

# 柴油机数字化研发环境构建技术研究

XXGC-KK-05-04-04/185-2021

---

## APP 软件开发需求分析报告

---

上海交通大学



# APP 软件开发需求分析报告

编 制

校 核

审 查

标准审查

审 定

批 准



# 目 次

1 引言 .....	1
1.1 编写目的 .....	1
1.2 项目背景 .....	1
1.3 引用文件 .....	1
2 APP 软件需求概述 .....	2
2.1 软件描述 .....	2
2.2 软件功能 .....	2
2.3 用户特点 .....	3
2.4 一般约束 .....	3
3 APP 软件需求 .....	3
3.1 需求概述 .....	3
3.1.1 APP 软件开发目的 .....	3
3.1.2 程序设计准则 .....	4
3.1.3 编程语言的选择 .....	5
3.1.4 APP 软件结构 .....	5
3.1.5 运行环境 .....	5
3.1.6 用户特点 .....	6
3.1.7 约束条件 .....	6
3.2 APP 软件描述 .....	6
3.2.1 总体性能计算 .....	6
3.2.2 增压系统匹配计算 .....	11
3.2.3 排气系统设计计算 .....	14
3.2.4 结构强度计算 .....	15
3.3 用户接口 .....	16

3.3.1 软件整体界面 .....	16
3.3.2 框架布局 .....	17
3.3.3 工作区 .....	17
3.3.4 基本按钮 .....	18
3.4 数据交换 .....	18
3.5 文件操作 .....	19
4 APP 软件开发流程 .....	19
4.1 总体开发计划 .....	19
4.1.1 开发方法 .....	19
4.1.2 产品标准 .....	20
4.2 详细开发计划 .....	23
4.2.1 软件开发环境的搭建 .....	23
4.2.2 软件框架的构建 .....	23
4.2.3 界面开发 .....	23
4.2.4 计算代码的编写 .....	24
4.2.5 软件的测试 .....	26
4.3 软件开发过程 .....	28
4.3.1 软件系统设计 .....	29
4.3.2 知识模型的开发 .....	30
4.3.3 软件的编码与测试 .....	30
4.3.4 软件应用 .....	30
4.3.5 项目总结 .....	31

## 图表目录

图 1 APP 软件结构及运行机制 .....	5
图 2 软件构架 .....	23
图 3 使用 Qt Quick 开发的第一版软件界面 .....	24
图 4 软件测试流程 .....	27
图 5 总体方案及技术路径 .....	28
表 1 油耗计算 IPO 表 .....	6
表 2 功率计算 IPO 表 .....	7
表 3 理想循环计算 IPO 表 .....	7
表 4 综合性能分析 IPO 表 .....	8
表 5 能量平衡分析 IPO 表 .....	9
表 6 放热率分析 IPO 表 .....	10
表 7 排气温度预测 IPO 表 .....	11
表 8 单增压器匹配 IPO 表 .....	12
表 9 两级增压匹配 IPO 表 .....	12
表 10 相继增压匹配 IPO 表 .....	13
表 11 进排气损失计算 IPO 表 .....	14
表 12 脉冲排气系统设计 IPO 表 .....	14
表 13 MPC 排气系统 IPO 表 .....	15
表 14 MIXPC 排气系统设计 IPO 表 .....	15
表 15 输入参数在内存中的存储数据结构 .....	25
表 16 ParameterTable 类定义的方法及其说明 .....	25

# APP 软件开发需求分析报告

## 1 引言

### 1.1 编写目的

本文档旨在针对柴油机数字化研发构建项目中的 APP 软件进行说明，确定所有 APP 软件面向用户时所需要完成的功能以及所有 APP 软件征程运行时对环境及硬件的要求。本文档还将对所有 APP 软件正常运行时所需要的接口等进行定义和规范约束条件。

本文档的预期读者对象为：

- 软件开发方设计人员
- 软件开发程序编写人员
- 软件开发方文档和测试人员
- 软件中应用人员

### 1.2 项目背景

为了构建一个柴油机数字化研发环境，需要将柴油机的开发知识以文档和程序的形式表现出来。本报告详细记录了柴油机数字化研发环境的构建过程。其内容主要包括满足客户需求所需交付的产品，为了完成本项目需进行的工作，实时整个软件开发的初步计划、开发方法和产品依据的标准，详细的软件开发过程描述等。

### 1.3 引用文件

- [1] 柴油机数字化项目任务书
- [2] 内燃机学. 周龙保
- [3] 内燃机性能仿真与优化. 刘永长
- [4] Internal Combustion Engine Fundamentals 2nd Edition. JOHN B. HEYWOOD



- [5] C++ primer plus. Stephen Prata
- [6] C++程序设计. 谭浩强
- [7] 柴油机设计手册 1~7 册
- [8] Qt on Android 核心编程.安晓辉
- [9] Qt Quick 核心编程.安晓辉

## 2 APP 软件需求概述

### 2.1 软件描述

APP 软件开发是《TKZJ 车辆数字化研发环境构建技术研究》项目的主要工作之一。该 APP 软件整合现有的柴油机整体性能，增压系统匹配，排气系统设计和结构强度中的主要计算方法以及设计经验，根据所得知识模型报告中的 APP 软件开发流程开发具体 APP。为柴油机设计提供 APP 计算工具。APP 软件采用交互式操作界面，结合界面上的数据表格、图形曲线和数据文件的方式输出，用户可自行选择输出方式。

### 2.2 软件功能

#### ➤ 整体性能计算：

油耗计算、功率计算、理想循环计算、综合性能计算、能量平衡分析、放热率分析、排气温度计算、加速加载计算。

#### ➤ 增压系统匹配计算：

单增压器匹配计算、两级增压匹配计算、相继增压匹配计算。

#### ➤ 排气系统设计计算：

脉冲排气系统设计计算、MPC 排气系统设计计算、MIXPC 排气系统设计计算、三通管压力损失计算。

#### ➤ 结构强度计算：

柴油机活塞连杆曲柄动力学计算、轴承载荷计算、曲拐刚度计算、曲轴

疲劳强度计算、接触副微动疲劳计算、接触副微动磨损计算、轴系扭转振动计算。

- 文件管理：模型文件、数据文件（打开、编辑、保存）。
- 交互操作：界面窗口、输入（文件输入、手动输入、对应参数输入提示）、输出（图形，表格、数值）
- 帮助说明：软件帮助文档

## 2.3 用户特点

从软件的生命周期看，涉及软件使用人员和软件开发人员：

- 软件开发人员：软件系统的开发、维护、升级。
- 软件使用人员：系统的安装、应用、卸载。

## 2.4 一般约束

本条对设计系统时限制开发者选择的其他一些项作一般性描述。而这些项将限定开发者在设计系统时的任选项。这些包括：

开发周期：系统的开发期限预计为 2020-2022 年，在此期间内应完成软件的分析、设计、编码、测试；

硬件的限制：所有 APP 软件运行对硬件没有特殊要求，普通 PC 即可；

所需的高级语言：所有 APP 软件界面以 Qt 语言为开发工具，计算内核和数据库链接以 C++语言为基础开发工具；

操作系统兼容性：在 Windows7 及以上版本操作系统运行。

## 3 APP 软件需求

### 3.1 需求概述

#### 3.1.1 APP 软件开发目的

发动机在开发初期大部分参数都无法得到，这就需要借鉴以往发动机的数据。在原有数据的基础上优化、改进以达到自己的设计需求，用计算机辅助设计来计算发动机性能，能大大提高发动机设计的效率。对不同的设计目

的，需采用不同的计算模型，例如对优化燃烧、减少排放，就需要采用燃烧模型，对优化内燃机增压系统，就需要对增压系统进行匹配。针对不同的目的编写不同的发动机计算工具，能显著提高设计、分析发动机效率。这便是本项目所要达到的目的。

该 APP 软件主要是集常规的公式计算和设计经验为软件计算内核，为柴油机早期设计提供最直接明了的设计准则和相关的柴油机计算参数。并在计算内核的基础上辅助以软件数据分析功能，使其能根据早期不同的柴油机试验参数，通过数据库的方式对输入数据进行筛选和修正。在输出时，则根据数据库数据集对计算结果进行合理的评估和修正、筛选，使其整体设计结果能够满足柴油机设计理论和真实运行数据双重准则。为柴油机早期开发提供系统的支持。

### 3.1.2 程序设计准则

随着面向对象编程计算的发展，面向对象编程的技术已经应用到很多软件开发工作中。面向对象编程能在很大程度上保证代码的重用性和修改。由于常 APP 软件开发周期短、任务重，在实际编写过程中一定要重视代码重用以及整体的可移植性，便于开发人员在后续工作中对软件进行修改和扩充。因此，所有 APP 软件的开发基于两种准则：

面向对象准则：面向对象准则适用于所有较复杂的 APP 软件，如综合性能计算 APP，在软件整体设计中，设定阀门类、气缸类、管道类等对象。对于界面的设计则整体采用面向对象技术，如窗口类、文本框类、图形显示窗口类等。通过该面向对象的技术，能方便后续根据需求对 APP 软件修改和版本的发布。

面向过程准则：面向过程准则适用于较简单的 APP 软件，该方法避免了类的设计的准则，根据知识模型中软件开发流程图进行流程开发，能极大的减少这类 APP 的开发时间。

### 3.1.3 编程语言的选择

由于常用工具软件需要在微软公司 Windows 平台下运行，所以使用 C++ 编程语言进行项目开发，C++ 语言具有如下优点：

1) C++ 是一种面向对象的编程语言，可以快速地编写各种基于 MICROSOFT .NET 平台的应用程序，MICROSOFT .NET 提供了一系列的工具和服务来最大程度地开发利用计算与通讯领域；

2) C++ 具有高运行效率，在复杂的 APP 软件计算中，能极大的提升计算效率。

3) 利用 C++ 语言进行开发，能够很好的与界面开发工具融合。

### 3.1.4 APP 软件结构

APP 软件的输入包含手动输入和文档输入，在经过数据库筛选后进入计算内核程序进行计算。计算结果经过数据库筛选处理后输出，显示输出的对象主要包含绘图，计算出的表格数据和关键参数。最后为计算结果的数据存储，通过文档类保存为 excel 文档或者 txt 文档。

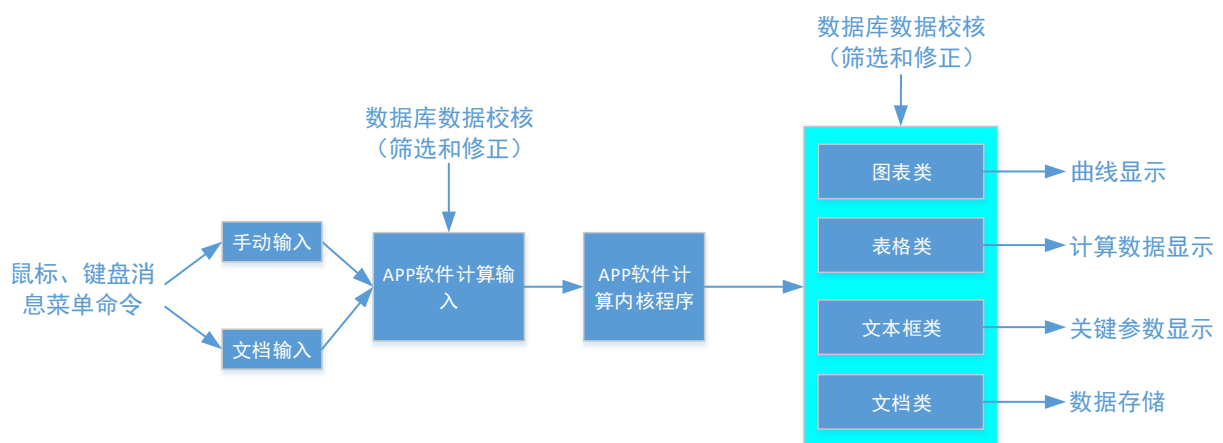


图 1 APP 软件结构及运行机制

### 3.1.5 运行环境

硬件平台：较低的硬件配置，普通家用电脑配置。

操作系统：Windows XP、Windows 7 等等 Windows 系列操作系统。

所有 APP 软件都不会与其他柴油机计算软件存在兼容性问题和共存性问题。

### 3.1.6 用户特点

目标用户为：APP 软件主要面向从事柴油机开发和柴油机性能计算的工作人员。

特点：该人群对柴油机结构和性能具有较深刻的认识，使用该软件主要是能够为工作人员提供快速的柴油机性能计算，从而提高工作效率。

### 3.1.7 约束条件

开发期限：

2020 年~2022 年，提交 APP 软件的第一版。

## 3.2 APP 软件描述

### 3.2.1 总体性能计算

#### ➤ 油耗计算 APP 开发

研究油耗计算方法，完成油耗计算知识模型报告，该模型由功率、转速和基本的尺寸参数计算发动机在该工况下的有效燃油消耗率，该模型的基本原理为由经验公式计算指示热效率和机械效率，而有效燃油消耗率是这两者的函数，有效燃油消耗率可以由指示热效率和机械效率求得。

表 1 油耗计算 IPO 表

输入参数	输出参数	处理方法
气缸数	燃油消耗率	经验公式
缸径		
冲程		
标定转速		
标定功率		

#### ➤ 功率计算 APP 开发

发动机的功率计算基于发动机的尺寸参数和发动机的类型，查找资料，统计多种类型发动机的平均有效压力范围，平均有效压力与发动机排量的乘积得到发动机功率范围。在此基础上完成发动机功率计算知识模型的定义和知识模型报告的撰写。

表 2 功率计算 IPO 表

输入参数	输出参数	处理方法
缸径	最大功率范围	平均有效压力定义式
转速		
冲程		
气缸数		
发动机类型		

➤ 理想循环分析 APP 开发

发动机的理想循环包括多种，在知识模型的研究中，选取卡诺循环、增压混合加热循环以及增压柴油机米勒循环作为研究对象，推导不同循环的热效率计算公式，将推导的过程和柴油该模型计算热效率的程序框图记录在理想循环知识模型的开发报告中。

表 3 理想循环计算 IPO 表

循环形式	输入参数	输出参数	处理方法
卡诺循环	等熵指数	循环热效率	热效率计算公式
	压缩比		
	环境温度		
	环境压力		
	预混燃烧比		
	空燃比		
涡轮增压柴油机循环	等熵指数	循环热效率	
	压缩比	循环最高压力	
	预混燃烧比	循环最高温度	
	空燃比		
	环境温度		
	环境压力		
	压气机压比		
	压气机等熵效率		
	涡轮增压器效率		

涡轮增压米勒循环	等熵指数	循环热效率	
	预混燃烧比	循环最高压力	
	空燃比	循环最高温度	
	环境温度		
	环境压力		
	发动机几何压缩比		
	发动机有效压缩比		
	压气机压比		
	压气机等熵效率		
	涡轮增压器效率		

➤ 综合性能分析 APP 开发

柴油机综合性能分析是最为复杂、输入输出参数最多的模型，该模型的开发是基于发动机容积法，即将发动机进排气管和气缸看作一个个容积，容积内的物理量只随时间变化，由基本的物理定律建立微分方程，求解计算发动机性能参数的模型。该模型包括发动机各个模块的子模型，如 Wiebe 燃烧模型、气缸传热模型、流动模型、容积变化模型、涡轮增压器模型等。撰写的综合性能开发报告详细描述发动机的各个子模型及各个模型结合起来的发动机整体模型的求解方法。

表 4 综合性能分析 IPO 表

输入参数		输出参数		处理方法
气缸设计参数	气缸直径	压力	进气管压力	求解发动机容积法
	冲程		增压器出口压力	
	曲柄连杆比		涡轮前废气平均压力	
	压缩比		平均指示压力	
	气缸数		平均有效压力	
	发火顺序		缸内最高爆发压力	
气缸燃烧	燃烧始点	温度	进气管温度	
	预混燃烧持续期		涡轮前废气平均温度	
	扩散燃烧持续期		最高燃烧温度	
	预混燃烧比例		增压空气温度	
	扩散燃烧比例	性能	有效功率	
气缸传热	缸壁温度		转速	
	活塞顶温度		指示燃油消耗率	
	缸头温度		有效燃油消耗率	
气门	气门直径		扫气系数	

	气门开启角		进气流量	
	气门升程曲线		燃烧开始角	
	气门流量系数		燃烧结束角	
	气门锥角		过量空气系数	
	摇臂比		机械效率	
进排气管	进气管容积	效率	指示热效率	
	排气管容积		有效热效率	
	排气管换热系数		涡轮增压器效率	
涡轮增压器	压气机 MAP 图	传热	充气效率	
	涡轮特性曲线		气缸传热率	
	涡轮增压器机械效率		排气管传热率	
	喷嘴环出口截面积	气门	进气门开角	
	涡轮动叶出口截面积		进气门关角	
中冷器	中冷器效率		进气阀最大升程	
	中冷器冷却水进口温度		排气门开角	
	中冷器压力损失系数		排气门关角	
环境	大气温度		排气阀最大升程	
	大气压力			
性能参数	发动机转速			
	发动机功率			
	机械效率			

### ➤ 能量平衡分析 APP 开发

发动机的能量平衡分析运用能量守恒定律，分析进出发动机的能量流动及其分布状况，并且研究发动机的运行和设计参数对能量分布的影响，进而达到提高能量利用的目的。发动机的能量平衡知识模型开发也是基于能量守恒和转化定律来推导的，在能量平衡知识模型开发报告中，将燃料燃烧所释放的能量分为做功、冷却水带走的能量、排气带走的能量和其它未计损失。

表 5 能量平衡分析 IPO 表

输入参数	输出参数	处理方法
有效功率	有效功率占比	利用公式将计算每一部分的能量
有效燃油消耗率	冷却液能量流占比	
过量空气系数	排气能量流占比	
进气温度	中冷损失占比	
排气温度	其他未记入损失	



	占比	
冷却水质量流量		
冷却水进口温度		
冷却水出口温度		
中冷器冷却水质量流量		
中冷器冷却水进口温度		
中冷器冷却水出口温度		

➤ 放热率分析 APP 开发

由实测示功图计算燃油燃烧的放热率，是研究燃烧过程的一种有效方法。真实的放热规律对燃烧过程不仅能做出定性的说明，而且能够提供定量的估计，是诊断燃烧的一种有效手段。同时，根据放热率也可以分析一些设计参数对燃烧过程的影响，如供油规律、喷油提前角、滞燃期、燃烧持续期、燃烧放热速度等。综合考虑放热率、最高爆发压力和压力升高率，提出理想的放热率曲线，使实际放热过程与其逼近，用以改进和组织燃烧。柴油机放热率计算知识模型研究报告根据缸内工作过程缸压变化来计算缸内的放热过程，计算时需要先知道缸内的初始成分，燃烧过程缸内成分不断发生变化，同时温度也不断变化。

表 6 放热率分析 IP0 表

输入参数	输出参数	处理方法
示功图数据	燃烧放热率	发动机燃烧 过程能量平 衡公式
缸径	累计放热量	
冲程		
连杆长度		
气缸数		
有效功率		
发动机转速		
有效燃油消耗率		

➤ 排气温度预测 APP 开发

增压器匹配计算中，排温模型是影响计算的重要模型之一。目前，柴油机排温计算模型有几种，包括苏联奥尔林教授介绍的与空气混合前后叠加的

排温计算法;建立在排气管孤立系统的能量方程上的排温计算法;利用发动机扫气的热平衡关系整理;顾宏中教授的排温计算模型;由热力学第一定律能量流入流出推导;苏联工程师西姆从热平衡方程角度介绍的排温计算方法，即热力学第一定律的修正模型。在以上列举的几种排温预测模型中，虽然表示形式不同，但都是从热平衡方程出发，通过能量守恒定律来求取，所不同的只是建立热平衡方程时所假设的孤立系统不同，同时在计算公式中会涉及不同的系数选择问题。前两种方法涉及到的不确定参数较多，并且每个参数的计算都会产生一定误差，从而计算得到的结果的可信度不高;而第三种方法一般适用于有进气管压力波数据的情况，此项数据较难得到，最后三种计算模型具有输入参数少、预测较准确等特点，在实际涡轮增压器匹配计算中应用较多。本项目采用顾宏中教授的排温计算模型来预测柴油机的排气温度，该模型适用于在标定点的排气温度的计算，排温计算模型开发报告中详细描述采用该模型的计算过程，画出排气温度的计算流程图。

表 7 排气温度预测 IPO 表

输入参数	输出参数	处理方法
气缸数	排气温度	排温计算模型
缸径		
冲程		
转速		
有效功率		
有效燃油消耗率		
过量空气系数		
扫气系数		
进气管中空气的温度		

3.2.2 增压系统匹配计算

➤ 单增压器匹配 APP 开发

结合增压器匹配原理，建立单增压器匹配知识模型。根据知识模型，完成单增压器匹配 APP 软件开发工作，并形成具体的 APP 软件开发报告，软件测试报告。

表 8 单增压器匹配 IPO 表

输入参数	输出参数	处理方法
气缸数	涡轮当量流通面积	1) 涡轮增压器能量平衡方程 2) 涡轮等效喷嘴流量公式 3) 发动机经验公式
缸径	最大扭矩点压气机流量	
冲程	最大扭矩点压气机压比	
涡轮效率	最大扭矩点涡轮膨胀比	
压气机效率	标定点总过量空气系数	
中冷器冷却水进口温度	废气旁通阀有效流通面积	
中冷器冷却效率		
中冷器压降		
扫气系数		
充气效率		
机械效率		
外界环境温度		
外界环境压力		
燃油低热值		
最大扭矩点总过量空气系数		
最大扭矩点转速		
最大扭矩点功率		
最大扭矩点燃油消耗率		
标定点过量空气系数		
标定点转速		
标定点功率		
标定点燃油消耗率		

### ➤ 两级增压匹配 APP 开发

结合增压器匹配原理，建立两级增压匹配知识模型，并在两级增压试验台上开展模型验证研究。根据知识模型，完成两级增压匹配 APP 软件开发工作，并形成具体的 APP 软件开发报告，软件测试报告。

表 9 两级增压匹配 IPO 表

输入参数	输出参数	处理方法
气缸数	高压级压气机压比	1) 涡轮增压器能量平衡方程 2) 涡轮等效喷嘴流量公式 3) 发动机参数经验公式
缸径	高压级压气机流量	
冲程	高压级涡轮膨胀比	
高压级涡轮效率	高压级涡轮当量流通面积	
高压级压气机效率	低压级压气机压比	
低压级涡轮效率	低压级压气机流量	
低压级压气机效率	低压级涡轮膨胀比	

中冷器冷却水进口温度	低压级涡轮当量流通面积	
中冷器冷却效率	标定点涡轮旁通阀当量流通面积	
中冷器压降		
扫气系数		
充气效率		
机械效率		
外界环境温度		
外界环境压力		
燃油低热值		
扭矩或功率		
转速		
燃油消耗率		
过量空气系数		

### ➤ 相继增压匹配 APP 开发

结合增压器匹配原理，建立相继增压匹配知识模型。根据知识模型，完成相继增压匹配 APP 软件开发工作，并形成具体的 APP 软件开发报告，软件测试报告。

表 10 相继增压匹配 IPO 表

输入参数	输出参数	处理方法
气缸数	设计点涡轮当量流通面积	4) 涡轮增压器能量平衡方程 5) 涡轮等效喷嘴流量公式 6) 发动及运行参数经验公式
缸径	设计点压气机压比	
冲程	设计点压气机流量	
涡轮效率	设计点涡轮膨胀比	
压气机效率	验证点过量空气系数	
中冷器冷却水进口温度		
中冷器冷却效率		
中冷器压降		
扫气系数		
充气效率		
机械效率		
外界环境温度		
外界环境压力		
燃油低热值		
设计点功率或扭矩		
设计点转速		
设计点燃油消耗率		
设计点过量空气系数		
验证点功率或扭矩		

验证点转速		
验证点燃油消耗率		

3.2.3 排气系统设计计算

➤ 进排气三通损失 APP 开发

根据流体力学原理、三通高速流动损失测试，建立三通损失系数知识模型。

表 11 进排气损失计算 IPO 表

输入参数	输出参数	处理方法
流量比	压力损失系数	7) 根据动量方程和连续性方程推导 8) 根据合理得经验公式假设
面积比		
夹角		
流型		
当地马赫数		
各管马赫数		

➤ 脉冲排气系统设计 APP 开发

结合一维流动工作过程计算模型、三通损失试验，建立脉冲系统设计知识模型。根据知识模型，完成脉冲排气系统设计 APP 软件开发工作，并形成具体的 APP 软件开发报告，软件测试报告。

表 12 脉冲排气系统设计 IPO 表

输入参数	输出参数	处理方法
气缸布置结构	排气总管直径	9) 排气管设计经验
气缸数	排气支管直径	
点火顺序	排气支管缩口率	
缸径	排气总管与支管夹角	
用途	脉冲类型	
	排气管布置	

➤ MPC 排气系统 APP 开发

结合一维流动工作过程计算模型、三通损失试验，建立 MPC 系统设计知识模型。根据知识模型，完成 MPC 排气系统设计 APP 软件开发工作，并形成具体的 APP 软件开发报告，软件测试报告。

表 13 MPC 排气系统 IPO 表

输入参数	输出参数	处理方法
缸径	排气总管直径	MPC 排气管设计经验公式
面积缩口率 (0.64-0.68)	排气支管进口直径	
冲程	排气支管出口直径	
转速	排气支管与总管夹角	
功率		

➤ MIXPC 排气系统设计 APP 开发

结合一维流动工作过程计算模型、三通损失试验，建立 MIXPC 系统设计知识模型。根据知识模型，完成 MIXPC 排气系统设计 APP 软件开发工作，并形成具体的单增压器匹配 IPO 表 APP 软件开发报告，软件测试报告。

表 14 MIXPC 排气系统设计 IPO 表

输入参数	输出参数	处理方法
布置结构	排气支管入口直径	MIXPC 排气系统设计经验
气缸数	MPC 管段总管直径	
发火顺序	脉冲排气支管缩口率	
排气门直径	MPC 管段支管缩口率	
缸径	MPC 总管缩口率	
功率	脉冲排气支管出口角	
转速	MPC 管段总管与支管的夹角	

3.2.4 结构强度计算

➤ 活塞曲柄连杆机构运动学计算 APP 开发

运用曲柄连杆机构运动基本原理，建立活塞曲柄连杆机构运动学知识模型。根据知识模型，完成活塞曲柄连杆机构运动学计算 APP 软件开发工作，并形成具体的 APP 软件开发报告，软件测试报告。

➤ 曲柄连杆机构动力学计算 APP 开发

运用曲柄连杆机构动力学基本原理，建立活塞曲柄连杆机构运动学知识模型。根据知识模型，完成曲柄连杆机构动力学计算 APP 软件开发工作，并形成具体的 APP 软件开发报告，软件测试报告。

### ➤ 轴承载荷计算 APP 开发

运用柴油机设计基本原理，建立轴承载荷知识模型。根据知识模型，完成轴承载荷计算 APP 软件开发工作，并形成具体的 APP 软件开发报告，软件测试报告。

### ➤ 轴系扭转振动计算 APP 开发

结合现有轴系扭振分析方法，建立适合柴油机轴系的扭转振动知识模型。根据知识模型，完成轴系扭转振动计算 APP 软件开发工作，并形成具体的 APP 软件开发报告，软件测试报告。

### ➤ 曲拐刚度分析 APP 开发

结合有限元方法、经验计算方法，建立柴油机曲轴刚度分析知识模型。

### ➤ 曲轴疲劳分析 APP 开发

结合有限元方法、经验设计方法，建立柴油机曲轴疲劳分析知识模型

### ➤ 接触副微动磨损分析 APP 开发

结合有限元方法、微动磨损模型、材料和部件的微动磨损试验，建立柴油机接触副微动磨损知识模型。

### ➤ 接触副微动疲劳分析 APP 开发

结合有限元方法、微动疲劳理论，建立柴油机接触副微动疲劳分析知识模型。

## 3.3 用户接口

### 3.3.1 软件整体界面

所有 APP 软件整体界面主要由标题栏、菜单栏、工具栏、工作区、工作区中的选项卡及信息提示区等组成。根据不同软件的复杂程度，合理的进行界面布局。

标题栏：左侧为 LOGO，标题为该 APP 软件名称。居左显示，右侧为最小化、最大化和关闭按钮；

菜单栏：包括文件、编辑、操作、视图、工具、结果、帮助等选项；

工具栏：包括新建、打开、保存、打印、帮助等快捷操作；

工作区：用于输入计算参数、输出计算结果、对用户信息提示区（输入出现遗漏或超出范围、为用户提供参考值等）；

对用户信息提示区：用于显示每一个参数的输入定义，输入范围，输入合理否提示，计算过程中输入出现遗漏或超出范围、为用户提供参考值等；

数据的输入：数据可以从界面输入，也可以直接从文件读入（需要增加一个数据读入按钮）。当在一个界面计算出某个参数并保存后，选择选项卡，进入另一个界面，这时上个界面计算的结果是输入参数，那么这时就用读入按钮读入数据，而不需重新输入，输入参数可以选择单位。

### 3.3.2 框架布局

软件整体框架布局通过 Qt 软件实现。通过 Qt 软件中的 Qt Quick 可快捷的实现积木式的软件整体框架布局，通过设定各个部分的位置和大小可实现 APP 软件整体框架的合理化布局。对于单个 APP，从上到小为标题栏，菜单栏，工具栏，工作区。根据软件功能的不同，各个部分的组成可能会发生相应的变化，但大体相同。

### 3.3.3 工作区

工作区主要包含参数输入区域、结果显示区域和按钮区域。根据不同软件的功能，工作区的内容可能会发生细微的变化。工作区的组成较为复杂，且一个控件可能会关联多个内容，且不同的提示信息等容量不同，为保证程序正常运行，工作区需预留一定空间，保证程序的可用性。



### 3.3.4 基本按钮

基本按钮主要包含计算、清除和保存按钮。

#### a) 计算

用户输入相应的参数，点击“计算”按钮，软件首先应该进行判断，如果用户未输入某项参数或者输入的某项参数存在错误，软件应该有错误提示；如果用户输入的计算参数均符合计算要求，则进行相应计算，显示计算结果。

#### b) 清除

用户点击“清除”按钮，则可以将该页面的计算参数和计算结果全部清除。

#### c) 保存

用户点击“保存”按钮，可以将相应的输入参数和计算结果进行保存，此时弹出对话框如下图所示，用户可以自由选择到已有的文件中也可以将结果保存到新文件中。其中保存文件是所有单一公式的信息列表。包含输入参数和计算结果的名称、数值和单位等。下次使用时，用户点击“打开”按钮，软件可以自动从保存结果中，读取相关参数。

每个 APP 软件的具体详细内容，会在 APP 软件开发报告中详细给出，请读者参阅 APP 开发报告。

## 3.4 数据交换

数据文件：采用 Excel 表格的形式保存，可以通过软件读入计算数据，也可以直接打开查看。

数据的存储：每个机型数据对应一个 Excel 文件。在数据文件中每个页面对应存储相应的输入数据和计算结果。各种类型的数据在表中的位置先定义好，软件根据数据位置的定义，从数据文件相应的位置读取数据，并根据相应的位置写入计算结果。在界面中输出的曲线图形不保存，只保存作图的数据。

数据输入：用户界面的数据有两种输入方式：

- 1) 在主工作区直接输入;
- 2) 从数据文件读入。

数据输出：用户界面计算的结果通过保存命令，保存到数据文件。

### 3.5 文件操作

文件下面主要包括：导入、保存、另存为等基本功能。

- 导入一个计算案例，在输入参数较多的案例中，可以通过导入已有的输入文件来设定输入参数；
- 保存是将计算案例进行保存，其中输入参数和输出参数将分为两个 Sheet 保存在 Excel 文件中，导入时是导入输入参数 Sheet 到输入参数面板上；
- 另存为是将计算案例保存到别的文件位置（此时应该弹出另存为对话框，用户可以自由选择保存路径。下次使用时，用户点击“打开”按钮，软件可以自动从保存结果中，读取相关参数）。

## 4 APP 软件开发流程

### 4.1 总体开发计划

#### 4.1.1 开发方法

现在流行的软件的开发方法主要有：结构化方法、面向数据结构的软件开发方法、面向问题的分析方法、原型化方法、面向对象的软件开发方法以及可视化开发方法。结构化方法的主要观点是采用自顶向下、逐步求精及模块化的程序设计方法；使用顺序、选择和循环三种基本控制结构构造程序。面向对象设计方法以对象为基础，利用特定的软件工具直接完成从对象客体的描述到软件结构之间的转换。面向对象设计方法的应用解决了传统结构化开发方法中客观世界描述工具与软件结构的不一致性问题，缩短了开发周期，解决了从分析和设计到软件模块结构之间多次转换映射的繁杂过程。面向对象设计方法和结构化方法目前仍是两种在系统开发领域相互依存的、不可替

代的方法。

本软件项目开发过程中，使用面向对象的设计方法和自顶向下的结构化设计方法相结合的方法。首先采用结构化设计方法，根据需求将程序分成几个大的功能模块，然后对每个功能模块逐步细化。在确定主要的程序框架和功能模块后，采用面向对象的程序设计方法，通过把类、对象、属性、操作作为最基本的构件来构造发动机的性能分析模型。根据发动机部件组成和性能仿真的特点，建立不同的类，形成建模、设计、实现等各个级别的可复用性的基础，并从类中实例化不同的对象。

在开发过程中，为了降低开发风险，采用迭代增量式的开发方法。首先根据现有模型，进行第一轮的开发，得到第一个简化版本；随着关键模型的研究深入，对研发版本中的相应模块进行改进和升级，不断推出更新版本，直至整个模型研发结束，软件升级到最后的版本，通过不断升级，改正先前版本的问题和不足。这样的开发方式，可以显著降低开发风险。

#### 4.1.2 产品标准

##### (1) 格式标准

采用 C++ 的标准语言格式

##### (2) 首部注释标准

标明代码的版本号、编制时间、代码的作用；代码实现功能所采取的方法（模型、算法）；程序的基本结构；程序变量的说明（意义、输入、输出、局部、全局等）

##### (3) 其他注释标准

注释行数大于 30%

##### (4) 变量、参数、程序包、过程和文档等的命名约定

###### ➤ 类型和常量命名

类、常量、结构、枚举和 typedef 标识符用“大驼峰法”拼写：大写类型和

常量名中每个单词的首字母可以使这些单词更为清晰。第一个单词首字母大写则为区分类型和变量（或者函数）提供了保证。用名词命名复合类型，用复数名词命名集合类型。

#### ➤ 函数命名

函数名用“小驼峰法”拼写：函数中的一个单词全小写而其它单词首字母大写。用动词命名函数，用 `is`、`set` 和 `get` 命名访问函数或者赋值函数。

#### ➤ 变量和参数的命名

变量和函数参数用“小驼峰法”拼写。用名词命名变量。用前缀或者后缀区别成员变量和其它变量，如 MFC 中的 `m_work`。

为函数全部参数命名：在函数的声明和实现中，应该试用同一个规则为所有参数命名。如果函数参数未试用，可以使用 `static_cast<>` 将参数转换为 `void` 类型，以消除编译器“未使用参数”的警告。例如：

```
class MyClass{
public:
    MyClass (float meaningfulName);
    //...
}

MyClass :: (float meaningfulName) {
    static_cast<void> (meaningfulName);
    //...
}
```

#### ➤ 函数参数和作为其赋值对象的成员变量应有相同名字

```
class Customer{
public:
    Customer (const string& name);
private:
```

```

    string name_;
};

Customer::Customer (const string& name) :name_(name){
}

```

### ➤ 文件命名

文件名要全部小写，可以包含下划线（\_）或短线（-），按项目约定来。

可接受的文件命名：

```

my_useful_class.cc
my-useful-class.cc
myusefulclass.cc

```

C++文件以.cpp 结尾，头文件以.h 结尾。通常，尽量让文件名更加明确。定义类时文件名一般成对出现，如 foo\_bar.h 和 foo\_bar.cc，对应类 FooBar。内联函数必须放在.h 文件中，如果内联函数比较短，就直接放在.h 中。如果代码比较长，可以放到以-inl.h 结尾的文件中。对于包含大量内联代码的类，可以有三个文件：

```

url_table.h          // The class declaration.
url_table.cc         // The class definition.
url_table-inl.h      // Inline functions that include lots of code.

```

### ➤ 通用原则

名字应该有实际意义；使用读者熟悉的名字；避免在名字中使用数字，避免依靠数字区分名字；避免使用过长的名字，实体的名字应该准确描述其用途，名字过长首先应该考察实体的设计和用途，而不是直接换个短的名字，必要时要重构实体；保留元音字母——使用完整单词；不要依靠大小写区分名称，C++对大小写敏感，但是依靠大小写区别易混淆；类型或常量名中第一个单词使用了缩写，则只大写该缩写词的首字母，如果缩写词紧跟其它单词之后这个更重要。

## 4.2 详细开发计划

### 4.2.1 软件开发环境的搭建

软件工程环境就是指支持软件产品开发、维护和管理软件系统。本软件开发的软件工程环境和工具如下：

软件文档编制：Word；

项目管理工具：Project；

软件开发绘图工具：Visio；

数据库设计工具：Power Designer；

面向对象软件设计工具：Rational Rose；

软件开发集成环境：Qt。

### 4.2.2 软件框架的构建

本软件的主要构架和数据交换过程如图 2 所示，软件采用键盘和 CSV 文件两种输入方式，输入的数据可以通过键盘直接在界面上输入，或者作为一个 CSV 文件读取到界面上；界面与计算内核之间的数据交换是由数据交换文件来实现的，界面输入传递给内核后，内核开始计算；若计算过程无异常则计算完成，生成一个结果文件，并将其储存在当前目录，程序读取生成的结果文件并将其展示在界面上，完成双向的数据交互。

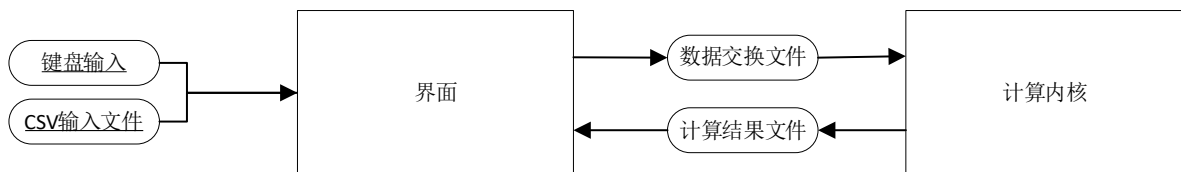


图 2 软件构架

### 4.2.3 界面开发

软件的界面使用 Qt Quick 设计，Qt Quick 是 Qt 4.7 中推出的一项新技术。Qt Quick 是一种高级用户界面技术，使用它可轻松地创建供移动和嵌入式设备使用的动态触摸式界面和轻量级应用程序。以一款放热率计算软件为例。

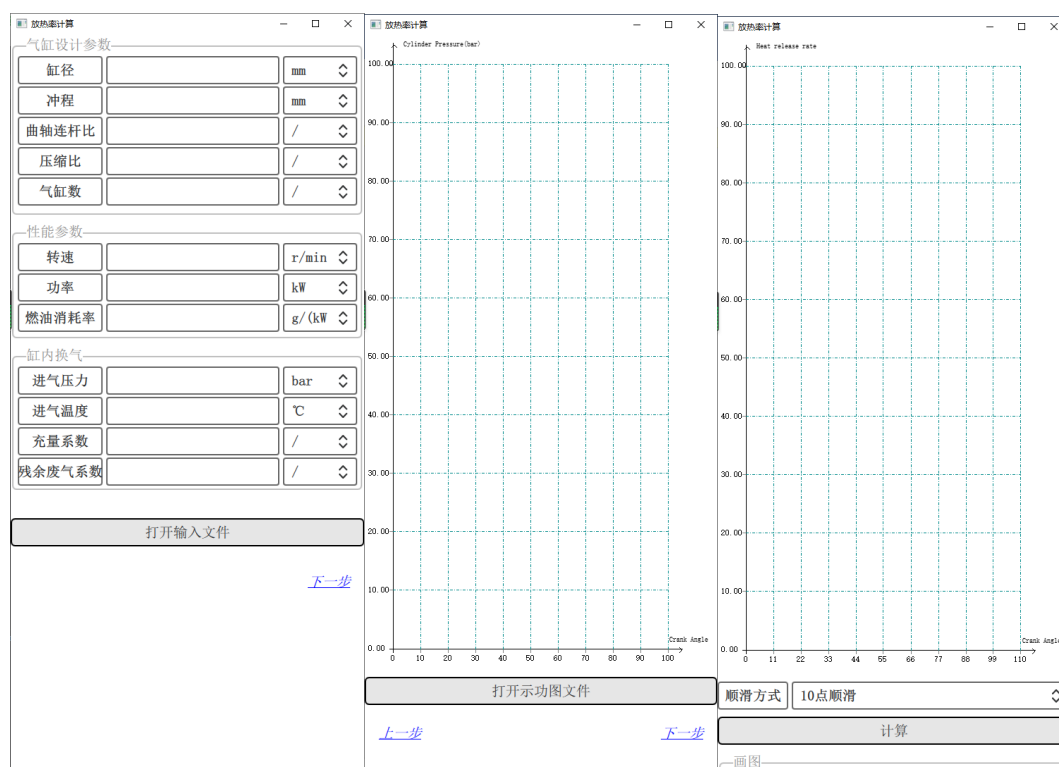


图 3 使用 Qt Quick 开发的第一版软件界面

使用 Qt Quick 开发的放热率计算 APP 界面如图 3 所示, 由于输入过程比较繁琐, 为方便用户使用, 采用上一步下一步的方式将界面分解为三部分: 第一部分为气缸设计参数、性能参数等标量参数的输入, 输入完成后按下下一步按钮, 程序先会检查输入的数据是否合法 (完整且无误), 若输入的数据合法则跳到示功图输入界面, 否则跳出报错窗口, 不进入下一步; 第二部分为示功图输入窗口, 由于示功图的输入数据达到上千行, 所以采用文件的格式输入较为方便, 点击打开示功图按钮, 弹出文件对话框, 当选择示功图数据文件之后, 程序自动画出示功图, 方便用户检查示功图是否有误; 第三部分为计算和展示窗口, 选择示功图的顺滑方式, 点击计算按钮, 计算完成后点击画图画出放热率曲线, 点击打开结果文件系统自动打开计算得到的结果文件, 默认打开的程序为 Excel 表格, 以方便后续的画图和结果数据的使用环节。

#### 4.2.4 计算代码的编写

##### 4.2.4.1 输入的处理

在处理输入文件时定义一个 `ParameterTable` 类来储存输入的数据，该类内的 `table` 指针指向一个线性表，线性表的每一个元素为一个结构，用来储存参数名称、参数的单位和参数值。

表 15 输入参数在内存中的存储数据结构

参数 1	参数 1 单位	参数 1 数值
参数 2	参数 2 单位	参数 2 数值
参数 3	参数 3 单位	参数 3 数值
...	...	...

在 `ParameterTable` 类初始化时，给定读取的文件名，默认文件名为“Input.csv”，读取完成后，将参数的单位化为国际标准单位，方便进行下一步的运算。`ParameterTable` 类还定义一系列的方法，例如为将输入参数赋值给变量，在 `ParameterTable` 类定义一个方法 `search(string index)`，该方法接受参数名称，返回 `table` 表格内对应的参数值，若 `table` 表格内不存在该参数，则报告异常。

`ParameterTable` 类的方法及其说明如下：

表 16 `ParameterTable` 类定义的方法及其说明

方法	说明
<code>bool readInFile(QString filename, QString startIn, int row);</code>	用于读取文件内容，输入文件名，表名称和表的行数，若读取成功则返回 <code>true</code> 。一个文件中可能存在多个表，用表名标识一个表的开始，同时用表行数来控制表读取的停止。表内的参数顺序可以交换，但是参数名称和参数个数均为程序给定。
<code>double searchValue(QString index2);</code>	用于搜索参数值，输入参数名，返回参数值。
<code>bool writeToFile(QString filename, QString tablename, QStringList data, int mode);</code>	用于得到的结果写入文件中，输入文件名、表名、字符串和输入模式， <code>mode=0</code> 时将文件内容删除重新写入， <code>mode=1</code> 时在文件末尾添加内容。
<code>void set_unit();</code>	用于单位转化，为方便用户输入，界面上的参数单位通常不是国际单位，将参数值用于计算时通常先要将其化为国际单位。

#### 4.2.4.2 文件输出

文件输出同样使用一个 `ParameterTable` 类，将计算得到的结果一一填入表



格中，使用 `ParameterTable` 类内的写入文件方法，该方法输入希望写入的文件名，调用该方法时在当前目录下生成一个存储结果的文件。

#### 4.2.5 软件的测试

界面测试主要包括以下几个方面：

##### 1) 界面测试

现代软件都是基于图形化的操作界面，如果界面操作存在 **Bug** 问题，则可能使软件在操作过程中出现闪退报错等现象。界面测试内容包括界面的滑动功能、选择控件选择功能、按钮功能和显示功能。确保在所有界面功能可操作。

##### 2) 输入测试

输入测试主要测试的是从 `Qt` 界面到 `C++` 的数据传输功能，在界面上输入运行所需要的参数，测试数据是否正确传输到内核中，或者打开文件，测试文件输入是否正确填写到界面上并传输到计算内核中。

##### 3) 输出测试

输出测试主要包含源码计算测试和从 `C++` 到 `Qt` 界面的数据传输，即计算的内容正确展示在界面上，还需检查输出文件内数据的正确性。

##### 4) 报错测试

软件运行过程中，会有很多其他的情况导致程序不能继续运行或导致计算出错，在这种情况下，软件应给出报错提示。本软件的报错系统主要包含两部分：一、输入报错。错误的输入信息会导致计算结果出错，或使内部计算过程不能收敛。二、计算报错。当内部的计算程序出现不可预料的错误时，软件会终止计算程序，并给出错误信息。

软件测试的流程如图 4 所示，测试完成后形成测试报告。

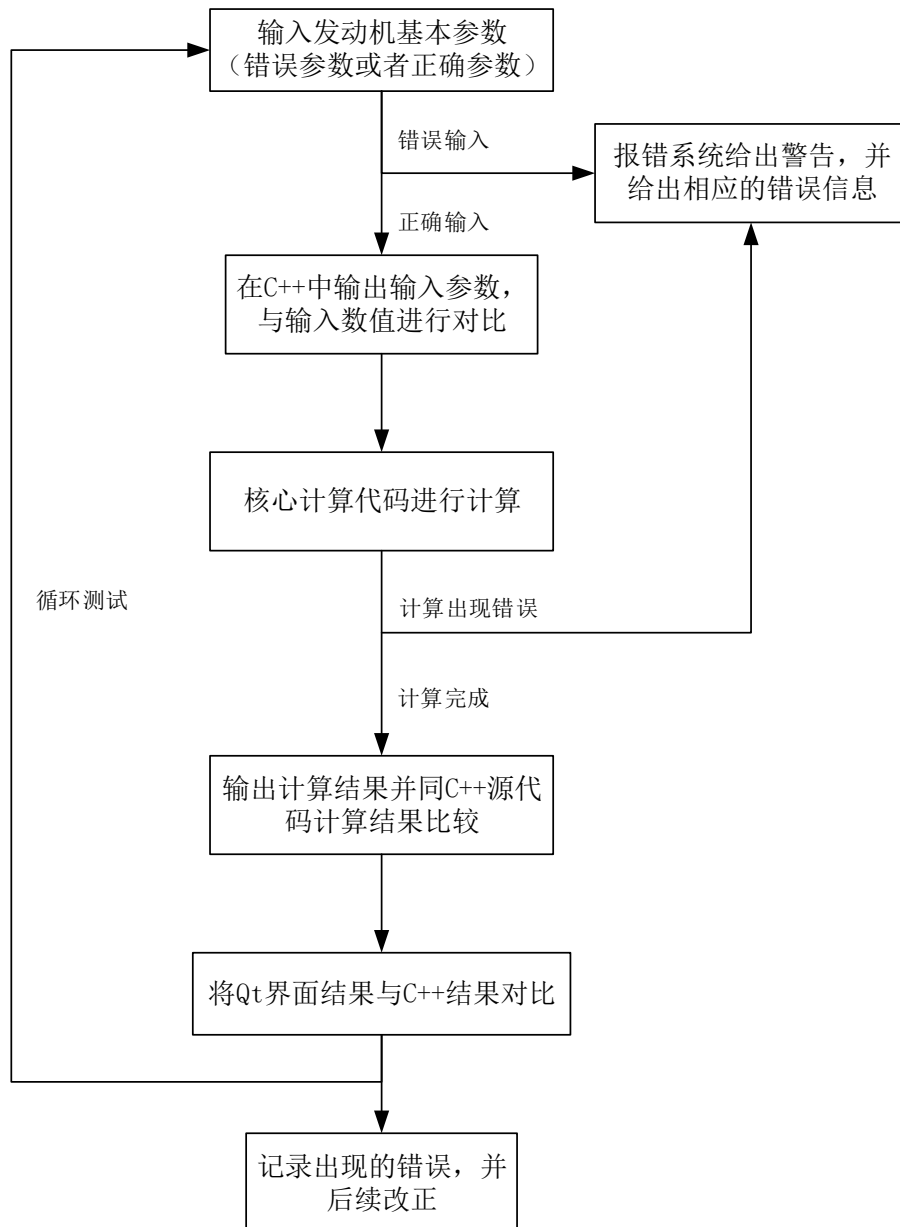


图 4 软件测试流程

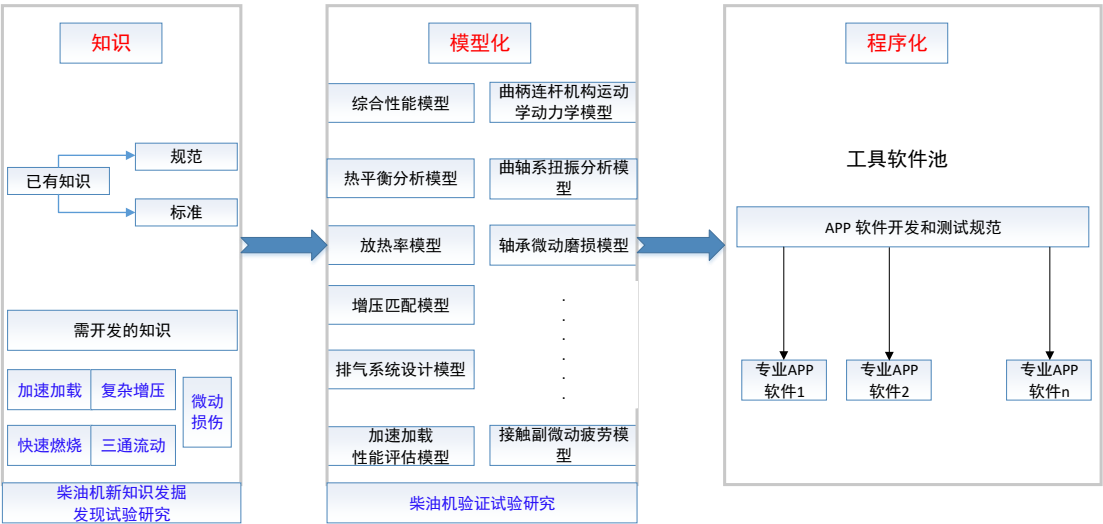


图 5 总体方案及技术路径

4.3 软件开发过程

需求开发阶段，用用户的语言进行描述，让用户和开发团队对于需求达成一致的理解，形成系统要具有的功能。系统设计阶段，结合用户对系统环境、开发语言以及运行的网络硬件等要求，确定开发工具等，对应用系统关系进行架构性设计，通过需求阶段对用户的分析归类，用图的方式描述出用户和各子系统或模块的全局视图，以及和其他系统的关系。

概要设计中除了高层架构设计，还需要设计网络拓扑图，以及系统部署图。概要设计比较重要的还有就是子系统、模块进行合理的划分。模块的名称很大程度上会成为用户的主要菜单，如何用用户的角度去取比较清楚的子系统和模块是很重要的。

代码开发和单元测试阶段，把模块进行合理划分，把项目总体计划的代码开发测试阶段划分为多个时间段，每个时间段都包括代码开发、单元测试和集成测试，这个阶段还需要对需求变更进行跟踪控制，如果需求有变更，那么要把需求文档、设计文档都重新跟上。

测试工作是项目的重要的环节，需要确定测试目标、测试方法，并编写测试用例。

文档工作在项目开发中也占有重要位置，它记录整个开发过程，是用户、团队成员相互沟通的形式之一，也是软件开发的重要成果。

#### 4.3.1 软件系统设计

完成对整个软件系统的分析设计，对开发方式、图形界面、计算模型、数据接口进行定义，对系统功能各模块进行详细设计、定义数据总体结构、编码命名规范。

软件的开发首先基于发动机的基本原理和现有机型的大量数据库，另外还需要大量查找相关论文，了解最新、最准确的计算方式，而不是用旧的、不准确的模型。有些计算无法直接表示为公式，而必须借鉴已知发动机的数据库。

根据性能分析软件的特点，拟把软件分为前处理、求解器和后处理等子系统。前处理子系统建立模型图、设置参数、进行模型预处理、生成模型文件和输入接口文件。求解器读取输入接口文件，完成计算过程后，把需要保存的结果存储在输出接口文件中。后处理子系统读取输出接口文件，并进行输出的图形化显示、处理和保存。

前处理和后处理采用相同的编程语言，用来保证前后处理界面风格的统一，本项目采用 Qt 作为界面的开发工具；编程语言采用面向对象的 C++ 编程语言。

Qt 是一个由 Qt Company 开发的跨平台 C++ 图形用户界面应用程序开发框架，支持在 Linux、Windows 等多个平台开发，开发语言为 C++，Qt 满足一次开发可以将源代码和图形界面编译到多个平台使用，避免了为了适应多个平台多次开发的繁琐过程，所以本软件拟采用 Qt 开发界面。使用 Qt 作为界面开发的软件优势如下：

##### 1. 优良的跨平台特性

Qt 支持下列操作系统:MicrosoftWindows95/98，MicrosoftWindowsNT，

Linux, Solaris, SunOS, HP-UX, DigitalUNIX(OSF/1, Tru64), Irix, FreeBSD, BSD/OS, SCO, AIX, OS390, QNX 等等。

## 2. 面向对象

Qt 的良好封装机制使得 Qt 的模块化程度非常高,可重用性较好,对于用户开发来说是非常方便的。Qt 提供了一种称为 signals/slots 的安全类型来替代 callback,这使得各个元件之间的协同工作变得十分简单。

## 3. 丰富的 API

Qt 包括多达 250 个以上的 C++类,还提供基于模板的 collections, serialization, file, I/Odevice, directorymanagement, date/time 类。甚至还包括正则表达式的处理功能。

除此之外还包括,支持 2D/3D 图形渲染,支持 OpenGL,大量的开发文档和对 XML 的支持等。

### 4.3.2 知识模型的开发

完成柴油机知识模型化的规范定义,将柴油机知识模型化分为知识概述、术语定义、模型 IPO 表、模型的原理和模型的实现过程程序框图几部分,后续的具体知识模型定义都是在此规范上进行的。

### 4.3.3 软件的编码与测试

根据软件的系统架构,进行前处理、求解器、后处理的代码编程和测试。为了避免错误积累,采用边开发边测试的基本模式,对每个模块都进行单独测试,系统联调及系统测试,对系统处理逻辑、例外处理能力、容错能力等进行大规模的测试,对发现的问题进行彻底纠正。

### 4.3.4 软件应用

开发的软件首先在两台实验机上试运行,两台发动机的运行参数先通过实验获得,然后用软件计算其性能,对比分析实验和计算结果的误差以验证软件的准确性。

#### 4.3.5 项目总结

根据项目结题要求，对项目研发过程进行总结备案，以利于项目经验的积累和后续改进。