## 概述

### 插件介绍

插件基于核心：

◆Drill\_CoreOfBallistics 系统 - 弹道核心

◆Drill\_CoreOfGaugeMeter 系统 - 参数条核心

◆Drill\_CoreOfGaugeNumber 系统 - 参数数字核心

插件本体：

◆Drill\_GaugeForBoss UI – 高级BOSS生命固定框

该插件的扩展：

◆Drill\_EnemyTextColor UI - 敌人文本颜色

◆Drill\_MiniPlateForState 鼠标 - 状态和buff说明窗口

◆Drill\_X\_GaugeBossFilter UI - 高级BOSS框的滤镜效果[扩展]

该插件的配置项非常多，需要花较长的时间耐心学习。

**这个插件的功能是实现 指定敌人的生命、魔法、怒气属性 可视化而存在的，只提供信息显示，并不影响敌人的真实属性值。**

### 相似文档

以下为设计方法相似的插件文档，可以结合理解：

|  |  |
| --- | --- |
| 插件名称 | 文档名称 |
| Drill\_EnemySimpleHud  战斗UI - 简单生命框 | 5.战斗UI > 关于战斗简单生命框.docx |
| Drill\_GaugeForBoss  UI – 高级变量固定框 | 13.UI > 关于高级BOSS生命固定框.docx |
| Drill\_GaugeForVariable  UI – 高级变量固定框 | 13.UI > 关于高级变量固定框.docx |
| Drill\_GaugeOfBufferTimeBar  UI - 缓冲时间条  Drill\_GaugeOfBufferTimeNum  UI - 缓冲时间数字 | 13.UI > 关于缓冲时间条与数字.docx |

### 插件关系

高级BOSS生命固定框的关系如下图。实线表示必须插件，虚线表示可断开的扩展插件。



## 固定框样式

### 结构

固定框的结构如下图所示。

**简单来说，主要较复杂的配置为 外框+6个参数数字+3个参数条。参数条和参数数字要去相应的核心先设计好，再贴到框样式中**。



固定框样式需要用过BOSS设置来绑定敌人。绑定敌人后，该敌人的生命、魔法、怒气参数会提供给相应的 参数条和参数数字，参数条和参数数字会根据参数的值，进行自我变化。

以 固定框样式-粗风格 为例，图解结构如下：



其中：

1）头像不在固定框样式中配置。

因为多个不同敌人可以使用同一个固定框样式，而头像和敌人是一对一的，所以头像的配置放在boss设置中，可见后面章节。

2）生命段数和生命数字是两套不同的参数数字样式。

生命数字和生命段数的值不一样。而且生命段数有前缀”x”号，生命数字可以变形为 100/100的格式，所以二者是两种不同的参数数字。需要分别配置。

### 参数条

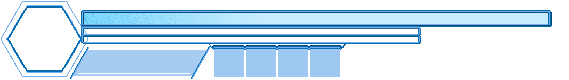
**1) 参数条样式**

下图为背景框与参数条的组合关系。

配置参数条坐标时，注意要使得参数条与背景框完美嵌入。

**参数条的详细样式配置，可以去看看”关于参数条.docx”，这里不赘述**。







图中有背景框和3个参数条：（前景框为空图片，参数数字都未显示）

生命条：段数8 + 流动 + 凹槽条 + 弹出条

魔法条：段数1 + 流动

怒气条：段数1 + 流动

**2) 加满动画**

加满动画是指，框出现时，生命条加满过程的动画。

该动画在许多动作类游戏中的boss都有展现。



### 参数数字

**1) 参数数字样式**

下图为背景框与参数数字的组合关系。

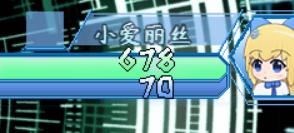
**参数数字的详细样式配置，可以去看看”关于参数数字.docx”，这里不赘述**。



图中有背景框，1个参数条，2个参数数字，其余的都未显示。

生命数字：右对齐

生命段数：右对齐 + ”x”乘号前缀



图中有背景框，2个参数条，2个参数数字，其余的都未显示。

生命数字：右对齐

魔法数字：右对齐

**2) 生命数字**

需要说明的是，最大生命值 = 生命数字的额定值 。（额定值是参数数字中的概念）

这里固定最大生命值为额定值。即显示”678/720”的格式时，720是最大生命值。

**3）生命段数**

生命段数 = 当前生命值 / 段上限

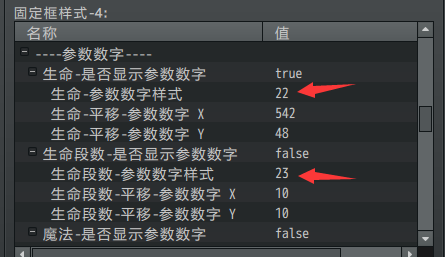
段数的数值，将会作为参数数字显示出来。你可以加”x”号前缀，或者自定义的特殊符号。

**4) 插件指令临时显示数字**

你可以使用插件指令设置参数数字在战斗进行时出现。



但是前提是，你必须配置样式，如果样式为0，则什么都看不见。

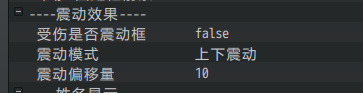


### 其他属性

**1) 震动效果**

Boss收到伤害时，会进行上下震动或作用震动。

注意，此功能不建议随意加，有时候加了反而影响看数字。



**2) 姓名显示**

显示Boss的姓名。

姓名能够支持使用渐变颜色，但是需要Drill\_EnemyTextColor敌人文本颜色插件的支持。



**3) 状态显示**

状态与数值的配置相似，可以设置并排，并 左对齐、右对齐、上对齐、下对齐 。

状态只能直线排列，不能折行。

## BOSS设置

### 绑定关联

Boss设置的结构如下图所示。

Boss设置的配置比较杂，因为敌人有很多个，多个敌人可以配置同一个固定框样式，所以敌人数据相关参数属性就都被单独拿出来进行控制了。

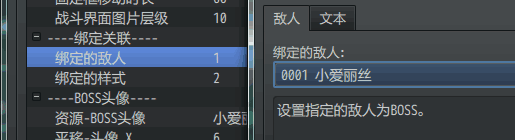


Boss设置 -> 固定框样式 是一对一 关系。

固定框样式 -> Boss设置 是一对多 关系。

也就是说，多个小爱丽丝可以用同一个样式，每个小爱丽丝boss只能用一个样式。

（你可以通过插件指令，在游戏中另外修改小爱丽丝boss的样式，不过该功能不常用。）



**注意，你需要在BOSS设置中分配固定框的平移位置，使得战场上出现多个boss生命框时，互不遮挡。**

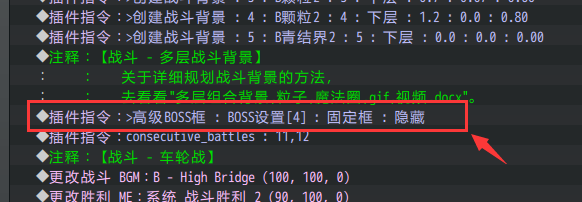
### 时机

**1) 中途出现**

BOSS处于"中途出现"的状态时，BOSS框也会显示。



所以你需要在战斗前隐藏框，再通过插件指令显示，达到"中途出现"框。



**2) BOSS变身**

BOSS变身后，相当于换了另一个敌人。

原来的BOSS框会消失，对应新的BOSS框会显现。

**3) 车轮战**

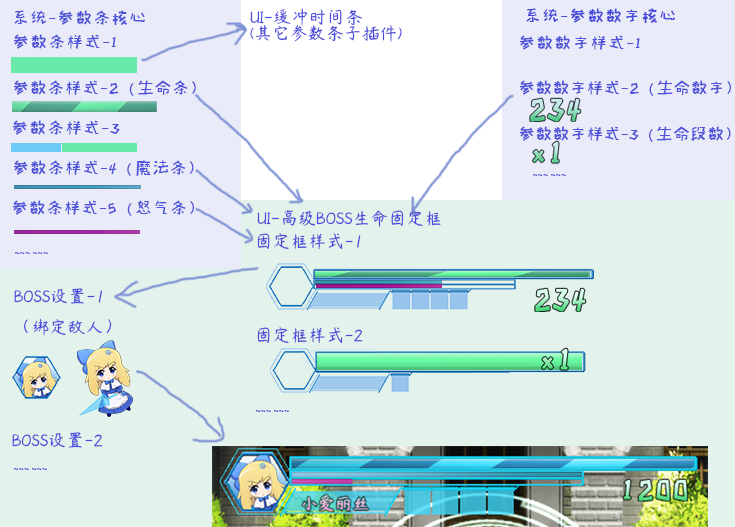
车轮战中，每一波战斗相当于一次完整的战斗。BOSS框会重复刷新。

由于mog车轮战和战斗镜头插件交互存在bug，BOSS框在修正坐标时可能会出界，看不见BOSS框，目前此问题待解决。

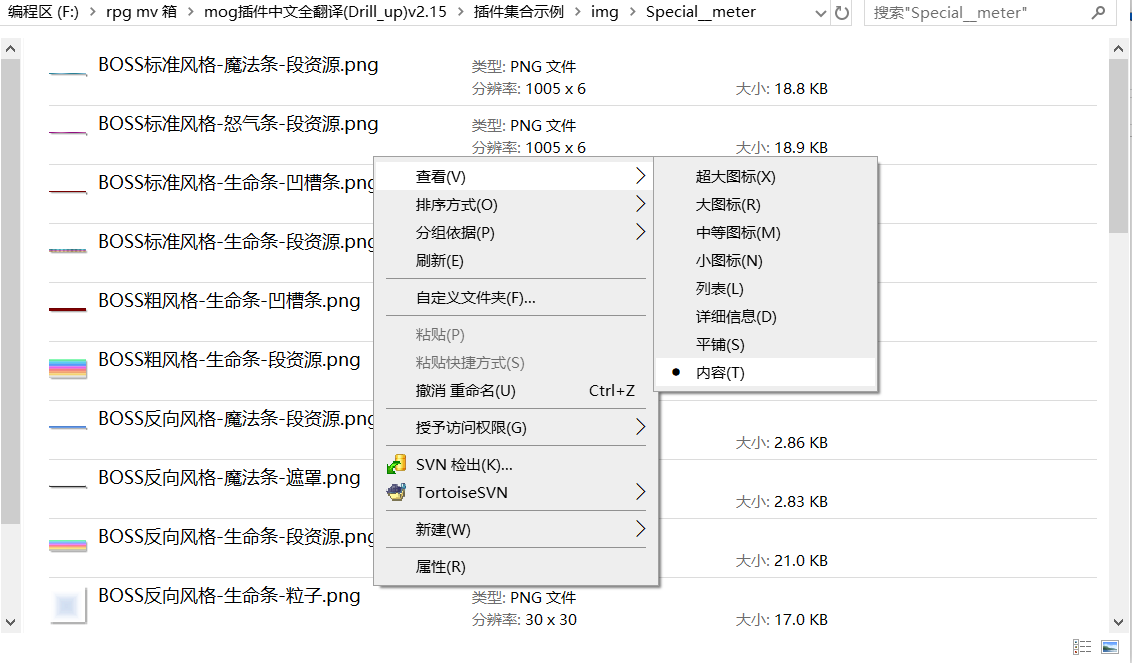
## 设计

### 配置流程

**由于boss生命框、参数条核心、参数数字核心中都有不同的关联配置，这里建议使用”风格”来区分不同的生命框样式**。防止各个配置纠缠在一起，难以区分。

****

**在文件夹中查看时，建议将内容平铺，方便看图片名称和分辨率**。



### 示例风格

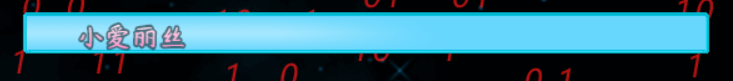
以下为示例中提供的风格：

**标准风格**：

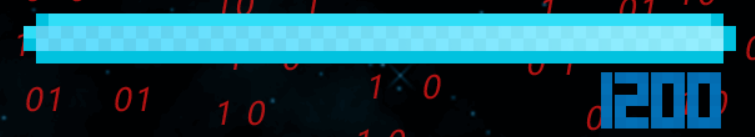
（去看看在 boss生命固定框插件、参数条核心插件、参数数字插件 中找到”标准风格”的字眼，他们相互关联。）



**简约风格**：



**像素风格**：

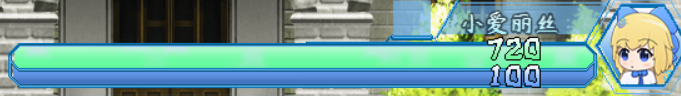


**粗风格**：



**反向风格**：

（由于该风格的生命条和魔法条都旋转了180度，所以称反向风格。）



### 参考对象

boss生命框是玩家比较在意的地方，你需要多花心思设计出好看的生命框。以下是一些游戏的生命框分解参考：

《鬼泣4》的boss生命框

背景 + 前景 + 加满动画 + 生命条（1段 + 不流动 + 凹槽条）

K:\rmmv\`D~7$5_5H4COM18DWZHWLHR.png

《鬼泣5》的boss生命框

背景 + 生命条（1段 + 不流动 + 凹槽条 + 粒子效果 + 游标）



《猎天使魔女》的boss生命框

背景 + 加满动画 + 生命条（4段 + 不流动 + 凹槽条）



DNF的boss生命框

背景 + 头像 + 名称 + 生命段数 + 生命条（8段 + 不流动 + 白色块弹出条）



《苍翼默示录》的boss生命框

背景 + 头像 + 生命条（1段 + 流动 + 凹槽条 + 粒子效果） + 魔法条（1段）



《血缘诅咒》的boss生命框

背景 + 名称 + 生命数字 + 生命条（1段 + 不流动 + 游标）



《死亡细胞》的boss生命框

背景 + 生命条（1段 + 不流动 + 白色弹出条 + 游标）



（不过说起来，这些完美的生命框，无论怎么表现，在大作游戏里面都是锦上添花，在rpg这种小游戏里面，实在形成不了亮点。）

## 从零开始设计

在开始设计前，你可以先了解一下 [BOSS框的配置流程](#BOSS框的配置流程) 。

|  |
| --- |
| **关于审美：**  经过群友提醒，作者我才注意到，设计出来的框架可能【不好看】。  这是因为作者我编写文档时，是以一个开荒者的角度思考问题的。开荒者关注的侧重点是一个满足功能的框架和一个完整的教程说明，美感是次要因素，所以自然显得难看。  作为读者的你们，更多要关注的是游戏的细节和UI的设计。所以，千万不要把我的拙劣的美术当成UI的能力上限。希望你们能在此教程基础上设计出更漂亮的UI。 |

### 1.设置一个目标

首先要想明白自己想创建一个什么样的boss生命框，这一点非常重要。Boss框的功能并不是堆得越多越好，重点还是要贴合你游戏的画风。

经过百度寻找灵感，根据想法建立了一个初稿：



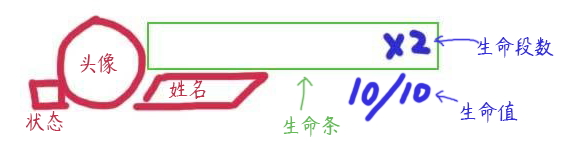
感觉单纯的一个长条太简陋了，改进一下结构：



### 2.结构分解，规划区域

你需要将 参数条、参数数字、外框 的结构分离。因为三者都在不同的插件中配置。

经过分析，可能会需要 1个背景外框、1个参数条，2个参数数字，然后就是姓名、状态、头像规划的区域。



这里魔法和怒气不考虑。

### 3.起草资源

**1) 生命条资源 + 框架**

由于生命条是重点对象，所以这里先从生命条着手。

设计时想让条显得更加粗一些，有打击感。于是将高度设为30像素。再拉个渐变。



直接设计的长方形，不需要考虑遮罩。

这里也不考虑流动效果。资源长度就是实际显示长度。

条的大小确定后，再根据条的大小设计背景外框和凹槽条。

凹槽条和生命条大小一样，加一点渐变，使其看起来像是凹进去的。



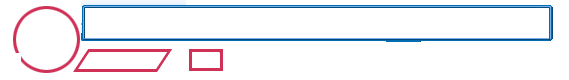
框先设为外面包裹的一层简单框架。



框架画好后，按照原来的布局想法，先把框放上：



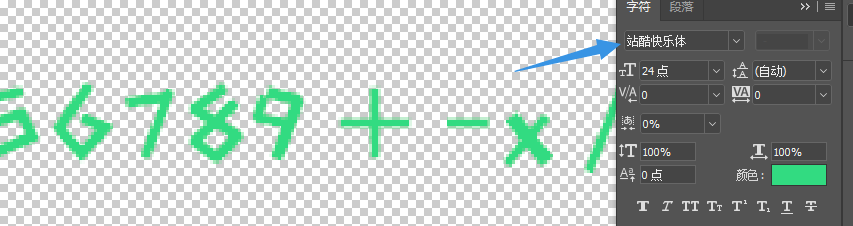
这里发现状态的位置反而挤出去了一点，有些不好看，于是放里面。



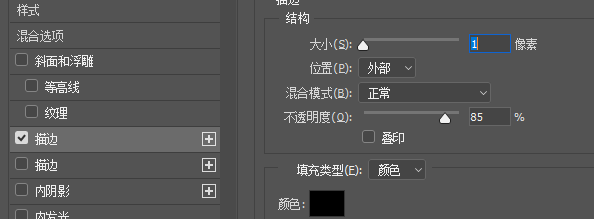
**2) 生命数字资源**

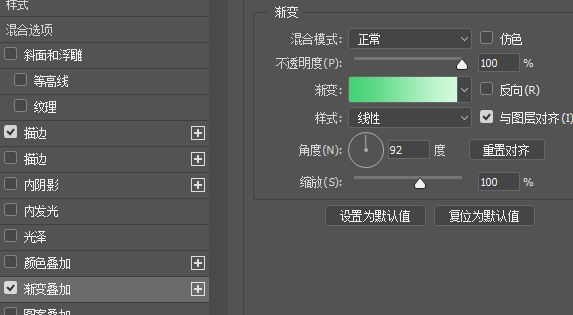
生命数字可以在ps中用铅笔画字符像素点，也可以使用一种字体，然后包裹一些特效。

这里使用站酷快乐体进行设计。



使用一层描边+白色渐变叠加就可以了，





然后栅格化文字，进行排布，分成等距的14份。



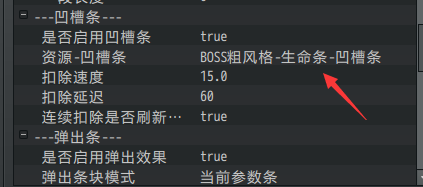


### 4.配置样式

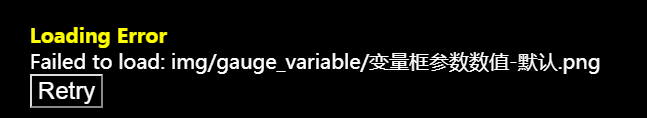
**1) 生命条配置**

下面把打开 参数条插件 配置一个参数条，由于不考虑遮罩，并且不流动，而且当前只画了绿色1段的结构，所以配置如下：





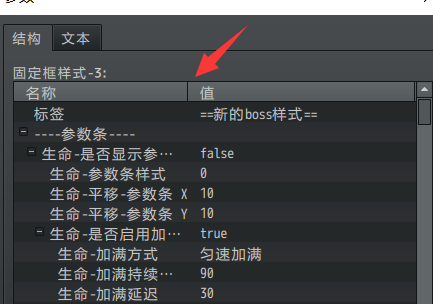
注意在测试前，有可能需要把部分默认资源设为空，如果默认的资源找不到，会报文件错误：



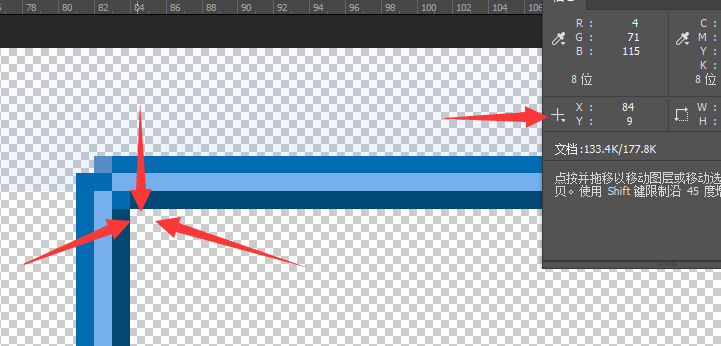
**2) 外框配置**

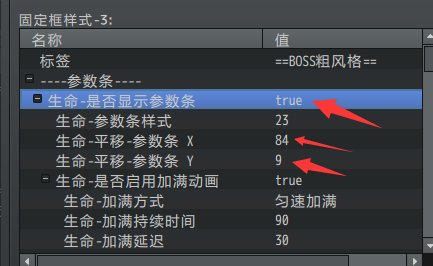
这里提及一下，固定框样式 和 BOSS配置 之所以分开，是因为前者是样式，后者是数据。*数据有可能会在游戏中临时修改，而样式是固定之后不再变化的。*

下面，打开高级boss生命框的配置，你会发现有些参数字太多了，显示不全，可以拉动下图的竖线，显示全。



**PS对坐标点**：这里比较关键的是要对齐条的位置与外框的坐标，由于参数条的中心锚点在左上角，所以你需要对齐外框左上角的像素点位置。（这里不考虑旋转情况，但是要留意旋转影响）





然后把背景加上。



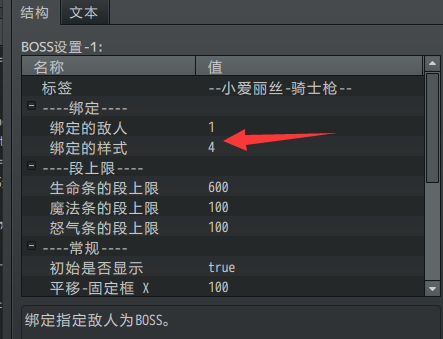
**3) 参数数字配置**

下面进入参数数字插件。参数数字相对来说，就不需要那么严格的吻合要求，不过要注意参数数字的锚点在中心，不是左上角。下图为生命数字和生命段数的配置。

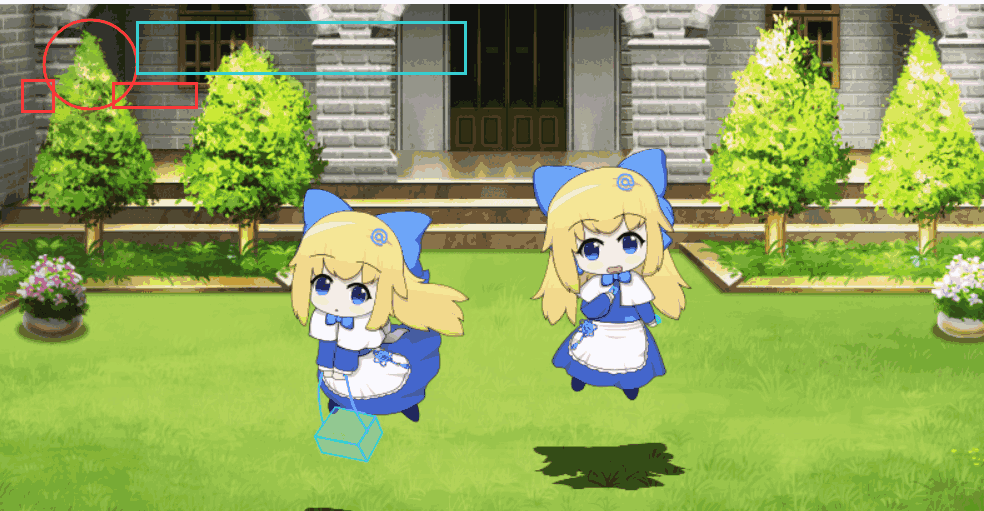


### 5.放置一个实例

固定框样式设计好了之后，就可以进行boss设置来绑定到敌人身上了。生命条的段上限设置等于敌人最大生命值，确保生命条的段，是满状态显示的。

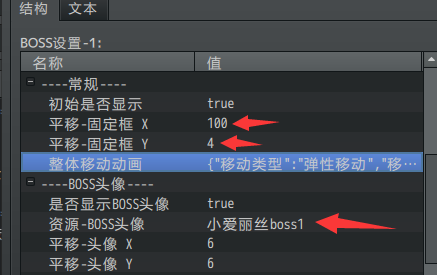


进行测试的时候，你会发现，boss生命条在战斗中，看起来短了许多。另外，框的坐标没有完全居中。

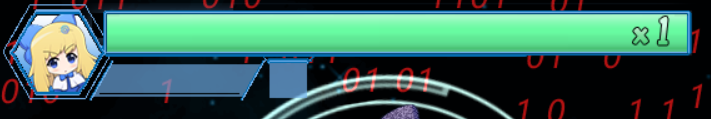


### 6.细节调整

通过在boss设置中，调整boss固定框的位置，顺带把boss头像加上。



可以看到，框相对来说居中了。（boss框的锚点在左上角，不是中心）

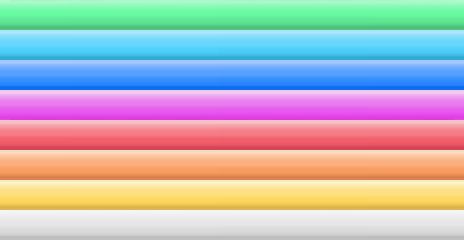
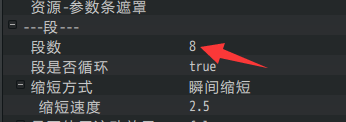


### 7.收尾部分

功能已经完成了大半，还剩下 名字、震动效果、状态 等杂项。这些项都可以通过 [PS对坐标点](#PS对坐标点) 的方式对应。



另外，参数中目前只设计了1段的资源，这里可以进行扩充，变成8段。

## 常见问题

### 如何做出最朴素的可缩短生命条

|  |  |
| --- | --- |
| **问题名称** | 如何做出最朴素的可缩短生命条 |
| **问题图示** | 无 |
| **问题描述** | 很多时候群友只想要一些简单的功能。  但由于插件的配置太多了，  第一反应就是在该插件中可能实现不了。  而这里其实是能实现的。这里专门说明一下。 |
| **解决方案** | 1.选择示例中的粗风格，    2.设置缩短方式为匀速缩短，设置匀速缩短速度。    3.关闭弹出条。    在示例中测试。（比如去15位数战斗测试）  即可完成最朴素的生命条缩短功能。  另外提一下，如果生命下降的速度太快了（比如瞬间打掉n段生命），那么匀速缩短的速度会根据 段上限情况，加速扣除。 |