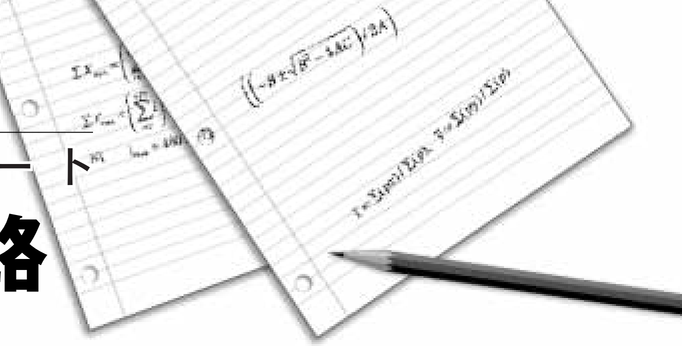


section 4

私のディジタル回路設計ノート

SIN・COS 発生回路

畔津明仁

これまでに取り上げた y/x，$\sqrt{}$，Arctan などと並び，正弦波(sin および cos)は代表的初等関数です．座標変換や変調・復調，画像や音声の処理など，たいへん多くの用途があります．ソフトウェアでもハードウェアでも，安直にこれをテーブルにする方法がありますが，じつは重大な落とし穴もあるのです．また逆に，正確な sin 関数にとらわれていると，簡単な解法を見失うこともあります．今回は，なめらかな周期関数の例として，sin および cos の発生回路を取り上げます．

1．正弦関数の用途

本連載でこれまで取り上げた Arctan(y/x)や$\sqrt{x}$などと並んで代表的な初等関数とされるのが$\sin(x)$，すなわち正弦関数です．

数学的には三角比を表すものとして平面三角法の解(辺長や面積など)を得るのになくてはならないものです．また定義域を拡張すれば“三角関数”となり，周期関数を代表するものとなります．

一方，工学的にはさらに重要な用途があります．その一例をあげてみましょう．

(1) 交流(sin 型)波形の代表として．

(2) 時間領域から周波数領域への変換のリファレンスとして．

すなわち，フーリエ変換やラプラス変換では，時間軸上の波形(関数)を多数の正弦波の和として表します．

(3) DDS(direct digital synthesizer)などの信号発生源として．

(4) 各種の変調や復調を行う場合の局所発振器(ローカル・オシレータ)として．

もちろんヘテロダイン周波数変換を含みます．

(5) 極座標系(θ，r)から直交座標系(x，y)への変換として．

すなわち，$x=r\cos(\theta)$，$y=r\sin(\theta)$となります．

(6) 直交座標と回転する際の係数として．

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix}$$

(7) パワー(ベクトル和，代表的には$\sqrt{x^2+y^2}$)が不変な演算に対する係数として(**図1**)．

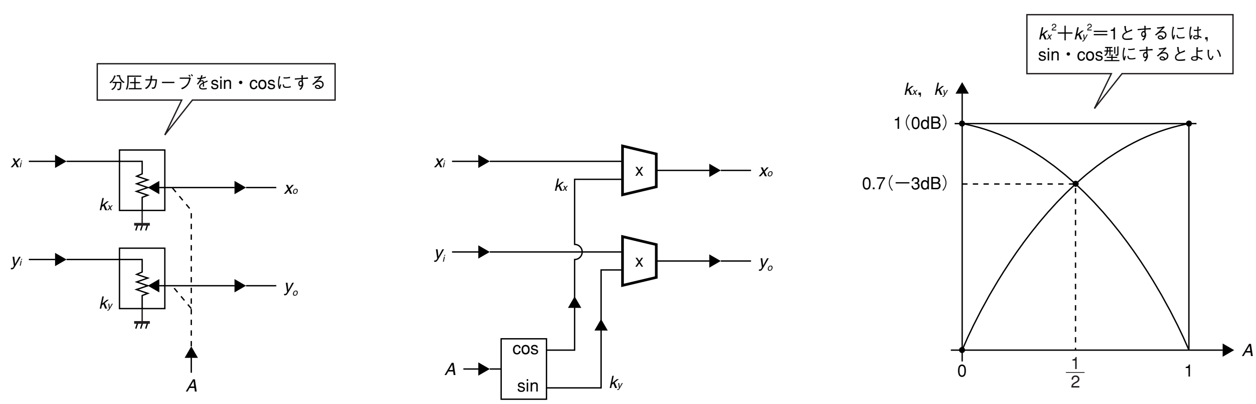
そのほかにも，sin や cos の登場場面は無数にあります．

ところが，これを実際のハードウェアとして実現する場合，それぞれの用途で少しずつ要求が異なることがあります．それは，“精度”であったり“見た目の形状”だったりしますが，連続周期波形として“歪み”や“なめらかさ”という項目があるのが特色でしょう．

今回は，関数の連続性と誤差との兼ね合いについても触れたいと思います．

2．波状周期関数

まずは，**図2**を見ていただきましょう．アナログ波形を見慣れた方なら，違いは一目瞭然です．しかし，波形を見慣れ



(a) アナログ・アッテネータ　(b) ディジタル・アッテネータ　(c) ゲイン・カーブ

〔図1〕 ($k_x^2+k_y^2$)を一定にしたい場合

ない方にとっては，(a)と(b)を同時に見比べない限り，両者の差は些細なものであって，どちらも十分に“正弦波”に見えるものです．

事実，筆者は，正弦波状の画像効果用(**図3**)のハードウェアを要求されたとき，当初の回路で(b)を使い，その後のバージョン変更時に(a)の波形に変更したことがありますが，画面を見てそれに気づいた人は，プロ・アマを問わずいませんでした．

これに似たような例(正弦波が歪んでいても気づかない)としては，リバーブやビブラートのような音響効果があります．また，本来は精度を要求されるはずの座標変換(極座標→直交座標)や変調波の復調においてすら，「ちょっと特性が悪いね」くらいで片づく場合があります．

図2の種明かしをしてしまうと，(b)は正弦波ですが，(a)は放物線，つまり次の波形です．

(a)… $f_1=4x(1-x)$　　(for $0 \leqq x < 1$)

(b)… $f_2=\sin(\pi x)$

この両者の誤差は，$x \fallingdotseq 0.15$ で最大約0.056となりますから，2進数の精度としては5ビットくらいしかありません．周波数軸で見ると，(a)には3次成分が−6.7dB，5次成分が−13.5dB(基本波に比べて)も含まれていますから，本当の(?)変調・復調には使えそうにありません．

しかしそれでも，視覚・聴覚に訴える用途では使えることが多く，欠点を承知の上でなら，精度が必要と思われる用途にすら使えることがあります．

放物線 $x(1-x)$ を求める回路は簡単です．正直に作ったとしても乗算器1個よりずっと小さい規模になります(なぜ？と思う人は修行がたりません．2進数の復習を勧めます)．したがって，回路設計方法として，ここで述べることはしません．

それよりも，ずっと重要な問題は次の2点です．

(1) “正弦波(状)”と要求された場合，それが本当に無歪みの正弦波でなくてはならないのか否か．

要求する側の言葉にとらわれずに，適切な方法を探し出してみると，思わぬ簡潔な解が見つかることが少なくありません．

(2) **図2**(a)くらいで役に立つのなら，もう少し誤差を減らした関数は，もっと使えるに違いない…という考え方は，裏切られることがある．

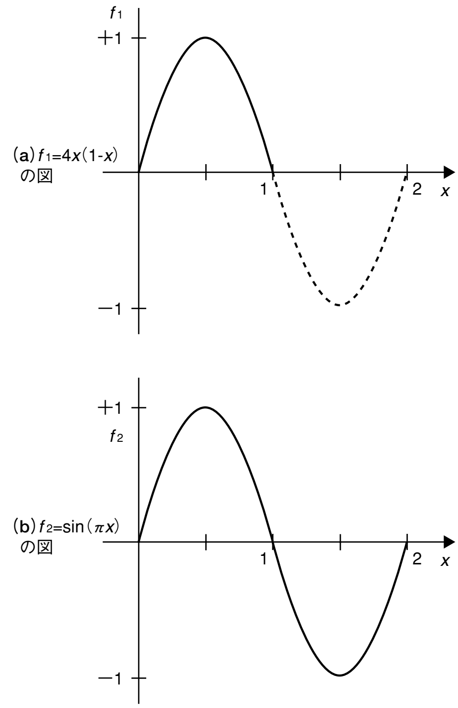
関数を作るとき，たんに誤差が小さいことが良いことではありません．

この(1)，(2)は，回路設計者にとって，非常に重大な問題を含んでいます．今回は(2)について少し考えてみます．

3．なめらかな関数の表現

◆テーブルを使うか否か

これまでの回で取り上げてきたさまざまな演算でもそうなのですが，回路設計の初心者は，「ROMかPLAかを使ってテーブルを作れば…」と簡単に考える傾向にあります．



x	f_1	f_2
0.1	0.36	0.31
0.2	0.64	0.58
0.3	0.84	0.81
0.4	0.96	0.95
0.5	1	1
0.6	0.96	0.95
0.7	0.84	0.81
0.8	0.64	0.58
0.9	0.36	0.31

〔図2〕正弦波状波形－ほとんど同じに見えるが…

たしかに，入出力の関係を“真理値表”とし，それをそのまま回路(ROMまたはPLA)化すれば，どのような関数でも原理的には実現できることになっています．とはいうものの，現実には欠点もあります．その一つは，(割り算の回でも触れたが)高精度の数値演算系ではビット数が増すため，ROMやPLAでの回路構成が現実的はなくなることです．

これに対して，もっとも経済的でかつシステム要求を満足すべく，回路設計者は日夜，設計に取り組んでいます．演算アルゴリズムの考案や回路化の方法はもちろん，誤差の量やその性質の解析が，システム要求との適合性を調べるためには重要となります．

図2(a)は，少し極端な例ですが，これがなぜ成立するかは大切なポイントを含んでいますから，本当の正弦波発生に触れる前に考えてみることにしましょう．

◆精度とビット数

前述のように，振幅が±1のときの最大誤差は0.056，つまり，2^{-5} くらいの誤差があります．ここで問題としたいのは，それだけの大きさの誤差が許される場合でも，ビット数を5ビットで打ち切らないことです．

図3の画像効果用途の場合，精度は5ビット分にすぎなくとも，波形の“なめらかさ”を保つために，データ長は12ビットとしてあります．