Faculty of ECMS / School of Computer Science

Software Engineering & Project
Safety Critical Systems

Safety Critical Systems

Lecture 13

Motivation

- Increasingly software is part of critical systems
- Failure of software can lead to catastrophic effects
  - Death or injury
  - Environmental damage
  - Security breaches
  - Financial loss
  - Mission failure
- Need to use more rigorous methods for engineering software
  - Above and beyond standard software engineering techniques

Overview

- Critical Systems
- Hazard Analysis
  - Identify potentially hazardous states of the system
  - Identify safety requirements
- Risk Analysis
  - Assess the risk of hazards
  - Assign integrity levels to safety requirements
- Designing for safety
  - Use rigorous development techniques to eliminate or remove software failures
    - Formal methods
    - Rigorous software development
    - Verification and validation

Critical Systems

- Software failures are common. If system failures result in significant economic losses, physical damage or threats to human life, such systems are called *critical systems.*

Critical Systems (cont.)

- Safety-critical systems
  - Failure results in loss of life, injury or damage to the environment;
  - E.g., Chemical plant protection system;
- Mission-critical systems
  - Failure results in failure of some goal-directed activity;
  - E.g., Spacecraft navigation system;
- Business-critical systems
  - Failure results in high economic losses;
  - E.g., Customer accounting system in a bank;

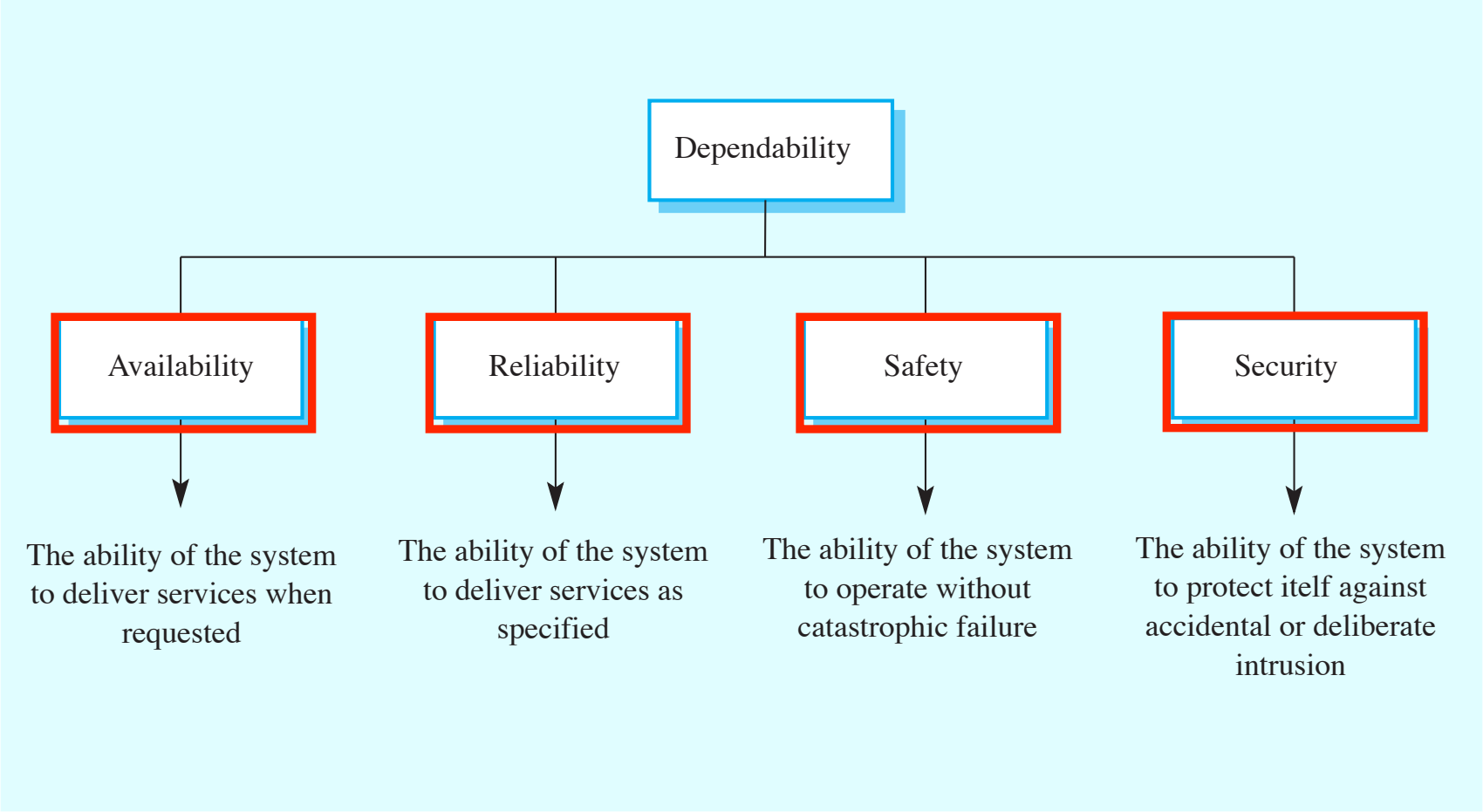
System Dependability

- The most important property of a system
- The dependability of a system reflects the ***user's degree of trust*** in that system. It reflects the extent of the user's confidence that it will operate as users expect and that it will not 'fail' in normal use.
- A dependable system is a system that is trusted by its users.

Importance of Dependability

- Systems that are not dependable and are unreliable, unsafe or insecure may be ***rejecte***d by their users. Perhaps also other products from the same company!!
- The costs of system failure may be very ***high***.
- Undependable systems may cause ***information loss*** with a high consequent recovery cost.

Dimensions of Dependability



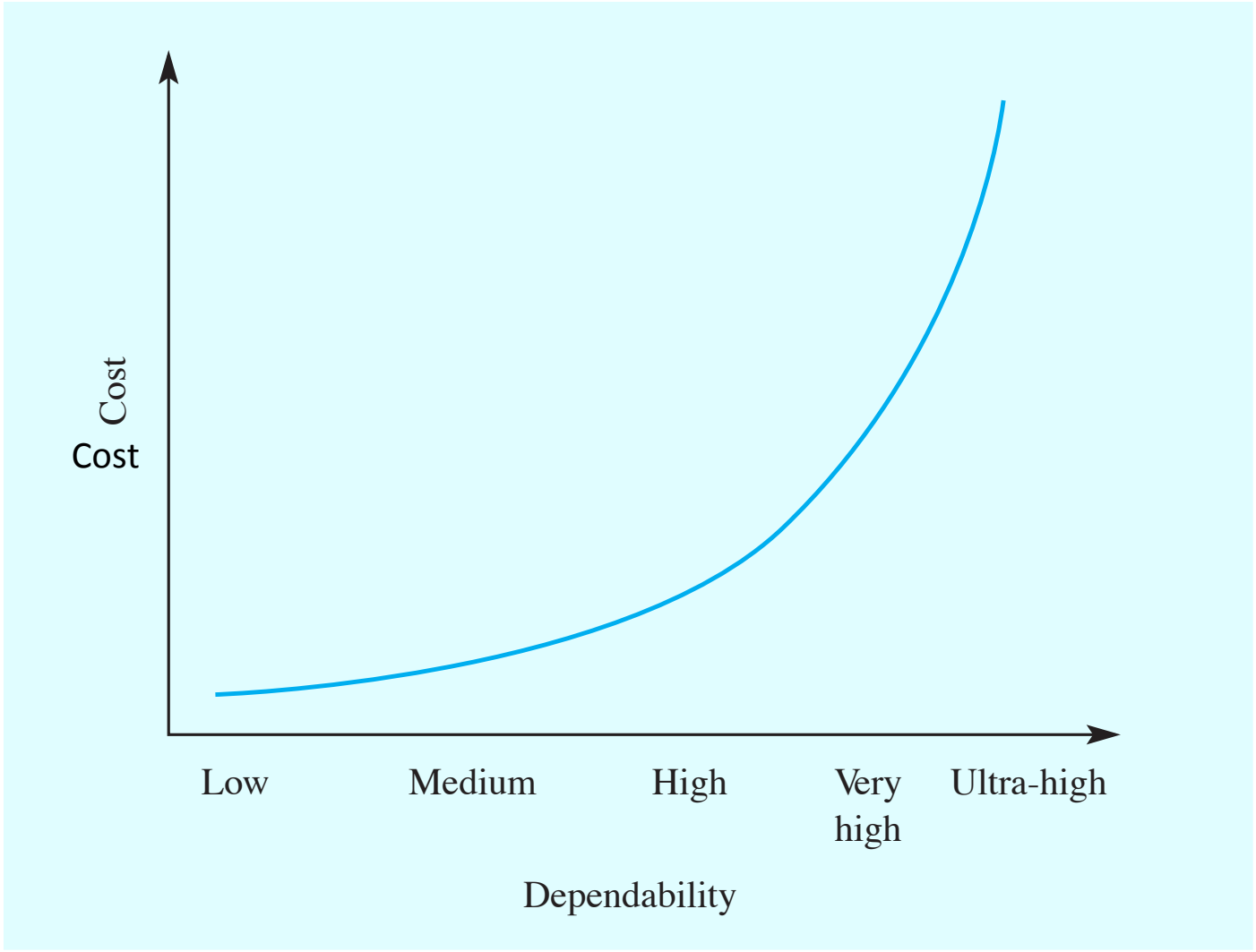
Other Dependability Properties

- Repairability: Reflects the extent to which the system can be repaired in the event of a failure
- Maintainability: Reflects the extent to which the system can be adapted to new requirements
- Survivability: Reflects the extent to which the system can deliver services whilst under hostile attack
- Error tolerance: Reflects the extent to which user input errors can be avoided and tolerated

Dependability Costs

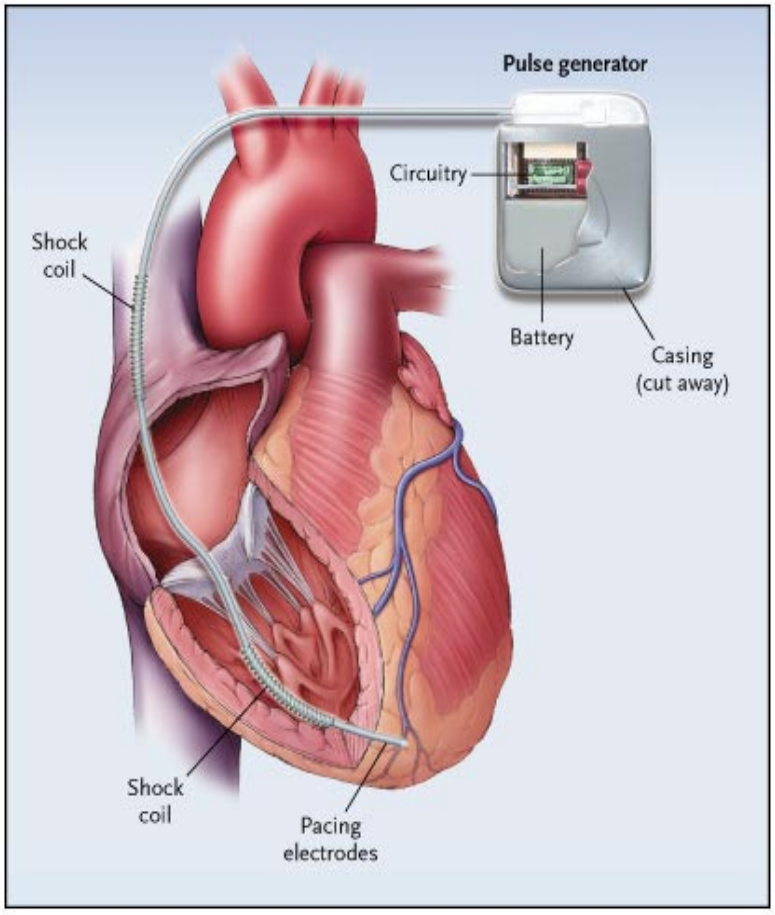
- Dependability costs tend to increase exponentially as increasing levels of dependability are required
- There are two reasons for this
  - The use of more expensive development techniques and hardware that are required to achieve the higher levels of dependability
  - The increased testing and system validation that is required to convince the system client that the required levels of dependability have been achieved

Costs vs Dependability



Drive-by-wire

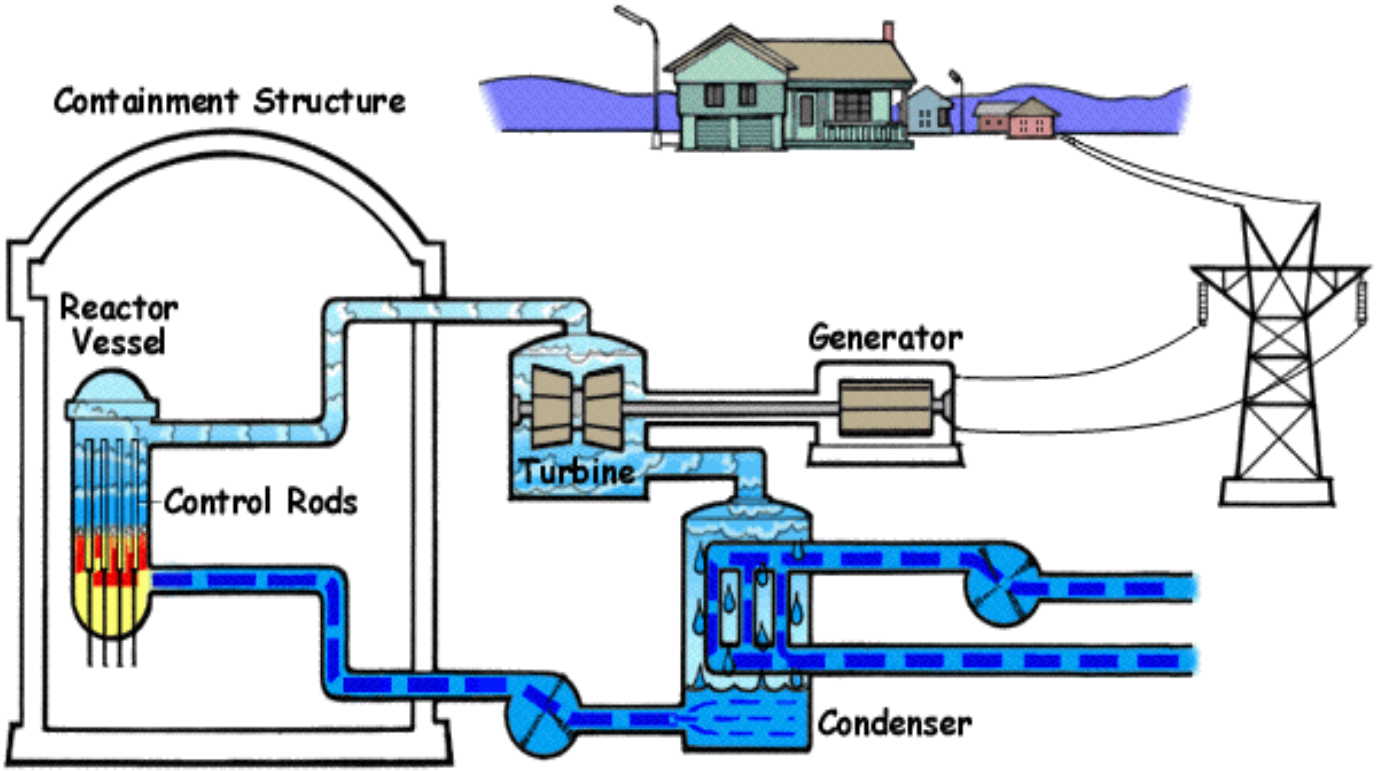
Implantable Defibrillators



Traffic Alert and Collision Avoidance System (TCAS)

On-board train control

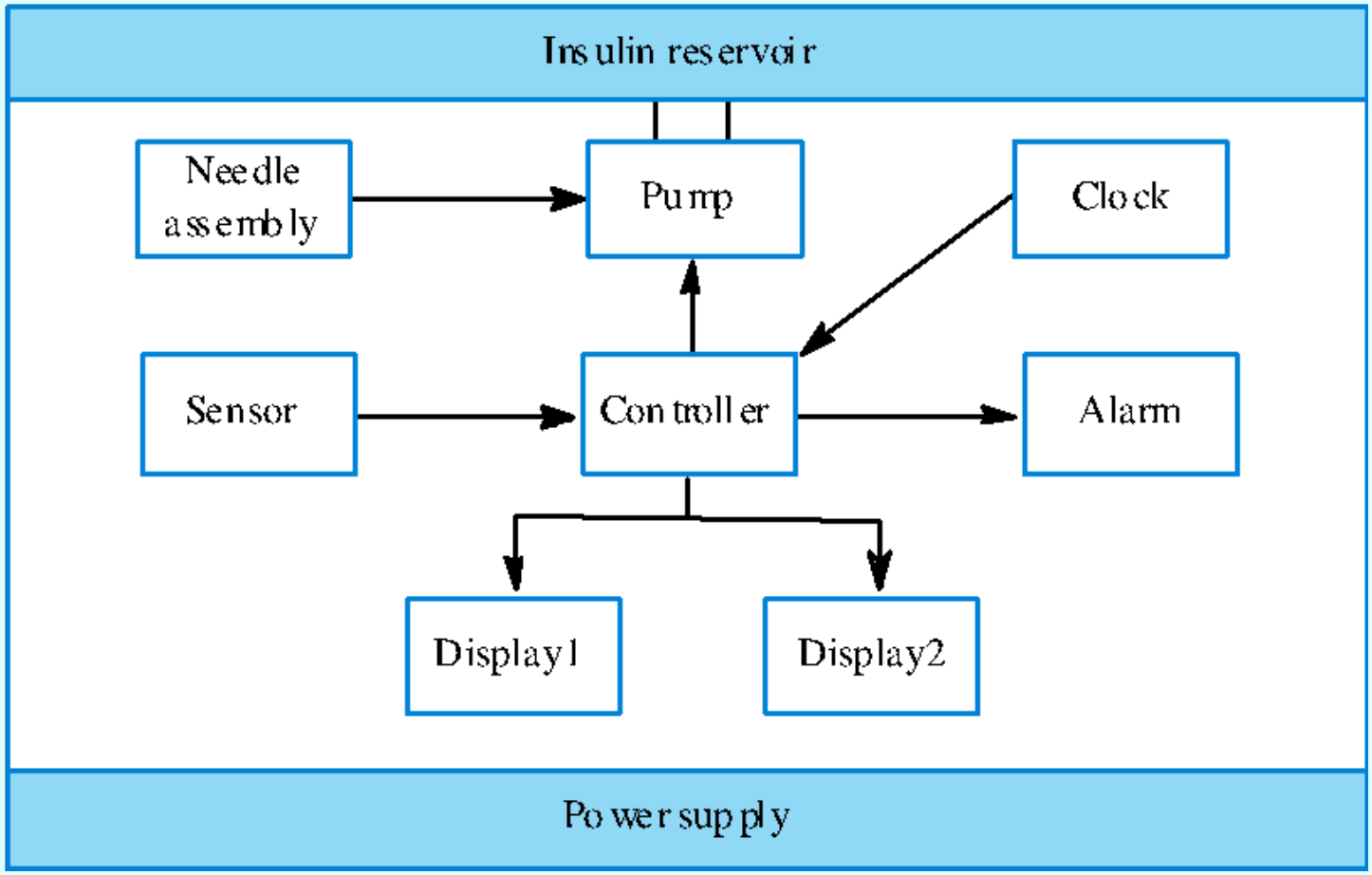
Nuclear power plants



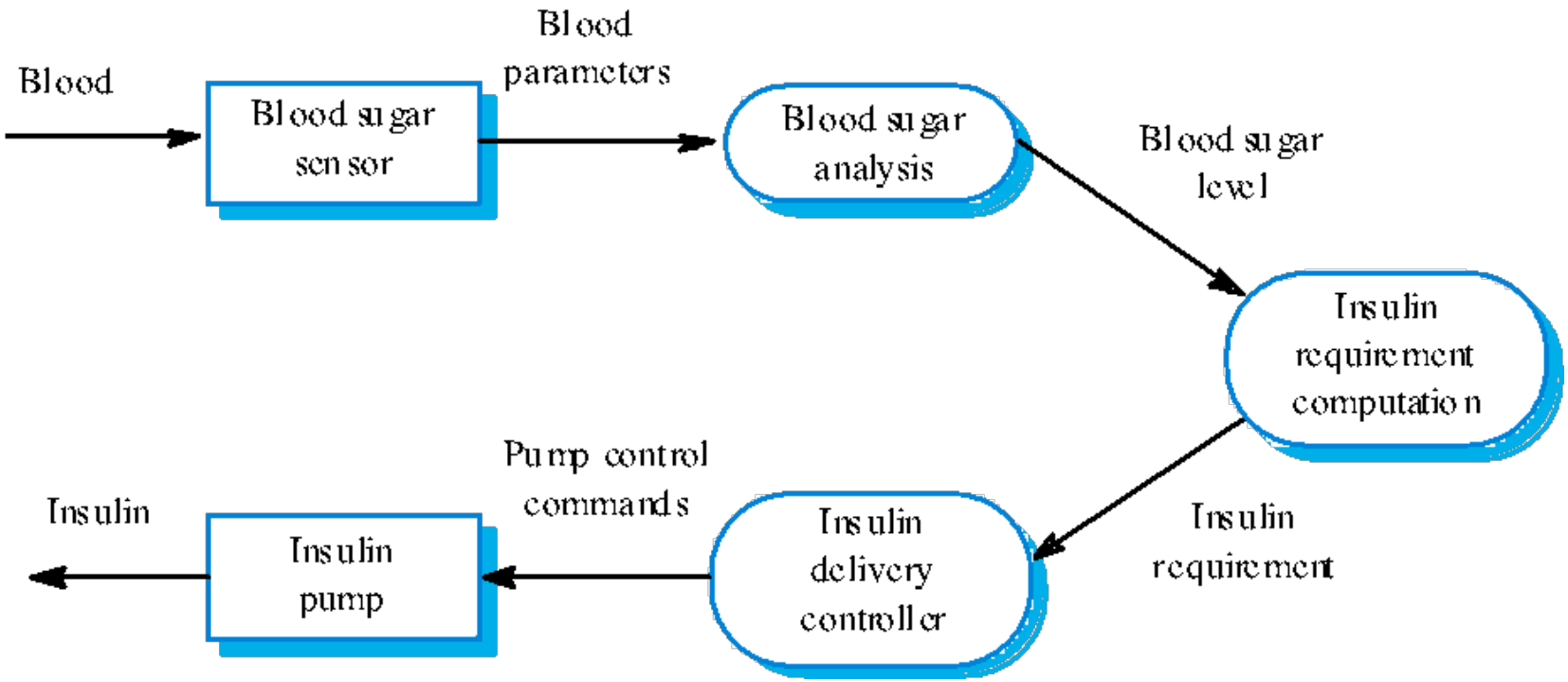
A software-controlled insulin pump

- Used by diabetics to simulate the function of the pancreas which manufactures insulin, an essential hormone that metabolises blood glucose.
- Measures blood glucose (sugar) using a micro-sensor and computes the insulin dose required to metabolise the glucose.

Insulin pump organisation



Insulin pump data-flow

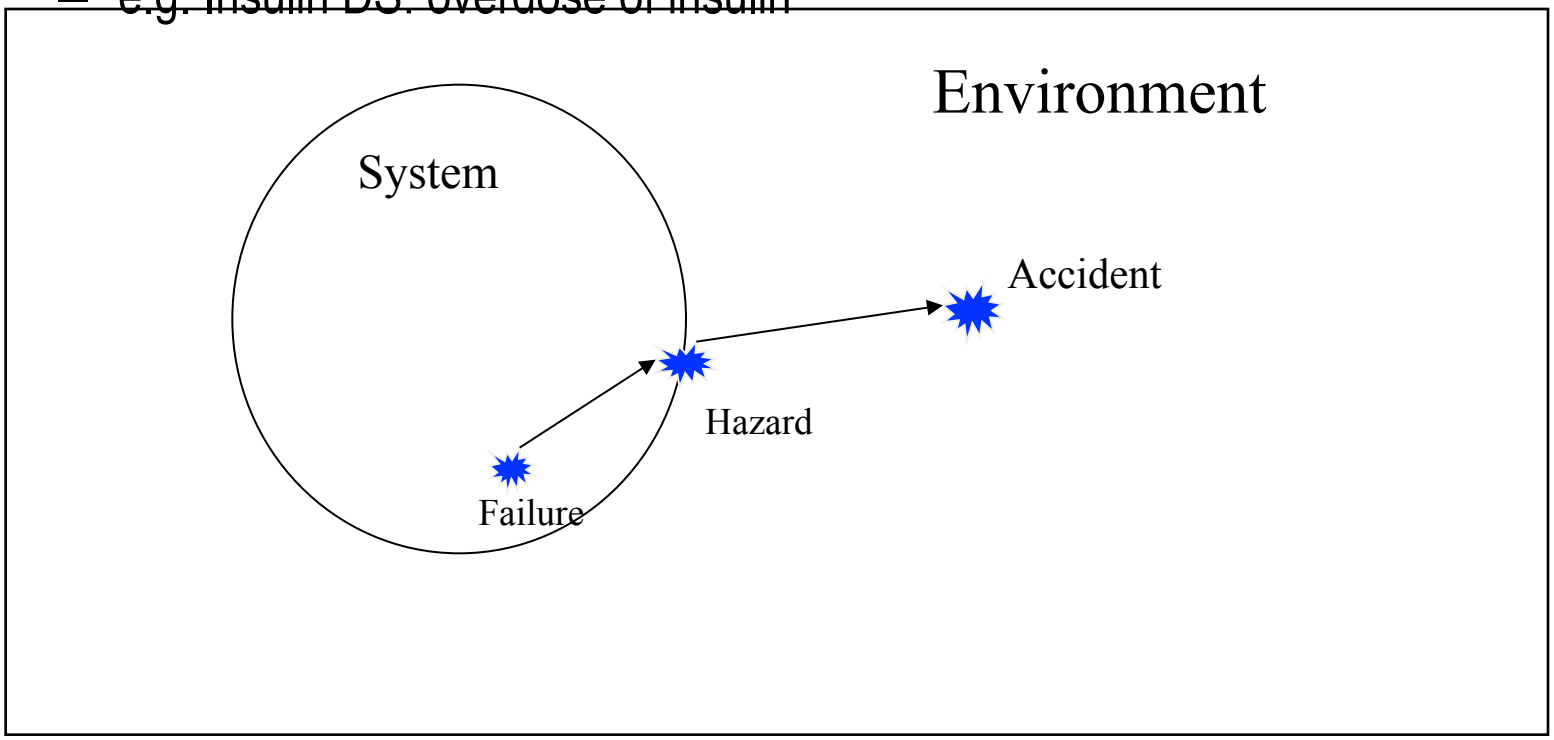


Terminology

- Accident
  - An unintended event or sequence of events that causes death, injury, environmental or material damage
  - e.g. Insulin DS: death of patient due to insulin overdose
- Incident
  - An unintended event or sequence of events that does not result in loss, but, under different circumstances, has the potential to do so.
- Hazard
  - State of a system that, together with other combinations in the environment of the system, will lead inevitably to an accident
  - E.g., Insulin PS: overdose of insulin

Terminology (cont.)

- Hazard
  - State of a system that, together with other combinations in the environment of the system, will lead inevitably to an accident
  - e.g. Insulin DS: overdose of insulin



Terminology (cont.)

- System failure
  - Occurs when the system fails to perform its required function
  - e.g. Insulin DS: failure of maximum daily dosage interlock
- Fault
  - Defect within the system
  - May lead to a failure
  - e.g. Insulin DS: overflow of timer variable
- Error
  - Deviation from the required operation of the system
  - Manifestation of a fault
  - e.g. Insulin DS: timer variable erroneously set to zero

Terminology (cont.)

- Risk
  - A combination of the severity of a hazardous event and likelihood of its occurrence
- Severity
  - A measure of the possible extent of harm
  - e.g. Insulin DS: patient loses life (critical)
- Likelihood
  - The probability or frequency of event occurrence

The nature of faults

- Random faults
  - Associated with hardware failure
  - Can often predict mean time to failure
  - Often fail gracefully
  - Maintenance can mitigate against random faults
- Systematic faults
  - Design faults
  - Mistakes made in the specification
  - Software faults
  - Will always occur give a certain set of conditions

Fault-free systems

- No faults = No errors = No System failures
- But impossible to achieve
  - All systems include physical components that are subject to random failure
    - Wear, ageing, etc
  - Impossible to achieve perfect design
    - Systematic faults persist

Fault Management

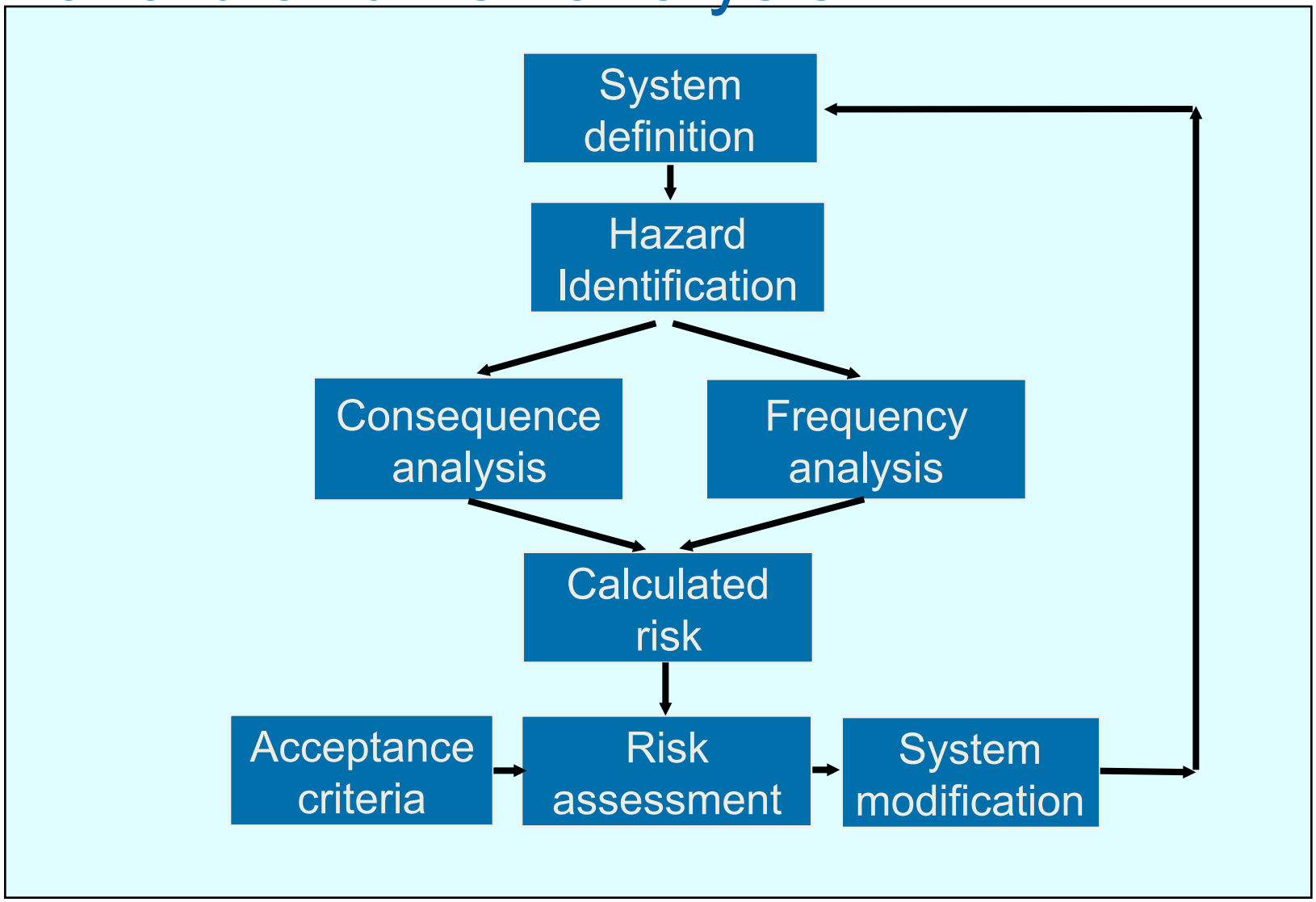
- Fault avoidance
  - Prevent faults entering the system during design
  - e.g. use of formal methods
- Fault removal
  - Find faults before systems enters service
  - e.g. software testing
- Fault detection
  - Detect faults during operation and minimise effect
- Fault tolerance
  - Allow system to operate in the presence of faults

Risk-Driven Analysis

Ch. 9.1

- Critical systems specification should be **risk-driven**.
- This approach has been widely used in safety and security-critical systems.
- The aim of the specification process should be to understand the risks (safety, security, etc.) faced by the system and to define requirements that reduce these risks.

Hazard and risk analysis



Hazard Analysis

- Variety of techniques for analysing systems
  - Different insight into characteristics of system
  - Evolved from other disciplines
- Widely used techniques include
  - Failure modes and effects analysis (FMEA)
  - Hazard and Operability Studies (HAZOP)
  - Fault Tree Analysis (FTA)
  - Event Tree Analysis (ETA)

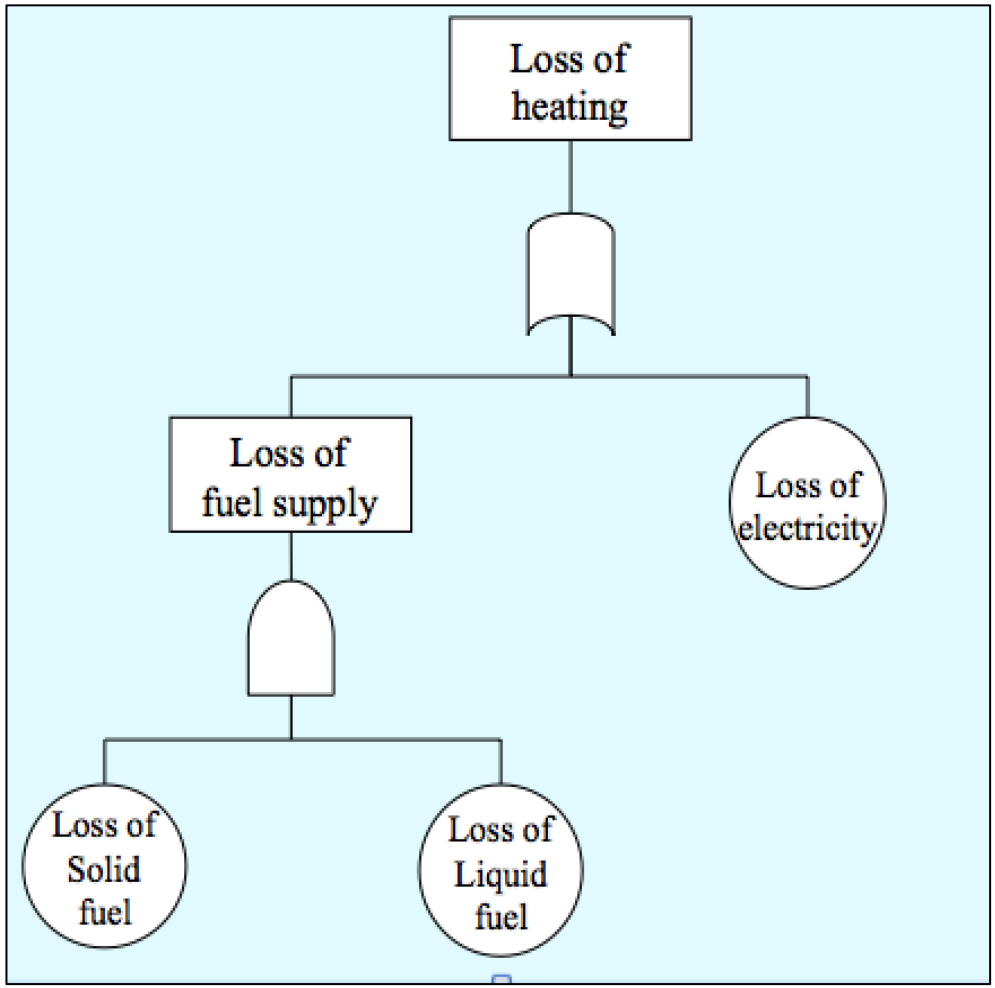
Classification of Hazard Analysis Techniques

	Unknown Consequences	Known Consequences
Unknown Cause	Exploratory Analysis HAZOP	Causal Analysis FTA
Known Cause	Consequence Analysis FMEA, ETA	Verification FMEA, ETA

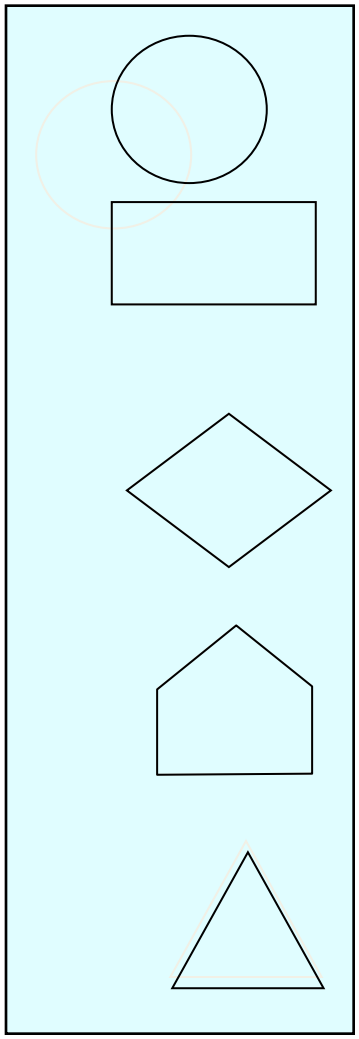
Fault Tree Analysis

- The most commonly used Causal Analysis technique
  - developed by Bell Labs & US Air Force in early '60's
  - now widely used in many industry sectors (nuclear, defence, …)
  - standardised: see International Standard IEC 61025
- **Goal:** create a 'cause-&-effect model' of the system
  - a tree with a *top event* at the root
  - *logic gates* at branches, linking each event with its immediate causes
  - *initiating faults* at leaves (eventually)

Example: Heating system



Event Symbols



Basic Event – requires no further development

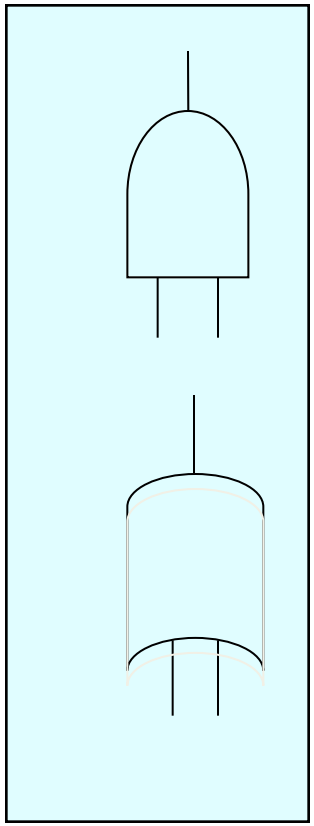
Intermediate Event – results from a combination of events through a logic gate

Undeveloped Event – not further developed. Event not consequential or information not available

Normal Event – expected to occur normally

Transfer

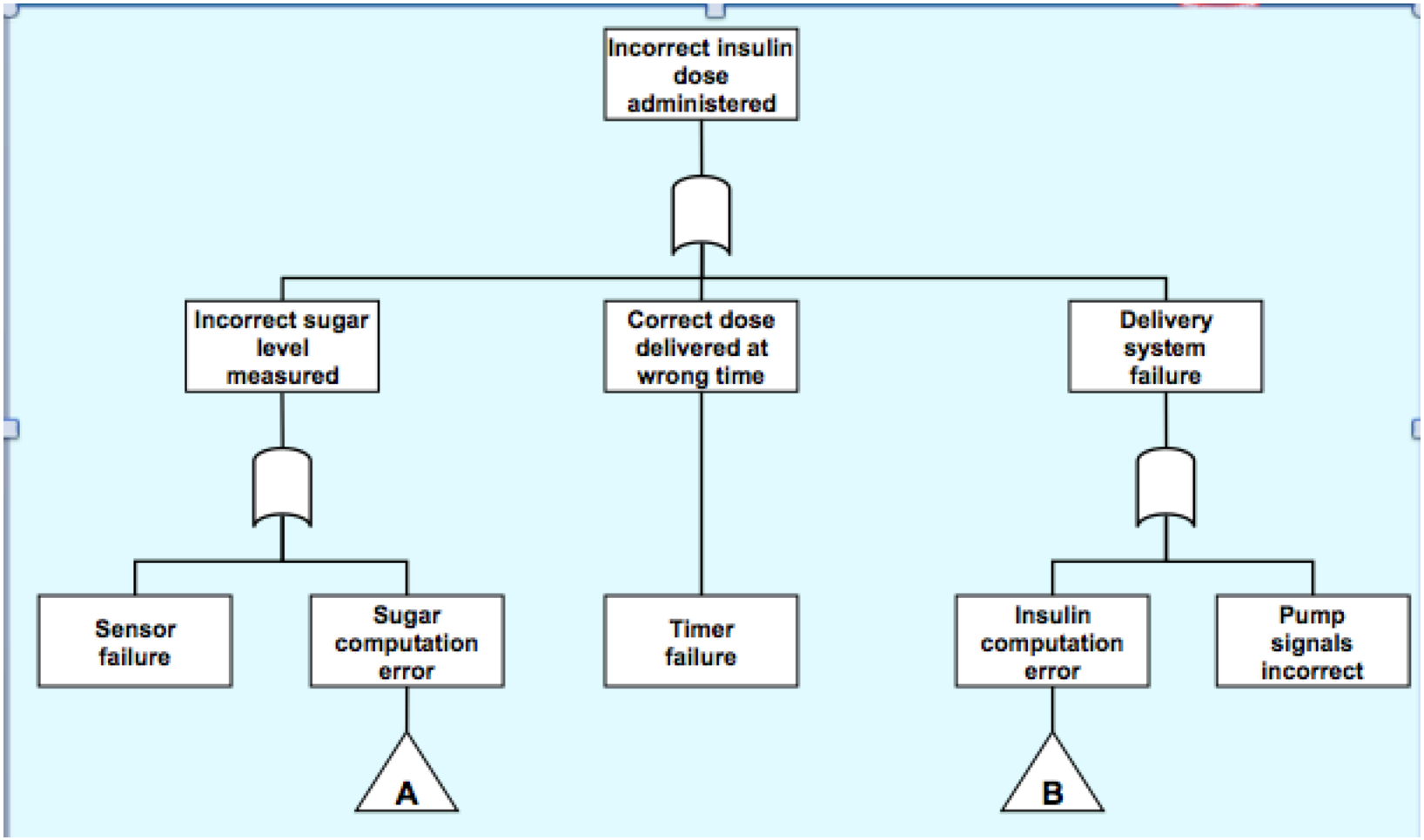
Gate Symbols



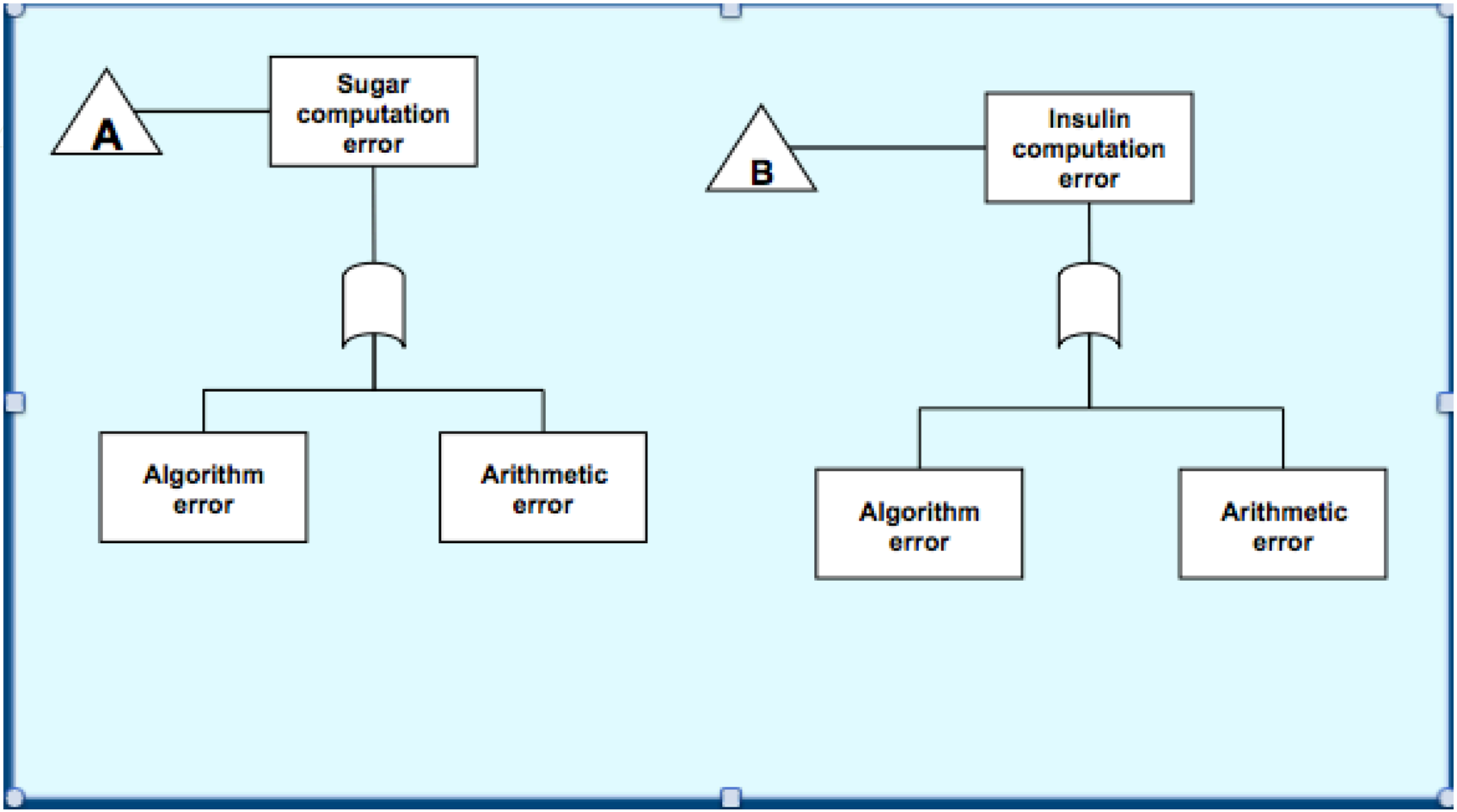
AND Gate – the output event occurs if ALL the inputs occur

OR Gate – the output event occurs if ANY of the inputs occur

Insulin Pump FTA



Insulin Pump FTA (cont)



Safety requirements

- Once hazards are identified and assessed, safety requirements are generated to mitigate the risk
- e.g. for Insulin DS
  - SR1: The system shall not deliver a single dose of insulin that is greater than a specified maximum dose for a system user
  - SR2: The system shall not deliver a daily cumulative dose of insulin that is greater than a specified maximum for a system user
  - SR6: In the event of an alarm in the system, insulin delivery shall be suspended until the user has reset the system and cleared the alarm
- Safety requirements form basis for subsequent development

Risk Analysis

- Need to access the risks associated with identified hazards
  - determine the relative importance
  - judge their acceptability
- Consider the possibility of a hazard occurring and its consequence
- Then apportion effort for developing components of a system based on relative risk

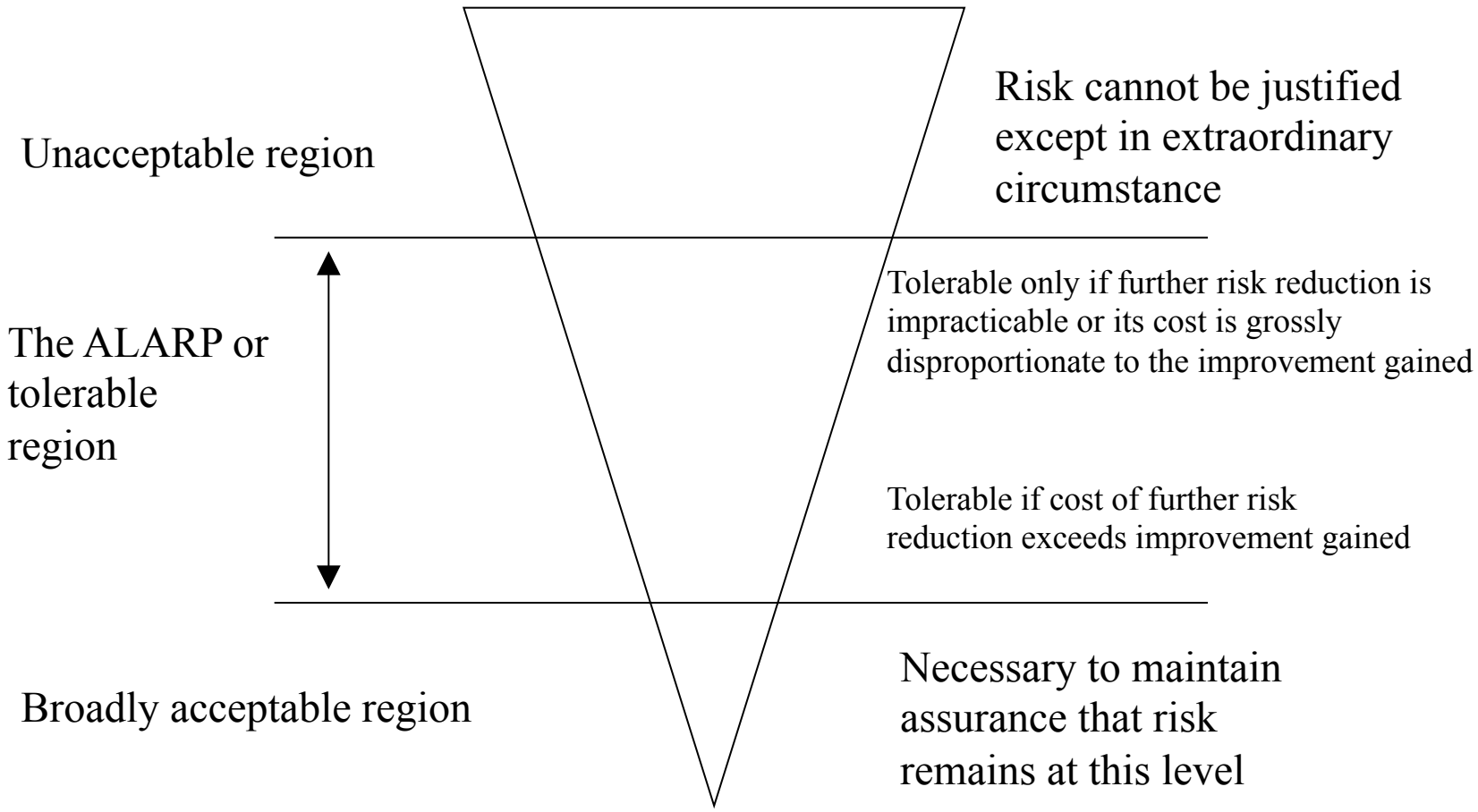
Risk Analysis Gone Bad

- Ford Pinto (1960s compact)
- Design flaw
  - Tests revealed that the gas tank would rupture in crashes over 25mph
  - Correction required changing and strengthening the design
- Risk Analysis
  - Ford estimated 180 deaths, 180 serious burns and 2100 burned vehicles per year
  - At a cost of $200000 per death, $67000 per injury and $700 per vehicle fix
  - Total cost ~ $50 million per year
  - Cost to fix vehicles estimated to be $11 per car
  - Total cost ~$137 million per year

Risk Analysis Gone Bad

- The outcome
  - Ford decided it would be cheaper to pay costs of accidents rather than fix the design flaw
  - But Ford was sued once cars started exploding in rear-end collisions
  - Actual cost of alterations found to $1 per car
  - Motor vehicle industry now has much tighter regulation

ALARP (As Low As Reasonably practical)



Risk Assessment - Insulin Pump

Identified hazard	Hazard probability	Hazard severity	Estimated risk	Acceptability
1. Insulin overdose	Medium	High	High	Intolerable
2. Insulin underdose	Medium	Low	Low	Acceptable
3. Power failure	High	Low	Low	Acceptable
4. Machine incorrectly fitted	High	High	High	Intolerable
5. Machine breaks in patient	Low	High	Medium	ALARP
6. Machine causes infection	Medium	Medium	Medium	ALARP
7. Electrical interference	Low	High	Medium	ALARP
8. Allergic reaction	Low	Low	Low	Acceptable

Severity

- Hazards classified in terms of their severity
  - Severity usually classified qualitatively
  - Classification depends on domain
  - May differ for military, industrial and civilian domains
- Typically distinguish between multiple deaths, single death, severe injuries and minor injuries

Example (IEC 61508)

Category	Definition
Catastrophic	Multiple deaths
Critical	A single death, and/or multiple severe injuries or severe occupational illnesses
Marginal	A single severe injury or occupational illness, and/or multiple minor injuries or occupational illnesses
Negligible	At most a single minor injury or minor occupational illness

Severity Analysis

- Identify all accidents associated with the system
  - Assign a severity to each accident using an agreed severity classification table
  - Provide justification for each assignment
- For Insulin Delivery System accidents could include
  - Critical: Fatal insulin overdose
  - Marginal: Insulin overdose induced coma
  - Marginal/Critical: Heart/kidney/eye problems (caused by ongoing insulin underdose)

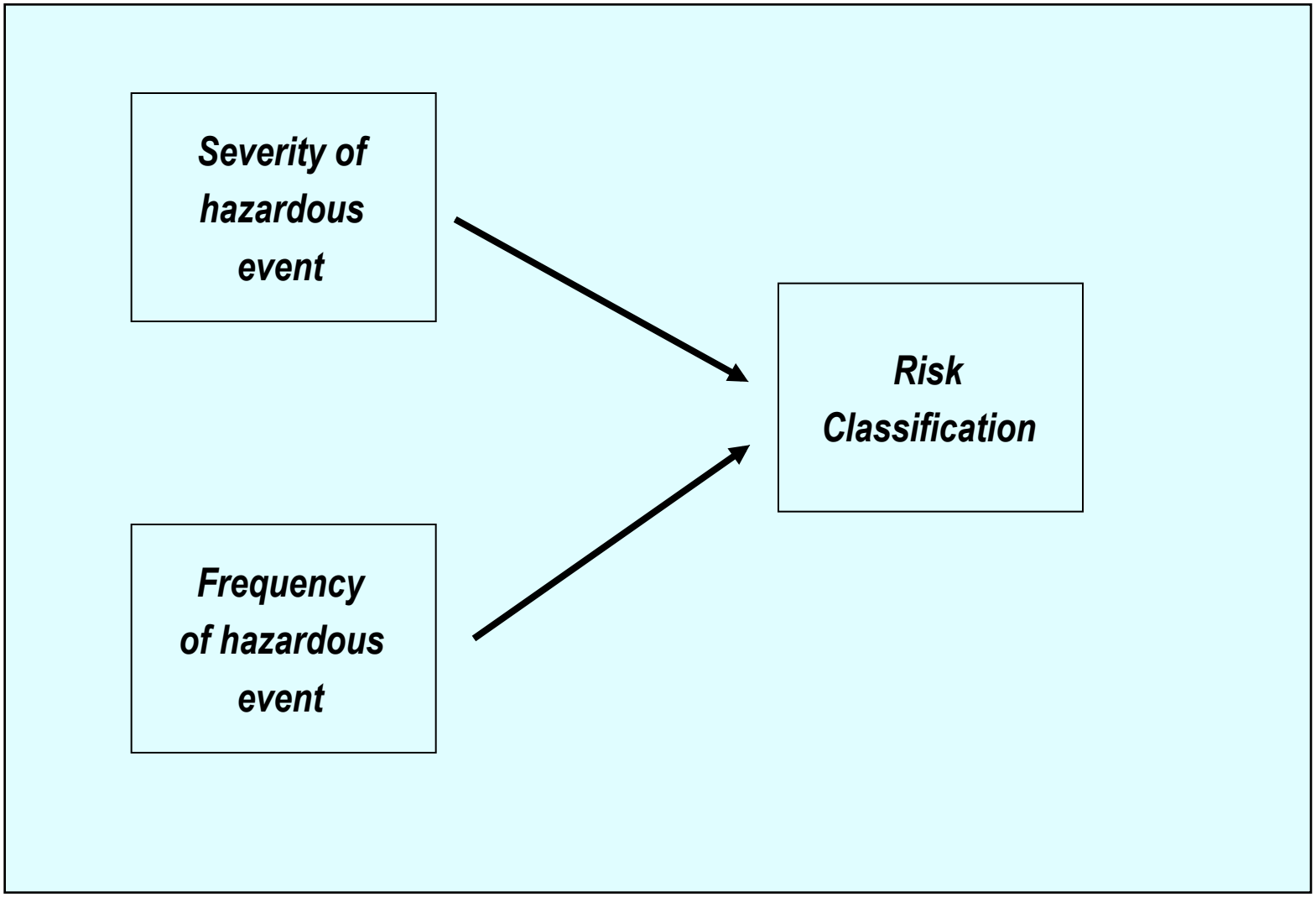
Frequency

- A measure of how often a hazard is likely to occur
  - Represented in various forms, quantitatively or qualitatively
- Qualitative measures
  - e.g. Frequent, probable, occasional, remote, improbable
- Quantitative measures
  - Occurrences per year
  - Occurrences per hour
  - Failure on demand (number of failures as a fraction of total number of uses)

Frequency Table (00-56)

Accident Frequency	Occurrence during operational life for all instances of system	Numerical equivalent probability
Frequent	Likely to be continually experienced	10000x10-6/operating hour
Probable	Likely to occur often	100x10-6/operating hour
Occasional	Likely to occur several times	1x10-6/operating hour
Remote	Likely to occur some time	0.01x10-6/operating hour
Improbable	Unlikely, but may exceptionally occur	0.0001x10-6/operating hour
Incredible	Extremely unlikely that the event will occur at all	0.000001x10-6/operating hour

Risk classification



Example: IEC 61508

Consequences				
Frequency	Catastrophic	Critical	Marginal	Negligible
Frequent	I	I	I	II
Probable	I	I	II	III
Occasional	I	II	III	III
Remote	II	III	III	IV
Improbable	III	III	IV	IV
Incredible	IV	IV	IV	IV

Risk classes

I	Intolerable risk
II	Undesirable risk; tolerable only if risk reduction is impracticable or costs are grossly disproportionate to improvement gained
III	Tolerable risk if the cost of risk reduction would exceed improvement gained
IV	Negligible risk

Safety integrity

- Likelihood of safety-related system satisfying safety requirements under all stated conditions within a stated period of time
- Safety requirements are allocated one of a number of safety integrity levels
  - Indicator of the required level of protection against failures
  - May be either quantitative or qualitative
  - Quantitative
    - e.g. measures failures per year
  - Qualitative
    - Gives a measurement of the level of rigour expected during development

Safety integrity (cont.)

- Hardware integrity
  - That part of the safety integrity relating to dangerous random hardware failures
  - Associate a target failure rate with hardware components
- Systematic integrity
  - That part of the safety integrity relating to dangerous systematic failures
  - Includes software
  - Cannot associate target failure rates
  - Instead SIL determines
    - Development methods used
    - Level of testing performed

Safety Integrity Levels

Safety Integrity Level	Tolerable failure rate
S4	Remote
S3	Occasional
S2	Probable
S1	Frequent

	Consequences			
Frequency	Catastrophic	Critical	Marginal	Negligible
Frequent	I	I	I	II
Probable	I	I	II	III
Occasional	I	II	III	III
Remote	II	III	III	IV
Improbable	III	III	IV	IV
Incredible	IV	IV	IV	IV

Using SILs

- SILs dictate the level of rigour required throughout various stages of development
- e.g. from Defense Standard UK MOD 00-56

Attributes	S4	S3	S2	S1
Requirements and Design Specification	Formal	Semiformal	Informal	Informal
Configuration Management	Full	Full	Yes	Manual
Coding standards	Safe Subset HLL	Safe Subset HLL	HLL	HLL preferred
Fault tolerant techniques	Yes	Preferred	Optional	Optional
Static Analysis	Yes	Yes	Optional	Optional

Key Points

- For safety critical software we need to
  - identify hazards and safety requirements
  - decompose hazards and safety requirements down to the component level
  - assess the risk associated with each hazard
  - determine acceptable levels of risk
  - apportion development effort based on risk
  - use appropriate development techniques as determined by safety integrity levels