

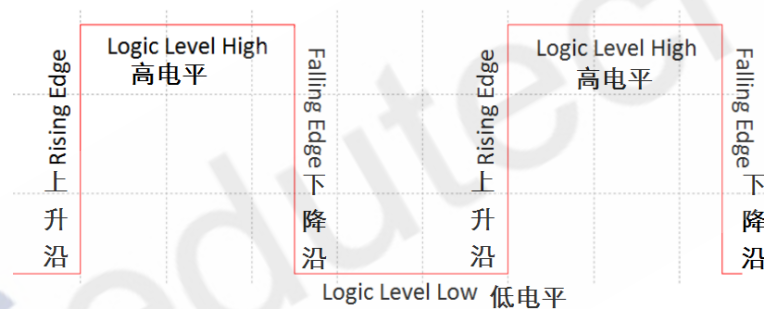
边沿触发与电平触发

目的：

触发是指程序在接收到预期的数字信号输入时激活程序的一个分支的过程。这意味着一旦触发器被激活，程序将执行一个功能。触发方式有两种。

首先是电平触发，当出现所需的输入电平时，程序就会发生分支跳转。电平触发可以是一个高电平或低电平。

当触发是从低电平变为高电平（上升沿）或从高电平变为低电平（下降沿）时，这种触发就是边沿触发（瞬间触发）。



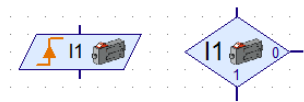
上图描述了电平改变状态时的逻辑波形。逻辑电平分为低电平（0）和高电平（1）。电平之间的转换如波形图中的边沿。上升沿是指从低电平变为高电平，下降沿是指从高电平变为低电。为了演示边沿触发和电平触发之间的区别，下面将构建一个简单的电路模型来进行测试。

设备：

软件 RoBo Pro、TXT 控制器、直流电源、开关、灯

过程：

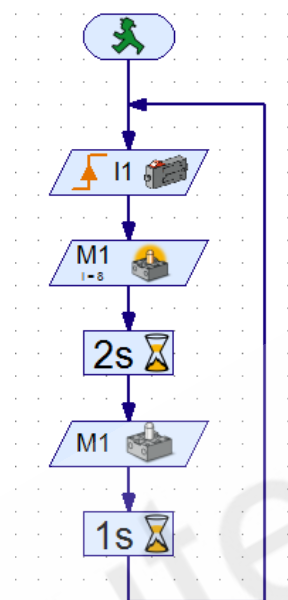
有两个模块决定你的程序如何处理外部输入的触发信号。下图左边的模块是“等待输入（Wait for input）”模块，如图所示，它被设置为等待上升沿（边沿触发）。下图右边的模块是“数字分支（digital branch）”模块，根据开关的状态电平（电平触发），它会选择相应的分支路径。



进行实际的硬件连接，将触发开关以常开触点（端子 1 和 3）的接法连接到 TXT 控制器上的 I1 输入端。将灯连接到控制器上的 M1 输出端。

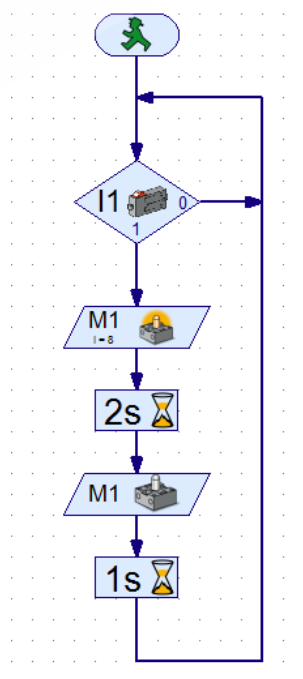
本节中控制器上的硬件连接不会改变，但程序会略有不同。观察程序的运行情况将有助于你理解两种触发方式的不同，并有助于你在以后的程序调试中解决问题和排除程序故障。

我们从边沿触发的程序开始。开始一个新的程序，并参照下图画出程序。



运行上面的程序并按住开关 15 秒钟，观察灯泡的闪烁情况。

下面改编你的程序，将“等待输入（Wait for input）”模块更改为“数字分支（digital branch）”模块。

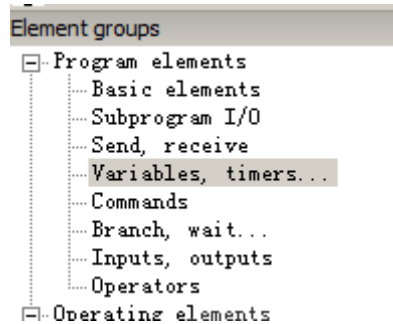


再次运行上面的程序并按住开关 15 秒钟，观察会发生什么。描述灯泡的闪烁情况，并注意与使用“等待输入（Wait for input）”模块时的差异。

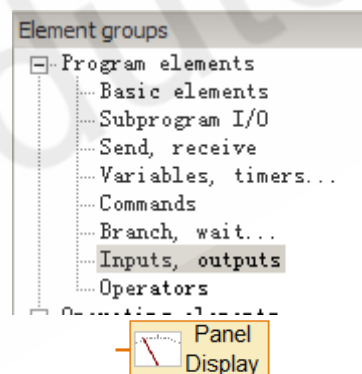
接下来我们将使用一个经典的计数显示功能来真实地显示两种触发方式的不同。

在软件中，设置使用级别为“Level 3：Variables（级别 3：变量）”，开始一个新的程序。

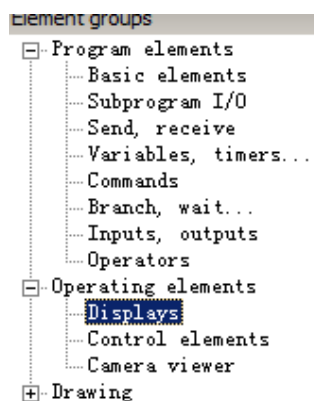
从“编程模块（program elements）”->“变量，定时器（variables, timers...）”子分类中，拖拽一个“全局变量（variables（global））”模块到右侧编程窗口中。



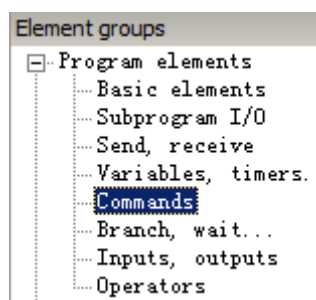
从“输入、输出（inputs、outputs）”子分类中，拖拽一个“面板显示（panel display）”模块到右侧编程窗口中。



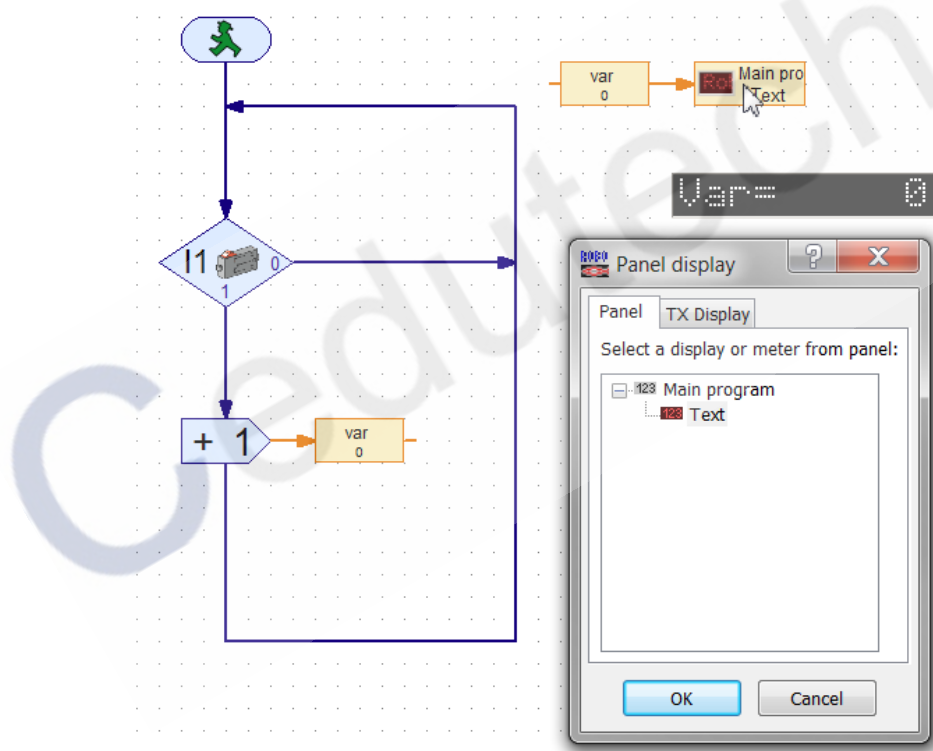
从“操作模块（operating elements）”->“显示（displays）”子分类中，拖拽一个“文本显示（text display）”模块到右侧编程窗口中。



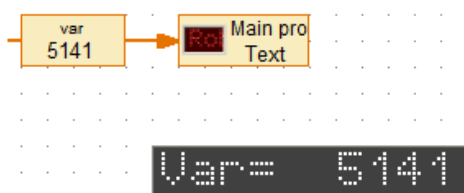
从“编程模块 (program elements)” -> “指令 (commands)” 子分类中，拖拽一个“加 (plus)” 模块到右侧编程窗口中。



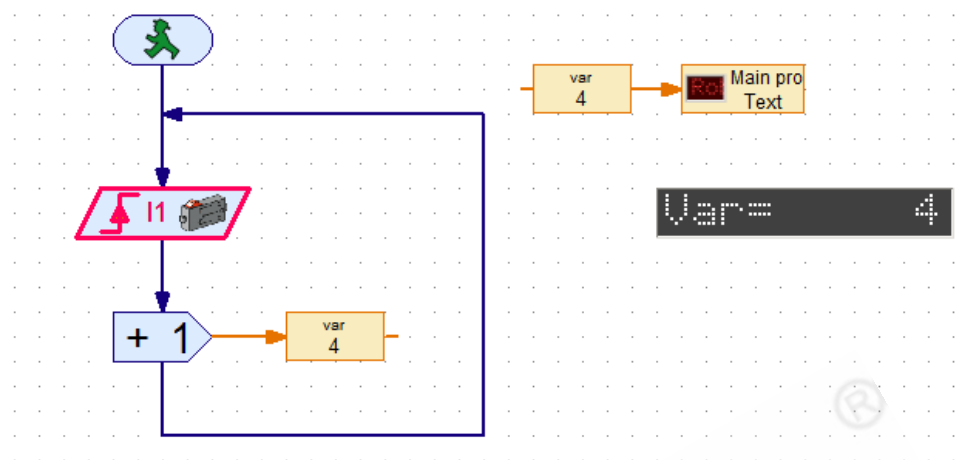
参考下图编辑程序。鼠标右键单击“面板显示 (panel display)” 模块，将其分配给编程窗口中的文字框 (text) 模块。注意：这个操作的意思就是将两个模块进行关联。



在线运行该程序，然后尽可能快地按下并释放开关。你会看到文本框中显示的数字，如果你按下与释放开关之间的时间变长，你只会看到文本框中出现问号 (因为数值变大会超出文本框中默认的数位显示宽度)。



将上面程序中的“数字分支 (digital branch)”模块替换为“等待输入 (Wait for input)”模块。再次运行程序并单击开关。



观察文本框中记录的数值与你触发开关的次数之间的逻辑关系，解释你所观察到的现象。体会“数字分支 (digital branch)”模块与“等待输入 (Wait for input)”模块的所代表的触发类型的不同。