



Fault Detection and Diagnosis

実装不良の検出と診断

Dr Bob Storey,
Principal Software Engineer
XJTAG Ltd

Degree in electronics, Ph.D in electrical engineering, from Cambridge University.

6 years embedded software experience in network equipment industry
7 years at XJTAG – responsible for design and development of automatic connection test, logic device representation.

Fault Detection and Diagnosis

実装不良の検出と診断

- Fault Detection vs Fault Diagnosis
実装不良の“検出”と“診断”の違いについて
- Detecting Faults
実装不良の検出
- Fault diagnosis – limitations
実装不良の診断- 制限は？
- Improving fault diagnosis
実装不良の診断を改善
- Tools to help
支援ツール

Detection vs Diagnosis

検出 と 診断 の違い

- **Detection:** “There is a fault”
検出: “実装不良がある”
- **Diagnosis:** “Net A13 is short-circuited to net A14.”
診断: “ネットA13はA14に対してショート”
- Where possible, make detection fast.
検出はできるだけ迅速に
- Diagnosis is only required if a fault exists.
診断が必要になるのは実装不良が見つかったときのみ

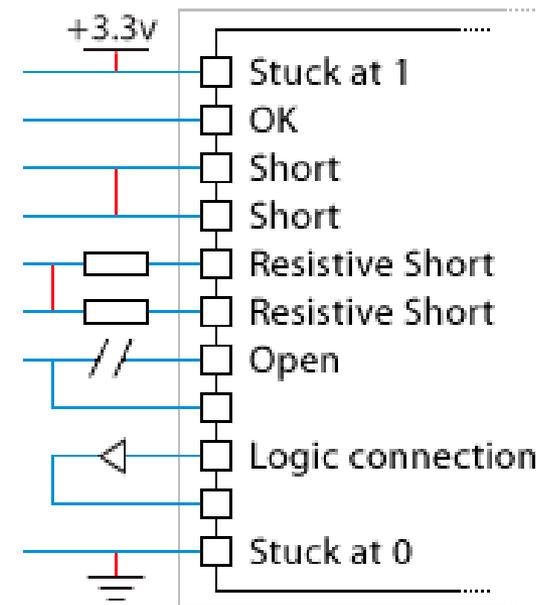
Detecting Faults

実装不良の検出

- It is usually easy to understand whether a fault can be detected.

実装不良検出の可否は判断しやすい

- Can the net be observed?
そのネットは観測できる？
- Can the net be driven?
そのネットはドライブできる？
- Writing a test may not be so easy:
テストはそう簡単ではない:
 - Speed 速度
 - Coverage カバレッジ
(例:PCOLA-SOQ)



Presence Correctness Orientation Live Alignment
Shorts Opens Quality

Fault diagnosis 実装不良の診断

- Ideal situation
理想的には



Error: Short found between nets: IO13, IO14.

- [+ Net Detail](#)
- [+ Error Detail](#)

CONNTEST failed

ネットIO13とIO14 間でショート

- ...but commonly we find:
でも通常は、、、

Error: Unexpected value on logic net A14A-A14.

- [+ Net Detail](#)
- [+ Error Detail](#)

Error: Unexpected value on logic net LOGIC.

- [+ Net Detail](#)
- [+ Error Detail](#)

Error: Unexpected value on logic net A13.

- [+ Net Detail](#)
- [+ Error Detail](#)

Error: Unexpected value on logic net NANDGATE.

- [+ Net Detail](#)
- [+ Error Detail](#)

[View all faults in the layout viewer](#)

CONNTEST failed

A14A-A14、LOGIC、A13、
NANDGATE に不良検出

Diagnosis – limitations

診断-制約について

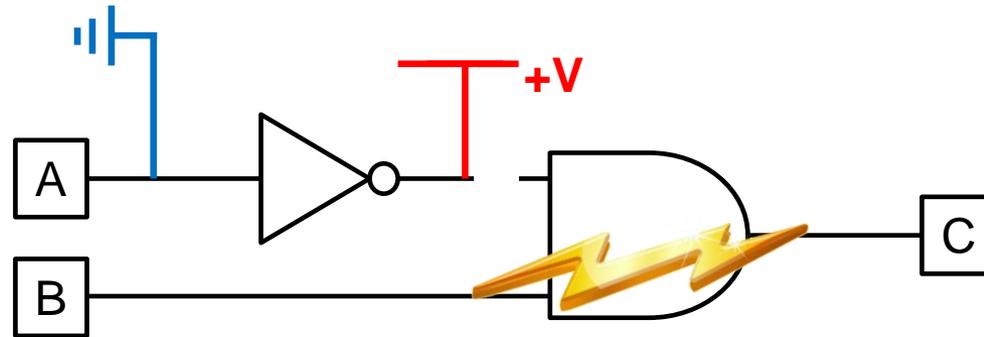
Why do tools not give us a simple diagnosis?
なぜツールでシンプルな診断が提供されないのか？

- Diagnosis can be difficult
診断は難しい場合がある
- Sometimes precise diagnosis is impossible
状況によっては正確な診断は不可能
 - But we may be able to make suggestions as to the likely fault
しかしながら、それなりの示唆を得ることはできる



Diagnosis example: Logic gates

診断例：ロジックゲート

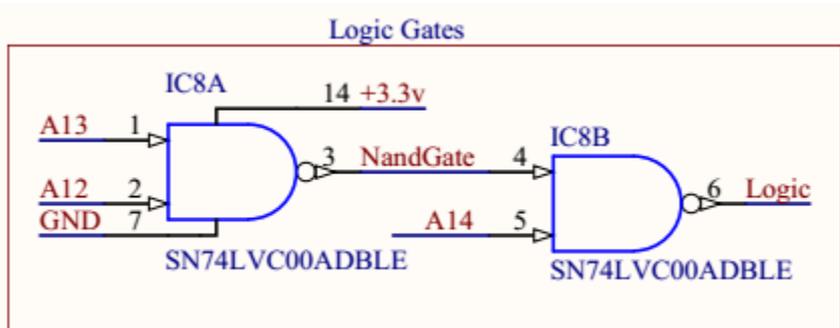


	A	B	C
	L	L	0
	L	H	1
	H	L	0
	H	H	1

- There is definitely a fault here
明らかな実装不良の例
- How many possible faults would give this symptom?
どれだけの実装不良がこの症状を示すでしょうか？

XJTAG with logic gate example

XJTAG でのロジックゲート検査例



Error: Unexpected value on logic net A14A-A14.

- + [Net Detail](#)
- + [Error Detail](#)

Error: Unexpected value on logic net LOGIC.

- + [Net Detail](#)
- + [Error Detail](#)

Error: Unexpected value on logic net A13.

- + [Net Detail](#)
- + [Error Detail](#)

Error: Unexpected value on logic net NANDGATE.

- + [Net Detail](#)
- + [Error Detail](#)

[View all faults in the layout viewer](#)

CONNTEST failed

A14A-A14、LOGIC、A13、
NANDGATE に不良検出

- Add Short Circuit
A13→A14 をショートさせると
- 1 error causes 4 reports
1エラーで4つのレポート
- Layout viewer gives clues
Layout viewerで手掛かりを得る
- XJAnalyser can increase confidence in diagnosis
XJAnalyserで診断に確信を得る
- In the end, an oscilloscope will confirm the short
そしてオシロでショートを確認

XJTAG Fault detection: Connection Test

XJTAG 実装不良の検出: コネクションテスト

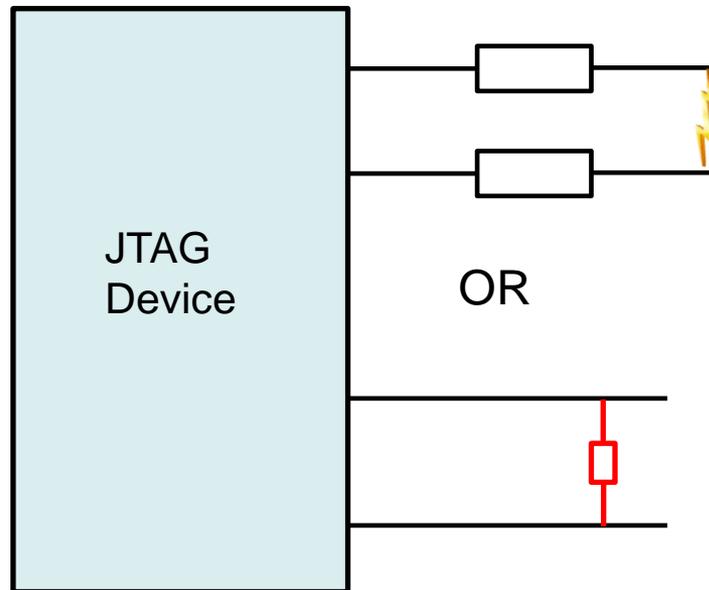
- To detect faults, each net 実装不良を検出するために各ネットは
 - must be driven to both values 2つの値にドライブできなければならない
 - Must be “differently” driven to *each* other net at least once 少なくとも1回は互いに異なる値にドライブできなければならない

Test step	Net A	Net B	Net C	Net D
1	0	-	0	-
	1	-	1	-
2	0	0	-	-
	1	1	-	-
3	-	0	-	0
	-	1	-	1
4	-	-	0	0
	-	-	1	1

- Can do this by driving opposite values 0 / 1にドライブしても良いが
- Preferable to switch between driving / not driving
ドライブする / しないを切替できることが望ましい
 - And add exceptions for buffer pins, etc バッファ等々の例外あり
- This scales $O(\log_2 N)$ のオーダー

Why not drive opposite values? なぜ 0 / 1 のドライブでは不十分なのか？

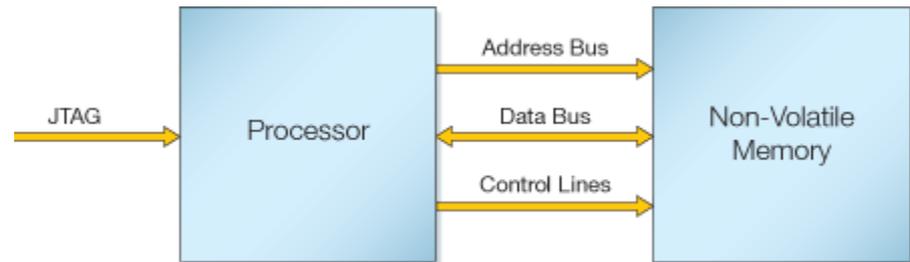
- Resistive fault detection 抵抗性短絡の検出



XJTAG Fault detection: Memory Test

XJTAG による実装不良検出:メモリテスト

- Need to test:
 - Data bus
 - Address bus
 - Control lines



データ、アドレスバス、制御信号をテストする必要がある

- Optimise for speed 速度の最適化
 - Test data and address busses at same time
テストデータとアドレスバスを同時に

Data bus test values

データバスのテスト値

- Example: 16-bit data bus 16ビットデータバスの場合
 - 5 Values required: 5種類のテスト値が必要:

```
0101 0101 0101 0101
0011 0011 0011 0011
0000 1111 0000 1111
0000 0000 1111 1111
1010 1010 1010 1010
```

- Add extra unique values to get to [width of Address bus] data values, then test DATA and ADDRESS at same time
さらにアドレスバスのビット数相当のテスト値を追加すれば
データバスと同時にアドレスバスのテストにもなる

Improving Fault Diagnosis

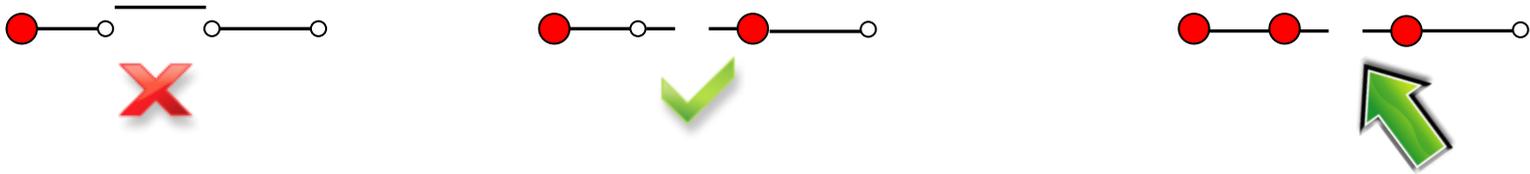
実装不良の診断を改善する方法

- If devices are JTAG-compliant, wire up the JTAG pins
JTAG 準拠のデバイスならJTAG信号を配線する
- Use JTAG devices where possible
できるだけJTAG準拠のデバイスを使用する
- Connect all I/O pins on JTAG devices
JTAG デバイスの全てのI/O ピンを接続する
 - If the pin is not used functionally it can still be used for test
機能的に使用されない信号でもテストに活用できる
- Consider open-circuit fault detection/diagnosis
オープンエラーの検出と診断を考慮に入れる

Improving Fault Diagnosis (Opens)

実装不良の診断を改善する(オープンエラー)

- Open-circuit fault diagnosis: オープンエラーの診断



To **detect** an open-circuit error we need: オープンエラー検出には

- two readable JTAG-device pins
2つのJTAGデバイス信号(読み可能な)
- or 1 JTAG-device pin and some other way to check the value at another point on the net
あるいは1つのJTAGデバイス信号と、そのネットの反対側の値をチェックする何らかの仕組みが必要

BUT – to **diagnose** where the open circuit error occurs we need more access than this.

どこにオープンエラーがあるかを突き止めるにはさらにアクセスが必要

Improving Fault Diagnosis

実装不良の診断を改善する方法

- If devices are JTAG-compliant, connect the JTAG pins
JTAG 準拠のデバイスならJTAG信号を配線する
- Use JTAG devices where possible
できるだけJTAG準拠のデバイスを使用する
- Connect all I/O pins on JTAG devices
JTAG デバイスの全てのI/Oピンを接続する
 - If the pin is not used functionally it can still be used for test
機能的に使用されない信号でもテストに活用できる
- Consider open-circuit fault detection/diagnosis
オープンエラーの検出と診断を考慮に入れる
- Use JTAG as well as test points/oscilloscope/DVM
JTAG に合わせてテストポイント、オシロ、デジボルなど活用する

Use tools to identify improvements

- XJTAG Design-for-test guide (DFTガイドライン)

http://www.fuji-setsu.co.jp/files/xjtag_dft-guidelines_jp.pdf

- DFT analysis in XJDeveloper DFT解析機能
- Test point reports テストポイント支援機能
- 30-day free trial of XJTAG available
30日間の無償評価版あります



JTAG Testing with Programmable Devices プログラマブルなデバイスとJTAGテスト

Rob Humphrey,
Principal Hardware Engineer
XJTAG Ltd

MEng Degree in electronic engineering from The University of York, UK.

8 years experience in circuit and IC design, development and testing.

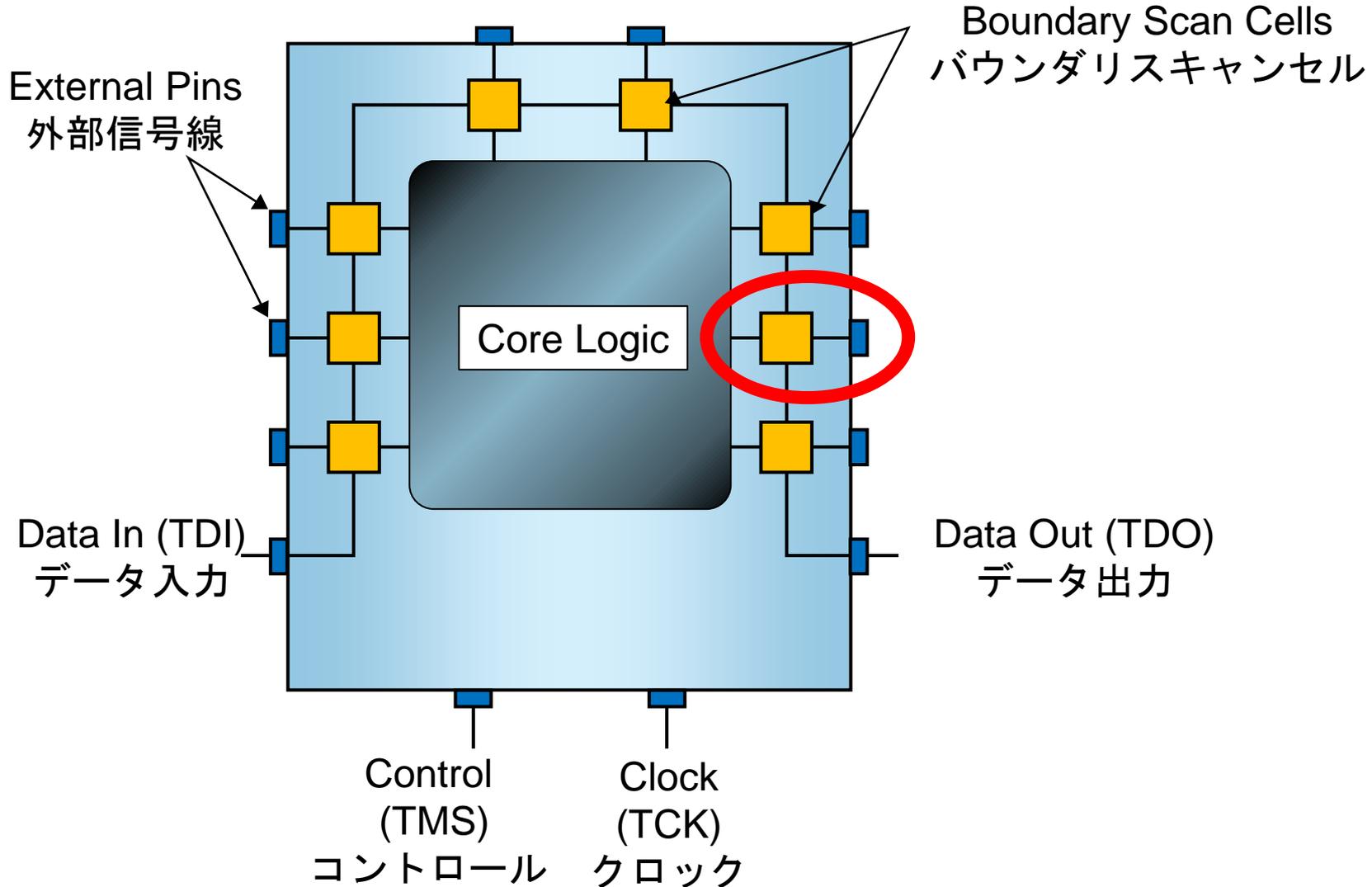
5 years at XJTAG – responsible for hardware development of all XJTAG's products and final level technical support.

Programmable Devices

- Very common 一般的:
 - CPLD (Complex Programmable Logic Device)
 - FPGA (Field Programmable Gate Array)
 - Microprocessors
- Important to include in JTAG chain
JTAGチェーン内に含めることが重要
 - Large pin count 多数の信号線
 - Significant test coverage テストカバレッジ改善に極めて有効
- Do not always behave as expected!
期待通りに動作しないこともある
- Can be used for more than just Boundary Scan
バウンダリスキャン以外の使い道も

Boundary Scan Overview

バウンダリスキャンの概要



Single I/O Pad with Boundary Scan バウンダリスキャン 単一のI/Oパッド

From previous pad
直前のパッドから

Boundary Scan Cells
バウンダリスキャンセル

Output Buffer
出力バッファ

External Pin
外部信号線

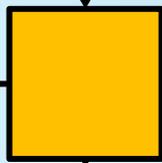
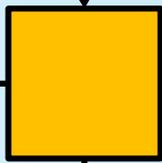
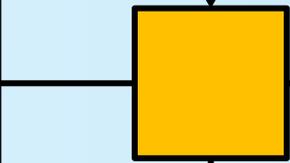
Input Buffer
入力バッファ

To next pad
次のパッドへ

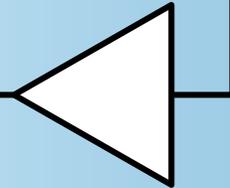
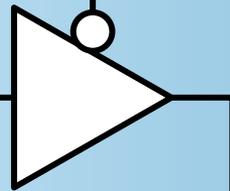
Core Logic
コアロジック

I/O Pad with External Pull-Down 外部プルダウンされるI/Oパッド

From previous pad
直前のパッドから



To next pad
次のパッドへ

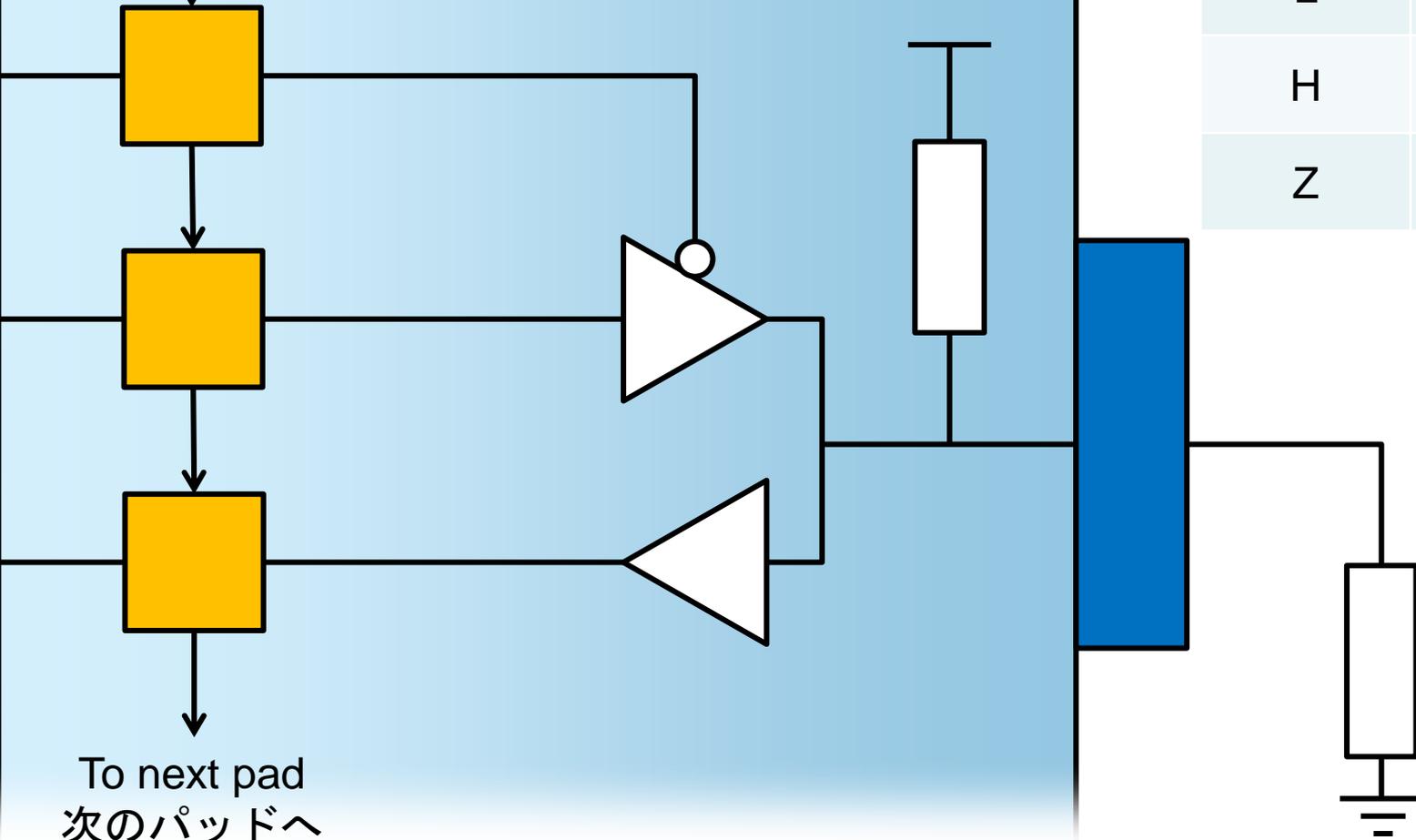


Write	Read
L	0
H	1
Z	0

I/O Pad with Internal Pull-Up Enabled

内部プルアップされるI/Oパッド

From previous pad
直前のパッドから



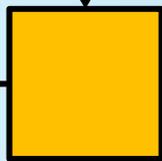
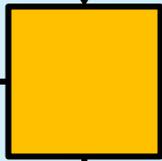
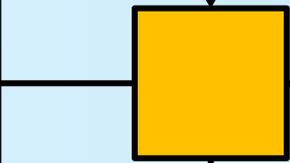
Write	Read
L	0
H	1
Z	1

To next pad
次のパッドへ

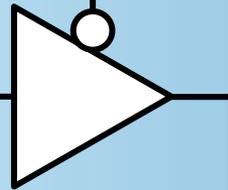
I/O Pad Configured as Output Only

出力のみに設定されるI/Oパッド

From previous pad
直前のパッドから



To next pad
次のパッドへ



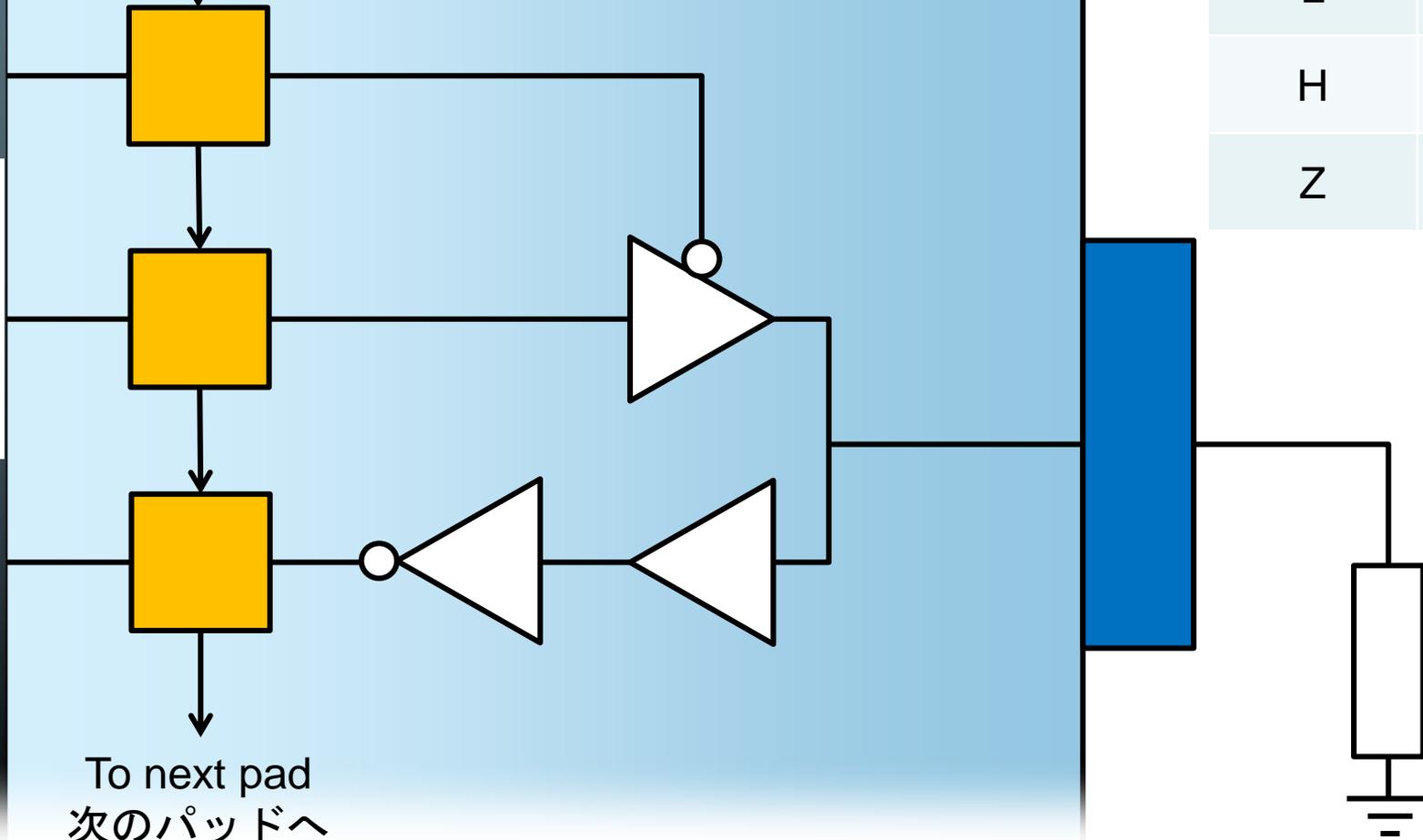
Write	Read
L	0
H	0
Z	0



I/O Pad with Input Inverted 入力が反転されるI/Oパッド

Write	Read
L	1
H	0
Z	1

From previous pad
直前のパッドから

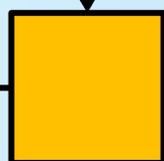
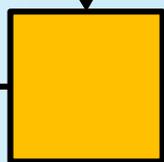
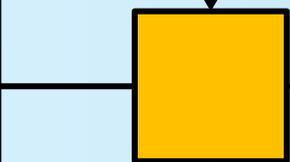


To next pad
次のパッドへ

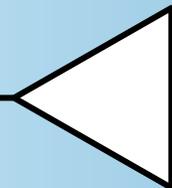
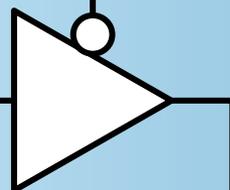
I/O Pad with Internal Pull-Down Enabled

内部プルダウンがイネーブルされるI/Oパッド

From previous pad
直前のパッドから

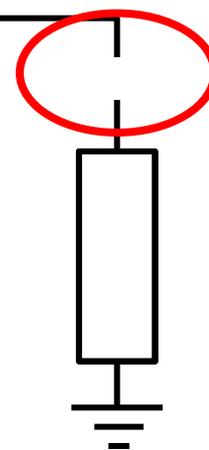


To next pad
次のパッドへ



Write	Read
L	0
H	1
Z	0

Not Detected
検出されない



When might this happen?

これら課題はいつ起こり得るのか？

- Microprocessors マイクロプロセッサの場合
 - After functional code execution
パッドを設定するコードが実行されると
- CPLD の場合
 - Post-configuration コンフィギュレーションされると
- FPGA の場合
 - Post-configuration コンフィギュレーションされると
 - Pre-configuration with some FPGAs!
コンフィギュレーションされる前でも、、、

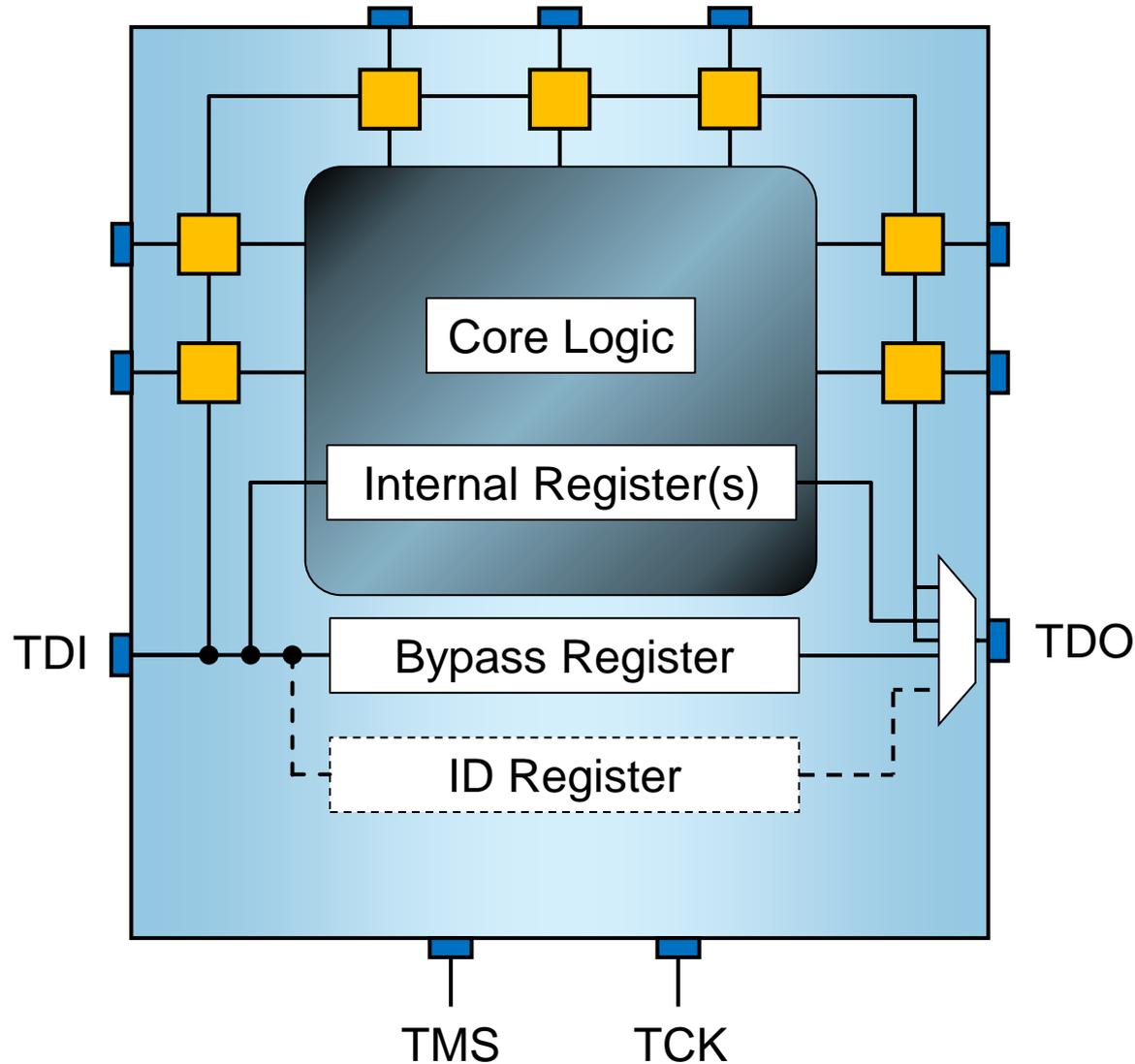
How can this be fixed?

これら課題への対処法は？

- Microprocessors マイクロプロセッサの場合
 - Hold soft-reset and pulse hard-reset
ソフトリセットをホールドしてハードリセットをかける
 - Reconfigure pads via JTAG
JTAG を介してパッドを再設定
- CPLD の場合
 - Erase CPLD 消去する
 - ~~Use a post-configuration BSDL file 設定後のBSDLファイルを使う~~
- FPGA の場合
 - Clear FPGA (and PROM if necessary) 消去する（必要ならPROMも）
 - Configure with custom 'blank' image 専用の設定ファイルを用いる
 - ~~Use a post-configuration BSDL file 設定後のBSDLファイルを使う~~

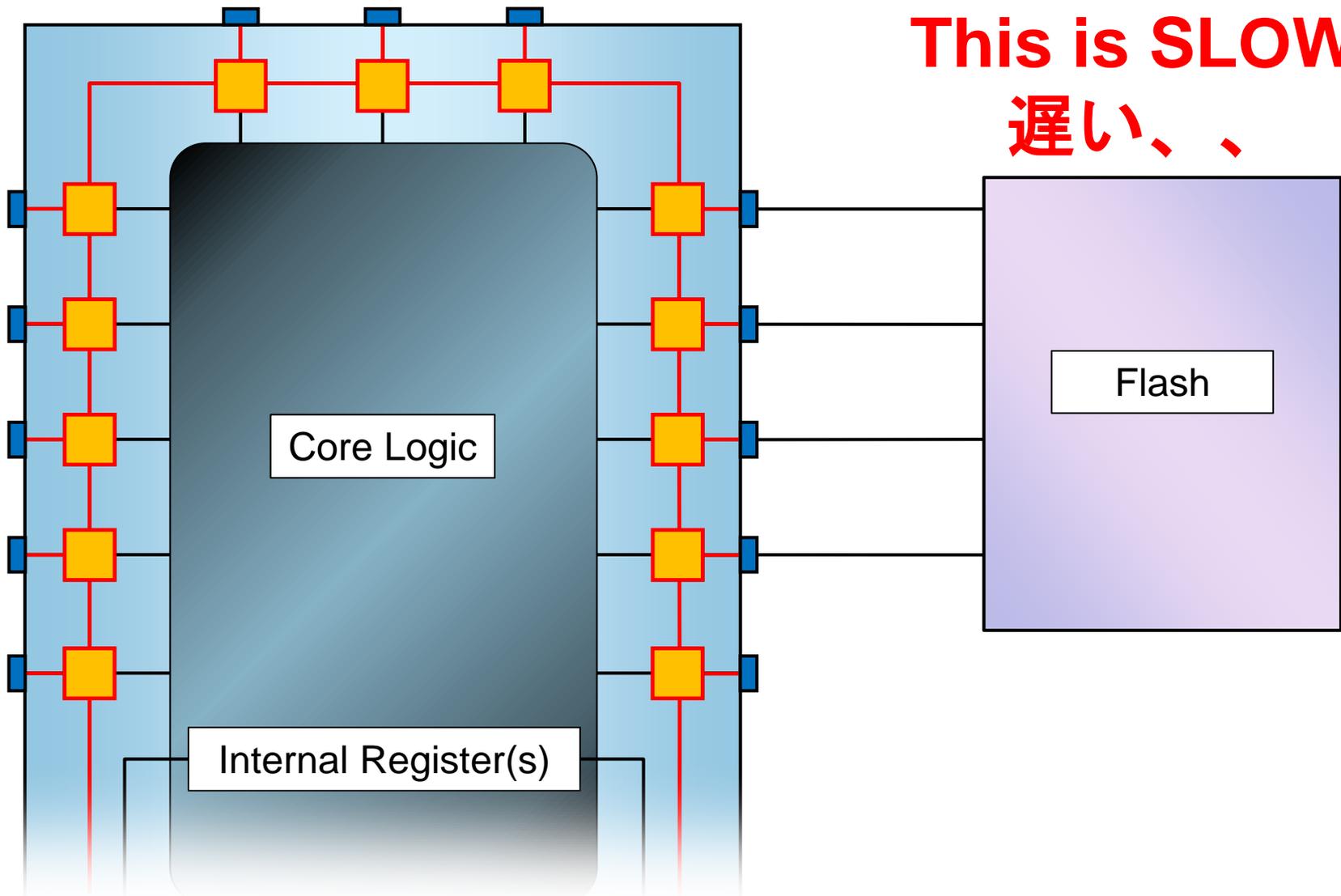
What else can JTAG be used for?

JTAG 他の活用は？



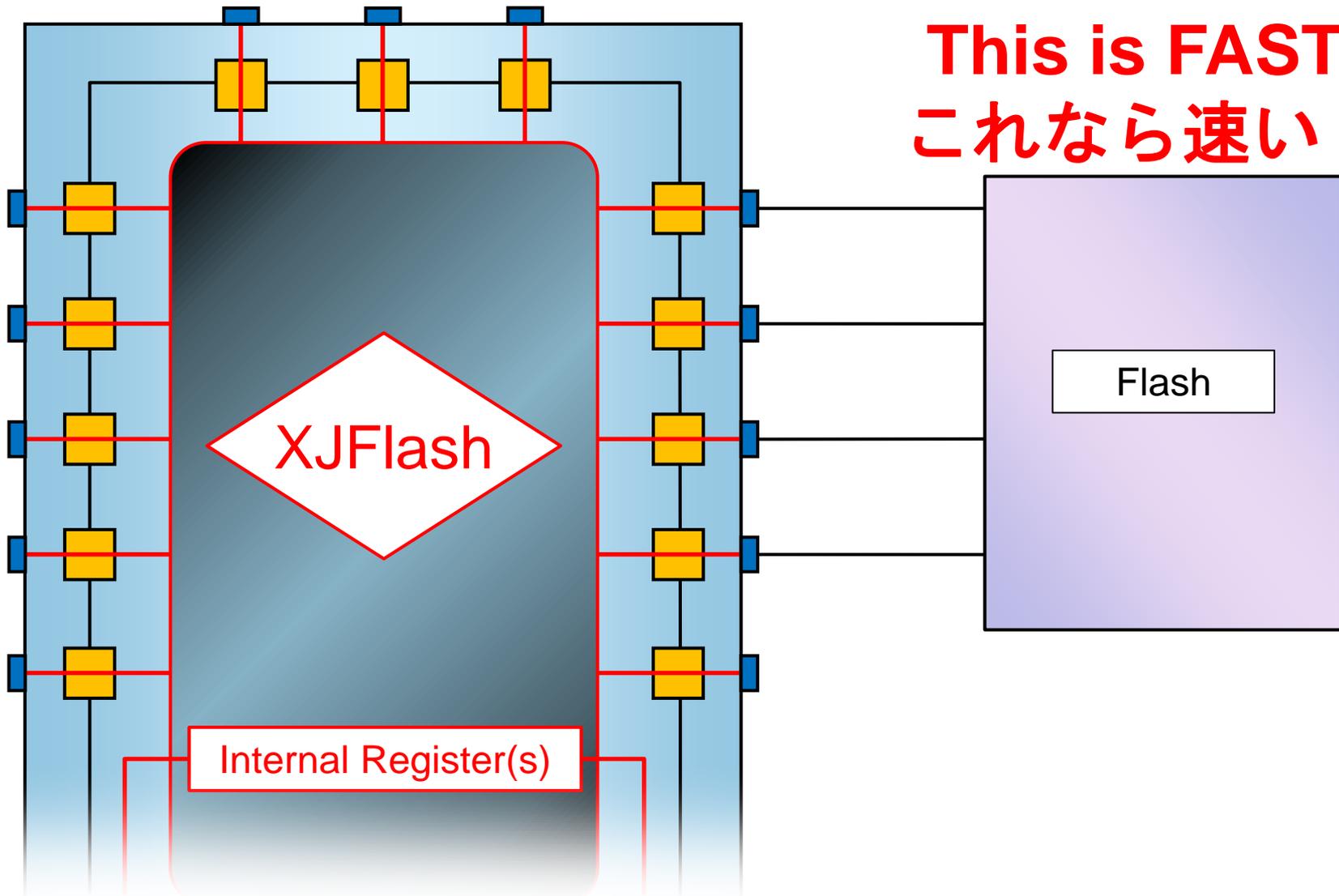
Flash Programming via Boundary Scan

バウンダリスキャン経由のFlashプログラミング



Flash Programming via XJFlash

XJFlash を介してFlash プログラミング



This is FAST!
これなら速い!

Make the most of Programmable Logic with XJTAG プログラマブルなデバイスと上手に付き合うには

- Always make JTAG TAP signals accessible
JTAG のTAP信号へアクセスできるようにする
- Clear/Blank a device if it is already configured
デバイスの設定はクリアする(ブランクに)
- Ensure internal pull resistors will not hide board faults
内部プルレジスタで実装不良が隠されないように留意する
- Use for fast in-circuit-programming of flash
デバイスプログラミングを活用して高速Flash 書込み
- JTAG is versatile and can be used for even more than this
(eg on-chip system monitoring/embedded instrumentation)
JTAG は上述以外にも多目的に活用できる

XJTAG バウンダリスキャンテスタ

電子基板の実装検査
不良解析を省力化

BGAなど高密度実装を
電氣的に検査



30日間 無償で お貸出しします

XJTAG はデバイスごとのテストライブラリを再利用して、テストを簡単に構成できるので、評価も容易に行えます。

もし基板の情報をお預かりできれば、無償でセットアップもいたします。



St John's Innovation Centre, Cowley
Road, Cambridge, CB4 0DS, UK
www.xjtag.com

富士設備工業株式会社 電子機器事業部
<http://www.fuji-setsu.co.jp>