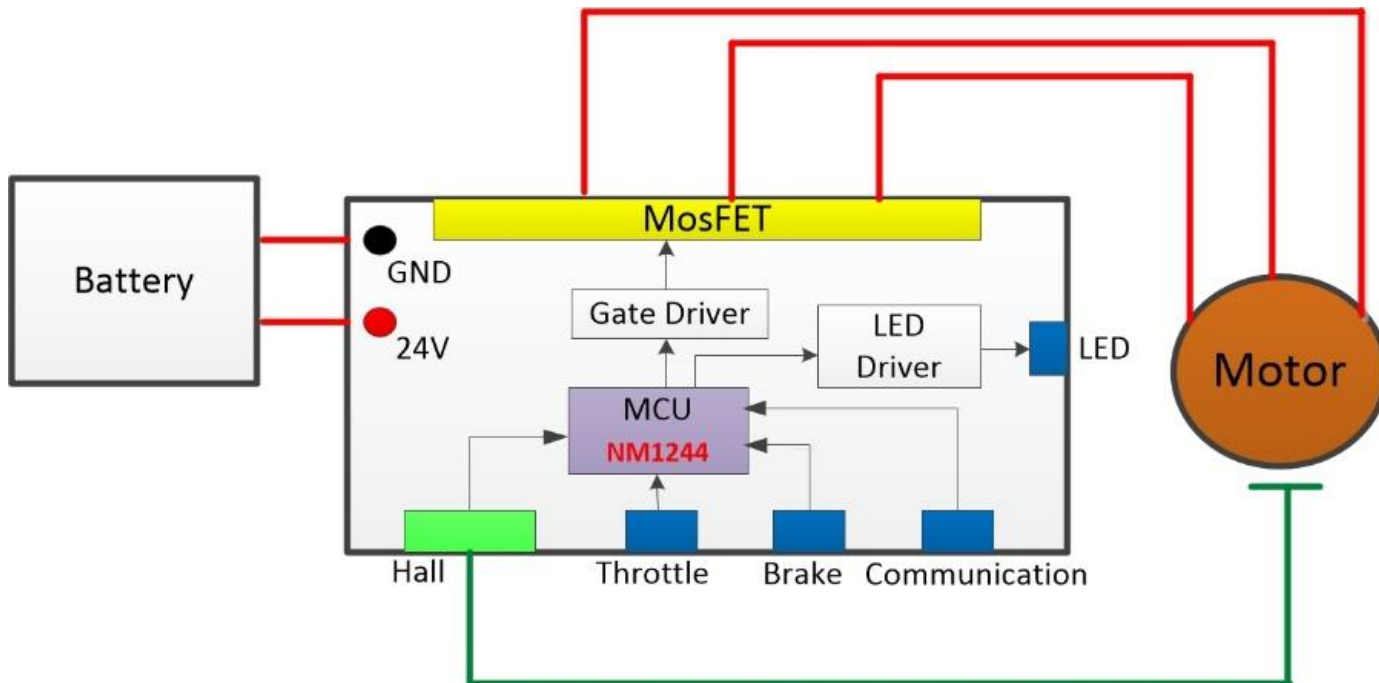


# NM1244实现FOC滑板车控制 硬件及软件内容说明

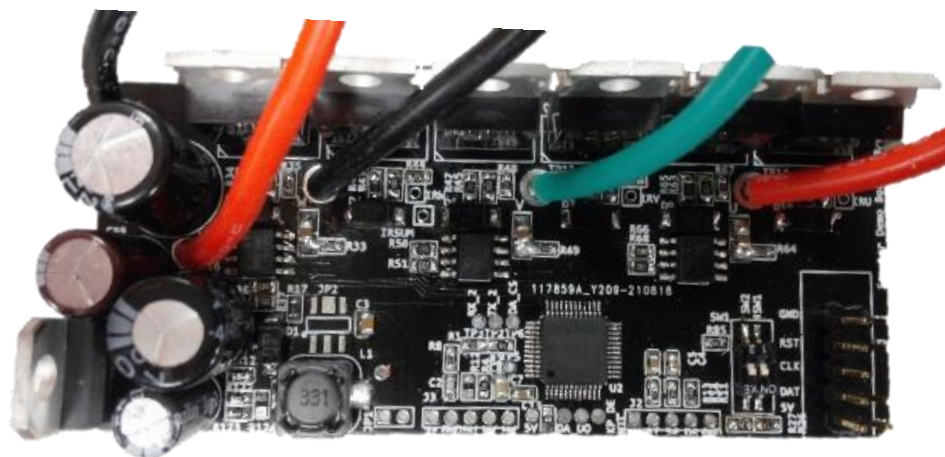
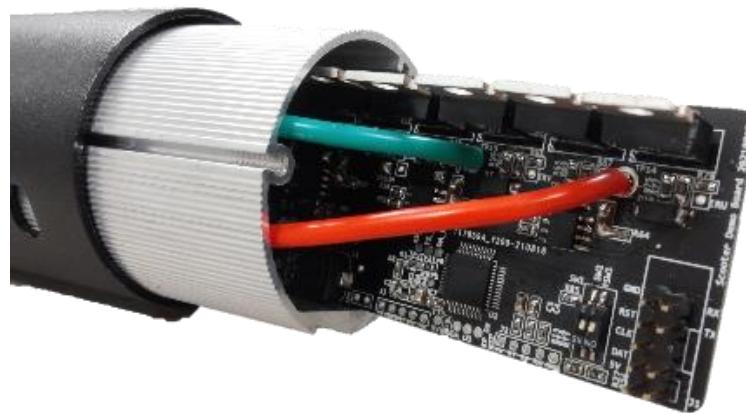
曾明雲



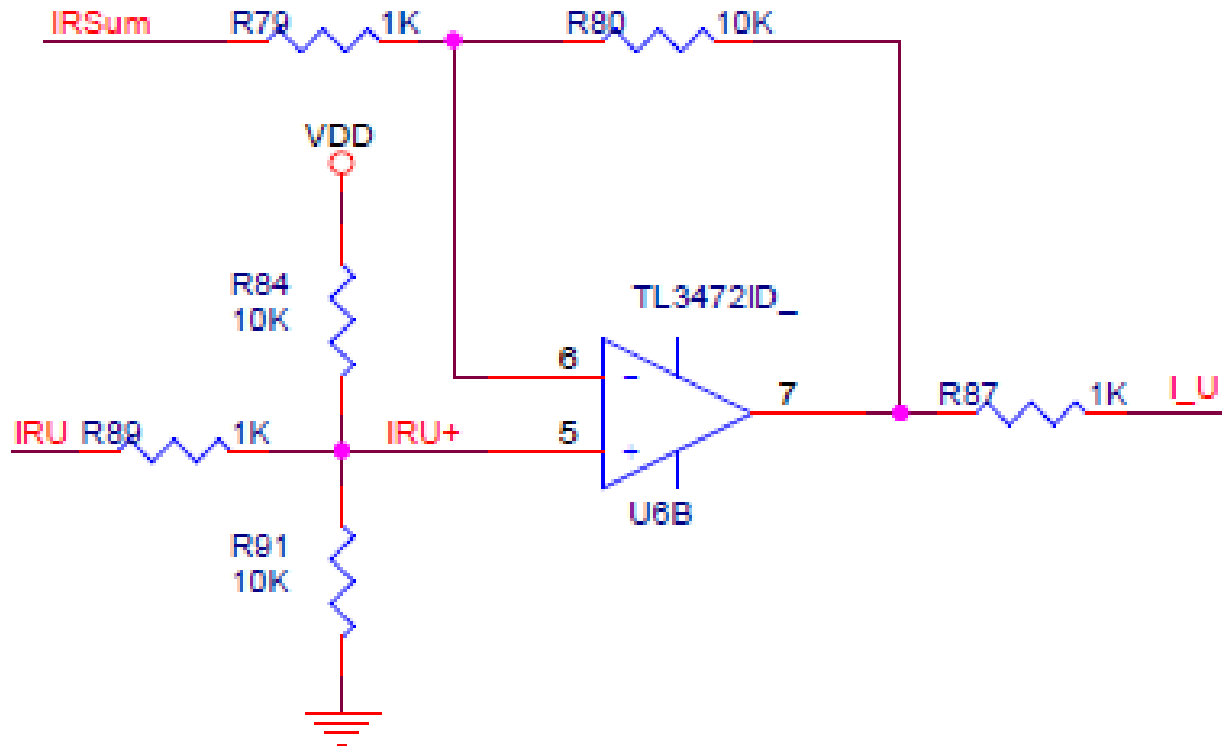
# Scooter control block diagram



# NM1244 with Driver Board

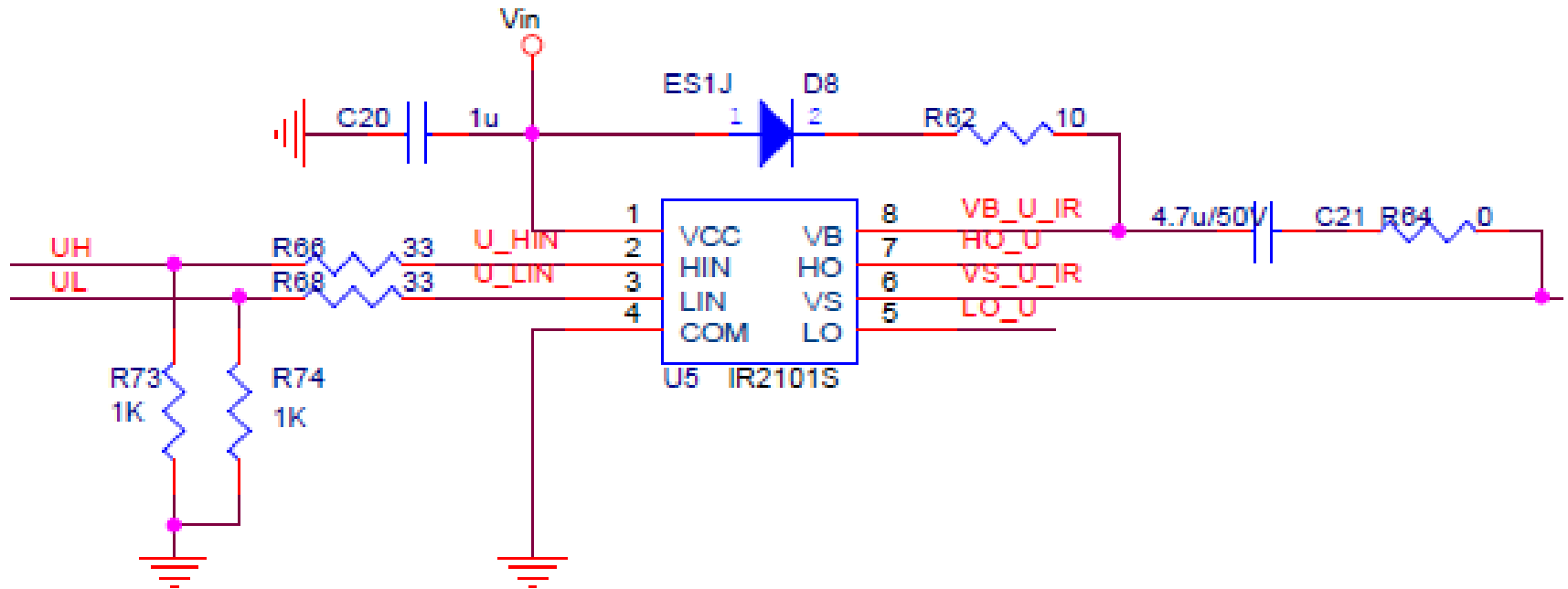


# HARDWARE(1)



Circuit for Current conversion into Voltage signal  
With Amplifier and level shift function

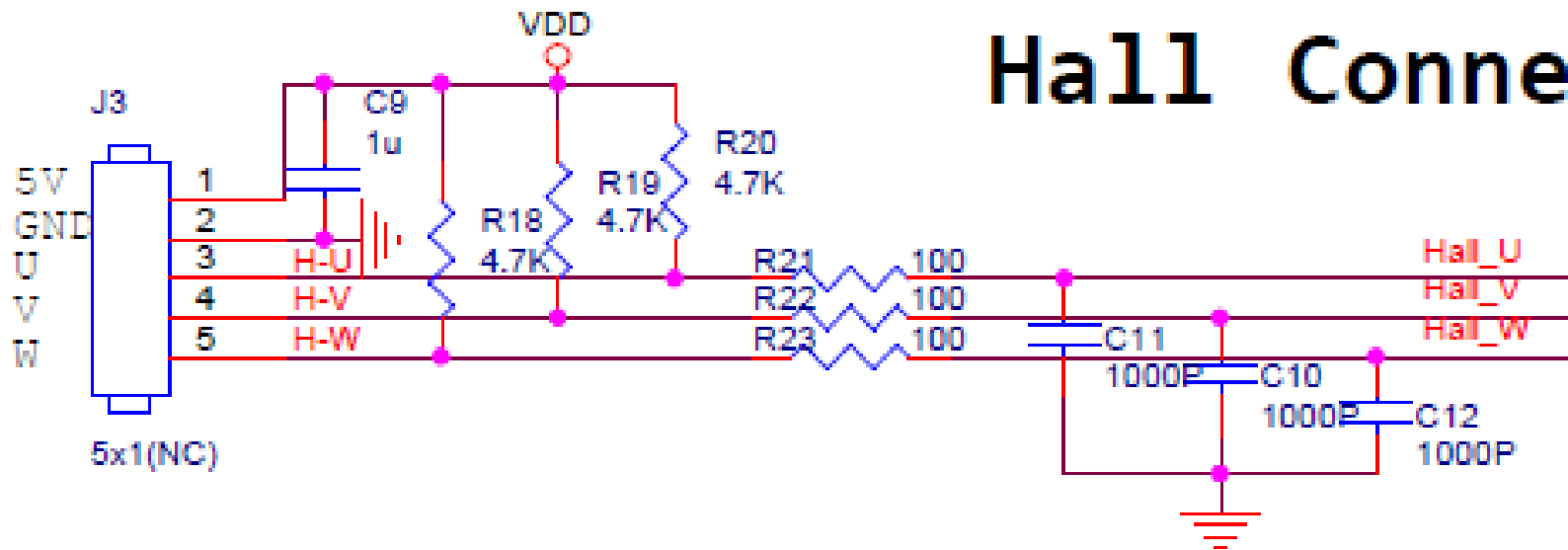
# HARDWARE(2)



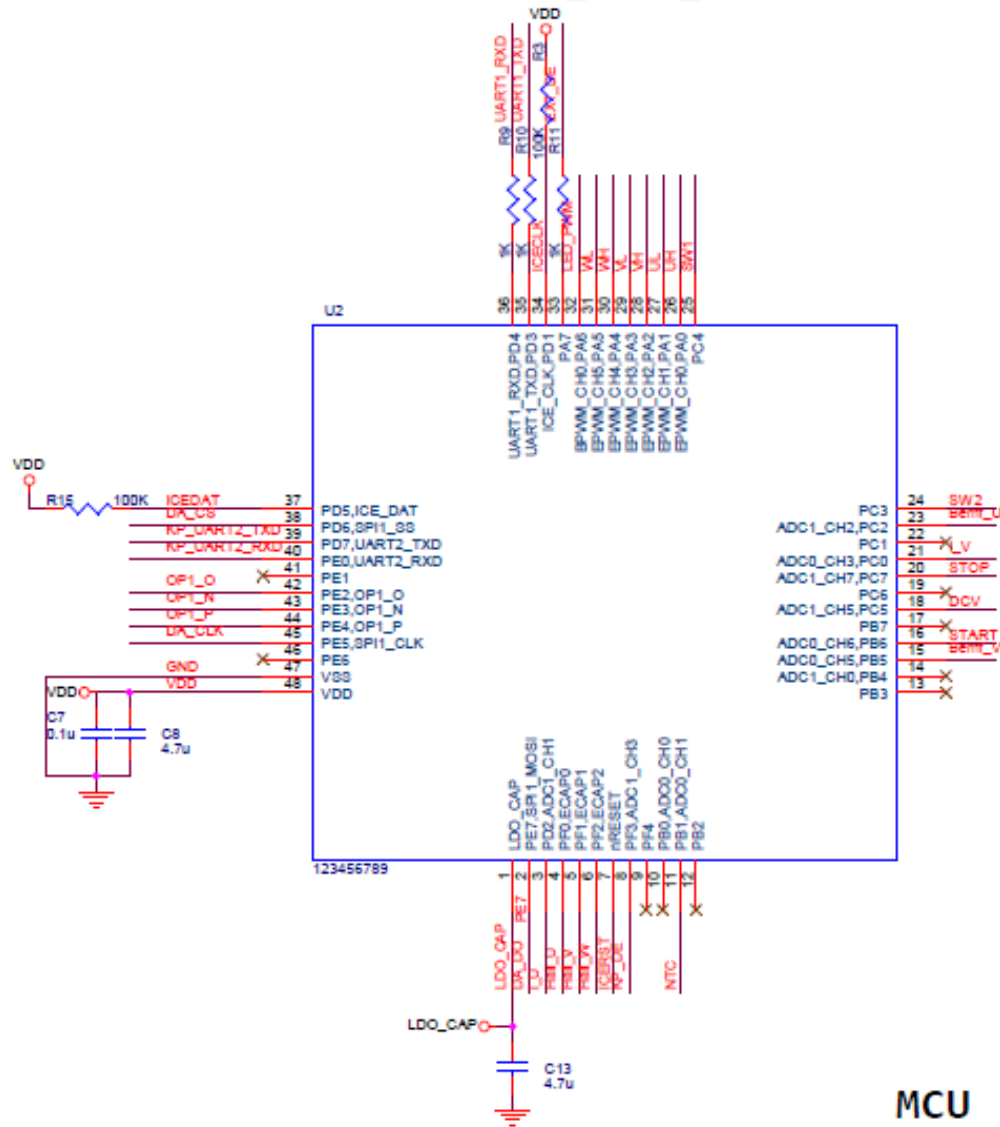
Gate driver circuit

# HARDWARE(3)

## Hall Connect



# HARDWARE(4)



MCU

# Pin Definition

## ▶ PWM output:

PA0 ->UH

PA1 ->UL

PA2 ->VH

PA3 ->VL

PA4 ->WH

PA5 ->WL

## ▶ Hall input:

PF0 ->Hall\_U

PF1 ->Hall\_V

PF2 ->Hall\_W

## • ADC input:

PB6 -> 油門

PC0 -> V相電流

PC5 -> DC Bus Voltage

PC7 -> 剎車

PD2 -> U相電流

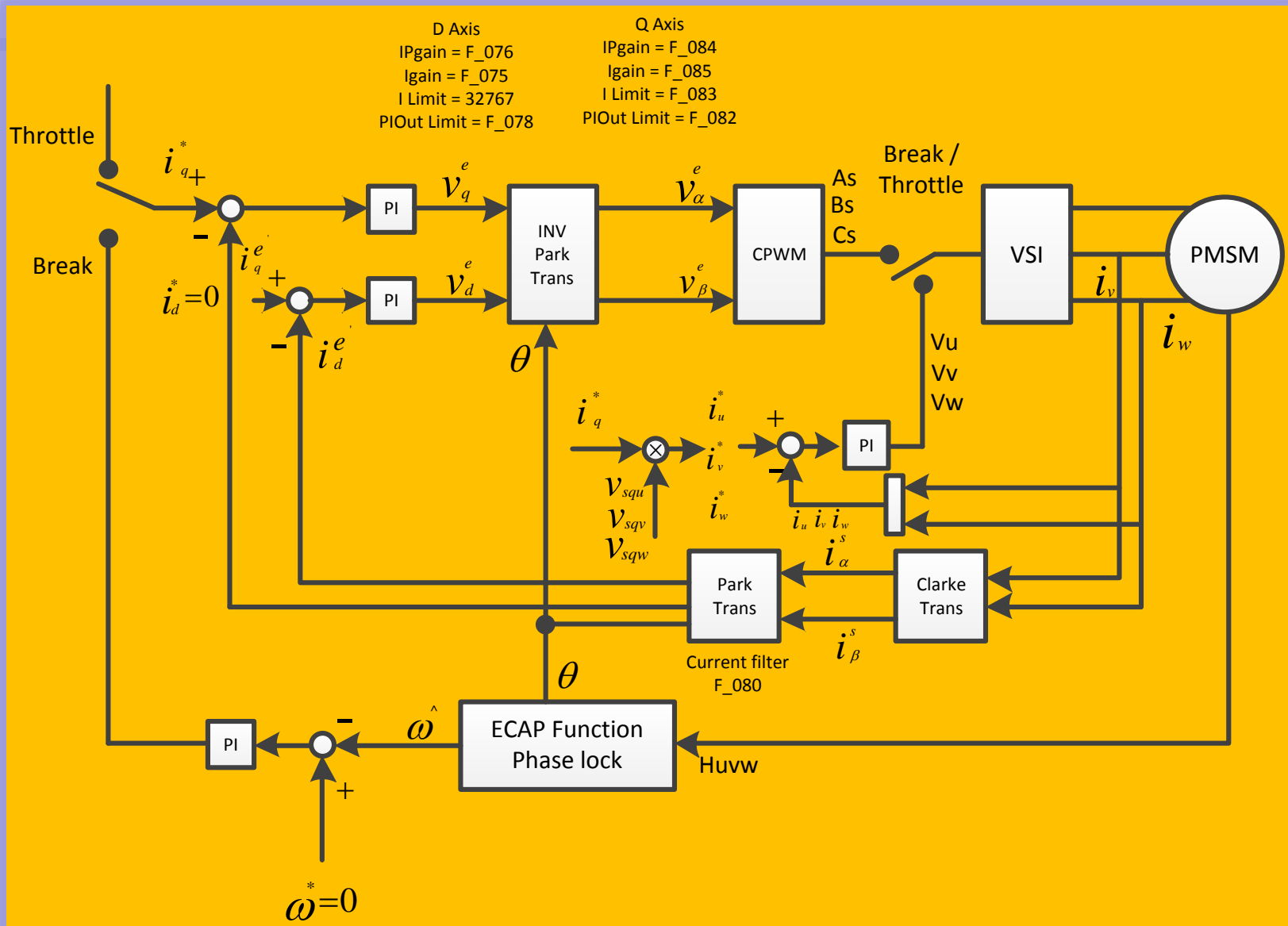
PB1-> NTC(Reserve)

PB5-> 反電勢 BEMF V

PC2->反電勢BEMF U



# Control Block of FOC with Hall



# Program file

| File Name         | Description  |
|-------------------|--|
| main.c            | 1.MCU 硬件层设定<br>2.参数矩阵  |
| Glb_reg_def.h     | 驱动器控制参数宣告  |
| SysTick_Handler.c | 1.DC-bus 解电压<br>2.速度环刹车功能<br>3. 刹车及油门处理  |
| SysTick_Handler.h | 控制变量宣告   |
| PWM_IRQHandler.c  | 1. 对Hall讯号锁相产生角度<br>2. 电流回授做框转换(Clarke and Park transform)<br>3. 三相独立电流控制<br>4. DQ轴电流PI控制<br>5. Inverse Park transform |
| PWM_IRQHandler.h  | 控制变量宣告   |
| Table.c           | 弦波表定义  |
| typedefine.h      | 变量属性定义   |

# Parameter in main.c

## 参数数组内容及如何设定说明

设定参数在main.c档案内有名为FUN\_Data\_Type const FATAB[MAX\_FUNC\_QTY+1]的矩阵中，参数共有120个，所有的设定应用参数都集中一起方便管理及提高效率，因为每个参数都有批注编号容易搜寻，没有用到的皆为保留项次，程序开始执行时会将设定值填入该参数此为Default，所以需要改变参数时，只要填入设定值然后经Keil C compile 在将档案load进 NM1244就完成，以下是以参数//F107作为范例说明

```
//F_107
```

```
{&FunPar.dbup_cnt ,0 ,255 ,B16|DOT0 ,30 ,0 ,0 ,0 ,0,0 ,0 ,0 },  
1 2 3 4 5 6~12
```

## 矩阵内容说明

| 1                | 2    | 3    | 4            | 5   | 6~12 保留 |
|------------------|------|------|--------------|-----|---------|
| 参数名称             | 参数下限 | 参数上限 | 参数长度         | 设定值 |         |
| &FunPar.dbup_cnt | 0    | 255  | B16 (16 bit) | 30  |         |

第五个字段即是设定值，填入欲改变的值在参数的上下限内经compile再load进NM1240就完成，参数变量宣告在Glb\_reg\_def.h内的结构名称PARAM内，若要修改变量名称要两边同时修改否则会出现错误。

| 参数No. | 参数名称           | 说明               | 下限     | 上限     | 設定值   |
|-------|----------------|------------------|--------|--------|-------|
| F_003 | First_cmd      | 油门增益             | 0.00   | 300.0  | 10.0  |
| F_004 | accdec_basef   | 保留无用             | 0.00   | 300.00 | 60.00 |
| F_005 | first_acctime  | 加速时间             | 0.0    | 3200.0 | 20.0  |
| F_006 | first_dectime  | 保留无用             | 0.0    | 3200.0 | 15.0  |
| F_015 | Sq_PGain       | 方波P Gain         | 0      | 4000   |       |
| F_016 | Sq_IGain       | 方波I Gain         | 0      | 4000   |       |
| F_017 | Sq_I_Limit     | 方波电流PI的积分器限制     | 0      | 4000   | 3276  |
| F_018 | q_Plout_Limit  | 方波电流PI的输出限制      | 0      | 4000   | 3276  |
| F_070 | Spd_I_Limit    | 速度回路积分器I限制*10    | 0      | 3276   | 3276  |
| F_071 | Spd_PI_Limit   | 速度回路PI Out限制*10  | 0      | 3276   | 2500  |
| F_072 | Spd_Igain      | 速度回路 I Gain      | 0      | 3276   | 15    |
| F_073 | Spd_Pgain      | 速度回路 P Gain      | 0      | 3276   | 20    |
| F_075 | Id_IGain       | D轴I Gain         | 0      | 250    | 100   |
| F_076 | Id_PGain       | D轴P Gain*10      | 0      | 4000   | 200   |
| F_077 | SpdFeedforward | 速度回路前馈补偿         | 0      | 8000   | 150   |
| F_078 | Id_PIOutLimit  | D轴PI OUT限制*10    | 0      | 3000   | 2649  |
| F_079 | Id_CurFilter   | D轴电流虑波           | 0      | 4096   | 2048  |
| F_080 | CurFilter      | Q轴电流回授Filter     | 0      | 4096   | 4000  |
| F_081 | FrqFbFilter    | 估测器回授频率Filter*10 | 1      | 2550   | 145   |
| F_082 | PI_OutLimit    | 电流PI输出限制*10      | 0      | 3276   | 2785  |
| F_083 | I_Limit        | 电流PI 积分限制        | 0      | 32767  | 32767 |
| F_084 | I_Gain         | 电流PI P Gain*10   | 0      | 3276   | 1776  |
| F_085 | P_Gain         | 电流PI I Gain      | 0      | 30000  | 1069  |
| F_086 | Iq_cmd         | (内部)Q轴电流命令*100   | 0      | 300    | 80    |
| F_087 | Id_cmd         | D轴电流命令           | -16384 | +16384 | 0     |

# SysTick\_Handler.c 函式功能说明

| 函式名称                         | 函式功能说明                      |
|------------------------------|-----------------------------|
| Drv_pro()                    | PWM 下臂导通对Bootstrap 电容充电时间处理 |
| Dc_Bus_Calculate()           | 计算Bus电压                     |
| Caculate_Motor_Speed()       | 计算电机转速                      |
| BreakSpeed_PI_Controller()   | 速度环PI运算，刹车产生                |
| spd_control()                | 判断命令来源为刹车或油门，刹车优先           |
| Speed_Pro()                  | 油门加速曲线                      |
| Judge_HallPatten_Direction() | 判断电机转向(正转或反转)               |
| Pwm_Process()                | 更新数据给PWM中断                  |

# EPWM\_IRQHandler.c 函式功能说明

| 函式名称                                       | 函式功能说明  |
|--|---|
| MotorControl_SqCmd()                       | 以ECAP取Hall周期做锁相产称 $\theta$                                    |
| CurrentRestructor_Process()                | 取AD转换的 $I_u, I_v$ 电流  |
| $I_u$ _CurrentPICalculate()                | U相电流PI运算  |
| $I_v$ _CurrentPICalculate()                | V相电流PI运算，结果再合成第三相   |
| ThreePhaseCurrentTransfor<br>ToD_qModel(); | 电流回授框转换，将 $I_u, I_v, I_w$ 转 $I_\alpha, I_\beta$ 再转 $I_d, I_q$ |
| Q_AxisCurrentPICalculate()                 | Q轴电流PI运算得到 $V_q$  |
| D_AxisCurrentPICalculate()                 | D轴电流PI运算得到 $V_d$  |
| InversParkTransfer_Process()               | 将 $V_d, V_q$ 做InversPark 得到 $V_u, V_v, V_w$                   |

# Table.c 函式功能说明

|                                  |          |
|----------------------------------|----------|
| u16 const Pwm_Sine2048_TAB[2048] | 2048点弦波表 |
| u16 const Pwm_Sine2048_TAB[1536] | 1536点弦波表 |
| u16 const Pwm_Sine2048_TAB[512]  | 512点弦波表  |

目前采用2048点，其他可以提供弹性应用



**nuvoTon**

**Thank You**

