

第七章. DC-DC 降压模块

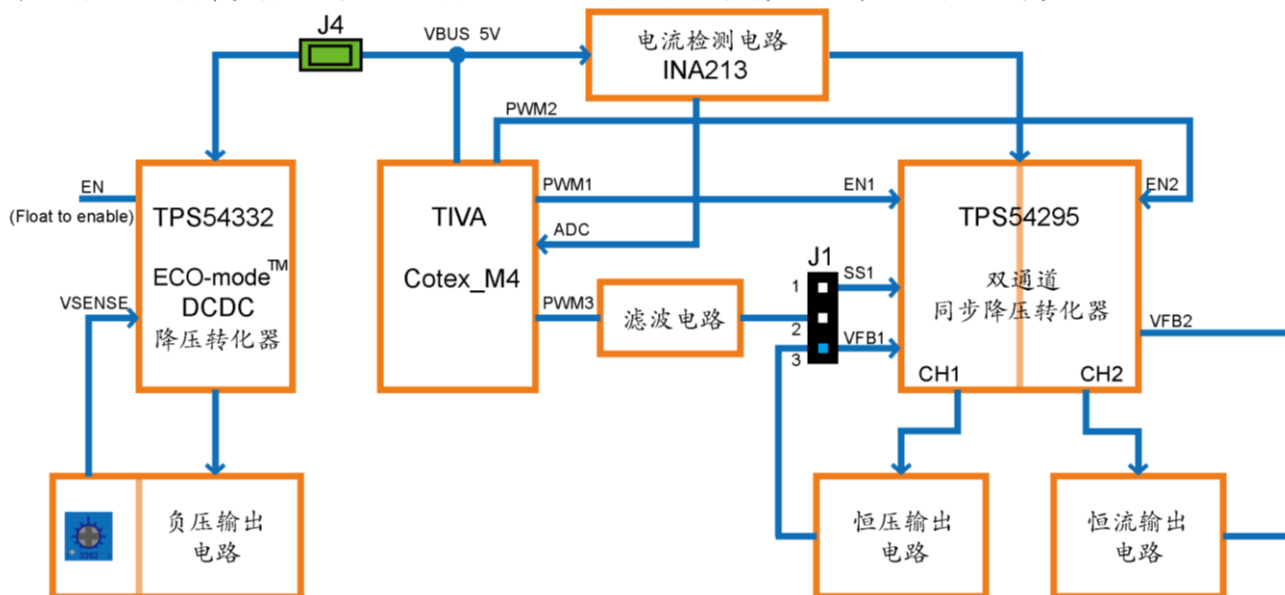
DC-DC Buck 实验模块介绍

实验目的:

- 1、理解并掌握buck; buck-boost电源拓扑。
- 2、理解并掌握输出电压的调节方式。
- 3、理解同步与非同步电路的差异。
- 4、LED恒流驱动的原理。
- 5、理解并掌握各类电源参数测量方法: 效率, 纹波, 软启动时间, 电感电压与电流的变化。
- 6、理解电源设计的PCB布线的基本原则。

实验简介:

本模块采用了一片双通道同步降压芯片TPS54295和一片单通道非同步降压芯片TPS54332实现buck拓扑的恒压、恒流和负压生成, 并提供多种输出调节模式: 1、PWM调节使能端; 2、PWM滤波后调节反馈端; 3、PWM滤波后调节软启动(SS)端; 4、调节反馈电阻。其中, 负压的产生采用了buck拓扑到 拓扑的转变, 为了评估电源的效率, 加入电流检测电路。



DC-DC Buck 模块布局及电路板跳线物理位置展示

■ PCB布局

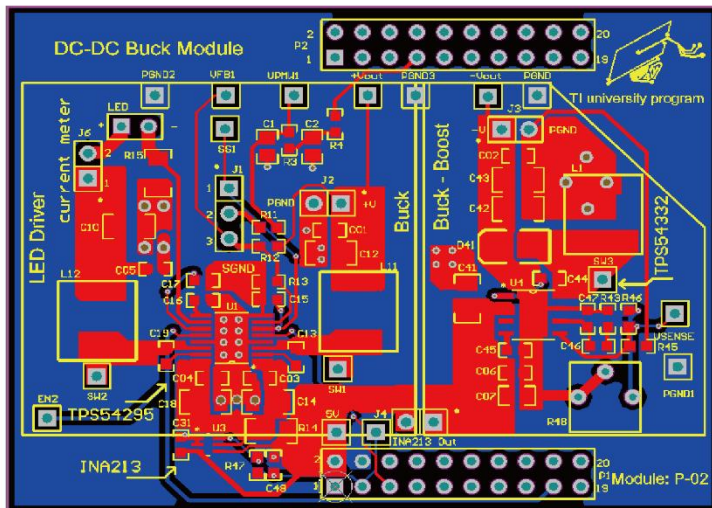


图6-1 PCB布局图

■ 实物图

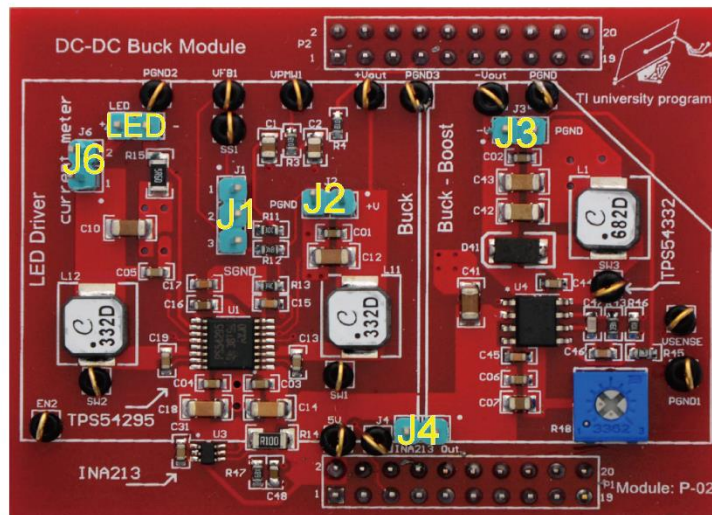
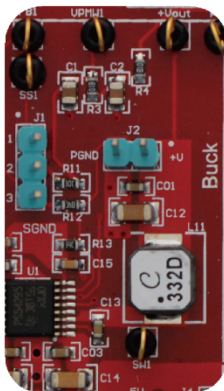


图6-2 实物图

PCB布局布线介绍

- 1、使用双路降压模块，一路恒流输出，一路恒压输出，理解恒压恒流电路的异同。
- 2、突出利用SS，FB和EN来调整输出电压/电流的方法，理解电源内部的反馈，软启动等特点。
- 3、使用Buck电源来实现Buck-Boost反向输出，帮助学生理解Buck和Buck-Boost的异同。
- 4、通过把同步和非同步电源放在一个板卡上，理解同步和非同步，并可测试关键结点的波形。

恒压源输出

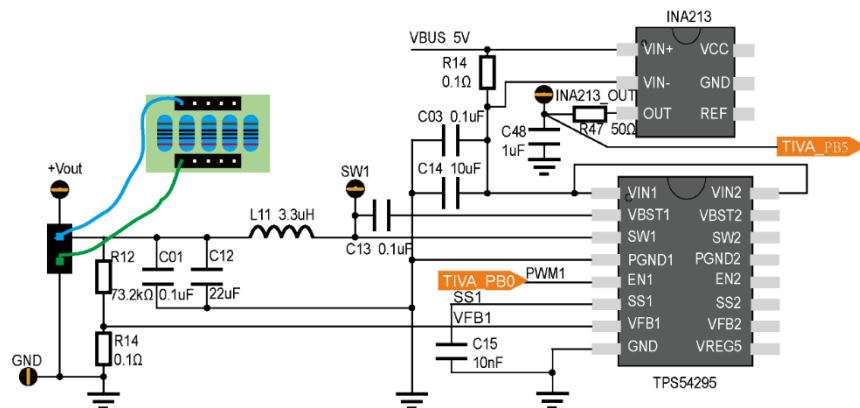


实物图

恒压源输出是开关电源能输出稳定的电压，且电压不会随着负载的变化而变化。该电路实现+5V输入，+3.3V输出的功能。如下为原理图，左侧为实物图。需要注意的是在恒流控制点亮LED时，负载是1W的LED灯珠。但是，在恒压源输出时负载不能仅仅是LED灯珠，这是因为二极管LED会使输出的电压发生钳位，导致电源芯片偏离了正常工作的轨道。因此，用功率电阻作为负载进行试验，S测试恒压源的输出和调节恒压输出的大小。

恒压输出的实现是：在反馈端FB和地之间接入电阻R14，则R14两端的电压被控制在一个设定值。由于FB端是运放的反相输入端，故其输入/输出电流几乎为零，这样，流过R14的电流就可以计算出来。在R14与L11之间接入电阻R12，把电压抬高。因为FB端的电压恒定，流过R12和R14的电流就恒定，所以，在R12和R14两端产生电压也恒定，从而实现了恒压源的输出。

电流检测电路：此电路中采用高测电流测量。INA213正好是共模高，失调电压小，温度漂移特性好的运算放大器。



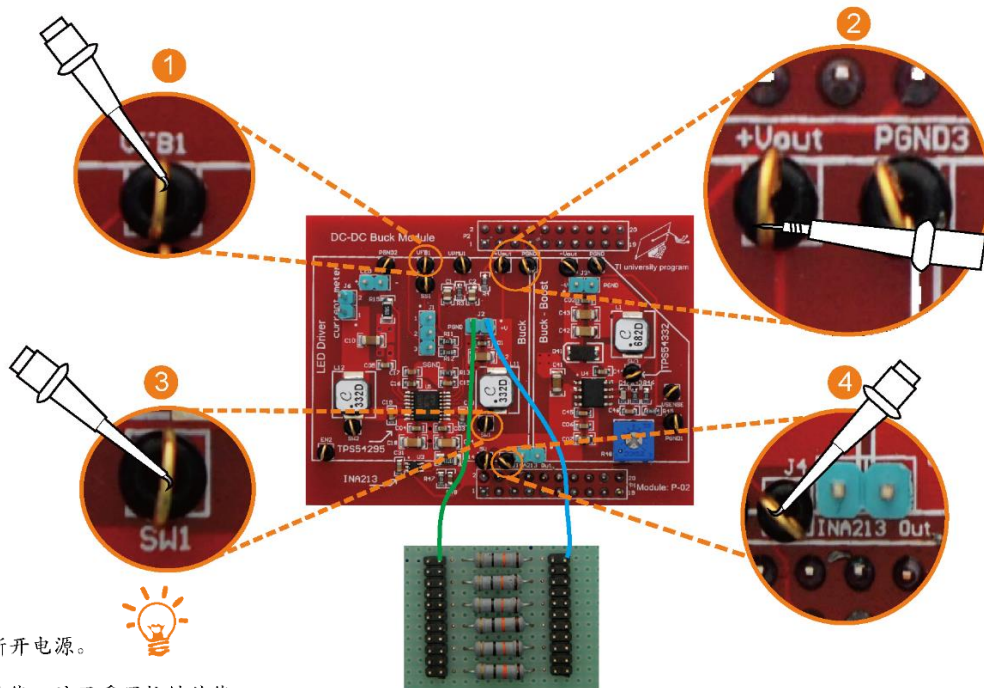
恒压测试实验

- 1、理解原理图，编写Launchpad代码（参考代码见网上资源）。
- 2、在母板上连接TIVA、LCD模块、DC-DC Buck(降压)模块，准备实验。
- 3、连接负载，如右图所示。注意负载功率和阻值大小。
- 4、在降压模块上完成恒压测试的仪表连接，如右图所示。其中测①观测反馈电压值；测②观测输出电压，通过调节示波器可以测量得到电压值及纹波情况；测③观测开关节点的波形；测④观测输入电流检测电路输出的电压值。
- 5、给TIVA上电，按LCD模块中按键S1（按键S1是通道1是否打开的切换键），将TPS54295的通道1使能，进入工作状态，在LCD上可以看到输入电流的大小。
- 6、断电后，改变负载，通过示波器观察电路恒压的情况，及纹波的变化。

注意

- 连接仪表及跳线时，断开电源。
- 电路进入工作前需要使能，这里采用按键使能。请勿忘记。
- 实验套件中没有提供电阻负载，实验中的负载需要自己制作。这里的负载是阻值13 Ω 、功率1W的电阻。通过电阻的串并联改变负载大小。

恒压测试主要步骤



TPS54295 电压调节方式介绍

DC-DC Buck模块提供了多种电压调节的方式。这里主要以TPS54295为中心，介绍三种电压调节方式，分别是使能端调节、软启动端调节及反馈端调节。

如右下图所示，为TPS54295部分内部简单原理图。图中主要指出该芯片内部5个重要的逻辑结构，即使能逻辑结构、软启动逻辑结构、比较器、控制结构及MOSFET。

使能端调节：

使能逻辑和反馈端（VFB）是一个线与逻辑，当EN输入为高电平，则整个芯片在电路连接完整的情况下是可以正常工作的，相反当EN输入为低电平，则反馈端电压被拉低，整个芯片不工作。

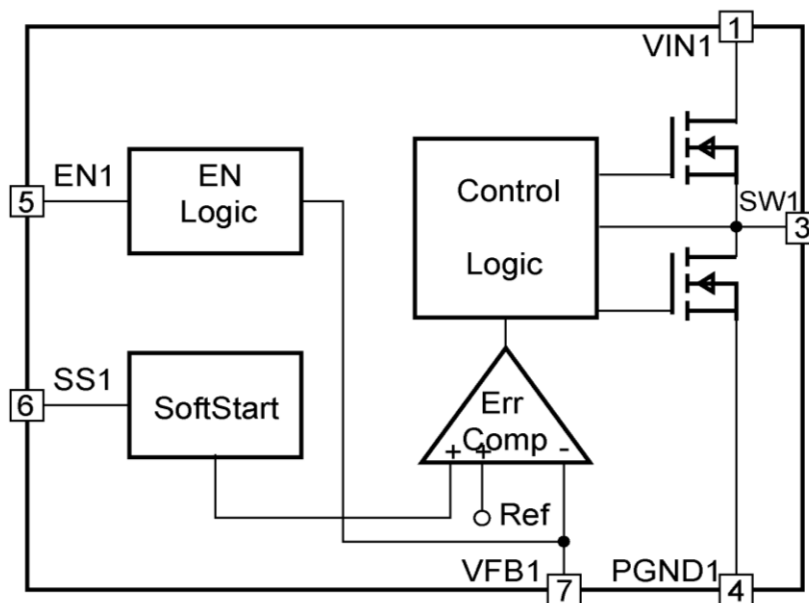
软启动调节：

电压由零慢慢提升到额定电压，使芯片能平滑地开启工作状态，不产生大的电流冲击。这就是软启动。

在TPS54295中，当芯片处于正常工作状态时，软启动处的电容呈现电压充满状态，这时对反馈端不产生影响。但当软启动处的电容充电不完全时，且电容的电压低于反馈电压，软启动的电压就会影响反馈端电压，使输出调整。

反馈端调节：

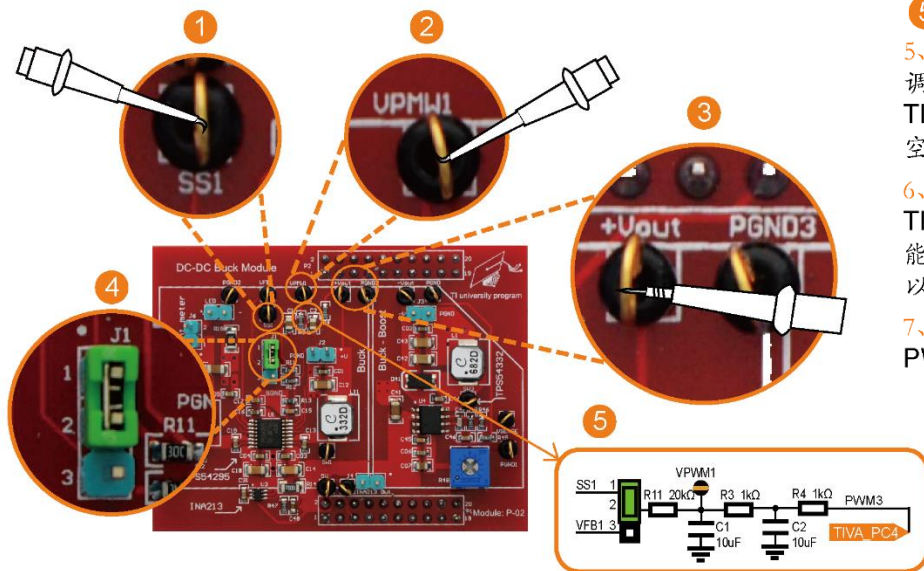
反馈端电压的调节是比较直接的。芯片会通过比较器输出的正负和其自身的逻辑控制来有效地调节开关的占空比，使输出达到稳定状态。



TPS54295内部简单原理图

电压调节测试主要步骤

——SS控制电压输出



SS控制电压输出测试实验

- 1、理解原理图，编写Launchpad代码（参考代码见网上资源）。
- 2、在母板上连接TIVA、LCD模块、DC-DC Buck(降压)模块，准备实验。
- 3、在降压模块上完成电压控制测试，如左图所示为SS控制实现电压调节。
- 4、SS控制电压调节的测试点连接，测①观测SS1的电压值；测②观测PWM3滤波后的电压值，测③观测输出电压的电压值和纹波情况；⑤是滤波电路。
- 5、给TIVA上电，观察VPWM1的值通过滚轮调节使VPWM1的最小（或者将表笔连接在TIVA的PWM3输出端口上，调节滚轮，使占空比最小。）。
- 6、关闭TIVA,如④所示短接J1的1,2。再次给TIVA上电，按一下LCD模块上的按键S1，使能TPS54295的EN1端口，缓慢调节滚轮，可以看到输出电压随滚轮转动发生变化。
- 7、记录PWM3的占空比和输出电压值，了解PWM3和输出电压之间的控制关系。

注意

- 连接仪表及跳线时，断开电源。
- 输出电压测量过程中，测纹波需要将示波器切换到交流模式，而测电压值则为直流模式。
- 如果了解电源的性能，可以接负载进行测试。负载可以选择阻值13Ω功率1W电阻，通过串并联实现负载变化。

VFB电压调节测试实验

- 1、理解原理图，编写Launchpad代码（参考代码见网上资源）。
- 2、在母板上连接TIVA、LCD模块、DC-DC Buck(降压)模块，准备实验。
- 3、在降压模块上完成电压控制测试，如右图所示为VFB控制实现电压调节。
- 4、如右图所示，完成跳线连接，即5中显示，短接J1的2,3。
- 5、在降压模块上，完成仪表连接，如图所示，测 2 观测反馈电压值，测 3 观测PWM3滤波后的电压值，测4观测输出电压情况。1 是滤波电路的原理图。
- 6、给TIVA上电，按LCD模块上的按键S1，使能TPS54295的EN1端口，拨动滚轮，记录TIVA输出的PWM3的占空比和输出电压值。了解PWM3可电压之间的控制关系。

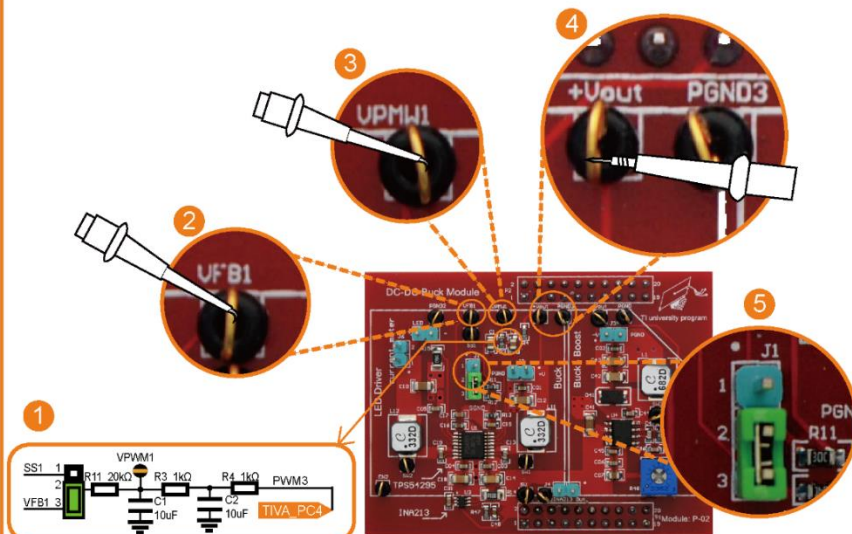
注意

- 连接仪表及跳线时，断开电源。
- 输出电压测量过程中，测纹波需要将示波器切换到交流模式，而测电压值则为直流模式。
- 如果了解电源的性能，可以接负载进行测试。负载可以选择阻值13Ω功率1W的电阻，通过串并联实现负载变化。



电压调节测试主要步骤

——VFB控制电压输出



TPS54295 提供两路输出：一路正压输出，一路高亮 LED 驱动。两个功能由 Tiva M4 分别控制。正压输出：通过 PC4 产生一路 PWM 波，PWM 波输入至 J1 跳线帽处，可跳过掉线将 PWM 波连接至 TPS54295 的 SS1 端或者 VFB1 端；通过 PB0 控制 TPS54295 的 EN1 端，PB0 配置成 GPIO 功能，其输出电平控制 EN1 端的使能或者禁能；PB0 输出电平状态由 LCD 开发板上的按键 S2 控制。高亮 LED 驱动：通过 PA6 产生一路 PWM 波控制 TPS54295 的 EN2 端，用于驱动 LED。这两路 PWM 波的占空比可以通过 LCD 开发板上的滚轮控制，同时 LCD 会显示当前测量点的电流值。

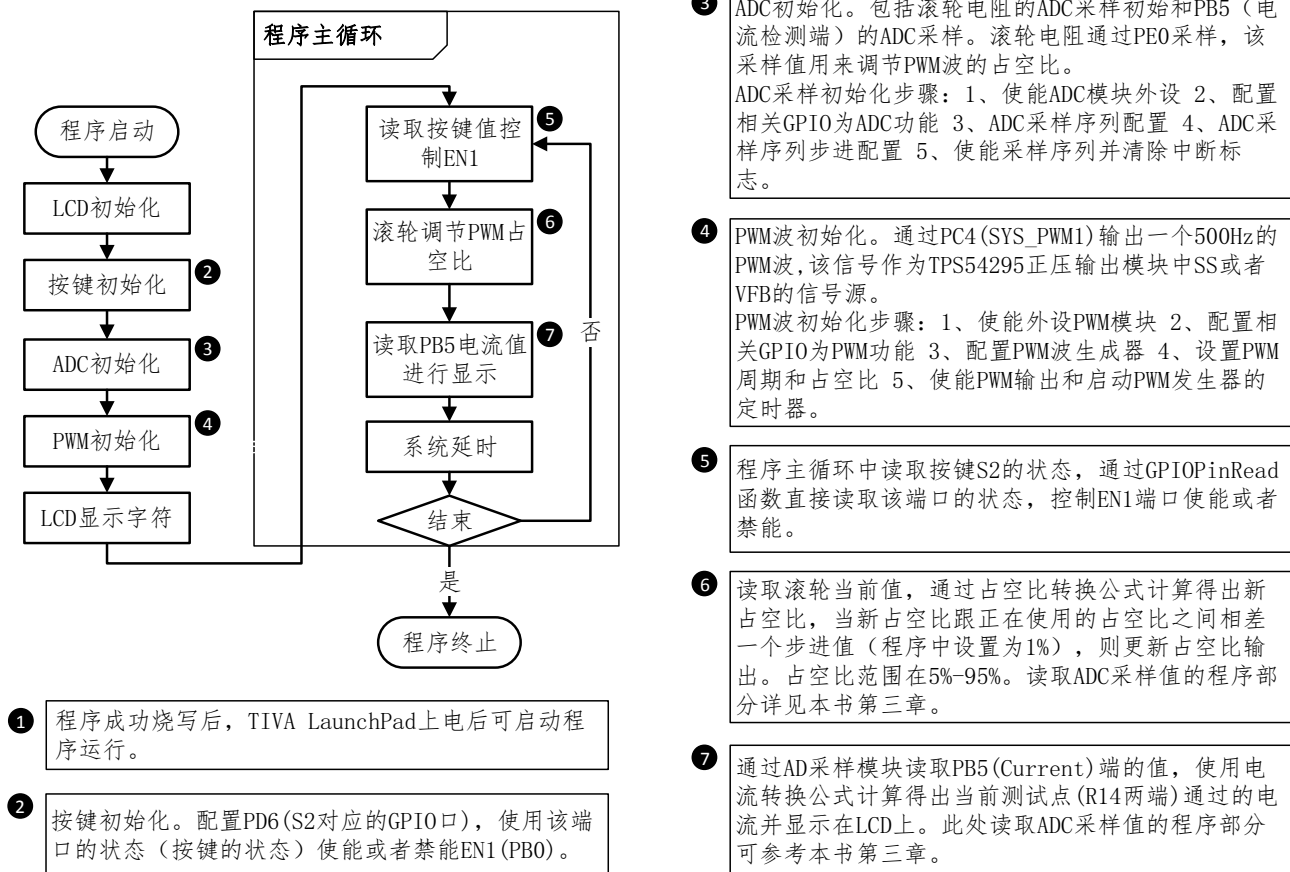


图 xx 程序流程图

此次实验程序涉及 LCD 显示，ADC 采样，按键控制以及 PWM 波功能。LCD 显示和 ADC 采样可参考本书第三章。

PWM 波功能实现

PWM 波实现中，其输出信息的周期是需要计算的，该计算公式在第六章中已经给出，本章将直接使用该公式，不在详细说明。其计算公式如下：

$$period = \frac{SysPWM_Fre}{OutputPWM_Fre}$$

程序中的时钟节拍值 PERIOD_TIME=25000，其表示的是 500Hz（系统主频 12.5Mhz，PWM 时钟频率配置为 SYSCTL_PWMDIV_1，也就是 1 分频），可有上述公式直接计算得出。

$$period = \frac{SysPWM_Fre}{OutputPWM_Fre} = \frac{12.5 \times 10^6 / 1}{500} = 25000$$

程序代码如下

```
/******  
* @berif 初始化PWM  
* @param none  
* @return none  
*  
* _____  
* |  
* M4 PC4|-->M0PWM6(SYS_PWM1)  
* _____| 25000 //500Hz  
void Init_PWM()  
{
```

```
//设置PWM时钟, 1分频
SysCtlPWMClockSet(SYSCTL_PWMDIV_1);

//使能外设PWM0模块
SysCtlPeripheralEnable(SYSCTL_PERIPH_PWM0);

//使能PWM0使用的外设GPIOC
SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOC);

//配置PC4复用功能是PWM
GPIOPinConfigure(GPIO_PC4_M0PWM6);

//配置PC4为PWM0模块使用
GPIOPinTypePWM(GPIO_PORTC_BASE, GPIO_PIN_4);

//配置PWM1模块, PWM1生成器1, 向下计数并且立即更新参数
PWMGenConfigure(PWM1_BASE, PWM_GEN_1, PWM_GEN_MODE_UP_DOWN | PWM_GEN_MODE_NO_SYNC);

//设置PA6 (M1PWM1) 产生的PWM周期
PWMGenPeriodSet(PWM1_BASE, PWM_GEN_1, PERIOD_TIME);

//设置PC4 (M0PWM6) 的占空比25%
PWMPulseWidthSet(PWM0_BASE, PWM_OUT_6, PERIOD_TIME / 4);

//使能输出
PWMOutputState(PWM0_BASE, PWM_OUT_6_BIT, true);

//启动PWM0发生器3的定时器
PWMGenEnable(PWM0_BASE, PWM_GEN_3);

//使能发生器模块计数器同步
PWMSyncTimeBase(PWM0_BASE, PWM_GEN_3);

}
```

PWM 输出信号的占空比调节部分程序代码如下

```
//根据滚轮采样值计算占空比
uint32_t cur_Duty = 5 + (90 * pui32ADCWheel_Value) / ADCMAX;
uint32_t period = cur_Duty * PERIOD_TIME / 100;

//改变PWM波的占空比
ROM_PWMGenPeriodSet(PWM0_BASE, PWM_GEN_3, PERIOD_TIME);
ROM_PWMPulseWidthSet(PWM0_BASE, PWM_OUT_6, period);
```

占空比计算公式在第六章已给出。

按键控制功能

按键使用 LCD 开发板上的 S2 完成，该按键连接至 PD6。按键初始化代码如下：

```
* @brief 对端口C、D进行按键初始化
* @param none
* @return none
*
* _____
* |
* TIVA PD6|<--Button2
* _____|
*****/

void Init_Key()
{
    //初始化外设GPIO
    ROM_SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOD);
```



```
// 设置PD为2MA, 弱上拉输出
ROM_GPIOPadConfigSet(GPIO_PORTD_BASE, GPIO_PIN_6,
                      GPIO_STRENGTH_2MA, GPIO_PIN_TYPE_STD_WPU);

//设置GPIO输入模式
ROM_GPIODirModeSet(GPIO_PORTD_BASE, GPIO_PIN_6, GPIO_DIR_MODE_IN);

}
```

对于按键状态读取采用键扫描完成, 程序代码如下:

```
// 监控PortD端口的变化
uint32_t readData = GPIOPinRead(GPIO_PORTD_BASE, GPIO_PIN_6);
```

按键控制的是 TPS54295 的 EN1 端, 过在读取了按键状态后就可控制 EN1 端的状态。控制程序代码如下:

```
// 响应按键2 (S2)
if(!(readData & GPIO_PIN_6))
{
    if(status1 == 0)
        status1 = 1;
    else
        status1 = 0;
    if(status1)
    {
        // PB0置高 (使能EN_1)
        GPIOPinTypeGPIOOutput(GPIO_PORTB_BASE, GPIO_PIN_0);
        GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_0, GPIO_PIN_0);
    }
}
```

```
}  
else  
{  
    //PB0置低（禁能EN_1）  
    GPIOPinTypeGPIOOutput(GPIO_PORTB_BASE, GPIO_PIN_0);  
    GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_0, 0);  
}  
}
```

实验的程序设计中

实验的程序设计中需要在 LCD 上显示检测点的电流值。电流值是采用 ADC 采样获得，ADC 采样配置以及获取采样值跟本书第三章中的滚轮控制部分相似，故该部分可参考第三章完成。获取 ADC 采样值后需要根据实际电路计算得出实际值。程序通过 PB5 口采样 INA213 的输出端获取采样值，而 LCD 上显示的是 INA213 输入端流经的电流值，即流经 R_{14} 的电流值。INA213 具有 50 倍增益，计算公式如下：

$$Current = \frac{Cur_ADC \times ADCRef}{ADCMax \times 50 \times R_{14}} = \frac{Cur_ADC \times 3.3}{4096 \times 50 \times 0.1}$$

程序中将该值扩大 1000 倍使用，故该公式简化为：

$$Current = \frac{Cur_ADC \times 3.3 \times 1000}{4096 \times 50 \times 0.1} = \frac{Cur_ADC \times 660}{4096} mA、$$

程序代码如下：

```
unsigned long Current_Sample = (pui32ADC_CurrentValue * 660) / 4096;
```

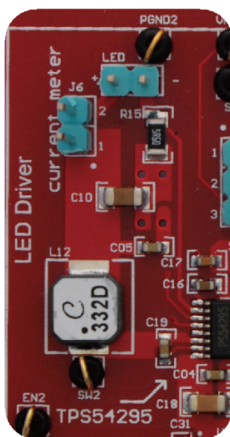
pui32ADC_CurrentValue 是 PB5 端口的 ADC 采样值。该值获取可参考本书第三章滚轮值的获取部分。

A. 恒流高亮 LED 驱动

实验内容

以单个大功率（1W）LED 为负载，通过调节恒流输出的电流大小，观察 LED 的亮度的变化和恒流电流的大小

恒流高亮LED驱动

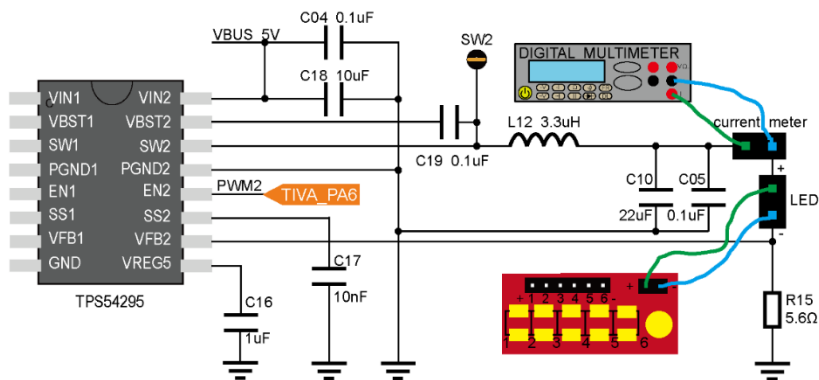


实物图

利用TPS54295第二通道搭建的电路是buck型开关电源电路，原理图如下图所示，实物图如左图所示。开关电源芯片TPS54295内部有MOSFET开关器件，它的开关频率达到700kHz。在开关导通时，电源给电感L12和电容C10充电；当开关关断时，相当于将L12的一端接地，L12释放能量。和buck型电路工作原理一样，通过电感L12和电容C10充放电，实现电压的转换。同时，该电路在FB端形成闭环通路，可以实现电压的稳定输出。

电流调节方式：

在本实验中通过PWM控制EN2起到平均电流的调节。由于使用LED做实验，一定的频闪对于肉眼来说是不可见的。故可以通过调节EN2来调整LED的亮度。



恒流高亮LED驱动 测试实验

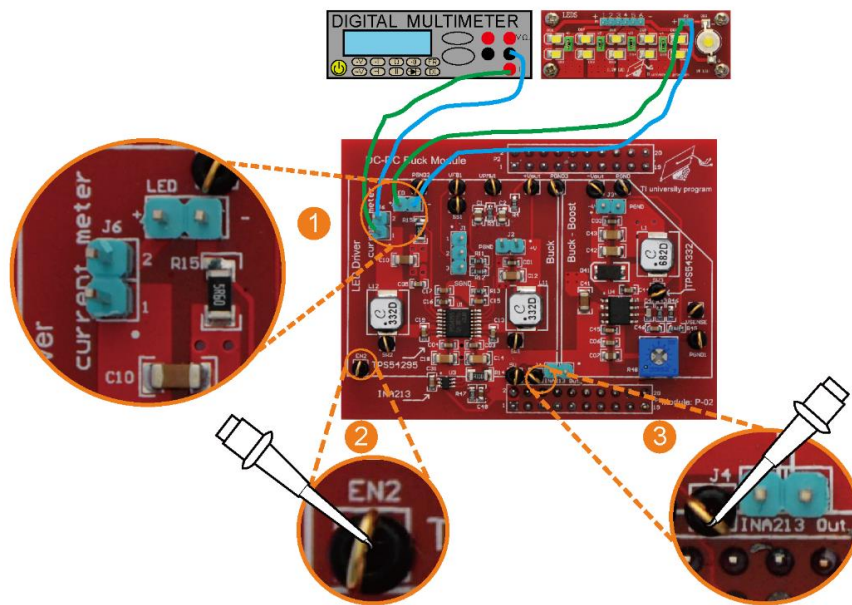
- 1、理解原理图，编写Launchpad代码（参考代码见网上资源）。
- 2、在母板上连接TIVA、LCD模块、DC-DC Buck (降压)模块，准备实验。
- 3、在降压模块上完成恒流高亮LED驱动测试的仪表连接，如右图所示。其中①中一个是电流测量的接口，J6的1接红表笔，J6的2接黑表笔；另一个的大功率LED的接口，连接时注意正负极。测②观测TPS54295的EN2端口处波形。测③观测输入电流检测电路输出的电压值。
- 4、连接负载，如右图所示。此处使用的是试验箱中的LED板，连接在电路中的是1W的LED，连接过程中注意正负极。
- 5、给TIVA上电，可以观察到LED点亮，拨动滚轮，可以看到LED的亮度发生变化，此处是通过控制EN2端实现调节作用，过程中可以记录EN2处波形的占空比和电流表显示的电流值，了解他们之间的控制关系。

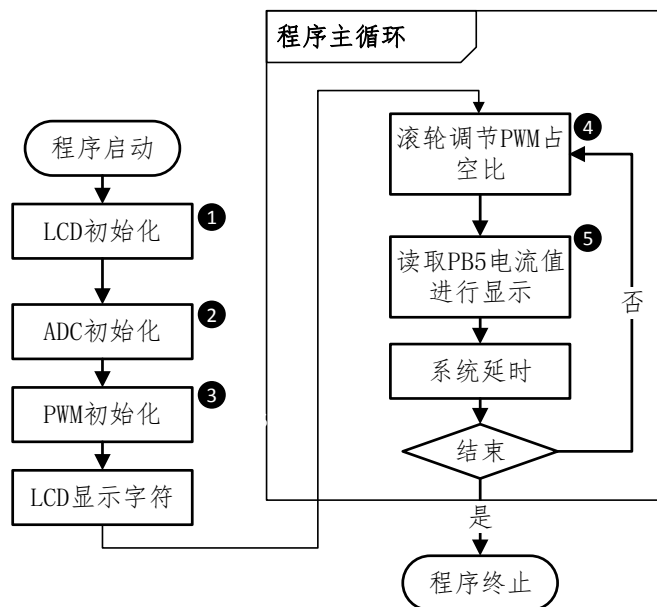
注意

- 连接仪表及跳线时，断开电源。
- 负载和电流表连接过程中注意正负极。



恒流高亮LED驱动测试主要步骤





① LCD初始化包括LCD端口使能、SSI通信协议配置、LCD配置初始化、LCD清屏4个步骤。每个步骤详情请见本书第三章。

② ADC初始化。包括滚轮电阻的ADC采样初始和PB5（电流检测端）的ADC采样。滚轮电阻通过PE0采样，该采样值用来调节PWM波的占空比。PB5采样得到的是当前电源的电流。

ADC采样初始化步骤：1、使能ADC模块外设 2、配置相关GPIO为ADC功能 3、ADC采样序列配置 4、ADC采样序列步进配置 5、使能采样序列并清除中断标志。

③ PWM波初始化。通过PA6 (EN_2) 输出一个频率为500Hz的PWM波，该PWM波输出至TPS54295中的EN2。其占空比可以通过滚轮进行调节。

PWM波初始化步骤：1、使能外设PWM模块 2、配置相关GPIO为PWM功能 3、配置PWM波生成器 4、设置PWM周期和占空比 5、使能PWM输出和启动PWM发生器的定时器。

④ 读取滚轮当前值，通过占空比转换公式计算得出新占空比，当新占空比跟正在使用的占空比之间相差一个步进值（程序中设置为1%），则更新占空比输出。占空比范围在5%-95%。通过更改占空比可以调节负载LED的亮度。滚轮当前值的读取详见本书第三章。

⑤ 通过AD采样模块读取PB5 (Current) 端的值，使用电流转换公式计算得出当前测试点 (R14两端) 通过的电流并显示在LCD上。AD采样值的读取可参考本书第三章。

图 xx 程序流程图

此次实验程序涉及 LCD 显示，ADC 采样，以及 PWM 波功能。LCD 显示和 ADC 采样可参考本书第三章。

PWM 功能实现

实验中通过 PA6 产生一路 PWM 信号直接控制 TPS54295 的 EN2 端，该实验产生的 PWM 信号跟本章实验 B 中的产生的 PWM 信号一致，唯一的不同是产生 PWM 信号的端口不同。

程序代码如下：

```
/*
*****
* @berif 初始化PWM获取两组反向的脉宽调制信号
* @param none
* @return none
*
*
*
* M4 PA6 |-->M1PWM2 (EN_2)
*
*
* *****
#define PERIOD_TIME 25000 //500Hz
//设置PWM时钟，1分频
SysCtlPWMClockSet(SYSCTL_PWMDIV_1);
//使能外设PWM1模块
SysCtlPeripheralEnable(SYSCTL_PERIPH_PWM1);
//使能PWM1使用的外设GPIOA
SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);
*/
```

```
//配置PA6复用功能为PWM
GPIOPinConfigure(GPIO_PA6_M1PWM2);

//配置PA6为PWM1模式使用
GPIOPinTypePWM(GPIO_PORTA_BASE, GPIO_PIN_6);

//配置PWM1模块, PWM1生成器1, 向下计数并且立即更新参数
PWMGenConfigure(PWM1_BASE, PWM_GEN_1,
                PWM_GEN_MODE_UP_DOWN | PWM_GEN_MODE_NO_SYNC);

//设置PA6 (M1PWM1) 产生的PWM周期
PWMGenPeriodSet(PWM1_BASE, PWM_GEN_1, PERIOD_TIME);

//设置PA6 (M1PWM1) 的占空比25%
PWMPulseWidthSet(PWM1_BASE, PWM_OUT_2, PERIOD_TIME / 4);

//使能输出
PWMOutputState(PWM1_BASE, PWM_OUT_2_BIT, true);

//启动PWM1发生器1的定时器
PWMGenEnable(PWM1_BASE, PWM_GEN_1);

//使能发生器模块计数器同步
PWMSyncTimeBase(PWM1_BASE, PWM_GEN_1);

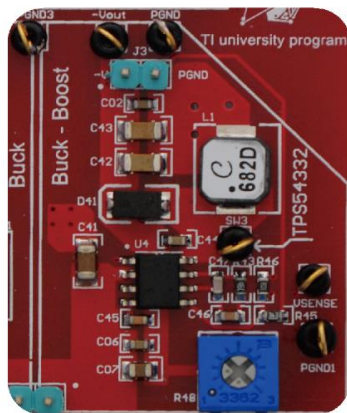
}
```

PWM 输出信号的占空比调节部分程序代码如下

```
//根据滚轮采样值计算占空比
uint32_t cur_Duty = 5 + (90 * pui32ADCWheel_Value) / ADCMAX;
uint32_t period = cur_Duty * PERIOD_TIME / 100;

//改变PWM波的占空比
ROM_PWMGenPeriodSet(PWM0_BASE, PWM_GEN_1, PERIOD_TIME);
ROM_PWMPulseWidthSet(PWM0_BASE, PWM_OUT_2, period);
```

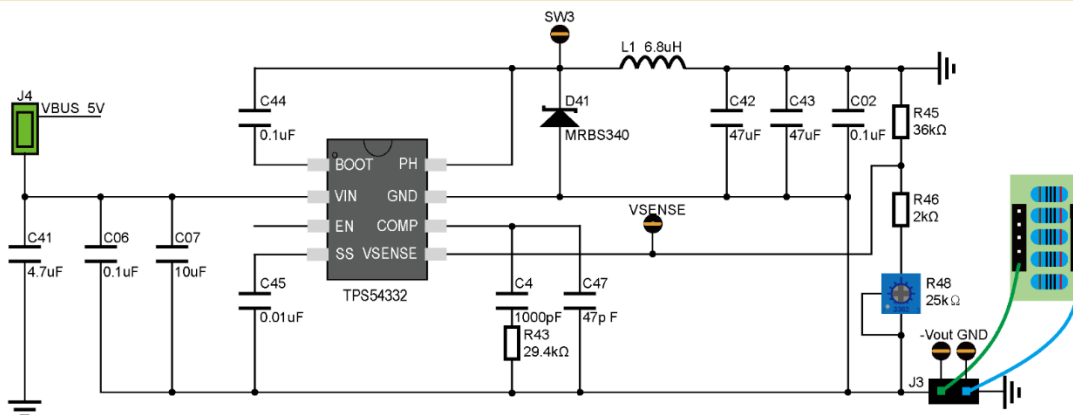
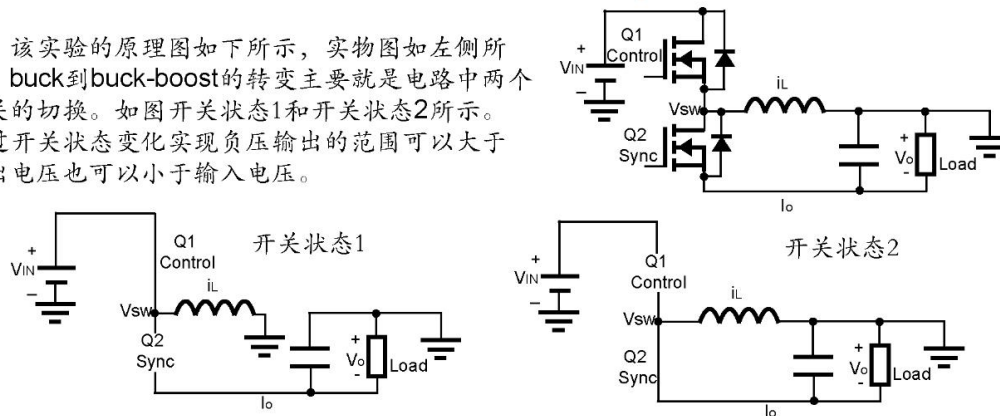
而早 LCD 上显示的内容跟使用 B 也是一致的，计算公式和实现代码可参见使用 B。



实物图

通过拓扑变换实现负压生成

该实验的原理图如下所示，实物图如左侧所示。buck到buck-boost的转变主要就是电路中两个开关的切换。如图开关状态1和开关状态2所示。通过开关状态变化实现负压输出的范围可以大于输出电压也可以小于输入电压。



通过拓扑变换实现负压生成测试实验

- 1、理解原理图，编写Launchpad代码（参考代码见网上资源）。
- 2、在母板上连接TIVA、LCD模块、DC-DC Buck (降压)模块，准备实验。
- 3、在降压模块上完成通过拓扑变换实现负压生成测试的跳线及仪表连接，如右图所示。其中①是负压输出连接负载的接口；②是负压输出的测试点；③是开关电源的开关节点；④是反馈电压的测试点；⑤是地的测试点；⑥是可调电阻，如果想改变输出电压值可以改变滑变阻值，起到电压调节作用；⑦是电路中唯一一个跳线连接，即短接J4完成电路连接。
- 4、连接负载，如右图所示。注意负载的功率和阻值大小。
- 5、给TIVA上电，先将电压调节到-15V输出，然后改变负载大小，对负压输出的性能进行测试。
- 6、调节电路中的可调电阻，观察输出电压的调节范围。

注意

- 连接仪表及跳线时，断开电源。



- 实验套件中没有提供电阻负载，实验中的负载需要自己制作。这里的负载是阻值100Ω、功率1W的电阻。通过电阻的串并联改变负载大小。

通过拓扑变换实现负压生成测试主要步骤

