# 「インデザイン」方式のフィジカル検証：DRC 自動修正

最先端の製造プロセスでは、これまで以上に多くの DRC（デザインルール・チェック ) を実行する必要があります。
インデザイン・フィジカル検証に含まれる DRC 自動修正 (ADR=Automatic DRC Repair) 機能を利用して、
サインオフ DRC 違反の検出から修正、再バリデーションまでを自動化する方法についてご説明します。

これまで、設計部門と製造部門の間では「インプリメント後に検証」というフローが一般的でした。配置配線後のインプリメンテーション環境でデザインが「クリーン」であっても、製造前のサインオフを達成するには、別のツールを使ってもう一度デザインを物理的に検証しなければなりません。サインオフ DRC では、配線テクノロジでは見つからないコーナーケースの問題が指摘されます。設計チームはこれらを修正してもう一度バリデーションを実行することになります（図 1）。
設計チームはサインオフ・レベルの違反をすべて特定したら、エラー・データベースをインプリメンテーション環境に戻して違反のあるジオメトリを手作業で修正します。また、場合によっては配線テクノロジ・ファイルを変更してデザイン全体の配線をよりクリーンなものにしていきます。

次に、設計チームはインプリメンテーション工程全体をやり直してタイミング収束を達成し、別の検証ツールを使ってフィジカル・サインオフを繰り返すことになります。以上の工程を、DRC 違反がなくなるまで繰り返しますが、これがテープアウトの遅延を招く原因となっています。

## 不確実性のリスクをなくす

これに対し、インデザイン方式のフィジカル検証では、DRC 違反をなくすまでのイタレーション回数を劇的に削減できます。このアプローチでは、フィジカル・インプリメンテーション・ツールとフィジカル検証ツールを

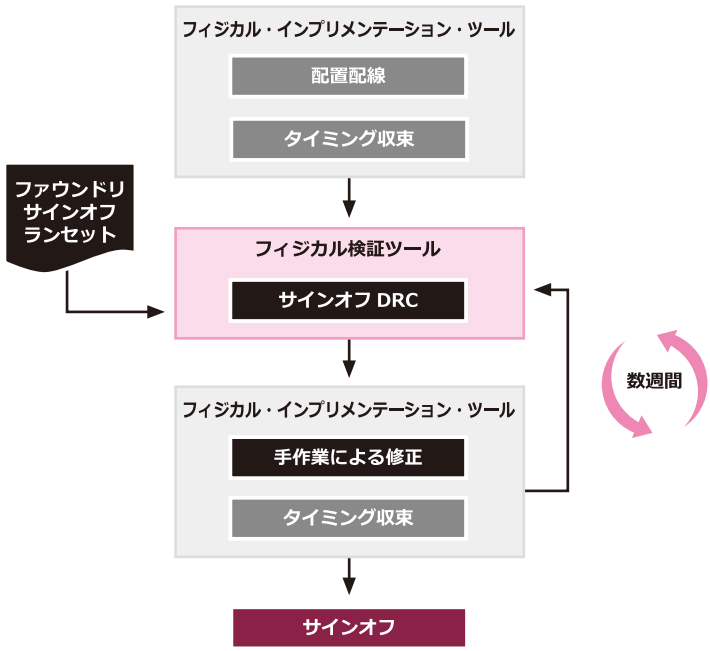


図 1.「インプリメント後に検証」という伝統的なアプローチによるフィジカル設計

連携させ、インプリメンテーションの段階でサインオフ・レベルの DRC 解析を実行します。このように早い段階でDRC違反を特定することにより、DRC 違反の修正を最小限のコストで効率よく行えるようになります。

## DRC 自動修正 (ADR) フロー

インデザイン・フィジカル検証を利用すればデザインに潜む DRC 違反の検出にかかる時間を短縮できますが、違反の修正には相変わらず多くの時間とノウハウが要求されます。

そこで、シノプシスはインデザイン方式のフィジカル検証に自動修正機能（図２）を追加し、生産性のさらなる向上を図っています。この自動修正機能により、検出したサインオフ DRC 違反のほとんどがツールによって修正されるため、手作業での修正は最小限に抑えられます。
ADR は、シノプシスの IC Compiler と IC Validator のインテリジェントな連携が基盤となっています。このフローでは、設計者は使い慣れた環境で DRC 結果を確認し、その時点でどのルールを ADR の対象とするかを決めることができます。

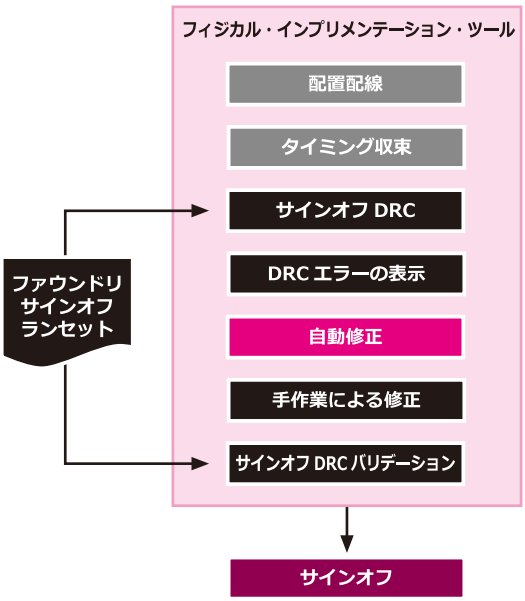


図 2. ADR 機能を追加したインデザイン・フィジカル検証フロー

ADR は、発見的手法に基づく修正を実行することにより、レイアウトの変更を最小限に抑えつつ高い成功率を達成できるよう最適なバランスをとっています。発見的手法に基づく修正には、特定アルゴリズムによる決定論的な解決手法よりも問題解決範囲が広く、収束までの時間が短く、結果がチップのタイミングに影響を与えにくいなどの利点があります。

設計チームは対象となるデザインルールに関して深い知識がなくても ADR フローをセットアップできるほか、修正 / バリデーションの実行レベルを指定することにより、フローの実行時間と修正率を自由に調整できます。対象となるジオメトリに対する ADR 処理は極めて高速で、毎秒約 3 ヵ所の修正を完了できます。

全体的に、ADR フローでは手作業で修正とバリデーションを行った場合に比べ、はるかに高いパフォーマンスと生産性が得られます。

## ケーススタディ

約 80 万インスタンスの 45nm デザインでサインオフ・レベルの DRC 違反 550 ヵ所に対して ADR を実行したところ、自動修正のイタレーションを 2 回繰り返して 432 ヵ所 (79%) の違反を修正できたという事例があります。
ADR は M1 ～ M3 レイヤ、およびビア・レイヤの間隔と幅の違反に対して非常に良好な性能を発揮しました。特に、ADR は下位レベルのライブラリ・セルに含まれるポリゴンと配線レベルの信号線の間で発生する 392 カ所の M1 配線間隔違反を修正しました。

このような高い修正率にもかかわらず、ADR がデザインに対して行った修正は非常に局所的であったため、タイミングへの影響はほとんどありませんでした。デザイン内で最もタイミングが厳しい 15,000 のクリティカルパスに関しても、その 99% でスラックがほとんど変化していません (+/-1 ps)。

ピンやパワー・ストラップなどの「移動不可」として指定したオブジェクトに関する違反、あるいは指定したレベルよりも踏み込んだ修正が必要な違反については、ADR による修正は行われませんでした。ADR は既存のレイアウトをなるべく保全することを最優先しているためです。

このデザインでは、タイミングにほとんど影響を与えることなく 80% の修正率を達成し、複数のメタル層とビア層に対する修正に成功しました。これまでなら 1 ～ 2 日の手作業が必要であったと考えられるこのようなケースでも、ユーザーからの入力は一切必要とせず、標準的な 4 CPU のハードウェア構成で 2 時間以内に完了しました。

## まとめ

最先端の半導体プロセスではデザインルールの数が増え、複雑化も進んでいるため、インプリメント終了後に従来のアプローチによるフィジカル検証を行うという方法が通用しなくなりつつあります。もはや、サインオフ・レベルの検証のために数日～数週間もの時間をかけて手作業による修正を行っている余裕はありません。

シノプシスは、IC Compiler と IC Validator を使用したインデザイン方式のフィジカル検証に、ADR 機能を追加しました。このフローでは、サインオフ・レベルの DRC 違反を自動で修正できます。サインオフ DRC 違反のほとんどは ADR を使って短時間で修正できるため、設計工程終盤での手作業による修正の必要を最小限に抑えられ、デザイン全体のイタレーションは不要になります。最後に ADR フローの利点をまとめます。

- サインオフ品質の独立した DRC 解析
- フィジカル構造 / タイミングへの影響を最小限に抑えつつ、配線ドリブンの高い修正率を実現
- 修正後のレイアウト領域に対してインクリメンタルなバリデーションを自動で実行
- 対象とするルールや修正ループの実行回数を柔軟に指定可能
- 簡単に使えるプッシュボタン式のインターフェイス