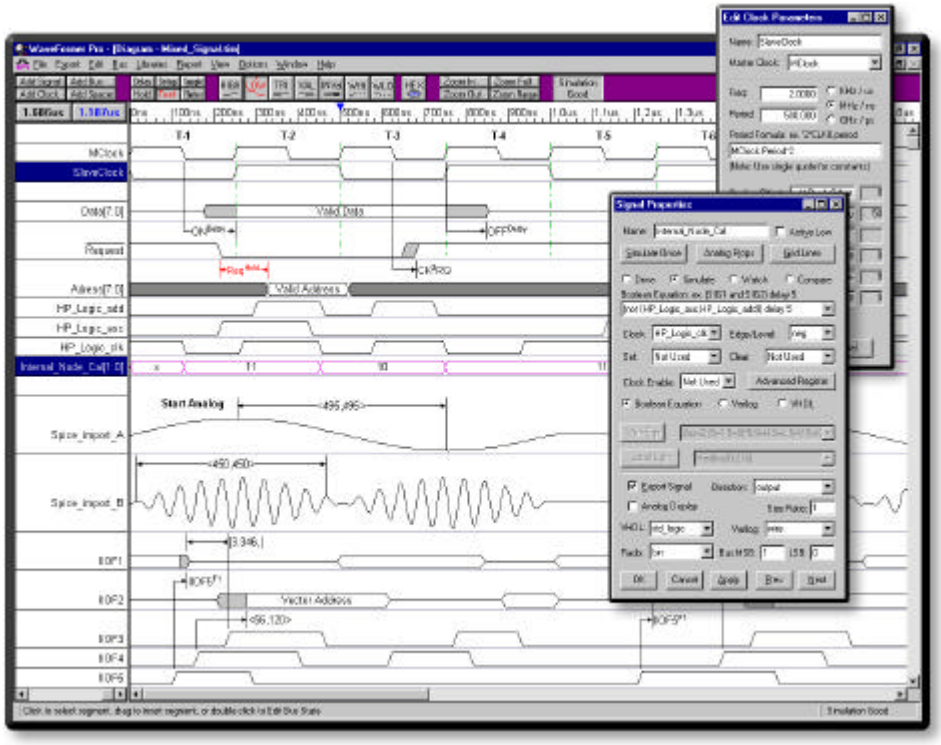
TestBencher Pro



タイミングダイヤグラム入力による、テストベンチ生成ツール

対応言語：VHDL、Verilog、OpenVera、e、
SystemC、及び、C++（TestBuilder）

TestBencher Pro

InterLink

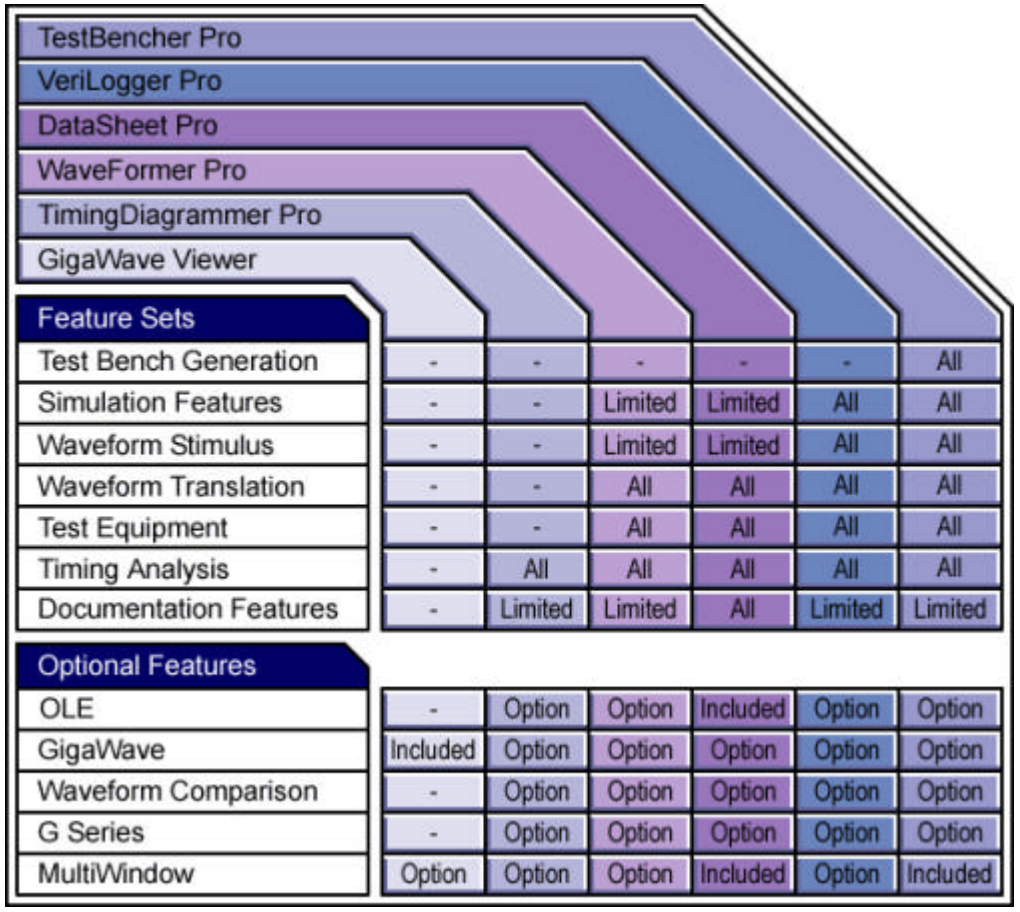
SYNAPTICAD

予定

- [] SynaptiCAD社製品群の中での、TestBencher Proの位置づけ
- [] タイミングダイヤグラム記述方法
- [] Setup、Holdパラメータによるタイミング解析例波形の例
- [] 波形記述例
- [] MS-Wordへの波形貼り付け
- [] TestBencher Pro 製品概要
- [] コード生成アーキテクチャ
- [] TestBencherデザインフロー
- [] 操作デモ－SRAMテスト－
- [] e テンポラル記述－出力例
- [] TestBencherデザインフロー：e
- [] TestBencherデザインフロー：OpenVera
- [] TestBencherデザインフロー：OpenVera、SystemC
- [] TestBencherデザインフロー：VHDL、Verilog、TestBuilder

SynaptiCAD社製品群の中での、TestBencher Proの位置づけ

TestBencher Proは
製品群の最上位製品
→（OLE機能を除く）
下位製品の全て
の機能を含む



Feature Sets						
Test Bench Generation	-	-	-	-	-	All
Simulation Features	-	-	Limited	Limited	All	All
Waveform Stimulus	-	-	Limited	Limited	All	All
Waveform Translation	-	-	All	All	All	All
Test Equipment	-	-	All	All	All	All
Timing Analysis	-	All	All	All	All	All
Documentation Features	-	Limited	Limited	All	Limited	Limited
Optional Features						
OLE	-	Option	Option	Included	Option	Option
GigaWave	Included	Option	Option	Option	Option	Option
Waveform Comparison	-	Option	Option	Option	Option	Option
G Series	-	Option	Option	Option	Option	Option
MultiWindow	Option	Option	Option	Included	Option	Included

TestBencher Pro

InterLink

SYNAPTICAD

SynaptiCAD社製品群の中での、TestBencher Proの位置づけ（続き）

TestBencher Pro：

- グラフィカルな入力によるテストベンチ生成ツールです。
- VeriLogger Proの機能を有します。
- 言語に依存しないタイミングダイヤグラムから、リアクティブ（自身で応答可能）なテストベンチやバスファンクショナルモデルを生成します。
- サポートしている言語：VHDL、Verilog、OpenVera、*e*、SystemC、およびC++（TestBuilder）

VeriLogger Pro：

- グラフィカルな入力によるテストベクタ生成機能付きの、Verilogシミュレータです。
- また、WaveFormer Proの機能を有します。
- 特徴として、タイミングダイヤグラムからテストベクタを生成する機能があり、テストパタンを用意しなくとも、タイミングダイヤグラムからの入力によりMUTをシミュレーションする事ができます（インターラクティブ・シミュレーション機能）。

SynaptiCAD社製品群の中での、TestBencher Proの位置づけ（続き）

DataSheet Pro：

- タイミングダイヤグラムを含むドキュメントの作成支援ツールです。
- WaveFormer Proの機能を有します。
- 主な特徴：
  - [] OLE機能は標準装備。
  - [] プロジェクト・ウィンドウにより、複数のタイミングダイヤグラムを一つのプロジェクトとして管理できます。

WaveFormer Pro：

- タイミングダイヤグラムエディタです。
- TimingDiagrammer Proの機能を有します。
- 主な特徴：
  - [] タイミングダイヤグラムから、HDLスティミュラスを生成します。
  - [] 多くのファイル形式に対して入出力を対応しています。
    - →ロジアナで吸い上げた実機データをシミュレータに利用
    - →シミュレータのデータ（vcdファイル）をパタンジェネレータで実機環境で利用
    - ・・・など

SynaptiCAD社製品群の中での、TestBencher Proの位置づけ（続き）

TimingDiagrammer Pro：

- タイミングダイヤグラムエディタです。

GigaWave Viewer：

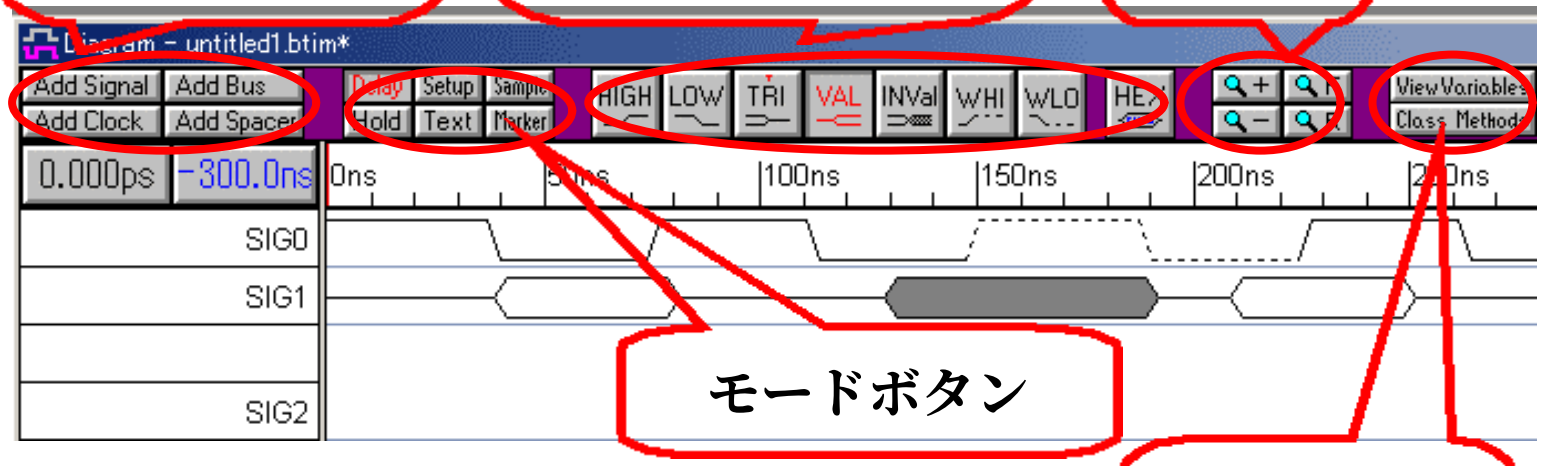
- 波形ビューワです。

タイミングダイヤグラム記述方法

新たな信号の追加ボタン

ステートの状況を表すボタン

ズームボタン



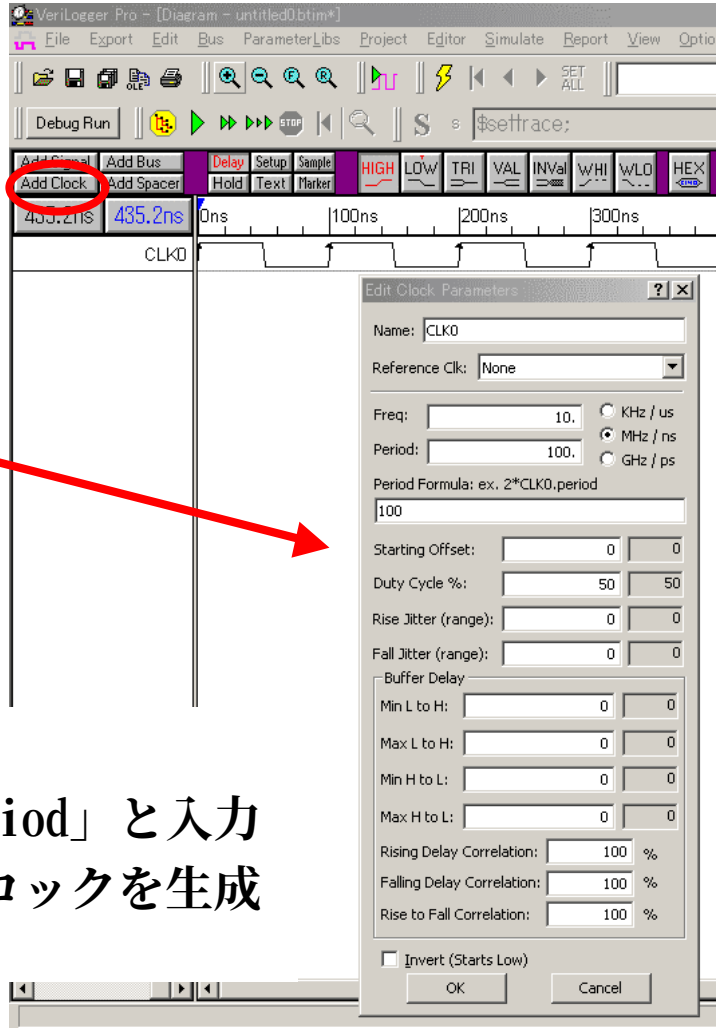
モードボタン

変数とクラス、メソッドボタン

ご参照：スケーリング操作
http://www.syncad.com/ani_app.htm
（アニメーション・アプリケーション）

タイミングダイヤグラム記述方法（続き）

クロック信号の定義：
Add Clockボタンを押し、
開くダイヤログの中で、
クロックのプロパティ
（周期等）を定義する



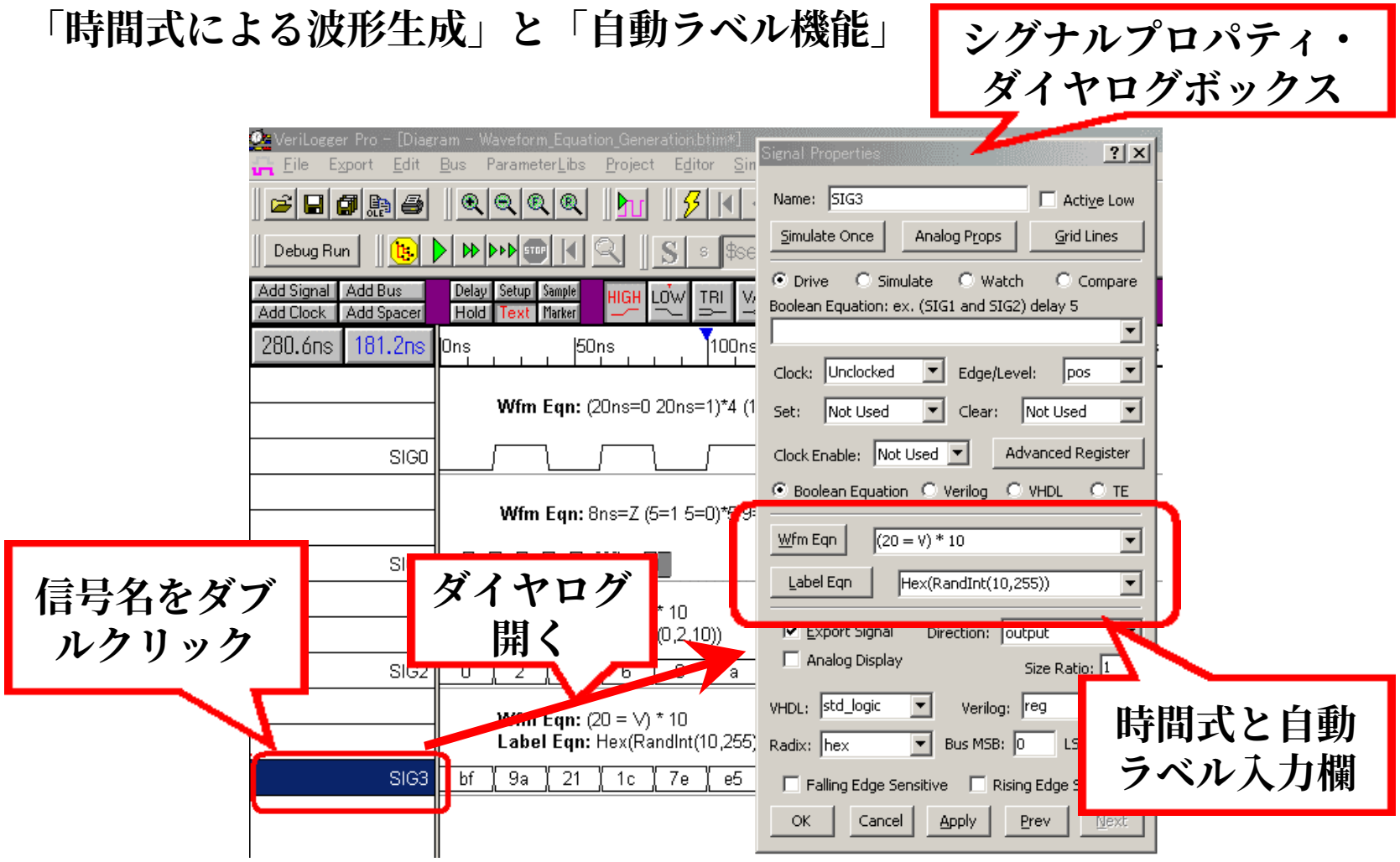
式による定義も可能

例）CLK1のPeriod Formula欄に「2*CLK0.period」と入力

→ 信号CLK0の２倍のクロック信号のクロックを生成する

タイミングダイヤグラム記述方法（続き）

「時間式による波形生成」と「自動ラベル機能」



TestBencher Pro

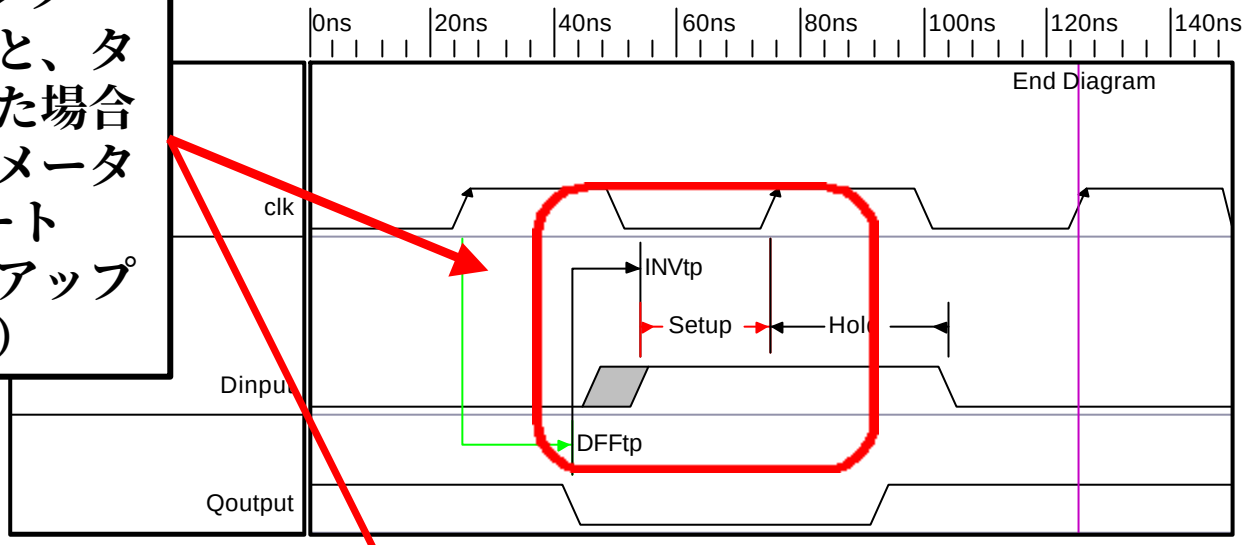
InterLink

SYNAPTICAD

タイミングダイヤグラム記述方法（続き）

―Setup、Holdパラメータによるタイミング解析例―

[Setup]、[Hold]タイミング・パラメータを使用すると、タイミング違反が発生した場合、ダイヤグラムとパラメータウィンドウでレポート（右図例では、セットアップエラーをレポート）



使用回路

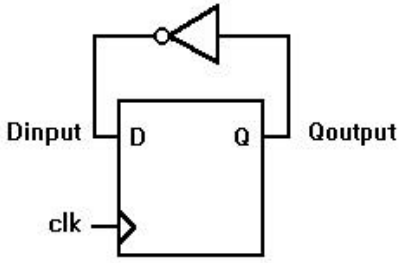
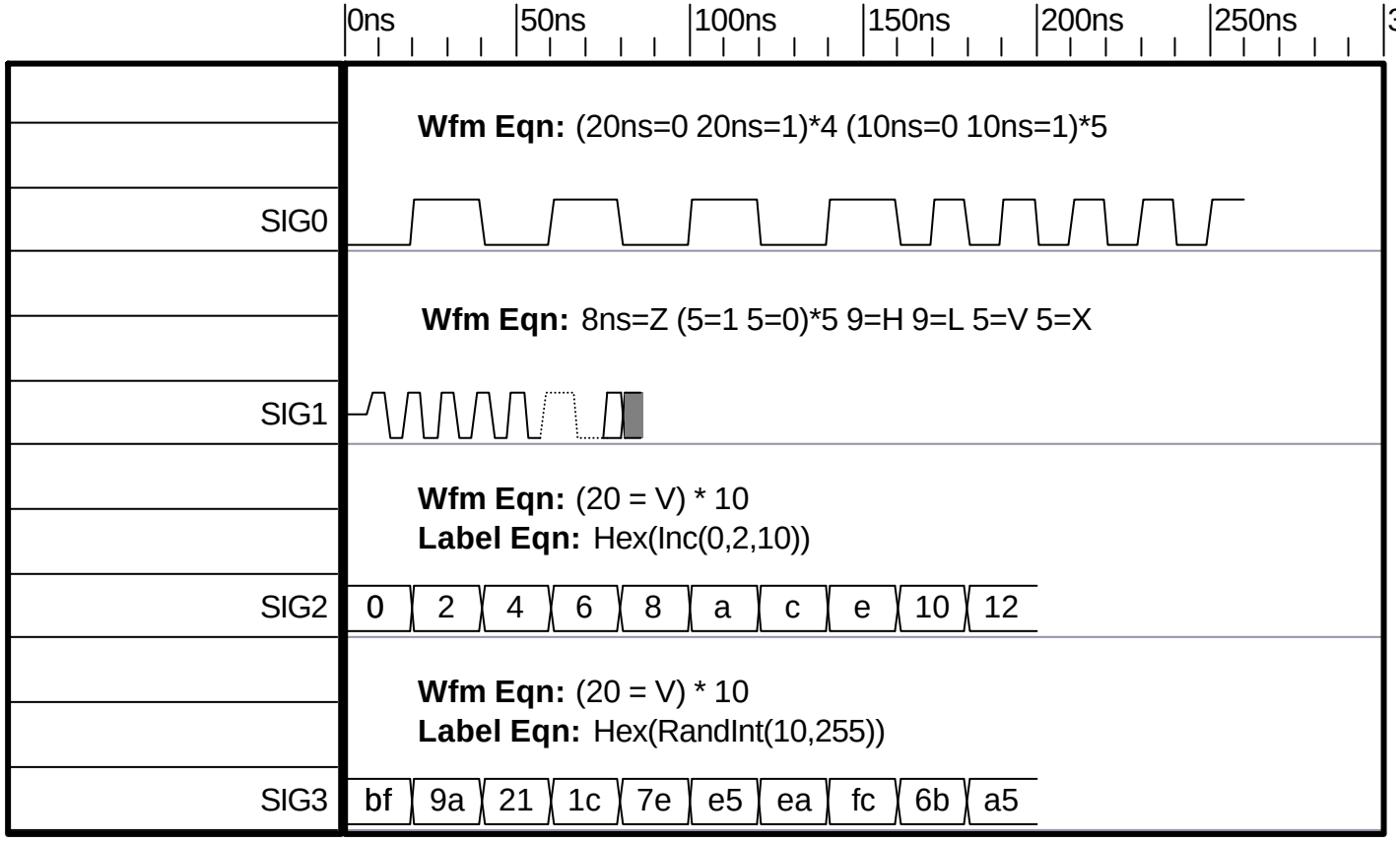


Figure 1: Tutorial circuit

name	min	max	margin	comment
DFFtp		18	na (delay)	clk to Qoutput propagation time
INVtp	3	11	na (delay)	INV inverter delay
Setup	25		-4	Check for metastable condition
Hold	1		28	

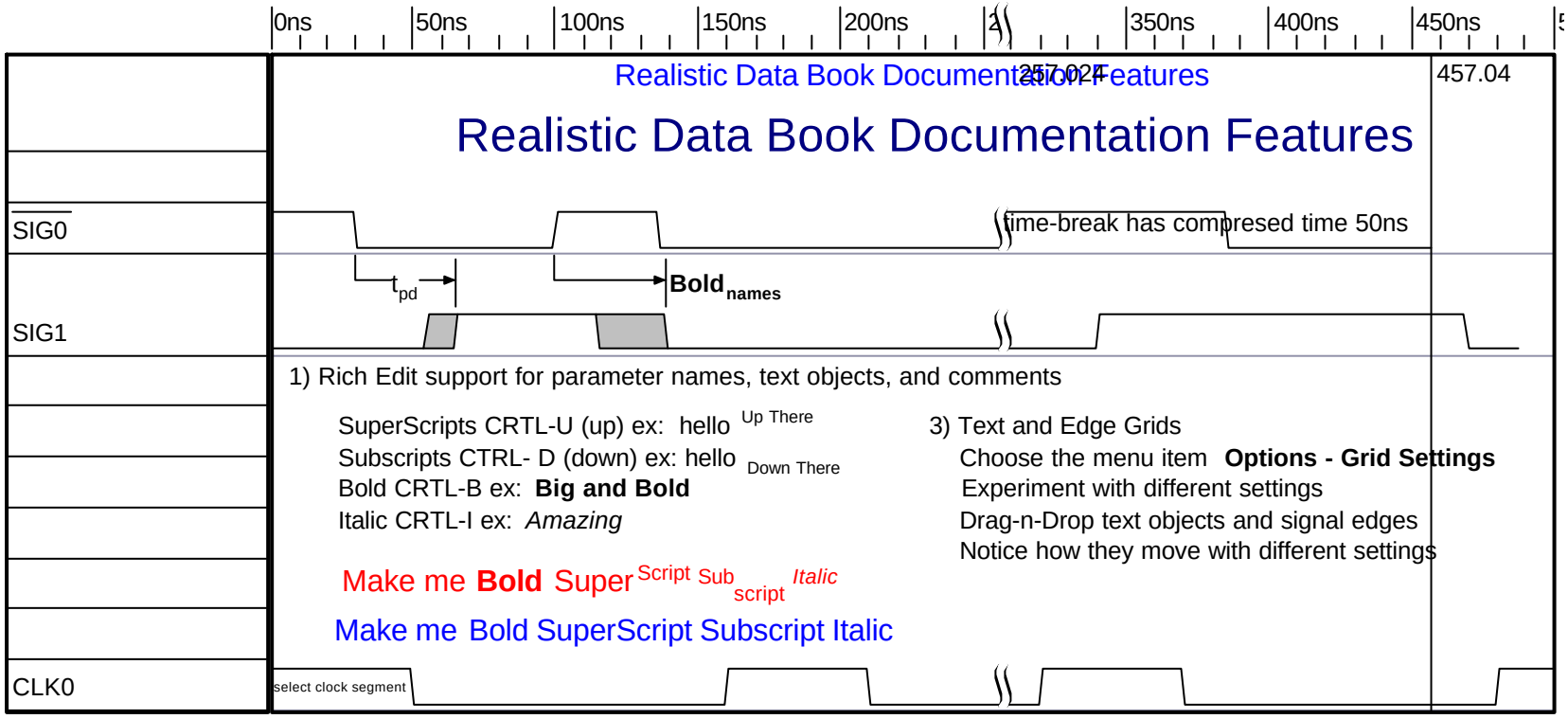
タイミングダイヤグラム記述方法（続き）

時間式による波形生成（[Wfm Eqn]ボタン）と
自動ラベル機能（[Label Eqn]ボタン）で生成した波形の例



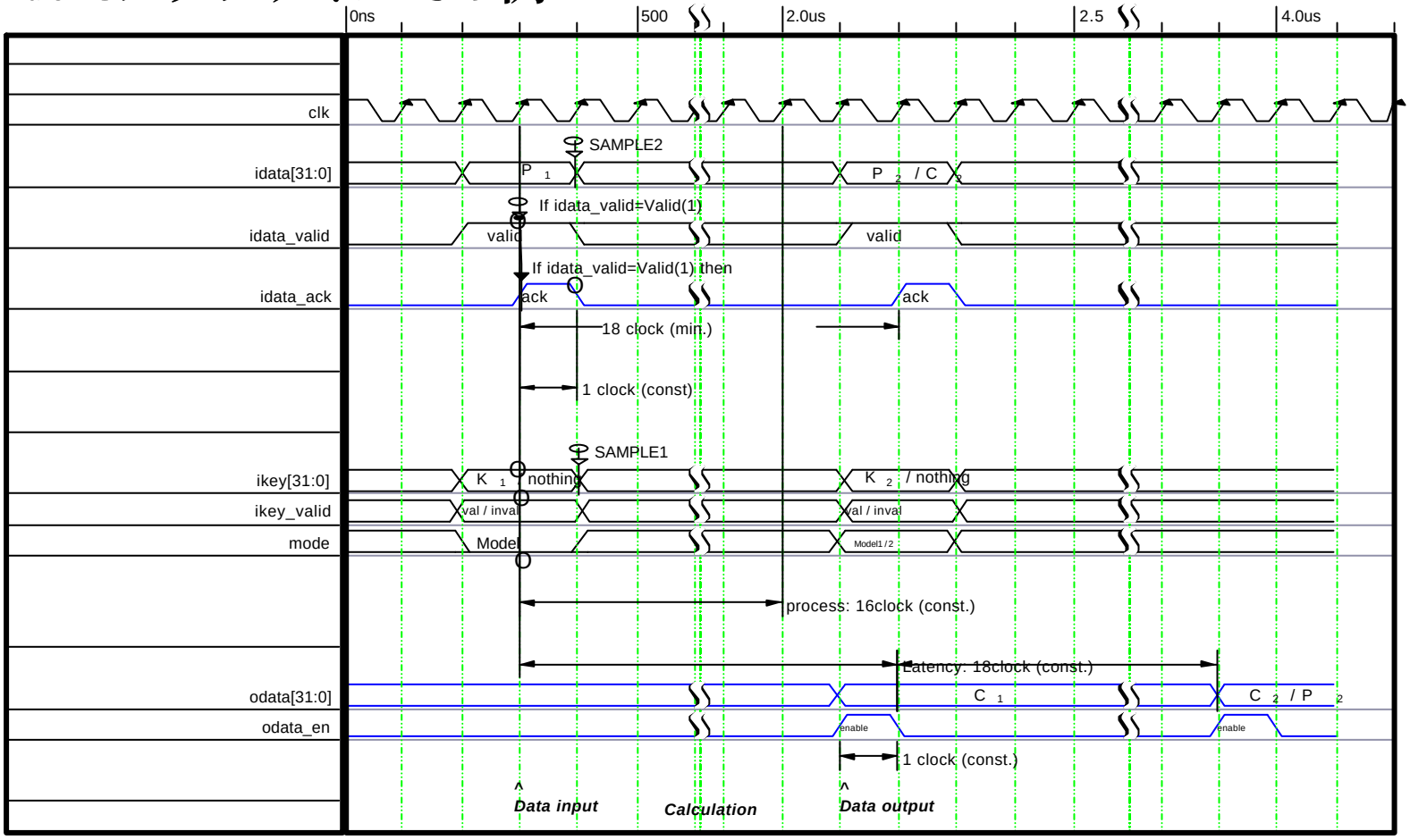
波形の例

評価用プログラムの Examples ディレクトリの中には、この他にも色々な波形例がございます。ご確認ください。



MS-Wordへの波形貼り付け

Windowsメタファイルでの例



TestBencher Pro

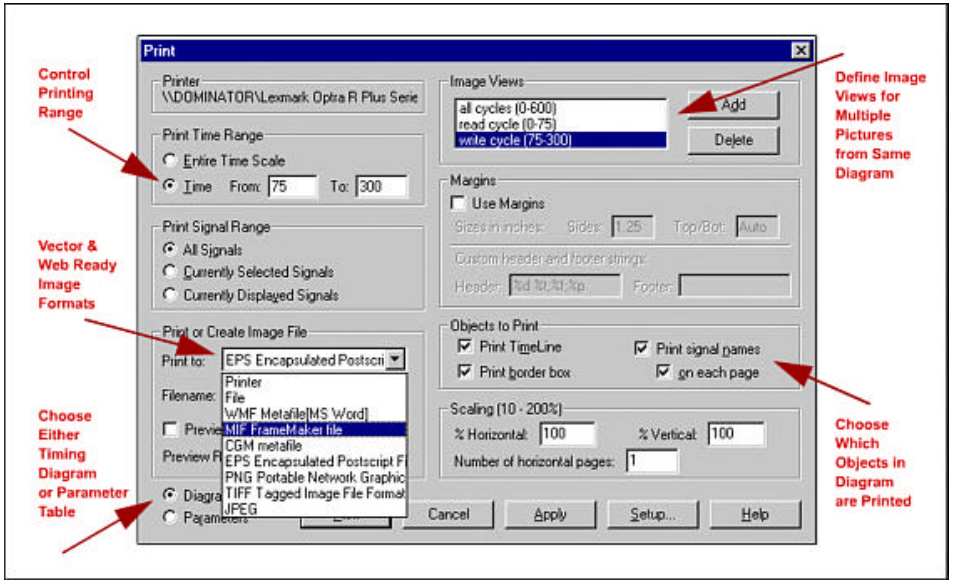
InterLink

SYNAPTICAD

MS-Wordへの波形貼り付け（続き）

対応フォーマット

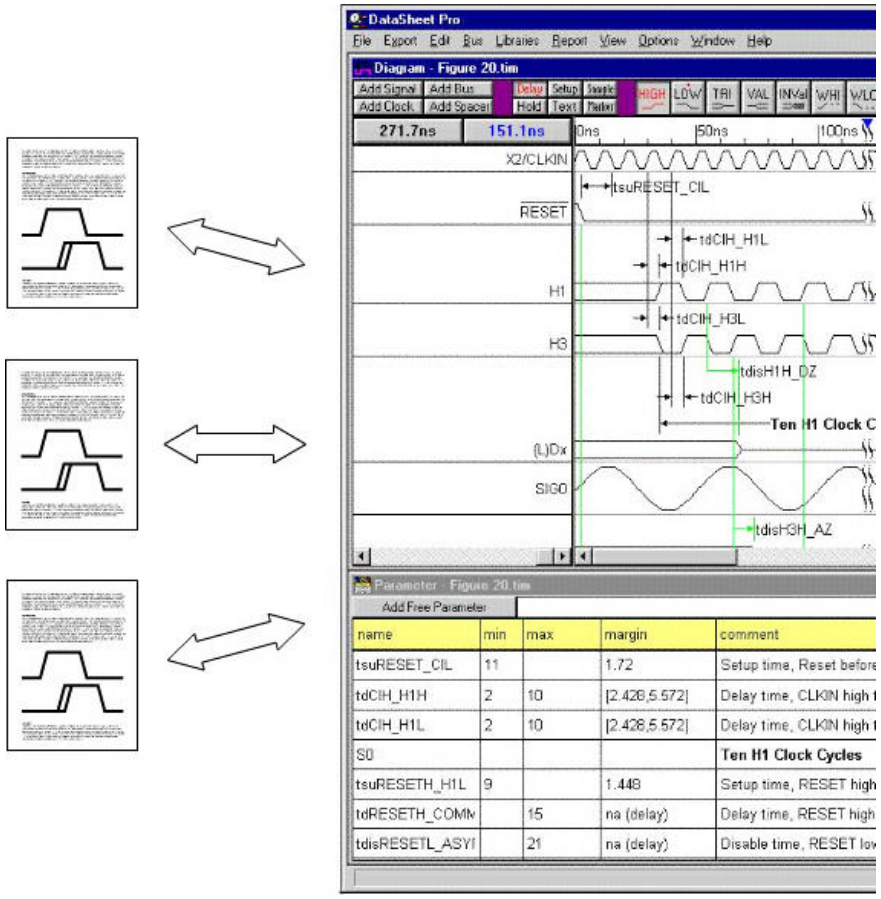
- Windowsメタファイル（.wfm）、フレームメーカMIFファイル、EPSファイル、CGMファイル、EMFファイル
- TIFF、PNG、JPEGはDataSheet Proで対応



MS-Wordへの波形貼り付け（続き）

OLE機能

※TestBencher Proでは、
オプション機能として
追加が必要



TestBencher Pro 製品概要

設計が大規模・複雑になるにしたがい、HDL設計全行程のうち、テストベンチ生成の工数の占める割合が多くなりました。

=> テストベンチ作成の為の工数を減らす手段が必要。。。

- ✧ TestBencherは、テストベンチ生成とメンテナンスに要する時間を削減します。
- ✧ テストベンチ開発の局面で遭遇する自動化できる作業（ポート情報の管理など）をTestBencherに行わせる事で、設計者は、本来の設計作業（トランザクション、シーケンスなど）に注力できます。
- ✧ 言語に非依存のタイミングダイヤグラムから、テストベンチを生成します。

TestBencher Pro 製品概要（続き）

- ✧ グラフィカルなタイミングダイヤグラムからバスファンクション・モデルを生成します。
- ✧ サイクルベース、イベントベース、両方のテストベンチをサポートします。
- ✧ MUT信号を抽出 － MUTからポート情報を自動抽出し、正しい信号属性の状態で、ダイヤグラムにセットアップします。

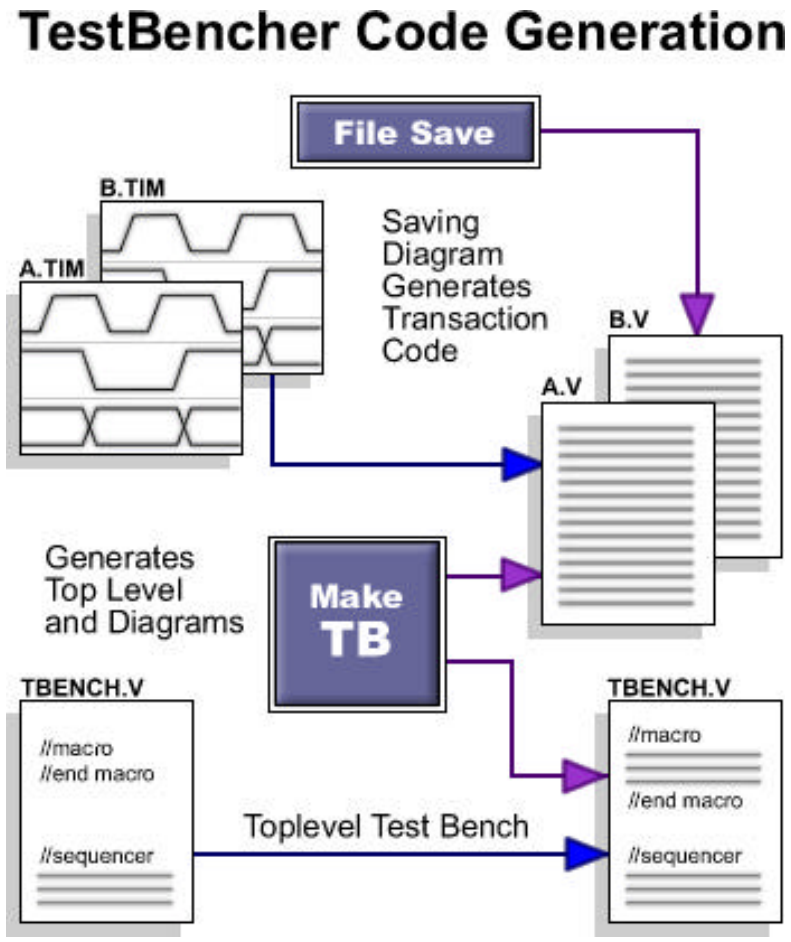
  =>方向、タイプ、サイズは自動でセットアップされます。
- ✧ Sample 機能 － MUTの出力結果や、他のダイヤグラムトランザクション結果に応答するテストベンチを生成します。
- ✧ クラス生成 － ユーザ定義のクラス（データ構造）を生成キュー、アレイなど

コード生成アーキテクチャ

✓「タイミング・ダイヤグラム」と「トップレベル・テンプレートファイル」からテストベンチを生成します。

✓トップレベル・コードはトランザクションのシーケンスやシミュレーション中の信号モニタをコントロールします。

✓ダイヤグラム保存時に、各トランザクションのコードは生成されます。
→ダイヤグラムの編集結果をすぐに確認できます。



TestBencher Pro

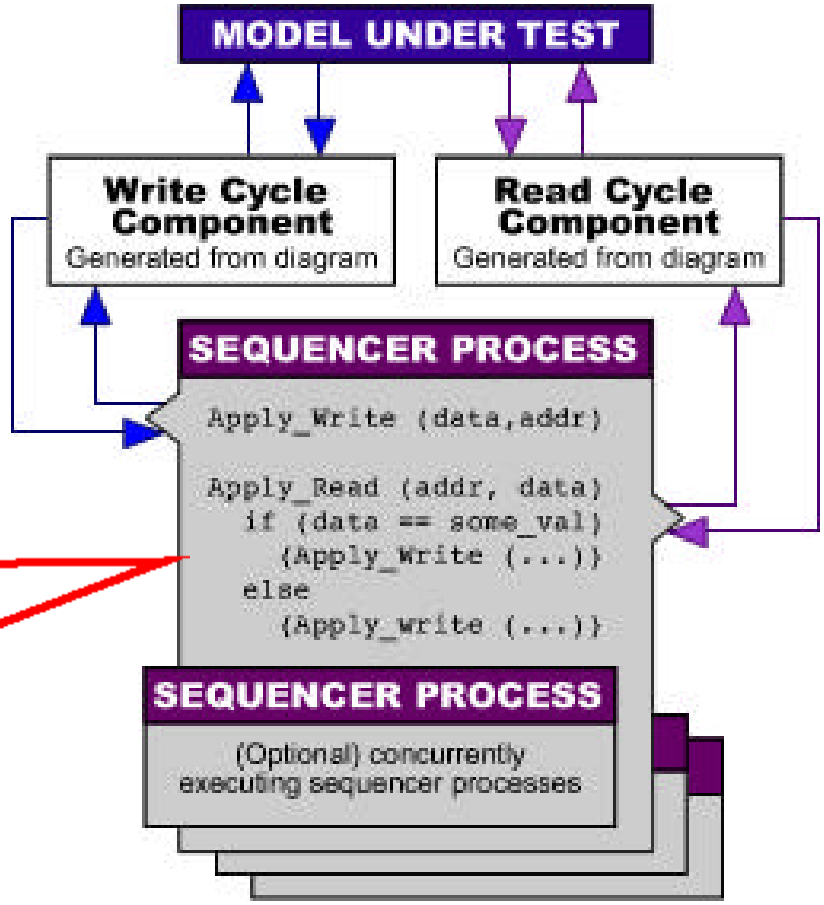
InterLink

SYNAPTICAD

コード生成アーキテクチャ（続き）

各タイミング・トランザクションごとのタスクが生成されます。

シーケンサ・プロセスで、MUTに対するタイミング・トランザクションの実行順序をコントロール

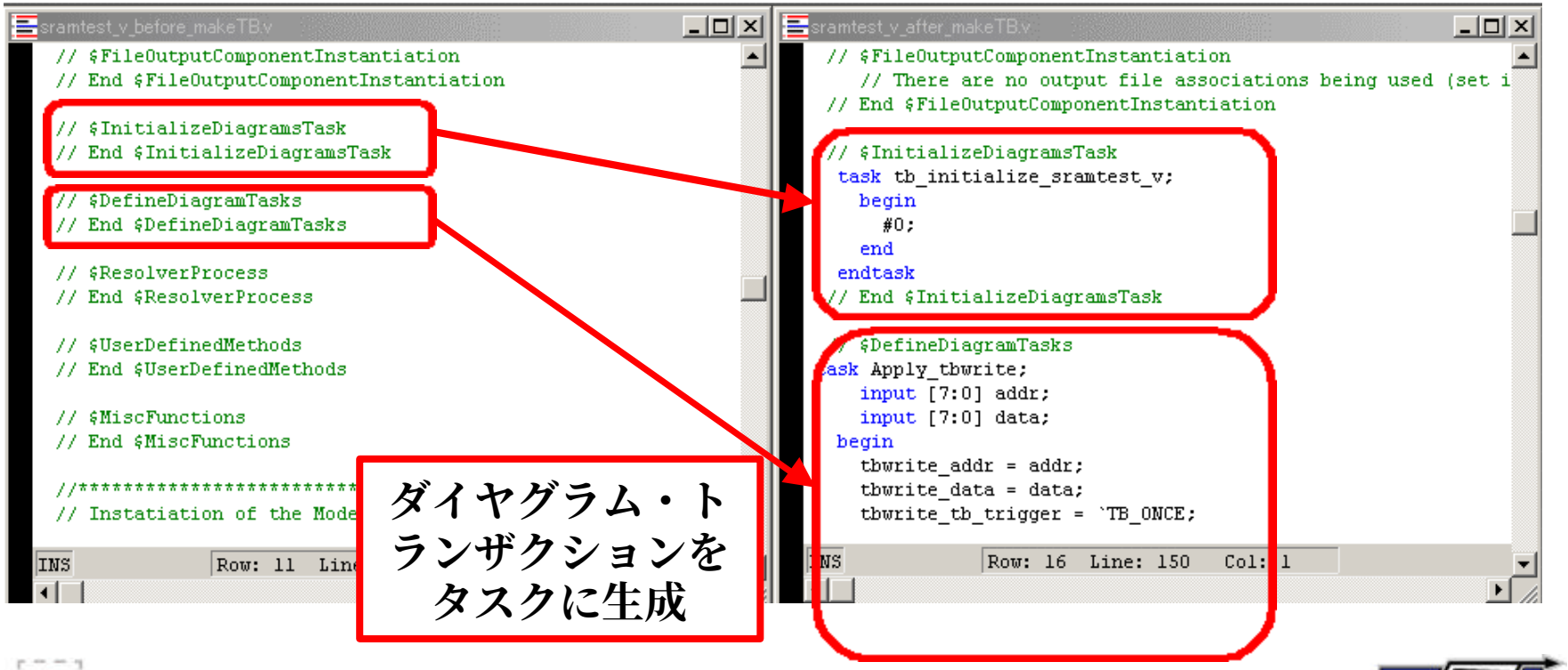


コード生成アーキテクチャ（続き）

トップレベルテンプレートファイルの展開（makeTBボタン実行）前後

makeTB実行前

makeTB実行後



TestBencher Pro

InterLink

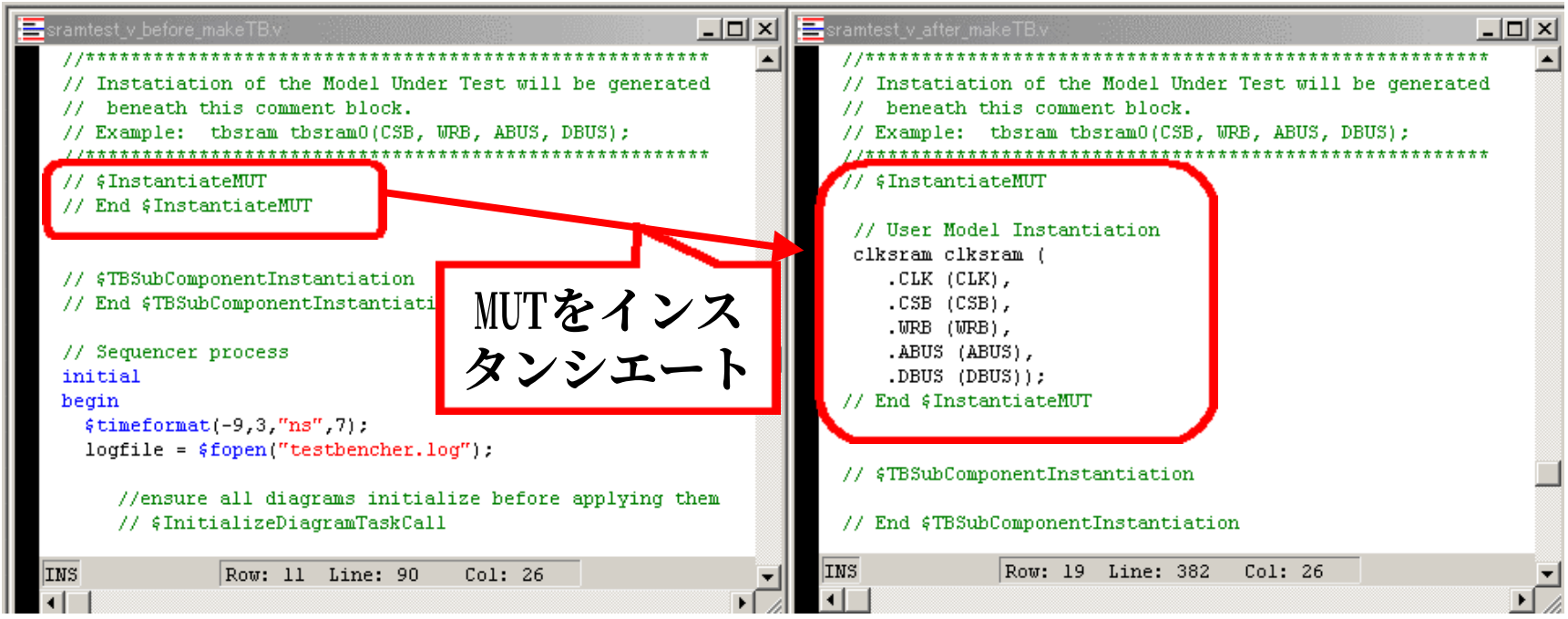
SYNAPTICAD

コード生成アーキテクチャ（続き）

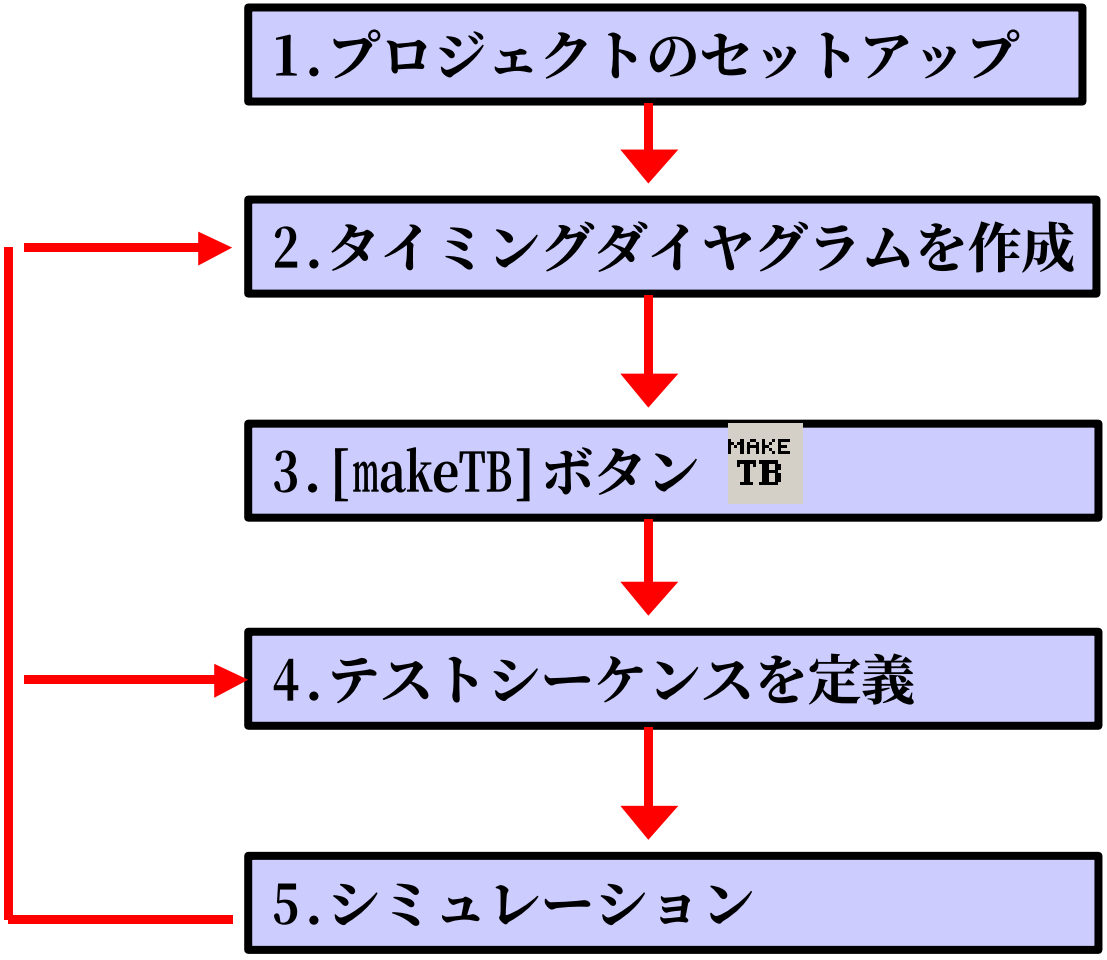
トップレベルテンプレートファイルの展開（makeTBボタン実行）前後

makeTB実行前

makeTB実行後

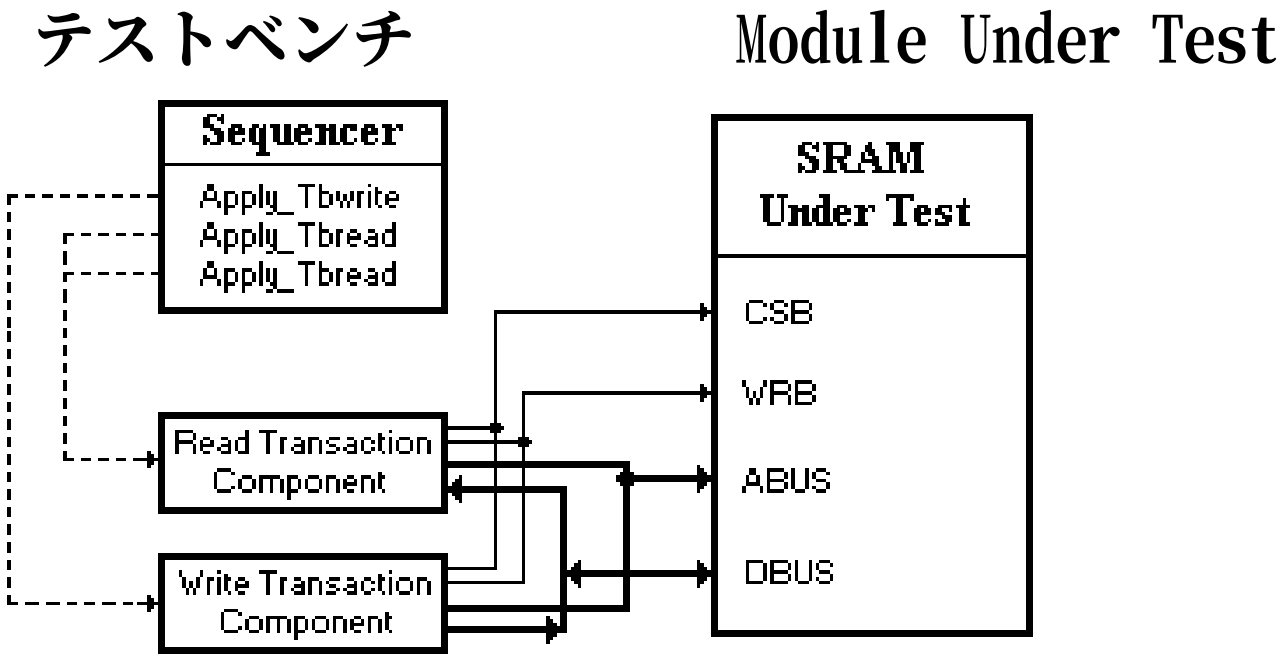


TestBencherデザインフロー



TestBencherデザインフロー
―例：SRAMテスト（Tutorial5）―

操作デモ用回路



TestBencherデザインフロー
―例：SRAMテスト（Tutorial5）―

デモシナリオ

- ✓ SRAMモデルを例にとり、実際のフローのご説明と操作デモを行います。
- ✓ このデモではまず、使用言語として Verilog を選択し、TestBencher 内蔵の Verilog シミュレータを使って、シミュレーションまでをご覧いただきます。
- ✓ Sample 機能をご確認いただく為に、ワーニングが発生するよう、故意にエラーを記述しています。
- ✓ 次に、*e /Verilog* 言語を選択し、*e* 言語が生成される事をご覧ください。
- ✓ また、Sample の設定内容が、*e* のキーワード"expect"を用いて生成される例をご覧ください。

TestBencherデザインフロー
―例：SRAMテスト（Tutorial5）―

1.プロジェクトのセットアップ

- ✓ウィザードによりプロジェクト設定
- ✓MUTをプロジェクトへ追加
- ✓MUTの信号情報を抽出 

2.タイミングダイヤグラムを作成

3.[makeTB]ボタン 

リセット（tbreset）、
ライト(tbwrite)、
リード(tbread)の
各トランザクションを作成

4.テストシーケンスを定義

5.シミュレーション

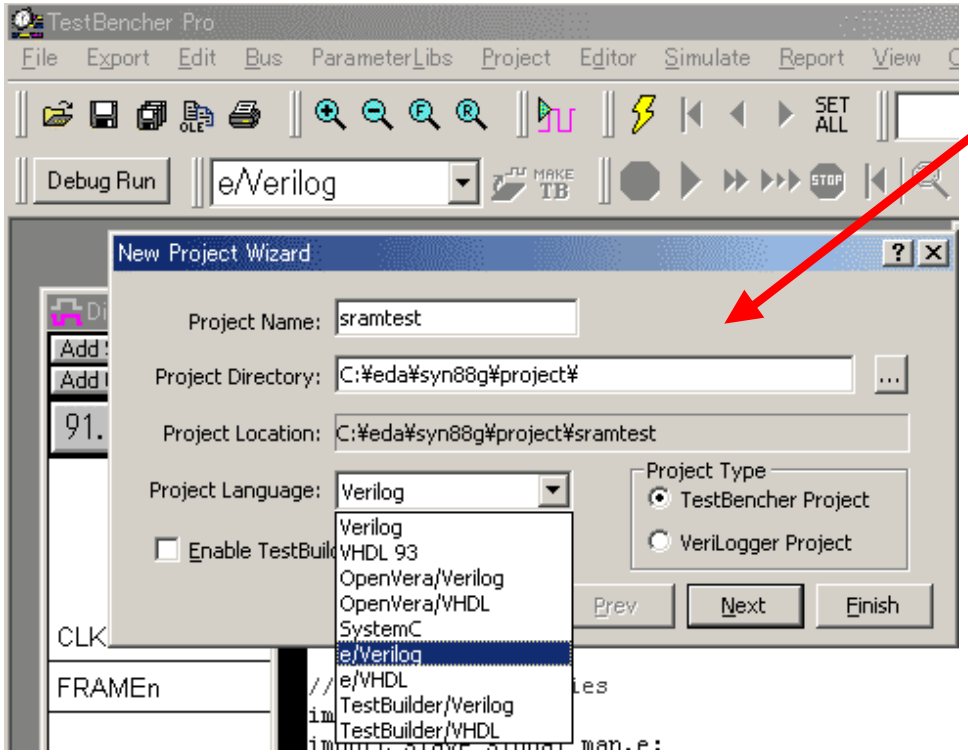
ダイヤログで各トランザクションを
トップレベルテストベンチ上に呼び
出す

InterLink

TestBencherデザインフロー
—例：SRAMテスト（Tutorial5）—

1.プロジェクトのセットアップ

✓ ウィザードによりプロジェクト設定



✓プロジェクト名、作業ディレクトリ、使用する言語（e、OpenVera等）を選択

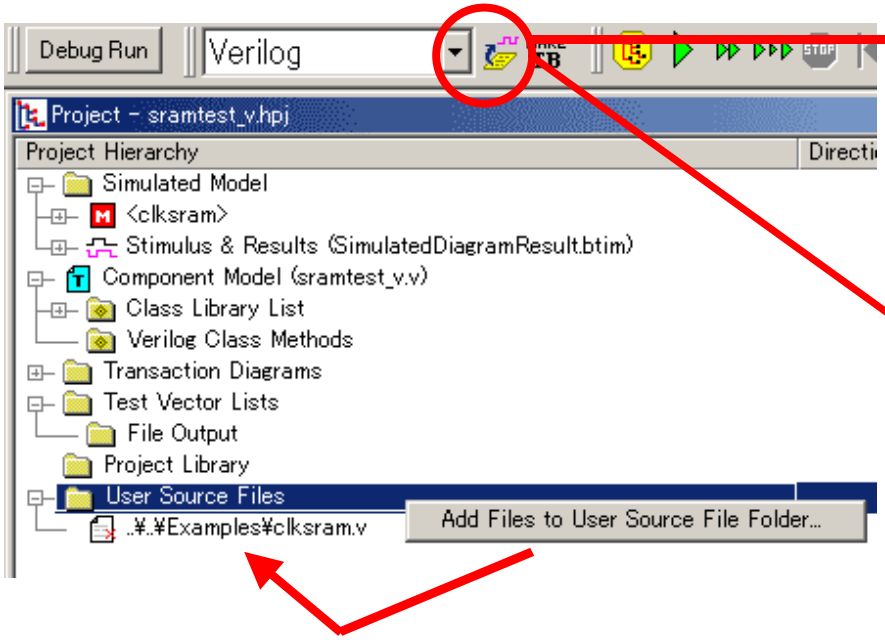
✓サイクルベースに必要なデフォルトクロック名なども設定

TestBencherデザインフロー
—例：SRAMテスト（Tutorial5）—

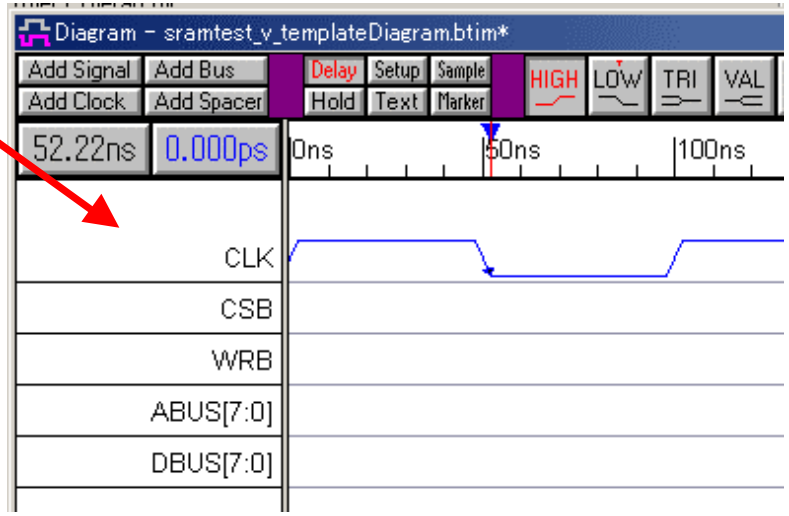
1.プロジェクトのセットアップ

✓ MUTをプロジェクトへ追加

✓MUTの信号情報を抽出



信号情報がダイヤグラムエディタにセットアップされる



コンテキストメニューを使って
MUTをプロジェクトへ追加

TestBencher Pro InterLink

SYNAPTICAD

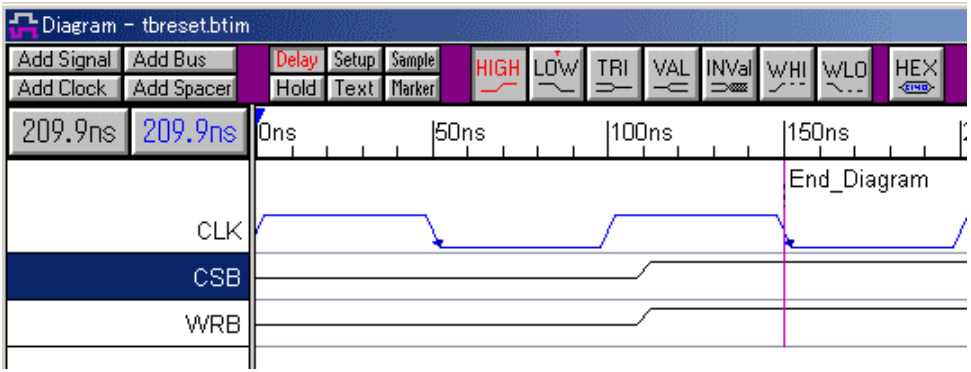
TestBencherデザインフロー
—例：SRAMテスト（Tutorial5）—

2.タイミングダイヤグラムを作成

✓ リセット（tbreset）

ステートボタン、モードボタン、マーカ等を使用してタイミングダイヤグラムを描画する

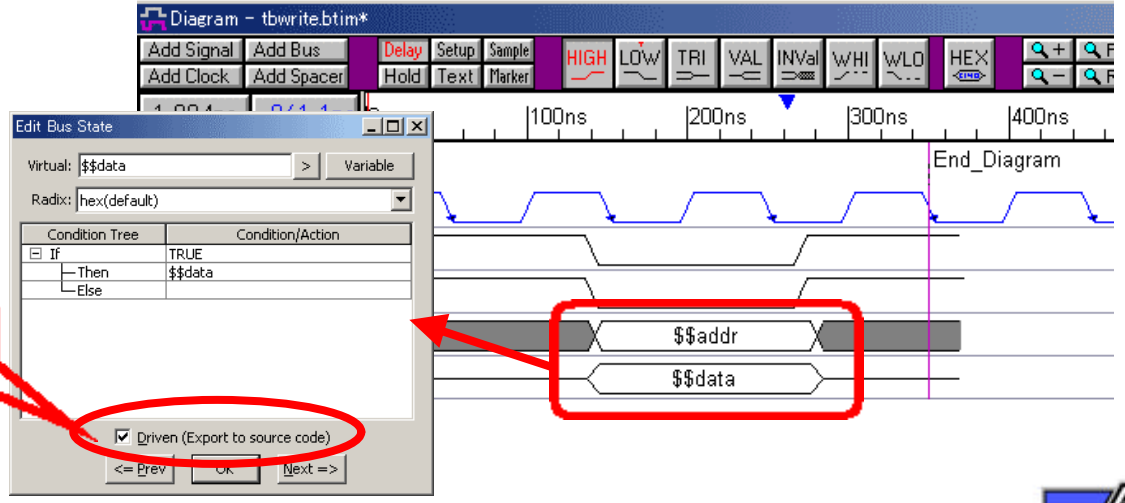
描画したダイヤグラムはプロジェクトへ追加する



✓ライト（tbwrite）

ABUS（アドレス・バス）、DBUS（データ・バス）の値は変数を用いる

信号方向のコントロール



TestBencher Pro InterLink

SYNAPTICAD

TestBencherデザインフロー
—例：SRAMテスト（Tutorial5）—

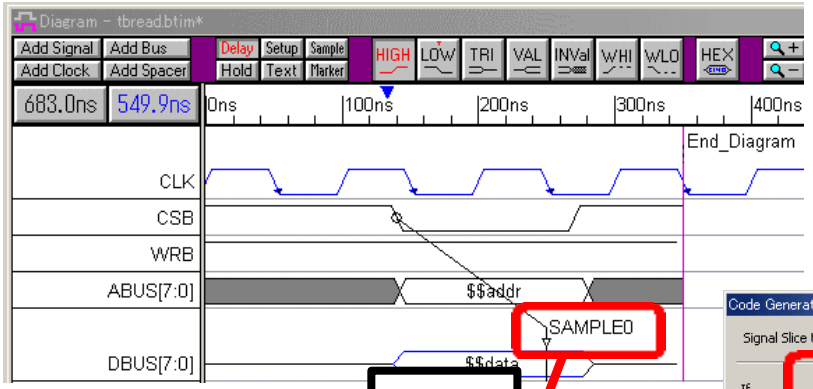
2.タイミングダイヤグラムを作成

✓ リード（tbread）

ライト・トランザクションと
リード・トランザクションとは、
「WRBの極性」と「DBUSの方向」
が違う

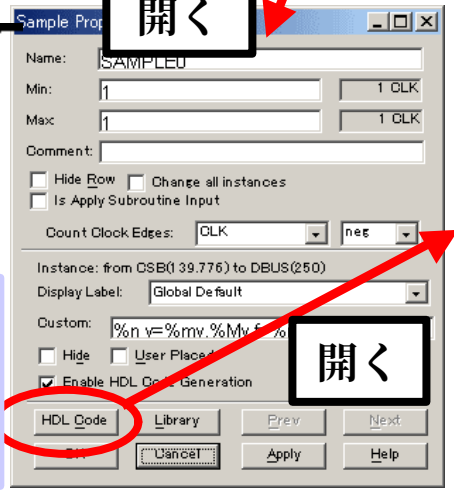
↓

tbwrite の名前を変えて保存&
編集して tbread を作成する

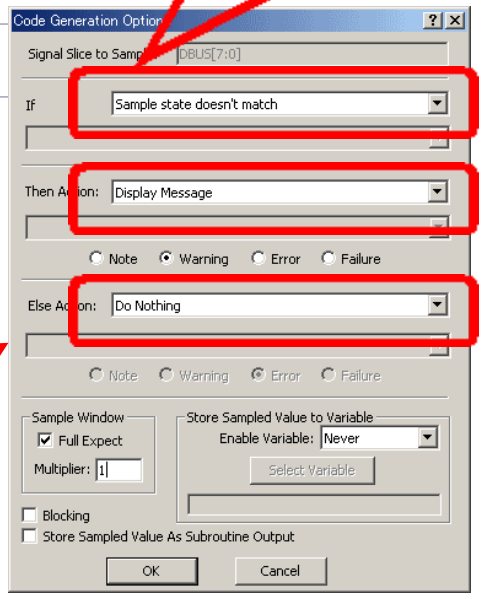


If～
Then～
Else～
のコンディション設定

開く



開く



Sample（サンプル）の振る舞いは、ダイヤログで設定する

SRAMテストでの Sample0 設定内容：
If「DBUSの値が期待値と合っていない場合には」
Then「ワーニングメッセージを表示させる」
Else「（期待値通りなので）何もしない」

TestBencher Pro

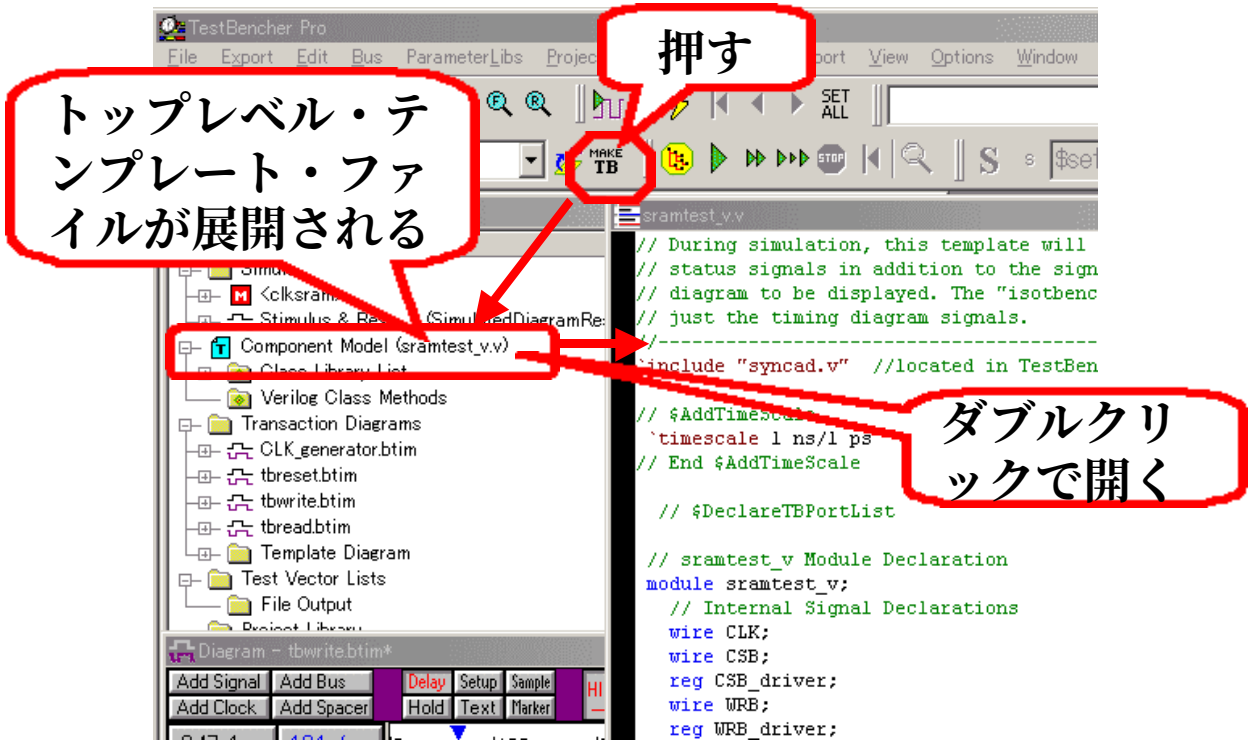
InterLink

SYNAPTICAD

TestBencherデザインフロー
―例：SRAMテスト（Tutorial5）―

3.[makeTB]ボタン

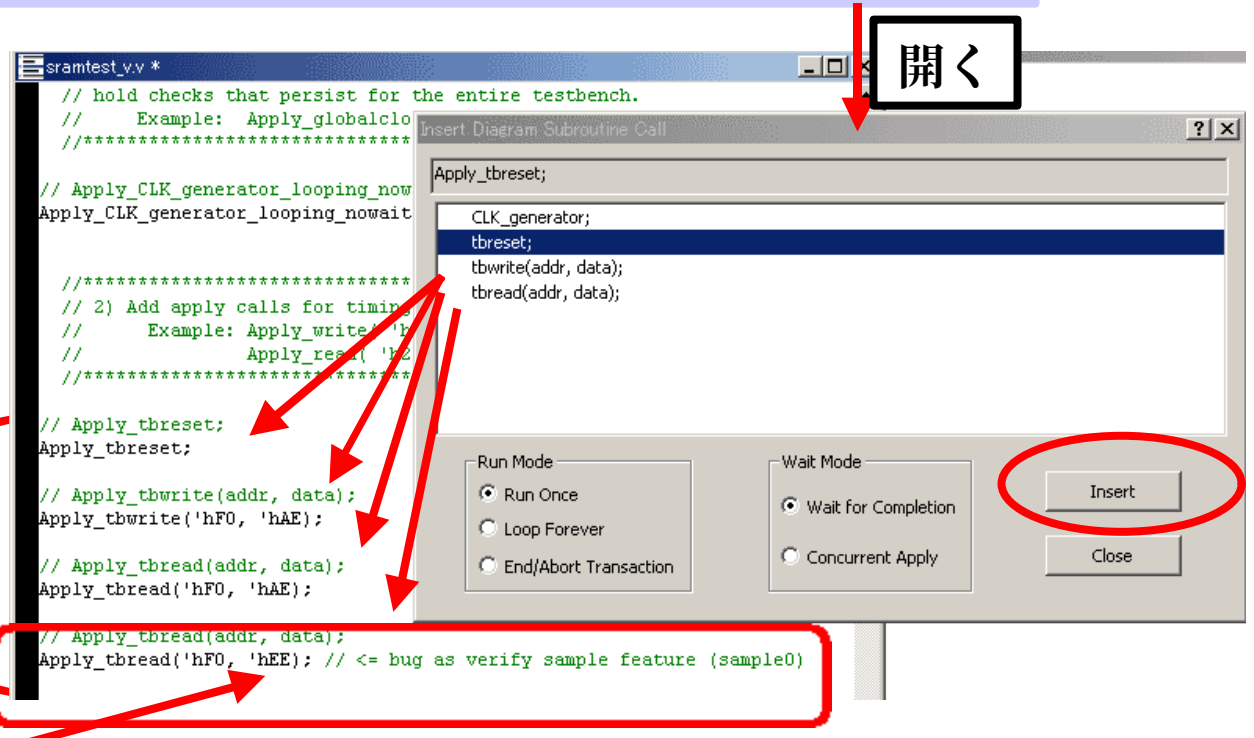
タイミングトランザクションを描画し終えたら、[makeTB]ボタンを押し、トップレベル・テンプレート・ファイルを展開します。



TestBencherデザインフロー
―例：SRAMテスト（Tutorial5）―

4.テストシーケンスを定義

トップレベル・テストベンチ・エディタ上でマウス右クリック
→メニューから[Insert Diagram Calls...]を選択する



ここでの定義
リセットx1
ライトx1
リードx2

Sample 機能を確認する為、２回目のリードでは誤った値をリードしてみる

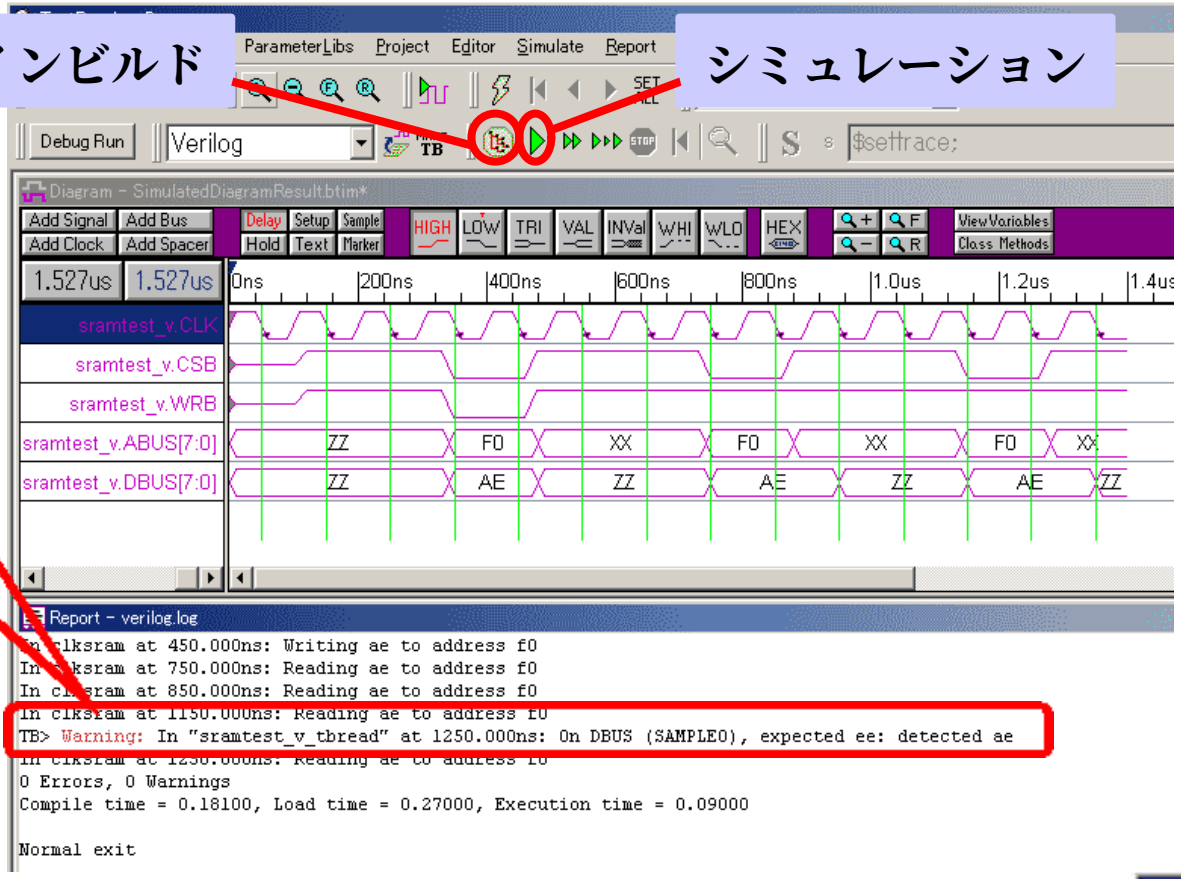
TestBencherデザインフロー
—例：SRAMテスト（Tutorial5）—

5.シミュレーション

TestBencher Pro内蔵のVerilogシミュレータでシミュレーション

デザインビルド

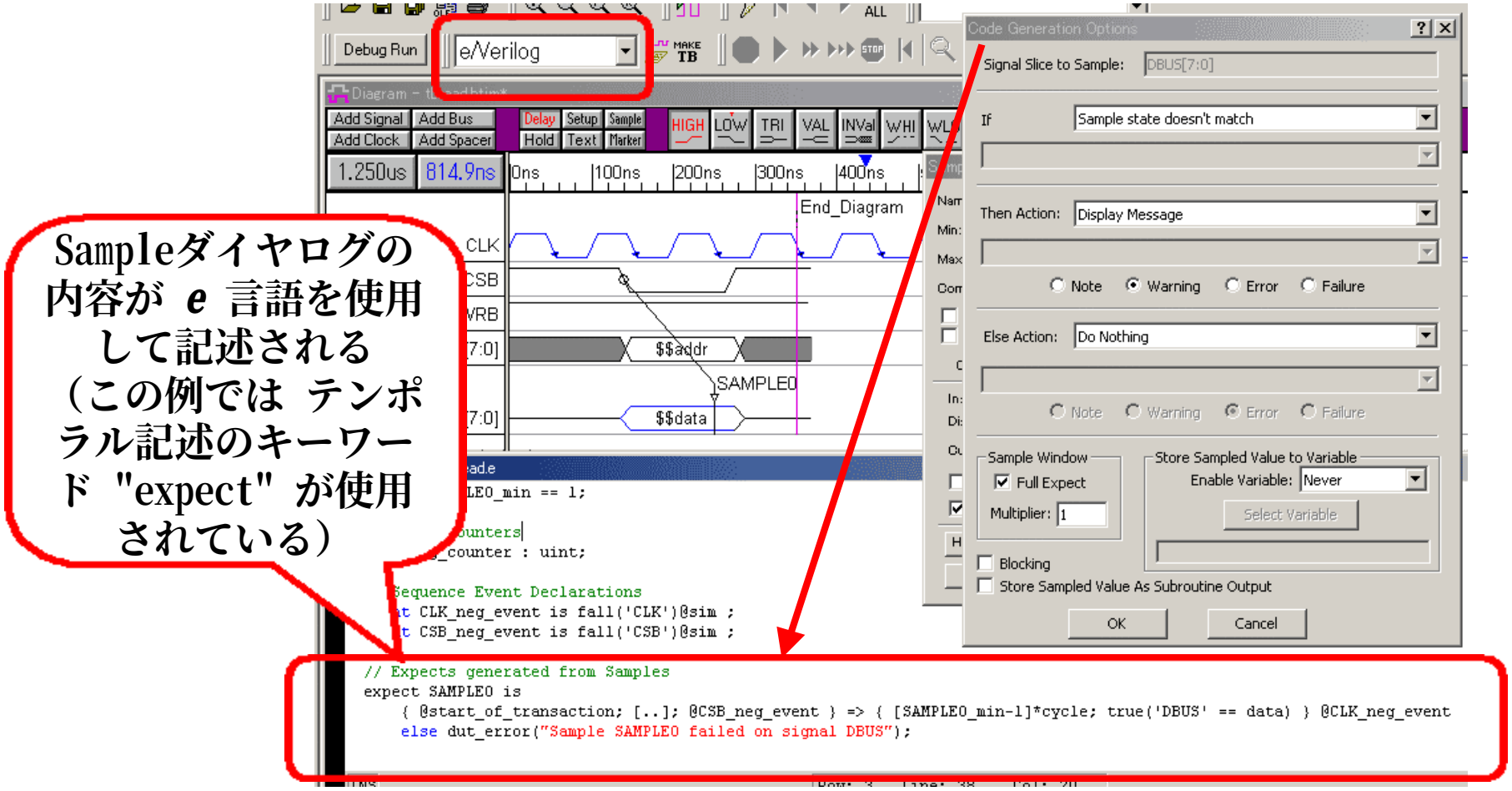
シミュレーション



Sample ダイヤログの設定内容により、レポートが表示される

TestBencherデザインフロー
―例：SRAMテスト（Tutorial5）―

SRAM テストで e/Verilog 言語を選択の場合：



Sampleダイヤログの内容が *e* 言語を使用して記述される（この例では テンポラル記述のキーワード "expect" が使用されている）

操作デモ

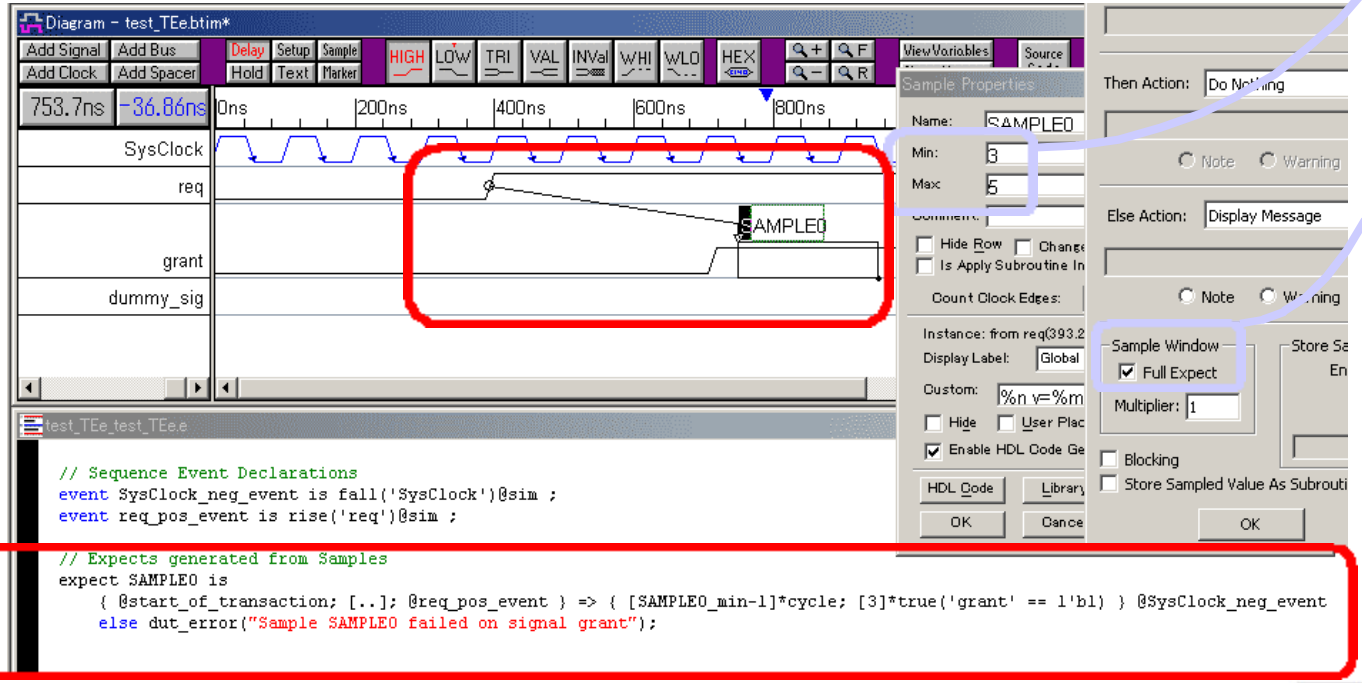
- SRAMテスト－一連の操作をご覧ください。
- Examplesディレクトリの中にある、PCIやATMのタイミングダイヤグラムをご確認ください。

  →タイミングダイヤグラム・ウィンドウ内の [Source Code]ボタンを押すことで、表示されている波形に対応する *e* や VERA などのコードが確認できます。

e テンポラル記述－出力例

Window サンプル; Full Expect チェック："req" の立ち上がり３クロック～５クロックで "grant" が立ち上がる → "grant" のスタビリティ・チェック

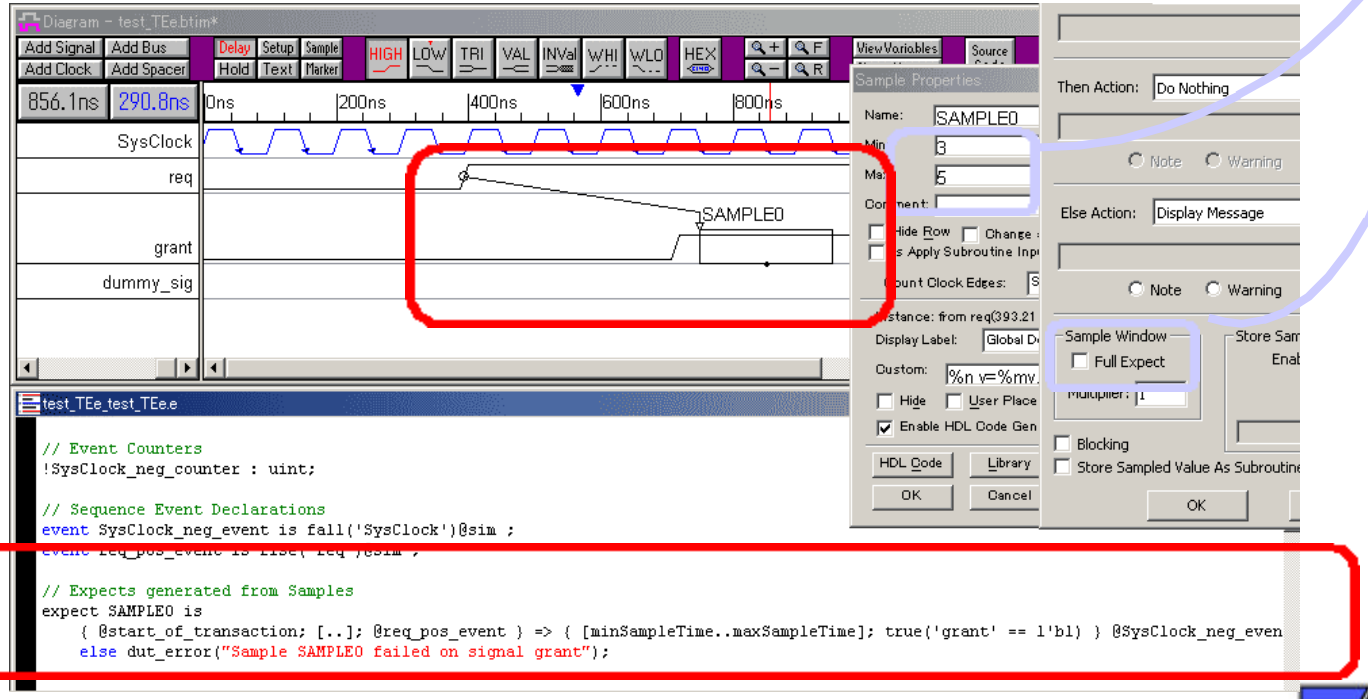
```
// Expects generated from Samples
  expect SAMPLE0 is
      { @start_of_transaction; [..]; @req_pos_event } => { [SAMPLE0_min-1]*cycle;
[3]*true('grant' == 1'b1) } @SysClock_neg_event
      else dut_error("Sample SAMPLE0 failed on signal grant");
```



e テンポラル記述－出力例

Window サンプル; Full Expect チェック外す："req" の立ち上がり３クロック～５クロックで
"grant" が立ち上がる → 指定クロック内で "grant" がアサートするかチェック

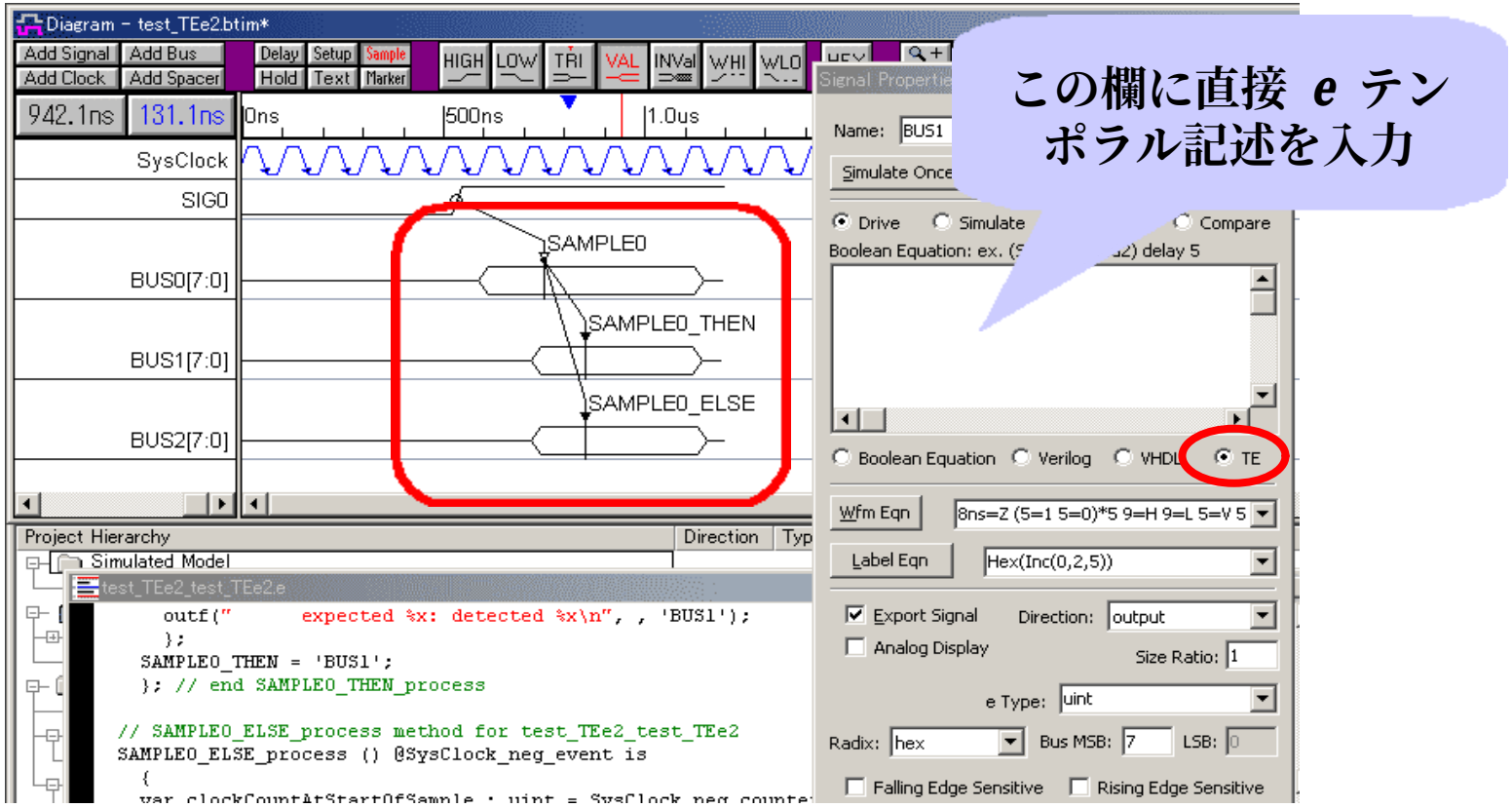
```
// Expects generated from Samples
  expect SAMPLE0 is
      { @start_of_transaction; [..]; @req_pos_event } => {
[minSampleTime..maxSampleTime]; true('grant' == 1'b1) } @SysClock_neg_event
      else dut_error("Sample SAMPLE0 failed on signal grant");
```



e テンポラル記述－出力例

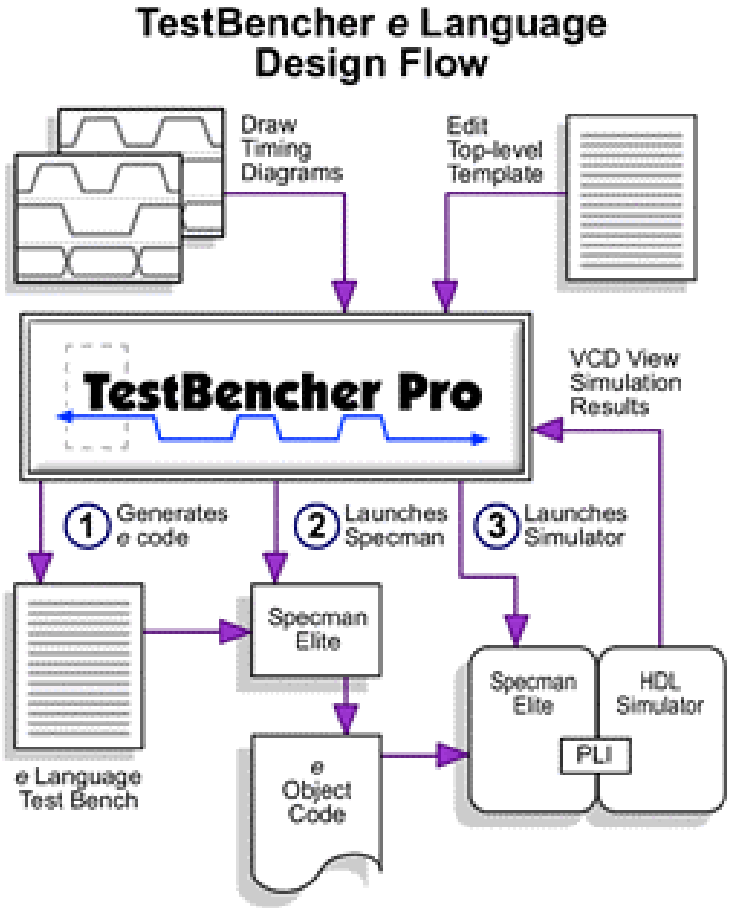
✓Sample 間の If ～ Then ～ Else 定義も可能

✓ 「Signal Properties」ダイヤログを使って、直接e言語を記述する事も可能



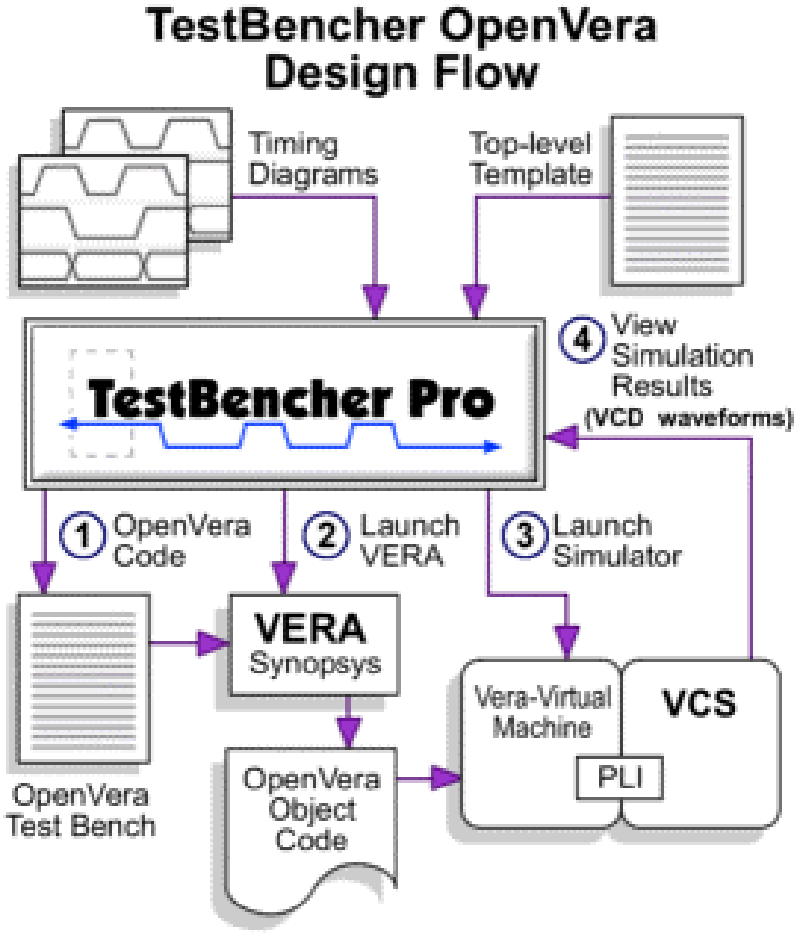
TestBencherデザインフロー：e

eでのデザインフロー



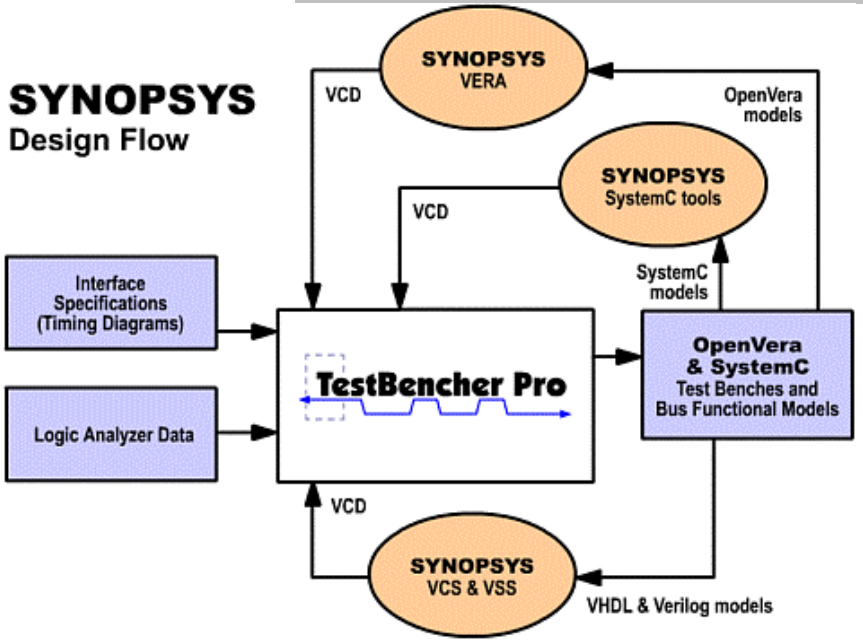
TestBencherデザインフロー：OpenVera

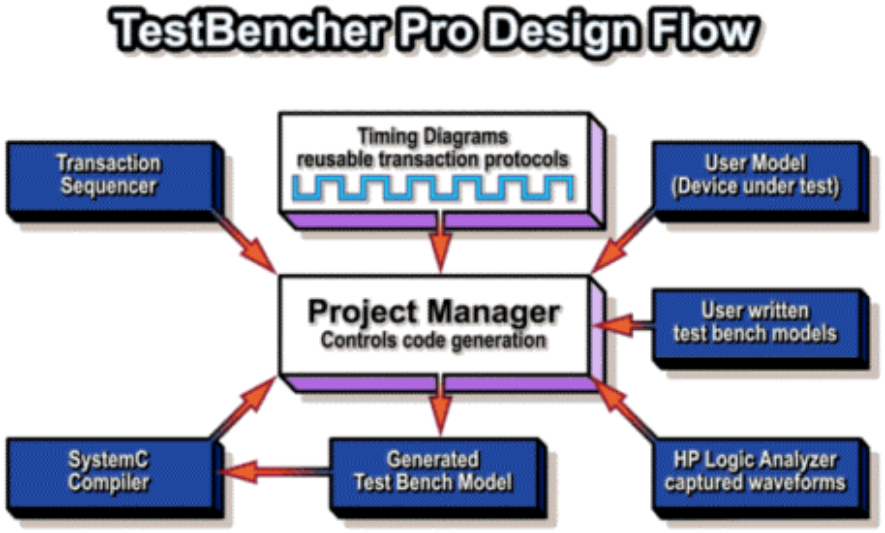
OpenVeraでのデザインフロー



TestBencher Pro InterLink SYNAPTICAD

TestBencherデザインフロー：OpenVera、SystemC





TestBencher Pro

InterLink

SYNAPTICAD

TestBencherデザインフロー：VHDL、Verilog、TestBuilder

TestBuilder でのデザインフロー