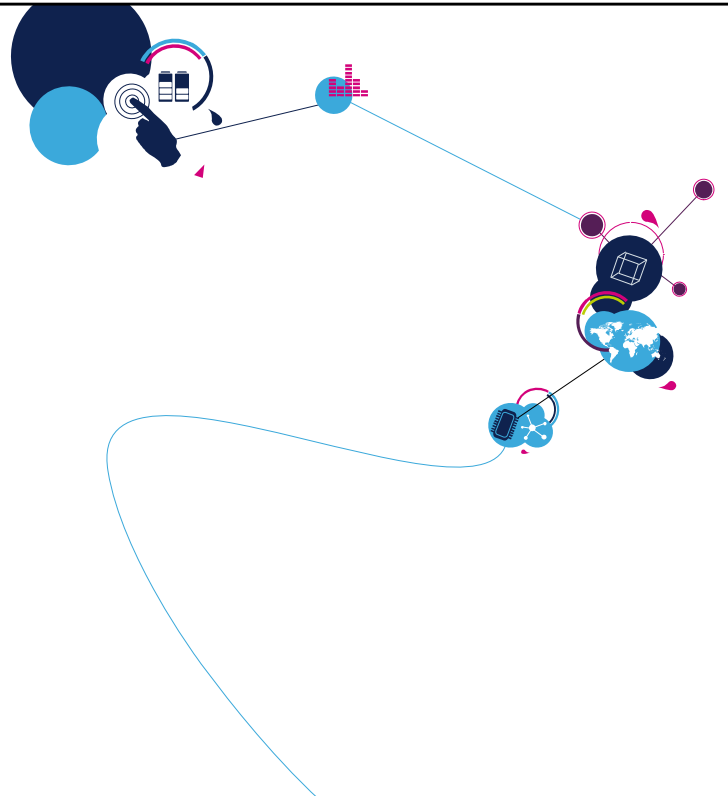


STM32G0 - DBG

デバッグとトレース

レビジョン 1.0



STM32 デバッグインタフェースのプレゼンテーションへようこそ。STM32G0 デバイスが提供するデバッグ機能について説明します。



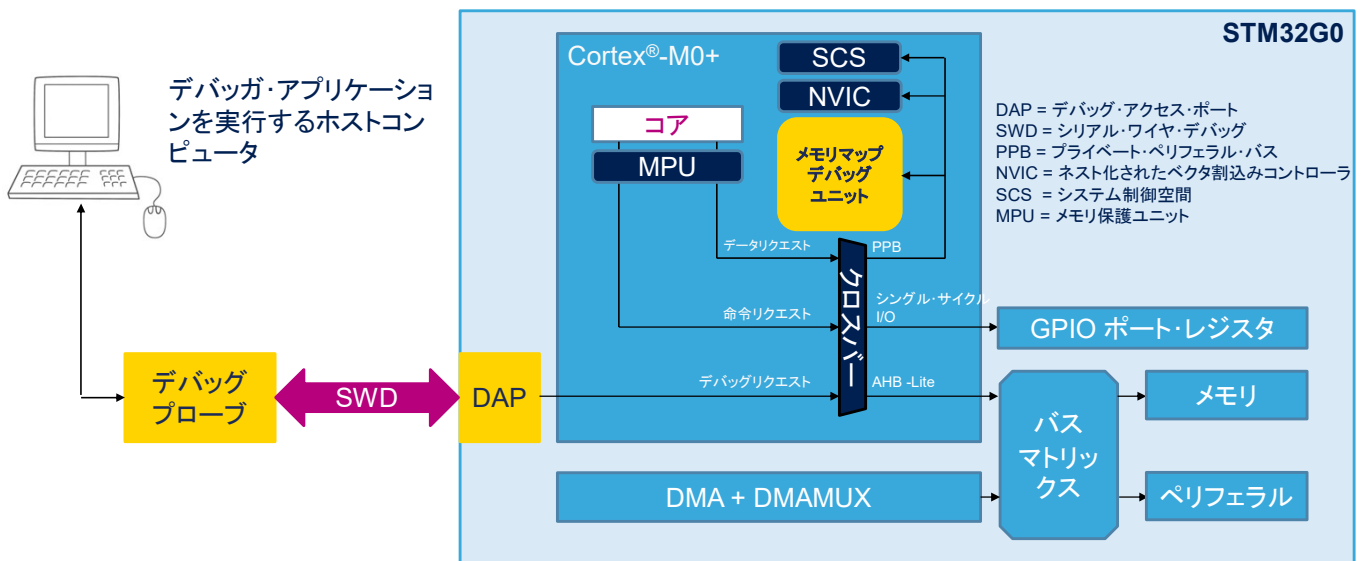
- STM32G0はデバッグに充実したサポートを提供
 - プログラムをRAMまたはFlashメモリにダウンロード
 - メモリとレジスタの内容を確認
 - ブレークポイントを挿入してプロセッサを停止
 - プログラムを実行またはプログラムを通じてシングル・ステップを実行
- Arm® CoreSight™ アーキテクチャ・ベース
 - 幅広い互換ツール
 - ST-LINKなど、プローブをデバッグするためのシリアルワイヤデバッグ標準インターフェース



STM32G0 には、STM32 ファミリのマイクロコントローラが提供されるお馴染みのデバッグ機能 (Flash ダウンロード、ブレークポイントデバッグ、レジスタ表示とメモリ表示) がすべて内蔵されています。デバッグインフラストラクチャでは、ほとんどのツールプロバイダでサポートされている ARM® CoreSight™ 規格を採用しています。

デバッグ・アーキテクチャ

3



デバッグアクセスポート(DAP)を使用すると、外部デバッグプローブは、Cortex®-M0+ コアからもアクセス可能な任意のメモリマッピングリソースにアクセスできるようになります。

メモリ保護ユニット(MPU)は DAP により開始されたリクエストを傍受することはありません。

デバッグプローブを Cortex®-M0+ に接続するのに使用されるシリアルワイヤデバッグ(SWD)プロトコルは、2本のワイヤに依存し、そのローピンカウントパッケージ(64、48、32 または 28 ピン)により、STM32G0 に適しています。

デバッグ・ポート

4

- デバッグはSWDデバッグ・ポート経由でSTM32G0にアクセス
 - シリアル・ワイヤ・デバッグ (SWD) ポートでは2つのポート・ピンのみ使用
 - デバッグが必要ではない場合、これらのデバッグ・ピンは機能的な用途のために再割当て可能

SWD	PA13	PA14
ピンアウト	SWDIO	SWCLK

- リセットすると、これらのピンはSWデバッグのオルタネート機能として設定され、PA13ピンの内部プルアップおよびPA14ピンの内部プルダウンが有効



デバッグの設定には、ピン PA13 および PA14 をシリアルワイヤデバッグ (それぞれ SWDIO および SWCLK) に割り当てる必要があります。

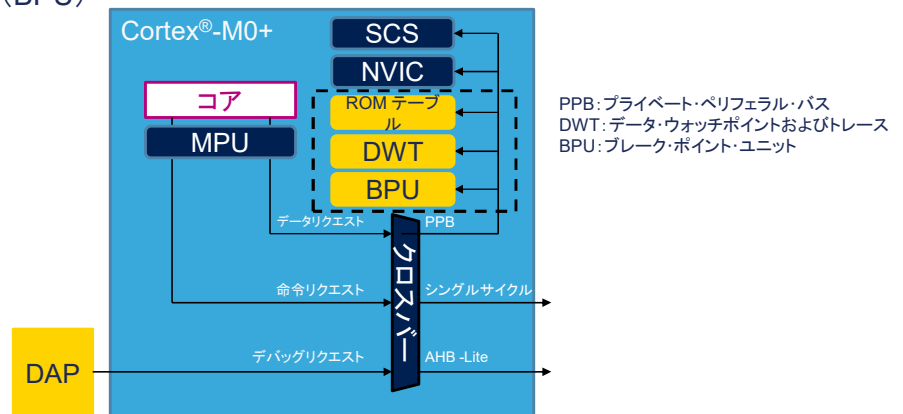
ST-Link ならびに大半の 3rd パーティデバッグアダプタ (Ulink など) は、シリアルワイヤデバッグに対応しています。

Cortex-Mメモリ・マップ・デバッグ・ユニット

5

- Cortex-Mコアは、以下のデバッグユニットを内蔵:

- 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 (BPU)
- ROMテーブル



デバッグプロセスに関わるすべてのユニットには、コアと DAP の両方からプライベートペリフェラルバスを通じてアクセスできるメモリアップレジスタがあります。

デバッグは、プロセッサの実行中にメモリマップリソースにアクセスできます。たとえば、プロセッサが命令を実行している間に、プライベートペリフェラルバスに接続されているブレークポイントユニットにアクセスし、デバッグによってブレークポイントを設定できます。ROM テーブルには、コアから認識でき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含まれています。一部のデバッグツールでは、ターゲットの CoreSight™ インフラストラクチャのトポロジを自動検出するために使用されます。

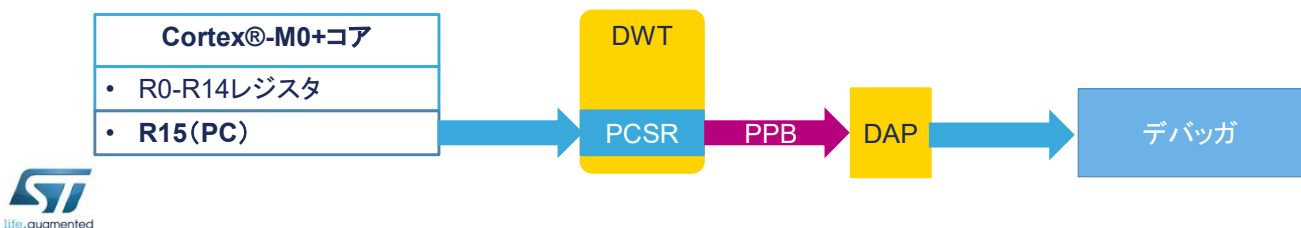
DAP には、ROM テーブルを指す読出し専用レジスタが含まれています。一部のデバッグツールでは、ターゲットの CoreSight™ インフラストラクチャのトポロジを自動検出するために使用されます。SCS (システム制御空間) は、CPU のリファレンスとレビジョンを示す CPUID レジスタを含むレジスタのブロックであり、デバッグが停止モードの開始と終了を制御するために使用します。

その他のユニットについては、以降のスライドで説明します。

データ・ウォッチポイントおよびトレース・ユニット

6

- DWTは命令アドレスとデータ・アドレスを比較できる2つのコンパレータを提供
 - 選択可能な方向(読出し、書込み、またはその両方)で特定のデータにアクセスしようとする試みを検出するのに非常に便利
 - 命令アドレスの比較では、ウォッチポイント・イベントはフェッチ時に取得
- PCサンプリング・レジスタ(PCSR)が独自のトレース機能を提供



DWT は、現在のデータまたは命令アドレスと、デバッガによってプログラムされたコンパレータの内容との一致によって発生するウォッチポイントイベントのトリガに使用されます。

アドレス照合の場合、コンパレータはマスクを使用することができるため、ある範囲のアドレスと一致します。

照合が成立すると、コンパレータは PC 値またはアクセスしたデータアドレスのいずれかでウォッチポイントデバッグイベントを生成します。

BPU は、命令が実行ユニットに入ろうとしたときにブレークポイントイベントが発生することから、命令ブレークポイントの実装により適しています。したがって、フェッチされた命令が分岐によって破棄された場合、ウォッチポイントイベントが発生する間、ブレークポイントイベントは発生しません。

R0 などのコアレジスタを読み書きするには、デバッガはコアを停止する必要があります。ただし、プログラムカウンタである R15 の現在の値は、PCSR と呼ばれる DWT に含まれているメモリマップレジスタで読み出すことができます。デバッガは、Cortex®-M0+ コアを停止せずに PC 値を読出しできます。

ブレークポイント・ユニット

7

- BPUを使用することでハードウェアのブレークポイントを設定することが可能。
 - このユニットには、一致が検出されたときに、命令フェッチアドレスを監視し、ブレークポイント命令を返すコンパレータを4つ内蔵
 - ブレークポイント命令の実行時には、プロセッサはデバッグ・モードで停止
 - ブレークポイント・ユニットは、STM32G0のFlashメモリに対応する0~0x1FFFFFFFの範囲内のアドレス比較のみをサポート
 - アドレス0x20000000以上では、RAMからコードを実行する場合に、ソフトウェア・ブレークポイントを使用可能



STM32G0 では、4 つのハードウェアブレークポイントをサポートしています。これらは不揮発性メモリにブレークポイントを設定するため、デバッガによって使用されます。

ブレークポイントユニットは、STM32G0 の Flash メモリに対応する 0~0x1FFF_FFFF の範囲内のアドレス比較のみをサポートします。アドレス 0x2000_0000 以上では、RAM からコードを実行する場合に、ソフトウェアブレークポイントを使用できます。ソフトウェアブレークポイントを使用すると、デバッガは、ユーザが停止したい命令を BKPT という専用の命令に置き換えます。

- 「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・レジスタを読み出すための標準の場所
 - 低消費電力モードのエミュレーション
 - デバイスが低消費電力モード (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源とクロックを維持
 - 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 タイマのカウンタを停止



DBGMCU はデバッグ APB バス上にあり、APB アクセスポート AP2 経由でデバッガからアクセスできます。また、デバッグ APB アドレス空間のプロセッサからもアクセス可能です。

DBG_IDCODE レジスタは、STM32 標準フォーマットのデバイス ID とレビジョンコードを提供します。このコードは、ソフトウェアデバッグポート (2 本のピン)、またはユーザソフトウェアによってアクセスできます。

低消費電力モードのエミュレーションは、低消費電力モードに入ってもデバッガとの接続が失われないことを意味します。この機能により、低消費電力移行コマンド (WFI/WFE など) を while() ループに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なかった場合と同じ状態になります (低消費電力モードエミュレーション中にデバッガが行った変更を除く)。

ペリフェラルクロックのフリーズは、デバッガを使用してウォッチドッグを再設定する必要がなく、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するのを防ぐのに特に便利です。また、タイマ値の検査が可能になるだけでなく、対応する割込みを「通常の」操作が再開するまでサスペンドすることが可能になります。