Industrial Embedded Systems
- Design for Harsh Environment -

Dr. Alexander Walsch
alexander.walsch@ge.com

IN2244

Part IV – Reliability Quantification

WS 2014/15

Technische Universität München

Failure Rate (Hardware)

Failure Rate

A time dependent measure of #failures/time. Commonly only random failures are considered. The symbol for failure rate is λ(t). A failure rate is tied to a failure mode. This is a hardware related metric.

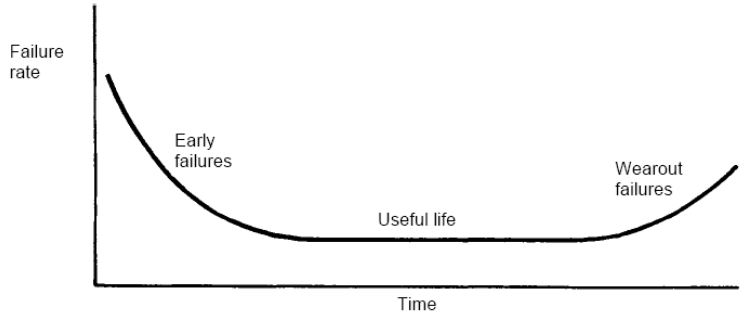
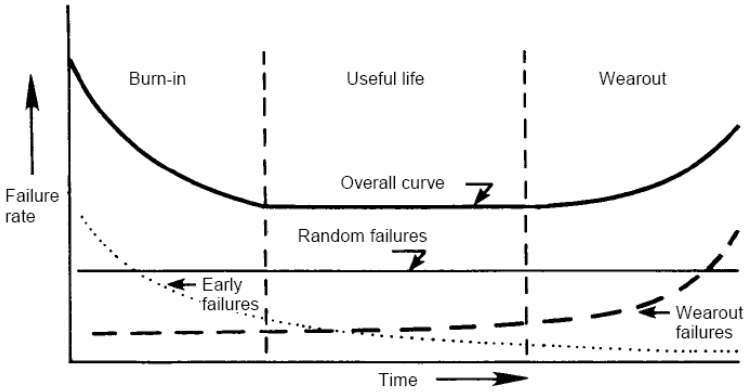


Figure 2.4

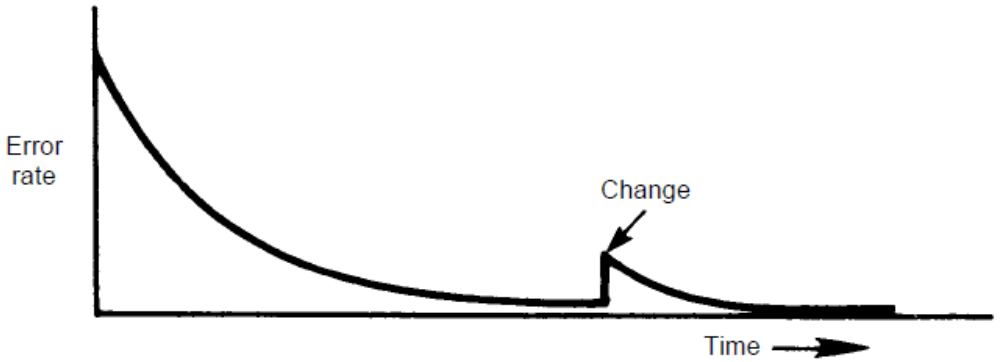


Source:

Smith: Reliability, Maintainability and Risk

Failure Rate (Software)

- A failure has been defined as deviation from the specification. This deviation can happen in two ways
  - Random (Hardware only) – happen randomly in time. The rate is predictable (statistical quantification). Previous slide.
  - Systematic (Hardware and Software) – linked to a certain cause (fault, defect, bug) which is present at time of commissioning
    They are not predictable. A rigorous design and qualification process must be applied.
  - On change (e.g. software update the error rate may increase)



Source:
Smith: Reliability, Maintainability and Risk

Reliability

Reliability

Reliability of a system or component is defined to be the probability that a given system or component will perform a required function under specified conditions for a specified period of time.

- "probability of non-failure (survival) in a given period"
- Reliability of a system function is modeled as: $R(t) = e^{-\lambda t}$ if the failure rate is constant.
- λ is often expressed as failures per 10^6 hours or FIT (failures per 10^9 hours).
- If "λt" small then R(t) = 1 - λt

Mean Time Between Failure (MTBF)

MTBF

Mean Time Between Failures (MTBF) is the average time a system will run between failures. The MTBF is usually expressed in hours.

Let us consider N items with k having failed at time t, T being the cumulative time.

$$N_s(t) = N - k \; ; number\ surviving\ at\ time\ t$$

$$R(t) = \frac{N_s(t)}{N}$$

$$T_{total} = \int_0^\infty N_s(t)\,dt$$

$$MTBF : \Theta = \int_0^\infty \frac{N_s(t)}{N}\,dt = \int_0^\infty R(t)\,dt = \int_0^\infty e^{-\lambda t}\,dt$$

$$\Theta = \lambda^{-1}, \lambda = const.$$

MTBF II

The observed MTBF (not all items have failed but k):

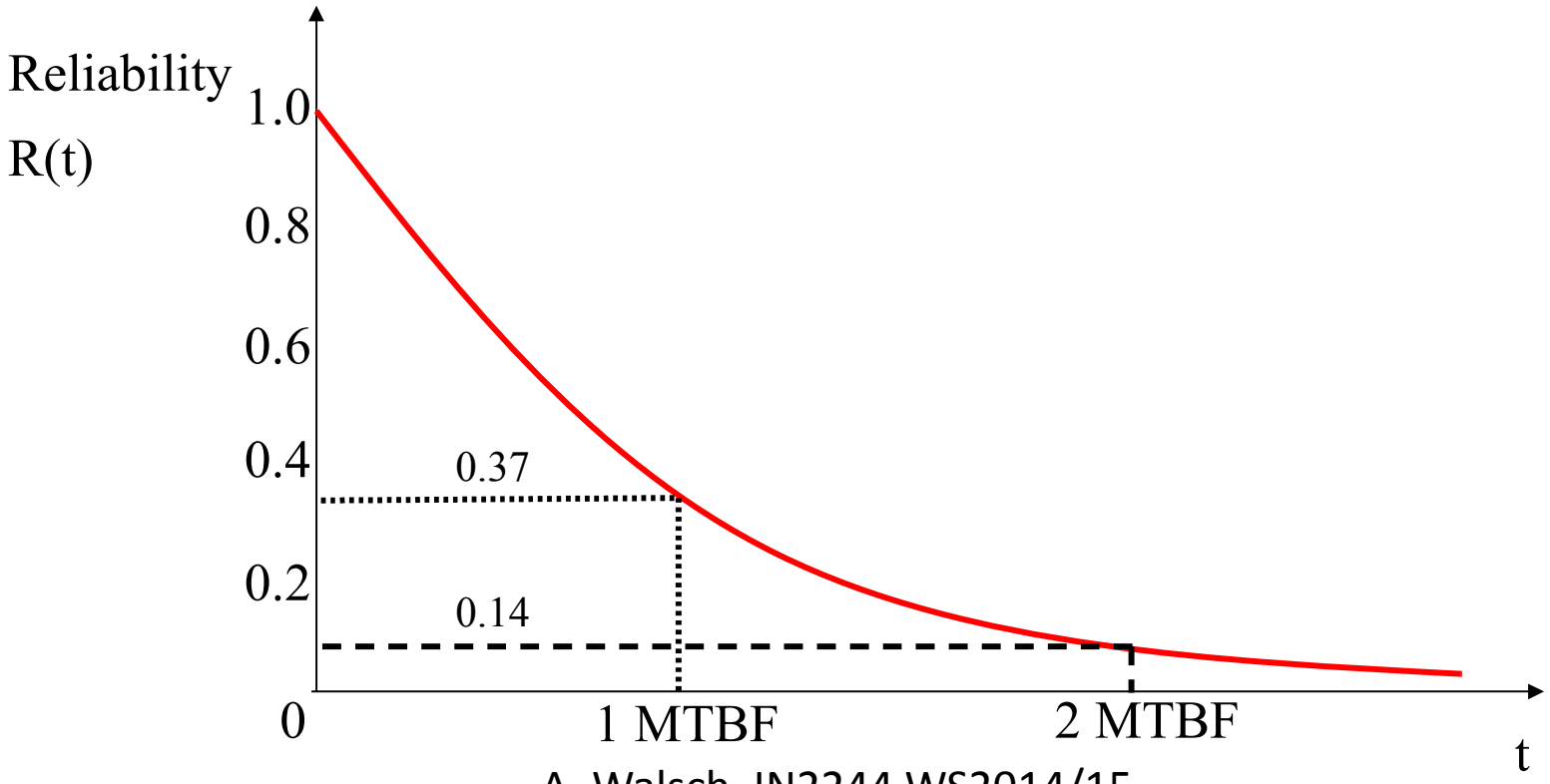
$$\hat{\Theta} = \frac{T}{k}$$ T = total time, k = failed items (total N)

Relation between Reliability and MTBF

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

$$t = \Theta \Rightarrow R = e^{-1} \approx 0.37$$

$$t = 2\Theta \Rightarrow R = e^{-2} \approx 0.14$$



Failure Rate Example

A system (S) has 10 components. Each component does have a failure rate of 5 per 10^6 hours (5000 FIT). Calculate the failure rate and MTBF of a function. Consider two cases:

- All components are required to perform the function (single point of failure).

- Each component performs a different function. Calculate the metrics for any of the functions.

We assume that there is only one failure mode for the component.

Failure Rate Example II

A) All components are required to perform the function (single point of failure)

λ_C = 5000 FIT

$\lambda_{function}$ = 10 * 5000 FIT = 50000 FIT ($5 * 10^{-5}$ failures/hour)

MTBF = 20000h

B) Each component performs a different function

$\lambda_C = \lambda_{function}$ = 5000 FIT = $5 * 10^{-6}$ failures/hour; MTBF = 200000h

MCU Example

Reliability Report - 2nd Quarter 2013 (CY) — **Publish Date: 12 Sep, 2013**

Reliability Die Monitor :

Search by Device : [ ] GO »

OR

Search by Device Family : dsPIC33F

OR

Search by Process : ---Select a Process---

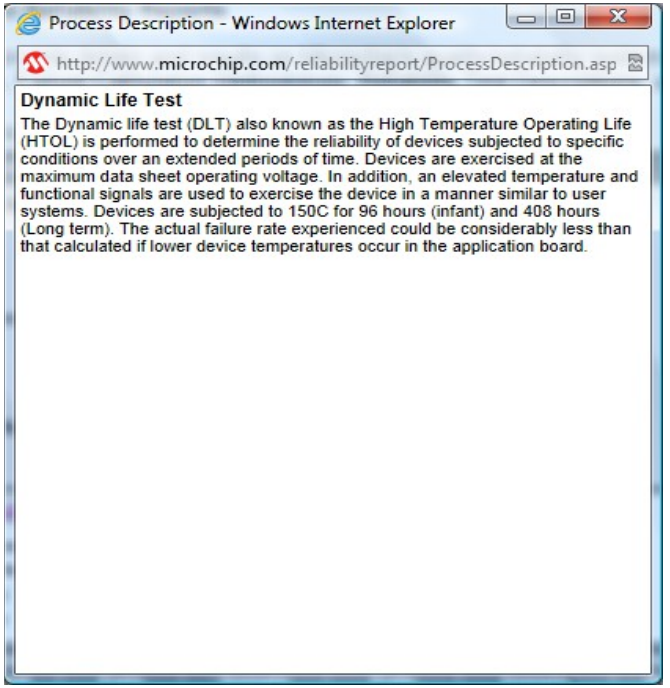
Reliability Package Monitor :

Search by Package : QFN

Dynamic Life Testing

Stress Temperature : 150 degrees C
Derated Temperature : 55 degrees C
Activation Energy : 0.7 eV
Acceleration Rate : 259

Device	Report Period	96 Hours Fails	96 Hours Sample	408 Hours Fails	408 Hours Sample	Device Hours	Total Life FIT Rate 60% Confidence	MTTF (Years)
dsPIC33F	YTD-13	0	3,080	0	3,075	1,255,080	3	40,509
	Rolling Yr - 12/13	1	6,158	0	6,141	2,507,160	3	36,665
	CUM 09-13	0	26,163	4	24,947	10,295,112	2	58,143
	QRT-13	0	1,680	0	1,675	683,880	5	22,073



Mean Down Time (MDT)

MDT

Mean Down Time (MDT) is the average time a system is in a failed state and can not execute its function.
MTBF can be understood as the mean up time.

MTTR

Mean Time to Repair (MTTR) is overlapping with MDT. Used for maintenance calculations. It can be visualized as the average time it takes (a technician) to repair the system such that it is up again.
We will not use MTTR in this lecture anymore.

For software the equivalent would be the time it takes to make a modification (e.g. bug fix, update) and install the new software function.

Availability

Availability

Availability is the probability that a system is functioning at any time during its scheduled working period (in percent).

$$A = \frac{up\,time}{total\,time} = \frac{up\,time}{up\,time + down\,time} = \frac{MTBF}{MTBF + MDT}$$

Reliability vs. Availability:

Reliability is inherent to a function given its specified conditions (internal properties). Availability takes failure and repair into account (internal and external properties).

Unavailability Example

$\lambda = 10^{-6}$ failures/hour ; MDT = 10h

Unavailability = ?

Unavailability Example

$\lambda = 10^{-6}$ failures/hour ; MDT = 10h

Unavailability = ?

$$U = \frac{down\,time}{total\,time} = \frac{MDT}{MTBF + MDT} \approx \lambda * MDT$$

$$\Rightarrow U = 10^{-5}$$

The Bernoulli Experiment applied to Reliability

We have a total number of n identical components. For each component only two states are defined: “functioning” or “has failed”. Both states have a certain probability assigned.
The Bernoulli experiment gives us the probability of finding k (out of n) components in a functioning state.

We state:

P(functioning) + P (failed) = 1 ;
P(functioning) = p; P (failed) = q

The Bernoulli Experiment II

The probability of k functioning components out of n total is

$$P(n, p, k) = \binom{n}{k} p^k q^{n-k}$$

Now we need the probability that a system function (spread across k components or sub-functions) is working -> reliability ("probability of survival")

$$P(n, p, k) = \binom{n}{k} R^k (1-R)^{n-k}$$

is the probability of having k functioning components in an assembly of n total.

Series Reliability Calculation

All n components above need to work such that the series assembly (system) is functioning.

The probability of having n functioning blocks out of n total is

$$R_S = P(n, n, k) = \binom{n}{n} R^n (1-R)^{n-n} = R^n$$

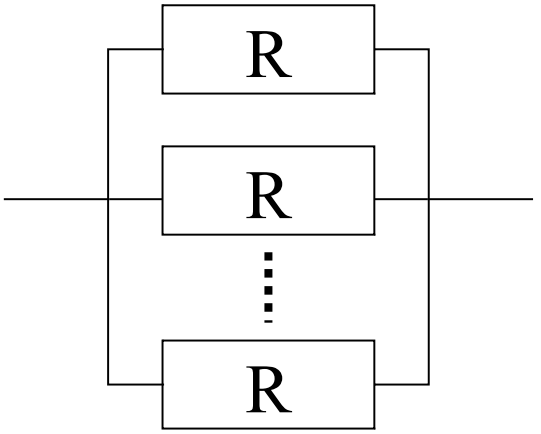
when using a Bernoulli experiment.

$$R_S = R * R * \ldots * R = R^n$$

when using the probability law for independent events.

Parallel Reliability Calculation
- Full active Redundancy -

At least 1 component needs to be functioning in full active redundancy configuration.



Therefore, the assembly is working if n or (n-1) or ... or 1 component work.

n=2: 2 or 1 component must be functioning.

$$R_S = \binom{2}{2} R^2 (1-R)^0 + \binom{2}{1} R^1 (1-R)^1 = 2\mathrm{R} - R^2 = R(2-R) > R$$

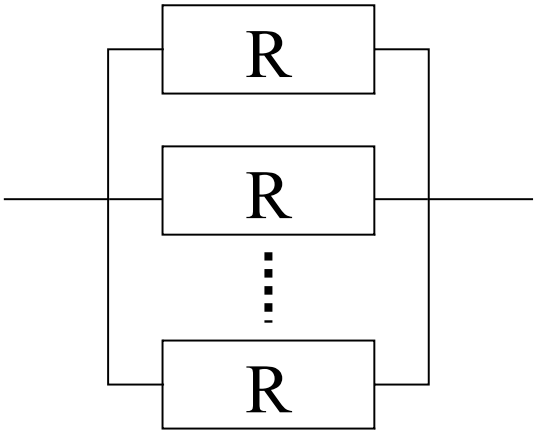
n=n:

$$R_S = \binom{n}{n} R^n (1-R)^0 + \ldots + \binom{n}{1} R^1 (1-R)^{n-1} = 1 - (1-R)^n$$

Parallel Reliability Calculation
- Partial active Redundancy -

At least m components need to be functioning in partial active redundancy configuration.

Therefore, the assembly is working if n or (n-1) or ... or m components work.



n=3: m = 2 (2oo3, spoken “two out of three”)

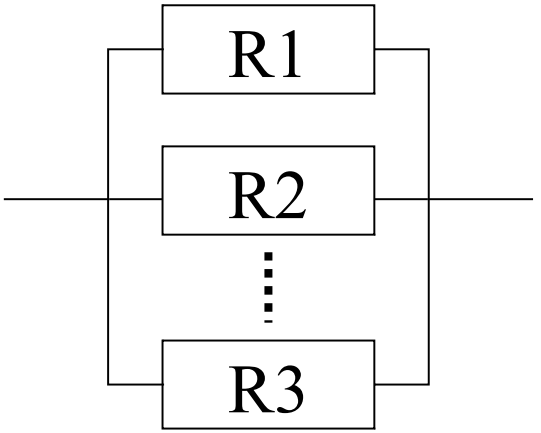
$$R_S = \binom{3}{3} R^3 (1-R)^0 + \binom{3}{2} R^1 (1-R)^1 = 3R^2 - 2R^3$$

n=N, m = M: (MooN, spoken “M out of N”)

$$R_S = \binom{n}{n} R^n (1-R)^0 + \ldots + \binom{n}{m} R^m (1-R)^{n-m}$$

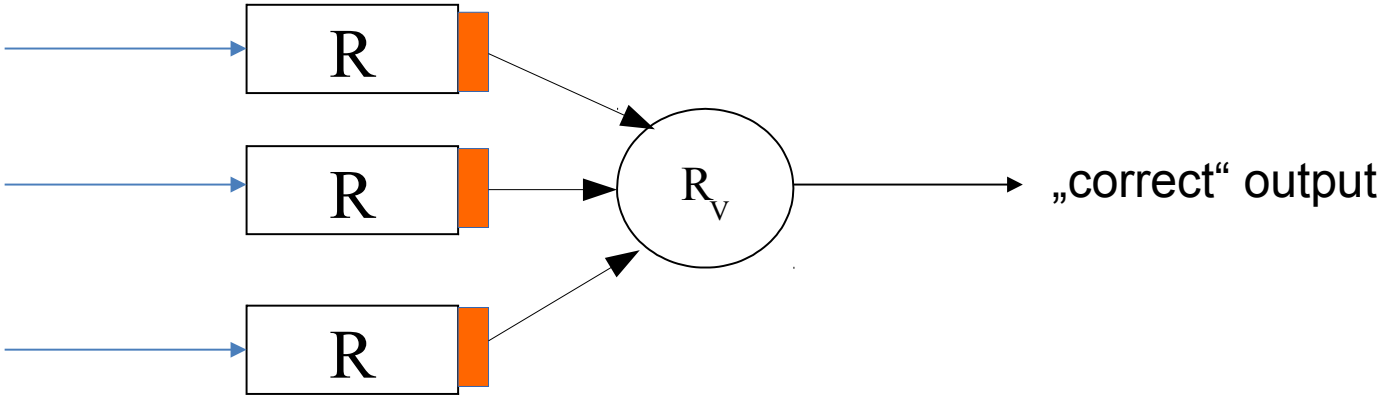
Replication and Diversity
- Avoidance of Common Cause Faults -

- Replication:
  identical copy of the original function
  (identical in specification for all phases
  of development and in implementation)



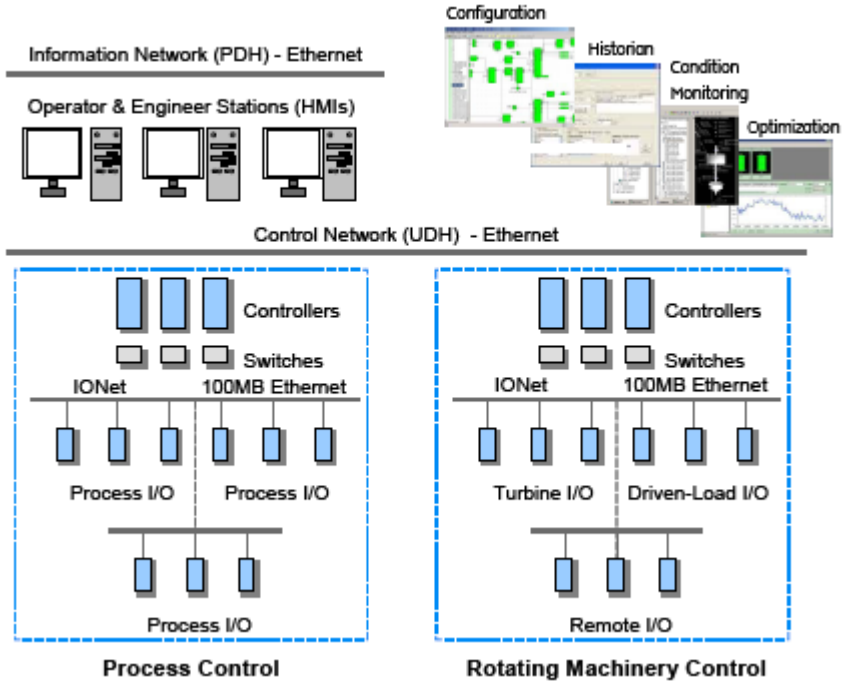
- Diversity:
  different copy of the original function
  (differences in specification and
  implementation – same interface to caller, same functional
  semantics – different behavioral semantics)
- From DO-178B (multiple dissimilar software, n-version programming): different programming languages, different compilers, dissimilar processor, different teams, different linkers and loaders, different design standards)

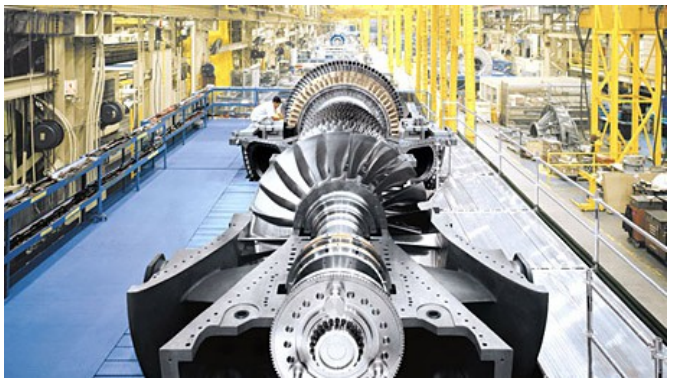
Partial Active Redundancy Example
- 2oo3 Majority Voter -



- Three inputs, one output: Triple Modular Redundancy (TMR)
- Input stages have reliability R, Voter and output stage have reliability R_V
- One unit may fail but no more (partial redundancy)
- Reliability: $R_S = 3R^2 - 2R^3 = R(3R - 2R^2) > R\,?$
- Adjudication method: majority, median, consensus

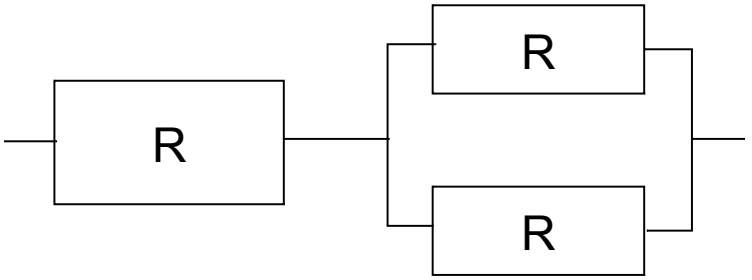
Partial Active Redundancy Example
- 2oo3 Majority Voter -





Source:
GE Energy

Complex Configuration Example

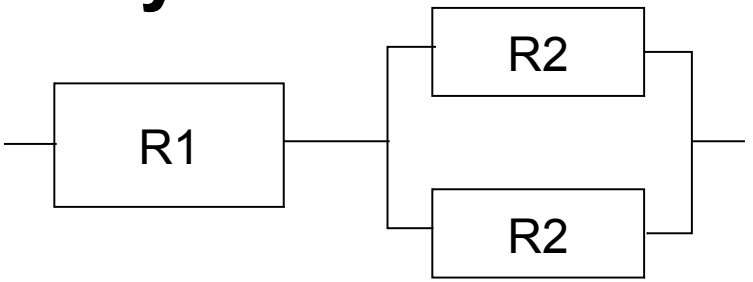


Calculate the MTBF of this system (S) made of identical components assuming constant failure rate and full active redundancy.

$$R_S = R\left(2\mathrm{R} - R^2\right) = 2\,e^{-2\lambda t} - e^{-3\lambda t}$$

$$\Theta = \int_0^\infty 2\,e^{-2\lambda t} - e^{-3\lambda t}\,dt = \ldots = \frac{2}{3\lambda}$$

Software Reliability



Definition:
Probability of failure-free software operation for a specified period of time in a specified environment (from „Standard Glossary of Software Engineering Terminology" STD-729-1991, ANSI/IEEE 1991)

Four Methods:

- Fault Prevention: avoid by construction (development: left wing of V-model)
- Fault Removal: detect by verification and validation (development: right wing of V-model)
- Fault Tolerance: provide service despite fault (operation)
- Fault Forecasting: estimate faults/failures by evaluation (future)