

UMLからソフト/ハード協調設計

CATS社

ソフトウェア事業部 第2技術課 アシスタントマネージャー

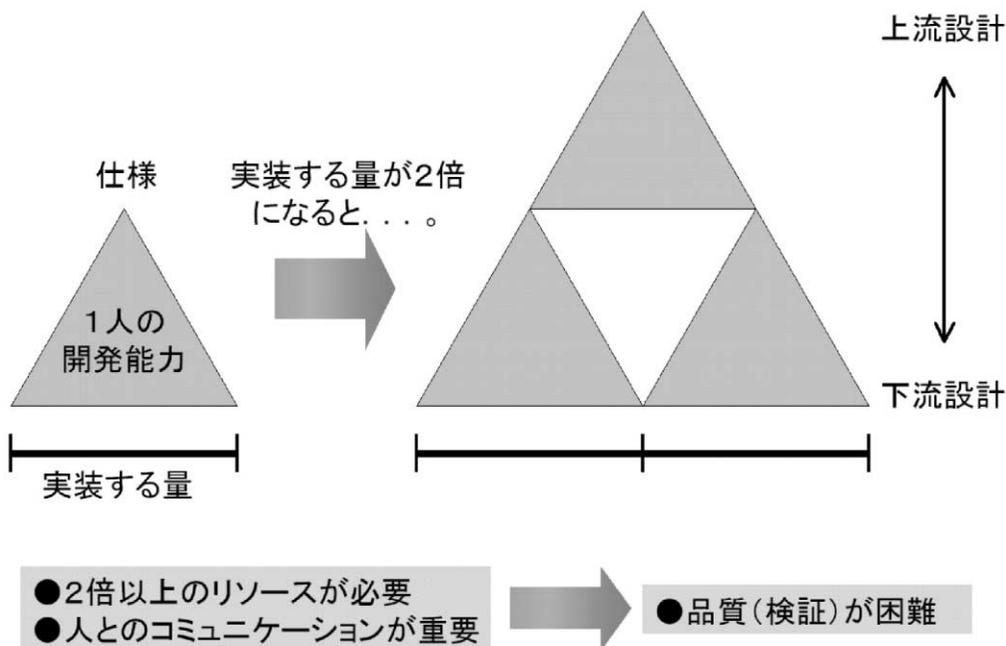
塚田 雄一

はじめに (開発の現状)

1965年、インテル共同設立者の Gordon Moore は「半導体チップに集積されるトランジスタの数は約2年ごとに倍増する」と予測しました。この事は「ムーアの法則」という名で広く知られています。開発の規模は膨大化/複雑化の一途をたどっています。また現在の対策としては「人海戦術」により人員の数を増やすといった対策がとられています。下図をご覧ください。単に人海戦術といっても、実装する量が2倍になると人員を単に2倍にすれば良いとい

ったものではありません。人員は3倍から4倍少なくとも2人以上の人員は必要になります。また人員が増えるという事はコミュニケーションミスが発生する可能性が増え、人と人のコミュニケーションを取ることが重要な課題となってきます。これでは製品に対する品質を保持する事が困難となってきます。この事は周知の事実であり誰もが何とかしなくてはイケナイと思っております。CATS(株)は、UMLからソフト/ハード協調設計により解決すると言った1つの仮説をたてました。

人海戦術とは？



UMLにより解決するといった仮説

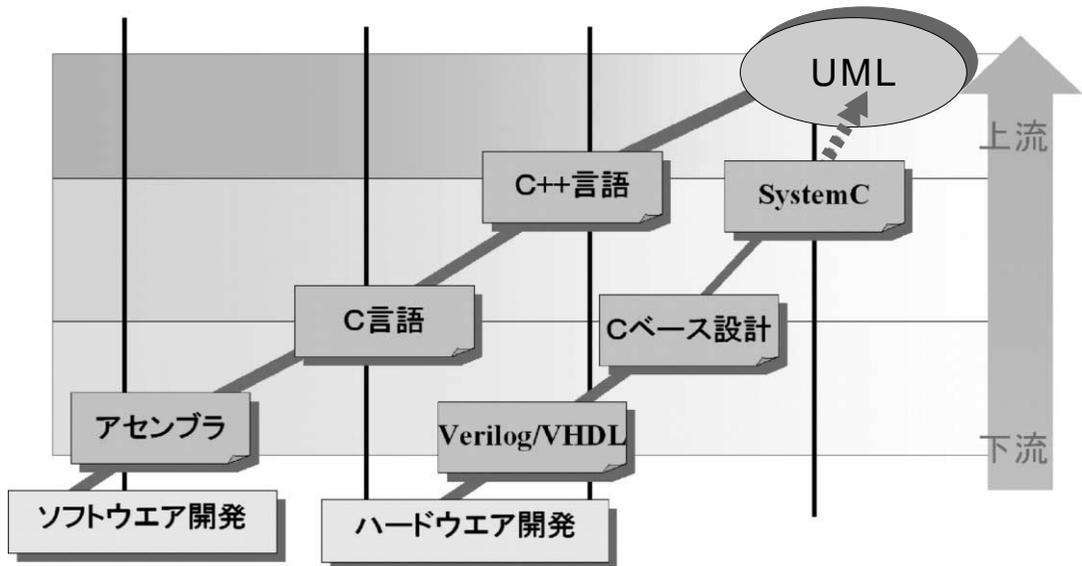
ソフトウェアの歴史をしてみると、昔はアセンブラで設計が行われていました。そして、開

発規模が膨大になると構造化言語であるC言語により開発が行われるようになり、そしてまた

さらに開発規模が膨大になるとオブジェクト指向言語であるC++言語により開発が行われるようになりました。そして、今まさにUMLによりモデルベース設計が行われています。ハードウェア（SoC）の世界にも今まさに同じような事が起こっています。ハードウェアも将来は、ソ

フトウェア開発と同様にUMLによりモデルベース設計が行われるようになるでしょう。そしてソフト/ハード協調設計を実現するためにはハードウェア（SoC）もUMLから開発を行うといった事が重要となってきます。

膨大化、複雑化をUML(モデルベース設計)により解決！



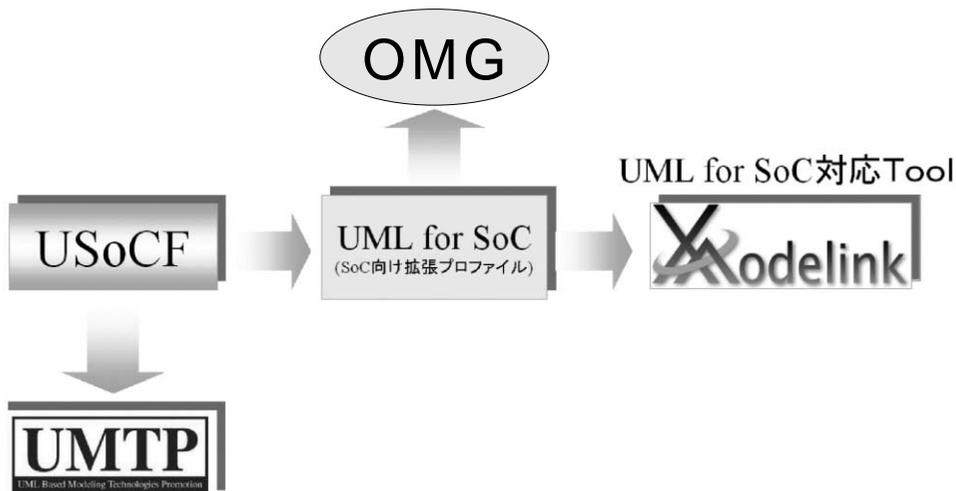
「UML for SoC」

OMG標準プロファイルとして策定

オブジェクト指向を行う上で、C++言語だけは開発は困難になります。と言うのは、C++のソースコードを他の人が参照し理解するのは非常に困難になるからです。そこでUMLなどの誰にでも判りやすく、誰にでも理解出来るドキュメントが必要になってきます。UMLは統一モデリング言語であり、誰にでも判りやすく理解する事が出来ます。しかしここで注意しないといけない事があります。それはUMLはあくまでも判りやすい表現に過ぎません。どの様な表現を行うか、どの様に開発に適用するかが重要になってきます。またUMLはソフトウェア開発用に考えられた言語であり、ハードウェア（SoC）

開発に適している訳ではありません。そこでハードウェア（SoC）開発に適した表現が必要になってきます。という事で2003年に富士通、ラショナル（現在のIBM）様とCATSの3社で「USoCF」（UML for SoC forum）を立ち上げました。その後数社の方々にご参加いただきました。そしてUMLをSoC開発に適用するための「SoC向けプロファイル」を策定しOMGに提案を行いました。USoCFはOMGに提案を行った事で活動は終了し、その後、富士通様が策定を繰り返し行い、結果2006年6月正式に標準化として認められました。UML Profile for System on a Chip（SoC）, v1.0のドキュメントは「<http://www.omg.org/cgi-bin/doc?formal/06-06-01>」よりファイルロード可能になります。

UML for SoC標準化

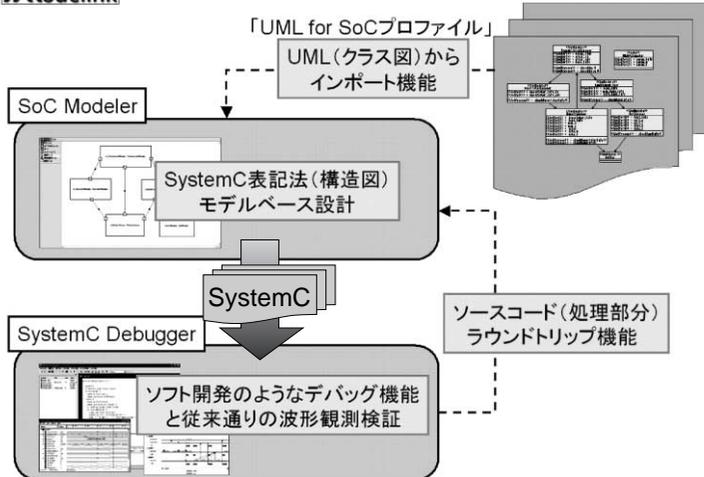


UML for SoCに対応したツール 「XModelink」

CATS製「XModelink SoC Modeler」は、「UML for SoC」に準拠しています。OMGに標準化申請の際、サポートツールが必要であり、標準化の際にはすでに「XModelink」が存在していました。UMLを使用しオブジェクト指向設計を行った後、「XModelink SoC Modeler」の

インポート機能によりSystemC表記法に取り込みが可能となります。その後「SystemCソースコード」を自動生成いたします。UMLを使用したSoCにおける上流設計を可能としたツールとなります。また、2004年ESECにおいて「UML SoC」への取り組みが評価され「LSIオブ・ザ・イヤー・グランプリ」を受賞致しました。

XModelink製品説明



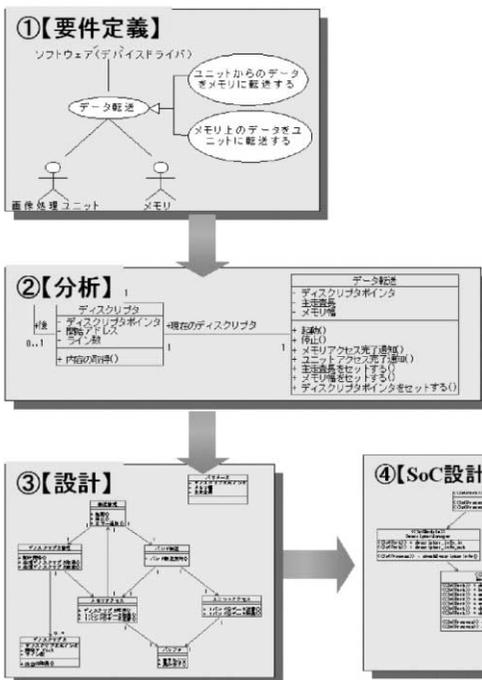
UML for SoC 種類	対応/未対応
module	対応
port	対応
clock	対応
reset	対応
interface	対応
channel	対応
protocol	未対応
process	対応
connector	対応

「UMLからSystemCへのモデリング」の事例

ここで「UMLからSystemCへのモデリング」の事例を1つご紹介いたします。USoCFのメンバーである株式会社リコー様の「DMAコントローラ」の事例です。この事例はディスクリプタ付きのDMAコントローラをモデリングしましたものです。まずはじめにユースケース図を使用し要求定義を行います。次にクラス図を使用

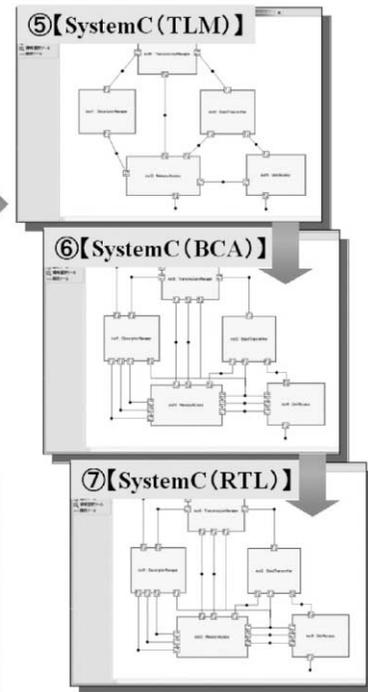
し分析を行い、設計レベルに落とし込みます。そして最後に「UML for SoC」プロフィールに変換を行います。以上がリコー様の事例となりますが、詳細においては本誌ZIPCウォッチャーVol.8のp58に記載されております。また「UML for SoC」表現以降は、SystemCにおけるTLMレベル、BCAレベル、RTLレベルと設計を行う事が可能となります。

リコー様UMLモデリング紹介



DeignWave 2004年12月号 UMLの紹介

キャッツSystemC設計紹介



DeignWave 2005年4月号 SystemCの紹介

現在の活動「UMTP」

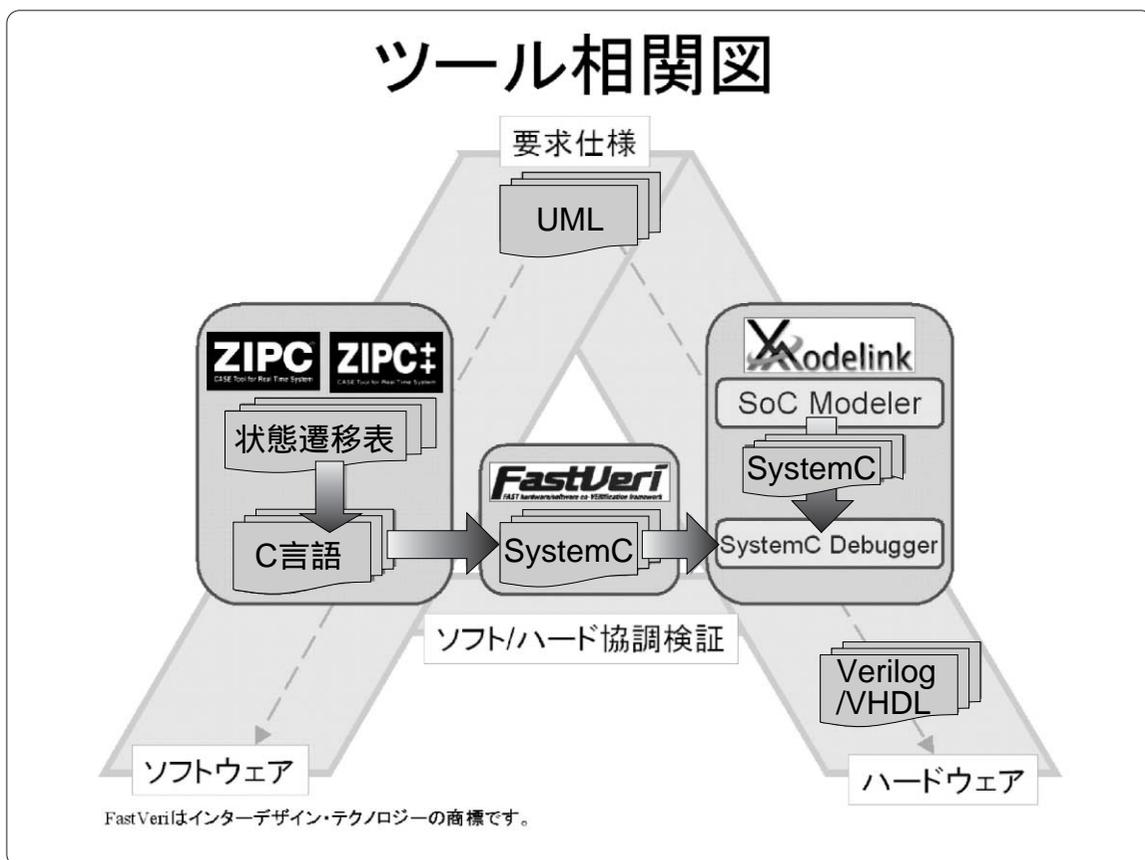
「UML (UML for SoC)」は実際に使用(モデリング)しなくては価値がありません。USoCFの活動後、現在は、「UMTP (UMLモデリング推進協議会)」のモデル共有部会の「SoC分科会」として現在活動を行っております。実際の活動内容と致しましては、「システム要求分析グループ」「ハード/ソフト分割グループ」「ハード設計グループ」の3つのグループに分かれてそれぞれモデリングを行っております。

ソフト/ハード協調検証を含めたツールチェーン

UMLは、元々ソフトウェア開発者のために開発されました。ハードウェア開発者もUMLにより開発が可能となるとUMLという共通のドキュメントを使用する事になります。組み込み開発は、ソフト/ハード片方だけでは実現出来ません。従ってソフト/ハードを含めてシステム設計を行う事が重要になってきます。それでは何故ソフト/ハード協調設計があまり行われてい

ないのでしょうか？ 1つの理由として、検証に非常に時間が掛かってしまう事があげられます。それが今、STARC(半導体理工学研究センター)様のバジェット追加技術により高速強調検証が可能となり解決致しました。従来のISS (Instruction Set Simulator)を使用した検証に比べると実に100倍から1000倍高速になります。これだけ高速になると多くのイテレーションが可能となります。バジェット追加技術とは、ソフトウェア開発されたC言語のソースコードにターゲットCPUの実行時間情報を埋め込みシミュレーションを行うといった技術です。そのバジェット(実行時間情報)をツールにより自動

的に埋め込むようにしたものがインターデザイン・テクノロジー様の「FastVeri」です。また「XModelink」を使用する事により、その検証を「Visual Studio」のようなホスト機の開発環境で実現可能となります。ソフトウェア開発者は、ハードウェアの設計完了を待たずに「Visual Studio」を使用して開発が行えます。また、周辺デバイス(ハードウェア)のシミュレーションも可能となり、ソフト/ハード平行開発が可能となります。この開発スタイルを図に示すと、アルファベットのAという文字になるため「A(エース)ソリューション」と名付けました。



最後に

UMLからソフト/ハードの協調設計を行うといった文化はまだありません。それは、新しい文化を創っていくという言わば挑戦です。ソフトウェアの世界においてUMLは定着しつつありますが、組み込み開発においてはまだまだ定着し

ているとは言えません。このままでは良いとは誰も思っているわけではなく、何とかしなくてはイケナイ事は確かです。新しい開発プロセス「UMLからソフト/ハード協調設計」に新しい一歩を踏み出しませんか。一緒に...

