车模：





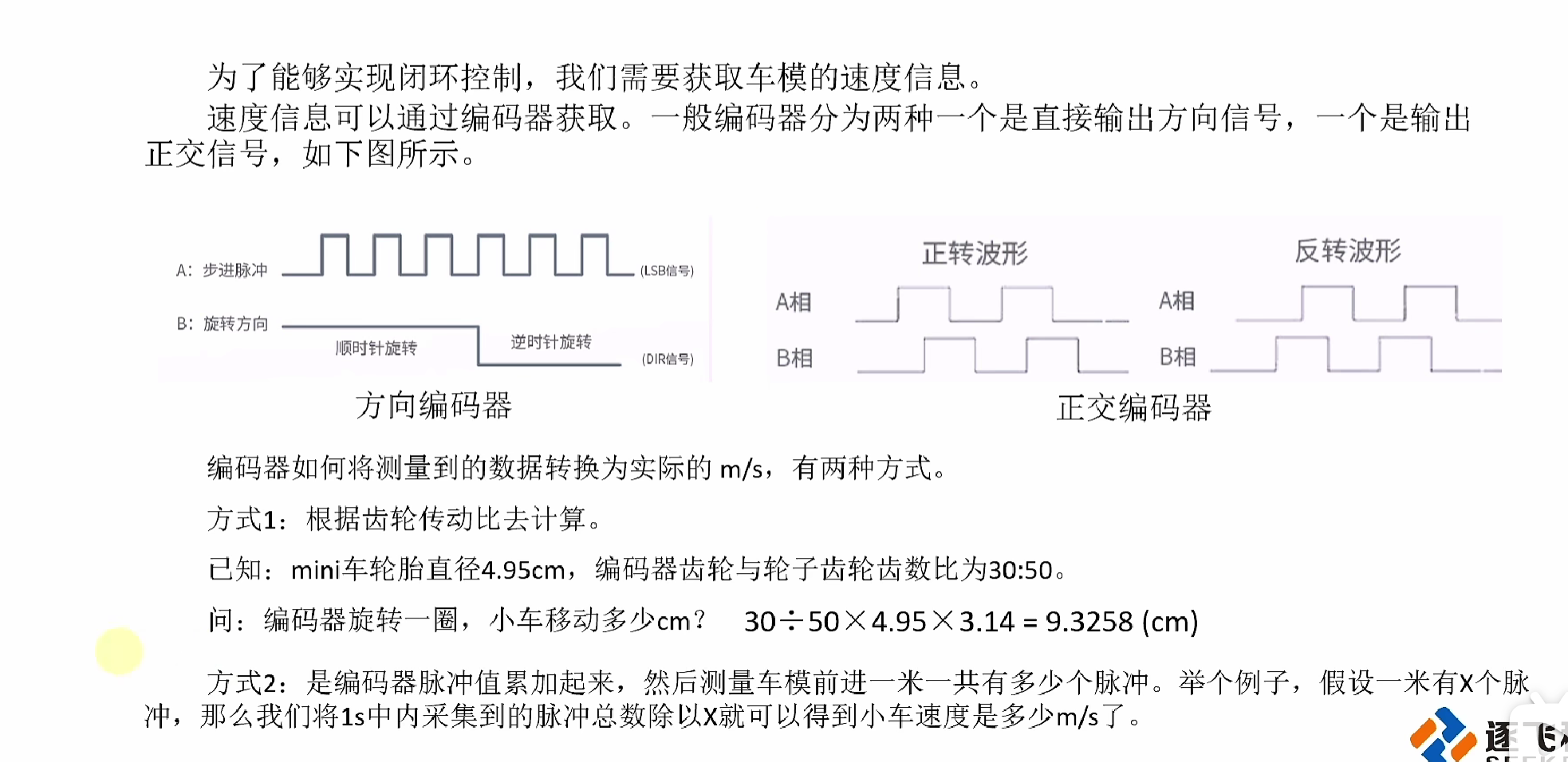
砖头路障区宽度30cm，车模宽度最好不超过20cm

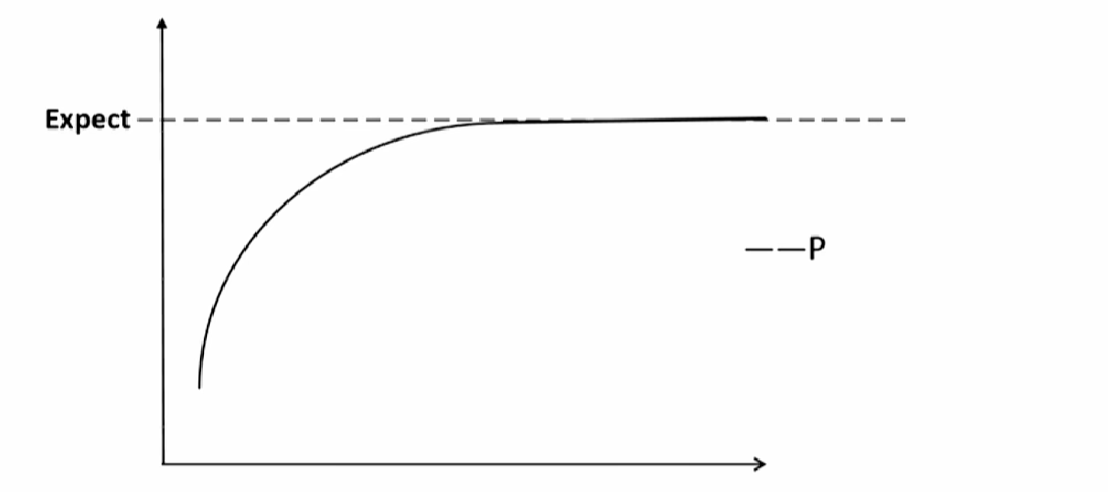
装配注意事项：较重元器件应当下沉，整体重心在小车中心，胎皮与轮毂之间封胶处理

速度闭环：

编码器获取速度信息，方向编码器，正交编码器

利用增量式PID闭环速度，增加后轮差速控制，增加转向灵活性，使用变速算法，实现弯道减速，直线加速；理想速度=最大速度-（最大速度-最小速度）\*abs(转角值/最大转角值)





转向控制：

使用陀螺仪增加转向稳定性，转角值=（当前图像偏差）\*KP+（当前图像偏差）\*abs（（当前图像偏差））\*KP2+（当前图像偏差-上一次图像偏差）\*KD+陀螺仪数值\*GKD

两个不同的转角D: 本幅图像偏差和上一幅图像偏差作为D量，这是最常用的一个D量，但这个D量的效果并不是完全是用来抑制抖动的，因为它本身还有预测的效果，这也就导致了调参上的困难。陀螺仪的角速度值来作为D量，它是实时反馈车身状态，效果单一更加适合抑制抖动。调试过程中反馈的效果也是陀螺仪的D量不容易出现超调震荡，并且有不错的抑制震荡效果，推荐在转向控制中加入陀螺仪。陀螺仪还能用于坡道检测，在上坡道时可以减速，避免飞坡的情况出现，下坡时进行加速，在安全的前提下最大限度的提高平均速度。

技术报告摘取

各模块的作用如下：

1．TC264主控模块：作为整个智能车的“首脑”，将采集到的图像信息，电感传感器、编码器等传感器的信号，根据控制算法做出控制决策，驱动直流电机和伺服电机完成对智能车的控制。

2．摄像头模块：是智能车的“眼睛”，可以通过一定的前瞻性，提前感知前方的赛道信息，为智能车的TC264主控模块做出决策提供必要的依据和充足的反应时间。

3. 传感器模块，为了辅助摄像头模块来感知赛道的信息为了使赛车更加稳定。

4．电源模块：为整个系统提供合适而又稳定的电源。

5．电机驱动模块：驱动直流电机和伺服电机完成智能车的加减速控制和差速转向控制。

6．速度检测模块：检测反馈智能车后轮的转速，用于速度的闭环控制。

7．辅助调试模块：主要用于智能车系统的功能调试、赛车状态监控等方面。

图像处理

图像采集：固定阈值法 平均阈值法 大津法

扫线 偏差计算 边界搜索 直弯道识别 圆环识别

赛道处理：左右环岛 三岔 十字 拐点 入库 出库 斑马线 圆环补线

经过图像分割后我们已经得到可以进一步处理的二值化图像。对于二值化图像处理部分主要分为：赛道边界提取，跳变点寻找，赛道中心线位置，赛道类型识别四部分。

**1.赛道边界提取**

对于赛道边界提取，最常用的方式为双边扫线，即指每行图像的起始扫线都是从上一行图像的中线位置开始的，从白点向左向右扫到黑点及寻找到所需要的边界线，进而拟合出赛道中心线。经过双边扫线得到左右边界后，我们需要进一步对得到的边界进行分析，譬如总计出左右两侧的边界丢失行数，边界连续性以及边界曲率方便我们对之后对于赛道类型的判断。

2**.跳变点寻找**

跳变点即边界突然变化的点，或者解释为赛道边界连续性的终点或起点。跳变点准确与否对于赛道类型判断是否准确起着决定性的作用。



图4.2.1十字拐点

如图4.2.1中红色圆圈即为十字拐点。我们将赛道拐点的提取分成两步。第一步对左右边界做连续性差分，不难看出在拐点处赛道将会出现过零。拿右边界下拐点举例，由下向上进行差，数值在拐点附近将会由正变负出现一个过零点，不妨称其为疑似拐点。第二步，取疑似拐点附近几行计算曲率当曲率超过一定阈值我们就将得到了赛道的真正拐点。同理可以推导出左拐点以及上拐点的计算方式。

3.**赛道中心线位置**

赛道中线位置即为我们期望小车在赛道上行驶的位置，所以我们就要考虑不同路况下如何保证拟合中线永远处于赛道中央位置。

1. 已知左右线求边线

在已知左右线的基础上的处理最为简单，仅仅取每行左右线的中心位置即可得到近似的赛道中心线。

1. 一边丢线的中线拟合

如果一侧的边线丢失，我们在最初的基础上使用最近一行能够找到的赛道宽

度作为之后丢线区域的赛道宽度，之后的丢线区域就将已知边线平移该宽度的

一半作为近似的赛道中心线。

1. 丢线不严重区域的中线拟合

我们首先将正常直道的赛道宽度保存在一个数组中，之后在丢线不严重的区域寻找丢线边线的恢复行，计算恢复行的宽度相较正常直道赛道宽度的比值，再丢线区域按照这个比值，平移已知边线，拟合出需要的赛道中心线。



图4.2.2赛道中线及边界

1.十字

十字识别一共分为三步，十字前，入十字，出十字

第一步：找到左右下拐点，并且左右边界均存在连续性丢失。此时我们取左右下拐点和其下方一定数量的点进行最小二乘法计算，得出左右边界的斜率Ka，截距Kb，根据公式Y=Ka\*X+Kb拟合出左右边界。



图4.3.1十字前

第二步：左右下拐点消失，并且图像底部出现边界连续性丢失。此时我们可以根据上拐点拟合出左右边界，也可以直接根据上拐点上方的中线拟合出全部中线。由于在十字处理过程中上拐点可信度并不高，存在边线错乱的情况所以此时我们常选用后者由部分中线去拟合全部中线，或者将中线强制赋值为图像中值。



图4.3.2入十字（中线拟合）



图4.3.3入十字（中线强打）

第三步：找到左右上拐点且左右边界连续丢边数很少，此时就可以判断已经驶出十字。这一步只需常规巡线，不再需要补线。



图4.3.4出十字

在十字扫线过程中由于中线的不稳定会导致上拐点的识别不稳定，所以在十字扫线前会根据上一帧图像下几行中线进行拟合得到预测中线，然后根据预测中线进行二次边界搜索，并对左右边界进行限幅。这种方式可以极大程度上保证十字过程中左右边界的准确性，斜入十字等问题也迎刃而解。

2.环岛

环岛处理一共分为五步，三步入环，两步出环

第一步：找到右下拐点且右边界连续性丢失，左边界连续且斜率小。此时根据右下拐点拟合出右边界。



图4.3.5入环岛一

第二步：右下拐点消失，但是右边界连续性丢失，左边界特征不变。此时我们根据左边界做赛道半宽平移得到中线。



图4.3.6入环岛二

第三步：找到右上拐点且行数超过一定阈值。将右上拐点与原左下边界一点进行直线斜率截距计算，从而拟合出新左边界。



图4.3.7入环岛三

第四步：找到左下拐点且左边界存在连续性丢失。此时我们可以根据左下拐点拟合出左边界，也可以将左下边界点与图像顶部边界点拟合，得到更加弯曲的左边界辅助小车快速出环。



图4.3.8出环岛一



图4.3.9出环岛一Plus





图4.3.10出环岛二

3.坡道

坡道元素的典型特征在摄像头上的体现就是突然出现非常多的非常规宽度道，这在上坡和下坡时会判断到两次。由于三轮车和直立车上均装有加速度计和陀螺仪，所以根据姿态解算结果在上下坡时车体运行角度会有很大变化，可以更加方便的判断坡道。由于车体在坡道上图像变得很不稳定，最终我们选用坡上切电磁控制转向。

4.出库

三轮出库有两种方式可以考虑，拉线出库和编码器里程出库。选用拉线出库就要考虑如何去除斑马线的干扰，我们的处理办法是使用腐蚀滤波将斑马线过滤掉，但是在出库过程中斑马线处于水平位置，常规腐蚀滤波并不能完全将其过滤，这就会影响摄像头元素识别，出线元素误判。最终我们选用编码器积分记录里程，并赋予其固定偏差让三轮走一个直角弯出库。

5.斑马线&&入库

以往斑马线识别采用的检测黑白跳变的方法，在车速较快情况会出现判断不及时的问题。这时我们可以考虑采用堆栈的方式，从图像数组一行的最左侧向右遍历，遇见黑色像素点，入栈，遇见白色像点，出栈并检验栈的长度是否符合斑马线宽度，如果符合就黑色块数量+1，当总数量满足我们设定的阈时，判断斑马线。在实际应用时，我们还用同样的方法计算了符合宽度的白色块的数量。实践证明，此方法判断成功率很高，并且几乎不会误判。

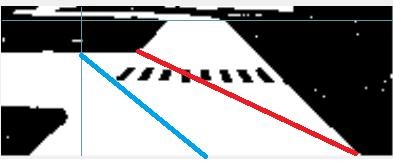


图4.3.11斑马线

在入库时我们可以采用图4.3.11利用拐点拉线的方式入库，也可以在识别到斑马线后给左右电机高差值PWM实现漂移入库，视觉效果极佳且速度比拉线入库快。

方向控制

基于位置式PID的方向控制

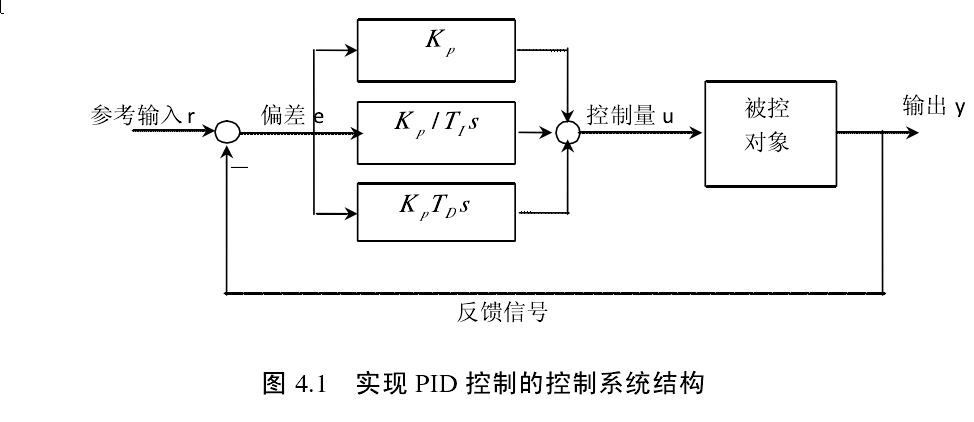
速度控制

基于增量式PID的速度控制

PID算法

### PID控制算法的介绍

作为工业过程控制中应用最广泛的一种控制器，PID即是将偏差的比例（Proportion）、积分（Integral）和微分（Differential）通过线性组合构成控制量， 用这一控制量对被控对象进行控制。PID 控制器结构简单、稳定性好、工作可靠、调节方便而成为工业控制的主要技术之一。当被控对象的结构和参数不能通过有效的测量手段来获取，即不能得到精确的数学模型时，系统控制器的结构和参数必须依靠现场调试和经验来确定，这时应用 PID 控制技术最为方便。PID控制流程图如下：



图表 23 PID流程图

PID算法公式： （式4-）

作为 PID 控制的输入，作为 PID 控制器的输出和被控对象的输入。式中：

—— 控制器的比例系数。

—— 控制器的积分时间常数。

—— 控制器的微分时间常数。

控制器各部分的作用：

* 比例部分：Kp\* e(t)

比例环节的作用是对偏差瞬间作出反应，及时成比例地反映控制系统的偏差信号，偏差一旦产生，控制器立即产生控制作用，以减少偏差。

* 积分部分：

积分环节的调节作用虽然会消除静态误差，但也会降低系统的响应速度，增加系统的超调量。

* 微分环节：

微分环节的作用是阻止偏差的变化。它是根据偏差的变化趋势（变化速度）进行控制。偏差变化的越快，微分控制器的输出就越大，并能在偏差值变大之前进行修正。微分作用的引入，将有助于减小超调量，克服振荡，使系统趋于稳定，特别对髙阶系统非常有利，加快系统的跟踪速度。

在实际过程控制中，并不是所有环节都适合引入控制系统。所以如何确定选用何种控制或控制组合来满足现场控制的需要显得十分重要。常用的各种控制规律和控制组合特点如下：

* 比例控制（P）: 比例作用是最基本、最主要也是最普遍的控制规律，它能较快地克服扰动的影响，使系统很快地稳定下来。
* 比例积分控制（PI）：积分能在比例的基础上消除余差，适合速度控制这样通道滞后较小、负荷变化不大的控制系统。
* 比例微分控制（PD）：微分具有超前作用，对于具有容量滞后的控制通道，引入微分参与控制，在微分项设置得当的情况下，对于提高系统的动态性能指标有着显著效果。常用于控制舵机打脚、寻迹闭环等场合。
* 比例积分微分控制（PID）：PID控制规律是一种较理想的控制规律，它在比例的基础上引入积分，可以消除余差，再加入微分作用， 又能提高系统的稳定性。

### 4.5.2 位置式PID

由于计算机控制是一种采样控制，它只能根据采样时刻的偏差计算控制量，而不能像模拟控制那样连续输出控制量，进行连续控制。所以（式4-2）中的积分项和微分项不能直接使用，必须进行离散化处理。离散后的PID公式如下：

（式4-）

其中： —— 采样序号， k ＝0，1，2，……；

—— 第次采样时刻的计算机输出值；

—— 次采样时刻输入的偏差值；

—— 第 k －1 次采样时刻输入的偏差值；

——积分系数， ；

——微分系数， ;

该式表示的控制算法是按PID 控制规律定义进行计算的，所以它给出了全部控制量的大小，因此被称为全量式或位置式 PID 控制算法。

位置式PID控制算法的缺点有：

由于全量输出，所以每次输出均与过去状态有关，计算时要对进行累加，工作量大；并且，因为计算机输出的对应的是执行机构的实际位置，如果计算机出现故障，输出的将大幅度变化，会造成很大的偏差。

### 4.5.3 增量式PID

所谓增量式 PID 是指数字控制器的输出只是控制量的增量 。当执行机构需要的控制量是增量，而不是位置量的绝对数值时，可以使用增量式 PID 控制算法进行控制。增量式PID控制算法可由位置式推导出。

由位置式PID公式（式 4－3）可以得到控制器的第 k－1 个采样时刻的输出值为：

（式4-）

将（式4-3）与（式（4-4）相减并整理即可得到增量式PID控制算法公式：

（式4-）

其中：

由增量式PID控制算法公式（式4-5）可知，如果控制系统采用恒定的采样周期T，一旦确定的控制参数KP、Ti、Td，A、B、C就确定了，只要使用前后三次测量的偏差值，就可以求出控制量，相比位置式PID计算量小，因此在计算机控制系统中较为常用。

上位机调试