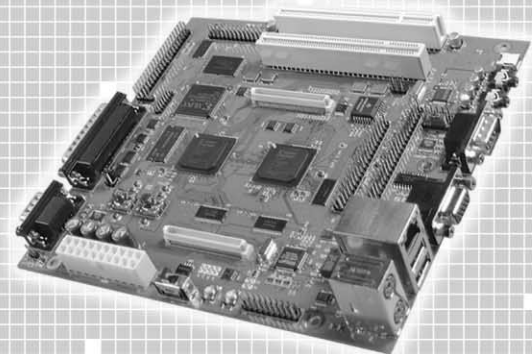


組み込みシステム 開発評価キット 活用通信



井倉 将実

第1回 これがBLANCAシステム・アーキテクチャだ

先月号(本誌2006年10月号)まで連載されていた「コンピュータ・システム技術学習キット活用通信」は、今月号からタイトルを一新し、新連載としてスタートします。連載タイトルは変わりますが、中身はこれまでと同じように、本評価キットの活用方法について解説していきます。

これまででは、MicroBlazeやM32Rソフト・コアなど、ソフトCPUコア側に焦点を当てて解説を進めてきました。今回からは、システム・アーキテクチャ全体について解説していきます。

1. BLANCAシステム・バスとは

● BLANCA って...何?

新連載を始める前の素朴な疑問として、以前の連載でもしばしば名前が登場している“BLANCA”とは何でしょうか?

これは、本評価ボードのコード・ネームともいうべきもので、開発者グループ内でそう呼んでいたものです。スペイン語で“白色”を意味することばで、真っ白なキャンバスに自由に絵を描くがごとく、まさるなFPGAに自由にシステムを組み込んでほしいという意味が込められています^{注1}。

● もっとも簡単な SoC 内部バス

BLANCA は、

- シングル・マスタ
- マルチ・スレーブ
- シングル・ワード転送専用

という、これ以上は仕様を削れないほど簡単なシステム・バス

を採用しています。

表1にBLANCAシステム・バスの信号を示します。すべてデバイス内部の信号なので、正論理(‘1’で有効状態を示す)で規定されています。

マスタとは、CPUやDMAコントローラなど、バス・アクセスをみずから開始できるリソースを示します。とはいえ、BLANCAシステム・バスはシングル・マスタ仕様なので、本バス上でマスタになれるのはCPUしか存在しません。逆にスレーブは、マスタからアクセスされる側のリソースです。

システム・クロックは、原クロックをCPUモジュールに入力した後、内部で倍/位相調整などを行った後、モジュールから出力される場合と、原クロックをCPUモジュールとスレーブ・モジュールに入力する場合の2とおりがあり、CPUコアの形態によって選択する必要があります。

CPUコアの仕様によっては、システム・リセットはCPUコアに入力されるだけでなく、CPUコアから出力される可能性も考えられます。その場合は、CPUコアが出力するシステム・リセット信号をBLANCAシステム・バスのシステム・リセット信号に接続します。

アドレス・バスはA31～A0と、きっちり32本用意されています。また、データ・バス幅は32ビットなので、バイト・イネーブルも4本あります。

リード/ライト信号は、リード時に‘1’、ライト時に‘0’を示します。

また、SoCやFPGAなど、デバイスの内部で使用するバスを

表1
BLANCAシステム・バスの信号一覧

信号名	入出力方向		信号 ビット数	信号の意味
	マスタ (CPU)	スレーブ		
SYS_Clock	出力(入力)	入力	1	システム・クロック
SYS_Reset	入力(出力)	入力	1	システム・リセット
SYS_Address	出力	入力	32	アドレス・バス
SYS_ByteEnable	出力	入力	4	バイト・イネーブル
SYS_ReadWrite	出力	入力	1	リード/ライト
SYS_AccStart	出力	入力	1	アクセス・スタート
SYS_WriteData	出力	入力	32	ライト・データ・バス
SYS_ReadData	入力	出力	32	リード・データ・バス
SYS_AccReady	入力	出力	1	アクセス・レディ

注1：筆者が飼っている猫の名前という説もある。

想定し、リード・データ・バスとライト・データ・バスを独立して用意しています。

そして、バス・アクセスの要となるアクセス・スタート信号とアクセス・レディ信号があります。

BLANCA システム・バスは、システム・クロックの立ち上がりタイミングに従う同期バスになっています。システム・リセットも正論理なので、たとえば VHDL の場合は、

```
if (SYS_Reset = '1') then
  (リセット時の処理)
elsif (SYS_Clock'event and
       SYS_Clock = '1') then
  (通常時の処理)
end if;
```

のように記述します。

● BLANCA システム・バスのタイミング

図1に BLANCA システム・バスのタイミング例を示します。

まず、クロックの段階でマスタがバス・アクセスを開始するために、アドレス・バスやバイト・イネーブル、リード/ライト方向を出力し、またライト・アクセス時であればライト・データも出力し、そしてアクセス・スタート信号 SYS_AccStart を '1' にします。アクセス・スタート信号のアサート期間は1クロックのみです。

クロック同期回路なので、各スレーブ・デバイスがアクセス・スタート信号を判定するのは次のクロックの段階です。アドレス・デコードによって選択されたスレーブ・デバイスは、内部でさらにレジスタの選択などが行われ、ライト・アクセス時ならライト・データ・バスの値を内部に取り込みます。ライト・データを取り込んだら、アクセス・レディ信号 SYS_

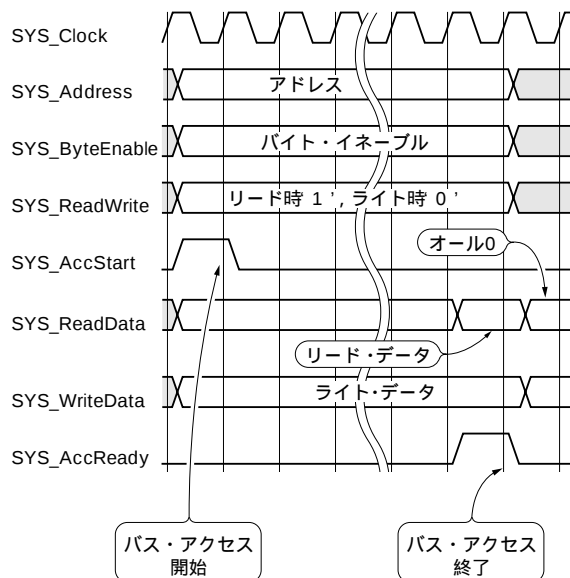


図1 BLANCA システム・バスのタイミング

AccReady を '1' にします。アクセス・レディ信号のアサート期間も1クロックのみです。

リード・アクセス時には、選択されたレジスタの値をリード・データ・バスに出力し、アクセス・レディ信号 SYS_AccReady を '1' にします。アクセス・レディ信号のアサート期間も1クロックのみなので、次のクロックの段階で '0' に戻すと同時に、リード・データ・バスの値もすべて 00h にする必要があります。これは後述するように、各スレーブ・デバイスのリード・データ・バスは、ワイヤード OR 接続によりマスタ・デバイスに接続されているためです。

● バス・エラーがない、ウェイトはひたすら待つのみ

このように非常に簡単な仕様のバスなので、アクセスした先にデバイスがなかった場合などのアドレス・エラーや、アクセス・レディ信号が一定時間以内に返ってこなかった場合のバス・エラーなどのしくみは用意されていません。したがって PCI バスや IDE など、ウェイトが要求されるリソースの場合は、アクセス・レディ信号が '1' になるまでひたすら待つことになります。

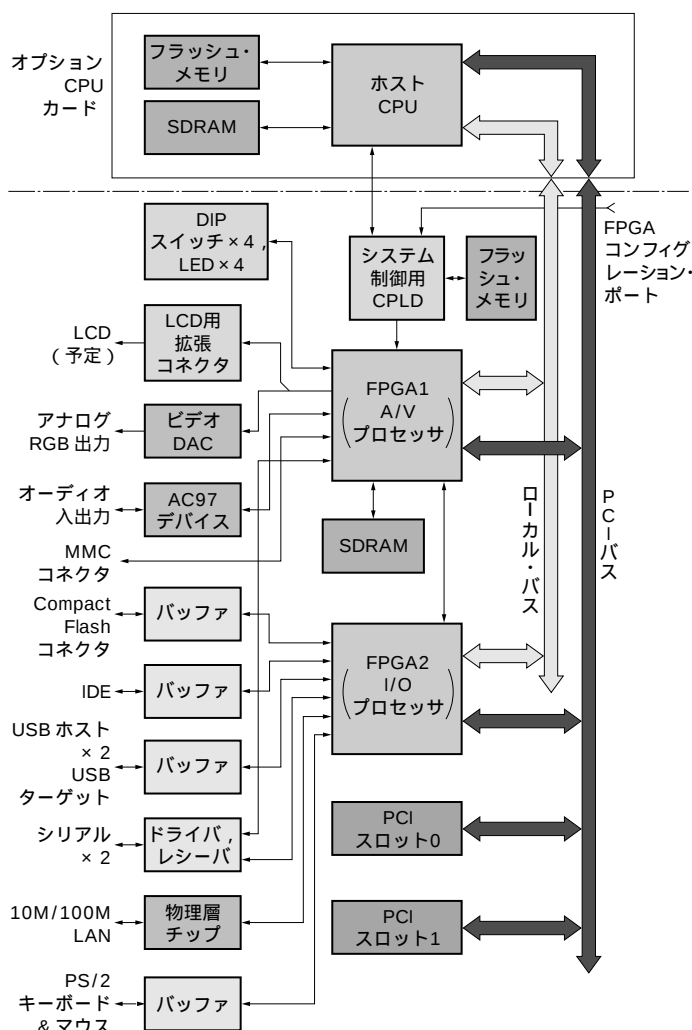


図2 組み込みシステム開発評価キットのブロック図