****



开题报告

**基于RTCA Do178C的**

**FADEC软件覆盖测试的研究**

|  |  |
| --- | --- |
| 学院名称 | 计算机学院 |
| 专业名称 | 计算机科学与技术 |
| 学生姓名 | 王鑫冶 |
| 学号 | **SY1506325** |
| 指导教师 | 曹庆华 |

2016年12月

# 1.选题依据

本课题来源于工信部的民用飞机专项科研“航空发动机电子控制系统适航审定关键技术”中的子课题“航空发动机电子控制器软件和硬件适航审定技术研究”。针对大型客机涡扇发动机电子控制系统软件的适航审定问题，开展相关的基础技术研究。

我国大型客机采用翼吊式大涵道比涡轮风扇发动机，将选用LEAP-1C和我国自主研制CJ-1000AX型号发动机提供动力。一方面，由于国外长期对航空发动机实施技术封锁，国内航空发动机研制相对国外具有较大差距，与之配套的适航审定经验不足，给航空发动机适航审定带来很大困难。另一方面，CJ-1000AX项目是《国家中长期科学和技术发展规划纲要》大型飞机专项的重要组成之一。其控制系统的适航性审查工作是CJ-1000AX验证机项目的重要组成部分，也是未来通过适航审定、取得型号合格证的基础，更是产品进入市场、取得商业成功、实现国家战略的基础。

# 2.国内外研究现状分析

## 测试

ＩＥＥＥ将软件测试定义为：使用人工或自动化手段运行或测定某个系统的过程，其目的在于检验它是否满足规定的需求或是发现预期结果与实际结果之间的差别。嵌入式软件测试建立在传统软件测试基础之上。

依据测试时是否需要执行被测软件，可以将软件测试技术分为静态测试和动态测试。静态测试是指不运行程序本身，主要采用代码审查和代码静态分析等方式，对于嵌入式软件，该测试只需在宿主机上进行即可。动态测试需要使用和运行被测软件，观察软件运行时的情况，从中发现错误。对于嵌入式软件来说，为了保证测试的真实性，一般要求在目标机上进行。

从测试是否针对系统的逻辑结构和处理过程这个角度，可以将测试方法分为白盒测试和黑盒测试。

黑盒测试（又叫功能测试或基于规格说明的测试）是站在最终用户的立场，检验输入输出信息及系统性能指标是否符合规格说明书有关功能需求及性能需求的规定。黑盒测试依据软件产品的需求规格说明书来设计测试用例，并在计算机上进行测试，以检测每个实现了的功能是否符合要求。黑盒测试并不涉及程序的内部特性和结构。黑盒测试方法包括等价类划分、边界值分析法、判定表、正交试验法等。在进行嵌入式软件黑盒测试时，须把单元模块、子系统、系统的预期用途作为重要依据，根据需求、设计（概要设计及详细设计）中对接口、负载、、性能等要求，判断软件是否满足设计要求。

白盒测试亦称为结构测试或逻辑驱动测试，主要是通过考查程序的结构和逻辑，验证所构造的程序是否符合设计要求。白盒测试方法的出发点是：因为程序的结构和逻辑是为了实现某种设计构造的，所以就要考察这些结构和逻辑在运行中的实际作用，验证它们是否符合设计要求。测试时设计一组特定的输入作为测试数据以确保特定的结构和逻辑在程序运行中实际运转起来。白盒测试方法认为程序是一系列的逻辑路径的动态组合，测试的目标之一是尽可能覆盖所有路径。白盒测试方法包括程序分析、程序插桩、覆盖测试等基于软件内部结构的测试技术。

覆盖测试技术

覆盖测试概述

覆盖测试是一种白盒测试方法，测试人员必须拥有程序的规格说明和程序清单，以程序的内部结构为基础，来设计测试案例。它的基本准则是设计足够的测试用例来尽可能多地覆盖程序的内部逻辑结构，以发现其中的错误和问题。

覆盖测试技术通过运行被测程序，计算得到程序各类语句执行的覆盖率，并对代码的执行路径覆盖范围进行评估、分析。覆盖测试的指标回答了测试的完全程度如何这一问题，帮助测试人员找出被测程序中的错误。覆盖测试以程序内部的逻辑结构为基础，设计若干测试用例，使程序在这些测试用例运行时，能执行程序中的每个语句／分支／条件／路径／子程序调用等等。因此，又称为逻辑覆盖。

基于需求的软件覆盖率分析

测试覆盖分析可分为需求覆盖分析和结构覆盖分析。需求覆盖分析首先分析与软件需求有关的测试用例，以确认选择的测试用例满足规定的准则；其次确认基于需求的测试程序测试了编码结构。

需求覆盖分析的目标是确定基于需求的测试怎样验证软件需求的实施。这种分析可揭示对附加的基于需求的测试用例的要求。需求覆盖分析应表明：

1. 每一个软件需求都有测试用例；

2. 测试用例满足正常范围测试和鲁棒测试的准则。

结构覆盖分析的目标是确定通过基于需求的测试程序未测试到的编码结构。

基于需求的测试用例可能没有测试到所有编码结构,所以应完成结构覆盖分析并进行附加验证, 以提供结构覆盖。分析应：

3. 证实与软件等级相应的结构覆盖率的程度；

4. 可在源代码上完成结构覆盖率分析，如果软件等级是 A 级且编译程序产

生不能直接追踪到源代码语句的目标码，那么应在目标码上完成附加验证以确定产生代码顺序的正确性。在目标码中,编译程序产生的数组边界检查是目标码不能直接追踪到源代码的一个例子；

5. 证实在代码部件之间的数据耦合和控制耦合。

结构覆盖分析可揭示在测试中没有测试到的编码结构。解决方法是进行附加软件验证过程活动。这个未执行的代码结构可能是由于下列因索造成的：

1. 基于需求的测试用例或测试程序的不足，这些测试用例应加以补充或更

改测试程序，以提供丢失的覆盖率。用于完成需求覆盖率分析的方法可

能需要加以评审；

2. 软件需求的缺陷，应修改软件需求,开发额外的测试用例，执行测试程序；

3. 死码，应消除这种代码，完成分析以评定这种影响及是否需要重新验证；

4. 无效码，对于在航空器或发动机中所用的任何配置中不打算执行的无效

码,可通过分析和测试来表明防止、隔离或消除无意中执行这种码的方法。对仅在目标机环境的某些配置中执行的无效码，要确定正常执行这种代码要求的操作配置,并且要开发附加的一些测试用例和测试程序以满足要求的覆盖率的目标。

2.3.2 机载软件测试特点

机载软件的基本测试方法同通用软件测试方法一样，但又存在与通用软件测试不一样的自身的特点。

它有两个互补的目标。一是证明软件满足其需求;二是以高置信度证明由系统安全性评估过程确定的可能导致不可接受的失效状态的那些错误已被消除。

机载软件测试的有三个特点：

1. 机载软件多为嵌入式软件，嵌入式软件具有两个显著特点:一是软件和硬件结合紧密，软件脱离特定系统往往无法运行，软件失效与硬件故障有时难以区分，甚至相互干扰。二是开发环境和运行环境不同。因而，嵌入式软件还必须进行目标环境的测试。

嵌入式软件的这两个特性给软件测试提出了新的挑战，一些新的问题必须重视。比如，宿主环境和目标环境测试的分配、软-硬件错误的界定、硬件特性对软件测试的影响、测试所需的硬件信号驱动及响应等等。

2. 机载软件实时性强。机载软件多是实时软件。不但要求软件执行实时性强，而且要求多个任务能协调执行。例如其中的飞控软件，一般根据任务组在飞控系统中所具有的作用、重要程度和其本身的变化率，分成各种不同速率组任务调度执行。为了满足不同任务优先级的要求，还要进行中断管理。因此在机载软件测试中，不但要对软件语句的执行逻辑关系进行验证，还要对该过程的执行时间和规定时间进行比较

3. 机载软件属于安全关键软件。其性能的不可靠将带来灾难性的后果，因此对其可靠性、安全性要求很高。为此在机载软件设计时采用一些提高可靠性、安全性的先进技术，主要有容错技术、N 版本技术、安全监控和安全隔离技术等。

先进飞机的航空发动机要求高推重比、低油耗、长使用寿命、大灵活性和高可靠性。发动机性能的充分发挥主要依靠控制系统来实现和保证。因此,控制系统的性能和可靠性对发动机的正常工作十分重要。

传统上是采用机械液压控制器控制航空发动机。随着飞机、发动机技术的发展,对发动机控制也提出了更高的要求,需要监视和控制的参数越来越多,控制回路不断增加,传统的机械液压控制技术已不能满足发动机技术日益发展的需求。当前发动机控制器的硬件已发展为电子控制器,尤其是近年来已日渐成熟的全权限数字电子控制系统(FullAuthorityDigitalEleetronieeontrol,队DEe),具有体积小、重量轻,其控制软件易于编写和修改,并可以实现复杂的控制规律和许多现代控制算法等特点,从而可以满足现代航空发动机控制的需求。近30年来,许多国家都大力从事用数字电子计算机作为航空推进系统核心控制器的研究,取得了巨大的发展和丰硕的成果。目前,国外已基本完成了用FADEC控制系统取代机械液压式控制系统的换代工作。

典型航空发动机FADEC系统的组成如图4所示。由图可见，FADEC系统由供油与能源部件、核心控制器、电液转换装置以及执行机构等部分组成。



图 4 典型航空发动机FADEC系统组成

1. 系统硬件

航空发动机FADEC系统的电子硬件指CPU模块、存储器模块、信号输入和输出模块、通信模块、故障检测和切换电路、电源模块、控制器机箱等。

典型的航空发动机数字式电子控制系统硬件结构如图5所示。虚线部分称为电子控制器（EEC），它是发动机数字式电子控制系统的核心部件.

1. 系统软件

航空发动机控制软件的设计首先根据发动机总体设计提出的性能指标（如推力指标和耗油率指标以及加、减速指标等）和控制规律，研究并确定实现性能指标和控制规律的控制模式和控制算法。控制软件是实现控制模式和控制算法的计算程序。除此之外，软件还要完成相关信号的输入与输出、量纲转换、数据存储、参数变化、数据通信、故障诊断、余度调度和系统重构等功能。



图 5 单通道航空发动机的数字式电子控制系统结构图

通过对发动机工作原理进行研究并分析了典型的FADEC结构图，提出了FADEC系统应该具有以下主要功能：

1. 发动机起动控制（含空中、地面）。该功能用于保证发动机正常点火和顺利启动。
2. 发动机慢车控制。发动机有两种慢车状态：地面慢车和空中慢车。地面慢车用于地面滑行时，转速、推力最小，最省油。空中慢车用于准备着陆时，转速略快于地面慢车，如果着陆不成功，能确保复飞能快速加速。飞机着陆成功后再改成地面慢车。
3. 发动机推力控制。飞机的不同飞行阶段（滑跑、起飞、爬升、巡航、下降、进近、复飞等）需要不同的推力（或功率），对应着发动机不同的工作状态。
4. 停车控制（含紧急停车）。地面停车指飞机着陆时减速至飞机停止。空中停车指飞机发动机在落地之前、起飞之后在空中运行的时候，由于机械故障、电子系统故障、外来物影响（例如鸟击）等原因所引起的发动机停止工作。
5. 加减速控制。快速推动油门操纵杆，使发动机从一种工作状态快速地过渡到另外一种工作状态，从而推力快速增大的过程被称为发动机加速。加速控制的目的是使供油量按一定的规律变化，即使得加速时间最短，但又不发生超温或喘振的现象。

快速收回油门操纵杆使得供油量快速减小，发动机从高转速状态快速降低至低转速状态，从而发动机推力快速减小的过程称为发动机减速。减速控制使得收油门时减油不能过猛，不然可能会导致燃烧室贫油而熄火。

1. 发动机状态参数限制。为了保证发动机的工作安全、可靠，设定各参数（转速限制，温度限制，压力限制，功率限制等）的限制值，各参数的限制值均为发动机的安全极限值，只有当被限制参数超过极限值时，限制器才参与工作。
2. 高压涡轮主动间隙控制。控制发动机不同级的引气，从而保证涡轮叶尖间隙为最佳间隙，减少燃气泄漏，改善涡轮的效率，提高发动机的性能。
3. 反推控制。为了缩短飞机着陆和滑跑距离，张开的反推装置可以更快地减小发动机的正推力。
4. 伺服燃油加热。伺服燃油加温器用发动机滑油去加热伺服燃油。加热燃油的目的是确保燃油中的水分不在伺服系统中冻结。
5. 对发动机滑油进行冷却。燃油从燃油泵流到整体传动交流发电机的滑油冷却器，接着流到滑油/燃油热交换器。滑油/燃油热交换器出来的燃油再流回到燃油泵。
6. 数控系统故障诊断与重构。检测出系统故障存在的情况下及时判明故障类别、找出故障部件、估测故障大小分离故障信号、重构故障信息。
7. 发动机状态监控。发动机状态监控通过从检测到的机械性能参数和气动热力参数中提取信息来识别发动机状态，寻找出发动机的故障，发现故障原因、部位、程度以及趋势，编制相应的检查和维修计划的技术系统，对保障安全飞行、提高发动机利用率、减少燃油的消耗、降低维修成本等方面均有很大的经济意义和社会效益。
8. 数据记录，参数调整和机上和地面维护[21]。

## 2.2 适航审定流程

对于航空机载软件的研制和适航审定，FAA确定了DO-178C、Order 8110.49chg1标准，其中，从适航符合性角度，DO-178C是基础标准，它定义了开发过程需要满足的与相应安全等级相适应的目标；Order8110.49 chg1则属于操作标准，定义了如何实际操作来审查一个项目的研制过程是否满足DO-178C的相应目标要求。DO-178C标准从软件安全性角度，按照机载软件的安全级别对开发过程和活动提出了一系列目标要求；后者则从符合性审查角度规定了具体的审查阶段、活动、涉及的数据和相应的是审查规则等。

Order 8110.49 chg1针对DO-178C定义的开发阶段和相应的目标，规定适航符合性包括四次评审：针对软件计划阶段的符合性审查（SOI#1）、针对软件开发阶段的符合性审查（SOI#2）、针对软件验证阶段的符合性审查（SOI#3）和针对最终适航认证阶段的符合性审查（SOI#4），如表1所示。根据DO-178C标准，软件计划过程针对软件开发全过程涉及的活动和标准进行计划；软件开发过程是一个综合过程，包括开发、验证、配置管理、质量保证、审定联络；软件验证过程综合使用审查、评审、测试、分析等对软件制品进行验证。下表总结了这四个符合性审查阶段关注的目标和涉及的生命周期数据：

表1 符合性审查目标、准则和数据要求

|  |  |  |  |
| --- | --- | --- | --- |
| 审查阶段 | 审查目标 | 合格判定准则 | 审查涉及数据 |
| 计划审查(SO#1) | 检查开发计划和标准是否为满足DO-178C目标要求定义了有效的开发和验证活动。 | 1. 开发计划和标准经过单位内部评审，且评审所发现问题已解决。 2. 开发计划和标准经过了质量保证部门评审，且评审所发现问题已解决。 3. 开发计划和标准得到了批准，并在配置管理中受控。 4. DO-178B/C中表格A-1（即计划阶段应满足的目标列表）所列目标得到了满足。 | 软件合格审定计划、软件开发计划、软件验证计划、软件验证结果、软配置管理计划、软件质量保证计划、软件需求、设计和编码标准、工具认证计划（如果涉及）、软件质量保证记录。 |
| 开发审查(SO#2) | 检查研制方是否按照相应计划开展了开发活动。 | 1. 形成了高层需求文档，经过了评审，并可追踪到系统需求。 2. 定义了软件体系结构，经过了评审和分析。 3. 形成了低层需求文档，经过了评审，并可追踪到高层需求。 4. 实现低层需求的代码可追踪到低层需求，并经过了评审。 | 软件需求、设计和编码标准、软件需求、设计描述、源代码、软件验证过程、软件验证结果、软件生命周期环境配置项、问题报告、软件配置管理记录、软件质量保证记录。 |
| 验证审查(SO#3) | 检查验证计划和过程的有效性和实现情况、是否完成了所有相关的软件配置管理和质量保证任务、是否软件需求、设计、代码和集成都进行了验证；是否软件验证过程达到了DO-178C关于覆盖的要求。 | 1. 开发阶段数据完整，经过了评审，并在配置管理中受控。 2. 测试用例和测试过程形成了文档，经过了评审，并在配置管理中受控。 3. 实施和执行了测试用例和过程。 4. 软件测试环境形成了文档，并受控。 | 软件需求、设计描述、源代码、目标代码、软件验证案例与过程、软件验证结果、软件生命周期环境配置项（含测试环境）、问题报告、软件配置管理记录、软件质量保证记录、软件工具认证数据。 |
| 最终审查(SO#4) | 检查最终产品是否满足DO-178B/C的目标要求；是否所有的开发、验证、质量保证、配置管理和认证联络活动完整执行；是否完成了符合性审查；检查最终的软件配置项SCI和软件项目总结报告SAS。 | 1. 软件符合性审查已完成，所有问题已解决。 2. SAS和SCI已完成且经过了评审。 3. 所有的软件生命周期数据都已完整、得到了批准，并在配置管理中受控。 | 软件验证结果、软件生命周期环境配置项、软件配置项、问题报告、软件配置管理记录、软件质量保证记录、软件项目总结报告SAS。 |

3.3.3

软件测试

软件测试是 DO-178B 中定义的软件验证的一种技术方法。测试的定义是：运行系统

或系统部件，以验证其满足规定需求、检测出错误的过程。机载软件的测试有两个目标：其一是展示软件满足其需求；其二是以高置信度展示那些系统安全评估过程中所确认的、可能导致不可接受失效状态的错误已被消除。从 DO-178B 对软件测试的这两个

目标可以看出，软件测试是对软件开发过程的结果的验证过程，即对软件集成过程的输出通过测试进行验证。

DO-178B 并未给出关于具体软件测试技术方法的指导，而是对于上述三类测试程从以下三个方面提出了软件测试的指导意见：

测试环境：将被测软件加载到目标机中，并在目标机环境的高保真仿真环境中对其进行测试

基于软件需求的测试：软件测试对各级需求的覆盖率需要达到 100%。

基于结构覆盖的测试：软件测试对代码结构的覆盖率需要达到软件重要度等级

所对应的要求。

通过执行测试这一验证过程，可生成与软件测试项目的软件验证用例（测试用例）、软件验证规程（测试规程）、软件验证结果（测试结果）等软件生命周期数据，这些数据可以作为 DO-178B 附录 A 对集成过程输出的测试的 5 个目标：可执行目标代码符合高层需求、可执行目标代码对高层需求是健壮的、可执行目标代码符合低层需求、可执行目标代码对低层需求是健壮的、可执行目标代码与目标计算机兼容的达到的证据。

测试覆盖率分析

测试覆盖率分析分为：基于需求的覆盖率分析和结构覆盖率分析。

3.3.4.2.1 基于需求的测试覆盖分析

基于需求的测试覆盖分析是分析测试用例与软件需求的关系，以确认所选取的测试用例满足相应的准则的，其目标是确定基于需求的测试对软件需求实现的验证程度。基于需求的测试覆盖分析的最终结果应能表明以下结果：

1） 每条需求都有测试用例。

2） 测试用例满足正常测试和健壮性测试的准则。

当基于需求的测试覆盖分析不能表明上述结果时，应进一步分析是软件需求的不当还是测试的不充分造成的。若是需求的不当，则应修改软件需求，编制相应的测试用例，并按测试规程进行补充测试。若是测试不充分，则应补充编制相应的测试用例，并按测试规程进行测试。

对于任何等级的软件，都要求测试对高层需求的覆盖达到 100%，而只对 A、B 级软件要求测试对低层需求的覆盖达到 100%。且对于 A 级软件，测试的需求覆盖率要求独立达到。

3.3.4.2.2 结构覆盖率分析

基于需求的测试用例可能未测试到所有代码结构，因此，需进行结构覆盖率分析，及其它的验证活动，以获得结构覆盖率。

结构覆盖率分析可基于源代码进行：除非软件等级是 A 级，并且编译器产生了不能直接追踪到源代码的目标代码，此时，应对目标码进行额外的验证，确认这些不能追踪至源代码的目标代码的正确性。也就是说对 B 级及以下等级的软件，只需要实现源

代码级的测试的结构覆盖率分析并达到覆盖目标即可。而对于 A 级软件，只进行源代码级的测试的结构覆盖率分析，一般是无法达到表 3- 3 的目标 5~目标 7 的。因为现代机载嵌入式软件的主流开发语言都是高级语言，包括面向结构的古老的 C 语言和更为先进的面向对象的 C++、Phyton 等语言。而使用这些高级语言开发的软件源代码，在使用相应的高级语言编译器编译产生目标代码后，都属于 DO-178B 所指的：编译器产生了不能直接追踪到源代码的目标代码。因此，除非 A 级软件是采用最底层的汇编语言开发的，否则 A 级软件几乎都需要对目标码进行额外的验证，确认这些不能追踪至源代码的目标代码的正确性，也就是通过目标码级的测试结构覆盖率分析来达到 A 级软件测试的结构覆盖率目标。

# 研究内容与实施方案

## 3.1主要研究内容

1. 1. 对fadec系统研发与审定过程进行研究，完成文档关系，文档属性关系的研究，标注工作。为do178c审定过程中目标验证提供准备。

2. 以Do-178C对软件验证过程输出结果的验证目标为标准，在业内已有的软件测试情况的基础上，结合嵌入式、高可靠性等情况。研究达成这些符合性要求的测试方法。

## 3.2拟采取实施方案

3.2.1对适航审定过程中目标和过程的研究

DO-178C规定了A级软件在适航审定过程中必须满足的71个目标。主要通过规定航空机载软件生命周期中各个过程的目标，规定达到这些目标和工程实现考虑和规定这些目标已经实现的证据记录来指导机载软件工作者工作。本文主要研究适航审定过程中，文档之间的关系和文档与适航审定过程的对应关系和最终的审定目标之间的关系，

DO-178C虽然给了软件生命周期的过程、数据和目标要求，但是绝大部分并未具体到可以直接研制工作的水平。DO-178C没有明确定义和解释活动之间的各种关联关系，例如，先后关系、协作关系、数据的传递关系及至资源的共享关系等。审查单位必须结合自身情况制定满足标准的可以操作的、完整的、具体的软件审定流程。

适航审定指南所指出的局方四次审定介入先后次序和依赖关系如下图：



图 3 机载软件的生命周期及局方四次审定介入的先后次序和依赖关系

# 关键技术与难点

# 5.预期成果

1. 毕业设计项目技术文档书写及代码实现各一份

2. 完成硕士学位论文完成硕士学位论文。

**6.学位论文工作计划**

|  |  |  |
| --- | --- | --- |
|  |  | |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**7.主要参考文献**