次世代LSI開発ソリューション

bluespec™

FAST, EARLY AND ACCURATE

複雑化、加速化するLSI開発に革新を!
CAEのサイバネットが提案する
ASIC/FPGAモデリングツールの決定版

CYBERNET つくる情熱を、支える情熱。

RTLの限界、浸透しないESL。
LSI設計にブレイクスルーを。

近年、製品の高機能化はとどまることなく、システムの複雑さが増大の一途をたどっています。この高機能化により、ASIC/FPGAの開発には新たな手法が求められています。新たな手法として、RTL記述の抽象度を上げシステムレベルでの設計が行われていますが、なかなか浸透していないのが現状です。これは、H/W設計者には馴染み難いC言語ベースという理由もありますが、制御系の回路では設計者の意図を詳細に反映したH/W化が非常に煩雑であることが大きな理由です。RTLとシステムレベル、この間の抽象度ギャップは非常に大きく、従来の手法ではその差を埋めることはとても大変です。システムレベルからRTLへのH/Wトランスペアレンシーを保証する、RTL設計者のための新たな手法が鍵となるのではないでしょうか。そこで開発されたのがBluespec System Verilog(BSV)です。BSVはMIT(Massachusetts Institute of Technology)で開発された革新的な言語であり、RTLよりも抽象度が高く、完全な並列性が表現できます。また、高速なシミュレーションとリファイメント性も兼ね備えています。BSVで記述されたモデルは、全て合成可能であり、システムレベル記述からのH/W実装を実現することが可能です。つまり、BSVはRTLとシステムレベルのギャップをH/Wの設計思想をベースにした実行可能なモデルでうめることにより、大規模・複雑化するLSI開発の救世主となります。

Bluespecのポリシー

優秀な設計者を退屈な作業から解放し、創造的な設計業務に集中させる

マイクロアーキテクチャ

創造性があり、楽しい
デザインの優劣を左右する → 設計者が制御

並列性の管理、制御など

面倒で退屈
エラーが発生しやすい
誰がやっても差が出にくい → コンパイラが自動化

Bluespec System Verilogの特徴

Bluespec Core tool

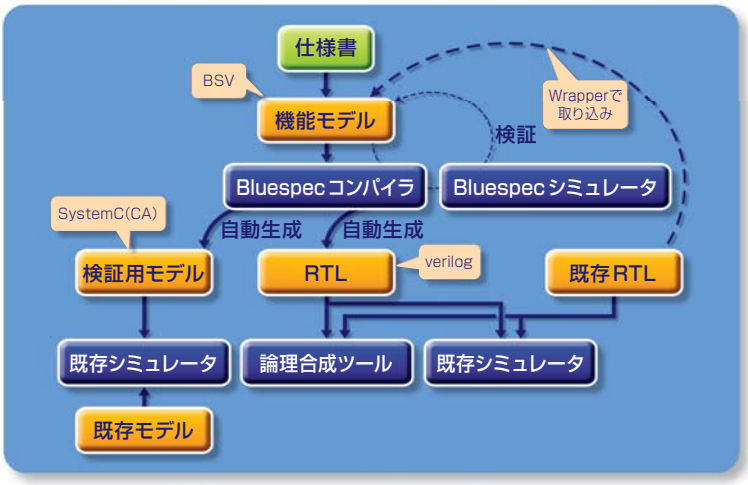
Bluespec コンパイラ

- 入力言語は、BSV(Bluespec System Verilog)
- 論理合成可能なverilog RTLを合成
- BSVで書かれたテストベンチは、RTLへ合成可能
- 設計制約ファイルは不要
- CAレベルのSystemCを合成可能

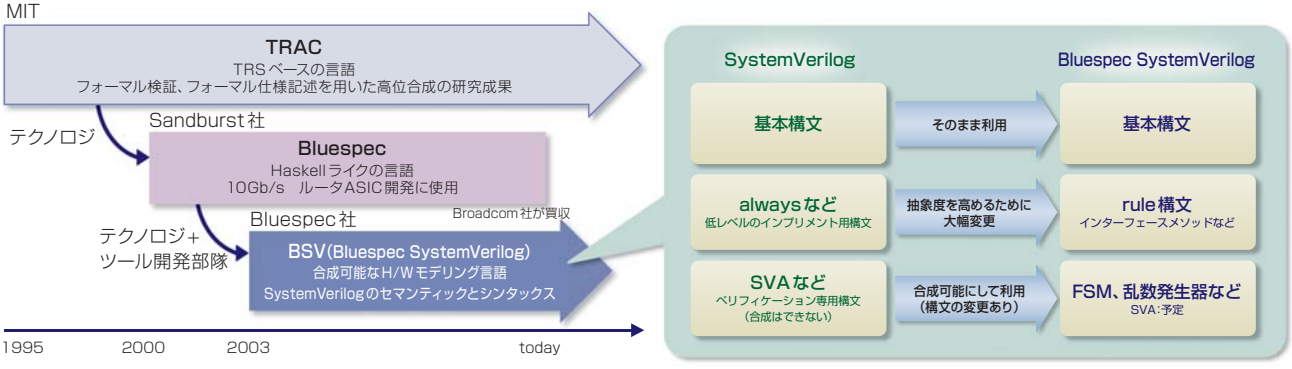
Bluespec シミュレータ

- BSVで記述されたモデルを検証するためのシミュレータ
- サイクルベース
- RTLシミュレータの3～10倍高速

Bluespecツールの設計フロー



BSVの位置づけ



HDLからの移行ソリューション

従来のverilog-HDLやVHDLなどのHDLによるASIC/FPGAの開発では、まず仕様書からアーキテクチャを意識して詳細化しながら、実装仕様書を作成します。次に実装仕様書をもとにRTLのコーディング(実装)を行い、実装の確からしさを検証します。これは、仕様書から実装仕様書、実装仕様書からRTLへと詳細化を行いながら、抽象度のギャップを人手で埋めていっているに他なりません。したがって、大規模化、高機能化が進めば進むほど、作業は膨大となります。
RTLの検証は、ギャップを埋めた結果が、正しいかを検証していることであり、さらに検証に多くの時間を費やす原因となっています。

BSVによる開発では、次の手順ような手順となります。
(1)モデルによる仕様の表現/定義「実行可能な仕様書」を作成する
(2)モデルのシミュレーションにより設計の詳細化、妥当性検証を行う
(3)モデルから自動コード生成により実装

BSVによるモデル作成は、ruleという構文を使用し、条件ごとの動作を記述します。
BSVの特徴として、以下のものがあげられます。
(1)ruleは、全て並列動作
(2)条件変更や追加が容易
(3)BSVで記述されたモデルやテストベンチは、全て合成可能
(4)オブジェクト指向(モデルのアクセスは、method)
(5)並列性の管理や競合の排他制御は、コンパイラが自動生成

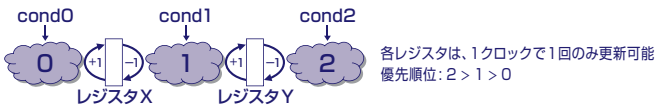
導入効果

デザイン	設計・検証の工数		効率化
	RTL	Bluespec	
マルチチャンネル OCP DMA コントローラ	7人月 (4:設計・単体検証、3:全体検証)	3.75人月 (2.75:設計・単体検証、1:全体検証)	1.9X
プログラマブル コントローラ	12人月	6人月 (5:設計・単体検証、1:全体検証)	2.0X
HD H.264 デコーダ	24人月 (見積もり)	10人月 (設計・単体検証・FPGA)	2.4X

これらの特徴により、非常に効率的にモデルを作成できるうえ、強力なパラメタライズ機能があるため、作成したモデルは、効率的な再利用が可能です。また、BSVはクロックやリセットを含め全てが型を持っており、コンパイラは非常に厳密な型チェックを行います。

違う型同士の代入は全てエラーとなるため、ケアレスミスによる代入ミスやクロックドメイン間をわたる代入を排除します。このためBSVは、非常に品質の高いモデルを作成することが可能であり、設計者は本来の仕様の妥当性の検証に専念することが可能です。

BSV vs. Verilog



Verilog

```
always @(posedge CLK)begin
  if (!cond2 && cond1)
    x <= x - 1;
  else if (cond0)
    x <= x + 1;
end

always @(posedge CLK)begin
  if (cond2)
    y <= y - 1;
  else if (cond1)
    y <= y + 1;
end
```

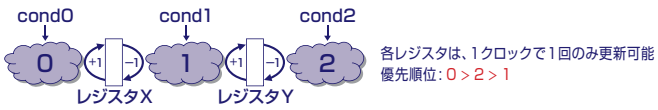
Bluespec SystemVerilog

```
(* descending_urgency = "proc2, proc1, proc0" *)

rule proc0 (cond0);
    x <= x + 1;
endrule

rule proc1 (cond1);
    x <= x - 1;
    y <= y + 1;
endrule

rule proc2 (cond2);
    y <= y - 1;
endrule
```

優先順位を記述

行いたい処理単位で記載

優先順位を変更
(=仕様の変更)



Verilog

```
always @(posedge CLK) begin
  if (cond0)
    x <= x + 1;
  else if (cond1 && !cond2)
    x <= x - 1;
end

always @(posedge CLK) begin
  if (cond2)
    y <= y - 1;
  else if (!cond0 && cond1)
    y <= y + 1;
end
```

Bluespec SystemVerilog

```
(* descending_urgency = "proc0, proc2, proc1" *)

rule proc0 (cond0);
    x <= x + 1;
endrule

rule proc1 (cond1);
    x <= x - 1;
    y <= y + 1;
endrule

rule proc2 (cond2);
    y <= y - 1;
endrule
```

優先順位を変更

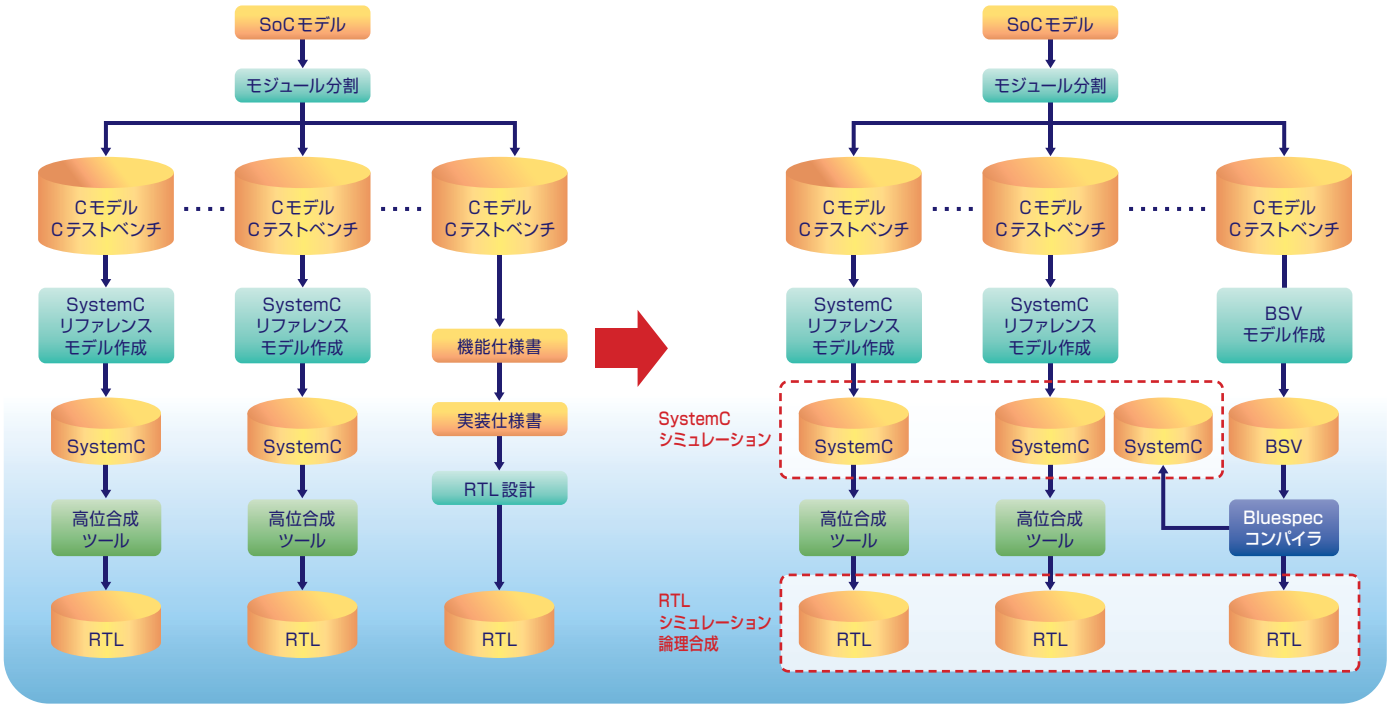
ruleの記述は変更なし

記述を大幅に変更

BSVは処理単位の記述手法(rule)を使用するため、仕様の追加/変更の際に、絶大な効果が現れます。

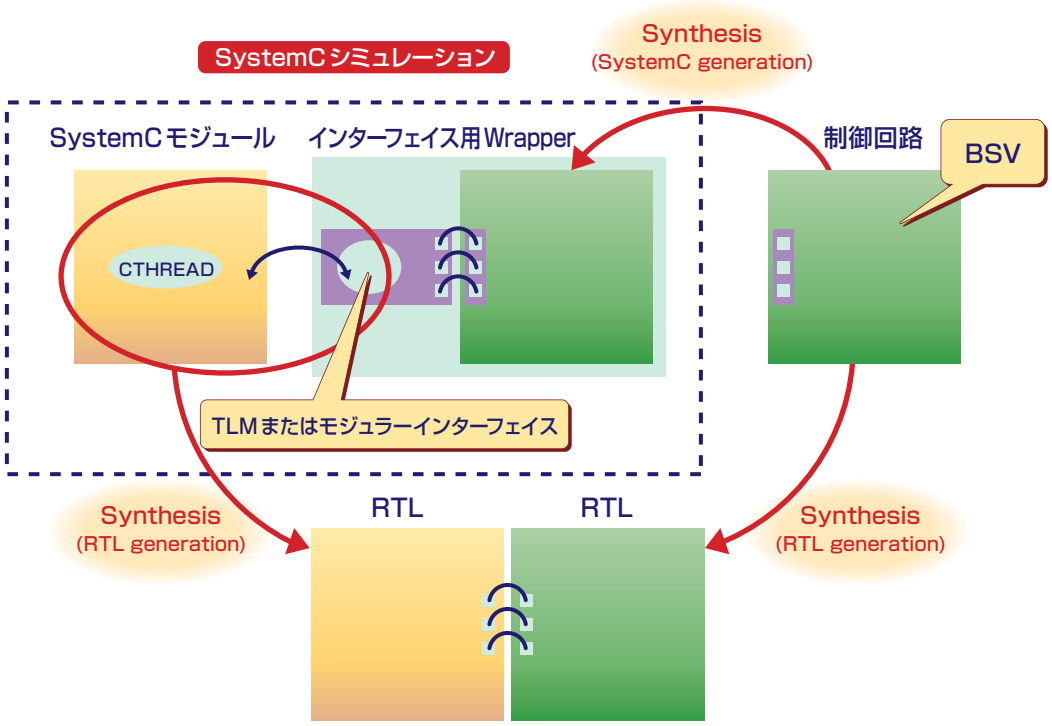
SystemCとの連携

SystemCによる高位合成が、実用レベルになってくるに従い、ESL設計が、注目を集めています。ESL設計では、Cでのモデル/アルゴリズムをモジュール分割し、SystemCでリファレンスモデルを作成するフローが一般化してきています。リファレンスモデルを作成する際、アルゴリズミックなモジュールや信号処理のようなデータパスでは、実績も多く高位合成も得意な分野です。しかし、制御回路などは、SystemCで記述しても、合成を考えるとクロックを意識する必要があるため、非常に煩雑になるケースが多々あり、verilog-HDLやVHDLでRTLを記述している場合も、まだまだ多いのが現実です。BSVは、ruleという構文を使用して、条件ごとの動作の記述をおこない、コンパイラは、記述より競合条件を検出し、排他制御を自動で合成するため、複雑な制御の実現に大きな効力を発揮します。また、BSVのコンパイラは、検証用のサイクルアキュレートなSystemCを合成可能です。したがって、煩雑になりがちなバスや制御系をBSVで記述しSystemCレベルでの検証が可能となります。



SystemCのモジュール間インターフェイスには、TLMでのインターフェイスとモジュラーインターフェイスを使うことが、一般的になっています。BSVから合成されるSystemCモジュールは、サイクルアキュレートでピンインターフェイスの検証用モデルです。しかし、SystemCシミュレーションのためにTLM、モジュラーインターフェイスをサポートします。

Bluespecツールは、SystemCモジュールとインターフェイスするため、TLMインターフェイス用Wrapperとモジュラーインターフェイス用Wrapperの2種類を生成します。このWrapperを使ってSystemCテストベンチで接続することにより、SystemCモジュールを修正することなくシミュレーションを行うことが可能です。



合成可能な仮想プラットフォームSVP
(Synthesizable Virtual Platform)

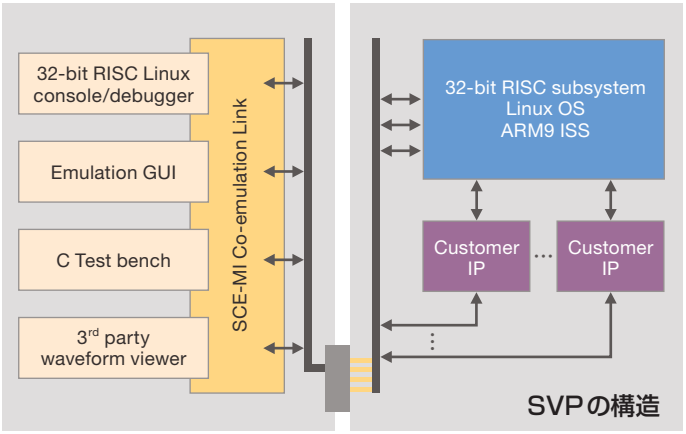
SVPの特徴

SVPには、32bit RISCプロセッサモデルとAXI/AHPシステムバスと必要なペリフェラルモデル(タイマ、割り込みコントローラ、メモリコントローラ、UART)が含まれています。全てのモデルはHW合成可能なBluespec SystemVerilog(BSV)で記述されており、このモデルはシミュレーションやエミュレーション上で動作させることができます。

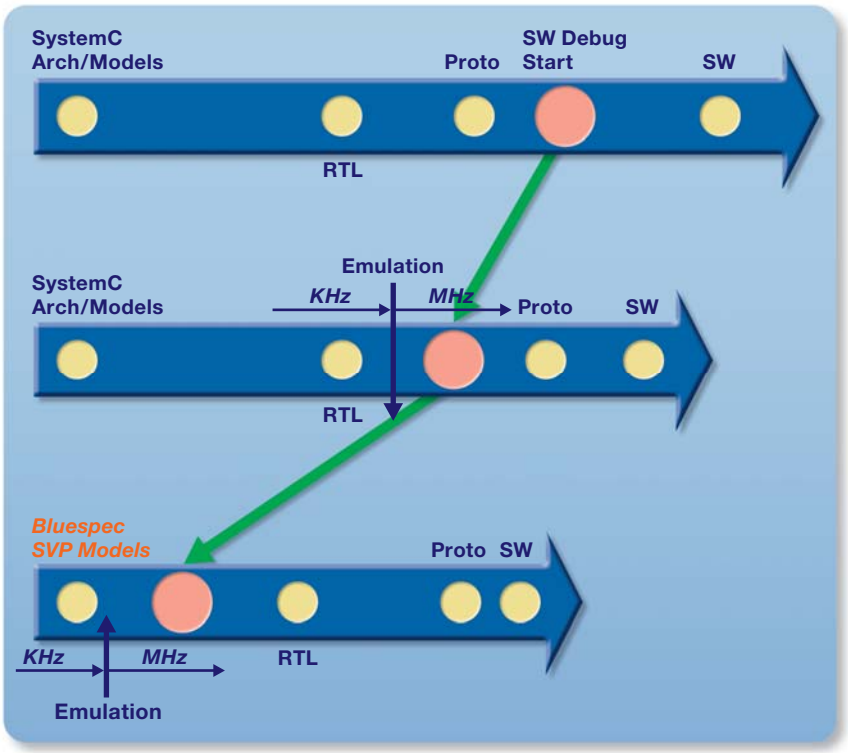
SVPとemVMとを組み合わせたSVPエミュレーションシステムでは、SCE-MIリンクを介してホストPCと汎用のFPGAボードを接続します。ホストPCでは、Linuxのコンソール、デバッガと、エミュレーションを制御するためのGUIが実行できます。

操作性

- ユーザーが設計したハードウェアIPの実装は、BSVを使用することによって容易に行うことができ、Linuxコンソールからそれを制御することができます。
- 観測したいバスやユーザーIPの信号は、GUI環境を使用して容易にプロービングすることができます。



Bluespec SVPモデルを使用した早期検証環境の実現



エミュレータ無しの場合、プロトタイプが完成するまでソフトウエアのデバッグが出来ません。

高価なエミュレータを使用することで、プロトタイプがなくても協調検証が可能となりますが、アーキテクチャ/RTLの完成を待たなくてはいけません。

Bluespecが提唱する「仮想プラットフォーム」では、BSV自体が合成可能な記述のためアーキテクチャ決定前に機能のみの実装でFPGAへのマッピングを行います。そのため初期段階からソフトウエアデバッグが可能となり、開発期間の短縮に貢献します。

合成可能なSoC仮想プラットフォームSVP(Synthesizable Virtual Platform)

Bluespec Synthesizable Virtual Platform(SVP)は、エミュレーション上で実行可能なSoCのバーチャルプラットフォームです。SVPは、ARM9の命令セットシミュレーション(ISS)とLinuxのコンソールをサポートします。(Coretex A9もサポート予定)

SVPをemVMと組み合わせてご利用頂くと、セットアップのためのわずかな工数と少ない投資で、システム全体のエミュレーションを実現することができます。

●SOC検証
SCO機能検証における
パフォーマンスの高速化

●ファームウェア開発
早期のファームウェア開発と
検証が可能

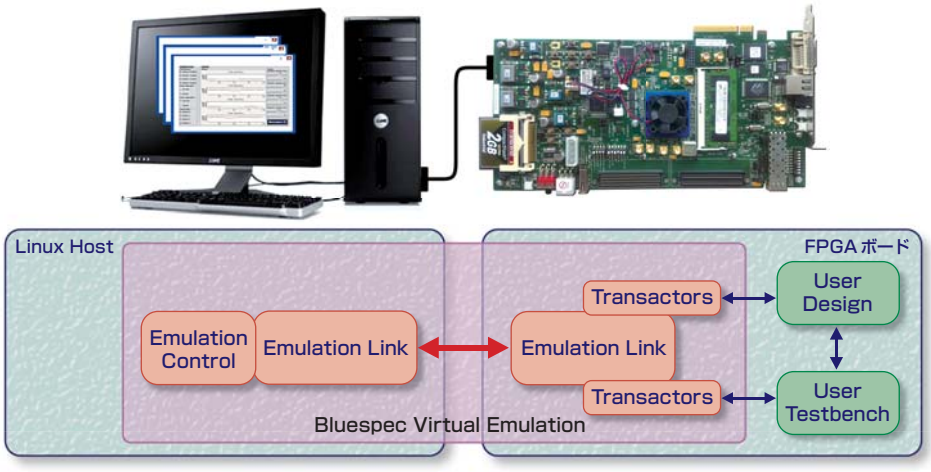
●ハードウェアIPの実装
容易なシステムバスへの接続
早期のシステム上での動作確認が可能

仮想エミュレーションシステムemVM

エミュレータを利用することは、検証を加速する手段としては、非常に有効です。しかしながら、エミュレータは非常に高価なシステムです。これに対して、汎用のFPGAボードを使用したプロトタイピングは、比較的安価で実現可能ですが、デバッグが非常に困難です。Bluespecでは、汎用のFPGAボードを利用して、安価にエミュレーターを実現する仮想エミュレーションシステムemVMをご提供いたします。emVMは、汎用FPGAボードを使用してエミュレータを実現するソリューションです。通常、FPGAボードに対象となる回路及びテストベンチをマッピングした場合、実行の制御、結果の確認、内部のモニタを行うことは非常に大変です。これに対して、emVMでは、FPGAにマッピングするためのRTLにPCと通信するためのEmulation LinkとTransactorを自動で挿入を行います。Emulation LinkとTransactorを挿入された回路は、PCI Expressを使用しPCとの通信が可能となります。このFPGAに対して、emVMのEmulation Control(GUI)を使用し、エミュレータとしてコントロールをすることが可能となります。

emVMの主なメリット

- 様々な汎用FPGAボードを使用可能
- 容易なハードウェアIPのシステムへの組み込み
- 容易なハードウェアIPの制御
- 容易なプローブ抽出
- 容易なデバッグ性
- 低価格



emVMは、ホストPCとFPGAボードとを接続するための標準インターフェース(Emulation Link)を提供します。また、GUIによって、容易にプローブの抽出を行ったり汎用Verilogシミュレータや波形表示ツールを起動することができます。FPGA側で用意されているGet/Put型のトランザクタを使用すると、ユーザーが設計したハードウェアIPとテストベンチを容易に接続することができます。更に複雑なプロトコルを変換するためのトランザクタ開発が必要となった場合は、emVMにプラグインされているSynthesizable Modeling Tool Kitを使用することで容易にそれを行うことができます。

emVMの特徴

- SCE-MI 1.0/2.0サポート
- PCIeサポート
- ホストPC側のC/SystemCのためのインターフェースライブラリ
- シミュレーション・エミュレーションの容易な切り替え
- SpringSoft社、GTKの波形表示ツールをサポート
- GUIからのプローブ信号の自動生成
- GUIからのプローブポイントのON/OFF切り替え

サポートする汎用FPGAボード例

Board	Supplier	Total gates
ML507※1	Xilinx	425K
XUPV5※1	Xilinx	660K
DN7002K10※2	Dini Group	13M
DN7006K10※2	Dini Group	39M

※1 Xilinx Virtex-5/6 FPGA搭載
※2 Altera Stratix-4 FPGA搭載

様々なサービスのご提供

当社は、お客様のニーズに合わせたemVMとSVPの環境をご提案いたします。

- RISCサブシステムの拡張
- トランザクタの開発
- 他の汎用FPGAボードへの移植
- 各種トレーニング
- その他

11.07.2000

CYBERNET

サイバネットシステム株式会社
EDA事業部　LSIソリューション室

本　　社　〒101-0022 東京都千代田区神田練塀町3番地 富士ソフトビル
Tel:(03)5297-3914　Fax:(03)5297-3646

http://www.cybernet.co.jp/bluespec
e-mail : bluespec@cybernet.co.jp

お問い合せ: