

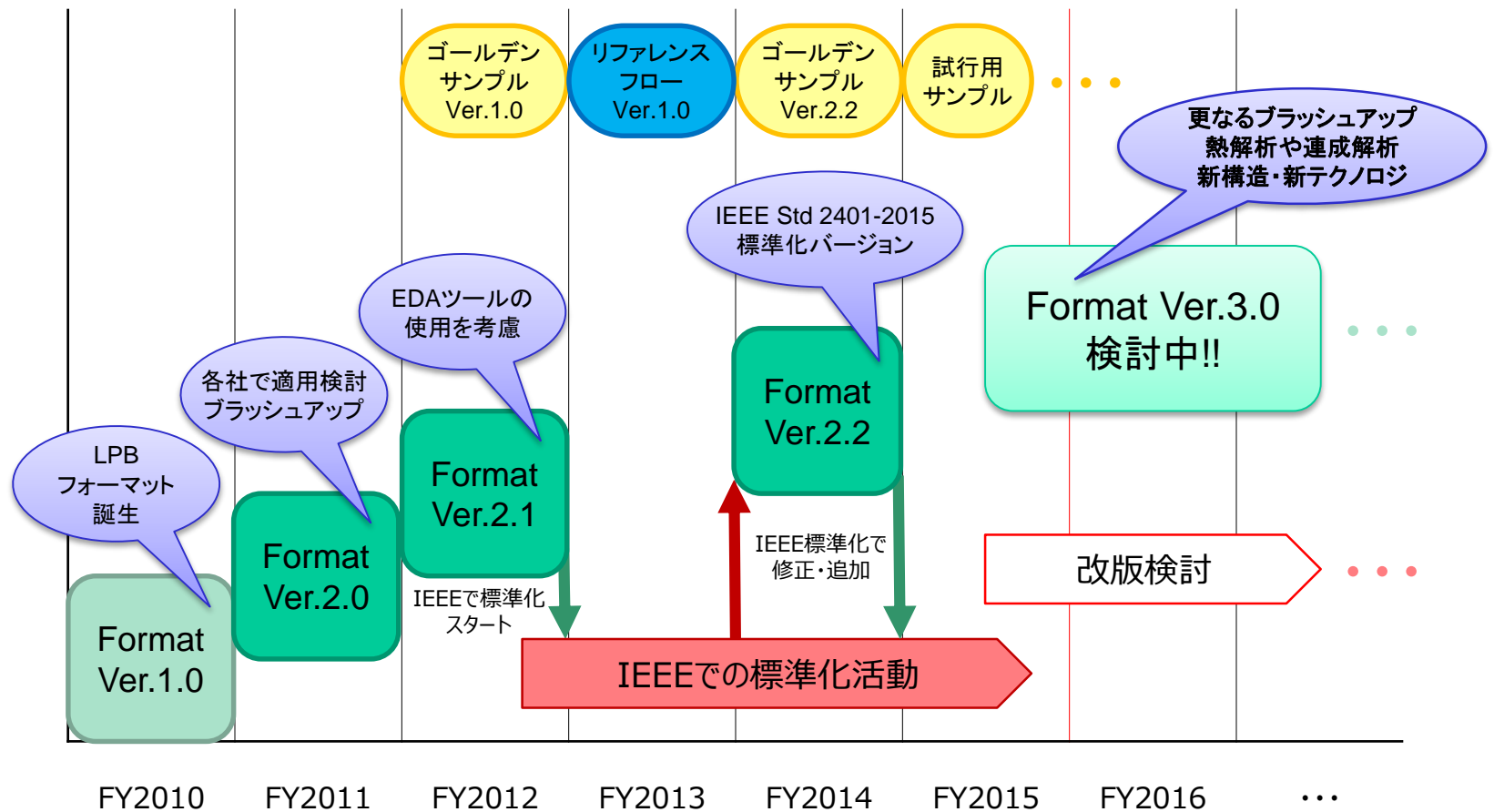
# LPBフォーマットのリリース計画

第9回LPBフォーラム

2017.3.10

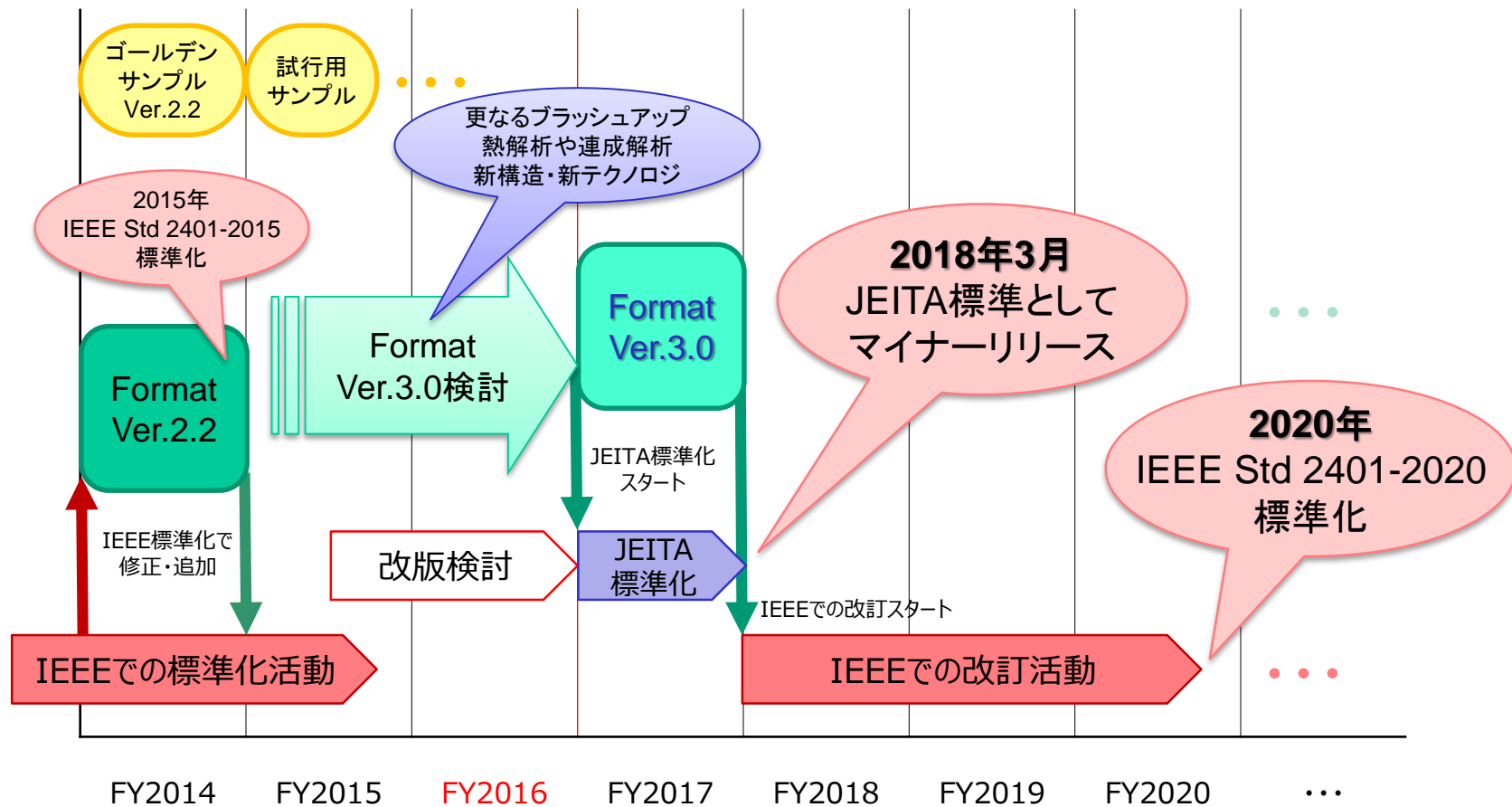
# LPBフォーマット ロードマップ ('16/3)

## ■ LPBフォーマット + aの開発状況



# LPBフォーマット ロードマップ ('17/3)

## ■ LPBフォーマット Ver.3.0 リリース計画



# LPBフォーマット ロードマップ ('17/3)

## ■ LPBフォーマット 開発計画大日程(案) [※はIEEE Std 2401-2015の実績ベース]

2017年		2018年		2019年		2020年	
1		1		1		1	
2		2		2		2	
3	LPBフォーマット リリース計画発表	3	JEITA標準 Ver.3.0 マイナーリリース	3		3	Draft作成(3rd) ※
4		4		4		4	Final Draft完成 ※
5		5		5	Draft (IEEE Style) 作成開始 ※	5	
6	JEITA標準活動開始	6	IEEE標準(改訂)活動 開始	6		6	
7	Draft (JEITA Style) 作成開始	7		7	Draft作成(1st) ※	7	
8		8		8		8	
9		9		9		9	IEEE Std 2401-2020 承認 ※
10	標準化提案(PAR) ※	10		10		10	
11		11		11		11	
12		12		12	Draft作成(2nd) ※	12	IEEE Std 2401-2020 発行 ※

# Ver.3.0でのアップグレード内容

## ■ その1 : フォーマット間の参照、制約強化



Ver.2.2までは各フォーマットの冗長性を排除し、極力独立した記述としていましたが、Ver.3.0では制約や構造定義の強化のために、C-FormatにN-FormatやR-Formatで定義されるネットやレイヤー、デザインルールを参照する項目を追加しています

これにより、

- ネット名を基準とした制約を設定できます  
制約の対象の指定が直接的で簡単になります また、インピーダンスマッチングのような一部のネットを対象としたL/Sや上下レイヤーの制約指定ができるようになります
- レイヤー名を直接指定した層設定ができます  
レイヤーを指定した制約が設定できるようになります  
埋め込み基板や3D-ICの部品配置や、ボンディングワイヤの設置点を直接指定できるようになります
- 局所的に適用したいデザインルールの領域の指定や変更ができます  
搭載部品直下の限定的なデザインルールとその領域を、搭載部品の配置にあわせて指定できるようになります

# Ver.3.0でのアップグレード内容

## ■ その2 : 熱解析も対象

New

Ver.2.2までは電気特性解析のみを対象としていましたが、Ver.3.0では熱解析も対象としています。そのため、搭載部品に外部の熱モデルを参照することや、熱解析を行う上で必要な情報定義を追加しています。

これにより、

- 外部の熱モデルを指定可能になります  
搭載部品に外部モデルとして2抵抗モデルやDelphiモデルを指定可能になります
- 周囲の熱解析に関する条件を指定できます  
C-Formatで定義されるモジュールごとに周囲の熱伝導率、輻射率、比熱容量、密度、消費電力をキャラメルボックスで設定できるようになります
- 基板の素材に熱解析関係の物性値を設定できます  
R-Formatで定義されるマテリアルに熱関連の物性値が設定可能となり、熱解析を行うための情報を定義できるようになります

# Ver.3.0でのアップグレード内容

## ■ その3 : 外部モデルの接続簡易化、IBIS強化

New

Ver.2.2までは外部モデルの接続定義を全て記述する必要がありましたが、Ver.3.0では接続をルールベースでマッチングさせる記述を追加しました  
また、外部モデルとしてIBISの対応も強化しています

これにより、

- 外部モデルの接続を自動でマッチングさせるルールを指定できます  
外部モデルの接続定義で、ポート名またはポートIDでのマッチングを指定できるようになります 対象となる外部モデルのフォーマットは、verilog、VHDL、def、xfl、ibis、ibis\_pkg、ibis\_ebd、LPBフォーマットです
- IBISの接続をピン名で指定できるようになります  
IBISの接続定義で従来のsignal nameでの指定に加え、pin nameで指定できるようになります
- IBISのPKGやEBDを指定できるようになります  
IBISのPKGやEBD記述を、独立した外部モデルIBIS\_PKG、IBIS\_EBDとして指定できるようになります

# Ver.3.0でのアップグレード内容

## ■ その4 : ユーザ拡張領域をサポート

New

Ver.3.0ではC-Format、R-Format、M-Formatの要素と属性でのユーザ独自の記述を、ユーザ拡張領域としてサポートします  
ただし、ユーザ拡張領域の識別名はJEITAで管理し、標準化を検討します  
これにより、

- ユーザは独自記述を利用できます

C-Format、R-Format、M-Formatで"ext\_識別名"で区別されたユーザ独自記述が可能となります

ユーザ定義要素 : `<ext_識別名> ~ </ext_識別名>`

ユーザ定義属性 : `ext_識別名=" ~ "`

- ユーザ拡張領域の識別名はJEITAで管理します

ユーザ拡張領域を示す識別名はユーザ申請に基づいてJEITAが発行・管理します  
申請されたユーザ拡張記述は一般公開され、標準化の検討対象となります  
(ただし、標準化を保証するものではありません)



# Ver.3.0でのアップグレード内容

## ■ 2015年(Ver.2.2)までのアイデア

Ver2.0からVer.2.2までに、LPBフォーラムなどでの議論からアイデアや要望が出るも、IEEE標準化のために見送っていた項目があり、再評価を実施  
Ver.3.0に採用する項目を決定、フォーマットへの落とし込みを実施

No.	Format	内容	Ver.3.0 対処
1	C-Format	アナログ電源のリップル率など、電圧のmin/max以外の表現ができない	必要性含めて検討する ⇒ <b>採用</b> リップル率：出力の規格として(Portの属性)か、入力の許容値として(power_domainの属性)か検討要 恐らく出力、継続検討
2	C-Format	swappable の意味	port入れ替え(I/Oセル入れ替え)可能となる表記を検討する ⇒ <b>採用</b> swappableではなく、未決定を表す属性を追加 キーワードは検討
3	C-Format	KEEPOUT領域が指定できない	検討する (R-Formatも含めて) ⇒ <b>採用</b> 要素moduleの中で定義する Keepoutの対象は部品のみ
4	C-Format	差動信号のSkewを定義するとき、Groupでmax.を定義すると、Pos-Negどっちがどっかわからない。PKG-PCBでの帳尻合わせをする場合 コントロールできない。	port の部分にskewを記載し、制約の部分は変えないというやり方はいかがか？ ⇒ <b>採用</b> portgroupに差動属性、Pos/Negの属性も追加、指定する
5	C-Format	C-FormatにはLayer定義が無いため、Layer指定ができない - padstack の ref_shape - component の placement	部品内蔵等含めた対応を考える ⇒ <b>採用</b> 要素mountに値「MIDDLE」追加、基準面からのz方向の部品積載順番の属性を追加、ユーザが値「正の整数 (TOPとBOTTOMは外に向かって数字が大きくなる、MIDDLEはTOP側からの順番とする)」を指定する
6	C-Format	次の設計フェーズに移れるかなど、判断結果の記述「Fixed」「Locked」などのプロパティを付けるか	全フォーマット対象、エレメントに指定、下位エレメントを含む ⇒ <b>採用</b> キーワードは継続検討
7	C-Format	差動信号の専用定義	⇒ <b>採用</b> No.5と同じ

# Ver.3.0でのアップグレード内容

## ■ 2015年(Ver.2.2)までのアイデア

No.	Format	内容	Ver.3.0 対処
8	C-Format	Power_domainで電源とGNDのペアとして指定できない	⇒ 採用 power_domainを要素「power」「ground」に分け、それぞれに属性portと属性min/typ/maxを指定する 複数指定可とする
9	R-Format	KEEPOUT領域が指定できない	検討する (C-Formatも含めて) ⇒ 採用 要素moduleの中で定義する Keepoutの対象は配線、VIAのみ
10	R-Format	bondingwire_def と ball_def で英語表記方法が異なる	機を見て見直す ⇒ 採用 対応済み
11	R-Format	デザインルールのエリア指定など、ユーザ由来と製造由来との区別 複数(ファイル)のエリア指定の場合の優先度	⇒ 採用 優先順位を付ける属性を追加する 値は正の整数 (大きい方が優先) 省略可能、省略時は「default」エリアは「0」、それ以外のエリアは「1」とする 同値の複数ルールは包含のみ許し、狭い領域を優先とする 属性のキーワードは継続検討
12	N-Format	各端子がどの電源系に属しているかの記述	⇒ 採用 オプションとして指定可能にする 電源系は /* PG_NET=「domain」 */ で指示 「domain」はユーザ指定 信号系に /* pow=「domain」 gnd=「domain」 */ で電源系を指定 pow、gndなどのキーワードは継続検討
13	M-Format	設計バージョンの履歴管理をフォーマット記述がサポートするか	⇒ 採用 セットを複数許可する、現状のフェーズとclass毎のフェーズを追加 キーワードは継続検討
14	C-Format	Reference記述でTouchStoneファイルの直接指定をポートするか	⇒ 採用 要素formatに値「TOUCHSTONE」を追加
15	C-Format	容量値や抵抗値など部品の特性値を記述できるようにするか	⇒ 採用 2-Portの特性値に限り記載可能にする。Reference記述ではなく、 デフォルト定義を行う。C、R、Lのみ指定可能、直列とする
16	C-Format	placementの部品名は何を用いるか 品名、製品名とは別に各社独自の名前に対応すべき	⇒ 採用 社内部品管理用の名称を記載可能にする。Placementに Symbol名を追加して対応。ref_module名は製品名=部品のmodule名

# Ver.3.0でのアップグレード内容

## ■ 2016年に検討の新規項目



2016年に熱解析や、パワーデバイスなどに対応すべくコンストレイントの拡充、IBISへの対応強化、ユーザ拡張領域などを検討  
Ver.3.0に採用する項目を決定、フォーマットへの落とし込みを実施

No.	Format	内容	Ver.3.0 対処
17	C-Format	ボンディングワイヤの設置位置を指定したい	⇒ 採用 要素moduleの中に要素componentを追加し、その中に新たに要素bondingwireを追加する R-Formatで定義されているボンディングワイヤの形状をを参照する
18	C-Format	搭載部品へ外部の熱簡易モデルの紐づけができない	⇒ 採用 要素referenceの属性formatに熱簡易モデルとして、2抵抗モデル：ctm_2resistor、Delphiモデル：ctm_delphiを追加する
19	C-Format R-Format M-Format	ユーザが独自に記述を定義できる拡張領域がほしい	⇒ 採用 C,R,Mの各フォーマットへ適用する ~はユーザ独自記述が可能とし、識別名はJEITAで管理し、標準化への反映を検討する ユーザ定義要素：<ext_識別名> ~ </ext_識別名> ユーザ定義属性：ext_識別名=" ~ "
20	R-Format	誘電体の物性値に周波数特性が記述できない	⇒ 採用 要素dielectricに要素frequency_characteristicを追加し、周波数と誘電率、誘電正接を記述できるようにする
21	R-Format	導体の物性値に温度特性が記述できない	⇒ 採用 要素conductorに要素temperature_characteristicを追加し、温度と体積低効率を記述できるようにする
22	C-Format	Reference記述でのIBISを、pin nameで接続したい	⇒ 採用 要素ibis:ref_portに属性pin_nameを追加
23	C-Format	Reference記述でのIBISのPKGとEBD記述を指定したい	⇒ 採用 要素referenceの属性formatに、IBIS_PKGとIBIS_EBDを追加し、要素connectionに、要素ibis_pkg:ref_portと、要素ibis_ebd:ref_portを追加する

# Ver.3.0でのアップグレード内容

## ■ 2016年に検討の新規項目



New

No.	Format	内容	Ver.3.0 対処
24	C-Format	Reference記述での外部モデルを参照・接続する際に、全接続を書かずに一括で接続を指定したい	⇒ <b>採用</b> 新たに要素connection_batchを追加し、属性match_byでマッチングにport_nameを使うか、port_idを使うかを指定する 対象とするformatは、verilog、VHDL、def、xfl、ibis、ibis_pkg、ibis_ebd、JLPB
25	C-Format	インピーダンスマッチングを行う場合などに、ネットごとに配線のL/Sや、上下層の配置指定や禁止を指定したい	⇒ <b>採用</b> 要素constraintの中で、新たに要素netを追加し、width、length、spacing、enclosure、keepoutを定義する
26	C-Format	部品間の配置制約(指定領域内)を定義したい	⇒ <b>採用</b> 対象を抵抗と容量に限定し、portに対する制約とする 要素portに、新たに要素dumpingと、要素decapを追加し、要求するportからの距離と抵抗値・容量値を定義する
27	C-Format	部品間の配置制約(指定領域外)を定義したい	⇒ <b>採用</b> 自モジュールに対する他モジュールの近接配置を禁止する制約とする 要素moduleに、新たに要素keepawayを追加し、領域の形状と対象層をSAME、OPPOSITE、BOTHとして定義する
28	C-Format	デザインルールを緩和する領域を定義したい	⇒ <b>採用</b> 搭載部品の領域に対して適用するデザインルールを指定する 要素placementに、新たに属性ref_rule_nameと、属性sizingを追加し、R-Formatで定義するデザインルールと、搭載部品が持つ形状のサイジング量を指定する
29	C-Format	熱解析を行うための筐体を定義したい	⇒ <b>採用</b> 自モジュールの周囲の仕様をキャラメルボックスで指定する 要素moduleに、新たに要素boundary_specificationを追加し、キャラメルボックスの形状、配置座標、熱伝導率、輻射率、比熱容量、密度、消費電力を定義する
30	C-Format R-Format	熱解析を行うための物性値が定義できない	⇒ <b>採用</b> 要素unitに、熱伝導率、比熱容量、密度の定義を追加する R-Formatの要素material_defの要素conductor、要素dielectricに熱伝導率、輻射率、比熱容量、密度を定義する

# 最後に

LPBフォーマットVer.3.0は、標準化に向け動き出しました  
2017年に1つのやま場をむかえ、2020年へと続きます  
LPBフォーラムにご参加いただき、ご意見・ご要望をお寄せください  
よろしく願いいたします