

## CODE COVERAGE

### コードカバレッジとは

コードカバレッジはもともとソフトウェアのプログラム開発において広く使われて来たテストの評価手法で、テスト対象となるプログラム全体に対して、どれくらいテストが実行されたかを示す割合を指します。RTLコードで記述されるハードウェア設計に対しても、このコンセプトの適用が進んでおり、2016年にワールドワイドで行われたWilson Research Groupの市場調査によれば、FPGAプロジェクトの64%においてコードカバレッジが採用されています。また航空宇宙など安全が重要視される産業界によっては、要件に対するカバレッジレポートを出力し、適切なテストが行われたことを示すドキュメントの提出が求められます。

コードカバレッジには、その測定方法により様々な種類に分類されますが、代表的なものはステートメントカバレッジと呼ばれるものです。ステートメントカバレッジはRTLコードにおける実行可能なステートメント構文に対して活性化された割合を示します。活性化されていればテストされたことになりカバレッジが上がります。活性化されていない場合、それはテストに抜けがあり追加テストが必要か、またはどのようなテストを追加しても活性化することができない到達不可能なコードか、あるいは再利用を繰り返すなかで使われなくなったモジュール、などの原因が考えられます。いずれの場合でも、測定されたカバレッジを解析することで、次のアクションへの指針となります。

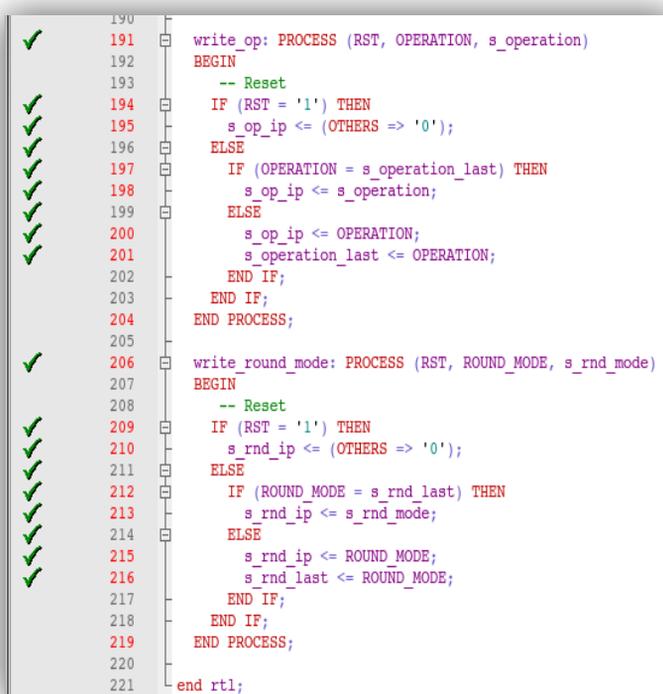
### コードカバレッジのメリット

コードカバレッジはシミュレーション実行時にスイッチを指定するだけで測定を開始することができます。その他のコード記述は一切必要ありません。活性化率を測定することで、テストの抜けや漏れを特定でき、テスト品質を定量化することができます。また納品先に、設計コード、テストコードに加え、カバレッジレポートを入れることで、テストプロセスへの高い信頼を得ることができます。

### コードカバレッジの種類

コードカバレッジには、以下に示すような様々な指標があり、多角的なテスト評価が可能です。

- ステートメントカバレッジ - 各ステートメントの実行回数を測定します。1行に複数ステートメントがあっても、それぞれ測定します
- ブランチカバレッジ - if/then/elseやcaseの条件による各分岐の実行回数を測定します。実行されなかった分岐条件の真偽も分かります
- コンディションカバレッジ - if文や3項演算子などにおける意思決定の分析し、ブランチカバレッジ機能を拡張します
- エクスプレッションカバレッジ - コンディションカバレッジに類似しますが、代入文の右辺にあるエクスプレッションを解析します
- FSMカバレッジ - 有限ステートマシンにおけるステート、遷移、パスの実行回数を測定します
- トグルカバレッジ - 各論理ノードがあるステートから他のステートへと遷移する回数を測定します
- SystemVerilogクラスカバレッジ - エラボーレートされたクラスタイプおよびパラメータ化されたクラスのスペシャライゼーションに関するデータを収集します



```
190
191 write_op: PROCESS (RST, OPERATION, s_operation)
192 BEGIN
193     -- Reset
194     IF (RST = '1') THEN
195         s_op_ip <= (OTHERS => '0');
196     ELSE
197         IF (OPERATION = s_operation_last) THEN
198             s_op_ip <= s_operation;
199         ELSE
200             s_op_ip <= OPERATION;
201             s_operation_last <= OPERATION;
202         END IF;
203     END IF;
204 END PROCESS;
205
206 write_round_mode: PROCESS (RST, ROUND_MODE, s_rnd_mode)
207 BEGIN
208     -- Reset
209     IF (RST = '1') THEN
210         s_rnd_ip <= (OTHERS => '0');
211     ELSE
212         IF (ROUND_MODE = s_rnd_last) THEN
213             s_rnd_ip <= s_rnd_mode;
214         ELSE
215             s_rnd_ip <= ROUND_MODE;
216             s_rnd_last <= ROUND_MODE;
217         END IF;
218     END IF;
219 END PROCESS;
220
221 end rtl;
```

ソースウィンドウ内のカバレッジ表示

