CS_313 High Integrity Systems/ CS_M13 Critical Systems

Course Notes
Chapter 4: Hazard Analysis

Anton Setzer
Dept. of Computer Science, Swansea University

http://www.cs.swan.ac.uk/~csetzer/lectures/critsys/14/index.html

November 17, 2014

4. Hazard Analysis

- We have seen **limitations of formal verification** of computer systems.
  - Formal methods don't take into consideration **hardware aspects.**
    - E.g. that the wires in a railway signalling system might not conduct well under extreme weather conditions.

Limitations of Formal Methods

- Formal methods are based on a **model**, which necessarily **idealises** the system and does not take into account all aspects of the system.
  - For instance, somebody verifying the system underlying the Therac 25, might have verified in detail that the process controlling the radiation machinery is fully correct.
  - But typically the fact that there is an IO-process operating in parallel might not be taken into account in such a verification.

Limitations of Formal Methods

- Formal methods can only verify that a system fulfils its specification – whether the specification is correct cannot be checked by such methods.
  - **Example:** When modelling a railway controller, I took initially only into account that, if a signal is green, then the next segment must be reserved for trains coming from there.
  - But that one has to make sure that a train doesn't vanish magically, was overlooked initially, and in fact they vanished – the system was not safe.
  - With formal specifications one can make safety requirements precise, and hope that errors can be found – however, that the safety requirements are sufficient in order to guarantee safety has to be validated differently.

Need for Hazard Analysis

- This shows that we need other methods for identifying hazards and the risk associated with them.
- Methods used have to be more **creative** – they aim at finding hazards, which are not obvious at first sight.
- Goal is as well to determine the **safety integrity level** associated with different subsystems of the complete system, and to determine the methods for developing this system.
  - E.g. the entertainment system of an aircraft has a different safety integrity level than the autopilot. So one can for instance use different languages (e.g. C++) for developing it.

Hazard Analysis

- In the following suitable techniques for identifying and classifying hazards are presented.
- These will be **analytical** and **systematic**, but **not formal**.
- Tool support for all those techniques exist.
- The techniques were developed in **general engineering**, especially in the chemical and armaments industry.
  - The adaption of those methods to computerised systems or creation of specific methods suitable for computerised systems has not come very far yet.
    - Probably due to the low importance of computer systems in critical systems yet (but that is changing rapidly).

Techniques Considered

(a) FMEA

- **FMEA** stands for **Failure Modes and Effects Analysis**
- FMEA is a systematic method for identifying and preventing product and process problems before they occur.
- First FMEAs were conducted in the aerospace industry in the mid-1960s, specifically looking at safety issues.
- Later FMEAs became a key tool for improving safety, especially in the chemical industries.
- FEMA, originally developed for safety improvement, is increasingly used as a quality improvement tool.
  - Often, carrying out FMEAs results in a substantial reduction of failures and increased productivity.

Overview over FMEA

- In FMEA a system is first divided into components.
- Then FMEA tries to identify **all ways a particular component can fail** and the **effects of a failure** on the system.
- Then these failures are systematically analysed.

Process of FMEA

- Define **scope** and **boundaries** of the main system and of this analysis.
- Break the main system down into **subsystems**.
- Assess each subsystem, and determine, **whether the failure of the subsystem would affect the main system**.
  - If it wouldn't, ignore that subsystem.
  - Otherwise, break this subsystem into further subsystems and repeat the above, until the component level is reached.

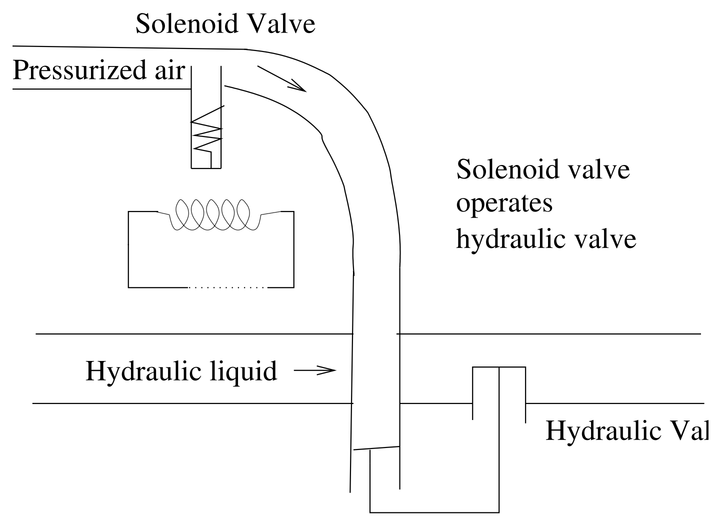
Process of FMEA (Cont.)

- For each **component** identified as above, do the following:
  - Look at the component's **failure modes** = the ways, the component can fail.

Process of FMEA (Cont.)

- Assess the failure's **effects**.
  - Usually the **worst-credible case** with consequence severity and probability of occurrence is assessed, if this is possible to calculate.
  - Determine its **mission phase** (installation, operation, maintenance, repair).
  - Identify, whether the failure is a **single-point failure**.
    (**Single point failure** = failure of a single component that could bring down the entire system.)
  - Determine methods of corrective action.
- Document the results in an FMEA worksheet.

Layout Analysed in the Next Table



Subsystem: Hydraulic Control Panel   Assembly: Junction Box A

Subassembly: Mechanical

Component number	Component name	Function	Failure mode	Mission phase
45-341	Solenoid valve	Electro-pneumatic interface and control of hydraulic panel valves	No pneumatic signal sent from valve due to loss of pressure – fail closed	Ops.
			Failed valve due to internal spring failure from excessive wear.	Ops.

Date: 10/13/96   Analyst: John Doe

Page: 13

Failure effects locally	Failure propagation next level	Single-point failure	Risk failure class	Control, recommendation
Rendered useless due to loss of working fluid	No pneumatic signal sent to hydraulic valve, resulting in longer response time to control valve 3-A	NO	4C	Manually operate hydraulic panel valve. Verify air supply inlet pressure from source. Inspect for leaks.
Continuous pneumatic flow through valve.	Possible hydraulic valve activation or deactivation due to inappropriate pneumatic pilot signal	NO	4C	Inspect and test regularly Assure correct and smooth spring-plunger alignment

Limitations of FMEA

- FMEA is primarily designed to create products which are **correct**, not to create products which are **safe**.
  - Example: If we apply FMEA to a gun, we obtain a gun, which has no failures.
    - So e.g. the barrel doesn't suddenly explode.

    However, the fact that if you direct it against a human being you can kill him, is a hazard, but no failure of the gun.
- In general hazards **need not be the result of a failure.**
- We can of course extend FMEA to treat all situations in which a gadget is used and find out failures in that constellation.
  - But that is in most cases **infeasible**.

Limitations of FMEA (Cont.)

- Direct hazard analysis will in the case of the gun **immediately identify the global hazard**.
- We see that FMEA is an **excellent engineering tool for creating perfectly functioning machinery**.
  This **contributes to** but **doesn't guarantee** safety.

Limitations of FMEA (Cont.)

- Further FMEA investigates only **single failures**.
  Often accidents have the origins in a combination of **multiple failures**, each of which on its own wouldn't have such severe consequences.

Conclusion (FMEA)

- Doesn't identify all hazards, since **a failure does not have to occur for a hazard to be present in a system.**
  - Example: A rocket is by its nature hazardous, even if it operates correctly.
- Therefore FMEA is **primarily an engineering tool**, not a **safety analysis tool**.

(b) FMECA

- **FMECA** = Failure Modes, Effects and Criticality Analysis.
  - As FMEA, but additionally determine (or estimate) for each failure:
    - the probability of its **occurrence**;
    - the probability of the **occurrence of the consequences**, provided the failure has occurred;
    - a number measuring the **criticality**.
  - The product of the 3 factors measures the **risk** associated with that failure. If the **risk exceeds a certain number**, **action** has to be taken.

Explanation of the Measure above

- The **product of the first 2 factors measures the probability** of the occurrence of this deviation followed by the consequence, i.e. **of this kind of accident**.
- Therefore the product of all 3 factors is the **product of the probability of an occurrence of the consequence and of a measure of the consequences**.
- Since **risk = product of the probability of occurrence and of the consequence**, the product of all 3 factors measures the risk. o

RPN Numbers

- Sometimes, instead of the measure above the **Risk Priority Numbers** (**RPN**) are calculated:
  - RPN = product of a measure for **severity**, **probability** and **detection**.
  - All three numbers are between 1 and 10.
  - Here detection is the likelihood that the cause of the failure is detected before reaching the customer.

(c) HAZOP

- **HAZOP** = Hazard and Operability Studies.
  - Technique developed and used mainly in **chemical industries**.
    - Studies to apply it to computer based systems have been carried out.
  - Underlying systems theory model:
    - **Accidents caused by deviations** from the design or operating intentions, e.g.:
      if there is **no flow** or **no control signal**, although there should be one.
  - HAZOP **considers** systematically **each process unit** in the design and **each possible deviation**.
    - Deviations are identified by using the **guide words** of HAZOP.

HAZOP

- HAZOP carried out by a team.

General Procedure of HAZOP

1. Define objectives and scope of the analysis.
2. Select a HAZOP team.
   - Requires a leader, who knows HAZOP well.
   - Requires a recorder, who documents the process of HAZOP.
3. Dissect design into nodes and identify lines into those nodes.
4. Analyse deviations for each line and identify hazard control methods.
5. Document results in a table.
6. Track hazard control implementation.

Nodes and Lines

- **Node** = location, where process parameters can change. Examples:
  - A chemical reactor
  - Pipe between two units.
  - Pump.
  - Sensor.
- **Line**= interface between nodes
  - E.g. pipe feeding into a reactor.
  - Electrical power supply of a pump.
  - Signals from a sensor to a computer.
  - Signals from a computer to an actuator.

Guide Words of HAZOP

We present in the following the guide words of HAZOP and possible interpretations (however the idea is that the guide words give room for creative ideas, which should not be limited by these interpretations).

Guide Word	Chemical Plants	Computer-based Systems
No	No part of intended result achieved.	No data or control signal exchanged.
More	Quantitative increase in the physical quantity	Signal magnitude or data rate too high.

Guide Words of HAZOP (Cont.)

Guide Word	Chemical Plants	Computer-based Systems
Less	Quantitative decrease in the physical quantity	Signal magnitude or data rate too low.

Guide Words of HAZOP (Cont.)

Guide Word	Chemical Plants	Computer-based Systems
As well as	Intended activity occurs, but with additional results	Redundant data sent in addition to intended value. Function has overlooked side effect.
Part of	Only part of intended activity occurs	Incomplete data transmitted.

Guide Words of HAZOP (Cont.)

Guide Word	Chemical Plants	Computer-based Systems
Reverse	Opposite of what is intended occurs, e.g. reverse flow within a pipe.	Polarity of magnitude changes reversed. Because of overflow error integer becomes negative.

Guide Words of HAZOP (Cont.)

Guide Word	Chemical Plants	Computer-based Systems
Other than	No part of intended activity occurs, and something else happens instead	Data complete but incorrect.

Guide Words for Computer based Systems

The following new guide words have been suggested for computer-based systems.
They are particularly important for concurrent systems.

Guide Word	Chemical Plants	Computer-based Systems
Early	Not used	Signal arrives too early w.r.t. clock time.
Late	Not used	Signal arrives too late w.r.t. clock time.

Guide Words for Computer based Systems

Guide Word	Chemical Plants	Computer-based Systems
Before	Not used	Signal arrives earlier than intended within a sequence
After	Not used	Signal arrives later than intended within a sequence

Steps in the HAZOP Process

For all lines.
  For all key words and associated **deviations**
    e.g. : "No flow".
   For all possible effects of that deviation.
    If that effect is **hazardous** or prevents efficient operation.
     If the **operator cannot recognise this deviation.**
       Identify, which **changes** in the plant will
         make him/her recognise that.
     Identify changes in plant or methods, **which**
       prevent deviation, make it less likely
       or mitigate its effects.

Steps in the HAZOP Process (Cont.)

For each such **change**
  If cost of change is justified
    Agree to changes.
    Agree, who is **responsible for action**.
    Follow up to see that **action has been taken.**

Example: Temperature Sensor

Line	Attribute	Guide word	Cause	Consequence	Recommend.
Sensor supply line	Supply voltage	No	Regulator or cable fault	Lack of sensor signal detected and system shuts down	
		More	Regulator fault	Damage to sensor	Consider overvoltage protection
		Less	Regulator fault	Incorrect temperature reading	Include voltage monitoring

Example: Temperature Sensor

	Sensor current	...
Sensor output	...	

(d) Event Tree Analysis (ETA)

- Origin of ETA is in the probabilistic risk assessment of nuclear power plants in the early 1970s.
- In ETA one **starts with faults**, which can cause accidents (e.g. broken pipe).
- Then one draws a **decision tree** in order to identify **sequences of faults** resulting in accidents.
- For each such sequence one **determines its outcome**.
- **Probabilities** can be assigned to each event to determine the likelihood of that scenario.
- **Product of the failures** on each path is the probability of that event sequence.

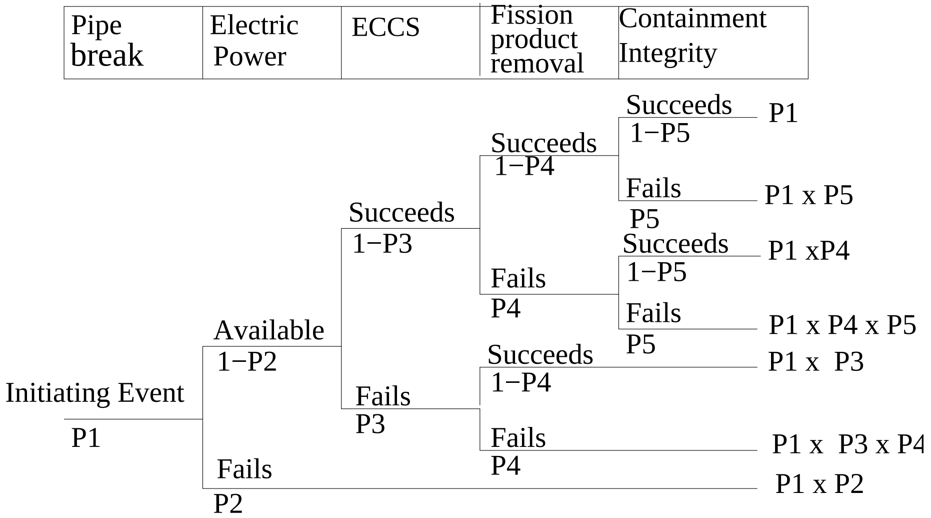
Event Tree Analysis (ETA; Cont)

- So Event tree draws a tree of possible sequences of unintended events (faults) and determines possible accidents as a result of these events.

Event Tree Analysis (ETA; Cont)

- Since **probability of failure** is usually very **low**, probabilities of success are usually **almost 1** and can be ignored in the product.

Example: Loss of cooleant accident in a nuclear power station
(ECCS = Emergency Core Cooling System)



Evaluation of Event Tree Analysis

- ETA handles **continuity of events** well.
- ETA good for **calculation of probability** of events.
  - Most widely method used for quantification of system failures.
- However, in the tree usually **many events** which **don't result in an accident occur**.
  - ⇒ ETA becomes **unnecessarily big**.
  - It is **necessary** to **cut away subtrees** which don't result in an accident.
- In general ETAs tend to become **very big**.

(e) Fault Tree Analysis (FTA)

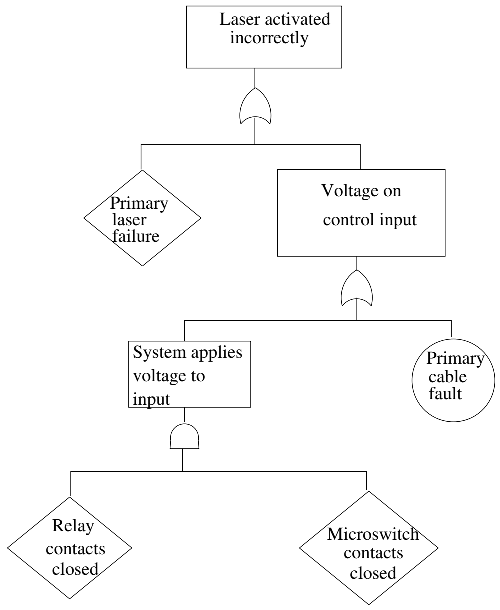
- Whereas ETA starts with faults and determines resulting accidents (events),
  FTA starts with a possible accident and determines **sequences of faults** resulting in that event.
- Usually these **conditions are disjunctive,**
  - if one of the conditions is satisfied the event occurs,

  or conjunctive,
  - if all of the conditions are satisfied the event occurs.
- The FTA is drawn **using logical gates**.

Fault Tree Analysis (FTA; Cont)

- So Fault Tree Analysis starts with possible accidents, and determines using logical gates possible combination of events leading to this accident.

Fault Tree of a Laser

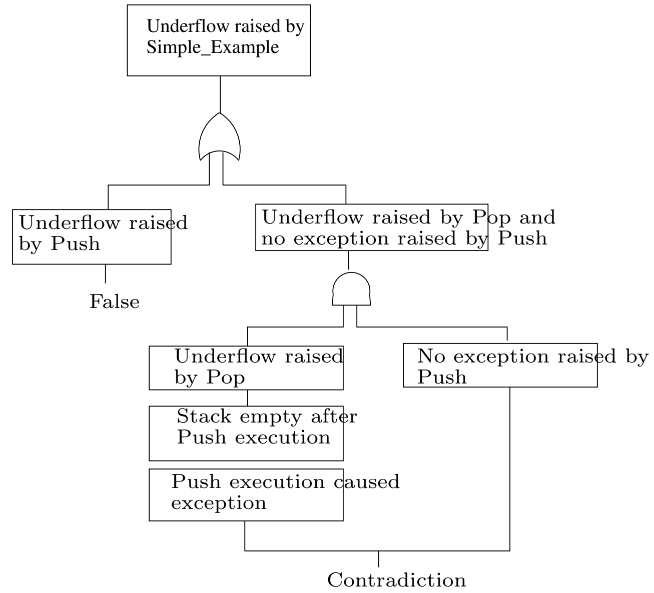


Software Example (Fault Tree)

(From [GM02], p. 169 - 171)
Assume the Ada program:

```
with Stack, Ada.Text_IO;
procedure Simple_Example (Element: in out
                                Type_Element) is
begin
  Stack.Push(Element);
  Stack.Pop(Element);
exception
  when Overflow ==> Ada.Text_Io.Put_Line
                        ("Stack Overflow");
end Simple_Example;
```

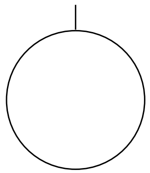

Software Example (Fault Tree)



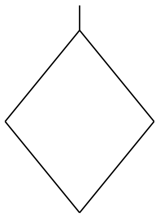
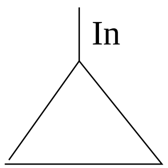
Elimination of Contradictory Paths

- One might label basic events in different branches of the fault tree which lead to contradictions.
- Then one can trace them back in order to eliminate whole branches of the fault tree.
  - In the example the complete fault tree can be eliminated.

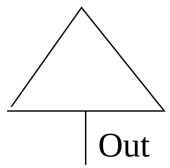
Fault Tree Symbols

Official Symbol	Meaning
	Fault event resulting from other event
	Basic event taken as input

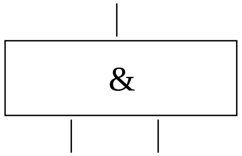
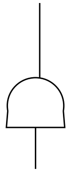
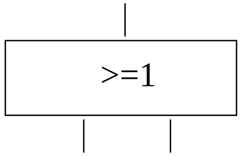
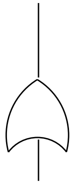
Fault Tree Symbols

Official Symbol	Meaning
	Fault event not fully traced. Taken as input but causes unknown
	Input from other fault tree

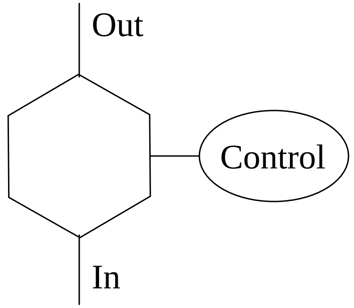
Fault Tree Symbols (Cont.)

Official Symbol	Alternative Symbol	Meaning
		Output to other fault tree

Fault Tree Symbols (Cont.)

Official Symbol	Alternative Symbol	Meaning
		Event occurs if all inputs occur
		Event occurs if at least one input occurs

Fault Tree Symbols (Cont.)

Official Symbol	Meaning
	Event occurs depending on control condition

Cut Sets

- Fault trees can be written as **Boolean formulas** (take and/or as Boolean and/or).
  - Laser Example:
    ((Relay Contacts Closed and Cond1)
    ∧ (Micro Switch Contacts Closed ∧ Cond2))
    ∨ Primary Cable Fault
    ∨ Primary Laser Failure
    (where Cond1 and Cond2 are conditions identified by continuing the fault trees below the rhombuses).
- Boolean formulas can then be rewritten in **disjunctive normal form** (i.e. as an **or of ands**).
  - Laser Example has to be unfolded if Cond1 or Cond2 contain ors.

Cut Sets (Cont.)

- Now **omit conjunctions**, which are implied by **shorter ones.**
  - E.g. In $(A \wedge B) \vee (C \wedge B) \vee B$, $(A \wedge B)$ and $(C \wedge B)$ can be omitted.
- Each conjunction determines a **minimal sequence of events** resulting in an accident.
  These conjunctions are called **minimal cut sets**.

Cut Sets (Cont.)

- Short cut sets indicate particular weaknesses of the system.
- If the faults in a cut set are independent, the probability of the events in one cut set occurring is the **product of the probabilities of the individual events**.
- If the cut sets are independent, the probability of the accident occurring is the **sum of the probabilities of the cut sets**.

Cut Sets (Cont.)

- Often however the events **in one cut set** are not independent.
  - Implies that the probability of them occurring is **much higher**.
  - Common mistake to **overlook independence**, which results in **too low risk estimates**.
- Cut sets can be generated **automatically** from fault trees.

Summary

- We have studied 5 techniques for Hazard analysis.
  - FMEA and FMECA.
    - Concentration on **avoidance of failures**.
    - Will usually only find **single point failures**.
    - Allows to produce **highly reliable systems**, but **does not** necessarily **identify all hazards**.
    - Best technique for areas where high reliability is crucial, esp. aerospace.
  - HAZOP.
    - Use of **guide words**.
    - Adaption to computer systems still in experimental state.
    - Most creative of the methods.

Summary

- ETA.
  - **Starts from faults**.
  - Event trees might **grow too big.**
  - Used in order to get good estimates for accidents of nuclear power stations.
- FTA.
  - **Starts from accidents.**
  - Seems to be **most suitable technique** in order to identify hazards – however, it seems to be useful to complement it with HAZOP (and with FMEA/FMECA, if reliability is crucial).