

UI 布局工具

1 工具简介

UI 的开发一共用到两个工具，一个是 UI 布局工具，另一个是资源生成工具。

1.1 UI 布局工具

UI 布局工具用于 UI 布局的设计，用户使用此工具创建页面、图层、布局和控件，工程数据将会保存在 xxx.json 文件中。UI 工具的程序为：



1.2 UI 资源生成工具

UI 资源生成工具用于 UI 素材资源的生成，生成的资源文件将要被下载到 FLASH 中供程序使用，生成的文件为 project.bin、result.bin 和 result.str（文件的作用将在后续说明）。资源工具的程序为：



2 UI 布局工具使用说明

2.1 界面说明

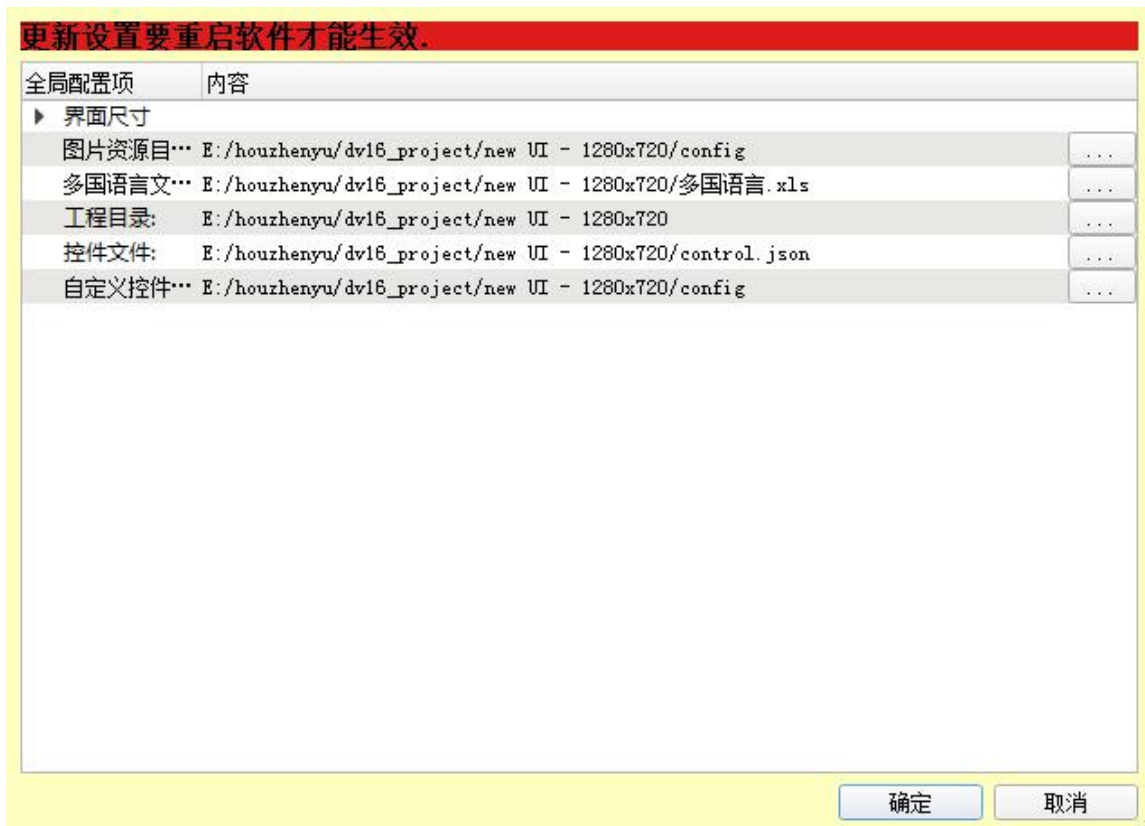
工具栏：

打开工程，可以看到窗口上方的工具栏，如下图：



- 新建工程
新建一个 UI 布局工程。

- 打开工程
打开一个 UI 布局工程。
- 保存工程
保存当前 UI 布局工程，每次修改布局后需要先保存工程再使用资源工具生成资源文件。
- 另存为
将当前工程另存为一个新工程文件。
- 新建页面
新建一个 UI 页面，页面为 UI 的最顶层。
- 删除当前页
删除当前选择的 UI 页面。
- 截屏
截取当前屏幕画面并保存。
- 全局设置
设置该工程的环境参数，每次设置完成后都需重新打开工程。



图片资源目录：此目录包含了 UI 工程所需要选择使用的所有图片资源文件存放目录，包括 PNG 和 JPG。

多国语音文件：此选项需要选择多国语言 XLS 文件。

工程目录：选择 UI 工程所在的目录。

控件文件：选择 control.json 文件，不需要修改。

自定义控件目录：不需要修改。

- 工程缩放

将当前分辨率缩放放到指定分辨率，工程布局按照缩放比例进行变换。



- 关于

查看当前工具版本。

控件列表：

控件列表用于选择在页面中新建某种控件，然后对其进行布局。



布局结构树状图：

工具最左边的列表为布局结构树状图，此图能够反映当前页面的布局结构，包括图层、布局和控制件。



对树状图的成员进行右键点击，可以进行删除、隐藏、复制、粘贴、移动等操作。隐藏操作是对某控件进行视觉隐藏，便于布局时的操作，对 UI 布局本身并无影响。隐藏后的控件坐标的眼睛会闭合并且控件会在页面上消失，移动操作可以将布局的上下层关系进行重新布置。



控件属性：

当新建了一个控件并选择它后，控件列表下面就会出现一个控件属性面板，此面板用于设置该控件的各种属性，包括坐标、背景色、背景图片等。



- ID 号
为该控件分配的名称，工具会默认分配该值，用户也可以根据自己喜好更改该名称。
- 对齐方式
为图片或者文字在控件内显示的对齐方式，有左对齐、右对齐和居中。
- 默认隐藏
默认为 **false**，就是该控件默认会跟随父控件一同显示出来，当设置为 **true** 时，改控件将不会跟随父控件进行显示，比如一个布局内分布了一个“状态栏”和一个“菜单”，由于菜单并不是一开始就要显示出来的，而是在按键按下时才会调出菜单，因此“菜单”控件就设置为默认隐藏 **false**。

- 位置坐标
位置坐标为该控件在布局中的相对坐标。
- 背景颜色
每个控件都可以设置一个背景颜色。
- 背景图片
每个控件都可以设置一个背景图片。
- 内边框线
内边框线即控件矩形边框，此处可以设置边框 4 个边的像素宽度和颜色。
- 图层颜色类型
该参数只有图层才具有，YUV 对应 JPG 图片，RGB256、OSD8 和 OSD16 对应 PNG 图或 BMP 图。YUV 对应硬件的 IMAGE 层，RGB256、OSD8 和 OSD16 对应硬件的 OSD 层。

页面大纲：

页面大纲显示着各个页面的缩略图。



2.2 页面

页面是一个布局的最顶层，要进行 UI 布局，都需先建立一个页面。每个页面的缩略图都会在大纲中被显示出来。点击**新建页面**新建一个页面，单击**删除当前页**删除当前页面。

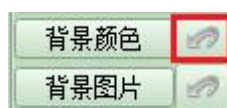


2.3 图层

新建了页面后，需要新建图层，所有控件或布局都将放在图层上。点击**图层**并拖动到绘制面板上新建一个图层，并设置它的图层颜色类型、坐标参数等属性。



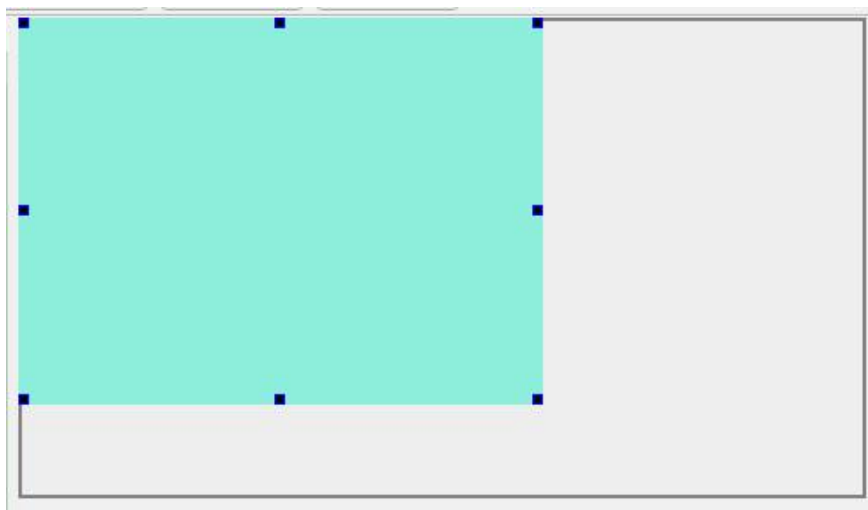
如果不需要背景颜色，可以点击其右边的箭头来删除背景颜色，背景色被删除后，箭头变为灰色不可选状态。



当图层用于显示 JPG 图时，需要选择为 YUV 类型，当图层用于显示 PNG 图时，需要选择为 RGB256、OSD8 或 OSD16 类型。



如下图所示，点击**图层**后拖动到绘制面板，在面板上单击并拖动图层锚点调整图层大小和坐标位置。蓝色区域为新建好的图层。



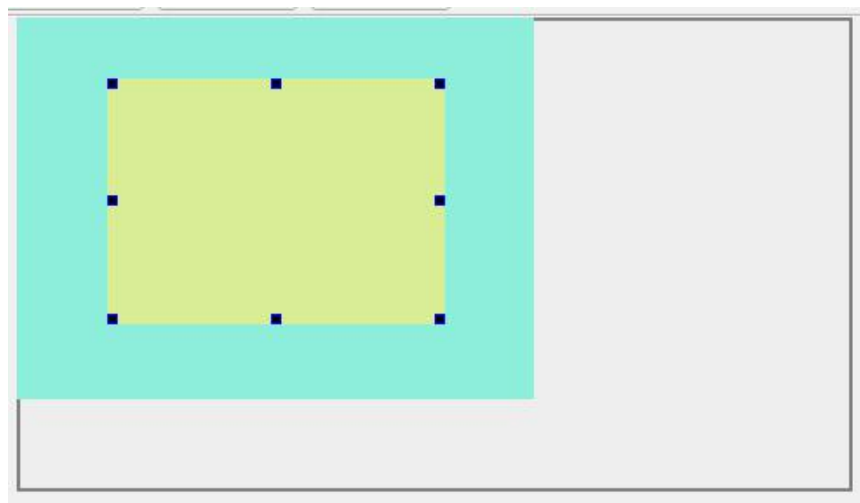
图层也可以添加其背景图片。

2.4 布局

新建了图层后，下一步需要先建立一个布局，控件需要用布局作为承载。先选择刚刚新建好的图层，然后点击**布局**并拖动到图层上，即可新建一个布局。



然后在图层中会出现一个矩形区域，该区域就为新建好的布局区域，调整好布局的位置和大小，如下图，黄色区域为布局。布局也可以设置背景颜色和背景图片。

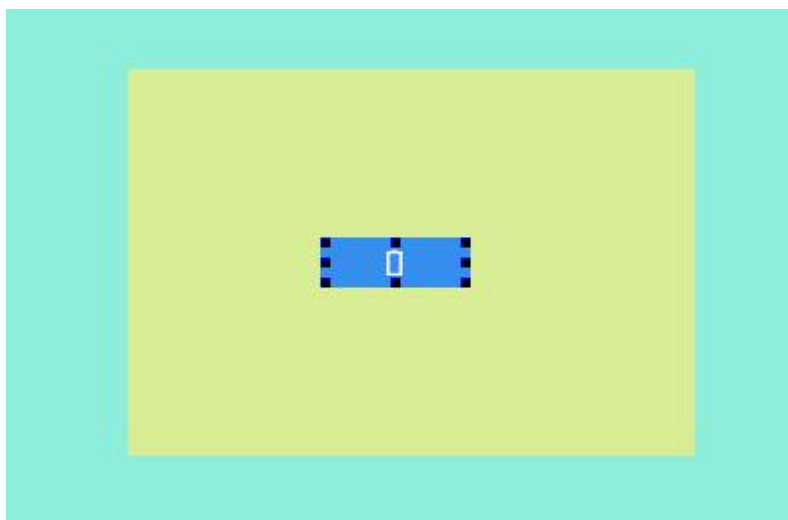


2.5 电池电量

选择一个布局，点击选择**电池电量**并拖动到布局上，即可新建一个电池控件。



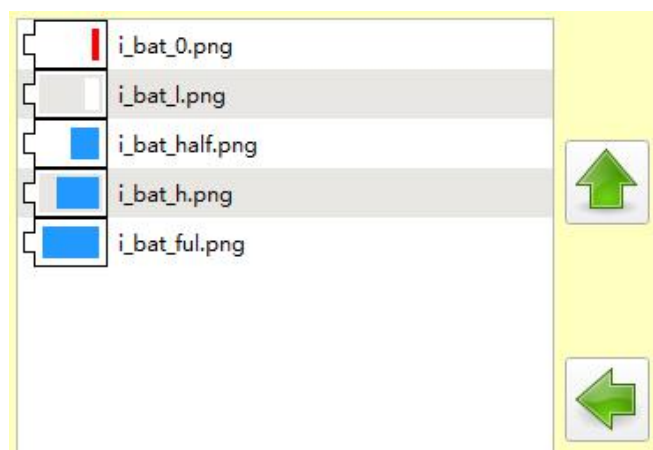
布局上出现一个电池控件矩形区域，根据需要调整好电池控件的大小和位置及其他属性。



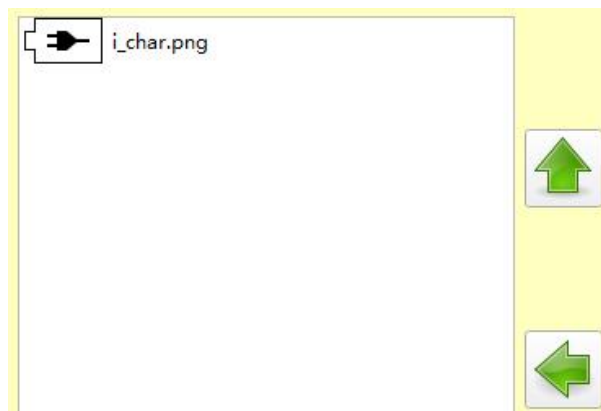
电池属性列表有两个特有属性，分别是**电量图片列表**和**充电图片列表**。下拉箭头为默认第一次显示的图片。



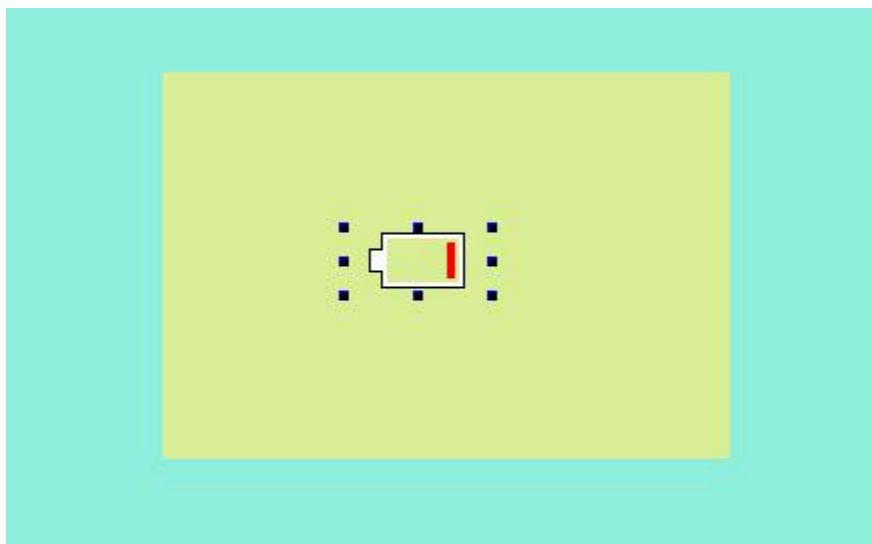
电量图片列表需要按照电量从 0%到 100%的顺序选取图片，图片数量无限制，UI 内核会根据百分比自动计算电量对应的图片。单击电量图片列表选择右边栏图片到左边栏：



充电图片列表需要选择充电状态下的图片，如果选择多个图片，系统将顺序显示，点击选择：



点击背景颜色去掉电池的背景色，并调整电池布局区域的大小，调整后效果如下图：



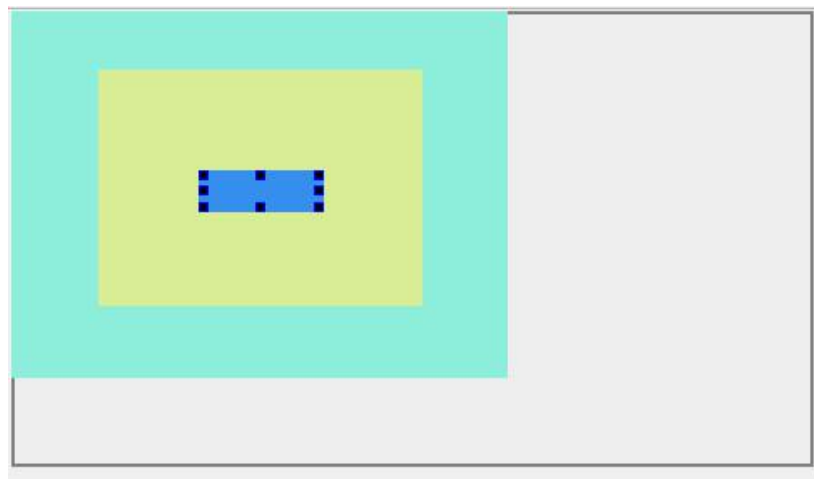
用户可根据具体需求设置其他属性，至此，一个电池控件就建好了。

2.6 按钮

选择一个布局，点击选择**按钮**并拖动到布局上，即可新建一个按钮。



布局上出现一个按钮矩形区域，如下图的深蓝色矩形区域：



接着用户可以设置该按钮属性内的背景色和背景图片，此处就按照默认分配设置。这种按钮控件一般用在触摸屏系统中，所以按钮按下与释放都会有不同的颜色或者图片，作为按钮动作的反应，此处我们以背景颜色的不同作为示范。

属性栏的 **css 属性_0** 为按钮未按下的显示状态属性，目前为蓝色。

ID号
NewFrame_112

CSS属性_0

对齐方式
ALIGN_CENTER

默认隐藏
false

标志
ELM_FLAG_NORMAL

位置坐标

X: 60

Y: 60

宽度: 75

高度: 25

背景颜色

背景图片

内边框线

左: 0

上: 0

右: 0

下: 0

边框

这里默认只有一个属性 0，我们需要新建一个属性 1，右键单击 **CSS 属性_0**，单击**复制添加**：

ID号
NewFrame_112

CSS属性_0 **复制添加**

对齐方式
ALIGN_CENTER

默认隐藏
false

标志

之后将会有个相同的 **CSS 属性_1** 被创建，我们设置其**背景颜色**为红色：

ID号
NewFrame_112

CSS属性_0 CSS属性_1

对齐方式
ALIGN_CENTER

默认隐藏
false

标志
ELM_FLAG_NORMAL

位置坐标
X: 60
Y: 60
宽度: 75
高度: 25

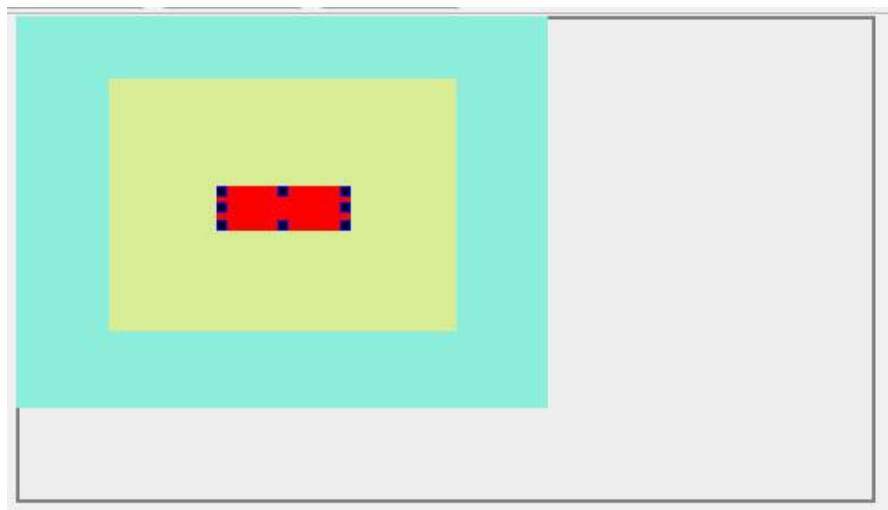
背景颜色

背景图片

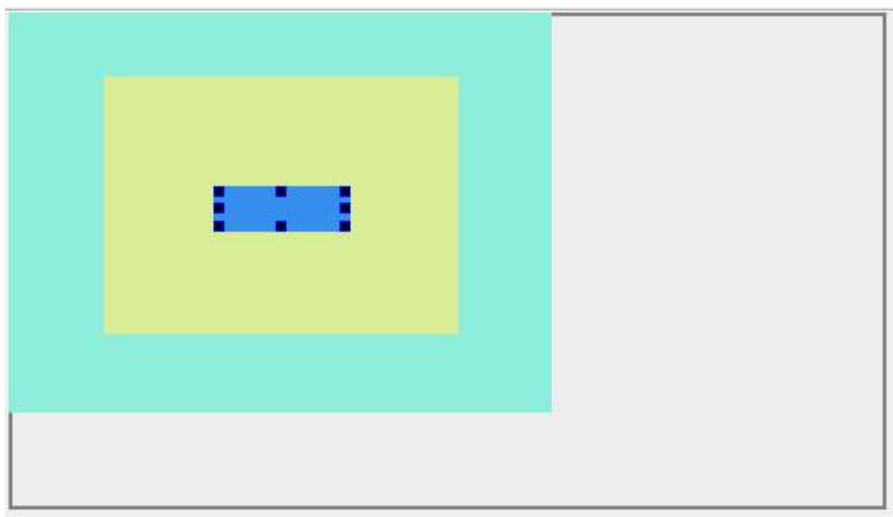
内边框线
左: 0
上: 0
右: 0
下: 0

边框

此时按钮按下状态就为 **CSS 属性_1**，效果如图：



按键不按下或被释放后的状态就为 **CSS 属性_0**，效果如图：



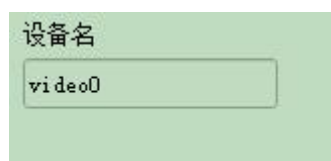
用户可根据需求设置其他属性。到此，一个按钮控件就建立好了。

2.7 Camera

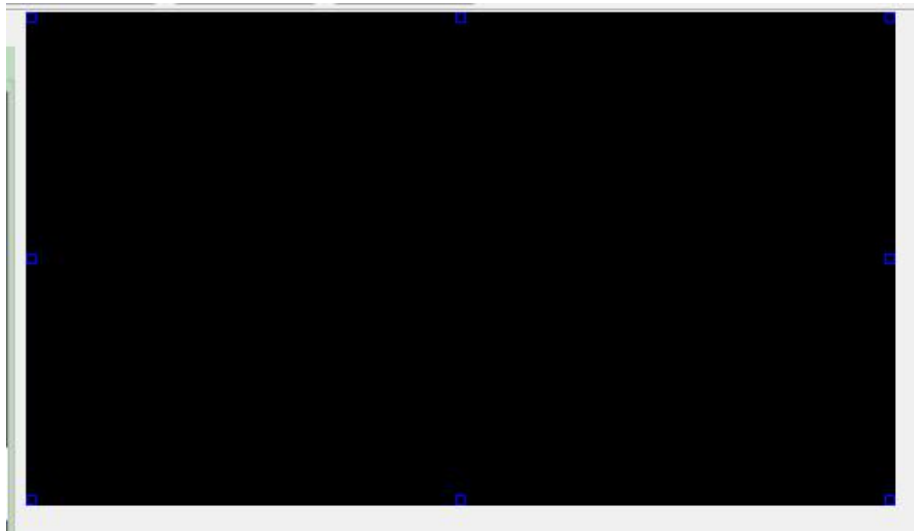
Camera 控件用于摄像头的图像显示，需配合 SDK 的摄像头驱动来进行使用。
在页面新建一个 YUV 图层，建立全屏大小的布局，在布局中新建一个 Camera 控件：



该控件有一个设备名，此处设为 **video0**，SDK 需要根据此设备名称来分配 Camera 设备：



调整好控件大小，效果如图：



至此，一个 Camera 控件就建立好了。

2.8 水平列表

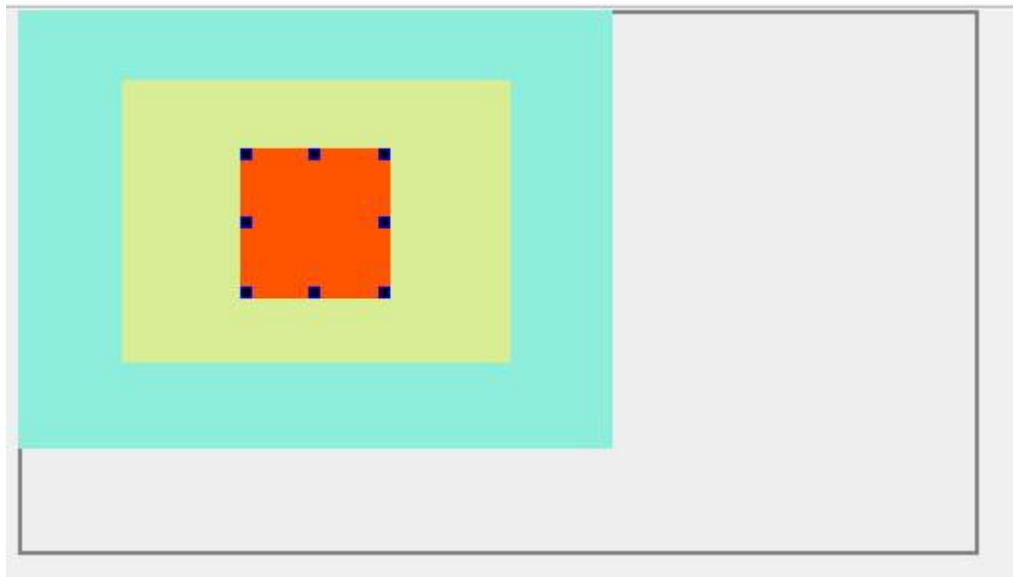
水平列表添加方式同垂直列表，请查看 2.13 节。

2.9 图片

选择一个布局，点击选择**图片**，新建一个图片控件：



布局上就会出现一个图片控件矩形区域：



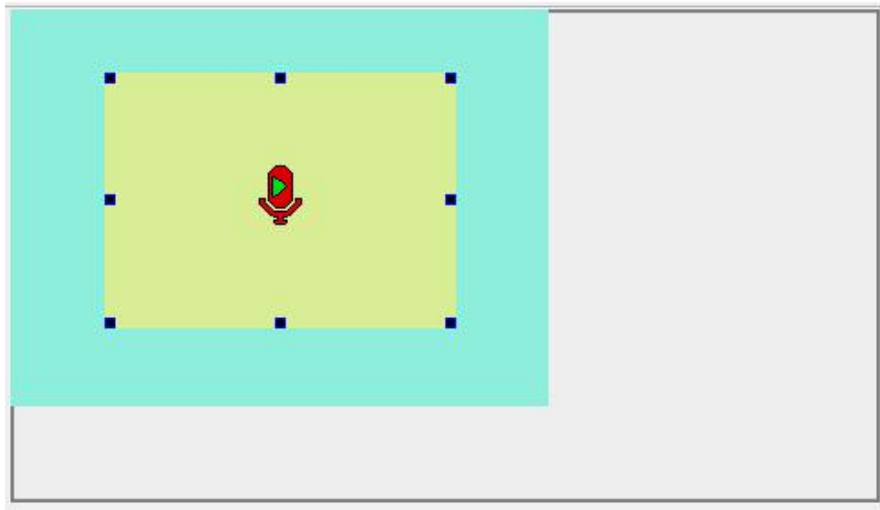
点击**图片列表**选择需要显示的图片，该图片控件可以在该图片列表中的图片进行相互切换，下拉列表为默认显示图片：



此处我们选择 3 张图片：



去掉背景颜色，调整好图片控件的位置和大小，效果如下图：



用户可根据需求设置其他属性。到此，一个图片控件就建立好了。

2.10 表格控件

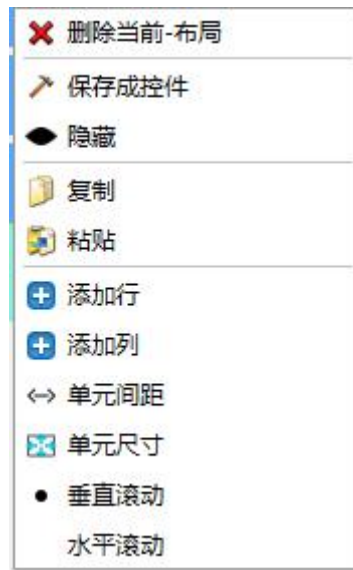
表格控件是作为表格形式的菜单来使用。新建一个布局，在布局中创建一个表格控件：



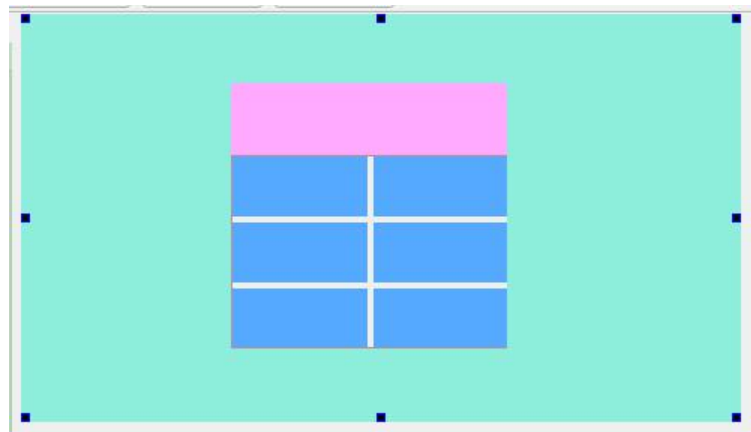
设置好行数和列数，设置好菜单项的长宽。

行:	<input type="text" value="2"/>
列:	<input type="text" value="2"/>
单元宽度:	<input type="text" value="30"/>
单元高度:	<input type="text" value="30"/>
<input type="button" value="确定"/> <input type="button" value="取消"/>	

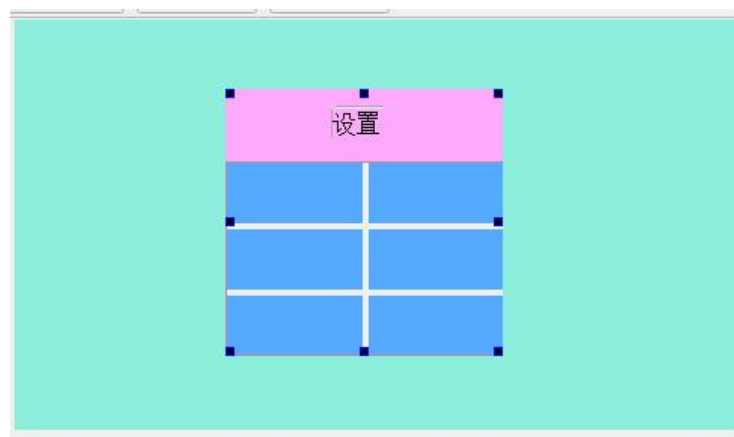
右键点击表格，可以添加行、添加列、设置单元间距、单元长宽等参数。



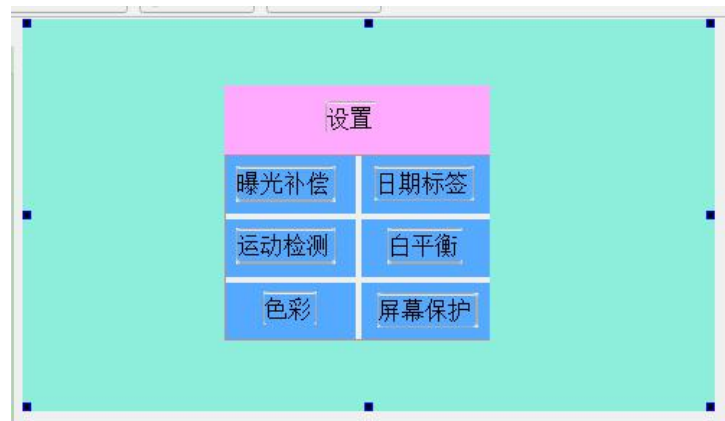
将布局设置为紫色，将表格各个单元设置为蓝色，设置好后效果如下图所示：



在紫色布局空隙中添加文字控件作为标题：



在菜单项添加文字控件作为菜单项文字：



对每个菜单项添加活动项属性，右键点击菜单项的 **CSS 属性_0**，选择复制添加，添加 **CSS 属性_1**。



将各个菜单项的 **CSS 属性_1** 设置为活动项颜色，此处设为黄色，效果如图：



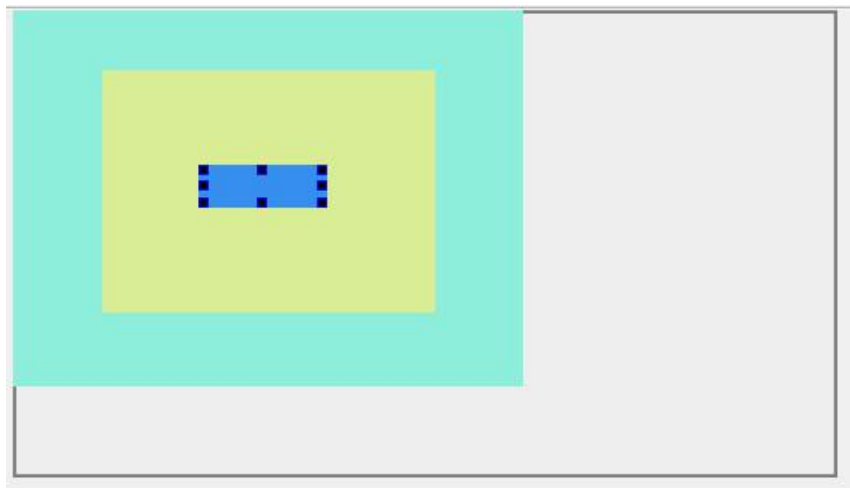
此时，菜单项的**默认状态**为蓝色，而被选中后的**活动项**变成黄色。至此，一个表格控件菜单就建立好了。

2.11 文字

选择一个布局，点击**文字**，新建一个文字控件：



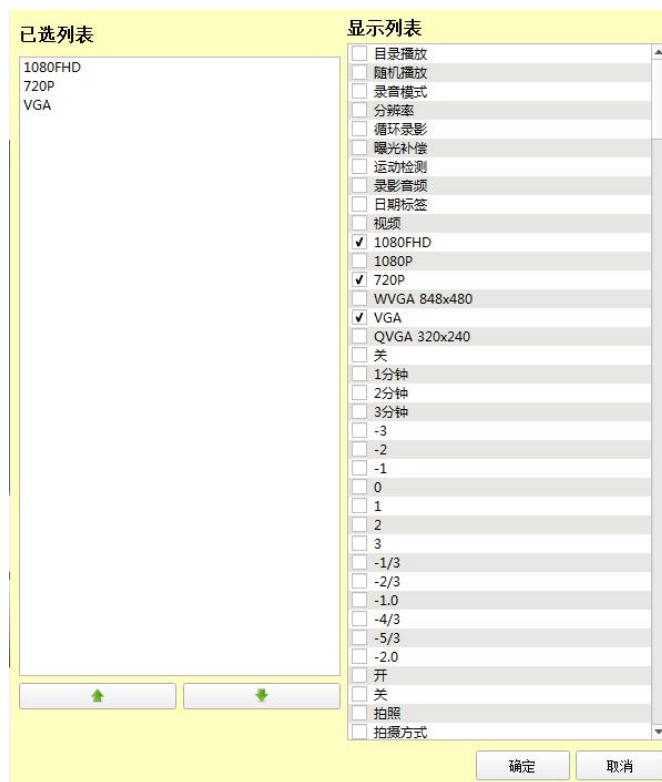
布局上就会出现一个文字控件矩形区域：



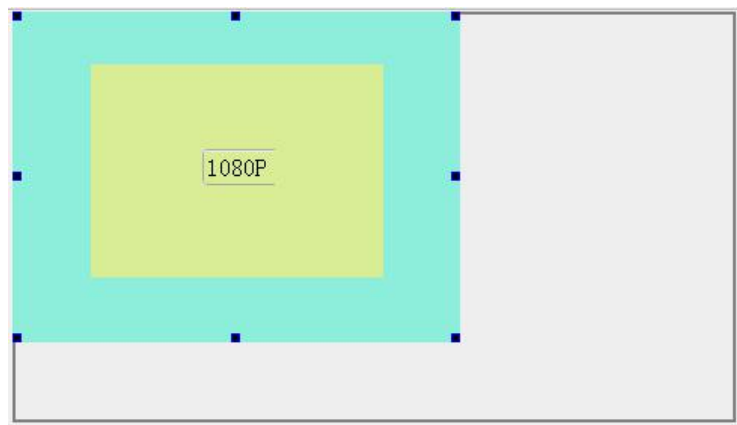
文字控件有两种形式，一种为多国语言，另一种为自定义文字，该属性设置项为**编码格式**，此处我们需建立一个多国语言文字，则编码格式编辑为 **strpic**。文字的显示颜色选择为黑色：



文字列表为该文字控件可以显示的多国语言，此处我们选择 3 个文字，在对话框右侧的文字左边勾选即可，对话框的左侧显示了已选文字的排列顺序，该顺序可以点击下方的绿色箭头进行调整，然后点击**确定**：



去掉背景色，调整位置和大小，最终结果如图：



如果需要实时改变文字，则需要选择文字控件的另一种形式，将属性的**编码格式**编辑为**ascii**，这时用户不需要添加文字列表，因为这种文字是通过程序设置实现的。如图：



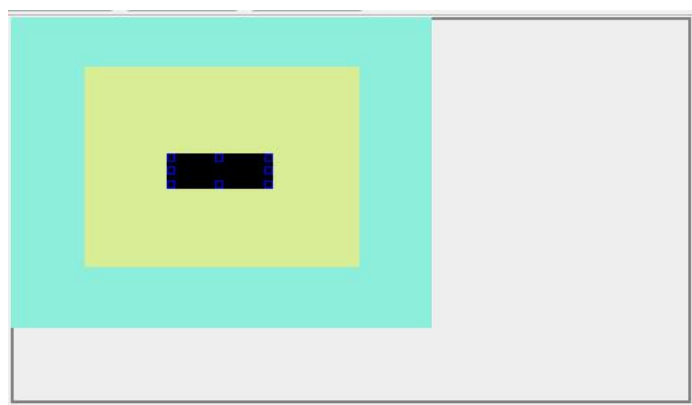
用户可根据需求设置其他属性。到此，一个文字控件就建立好了。

2.12 时间

选择一个布局，点击**时间**，新建一个时间控件：



在布局上就会出现一个时间控件矩形区域：



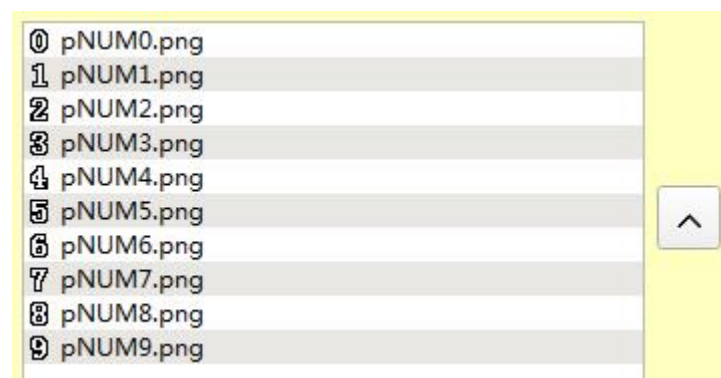
接下来需要设置时间的显示格式，格式的关键字为 **YMDhms**，分别表示年、月、日、时、分、秒。其它文字不做解析，直接通过 ASCII 或者图片显示。



如果不设置图片列表，则所有时间文字都通过 ASCII 字模来显示。此处我们打算用数字图片来显示时间，我们可以把格式中的 Y/M/D 删除。



设置数字图片列表，图片需要按照从数字 0-9 的顺序选择图片。



设置分隔符图片列表，分隔符图片列表需要根据顺序设置各个分隔符的图片。这里有两个“:”符号，则需要选择两个“:”图片，用户可以复制原来的图片作为第二个“:”，图片选择如图：



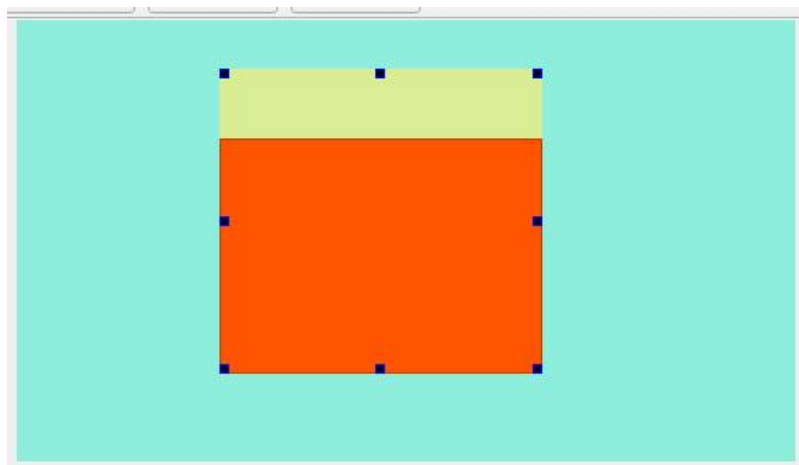
用户可根据需求设置其他属性。到此，一个时间控件就建立好了。

2.13 垂直列表

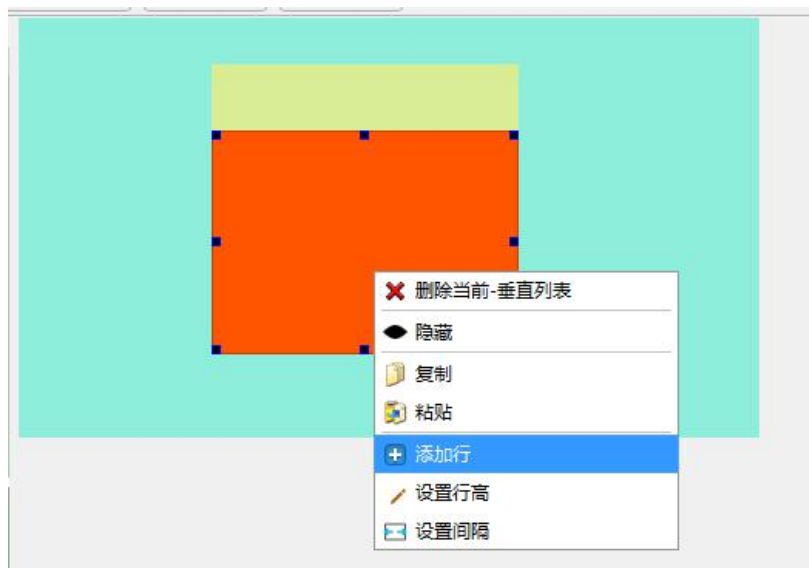
垂直列表是用在垂直菜单中的，一个菜单最好占用一个布局。选择一个布局，点击**垂直列表**，新建一个垂直列表控件：



调整好布局和垂直列表控件的大小，预留布局上方空档作为菜单的标题项，如图黄色部分为空档，红色部分为垂直列表区域：



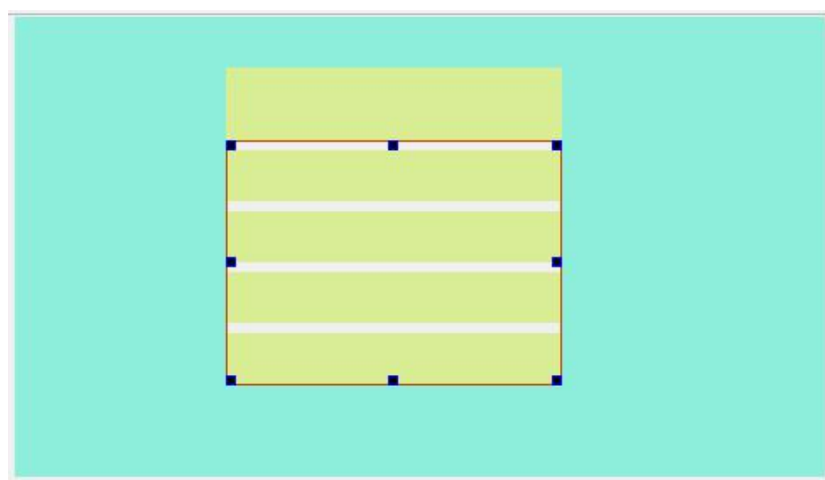
接着要建立菜单项，右键点击垂直列表区域（红色区域），选择**添加行**：



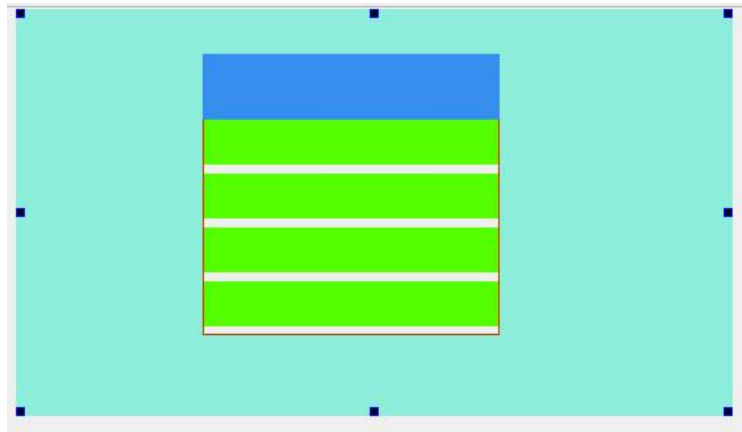
弹出对话框，设置添加的行数，假如该菜单有 10 项，则填写 10：



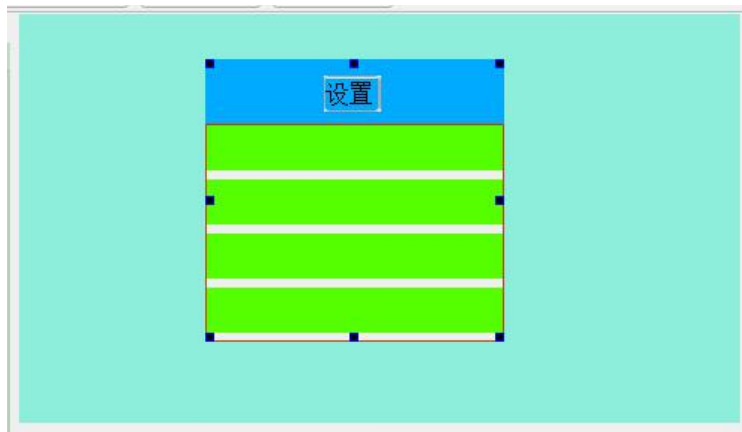
添加好后，工具会自动生成等距等长等宽的行，如果垂直列表控件不够显示所有行，可以通过鼠标滚轮来进行行切换。下图为已经添加好的行，一面只够显示 4 行：



接下来我们可以改变行的背景色为绿色，将布局改为蓝色，如图：



添加标题，需要在蓝色的布局部分添加一个**文本控件**，编码类型为 **strpic**，选择其图片列表为**设置**字体颜色为黑色，调整好控件大小并去除背景色，效果如图：



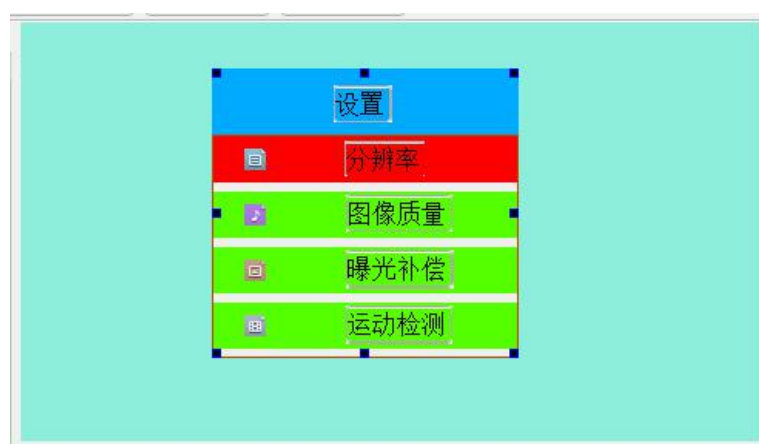
然后需要对菜单项进行文字和图片布局，逐行选择并进行菜单项的图片控件与文字控件添加，最后效果如图：



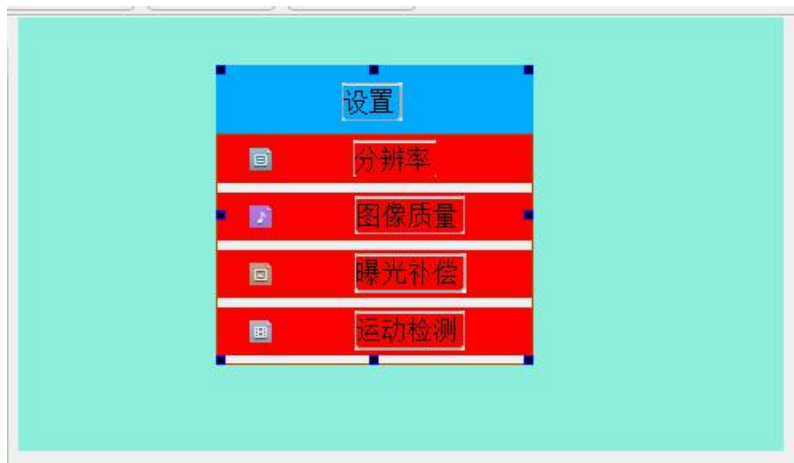
到此，菜单项的所有行的目前非活动项属性都设置完毕，也就是当前的 **CSS 属性_0**。接下来要为所有菜单项都添加一个 **CSS 属性_1** 作为活动项属性。选择行，右键点击 **CSS 属性_0**，选择**复制添加**，生成一个 **CSS 属性_1**：



我们将作为活动项状态的 **CSS 属性_1** 的背景色改为红色，此时第一行的活动项动作及表现为背景色变成红色，效果如图：



接着将其它行的活动项状态都设置为红色，亦即将 **CSS 属性_1** 背景色改为红色，最终效果如图：



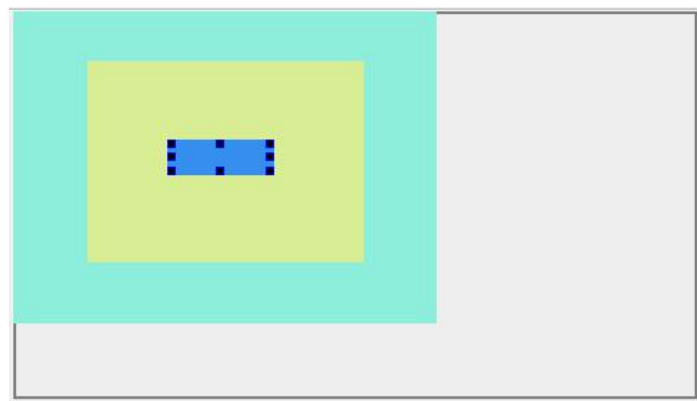
此时，非活动项为绿色，被选中的活动项将变成红色。一个垂直列表菜单就做好了。

2.14 动画

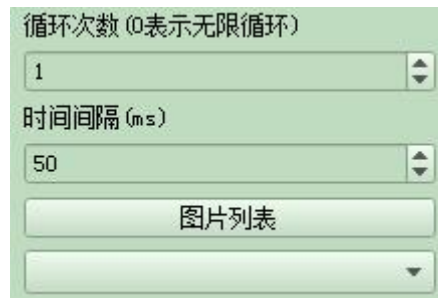
选择一个布局，点击**动画**，新建一个动画控件：



布局上会出现新的动画控件矩形区域：

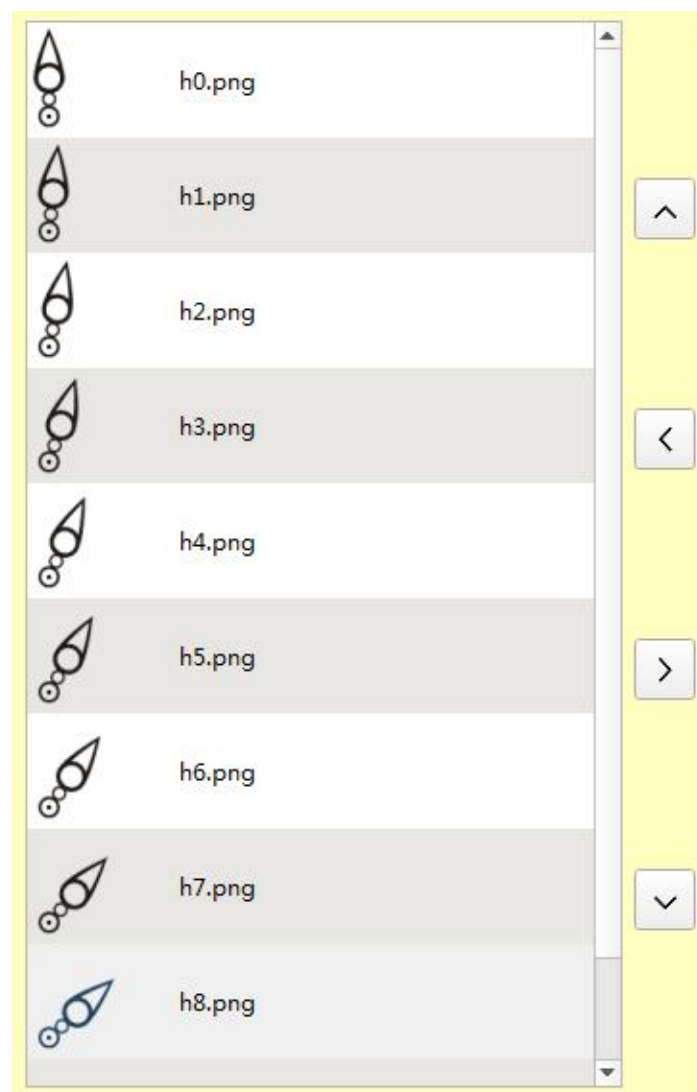


动画需要设置循环播放次数、播放帧率（时间间隔）和图片列表，**循环次数**为动画每次触发播放后自动播放的循环次数，0 表示无限循环，**时间间隔**则为每张图片播放的时间间隔，也就是动画的帧率，图片列表则为该动画的播放列表：

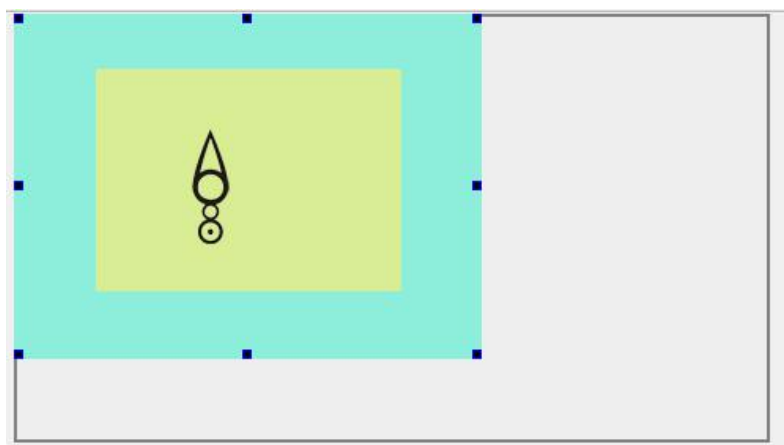


A green-themed dialog box for animation settings. It contains three main sections: '循环次数 (0表示无限循环)' with a spinner set to 1, '时间间隔 (ms)' with a spinner set to 50, and '图片列表' with a button and an empty dropdown menu below it.

点击图片列表选择动画播放的图片：



去掉背景色，调整好空间大小和位置，效果如图：



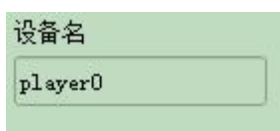
用户可根据需求设置其他属性。到此，一个动画控件就建立好了。

2.15 播放器

播放器用于用户播放图片或者 AVI 视频，该功能需要 SDK 的视频播放接口搭配使用。选择一个布局，点击**播放器**，新建一个播放器控件：



播放器的识别方式为**设备名**，这里设置为“**player0**”，此参数用于 UI 库底层进行不同播放器的识别：



去除背景色，调整控件大小和位置：

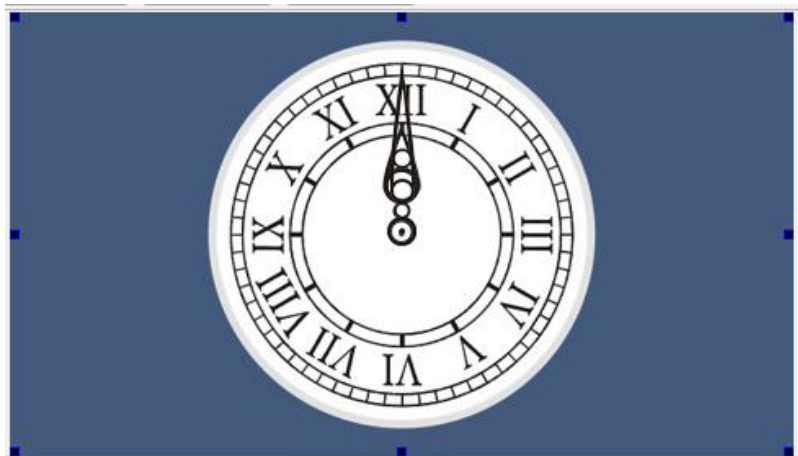


用户可根据需求设置其他属性。到此，一个播放器控件就建立好了。

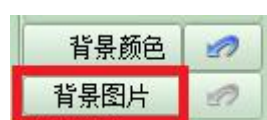
2.16 watch 钟表

钟表控件比较特别，其基本模型包含一个背景，一个时针，一个分针，一个秒针。此控件需要进行自定义，而不是在控件列表中直接选择。

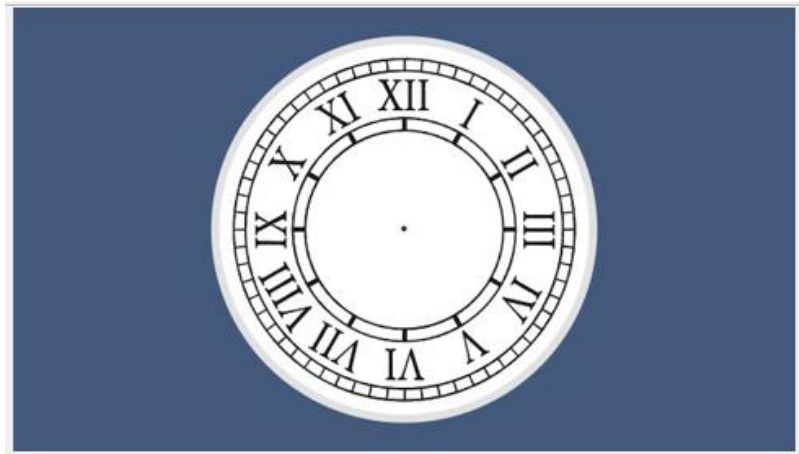
最终效果图为：



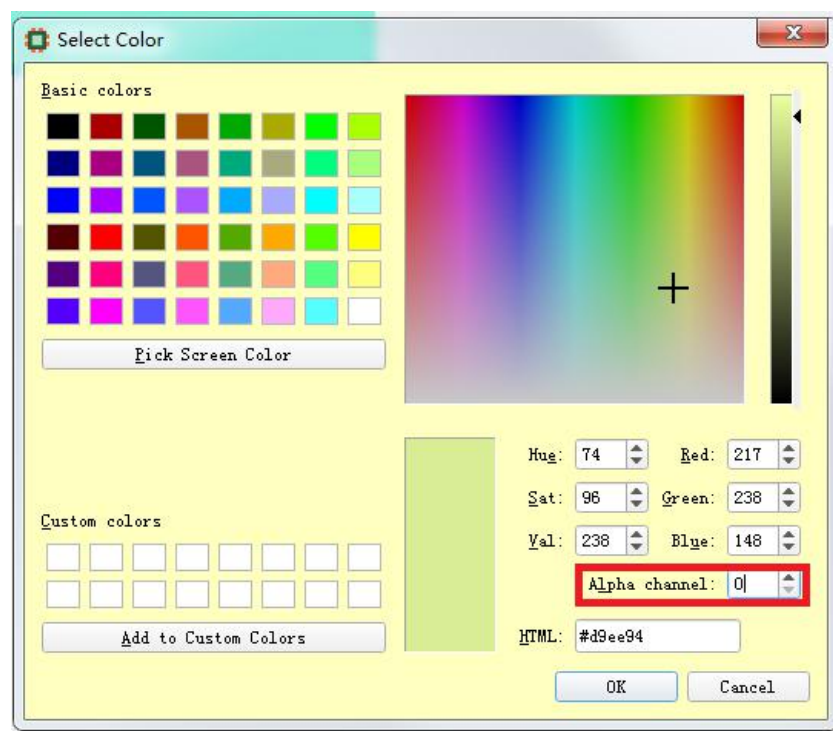
首先需要自定义一个 watch 控件。新建一个页面，创建一个 YUV 图层（如果背景为 JPG 图片），大小为全屏大小，点击**背景图片**选择一个钟表背景。



图层效果为:



接着,新建一个全屏布局,此布局将会作为自定义 watch 控件,点击背景颜色的蓝色箭头去掉背景色,然后单击背景颜色,将 **Alpha channel** 设置为 0,也就是该布局为背景全透明布局:



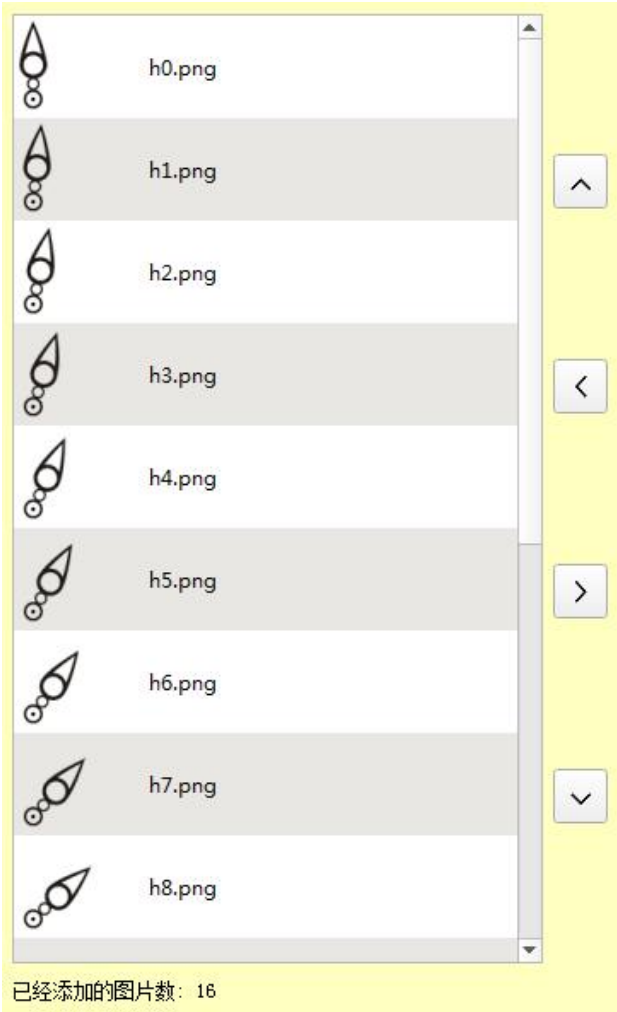
此时页面树状图为下图所示,图层为背景图,布局为透明布局:

结点	属性
▼ 图层_112	NewLayer
▼ 布局_113	NewLayout

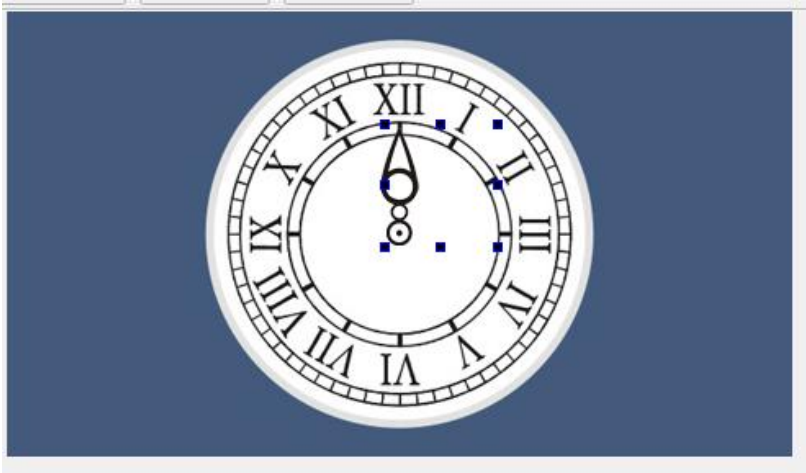
然后需要添加时针、分针和秒针，顺序为时针、分针、秒针，如果不需要秒针，则可以不建立秒针。首先建立时针，在布局上新建**图片**控件：



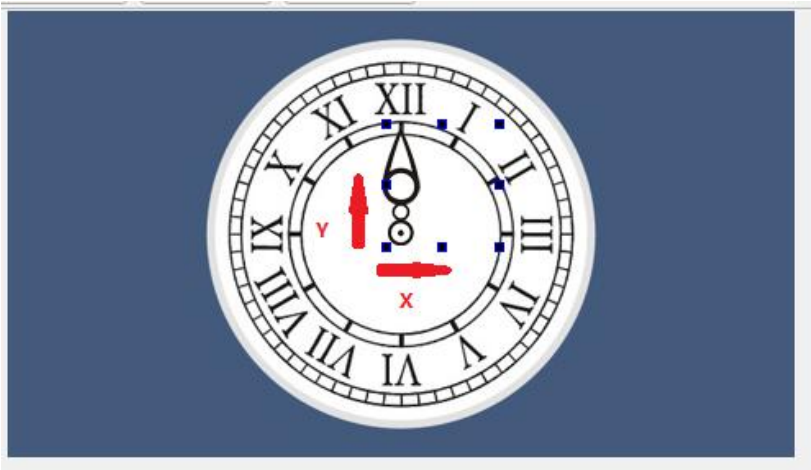
设置时针的第一象限 16 个图片，顺序为顺时针，列表如图：



去除背景色，根据时针图片的真实长宽来设置图片长宽，然后将时针的中心对准背景图的中点：



接下来要设置时针图片的旋转对称中心坐标，坐标的 XY 原坐标为图片的左下角：



此处旋转中心就是指针的黑点位置，设置为(10, 9)，注意一定要设置准确，否则其它象限的指针将会显示不准确。

默认高亮

0

旋转中心点: X

10

旋转中心点: Y

9

图片列表

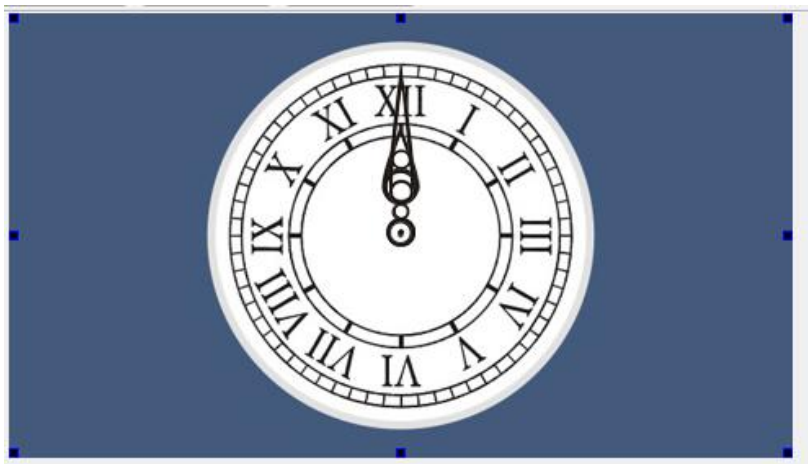
h0.png

高亮图片列表

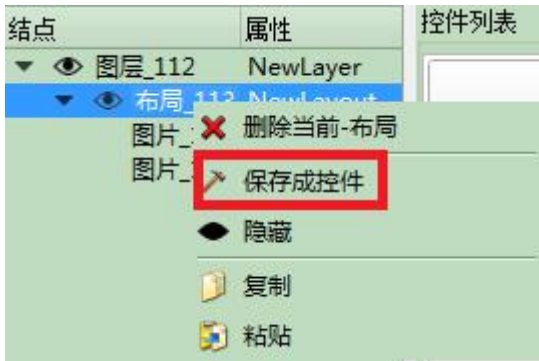
到此，时针设置完毕。分针的设置也跟时针一样，最后的布局树状图为下图所示，布局下的两个图片分别为时针和分针：

结点	属性
▼ 图层_112	NewLayer
▼ 布局_113	NewLayout
图片_113	NewFrame
图片_114	NewFrame

自定义控件效果图如图所示：



自定义控件布局好后，需要保存该控件，在树状图的布局上点击右键，选择**保存成控件**：



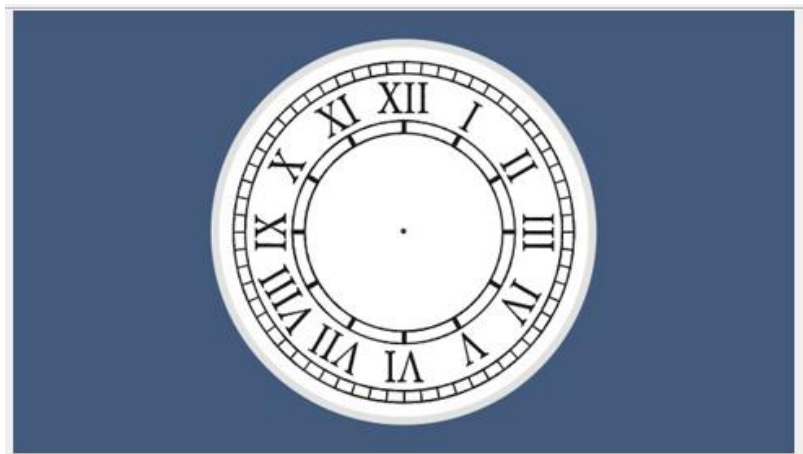
弹出对话框，设置控件的名称，一定要设置为 **watch**：



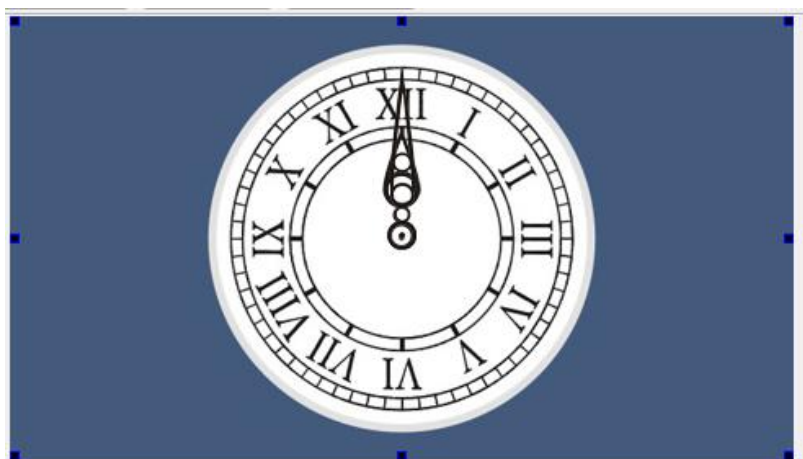
然后关闭该页面，重新打开工程，即可看到控件列表上多出来一个 **watch** 控件：



接下来就可以使用该自定义控件了。首先新建一个用来放置钟表的页面，新建一个 YUV 图层，背景为钟表背景：



而后再创建一个 RGB256 图层，创建一个布局，去掉背景色，调整大小为全屏。在布局上新建刚刚的自定义控件 **watch**，调整好控件的位置：



最终该页面的树状图结构如下图，图层 6 为 YUV 背景，图层 7 为 RGB256，布局 8 承载着 watch9 控件：

结点	属性
👁 图层_6	NewLayer
▼ 👁 图层_7	NewLayer
▼ 👁 布局_8	NewLayout
▼ 👁 watch_9	NewLayout
图片_10	NewFrame
图片_11	NewFrame

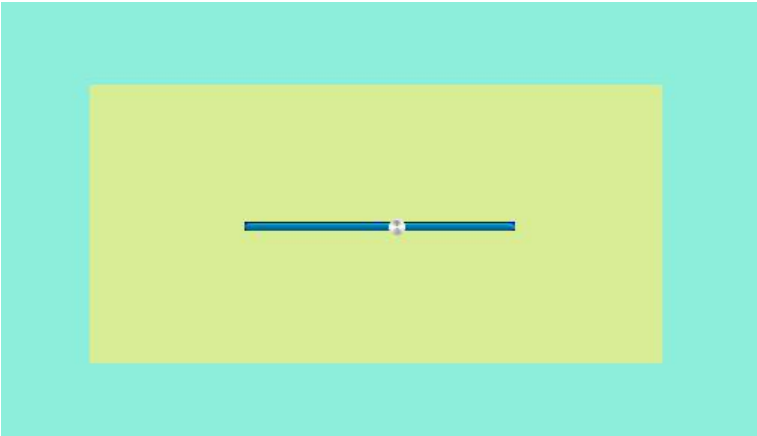
到此，一个 watch 控件就建立好了。

2.17 slider 滑块

点击并拖动 slider 控件到布局上，新建一个 slider 滑块控件。



如下图，一个 slider 滑块就创建出来了，滑块一般用于显示播放进度条。



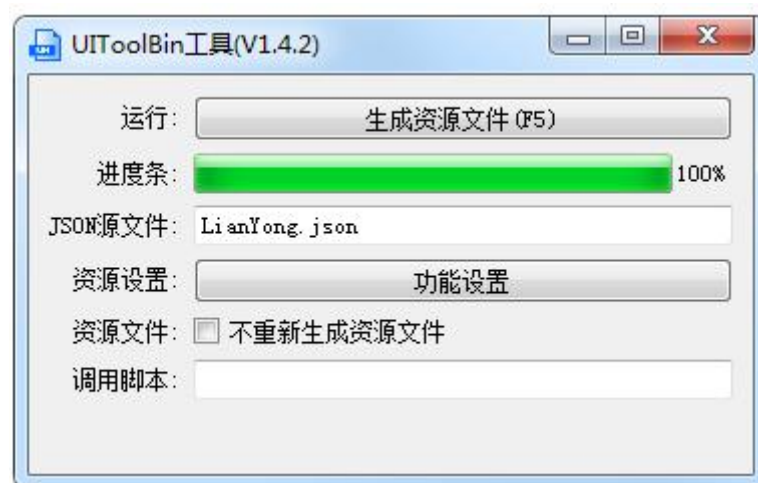
Slider 滑块控件的结点结构如下图所示，right_pic 为底部黑色条，left_pic 为顶部进度蓝色条，slider_pic 为圆点滑块。



用户可以根据需求，改变这三个图片控件的属性，至此，一个 slider 控件就建好了。

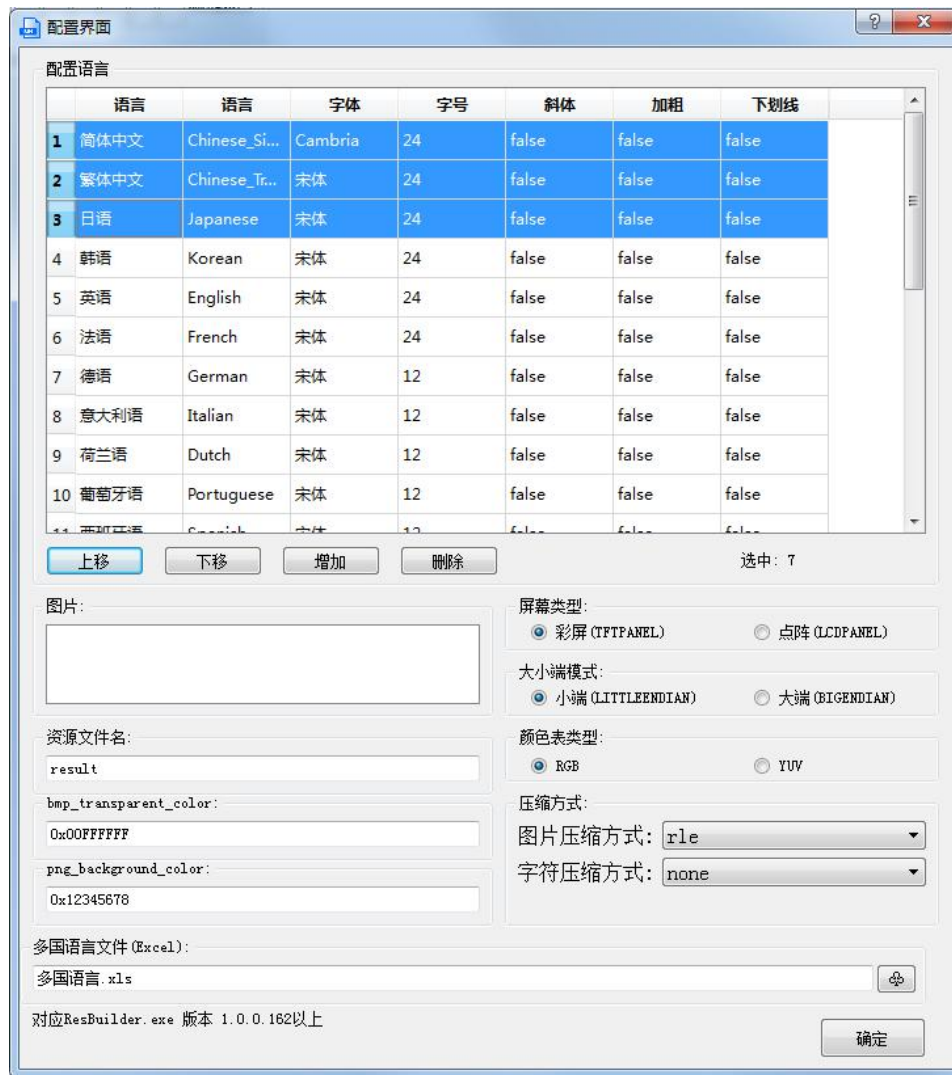
3 UI 资源生成工具使用说明

使用 UI 布局工具布局好 UI 并且保存好工程后，需要使用资源工具生成 UI 资源文件。打开 UI 资源工具：



在弹出的对话框中，填入需要生成资源的工程文件，比如此处为 LianYong.json。不生成资源文件选项一般不勾选，如果勾选则只会生成 project.bin 工程布局文件。

点击配置多国语言可以对多种语言进行参数设置，单击某种语言变成蓝色即为选择生成该种语言，至少选择一种语言！双击某种语言即可设置语言文字属性。



配置好参数后，点击**生成资源文件（F5）**，等待资源生成完成显示**100%完成**。工具会在同级目录生成**6**个文件：

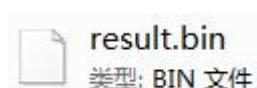
- 控件 ID 号命名头文件 **ename.h**



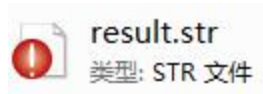
- 工程布局文件 **project.bin**



- 图片资源文件 **result.bin**



- 多国语言资源文件 **result.str**



- 多国语言 ID 定义文件 **result.h**



- 资源版本号定义文件 **res_ver.h**



用户需要将生成好的文件放到 SDK 工程中使用，可以参考 **copy.bat** 批处理进行文件复制。每次只需运行该批处理就可以完成繁琐的文件复制与重命名操作。用户根据自己的工程目录来编辑批处理命令：

```
copy.bat - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
copy .\project.bin .\res\LY.sty
copy .\result.bin .\res\menu.res
copy .\result.str .\res\str.res
copy .\ename.h ..\verify\car_app\code\disp_ctrl\ename.h
copy .\result.h ..\verify\car_app\code\disp_ctrl\res.h
copy .\res_ver.h ..\verify\car_app\code\disp_ctrl\res_ver.h
```

最后，UI 资源文件操作流程已经完成，用户可以编译 SDK 并且升级程序到开发板运行。