

# 第 2 章

デジタル信号処理ICを上手に使いこなそう!

## DSPの基礎知識と開発環境

### 2-1 DSPとは何か?

#### ■ DSPと普通のプロセッサの処理の違い

DSPのハードウェアについて詳しい話をする前に、DSPと普通のプロセッサの処理の違いを調べてみましょう。DSPと普通のプロセッサの処理の違いは図2-1、図2-2のようになります。

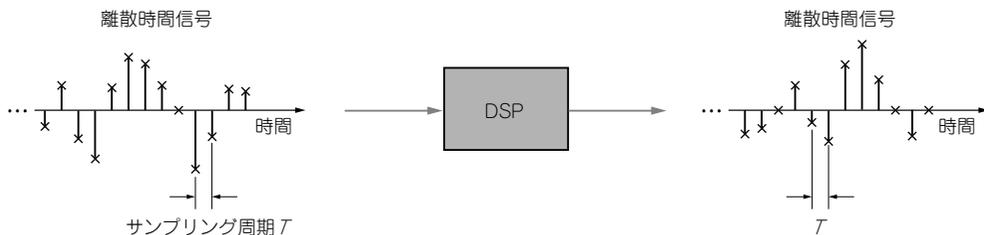
#### ● DSPは単純なパイプライン処理

DSPは、図2-1に示すように一定のサンプリング周期の離散時間信号を入力して、その処理結果を同じく離散時間信号として出力するのがもっとも基本的な処理です。入力および出力はともに途切れなく連続的に続く離散的データ列となります。

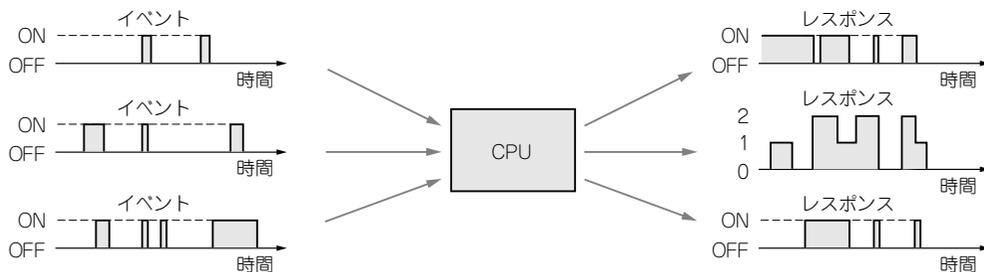
まさに、離散時間信号処理のための算術演算(加減算や乗算)が主体のプロセッサがDSPです。なぜDSPが特殊なハードウェア構成をもつのかというと、このようなデータ・ストリームの処理を一般的なプロセッサよりも高速に実現するためです。

#### ● 一般的なプロセッサはイベントドリブン処理

一方、図2-2に示すように一般的に普通のプロセッサの入力となるのは、外部からの複数の非同期



〈図2-1〉 DSPの処理



〈図2-2〉 普通のプロセッサの処理の一例

のイベント(event)となります。入力したイベントに応じた相異なる論理演算主体の処理を行うのがプロセッサの役割です。DSPの場合は、入力信号のサンプリング周期は一定でしたが、普通のプロセッサでは図2-2のように各イベントの発生するタイミングは一定していないことも大きな違いです。

## ■ DSPの定義とハードウェア構成

DSPとは離散時間信号処理を行うためのプロセッサであることを説明しましたが、そのための特別なハードウェア構成とはどのようなもののでしょうか？ もっとも単純なDSPの定義は、1クロックで連続的に積和演算ができるプロセッサです。

積和演算とは、次式に示す形の演算です。

$$w \leftarrow w + xy \quad \dots\dots\dots (2-1)$$

ただし、 $x$ および $y$ ：入力、 $w$ ：出力

乗算(積)と加算(和)を組み合わせた演算なので積和演算と呼ばれます。実は、信号処理の演算の多くはこの積和演算の形式を取っているのです。差分や累和演算、さらにはデジタル・フィルタや各種の相関演算も積和演算の一種として処理可能です。

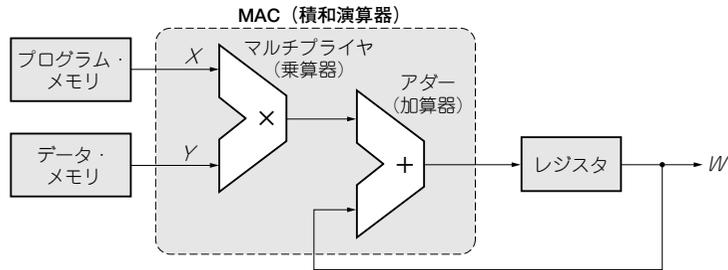
### ● FIRデジタル・フィルタの演算

一例を挙げると、FIRデジタル・フィルタの演算は次式のように定義されますが、これは式(2-1)の積和演算を $N$ 回繰り返していることがわかります。

$$w[n] = \sum_{i=0}^{i < N} x[n-i]y[i] \\ = x[n]y[0] + x[n-1]y[1] + x[n-2]y[2] + \dots + x[n-(N-1)]y[N-1] \quad \dots\dots\dots (2-2)$$

ただし、 $N$ ：フィルタ次数(タップ長)、 $x[n]$ ：入力信号、 $y[n]$ ：フィルタ係数、 $w[n]$ ：出力信号

積和演算を高速で処理できるということは、言い換えれば離散時間信号処理を高速処理できることです。



〈図2-3〉DSPのハードウェア構成(ハーバード・アーキテクチャと積和演算器)

### ● 積和演算を1クロックで処理するハードウェア構成

積和演算そのものは普通のプロセッサでも実行可能なのですが、DSPはこれを1クロックで高速処理するために、次に示すような特徴のあるハードウェア構成をしています。

- 1クロックで乗算が可能な乗算器(積和演算器)を内蔵
- ハーバード・アーキテクチャをもつ

DSPは、図2-3のように積和演算を高速で処理するために、1クロックで処理が可能な乗算器と加算器を内蔵しています。乗算器と加算器は一体化した構成になっているので、二つをまとめて積和演算器と呼びます。

積和演算器には、二つの入力  $X$  と  $Y$  がありますが、1クロックで積和演算を行うためには、二つの入力データの読み込みも1クロックで同時に行わなければなりません。そのために、DPSではプログラム・メモリとデータ・メモリおよびそれぞれのバスが分離、独立したハーバード・アーキテクチャと呼ばれる構成が用いられています。

### ● その他の機能と特徴

現在市販されているDSPは、そのほかにも信号処理に適した次のような機能や特徴をもっています。

- ゼロ・オーバーヘッド・ループ命令(連続的に高速に積和演算を完行するため)
- ビット・リバース・アドレッシング<sup>注21</sup>機能(FFT演算用)
- サークュラ・アドレッシング<sup>注22</sup>機能(FIRフィルタ処理などの用途向け)
- 高速I/O(高性能のDMA<sup>注23</sup>、マルチプロセッサ・システム用リンク・ポートなど)
- 大容量メモリ搭載

これらの特徴の有無にはかかわりなく、1クロックで積和演算ができるプロセッサはDSPと呼んでも差し支えないでしょう。

注2-1：ビット・リバース・アドレッシング▶FFT演算に使うデータの並べ替えに対応したアドレッシング・モード。

注2-2：サーキュラ・アドレッシング▶デジタル・フィルタ(FIR)などの処理に使うバッファを循環アクセスするアドレッシング・モード(リング・バッファの処理用)。

注2-3：DMA(Direct Memory Access)▶CPUが逐次レジスタを経由してデータ転送するのではなく、ハードウェアでデータを直接転送する。CPUの演算処理と平行してデータ転送を行える。

## 2-2 DSPの処理性能と信号処理に必要な演算量

一口にDSPといっても、市場にはさまざまな特徴をもった製品があふれています。そのようなDSPの中から用途に適したものを選ぶための予備知識として、DSPの処理能力の仕様の見方を説明しましょう。

### ■ 積和演算性能(MAC)がもっとも重要

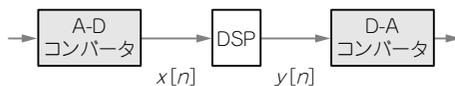
信号処理演算には、積和演算性能(MAC, Multiply ACcumulate)がもっとも重要です。基本的に、DSPの能力は1秒間の処理可能な積和演算回数(MAC/sec)で評価するのが基本です。DSPは、1クロックで1回の積和演算が可能ですから、基本的にクロック周波数は積和演算性能となります。

ただし、2クロックで1回の積和演算を行う特殊な構成のDSPや、外部から供給したクロックを内部のPLLで通倍して動作クロックを作り出しているものもあるので、性能比較をするときには注意する必要があります。

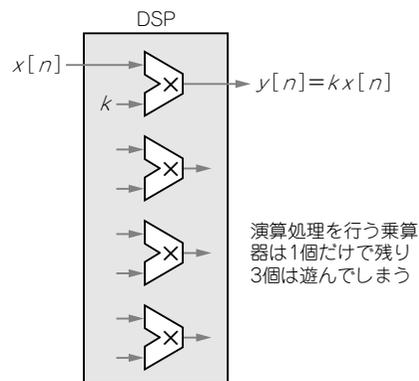
### ● 複数の演算器を内蔵するDSP

複数の積和演算器を内蔵しているDSPでは、演算器の数だけカタログ上の積和演算処理速度が大きくなります。しかしすべてのアプリケーションで、複数の積和演算器が効率良く並列動作するとは限りません。例えば、図2-4のように単純に、1サンプルずつA-D変換した信号をそのままDSPに入力し、DSP内部ではバッファリングなしで処理結果を速やかにD-A変換するようなシステムを考えてみましょう。このシステムの理論上の処理遅延は1クロックです。

単純な例を挙げると、入力信号 $x[n]$ を定数倍した $kx[n]$ が出力となるようなシステムです。このとき、図2-5に示すようにDSP内部に複数の乗算器をもっていたとしても、有効な演算処理を行えるのはその中の一つだけで、残りは遊んでしまうことになります。



〈図2-4〉最も基本的なDSPの利用形態



〈図2-5〉乗算器を複数(4個)搭載したDSPで図2-4を処理するようす( $x[n] = ky[n]$ )の演算の処理例)