転送なしを期待	ファンクションからのデータ受信を期待	ファンクションへのデータ送信を期待
ファンクションは	データ転送なしを意図	(1) $H_n = D_n$	(4) $H_i > D_n$	(9) $H_o > D_n$
	ホストへのデータ送信を意図	(2) $H_n < D_i$	(5) $H_i > D_i$	(10) $H_o < > D_i$
			(6) $H_i = D_i$	
			(7) $H_i < D_i$	
	ホストからのデータ受信を意図	(3) $H_n < D_o$	(8) $H_i < > D_o$	(11) $H_o > D_o$
				(12) $H_o = D_o$
(13) $H_o < D_o$				

表 19 ホスト-ファンクション間データ転送状態解説

CASE	ホスト-ファンクション間での関係
(1)	ホストはデータ転送なしを期待し、ファンクションもデータ転送なしを意図する場合
(2)	ホストはデータ転送なしを期待し、ファンクションはホストへのデータ送信を意図する場合
(3)	ホストはデータ転送なしを期待し、ファンクションはホストからのデータ受信を意図する場合
(4)	ホストはファンクションからのデータ受信を期待し、ファンクションはホストへのデータ転送なしを意図する場合
(5)	ホストが期待したファンクションからのデータ受信数より、ファンクションがホストへ送信するデータ数が少ない場合
(6)	ホストが期待したファンクションからのデータ受信数と、ファンクションがホストへ送信するデータ数が同じ場合
(7)	ホストが期待したファンクションからのデータ受信数より、ファンクションがホストへ送信するデータ数が多い場合
(8)	ホストはファンクションからのデータ受信を期待し、ファンクションはホストからのデータ受信を意図する場合
(9)	ホストはファンクションへのデータ送信を期待し、ファンクションはデータ転送なしを意図する場合
(10)	ホストはファンクションへのデータ送信を期待し、ファンクションはホストへのデータ送信を意図する場合
(11)	ホストが期待したファンクションへのデータ送信数より、ファンクションがホストから受信するデータ数が少ない場合
(12)	ホストが期待したファンクションへのデータ送信数と、ファンクションがホストから受信するデータ数が同じ場合
(13)	ホストが期待したファンクションへのデータ数より、ファンクションがホストから受信するデータ数が多い場合

表 20 に発生する可能性のあるエラー状況例を示します。

表 20 エラー状況例

CASE	エラー状況
(2)	ホストから READ コマンドが発行される際、USB のデータトランスポートで転送するデータ数が 0 で、SCSI コマンドで指定されたデータ数が 0 以外の場合
(3)	ホストから WRITE コマンドが発行される際、USB のデータトランスポートで転送するデータ数が 0 で、SCSI コマンドで指定されたデータ数が 0 以外の場合
(4)	ホストから READ コマンドが発行される際、USB のデータトランスポートで転送するデータ数が 0 で、SCSI コマンドで指定されたデータ数が 0 の場合
(5)	ホストから READ コマンドが発行される際、USB のデータトランスポートで転送するデータ数より、SCSI コマンドで指定されたデータ数が少ない場合
(7)	ホストから READ コマンドが発行される際、USB のデータトランスポートで転送するデータ数より、SCSI コマンドで指定されたデータ数が多い場合
(8)	ホストから WRITE コマンドが発行されたのに、ホストが USB のデータトランスポートでデータを要求する場合
(9)	ホストから WRITE コマンドが発行される際、USB のデータトランスポートで転送するデータ数が 0 以外で、SCSI コマンドで指定されたデータ数が 0 の場合
(10)	ホストから READ コマンドが発行されたのに、ホストが USB のデータトランスポートでデータを送ってくる場合
(11)	ホストから WRITE コマンドが発行される際、USB のデータトランスポートで転送するデータ数より、SCSI コマンドで指定されたデータ数が少ない場合
(13)	ホストから WRITE コマンドが発行される際、USB のデータトランスポートで転送するデータ数より、SCSI コマンドで指定されたデータ数が多い場合

エラー状況に対するファンクションの対応動作は表 21 のようになります。

表 21

CASE	エラー時におけるデータトランスポートでのファンクション対応動作
(2), (3)	<ul style="list-style-type: none"> CSW のステータスに H'02 を設定する。
(4), (5)	<ul style="list-style-type: none"> ファンクションは dCBWDataTransferLength で示されたデータ長になるようにデータを付加し、ホストにデータを送信する。 CSW の dCBWDataResidue にデータトランスポートで付加したデータ数を設定する。 CSW のステータスに H'00 を設定する。
(7), (8)	<ul style="list-style-type: none"> ファンクションは dCBWDataTransferLength で示されたデータ長まで、ホストにデータを送信する。 CSW のステータスに H'02 を設定する。
(9), (11)	<ul style="list-style-type: none"> ファンクションは dCBWDataTransferLength で示されたデータ長分、データを受信する。 データトランスポートで受信したデータ数とファンクションで処理したデータ数の差を CSW の dCBWDataResidue に設定する。 CSW のステータスに H'01 を設定する。
(10), (13)	<ul style="list-style-type: none"> ファンクションは dCBWDataTransferLength で示されたデータ長分、データを受信する。 CSW のステータスに H'02 を設定する。

データ転送時のエラー処理フローは、図 8、図 9、図 10 のようになります。

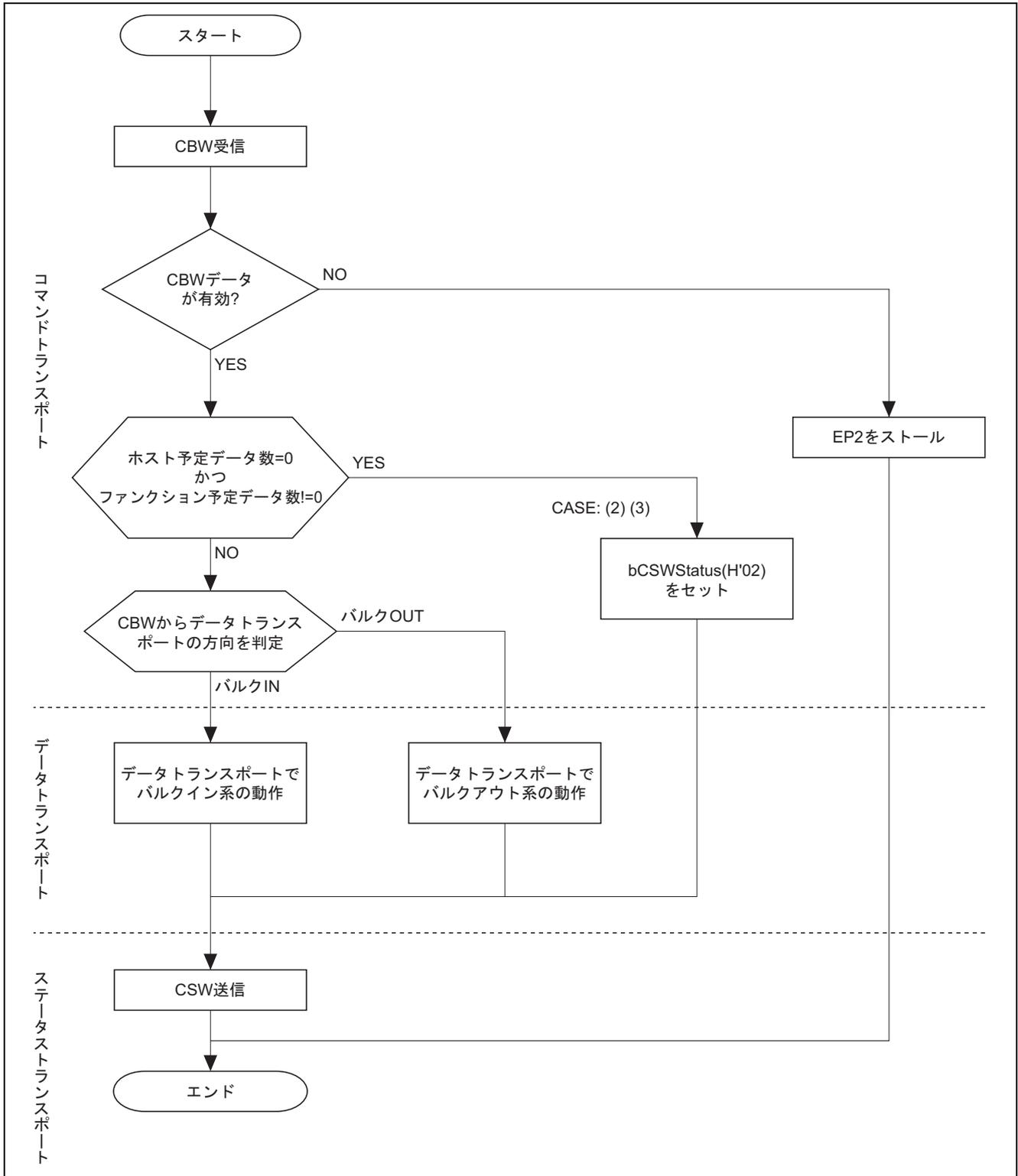


図 8 データ転送時のエラー処理フロー (1)

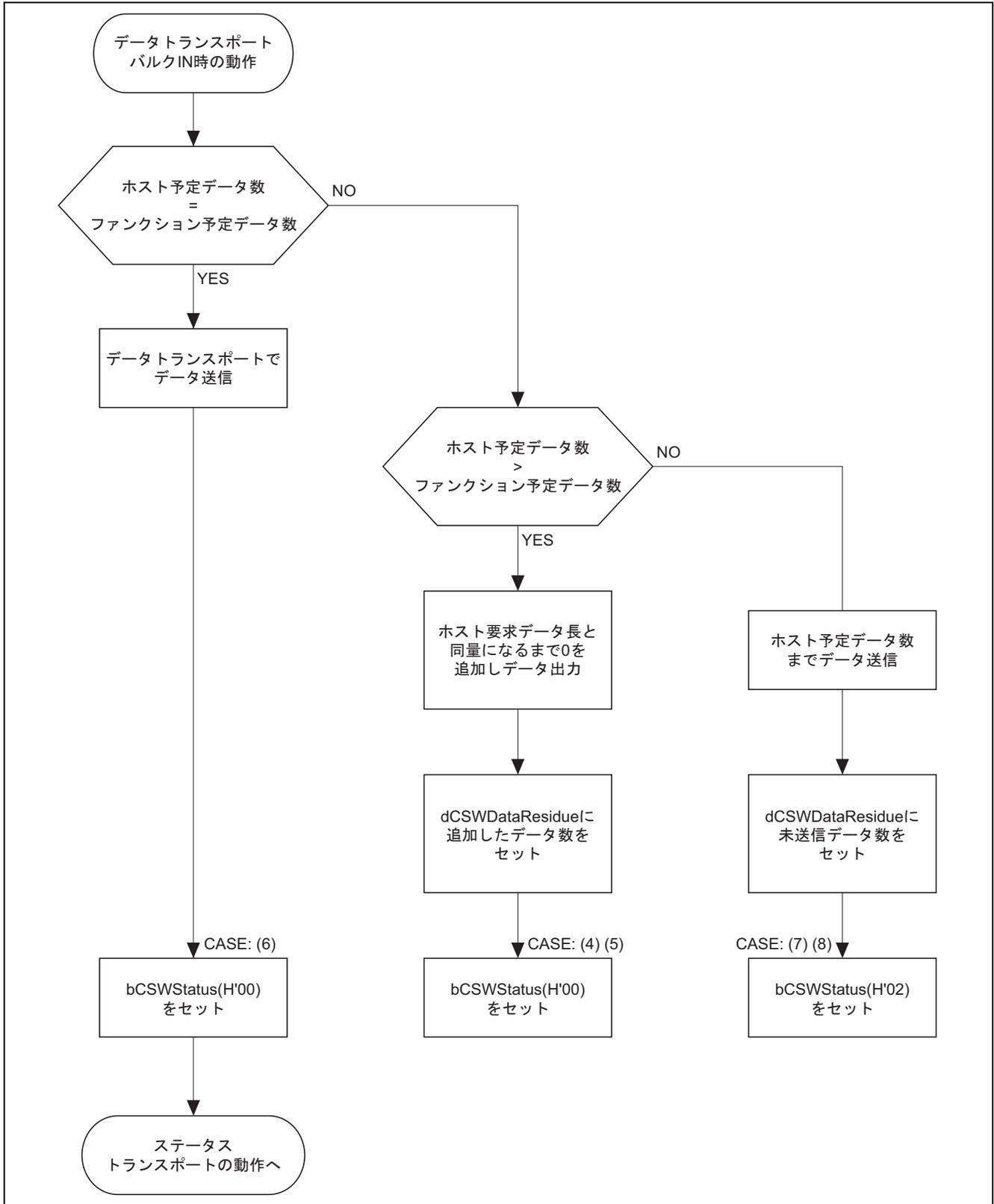


図9 データ転送時のエラー処理フロー (2)

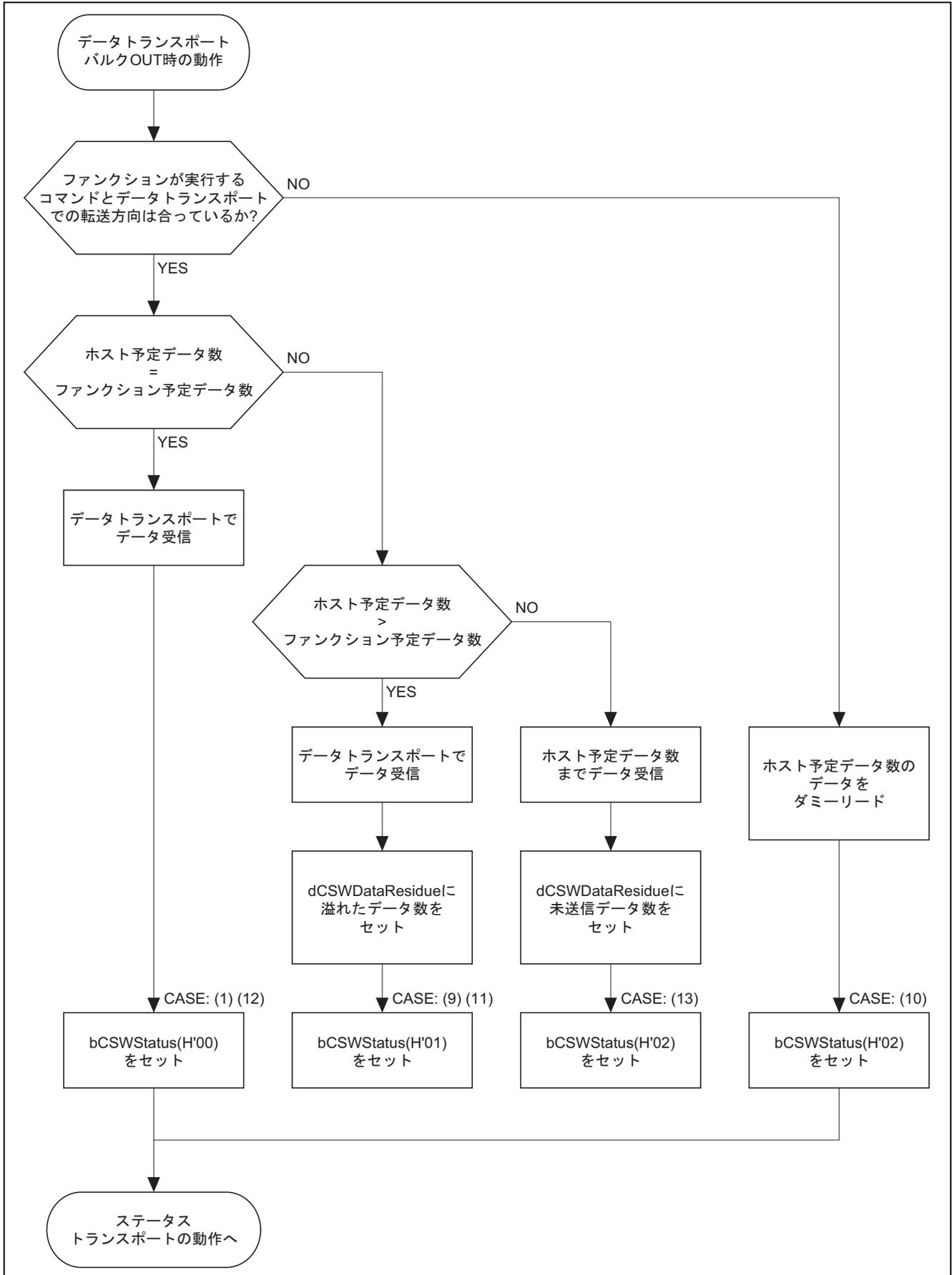


図 10 データ転送時のエラー処理フロー (3)

USB マスストレージクラス (Bulk-Only Transport) の転送を行う際、CBW トランスポートで一連のデータ転送が始まり、ホスト PC に CSW トランスポートで一連の転送結果 (ステータス) を返します。このためデータ転送処理を行う際に、CSW トランスポートで返答する内容も作成します。返答内容としては 2 項目あり、転送処理の結果を表す dCSWStatus と、データ転送エラーバイト数を表す dCSWDataResidue があります。

本サンプルプログラムでは、この 2 項目を作成するために、

- CBW パケットの dCBWDataTransferLength フィールド
- CSW パケットの dCSWDataTransferResidue フィールド

を使用します。

CBW パケットの dCBWDataTransferLength フィールドはホスト PC が指定するデータトランスポートで扱うデータバイト数を入れる変数として使用します。

CSW パケットの dCSWDataTransferResidue フィールドはファンクションがデータトランスポートで扱うデータバイト数を入れる変数として使用します。

CBW トランスポートが終了すると、dCBWDataTransferLength フィールドと dCSWDataTransferResidue フィールドにはデータトランスポートで扱う予定データバイト数がそれぞれ格納されます。

データトランスポートでデータ転送する際にはフロー図で示した流れで動作を行います。

ホスト-ファンクション間でエラーなく処理が行われるときは、データトランスポートでデータ転送する度に dCBWDataTransferLength フィールドと dCSWDataTransferResidue フィールドの値を転送バイト数分減算します。それ以外の場合は、PC が要求するデータトランスポートで扱うデータバイト数とファンクションがデータトランスポートで扱ったデータバイト数の「差」を、CSW パケットの dCSWDataTransferResidue フィールドに設定し、ステータストランスポートに移行します。

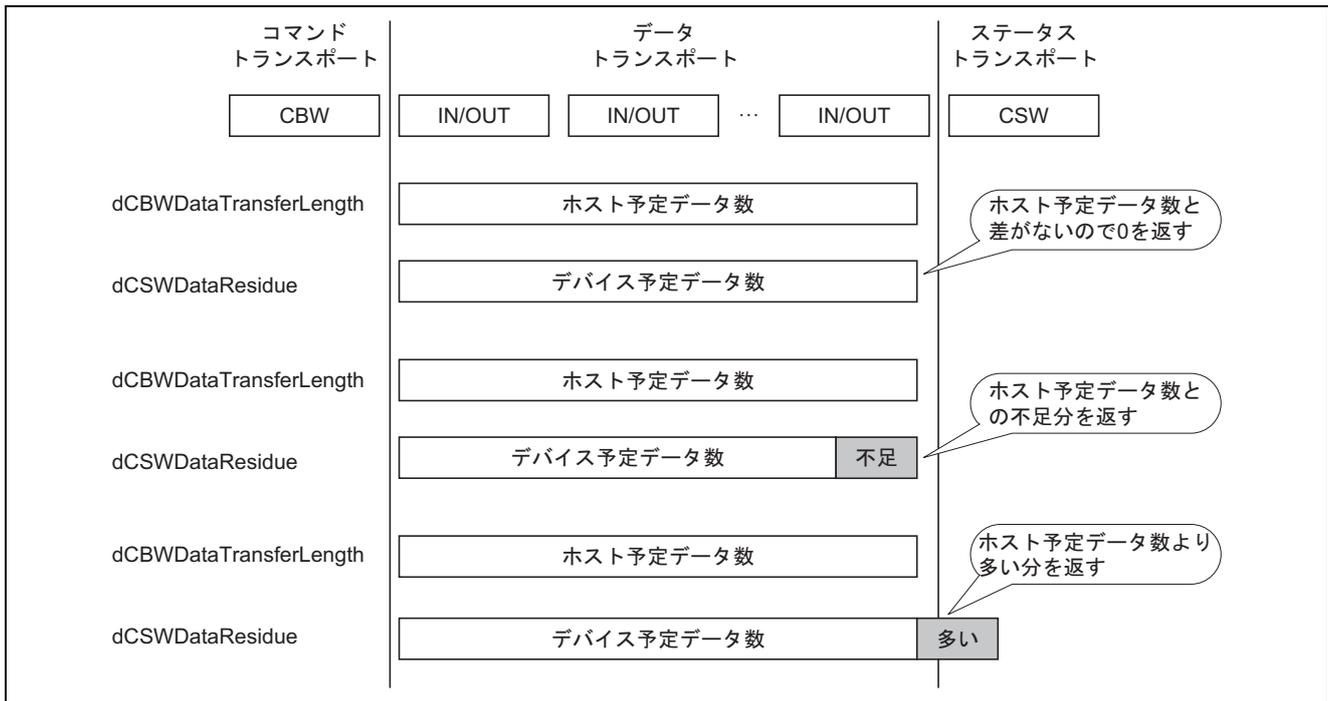


図 11 Bulk-Only Transport における各ステージ

6. 参考ドキュメント

- ソフトウェアマニュアル
 - (1) SH-2A, SH2A-FPU ソフトウェアマニュアル (RJJ09B0086)
(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
 - (2) SH7216 グループ ハードウェアマニュアル (RJJ09B0575)
(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)
- USB 規格関連
 - (3) Universal Serial Bus Specification
 - (4) Universal Serial Bus Mass Storage Class Specification Overview
 - (5) Universal Serial Bus Mass Storage Class (Bulk-Only Transport)
 - USB 開発者向けホームページ
<http://www.usb.org/developers>

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	2009.07.15	—	初版発行
2.00	2010.02.10	2	動作周波数を修正 統合開発環境のバージョンを修正
		11	図 4 を修正
		12	4.3 の説明を修正

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事務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しては、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のですが、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。
 - 1 1. 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。
 - 1 2. 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。
 - 1 3. 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