

IFT測定ユニット(PIC16F876)

概要

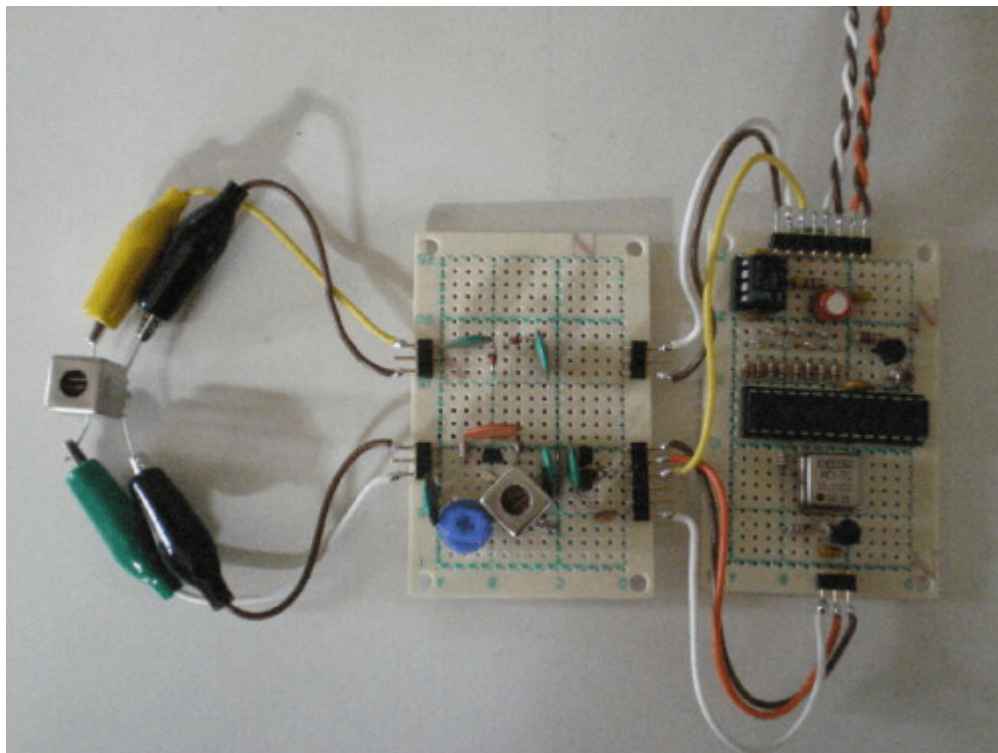
殆どの受信機では、スーパーヘテロダイン方式(super-heterodyne system)が採用されておりIFTが使用されています。IFT:(Intermediate-Frequency Transform) そこでこれらのIFTやセラミックフィルタの特性を測定するユニットを作成しました。但、今回は手持ちの部品の都合もあり455KhzのIFTおよびセラミックフィルタに限定しました。

動作原理

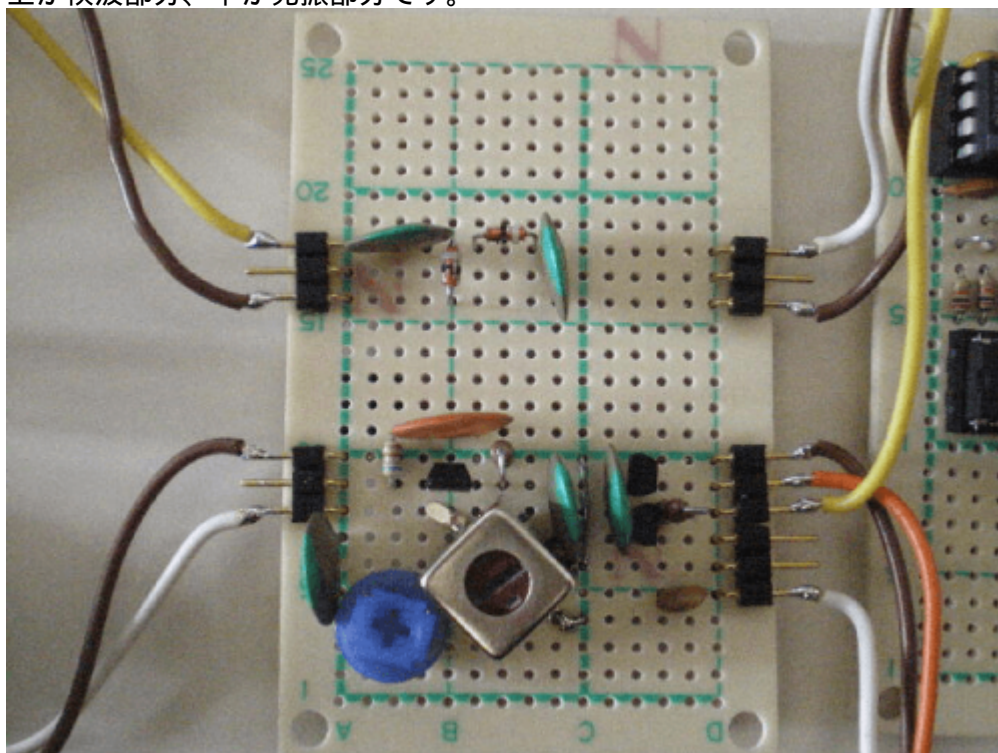
IFTやセラミックフィルタに、信号(約400Khz~500Khz)を入力し、その時の出力信号を測定し、それをグラフ表示させることによりIFTやセラミックフィルタの特性を視覚的に把握する。

1. 周波数を変化させるためにVCO(Voltage Controlled Oscillator)部分を作成します。尚、VCOの原理については、実験室のVCOを参照して下さい。
2. VCOに与える電圧は、PICのポートを10個使用し、10ビットのD/A変換(抵抗ラダー方式を採用)とします。
この時の出力電圧は、約3.5Vとなります。これはその後段で使用しているオペアンプ(LM358)が規格上、出力電圧の最大が、VCC(今回は5V)よりも1.5V低くなるためです。
これらの事よりVCOの精度は約3.4mV($3.5V \div 1024$)となります。
3. VCOから出力される信号は、次に非測定物であるIFTやセラミックフィルタに入力されます。
4. この時に、VCOからの信号の周波数をPICでカウントします。(パソコンでの表示のため)
5. 非測定物からの出力電圧は、ダイオードにより倍電圧検波します。
6. この検波電圧をPICでA/D変換し取り込みます。
7. これらのデータをパソコンへRS232C経由で送ります。以下の3項目を送信。
 - 1-1024のカウント値
 - VCOから出力される信号の周波数(Hz)の値
 - 検波後の信号レベル(mV)の値
8. パソコン側では、ハイパーターミナル(標準添付ソフト)でデータを受信します。
9. 最後に、このデータをExcelに取り込んでグラフ表示させれば、特性図が表示されることとなります。

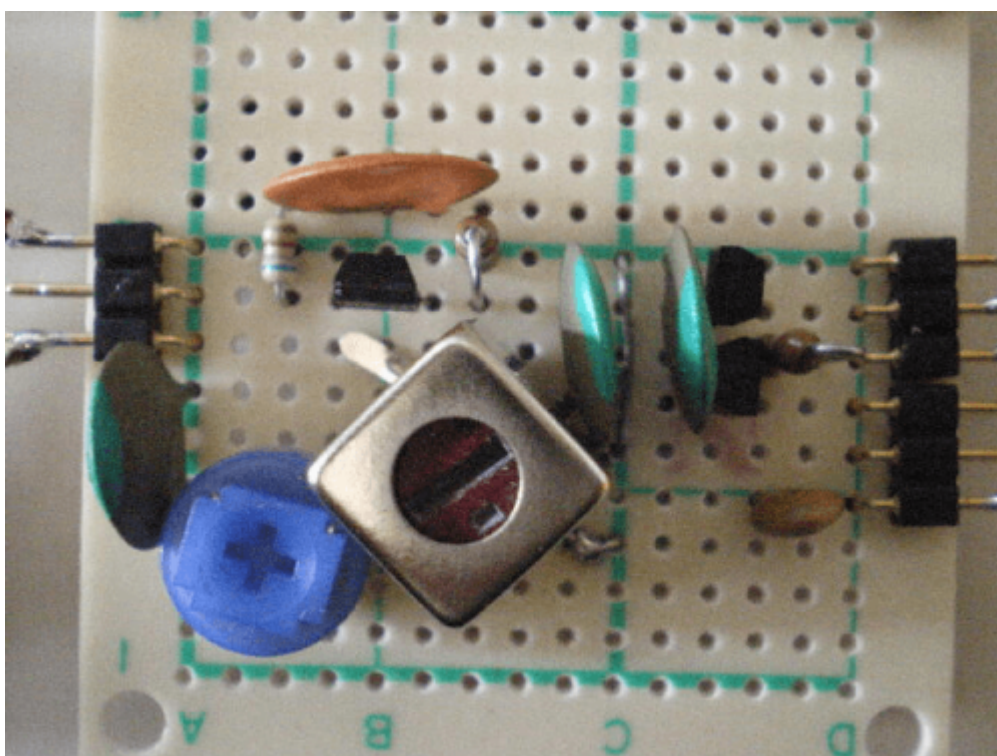
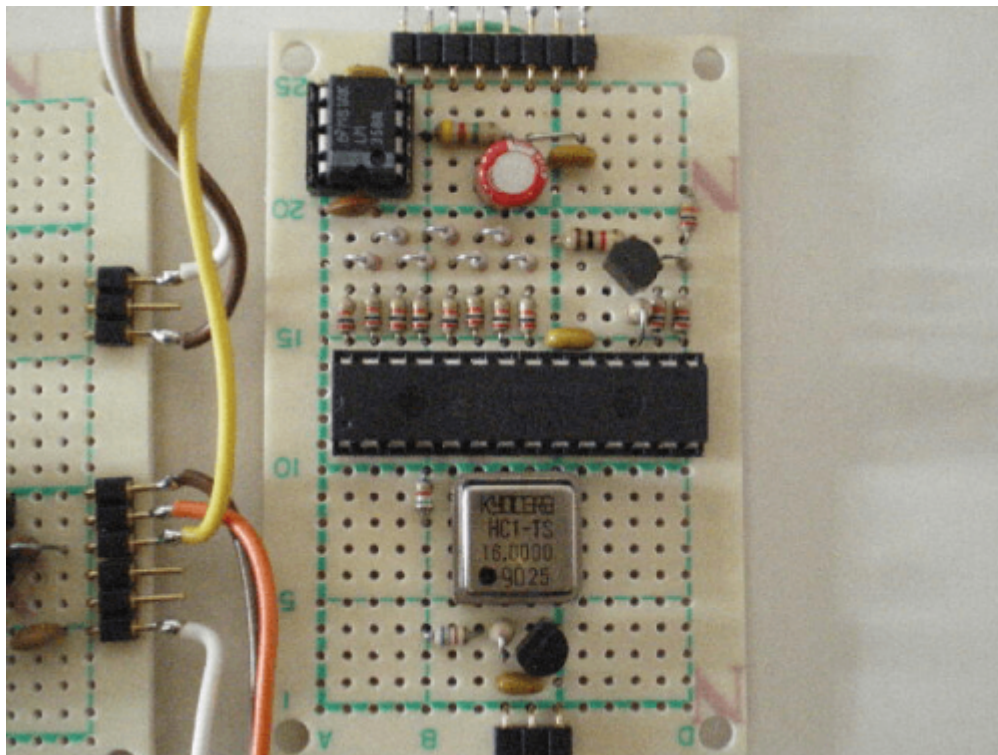
基板を制御部とそれ以外で2枚に分けました。



上が検波部分、下が発振部分です。

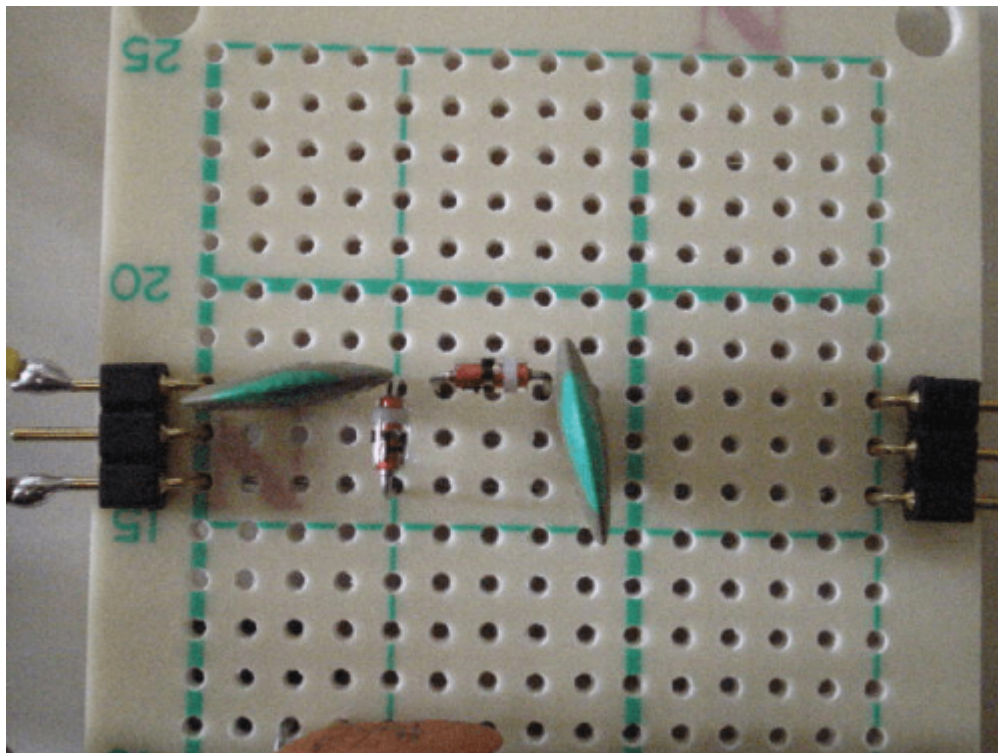


制御部です。抵抗が沢山並んでいる箇所がD/A変換(ラダー抵抗)部分です。

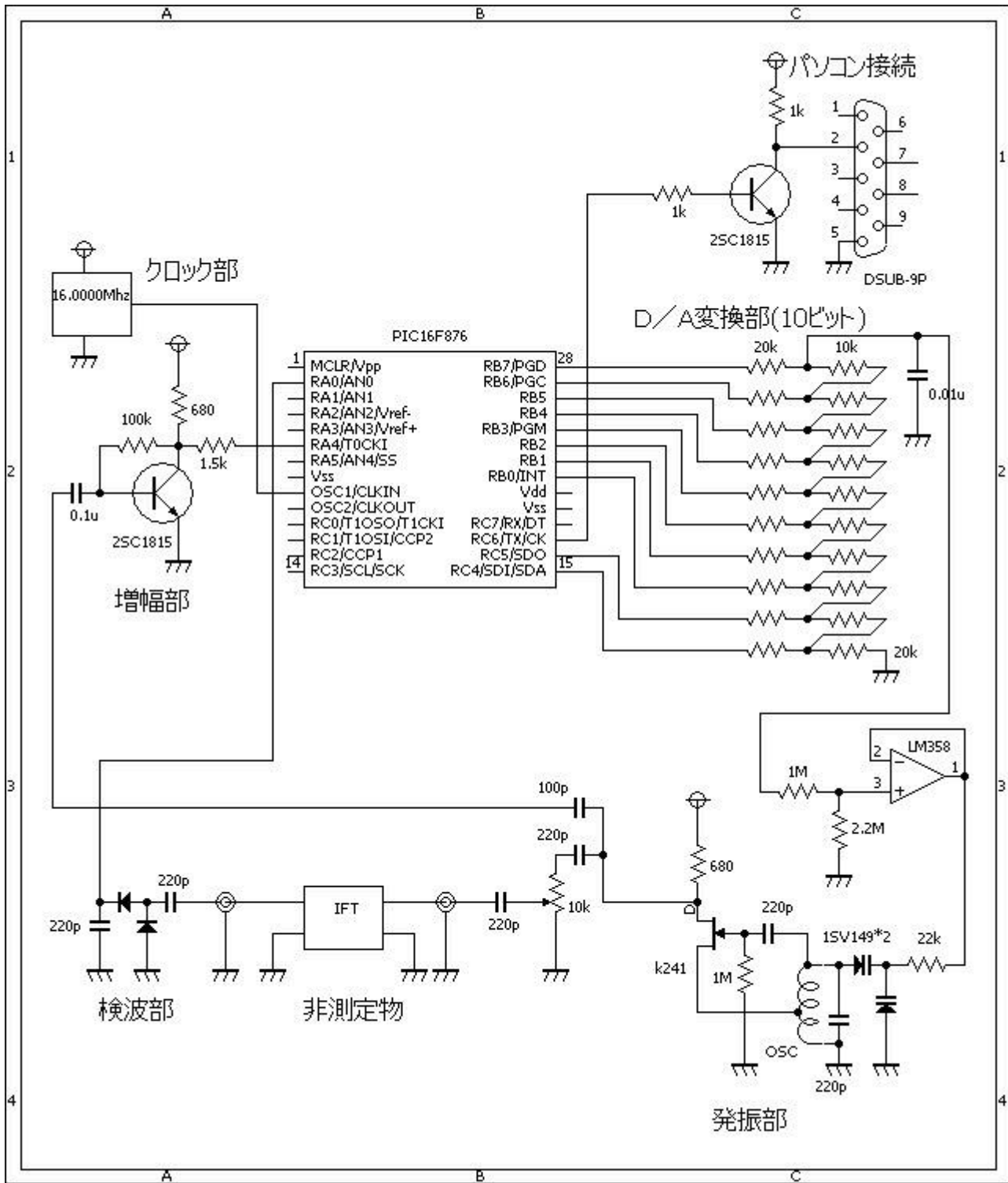


VCO部です。

検波部です。ダイオード2個、コンデンサ2個ととてもシンプルな回路です。



回路図



ソースコード

[iftTester.c](#)

```

//*****
*

void interrupt(){
    if (PIR1.CCP1IF == 1) {

```

```
    PIR1.CCP1IF = 0;
    //
    TRISA.F4 = 0;      // ゲートを閉める。
    PORTA.F4 = 0;
    T1CON.TMR1ON = 0; // TIMER1を停止する。
}
}

//*****
*

void Usart_Write_String(char *buf)
{
    short    len, i;
    len = strlen(buf);
    for (i = 0; i < len; i++) {
        Usart_Write(buf[i]);
    }
}

//*****
*

void main()
{
    // 変数の定義
    static unsigned long    freq; // 0...4294967295
    static unsigned int     ad0, cnt;
    static unsigned char    buf[20];
    // ポートの設定
    TRISA = 0b11110011;
    TRISB = 0b00000000;
    TRISC = 0b10000000;
    // アナログの設定
    ADCON1.F3 = 1;
    ADCON1.F2 = 1;
    ADCON1.F1 = 1;
    ADCON1.F0 = 0;
    // TIMER0の設定 今回は使用しない。
    INTCON.T0IE = 0;
    INTCON.T0IF = 0;
    TMR0 = 0;
    OPTION_REG.T0CS = 1;
    OPTION_REG.T0SE = 0;
    OPTION_REG.PSA = 0;
    OPTION_REG.PS0 = 0;
    OPTION_REG.PS1 = 1;
    OPTION_REG.PS2 = 0;
    // TIMER1の設定
    PIE1.TMR1IE = 0;
```

```
PIR1.TMR1IF = 0;
TMR1L = 0;
TMR1H = 0;
T1CON.T1CKPS0 = 1;
T1CON.T1CKPS1 = 1;
T1CON.TMR1ON = 0;
// CCPの設定
PIE1.CCP1IE = 1;
PIR1.CCP1IF = 0;
CCP1CON = 0b00001011;
CCPR1L = 0x50; // 0.1sec...10hz...クロックが16Mhzの時
CCPR1H = 0xC3; // 0.1sec...(1÷16000000)*4*8*50000
// RS232Cの設定
Usart_Init(9600);
Delay_ms(1000);
Usart_Write_String("\r\n");
Usart_Write_String("PIC16F876 IFT Tester R1.00\r\n");
//
TRISB = 0b00000000;
TRISC = 0b10000000;
// 割り込み(全体)の設定
INTCON.PEIE = 1;
INTCON.GIE = 1;
//
while(1) {
    for (cnt = 0; cnt < 1024; cnt++) {
        WordToStr(cnt + 1, buf);
        Usart_Write_String(buf);
        // D/A出力(10ビット)
        PORTC.F4 = (cnt & 0x1) == 0 ? 0 : 1;
        PORTC.F5 = (cnt & 0x2) == 0 ? 0 : 1;
        PORTB = (cnt >> 2) & 0xFF;
        // 周波数カウンタ関連の初期化
        T1CON.TMR1ON = 0;
        TRISA.F4 = 0;
        PORTA.F4 = 0;
        TMR0 = 0;
        INTCON.T0IF = 0;
        TMR1L = 0;
        TMR1H = 0;
        PIR1.TMR1IF = 0;
        freq = 0;
        // 開始
        T1CON.TMR1ON = 1;
        Delay_Cyc(3);
        asm nop;
        asm nop;
        asm nop;
        asm nop;
        TRISA.F4 = 1; //ゲートを開ける。
        // 周波数の測定
```

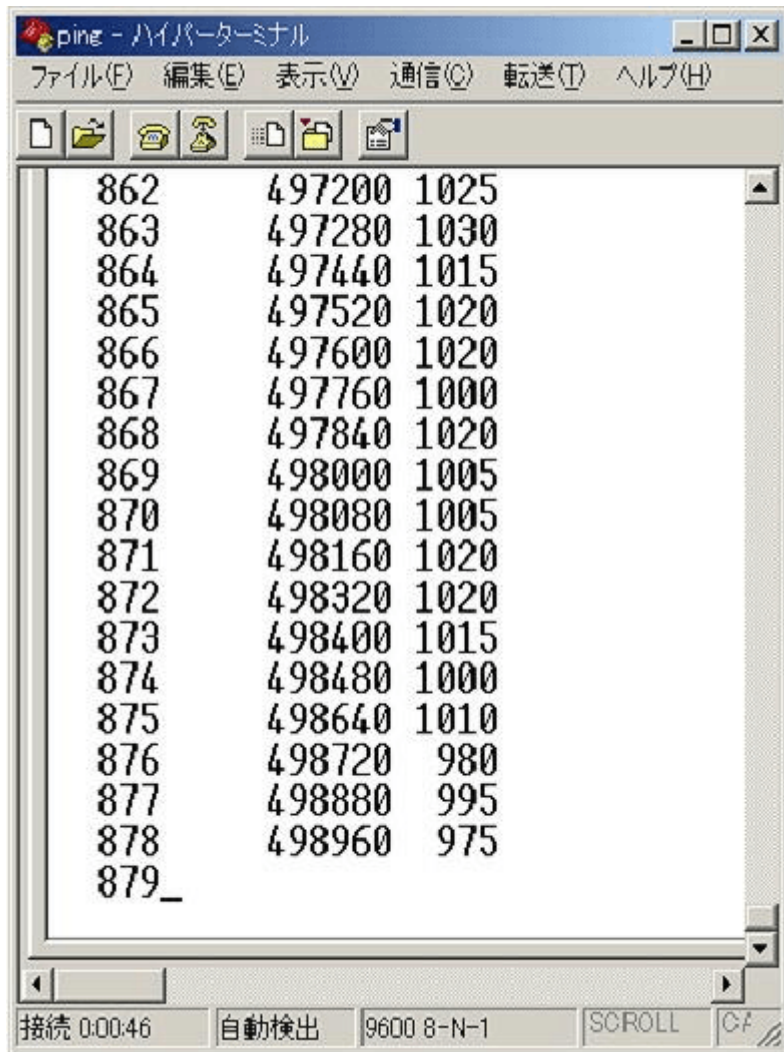
```
while (TRISA.F4 == 1) {
    if (INTCON.T0IF == 1) {
        INTCON.T0IF = 0;
        freq++;
    }
}
freq *= 256;
freq += TMR0;
// 周波数の補正
freq *= 8;
freq *= 10;
// 周波数の表示
LongToStr(freq, buf);
Usart_Write_String(buf);
// A/D変換と表示
ad0 = Adc_Read(0);
WordToStr(ad0 * 5, buf);
Usart_Write_String(buf);
Usart_Write_String("\r\n");
}
}
}

//*****
```

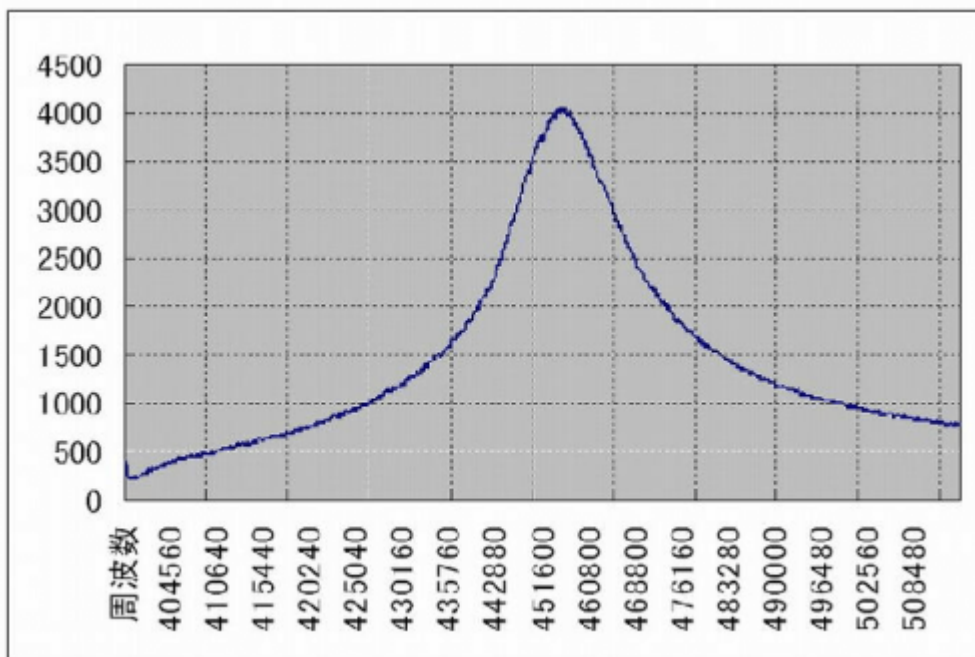
動作確認

ハイパーターミナルでPICから送られてくるデータを受信し表示します。

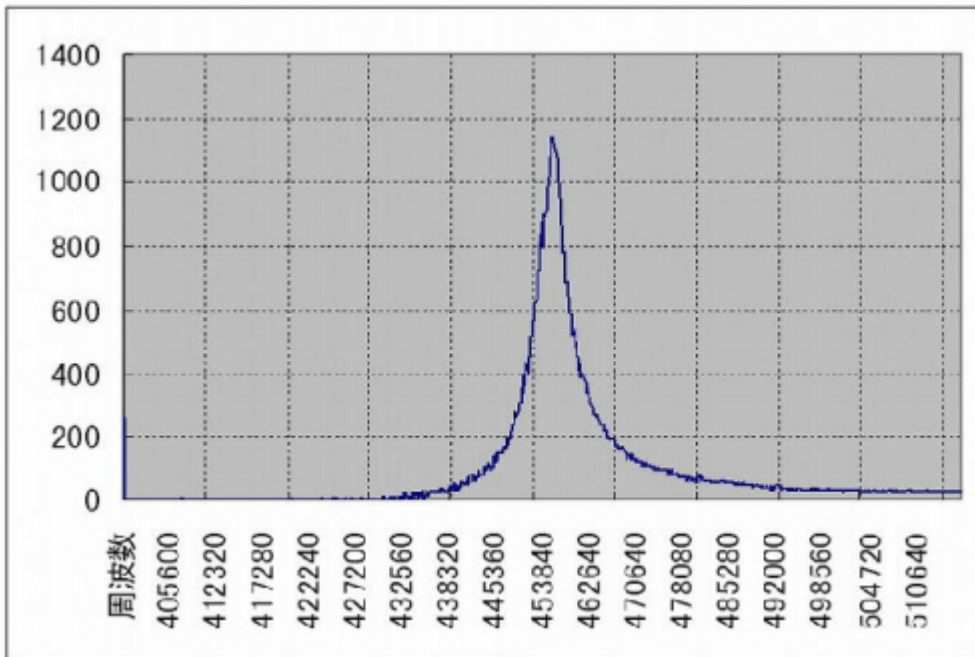
- 1~1024のカウント値
- VCOから出力される信号の周波数(Hz)の値
- 検波後の信号レベル(mV)の値



ハイパーターミナルのデータをExcelにカットアンドペーストしてグラフ表示します。一般的なトランジスタラジオで使用されているIFTを測定してみました。



次に、セラミックフィルタを測定しました。中々シャープな切れ味ですね。でも感度は？



From: <http://www.deepsky.jp/wiki/> - うごくといいな

Permanent link: <http://www.deepsky.jp/wiki/doku.php?id=elechobby:picdic:otherpic:162&rev=1588235881>

Last update: 2025/10/17 14:27

