

# コンピューターの高度な概念: 用語集パート 2

この記事は、インテル® デベロッパー・ゾーンに公開されている「[Advanced Computer Concepts for The \(Not So\) Common Chef: First Some Terminology Part 2](#)」の日本語参考訳です。

スレッドとは何かについて正しく理解していますか？

コアが何か分かったところで、別の混乱しがちな用語について見てみましょう。

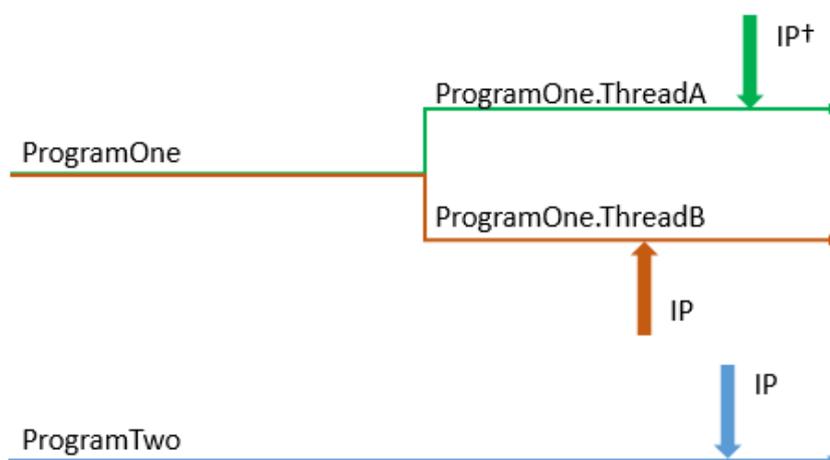
ここでは、やや専門的な内容を取り上げます。興味のない方は、この記事スキップしても、ほかの記事を理解するのに影響はないでしょう。すでにスレッドについてご存知の方は、この記事を読むことでスレッドについてより詳しく理解できるかもしれません。

ハードウェア・スレッドとソフトウェア・スレッドの相違点を説明する前に、「スレッド」とは何か考えてみましょう。コンピューター科学者が慣れ親しんでいる従来のスレッドと、コアでプログラムを実行するためのスレッドは別のものです。例えば、ハイパースレッディングは、1つのコアでスポンされたプログラムを2つのスレッド (例えば、MyProgram.ThreadA と MyProgram.ThreadB) で実行できることを指しているわけではありません。関連はありますが、厳密には違います (ハイパースレッディングについては、将来の記事で取り上げる予定です)。コンピューター・アーキテクトにとってスレッドとは、「実行スレッド」であって、プログラムではありません。ハードウェア・レベルにプログラムは存在しません。あるのはハードウェアが実行する一連の命令だけです。つまり、「実行スレッド」です。

(この記事では、「ハードウェア・スレッド」を省略して「HW スレッド」、「ソフトウェア・スレッド」を省略して「SW スレッド」としています。)

以下の SW スレッドの図を見てみましょう。

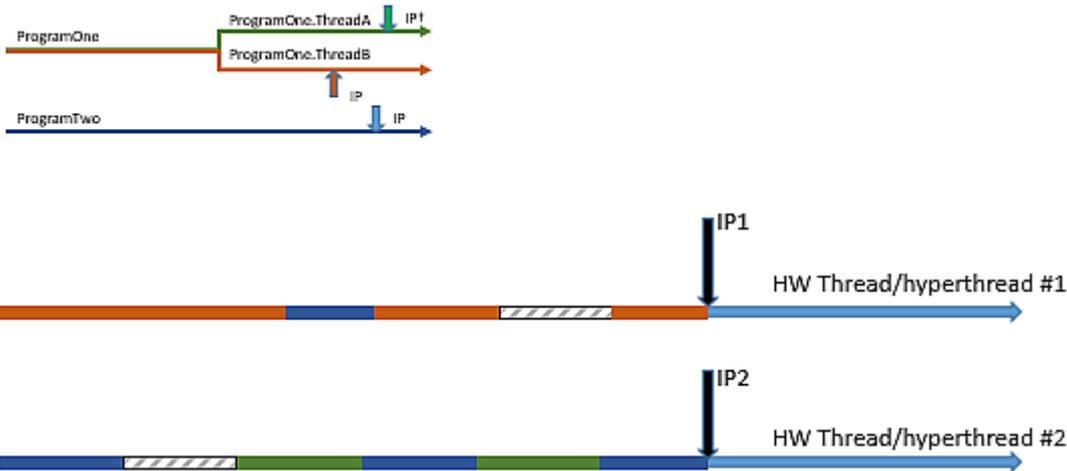
ProgramOne によってスポンされた2つのスレッド (ThreadA と ThreadB) と、2つ目のプログラム ProgramTwo に対応する1つのスレッドがあります。



SW スレッドの図: プログラマーから見たハイパースレッディングが有効なシングルコア・マシンで2つのプログラムを実行する場合 (IP は命令ポインターと呼ばれる、次に実行する命令をコンピューターに知らせるレジスターです。)

これはよくあるケースです。プログラムを実行中で、命令ポインター (IP) は次に実行する命令を指しています。この図をよく見ると、IP がおかしいことに気付くでしょう。ハードウェアでは同時に 2 つのスレッドを実行できますが、IP が 3 つあります。また、IP が同じ縦位置になく、バラバラです。これは、各 SW スレッドに実行中のコンテキストに関連付けられた個別の IP アドレスがあるためです。ここで "コンテキスト" とは、レジスタや SW スレッドの有効化/無効化に関連するその他のものを含む、オペレーティング・システム (OS) のコンテキストを指します。("無効化" とはスレッドをスリープ状態にすること、"有効化" とはスレッドをウェイクアップすることです。) つまり、IP アドレスは、実際のハードウェアではなく、各 SW スレッドのコンテキストに関連付けられます。そのため、プロセッサ上で実行中の SW スレッドの数と同じ数の IP が存在します。

次に、HWTHREAD1 の図を見てみましょう。ハードウェア・スレッドとソフトウェア・スレッドの違いを簡単に比較できるように、左上に SW スレッドの図の縮小版を追加しています。HWTHREAD1 の図は、ハードウェアが SW スレッドをどのように実行するかを示しています。SW スレッドのスライス、オペレーティング・システムによって便宜的に 2 つの HW スレッドに分配されます。このことから、SW スレッドと HW スレッドの違いが分かります。SW スレッドは、プログラマーが並列に実行することを意図したプログラムの一部です。そのため、SW スレッドは任意の数にすることができます。一方、プロセッシング・ユニットによる一連の命令の実行が HW スレッドです。コアごとの HW スレッドの数は各コアで固定されています。ただし、最近のプロセッサには、コアごとの HW スレッドの数を変更できる機能があるものもあります。(ハイパースレッディングは、1 コアで 1 スレッドではなく 2 スレッドを実行することができるモードです。詳しくは、将来の記事で取り上げる予定です。) 一連の命令は、カーネルを含む異なる SW スレッドのコンテキストからのものです。カーネルは、1 つの SW スレッドのコンテキストを格納し、別の SW スレッドのコンテキストをロードして、複数の SW スレッド間の切り替えを行います。



### HWTHREAD1 の図: ソフトウェア・スレッドのハードウェア・スレッドへの割り当て

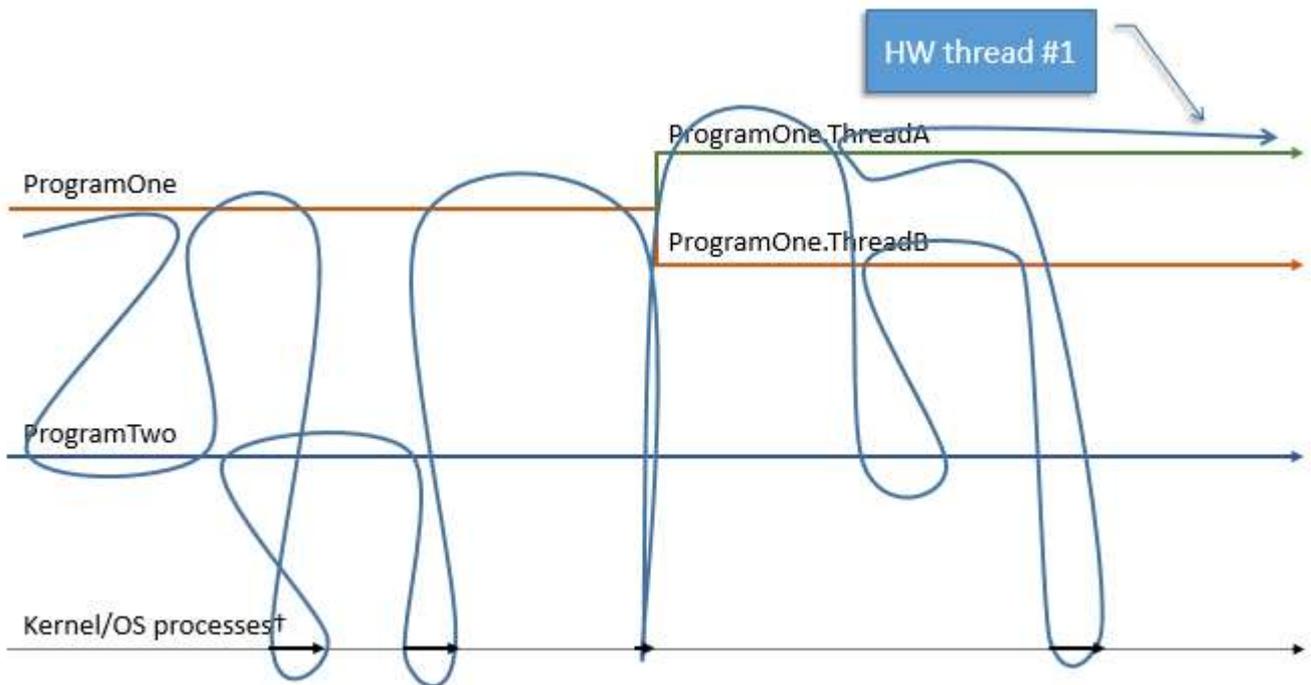
プログラマーやプログラムにとって、これらの情報は関係あるのでしょうか? インテル® Edison ボードベースの IoT (Internet of Things) や Android\* 向け料理アプリなどを通して、SW スレッドの概念を理解している場合のみ関係あります。SW スレッドと HW スレッドは、プログラマーを混乱させる可能性があります。

追記: ハードウェア・スレッドをソフトウェア・スレッドに割り当てる別の方法

SW スレッドの図は、プログラマーの観点から SW スレッドを表したものです。HWTHREAD1 の図は、プログラムの SW スレッドが、どのようにプロセッサの HW スレッドへ割り当てられるかを示しています。

HWTHREAD2 の図は、(SW スレッドの HW スレッドへ割り当てではなく) プロセッサの HW スレッドの 1 つが、どのようにソフトウェア・スレッドへ割り当てられるかを示しています。ただ単に言い回しが違うだけ

のようには見えますが、実際には大きな違いがあります。さまざまなプロセスにジャンプしている青色の波線が HW スレッドです。ここでは、詳しく説明しません。詳細がお知りになりたい場合は、コメントを残していただければ回答します。



## HWTHREAD2 の図: ソフトウェア・スレッド (およびオペレーティング・システム) 間をジャンプするハードウェア・スレッドの追跡

次の記事: 最近のプロセッサの仕組み

注:

\*オペレーティング・システム (OS) は、プログラム (例えば、スプレッドシートなど) の実行に必要なジョブを実行します。これらのジョブは、実際に製品を販売したり、製造しなくても、ビジネスの遂行には必要なマネージャーやスタッフと考えることができます。

同様に、カーネルはオペレーティング・システムの中核であり、処理に必要な重要な機能を実行します。例えば、マネージャーがいなくても問題ないかもしれませんが、製造装置の保守・点検を行うスタッフは (製品の製造に直接関わってなくても) 必要です。

コンパイラーの最適化に関する詳細は、[最適化に関する注意事項](#)を参照してください