

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。

1. 資料概要

1.1. 概要

本資料(以降アプリケーションノートと記載)は、ルネサス テクノロジ製Hi-Speed USB汎用ASSPデバイスM66591を制御するための応用技術資料です。

1.2. アプリケーションノートの特徴

アプリケーションノートは以下のような特徴を持っています。

- ・ M66591制御に関する手法を、記述例をあげて具体的に記載しています。
- ・ M66591を使用する場合の注意事項を記載しています。
- ・ 「ルネサス テクノロジ汎用 ASSP M66591 USB Sample Firmware」とリンクしています。
- ・ 豊富な機能用途を具体的に記載しています。

なお、「ルネサス テクノロジ汎用ASSP M66591 USB Sample Firmware」は一部のUSB特殊信号の制御には対応していません。また、本資料はM66591の制御方法の一例です。お客様の応用に合わせて変更する必要があります。

1.3. 関連ドキュメント

[1] M66591データシート

[2] アプリケーションノートM6659x共通編

【<http://www.renesas.com>】

[3] ルネサス テクノロジ汎用 ASSP M66591 USB Sample Firmware

【MAEC よりユーザに配布】

[4] USB の基礎

[5] USB2.0 の基礎

【近日公開予定】

[6] USB 仕様 Ver.1.1 「Chapter 8 Protocol Layer」

[7] USB 仕様 Ver.1.1 「Chapter 9 Device Framework」

【<http://www.renesas.com>】

[8] USB Revision2.0 Specification

【<http://www.usb.org/developers/docs.html>】

2. バスインタフェース

2.1. バスインタフェース

M66591は16bit セパレートバス/マルチプレクスバスの選択が可能です。M66591のバスモード選択は以下の設定で行ってください。

表2.1 セパレート/マルチプレクスバス設定

端子	設定	動作
MPBUS	“L” レベル入力	16bit セパレートバス
	“H” レベル入力	16bit マルチプレクスバス

※ 動作中にこの端子の切り替えはしないでください。

2.1.1. セパレートバス選択時の動作

- (1) データバス (D15-0) <入出力>
CPU からM66591のレジスタにアクセスするデータバス (D15-0) として動作します。
- (2) アドレスバス (A7-1) <入力>
CPU からM66591のレジスタにアクセスするアドレスバス (A7-1) として動作します。

セパレートバス選択時のタイミングは「M66591 データシート 4.9 章」を参照ください。

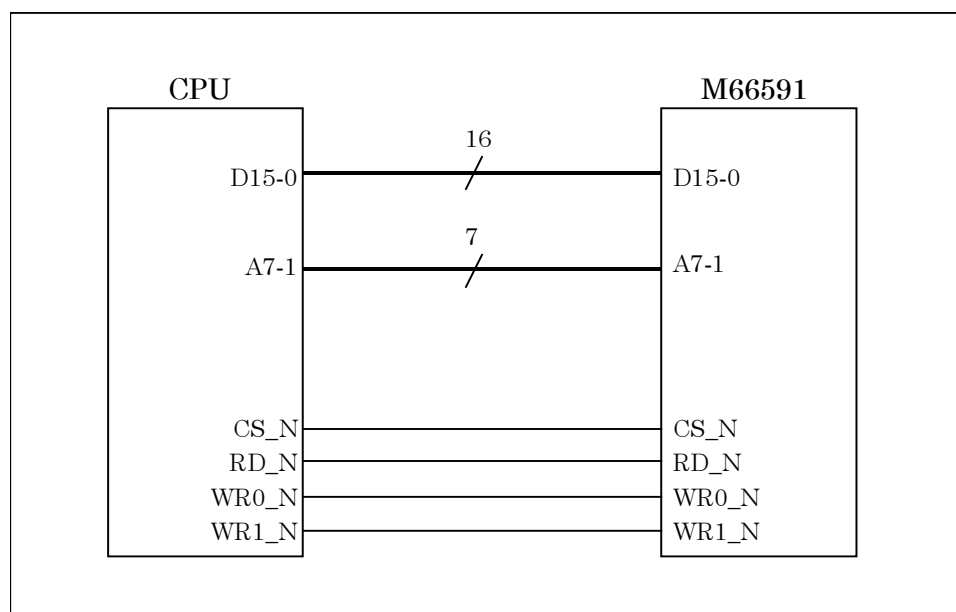


図2.1 セパレートバスでの接続例

2.1.2. マルチプレクスバス選択時の動作

- (1) データバス (D15-8、D7-1/AD7-1/D0) <入出力>
- D15-8 : CPU からM66591のレジスタにアクセスするデータバス (D15-8) として動作します。
- D7-1/AD7-1 : データ D7-0 の入出力と、アドレス AD7-1 の入力を時分割で行います。
アドレス入力時、D0 は未使用です。
- (3) アドレスバス (A7/ALE、A6-1) <入力>
- A7/ALE : A7 は ALE 端子として動作し、立ち下がりエッジでアドレスをラッチします。
- A6-1 : 使用しません。

マルチプレクスバス選択時のタイミングは「M66591 データシート 4.9 章」を参照ください。

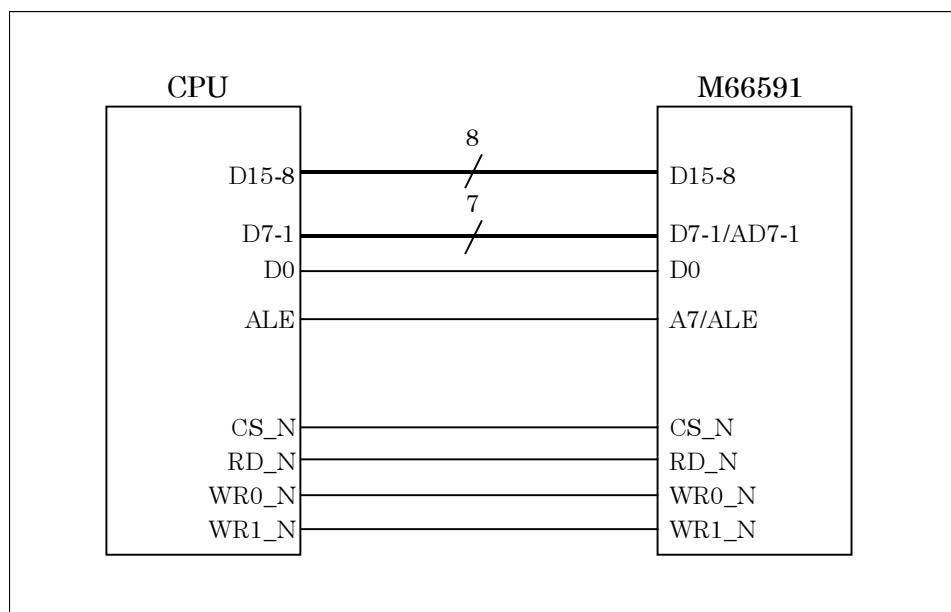


図2.2 マルチプレクスバスでの接続例

3. 接続例

3.1. CPU との接続方法(16bit バス幅)

接続される CPU からのアクセスについて考える場合、CPU のエンディアンが問題になります。接続が想定される CPU のエンディアンを表 3.1 に列挙します。

表3.1 接続が想定されるCPUとそのエンディアン

No.	ビット エンディアン	バイト エンディアン	CPU 例
1	リトル	リトル	7700 シリーズ、7900 シリーズ、M16C ファミリなど
2	リトル	ビッグ	SH シリーズ、H8S シリーズ
3	ビック	リトル	(一般的ではないので掲載省略)
4	ビック	ビック	M32R ファミリ

この章では、16bit バス幅での各 CPU と M66591 との結線方法、及びアクセス方法について説明します。M66591 へのアクセスは、レジスタへのアクセス、FIFO ポートへのアクセスに大別され、FIFO ポートへのアクセスには、CPU によるアクセスと DMA 転送によるアクセスがあります。また、奇数バイト数のデータを FIFO バッファに書きこむ場合の 1 バイトアクセスについての考察も述べます(奇数バイト数データの読み出しについては、アプリケーションノート M6659x 共通編 5.1 章 FIFO バッファを参照ください)。

各種エンディアンの CPU と M66591 の接続性を表 3.2 にまとめます。

表3.2 CPUのエンディアンによる接続性(16bit接続時)

エンディアン		接続方法	big_end 設定値* ₂	アクセス	
ビット	バイト			レジスタ* ₁	FIFO
リトル	リトル	3.1.1章の方法	'0'	ok	ok
リトル	ビッグ	3.1.2.1章の方法	'1'	ok	ok
		3.1.2.2章の方法	'0'	invert	ok
ビッグ	ビッグ	3.1.3.1章の方法	'1'	ok	ok
		3.1.3.2章の方法	'0'	invert	ok

*1 : ok : データをありのまま読み出せる、或いは書き込めることを示します。
invert : バイトが逆転して読み出せる、或いは書き込めることを示します。

*2 : 16bit アクセス時の設定値を示します(8bit アクセス時は必ず big_end='0' に設定してください)。

M66591 と CPU を結線するときに、データのバイト単位での逆転がなく、ありのままで転送できるかどうかは大きな問題です。M66591 では、プログラムを工夫することによりこの問題を解決することが可能です。

実際に設計されるときは、お客様の使い方にあわせて結線方法を選択してください。

3.1.1. ビットエンディアンがリトル、バイトエンディアンがリトルの CPU の場合(16bit バス幅)

ルネサス テクノロジーの 7700 シリーズ、7900 シリーズ、M16C ファミリーなどはこの種類の CPU です。
 <接続>

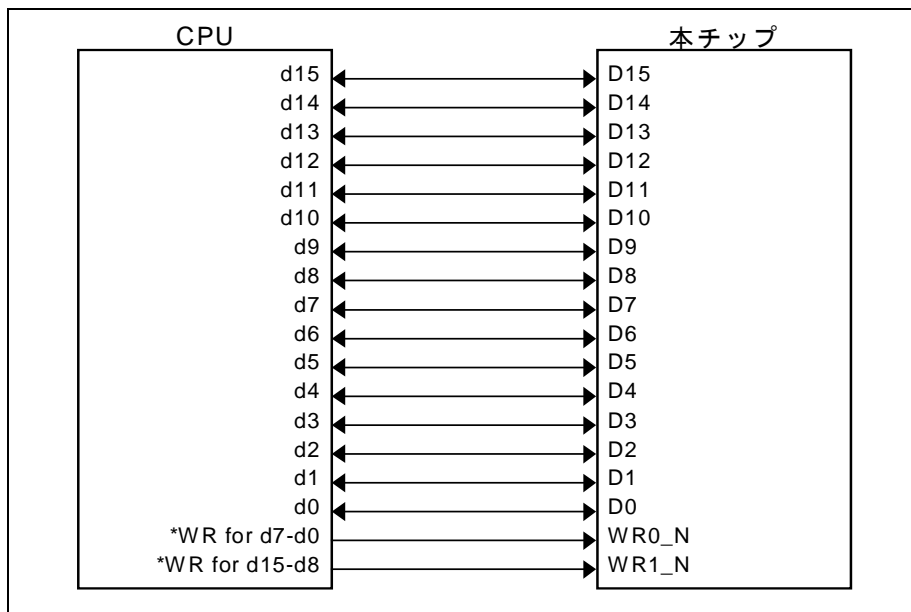


図3.1 ビットエンディアン:リトル、バイトエンディアン:リトルの接続例(16bit バス幅)

<レジスタアクセス(C_FIFO/D0_FIFO ポートレジスタ以外)>

読み出し: レジスタのデータがそのまま、CPU のレジスタ或いはメモリに読み出されます。

書き込み: CPU のレジスタ或いはメモリ上のデータがそのまま、レジスタに書き込まれます。

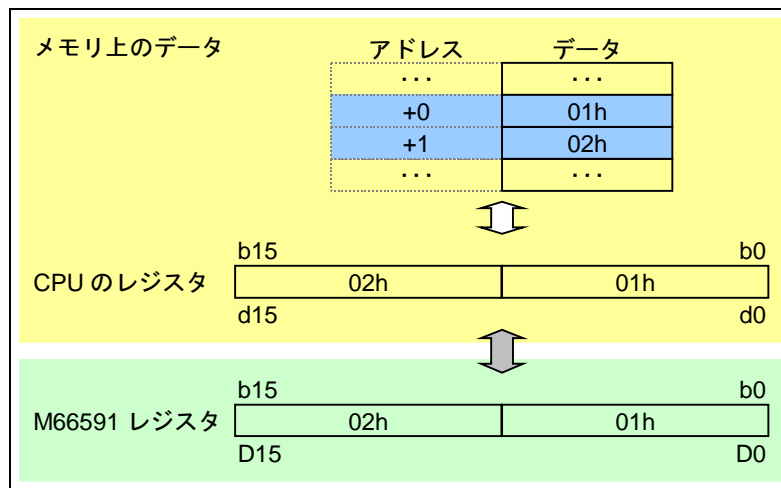


図3.2 ビットエンディアン:リトル、バイトエンディアン:リトルのレジスタアクセスイメージ(16bit バス幅)

<FIFO ポート(C_FIFO/D0_FIFO ポートレジスタ)アクセス>

(1)FIFO に対する CPU アクセス

基本的に、big_end(ビッグエンディアンモード;Address 0Ah, bit 8)=0、MBW(FIFO アクセス最大ビット幅;Address 28h,30h, bit 10)=1を設定してアクセスします。

奇数バイト数データの最後の 1 バイトのみ、big_end=0、MBW=0を設定し、CPU の d7-d0 に有効データを出力し、8 ビット単位で書き込みます。

読み出し: FIFO レジスタから読み出したデータがそのまま、順番にメモリに格納されます。

書き込み: メモリに格納されたデータがそのまま、順番に FIFO バッファに書き込まれます。

(2)FIFO に対する DMA アクセス

CPU アクセスと同様です。D0_FIFO ポートに対して設定してください。

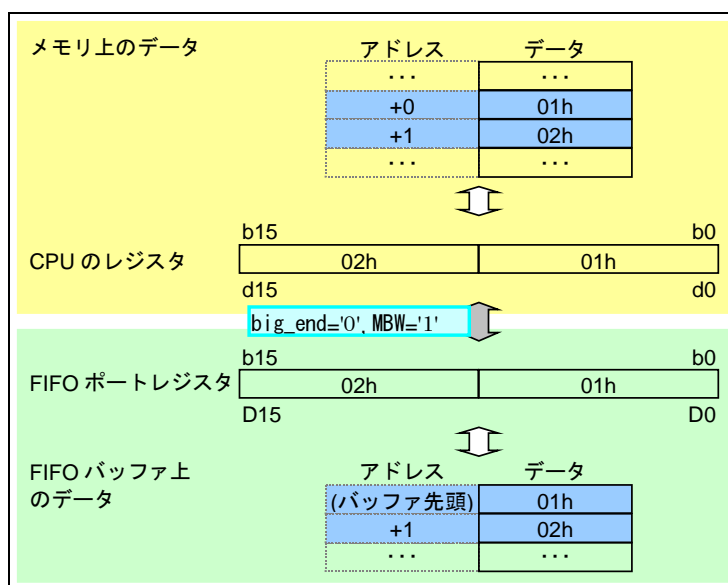


図3.3 ビットエンディアン：リトル、バイトエンディアン：リトルの FIFO ポートアクセスイメージ 1(16bit バス幅、16 ビットアクセス)

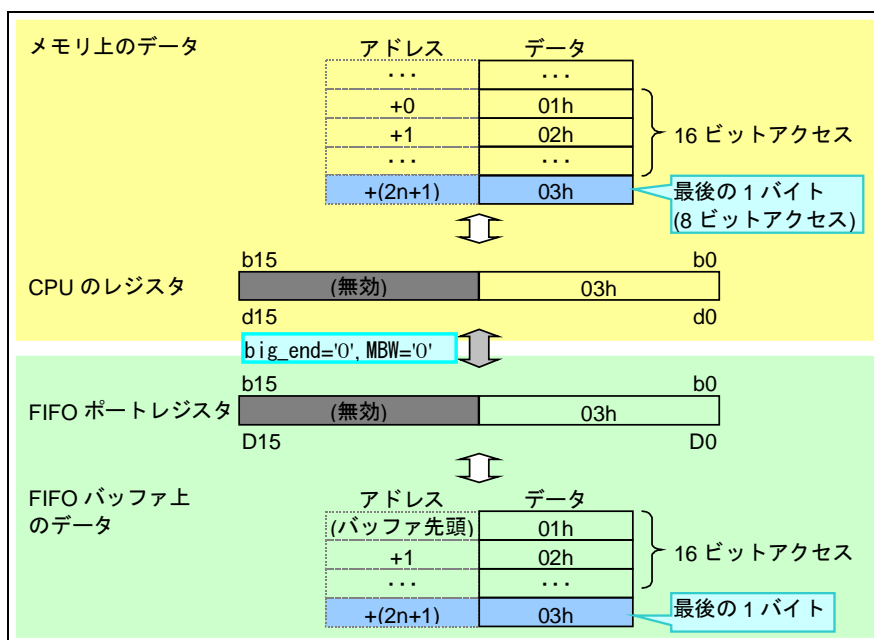


図3.4 ビットエンディアン：リトル、バイトエンディアン：リトルの FIFO ポートアクセス 2(16bit バス幅、バイト書き込み)

3.1.2. ビットエンディアンがリトル、バイトエンディアンがビッグの場合(16bit バス幅)

ルネサス テクノロジの SH シリーズ、H8S シリーズなどはこの種類の CPU です。

M66591 との接続を考える場合には、基本的には方法(1)(3.1.2.1章)を推奨します。参考資料として方法(2)(3.1.2.2章)を併記します。

方法(1)の場合、レジスタアクセスと FIFO アクセス (**big_end** 使用) ともデータを逆転する必要はありません。

方法(2)の場合、レジスタアクセスはデータ逆転が必要になります。FIFO アクセス (**big_end** 未使用) はデータを逆転する必要はありません。

3.1.2.1. ビットエンディアンがリトル、バイトエンディアンがビッグの場合(16bit バス幅) (1)

<接続>

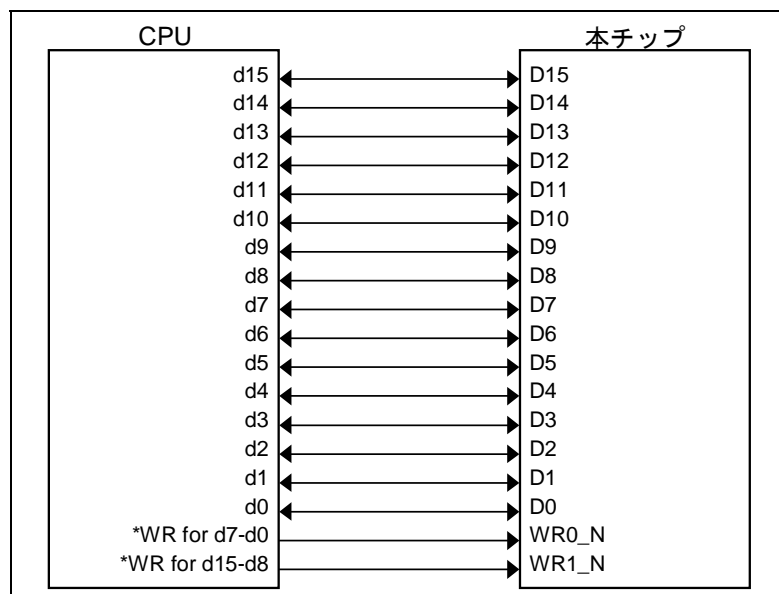


図3.5 ビットエンディアン：リトル、バイトエンディアン：ビッグの接続図 (16bit バス幅) (1)

<レジスタアクセス(C_FIFO/D0_FIFO ポートレジスタ以外)>

読み出し: レジスタのデータがそのまま、CPU のレジスタ或いはメモリに読み出されます。

書き込み: CPU のレジスタ或いはメモリ上のデータがそのまま、レジスタに書き込まれます。

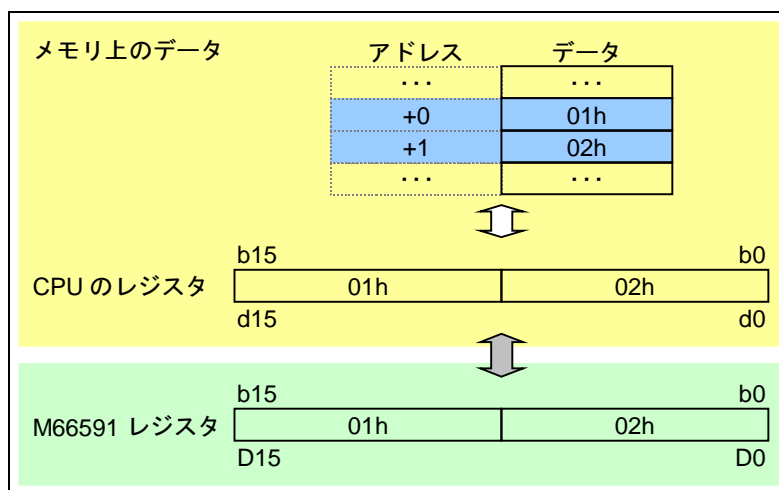


図3.6 ビットエンディアン：リトル、バイトエンディアン：ビッグのレジスタアクセスイメージ(16bit バス幅) (1)

<FIFO ポート(C_FIFO/D0_FIFO ポートレジスタ)アクセス>

(1)FIFO に対する CPU アクセス

基本的に **big_end**(ビッグエンディアンモード;Address 0Ah, bit 8)='1'、**MBW**(FIFO アクセス最大ビット幅; Address 28h,30h, bit 10)='1'を設定してアクセスします。**big_end**='1'の場合、FIFO ポート-FIFO バッファ間のデータ転送時にM66591内部でバイト単位で上位・下位を入れ替えます。即ち、読み出し時は FIFO バッファの下位バイトが D15-D8 に、上位バイトが D7-0 に読み出されます。書き込み時は、D15-D8 のデータが FIFO バッファの下位バイトに、D7-D0 のデータが上位バイトに書き込まれます。奇数バイト数データの最後の 1 バイトのみ、**big_end**='0'、かつ **MBW**='0'を設定し、CPU の D7-D0 に有効なデータを出し、8 ビット単位で書き込みます。

読み出し: FIFO バッファに格納された順番で、メモリに格納されます。

書き込み: メモリに格納された順番で、FIFO バッファに格納されます。

(2)FIFO に対する DMA アクセス

CPU アクセスと同様です。D0_FIFO ポートに対して設定してください。

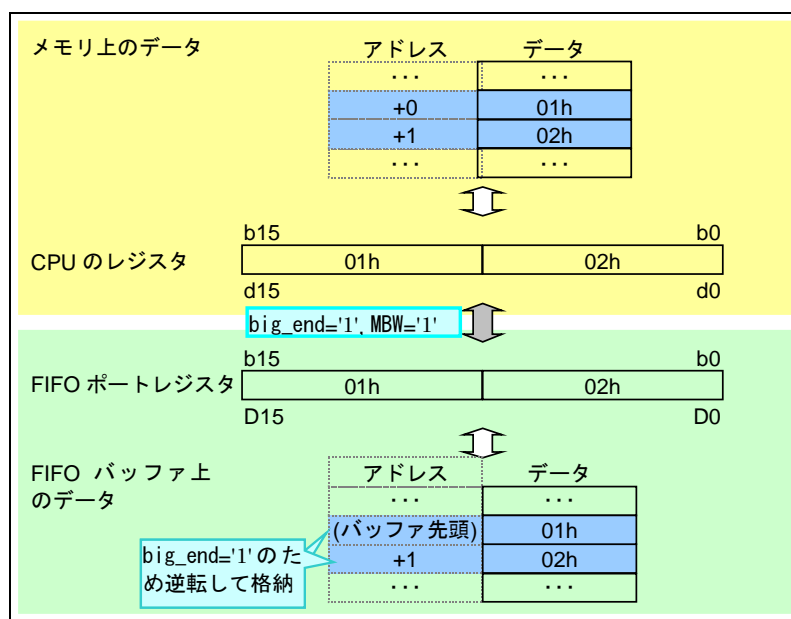


図3.7 ビットエンディアン：リトル、バイトエンディアン：ビッグ
の FIFO ポートアクセスイメージ(16bit バス幅) (1) (16 ビットアクセス)

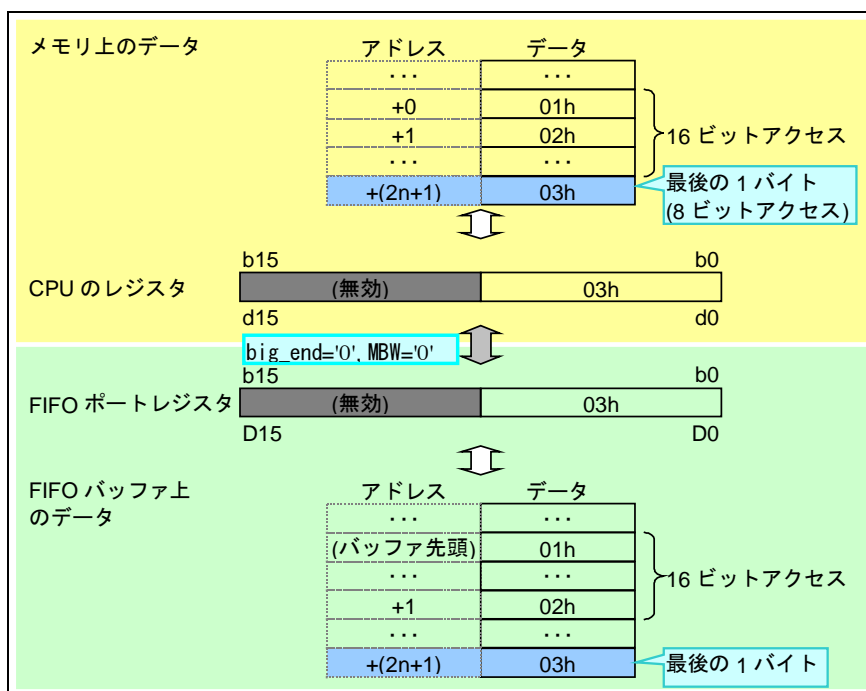


図3.8 ビットエンディアン：リトル、バイトエンディアン：ビッグ
の FIFO ポートアクセスイメージ(16bit バス幅) (1) (バイト書き込み)

3.1.2.2. ビットエンディアンがリトル、バイトエンディアンがビッグの場合(16bit バス幅) (2)

<接続>

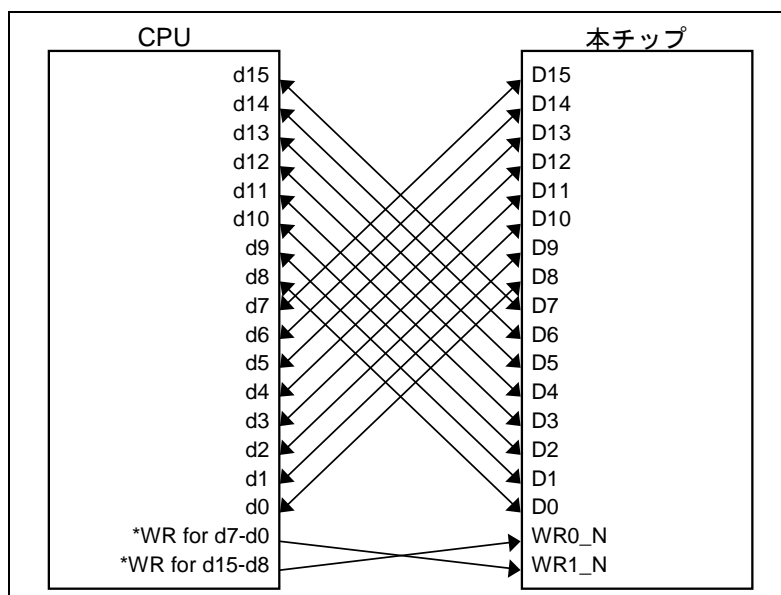


図3.9 ビットエンディアン：リトル、バイトエンディアン：ビッグの接続図
(16bit バス 幅) (2)

<レジスタアクセス(C_FIFO/D0_FIFO ポートレジスタ以外)>

読み出し: レジスタのデータがバイト単位で逆転して、CPU のレジスタ或いはメモリに読み出されます。

書き込み: CPU のレジスタ或いはメモリ上のデータがバイト単位で逆転して、レジスタに書き込まれます。

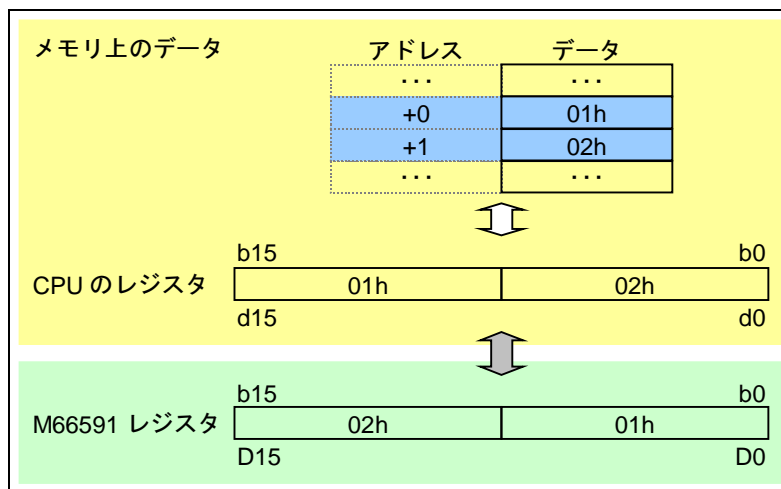


図3.10 ビットエンディアン：リトル、バイトエンディアン：ビッグ
のレジスタアクセスイメージ(16bit バス幅) (2)

<FIFO ポート(C_FIFO/D0_FIFO ポートレジスタ)アクセス>

(1)FIFO に対する CPU アクセス

基本的に `big_end`(ビッグエンディアンモード;Address 0Ah, bit 8)=0'、`MBW`(FIFO ポートアクセスビット幅;Address 28h,30h, bit 10)='1'の設定でアクセスします。奇数バイト数書き込み時の最後の1バイトのみ、`big_end`=0'、かつ `MBW`=0'を設定し、CPU の D15-D8 に有効なデータを出力し、8ビット単位で書き込みます。

読み出し: FIFO バッファに格納された順番で、メモリに格納されます。

書き込み: メモリに格納された順番で、FIFO バッファに格納されます。

(2)FIFO に対する DMA アクセス

CPU アクセスと同様です。D0_FIFO ポートに対して設定してください。

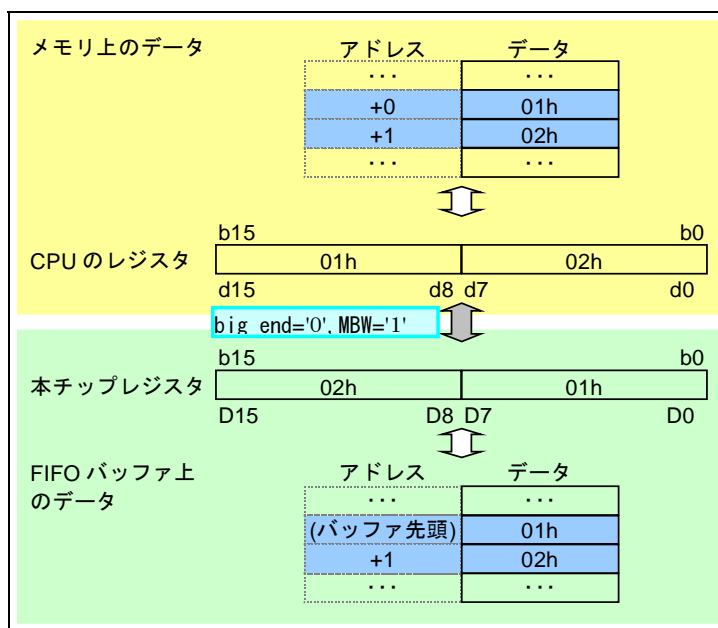


図3.11 ビットエンディアン：リトル、バイトエンディアン：ビッグ
の FIFO ポートアクセスイメージ(16bit バス幅) (2) (16 ビットアクセス)

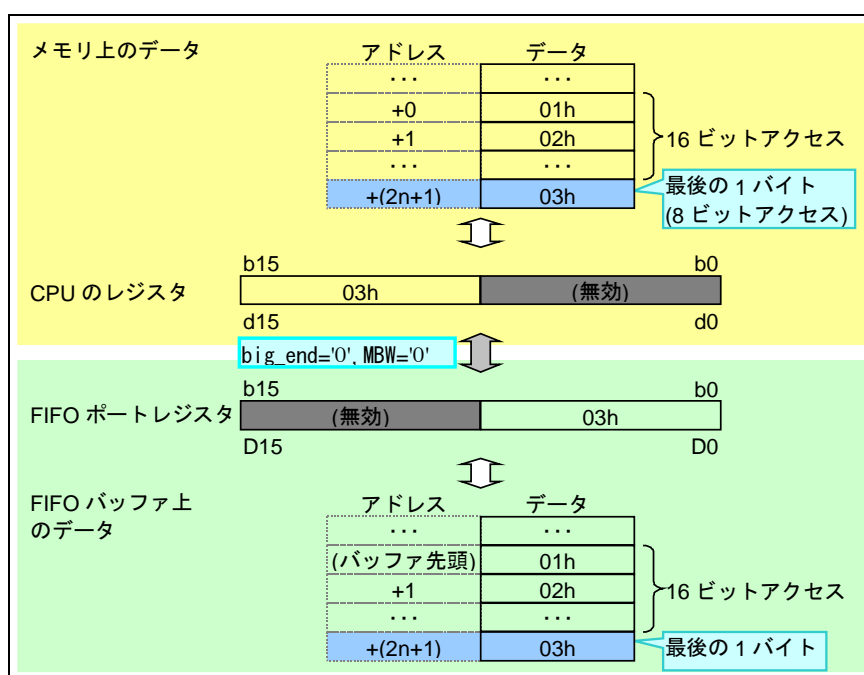


図3.12 ビットエンディアン：リトル、バイトエンディアン：ビッグ
の FIFO ポートアクセスイメージ(16bit バス幅) (2) (バイト書き込み)

3.1.3. ビットエンディアンがビッグ、バイトエンディアンがビッグの場合(16bit バス幅)

ルネサス テクノロジの M32R ファミリーなどはこの種類の CPU です。M66591 との接続を考える場合には、基本的には方法(1)(3.1.3.1章)の方法を推奨します。参考資料として方法(2)(3.1.3.2章)を併記します。

方法(1)の場合、レジスタアクセスと FIFO アクセス (big_end 使用) ともデータを逆転する必要はありません。

方法(2)の場合、レジスタアクセスはデータ逆転が必要になります。FIFO アクセス (big_end 未使用) はデータを逆転する必要はありません。

3.1.3.1. ビットエンディアンがビッグ、バイトエンディアンがビッグの場合(16bit バス幅) (1)

<接続>

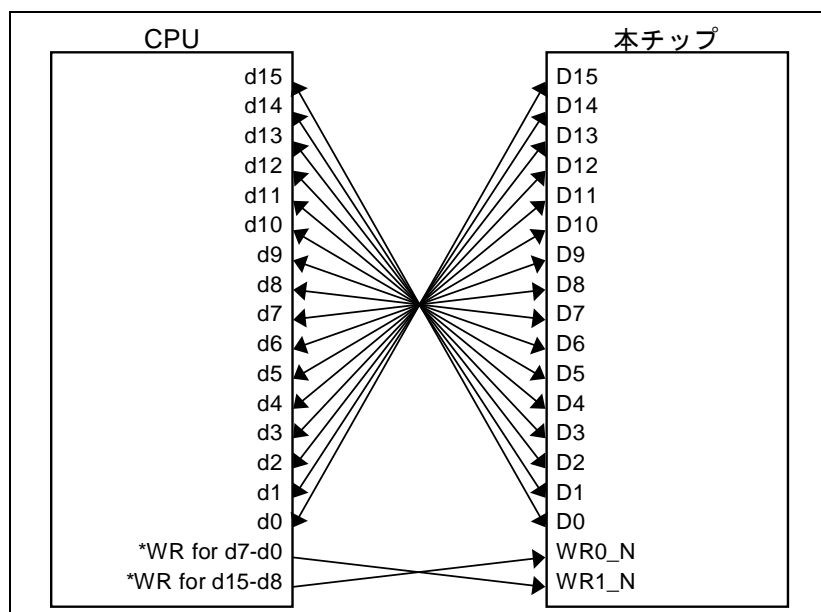


図3.13 ビットエンディアン：ビッグ、バイトエンディアン：ビッグの接続例 (16bit バス幅) (1)

<レジスタアクセス(C_FIFO/D0_FIFO ポートレジスタ以外)>

読み出し: レジスタのデータがそのまま、CPU のレジスタ或いはメモリに読み出されます。

書き込み: CPU のレジスタ或いはメモリ上のデータがそのまま、レジスタに書き込まれます。

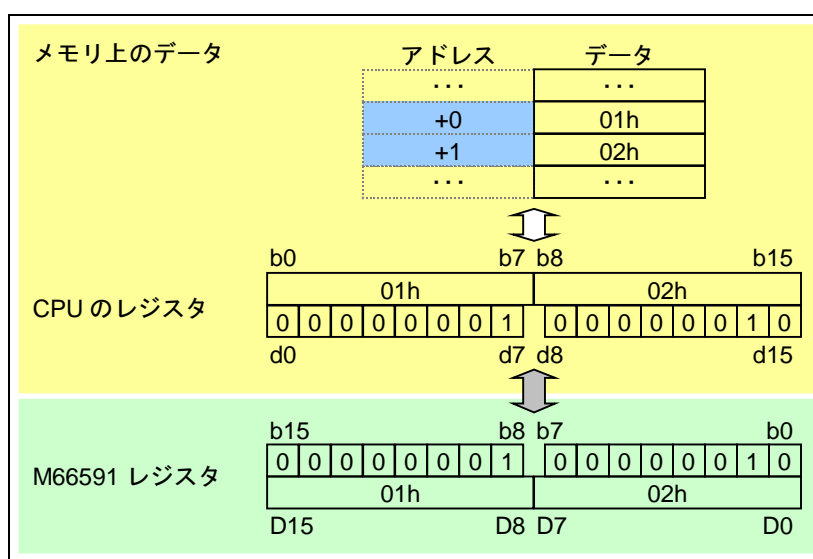


図3.14 ビットエンディアン：ビッグ、バイトエンディアン：ビッグのレジスタアクセスイメージ(16bit バス幅) (1)

<FIFO ポート(C_FIFO/D0_FIFO ポートレジスタ)アクセス>

(1)FIFO に対する CPU アクセス

基本的に **big_end**(ビッグエンディアンモード;Address 0Ah, bit 8)='1'、**MBW**(FIFO アクセス最大ビット幅;Address 28h,30h, bit 10)='1'の設定でアクセスします。**big_end**='1'の場合、FIFO ポート-FIFO バッファ間のデータ転送時にM66591内部でバイト単位で上位/下位を入れ替えます。即ち、読み出し時は FIFO バッファの下位バイトが D15-D8 に、上位バイトが D7-D0 に読み出されます。書き込み時は、D15-D8 のデータが FIFO バッファの下位バイトに、D7-D0 のデータが上位バイトに書き込まれます。奇数バイト数データの最後の 1 バイトのみ、**big_end**='0'、かつ **MBW**='0'を設定し、CPU の D15-D8 に有効なデータを出し、8 ビット単位で書き込みます。

読み出し: FIFO バッファに格納された順番で、メモリに格納されます。

書き込み: メモリに格納された順番で、FIFO バッファに格納されます。

(2)FIFO に対する DMA アクセス

CPU アクセスと同様です。D0_FIFO ポートに対して設定してください。

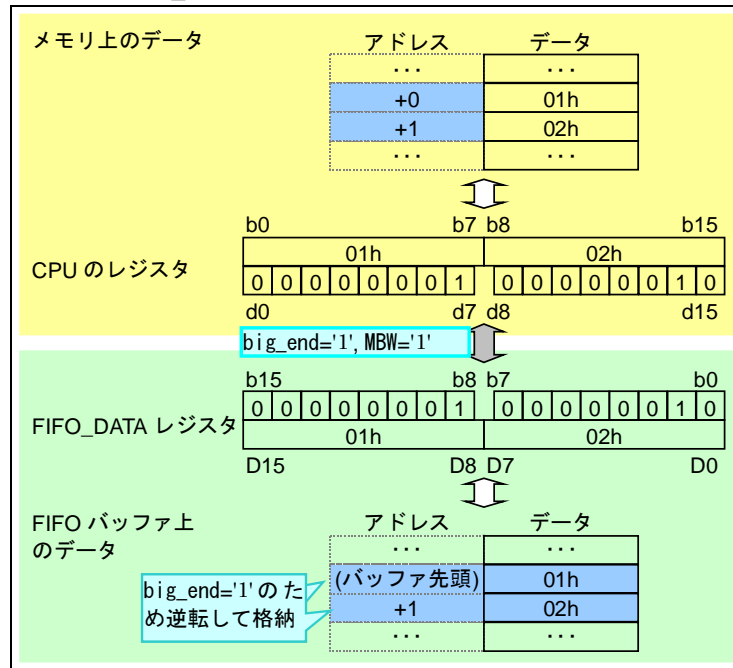


図3.15 ビットエンディアン：ビッグ、バイトエンディアン：ビッグの FIFO ポートアクセスイメージ(16bit バス幅) (1) (16 ビットアクセス)

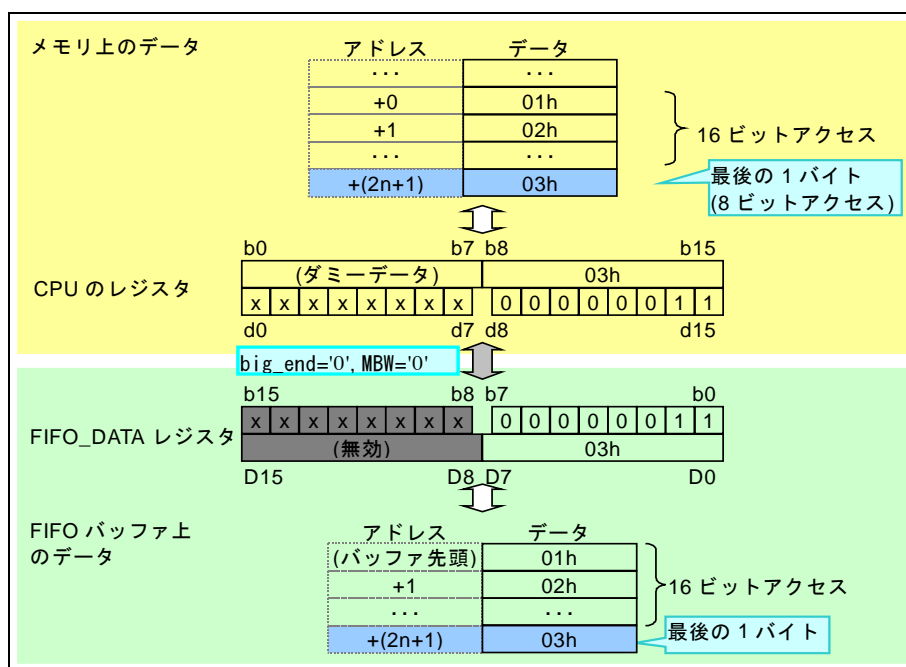
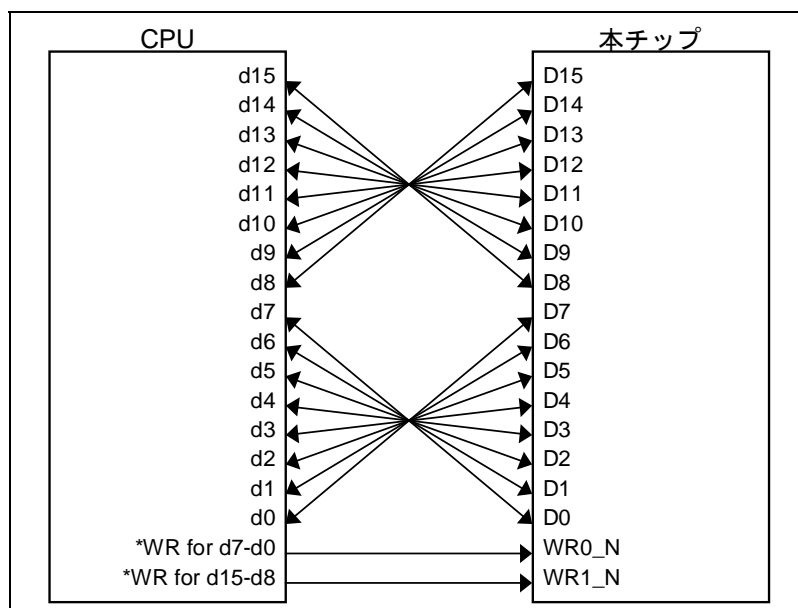


図3.16 ビットエンディアン：リトル、バイトエンディアン：ビッグの FIFO ポートアクセスイメージ(16bit バス幅) (1) (バイト書き込み)

3.1.3.2. ビットエンディアンがビッグ、バイトエンディアンがビッグの場合(16bit バス幅)(2)

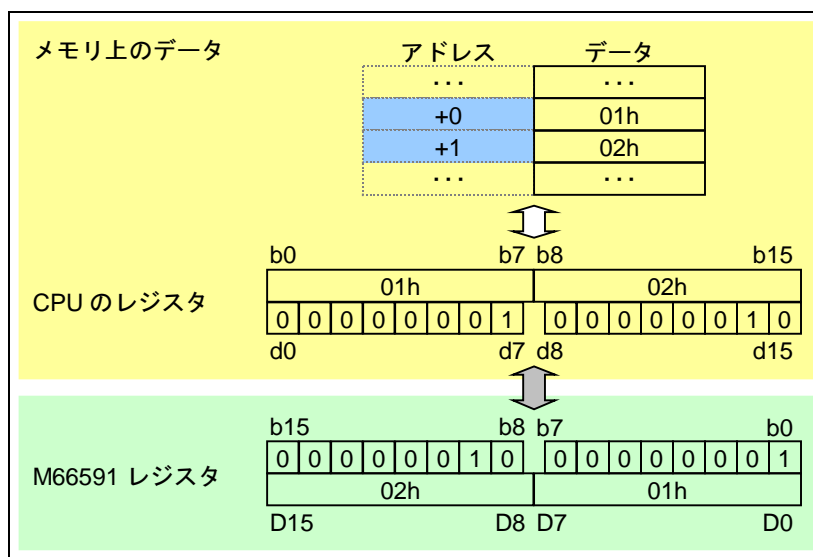
<接続>

図3.17 ビットエンディアン：ビッグ、バイトエンディアン：ビッグの接続例
(16bit バス幅) (2)

<レジスタアクセス(C_FIFO/D0_FIFO ポートレジスタ以外)>

読み出し: レジスタのデータがバイト単位で逆転して、CPU のレジスタ或いはメモリに読み出されます。

書き込み: CPU のレジスタ或いはメモリ上のデータがバイト単位で逆転して、レジスタに書き込まれます。

図3.18 ビットエンディアン：ビッグ、バイトエンディアン：ビッグ
のレジスタアクセスイメージ(16bit バス幅) (2)

<FIFO ポート(C_FIFO/D0_FIFO ポートレジスタ)アクセス>

(1)FIFO に対する CPU アクセス

基本的に big_end(ビッグエンディアンモード;Address 0Ah, bit 8)=0、MBW(FIFO アクセス最大ビット幅;Address 28h,30h, bit 10)=1の設定でアクセスします。奇数バイト数書き込み時の最後の1バイトのみ、big_end=0、かつ MBW=0を設定し、CPU の d0-d7 に有効なデータを出力し、8ビット単位で書き込みます。

読み出し: FIFO バッファに格納された順番で、メモリに格納されます。

書き込み: メモリに格納された順番で、FIFO バッファに格納されます。

(2)FIFO に対する DMA アクセス

CPU アクセスと同様です。D0_FIFO ポートに対して設定してください。

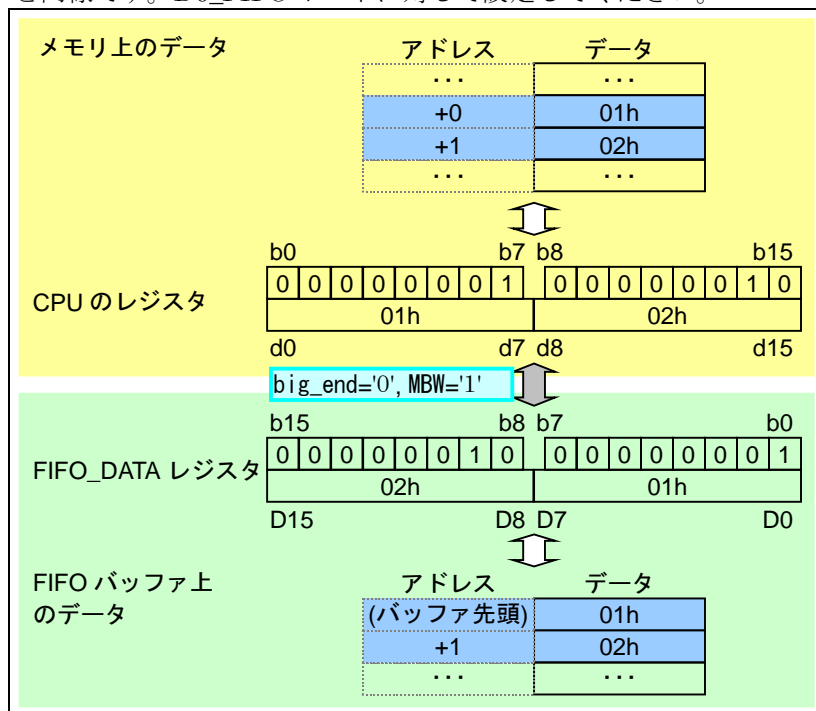


図3.19 ビットエンディアン：ビッグ、バイトエンディアン：ビッグの FIFO ポートアクセスイメージ(16bit バス幅) (2) (16 ビットアクセス)

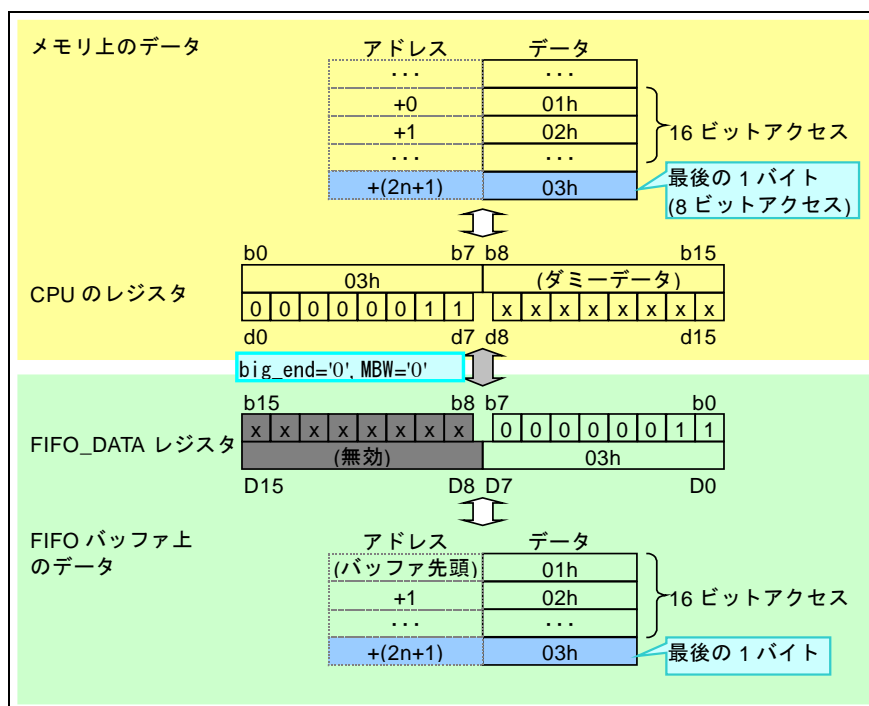


図3.20 ビットエンディアン：ビッグ、バイトエンディアン：ビッグの FIFO ポートアクセスイメージ(16bit バス幅) (2) (バイト書き込み)

4. 初期設定

USB 通信用の初期設定は、アプリケーションノート M6659x 共通編 2.1 概略フロー中「USB 初期設定」に示したフローとなります。

4.1. 初期設定

M66591の初期設定処理では、USB 動作を許可して VBUS 端子の入力を VBUSSTS ビットで確認してください。VBUS 端子の入力状態によって VBUS 割り込みを待つ処理または USB アタッチ処理を行ってください。

VBUS 端子の入力ステータスが'0'の場合には、ホストに接続されていない状態であるため VBUS 割り込みを許可し、ホストへの接続を待ってください。

VBUS 端子の入力ステータスが'1'の場合には、初期設定の前にすでにホストに接続されているためアタッチ処理を行ってください。

具体的な処理フローチャートを図 4.1に示します。

なお、VBUS 端子は USB コネクタの VBUS 端子からダイレクト入力しているため入力ステータスを確認するときにソフトウェアによるチャタリング除去処理が必要です。

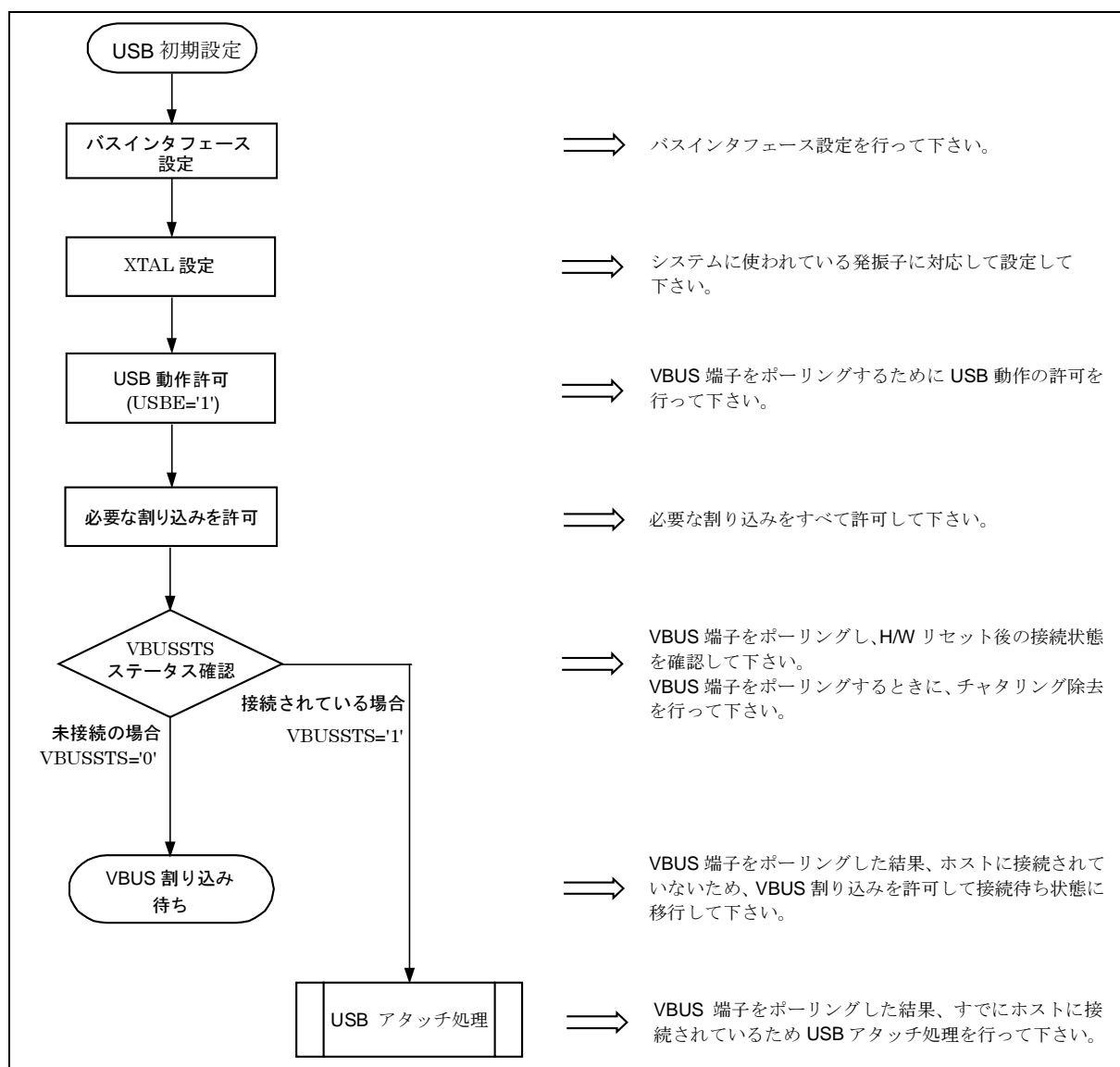


図4.1初期設定フローチャート

4.2. 接続／切断時の処理

M66591の初期設定処理では、VBUS 端子の状態によりアタッチ処理／デタッチ処理が必要です。詳しくは、アプリケーションノート M6659x 共通編 8.7 章 Vbus 変化の検出処理を参照ください。

4.3. バスインタフェース設定

4.3.1. 駆動電流の調整

M66591は、以下に示す出力端子の駆動電流を VIF の電圧に対してデータピン&FIFO/DMA 制御ピンコンフィギュレーションレジスタ 1 の LDRV ビットにより調整することが可能です。

- ・ D15-0 : データバス
- ・ INT : 割り込み端子
- ・ SUSP_ON : USB サスペンド信号出力端子
- ・ CONF_ON : USB Configured 出力信号端子
- ・ SD7-0 : スプリットデータバス
- ・ DREQ : DREQ 端子
- ・ DEND : DEND 端子

以下表 4.1に、駆動電流調整の設定を示します。

表4.1 駆動電流調整の設定

ビット	設定	
LDRV	'0'	VIF = 1.7-2.0V 時
	'1'	VIF = 2.7-3.6V 時

4.3.2. エンディアン設定

M66591は、接続される CPU に合わせて C_FIFO/D0_FIFO ポートにアクセスするエンディアンの設定が可能です。

以下表 4.2に、エンディアンの設定を、表 4.3に、エンディアンによる振る舞いを示します。

表4.2 エンディアンの設定

ビット	設定	
big_end	'0'	リトルエンディアン
	'1'	ビッグエンディアン

表4.3 エンディアンによる振る舞い

エンディアン	b15-b8	b7-b0
リトルエンディアン	奇数アドレス	偶数アドレス
ビッグエンディアン	偶数アドレス	奇数アドレス

4.3.3. 割り込み信号設定

M66591は、接続される CPU に合わせて割り込み信号の極性及び出力センスの設定が可能です。

以下表 4.4に、割り込み信号の極性／出力センスを示します。

割り込み信号のネゲート及びアサートタイミングは、M66591 データシートを参照ください。

表4.4 割り込み信号の極性／出力センス

ビット	設定	
INTA	'0'	Low アクティブ
	'1'	High アクティブ
INTL	'0'	エッジセンス
	'1'	レベルセンス

4.3.4. DMA 制御信号極性設定

M66591は、接続される CPU に合わせて DMA 制御信号の極性設定が可能です。
以下表 4.5に、DMA 制御信号の極性設定を示します。

表4.5 DMA制御信号の極性設定

ビット	設定	
DreqA	'0'	Low アクティブ
	'1'	High アクティブ
DackA	'0'	Low アクティブ
	'1'	High アクティブ
DendA	'0'	Low アクティブ
	'1'	High アクティブ

4.3.5. VBUS ステータスの確認時の注意

USB の初期設定で、すでにホストと接続されているか確認する場合 VBUS ステータスを確認しますが、VBUS 信号に「USB Revision2.0 Specification」で推奨されている "Flyback Voltage を防ぐ Bypass Capacitance" を搭載している場合、注意が必要です。

M66591の H/W 関連資料及び、評価基板 M3A-0037 には"1.0uF"のコンデンサを搭載しています。

M66591の電源が入っていない状態で、ホストと接続された場合、上記コンデンサにチャージされ、その後ホストと切断されてもコンデンサの電荷が抜けません。その後 M66591 の電源投入で M66591 内部の 500K Ω プルダウンにより約 2 秒後に電荷がディスチャージされるため、VBUSSTS を確認する場合は注意してください。

5. データ転送

5.1. データ転送の特徴

M66591は、USB 通信で規定されているコントロール、バルク、インタラプト転送が可能です。また、各 PIPE にて行うデータ転送プログラムが簡単に記述できるように以下の機能を備えています。

- (1) 豊富な PIPE 割り込み 【共通編 2.3 割り込み機能参照】
- (2) 合計 3.5K バイトの FIFO バッファ 【共通編 5.1 FIFO バッファ、本資料5.6.1、5.6.2参照】
- (3) 連続送受信機能(FS バルク転送) 【本資料5.5参照】

データ転送タイプによる特徴は以下のとおりです。

表5.1 データ転送タイプ一覧

転送タイプ	バルク転送		インタラプト転送	
実時間性	低い		リアルタイム	
転送間隔	非保証		保証	
データ転送量	大量		少量	
パケット数	3		3	
Data パケット	Data0/Data1		Data0/Data1	
Handshake	有り		有り	
最大パケットサイズ	8,16,32,64(FS)	512(HS)	1-64(FS)	1024 * 3 / n uFrame(HS)
連続送受信機能	有効		使用禁止	
ダブルバッファ機能	有効/無効		有効/無効	
zero-length パケット自動送信機能	有効		使用禁止	
DMA アクセス	有効		有効	

M66591は PIPE1-2 にダブルバッファ機能を備えています。ダブルバッファを使用することにより、デバイスがホストとデータ通信中でも、FIFO バッファ空きを待つことなくバッファアクセスを行うことが可能です。ダブルバッファ機能を使用することにより USB バスのトラフィックを向上させることができます。

また、FS バルク転送に対し連続送受信機能を使用することができます。データ転送における連続送受信機能は、最大 512 バイトまで割り込みを発生せずに連続してデータを送受信する機能です。連続送受信機能を用いることで、バルク転送では 1 トランザクションごとに FIFO バッファアクセスを行う必要がなくなり、CPU 稼働率を高めることができます。

M66591は、FIFO バッファがフルでホストからのデータを受信できない状態、または FIFO バッファが空でホストからの IN トークンに対してデータ送信を行えない状態のときに、ホストからのトークンパケットを受信すると自動的に”NAK”応答を行います。

5.2. PIPE と PIPE コントロール

M66591はデフォルトコントロール PIPE 以外に、PIPE1～PIPE6 の 6 本 PIPE を内蔵しています。この 6 本の PIPE (PIPE1～PIPE6) はそれぞれバルク転送とインタラプト転送に設定することができます。表 5.2 は PIPE1～PIPE6 の構成を示します。

表5.2 PIPE (PIPE1～PIPE6) 設定項目一覧表

機能	レジスタビット	PIPE1～PIPE6
PIPE 許可	PEN	PIPE 許可／不許可を設定
転送方向	DIR	IN/OUT に設定可能 PIPE1～PIPE4 のみ設定可能です。PIPE5～6 は IN 固定です。
エンドポイント番号	EP_NUM [2:0]	PIPE のエンドポイント番号を設定可能 エンドポイント番号は PIPE 番号と同じです。
ダブルバッファ	DBLB	PIPE1～PIPE2 のみ設定可能
連続送受信	CNTMD	Full-Speed バルク転送 (PIPE1～PIPE4) のみ設定可能
バッファサイズ	PIPE により固定	PIPE1～PIPE4 : 512 バイト固定 PIPE5～PIPE6 : 64 バイト固定
応答 PID	PID [1:0]	NAK、STALL 及び BUF に設定可能
NYET ハンドシェイクモード	NYETMD	NYET ハンドシェイクの使用するかどうかを設定可能 PIPE1～PIPE4 のみ設定可能です。
受信データ読み捨てモード	ACLR	PIPE1～PIPE6 に設定可能
バッファレディ割り込み出力	BFRE	PIPE1～PIPE4 のみ設定可能
インタラプト転送トグルモード	ITMD	インタラプト転送 (PIPE5～PIPE6) のみ設定可能
マックスパケットサイズ	PIPE により固定	PIPE1～PIPE4 : Full-Speed は 64 バイト、 Hi-Speed は 512 バイト固定 PIPE5～PIPE6 : 64 バイト固定

M66591は 3.5K バイトの内部メモリを内蔵しています。この 3.5K バイトのメモリ領域は PIPE0 と PIPE1～PIPE6 の各 PIPE のバッファメモリとして使用されます。

5.3. PIPE (エンドポイント) 初期設定

PIPE 初期設定制御手順例を図 5.1 に示します。

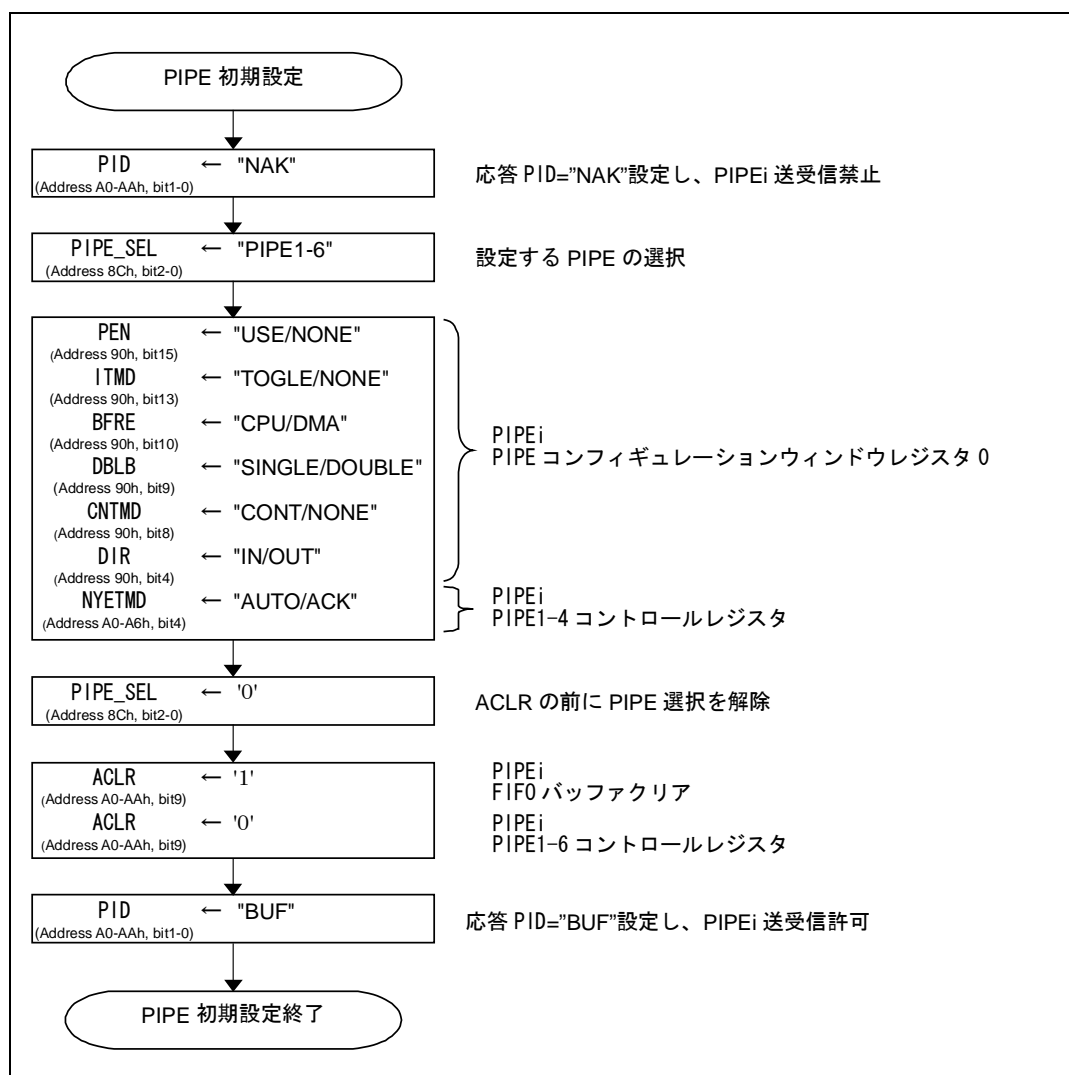


図5.1 PIPE 初期設定制御手順例

5.4. PING/NYET コントロール

PING 転送コントロールはハイスピード (Hi-Speed) のバルク OUT 転送、コントロールライトデータステージの OUT 転送の時にのみ有効です。

(1) NYET ハンドシェイク

各 PIPE の NYET ハンドシェイクの振る舞いは PIPEi コントロールレジスタの NYETMD ビットによって設定します。

NYETMD = '0' : バッファの状態に応じて自動的に NYET ハンドシェイクを行います。

この設定の場合、バッファのモード設定によって以下のように NYET を応答します。

- PIPE バッファがシングルバッファの場合は、常に NYET を応答します。
- PIPE バッファがダブルバッファの設定の場合は、FIFO バッファの状態によって NYET を応答します。

ショートパケットを受信した場合、バッファの状態に関係なく ACK を応答します。

NYETMD = '1' : 常に ACK/NAK ハンドシェイクを行い、NYET ハンドシェイクを行いません。

(2) PING フローコントロール

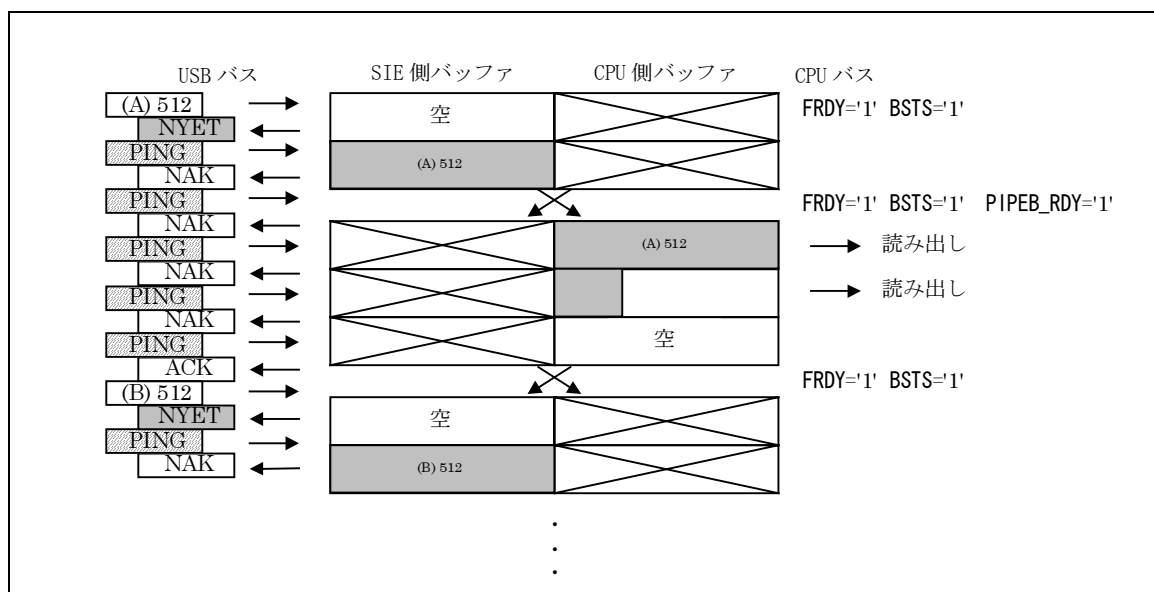
対応する PIPE のバッファがデータを受信できる状態であれば、ACK ハンドシェイクを応答します。そうでなければ、NAK を応答し、バッファノットレディ割り込みが発生します。

5.4.1. PING/NYET 例

以下(1)~(2)に、PING/NYET の例を示します。

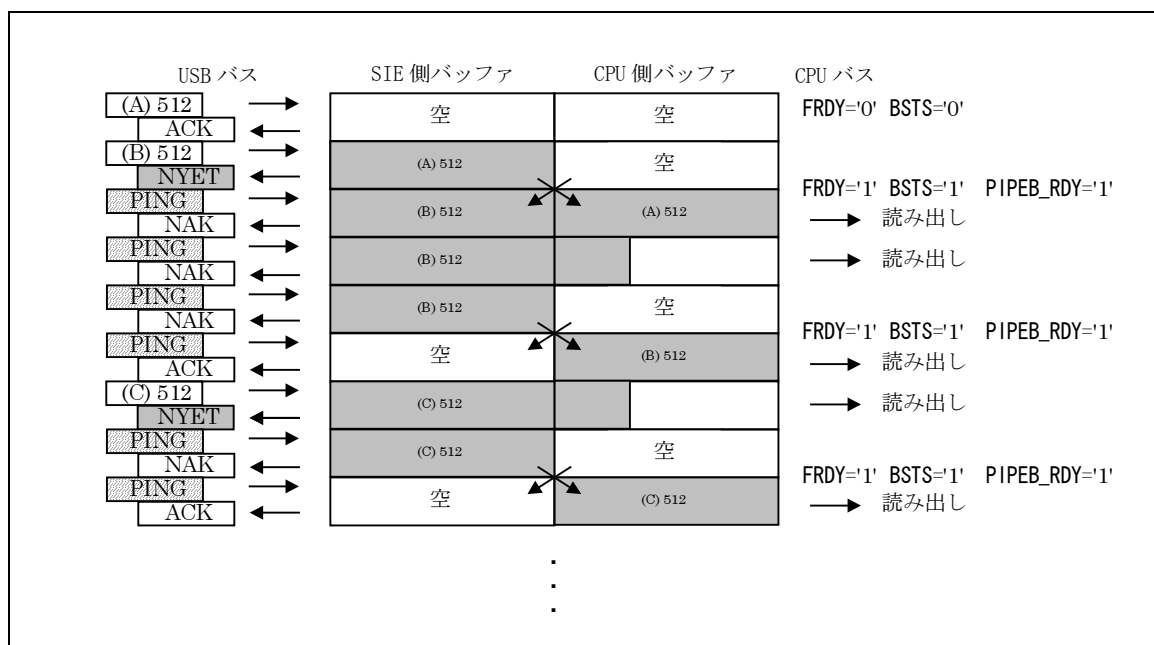
(1) Hi-Speed、Bulk OUT 転送設定、非連続受信、シングルバッファ時のバッファの状態

図 5.2に Hi-Speed、Bulk OUT 転送設定、非連続受信、シングルバッファに設定したときのバッファの状態と FRDY、BSTS ビット、バッファレディ割り込み及び PING/NYET の関係を示します。



(2) Hi-Speed、Bulk OUT 転送設定、非連続受信、ダブルバッファ時のバッファの状態

図 5.3に Hi-Speed、Bulk OUT 転送設定、非連続受信、ダブルバッファに設定したときのバッファの状態と FRDY、BSTS ビット、バッファレディ割り込み及び PING/NYET の関係を示します。



5.5. 連続転送機能

CNTMD（連続送受信モードビット）により、バルク転送時の送受信モードを設定します。このビットは PIPE1～PIPE4 にのみ有効です。PIPE5 と PIPE6 には無効です。

表5.3 連続転送設定

ビット	Full-Speed 時	Hi-Speed 時
CNTMD	'0': 非連続送受信モード	'0': 設定禁止
	'1': 連続送受信モード	'1': 非連続送受信モード

(1) Full-Speed で動作している場合

表5.4 Full-Speed連続転送設定

	IN				OUT		
	送信完了		書き込み完了		受信完了		
CNTMD=0	64 バイト 送信	ショート パケット	64 バイト 書き込み	BVAL=1 書き込み	64 バイト 受信	ショート パケット	TRNCNT= パケット 受信回数
CNTMD=1	64×8=512 バイト 連続送信	or zero-length パケット送信	64×8=512 バイト 書き込み		64×8=512 バイト 連続受信	or zero-length パケット受信	

CNTMD = '0' : 非連続送受信モード

PIPE が IN 設定の時の送信完了条件 :

- ・ 64 バイト分のデータを送信
- ・ ショートパケットを送信または、zero-length パケットを送信

PIPE が IN 設定の時の書き込み完了条件 :

- ・ 64 バイト分のデータをバッファへ書き込み
- ・ BVAL ビットへの '1' の書き込み

PIPE が OUT 設定の時の受信完了条件 :

- ・ 64 バイト分のデータを受信
- ・ ショートパケットを受信または、zero-length パケットを受信
- ・ TRNCNT に設定した値と、パケットの受信回数が一致した場合

CNTMD = '1' : 連続送受信モード

PIPE が IN 設定の時の送信完了条件 :

- ・ 64 バイト分のデータを自動的に複数回送信し、512 バイト分のデータを送信
- ・ ショートパケットを送信、または zero-length パケットを送信

PIPE が IN 設定の時の書き込み完了条件 :

- ・ 512 バイト分のデータをバッファへ書き込み
- ・ BVAL ビットへの '1' の書き込み

PIPE が OUT 設定の時の受信完了条件 :

- ・ 64 バイト分のデータを自動的に複数回受信し、512 バイト分のデータを受信
- ・ ショートパケットを受信、または zero-length パケットを受信
- ・ TRNCNT に設定した値と、パケットの受信回数が一致した場合

(2)Hi-Speed で動作している場合、

表5.5 Hi-Speed連続転送設定

	IN				OUT		
	送信完了		書き込み完了		受信完了		
CNTMD=0	設定禁止						
CNTMD=1	512 バイト 連続送信	ショート パケット or zero-length パケット送信	512 バイト 書き込み	BVAL=1 書き込み	512 バイト 連続受信	ショート パケット or zero-length パケット受信	TRNCNT= パケット 受信回数

Hi-Speed で動作している場合は、非連続送受信モードのみ動作します。但し、このビットを'1'に設定する必要があります。

CNTMD = '0' : 設定禁止

CNTMD = '1' : 非連続送受信モード

PIPE が IN 設定の時の送信完了条件 :

- ・ 512 バイト分のデータを送信
- ・ ショートパケットを送信、または zero-length パケットを送信

PIPE が IN 設定の時の書き込み完了条件 :

- ・ 512 バイト分のデータをバッファへ書き込み
- ・ BVAL ビットへの'1'の書き込み

PIPE が OUT 設定の時の受信完了条件 :

- ・ 512 バイト分のデータを受信
- ・ ショートパケットを受信、または zero-length パケットを受信
- ・ TRNCNT に設定した値と、パケットの受信回数が一致した場合

このビットを設定する前に、PIPE_i コントロールレジスタ(i=1~4)の PID ビットを'00' (NAK) に設定してください。

5.6. データ送受信割り込み

M66591はバッファレディ、バッファノットレディ、バッファエンプティの3種類のデータ転送割り込み機能を備えています。また、各割り込みは割り込みステータスレジスタ 1~3 により、割り込み要因の発生した PIPE を認識することが可能です。データ転送割り込みは転送タイプ、転送方向及びアクセス方法により発生要因が異なりますのでご注意ください。要因の差異に関してはアプリケーションノート M6659x 共通編 2.3 割り込み機能を参照ください。

データ送受信制御手順例を図 5.4に示します。

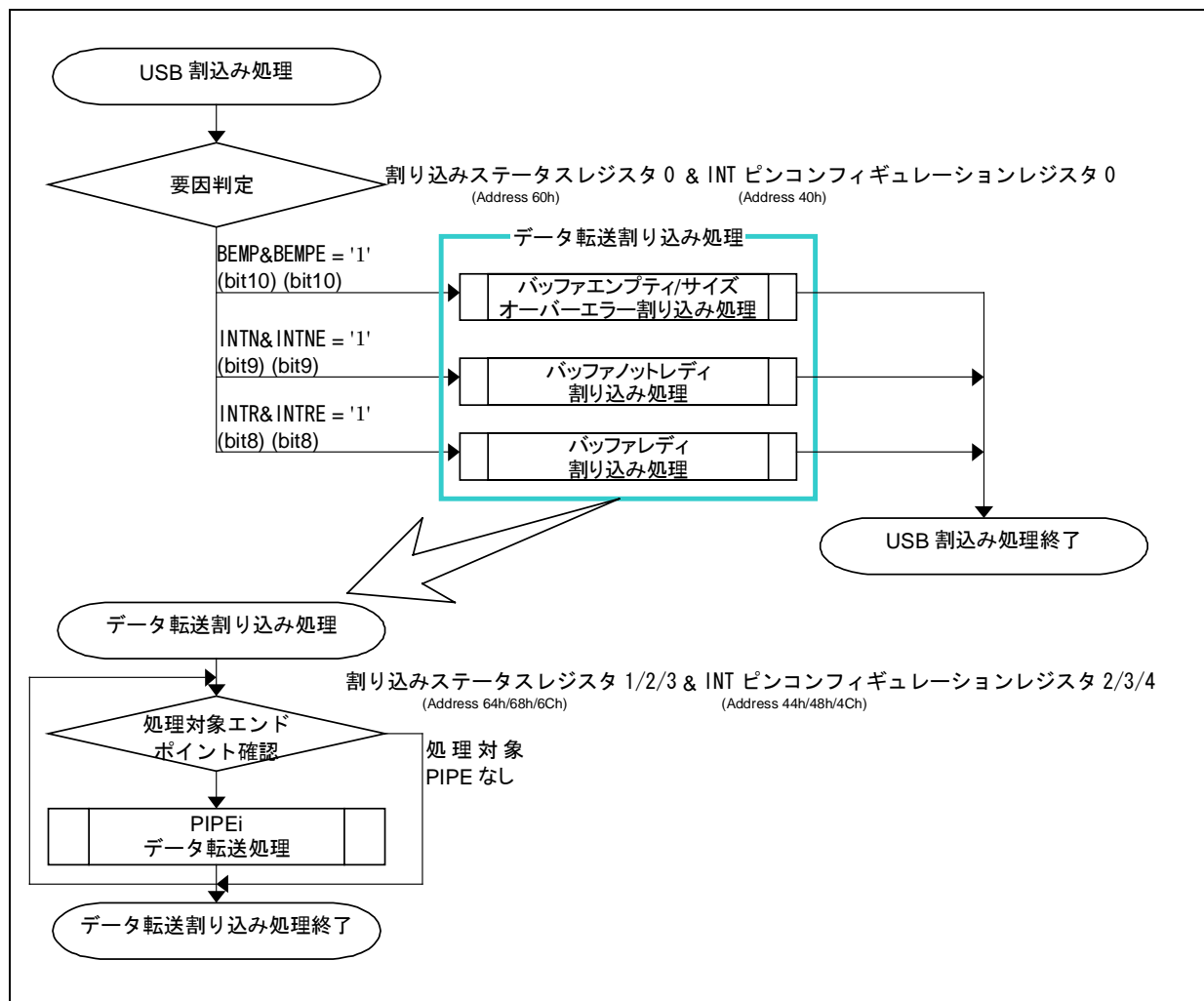


図5.4 データ送受信制御手順例

5.6.1. FIFO アクセス(データ送信:IN 方向)

デバイス側に、ホストへの送信データが存在する場合は、IN トランザクションによりデータを送信します。USB 通信はホスト PC のサンプリングによりデータ転送が行われるため、デバイス側アプリケーションは送信データが発生したら FIFO バッファに送信するデータを書き込み、ホストからの IN トークンを待ちます。データ送信が終了し、FIFO バッファに CPU 側からの書き込みが可能となった時点で、CPU に対してバッファレディ割り込みを通知します。要因の詳細に関してはアプリケーションノート M6659x 共通編 2.3 割り込み機能を参照ください。

データ送信設定制御手順例を図 5.5 に示します。

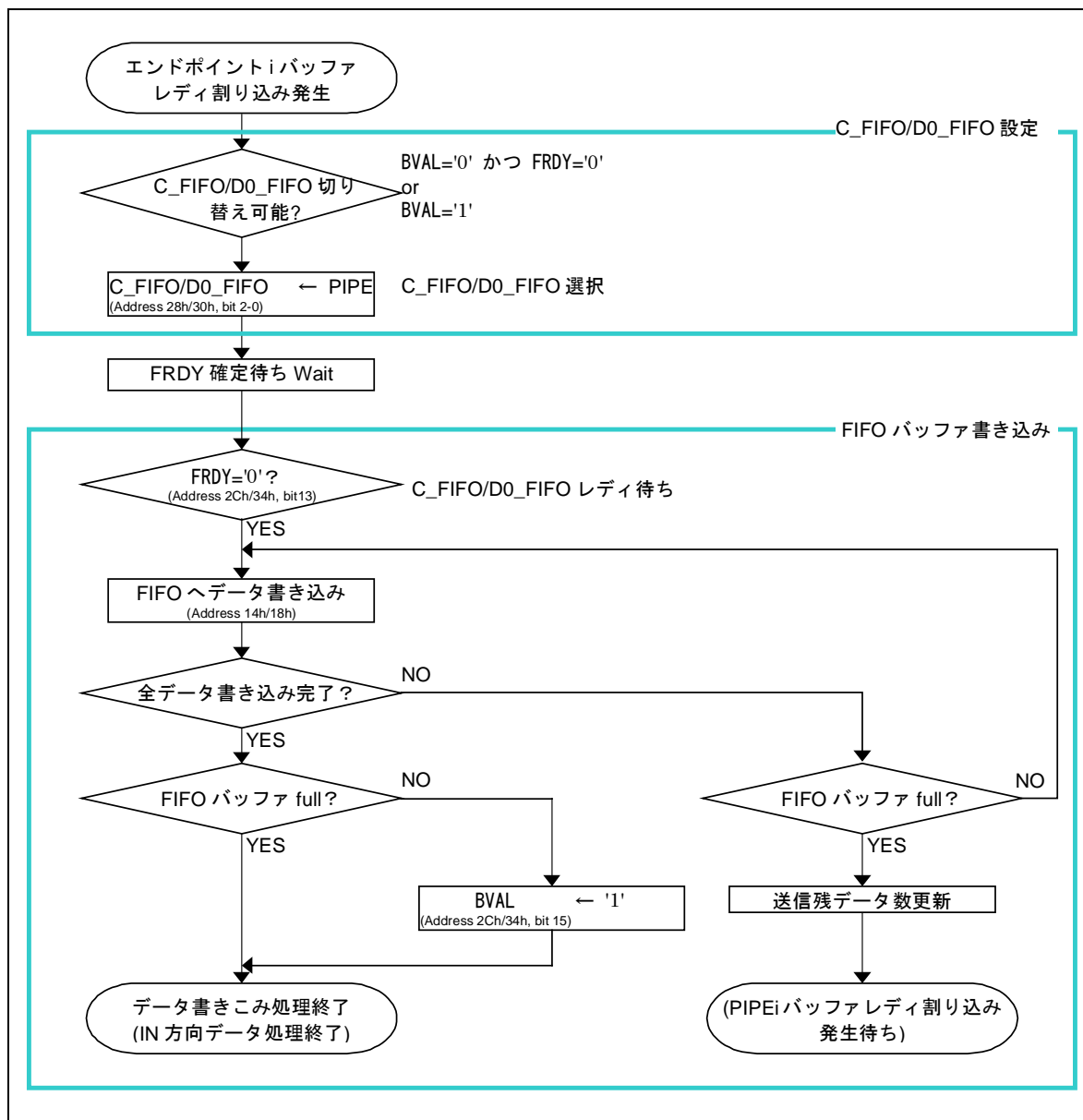


図5.5 データ送信設定制御手順例

5.6.2. FIFO アクセス(データ受信:OUT 方向)

デバイスがホスト PC からデータを受信する場合は、OUT トランザクションによりデータ転送を行います。USB 通信はホスト PC のサンプリングによりデータ転送が行われるため、デバイス側アプリケーションでは、あらかじめ M66591 の FIFO バッファを空き状態(受信可能状態)にしておき、ホストからの OUT トークンを待つ処理を行ってください。M66591 は、受信完了により FIFO バッファの読み出しが可能となったことを、CPU に対してバッファレディ割り込み発生させることによって通知します。要因の詳細に関してはアプリケーションノート M6659x 共通編 2.3 割り込み機能を参照ください。

データ受信設定制御手順例を図 5.6 に示します。

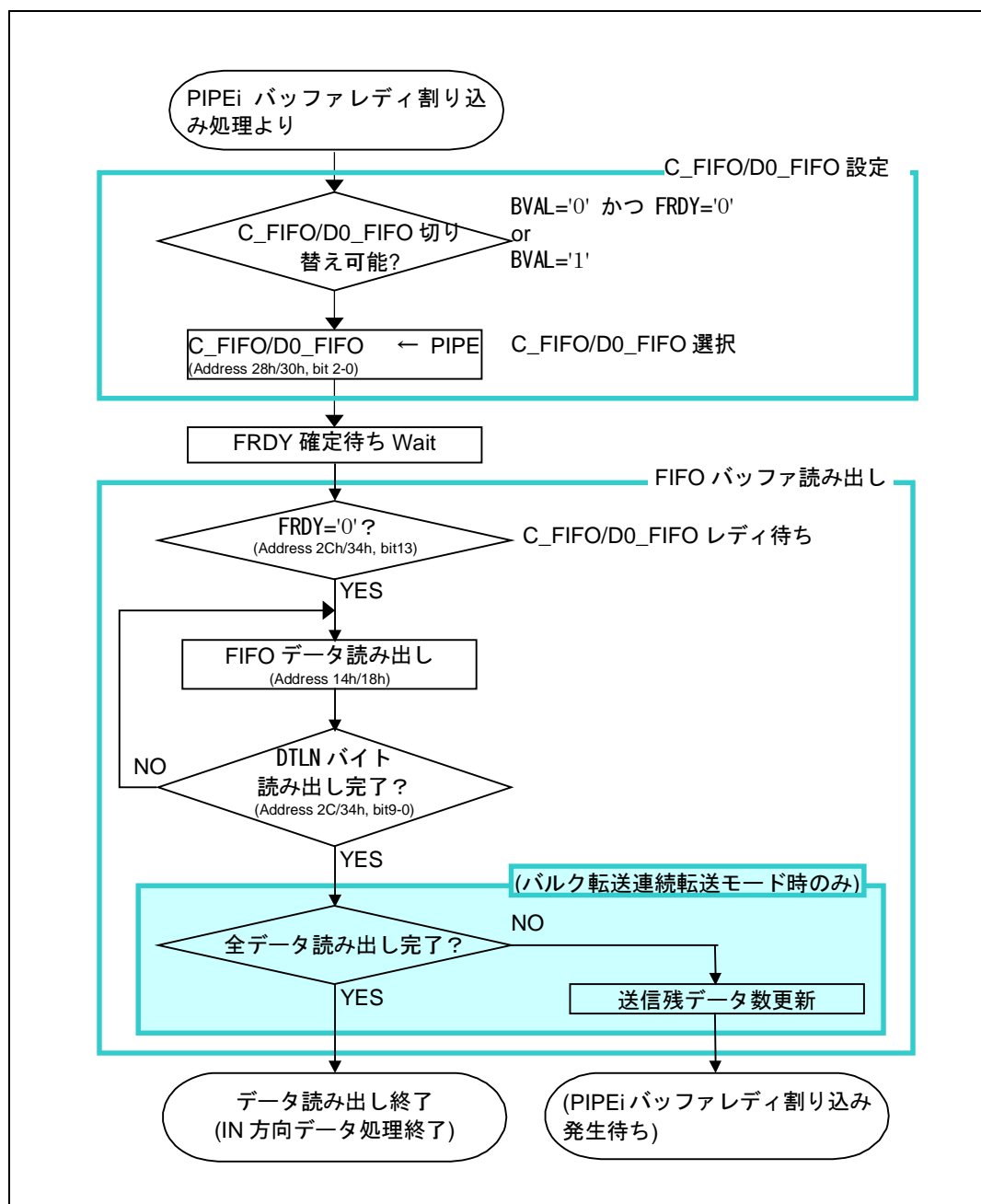


図5.6 データ受信設定制御手順例

5.6.3. リードカウントモード

C_FIFO、D0_FIFO の RCNT ビットにより、それぞれ CPU_DTLN ビットと、DMA_DTLN ビットのカウントダウンモードの選択が可能です。

RCNT='0'の場合：

CPU_DTLN は、C_FIFO ポートレジスタからデータの読み出しを行っても、値が変化せず全てのデータを読み出した時にクリアされます。

DMA_DTLN は、D0_FIFO ポートレジスタからデータの読み出しを行っても、値が変化せず全てのデータを読み出した時にクリアされます。

RCNT='1'の場合：

CPU_DTLN は、C_FIFO ポートレジスタからデータを読み出されるたびに、値がカウントダウンします。

DMA_DTLN は、D0_FIFO ポートレジスタからデータを読み出されるたびに、値がカウントダウンします。

5.6.4. バッファリワインド

C_FIFO、D0_FIFO の REW ビットを'1'に設定することにより、それぞれ C_FIFO ポートレジスタと D0_FIFO ポートレジスタの読み出し/書き込みポインタをリワインドできます。このビットへの'0'書き込みは無効です。

- (1) Current_PIPE に指定された PIPE が OUT 方向の場合：
リワインド操作後、バッファのデータ先頭から再度読み出し可能です。

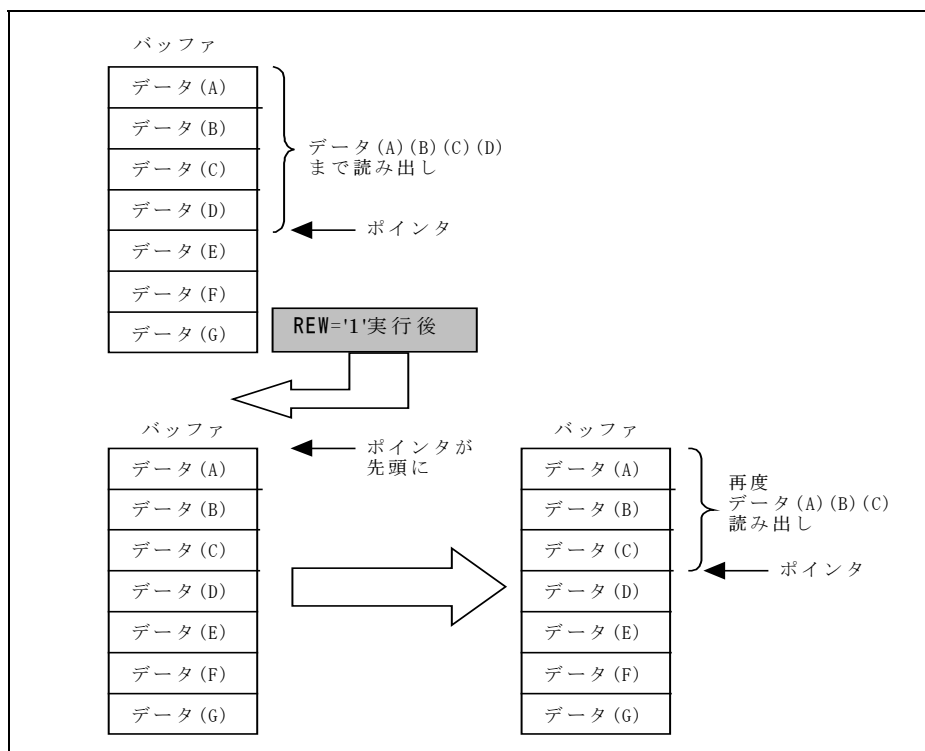


図5.7 バッファリワインド操作例

- (2) Current_PIPE に指定された PIPE が IN 方向の場合：
リワインド操作後、それまでに書き込んだデータは無効になり、バッファ先頭から再度書き込み可能です。

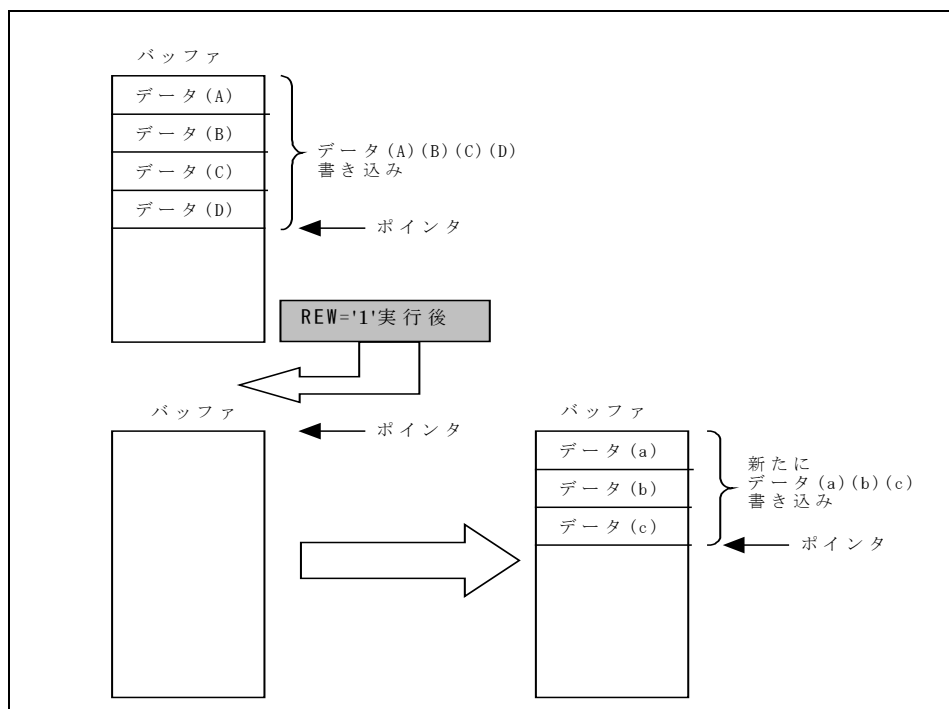


図5.8 バッファリワインド操作例

5.6.5. 自動バッファクリア機能 (ABCR ビット「D0_FIFO ポートコントロールレジスタ 0」)

自動バッファクリア機能は、OUT 転送において D0_FIFO を使用し、S/W による制御を行わず転送を行いたい場合に効果があります。通常、DMA モード (BFRE = '1') を使用した DMA 転送では、ショートパケットデータ等を受信した時に、DMA_DLTN を読み出すことによって最後に読み出したデータサイズを確認することが出来ます。その際には、H/W によって BVAL 等のバッファ情報がクリアされないため、S/W によるバッファクリア処理 (BCLR = '1') が必要になります。

ABCR ビットは、Current_PIPE に設定された PIPE の方向が OUT の時のみ有効です。

このビットを'1'に設定した場合、下記(1)または(2)の時に、H/W によって FIFO をクリアします。

- (1) バッファが空いている状態で zero-length パケットを受信した時
- (2) Current_PIPE に設定された PIPE の BFRE ビットが'1'で、ショートパケットを受信した時 (zero-length パケットを含む) または、トランザクションカウンタ分のパケットを受信した時

ABCR='0': 自動バッファクリアモード禁止

上記(1)または(2)の場合、バッファのデータを全て読み出しても、バッファの状態はクリアされません。BVAL = '1'の状態を保持します。その為 DMA 転送により読み出し終了後、DMA_DTLN (RCNT = '0') を読み出すことによって、最後の転送が何バイトあったかを確認する事ができます。バッファをクリアするには BCLR ビットに'1'を設定してください。

表5.6 パケット受信とRDY割り込みおよびバッファクリア処理

バッファ状態	BFRE = '0' (CPU モード)		BFRE = '1' (DMA モード)	
	RDY 割込	BCLR 処理	RDY 割込	BCLR 処理
マックスパケット受信 + バッファフル	受信時に発生*1	不要	発生しない	不要
zero-length パケット受信	受信時に発生*1	要	受信時に発生*1	要
zero-length 以外のショートパケット受信	受信時に発生*1	不要	データ読み出し完了時に発生	要
トランザクションカウンタ終了	受信時に発生*1	不要	データ読み出し完了時に発生	要

*1: ダブルバッファ設定の時には、受信時にもう片方のバッファがまだ読み出し完了していなければ、読み出しが完了するまで RDY 割り込みは発生しません。

ABCR='1': 自動バッファクリアモード許可

バッファのデータを全て読み出し後、自動的にバッファがクリアされ、次のデータを受信可能状態になります。

表 5.6中の BCLR 処理の"要"が全て"不要"になります。

表5.7 パケット受信とRDY割り込みおよびバッファクリア処理 (ABCR='1')

バッファ状態	BFRE = '0' (CPU モード)		BFRE = '1' (DMA モード)	
	RDY 割込	BCLR 処理	RDY 割込	BCLR 処理
マックスパケット受信 + バッファフル	受信時に発生*1	不要	発生しない	不要
zero-length パケット受信	受信時に発生*1	不要	受信時に発生*1	不要
zero-length 以外のショートパケット受信	受信時に発生*1	不要	データ読み出し完了時に発生	不要
トランザクションカウンタ終了	受信時に発生*1	不要	データ読み出し完了時に発生	不要

*1: ダブルバッファ設定の時には、受信時にもう片方のバッファがまだ読み出し完了していなければ、読み出しが完了するまで RDY 割り込みは発生しません。

S/W による制御なしにショートパケット受信を含んだ連続したトランザクションを DMA 転送するシステムの場合には、自動バッファクリア機能を許可 (ABCR = '1') にしてください。この場合、データ読み出し完了時に H/W によって自動にバッファがクリアされ、次のデータを受信可能状態になります。

また ABCR = '1'の場合でも RDY 割り込みの発生条件には影響しません。この機能を使用した PIPE に対して RDY 割り込みを発生させたくない場合には、PIPEB_RE ビットの対応するビットを'0'にして割り込み出力をマスクしてください。なお、Current_PIPE に IN 方向の PIPE を指定している場合には、ABCR ビットの設定は無効になります。

5.6.5.1. 自動バッファクリア使用例

(1) 最後に受信したデータ長を知りたい場合、又はトランザクションカウント終了を知りたい場合

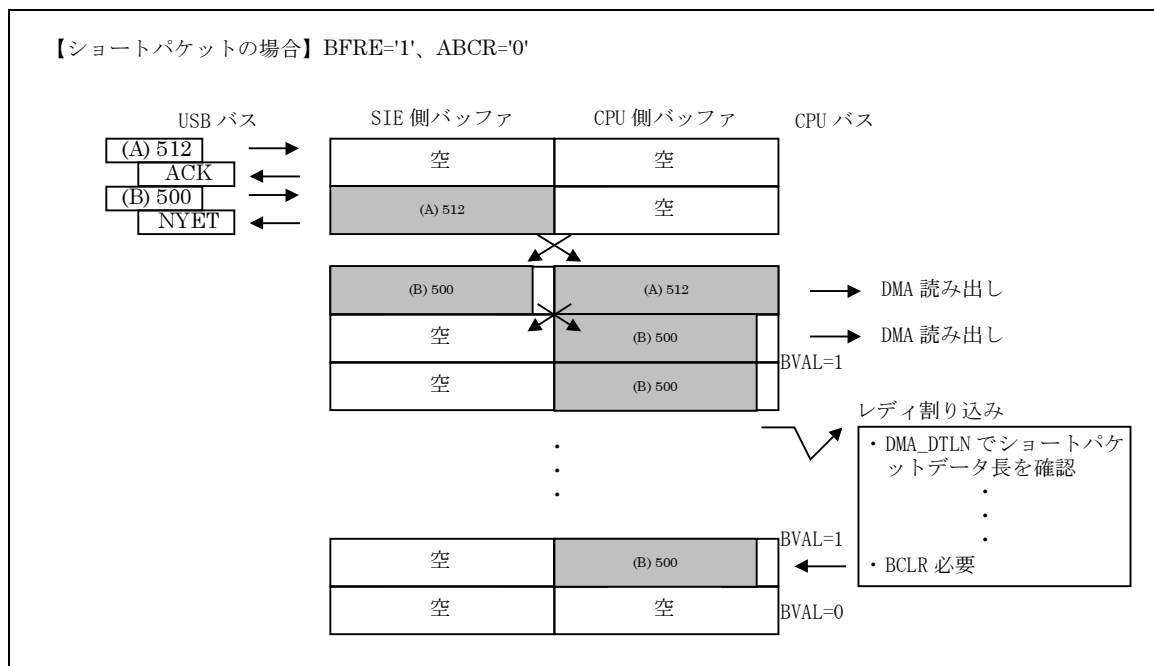


図5.9 ショートパケット受信時の動作

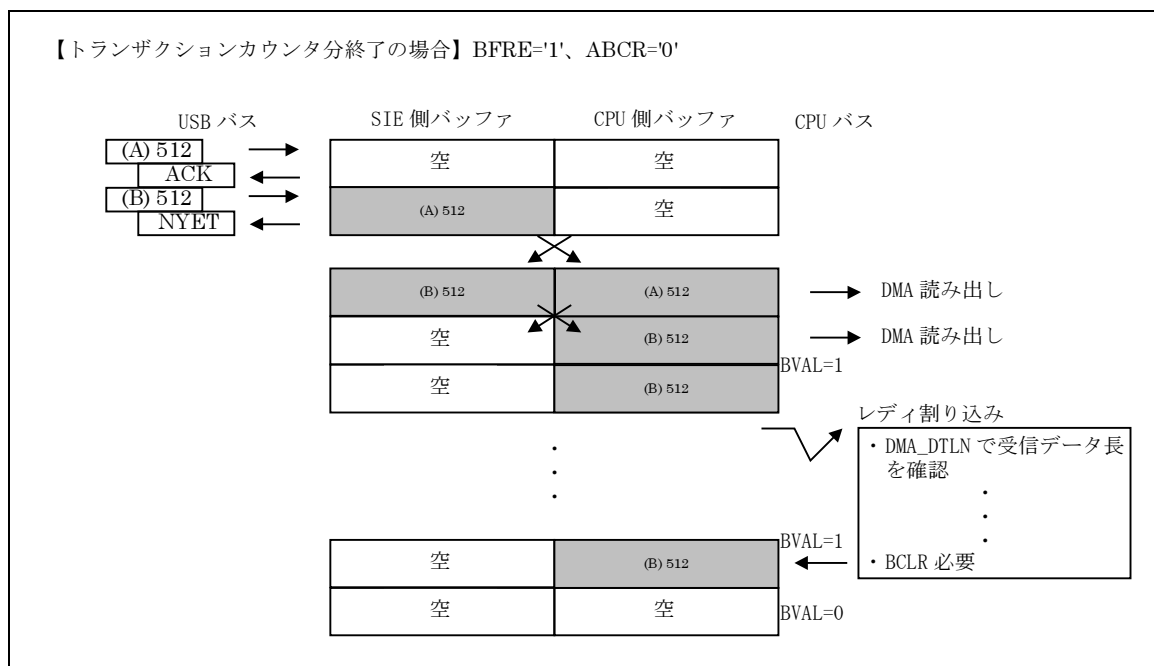


図5.10 トランザクションカウンタ分終了時の動作

(2) ショートパケットを含んだ連続したトランザクション受信を S/W 介在なしに行いたい場合

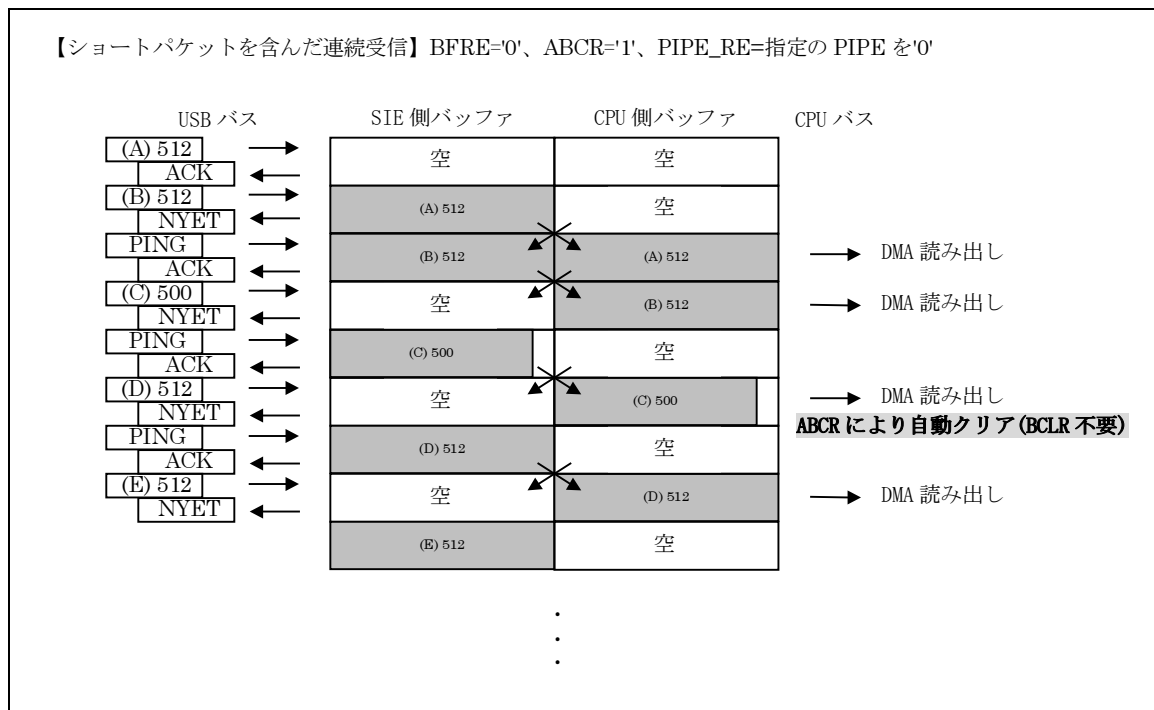


図5.11 ショートパケットを含んだ連続受信時の動作

5.6.6. バッファ強制トグル機能 (TGL ビット「C_FIFO ポートコントロールレジスタ 2」)

連続転送モードにおいて、FIFO バッファがフルになっていない状態でこのビットに'1'を書き込むことにより、SIE 側のバッファを CPU 側のバッファに切り替えることができます。このときに、バッファレディ割り込みが発生します。

このビットは OUT 方向の PIPE にのみ使用可能です。

なお、Current_PIPE ビットに設定された PIPE がデフォルトコントロール PIPE の場合には、このビットへの'1'書き込みは無効です。

このビットへの'0'書き込みは無効です。

バッファ強制トグル機能は、ショートパケットで終了しない且つ、総転送数が不明でトランザクションカウント機能が使用できない場合に、最終データのバッファを読み出す際に使用可能です。

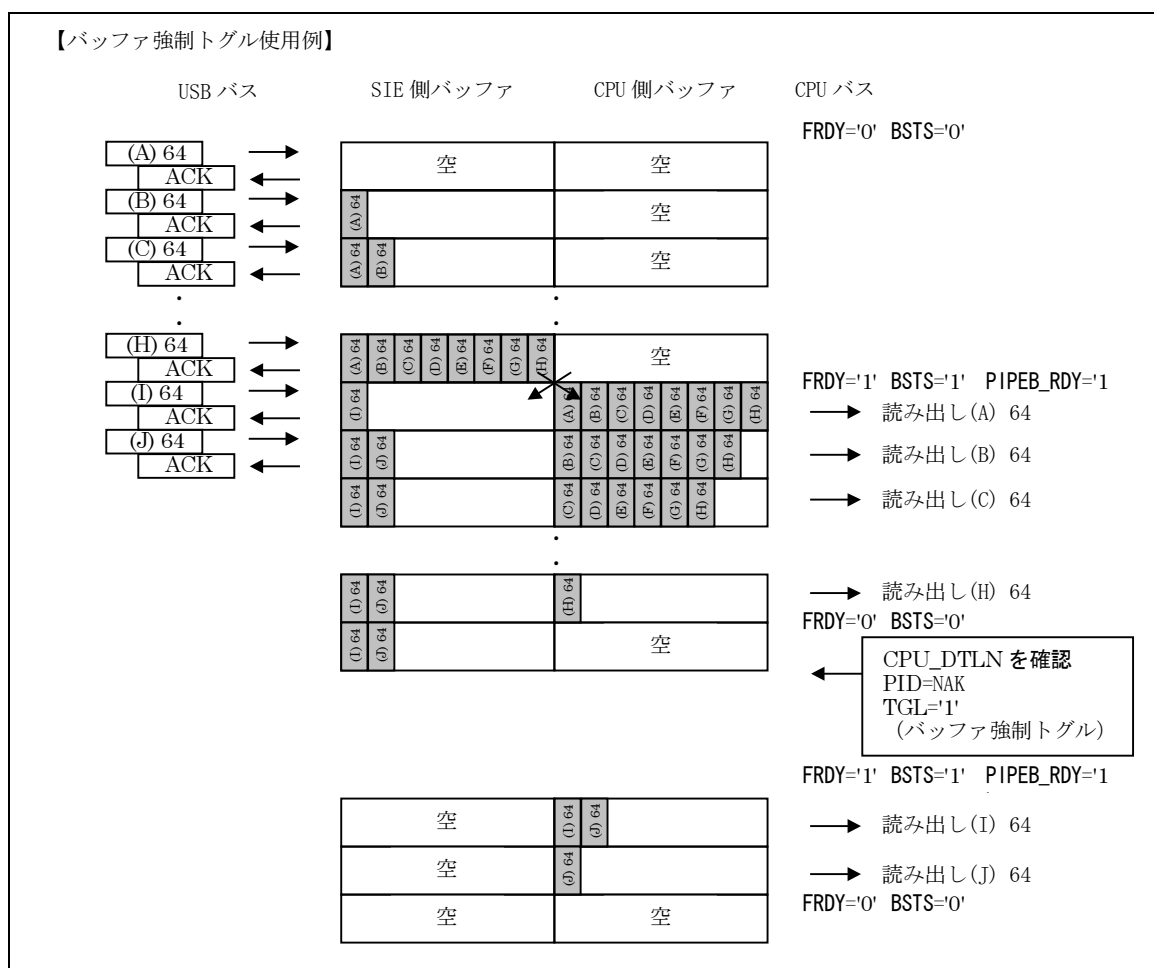


図5.12 バッファ強制トグル時の動作

5.6.7. トランザクションカウント機能 (TRENb「D0_FIFO ポートコントロールレジスタ 0」)

トランザクションカウント機能は、連続転送機能を使用したバルク OUT 転送で最終パッケージがマックスパッケージサイズで終了することがある場合に有効です。

トランザクションカウント機能を使用しない場合には、最終パッケージがマックスパッケージであり、かつバッファがまだフルになっていない場合には、FIFO ポートが読み出し可能状態にならないため、最後の数パッケージのデータを読み出すことができません。その場合トランザクションカウント機能を用いて、H/W にてパッケージ数をカウントし、指定したパッケージ数になれば受信終了として FIFO ポートから読み出し可能状態にすることができます。但しトランザクションカウント機能を使用する場合には、指定する PIPE の総パッケージ数を S/W によって把握する必要があります。総パッケージ数が不明の場合には、トランザクションカウント機能および連続転送機能は使用できませんので注意が必要です。

なお、C_FIFO の場合には、TGL ビットを使用することにより FIFO ポートを読み出し可能状態にすることができます。

この機能は IN 方向に設定されている PIPE には無効です。

トランザクションカウント機能は、以下のような設定手順でご使用ください。

- ① PIPE コンフィギュレーションウィンドウレジスタ 0 を設定
- ② Current_PIPE、TRENb = '1' を設定
- ③ TRNCNT ビット値の設定
- ④ PIPE_i コントロールレジスタ (i=1~4) の PID = '01' (BUF) 設定 → 受信開始

注意事項：

- (1) トランザクションカウント機能を使用する場合には、指定した PIPE のトランザクションが終了するまで Current_PIPE の変更は行わないで下さい。変更する場合には、トランザクションカウント機能を禁止 (TRENb='0') し、TRclr ビットへの '1' 書き込みによってカレントカウントレジスタをクリアしてから、Current_PIPE ビットを変更してください。
- (2) トランザクションカウント許可 (TRENb='1') かつ PID='01' (BUF 応答) の時には、TRclr ビットに '1' を書き込まないでください。TRclr ビットへの '1' 書き込みによってカレントカウントレジスタをクリアし、カウントを再開する場合には、一旦 PID = '00' (NAK 応答) に設定後、C_FIFO ポートレジスタと D0_FIFO ポートレジスタを使用してその PIPE のデータをすべて読み出すか、バッファオートクリアモード (ACLR) 等によって FIFO バッファ内のデータをクリアしてから、TRclr='1' を書き込んでください。

5.6.8. トランザクションカウント機能使用例

(1) トランザクションカウンタ最大値以内の場合 (Hi-Speed)

- ・ダブルバッファ
- ・非連続転送 (CNTMD='1')
- ・BFRE='1'
- ・転送パケット数=100

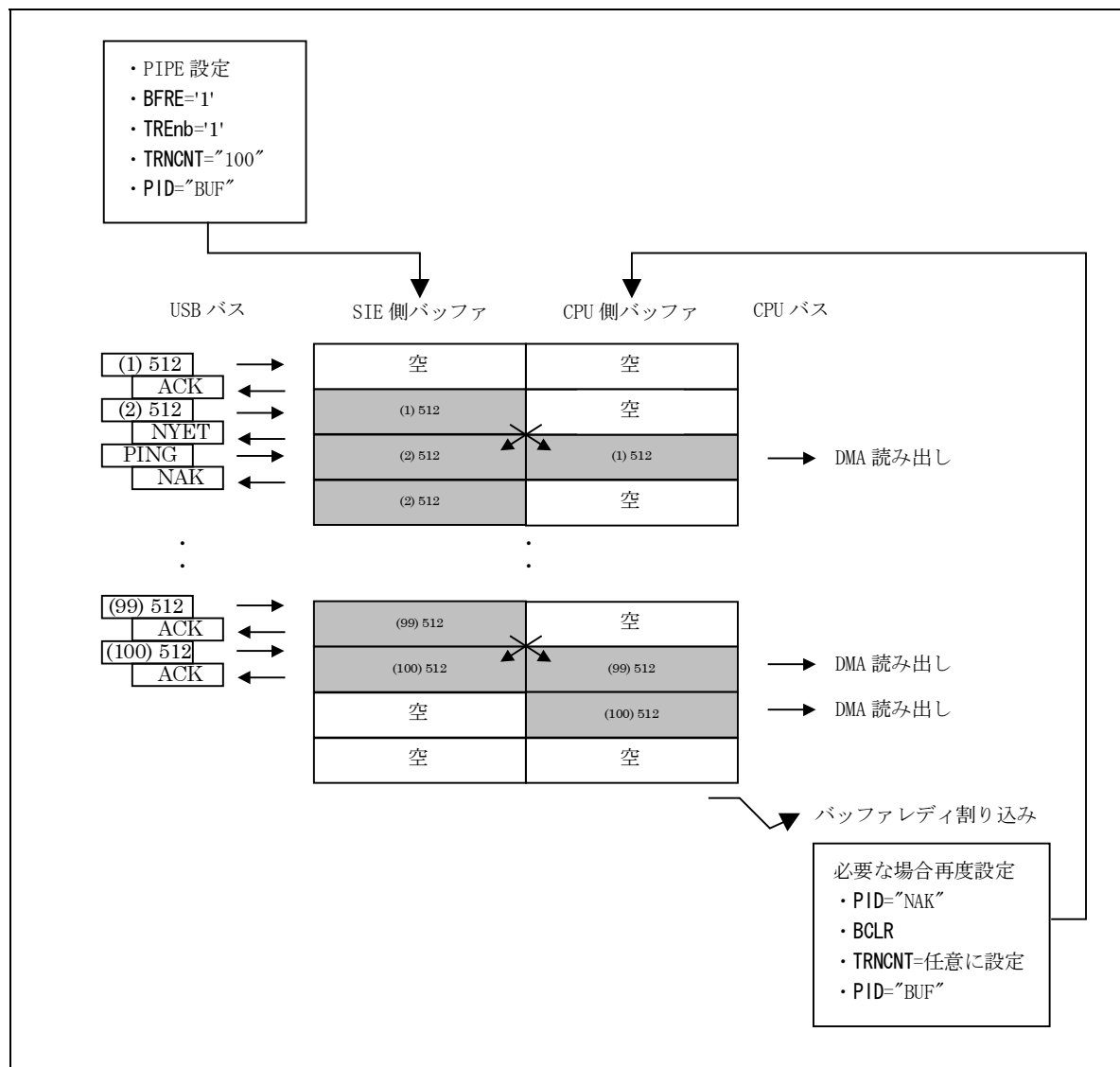


図5.13 トランザクションカウンタ最大値以内の場合 (Hi-Speed)

(2) トランザクションカウンタ最大値以上の場合 (Hi-Speed)

- ・ダブルバッファ
- ・非連続転送 (CNTMD='1')
- ・BFRE='1'
- ・転送パケット数=65540 (32770×2)

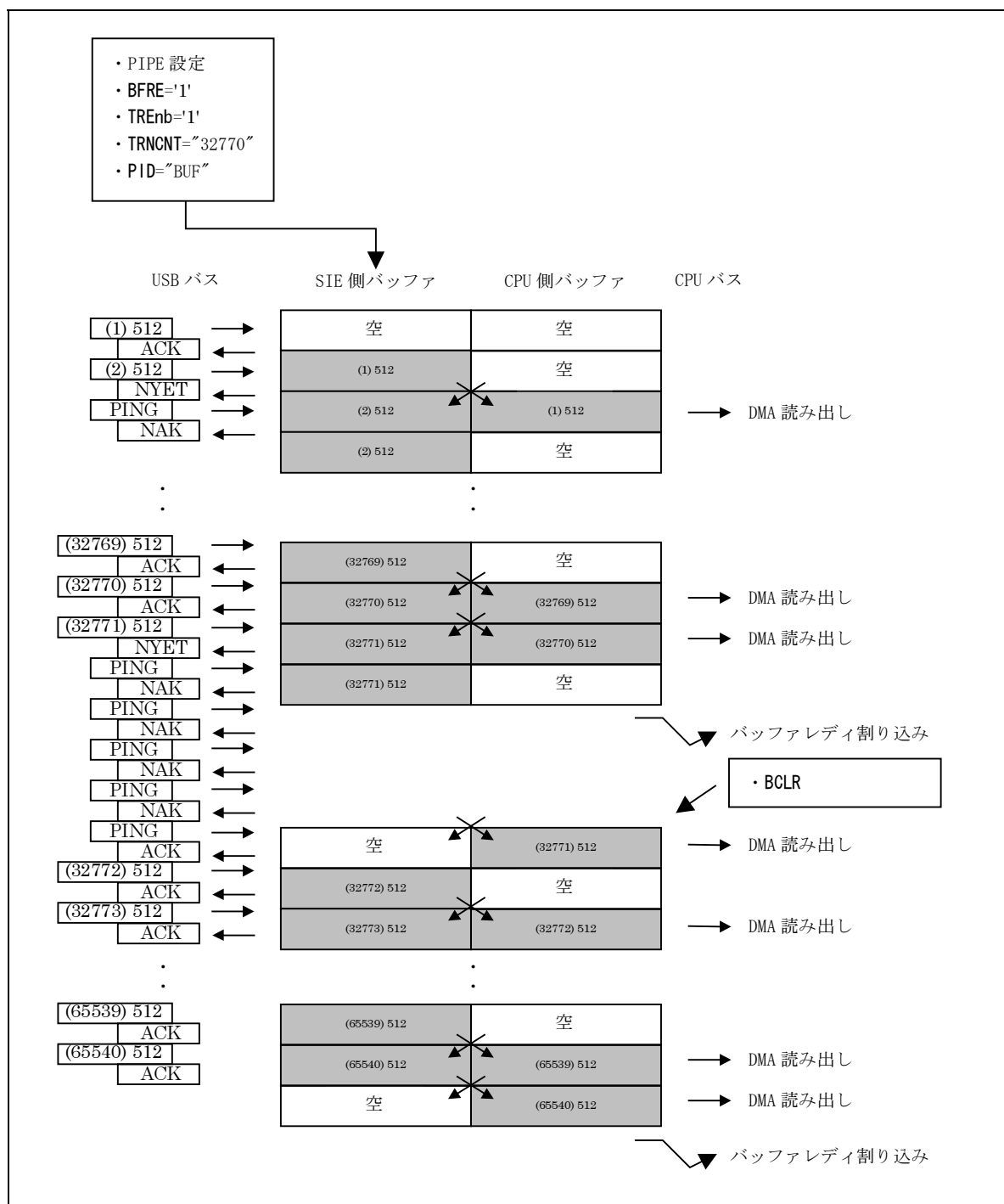


図5.14 トランザクションカウンタ最大値以上の場合 (Hi-Speed)

(3) トランザクションカウンタ最大値以内の場合 (Full-Speed)

- ・ダブルバッファ
- ・連続転送 (CNTMD='1')
- ・BFRE='1'
- ・転送パケット数=100

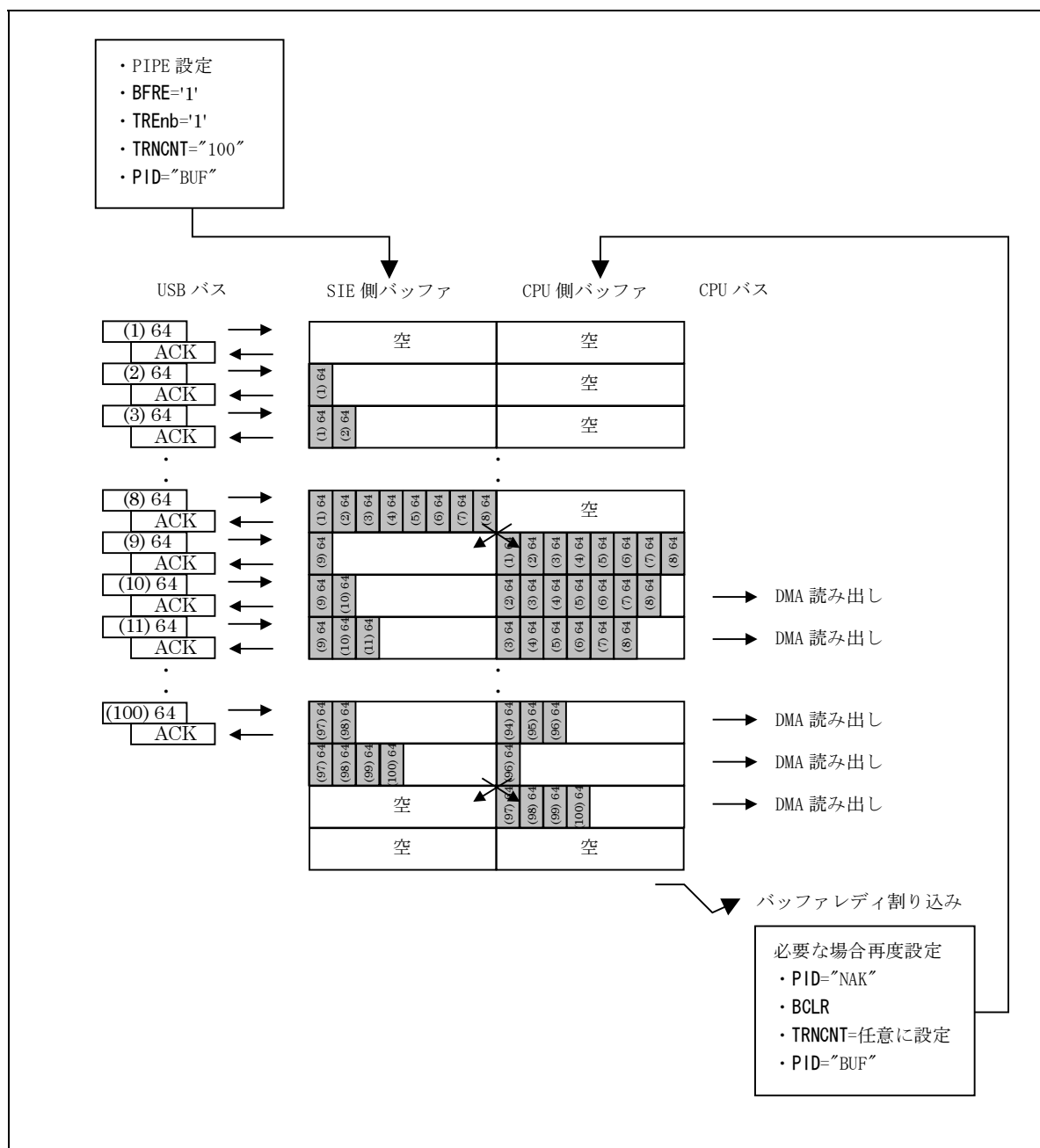


図5.15 トランザクションカウンタ最大値以内の場合 (Full-Speed)

(4) トランザクションカウンタ最大値以上の場合 (Full-Speed)

- ・ダブルバッファ
- ・連続転送 (CNTMD='1')
- ・BFRE='1'
- ・転送パケット数=65540 (32770×2)

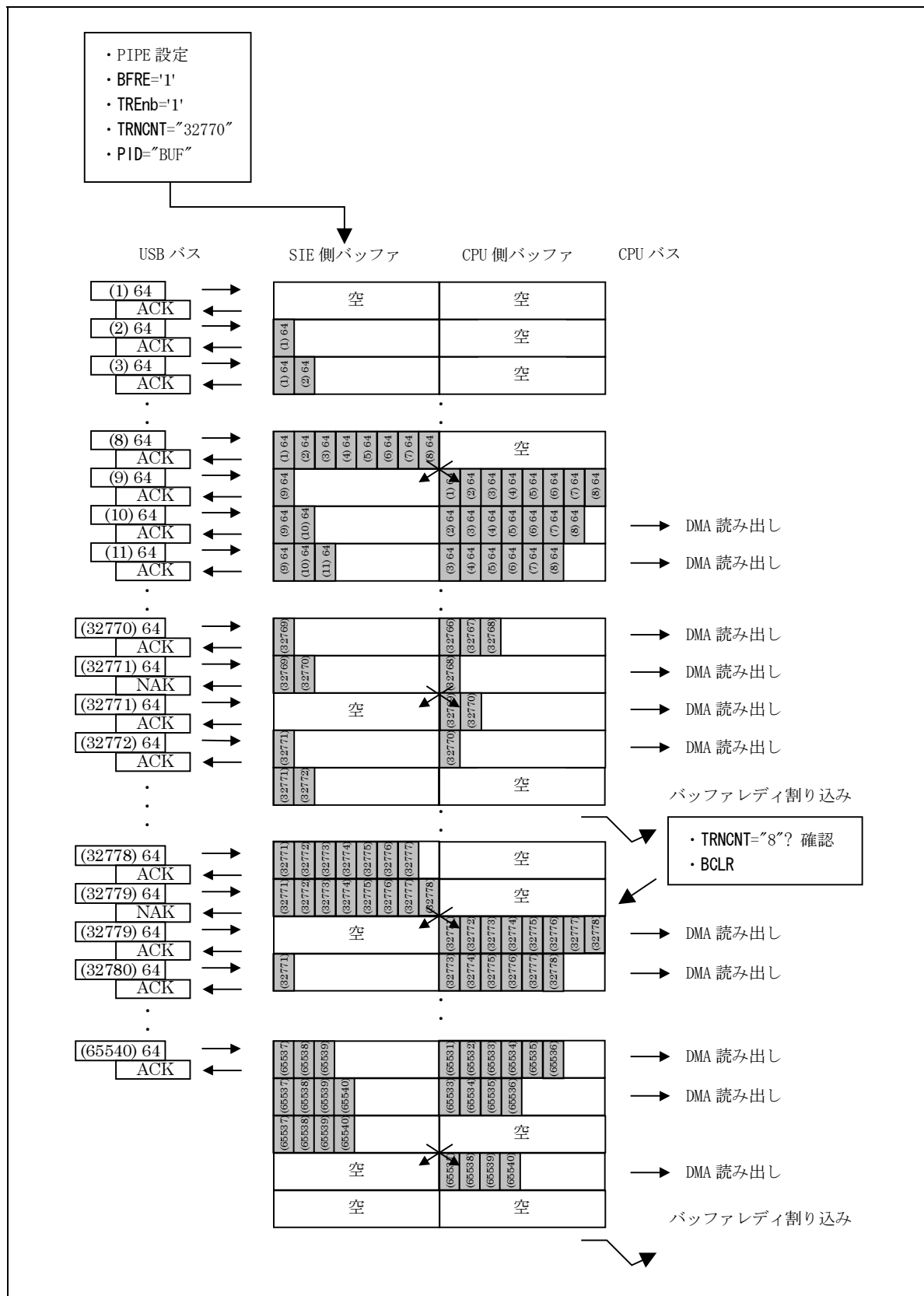


図5.16 トランザクションカウンタ最大値以上の場合 (Full-Speed)

5.6.9. FIFO ポートレディ (FRDY ビット「C_FIFO ポートコントロールレジスタ 1」 「D0_FIFO ポートコントロールレジスタ 2」)

これらのビットが'1'にセットされている時に、それぞれ C_FIFO ポートレジスタ 0、D0_FIFO ポートレジスタ 0 にアクセスが可能であることを示します。

(1) IN 方向の場合

シングルバッファの場合 : FRDY='1'でバッファが空または、書き込み途中であることを示します。バッファがフル状態になるまで'1'になります。

バッファフルで FRDY='0'になります。

ダブルバッファの場合 : FRDY='1'で少なくとも一方のバッファが空または、書き込み途中であることを示します。双方のバッファがフル状態になるまで'1'になります。

双方のバッファがフルで FRDY='0'になります。

(2) OUT 方向の場合

シングルバッファの場合 : FRDY='1'でバッファにデータがあり読み出し可能であることを示します。バッファから全てのデータを読み出すと'0'になります。

ダブルバッファの場合 : FRDY='1'で少なくとも一方のバッファに読み出し可能なデータがあることを示します。双方のバッファから全てのデータをを読み出すと'0'になります。

C_FIFO ポートレジスタ 0、D0_FIFO ポートレジスタ 0 に設定された Current_PIPE ビットを変更した場合、FRDY ビットの不定期間があります。詳しくはアプリケーションノート M6659x 共通編 2.4 章 FIFO 制御を参照ください。

5.6.10. シーケンスビットクリア (SQCLR ビット「PIPE_i コントロールレジスタ(i=1~4)」)

このビットにより、PIPE₁~PIPE₄のシーケンスビットをクリアし、次のデータ PID を"DATA0"にします。

シーケンスビットをクリアした以降の転送からは、H/W 制御によりシーケンスビットがトグルします。

なお、USB バスリセットでは、シーケンスビットはクリアされません。S/W によりシーケンスビットをクリアする必要があります。

Set_Configuration、Set_Interface、Clear_Feature リクエストを受けた場合には、このビットでシーケンスビットをクリアする必要があります。

このビットへの'0'書き込みは無視されます。このビットは常に'0'を読み出します。

このビットを設定する前に、PID を'00' (NAK) に設定してください。

6. スプリットバス

6.1. スプリットバスの特徴

M66591は、CPUバスとは非同期で動作する8bit スプリットバス(DMA Interface)を有します。スプリットバスはSD7-0/PA7-0端子に割り当てられます。スプリットバスを使用することにより、CPUを介在することなく、高速なデータ転送が可能です。

スプリットバスを使用したDMA転送と、CPUバスでのDMA転送との違いはDSTB_Nを使用することのみの違いです。

DBcfgビットの設定により、SD7-0/PA7-0の振る舞いを設定できます。

DBcfg='0' : SD7-0/PA7-0は汎用ポート(GPIO)として動作します

DBcfg='1' : SD7-0/PA7-0は、D0_FIFOポートに対するスプリットバスとして動作します

DBcfg='1' (スプリットバス) に設定した場合、D0_FIFOポートレジスタに対するCPUアクセスは無効になります。

また、スプリットバスを使用する際のRD/WRストロブ信号は”DSTB_N”になります。その際データピン&FIFO/DMA制御ピンコンフィギュレーションレジスタ2のRWstbビットを'1'に設定してください。DSTB_N信号のタイミングについてはM66591データシートを参照ください。

スプリットバスの接続例は図6.1を参照ください。

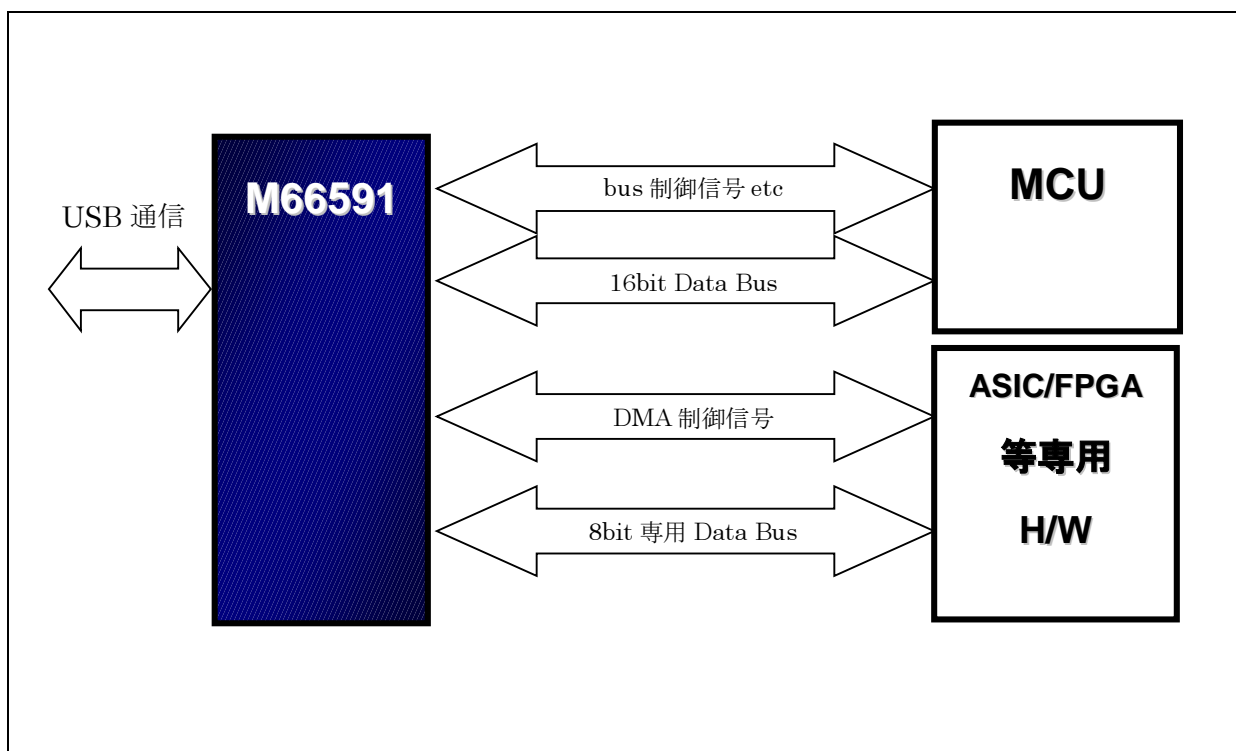


図6.1 スプリットバス接続例

6.2. スプリットバス DMA の設定例

以下表 6.1 に CPU バスでの DMA 設定と、スプリットバスでの DMA 設定例を示します。

表6.1 CPUバス／スプリットバスでのDMA設定例

	CPU DMA	Split DMA	設定例
D0_FIFOPortCtrl0(H'30)			
MBW	0	0	8bit幅
Current_PIPE	001	001	PIPE1
PipeCfgWin0(H'90)			
PEN	1	1	PIPE使用
DBLB	1	1	ダブルバッファ
CNTMD	1	1	Hi-Speedは"1"
DIR	1	1	IN転送
EP_NUM	001	001	EP1
PinCtrlCfg1(H'0A)			
DBcfg *1	0	1	0: CPUバス 1: Splitバス
PinCtrlCfg2(H'0C)			
DreqA	0	0	Lowアクティブ
Burst	1	1	バースト転送
DreqE	1	1	Dreq使用
DackA	0	0	Lowアクティブ
RWstb *1	0	1	0: WR0_N、RD_N 1: DSTB_N
DackE	1	1	Dack使用
DendA	0	0	Lowアクティブ
Pktmd	0	0	Transaction終了出力モード
DendE	1	1	Dend使用
Obus	0	0	高速駆動モード

*1: スプリットバス DMA 使用時の、CPU バス DMA との違いは、この 2 つのビットの設定です。
他のビットに関しては設定例ですので、システムに合わせた設定を行ってください。

7. USB バス状態出力

M66591は、USB バスの状態を外部に通知できる以下2本の出力端子を有します。

- ・ CONF_ON : USB バスが Configured ステートであるときに、CONF_ON 端子から”L”を出力します
- ・ SUSP_ON : USB バスが Suspended ステートであるときに、CONF_ON 端子から”L”を出力します

7.1. Configured 出力信号許可

CONFEN ビットにより、Configured 出力信号の許可/禁止が可能です。

- CONFEN='0' : Configured 出力信号を禁止します
- CONFEN='1' : Configured 出力信号を許可します

CONFEN='1'設定時、DVSQ ビットが'x11'(Configured ステート)の状態である間、CONF_ON 端子に”L”出力します。

以下図 7.1に CONF_ON 信号出力例を示します。

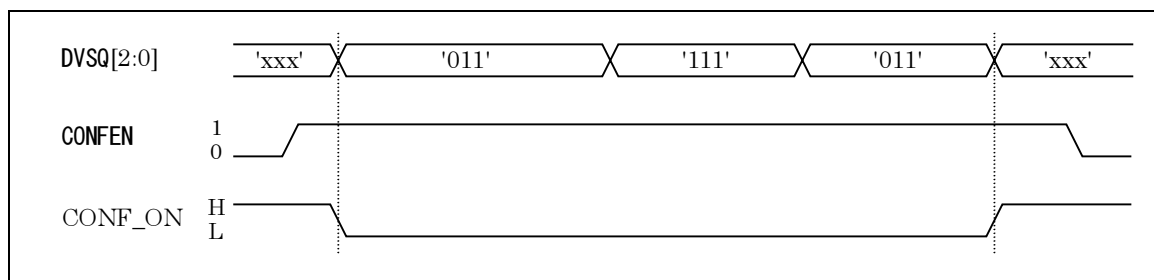


図7.1 CONF_ON 信号出力例

7.2. Suspend 出力信号許可

SUSPEN ビットにより、Suspend 出力信号の許可/禁止が可能です。

- SUSPEN='0' : Suspend 出力信号を禁止します
- SUSPEN='1' : Suspend 出力信号を許可します

SUSPEN='1'設定時、DVSQ ビットが'1xx'(Suspended ステート)の状態である間、SUSP_ON 端子に”L”出力します。

以下図 7.2に SUSP_ON 信号出力例を示します。

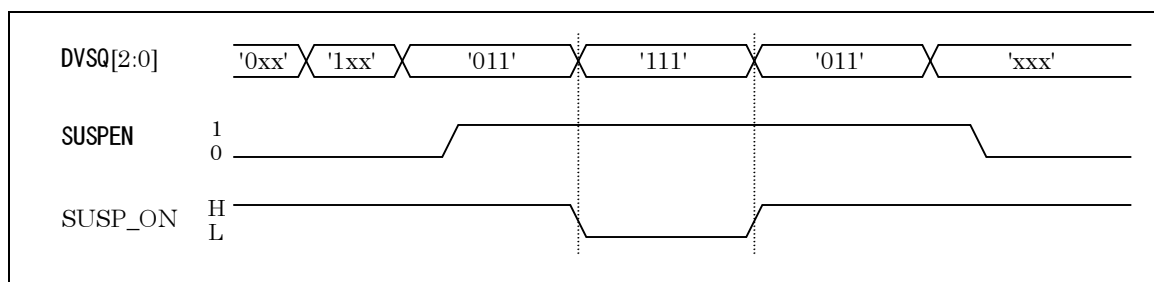


図7.2 SUSP_ON 信号出力例

7.3. USB 状態出力を使用したパワーマネージメント

USB 状態出力機能を使用することにより、消費電流が増大する Hi-Speed 通信時に、システムの電源を使用せず USB ラインからの電源を使用することが可能です。

バッテリー駆動製品等に最適であり、バッテリーの消費を抑えながら Hi-Speed 通信が可能です。

7.3.1. Vbus から引き出せる電流

USB の各デバイスステートにおいて、Vbus から引き出せる電流値を下記図 7.3 に示します。

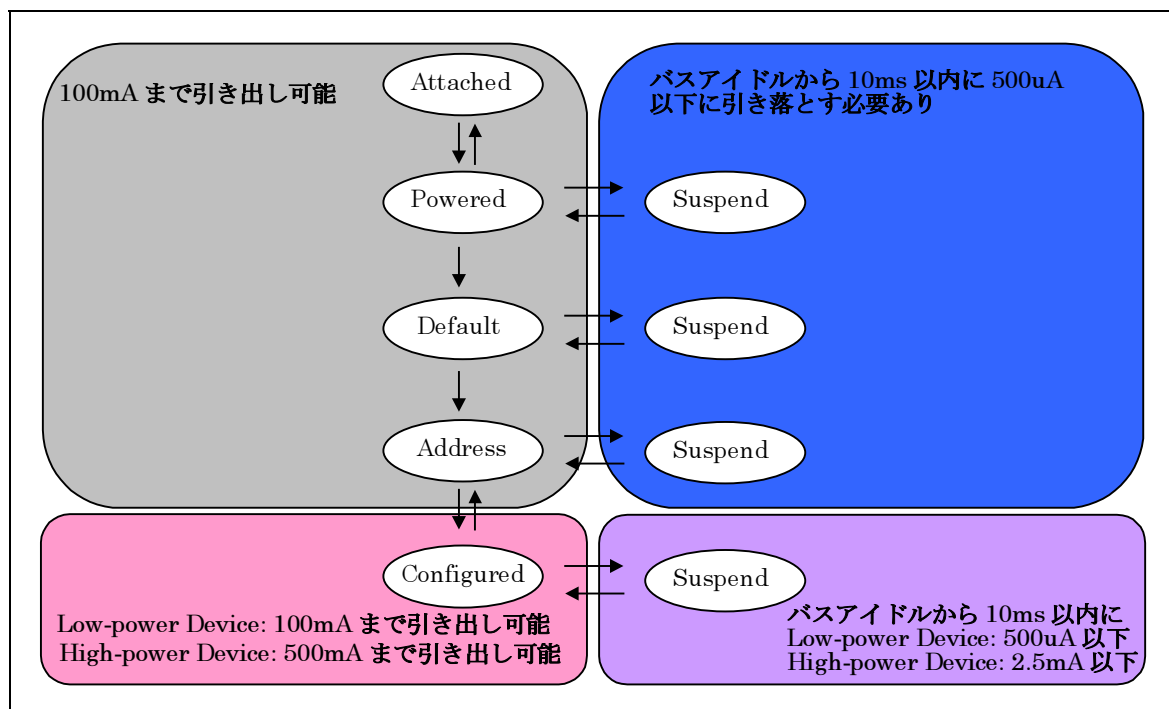


図7.3 Vbus から引き出せる電流値

7.3.2. USB 状態出力使用例ブロック図

以下図 7.4 に USB 状態出力使用例を示します。

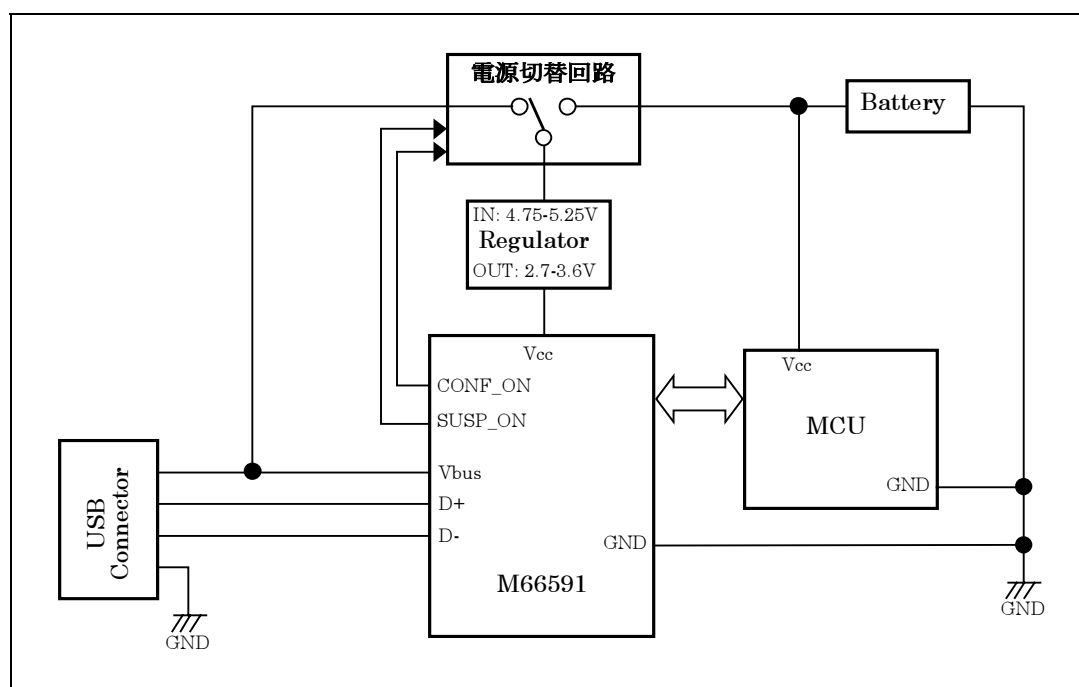


図7.4 USB 状態出力使用例

7.3.3. 電源切替回路例ブロック図

以下図 7.5に電源切替回路の例を示します。

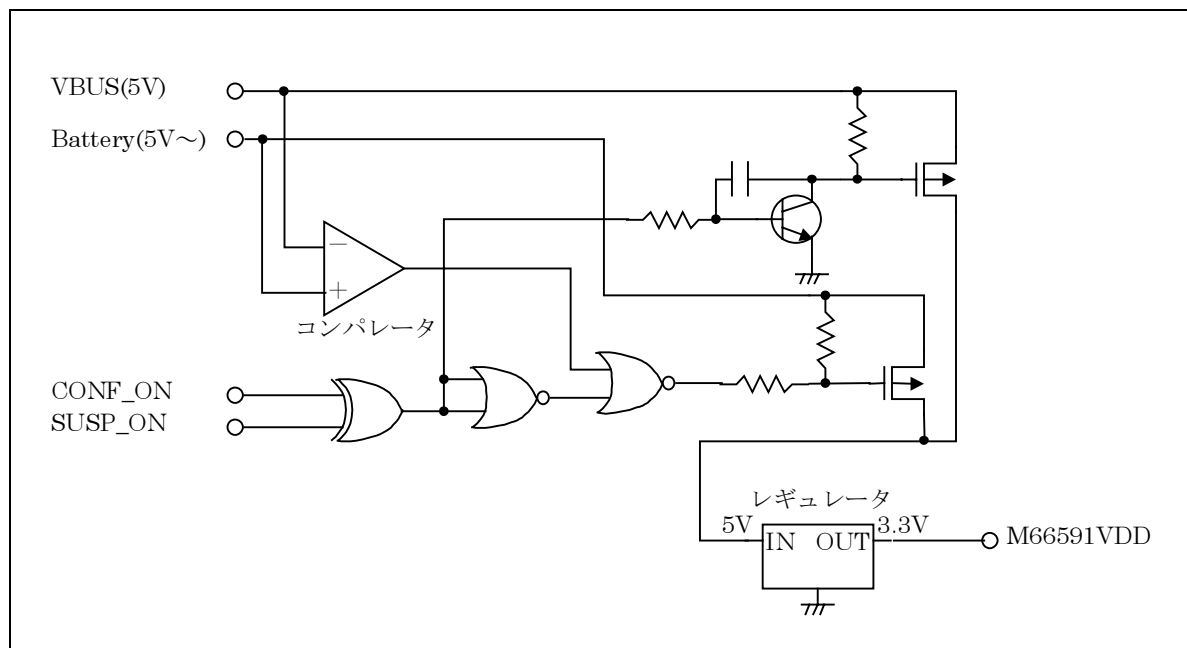


図7.5 電源切替回路例

7.3.4. USB 状態遷移と M66591 電源切替

以下図 7.6に USB 状態遷移と M66591 電源切替の例を示します。

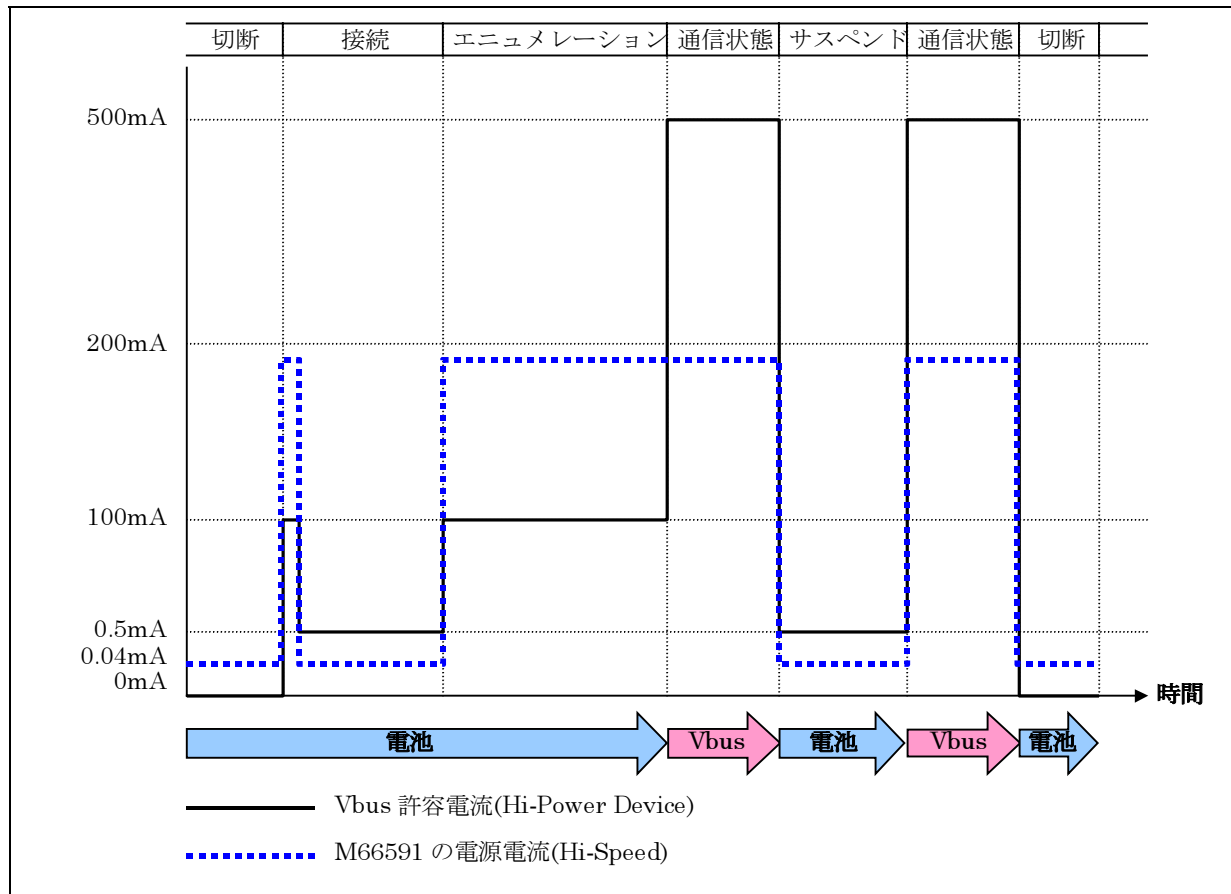


図7.6 USB 状態遷移と電源切替

7.4. 基板設計

基板設計に関する詳細は、別紙「基板作成ガイドライン」を参照ください。

改訂記録	M6659x アプリケーションノート M66591 編
------	--------------------------------

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2003.11.05	-	初版発行

安全設計に関するお願い

- ・ 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご留意ください。

本資料ご利用に際しての留意事項

- ・ 本資料は、お客様が用途に応じた適切なルネサス テクノロジー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ルネサス テクノロジーが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
- ・ 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の使用に起因する損害、第三者所有の権利に対する侵害に関し、ルネサス テクノロジーは責任を負いません。
- ・ 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全ての情報は本資料発行時点のものであり、ルネサス テクノロジーは、予告なしに、本資料に記載した製品または仕様を変更することがあります。ルネサス テクノロジー半導体製品のご購入に当たりましては、事前にルネサス テクノロジー、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジーホームページ (<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。
- ・ 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。万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジーはその責任を負いません。
- ・ 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位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。ルネサス テクノロジーは、適用可否に対する責任を負いません。
- ・ 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジー、ルネサス販売または特約店へご照会ください。
- ・ 本資料の転載、複製については、文書によるルネサス テクノロジーの事前の承諾が必要です。
- ・ 本資料に関し詳細についてのお問い合わせ、その他お気づきの点がございましたらルネサス テクノロジー、ルネサス販売または特約店までご照会ください。