

# 6. Reliability

**DR. B. I. MORSHED**

ASSOCIATE PROFESSOR  
COMPUTER SCIENCE  
TEXAS TECH UNIVERSITY

---

**Embedded Systems/Cyber Physical Systems**

**CS 4380 / CS 5331**

# Reliability: Chapter Outline

---

- Failure, Error, Fault
- Reliability
- Fault Tree Analysis (FTA)
- Fault Injection
- Test pattern
- Fault coverage
- Redundancy for reliability

# Failure, Error, Fault

---

*“A **failure (of service)**, often abbreviated here to **failure**, is an event that occurs when the delivered service of a system deviates from the correct service.”*

*“The definition of an **error** is the part of the total state of the system that may lead to its subsequent service failure”.*

*“The adjudged or hypothesized cause of an error is called a **fault**. Faults can be internal or external of a system.”*

Examples:

[Laprie et al., 1992, 2004]

- Transient **fault** flipping a bit in memory.
- After this bit flip, the memory cell will be in **error**.
- **Failure**: if the system service is affected by this error.

We will consider **failure** rates & **fault** models.

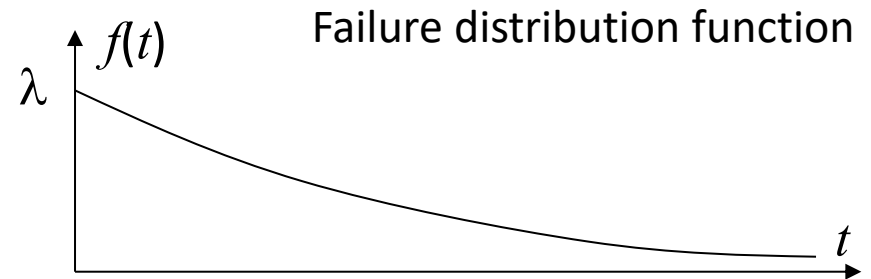
***A **fault** in a system causes an **error**, which may lead to a system **failure**.***

# Reliability: $f(t)$ , $F(t)$

- Let  $T$  be time until first failure (random variable)
- Let  $f(t)$  be the density function of  $T$

Example: Exponential distribution

$$f(t) = \lambda e^{-\lambda t}$$

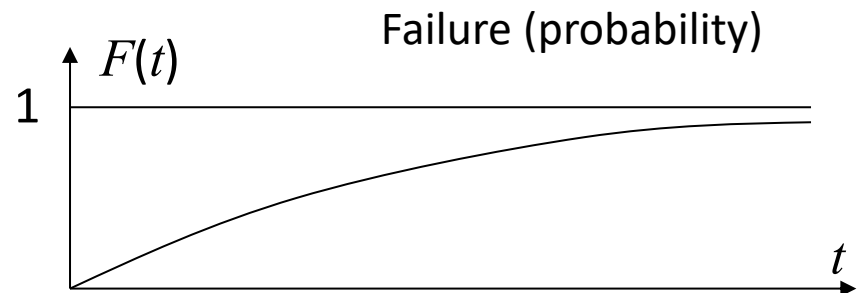


- $F(t)$  = probability of the system being faulty at time  $t$  :

$$F(t) = \Pr(T \leq t) \qquad F(t) = \int_0^t f(x) dx$$

Example: Exponential distribution

$$F(t) = \int_0^t \lambda e^{-\lambda x} dx = -[e^{-\lambda x}]_0^t = 1 - e^{-\lambda t}$$



# Reliability: $R(t)$

---

- **Reliability**  $R(t)$  = probability that the time until the first failure is larger than some time  $t$  :

$$R(t) = \Pr(T > t), t \geq 0$$

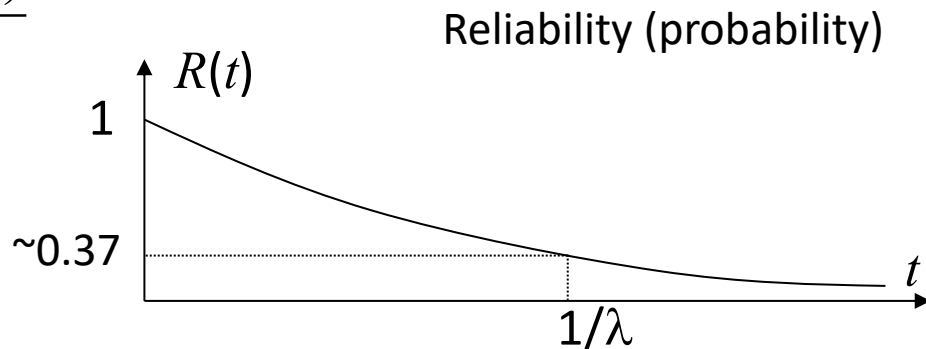
$$R(t) = \int_t^{\infty} f(x) dx$$

$$F(t) + R(t) = \int_0^t f(x) dx + \int_t^{\infty} f(x) dx = 1$$

$$R(t) = 1 - F(t) \quad f(t) = -\frac{dR(t)}{dt}$$

Example: Exponential distribution

$$R(t) = e^{-\lambda t}$$



# Failure Rate

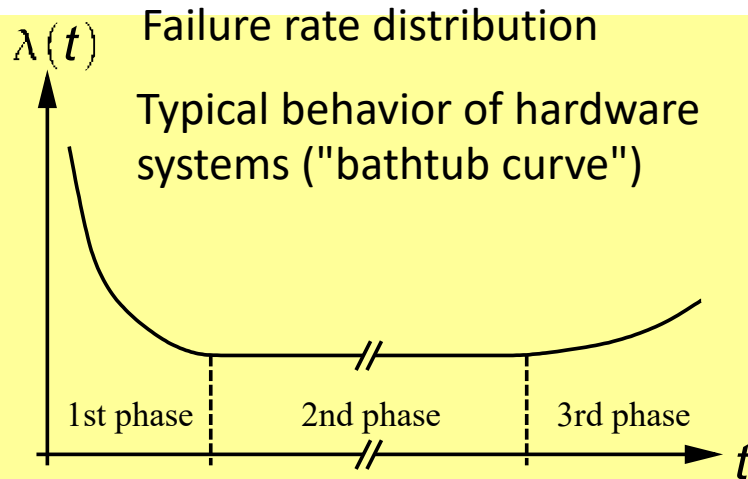
The failure rate at time  $t$  is the probability of the system failing between time  $t$  and time  $t+\Delta t$  :

$$\lambda(t) = \lim_{\Delta t \rightarrow 0} \frac{\Pr(t < T \leq t + \Delta t \mid T > t)}{\Delta t} = \lim_{\Delta t \rightarrow 0} \frac{F(t + \Delta t) - F(t)}{\Delta t R(t)} = \frac{f(t)}{R(t)}$$

Conditional probability ("provided that the system works at  $t$ ");

$$\Pr(A|B) = \Pr(AB) / \Pr(B)$$

Why salesman is so keen to sell insurance?



For exponential distribution:

$$\frac{f(t)}{R(t)} = \frac{\lambda e^{-\lambda t}}{e^{-\lambda t}} = \lambda$$

**FIT (Failure In Time)** = expected number of failures in  $10^9$  hrs.

# MTTF, MTTR and MTBF

MTTF = Mean Time To Failure

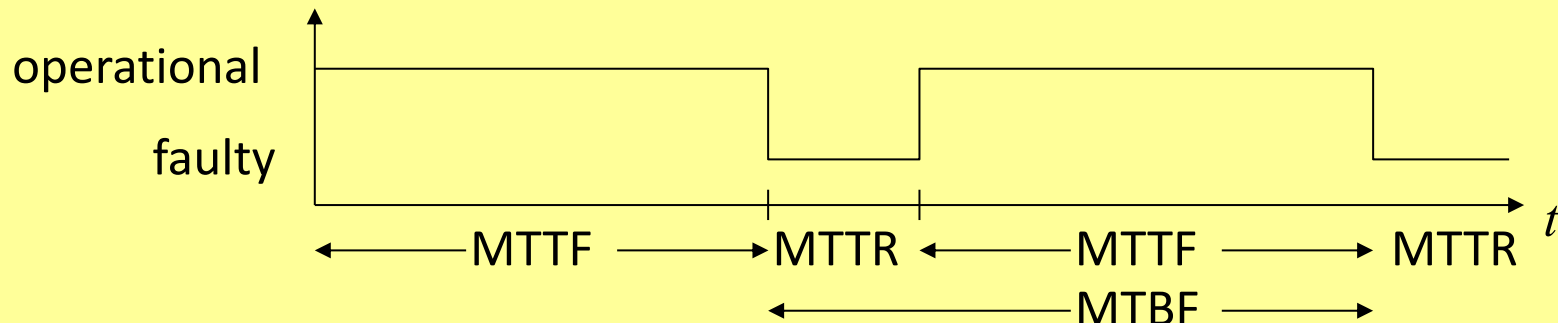
MTTR = Mean Time To Repair

(average over repair times using distribution  $M(d)$ )

MTBF = Mean Time Between Failures = MTTF + MTTR

$$\text{Availability } A = \lim_{t \rightarrow \infty} A(t) = \frac{\text{MTTF}}{\text{MTBF}}$$

□ Ignoring the statistical nature of failures ...



MTBF is often mixed up with MTTF, if starting in operational state is implicitly assumed

# Analysis of Failure or Fault

---

Mode of failure and remedy should be incorporated at the design time

- Reliable
- Robust

Several established methods:

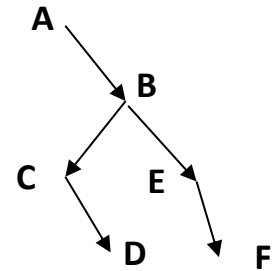
- Fault Tree Analysis (FTA)
- Failure Mode and Effect Analysis (FMEA)
- Fault injection method



# Fault Tree Analysis (FTA)

---

- ❑ Damages are resulting from hazards/risks.
- ❑ For every damage there is a **probability** and a **severity**.
- ❑ Several techniques for analyzing risks:
  - ❑ **FTA is a top-down method** of analyzing risks. Analysis starts with possible damage, tries to come up with possible scenarios that lead to that damage.
  - ❑ FTA typically uses a graphical representation of possible damages, including symbols for AND- and OR-gates.
  - ❑ OR-gates are used if a single event could result in a hazard.
  - ❑ AND-gates are used when several events or conditions are required for that hazard to exist.



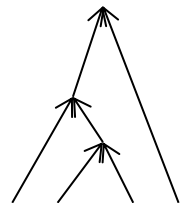
# Failure Mode and Effect Analysis (FMEA)

---

- FMEA starts at the components and tries to estimate their reliability. The first step is to create a table containing components, possible faults, probability of faults and consequences on the system behavior.

<i>Component</i>	<i>Failure</i>	<i>Consequences</i>	<i>Probability</i>	<i>Critical?</i>
...	...	...	...	...
Processor	metal migration	no service	$10^{-7}$ /h	yes
...	...	...	...	...

- Using this information, the reliability of the system is computed from the reliability of its parts (corresponding to a **bottom-up analysis**).



# Fault Injection

---

- ▣ Fault simulation may be too time-consuming
- ☞ If real systems are available, faults can be injected.



Two types of fault injection:

1. Local faults within the system (Internal),
2. Faults in the environment, such as behaviors which do not correspond to the specification (External).

For example, we can check how the system behaves if it is operated outside the specified temperature or radiation ranges.

# Software Fault Injection

---

Errors are injected into the memories (code or data).




Advantages:

- **Predictability:** it is possible to reproduce every injected fault in time and space.
- **Reachability:** possible to reach storage locations within chips instead of just pins.
- **Less effort** than physical fault injection: no modified hardware.

Same quality of results?

# Physical (HW) Fault Injection

- Hardware fault injection requires major effort, but generates precise information about the behavior of the real system.
- 3 techniques compared in the PDCS project on the MARS hardware [Kopetz]:

Injection Technique	Heavy-ion 	Pin-level 	EMI 
Controllability, space	Low	High	Low
Controllability, time	None	High/medium	Low
Flexibility	Low	Medium	High
Reproducibility	Medium	High	Low
Physical reachability	High	Medium	Medium
Timing measurement	Medium	high	Low

# Fault coverage

---

A certain set of test patterns will not always detect all faults that are possible within a fault model

$$(\textit{Fault-})\textit{Coverage} = \frac{\text{No. of detectable faults for a given test pattern set}}{\text{No. of faults possible due to the fault model}}$$

For actual designs, the fault coverage should be at least in the order of 98 to 99%.

Furthermore, non-faulty systems must be recognized as such  
**(correctness coverage)**

# Test pattern-based fault detection

---

## Test pattern generation typically

- considers certain fault models and
- generates patterns that enable a distinction between the faulty and the fault-free case.
- Examples:
  - Boolean differences
  - *D*-Algorithm
  - Self-test programs

# Other Hardware fault models

---

Fault models include:

- **stuck-open faults:**  
for CMOS, open transistors can behave like memories
- **delay faults:** circuit is functionally correct, but the delay is not.



# Dependability Requirements

---

Allowed failures may be in the order of 1 failure per  $10^9$  hours.  
~1000 times less than typical failure rates of chips.

- 👉 **For safety-critical systems, the system as a whole must be more dependable than any of its parts.**
- 👉 Fault-tolerance mechanisms must be used.

Low acceptable failure rates only for Systems not 100% testable and not safety critical.

- 👉 Safety must be shown by a **combination of testing and reasoning.**
- 👉 Abstraction must be used to make the system explainable using a hierarchical set of behavioral models. Design faults and human failures must be taken into account.

# Reliability & Redundancy

---

Reliability can either be defined as a characteristic for an item or as a performance measure.

## Reliability Block Diagram (RBD) technique

if A and B are two **independent events** with probabilities  $P(A)$  and  $P(B)$  of occurring, then the probability  $P(AB)$  that both events will occur is the product:

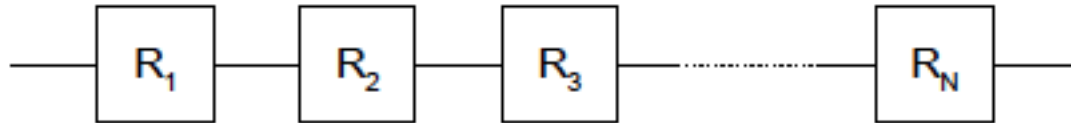
$$P(AB) = P(A).P(B)$$

if two events A and B are **mutually exclusive** so that when one occurs the other cannot occur, the probability that either A or B will occur is:

$$P(AB) = P(A) + P(B)$$

# Series System Have Low Reliability

---



The probability of survival of the system is the probability that all items survive.

$$R_S = R_1 \cdot R_2 \cdot R_3 \cdot \dots \cdot R_N$$

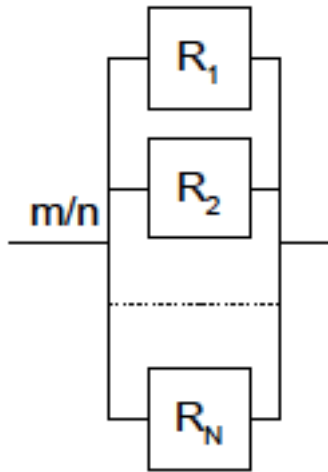
$$R_S = \prod_{i=1}^N R_i$$

When the  $R_i$  are all equal (to  $R$  say), then:

$$R_S = R^N$$

As  $R < 1$ , hence if  $N \uparrow$ , then  $R_S \downarrow$

# Improving reliability with Redundancy



The system is only failed when all items are failed.

The probability of an individual item failing is  $(1 - R)$ , so that  $P$ , the probability that all fail, is:

$$P_f = \prod_{i=1}^N (1 - R_i)$$

Since  $R_s = 1 - P_f$ :

$$R_s = 1 - \prod_{i=1}^N (1 - R_i)$$

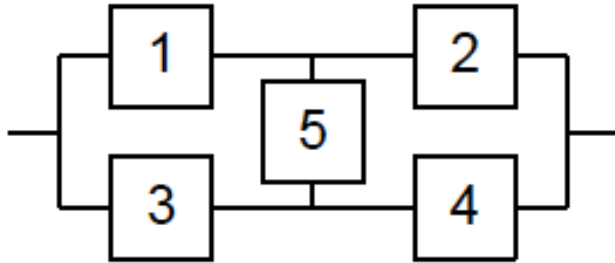
$$R_s = 1 - (1 - R_i)^N$$

When all the  $R_i$  are equal (to  $R$  say) then:

As  $R < 1$ , so  $(1 - R) < 1$ , hence if  $N \uparrow$ , then  $R_s \uparrow$

*Note:  $N$  of this slide is not the same as  $N$  of last slide.*

# System with complex redundancy

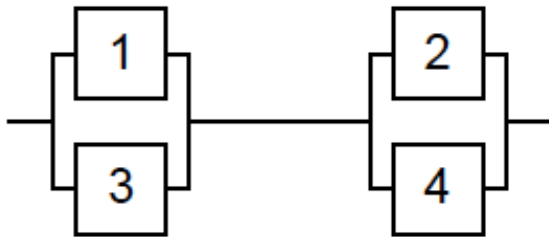


Bayes theorem :

$$P(B) = P(A) \cdot P(B|A) + P(\bar{A}) \cdot P(B|\bar{A})$$

Let,  $P(A) = R_5$ , and  $P(\bar{A}) = 1 - R_5$ ,

If block 5 survives:



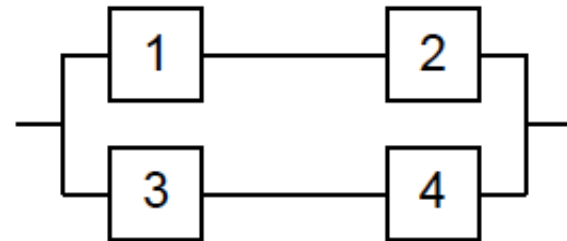
$$RS_{13} = 1 - (1 - R_1)(1 - R_3)$$

$$RS_{24} = 1 - (1 - R_2)(1 - R_4)$$

$$RS_{1234} = (1 - (1 - R_1)(1 - R_3)) \\ (1 - (1 - R_2)(1 - R_4))$$

$$RS' = R_5(1 - (1 - R_1)(1 - R_3)) \\ (1 - (1 - R_2)(1 - R_4))$$

If block 5 fails:



$$RS_{12} = R_1 R_2$$

$$RS_{34} = R_3 R_4$$

$$RS_{1234} = 1 - (1 - RS_{12})(1 - RS_{34}) \\ = 1 - (1 - R_1 R_2)(1 - R_3 R_4)$$

$$RS'' = (1 - R_5)(1 - (1 - R_1 R_2)(1 - R_3 R_4))$$

$$RS = R_5(1 - (1 - R_1)(1 - R_3))(1 - (1 - R_2)(1 - R_4)) + (1 - R_5)(1 - (1 - R_1 R_2)(1 - R_3 R_4))$$

# Incorporation of redundancy

---

## 1. Active Redundancy

- All parallel sub-system are active at all time
- Can be pure parallel or shared parallel form

## 2. Standby Redundancy

- Redundant sub-system becomes active upon failure of the first item
- Standby can be **hot** standby (active standby), **cold** standby (passive standby), or **warm** standby

## 3. R-out-of-n systems

- Sub-systems have n-items, in which r of them must function

# Chapter Summary

---

- Failure, Error, Fault
- Reliability and failure rate
- MTTF, MTTR, and MTBF
- Fault analysis
  - Fault tree analysis
  - Failure mode and effect analysis
  - Fault injection
- Fault coverage
- Test pattern
- Redundancy to improve reliability