
セクション 24. プログラミングと診断

ハイライト

本セクションには以下の主要項目を記載しています。

24.1	はじめに	24-2
24.2	インサーキット シリアル プログラミング ™	24-3
24.3	拡張インサーキット シリアル プログラミング	24-6
24.4	JTAG バウンダリ スキャン	24-7
24.5	関連アプリケーション ノート	24-16
24.6	改訂履歴	24-17

Note: 本セクションは、デバイス データシートの内容の補足を目的としています。本セクションの内容は、dsPIC33F/PIC24H ファミリの一部のデバイスには対応していません。

本書の内容がお客様のご使用になるデバイスに対応しているかどうかは、最新デバイス データシート内の「プログラミングと診断」の冒頭に記載している注意書きでご確認ください。

デバイス データシートとファミリ リファレンス マニュアルの各セクションは、マイクロチップ社のウェブサイトからダウンロードできます。
<http://www.microchip.com>

24.1 はじめに

dsPIC33F/PIC24H は、各種アプリケーションの柔軟性を高める事ができるプログラミングと診断に関する幅広い機能を有しています。これらの機能により、システム設計者は以下の事が行えます。

- 2 線式インターフェイスを用いた簡易フィールド プログラム機能
- 強化したデバッグ機能
- デバイスと実装ボードの診断に役立つバウンダリ スキャン テスト

dsPIC33F/PIC24H はプログラミングと診断に 3 つのモードをサポートしており、アプリケーション開発者にとって便利な各種機能が利用できます。表 24-1 にそれらの機能をまとめます。

表 24-1: dsPIC33F/PIC24H のプログラミングと診断方法の比較

方法	インターフェイス	デバイス インテグレーション	機能
インサーキット シリアル プログラミング (ICSP™)	PGCx および PGDx ピン	デバイスコアに統合	プログラミング、 デバッグ
拡張 ICSP プログラミング	PGCx および PGDx ピン	ハードウェアは デバイスコアに統合、 ファームウェア ベースの制御	プログラミング
JTAG (Joint Test Action Group)	TDI、TDO、 TMS、TCK ピン	デバイスコアの周辺 モジュール、一部 I/O ロジックに統合	プログラミング、 バウンダリ スキャン テスト (BST) 診断

24.2 インサーキット シリアル プログラミング™

インサーキット シリアル プログラミング (ICSP™) 機能は、ターゲット アプリケーションでマイコンをプログラムするためのマイクロチップ社独自のプロセスです。ICSP は、システムに組み込まれたコントローラとデバイス プログラマにセットされたデバイスをプログラムする最も直接的な方法です。

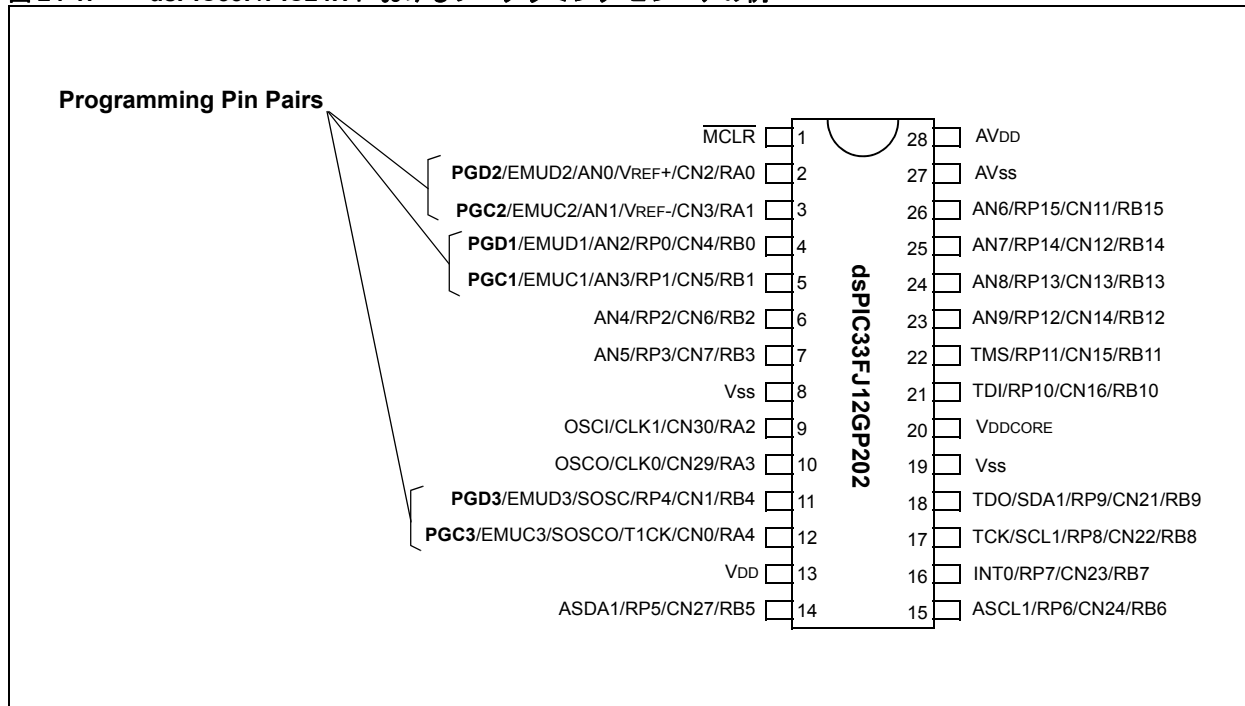
24.2.1 ICSP インターフェイス

ICSP インターフェイスの中核は 2 本のピンです。プログラミング データ (PGD) ピンは入力と出力の両方の機能を持っており、コマンドによりプログラミング データの読み込みとデバイス情報の読み出しが可能です。プログラミング クロック (PGC) ピンは、データに同期し、プロセス全体を制御します。

dsPIC33F/PIC24H の多くが PGC ピンと PGD ピンの組み合わせを複数備えています。これらのピンは図 24-1 に示すように他の I/O または周辺機能と多重化されています。それぞれの ICSP ピンの組み合わせは数字で識別し (例: PGC1/PGD1)、総称として PGCx/PGDx のように表現します。PGCx/PGDx ピンのペアが複数ある事で、回路設計上都合の良いピンペアに ICSP を適用できるため、柔軟なシステム設計が可能です。PGCx/PGDx ピンペアは全て機能上互いに連結しており、挙動は同じです。デバイス プログラミングはどのピンペアを使っても行う事ができます。ただ 1 つの制約事項は、同一番号のピンペアを使う必要があるという事のみです。

ICSP では、PGCx/PGDx ピンに加えてデバイスの全ての電源ピンとグランドピンを接続する必要があります。また、MCLR ピン (PGCx と共に用いてプログラミング プロセスで制御します) も、プログラマに接続しておく必要があります。

図 24-1: dsPIC33F/PIC24H におけるプログラミング ピンペアの例



24.2.2 ICSP 動作

ICSP モードでは、内部ハードウェアと外部制御の組み合わせによってターゲットデバイスをプログラムします。プログラミング データと命令は PGD ピンから与えます。特殊な 4 ビットのコマンドセットと dsPIC33F/PIC24H 標準コマンドを用いてプログラムメモリへ書き込みます。PGD ピンは外部プログラマへの問い合わせ応答のデータも返します。

このプログラミング プロセスは、PGC ピンと $\overline{\text{MCLR}}$ ピンを操作する事によって制御します。ICSP モードに入る (あるいは出る) ためには、PGD ヘコードシーケンス、PGC ヘクロックを与えながら $\overline{\text{MCLR}}$ に電圧をかける (あるいは取り除く) 必要があります。

プログラミングには、どの PGCx/PGDx ピンペアでも使用できます。プログラミング中、PGC 上のクロックパルス列は、4 ビットコマンド、プログラミング制御コマンド、プログラムされるデータを区別するためにも使用されます。

この内部プロセスは、dsPIC33F/PIC24H コアロジック内に構築されたステートマシンが制御します。しかし、プロセス全体の制御は外部プログラミング デバイスで行う必要があります。MPLAB® PM3 (MPLAB IDE の開発ソフトウェアと共に用いる) のようなマイクロチップ プログラミング デバイスは、dsPIC33F/PIC24H のプログラミング プロセスを管理するために必要なハードウェアとアルゴリズムを備えています。dsPIC33F/PIC24H に独自のプログラミング インターフェイスを設計する場合や詳細な情報が必要な場合、『dsPIC33F/PIC24H プログラミング仕様』(DS70152) を参照してください。

24.2.3 ICSP とインサーキット デバッグ

ICSP はインサーキット デバッグのためのハードウェア チャンネルも備えており、ソフトウェア デバッグの外部制御ができます。適切なハードウェア インターフェイスとソフトウェア環境を用いる事により、デバイスに対して命令コードのシングルステップ実行、複数レジスタの内容のトラッキング、ソフトウェア ブレークポイントの設定ができます。

インサーキット デバッグを行うには、外部システムによりデバッグの実行プログラムをマイコンに読み込む必要があります。読み込みは、MPLAB ICD 2 のような多くのデバッグ ツールでは自動的に実行されます。dsPIC33F/PIC24H では、プログラムはコンフィグレーション メモリ空間の実行プログラムメモリに読み込まれます。メモリは実装されていてこの位置から命令コードが実行可能であったとしても、通常動作モード時には実行メモリ空間をユーザ アプリケーションから利用できません。詳細は、『dsPIC33F/PIC24H プログラミング仕様』(DS70152) を参照してください。

デバッグ実行プログラムを使用しても、メモリ位置が異なるためテスト中のアプリケーションサイズに影響はありません。この実行メモリ空間のおかげで、プログラムメモリの全体をプログラムコード用に使用する事ができます。アプリケーション デバッグのため空間を残しておく必要はありません。また、このデバッグ実行メモリ空間を使用すると、プログラムメモリの内容が通常時とデバッグ時で同一であることを意味するため、トラブルシューティングが容易になります。

dsPIC33F/PIC24H の種類によって、プログラミング用に使用できる ICSP ポートが 1 つのものと複数のものがあります。ただし、インサーキット デバッグに使用できる ICSP ポートは 1 つのみです。デバッグ用に有効化する部分を選択するには、MPLAB IDE の設定で以下の手順を実行します。

1. MPLAB IDE で、[Configure]>[Configuration Bits] を選択して [Configuration Bits] ウィンドウを表示します。
2. [Configuration Bits] ウィンドウ内の [Comm Channel Select Category] の下で適切なデバッグ ペア設定を選択します。

Note: コンフィグレーション メモリ空間の詳細は、『dsPIC33F/PIC24H フラッシュ プログラミング仕様』(DS70152) を参照してください。

24.3 拡張インサーキット シリアル プログラミング

拡張 ICSP プロトコルは、ICSP の拡張版です。拡張 ICSP は、ICSP と同じ物理インターフェイスを使用しますが、プログラミング制御の場所と実行が変更されています。

ICSP モードでは、単純なステートマシンを用いてプログラミング プロセスの各ステップを制御しますが、このステートマシンは外部プログラマによって制御します。それに対して拡張 ICSP では、管理プログラムともいう内蔵ブートローダを用いてプログラミング プロセスを管理します。デバイス プログラミング全体は依然として外部プログラマから監視しますが、標準 ICSP ではプログラマが直接制御しなくてはならない操作の大部分を管理プログラムが管理します。

管理プログラムは、独自のコマンドセットを実装しています。従来の ICSP より数が多く、マイコンのプログラムメモリを直接消去、プログラム、ベリファイする事ができます。これにより、簡単なタスクを実行するために ICSP コマンドシーケンスを繰り返す必要がなくなります。結果として、拡張 ICSP モードではデバイスのプログラムあるいは再プログラムを従来の ICSP モードより迅速に行えます。

インサーキット デバッグが実行プログラムと同様、管理プログラムはユーザ アプリケーションプログラム メモリ空間には配置されません。管理プログラムのプログラムメモリに読み込まれます。デバッグが実行プログラムと拡張 ICSP 管理プログラムは両方ともこの同じメモリ空間を使用するため、プログラミングで拡張 ICSP モードを使用している間はインサーキット デバッグを使用できません。

管理プログラムは dsPIC33F/PIC24H に前もってプログラムされているわけではありません。拡張 ICSP を使いたい場合、標準 ICSP を用いて管理プログラムを管理プログラムメモリ空間にプログラムする必要があります。これはユーザのソフトウェアに直接に設定する事もできれば、マイクロチップ社製プログラマを用いて自動的に設定する事もできます。

拡張 ICSP と管理プログラムの詳細は、『dsPIC33F/PIC24H プログラミング仕様』(DS70152)を参照してください。

24.4 JTAG バウンダリ スキャン

ボード設計が複雑になり集積度が増すにつれて、実装ボード上の部品間の電気的接続のテストが課題となっています。この課題を解決するために、Joint Test Action Group (JTAG) がバウンダリ スキャン テスト手法を開発し、後に IEEE 1149.1-2001、『IEEE Standard Test Access Port and Boundary Scan Architecture』として標準化されました。標準として採択されて以来、多くのマイコンメーカーがテストポートの機能にデバイス プログラミングを追加しました。

JTAG バウンダリ スキャン方式では部品の各 I/O ピンに隣接した場所にシフトレジスタ段を設け、あらかじめ定義した一連のスキャンテスト基準を用いて部品周辺の信号の制御と観測を行います。外部テストまたはコントローラでシリアルに命令を供給して結果を読み出します。この外部デバイスからは共通クロックと制御信号も供給します。実装方法にもよりますが、テスト信号の入出力は全て 4 ピンまたは 5 ピンの標準インターフェイスを介して行います。

システムレベルのアプリケーションでは、(通常のアプリケーション固有の接続に加えて) 個々の JTAG 対応デバイスを各種テスト インターフェイスを介して接続します。デバイスは直列またはデジチェーン (数珠つなぎ) 方式で接続します。すなわち、あるデバイスのテスト出力を次のデバイスのテスト入力に接続してチェーンを形成します。JTAG バウンダリ スキャン プロトコルの命令を使用すると、このチェーン全体ではなく任意のデバイス (1 つまたは複数) をテストできます。この方法では、アプリケーションの境界における接続だけでなく部品間の接続もテストできます。

図 24-2 に JTAG バウンダリ スキャン インターフェイスを組み込んだ代表的なアプリケーションを示します。この例では、dsPIC33F/PIC24H デジタル シグナル コントローラ (DSC) が別の JTAG 対応デバイスにデジチェーンで接続されています。外部テストからの TDI (Test Data Input) ラインがチェーンの最初のデバイスの TDI ピンにデータを供給します (この例では DSC です)。この 2 つのデバイスからなるチェーンのテスト結果は、2 番目のデバイスの TDO (Test Data Output) ピンからテストの TDO ラインに供給されます。

本セクションでは、JTAG モジュールとその一般的な使用方法について説明します。デバイスプログラミングのための JTAG インターフェイスの使用の詳細は、24.4.6「JTAG デバイス プログラミング」を参照してください。

図 24-2: デバイスをデジチェーンに構成した dsPIC33F/PIC24H ベースの JTAG 対応アプリケーションの概要

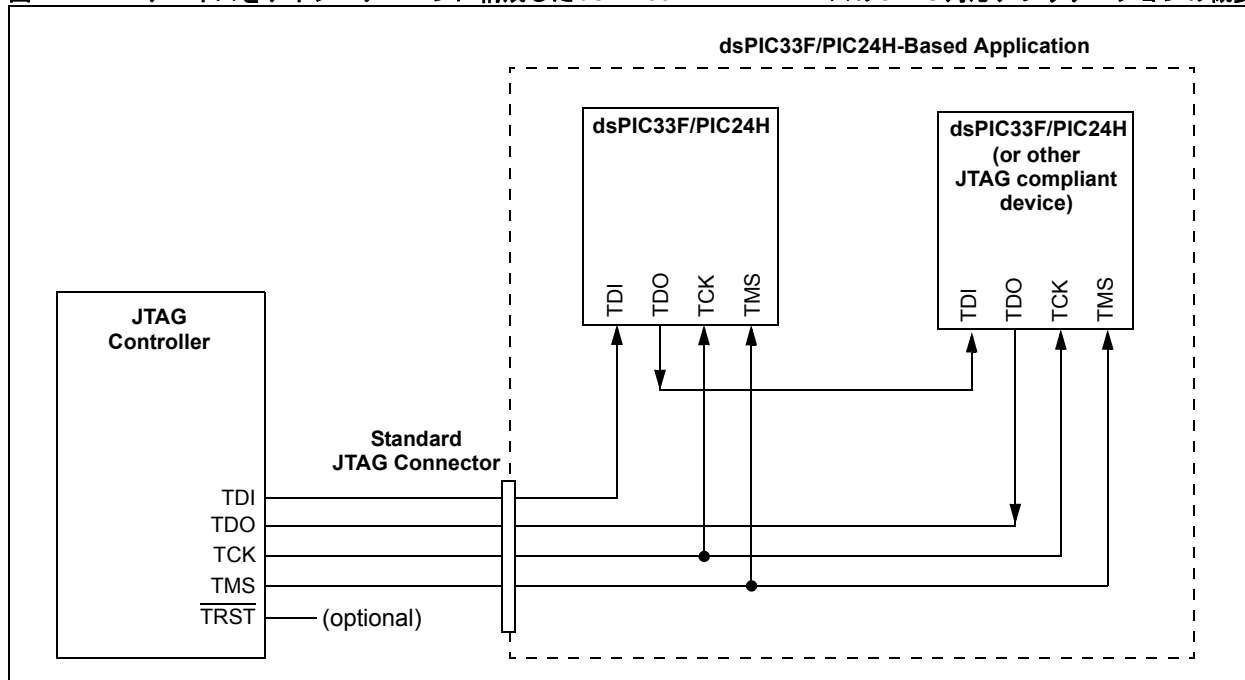


表 24-2: 4 ピン JTAG インターフェイス

図 24-3 に示す JTAG モジュールの論理ブロック図は、以下の主要要素で構成されます。

- TAP インターフェイス ピン (TDI、TMS、TCK、TDO)
- TAP コントローラ
- 命令シフトレジスタと命令レジスタ (IR)
- データレジスタ

The diagram shows the JTAG TAP Controller and its internal components. The TAP Controller is a central unit that receives TDI, TMS, and TCK signals. It controls the Instruction Shift Register (which contains the Instruction Register and Instruction Decode) and the Data Registers (which include the Bypass Register, Device ID Register, MCHP Command Shift Register, MCHP Command Register, and Boundary Scan Cell Registers). The TAP Controller outputs Capture-IR, Shift-IR, Update-IR, Capture-DR, Shift-DR, and Update-DR signals to the respective registers. The Instruction Shift Register outputs to the TDO Selector (MUX), which then outputs to the TDO pin. The Data Registers output to the Data Selector (MUX), which then outputs to the Output Data Sampling Register and finally to the TDO pin. The MCHP Command Register is connected to internal logic, and the Boundary Scan Cell Registers are connected to MCHP Scan Data from Internal Logic.

24.4.1 テスト アクセス ポート (TAP) と TAP コントローラ

dsPIC33F/PIC24H ファミリのテストアクセスポート (TAP) は汎用ポートで、多くの内蔵サポート機能と IEEE 標準 1149.1 に定義されているテストロジックを利用できます。TAP コントローラと関連のパウンダリ スキャンピンは、コンフィグレーション レジスタ (FICD) の JTAG イネーブルビット (JTAGEN) を「0」にすると無効 (ディセーブル) にできます。JTAG イネーブルビットが初期設定状態では、TAP コントローラは有効 (イネーブル) です。イネーブル時、対応する I/O ピンは専用の TAP ピンとなります。JTAG ポートを有効 / 無効化するには、MPLAB IDE の設定で以下の手順を実行します。

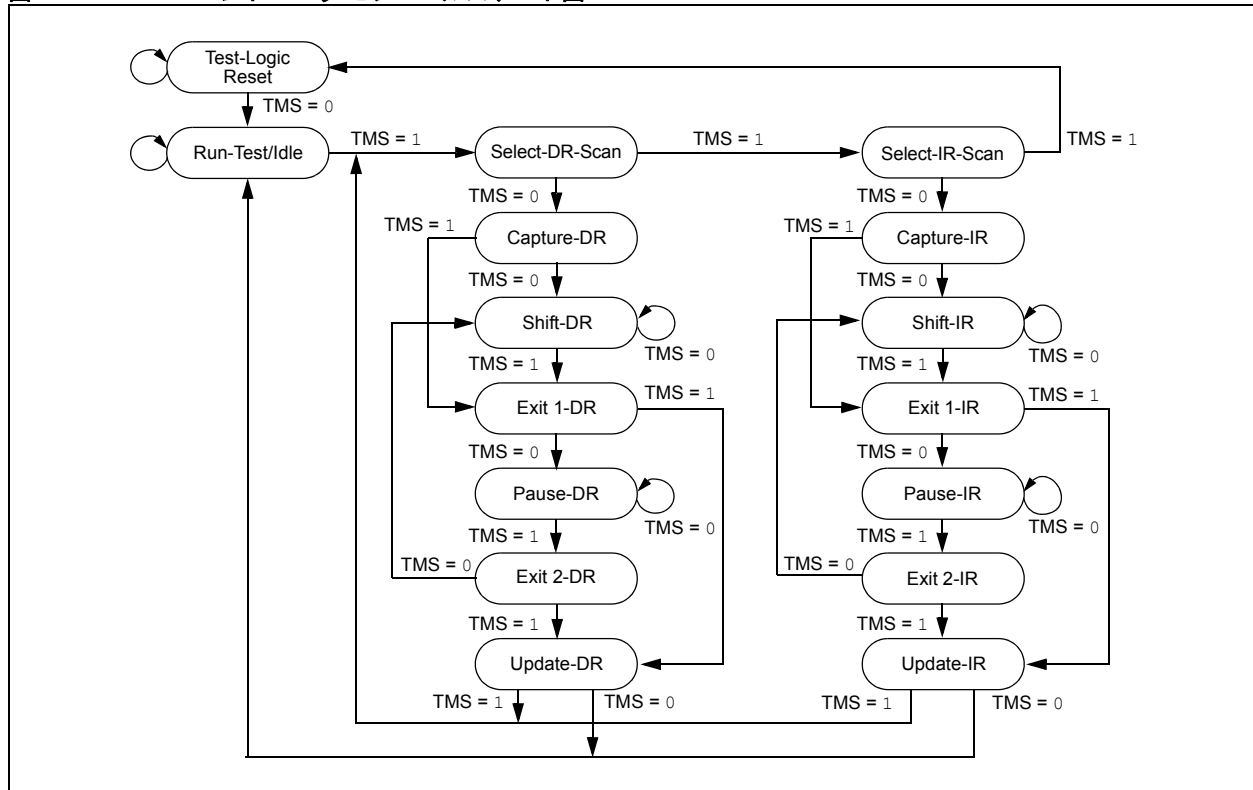
1. MPLAB IDE で、[Configure]>[Configuration Bits] を選択して [Configurations Bits] ウィンドウを表示します。
2. [Configurations Bits] ウィンドウ内の [JTAG Port Enable Category] で [Enable/Disable setting] を選択します。

Note: FICD レジスタの詳細は、『dsPIC33F/PIC24H フラッシュ プログラミング仕様』 (DS70152) を参照してください。

JTAG スキャンのために I/O 数が減少するのを最小限にするため、dsPIC33F/PIC24H には JTAG 標準で指定されているオプションの TAP リセット (TRST) 入力ピンは実装されていません。便宜上、TMS ピンと TCK ピンを用いた「ソフト」TAP リセットが TAP コントローラに用意されています。ポートを強制的にリセットするには、TCK の立ち上がりエッジ 5 つ以上 TMS ピンに論理 High 信号を印加します。パワーオン リセット (POR) を含むデバイスリセットによって自動的に TAP リセットされる事はありません。TAP リセットを行うには、外部 JTAG コントローラからソフト TAP リセットを使用する必要があります。

dsPIC33F/PIC24H ファミリの TAP コントローラは同期式有限状態マシンで、JTAG スキャン標準の 16 ステートを実装しています。図 24-4 に、TAP コントローラの全モジュール ステートを示します。全てのパウンダリ スキャン テスト (BST) 命令とテスト結果は、TDI ピンから TAP を経由して最下位ビット (LSb) を先頭とするシリアル形式で送信されます。

図 24-4: TAP コントローラ モジュールステート図



TMS のステートと TCK のクロックパルスを操作する事により、TAP コントローラは定義された全てのモジュール ステートに移行し、各種の命令レジスタまたはデータレジスタをキャプチャ、シフト、更新できます。図 24-4 に、コントローラがステートマシンによって遷移するに従い TMS のステートが変化する様子を示します。図 24-5 に、命令をシフト入力する際のモジュール ステートによってコントローラが遷移する時の TMS と TCK のタイミングを示します。この例では、TAP コントローラが命令をどのように読み出すかを順に示しています。

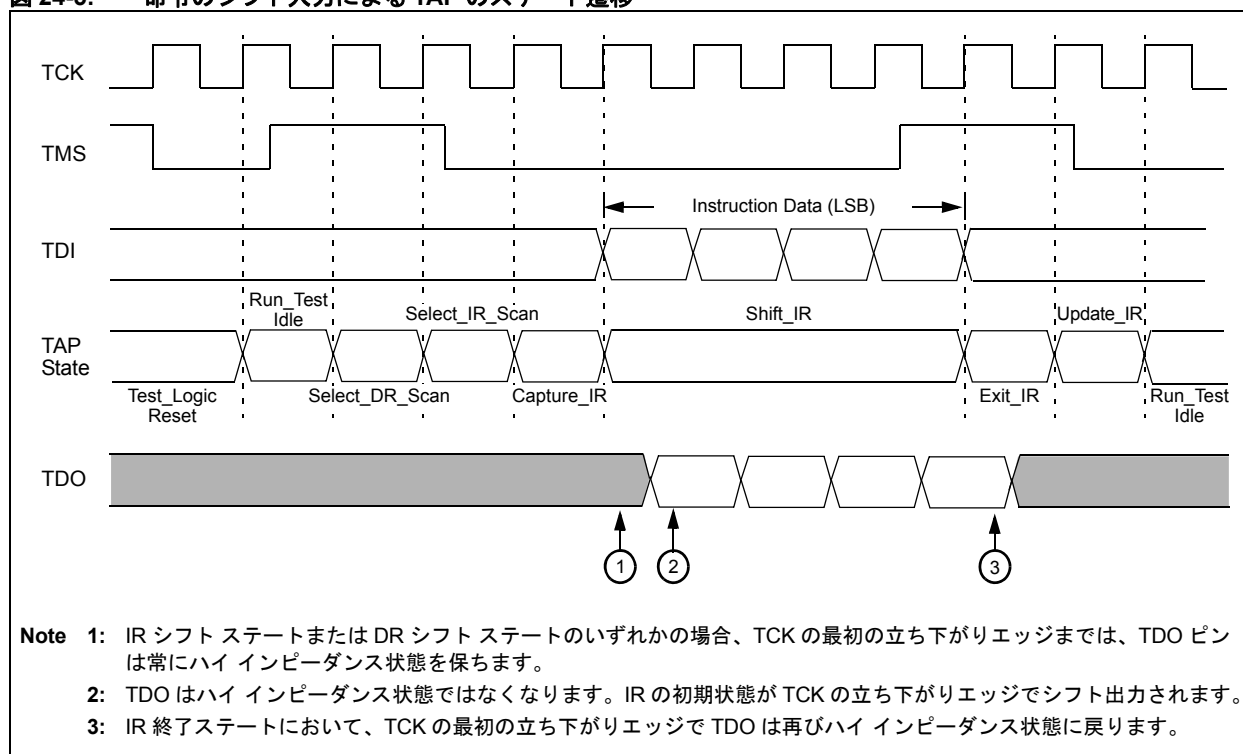
TAP コントローラのステートは全て、TCK ピン上の信号の立ち上がりエッジで開始します。TAP コントローラは、テストロジック リセット ステートで開始します。TAP コントローラのステートは直前の命令によって決まり未知であるため、テストロジック リセット ステートから開始する事を推奨します。

TCK の次の立ち上がりエッジで TMS が Low に駆動されると、TAP コントローラはテスト実行 / アイドル ステートに移行します。TCK の続く 2 つの立ち上がりエッジで TMS が High になると、TAP コントローラは IR スキャン選択ステートに移行します。

TCK の続く 2 つの立ち上がりエッジで TMS が Low を維持すると、TAP コントローラは IR シフト ステートに移行します。TCK の次の 4 つの立ち上がりエッジで、TDI を介して命令が命令シフトレジスタにシフト入力されます。TAP コントローラがこのステートに入った後、TDO ピンはハイ インピーダンスからアクティブに移行します。コントローラは、TCK の立ち下がりエッジで命令レジスタ (IR) の初期状態を TDO ピンにシフト出力し、IR シフト ステートにある間 IR の内容をシフト出力し続けます。シフト ステートを抜けた後、TCK の最初の立ち下がりエッジで TDO はハイ インピーダンス状態に戻ります。

TCK の次の 3 つの立ち上がりエッジで、TAP コントローラは IR シフト ステートを抜け、IR を更新し、その後テスト実行 / アイドル ステートに戻ります。この時点で、データまたは別の命令を適切なデータレジスタまたは IR にシフト入力できます。

図 24-5: 命令のシフト入力による TAP のステート遷移



24.4.2 JTAG レジスタ

JTAG モジュールは、動作の一環で各種サイズのレジスタをいくつか使用します。JTAG レジスタはいずれもデバイスのデータメモリ空間には配置されていません。通常動作モードでは、ユーザアプリケーションから JTAG レジスタに直接アクセスする事はできません。

24.4.2.1 命令シフトレジスタと命令レジスタ

4 ビット IR により、命令をデバイスにシフト入力できます。この命令によってアクセスするデータレジスタを選択します。

命令レジスタからのパラレル出力はラッチされ、新規の命令データの入力に伴って起こるシフトレジスタ段でのデータパターンの過渡的变化から保護されます。このラッチされたパラレル出力は、コントローラの IR 更新ステートとテストロジック リセット ステート時のみステートを変更できるように管理されています。

24.4.4 「JTAG 命令」 に、実装された命令の一覧と説明を示します。

24.4.2.2 データレジスタ

dsPIC33F/PIC24H ファミリは、表 24-3 に示す JTAG データレジスタをサポートしています。

表 24-3: JTAG データレジスタ

レジスタ	機能
バイパスレジスタ	TDI と TDO 間でテストデータを移動させる最短のシリアル経路を提供します。ボードレベルのテスト動作時で他のテスト データレジスタにアクセスする必要がない場合、この経路を選択できます。 デバイスでバイパスレジスタを使用すると、ボードレベルのテストデータ経路上の他のデバイスのテスト データレジスタへのアクセス速度を高める事ができます。
マイクロチップ コマンド シフトレジスタ	この 8 ビット シフトレジスタはマイクロチップ社デバイス独自のコマンドをシフト入力できます。シフトレジスタからのパラレル出力はラッチされ、新規のコマンド入力に伴って起こるシフトレジスタ段でのデータパターンの過渡的变化から保護されます。
デバイス ID レジスタ	この 32 ビットデバイス IR を使用すると、メーカー、製品番号、デバイスタイプを識別できます。デバイス ID レジスタと JTAG ID レジスタの詳細は、『dsPIC33F/PIC24H フラッシュ プログラミング仕様』(DS70152) を参照してください。
バウンダリ スキャン レジスタ	複数のセルを組み合わせて構成され、対応する命令が選択された時、TDI と TDO 間を接続する 1 つのシフトレジスタ ベースの経路を形成します。

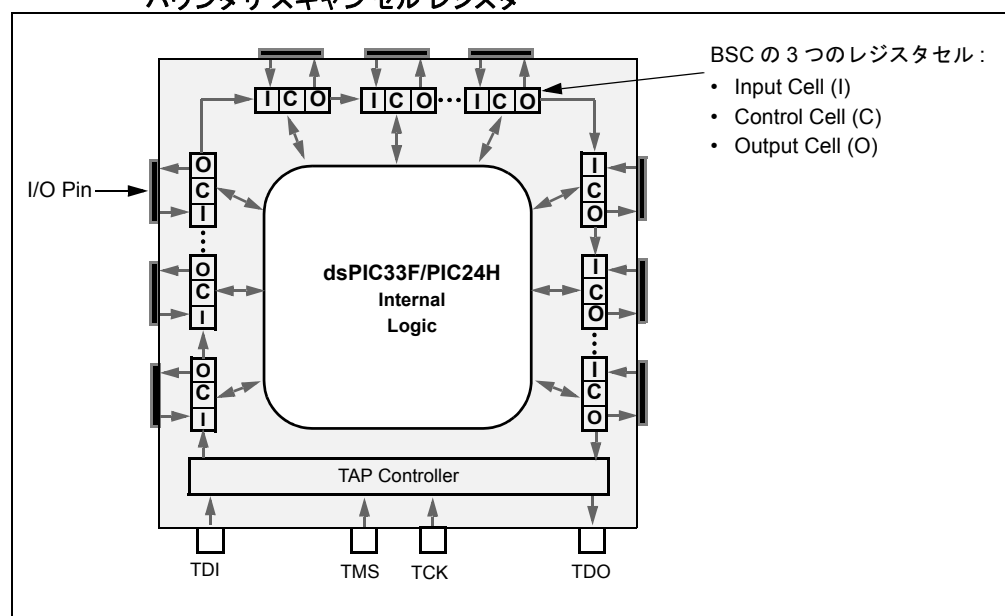
24.4.3 バウンダリ スキャン レジスタ

バウンダリ スキャン レジスタ (BSR) は巨大なシフトレジスタで、図 24-6 に示すように、全 I/O バウンダリ スキャン セルによるデジチェーン接続で構成されます。各 I/O ピンに 1 つのバウンダリ スキャン セル (BSC) があります。各 BSC には入力、出力、制御の 3 つの BSC レジスタがあります。SAMPLE/PRELOAD または EXTEST 命令がアクティブな場合、BSR は TDI ピンと TDO ピンの間に配置され、TDI ピンが入力で TDO ピンが出力となります。

BSR のサイズはデバイスの I/O ピンの数によって決まります。例えば、dsPIC33FJ256GP710 には I/O ピンが 82 本あります。82 本の I/O に対して各 3 つの BSC レジスタがあるため、BSR の長さは 246 ビットです。特定のデバイスの I/O ポートピン数の情報は、その製品の BSDL ファイルを参照してください。

Note: BSC は電源ピン (VDD、VDDCORE、VSS、AVDD、AVSS) には使用しません。JTAG の相互接続機能と制御機能を持つピンはスキャンチェーンに含まれず、JTAG テストはできません。

図 24-6: dsPIC33F/PIC24H デジタル シグナル コントローラのデジチェーン接続されたバウンダリ スキャン セル レジスタ



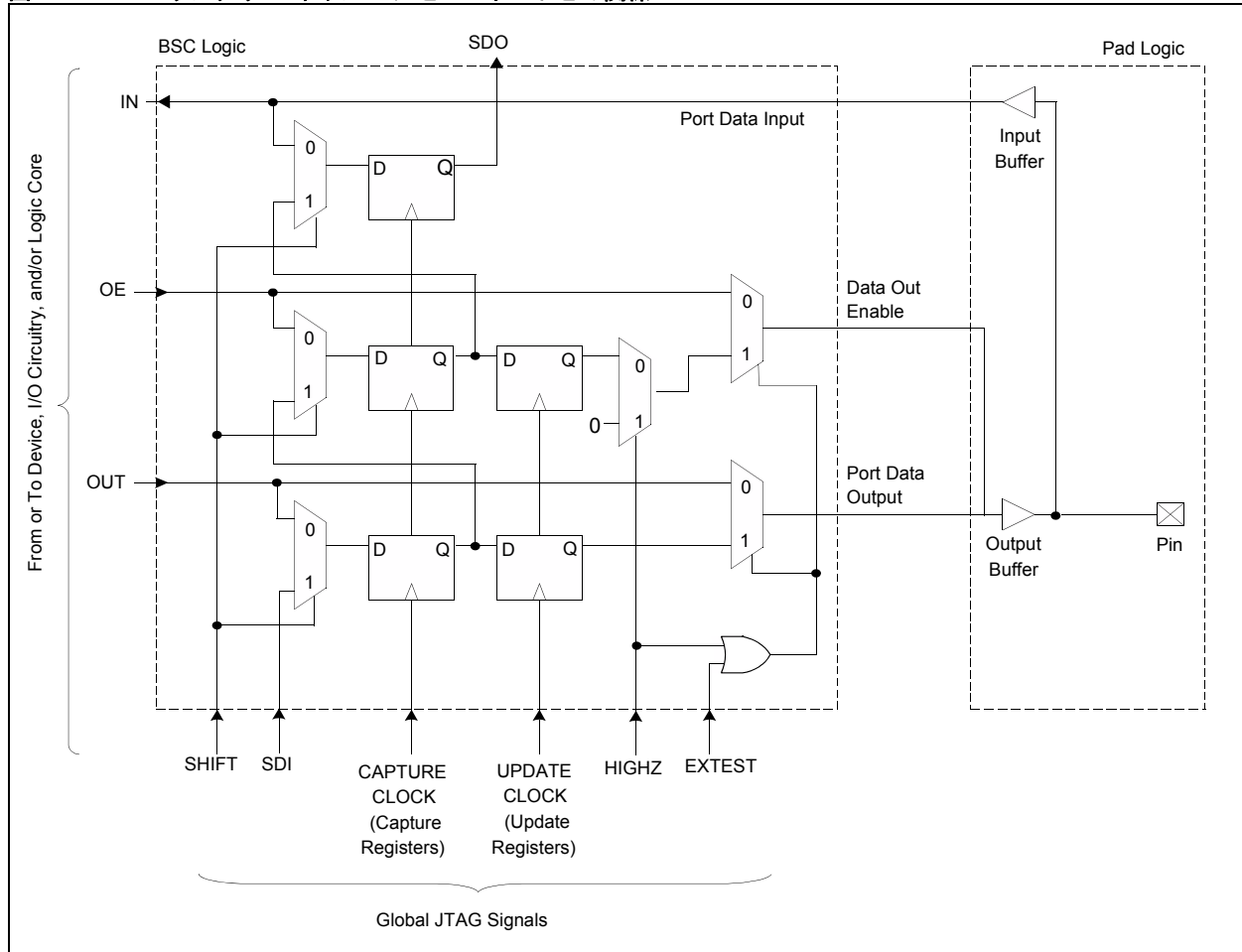
24.4.3.1 バウンダリ スキャン セル (BSC)

JTAG がアクティブな時、BSC は I/O の入出力データ値をキャプチャしてオーバーライドします。BSC は 3 つのシングルビット キャプチャ レジスタセルと 2 つのシングルビット保持レジスタセルから構成されます。キャプチャセルはデジタイゼーション接続され、ポートの入力をキャプチャし、データの出力と制御 (出力イネーブル) を行います。また、キャプチャセルは JTAG データを BSR 経由で送ります。TAP コントローラからの命令信号によって、JTAG データをキャプチャするかどうか、いつどのように BSC からクロック同期して出力するかを決定します。

1 番目のレジスタは、出力ドライバに送られた内部データをキャプチャするか、シリアルにスキャン入力されたデータを出力ドライバに供給します。2 番目のレジスタは、出力ドライバからの内部出力イネーブル制御信号をキャプチャし、シリアルスキャンされた出力イネーブル値を供給します。3 番目のレジスタは、I/O 入力バッファからの入力データをキャプチャします。

図 24-7 に、代表的な BSC と I/O ポートとの関係を示します。

図 24-7: バウンダリ スキャン セルと I/O ポートとの関係



24.4.4 JTAG 命令

dsPIC33F/PIC24H は IEEE 1149.1 で定義されている必須の命令セットをサポートしており、さらに同仕様で定義されているオプション命令の一部もサポートしています。これらのデバイスはマイクロチップ社独自の命令も実装しています。表 24-4 に、JTAG の必須の命令、オプションの命令、マイクロチップ社独自の命令をまとめています。

表 24-4: JTAG 命令

JTAG 命令	16 進値	説明
JTAG の必須命令		
BYPASS	0x0F	テストチェーン内でデバイスをバイパスします。バイパスモードでは、1 つのシフトレジスタ段で TDI ピンと TDO ピン間の最短シリアル経路が得られます。
SAMPLE/PRELOAD	0x01	実装済みのボードで通常動作に干渉する事なく、デバイスの入出力信号のスナップショットを取ります。スナップショットは DR キャプチャ コントローラ ステートの TCK 立ち上がりエッジで取ります。データはデバイスの TDO 出力で参照できます。 この命令によって、内蔵システム ロジックの通常動作に干渉せずに BSR のスキャンを行う事も可能です。例えば、EXTEST 命令を選択する前に PRELOAD 命令を用いて、ラッチされたパラレル出力上にデータを読み込む事ができます。EXTEST 命令が命令レジスタのパラレル出力に転送されると即座に、プリロードされたデータがシステム出力ピンを介して出力されます。これにより、EXTEST 命令が入力されると即座にボードレベルに一致した既知のデータが駆動されます。PRELOAD 命令がないと、最初のスキャン シーケンスが完了するまでは不確定なデータが駆動される可能性があります。
EXTEST	0x03	外部回路とボードレベルの相互接続テストを行う事ができます。EXTEST 命令を選択する前に PRELOAD 命令を用いる事で、通常はバウンダリ スキャン シフトレジスタ段のラッチされたパラレル出力上にデータが読み込まれます。出力ピンにある BSR セルはテスト信号を供給するために用います。入力ピンにある BSR セルはテスト結果をキャプチャするために用います。
JTAG のオプション命令		
IDCODE	0x02	DR シフト コントローラ ステートにおいて、TDI と TDO 間のシリアルアクセスのために接続する 32 ビットの ID レジスタを選択します。この命令により、32 ビットのデバイス識別ワードが TDO ピンからシフト出力されます。
HIGHZ	0x04	デバイスを、システムロジックの全ての出力を非アクティブにするステート (例えば ハイ インピーダンス) にします。このステートではデバイスを損傷する事なく、通常はデバイス出力が駆動する接続部に、インサーキット テストシステムから信号を駆動できます。HIGHZ モードでは、DR シフト ステート時にバイパスレジスタが TDI と TDO 間に接続されます。
マイクロチップ社独自の JTAG 命令		
MCHP_SCAN	0x07	DR シフト コントローラ ステートにおいて、TDI と TDO 間のシリアルアクセスのために接続するマイクロチップ社独自の内部スキャンレジスタを選択します。
MCHP_CMD	0x08	DR シフト コントローラ ステートにおいて、TDI と TDO 間のシリアルアクセスのために接続するマイクロチップ社独自の 8 ビットコマンド シフトレジスタを選択します。このシフトレジスタは、最大 256 のコマンドをサポートします。ユーザが利用できるのは以下の 2 つのコマンドで、残りは予約済みです。 <ul style="list-style-type: none"> JTAG MCLR (01h): JTAG インターフェイスがアクティブの時にデバイスのマスタ クリア リセットを行います。機能的にはハードウェア MCLR と同等です。TAP インターフェイス自体はリセットされません。 JTAG MUX (02h): JTAG インターフェイスを ICSP 動作に切り換えます。このコマンドの後、TDI と TDO は PGD 機能として (入力と出力に分離)、TCK は PGC として機能します。

24.4.5 バウンダリ スキャン テスト (BST)

バウンダリ スキャン テストは、JTAG 準拠デバイスの境界ピンをソフトウェアで制御し観測する手法です。BST では、複数の JTAG 準拠デバイスをデジタイズチェーン接続で 1 つのスキャンチェーンとして構成し、デバイス間の接続をテストできます。1 枚の PCB 上に複数のスキャンチェーンを構成することもできます。これらのスキャンチェーンを同時に駆動し、多数のデバイスを並行してテストすることができます。スキャンチェーンには JTAG 準拠デバイスと非準拠デバイスが混在してもかまいません。

BST の主な利点は、物理的なテストプローブを設けなくても実行できる事です。BST を実装するには、4/5 線式のインターフェイスと適切なテスト プラットフォームが必要です。JTAG バウンダリ スキャンは標準化されてから何年も経つため、複雑な物理ブロービングなしでスキャンチェーンをテストするための多くのソフトウェアツールが出回っています。BST の欠点は、デジタル信号と回路接続しか評価できない点です。入出力の電圧レベルまたは電流は計測できません。

24.4.5.1 JTAG 関連ファイル

BST を実装するには、全ての JTAG テストツールで BSDL (Boundary Scan Description Language) ファイルが必要です。BSDL は、VHDL (VHSIC ハードウェア記述言語) のサブセットで、IEEE 1149.1 の一部として記述されています。デバイスごとの BSDL ファイルには、そのデバイスに対する標準の実装方法と、動作方法が記述されています。デバイスごとの BSDL ファイルには以下の情報が記述されています。

- ピン配置とパッケージ構成
- TAP ピンの物理的位置
- デバイス ID レジスタとデバイス ID
- IR の長さ
- サポートする BST 命令とバイナリコード
- バウンダリ スキャン レジスタの長さ
- バウンダリ スキャン セルの定義

デバイスごとの BSDL ファイルはマイクロチップ社のウェブサイト www.microchip.com でご覧ください。各 BSDL ファイルのファイル名はデバイス名とシリコン リビジョンで構成されます。例えば、dsPIC33FJ256GP710 の BSDL ファイルは dsPIC33FJ256GP710.BSD です。

24.4.6 JTAG デバイス プログラミング

JTAG インターフェイスは dsPIC33F/PIC24H をターゲット アプリケーションでプログラムするためにも使用できます。JTAG インターフェイスを利用すると、アプリケーションの設計者はテストおよびプログラミング用のポートをアプリケーションに組み込むことができます。4 ピン インターフェイスを 1 つ設けるだけで済み、ICSP インターフェイスのような回路制約はありません。

JTAG デバイス プログラミングは実際、4 ピンの TAP インターフェイスで標準の ICSP を使用します。適切な JTAG コマンド シーケンスによってトリガされた場合、TDI/TDO ピンと TCK ピンは PGD ピンと PGC ピンとして機能します。このピンリマッピングを除いて、JTAG インターフェイスを使う ICSP プログラミングは、標準 ICSP インターフェイスを用いた場合と全く同様に機能します。

TAP インターフェイスの切り換えに追加の時間がかかるために、JTAG デバイス プログラミングは、PGC ピンと PGD ピンを使う標準 ICSP プログラミングより若干時間がかかります。JTAG プログラミングでは、拡張 ICSP プログラミングは使用できません。

JTAG デバイス プログラミングに必要な手順を以下に示します。

1. MCHP_CMD 命令 (0x08) を命令シフトレジスタにシフト入力します。この命令によって 8 ビットのマイクロチップ コマンドレジスタが選択され、TDI ピンと TDO ピン間のシリアルアクセス用に接続されます。
2. JTAG_MUX 命令 (0x02) をマイクロチップ コマンドレジスタにシフト入力します。この命令は JTAG インターフェイスを ICSP 動作に切り換えます。このコマンドにより、TDI ピンと TDO ピンは PGD として、TCK ピンは PGC として機能します。

24.5 関連アプリケーション ノート

本セクションに関連するアプリケーション ノートの一覧を以下に示します。一部のアプリケーション ノートは dsPIC33F/PIC24H デバイスファミリ向けではありません。ただし概念は共通しており、変更が必要であったり制限事項が存在するものの利用が可能です。プログラミングと診断に関連する最新のアプリケーション ノートは以下の通りです。

タイトル	アプリケーション ノート番号
現在関連するアプリケーション ノートはありません。	

Note: dsPIC33F/PIC24H ファミリ向けのその他のアプリケーション ノートとサンプルコードは、マイクロチップ社のウェブサイト (www.microchip.com) をご覧ください。

24.6 改訂履歴

リビジョン A (2007 年 4 月)

本書の初版

リビジョン B (2010 年 2 月)

このリビジョンでの変更内容は以下の通りです。

- ファミリ リファレンス マニュアルのタイトルを dsPIC33F から dsPIC33F/PIC24H へ変更
- 本書内の「dsPIC33F」を全て「dsPIC33F/PIC24H」へ変更
- 図：
 - 図 24-6: デバイス ID レジスタを削除
- Note:
 - 本書の冒頭 (24.1 「はじめに」の前) にファミリ リファレンス マニュアルの一連のセクションとデータシートの活用方法に関する情報を追加
- セクション：
 - 24.2 「インサーキット シリアル プログラミング™」における下記の不要ポイントを削除：元々 8 ビット PIC16 用に導入されたが、この方式は事実上マイクロチップ社製の全てのマイコンに使用されているため。
- 表：
 - 表 24-3 における下記の図 24-6: デバイス ID レジスタへの言及を削除：
図 24-3 に dsPIC33F/PIC24H のビット フォーマットを示します。IEEE によって割り当てられた 11 ビットの製造者 ID (マイクロチップ社は 29h)、デバイス製品番号、デバイス リビジョン番号で構成されています。
例えば、dsPIC33FJ256GP710 の JTAG ID は、以下の通りです。
製造者 ID = 0x29
製品番号 = 0XFF
シリコン リビジョン = A2
JTAG ID = 0x03FC2053
 - 表 24-3 に以下の参照を追加 (表内の「デバイス ID レジスタ」の「機能」列参照): デバイス ID レジスタと JTAG ID レジスタの詳細は、『dsPIC33F/PIC24H フラッシュ プログラミング仕様』 (DS70152) を参照してください。
- 上記に加えて、表現と体裁の変更等、本書全体の細部を修正。

ISBN: 978-1-60932-873-3

NOTES: