

デジタル制御システム

Digital Control System

平成 28 年

辻 峰男

まえがき

本テキストではデジタル制御とコンピュータ (マイコン) 制御システムの入門として、まず理解しておかなくてはならないこと及び実際の応用で重要と思われることを簡単な例を用いてできるだけ丁寧に述べている。

第 1 章では、最も簡単な場合を例に取り、デジタル制御の概要を説明する。デジタル制御システムではサンプリング周期ごとに制御対象の状態量を検出し、それを基に必要な入力量をマイコンで演算して制御対象に加える。デジタル制御システムを数学的に表現すると、数列あるいは差分方程式になる。微分方程式で記述される連続系の場合にはラプラス変換を使ってシステムのブロック線図を作るが、数列あるいは差分方程式の場合には z 変換を用いる。第 1 章では z 変換についても説明する。

第 2 章では、 z 変換により得られる伝達関数を用いてブロック線図を作る方法を説明する。一般に制御対象は連続系であるから、加える操作量は連続量でなくてはならない。このために、マイコンで演算される値をサンプリング周期の間一定に保つ零次ホールドの機能が必要になる。零次ホールドと制御対象の伝達関数(ラプラス変換表示)をまとめて z 変換することでブロック線図を得ることができる。従来のテキストでは、偏差を求めるところまで連続系で考えられているものが少なくないが、偏差はマイコンの中で演算されるのが一般的であろう。また、マイコンで演算した後のサンプラーも不要と思われる。

第 3 章は、デジタル制御器の話である。多くのデジタル制御法が提案されているが、本稿では実際に良く利用されている PI 制御や補償要素について述べる。PI 制御や補償要素は、連続系として微分方程式や伝達関数 (ラプラス変換) で表わされているものが判り易く、良く知られている。これらをデジタル制御に利用するには差分方程式に離散化 (デジタル再設計という) しなくてはならない。離散化すればプログラムして製品に組み込むことができる。離散化の方法は多く存在するが、良く利用される方法を説明する。さらに実際の応用では入力量は無制限に大きくできないので、リミッタが必要となる。本章ではリミッタについても詳しく述べている。

第 4 章では、ブロック線図で表されたデジタル制御系の安定判別法を説明する。 z の双一次変換を行うと連続系の安定解析でよく知られたラウスの方法が利用できて便利である。また、デジタル制御特有の制御法であるデッドビート制御についても例題を通して説明する。

第 5 章では周波数応答の計算法を述べる。連続系のラプラス変換で表した伝達関数の周波数応答は $s = j\omega$ と置くことで得られるが、 z 変換で表した伝達関数の周波数応答は $z = e^{j\omega T}$ と置くことで求まる。周波数応答を利用したナイキストの安定判別についても説明する。また、 z 変換による周波数応答とフーリエ変換の関係について述べる。

第 6 章では、 z 変換を使わないで差分方程式を用いて直接解析する方法を説明する。行列を用いた連立差分方程式(状態方程式)によってデジタル制御システムを記述すると時間応答の計算(解析プログラムの作成)が容易で、系行列の固有値から安定判別ができる。状態方程式は z 変換によるブロック線図に比べ複雑なシステムにも適用しやすく、パソコンによる解析に向いていると言えよう。

第 7 章は、実際に多く用いられているマイコンによるデジタル制御システムの話である。マイコンの歴史を築き現在でも一部に利用されている 8 ビット CPU Z80 を取り上げる。原理を理解するには、構造が簡単な方がわかりやすく、Z80 を使ってマイコンの仕組みを説明する。また割り込み周期がサンプリング周期になることが多いので、その理解はデジタル制御に欠かせない。最後に最も簡単な電流制御の場合を例にとりマイコン制御システムのハードウェアとソフトウェアの概要を述べる。

デジタル制御器の設計法としては、ラプラス変換により連続系として設計し離散化する方法(デジタル再設計)とデジタル制御系として直接設計する方法がある。デジタル再設計の場合、デジタル制御系としての安定解析、ステップ応答、周波数応答などの計算で性能をチェックしておく必要がある。サンプリング周期が十分短かければ、連続系の特性に近づくと考えられるが、サンプリング周期が短くとれない場合や制御ゲインが大きい場合には特に注意を要する。デジタル制御系として直接設計する場合は、伝達関数の周波数特性を考えることにより、ナイキスト線図やボード線図が描け、連続系に似た設計が可能となる。また、行列を用いた時間領域での設計法もある。実際の設計では、デジタル再設計の方が広く普及しているようである。これはラプラス変換した伝達関数がこれまでの経験もあって判りやすいこととコンピュータの高速処理によりサンプリング周期が短く選べるケースが多いことが要因であろう。

制御法に関しては連続系と同様、PID 制御や各種補償要素が良く用いられている。最適制御、状態観測器(オブザーバ)、適応制御、 H_{∞} 制御(ロバスト制御)、パラメータ同定なども高性能化を図るために使われているが、本テキストでは述べておらず、これらについては文献(5),(17)などを参照して欲しい。これらにも連続系としての設計法とデジタル制御系としての設計法が考えられるが、状態観測器の一種であるカルマンフィルタや最小 2 乗法によるパラメータ同定は離散時間系として取り扱われている。適用が難しい制御法にチャレンジして、性能の向上を図ることこそ技術者魂であろう。

また、本テキストでは A/D 変換器のビット数で決る量子化誤差(一種の雑音と考えられる)に関しては議論していない。A/D 変換器のビット数が小さいと制御ゲインが大きくなりできないことが考えられる。ただし、通常は、量子化誤差の影響が出ないようにビット数を選ぶので、問題となることは少ないように思われる。

目次

1. デジタル制御とは？	
1.1 数列	1
1.2 z 変換	2
1.3 z 変換による差分方程式の解法	5
1.4 かんたんなデジタル制御系	9
1.5 デジタル制御系の解析法まとめ	13
2. ブロック線図	
2.1 インパルス列を用いた z 変換	15
2.2 z 変換によるデジタル制御系のブロック線図	16
3. デジタル P I 制御と離散化	
3.1 デジタル PI 制御	27
3.2 フィルタおよび補償要素	30
3.3 FIR フィルタおよび IIR フィルタ	33
3.4 微分方程式で与えられた制御則の離散化	35
3.5 リミッタ	37
4. 安定解析	
4.1 安定条件	42
4.2 安定判別	46
4.3 制御器の演算時間を考慮した解析	51
5. 周波数応答	
5.1 周波数応答	55
5.2 ナイキストの安定判別	61
5.3 z 変換とフーリエ変換の関係	63
6. 時間領域での解析	
6.1 連立差分方程式	71
6.2 デジタル制御系の安定判別	75
6.3 P および PI デジタル制御系	80
6.4 一般的なデジタル制御系の解析	87
7. マイコン制御システム	
7.1 CPU	97
7.2 CPU とメモリ	98
7.3 アセンブリ言語	102
7.4 マイコンの構成	103
7.5 割り込みを利用したマイコン制御システム	108
参考文献・付録・索引	111