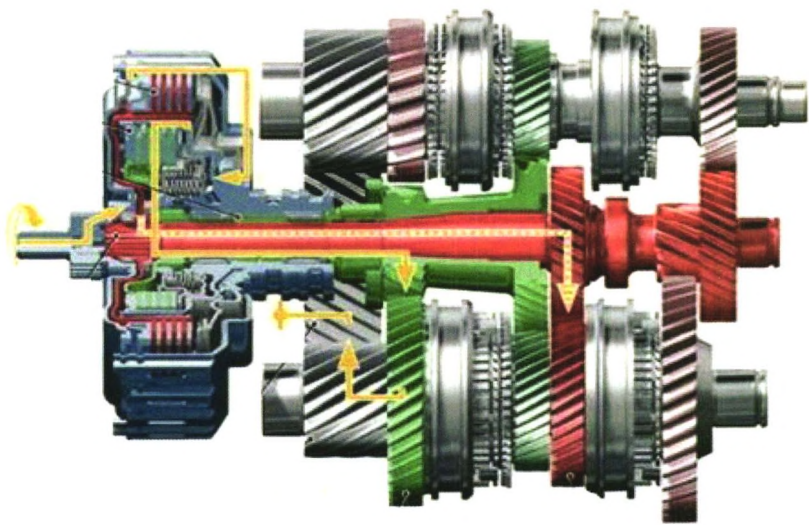


江苏驰超变速器有限公司/徐晶晶



湿式双离合变速器系统开发

双离合变速器（DCT）由两个湿式离合器、两个输入轴分别与两个湿式离合器连接、换挡同步器按奇偶数挡位分别布置在两个输入轴上及相应齿轮组、自动换挡控制系统以及电控系统TCU（Transmission Control Unit）组成。其中变速器控制系统TCU主要负责采集车辆运行信息及驾驶员的操作指令，然后进行判断并控制DCT的运行。TCU作为实现自动变速器功能的关键部件，具有内部结构复杂，开发周期长，验证困难等特点。

本文根据湿式双离合自动变速器的控制策略，研究了V-模式开发流程，在此基础上针对实际测试故障数据（拨叉挂挡EP报错）为研究对象，搭建单元测试模型，进行仿真故障复现，在模型中逐条排查故障发生原因，修订模型，通过仿真数据验证最终解决故障问题。

TCU控制单元

TCU是整个双离合变速器的核心。首先，各种传感器检测到的汽车的运行状态、行驶工况、驾驶员的意图等参数输入给TCU，经过微处理器分析处理后按照设定的换挡规律输出控制指令，并通过电液转换执行机构实现所需的控制功能。

TCU一般由微处理器和一些接口电路组成，微处理器采用16位或32位，通过CAN总线接口实现与发动机控制系统（ECU）之间的信息传递。其基本组成如图1所示，共由13个传感器和六个执行器组成。

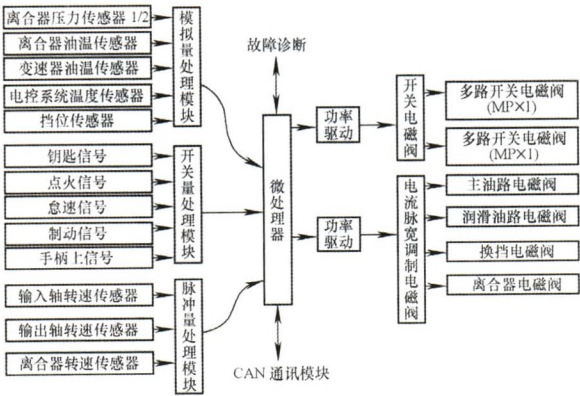


图1 TCU基本组成



各个传感器输入的信号经过相应的开关量、模拟量、频率处理模块输入到微控制器，微控制器将这些信号经过不同的算法转换成有用的变量，通过判断这些变量的数值大小采用不同的控制策略，最后根据不同的控制策略控制相应的执行机构。TCU整个工作过程是实时的，微处理器不断采集传感器信号，同时根据这些信号不断地采用对应的控制策略，改变控制机构的控制方式，保证车辆能够正常、高效地行驶。

TCU控制策略开发流程

1. 基于Despace进行V-模式开发

传统的嵌入式系统开发流程存在一系列缺点：直到台架测试，控制器才与被控对象结合，增加了解决问题的成本；问题不易单独解决，错误难以回溯；系统仿真阶段与实践阶段脱离，不能实时运行。基于这些缺点，在实际开发时，选择V-模式的开发流程包括七个阶段（见图2）：功能需求分析、功能设计与simulink仿真模型、快速原型、单元测试、硬件在环回路（HIL）测试和标定（Calibration）。该开发流程的主要特点是Despace为开发、编程和测试提供了统一的测试平台，使软件的整个开发过程得到验证，可以加速和简化开发流程，实时修订过程错误，大大提高效率和节约成本。



图2 V-模式的开发流程

2. Simulink仿真模型建立及快速原型生成

建立Simulink仿真模型的具体步骤如下：

- 1) 建立对象数字模型。
- 2) 设计控制方案。
- 3) 以Matlab为平台建立模型并进行仿真。
- 4) 配置相应的硬件I/O接口。

双离合变速器软件模型大致可以分为四层：控制动力系统的高层功能块、控制动力系统精确度的低层功能块、判别故障的安全功能块以及通讯功能块（见图3）。

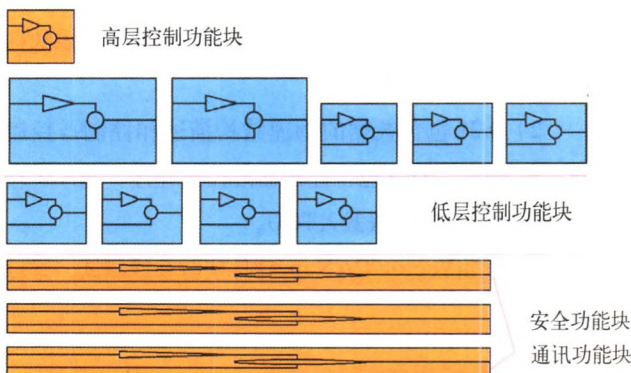


图3 仿真模型

3. 硬件在环回路（HIL）测试

使用 Despace的实时控制仿真平台作为实时环境的硬件载体，模拟整车模型。

将已生成的代码下载到微处理器。

TCU基于整车模型发出的状态信号发出对变速器模型的控制信号。

对于功能测试，可以模拟各种状态，特别是现实中难实现的极端状态，评估TCU的控制效果。

该实时控制仿真平台与TCU之间的接口如图4所示。

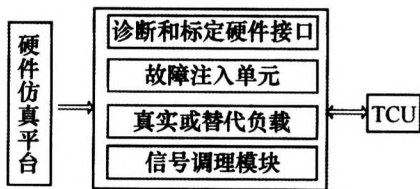


图4 仿真平台与控制器接口

4. 标定

标定流程如图5所示，具体如下：



图5 标定流程

- 1) CAN接口标定使用的为CCP (CAN Calibration Protocol) 协议, CCP的标准是基于CAN总线技术的标准通讯协议。
- 2) A2L包含数据的物理结构描述和控制器控制程序的参数; HEX包含控制器的控制程序, 将A2L和HEX文件通过CAN线载入TCU。
- 3) 利用Vector CANape软件实时在线测试, 修订标定量。

模型单元测试

单元测试的对象是模型的最小单位——模块。单元测试应对模块中所有的功能设置测试用例, 对覆盖度不足的测试用例进行补充及测试, 校核单元测试结果, 评估测试结果的正确性。通过单元测试用例, 可以保证所建模型功能达到预期的运行结果。

本文以实车测试中的故障数据(见图6)为分析对象, 找出故障原因, 复测故障内容, 修订软件模型。从图6中可以看到, 挡位挂在D挡, 在车速为3.09km/h时, TCU开始控制13拨叉挂1挡, 挂挡0.966s左右, TIRS021故障即报出, 从AR2电流被关闭后, 随即进行了三次挂1挡拨叉动作, 但仅维持了一个周期(5ms)即被关闭, 往复三次后, 挂挡失效的DFC (TIRSCUS_021) 就被激活, 其与软件需求不符(应连续尝试挂1挡三次均失败后才能激活该DFC), 要求在单元测试中复现该故障, 分析原因, 修改模型解决问题。

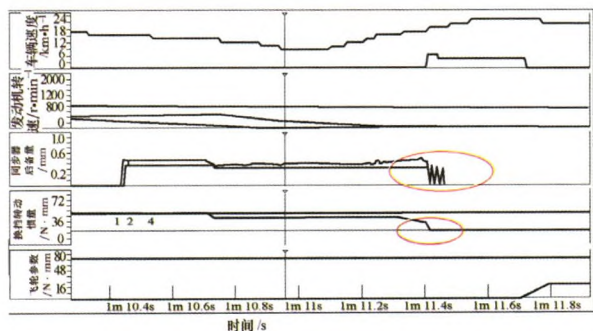


图6 拨叉挂挡EP故障数据

1. 单元测试模型建立

针对挂挡问题由软件模型中GearController模块控制, GearController模块主要控制变速器的挡位问题, 主要包括两部分: 控制拨叉动作的判断逻辑和制动执行机构的需求电流计算。

将实车CANAPE采集的信号转换为MATLAB对

应的.m文件, 单元测试模型中输入信号采用From workspace信号块, 信号块对应指定工作区数据通过MATLAB矩阵的表达式(2D数组)来判定, 一个结构包含一组信号值和时间步骤, 或一个时间序列对象。由于实车采集信号时, 有些变量并未采集到, 需要进行算法换算, 添加AuxiliaryControls功能模块辅助测试。

单元测试框架包括Signal input模块(车辆采集输入信号)、Input模块(逻辑默认输入信号)、GearController(齿轮控制器)功能模块、AuxiliaryControls(辅助控制)功能模块、输出信号模块及scope模块。

2. 功能块输入信号设置

除了车辆已采集信号可以用2D数组来定义, 余下信号可以分为两类: 一类为逻辑默认认定值量, 一类可以借助其他功能模块来计算。在本单元测试案例中, 针对GearController功能模块中信号分析, 逻辑默认认定量见附表。

信号赋值

信号	描述	值
enable_synchro	离合器互换子系统中允许同步为1	1
SIG_ODD/ EVN_CLU_LCKD	奇/偶数离合器锁定为1	0
DSYN_LIMP	当离合器被锁定或换挡电磁阀失效时为1	0
GCT_run	齿轮/离合器自学习触发时为1	0
shift_active	检测离合器是否正在交换时为1	0

AuxiliaryControls功能模块通过计算采集拨叉信号得到当前1/3、5/N、2/4、6/R挡的拨叉位置信号, 传送给GearController功能模块运行仿真。

3. 仿真结果分析

针对模型在环境文件中进行仿真分析, 根据仿真信号(见图7)可以看出问题数据中, 1挡拨叉经过大于0.8s的时间, 第一个周期检测挂到位后, 在第二个周期的检测中, 拨叉位置有所反弹, 检测挂挡不到位。

在模型中分析故障原因: 允许挂挡的时间阈值t_scr在0.8s和1.4s之间来回切换, 软件中三次误进入“挂1挡拨叉超时”, 最终报出“一档拨叉挂挡失败”的故障, 在模型中可以看出, t_scr受LIM_EVN_ACT控制, 继续排查, LIM_EVN_ACT由开关量GC_C_ENG2_ACT反复在0、1间变化, 最终查找到原因是定

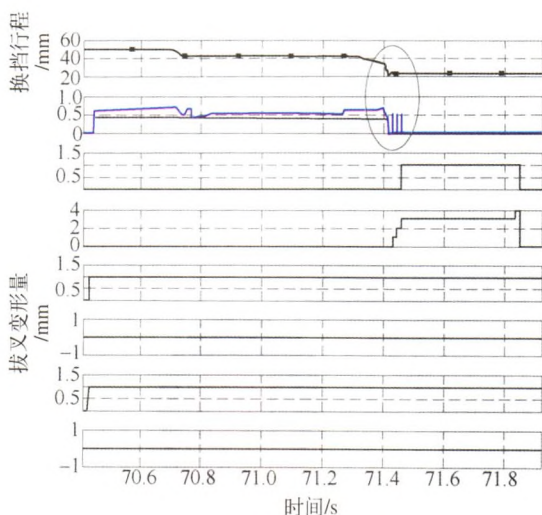


图7 故障复现结果

位于 (State Flow) shift_Fork_Current_Control模块在两个周期内关闭与激活导致。

4. 故障原因分析

运行模型, 设置 (State Flow) shift_Fork_Current_Control模块中的软件执行周期为5ms, 进行逐步分析。

经过分析, 进入mpx时, 经过一个周期delay, t_{scr} 由1.4s变为0.8s, 由于实际挂挡时间超过0.8s, 所以执行超时操作, $c1++$, 误认为是挂1挡拨叉超时故障, 继续执行Engage_Coordinator\Chart, GC_C_ENG2_FLAG=1, 导致GC_C_ENG2_ACT=1, 由于上一周期GC_C_ENG2_ACT=1, t_{scr} 由0.8s变为1.4s, 检测挂挡不超时, $c1$ 重置为0, 由于上一周期Reset_Current, 电流控制模块被关闭, 此周期设置电流后, 电流控制模块激活, 但Engage_Coordinator\Chart只进入默认原始状态, 不执行转移, GC_C_ENG2_ACT=0, 再进入MPX时 t_{scr} 由1.4s变为0.8s, 由于实际挂挡时间超过0.8s, 所以执行超时操作, $c1++$, 第二次误认为是挂1挡拨叉超时故障。以此类推, 出现三次挂一挡拨叉“假性”超时故障, 最终报出挂挡失效。

5. 故障解决

根据以上分析, 出现该故障的根本原因为拨叉判断到位两个周期内出现允许最大挂挡时间 t_{scr} 的往复变化。为了解决这一问题, 在State Flow中增加允许最大挂挡时间冻结标志位 t_{scr_frz} , 在第一次检测到拨叉到位后即将允许最大挂挡时间冻结, 不再实时更新, 正常工况下 (未到位) 通过实时更新允许最大挂挡时间满足换挡时间极限变化的需求, 在检测拨叉到位

时, 取当前周期的允许时间并置冻结标志位, 在下一个判断周期内, 通过冻结后的允许时间判断挂挡是否超时。

6. 策略修改单元测试验证

保证输入量和环境量不变, 载入新模型, 运行仿真, 结果如图8所示。

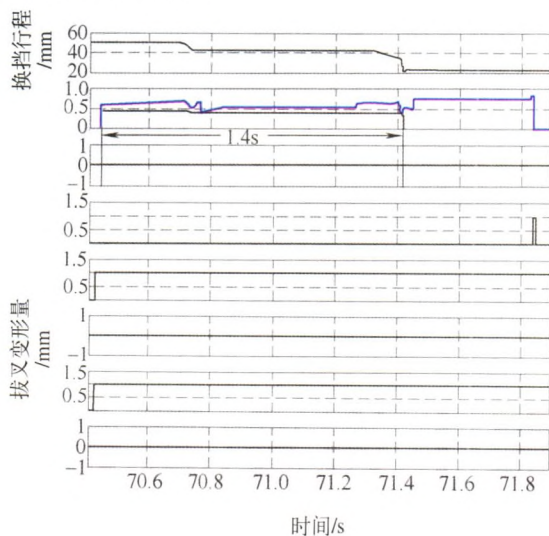


图8 单元测试结果

从图8中可以看出, 重复故障复现时的数据作为输入条件, 即13拨叉在第一个周期内到位, 第二个周期内不到位, 以后一直不到位进行模块的单元测试, 仿真结果显示, 使用与之前复现故障相同的输入条件, 允许挂挡的时间阈值一直保持1.4s, 符合软件要求, 没有误报挂1挡拨叉故障, 测试通过, 故障解决。

结语

本文具体介绍了湿式双离合变速器V-模式的开发流程, 验证了V-模式的高效性和稳定性。单元测试作为其中的一个重要环节, 针对实际具体案例阐述了单元测试的优势, 其在模型建立与仿真阶段, 通过测试对功能性能进行故障分析和原因查找, 通过实时修改模型和同环境仿真验证, 使所建模型达到需求, 并保证顺利进行下一阶段的工作。

关键词:

湿式双离合变速器 系统开发 单元测试

(收稿日期20140213)