『組み込みシステム開発評価キット』におけるブロックRAM活用例

Appendi 2

Ethernet & USBターゲット& MMCカード・コントローラのバッファ

山武 一朗

●『組み込みシステム開発評価キット』におけるブロックRAM活用例

『組み込みシステム開発評価キット』では，Ethernet（10Base-T，100Base-TX）やUSBターゲット，さらに別売のソケット・コネクタ基板を実装することでMMCカードにも対応しています．これらのコントローラではFPGAの内蔵ブロックRAMを活用したバッファを設けて，FPGAに実装したソフトCPUコアなどとのデータ転送を円滑に行えるようにしています．

● Ethernetコントローラにおけるブロック RAM

本評価キットでは，Ethernetコントローラを I/Oプロセッサと呼ぶSpartan-3/400に実装します．このFPGAには後述するUSBターゲット機能なども実装するので，内蔵のブロックRAMをどのモジュールにどの程度割り当てるかは，システム設計者次第です．

図1に本評価キットに添付されているEthernetコントローラのブロック図を示します．この設計では，Ethernetパケット1個分（最大1500バイト）を保持できるようSpartan-3のブロックRAMを送信バッファに1個（2Kバイト），受信バッファに8個（16Kバイト）実装しました．

この受信バッファの例のようにブロックRAMを複数個組み合わせて容量の大きなメモリ・モジュールを実現するには，アドレスの上位ビットをデコードしてチップ・セレクト信号を作成し，それぞれのブロックRAMを選択させ，データ・バスを切り替える回路を実装します．例として，**リスト1**に受信バッファ用に用意した，データ・バス幅32ビット，容量16Kバイトのデュアル・ポートRAMモジュールの例を示します．

なお，FPGAによるEthernetコントローラの設計の詳細については，参考文献(1)の第15章を参照してください．

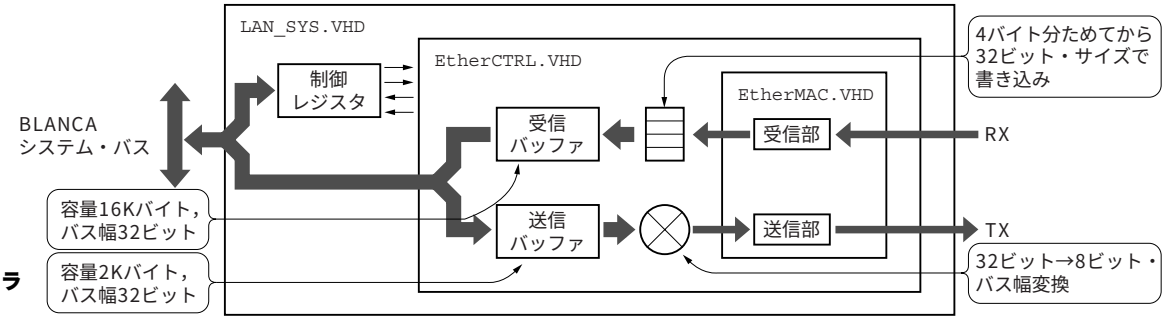

図1 Ethernetコントローラのブロック図

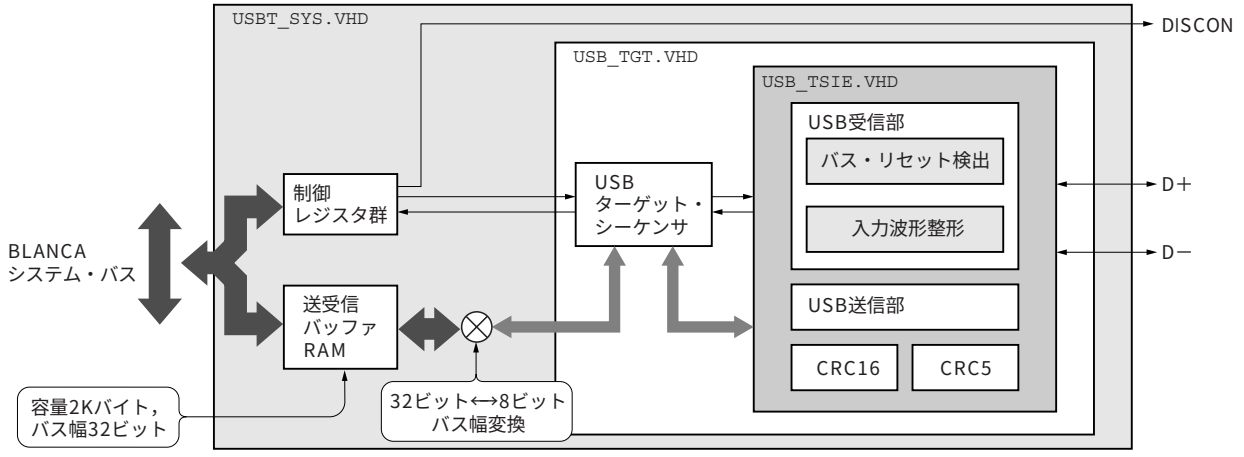

図2 USBターゲット・コントローラのブロック図

KEYWORD——FPGA内蔵ブロックRAM，送信バッファ，受信バッファ，デュアル・ポートRAM

● USB ターゲット・コントローラにおけるブロック RAM

本評価キットでは，USB ターゲット・コントローラを I/O プロセッサと呼ぶ Spartan-3/400 に実装します．**図 2** に本評価キットに添付されている USB ターゲット・コントローラのブロック図を示します．この設計ではブロック RAM を一つだけ組み込んでいます．将来的にダブルバッファなどを構成できるように一つ当たり最大 256 バイトまで使えるエンドポイント用バッファを構成しています．合計でエンドポイントを四つ，さらにコントロール・エンドポイントとディスクリプタ情報をブロック RAM に格納しているので，合計メモリ容量は 1.25K バイトとなります．これを Spartan-3 のブロック RAM 一つに格納しています．

なお，FPGA による USB ターゲット・コントローラの設計の詳細については，参考文献(2)の第 5 章を参照してください．

● MMC カード・コントローラにおけるブロック RAM

本評価キットでは，MMC カード・コントローラを A/V プロセッサと呼ぶ Spartan-3/1500 に実装します．本評価キットを使う場合，この FPGA に MicroBlaze や M32R ソフト・コアなどのソフト CPU コアを同時に実装します．それらの CPU モジュールでもブロック RAM を多用するので，もし足りなくなった場合は CPU コアのキャッシュ容量の設定などを変更してブロック RAM の使用量を減らすなどの調整が必要です．

図 3 に，本評価キットに添付されている MMC カード・コントローラのブロック図を示します．MMC カードは 1 セクタ 512 バイトなので，最小 512 バイトの容量があれば足りるのですが，将来的にリード/ライトで独立してバッファを持たせ，しかもそれぞれデュアルバッファ構造にすることで性能の向上が図れるように，2K バイトとしました．

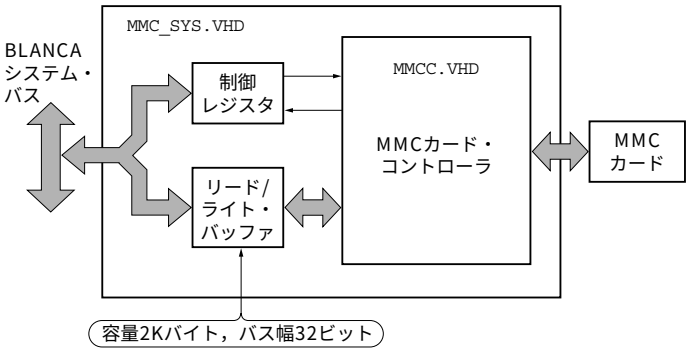

図 3　MMC カード・コントローラのブロック図

なお，FPGA による MMC カード・コントローラの設計の詳細については，参考文献(3)の第 6 章を参照してください．

参考文献

(1) Ethernet のしくみとハードウェア設計技法，TECH I Vol.32，CQ 出版社，2006 年.
(2) 山武 一朗；FPGA による USB ターゲット・コントローラの設計事例，Interface，2007 年 1 月号，pp.96-111.
(3) フラッシュ・メモリ・カードの徹底研究，TECH I Vol.35，CQ 出版社，2006 年.

やまたけ・いちろう　来栖川電工(有)

リスト 1　デュアルポート RAM モジュール(容量 16K バイト)

```
-- ********************************************************* --
-- ***** FPGA 内蔵ブロック RAM　デュアルポート RAM 化 モジュール
-- ***** ポート A　データ 32 ビット/アドレス 12 ビット　16384 バイト
-- ***** ポート B　データ 32 ビット/アドレス 12 ビット
-- ********************************************************* --
library ieee;
use ieee.std_logic_1164.all;

entity DualRAM32_32_16K is
    port (
    -- ポート A バス信号 --
         Clock_A        : in std_logic;
         Address_A      : in std_logic_vector(13 downto 2);
         ByteEnable_A   : in std_logic_vector( 3 downto 0);
         ReadData_A     : out     std_logic_vector(31 downto 0);
         WriteData_A    : in std_logic_vector(31 downto 0);
         WriteEnable_A  : in std_logic;
    -- ポート B バス信号 --
         Clock_B        : in std_logic;
         Address_B      : in std_logic_vector(13 downto 2);
         ByteEnable_B   : in std_logic_vector( 3 downto 0);
         ReadData_B     : out     std_logic_vector(31 downto 0);
         WriteData_B    : in std_logic_vector(31 downto 0);
         WriteEnable_B  : in std_logic
    );
end entity DualRAM32_32_16K;

architecture RTL of DualRAM32_32_16K is

-- ***** Spartan3 内蔵ブロック RAM モジュール定義 ***** --
component RAMB16_S36_S36
    port (
    -- ポート A
         WEA       : in std_logic;          -- ライト・イネーブル
         ENA       : in std_logic;          -- イネーブル
         SSRA      : in std_logic;          -- リセット
         CLKA      : in std_logic;          -- クロック
         -- アドレス・バス
         ADDRA     : in std_logic_vector( 8 downto 0);
         -- 入力データ・バス
         DIA       : in std_logic_vector(31 downto 0);
         -- 入力パリティ
         DIPA      : in std_logic_vector( 3 downto 0);
         -- 出力データ・バス
         DOA       : out     std_logic_vector(31 downto 0);
         -- 出力パリティ
         DOPA      : out     std_logic_vector( 3 downto 0);
    -- ポート B
         WEB       : in std_logic;          -- ライト・イネーブル
         ENB       : in std_logic;          -- イネーブル
         SSRB      : in std_logic;          -- リセット
         CLKB      : in std_logic;          -- クロック
         -- アドレス・バス
         ADDRB     : in std_logic_vector( 8 downto 0);
         -- 入力データ・バス
         DIB       : in std_logic_vector(31 downto 0);
         -- 入力パリティ
         DIPB      : in std_logic_vector( 3 downto 0);
         -- 出力データ・バス
         DOB       : out     std_logic_vector(31 downto 0);
         -- 出力パリティ
         DOPB      : out     std_logic_vector( 3 downto 0)
    );
end component;

    -- ***** ブロック RAM チップ・セレクト ***** --
```