待解决问题：

* 四驱小车直线走直线、转弯问题：查阅相关资料
* float、double存储格式
* 定时器不同通道可以设置为不同模式吗
* 超声波滤波？超声波在斜面不反射波形？测试超声波在不同角度斜面反射数值
* 使用WiFi或其他方式运行时，打印超声波等数据分析
* 小车位姿，平面是可以只考虑偏航角的改变，如四元数更新算法，一阶龙格库塔法
* 异常检测
* 1

待验证设想：

* 小车避障：记住小车遇到障碍物时的偏航角和舵机角度，通过转动超声波找到没有障碍物的点，记住角度，设定期望偏航角，然后根据转向直到达到期望偏航角

步骤：

开学前：目的是理解体会工作时编程思想，设计好框架，练习编程能力

1. 遥控板
2. 超声波
3. IMU，用串口打印角度信息，待之后用于视觉+IMU
4. 按面向对象的设计思想设计简单避障、遥控两个流程，可加以两个显示灯和蜂鸣器

开学后：目的是练习编程能力，加深对slam的理解

1. 在读完ORB-SLAM后，可考虑将其移植到树莓派，主要用于小车定位
2. 在读完okvis（多目+IMU）、VINS（单目+IMU）后，结合视觉和IMU

待完善（以后再做）：

* 编写控制界面，用串口打印数据方便分析

1. 自己总结

* 重要环节：
* 把整个系统作为一个结构体或函数，包含其他所有类和源文件，自己的变量、函数放在自己的源文件
* 系统参数的初始化，一些重要的标志位、变量、数组、列表、指针，其他可以使用默认初始化值
* 各源文件和类之间设置尽量少的接口，接口包括一些标志位、变量、函数
* 设置文件，一次读取完就可以不要了
* 一些共用的头文件、源文件，包括参数头文件，与设置文件有所不同，一个是读取、一个是编译
* 注意别名（引用）、指针的区别，指针自身是会占内存的
* 拿到一个新程序该怎么做：找主函数，理清各个类的用处，先看外层函数，再看内层，找出一些重要参数
* 姿态传感器对时间精度要求比较高，程序间隔4ms
* 1

1. C和C++对比，可拓展至python

* 将每个原文件作为一个类，C语言放在头文件变量、函数可看成public，使用回调函数调用或原文件内未在头文件申明的函数可看成private，仔细体会他们的应用

1. 使用引脚

开发板固有

* LED0,LED1-PF9、PF10
* 蜂鸣器-PF8
* 红外-PA8
* MPU6050-PB8、PB9、PC0
* 系统时钟：tim4

其他

* 超声波：PA0(TIM5\_1)、PF0

1. 原装烙铁线序：

对于手柄电路板，以弹簧（烙铁头端）朝上，接线面朝自己，

1-红-右上（中间点）；2-黄-左上；3-棕-烙铁头弹簧；4-绿-右下；5-白-左下

注意：1,2为温控；3为地；4、5为正负极，正负极接反会导致温度不断上升，指示灯长亮。

1. 程序设计步骤

* 根据外设模块确定源文件、头文件；设计程序整体框架，先用空函数结构体搭好，根据面向对象的思想（把小车看成一个整体，各零件作为小车component），根据具体功能（前进、后退）设计执行函数；画流程图
* 完善执行函数

1. 流程

* 初始化流程
* 手动流程
* 异常流程

1. 源程序（模块）

* common\_defines.h：共用定义头文件，程序共享的结构体，如开关枚举
* sfr.h：special function register，定义板上外设
* parameter.h：参数头文件，机器人主要参数
* share\_defines.h：共享头文件，与树莓派共享头文件，方便协作
* flow.c：流程模块，控制流程及切换函数
* system.c：系统模块，包括初始化参数等函数
* control.c：控制模块，直接控制机器人电机、舵机等
* bsp.c：外设模块，直接操作板上外设，如：定时器、gpio等
*  isr.c：中断服务函数模块
*  mpu6050.c：姿态传感器模块，imu
*  sensor.c：传感器模块，超声波
* abnormal.c：异常检测模块，处理机器运行过程中异常
*  cmd.c：命令模块，主要用于解析WiFi命令
*  common.c：共用模块，如一些通用asc-十进制转换函数

 period.c：不同周期检测模块

 flash.c：flash模块，保存、读取一些设置数据

 data\_debug.c：调试数据模块，后期添加，为了方便调试

spi\_ex.c：spi通信模块

WiFi.c：WiFi模块，使用单片机WiFi模块

 wire.c：有线模块

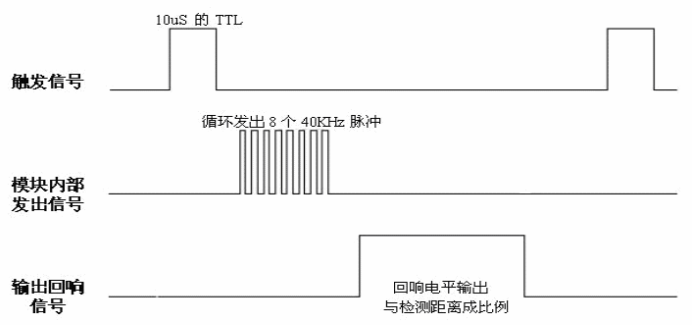
1. 问题总结

* 头文件不能相互包含，相互包含有一个得放在.c文件中，头文件开头#ifndef \_\_AAA\_H
* switch（表达式不能用枚举），但可用case 枚举：
* 实时更新调试数据：mdk下硬件仿真时，将View -> periodic windows Update 选项勾上即可
* 超声波信号线接触不良，造成偶尔才能读出数据；超声波需要障碍物的面积在0.5平方米左右，后面考虑在超声波左右两个面加挡板
* 注意输入输出上下拉情况
* 注意设计测试实验测试某项是否正常，如测试系统时钟、超声波测距是否准确
* TIM2\_CH2(PA1)做定时器使用有问题
* 用ms改变标志位，用10us验证结果出错

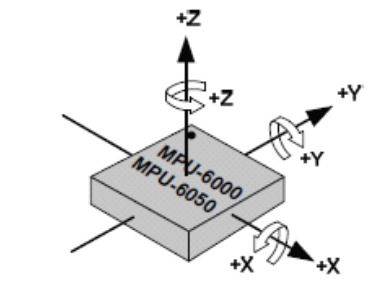
1. 红外遥控：接收波形如下图，其为遥控器发射出的波形已经固定，通过定时器输入捕获PA8正脉冲宽度来实现解码。具体见《第62讲 红外遥控器实验-M4.ppt》



1. Flash的理解：float的存取，float以uint32\_t存储，再以uint32\_t读出，再转化成float
2. 超声波：电压5V，控制端（trig）、接收端（ECHO）波形如下图，测量具体2cm-4m，计算公式：距离=脉宽\*340（m/s）/2,感应角度15度，最高精度0.3cm



1. 舵机：电压5V，正电平脉宽0.5-2.5ms,低电平脉宽0.5-30ms（我选2.5ms），正电平0.5-2.3ms对应0-180度，最好留些余量防止卡住
2. mpu6050：点在靠近开关的角，各轴如下图



1. 1