

教学版两轮电脑鼠

AI MicroMouse

电脑鼠入门的最佳选择



这些问题 是否困扰着你



电脑鼠价格高昂



电脑鼠入门困难



资料不齐全



现有方案老旧

设计简约 性能强劲



32位微控制器



简约外观设计



4096线磁编码器



高速空心杯电机



金属齿轮传动



调试方便快捷

方案可靠 用料扎实



超高速空心杯电机

额定电压下最高转速52000r/min
保证了电脑鼠的高速冲刺



铜齿

金属传动齿轮 坚固耐用



日本进口光头胎

使电脑鼠与迷宫接触面积更大
实现平稳冲刺



进口高品质精密轴承

传动轴承使用了具有耐高温、耐磨、
高转速性能的高品质精密轴承

人机交互 简单快捷



8颗功能指示LED

可以自由配置单独功能
实现脱机调试
灵活方便



STLINK+UART

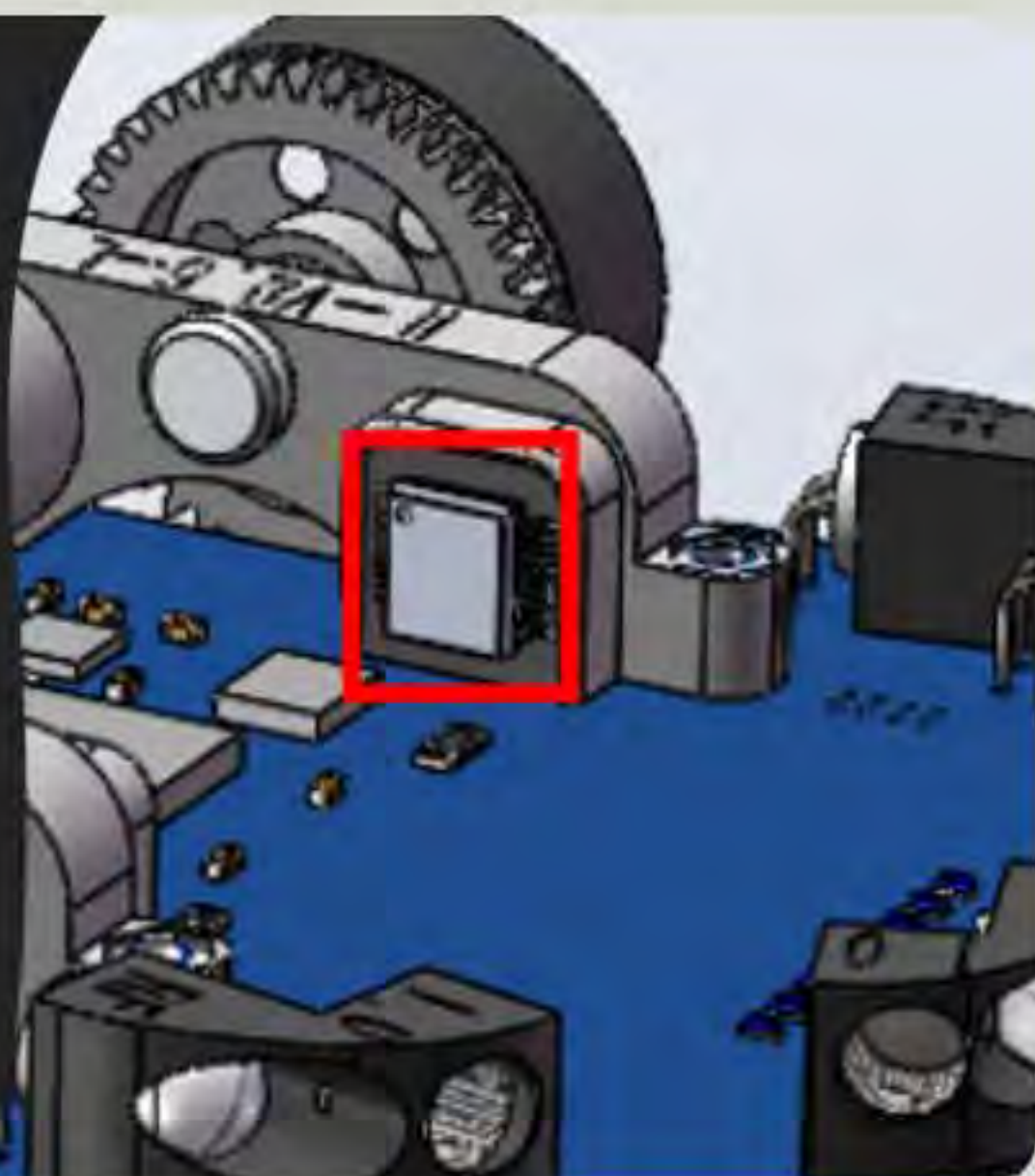
搭配专用调试器
实现一键下载调试

产品细节 处处用心

product
details

4096线高精度 磁编码器

片上集成了16位ADC、可编程数字滤波器、1%的温漂且具有SPI和IIC接口，输出AB正交增量



product
details

可调电位器设计

手动调节红外接收管匹配电阻，调试更加灵活
进一步方便课堂教学的原理演示与个人理解



product
details

防松自锁螺母

防松、抗震
耐磨损、可重复拆卸



配套资料 完整齐全

原理图

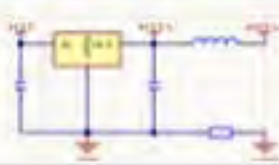
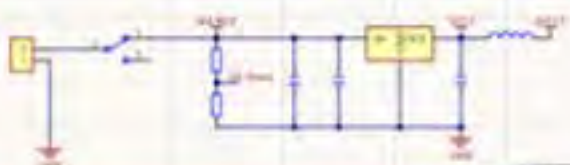
PCB

程序

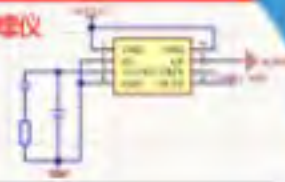


教学版两轮电脑鼠

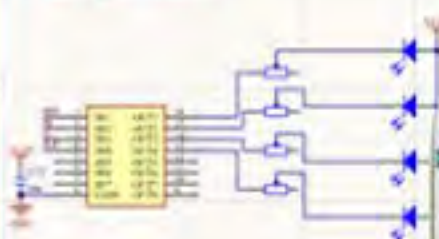
电源



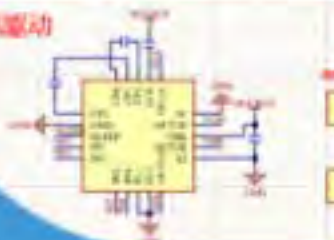
陀螺仪



红外发射与接收



电机驱动



#include "global.h"

```
3 unsigned char StartX = 0; // 起始横坐标 //
4 unsigned char StartY = 0; // 起始纵坐标 //
5
6 unsigned char GoalX1 = 0; // 目标x坐标, 有两个值 //
7 unsigned char GoalX2 = 0; // 目标x坐标, 有两个值 //
8 unsigned char GoalY1 = 0; // 目标y坐标, 有两个值 //
9 unsigned char GoalY2 = 0; // 目标y坐标, 有两个值 //
10
11 unsigned char HouseTask = 0; // 标志, 初始状态为等待 //
12
13 char GoalX = 0;
14 char GoalY = 0;
15
16 unsigned char WallMapValue[MAZE_SIZE][MAZE_SIZE] = {0}; // 保存各坐标的墙高值 //
17 unsigned char WallMapValueC[MAZE_SIZE][MAZE_SIZE] = {0}; // 保存各坐标的墙高值 //
18 unsigned char WWallMapValue[MAZE_SIZE][MAZE_SIZE] = {0}; // 保存各坐标的墙高值 //
19 unsigned char WWallMapValueC[MAZE_SIZE][MAZE_SIZE] = {0}; // 保存各坐标的墙高值 //
20
21 Coord WallMapStack[MAZE_SIZE * MAZE_SIZE] = {0}; // 在StackMap()中作堆栈使用 //
22 Coord HWallMapStack1[MAZE_SIZE * MAZE_SIZE] = {0};
23 Coord HWallMapStack2[MAZE_SIZE * MAZE_SIZE] = {0};
24 Coord CoordPaseed[MAZE_SIZE * MAZE_SIZE] = {0}; // 在Main()中暂时未走过坐标 //
25
26 Coord HouseCoord = 0, 0; // 标志HouseCoord (电脑鼠横坐标) //
27 // 标志HouseCoord (电脑鼠纵坐标) //
28
29 unsigned char AbsoluteDirection = 0; // AbsoluteDirection = 0; // 电脑鼠前进的绝对方向 //
30 // 以变量 AbsoluteDirection 记录电脑鼠前进方向上的绝对方向值, 如电脑鼠前方的绝对方向值为 AbsoluteDirection //
31 // 假如当前电脑鼠前进方向为绝对方向值的正负, 则此时 AbsoluteDirection //
32
33 unsigned char MakeWallData[MAZE_SIZE][MAZE_SIZE] = {0}; // MakeWallData[2][2] //
34 // a, 横坐标, y, 纵坐标 //
35 // 0x0-0x0f 分别代表左右上下 //
36 // 0x10-0x1f 代表左右上下 //
37
38 unsigned char PassOrNot[MAZE_SIZE][MAZE_SIZE] = {0}; // 统计是否有走过该坐标, 走过为1, 否则为0 //
39 // 统计是否有走过该坐标, 走过为1, 否则为0 //
40 // 统计是否有走过该坐标, 走过为1, 否则为0 //
41 // 统计是否有走过该坐标, 走过为1, 否则为0 //
42
43 unsigned char SportHaze[MAZE_SIZE][MAZE_SIZE] = {0}; // 统计是否有走过该坐标, 走过为1, 否则为0 //
44 unsigned char SportRelativeDir[MAZE_SIZE * MAZE_SIZE] = {0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7}; // 统计是否有走过该坐标, 走过为1, 否则为0 //
45 unsigned char SportDirCnt[7] = {0};
```

玩转电脑鼠 你只需要它



配套自制书籍

+



自研低成本平台

+



详细调试手册

不限年龄段
懂C就能开发