

## 第8章 マイコン操作に最低限必要な 基本事項を理解しよう

# 整数演算のしくみと アドレッシングの基礎知識

山本 秀樹  
Hideki Yamamoto

この章では、今までの章では十分に取り上げられなかった、マイコンに関する基本的な事項について説明します。

整数の演算は第4章でも取り上げましたが、もう少し詳しく見ておきます。

### 符号なし整数とその演算方法

#### ● 符号なし整数とは

マイコンで扱う整数の型は、大きく分けて**符号なし整数**と**符号付き整数**からなります。C言語でも、unsignedとsignedの整数があるのと同じです。

符号なし整数は0以上の整数であり、単純な2進数で表現されます。nビットの符号なし整数は、0から $2^n-1$ までの整数を表現でき、マイコンでは8ビット、16ビット、32ビット程度が通常使われます。8ビットなら最大値は255、16ビットなら65535、32ビットなら4294967295です。

#### ● 符号なし整数の加算

符号なし整数の加算について説明の必要はないでしょう。問題になるのは、加算した結果が最大値を超えるかどうかです。最大値を超える場合は、それを表現するために、一般に**キャリ・フラグ**が使われます。付録マイコンでは、**フラグ・レジスタのCフラグ**に相当します。加算を行った結果Cフラグが‘1’であれば、加算結果が最大値を越えたことがわかります。

例えば、16ビット整数の加算を行った結果Cフラグが‘1’になった場合でも、16ビットの範囲は正しく計算されています。16ビット整数どうしの加算であれば、結果は17ビットで表現できるので、Cフラグを最上位の17ビット目と考えると正しい結果にな

ります。この特徴を使えば、マイコンの命令では扱えないような大きい整数の加算も可能になります。

#### ● ADC命令を使った32ビット以上の整数の加算

付録マイコンの加算命令では、16ビット整数までしか扱えません。しかし、大きい整数の加算を行うために、**ADC命令**が用意されています。**ADC命令は、通常の加算に加えてCフラグも加算**します。Cフラグの加算では、フラグの値が‘1’なら数値1、‘0’なら数値0を加算します。

ADC命令を使って32ビット整数どうしを加算する方法を、**図1**に示します。

この方法は、加算する数値をマイコンで扱える大きさに分割します。この図では、上下16ビットずつに分割します。まず下位16ビット側をADD命令で加算します。その結果、16ビットの値とCフラグが得られます。もし下位側の加算により桁上がりが起こっていると、Cフラグが‘1’になっています。

次に上位16ビット側の加算を行います。そのとき、数値A、Bのほかにも、Cフラグの値も加算します。Cフラグの値は下位の加算結果の17ビット目に相当するので、上位16ビットの最下位ビットに相当します。Cフラグの値が‘1’なら上位の加算結果に1を加えます。その結果、上位側16ビットの加算結果と、ふたたびCフラグが得られます。このCフラグの値は、32ビット整数どうしの加算における桁上りの有無を示します。

32ビット以上の大きさの加算でも、下位側から繰り返し加算していけば、同様に実現できます。

#### ● 符号なし整数の減算

同様に符号なし整数の減算を考えてみましょう。こ

### Keywords

符号なし整数、整数演算、キャリ・フラグ、フラグ・レジスタ、Cフラグ、ポロー・フラグ、ADC命令、SBB命令、2の補数、1の補数、ゼロ・フラグ、Zフラグ、サイン・フラグ、Sフラグ、オーバーフロー・フラグ、Oフラグ、条件判断、アドレッシング・モード、アドレス修飾

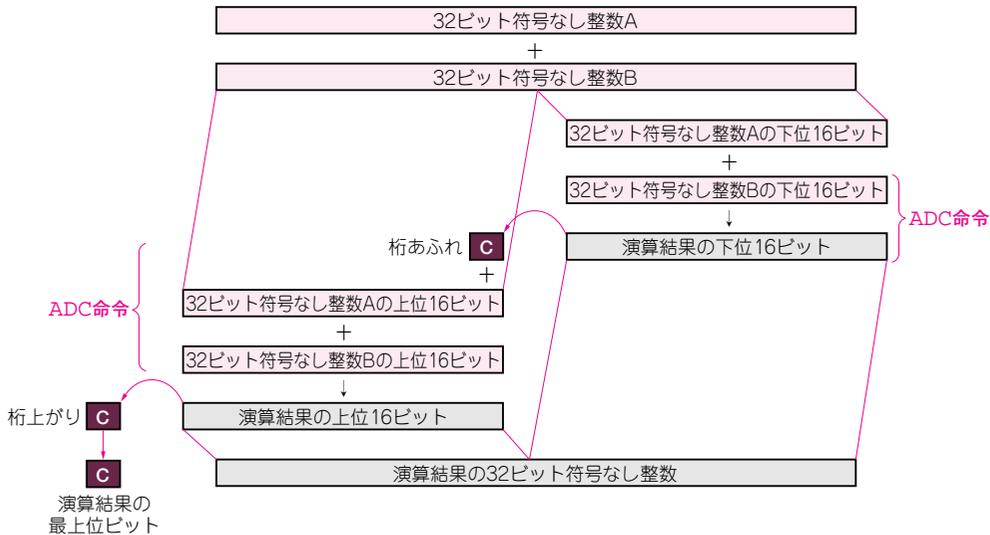


図1 ADC命令を使えば付録マイコンで32ビット整数どうしの加算が可能になる

の場合に問題になるのは、結果が0未満になるかどうかです。もし0未満になると、それを表現するために、一般に**ポロー・フラグ**が使われます。通常、キャリ・フラグとポロー・フラグは、(キャリとポローが同時に発生しないので)同じフラグが兼ねています。付録マイコンでも**フラグ・レジスタのCフラグ**がこれらを兼ねています。ただし付録マイコンでは、**ポロー・フラグはCフラグの反転であることに注意が必要**です。つまり、減算を行った結果が0未満であればCフラグが'0'になり、結果が0以上であればCフラグは'1'です。

減算も加算の場合と同様に、ポロー・フラグを使うことで、マイコンの命令で扱えない大きい整数の減算が可能です。付録マイコンの減算命令も、16ビット整数までしか扱えませんが、減算命令(SUB命令)と、ポロー・フラグも減算するSBB命令を使って、16ビットを越える整数の減算が可能です。その方法は加算の場合と同様なので、省略します。

### ● 符号なし整数の乗除算

付録マイコンの**乗算命令**は**サイズ指定によって、8ビット整数どうしを乗算して16ビットの結果を得るか、16ビット整数どうしを乗算して32ビットの結果を得るかのいずれか**です。いずれの場合も、結果を表現するのに十分なビット数があるので、フラグの変化はありません。

一方、**除算命令**は**サイズ指定によって、16ビット整数を8ビット整数で除算して8ビットの商と余りを得るか、32ビット整数を16ビット整数で除算して16ビットの商と余りを得るかのいずれか**です。商がそのビット数で収まりきらない場合や、除数が0の場合は

オーバーフローを表すOフラグが'1'になります。Cフラグは変化しません。

## 符号付き整数とその演算方法

### ● 符号付き整数は2の補数を使って表現する

符号付き整数は、0以上の整数に加えて、負の整数も表現することができます。符号付き整数を表現する方法として、一般に**2の補数**と呼ばれる表現が使われます。それ以外に絶対値と符号ビットによる表現もありますが、2の補数による表現のほうが加減算の容易さで優れています。2の補数による表現では、表現する値が0以上であれば、符号なし整数と表現は同じになります。

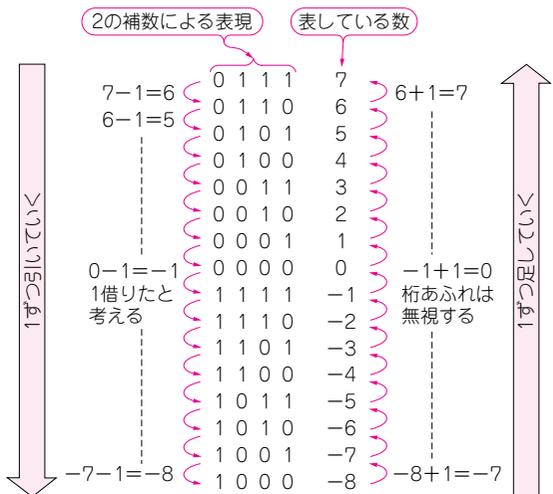


図2 2の補数による表現