

# IBISモデルの開発方法【Part 1】

## IBISモデルの開発が設計の成功に不可欠な理由

著者：Jermaine Lim、プロダクト・アプリケーション・エンジニア  
 Keith Francisco-Tapan、設計検証エンジニア

### IBISモデルとは何なのか？

IBIS (Input/Output Buffer Information Specification) モデルは、デバイスが備えるデジタル・ピンの特性や振る舞いを表すシミュレーション用のモデルです。設計段階で行われる高速シミュレーションで使用できるように、ICベンダーから顧客に対して提供されています。IBISモデルでは、IBIS Open Forumによって定められたパラメータを使用して、ICが備えるI/Oの動作を表現します。IBIS Open Forumは、IBISモデルの仕様と規格の管理／更新を担う業界団体です。

IBISモデルの実体は、ASCII形式のテキスト・ファイルです。電圧と電流、電圧と時間の関係を表す情報が表形式で記述されていますが、プロプライエタリな情報は含まれません。つまり、トランジスタのサイズ、バッファ回路の設計に用いられるデバイス・モデルのパラメータ、回路の構成など、ICの設計に関連する情報はIBISモデルには含まれないということです。また、ほとんどのEDAベンダーは、IBISモデルをサポートしています。言い換えれば、一般的に使用されているほとんどのプラットフォーム上では、IBISモデルを使用したシミュレーションを実施できるということです。

### なぜIBISモデルを使用するのか？

何らかのICを評価するためのテストを実施したところ、その結果は合格であったとします。それに続いて、そのICを使用する基板を設計し、直ちに製造を承認したとしましょう。ところが、基板を製造した後になって、その基板の性能が基準を満たしていないことが発覚しました。シグナル・インテグリティに関する何らかの問題が原因で、クロストーク、信号のオーバーシュート／アンダーシュート、インピーダンス・ミスマッチによる反射が生じることが判明したのです。このようなケースでは、その後どのような展開が続くことになるのでしょうか。当然のことながら、基板の設計と製造をやり直す必要があります。その結果、大きな時間のロスが生じ、コストが増大することになります。

上記のような事態が発生する原因はどこにあったのでしょうか。それは、プリシミュレーションという非常に重要な工程が実施されていなかったという点にあります。つまり、システム設計者は、本来であれば基板を構築する前に、シミュレーション・モデルを使用して、設計した回路のシグナル・インテグリティについて検証しなければならなかったのです。このプリシミュレーションの工程によってシグナル・インテグリティの問題を予測し、製造に移る前に対処していれば、時間やコストの浪費を最小化することができていたはずですが、このような作業を支援することを目的として、SPICEやIBISなどをベースとするシミュレーション・モデルが広く開発されています。プリシミュレーションの工程は、基板が試験で不合格になる確率を下げる効果をもたらします。

### IBISモデルの歴史

PCが普及し始めた1990年代、Intel<sup>®</sup>は最高クロック周波数が約33MHzで低消費電力のASIC向けに新たなI/Oバスを開発することにしました。それにあたっては、シグナル・インテグリティが損なわれないようにすることが重要でした。それについての検証を行うために開発されたのがIBISです。Donald Telian氏が率いていたチームは、I/Oバッファに関する情報をまとめたシートを作成しておき、Intelが設計した基板の試験にそれらの情報を活用するというアイデアを考案しました。以降、そのシートは、プロプライエタリな情報を一切開示することなく、顧客の基板設計を支援する手段として使われるようになりました。つまり、同社の顧客もそのシートを共有するようになったということです。当初は紙ベースのスプレッドシートを使っていましたが、顧客のシミュレータに確実に情報を伝えられるようにするために、IntelはEDAベンダーやその他のコンピュータ・メーカーと提携する決断を下しました。そのような過程を経て設立されたのがIBIS Open Forumです。同団体は、バッファの情報が記述され、機械で読み取ることができるテキスト・ベースのファイルのフォーマットを標準化する役割を担っています。



当初、IBISという名称は「Intel Buffer Information Sheet」の頭文字をとったものとされていました。その後、IBISの正式名称は「Input/Output Buffer Information Specification」に変更されました。IBISのバージョン1.0がリリースされたのは、1993年のことです。それ以来、IBIS Open ForumはIBISの普及推進に尽力続けています。具体的には、ツールや文書を提供し、規格を改良して特殊な領域に対応するための機能を追加しています。2019年には、IBISのバージョン7.0が承認されました。このように、IBISは、技術に対する新たなニーズに応じて継続的に改良されています。

## IBISモデルの開発プロセス

IBISモデルは、ICが備えるレシーバーとドライバのバッファ動作を、プロプライエタリなプロセス情報を公開することなく、汎用的にモデル化（モデリング）したものです。IBISの仕様即し、標準的なバッファの要素／動作を表形式のV-I（電圧 - 電流）データとV-t（電圧 - 時間）データを交えて表現することで、モデルを構築します。図1に、IBISモデルを開発する際の3つの主要な工程を示しました。以下、各工程について詳しく説明していきます。

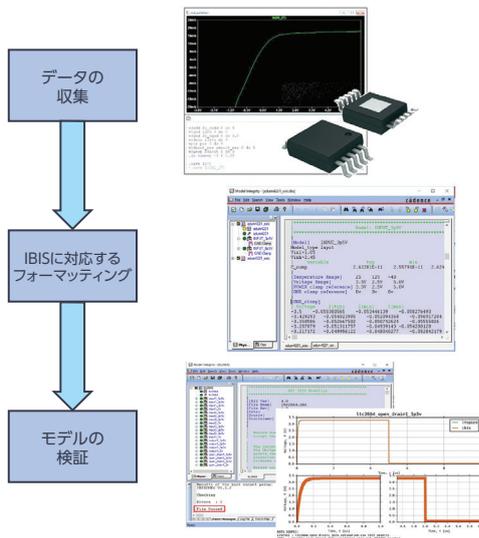


図1. IBISモデルの開発プロセス

## データの収集

IBISモデルの構築に向けた最初の工程では、データの収集を行います。その方法には、以下の2つがあります。

- ▶ シミュレーションによる方法
  - I/Oの回路図、データシート、パッケージの寄生要素（RLC）の値を使用します。
- ▶ 実測による方法
  - 実際のユニット／評価用ボード、データシート、パッケージの寄生要素の値を使用します。

図2は、IBISモデルの4つの主要な要素／コンポーネントについて説明したものです。図中の回路の出力ピンには、2つのダイオードが接続されています。これらは、その動作仕様／設計に応じ、入力が動作範囲やバッファの限界を超えている場合にバッファを保護する役割を果たします。バッファの限界は、モデルの中ではキーワード[Power Clamp Reference]（一般的には $V_{DD}$ ）または同[GND Clamp Reference]（一般的には $-V_{DD}$ ）を使って記述します。これらのダイオードは、ESDに対するクランプ動作を実現する保護素子として機能し、必要に応じてオンになります。一方、キーワード[Pullup]/[Pulldown]が付加されたコンポーネントは、バッファによるハイの状態／ローの状態の駆動動作に対応しています。したがって、[Pullup]/[Pulldown]のデータは、バッファが動作モードにあるときに取得します。

[Power Clamp]、[GND Clamp]、[Pullup]、[Pulldown]の各キーワードは、モデルにおける4つの主要な要素に対応しています。実際のモデルでは、これらのキーワードの下にV-Iデータを記述します。また、I/Oバッファのスイッチング動作は、モデルの中でV-tデータとして記述されます。

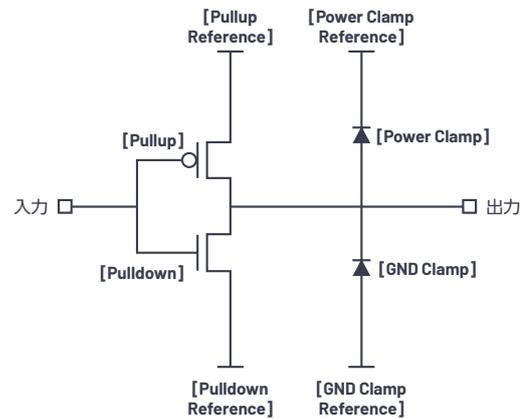


図2. IBISモデルの主要な要素とキーワード

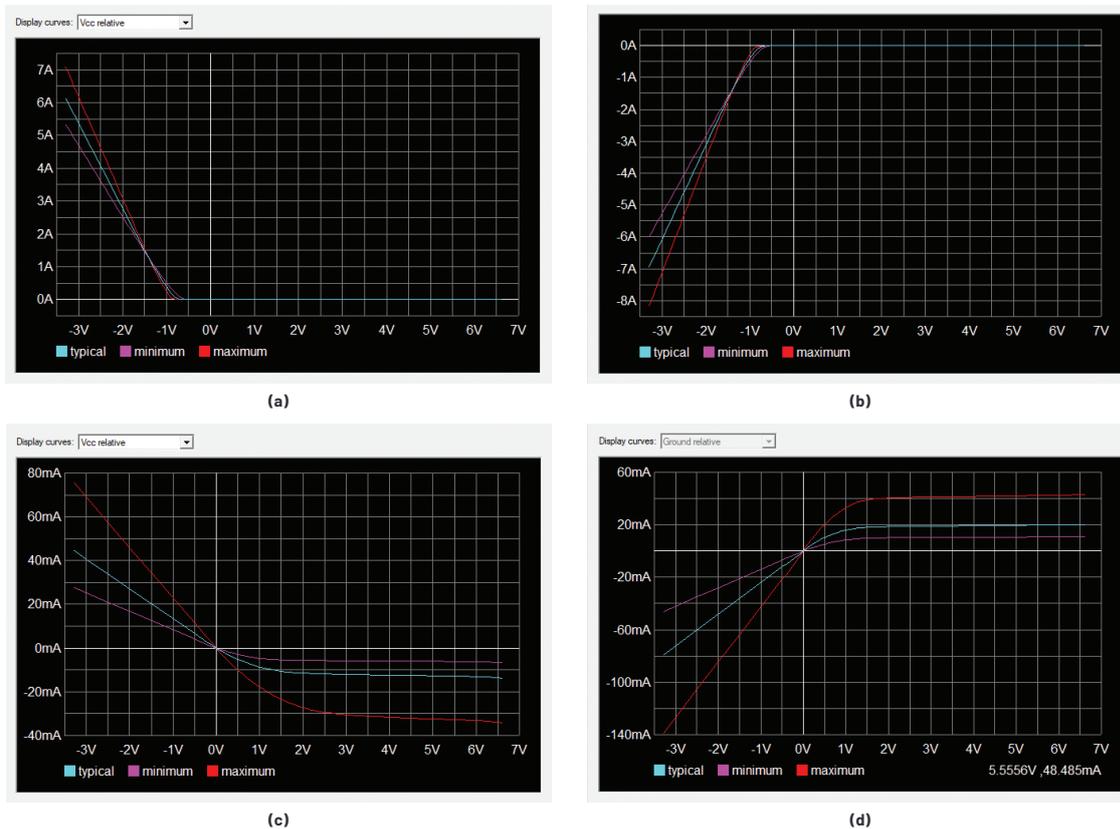


図3. V-I曲線の例。(a) は [Power Clamp]、(b) は [GND Clamp]、(c) は [Pullup]、(d) は [Pulldown] に対応しています。

### V-I動作に対応するキーワード

上記のとおり、V-I動作に対応するキーワードとしては、以下の4つが挙げられます。

- ▶ **[Power Clamp]** : デジタル I/O ピンに付加された電源クランプ用の ESD 保護ダイオードが高インピーダンスの状態にある場合を対象として、電源クランプ用の電圧リファレンスを基準とする V-I 動作を表します。
- ▶ **[GND Clamp]** : デジタル I/O ピンに付加されたグラウンド・クランプ用の ESD 保護ダイオードが高インピーダンスの状態にある場合を対象として、グラウンド・クランプ用の電圧リファレンスを基準とする V-I 動作を表します。
- ▶ **[Pullup]** : I/O バッファのプルアップ用コンポーネントがハイを駆動する場合を対象として、プルアップ用の電圧リファレンスを基準とする V-I 動作を表します。
- ▶ **[Pulldown]** : I/O バッファのプルダウン用コンポーネントがローを駆動する場合を対象として、プルダウン用の電圧リファレンスを基準とする V-I 動作を表します。

これらのキーワードに対するデータとしては、 $-V_{DD} \sim 2 \times V_{DD}$  の推奨電圧範囲における標準値、最小値、最大値の3つを記述します。標準値は、公称電圧、公称プロセス条件、公称温度におけるバッファの動作を表します。最小値は、最小電圧、最も能力の低いプロセス条件、最高動作ジャンクション温度 (CMOSの場合)

／最低動作ジャンクション温度 (バイポーラの場合) におけるバッファの動作を表します。最大値は、最大電圧、最も能力の高いプロセス条件、最低動作ジャンクション温度 (CMOSの場合) ／最高動作ジャンクション温度 (バイポーラの場合) におけるバッファの動作を表します。

ピンにおいて電圧を掃引し、それに対応する電流を測定することにより、IBISに基づくバッファのモデリングに必要なV-Iデータを取得します。図3に示したのは、上記4つのキーワードに対応するV-I曲線の例です。いずれも、標準値、最小値、最大値に対応する3つの条件下で取得した結果をプロットしたものです。

### スイッチング動作

IBISモデルでは、V-Iデータに加えてV-tデータも使用します。具体的には、I/Oバッファのスイッチング動作を、立上がり波形 (ローからハイへの出力遷移) と立下がり波形 (ハイからローへの出力遷移) のV-tデータ表という形で記述します。このデータは、出力に負荷を接続した状態で取得します。一般的には、標準的な伝送線の特性インピーダンスを表すために、負荷としては50Ωが使用されることが多いでしょう。しかし、IBISモデルについては、出力バッファが実際に駆動する負荷を使用するのが最良だと言えます。システムで75Ωの伝送線またはパターンを使用する場合には、75Ωの負荷を接続してV-tデータを取得することが推奨されます。

標準的なプッシュプル型CMOSのIBISモデルには、以下に示す4種類のV-tデータを含めることが推奨されます。

- ▶  $V_{DD}$ を基準とする負荷を使用した場合の立上がり波形
- ▶ グラウンドを基準とする負荷を使用した場合の立上がり波形
- ▶  $V_{DD}$ を基準とする負荷を使用した場合の立下がり波形
- ▶ グラウンドを基準とする負荷を使用した場合の立下がり波形

2つの立上がり波形のデータは、キーワード[Rising Waveform]の下に記述します。それらは、負荷が $V_{DD}$ またはグラウンドに接続されている状態でI/Oバッファの出力がローからハイへ遷移する際の特性を表します。一方、2つの立下がり波形のデータは、キーワード[Falling Waveform]の下に記述します。それらは、負荷が $V_{DD}$ またはグラウンドに接続されている状態でI/Oバッファの出力がハイからローへ遷移する際の特性を表します。なお、出力に負荷が接続されていることから、出力はフル振幅でスイングすることはないという点に注意してください。V-Iデータと同様に、V-tデータについても3つの値を取得します。図4に示すような遷移データを取得するという事です。

V-tデータからは、上昇率/下降率の値を抽出できます。ここで言う上昇率/下降率とは、電圧がある状態から別の状態に遷移するまでの変化率のことを指します。それらの値は、立上がりエッジ/立下がりエッジの20%~80%の範囲を対象として計算します。IBISモデルにおいて、上昇率/下降率の値(dV/dt)はキーワード[Ramp]の下に記述します。通常、同キーワードはV-tデー

タの表の後に記述します。上昇率/下降率の値は、出力バッファの本質的な立上がり/立下がり特性だけを表すためのものなので、パッケージの寄生要素の影響を排除して取得します。

上に挙げた項目だけでなく、IBISモデルには、シミュレーションのベースとなる様々な情報を盛り込みます。様々な情報とは、データシートに記載されている特性値などのことを指します。具体的には、動作電圧や動作温度範囲、入力論理に対応する電圧閾値、タイミング・テスト用の負荷の値、バッファの容量、ピンの構成といった情報を記述します。先述したように、パッケージの寄生要素の値も含めます。この情報はデータシートには記載されていませんが、高速システムのパターンについてシミュレーションする場合には、非常に重要な意味を持ちます。実際の基板上でもシミュレーションでも、それらの寄生要素は負荷として働くからです。つまり、それらは伝送線でやり取りされる信号の質に影響を及ぼすということです。

### IBISに対応するフォーマット

続いて、IBISモデルの開発プロセスにおける2つ目の工程について説明します。それは、IBISに対応するフォーマットの工程です。この工程では、IBISの仕様に即した形で実際にモデルを構築します。必要なすべてのデータを収集したら、モデルを作成する作業に移ることができます。IBISモデルは、メイン・ヘッダ・ファイル、コンポーネント・ディスクリプション、バッファ・モデルという3つの主要な要素から構成されます。

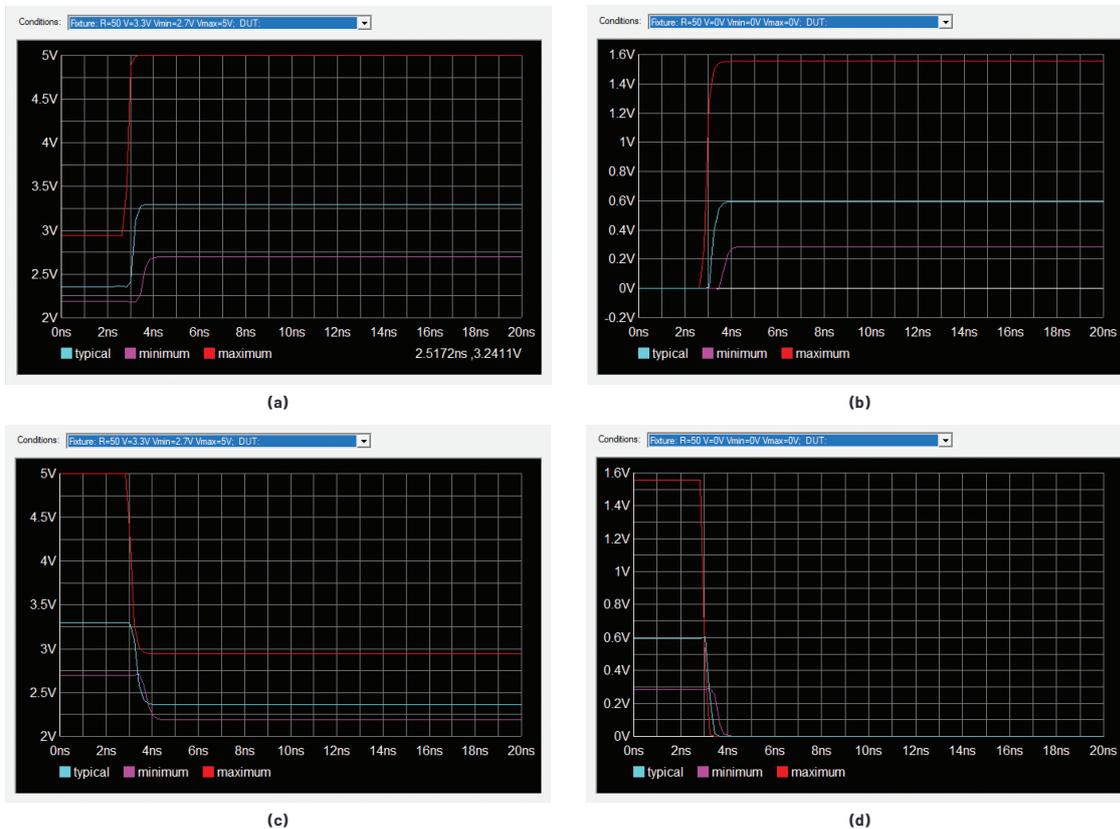


図4. I/Oバッファのスイッチング動作波形。このような波形から、立上がり/立下がりの遷移データを取得します。(a)は $V_{DD}$ を基準とする負荷を使用した場合の立上がり波形、(b)はグラウンドを基準とする負荷を使用した場合の立上がり波形、(c)は $V_{DD}$ を基準とする負荷を使用した場合の立下がり波形、(d)はグラウンドを基準とする負荷を使用した場合の立下がり波形です。

メイン・ヘッダ・ファイルには、モデルに関する一般的な情報を記述します。その情報とは、以下に列挙するようなものです。

- ▶ **IBISのバージョン**：キーワード [IBIS Ver] を使用し、モデルのベースとなる IBIS のバージョンを宣言します。それに基づき、シミュレータのパーサ (Parser) は、どのような種類のデータがファイルに含まれているのかを把握します。したがって、この情報はモデルがパーサによる検証に合格するか否かを定める重要な要素となります。
- ▶ **ファイル名**：キーワード [File Name] により、モデルについて記述したファイルの名前を小文字で宣言します。ファイル名には、適切な拡張子 (.ibs) を付加します。
- ▶ **改訂番号**：キーワード [File Rev] により指定します。ファイルの改訂レベルの管理に役立ちます。
- ▶ **日付**：キーワード [Date] により、モデルの作成日を明示することができます。
- ▶ **特記事項**：キーワード [Notes] を使って、モデルに関する注意事項を顧客に伝えるために記述します。例えば、モデルのデータはシミュレーションによって取得したものなのか、それとも実測によって取得したものなのかといった情報を記します。
- ▶ **ソース**：キーワード [Source] によって、モデルの基になったデータやモデルの提供元に関する情報を記述します。
- ▶ **免責事項**：キーワード [Disclaimer] を使って記述します。
- ▶ **著作権**：キーワード [Copyright] を使って記述します。

最初の3つの項目は、メイン・ヘッダ・ファイルに必須の要素です。それ以外の項目は必須ではありません。ただ、より詳細な情報を追加することにより、ファイルについて把握するのが容易になるので、記述しておくといでしょう (図5)。

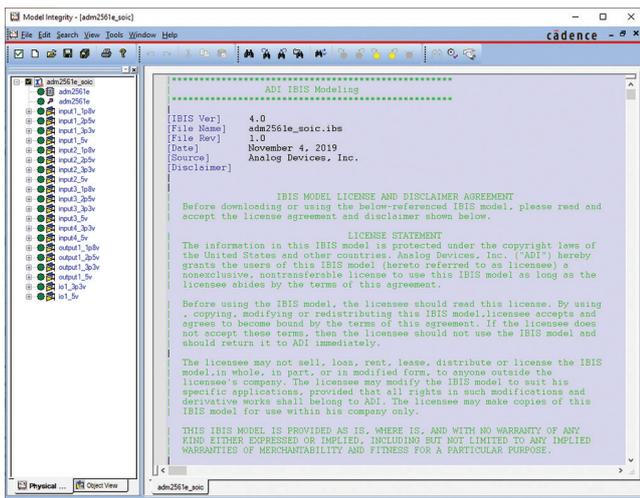


図5. IBISモデルのメイン・ヘッダ・ファイルの例 (Cadence Model Integrity を使用)

IBISモデルの2つ目の主要な要素はコンポーネント・ディスクリプションです (図6)。この部分には、コンポーネントに関する以下のような情報を記述します。

- ▶ **コンポーネント名**：キーワード [Component] を使って、モデル化するデバイスの名前を宣言します。
- ▶ **ピンのリスト**：キーワード [Pin] を使用し、ピン番号、ピン名、モデル名という少なくとも3つの列を使って情報を列挙します。このリストはデータシートに基づいて記述しますが、ピン番号とピン名が正しく対応していることの確認は必須です。また、IBISモデルでは、ピンごとに専用のモデル名が付与されることにも注意しなければなりません。ピンのモデル名はモデルの作成者が自由に付与できます。つまり、データシートに記載されているピン名とは必ずしも一致しません。また、複数のピンに対して同じモデル名が付与されるケースもあります。例えば、同じ回路図に基づく複数のバッファが存在する場合にそのような状況が発生します。それらは同じ動作を示すはずなので、1つのデータ・セットで表すことができます。
- ▶ **メーカー名**：キーワード [Manufacturer] を使用し、モデル化するコンポーネントのメーカーの名前を宣言します。
- ▶ **パッケージの寄生要素**：キーワード [Package] を使用し、抵抗、インダクタ、容量の集中定数で定義されるパッケージの電気的特性を宣言します。ピンの寄生要素の値もわかる場合には、キーワード [Pin] の下のピン・リストにそれらの値を記述します。その結果、より正確なモデルを作成することができます。その位置に記述されている値は、キーワード [Package] の下に記述された寄生要素の値よりも優先されます。

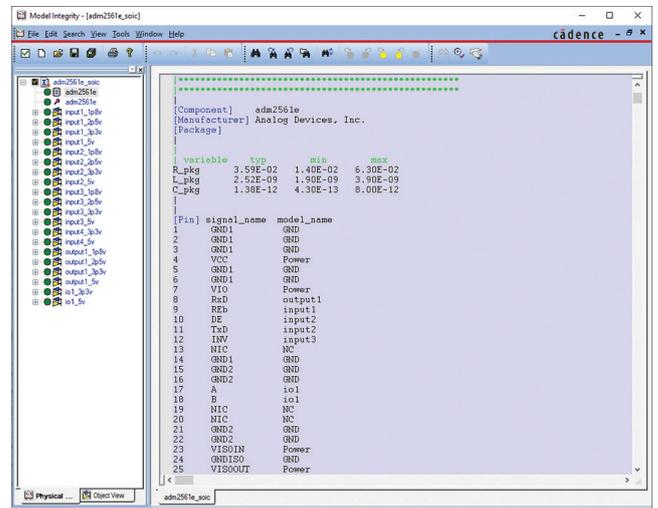


図6. IBISモデルのコンポーネント・ディスクリプションの例 (Cadence Model Integrity を使用)

IBISモデルの3つ目の主要な要素はバッファ・モデルです。この部分には、I/Oバッファの動作、特にV-IデータとV-tデータの値を記述します。最初に、キーワード[Model]を使用してモデルの名前を宣言します。その名前は、上述したキーワード[Pin]の下の3列目に記載されているものと一致していなければなりません。各バッファ・モデルに対しては、Model\_typeというパラメータを指定する必要があります。また、バッファの容量をC\_compというパラメータに続いて記述します。ここで言うバッファの容量とは、パッド側からバッファを見た場合の容量のことです。

IBISでは、複数種のバッファをモデル化することが可能であり、それぞれには固有のルールが適用されます。以下、4種の最も一般的なバッファと、IBISモデルにおけるそれぞれの要件について説明します。

- ▶ **入力バッファ**：モデル・タイプはInputです。このモデル・タイプには、入力論理に対応する閾値として、Vinl、Vinhという2つのパラメータを定義する必要があります。未定義の場合には、シミュレータによって、それぞれ0.8V、2Vというデフォルトの値が適用されます。これらのパラメータは、シミュレータによるタイミングの計算と、信号・インテグリティに関するルール違反の検出に役立ちます。

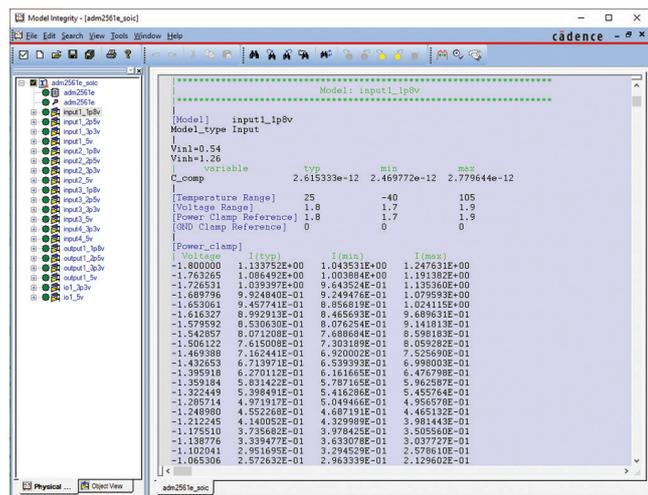


図7. 入力バッファの記述例 (Cadence Model Integrityを使用)。

- ▶ **2ステートの出力バッファ**：モデル・タイプはOutputです。これは、ハイを駆動する場合もローを駆動する場合も常にインエーブルの出力バッファに相当します。この部分には、Vref、Rref、Cref、Vmeasの各パラメータによって、タイミング・テスト用の負荷の値を指定します。これらのパラメータは必須ではありませんが、記述しておけば、シミュレータによる基板レベルのタイミングの計算に役立ちます。なお、この種のバッファはディスエーブルにできません。そのため、キーワード[Power Clamp Reference]と[GND Clamp Reference]による記述は使用しません。キーワード[Power Clamp]と[GND Clamp]によるV-Iデータも記述しません。

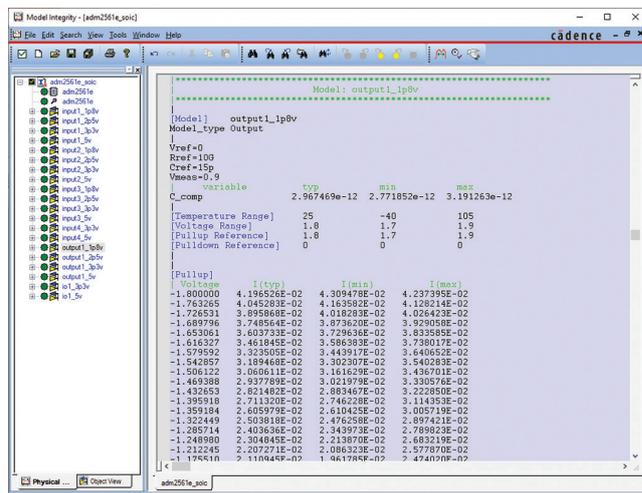


図8. 2ステートの出力バッファの記述例 (Cadence Model Integrityを使用)

- ▶ **3ステートの出力バッファ**：モデル・タイプはThree-stateです。このモデル・タイプは、ハイを駆動する状態とローを駆動する状態に加えて、高インピーダンスの状態を持ちます。つまり、ディスエーブルが可能な出力バッファに相当します。上記2ステートの出力バッファ (モデル・タイプはOutput) と同様に、Vref、Rref、Cref、Vmeasの各パラメータによって、タイミング・テスト用の負荷の値を指定します。これらのパラメータを記述しておくと、シミュレータによる基板レベルのタイミングの計算に役立ちます。

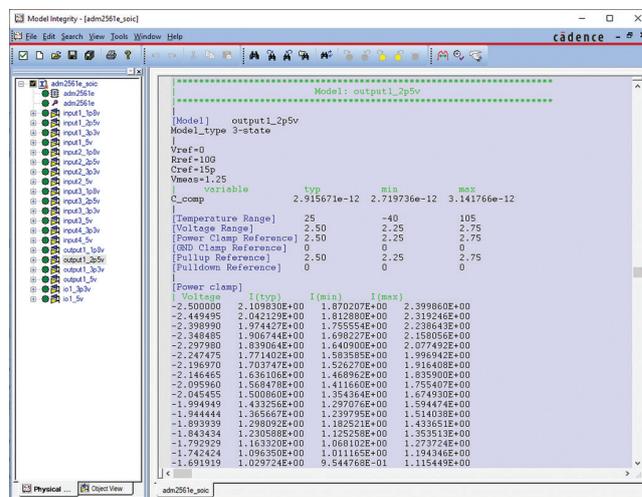


図9. 3ステートの出力バッファの記述例 (Cadence Model Integrityを使用)

- ▶ I/Oバッファ：モデル・タイプはI/Oです。I/Oバッファは、入力バッファと出力バッファを組み合わせたものです。したがって、このモデルには Vinl、Vinh、Vref、Rref、Cref、Vmeas の各パラメータが含まれます。

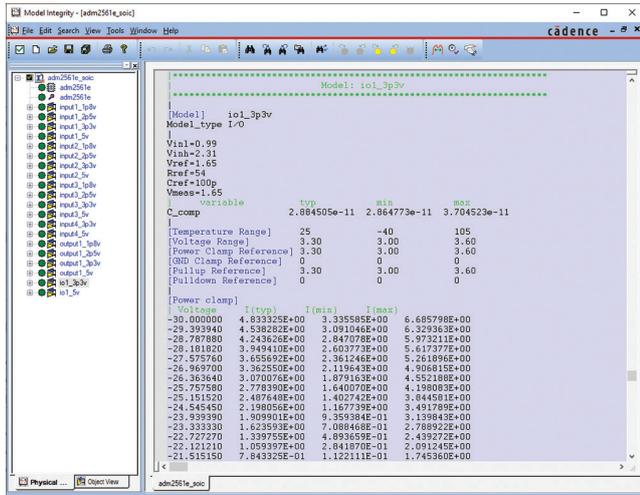


図 10. I/Oバッファの記述例 (Cadence Model Integrityを使用)

## IBIS Modeling Cookbook の活用

モデルの作成者は、上記のようなガイドラインに従って IBIS モデルを正しく構築する必要があります。さもないと、作成したモデルは、次に説明する検証に合格できません。モデルの作成方法の詳細については、IBIS Open Forum のウェブサイトに掲載されている IBIS Modeling Cookbook を参照してください。

## モデルの検証

IBIS モデルの検証は、パーサによるテストと関連の確認という 2 つのプロセスで構成されます。

## パーサによるテスト

モデルを構築する際には、Golden Parser を搭載しているソフトウェアを使用すべきです。Golden Parser とは、作成したモデルを IBIS の当該バージョンの仕様に照らし合わせて、構文のチェックを実行し、データの整合性を確認するためのプログラムのことです。この機能を搭載するソフトウェアとしては、Cadence Model Integrity や Hyperlynx Visual IBIS Editor などが挙げられます。

パーサによるテストに合格すれば、そのモデルは、標準的なフォーマットと仕様に従っており、V-I データと V-t データがマッ

チしているということになります。不合格になった場合には、エラーが生じる原因を特定しなければなりません。考えられる最も単純な原因は、IBIS の仕様に従っていないフォーマットやキーワードがモデルの中で使われているというものです。この種の誤りは簡単に修正できます。もう一つ、V-I データと V-t データのミスマッチもエラーの原因になります。このエラーが発生した場合、プルアップ/プルダウンに関する V-I データもしくは V-t データに誤りが存在する可能性があります。V-I データで表される動作が、V-t データで表される動作とマッチしていないということです。この問題を解決するには、再度シミュレーションや実測を行わなければならないかもしれません。しかし、その前に、まずはモデルの中に記述した電圧と負荷の値が正しいかどうかを確認してください。その結果、誤った電圧の値を定義していただけだったという単純なミスが見つかるかもしれません。その場合、再度シミュレーションや実測を行う手間と時間を省くことができます。

図 11 は、パーサによるテストに合格しなかった IBIS モデルの例です。これを見ると、エラーになった個所にフラグが立っていることがわかります。このような情報が得られるので、簡単にモデルを修正して次の検証工程に進むことができます。この例では、バッファに対するモデル・タイプの設定に誤りがあることからエラーが発生しています。具体的には、モデル・タイプ (キーワード Model\_type) として 「i/o」 という小文字の記述が行われています。IBIS では、このモデル・タイプは 「I/O」 と大文字で記述する必要があります。

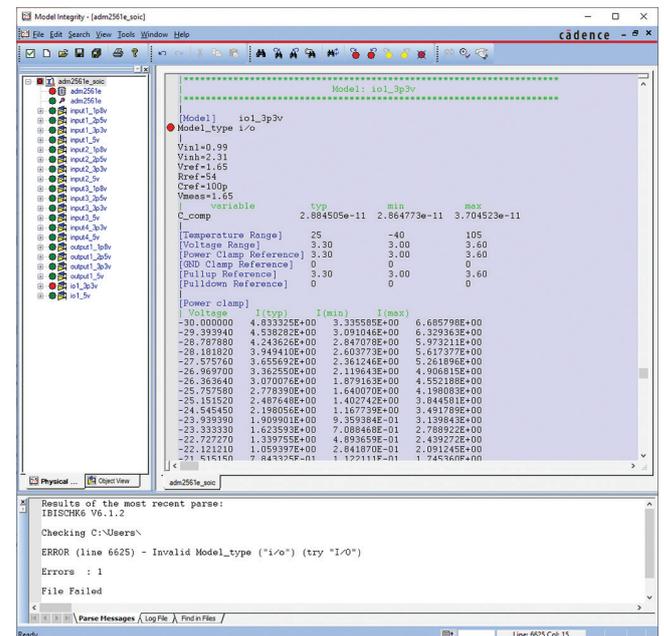


図 11. パーサによるテストに不合格となった IBIS モデル (Cadence Model Integrityを使用)

図12に示したのは、パーサによるテストに合格したモデルです。ご覧のように、「I/O」という大文字を使ってモデル・タイプを宣言するよう修正されています。その結果、エラーが解消されました。

パーサによるテストに合格した場合のみ、次に説明する関連の確認プロセスに進むことができます。

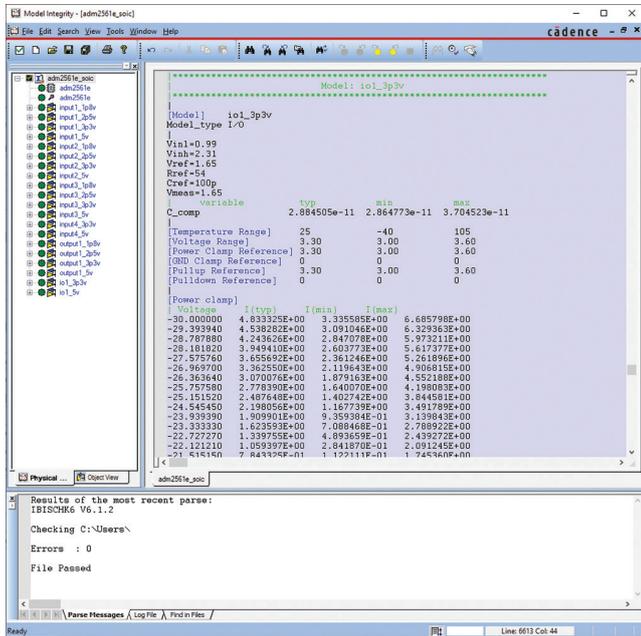


図 12. パーサによるテストに合格した IBIS モデル (Cadence Model Integrity を使用)

## 関連の確認プロセス

パーサによるテストに合格すれば、構文エラーなどは存在していないということがわかります。ただ、構築したモデルが現実のデバイスの動作を正確に模擬できているか否かは、どのようにして確認すればよいのでしょうか。この疑問は、関連の確認プロセスによって解消します。

IBISモデルには、表1に示すような品質レベルが存在します。ここでは、品質レベルの2aを満たすための関連の確認プロセスについて簡単に説明します。パーサによるテストに合格したら、そのモデルに外部負荷を追加し、パッケージの寄生要素の影響を含めてシミュレーションを実施します。通常、負荷としては、I/Oバッファの特性評価に用いられるタイミング・テスト用の値を使用します。それらの値は、データシートに記載されています。同様に、I/Oバッファの回路図を対象としたシミュレーションを同じ設定と負荷で実施します。両方のシミュレーション結果を重ね合わせることで、構築したモデルの動作が回路図に基づく動作とどの程度マッチしているのか確認できます。なお、オープンソースのソフトウェアを使用してIBISモデルを開発する例を別の記事で紹介する予定です。

表 1. IBISモデルの品質レベル

品質レベル	説明
レベル0	Golden Parser (ibischk) に合格
レベル1	チェックリスト・ドキュメントの定義に完全に従っている
レベル2a	シミュレーションによって関連を確認済み
レベル2b	実測によって関連を確認済み
レベル3	上記のすべてを満たす

## IBISモデルがシミュレーションに不可欠である理由

IBISモデルは、多くのEDAベンダーによって広くサポートされています。使いやすくファイルのサイズが小さいので、迅速にシミュレーションを実施することができます。また、プロセスや回路に関するプロプライエタリな情報は含まれていないので、半導体ベンダーはIBISモデルをそのまま顧客に提供することができます。デバイスのI/Oの動作を正確に表現できるだけでなく、上記のようなメリットも提供されるということです。

IBISモデルを利用することによって、基板のプロトタイプングや製造の工程に移る前に、シグナル・インテグリティの問題を予測／解決することができます。これは、基板の開発時間、ひいては製品を市場に投入するまでの時間を短縮することにつながります。

お客様がIBISモデルを利用する理由は、端的に言えば次のようになります。すなわち、IBISモデルを使ってシミュレーションを実施すれば、基板を設計し、デバッグを行って、それによる収益を上げるまでにかかる時間とコストを削減できるからです。

アナログ・デバイセズの製品に対応するIBISモデルは、[こちら](#)で検索することが可能です。

## 参考資料

Mercedes Casamayor [AN-715 – A First Approach to IBIS Models: What They Are and How They Are Generated (初めてのIBISモデル：モデルの概要と開発方法)] Analog Devices, 2004年

IBIS Modeling Cookbook for IBIS Version 4.0 (IBISモデリング・クックブック IBIS Version 4.0対応版)、IBIS Open Forum, 2005年9月

IBIS Version 7.0、IBIS Open Forum, 2020年4月

Roy Leventhal, Lynne Green [Semiconductor Modeling: For Simulating Signal, Power, and Electromagnetic Integrity (半導体のモデリング：信号、電力、電磁界インテグリティのシミュレーション)] Springer, 2006年



#### 著者について

Jermaine Lim ([jermaine.lim@analog.com](mailto:jermaine.lim@analog.com)) は、アナログ・デバイセズのプロダクト・アプリケーション・エンジニアです。2014年10月に入社して以来、アナログ・デバイセズの様々な製品に対応するIBISモデルの開発などに取り組んでいます。マニラ市立大学で電子工学の学士号を取得しています。



#### 著者について

Keith Francisco-Tapan ([keith.francisco@analog.com](mailto:keith.francisco@analog.com)) は、アナログ・デバイセズのAMS設計検証エンジニアです。2012年3月に、アナログ設計エンジニアとして入社しました。当初はアナログ・デバイセズの様々な製品に対応するIBISモデルの開発を担当。ADGTでモデル開発の技術を高めていきました。ミンダナオ州立イリガン工科大学で電子工学の学士号を取得しています。