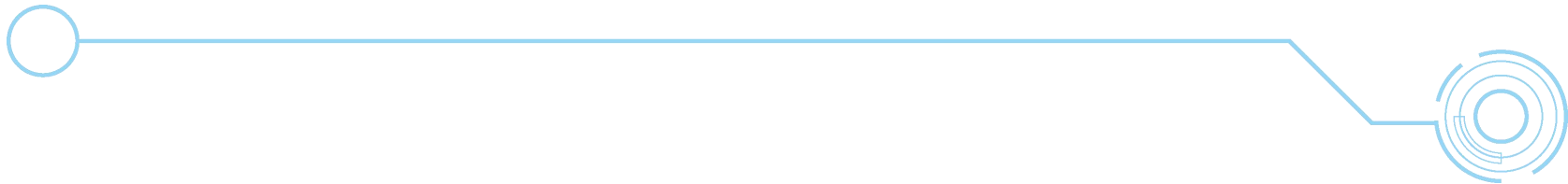




共创AI·耀星际

智能设计竞速挑战赛规则解读



2024年5月
规则如有变动另行通知

目录

1 比赛介绍

4 综合实践

2 智能车硬件介绍

5 麦克纳姆轮

3 初识Mixly软件



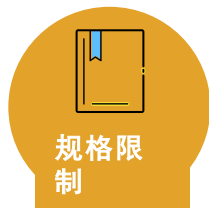


01 比赛介绍

TANIGSTAR

器材要求

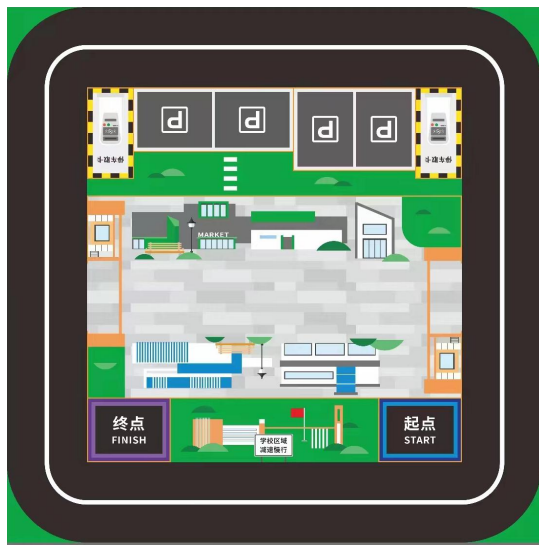
本届比赛可自主搭建智能车模型，智能车模型需搭载各类电子元器件辅助自动驾驶。智能车模型规格限制如下：



- 01 **尺寸：**智能车在出发位置的垂直投影最大尺寸不超过38cm×29cm（长×宽）。
- 02 **轮胎尺寸：**50mm<直径<70mm。
- 03 **结构：**智能车模型需为4车轮结构，探测搬运机器人模型内部需搭载各类传感器、控制板以及4个直流减速电机，实际布局可自行设计。
- 04 **主板：**可采用Arduino Uno、Arduino Nano、Arduino Pro Mini、Arduino Mega2560或其兼容款作为机器人模型中唯一可编程控制器件。
- 05 **传感器以及执行器：**传感器以及执行器运行过程中不得超出车身投影，电机需用TT马达，其他不做限制。
- 06 **电源：**每台智能车必须自带独立电池，不得连接外部电源，电池电压≤15V，不得使用升压电路。
- 07 为保证参赛设备稳定性，智能车为铝合金结构外壳及底盘。车身外部可搭建结构件扩展。。

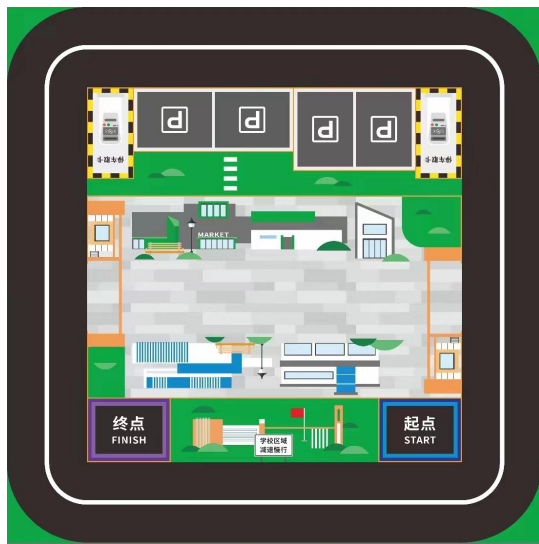
场地介绍

对抗赛整体环境为两个相同的 $2.4\text{m} \times 2.4\text{m}$ （长 \times 宽）的长方形区域，主要包含起点区、自动泊车区、停车取卡区，城市小巷区、终点区。车道线为白色，线粗 2cm ，车道宽 36cm 。整体场地如下图所示：

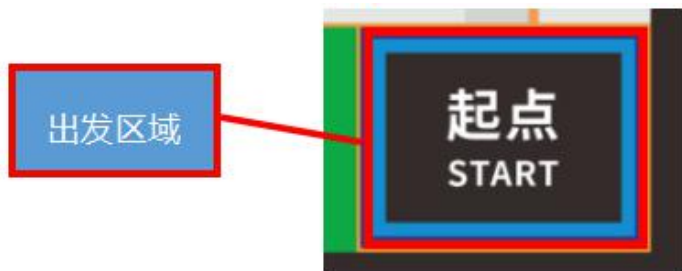


任务概述

智能车由起点区域出发，完成巡线行驶、停车取卡、互联泊车、城市小巷4个任务，最后到达终点区域且处于静止状态。



任务介绍



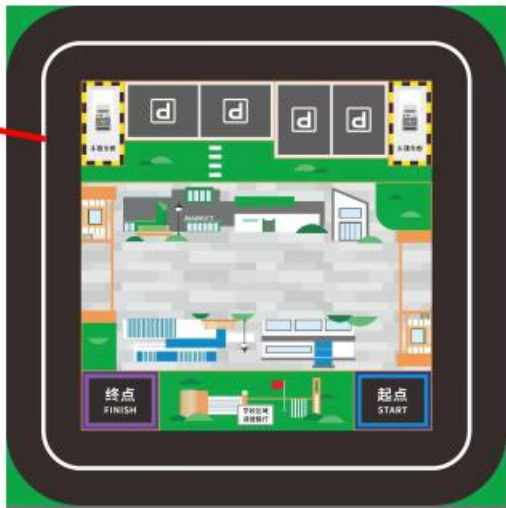
01

启动出发

智能车自主行驶且全车垂直投影完全驶离出发区域视为成功，示意图如图所示。

任务介绍

白色车道线

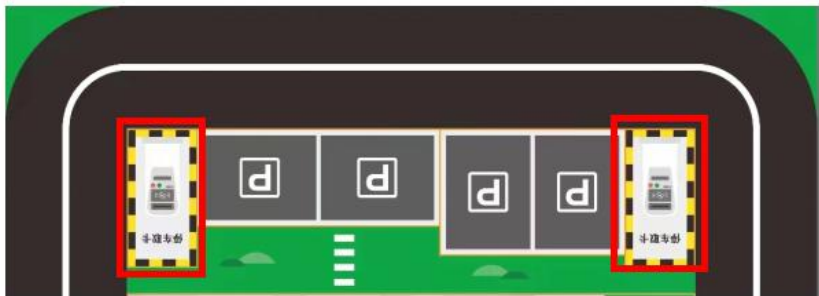


02

巡线行驶

智能车从起点出发后，沿车道线行驶，自行规划路线，全程不得脱离场地。

任务介绍



03

停车取卡

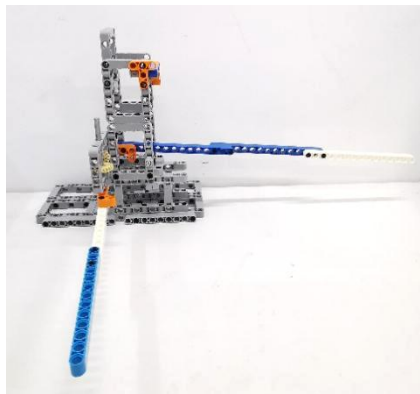
比赛现场红框区域会放置模拟道闸，智能车到达道闸前需停车，并通过触碰左侧取卡装置打开道闸，道闸打开后方可继续行驶。

小学组：1个道闸，位置固定（图中右侧）。

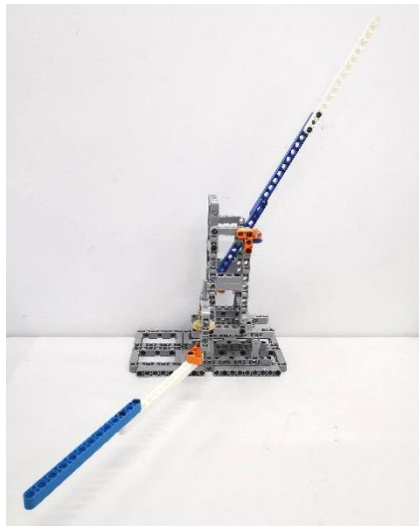
初、高中组：2个道闸。

任务介绍

道闸关闭及打开状态如图所示（如打开失败，此任务不得分）：



关闭状态



打开状态

任务介绍



04

互联泊车

如图所示，车道右侧存在4个车位（2个侧方停车位、2个倒车停车位），其中3个车位将被占用，智能车需找到空余车位，并在车位上保持停止至少3秒，停车场出入口如图所示。

空余车位位置随机，于调试后现场公布。

任务介绍



01

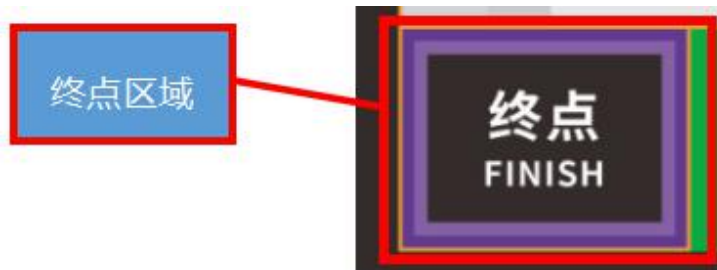
城市小巷

如下图红框位置所示，该区域是长度160cm，宽度90cm被挡板围住的城市小巷。挡板高度150mm，挡板整体横向放置，挡板位置将随机摆放，不会出现斜向放置的情况。迷宫布局现场公布。要求智能车自主穿过“城市小巷”，在小巷行驶期间需要全程亮黄色灯光。

小学组：挡板位置固定，于调试前现场公布。

初、高中组：挡板位置随机，于调试后现场公布。

任务介绍



06

到达终点

智能车到达终点区域，全车垂直投影不超出终点停放区域。

用时与次数

组别	现场编程调试时长	规定任务时长	规定任务次数
小学组	60-90分钟	120秒/次	2次
初中组	60-90分钟	120秒/次	2次
高中组	60-90分钟	120秒/次	2次

- 1.现场编程调试时长：在此时间内，每个组别所有参赛队伍统一进行编程与调试。
- 2.规定任务时长：智能车完成比赛所限定的起止时间，未在规定时间内完成比赛则强制结束本次比赛。

运行与结束

（一）智能车运行

1. 智能车出发前垂直投影完全在启动区域内，允许采用“按下按钮”或“给传感器信号”的方式进行启动，智能车启动后须自主运行。
2. 智能车可连续完成两次规定任务。
3. 在任务完成所限定的时间内无暂停，可申请重试。
4. 在任务完成所限定的时间内，参赛智能车如发生结构脱落，在不影响智能车正常运行的情况下，参赛选手可请求裁判帮助取回脱落件。
5. 比赛任务执行过程中不得更换智能车，不可以对智能车软硬件进行变更。
6. 裁判现场抽签确定选手比赛顺序。

运行与结束

(二) 比赛结束

1. 规定时间结束。
2. 规定时间内完成所有任务。
3. 智能车行进过程中发生侧翻或仰翻。
4. 智能车行进过程中，参赛选手触碰到智能车的任意部位。
5. 智能车整体投影完全脱离竞赛场地区域，当智能车四个车轮全部脱离场地视为完全脱离竞赛场地区域。
6. 智能车启动区10秒内无法启动或行进过程中静止且10秒内没有动作的可能性。

计分说明

指标	描述	分值
启动出发	智能车自主行驶且全车垂直投影完全驶离出发区域。	20分
巡线行驶	智能车沿车道线行驶，全程不得脱离赛道。	25分
停车取卡	小学组：触碰左侧取卡装置打开道闸，共1个道闸。	30分/个
	初中组、高中组：触碰左侧取卡装置打开道闸，共2个道闸。	
互联泊车	智能车驶入空余车位后停止至少3秒，且停止时智能车四个车轮均不超出“停车位”区域。	30分
城市小巷	智能车进入城市小巷任务区并成功穿越并离开。	20分
	智能车行驶在小巷期间，全程亮黄色灯光。	10分
	智能车全程未触碰挡板，得该项分值中的40分。每触碰一块挡板，挡板分值减少10分，因触碰一块挡板导致触碰到的挡板影响到其他的挡板的同样扣分。	40-10*n
到达终点	智能车到达终点区域，全车四个车轮均不超出终点停放区域。	20分

注：若智能车重复通过相同任务路段，则需重复执行该任务，但不计分。

成绩计算



1. 规定任务时长内只完成部分任务，按实际完成的任务计算得分。
2. 取两次比赛得分高的一次计为最终成绩，成绩高者排名靠前，若成绩相同，完成任务时长少者排名靠前。
3. 若分数、完成任务时长均相同，则判定为并列名次。

不予评奖

1. 取消比赛资格。
 - (1) 重复或虚假报名。
 - (2) 找他人替赛或替他人比赛。
 - (3) 参赛队伍选手迟到15分钟以上。
 - (4) 参赛队伍选手未全部到场比赛。
 - (5) 参赛队伍选手全部未到场比赛。
2. 参赛队伍选手蓄意损坏比赛场地。
3. 参赛队伍选手不听从裁判（评委）的指示。
4. 参赛队伍被投诉且成立。
5. 编程车不符合第五项“竞赛器材”要求。
6. 借给或借用其他队伍机器人比赛。



02 智能车硬件介绍

TAISSTAR

探索者

1

4 电机驱动

2

4 麦克纳姆轮结构

3

内部可搭载主控板以
及各类传感器

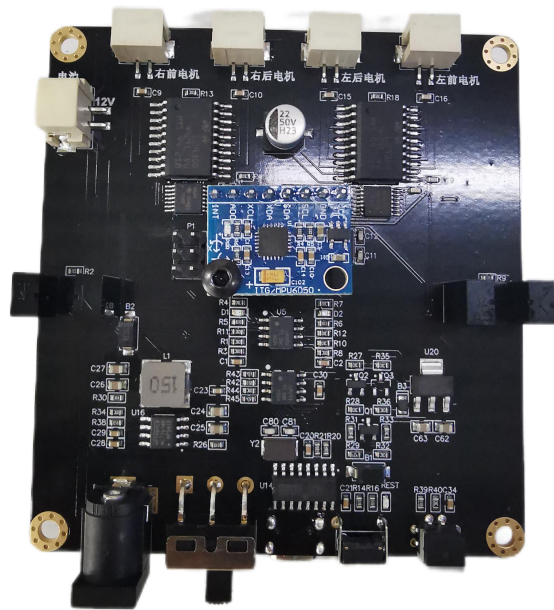




电子元器件简介

元器件名称： 主控板

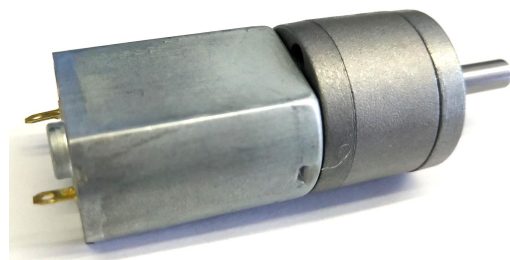
作用： 主控板是机器的控制器，上面有各种各样的端口，既可以接收外界信息，又可以发出控制指令。它的工作是：接收信号，处理信号，发出信号。



■ 元器件名称： 电机

■ 作用： 电机又叫作“马达”，它是一种将电能转换为旋转运动的动力输出设备。

探索者共有4个电机，每个电机单独控制了1个轮子，且均带有编码器，可通过主控板准确的控制电机的转速。



■ 元器件名称： 麦克纳姆轮

■ 作用： 麦克纳姆轮是一种全向轮，由四个轮子组合运动，实现全向移动的功能。



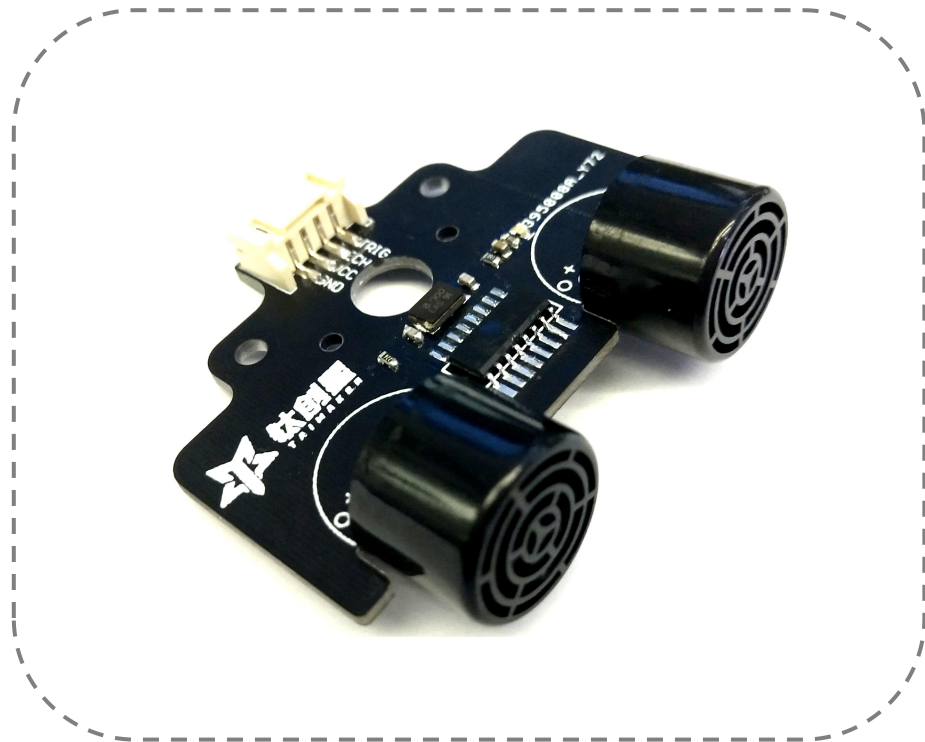
■ 元器件名称： 4路灰度传感器

■ 作用： 传感器巡线，可巡单线。



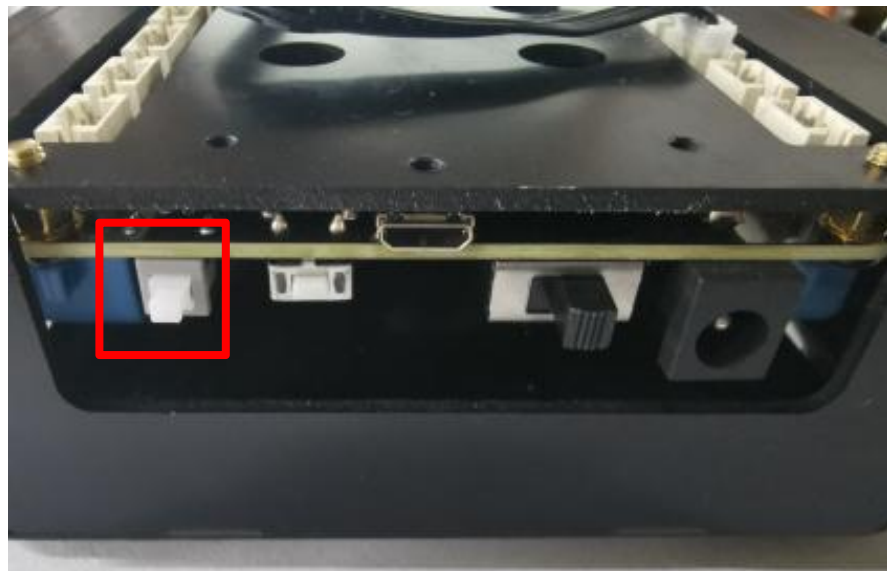
■ 元器件名称： 超声波传感器

■ 作用： 检测一定范围内的障碍物。



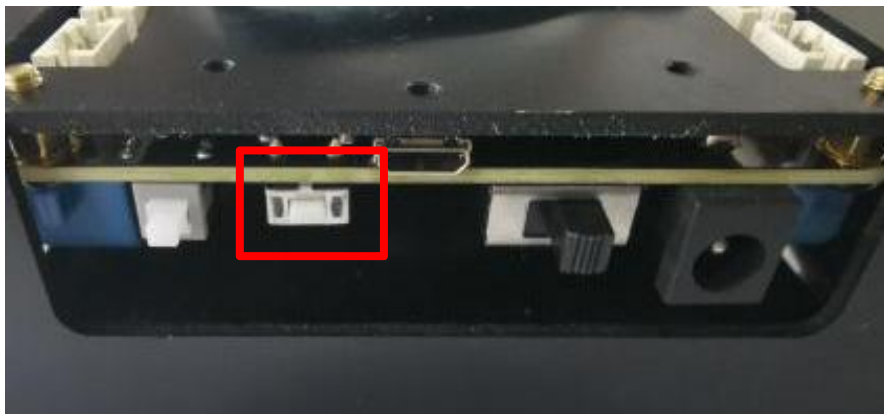
■ 元器件名称： 自锁按键

■ 作用： 通过按键可切换小车状态。



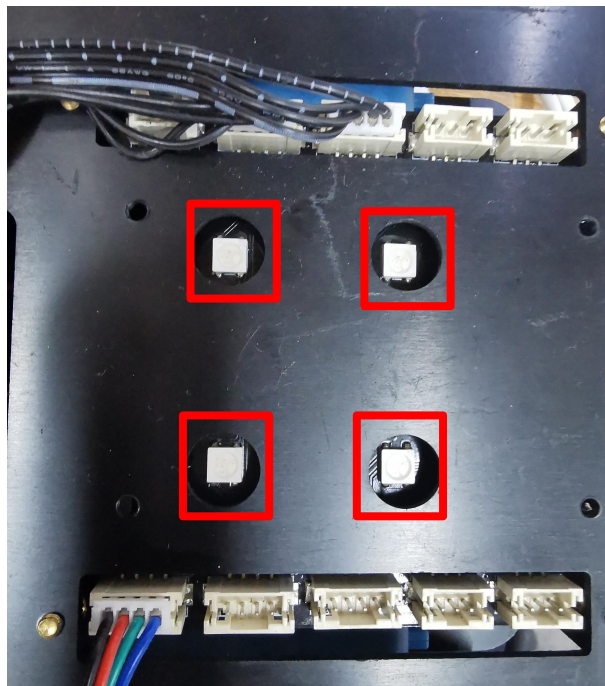
元器件名称： 复位按键

作用： 按下按键可使主板内所存放的程序重新开始运行。位于USB接口旁。



■ 元器件名称： RGB灯

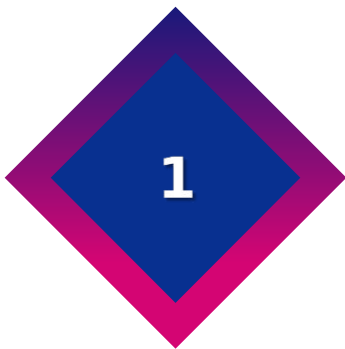
■ 作用： 探索者上有4个RGB灯珠，可
编程调节亮度、颜色、
控制亮灭情况。





03 初识Mixly软件

TRANSSTAR



安装Mixly软件

安装步骤

解压缩



Mixly

打开文件夹



mixly

点击Mixly.exe

- arduino-1.8.2
- blockly
- company
- Mixly_lib
- mixlyBuild
- mylib
- sample
- setting
- testArduino
- Mixly.exe**
- Mixly.jar



安装驱动

安装驱动

打开Mixly文件
夹

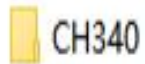


打开里面的
驱动文件夹

- arduino-1.8.5
- blockly
- company
- microbitBuild
- Mixly_lib
- mixlyBuild
- mylib
- sample
- setting
- testArduino
- 驱动**
- 示例程序
- 0.998.txt
- hs_err_pid31412.log
- Mixly.exe
- Mixly.jar

安装驱动

选择CH340驱动

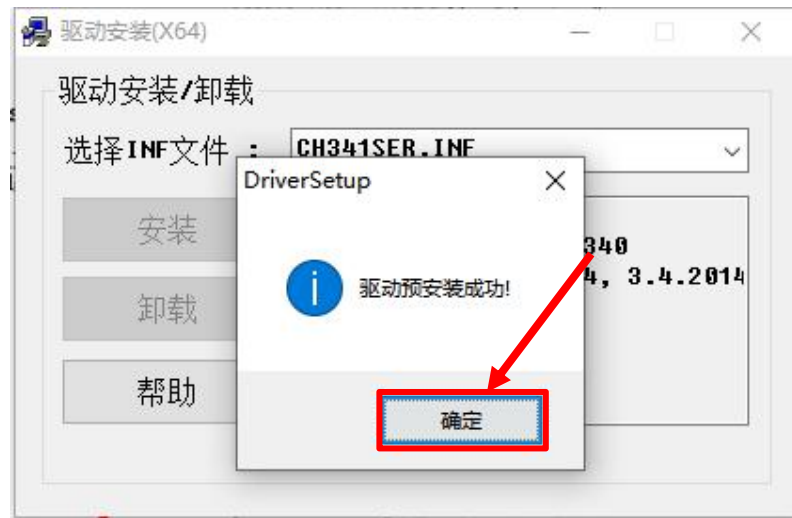


双击CH340驱动程序，安装驱动。



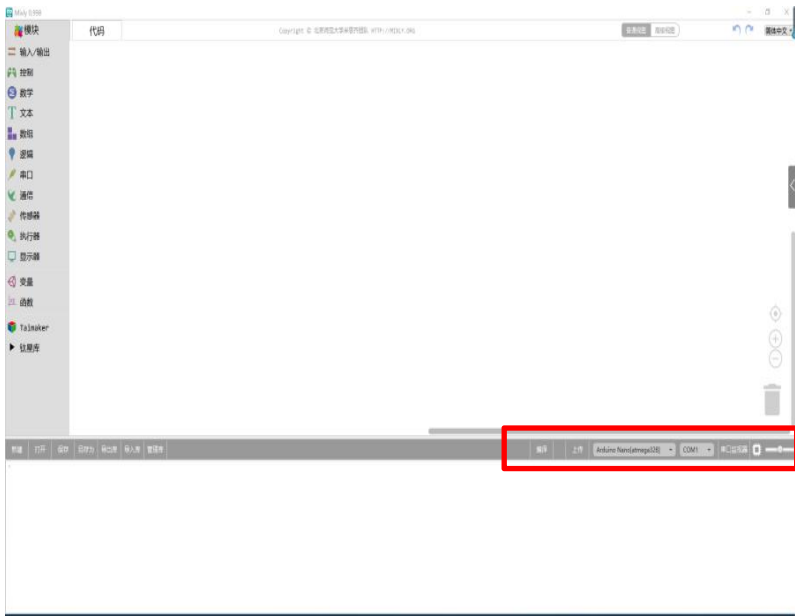
CH340驱动程序

安装驱动



注意事项

A. 打开Mixly编程软件，出现主页面：

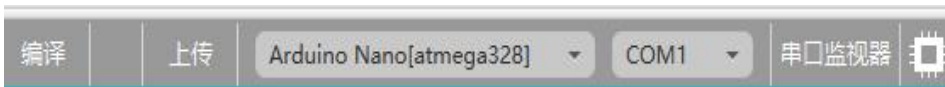


B. 关注右下角com列表处，会两种情况：

情况一：没有识别任何com口，如下图：



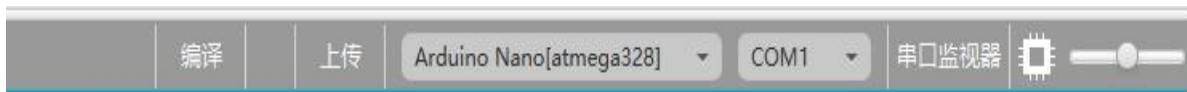
情况二：有电脑自带的com口，例如下图中，自带com口为com1（com口数字不固定）：



注意事项

C、将智能车与电脑相连，并且打开电源开关，则上述两种情况应会出现以下变化：

情况一：出现识别到主控板的com口（com口数字不固定）如下图：



情况二：在原有的com口基础上，出现一个新增的com口，则新增的com口为识别到的主控板com口，如下图，主控板的com口为com6（com口数字不固定）：



注意事项

D、若无上述变化，可能存在以下几种情况：

- ▶ **驱动安装不成功**，需要重新安装；
- ▶ **电脑供电不足**，无法识别com口，此时可以多次插拔数据线、或者更换电脑数据线连接口、或者重新启动电脑；

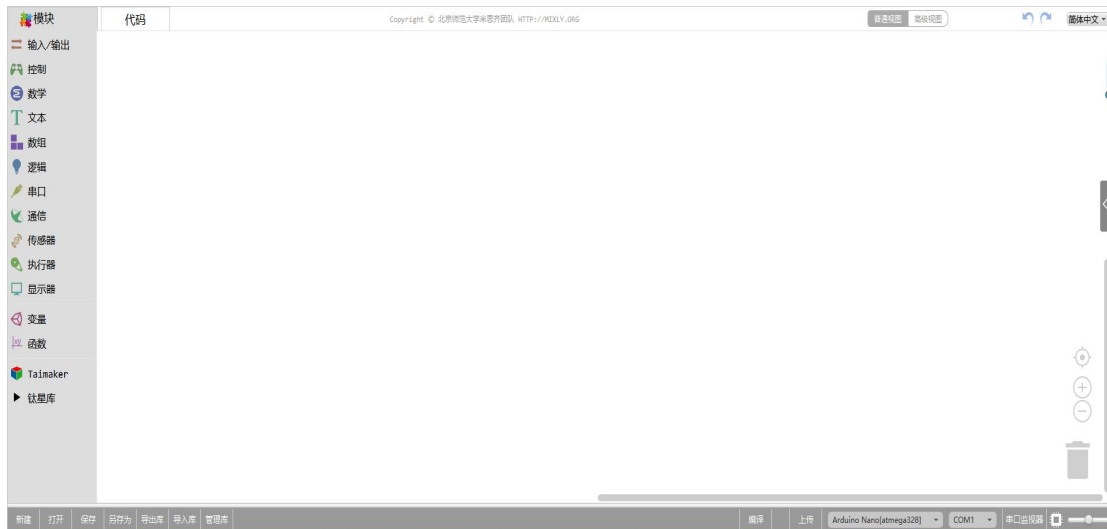


Mixly软件界面

初识Mixly软件——Mixly软件界面



界面最左侧有一列“模块”列表，点击列表下方的各个功能，每个功能里都会包含很多像拼图一样的模块，和我们传统意义上的编写代码不同，由于同学们的认知能力和思维能力还达不到直接写代码的程度，所以软件的发明者将这些复杂的代码简化封装成了一个一个的拼图模块，我们只需要将这些模块拖拽到“程序构建区”，按照编程的逻辑思维进行排列组合，最后点击“上传”，就可以达到同样的编程效果啦。相对的，如果你想查看模块化的程序的底层代码，只需要点击左上方的“代码”即可。



B

模块区：这里包含了Mixly中所有能用到的程序模块，根据功能的不同，大致分为以下几类：

输入/输出、控制、数学、文本、数组、逻辑、串口、通信、传感器、执行器、显示器、变量、函数。





程序构建区：按住鼠标左键拖动模块区的模块，可将它们放到程序构建区，拖至程序构建区的模块会组合成一段有一定逻辑关系的程序块。该区域右下角有一个垃圾桶，想要删除模块时，只需将模块拖至垃圾桶里。垃圾桶的上方有3个圆形按钮，可以对构建区进行放大、缩小和居中。



D

基本功能区：类同一般软件的菜单区。这里不仅包含了新建、打开、保存、另存为等常见的功能按钮，还包含了编程软件中需要用到的编译、上传、控制板选择、连接端口选择以及串口监视器这些按钮。



初识Mixly软件——Mixly软件界面

E

提示区：此区域在软件编译、上传的过程中会显示相应的编程信息。我们可以通过编程信息来解决编译、上传中出现的一些问题。



初识Mixly软件——Mixly软件界面

F

通过界面右上角的下拉菜单可以选择不同的语言版本；
单击界面左上角“代码”，可以进入纯代码模式。





Mixly模块区介绍

A

输入输出功能

该功能中包含一些管脚的输入输出（按信号类型可分为数字信号和模拟信号）模块、管脚电平中断模块、管脚持续一种状态的时间长度模块等等。



B

控制功能

支撑整个程序逻辑关系的功能，主要执行的内容是对程序结构进行的相应控制，实现不同程序模块的选择和跳转。



C

数学功能

该功能里是一些与数字相关的模块，该部分主要完成数学的变换功能。



D

文本功能

该功能中包含一些与字符、字符串相关的模块。



E

数组功能

该功能中包含一些数组模块，数组可以理解为一串用来存储数据的空间。



F

逻辑功能

该功能中包含一些执行逻辑运算处理的模块。



G

串口、通信功能

控制板可以通过串行数据的形式与计算机或其他设备进行数据交换，也能扩展很多外围的硬件模块，还能实现多个控制板之间信息的互联互通。



H

传感器、执行器、显示器功能

传感器模块涉及超声波传感器和温/湿度传感器，执行器模块涉及舵机、蜂鸣器等，显示器模块设计RGB灯、数码管、LCD显示屏灯。



1

变量功能

给一个变化的量起一个固定的名字并定义一个类型（整数、小数、长整数等），每个类型有一个数值范围，变量的数值可在此数值范围内变化。定义后的变量也可在此功能内找到并直接使用。



J

函数功能

该功能中包含了定义函数、执行函数的模块，简单来讲，函数就像一个箱子，我们把需要用到的东西全部放入箱子里，等到想用这些东西的时候直接拿起箱子使用即可。



▶ 钛创星针对不同系列的课程定制了专属的编程模块，在“模块”列表的最后一行，可以看到：



▶ 钛星库

▶ 点击“钛星库”左侧的三角形，滑动滚轴下拉列表，可以看到“探索者”：



K

钛星库——探索者X1列表

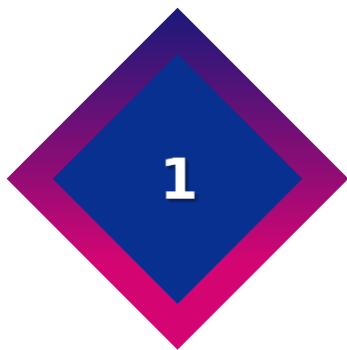
本次培训我们将使用“探索者X1”列表中的“基础底盘”功能模块，该功能列表中包含了大部分控制探索者元器件的模块，如电机控制、灰度获取模块、超声波检测模块等等。可通过拖取模块来控制元器件或利用模块获取传感器的信息内容。





04 综合实践

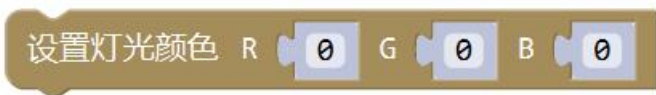
TRANSSTAR



点亮RGB灯



从探索者X1——基础底盘中找到的“设置灯光颜色”模块，该模块可设置探索者RGB灯的亮灭。



修改“R”、“G”、“B”三个数值可调制出不同颜色和亮度的灯光。

R (RED)、**G (GREEN)**、**B (BLUE)**

综合实践任务：

利用“灯光颜色”模块编写程序，使智能车亮起**黄色**的灯光。



编写完成后先点击

编译

查看程序有无编译错误，如显示“编译成功”，则选主控板型号

Arduino/Genuino Uno

，电脑端口选择com接口，随后点击

上传

，将程序上传

至Taidiuno控制板中，观察顶灯效果。

程序参考：

设置灯光颜色 R 255 G 255 B 0



基本行驶方式

电机控制模块

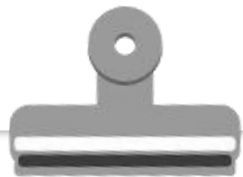
设置 左前轮 赋值为 0

A、此模块为设置电机转速模块，模块中有左前轮、左后轮、右前轮、右后轮四个选项：

设置 左前轮 赋值为 0

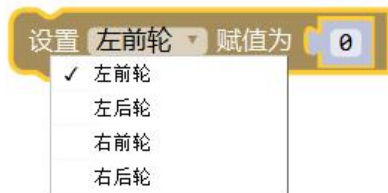
- ✓ 左前轮
- 左后轮
- 右前轮
- 右后轮

B、电机转速的范围值在 $-255 \sim 255$ 之间。



任务 1

利用电机控制模块使智能车实现前进、后退。



前
进



后
退

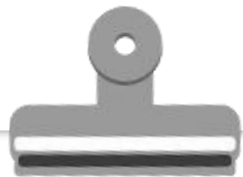


前进或后退仅需将4个电机以相同方向、相同速度转动即可。

当4电机同时正转时，则智能车直行前进；当4电机同时反转时，则后退。

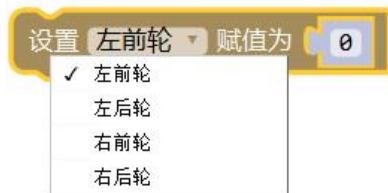
正转参数为正值，反转参数为负值。

在0~255范围内，电机正转数值越大速度越快；在-255~0范围内，电机反转数值越小速度越快。



任务 2

利用电机控制模块使智能车实现左转、右转。



左

转



Scratch code blocks for a left turn:

- 设置 左前轮 赋值为 150
- 设置 左后轮 赋值为 150
- 设置 右前轮 赋值为 200
- 设置 右后轮 赋值为 200

右

转

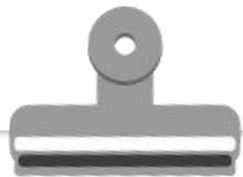


Scratch code blocks for a right turn:

- 设置 左前轮 赋值为 200
- 设置 左后轮 赋值为 200
- 设置 右前轮 赋值为 150
- 设置 右后轮 赋值为 150

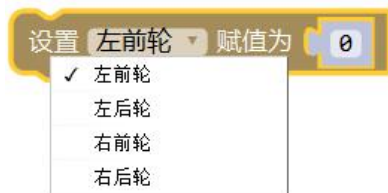
当右侧两个电机的转速大于左侧两个电机的转速时，左右电机就会产生速度差，进而形成偏差角，实现向左转向。转向角度根据左右电机速度差值决定。差值越大则转向角越大，差值越小则转向角越小。

当左侧两个电机的转速大于右侧两个电机的转速时，左右电机就会产生速度差，进而形成偏差角，实现向右转向。转向角度根据左右电机速度差值决定。差值越大则转向角越大，差值越小则转向角越小。



任务 3

利用电机控制模块使智能车实现向右原地转
(以车身中间为中心进行旋转)。



向右原地

转

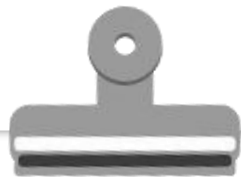
设置 左前轮 ▾ 赋值为 200

设置 左后轮 ▾ 赋值为 200

设置 右前轮 ▾ 赋值为 -200

设置 右后轮 ▾ 赋值为 -200

当左侧电机正转，右侧电机反转，且4个电机转速相同时，实现向右原地转，即转向角为0。



任务 4

智能车前进3秒后停车。

程序参考



延时模块



A、该模块可以打断当前程序，并在保持当前程序状态的情况下使程序等待一段时间。

B、模块中有两个参数可以修改，一个参数是延时的时间单位，单击下拉菜单可选择毫秒或微秒：



另一个参数是修改延时的时间，我们基本的延时时间单位为毫秒， 1000 毫秒=1 秒， 1000 微秒=1 毫秒。

停止程序模块



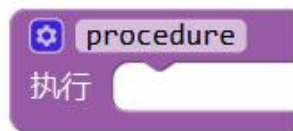
停止程序

A、该模块并非是让程序停止，而是使用该模块时程序无法执行结束，因此整个程序会卡在此模块处，就像延时无限长的时间一样，间接达到了停止程序的作用。




函数模块

函数创建模块



A、该模块在【函数】功能中。

B、创建好该“函数”模块后，在函数功能列表里就能找到对应的“执行模块”。

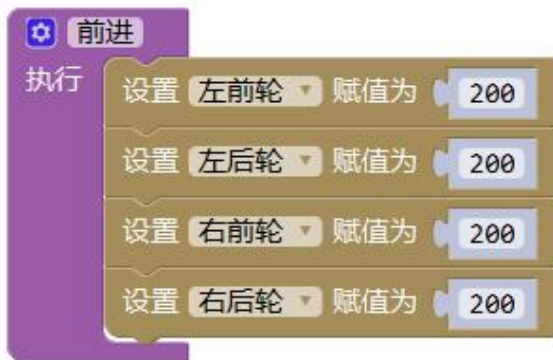
C、当主程序中再次需要放入这一大段程序时，我们只需把“执行模块”拖入，就可达到同样的效果，使用该模块主要用于方便编写以及增加程序的可读性。

任务要求



利用函数模块将刚才的程序简化。

综合实践3——函数模块





麦克纳姆轮



探索者装有两种方向的麦克纳姆轮，每个大轮子由许多小轮子组成。



0
2

轮子正转时，里面的小轮子也随之旋转，旋转方向如下图红色箭头所示方向。



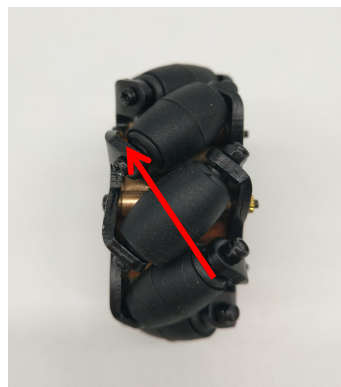
0
3

轮子反转时，里面的小轮子也随之旋转，旋转方向如下图红色箭头所示方向。



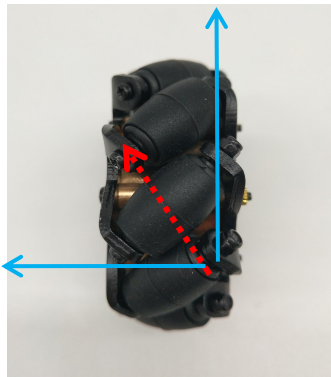
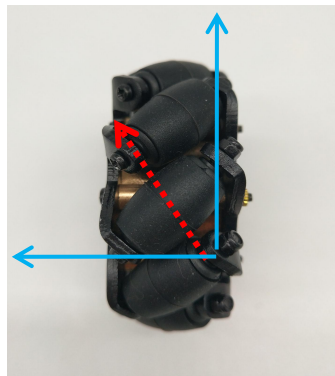
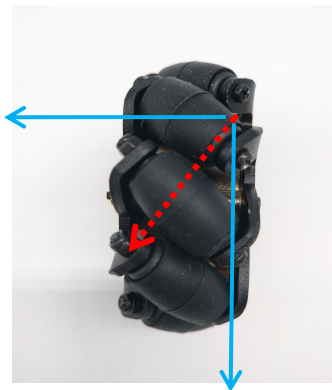
0
4

假设现在需要智能车向左平移，即左前轮与右后轮反转、右前轮与左后轮正转，那么小轮子的旋转方向如下图红色箭头所示方向。



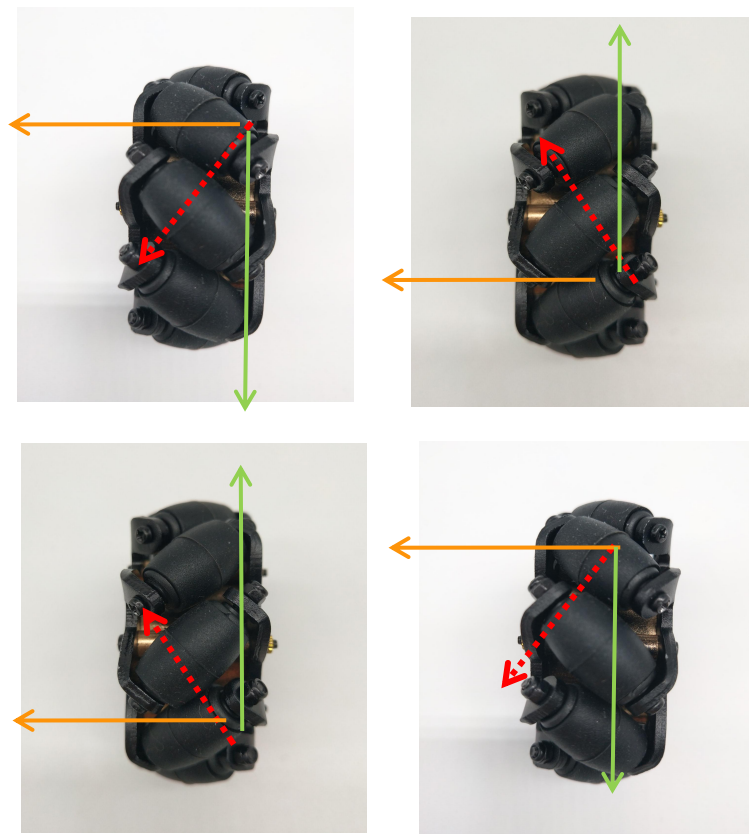
05

根据力的分解原理，可将一个斜向力分解为一个水平、一个垂直的力，下图中红色箭头为原来的斜向力，蓝色箭头为分解的力。



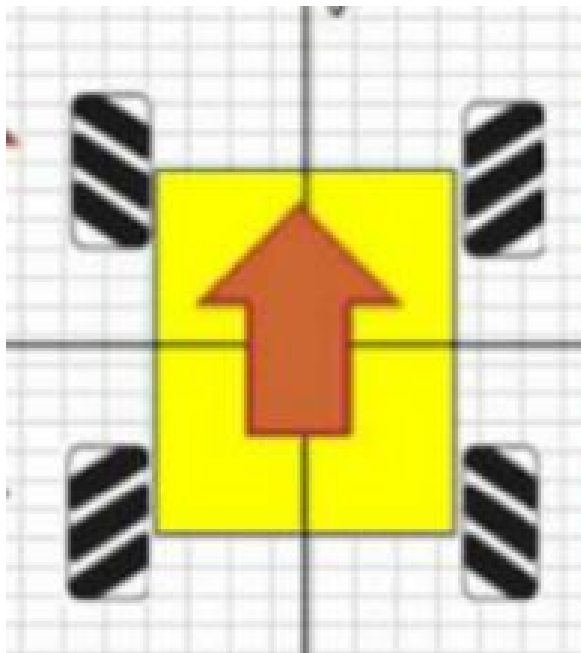
0
6

根据力的合成原理——相反方向的力可以相互抵消，相同方向的力可以叠加；则轮子最后只剩下向左移动的力，下图中橙色箭头为叠加的力，绿色箭头为抵消掉的力。

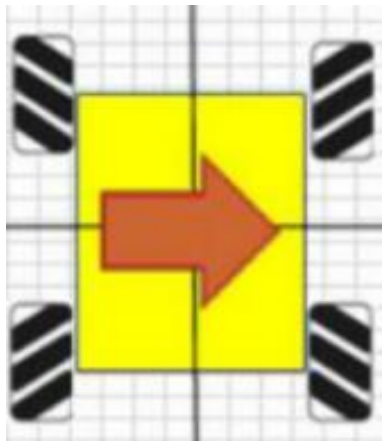




红色的大箭头是智能车的运动方向，要实现左/右平移，四个轮子需要怎么配合？

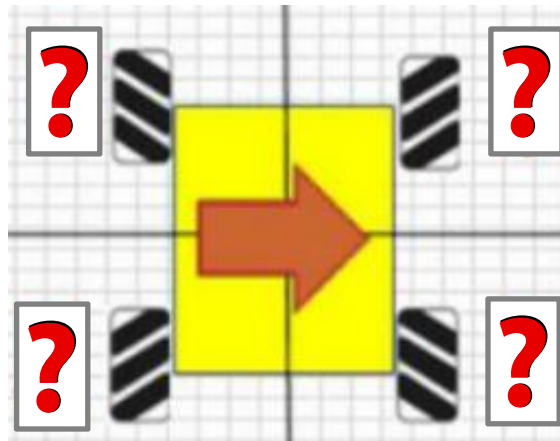


运动方向



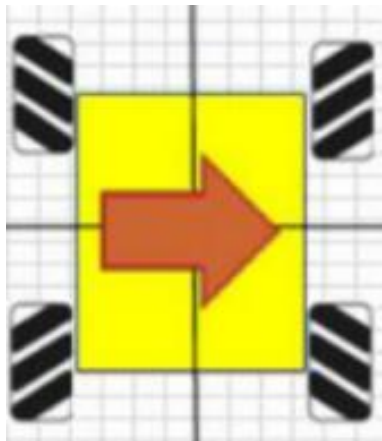
右平移

4个轮子的转向



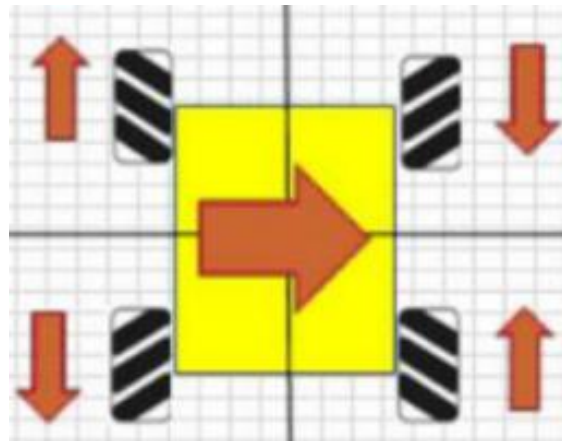
用箭头画出每个轮子的运动方向

运动方向



右平移

4个轮子的转向



左前轮与右后轮正转、右前轮与左后轮反转

右平移

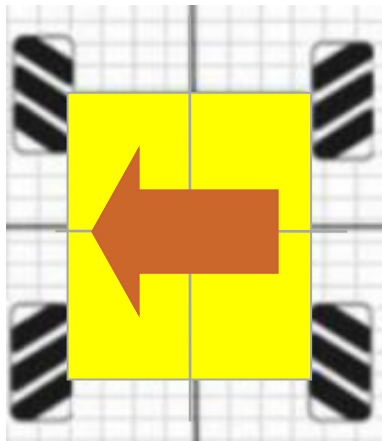
设置 左前轮 ▾ 赋值为 200

设置 左后轮 ▾ 赋值为 -200

设置 右前轮 ▾ 赋值为 -200

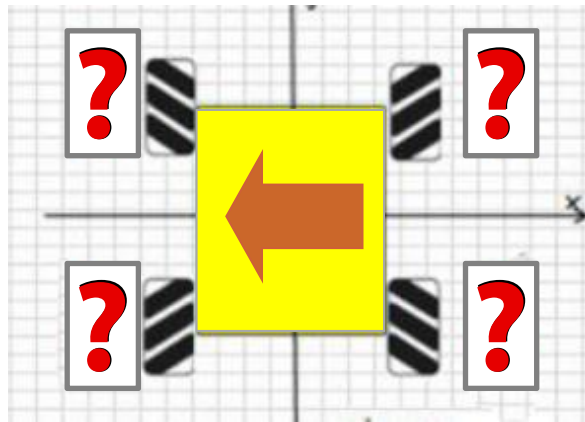
设置 右后轮 ▾ 赋值为 200

运动方向



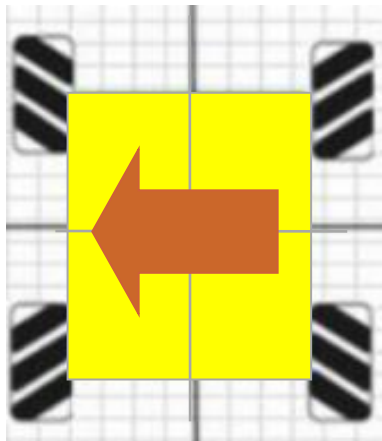
左平移

4个轮子的转向



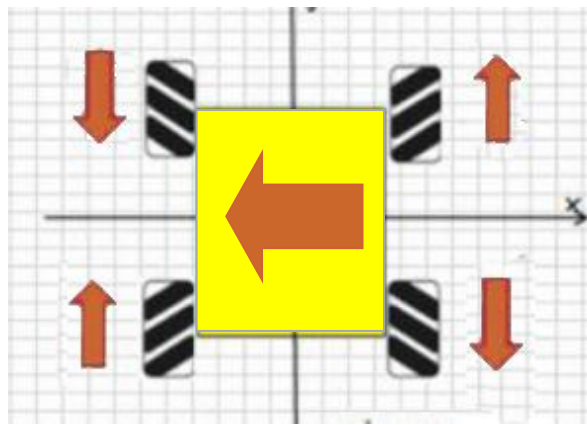
用箭头画出每个轮子的运动方向

运动方向



左平移

4个轮子的转向



左前轮与右后轮反转、右前轮与左后轮正转

左平移

设置 左前轮 赋值为 -200

设置 左后轮 赋值为 200

设置 右前轮 赋值为 200

设置 右后轮 赋值为 -200



定向移动/旋转模块

定向移动/旋转控制模块



注意：使“向……移动/旋转”模块和时，为准确进行前进、后退、左平移、右平移、左转、右转，需要学生先将智能车放置在赛道上，再打开智能车开关。

任务要求



智能车前进20cm后，向右转90度后，向右平移20cm，最后停车。

程序参考



共创AI·耀星际

THANKS!
