

CFormatの使い方 Ver1.1

2019/3/3

<http://www.lpb-forum.com/>

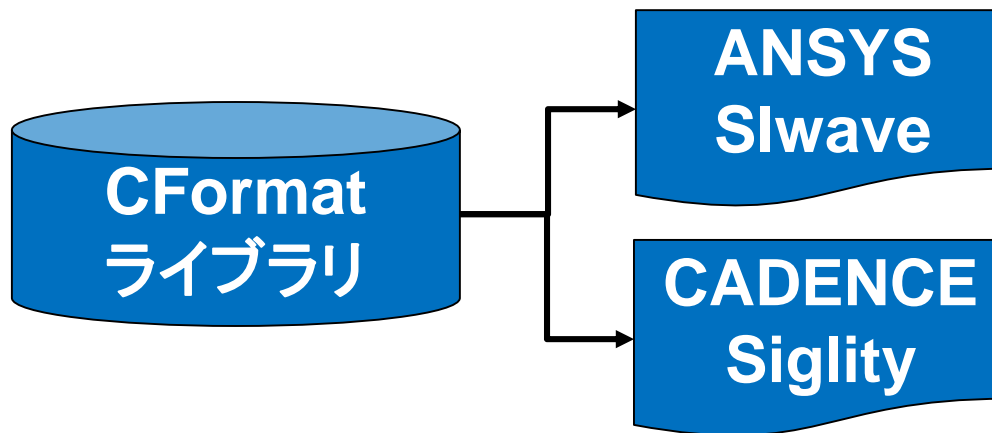
What's New

2019/2/28:

- 村田製作所様から積層セラミックコンデンサのCFormatライブラリが公開されました。

<https://www.murata.com/ja-jp/tool/c-format>

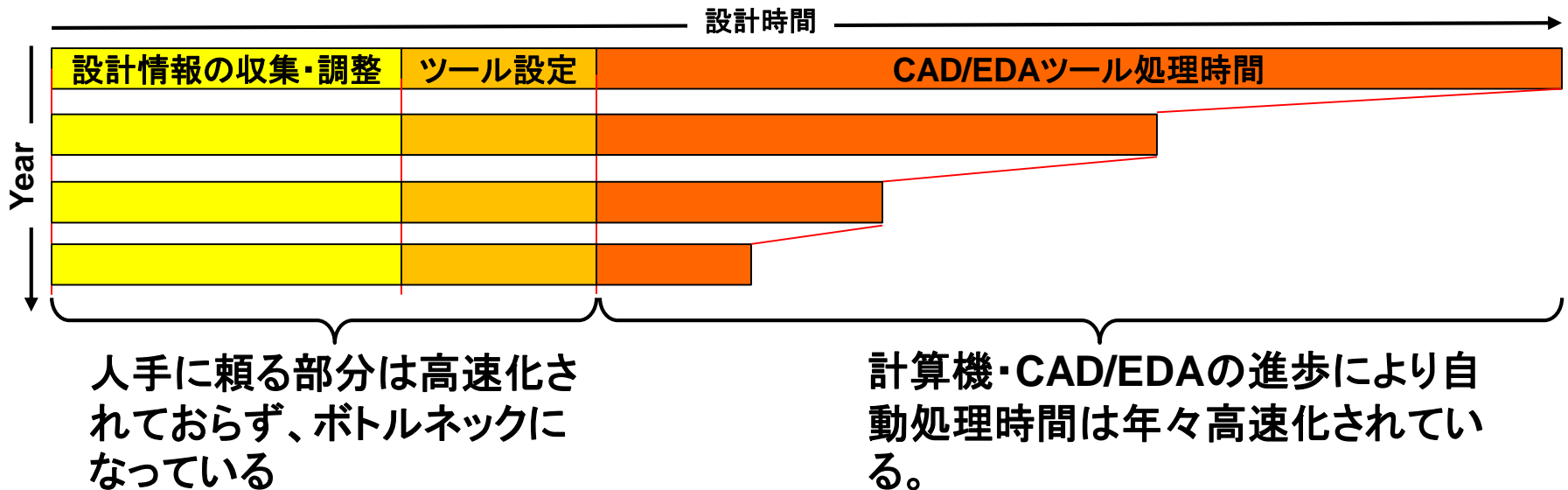
- CFormatライブラリをANSYS/Siwave, CADENCE/Siglitのライブラリに変換できます。



LPB FORMATの目的

設計現場における課題

- 計算機の高速化・アルゴリズムの進歩によりCAD/EDAの処理時間は短縮されています。
- しかし、人手に頼る設計情報の収集・調整、ツール設定に費やす時間は手つかずのままです。



開発期間の短縮には、人手に頼っている部分の効率化が必要

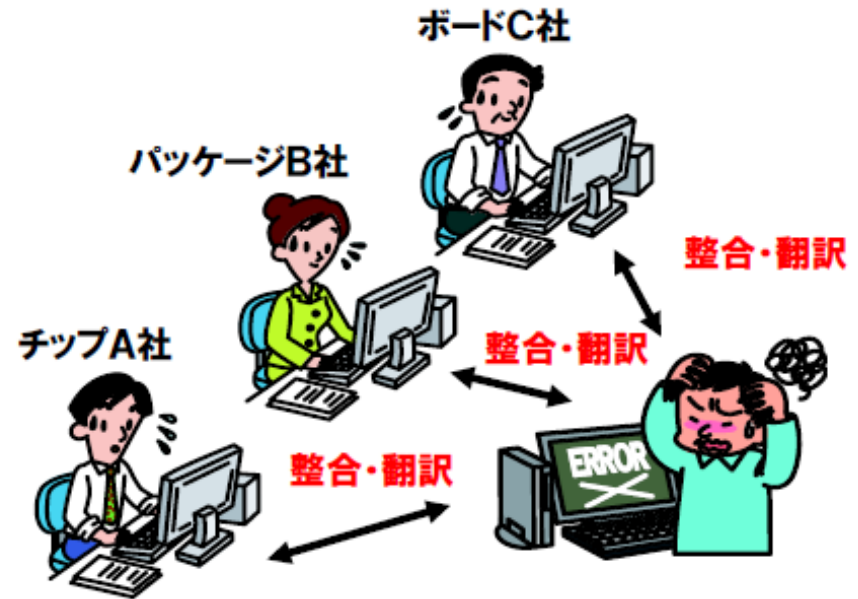
LPB Formatの目的

水平分業では、人手で設計文化の違いを調整することが必要となります

- 仕様書の書式の違いの調整
- 使用CAD/EDAが違いによるデータフォーマットの変換 など

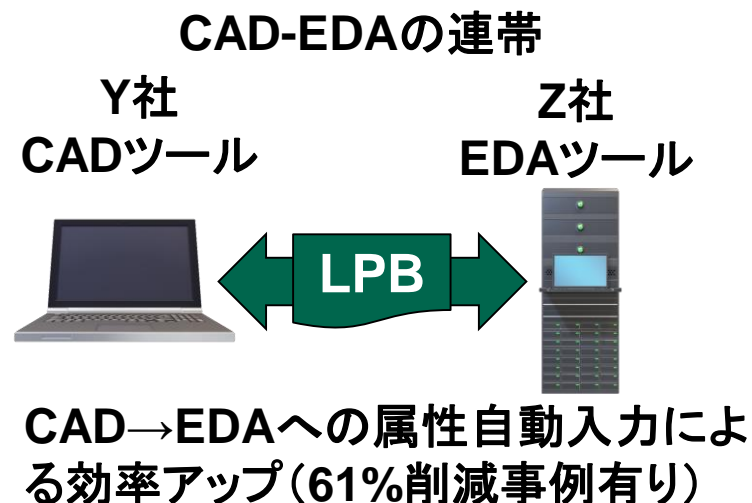
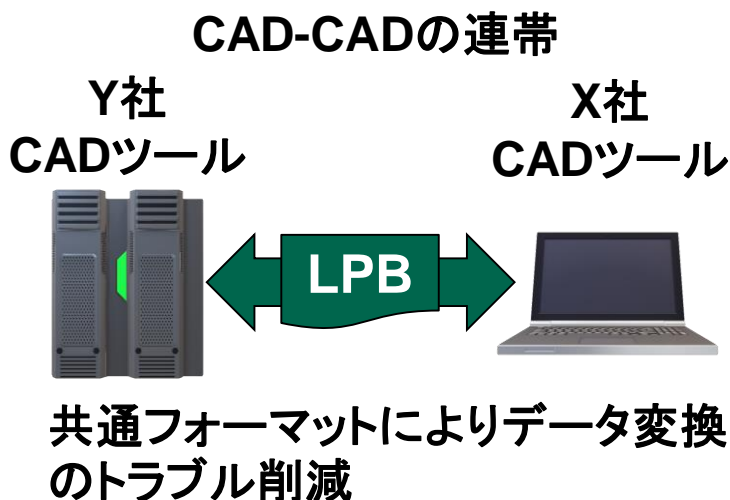
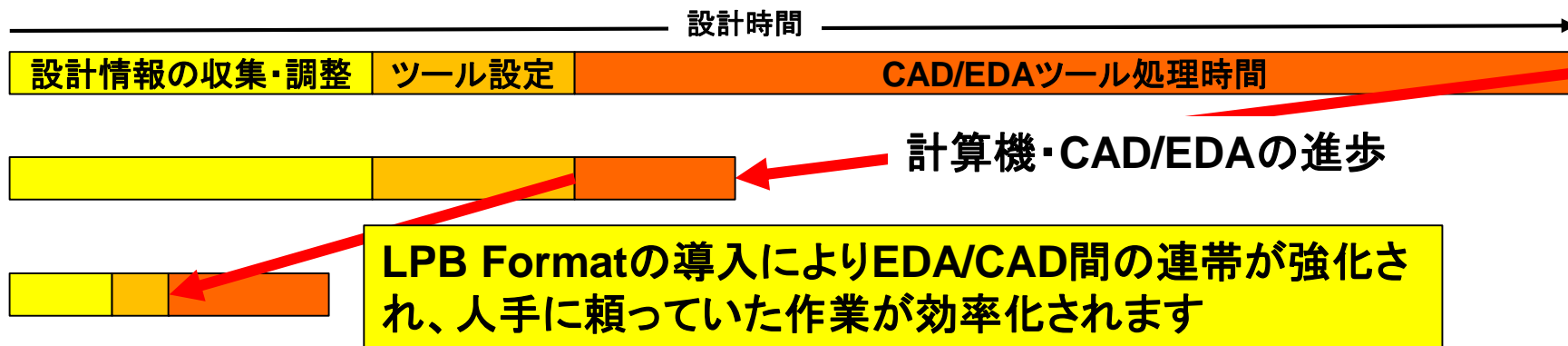
その他、日々の業務では、

- シミュレーションモデルの収集
 - CAD/EDAのセットアップ など
- 人手による作業が必要となります。



LPB Formatは人手に頼っていた部分の効率化を目的とした標準規格です

LPB Format 活用効果



事例は以下を参照ください:

http://www.jeita-sdtdc.com/jeita-edatc/wg_lpb/home/docs/eds11nov_sp02.pdf

LPB FORMATの概要

LPB Formatとは

LPB Formatは従来、人手に頼っていた作業を効率化するためにIEC/IEEEで標準化されたデータフォーマットです

Preview

<https://www.techstreet.com/products/preview/1908285>

http://jeita-sdtd.com/publishedmaterials/lpbformatv3_preview_download/

サンプル

http://jeita-sdtd.com/publishedmaterials/lpbformat_download/

IEC 63055:2016

Format for LSI-Package-Board Interoperable design

<https://webstore.iec.ch/publication/26181>

IEEE2401-2015

IEEE Standard Format for LSI-Package-Board Interoperable Design

https://www.techstreet.com/standards/ieee-2401-2015?product_id=1908285



LPB Formatの採用状況

LPB Formatの採用状況

- 13社以上のCAD/EDAメーカーがデータインターフェイスとして採用しています
左図
- リファレンスボードの設計データをLPB Formatを利用して公開しています
東芝他
- 部品ライブラリをLPB Formatを利用して提供しています
村田製作所、東芝ディスクリート他



データフォーマットを標準化

LPB Formatは以下の5つのデータファイルで構成されています

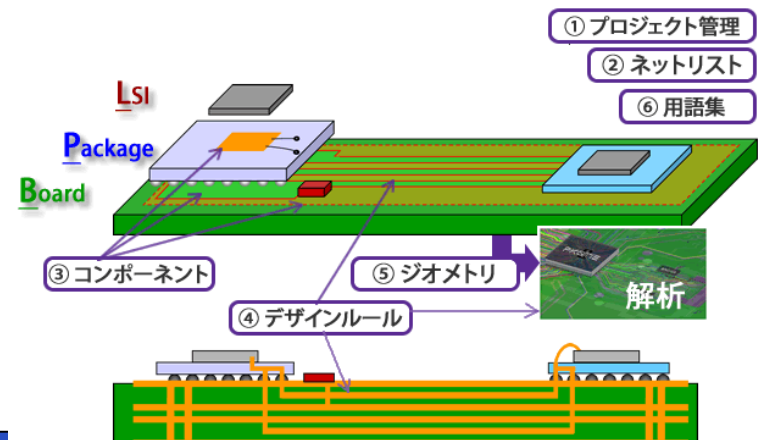
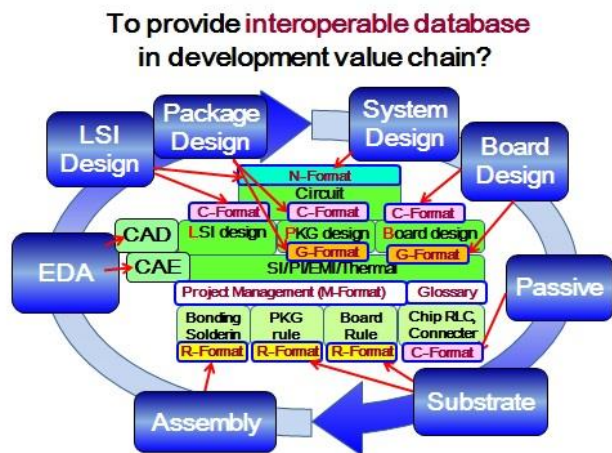
M-Format : プロジェクト管理

N-Format : ネットリスト。部品間の接続情報を定義します

C-Format : コンポーネント定義。設計制約や部品のフットプリントやシミュレーション用のモデルを定義します

R-Format : デザインルール。基板構造やLine & Space等のルールを定義します

G-Format : Layer Stackup構造(2次元)の図形データ



LPB FORMATで 何ができるのか？

どうやって使うの？

LPB Formatを使った事例は下記のURLで公開されています

http://jeita-sdtdc.com/publishedmaterials/lpbforum_documents/

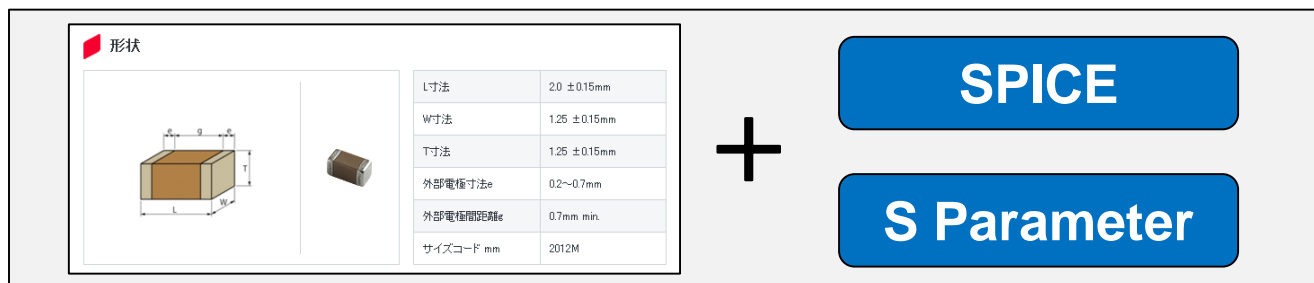
ここでは、CFormatを題材に使用例を説明します。

CFormatとは以下の情報を統合したものです。

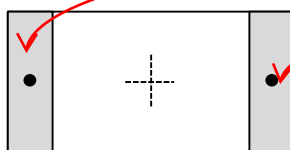
- 部品のFoot Print、端子の位置、形、名称などの形状情報
- SPICEやIBIS等のシミュレーションモデル
- Skew、インピーダンス、遅延条件などの設計制約
- フロアプラン、部品座標などのレイアウト情報

CFormatの基本形

最も基本的なCformatは、部品のFootPrintとシミュレーション用モデルをパッキングしたものです



CFormatでFootPrint(端子)とモデルの入出力ノードの対応が定義されています



```
.SUBCKT GA243DR7E2472MW01 Port1 Port2
C01 Port1 N01 4.77e-09
R01 Port1 N01 2.00e+09
.....
```

村田製作所から提供されるコンデンサー部品のCFormatも、この形のもので

CFormatで何ができるの？

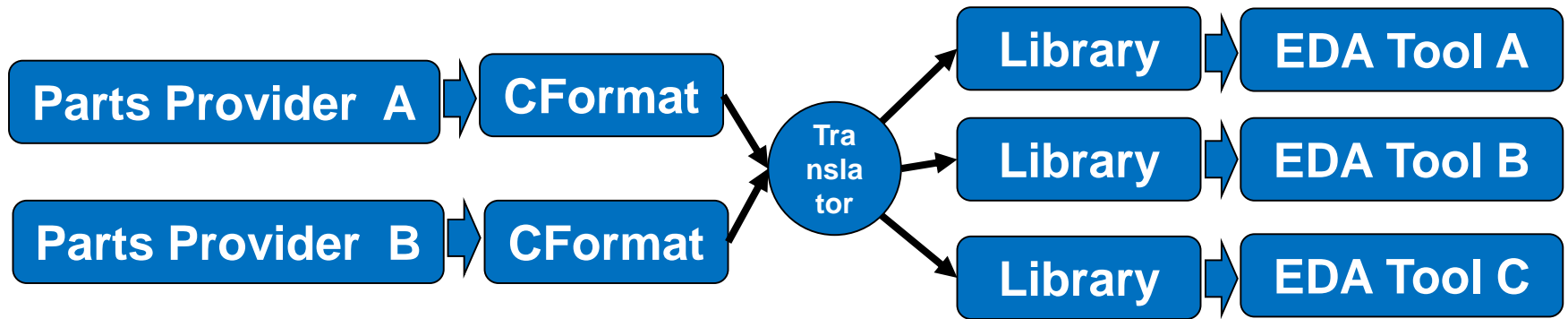
物理形状とシミュレーションモデルが統合されているので、シミュレーションツールのセットアップを効率的に行うことができます

LPB Forumでは以下のようなCFormatを使った事例が発表されています

- **LPBフォーマットとIBIS5.0を活用したPI解析**
http://jeita-sdtec.com/download/lpbforum/20170310_LPB-Forum/6_Pi_analysis_with_LPB_Format_and_IBIS5.pdf
- **Cフォーマット対応によるコンデンサ最適化のTAT短縮**
http://jeita-sdtec.com/download/lpbforum/20160311_LPB-Forum/6a_DecouplingOptimizationByCFormat.pdf
- **スクリプトによるLPBシミュレーション効果**
http://jeita-sdtec.com/download/lpbforum/20160311_LPB-Forum/6b_LPBSimulationByScripting.pdf

CFormatで何ができるの？

部品ベンダーから供給されたCFormatをEDAツールのライブラリに変換することができます(※)。部品ベンダー毎に、EDAツール毎にライブラリを準備する手間がなくなります。



現時点(2019/1)では村田製作所からコンデンサー部品のCFormatが提供されています。JEITAは様々な部品ベンダーに対しCFormatの提供を働きかけています。

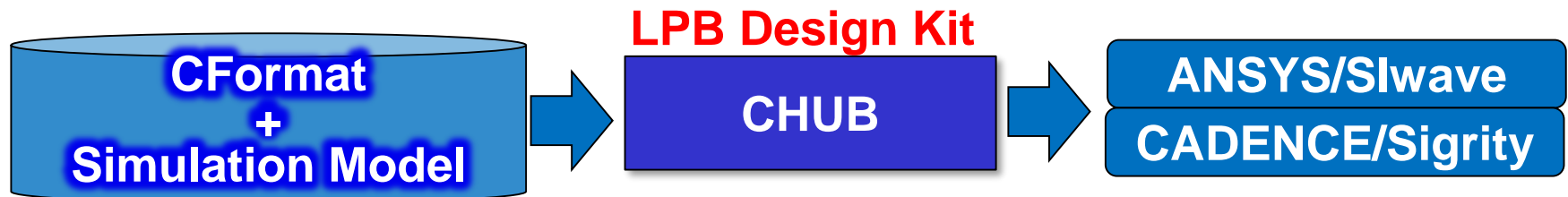
(※)CFormatを入力可能なツールは変換の必要は有りません。

- http://jeita-sdtdc.com/download/lpbforum/20150319_LPB-Forum/3-2_DesignForce_Zuken.pdf
- <http://gemdt.com/gempackage/>

どうやってCFormatを変換しますか？

- JEITAはライブラリ変換やBOM管理の事例として、村田製作所公開のCFormat(コンデンサー)をANSYS/Slwave, CADENCE/Sigrityのライブラリに変換するツール(LPB Design Kit/CHUB)を提供しています

<http://www.lpb-forum.com/lpb-open-source-project/download/>



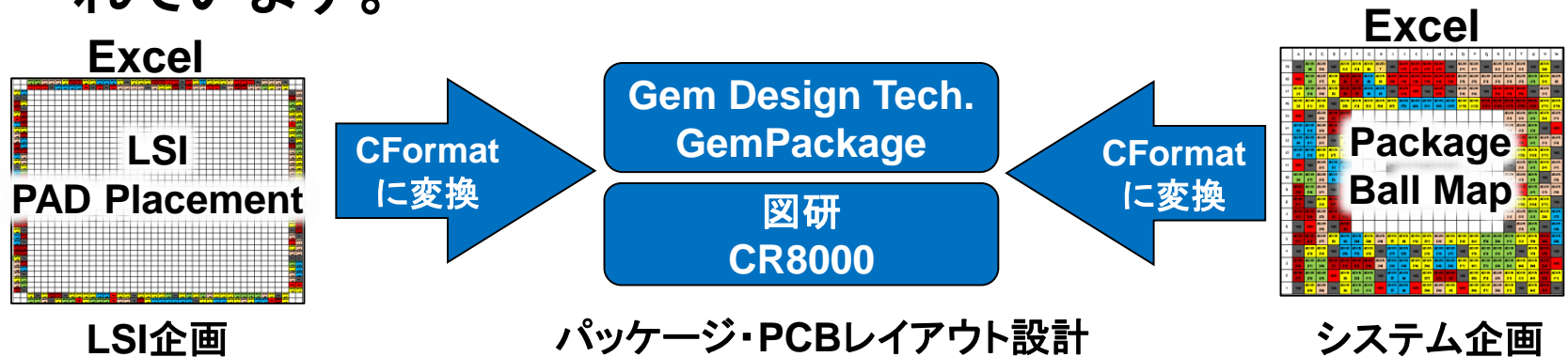
- CFormatは標準的なXMLフォーマット(※)で定義されているので変換ツールを自作することも容易です
- LPB Design Kitはソースコードも公開しています。ツールを自作する上での参考にしてください。

その他の事例

LPB Formatはロイヤリティ・フリーのデータフォーマットです。
社内の設計インフラの構築に使用することができます

例えば...

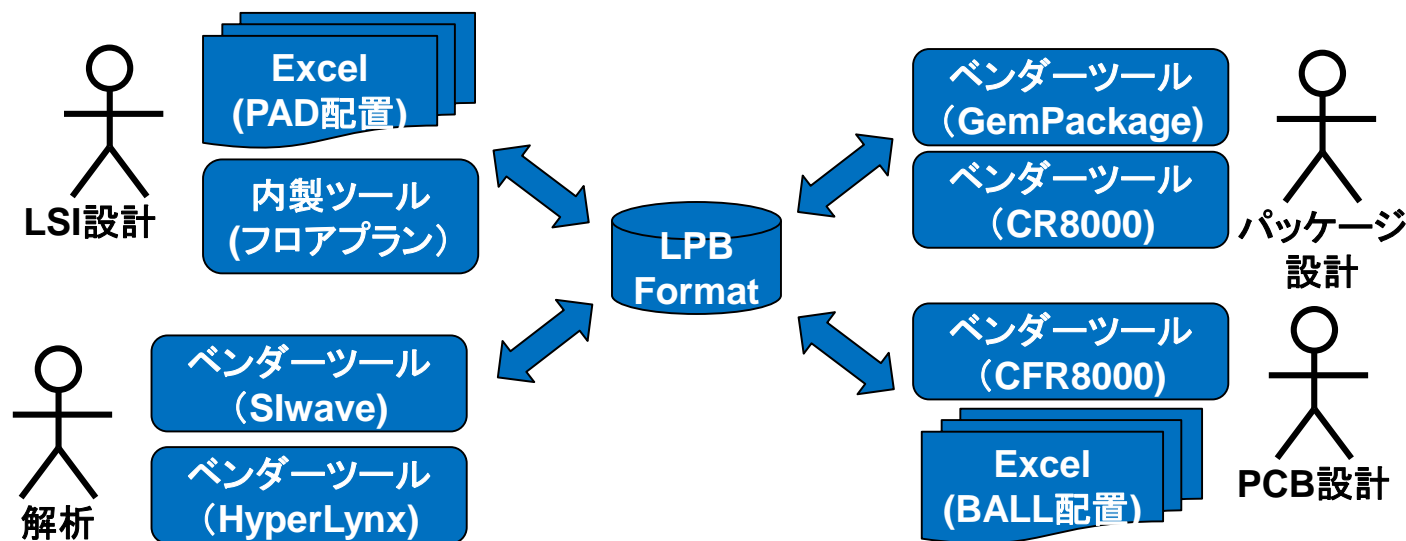
Excelで設計書を作ることは、しばしば行われています。しかし設計課により書式がバラバラな為、EDAツールとのインターフェイスが取ることが困難です。シンプルなプログラムでExcelをCFormatに変換しEDAツールに取り込むことができます。LPB Design Kit(※)には、参考プログラムが含まれています。



設計インフラの構築にご興味のある方は、

- LPB Formatを核とした設計インフラの事例は、公開が難しいため資料がありません。ご興味のある方は、LPBフォーラム等のイベントにご参加ください

<http://jeita-sdtec.com>



デザインキット (C-HUB) で出来るようになること

ANSYS/Slwaveでの応用例

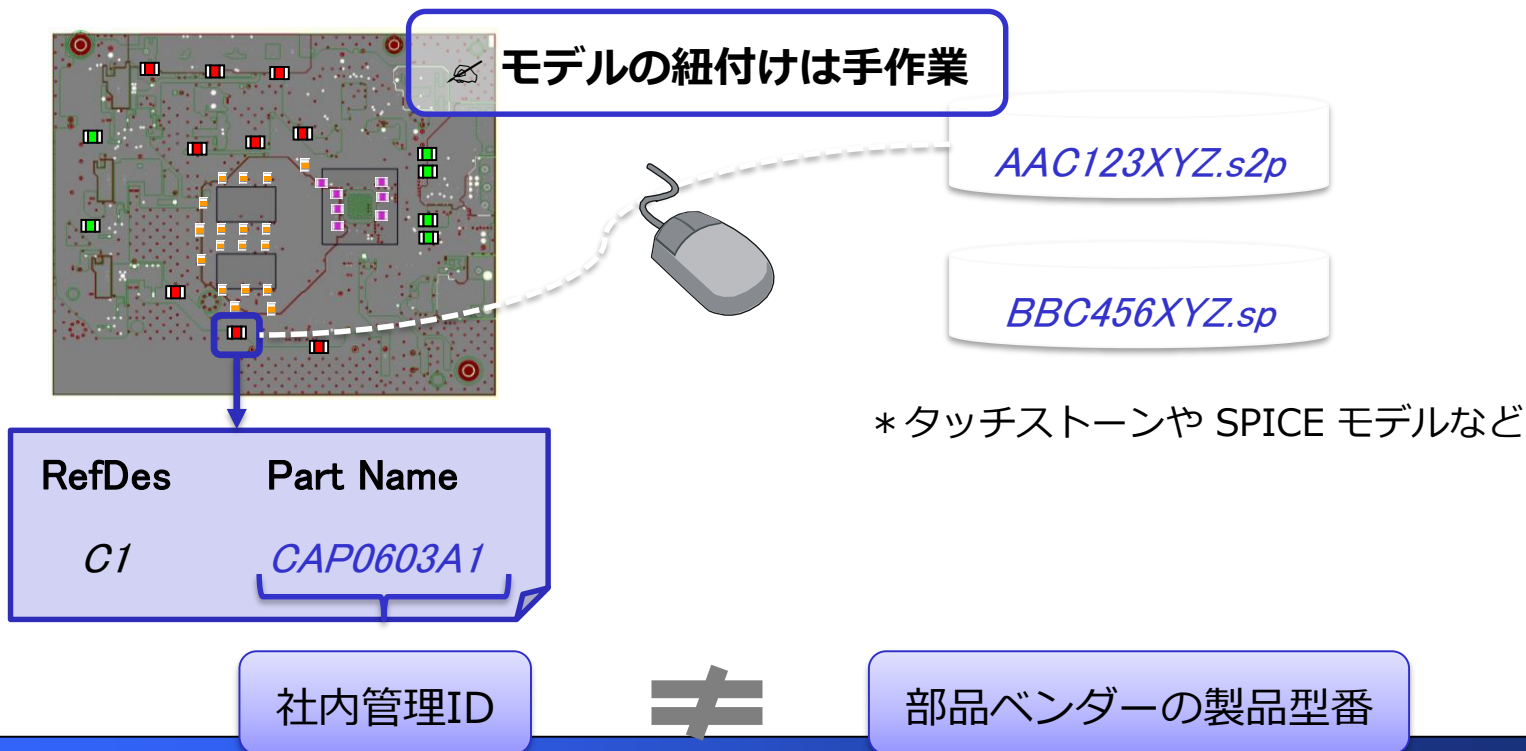
デザインキット (C-HUB) で出来るようになること

コンデンサのシミュレーション設定 (C-フォーマット無し)

- 各コンデンサに電気特性を表現したモデルを紐付けするのだが・・・

【レイアウトの部品情報】

【設定したい電気特性モデル】



デザインキット (C-HUB) で出来るようになること

コンデンサのシミュレーション設定 (C-フォーマット無し)

- 部品ベンダー提供の電気特性モデルは？

【設定したい電気特性モデル】

AAC123XYZ.s2p

BBC456XYZ.sp

*タッチストーンや SPICE モデルなど

- ✍ 部品ベンダー毎に提供の仕方は異なる
- ✍ EDA ベンダー毎にライブラリは異なる



部品ベンダーの製品型番

デザインキット (C-HUB) で出来るようになること

各フォーマットに含まれる情報

基板設計者 (または CAD) 提供

部品ベンダー提供



JEITA

File Convert Window Help

0201M 1005M 0603M 0402M

	Category	File	Module	InHouseName	FootPrint
1	0201 M	GRM011 R60 J...	GRM011 R60 J...	NB1	0201 M_1 /fp_0: ▾
2	0201 M	GRM011 R60 J...	GRM011 R60 J...	NB2	0201 M_1 /fp_0: ▾
3	0201 M	GRM011 R60 J...	GRM011 R60 J...	NB3	0201 M_1 /fp_0: ▾
4	0201 M	GRM011 R60 J...	GRM011 R60 J...	NB4	0201 M_1 /fp_0: ▾
5	0201 M	GRM011 R60 J...	GRM011 R60 J...	NB5	0201 M_1 /fp_0: ▾
6	0201 M	GRM011 R60 J...	GRM011 R60 J...	NB6	0201 M_1 /fp_0: ▾
7					
8					
9					
10					

C-HUB

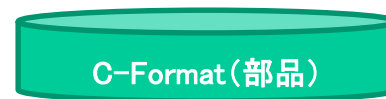


• “社内管理ID”と
“製品型番”対照表



*Import Setup File

村田製作所様が提供開始!



- 部品情報
- 製品型番
- 電気特性モデル

SPICEモデル

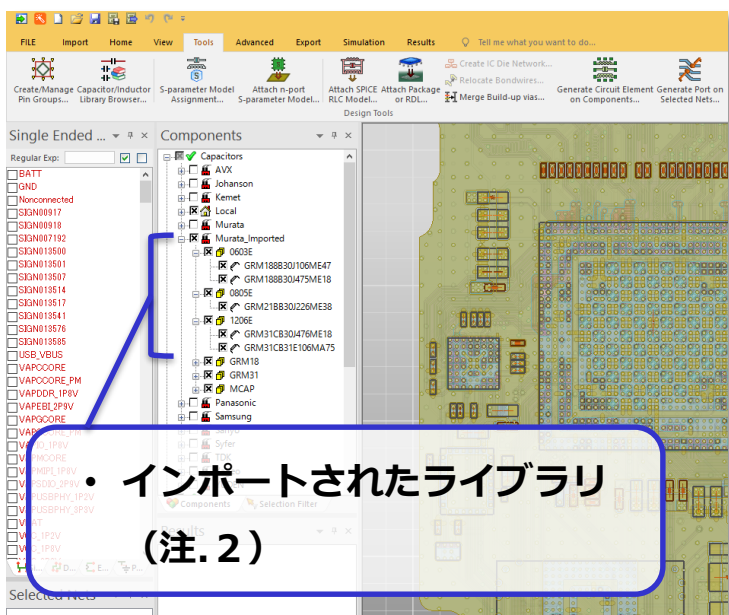
Touchstone

- 電気特性モデル

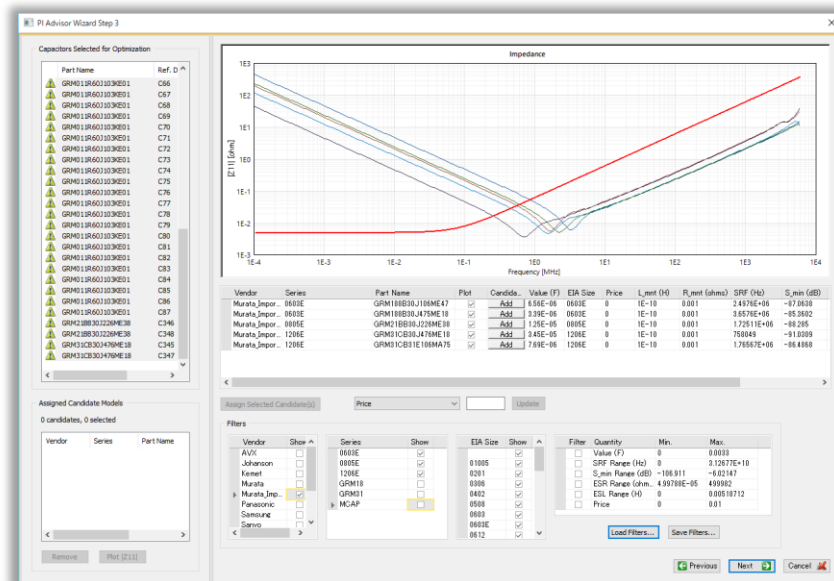
デザインキット (C-HUB) で出来るようになること

C-HUB でインポートされたライブラリの活用 (注. 1)

- コンデンサの電気特性の自動割り当て (名前で自動紐づけ)
- コンデンサの選択・配置の最適化 (PI Advisor)



ANSYS SIwave



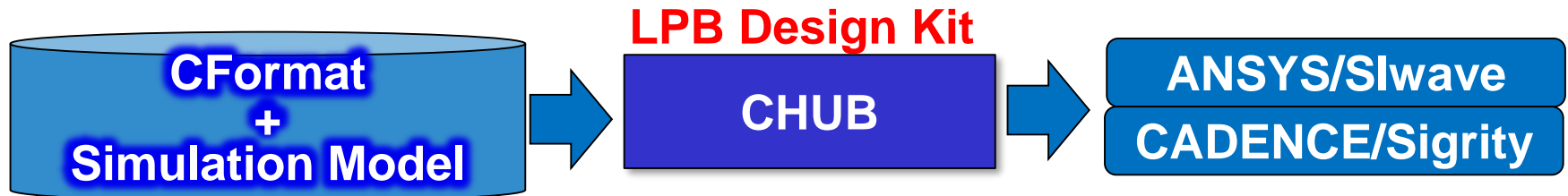
PI Advisor (ANSYS SIwave)

注. 1 弊社製品は C-フォーマットへ未対応な為、.cmpファイルなどで基板の部品情報をインポートする必要があります。
注. 2 本テスト時には社内管理 ID は設定せず、村田製作所さんの製品型番をそのまま採用しています。

LPB DESIGN KIT CHUBの使い方

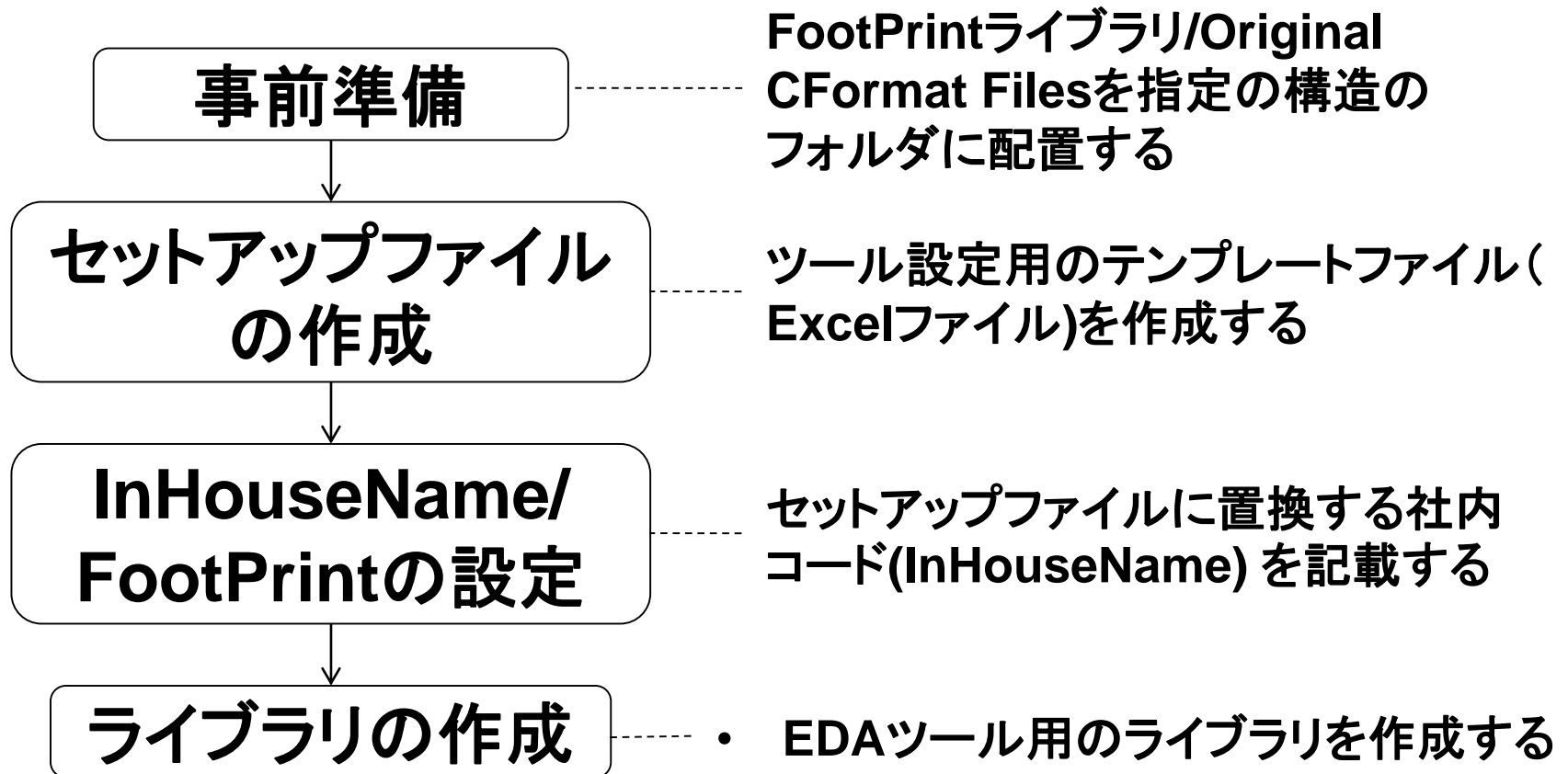
CHUBとは？

- CHUBとは、村田製作所公開のCFormat(コンデンサー)をANSYS/SIwave, CADENCE/Sigrityのライブラリに変換するツールです。



- その他、以下の機能を持っています。
 - 部品名やファミリー名を社内管理用の社内コードに変更
 - 部品ベンダー設定のFootPrintを、製造ルールにあわせたFootPrintに変更

CHUBを使ったライブラリ変換処理の流れ



LPB DesignKitをインストールする

下記URLからDesignKitを申し込んでください。

<http://www.lpb-forum.com/lpb-open-source-project/download/>

- ダウンロード用のURLが記載されたメール返送されます。
- ダウンロードしたzipファイルを展開してください。
- どこにインストールしても動作します。任意のフォルダーに展開してください。

村田製作所のライブラリをインストールする

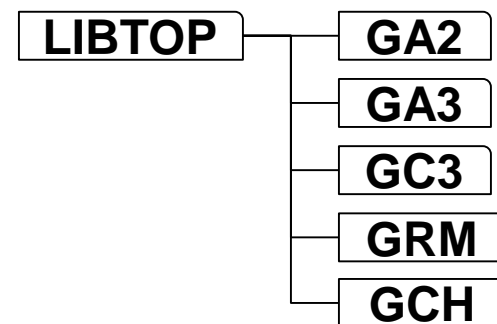
1. FTPからダウンロード

<https://www.murata.com/ja-jp/tool/c-format>

Zipファイルをダウンロードする。シリーズ毎にZIPされているので、必要な分だけダウンロードする。(全部、落としてもOK)

2. 落としたZIPファイルを展開する。

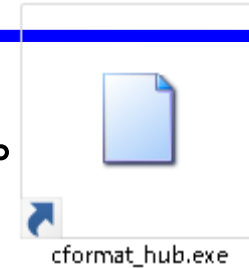
ここでは LIBTOP というフォルダを作り、その下でZIPを展開したとします(左図)。



3. ダウンロードしたzipファイルには、CFormatおよびSPICEモデル、Sparameterが含まれています。

CHUBを起動する

cformat_hub.exeをダブルクリックするとCHUBが起動します。



コマンド実行ボタン

Vendor	InHouse
1 GA2	
2 GA3	
3 GC3	
4 GCD	
5 GCE	
6 GCH	
7 GCJ	
8 GCM	
9 GCQ	
10 GGD	
11 GGM	
12 GJ4	
13 GJ8	

Category	Family	File	Module
1 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
2 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
3 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
4 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
5 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
6 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
7 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
8 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
9 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
10 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
11 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...
12 2828_1111	GQM	GQM22M5C2H...	GQM22M5C2H...

Messages

2018-12-25 17:28:18: INFO : 2012_0805
2018-12-25 17:28:27: INFO : 3216_1206
2018-12-25 17:28:32: INFO : 3225_1210
2018-12-25 17:28:33: INFO : 1608_0603
2018-12-25 17:28:56: INFO : 1005_0402


Progress

シリーズ名リスト
ベンダー提供のシリーズ名を、社内管理名称に変更するときに入力します。
詳細は次々ページ参照

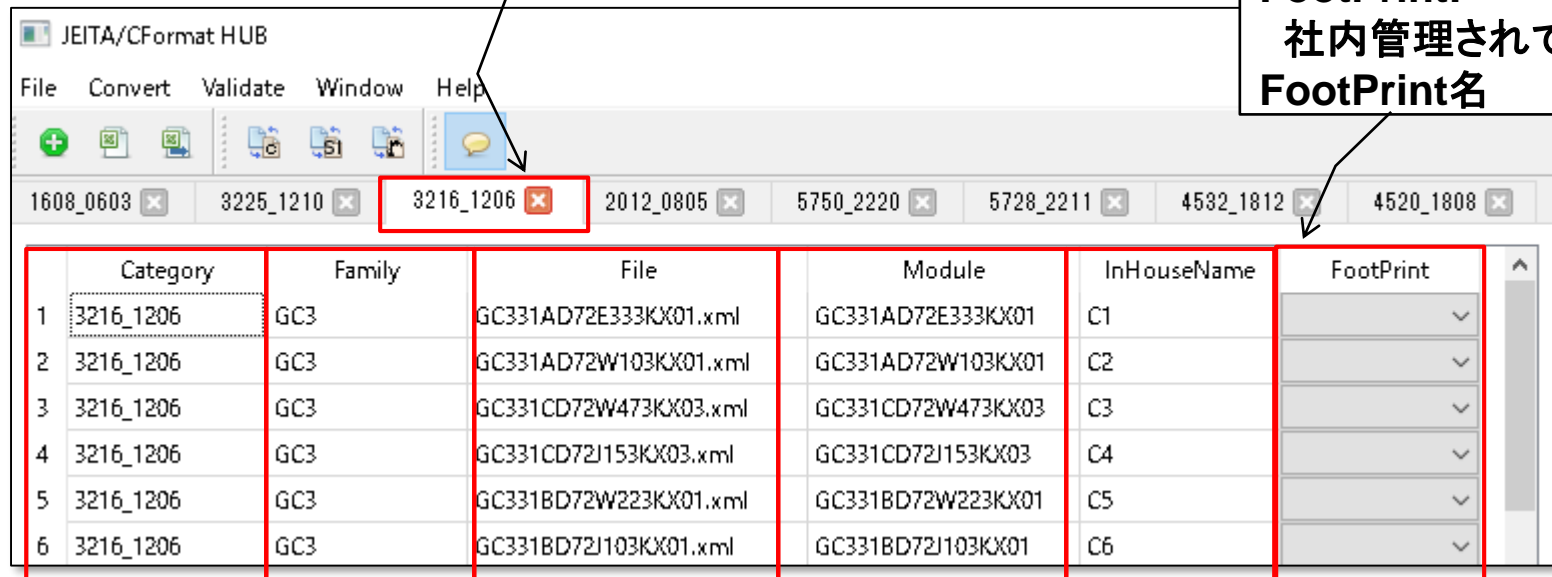
部品リスト:
ここに表示されている部品が管理対象となります。
詳細は次ページ参照

実行中のメッセージ
を表示します

表示画面の説明(部品リスト)

3216_1206  Category(部品サイズ)毎に分類されている。不要なカテゴリは×で削除する

FootPrint:
社内管理されている
FootPrint名



	Category	Family	File	Module	InHouseName	FootPrint
1	3216_1206	GC3	GC331AD72E333KX01.xml	GC331AD72E333KX01	C1	▼
2	3216_1206	GC3	GC331AD72W103KX01.xml	GC331AD72W103KX01	C2	▼
3	3216_1206	GC3	GC331CD72W473KX03.xml	GC331CD72W473KX03	C3	▼
4	3216_1206	GC3	GC331CD72I153KX03.xml	GC331CD72I153KX03	C4	▼
5	3216_1206	GC3	GC331BD72W223KX01.xml	GC331BD72W223KX01	C5	▼
6	3216_1206	GC3	GC331BD72I103KX01.xml	GC331BD72I103KX01	C6	▼

Category :
部品のサイズコード
JIS_EIA

Family:
シリーズ名

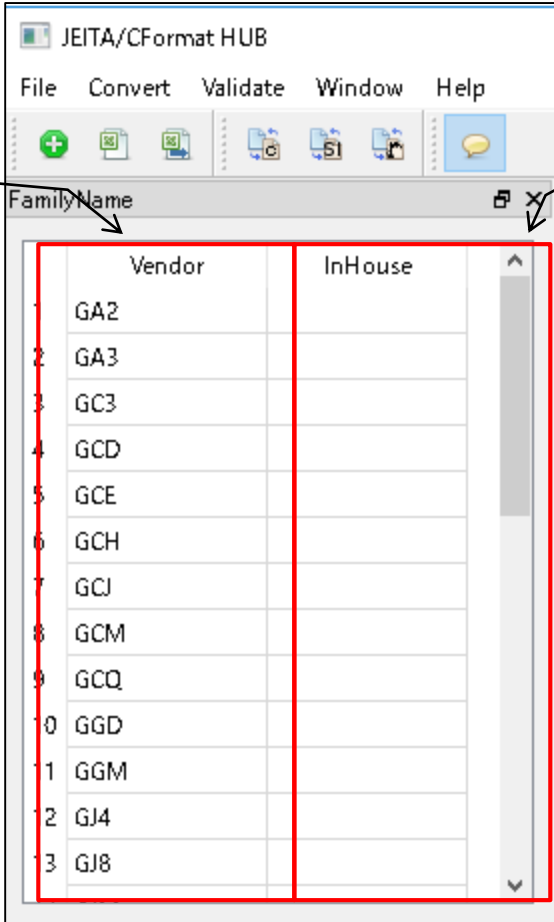
File:
Cformatファイル名

Module:
部品名

InHouseName:
社内管理コード
ベンダー提供の部品
名を社内管理コード
に変更する場合は、
ここに入力します

表示画面の説明(シリーズ名リスト)

ベンダー提供のシリーズ名



The screenshot shows the JEITA/CFormat HUB application window. The title bar reads 'JEITA/CFormat HUB'. The menu bar includes 'File', 'Convert', 'Validate', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons. The main area is a table titled 'FamilyName' with two columns: 'Vendor' and 'InHouse'. The table contains a list of series names: GA2, GA3, GC3, GCD, GCE, GCH, GCJ, GCM, GCQ, GGD, GGM, GJ4, and GJ8. A red box highlights the entire table area. An arrow points from the text 'ベンダー提供のシリーズ名' to the 'Vendor' column header.

	Vendor	InHouse
1	GA2	
2	GA3	
3	GC3	
4	GCD	
5	GCE	
6	GCH	
7	GCJ	
8	GCM	
9	GCQ	
0	GGD	
1	GGM	
2	GJ4	
3	GJ8	

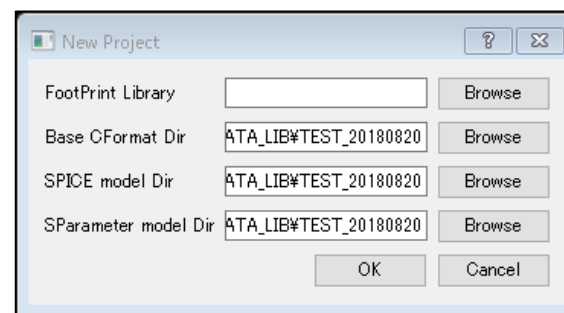
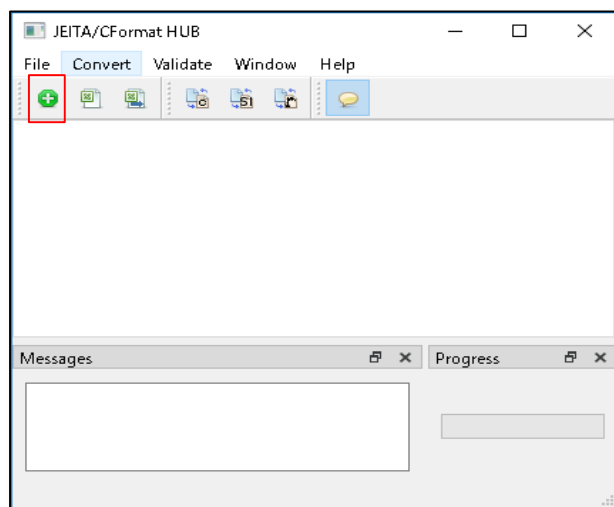
ベンダー提供のシリーズ名を社内管理名称に変更する場合は、ここに社内名称を入力します

(※)このリストはExcelファイルから作ることができます。
Excelファイルを作成してCHUBに入力することを推奨します。

村田製作所ライブラリを入力する

1. File -> New project

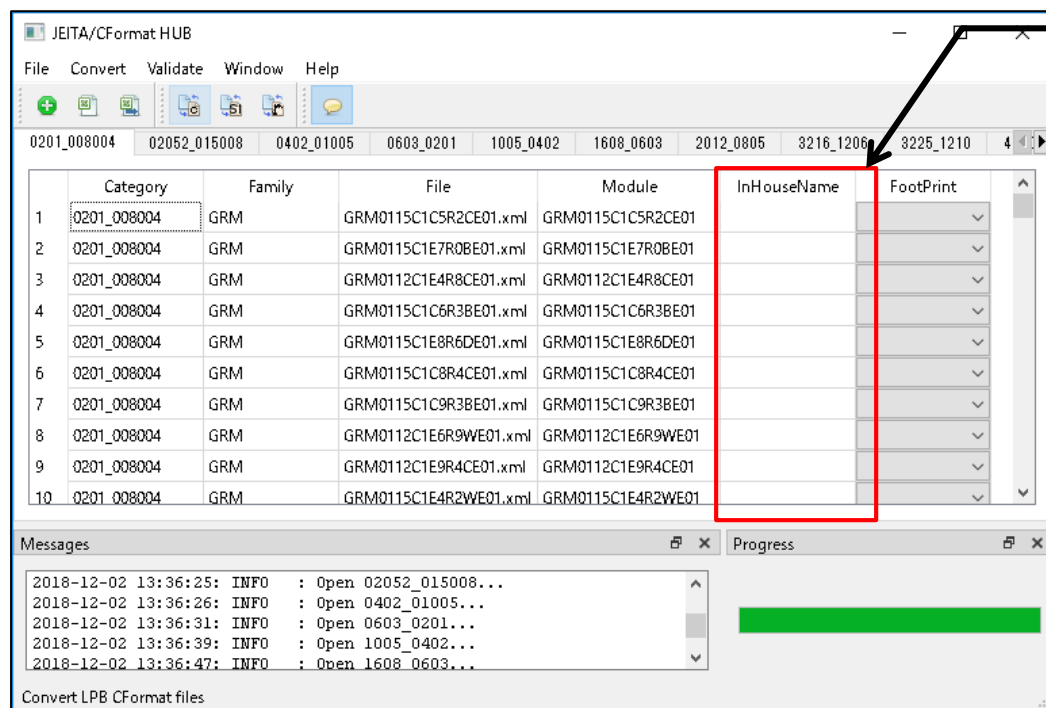
- FootPrint Library : FootPrintライブラリを使わない場合は入力しない
- Base CFormat Dir : Browseボタンを押して、村田製作所のライブラリを保存したフォルダを選択する(※)
- SPICE model Dir, Sparameter model Dir : 自動的に入力されるので、何もする必要なし。



(※)選択したフォルダの下にあるすべてのファイルが変換対象となります。LIBTOPを選ぶと処理に非常に時間がかかります。シリーズ毎に処理することを推奨します。

村田製作所ライブラリを入力する

- OKを押す。CFormatの入力が始まる。ファイルの数が多いと、しばらく時間がかかります。
LIBTOPを選ぶとCore 2 Duo T7250(10年ぐらい前のノートPC)では20分ぐらいかかります。気長に待ってください。Doneが表示されれば終了。
YESを押して閉じてください。



村田製作所の部品名を社内管理コードに変換変更する場合は、ここに社内コードを入力します。
Exceファイルから設定する方法もあります。次ページを参照してください。
社内管理コードに変換しない場合は飛ばしてください。

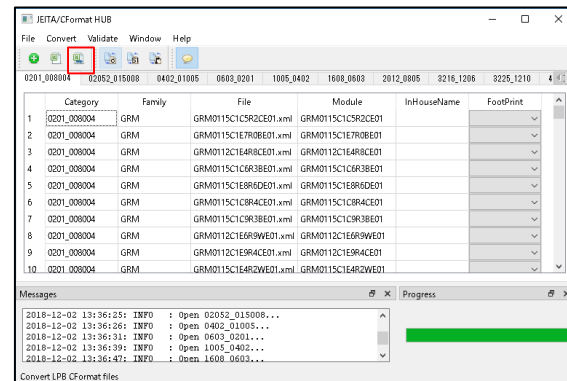
社内管理設定ファイルを作る

3. File -> Export Setup file



設定ファイル(Excel形式)を保存します

Excelファイルには複数のシートが含まれています。



- FamilyName : ベンダー提供のシリーズ名を社内管理名称に変更する場合に入力します。
- その他 : Category(部品サイズ)毎にシートが分かれています。ベンダー提供の部品名を社内管理コードに変更する場合に入力します。不要はサイズはシート自体を削除してください。

社内管理設定ファイルを編集する

4. シリーズ名(FamilyName)

ベンダー提供のシリーズ名を社内管理名称に変更する場合に入力します。変更しない場合は編集不要です。

	A	B	C	D	E	F	G
1	# 2018/12/24 13:46:55						
2	//base_cfor	C:\Users\%akiw%\Development\MU					
3	//footprint_lib						
4	//spice_dir	C:\Users\%akiw%\Development\MU					
5	//spara_dir	C:\Users\%akiw%\Development\MU					
6	#Vendor	#InHouse					
7	GA2						
8	GA3						
9	GC3						
10	GCD						
11	GCE						
12	GCH						
13	GCJ						
14	GCM						
15	GCQ						
16	GGD						
17	GGM						

#InHoust列に社内管理名称を入力してください。
入力されていない場合(空欄の場合)はシリーズ名は変更されません

社内管理設定ファイルを編集する

5. 部品名

Category(部品サイズ)毎にシートが分かれています。

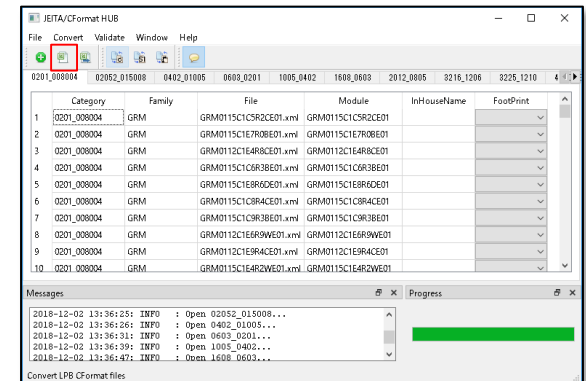
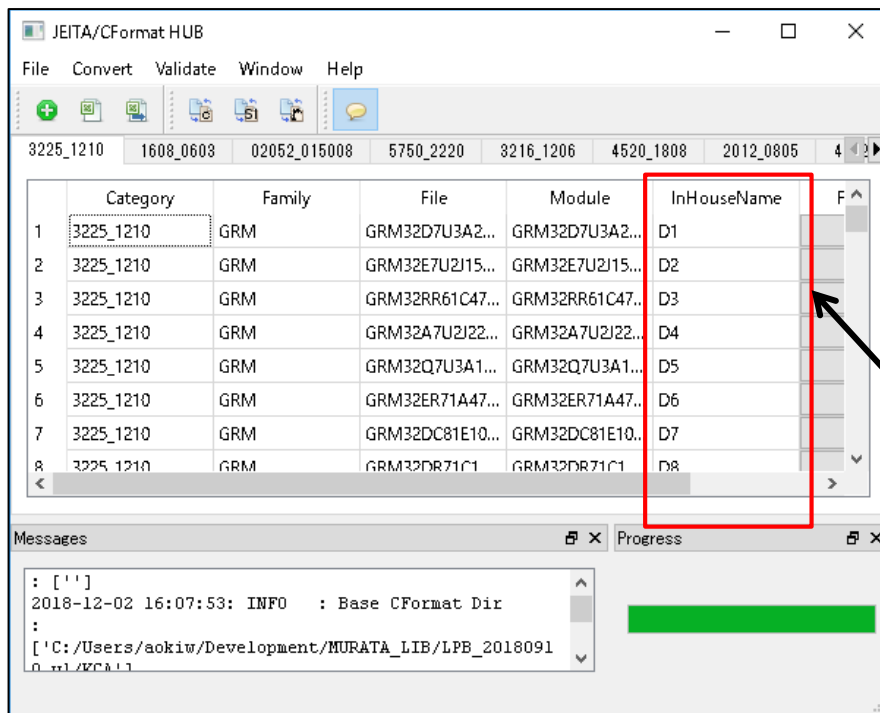
ベンダー提供の部品名を社内管理コードに変更する場合は、
#InHouseName列に入力してください。変更しない場合は、編集不要です。
不要なCategory(部品サイズ)はシート自体を削除してください。

#Category	#Family	#File	#Module	#InHouseName	#FootPrint
5750_2220	GRM	GRM55DR72-J224KW01.xml	GRM55DR72-J224KW01		
5750_2220	GRM	GRM55ER72A475KA01.xml	GRM55ER72A475KA01		
5750_2220	GRM	GRM55RB11H105KA01.xml	GRM55RB11H105KA01		
5750_2220	GRM	GRM55DR72H334KW10.xml	GRM55DR72H334KW10		
5750_2220	GRM	GRM55DR73A683KW01.xml	GRM55DR73A683KW01		
5750_2220	GRM	GRM55D7U2H473JW31.xml	GRM55D7U2H473JW31		
5750_2220	GRM	GRM55DR72E334KW01.xml	GRM55DR72E334KW01		
5750_2220	GRM	GRM55RR11H105KA01.xml	GRM55RR11H105KA01		
5750_2220	GRM	GRM55EB11H475KA01.xml	GRM55EB11H475KA01		
5750_2220	GRM	GRM55DR72E105KW01.xml	GRM55DR72E105KW01		
5750_2220	GRM	GRM55D7U3A103JW31.xml	GRM55D7U3A103JW31		
5750_2220	GRM	GRM55DR72J154KW01.xml	GRM55DR72J154KW01		
5750_2220	GRM	GRM55DR61H106KA88.xml	GRM55DR61H106KA88		
5750_2220	GRM	GRM55ER61H475KA01.xml	GRM55ER61H475KA01		
5750_2220	GRM	GRM55RR71H105KA01.xml	GRM55RR71H105KA01		
5750_2220	GRM	GRM55DR11E106KA01.xml	GRM55DR11E106KA01		
5750_2220	GRM	GRM55D7U2J473JW31.xml	GRM55D7U2J473JW31		

#InHoustName列に
社内管理コードを記載
し保存してください。

社内管理設定ファイルを入力する

6. CHUBを再起動して保存した設定ファイルを入力します。
File -> Import Setup file 

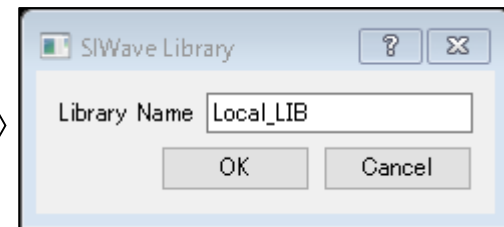
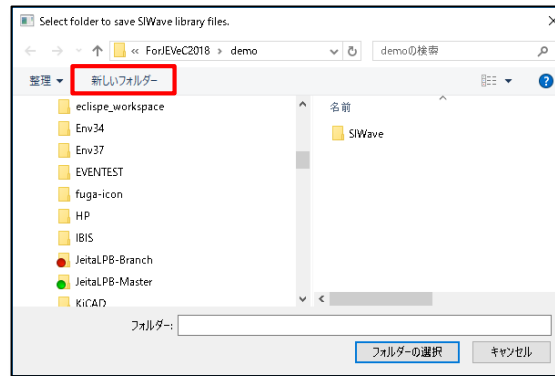
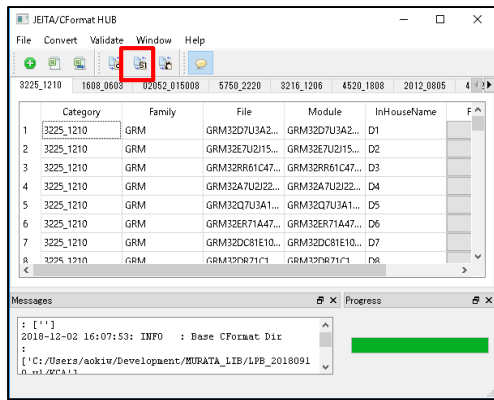


設定ファイル(Excelファイル)に入力した社内管理コードが、InHouseName列に表示されていることを確認してください

SIwaveライブラリに変換する

7. Convert -> SIwave Lib

SIwaveのライブラリを保存するフォルダを選択します。希望するフォルダがない場合は「新しいフォルダ」を押して新しいフォルダを作ります。



- Library Name : SIwave上で表示されるライブラリ名です。デフォルトは Local_LIBとなっています。

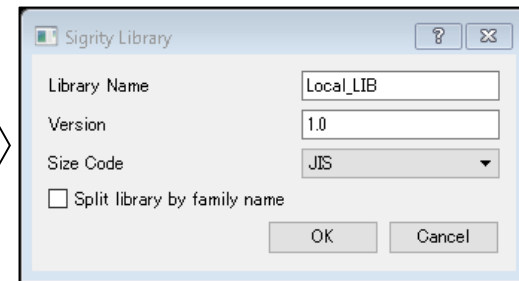
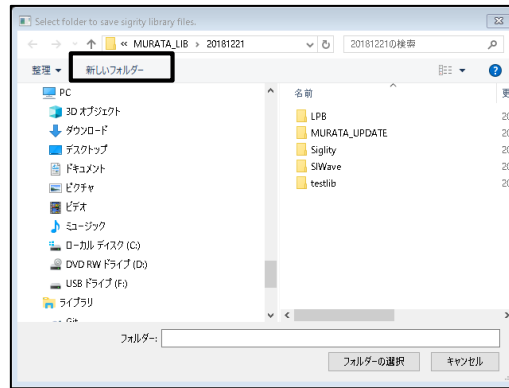
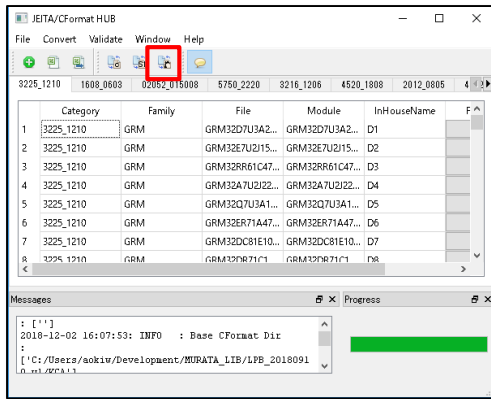
OKボタンで変換が始まります。

Sigrityライブラリに変換する

8. Convert -> Sigrity Lib



Sigrityのライブラリを保存するフォルダを選択します。希望するフォルダがない場合は「新しいフォルダ」を押して新しいフォルダを作ります。



- **Library Name** : Sigrity上で表示されるライブラリ名です。デフォルトはLocal_LIBとなっています。
- **Version** : ライブラリ管理用のバージョン番号です。任意のバージョン番号を入力してください。
- **Size Code** : Sigrityで管理されるサイズコードの種別です。JISとEIAの何れかを選択してください。
- **Split library by family name** : シリーズ毎にライブラリファイルを分解する場合はチェックしてください。デフォルトでは全てのシリーズを一つのライブラリで管理します

OKボタンで変換が始まります。

付録1

FOOTPRINTライブラリ

FootPrintライブラリとは

- 部品のFootPrintの形状だけをCFormatを使って記述したもの。

```
<?xml version="1.0" ?>
<JEITA_LPB_CFORMAT version="3.0">

  <module name="0603M_1" shape_id="1" type="PKG">
    <size_code metric="0603" imperial="0201" />
    <socket name="fp0603_1M">
      <default>
        <port_shape padstack_id="1"/>
      </default>
      <port id="p1" x="0.13" y="0"/>
      <port id="p2" x="-0.13" y="0"/>
    </socket>
  </module>

  <module name="0603M_2" shpae_id="1" type="PKG">
    <size_code metric="0603" imperial="0201" />
    <socket name="fp0603_2M">
      <default>
        <port_shape padstack_id="2"/>
      </default>
      <port id="p1" x="0.15" y="0"/>
      <port id="p2" x="-0.15" y="0"/>
    </socket>
  </module>

</JEITA_LPB_CFORMAT>
```

- CFormatの<global>/<shape>/<padstack>と<module>/<socket>/<port>を使って部品のフットプリントを表現
- 同じ部品コードのFootPrintは、同一ファイルに記載すること(異なるサイズコードのFootPrintを同一ファイルには記載できない)。
- <port>のid(左例の赤文字)は、部品ベンダから入手したCFormat(以後、Original CFormat Filesと呼称)のidとあわせておくこと。

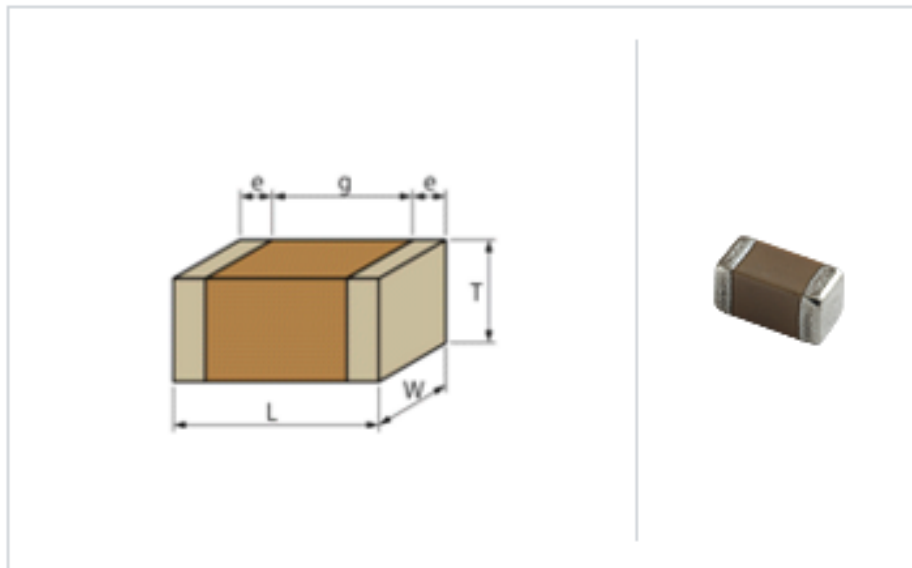
付録2

CFORMAT詳細

最も簡単なCFormatを見て見ましょう

村田製作所のチップコンデンサのCFormatを例にして説明します
GRM21BB30J226ME38

形状



L寸法	2.0 ±0.15mm
W寸法	1.25 ±0.15mm
T寸法	1.25 ±0.15mm
外部電極寸法e	0.2~0.7mm
外部電極間距離g	0.7mm min.
サイズコード mm	2012M

GRM21BB30J226ME38

```
<?xml version="1.0" ?>
<LPB_CFORMAT version="2020">
<header company="MURATA" date="Wednesday Dec. 19 2018"
  design_revision="1.0" project="GRM"/>
<global>
  <unit>
    <distance unit="mm"/>
    <time unit="ps"/>
    <angle unit="degree"/>
    <capacitance unit="uF"/>
    <resistance unit="ohm"/>
    <inductance unit="H"/>
  </unit>
  <shape>
    <rectangle height="1.25" id="1" width="2"/>
    <rectangle height="1.25" id="2" width="0.45"/>
  </shape>
  <padstack_def>
    <padstack id="1">
      <ref_shape shape_id="2" x="0" y="0" pad_layer="BOTTOM"/>
    </padstack>
  </padstack_def>
</global>
<module name="GRM21BB30J226ME38" shape_id="1" thickness="0"
  type="C" x="0" y="0">
  <size_code imperial="0805" metric="2012"/>
  <socket name="socket">
    <default>
      <port_shape padstack_id="1"/>
    </default>
    <port id="1" x="-0.775" y="0.0"/>
    <port id="2" x="0.775" y="0.0"/>
  </socket>
<specification>
  <capacitance typ="22"/>
</specification>
```

```
<reference xmlns:spice="http://www.jeita.or.jp/LPB/spice"
  reffile = "GRM21BB30J226ME38.mod"
  format="SPICE" >
  <connection socket_name="socket" port_id="1">
    <spice:ref_port subckt="GRM21BB30J226ME38" portid="1"/>
  </connection>
  <connection socket_name="socket" port_id="2">
    <spice:ref_port subckt="GRM21BB30J226ME38" portid="2"/>
  </connection>
</reference>
<reference
  xmlns:touchstone="http://www.jeita.or.jp/LPB/touchstone"
  reffile = "GRM21BB30J226ME38.s2p"
  format="TOUCHSTONE" >
  <connection socket_name="socket" port_id="1">
    <touchstone:ref_port portid="1"/>
  </connection>
  <connection socket_name="socket" port_id="2">
    <touchstone:ref_port portid="2"/>
  </connection>
</reference>
</module>
</LPB_CFORMAT>
```

LPB Formatの前に、XMLを少し...

C/N/R/M FormatはXMLで記述されています

```
<element_name attribute="xx" attribute="xx">
```

```
  <element_name> attributeは、有っても無くても良い
```

```
    :
```

```
  </element_name>
```

```
  <element_name attribute="xx" />
```

子供の階層が無い場合

```
</element_name>
```

ネットで検索するとだいたい分かります。

全体の構成

<?xml version="1.0" ?> ← これは、おまじない

```
<JEITA_LPB_CFORMAT version="2.2">
```

```
<header company="MURATA" date="Wednesday Dec. 19 2018"  
  design_revision="1.0" project="GRM"/>
```

```
<global>  
  :  
</global>
```

```
<module name="GRM21BB30J226ME38"  
  shape_id="1" thickness="0" type="C" x="0" y="0">  
  :  
</model>
```

```
</JEITA_LPB_CFORMAT>
```

<global> section

```
<global>
```

```
<unit>  
  <distance unit="mm"/>  
  <time unit="ps"/>  
  <angle unit="degree"/>  
  <capacitance unit="uF"/>  
  <resistance unit="ohm"/>  
  <inductance unit="H"/>  
</unit>
```

このCFormatファイルの中での単位系

```
<shape>  
  <rectangle height="1.25" id="1" width="2"/>  
  <rectangle height="1.25" id="2" width="0.45"/>  
</shape>
```

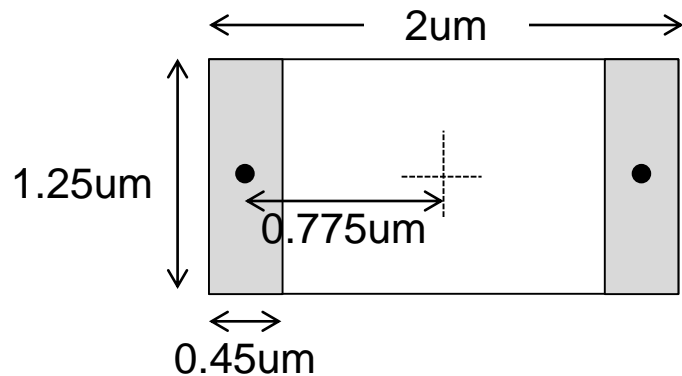
図形、この例では矩形を定義している

```
<padstack_def>  
  <padstack id="1">  
    <ref_shape shape_id="2" x="0" y="0" pad_layer="BOTTOM"/>  
  </padstack>  
</padstack_def>
```

PAD STACK、端子やVIAの形状、図形(shape)と層(layer)の組み合わせ

```
</global>
```

CFormat/部品外形、FootPrint



部品外形

```
<rectangle id="1"
  height="1.25" width="2"/>
```

```
<module name="GRM21BB30J226ME38"
```

```
  shape id="1" thickness="0" type="C" x="0" v="0">
```

```
<size_code imperial="0805" metric="2012"/> 部品のサイズコード
```

```
<socket name="socket">
```

```
  <default>
```

```
    <port_shape padstack_id="1"/>
```

```
  </default>
```

```
  <port id="1" x="-0.775" y="0.0"/>
```

```
  <port id="2" x="0.775" y="0.0"/>
```

```
</socket>
```

端子の形(PADSTACK)

```
<rectangle id="2"
  height="1.25" width="0.45"/>
<padstack id="1">
  <ref_shape shape_id="2"
    x="0"y="0"
    pad_layer="BOTTOM"/>
</padstack>
```

<socket> ... 端子の集合

CFormat/ 容量値の定義

```
<module name="GRM21BB30J226ME38"  
  shape_id="1" thickness="0" type="C" x="0" y="0">  
  <socket>  
    :  
  </socket>  
  <specification>  
    <capacitance typ="12.5"/>  
  </specification>
```

カタログ値の定義

CFormat/モデルのラップ

```
<module name="GRM21BB30J226ME38" ..... >
```

```
:
```

```
<reference  
  xmlns:spice="http://www.jeita.or.jp/LPB/spice"  
  format="SPICE"  
  reffile = "GRM21BB30J226ME38.mod">  
  :  
</reference>
```

SPICE

SPICEファイル名

```
<reference  
  xmlns:touchstone="http://www.jeita.or.jp/LPB/touchstone"  
  format="TOUCHSTONE"  
  reffile = "GRM21BB30J226ME38.s2p">  
  :  
</reference>
```

S-Para

S-Paraファイル名

```
</module>
```

CFormat/SPICEモデル

```
<module name="GRM21BB30J226ME38" ... >
<socket name="socket">
  :
  <port id="1" x="-0.775" y="0.0"/>
  <port id="2" x="0.775" y="0.0"/>
</socket>

<reference
  xmlns:spice="http://www.jeita.or.jp/LPB/spice"
  format="SPICE"
  reffile = "GRM21BB30J226ME38.mod" >
  <connection socket_name="socket" port_id="1">
    <spice:ref_port subckt="GRM21BB30J226ME38" portid="1"/>
  </connection>
  <connection socket_name="socket" port_id="2">
    <spice:ref_port subckt="GRM21BB30J226ME38" portid="2"/>
  </connection>
</reference>
```

subckt(GRM21BB3J226ME38)の2番目のノード(portid="2")
は、id="2"のport(port_id="2")に対応する

CFormat/S-Paraモデル

```
<module name="GRM21BB30J226ME38" ... >
<socket name="socket">
  :
  <port id="1" x="-0.775" y="0.0"/>
  <port id="2" x="0.775" y="0.0"/>
</socket>

<reference
  xmlns:touchstone="http://www.jeita.or.jp/LPB/touchstone"
  reffile = "GRM21BB30J226ME38.s2p"
  format="TOUCHSTONE" >
  <connection socket_name="socket" port_id="1">
    <touchstone:ref_port portid="1"/>
  </connection>
  <connection socket_name="socket" port_id="2">
    <touchstone:ref_port portid="2"/>
  </connection>
</reference>
```

S-Paraの2番目のノード(portid="2")は、
id="2"のport(port_id="2")に対応する

付録3

LPB DESIGN KIT

デザインキットの目的

1. 教育

LPBフォーマットを作ってみる

LPBフォーマットを扱うプログラムの例

2. 発想の種

LPBフォーマットを使った応用例

デザインキットの内容

デザインキットはpythonスクリプトで作った簡易ツールです

【配布形態】

- すぐに使える実行形式ファイル
 - とにかく使ってみたい
- Pythonスクリプト（pythonの実行環境が必要）
 - 自分でも何か作ってみたい
 - 参考になるコードが欲しい

デザインキットの配布形態

- ライセンス

GNU Library or Lesser General Public License (LGPL)

- デザインキットは自由に利用可能です
- デザインキットの利用により生じた不利益や損害などに対して、一切の責任を負いません

デザインキットを改造したい

- ダウンロードは、こちらから

<http://www.lpb-forum.com/lpbdesignkit/>

- デザインキットのコードは gitlab で公開・管理しています

<https://gitlab.com/aoki-lpb-forum/jeitaLPB>

