



西安交通大学
XI'AN JIAOTONG UNIVERSITY

课程：计算机控制实验

实验项目：实验三 声控小车+红外/超声避障

学生：郝世龙 学号：2171311028 班级：自动化 71

学生：田瑞峰 学号：2171311105 班级：自动化 71

学生：任泽华 学号：2171411498 班级：自动化 71

指导老师：景洲 联系方式：13720424008

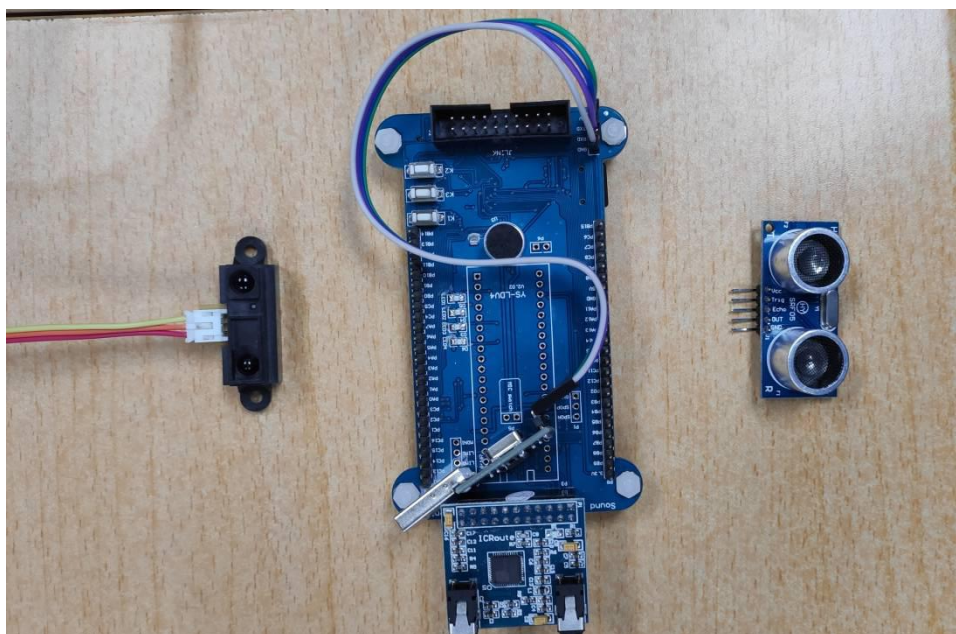
实验地点：西一楼 A304

实验时间：2020/7/15

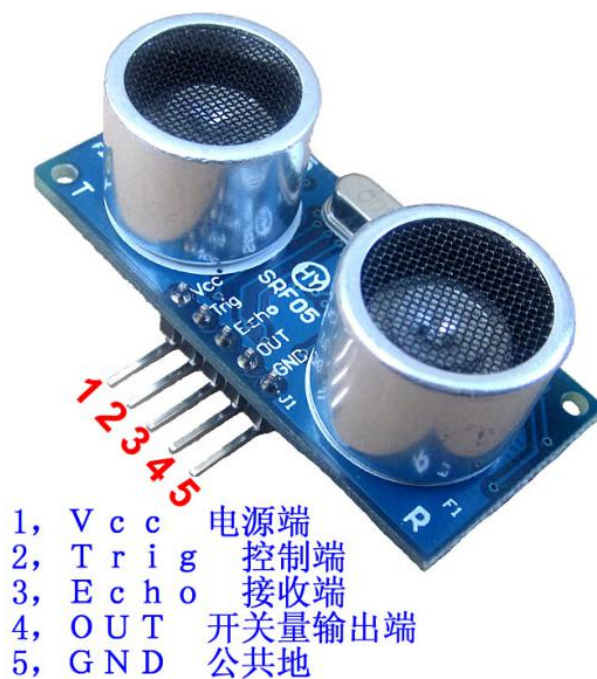
报告提交时间：2020/7/19

一、硬件框架及原理

在本次实验中，我们使用到的三个传感器分别为：语音识别模块、超声波传感器模块、远红外传感器，三个传感器如下图所示，其中语音识别模块是由上方方形板的 LD3320A 语音识别芯片配合 STM32F103 上位机构成。



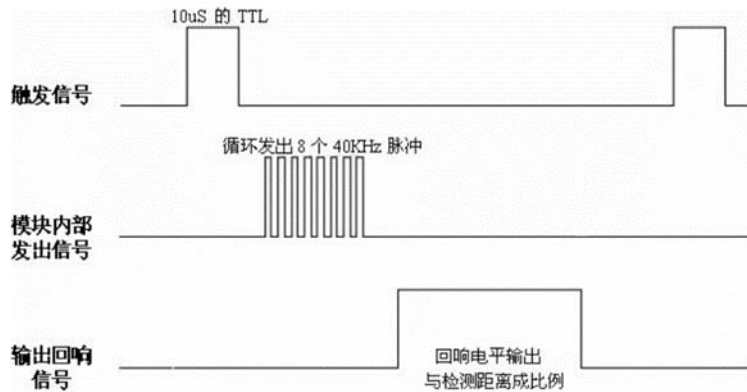
1. 超声



HC-SR04 超声波测距模块可提供 2cm-400cm 的非接触式距离感测功能，测距精度可达高到 3mm；模块包括超声波发射器、接收器与控制电路。像智能小车

的测距以及转向，或是一些项目中，常常会用到。智能小车测距可以及时发现前方的障碍物，使智能小车可以及时转向，避开障碍物。

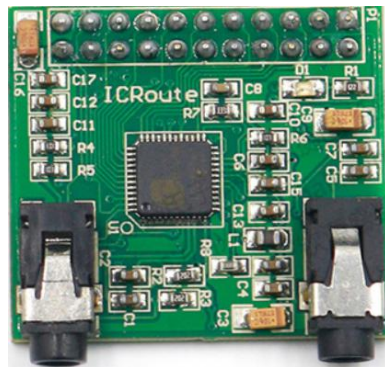
- (1)、采用 IO 口 TRIG 触发测距，给至少 10us 的高电平信号；
 - (2)、模块自动发送 8 个 40khz 的方波，自动检测是否有信号返回；
 - (3)、有信号返回，通过 IO 口 ECHO 输出一个高电平，高电平持续的时间就是超声波从发射到返回的时间。测试距离= $(\text{高电平} \times \text{声速} (340\text{m/s}) / 2)$ ；
- 工作原理如下图所示：



但实际上由于我们需要在小车工作的时候实时测量小车到墙壁的距离，所以我们需要一个能够一段时间就能给出一段触发信号的 TRIG 输入源，我们选择使用 PWM 波进行驱动。

2. 语音

语音模块我们使用的是 LD3320A 语音识别芯片配合 STM32F103 单片机进行实现，通过识别语音内容并在单片机引脚上做不同输出来使得 myRIO 上位机得到信息。



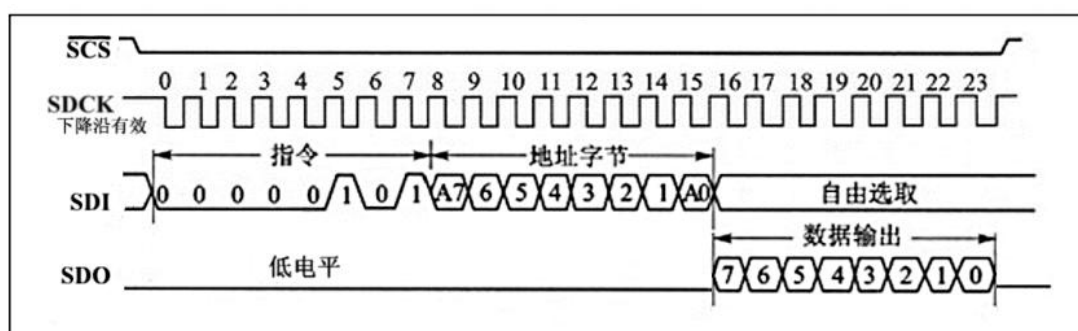
LD3320A 是一颗基于非特定人语音识别 (Speaker-Independent Automatic Speech Recognition) 技术的语音识别/声控芯片，提供了真正的单芯片语音识别

解决方案。LD3320 芯片上集成了高精度的 A/D 和 D/A 接口，不再需要外接辅助的 Flash 和 RAM，即可以实现语音识别/声控/人机对话功能。并且，识别的关键词语列表是可以动态编辑的。

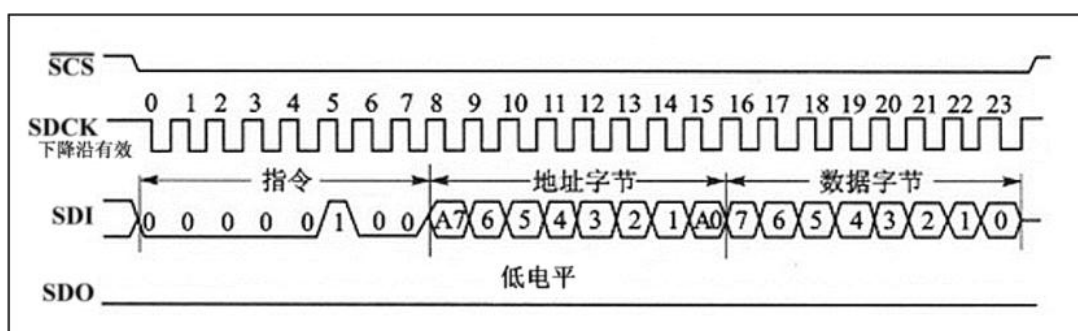
其支持用户自由编辑 50 条关键词语条：在同一时刻，最多在 50 条关键词语中进行识别，终端用户可以根据场景需要，随时编辑和更新这 50 条关键词语的内容。并且芯片的识别关键词语列表是可以动态编辑的：只需要把识别的关键词语以字符串的形式传送进芯片，即可以在下次识别中立即生效。

该芯片采用 SPI 总线通信协议，可以通过该总线通信协议进行与上位机的 SPI 串口通信，其通信时序图如下：

读时序：



写时序：



我们使用 STM32F103 单片机作为上位机进行与模块的通信。STM32F103 是一款由意法半导体 (ST) 出品的 Cortex-M3 内核的单片机，属于中低端的 32 位 ARM 芯片。在工业控制、电力电子等领域中均有广泛应用。

我们在写入 STM32 单片机的程序中可以定义 LD3320A 的 50 条关键词语条以及当语音触发相关语条的时候进行一系列操作。我们使得在触发某语条的时候定义好的端口分别置高置低，便可以让其输出特定的数字编码。为了便于调试，我

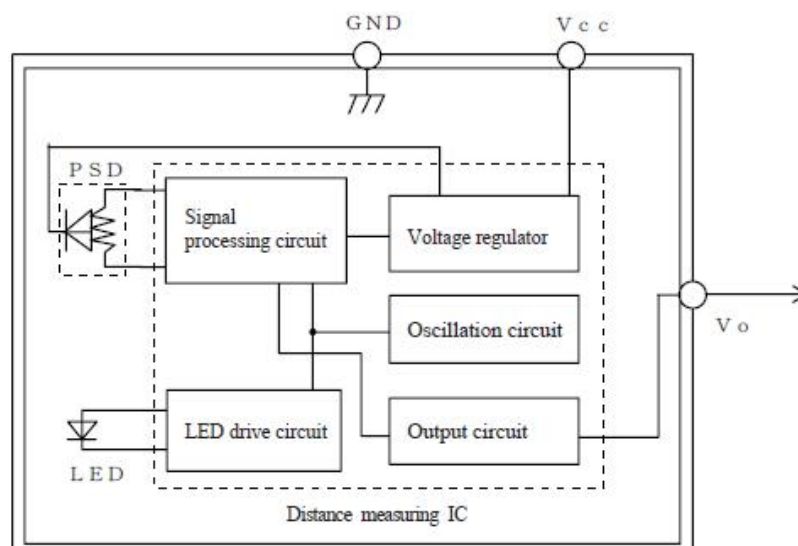
们也同时使语条的触发与串口发送关联起来，从而便于调试。调试使用的串口工具为 XCOM 2.0。

3. 红外距离传感器 GP2Y0A21YK0F

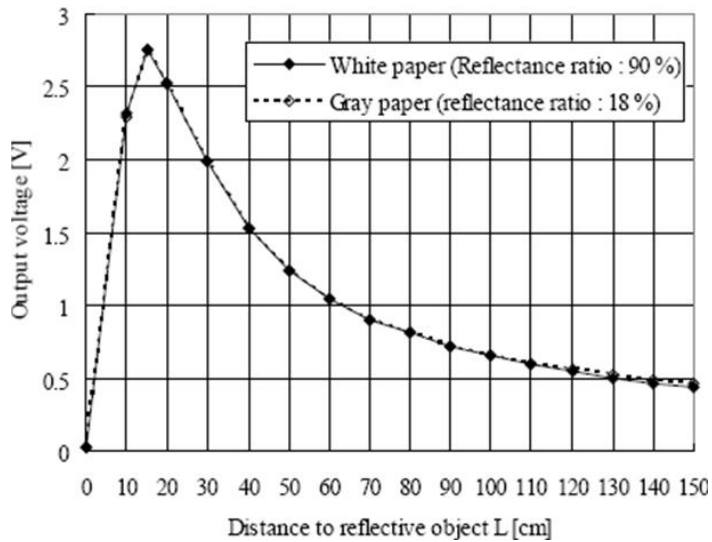


GP2Y0A21YK0F 是一种距离测量传感器单元，由 PSD（位置灵敏探测器）、IRED（红外发光二极管）和信号处理电路集成组成。由于采用了三角测量法，物体的反射率、环境温度和工作持续时间的变化不易对距离检测产生影响。该装置输出对应于检测距离的电压。因此，该传感器也可以用作接近传感器。该款传感器信号输出类型为电压模拟量，且与测量距离的输出关系呈类反比的非线性关系。

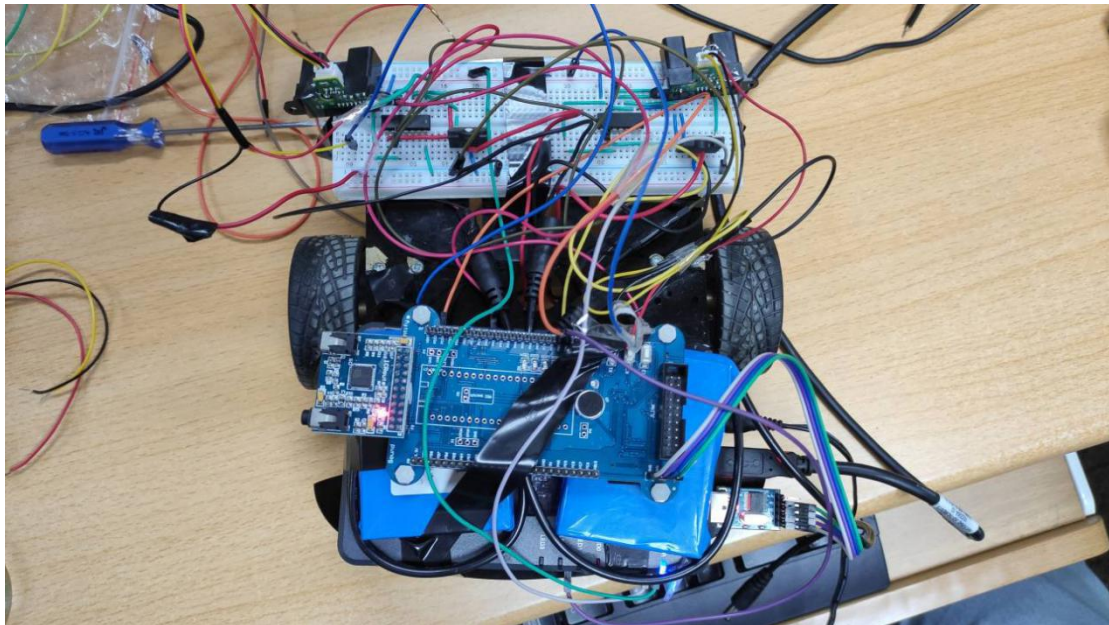
其内部原理图如下：



其输出的电压模拟量随测量距离的变化关系如图所示：



4. 连线图与说明



将语音模块放置在 myRIO 上，将红外或超声距离传感器粘贴或插在 小车前端的面包板上，实现对前进路上障碍物的检测和对于语音指令的响应。

连线上，语音模块程序上设计的是 PB1 与 PB2 的输出表明识别状态，将 PB1 与 PB2 接到程序中设计好的相应端口即可（B / DIO0 与 C / DIO0）因为识别的状态只有“前进”“左转”“右转”，所以编码上完全是足够的。

传感器模块连接到 myRIO 各个端口即可，超声波模块 TRIG 端连接 PWM 输出，ECHO 端口连接普通的数字端口 DIO 端口即可。对于红外传感器，由于电压输出是模拟量，所以连接 myRIO 的模拟端口 AIO 即可。

二、LabVIEW 设计过程

第三次实验我们首先使用了语音模块和超声波传感器。一方面，我们允许操作者语音控制小车的行驶状态；另一方面，我们使用超声波传感器探测障碍，在语音操作出现失误时自动避障。

但由于在实际使用过程中，使用的超声波传感器测距的方式不够精确（在实验后复盘发现有程序设计的问题），在实际的避障过程中虽然会有所避障，但有些时候会撞墙，并且在滤波后发现传感器数据仍然跳动很大，所以我们换用了远红外传感器实现了较为理想的效果。由于在两个传感器的实验中都有较大的投入和比较明显的成果以及经验教训，所以我们对两种方式均会进行详细的分析。

1. 原理分析：

1.1 语音模块

小车相关运动逻辑仍然不变，我们在此仅分析传感器接收信号与小车相关状态变化之间的关系。需要强调的是，我们在下文说到的方向，无特殊说明均指正向的方向。

对于超声波模块和语音模块的配合，我们设计为当超声波模块没有发现障碍时，使用语音模块对小车进行控制，而当超声波传感器发现障碍物时，小车被另一套控制逻辑接管，不再受语音模块控制。

对于单独的语音模块，由于超声波传感器只能探测某一方向的路障，且超声波传感器装在车尾，只能探测后向障碍。所以从功能上看，语音模块控制中只允许有后退、左转、右转、停车四种逻辑。但是为了防止停车状态下，语音模块误触发导致小车运动，所以我们选择手动将速度归零用于进入停车状态。故最终，我们仅使用语音为小车加入后退、左转、右转三种逻辑控制，但是语音操控过程中为了方便起见，我们的指令为“前进”“左转”“右转”，分别对应后退，右转，左转。

1.2 超声模块

对于超声波模块，我们是基于避障的目的将其引入。所以在控制设计中，若是左侧超声波模块检测到障碍，则说明障碍靠近车左侧，右转避让；若是右侧超声波模块检测到障碍，则说明障碍靠近车右侧，左转避让。两侧超声波均检测到

障碍时，无法转向避让，直线退避。

1.3 红外模块

对于红外模块，由于在前期使用超声波测距模块的时候测距噪声大且不是很准确，所以我们采用远红外模块进行测距。由于我们是基于避障的目的将其引入，所以在控制设计中，原理上和超声波传感器的控制原理相同。若是左侧红外模块检测到障碍，则说明障碍靠近车左侧，右转避让；若是右侧红外模块检测到障碍，则说明障碍靠近车右侧，左转避让。两侧红外传感器均检测到障碍时，无法转向避让，直线退避。

2. 控制逻辑：

首先，为了方便解释，我们将给定的 PWM 占空比大小直接等价于速度。

在控制说明中，若未特定说明，左转、右转均指沿正方向观察意义下的左右，而不是沿运动方向观察的左转、右转。

2.1 传感器输出逻辑

在控制逻辑中，我们先对超声波传感器的状态进行判断，我们设传感器测得到前方障碍距离小于阈值时输出为 T。那么，我们对两超声波传感器进行或非运算，结果为 T 时说明两传感器均未检测到障碍，小车受语音控制。否则结果为 F，说明至少有一个传感器检测到障碍，语音控制失效，需要进行避障。

在对超声波传感器进行实验之后，我们基本实现了避障功能，但实际上的避障功能效果不是很好，所以我们在后面的实现中采用了红外线传感器。红外线传感器的工作原理以及输出的逻辑与超声波类似，不同的是由于远红外在距离较远（我们需要用到的距离范围）内，距离越远输出电压越小，所以传感器其测得前方障碍距离反映的电压值大于阈值时，输出为 T，反之为 F。

2.2 传感器结果处理以及小车动作控制

那么，当或非运算结果为 T 时，我们设计语音控制逻辑。令单片机语音模块检测语音后，输出 2 位 01 变量用于控制小车运动状态。此时，有关 01 状态对小车状态的控制同 xunxian2。当我们说“前进”时，单片机输出为 11，我们令小车直线后退，两轮轮速均为 v_0 。当我们说“左转”时，单片机输出为 10，我们令小车左

轮由原速度 v_0 变速为 v_1 ，右轮速度归 0，小车右转（沿后退方向看为左转）。当我们说“右转”时，单片机输出为 01，我们令小车右轮由原速度 v_0 变速为 v_1 ，左轮速度归 0，小车主转（沿后退方向看为右转）。

当运算结果为 F 时，说明超声波传感器发现障碍，需要避障。我们此部分继承了 xunxian2 程序中的精简逻辑来控制小车。当仅有左侧传感器数值为 T，即检测到左侧障碍时，小车主轮速度为 0；左轮由原速度 v_0 变速为 v_1 ，小车整体右转。当仅有右侧传感器数值为 T，即检测到右侧障碍时，小车主轮速度为 0；右轮由原速度 v_0 变速为 v_1 ，小车整体左转。而当两传感器同时数值为 T，也就是左右传感器同时检测到障碍，我们认为无法避障。此时在直线前进 v_0 的基础上，我们将 forward 逻辑取反进行倒车，直至远离障碍，此后再用语音控制小车绕过/远离障碍（实际上由于小车走的路线不会与障碍绝对垂直，并且小车的直线不是绝对标准，所以可以自己实现拐弯避开障碍）。

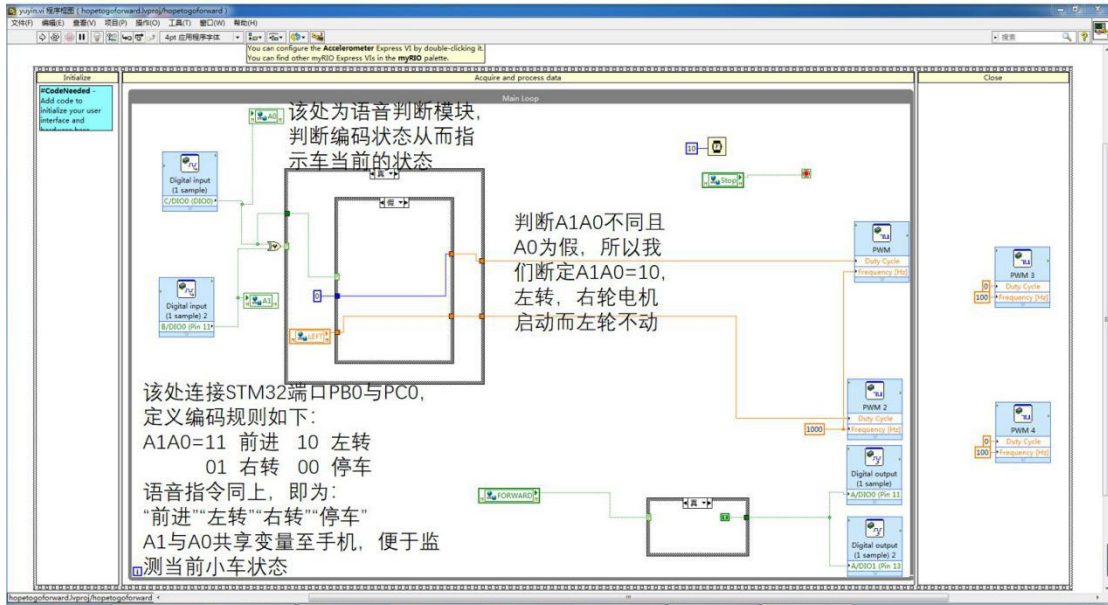
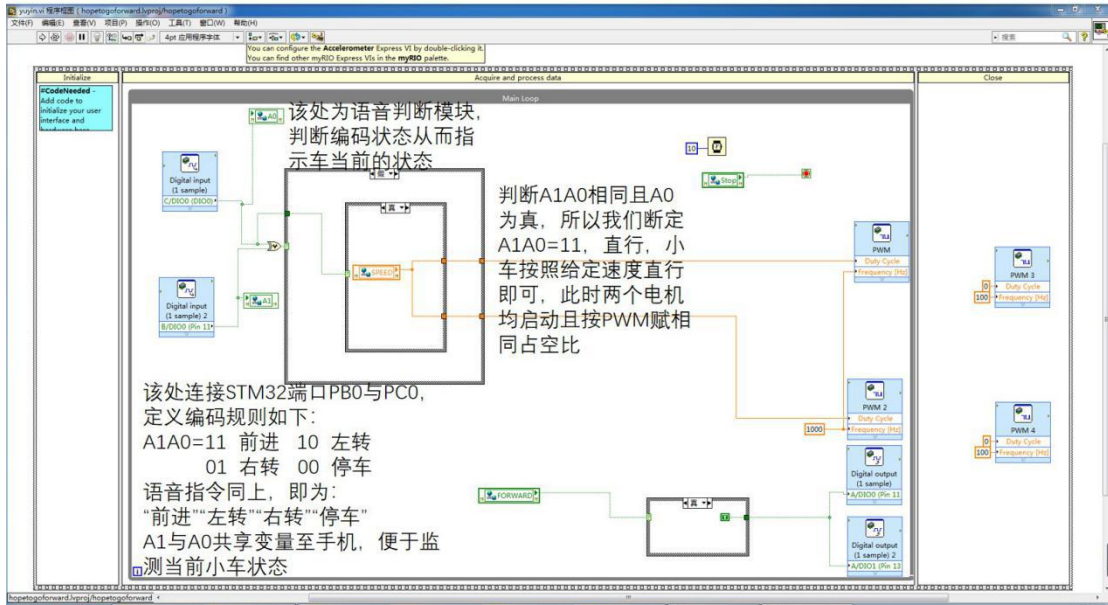
我们用真值表进行表示，结果如下：

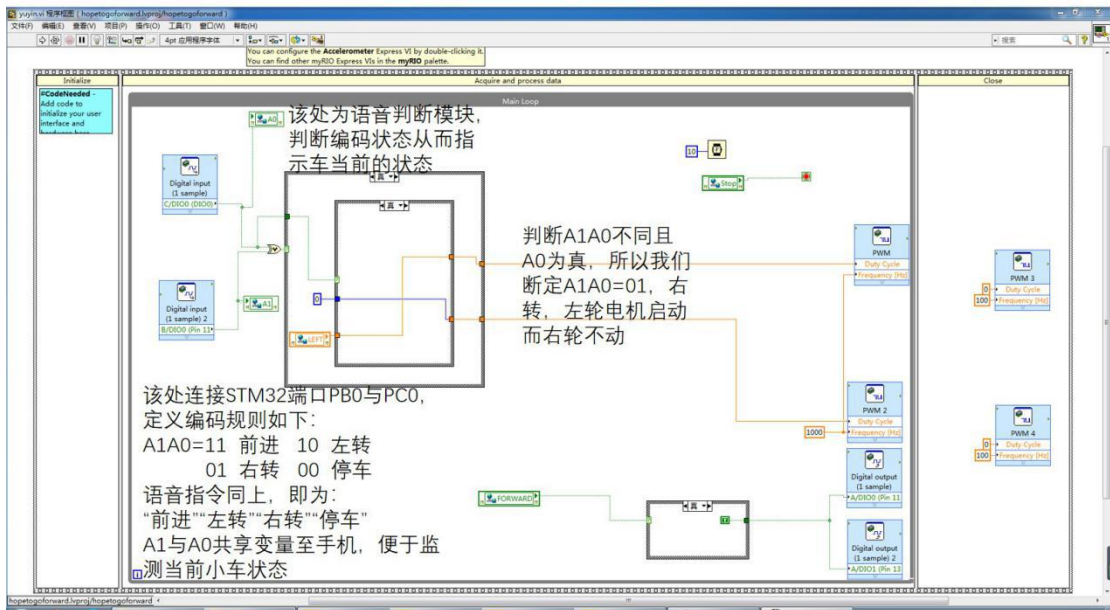
左侧传感器输出	右侧传感器输出	小车避障决策
T	T	倒车（即正方向的直行）
T	F	右转（即倒车意义上的左转）
F	T	左转（即倒车意义上的右转）
F	F	听从语音控制逻辑

3. 程序框图设计以及框图的文字分析

3.1 语音模块

我们首先对语音模块的相关配合程序进行了设计，从而便于小车进行语音控制。我们通过 STM32 板上的 PB0、PB1 端口表现出的编码来传递信息，从而 myRIO 通过判断端口上的状态来判断指令内容从而做出反应。

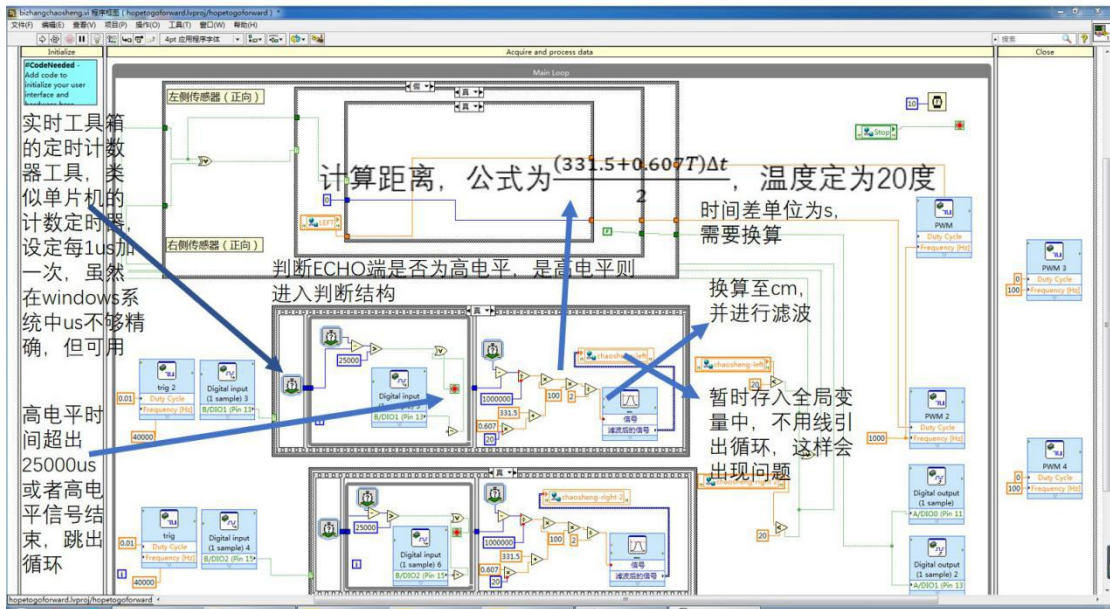


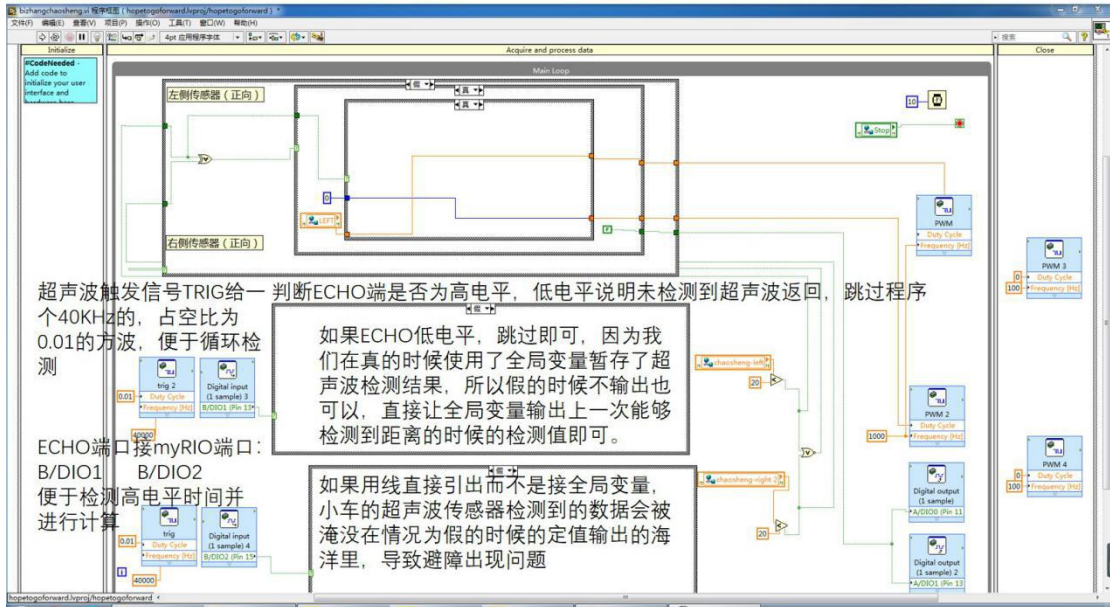


3.2 语音模块加超声波模块

3.2.1 超声波模块的距离运算分析

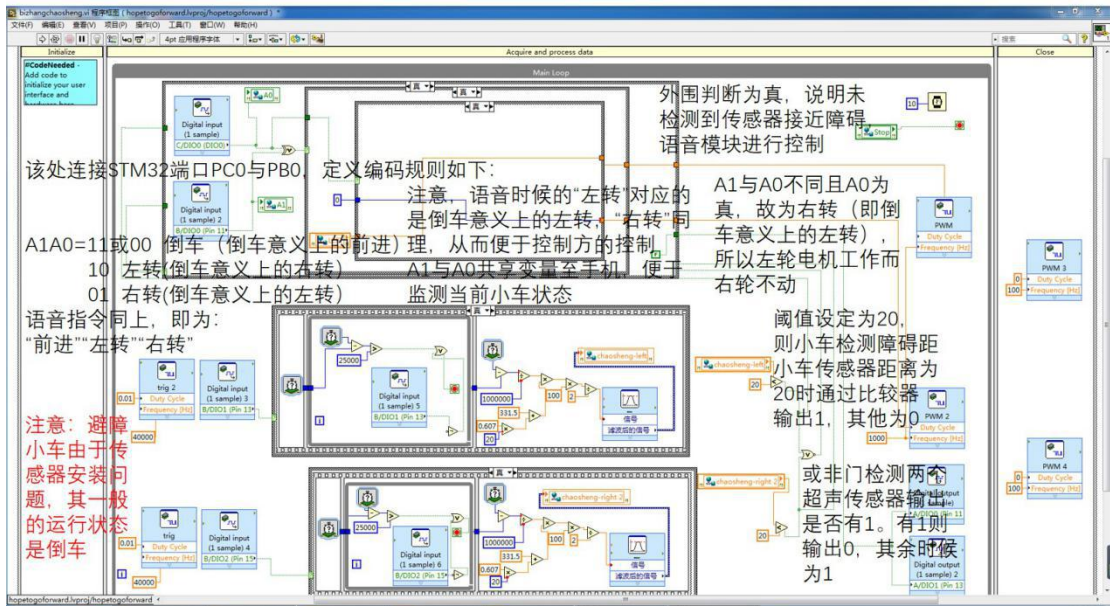
超声波模块是依据 ECHO 端口上的高电平时间来计算距离的，所以其高电平时间的计算是重点。我们利用 LabVIEW 的 RT-Time 模块进行计时，得到计时结果之后可以便可以进行距离的计算，计算公式为声速*时间/2 即可。

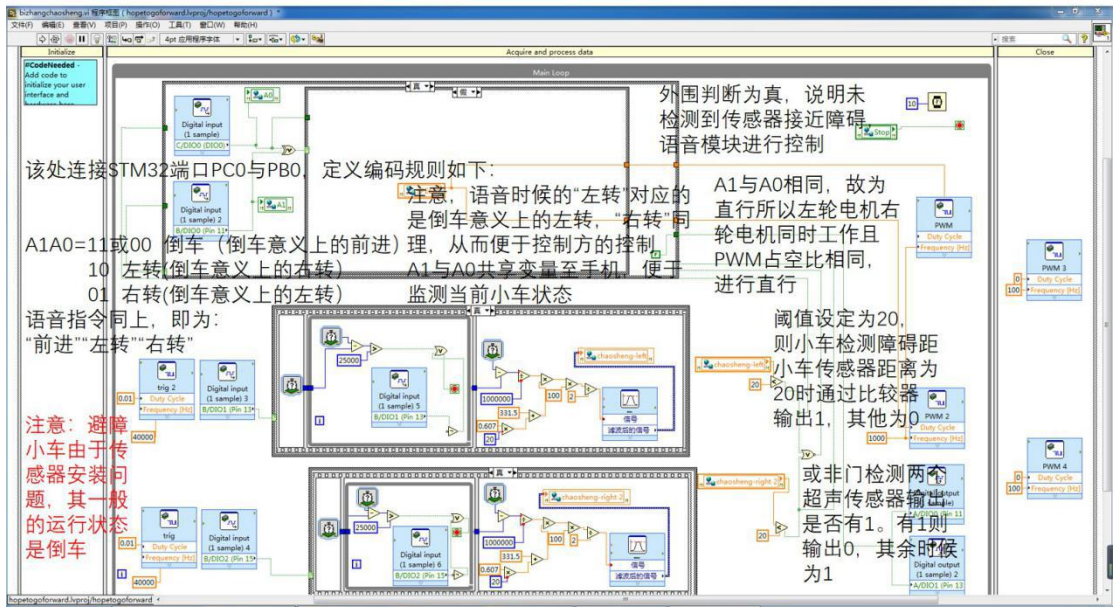
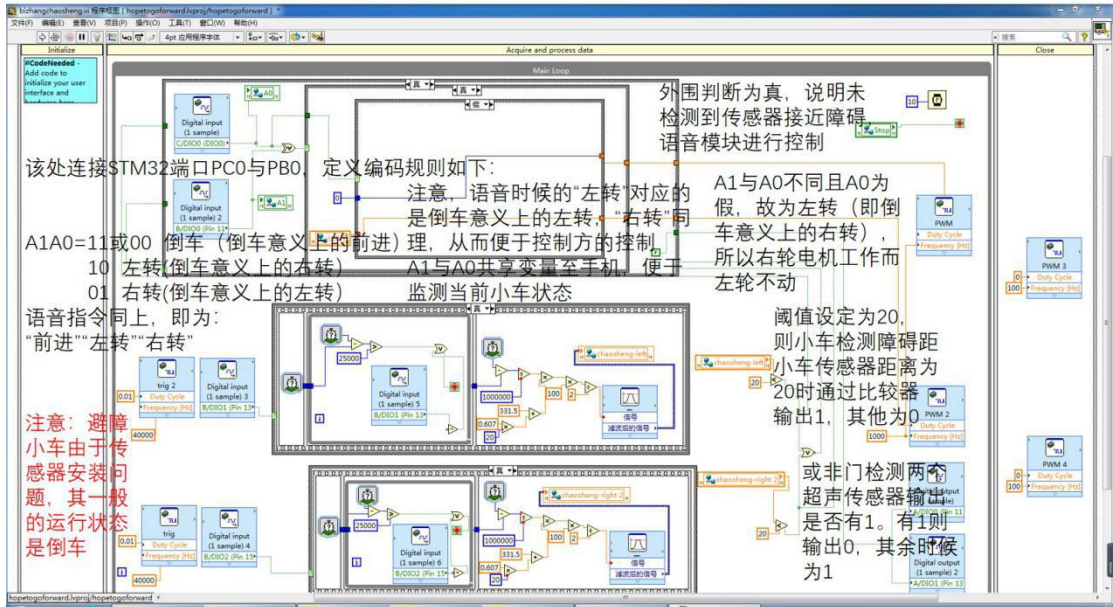


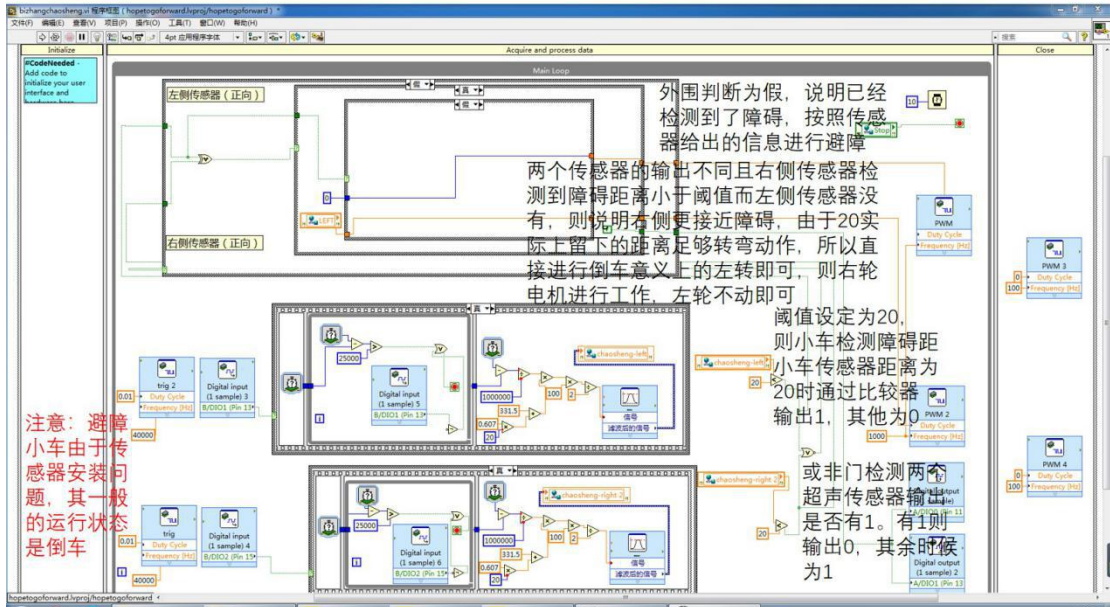
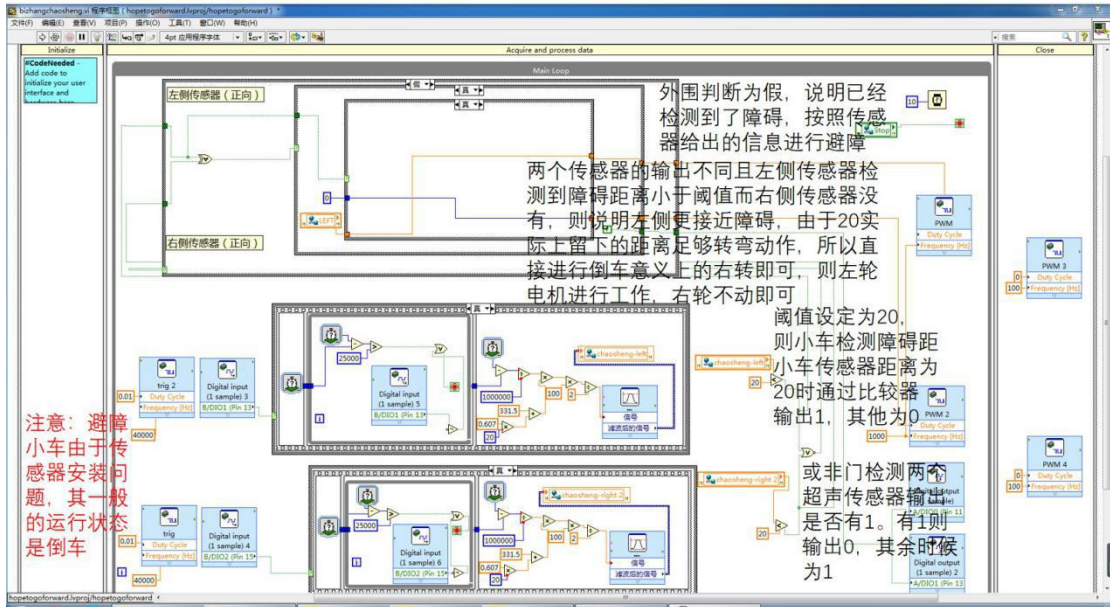


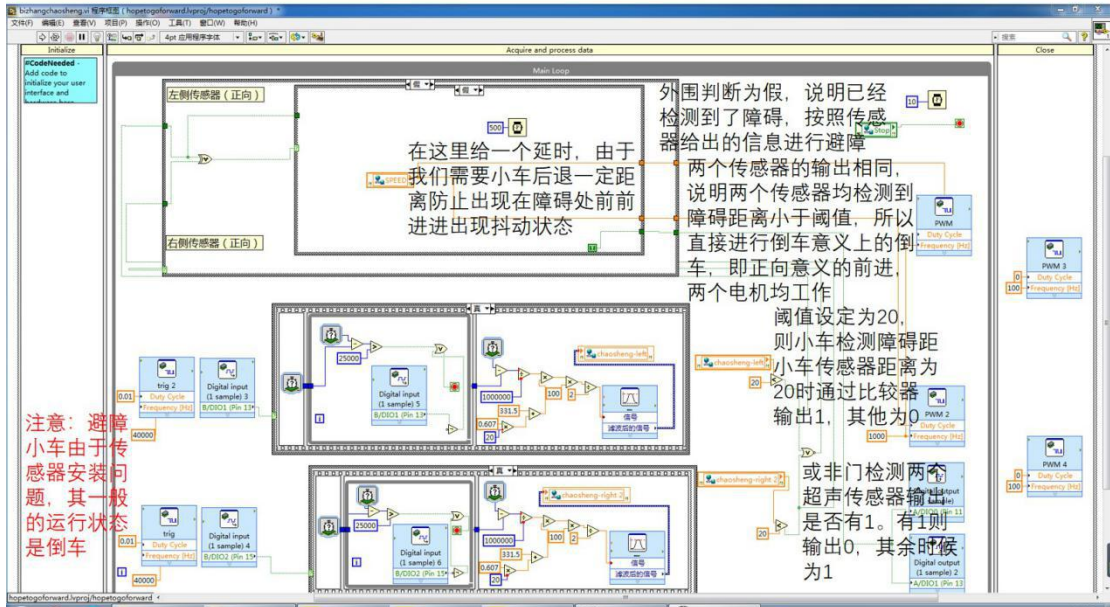
3.2.2 超声波传感器控制小车逻辑

小车在超声波传感器得到的距离数据下进行小车的控制。但实际上由于传感器的精度问题以及检测到的数据不稳定，所以小车不能很好的进行避障。注意小车是倒着走的，所以此时的“前进”为倒退，“左转”为倒车状态下，以倒车方向为正的左转，“右转”同理。



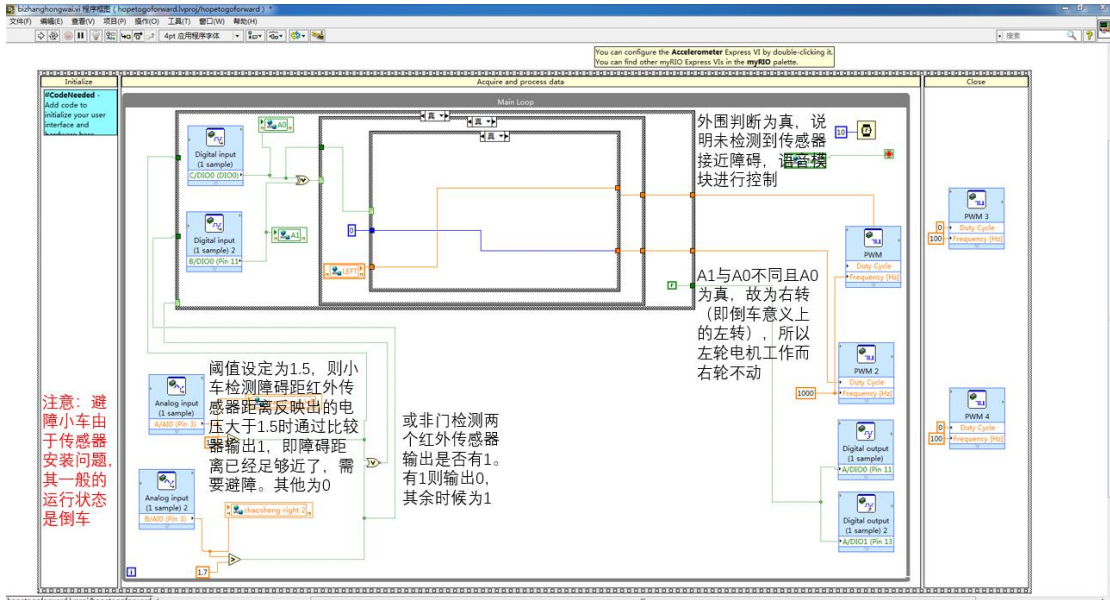


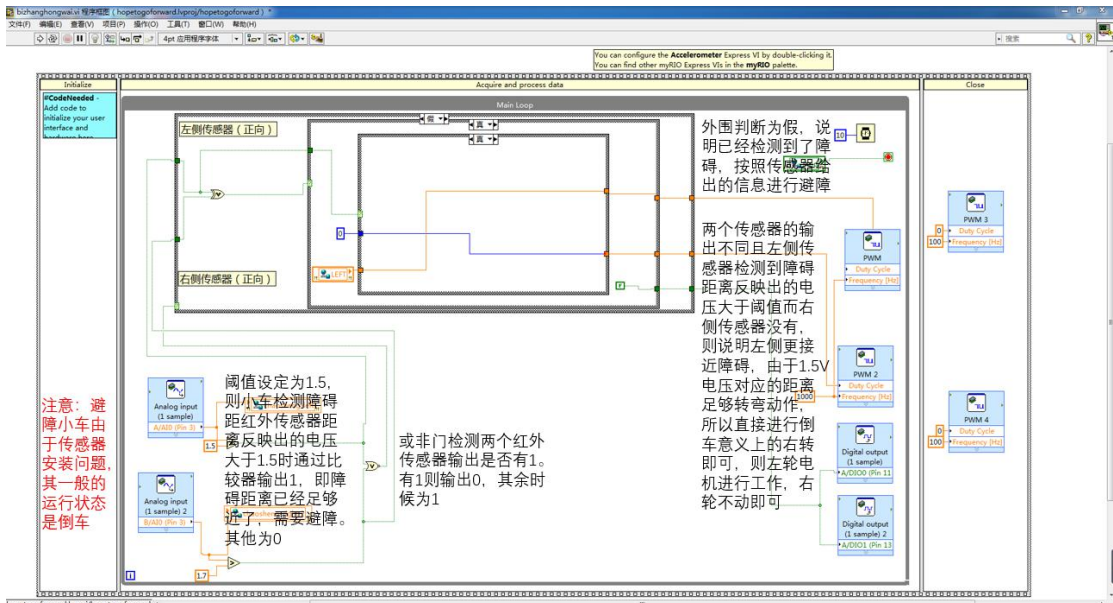
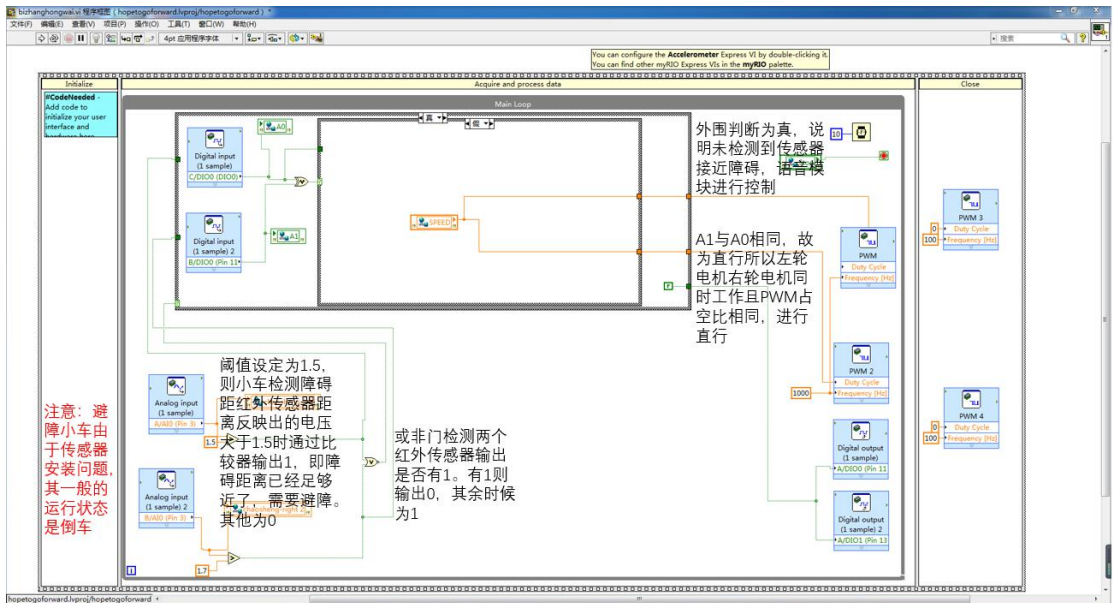
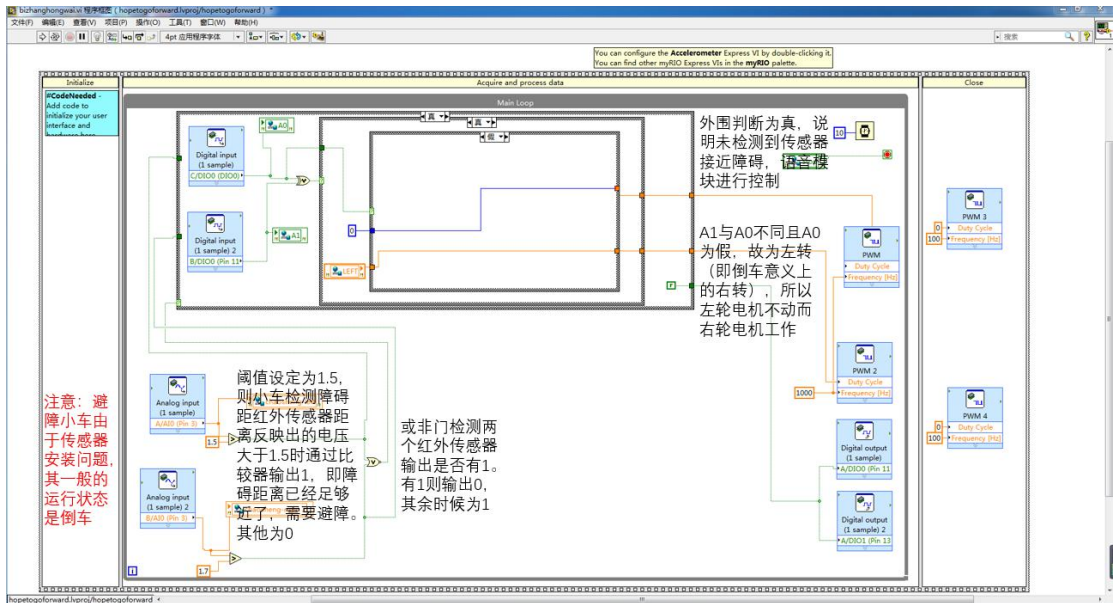


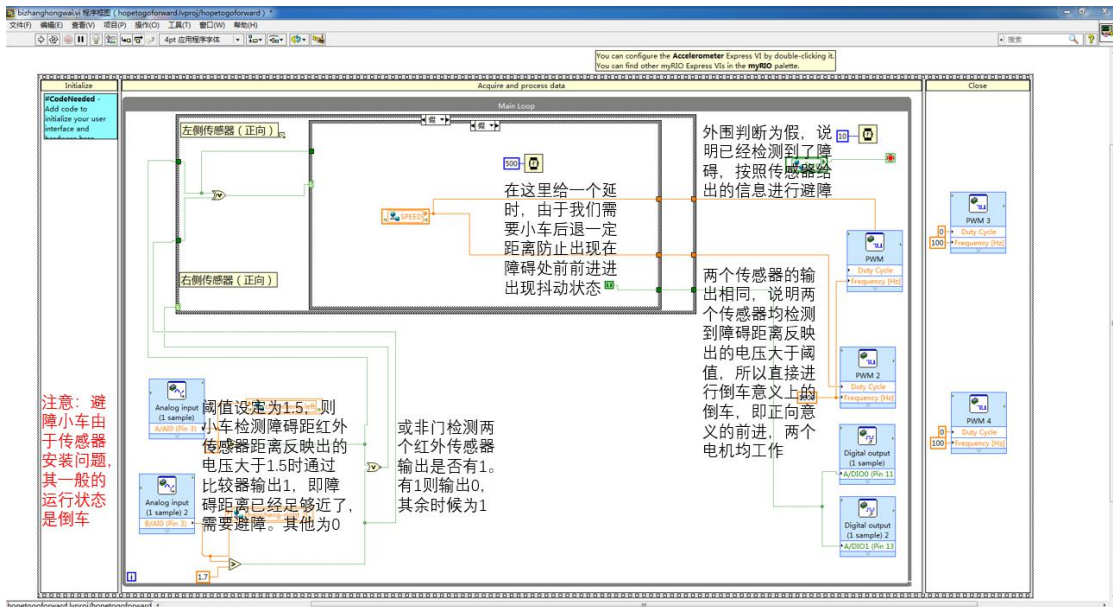
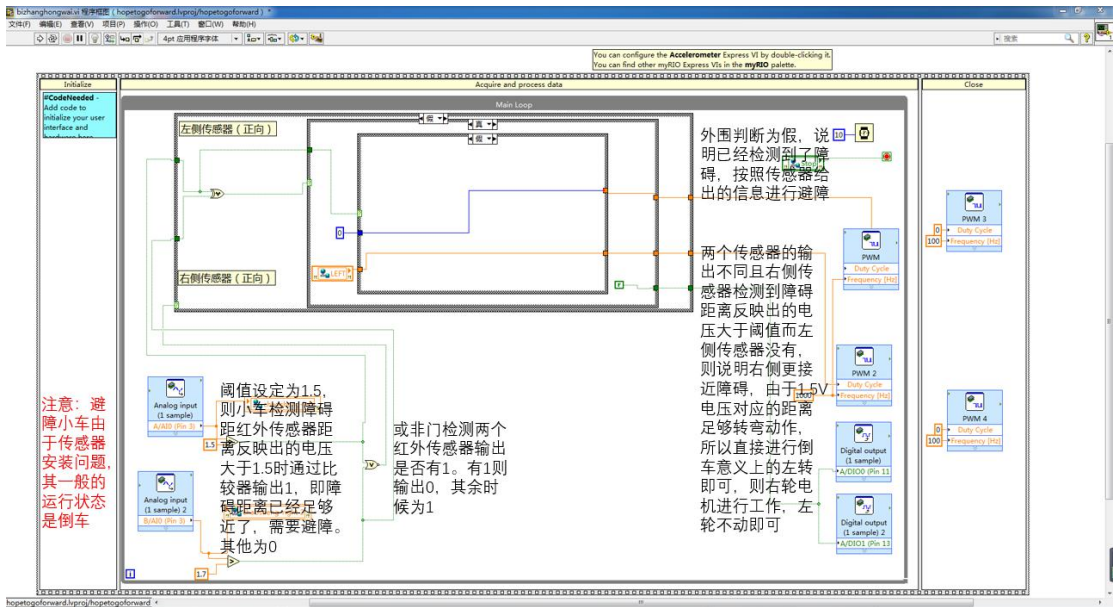


3.3 语音模块加红外模块

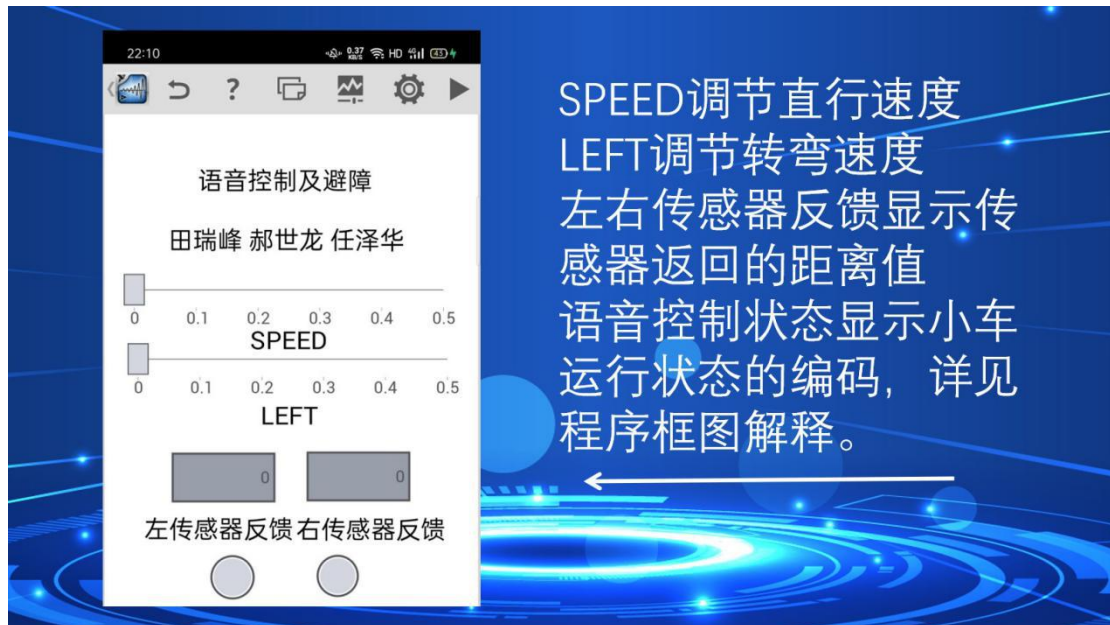
小车在红外传感器得到的距离数据下进行小车的控制。在红外传感器的控制下, 小车能够进行很好的运行并且能够有较佳的避障效果。由于其输出稳定, 所以我们也不必加入滤波模块进行滤波。注意小车是倒着走的, 所以此时的“前进”为倒退, “左转”为倒车状态下, 以倒车方向为正的左转, “右转”同理。







三、Data dashboard 设计过程



我们使用该 Data dashboard 进行小车的控制。SPEED 与 LEFT 进行直线速度与转弯速度的控制，左右传感器反馈框可以返回两个传感器的返回值，其中超声波传感器程序返回的是测距值，远红外传感器返回的是输出电压值。语音控制状态由两个显示灯进行显示，编码与状态的对应详见 LabVIEW 程序分析项。

四、控制参数的选取过程

控制参数的选择实际上和前面实验中提到的是类似的。考虑到传感器的动态响应性能等方面，我们的速度应该慢一点才便于小车能够很好避障，但转弯速度应当给大一些，这样小车在避障时才能响应迅速不会撞到障碍。我们的直行速度 SPEED 给的是 0.26，而转弯速度 LEFT 给出的是 0.3，在红外传感器与语音模块的配合下进行避障效果较佳，能够有较为明显的避障效果。

五、遇到的问题与解决方法

1. 语音模块在测试的时候会出现无法显示正确的结果

在实验的过程中，我们首先采用三个 STM32 的引脚进行状态输出，但发现实际上引脚的状态输出无法反映到 myRIO 的输入端口上。在调整之后我们只采用两个引脚输出状态，但实际上这样也会出现无法识别 00 状态的情况，所以我们在设计避障+语音的时候将停止状态去掉了。

在实验后的思考中，我认为可能是我们输出编码的时候在单片机中是一位位

使用 `GPIOResetBits` 与 `GPIOSetBits` 两个函数进行状态的修改，这样势必会产生状态的中间态，导致程序出现问题。所以我们应当设计一个格雷编码便于状态间的转换。但考虑到人在语音控制的时候，小车的状态是随意转化的，所以很难设计出一个合适的格雷编码。所以我们应当采用串口收发进行控制更为合理。

2. 将面包板上的电机驱动电路移动位置之后电机出现失速现象

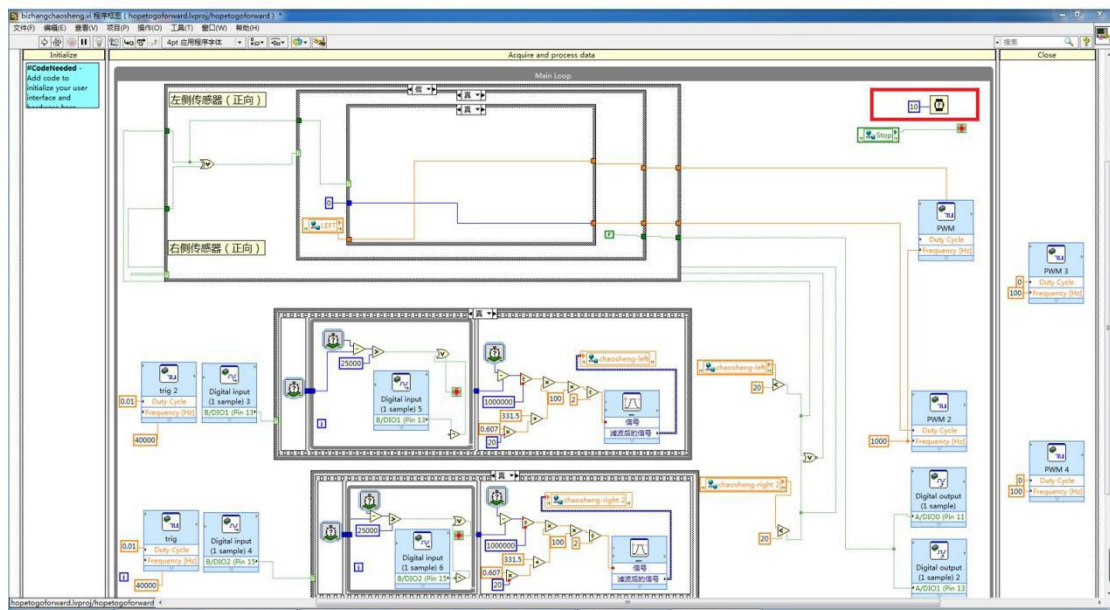
由于我们要在面包板上空出 5×2 个空列来插超声波传感器，在这样的设计要求下，我们需要将原来搭建的电机驱动电路向中间移动。但我们在移动之后发现一个轮子的电机在板子上电之后失速转动，不受 PWM 波的控制。

在对电驱动电路进行仔细排查之后发现电路没有问题，最后用万用表进行逐点排查时，发现驱动电路上 74HC02 的一个引脚电平悬空，所以发现了面包板上同一列中有一个坏点未与其他四个点相通。最后使用导线将芯片引脚套环之后进行与另外四个点的短路解决了问题。

3. 超声波测距不准确，数据跳动过大

超声波在测距的时候准确度不够高，导致小车的避障无法顺利进行，并且输出的测距数据跳动极大，导致小车在有些时候无法迅速响应避障，但却在一些没有障碍的地方胡乱避障。

当时由于排查时间不多，所以我们换用了红外传感器。但在后面的思考中，我认为这个问题的产生主要有两方面原因。第一，我们采用的计时方式是使用 RT-Time 工具包的 32 位计数定时器，会有溢出情况出现，所以在计时过程中会有出现结果为 0 的情况。并且实际上我们可以发现，由于例程中每一个主函数的循环有一个 10us 的延时（红框部分），导致超声波响应 ECHO 口的高电平的情况也不是很好，所以会有较大的跳动。但由于未能进行修改实践，所以这也仅为本人猜测。



4. 超声波测距发现距障碍距离短时小车无反应

在第一次使用超声测距控制小车进行避障时，我们发现小车离障碍较近时，小车并不会进行避障，虽然传感器已经反应过来并反映出来了。

经过检查程序发现，我原来是将超声测距得到的结果直接进行输出，但超声波传感器并不是时刻进行距离的输出，所以在 ECHO 输出为假的时候我输出的是一个常值，导致程序中常值的输出占大部分情况，导致小车不受控制。进行调整后，我是用两个全局变量进行对距离的缓存，ECHO 为假的时候不进行任何对全局变量的操作，这样便能将距离真实进行反映。

六、实验总结与建议

本次实验过程十分曲折，我们第一次调试时由于一个 IO 口没有输出信号，我们以为是语音模块的问题，拔线进行调试，但不小心造成短路烧掉了语音模块。在重新购买以后继续调试时我们非常小心，做电子线路时，一定要注意用电安全，这不仅是对人，更是对设备，一旦发生短路造成烧毁，就会导致之前的辛苦付之东流。

在调试超声模块时，我们在进行噪音滤波、改变阈值等等方式都尝试了的基础上，效果仍然不理想。超声传感器的检测效果时好时坏，这也使得小车经常会撞到墙壁，但是由于算法的控制，在检测到距离过近以后，小车还会自动退出。不得已只好使用红外距离传感器，但是在换用红外以后也存在着传感器不灵敏的问题，在仔细排查接线和各种问题以后，在最后我们终于实现了预想的功能。

在调试是有一次因为接线短路，一个车轮始终有一个恒定的速度，我们在拆除了小车后仔细排查才发现是面包板的质量问题导致一个点与其余四个点应该导通但却未导通。我们通过将导线套环在芯片引脚上并与其余四个点进行联通才解决了问题。