

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願い申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日

ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無いことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

H8SX ファミリ

バウンダリスキャン入門編

要旨

この資料は、バウンダリスキャンの基礎を解説したものです。

本編は入門編です。具体的なデバイスを使用した使い方や応用例は、別アプリケーションノートで説明します。

1項、2項、3項では、バウンダリスキャンとは何か、バウンダリスキャンでは何ができるのかを紹介し、4項では、バウンダリスキャンテストに対応したデバイスの仕組みを説明し、5項ではバウンダリスキャンテストに必要となる BSDL ファイルについて説明しています。最後に6項では参考としてバウンダリスキャンに必要となるファイルやソフトウェアの一例をあげています。

目次

1. バウンダリスキャンとは？	2
2. バウンダリスキャンテスト概要	4
3. バウンダリスキャンテスト例	6
4. デバイスの仕組み	9
5. BSDL ファイル	32
6. バウンダリスキャンテストシステム構成例	42
7. 参考文献	44
8. 参考ドキュメント	44

1. バウンダリスキャンとは？

バウンダリスキャンとは、ICの内部と外部の境界（バウンダリ）を検査（スキャン）する機能です。バウンダリスキャン技術を用いたICやプリント基板のチェックテストのことをバウンダリスキャンテスト、あるいはJTAGテストといいます。

JTAGテストは1985年にヨーロッパで考案され、1990年にIEEE std 1149.1-1990 Standard Test Access Port and Boundary-Scan Architectureとして規格化されました。その際の委員会の名称がJETAG（Joint European Test Action Group）であり、その後ヨーロッパ以外のメンバーが加わったためにEuropeanが取れ、JTAG（Joint Test Action Group）と改められました。この規格は現在、委員会名であるJTAGと正式な規格名称が同等に扱われています。

従来のICのピンをテストプローブにあてて行うインサーキットテストは、現在のICの集積化でピン間隔がテストプローブの間隔を下回っているためチェックテストが難しくなっています（図1(a)）。

バウンダリスキャンテストはこの問題を解決します。バウンダリスキャンテスト対応デバイスでは、ICの内部と外部の境界部分（バウンダリ）にバウンダリスキャンセルというシフトレジスタを仕込み、バウンダリスキャン用の入力端子と出力端子を持ちます。バウンダリスキャンテストでは、端子の入出力状態をバウンダリスキャンセルに取得させ、バウンダリスキャン用出力端子からセルデータを出力することにより、端子の状態を知ることができます（図1(b)）。

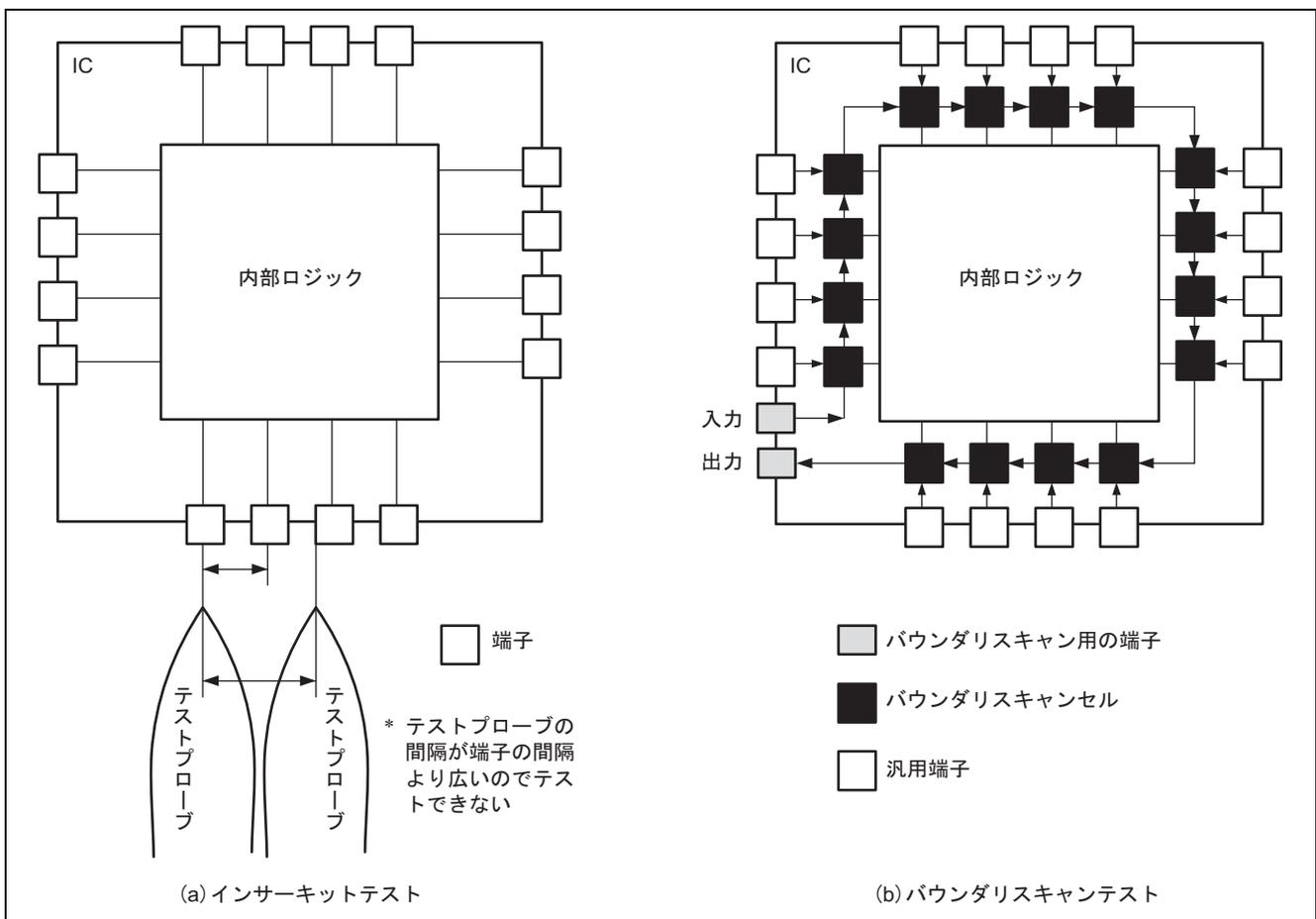


図1 インサーキットテストとバウンダリスキャンテスト

また、現在 IC は、IC の四辺から端子が出ている QFP (Quad Flat Package) タイプ (図 2(a)) と、半田による小さなボール状電極が IC 下部に並んでいる BGA (Ball Grid Array) タイプ (図 2(b)) が普及しています。BGA タイプの IC でインサーキットテストを実施しようとした場合、IC の直下に端子があるために、実装時にテストプローブを端子にあてるのが物理的に不可能です。しかし、バウンダリスキャンテストでは内部にバウンダリスキャンセルを持つために IC の形状に影響されことなくテストすることができます。

また、バウンダリスキャンテストでは IC 本来の動作に影響を与えず端子ごとの状態を調べることや、IC 本来の動作を止めて IC 内部や外部のテストを行うことができます。

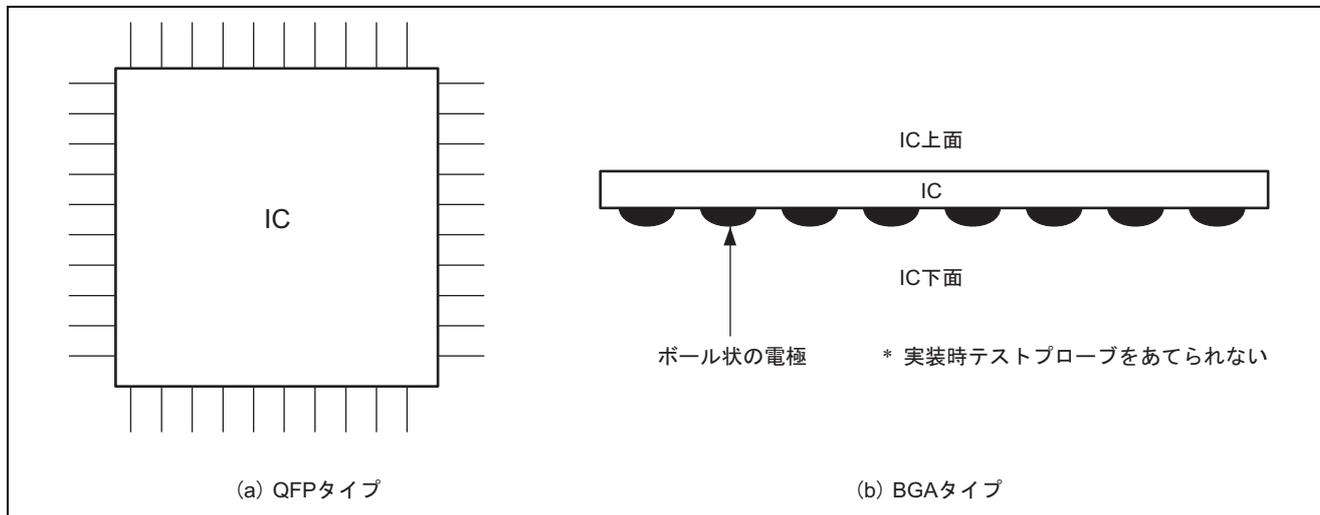


図 2 IC のパッケージタイプ

2. バウンダリスキャンテスト概要

バウンダリスキャンには、以下のような特長があります。

- バウンダリスキャンに必要な端子は5つ
- 対応デバイスは内部に TAP コントローラ、各種レジスタを内蔵する
- デバイスは専用のコマンドを入力にすることにより操作する
- コマンドにより、テストモード（デバイスの通常動作を無効にして端子状態／内部ロジックを制御するモード）またはノーマルモード（デバイスを通常動作させた状態で端子状態／内部ロジックを監視するモード）での動作が可能

バウンダリスキャンデバイスのイメージを図3に示します。バウンダリスキャンデバイスは、外部端子として5つの TAP（テストアクセスポート）を備え、その内部には内部ロジック（コア）と独立して、TAP コントローラ、バウンダリスキャンセル、インストラクションレジスタなどを内蔵しています。

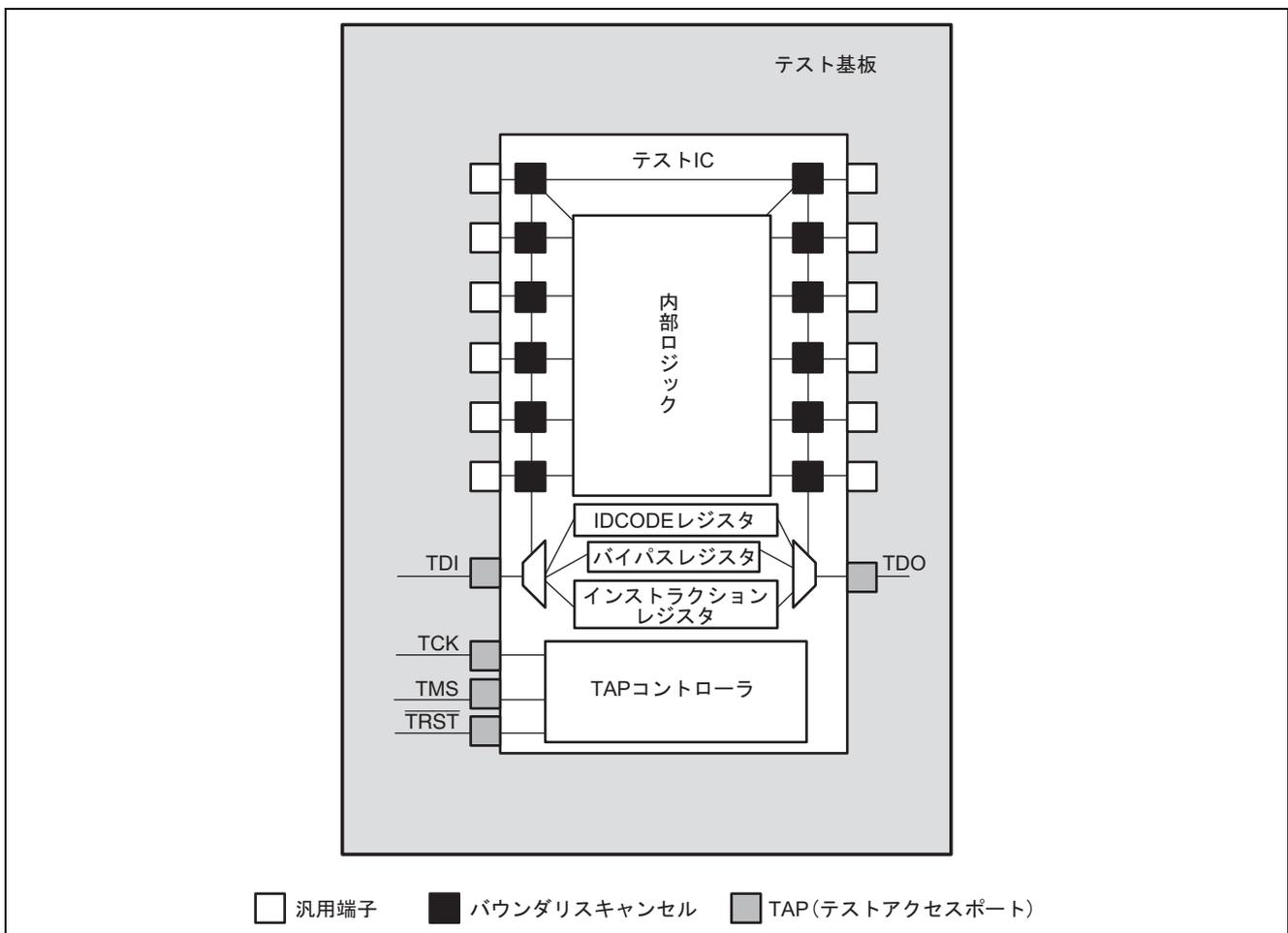


図3 バウンダリスキャンデバイスイメージ

バウンダリスキャンの動作を管理・制御しているのが TAP コントローラと呼ばれるステートマシンであり、TAP の入力操作によりその状態を制御することができます。TAP はそれぞれ異なる機能を持った5つの端子からなります（表1）。TAP コントローラの制御に必要な端子が TMS 端子であり、'0'と'1'の入力で TAP コントローラのステートを操作可能です。TAP コントローラはその状態に応じて、TDI 端子と TDO 端子の接続先を変更します。たとえば、TDI 端子がインストラクションレジスタに接続されたときには、コマンドの入力が可能となり、TDI 端子がバウンダリスキャンセルに接続されたときには、セルに任意の値を設定することが可能となります。TDO 端子についても同様に、接続先が変更されることでインストラクションレジスタ値（IR ステータスワードと呼ばれる）やバウンダリスキャンセル値を同一端子から取り出すことができます。TAP については「4.2 TAP（テストアクセスポート）」で、TAP コントローラについては「4.3 TAP コントローラ」でそれぞれ詳しく説明します。

表 1 TAP の機能

端子名	入出力	機能
TDI (Test Data In)	入力	命令やデータをシリアル入力する端子
TDO (Test Data Out)	出力	データをシリアル出力する端子
TCK (Test Clock)	入力	クロックを供給する端子
TMS (Test Mode Select)	入力	TAP コントローラのステートを制御する端子
TRST (Test Reset)	入力	TAP コントローラをリセットする端子

バウンダリスキャンで使用するレジスタの種類と機能は表 2 のとおりです。これらのレジスタは TAP コントローラを操作することでのみ、アクセスが可能です。バウンダリスキャンテストでは端子状態の操作や端子状態の監視を行います。この機能を実現可能にしているのがバウンダリスキャンレジスタです。バウンダリスキャンレジスタとは、デバイス内部で複数のバウンダリスキャンセルがチェーン状につながって形成するシフトレジスタのことを指します。バウンダリスキャンレジスタは外部端子と内部ロジックの境界に存在し、かつバウンダリスキャンセル同士が内部で直列に接続されているため、TDI/TDO 端子から一括で入出力を行うことが可能となります。レジスタについては、「4.4 レジスタ」で詳しく説明します。

表 2 レジスタの種類と機能

レジスタの種類		機能
インストラクションレジスタ		命令コード (SAMPLE 命令など) を格納し、デコードする
データレジスタ	バイパスレジスタ	TDI から入力されたデータを通過させ TDO にバイパスさせる
	バウンダリスキャンレジスタ	各端子の状態を格納する
	IDCODE レジスタ	デバイスの IDCODE を格納する

バウンダリスキャンテストにおいてデバイスの端子状態制御を行うには、バウンダリスキャン専用のコマンドを発行する必要があります。コマンド発行の際は、TAP コントローラを操作し、TDI 端子をインストラクションレジスタに接続した状態でコマンドコードを入力します。コマンドにはいくつかの種類があり、端子状態制御のほかにも IDCODE と呼ばれるデバイス固有 ID の取得などを行うことができます (表 3)。また、コマンド動作には、ノーマルモードとテストモードの 2 つのモードがあります。ノーマルモードではデバイスを通常動作させた状態で端子状態/内部ロジックを監視することができ、テストモードではデバイスの通常動作を無効にして端子状態/内部ロジックを制御することができます。ノーマルモードの代表例としては SAMPLE/PRELOAD 命令があり、この命令ではデバイスの通常動作状態 (マイコンであれば、プログラムを実行させた状態) において、端子状態をバウンダリスキャンセル経由で取得することが可能です。テストモードの代表例としては EXTEST 命令があり、この命令ではデバイスの動作と関係なく、端子状態を任意の値に操作することができます (内部ロジックから端子への出力は遮断されます)。コマンドについては、「4.5 コマンド」で詳しく説明します。

表 3 コマンドの種類と機能

コマンドの種類	機能	動作モード
SAMPLE/PRELOAD	端子状態や内部ロジックのデータをバウンダリスキャンセルに渡す	ノーマルモード
BYPASS	入力データを TDI と TDO 間でバイパスする	ノーマルモード
EXTEST	バウンダリスキャンセルの値を出力する	テストモード
IDCODE	デバイスの IDCODE を識別する	ノーマルモード
CLAMP	バウンダリスキャンセルの値を出力し、かつバイパスする	テストモード
HIGHZ	すべての出力端子をハイインピーダンスし、かつバイパスする	テストモード

3. バウンダリスキャンテスト例

バウンダリスキャン機能を使うとどのようなことができるでしょうか？ここではインタコネクテストとクラスタテストという2つの便利なテスト方法を紹介します。

3.1 インタコネクテスト

インタコネクテストとは、複数のバウンダリスキャン対応 IC を用いて、片方の IC からもう片方の IC に正しくデータが届くかどうかをテストするものです。

簡略イメージを図4に示します。このテストでは、デバイス1のセルにあらかじめテストパターンをセットしておいて、EXTEST 命令でそれぞれの端子状態を出力し、デバイス2でSAMPLE/PRELOAD 命令を行います。正常なテストパターンをデバイス2側で確認することができれば「IC 端子同士がプリントパターンを通して正しく接続されている」という確認が取れます。

プリントパターン内で断線やショートが起きていた場合、デバイス2側でテストパターンどおりの値は確認されません。

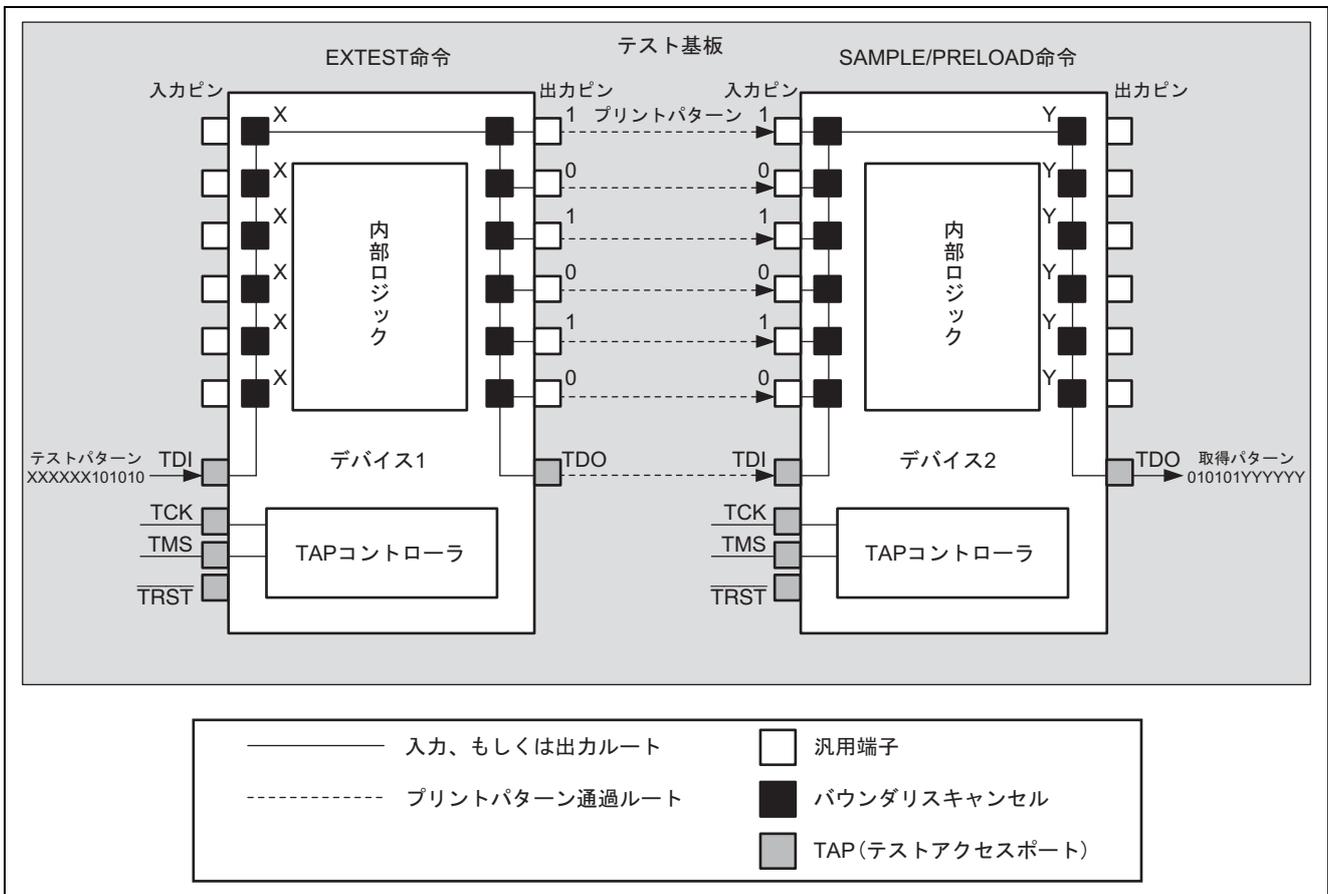


図4 インタコネクテスト簡略図

例として、端子数が 6 ピンで、インタコネクテストを 8 つのテストパターンで各 1 回ずつ (計 8 回) 行った場合に取得されるパターンを図 5 に示します。例では同一番号端子同士を接続しています。

プリント基板の導通に問題がない場合、図 5(a)のテストパターンをデバイス 1 にセットすると、デバイス 2 の各端子から同様のサンプルパターンを取得できます。

しかし、図 5(b)では、図 5(a)のテストパターンを与えているにもかかわらず、パターン 4 の pin4 とパターン 5 の pin3 でそれぞれ理論値と異なる '1' を取得しています。この場合、全パターンを通して pin3 と pin4 で全く同じ値が取得されていることから、pin3 と pin4 がテスト基板上でショートしているものと予想できます。

また、図 5(c)では、図 5(a)のテストパターンを与えているにもかかわらず、パターン 4 およびパターン 8 の pin3 で '0' を取得しています。この場合、全パターンを通して pin3 の値が '0' となっていることから、pin3 が基板上で断線している (もしくはグラウンドにショートしている) ものと予想できます。

	pin1	pin2	pin3	pin4	pin5	pin6
パターン1	0	0	0	0	0	0
パターン2	1	0	0	0	0	0
パターン3	0	1	0	0	0	0
パターン4	0	0	1	0	0	0
パターン5	0	0	0	1	0	0
パターン6	0	0	0	0	1	0
パターン7	0	0	0	0	0	1
パターン8	1	1	1	1	1	1

(a) 正常なサンプルパターン
(テストパターンと同じ)

	pin1	pin2	pin3	pin4	pin5	pin6
パターン1	0	0	0	0	0	0
パターン2	1	0	0	0	0	0
パターン3	0	1	0	0	0	0
パターン4	0	0	1	1	0	0
パターン5	0	0	1	1	0	0
パターン6	0	0	0	0	1	0
パターン7	0	0	0	0	0	1
パターン8	1	1	1	1	1	1

(b) ショートしている場合
(pin3とpin4がショートしている)

// // // // // 問題がある箇所

	pin1	pin2	pin3	pin4	pin5	pin6
パターン1	0	0	0	0	0	0
パターン2	1	0	0	0	0	0
パターン3	0	1	0	0	0	0
パターン4	0	0	0	0	0	0
パターン5	0	0	0	1	0	0
パターン6	0	0	0	0	1	0
パターン7	0	0	0	0	0	1
パターン8	1	1	0	1	1	1

(c) 断線している場合
(pin3が断線している)

図 5 サンプルパターン

3.2 クラスタテスト

クラスタテストは、プリント基板上で複数のバウンダリスキャン対応 IC の間にバウンダリスキャンに非対応の IC、抵抗、コンデンサなどのデバイスを挟み、バウンダリスキャン非対応のデバイスの動作をテストする方法です。動作の仕組みはインタコネクテストと同じですが、このテストを行う際は、あらかじめバウンダリスキャン非対応デバイスの入出力仕様が明確になっている必要があります。

クラスタテストは、デバイス 1 のセルにテストパターンをセットしておいて、EXTEST 命令でそれぞれの端子状態を出力します。デバイス 2 では SAMPLE/PRELOAD 命令を行います。テストパターンは、デバイス 1 側端子→プリントパターン→バウンダリスキャン非対応デバイス→プリントパターン→デバイス 2 側端子、という順序でデバイス 2 側端子に送られます。あらかじめ、バウンダリスキャン非対応デバイスの正常動作によってテストパターンがどのように変化するかを期待値として予測しておき、その値がデバイス 2 で確認された場合に、バウンダリスキャン非対応のデバイスの動作が正常であると判断します。

簡略イメージを図 6 に示します。

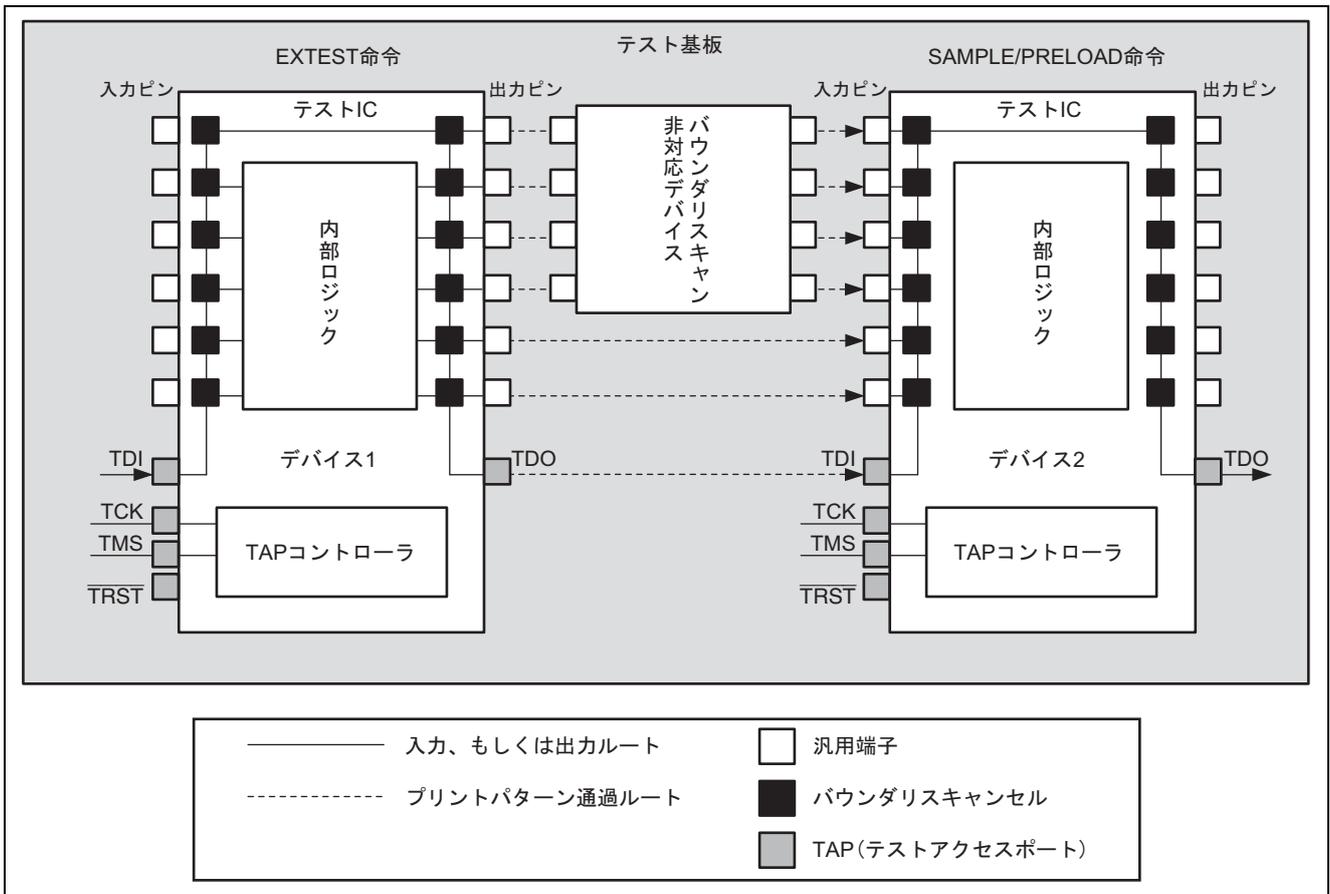


図 6 クラスタテスト簡略図

4. デバイスの仕組み

4.1 バウンダリスキャンデバイス構成

バウンダリスキャンテストは、「2. バウンダリスキャンテスト概要」でも述べたように、IC に内蔵されたバウンダリスキャンセル、バウンダリスキャン用のレジスタ、TAP コントローラ、TAP を用いて行います。また、TAP はホストコンピュータの JTAG コントローラに接続します。JTAG コントローラには TAP 信号をテスト基板に送受信する役割があります。本項では、TAP、TAP コントローラ、レジスタ、そしてコマンド（命令）について詳しく説明します。

バウンダリスキャンテストのデバイス構成を図7に示します。

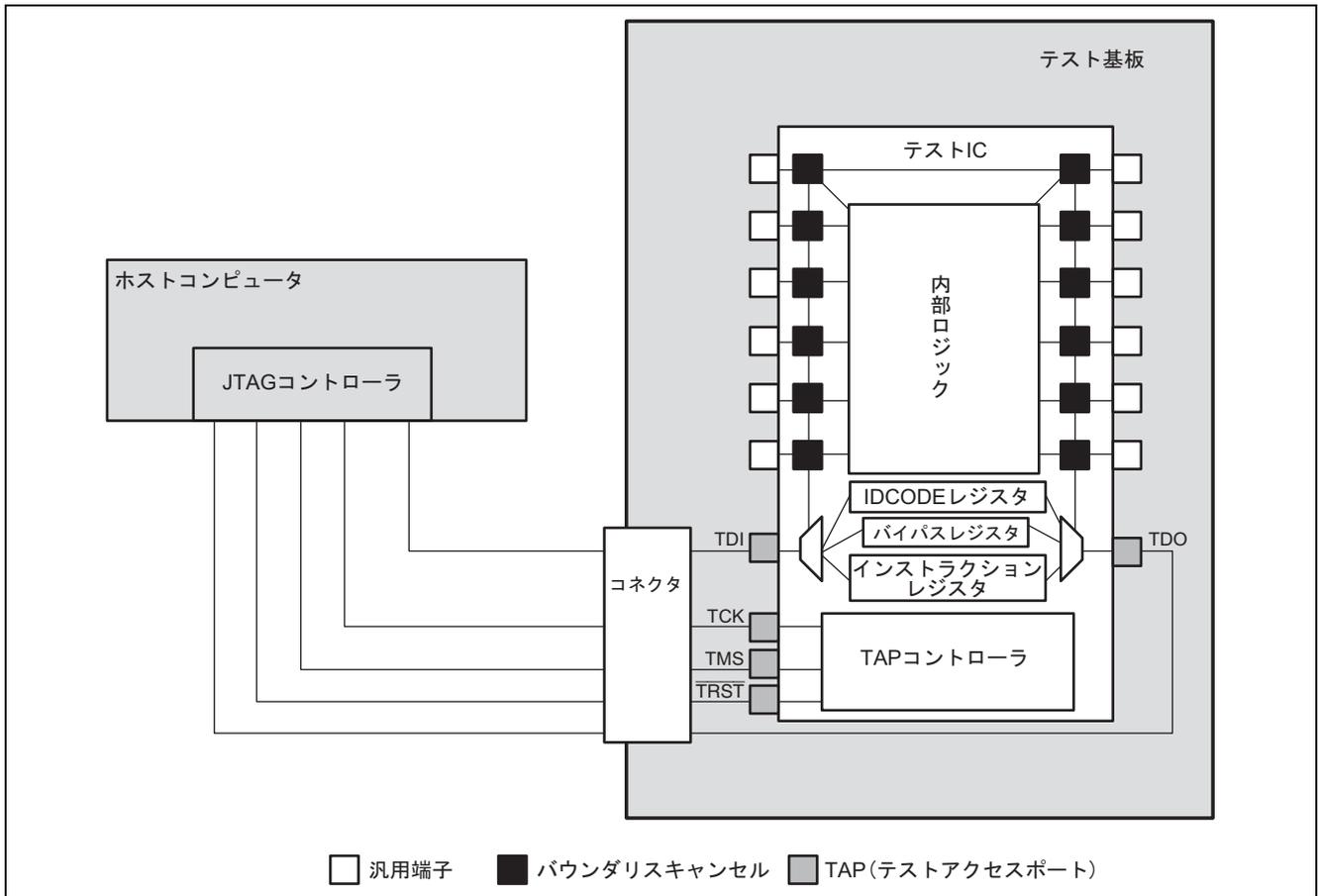


図7 バウンダリスキャンデバイス構成図

4.2 TAP (テストアクセスポート)

TAP とはバウンダリスキャンテストを行うための命令やデータの入出力をするためのポートです。TAP には TDI、TDO、TCK、TMS、 $\overline{\text{TRST}}$ の 5 つの端子があり、外部のホストコンピュータから制御されます。

表 4 に TAP 機能を示します。

表 4 TAP の機能

端子名	入出力	機能
TDI (Test Data In)	入力	命令やデータをシリアル入力する端子
TDO (Test Data Out)	出力	データをシリアル出力する端子
TCK (Test Clock)	入力	クロックを供給する端子
TMS (Test Mode Select)	入力	TAP コントローラの状態を制御する端子
$\overline{\text{TRST}}$ (Test Reset)	入力	TAP コントローラをリセットする端子

TCK の立ち上がりに同期して TDI、TMS、 $\overline{\text{TRST}}$ の各信号状態が取り込まれ TDO は TCK の立ち下がりに同期して出力されます。この TCK とその他の TAP 信号の関係は、「4.5 コマンド」でタイミングチャートの例により示します。

また、複数の IC を用いてテストを行う場合の配線例を図 8 で示します。

データの入出力に使用する TDI 端子と TDO 端子の配線は、隣り合う IC の TDO 端子と TDI 端子を接続した直列の配線となります。図 8 の場合、左端 IC の TDO 端子から出力されたデータは中央および右端 IC の内部を通過して JTAG コントローラの TDO 端子に入力されます。その途中経路で各データが変化することはありません。このような配線とすることで複数 IC のデータを 1 つながりのデータとして TDO からシフト出力させることができます。

入力端子である TCK 端子、TMS 端子、 $\overline{\text{TRST}}$ 端子は図中のすべての IC で共通の信号を使用するため、並列に接続しています。

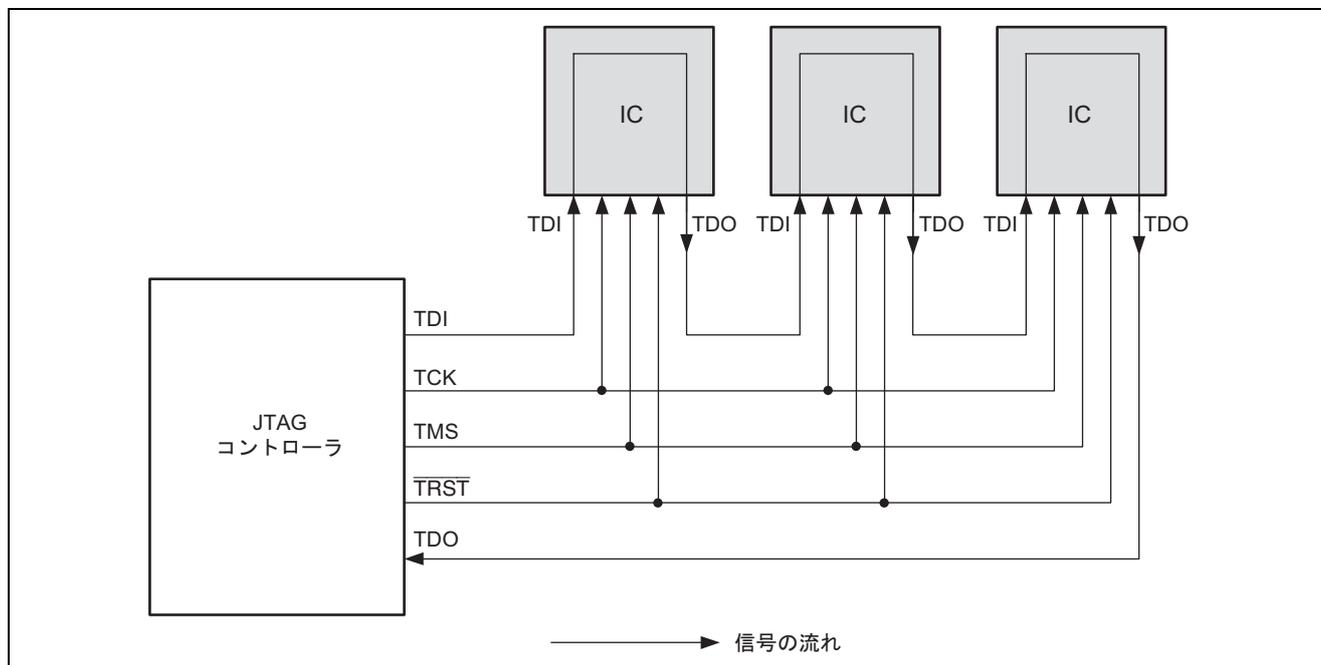


図 8 複数 IC 接続時の TAP 配線例

4.3 TAP コントローラ

TAP コントローラは 16 のステート (状態) をもつ順序回路で、それぞれのステートでレジスタを制御します。ステートから次のステートへの遷移は、TAP の TMS と TCK により制御されます。

ステートの遷移図を図 9 に示します。

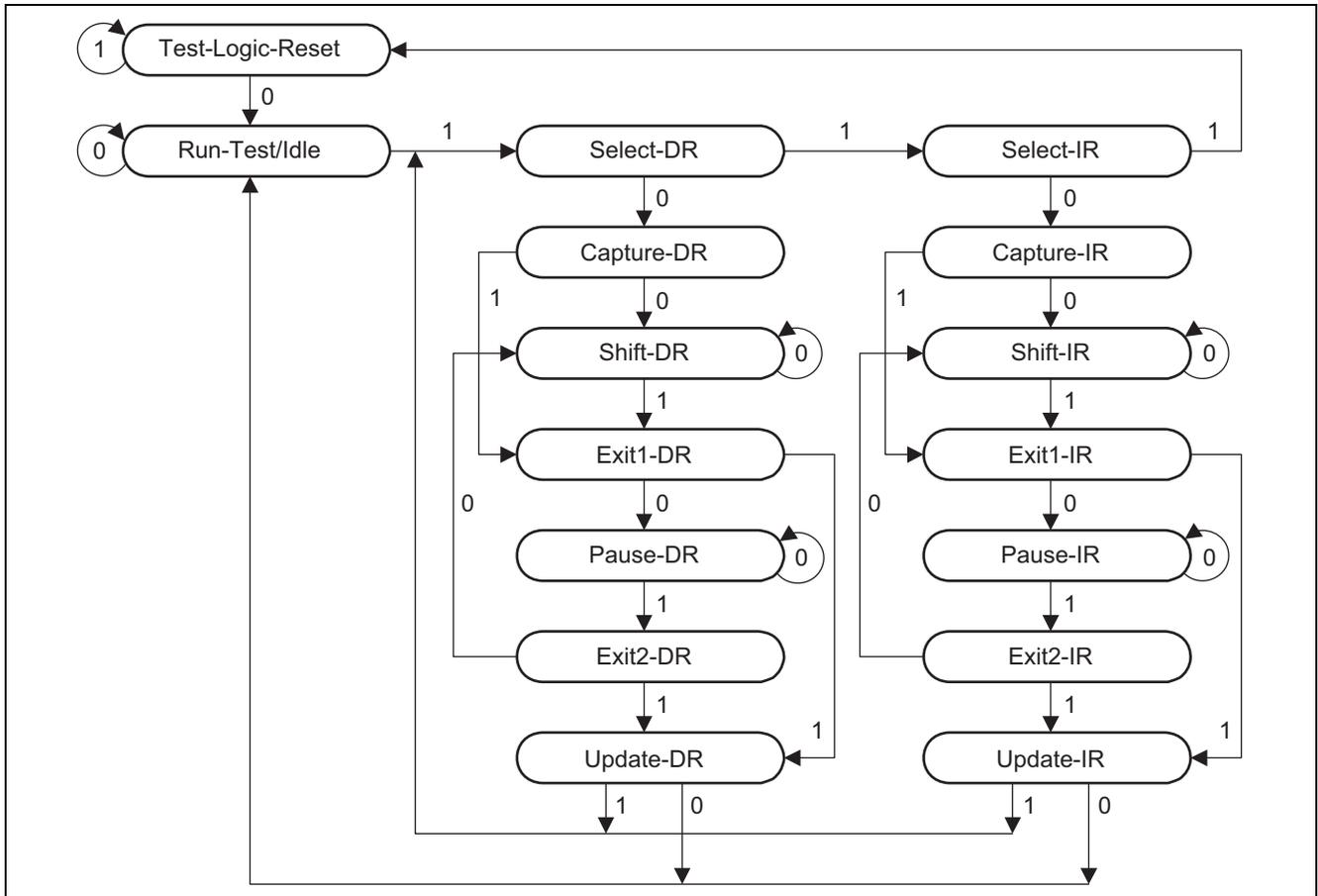


図 9 TAP コントローラのステート

図 9 では丸枠で囲まれた名前がステート名を示しています。矢印に付随した 0 と 1 の数字は TCK 立ち上がり時の TMS の値です。ステートは TCK の立ち上がりのタイミングで TMS の値どおりに矢印で示す方向に遷移していきます。

4.3.1 TAP コントローラのステート

TAP コントローラは、Test-Logic-Reset と Run-Test/Idle、そして7つの DR ステートと7つの IR ステートから成ります。DR とはデータレジスタのことで、命令コードによって制御されるレジスタです。IR とはインストラクションレジスタのことで、命令コードを格納するためのレジスタです。レジスタについて詳しくは「4.4 レジスタ」で述べます。

表 5 に TAP コントローラの 16 のステートを示します。

表 5 TAP コントローラのステート

ステート	操作対象	機能
Test-Logic-Reset	—	テストロジックのリセット状態（通常開始状態）
Run-Test/Idle	—	Test-Logic-Reset ステートからの通過状態
Select-DR	DR	次のステートを選択するための一時的なステート
Capture-DR		シフトレジスタへデータを取得する
Shift-DR		シフトレジスタのデータを1ビットシフトする
Exit1-DR		次のステートを選択するための一時的なステート
Pause-DR		レジスタのシフト動作を一時停止する
Exit2-DR		次のステートを選択するための一時的なステート
Update-DR		シフトレジスタのデータをラッチ回路にセットする
Select-IR		IR
Capture-IR	シフトレジスタへ IR ステータスワードを取得する	
Shift-IR	シフトレジスタのデータを1ビットシフトする	
Exit1-IR	次のステートを選択するための一時的なステート	
Pause-IR	レジスタのシフト動作を一時停止する	
Exit2-IR	次のステートを選択するための一時的なステート	
Update-IR	シフトレジスタのデータをラッチ回路にセットする	

(1) Test-Logic-Reset ステート

Test-Logic-Reset は、TAP コントローラの初期状態です。電源投入時や $\overline{\text{TRST}}$ 入力時にこのステートに遷移します。また、TMS を '1' にしたまま 5 回以上の TCK を入力した場合もこのステートに必ず遷移します。このように、TMS と TCK により同じ動作ができるため、 $\overline{\text{TRST}}$ はオプション信号として実装されていないことがあります。

また、Test-Logic-Reset ステートに遷移すると、IDCODE 命令がセットされます。IDCODE 命令が実装されていない場合は、BYPASS 命令がセットされます。

(2) Run-Test/Idle ステート

Run-Test/Idle は、アイドル状態のステートです。TMS に '0' を入力する限り、このステートに留まっていることができます。この期間は DR、IR ともに変化しません。

(3) Select ステート

Select ステートは次のステートを選択するための一時的なステートです。

(4) Capture ステート

Capture ステートは、シフトレジスタにデータを取得することができるステートです。

Capture-DR ではバウンダリスキャンセルが入力セルの場合、端子の状態をバウンダリスキャンセルに取得し、出力セルの場合、内部ロジックの値をバウンダリスキャンセルに取得します。入力セル、出力セルについては「4.4.2.(1).(b) バウンダリスキャンレジスタ」で述べます。

Capture-IR ではシフトレジスタに必ず IR ステータスワード（固定値）が取得されます。

(5) Shift ステート

Shift ステートでは、Capture ステートでシフトレジスタに取得したデータを TDO の方向に 1 ビットずつシフト出力するとともに、TDI から 1 ビットずつシフト入力します。

Shift ステートは、TMS に'0'を入力する限り、任意の回数だけシフトさせることができます。最終ビットは Exit1 ステートに遷移する際に出力します。

(6) Exit ステート 1

Exit1 ステートは次のステートを選択するための一時的なステートです。

(7) Pause ステート

Pause ステートは TMS に'0'を入力する限り、シフト動作を一時停止させることができます。

(8) Exit ステート 2

Exit2 ステートは次のステートを選択するための一時的なステートです。

(9) Update ステート

Update-DR ステートでは、バウンダリスキャンセルのシフトレジスタの値をラッチ回路に取り込みます。

Update-IR ステートでは、シフトレジスタの値をインストラクションレジスタに格納します。その値が命令となり、実行されます。

4.4 レジスタ

レジスタには、インストラクションレジスタとデータレジスタがあります。インストラクションレジスタは命令コードを格納するレジスタで、データレジスタは命令により制御されるレジスタです。

データレジスタには、必須レジスタとしてバイパスレジスタとバウンダリスキャンレジスタがあり、さらにオプションレジスタとして IDCODE レジスタがあります。

レジスタの種類と機能の一部を表 6 に示します。

表 6 レジスタの種類と機能

レジスタの種類			機能
インストラクションレジスタ			命令コード (SAMPLE 命令など) を格納し、デコードする
データ レジスタ	必須 レジスタ	バイパスレジスタ	TDI から入力されたデータを通過させ TDO にバイパスさせる
		バウンダリスキャン レジスタ	各端子の状態を格納する
	オプション レジスタ	IDCODE レジスタ	デバイスの IDCODE を格納する

4.4.1 インストラクションレジスタ

インストラクションレジスタは TDI から取り込まれた命令コードを格納し、デコードするレジスタです。命令コードはメーカーによって長さや値が異なるため、レジスタのビット長もメーカーやデバイスにより異なります。

インストラクションレジスタは TAP コントローラのステートが IR ステートの際に使用されます。

4.4.2 データレジスタ

データレジスタは、命令コードによって制御されるレジスタです。データレジスタには、必須レジスタとして、バイパスレジスタ、バウンダリスキャンレジスタがあります。また、オプションレジスタとして、IDCODE レジスタなどがあります。

データレジスタは、TAP コントローラのステートが DR ステートの際に使用されます。

これらのレジスタは、命令コードごとにどのレジスタを使用するのかが指定され、指定されたレジスタは TDI と TDO の間に接続されます。

(1) 必須レジスタ

(a) バイパスレジスタ

バイパスレジスタは、TDI・TDO 間をバイパスするための 1 ビット長のレジスタです。

TDI から入力されたデータをバイパスレジスタ経由で TDO から出力させることにより、テスト経路を短縮させることができます。たとえば、複数の IC や基板のテストをする際にテストが不要なデバイスのテスト経路のバイパスに使用します。

(b) バウンダリスキャンレジスタ

バウンダリスキャンレジスタとは、デバイス内部で複数のバウンダリスキャンセルがチェーン状につながって形成するシフトレジスタのことを指します。バウンダリスキャンレジスタは汎用端子と内部ロジックの間に設置され、各端子の状態を格納します。TAP コントローラのステートが Shift ステートの場合、バウンダリスキャンセルの持つ値を TDO の方向にシフトしていきます。バウンダリスキャンセルの個数はメーカーやデバイスにより異なります。

バウンダリスキャンセルは、図 10 に示すように 1 ビットのシフトレジスタと 1 ビットのラッチによって構成されています。また、バウンダリスキャンセルには、「入力セル」と「出力セル」と「コントロールセル」があります。「入力セル」とは、バウンダリスキャンセルの信号入力を端子に接続し、信号出力を内部ロジックに接続しているものをいいます。「出力セル」とは、バウンダリスキャンセルの信号入力を逆に内部ロジックに接続し、信号出力を端子に接続したものです。「コントロールセル」は出力セルからの出力を許可するかどうかを指定するセルです。

図 11 に端子とバウンダリスキャンセルと内部ロジックの関係例を示します。汎用端子が入力ピンの場合、内部ロジックとの間に入力セルを置きます。入力ピンから入力セルに信号を入力し、内部ロジックに信号を出力します (図 11(a))。

汎用端子が出力ピンの場合、内部ロジックとの間に出力セルとコントロールセルを置きます。内部ロジックから出力セルに信号を入力し、コントロールセルが出力セルからの出力を許可する場合のみ、汎用端子に信号を出力できます (図 11(b))。また、出力ピンについてはコントロールセルを持たず、出力セルだけの構成もあります。この場合、出力セルの値が汎用端子へ常時出力されます。

汎用端子が双方向ピンの場合、内部ロジックとの間に入力セル、出力セル、コントロールセルを置きます (図 11(c))。入力セルは、通常の入力ピンと同様に汎用端子から信号を入力し、内部ロジックに信号を出力します。出力セルは、内部ロジックから信号を入力しますが、コントロールセルが出力セルからの出力を許可する場合のみ、汎用端子に信号を出力できます。

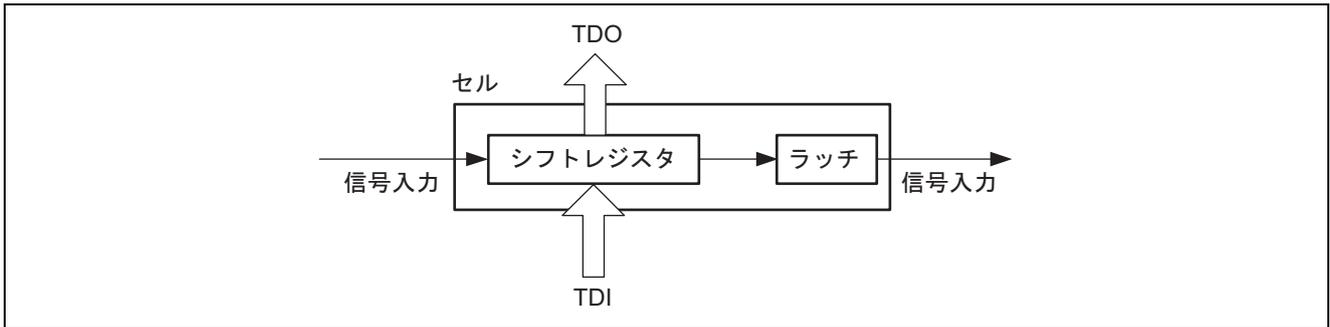


図 10 セルの構造

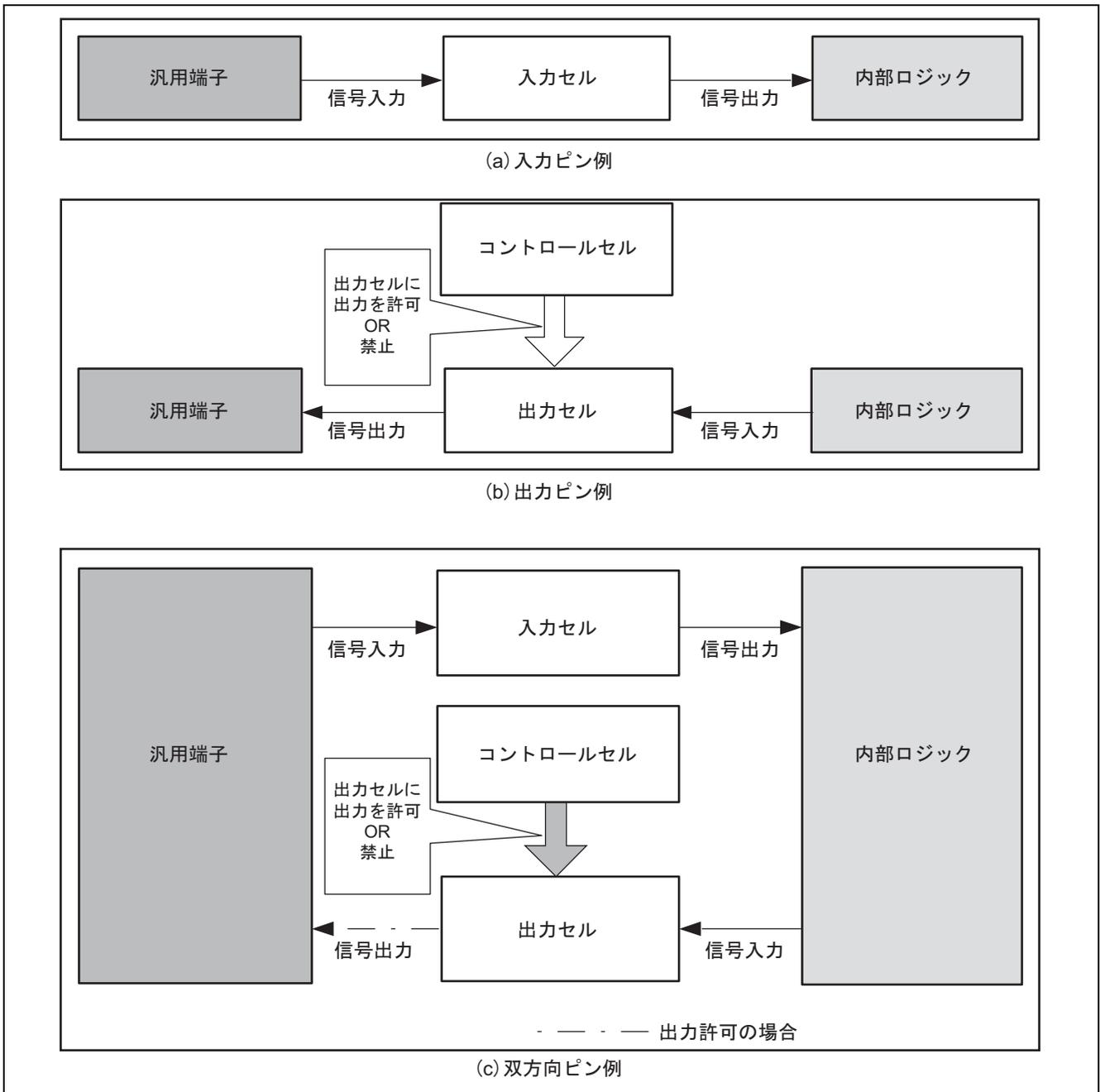


図 11 セルの動作例

(2) オプションレジスタ

(a) IDCODE レジスタ

IDCODE レジスタはデバイスの IDCODE を格納しているレジスタです。IDCODE 命令が発行されると IDCODE レジスタから TDO に固有の IDCODE が出力されます。

IDCODE レジスタは 32 ビット長のレジスタです。上位 4 ビット (31~28 ビット) は、バージョン数を表すビットで、その次の 16 ビット (27~12 ビット) はパート番号を表すビット、次の 11 ビット (11~1 ビット) は製造者 ID を表すビットです。最下位の 1 ビットは IEEE1149.1 により、'1'にすることが定められています (図 12)。

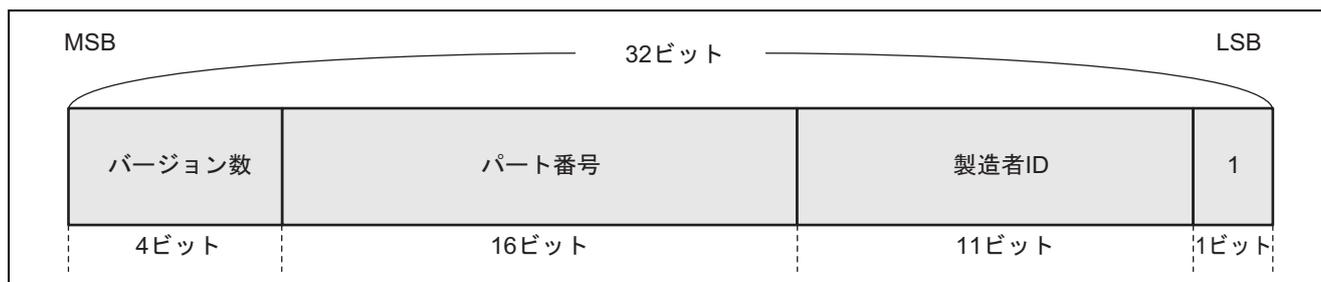


図 12 IDCODE レジスタ

4.5 コマンド

インストラクションレジスタが命令コードを格納すると IC はそれに応じた動作を始めます。

コマンドには必ず実装されていなければならない必須命令と、実装されていなくてもかまわないオプション命令があります。また、コマンドを実行する際のモードにはノーマルモードとテストモードの2つがあります。

ノーマルモードでは、デバイスの通常動作に影響を与えずにバウンダリスキャン動作を実行することができます。よって、デバイスを動作させた状態で、バウンダリスキャンセルに取得したデータを TDO から出力して確認することができます。ノーマルモードのコマンドには、必須命令として SAMPLE/PRELOAD 命令、BYPASS 命令があり、オプション命令として IDCODE 命令などがあります。

テストモードでは端子と内部ロジックを分離してテストを行うことができます。これにより内部ロジックは TAP 以外の端子から入出力を受け付けなくなります。このため内部ロジックに影響を与えずにデバイス外部のテストを行うことができます。テストモードのコマンドには、必須命令として EXTEST 命令、オプション命令では CLAMP 命令、HIGHZ 命令などがあります。

コマンドの種類と機能の一部を表 7 に示します。

表 7 コマンドの種類と機能

コマンドの種類		機能	動作モード
必須命令	SAMPLE/PRELOAD	端子状態や内部ロジックのデータをバウンダリスキャンセルに渡す	ノーマルモード
	BYPASS	入力データを TDI と TDO 間でバイパスする	ノーマルモード
	EXTEST	バウンダリスキャンセルの値を出力する	テストモード
オプション命令	IDCODE	デバイスの IDCODE を識別する	ノーマルモード
	CLAMP	バウンダリスキャンセルの値を出力し、かつバイパスする	テストモード
	HIGHZ	すべての出力端子をハイインピーダンスし、かつバイパスする	テストモード

4.5.1 必須命令

(1) SAMPLE/PRELOAD 命令

SAMPLE/PRELOAD 命令は、内部ロジックや外部回路に影響を与えずに IC の汎用端子からバウンダリスキャンセルに入力データを取り込むことや、TDI から任意のデータを取り込んでバウンダリスキャンセルにセットすることができます。ノーマルモードのコマンドであるため、バウンダリスキャンテスト中でもデバイスは通常動作が可能です。

SAMPLE 命令と PRELOAD 命令は同じ命令コードを持っており、SAMPLE/PRELOAD 命令という、共通のコマンドとして取り扱われます。TAP コントローラのステートが Capture-DR ステートの場合 SAMPLE 動作が行われ、Update-DR ステートの場合 PRELOAD 動作が行われます。

SAMPLE 動作では、バウンダリスキャンセルが入力セルの場合は IC の汎用端子の値をバウンダリスキャンセルに取得し、出力セルの場合は内部ロジックの値をバウンダリスキャンセルに取得します。

PRELOAD 動作では、バウンダリスキャンセルの値をラッチ回路に取りこんでバウンダリスキャンセルに固定します。

図 13 に SAMPLE/PRELOAD 命令のイメージを示します。

SAMPLE/PRELOAD 命令ではまず、TAP コントローラを Capture-DR ステートに遷移させます。そしてバウンダリスキャンセルが入力セルの場合、IC の汎用端子の値をバウンダリスキャンセルに取得し、出力セルの場合、内部ロジックの値をバウンダリスキャンセルに取得します (図 13(a))。任意のテストパターンをセットしたい場合、もしくはバウンダリスキャンセルの状態を確認したい場合は Shift-DR ステートに遷移し、バウンダリスキャンセルの値を必要なだけシフトして TDO から出力します (図 13(b))。最後に Update-DR ステートに遷移し、バウンダリスキャンセルの値をラッチ回路に取り込んでバウンダリスキャンセルに固定します (図 13(c))。

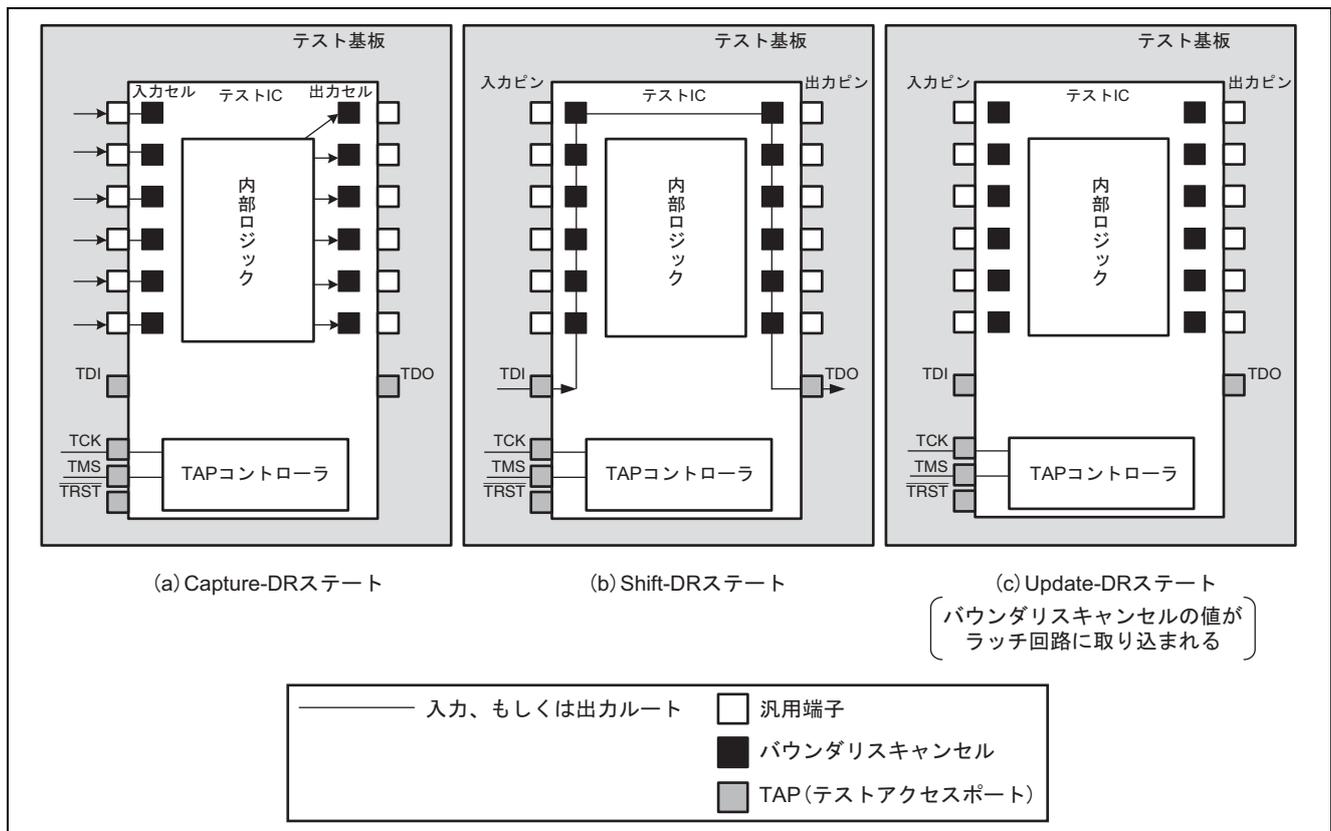


図 13 SAMPLE/PRELOAD 命令

図 14 に SAMPLE/PRELOAD 命令のタイミングチャートを示します。

タイミングチャートでは、4つの TAP (TCK、TMS、TDO、TDI) の端子状態からデバイスの動きを説明します。TAP コントローラのステートは、TCK の立ち上がりと TMS の値によって「4.3 TAP コントローラ」の図 9 のとおりに遷移しています。

4つの TAP のデフォルト端子状態は '1' としています。

図 14 の一番上にある A、B、C は、バウンダリスキャンテストの動作ごとに区間分けをしています。A 区間は、TAP コントローラのステートを初期状態に遷移させる動作を行います。B 区間は、インストラクションレジスタで命令を発行するまでの動作を行います。C 区間は、B 区間で発行した命令を実行します。

また、このタイミングチャートは、インストラクションレジスタが 8 ビットで、バウンダリスキャンセルが 15 個の場合のものです。図中の X は、任意の値です。以降のコマンドでも同様のタイミングチャートを用います。

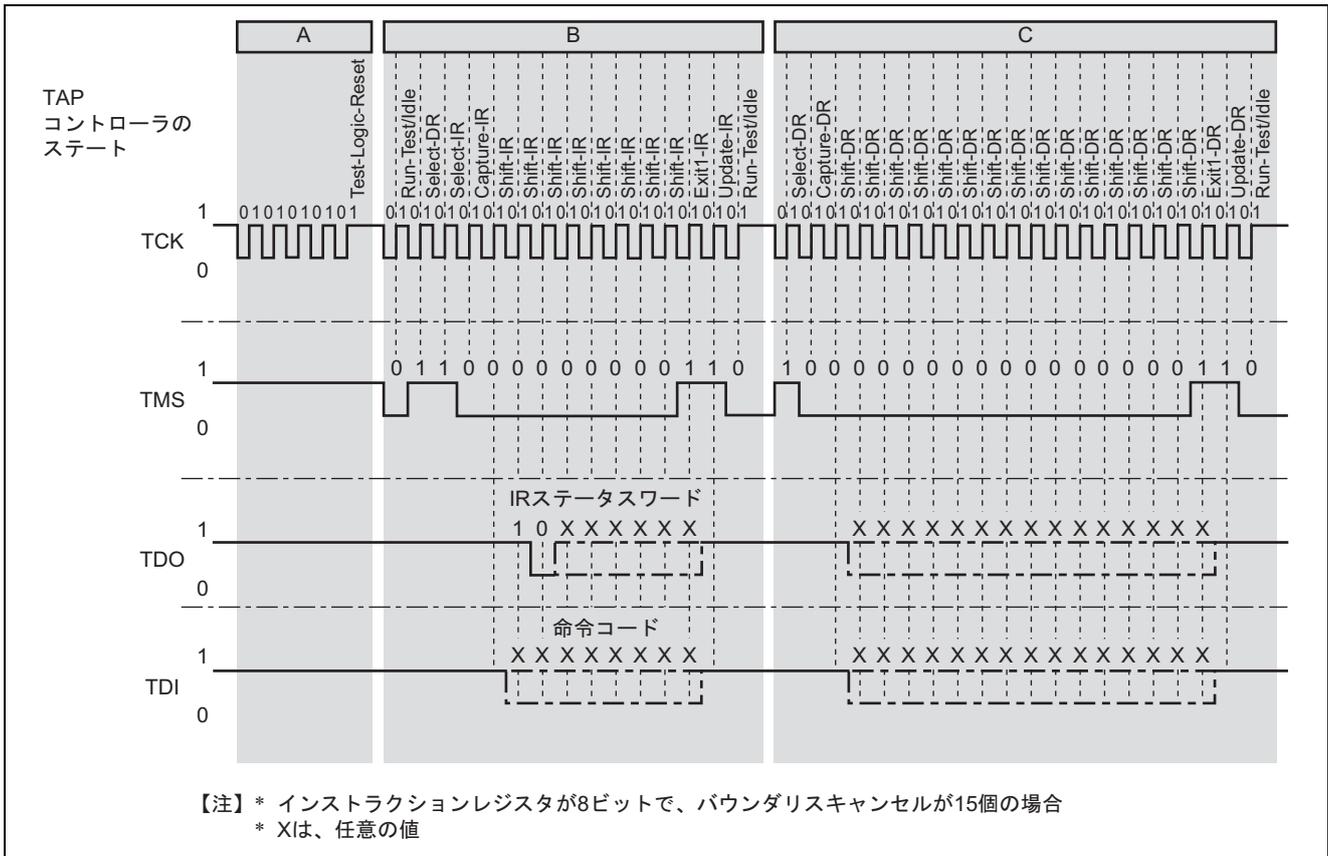


図 14 SAMPLE/PRELOAD 命令タイミングチャート

このタイミングチャートに沿って、SAMPLE/PRELOAD 命令の発行と実行について詳しく説明します。

まず A 区間では、TMS を '1' に固定したまま TCK を 5 回立ち上げます。これにより 5 回目には必ず Test-Logic-Reset ステートになります。これより前に Test-Logic-Reset ステートに遷移しても Test-Logic-Reset でループに入ります。A 区間はステートの状態を初期化するためのものなので、ステートの初期化を行う必要のない場合は省略できます。

次に B 区間では、TMS の値と TCK の立ち上がりによって TAP コントローラのステートを、Run-Test/Idle ステート→Select-DR ステート→Select-IR ステート→Capture-IR ステート→Shift-IR ステート (8 回) →Exit1-IR ステート→Update-IR ステート→Run-Test/Idle ステートの順で遷移させます。この遷移によって、SAMPLE/PRELOAD 命令が実行されます。

この遷移での Capture-IR ステートでは、インストラクションレジスタから IR ステータスワードをキャプチャします。

Shift-IR ステートでは、TCK 立ち下がりと同期して TDO から IR ステータスワードが 1 ビット出力され、TDI から SAMPLE/PRELOAD 命令コードが 1 ビット入力されます。これらの動作がインストラクションレジスタのビット数分 (今回は 8 回) 繰り返されます。TDI、TDO から入出力される最後のビットは Shift-IR ステートを抜ける際の TCK の立ち下がり時に入出力されます。

Update-IR ステートでは、インストラクションレジスタにセットされた SAMPLE/PRELOAD 命令コードがデコードされて、TDI と TDO にバウンダリスキャンレジスタを接続します。

最後に C 区間では、Run-Test/Idle ステート→Select-DR ステート→Capture-DR ステート→Shift-DR ステート (15 回) →Exit1-DR ステート→Update-DR ステート→Run-Test/Idle ステートの順で遷移させます。この遷移によって SAMPLE 動作と PRELOAD 動作が行われます。

Capture-DR ステートでは、入力セルには汎用入力ピンの値が、出力セルには内部ロジックの値がそれぞれセットされます。これが、SAMPLE 動作となります。

次に、Shift-DR ステートでは、バウンダリスキャンセルの数 (今回は 15 回) だけセットしたいデータを TDI からシフト入力していきます。

Update-DR ステートでは、バウンダリスキャンセルの値をラッチ回路に取り込みます。これが、PRELOAD 動作となります。

また、C 区間を繰り返すことにより、常に SAMPLE/PRELOAD 動作をして入力ピンと内部ロジックの値をバウンダリスキャンセルに取り込むことも可能です。

(2) BYPASS 命令

BYPASS 命令は TDI と TDO の間に 1 ビットのバイパスレジスタを挿入して TDI と TDO をバイパスします。これにより、複数デバイスを使用する際にテストに関与しない経路を短縮することができます。ノーマルモードのコマンドのため、バウンダリスキャンテスト中もバウンダリスキャンと関係なく、通常動作が行われています。また、バイパスレジスタ以外のレジスタには影響を与えません。

図 15 に BYPASS 命令のイメージを示します。

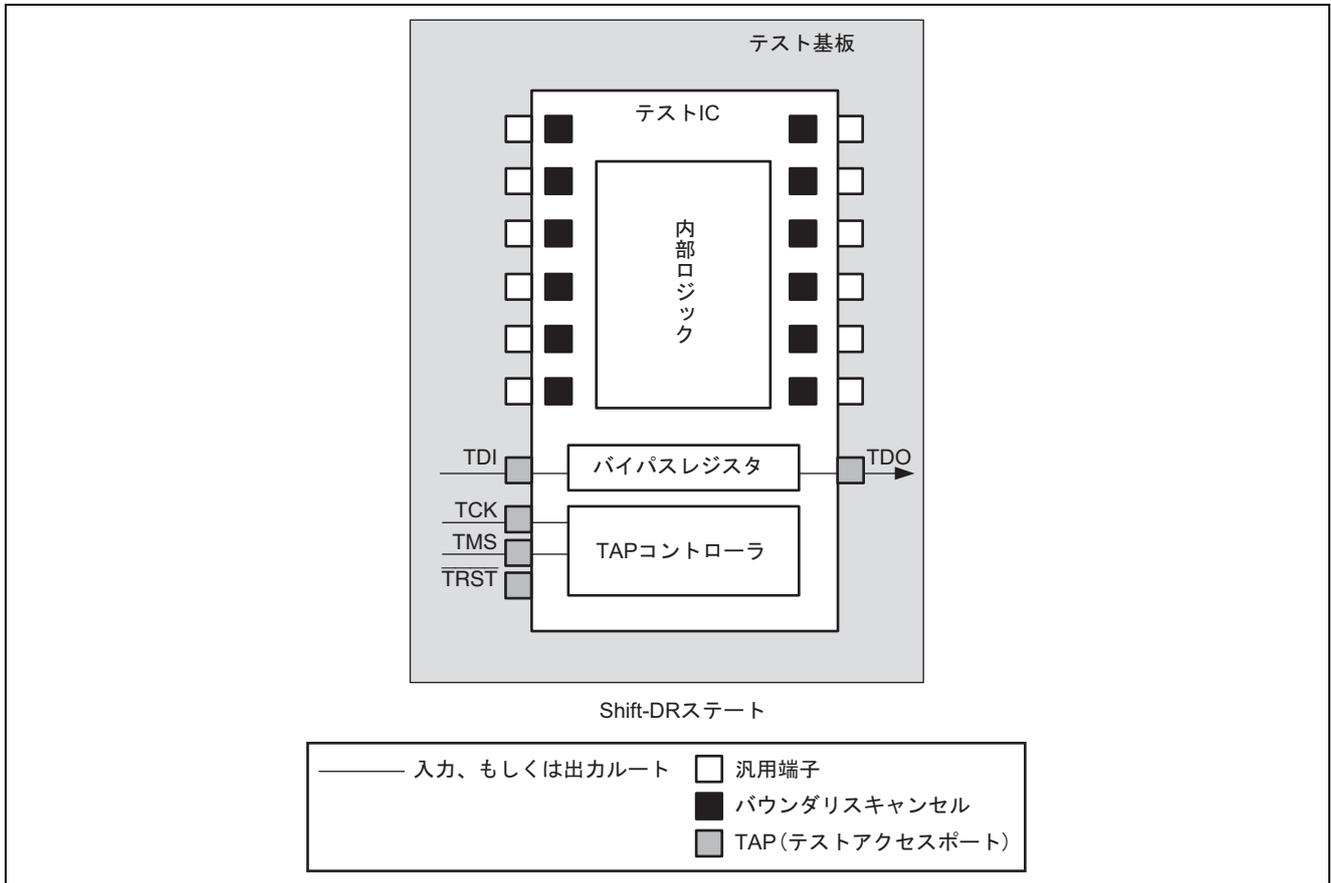


図 15 BYPASS 命令

図 16 に BYPASS 命令のタイミングチャートを示します。

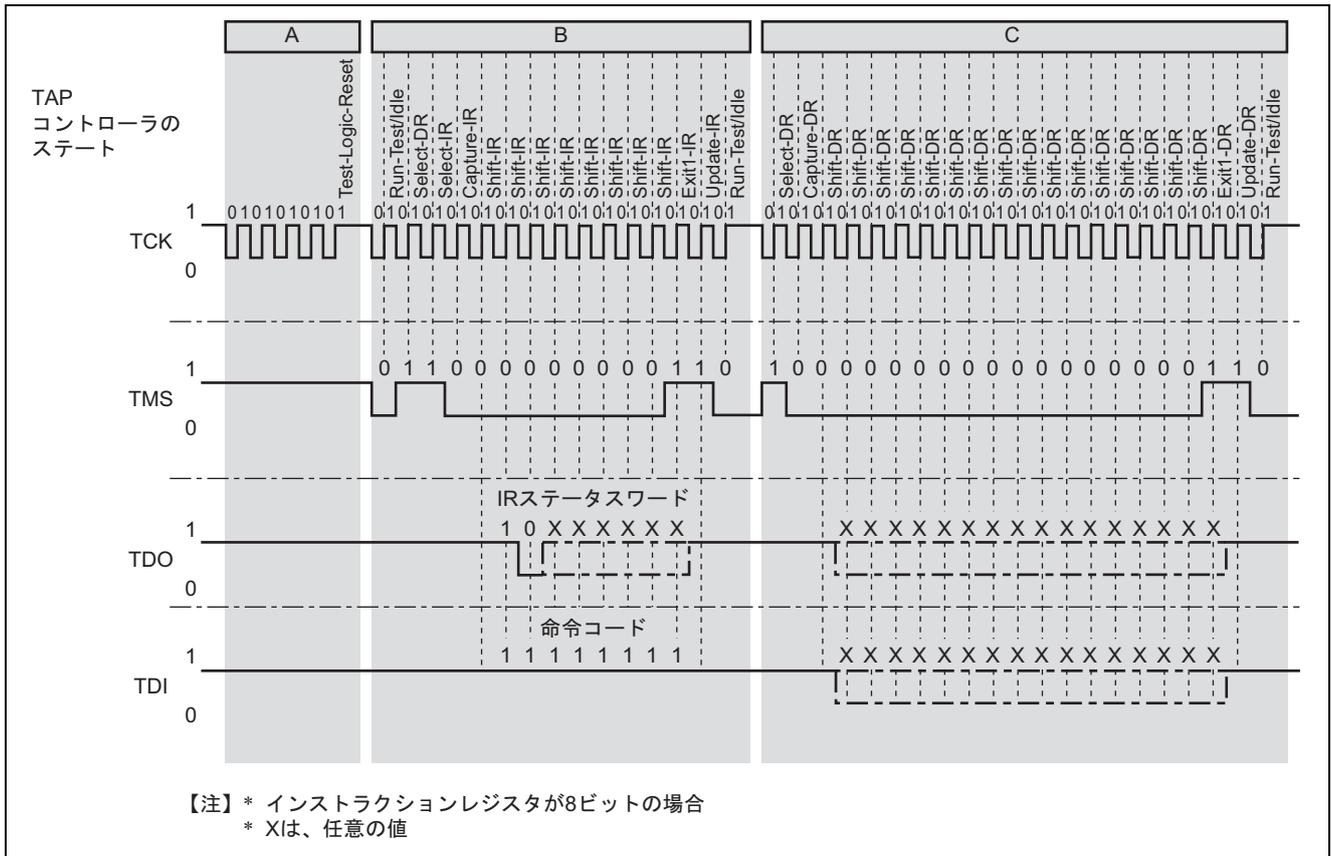


図 16 BYPASS 命令タイミングチャート

まず A 区間では、ステータスの状態を初期化するための動作を行います。

次に B 区間では、BYPASS 命令コードをインストラクションレジスタにセット、デコードを行い、TDI と TDO にバイパスレジスタを接続します。なお、BYPASS 命令はレジスタのビット長によらず全ビット 1 に規定されていますが、追加でその他のコードを設定することもできます。

最後に C 区間では、Run-Test/Idle ステータス→Select-DR ステータス→Capture-DR ステータス→Shift-DR ステータス (任意の回数) →Exit1-DR ステータス→Update-DR ステータス→Run-Test/Idle ステータスの順で遷移させます。この遷移によって BYPASS 動作を行います。

BYPASS 命令では、1 ビットのバイパスレジスタを通して TDI から TDO へとシフトさせることが主な機能であるため、Capture-DR や Update-DR での動作は意味を持ちません。

この遷移での Shift-DR ステータスでは、バイパスさせるデータを TCK の立ち上がり時に TDI からバイパスレジスタに 1 ビットずつ入力し、TDO からバイパスレジスタのデータを 1 ビットずつ出力させます。

(3) EXTEST 命令

EXTEST 命令は IC 外部の接続（プリント基板や半田の導通）をテストするための命令です。この命令は内部ロジックと分離されて行われます。テストモードのコマンドのため、バウンダリスキャンテスト中のデバイスは通常動作しません。

図 17 に EXTEST 命令のイメージを示します。

EXTEST 命令を行うためには、事前に SAMPLE/PRELOAD 命令によりバウンダリスキャンセルにデータを取り込んでおきます。そして EXTEST 命令を発行し、IC の汎用出力ピンにデータを出力します。EXTEST 動作は Update-IR ステータス通過した時点から行われます。

また、EXTEST 命令発行後に連続して、Capture-DR ステータス、Shift-DR ステータス、Update-DR ステータスに遷移することでバウンダリスキャンセルの内容を変更して出力させることも可能です。この場合の出力の更新は Update-DR ステータスで行われます。

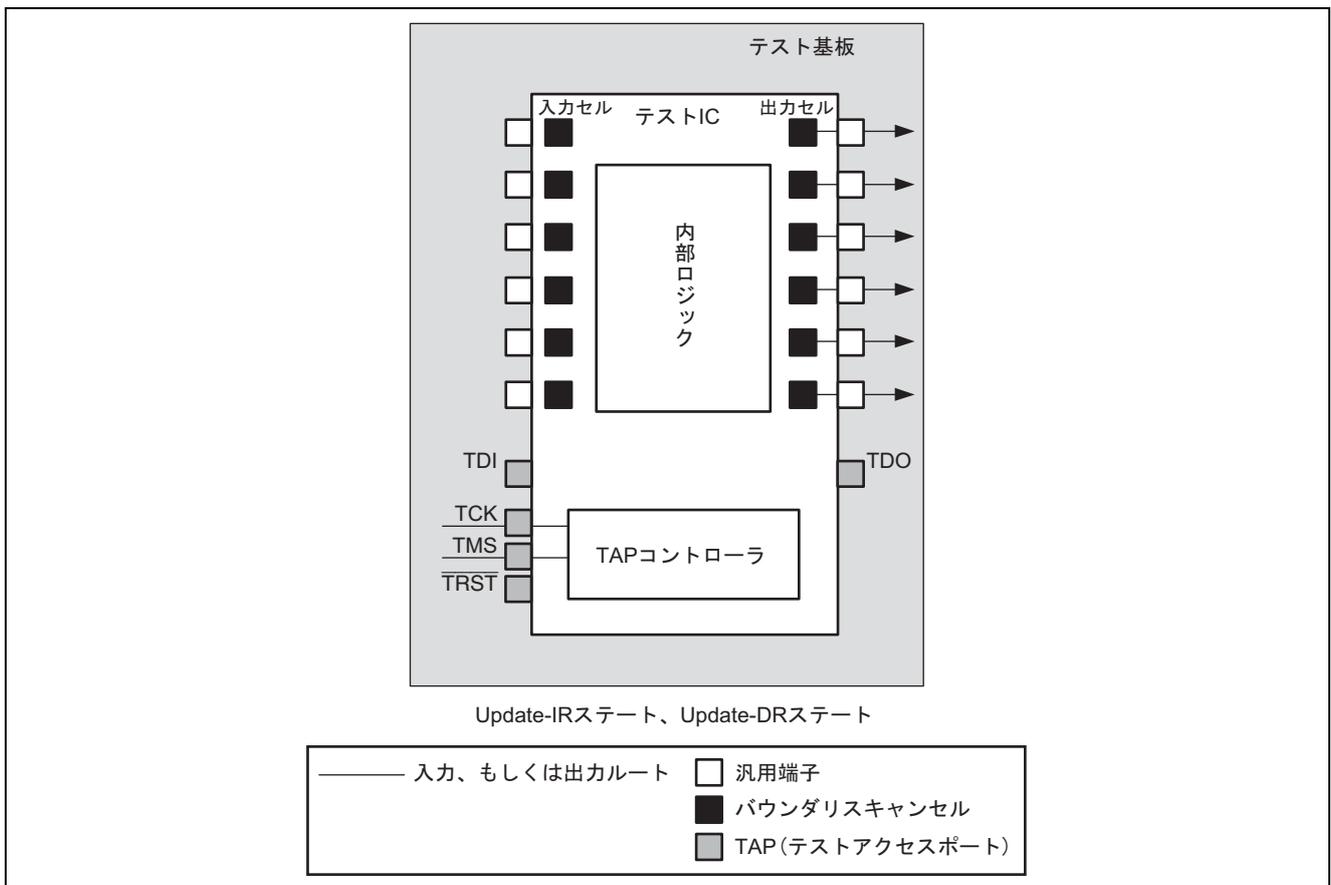


図 17 EXTEST 命令

図 18 に EXTEST 命令のタイミングチャートを示します。

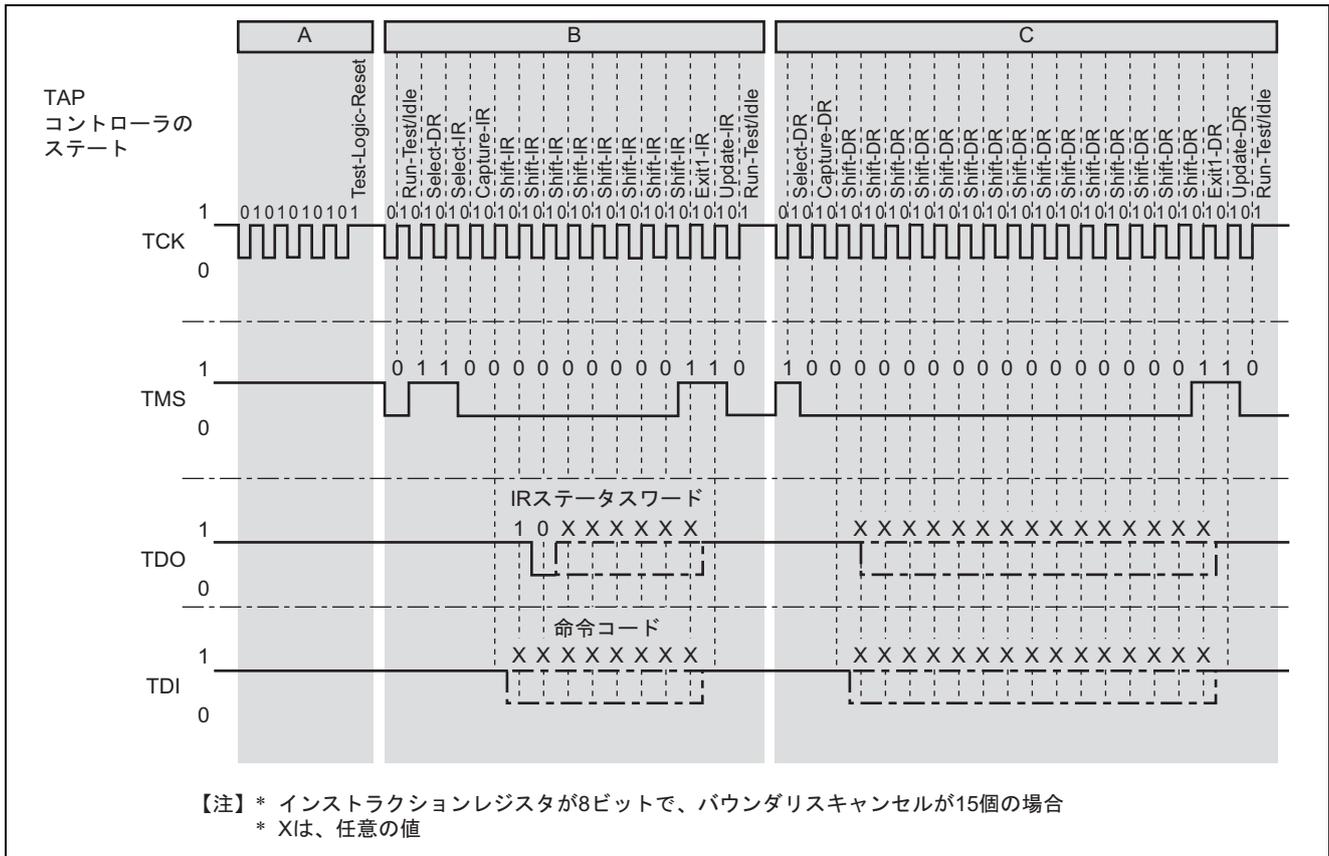


図 18 EXTEST 命令タイミングチャート

まず A 区間ではステートの状態を初期化するための動作を行います。

次に B 区間では、EXTEST 命令コードをインストラクションレジスタにセット、デコードを行い、TDI と TDO にバウンダリスキャンレジスタを接続します。EXTEST 命令が発行された時点で、事前に SAMPLE/PRELOAD 命令でセットされたバウンダリスキャンセルの値が IC の汎用出力ピンに出力されます (EXTEST 動作)。EXTEST 命令がキャンセルされるまで、EXTEST 動作は続きます。

最後に C 区間では、Run-Test/Idle ステート→Select-DR ステート→Capture-DR ステート→Shift-DR ステート (15 回) →Exit1-DR ステート→Update-DR ステート→Run-Test/Idle ステートの順で遷移させます。この遷移によってバウンダリスキャンセルの値を再設定し、汎用出力ピンからの出力値が更新できます。

この遷移での Capture-DR ステートでは、入力セルには汎用入力ピンの値が、出力セルには内部ロジックの値がそれぞれセットされます。

次に、Shift-DR ステートでは、バウンダリスキャンセルの数だけセットしたいデータを TDI からシフト入力していきます。

Update-DR ステートでは、バウンダリスキャンセルの値をラッチ回路に取り込みます。Update-IR ステートを通過した時点で、汎用出力ピンの出力値が更新されます。

また、C 区間を繰り返すことにより、常に EXTEST 動作をしながら、入力ピンと内部ロジックの値をバウンダリスキャンセルに取り込み、汎用出力ピンの出力値を更新することも可能です。

4.5.2 オプション命令

(1) IDCODE 命令

IDCODE 命令は、デバイスの IDCODE を識別するための命令です。IDCODE 命令はオプション命令のため、IDCODE レジスタ内蔵の IC でのみ使用できます。ノーマルモードのコマンドのため、バウンダリスキャンテスト中もバウンダリスキャンと関係なく、通常動作が行われています。

図 19 に IDCODE 命令のイメージを示します。

IDCODE 命令を行うと、Capture-DR ステートで IDCODE レジスタから IDCODE が取り込まれ、Shift-DR ステートで TDO から出力されます。Shift-DR ステートを 32 回 (IDCODE のビット分) 繰り返すことで、IDCODE を取得することができます。

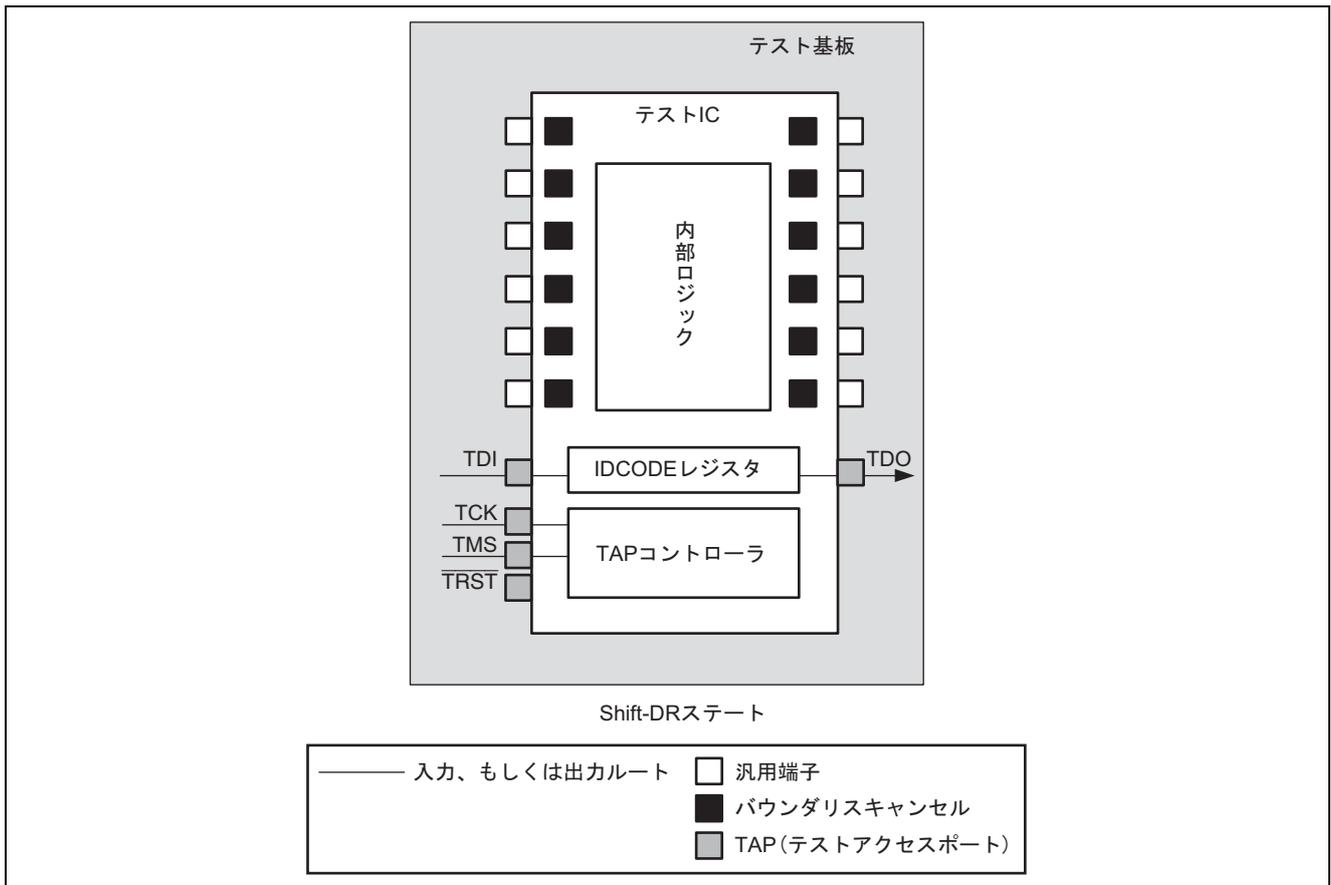


図 19 IDCODE 命令

図 20 に IDCODE 命令のタイミングチャートを示します。

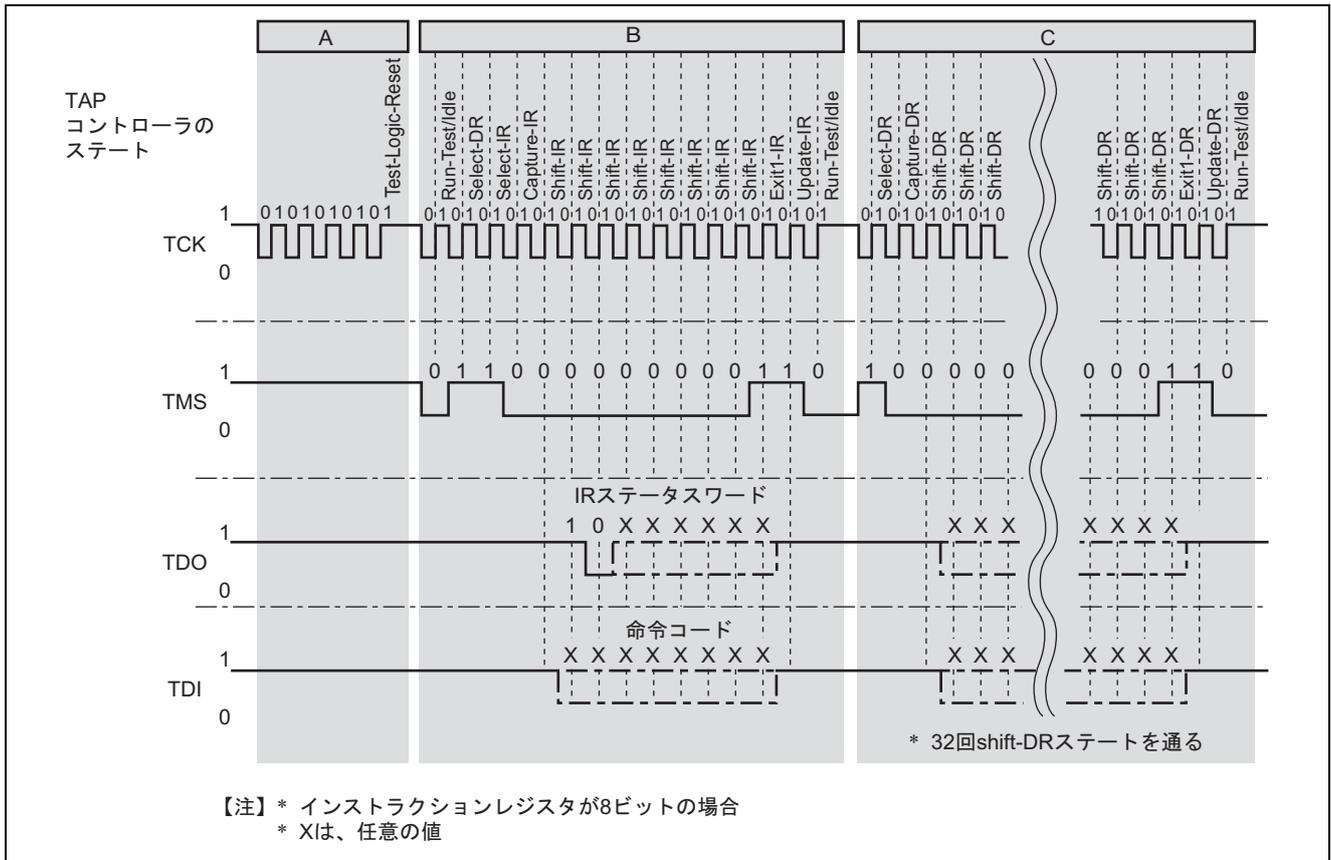


図 20 IDCODE 命令タイミングチャート

まず A 区間ではステータスの状態を初期化するための動作を行います。

次に B 区間では、IDCODE 命令コードをインストラクションレジスタにセット、デコードを行い、TDI と TDO に IDCODE レジスタを接続します。

最後に C 区間では、Run-Test/Idle ステータス→Select-DR ステータス→Capture-DR ステータス→Shift-DR ステータス (32 回) →Exit1-DR ステータス→Update-DR ステータス→Run-Test/Idle ステータスの順で遷移させます。

この遷移での Capture-DR ステータスでは、IDCODE レジスタからシフトレジスタにデバイス固有の 32 ビットの IDCODE をセットします。

次に、Shift-DR ステータスでは、IDCODE を TDO から 1 ビットずつ出力させます。

Update-DR ステータスでは、特にすることはありません。

(2) CLAMP 命令

CLAMP 命令はバウンダリスキャンセルの値を汎用端子に出力し、かつバイパスさせる命令です。テストモードのコマンドのため、バウンダリスキャンテスト中のデバイスは通常動作しません。

通常、テスト対象外の IC はバイパスしてテスト経路を短縮させますが、バイパス動作対象の IC の汎用出力ピンを特定のレベルに固定したい場合があります。その場合、BYPASS 命令はノーマルモード命令であるため、汎用出力ピンの値が変わってしまう可能性があります。CLAMP 命令は、IC の動作を停止させて汎用出力ピンの値を変えずにバイパス動作を行うことができます。

図 21 に CLAMP 命令のイメージを示します。

CLAMP 命令では、まずあらかじめ SAMPLE/PRELOAD 命令によりバウンダリスキャンセルに値を取得させます。その後、CLAMP 命令を発行し、汎用出力ピンから端子状態を出力させながらバイパス動作を行います。

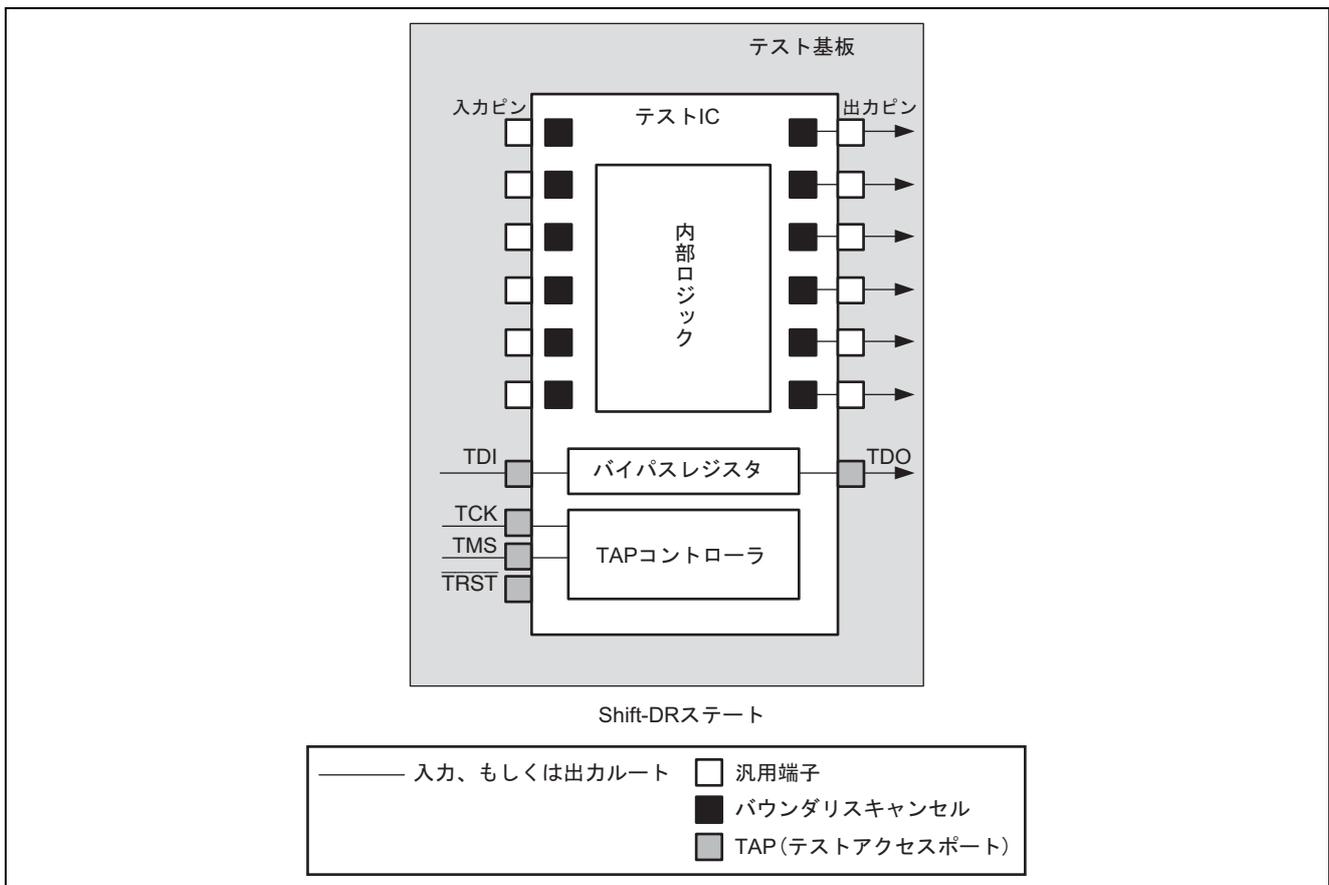


図 21 CLAMP 命令

図 22 に CLAMP 命令のタイミングチャートを示します。

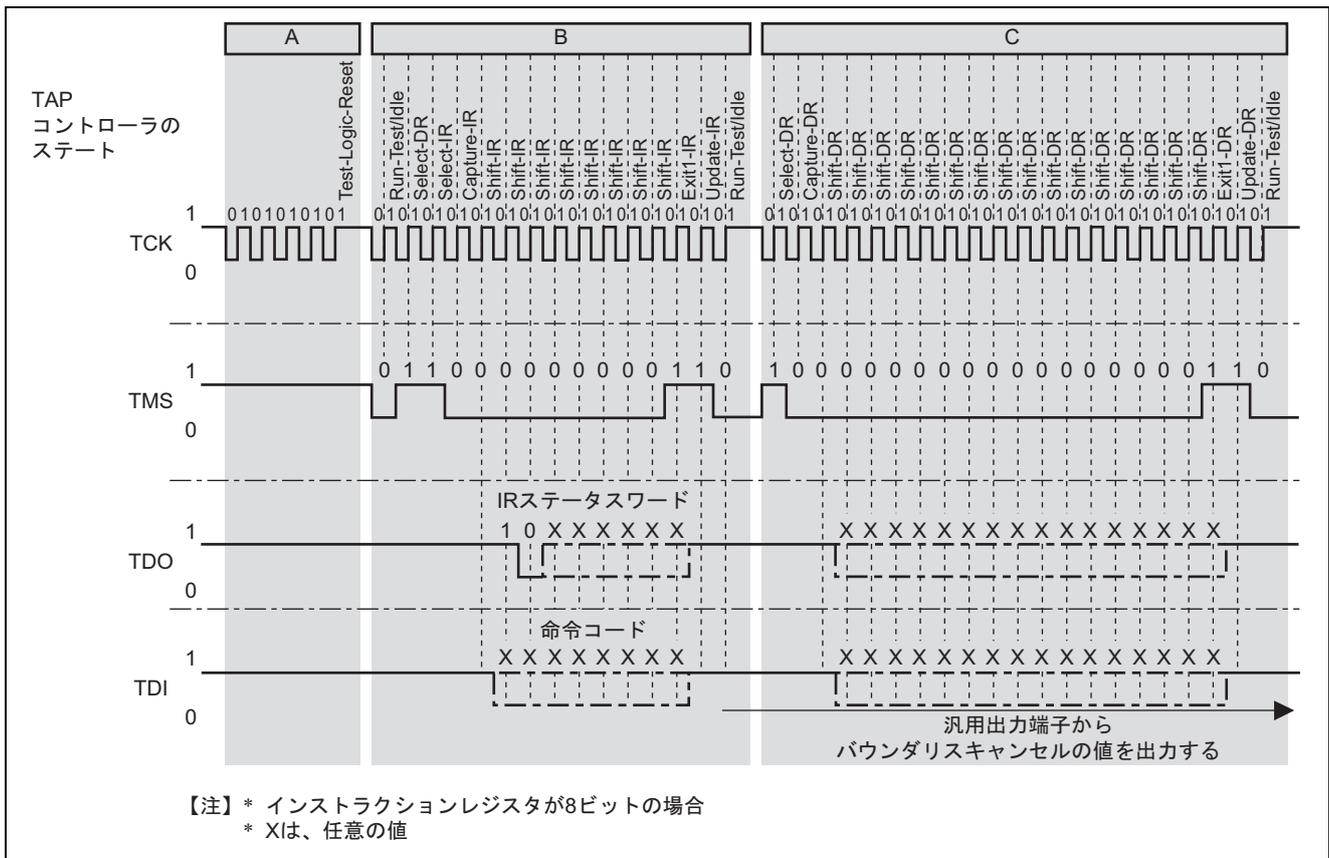


図 22 CLAMP 命令タイミングチャート

まず A 区間ではステータスの状態を初期化するための動作を行います。

次に B 区間では、CLAMP 命令コードをインストラクションレジスタにセット、デコードを行い、TDI と TDO にバイパスレジスタを接続します。CLAMP 命令を発行した時点で、事前に SAMPLE/PRELOAD 命令によってバウンダリスキャンにセットした値を汎用出力ピンより出力し続けます。

最後に C 区間では、Run-Test/Idle ステータス→Select-DR ステータス→Capture-DR ステータス→Shift-DR ステータス (任意の回数) →Exit1-DR ステータス→Update-DR ステータス→Run-Test/Idle ステータスの順で遷移させます。

CLAMP 命令では、1 ビットのバイパスレジスタを通して TDI から TDO へとシフトさせることが主な機能であるため、Capture-DR や Update-DR での動作は意味を持ちません。

この遷移での Shift-DR ステータスでは、バイパスさせるデータを TCK の立ち上がりに応じて TDI からバイパスレジスタに 1 ビットずつ入力し、TDO からバイパスレジスタのデータを 1 ビットずつ出力させます。

(3) HIGHZ 命令

HIGHZ 命令は、バウンダリスキャンに対応した汎用出力ピンをハイインピーダンス状態にしなから、TDI と TDO をバイパスします。端子をハイインピーダンス状態にすることで、他デバイスへ影響を与えることなくテスト経路を短縮したい場合に使用します。テストモードのコマンドのため、バウンダリスキャンテスト中のデバイスは通常動作しません。

図 23 に HIGHZ 命令のイメージを示します。

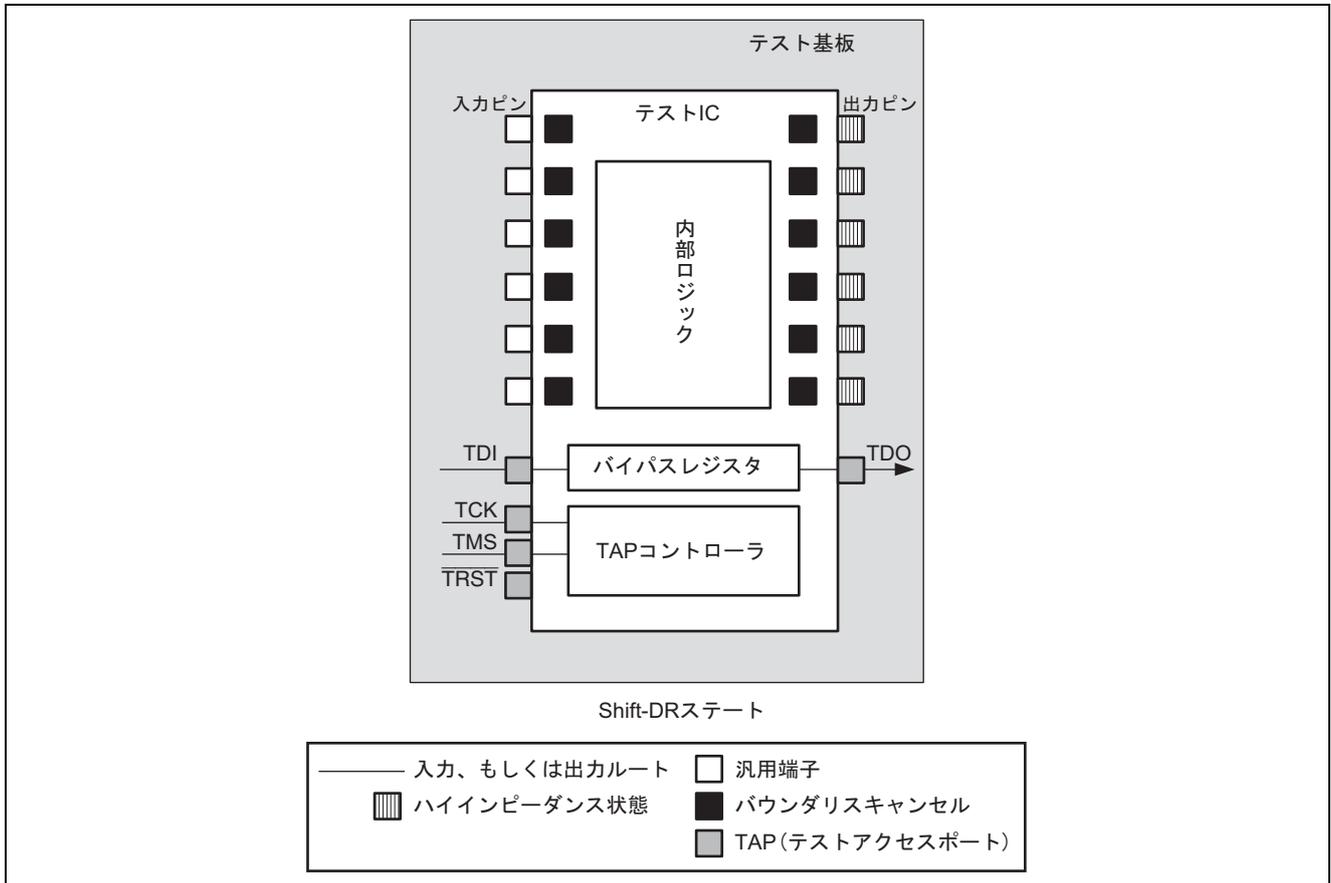


図 23 HIGHZ 命令

図 24 に HIGHZ 命令のタイミングチャートを示します。

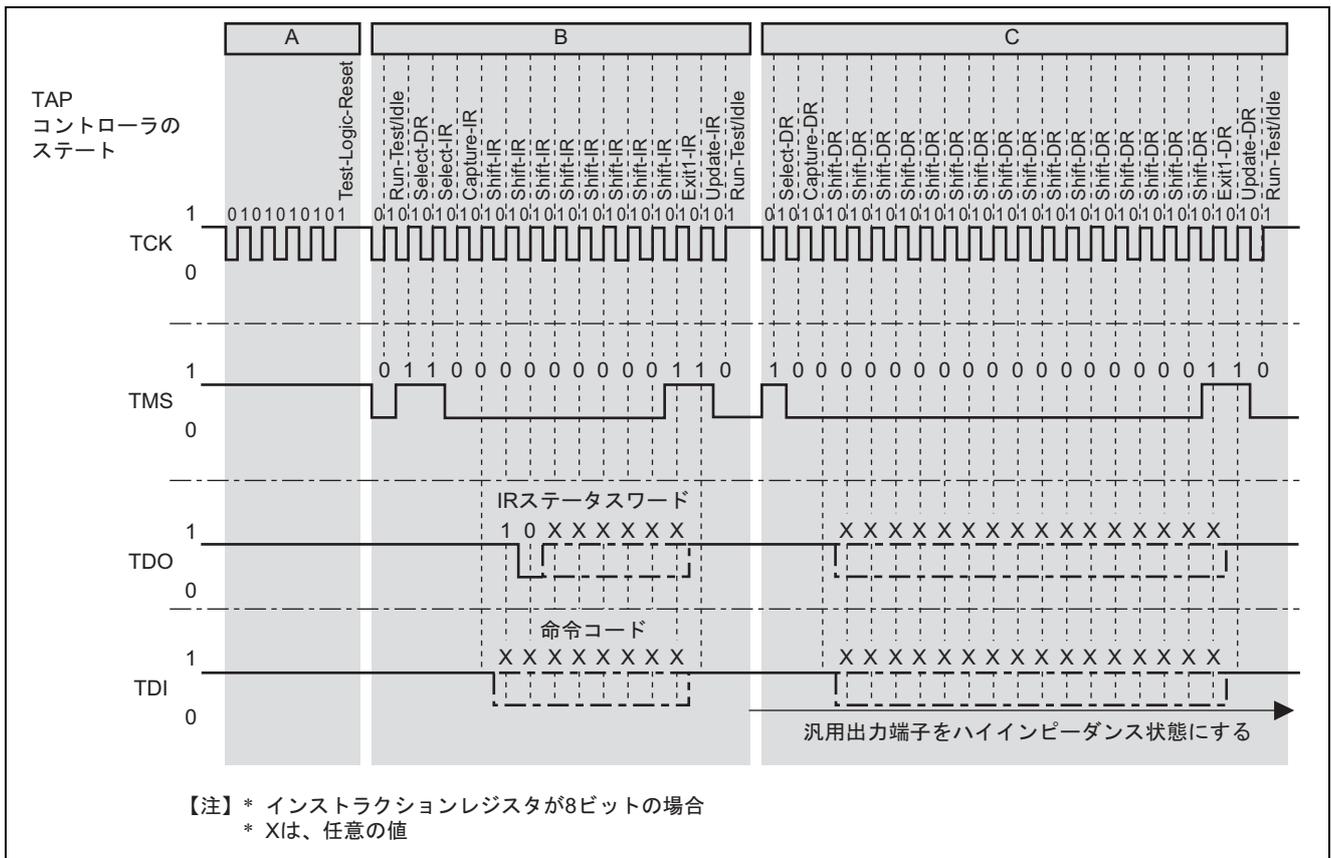


図 24 HIGHZ 命令タイミングチャート

まず A 区間ではステートの状態を初期化するための動作を行います。

次に B 区間では、HIGHZ 命令コードをインストラクションレジスタにセット、デコードを行い、TDI と TDO にバイパスレジスタを接続します。HIGHZ 命令を発行した時点で、汎用出力ピンをハイインピーダンス状態にします。

最後に C 区間では、Run-Test/Idle ステート→Select-DR ステート→Capture-DR ステート→Shift-DR ステート (任意の回数) →Exit1-DR ステート→Update-DR ステート→Run-Test/Idle ステートの順で遷移させます。

HIGHZ 命令では、1 ビットのバイパスレジスタを通して TDI から TDO へとシフトさせることが主な機能であるため、Capture-DR や Update-DR での動作は意味を持ちません。

この遷移での Shift-DR ステートでは、バイパスさせるデータを TCK の立ち上がりに応じて TDI からバイパスレジスタに 1 ビットずつ入力し、TDO からバイパスレジスタのデータを 1 ビットずつ出力させます。

5. BSDL ファイル

バウンダリスキャンテストに必要なファイルである BSDL (Boundary Scan Description Language) ファイルについて簡単に説明します。実際のシステムを作る際の参考としてください。

BSDL は米国国防省が提唱して IEEE1149.1b として標準化されています。BSDL は VHDL (Very high speed integrated circuit Hardware Description Language) というハードウェア言語を基本とした言語です。BSDL にはテスト対象 IC に JTAG でアクセスするために必要な情報が記述されます。

BSDL ファイルは、バウンダリスキャンに対応した IC を製造しているメーカーが提供しています。

5.1 使用できる文字、記号

5.1.1 使用できる文字、記号

BSDL ファイルで使用できる文字、記号は以下のとおりです。また、すべて半角で記述してください。

アルファベット (大文字小文字の区別はありません)

数字 (0~9)

特殊文字 (" & ' () * , - . : ; < = > _)

区切り文字 (スペース、タブ文字、改行、改ページ、ラインフィード)

5.1.2 単語

BSDL で使用する単語はアルファベット、数字、アンダーバー (_) によって構成されます。文字数の上限はありません。アンダーバーは単語の最後に使用することはできません。

5.1.3 コメント

-- (ハイフン 2 個) で行の最後までをコメントとすることができます。-- の後の文章は実質的な内容に反映されません。

5.1.4 文字列

文字列は " (ダブルクォーテーションマーク) で囲みます。また、文字列は & (アンドマーク) で連結することができます。

5.1.5 予約語

BSDL と VHDL には予約語があり、予約語はユーザが任意に使用する単語として使えません。ただし、コメントでは使用できます。

(1) BSDL 予約語

BSDL の予約語には以下のようなものがあります。予約語の一部を例としてあげます。

“EXTEST”	“HIGHZ”	“CLAMP”	“SAMPLE”
“ID_CODE”	“INTEST”	“RUNBIST”	“BYPASS” “USER_CODE”
“CLOCK”	“BOUNDARY”	“BOUNDARY_LENGTH”	“CAPTURES”
“AT_PINS”	“BIDIR” “BIDIR_IN”	“BIDIR_OUT”	“BORTH”
“STD_1149_*”	“OUTPUT2”	“OUTPUT3”	“ID_CODE_REGISTER”
“OBSERVING”	“OBSERVE_ONLY”	“INTERNAL”	“CONTROL” “CONTROLR”
“BOUNDARY_REGISTER”		“BSCAN_INST”	“BSDL_EXTENSION”
“PHYSICAL_PIN_MAP”		“PI”	“PIN_MAP” “PINM_MAP_STRING”
“PO”		“PORT_GROUPING”	“COMPLIANCE_PATTERN”
“COMPORNENNT_CONFORMANCE”		“TAP_SCAN_IN”	“X”
“Z”	“ZERO”	“ONE”	
“BC_0”	“BC_1”	“BC_2”	“BC_3” “BC_4” “BC_5” “BC_6” “BC_7” “BC_8” “BC_9” “BC_10”
“BC_11”	“BC_12”	“BC_13”	“BC_14” “BC_15” “BC_16” “BC_17” “BC_18” “BC_19” “BC_20”
“BC_21”	“BC_22”	“BC_23”	“BC_24” “BC_25” “BC_26” “BC_27” “BC_28” “BC_29” “BC_30”

(2) VHDL 予約語

VHDL の予約語には以下のようなものがあります。予約語の一部を例としてあげます。

“ABS”	“ACCES”	“AFTER”	“all”	“AND”	“attribute”	“bit”
“BLOCK”	“body”	“buffer”	“BUS”	“CASE”	“COMPONENT”	“constant”
“ELSE”	“end”	“entity”	“EXIT”	“FALSE”	“FILE”	“FOR”
“FUNCTION”	“generic”	“GROUP”	“IF”	“in”	“INTERNAL”	“inout”
“is”	“LIBRARY”	“linkage”	“LITERAL”	“MAP”	“NAND”	“HEW”
“NEXT”	“NOR”	“NOT”	“NULL”	“of”	“out”	“package”
“port”	“PROCESS”	“REGISTER”	“record”	“RETURN”	“SELECT”	“signal”
“string”	“to”	“true”	”type”	“use”	“WAIT”	“WHEN”
“WHILE”	“WITH”	“XNOR”	“XOR”			

5.2 項目

BSDL ファイルに記述する主な項目は以下のとおりです。

エンティティ記述
パラメータ記述
論理ポート記述
standard use 文
スタンダード版指定
端子割り当て
TAP 信号定義
(compliance 文)
インストラクションレジスタ記述
(オプションレジスタ定義)
(レジスタ入出力記述)
バウンダリスキャンレジスタ記述

括弧書きの項目は必要に応じて記述する項目であり、記述しなくても文法上の間違いにはなりません。また、これらすべての項目（括弧書き項目も含む）は上記の順番で書かなくてはなりません。

5.2.1 エンティティ記述

エンティティ記述では、エンティティに任意の名前を付けます。"entity エンティティ名 is"の後にエンティティ記述以外の項目を入れ、最後は"end エンティティ名"で閉じます。

例)

```
entity SAMPLE_DEVICE is
    エンティティ記述以外の項目 (パラメータ記述など)
end SAMPLE_DEVICE
```

上記の例ではエンティティ名を SAMPLE_DEVICE としています。

5.2.2 パラメータ記述

パラメータ記述では、論理ポートに物理ポート名を割り付けます。パッケージがある場合はパッケージを参照することにより割り付けを行います。

例)

```
generic (PHYSICAL_PIN_MAP: string:= "SAMPLE_PACK");
```

上記の例では、パッケージ名を SAMPLE_PACK としています。

5.2.3 論理ポート記述

論理ポート記述では、端子の名前とその性質を設定します。端子の並びは順不同ですが、すべての端子が設定されなくてはなりません。

性質には、in (入力)、out (出力)、inout (入出力双方向)、linkage (電源、アナログ信号、グラウンド、ノンコネクタなどのデジタル以外の信号) があります。

例)

```
port(
  PX1 : in    bit;
  PX2 : linkage bit
);
```

上記の例では、端子の名前を PX1、および PX2 としています。

5.2.4 standard use 文

standard use 文では、使用する standard VHDL package を指定します。

STD_1149_1_1994 であれば IEEE std. 1149.1-1994 に対応します。

例)

```
use STD_1149_1_1994.all;
```

5.2.5 スタンダード版指定

スタンダード版指定では、デバイスが対応するスタンダードの版数を指定します。

STD_1149_1_1993 であれば IEEE std. 1149.1-1993 に対応します。

例)

```
attribute COMPONENT_CONFORMANCE of SAMPLE_DEVICE: entity is "STD_1149_1_1993";
```

5.2.6 端子割り当て

端子割り当てでは、デバイスのピン名に対応したピン番号を設定します。

例)

```
attribute PIN_MAP of SAMPLE_DEVICE: entity is PHYSICAL_PIN_MAP;
constant SAMPLE_PACK: PIN_MAP STRING :=
  "PX1    : 1," &
  "PX2    : 2";
```

5.2.7 TAP 信号定義

TAP 信号定義では、TAP 信号 (TDI, TDO, TCK, TMS, TRST) の定義を行います。

TCK の定義では、最高動作周波数を設定します。また、BOTH では TCK が HIGH でも LOW でも信号を停止することが可能です。

例)

```
attribute TAP_SCAN_CLOCK of TCK    : signal is (20.0e6, BOTH);
attribute TAP_SCAN_IN   of TDI     : signal is true;
attribute TAP_SCAN_MODE of TMS     : signal is true;
attribute TAP_SCAN_OUT  of TDO     : signal is true;
attribute TAP_SCAN_RESET of TRST   : signal is true;
```

上記の例では、最高動作周波数を 20MHz としています。

5.2.8 compliance 文

compliance 文では、安全にバウンダリスキャンテストを行うために、信号パターンを制限します。

バウンダリスキャンテストを行う際は、ここで設定した端子は設定したパターンどおりの値を入力しつづけます。

例)

```
attribute COMPLIANCE_PATTERNS of SAMPLE_DEVICE:entity is
    "(PX6, PX7) (11)";
```

上記の例では PX6 ピンに 1 を、PX7 ピンに 1 を入力しつづけます。

5.2.9 インストラクションレジスタ記述

インストラクションレジスタ記述では、インストラクションレジスタの設定（命令コードの長さ）、各命令のコード、IR ステータスワードの定義を行います。Capture-IR ステートの際のビットパターンでの下位 2 ビットは必ず 01 です。なお、BYPASS 命令はレジスタのビット長によらず全ビット 1 に規定されていますが、追加でその他のコードを設定することもできます。

例)

```
attribute INSTRUCTION_LENGTH of SAMPLE_DEVICE: entity is 4
attribute INSTRUCTION_OPCODE of SAMPLE_DEVICE: entity is
    "BYPASS (1111)," &
    "EXTEST (0000)," &
    "SAMPLE (1000)," &
    "CLAMP (1010)," &
    "HIGHZ (1011)," &
    "IDCODE (0101)";
attribute INSTRUCTION_CAPTURE of SAMPLE_DEVICE: entity is "1101";
```

上記の例では、命令コードの長さが 4 ビットで、IR ステータスワードは"1101"となっています。

5.2.10 オプションレジスタ定義

オプションレジスタの定義では、IDCODE レジスタの定義を行います。

例)

```
attribute IDCODE_REGISTER of SAMPLE_DEVICE: entity is --ID code H'04632B97
    "0000" & -- 4-bit version number
    "0100011000110010 " & -- 16-bit part number
    "10111001011" & -- 11-bit identity of the manufacturer
    "1"; -- Required by IEEE Std 1149.1
```

上記の例では、IDCODE を H'04632B97 と設定しています。

これをバイナリに直すと、0000 0100 0110 0011 0010 1011 1001 0111 となります。

この上位 4 桁 (0000) はバージョン数を表します。次の 16 桁 (0100011000110010) はパート番号を表し、その次の 11 桁 (10111001011) は製造者 ID です。また、最後の 1 桁 (1) は IEEE により 1 であることが決められています。

5.2.11 レジスタ入出力記述

レジスタ入出力記述では、どの命令のときにどのレジスタを TDI と TDO の間に配置するのかを決定します。

例)

```
attribute REGISTER_ACCESS of SAMPLE_DEVICE: entity is
    "BYPASS (BYPASS, CLAMP, HIGHZ)," &
    "BOUNDARY (EXTEST, SAMPLE)," &
    "DEVICE_ID (IDCODE)";
```

上記の例では、BYPASS、CLAMP、HIGHZ 命令ではバイパスレジスタを、EXTEST、SAMPLE 命令ではバウンダリスキャンレジスタを、IDCODE 命令では IDCODE レジスタを TDI と TDO の間に接続します。

5.2.12 バウンダリスキャンレジスタ記述

バウンダリスキャンレジスタ記述では、バウンダリスキャンセルのリストを設定します。

バウンダリスキャンテストに使用するリストなので、信号性質が linkage の端子はこの記述は不要です。

まず、セルの長さ（例では 6）を設定します。そしてセルのリストを記述していきます。セルの順番は順不同ですが、すべてのセルが設定されていなければいけません。0 番目のセルが TDO に一番近く、番号が一番大きいセル（例では 5）が TDI に一番近くなります。

リストに記述する内容は、number（セル番号）、cell（セル構造）、port（接続ポート）、function（機能）、safe（安全値）の 5 つです。

number は、TDO に一番近いセルを 0 としてセルの長さより 1 少ない値までを設定します。

cell は、使用されるセル構造を定義します。セル構造の値は Standard VHDL package か VHDL package で定義されている値です。セル構造の値には、たとえば入力のみが可能な BC_4 や、すべての機能を持っている BC_1 などがあります。

port は、そのセルに接続する端子の名前です。

function では、セルの主な機能を設定します。主な function の意味は表 8 のとおりです。

表 8 Function の機能

値	機能	信号性質
input	モニタすることができ、外部ピンに接続する	in もしくは inout
observe_only	input と同様の機能	in もしくは inout
control	1 つ以上の出力信号、もしくは双方向信号の有効制御や方向制御を行う	—
output3	3 ステート出力を持つ外部ピンと接続する	out もしくは inout
internal	デバイス内部で使用する。外部とは接続されない	—

safe では、バウンダリスキャンテストパターン生成ソフトウェアがランダムにテストパターンを生成する際に安全な値を設定するようにします。

例)

```

attribute BOUNDARY_LENGTH of SAMPLE_DEVICE: entity is 6;
attribute BOUNDARY_REGISTER of SAMPLE_DEVICE: entity is
--   num   cell   port           function      safe  [ccell  disval  rslt]
    "5   (BC_4,  PX2,           observe_only,  X),
    "4   (BC_1,  *,            control,       0),
    "3   (BC_1,  PX2,           output3,       X,    4,    0,    Z),
    "2   (BC_4,  PX1,           observe_only,  X),
    "1   (BC_1,  *,            control,       0),
    "0   (BC_1,  PX1,           output3,       X,    1,    0,    Z)  " ;
    
```

Xは0でも1でも構わないことを示します。functionがoutput3の場合、ccellの指すセルに対し、disvalの指す値を設定すると、rsltの指す値が対象端子から出力されます。また、ccellの指すセルのsafeビットの値とdisvalビットは通常同じ値となります。Zはハイインピーダンスを指します。

5.3 BSDL ファイルサンプル

```
-- *****
-- BSDL file for SAMPLE_DEVICE
-- Designer:
-- Company: Renesas, Ltd.
-- Date: FRI 07 13 2007
-- *****
```

タイトル

entity SAMPLE_DEVICE is]エンティティ記述 (始まり)

```
-- This section identifies the default device package selected.
generic (PHYSICAL_PIN_MAP: string:= "SAMPLE_PACK");
```

] パラメータ記述

```
-- This section declares all the ports in the design.
```

```
port (
    PX1      : inout  bit;
    PX2      : inout  bit;
    PX3      : inout  bit;
    PX4      : linkage bit;
    TRST     : in     bit;
    TMS      : in     bit;
    TDI      : in     bit;
    TCK      : in     bit;
    TDO      : out    bit;
    PX5      : in     bit;
    PX6      : in     bit;
    PX7      : in     bit
);
```

] 論理ポート記述
(順不同)

```
use STD_1149_1_1994.all; ]standard use 文
```

```
attribute COMPONENT_CONFORMANCE of SAMPLE_DEVICE: entity is "STD_1149_1_1993";
```

] スタンド版指定

```
attribute PIN_MAP of SAMPLE_DEVICE: entity is PHYSICAL_PIN_MAP;
```

```
-- This section specifies the pin map for each port. This information
-- is extracted from the port-to-pin map file that was read in using
-- the "read_pin_map" command.
```

```
constant SAMPLE_PACK: PIN_MAP_STRING :=
    "PX1      : 1, " &
    "PX2      : 2, " &
    "PX3      : 3, " &
    "PX4      : 4, " &
    "TRST     : 5, " &
    "TMS      : 6, " &
    "TDI      : 7, " &
    "TCK      : 8, " &
    "TDO      : 9, " &
    "PX5      : 10, " &
    "PX6      : 11, " &
    "PX7      : 12";
```

] 端子
割り当て
(順不同)

```
-- This section specifies the TAP ports.
-- For the TAP TCK port, the parameters in the brackets are:
-- First Field : Maximum TCK frequency.
-- Second Field: Allowable states TCK may be stopped in.
attribute TAP_SCAN_CLOCK of TCK      : signal is (20.0e6, BOTH);
attribute TAP_SCAN_IN   of TDI       : signal is true;
attribute TAP_SCAN_MODE of TMS       : signal is true;
attribute TAP_SCAN_OUT  of TDO       : signal is true;
attribute TAP_SCAN_RESET of TRST     : signal is true;
```

TAP 定義

```
attribute COMPLIANCE_PATTERNS of SAMPLE_DEVICE:entity is
    "(PX6, PX7) (11)";
```

COMPLIANCE 文

```
-- Specifies the number of bits in the instruction register.
attribute INSTRUCTION_LENGTH of SAMPLE_DEVICE: entity is 4;
```

```
-- Specifies the boundary-scan instructions implemented in the
-- design and their opcodes.
```

```
attribute INSTRUCTION_OPCODE of SAMPLE_DEVICE: entity is
    "BYPASS (1111)," &
    "EXTEST (0000)," &
    "SAMPLE (1000)," &
    "CLAMP (1010)," &
    "HIGHZ (1011)," &
    "IDCODE (0101)";
```

インスト
ラクション
レジスタ
記述

```
-- Specifies the bit pattern that is loaded into the instruction
-- register when the TAP controller passes through the Capture-IR
-- state. The standard mandates that the two LSBs must be "01".
-- The remaining bits are design specific.
```

```
attribute INSTRUCTION_CAPTURE of SAMPLE_DEVICE: entity is "1101";
```

```
-- Specifies the bit pattern that is loaded into the DEVICE_ID
-- register during the IDCODE instruction when the TAP controller
-- passes through the Capture-DR state.
```

```
attribute IDCODE_REGISTER of SAMPLE_DEVICE: entity is --ID code H'04632B97
    "0000" & -- 4-bit version number
    "0100011000110010" & -- 16-bit part number
    "10111001011" & -- 11-bit identity of the manufacturer
    "1"; -- Required by IEEE Std 1149.1
```

オプションレジスタ定義

```
-- This section specifies the test data register placed between TDI
-- and TDO for each implemented instruction.
```

```
attribute REGISTER_ACCESS of SAMPLE_DEVICE: entity is
    "BYPASS (BYPASS, CLAMP, HIGHZ)," &
    "BOUNDARY (EXTEST, SAMPLE)," &
    "DEVICE_ID (IDCODE)";
```

レジスタ
入出力記述

```
-- Specifies the length of the boundary scan register.
attribute BOUNDARY_LENGTH of SAMPLE_DEVICE: entity is 13;

-- The following list specifies the characteristics of each cell
-- in the boundary scan register from TDI to TDO.
-- The following is a description of the label fields:
-- num      : Is the cell number.
-- cell     : Is the cell type as defined by the standard.
-- port     : Is the design port name. Control cells do not
--           have a port name.
-- function: Is the function of the cell as defined by the
--           standard. Is one of input, output2, output3,
--           bidir, control or controlr.
-- safe     : Specifies the value that the BSR cell should be
--           loaded with for safe operation when the software
--           might otherwise choose a random value.
-- ccell    : The control cell number. Specifies the control
--           cell that drives the output enable for this port.
-- disval   : Specifies the value that is loaded into the
--           control cell to disable the output enable for
--           the corresponding port.
--         rslt   : Resulting state. Shows the state of the driver
--                 when it is disabled.
attribute BOUNDARY_REGISTER of SAMPLE_DEVICE: entity is
--
--   num  cell  port      function      safe [ccell  disval  rslt]
--
--   "12 (BC_4,  PX5,      observe_only, X),
--   "11 (BC_4,  *,        internal,      X),
--   "10 (BC_1,  *,        internal,      X),
--   "9  (BC_1,  *,        internal,      X),
--   "8  (BC_4,  PX3,      observe_only, X),
--   "7  (BC_1,  *,        control,       0),
--   "6  (BC_1,  PX3,      output3,      X,  7,  0,  Z),
--   "5  (BC_4,  PX2,      observe_only, X),
--   "4  (BC_1,  *,        control,       0),
--   "3  (BC_1,  PX2,      output3,      X,  4,  0,  Z),
--   "2  (BC_4,  PX1,      observe_only, X),
--   "1  (BC_1,  *,        control,       0),
--   "0  (BC_1,  PX1,      output3,      X,  1,  0,  Z) ;

end SAMPLE_DEVICE; ) エンティティ文 (終わり)
```

バウンダリ
 スキャン
 レジスタ
 記述

6. バウンダリスキャンテストシステム構成例

最後に、バウンダリスキャンテストを使ったシステム構成例を図 25 に示します。

必要となるソフトウェアとファイルは主に「**BSDL** ファイル」、「テストパターンジェネレータソフト」、「**JTAG** 制御プログラム」、「システム制御プログラム」、「テストパターン作成プログラム」、「テスト結果解析プログラム」などです。「テストパターンジェネレータソフト」は、命令定義やテストベクタ定義などを行います。「**JTAG** 制御プログラム」は、**JTAG** プロトコルに準拠した制御を行います。「システム制御プログラム」は、評価システム全体のコントロールを行います。「テストパターン作成プログラム」は、テストを行う際出力するパターンデータを作成します。「テスト結果解析プログラム」は、バウンダリスキャンテストで得られたテスト結果（0 と 1 の羅列）が正常かどうかを判定します。

ホストコンピュータで **BSDL** ファイルをテストパターンジェネレータソフト、**JTAG** 制御プログラム、システム制御プログラム、テストパターン作成プログラムなどに取り込み、**JTAG** コントローラで **TAP** 信号を入出力することにより、バウンダリスキャンテストを行います。テスト結果は、テスト解析プログラムによって解析され、テスト判定結果として正常か異常かを出力します。

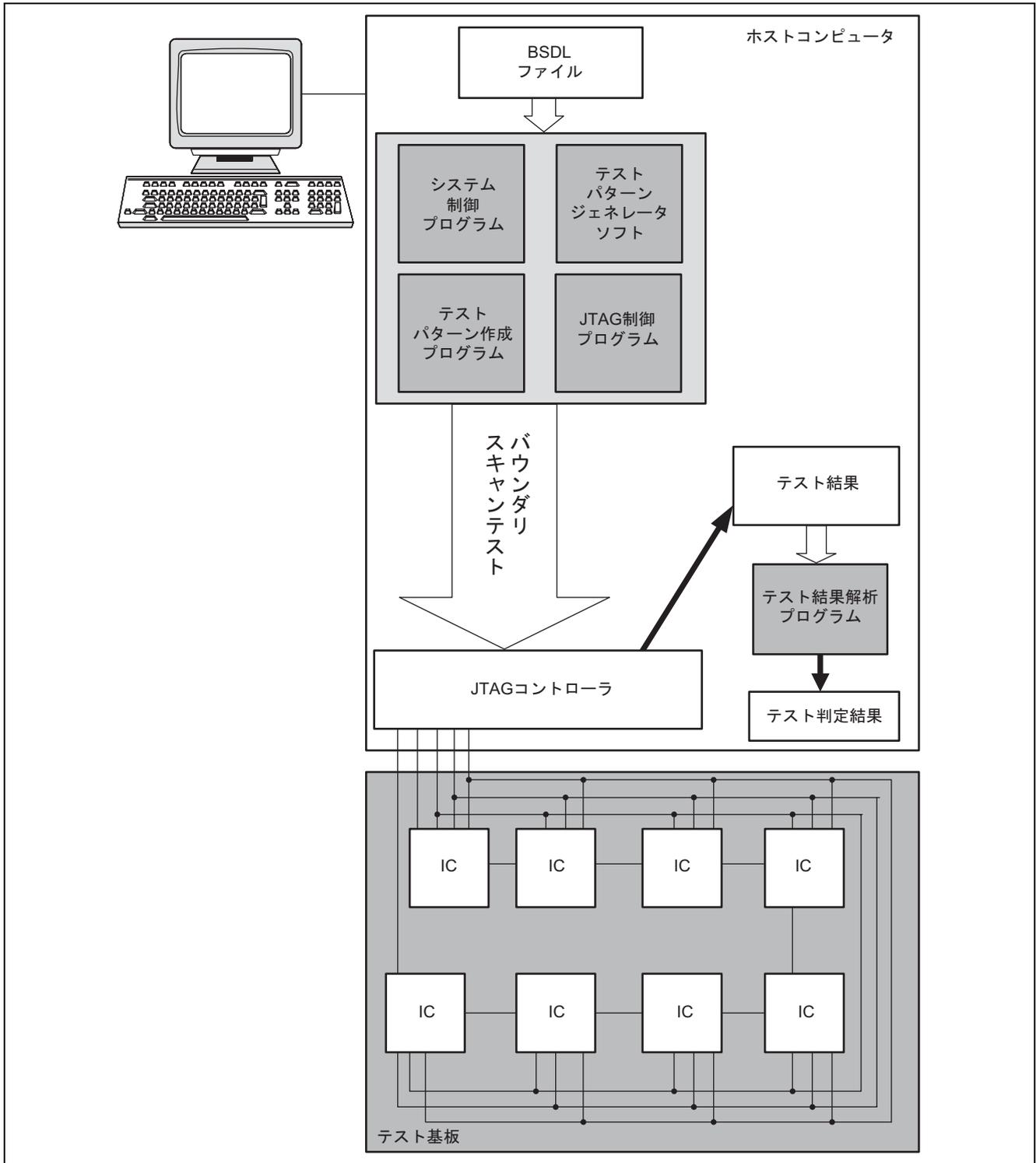


図 25 バウンダリスキャンテストシステム構成例イメージ

7. 参考文献

坂巻佳壽美著 (2003) 『JTAG テストの基礎と応用 新時代の電子回路基板のテスト手法と様々な応用事例』
CQ 出版

8. 参考ドキュメント

- ハードウェアマニュアル
H8SX/1638 グループハードウェアマニュアル
H8SX/1648 グループハードウェアマニュアル
(最新版をルネサス テクノロジホームページから入手してください。)
- テクニカルニュース/テクニカルアップデート
(最新の情報をルネサス テクノロジホームページから入手してください。)

ホームページとサポート窓口

ルネサステクノロジホームページ

<http://japan.renesas.com/>

お問合せ先

<http://japan.renesas.com/inquiry>
csc@renesas.com

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2009.01.19	—	初版発行

すべての商標および登録商標は、それぞれの所有者に帰属します。

本資料ご利用に際しての留意事項

1. 本資料は、お客様に用途に応じた適切な弊社製品をご購入いただくための参考資料であり、本資料中に記載の技術情報について弊社または第三者の知的財産権その他の権利の実施、使用を許諾または保証するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例など全ての情報の使用に起因する損害、第三者の知的財産権その他の権利に対する侵害に関し、弊社は責任を負いません。
3. 本資料に記載の製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的、あるいはその他軍事用途の目的で使用しないでください。また、輸出に際しては、「外国為替および外国貿易法」その他輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
4. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例などの全ての情報は本資料発行時点のものであり、弊社は本資料に記載した製品または仕様等を予告なしに変更することがあります。弊社の半導体製品のご購入およびご使用に当たりましては、事前に弊社営業窓口で最新の情報をご確認いただきますとともに、弊社ホームページ(<http://www.renesas.com>)などを通じて公開される情報に常にご注意ください。
5. 本資料に記載した情報は、正確を期すため慎重に制作したものです。万一本資料の記述の誤りに起因する損害がお客様に生じた場合においても、弊社はその責任を負いません。
6. 本資料に記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を流用する場合は、流用する情報を単独で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。弊社は、適用可否に対する責任を負いません。
7. 本資料に記載された製品は、各種安全装置や運輸・交通用、医療用、燃焼制御用、航空宇宙用、原子力、海底中継用の機器・システムなど、その故障や誤動作が直接人命を脅かしあるいは人体に危害を及ぼすおそれのあるような機器・システムや特に高度な品質・信頼性が要求される機器・システムでの使用を意図して設計、製造されたものではありません（弊社が自動車用と指定する製品を自動車に使用する場合を除きます）。これらの用途に利用されることをご検討の際には、必ず事前に弊社営業窓口へご照会ください。なお、上記用途に使用されたことにより発生した損害等について弊社はその責任を負いかねますのでご了承願います。
8. 第7項にかかわらず、本資料に記載された製品は、下記の用途には使用しないでください。これらの用途に使用されたことにより発生した損害等につきましては、弊社は一切の責任を負いません。
 - 1) 生命維持装置。
 - 2) 人体に埋め込み使用するもの。
 - 3) 治療行為（患部切り出し、薬剤投与等）を行うもの。
 - 4) その他、直接人命に影響を与えるもの。
9. 本資料に記載された製品のご使用につき、特に最大定格、動作電源電圧範囲、放熱特性、実装条件およびその他諸条件につきましては、弊社保証範囲内でご使用ください。弊社保証値を越えて製品をご使用された場合の故障および事故につきましては、弊社はその責任を負いません。
10. 弊社は製品の品質および信頼性の向上に努めておりますが、特に半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。弊社製品の故障または誤動作が生じた場合も人身事故、火災事故、社会的損害などを生じさせないよう、お客様の責任において冗長設計、延焼対策設計、誤動作防止設計などの安全設計（含むハードウェアおよびソフトウェア）およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特にマイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
11. 本資料に記載の製品は、これを搭載した製品から剥がれた場合、幼児が口に入れて誤飲する等の事故の危険性があります。お客様の製品への実装後に容易に本製品が剥がれることがなきよう、お客様の責任において十分な安全設計をお願いします。お客様の製品から剥がれた場合の事故につきましては、弊社はその責任を負いません。
12. 本資料の全部または一部を弊社の文書による事前の承諾なしに転載または複製することを固くお断りいたします。
13. 本資料に関する詳細についてのお問い合わせ、その他お気付きの点等がございましたら弊社営業窓口までご照会ください。