

第3章

フィルタの定義から簡単な構成例までを理解しよう フィルタのいろいろな 特性とその分類

本章では、アナログでもデジタルでも共通な信号処理システムとしての“フィルタ”とはどういうものであるのかというイメージの組み立てから始めて、その性質を明らかにする。簡単な定義からスタートして、直感的な理解ができるように説明のステップを進めていく。

3.1 周波数選択性による分類

一般に、『入力信号の中から必要な周波数成分だけを通過させ、不要な周波数成分を除去して出力する』という働き、すなわち周波数選択性を有するシステムをフィルタ (filter) と言います (図 3.1)。たとえば、家庭におけるゴミの分別処理といったところでしょう。フィルタにはさまざまな定義のしかたがあります。そこでまず、周波数選択性に基づく分類の説明から始めましょう。

3.1.1 フィルタのもつ周波数選択性

本論に入る前に、フィルタのもつ周波数選択性の雰囲気をつかみましょう。まず、Scilab で記述された [プログラム例 3.1] をエディタで注意深く作成します。

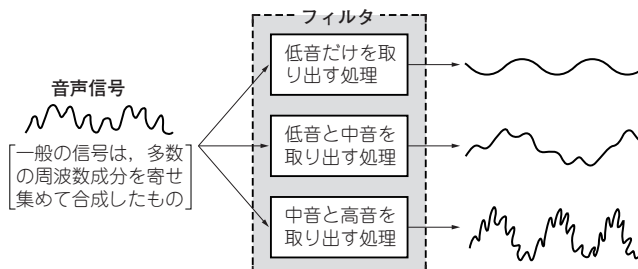


図 3.1 フィルタの働き

プログラム例 3.1 (周波数選択性)

```
//frequency characteristic ***** .....①  
  
global sfrq,tmax; .....②  
T = 0.002; .....③
```

```

sfrq = 1/T; .....④
tmax = 0.5/T; .....⑤
t = 0:T:(0.5-T); .....⑥

while(1) .....⑦
    printf('\n'); .....⑧
    frq = input('input frequency(0-50[Hz])'); .....⑨
    if frq == [] then break end; .....⑩

        amp = 1/sqrt(1+(frq/20)^2); .....⑪
        pha = arctan(1/(1+i*(frq/20))); .....⑫
        xin = cos(2*pi*frq*t); .....⑬
        yout = amp*cos(2*pi*frq*t+pha); .....⑭

        aoscg(xin,yout,0); .....⑮
        disp(max(abs(yout)), 'ymax = '); .....⑯
end .....⑰
// ***** .....⑱

```

[プログラム例 3.1 の説明]

- ② 複数の関数にまたがる変数を共通利用するために、グローバル変数 (sfrq, tmax) を定義する
- ③ グラフ表示の時間間隔 (変数 T) を 0.002 [秒] に設定する
- ④ 1 秒間の分割数 (変数 sfrq) を設定する
- ⑤ グラフ表示する横軸 (時間軸) の最大値 (変数 tmax) を設定する
- ⑥ サンプル時刻 (変数 t) を設定する
- ⑦～⑰ while(1) と end ではさまれるプログラムを繰り返し実行する
- ⑧ 改行する
- ⑨ 入力信号の周波数 (変数 frq) の入力待ちになる
- ⑩ 周波数を入力せず、キーボードから キーを押したときに、プログラムの実行が終了する。なお、ほかのプログラムを実行するには、かならず実行中のプログラムを終わらせてからにしなければならない
- ⑪ フィルタの利得 (変数 amp) を計算する。利得は、入力信号が何倍で出力されるかを表すパラメータであり、詳細は「3.2 利得特性による分類」で後述する
- ⑫ フィルタの位相 (変数 pha) を計算する。位相は、入力信号がどれくらいずれて出力されるかを表すパラメータであり、詳細は「3.3 位相特性による分類」で後述する
- ⑬ 入力信号 (変数 xin) として、周波数 (変数 frq) の cos 波形の信号値を設定する
- ⑭ 利得と位相を考慮して、出力信号 (変数 yout) を計算する
- ⑮ 入力信号および出力信号を 2 現象オシロスコープ表示する。
関数コマンド aoscg は、