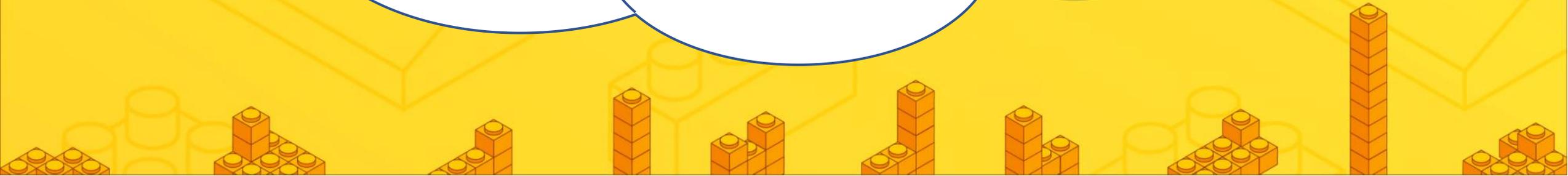




# 轨道小车 (上)



## 课程目标

- 学习使用一个灰度传感器成基础巡线动作
- 掌握程序逻辑中选择结构的应用
- 完成跳出循环的学习

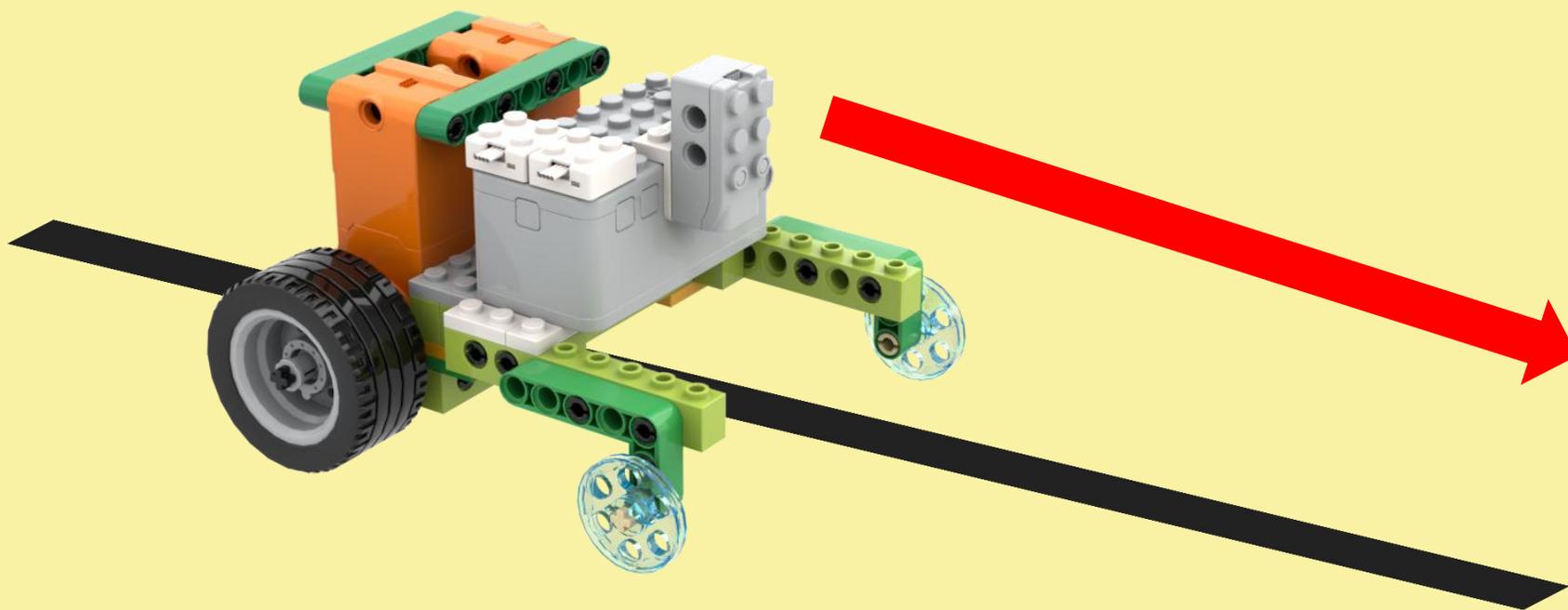


# 01 任务内容



# 任务

- 任务1：利用灰度传感器沿着黑线走



# 编程技巧1

## 变量使用方式与查看传感器数值

变量

建立一个变量

我的变量

将 我的变量 设为 0

将 我的变量 增加 1

显示变量 我的变量

隐藏变量 我的变量

**注意：变量名称只是用英文字母**

新建变量

新变量名:

light

适用于所有角色  仅适用于当前角色

取消 确定

light 0

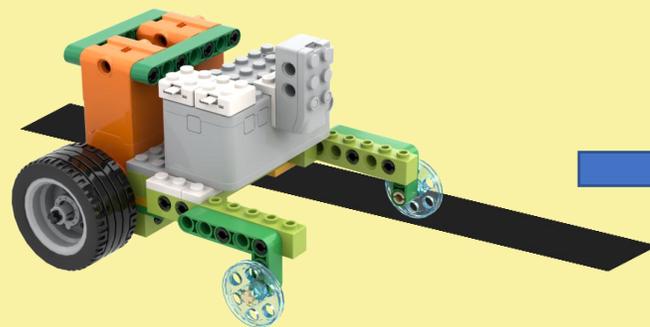
# 编程技巧1

## 变量使用方式与查看传感器数值

当 被点击

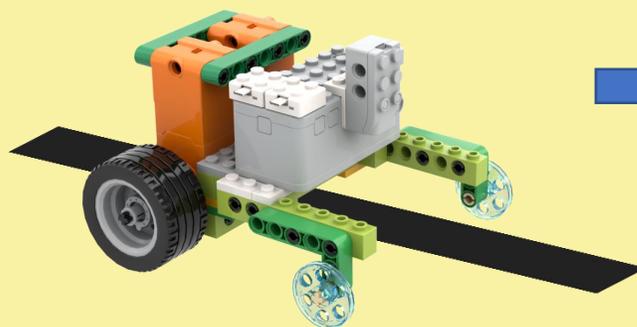
重复执行

将 **light** 设为 1# 数字灰度传感器的值



light 18

注意观察在黑线上与不在黑线上的区别哟



light 29

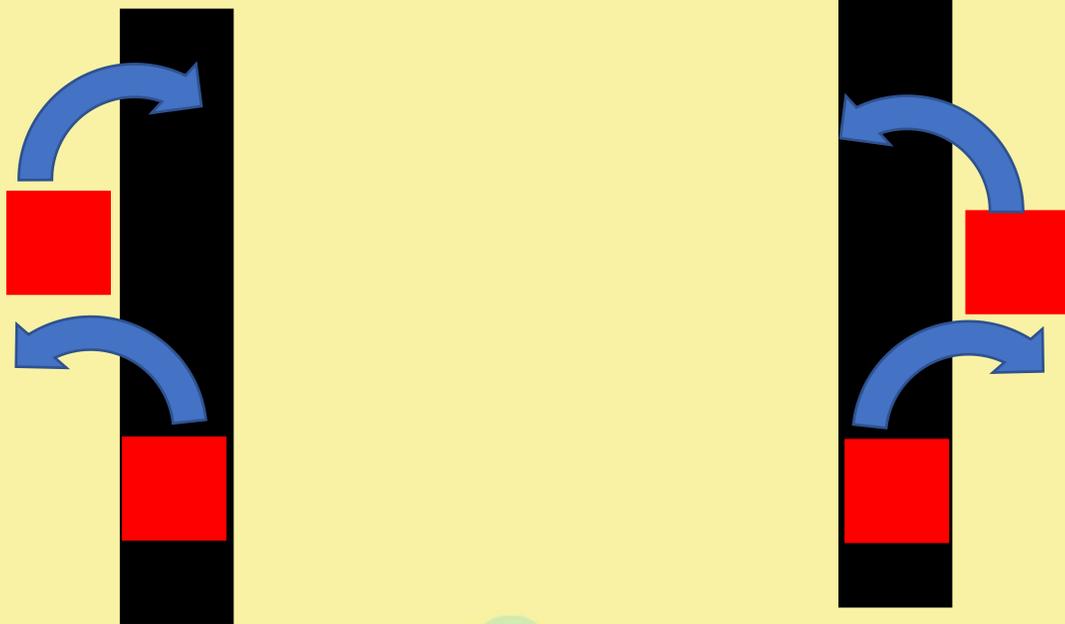


## 编程技巧1

- 单个灰度传感器的机器人巡线并不是沿着黑线走
- 而是沿着黑线与白色交界的线走(左图红色线)
- 而黑线有两条与白色交界的线，该沿着哪边走呢？

## 编程技巧1

- 沿着哪边走，取决于你在黑线上的动作，观察下图
- 现在只检测出了两个状态，因此对应的只能有两个动作



# 任务

## 参考程序

别忘记了上传机器人运行哟，否则可能不成功

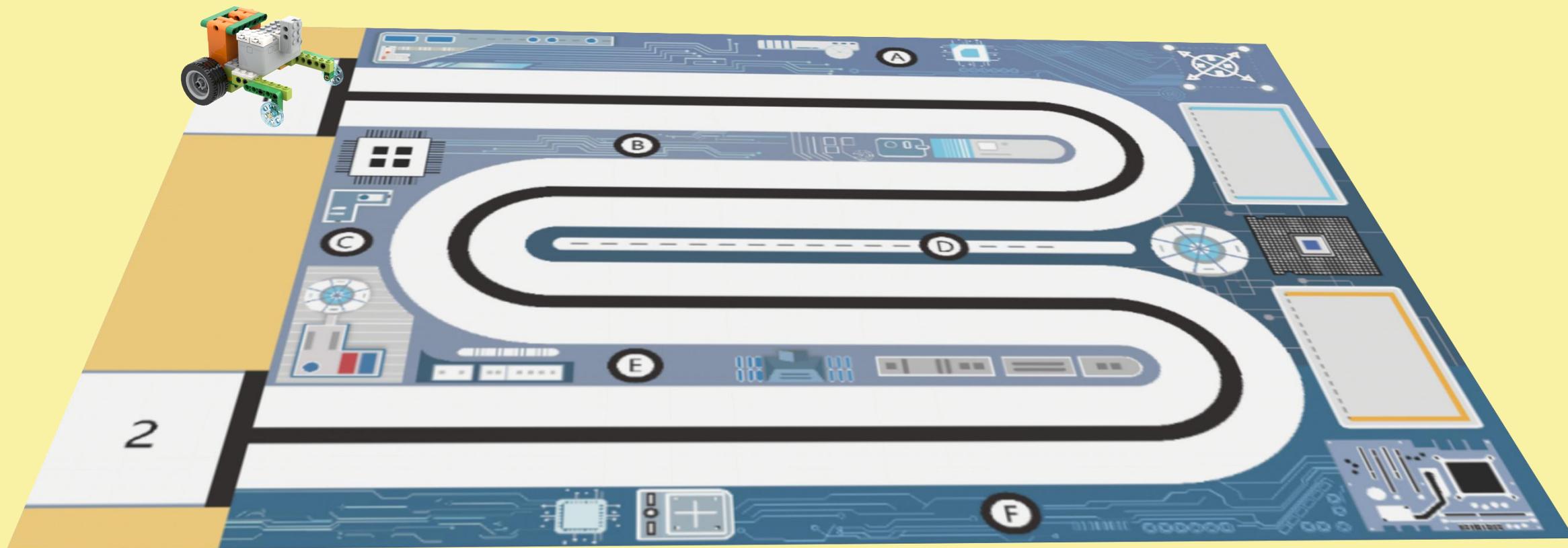
$$\text{阈值} = \frac{(\text{黑线数值} + \text{白色数值})}{2}$$

建议切换功率模式

```
当被点击
设置 1# 伺服电机的当前位置为原点
设置 2# 伺服电机的当前位置为原点
等待 0.3 秒
重复执行
  如果 1# 数字灰度传感器的值 > 15 那么
    设置 1# 伺服电机以 0 (-100~100)% 功率 来 逆时针 持续运转
    设置 2# 伺服电机以 30 (-100~100)% 功率 来 顺时针 持续运转
  否则
    设置 1# 伺服电机以 30 (-100~100)% 功率 来 逆时针 持续运转
    设置 2# 伺服电机以 0 (-100~100)% 功率 来 顺时针 持续运转
```

# 任务

- 任务2：利用灰度传感器沿着黑线走一定的距离**停下**，停在D点旁



# 任务

## 参考程序

先直走移动出1号起点

利用重复执行直到顺时针转动的电机相对角度大于2600的指令，停止循环

The diagram shows a sequence of blocks for controlling two servo motors:

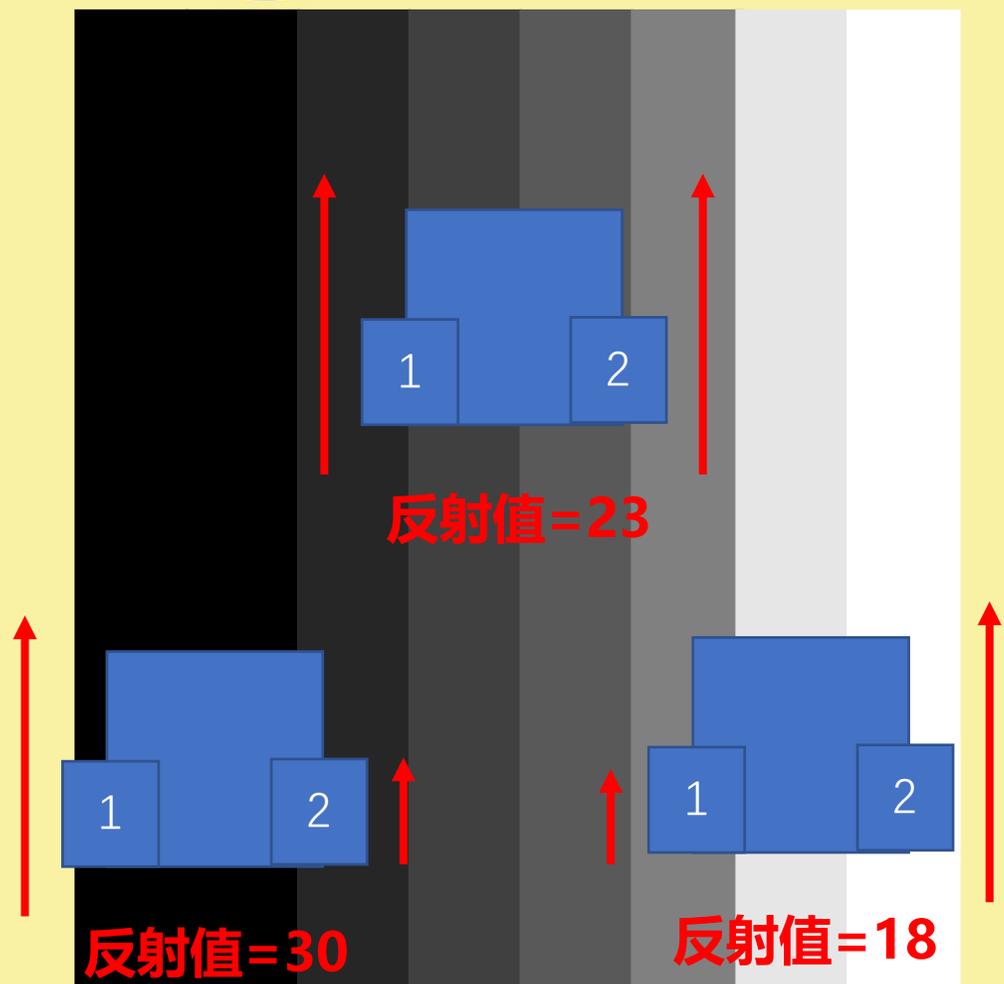
- Start:** A '当 被点击' (When clicked) block.
- Initialization:** Two '设置' (Set) blocks: '设置 1# 伺服电机的当前位置为原点' and '设置 2# 伺服电机的当前位置为原点'.
- Delay:** A '等待 0.3 秒' (Wait 0.3 seconds) block.
- Red Boxed Section:** Two '设置' blocks: '设置 1# 伺服电机以 30 (-100~100)% 速度 来 逆时针 持续运转' and '设置 2# 伺服电机以 30 (-100~100)% 速度 来 顺时针 持续运转'.
- Delay:** A '等待 1 秒' (Wait 1 second) block.
- Loop:** A '重复执行直到' (Repeat until) block with the condition '2# 伺服电机的当前 相对角度 > 2600'.
  - 如果 (If):** '如果 1# 数字灰度传感器的值 > 15 那么' (If digital grayscale sensor value > 15 then).
    - Block 1: '设置 1# 伺服电机以 0 (-100~100)% 功率 来 逆时针 持续运转'.
    - Block 2: '设置 2# 伺服电机以 30 (-100~100)% 功率 来 顺时针 持续运转'.
  - 否则 (Else):**
    - Block 1: '设置 1# 伺服电机以 30 (-100~100)% 功率 来 逆时针 持续运转'.
    - Block 2: '设置 2# 伺服电机以 0 (-100~100)% 功率 来 顺时针 持续运转'.
- End:** A '停止 全部 伺服电机' (Stop all servo motors) block.

## 附加任务

- **注意：此任务不强制要求学习，最多只需要知道调整什么参数即可。**
- **附加任务：利用比例模式进行单个颜色传感器巡线动作。**



## 编程技巧2



- 黑白交界处是一个渐变模式
- 箭头代表功率大小
- 发现反射值与电机功率之间的比例关系了吗？



## 编程技巧2

初始功率  
(20~30)

$$\text{阈值} = \frac{(\text{黑线数值} + \text{白色数值})}{2}$$

The image shows a Scratch script starting with a '当被点击' (When clicked) event block, followed by a '重复执行' (Repeat) loop. Inside the loop, there are two '设置' (Set) blocks for servos. The first block is for servo 1# and the second for servo 2#. Both blocks have three input fields for the servo's position: 25, 2, and 10. These three fields are highlighted with red boxes. Red arrows point from the text '初始功率 (20~30)' to the first box (25), and from the equation '阈值 = (黑线数值 + 白色数值) / 2' to the second box (2). A red arrow points from the text '比例值越大摆动越激烈' to the third box (10). The servo 1# block is configured to rotate counter-clockwise ('逆时针') and the servo 2# block is configured to rotate clockwise ('顺时针'). Both are set to '持续运转' (Continuous rotation) and '功率' (Power) mode.

比例值越大摆动越激烈