Article can be accessed online at http://www.publishingindia.com

An Uml Based Safety Analysis for Safe Software

Bandaru Esther Sunanda*

Abstract

Software safety involves with the assurance that software will execute within a system context without resulting in unacceptable risk. The Safety-Critical Systems have become more prominent as there are being used in medical field and for monitoring purpose. To improve the effectiveness of the electronics systems embedded computer systems are used which includes the safety-critical software. While controlling the embedded computer systems with the software failures may occur. Due to the advances in the technologies and the usage of embedded computer systems the complexity may increase leading to risks. This paper aims at the description of the already the existing methods and metrics used for analysis of the developed embedded system and analyse its failures.An application called the Railroad Crossing Control System (RCCS) is used and UML is used as a technique to identify the hazards and they are rectified using UML diagrams with the safety issues.

Keywords: Safety-Critical Systems, RCCS, FMEA, UML.

1. Introduction

Safety critical system is a computer embedded system which involves the safety issues and prevents hazards (Stephen, 2003). Safety of the system depends on the working of the system components i.e. including software and hardware components. The safety of the safety critical systems depends on the inbuilt control systems. An example for the embedded control system is the railway signals which must make the trains go in right direction without accidents. However, there is always the option of stopping all trains if the integrity of the system becomes suspect. You can't just stop an aircraft while the fly by wire system is fixed! (Limerick, 2011) In medical field the people should be given proper dosages of radiations which depend on the safe working of the embedded systems. The safety of the software is involved regarding the suggestion of dosage of medicines for the patients as mentioned by the doctor. Both types of system can impact the safety of the patient. Even something as simple as traffic lights can be viewed as safety critical. In the process of signals, an incorrect signal may lead to collision of vehicles.

Need For Methods And Metrics: Safety critical systems are used everywhere around the world in nuclear plants, vehicles, banks, etc., When the systems with safety software are used the controlling of the systems must be done in a proper manner. (John C. Knight, 2002) The safety concepts must be dealt in the stages of design where extra care needs to be taken up to avoid the occurrence of accidents. In the design phase of the software development extra care need to be taken up for the safety of the development stages to control the hazards with the usage of certain rules and regulations. Though many things are considered for the safe software the happening of hazards didn't stop. (Limerick, 2011)When a hazard occurs the result of it may be very serious i.e. it leads to the death of the people which is called as catastrophic. Table 1 represents the list of (W.Eric, 2010) catastrophic accidents in the last decade.

In this paper, the second section of the paper describes the various methods used for the safety analysis. In the third section of the paper various analysis techniques are explained. In the fourth section of the paper the description of RCCS is explained followed by the next section where UML is described in brief and corresponding UML diagrams for RCCS are drawn.

* Department of Computer Science and Systems Engineering, Andhra University, Visakhapatnam, Andhra Pradesh, India. E-mail: bonnie.sunanda@gmail.com

Table 1. Catastrophic Accidents in the Last Decade

NAME OF ACCIDENT	YEAR	CAUSE	LEVEL OF DAMAGE
Railway signaling failure , Sydney	April 12 ,2011	The s/w known as ATRICS was unable to cope with flaky network.	Medium.
Saxony-Anhalt train accident, Germany	January 29, 2011	Software that detects traffic didn't work.	Medium.(20-30 died)
Collision of two trains, India (T1 : Local train , T2 : Indore-Gwalior Intercity Express)	20 September, 2010	The loco-pilot of the local train overshot the signal and points-man was unable to divide the loop line from the main line.	Medium.(40 died)
Goa Express & Mewar Express Accident, India.	21 October 2009	The signal was given green due to problem in signaling software.	Medium.(30 died)
Cedar Sinai Medical Centre in Los Angeles, California	August 2009-February 2008	Software misconfiguration in CT scanner used for brain perfusion scanning.	Very high.(wrong dosage to 206)
Crash of Air France Flight 447	May 31, 2009	Software related to the onboard automating reporting system transmitted several error messages.	Very high.(228 died)
Emergency-Shutdown of the Hatch Nuclear Power Plant	March 7, 2008	Installation of a software upgrade on a computer caused to reset the data on the control system.	High.(Loss of $5 million)

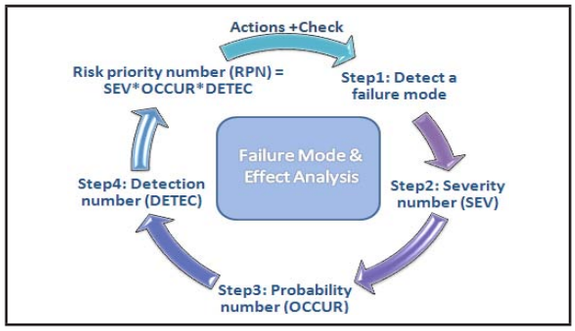
2. Safety Analysis Approaches

Safety-critical systems are the systems which will spoil the environment around and can be dangerous to the people around as they may lead to accidents if they fail. The analysis of the occurrence of accidents is not correct as it is done through manual procedures. There are methods(Stephen, 2003)to analyze the drawbacks of the safety-critical systems like Failure Mode and Effects Analysis (FMEA), Failure Modes Effects and Criticality Analysis (FMECA), Fault Tree Analysis (FTA), Event Tree Analysis (ETA), Failure Mode Factors and Effects Analysis (FMFEA), etc.,

2.1 Failure Mode and Effects Analysis (FMEA)

Failure Mode and Effects Analysis (FMEA) (Stephen, 2003) is an analytical method assurance of getting good qualitative results. It serves to find the potential failure of a product/process, to recognize and evaluate its importance and to identify appropriate actions to prevent the potential failure or to discover it in time. The systematic analysis and removal of weak Failure Mode and Effects Analysis (FMEA) is an analytical method of the preventive quality assurance. The failures are to identified in time and they need to be rectified based on their occurrence and they need to be modified in time. The failure points need to be reduced in a proper manner i.e. in a systematic manner. It is used to get the analysis of the risks that occur. The individual risks are weight against each other to recognize priorities. FMEA does not provide a statement on the total failure risk. For the analysis of failure combinations, the fault-tree analysis is more appropriate. The overall operations in FMEA are shown in the diagram 1.

Diagram 1. Steps of FMEA

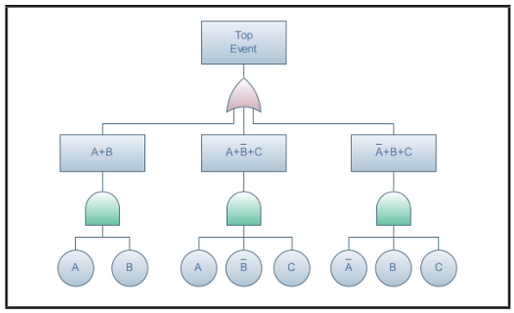


2.2 Fault Tree Analysis (FTA)

Fault Trees (Debra S. Herman, 2002) are the methods used for the analysis of a system and the get the probability of the failures. A Fault Tree is used for arranging the occurrence of events in the hierarchical order of their occurrence. It is used for analysing the various types of failures that may occur due to system failure, human failures, etc. The failure that occurs is represented at the top of the tree. A deductive analysis is used to analyse the failure which at the top of the tree and it provides the alternatives at the

end that result into the failures. Representation of a Fault Tree is given in diagram 2.

Diagram 2. Reprentation of a Fault Tree



3. Techniques for Analysis

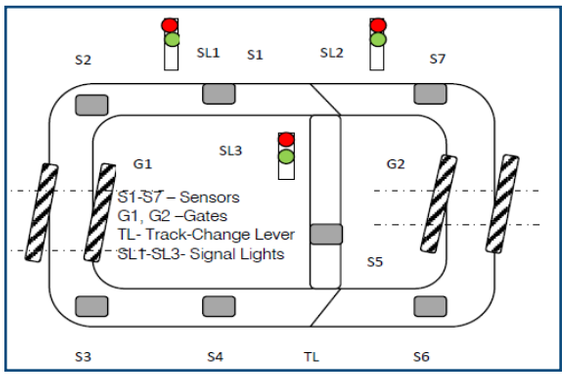
Graphical Requirement Analysis (GRA) generates logic bases graphical representation from functional requirements. Functional requirements (Debra S. Herman, 2002) are recognized by the GRA but are not in full detail, so they are not used fully for the safety analysis. Deductive Cause-Consequence Analysis (DCCA) takes the mathematical procedures to recognize the failures of a system. DCCA uses CTL and here the CTL uses the time requirements based on branching and FTA uses the time requirements based on linearity. FSSA is one the technique used for analyzing the system as well as the fault tolerance of the system in which the problem is present. UML can be used for the analysis of the system and can be combined with other methods for the analysis.

4. Case Study: Safety Analysis of Rccs

4.1 Railroad Crossing Control System (RCCS):

Safety-critical systems, by definition those systems whose failure can cause catastrophic results for people, the environment, and the economy, are becoming increasingly complex both in their functionality and their interactions with the environment. Here the safety-analysis of RCCS (Ben Swarup, 2011) is examined and an approach to avoid the accidents is specified. The working of RCCS is done by using the train which is powered by a power supply relay. Initially, when the power is on the train moves with the power provided to the wheels of the train. The train stops when power is not supplied. Sensors (See Diagram 3) which are present on the tracks sense the movement of the train. Altogether RCCS employs seven sensors. Two pairs of sensors detect the train position before and after the gates. Two sensors are present at the place where the track divides into two. Sensors are used for detecting the position of the train based on the position of the platform. Information from each of the sensors is passed to controller. An 8051 is used as a controller for RCCS. The controller continuously monitors the sensors and controls the gate actuators, track change lever, and the signal lights. RCCS has two sets of gates on either side of the track layout. The gate receives signals from the controller. When it receives lower command, arms of the gate moves down and closes the gate, preventing the road traffic at the intersection. When the gate receives raise, it moves up allowing the traffic to pass through. The gates are operated by means of a motor-based mechanism. RCCS contains three train signals, erected beside the track. One signal is at the platform to signal a halt at the platform. Other sensors are present at the place where the tracks meet.

Diagram 3. Block Diagram of RCCS



4.2 Modes Where Safety Hazards May Occur

1. There may be failure in the seven sensors placed at various positions.
2. There may be failure in the working system of the gates.
3. There may be failure in the working system of the lever which is used to change the track of the train.
4. There may be failure un the working condition of the signalling lights which allow the passage of the train on the track.
5. There may be problem in the working condition of the micro-controller that is used for the running of the system.

If these problems occur while the system is working then hazards may occur. To avoid the hazards a mechanism must be followed to identify and rectify them in the design phase itself. Techniques like FTA can be used in the design phase to identify the hazards that may occur. See FTA diagram 4 for the safety analysis of the RCCS.

There are some disadvantages while using the FTA i.e.

1. It involves a complicated process.
2. It requires considerable amount of time to complete.

So, the techniques like UML diagrams can be used to avoid the about complexities.

4. USING UML DIAGRAMS TO REPRESENT FAULTS:

The Fault trees may be difficult to understand and is difficult to represent every stage of the development. The hazard is represented by using a FTA diagram (Wikipedia) which may be better represented by using the UML diagrams. Advantages of UML:

- You know exactly what you are getting.
- You will have lower development costs.
- Your software will behave as you expect it to. Fewer surprises.
- The right decisions are made before you are given poorly written code. Less overall costs.
- We can develop more memory and processor efficient systems.
- System maintenance costs will be lower. Less re-learning takes place.
- Working with a new developer will be easier.
- Communication with programmers and outside contractors will be more efficient.

4.1 UML Diagrams of RCCS

4.1.1 Use Case Diagram

A use case represents the interactions of the activities going on in the systems. A use case is the activity that is going on in the system with the actors. System refers to the environment where the interactions are done. Use case diagrams (Grady Booch, 1999) are used in UML (Unified Modelling Language) to show how a person interacts with the system. Diagram 5 shows the Use case diagram for the RCCS. A use case diagram contains four components.

- The system specifies the area of interaction to which they are restricted to.
- The actors are the people who play various roles in the system.
- The use cases specify the various actions done by the actors.
- The relationships specify the interaction between the actors and use cases.

Diagram 4. Fault Tree for Gate1 Not Working

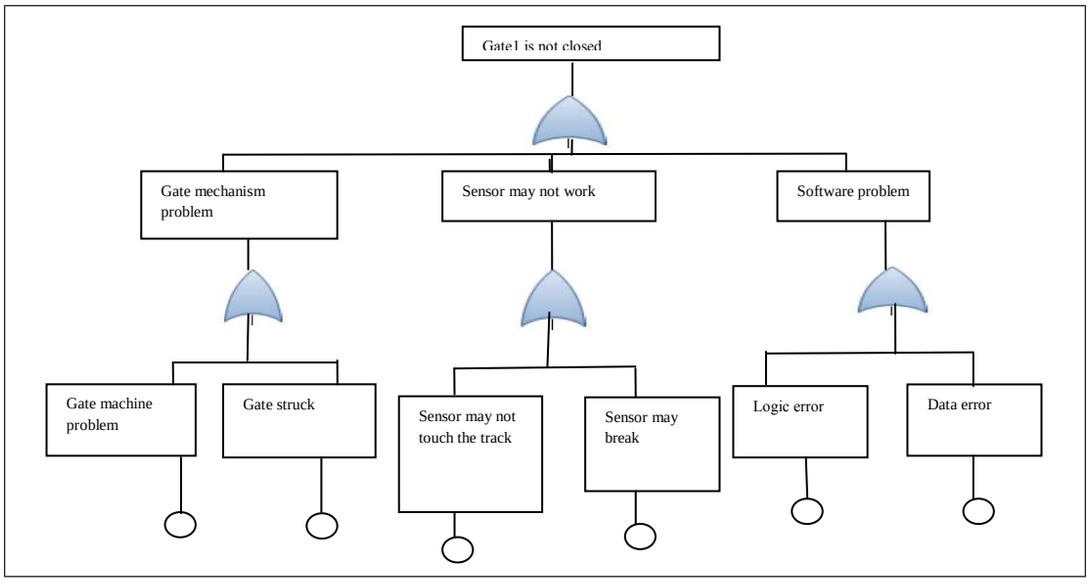


Diagram 5. Use Case Diagram of RCCS

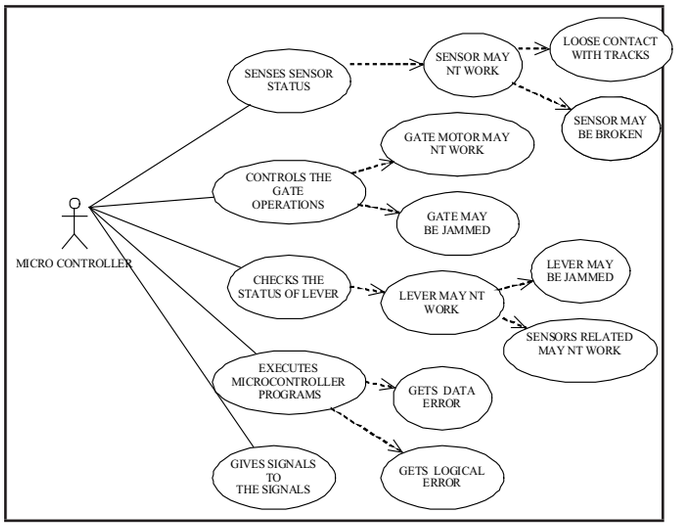


Diagram 6. Class Diagram for RCCS

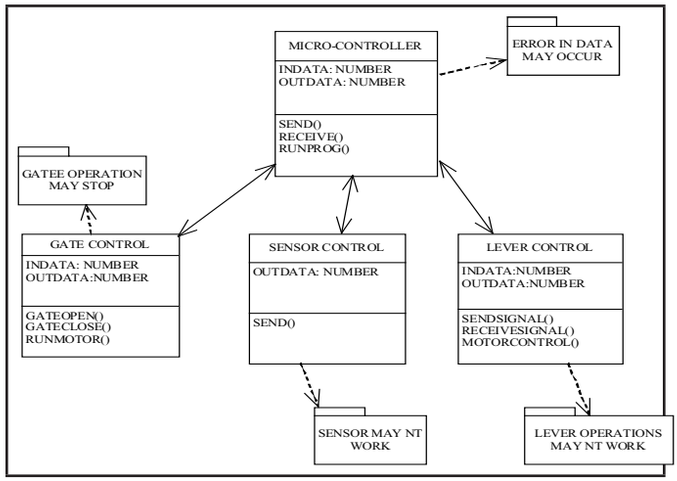


Diagram7. Sequence Diagram for RCCS

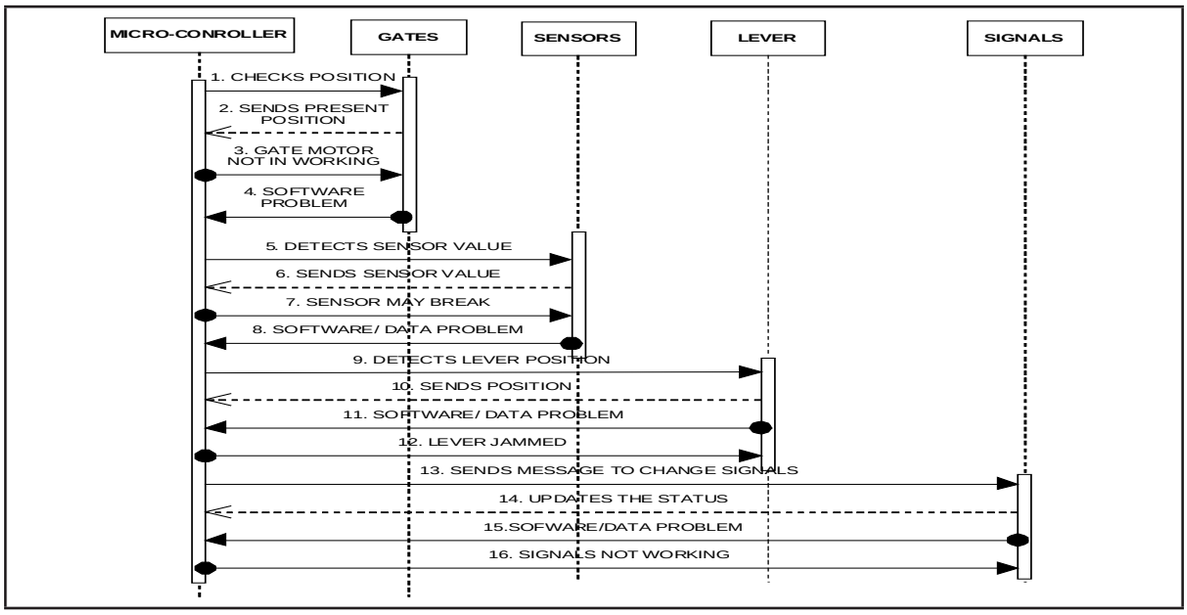


Table 2. Differences between FMEA and UML

Specifications	*Fmea*	*Uml*
Width of items	Applicable items are selected from functional diagrams	Applicable items are selected from system
Ability to detect potential problems	Limited to single failure	Applicable to all failures
Analysis safety in misuse	Cannot be done	Can be done
Flexibility	Reliability only	Reliability and safety
Starting point of analysis	Component failure mode	System design mode
Maintainability	Easy to maintain	Difficult to maintain
Scope for extension	Cannot be extensible	Can be extensible
Application suitability	Used at higher end	Used at lower end
Easy of analysis	Each component is to be analyzed	System is analyzed
Direction of analysis	Components to system(bottom to top)	Components to system(bottom to top)

4.1.2 Class Diagram

A class diagram is for representing the classes and their member variables and member functions and their dependencies." must be modified to "A class diagram is for representing the classes, their member variables, member functions and their dependencies.

4.1.3 Sequence Diagram

A sequence diagram is used to represent all the operation going on and the order of their processes. It is a construct of a Message Sequence Chart. A sequence diagram (Grady Booch, 1999) specifies the object interaction based on the time sequence. It shows the objects and classes used to solve a problem and specify the sequence of events need to be done to accomplish the functionality. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams. In the construction of a sequence diagram the horizontal lines represent the object's life and the vertical lines represent the transfer of messages between the objects. Diagram 7 shows the sequence diagram for RCCS.

5. Comparison of FMEA over UML

Comparison of different aspects of UML and FMEA is done to analyze which of the methods is more efficient and see Table 2 for the comparison of FMEA and UML.

6. Conclusion

RCCS is taken as application to analyze the failures that occur while the system is in running condition. UML is taken as a basis to analyze the RCCS system to develop diagrams that show the occurrences of failures in the system. When the most used method of analyzing the system FMEA is taken and compared with UML, the efficient one is the UML technique to analyze the occurrences of failures in the system.

References

[1] Herman, D. S. (2000). Software safety and reliability basics (chapter 2). *Software Safety and Reliability: Techniques, Approaches, and Standards of Key Industrial Sectors.* Wiley-IEEE Computer Society Press.

[2] Petit, B. (2006). *Developing Safety-Critical Systems*. Railway Age C&S Buyers Guide.

[3] Medikonda, B. S., Ramaiah, P. S. & Gokhale, A. A. (2011). FMEA and fault tree based software safety analysis of a railroad crossing critical system. *Global Journal of Computer Science and Technology*, May, 11(8), 57-62.

[4] Booch, G., Rumbaugh, J. & Jacobson, I. (1999). *The Unified Modeling Language User Guide*. India: Pearson Education.

[5] http://www.fault-tree.net/papers/ericson-fta-history.pdf

[6] http://en.wikipedia.org/wiki/Fault_tree_analysis

[7] Knight, J. C. (2002). *Safety critical systems: Challenges and directions.* Proceedings of the 24th International Conference on Software Engineering (ICSE), Orlando, Florida.

[8] Kan, S. H. (2003). *Metrics and Models in Software Quality Engineering* (2nded.). India: Pearson Education.

[9] Wong, W. E., Debroy, V., Surampudi, A. & Kim, H. J. & Siok, M. F. (2010). *Recent Catastrophic Accidents: Investigating How Software Was Responsible*. 4th IEEE International Conference on Secure Software Integration and Reliability Improvement.

[10] Lutz, R. R. (2000). *Software Engineering for Safety: A Roadmap*. Proceedings of the Conference on The Future of Software Engineering, (pp. 213-226). Limerick, Ireland.

[11] Halawani, S. M. (2005). Safety issues of computer failure. *Egyptian Computer Science Journal*, 27(2).