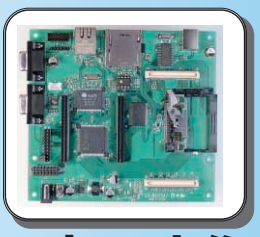
大変長らくお待たせしました！

本誌付属 CPU 基板にネットワークとストレージ機能を追加する

SH-2 & V850 付属基板対応 拡張ベース・ボードの設計（中編）



山武 一朗/吉田 幸作

今回は CPLD 内に実装したステート・マシンや，各種コントローラと CPU との接続部分について解説する．SH-2 と V850 はそれぞれエンディアンが異なるが，それでも拡張ベース・ボードは共通なものを使えるように工夫している．（編集部）

前回（2007 年 12 月号，pp.128-134）に続き，中編の今回は，CPLD 内に実装した制御回路について，少し詳しく解説します．

1．CPLD に実装したアドレス・デコーダやステート・マシン

● SH-2 と V850 両方の対応化

設計した拡張ベース・ボードは，搭載する CPU をジャンパの設定で選択できるように設計しています．**写真 1** に付属 CPU 基板をスタック接続した様子を示します．接続可能な CPU は現状は 2 種類なので，ジャンパの設定は 1 本で選択可能ですが，もう一つは CPU として SH-2 を選択したとき，CPU 基板裏面に SRAM を増設した場合と増設していない場合を選択するジャンパとしています．付属基板の状態では CPU 基板裏面には SRAM は実装されていませんが，イーエスピー企画（http://www.esp.co.jp/）より発売されているリピート基板では SRAM が実装されているためです．

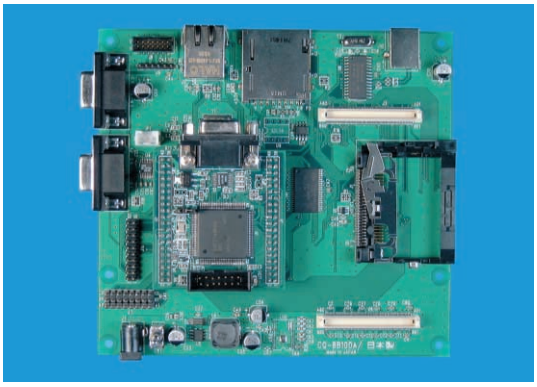
● CPU 選択部とアドレス・デコード

リスト 1 に CPU 選択とアドレス・デコード処理部を示します．SH-2 の場合は $\overline{CS0}$ ～ $\overline{CS3}$ までのチップ・セレクト信号 4 本を使って，SRAM 接続空間（二つ）と 16 ビットおよび 8 ビット I/O 空間を選択します．CPU 基板の裏面に SRAM を増設している場合は，SRAM 空間がぶつからないように拡張ベース・ボード上の SRAM を $\overline{CS3}$ にマッピングしています．V850 を接続した場合はアドレス・バスの A21 が $\overline{CS1}$ に，A20 が $\overline{CS0}$ に配線されるので，この 2 ビットを使って三つの空間をデコードさせます．

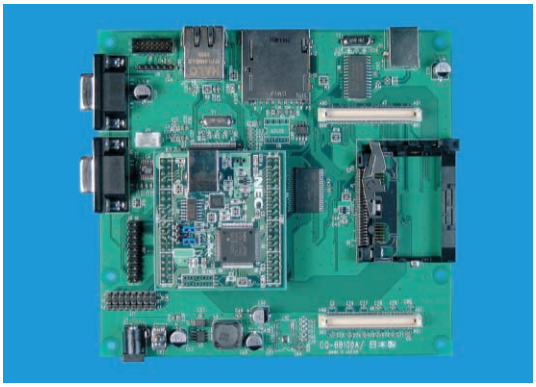
また I/O 空間のベース・アドレスが決まったら，アドレス・バスの A15 ～ A12 までを使って LAN コントローラ，CF カード・コントローラ，MMC カード・コントローラ，USB コントローラのベース・アドレスをデコードしています．

● 周辺コントローラ用制御信号生成部

リスト 2 に CPU ローカル・バスおよび周辺コントローラ用の制御信号生成部を示します．CPU ローカル・バスで制御が必要なのは，CPU が読み出し動作をしたときの



（a）SH-2 搭載時



（b）V850 搭載時

写真 1 付属 CPU 基板をスタック接続した様子

KEYWORD──SH-2，V850，CPU 選択，アドレス・デコーダ，ローカル・バス，ステート・マシン，エンディアン，ISA バス・モード，186 モード，68K モード，AX88796L，USBN9604

リスト1　CPU選択とアドレス・デコード処理

```
-- ***** SH-2 or V850 CPU選択 ***** --
CPUselCtrl : process(CPU_nSL, CPU_nCS)
begin

    if (CPU_nSL(0) = '0') then  -- SH-2
        -- ***** エンディアン設定 ***** --
        LITTLE_ENDIAN <= 0;      -- リトル・エンディアンのとき1，ビッグ・エンディアンのとき0
        -- ***** SH-2用チップ・セレクト・セレクタ ***** --
        if (CPU_nSL(1) = '1') then
            ChipSelect(0) <= not CPU_nCS(1);-- CS1 0040_0000h SRAMメモリ空間(16ビット)
        else
            ChipSelect(0) <= not CPU_nCS(3);-- CS3 00C0_0000h SRAMメモリ空間(16ビット)
        end if;
        ChipSelect(1) <= not CPU_nCS(0);  -- CS0 0020_0000h 周辺コントローラ空間(16ビット)
        ChipSelect(2) <= not CPU_nCS(2);  -- CS2 0080_0000h 周辺コントローラ空間(8ビット)
        -- ↑CS0～2を順番に使いたいところだが，付録基板ではCS1にSRAMをマッピングしたため
        -- それとの互換性を考慮してCS1をSRAM空間とした
        CFMEM_WAIT     <= "11010";  -- 28-1クロック・ウェイト
        CFIO_WAIT <= "00111";       -- 8-1クロック・ウェイト

    else                            -- V850
        -- ***** エンディアン設定 ***** --
        LITTLE_ENDIAN <= 1;      -- リトル・エンディアンのとき1，ビッグ・エンディアンのとき0
        -- ***** V850用アドレス・デコーダ ***** --
        if (CPU_nCS(1 downto 0) = "01") then -- ブロック0 0010_0000h SRAMメモリ空間(16ビット)
            ChipSelect(0) <= '1';
        else
            ChipSelect(0) <= '0';
        end if;
        if (CPU_nCS(1 downto 0) = "10") then -- ブロック1 0020_0000h 周辺コントローラ空間(16ビット)
            ChipSelect(1) <= '1';
        else
            ChipSelect(1) <= '0';
        end if;
        if (CPU_nCS(1 downto 0) = "11") then -- ブロック2 0070_0000h  周辺コントローラ空間(8ビット)
            ChipSelect(2) <= '1';
        else
            ChipSelect(2) <= '0';
        end if;
        CFMEM_WAIT     <= "01011";  -- 12-1クロック・ウェイト
        CFIO_WAIT <= "00010";       -- 3-1クロック・ウェイト

    end if;
end process CPUselCtrl;

-- ***** 拡張ベースボード・アドレス・デコーダ ***** --
RAM_Select    <= '1' when ( ChipSelect(0) = '1') else '0';
REG_Select    <= '1' when ( ChipSelect(1) = '1'                          and CPU_A(15 downto 14) = "00" ) else '0';-- +0000h～
LAN_Select    <= '1' when ((ChipSelect(1) = '1' or ChipSelect(2) = '1') and CPU_A(15 downto 13) = "010") else '0';-- +4000h～
USB_Select    <= '1' when ( ChipSelect(1) = '1'                          and CPU_A(15 downto 13) = "011") else '0';-- +6000h～
CFC_Select    <= '1' when ((ChipSelect(1) = '1' or ChipSelect(2) = '1') and CPU_A(15) = '1'             ) else '0';-- +8000h～
-- ↑LANとCFコントローラには、16ビット空間と8ビット空間の両方をマッピングしている

-- ***** CFカード各種空間アドレス・デコーダ ***** --
CF_AtrMem     <= '1' when (CPU_A(13 downto 12) = "00") else '0';       -- +8000h/C000h(データ・バス上位下位逆接続空間)
CF_ComMem     <= '1' when (CPU_A(13 downto 12) = "01") else '0';       -- +9000h/D000h(データ・バス上位下位逆接続空間)
CF_Io         <= '1' when (CPU_A(13 downto 12) = "10") else '0';       -- +A000h/E000h(データ・バス上位下位逆接続空間)
-- ↑アドレス・バスA14を使って，データ・バス通常接続空間と上位下位逆接続空間の2空間を用意
CF_Memory     <= CF_AtrMem or CF_ComMem;   -- CFカード・メモリ空間選択信号
```

データ・バスのドライブ制御と，アクセス・タイミングを制御する$\overline{\text{WAIT}}$信号です．

SH-2とV850のどちらも，バスのアクセス・タイミングを延ばすときには周辺回路が$\overline{\text{WAIT}}$信号に"L"レベルを出力します．拡張ベース・ボード上のリソースでアクセス・タイミングが固定されていないのは，LANコントローラとCFカード・コントローラです．LANコントローラからのウェイトは，LANコントローラの空間をアクセスしているときだけLANコントローラからのウェイト信号(AX_RDY)によってCPUにウェイトを要求します．CFカード・コントローラからのウェイトは，CPLD内部のステート・マシンで制御するウェイト信号を論理反転してCPUに出力しています．

また，拡張ベース・ボード上に実装しているSRAMは，データ・バス幅を16ビットで接続するので，バイト単位でアクセスしたときにそれが上位バイトへのアクセスなのか下位バイトへのアクセスなのかを示すSRAM_nLB信号やSRAM_nUB信号をCPLD内部で生成します．