STM32 ネスト化されたベクタ割込みコントローラ（NVIC）のプレゼンテーションへようこそ。このコントローラの機能について説明します。

概要

- NVICはCortex®-M0+ CPUに内蔵:
  - 32のマスカブル割込みチャネル
  - 4つのプログラム可能な優先順位レベル
  - 遅延時間の少ない例外および割込み処理
  - 電源管理制御

アプリケーション側の利点

- 動的制御可能な優先順位レベルをサポート
- 割込みリクエストへの高速応答
- 再配置可能なベクタ·テーブル

life.augmented

割込みコントローラは Cortex®-M0+ CPU に内蔵され、プロセッサコアとの密接なカップリングを実現しています。
主な機能:

- 32 の割込みソース
- 4 つのプログラム可能な優先順位レベル
- 遅延時間の少ない例外および割込み処理
- 自動ネスティング
- 電源管理制御

アプリケーションは、割込みレベルの動的優先順位付け、低遅延応答とテールチェインによるリクエストへの高速応答、およびベクタテーブルの再配置という利点を活用できます。

主な機能

- 割込みリクエストへの高速応答
  - NVICの割込みリクエスト･エントリ数は32だが、STM32G0のSYSCFGモジュールにより、STM32G0に実装されている**割込みイベントの数は32以上。**
    - Arm® Cortex®-M0+制限
    - STM32G0では割込みイベントの数が32を超えているため、SYSCFGユニットで複数の割込みを同じ割込みラインに結合
    - ユーザ･ソフトウェアは、SYSCFGモジュールのITLINExレジスタを読み出すことによって、割込みをリクエストしたペリフェラルを迅速に特定可能
- 割込みの動的な再優先度付け。
- 割込みベクタ･テーブルの動的な再配置

life.augmented

NVIC は割込みリクエストに対して高速に応答するので、アプリケーションは速やかに受信イベントを処理できます。ARM V6-M は、NVIC の割込みリクエストの数を 32 に制限しています。
ただし、STM32G0 に実装されている割込みイベントの数は 32 より多いです。SYSCFG と呼ばれる別のユニットが、複数の割込みを同じ割込みラインに結合する役割を持ちます。ソフトウェアは、SYSCFG モジュールの ITLINE レジスタを読み出すことによって、割込みをリクエストしたペリフェラルを迅速に特定できます。
各割込みリクエストに割り当てられている優先順位はプログラム可能であり、動的に変更できます。
割込みベクタテーブルは再配置することもでき、システム設計者は、アプリケーションのメモリレイアウトに合わせて、割込みサービスルーチンの配置を調整できます。たとえば、ベクタテーブルを RAM に再配置できます。

優先順位処理

- Cortex®-M CPUの例外管理では値が小さいほど優先順位が高い

例外ソース	優先順位レベル	
リセット	-3	ハードコードされた固定優先順位
ノンマスカブル割込み（NMI）	-2	
ハードフォールト	-1	
他の例外（次を含む）: - **ペリフェラル割込み** - ソフトウェアの例外	0～3 のレベルをプログラム可能	

life.augmented

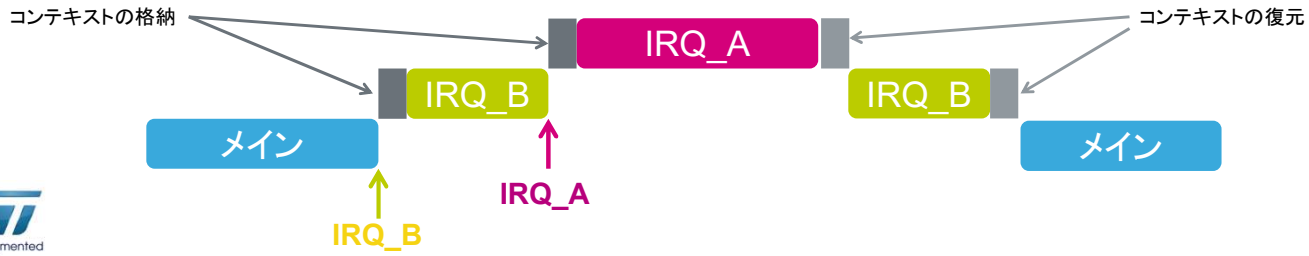
ソフトウェアは、各割込みに加えて、リセット、NMI およびハードフォールトを除くすべての例外ソースに対する優先順位レベルを割り当てる役割を持っています。
スーパーバイザコール命令の実行と同時にペリフェラル割込みがリクエストされた場合は常に、これらのハードウェアとソフトウェアの例外の相対的優先順位によって、どれが最初に処理されるかが決定されます。
STM32G0 では、NMI は、SRAM パリティエラー、Flash ダブル ECC エラー、またはクロック障害のいずれかの原因により発生します。
32 種類のペリフェラル割込みリクエストのいずれかの優先順位は、Cortex®-M0+ NVIC レジスタにある専用の優先順位フィールドにプログラム可能です。

テールチェインとネスティング

- テールチェインとネスティングのメカニズムを説明するために、次のペリフェラル割込みソースを想定

割込みソース	優先順位レベル
IRQ_A	0
IRQ_B	1

- プリエンプションと割込みネスティング



NVIC は、例外を効率的に処理するためのさまざまな機能を提供します。
割込みの処理中にそれより優先順位の高い新しいリクエストを受信した場合、新しい例外が処理中の例外に代わって優先権を得ます。これは、ネスト化された例外処理と呼ばれます。優先順位の高い例外が処理された後に、その前に処理中だった例外ハンドラが実行を再開します。
コンテキストは、Cortex®-M0+ に存在するマイクロコードにより自動的に現在のスタックにプッシュされ、割込みからの復帰時に復元されます。

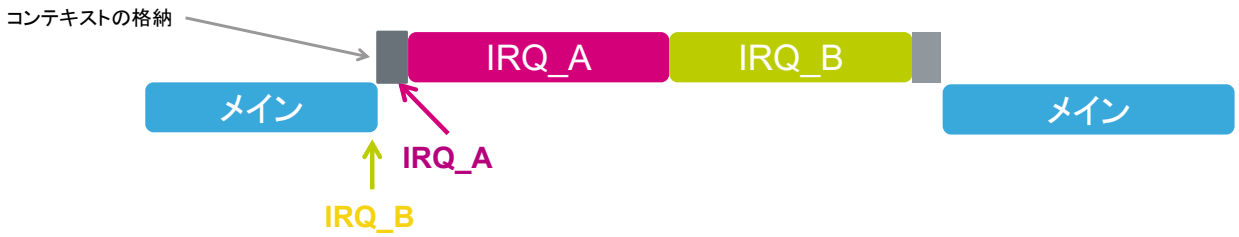
例外の開始と復帰

- テールチェイン
  - 例外ハンドラが完了したとき、ペンディングされている割込みが存在する場合はコンテキストの格納がスキップされ、前のハンドラが完了すると、制御は直ちに新しい例外ハンドラに転送。

割込みハンドラの実行中にそれ以下の優先順位の割込みリクエストが発生した場合、そのリクエストはペンディングされます。現在実行中の割込みハンドラが完了した後、割込みの遅延を少なくするために、コンテキストの保存と復元のプロセスはスキップされ、制御は直ちに新しい例外ハンドラに直接転送されます。
したがって、実行中の割込みより優先順位が低い（優先順位値が大きい）割込みが連続して発生する場合、それらは数クロックサイクルという非常に短い遅延で連鎖します。

例外の開始と復帰

- 後着
  - 例外を処理するために状態を保存している間にさらに優先順位の高い例外が発生した場合、プロセッサは直接優先順位の高い例外の処理に切り替え



- 復帰
  - 例外ハンドラが完了したとき、他の例外がペンディングされていなければ、プロセッサはスタックをポップし、割込みが発生する前のプログラム状態を復元

life.augmented

プロセッサは、割込みを受信すると、割込みハンドラを実行する前に、まずプログラムコンテキストを保存します。プロセッサがこのコンテキスト保存動作を実行している間にそれより優先順位の高い割込みを受信した場合、プロセッサはプログラムコンテキストの保存を完了した後、直接優先順位の高い割込みの処理に切り替えます。その後、IRQ_B 割込みサービスルーチンを実行する前に、テールチェインが使用されます。
すべての例外ハンドラが実行されて、他の例外がペンディングされていない場合、プロセッサは、スタックから前のコンテキストを復元し、通常のアプリケーション実行に復帰します。

NVICのヒントとコツ

- ソフトウェアで正しく整列されたレジスタ·アクセスを使用
- 割込みは無効であってもペンディングされる可能性
  - 割込みを無効にしても、プロセッサが割込みを受信しないだけ
- ベクタ·テーブルを再配置する前に、すべての有効な割込みに対して新しいエントリが正しくセットアップされていることを確認
  - これにはフォールトハンドラとNMIを含む
  - ベクタ·テーブルを再配置するためにVTOR レジスタをプログラミングする前にこれを実行

life.augmented

NVIC レジスタにアクセスする場合、コードで正しく整列されたレジスタアクセスを使用していることを確認します。NVIC レジスタだけでなく、Cortex®-M0+ でメモリにマップされたすべてのレジスタで、整列されていないアクセスはサポートされていません。
割込みは、発生元がサービスを要求した時点でペンディングされます。割込みを無効にしても、プロセッサが割込みを受信できなくなるだけです。割込みベクタを有効にする前に、関連する割込みフラグがクリアされていることを確認します。
VTOR レジスタを使用してベクタテーブルを再配置する前に、フォールトハンドラ、NMI、およびすべての有効な割込みが新しい配置先で正しくセットアップされていることを確認します。

関連ペリフェラル

- タイマにリンクされている次のペリフェラルのトレーニング資料を参照
  - SYSCFG
    - 特に、ペリフェラルによって生成された割込みソースを、NVICに接続された割込みリクエスト信号に結合する役割を持つ
  - Cortex®-M0+
    - CPUはソフトウェア例外とハードウェア例外の両方の処理に使用される例外メカニズムを実装している

life.augmented

NVIC は、SYSCFG モジュールおよび Cortex-M0+ CPU にリンクされています。詳細については、関連するプレゼンテーションを参照してください。

参照

- 詳細については、次の文書を参照：
  - PM0223 Cortex®-M0+プログラミングマニュアル
  - STM32G0リファレンスマニュアル

life.augmented

詳細については、Cortex®-M0+ コアのプログラミングマニュアルおよび STM32G0 のリファレンスマニュアルも参照してください。