**从零开始的有刷直流电机驱动**

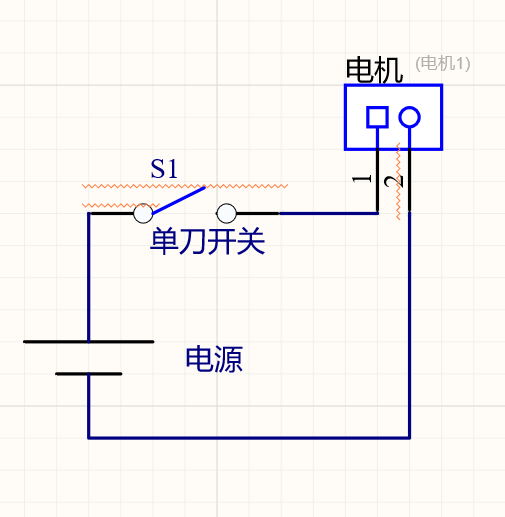
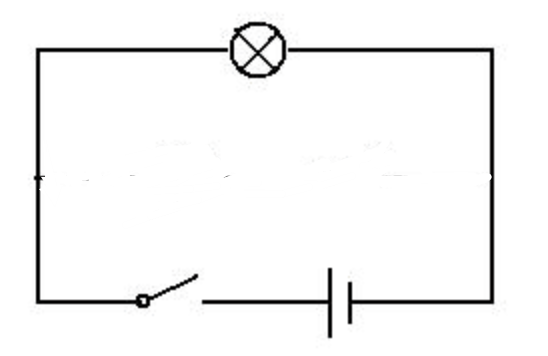
先明确一下我们的最终目标：让小车跑起来

小车是靠轮子跑的，轮子是靠电机带着转的

电机是用的有刷直流电机

所以我们现在的任务是：如何实现直流有刷电机的转动。

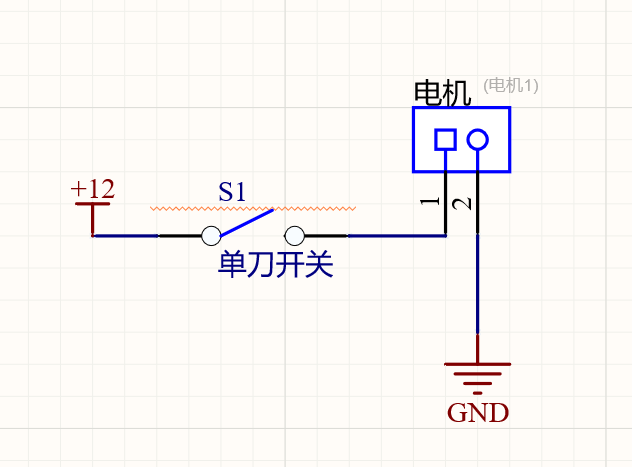
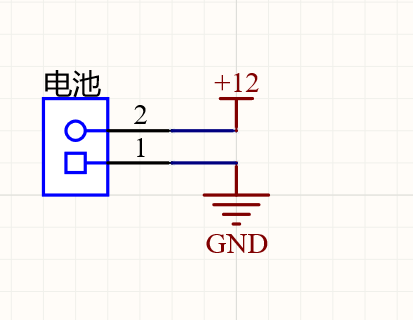
怎么让直流有刷电机转起来呢：一端接正一端接负，给电就会转，简单友好符合直觉。



左图是初高中物理中常见的电路图，现在我用绘制电路原理图的软件把它画到原理图中去，如右图，（红色的波浪线是一些报错，不要理会）

左图的×一般是灯泡，或者用电器，在这里用电器是电机，在右图中，电机用了一个有两根线的端子表示。

现在我们换一下电源的符号，改成电路原理图常用的表示接电的方式



可以看到，我把电源的符号也换成了一个端子，并且把电源和用电器断开了，在断开处分别加上了+12和GND的标签

在电路原理图中，接12V电源和接地(ground)都会用这样的符号表示，而不会用线直接连到电池画成一个回路。

显而易见的，哪怕没有用线直接连起来，有相同标签的地方都是要连起来的，比如+12和GND。用网络标签可以避免电路复杂的时候，线乱七八糟地连来连去。

这里讲一下接地，地是一个参考0电势面，电压都是相对于0电势面而言的，如果电路接入大地，则一般大地为0V，我们一般使用电池供电，则电池的负极是地，是0电势面，是0V。接地是很重要的，从电源出发，电流最终都会回流到地，即电池的负极，形成回路，像我们熟知的那样。

光是通电让电机转起来是不够的，通常我们会希望能够对电机进行更多的控制，比如调速和正反转。

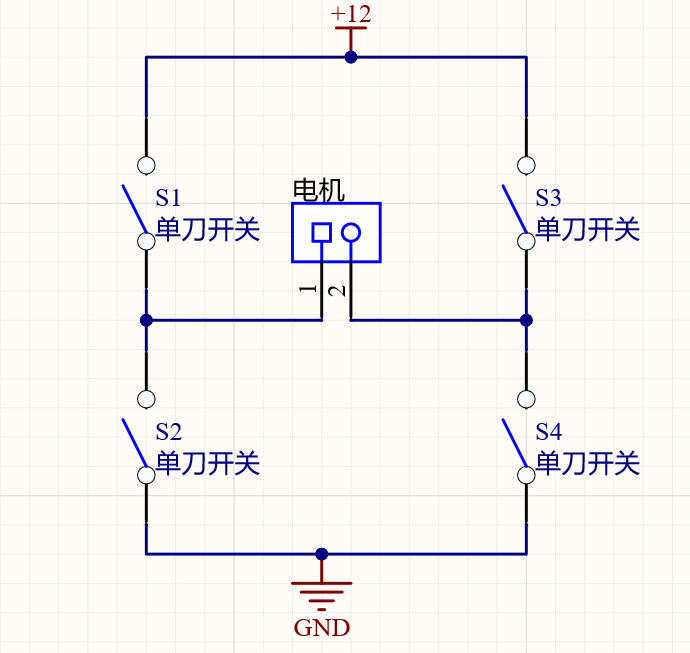
关于调速，这里引入一条电机的调速特性：电压与转速成正比。Speed=K\*V

电压越大转得越快，这很符合直觉，所以这里不解释为什么电机有这样的特性。

关于正反转，电机本身是没有正负极的，想要让电机反转也很简单，把原来的正负反接就好了。同样很符合直觉，不解释为什么。

说起来容易，但实际上我们根本不可能随时地把电机拔出来再反接回去，或者是把电池拔下来反接回去。实际工作时的电路都是固定好的，所以我们不得不让电路结构变得复杂一些，以实现电机正反转。

于是我们引入了H桥电路。

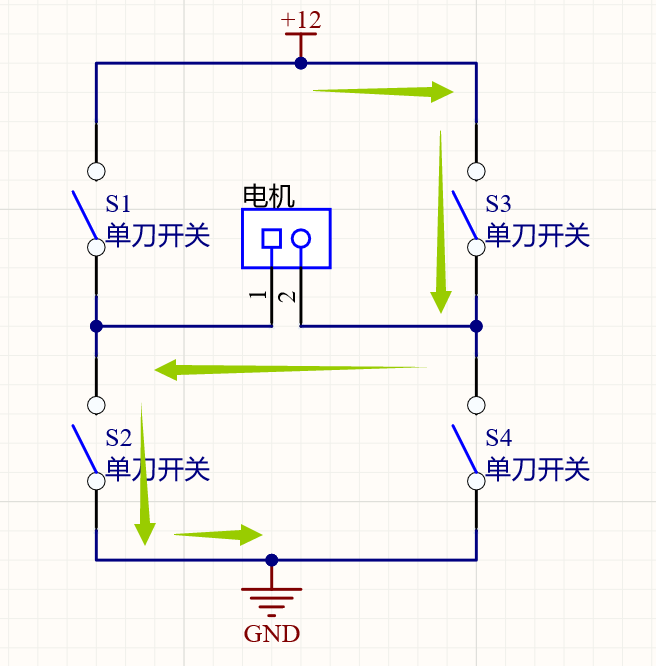
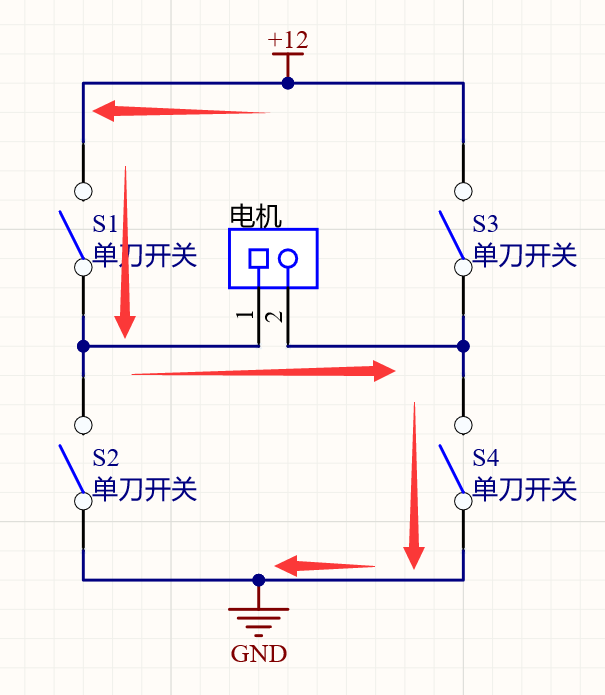


观察H桥的结构，可以看到，H桥的4条腿上有4个开关，”H”的横杠上连着电机，H桥的上端连着电源正极，下端接地。

当开关1、4闭合，电流从电机的1流到2，如红线所示。

当开关2、3闭合，电流从电机的2流到1，如绿线所示。

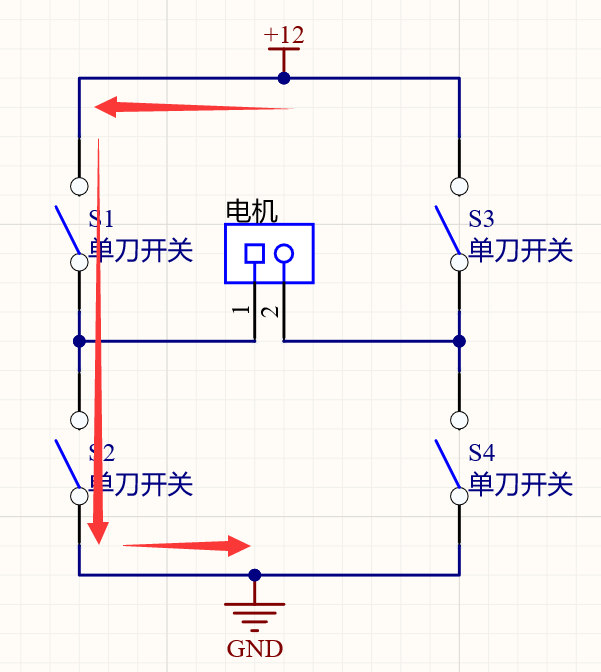
这样就完成了电机正反转的切换，不需要再改动电路改动接线，只需要按下开关。



这里要注意一个问题，H桥的同侧是不能同时导通的

比如S1、S2同时闭合，则+12直接连到了GND，相当于电池的正负极直接连到一起，短路了，会火花带闪电的，危！

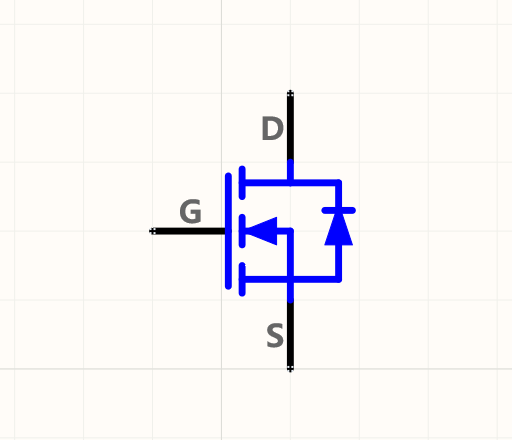
所以S1和S2是互斥的关系，有你没我，有我没你。



刚才我们说到，只需要按下相应的开关就可以实现电机正反转了，但按开关这个操作实在太蠢了，你会按那么多开关吗。我们这是电路，当然得用电来控制开关，怎么能用人力呢，再说了人力也不靠谱，一不小心同时按一下S1和S2就是刺啦放烟花。

所以我们接下来需要引入实现开关功能的电子元器件：MOSFET，人称mos管。

增强型N型功率Mos的电路符号和实物如下图所示



可以看到mos有三根脚，分别叫G、S、D，全称为：Gate，Source，Drain，

中文名是：栅极，源极，漏极

Mos会在模电中具体学习原理，我在这里只提一些必要的特性，我们只需要会用就好。比如mos的电路符号为什么是这么画的，大可不必理会，只需要当成是一个黑盒子来使用就可以了。比如mos有N型和P型，但我这里只讲N型。当然期末考试考不考原理我不知道，那得看老师讲了多少吧。

好了接下来让我们来了解它的特性：它是怎么实现开关的？

先来点感性认知：D和S就像是单刀开关的两侧，G是控制信号，是门(gate)，也是按下开关的手。

显然我们按开关是要一定的力度的，要用力啊年轻人，所以要按下G也有一个阈值(threshld)。

值得注意的是，这里衡量按下G的“力度”，指的是G点相对于S点的电位，即GS的压差

即VGS=VG-VS

而不能是VGD,，不能说我给个VGD来让D、S导通，毕竟mos只是像开关，不是真的两头都一样的开关，D和S名字都不一样，肯定不能互换的啦。

重新叙述一下：

在G、S之间加上一个大于阈值的电压，就可以使D、S之间导通。即VGS>Vth

也就是说，MOS是电压控制型的器件，控制信号为电压

N型mos的Vth是一个大于0的正值，所以VGS< Vth或者VGS为负值，D、S之间都不导通，等同于开关的断开状态。

讨论完G、S间的电压，接下来就要讨论D、S间的电压了

让我们先来看看模电教科书是怎么画的，这里第一次看不是太好理解，不想看也可以稍作跳过

在模电教科书中，一般是VD>VS，可以让S极接地，让D极接高电位，如+12V，如图

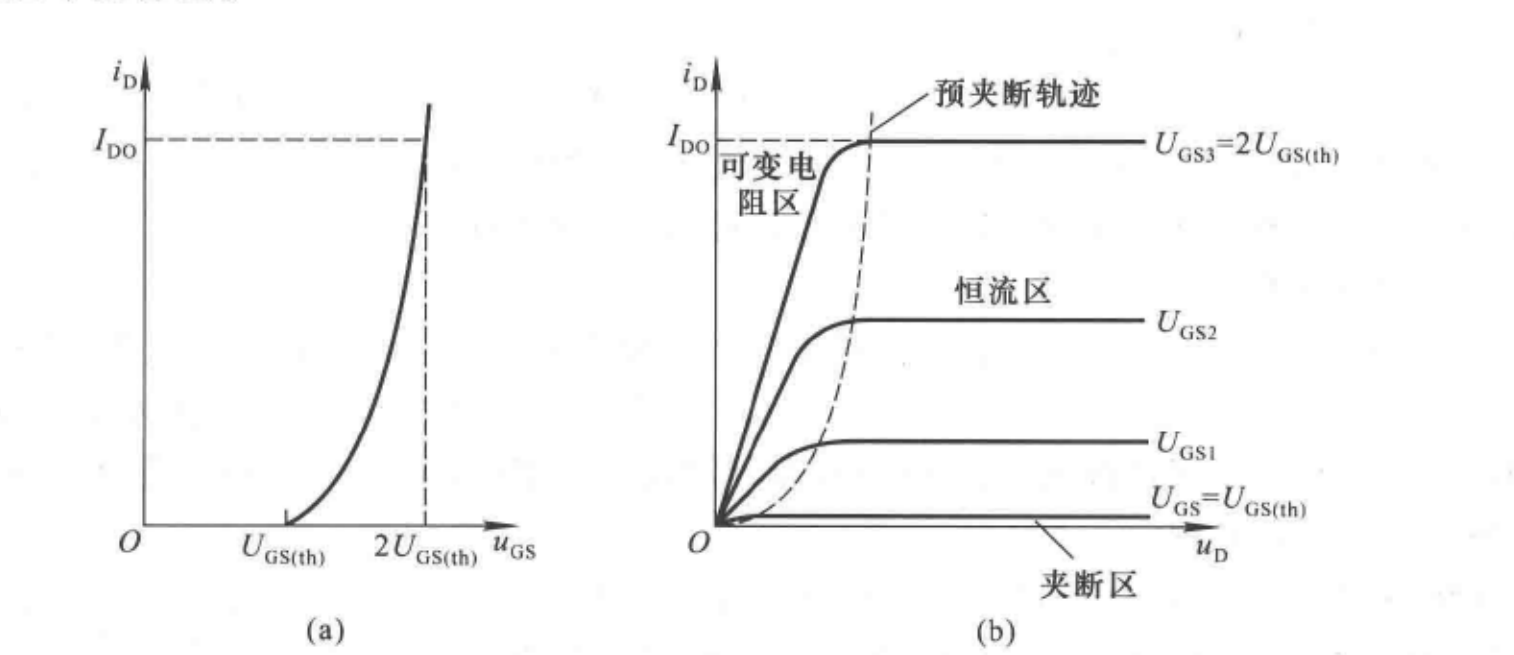
ID定义为是从D流向S的电流

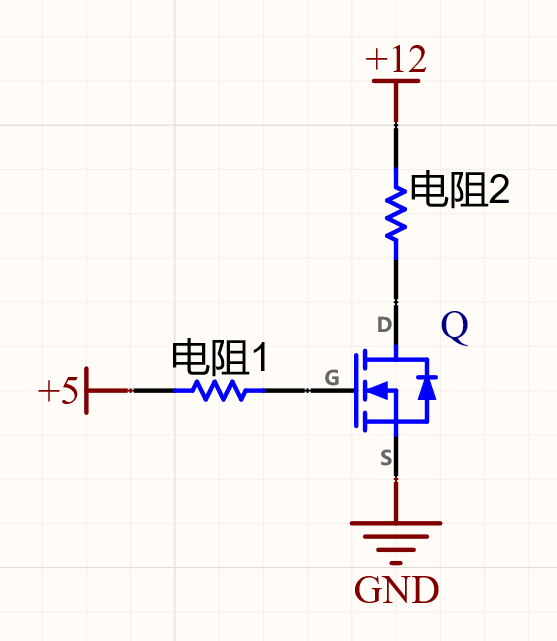
从(a)中可以看到，固定VDS，改变VGS，当VGS小于Vth时，mos不导通，ID为0

当VGS增大到大于Vth后，ID逐渐增大

从(b)中可以看到，当VGS>=Vth,，ID才不为0；并且固定一个大于阈值的VGS，改变VDS时，ID先是变大，增大到一定值后趋于平缓，不再随着VDS的增大而增大。

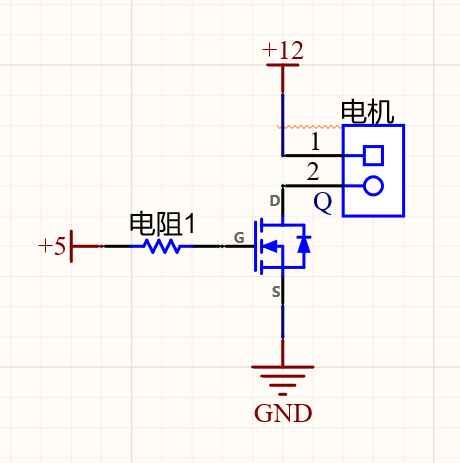
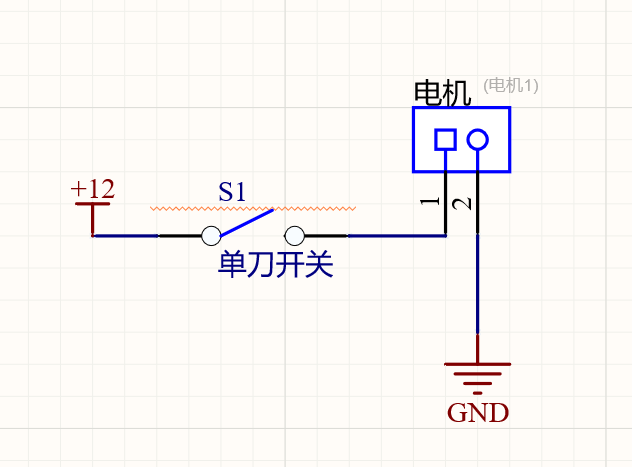
电路可用下图示意（假设5V大于Vth）





这样我们就完成了开关闭合时电压情况描绘：VGS>Vth，VDS>0

接下来我们把开关换成mos，把电机接入电路，对比如下



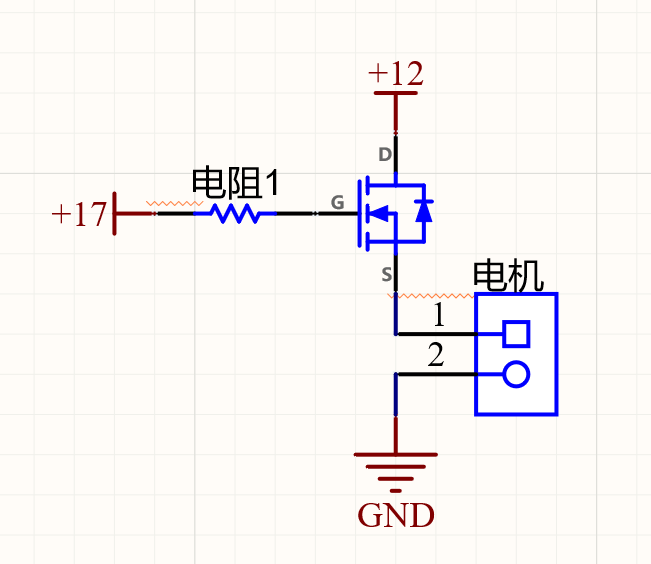
这里大家注意到，原来的开关我是放在了+12和电机的1号口之间，mos放在了2号口和GND之间。

开关放哪都无所谓，但mos不一样，mos的导通是要条件的，VGS>Vth

如下图，mos导通后，DS导通，S极的电位为12V，此时要满足VGS>Vth

，就必须使G的电位抬升，若要维持原来的VGS=5V，就得让VG=17V

所以我把mos放在了靠近GND的一侧，称为低侧，让S极与GND相连



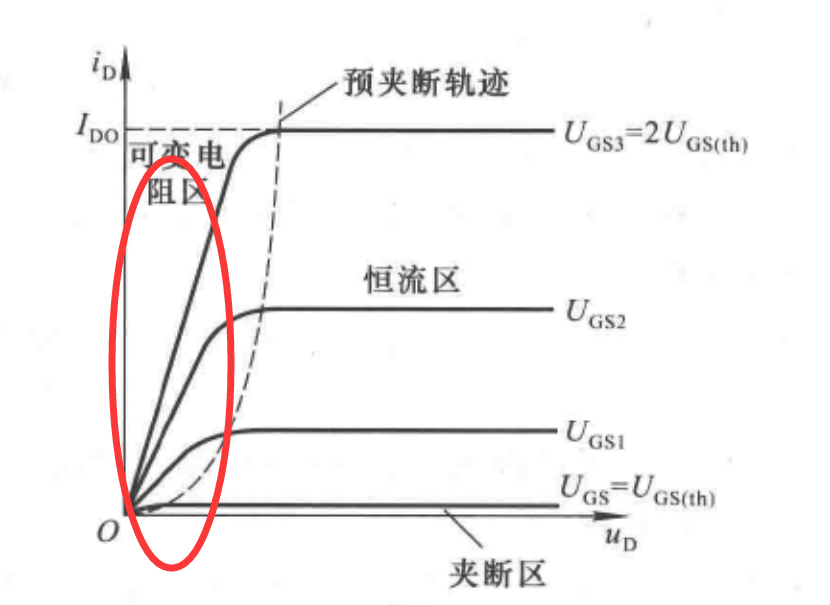
这里我们又引入一个重要概念：DS之间导通时，DS之间可以等效为一个动态小电阻，称为Ron或RDS，一般为mΩ级别，从几mΩ到几百mΩ都有。

之所以说是动态，是因为这个等效电阻的阻值与VGS的大小有关，VGS越大，RON越小，

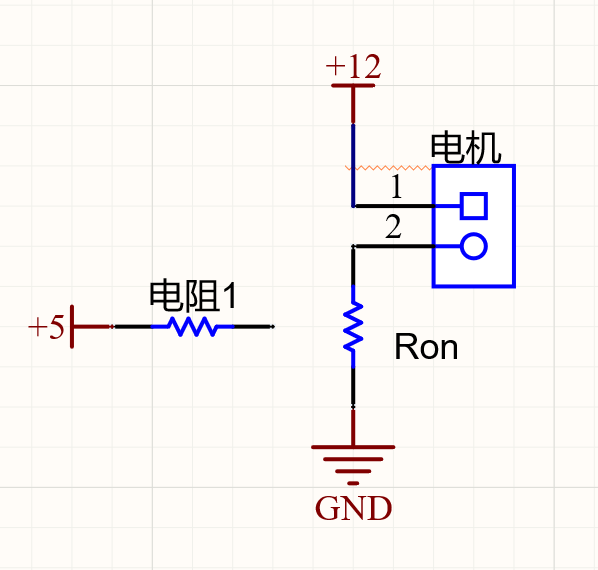
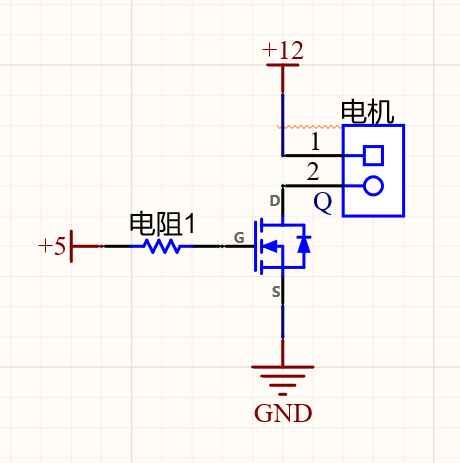
也就是说，mos作开关时，断开时是在夹断区，导通时是工作在如图所示的可变电阻区

由图可知，在可变电阻区，ID和VDS成正比：I=U/R

斜率为1/R，VGS越大，Ron越小，斜率越大



导通时的等效电路对比如下



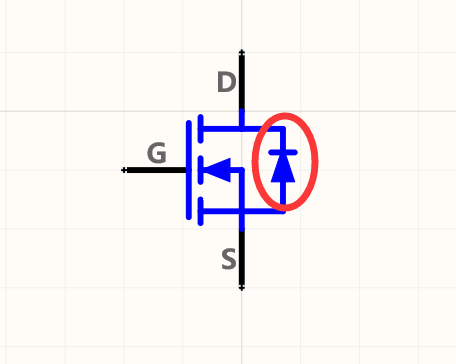
这样，我们就已经用mos实现了开关功能

上面模电教科书的曲线只描绘了VDS大于0的情况，并没有讨论VDS小于0，即VS>VD

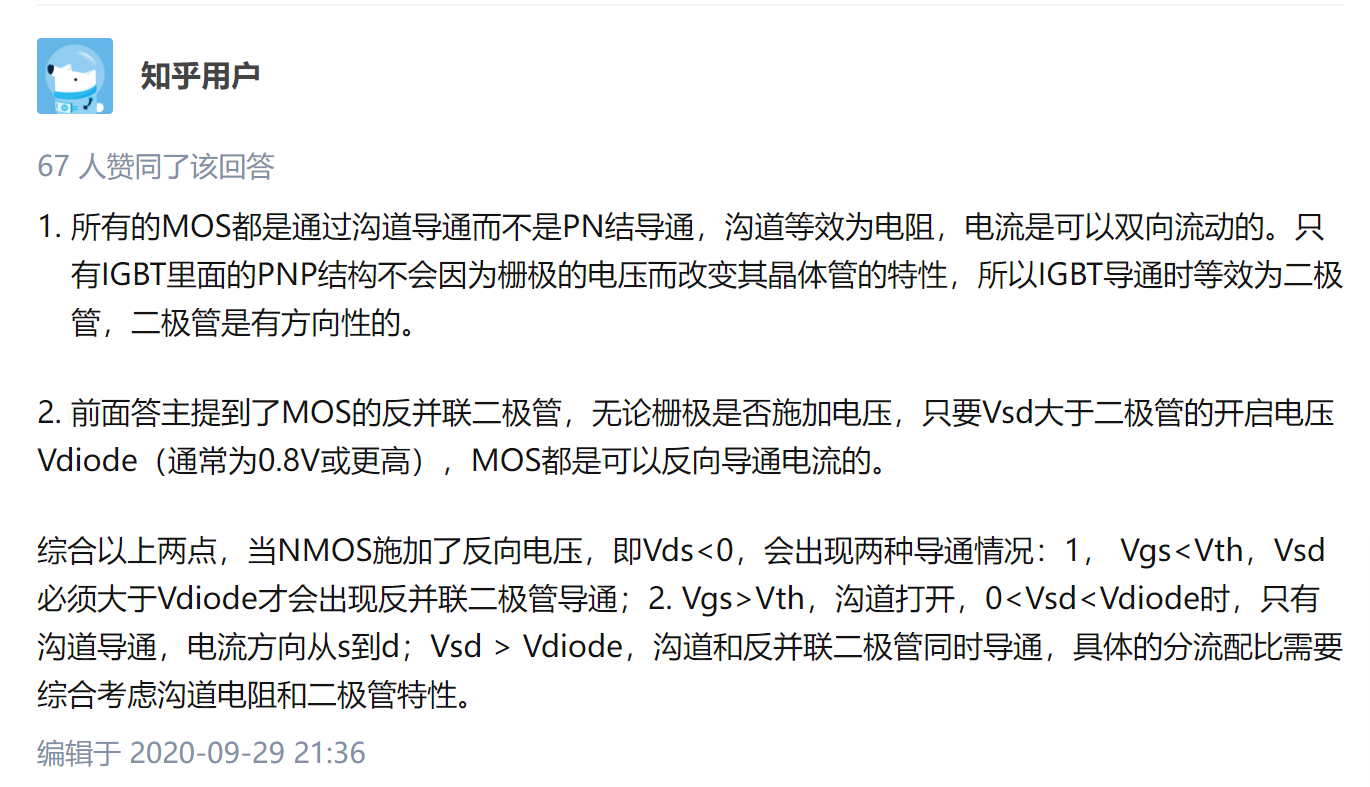
细心的同学可能会发现，这个mos符号里有一个二极管的符号

这是由制造工艺，或者说内部结构决定的，称为体二极管或寄生二极管

显然，VS>VD时可以使体二极管导通，所以此时mos也是可以导通的。

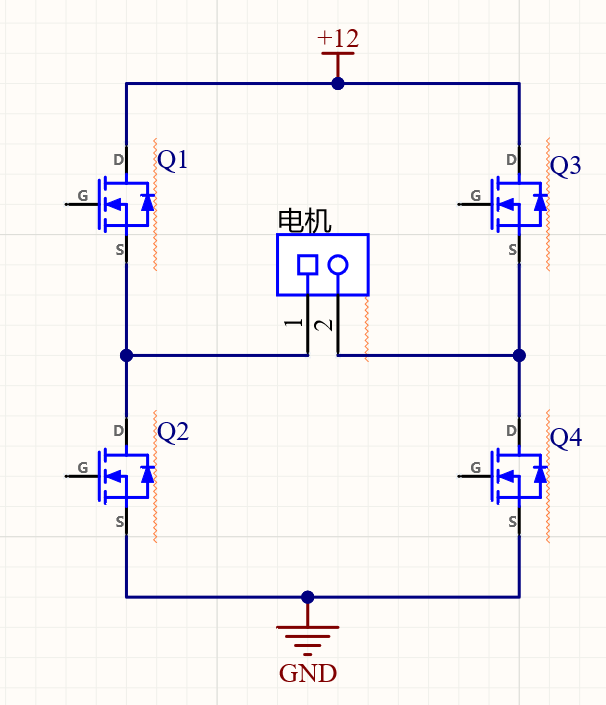
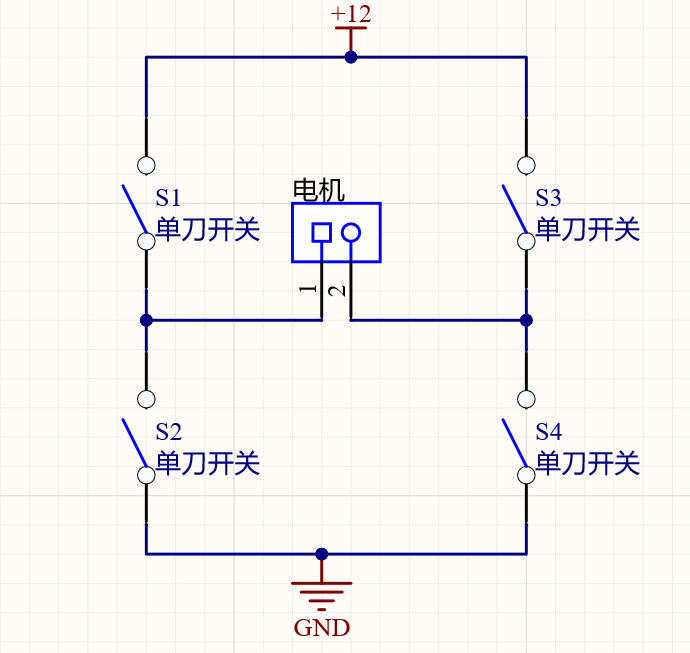


参考知乎的一个回答，所以只要VGS>Vth，mos的DS之间就等效为一个小电阻，VDS>0或VDS<0均可导通。



好了花了这么长的篇幅终于把mos告一段落了

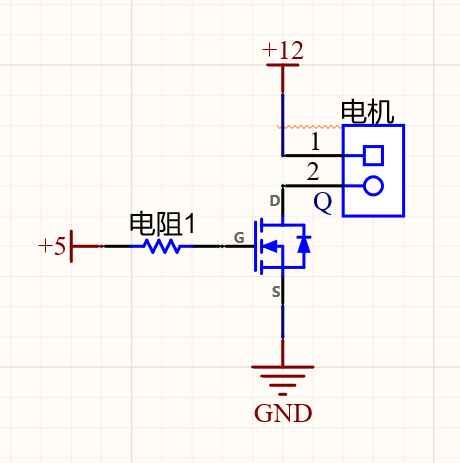
让我们把H桥上的开关都换成mos



我们用H桥解决了电机的正反转问题，用MOS管实现了开关功能，但从上面的电路图可以看出，G极还没有连接，所以现在我们的问题是：如何随心地控制mos的导通和断开，即控制mos 的栅极电压，即令VGS>Vth和VGS<Vth

这个问题暂时按下不表，因为控制H桥的栅极电压时我们要把正反转和调速一起解决。已经在正反转停留太久了，让我们先看看电机是如何实现调速的

先看看不用实现正反转的简单电路。



如上图，给G极加5V或者不加可以实现电机的转动与停止，加到电机两端的电压为12V或0V，但这样是不够的，我们需要一个转速可调的电机。那么如何使加到电机两端的电压在0~12V之间改变呢

依然是靠——开关。

开了电机转，关了电机不转

但我们开关得够快呢？

电机是有惯性的，开关不能马上反应，得等一小小小小会，有一个加速和减速的过程

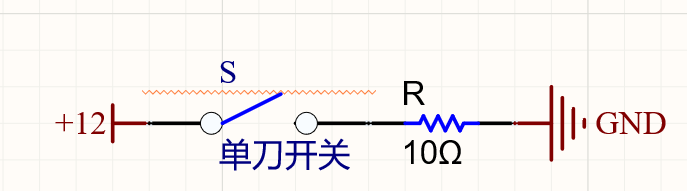
在电机没有反应过来之前，哎，我关上了，又趁电机没有反应过来，哎，我又开了。

只要开关的频率够小，电机就来不及停下，而是在加速和减速过程中来回变动

只要开关的频率够小，电机加速和减速的幅度就会够小，时间就会够短，小到肉眼看不出来，那么在宏观看来，电机就是以一个稳定的速度在转动。

那么这个在稳定高频的开关下保持稳定的速度是多少？

这就是能量守恒的问题了



如上图所示，我们共考虑10s的长度

我先开5s，再关5s，开关周期10s，频率为0.1HZ，功率P=12/10\*5=6W

我先开1s，再关1s，再开再关，开关周期2s，频率为0.5HZ，功率P=12/10\*1\*5=6W

我先开0.1s，再关0.1s, 再开再关，开关周期0.2s，频率为5HZ，功率P=12/10\*0.1\*50=6W

我先开0.0001s，再关0.0001s, 再开再关，开关频率为5000HZ，功率还是6W

不论开关频率有多大或者有多小，都不影响功率大小

开关时间的占比才影响

我先开0.00002s，再关0.000018s，再开再关，开关频率还是5000HZ，功率是1.2W

P=U2/R，R一直都是10Ω，是一个恒定量，微观上，虽然每次开关闭合加到电阻两端的电压都是12V，可用P=U2/R等效一下，在宏观看来，加到电阻两端的等效电压就变小了。

等效后的电压U‘=U\*D

其中D是开通时间占据整个开关周期的百分比，称为占空比(duty cycle或duty ratio)。

（明明叫“占”和“空”的比例，实际比的确实“占”和“占”+“空”）

前面我们提到了，电机的转速和电压成正比

这个电压依然可以是等效后的电压

于是我们在开关频率足够大的时候，通过改变占空比，调节了电压，实现了调速。

这么快的开关频率也不是人力能达到的，还好我们之前已经确定是通过电信号来控制mos开关，电可就跑的非常快了，区区几k十几k的频率不在话下。

那么接下来就得看一下我们的控制信号：PWM波

（PWM，脉宽调制，Pulse Width Modulation）

在讲PWM之前先提一下电平和模拟量，数字量的概念

数字量是离散的，如0、1、2、3……，每个数字之间有最小间隔，在计算机中，一般以二进制表示数，只用0和1，例如9是1001

模拟量是连续的，如1和2之间可以有1.5342，可以有1.293875

电压就是一个现实世界中的模拟信号，可以有3.2932V，，可以有4.9238V

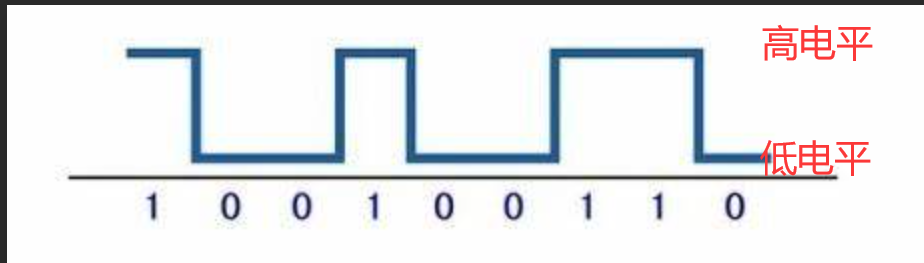
高低电平就是在用模拟的电压表征数字的1和0

高电平就是1，低电平就是0，亦是写程序时的1和0

低电平对应的模拟的电压就是0V，而高电平对应的电压标准则有所不同，常见的有5V和3.3V等

数字转模拟时一般比较精准，比如在高电平标准是3.3V时，把1“翻译”成3.2932V

模拟转数字时一般是允许有一定的误差范围，比如在高电平标准是3.3V时，2.8~3.5V可能都会“翻译”成1，1V会“翻译”成0。当然，这个“误差范围”有多大，翻译成1和0的分界点在哪，得看具体元器件的规格。

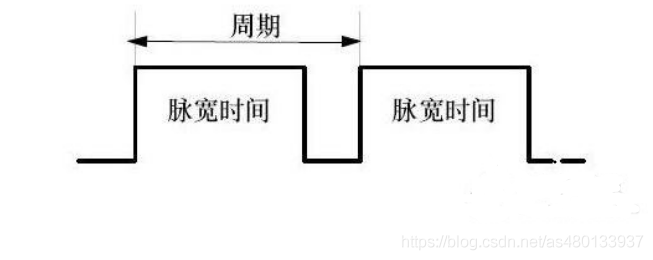


如下图，PWM波表现为一个周期性地高低电平切换的矩形波信号

PWM波有两个重要参数：

频率：周期的倒数

占空比 = （脉宽时间 / 周期 ）\*100%



调节PWM的占空比可以调节它输出的等效平均电压

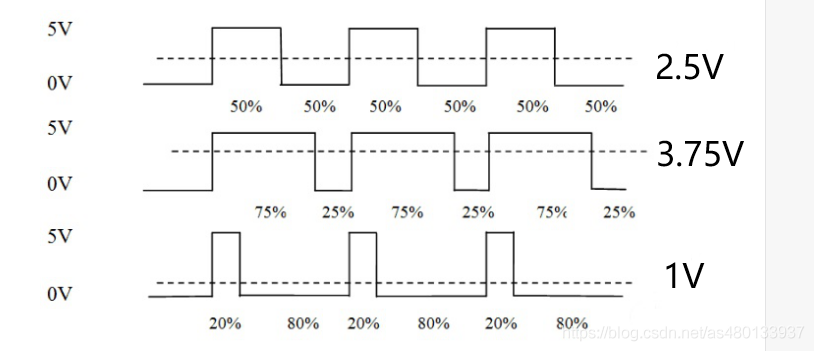
PWM波一般由MCU产生

（MCU，微控制单元(Microcontroller Unit ，又称单片微型计算机(Single Chip Microcomputer )或者单片机）

单片机的输出的是数字信号，只能输出高电平和低电平

通过PWM，单片机完成了模数转换，输出了一个模拟的电压值

至于单片机怎么产生的PWM，在此不做论述。

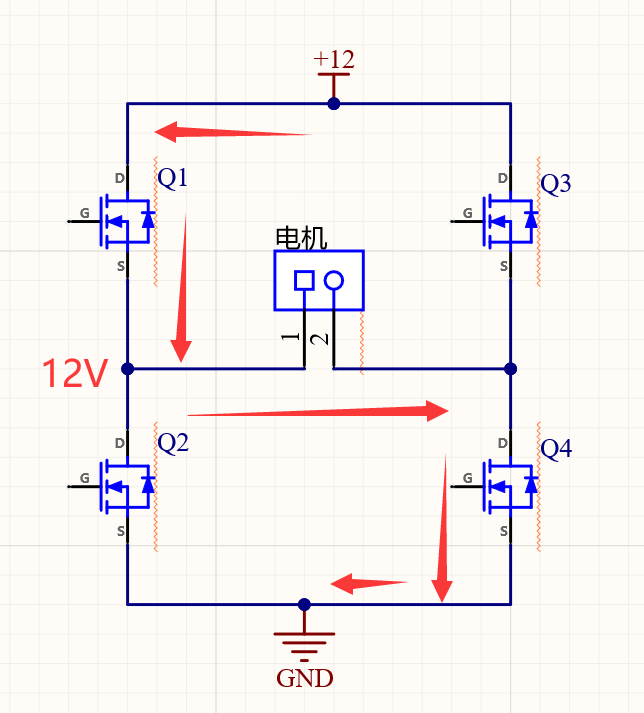
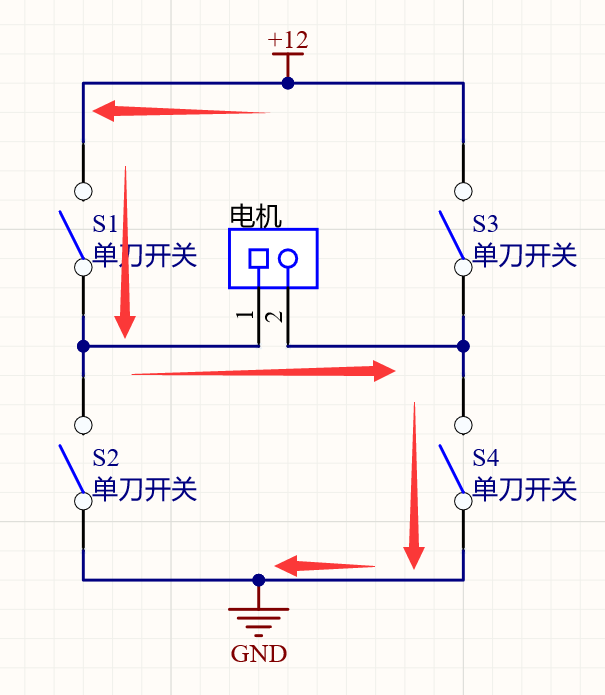


现在我们有了实现正反转的H桥，有了作开关的MOS，有了可以调速的控制信号PWM

是不是就可以直接把PWM加到H桥的4个栅极上去呢？

是不行的。

Mos的栅极驱动还是有点复杂的，不是那么容易能够实现，比如说如下过程中，Q1、Q4导通，12V正压加到了电机的1号口，那么此时Q1在高侧，要导通Q1，Q1的栅极信号的电压需要大于Vth+12V，单片机可输出不了那么高的电压。



所以我们得让专业的家伙出场了

经典的半桥/栅极驱动芯片——IR2104

接下来，让我们看看它有什么功能，又是怎么用的呢

参考链接：

[IR2104电机驱动\_zhuimeng\_ruili的博客-CSDN博客\_ir2104驱动电路的使用方法](https://blog.csdn.net/zhuimeng_ruili/article/details/104378131)

[(10条消息) 【大电流H桥电机驱动电路的设计与解析（包括自举电路的讲解，以IR2104+LR7843为例）】\_默默无闻小菜鸡的博客-CSDN博客\_lr7843](https://blog.csdn.net/qq_44897194/article/details/107397079?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522164338866116780255219863%2522%252C%2522scm%2522%253A%252220140713.130102334..%2522%257D&request_id=164338866116780255219863&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~top_click~default-2-107397079.first_rank_v2_pc_rank_v29&utm_term=ir2104&spm=1018.2226.3001.4187)

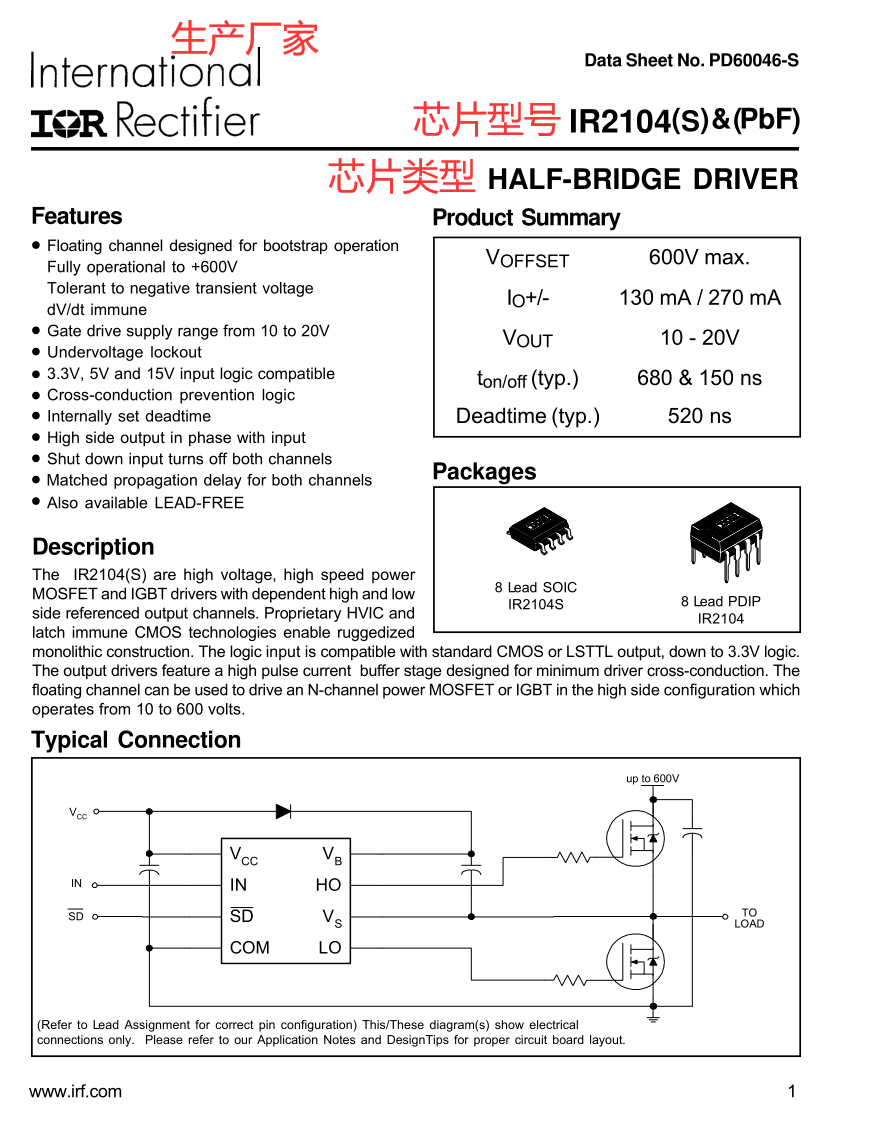
[STM32+IR2104的H桥电机驱动电路详解 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/261724402)

来看看它的data sheet吧

（数据手册，大概类似于说明书带有一些测试数据，很遗憾，data sheet基本上是全英的，而且对于新手来说，上面全是不知所云的专业名词）

（更遗憾的是，我至今仍不能完全读懂任何data sheet，只能勉强把我能懂的一小部分翻译给大家）

这是IR2104的data sheet的第一页，通常第一页都会给出最重要的信息，供各位攻城狮挑选芯片。



我们暂时涉及不到具体参数的计算，所以先忽略它们吧，我们要看的是Typical Connection，典型电路设计，并对此做一些原理上的说明。

1. IR2104驱动原理图讲解
2. IR2104驱动参数计算

先来举个例子，小明有个一个水龙头，这个水龙头不能调节，只能开到最大，但小明想要小点的但是连续的水流，于是小明拿了个大水缸子，水缸子上开个洞，从洞里流出的水流刚好是小明想要的大小，于是小明把大水缸子放到水龙头底下，打开水龙头，于是这个大水缸子一边注水一边放水……

问多久能把水缸注满？

啊不是，我们不是要把水缸放满水，我们只是要个小水流罢了，所以不能让水满出来，那太浪费了，看到差不多了就把水龙头关上就行，然后水下去的差不多了又把水龙头打开。就这么开开关关，就得到了稳定的连续的小水流。

显然， 流进水缸的水量是等于流出水缸的水量的，只有进出平衡了，水缸才不会溢出或者见底。