

# 第5章

割り込み駆動で経過時間を計ったり、クロックの数を数えたり

## タイマ・コントローラと割り込みコントローラの使い方

C言語には基本的に「ある時間にある関数を実行する」というような「タイミング」の概念がありません。しかし割り込みがあれば、タイミングに依存するプログラムを簡単に書けるようになります。ここでは、割り込みが特殊な関数呼び出しによって実現されていることを示し、通常のmain()関数からの呼び出しとは関係なく実行されるプログラムが書けることを理解してもらいます。

また、割り込みのトリガとしてタイマを用いれば一定間隔でプログラムを呼び出せる、C言語の仕様にはない「時間」という概念を扱う方法を理解します。



関連データ

大中 邦彦

### 1. 「割り込み」とは何か

#### ●「仕事すごろく」の例で割り込みを考える。

本誌2009年4月号の第1章(pp.40-52)で、CPUがプログラムを実行するようすをすごろくにたとえて説明しました。図1を見てください。このすごろくを「必ず1しか出ないさいころ」を使って遊んでもらうと、「玄関のチャイムが鳴ったら相手を確かめ、宅配便だったら印鑑を押す」という仕事をしてもらえます。この「1しか出ないさいころで仕事をさせるすごろく」を「仕事すごろく」と命名しました。

さて、図1のすごろくを「毎朝の仕事をしながらか、さらに宅配便の受け取りもやってもらう」場合はどうしたらよいでしょう。ちょっと複雑ですが、図2のようなすごろくを考えてみました。少し仕事をしたらチャイムに気を配り、またちょっと仕事をして…、ということを繰り返します。もしチャイムが鳴ったら20番の升到飛びます。これは、いわゆる「サブルーチン呼び出し」の処理です。サブルーチンの処理が終了したら、このサブルーチン呼び出した次の升に戻ります。

面倒ですね。このように、何か別のことを気にしながら処理を進めるすごろくを作るのは非常に骨が折れます。

#### ●朝食を作っている途中で宅配便が来たらどうする？

図2の4番升で朝食を作っている途中でチャイムが鳴ったらどうなるでしょう。「ピンポン」と鳴っているのですが、朝食の準備がまだ途中なので玄関には向かえません。やっと朝食を作り終えたときにはすでにチャイムは鳴り終わっていて、宅配便を受け取り損ねます。そこで、すごろ

くを図3のように修正し、朝食を作っている途中でも宅配便を受け取れるようにしてみました。

つまり、今まで「朝食を作る」と書かれていた部分を「冷蔵庫から卵とベーコンを取る」、「フライパンに油をひく」といった細かな仕事に分解し、より細かい単位でチャイムが鳴っていないかどうか確認できるようにしたのです。これにより、宅配便が来たときに玄関に出損ねることがなくなります。このように、何かが起こっていないかを逐一調べることを「ポーリング (polling)」といいます。

図3のように「いつ起こるかかわからない仕事」をポーリングで適切に処理する場合、ポーリングをできるだけ定期的に起こるようにする必要があります。しかし、そのためには仕事を細かく分解しなければならず、図3の例のように朝食を作るという一つの仕事が分断され、わかりにくくなってしまいました。しかも朝食のメニューがベーコン・エッグに固定されてしまっています。別のメニューに変更したい場合は、その手順を細かく書いて、1升ごとにチャ

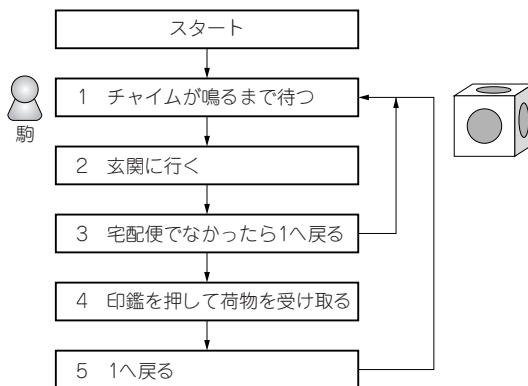


図1 玄関のチャイムが鳴ったら宅配便を受け取る「仕事すごろく」