

# Test Standの有効活用で見えた 自動評価と汎用システムの開発手法



株式会社 ペリテック  
井澤 年宏



# 目次

- .システム開発工数を削減するには？
- .試験開発の効率化を上げる方法とは？
- .TestStandで作成するシステムのご紹介
- .TestStandの同測システムのご紹介
- .TestStandが評価から生産に向いている訳



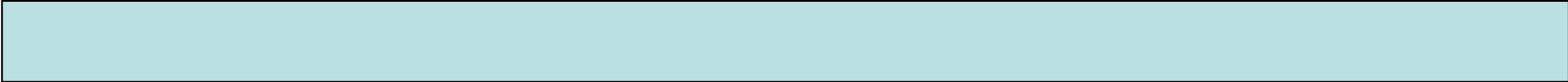

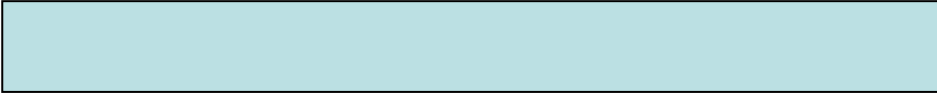



# テストスタンドの有効活用で実現できる Case

## システム開発工数の削減と 資産の再利用



## TestStandはなぜ必要？

### システム開発期間から見た効果

- 一般的な開発言語 (Visual Basic、C言語etc)  
 4ヶ月
- LabVIEW   
 2ヶ月
- TestStand  + LabVIEW   
 1ヶ月



## TestStandはなぜ必要？

### システム開発期間から見た効果

- 一般的な開発言語 (Visual Basic、C言語etc)

|    |    |          |                   |
|----|----|----------|-------------------|
| 試験 | 画面 | 全体のフロー処理 | グラフ・表などの画面表示や演算作成 |
|----|----|----------|-------------------|

4ヶ月

- LabVIEW 

|    |    |          |  |
|----|----|----------|--|
| 試験 | 画面 | 全体のフロー処理 |  |
|----|----|----------|--|

2ヶ月

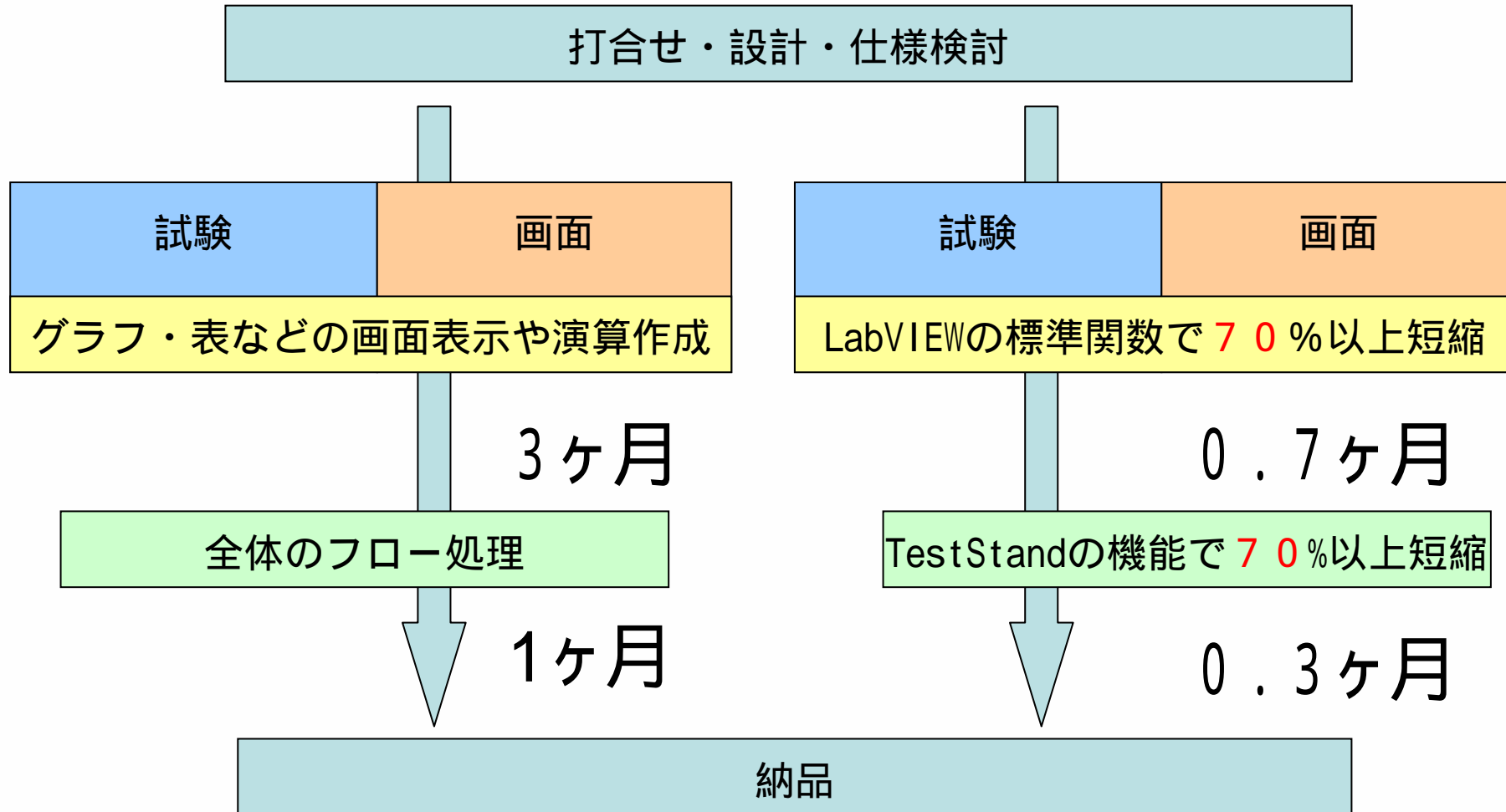
- TestStand  + LabVIEW 

|    |    |  |  |
|----|----|--|--|
| 試験 | 画面 |  |  |
|----|----|--|--|

1ヶ月

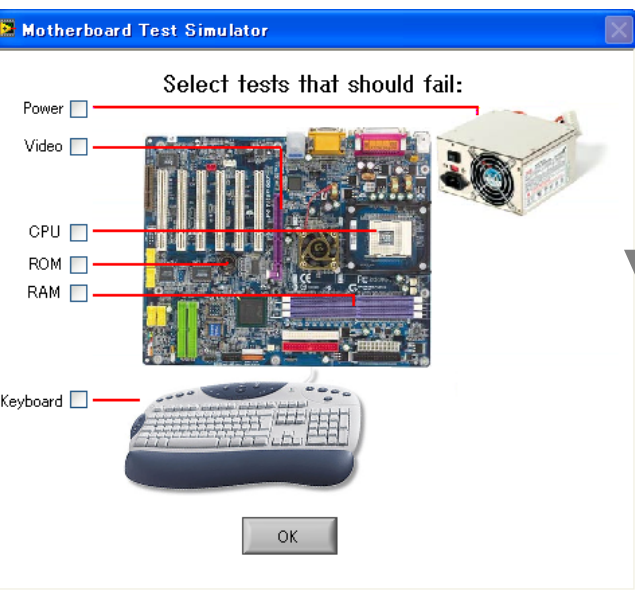


# システム開発期間での比較



# 今までの資産を再利用可能

共通の画面



デモ

**TestStand**

Step

- Setup (2)
  - Simulation Dialog
  - Turn Vacuum Table On
  - <End Group>
- Main (19)
  - Powerup Test
  - If
    - CPU Test
    - ROM Test
    - RAM Test
    - Register Test
    - Cache Test
    - Video Test
    - Keyboard Test
  - If
    - CPU Diagnostics
    - ROM Diagnostics
    - RAM Diagnostics
    - Video Diagnostics
    - Keyboard Diagnostics
  - End
  - Else
    - Powerup Diagnostics
  - End
- <End Group>
- Cleanup (1)
  - Turn Vacuum Table Off
  - <End Group>



共通のインターフェイス

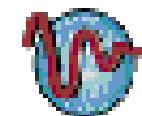
C言語

1. ROM Test
2. RAM Test



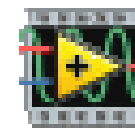
Visual Basic

1. Register Test
2. Cache Test



LabVIEW

1. Video Test
2. Keyboard Test



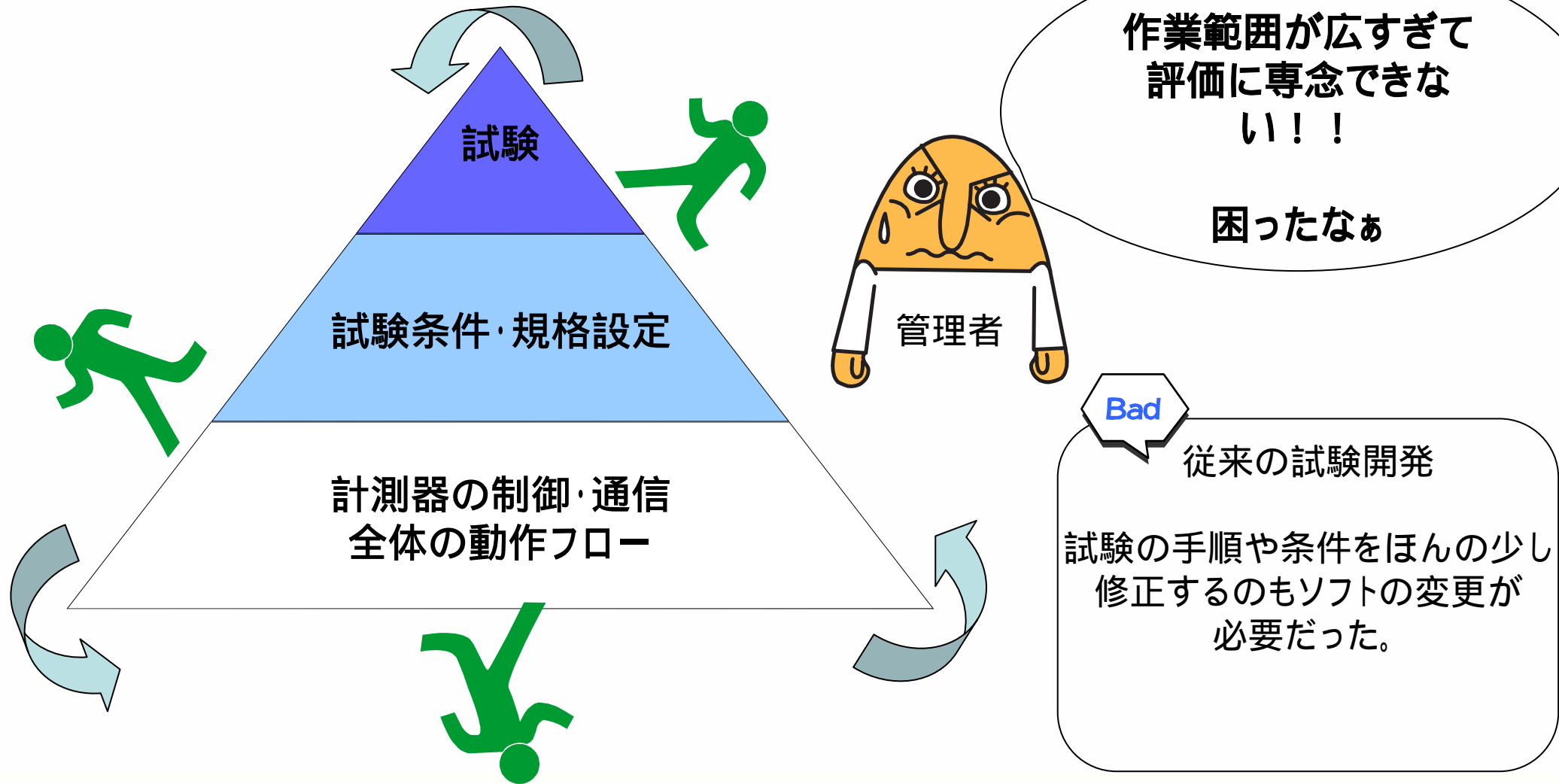
# テストスタンドの有効活用で実現できる Case

開発・検討・評価のフィードバックで  
製品・試験開発の効率化をUp

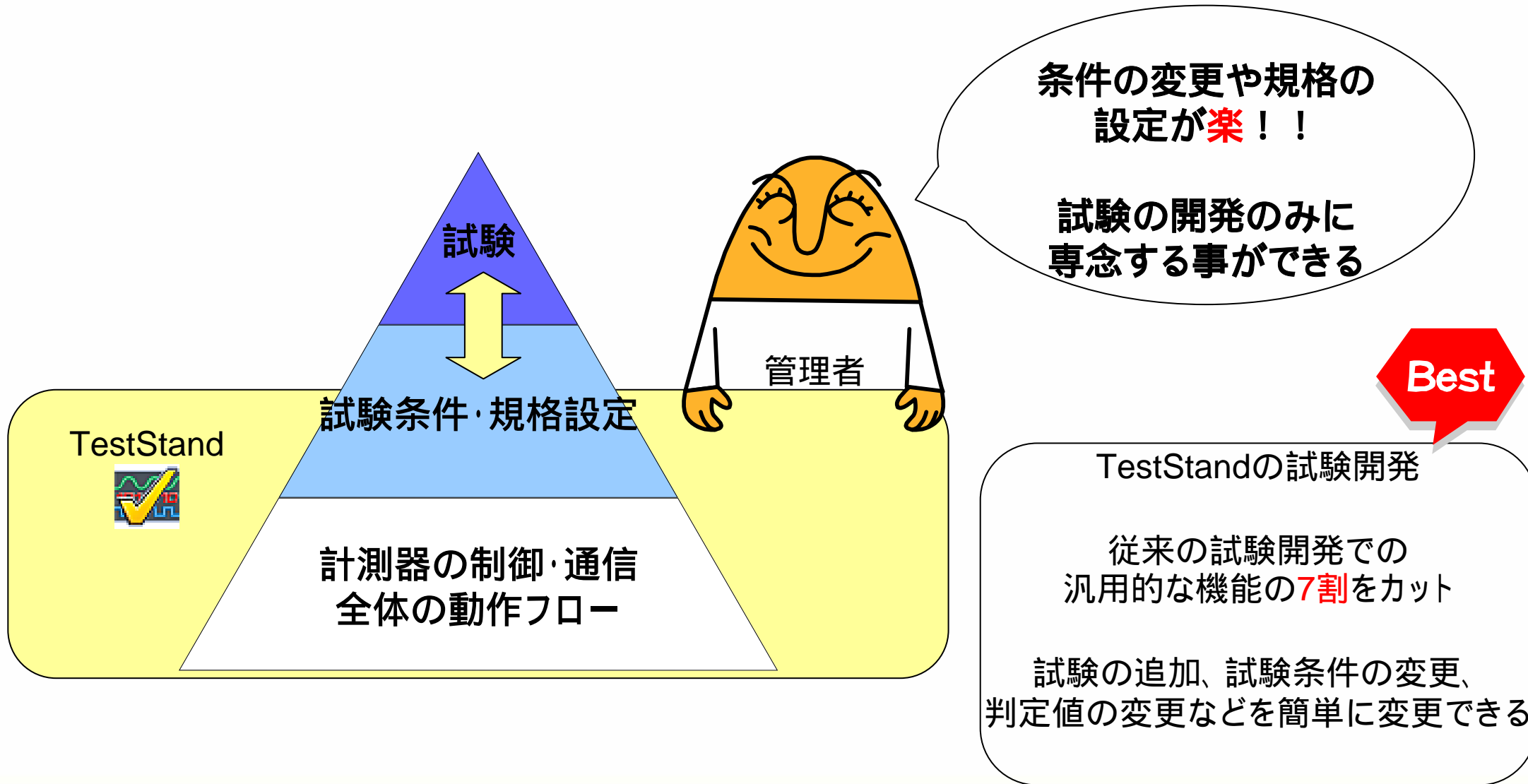




# 従来の開発評価試験



# 試験の作成はTestStandで実現

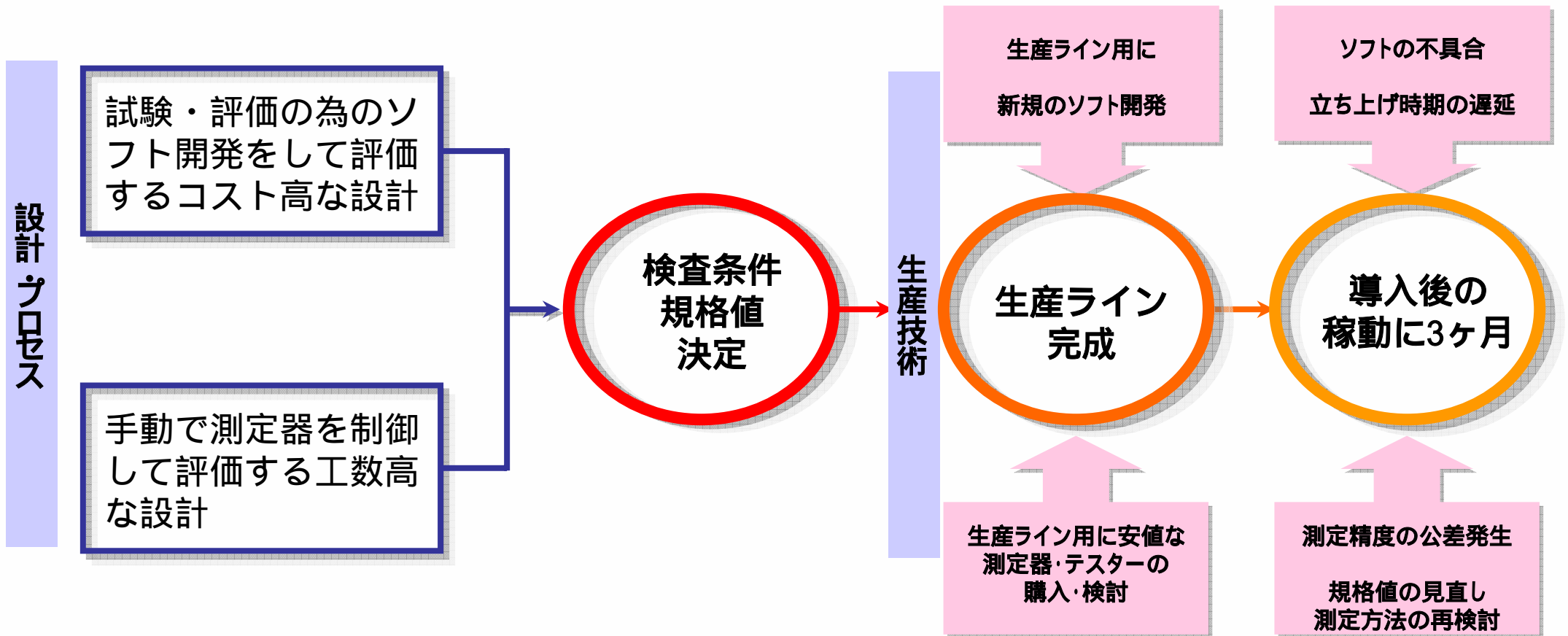


# テストスタンドの有効活用で実現できる Case

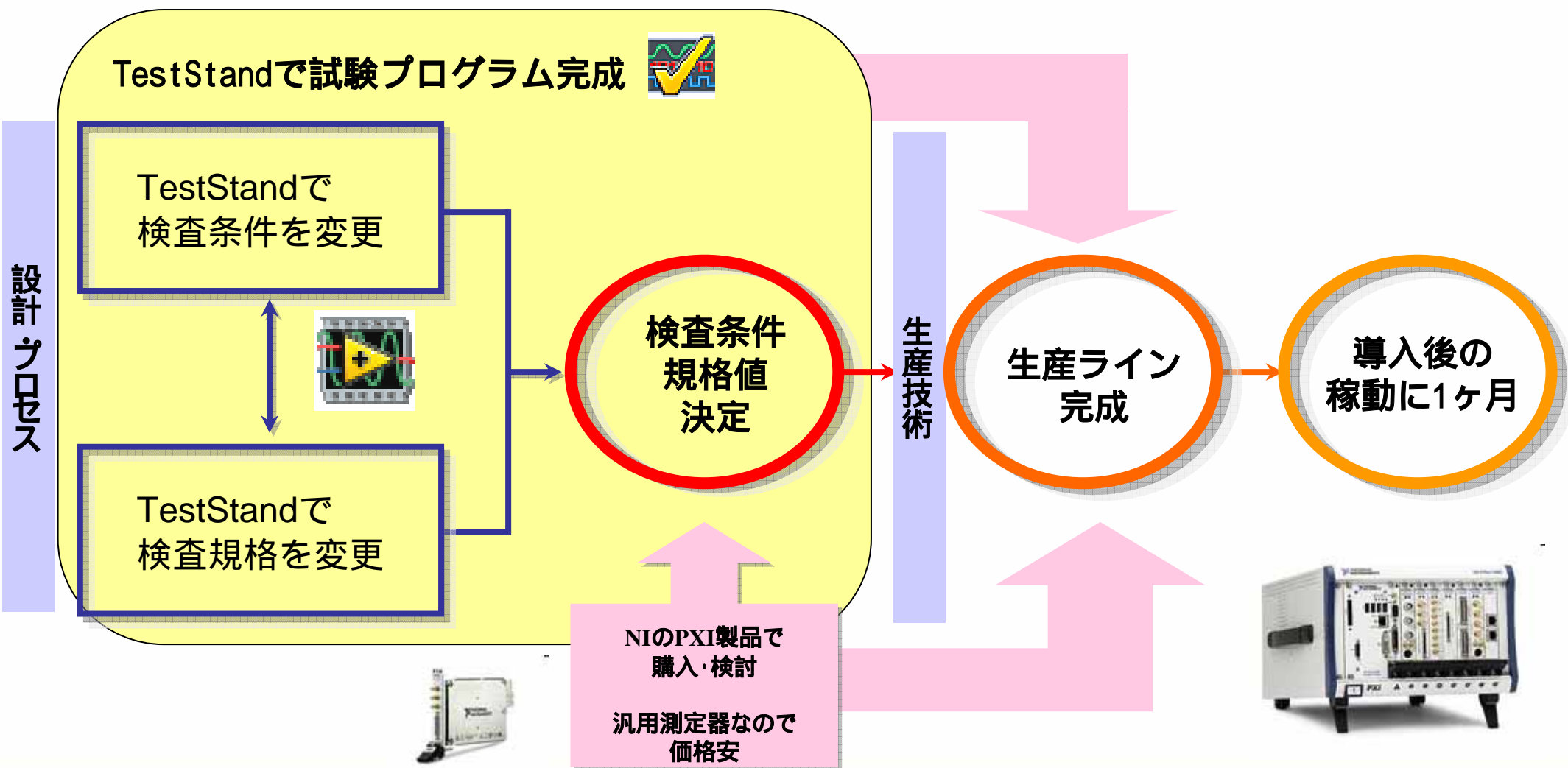
評価から運用まで幅広くカバーする  
Test Standの汎用システム



# 従来の試験開発のプロセス



# TestStandを使用した試験開発のプロセス

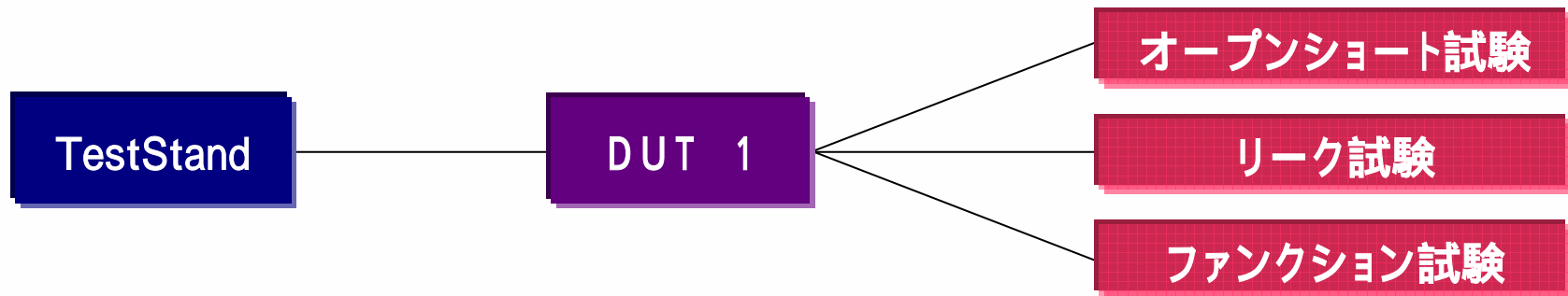


# テストスタンドの有効活用で実現できる Case

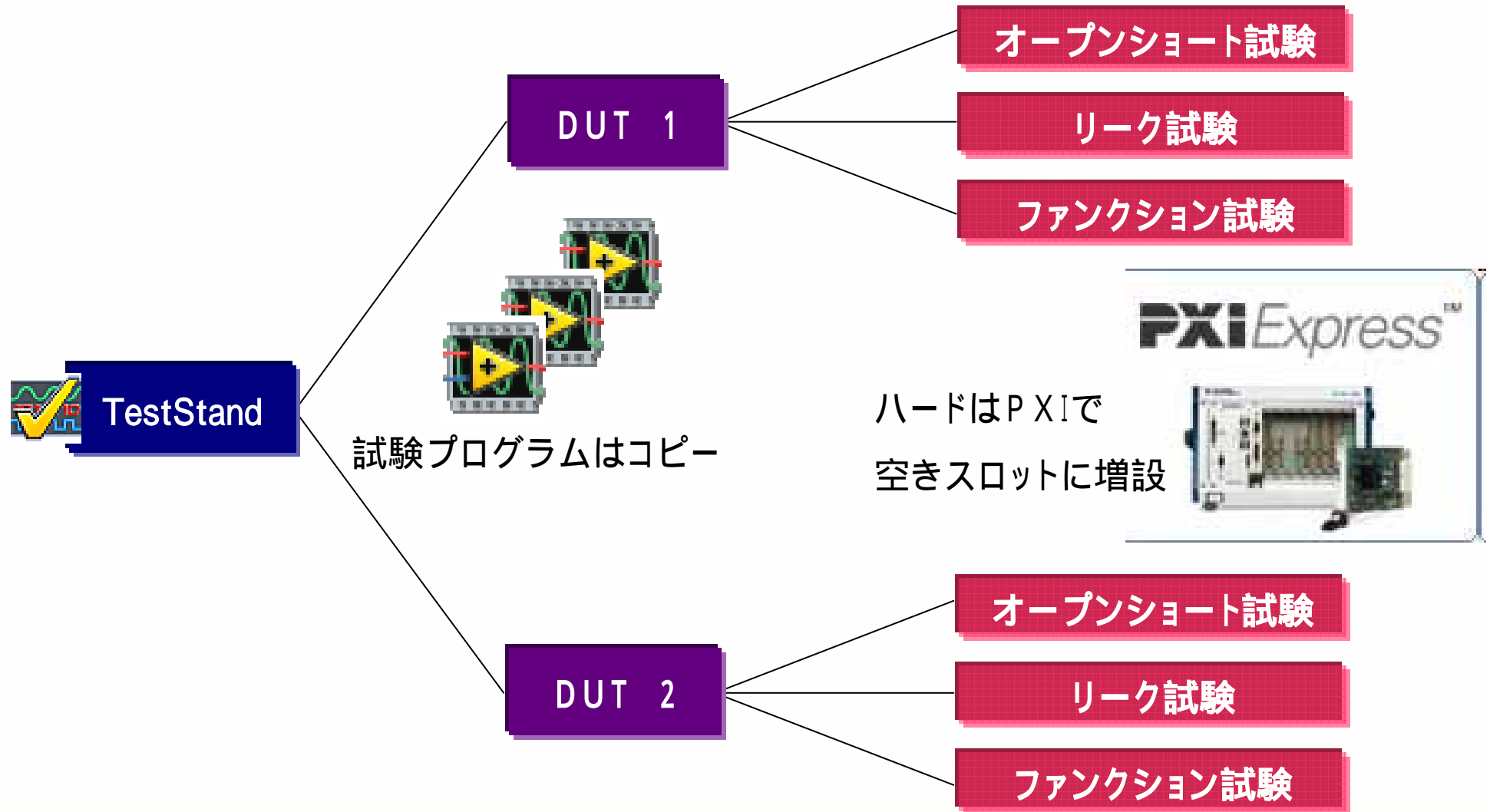
## 同じ試験を並列で実行可能にする 同測システムのご紹介



## 通常のシングル測定の場合



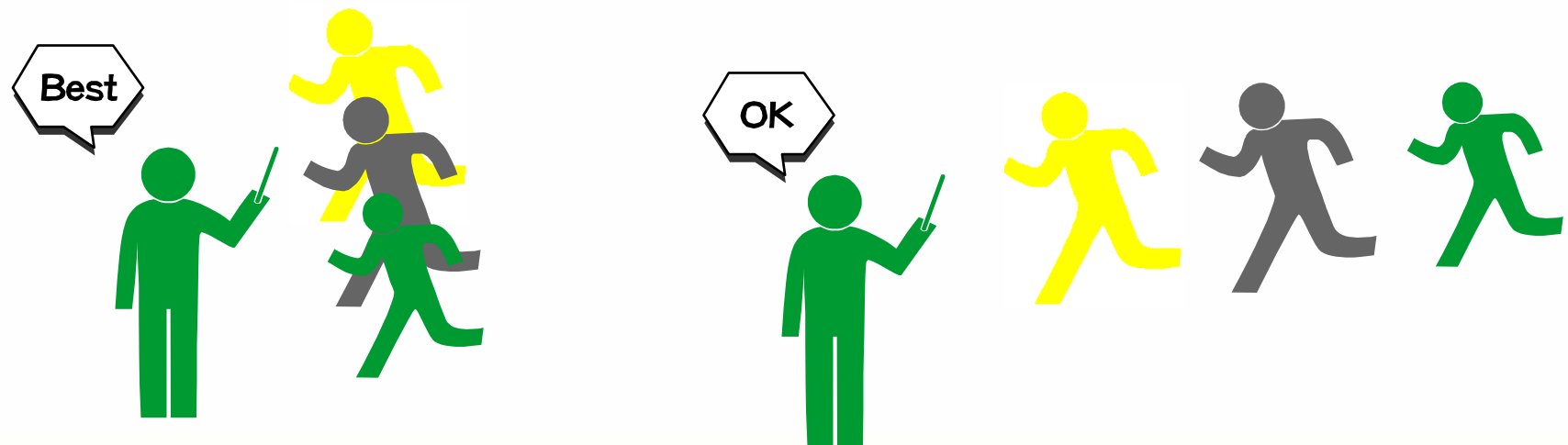
# 複数のターゲット測定の場合





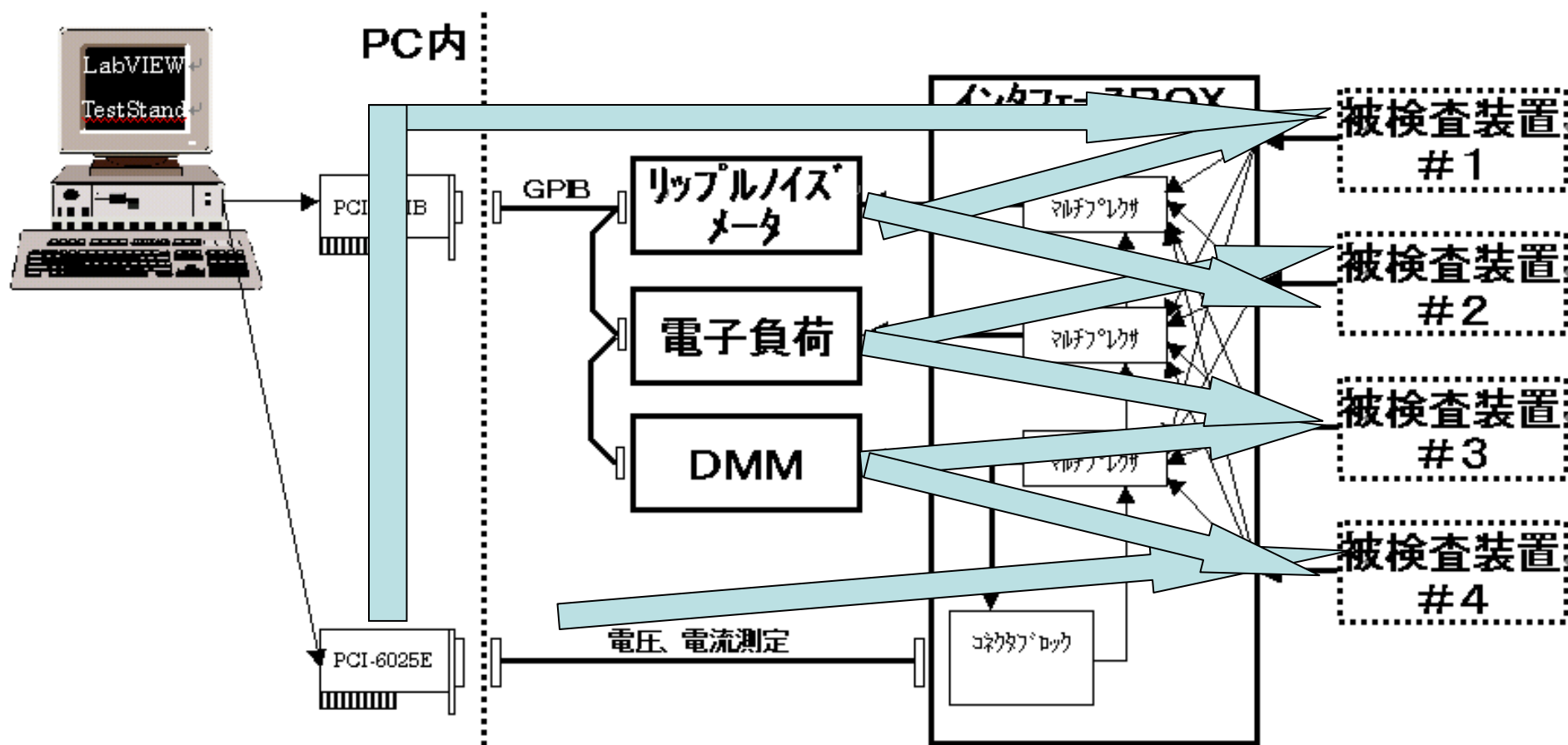
# テストスタンドの有効活用で実現できる Case

## 別の試験同士を同期させる 並列実行システムのご紹介



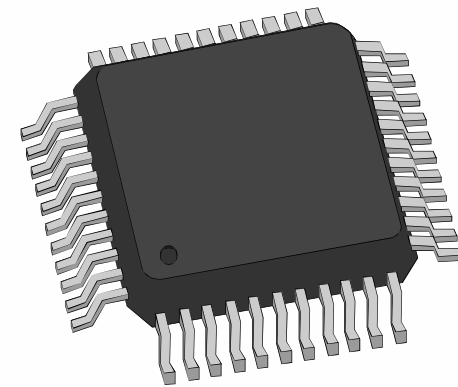
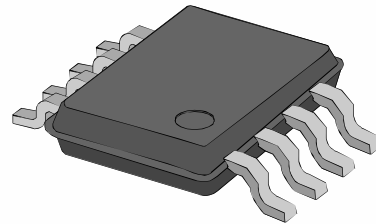
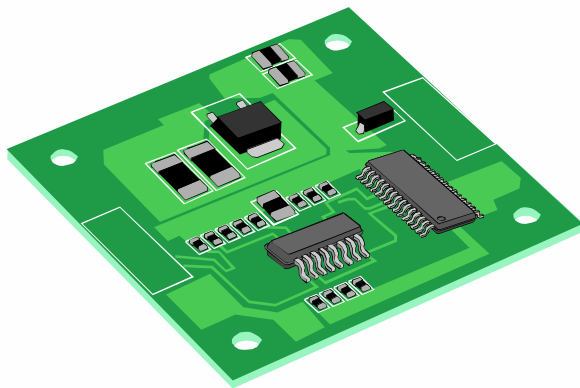
# 並列処理でタクトタイムが4分の1に！！

## 汎用電源検査システム ブロック図



# テストスタンドの有効活用で実現できる Case

## 多品種にも簡単に対応可能な TestStandの開発方法のご紹介



# TestStandでの汎用ソフトの考え方

