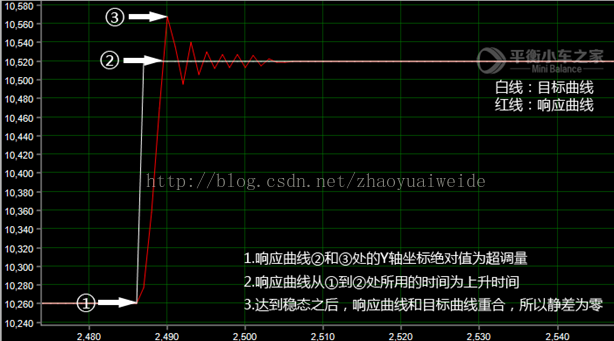
# 直流电机控制 pwm 和 pid 算法

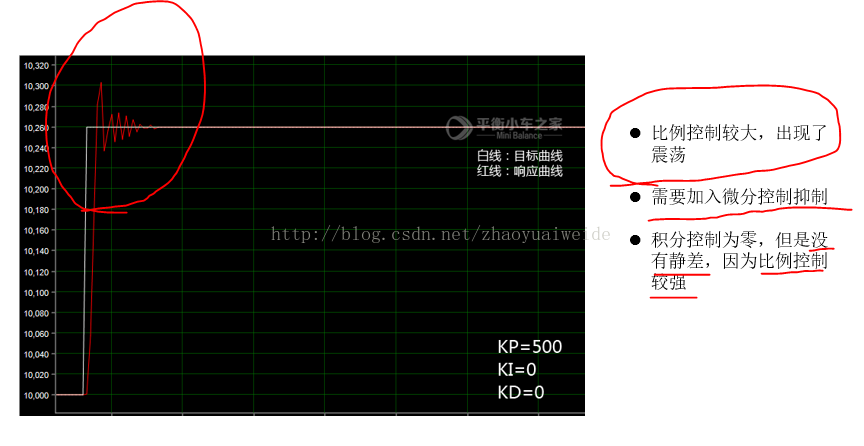
平衡小车和倒立摆都是用的位置式PID控制器  
/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
函数功能：位置式PID控制器  
1.入口参数：编码器测量位置信息，目标位置  
2.入口参数：编码器测量速度 , 目标速度  
返回 值：电机PWM  
根据位置式离散PID公式  
pwm=Kp\*e(k)+Ki\*∑e(k)+Kd[e（k）-e(k-1)]  
e(k)代表本次偏差  
e(k-1)代表上一次的偏差  
∑e(k)代表e(k)以及之前的偏差的累积和;其中k为1,2,,k;  
pwm代表输出  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
int Position\_PID (int Encoder,int Target)  
{  
static float Bias,Pwm,Integral\_bias,Last\_Bias;  
Bias=Encoder-Target; //计算偏差  
Integral\_bias+=Bias; //求出偏差的积分  
Pwm=Position\_KP\*Bias+Position\_KI\*Integral\_bias+Position\_KD\*(Bias-Last\_Bias); //位置式PID控制器  
Last\_Bias=Bias; //保存上一次偏差  
return Pwm; //增量输出  
}

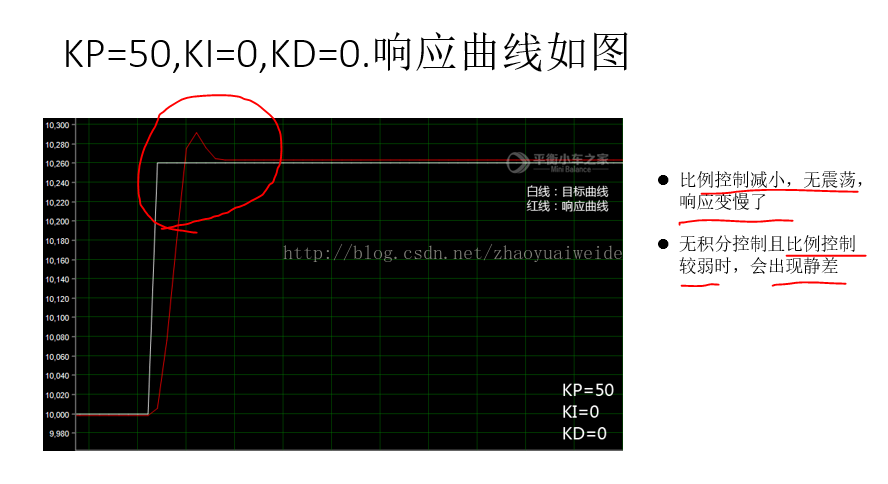
P:响应速度 过小响应速度太慢，基本不震荡；过大响应速度快，高频振荡；  
I:静差 当有静差的时候把I加大点即可消除  
d:稳态 减小最大超调量，该参数过大会低频振荡  
总结：：P用于提高响应速度、I用于减小静差、D用于抑制震荡。

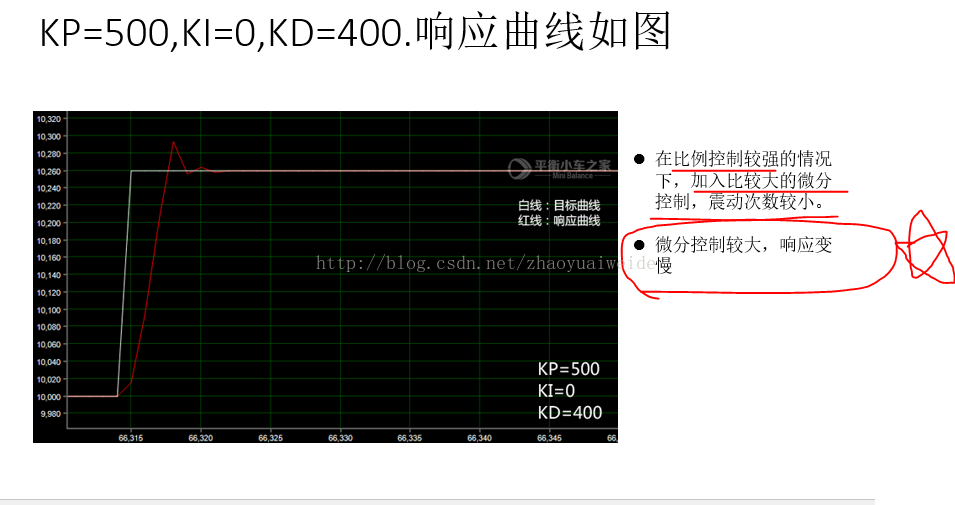
l最大超调量是响应曲线的最大峰值与稳态值的差，是评估系统稳定性的一个重要指标；  
l上升时间是指响应曲线从原始工作状态出发，第一次到达输出稳态值所需的时间，是评估系统快速性的一个重要指标；  
l静差是被控量的稳定值与给定值之差，一般用于衡量系统的准确性，

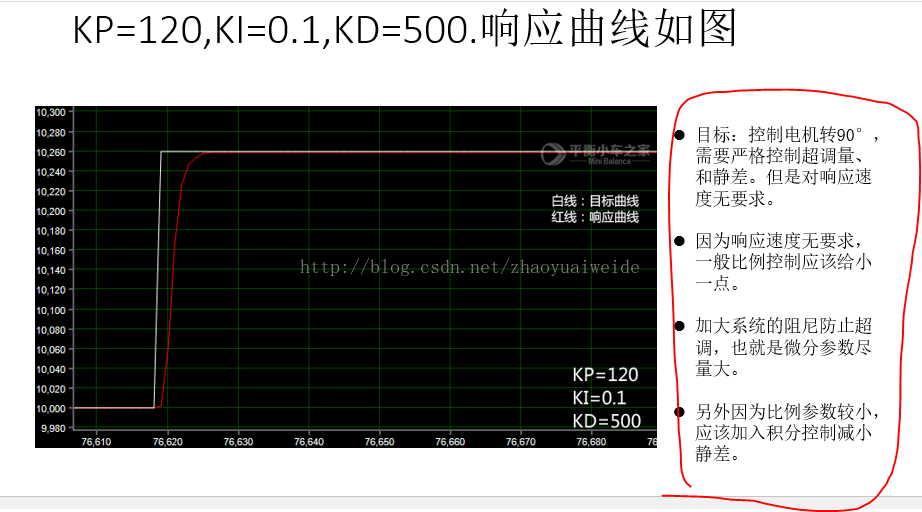
具体的参数调试过程：见位置pid参数控制整定如下：  
KP=500,KI=0,KD=0.响应曲线如图











以上是入口参数的位置的参数整定过程，下面说一说入口参数是速度的参数整定过程：

对于位置式的pid用于调试达到目标速度，就用PI控制器就足够了，因为如果加微分的话会有很大抖动，所以大家一般都用PI控制器来调试速度。  
具体参数整定原则：一开始先加大比例P，p小了会达不到目标速度，会差很多，所以再加大p，知道电机出现‘嗒嗒嗒’的抖动或者观察上位机的波形剧烈抖动的时候，这时候p就过大了，实际上p可以不是很大，比如400，调试时候增加幅值可以设为20，I稍微来点就可以，平衡车的工程经验是ki=kp/200，但是这里我给的是0.01，还要看具体情况，积分参数过大，实际速度和目标速度的静差会很大。  
---------------------   
作者：假如明天没有太阳   
来源：CSDN   
原文：https://blog.csdn.net/zhaoyuaiweide/article/details/54573676   
版权声明：本文为博主原创文章，转载请附上博文链接！