代码讲解

主要的代码放置于Control.c之中，定义了PID的结构体:

typedef struct

{

float Kp;

float Ki;

float Kd;

int Err;

float LastErr;

float ErrSum;

float Summax;

float Summin;

float Max;

float Min;

float Pid\_out;

}PID;

**对于每个轮子：**

定义轮子相应的结构体WHEEL，成员变量的说明:

- uint8\_t Direction 当前轮子的方向，通过函数 void CarUpdate(ToyCar \* car) 进行更新

-int16\_t CurrentSpeed 当前轮子的速度，正向为正值，反向为负值，由编码器值赋值，最大值2199

- int16\_t DesireSpeed 设定的目标速度

- PID VelocityPID 该轮子的控制PID结构体

- int32\_t ControlPluse 控制该轮子PWM占空比的数值，最大值9999

typedef struct

{

uint8\_t Direction;

int16\_t CurrentSpeed;

int16\_t DesireSpeed;

PID VelocityPID;

int32\_t ControlPluse;

}WHEEL;

**对于整个小车系统：**

定义小车整体相应的结构体WHEEL，成员变量的说明:

-uint16\_t State 小车目前的状态，可取的值为Wheel\_Reverse/ Wheel\_Forward /Wheel\_Stop

- WHEEL xxxxxWheel 某轮子的成员

typedef struct

{

uint16\_t State;

WHEEL FrontLeftWheel;

WHEEL FrontRightWheel;

WHEEL RearLeftWheel;

WHEEL RearRightWheel;

}ToyCar;

```

控制的初始化只需创建变量\*\*ToyCar McNhamCar\*\* 并且调用函数\*\*void CarInit(ToyCar \* Car)\*\*，同时需要定时调用函数 \*\*void VelocityPidCallFnx(ToyCar \* Car)\*\* 如：将其加入定时器事件中断的回到函数中，可以得到PWM占空比的控制值，再调用\*\*void CarUpdate(ToyCar \* car)\*\* 就可控制小车的运动。

通过设定各个轮子的\*\*DesireSpeed\*\* 即可对小车实现控制。

**接下来我们讲解一下每个文件具体都有什么作用：**

**Control.c/h:**

主要是小车控制方面的程序，其中主要的函数为：

1. VelocityPidCallFnx(ToyCar \* Car)：设置各个轮子的速度，通过蓝牙以及串口改变Car->XXXWheel.DesireSpeed来改变轮子的速度，而具体怎么通过这个变量来改变速度则是PWM控制部分的函数完成。
2. CarInit(ToyCar \* Car)：小车初始化，里面包含了各种参数的初始值。
3. CarUpdate(ToyCar \* car)：小车状态更新，我们也主要是在完成相应操作后，通过这个函数来控制小车的。

**L298N.c/h:**

这个文件主要是电机模块的驱动程序，其中主要的函数为：

1. L298N\_GPIO\_INIT(void)：配置了电机与单片机的连接方式。我们就是通过这个文档来确定外部电路是怎么连接的。
2. L298N\_SetDirection(uint8\_t wheel, uint8\_t dir)：设置电机模块电流的输入方向以及是否输出，来控制各个轮子的转动方向。

**pid.c/h:**

这个文件是稍微底层一些的文件，更接近于对电路和器件的直接控制，是上面两个文件得以快速实现的基础。

1. PID\_Init(PID \*PID,float kp,float ki,float kd,float max,float min,float Summax,float Summin)：配置PID结构，将各个参数组建在一个结构体中，方便控制。
2. PID\_Cal(PID \*PID,float goal,float now\_val)：对1之外的其他PID成员进行操作的函数。
3. Motor\_SetPWM(uint8\_t wheel, int pluse)：对PWM输出的直接控制，也是后续在Control.c文件中速度控制的根基函数。

**USART2/3.c/h:**

这两个文件主要是串口通信相关的文件，可以不用深究，主要作用就是配置串口，形成串口通信的调节，我们并不会怎么使用这两个文件的函数。

这些文件都讲解完了，我们就讲解一下这些代码怎么相互配合最终驱动小车运行的：

1. 在main函数中，配置好了各个时钟以及通道，方便pwm输出来控制电机的速度和方向。
2. Main函数最终会执行到一个while（1）的循环语句，之后不再执行除中断外的其他程序，只有中断能再控制小车的状态。
3. 我们通过蓝牙，在手机上就能向单片机发送串口数据，单片机接收到串口数据后，进入串口中断，并处理串口数据。
4. 在串口中断程序中，处理好数据，最终调用Control.c文件中的函数来对小车进行控制，进而实现手机控制小车的功能。