


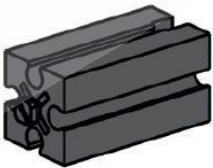



《摄像头的颜色探测》

功能介绍

在进行本功能测试前要注意，外部光源对摄像头的颜色识别是有影响的，在编程测试期间请保证不变的外部光源，否则测试数据会受到影响，还有当摄像头前的探测区域没有明显反光时，识别颜色也较为准确。最好提供一个黑色、白色或者单色的背景色，被测物体的颜色最好也是单色。颜色探测时被测物要保持稳定。

实验组件：

 Link 15 X2	 Building Block 30 X7	 USB Camera X1
--	--	---

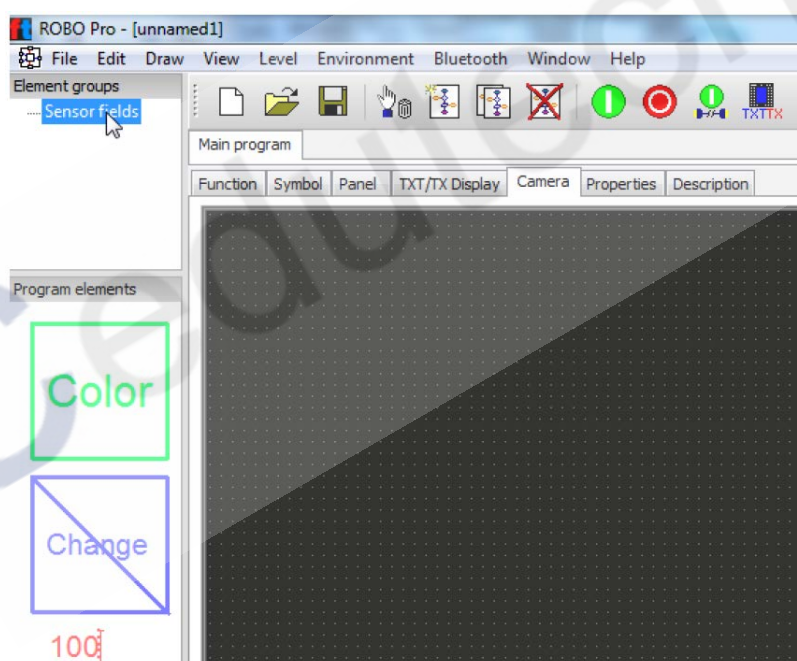
按照下图组装结构：



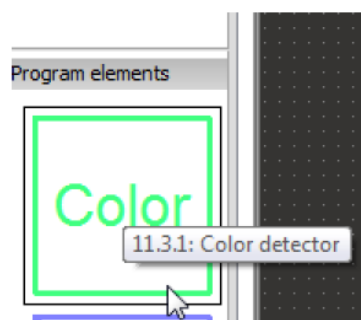
按照下图安装上摄像头并组装好试验台：



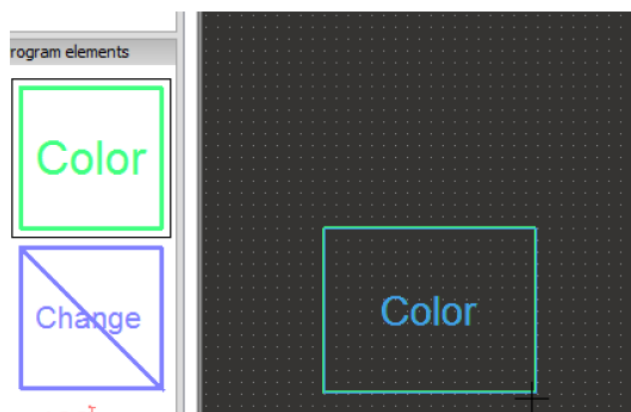
在软件中新建一个程序，打开摄像头窗口。点击左上角的“传感器域”，在下面的程序模块中将显示出 5 种摄像头探测模块：



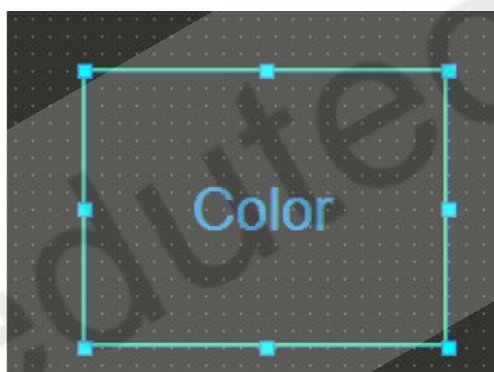
本实例中，选择“颜色探测”模块：



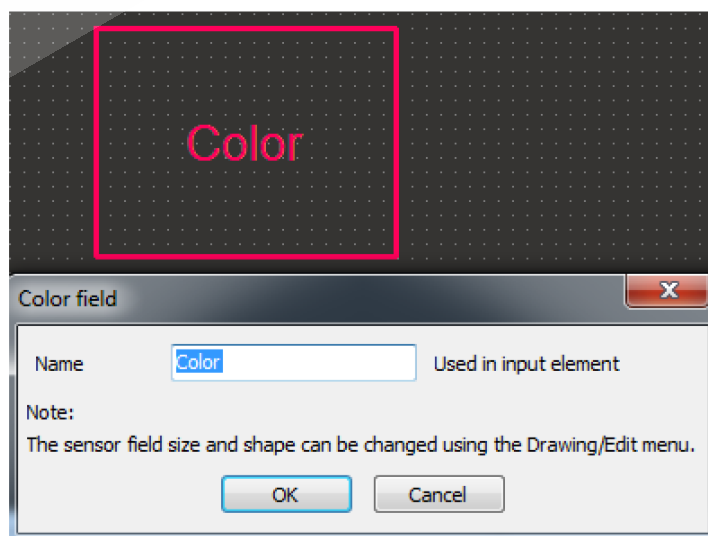
单击“颜色探测”模块后，鼠标形状变为笔形，现在在视频监控窗口中绘制监测区域：



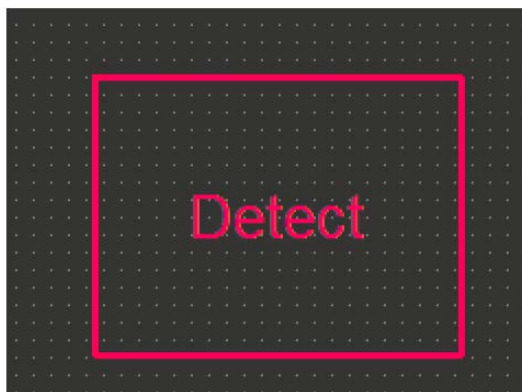
可以拖动监测区域框四周的锚点以改变形状，确定形状后，鼠标点击监测区域以外的地方以确定设置，鼠标变回光标形状：



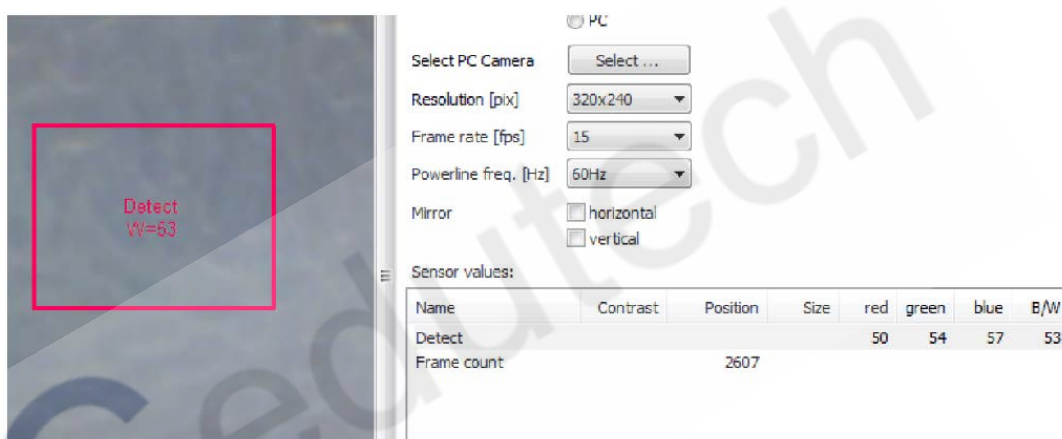
右键单击设置好的监测区域，弹出属性面板，在这里给监测窗口设置一个唯一的名称，这里我们将名称改为“Detect”：



确认后，你会看到，监测区域框内的名称变更为你设置的“Detect”：



勾选“实时预览”复选框，你将看到摄像头拍摄的实时画面：



我们的电脑是使用 RGB 颜色标准的。RGB 颜色标准是指每种颜色都是由：Red（红）、Green（绿）、Blue（蓝）三种颜色按照不同比例混合而成。R、G、B 每种色各分为 256 阶亮度，数值从 0 到 255 表示，一些颜色的 RGB 表示如下：

颜色名称	红色值 Red(0 - 255)	绿色值 Green(0 - 255)	蓝色值 Blue(0 - 255)
黑色	0	0	0
蓝色	0	0	255
绿色	0	255	0
青色	0	255	255
红色	255	0	0
洋红色	255	0	255
黄色	255	255	0
白色	255	255	255

此外，监测系统还提供一个 B/W 属性来表示不同颜色的总体亮度值。

现在你可以从电脑上打印几张不同颜色的纸，并将其 RGB 值和 B/W 总体亮度值记录在其上，以备后面使用：



程序测试

切换到程序界面，从“输入，输出”子分类中拖拽“摄像头输入”和“面板显示”模块到程序窗口中：



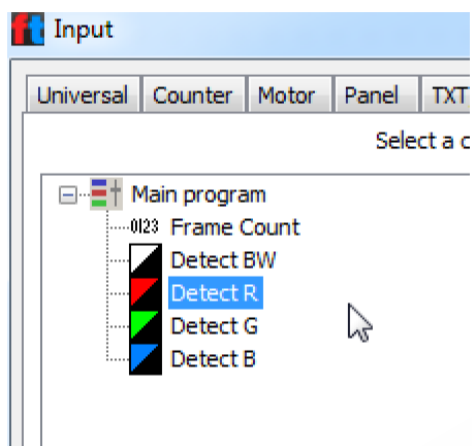
从操作模块下的“显示”子分类中，拖拽一个“文本显示”模块到程序窗口中：



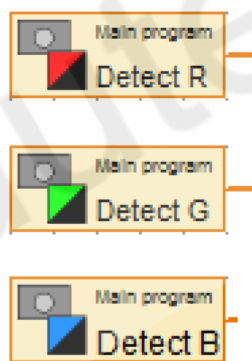
复制上面的三个模块如下：



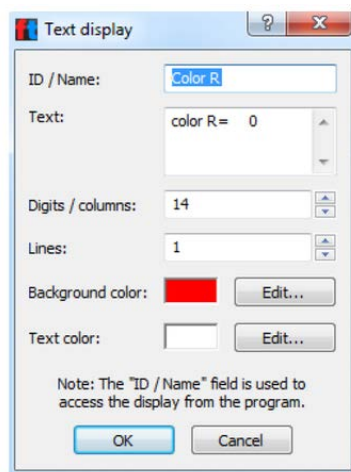
右键单击第一个“摄像头输入”模块，在其属性面板上，将其指向“Detect 监测区域”中的“R”属性：



重复这个操作，将其他的两个“摄像头输入”模块分别指向“Detect 监测区域”中的“G”属性和“B”属性：



右键单击第一个“文本显示”模块，在其属性面板中，设置其名称为“Color R”，修改其文字显示内容为“Color R= 0”，然后将背景色改为“红色”：



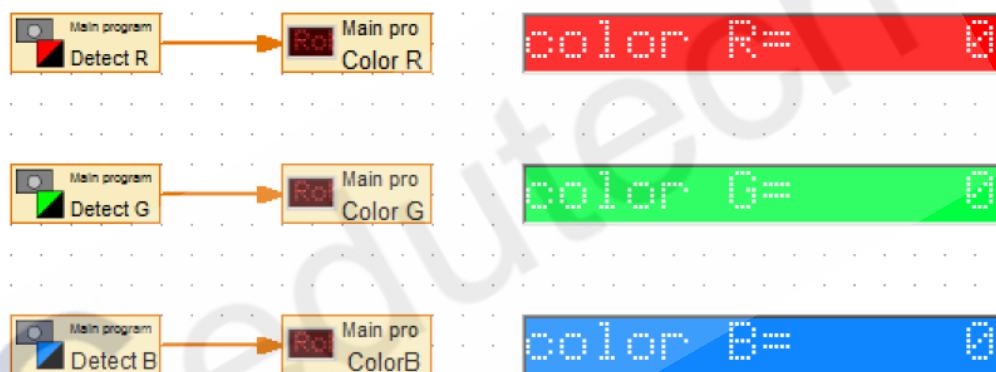
之后重复这个操作，将其他两个文本框分别设置显示“绿色”和“蓝色”：

color R= 0

color G= 0

color B= 0

接下来，分别设置三个“面板显示”模块指向上面这三个“文本显示”模块，使程序显示如下：



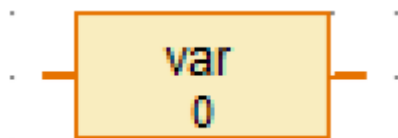
现在运行程序，将不同颜色的彩纸放在架子上，观察测得的数据应该接近彩纸上记录的RGB色值。

接下来我们进一步修改这个程序，使其功能更完善：

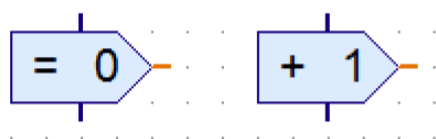
拖拽“起始”模块到程序窗口中：



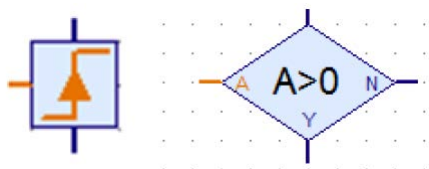
从“变量、定时器...”子分类中，拖拽2个“变量”模块到程序窗口中：



从“指令”子分类中，拖拽“赋值”模块和“加”模块到程序窗口中：



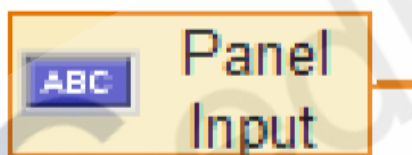
从“分支、等待...”子分类中，拖拽“等待...”和“带数据输入的分支”模块到程序窗口中：



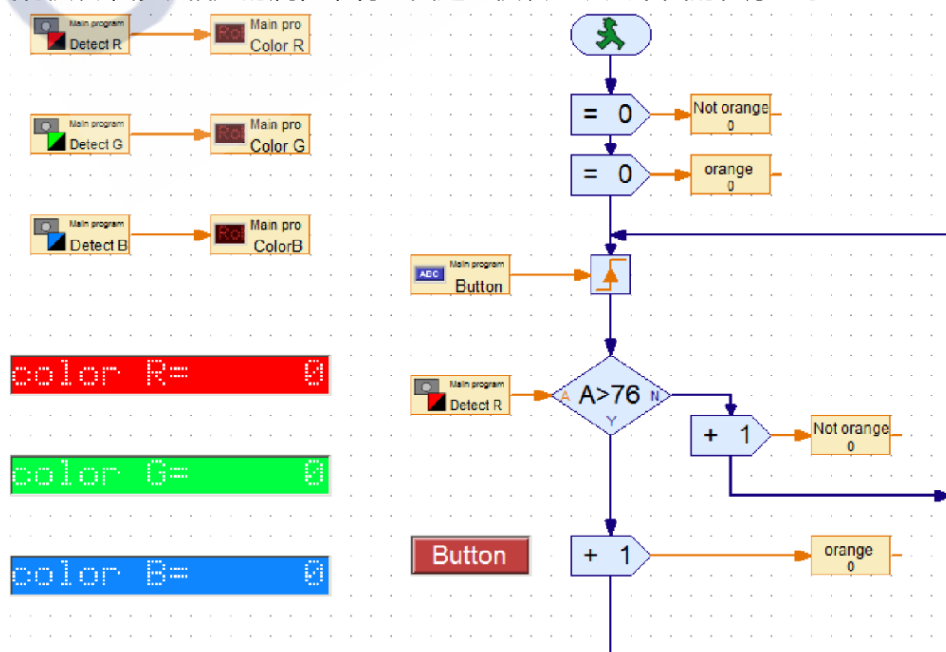
从“操作模块”中的“控制模块”子分类中，拖拽“按钮”模块到程序窗口中：



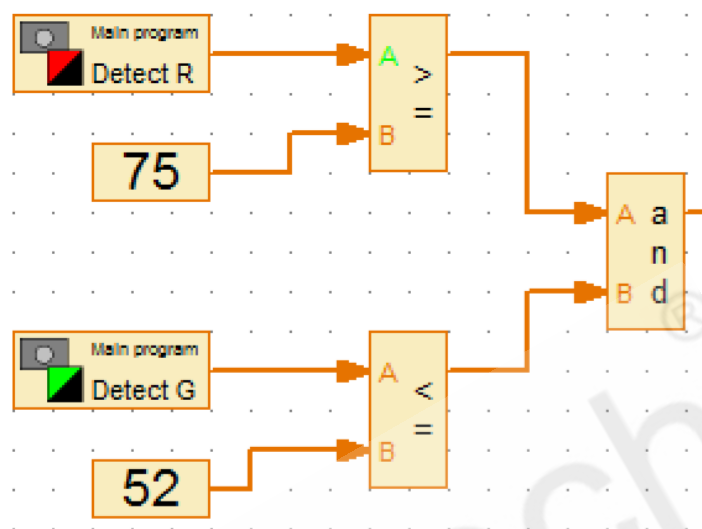
从“输入输出”子分类中，拖拽“面板输入”模块到程序窗口中，并在其属性面板中设置指向“按钮”模块：



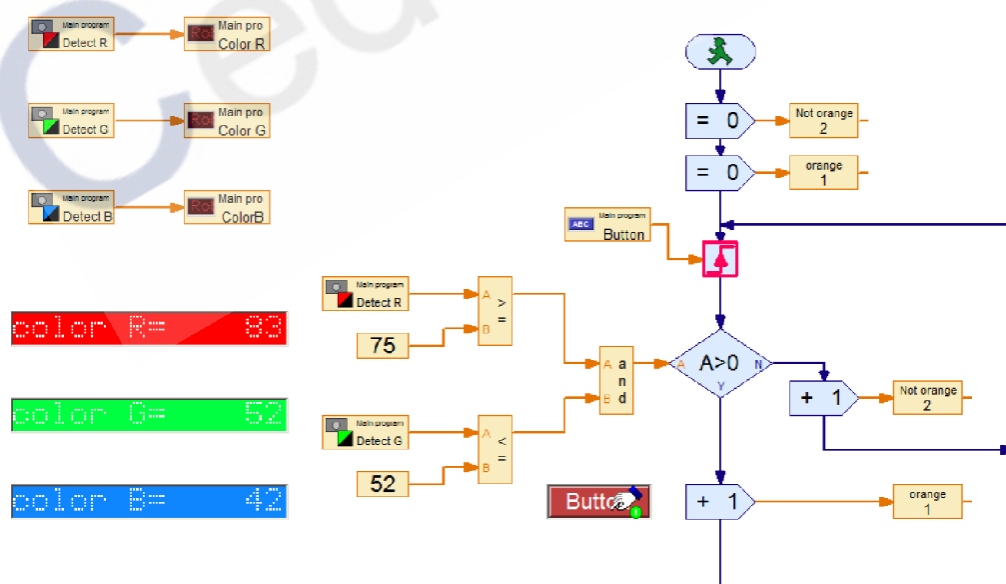
复制模块，修改相应的属性，将上面这些模块组织成下面的程序显示：



在上面这个程序里，我们用 RGB 中的“R”值作为分支判断的数据输入值，我们修改分支判断的条件数值，以便判断每次放置的颜色卡片是否是我们要监测的颜色。但是每种颜色的 RGB 三个通道的色值有时是非常接近的，因此用一个颜色通道的数值来做条件判断往往不够准确，这时我们可以考虑用多个条件来做颜色判断：



如果在自然光干扰较小的环境中，如果颜色纸张的测试数值较为稳定，那么我们可以用上面的这种多通道颜色的“等值”判断来确认颜色，但是如果数值不够稳定，这种颜色的判断也是不准确的会存在误差。完整的测试程序如下：



实验

有多种逻辑来选择你想要的颜色，请尝试不同的方法编程来实现颜色筛选功能。