第2章 Cortex-M33搭載STM32H5を例に

TrustZoneに対応したSTマイコンのセキュリティ

矢郷 洋一

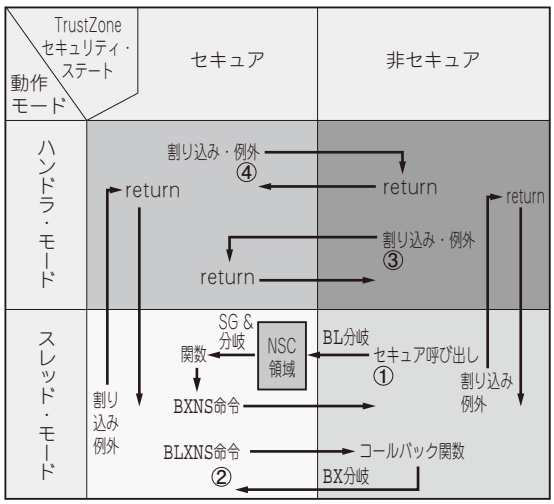

図1 TrustZoneセキュリティ・ステートが切り替わる要因

Cortex-MとTrustZoneとの関係

● TrustZoneの仕組み

TrustZoneを端的にまとめると，次のような技術です．

- アドレス空間上の任意の位置に，任意の大きさでアクセスしづらい領域を作る
- そこで重要な資源（メモリ，ペリフェラル）や情報を領域外アクセスから保護する
- また，そこで重要なソフトウェア処理を実行して処理の状態や結果を秘匿（ひとく）する

Armの用語では，アクセスしづらくした領域のことをセキュア領域と呼び，これまで通り自由にアクセスできる領域のことを非セキュア領域と呼びます．

セキュア領域だけあれば安全という訳ではなく，これまで通り自由にアクセスできる通常領域（非セキュア領域）とアクセスしづらくしたセキュア領域を2つ用意して，アドレス空間の各アドレスへのアクセスのしやすさ/しづらさに段階を設けてセキュリティを作る仕組みがTrustZoneです．

▶セキュア/非セキュアな状態をハードウェアで切り替える

Cortex-MのTrustZoneには，TrustZoneセキュリティ・ステートというハードウェア・ステートがあります．通常の状態では非セキュア・ステートで動作します．非セキュア・ステートでアクセスできるのは非セキュア領域のみです．セキュア・ステートに切り替わるとセキュア領域にアクセスできるようになります．TrustZoneセキュリティ・ステートが切り替わる要因は**図1**に示すように4つあります．

①非セキュア・ステートでセキュア呼び出し関数を実行
②セキュア・ステートで非セキュア・コールバック関数を実行
③非セキュア・ステートで割り込み・例外ハンドラを実行中に高優先度のセキュア割り込み・例外が発生
④セキュア・ステートで割り込み・例外ハンドラを実行中に高優先度の非セキュア割り込み・例外が発生

▶マイコン・ファームウェアの動作形態

ユーザは，TrustZone対応マイコンでファームウェアを作る場合に，**図2**の作法に従う必要があります．

- リセットがかかるとセキュア領域で起動する
- セキュア領域でセキュリティ設定と必要に応じて初期化処理を実行した後，非セキュア領域に移行する
- 非セキュア領域で非セキュア・ステートで使うペリフェラルと割り込みを設定する
- 非セキュア領域でmainループが動作する
- セキュア領域にセキュア呼び出し関数を必要な数だけ用意する
- セキュア領域に配置されている重要な資源や情報にアクセスし，また，セキュア領域で重要なソフトウェア処理を実行する場合，非セキュア領域からセキュア呼び出し関数を実行する．セキュア呼び出し関数を経由しないでセキュア領域に無理矢理アクセスするとフォールト例外が発生する