

组合逻辑

目的：

当被要求设计一个控制系统时，你可能首先需要明确需求，这可能是一个问题的描述或对一个输入预期的输出结果。为了开始设计控制系统，你必须把问题转化为数字逻辑电路。第一步是确定各种输入和输出，大多数人会把将设计规范转换为真值表作为设计工作的开始。这个真值表将显示控制系统的每个输入经过控制系统后会输出什么结果，组合逻辑描述了你的系统将如何处理这些输入。一旦真值表被设计出来，它们就可以被转换成布尔方程，这使得逻辑电路的创建过程更容易。

这节您将学习将规范转换为真值表，将布尔表达式转换为图形化的逻辑表达。在上一节中，您已经学习了真值表和简单的布尔表达式。

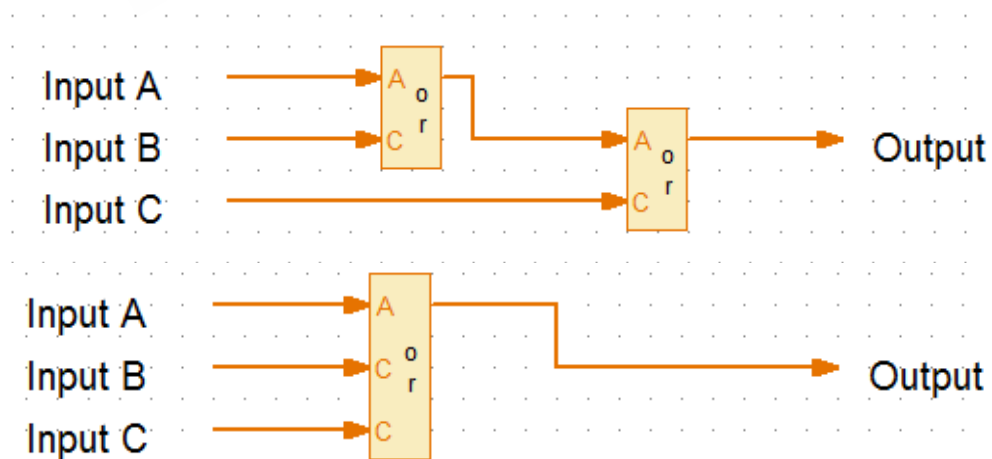
设备：

RoBo Pro 软件、TXT 控制器、灯、导线

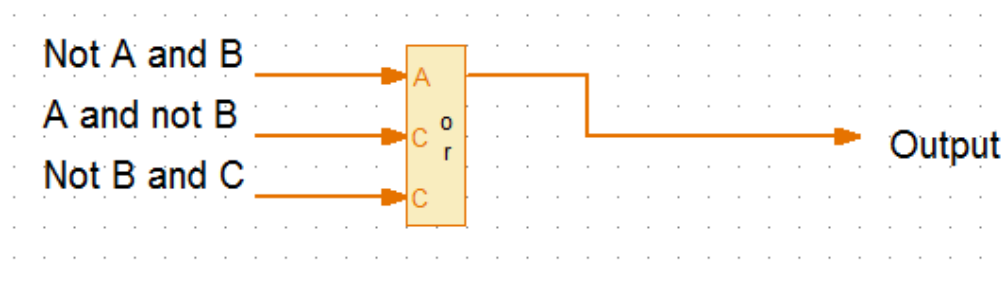
过程：

学习将设计规范转换为逻辑要求你掌握几个中间步骤。首先我们来看一下布尔表达式是如何转换成逻辑电路的。

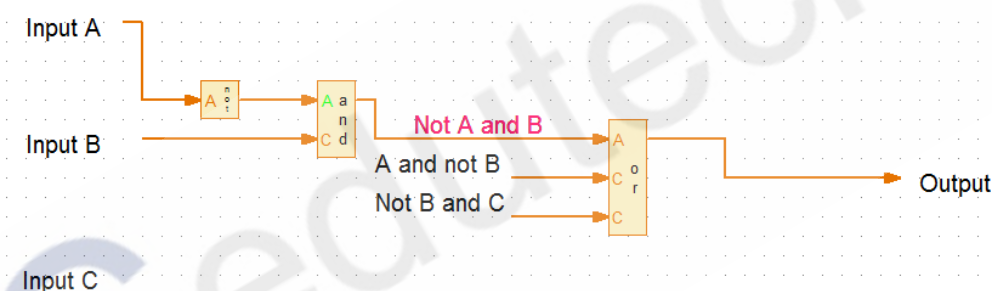
下面这个问题的设计规范定义了三个独立的输入和一个输出。当任何一个输入为高时，要求产生输出。该表达式被表示为 $A + B + C = Y$ 。这将被读作 A 或 B 或 C 输出 Y。在上节中，我们已经了解了包含 2 个输入和 1 个输出的“或”门。下图是两种不同的图形化表示。



你会注意到上图中有一个是三输入的“或”门。它只需要把“Or (或)”模块的输入端定义为“3”即可。现在如果给出一个布尔表达式如 $A^{\sim}B + AB^{\sim} + B^{\sim}C = Y$ ，我们需要修改这个程序，依然使用一个三输入的“或”门模块来实现这个组合逻辑。



我们先创建其中的一个输入，从“Not A and B”开始，参照下面图的图例。你可以看到为创建“Not A”，我们采用了为输入 A 连接一个“nor (非)”门模块。当输入 A 为低电平时，“nor (非)”门将输入信号取反变为高；当输入 B 为高电平且输入 A 为低电平时，通过后面的“and (与)”门得到的输出为高。

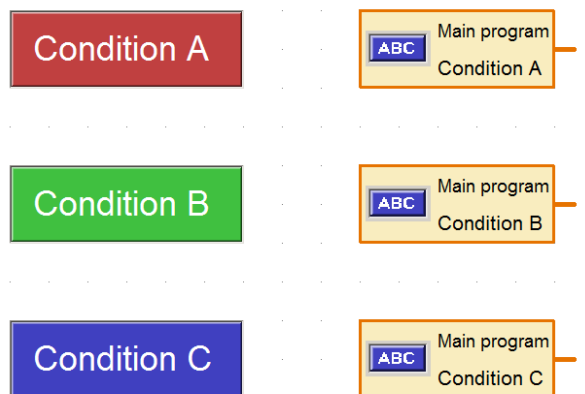


参照“Not A and B”电路，我们另外的两个输入“A and not B”、“Not B and C”分别创建电路。

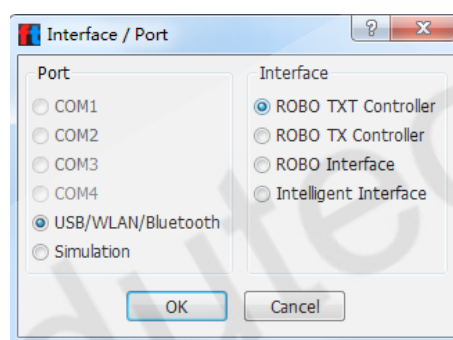
组合逻辑允许你使用 AND (与)、OR (或) 和 NOT (非) 这些基础的逻辑门，依据布尔表达式创建逻辑电路。现在我们建立三个按钮控件为上面的逻辑组合提供实际的信号输入。

从“操控模块 (Operating elements)” -> “控制模块 (Control elements)”子分类中，将“按钮 (Button)”模块拖拽到编程窗口中。通过复制一共得到三个按钮控件。为每个按钮控件设置唯一的名称，同时勾选“Pushbutton switch (按钮开关)”这个选项。

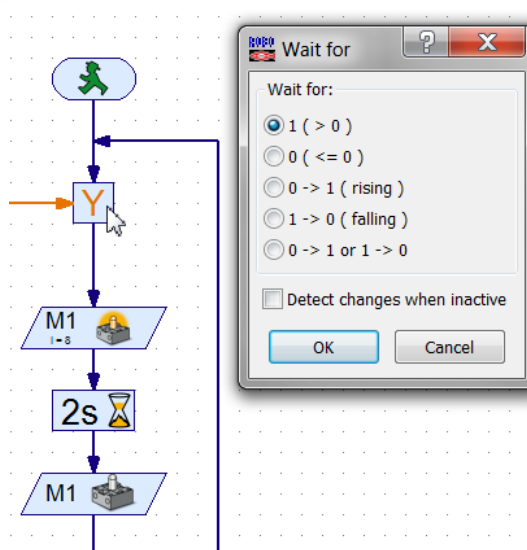
再从“编程模块 (Program elements)” -> “输入输出 (Inputs,outputs)”子分类中，将“面板输入 (Panel input)”模块拖拽到编程窗口中，建立 3 个，并分别将它们分配给 3 个按钮控件。参照下图所示：



我们通过 USB 线将 TXT 控制器与电脑相连，在 TXT 控制器上的 M1 输出端连接一个灯来显示逻辑结果。之后点击工具栏中的“COM / USB”图标，如下图所示配置连接方式。



在下面的程序中，你可以看到组合逻辑的输出通过一根黄色的数据线连接到“等待(wait for)”模块。我们将“等待(wait for)”模块的信号设置为等待高电平：1 (> 0)，当这个输入条件出现时，该模块才会允许程序流程进入下一步，而只要输入的信号逻辑为低时，它就会等待。根据后面的程序模块，我们知道当“等待(wait for)”模块的输入信号为高电平时，程序向下执行，控制 M1 灯点亮。



完整的程序如下图所示，在线运行这个程序，点击按钮控件，观察灯的亮灭。

