

BST(バウンダリスキャンテスト)  
及び  
ISP(インシステムプログラミング)  
の実際

---

資料番号 BS - 02 - 0001 (第2版)  
発行年月 2008年10月

---

株式会社ニューリー・土山 滋賀県甲賀郡土山町北土山979  
TEL : 0748-66-1681 FAX : 0748-66-0915 E-mail : [nltmail@newly-t.com](mailto:nltmail@newly-t.com) <http://newly-t.com>

# 1. バウンダリスキャンの概要

## 1.1 バウンダリスキャンの背景

今日のエレクトロニクス業界の技術的動向として、表面実装 (SMT)、デバイスの小型化、マルチチップ・モジュール、複雑な ASIC などの先端技術の進展があげられます。これらの技術によって新しい設計および製造方法の改善が図られますが、一方で従来のテスト方法では対応できない状況も発生して来ています。

現在のインサーキット・テスト、ファンクション・テストはノード・アクセスの問題によって全ノードをカバーできないという状況が起こってきています。また実装基板の回路が複雑になり、テストプログラムの開発に時間がかかりテストのためのコストは増大するが、製品化は遅れるといったことが起っています。

これらの問題に対処するため、1985年に数社のヨーロッパ企業が JETAG (Joint European Test Action Group) と呼ばれるグループを結成し (後に TI、HP などのアメリカ企業の参加により JTAG (Joint Test Action Group) と改称される)、バウンダリスキャン技術を提案しました。この提案は IEEE の作業グループによって詳細な検討および整理が行われ、1990年に IEEE Standard 1149.1-1990 “IEEE Standard test Access Port and Boundary-Scan Architecture” として規格化されました。

この規格はデジタルデバイスのテストの容易化をはかりテストビリティ向上のため、デジタルデバイスにどのようにセルを組み込むかを定義したものです。

## 1.2 なぜバウンダリスキャンか

実装基板製造ラインの最後で期待されることは、実装基板が仕様を満たすことです。仕様を満たすためには、

- (A) すべての部品がそれぞれの仕様を満足する
- (B) すべての部品が正しい場所に実装されている
- (C) すべての部品のピンが基板に正しくハンダ付けされている
- (D) すべての部品の複合動作が仕様通りである

ことが必要です。受け入れ検査で (A) を確認し、インサーキット・テストで (B)、(C) を確認し、ファンクション・テストまたはシステム・テストで (D) を確認するといったことが必要になります。しかしこれには次のような問題点があります。

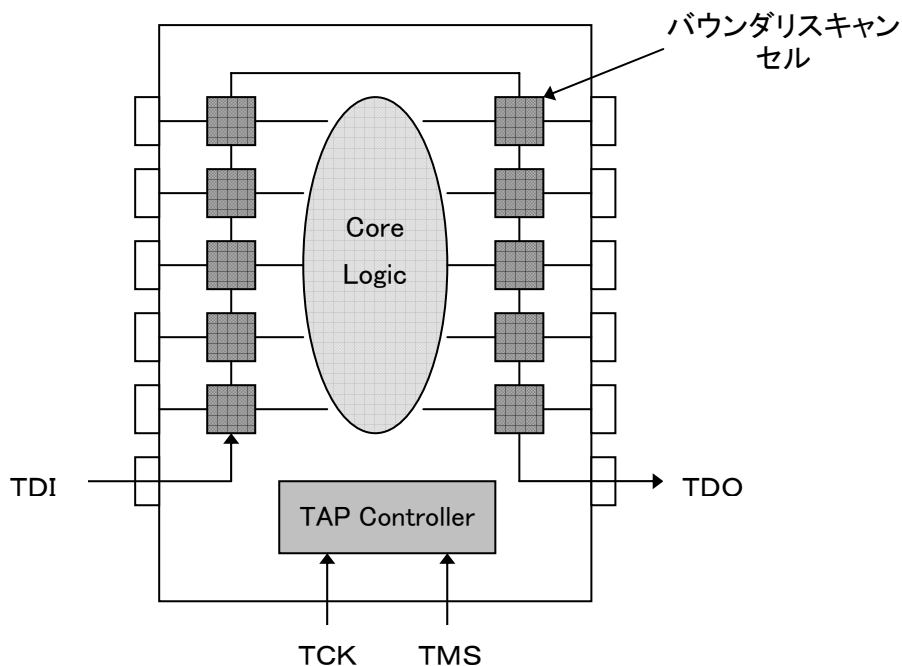
- 1) 複雑な IC および ASIC のテスト開発には時間がかかり、製品化の次期を遅らせる
- 2) アクセスできるノード数に制限があり、インサーキット・テストのテストカバレッジが落ちる

バウンダリスキャンテストは設計の複雑化およびノードアクセスの問題を解決できるデバイス組み込み設計を取り入れたテストビリティ向上設計 (DF T) の一つです。バウンダリスキャンは主に (B)、(C) の対応を目的としますが (A) にも対応できます。最近、アナログ回路を含んだテストにも対応できる標準規格が制定されています (IEEE Std1149.4) 。

## 2. バウンダリスキャンテスト

### 2.1 バウンダリスキャンテスト

バウンダリスキャンテストはどのようなデバイスでも実行できるものではありません。デバイスを製造するときにあらかじめある仕掛けをデバイスに組み込んでおく必要があります。それはデバイスピンとデバイス本来の機能を持っているデバイスの Core Logic との間にセルとよばれるシフトレジスタを配置しておくこととその他のレジスタ及びそれらをコントロールする TAP コントローラと呼ばれるステートマシンを組み込んでおくことです。このようなデバイスをバウンダリスキャン対応デバイスとよびます。



バウンダリスキャンテストを行うためには、テスト対象の基板上にバウンダリスキャン対応デバイスが搭載されていることが前提となります。ただしすべてのデバイスがバウンダリスキャン対応デバイスである必要はありませんが、配線テストを行う場合、バウンダリスキャンテストは一部を除いてバウンダリスキャン対応デバイス間をテストするので、バウンダリスキャン対応デバイス同士の接続が多ければ多いほどテストカバレッジが上がります。

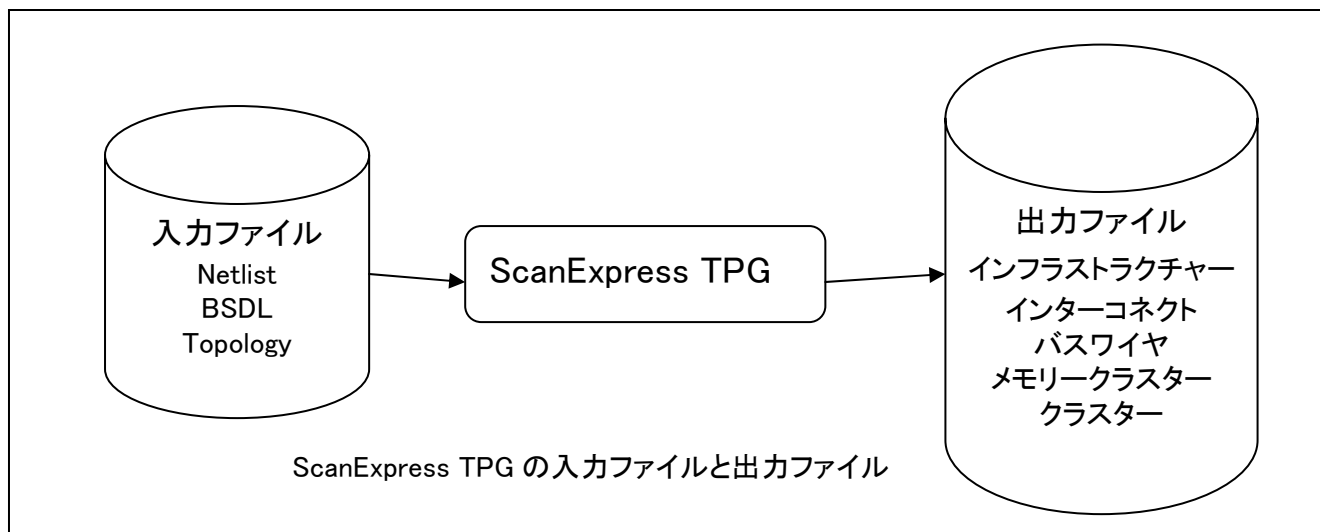
### 2.2 Corelis の ScanExpress によるバウンダリスキャンテスト

ScanExpress でバウンダリスキャンテストを実行するには次のハードウェアとソフトウェアが必要です。

- 1) ハードウェア
  - a) パーソナルコンピュータ  
デスクトップまたはノート
  - b) バウンダリスキャン・コントローラ  
PCI-1149.1、PCMCIA-1149.1/E、USB-1149.1/E、NetUSB-1149.1/E

- 2) ソフトウェア
  - a) ScanExpress TPG (Test Program Generator)
  - b) ScanExpress Runner
  - c) ScanExpress ADO (Advanced Diagnostics Option)

ScanExpress TPG はターゲットとするデバイスまたはボードに応じてバウンダリスキャンテストのテストベクターを作成します。



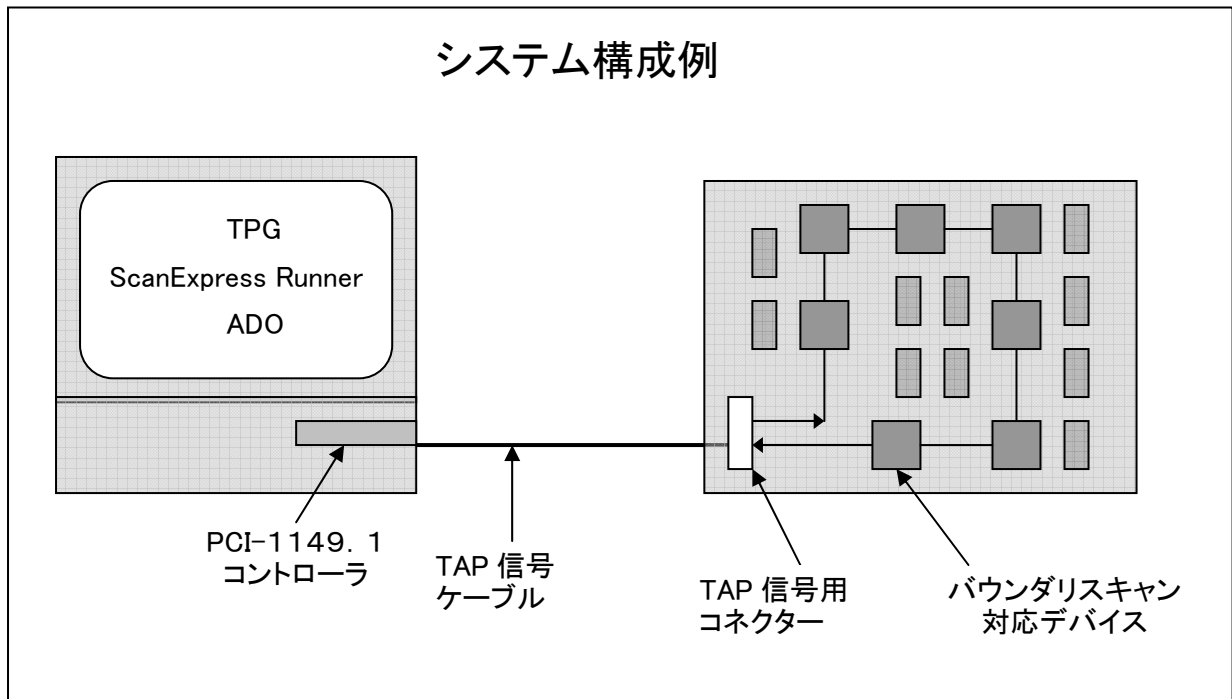
ScanExpress TPG で作成されたテストベクターを ScanExpress Runner でバウンダリスキャンテストを実行します。バウンダリスキャンテストを実行し FAIL したとき、ScanExpress ADO で不良解析します。

ScanPlus で実行されるバウンダリスキャンテストは次の通りです

- 1) Infrastructure Test
- 2) Interconnect Test
- 3) Buswire Test
- 4) Memory Cluster Test
- 5) Cluster Test

これらのテストの詳細については次ページ以降で紹介します。

## システム構成例



ScanExpress Debugger は上図の状態ではバウンダリスキャン対応デバイス間のある特定のネット、あるいはバスの接続状態をマニュアルで操作してテストするソフトウェアです。

### 2.3 ScanExpress による ISP (In System Programming)

バウンダリスキャンテストではありませんが、ScanExpress Runner はバウンダリスキャン対応の FPGA や CPLD を On Board でプログラミングできます。また、Flash Memory もボード上でプログラムできます。(Flash Memory のアドレスバス、データバス、コントロールバスすべてがバウンダリスキャン対応のデバイスに接続されている必要があります。)

#### 2.3.1 FPGA、CPLD の場合

デバイスメーカーから提供されたツールで生成された JAM file または SVF(Serial Vector Format) file を使って ScanExpress Runner でプログラムします。

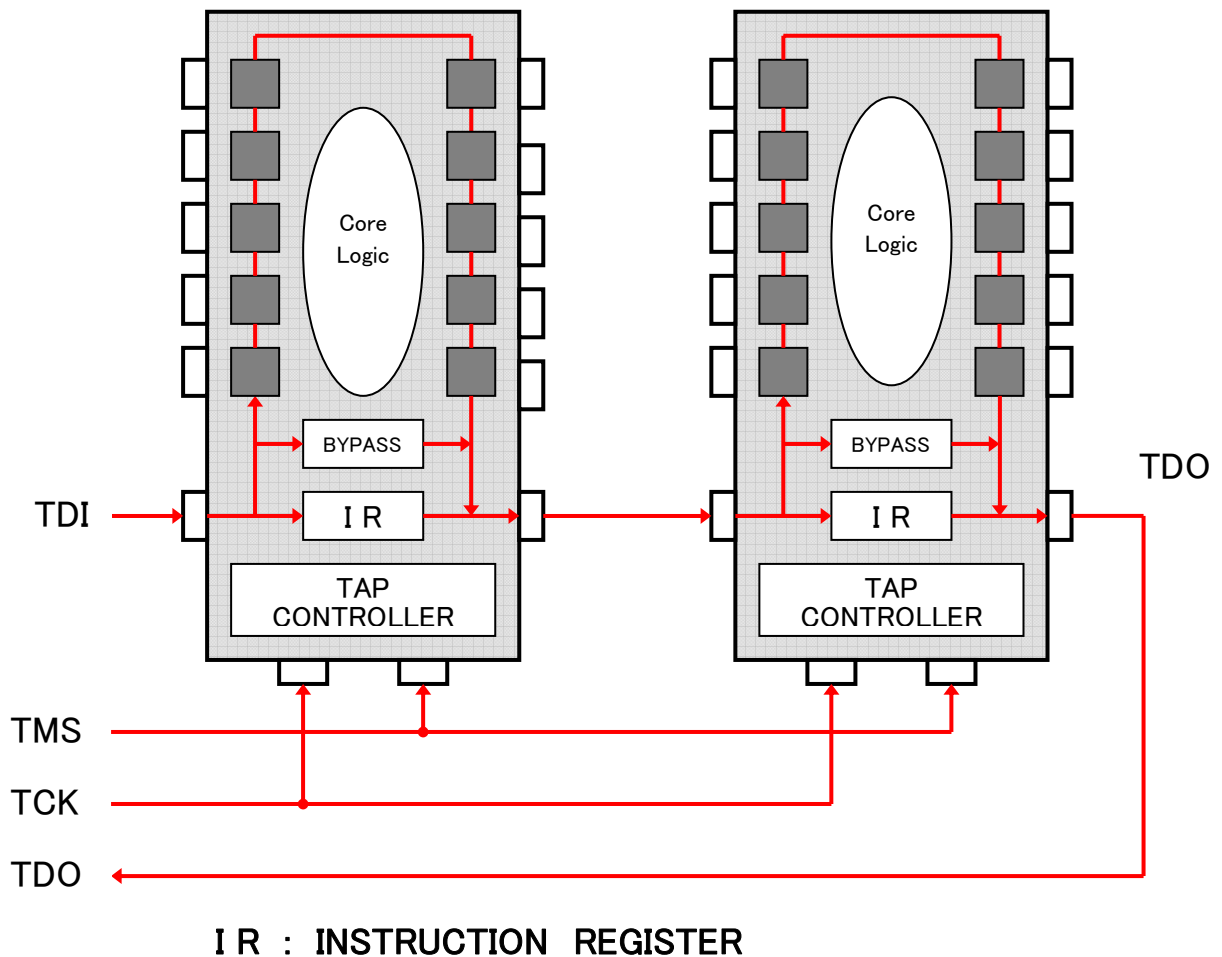
#### 2.3.2 Flash ROM の場合

ScanExpress Runner でプログラミングできますが、プログラミングするための Board file を作成するためには、ScanPlus Generator が必要です。また Flash ROM 専用のプログラミングツール ScanExpress Programmer もございます。

# INFRASTRUCTURE TEST

バウンダリスキャンテストの基本となる信号(TDI、TDO、TMS、TCK、TRST)が正しく接続されていることを確認するテスト

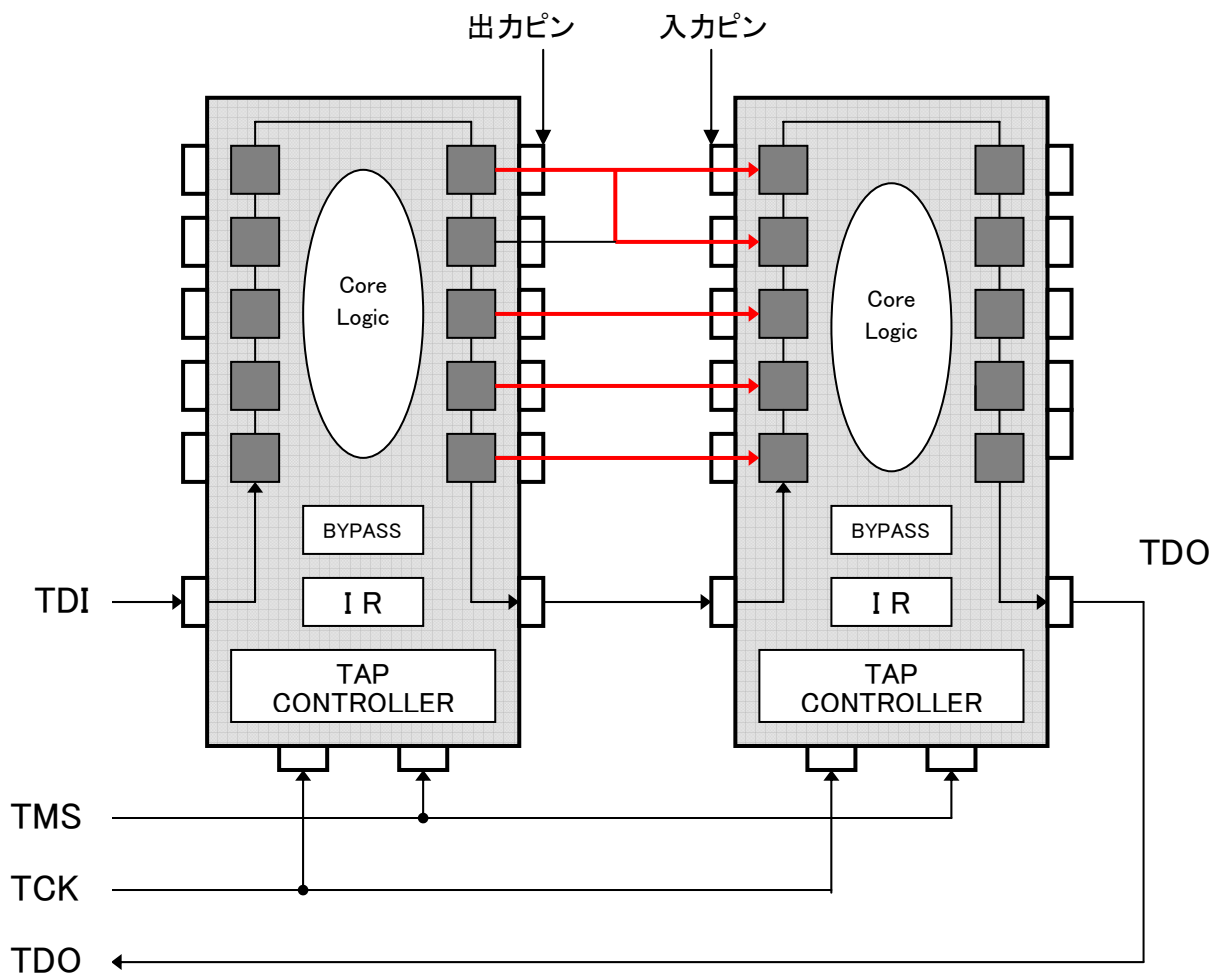
1. BSDL ファイルに書かれている INSTRUCTION\_CAPTURE ビットパターンと、デバイスに実際に書き込まれているビットパターンとを比較します。
2. BSDL ファイルに書かれている ID コードと、デバイスに実際に書き込まれているビットパターンとを比較します。(ID コードを持っているデバイスの場合)
3. BSDL ファイルに書かれている USER コードと、デバイスに実際に書き込まれているビットパターンとを比較します。(USER コードを持っているデバイスの場合)
4. バウンダリスキャンレジスターセルが BSDL ファイルに書かれているとおりかどうかをチェックします。



# INTERCONNECT TEST

バウンダリスキャン対応デバイス同士のピン間が正しく接続されていることを確認するテスト

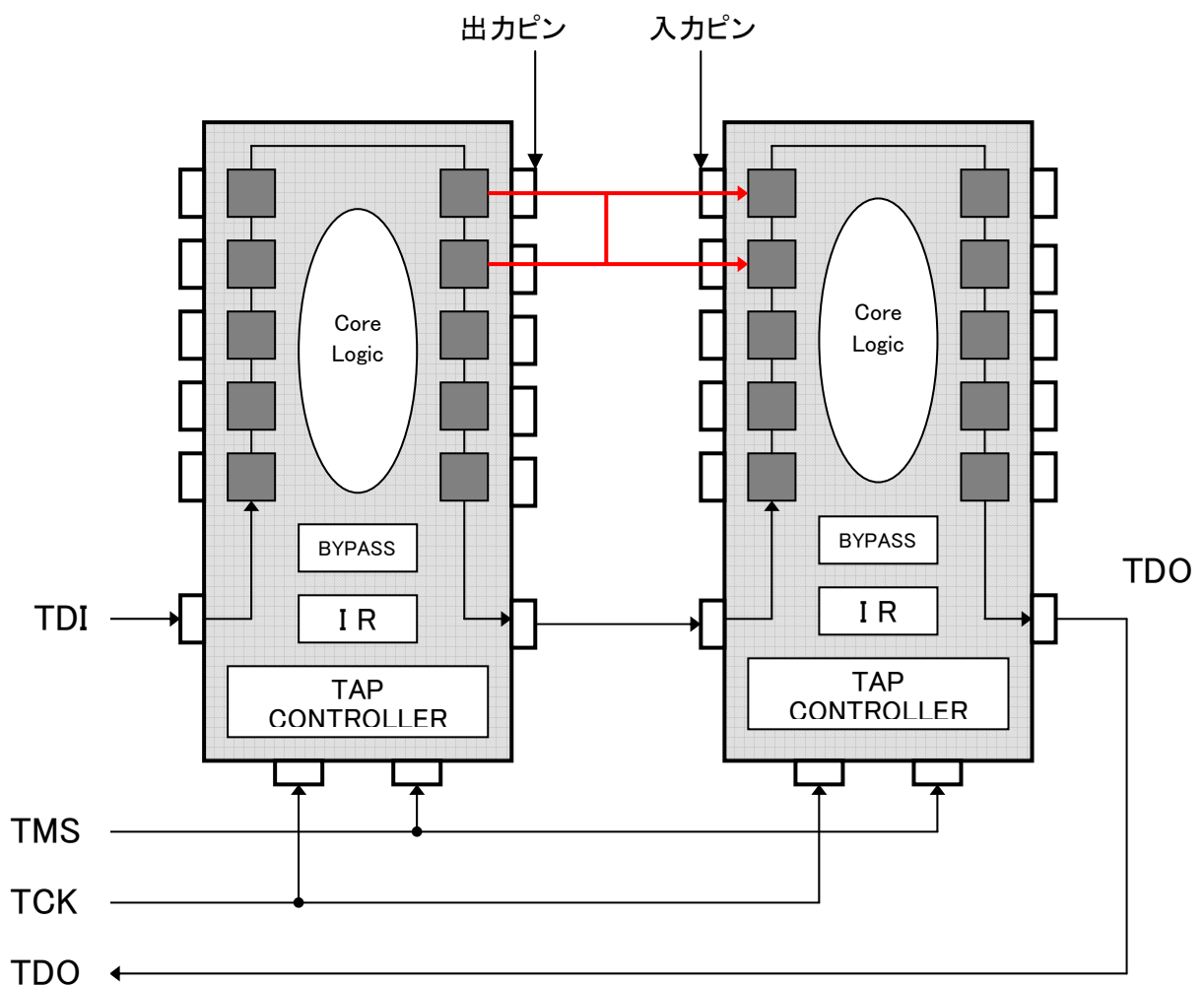
1. 出力ピンからあらかじめ定められたテスト信号を出力し、入力ピンがテスト信号を入力し出力テスト信号と入力された信号を比較します。
2. ピン間のオープンだけでなく VCC、GND へのショート、他のピン、ネットとのブリッジも検出します。
3. 出力ピンが二つ以上あるネットは、一つの出力ピンのみでテストします。  
(ショートによる基板へのダメージをさけるためテスト時間の短縮を図っている。他の出力ピンは BUSWIRE TEST でテストされる。)
4. バウンダリスキャン対応デバイス同士の接続であっても、入力ピン同士、出力ピン同士のネットはテストできません。



# BUSWIRE TEST

出力ピンが二つ以上あるネットのすべての出力ピンが正しく接続されていることを確認するテスト

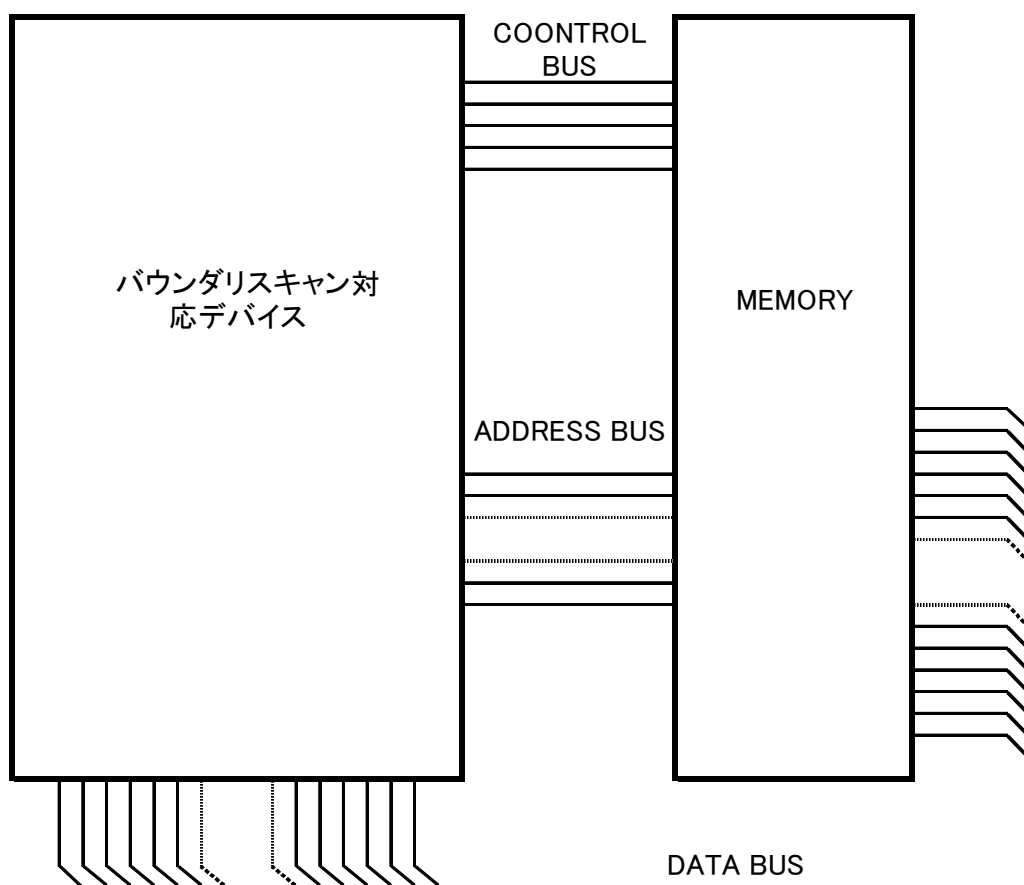
1. BUSWIRE TEST では複数の出力ピンを持つネットのすべての出力ピンをテストします。
2. INTERCONNECT TEST では不良のネットの検出はできますがどちらのピンが不良かは分かりませんが、BUSWIRE TEST では不良のピンまで検出できます。



# MEMORY CLUSTER TEST

DRAM、SRAM、FIFO、SDRAM、SGRAM、SSRAM 等のメモリーとバウンダリースキャンデバイスが正しく接続されていることを確認するテスト

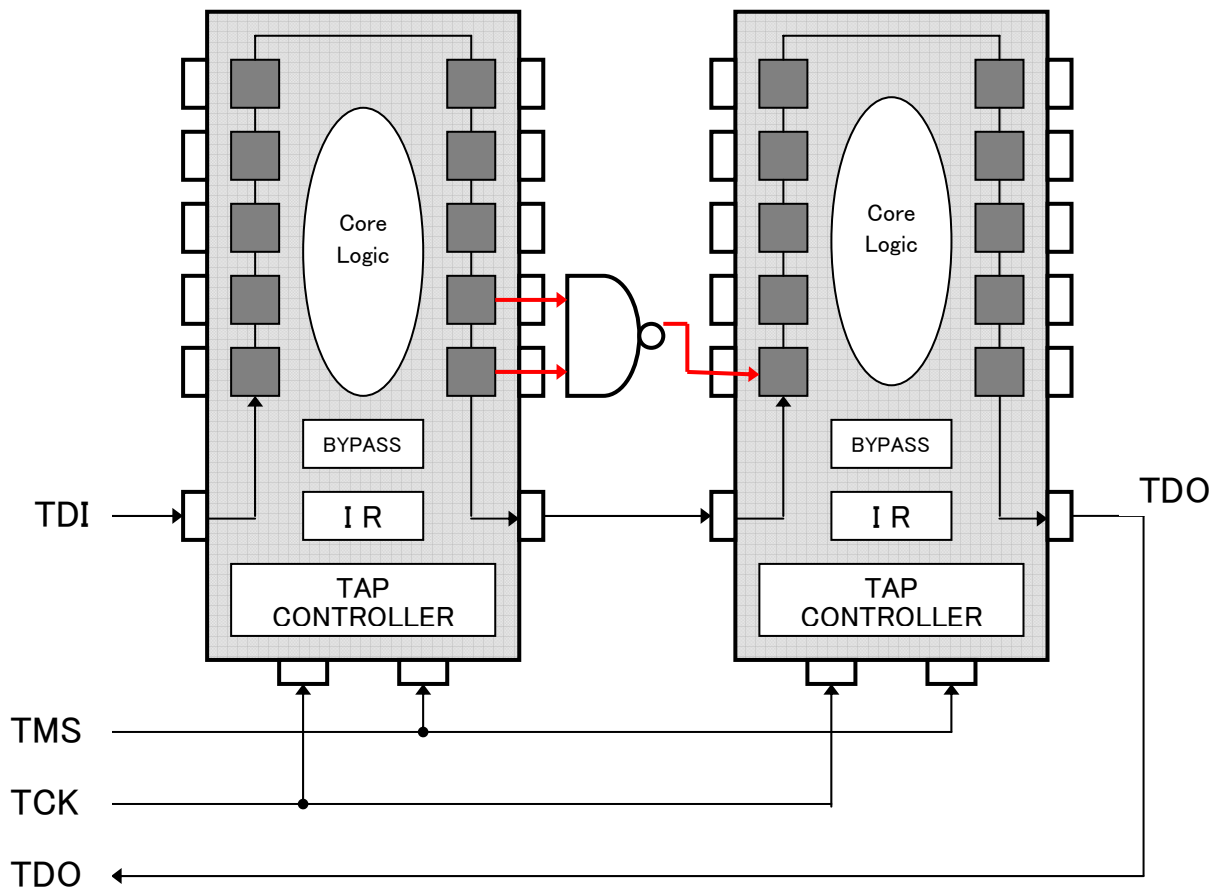
1. 実際にメモリーにデータを書き込み (WRITE) そのデータを読み出して (READ) 比較しアドレスバス、データバス、コントロールバスの接続をテストします。
2. アドレスバス、データバス、コントロールバスのすべてがバウンダリスキャン対応のデバイスに接続されている必要があります  
(SDRAM、SGRAM、SSRAM では CLK もバウンダリスキャンでコントロールする必要があります)



# CLUSTER TEST

バウンダリスキャンデバイスには含まれたバウンダリスキャン未対応のロジックデバイスが正しく接続されていることを確認するテスト

1. バウンダリスキャン未対応のロジックデバイスの真理値表にしたがってテストパターンをバウンダリスキャンレジスターセルから入力し、ロジックデバイスの出力をバウンダリスキャンレジスターセルに取り込み真理値表と比較
2. ロジックデバイスの真理値表が必要





## 3. バウンダリスキャンテストの利点

### 3.1 バウンダリスキャンテストの利点（Corelis の ScanExpress の場合）

- 1) 従来の ICT ではプローブを立てないとテストをできませんでしたが、バウンダリスキャンテストではバウンダリスキャン対応デバイス間はプローブを立てなくてもテストが可能です。このためプローブが削減でき、フィクスチャー治具のコスト削減が図れます。
- 2) 従来のファンクションテストではテストの結果、テスト対象ボードの機能不良は分かっても、不良のノードを特定することは非常に困難です。しかしバウンダリスキャンテストでは不良のノードを特定できます。
- 3) ASIC をボード上でテストするのは難しいものがあります。ASIC は一つ一つの設計が違うためテストのためのライブラリがありません。また ASIC はほとんどの場合、ボードレベルでのテストを考えて設計されていません。しかし ASIC にバウンダリスキャンを組み込んでおけば、簡単に接続テストができます。
- 4) テストプログラムの作成にはプリント基板やデバイスの専門知識を必要としません。PC 上でファイルを選択するだけで作成できるので、専門のエンジニアを配置する必要はありません。ファンクションテストなどに比べ、テストプログラム作成の時間が大幅に短縮できます。そのため新製品の早期立上げが可能になります。またテストプログラムの開発費用が削減できます。
- 5) 充実した不良診断ソフトウェアを使うと、製造工程不良を早期に改善でき、修理工数の大幅な削減が図れます。

ただし、バウンダリスキャンテストは今のところボード全体をテストできるものではありません。電源回路はもちろんのこと、JTAG 未対応のアナログ回路もテストできません。そのためバウンダリスキャンテスト以外の手法を組み合わせることでボード全体のテストカバレッジを上げる必要があります。

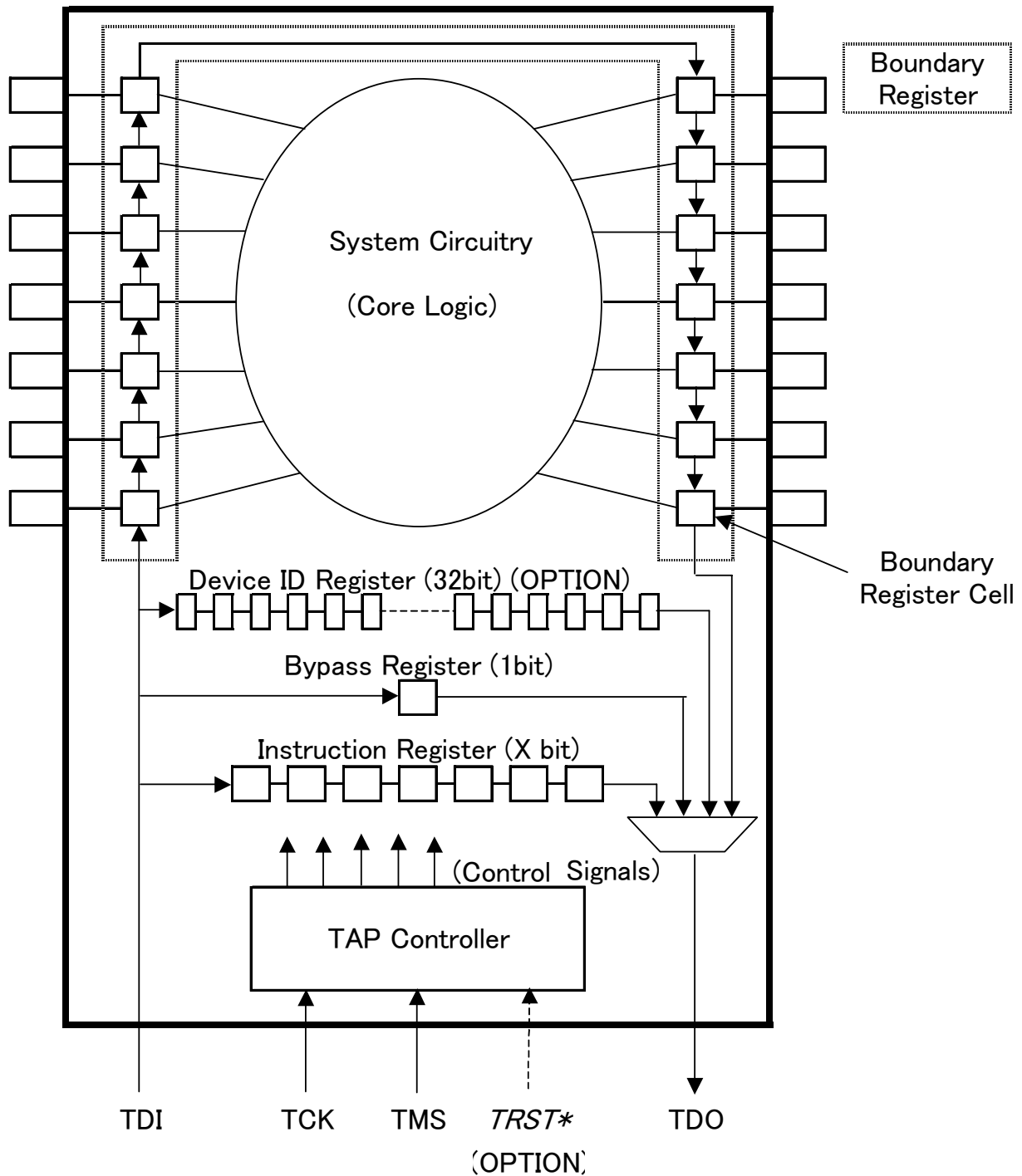
そのためには、新しいボードを設計する段階でボードのテストをどのように実施するかをあらかじめ検討されることをお勧めします。

当社ではバウンダリスキャンテストを新しく採用されるユーザーの方々のために“BST（バウンダリスキャンテスト）及び ISP（インシステムプログラミング）の設計のためのガイドライン”を準備いたしております。ご連絡いただければバウンダリスキャンテストを組み込むための設計から検証までをお手伝いさせていただきます。

## 4. バウンダリスキャンデバイスの構造

### 4.1 バウンダリスキャンデバイスの構造

バウンダリスキャンデバイスの概略構造を下に示します。バウンダリスキャンデバイスにはデバイス本来の機能を持っているコアロジックおよび入出力ピン以外に、TAP (Test Access Prot) と呼ばれるコントローラ、インストラクションレジスタ、データレジスタ (バイパスレジスタ、ID コードレジスタ、バウンダリスキャンレジスタ) および TAP 信号ピンがあります。



## 4.2 TAP 信号

TAP コントローラが使用している TAP 信号には、TCK (Test Clock)、TMS (Test Mode Select)、TDI (Test Data In)、TDO (Test Data Out) およびオプションの TRST (Test Reset) があります。これらの信号をパーソナルコンピュータで制御することによってバウンダリスキャンテストが行えます。

**TCK** : テストロジックのクロックを供給します。TCK は、シリアルテストデータ経路をデバイス固有のシステムクロックから独立して使用できるようにするための専用入力です。

**TMS** : TMS の論理レベルは、TCK の立ち上がりエッジと一致したとき、TAP コントローラの動作ステート間の遷移を生じます。そのステート遷移がさらにステートマシン内のデータおよび TAP 命令の移動を可能にします。

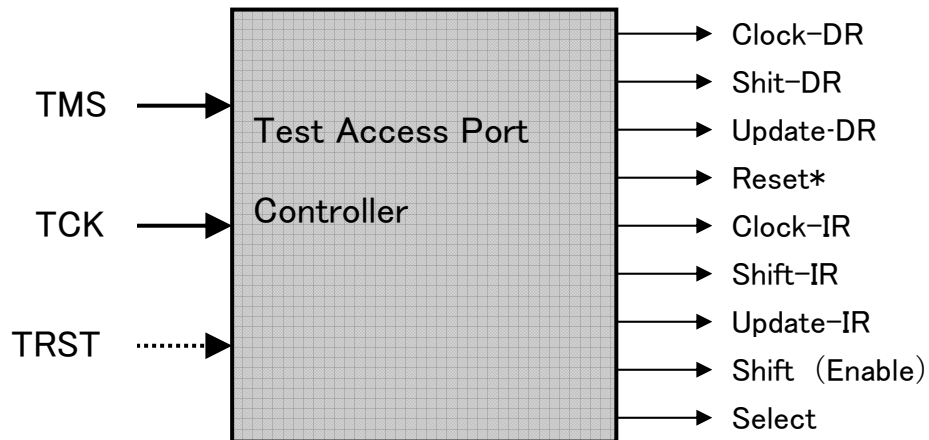
**TDI** : IR (インストラクションレジスター) にセットされるテスト命令と、バウンダリスキャンレジスターまたはその他の DR (データレジスター) にデータをセットするためのシリアル入力です。命令、データは、TCK の立ち上がりエッジで選択されたレジスターにセットされます。

**TDO** : TDO は、IR または DR からのテスト命令またはデータのシリアル出力です。選択されたレジスターの内容 (命令またはデータ) は TCK の立下りエッジでシフトされます。TDO は Shift-IR ステートまたは Shift-DR ステートのときだけアクティブになりその他のステートでは High-impedance 状態になっています。

**TRST\*** : TRST\* は負論理入力で TAP コントローラを非同期的に初期化します。TAP コントローラが初期化されることによって、設計に含まれているその他のテストロジックの非同期的初期化が行われます。このリセット信号はデバイスを通常の動作モードにし、バウンダリスキャンレジスターを非動作状態にします。この TRST はオプションで、必ず備える必要はありません。

### 4.3 TAP コントローラ

TAP コントローラは、TMS、TCK からの入力によって 16 ステートを動きますが、そのときのステートに応じて各種の制御信号を作成し、ターゲットとしているレジスターに送ります。



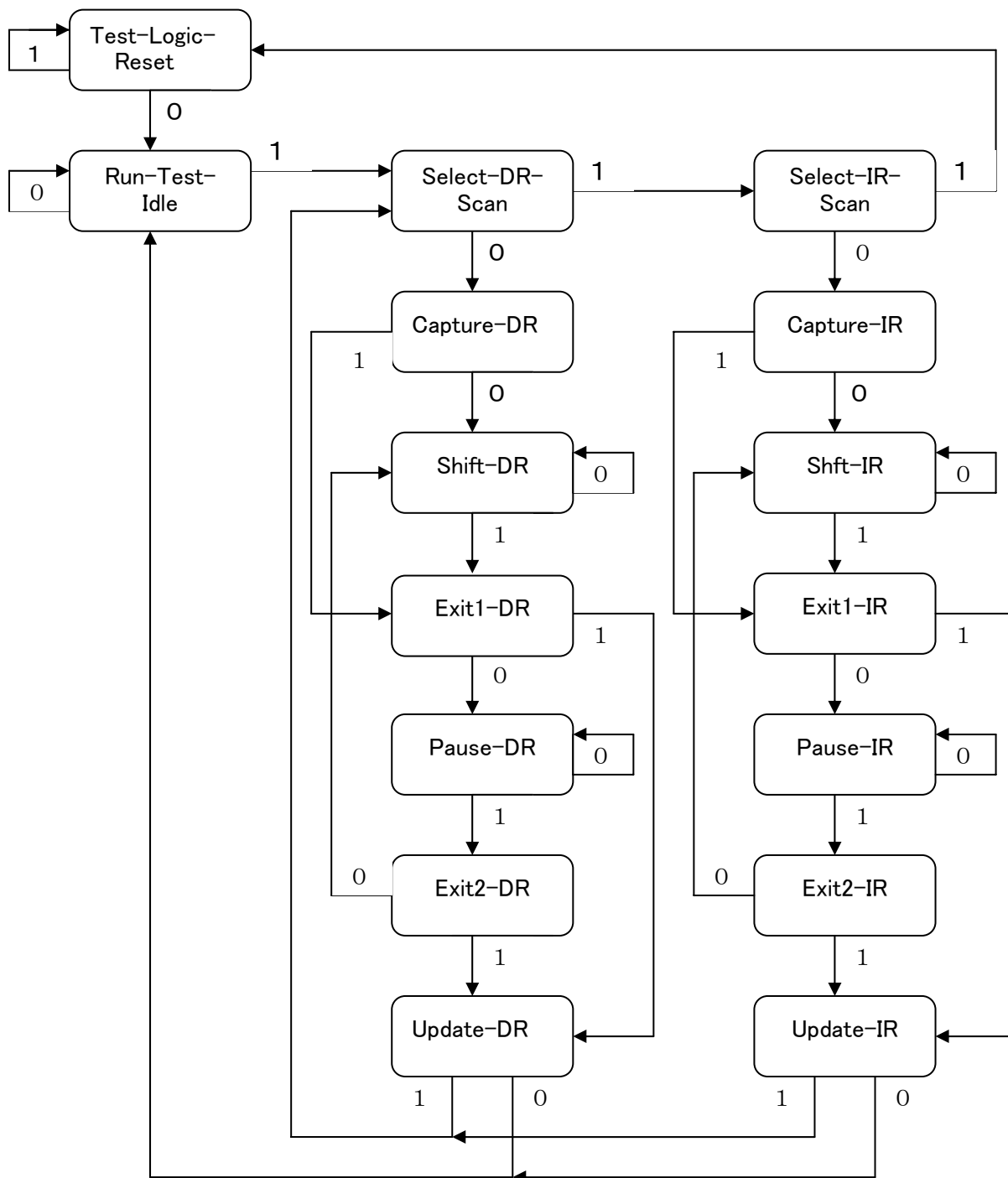
これらの信号には IR 専用の信号 (Clock-IR、Shift-IR、Update-IR)、全データレジスターに共通の信号 (Clock-DR、Shift-DR、Update-DR)、TDO にデータレジスターを接続するか、インストラクションレジスターを接続するかを選択する信号 (Select)、TDO をハイインピーダンス状態からアクティブにする Shift 信号 (Enable)、それに Reset 信号があります。

#### 4.4 TAP コントローラステート

TAP コントローラは TMS と TCK によって次の 16 ステートを動くステートマシンで、後で述べる Instruction (命令) と組み合わせることで全レジスタを制御します。ステート名の一部が DR となっている左側のパスと IR となっている右側のパスの二つの主要なパスがあります。

DR はデータレジスタのことでバウンダリスキャンレジスタ、バイパスレジスタ、ID コードレジスタ、USER コードレジスタなどがあります。対象となるレジスタはカレント命令によって決まります。IR はインストラクションレジスタのことです。

各ステートの周囲の“0”、“1”の数字はステートが移るとき TMS への入力値です。これらのステートのなかで、Capture、Shift、Update のステートが特に重要です。

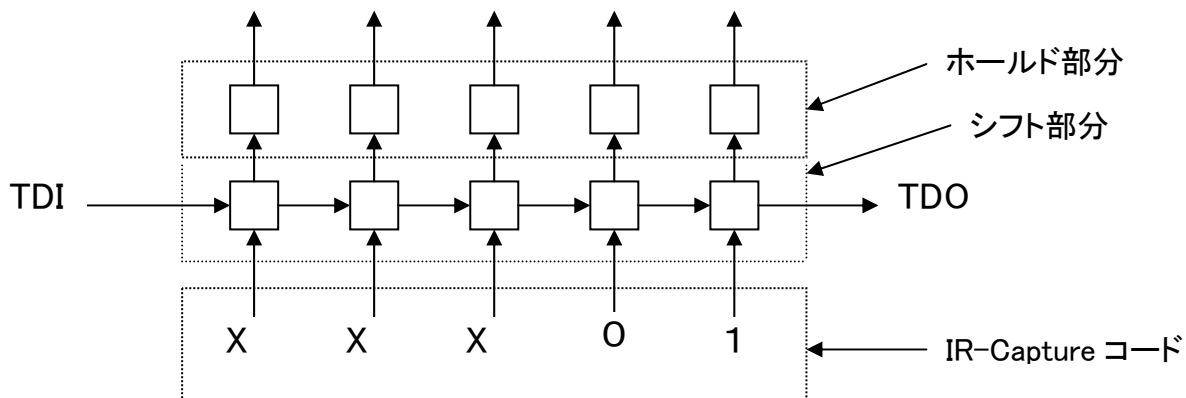


#### 4.5 バウンダリスキャンのレジスター

バウンダリスキャンデバイスのレジスターは、インストラクションレジスターとデータレジスターに分けられます。データレジスターには、バイパスレジスター、バウンダリレジスター、ID コードレジスター、ユーザーコードレジスターなどがあります。このうちインストラクションレジスター、バイパスレジスター、バウンダリスキャンレジスターは必須ですが、その他のレジスターはオプションです。

##### 4.5.1 インストラクションレジスター

インストラクションレジスターはデータレジスターが動作する Mode を定義します。(Mode については後で述べます。) インストラクションレジスターには下図のように TDI、TDO に接続されるシフト部分とホールド部分があります。インストラクションレジスターのビット長は最低 2 ビットと定められていますが、最大長は定義されていません。



##### • Test-Idle ステート

TAP が Test-Idle ステートのときインストラクションレジスターのホールド部分には、IDCODE 命令のビットパターンがセットされます。IDCODE レジスターはオプションなので、ID コードレジスターがないデバイスの場合は BYPASS 命令がセットされます。

##### • Capture-IR

TAP コントローラが Test-Logic-Reset ステートのとき、TMS に “0 1 1 1 0” を入力すると TAP コントローラは Capture-IR ステートに移行します。このときあらかじめデバイスに書き込まれている IR-Capture コードがシフト部分にパラレルロードされます。IR-Capture コードの最下位 2 ビットは “0 1” と定められていますが、上位ビットは定義されていません。

##### • Shift-IR

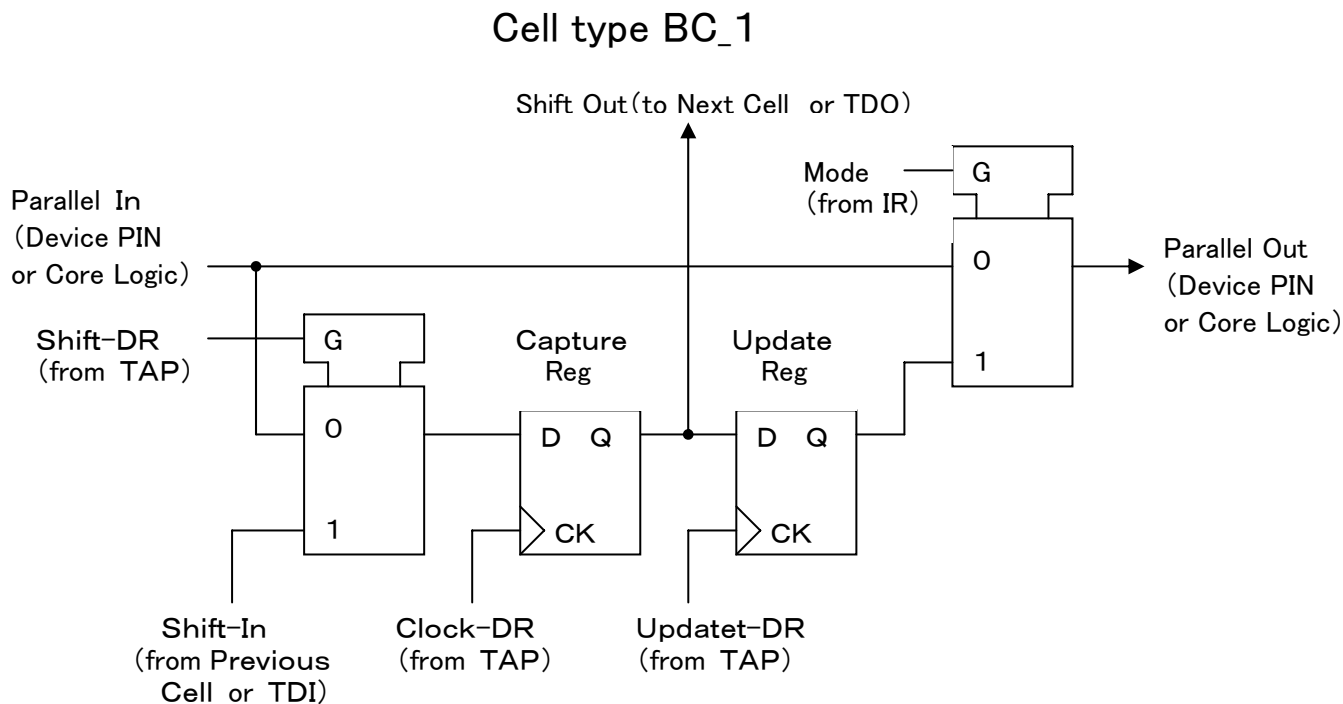
Capture-IR ステートで TMS に “0” を入力すると Shift-IR に移ります。そのまま TMS に “0” を入れ続けるとシフト部分にパラレルロードされた IR-Capture コードが TDO からシフトアウトされます。シフトアウトされた IR-Capture コードは、TAP 信号が正しく接続されているかどうかのテストのデータになります。IR-Capture コードを TDO からシフトアウトするとき、同時に TDI から次の命令コードをシフトインします。

##### • Update-IR

次の命令コードのシフトインが終わったら、TMS に “1 1” を入力すると Update-IR ステートに移ります。このときシフト部分にシフトインされた命令コードがホールド部分にパラレル出力され、ラッチされるとこの命令がカレント命令になります。

#### 4.5.2 バウンダリレジスタ

バウンダリレジスタは、デバイスのコアロジックとデバイスピン間に配置され、バウンダリスキャンテストを行う上で最も重要なレジスタで、通常複数個のバウンダリレジスタセルで構成されます。バウンダリレジスタセルの構造の一例を下図に示します。この例の場合、入力部、出力部の二つのマルチプレクサと二つのフリップフロップから構成されています。このセルは入力セルとしても出力セルとしても使えます。



##### • Capture-DR

TAP が Capture-DR ステートに移ったとき、入力部マルチプレクサは“0”に設定されているので Parallel In (デバイスピンまたはコアロジック) の状態が Capture Register の出力部、すなわちシフト経路にパラレルロードされます。

##### • Shift-DR

Capture-DR で TMS に“0”を入力すると Shift-DR に移ります。このとき入力部マルチプレクサは“1”に設定され、さらに TMS に“0”を入れ続けるとシフト部分にパラレルロードされたデータがバウンダリレジスタを伝って TDO からシフトアウトされます。

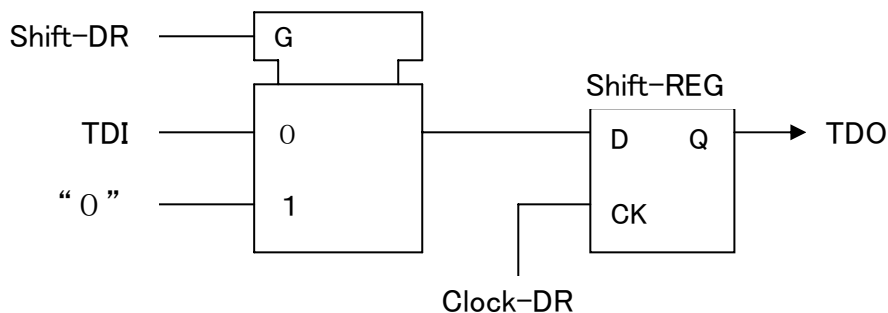
##### • Update-DR

このステートではシフト経路の信号が Update Register から出力されます。出力部マルチプレクサはそのときの命令によって Mode 信号が変わります。この例のセルでは、命令が EXTEST のとき Mode 信号は“1”でシフト経路の信号が Parallel Out に出力されます。また命令が SAMPLE/ PRELOAD のときは Mode 信号が“0”でシフト経路の信号は Update レジスタにラッチされますが、Parallel Out には出力されません。

バウンダリスキャンによるデバイス間の接続テストはバウンダリレジスタをターゲットとして、上記の Capture-DR、Shift-DR、Update-DR を繰り返すことによって行います。

### 4.5.3 バイパスレジスタ

バイパスレジスタの一例を示します。



バイパスレジスタは、BYPASS 命令で TDI と TDO の間に選択される 1 ビットのレジスタで、パラレル機能はありません。ターゲット基板に複数のバウンダリスキャン対応のデバイスがある場合、そのときのテストに関係のないデバイスをバイパスにしておくことでテスト時間を短縮できます。

#### • Capture-DR

カレント命令が BYPASS で Capture-DR ステートに移ったとき、シフトレジスタは “0” を取り込むことが定められています。

#### • Shift-DR

TDI から入力された信号を TDO へ出力します。

#### • Update-DR

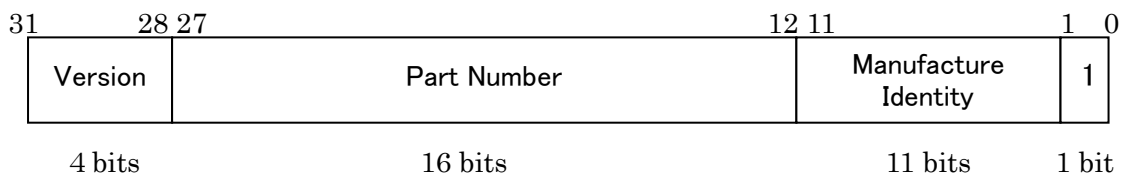
パラレル機能がないので Update-DR ステートに移っても何も影響されません。

### 4.5.5 ID コードレジスタ

32 ビットのレジスタで Capture と Shift 動作を行い、Update では何も影響されません。個々のレジスタの構造はバイパスレジスタと同じものが使えます。

#### • Capture-DR

カレント命令が IDCODE で Capture-DR ステートに移ったとき、シフトレジスタは デバイス固有のビットパターンを取り込みます。



#### • Shift-DR

シフトレジスタのビットパターンを TDO へ出力します。

#### • Update-DR

パラレル機能がないので、Update-DR ステートに移っても何も影響されません。



## 5. バウンダリスキャンの動作モードと命令

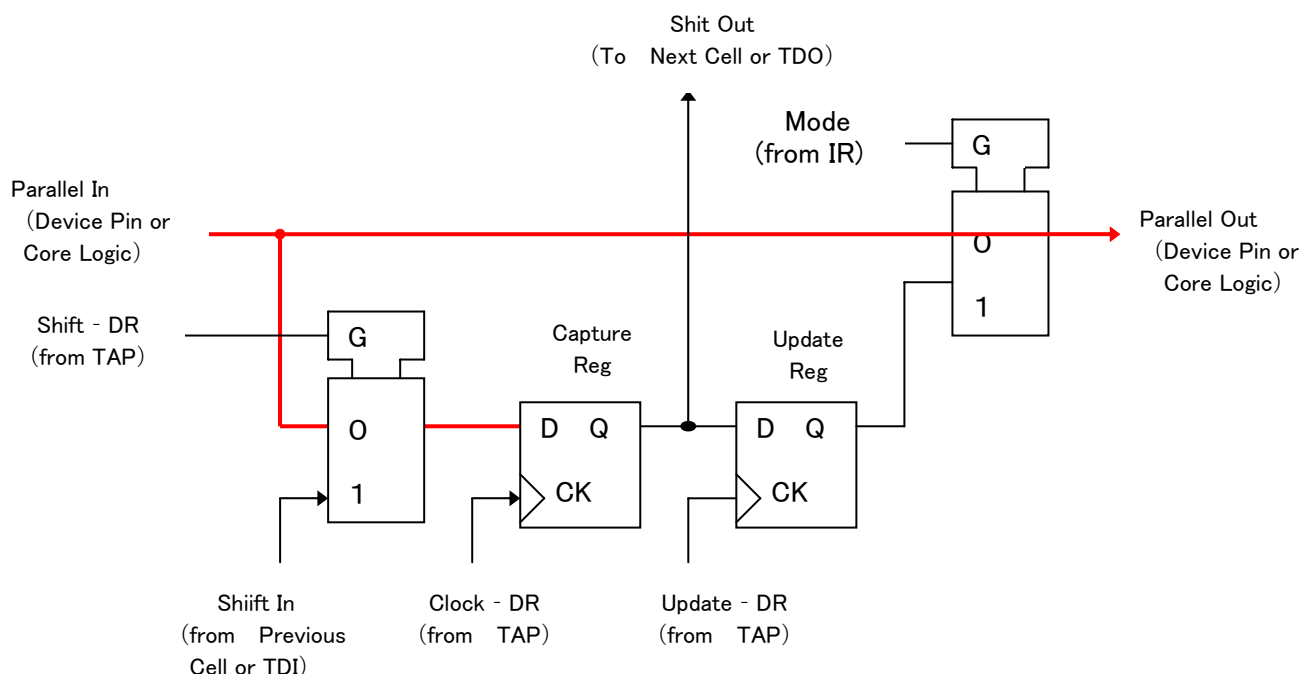
### 5.1 動作モード

バウンダリスキャンデバイスには、ノーマルモードとテストモードの二つの動作モードがあります。ノーマルモードのときはデバイスピンと内部ロジックは接続されており、通常動作していてもかまいません。

テストモードではデバイスピンと内部ロジックは切り離され、デバイスは通常動作はできません。バウンダリスキャンの命令は命令によってどちらのモードになるかが定められています。

#### 5.1.1 ノーマルモード

ノーマルモードでは **SAMPLE/PRELOAD**、**BYPASS**、**IDCODE**、**USERCODE** 命令を実行できます。下図のように、通常の信号はバウンダリスキャンセルの出力部のマルチプレクサが“0”なのでセルを自由に通過します。カレント命令が **SAMPLE/PRELOAD** のとき、この通常の信号をあるタイミングで **Capture** し、シフトアウトしてテストデータとして解析します。

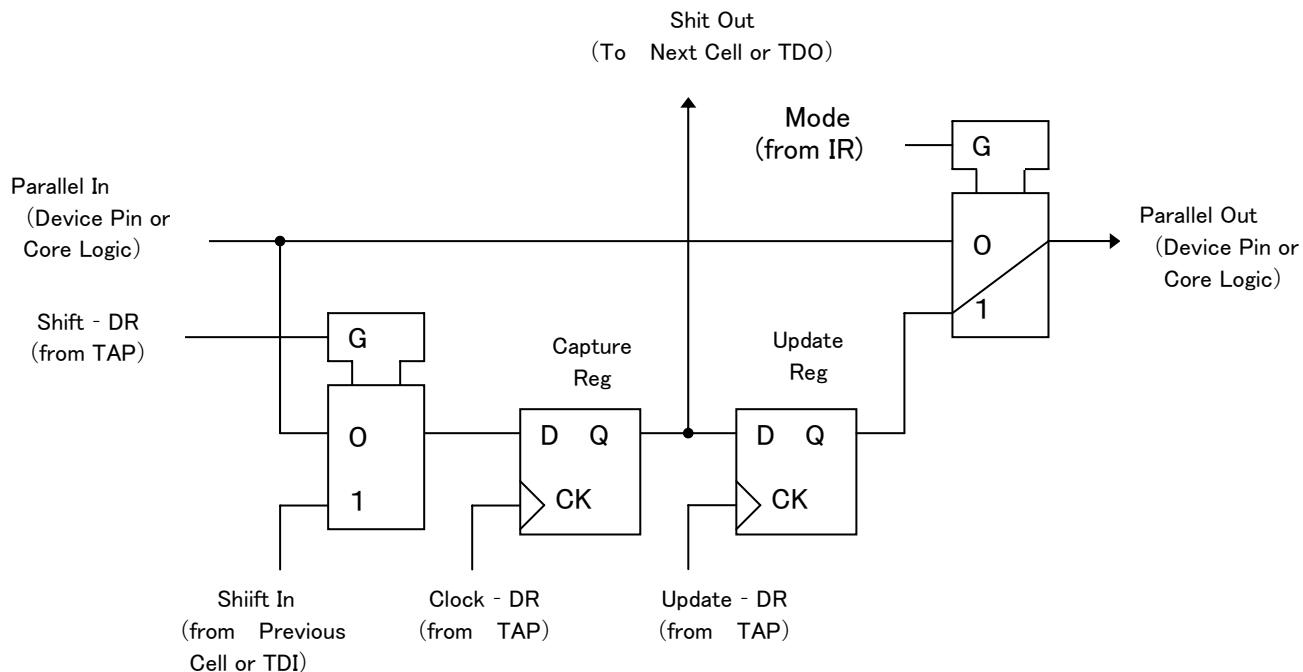


この動作を、順を追って説明します。

**Shift-IR** ステートで **SAMPLE/PRELOAD** 命令をセットします。Update-IR ステートを通過すると TDI、TDO の間にバウンダリレジスターが選択されます。このときセルの入力部マルチプレクサは“0”に設定されているのでデバイスピンの状態は **Capture Register** の入力部に接続されています。あるタイミングで、TAP コントローラを **Capture-DR** ステートに移すとそのときのデバイスピンの状態が **Capture Register** の出力部（シフト経路）に取り込まれます。**Shift-DR** に移ると入力部マルチプレクサは“1”に設定され、**Clock-DR** を入れ続けると、取り込んだデータがバウンダリスキャンレジスターチェーンを経由して **TDO** から出力されます。

### 5.1.2 テストモード

テストモードでは EXTEST、INTEST、RUNBIST 命令が実行できます。下図の例のセルでは出力部のマルチプレクサは“1”になっているのでデバイスピンと内部ロジックが直接接続されません。デバイスピンまたは内部ロジックへの Parallel Out 信号はバウンダリスキャンセルが完全にコントロールできることになります。出力すべき信号を Shift-DR ステートで TDI から送り込み、シフト経路にセットします。次に、Update-DR ステートに移ってセットした信号を出力部マルチプレクサから出力します。



また Parallel In からの入力信号は Capture-DR ステートでシフト経路に取り込んだあと、Shift-DR ステートで TDO からシフトアウトしてデータの解析ができます。

## 5.2 命令

バウンダリスキャンの命令は、大きくパブリック命令とプライベート命令に分けられます。パブリック命令は、その命令が実行される時、その実行内容が **IEEE-1149.1 Standard** で定められています。また、パブリック命令は必須命令とオプション命令に分けられ、三つの必須命令は必ず実装されている必要がありますが、オプション命令はデバイスの設計者にまかされています。

バウンダリスキャンテストには二つの動作モードがあります。ノーマルモードでは、デバイスのコアロジックと入出力ピンが接続された状態になっています。デバイスが通常動作をしているときにこれらの命令を実行しても通常動作に影響を与えることはありません。テストモードではデバイスのコアロジックと入出力ピンは切り離されています。そこでバウンダリレジスターにテストパターンを送り込んで外部の接続テストを行うことができます。

命令の種類		動作モード
パブリック命令	必須命令	EXTEST BYPASS SAMPLE/PRELOAD テスト ノーマル ノーマル
	オプション命令	INTEST IDCODE USERCODE RUNBIST CLAMP HIGHZ テスト ノーマル ノーマル テスト テスト テスト
プライベート命令		???

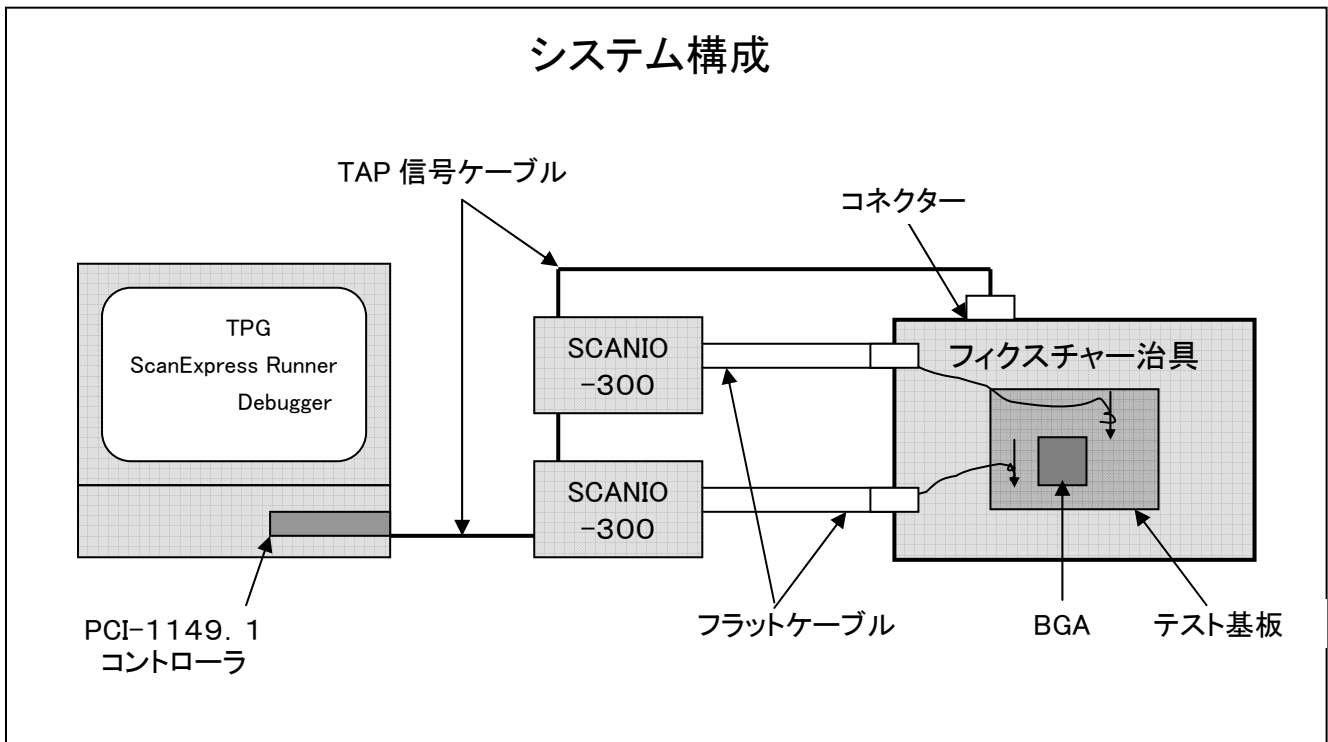
## 6. ScanExpress 製品 導入例

### 6.1 導入例-1

テスト対象基板 : PDA (Personal Digital Assistance) 用基板

テスト目的 : BGA の実装状態をテスト

基板の特徴 : バウンダリスキャン対応のデバイスが 1 個



- この例では、テスト基板にバウンダリスキャン対応のデバイスが 1 個しか搭載されていないので、テスト基板に TP を設けていただきプローブでアクセスして、外部の SCAIO-300 と接続してテストすることにしました。
- SCAIO-300 にはバウンダリスキャン対応の ASIC が 6 個搭載されており、300 ピンの双方向ピンがあります。BGA のピン数が 400 ピンだったので SCANIO-300 を 2 個使用しています。
- 後継機種では、1 台のフィクスチャー治具で ICT と BST を In-line で実施できるようにしました。

## 6.2 導入例-2

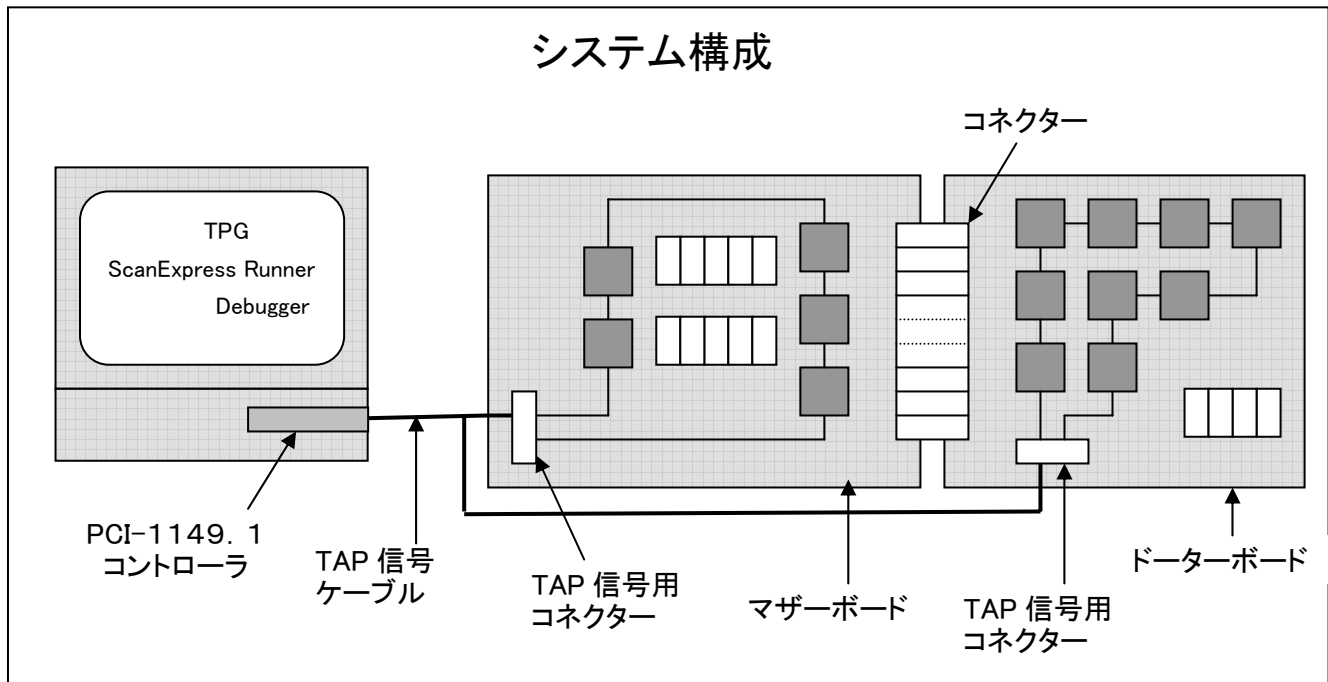
テスト対象基板：ネットワーク制御用基板

テスト目的：バウンダリスキャン対応デバイス同士およびメモリーとの接続状態をテスト

基板の特徴：マザーボードとドーターボードの2枚一組

マザーボードにはバウンダリスキャン対応デバイス5個とメモリーが25個

ドーターボードにはバウンダリスキャン対応デバイスが9個とメモリーが8個



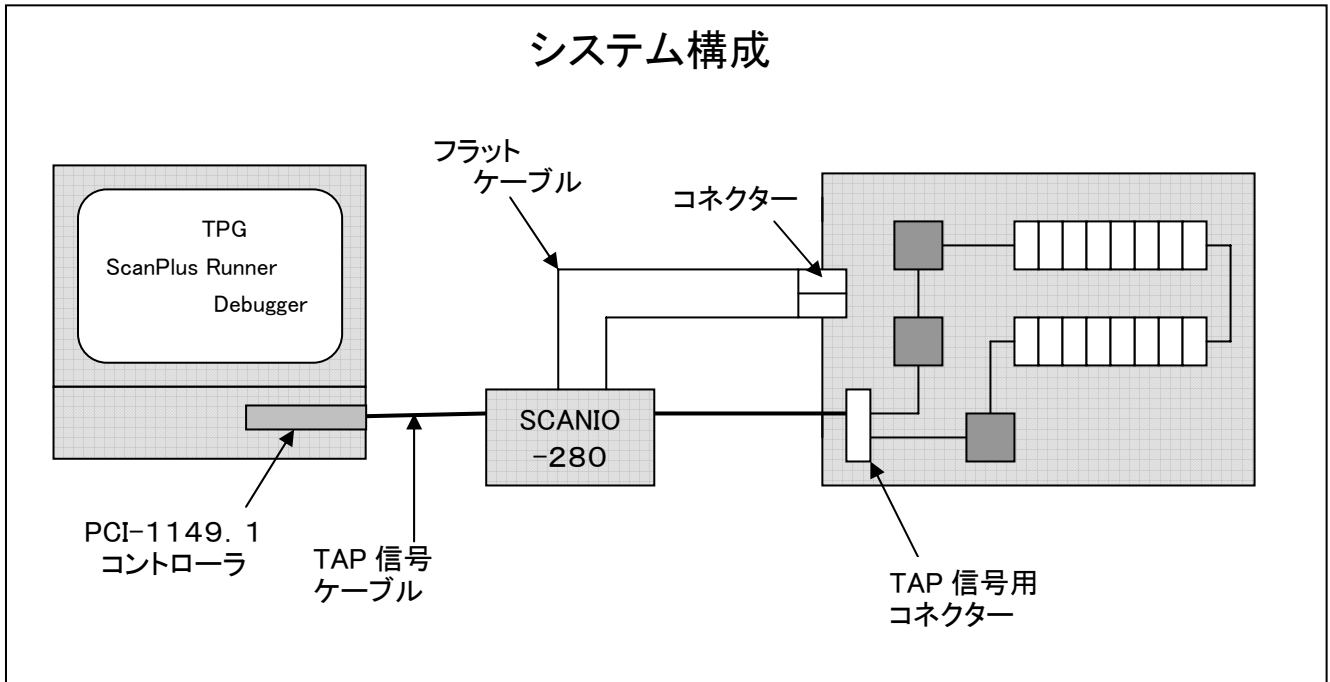
- この例では、まずマザーボードはマザーボード単体で、ドーターボードはドーターボード単体でテストできるようにしました。しかしそれではコネクターで接続されている約200の回路が検査できないので、マザーボードとドーターボードをコネクターで接続した状態でもテストできるようにしました。
- メモリーはSDRAM、SSRAMでしたので、設計段階でClock回路を考慮していただき、バウンダリスキャンテスト中はバウンダリスキャンデバイスからこれらのメモリーにClockを入力できるようにしていただきました。
- また、メモリークラスターテストのときバウンダリスキャン対応デバイスとメモリーの間のバッファ（バウンダリスキャンに対応していないデバイス）も同時にテストしています。その結果テストカバレッジは総デジタル回路の90%に達しました。

### 6.3 導入例-3

テスト対象基板：カラープリンター用基板

テスト目的：量産時は HP3070 で ICT（アナログ、デジタル）、BST を実施するが試作段階でバウンダリスキャンテストを確認する

基板の特徴：バウンダリスキャン対応の ASIC が 3 個とバウンダリスキャン対応のメモリーが 16 個



- この例はバウンダリスキャンテストの検証ということで導入された例です。BSDL ファイルに不備があり、HP3070 での BST がなかなか立ち上がらないということで事前に ScanPlus でバグを取り除く目的で導入されました。
- このボードからは外部にアドレスバス、データバスが約 100 回路出ていますので SCANIO-280 を使用してそれらの回路も検査しました。
- アメリカで設計されたボードですのでバウンダリスキャンテストの設計については事前に打合せを行っていません。バウンダリスキャン対応デバイスが合計 19 個あり TCK、TMS 信号が心配でしたがこれらの回路にはきちんとバッファが入れられていました。
- バウンダリスキャン対応のメモリー以外に SDRAM が 2 個搭載されていますが Clock 端子だけがバウンダリスキャン対応のデバイスに接続されておらずテストできませんでした。事前に打合せを行うことができていればこれらもテストできていました。