

音レベル表示とリレー付き

シンプルな音センサー装置

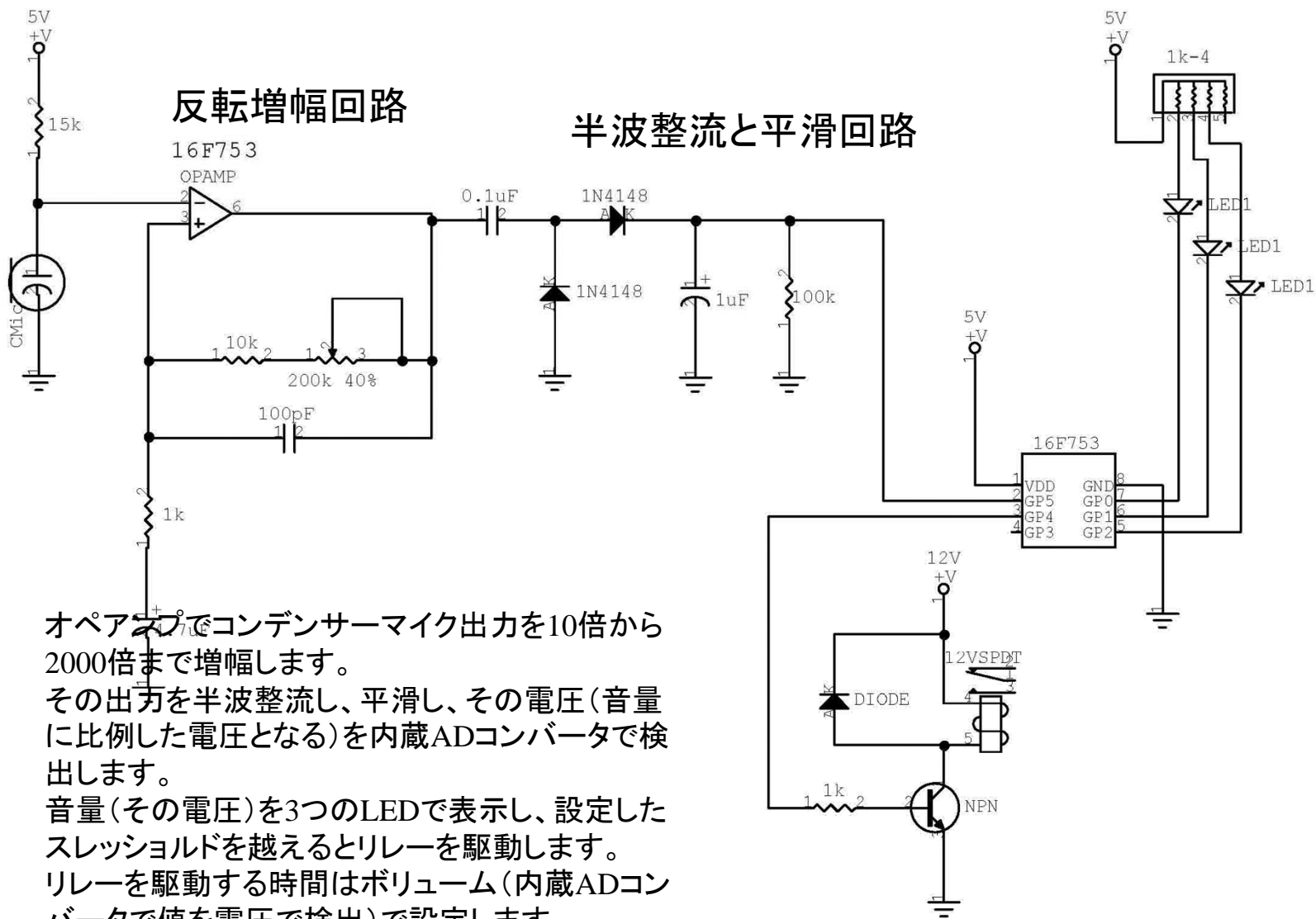
2014/3/30 川口昌良



概要

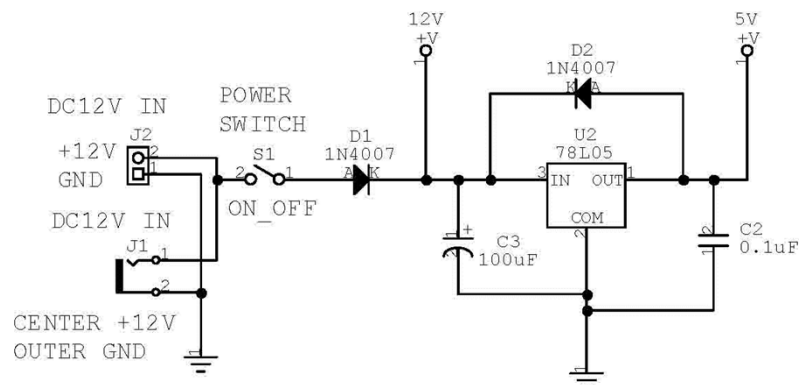
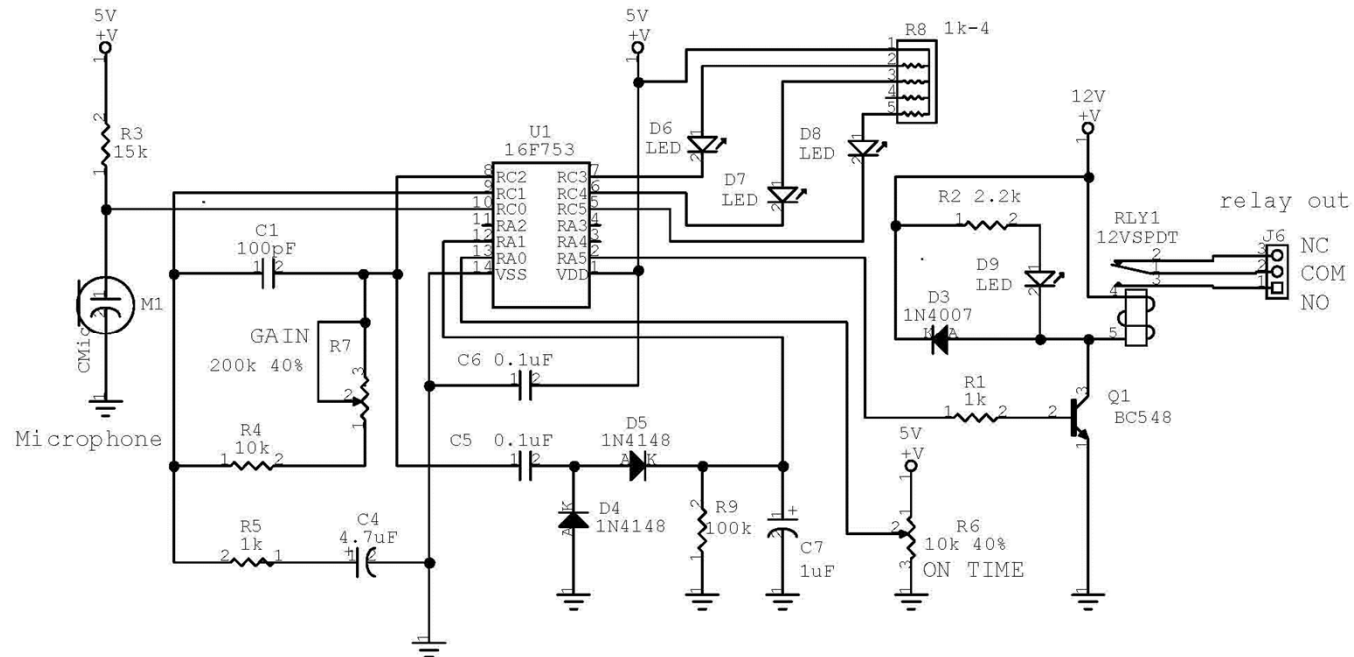
- オペアンプが1個内蔵された16F753(14ピン)をマイクアンプとして使用したレベル計とリレー付き音センサー装置を作りました。
- オペアンプでコンデンサーマイク出力を10倍から2000倍まで増幅します。
- その出力を半波整流し、平滑し、その電圧(音量に比例した電圧となる)を内蔵ADコンバータで検出します。
- 音量(その電圧)を3つのLEDで表示し、設定したスレッショルドを越えるとリレーを駆動します。
- リレーを駆動する時間はボリューム(内蔵ADコンバータで値を電圧で検出)で設定します。

回路の説明(ブロック図)



オペアンプでコンデンサーマイク出力を10倍から2000倍まで増幅します。
その出力を半波整流し、平滑し、その電圧(音量に比例した電圧となる)を内蔵ADコンバータで検出します。
音量(その電圧)を3つのLEDで表示し、設定したスレッシュホールドを越えるとリレーを駆動します。
リレーを駆動する時間はボリューム(内蔵ADコンバータで値を電圧で検出)で設定します。

回路の説明(詳細)

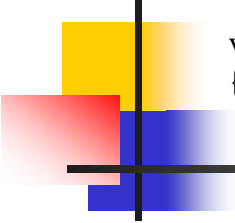


オペアンプでコンデンサーマイク出力を10倍から2000倍まで増幅します。

その出力を半波整流し、平滑し、その電圧(音量に比例した電圧となる)を内蔵ADコンバータで検出します。

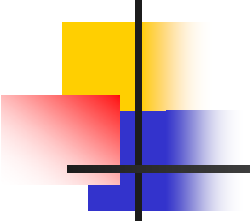
音量(その電圧)を3つのLEDで表示し、設定したスレッシュドを越えるとリレーを駆動します。リレーを駆動する時間はボリューム(内蔵ADコンバータで値を電圧で検出)で設定します。

プログラムの説明1



```
void initializing()
{
    //
    setup_oscillator(OSC_4MHZ);
    //
    set_tris_a(Amode);      //
    set_tris_c(Cmode);      //
    //
    setup_comparator(NC_NC_NC_NC);
    //
    // A/D converter initialize
    setup_adc(ADC_CLOCK_DIV_8);      // 2usec
    setup_adc_ports(sAN0 | VSS_VDD); // PIN_A0
    //
    setup_opamp1(OPAMP_ENABLED | OPAMP_I_TO_OPAIN | OPAMP_NI_TO_OPAIN);
    //
    // timer 1 initialize for hysteresis count down
    setup_timer_1(T1_INTERNAL|T1_DIV_BY_8); //8us
    //
    set_timer1(0xE797);      // timer 1 setting almost 50msec
    enable_interrupts(INT_TIMER1); // timer 1 enable
    enable_interrupts(GLOBAL); // global enable
    //
    LED1_OFF;
    LED2_OFF;
    LED3_OFF;
    //
    delay_ms(100);
}
```

プログラムの説明2



```
void main()
{
    initializing();
    // main loop
    while(1)
    {
        get_sound();
        drive_led();
        //
        if(value_sound > vox_mid) // around 1.8V
        {
            //
            disable_interrupts(GLOBAL); // global enable
            get_time();
            count1 = value_time; // 2 to 255
            count1 = count1 * 4; // 8 to 1024, max around 50 sec
            enable_interrupts(GLOBAL); // global enable
        }
        if(count1 > 0) RELAY_ON;
        else RELAY_OFF;
        //
    }
}
```

実際の動作

