第一组第一次展示

一、Arduino 语言介绍

Arduino 语言是建立在 C/C++基础上的，其基础是 C 语言，Arduino 语言只不过把 AVR 单片机（微控制器）相关的一些参数设置都函数化，不用我们去了解他的底层，让不了解 AVR 单片机（微控制器）的朋友也能轻松上手。

（1）结构

void setup() 初始化变量，管脚模式，调用库函数等

void loop() 连续执行函数内的语句

（2）功能

- 数字 I/O

pinMode(pin, mode) 数字 IO 口输入输出模式定义函数，pin 表示为 0～13， mode 表示为 INPUT 或 OUTPUT。

digitalWrite(pin, value) 数字 IO 口输出电平定义函数，pin 表示为 0～13，value 表示为 HIGH 或 LOW。比如定义 HIGH 可以驱动 LED。

int digitalRead(pin) 数字 IO 口读输入电平函数，pin 表示为 0～13，value 表示为 HIGH 或 LOW。比如可以读数字传感器。

- 模拟 I/O

int analogRead(pin) 模拟 IO 口读函数，pin 表示为 0～5（Arduino Diecimila 为 0～5，Arduino nano 为 0～7）。比如可以读模拟传感器（10 位 AD，0～5V 表示为 0～1023）。

analogWrite(pin, value) - PWM 数字 IO 口 PWM 输出函数，Arduino 数字 IO 口标注了 PWM 的 IO 口可使用该函数，pin 表示 3, 5, 6, 9, 10, 11，value 表示为 0～255。比如可用于电机 PWM 调速或音乐播放。

- 扩展 I/O

shiftOut(dataPin, clockPin, bitOrder, value) SPI 外部 IO 扩展函数，通常使用带 SPI 接口的 74HC595 做 8 个 IO 扩展，dataPin 为数据口，clockPin 为时钟口，bitOrder 为数据传输方向（MSBFIRST 高位在前，LSBFIRST 低位在前），value 表示所要传送的数据（0～255），另外还需要一个 IO 口做 74HC595 的使能控制。

unsigned long pulseIn(pin, value) 脉冲长度记录函数，返回时间参数（us），pin 表示为 0～13，value 为 HIGH 或 LOW。比如 value 为 HIGH，那么当 pin 输入为高电平时，开始计时，当 pin 输入为低电平时，停止计时，然后返回该时间。

- 时间函数

unsigned long millis() 返回时间函数（单位 ms），该函数是指，当程序运行就开始计时并返回记录的参数，该参数溢出大概需要 50 天时间。

delay(ms) 延时函数（单位 ms）。

delayMicroseconds(us)　延时函数（单位 us）。

- 数学函数

min(x, y)　　求最小值
max(x, y)　　求最大值
abs(x)　　计算绝对值
constrain(x, a, b)　约束函数，下限 a，上限 b，x 必须在 ab 之间才能返回。
map(value, fromLow, fromHigh, toLow, toHigh)　约束函数，value 必须在 fromLow 与 toLow 之间和 fromHigh 与 toHigh 之间。
pow(base, exponent)　开方函数，base 的 exponent 次方。
sq(x)　　平方
sqrt(x)　　开根号

- 三角函数

sin(rad)
cos(rad)
tan(rad)

- 随机数函数

randomSeed(seed)　　随机数端口定义函数，seed 表示读模拟口 analogRead(pin)函数　。
long random(max)　　随机数函数，返回数据大于等于 0，小于 max。
long random(min, max)　随机数函数，返回数据大于等于 min，小于 max。

- 外部中断函数

attachInterrupt(interrupt, , mode)　外部中断只能用到数字 IO 口 2 和 3，interrupt 表示中断口初始 0 或 1，表示一个功能函数，mode：LOW 低电平中断，CHANGE 有变化就中断，RISING 上升沿中断，FALLING 下降沿中断。
detachInterrupt(interrupt)　中断开关，interrupt=1 开，interrupt=0 关。

- 中断使能函数

interrupts()　　使能中断
noInterrupts()　禁止中断

- 串口收发函数

Serial.begin(speed)　　串口定义波特率函数，speed 表示波特率，如 9600，19200 等。
int Serial.available()　判断缓冲器状态。
int Serial.read()　　读串口并返回收到参数。
Serial.flush()　　清空缓冲器。
Serial.print(data)　　串口输出数据。
Serial.println(data)　串口输出数据并带回车符。

- Arduino 语言库文件

①官方库文件
EEPROM　　- EEPROM 读写程序库
Ethernet　　- 以太网控制器程序库

LiquidCrystal - LCD 控制程序库
Servo - 舵机控制程序库
SoftwareSerial - 任何数字 IO 口模拟串口程序库
Stepper - 步进电机控制程序库
Wire - TWI/I2C 总线程序库
Matrix - LED 矩阵控制程序库
Sprite - LED 矩阵图象处理控制程序库
②非官方库文件
DateTime - a library for keeping track of the current date and time in software.
Debounce - for reading noisy digital inputs (e.g. from buttons)
Firmata - for communicating with applications on the computer using a standard serial protocol.
GLCD - graphics routines for LCD based on the KS0108 or equivalent chipset.
LCD - control LCDs (using 8 data lines)
LCD 4 Bit - control LCDs (using 4 data lines)
LedControl - for controlling LED matrices or seven-segment displays with a MAX7221 or MAX7219.
LedControl - an alternative to the Matrix library for driving multiple LEDs with Maxim chips.
Messenger - for processing text-based messages from the computer
Metro - help you time actions at regular intervals
MsTimer2 - uses the timer 2 interrupt to trigger an action every N milliseconds.
OneWire - control devices (from Dallas Semiconductor) that use the One Wire protocol.
PS2Keyboard - read characters from a PS2 keyboard.
Servo - provides software support for Servo motors on any pins.
Servotimer1 - provides hardware support for Servo motors on pins 9 and 10
Simple Message System - send messages between Arduino and the computer
SSerial2Mobile - send text messages or emails using a cell phone (via AT commands over software serial)
TextString - handle strings
TLC5940 - 16 channel 12 bit PWM controller.
X10 - Sending X10 signals over AC power lines

以上库文件都需要下载到编译环境（如下目录：arduino-0022\hardware\libraries）中才能使用。

二、基本行动设计部分

- run() 前进函数

digitalWrite 左右轮分别前进
analogWrite 调节左右轮，PWM 比例 0~150 调速，左右轮差异略增减

- brake() 停车函数

digitalWrite 左右电机 PWM
analogWrite PWM 调速为 0

- back() 后退函数

digitalWrite 左右电机分别后退
analogWrite PWM 比例 0~150 调速，左右轮差异略增减

- left()　　左转函数(左轮不动，右轮前进)

digitalWrite　左电机不转，右电机前进
analogWrite　PWM 比例左 0，右 150

- spin_left(int time)　　急左转函数(左轮后退，右轮前进)

digitalWrite　左电机后退，右电机前进
analogWrite　PWM 比例左 150，右 150

三、避障简易程序设计部分

（1）小车程序模块
- 基于红外线探测传感器避障
- 基于超声波传感器避障
- 基于红外线超声波传感器的综合避障

①基于红外线探测传感器避障
- 元器件：红外线传感器两个，分别安装在小车前方两侧
- 工作原理：红外线传感器发出红外线信号，若没有障碍物，则不会传回任何信息；若有障碍物，传回信息。具体表现在硬件上是高低电平的变化情况。

红外避障基本思想：
1）基于单节点单步长的深度优先搜索
2）每一次程序循环开始时，读取 SR_2 SL_2 参数，获取红外传感器的信息。如果两个参数均为高电平，表示前方没有任何的障碍物。如果左高右低，说明小车右边遇到障碍物；如果左低右高，小车左边遇到障碍物；如果同时出现障碍物，小车首先向后退，然后随机选择一个方向，继续前进

示例程序(同时低电平)：

```
brake();
delay(300);
back();
delay(400);
int a = random(11,13);
if(a%2 == 0){
left();
delay(500);
}
else{
 right();
 delay(500);
```

}

②基于超声波传感器避障

- 元器件：超声波（建立在舵机平台） 一个
- 工作原理: 与之前的红外线传感器不同的是：超声波传感器返回的信号为模拟信号，因此可以利用模拟信号较为精确的测量出相关的数据。具体做法如下：1. 发出超声波信号，遇见障碍物后反弹，收到信号。2. 计算两次信号出入的时间。 3. 用时间/58
- 58 的计算方法： T(秒）= （2* D(米)）/344 (假设声音速度为 344）——————> T = 0.0058 * D --☾ 厘米 = 微秒 / 58

基于超声波传感器避障基本思想：

1）遇见障碍物之前，无任何变化，调用 run()

2）遇见障碍物思想同红外线

两种方案的比较：

- **红外避障：数字电路的输入输出，因此程序简单，小车反应速度灵敏。但是硬件基础的特性使得避障不够灵敏**
- **超声波避障：模拟信号的产生，程序复杂。但是避障效果好，但是盲区较大，因此边际会发生碰撞**

③基于红外线超声波传感器的综合避障

- 优先级顺序：红外线 > 超声波
- 可以在突发事件出现时，最快速度停下来。

后续优化设想：

①基于蓝牙连接的控制

②图像识别初步