



巧用基础任务 提升编程技能

文_李 杨 / 宁夏银川贺兰县第四中学

在杭州萝卜圈网络有限公司出品的虚拟机器人 IRobotQ3D 入门级的“基础试玩”中，“向左走”、“向右走”是最基础的训练任务。一般，大家觉得这是基础任务，完成之后往往就开始了“汽车总动员”、“野外生存”等高难度任务的练习。如果能很好地挖掘这两个基础任务的潜能，将会有很大的收获，甚至在汽车总动员、野外生存等大型任务中都可以用到这个基本任务中的一些编程思路和技巧。学生一旦熟悉、掌握了这些编程思路，那么在复杂场景中，就可以把重点放到其他地方，便于学生理解，达到更好的训练效果。

我们在完成“向左走”、“向右走”任务时，先要搭建四轮机器人小车，如图 1 所示。编程是以前进、延时、控制左右电机转向、接着前进来完成的。所有电机为 10 倍精度直流电机，左侧电机连接端口 1、2，电机反转，右侧电机连接端口 3、4，电机正转（如图 2）。

在学生掌握了基本方法后，启发他们换一种编程思路，解决同样的问题。

思路 1：使用“计时”模块 + “条件循环”模块

车型不变，编程如图 3 所示。程序虽然长了，但是可以训练学生用定时的方法来完成这个动作。由于运行路径很短，调试很方便，所有参数基本不变。这样，学生就可以把注意力集中到使用循环上来。学生掌握后，用这种思路完成汽车总动员等复杂任务也是没有问题的。

思路 2：使用“直流电机驱动”模块

“多直流电机驱动”简单好用，但是有一个弱点：不能用参数给定电机转速，换句话说，电机的速度是固定的，不能用程序设定。而“直流电机驱动”则可以用程序改变电机转速。美中不足的是：需要给每个电机赋值，稍显麻烦。

在这里，使用了一个全局变量：速度。在程序开始时，先给速度赋初值，并赋值给每个电机。

以左转为例，左侧电机需要反转，那么速度就是 -10，所以用了一个常量 0 减去 10，就得到了需要的值“-10”；右侧电机速度与全局变量相同，程序如图 4 所示。转弯后，继续直行，与程序开始处基本一致，只需调整时间参数即可。

对于汽车总动员或野外生存等复杂任务，如

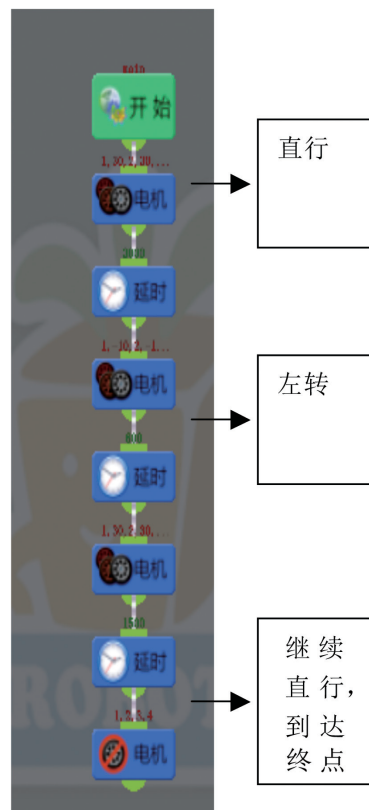


图 2 最基础编程

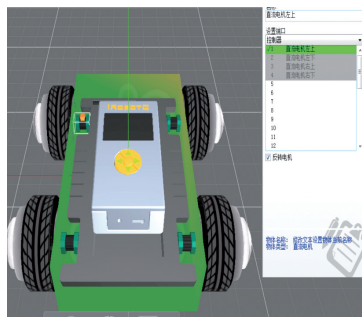


图 1 搭建的机器人小车

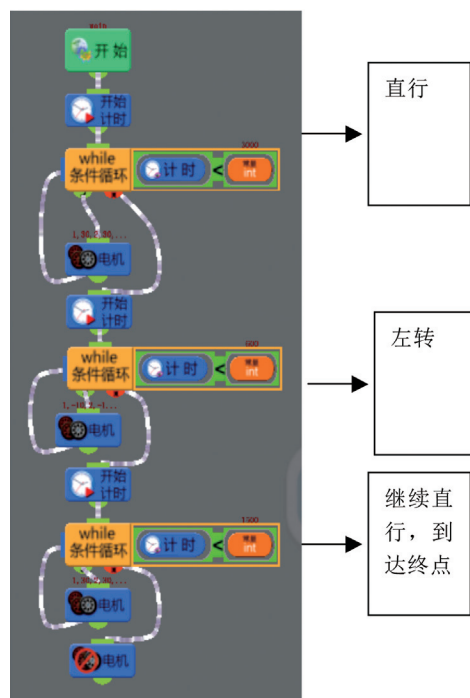


图3 使用“计时”+条件循环编程



图4 左转的程序部分

果需要根据传感器（如距离传感器）的输出来改变电机的转速，就需要学生掌握这种编程方法。虽然很麻烦，但是可以自动控制每个电机的速度，完成复杂的动作。

思路3：使用指南针传感器完成转向

首先，我们给机器人小车加上指南针传感器并给它相应的端口。左转的程序如图5所示。

指南针传感器使用得当，程序的调试将会比用控制电机的方法来转向更容易。因为，如果速度变化，时间就需要跟着变，控制电机的程序需要反复调试。在汽车总动员等大型任务中，在需要用提速来提高成绩时，用指南针的程序则大大节省了调试时间。

思路4：用“逐渐加速”、“逐渐减速”更稳定地控制机器人小车的姿态

在程序开始处，给定初始速度为0，然后在3000毫秒内，如果速度的值还没有达到50，就每次增加2，直到速度加到50。学生掌握了这种思路，就可以很稳定地给机器人小车加速，有效避免开始跑翘头、刹车翘尾发生，更好的控制机器人的姿态。

思路5：用子程序完成任务

以上程序中，电机的速度等小段程序与主程序放在一起，不仅难以阅读，修改起来也很麻烦。这些部分可以从主程序中分离出来，变成子程序，将大大增加程序的可读性，减轻编程难度，方便程序调试。例如，前进程序段可以变成前进子程序，左转程序段可以变成左转子程序。这样主程序整体就显得比较简洁。如果什么地方出现问题，直接去相应的子程序修改就可以了。其实，这样的编程思路就是完成大型、复杂任务的编程思路。如果继续把前进部分再单独做一个子程序，那么主程序就更加简洁，更容易调试。学生在完成小任务时，采用完成大任务的程序思路，就比较容易的调试成功。如果熟悉了，可以很快上手复杂任务。这样也许比直接在复杂任务中训练效果好很多。

虚拟机器人作为实体机器人以外的另一种展现形式，有着很多优点。基础试玩部分，免费注册账户就可以登录，可以让更多有兴趣的学生参与到这项活动中来。

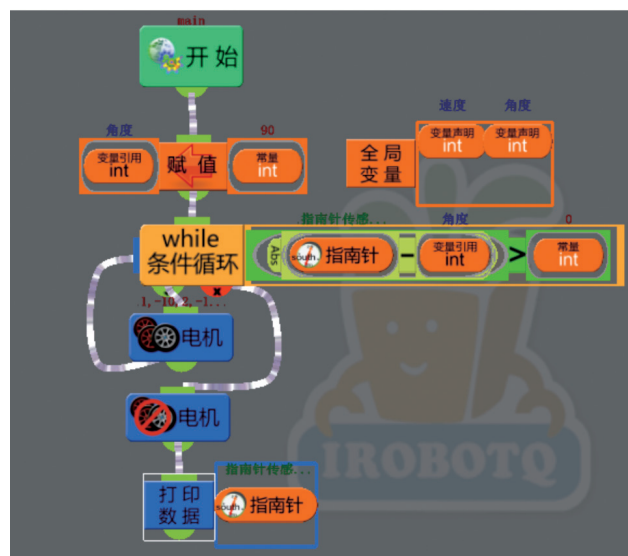


图5 用指南针转弯的程序

在平时的训练过程中，将一些基础任务限定编程方法来加大难度，或者引导学生用不同的方案来实现同一任务，既增加了学生的学习热情，又可以培养学生们的探索精神，还可以把知识的难点消化在简单的任务中，让简单任务不简单。基础任务，我们也可以玩的很High！