Managing Design Changes using Safety-Guided Design for a Safety Critical Automotive System

by

John Sgueglia

B.S. Electrical Engineering
Rochester Institute of Technology, 2000

SUBMITTED TO THE SYSTEM DESIGN AND MANAGEMENT PROGRAM IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN ENGINEERING AND MANAGEMENT
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

JUNE 2015

Signature of Author_____
John Sgueglia
System Design and Management Program
May 18, 2015

Certified by_____
Nancy Leveson
Professor of Aeronautics and Astronautics and Engineering Systems
Thesis Co-supervisor

Certified by_____
John Thomas
Research Engineer, Department of Aeronautics and Astronautics
Thesis Co-supervisor

Accepted by_____
Patrick Hale
Director System Design and Management Program

Page intentionally left blank.

Managing Design Changes using Safety-Guided Design for a Safety Critical Automotive System

by

John Sgueglia

Submitted to the System Design and Management Program on
May 18, 2015 in partial fulfillment of the requirements for the
Degree of Master of Science in Engineering and Management

ABSTRACT

The use of software to control automotive safety critical functions, such as throttle, braking and steering has been increasing. The automotive industry has a need for safety analysis methods and design processes to ensure these systems function safely. Many current recommendations still focus on traditional methods, which worked well for electro-mechanical designs but are not adequate for software intensive complex systems. System Theoretic Accident Model and Process (STAMP) and the associated System Theoretic Process Analysis (STPA) method have been found to identify hazards for complex systems and can be effective earlier in the design process than current automotive techniques. The design of a complex safety-critical system will require many decisions that can potentially impact the system's safety. A safety analysis should be performed on the new design to understand any potential safety issues. Methods that can help identify where and how the change impacts the analysis would be a useful tool for designers and managers. This could reduce the amount of time needed to evaluate changes and to ensure the safety goals of the system are met.

This thesis demonstrates managing design changes for the safety-guided design of an automotive safety-critical shift-by-wire system. The current safety related analysis methods and standards common to the automotive industry and the system engineering methods and research in the use of requirements traceability for impact analysis in engineering change management was reviewed. A procedure was proposed to identify the impact of design changes to the safety analysis performed with STPA. Suggested guidelines were proposed to identify the impact of the change on the safety analysis performed with STPA. It was shown how the impact of the design changes were incorporated into the STPA results to ensure safety constraints are managed with respect to these changes to maintain the safety controls of the system throughout the design process. Finally the feasibility of the procedure was demonstrated through the integration of the procedure with requirements traceability based on system engineering practices.

Thesis co-supervisor: Nancy Leveson
Title: Professor of Aeronautics and Astronautics and Engineering Systems

Thesis co-supervisor: John Thomas
Title: Research Engineer, Department of Aeronautics and Astronautics

Page intentionally left blank.

ACKNOWLEDGMENTS

I would like to thank Professor Nancy Leveson for inviting me to work in her research group and teaching me new ways to approach engineering the systems that make up our world. I would also like to thank Dr. John Thomas for all the guidance and support he gave me while I was learning how to use STAMP and STPA.

A special thanks to my wife Amy, it would have not been possible for me to accomplish what I have without all her support. To my two sons, Alex and Chris, they make everything I do worthwhile and everything I do is for them. To my parents, I thank them for always supporting my decisions and challenging me to do better.

Finally, many thanks for the support from the SDM staff, my SDM cohort, and the other researchers I had the pleasure to work with in the Systems Engineering Research Lab. They all helped to enrich my time at MIT and make it more enjoyable.

Page intentionally left blank.

Table of Contents

List of Figures

List of Tables

This page intentionally left blank.

1. Introduction

1.1 Motivation

The use of software to control automotive safety critical functions, such as throttle, braking and steering has been increasing. The demand for more features, better fuel economy with less emissions, and increased safety require the precise control of vehicle control systems that these computer-controlled systems enable. The automotive industry has a need for safety analysis methods and design processes to ensure these systems function safely. Many current recommendations still focus on traditional methods, which worked well for electro-mechanical designs but are not adequate for software intensive complex systems consisting of many interactions between components [1]. Safety is an emergent property of a complex system and can only be properly analyzed using a top-down system thinking approach. System Theoretic Accident Model and Process (STAMP) and the associated System Theoretic Process Analysis (STPA) method [1] have been found to identify hazards for complex systems and can be effective earlier in the design process than current automotive techniques.

Performing the hazard analysis and design tasks independently, especially when the confirmation is left until the design is complete, can lead to design rework late in a project when it is most costly to correct, both in effort and money. A safety-guided design process [1] that uses STPA to integrate analysis and design can help reduce

the occurrence of late rework by building safety into the design through continual analysis with STPA.

As the system is analyzed, violations to safety constraints are very likely to be found and control reestablished through modifying the design. As design changes are made, the impact to the safety analysis needs to be systematically verified to ensure safety controls in other areas are not compromised. While implementing a safety-guided design process can ensure safety is not overlooked during design, the process of updating the hazard analysis for each design change must be efficiently managed. A method to incorporate design changes and minimize the amount of analysis needed relative to the change would increase the efficiency of the overall process and help engineers concentrate their resources.

1.2 Automotive Safety

The automotive industry is regulated in terms of the crash safety of a vehicle, fuel economy, emissions, various other standards regarding proper labeling, and the reporting and handling of safety defects. As software becomes an integral part of the functioning of automotive systems, proper design and verification methods are crucial to ensure safe operation.

In the past, most automotive systems were primarily electromechanical and the systems were typically coupled mechanically. The mechanical system was available as a backup in case the electronic system malfunctioned. For example, an antilock

braking system uses a computer to increase braking performance under certain operating conditions, but if the main controller were to fail the hydraulic system used to operate the brakes would still function. With electric power-assisted steering, when the assist system fails the vehicle can still be steered through the mechanical link from the steering wheel to the tires. These mechanical systems have become more and more reliable over the one hundred years of automobile development and the designs are well understood and documented.

Although these systems function well and have proven to be safe and reliable, there is a trend to replace mechanical systems with electrical systems utilizing software to control the primary functions. These systems are typically known as 'X-by-wire' systems. Where the 'X' stands for any of the main control systems of an automobile: throttle, gear selection, braking, and steering. These systems provide the necessary control required for better fuel economy and less emissions, driver-assisted safety features, and possibly some day fully autonomous driving.

With no common electronic system safety development standards in place and as more safety-critical systems rely on software and electronics, the automotive industry began investigating the proper development procedures for safety critical-electronic systems. With this motivation the automotive industry created the ISO 26262 Functional safety standard that automakers have adopted as the de-facto safety standard for the industry. ISO26262 covers the product lifecycle for safety-critical automotive electric/electronic systems. It does not specify how a system

shall be designed, what analysis methods to use, or a probability the system should be confirmed as having. Instead the standard prescribes how the development process should be managed, what important steps in the process are required, such as, the identification of safety goals, the performance of a hazard analysis, and what documents are required to be compliant with the standard.

1.3 The Design of Safety Systems

As Leveson explains, the systems being built today have stretched the limits of current safety engineering practices [1]. When used for safety-critical systems, they must not only be more reliable with respect to random hardware failures but they must be designed to always function in a safe manner. Although reliability analysis can help increase the reliability of a system, it does not guarantee the system is safe. Reliable systems are not necessarily safe, and safe systems are not necessarily reliable [1].

Since development activities are fundamentally iterative, where design decisions and assumptions must be continually updated as more information is found, there is a need to manage the safety analysis process. Changes made must be analyzed for their impact on the safety analysis. This may lead to whole parts of the analysis requiring re-work or just new requirements being formed. Typically the amount of work needed to redo the entire analysis is usually prohibitive. Most system safety standards require changes be managed and traceable for the design to be compliant.

The design of a complex safety-critical system will require many decisions that can potentially impact the system's safety. A safety analysis should be performed on the new design to understand any potential safety issues. Methods that can help identify where and how the change impacts the analysis would be a useful tool for designers and managers. This could reduce the amount of time needed to evaluate changes and to ensure the safety goals of the system are met. System engineering processes for change management and related standards could be used to track these changes, but these methods are primarily concerned with changes made after the design is considered complete. Formal change management systems define the process for tracking and approving design changes made during this later phase of the project. Design is an iterative process and will involve many decisions during the process requiring analysis. The use of a formal change management system during this part of the design process may be too burdensome for engineers. Using safety guided design, the design process and safety analysis process are "tightly intertwined" resulting in safety being built into the design [1] and can help to avoid some of this late rework due to safety concerns.

1.4. Objectives and Approach

This thesis will demonstrate the use of safety-guided design on an automotive safety-critical system. The impact of design decisions to the safety analysis performed with STPA will be analyzed. Guidelines to help identify the potential impact of a design decision on the safety analysis performed with STPA will be

suggested. A procedure for managing design decisions during the safety-guided design process will be proposed.

The primary research objective will be to analyze how design decisions impact the STPA results. The information can be used to manage design decisions made during the iterations of the safety-guided design process.

To achieve this objective, the following approach is used in this thesis:

1. Review the safety design methods and standards within the automotive industry and the system engineering process for change management.
2. Identify the impact of design changes on STPA by analyzing how they could affect the hazards and safety control structure.
3. Demonstrate a procedure to manage the design decisions using an automotive safety-critical system.
4. Integrate the procedure to manage the design changes in STPA with recommended system engineering practices for requirements traceability.
5. Demonstrate managing design changes using requirements traceability with the STPA results for an automotive safety-critical system.

2. Background

2.1 Safety Critical Automotive Electronic Systems

Increasing demand for vehicles with more features, better fuel efficiency with less emissions, and higher levels of safety have driven the automotive industry to use more software-intensive electronic systems. X-by-wire systems, which replace traditional mechanical linkages with an electronic controller and actuator, are becoming increasingly common for throttle control, transmission range selection, steering and braking. Miller indicates, "… by themselves x-by-wire systems do not and will not sell cars" but they provide the necessary control needed for increased fuel economy requirements and performance of active safety systems, and allow for reductions in weight and packaging [2]. Active safety systems provide features such as, crash avoidance, lane keeping assist, and semi-automated highway driving like Tesla's Autopilot [3]. All these features taken together build the foundation for the integration of more advanced sensors with vehicle-to-vehicle and vehicle-to-infrastructure communication enabling fully automated vehicles.

2.2 Functional Safety in the Automotive Industry

The integration of safety focused methods and tools in the development process for software-intensive safety critical systems has become an important goal in the automotive industry. In the past, automotive companies relied on internal processes coupled with system safety standards from other industries [4]. With the lack of an automotive specific standard, many companies relied on IEC 61508

Functional safety of electrical/electronic/programmable electronic safety-related systems [5] as the basis to guide development efforts [6]. The ISO 26262 Road Vehicles – Functional Safety standard was developed to provide a common automotive tailored safety development process for electronically controlled safety-critical systems for the industry.

ISO 26262 is an automotive functional safety standard for electric/electronic systems adapted from IEC 61508 for the entire safety lifecycle [7]. The introduction to the standard states, "ISO 26262:

a) provides an automotive safety lifecycle (management, development, production, operation, service, decommissioning) and supports tailoring the necessary activities during these lifecycle phases;
b) provides an automotive-specific risk-based approach to determine integrity levels [Automotive Safety Integrity Levels (ASIL)];
c) uses ASILs to specify applicable requirements of ISO 26262 so as to avoid unreasonable residual risk;
d) provides requirements for validation and confirmation measures to ensure a sufficient and acceptable level of safety being achieved; provides requirements for relations with suppliers." [7]

The reference development process considered for the standard is based on the "V" model, which is popular within the automotive industry. Figure 1 gives an overview of ISO26262 within the "V" model including the roles of management and supporting processes.

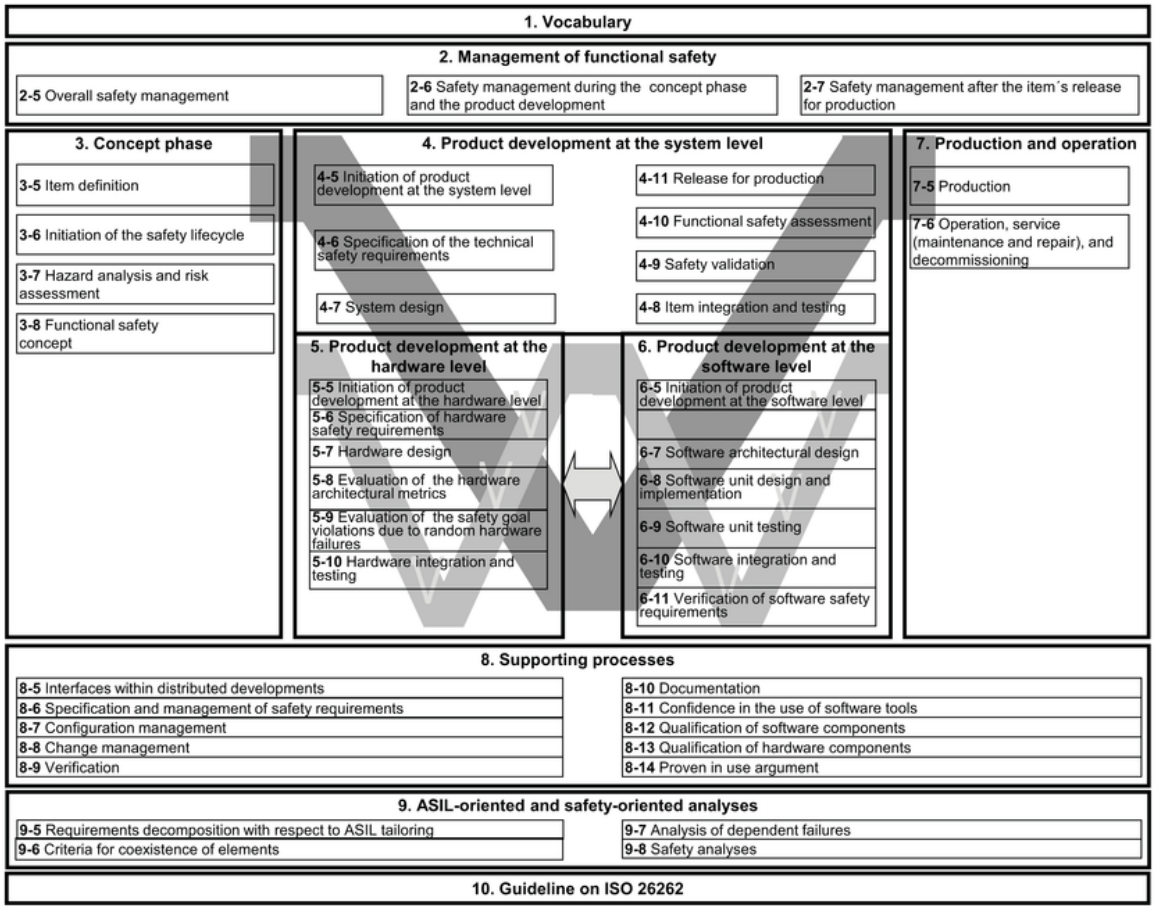


Figure 1 Overview of ISO26262 [7]

ISO 26262 does not prescribe any specific method for hazard or safety analysis. For each development phase, it suggests possible methods available but in general terms, that is, the minimum items to consider and the work products. Part 3 Clause 7.4.2.2.1 states that hazard identification should be determined systematically and notes “techniques such as brainstorming, checklists, quality history, FMEA and field studies” can be used [8].

Part 4 Clause 7.4.3.1 describes identifying the causes and effects of systematic failures[1] using a safety analysis. Also the analysis type, inductive or deductive, is recommended based on the ASIL ranking. It notes the possible methods related for each type as, "deductive analysis methods include FTA, reliability block diagrams, Ishikawa diagrams" and "inductive analysis methods include FMEA, ETA, Markov modeling", but does not specify which method [9].

The objective of safety analysis within the scope of ISO 26262 is discussed in Part 9 Clause 8 "is to examine the consequences of faults[2] and failures[3] on the functions, behavior and design …" and provide "information on conditions and causes that could lead to the violation of a safety goal or safety requirement."[10] Both qualitative and quantitative methods are listed. Methods that could be used qualitatively are listed as FMEA, FTA, HAZOP and ETA. Methods that could be used quantitatively are listed as FMEA, FTA, ETA, Markov models, and reliability block diagrams. It further notes that qualitative methods "can be used for software where no more appropriate software-specific analysis method exists" [10]. It further notes that "quantitative methods only address random hardware failures" and are not used for systematic failures, which includes design errors [10].

---

[1] Systematic failures are defined in ISO 26262 as a "failure, related in a deterministic way to a certain cause, that can only be eliminated by a change of the design or of the manufacturing process, operational procedures, documentation or other relevant factors" [7].

[2] Faults are defined in ISO 26262 as an "abnormal condition that can cause an element or an item to fail" [7].

[3] Failures are defined in ISO 26262 as a "termination of the ability of element to perform a function. Incorrect specification is a source of failure" [7].

2.3 Analysis Methods in the Automotive Industry

There are no common standardized safety analysis methods used universally throughout the automotive industry. FMEAs and FTAs are the most commonly used methods and only FMEAs have an automotive specific standard of which there are multiple [11]. Automotive manufacturers typically will have their own internal standards and practices they will require suppliers to conform to. The most common methods described in the literature along with other methods that are beginning to be used in safety-critical automotive system development are reviewed in the rest of this chapter.

2.3.1 Fault Tree Analysis

Fault Tree Analysis (FTA) is a deductive analysis technique utilizing a top-down search method to identify how causes can lead to an undesirable event (also referred to as the top event). It can be used quantitatively to perform Probabilistic Risk Assessment (PRA). The analysis is represented graphically by a tree structure with each level connected by logic gates. Each level represents the combination of possible events leading to the event above. The model can be transformed into a logic equation if it is constructed correctly. This feature can be used to facilitate the PRA calculations and allow the use of algorithms to help with the mathematical analysis of the fault tree.

FTA was developed for use on the Minuteman Guidance System by Bell Labs and later applied by Boeing to the whole Minuteman Weapon System [12]. It was found to be a useful technique and spread to other industries, such as, nuclear and automotive. The use of FTAs in the automotive industry is recognized as a valuable tool to analyze the hazards and failures of automotive electronic systems [13]. Although FMEAs are the primary tool used by many automotive OEMs, FTAs have been successfully applied to automotive systems [14][15] and an example of performing a FTA is given in ISO 26262 Part 10 Annex B [16]. The uses of "shallow" fault trees have been proposed to help derive functional safety requirements required by ISO 26262 [17]. There are no automotive specific standards regarding FTAs, but the U.S. Nuclear Regulatory Commission Fault Tree Handbook NUREG-0492 [18] is typically referenced. There is an international standard for FTAs, IEC 61025 Fault tree analysis (FTA) [5] and the FTA Handbook for Aerospace Systems written by NASA [19].

FTA is performed top-down from the top-events (i.e., hazards), which must be specified before the analysis can be started. Scoping the top-events in a crucial step in the process. If the scope is too general the analysis can become unmanageable and if it is too specific it may not provide a broad view of the system [18]. Gates are used to connect events at each level of the tree. Fault logic in the FTA model progress up the tree to the top-event. These gates describe what combination of the lower events is needed to propagate up the next level. An example of a basic fault tree for a car ignition system is shown in Figure 2. Figure 3 is an adapted

description from NUREG-4092 showing the gate and event symbols typically used in constructing a fault tree.

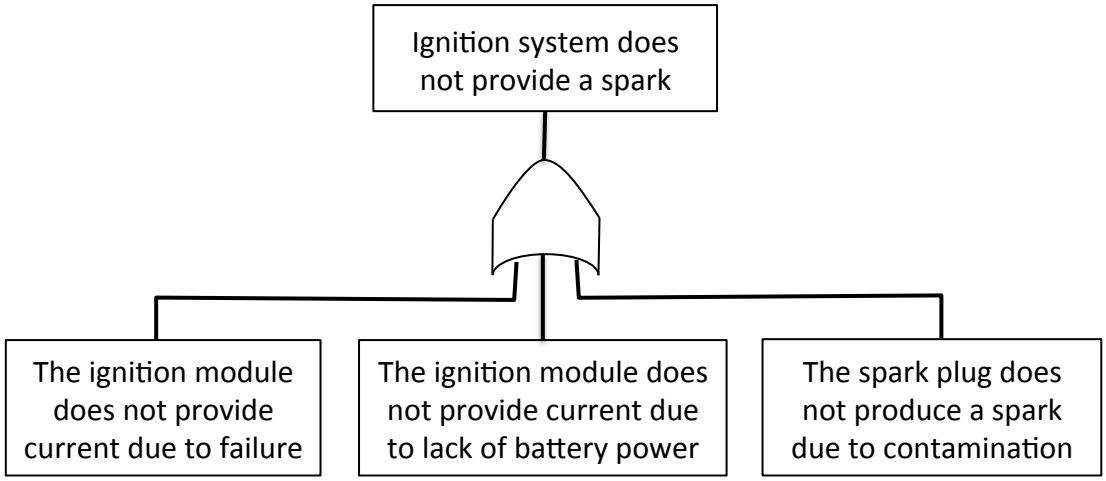


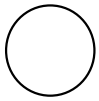
Figure 2 Example of basic fault tree

To build the fault tree, the "immediate, necessary, and sufficient causes for the occurrence of the top event" must be determined by the analyst [18]. These criteria are applied down to each level until the basic events are found or a stopping point is reached. There is no defined method to determine when to stop constructing the fault tree. It is stopped when there are no more reasonable levels left to analyze or the analyst determines the depth of the fault tree is sufficient. Vesely's basic rules for fault tree construction are summarized below:

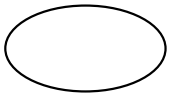
Ground rule 1: Enter the event as a fault describing what the fault is and when it occurs.

Ground rule 2: Determine if the fault is caused by a component failure. If yes then place an OR gate below the event and find the primary, secondary, and command modes. If no then the event may require AND gate, INHIBIT gate, OR gate, or maybe no gate.

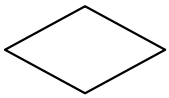
Primary Event Symbols



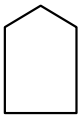
BASIC EVENT – A basic initiating fault requiring no further development



CONDITIONING EVENT – Specific conditions or restrictions that apply to any logic gate



UNDEVELOPED EVENT – An event which is not further developed because it is of insufficient consequence or because information is not available



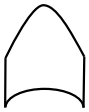
EXTERNAL EVENT – An event which is normally expected to occur

Intermediate Event Symbols

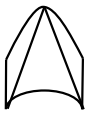
INTERMEDIATE EVENT – A fault event that occurs because of one or more antecedent causes acting through logic gates

Gate Symbols

AND – Output fault occurs if all of the input fault occurs

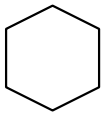


OR – Output fault occurs if at least one of the input fault occurs



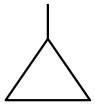
EXCLUSIVE OR – Output fault occurs if exactly one of the input fault occurs

PRIORITY AND – Output fault occurs if all of the input fault occur in a specific sequence



INHIBIT – Output fault occurs if the (single) input fault occurs in the presence of the enabling condition

Transfer Symbols



TRANSFER IN – Indicates that the tree is developed further at the occurrence of the corresponding TRANSFER OUT

TRANSFER OUT - Indicates that this portion of the tree must be attached at the corresponding TRANSFER IN

Figure 3 Fault Tree event and gate symbols adapted from [18]

Causality never passes through an OR gate. The output is the same as the input but the input is more specific. A causal relationship is identified between the inputs and outputs of an AND gate but it does not give any information about what preceded the inputs. Any dependencies between input events of an AND gate must be included in the event descriptions.

Minimal cut sets of the fault tree are the smallest number of event combinations leading to the top event. Identifying minimal cut sets can be performed automatically using algorithms if the fault tree is converted to its Boolean equivalent equations. A simple example of minimal cut sets of a fault tree is shown in Figure 4. Where the minimal cuts sets would be E1 only, E2 AND E3, E4 only, and E5 only.

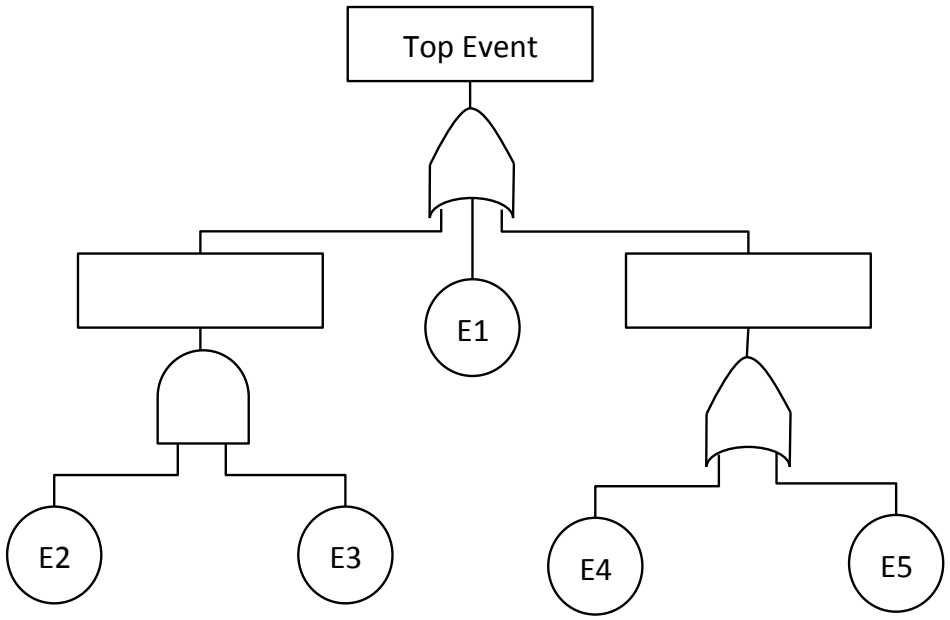


Figure 4 Example for finding minimal cut sets in a fault tree

The minimal cut sets can be analyzed qualitatively for dependencies and common cause failures [19]. A probabilistic analysis can be performed if the failure rates for

each event are known. Using the minimal cut sets, the probability of the hazard can be calculated by summing all the minimal cut set probabilities if they are all independent. If the probabilities are not independent then the conditional probabilities between the events must be used but these may be impractical to determine within a complex system. For large fault trees with many AND and OR gates, using minimal cut sets to calculate the exact top probability can become time consuming. Methods to truncate the sets may be required but can affect the accuracy of the calculation. An alternative approach using Binary Decision Diagrams (BDD) can be used to perform a more efficient and accurate calculation of the top event probability [19].

A fault tree model assumes the events are independent but failures of initiating components can be related to the same common cause. Minimal cut sets can assist in identifying possible events susceptible to a common cause failure but the actual failure must be identified using domain experience and knowledge [18]. The state transitions that occur within the system are also difficult to model with a FTA [20]. Using extensions of the Fault Tree model, such as, Dynamic Fault Trees, it is possible but there is added complexity and overhead to model such systems [19]. The logical loops created by feedback within a system cannot be modeled directly in a fault tree. Figure 5 shows a proposal by Vesely and Stamatelatos to model feedback in a fault tree by only considering the internal failure of the system receiving the signal and the internal failure of the system producing the signal [19]. Although this method

avoids creating loops within the fault tree it does not seem to capture the failures related to the feedback loop itself (i.e. delays, incorrect or missing information, etc.).

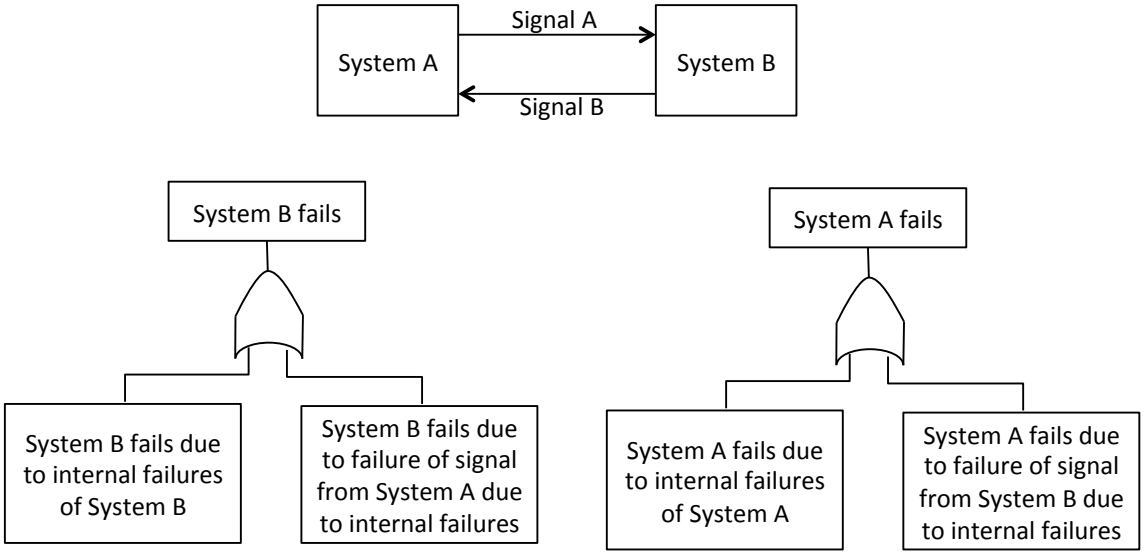


Figure 5 Modeling a logical loop in a fault tree from [19]

2.3.2 Failure Mode and Effects Analysis

Failure Mode and Effects Analysis (FMEA) is considered an inductive analysis technique utilizing a bottom-up search method to identify the effect and causes of a specified failure mode of an item. It is primarily a reliability analysis method to identify component failures and understand how often they are likely to occur. Qualitative and quantitative analysis are possible with FMEAs. The analysis follows a chain-of-events model of failures [20] to link failure modes to effects and causes.

FMEAs were developed in the 1940s and first standardized by the US Military in 1949 under Mil-P1629 “Procedure for performing a failure mode and criticality

analysis" [21]. The automotive industry began to adopt the use of FMEAs in the late 1960s [22] and have developed various standards on how they should be performed. FMEAs are used within aerospace, military, medical devices, and automotive along with many other industries [21].

The analysis results are recorded in a table showing the item, failure mode, failure rate, causal factors, and effects. Additional information about the design, such as detection method, controls, and recommendations can be recorded as well. An example of a simple table is shown in Figure 6.

Item	Failure Mode	Failure Rate	Causal Factors	Effect
Relay Coil (Normally Open relay)	Open circuit	$2x10^{-3}$	Excessive coil current burns wire	Relay contacts can not close
	Short circuit	$3x10^{-3}$	Conductive material contamination	Relay contacts can not open

Figure 6 Example of a simple FMEA table

The FMEA proceeds by listing all the items within the scope of the analysis. For each item all the possible failure modes need to be known and listed. If a quantitative assessment is to be performed, the associated failure rates for each failure mode will need to be known. This data may be found using a reliability prediction model. Many handbooks and standards are available for standard components, but care should be used to confirm the model used matches the intended application. There are no failure rates for software as there are no defined failure modes [12]. The effects of the failure are listed and then can be analyzed to understand if the effects are detectable. The causes are also listed for each failure mode along with a description

of any controls in place. A recommendation for the action to take for each of the effects is typically included.

FMEA uses a forward search method that identifies hazardous and non-hazardous effects of the failure mode. If the goal of the analysis is to identify effects that are hazardous, the identification of non-hazardous effects can be a source of inefficiency in the analysis [20]. Procedures have been developed and used in many industries to prioritize the results of the FMEA. The most common tool is called FMECA (Failure Modes Effects and Criticality Analysis). FMECA requires the designers to assign a severity value for each effect, an occurrence value for each cause, and a detection value for each control or detection method. Each value is then used to calculate what is typically called the Risk Priority Number (RPN). It is calculated using:

$$RPN = Occurrence \times Severity \times Detection$$

RPN is just one method to quantify the risk of the failure mode and specific procedures to calculate the values are typically found within the appropriate standards or regulations required for the system under development. Representing risk only with the RPN value has many limitations: it is subjective, RPN values are typically ordinal and therefore not continuous, leading to duplicate RPN values with different meanings, and using RPN thresholds tends to lead to setting arbitrarily low settings [23][24]. When using RPN, most experts recommend any high severity effect must be considered regardless of the RPN value [23].

In 1994 the first automotive FMEA standard SAE J1739, created jointly by General Motors, Ford Motor Company and Chrysler, was released. The last version SAE J1739 Revision 2009-01-05 describes an FMEA "as a systematic group of activities intended to: (a) recognize the potential failure of a product/process and the effects and causes of that failure, (b) identify actions that could eliminate or reduce the chance of the potential occurring, and (c) document the process." [22]. Similar to the SAE J1739 standard, the Automotive Industry Action Group (AIAG) also published a FMEA standard for common methods for the US automotive industry [25].

SAE J1739 covers Potential Failure Mode and Effects Analysis in Design (DFMEA) and Potential Failure Mode and Effects Analysis in Manufacturing and Assembly Process (PFMEA). A DFMEA is used during the design process to assist the design engineering team in assessing the potential failure modes in the design. A PFMEA is used during the analysis of the manufacturing process by a manufacturing engineering team to assess the potential failure modes of the manufacturing or assembly process. The PFMEA assures the manufacturing or assembly process creates a product that meets its design intent.

A Risk Priority Number (RPN) is calculated, similar to the method described for the FMECA, and is used to rank order the potential controls and actions. The latest version, Revision 2009-01-15, "de-emphasizes the use of an RPN threshold as the primary factor in determining preventive or corrective action efforts" [22].

Figure 7 shows an example of a standard from SAE J1739 worksheet for a DFMEA.

Item / Function	Potential Failure Mode	Potential Effect(s) of Failure	Sev	Class	Potential Cause(s) / Mechanism(s) of Failure	Occur	Current Design Controls	Detect	RPN	Recommended Action(s)	Responsibility & Target Completion Date	Action Results				
												Actions Taken	Sev	Occur	Detec	RPN
Front left headlight low beam bulb/ Provide light/ xx Lumens	Open filament	Driver: Reduced visibility in low light driving conditions	7	DR	Filament not strong enough for vibrations at mounted location	4	Vibration Test XXYY	5	140	Verify test profile matches vehicle mounting G forces	Engineer 1	Vehicle mouting location Gs verified to be below test profile	7	2	5	70

Figure 7 Example of Design FMEA worksheet from SAE J1739

The German automotive industry developed their own VDA-standard for FMEAs in 1996 [11]. The format and the method are similar to the SAE standard but does not completely match. “The preventive objective for both approaches is comparable but the process is a totally different one.“ [11] This is potentially a concern to suppliers in a global market trying to follow a common internal process but who must create different results and documents for the same type of analysis. This point can become even more difficult when considering most automotive manufacturers also define their own set of unique requirements for performing analysis and the format for deliverables.

FMEAs were originally developed as a reliability tool to understand the potential effects of a component failure and to quantify the likelihood of the failure occurring. It further was developed into a risk analysis tool to help rank the severity of each failure, but there are some limitations to its use as an effective hazard analysis tool, especially for a complex system. Only single item failures are considered and not

combinations of them and their interactions [12][21]. As specific point Ericson states,

> *"FMEA is not recommended as the sole tool for hazard identification. FMEA should only be used in conjunction with other hazard analysis techniques." [12]*

Some case studies of the use of different techniques to automotive applications have demonstrated the use of DFMEAs in conjunction with a FTA [14][26] to overcome the single point failure and combination of failure concerns. Henshell et al further extends the use of DFMEA by applying the analysis at different levels of system decomposition [26]. This hierarchical approach to performing FMEAs is also described by others [21][27]. Carlson notes that linking between different levels may become problematic because the FMEA is not necessarily completed for all items on all levels due to cost and prioritization reasons [21].

2.3.3 Hazard and Operability Analysis

Hazard and Operability Analysis (HAZOP) is primarily a qualitative method to identify hazards caused by deviations from the design or operating intention [20]. It uses key guidewords combined with system conditions to explore the potential deviations from normal system operation. The Institute of Chemical Industry (ICI) in the UK first formalized HAZOP in the early 1970s for use in chemical processing plants [12]. Its use has been adopted by the petroleum, nuclear, and railroad industries.

The key guidewords and parameters should be selected specific to the system and application. The combinations of these pairings are used to explore the system operation and design for deviations that potentially lead to hazards. The analysis begins with pairing guidewords with the system parameters to generate questions to analyze for deviations. The questions take the form: Guideword + parameter = deviation [12].

HAZOP is not widely used in the automotive industry but there are some references to its use mainly as a tool for Preliminary Hazard Analysis (PHA). Jesty describes a PHA for the engine management and transmission embedded controller system within Rover Group, LTD utilizing HAZOP similar to the process described in the UK Defence Standard 00-58 [28]. A HAZOP based method to identify vehicle level hazards regarding motor vehicle accident scenarios was shown to be feasible by Kazmierczak [29]. More recently, HAZOP was mentioned as one of the techniques used to perform analysis of safety-critical automotive systems by the U.S. DOT Volpe Transportation Systems Center for NHTSA's electronic systems reliability research [30]. The SAE Functional Safety Committee, made up of all major automotive manufacturers and suppliers, used a HAZOP approach for the ongoing development of the SAE J2980 standard to provide a common ISO 26262 ASIL hazard classification for certain safety-critical automotive systems [31]. A simplified example of some system functions and guidewords being used by the committee are shown in Figure 8.

	Guidewords					
System function vs. Guidewords	Loss of function	Incorrect function-i (more than requested)	Incorrect function-ii (less than requested)	Incorrect function-iii (wrong direction)	Unintended activation (incorrect timing)	Locked/ Stuck function
Electric Steering Assist	Loss of steering assist	Excessive steering assist	Reduced steering assist	Steering in the opposite direction	Unintended steering	Locked steering

Figure 8 HAZOP application from SAE J2980 Working Group activity [31]

One of the limitations of HAZOP is it only focuses on single events and not combinations. Only hazards related to guidewords will be identified. Any hazards related to a missing guideword would not be found.

2.3.4 System-Theoretic Process Analysis

System-Theoretic Process Analysis (STPA) is a top-down hazard analysis process based on the STAMP causality model of accidents. It was developed to analyze software-intensive complex systems for which other methods, designed to analyze the difficulties associated with the electro-mechanical systems of the time, were not intended to analyze. STPA can identify more than only electro-mechanical related causal factors that contribute to accidents. Factors related to design errors (including software design flaws), component interactions, human decision-making errors, and the social, organizational and managerial structure of the system can be found [1].

Using STAMP, the system level accidents and hazards are identified and the safety constraints to control the system level hazards are defined. A hierarchical functional safety control structure is used to model the system to perform STPA. Using a basic

Electronic Throttle Control (ETC) system for an automobile, the accidents and hazards for this system could be defined as:

Accident A-1: Vehicle collides with another vehicle

Hazard H-1: Vehicle does not maintain separation from another vehicle [A-1]

Figure 9 shows a simplified hierarchal control structure (only describing the control action and feedback and not the actuator or sensors for clarity) for the ETC system.

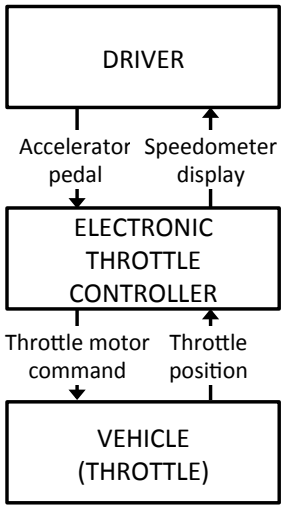


Figure 9 Hierarchical safety control structure for ETC system

For each controller in the safety control structure STPA can be used to identify the unsafe control actions and to find how they can lead to a hazard. From the unsafe control actions safety constraints can be defined. These safety constraints can be used to guide the design decisions that can enforce the safety of the system. The design can be made to eliminate, control or mitigate the hazard

STPA can be performed with two steps. Step 1 identifies what unsafe control actions can cause a hazard. Step 2 identifies how each unsafe control actions identified in

Step 1 can occur as well as how safe control actions may not be followed [1]. In Step 1, each control action is assessed to determine if it has the potential to lead to a hazard in one of the ways below:

1. The control action is not provided.
2. The control action is provided.
3. The control action is provided too late, too early or out of sequence.
4. The control action is stopped too soon or provided for too long.

Figure 10 is an example of some Step 1 results from the Throttle motor control action in the ETC system.

Control Action	Not Provided	Provided	Provided too long/ too short/ out of sequence	Stopped too soon/ applied too long
Throttle motor command	UCA-1: ETC does not provide a throttle command when the driver releases the accelerator pedal [H-1]	UCA-2: ETC provides throttle motor command when the driver does not depress the accelerator pedal [H-1]	UCA-3: ETC provides the accelerator command for too long after the driver depresses the accelerator pedal [H-1]	UCA-4: ETC stops the throttle motor command to soon after the driver releases the accelerator pedal [H-1]

Figure 10 Unsafe control actions for the Throttle motor command

The unsafe control actions found during Step 1 can be turned into system level safety constraints on the system. Some of the safety constraints generated from Figure 10 would be:

From UCA-1, SC-1: ETC does must provide a throttle command when the driver releases the accelerator pedal. [H-1]

From UCA-2, SC-2: ETC must not provide throttle motor command when the driver does not depress the accelerator pedal. [H-1]

A more rigorous analysis can be performed using the extension to STPA from Thomas [32]. The context for each unsafe control action variables can be assigned and a context table could be used to perform a more thorough search of possible unsafe control actions. For each combination of the context variables, the analyst determines if the resulting combination is hazardous. In a more detailed analysis of the ETC system, additional context could be added besides accelerator pedal being pressed or released. Some examples of additional context could be the engine condition or the brake pedal application. Figure 11 is an example of a context table for the ETC system with additional context.

				Hazardous?	
Control Action	Accelerator application	Engine condition	Brake application	Not Provided	Provided
Throttle motor command	Don't Care	Not Running	Don't Care	No	No
	Applied	Running	Applied	No	Yes
	Applied	Running	Not Applied	No	No
	Not Applied	Running	Applied	No	Yes
	Not Applied	Running	Not Applied	No	Yes

Figure 11 Context table example for ETC Step 2 using method from Thomas

How each unsafe control action found in Step 1 could occur is analyzed in Step 2. Step 2 is used to find the causal scenarios that lead to the unsafe control action and to determine how the safety controls could degrade over time. To identify the causal

scenarios leading to the unsafe control actions, the control loop for the control action is analyzed. The basic control loop for a control action is shown in Figure 12.

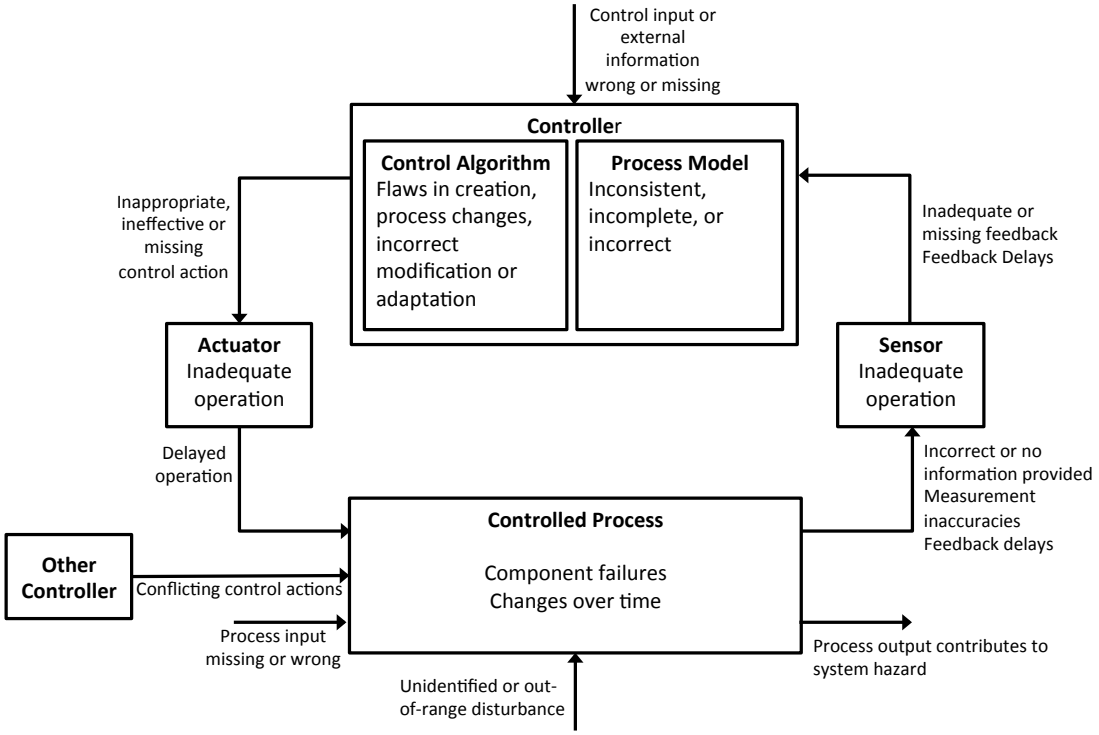


Figure 12 Control loop for Step 2 to help illicit causal scenario creation [1]

Using the control loop as a guide, causal factors can be considered and evaluated throughout the system to determine if some condition or system status can lead to a violation of a safety constraint. If a scenario is found, then the design can be changed to eliminate, control, or mitigate the causal factors. The causal accident scenarios can be identified for the ETC's unsafe control actions using the control loop in Figure 13.

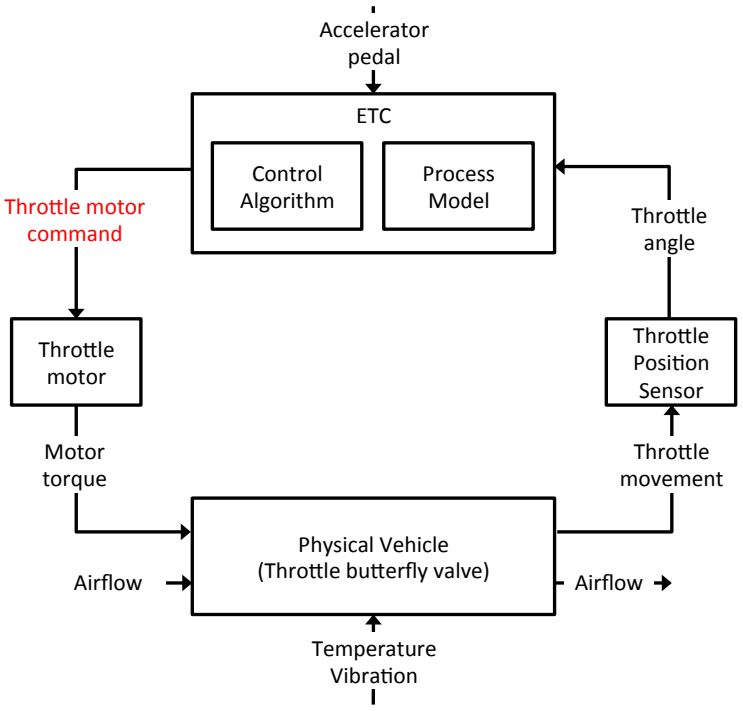


Figure 13 ETC control loop to identify causal scenarios for Step 2

Using the control loop, the following causal scenarios could be found for the UCA-2: *ETC provides throttle motor command when the driver does not depress the accelerator pedal and the engine is running*:

> Causal Scenario: Vehicle surges forward and collides with an object because a flaw in the ETC Process Model incorrectly believes the accelerator pedal input was received, throttle motor torque applied constantly even when pedal is not depressed, or the throttle butterfly valve is stuck open due to a faulty return mechanism.

For each causal scenario identified, determine if there are safety controls in the system to address them. If the causal factors are not controlled, then the system may need to be modified. Design changes can then be generated to control the causal factors found to be safety concerns.

The next part of Step 2 is to determine how implemented safety controls could potentially degrade over time. Controls could degrade over time for reasons related to environmental conditions and durability. For example, in the ETC example used above, the throttle butterfly valve stuck open can potentially lead to a vehicle collision hazard. A build up of contamination could cause this from use over time. A possible mitigation would be to include cleaning these components as part of the recurring service procedure. A better control would be to eliminate the possibility of built contamination resulting in a stuck open condition. In the automotive example above, controls could be placed to monitor changes of the system over time leading to a violation of the safety controls through fleet monitoring of vehicle data and service records.

Safety-critical automotive electronic systems are complex systems with many interactions between components that STPA was designed to analyze. STPA is a relatively new technique but seems well suited to analyze automotive systems [33]. Major automotive manufacturers are investigating the use of STPA within their development processes [34][35], and safety analysis for automotive electronic systems is being performed at the DOT Volpe National Transportation Systems Center [30]. The feasibility of finding interaction problems between multiple automotive controllers has been demonstrated [36] as well as the use of STPA in the early design phase of an automotive system [37].

2.4 Design Change Management in System Engineering

Design of complex automotive systems is an iterative process of refinement from requirement to the realization of the system [38]. Product development processes following the "V" model do not normally progress in a step-like manner from one task to another but require iterations within and between tasks. Design changes can occur due to changes in system requirements, application, components during development, schedules, responsibilities, suppliers, and the process or tool environment [39]. Many industries involved in the development of complex systems identify requirements as a source of change, and some have developed formal processes to manage the change. These processes are typically very different due to the lack of a common tool [40]. Changes to complex safety critical systems must be analyzed to ensure emergent systems properties like safety are controlled. A change to the design may lead to a violation of a safety constraint that may cause an accident. Safety-guided design allows safety to be considered while design decisions are being made and requires "tight intermingling" of the design process and safety analysis process [1]. Within the design process the safety analysis should be properly managed to ensure that the safety-related impacts of a change are understood and controlled as needed.

2.4.1 Engineering change management

Change management or configuration management are considered best practices within system engineering lifecycle management and are required by most safety-critical system development standards. The SEBok (System Engineering Body of

knowledge) [41] gives a definition for change management from Mooz as "the comprehensive evaluation and approval or disapproval of a change that takes into consideration all effects of the change. " The automotive functional safety standard, ISO 26262, considers configuration management and change management as two separate processes [42]. Part 8 Clause 7 states configuration management's objective is to "… ensure that the work products, and the principles and general conditions of their creation, can be uniquely identified and reproduced in a controlled manner at any time" and " … ensure that the relations and differences between earlier and current versions can be traced " [42]. Part 8 Clause 8 states change management's objective is to "… analyze and control changes to safety-related work products throughout the safety lifecycle " [42]. Both SEBoK and ISO26262 describe the same basic idea of the need to manage the process of analyzing, approving, and tracking changes. The definition of change management from ISO26262 is more useful for the discussion of safety-guided design and therefore references to change management will be based on this definition.

2.4.2 Change impact analysis

Change impact analysis is described as the activity of estimating the alterations to the requirements and design [43]. It is the activity associated with change management that analyzes proposed changes to the design and requirements. The key steps to the process [44] can be simply laid out as:

1. Analyze change proposal
2. Identify apparent/known impacts

3. Account for ripple effects
4. Implement change

Bohner and Arnold identify two types of impact analysis techniques. The first is dependency analysis, most applicable to software, where detailed dependency relationships between the variables, logic, modules, etc. are analyzed from the design, for example, the source code for software [44]. The second is traceability analysis where dependency relationships among all types of objects, for example, requirements and design components, are typically linked together through some identifier [44]. The prescribed impact analysis technique for system engineering is top-down requirements traceability technique [43]. Kiplinen finds that an ad hoc analysis is typically used in practice due to the lack of requirements and proper traceability tools [43]. With the growing use of Model Based System Engineering, research in methods developed for software dependency analysis may become useful to assist with analysis and traceability for a complex system [45][46].

2.4.3 Requirements traceability

Requirements are typically captured in natural language sentences and a "good requirement" is defined by Hooks as stating "something that is necessary, verifiable, attainable" with clarity [47]. As requirements are developed, links should be placed between requirements and the fulfilling design components (hardware and software) to enable traceability.

Traceability is a tool to be used to improve requirements integrity and accuracy, allow tracking and allocation, and support easier maintenance and change implementation [48]. Requirements traceability should be able to trace "forwards and backwards" from the source through the requirement to the design [49]. Relationships between requirement levels in complex systems are typically many-to-many. The layers of interconnections formed can lead to high costs in managing the requirements and difficulties in being able to reliably leverage the requirements traceability to perform change impact analysis [38]. Many have also stressed the importance of capturing assumptions and rationale during the requirements process to assure the information can also be properly traced [47][50][51]. Capturing assumptions assist the change impact analysis when reasoning about the change because the proper background information underlying the original intention of the requirement is available. Intent specifications could be used, which can facilitate the capture of assumptions and rationale throughout the entire process [50].

Links between requirements for traceability can be captured and represented in some basic ways: with a requirements traceability matrix, through hyper-linked documents, or linking statements using some kind of database support. For small projects the requirement links are typically managed manually but this becomes quickly infeasible for large projects and the use of a requirements management tool is typically recommend. Although the use of requirements managements tools can easily enable traceability, some research has found the intended use and typical

industry practice does not usually match [43]. Many challenges are typically found in the use and operation of these systems from cost, maintenance (maintaining the traceable links as the requirements changes), differing view of purpose, organization problems, and poor tool support [52]. Requirements traceability use by the automobile industry shows many difficulties to its practical use. Many times only the detailed requirements are written and the systematic-high level requirements are not documented. There can also be difficulty in tracing requirements between types, like manufacturer requirements and supplier system specifications, because information may have become embedded in one or the other due to business relationships [39]. For example, detailed algorithms may be part of the manufacturer's requirements, and what the system should do as opposed to how may become part of the supplier's provided system.

When a change is requested for a single requirement, all the requirements that trace to it are suspect until its true impact is ascertained, causing a cascade of potential latent changes [51]. These changes can cause ripple effects in both higher and lower level requirements and can be especially difficult to analyze in a complex systems where there are many interactions through feedback mechanisms. Managing and tracing the change propagation for these types of systems can be difficult. Best practices suggest the design be kept uncoupled as much as possible.

There are many methods that can be used to reduce the coupling in design that can ease the change impact analysis efforts, such as, semantic decoupling [53] or

through the use of Axiomatic design principles [54]. Typically the scope of the items to be analyzed for the change impact would be limited by expert knowledge [39][43]. Some risks with only relying on expert knowledge may be that more subtle effects are overlooked or that it is difficult to see all the interactions within the system that exist. Using system models such as STAMP can help analysis because only control of the safety critical aspects based on the safety constraints are considered.

Model Based System Engineering efforts under development may provide another method to automate the generation of traceability links from the requirements to the design and managing the changes to the links when design changes are made. Tracing requirements to the design elements has been demonstrated using a SysML model of a complex telescope project, demonstrating requirements capture and traceability [55]. More fundamentally it was shown using a MBSE approach how elements of the modeling language specify and track the requirements directly with the model artifacts [56]. MBSE is not fully used in the automotive industry, where a mixture of both models and textual requirements is typical and many tools have many shortcomings in providing ways to make the coupling between them feasible for use during a project [39].

3. Managing Design Changes in STPA for a Shift-By-Wire System

3.1. Introduction

The design of safety-critical automotive systems is typically an iterative process involving many design decisions during development. As these changes are being made, they must be managed so their impact to the safety of the system can be analyzed. Managing the design decisions can be challenging for a complex system where interactions between the system components may be nonobvious and may have nonlinear effects.

A preliminary hazard analysis to identify the system hazards and safety goals is typically performed at the start of the design process. After the design is complete, a final hazard analysis of the design is typically used to verify that the safety goals of the system are met. In practice, the design will change continuously throughout the development process and violations of the system's safety goals may go unnoticed until the final hazard analysis is performed. Performing the hazard analysis on the completed design is not as effective as understanding the impact of the design decision to system's safety when it is being made.

Using a safety-guided design process that integrates the design decisions with the hazard analysis can ensure safety is designed in during the process and increase safety without increasing complexity and cost [1].

Figure 14 shows the high level concept of this process. In safety-guided design, after design decisions are made the safety constraints are verified to ensure they are still enforced. For simple systems, experts familiar with both the system's design and STPA may be able to infer what the impact of a design decision is to the system's safety. Interactions between components in a complex system can have unintended consequences where safety constraint violations may not be as easily identified. An iterative safety-guided design process, as in [37], can keep the analysis focused on safety and only add the necessary components as the design evolves.

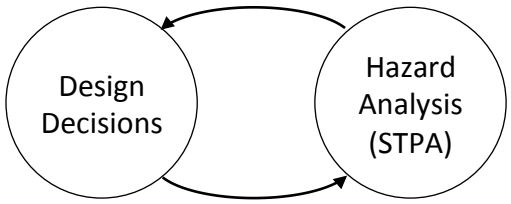


Figure 14 Safety-guided design intertwines design and hazard analysis [1]

When the design begins, it is trivial for the designer to manage the changes being proposed and perform STPA. As the system grows in complexity, however, the challenge will be to understand where to spend effort in the analysis. Performing the entire analysis can be time consuming and an inefficient use of resources. Equally as problematic, only selecting parts of the analysis without clearly understanding all the interactions in the system may lead to an incomplete analysis. The ability to determine what parts of the analysis need to be revisited after a change is made can help assist with resource planning and engineering change management procedures.

3.2. Identifying the Impact of Design Decisions on STPA

After a design decision is made the hazard analysis is performed to answer the basic question:

How will the design decision impact the safety of the system?

Specifically with STPA, the goal is to determine if all the safety constraints are enforced (not violated). The system could change in such a way that the safety constraint is no longer enforced or additional safety constraints are required. For simple systems, the task to find the impact of design changes in STPA could be trivial for experts familiar with the design and hazard analysis. In a complex system, where there are many interactions between components, the task would be much more difficult. Components of the system may have become coupled, the process model of the controller may no longer be correct for the control process, delays may have been added or changed, or assumptions about the system may no longer be valid [1]. Identifying the parts of STPA affected by the design decision may make it possible to select specific portions of the analysis to verify. Focusing efforts only on selected portions of the analysis will reduce the amount of effort and time to perform STPA in the overall design/analysis iteration.

3.2.1 Changes to the Safety Control Structure

If inspection of the safety constraints is not sufficient for determining the effect of the design decision, then a more systematic process is needed. Any design decision

made related to controlling a safety constraint should be reflected in the safety control structure. Design decisions made for reasons other than safety may need to be added to the safety control structure as well and checked for any safety constraint violations. The safety control structure can change in many ways:

- A controller, actuator, or sensor is added, removed, or changed.
- A controller's control action, input, or feedback is added, removed, or changed.
- A controller's control algorithm or process model changed.
- A controlled process changed, the input, output, or a disturbance is added, removed, or changed.
- Another controller for the process is added, removed, or changed.

The designer will use their knowledge of the system to update the safety control structure to reflect all the changes needed to implement the design change. For example, if the design change requires a new sensor the feedback will be added to the safety control structure along with the associated signals. How the controller uses the new feedback signal must be updated. Will the control algorithm or the process model or both use the new feedback information? Additionally, any assumptions or rationale made about controllers, signals, process, or the environment must be tracked and confirmed during the analysis. A design decision based on an assumption about the system will not cause a change in the safety control structure but should be verified during the analysis. For example, a controller's control algorithm is changed based on the assumption the operator will take a specific action during an event. This would only be valid if the operator's

process model included the event and the expected action was planned. While performing STPA for the new design the operator's process model should be checked to verify the assumption. If it is not valid then this could trigger an additional change to the system.

3.2.2 Identifying the impact to STPA

The first step is to update the safety control structure. Once the safety control structure is updated, the parts of STPA to be reviewed can be determined. Step 1 in STPA identifies what unsafe control actions can lead to a hazard. Step 2 identifies how these unsafe control actions found in Step 1 could potentially occur and how the controls could degrade over time. The causal relationships in the safety control structure are used in STPA to determine the unsafe control actions and their causal scenarios. When the safety control structure changes, the impact to the related unsafe control action or causal scenario can be determined.

In Step1, each control action for a controller is analyzed to generate unsafe control actions. Each type of unsafe control action is dependent on the controller, the control action, and the context [32]. Using the convention from Thomas shown in Figure 15, the design change can be checked to see if it modifies any of the elements of the unsafe control action or could introduce new ones. If the unsafe control action is not changed then the previous Step 1 results can be reused.

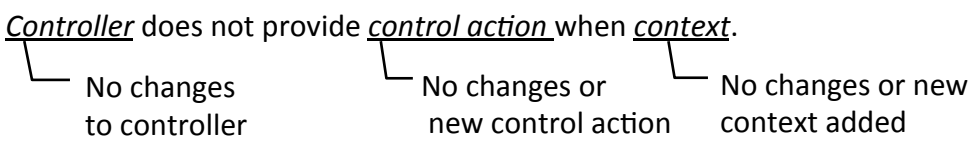


Figure 15 Change analysis of unsafe control action using description from [32]

Changes to any of these items will impact the step 1 results for the controller. In the simplest cases, adding or removing a controller or control action will require unsafe control actions and related safety constraints to be modified. Changes to the control action may result in new unsafe control actions. For example, if the signal is changed from discrete to continuous, the *provided too long* or *stopped to early* unsafe control action types should be examined to verify if they could lead to a hazard. The context of the unsafe control actions could be impacted by changes to the controller's process model, inputs, or environmental conditions. Context added that was not initially considered might lead to new unsafe control actions. Generally, any changes to unsafe control action changes will require Step 2 to check existing causal scenarios and identify any new ones.

Step 2 generates causal scenarios to identify how unsafe control actions could occur. Individual causal factors in the control loop are investigated to determine if they can cause or contribute to the unsafe control action. It is clear when step 2 must be performed again—when an unsafe control action is added. The control loop in Figure 16 from [1] can assist in deciding where the analysis effort is focused on the control and feedback parts of the control loop during Step 2 [32]. If the design change made impacts the control side of the control loop, then the effort can be used

to develop new causal scenarios and examine previous causal scenarios related to this area. Design changes made to the control side should be investigated for how a safe control action being provided but not followed could occur. Similarly, changes made to the feedback side of the control loop can limit the analysis to causal scenarios related to the other unsafe control actions.

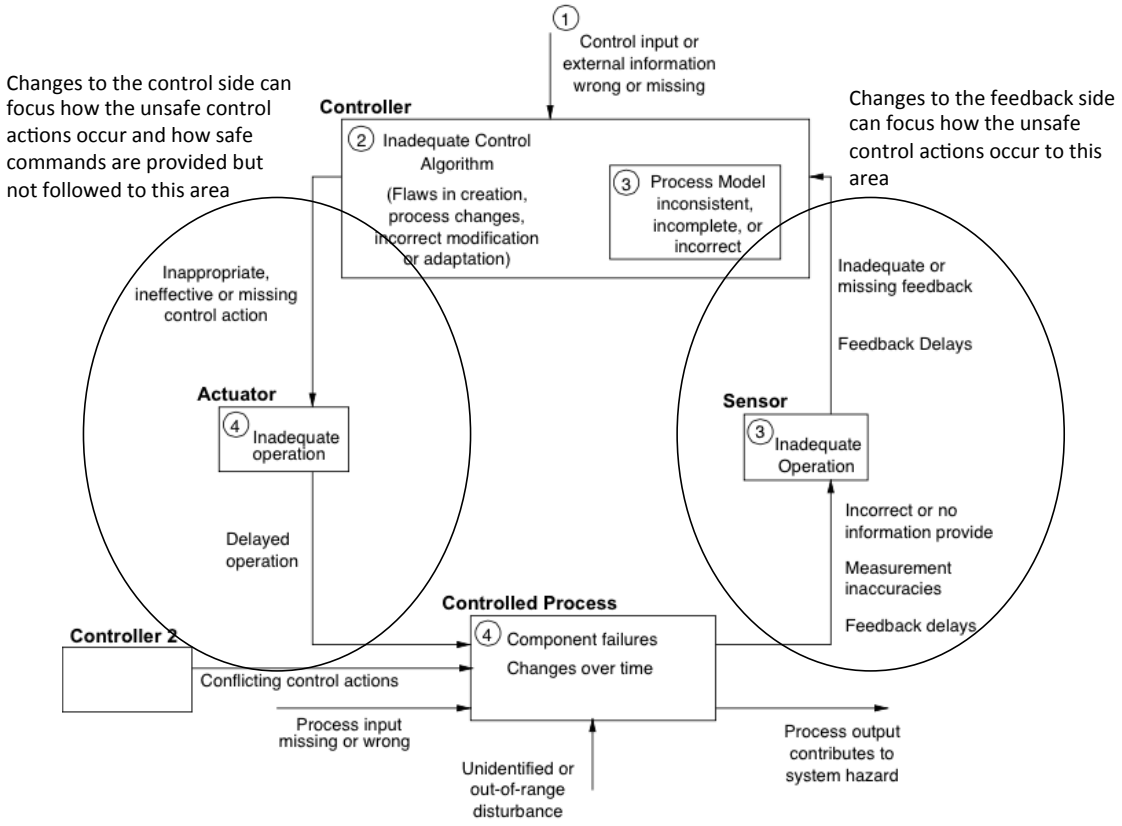


Figure 16 Control loop for Step 2. Adapted from [1].

Changes related to actuators, sensors, controllers, and the controlled processes may leverage previous causal scenarios and the specific causal factors related to the change can be examined. For example, an additional sensor is added to control when the feedback is incorrect or missing. The original scenario may still be valid but with

a new causal factor, such as, one related to a common cause failure of the two sensors. Causal scenarios that were not related to the design change may need to be verified for the new causal factor. An example of this could be when a second controller is added. Previously there was not a causal factor related to conflicting commands from multiple controllers to the controlled process but now this becomes a potential cause of an unsafe control action. Fundamentally, for any change to the control structure, it may not be possible to avoid verifying parts of the Step 2 analysis but it may be possible to focus only on specific scenarios.

Design decisions can also have an impact on the hazards defined for the system. It could be the case that a design decision is made to eliminate the possibility of the hazard. For example, in the design of a railroad crossing, the designer may decide to switch the design from using gates to building a bridge over the railroad tracks. The hazard of the train colliding with a car would be removed but a new hazard of the car falling off the bridge would be created. Therefore after the design decision, it is necessary to confirm the hazards are still applicable and no new hazards may have been created. Even when a hazard is eliminated, the STPA results may have only been affected by a small amount. Typically an unsafe control action can cause multiple hazards. Causal scenarios that address a single hazard could be removed in the case where the specific hazard was eliminated.

Based on the discussion above some general suggestions are summarized in Table 1 for investigating the impact to STPA based on the safety control structure change.

Table 1 Impact of Safety Control Structure change to STPA suggestions

Control Structure Element	Modification	Impact to STPA to consider	
		Step 1	Step 2
Controller	Remove	Results can be eliminated	Exclusive causal scenarios eliminated
	Add	Must be performed	Must be performed
Control Algorithm	Change	Use previous result	Check previous and newly related scenarios
Process Model	Change	Check unsafe control actions	Check previous and newly related scenarios
Control Input	Change	Check unsafe control actions	Check previous and newly related scenarios
	Remove	Check unsafe control actions	Check previous and newly related scenarios
	Add	Check unsafe control actions	Must be performed
Control Action	Change	Must be performed	Must be performed
	Remove	Results can be eliminated	Exclusive causal scenarios eliminated
	Add	Must be performed	Must be performed
Actuator	Change	Use previous result	Check scenarios for safe but not followed
	Remove		Exclusive causal scenarios/ factor eliminated
	Add		Check scenarios for safe but not followed
Sensor (Feedback), Controlled process (Inputs, Outputs), Disturbances, Other controllers	Change		Check previous and newly related scenarios
	Remove		Exclusive causal scenarios eliminated
	Add		Must be performed

3.3 Procedure to Manage Design Decisions

After the design decision is made, the impact to STPA can be identified and the hazard analysis portion of the safety-guided design process can focus on those items. Based on the discussion above, a procedure can be made to assist in the identification of the effects. A simple representation of the procedure within the safety-guided design process is shown in Figure 17.

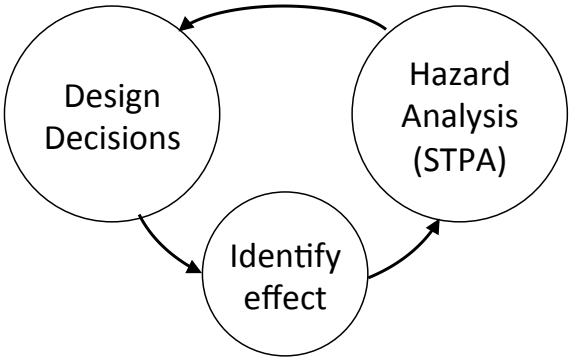


Figure 17 Identifying effects of the design decisions. Adapted from [1].

The procedure to identify the effects of the design decisions is defined in the following steps:

1. Identify the parts of the control structure changed by the design decision
2. Update the safety control structure
3. Determine if the hazards are still valid and if there are any new hazards
4. Determine the impact to the previous STPA results and which new items require analysis
5. Select the items to focus on during STPA

The procedure above is shown in Figure 18.

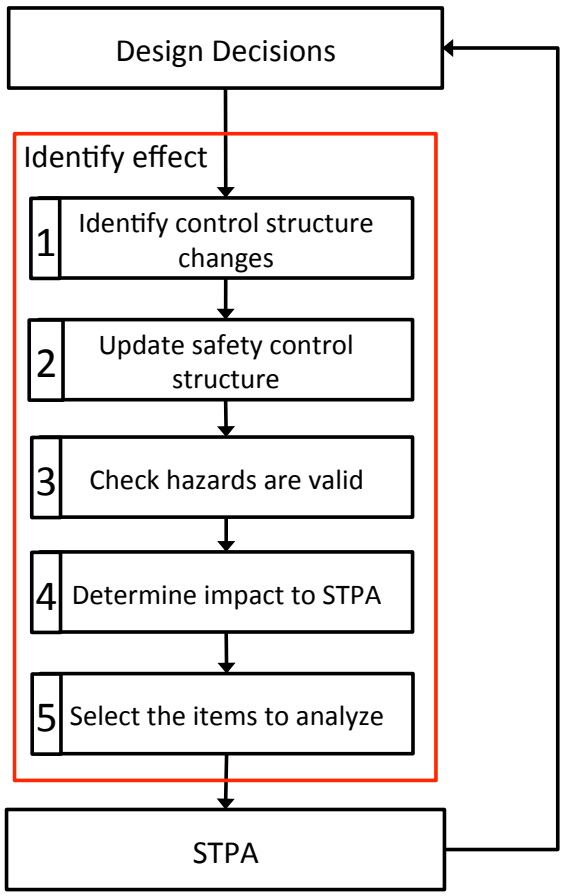


Figure 18 Procedure to manage design decisions on STPA in safety-guided design

The procedure starts with identifying the parts of the safety control structure changed by the design decision. As described in the Section 3.2.1, the designer will determine how the design decision will change the control structure. Any relationships to assumptions or design rationale used should be noted. Next the safety control structure is updated to reflect all the changes. After the control structure is updated, it is determined based on the new design if the hazards are still valid and if new hazards are introduced. If the hazards are changed or a new hazard is added, then the unsafe control action leading to the hazard will need to be identified when STPA is performed. Next, determine the impact to the previous

STPA analysis and if new items will require analysis as discussed in Section 3.2.2. Table 1 can be used to provide guidance for where to focus the next iteration of STPA. Some changes may require the analysis to be performed fully, but as the design matures the problem should change from redoing large portions of the analysis to finding the specific portions of the analysis the change could propagate to.

3.4 Demonstration of the Procedure to Manage Design Decisions

The procedure to manage design changes can be demonstrated using the shift-by-wire system detailed in Appendix A. The hazards (Table 2) and safety control structure (Figure 19) generated up to this point will be used as the starting point.

Table 2 System level safety hazards for SBW system [57]

Hazard	Description	Accident
H-1	Vehicle does not maintain safe distance from nearby vehicles	A-1
H-2	Vehicle does not maintain safe distance from terrain and other obstacles	A-2, A-3
H-3	Vehicle enters uncontrollable/unrecoverable state	A-1, A-2, A-3, A-4
H-4	Vehicle occupants exposed to harmful effects and/or health hazards	A-4

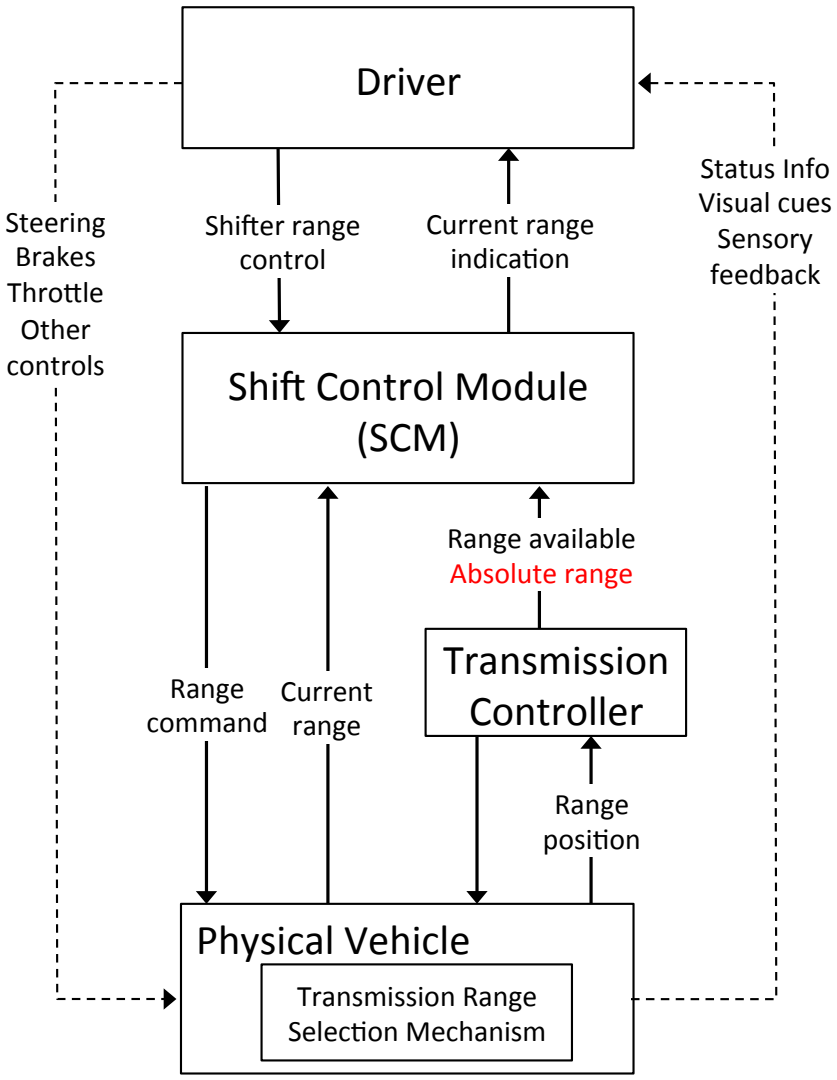


Figure 19 Safety control structure for the shift-by-wire system

The causal scenario S-SCM-1 from Appendix A,

> The vehicle moves in the wrong direction or is not secured because the driver selection was not sent for the vehicle operating conditions. This could be caused by the current range feedback incorrectly indicating the current range has been met.

A design decision can be made to add an absolute range position feedback to verify that the current range feedback of the system is correct. The absolute range position feedback information is determined to be available from the Transmission Control Unit. Adding the additional feedback requires further design decisions to be made as well. Comparing the two sensors should help guarantee the current range feedback is correct based on the design rationale that the two sensors are independent. Therefore during the causal analysis, this assumption should be confirmed. Another design decision must be made with regard to what action to take when the two signals do not match. One possible design solution is to leave the transmission in the current gear when the signals do not match. This design solution could be determined to be feasible based on the assumption that leaving the vehicle in current gear engaged is safe because the driver will be aware the gear they had selected was not engaged and take the appropriate action. This assumption should be recorded with the design decision so it can be used to identify the relationships to the safety control structure. For the convenience of the example the design changes are listed in Table 3.

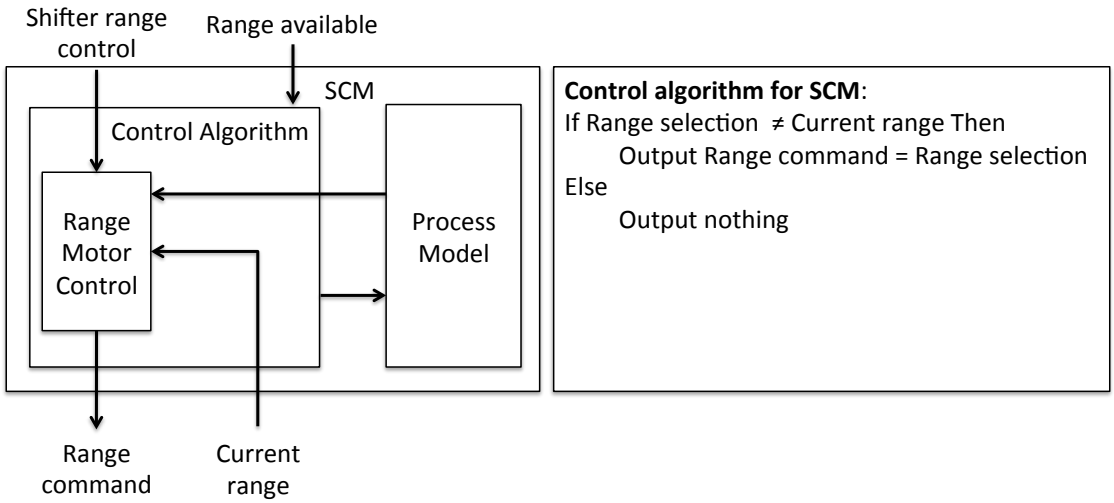
Table 3 Design decisions for absolute range position feedback

Design change	Rationale/Assumption
Absolute range information from TCU used to compare relative range position to determine current range of the transmission	The two sensors are independent
If a new range is selected and the signals are not matched leave the transmission in the current gear	The driver will be aware the gear they had selected was not engaged and take an appropriate action

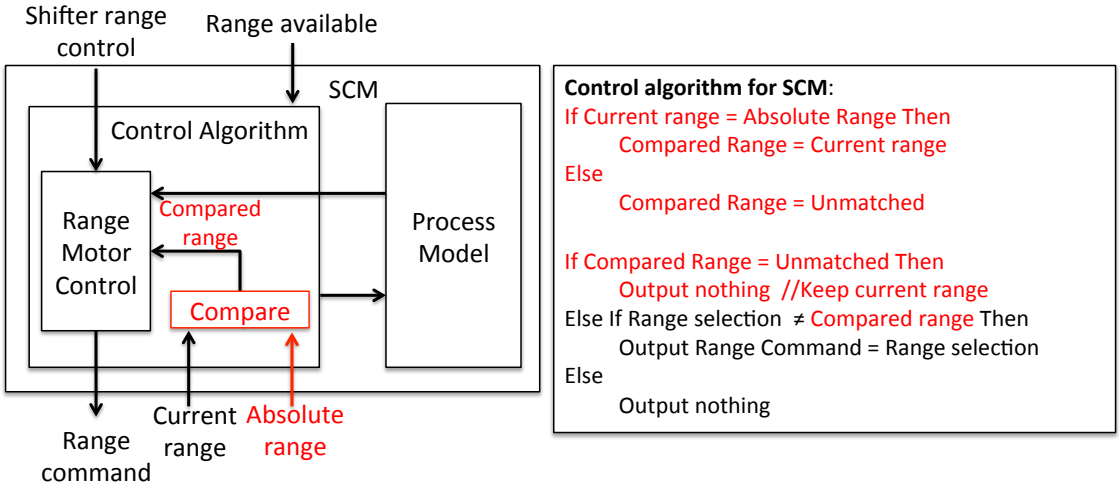
Once the design decision has been made and the assumptions recorded, the following steps can be preformed:

1. Identify the parts of the control structure changed by the design decision
2. Update the safety control structure
3. Determine if the hazards are still valid and if there are any new hazards
4. Determine the impact to the previous STPA results and which new items require analysis
5. Select the items to focus on during STPA

First the control structure changes due to the design decision are identified. Feedback for the absolute range sensor is added from the Transmission Control Unit (TCU) to the Shifter Control Module (SCM). The assumption of independence of the relative range position sensor and the absolute range position sensor is related to the current range feedback in the control structure and therefore will be captured when Step 2 is performed for the new feedback. The Shifter Control Module's control algorithm is updated to reflect the new comparison logic and action to take when the sensor values do not match. Descriptions of the Control Algorithm before and after the change are shown using a block diagram and a description of the control logic in Figure 20. The new control algorithm assumes the current gear the vehicle is in is safe because the driver will be aware the requested range was not engaged and take the appropriate action. Based on this assumption, the driver feedback in the control structure and the driver's process model should be checked to confirm the assumption made is valid.



(1) Before



(2) After

Figure 20 Before (1) and after (2) control algorithm diagram and operating logic

The related safety control structure items for the design change are summarized in Table 4.

Table 4 Summary of design change related safety control structure items

Design change/Rationale/Assumption	Control Structure Item
Absolute range information from TCU used to compare relative range position to determine current range of the transmission	• New absolute range feedback signal • Transmission Control Unit
The two sensors are independent	• Current range signal
If a new range is selected and the signals are not matched leave the transmission in the current gear	• SCM Control Algorithm
The driver will be aware the gear they had selected was not engaged and take an appropriate action	• Driver's Process Model • Driver's current range indication feedback

Next, the safety control structure will be updated based on the design decision. The absolute range feedback is already shown in Figure 19 and the SCM control algorithm update is already shown in Figure 20. Once the control structure is updated, the hazards can be analyzed to determine if they are still valid and if new hazards are required. The hazards from Appendix A are shown again in Table 5.

Table 5 System level safety hazards for SBW system [57]

Hazard	Description	Accident
H-1	Vehicle does not maintain safe distance from nearby vehicles	A-1
H-2	Vehicle does not maintain safe distance from terrain and other obstacles	A-2, A-3
H-3	Vehicle enters uncontrollable/unrecoverable state	A-1, A-2, A-3, A-4
H-4	Vehicle occupants exposed to harmful effects and/or health hazards	A-4

In this case, the proposed design changes regarding the feedback and the control algorithm of the shift-by-wire system do not change the system level hazards or create any new hazards.

Next, determine the impact to the previous STPA results and if any new items require analysis. No new controllers or control actions were added and the absolute range feedback is not being used as part of the context for any of the unsafe control actions found previously during Step 1.

Examining all the unsafe control actions from Appendix A shown in Table 6, it can be determined no parts of any of the unsafe control actions were affected by the feedback change and the previous Step 1 results can be used.

Table 6 Unsafe control actions for Range command

Control Action	Not Provided	Provided	Too early/too late/wrong order	Stopped too soon/ Applied too long
Range Command	UCA-SCM-1: Shifter Control Module does not provide range command when driver selects an appropriate and available range [H-1, H-2, H-3]	UCA-SCM-2: Shift Control Module provides range command when the range is not appropriate [H-1, H-2, H-3] UCA-SCM-3: Shift Control Module provides range command when that range is not available [H-1, H-2, H-3]	UCA-SCM-4: Shifter Control Module provides range command too late for an appropriate and available range [H-1, H-2, H-3]	N/A

The absolute range signal is part of the feedback loop and will be analyzed during Step 2 and could possibly be an additional causal factor for a causal scenario leading to the unsafe control actions. Using Figure 21 to zoom in on the control structure shows the change is only to the feedback and this can be the focus for Step 2.

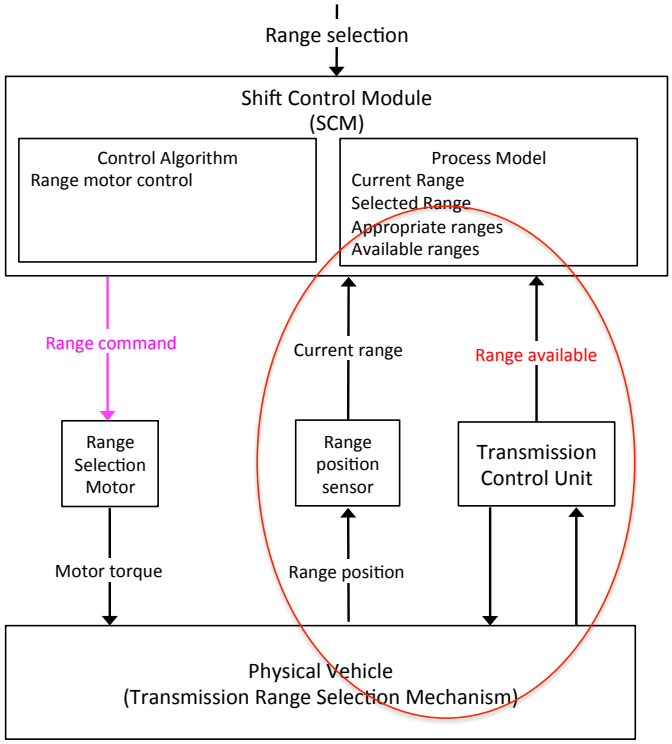


Figure 21 Zoom in for Step 2 showing area of focus for feedback causal factors

When performing Step 2, each causal scenario can be examined to see if it is related to the feedback or if new causal scenarios are generated. For example, because two sensors are being compared, the scenarios involving a delay in the signals or a difference in hysteresis should be examined. Also both feedback signals should be analyzed for possible common cause failures based on the assumption of independence made about the two sensors. Causal scenarios with causal factors

related to common causes should be specifically evaluated, such as, common power supply and communication channel.

For the control algorithm change, a similar analysis used for the feedback can determine whether any of the unsafe control actions are affected. Using the control algorithm logic, it may help determine which unsafe control action should be examined again for Step 2. The control algorithm change includes a new possibility for the control action to be not provided when the two sensors are not matched (see Figure 22). The change to the control algorithm did not change the conditions for when the control action is provided. Therefore it may be possible to examine the Step 2 results for the *not provided unsafe control actions* only. Since there were no actuator changes or changes to the condition of when a range command is provided the causes of a safe command not being followed should not be changed.

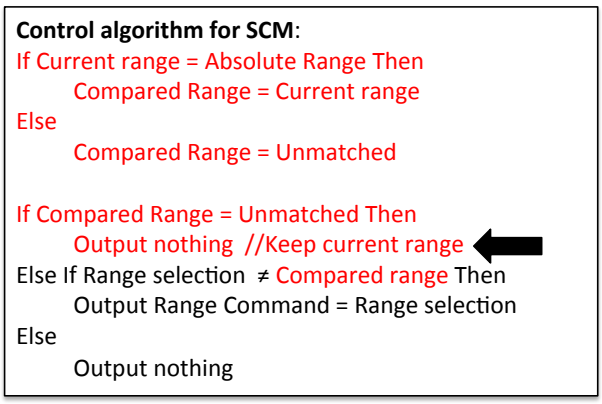


Figure 22 Updated SCM Control Algorithm analysis

Additionally examining the design rationale and assumptions made for the design change can give guidance on what parts of STPA to investigate. As previously explained, the assumptions for the control algorithm change involves the driver

feedback and the driver's process model. By evaluating this assumption, it may be possible to determine if the other parts of the control structure not related to the control loop where the design change was made could be affected by the design decision. Figure 23 shows the design change impact to the control structure.

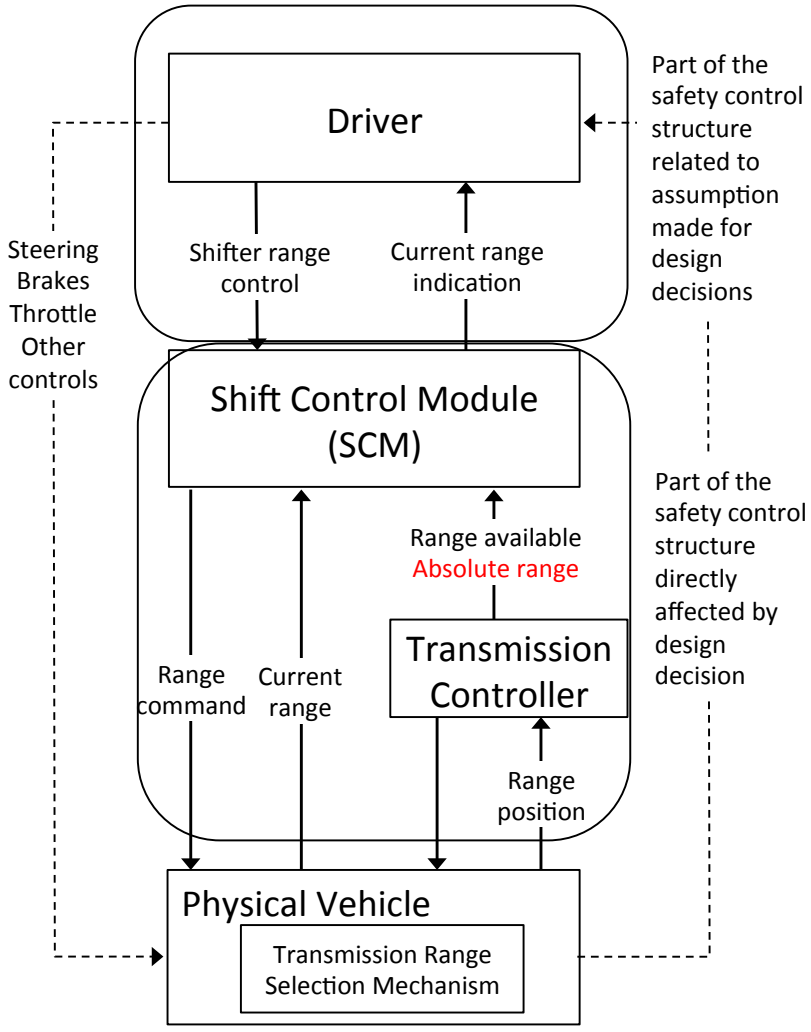


Figure 23 Overall safety control structure areas impacted by the design change

The areas of STPA selected to be examined based on adding the absolute range position sensor is summarized in Table 7.

Table 7 Summary of impact to STPA based on control structure

Design change/Assumption	Control Structure Item	STPA Item
Absolute range information from TCU used to compare relative range position to determine current range	New absolute range feedback signal Transmission Control Unit	Step 2 for new feedback signal for all SCM Range command unsafe control actions
The two sensors are independent	Current range signal	Step 2 focused on range feedback based on two sensors. With attention to common causes.
If a new range is selected and the signals are not matched leave the transmission in the current gear	SCM Control Algorithm	Step 2 for SCM Range command not provided unsafe control action
The driver will be aware the gear selected was not engaged and take the appropriate action	Driver's Process Model Driver's current range indication feedback	Step 1 and Step 2 for the driver feedback and process model

For the next iteration of the process, STPA can start from Step 2 for the unsafe control action when SCM does not provide the range command (UCA-SCM-1). The control algorithm now has the ability to not send a range command when the driver selects an available range and the two sensors do not match. Some scenarios from Appendix A generated for UCA-SCM-1 up to this point are shown in Table 8.

Table 8 Causal scenarios for UCA-SCM-1

Scenario	Description
S-SCM-1-48	The vehicle moves in the wrong direction or is not secured because the driver selection was not sent. The SCM's process model believes the selected range is not available due to an available range process model variable flaw
S-SCM-1-50	The vehicle moves in the wrong direction or is not secured because the driver selection was not sent caused by the current range feedback incorrectly indicates the current range has been met.

Each of the scenarios should be examined based on the design changes to the control algorithm. S-SCM-1-48 is caused by process model flaws and are not affected by the control algorithm change. S-SCM-1-50 was the causal scenario being controlled by the addition of the absolute range sensor. Focusing on conditions when the driver selects the range but the control algorithm does not allow the new range could lead to a hazard based on a new causal scenario:

> The vehicle may rollaway after the driver selects Park and exits the vehicle because the transmission stayed in its current gear. This could be caused by:
>
> - SCM control algorithm inadequate: SCM does not take action if conflicting feedback received,
> - SCM current range feedback timing: with 2 sources, timing delays causes unmatched range,
> - Driver's process model: driver incorrectly believed the vehicle was in park due to inadequate current range indication feedback.

New controls need to be determined for the new causal scenario identified. Using the causal factors related to the scenario, several possible design changes could be proposed:

- Provide a fail safe action based on the vehicle's operating mode.
- Return to the single range feedback and explore a different option.
- Improve the feedback to the driver when the range is not matched or the range is not appropriate for the operating mode, i.e. warning message, audible alarm.

The design change selected would be based on the usual engineering analysis considering schedule, cost, technical difficulty, and the impact to other systems. Using safety-guided design, the safety of the system would be built in during the process along with the other system requirements. To continue the example, the design change chosen to control the scenario is to provide some failsafe action based on the vehicle's operating mode when the sensor values do not match.

Following the procedure in Figure 18, the parts of the control structure changed by the design decision are identified. The SCM's control algorithm must output the appropriate range when the operating mode of the vehicle is determined. The operating mode of the vehicle can be determined using additional signals from the vehicle. These signals will be used to add a process model variable to the SCM's process model that describes the vehicle's operating mode. The additional signals required for the SCM typically would come from other vehicle controllers and the specific signals would be based on the vehicle's operating mode (parking, driver exiting, vehicle in motion, etc.). For instance, a decision can be made to automatically apply park when the driver exits the vehicle based on the rationale that this will avoid a vehicle rolling away with no driver.

Using the updated process model, the range command for Park must be provided when it is determined the driver has exited the vehicle. The control algorithm logic needs to determine whether to command Park when the driver selects park, the two range sensors do not match and the driver exits the vehicle. The updated control

algorithm block diagram and logic are shown in Figure 24 and the updated safety control structure is shown in Figure 25.

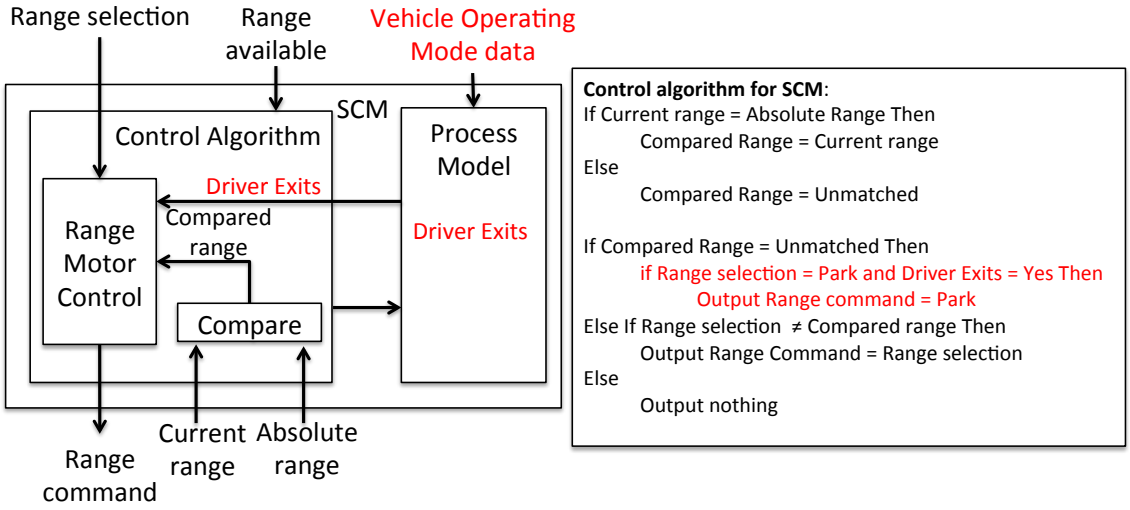


Figure 24 Updated control algorithm for new causal scenario

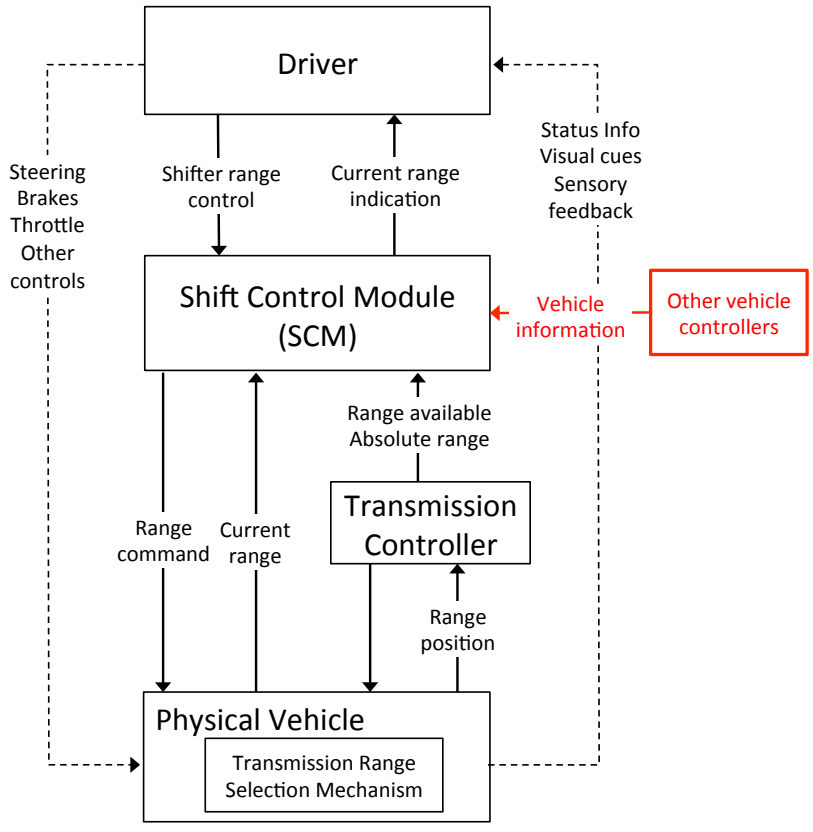


Figure 25 Updated safety control structure for new causal scenario

The hazards in Table 5 are still valid for the new design and no new hazards were created**.** For each safety control structure change, the impact to STPA can be determined. The SCM process model change requires new context to be analyzed for each of the unsafe control actions for the range command. It must also be determined if a new, potentially unsafe control action is needed. The new inputs for the process model variable will be a factor to consider during Step 2 for any unsafe control action using the new context. The control algorithm change creates the possibility to provide a command and therefore the Step 2 results for providing an unsafe control action need to be confirmed. The design rationale is to prohibit the vehicle from rolling away when the driver is not present. This was accomplished by commanding the transmission to park when the driver exits the vehicle . There may be other operating modes of the vehicle where this rationale is not adequate. For each of the unsafe control actions for the range command values—park, reverse, neutral, and drive—would need to be analyzed to determine if the decision is adequate. Therefore it is determined Step 1 should be performed for the SCM Park command control action based on the analysis.

STPA can now be started for the design change starting with the new context created by the process model change and using the value Park for the SCM range command control action. Step 1 generates multiple changes to the unsafe control actions based on the new context. The Step 1 results are summarized in Table 9.

From these unsafe control actions new safety constraints would be generated for the SCM.

Table 9 Step 1 results for SCM process model change

Control Action	Not Provided	Provided	Too early/too late/wrong order	Stopped too soon/applied too long
Park	UCA-SCM-P-1: SCM does not provide Park when the current range is not Park and the driver exits the vehicle UCA-SCM-P-2: SCM does not send a Park command when the driver selects Park.	UCA-SCM-P-3: SCM provides Park when the vehicle is moving and the driver exits the vehicle UCA-SCM-P-4: Shift Control Module provides Park when Park is not available	UCA-SCM-P-5: SCM provides Park too late when the driver is exiting the vehicle	N/A

Step 2 would then be performed for the unsafe control actions. Causal scenarios leading to the unsafe control action can be generated from analyzing UCA-SCM-P-3 (SCM provides Park when the vehicle is moving and the driver exits the vehicle). The new control algorithm will allow the vehicle to go to Park when the driver exits the vehicle. How to detect if the driver has exited the vehicle has not been discussed at this point, but no matter the detection method a possible scenario can be found when the detection is incorrect. A new causal scenario can be:

> The driver accidently selects park while the vehicle is moving and Park is engaged. Possibly due to,

- The vehicle information feedback to determine the vehicle's operating mode is incorrect. This leads the SCM process model to incorrectly believe the driver is detected to be exiting the vehicle.
- The SCM's process model is incomplete. The data being used to detect the driver exiting may not be sufficient under the operating mode. For example, if the door switch was used to determine the driver has exited the vehicle it could indicate open while the vehicle is moving.

Alternative designs can be considered at this point, such as apply more conditions to determine the correct vehicle's operating mode, improve the feedback, etc.

The rest of STPA can be performed for the other areas identified, and any safety constraints violations can be controlled. The process can be continued until the design can control all the safety constraints.

3.5 Summary

A procedure for managing the design changes for an automotive safety-critical system using an iterative safety-guided design method, described by Thomas [37], as the system's design grows in complexity was proposed. Guidance for analyzing control structure changes due to design decisions to determine the impact to a previously performed STPA was proposed to reduce the amount of rework necessary. The procedure was demonstrated using a complex shift-by-wire system, and the guidance was used to determine which steps in STPA need to be performed based on the control structure changes.

Further considerations should be given to the benefits of keeping traceability of the system from design elements to the safety constraints, and also the safety control structure and STPA artifacts. Requirements traceability is considered a best practice in system engineering and is a typical requirement for safety-critical systems. More detailed analysis can also be investigated to study how the design change may propagate revealing further areas possibly requiring more scrutiny. Finally, additional effort to understand how Step 2 can be efficiently analyzed based on the design change would be beneficial.

This page intentionally left blank.

4. Managing Design Changes with Requirements Tracing

4.1 Introduction

During the safety-guided design process, as the design of the system gets more mature, the number of safety constraints and design elements will increase. Managing changes will be a relatively trivial task at the beginning. However, as more details are added, it will become more difficult for the designers to assess all the potential impacts. Typically, most safety-critical system design standards and regulations require the use of requirements traceability, which is considered a best practice within system engineering.

The requirements traceability establishes a link between the requirement and the design element used to fulfill that requirement. It provides the design team with the ability to verify that all the requirements have been met by the system's design. For safety, it allows the system to be audited to see if all the identified safety constraints are controlled by the design. In most complex systems where components interact with each other and systems interact with other systems, a change can propagate to those other systems requiring a change as well [58]. Traceability also allows the analysis of the propagation of changes to direct and indirect elements [59]. The effectiveness of the requirements traceability will depend on which links are established and how they are maintained through the design process.

The INCOSE SE handbook [48] recommends requirements be traceable from high-level stakeholders down to the individual component requirements for hardware and software. ISO 26262 Part 8 Clause 6.43.2 requires safety requirements to be traced from their source at an upper hierarchical level to the technical safety requirements leading to specific hardware and software requirements [42].

Typically these relationships can be captured in a requirements traceability matrix or for large projects a database tool, such as, IBM Rational DOORS. Using a more complete representation of the system, such as an Intent specification [50], allows the capture of many other aspects than just requirements and design elements. It allows for the system to be described down the means-ends abstraction and across the whole-part decomposition. Elements such as the design assumptions and the results of hazard analysis can be captured and are traceable up and down the intent abstraction and left and right across the whole-part decomposition [60].

4.2 Propagation of Design Changes in STPA

The procedure in Chapter 3 demonstrates how the specific areas of the STPA analysis can be selected based on the design change but does not specifically address how the design change can propagate to other elements. The elements of the control structure related to the design change can be easily identified based on the direct link between them. The design change may also indirectly impact other areas of the control structure, but these may be more difficult to identify because there is no direct link between them. As discussed, requirements traceability can

enable impact change analysis and the matching of the impact of the propagation of the design change to the STPA results.

In STPA, high-level safety constraints can be generated from the system-level hazards. Using STPA, additional safety constraints can be generated during Step 1 and Step 2. The safety constraints generated throughout the entire hazard analysis process can be related to each other simply in a requirements tree as shown in Figure 26.

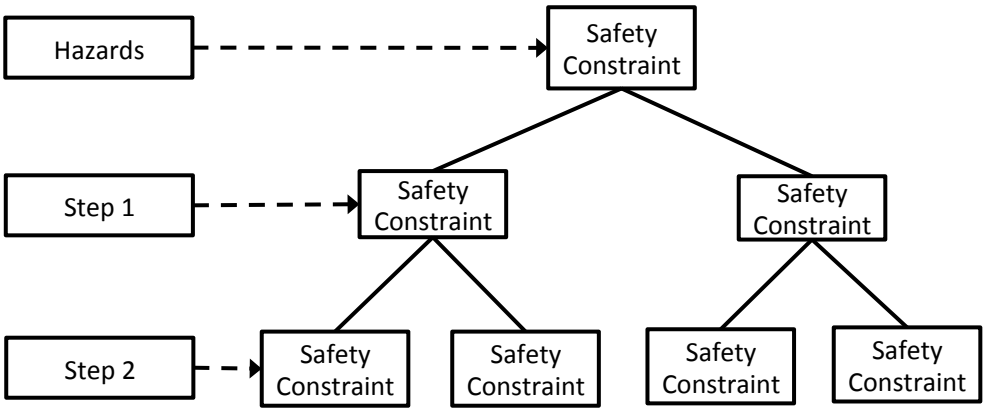


Figure 26 Safety constraints generated from STAMP and STPA

High-level safety constraints are generated from the system hazards and these would trace down to the safety constraints generated by Step1 of STPA. A change to any of the hazards could propagate down to all the safety constraints below it, as Figure 27 shows. Any lower-level safety constraints exclusively needed to enforce the hazard can be removed when a hazard is no longer applicable to the system. The same can be said for a safety constraint generated from an unsafe control action.

Safety constraints generated by Step 1 of STPA would trace down to additional safety constraints that were identified from the causal scenarios from Step 2. Any changes to the safety constraints generated by the unsafe control actions could then also propagate down to the lower levels as well.

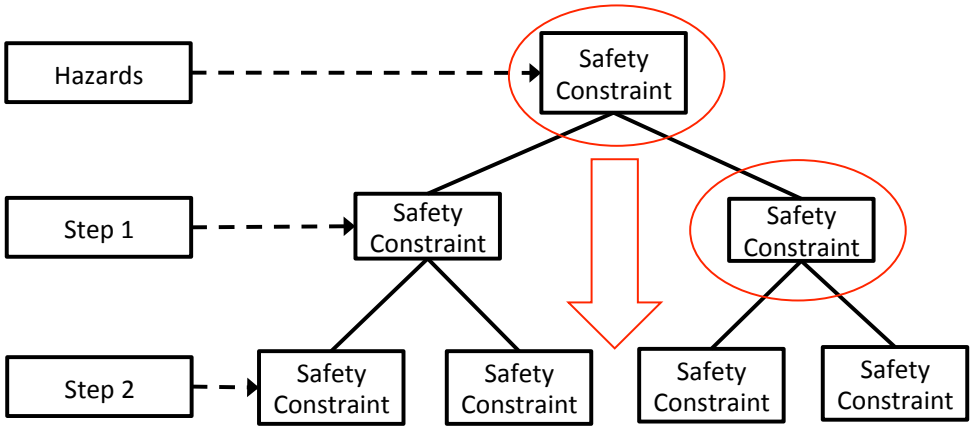


Figure 27 Change propagating down through the safety constraints

Safety constraints generated during Step 2 of STPA would trace down from the related safety constraint corresponding to the unsafe control action. The safety constraints generated by Step 2 of STPA would generally be at the bottom levels and will form the leaves of the requirements tree. A design change that impacts the results from Step 2 may propagate up and impact the unsafe control actions, as shown in Figure 28. For example, if a new sensor is added to the control structure it may change the causal scenario of the unsafe control action. This could result in another change requiring the controller's process model to change due to the initial sensor change.

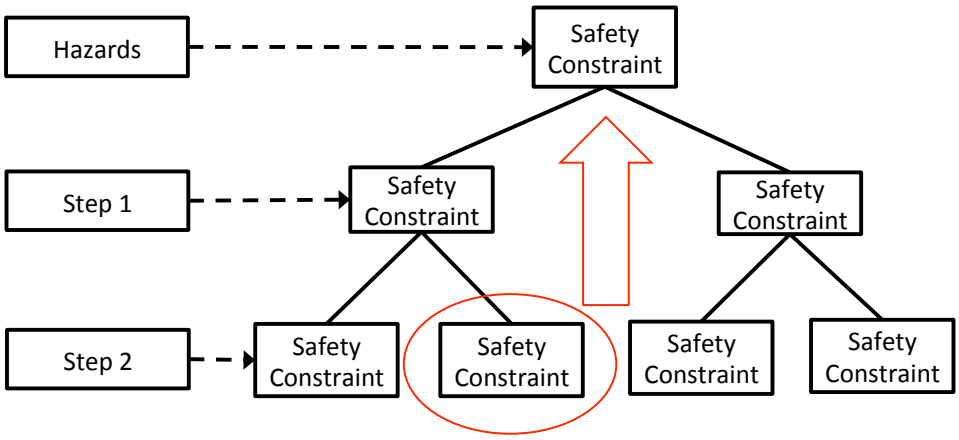


Figure 28 Change propagating up through the safety constraints

.In the sensor example above it would be possible to establish which safety constraints were generated involving that sensor and if the change to the sensor would require the Step 2 analysis to be re-performed. Figure 29 shows the relationships for the example. The sensor can be traced directly to Safety Constraint 1 and Safety Constraint 3. Safety Constraint 1 was generated from the causal factor related to the sensor in Scenario 1 of UCA1. Safety Constraint 3 was also generated from a causal factor related to the sensor in Scenario 3 of UCA 2. Therefore, Scenario 1 and Scenario 3 both would need to be re-evaluated for the impact of the sensor change. Scenario 2 is not directly related to the sensor change, but should be re-evaluated if UCA 1 changes due to any propagation from changes to Scenario 1.

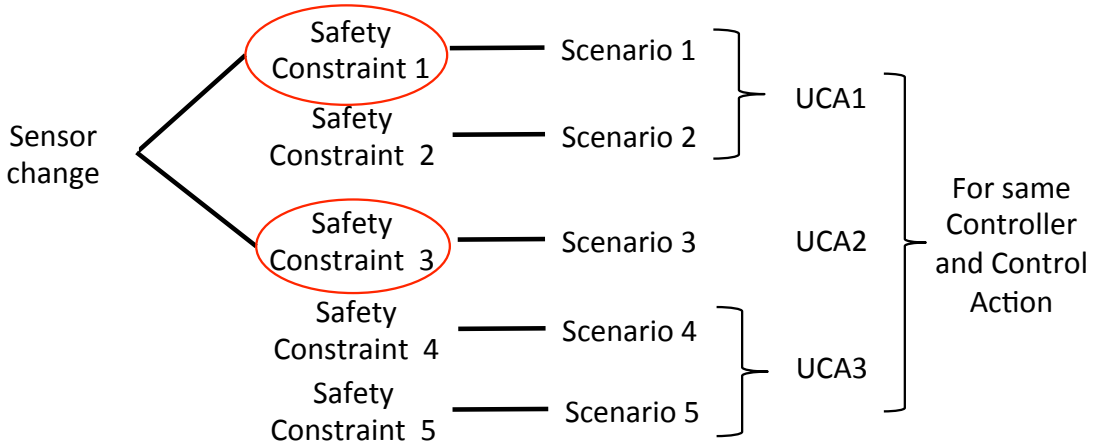


Figure 29 An example of traceable relationships between STPA elements found

UCA 3 is not directly linked to the Safety Constraints and should not need to be analyzed unless the context of the UCA changes due to the sensor change. Using the control loop in Figure 30, which was explained in Chapter 3, can assist in analyzing if UCA 3 should be examined. If the change is related to the control side of the loop, such as an actuator, control algorithm (related to the control action), or the controlled process, any unsafe control action related to these items, specifically when a safe control action is provided but not followed, should be verified. Changes related to the feedback side of the control loop that impact the context or the control algorithm of the controller should be verified.

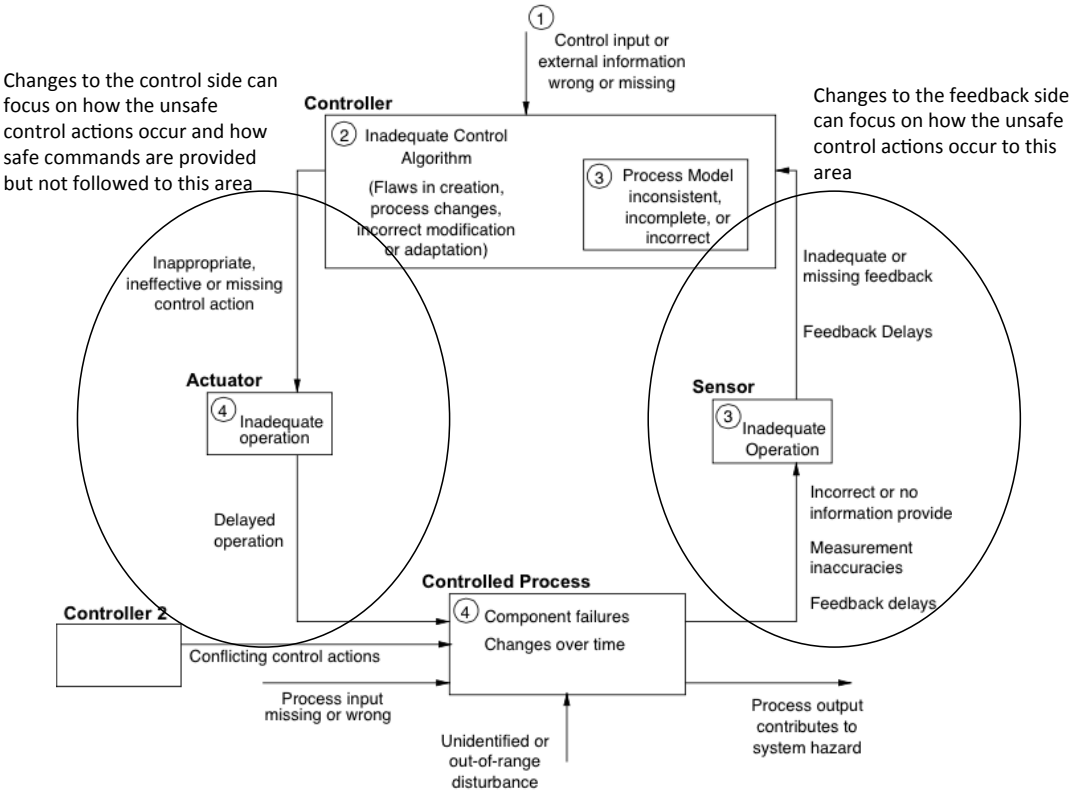


Figure 30 Control loop from Chapter 3 [1]

4.3 Traceability of the Design Change Propagation in the SBW System

The STPA analysis used for the SBW system in Appendix A can be used to demonstrate the use of traceability to assist in the management of the design change. The STPA results of the system can be traced to the design elements using an intent specification. The traceability links documented can then be used to analyze the impact of a design change to the system. Along with the procedure from Chapter 3 to identify the effects of the design change,

1. Identify the parts of the control structure changed by the design decision
2. Update the safety control structure

3. Determine if the hazards are still valid and if there are any new hazards
4. Determine the impact to the previous STPA results and which new items require analysis
5. Select the items to focus on during STPA

The SBW design in Appendix A is used for the example. The traceable links from the safety constraints to the STPA results and the system design elements modeled by the safety control structure are shown. A portion of the traceability matrix for the SBW example from Appendix B is shown in Figure 31.

	STPA														Control Strcuture																							
Safety Constraint	UCA-DR-1	UCA-DR-2	UCA-SCM-1	UCA-SCM-2	S-DR-1	S-DR-2	S-DR-3	S-DR-4	S-SCM-1	S-SCM-2	S-SCM-3	S-SCM-4	S-SCM-5	Driver	Driver Process Model	Driver Control Algorithm	Range request	Selector	Range selection	Cuurent range indication	Error message	Instrument Panel Display	Error message display	SCM	SCM Process Model	SCM Control Algorithm	Range command	Range Selection Motor	Range Selection	Range Posiion Sensor	Appropriate range	Available ranges	Other vehicle controls	Physical feedback			
SC-7	X													X	X		X																				
SC-9		X												X	X		X																				
SC-15																								X	X		X										
SC-16																								X	X		X										
SC-17					X															X																	
SC-18					X															X																	
SC-19					X																																
SC-20					X																																
SC-21						X																	X														
SC-22						X																	X														
SC-23						X	X								X																						
SC-24							X																														X
SC-25								X								X																					
SC-26								X								X																					

Figure 31 A portion of the traceability matrix used in the analysis

A design change is requested to increase the transmission cycle time of the range-available signal due to a high busload condition on the communication channel used by the range-available signal. Some possible design solutions to address the change request could be to just increase the transmission time as requested, to make the message event triggered, or do both to ensure the latency requirements for the system can still be achieved. In this example, the solution to change the message transmission method from periodic to event-triggered is chosen. This results in the available range data only being transmitted when the actual range changes.

Using the procedure from Chapter 3, the first step is to update the control structure elements based on the design decision. In this example, the available range signal's transmission method from the TCU to the SCM was changed from periodic to event-triggered. This change is in the detail design and should not modify any of the system hazards. The design change can then be traced to establish which safety constraints are related to the change using the traceability matrix from Appendix B (a reduced version is shown in Figure 32). Figure 32 shows the traces from the initiating design change to the asbolute range sensor to each related safety constraint. The identified safety constraints are SC-73, SC-74, and SC-81 and are summarized in Table 10.

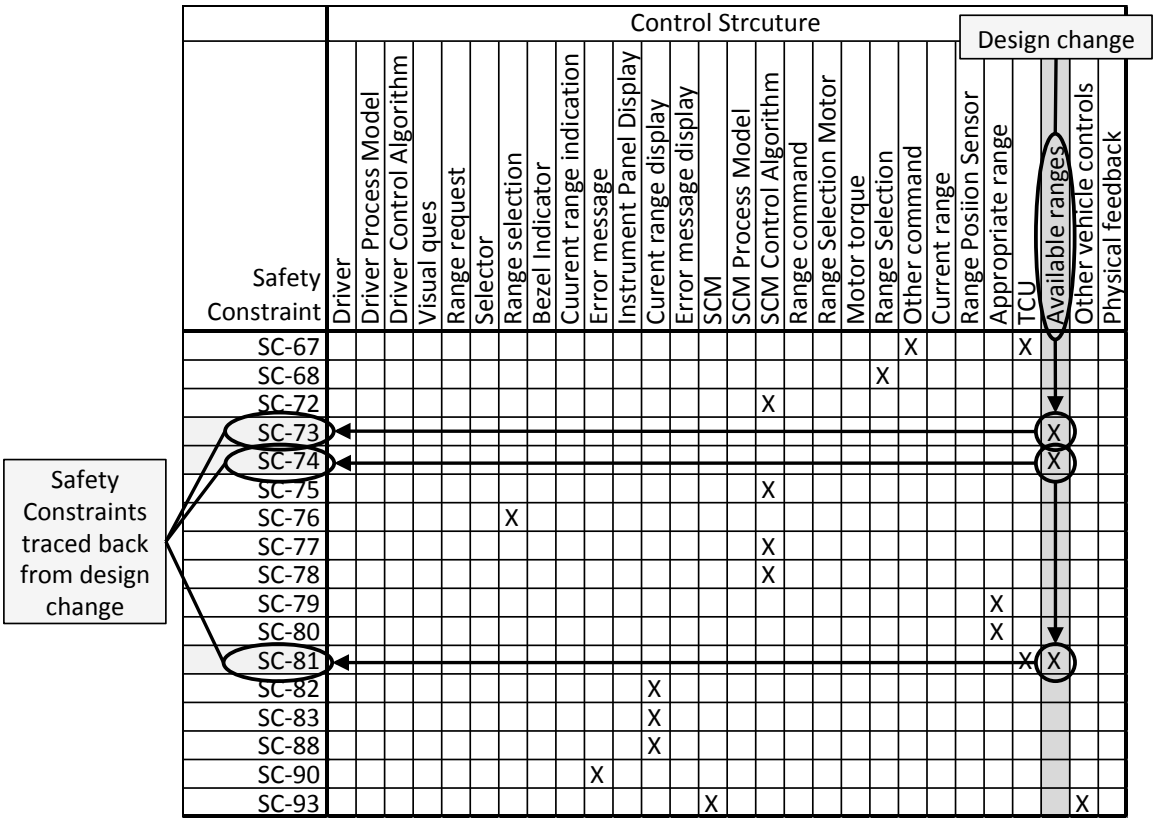


Figure 32 Traceability matrix for safety constraints to design elements

Table 10 Safety constraints and design requirements traced from design change

SC Id	Safety Constraint	Related Design Requirement
SC-73	The SCM shall identify when the available range data is not sent	The available range message will be checked that it arrived every 300 ms
SC-74	The transmission of the available range data shall be with in the tolerable range	Available range indications shall be transmitted cyclically every 300 ms
SC-81	The available range feedback shall not inhibit the appropriate range selection attempt	The appropriate range shall always be attempted to engage

Next, the safety constraints can be traced back to the STPA results that generated them. Figure 33 shows the tracing of the safety constraint back to the related causal scenarios.

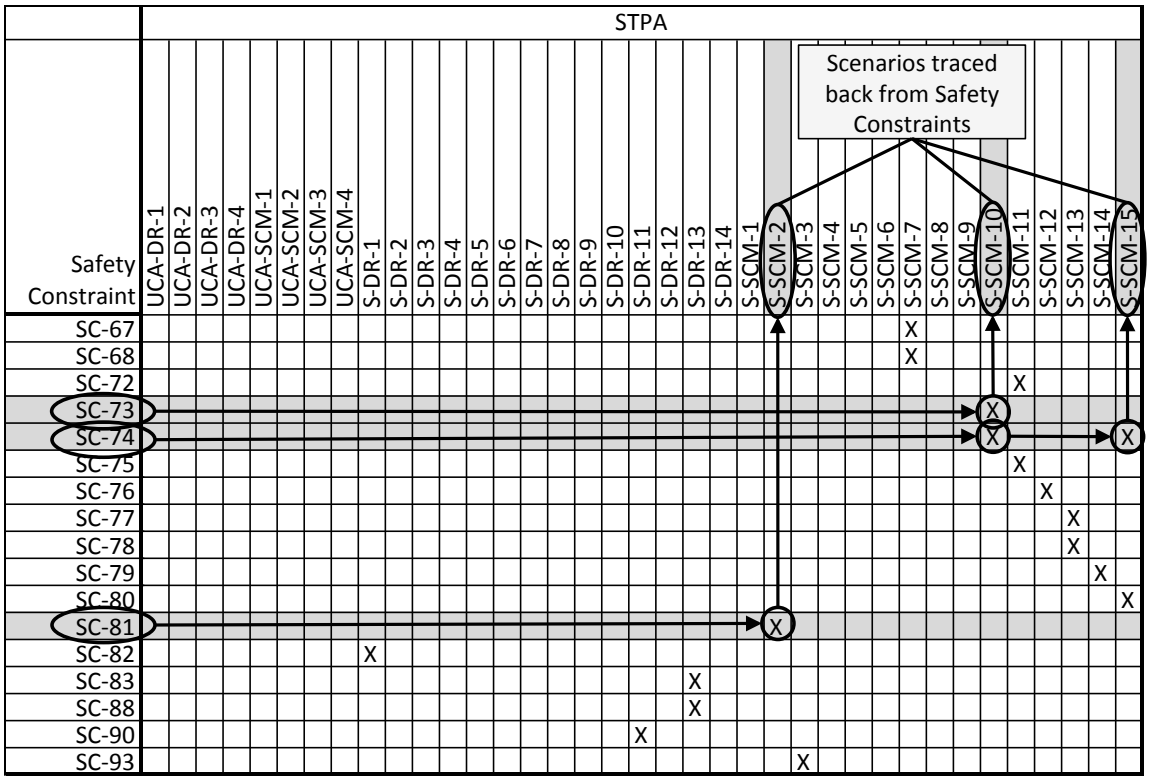


Figure 33 Traceability matrix for safety constraints to STPA results

The identified causal scenarios related to the available range data are listed below:

S-SCM-2: The vehicle moves in the wrong direction or is not secured because:

- The SCM's process model incorrectly believed that the selected range was available due to the available range data being incorrect.

S-SCM-10: The vehicle moves in the wrong direction or is not secured because:

- The SCM's control algorithm is flawed and does not properly execute the control action when the available range data is not sent by the TCU.
- The SCM's process model incorrectly believed that the selected range was available due to a delay in updating signal by the TCU.

S-SCM-15: The vehicle moves in the wrong direction or is not secured because the range was sent too late for the vehicle operating conditions caused by:

- The SCM's process model incorrectly believed that the selected range was available due to a delay in updating signal by the TCU.

An initial decision can be made related to the parts of STPA that should be re-verified based on the results above from using the traceability matrix. Safety constraint SC-73 directly traces to scenario S-SCM-10. Both S-SCM-10 and S-SCM-15 directly trace to SC-74. Based on the causal factors in each of these scenarios the design change will very likely have an impact and therefore should be re-evaluated in STPA. Scenario S-SCM-2 traces to the available range data, but the causal factor in the scenario, S-SCM-2, does not appear to be impacted by a change in transmission timing or method and therefore should not require re-evaluation.

The indirect causal scenarios related to the design change can be traced through the common unsafe control actions. Scenario S-SCM-11 is related to scenario S-SCM-10 by UCA-SCM-3. Scenarios S-SCM-12, 13, & 14 are related to scenario S-SCM-15 by UCA-SCM-4. Using the guidance suggested in Chapter 3, the scenarios described above should be examined. The results are summarized in Table 11.

Table 11 Verification decision for indirect scenarios

Scenario	Should be verified	Reason
S-SCM-11	Yes	Related to available range
S-SCM-12	No	Related to selected range input
S-SCM-13	No	Related to current range in control algorithm
S-SCM-14	No	Related to appropriate range

The other unsafe control action, UCA-SCM-3, not directly related to the change will need a closer examination to determine if there is an indirect link. UCA-SCM-3**,** Shift Control Module provides range command when the range is not appropriate, does not use the available range signal and also does not need to be re-verified. The following STPA items: scenarios S-SCM-10, S-SCM-11, and S-SCM-15 are selected to be re-analyzed.

4.4 Summary

The procedure to identify effects of design changes and the suggested guidance to select related STPA items introduced in Chapter 3 can be used within a system utilizing requirements traceability based on system engineering best practices. The direct and indirect propagation of the design change impact can be analyzed and the amount of reanalysis with STPA can be reduced without having to select only the obvious elements. The design of a complex system can easily generate hundreds or even thousands of requirements. To manage such a large set of requirements usually requires some type of requirements management tool. Future work could attempt to incorporate the procedure into one of these tools.

This page intentionally left blank.

5. Conclusion

This thesis demonstrates managing design changes for the safety-guided design of an automotive safety-critical shift-by-wire system. Current safety-related analysis methods and standards common to the automotive industry were reviewed. Also, the system engineering methods and a discussion of the use of requirements traceability for impact analysis in engineering change management was presented. A procedure was proposed to identify the impact of design changes on the safety analysis performed with STPA. Suggested guidelines were proposed to identify the impact of the change on the safety analysis performed with STPA. It was shown how to manage the impact of the design decisions in a safety-guided design process to ensure that the safety constraints are enforced. Finally, the procedure was integrated and demonstrated with a requirements traceability method.

Some possible future work related to this thesis would be:

- Establish a formal framework for identifying the impact of the control structure change to STPA.
- Integrate the framework into a requirements management tool to help identify the propagation of design changes due to the control structure changes.
- Extend the procedure to include providing guidance for making design changes based on safe design precedence, as described in [20].

This page intentionally left blank.

Bibliography

[1] N. Leveson, *Engineering a safer world : systems thinking applied to safety.* Cambridge, Mass. : MIT Press, c2011., 2011.

[2] P. Miller, "Future trends in x-by-wire systems," *SAE International X-By-Wire Automotive Systems*, 2009.

[3] E. Ackerman, "Tesla Model S: Summer Software Update Will Enable Autonomous Driving," *IEEE Spectrum Cars That Think*, 23-March 2015. .

[4] B. J. Czerny, J. G. D'Ambrosio, P. O. Jacob, and B. T. Murray, "Identifying and Understanding Relevant System Safety Standards for use in the Automotive Industry," in *SAE Technical Paper 2003-01-1293*, 2003.

[5] "IEC 61025 Fault Tree Analysis (FTA)." International Electrotechnical Commission, 1990.

[6] P. Kafka, "The Automotive Standard ISO 26262, the Innovative Driver for Enhanced Safety Assessment & Technology for Motor Cars," *Procedia Engineering*, vol. 45, pp. 2–10, 2012.

[7] "ISO 26262-1:2011(E) Road Vehicles - Functional Safety - Part 1: Vocabulary." ISO, 2011.

[8] "ISO 26262-3:2011(E) Road Vehicles - Functional Safety - Part 3: Concept phase." ISO, 2011.

[9] "ISO 26262-4:2011(E) Road Vehicles - Functional Safety - Part 4: Product development at the system level." ISO, 2011.

[10] "ISO 26262-9:2011(E) Road Vehicles - Functional Safety - Part 9: Automotive Safety Integrity Level (ASIL)- oriented and safety-oriented analyses." ISO, 2011.

[11] R. Fritzsche, "Failure Mode and Effects Analysis (FMEA): A Comparison Between VDA-Approach Versus QS-9000," in *SAE Technical Paper 2011-01-2280*, 2011.

[12] C. A. Ericson, *Hazard analysis techniques for system safety*. Hoboken, NJ : Wiley-Interscience, c2005, 2005.

[13] H. E. Lambert, "Use of Fault Tree Analysis for Automotive Reliability and Safety Analysis," in *SAE Technical Paper 2004-01-1537*, 2004.

[14] M. H. Down, "DFMEA and FTA Applied to Complex Hybrid and Fuel Cell Systems," SAE Technical Paper 2004-01-1537, 2011.

[15] F. Campean and E. Henshall, “A Function Failure Approach to Fault Tree Analysis for Automotive Systems,” in *SAE Technical Paper 2008-01-0846*, 2008.

[16] “ISO 26262-10:2011(E) Road Vehicles - Functional Safety - Part 10: Guideline on ISO 26262.” ISO, 2011.

[17] D. Sexton, A. Priore, and J. Botham, “Effective Functional Safety Concept Generation in the Context of ISO 26262,” *SAE International Journal of Passenger Cars – Electronic and Electrical Systems*, vol. 7, pp. 95–102, 2014.

[18] W. E. Vesely, F. F. Goldberg, N. H. Roberts, and D. F. Hassl, “NUREG-0492 Fault Tree Handbook.” U.S. Nuclear Regulatory Commission, January 1981.

[19] W. Vesely and M. Stamatelatos, “Fault Tree Handbook with Aerospace Applications Version 1.1.” NASA, August 2002.

[20] N. Leveson, *Safeware: System Safety and Computers*. Addison-Wesley Professional, 1995.

[21] C. Carlson, *Effective FMEAs*. Hoboken, N.J. : John Wiley & Sons, 2012.

[22] “SAE J1739: Potential Failure Mode and Effects Analysis in Design (Design FMEA), Potential Failure Mode and Effects Analysis in Manufacturing and Assembly Processes (Process FMEA).” January 2009.

[23] C. S. Carlson, “Understanding and Applying the Fundamentals of FMEAs,” in *Proceedings of the 2014 Annual Reliability and Maintainability Symposium*, 2014.

[24] J. B. Bowles, “The new SAE FMECA standard,” in *Reliability and Maintainability Symposium, 1998. Proceedings., Annual*, 1998, pp. 48–53.

[25] “AIAG: Potential Failure Mode and Effect Analysis (FMEA) 4th Edition.” Automotive Industry Action Group, June 2008.

[26] E. Henshall, I. F. Campean, and B. Rutter, “A Systems Approach to the Development and Use of FMEA in Complex Automotive Applications,” *SAE International Journal of Materials and Manufacturing*, vol. 7, pp. 280–290, 2014.

[27] B. Czerny and J. D’Ambrosio, “An Hierarchical FMEA Unified System Model for Comprehensive Hazard Analysis,” in *Proceedings of the International System Safety Conference*, 2006.

[28] P. H. Jesty, K. M. Hobley, R. Evans, and I. Kendall, “Safety analysis of vehicle-based systems,” in *Proceedings of the 8th Safety-critical Systems Symposium*, 2000, pp. 90–110.

[29] E. Kazmierczak, T. Mahmood, and D. Plunkett, “Improving Hazard Identification,” in *SAE Technical Paper 2007-01-1491*, 2007.

[30] Q. Van Eikema Hommes, “Safety Analysis Approaches for Automotive Electronic Control Systems,” presented at the SAE 2015 Government Industry Meeting, Washington, DC, USA, January 2015.

[31] D. Hartfelder, “Objectives of the SAE Automotive Functional Safety Committee,” presented at the 2012 SAE Government/Industry Meeting, Washington, DC, USA, January 2012.

[32] J. P. Thomas IV, “Extending and automating a systems-theoretic hazard analysis for requirements generation and analysis,” Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA, 2013.

[33] Q. V. E. Hommes, “Review and Assessment of the ISO 26262 Draft Road Vehicle - Functional Safety,” in *SAE Technical Paper 2012-01-0025*, 2012.

[34] P. Sundaram and M. Vernacchia, “Application of STPA to an Automotive Shift-by-Wire System,” presented at the 2014 STAMP Workshop, Cambridge, MA, March 2014.

[35] T. Morita, “Application of STPA to Hierarchical Design Approach,” presented at the Fourth STAMP Workshop, Cambridge, MA, March 2015.

[36] S. Placke, J. Thomas, and D. Suo, “Integration of Multiple Active Safety Systems using STPA,” in *SAE Technical Paper 2015-01-0277*, 2015.

[37] J. Thomas, J. Sgueglia, D. Suo, N. Leveson, M. Vernacchia, and P. Sundaram, “An Integrated Approach to Requirements Development and Hazard Analysis,” in *SAE Technical Paper 2015-01-0274*, 2015.

[38] T. Gülke, B. Rumpe, M. Jansen, and J. Axmann, “High-Level requirements management and complexity costs in automotive development projects: a problem statement,” in *Requirements Engineering: Foundation for Software Quality*, Springer, 2012, pp. 94–100.

[39] M. Weber and J. Weisbrod, “Requirements engineering in automotive development-experiences and challenges,” in *Requirements Engineering, 2002. Proceedings. IEEE Joint International Conference on*, 2002, pp. 331–340.

[40] C. Eckert, J. Clarkson, O. de Weck, R. Keller, and others, “Engineering change: drivers, sources, and approaches in industry,” in *DS 58-4: Proceedings of ICED 09, the 17th International Conference on Engineering Design, Vol. 4, Product and Systems Design, Palo Alto, CA, USA, 24.-27.08.*, 2009.

[41] *Guide to the Systems Engineering Body of Knowledge (SEBoK) — Guide to the Systems Engineering Body of Knowledge (SEBoK), version 1.3.1, 2015*.

[42] "ISO 26262-8:2011(E) Road Vehicles - Functional Safety - Part 8: Supporting processes." ISO, 2011.

[43] M. Kilpinen, P. Clarkson, C. Eckert, and others, "Change impact analysis at the interface of system and embedded software design," in *DS 36: Proceedings DESIGN 2006, the 9th International Design Conference, Dubrovnik, Croatia*, 2006.

[44] S. A. Bohner, "Software change impact analysis," 1996.

[45] S. Nejati, M. Sabetzadeh, D. Falessi, L. Briand, and T. Coq, "A SysML-based approach to traceability management and design slicing in support of safety certification: Framework, tool support, and case studies," *Information and Software Technology*, vol. 54, no. 6, pp. 569 – 590, 2012.

[46] J. Grob and S. Rudolph, "Dependency analysis in complex system design using the FireSat example," in *INCOSE International Symposium*, 2012, vol. 22, pp. 1856–1869.

[47] I. F. Hooks, *Guide for managing and writing requirements*. Houston, TX: Compliance Automation, Inc., 1994.

[48] M. Krueger, C. D. Walden, and E. R. D. Hamelin, "SYSTEMS ENGINEERING HANDBOOK," 2011.

[49] O. C. Gotel and A. C. Finkelstein, "An analysis of the requirements traceability problem," in *Requirements Engineering, 1994., Proceedings of the First International Conference on*, 1994, pp. 94–101.

[50] N. G. Leveson, "Intent specifications: An approach to building human-centered specifications," in *Requirements Engineering, 1998. Proceedings. 1998 Third International Conference on*, 1998, pp. 204–213.

[51] E. Hull, K. Jackson, and J. Dick, *Requirements engineering*. New York ; London : Springer, 2004., 2004.

[52] A. Kannenberg and H. Saiedian, "Why software requirements traceability remains a challenge," *CrossTalk The Journal of Defense Software Engineering*, vol. 22, no. 5, pp. 14–19, 2009.

[53] I. Navarro, N. Leveson, and K. Lunqvist, "Semantic decoupling: reducing the impact of requirement changes," *Requirements engineering*, vol. 15, no. 4, pp. 419–437, 2010.

[54] N. P. Suh, “Axiomatic design theory for systems,” *Research in Engineering Design*, vol. 10, no. 4, pp. 189–209, 1998.

[55] R. Karban, R. Hauber, and T. Weilkiens, “MBSE in Telescope Modeling,” *INCOSE INSIGHT*, vol. 12, no. 4, pp. 24–31, 2009.

[56] Y. Bernard, “Requirements management within a full model-based engineering approach.,” *Systems Engineering*, vol. 15, no. 2, pp. 119–139, 2012.

[57] M. S. Placke, “Application of STPA to the Integration of Multiple Control Systems: A Case Study AND New Approach,” Masters Dissertation, Massachusetts Institute of Technology, Cambridge, MA, 2014.

[58] C. Eckert, P. J. Clarkson, and W. Zanker, “Change and customisation in complex engineering domains,” *Research in Engineering Design*, vol. 15, no. 1, pp. 1–21, 2004.

[59] P. Clarkson, C. Simons, and C. Eckert, “Predicting change propagation in complex design,” in *Proceedings of the 13th International Conference on Design Theory and Methodology; part of (DETC’01)*, 2001, vol. 4, pp. 155–164.

[60] M. Stringfellow Herring, B. D. Owens, N. Leveson, M. Ingham, and K. W. Weiss, “A Safety-Driven, Model-Based System Engineering Methodology, Part I.” MIT Technical Report, December 2007.

[61] A. Ghosal, B. Czerny, and J. D’Ambrosio, “Fault-Tree Generation for Embedded Software Implementing Dual-Path Checking,” in *SAE Technical Paper 2011-01-1004*, 2011.

[62] P. E. Ross, “Robot, you can drive my car,” *IEEE Spectrum*, no. 51, pp. 60–90, 2014.

[63] R. D. A. (EIC) BKCASE Editorial Board, Ed., *Guide to the Systems Engineering Body of Knowledge (SEBoK) — Guide to the Systems Engineering Body of Knowledge (SEBoK), Version 1.3.1*. Hoboken, NJ, 2015.

[64] “ISO 26262-2:2011(E) Road Vehicles - Functional Safety - Part 2: Management of functiional safety.” ISO, 2011.

[65] N. R. C. (US) C. on E. V. Controls and U. Acceleration, *The Safety Promise and Challenge of Automotive Electronics: Insights from Unintended Acceleration*, vol. 308. Transportation Research Board, 2012.

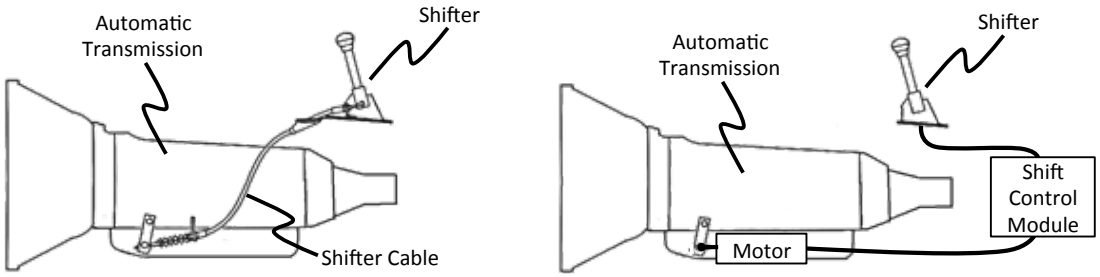
[66] R. Smaling and O. de Weck, “Assessing risks and opportunities of technology infusion in system design,” *Systems Engineering*, vol. 10, no. 1, pp. 1–25, 2007.

This page intentionally left blank.

Appendix A: STPA Applied to a SBW System

System Description

A shift by wire (SBW) control system replaces the mechanical cable and linkage from a mechanical shifter to the transmission that is used to select the transmission range by the driver (see Figure 34). The shifter sends electrical signals to the Shift Control Module that then sends a command to a motor to select the transmission range. A second electronic controller receives the motor range command sent by the Shift Control Module and provides the electrical signal to the motor. The motor then rotates a shaft to the proper position for the range commanded.



(a) Mechanical cable shifter system (b) Electronic shift-by-wire system

Figure 34 Example of (a) mechanical and (b) electronic shifter systems

Based on the physical design of the transmission and the vehicle the following basic operating constraint for the system can be described.

Generally, if the vehicle is travelling above a certain speed engaging Park or the range opposite the direction of the vehicle motion may cause damage to the vehicle or make the vehicle difficult to control. Although cases may exist where applying Park is preferable to the alternatives.

If the vehicle is not moving and the customer exits the vehicle, engaging Park or Neutral with the emergency brake is required to stop the vehicle from rolling away. If the vehicle needs to move freely in the case of being towed or using a car wash with a conveyer then the system will need to allow an exception to holding the vehicle.

System Goals

The system shall provide the means to:

1. Allow the driver the to select the required range for operating the vehicle
2. Display the selected and available ranges to the driver
3. Allow for the controlled operation of the vehicle

Accidents

Generic automotive system level accidents were described by [57]. Provided below are examples and brief descriptions of the accidents listed in Table 1 that identify the undesired losses and provide the basis for hazard identification.

A-1: Two or more vehicles collide

A-2: Vehicle collides with non-fixed obstacle[4]

A-3: Vehicle crashes into terrain[5]

A-4: Vehicle occupants injured without vehicle collision

System-level Hazards

After identifying the undesired losses and associated accidents, we identify the system level hazards that may lead to an accident. A set of generic safety focused hazards for automobiles is described in [57] are used.

H-1: Vehicle does not maintain safe distance from nearby vehicles [↑A-1]

H-2: Vehicle does not maintain safe distance from terrain and other obstacles [↑A-2, ↑A-3]

H-3: Vehicle enters uncontrollable/unrecoverable state [↑A-1, ↑A-2, ↑A-3, ↑A-4]

H-4: Vehicle occupants exposed to harmful effects and/or health hazards [↑A-4]

SBW system specific hazards

Based on the generic hazards system specific sub-hazards can be described related to the transmission system and the goals of the Shift-By-Wire system.

H-1.1: Vehicle rolls away (not secured) [↑H-1, ↑H-2]

H-1.2: Vehicle moves in the opposite direction than intended [↑H-1, ↑H-2]

H-3.1: Vehicle decelerates too quickly while at high speed [↑H-3]

H-3.2: Vehicle stops suddenly while at a low speed [↑H-3]

H-3.3: Vehicle cannot roll freely when intended [↑H-3]

System Level Safety Constraints for the SBW system

From the system level hazards we can sate the system level safety constraints:

---

[4] 'Other obstacle' includes pedestrians, bikers, animals, etc.

[5] Terrain' includes fixed, permanent objects such as guardrails, trees, bridges, signage, pavement, etc.

SC-1: Vehicle must be secured so it cannot roll away [←H-1.1]

SC-2: Vehicle must move in the direction intended [←H-1.2]

SC-3: Vehicle must not decelerate too quickly [←H-3.1]

SC-4: Vehicle must not stop suddenly [←H-3.2]

SC-5: Vehicle must be able to roll feely when intended [←H-3.3]

SC-6: Vehicle must not expose occupants to harmful effects and/or health hazards [←H-4]

Safety Control Structure

The safety control structure for the SBW is constructed from the components of the system involved in controlling the physical process of interest. Figure 35 shows the high-level safety control structured.

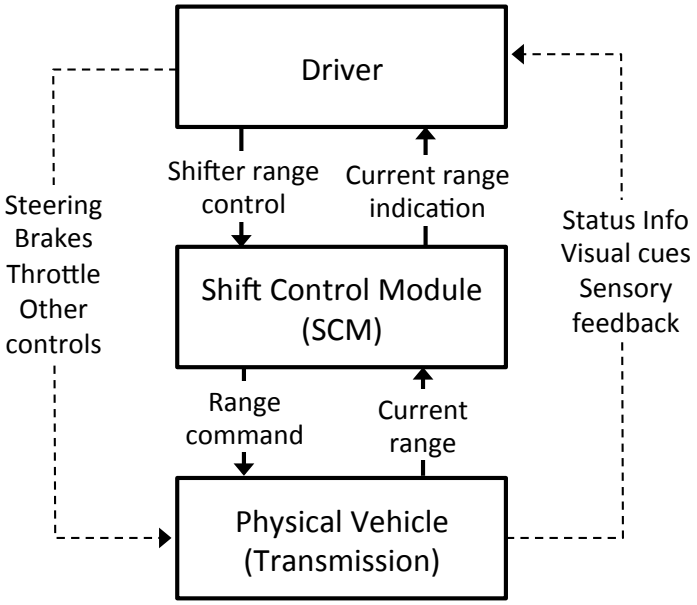


Figure 35 High Level Safety Control Structure

The Driver uses information from the vehicle's displays and external information to determine the appropriate range to select based on what operation is intended. The driver selects the range by engaging the shifter that will send a signal to the Shifter Control Module (SCM), which in turn engages a motor to move the range selector mechanism until the desired range is achieved. The SCM is a computer controller consisting of embedded hardware and software logic. The range selector mechanism is part of the transmission.

Possible control actions from the driver would be:

Shifter range control, this would be the range desired by the driver and would be Park, Reverse, Neutral, or Drive. The individual gears within Drive, 1st, 2nd, 3rd, etc. could also be

selected but are not included in this analysis and would be similar to the analysis for Drive for most of the analysis.

Steering, *Brakes*, and *Throttle* are the other possible dynamic control actions the driver can use to control the vehicle. There are also other control actions possible the Driver could provide not described, for example, the ignition switch, the parking brake, etc.

The Driver would receive feedback about the range engaged by the *Current Range Indication* from the SCM.

External feedback from the vehicle and environment would also be received by the driver used to assist in determining the state of the vehicle and the conditions, such as, other status information from the vehicle's display, visual cues from outside, and sensory information (vehicle motion, audible feedback like the locking of the door, etc.).

To select the range for operating the vehicle the basic information required by the Driver's process model can be assumed to be:

The *Operating mode* of the vehicle, such as, driving forward, backing up, parking, picking some one up or dropping them off, etc.

The current *Driving conditions* effecting the vehicle and driver, such as, weather, traffic, type of road, and time of day.

The *Current range* the transmission is in which could be Park, Neutral, Reverse, or Drive.

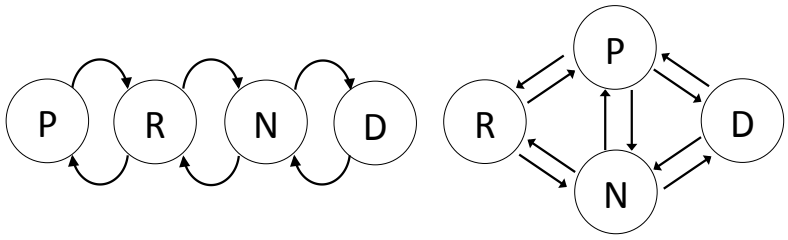
The *Selected range* by the Driver, which could be Park, Reverse, Neutral, Drive, or None.

The *Appropriate* range for the vehicle based on the operating mode and driving conditions.

The *Available ranges* for the vehicle based on information from the SBW system.

The *System status* of the shift by wire system providing information if it is working properly, for example, warning messages or other indications of a problem and potential actions to take.

The Driver's Control Algorithm would consist of how to select the ranges based on the shifter design and the understanding the driver has of how the shifter operated. This would probably include the amount of force required, the sequence or pattern, and the duration of time needed to engage a range. For example in the traditional mechanical systems the ranges could only be selected in a linear fashion as dictated by the shifter design shown in Figure 36 where Park can be selected from any range but Reverse, Neutral, and Drive must be operated in sequence, as some newer SBW designs.



(a) typical design (b) possible design

Figure 36 Range selection direction sequence

For now the design of the how to select the gear is not fixed and it is assumed the Driver's control algorithm is enough to operate the shifter.

The control actions for Shift Control Module (SCM) would be:

The *Range command* to the motor for one of the ranges, Park, Reverse, Neutral, or Drive.

The SCM would receive the *Current range* feedback from the transmission indicating which range the transmission is in, Park, Reverse, Neutral, or Drive.

The basic process model for the SCM would assume to consist of:

The *Selected range* from the Driver (the driver's requested range).

If the selected range was *Appropriate* and if the selected range was *Available*.

The Control Algorithm for the SCM would be required to be able to provide the correct signal to control the range motor selecting the range. The details are left out for now because it would be based on the type of motor and other factors unknown at this early design stage. The Range Motor Control function controls the Range command based on the current range and selected range. Additionally, the control algorithm would decide to send the selected range command based on if the range is available and appropriate.

The basic control algorithm can be expressed functionally and the logic is expressed using pseudo code in Figure 37.

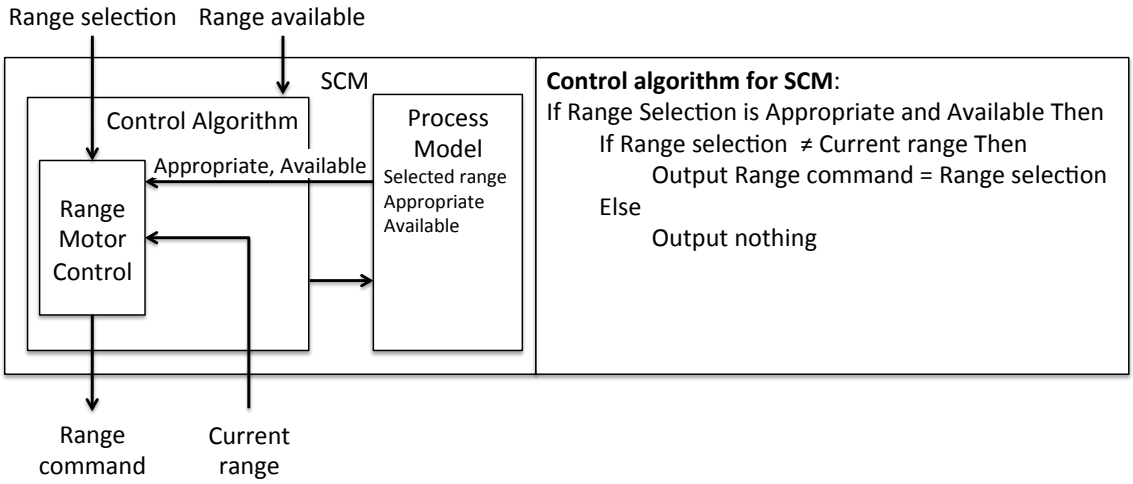


Figure 37 Control algorithm for SBW Controller

The physical process being controlled is the gear range selection mechanism in the transmission. The gear range selection mechanism is rotated to the position for the desired range (see Figure 38). This typically controls hydraulics inside the transmission to allow fluid to engage specific clutches and allow torque to transfer or not from the input to the output. When Park is selected engaging a lever called a parking pawl also mechanically locks the transmission.

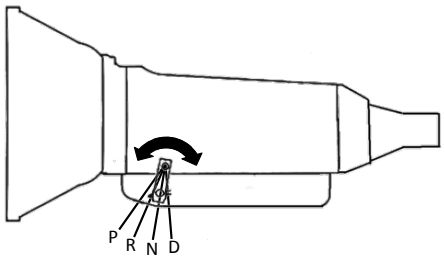


Figure 38 Operation of range selection mechanism external to transmission

The transmission is an electro-hydraulic device to provide power generated by the engine to the vehicle's driven wheels. The operation of the transmission is controlled by the Transmission Control Unit (TCU) to decide when and how to shift the forward gears during driving. It controls the hydraulic fluid to clutches to allow the power to be modified by different gear ratios, allow the output shaft to rotate in the opposite direction for reverse, and disconnect the power flow from input to output for Neutral and Park ranges. The gear range selection mechanism is responsible for selecting the range.

Driver

Identify Unsafe Control Actions (Step 1)

Table 12 Unsafe control actions for Driver

Control Action	Not provided	Provided	Too early/too late/ wrong order	Stopped too soon/ applied too long
Range selection	**UCA-DR-1** Driver does not provide new range selection when appropriate [↑H-1, ↑H-2,↑H-3]	**UCA-DR-2:** Driver provides range selection when the range selected is not appropriate [↑H-1, ↑H-2, ↑H-3] **UCA-DR-3:** Driver provides range selection when the range selected is not available [↑H-1, ↑H-2, ↑H-3]	**UCA-DR-4:** Driver provides range selection too late when the range selected is appropriate [↑H-1, ↑H-2, ↑H-3]	N/A

Safety Constraints from Step 1 for the Driver

The Driver is a human controller so technical safety constraints cannot be enforced but they may be useful to assist in designing the system to be operated in a safe manner. Also the safety constraints may also be useful as guidance for determining possible training and what information may be necessary or need emphasis within an owner's manual.

The safety constraints are stated as:

SC-7: Driver must provide new range selection when appropriate [←UCA-DR-1, ↑SC-1, ↑SC-2, ↑SC-3, ↑SC-4, ↑SC-5**]**

SC-9: Driver must provide range selection that is appropriate [←UCA-DR-2, ↑SC-1, ↑SC-2, ↑SC-3, ↑SC-4, ↑SC-5**]**

SC-10: Driver must provide range selection that is available [←UCA-DR-3, ↑SC-1, ↑SC-2, ↑SC-3, ↑SC-4, ↑SC-5**]**

SC-11: Driver must provide range selection too late when the range selected is appropriate [←UCA-DR-4, ↑SC-1, ↑SC-2, ↑SC-3, ↑SC-4, ↑SC-5**]**

Determine How Unsafe Control Actions Occur (Step 2)

For each controller determine the causal scenarios that could lead to the unsafe control actions, when a safe control action is provided but not followed, and how the controls could degrade over time.

Driver (Operator)

How unsafe control actions could occur:

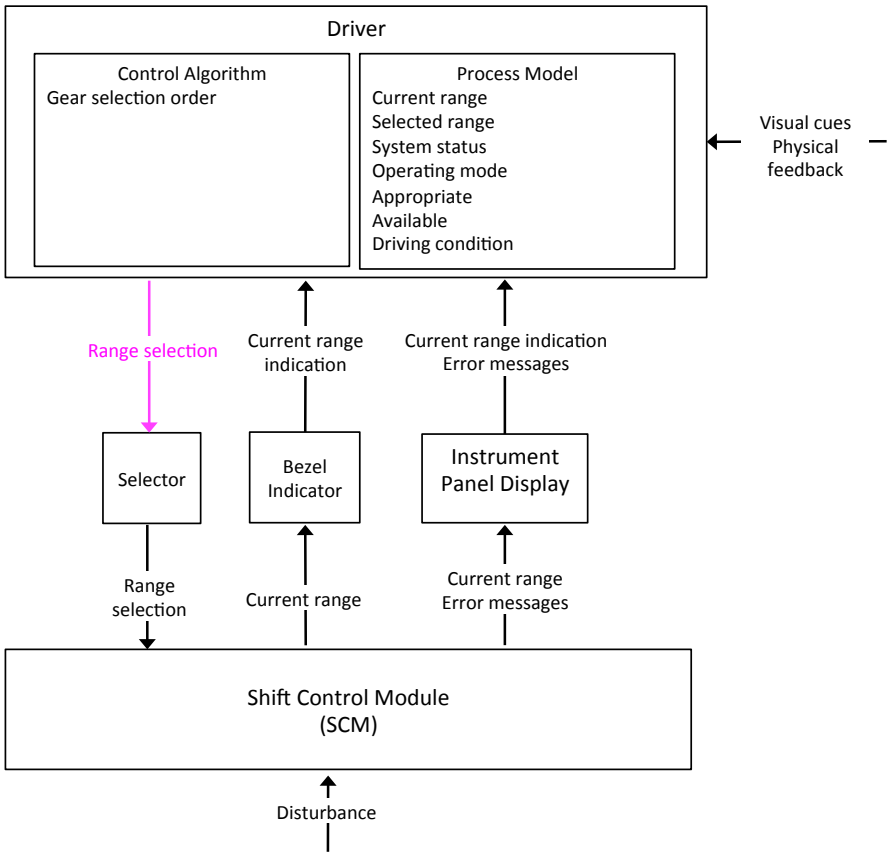


Figure 39 Driver control loop

Causal scenarios for **UCA-DR-1**: Driver does not provide new range selection when appropriate.

> The vehicle moves in the wrong direction or is not secured because the driver does not select an appropriate range for the operating mode, driving condition, and system status. This could be caused by:
>
> **S-DR-1**: The driver's process model incorrectly believes the current range is appropriate due to:
>
> CF-1: Receiving conflicting current range feedback from the Bezel and Instrument Panel (information is not matched) [↑UCA-DR-1]

SC-17: Current range indication shall be consistent or indicate to the driver when an error has occurred for all display sources [←S-DR-1, ↑SC-7]

CF-2: Current range feedback information is matched from both sources but not correct because the source of the current range is incorrect [↑UCA-DR-1]

SC-18: Current range information source shall guarantee correct range or indicate an error has occurred [←S-DR-1, ↑SC-7]

CF-3: Current range feedback information is not available from bezel or instrument panel due to the lack of power for both displays [↑UCA-DR-1]

SC-19: Current range indicators shall have separate means to provide current range [←S-DR-1, ↑SC-7]

CF-4: Current range feedback indicator confused with other vehicle indicator or view is obstructed -layout of indicators not done correctly [↑UCA-DR-1]

SC-20: Current range indicator shall be unobstructed for a seat belted driving position and separated and distinct from other indicators [←S-DR-1, ↑SC-7]

CF-80: Receiving conflicting current range feedback from the Bezel and Instrument Panel (information is not matched) [↑UCA-DR-1]

SC-82: Current range indication shall be consistent or indicate to the driver when an error has occurred for all display sources [←S-DR-1, ↑SC-7]

CF-82: Current range feedback information is matched from both sources but not correct because the source of the current range is incorrect [↑UCA-DR-1]

SC-84: Current range information source shall guarantee correct range or indicate an error has occurred [←S-DR-1, ↑SC-7]

CF-83: Current range feedback information is not available from bezel or instrument panel due to the lack of power for both displays [↑UCA-DR-1]

SC-85: Current range indicators shall have separate means to provide current range [←S-DR-1, ↑SC-7]

S-DR-2: The driver's process model incorrectly believes the appropriate range is unavailable and does not know what the correct action to take due to:

CF-5: Receives a confusing error message feedback [↑UCA-DR-1]

SC-21: Error message shall specify the ranges that are not available [←S-DR-2, ↑SC-7]

CF-6: Error message feedback needed for clarification of situation is missing [↑UCA-DR-1]

SC-22: Error message shall specify the what action to take for the ranges that are not available [←S-DR-2, ↑SC-7]

CF-7: Process model to handle specific error message incorrect or missing (i.e. Park not available therefore apply parking brake and turn off ignition) [↑UCA-DR-1]

SC-23: Error message shall specify actions to take that do not require special sequence of controls [←S-DR-2, ↑SC-7]

CF-3: Error message display is not working (broken, loss of power) [↑UCA-DR-1]

SC-19: Current range indicators shall have separate means to provide current range [←S-DR-2, ↑SC-7]

CF-83: Error message display is not working (broken, loss of power) [↑UCA-DR-1]

SC-85: Available ranges shall be indicated by all displays [←S-DR-2, ↑SC-7]

S-DR-3: The driver's process model incorrectly believes the appropriate range will be automatically selected by the system due to:

CF-7: Process model variable incorrect or missing about vehicle operation [↑UCA-DR-1]

SC-23: Error message shall specify actions to take that do not require special sequence of controls [←S-DR-2, ↑SC-7]

CF-10: External feedback indicates correct range has been selected (i.e. parked on a level surface and vehicle does not move when brake is released) [↑UCA-DR-1]

SC-24: Driver shall be informed when an inappropriate range is selected for the vehicle operation [←S-DR-3, ↑SC-7]

Causal scenarios for when the Driver range selection is appropriate but not followed.

The vehicle moves in the wrong direction or is not secured because the driver selects a safe range but the range is not selected. This could be caused by:

S-DR-4: Driver's Control Algorithm flaw resulting in the selector being incorrectly used and the selected range is not seen as an input possibly related to:

CF-11: Partial application of the input (did not engage the device fully) [↑H-1, ↑H-2]

SC-25: The selection of a gear shall indicate some confirmation to the driver [←S-DR-4, ↑SC-1, ↑SC-2]

CF-12: Incorrect timing associated with selecting the range (released too quickly or provided too long) [↑H-1, ↑H-2]

SC-26: The selection of a gear shall indicate some confirmation to the driver [←S-DR-4, ↑SC-1, ↑SC-2]

S-DR-5: The Selector does not send the signal due to:

CF-13: The input is not recognized due to external interference (EMI) [↑H-1, ↑H-2]

SC-27: The selector shall be operational within specified EMI requirements [←S-DR-5, ↑SC-1, ↑SC-2]

CF-14: The input is not recognized due to internal interference [↑H-1, ↑H-2]

SC-28: The selector shall be operational within specified EMI requirements [←S-DR-5, ↑SC-1, ↑SC-2]

CF-15: The selector has stop operating due to no/low voltage, durability, or temperature [↑H-1, ↑H-2]

SC-29: The selector shall operate within the specified operating conditions of the vehicle [←S-DR-5, ↑SC-1, ↑SC-2]

CF-16: The input is ignored because the selector cannot accept any inputs in its current mode (diagnostics, power-saving, etc.) [↑H-1, ↑H-2]

SC-30: The selector shall allow selection of specific ranges during any mode [←S-DR-5, ↑SC-1, ↑SC-2]

S-DR-6: The SCM receives the signal but does not send the signal due to:

CF-17: The signal is not recognized due to external interference (EMI) [↑H-1, ↑H-2]

SC-31: The SCM shall be operational within specified EMI requirements [←S-DR-6, ↑SC-1, ↑SC-2]

CF-18: The signal is not recognized due to internal interference [↑H-1, ↑H-2]

SC-32: The SCM shall be operational within specified EMI requirements [←S-DR-6, ↑SC-1, ↑SC-2]

CF-19: The SCM has stop operating due to no/low voltage, durability, temperature [↑H-1, ↑H-2]

SC-33: The SCM shall operate within the specified operating conditions of the vehicle [←S-DR-6, ↑SC-1, ↑SC-2]

CF-20: The SCM is in an operating mode that it ignores the input [↑H-1, ↑H-2]

SC-34: The SCM shall be able to provide the range for in any operating mode or only move to a mode when the appropriate range is the current range [←S-DR-6, ↑SC-1, ↑SC-2]

S-DR-7: The SCM received a delayed signal but does not send the signal due to:

CF-21: The range is no longer appropriate for the operating mode [↑H-1, ↑H-2]

SC-35: The SCM shall be able to provide the range for in any operating mode or only move to a mode when the appropriate range is the current range [←S-DR-7, ↑SC-1, ↑SC-2]

CF-22: The range has become unavailable [↑H-1, ↑H-2]

SC-36: The driver shall be informed when the selected range is not available [←S-DR-7, ↑SC-1, ↑SC-2]

Causal scenarios for **UCA-DR-2:** Driver provides range selection when the range selected is not appropriate.

The vehicle moves in the wrong direction or is not secured because an inappropriate range was selected. This could be caused by:

S-DR-8: The input to the selector is incorrect due to:

CF-23: The driver or something/someone else in the vehicle accidentally, i.e. input bumped. [↑UCA-DR-2]

SC-37: The selector shall have a means to confirm intended selections [←S-DR-8, ↑SC-9]

CF-24: The selector is awkward to use due to poor design [↑UCA-DR-2]

SC-38: The selector shall be designed following the specified ergonomic requirements [←S-DR-8, ↑SC-9]

S-DR-9: The driver's Control Algorithm flaw causes the incorrect range to be selected due to:

CF-25: Incorrect direction for range selection (the driver does not know how to use the shifter) [↑UCA-DR-2]

SC-39: The sequence of operation shall be indicated next to the shifter [←S-DR-9, ↑SC-9]

CF-26: Incorrect sequence to select range (the sequence is non-standard, different than conventional systems and not intuitive) [↑UCA-DR-2]

SC-40: The sequence of operation shall be indicated next to the shifter [←S-DR-9, ↑SC-9]

CF-12: Incorrect application of input based on timing [↑UCA-DR-2]

SC-26: The selection of a gear shall indicate some confirmation to the driver [←S-DR-9, ↑SC-9]

S-DR-10: The driver's process model incorrectly believes the selected range is appropriate due to:

CF-28: Missing or flawed process model variable regarding the appropriate range [↑UCA-DR-2]

SC-41: The driver shall be indicated if the appropriate range can be determined [←S-DR-10, ↑SC-9]

CF-29: Missing or flawed process model variable regarding the operating mode of the vehicle [↑UCA-DR-2]

SC-42: The driver shall be indicated if the appropriate range can be determined [←S-DR-10, ↑SC-9]

Causal scenarios for **UCA-DR-3:** Driver provides range selection when the range selected is not available.

The vehicle moves in the wrong direction or is not secured because an unavailable range was selected. This could be caused by:

S-DR-11: The driver's process model incorrectly believes the selected range is available due to:

CF-5: Error message feedback regarding available ranges is confusing [↑UCA-DR-3]

SC-21: Error message shall specify the ranges that are not available [←S-DR-11, ↑SC-10]

CF-31: Error message feedback needed for clarification of situation is missing [↑UCA-DR-3]

SC-43: Error message shall specify the what action to take for the ranges that are not available [←S-DR-11, ↑SC-10]

CF-7: Process model to handle specific error message incorrect or missing (i.e. Park not available therefore apply parking brake and turn off ignition) [↑UCA-DR-3]

SC-23: Error message shall specify actions to take that do not require special sequence of controls [←S-DR-11, ↑SC-10]

CF-33: The bezel range feedback does not correctly display the available ranges [↑UCA-DR-3]

SC-44: The bezel indicator shall display the available ranges based on the current range data [←S-DR-11, ↑SC-10]

CF-34: The bezel range feedback is confusing regarding the missing ranges [↑UCA-DR-3]

SC-45: The bezel indicators shall clearly indicate which ranges are available or not [←S-DR-11, ↑SC-10]

CF-35: The available range process model variable is incorrect from both the bezel and instrument display (same source) [↑UCA-DR-3]

SC-46: Available range indication shall be consistent or indicate to the driver when an error has occurred for all display sources [←S-DR-11, ↑SC-9]

CF-3: The available range feedback is missing from the bezel and the instrument display or cannot be displayed due to no/low voltage or a broken component [↑UCA-DR-3]

SC-19: Available range indicators shall have separate means to provide available range [←S-DR-11, ↑SC-10]

CF-37: The bezel and instrument panel display conflicting available range feedback [↑UCA-DR-3]

SC-48: Available range indication shall be consistent or indicate to the driver when an error has occurred for all display sources [←S-DR-11, ↑SC-10]

CF 39: The error message feedback for the available range is delayed [↑UCA-DR-3]

SC-49: The error message data shall be available within XX of being valid [←S-DR-11, ↑SC-10]

CF-10: External feedback (vehicle does not move when brake is released) indicates the range was selected but was not [↑UCA-DR-3]

SC-24: Driver shall be informed when an inappropriate range is selected for the vehicle operation [←S-DR-11, ↑SC-10]

CF-6: The error message feedback for the available range is delayed [↑UCA-DR-3]

SC-22: Error message shall specify the what action to take for the ranges that are not available [←S-DR-11, ↑SC-10]

CF-83: The available range feedback is missing from the bezel and the instrument display or cannot be displayed due to no/low voltage or a broken component [↑UCA-DR-3]

SC-85: Available range indicators shall have separate means to provide available range [←S-DR-11, ↑SC-10]

CF-85: The bezel and instrument panel display conflicting available range feedback [↑UCA-DR-3]

SC-87: Available range indication shall be consistent or indicate to the driver when an error has occurred for all display sources [←S-DR-11, ↑SC-10]

CF-88: The error message feedback for the available range is delayed [↑UCA-DR-3]

SC-90: The error message data shall be available within XX of being valid [←S-DR-11, ↑SC-10]

Causal scenarios for **UCA-DR-4:** Driver provides range selection too late when the range selected is appropriate.

The vehicle moves in the wrong direction or is not secured because the appropriate range was selected too late. This could be caused by:

S-DR-12: The driver's process model incorrectly believes the current operating mode of the vehicle due to:

CF-10: External feedback (vehicle does not move when brake is released) indicates the range was selected but was not. [↑UCA-DR-4]

SC-24: Driver shall be informed when an inappropriate range is selected for the vehicle operation [←S-DR-12, ↑SC-11]

CF-41: Recognition of the external feedback about the vehicle condition was delayed [↑UCA-DR-4]

SC-50: Driver shall be informed when an inappropriate range is selected for the vehicle operation [←S-DR-12, ↑SC-11]

CF-29: Current operating mode process model is incorrect [↑UCA-DR-4]

SC-42: The driver shall be indicated if the appropriate range can be determined [←S-DR-12, ↑SC-11]

S-DR-13: The driver's process model for the appropriate range was delayed to update:

CF-43: Current range feedback was delayed from the Bezel and Instrument Panel [↑UCA-DR-4]

SC-51: The current range shall be available within XX of being valid [←S-DR-13, ↑SC-11]

CF-1: Current range feedback was delayed from the Bezel and Instrument Panel [↑UCA-DR-4]

SC-17: Current range indication shall be consistent or indicate to the driver when an error has occurred for all display sources [←S-DR-13, ↑SC-11]

CF-45: Current range feedback was missing from the Bezel and Instrument Panel [↑UCA-DR-4]

SC-52: Current range indicators shall have separate means to provide current range [←S-DR-13, ↑SC-11]

CF-81: Receiving conflicting current range feedback from the Bezel and Instrument Panel (information is not matched) [↑UCA-DR-4]

SC-83: Current range indication shall be consistent or indicate to the driver when an error has occurred for all display sources [←S-DR-13, ↑SC-11]

CF-86: Current range feedback information is not available from bezel or instrument panel due to the lack of power for both displays [↑UCA-DR-4]

SC-88: Current range indicators shall have separate means to provide current range [←S-DR-13, ↑SC-11]

CF-89: Current range feedback was delayed from the Bezel and Instrument Panel [↑UCA-DR-4]

SC-91: The current range shall be available within XX of being valid [←S-DR-13, ↑SC-11]

S-DR-14: The driver's selection was delayed due to a control algorithm is flawed related to:

CF-25: The correct operation to select the gear [↑UCA-DR-4]

SC-39: The sequence of operation shall be indicated next to the shifter [←S-DR-14, ↑SC-11]

CF-12: Timing delay in operating the selector (long time to select the correct range) [↑UCA-DR-4]

SC-26: The sequence of operation shall be indicated next to the shifter [←S-DR-14, ↑SC-11]

Shifter Control Module (SCM)

Identify Unsafe Control Actions (Step 1)

Table 13 Unsafe control actions for SCM

Control Action	Not Provided	Provided	Too early/too late/wrong order	Stopped too soon/ applied too long
Range Command	**UCA-SCM-1**: Shifter Control Module does not provide range command when driver selects an appropriate and available range [↑H-1, ↑H-2, ↑H-3]	**UCA-SCM-2:** Shift Control Module provides range command when the range is not appropriate [↑H-1, ↑H-2, ↑H-3] **UCA-SCM-3:** Shift Control Module provides range command when that range is not available [↑H-1, ↑H-2, ↑H-3]	**UCA-SCM-4:** Shifter Control Module provides range command too late for an appropriate and available range [↑H-1, ↑H-2, ↑H-3]	N/A

Safety Constraints from Step 1 for the SCM

The safety constraints for the SM based on the unsafe control actions are:

SC-12: Shifter Control Module must provide range command when driver selects an appropriate and available range [←UCA-SCM-1, ↑SC-1, ↑SC-2, ↑SC-3, ↑SC-4, ↑SC-5**]**

SC-13: Shift Control Module must not provide range command when the range is not appropriate [←UCA-SCM-2, ↑SC-1, ↑SC-2, ↑SC-3, ↑SC-4, ↑SC-5**]**

SC-14: Shift Control Module must not provide range command when the range is not available [←UCA-SCM-3, ↑SC-1, ↑SC-2, ↑SC-3, ↑SC-4, ↑SC-5**]**

SC-15: Shifter Control Module must provide range command by XX when the range is appropriate and available [←UCA-SCM-4, ↑SC-1, ↑SC-2, ↑SC-3, ↑SC-4, ↑SC-5**]**

Determine How Unsafe Control Actions Occur (Step 2)

Shifter Control Module

Determine how each UCA from Table 13 can occur for the range command control action and the process model variables shown in Figure 40.

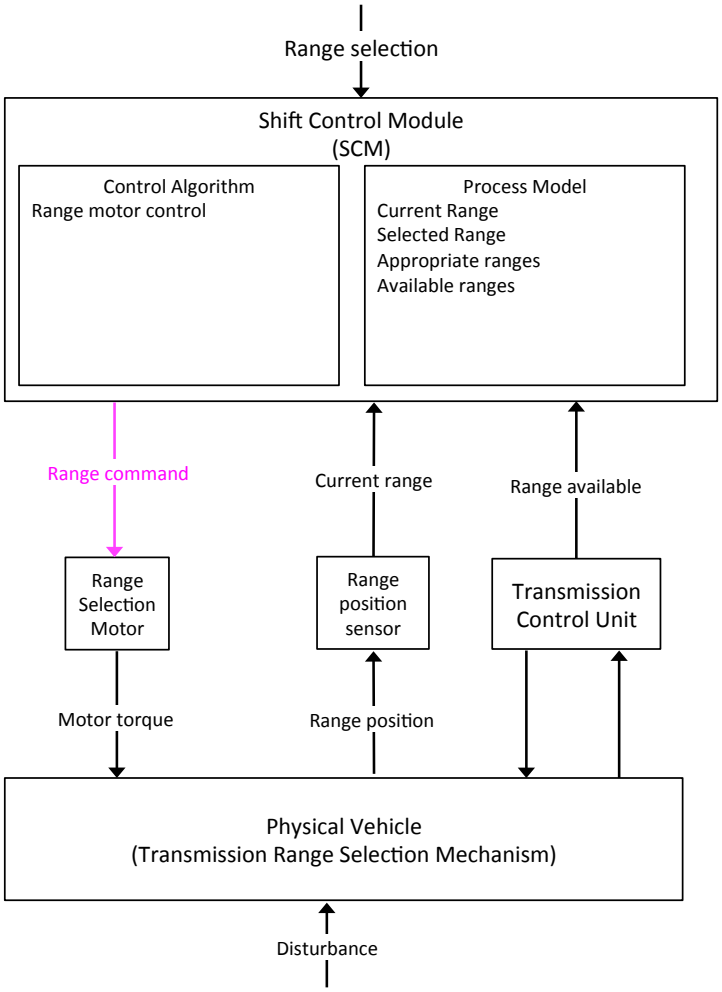


Figure 40 SBW control loop

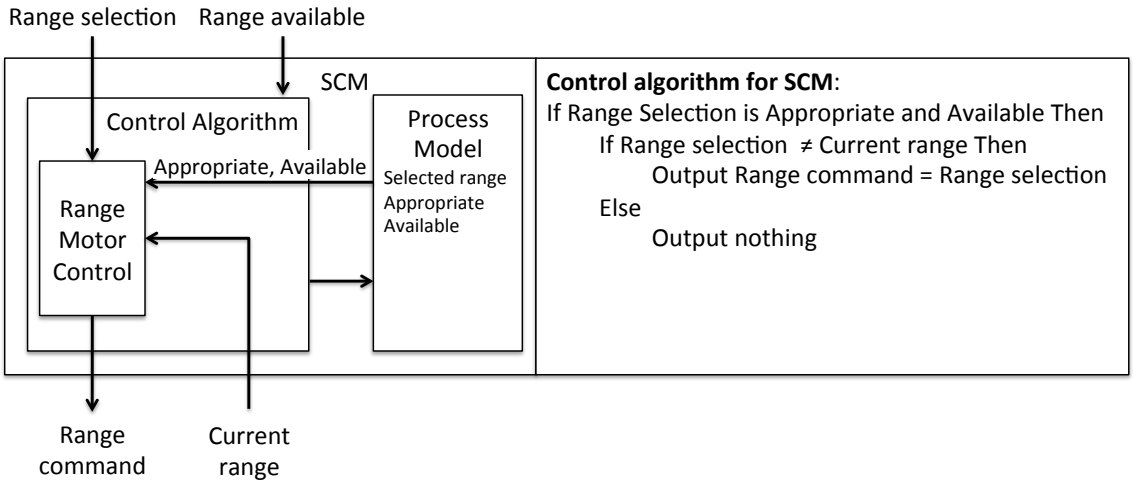


Figure 41 SBW control algorithm for Step 2

Causal Scenarios for **UCA-SCM-1**: Shifter Control Module does not provide range command when driver selects an appropriate and available range.

The vehicle moves in the wrong direction or is not secured because the appropriate range was not sent for the vehicle operating conditions. This could be caused by:

S-SCM-1: The SCM's process model believes the selected range is not appropriate due to:

CF-48: Appropriate range process model variable flaw (missing information or incorrect) [↑UCA-SCM-1]

SC-53: The SCM process model for the appropriate range shall be consistent with operating mode and situation [←S-SCM-1, ↑SC-12**]**

CF-50: Available range feedback is incorrect [↑UCA-SCM-1]

SC-55: The appropriate range feedback shall be verified from multiple sources [←S-SCM-1, ↑SC-12**]**

S-SCM-2: The SCM's process model believes the selected range is not available due to:

CF-49: Available ranges Process model variable flaw (missing information or incorrect) [↑UCA-SCM-1]

SC-54: The SCM process model for the available range shall be consistent with available range data [←S-SCM-2, ↑SC-12**]**

CF-79: Range is available but feedback indicates otherwise [↑UCA-SCM-1]

SC-81: The available range feedback shall not inhibit the appropriate range selection attempt [←S-SCM-2, ↑SC-12**]**

S-SCM-3: The SCM's control algorithm does not output selected range due to:

CF-51: Incorrect current range feedback indicating current range has been met [↑UCA-SCM-1]

SC-56: The current range feedback shall be confirmed from more than one common source [←S-SCM-3, ↑SC-12**]**

CF-52: Control algorithm flaw that causes no output to be issued [↑UCA-SCM-1]

SC-57: System shall provide a safe state when appropriate for vehicle's operating condition [←S-SCM-3, ↑SC-12**]**

CF-90: Range position sensor not working (no current range feedback)

SC-92: The current range position sensor should be checked for operation [←S-SCM-3, ↑SC-12**]**

CF-91: The SCM does not output the command because there was no power after the driver shut off the ignition [↑UCA-SCM-1]

SC-93: Power should be available to the SCM to provide range commands for safe range after ignition power is shut off [←S-SCM-3, ↑SC-12**]**

S-SCM-4: The SCM's process model believes there is no selected range due to:

CF-53: Selected range process model flaw [↑UCA-SCM-1]

SC-59: The process model shall be consistent with driver selection [←S-SCM-4, ↑SC-12**]**

CF-54: Selected range input was not received [↑UCA-SCM-1]

SC-60: The SCM shall provide the appropriate range for the vehicle operating condition [←S-SCM-4, ↑SC-12**]**

Causal Scenarios for when the Shifter Control Module range command is safe but not followed.

The vehicle moves in the wrong direction or is not secured because the selected range that is appropriate for the vehicle operating conditions was not followed. This could be caused by:

S-SCM-5: The range selection motor does not send the range command due to:

CF-55: The range command is not recognized due to interference [↑H-1, ↑H-2]

SC-61: The motor shall meet the requirements for EMI specified [←S-SCM-5, ↑SC-1, ↑SC-2]

CF-56: The range selection motor does not operate (loss of power or broken) [↑H-1, ↑H-2]

SC-62: The system shall have a method to move to a safe state [←S-SCM-5, ↑SC-1, ↑SC-2]

CF-57: The range command does not match the expected format of the range motor (interface problem) [↑H-1, ↑H-2]

SC-63: Range command shall be specified to match range motor [←S-SCM-5, ↑SC-1, ↑SC-2]

S-SCM-6: The motor torque supplied by the motor is insufficient due to:

CF-58: The range motor torque was not specified correctly (design error) [↑H-1, ↑H-2]

SC-64: The range motor torque shall be appropriate for the load [←S-SCM-6, ↑SC-1, ↑SC-2]

S-SCM-7: The range selection mechanism (controlled process) receives the motor torque but the range is not selected due to:

CF-59: The torque required for the range selection mechanism is higher than specified [↑H-1, ↑H-2]

SC-65: The SCM shall take appropriate action when specified range not engaged [←S-SCM-7, ↑SC-1, ↑SC-2]

CF-60: The range selection mechanism changes but has no impact because the transmission is broken (the range is not available) [↑H-1, ↑H-2]

SC-66: The SCM shall indicate to Driver system is not operating [←S-SCM-7, ↑SC-1, ↑SC-2]

CF-61: The Transmission Control Unit sends a conflicting command that overrides the range command [↑H-1, ↑H-2]

SC-67: The TCU and SCM commands shall have a specified priority [←S-SCM-7, ↑SC-1, ↑SC-2]

CF-62: The transmission is locked into a range for maintenance [↑H-1, ↑H-2]

SC-68: The system shall indicate when transmission is in maintenance mode where range cannot be changed [←S-SCM-7, ↑SC-1, ↑SC-2]

Causal Scenarios for **UCA-SCM-2:** Shift Control Module provides range command when the range is not appropriate.

The vehicle moves in the wrong direction or is not secured because an inappropriate range was sent for the vehicle operating conditions. This could be caused by:

S-SCM-8: The SCM's process model incorrectly believed that the selected range was appropriate due to:

CF-48: The range appropriate process model variable is flawed (missing or incorrect) [↑UCA-SCM-2]

SC-53: The SCM process model for the appropriate range shall be consistent with operating mode and situation [←S-SCM-8, ↑SC-12]

CF-64: The feedback for range appropriate is missing [↑UCA-SCM-2]

SC-69: The SCM shall not provided a range when appropriate range cannot be determined [←S-SCM-8, ↑SC-12]

S-SCM-9: The SCM's control algorithm is flawed resulting in:

CF-65: The range selected is always judged as appropriate [↑UCA-SCM-2]

SC-70: The SCM shall verify the appropriate range [←S-SCM-9, ↑SC-12]

CF-66: The incorrect range command is sent (does not match input) [↑UCA-SCM-2]

SC-71: The SCM shall verify the range command is appropriate [←S-SCM-9, ↑SC-12]

Causal scenarios for **UCA-SCM-3:** Shift Control Module provides range command when that range is not available.

The vehicle moves in the wrong direction or is not secured because an unavailable range was sent. This could be caused by:

S-SCM-10: The SCM's process model incorrectly believed that the selected range was available due to:

CF-49: The range available process model variable is flawed [↑UCA-SCM-3]

SC-54: The SCM process model for the available range shall be consistent with available range data [←S-SCM-10, ↑SC-13]

CF-68: The feedback for range available from the TCU was not sent [↑UCA-SCM-3]

SC-73: The SCM shall identify when the available range data is not sent [←S-SCM-10, ↑SC-13]

CF-50: The feedback for range available from the TCU was incorrect (interference or TCU error) [↑UCA-SCM-3]

SC-55: The appropriate range feedback shall be verified from multiple sources [←S-SCM-10, ↑SC-13]

CF-70: The feedback for range available from the TCU was delayed [↑UCA-SCM-3]

SC-74: Available range indications shall be transmitted every XX ms [←S-SCM-10, ↑SC-13]

S-SCM-11: The SCM's control algorithm is flawed resulting in:

CF-71: The range selected is always judged as available [↑UCA-SCM-3]

SC-75: The SCM shall verify the available range [←S-SCM-11, ↑SC-13]

CF-66: The incorrect range command is sent [↑UCA-SCM-3]

SC-72: The SCM shall verify the range command is available [←S-SCM-11, ↑SC-13]

Causal Scenarios for **UCA-SCM-4:** Shifter Control Module provides range command too late for an appropriate and available range.

The vehicle moves in the wrong direction or is not secured because the range was sent too late for the vehicle operating conditions. This could be caused by:

S-SCM-12: The input to the SCM related to:

CF-73: The selected range input is delayed from the selector [↑UCA-SCM-4]

SC-76: The selected range input shall be transmitted form the selector by XX ms [←S-SCM-12, ↑SC-14]

S-SCM-13: The SCM's control algorithm is flawed due to:

CF-74: The time to process a new input takes too long [↑UCA-SCM-4]

SC-77: The SCM shall process new inputs within XX ms [←S-SCM-13, ↑SC-14]

CF-75: The time to output the new command takes too long [↑UCA-SCM-4]

SC-78: The SCM shall provide new range command within xx ms of recpetion [←S-SCM-13, ↑SC-14]

S-SCM-14: The SCM's process model incorrectly believes the selected range is not appropriate due to:

CF-52: Delay in the feedback (takes too long to update based on the operation) [↑UCA-SCM-4]

SC-57: System shall provide a safe state when appropriate for vehicle's operating condition [←S-SCM-14, ↑SC-14]

S-SCM-15: The SCM's process model incorrectly believes the selected range is not available due to:

CF-70: Delay in the available range feedback caused by TCU processing [↑UCA-SCM-4]

SC-74: Available range indications shall be transmitted every XX ms [←S-SCM-15, ↑SC-14]

CF-78: Delay in the available range feedback caused by the transmission method [↑UCA-SCM-4]

SC-80: The transmission delay from the TCU shall be no more than XX ms [←S-SCM-15, ↑SC-14]

CF-17: The signal is not recognized due to external interference (EMI) [↑UCA-SCM-4]

SC-31: The SCM shall be operational within specified EMI requirements [←S-SCM-15, ↑SC-14]

CF-18: The signal is not recognized due to internal interference [↑UCA-SCM-4]

SC-32: The SCM shall be operational within specified EMI requirements [←S-SCM-15, ↑SC-14]

This page intentionally left blank.

Appendix B: Requirements Traceability Matrix

	STPA																																										
Safety Constraint	H-1.1	H-1.2	H-3.1	H-3.2	H-3.3	H-4	UCA-DR-1	UCA-DR-2	UCA-DR-3	UCA-DR-4	UCA-SCM-1	UCA-SCM-2	UCA-SCM-3	UCA-SCM-4	S-DR-1	S-DR-2	S-DR-3	S-DR-4	S-DR-5	S-DR-6	S-DR-7	S-DR-8	S-DR-9	S-DR-10	S-DR-11	S-DR-12	S-DR-13	S-DR-14	S-SCM-1	S-SCM-2	S-SCM-3	S-SCM-4	S-SCM-5	S-SCM-6	S-SCM-7	S-SCM-8	S-SCM-9	S-SCM-10	S-SCM-11	S-SCM-12	S-SCM-13	S-SCM-14	S-SCM-15
SC-1	X																																										
SC-2		X																																									
SC-3			X																																								
SC-4				X																																							
SC-5					X																																						
SC-6						X																																					
SC-7							X																																				
SC-9								X																																			
SC-10									X																																		
SC-11										X																																	
SC-12											X																																
SC-14												X																															
SC-15													X																														
SC-16														X																													
SC-17															X												X																
SC-18															X																												
SC-19															X										X																		
SC-20															X																												
SC-21																X									X																		
SC-22																X																											
SC-23																X	X																										
SC-24																	X								X	X																	
SC-25																		X																									
SC-26																		X					X					X															
SC-27																			X																								
SC-28																			X																								
SC-29																			X																								
SC-30																			X																								
SC-31																				X																							X
SC-32																				X																							X
SC-33																				X																							
SC-34																				X																							
SC-35																					X																						
SC-36																					X																						
SC-37																						X																					
SC-38																						X																					
SC-39																							X					X															
SC-40																							X																				
SC-41																								X																			
SC-42																								X		X																	
SC-43																									X																		
SC-44																									X																		
SC-45																									X																		
SC-46																									X																		
SC-47																									X																		
SC-48																									X																		
SC-49																									X																		
SC-50																										X																	
SC-51																											X																
SC-52																											X																
SC-53																													X							X							
SC-54																														X								X					
SC-55																													X														
SC-56																															X												
SC-57																															X												
SC-59																																X											
SC-60																																	X										
SC-61																																	X										
SC-62																																	X										
SC-63																																		X									
SC-64																																			X								
SC-65																																			X								
SC-66																																			X								
SC-67																																			X								
SC-68																																			X								
SC-69																																				X							
SC-70																																					X						
SC-71																																					X						
SC-72																																							X				
SC-73																																						X					
SC-74																																						X					X
SC-75																																							X				
SC-76																																								X			
SC-77																																									X		
SC-78																																									X		
SC-79																																										X	
SC-80																																											X
SC-81																														X													
SC-82															X																												
SC-83																											X																
SC-84															X																												
SC-85															X	X									X																		
SC-86															X																												
SC-87																									X																		
SC-88																											X																
SC-90																									X																		
SC-91																											X																
SC-92																															X												
SC-93																															X												

	Control Strcuture																											
Safety Constraint	Driver	Driver Process Model	Driver Control Algorithm	Visual ques	Range request	Selector	Range selection	Bezel Indicator	Cuurent range indication	Error message	Instrument Panel Display	Curent range display	Error message display	SCM	SCM Process Model	SCM Control Algorithm	Range command	Range Selection Motor	Motor torque	Range Selection	Other command	Current range	Range Posiion Sensor	Appropriate range	TCU	Available ranges	Other vehicle controls	Physical feedback
SC-1																												
SC-2																												
SC-3																												
SC-4																												
SC-5																												
SC-6																												
SC-7	X	X			X																							
SC-9	X	X			X																							
SC-10	X	X			X																							
SC-11	X	X			X																							
SC-12														X	X		X											
SC-14														X	X		X											
SC-15														X	X		X											
SC-16														X	X		X											
SC-17									X																			
SC-18									X																			
SC-19								X																				
SC-20								X																				
SC-21													X															
SC-22													X															
SC-23		X																										
SC-24				X																								X
SC-25			X																									
SC-26			X																									
SC-27					X																							
SC-28						X																						
SC-29						X																						
SC-30						X																						
SC-31						X																						
SC-32						X																						
SC-33														X														
SC-34																X												
SC-35							X																					
SC-36							X																					
SC-37						X																						
SC-38						X																						
SC-39			X																									
SC-40			X																									
SC-41		X																										
SC-42		X																										
SC-43													X															
SC-44									X																			
SC-45									X																			
SC-46																						X						
SC-47																						X						
SC-48								X																				
SC-49										X																		
SC-50																												
SC-51									X																			
SC-52									X																			
SC-53															X													
SC-54															X													
SC-55																								X				
SC-56																						X						
SC-57																X												
SC-59															X													
SC-60							X																					
SC-61																	X											
SC-62																		X										
SC-63																	X											
SC-64																			X									
SC-65																			X									
SC-66																				X								
SC-67																					X				X			
SC-68																				X								
SC-69																								X				
SC-70																X												
SC-71																X												
SC-72																X												
SC-73																										X		
SC-74																										X		
SC-75																X												
SC-76							X																					
SC-77																X												
SC-78																X												
SC-79																								X				
SC-80																										X		
SC-81																									X	X		
SC-82												X																
SC-83												X																
SC-84												X																
SC-85											X																	
SC-86											X																	
SC-87											X																	
SC-88												X																
SC-90										X																		
SC-91												X																
SC-92																							X					
SC-93														X													X	

This page intentionally left blank.