

Outline

- Why study fault-tolerant/safety-critical systems?
- What is a “fault-tolerant” or “safety-critical system”?
- Fundamental concepts of dependability
 - The “dependability” concept
 - Threats: fault, error, failure
 - Attributes: reliability, availability

Does the use of computer control increase safety? Some “myths”...

- The cost of computers is lower than that of analog or electromechanical devices: *not if it has to be certified and maintained*
- Software is easy to change: *very difficult without introducing errors*
- Computers provide greater reliability than the devices they replace: *not true for software, there's no “bug-free” software; reliability $\neq$ safety*
- Increasing software/hardware reliability will increase safety: *even if the design conforms to specifications, the specs can be wrong*
- Testing software/hardware or “proving” correctness can remove all the errors: *verification is very difficult, especially if it has to consider timing*
- Reusing software increases safety: *the opposite is true (Therac and Ariane)*
- Computers reduce risk over mechanical systems: *debatable*

Safety-critical systems

- Definitions
 - **Safety** is a property of a system that will not endanger human life or the environment.
 - The **integrity** of a system is its ability to detect faults in its own operation and to inform the human operator.
- **Safety-critical system** is:
 - Safety-related system, or
 - High-integrity system
 - Failure could result large financial loss
 - Examples: banking system, stock market system, server clusters for cloud computing, communication satellites

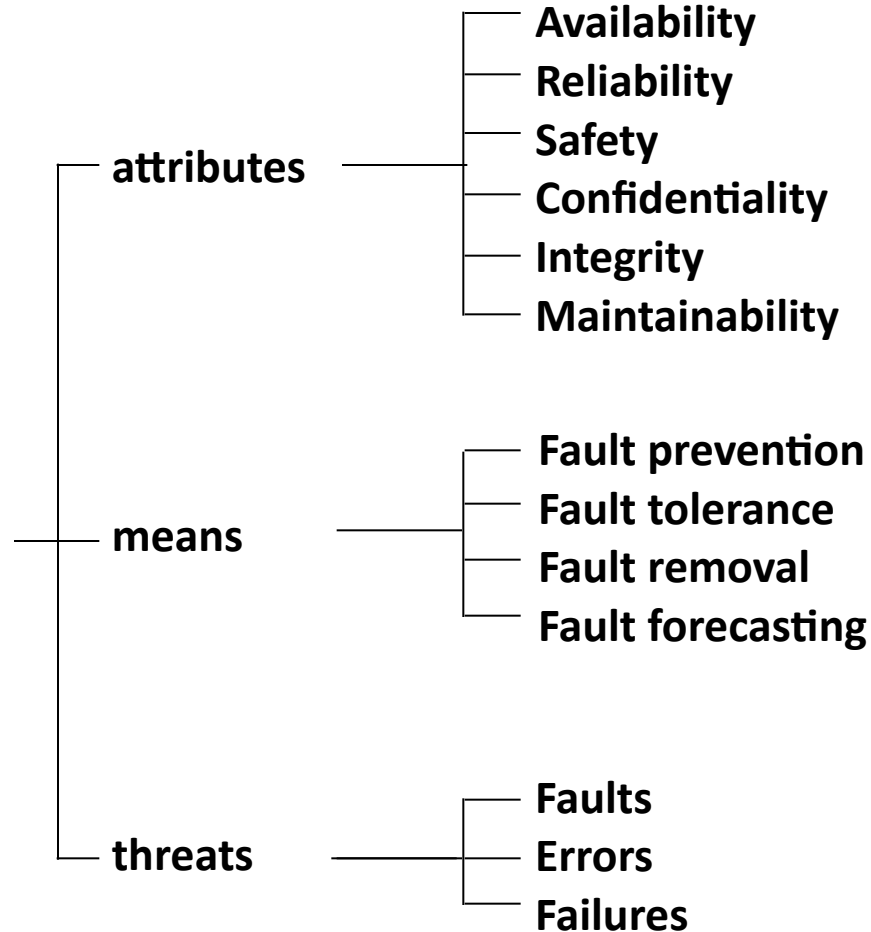
Failsafe operation

- Definition
 - A system is **failsafe** if it adopts “safe” output states in the event of failure and inability to recover.
- Notes
 - Example of failsafe operation
 - Railway signaling system: failsafe corresponds to all the lights on red
 - Many systems are not failsafe
 - Fly-by-wire system in an aircraft: the only safe state is on the ground
 - How to ensure continued operation even in case of failures?

Dependability: an integrating concept

- **Dependability** is a property of a system that justifies placing one's reliance on it.

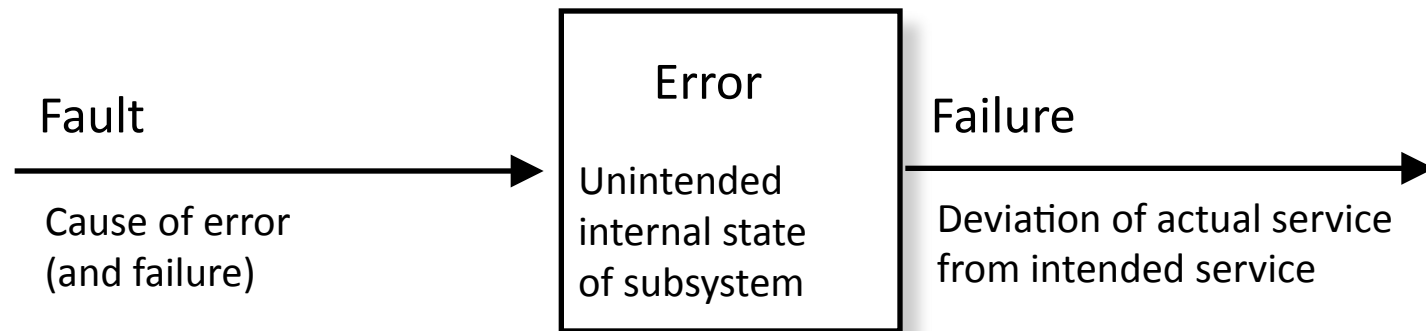
dependability



Dependability attributes

- **Reliability:** continuity of correct service
- **Availability:** readiness for correct service
- **Safety:** absence of catastrophic consequences on the user(s) and the environment
- **Confidentiality:** absence of unauthorized disclosure of information
- **Integrity:** absence of improper system alterations
- **Maintainability:** ability to undergo, modifications, and repairs
- **Security:** the concurrent existence of (a) availability for authorized users only, (b) confidentiality, and (c) integrity with 'improper' taken as meaning 'unauthorized'.

Threats: Faults, Errors & Failures



Threats: Faults, Errors & Failures, cont.

- **Fault**

- Physical defect, imperfection, or flaw that occurs within some hardware or software component.
- Examples
 - Shorts between electrical conductors
 - Physical flaws or imperfections in semiconductor devices
 - Program loop that when entered can never be exited
- Primary cause of an error (and, perhaps, a failure)
 - Does not necessarily lead to an error
e.g., a bit in memory flipped by radiation
 - can cause an error if next operation on memory cell is “read”
 - causes no error if next operation on memory cell is “write”

Threats: Faults, Errors & Failures, cont.

- **Error**

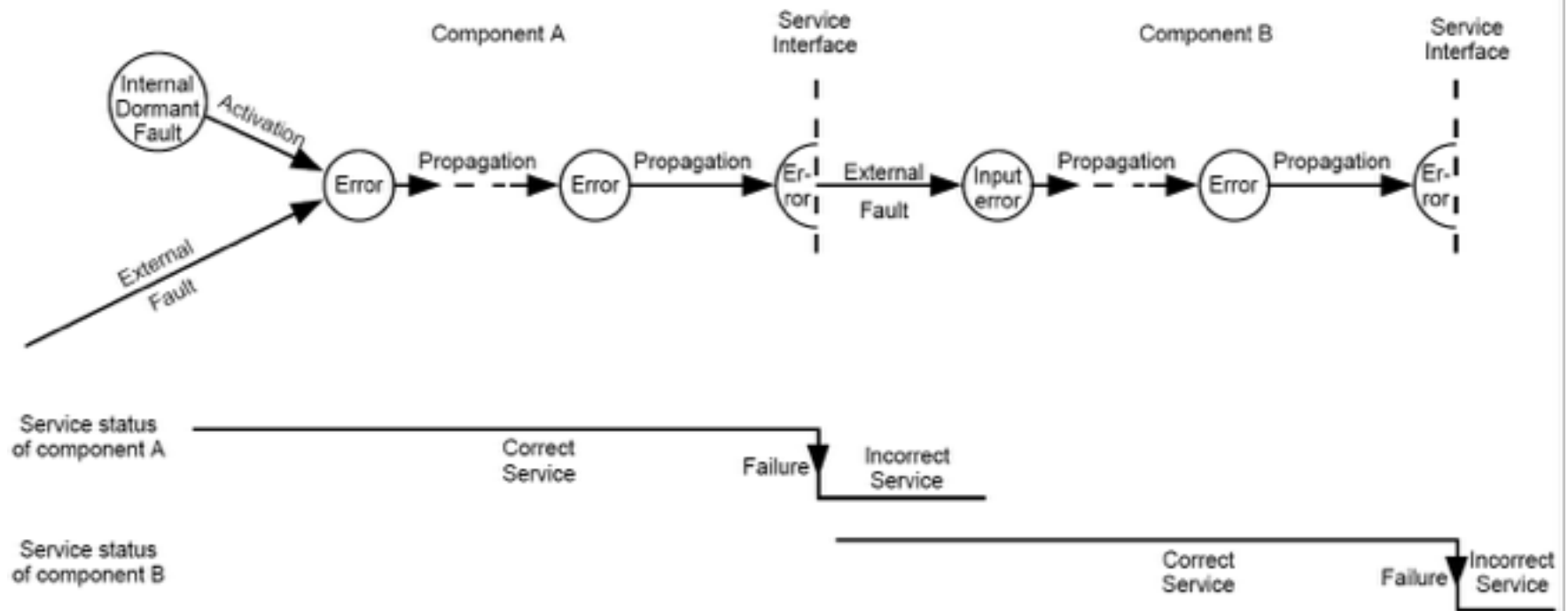
- An incorrect internal state of a computer
 - Deviation from accuracy or correctness
- Example
 - Physical short results in a line in the circuit permanently being stuck at a logic 1. The physical short is a fault in the circuit. If the line is required to transition to a logic 0, the value on the line will be in error.
- The manifestation of a fault
- May lead to a failure, but does not have to

Threats: Faults, Errors & Failures, cont.

- **Failure**

- Denotes a deviation between the actual service and the specified or intended service
- Example
 - A line in a circuit is responsible for turning a valve on or off: a logic 1 turns the valve on and a logic 0 turns the valve off. If the line is stuck at logic 1, the valve is stuck on. As long as the user of the system wants the valve on, the system will be functioning correctly. However, when the user wants the valve off, the system will experience a failure.
- The failure is an event (i.e. occurs at some time instant, if ever) caused by an error

The pathology of failure



Three-universe model

1. **Physical universe:** where the **faults** occur

- Physical entities: semiconductor devices, mechanical elements, displays, printers, power supplies
- A fault is a physical defect or alteration of some component in the physical universe

2. **Informational universe:** where the **error** occurs

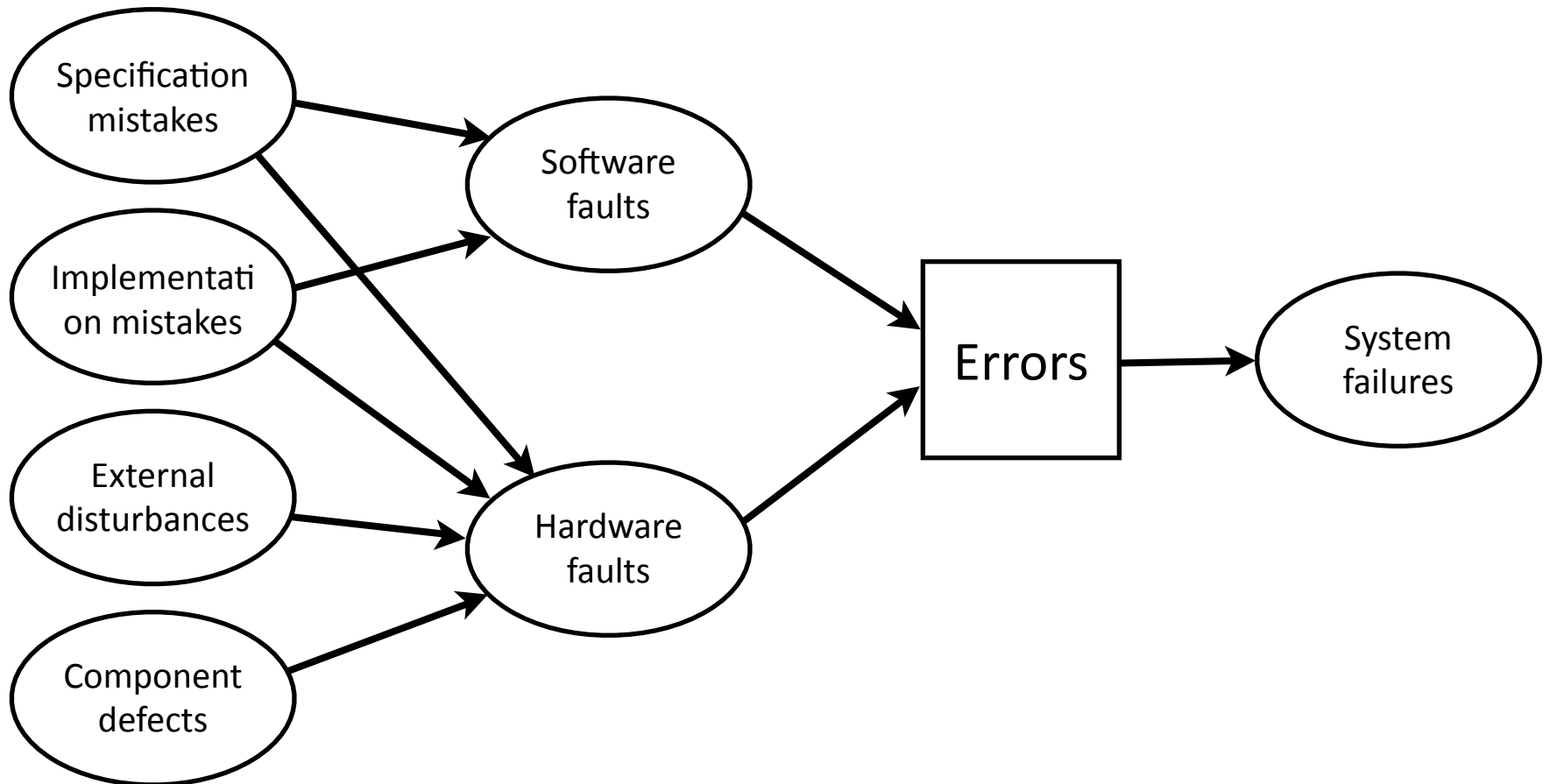
- Units of information: bits, data words
- An error has occurred when some unit of information becomes incorrect

3. **External (user's universe):** where **failures** occur

- User sees the effects of faults and errors
- The failure is any deviation from the desired or expected behavior

Causes of faults

- Problems at any stages of the design process can result in faults within the system.



Causes of faults, cont.

- Specification mistakes
 - Incorrect algorithms, architectures, hardware or software design specifications
 - Example: the designer of a digital circuit incorrectly specified the timing characteristics of some of the circuit's components
- Implementation mistakes
 - Implementation: process of turning the hardware and software designs into physical hardware and actual code
 - Poor design, poor component selection, poor construction, software coding mistakes
 - Examples: software coding error, a printed circuit board is constructed such that adjacent lines of a circuit are shorted together

Causes of faults, cont.

- Component defects
 - Manufacturing imperfections, random device defects, component wear-out
 - Most commonly considered causes of faults
 - Examples: bonds breaking within the circuit, corrosion of the metal
- External disturbance
 - Radiation, electromagnetic interference, operator mistakes, environmental extremes, battle damage
 - Example: lightning

Failure modes

- **Failure domain**
 - Value failures : incorrect value delivered at interface
 - Timing failures : right result at the wrong time (usually late)
- **Failure consistency**
 - Consistent failures : all nodes see the same, possibly wrong, result
 - Inconsistent failures : different nodes see different results
- **Failure consequences**
 - Benign failures : essentially loss of utility of the system
 - Malign failures : significantly more than loss of utility of the system; catastrophic, e.g. airplane crash
- **Failure oftenness** (failure frequency and persistency)
 - Permanent failure : system ceases operation until it is repaired
 - Transient failure : system continues to operate
 - Frequently occurring transient failures are called intermittent

Failure modes, cont.

- **Consistent failures**
 - Fail-silent
 - system produces correct results or remains quiet (no delivery)
 - Fail-crash
 - system produces correct results or stops quietly
 - Fail-stop
 - system produces correct results or stops (made known to others)
- **Inconsistent failures**
 - Two-faced failures, malicious failures, Byzantine failures