Gui在构建的前期，需要为之构建一些基础组件，基础组件主要为了整体的系统结构的使用，基础组件的构建度越完整，在上层ui进行开发时所需要用到的一些接口也就越丰富，而不再需要依赖系统中提供，完全由Gui自身去供应，通常Gui特殊的基础组件有以下的内容：

**设计参考（**见scui\_core.h**）:**

**数学库：**

必备的三角函数sin，cos，tan，以及平方根sqrt，通常分为俩类，一类是整数级，因为大部分三级函数都是计算坐标点，而Gui中的坐标点都是整数，所以需要对其进行放大以保留足够的角度，第二类是2.5**D**浮点数扩展的三角函数

扩展的部分如贝塞尔曲线，以及map复用映射，主要做一些线性动画扩展效果使用

**坐标与区域：**

区域的一些基本运算，求俩个区域交集和并集，区域与点包含关系等等

**坐标与区域****（2.5****D扩展，可选）：**

3D坐标点和面的描述，3D矩阵的描述，3D坐标点，3D面的矩阵变换，以及2D到3D的相互转换，以及坐标点的视点变换等接口

**矩阵库（2.5D扩展，可选）：**

矩阵乘法以及乘法逆元，然后就是常规的矩阵变换，如缩放矩阵计算，旋转矩阵计算，放射矩阵计算，投影矩阵计算等等

**基础图元描述：**

描述颜色，透明度，像素，画布相关的一些内容

**句柄（框架可选）：**

因为Gui是面向对象的简要设计，里面一般或多或少都会借鉴相关的一些要求，句柄是从Windows窗口设计中导入的概念，主要用于描述资源的泛用体现，任何资源都可以表示为句柄，它可以是一个控件实例，可以是一个图片描述，可以是一个字库名字，可以是一个多国语的字符串等等，具体看实际的使用

**动画：**

当前Gui的动画设计参考借鉴lvgl的动画设计，其中主要的吸收部分是行程以及路径的概念，除此以外其他部分和软件定时器倒没有别的区别，可以使用软件定时器做稍微的修改得到

**系统事件表：**

系统事件调度描述了Gui应该处理关心的一些，必不可少的内容，每一个事件背后都对应Gui对于该类问题的处理流程，并且Gui的事件也具备与上层UI相互交互的效果，这就保证了UI设计足够的灵活性

**画布帧缓冲：**

双buffer的多线程异步，使用画布作为数据承接载体，搭配互斥锁和信号量，完成绘制与送显的异步的一个可能的泛用模板