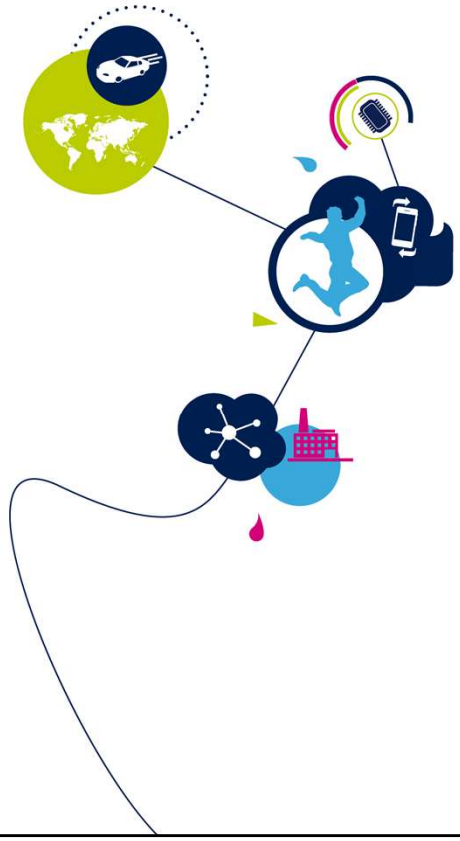
STM32MP1 - DBG

デバッグおよびトレース
1.0 版



こんにちは、STM32 デバッグおよびトレースインタフェースのプレゼンテーションへようこそ。ここでは、STM32MP1 デバイスが提供するデバッグとトレースの機能を説明します。

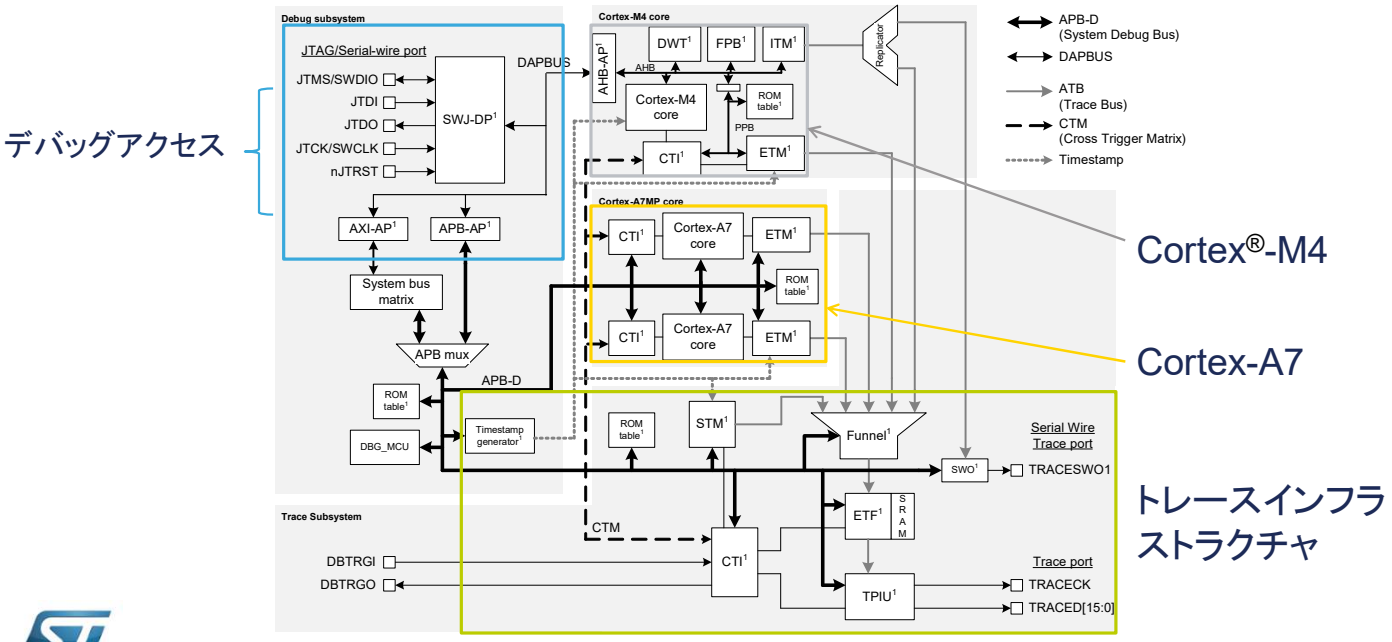
概要

- STM32MP15 は、豊富なデバッグおよびトレース機能を搭載
  - プログラムを RAM または Flash メモリにダウンロード
  - メモリやレジスタの内容の調査
  - ブレークポイントを挿入し、プロセッサを停止
  - プログラムの実行または シングル ステップ実行
  - プログラムの実行をトレース
- Arm® CoreSight™ アーキテクチャに基づく
  - 広範囲の互換ツール
  - 標準インタフェース（JTAG／シリアルワイヤ）

life.augmented

STM32MP1 マイクロプロセッサには、STM32 ファミリの MCU で提供されている使い慣れたデバッグ機能（Flash ダウンロード、ブレークポイントによるデバッグ、レジスタおよびメモリの参照、シリアルワイヤトレース）がすべて組み込まれており、STM32MP1 ファミリーのマルチコアバージョンのクロストリガ機能のほか、広バンド幅の命令トレース機能が追加されています 。デバッグおよびトレースのインフラストラクチャには、ほとんどのツールプロバイダによって十分にサポートされている Arm CoreSight™ 標準を使用しています。

デバッグアーキテクチャ



1.CoreSight™ コンポーネント

デバッグとトレースのインフラストラクチャは以下の 4 つの個別の機能ドメインで構成されています。

・デバッグアクセスインフラストラクチャ – これには、デバッグポート（SWJ-DP）とアクセスポート（AP）が含まれており、外部デバッガがターゲットのトレースおよびデバッグ機能にアクセスできるようにします。

・トレースインフラストラクチャ – これには、シリアル（SWO）およびパラレル（TPIU）トレースポート、トレースフローの平滑化に使用されるトレース FIFO（ETF）、各ソースからのトレースを単一フローに結合するトレースファネルが含まれています。システムトレースモジュール（STM）もあり、ソフトウェアで生成されたデバッグ情報とハードウェアイベントをトレースできます。

・Cortex-A7 コア – ここには、プロセッサ（シングルコアまたはデュアルコア）および組込みトレースモジュール（ETM）が含まれています。

・Cortex-M4 コア – ここには、プロセッサならびに、関連するデバッグおよびトレースユニット（DWT、FPB、ITM、ETM）が含まれています。

さらに、下記のようなシステムデバッグ機能があります。

・クロストリガインタフェースおよびクロストリガマトリックス（CTI、CTM） – これらにより、両方のコアの同時停止、トレースのトリガなどが可能になります。

・グローバルタイムスタンプジェネレータ – さまざまなトレースソースに共通の時間基準を提供します。

・DBG_MCU – デバッグ中のタイマの停止などの独自機能を提供します。

・外部トリガ入出力 – これにより、外部信号がデバッグまたはトレースをトリガするようにできたり、外部機器やコンポーネントを同期するためのトリガパルスを生成したりできます。

デバッグポート

- デバッガ は JTAG/SWD デバッグポートを介して STM32MP15 にアクセス。
  - 標準 5ピンの JTAG ポートはバウンダリスキャンと DFT にも使用。
  - シリアルワイヤデバッグ（SWD）ポートは JTAG ポートピンの 2 つを使用。

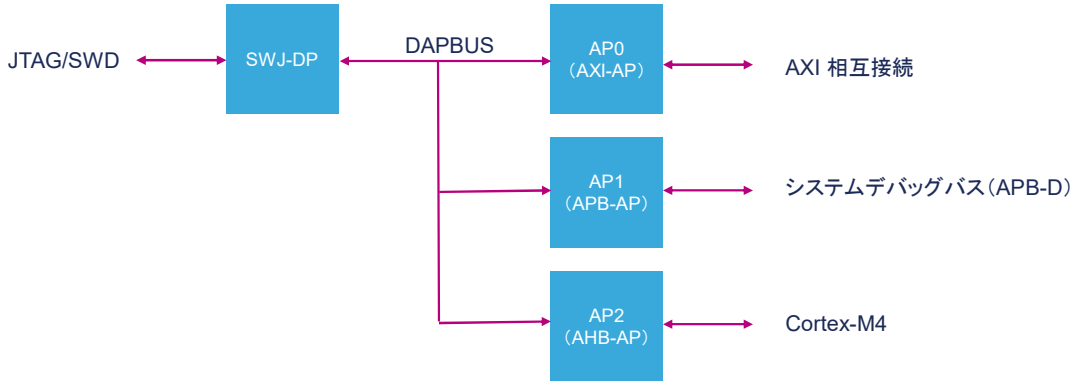
ピン	JTAG デバッグポート	SW デバッグポート
JTMS-SWDIO	テストモード選択	シリアルワイヤデータ（入出力）
JTCK-SWCLK	テストクロック	シリアルワイヤクロック
JTDO	テストデータ出力	-
JTDI	テストデータ入力	-
nJTRST	テストリセット	-

life.augmented

デバッグポートは、すべての STM32MP1 パッケージの専用ピンで使用できます。
シリアルワイヤデバッグでは、SWDIO（JTMS）の入力上でデバッガによって駆動される特別なシリアルコードが使用されます。これは、SWD モードに切り替わる SWJ-DP によって認識されます（リセット後、JTAG モードがデフォルトで設定されます）。
ST-Link、およびほとんどのサードパーティのデバッグアダプタ（Ulink など）は、シリアルワイヤデバッグをサポートしています。

アクセスポート

- 3 つのアクセスポート（AP）がバスマスタとして機能し、デバッガがメモリおよびレジスタに対する読出し／書込みトランザクションを実行可能。



AP0：AXI 相互接続へのアクセスを可能にします。これにより、デバッガは直接すべてのメモリとペリフェラルのレジスタにアクセスできます。
AP1：システム APB デバッグバスに搭載されているデバッグおよびトレース機能、すなわち、Cortex-A7 のデバッグ機能およびトレースサブシステムへのアクセスを可能にします。
AP2：内部 AHB バスを介して、Cortex-M4 プロセッサコアに組み込まれているデバッグおよびトレース機能へのアクセスを可能にします。
どちらのプロセッサで実行されているアプリケーションも、システムデバッグバスにあるデバッグ機能にアクセスできます。それらは統合アドレス空間にマッピングされているためです。この機能には、トレースサブシステム（STM、TPIU、TSGEN、ETF）および Cortex-A7 機能（ETM、CTI、DBG）が含まれます。ただし、Cortex-M4 のプライベートバスにある機能には Cortex-M4 のみがアクセスできます。

認証とセキュリティ

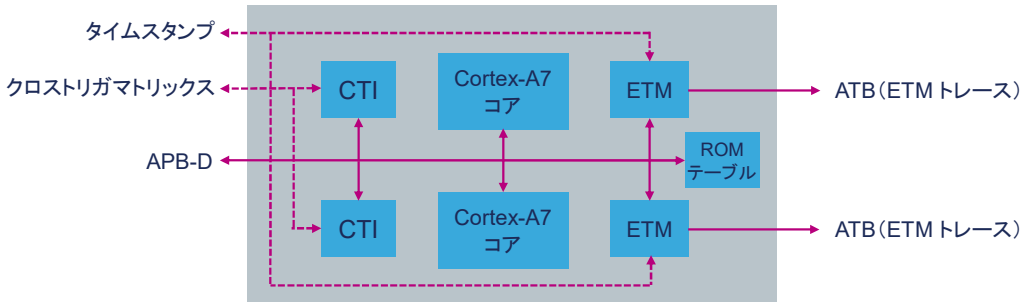
- ブートおよびセキュリティ(BSEC)ユニットからの 6 つの認証信号によって、デバッグ機能へのアクセスを制御。
  - DEVICEEN:外部デバッグポートを介したデバッグコンポーネントへのアクセスを制御。
    - 0:外部デバッガはデバッグコンポーネントへのアクセス不可。システム相互接続へのデバッガのアクセスは影響は無し。
    - 1:外部デバッガはデバッグコンポーネントへアクセス可能。
  - DBGEN:すべてのデバッグおよびトレース機能に対してグローバルに有効。
    - 0:すべてのデバッグ機能が無効。デバッグコンポーネントへの外部アクセスおよびソフトウェアアクセスを維持。システム相互接続への外部アクセスは無効。
    - 1:非セキュアモードのデバッグ機能が有効。非セキュアシステム相互接続への外部アクセスは可能。セキュアモードのデバッグ機能およびセキュアシステム相互接続への外部アクセスは SPIDEN 信号の状態に依存。
  - SPIDEN:DBGEN = 1 の場合にセキュア特権モードでのデバッグを有効化。
    - 0:デバッグ機能がセキュア特権モードで無効。セキュアシステム相互接続への外部アクセスは無効。
    - 1:デバッグ機能がセキュア特権モードで有効。セキュアシステム相互接続への外部アクセスは可能。
  - NIDEN:トレースと性能の監視が有効(非侵入型デバッグ)。
    - 0:トレース生成が無効。
    - 1:非セキュアモードのトレース生成が有効。セキュアモードのトレース生成は SPNIDEN 信号の状態に依存。
  - SPNIDEN:NIDEN = 1 の場合にセキュア特権モードでのトレースと性能の監視を有効。
    - 0:トレース生成がセキュア特権モードで無効。
    - 1:トレース生成がセキュア特権モードで有効。
  - DBGSWEN:自己ホスト型デバッグを有効。
    - 0:すべてのデバッグコンポーネントへのソフトウェアアクセスは無効。
    - 1:すべてのデバッグコンポーネントへのソフトウェアアクセスが有効。

life.augmented

認証信号の状態はブートおよびセキュリティ(BSEC)ユニットで設定されます。これらの信号のデフォルト状態は、デバイスの出荷時の状態(オープンまたはクローズ)によって決まります。この状態は、セキュアソフトウェアによって変更できます。
デバッガは、セキュア特権トランザクションを使用してセキュアアドレスにアクセスする必要があります。これらのトランザクションは、SPIDEN 信号がアサートされている場合のみ成功します。

Cortex-A7 T&D 機能

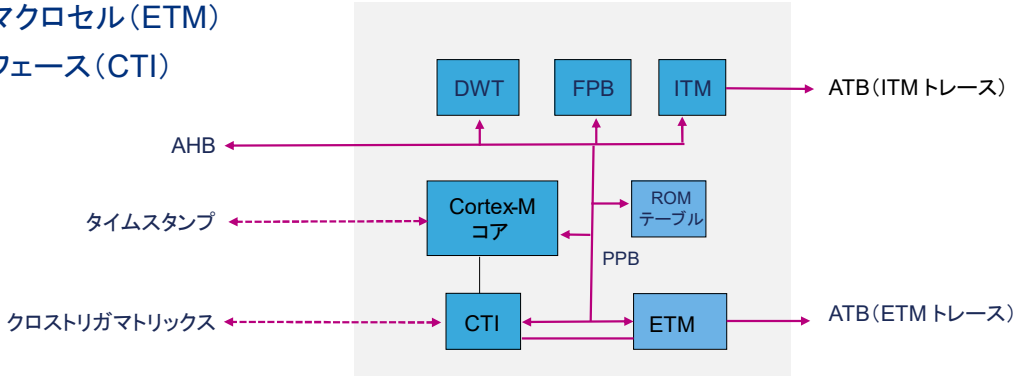
- Cortex-A7 には、以下のデバッグコンポーネントを含む。
  - コアあたり 1 つのデバッグユニット（DBG）
  - コアあたり 1 つの組込みトレースマクロセル（ETM）
  - コアあたり 1 つのクロストリガインタフェース（CTI）
  - ROM テーブル



Cortex-A7 コアのすべてのデバッグ関連レジスタは、アクセスポート AP1 を通じてシステムデバッグバス APB-D を介してアクセスされます。
ROM テーブルには、コア内の各デバッグコンポーネントのベースアドレスへのポインタが格納されています。ROM テーブルは、いくつかのデバッグツールでターゲット内の CoreSight™ インフラストラクチャのトポロジを自動的に検出するために使用されます。
デバッグユニット（DBG）には、デバッグモード中にプロセッサコアを制御するためのレジスタが含まれています。

Cortex-M4 T&D 機能

- Cortex-M4 には、以下のデバッグコンポーネントを含む。
  - システム制御空間（SCS）
  - データウォッチポイントおよびトレースユニット（DWT）
  - ブレークポイントユニット（FPB）
  - 計装トレースマクロセル（ITM）
  - 組み込みトレースマクロセル（ETM）
  - クロストリガインタフェース（CTI）
  - ROM テーブル



Cortex-M4 コアのすべてのデバッグ関連レジスタは、専用 AHB アクセスポート AP0 を介してアクセスされます。
ROM テーブルには、AP から見える各デバッグコンポーネントのベースアドレスへのポインタが格納されています。ROM テーブルは、いくつかのデバッグツールでターゲット内の CoreSight™ インフラストラクチャのトポロジを自動的に検出するために使用されます。
SCS（システム制御空間）には、デバッグモード中にプロセッサコアを制御するためのレジスタが含まれています。
その他のユニットは次のスライドで説明されています。

データウォッチポイントおよびトレースユニット

9

- DWT には 4つのコンパレータがあり、それぞれ以下の役割に使用可能。
  - ウォッチポイント
  - ETM トリガ
  - PC サンプリングトリガ
  - データアドレスサンプリングトリガ
  - データコンパレータ
  - クロックサイクルカウンタコンパレータ
- DWT はソフトウェアプロファイリングのための以下のカウンタも内蔵。
  - クロックサイクル数
  - フォールドされた命令数
  - ロードストアユニット(LSU)の動作数
  - スリープサイクル数
  - 命令当たりのサイクル数
  - 割込みオーバーヘッドの回数

データウォッチポイント(DWT)コンパレータは、以下の項目のうちの 1 つを、DWT_COMP レジスタに保持されている値と比較します。

- データアドレス
- 命令アドレス
- データ値
- サイクルカウント値(コンパレータ 0 のみ)

アドレス照合の場合、コンパレータはマスクを使用できるため、ある範囲のアドレスと照合することができます。
照合が成立すると、コンパレータは以下のうちの 1 つを生成します。

- 1 つ以上の DWT データトレースパケットで、次のものを 1 つ以上含むもの。
  - ・データアクセスを伴う命令のアドレス
  - ・アドレスオフセット(データアクセスアドレスのビット [15:0])
  - ・一致したデータ値
- PC 値またはアクセスしたデータアドレスのいずれかで発生するウォッチポイントデバッグイベント
- DWT ユニット外での一致を信号で伝える CMPMATCH[N] イベント

ブレークポイントユニット

- ブレークポイントユニット（FPB）を使用することでハードウェアのブレークポイントが設定可能。
  - このユニットには、命令フェッチアドレスを監視し、一致が検出されたときに、ブレークポイント命令を返すコンパレータを8 つ内蔵。
  - デバッグモードでブレークポイント命令が実行されると、プロセッサは停止。

life.augmented

Cortex-M4 ブレークポイントユニット（FPB）は、Flash メモリのパッチ当てもサポートしています。この機能は、指定されたアドレスの揮発性メモリを実行することによって、誤ったコードにパッチを当てることを目的としています。

計装トレース

- 計装トレースマクロセル（ITM）は下記の 4 つのソースからトレースパケットを生成。
  - ソフトウェアトレース：
    - 32 個のスティムラスレジスタのいずれかへのソフトウェア書込み
  - DWT からのハードウェアトレースパケット
    - これには、データトレースイベント、PC サンプル、カウンタのラップアラウンドなど。
  - ローカルタイムスタンプ
    - ITM の 21bitカウンタにより、前のパケットを基準とした各トレースパケットの相対タイムスタンプが提供。
  - グローバルタイムスタンプ
    - タイムスタンプは、TSGEN から受信したシステム全体の 64bitのタイムスタンプを使用して生成することも可能。
- トレースパケットは ATB トレースバスに出力。

life.augmented

ソフトウェアは、32 x 32bitの 計装トレースマクロセル（ITM）スティムラスレジスタのいずれかに直接書込みを行って、パケットを生成します。各ポートの許可レベルはプログラムで設定できます。ソフトウェアが有効なスティムラスポートに書き込むと、ITM は FIFO に書き込むパケットの中に、ポートの ID、書込みアクセスのサイズ、および書き込まれたデータを統合します。ITM は FIFO からトレースバスにパケットを出力します。スティムラスポートレジスタを読み出すと、ビット 0 のスティムラスレジスタのステータス（エンプティまたはペンディング）を返します。
パケットが複数のソースから同時に生成される場合、ITM はパケットの出力順番についてアービトレーションを行います。ここで、ソースは優先度の高いものから順に並べられます。
タイムスタンプジェネレータ（TSGEN）は、すべてのトレースパケットのタイムスタンプに 64bitの共通タイムベースを提供します。これにより、トレースアナライザは、トレースが生成された時間に応じて、さまざまなソースからのトレースを整列させることができます。ローカルタイムスタンプは同期されておらず、異なる周波数で実行される可能性があるため、トレースが生成された正確なタイミングを知ることができません。
注：Cortex-M4 は、グローバルタイムスタンプの最下位 48bitのみを使用します。

システムトレース

- システムトレースマクロセル（STM）は下記の 2 つのソースからトレースパケットを生成。
  - ソフトウェアトレース：
    - バスマスタ（CPU、DMA、デバッガ）は、65536 のスティムラスポートのいずれにも書き込み可能。
    - 各ポートには 64 x 16bitの仮想レジスタを含む。レジスタアドレスによって、生成されるトレースパケットのタイプを決定。
    - トレースパケットのタイプには、データ、フラグ、トリガがあり、タイムスタンプが含まれる場合と含まれない場合あり。
  - ハードウェアイベントのトレース：
    - 任意の DMA リクエスト／確認応答、および任意の割込みアサーション／デアサーションに対してトレースパケットを生成するようにプログラム可能。
- トレースパケットは ATB トレースバスに出力。

life.augmented

システムトレースマクロセル（STM）はソフトウェアの計装のために使用できます。Cortex-M4 には単純化されたソフトウェアトレースユニットである ITM が含まれているため、STM は主に Cortex-A7 を対象としています。それでも、STM は Cortex-M4 コアと DMA および MDMA エンジンからアクセスできます。デバッガによって使用されることもできます。

スティムラスポートへのアクセスは、«保証»または«タイミング不変»にできます。

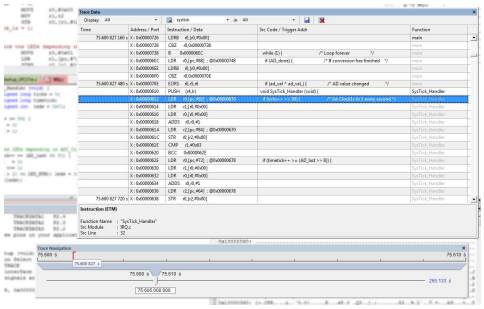
- 保証されたアクセスでは、常にトレースパケットを生成し、STM がそれを受け入れるまでアクセスを停止します。
- タイミング不変アクセスは、常に即座に終了するので、あまり侵入的になりませんが、STM のバッファがいっぱいで書込みを受け入れる準備ができていない場合、データは破棄され、パケットは生成されません。

トレースパケットには、それらを生成したマスタの ID が常に含まれます。

STM 制御レジスタはシステムデバッグバス（APB-D）を介してアクセスできます。

命令トレース

- 組込みトレースマクロセル（ETM）によって、ソフトウェアの実行を観察できるトレースパケットが生成。トレース情報の内容は以下。
  - 同一サイクルで実行される命令の数
  - プログラムフローの変更
  - プロセッサの現在の命令ステート
  - ロードおよびストア命令によってアクセスされるメモリ位置のアドレス
  - 転送のタイプ、方向、およびサイズ
  - 条件コードに関する情報
  - 例外情報
  - 割込み待ちステートに関する情報
- トレースパケットは ATB トレースバスに出力。

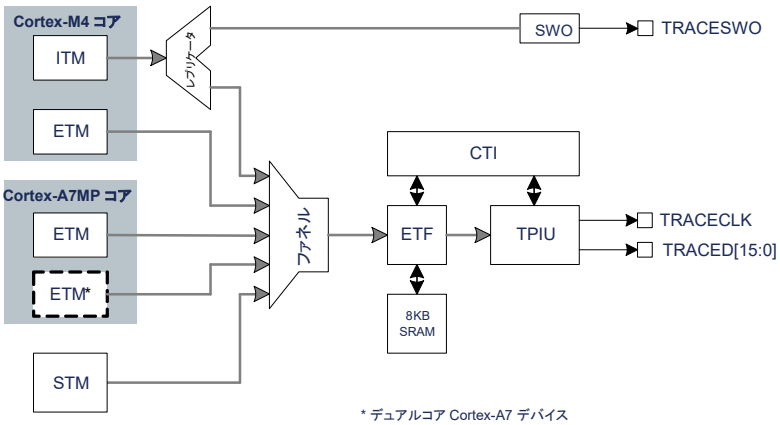


life.augmented

STM32MP1 シリーズでは、組込みトレースマクロセル（ETM）は命令トレース専用に設定されているため、トレース情報にデータアクセスは含まれません。

トレースインフラストラクチャ

- ITM および ETM からのトレース情報は、AMBA トレースバス（ATB）を介して 1 つ以上のトレースシンクにルーティング。
  - ルーティングは 2 つの ATB インフラストラクチャコンポーネントによって実行。
    - レプリケータ – 2 つの ATB ブランチにトレースパケットを複製
    - ファネル – 複数の ATB ブランチを 1 つに多重化



ITM、ETM、STM によって生成されたトレースストリームはトレースファネルで結合されます。一部のファネルパラメータは変更できます。たとえば、別の入力に切り替える前に 1 つの入力から受信するバイト数です – 発生する切り替えが少ないほど、オーバーヘッドは低くなりますが、遅延が増加します。トレースをフィルタリングすることもできます。たとえば、TPIU から ITM トレースを削除できます（代わりに SWO に出力できます）。

ITM（ETM ではなく）からのトレースをシングルワイヤトレースポートに送信できます。デュアルコアデバイスでは、両方のコアからの ITM トレースを SWO に送信し、SWO トレースファネルで結合することができます。ただし、SWO にはフォーマットがないため、トレースポートアナライザでトレースストリームを分離することはできません。そのため、SWO に出力する場合はファネルを使用して一度に 1 つの ITM を手動で選択することをお勧めします。両方が必要な場合は、TPIU を使うべきです。

トレースシンク

- トレースパケットは、次の 3 つの転送先すなわち「シンク」のいずれかに転送。
  - 組込みトレース FIFO（ETF）
    - これは、トレースパケットをサーキュラバッファに保存できる 8KB のメモリ。トレースはソフトウェアまたはデバッガによって読み出し可能。
  - トレースポートインタフェース（TPIU）
    - トレースパケットは、同期クロック信号とともに 16ピンパラレルポートを介してデバイスからストリーミングされます。これには、ULINKpro や DStream などのトレースポートアナライザのプローブの接続が必要。
  - シングルワイヤトレースポート（SWO）
    - ITM トレース（Cortex-M4 から）は、非同期プロトコル（NRZ またはマンチェスタ）を使用して SWO に向けて出力可能。これは、ST-Link や他のアダプタ、およびほとんどの商用デバッガツールを使って読み出し可能。

life.augmented

ETF は、トレースをオンチップで保存するためのトレースバッファとして使用できます。トレースはソフトウェアまたはデバッガによって読み出すか、トレースポート経由で一掃できます。サーキュラバッファとして構成されている場合、トレースは継続的に保存されるため、最新のトレースによって最も古いトレースが上書きされます。または、FIFO フルフラグを使用して、バッファがいっぱいになったときにトレースを停止し、特定の時点でトレースを取得できます。
また、ETF は TPIU へのトレースの流れを平滑化するように（ハードウェアモードで）機能します。トレースストリームは本質的にバーストする傾向があり、瞬間バンド幅はトレースポートのバンド幅よりもはるかに高いため、バッファはピークを吸収し、トレースポートの最大連続バンド幅へフローを調整する働きをします。

トレースポート

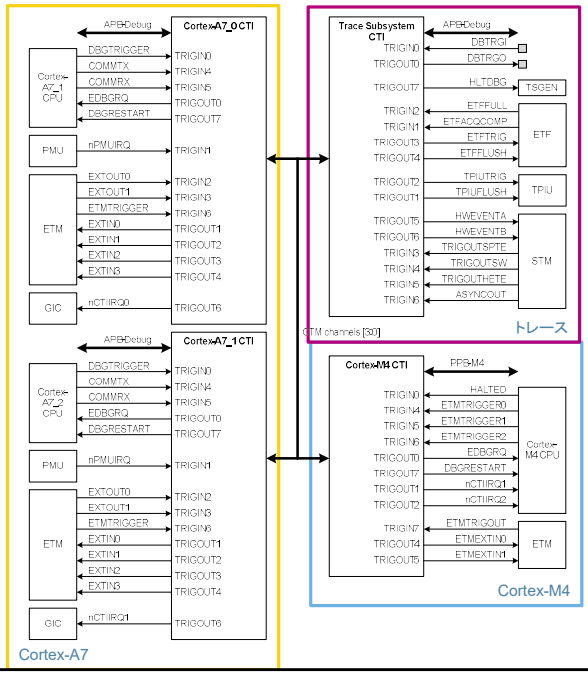
- TPIU パラレルポート
  - 1～16 本のデータピンとクロックをトレースに割当て可能（デフォルトでは GPIO）。
  - TRACECLK は DDR モードで最大 133MHz で動作可能。
  - ✓1Gbps の最大バンド幅（ULINKpro では 800Mbps）
- SWO シングルワイヤシリアルポート（Cortex-M4 のみ）
  - JTDO と多重化される 1 本の非同期データピン
  - 100Mbps のバンド幅（マンチェスタ符号化）

life.augmented

トレースポート幅は 1～4ピンにプログラムできます。バンド幅は、ピンの数と TRACECLK 周波数（RCC の分周器で選択可能）に比例してスケーリングされます。最大クロック周波数でのフルデュアルコア命令トレースには、最大バンド幅が必要になる可能性があります。フィルタとトリガをトレースソース（特に ETM）に適用することにより、トレースデータの平均量を削減し、クロックレートの低減やピン数の削減を実現できます。
TRACESWO ピンは、JTAG インタフェースの一部である JTDO 信号と多重化されます。したがって、シングルワイヤトレースはシリアルワイヤデバッグ（SWD）インタフェースが有効になっている場合にしか使用できません。

クロストリガインタフェース

- CTI は、トリガイベントを他のデバッグおよびトレースコンポーネントに伝播。
  - トリガイベントのソースになり得るもの:
    - データ・ウォッチポイント
    - ハードウェアブレークポイント
    - プロファイリングカウンタイベント
    - トレースバッファフル／エンプティ
    - 外部トリガ信号
    - プロセッサ停止／リスタート
  - 転送先でトリガイベントが引き起こすこと:
    - トレースの開始／停止
    - トレースバッファの一掃
    - プロセッサ停止／リスタート
    - 外部トリガ信号の出力
    - プロセッサ割込み



マルチコアデバイスでクロストリガを使用すると、両方のコアを同時に停止できます。1 つのコアがブレークポイントに達すると、その「停止」出力（デバッグモードに入ったことを示す）が他方のコアに伝播し、同様にデバッグモードに入ります。同様に、両方のコアを同時にリスタートできます。

クロストリガ機能を使用して、外部トリガ信号（これは、IO ピンの 1 つでのエッジの可能性があります）でプロセッサを停止することもできます。

プロセッサコアごとにクロストリガインタフェース（CTI）があり、また、トレースコンポーネント（ETF、TPIU、STM）および外部トリガ信号に接続されたシステム CTI もあります。

クロストリガ機能のいずれかを使用するには、それに応じてデバッガによって CTI をプログラムする必要があります。必要なトリガ入力信号（TRIGINn）とトリガ出力信号（TRIGOUTn）は、クロストリガマトリックス（CTM）に接続する必要があります。CTM は最大 4 つのチャネルで構成され、4 つの異なるイベントを並行して伝播できます。複数のトリガ入力は CTI で結合することができ、そうすることで結合された入力のどれか 1 つが、接続されたチャネルにイベントを発生させることができます。同様に、1 つのイベントが複数のアクションをトリガできるように、チャネルを複数のトリガ出力に接続できます。

DBGMCU

- 「MCU デバッグ」ブロックはデバイス固有のデバッグ機能を有効化。
  - デバイス ID
    - デバイス ID コードレジスタを読み取るための標準の場所
  - 低電力モードのエミュレーション
    - デバイスが低電力モード（SLEEP、STOP、STANDBY）になったときに、デバッグアクセスが引き続き可能となるように電力とクロックを維持。
  - デバッグモード時のペリフェラルクロックの「停止」
    - プロセッサが停止している間は、RTC、TIM、LPTIM、およびウォッチドッグ（IWDG、WWDG）タイマのカウンタ、ならびに SMBUS および FDCAN タイムアウトカウンタを停止。
  - ドメインデバッグクロックの有効化
    - 電力を節約するために、不要な場合はクロックを無効にして各電源ドメインのデバイスのデバッグが可能。
  - 外部トリガの方向
    - 双方向 TRGIO 外部トリガピンの方向（入力／出力）を制御。

life.augmented

DBGMCU はデバッグ APB バス上にあり、デバッガは APB アクセスポート AP2 を介してアクセスできます。また、デバッグ APB アドレス空間にあるプロセッサからもアクセスできます。
DBGMCU_IDC レジスタに、デバイス ID とバージョンコードが STM32 標準形式で提供されています。この情報は、デバッグポート（DP_TARGETID レジスタ – 外部デバッガーのみアクセス可能）およびシステムデバッグ ROM テーブルレジスタ（SYSROM_PIDR [2:0] – ソフトウェアでもアクセス可能）でも入手できます。
低電力モードエミュレーションとは、低電力モードに入った場合にデバッガ接続が失われないことを意味します。これにより、低電力移行コマンド（WFI/WFE など）を while（）ループで置き換える必要がなくなります。終了時に、デバイスは、エミュレーションがアクティブではなかった場合と同じ状態になります（低電力モードエミュレーション中にデバッガによって行われた変更を除いて）。
ペリフェラルクロックの停止は、デバッグ中にウォッチドッグタイムアウトがデバイスをリセットしないようにするのに特に役立ちます。デバッガでウォッチドッグを再設定する必要はありません。また、タイマ値を検査し、「通常の」動作が再開されるまで、対応する割込みを一時停止できます。
デバッグクロックイネーブルビットは、必要なときにのみデバッグブロックがクロック供給されるようにします。これにより、不必要な電力消費が回避されます。これは、DAP を除いて、すべてのブロックがゲートされていないドメインクロックでクロック供給されるからです。
特定のパッケージでは、TRGIN ピンと TRGOUT ピンは用意されておらず、双方向ピンのみが使用され、TRGOEN ビットを使用して方向を選択する必要があります。