情報工学実験 2
A/D 変換

学籍番号 045718D : 翁長絵美

実地日 : 平成 17 年 11 月 07 日
実地日 : 平成 17 年 11 月 14 日

共同実験者
上原 裕亮 : 045711G
杉山 千秋 : 045727C
又吉 美奈 : 045751F

1　実験目的

A/D 変換の仕組みを学ぶと共に、市販の A/D ボードの使用法を修得することを目的とする。

2　使用した器具

- ノートパソコン
- AD12-8(PM)

3　報告事項

3.1　(1) 各実験について結果を報告しなさい。

サンプルプログラムが手元のノートパソコン上で動作することを確認せよ。
※若干プログラムを書き換えないと正しく動かない。

以下のように訂正を加え、コンパイル・実行した。結果、動作することを確認することができ、TA に確認させることができた。

–サンプルプログラム–

```
#include<stdio.h>
#include<conio.h>
#include<io32.h>

union _tag{         /* 訂正    ”_ ”の前にスペース */
long l;
char c[4];
};

#define ADR 0x280
#define CHLS 8

void main(void)
{
int ModeData = 5;
union _tag ClockData = {799};
union _tag CountData = {9};
```

```
int AiData,i,j,Count=0;
float AiVolt;

outp(ADR+6,0);
outp(ADR+6,1);
outp(ADR+7,ModeData);
outp(ADR+6,2);
outp(ADR+7,ClockData.c[0]);
outp(ADR+7,ClockData.c[1]);
outp(ADR+7,ClockData.c[2]);
outp(ADR+6,3);
outp(ADR+7,ClockData.c[0]);
outp(ADR+7,ClockData.c[1]);

printf("        ");
for(i = 0;i < CHLS; i++)
  printf("       ch%d" ,i);
printf("\n");

outp(ADR+2,CHLS-1);
  do {
      if(inp(ADR+2) &2){
      printf("  %5d: ",++Count);

      for(i=0;i<CHLS;i++){
      AiData = inpw(ADR);
      AiVolt = (float)AiData * 20 / 4096 -10;
      printf("%+7.3fv ", AiVolt);
      }
      print("\n");
      }
   }              /* 訂正   do の命令を終わりを付け加える  */
}while( inp( ADR+2 ) & 3 );
}                 /* 訂正   main() の終わりを付け加える */
```

(2) 以下のプログラムを作成せよ。「何かキーを押すまで、全チャンネルで 0.1 秒おきに A/D 変換を行い、1 秒おきに表示」

このプログラムは、サンプルプログラムのサンプリングクロックと表示する時間の間隔を変えればよい。したがって、ClockDate を

$$ClockData = \frac{SamplingClock}{100} - 1$$

の式から求める。SamplingClock は 100nsec 単位で設定するので、

$$\frac{100000000}{100} - 1 = 999999$$

よって、ClockDate は 999999 になる。

また、このままでは表示する時間は 0.1 秒になってしまうので、0.1 秒を 10 回繰り返したら表示 (つまり 1 秒ごとに表示) というようにプログラムを変える。具体的には、if 文を使って kbhit() が押されなかったら $j++$ をしていき、j=10 になったら $printf()$ で表示するというようにした。

その他に I/O アドレスはそれぞれ異なるため、ノートパソコンで調べ、適切な値に変更した。
以下に作成したプログラムを示す。

–プログラム–

```
#include<stdio.h>
#include<conio.h>
#include<io32.h>

union _tag{
long l;
char c[4];
};

#define ADR 0x260
#define CHLS 8

void main(void)
```

```
{
int ModeData = 5;
union _tag ClockData = {999999};

/* サンプリングクロックを 1 秒表示にするため 999999 に変更 */


union _tag CountData = {9};
int AiData,i,j,Count=0;
float AiVolt;
outp(ADR+6,0);
outp(ADR+6,1);
outp(ADR+7,ModeData);
outp(ADR+6,2);
outp(ADR+7,ClockData.c[0]);
outp(ADR+7,ClockData.c[1]);
outp(ADR+7,ClockData.c[2]);
outp(ADR+6,3);
outp(ADR+7,ClockData.c[0]);
outp(ADR+7,ClockData.c[1]);

printf("         ");
for(i = 0;i < CHLS; i++)
printf("       ch%d" ,i);
printf("\n");

outp(ADR+2,CHLS-1);

j=0;
do {

kbhit();
if(kbhit()) {outp( ADR+6,4); }

/* kbhit() でキーが押されたか判定 */

else if(inp(ADR+2) &2){
j++;
```

```
/* 以下から while 文の前まで 1 秒おきに表示されるされるように設定 */

if(j%10 ==0) printf("  %5d: ",++Count);

for(i=0;i<CHLS;i++){
AiData = inpw(ADR);
AiVolt = (float)AiData * 20 / 4096 -10;
if(j%10 == 0)printf("%+7.3fv ", AiVolt);
}
if(j%10==0) printf("\n");
}
}while( inp( ADR+2 ) & 3 );
}
```

コンパイルをして実行すると図 1 のように表示され、プログラムが正しいことが確認できた。

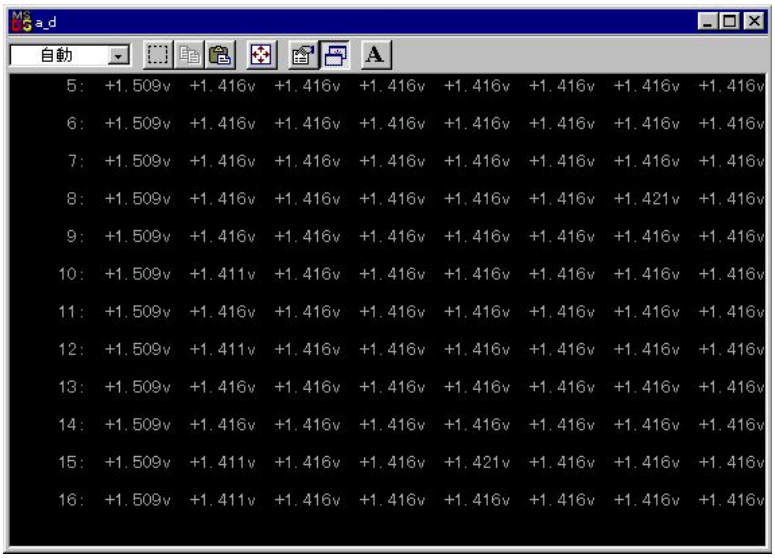

図 1: 実行結果

3.2 A/D 変換探は逐次比較型以外にもいくつか存在する。代表的な物について調べ、それぞれの利点、欠点を述べよ。

- フラッシュ型

  フラッシュ型は、逐次比較型で 1bit ずつ操作し変換していたことを 1 回で変換しようと考えられた型である。これは例えば、n ビットの A/D 変換をする場合 2^n 通りの基準電圧と 2^n 個のコンパレータを用意し、変換したい電圧を全ての基準電圧と一度に比べるという方式である。そのため、フラッシュ型は高速の A/D 変換に最適である。しかし、2^n ということから 1 ビット増える度回路が大きくなってしまうことが欠点である。

- サブレンジング型

  サブレンジング型は、フラッシュ型を 2 つ使いそれぞれに上位ビットと下位ビットの A/D 変換を分担させ、2 ステージで A/D 変換する方式である。フラッシュ型よりもコンパレータ数が少なく、低コストである。

- デルタシグマ型

  デルタシグマ型変調器により入力電圧を 1 ビットで量子化し、時間情報をもつビット・ストリーム信号に変換しフィルタで処理してディジタル値を出力する方式である。デルタシグマ型は、極めて分解能が高いにもかかわらず、比較的低価格で構成できる。しかし、高速度な A/D 変換はできない。

4 考察

本実験は A/D 変換器について学習した。実際のいろんな機器類には A/D 変換器でデジタル信号に変えて処理した後、デジタル信号を私達がわかるように出力しないとならないため、アナログ信号にまた変換するという作業をしなければならない。それが D/A 変換である。ディジタルデータは一定間隔のデータ値しか持っていないが、アナログデータは連続した値である。D/A 変換はこの切れた値をつなぐ働きをする。

5 感想

今回の実験は普段使い慣れていないノートパソコンを使ったので操作の仕方が歯痒かった。また、コンパイラがないパソコンなどがあったりといろいろ大変だった気がする。しかし、今回の実験で自分が慣れたパソコン以外にもいろんなものも触ってもっと経験を積もうっ！と思いました。

参考文献

[1] A/D 変換について
http://www.crt.or.jp/ kokochi/HENadc1j.htm#jurai
http://www2.117.ne.jp/ vision/paf/index.htm
http://www.yokogawa.co.jp/tm/TI/keimame/ad/ad_5.htm