Case Study:
Autonomous Vehicle Sensor Module Software Testing

Integration of VectorCAST with Simulink

Speaker: Jeff Lovell
Senior Systems Engineer, Pi Innovo LLC
Abstract

- Pi Innovo was responsible for development of custom hardware and software, for use as a sensor module within an autonomous passenger car.
- This session presents Pi Innovo’s activity as a case-study of modern application software testing, which included the use of a combined Simulink and VectorCAST environment for software unit testing. This environment enabled re-use of familiar methods and test-scripts, while adding capability to quantify new code-coverage metrics after Simulink autocode generation.
Presentation outline

- **System overview**
- Pi Innovo responsibilities
- Pi Innovo development process
- Purpose and usage of VectorCAST
- Results
Autonomous passenger car

- This software application is part of a system within an autonomous passenger car.
High-integrity sensor module

- The ECU being developed is a sensor module, responsible for reporting data to an autonomy computing platform.
- The required behaviors included:
  - Measurement and processing of high-resolution wheel position encoders.
  - Data timestamping based on the vehicle’s master clock module.
  - Data transmission using 100BASE-T1 automotive ethernet (2-wire).
  - Fail-operational robustness to single-point failures (internal redundancy).
Presentation outline

- System overview
- **Pi Innovo responsibilities**
- Pi Innovo development process
- Purpose and usage of VectorCAST
- Results
Pi Innovo responsibilities

- The OEM assigned Pi Innovo the Tier 1 responsibilities of development and supply of the custom ECU.
- Pi Innovo performs the design and development activities in-house.
- Special tests are outsourced to test houses.
- ECU manufacture is performed on-contract by partner companies.
Pi Innovo responsibilities

- Custom ECU hardware development
  - New PCB & enclosure
  - DV-PV
- Custom ECU firmware development
  - (New microprocessor type)
  - OS (ported from OpenECU)
  - Hardware drivers
  - Software API
  - V&V
- Custom ECU software development
  - Application logic
  - V&V
Pi Innovo responsibilities

- Custom ECU hardware development
  - New PCB & enclosure
  - DV-PV
- Custom ECU firmware development
  - (New microprocessor type)
  - OS (ported from OpenECU)
  - Hardware drivers
  - Software API
  - V&V
- Custom ECU software development
  - Application logic
  - V&V

VectorCAST test cases for C language

VectorCAST coverage analysis for Simulink (this presentation)
Go-fast timing

- Functional A-samples delivered to customer 11 months after design kickoff (no prior design existed).
- Rapid development requires maximum re-use of existing products and techniques, even for this ‘from-scratch’ ECU.
  - Modular circuit designs (Pi OpenECU)
  - Hardware development environment
  - Modular software designs (Pi OpenECU)
  - Software development environment
  - Test tools
  - QA process (Pi BMS)
Presentation outline

- System overview
- Pi Innovo responsibilities
- **Pi Innovo development process**
- Purpose and usage of VectorCAST
- Results
Pi Innovo’s Business Management System (BMS) is based around the requirements of ISO9001:2015 and ISO15504 Automotive SPICE and provides a full suite of operational and project-level processes serving the Organization as a whole, Quality Assurance, Project/Program Management and Engineering departments.
Model-based application software development (traditional vehicle systems)
Presentation outline

- System overview
- Pi Innovo responsibilities
- Pi Innovo development process
- **Purpose and usage of VectorCAST**
- Results
Additional V&V required for high-integrity ECU software

- This product required formal measurements of code coverage during software testing.
  - Branch coverage
  - Statement coverage
  - Modified condition / decision coverage (MCDC)

- **GAP:** these concepts apply to the C-language source; not our model-based source.
  - We therefore require a method to **monitor code coverage of the autogenerated C program**, during our unit tests.
Model-based application software development (high-integrity applications)

- Requirements
  - Software units (Simulink)
  - Code generation (C)
  - Compiler output (.s37)
  - Simulink simulation (SIL)
  - On-target execution

- System Validation (on-vehicle)
  - HIL / BB tests
  - Unit tests
    - VectorCAST

Copyright © 2019
Model-based application software development (Solution)

Configured Simulink simulation to SIL-mode (generates C and compiles for execution on x86 PC)

Unit tests (scripts) for Simulink simulation do not change. (Pi Innovo scripting environment using Excel & Simulink)

VectorCAST C++ (for C) instruments the auto-generated C, and records coverage metrics while monitoring its execution.
Tool qualification

- VectorCAST is a software tool the industry recognizes from prior experience, and approves for high-integrity development.
- (Tool is ‘pre-approved’ by major OEM’s - saves project time vs new tool cert.)
Presentation outline

- System overview
- Pi Innovo responsibilities
- Pi Innovo development process
- Purpose and usage of VectorCAST
- Results
Results example

- Simulink controls the software execution, VectorCAST monitors the coverage.
- Coverage reports (shown here) are saved as evidence of the test result.
Autogenerated C (from Simulink)
Lines of code are reported as covered by the test...

...or by human analysis.

(Necessary when impossible cases are present in autogenerated C)
Impact

- This approach enabled re-use of a familiar testing method (Simulink simulation), while adding the required code coverage measurements.
- Re-use of test scripts and simulation wrappers created substantial time-savings, and enabled accelerated delivery of product to the customer.
Pi Innovo LLC is an expert partner for the design and development of innovative electronics systems to the automotive, transportation, defense, industrial, and aviation industries. Our uniquely adaptable business engagements, based on Pi Team services and OpenECU products, enjoy a strong reputation for delivering results of the highest quality, providing outstanding value for our customers.

OpenECU is a wide range of adaptable, field-ready products and intellectual property designed to accelerate electronics system development. The philosophy behind OpenECU is the creation of modular, reusable technology that is implemented to volume production standards and is fully “open” to custom configuration and further development.