CVT control software testing

Lionel Belmon, Liu Fei – Global Crown Technology, Beijing
Zeng Weihua - Jianglu Rongda, Changsha
议程

1. Motivation
2. Vehicle plant modeling
3. TCU embedded software
4. Simulation validation
5. TCU Testing
Motivation and objectives

Develop and test TCU control software

High quality vehicle model
Realistic simulation

Interaction road - vehicle - hydraulics - TCU

Cover large number of test cases
Vehicle plant model
Vehicle model overview

Driver
CAN
Hydraulics
Platform: SimulationX

Car
Engine/EMS
Torque converter
RND clutch
Variator Belt
Longitudinal Vehicle dynamics with left/right tyre slip

Torque factor

Engine/EMS
Driver

Engine/EMS
Driver

Engine/EMS
Driver
Hydraulic module

Main flow pressure dynamics

Spool valves characteristics

Solenoid valve - pilot pressures

Torque converter lock-up

RND clutches

Extraction of CAD parameters

Primary / Secondary pistons for variator
Hydraulic Unit tests and validation

Adjustment of solenoid characteristics

Primary cylinder pressure

```
from 动缸压力—电磁阀原特性曲线对应的仿真结果
from 动缸压力—实车测试
from 动缸压力—电磁阀修改后的特性曲线对应的仿真结果
```

- original(Simulation)
- measurements
- changed_solenoid(Simulation)

Solenoid current
Variator belt model concept

Variator ratio change by controlling primary/secondary pressure

\[ \frac{di}{dt} = K_i n_{pri} (F_{pri} - F^{*}_{pri}) \]

Actual force

steady-state force required

Max torque limit depends on pressure!

\[ P_{secmx} = \frac{T_{pri} \beta \cos \alpha}{2 \mu_{sec} R_{pri} A_{sec}} \]
Real-time simulation

Adjustment of hydraulic volumes and mechanical stiffness

Fixed-step solver – 0.5 ms

~80 ODE state variables

~200 inputs/outputs variables (CAN, sensors, actuators)

Code generation from SimulationX to FMU cosimulation

Execution in QTronic Silver

Simulated time: 98.38s; Actual time spent computing: 18.62s; Speedup factor: 5.28
TCU embedded software
TCU software generation process

1. Functional requirements
2. ETAS ASCET model
3. Embedded c-code
4. A2L and calibration data
5. ECU hardware

ETAS INCA
Virtual TCU with Silver

1. Functional requirements
2. ETAS ASCET model
3. Embedded c-code
4. A2L and calibration data
5. ETAS INCA
6. Silver virtual ECU
Virtual TCU platform

Configurable GUI

Attach to Process
Python scripts
Read/Write

Debug
Microsoft Visual Studio
Glue
MDF CSV
Tests

ETAS ASCET CVT Control Model

Code generation

Vehicle model SimulationX

C Code
Plant Model FMU

Silver

Virtual ECU

PAR DCM
Task scheduler
DBC CAN

CVT Control Software:
Tasks
Fixed-point C code
Validation of vehicle simulation
Test cases for simulation validation

~20 test drive cases to test various model characteristics:

10% pedal acceleration – coasting
30% pedal acceleration – coasting

…

Lever position changes: N-R-N-D-N-R-P-R-N…

Heavy braking with ABS

Tip-in, Tip-off

…
Example of test drive validation

- **Engine speed**: measurement - simulation
- **Car speed**: measurement - simulation
- **Turbine speed**: measurement - simulation

Hydraulic primary – secondary pressures
TCU embedded software testing
Test requirements

Usual test drive cases
  NEDC, 100km/h acceleration...

Calibration parameters
Software functions activation/deactivation

Various environment conditions:
  - slope, snow/water...

Fault monitoring and reaction
Fault insertion

- Solenoid and hyd faults
- Wheel sensor faults
- Bus faults
- Temperature sensors faults
- Speed sensors faults
Fault insertion – how it works

Example on solenoid valve fault

TCU application software

Target current

Feedback current

Vehicle model

Fault flag

TCU Basic software
Fault insertion – how it works

Example on solenoid valve fault

TCU
application software

Current target

TCU BS emulation python

Fault flag from BS

Python

Vehicle model

Fault mode selection
Short to GND
Short to UBat
Open line
Fault insertion – how it works

Example on solenoid valve fault

```python
fault_sdi_tcs1_BOB = Chooser(
    # tw must be a WeaverConnection instance
tw,
    # Sil/Hil Variable instance, or None if none
    Variable('fault_sdi_tcs1_BOB'),
    # instrument name shown by TestWeaver; if None use Variable name
    'fault_sdi_tcs1_BOB',
    # instrument unit shown by TestWeaver
    ':
    # instrument description shown by TestWeaver
    ':
    # partition definition
    # min and max, or value; name of partition; optional: severity/occurrence rate, comment
    [[-2, 'hyd0', Occurrence.FaultLow, 'hyd0'],
    [-1, 'hyd1', Occurrence.FaultLow, 'hyd1'],
    [0, 'no', Occurrence.OK, '0'],
    [2**1, 'SG', Occurrence.FaultLow, 'bit1'],
    [2**2, 'SB', Occurrence.FaultLow, 'bit2'],
    [2**3, 'OP', Occurrence.FaultLow, 'bit3'],
    [2**6, 'UBATT', Occurrence.FaultLow, 'bit6']],
    # type of the instrument: Default, Real, Integer, String
    Type.Integer
)
```
Use of QTronic TestWeaver

- Formal requirements monitoring
- Hand written test scripts
- Automatic test cases generation
- Automatic test report generation
ABS simulation example

Clutch locked + Strong braking → Wheel block → **strong engine deceleration**
→ Transmission torque exceeded → belt Slip!

→ Belt slips are reported automatically in TestWeaver reports
QTronic TestWeaver automatically generates drive sequences

Example of Belt Slip found by TestWeaver
MC/DC coverage
Vehicle model + TCU + calibration:

validated simulation, reproduce very well the actual execution of TCU in the real vehicle

Fault insertion and validation of TCU fault management

Automatic generation and evaluation of test cases
TestWeaver + requirements modeling

Very efficient support for TCU development and testing!
谢谢！
Thank you！