

簡易トランジスタ判定ユニット(PIC18F2550)

概要

新旧内外含め各種トランジスタを保有(数百個程度)しています。しかし、物によっては、型番が分からなかったり、古過ぎて規格表に記載されていないために、ピン(足)のどれが、エミッタ、コレクタ、ベー



スの分からないものも有ります。

回は、非測定物として、3本の足を無作為に接続したトランジスタを自動的に測定し、

そこで今

- 型式(NPN/PNP)
- ピン配置(エミッタ/コレクタ/ベース)

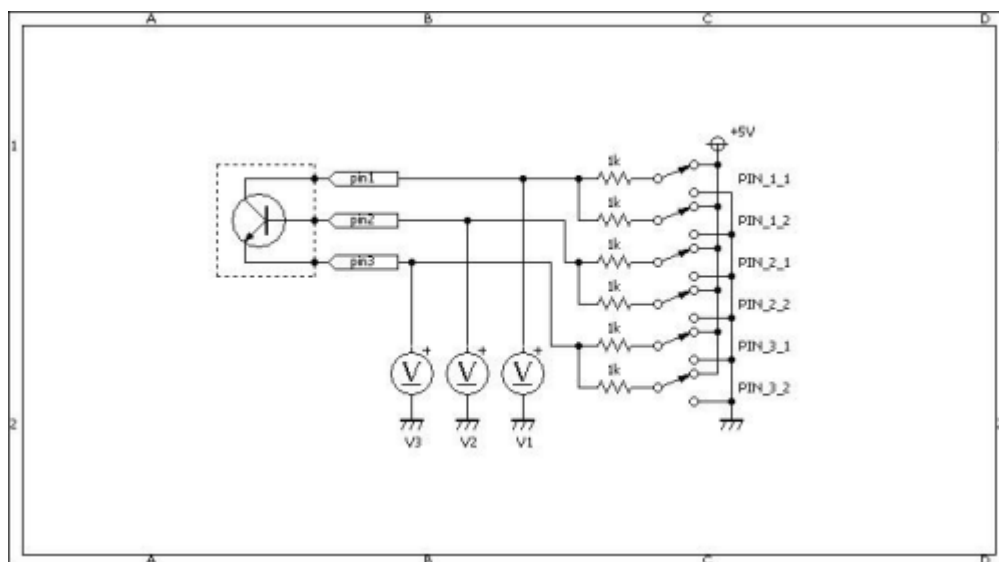
を表示させるユニットを製作してみました。

動作原理

出来るだけ“ シンプル ” な構成で実現してみました。 トランジスタの各ピン(足)の接続は、次の3通りです。

1. $+5V$ に対して「 500Ω ($1k\Omega$ 2個の並列)」で接続する。
2. GND に対して「 500Ω ($1k\Omega$ 2個の並列)」で接続する。
3. $+5V$ に対して $1k\Omega$ GND に対して $1k\Omega$ で接続する。

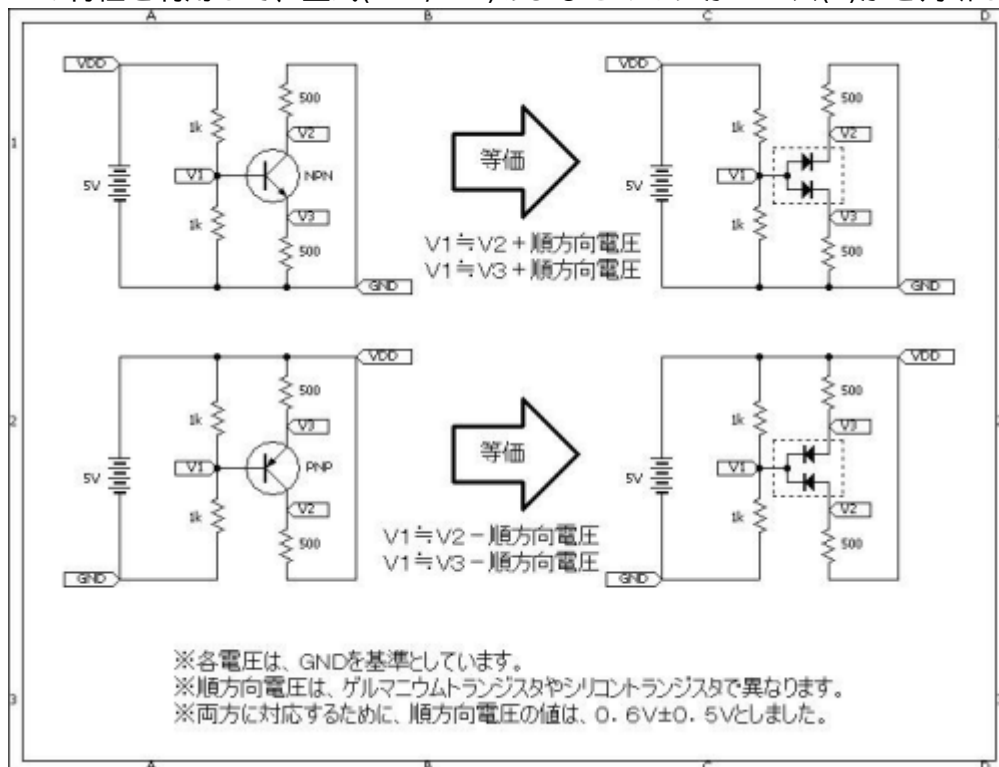
接続を切り換える毎に、各ピンの電圧を測定し、その電圧値より型式やピン配置を判断します。



<型式(NPN/PNP)の判定> ベース・エミッタ間、ベース・コレクタ間は、一種のダイオードと見なすことが出来ます。

- NPNの場合には、ベースからコレクタおよびエミッタに向けて電流が流れます。
- PNPの場合には、コレクタおよびエミッタからベースに向けて電流が流れます。

この特性を利用して、型式(NPN/PNP)およびどのピンがベース(B)かを判断します。



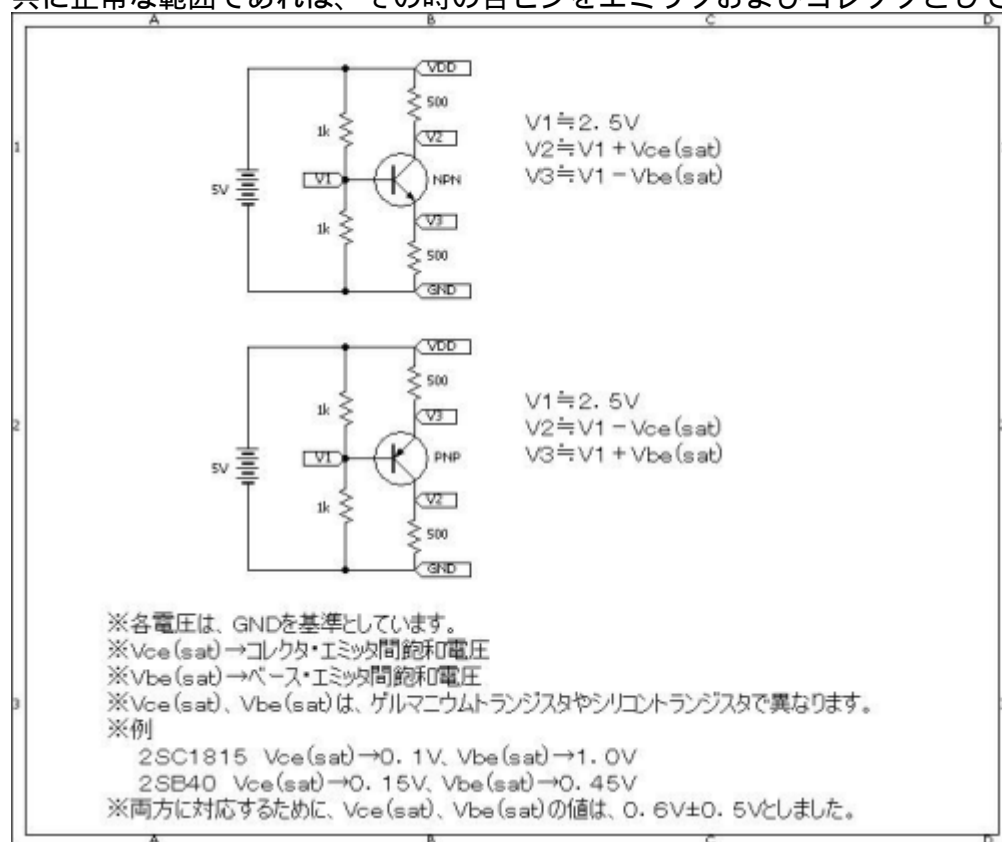
その時の組み合わせは、6パターンです。

型式	ピン(1)	ピン(2)	ピン(3)	PIN_1_1	PIN_1_2	PIN_2_1	PIN_2_2	PIN_3_1	PIN_3_2
NPN	B	?	?	+5V	GND	GND	GND	GND	GND
	?	B	?	GND	GND	+5V	GND	GND	GND
	?	?	B	GND	GND	GND	GND	+5V	GND
PNP	B	?	?	GND	+5V	+5V	+5V	+5V	+5V
	?	B	?	+5V	+5V	GND	+5V	+5V	+5V
	?	?	B	+5V	+5V	+5V	+5V	GND	+5V

<ピン配置(エミッタ/コレクタ/ベース)の判定> 型式とベースが分かったので、次はエミッタとコレクタを判断します。型式とベースが分かっているので、ベースに電圧を加え、次の2項目をチェックします。

- コレクタ・エミッタ間飽和電圧が正常な範囲であるか?
- ベース・エミッタ間飽和電圧が正常な範囲であるか?

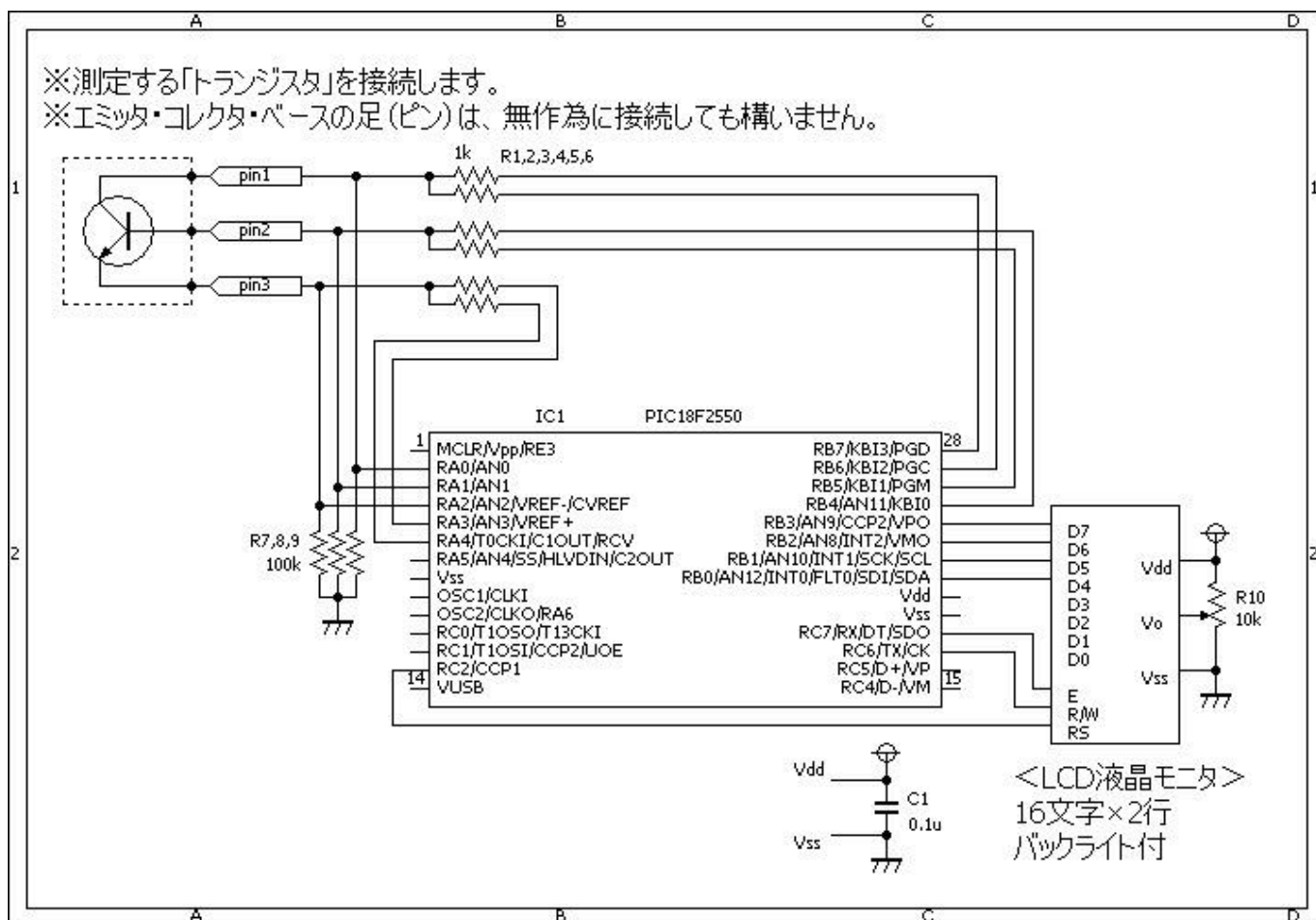
共に正常な範囲であれば、その時の各ピンをエミッタおよびコレクタとして判断します。



ピン(1)	ピン(2)	ピン(3)
E	C	B
C	E	B
C	B	E
E	B	C
B	C	E
B	E	C

その時の組み合わせは、6パターンです。

回路図



ソースコード

[tr_checker_v2.c](#)

```
//*****  
*  
/*  
    <トランジスタチェッカー (タイプ、ピン配置自動判定)>  
*/  
//*****  
*  
  
#define      PIN_1_1      PORTB.F7  
#define      PIN_1_2      PORTB.F6  
#define      PIN_2_1      PORTB.F5  
#define      PIN_2_2      PORTB.F4  
#define      PIN_3_1      PORTA.F3  
#define      PIN_3_2      PORTA.F4  
  
#define      AD1          0  
#define      AD2          1  
#define      AD3          2
```

```
#define      NPN_TYPE_1      1
#define      NPN_TYPE_2      2
#define      NPN_TYPE_3      3
#define      PNP_TYPE_1      4
#define      PNP_TYPE_2      5
#define      PNP_TYPE_3      6
#define      NON_TYPE        0

#define      RANGE_1          600
#define      RANGE_2          500

#define      ON                1
#define      OFF               0

//*****
*

unsigned Adc_Read_Ex(unsigned short channel)
{
    static unsigned int      ad;
    static unsigned short    cnt;
    //
    ad = 0;
    for (cnt = 0; cnt < 50; cnt++) {
        ad += Adc_Read(channel);
    }
    return (ad / 10);
}

//*****
*

static char      buf[8];
static unsigned  ad1, ad2, ad3;

void measurement()
{
    ad1 = Adc_Read_Ex(AD1);
    ad2 = Adc_Read_Ex(AD2);
    ad3 = Adc_Read_Ex(AD3);

    WordToStr(ad1, buf);
    Lcd_Custom_Out(1, 1, &buf[1]);
    WordToStr(ad2, buf);
    Lcd_Custom_Out(1, 6, &buf[1]);
    WordToStr(ad3, buf);
    Lcd_Custom_Out(1, 11, &buf[1]);
}

//*****
```

```
*

short range_chk(unsigned a, unsigned b, unsigned c)
{
    if ((a + b) < c)
        return (-1);
    if ((a - b) > c)
        return (-1);
    return (0);
}

//*****
*

char *msg_BEC = "BEC";
char *msg_BCE = "BCE";
char *msg_EBC = "EBC";
char *msg_CBE = "CBE";
char *msg_ECB = "ECB";
char *msg_CEB = "CEB";

char *msg_NON = "???";
char *msg_NPN = "NPN";
char *msg_PNP = "PNP";

void npn(short type)
{
    switch (type) {
    case NPN_TYPE_1:
        PIN_1_1 = ON;    //(B)
        PIN_1_2 = OFF;   //(B)
        PIN_2_1 = ON;    //(C)
        PIN_2_2 = ON;    //(C)
        PIN_3_1 = OFF;   //(E)
        PIN_3_2 = OFF;   //(E)
        measurement();
        if ((range_chk(ad1 + RANGE_1, RANGE_2, ad2) == 0) &&
            (range_chk(ad1 - RANGE_1, RANGE_2, ad3) == 0)) {
            Lcd_Custom_Out(2, 5, msg_BCE);
            return;
        }
        PIN_1_1 = ON;    //(B)
        PIN_1_2 = OFF;   //(B)
        PIN_2_1 = OFF;   //(E)
        PIN_2_2 = OFF;   //(E)
        PIN_3_1 = ON;    //(C)
        PIN_3_2 = ON;    //(C)
        measurement();
        if ((range_chk(ad1 + RANGE_1, RANGE_2, ad3) == 0) &&
            (range_chk(ad1 - RANGE_1, RANGE_2, ad2) == 0)) {
```

```
Lcd_Custom_Out(2, 5, msg_BEC);
return;
}
break;
case NPN_TYPE_2:
    PIN_1_1 = OFF;    //(E)
    PIN_1_2 = OFF;    //(E)
    PIN_2_1 = ON;     //(B)
    PIN_2_2 = OFF;    //(B)
    PIN_3_1 = ON;     //(C)
    PIN_3_2 = ON;     //(C)
    measurement();
    if ((range_chk(ad2 + RANGE_1, RANGE_2, ad3) == 0) &&
(range_chk(ad2 - RANGE_1, RANGE_2, ad1) == 0)) {
        Lcd_Custom_Out(2, 5, msg_EBC);
        return;
    }
    PIN_1_1 = ON;     //(C)
    PIN_1_2 = ON;     //(C)
    PIN_2_1 = ON;     //(B)
    PIN_2_2 = OFF;    //(B)
    PIN_3_1 = OFF;    //(E)
    PIN_3_2 = OFF;    //(E)
    measurement();
    if ((range_chk(ad2 + RANGE_1, RANGE_2, ad1) == 0) &&
(range_chk(ad2 - RANGE_1, RANGE_2, ad3) == 0)) {
        Lcd_Custom_Out(2, 5, msg_CBE);
        return;
    }
    break;
case NPN_TYPE_3:
    PIN_1_1 = OFF;    //(E)
    PIN_1_2 = OFF;    //(E)
    PIN_2_1 = ON;     //(C)
    PIN_2_2 = ON;     //(C)
    PIN_3_1 = ON;     //(B)
    PIN_3_2 = OFF;    //(B)
    measurement();
    if ((range_chk(ad3 + RANGE_1, RANGE_2, ad2) == 0) &&
(range_chk(ad3 - RANGE_1, RANGE_2, ad1) == 0)) {
        Lcd_Custom_Out(2, 5, msg_ECB);
        return;
    }

    PIN_1_1 = ON;     //(C)
    PIN_1_2 = ON;     //(C)
    PIN_2_1 = OFF;    //(E)
    PIN_2_2 = OFF;    //(E)
    PIN_3_1 = ON;     //(B)
    PIN_3_2 = OFF;    //(B)
    measurement();
```

```
        if ((range_chk(ad3 + RANGE_1, RANGE_2, ad1) == 0) &&
            (range_chk(ad3 - RANGE_1, RANGE_2, ad3) == 0)) {
            Lcd_Custom_Out(2, 5, msg_CEB);
            return;
        }
        break;
    }
}

//*****
*

void    pnp(short type)
{
    switch (type) {
    case PNP_TYPE_1:
        PIN_1_1 = ON;    //(B)
        PIN_1_2 = OFF;   //(B)
        PIN_2_1 = OFF;   //(C)
        PIN_2_2 = OFF;   //(C)
        PIN_3_1 = ON;    //(E)
        PIN_3_2 = ON;    //(E)
        measurement();
        if ((range_chk(ad1 + RANGE_1, RANGE_2, ad3) == 0) &&
            (range_chk(ad1 - RANGE_1, RANGE_2, ad2) == 0)) {
            Lcd_Custom_Out(2, 5, msg_BCE);
            return;
        }
        PIN_1_1 = ON;    //(B)
        PIN_1_2 = OFF;   //(B)
        PIN_2_1 = ON;    //(E)
        PIN_2_2 = ON;    //(E)
        PIN_3_1 = OFF;   //(C)
        PIN_3_2 = OFF;   //(C)
        measurement();
        if ((range_chk(ad1 + RANGE_1, RANGE_2, ad2) == 0) &&
            (range_chk(ad1 - RANGE_1, RANGE_2, ad3) == 0)) {
            Lcd_Custom_Out(2, 5, msg_BEC);
            return;
        }
        break;
    case PNP_TYPE_2:
        PIN_1_1 = ON;    //(E)
        PIN_1_2 = ON;    //(E)
        PIN_2_1 = ON;    //(B)
        PIN_2_2 = OFF;   //(B)
        PIN_3_1 = OFF;   //(C)
        PIN_3_2 = OFF;   //(C)
        measurement();
        if ((range_chk(ad2 + RANGE_1, RANGE_2, ad1) == 0) &&
```

```
(range_chk(ad2 - RANGE_1, RANGE_2, ad3) == 0)) {
    Lcd_Custom_Out(2, 5, msg_EBC);
    return;
}
PIN_1_1 = OFF;    //(C)
PIN_1_2 = OFF;    //(C)
PIN_2_1 = ON;     //(B)
PIN_2_2 = OFF;    //(B)
PIN_3_1 = ON;     //(E)
PIN_3_2 = ON;     //(E)
measurement();
if ((range_chk(ad2 + RANGE_1, RANGE_2, ad3) == 0) &&
(range_chk(ad2 - RANGE_1, RANGE_2, ad1) == 0)) {
    Lcd_Custom_Out(2, 5, msg_CBE);
    return;
}
break;
case PNP_TYPE_3:
    PIN_1_1 = ON;    //(E)
    PIN_1_2 = ON;    //(E)
    PIN_2_1 = OFF;   //(C)
    PIN_2_2 = OFF;   //(C)
    PIN_3_1 = ON;    //(B)
    PIN_3_2 = OFF;   //(B)
    measurement();
    if ((range_chk(ad3 + RANGE_1, RANGE_2, ad1) == 0) &&
(range_chk(ad3 - RANGE_1, RANGE_2, ad2) == 0)) {
        Lcd_Custom_Out(2, 5, msg_ECB);
        return;
    }
    PIN_1_1 = OFF;    //(C)
    PIN_1_2 = OFF;    //(C)
    PIN_2_1 = ON;     //(E)
    PIN_2_2 = ON;     //(E)
    PIN_3_1 = ON;     //(B)
    PIN_3_2 = OFF;    //(B)
    measurement();
    if ((range_chk(ad3 + RANGE_1, RANGE_2, ad2) == 0) &&
(range_chk(ad3 - RANGE_1, RANGE_2, ad1) == 0)) {
        Lcd_Custom_Out(2, 5, msg_CEB);
        return;
    }
    break;
}
}

//*****
*

short npn_pnp_chk()
{
```

```
PIN_1_1 = OFF;
PIN_1_2 = ON;
PIN_2_1 = OFF;
PIN_2_2 = OFF;
PIN_3_1 = OFF;
PIN_3_2 = OFF;
measurement();
if ((range_chk(ad1 - RANGE_1, RANGE_2, ad2) == 0) && (range_chk(ad1
- RANGE_1, RANGE_2, ad3) == 0)) {
    return (NPN_TYPE_1);
}
//
PIN_1_1 = OFF;
PIN_1_2 = OFF;
PIN_2_1 = OFF;
PIN_2_2 = ON;
PIN_3_1 = OFF;
PIN_3_2 = OFF;
measurement();
if ((range_chk(ad2 - RANGE_1, RANGE_2, ad1) == 0) && (range_chk(ad2
- RANGE_1, RANGE_2, ad3) == 0)) {
    return (NPN_TYPE_2);
}
//
PIN_1_1 = OFF;
PIN_1_2 = OFF;
PIN_2_1 = OFF;
PIN_2_2 = OFF;
PIN_3_1 = OFF;
PIN_3_2 = ON;
measurement();
if ((range_chk(ad3 - RANGE_1, RANGE_2, ad2) == 0) && (range_chk(ad3
- RANGE_1, RANGE_2, ad1) == 0)) {
    return (NPN_TYPE_3);
}
//
PIN_1_1 = OFF;
PIN_1_2 = ON;
PIN_2_1 = ON;
PIN_2_2 = ON;
PIN_3_1 = ON;
PIN_3_2 = ON;
measurement();
if ((range_chk(ad1 + RANGE_1, RANGE_2, ad2) == 0) && (range_chk(ad1
+ RANGE_1, RANGE_2, ad3) == 0)) {
    return (PNP_TYPE_1);
}
//
PIN_1_1 = ON;
PIN_1_2 = ON;
```

```

    PIN_2_1 = OFF;
    PIN_2_2 = ON;
    PIN_3_1 = ON;
    PIN_3_2 = ON;
    measurement();
    if ((range_chk(ad2 + RANGE_1, RANGE_2, ad1) == 0) && (range_chk(ad2
+ RANGE_1, RANGE_2, ad3) == 0)) {
        return (PNP_TYPE_2);
    }
    //
    PIN_1_1 = ON;
    PIN_1_2 = ON;
    PIN_2_1 = ON;
    PIN_2_2 = ON;
    PIN_3_1 = OFF;
    PIN_3_2 = ON;
    measurement();
    if ((range_chk(ad3 + RANGE_1, RANGE_2, ad2) == 0) && (range_chk(ad3
+ RANGE_1, RANGE_2, ad1) == 0)) {
        return (PNP_TYPE_3);
    }
    //
    return (NON_TYPE);
}

//*****
*

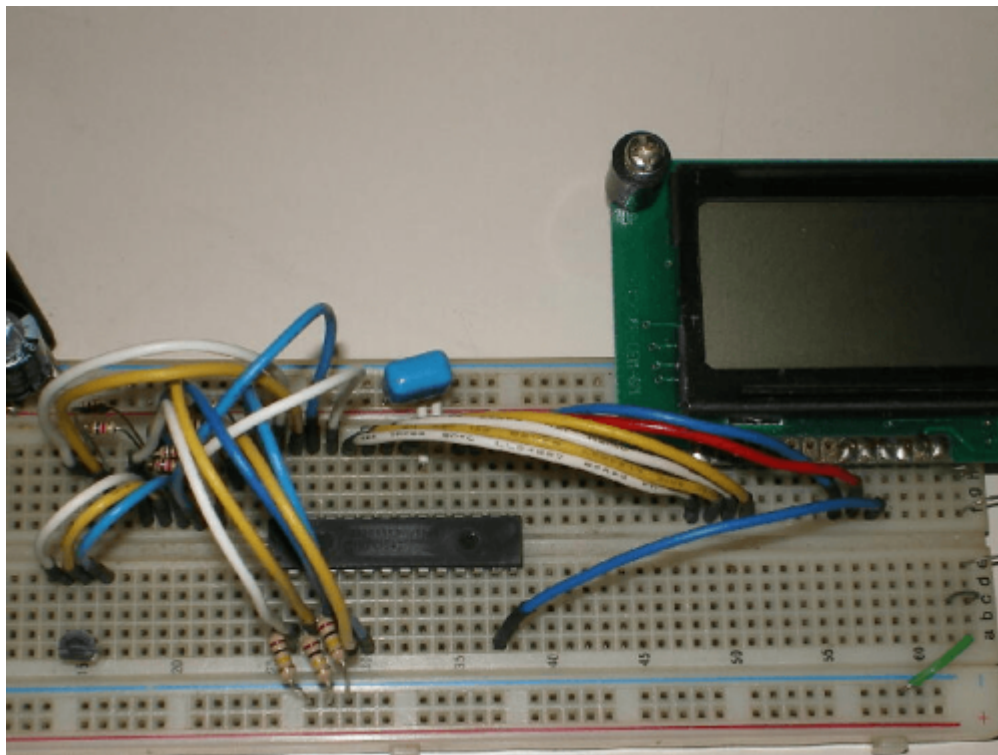
void main()
{
    //変数の定義
    static short type;
    //
    OSCCON.IRCF2 = 1;
    OSCCON.IRCF1 = 1;
    OSCCON.IRCF0 = 1;
    OSCCON.SCS1 = 1;
    OSCCON.SCS0 = 1;
    //変換の設定
    ADCON1.PCFG3 = 1;
    ADCON1.PCFG2 = 1;
    ADCON1.PCFG1 = 0;
    ADCON1.PCFG0 = 0;
    //ポートの設定
    TRISA = 0b00000111;
    TRISB = 0b00000000;
    TRISC = 0b00000000;
    //変数の初期化
    //変数の初期化
    Lcd_Custom_Config(&PORTB,3,2,1,0,&PORTC,2,6,7);
    Lcd_Custom_Cmd(LCD_CURSOR_OFF);

```

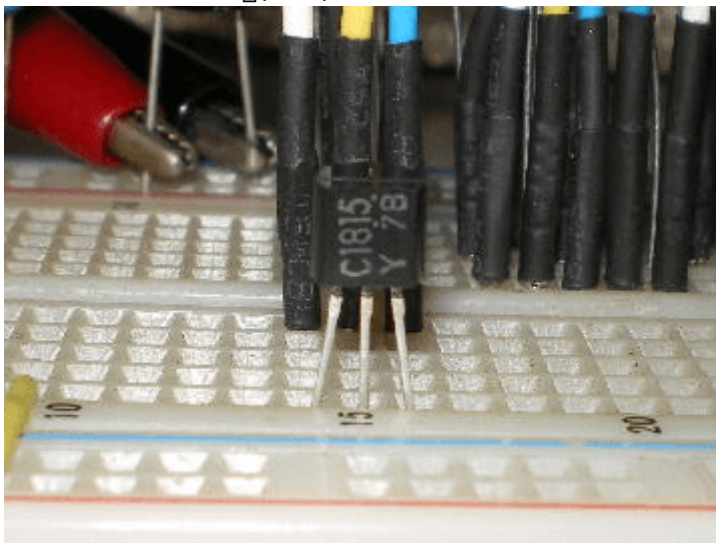
```
Lcd_Custom_Cmd(LCD_CLEAR);
//
while (1) {
    //
    type = npn_pnp_chk();
    switch (type) {
    case NPN_TYPE_1:
    case NPN_TYPE_2:
    case NPN_TYPE_3:
        Lcd_Custom_Out(2, 1, msg_NPN);
        npn(type);
        break;
    case PNP_TYPE_1:
    case PNP_TYPE_2:
    case PNP_TYPE_3:
        Lcd_Custom_Out(2, 1, msg_PNP);
        pnp(type);
        break;
    case NON_TYPE:
        Lcd_Custom_Out(2, 1, msg_NON);
        Lcd_Custom_Out(2, 5, msg_NON);
        break;
    }
    //
    Delay_ms(1000);
}
} //~!

//*****
*
```

動作確認

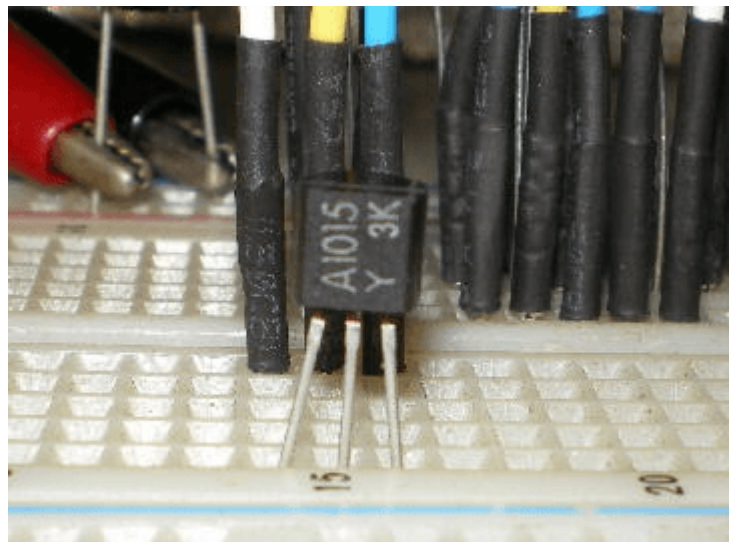
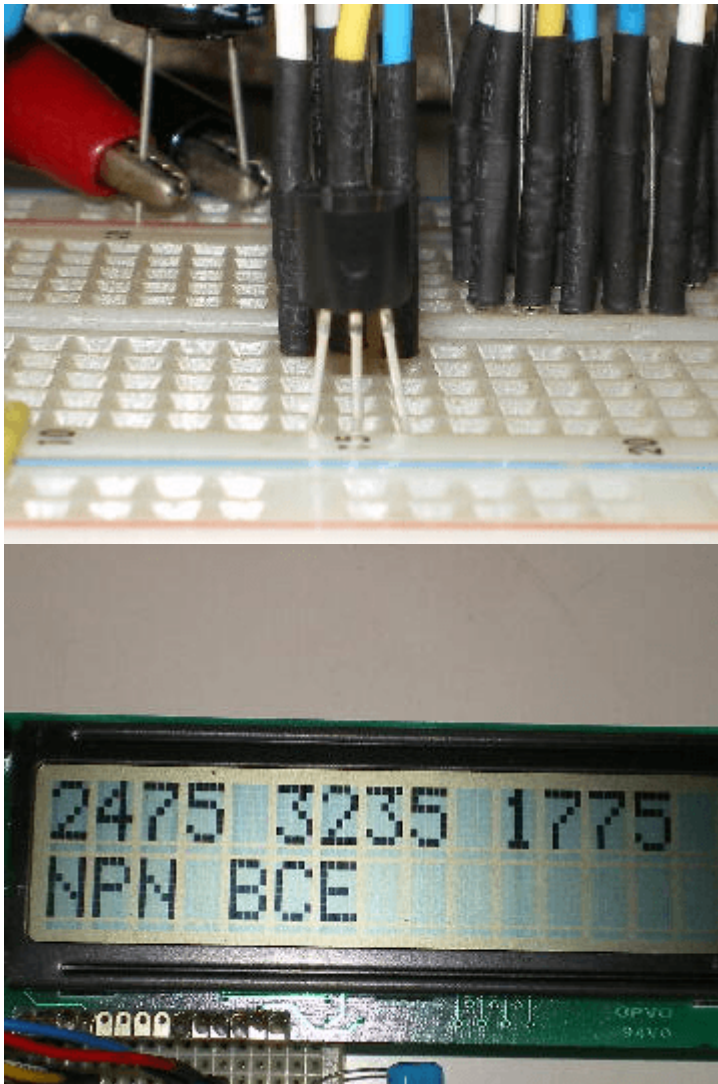


定番の、2SC1815を測定しました。上2枚が、印刷面を前にして測定した結果です□(ECB) 下2枚が、印刷面を後ろにして測定した結果です□(BCE) LCDの上部に表示している数値は、ピン1、2、3の電圧値で



す。(参考までに)





同じく定番の、2SA1015を測定しました。



いろいろ手持ちのトランジスタを測定してみました。



如何ですか? これでジャンク箱に眠っているトランジスタも日の目を見る機会が増えそうですね!?(😊)!
いろいろ手持ちのトランジスタを測定してみました。

Last update:
2025/10/17 14:27

elechobby:picdic:otherpic:181 <http://deepsky.jp/wiki/doku.php?id=elechobby:picdic:otherpic:181&rev=1588330837>

著作権表示 **copyright notice**

このページは稲崎様の閉鎖したHPのコピーで、著作権は稲崎様にあります。[詳細](#) This page is a copy of Mr. Inasaki's closed website, and the copyright is held by him.[Details](#)

From:

<http://deepsky.jp/wiki/> - うごくといいな

Permanent link:

<http://deepsky.jp/wiki/doku.php?id=elechobby:picdic:otherpic:181&rev=1588330837>

Last update: **2025/10/17 14:27**

