

標本化定理の概要を半日で理解に導くための パソコンによる学生実験の改良例

古宇田 フミ子, 小林 恭子, 田中 稲実, 加藤 一弘, 渡邊 廣次
東京大学 工学部 電気系 学科

パソコンを用いて、標本化定理の様子を観察する学生実験課題を、3時間の実験時間内で、予備知識が一律ではない工学部非電気系学科所属の3年生を対象として行なっている。この実験課題の改良例について、述べる。

実験の狙いを、対話的に進める、標本化定理で述べられている復元可能性を調べる、計算機を利用して信号の標本化と復元処理を行なう手法を学ぶと言う点に絞り、主にソフトウェアと実験テキストを手直しすることで、改良を進めた。

実験時に学生にアンケートをとり、理解度を調べた。この改良法では、事前に標本化定理を知らないグループでも7割が理解を示した。

An improvement of an experiment course which intends for the non-electrical students to understand overview of the sampling theorem by using a personal computer and by an AD converter equipment within half a day

Fumiko KOUDA, Kyoko KOBAYASHI, Inami TANAKA, Kazuhiro KATO and Koji WATANABE
Faculty of Engineering, University of Tokyo

An improvement for an experiment course is presented. We prepare a set of electrical-related experiments for the non-electrical, engineering students, which must be finished within 3 hours.

We have improved one of the above experiments, which examines the sampling theorem by using a personal computer and an AD converter equipment.

To do this, we have made a new software for observing the sampling theorem with the facility of interactive asking, displaying intermediate data, and printing flexible outputs, and have revised a text.

We took questionnaire to the students who attended the experiments and got the result that 70 percents of students who had not known the theorem before said "understood it".

1 はじめに

電気系の学科では、他学科の学部3年生に、電気系分野の学生実験を用意している。

この実験カリキュラムで特に考慮しなければならない点は、(1)実験時間が3時間に制限されていること、(2)受講する学生は工学部非電気系学科所属なので予備知識が一樣ではないことである。又、できれば、参加する学生が「実験をやって面白かった」と言う方向に持っていきたい。これらの制約の中で、2.1節で述べるようなテーマの実験を行なっている。

小論では、パソコンを用いて、標本化定理の様子を観察する実験課題の改良例について報告する。

このテーマでは、標本化定理をどこまで学習し、これを計算機を用いてどのように観察し、処理を行なうか、等の実験法が問題となる。

そこで、実験の狙いを以下の項目に定めて、実験内容の改良を試みた。

1. 実験装置上で、処理過程のデータ形式を観察できるようにする。例えば、アナログ信号を標本化パルスを用いて、標本化し、標本値をAD変換により量子化する過程で、標本値の列が量子化により、 $\sum a_i 2^i$ の形になっていることを確認すること等。
2. 標本化定理で示されている標本化パルスの周波数と入力信号周波数の関係を観察し、標本化定理で述べられている復元可能性を調べる。
3. 計算機を利用して信号の標本化と復元処理を行なう手法を学ぶ。I/Oインタフェースは二種類あること、それぞれのI/Oで、データ取り込み法が異なること、I/Oの処理時間と、パソコンでの計算処理時間の違いがあること、データ処理の仕方次第で結果の表示が変わること等をプログラミングを通して学ぶ。

次章では、この実験テーマの背景を述べ、3章では、実験改良の方法を述べる。又、4章では、学生へのアンケートを基に改良結果について述べる。

2 背景

2.1 実験カリキュラム構成

対象とする学生は、夏学期は化学系、金属系、資源(地球システム)、冬学期は、航空、機械系、船舶海洋の各学科の3年生の希望者である。実験項目は、エネルギー制御分野と情報電子分野に大きく二分される。前者の課題は、交流回路、変圧器、直流分巻電動機および発電機、インバータによる三相誘導電動機の駆動、直流サーボ系設計の基礎、高電圧実験、三相同期発電機、サイリスタ、

が用意されている。後者には、電気測定、電気回路、半導体素子と回路、アナログIC回路、発振回路、デジタルIC回路、フィルタ、マイクロコンピュータ、AD変換器とDA変換器、の課題がある。実験時間は、1回につき、3時間であり、この時間内に一つの課題を終らせる構成になっている。一つの実験グループは、原則として4人であり、3人又は、5人のこともある。

2.2 これまでの実験方法

パソコンを用いて標本化定理の様子を観察する実験はこれまでも「マイクロコンピュータ」の課題として行なわれていた[1]。

実験装置は、既製の標本化パルス発生装置とアナログ信号発生装置、自家製のAD変換装置、パーソナルコンピュータ(PC98)、プロッタ、プリンタ等から構成される。この中で、AD変換装置の構成は図1のようである。アナログの入力値はAD変換により、+5Vの値が、00Hに、0Vが7FHに、-5VがFFHに対応するデジタル値に変換される。

ソフトウェアとしては、BASICで書かれた処理が、3種類用意されている。何れもデータ数を問い合わせ、数の入力があると、その個数だけデジタル値に変換されたデータを読み込み、これを十進数に補正して、各々(後述)の処理を行ない、決められた大きさでプロッタに出力する。これらの違いは、読み込んだデータの処理の仕方であり、それぞれ階段関数として出力する、折れ線で出力する、以下の近似式を用いて計算して出力する

$$D_x = \sum_{i=1}^n D_i \cdot \frac{\sin \pi(x-i)}{\pi(x-i)}$$

処理をする点である。

処理の流れは、標本化パルス発生装置とアナログ信号発生装置から、それぞれ、パルスとしての矩形波とアナログ信号としての正弦波をAD変換装置に取り込み、正弦波をパルスの間隔でAD変換し、同時に、量子化した値をパソコンに取り込んで上記のBASICプログラムでプロッタ向きに加工し、出力することで終わるものである。

実験では、学生は、各種装置の電源を入れ、BASICプログラムの一つを読み出し、アナログ信号の周波数を変えながら、RUNを実行させる。すると、”自動的に”プロッタに処理波形を出すので、これを観察する、と言うものであった。

3 実験方法の改訂

3.1 実験方法改良の方針

1章で述べた制約を満ちつつ、実験の狙いを簡単に実現でき、「実験をやって面白かった」と感じることが

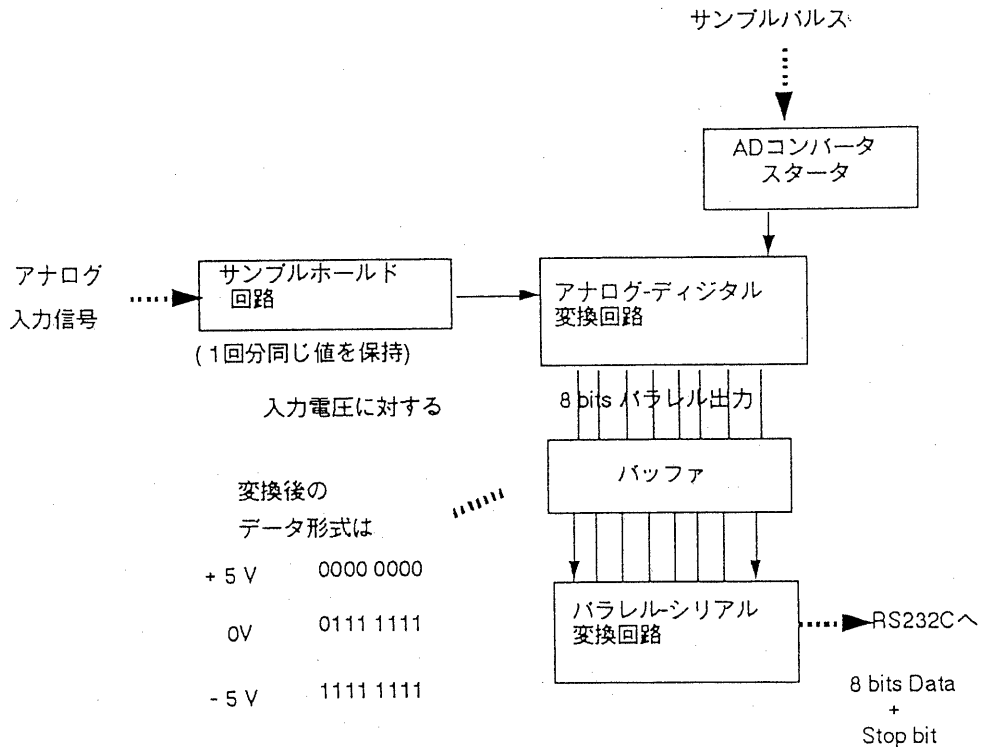


図 1: 標本化と量子化の回路図

できるような一つの方向としては、これまでに用いた実験方法のうち、ハードウェアはそのまま用いて、主に、ソフトウェアを手直しすることで最も容易に実現できると思われる。

実験時間が短いことから、つい、釦を押せばすぐ結果がでるような実験になってしまいがちとなるが、これでは、ブラックボックス化してしまい、学生には、原理等が理解しにくくなると思われる。そこで、実験過程で学生の参加を求めるような構成にしたい。この実験では、観察を行なうためのプログラムをより interactive にしたり、単なる測定だけでなく、プログラミングを採り入れることも考える。

そこで、実験に用いる標本化定理観察、測定用のプログラム(以下、処理プログラムと呼ぶ)は、

1. 対話的に処理を進める、
2. 標本化定理を見せる、

3. I/O 処理には二種類の違った方法があることを示す、
4. データ処理の仕方を変えることができるようにする、
5. 処理途中の状況が観察できるようにする、

等の機能が実現できるように作成することとした。

一方、これらの理解を容易にさせるために、学生の実験項目に対する事前の理解が一律ではないことを考えて、実験テキストの構成は、(1) 3時間で、初歩から基本原理が理解できるように、基本原理を簡単に説明し、(2) この原理のうち、実験ではどの部分を調べるか、(3) その方法、等を書くことにする。その理由は、実験中は疑問を感じても図書館等に調べに行く余裕がないことや、実験をやる時に基本原理を知らなかった場合、その場で原理について知ることができれば、興味も湧くと思われるし、実験時間を無駄に過ごすこともなくなると思われるからである。

更に、実験項目は、最低限理解すべきものと発展的な

ものに分けることにする。

3.2 実験方法改良のための準備

3.2.1 実験環境

3.1節の方針に沿って、実験環境を改良することから始める。BASICを用いる環境は止めて、OSをMS-DOSに変え、Turbo-Cも入れて、プログラミングにはC言語を用いることとした。ハードウェア構成は、パソコンの記憶領域をハードディスクや拡張メモリにより拡張し、AD変換装置はこれまで用いていたものをそのまま引き継いだ。

3.2.2 処理プログラム

用意した処理プログラムは、図2のように、利用者の引数設定、利用者にプロッタ描画の精度を聞く、等、対話的に処理が進むように構成されている。

処理途中のデータは、この処理図にはでていないが、indata, stddata(図3参照)、プロッタに描画する時の各座標、等はこれらを計算処理する時に、同時に、printf()により、その計算値を画面に表示している。

利用者の選択により(利用者にプロッタの描き方の条件を聞く)、量子化され、十進数に変換された値をそのままプロッタに出力する場合と、この値を標準化関数を用いて、復元処理して出力する、両方の場合を可能とした。

利用者引数には、図3に示すように、usrarg[0]からusrarg[5]までの6種類ある。このことにより、データ数、信号周波数、等を自由に選べるようにした。

処理プログラムの流れは、図2にあるように、AD変換用のI/Oポートの初期化を行ない、利用者に、データ数、アナログ周波数、サンプル周波数、等を問い合わせる。続いて、量子化されたデータを引数で指定された個数だけ読み込み、十進数に変換し、これらのデータを画面表示する。その後、再び、利用者にプロットの仕方(量子化した値をそのまま出力するか、標準化関数を使って、復元した波を出力するか)、を問い合わせ、その選択に従って、処理を行なう。

3.2.3 実験の進め方

実験では、電源を入れると幾つかの質問に答えることで、ほぼ自動的に実験環境に入り、ここに置かれているC(図2参照)で書かれた処理プログラムを一度コンパイルすれば、対話的に使用可能な方法を用意した。

3.2.4 実験課題

実験課題には、最低限理解すべきものとして、標準化定理の観察と確認、発展課題として、与えられたプロ

ラム(skeleton.c)を基にして簡単なプログラミングを行なうことの二種類を用意した。

基本課題では、上記のskeleton.cを動かし、サンプル周波数やアナログ周波数を変えて、出力をとり、両者間の周波数比と、復元の関係を観察する。

プログラミングでは、(1)skeleton.cの制御構造を変えることで、同じ量子化データを用いて、異なるプロット間隔でプロッタに出力することができるようにすること、(2)処理途中のデータを画面表示から、紙出力に変える(プログラム中、printf()の部分をfprintf()に変えれば良い)こと、(3)復元計算で計算しながらプロッタに出力している処理を、先に、まとめて計算し、後からまとめて、プロッタに出力する処理に変えること、の三点を課題とした。

3.2.5 実験テキスト

実験テキストには、基本原理の説明として(1)標準化定理の概要、(2)二種類のI/Oインタフェースについてのプログラム側の処理の説明、(3)処理プログラムの概要、等をつけた。

(1)については、キーワードの簡単な説明、定理概要を述べた。実験では、正弦波の場合について観察を行なうので、正弦波を例として、標準化定理の簡単な解説を付け、処理プログラムで用いる標準化関数を用いた(畳み込み積分)復元の式を導いた。具体的には、以下の式である。入力信号が、

$$f(t) = A \cos(\omega_0 t + \alpha) = \frac{A}{2} e^{i(\omega_0 t + \alpha)} + \frac{A}{2} e^{-i(\omega_0 t + \alpha)}$$

の場合、標準化されると、

$$p(t) = f(t) \sum_{n=-\infty}^{\infty} \delta(t - nT) = \sum_{n=-\infty}^{\infty} f(nT) \delta(t - nT) \\ = \begin{cases} f(nT) & \text{if } t = nT \\ 0 & \text{otherwise} \end{cases}$$

となるから、復元の式は、 $T = 2\pi/\omega_0$ ($\omega_0 = 2\pi f_0$)、 $S_a(x) = \frac{\sin x}{x}$ (標準化関数)、*は畳み込み積分として、

$$f(t) = p(t) * S_a\left(\frac{\omega_0}{2}t\right) \\ = \sum_n f(nT) \delta(t - nT) * S_a\left(\frac{\omega_0}{2}t\right) \\ = \sum_n f(nT) S_a\left(\frac{\omega_0}{2}(t - nT)\right) \\ = \sum_n f(nT) S_a\left(\frac{\omega_0}{2}t - n\pi\right) \\ = \sum_n f(nT) \frac{\sin(\pi f_0 t - n\pi)}{\pi f_0 t - n\pi}$$

```

main( )
{
    int usrgarg[I5];
    unsigned char  indata [Ia];
    float stddata[Ia], cn[Ia];

    /* *** Program starts *** */
    set_cn( cn, usrgptr ); /* setting n*Pi table */
    init_rs232c(); /* RS232C 通信ポートの設定 */
    get_usrgdata( usrgptr ); /* 利用者の引き数設定 */
    get_qu_data( usrgarg[0], &indata[0] );
    /* 量子化されたデータを RS232C から取り込む。*/
    convert_to( usrgarg[0], &indata[0], &stddata[0] );
    /* 量子化データを十進数の本来の値に変換する。*/
    printf(“ 描き方?”); getchar(); /* 利用者にプロッタの描き方の条件を聞く */
    if ( 標本化関数で計算する ){
        printf(“精度?”); getchar(); /* 利用者にプロッタ描画の精度を聞く */
        plot_axis ( usrgarg ); /* X-Yプロッタに座標軸を引く。 */
        sinewave_plot( usrgarg, &stddata[0], cn );
        /* 標本化関数で計算しプロッタに描画 */
    }
    if ( そのまま ){
        plot_axis ( usrgarg ); /* X-Yプロッタに座標軸を引く。 */
        dot_plot( usrgarg ,stddata); /* 標本点をプロッタに描画 */
    }
    if ( その他 ){
        printf(“プロットしません。 \n”);
    }
    exit(0);
}

```

----- skeleton.c の処理 -----

図 2: 実験のために用意したプログラム

となる。

ここで、 $f(t)$ をプロットする場合、 t の間隔を決める必要がある。これは、usrgarg[5] として、利用者から指定できるようにした。このことをデータ処理の仕方を変える実験課題に利用した。

(2) については、量子化データの取り込み方と、fprintf() で表される出力法の説明をつけた。これは、発展課題に関係させた。

プログラミング能力には個人差が大きいと思われるので、書くためのヒントをテキストに入れた。

実験テキストには、実験項目として、プログラミング、観察、測定、の順で挙げた。ところが、このことで、実験の手順が、この順番であると勘違いされるような問題が生じた(4.2節参照)。

4 改良法と評価

4.1 改訂実験に参加した学生

93 年度冬学期には、航空 10 人、産業機械、機械情報等の機械系 48 人、船舶海洋 51 人が受講した。このうち、33 人が、デジタル処理とパーソナルコンピュータの課題を選択した。内訳は、船舶海洋 9 人、機械 4 人、機械情報 8 人、産業機械 5 人、航空宇宙 7 人であった。この学生に、標本化定理を知っているかどうか、C 言語を知っているか、等について、この課題を受講した学生に、以下のようなアンケートをとった。

- 1) 標本化定理を知っていましたか。
 - a) 完全に理解していた。

```

int usrarg[I15]; /* 利用者からの引数やプロットに係わる引数を格納する表 */
/* usrarg[0] = データ数,
 *      1 = サンプル周波数,
 *      2 = アナログ信号の周波数,
 *      3 = プロット上で 5Hz の信号を描く場合の 1 周期の長さ
 *      4 = プロットに描く波形の区別 -> そのまま 復元
 *      5 = 復元時の精度因数
 *      6 = empty
 *      7 = empty
 *      8 = x0      プロットに描く時 x座標の初期値
 *      9 = delta x プロットに描く時 x座標の変化値
 *     10 = y0      プロットに描く時 y座標の初期値
 *     11 = Empty
 *     12 = empty
 *     13 = empty
 *     14 = empty
 *     15 = empty
*/
unsigned char  indata [Ia]; /* RS232C から取り込んだデータを格納する表 */
float stddata[Ia], /* 通常の値に変換した入力データを格納する表, Volt */
      cn[Ia]; /* table of n*Pi list */
int *usrptr = usrarg; /* 先頭アドレスの指定 */

```

図 3: 用意した利用者引数

- b) ある程度内容は、知っている。
 - c) 定理の名前だけは、どこかで聞いたことがある。
 - d) 全く知らない。
- a) 完成した。
 - b) 一部の項目はできた。これを用いて測定できた。
 - c) 部分的にプログラムを書いたが、完成しなかった。
 - d) 全く、お手上げで、プログラムは書けなかった。
- 2) C 言語を知っていますか。
- a) 良く知っている。
 - b) ある程度知っている。
 - c) どこかで聞いたことがある。
 - d) 知らない。
- 3) C 言語以外で知っている言語を挙げて下さい (幾つでも可)。
- 4) 時間配分についてお聞きします。
プログラミングと測定のそれぞれにどのくらい時間を要したか書いて下さい。

```

skeleton
データ加工の違い
データの記録
処理の分離

```

- 5) プログラミングは完成しましたか。

- 6) 実験を通して、標準化定理を理解できましたか。
 - a) 良く理解できた。
 - b) まあまあ分かった。
 - c) 全然分からなかった。
- 8) 最後に、事前にテキストを読みましたか。
 - a) 詳しく読んだ。
 - b) ザアッと目を通した。
 - c) ちらりと見た。
 - d) 全く見なかった。
- 9) その他、御意見があればお書き下さい。

その結果は、図 4 と、図 5 に示されている。図 4 で、「標準化」は、アンケート 1) の質問を表し、「C」はアンケート 2) を表す。「実験後の理解」はアンケート 6) を表す。「プログラミング」はアンケート 5) を表す。また、

標本化	C	人数	実験後の理解			プログラミング				
			a	b	c	a	b	c	d	
a	a	2								
a	b		2				1	1		
a	c									
a	d									
b	a	3	1	2			2	1		
b	b	6	1	5			5	1		
b	c									
b	d									
c	a	7								
c	b		6	1			4	1	1	
c	c		3	2	1			2	1	
c	d									
d	a	4								
d	b		3	1			3	1		
d	c		8	5	3			2	1	4
d	d									

図4: 実験に関するアンケート結果、事前の知識と理解度

aからdは、それぞれの項目を表す。数字は対応する人数を表す。

図5では「時間配分」はアンケート4)の、測定とプログラミングにかかった時間の総和を示す。縦軸の単位は1時間を表す。「プログラム」はアンケート5)を表し、a,...,dは完成度を表す。「理解したか」はアンケート6)を表し「事前に読んだか」はアンケート8)を表す。「理解」と「事前」の中の数字は、それぞれ対応する人数を示す。横軸の単位は一実験日(一グループ)を表す。左から右に日が経つ。

図4より、標本化定理を完全に理解していたのは2人、ある程度知っていたのは、9人、定理の名前だけ知っていたのは、10人、全く知らなかったのは、12人であった。

C言語については、良く知っていたのは、3人、ある程度知っていたのは、19人、どこかで聞いたことのある人は、11人、全く知らない人はいなかった。C以外の言語で知っている言語は、FORTRAN 26人、BASIC 22人、Pascal 26人、Smalltalk 2人、機械語、アセンブリ言語4人、C++2人、Lisp 11人、LOGO 1人、Prolog 1人、であった。

アンケートにはとらななかったが、実験中パソコンの使い方を質問に来た学生はいなかった。説明無しに直ちに実験環境に入っていった。

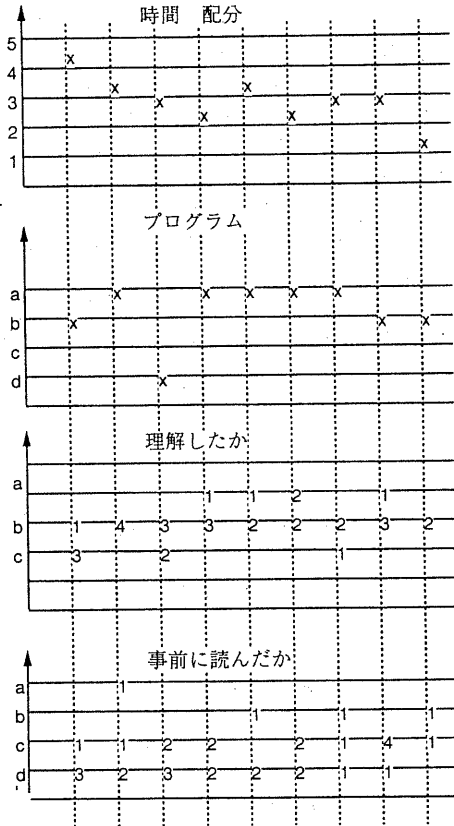


図5: 実験に関するアンケートの結果

4.2 実験手順の問題

当初のテキストでは、実験手順がプログラミングをしてから、まとめて、測定するように読めたので、最初の二回の実験では、学生は、プログラミングを行なってから測定した。この結果は、図5から分かるように実験時間が3時間を越えてしまった。そこで、実験テキスト中の実験手順を、先ず、標本化定理の観察を行ない、その後、余った時間でプログラミングを行なうように変えた。すると、殆どどのグループは、時間内にプログラムまで、できるようになった。観察だけで3時間かかっているグループが、一組だけある。これは、学生の子備知識の差から、生じたと思われる。

尚、最後のグループは(彼らの)試験直前のため実験を途中で止めたので、実験時間が他に比べて短い。

4.3 学生の子備知識と理解度

図4より、標本化定理を知らないグループ(c)とd)は、実験後の理解はまあまあ、分かったに達した人と、

全く理解できなかった人に分かれた。それぞれ、16人と6人で、標本化定理を全く知らなかった場合は、2:1の比率、定理の名前だけは、どこかで聞いたことがある場合は、4:1の比率であった。知らなかったグループでも7割がある程度の理解をしたと言える。

標本化定理をある程度以上は理解していたグループ(a)とb))は、実験後の理解も良く分かった、が4人、まあまあ理解した、が7人で、当然であるが、理解できなかった人はいなかった。

図5によると、大部分の人は事前に実験テキストを読んでいなかったように見える。それにも係わらず、実験をまあまあ理解した、と言う答が多かったことから、実験時間内に理解できるような実験だったと言えるであろう。

5 おわりに

標本化定理の概要を予備知識無しでも短時間で理解できるように、実験装置のうちソフトウェアを手直しし、この処理ソフトウェアを対話的に作り、実験テキストには原理が分かり易いように解説を加えた。

実験時に学生にアンケートをとり、予備知識や実験の理解度について尋ねた。この結果は、事前に全く標本化定理を知らない場合でも、7割が理解を示していた。問題点は理解に達しなかった人の存在であるが、この点は、課題として残る。

短時間の実験で理解に導くためには、実験手順も重要であることが分かった。先ず始めに、実験内容が容易に理解できるように、すぐ使える処理プログラムを用いて、様子を観察して実験内容に馴染みを持たせ、発展として、自らのプログラミングを行なうようにすることで、短時間で実験がこなせることも分かった。

参考文献

- [1] 一般電気工学実験 指導書 1991.9 (学生実験テキスト)