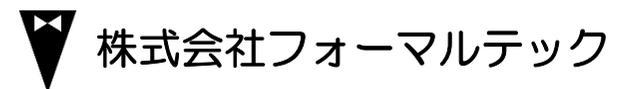


【フォーマルテックがご提供するサービス】

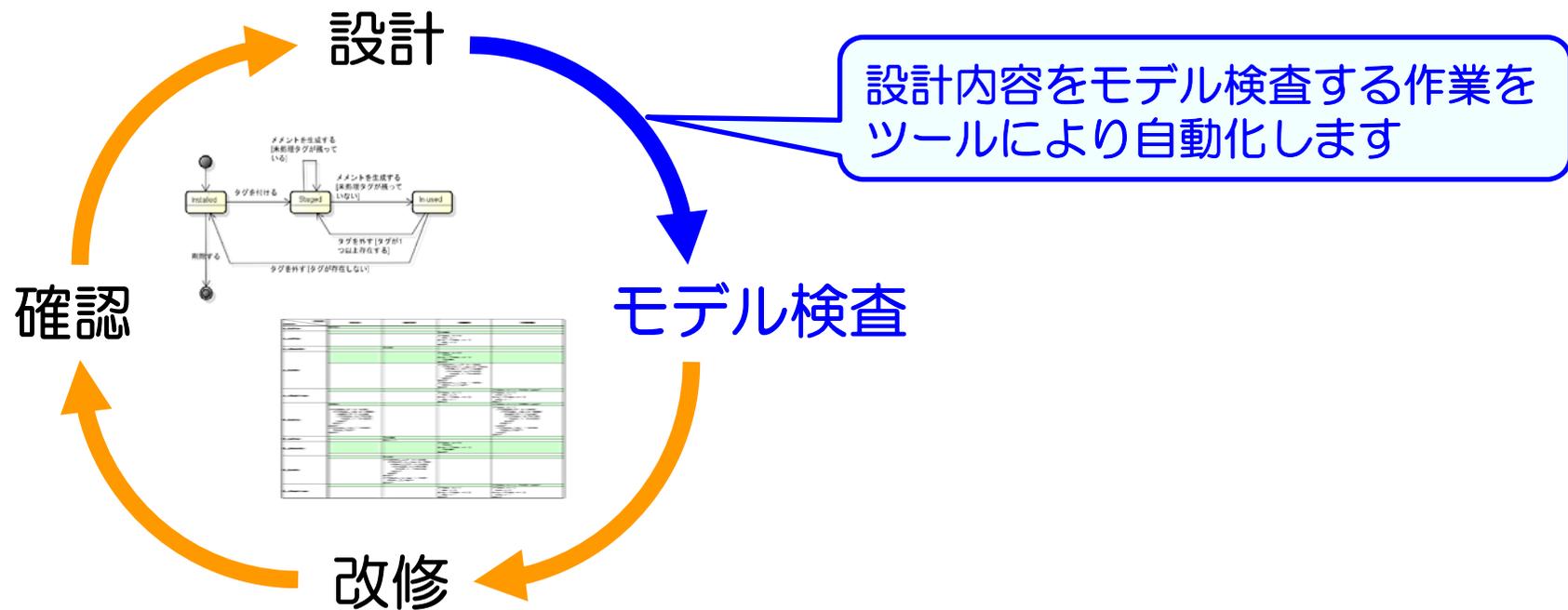
# 設計モデル検査ツールの開発



# 1. 概要

設計プロセスでのモデル検査を自動化するツールの開発を承ります。

設計 → モデル検査 <sup>不具合があれば</sup> → 改修 → 確認のサイクルを繰り返して  
システム開発の上流プロセスでの品質向上を図ります。



## 2. 設計手法について

御社が採用している設計手法に合わせて開発します。

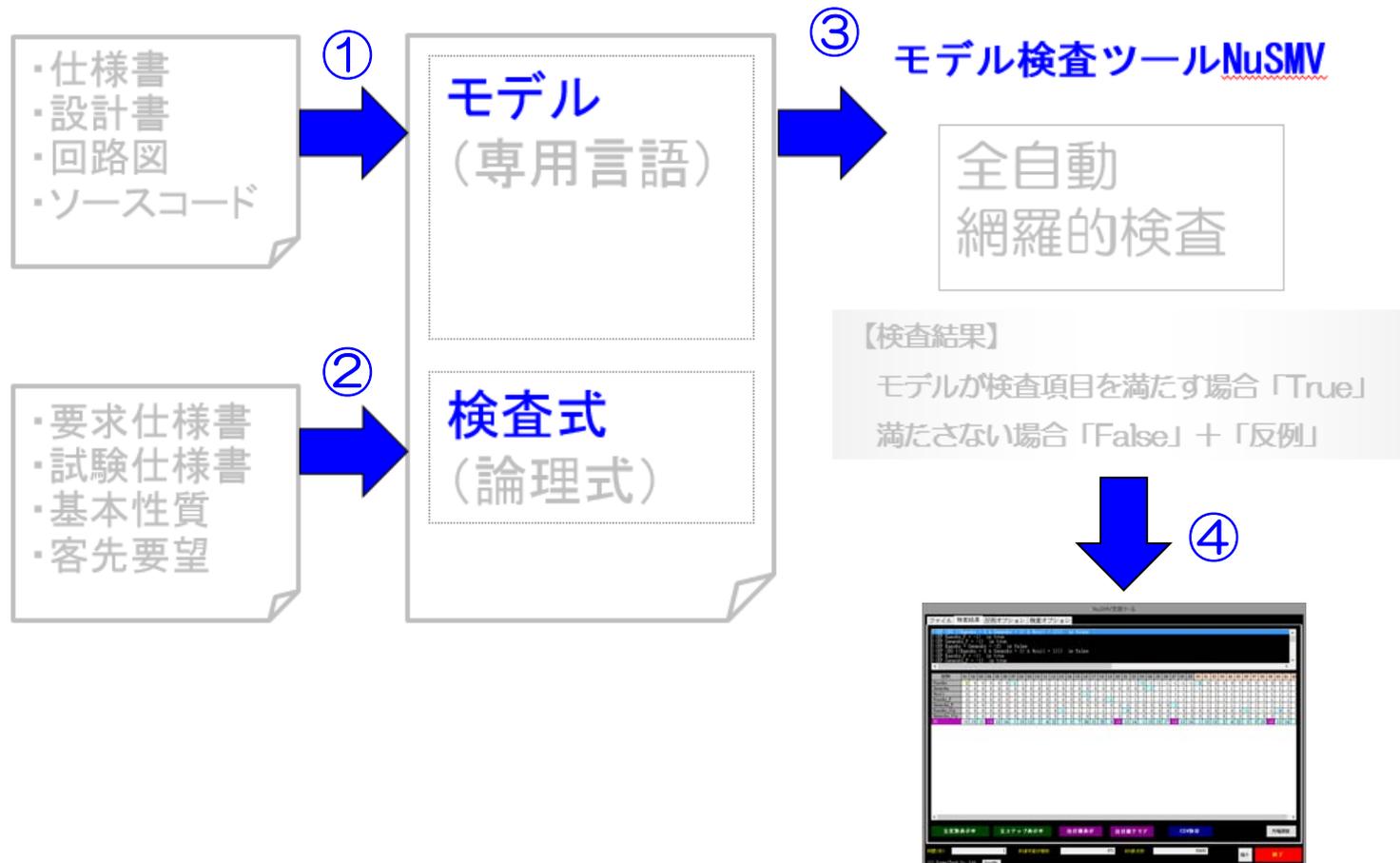
### 設計手法の例

- > 状態遷移図／状態遷移表
- > UML図
- > フローチャート
- > 回路図／ブロック図
- > 擬似プログラミング言語
- > ソースコード※
- > 独自の設計手法
- > その他（理論的に記述されていれば対応可能です）

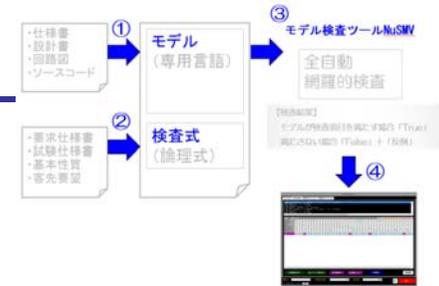
※ モデル検査ツールの専用言語は表現力の低い低級言語であるため、ソースコードについては全てを自動化できない場合もあります（応相談）。

### 3. 自動化(ツール化)が可能な作業

図の ➡ の部分の自動化が可能です。



## 4. 自動化（ツール化）の詳細



### ① 自動モデル生成

各種設計ツールからエクスポートしたデータを入力してモデル検査ツールNuSMVの専用言語のモデルを自動生成します。

### ② 検査式の作成

安全性 : XXXは絶対に発生しない (起こらない)  
活性 : (YYYが起きれば) ZZZが必ず起こる

デッドロックや安全性、活性等の検査式を自動生成します。  
その他、自動生成できる検査式はたくさんあります。  
設計者によるドメインに特化した検査式の作成も支援します。

### ③ モデル検査ツールNuSMVの実行

ツール内でNuSMVを自動的に実行して検査結果を取得します。  
NuSMVは高速化&省メモリ化（チューンナップ）を施したものを  
ご提供します。

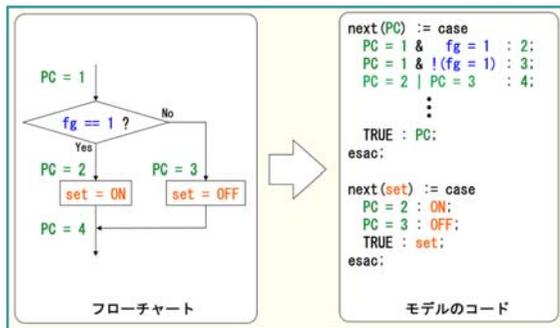
### ④ 反例の解析支援

反例を一覧表示したり、シミュレーション実行等で見やすくします。  
ご利用の設計ツールによっては、画面上にフィードバックも可能です。

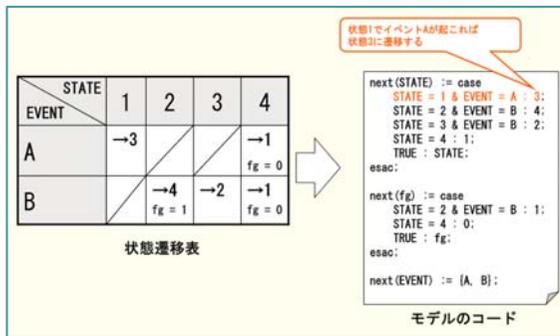
# 5. 自動モデル化の開発実績

前頁の ①自動モデル生成 の開発実績をご紹介します。

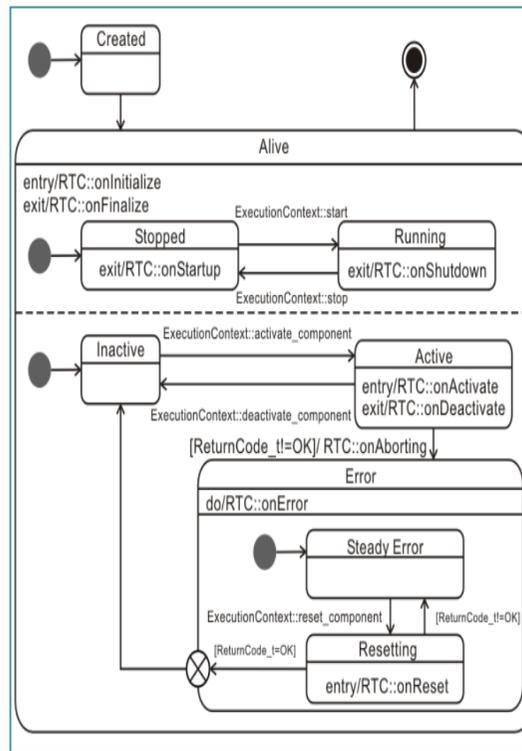
以下の4つからNuSMVのモデルを自動生成するツールを開発した実績があります。



フローチャート



状態遷移表



UML(ステートマシン図)

```

/* 加減速状況表示プログラム */
#include <stdio.h>

/* 外部変数 */
int Hyoji;
int Kasoku_P;
int Gensoku_P;

/* 関数プロトタイプ */
int indicate_state (int Kasoku, int Gensoku);

/* 関数本体 */
int indicate_state(int Kasoku, int Gensoku)
{
    /* 内部変数 */
    int Kasoku_flg;
    int Gensoku_flg;

    /* 処理開始 */
    if (Kasoku == -1) && (Gensoku == -1) {
        return Hyoji;
    }
    if (Kasoku == -1) {
        Gensoku_P = Gensoku;
        return Hyoji;
    }
    if (Gensoku == -1) {
        Kasoku_P = Kasoku;
        return Hyoji;
    }
    if (Kasoku == 1) && (Gensoku == 1) {
        return Hyoji;
    }
    Kasoku_flg = Kasoku - Kasoku_P;
    Gensoku_flg = Gensoku - Gensoku_P;

    if (Kasoku_flg == 1) {
        Hyoji = 1;
    } else {
        if (Gensoku_flg == 1) {
            Hyoji = -1;
        } else
            return Hyoji;
    }
    if (Hyoji == 1) && (Kasoku == 0) {
        Hyoji = 0;
    }
    if (Hyoji == -1) && (Gensoku == 0) {
        Hyoji = 0;
    }
    Kasoku_P = Kasoku;
    Gensoku_P = Gensoku;
    return Hyoji;
}
    
```

C言語プログラム