

# L. 半導体ガスセンサによる匂い測定

## 1 目的

本実験は、酸化錫 ( $S_nO_2$ ) 系半導体ガスセンサの動作原理とその特性について理解を深め、計測制御システムを構築し、複数種類の匂いサンプルを実際に測定することにより、半導体ガスセンサの実践的な利用法を修得することを目的とする。

## 2 解説

人間はその生活を営んでいく上でいわゆる五感（視覚・聴覚・触覚・味覚・嗅覚）を利用している。これらの感覚の中で視覚・聴覚・触覚は物理量を対象としている一方、味覚・嗅覚は化学量を対象としている。量子力学により理論的な裏付けを得て半導体工学が進歩するのに伴い、物理量を対象にしたセンシングデバイスは急速な進歩を遂げた。この結果、物理量を対象にした工学体系も急速に充実した。一方、化学量を対象としたセンシングデバイスは開発途上にあり、味覚・嗅覚を対象とした工学体系は、未だに確立されていない。このため、現在でも、様々な独創的アイディアによる試行錯誤が行われている。

生命体において化学量を対象とする感覚は極めて重要な役割を果たしている。とりわけ、食物・環境等における危険を検知する嗅覚の役割は大きい。匂い識別の自動化および匂いの定性的・定量的評価は、防災・環境計測・食品管理など我々の日常生活と直接・間接的に関わる分野において、その実現が望まれている。ガス分析には、一般にガスクロマトグラフィー法が用いられるが、この方法では数時間もの測定時間を要してしまう。その上、結果として得られる莫大な数の分子種のピークに対して匂い識別を行うとなると、莫大な量の分析および計算を行う必要が生じてしまい、実用的な匂い測定系を開発することが難しくなる。そのため、匂い識別を短時間で簡便に行うことのできるシステムの実現が望まれている。

本実験で構築する匂いセンシングシステムは、センサ部に4種類の  $S_nO_2$  系半導体ガスセンサを用いて匂い応答パターンを測定し、得られた応答パターンから匂い識別処理を行う、パターン認識型匂いセンシングシステムである。最初にパターン認識型匂いセンシングシステムを考案・試作したのは、池上 et.al. である。彼等は、厚膜製作技術を用いて6種類の半導体ガスセンサをアルミナ基板上に集積化し、その応答パターンにより有機ガスの種類を識別した。半導体ガスセンサは、1968年にフィガロ技研により実用化されて以来、家庭用ガス警報器などに利用され、現在最も普及しているガスセンサである。半導体ガスセンサは感度が高い反面、ガス選択性が悪く、さらにセンサの信頼性において重要なファクタである再現性、環境依存性等に関して解決すべき点がある。しかし、ガス選択性の悪さを逆にうまく利用すれば、これまで難しいとされてきた複合臭の識別が実現される可能性がある。

そこで本実験では、半導体ガスセンサを用いた匂い測定装置を使用し、複数種類の匂いサンプルについて応答パターンを得て、匂い識別が可能かどうかを検討する。半導体ガスセンサとしては、アルコールガスセンサ、アンモニアガスセンサ、硫化水素ガスセンサ、メタンガスセンサの4種類のセンサを使用する。匂いサンプルとしては、清酒、焼酎、ポカリスエット、コーヒー、お茶の5種類を使用する。

## 2.1 匂いの種類とその受容機構に関する種々の仮説

匂いの実体は「揮発性物質が分子の熱運動によって気化したもの」であり、すなわち化学物質である。匂いを識別することは、化学物質を識別することを意味している。つまり分子レベルでの識別が必要となる。分子の違い＝匂いの違いであると考えた場合、我々の周囲には、膨大な種類の匂いが存在していることになる。

これまでに行われてきている匂いの種類に関する研究により、その数は数十万種ともいわれる。匂いに基本臭があると仮定して、樟脳臭・じゃ香臭・花臭・ハッカ臭・エーテル臭・刺激臭・腐敗臭の7種類が基本臭であるとする報告もあれば47種以上もの基本臭があるとする報告もある。嗅覚は、個人差が大きいという事実から、客観的にある種の基本臭の設定を行うことは現在でも困難である。表1に、生体における匂い受容機構の諸説を示す。

表 1: 生体における匂い受容機構の諸説

諸説名	概要
振動説	匂いは、光や音と同じように振動として伝わり、嗅粘膜を刺激するという説
化学説	匂い分子が嗅細胞の受容膜との間で化学反応を起こして刺激すると匂いの違いが生じるという説
酵素説	受容器の表面にある酵素系に匂い分子が与える影響によって匂いの違いが生じるという説
立体構造説	マンクリーフとアモーレによる。匂い分子と受容器は鍵と鍵穴の関係にあるという「鍵と鍵穴」説で匂いの違いは匂い分子の外形とそのディメンジョンによって決まるという説

以上の諸説は実験的根拠に乏しく不完全なものであることから、これらの説により匂い受容機構を完全に説明することはできない。現在では電気生理学的研究から、受容器細胞内で生じる脱分極の波が神経インパルスとなることで匂いの伝達が行われると考えられている。

## 2.2 半導体ガスセンサの動作原理

センサ部の素子が加熱され、定常状態になった後、素子表面へガスが接して吸着すると、まず吸着分子は表面を自由に拡散し（物理吸着）、その運動エネルギーを失う。この間に一部の分子は蒸発し、残りの分子は熱解離し吸着席へ局在（化学吸着）する。このとき、素子の仕事関数が吸着分子の電子親和力より小さければ、吸着分子は素子から電子を奪って陰イオン吸着する。この陰イオン吸着の傾向を持つ（酸化形ガスまたは電子受容性ガスと呼ばれる）ものには、 $NO_x$  などがある。しかし、素子の仕事関数が吸着分子のイオン化エネルギーよりも大きければ、吸着分子は素子へ電子を与え陽イオン吸着する。この陽イオン吸着の傾向を持つ（還元形ガスあるいは電子供与性ガスと呼ばれる）ものには、 $H_2$ 、 $CO$ 、炭化水素、アルコール類等がある。この酸化形ガスがn形半導体へ、還元形ガスがp形半導体へ吸着するとキャリアを減少させて電気抵抗を増加させる。一方、還元形ガスがn形、酸化形ガスがp形半導体へ吸着するとキャリアを増加させて電気抵抗を減少させる。n形半導体へガスが接した場合に起こる素子抵抗変化のグラフを図1に示す。

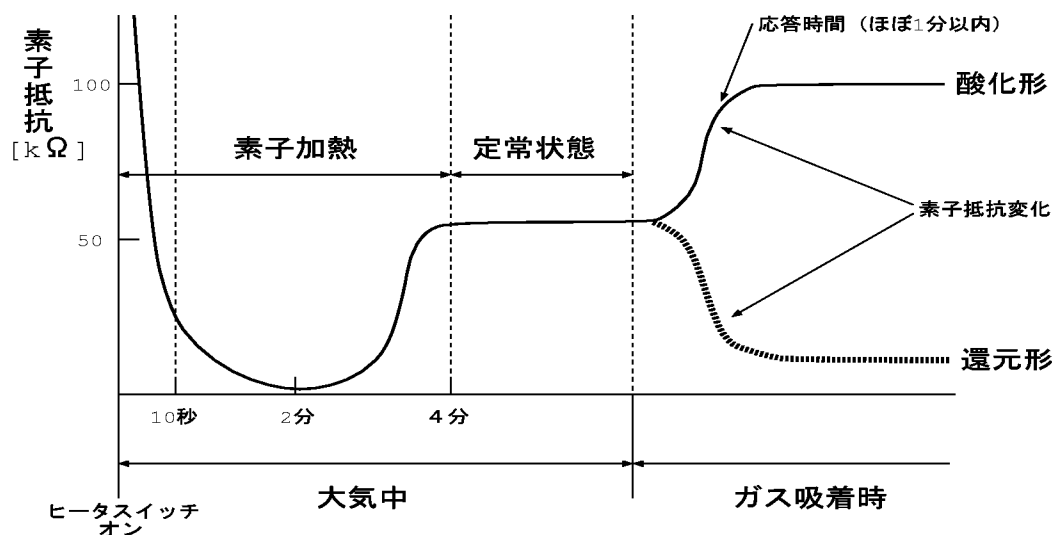


図 1: n形半導体ガスセンサにガスが吸着したときの素子抵抗の変化

## 2.3 $S_nO_2$ 系半導体ガスセンサの基本特性

図2に示すように、いずれのガスについても検出感度が最大となる温度が存在し、それはガスによって、また、添加剤によって大きく異なってくる。このことは、動作温度を適切に組み合わせれば、ガス選択性の賦与がある程度可能であることを示している。そこで本実験では、半導体ガスセンサ内部のヒータのオンオフ制御を任意の時間間隔で行い、半導体表面の温度を変化させることで、従来の測定方法よりも、微妙な匂いの違いを識別することが可能であるかどうかを調べる。

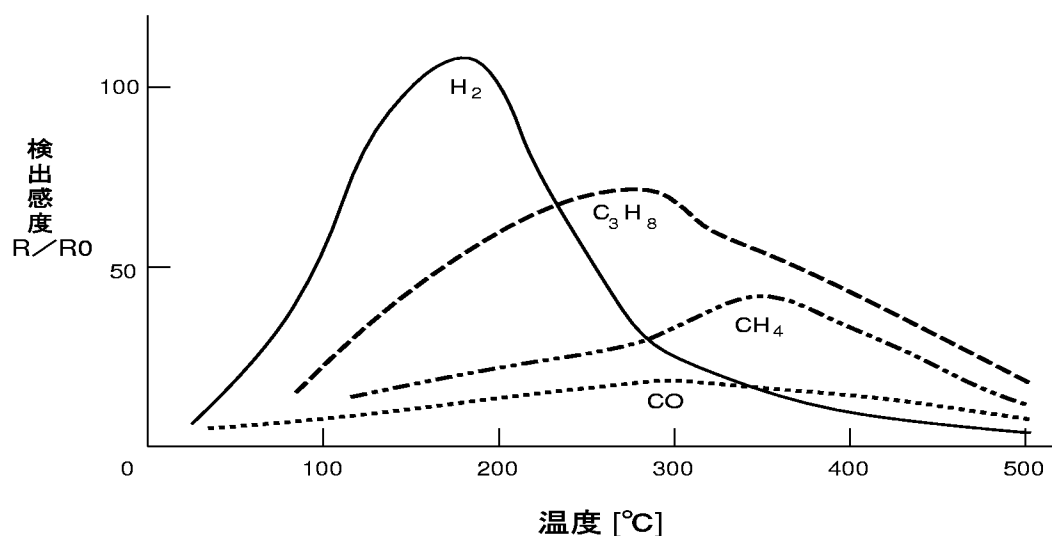


図 2: 半導体ガスセンサの動作温度とガス選択性の関係

### 3 匂い測定法

#### 3.1 計測制御システムの基礎

計測制御システムの基本構成を図 3 に示す。計測制御システムは一般に、物理量をセンサにより電気的信号に変換しコンピュータなどの信号処理装置に入力する「計測部」と、計測結果に応じて対象に操作を加える「制御部」により構成される。

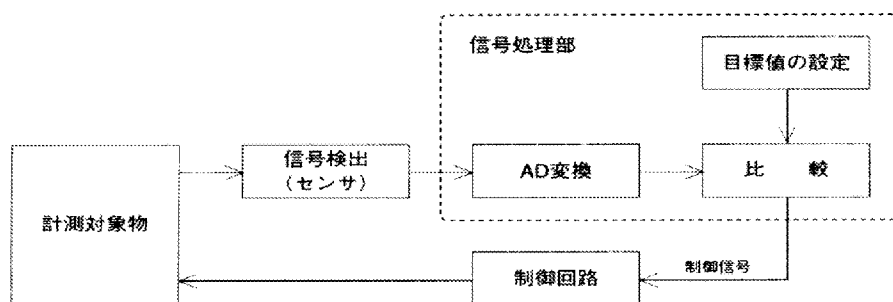


図 3: 計測制御システムの基本構成

本実験での計測対象物はガス（匂いサンプル）であり、信号検出には半導体ガスセンサ、入力インターフェースには A-D 変換ボード（CT-421）を利用する。A-D 変換後の信号処理と表示はポケットコンピュータ（PC-G850 (S)）により行う。なお、匂い測定装置は電力制御ボード（CT-421-02）を介してポケットコンピュータにより自動制御する。

### 3.2 測定装置の仕様

本実験では、匂い検出素子として4種類の半導体ガスセンサを使用する。使用する半導体ガスセンサの名称と特徴を表2に示す。

表 2: 使用するセンサ名とその特徴

センサ名	特徴
アルコールガスセンサ	アルコールなどの有機溶剤ガスに対する感度が優れる
アンモニアガスセンサ	アンモニアガスを30 ppmから検出可能
硫化水素ガスセンサ	硫化水素ガスを5 ppmから検出可能
メタンガスセンサ	メタンガス・イソブタンガスに対する感度が優れる

半導体ガスセンサを使用する際には、以下に示す点について配慮する必要がある。

(1) 半導体ガスセンサはガス濃度に対して非常に高感度である。

(2) 半導体ガスセンサは雰囲気中の気温・湿度の影響を受ける。

さらに、匂い測定装置を使用する際には、以下に示す点に注意する必要がある。

(1) 被測定ガスを発生させるまでは、半導体ガスセンサをある程度一定の気温と湿度の空気中に入れておく。

(2) マイクロシリンジを用いて一定の量の被測定サンプルをヒータによって気化させるようにする。

(3) 揮発したばかりの高温の状態の被測定ガスを半導体ガスセンサに接触させないようにする。

(4) 被測定ガスは、実験装置の中で一様に分布するように、循環用のファンを用いて十分に拡散させる。

本実験で使用する匂い測定装置の見取り図を図4に示す。匂い測定装置は主に測定チャンバ、サンプル気化チャンバ、標準空気生成チャンバの3つのチャンバから構成される。換気兼拡散用ファンと4つのモータにより制御されるシャッターにより、測定装置内の空気の流れは以下の3通りのモードで循環する仕組みになっている。

**モード1** 測定装置内の換気（残留ガスを排出）

**モード2** 標準空気の生成（気温・湿度を調整）

**モード3** 被測定ガスの循環・拡散

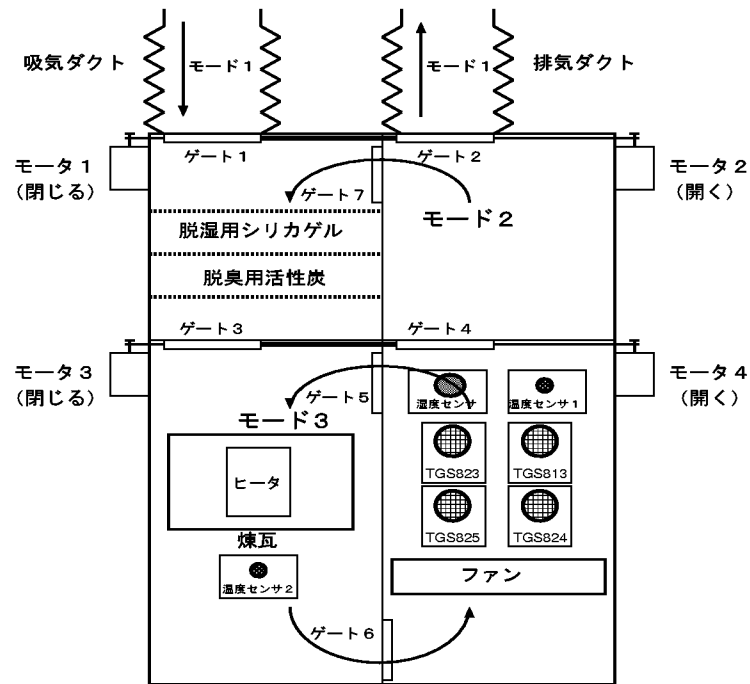


図 4: 匂い測定装置

### 3.3 匂いセンシングシステムの構成方法

ポケットコンピュータと A-D 変換ボードおよび電力制御ボードの接続図を図 5 に示す。図 5 に示すシステムに前節で述べた匂い測定装置を接続することにより匂いセンシングシ

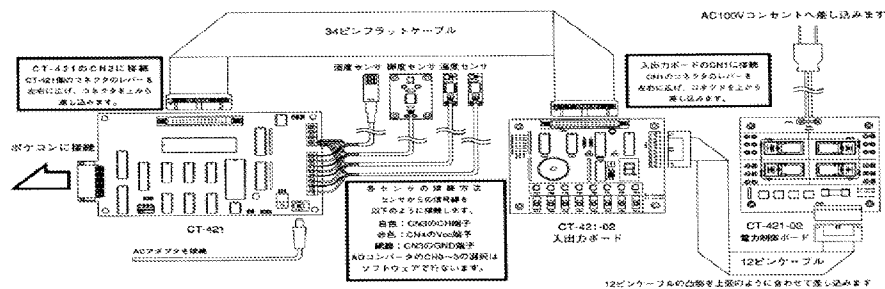


図 5: 各ボードの接続図

ステムを構成する。具体的には、各ガスセンサの負荷抵抗の端子電圧をオペアンプを介して CT-421 の A-D 変換端子に接続する。さらに、ガスセンサ内部ヒータ・匂いサンプル気化用ヒータ・換気兼拡散用ファンの電力を制御するため、対応する端子を電力制御ボードのリレーに接続する。

## 4 匂い測定実験

### 4.1 実験手順

前節に示した匂い測定装置を用いて、表 3 に示す 5 種類のサンプルについて、各ガスセンサからの応答パターンを測定する。実験手順を図 6 に、測定手順を表 4 に示す。

表 3: 本実験で使用する 5 種類の匂いサンプル

サンプル名	分類
A：清酒	アルコール飲料
B：焼酎	アルコール飲料
C：ポカリスエット	清涼飲料水
D：コーヒー	カフェイン含有飲料
E：日本茶	カフェイン含有飲料

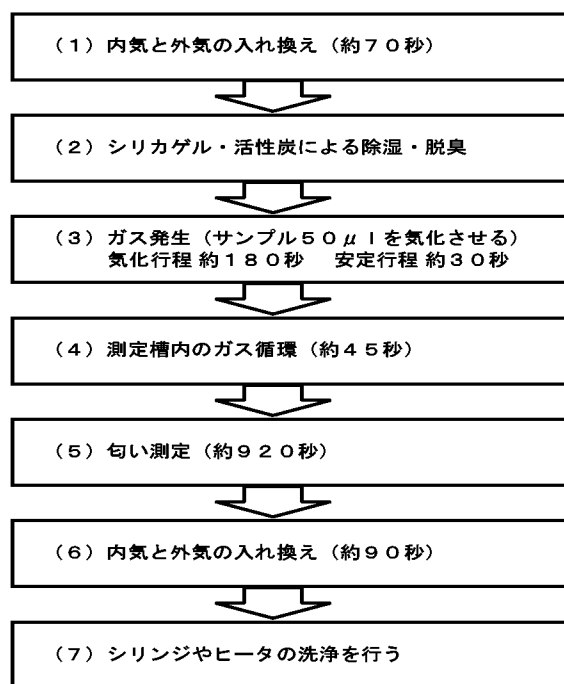


図 6: 実験手順

半導体ガスセンサの抵抗変化は負荷抵抗の端子電圧の変化として現れるので、負荷抵抗の端子電圧を逐次記録する。図 6 に示した実験手順をサンプル A から E について適用する。端子電圧の測定については、次節で述べる計測制御プログラムにより自動化する。

表 4: 測定手順（ヒータオンオフ間隔）

手順	オンオフ間隔（秒）	オンオフ回数
1	9 0	1
2	3 0	1
3	1	2 0
4	2	1 0
5	4	8
6	6	8
7	9	6
8	1 2	6
9	1 5	4
1 0	1 8	4
1 1	2 1	4
1 2	2 4	4
1 3	2 7	4
1 4	3 0	4

なお，半導体ガスセンサ（TGS823,824,825,813）の素子抵抗は次式により求めることが出来る．

$$R_s = R_L \left( \frac{V_c}{V_{RL}} - 1 \right) \quad (1)$$

ここで， $R_s$  は素子抵抗， $R_L$  は負荷抵抗， $V_c$  は回路電圧， $V_{RL}$  は測定電圧である．各ガスセンサの負荷抵抗  $R_L$  と回路電圧  $V_c$  については図 7 を参照されたい．

## 4.2 計測制御プログラムの実装

匂いを測定する際には，4 種類の半導体ガスセンサからの応答を任意の時間間隔で同時に読み込む必要があり，人手では困難である．そこで，ポケットコンピュータを利用して匂い測定行程を自動化する．測定の自動化は，実験の再現性を高め，得られるデータの信頼性を向上させるという点においても大きなメリットがある．

計測制御プログラムの例を以下に示す．本プログラムは，ガスセンサ内部ヒータのオンオフ制御，匂いサンプル気化用ヒータのオンオフ制御，換気兼拡散用ファンのオンオフ制御を行う．匂い測定装置のゲート開閉と匂いサンプルの導入についてはマニュアル作業とする．なお，本プログラムでは匂い測定の時間間隔を 10[sec] に設定している．



```

#define CH1 0x24
#define CH2 0x25
#define CH3 0x26
#define CH4 0x27
#define RELAY 0x21
#define BUZZER 0x22
#define P8255 0x23

struct PROCEDURE
{
    char msg[80];
    int sec;
    int Relay1; /* Heaters for gas sensors */
    int Relay2; /* A heater for gas vaporization */
    int Relay3; /* Ventilation fan */
} Proc[7] =
{
    { "(1) Ventilation of container air      ", 70,1,0,1 },
    { "(2) Dehumidification and deodorization", 60,1,0,1 },
    { "(3) Vaporization of sample gas        ", 300,1,1,0 },
    { "(4) Circulation of sample gas         ", 45,1,0,1 },
    { "(5) Measurement of smell response     ", 920,0,0,0 },
    { "(6) Ventilation of container air      ", 90,1,0,1 },
    { "(7) Clean syringe and sample heater   ", 120,0,0,0 }
};

struct HEATER_CONTROL
{
    int interval;
    int repeat;
} HeaterControl[14]=

{
    { ???, ??? }, /* Phase 1 */
    { ???, ??? }, /* Phase 2 */
    { ???, ??? }, /* Phase 3 */
    { ???, ??? }, /* Phase 4 */
    { ???, ??? }, /* Phase 5 */
    { ???, ??? }, /* Phase 6 */
    { ???, ??? }, /* Phase 7 */
    { ???, ??? }, /* Phase 8 */
    { ???, ??? }, /* Phase 9 */
    { ???, ??? }, /* Phase 10 */
    { ???, ??? }, /* Phase 11 */
    { ???, ??? }, /* Phase 12 */
    { ???, ??? }, /* Phase 13 */
    { ???, ??? } /* Phase 14 */
};
/* 906[s] in all. 40 bytes for each sampling point */

WaitLoop( int s ) /* Wait s seconds for doing nothing */
{
    int i,i2;

    for( i=0; i<s ;i++ )
        for( i2=0; i2<2300 ;i2++ );

    return 0;
}

Relay1_ON() /* Turn on Relay1 */
{
    unsigned char p;

    p = inport( RELAY ) | 0x01 ;
    outport( RELAY, p );
}

```

```

    return 0;
}

Relay1_OFF() /* Turn off Relay1 */
{
    unsigned char p;

    p = inport( RELAY ) & 0xfe ;
    outport( RELAY, p );
    return 0;
}

Relay2_ON() /* Turn on Relay2 */
{
    unsigned char p;

    p = inport( RELAY ) | 0x02 ;
    outport( RELAY, p );
    return 0;
}

Relay2_OFF() /* Turn off Relay2 */
{
    unsigned char p;

    p = inport( RELAY ) & 0xfd ;
    outport( RELAY, p );
    return 0;
}

Relay3_ON() /* Turn on Relay3 */
{
    unsigned char p;

    p = inport( RELAY ) | 0x04 ;
    outport( RELAY, p );
    return 0;
}

Relay3_OFF() /* Turn off Relay3 */
{
    unsigned char p;

    p = inport( RELAY ) & 0xfb ;
    outport( RELAY, p );
    return 0;
}

RelayControl( int m ) /* Control relays for each procedure */
{
    if ( Proc[m].Relay1 ) Relay1_ON(); /* Turn on Relay1 */
    else Relay1_OFF(); /* Turn off Relay1 */

    if ( Proc[m].Relay2 ) Relay2_ON(); /* Turn on Relay2 */
    else Relay2_OFF(); /* Turn off Relay2 */

    if ( Proc[m].Relay3 ) Relay3_ON(); /* Turn on Relay3 */
    else Relay3_OFF(); /* Turn off Relay3 */

    return 0;
}

ReverseRelay() /* Reverse the state of relay for gas sensor's heaters */
{

```

```

static int s=0;
unsigned char p;

if ( s==0 ) /* On */
{
    p=inport( RELAY ) | 0x01 ;
    s=1;
} else
{ /* Off */
    p=inport( RELAY ) & 0xfe ;
    s=0;
}
outport( RELAY, p );
return 0;
}

SmellSense( char *fname )
{
    int i,i2,i3,i4,i5,n;
    unsigned char indata[4];
    float chdata[4];
    FILE *fp;

    if ( (fp=fopen(fname,"w"))==NULL )
    {
        printf("Failed to open file.\n");
        exit(1);
    }

    n=0;
    for( i=0; i<14 ;i++ )
    {
        for( i2=0; i2<HeaterControl[i].repeat ;i2++ )
        {
            ReverseRelay();
            for( i3=0; i3<HeaterControl[i].interval ;i3++ )
            {
                if ( n++ % 10 == 0 )
                {
                    for( i4=0; i4<4 ;i4++ )
                    {
                        outport( CH1+i4,0 ); /* Start A-D conversion */
                        for( i5=0; i5<300 ;i5++ ); /* Wait loop */
                        indata[i4] = inport( CH1+i4 ); /* Read out data */
                        chdata[i4] = 5.*(float)indata[i4]/255.; /* Scale input to analogue value */
                    }
                    printf( "n=%d ch1=%f, ch2=%f, ch3=%f, ch4=%f\n"
                        ,n ,chdata[0],chdata[1],chdata[2],chdata[3] );
                    fprintf( fp,"%f %f %f %f\n"
                        ,chdata[0],chdata[1],chdata[2],chdata[3] );
                }
                WaitLoop( 1 );
            }
        }
    }
    fclose( fp );
    return 0;
}

main()
{
    int i;
    char fname[14]; /* File name must be within 8 characters */

    outport( P8255 ,0x90 ); /* Initialize 8255 */

```

```

outport( BUZZER,0 ); /* Turn off buzzer */

for( i=0; i<7 ;i++ )
{
    printf("Procedure %d\n",i+1);
    printf("%s (%3d)\n", Proc[i].msg, Proc[i].sec );

    switch( i )
    {
        case 0: /* Ventilation of container air */
            printf("Open the internal and external gates.\n");
            RelayControl( i );
            WaitLoop( Proc[i].sec );
            break;

        case 1: /* Dehumidification and deodorization */
            printf("Close the external gate.\n");
            RelayControl( i );
            WaitLoop( Proc[i].sec );
            break;

        case 2: /* Vaporization of sample gas */
            printf("Close the internal gate.\n");
            printf("Inject the odor sample onto the heater.\n");
            RelayControl( i );
            WaitLoop( Proc[i].sec );
            break;

        case 3: /* Circulation of sample gas */
            printf("Circulating sample gas. Wait.\n");
            RelayControl( i );
            WaitLoop( Proc[i].sec );
            break;

        case 4: /* Measurement of smell response */
            RelayControl( i );
            printf("Input file name: ");
            scanf("%s",fname);
            printf("Recording smell response to file: %s\n",fname);
            SmellSense( fname );
            break;

        case 5: /* Ventilation of container air */
            printf("Open the internal and external gates.\n");
            RelayControl( i );
            WaitLoop( Proc[i].sec );
            break;

        case 6: /* Clean syringe and heaters */
            printf("Please clean sample heater and syringe.\n");
            RelayControl( i );
            WaitLoop( Proc[i].sec );
            break;

        default: break;
    }
}

printf("Finished measuring odor sample.\n");

return 0;
}

```

(注) 実験データを格納するラムデータファイル領域を事前に Init 機能を利用して確保しておくこと (1 データファイル当たり 3 6 4 0 [byte])。Init 機能の詳細は、PC-G850 取扱説明書を参照すること。

## 5 実験結果

- (1) ポケットコンピュータ内のラムデータファイルに格納した測定データを読み出し、各ガスセンサの素子抵抗を求めるプログラムをC言語により作成せよ。
- (2) 各匂いサンプルごとに、各ガスセンサの素子抵抗変化を描画せよ（横軸を時間 [sec] とする）。素子抵抗値が無限大となる場合があるが、その場合は、抵抗値を 0 として処理すること。
- (3) 各区間における各ガスセンサの素子抵抗の平均値と分散を求めて表にまとめよ。

## 6 考察及び検討

- (1) 表 4 の測定手順 1 ～ 14 の内、匂いサンプル間の差異が最大となる区間はどれか？
- (2) 各匂いサンプルについて最適と思われる半導体ガスセンサを選ぶ際、どのような特徴に注目すれば良いか？
- (3) 半導体ガスセンサ内部ヒータのオンオフ制御の有効性について考察せよ。
- (4) ガスクロマトグラフィー法について調べよ。
- (5) 半導体ガスセンサによる匂い測定法が、ガスクロマトグラフィー法に勝る点を挙げよ。
- (6) ポケットコンピュータを計測制御に利用する長所・短所を挙げよ。
- (7) 測定データに基づき匂いを自動識別する際、どのような工学技術が必要になるか？

## 参考文献

- [1] 森泉豊栄：“匂いセンシングシステム” 電気化学学会論文誌 No.3 (1994)
- [2] 高木貞敬，渋谷達明：“匂いの科学” 朝倉書店 (1989)
- [3] 長谷川香料（株）編：“においの化学” 裳華房 (1988)
- [4] 都甲潔，山藤馨：“美味しさを測る” ブルーバックス，講談社
- [5] 工業調査会編集部編 工業調査会：“センサ活用技術” pp.276-277, pp.396-401
- [6] 自動化技術編集部編 工業調査会：“やさしいセンサ技術” pp.48-60, pp.238-250

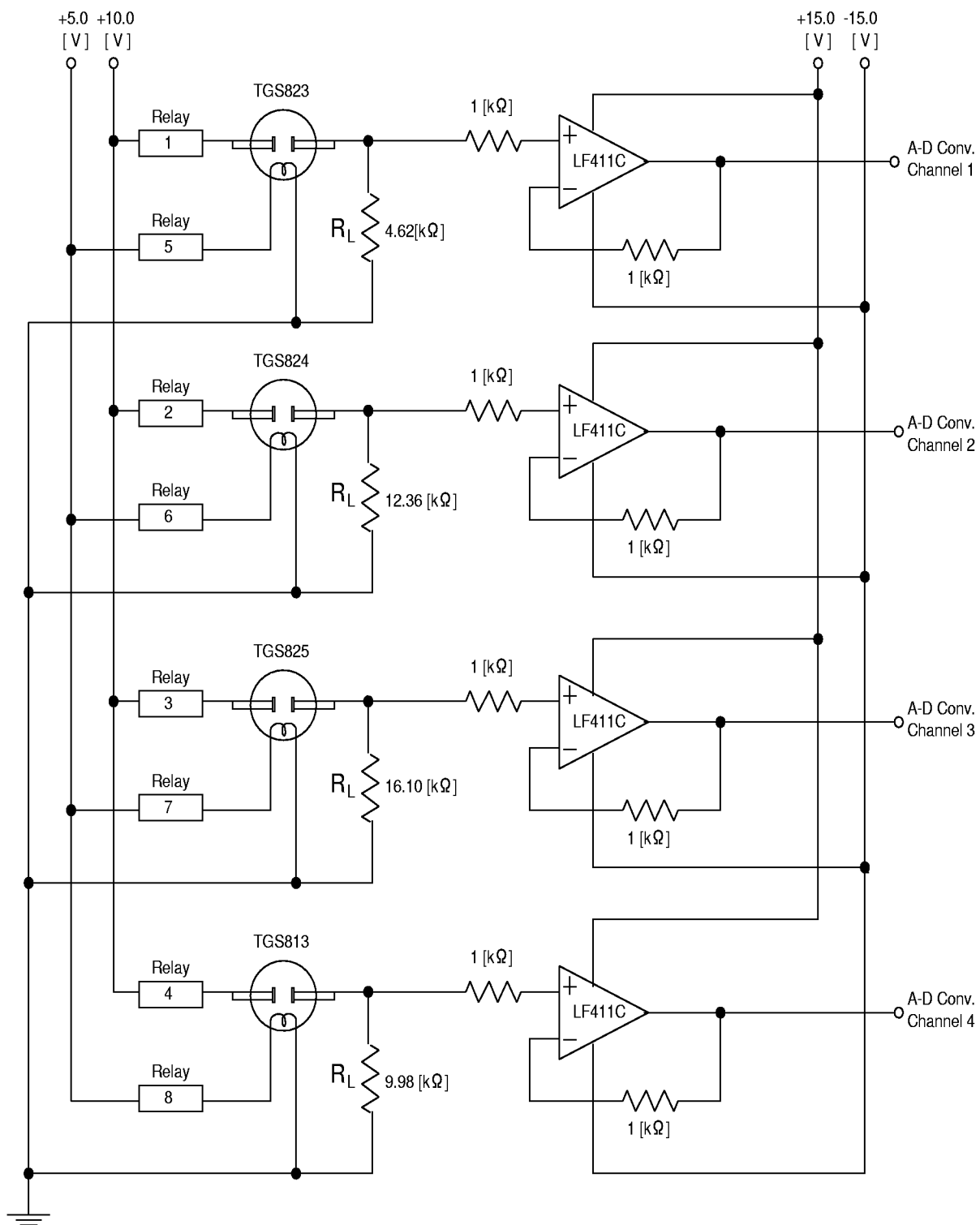


図 7: 半導体ガスセンサ基本測定回路図