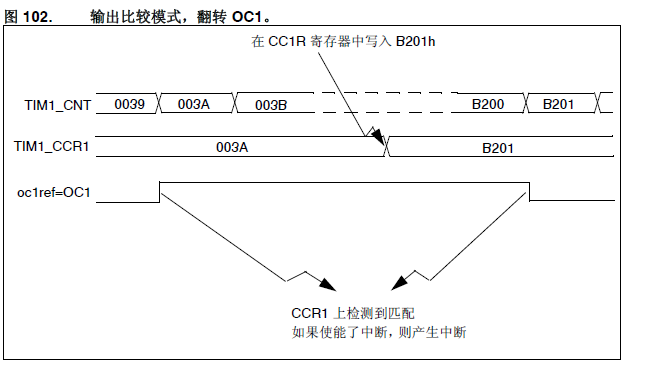
**四轴57&42步进电机梯形加减速实现**

本例程使用TB6600步进电机驱动器控制57&42步进电机，使用专门的驱动器可以做到事半功倍的效果，控制电机的时候，只需要使用定时器输出可调频率的脉冲方波即可控制电机的速度，不需要再考虑步进电机的通电顺序等。

STM32的脉冲输出方式是很灵活的，不只是有PWM模式一种，例程使用的方式是**输出比较模式**，意思就是当比较值与计数器数值相等的时候就翻转电平，**不是使用PWM模式，改变脉冲频率不需要改变ARR寄存器（自动重装载值）。！！！**



从图中可以看到计数器CNT一直在计数，当计数器与比较器（CCR）相等的时候就翻转输出电平：

使能了定时器比较输出中断，设置了通道的脉冲数为Toggle\_Pulse（500），在启动定时器运行之后，定时器从0开始计数，等计数到Toggle\_Pulse（即500）值时就会产生中断，翻转通道引脚，并执行

HAL\_TIM\_OC\_DelayElapsedCallback函数，在函数内，我们读取当前定时器计数值保存在变量count（此时该值为500），并设置新的比较值为为count+Toggle\_Pulse（即500+500=1000）。接下来，定时器继续计数，等到计数值到1000时，就又产生中断，翻转通道引脚，并把比较值设置为1500（1000+ Toggle\_Pulse），如此循环执行…..最终的效果也就在定时器通道引脚输出持续的脉冲信号，并且脉冲信号的周期为2\* Toggle\_Pulse，也就是通过改变Toggle\_Pulse来改变脉冲频率。

假设：count = 65530（0xFFFA），Toggle\_Pulse = 1000（0x03E8）

count+Toggle\_Pulse = 66530（0x1 03E2）

由于定时器计数器和比较值是16位的，最大值只能是0xFFFF，计数溢出之后，会从0开始（0xFFFFF -> 0）。按照上面的计算结果，如果直接将66530（0x1 03E2）赋值给CCR，则实际上是将0x03E2 赋值给CCR，那么定时器下一次中断的时候就是在0x03E2，两次中断的时间就是（0x03E2 – 0 ） + （0xFFFFF – 0xFFFA） = 0x03E7(999) ,这里数值上不等于Toggle\_Pulse，但时间上却是间隔了1000（道理就跟数组元素是[10]，但下标最大是[9]一样）。

本例程控制步进电机实现梯形加减速算法。电机从0开始加速，再匀速，最后减速到停止。

### 软件版本信息

|  |  |
| --- | --- |
| 软件 | 版本 |
| STCubeMX | 4.21.0 |
| Cube F4 Firmware Package | 1.18.0 |
| Keil | 5.22 |
| IAR | 7.4 |

### 跳线帽情况

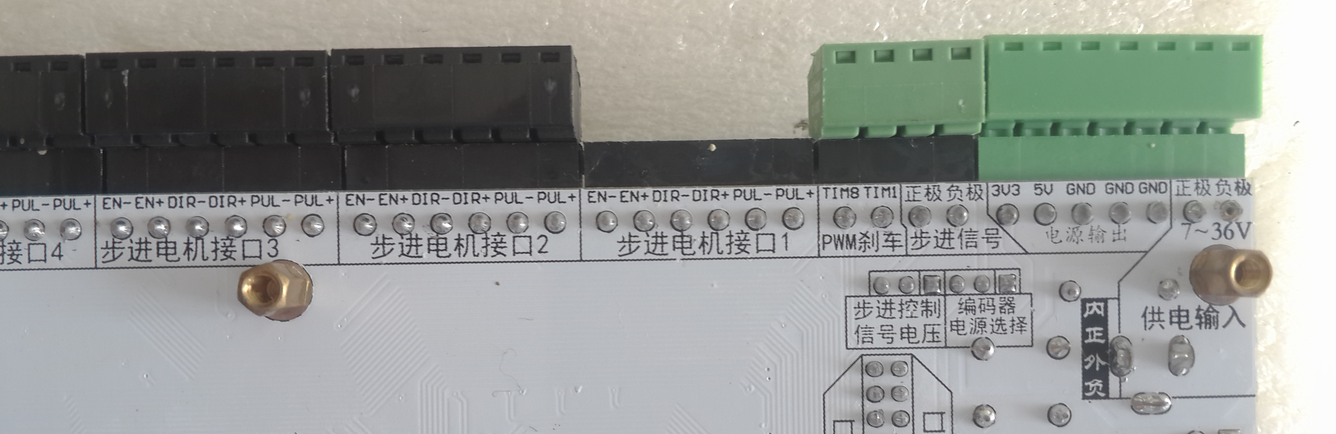
/\*\*\*\*\*\*\* 为保证例程正常运行，必须插入以下跳线帽 \*\*\*\*\*\*\*\*\*\*/

|  |  |  |  |
| --- | --- | --- | --- |
| 丝印编号 | IO端口 | 目标功能引脚 | 出厂默认设置 |
| JP2 | PB6 | RXD | 已接 |
| JP2 | PB7 | TXD | 已接 |

### 接线说明

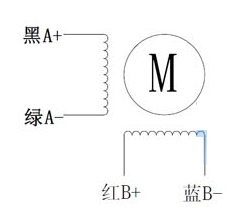
|  |  |
| --- | --- |
| 脉冲输出 | TB6600驱动器 |
| PI5 | **PUL-** |
| PD3 | **DIR-** |
| PD7 | **ENA-** |

使用板上专用的步进电机接口接口与驱动板连接。开发板背面有丝印指示接线，可以与驱动器直接连接。



步进电机接线：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **引线** | **A+** | **A-** | **B+** | **B-** |
| **颜色** | **黑色** | **绿色** | **红色** | **蓝色** |



开发板的供电可以选择使用USB接口的5V供电，或者使用DC座的7~36V电源供电。

驱动器使用24V电源供电。

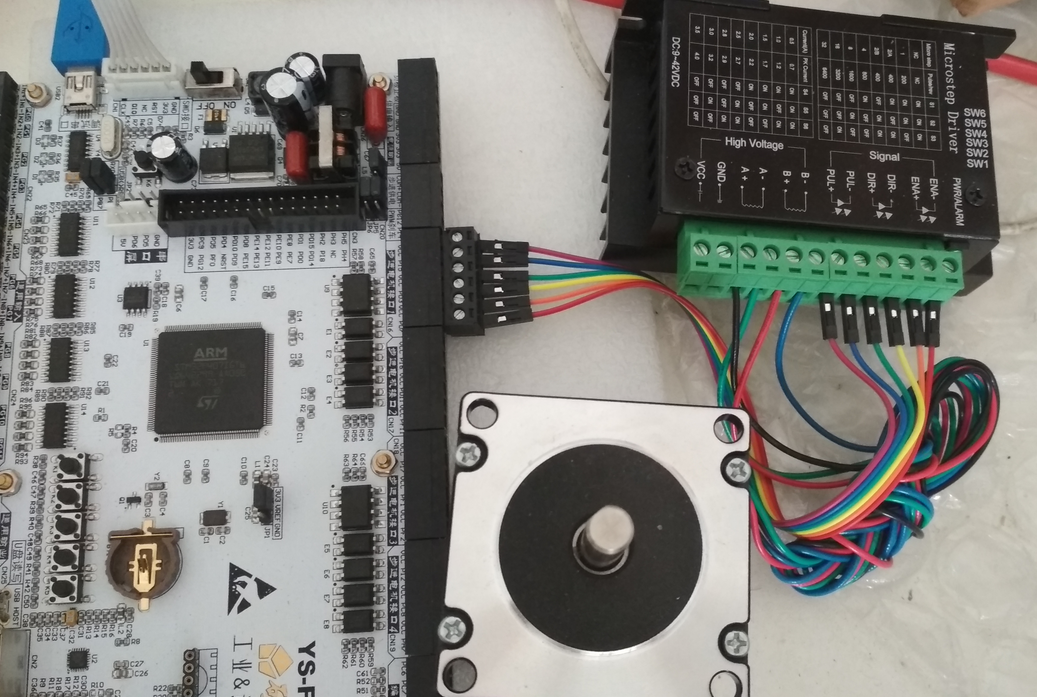
### 操作与现象

使用开发板配套的Mini USB线连接到开发板标示“调试串口”字样的Mini USB接口为开发板供电。

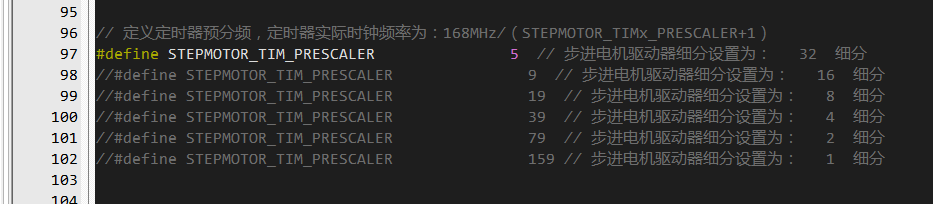
下载完程序之后:

按下KEY1,步进电机开始加速,加到一定速度之后匀速,然后开始减速,最后到达

目标位置就停下,加速度,减速度,匀速度,目标位置都可以通过程序控制.



代码里面设置的定时器分频跟驱动器的细分数是没有关系的。



仅仅是用于计算方便而已，实际上的细分设置是在驱动器上面的波动开关。