

プリント基板開発を体験する

—2005年1月号付属FPGA基板をリバース・エンジニアリング

城野幸男

ボード設計に携わる新人技術者に割り当てられるしごとの一つとして、過去に先輩技術者が開発したボードの設計情報を利用して、派生製品のためのボードを作るといった作業がある。ここではこうしたリバース・エンジニアリングの事例を紹介する。本誌2005年1月号に付属したFPGA基板のデータを復元したり、変更したりする。なお、本稿で作成した設計データは、すべて本誌付属のCD-ROMに収録されている。(編集部)

ここではプリント基板の開発事例として、リバース・エンジニアリングの実例を紹介します。すでにでき上がっている基板の製造データを使って、元の設計データを復元します。そして、これを再利用するための方法について説明します。合わせて、リバース・エンジニアリングの意味についても考えてみます。サンプルとして、本誌2005年1月号に付属していたFPGA基板を取り上げました。リバース・エンジニアリングの手順を理解していただくと同時に、このFPGA基板の応用の一つとして役立てていただきたいと思います。

1 リバース・エンジニアリングとは

普通のもの作りでは、設計情報を加工手段に置き換え、それを用いて製造するという手順でしごとを進めます。例えばプラスチック成型品を製造する場合、その製品の設計図面から金型を起こし、この金型を使ってプラスチックの成型を行います。

一方、この逆を行うのが、「リバース・エンジニアリング」です。製品や加工手段から設計情報を抽出します。すなわち、でき上がったプラスチック製品を採寸して、その金型を設計したり、製品や金型の採寸から製品の設計図を作ったりすることが、リバース・エンジニアリングの手順です(図1)。

● 模倣 vs. リバース・エンジニアリング

リバース・エンジニアリングは、解析によってその技術を吸収すること、そして独自性を加えてその技術を再利用することを目的として行うものです。例えば、競合する他

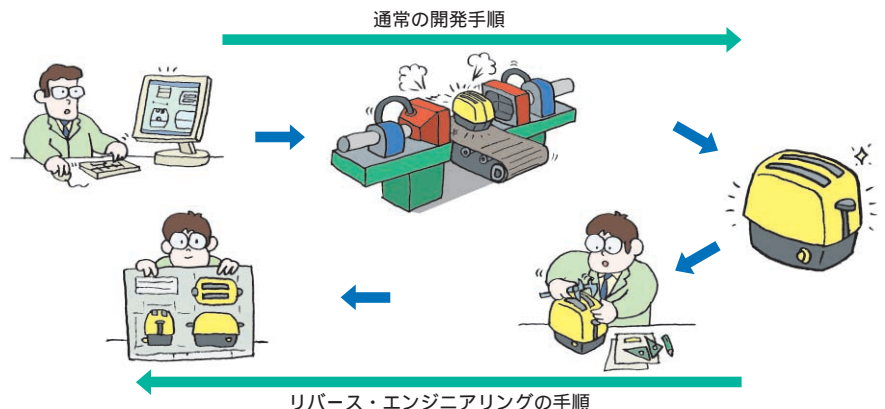
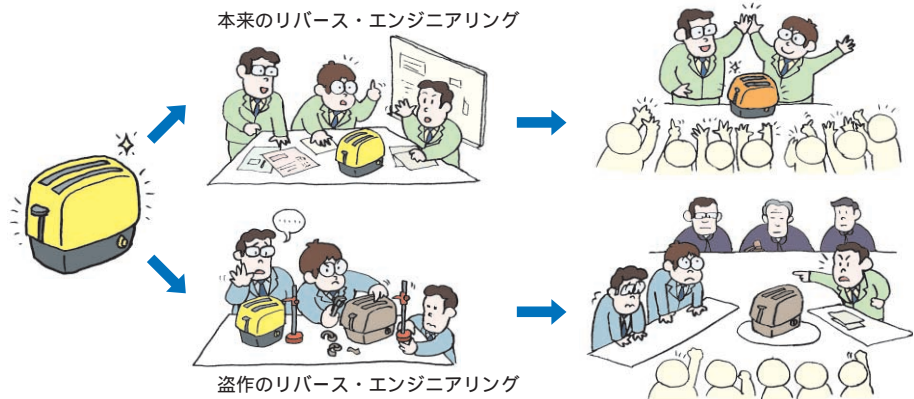


図1
リバース・エンジニアリングは工程を逆行する

リバース・エンジニアリングでは、普通のもの作りと逆の工程になる。例えば、でき上がったプラスチック製品を採寸してその金型を設計したり、製品や金型の採寸から製品の設計図を作ったりする。

図2
リバース・エンジニアリングでは、新たな価値を創造することが重要

単なる模倣だけを目的とした商品の解体調査は、正当なリバース・エンジニアリングとは言えない。リバース・エンジニアリングが妥当な行為として認知され、盛んに行われるようになったのは、その成果が社会に還元されるからである。



社の新製品を模倣すれば、手っ取り早く対抗商品を作ることができます。しかし、単なる模倣だけを目的とした商品の解体調査は、正当なリバース・エンジニアリングではなく、盗作の技術です。リバース・エンジニアリングが妥当な行為として認知され、盛んに行われるようになったのは、その成果が社会に還元されるからです。すなわち、蓄積された知的資産は利己的に再利用するだけではなく、新たな価値の創造や課題の解決によって社会に貢献しなくてはなりません。

余談になりますが、筆者が以前、OEM(original equipment manufacturing)受託生産のための設計を担当していたときの話です。OEM 提携先へのデモンストレーションのために、入手したサンプルを解体調査してデッド・コピーしたことがあります。回路図も描かず現物だけを見てまったく同じものを作りました。そのときこれを見た同僚が、「それは“技術”ではなく“奇術”だね」と言ったのを覚えています。今思うとまさにそのとおりです。もしこのときにリバース・エンジニアリングの意味を理解していれば、会社の業績が伸び、筆者も回路設計業務を続けていたかもしれません(図2)。

● さまざまな目的で行われるリバース・エンジニアリング

リバース・エンジニアリングの目的とその例を挙げてみます。

1) 開発の効率化

リバース・エンジニアリングは、他社の製品を解体調査することにより、競争力のある商品を効率良く開発するという目的で行われます。半導体の業界などで活発に行われ、リバース・エンジニアリングを専門に請け負う企業もあるようです。

2) 互換製品の開発

リバース・エンジニアリングは、互換性を実現するための解析に用いられます。話は少し古くなりますが、セイコーエプソンが NEC のパソコン「PC-9800 シリーズ」をリバース・エンジニアリングして互換製品を開発した例はこの典型です。現在の Windows パソコンのルーツである IBM-XT/AT でも多くの互換機が開発されました。しかし、この場合、BIOS(basic input/output system)のソース・コードを含むシステム全体の仕様が公開されていたので、包括的なリバース・エンジニアリングは必要なかったと思います。互換製品を開発するためのリバース・エンジニアリングは、ハードウェアとソフトウェアの双方において行われています。

3) システムやソフトウェアの保守

ソフトウェアを含むシステムでは、プログラムのソース・コードがないとトラブルの原因究明が困難です。これが手に入らない場合、逆コンパイルによるソースの抽出が必要になります。とくに、ソフトウェアがハードウェアを直接制御している箇所については、リバース・エンジニアリングは不可欠な手段です。筆者も以前、ROM からバイナリをダンプして、逆アセンブルした記憶があります。外部に発注して開発したカスタム・ソフトウェアの保守や、パッケージとして流通しているソフトウェアのセキュリティ対策を行う場合などにもリバース・エンジニアリングが用いられます。

4) 既存資産の再利用

図面やオリジナル・データが存在しない古い資産を再利用する場合にも、リバース・エンジニアリングの手法が用いられます。CAD(computer aided design)データのないプリント基板(PCB: print circuit board)を設計変更した