2016 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM

POWER & MOBILITY (P&M) TECHNICAL SESSION
AUGUST 2-4, 2016 - NOVI, MICHIGAN

Developing a brake based stability control system to satisfy military and federal safety requirements for wheeled, light tactical vehicles.

Wendell Tackett
AM General
Livonia, MI

Mark Clark
Pi Innovo
Plymouth, MI

ABSTRACT

This paper provides detail of the system architecture and systems engineering process utilized by AM General to develop a new stability control system that satisfies all military and federal safety requirements for wheeled, light tactical vehicles.

Introduction

Since the introduction of brake-based Electronic Stability Control (ESC) systems on passenger cars in the mid-1990's, the added safety benefits to the driver have been pronounced. Based on NHTSA estimates, the addition of ESC reduces single vehicle SUV crashes by 59% and SUV rollovers by 84%[1] This prompted the US government to mandate ESC on all passenger cars and light trucks under 10,000 pounds GVWR as of the 2012 model year.

Wheeled military vehicles would expect to see similar benefits from the implementation of ESC, however several technical challenges have contributed to a delay in the introduction of COTS products on many platforms. Firstly, COTS products do not meet the following military requirements:

1. Deep water fording
2. Electromagnetic interference and susceptibility
3. Compatibility with DoT 5, silicone brake fluid
4. Functionality to -50F

Secondly, the performance of COTS products falls short of military expectations, simply because they are optimized for vehicles with greatly different usage profiles. The primary focus of COTS software is performance on roadways, since that is their typical customer application and all federal validation testing is on paved roads. Military applications have more extreme off-road mobility requirements, survivability requirements, much higher cargo loading ranges, higher gross vehicle weights and higher centers of gravity. COTS manufacturers will not adapt their software to satisfy military off-road applications, due to the relatively low production volumes.

In an effort to satisfy customer requirements and to provide a system with optimal safety and performance, AM General developed a completely new, purpose-built stability control system for military applications.

Project Scope and Definition

The standard building blocks of a vehicle stability control system are: anti-lock brakes (ABS), traction control system (TCS), electronic brake-force distribution (EBD – dynamic management of rear brake pressure in all brake applications) and electronic stability control (ESC). To capitalize on the flexibility of internally developing a custom-built system, AM General included the following features to the scope of the program, which are not commonly found in COTS products: automatic brake modulation (ABM - a function that maintains constant brake pressure across an axle for extreme off road mobility enhancement), rollover mitigation (ROM - an added functionality to ESC particularly suited to up-armored, off-road vehicles with a high center of gravity), enhanced off-road detection and a "flight recorder". These features were added to the scope to meet the unique needs of tactical wheel vehicles, something a COTS brake system could not achieve. This unique feature list mandated a new software architecture. Additionally, it was clear that a unique approach to the hydraulic, mechanical and electrical architectures would be required in order to meet the aforementioned customer requirements (i.e. fording, EMI, fluid compatibility and low temperature performance) in addition to easily accommodating future upgrades, such as autonomous missions and brake energy regeneration. To

[1] Federal Register/Vol. 72, No. 66, p. 17236

coordinate this monumental product development task, a systems engineering approach was adopted from the outset. Additionally, this development program was used as a benchmark to acquire CMMI Level 3 appraisal status, which further enhanced adherence to the SE processes put in place.

Systems Engineering Structure

An Integrated Project Team (IPT) composed of key individuals from each division of the organization was formed to achieve this task, meeting on a weekly basis. Top level endorsement from upper management and diligent reporting of meeting minutes ensured adherence to the processes put in place. Key processes to control the technical development and project management as defined by INCOSE were strictly followed. Where applicable and deemed logical, forms and processes from a standard automotive APQP (Advanced Product Quality Plan) were introduced.

The team started by gathering stakeholder requirements, regulatory requirements, and voice of the customer data to develop a set of top level requirements to guide the program. These requirements were reviewed and challenged by all stakeholders prior to being formally issued to the product team for execution. By analyzing a number of military and regulatory specifications (e.g. TOP 2-2-608, FMVSS 105, 121, and 126), and a collaborative effort between hydraulic and brake control experts, a high level system requirement document was generated. This document served as the highest level of definition for what the system would do and provided constraints to assist in hardware and software design. Features like anti-lock braking (ABS), traction control (TCS), and electronic stability control (ESC) are described in high level terms in order to communicate broad functionality.

Once the product requirements were established the team set about developing a candidate architecture for the brake system that would meet performance, cost, reliability, and other requirements from the product specification. Several iterations and architectures were evaluated in a methodical decision matrix until and prime architecture was selected. During this architecture analysis the team also addressed functional safety and analyzed the proposed architectures for hazards and using DFMEA and FTA techniques iterated the architecture to mitigate the identified hazards.

The team decomposed the architecture into hydraulic, electromechanical, software, and electronics domains by attributing requirements to the various components in the proposed architecture.

Using the items discovered in the aforementioned studies, Add on features like ABM –and a flight recorder are also depicted. For the hydraulic hardware, design elements like brake actuation components and the brake hydraulic circuit within a manifold were created. An off the shelf ECU was identified early in the project and the system requirements defined all the input sensors, signals, and driver circuits utilized for a brake based stability control system. High level requirements such as FMVSS 126 and J1939 compatibility drove software content and design.

As part of a system development approach, the issue of traceability becomes a high level of concern. The system requirements document employed a ‘tagging’ convention where each requirement would have a unique identifier that is used as a reference in other work products throughout the project. This tagging method allowed high level requirements to be traced down the chain to ensure that there is 100% implementation and test coverage. The systems requirement document became the highest level document that all the lower level work products referenced. Once the system was appropriately ‘scoped’ via the systems requirements, the mechanical, electrical, hydraulic and software architectures were developed via independent paths.

Hydraulic, Electrical and Mechanical Development

Some key challenges behind the hydraulic, electrical and mechanical development were not only the technical requirements driven by the customer which COTS products cannot satisfy, but also the fact that a hydraulic solution for a vehicle in the target gvw range did not exist. As mentioned earlier, current government regulations only mandate ESC for vehicles less than 10,000 pounds. Hydraulic stability solutions for vehicles closer to 20,000 pounds were not readily available. A full system architecture (including brake actuation, stability control and foundation brakes) was first developed to satisfy all federal regulations, with special emphasis on increased survivability. For example, rather than having a single power booster, a redundant power booster was implemented. And in case both power boosters were disabled, a tertiary non-powered back up braking system was in place that would stop the vehicle within federally mandated stopping distances. In order to develop a unique system for a small volume application, yet maintain high levels of reliability, a partnership with an established hydraulic supplier was formed. The experts at Hydac USA were called on to fill that need. Off the shelf components were used where possible, and when needed, custom designs were developed, tested and implemented.

Software Development and Implementation

Challenges regarding software development not only involved creation of a brake based stability algorithm from the ground up, but also producing a safety system that would always return the brake system to a safe state in the event of a stability system failure. All of the software behavior was initially designed in a software architecture document based on the system requirements and hazard studies previously mentioned. This document captured some detail of all sub-functions or behaviors and grouped them into a coherent software architecture. The first identified sub-function was how information would be input to the system. The goal of the input feature is to sample the low level voltages and CAN information, bring them into the application software, and organize the information so other sub-functions can access the information. The architecture continues by describing how the raw input information is diagnosed for faults, how raw information is converted to engineering units, further system level diagnosis, how calculated signals like vehicle speed or driver requested torque occur, how desired slip regulation occurs, and how brake manifold valve control commands or CAN transmits occur. All of this information compiled into one document provided a high level description of the software design that could be used as a guide for lower level requirements development. Also, the architecture document employed the same tagging convention described above and referenced a system requirement by each architecture requirement. Using this method justifies the need for a given architecture requirement as well as proof system requirement coverage or consideration exists within the software design itself.

Wrapping up the software definition and behavior involved creation of the list of features or sub-functions defined in the architecture document. This list provided a convenient capture of the software totality and was used as a quick reference guide to track feature development status. For each feature, a detailed functional requirements document was created that served as a guide for software implementation. The functional requirements were drafted with consideration/reference to the higher level architecture and system requirements.

Once the definition and design phase was nearing completion, algorithm development took off using the powerful Matlab/Simulink model based language. Use of this tool allows for rapid prototyping of control concepts and thoughts and the ability to perform software-in-the-loop (SIL) simulations to scrutinize any idea. Use of this prototype testing also served to verify the software architecture and refine the functional requirements.

For each feature functional requirements document, a Simulink library was created. A library is a block within the Simulink language that can be used as a modular collection of software functions that fits within a larger model. Each library is initially created to receive information, relevant information to perform calculations and decisions, from an architecture wide bus. A Simulink bus is a grouping of various signals that are essentially ‘packed’ into a singular transmission line that is passed throughout the software model. The single transmission line can be connected to multiple locations in the software model and all signals within the bus are available for use. Utilizing a Simulink bus structure greatly reduces signal transmission complexity and allows for a convenient, functional testing setup that can be used during white box testing. This level of testing is made possible through use of a software wrapper that basically ‘wraps’ around each Simulink feature library and passes signal information into and out of the sub-function under test. The software elements within the wrapper are able to be modified manually or through use of an automated script. The objective of the white box testing is to verify every requirement listed in the functional requirements is tested. The advantage of an automated script is that once generated, a script can be executed each time a software change has been made with confidence that the current test execution is 100% consistent to previous rounds of testing. Thus if any error is discovered in the current test, debugging and solution discover occurs early at this stage of the development cycle prior to any system integration testing or even software integration.

Software Integration and SIL Testing

Concurrent to functional requirement development, implementation, and feature unit testing, a vehicle model was created to be able to simulate vehicle dynamic behavior in a software environment. This vehicle model was generated through physics based model development of vehicle characteristics like the drive train, tire forces, body motion, brake hydraulic performance, and the sensors used for the stability control system. The vehicle model was created with targeted vehicle characteristics but was never correlated to real world behavior. This ‘medium’ fidelity model provided an opportunity to perform software-in-the-loop testing with the integrated (all features identified in the architecture document) software and be able to debug integration issues prior to target deployment. This level of testing proved very beneficial as the interdependencies between the features as well as the inputs to the system were verified through several iterations of changes and improvements. Use of the vehicle plant model and SIL testing allowed for further feature development and some ‘coarse’ calibration.

HIL Testing

Once an acceptable level of software performance was achieved through SIL testing, the next phase of development involved actual hardware testing. As stated previously, the target ECU used for the stability control system was a COTS developer ECU with small modifications to accommodate the unique characteristics of the system. By utilizing a bench top simulator that could receive and generate ECU compatible signals and through developing a load box that represented the brake hydraulic valve loads, the integrated control software could be tested on the target ECU against the generated vehicle model with manual or automated stimulus. Driver input scripts were created that communicated with the bench top simulator and would generate input signals to the ECU that were typical of stability control events. The control software would then receive these inputs and execute its routine to accomplish the tasks it is programmed to perform. A data logger was connected to this HIL system to acquire the necessary information to understand control software decisions. This level of testing allowed for further software integration verification as well as system interaction testing as the inputs and outputs of the stability control system were simulated by the bench top simulator and load box. Furthermore, this testing verified software behavior on actual hardware and confirmed performance metrics like code throughput and memory usage were within acceptable limits.

An additional hardware piece was added to the HIL setup after successful simulations w/ the bench simulator and target ECU. Replacing the hydraulic valve load box was the actual hydraulic manifold setup. The manifold was connected to the target vehicle brake equipment (brake pedal, master cylinder, and brake calipers) creating a test rig representative of the target vehicle. The test rig was mated to the bench simulator and the target ECU was connected to the manifold through a series of harnesses. The same simulations available in the SIL environment were made available in this HIL environment providing a test system that could simulate stability control events and have the control software execute brake commands that would occur in real world events. This level of testing was the final phase of developing and verifying the stability control system was ready for deployment to the target vehicle.

Vehicle Testing

The system performance was further refined through vehicle level testing at various test facilities throughout the US. The system was deployed on multiple vehicles including AM General's ECV2 and the Sustainment HMMWV. Much test track time was given to finalize a production ready vehicle level performance for all driving scenarios including asphalt emergency maneuvers, winter driving conditions, as well as off road events frequently seen by these vehicles. The majority of vehicle testing involved system calibration however when software changes were warranted, a change was quickly (at times, even the same hour) made available through use of the software tool chain utilized on this project.

System diagnostic detection and remedial mode actions were also refined during vehicle testing. A full suite of fault insertion tests were run mostly in a garage or laboratory setting to verify appropriate response. Then, when applicable, the reduced system functionality was verified through relevant test track events to ensure vehicle response always returned to the safest possible state.

Finally, the full system performance was verified through vehicle durability testing. As stated previously, the system was tested on multiple vehicles over the course of a few years and thousands of durability miles. The greater part of the durability testing involved harsh, off road style events typical of a light, tactical vehicle's life cycle.

The following data traces demonstrate the effectiveness of the developed control system as discovered by use of the HIL simulation with a comparison to actual vehicle testing on a test track. The traces below reveal the ability to match actual vehicle performance in simulation. All the data plots show individual wheel speeds.

Figures 1 and 2 show a HIL generated ABS event and the actual vehicle ABS event respectively. The ABS event is performed on an ice surface where initial wheel slip at brake onset is large followed by smaller subsequent wheel slips as the system attempts to provide maximum braking while maintaining a desirable level of wheel slip. Figures 3 and 4 illustrate a HIL generated and actual vehicle TCS event. The TCS event is also performed on an ice surface where initial wheel slip at throttle onset is large followed by engine and brake intervention that limit the amount of subsequent wheel slip. Later in the acceleration event, the wheels no longer slip which was a characteristic of the target vehicle once the vehicle gained more speed. Figures 5 and 6 display an ESC under steer event. The ESC event is performed on packed snow where a constant, linear steering input causes the vehicle to understeer. The system attempts to provide more vehicle turning by cutting engine torque and regulating brake pressure on the inside rear wheel, thus producing a desirable turning moment.

Conclusion

The achievement of the above described system development approach resulted in a successful completion of the FMVSS 126 stability control event on the target vehicle within 14 months of project inception. Additionally, this system provided increased mobility for the target vehicle while reducing driver workload. Starting from scratch, this project appropriately scoped the system and defined its requirements and behaviors. A team of experienced engineers were given freedom to develop strategies based on design requirements in a SIL simulation environment that provided opportunities of experimentation early in the development cycle prior to any hardware deployment. The HIL test rig and developed vehicle plant model allowed for control software integration testing that led to a successful vehicle deployment with minimal interface issues. The tagging convention utilized throughout the project work products allow for traceability to verify each developed requirement has been tested and verified. This process resulted in less debugging time, resulting in more effective and value added time spent tuning the vehicle performance.

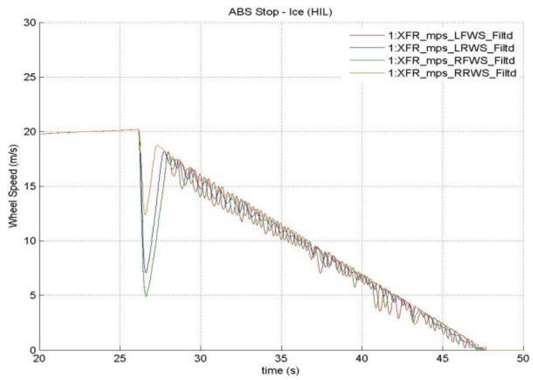

Figure 1: HIL trace of an ABS event on ice

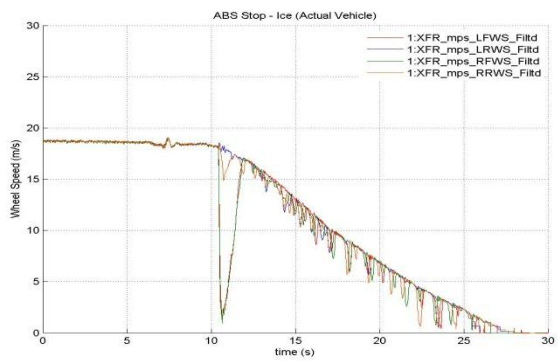

Figure 2: Actual vehicle trace of an ABS event on ice

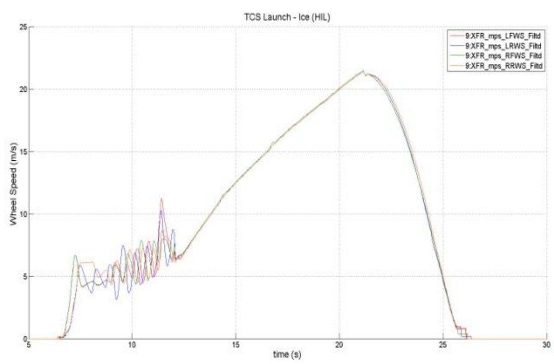

Figure 3: HIL trace of a TCS event on ice

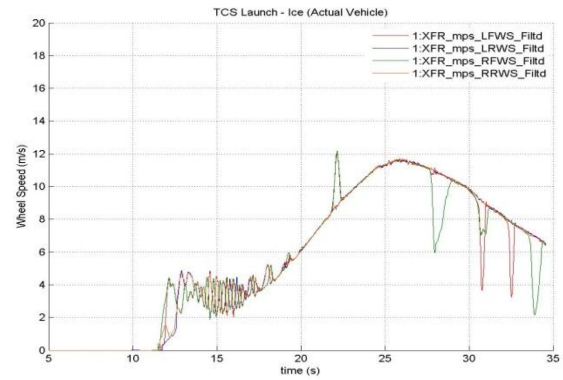

Figure 4: Actual vehicle trace of a TCS event on ice

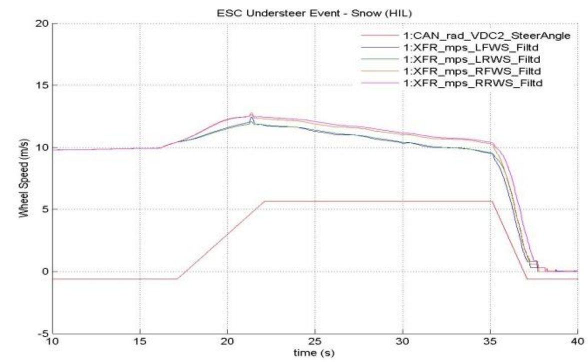

Figure 5: HIL trace of an ESC event on snow

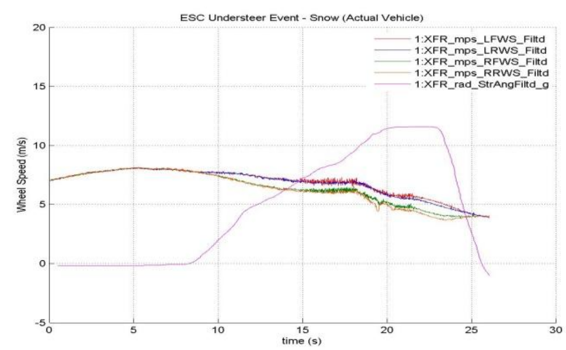

Figure 6: Actual vehicle trace of an ESC event on snow