智能物流小车自控设计方案

|  |  |
| --- | --- |
| 作品名称：智能物流小车1.0 | |
| 小组成员（签字） | 周一魏 李嘉鹏 李博林 |
| 设计要求：  自主设计并制作一种智能制造工程背景中的物流小车，小车应具有车间作业中的物料抓取、搬运、码垛、避障等功能  （1）、场地    （2）、物料形状：50\*50\*60的长方体    （3）关于任务：智能物流小车在规定时间内完成从起点出发，到达上料区抓取物料，通过指定障碍区，到达下料区将物料放置于指定位置，最后到达终点停止运行等任务。 | |
| 小车初步设计方案（简述其总体设计思路、及电气控制方案等）  （1）、传感器的分布，请在下图中，简易标出，并且列出其数量和作用，选购的传感器型号及其参数       * 1. 避障传感器 1路光敏二极管传感器\*1     1.2、循迹传感器 XLJG100CM激光漫反射传感器\*2    Risym循迹模块 寻迹避障模块寻迹传感器 TCRT5000红外反射传感器 1 .3、定点停车传感器 TCRT5000红外反射传感器\*2  （2）、小车的行走部件，列出你选购的型号及其参数，以及控制模式。  电动机型号：Y90L-4 额定功率：1.5KW 电流满载：3.7 转速满载：1400 功率因数：0.82 效率：0.81 额定电流：6.5 额定转矩：2.2 最大转矩：2.2 外形尺寸：335\*245\*190  （3）、机械手的选择，列出机械手控制的舵机数，以及小车主板与舵机控制板的沟通交流模式。    舵机数：6  沟通交流模式：串口通信控制  （4）、小车控制板的选择和输入输出端口定义  小车控制板：Arduino uno  pinMode(MotorLeft1, OUTPUT);  pinMode(MotorLeft2, OUTPUT);  pinMode(MotorRight1, OUTPUT);  pinMode(MotorRight2, OUTPUT);  pinMode(SensorLeft, INPUT);  pinMode(SensorRight, INPUT);  pinMode(SensorMid, INPUT);  pinMode(SensorLeft0, INPUT);  pinMode(SensorRight0, INPUT);  （5）、程序编写  5.1：接口定义  const int MotorLeft1=8;//左电机后退  const int MotorLeft2=9; //左电机前进  const int MotorRight1=11;// 右电机后退  const int MotorRight2=10;// 右电机前进  const int SensorLeft=4; //左测循迹  const int SensorRight=3; //右测循迹  const int SensorMid=6;//避障  const int SensorLeft0=2;//左侧停车  const int SensorRight0=5;//右侧停车  5.2：循迹方案及程序  /\*\*  \* 功能：小车循迹  \*/    const int MotorLeft1=8;  const int MotorLeft2=9;  const int MotorRight1=11;  const int MotorRight2=10;  const int SensorLeft=4; //左传感器  const int SensorRight=3; //右传感器  int SL; //左状态  int SR; //右状态  void setup() {  Serial.begin(9600);    pinMode(MotorLeft1, OUTPUT);  pinMode(MotorLeft2, OUTPUT);  pinMode(MotorRight1, OUTPUT);  pinMode(MotorRight2, OUTPUT);    pinMode(SensorLeft, INPUT);  pinMode(SensorRight, INPUT);  }  void loop() {  SL=digitalRead(SensorLeft);  SR=digitalRead(SensorRight);  if(SR==LOW&&SL==LOW){ //前进  analogWrite(MotorLeft2,100);  analogWrite(MotorRight2,100);  // digitalWrite(MotorLeft2,HIGH);  digitalWrite(MotorLeft1,LOW);  // digitalWrite(MotorRight2,HIGH);  digitalWrite(MotorRight1,LOW);  }  else if(SL==LOW&&SR==HIGH){ //左转  analogWrite(MotorRight2,70);  digitalWrite(MotorLeft2,LOW);  digitalWrite(MotorLeft1,LOW);  // digitalWrite(MotorRight2,HIGH);  digitalWrite(MotorRight1,LOW);  }  else if(SL==HIGH&&SR==LOW){ //右转  analogWrite(MotorLeft2,70);  digitalWrite(MotorRight2,LOW);  digitalWrite(MotorRight1,LOW);  // digitalWrite(MotorLeft2,HIGH);  digitalWrite(MotorLeft1,LOW);  }  }  5.3：避障方案及程序  /\*\*  \* 功能：小车避障  \*/  const int MotorLeft1=8;  const int MotorLeft2=9;  const int MotorRight1=11;  const int MotorRight2=10;  const int SensorLeft=4; //左传感器  const int SensorRight=3; //右传感器  const int SensorMid=6;    int SL; //左状态  int SR; //右状态  int SM;  void stop(){  digitalWrite(MotorLeft2,LOW);  digitalWrite(MotorLeft1,LOW);  digitalWrite(MotorRight2,LOW);  digitalWrite(MotorRight1,LOW);  }  void straight(){  analogWrite(MotorLeft2,100);  analogWrite(MotorRight2,100);  // digitalWrite(MotorLeft2,HIGH);  digitalWrite(MotorLeft1,LOW);  // digitalWrite(MotorRight2,HIGH);  digitalWrite(MotorRight1,LOW);  }  void left(){  analogWrite(MotorRight2,100);  digitalWrite(MotorLeft2,LOW);  digitalWrite(MotorLeft1,LOW);  // digitalWrite(MotorRight2,HIGH);  digitalWrite(MotorRight1,LOW);  }  void right(){  analogWrite(MotorLeft2,100);  digitalWrite(MotorRight2,LOW);  digitalWrite(MotorRight1,LOW);  // digitalWrite(MotorRight2,HIGH);  digitalWrite(MotorLeft1,LOW);  }  void setup() {  Serial.begin(9600);    pinMode(MotorLeft1, OUTPUT);  pinMode(MotorLeft2, OUTPUT);  pinMode(MotorRight1, OUTPUT);  pinMode(MotorRight2, OUTPUT);    pinMode(SensorLeft, INPUT);  pinMode(SensorRight, INPUT);  pinMode(SensorMid, INPUT);  }  void loop() {  SL=digitalRead(SensorLeft);  SR=digitalRead(SensorRight);  SM=digitalRead(SensorMid);  if(SR==LOW&&SL==LOW&&SM==HIGH){  straight();  }  else if(SM==LOW){  left();  delay(800);  straight();  delay(800);  right();  delay(800);  straight();  delay(1500);  right();  delay(800);  straight();  delay(800);  left();  delay(800);  }    else if(SR==HIGH&&SL==LOW&&SM==HIGH){  left();  }  else if(SL==HIGH&&SR==LOW&&SM==HIGH){  right();  }    }  5.4：定点停车方案及程序  /\*\*  \* 功能：小车定点停车  \*/  const int MotorLeft1=8;  const int MotorLeft2=9;  const int MotorRight1=11;  const int MotorRight2=10;  const int SensorLeft=4; //左传感器  const int SensorRight=3; //右传感器  const int SensorMid=6;  const int SensorLeft0=2;  const int SensorRight0=5;  int SL; //左状态  int SR; //右状态  int SM;  int SL0;  int SR0;  void stop(){  digitalWrite(MotorLeft2,LOW);  digitalWrite(MotorLeft1,LOW);  digitalWrite(MotorRight2,LOW);  digitalWrite(MotorRight1,LOW);  }  void straight(){  analogWrite(MotorLeft2,100);  analogWrite(MotorRight2,100);  // digitalWrite(MotorLeft2,HIGH);  digitalWrite(MotorLeft1,LOW);  // digitalWrite(MotorRight2,HIGH);  digitalWrite(MotorRight1,LOW);  }  void left(){  analogWrite(MotorRight2,100);  digitalWrite(MotorLeft2,LOW);  digitalWrite(MotorLeft1,LOW);  // digitalWrite(MotorRight2,HIGH);  digitalWrite(MotorRight1,LOW);  }  void right(){  analogWrite(MotorLeft2,100);  digitalWrite(MotorRight2,LOW);  digitalWrite(MotorRight1,LOW);  // digitalWrite(MotorRight2,HIGH);  digitalWrite(MotorLeft1,LOW);  }  void setup() {  Serial.begin(9600);    pinMode(MotorLeft1, OUTPUT);  pinMode(MotorLeft2, OUTPUT);  pinMode(MotorRight1, OUTPUT);  pinMode(MotorRight2, OUTPUT);    pinMode(SensorLeft, INPUT);  pinMode(SensorRight, INPUT);  pinMode(SensorMid, INPUT);  pinMode(SensorLeft0, INPUT);  pinMode(SensorRight0, INPUT);  }  void loop() {  SL=digitalRead(SensorLeft);  SR=digitalRead(SensorRight);  SM=digitalRead(SensorMid);  SL0=digitalRead(SensorLeft0);  SR0=digitalRead(SensorRight0);  if(SL0==HIGH&&SR0==HIGH){  stop();  }  if(SR0==HIGH){  stop();  delay(1000);  straight();  delay(500);  }  else if(SR==LOW&&SL==LOW&&SM==HIGH){  straight();  }  else if(SM==LOW){  left();  delay(800);  straight();  delay(800);  right();  delay(800);  straight();  delay(1500);  right();  delay(800);  straight();  delay(800);  left();  delay(800);  }    else if(SR==HIGH&&SL==LOW&&SM==HIGH){  left();  }  else if(SL==HIGH&&SR==LOW&&SM==HIGH){  right();  }    } | |