

# CuraEngine项目分析

author@YonghaoFu

## 项目简介

CuraEngine 是Github上面一个开源项目。主要功能是将STL文件变成Gcode代码。

## Gcode&&STL

- Gcode  
&emsp;这里对Gcode代码进行简单介绍。Gcode代码类似于汇编代码，可以直接在底层硬件上运行，控制电机等进行运动。Gcode代码跟汇编代码一样不只一套标准。在不同类型（品牌）的打印机Gcode代码略有不同。点击[这里](#)对Gcode进一步了解。
- STL  
STL文件分为**二进制STL**、**ASCII类型STL**。CuraEngine项目只能处理STL文件，不能像其项目介绍中写得那样能处理OBJ等模型文件。对STL文件的进一步了解可以看看这个[STL参考资料](#)。

## CuraEngine的依赖

CuraEngine项目还依赖于两个需要单独编译的外部库——**protobuf&&libArcus**，如果对其依赖图感兴趣，看[这里](#)

- **protobuf**  
protobuf的功能类似[XML](#)，主要功能是用在数据存储和传输协议等场合。实现的主要功能是将数据序列化和反序列化。
- **libArcus**  
libArcus和protobuf之间的联系特别紧密。它创建与后端之间的数据传输端口。简单来说就是基于protobuf创建线程之间的通信socket。
- **Clipper**  
Clipper实现对图形的剪裁和偏移

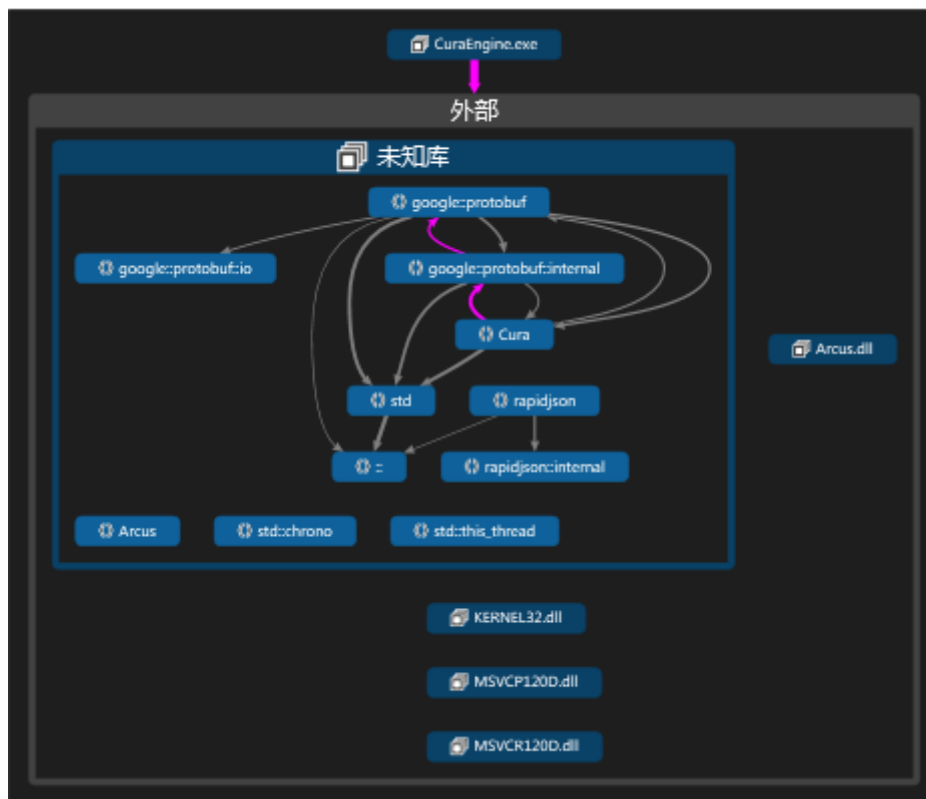
## 路径规划和Gcode代码生成

CuraEngine中的Gcode代码生成模块主要由一下几个组成：

1. **PathOrderOptimizer:**  
这部分代码旨在计算最优路径。找到最优的打印顺序。这个打印顺序是指多边形 ( Polygon ) 的顺序。
2. **Infill:**  
决定了多边形的顺序之后就需要用线将多边形区域填充。
3. **Comb:**  
用于检测当打印机不打印移动时是否出现漏洞。和最终生成GcodeExport联系紧密。
4. **GCodeExport:** 分为两步生成最终的Gcode代码。首先对每一层打印需要的动作进行整合。之后在生成最终的代码。这里将会决定生成的Gcode适合用于哪种机器。

[Gcode生成参考链接](#)

## CuraEngine结构图



## 参考资料

- [林楠的博客](#)
- [CuraEngine小白学习笔记](#)