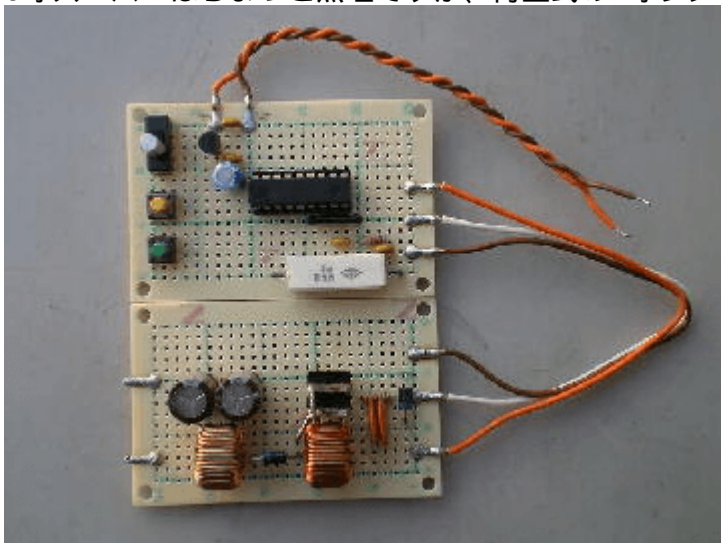


昇圧型B電源ユニット(PIC16F819)

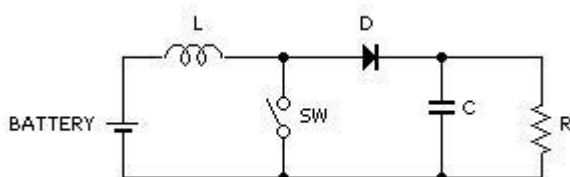
概要

私はよくPICやオペアンプやトランジスタや真空管等をハイブリッド [hybrid] 的に使うことが多いです。そんな時に、真空管を使って何かを作ると、どうしても高価で重たい電源トランスが必要となるのですが、そこで、これが何とかならないものかといろいろと思案していました。

理想的には、ヒーター電源(普通は6.3V)くらいの電圧のみでラジオが動作してくれば良いのになあ...そこで今回は、DC6.3V(1A以内)のみを供給して、真空管のラジオを動作可能な電源に挑戦してみました。5球スーパーはちょっと無理ですが、再生式の1球ラジオくらいなら何とか出来そうです。



動作原理

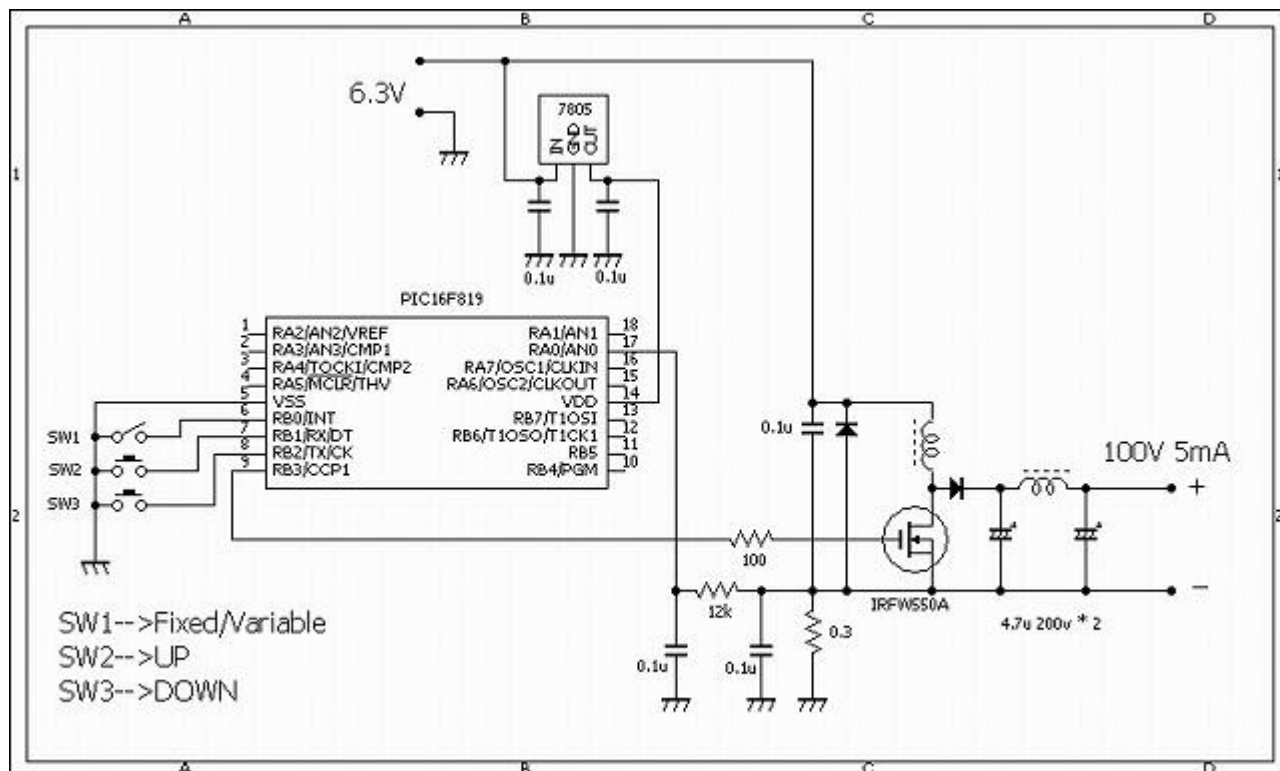


基本的な回路で説明します。

- 最初にスイッチ(SW)がON状態になるとコイル(L)にエネルギーが蓄えられます。
- このとき、コイル(L)の入力側はプラス、出力側はマイナスの電位になります。
- 次にスイッチ(SW)がOFF状態になるとコイル(L)は電流を流し続けようとして(レンツの法則)、蓄えたエネルギーを放出します。
- この時コイル(L)の出力側がプラス、入力側がマイナスになります。
- スイッチ(SW)はOFFなので電流はダイオード(D)を通してコンデンサ(C)および負荷(R)に流れます。

レンツの法則 電磁誘導の起電力の向きに関する法則です。「誘導起電力はそれを生ずる原因をさまたげる向きに生じるという法則」1834年にドイツのレンツが発見しました。

回路図



コイル(L)にはTOKINのSN8D-500を使用しました。FETにはFAIRCHILDセミコンダクタのIRFW540A(秋月電子で4個200円でした)を2個並列にして使用しました。

処理説明

- PWM方式によってFETをON/OFFさせます。
- PWMの周波数は約10Khzとし、デューティは1024ステップで設定します。
- 制御には、次の2つのモードを持たせました。
 - 【固定モード】
 - コイルに流れる電流値を400mA固定にする。
 - 抵抗が0.3ΩADコンバータの精度が約5mVなので0.3×400mA=120mV
 - 120mV÷5mV=24
 - つまりAD変換した値が24になるように制御します。
 - 【可変モード】
 - UP/DOWNのスイッチによってデューティの幅を増減させる。
- 尚、モードの切り替えは回路図のSW1によって行います。

ソースコード

[dc2dcConv.c](#)

```

//*****
*
/*
  
```

```

□DC→DCコンバーター>
*/

//*****
*

void interrupt()
{
    if (INTCON.TMR0IF == 1) { // 約33msecの周期
        INTCON.TMR0IF = 0;
        PORTB.F4 = ~PORTB.F4;
    }
    if (PIR1.TMR1IF == 1) { // 約250msecの周期
        PIR1.TMR1IF = 0;
        PORTB.F5 = ~PORTB.F5;
    }
}

//*****
*

void Pwm_Change_DutyEx(unsigned int duty_ratio)
{
    CCP1L = duty_ratio >> 2;
    CCP1CON.F6 = duty_ratio & 0b00000001;
    CCP1CON.F7 = (duty_ratio & 0b00000010) >> 1;
}

//*****
*

void main()
{
    static unsigned int ad0;
    static unsigned int duty_ratio;
    //
    ADCON0 = 0b01000000;
    ADCON1 = 0b11001110; // AN0を使用する。
    TRISA = 0b00001011;
    TRISB = 0b00000111; // CCP1はRB3を使用する□LED1はRB4□LED2はRB5を使用する。
    OSCCON = 0b01110000; // クロックは内蔵の8Mhzを使用する。
    OPTION_REG = 0b10000111;
    PIE1.TMR1IE = 1;
    PIR1.TMR1IF = 0;
    T1CON = 0b00110001;
    T2CON = 0b00000100;
    INTCON = 0b01100000;
    //
    Pwm_Init(10000); // 10Khz
    Pwm_Change_DutyEx(1024 / 2);
}

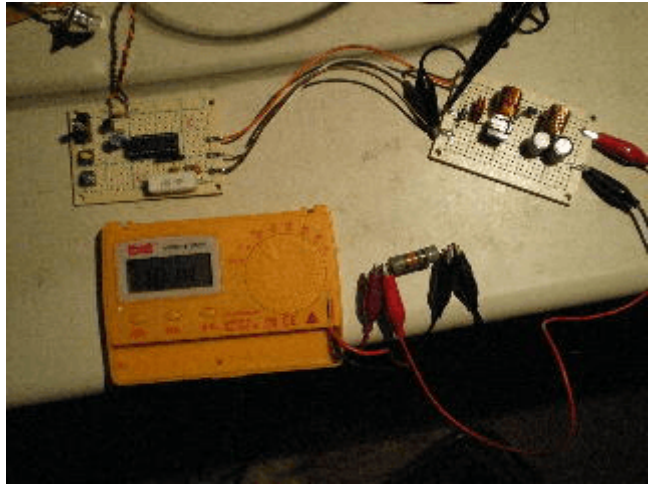
```

```
PR2 = 0xFF;
Pwm_Start();
//
TRISB = 0b00000111; // 何故か再設定が必要！
duty_ratio = 0;
//
INTCON.GIE = 1; // これ以降の処理で割り込みを許可する。
//
while (1) {
    if (PORTB.F0 == 0) { // ホールド状態にする。
        if (PORTB.F1 == 0) // 電流値を上げる。
            duty_ratio++;
        if (PORTB.F2 == 0) // 電流値を下げる。
            duty_ratio--;
        Pwm_Change_DutyEx(duty_ratio);
        Delay_ms(10);
        continue;
    }
    ad0 = Adc_Read(0); // 電流値を読み込む。
    if (ad0 < 24) { // 電流を400mAに設定[Vcc=5V][0.3Ω]AD
分解能=5mV
        if (duty_ratio < 1024) // 電流値を上げる。
            duty_ratio++;
    } else {
        if (duty_ratio > 0) // 電流値を下げる。
            duty_ratio--;
    }
    Pwm_Change_DutyEx(duty_ratio);
    Delay_ms(10);
}
}

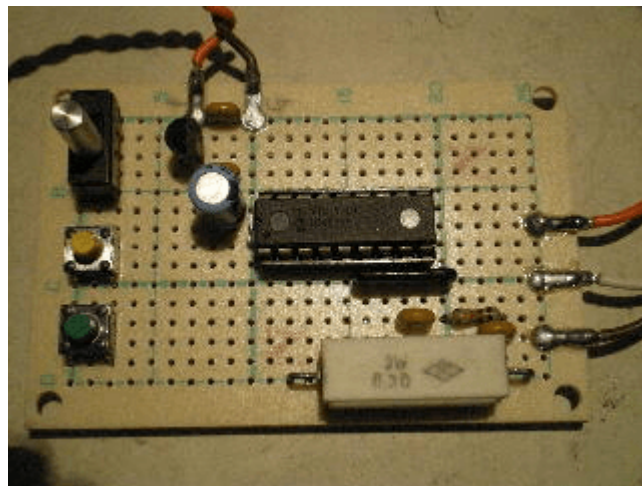
//*****
*
```

動作確認

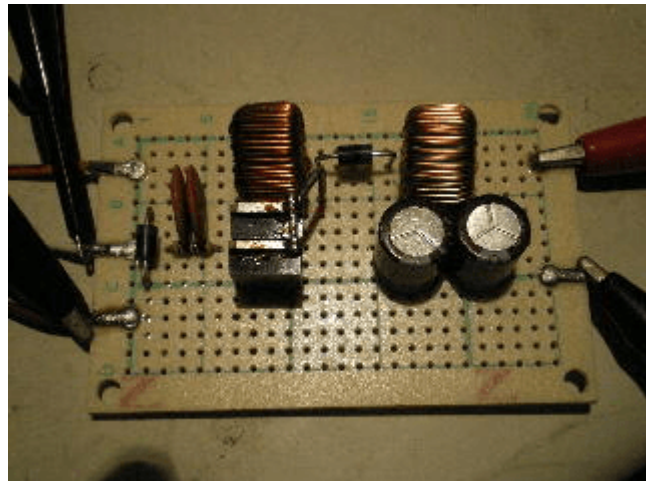
以下の画像は、本ユニットを利用して6.3Vの電圧を昇圧し、負荷をかけている所です。出力電圧は、



約100Vで18k Ω の負荷(5.5mA)をかけています。



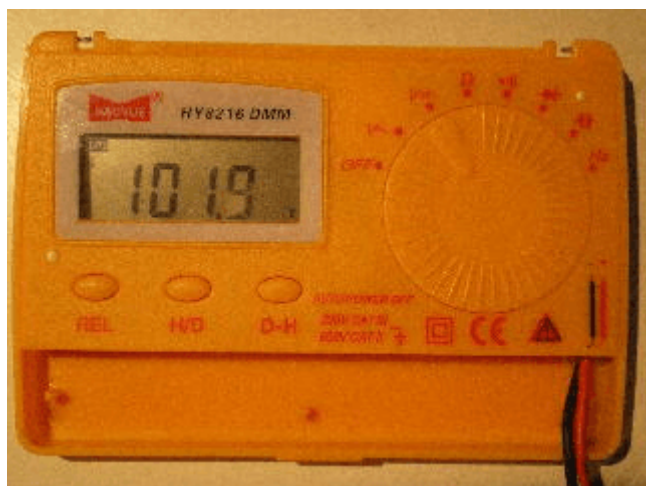
制御部です。0.3 Ω は電流検出用です。



昇圧部です[FETは2個並列にして使いました。

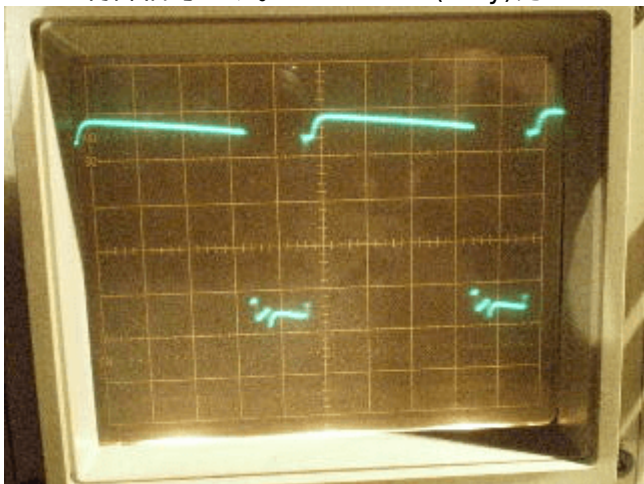


負荷をかけます。



出力電圧は約100Vです。

FETの制御信号です。デューティ(duty)比によって出力される電圧を制御できます。



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://www.deepsky.jp/wiki/> - うごくといいな

Permanent link:

<http://www.deepsky.jp/wiki/doku.php?id=elechobby:picdic:otherpic:158>

Last update: **2025/10/17 14:29**

