POLITECNICO
MILANO 1863

Dependable Systems

Dependability Analysis

Luca Cassano
luca.cassano@polimi.it
cassano.faculty.polimi.it/ds.html

Most of the material of these slides has been provided by Prof. Cristiana Bolchini, Politecnico di Milano, Italy

TOPIC QUESTIONS

How does the system react to the occurrence of a fault?

What are the most critical faults?

How reliable or available is the system?

Dependability analysis

Goal: estimate dependability-related properties

- Reliability (MTTF, fault coverage …)
- Availability
- …

Dependability analysis

Importance of design-time analysis

- to evaluate a design before production
- a metric to compare different designs
- to provide feedback to the designer during early design stages
- To certificate the system w.r.t. the considered safety standard (if necessary)

Reliability & Availability

Basic Concepts:

Failure Rate :

$$\lambda = \frac{\text{Failures per unit time}}{\text{Components exposed to functional failure}}$$

$$1\ \text{FIT} = 1 \times 10^{-9}\ \text{Failures per hour}$$

$$\text{MTBF} = \text{MTTF} + \text{MTTR}$$

$$\text{MTTF} = \text{MTBF} - \text{MTTR} = \frac{1}{\lambda}$$

$$\text{Availability} = \frac{\text{Operating Time}}{\text{Operating Time} + \text{Repair Time}} =$$

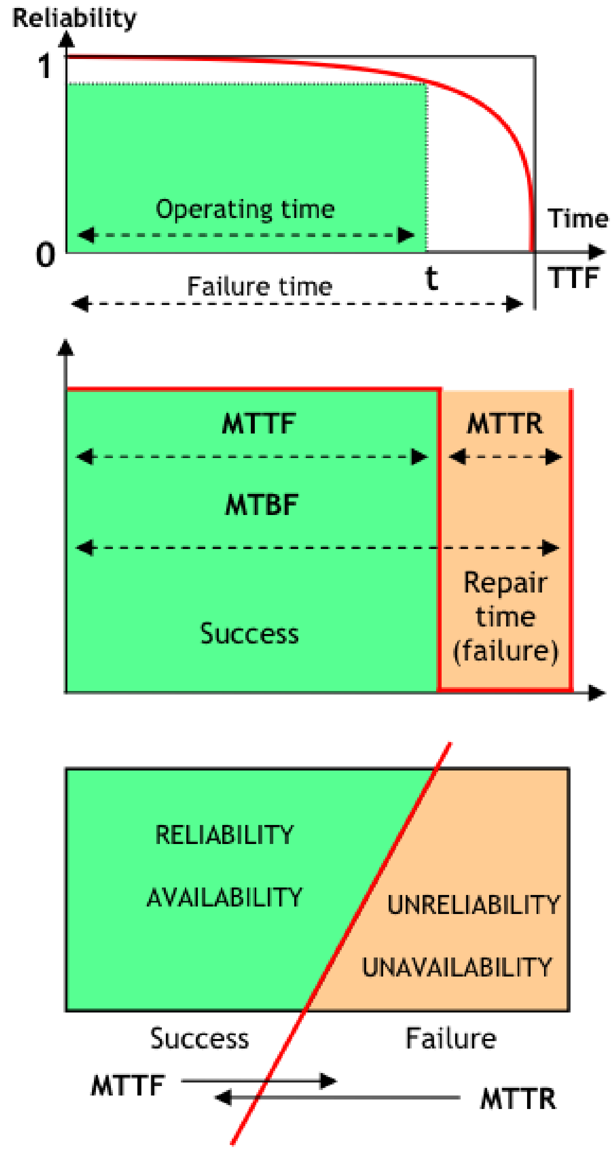
$$= \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}} = \frac{\text{MTTF}}{\text{MTBF}} = \frac{\mu}{\mu + \lambda} =$$

$$= \frac{\text{MTBM}}{\text{MTBM} + \text{MSD}}$$

$$\text{Unavailability} = 1 - \text{Availability} = \frac{\lambda}{\mu}$$

Acronyms:

MTBF: Mean Time Between Failures
MTTF: Mean Time To Failure
MTTR: Mean Time To Repair
MTBM: Mean Time Between Maintenance
MSD: Expected Mean System Downtime



Reliability-related analyses

Evaluation of the fault-error relationship

- For each fault, what are the effects (errors) and the consequent failures?
- And conversely, for each failure, which are the possible causes?

Compute/estimate reliability/availability metrics starting from the system components and adopted fault models

- MTTF: if you need to measure the operating time of the system
- Fault coverage: if you need to measure how many faults will the system tolerate
- ….

Fault models (in a nutshell)

Models of the effects of faults occurring in the components of a system

For example:

- You may model the effect of a transistor break within a circuit as a signal stuck at 0/1 (the stuck-at fault model)

- You may model the effect of a radioactive particle hitting a memory cell as the change of the content of the cell (bit-flip fault model)

Analysis approaches

Forward
Starting from a set of events the effects of these events on the system are evaluated …

Backward
Starting from the observed malfunctioning behaviors, possible causes (events) are analyzed and identified

Forward & Backward Analysis

Failure Mode and Effects Analysis (FMEA) exploits the forward approach
… given these events, what will happen?

Fault Tree Analysis (FTA) follows the backward method
… what are the events that cause the observed failure?

Forward & Backward Analysis

In both cases the goal is

to identify a causal relationship between events and failures

Events include failures in

- Hardware/Software
- Human behavior
- Environmental conditions

Analytical techniques

- Reliability Block Diagrams – RBD
- Fault tree analysis – FTA
- *Failure modes and effects analysis – FMEA*
- *Failure modes, effects and criticality analysis FMECA*
- *Failure modes, effects and diagnostic analysis FMEDA*
- Hazard and operability studies – HAZOP
- Event tree analysis – ETA
- Risk analysis – RA

Reliability Block Diagrams

Reliability Block Diagrams

An inductive model where a system is divided into blocks that represent distinct elements such as components or subsystems.

Reliability Block Diagrams

An inductive model where a system is divided into blocks that represent distinct elements such as components or subsystems.

Every element in the RBD has its own reliability (previously calculated or modelled)

Reliability Block Diagrams

An inductive model where a system is divided into blocks that represent distinct elements such as components or subsystems.

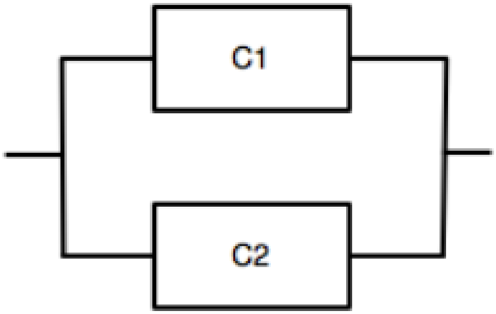
Every element in the RBD has its own reliability (previously calculated or modelled)

Blocks are then combined together to model all the possible *success paths*

Reliability Block Diagrams

RBDs are an approach to compute the reliability of a system starting from the reliability of its components

components in series



components in parallel

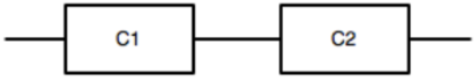
All components must be healthy for the system to work properly

If one component is healthy the system works properly

Reliability Block Diagrams

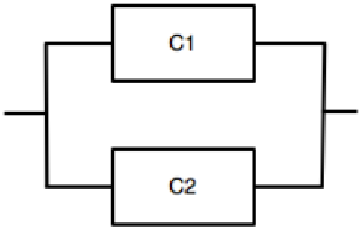
Series:

$$R_S(t) = R_{C1}(t) * R_{C2}(t)$$



Parallel:

$$R_S(t) = 1 - \left[(1 - R_{C1}(t)) * (1 - R_{C2}(t))\right]$$



$$R_S(t) = R_{C1}(t) + R_{C2}(t) - R_{C1}(t) * R_{C2}(t)$$

Reliability Block Diagrams

series

In general, if system S is composed by components with a reliability having an exponential distribution (very common case)

Failure in time

$$R_s(t) = e^{-\lambda_s t}$$

where

$$\lambda_s = \sum_{i=1}^{n} \lambda_i$$

Reliability Block Diagrams

series

In general, if system S is composed by components with a reliability having an exponential distribution (very common case)

Failure in time

$$R_s(t) = e^{-\lambda_s t} \quad \text{where} \quad \lambda_s = \sum_{i=1}^{n} \lambda_i$$

$$MTTF_S = \frac{1}{\lambda_S} = \frac{1}{\sum_{i=1}^{n} \lambda_i} = \frac{1}{\sum_{i=1}^{n} \frac{1}{MTTF_i}}$$

Reliability Block Diagrams

A special case: when all components are identical

$$R_s(t) = e^{-\lambda_s t}$$

$$R_S(t) = e^{-n\lambda t} = e^{-\frac{nt}{MTTF_1}} \qquad MTTF_S = \frac{MTTF_1}{n}$$

Reliability Block Diagrams

series

Availability:

$$A_S = \prod_{i=1}^{n} \frac{MTTF_i}{MTTF_i + MTTR_i}$$

When all components are the same:

$$A_S(t) = A_1(t)^n \qquad A = \left(\frac{MTTF_1}{MTTF_1 + MTTR_1} \right)^n$$

Reliability Block Diagrams

parallel

System P composed by *n* components

$$R_P(t) = 1 - \prod_{i=1}^{n} \left(1 - R_i(t)\right)$$

Availability

$$A_P(t) = 1 - \prod_{i=1}^{n} \left(1 - A_i(t)\right)$$

$$A_P = 1 - \prod_{i=1}^{n} \left(1 - A_i\right) = 1 - \prod_{i=1}^{n} \frac{MTTR_i}{MTTF_i + MTTR_i}$$

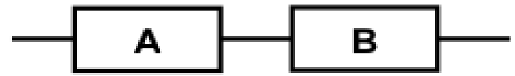
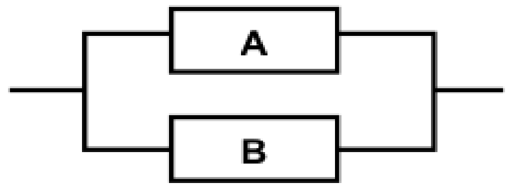
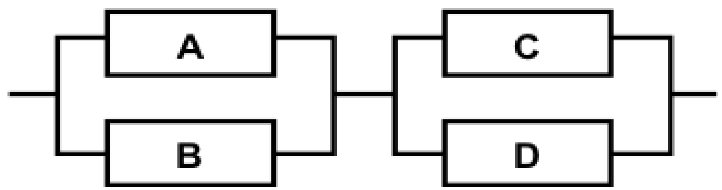
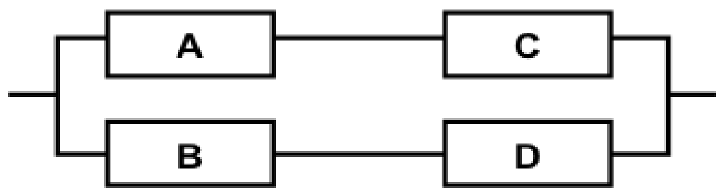
Reliability Block Diagrams (recap)

$$R_s = \prod_{i}^{n} R_i$$

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

Component redundancy

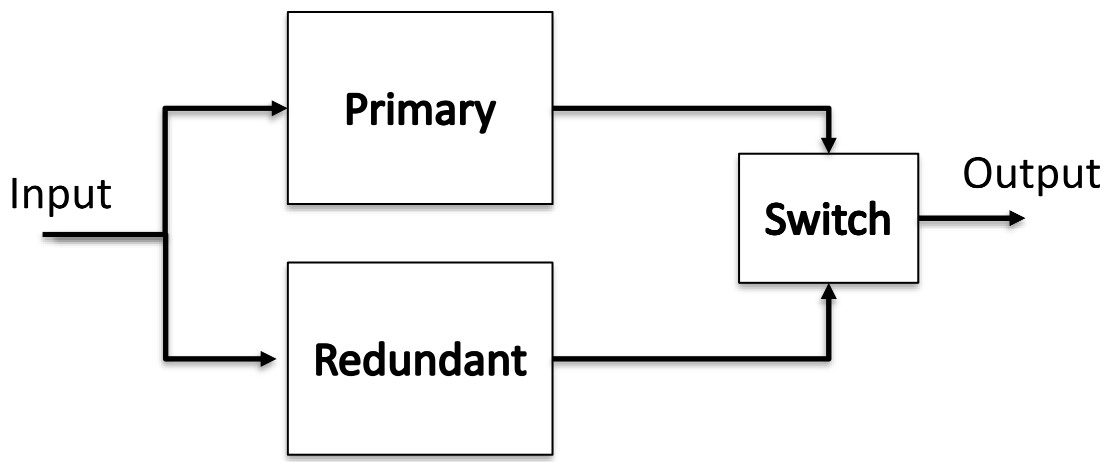
System redundancy

Type	Block Diagram Representation	System Reliability (R_S)
Series		$R_S = R_A R_B$ R_A = reliability, component A R_B = reliability, component B
Parallel		$R_S = 1-(1-R_A)(1-R_B)$
Series-Parallel		$R_S = [1-(1-R_A)(1-R_B)]*$ $[1-(1-R_C)(1-R_D)]$ R_C = reliability, component C R_D = reliability, component D
Parallel-Series		$R_S = 1-(1-R_A R_C)*$ $(1-R_B R_D)$

Standby redundancy

A system may be composed of two parallel replicas:

- The primary replica working all time, and
- The redundant replica (generally disable) that is activated when the primary replica fails



Standby redundancy

A system may be composed of two parallel replicas:

- The primary replica working all time, and
- The redundant replica (generally disable) that is activated when the primary replica fails

What do we need for such a redundancy to be operational?

Standby redundancy

A system may be composed of two parallel replicas:

- The primary replica working all time, and
- The redundant replica (generally disable) that is activated when the primary replica fails

Obviously we need:

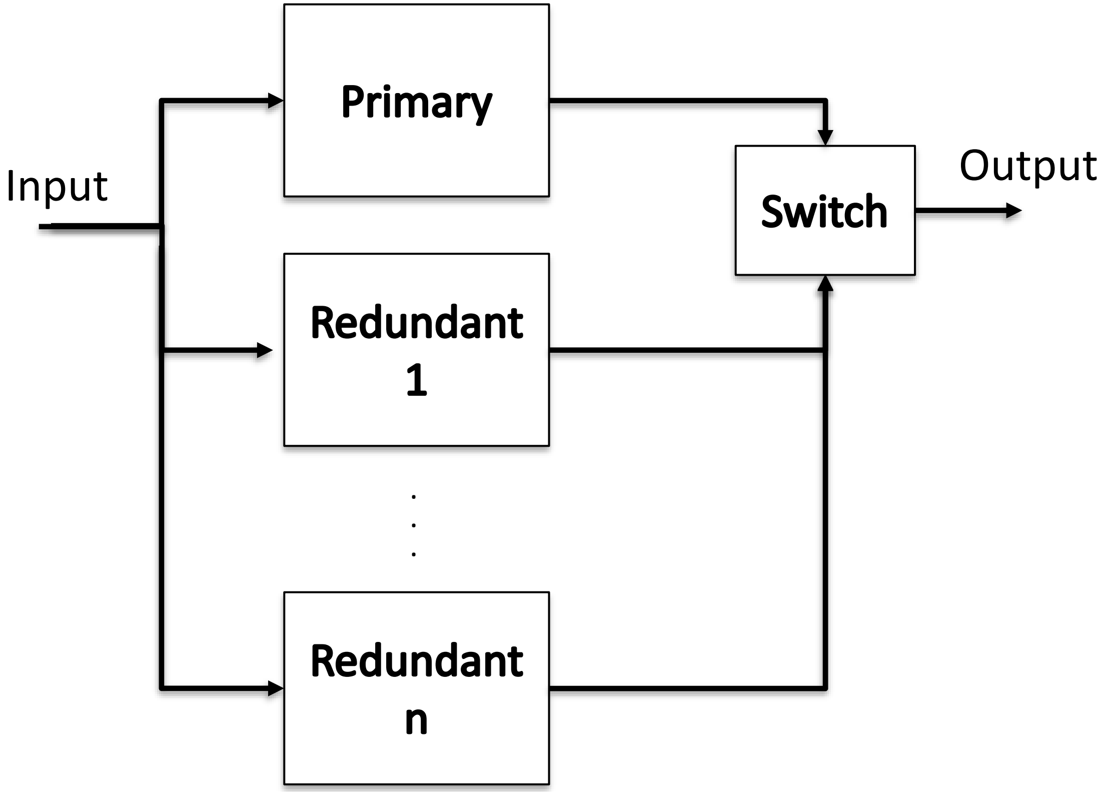
- A mechanism to determine whether the primary replica is working properly or not (on-line self check)
- A dynamic switching mechanism to disable the primary replica and activate the redundant one

Standby redundancy

Standby Parallel Model	**System Reliability**
Equal failure rates, perfect switching	$R_s = e^{-\lambda t}(1+\lambda t)$
Unequal failure rates, perfect switching	$R_s = e^{-\lambda_1 t} + \lambda_1 (e^{-\lambda_1 t} - e^{-\lambda_2 t}) / (\lambda_2 - \lambda_1)$
Equal failure rates, imperfect switching	$R_s = e^{-\lambda t}(1+ R_{switch} \lambda t)$
Unequal failure rates, imperfect switching	$R_s = e^{-\lambda_1 t} + R_{switch} \lambda_1 (e^{-\lambda_1 t} - e^{-\lambda_2 t}) / (\lambda_2 - \lambda_1)$
where R_s = System reliability λ = Failure rate t = Operating time R_{switch} = Switching reliability	

Standby redundancy

More in general, a system having one primary replica and *n* redundant replicas (with identical replicas and perfect switching)



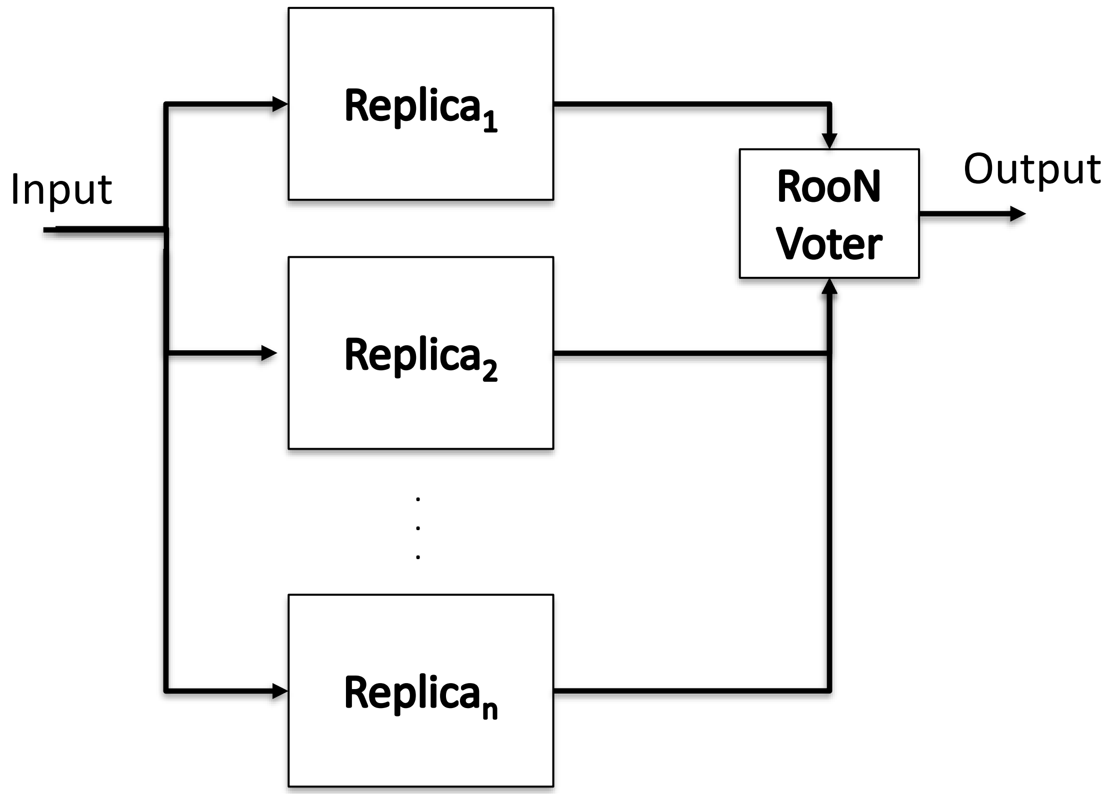
Standby redundancy

More in general, a system having one primary replica and *n* redundant replicas (with identical replicas and perfect switching)

$$R(t) = e^{-\lambda t} \sum_{i=0}^{n-1} \frac{(\lambda t)^i}{i!}$$

r out of *n* redundancy (RooN)

A system composed of *n* identical replicas where at least *r* replicas have to work fine for the entire system to work fine



r out of *n* redundancy (RooN)

RBDs

R_s = System reliability
R_c = Component reliability
R_V= Voter Reliability
n = Number of components
r = Minimum number of components which must survive

$$R_S(t) = RV \sum_{i=r}^{n} R_c^i (1 - R_C)^{n-i} \frac{n!}{i!\,(n-i)!}$$

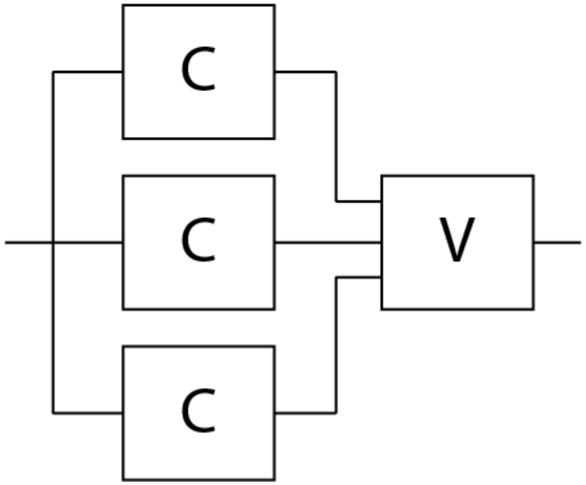
r out of *n* redundancy (RooN)

R_s = System reliability
R_c = Component reliability
R_V= Voter Reliability
n = Number of components
r = Minimum number of components which must survive

$$R_S(t) = RV \sum_{i=r}^{n} R_c^i (1 - R_C)^{n-i} \frac{n!}{i!\,(n-i)!}$$

Binomial coefficient
$\binom{n}{i}$

Triple Modular Redundancy – TMR



System works properly if

- 2 out of 3 components work properly AND the voter works properly

$$R_{TMR} = R_v[\sum_{i=2}^{3}\binom{3}{i}R_m^i(1-R_m)^{3-i}] = R_v[R_m^3 + 3R_m^2(1-R_m)] = R_v(3R_m^2 - 2R_m^3)$$

$$MTTF_{TMR} = \int_0^\infty R_{TMR}dt = \int_0^\infty R_v(3R_m^2 - 2R_m^3)dt = \int_0^\infty e^{-\lambda_v t}(3e^{-2\lambda_m t} - 2e^{-3\lambda_m t})dt$$

$$= \frac{3}{2\lambda_m + \lambda_v} - \frac{2}{3\lambda_m + \lambda_v} \cong \frac{3}{2\lambda_m} - \frac{2}{3\lambda_m} = \left(\frac{5}{6}\right)\left(\frac{1}{\lambda_m}\right) = \frac{5}{6}MTTF_{simplex}$$

TMR

- $MTTF_{TMR}$ is shorter than $MTTF_{symplex}$
- Can tolerate transient faults and permanent faults
- Higher reliability (for shorter missions)

When do we have the same reliability?

- $R_{TMR}(t) = R_C(t)$

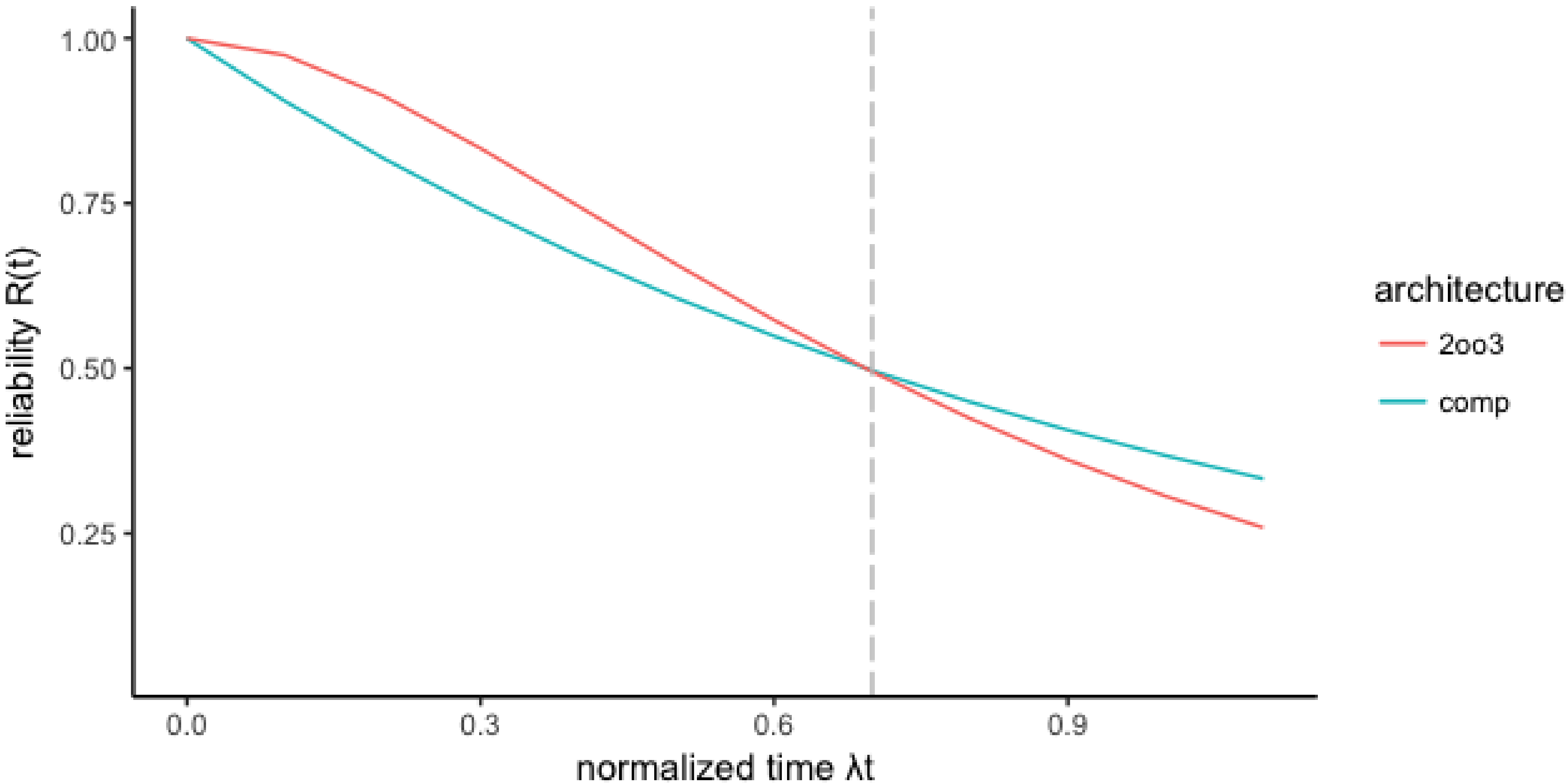
$$3e^{-2\lambda_m t} - 2e^{-3\lambda_m t} = e^{-\lambda_m t}$$

$$t = \frac{\ln 2}{\lambda_m} \cong 0.7 \text{ MTTF}_C$$

➡ $R_{TMR}(t) > R_C(t)$ when the mission time is shorter than 70% of $MTTF_C$

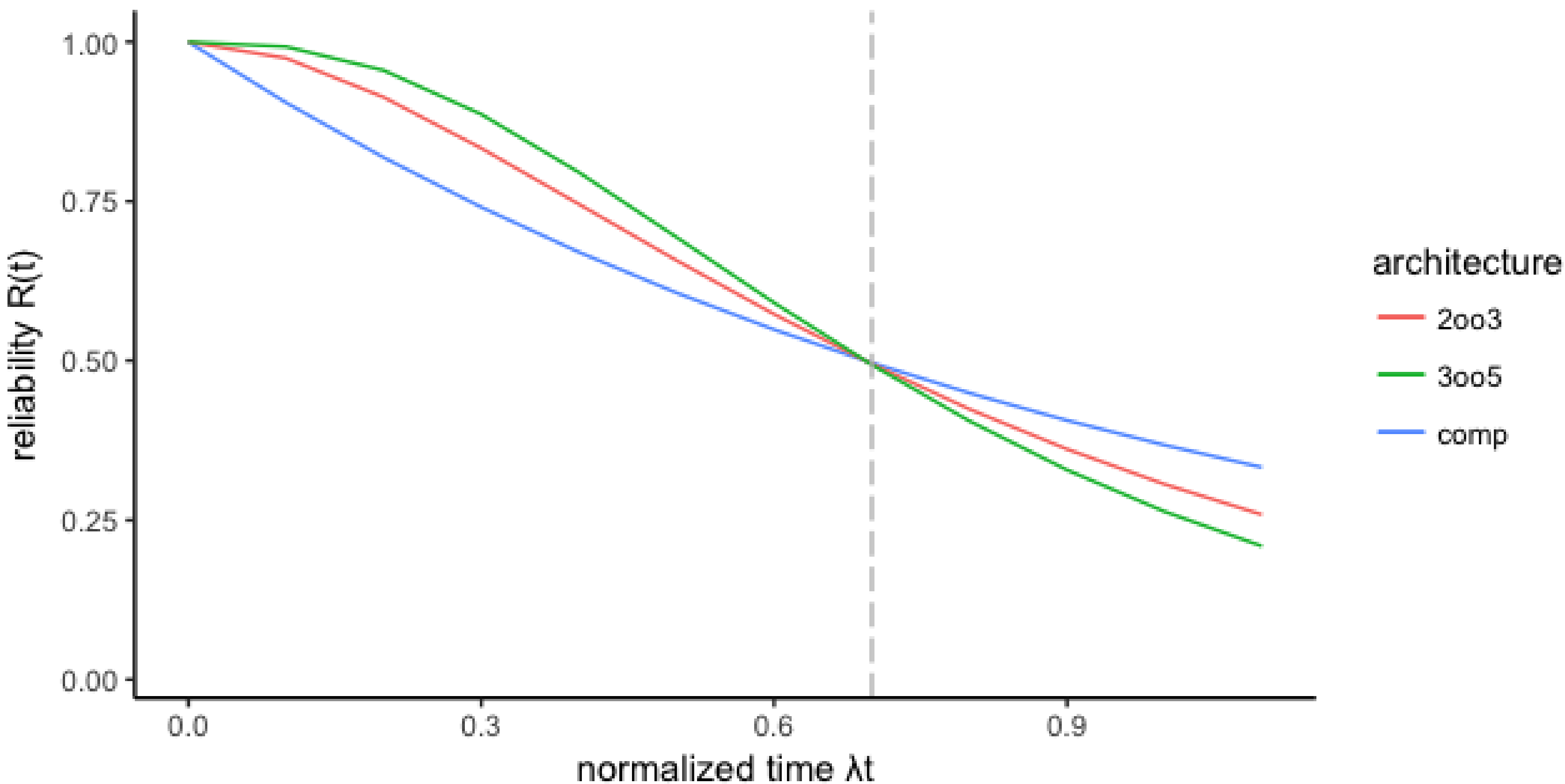
TMR

TMR: 2 out of 3 components (voter is a ‘perfect’ element)



nMR

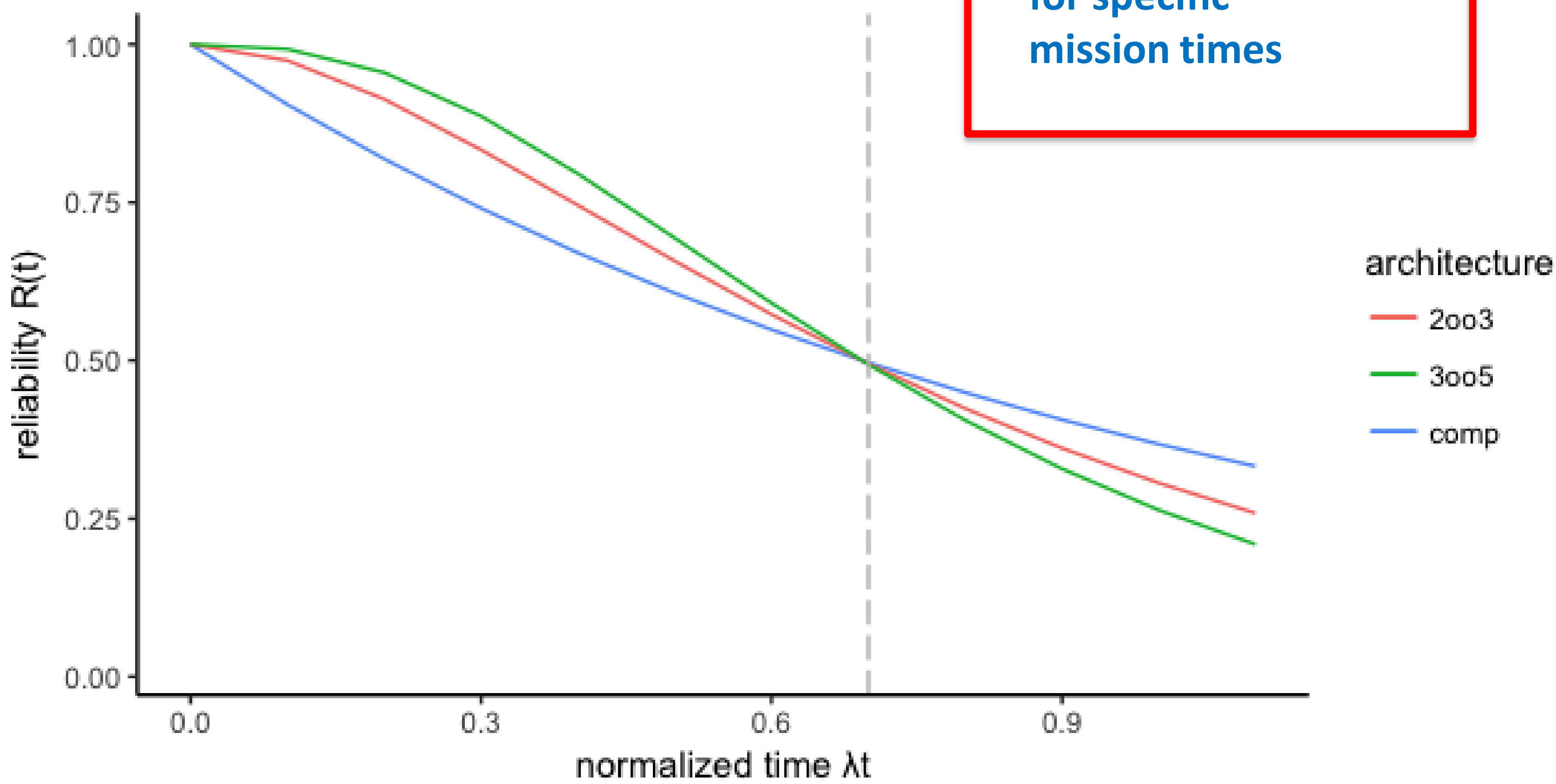
TMR: 2oo3 and nMR: 3oo5



nMR

RBDs

TMR: 2oo3 and nMR: 3oo5



Redundancy is useful for specific mission times

Example 1

$R_A = 0.95$

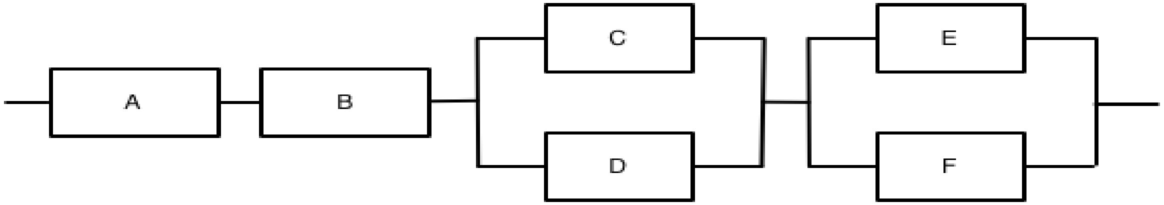
$R_B = 0.97$

$R_C = 0.99$

$R_D = 0.99$

$R_E = 0.92$

$R_F = 0.92$



Example 1

$R_A = 0.95$

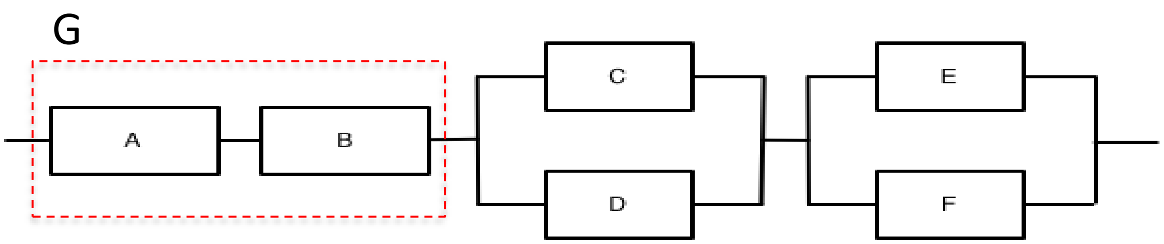
$R_B = 0.97$

$R_C = 0.99$

$R_D = 0.99$

$R_E = 0.92$

$R_F = 0.92$



$R_G = R_A * R_B$

$R_G = 0.9215$

Example 1

RBDs

$R_A = 0.95$

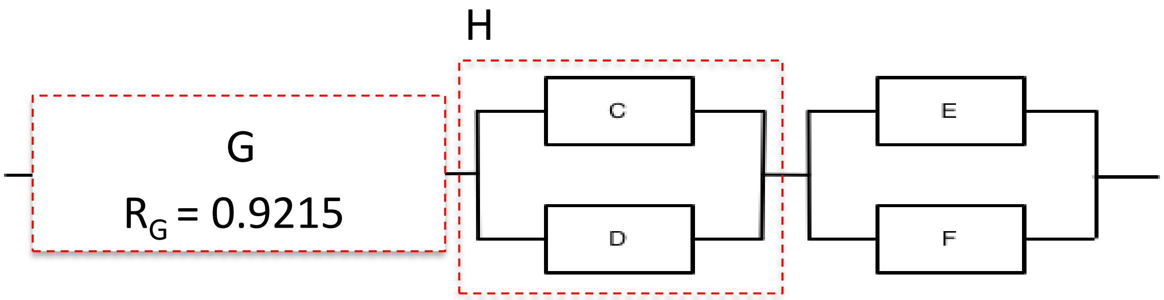
$R_B = 0.97$

$R_C = 0.99$

$R_D = 0.99$

$R_E = 0.92$

$R_F = 0.92$



$R_H = 1-[(1-R_c)*(1-R_D)]$

$R_H = 0.9999$

Example 1

$R_A = 0.95$

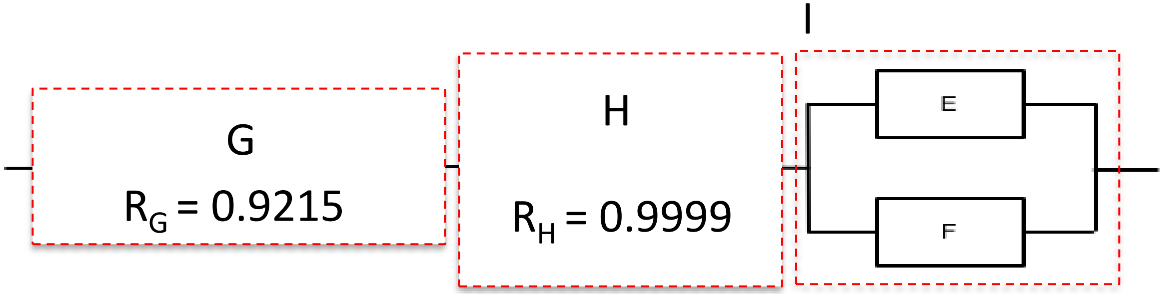
$R_B = 0.97$

$R_C = 0.99$

$R_D = 0.99$

$R_E = 0.92$

$R_F = 0.92$



$R_I = 1-[(1-R_E)*(1-R_F)]$

$R_I = 0.9936$

Example 1

RBDs

$R_A = 0.95$

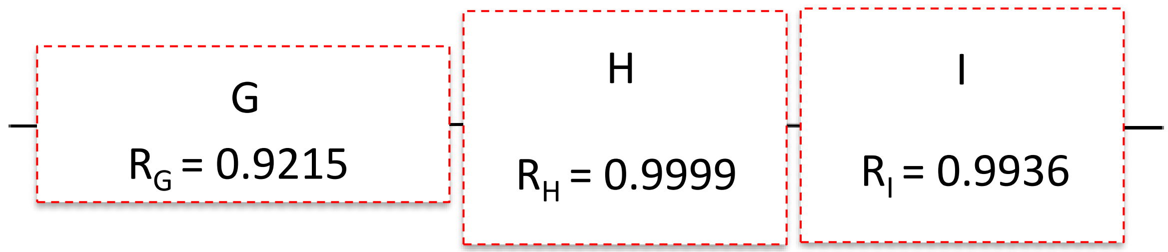
$R_B = 0.97$

$R_C = 0.99$

$R_D = 0.99$

$R_E = 0.92$

$R_F = 0.92$

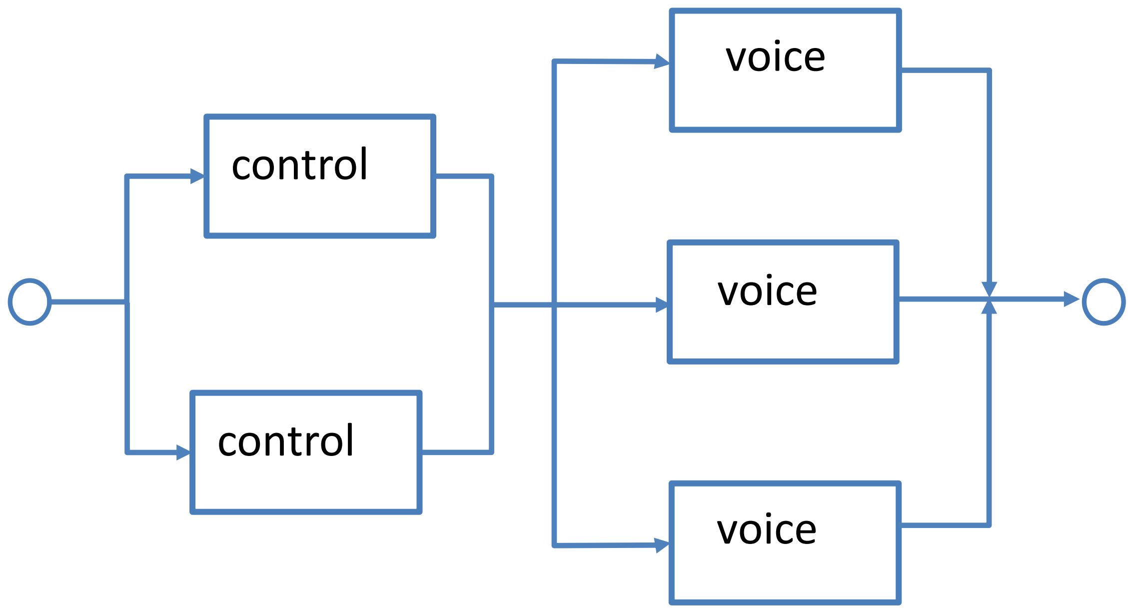


$R_S = R_G * R_H * R_I = 0.9155$

Example 2

2 control blocks and 3 voice channels:

- system is up if at least 1 control channel and at least 1 voice channel are up



Example 2 – cont'd

- Each control channel has reliability R_c
- Each voice channel has reliability R_v

- Reliability:

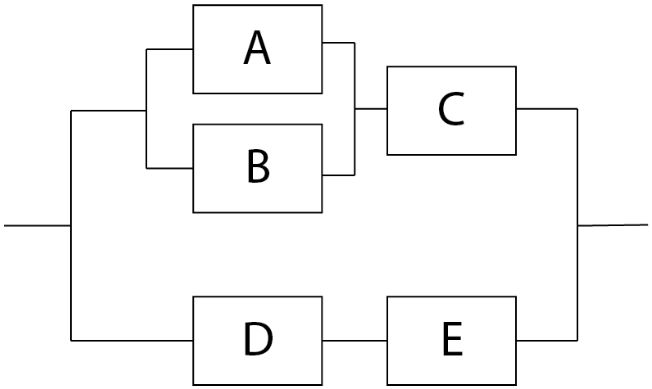
Example 2 – cont'd

- Each control channel has reliability R_c
- Each voice channel has reliability R_v

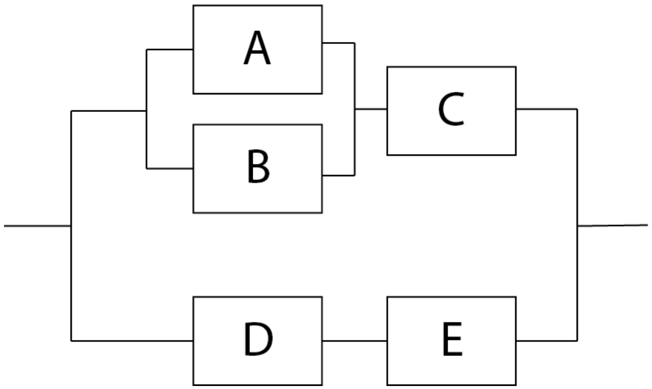
- Reliability:

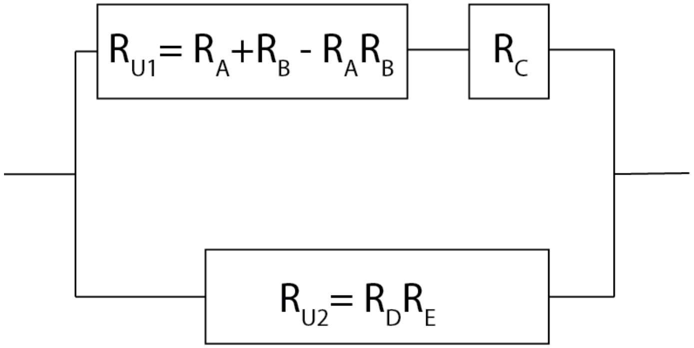
$$R = [1-(1-R_c)^2][1-(1-R_v)^3]$$

Example 3

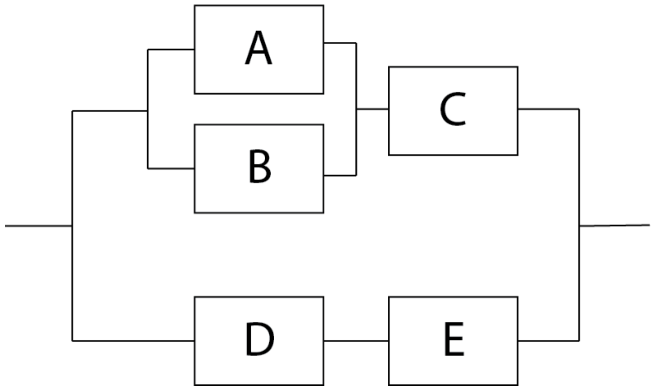


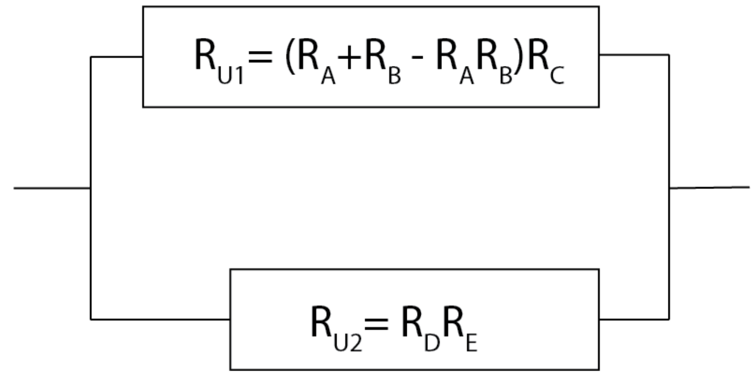
Example 3

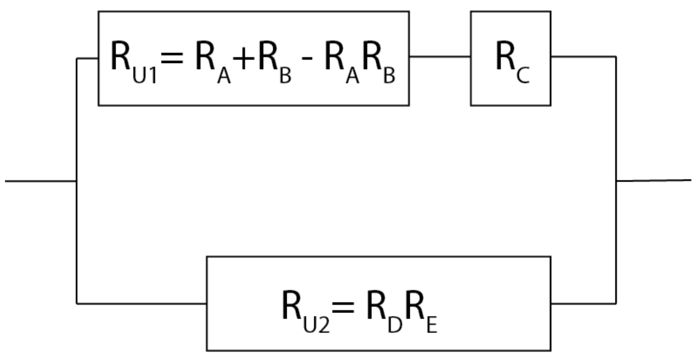




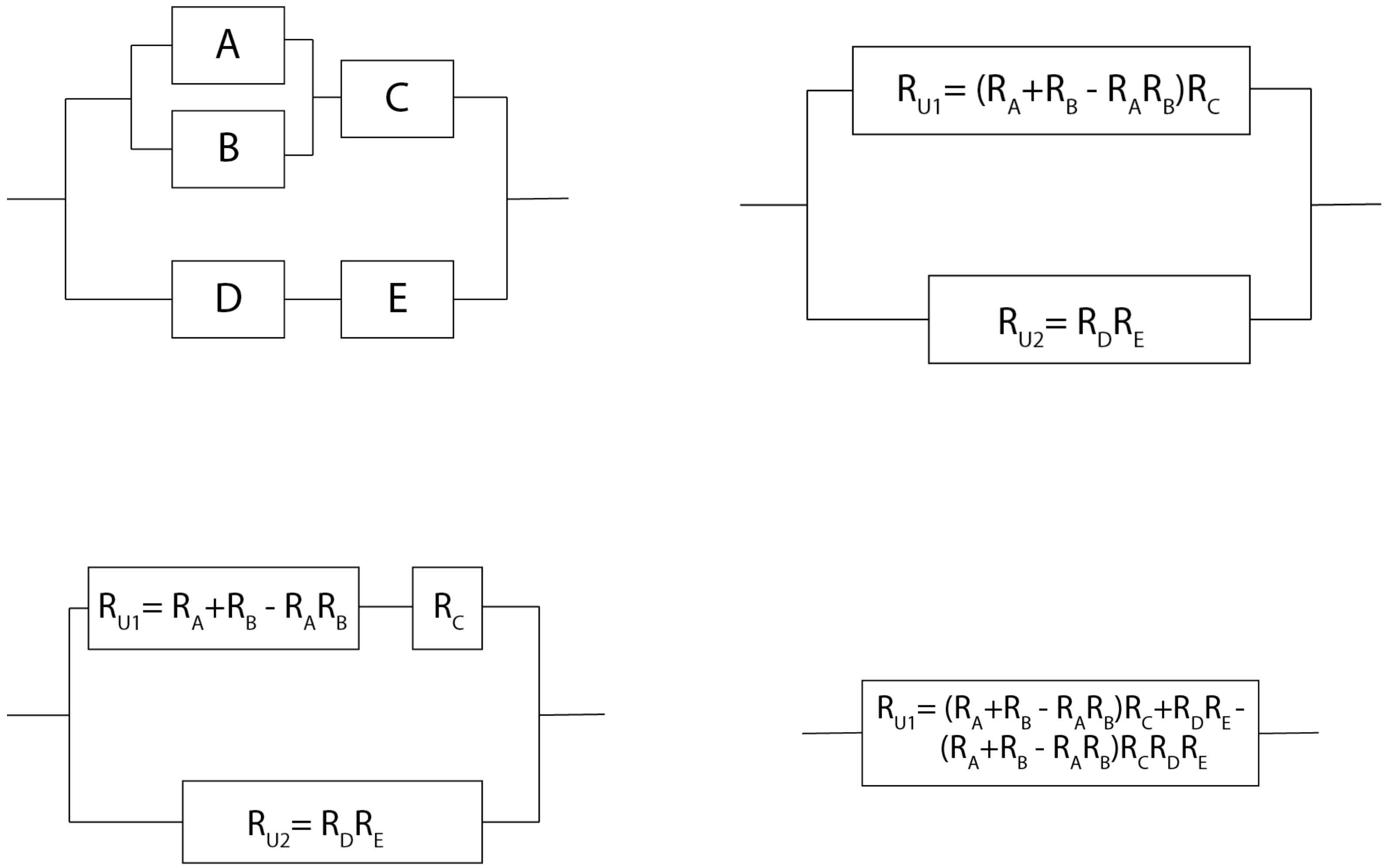
Example 3







Example 3


A
B
C
D
E
$R_{U1} = (R_A + R_B - R_A R_B) R_C$
$R_{U2} = R_D R_E$
$R_{U1} = R_A + R_B - R_A R_B$
R_C
$R_{U2} = R_D R_E$
$R_{U1} = (R_A + R_B - R_A R_B) R_C + R_D R_E - (R_A + R_B - R_A R_B) R_C R_D R_E$

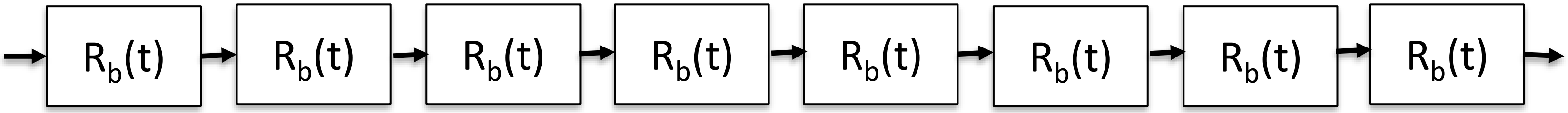
RBD: used to model a system and calculate its reliability

We have an 8-bit parallel bus within a System-on-Chip; each line of the bus may fail independently of the others; the reliability of each line of the bus is $R_b(t)$.

How would you model the entire bus using a RBD?

RBD: used to model a system and calculate its reliability

We have an 8-bit parallel bus within a System-on-Chip; each line of the bus may fail independently of the others; the reliability of each line of the bus is $R_b(t)$.

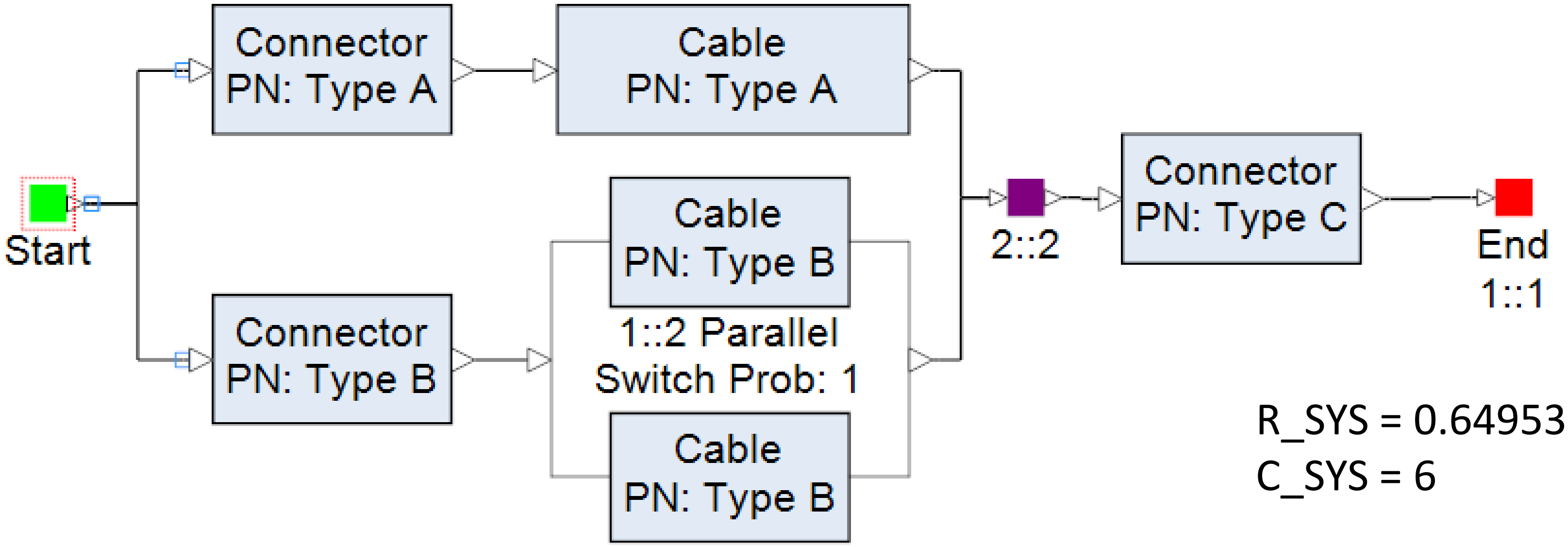


RBD: used to compare different alternatives

Cable Bundle

Each block has R = 0.9

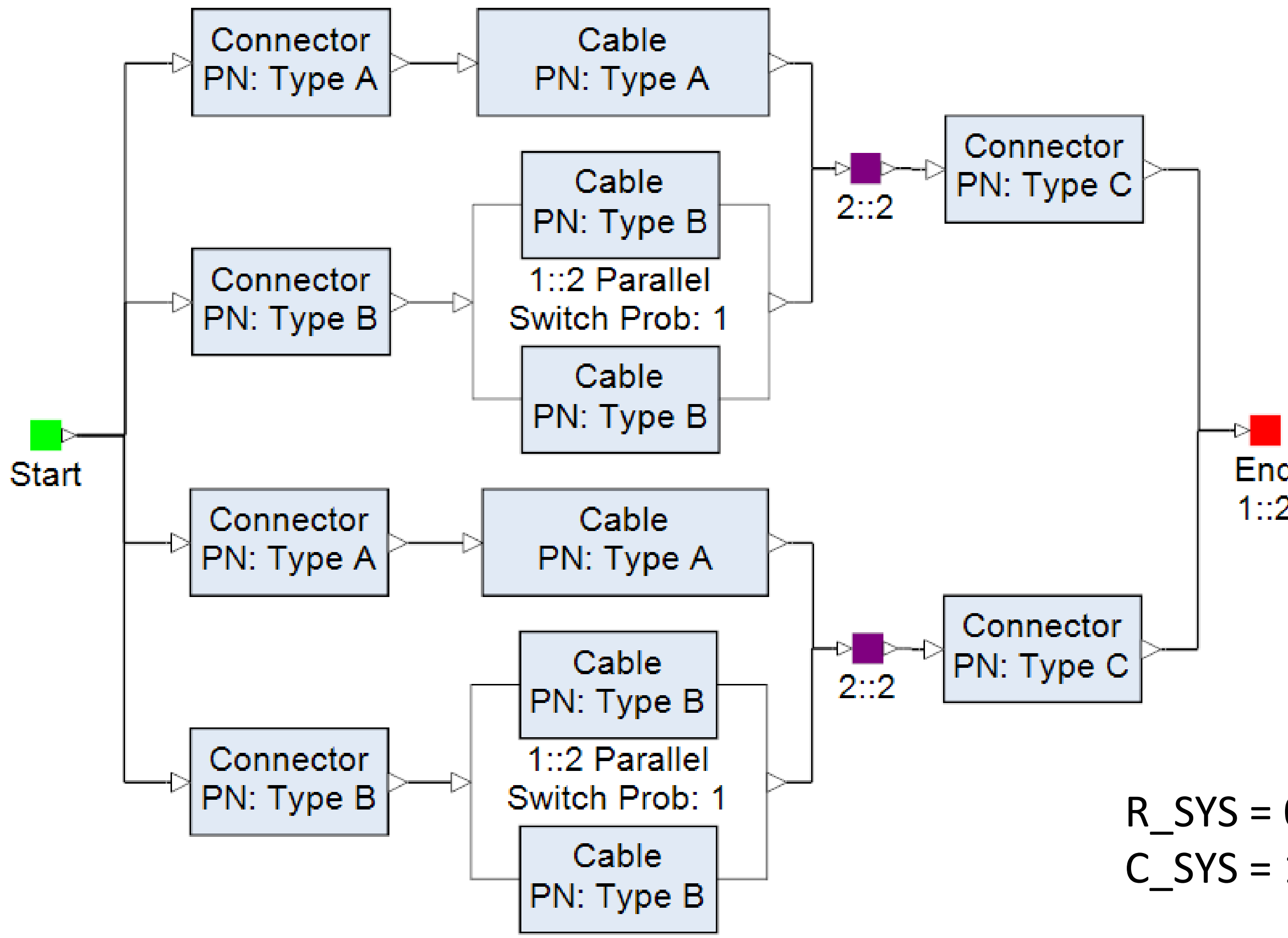
Each block costs 1



R_SYS = 0.649539

C_SYS = 6

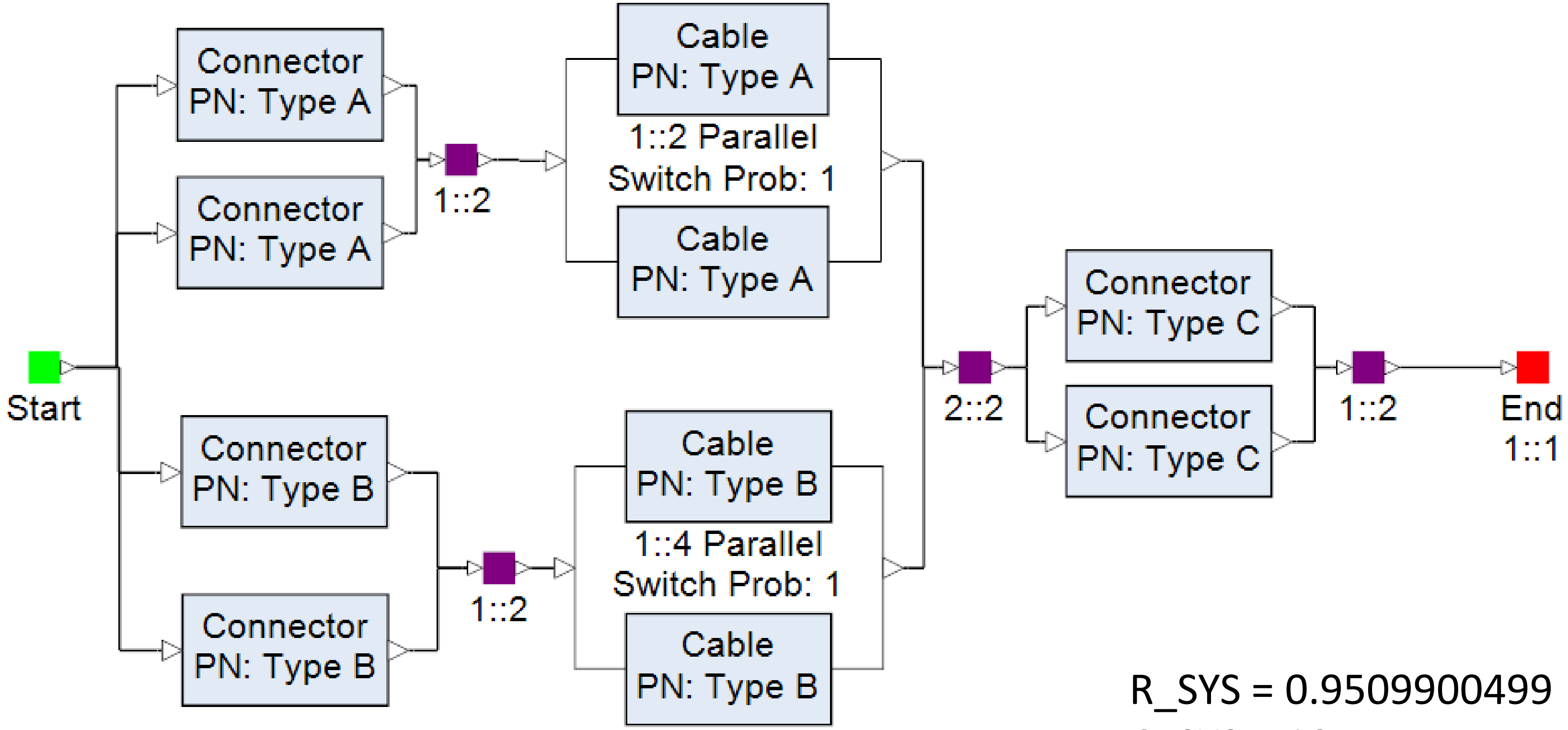
Alternative 1



R_SYS = 0.877177

C_SYS = 12

Alternative 2



R_SYS = 0.9509900499

C_SYS = 10

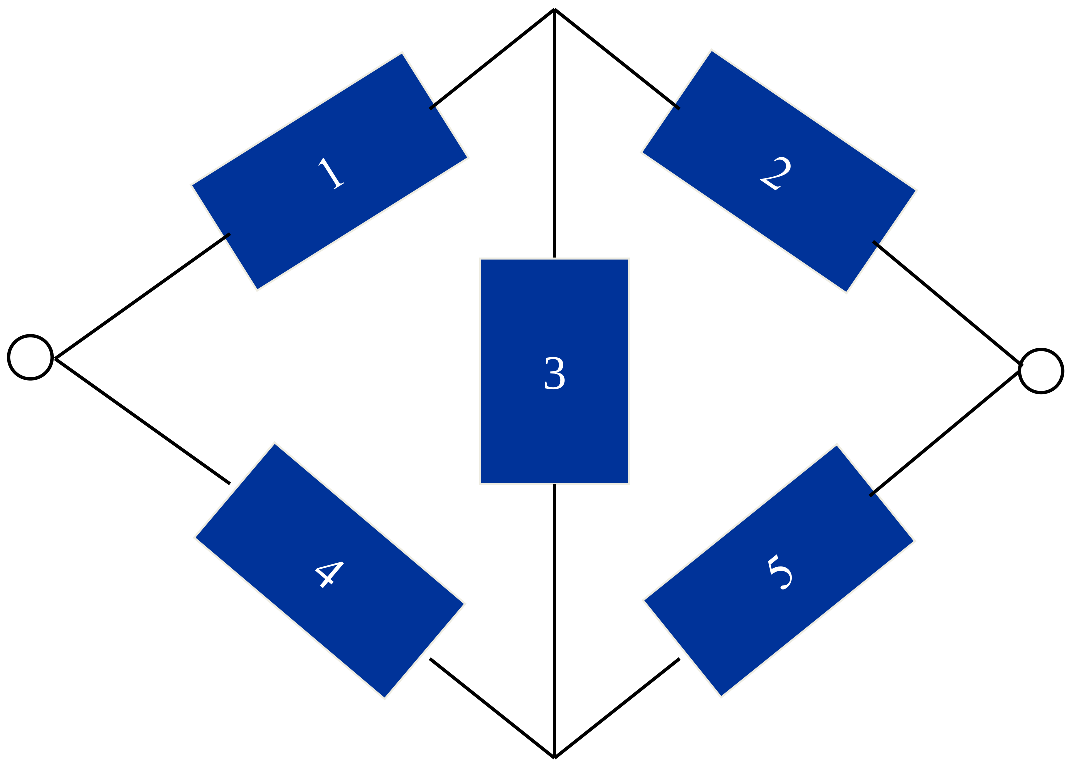
Methods for non-series-parallel systems

- State enumeration (Boolean Truth Table)

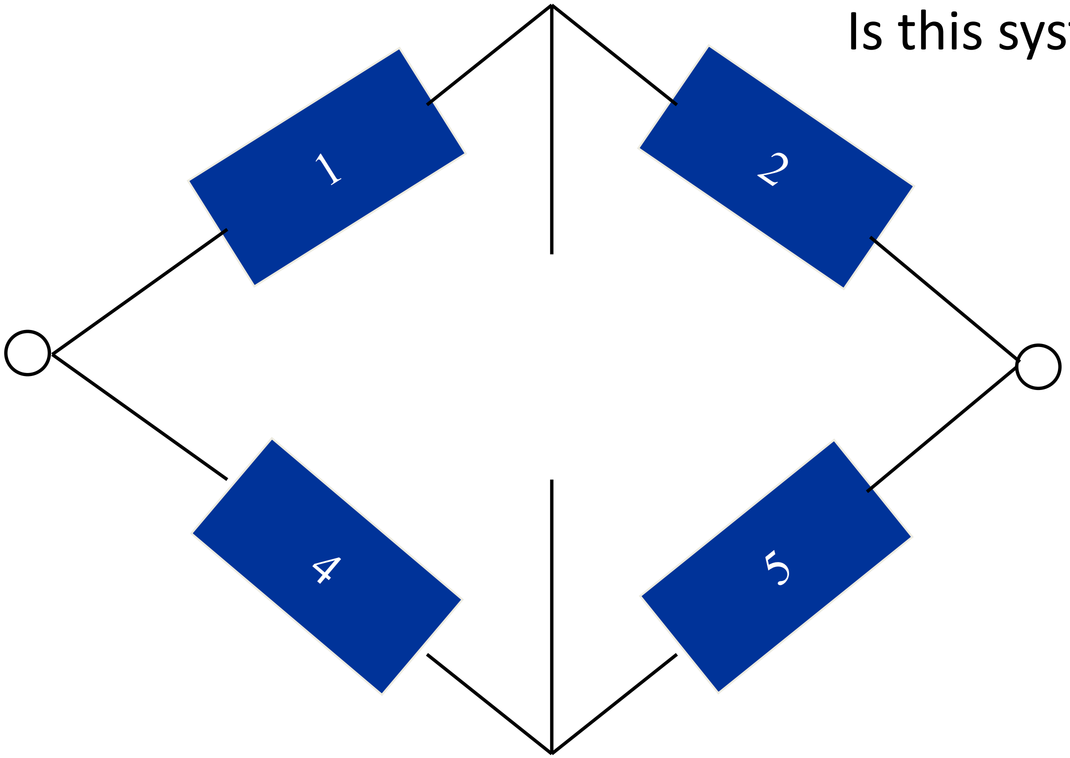
- Factoring/conditioning
- Binary Decision Diagrams

Implemented in SHARPE

non-series-parallel systems – State enumeration



non-series-parallel systems – State enumeration

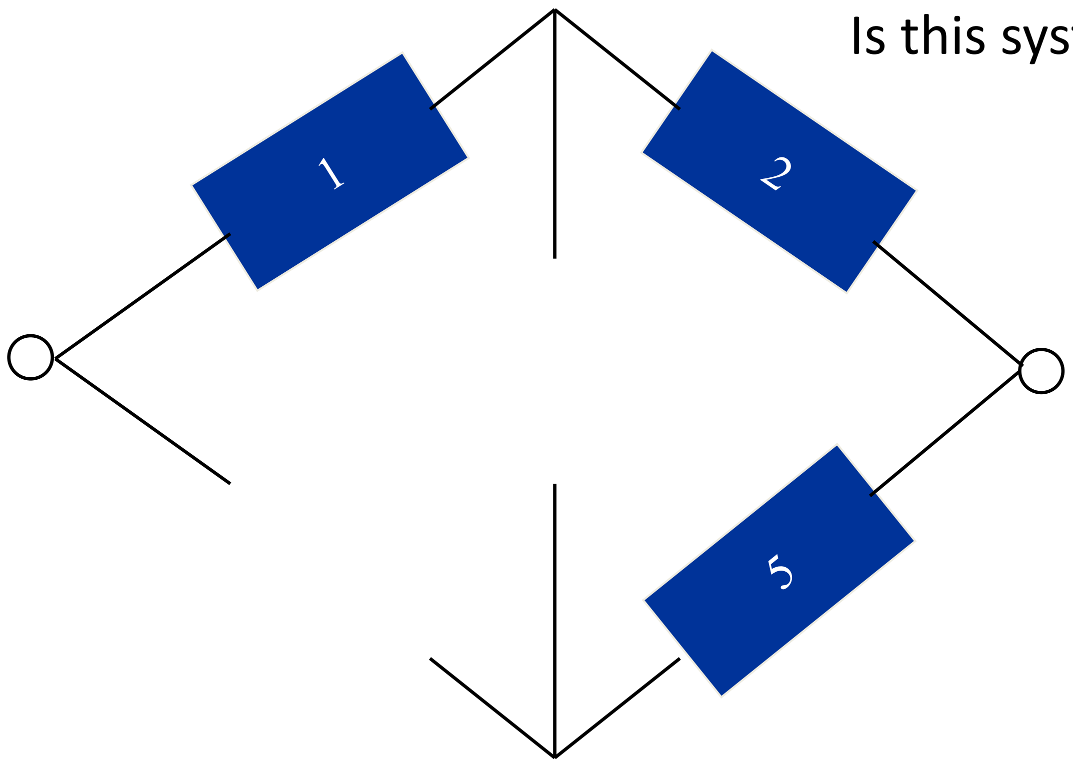


Is this system working?

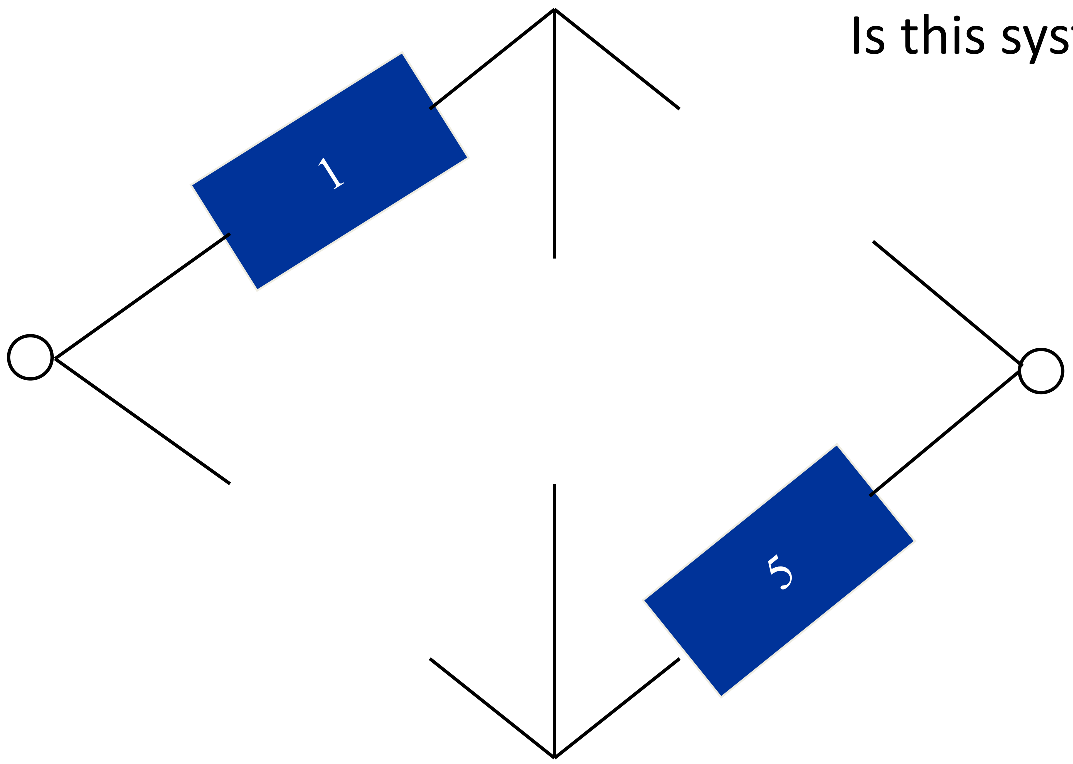
non-series-parallel systems – State enumeration

RBDs

Is this system working?



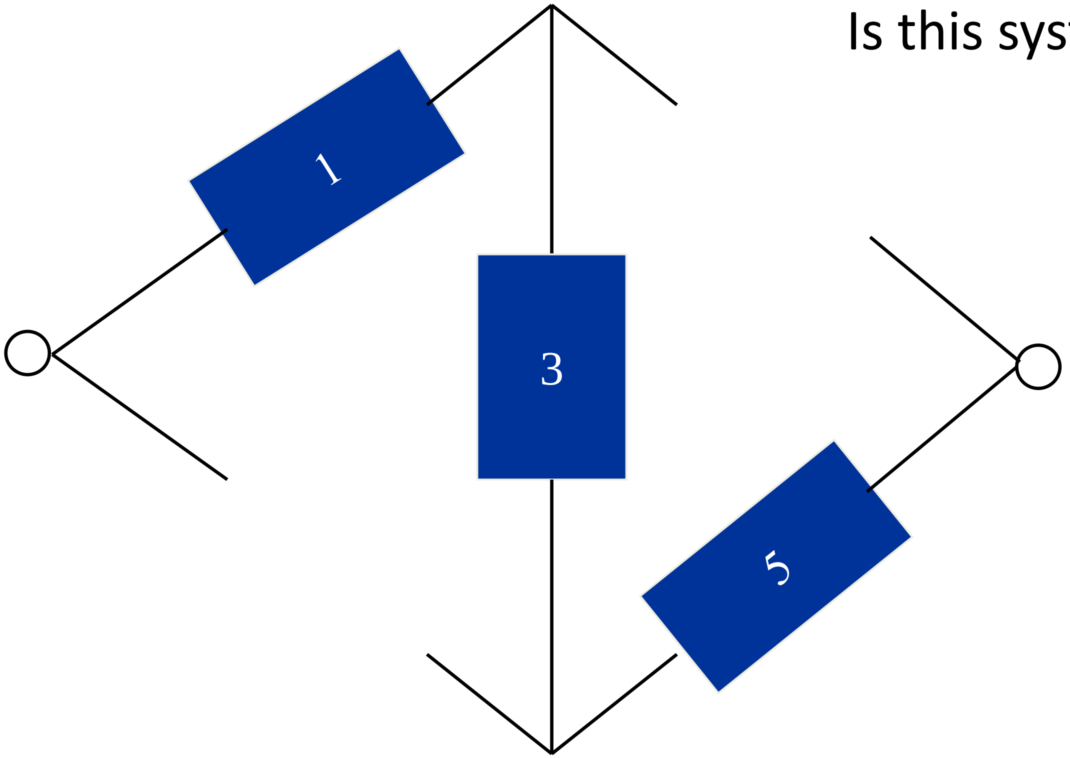
non-series-parallel systems – State enumeration



Is this system working?

non-series-parallel systems – State enumeration

Is this system working?



non-series-parallel systems – SE – Example

RBDs

1	2	3	4	5	System	Probability
1	1	1	1	1	1	$R_1R_2R_3R_4R_5$
1	1	1	1	0	1	$R_1R_2R_3R_4!R_5$
1	1	1	0	1	1	$R_1R_2R_3!R_4R_5$
1	1	1	0	0	1	$R_1R_2R_3!R_4!R_5$
1	1	0	1	1	1	$R_1R_2!R_3R_4R_5$
1	1	0	1	0	1	$R_1R_2!R_3R_4!R_5$
1	1	0	0	1	1	$R_1R_2!R_3!R_4R_5$
1	1	0	0	0	1	$R_1R_2!R_3!R_4!R_5$
1	0	1	1	1	1	
1	0	1	1	0	0	
1	0	1	0	1	1	
1	0	1	0	0	0	
1	0	0	1	1	1	
1	0	0	1	0	0	
1	0	0	0	1	0	
1	0	0	0	0	0	

non-series-parallel systems – SE – Example

RBDs

1	2	3	4	5	System	Probability
1	1	1	1	1	1	$R_1R_2R_3R_4R_5$
1	1	1	1	0	1	$R_1R_2R_3R_4!R_5$
1	1	1	0	1	1	$R_1R_2R_3!R_4R_5$
1	1	1	0	0	1	$R_1R_2R_3!R_4!R_5$
1	1	0	1	1	1	$R_1R_2!R_3R_4R_5$
1	1	0	1	0	1	$R_1R_2!R_3R_4!R_5$
1	1	0	0	1	1	$R_1R_2!R_3!R_4R_5$
1	1	0	0	0	1	$R_1R_2!R_3!R_4!R_5$
1	0	1	1	1	1	
1	0	1	1	0	0	
1	0	1	0	1	1	
1	0	1	0	0	0	
1	0	0	1	1	1	
1	0	0	1	0	0	
1	0	0	0	1	0	
1	0	0	0	0	0	

} R_1R_2

non-series-parallel systems – SE – Example

RBDs

1	2	3	4	5	System	Probability
1	1	1	1	1	1	$R_1R_2R_3R_4R_5$
1	1	1	1	0	1	$R_1R_2R_3R_4!R_5$
1	1	1	0	1	1	$R_1R_2R_3!R_4R_5$
1	1	1	0	0	1	$R_1R_2R_3!R_4!R_5$
1	1	0	1	1	1	$R_1R_2!R_3R_4R_5$
1	1	0	1	0	1	$R_1R_2!R_3R_4!R_5$
1	1	0	0	1	1	$R_1R_2!R_3!R_4R_5$
1	1	0	0	0	1	$R_1R_2!R_3!R_4!R_5$
1	0	1	1	1	1	$R_1!R_2R_3R_4R_5$
1	0	1	1	0	0	
1	0	1	0	1	1	$R_1!R_2R_3!R_4R_5$
1	0	1	0	0	0	
1	0	0	1	1	1	$R_1!R_2!R_3R_4R_5$
1	0	0	1	0	0	
1	0	0	0	1	0	
1	0	0	0	0	0	

R_1R_2 (sum of the first eight rows)

non-series-parallel systems – SE – Example – cont'd

RBDS

1	2	3	4	5	System	Probability
0	1	1	1	1	1	$!R_1R_2R_3R_4R_5$
0	1	1	1	0	1	$!R_1R_2R_3R_4!R_5$
0	1	1	0	1	0	
0	1	1	0	0	0	
0	1	0	1	1	1	$!R_1R_2!R_3R_4R_5$
0	1	0	1	0	0	
0	1	0	0	1	0	
0	1	0	0	0	0	
0	0	1	1	1	1	$!R_1!R_2R_3R_4R_5$
0	0	1	1	0	0	
0	0	1	0	1	0	
0	0	1	0	0	0	
0	0	0	1	1	1	$!R_1!R_2!R_3R_4R_5$
0	0	0	1	0	0	
0	0	0	0	1	0	
0	0	0	0	0	0	

(first two rows combined:) $!R_1R_2R_3R_4$

non-series-parallel systems – SE – Example – cont'd

Reliability: $R_1R_2 + !R_1R_2R_3R_4 + !R_1R_2!R_3R_4R_5 + !R_1!R_2R_3R_4R_5 + !R_1!R_2!R_3R_4R_5 + !R_1R_2!R_3R_4R_5 + !R_1!R_2R_3R_4R_5 + !R_1!R_2!R_3R_4R_5$

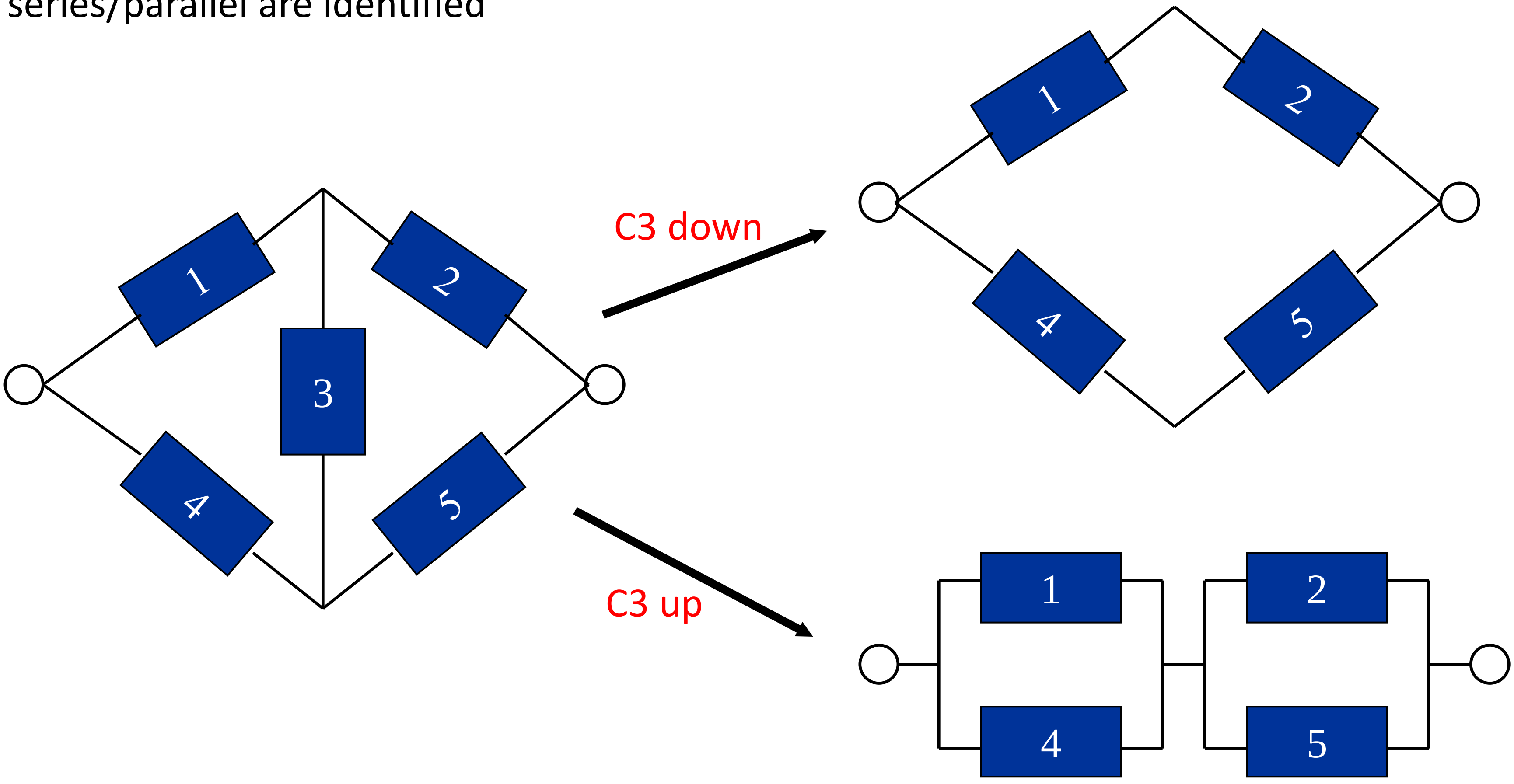
… simplifying and optimizing …

Reliability: $R_1R_2 + R_4R_5 + R_1R_3R_5 + R_2R_3R_4$

BTW, remember that: $!R = (1 – R)$

non-series-parallel systems – conditioning – Example

Components that prevent the system from being purely series/parallel are identified

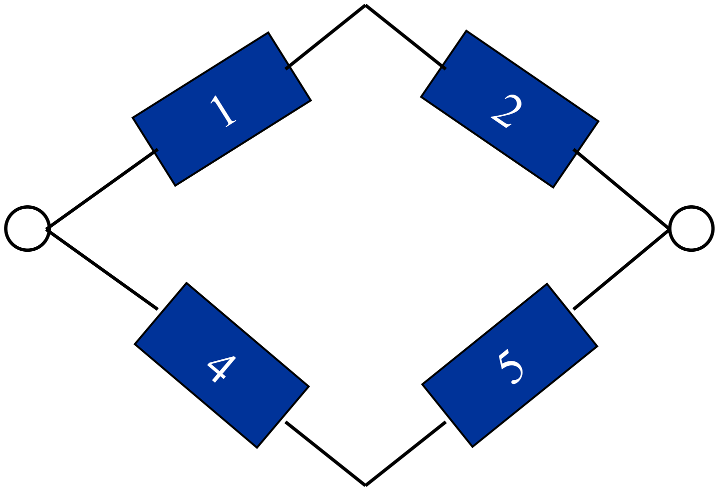


non-series-parallel systems – conditioning – Example – cont'd

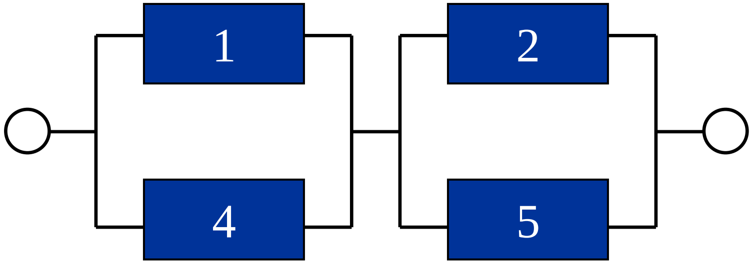
- Component C3 is chosen to factor on (or condition on)
- Upper resulting block diagram: C3 is down
- Lower resulting block diagram: C3 is up
- Series-parallel reliability formulas are applied to both the resulting block diagrams
- Use the theorem of total probability to get the final result

non-series-parallel systems – conditioning – Example – cont'd

$R_{C3down} = 1 - (1 - R_1 R_2)(1 - R_4 R_5)$

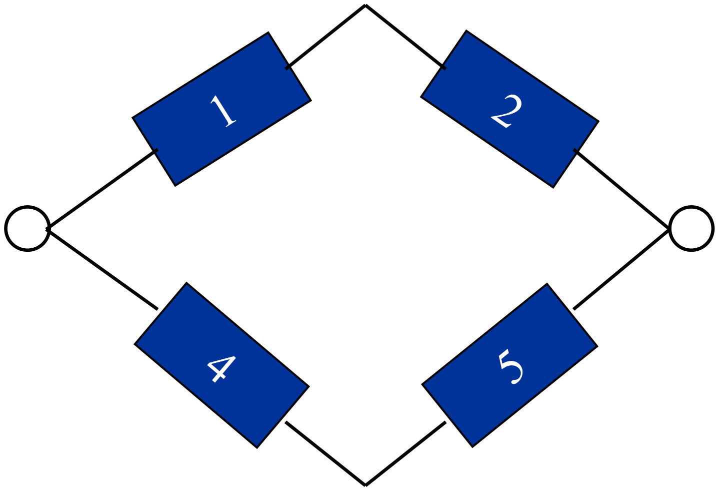


$R_{C3up} = [1 - (1-R_1)(1-R_4)]\,[1 - (1-R_2)(1-R_5)]$

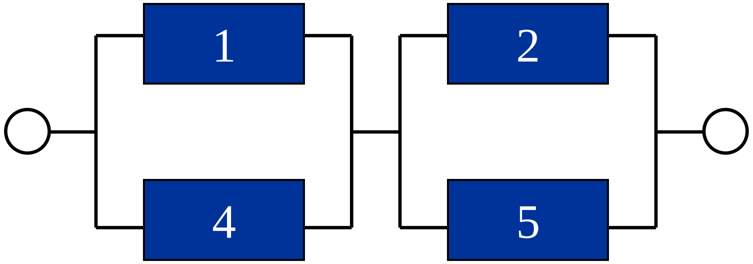


non-series-parallel systems – conditioning – Example – cont'd

$$R_{C3down} = 1 - (1 - R_1 R_2)(1 - R_4 R_5)$$



$$R_{C3up} = [1 - (1-R_1)(1-R_4)]\,[1 - (1-R_2)(1-R_5)]$$



$$R_{sys} = R_{C3down}(1-R_3) + R_{C3up} R_3$$

Pros and cons

Advantages

- An RBD allows an early assessment of the reliability of a design and allows to easily visualize the system logic

- Easy to calculate by mathematical solving

Pros and cons

Limitations

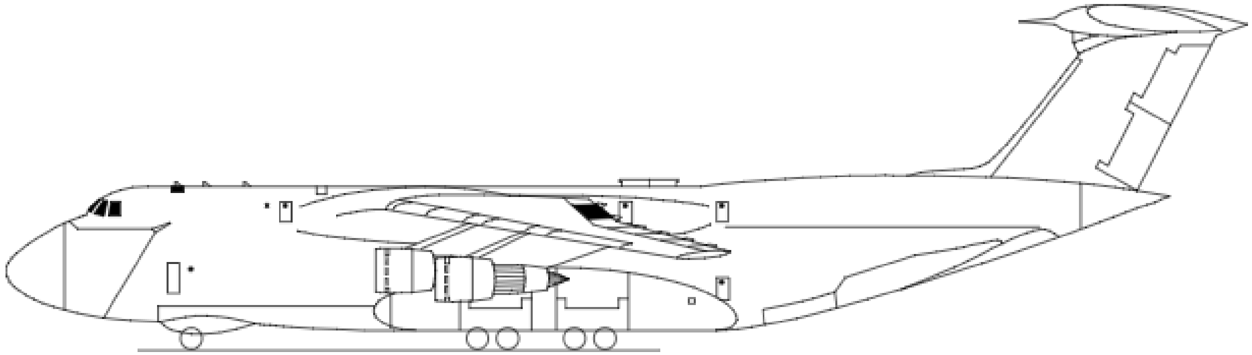
- breaking down the systems to identify multiple levels of components may require a considerable effort

- analyzing complex reliability diagrams can be difficult…not simple series / parallel configurations

- modeling non-hardware failure mitigation measures, such as training and procedures, is difficult using this technique

Exercise (at home)

- Out of the 12 identical AC generators on the C-5 aircraft, at least 9 of them must be operating in order for the aircraft to complete its mission. Failures are known to follow an exponential distribution with a failure rate of 0.01 failure per hour. What is the reliability of the generator system over a 10 hour mission in case the switch is perfect?



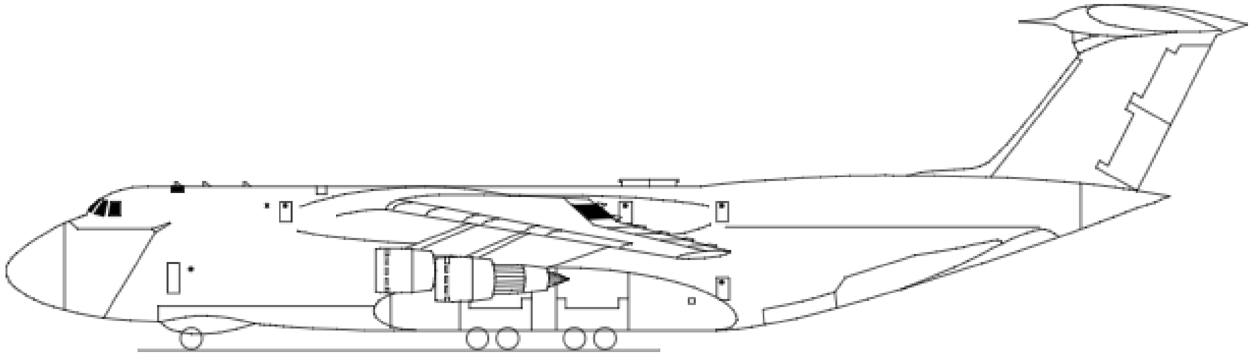
Exercise (at home)

$R_m(t) = e^{-\lambda t}$ with λ=0.01 and t=10

$R_m(t) = e^{-(0.1)}$=0.9048374

$$R_S(t) = \sum_{i=9}^{12} R_m^i (1 - R_m)^{12-i} \frac{12!}{i!\,(12-i)!}$$

$= 165\ R_m^{12} + 540\ R_m^{11} + 594 R_m^{10} + 220\ R_m^{9}$=0.9782773



Fault Tree Analysis (FTA)

Fault Tree Analysis

A deductive, backward and top-down failure analysis

Fault Tree Analysis

A deductive, backward and top-down failure analysis

Defines a correlation between possible events and failures of the system

- Events are composed by means of “logic gates”

Fault Tree Analysis

A deductive, backward and top-down failure analysis

Defines a correlation between possible events and failures of the system

– Events are composed by means of "logic gates"

The approach offers a tree model of the events and conditions that lead to a failure

Fault Tree Analysis

A deductive, backward and top-down failure analysis

Defines a correlation between possible events and failures of the system

- Events are composed by means of “logic gates”

The approach offers a tree model of the events and conditions that lead to a failure

It can be used to characterize a system and to evaluate the overall dependability properties

Fault Tree Analysis

Approach

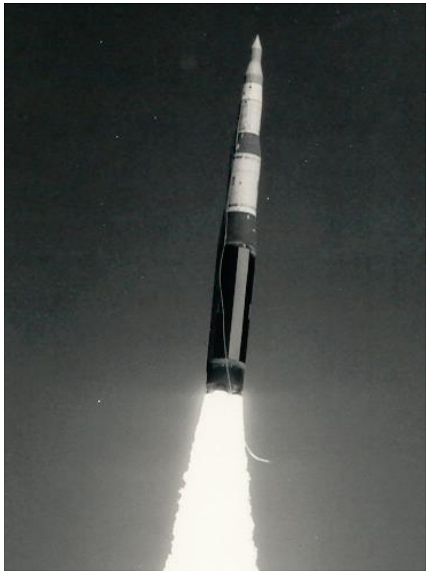
Starting with a potential undesirable event (accident) called a **TOP event**, a FTA determines all the ways it can happen

– What (combination of) events can lead to the TOP event

A FTA starts with the undesired event and traces backward to the **necessary** and **sufficient** causes (**BASIC events**)

Historical perspective

Bell Telephone Laboratories developed the concept in 1962 for the US Air Force for use with the Minuteman system



Later improved by Boeing Company

One of many symbolic "analytical logic techniques" found in operations research and in system reliability

Fault Trees

- Combinatorial (non-state-space) model type
- Events are represented as nodes
- Correlation between events are represented as logic gates

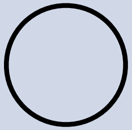
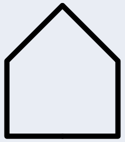
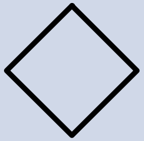
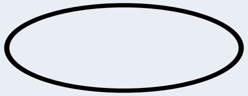
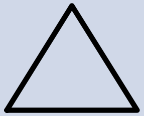
Fault Trees

- Combinatorial (non-state-space) model type
- Events are represented as nodes
- Correlation between events are represented as logic gates

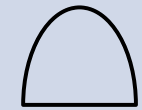
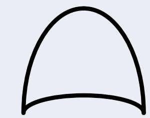
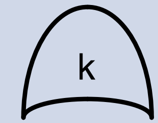
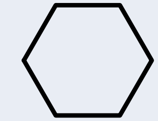
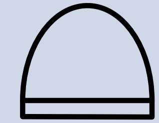
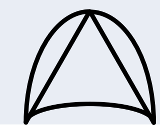
In particular:

- Components or subsystems in series are connected to OR gates
- Components or subsystems in parallel are connected to AND gates
- Components or subsystems in k-of-n (RBD) are connected as (n-k+1)-of-n gate

Events

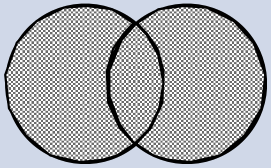
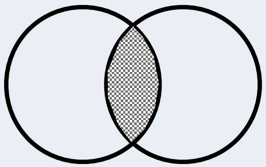
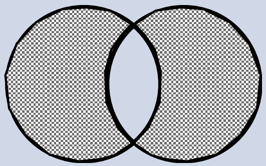
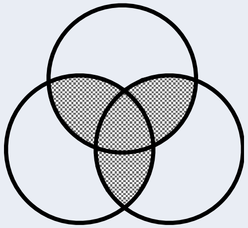
Events	Meaning	Symbol
Basic Event	A basic initiating fault (or failure event)	
External Event (House Event)	An event that is normally expected to occur. In general, these events can be set to occur or not occur, *i.e.* they have a fixed probability of 0 or 1.	
Undeveloped Event	An event for which not enough information is available or that is of no consequence.	
Conditioning Event	A specific condition or restriction that can apply to any gate.	
Transfer	Indicates a transfer continuation to a sub tree. Used to connect sub-trees .	

Gates

Gates	Meaning	Symbol
AND	The output event occurs if all input events occur	
OR	The output event occurs if at least one of the input events occurs	
Voting OR (k-out-of-n)	The output event occurs if k or more of the input events occur	
Inhibit	The input event occurs if all input events occur and an additional conditional event occurs	
Priority AND	The output event occurs if all input events occur in a specific sequence	
XOR	The output event occurs if exactly one input event occurs	

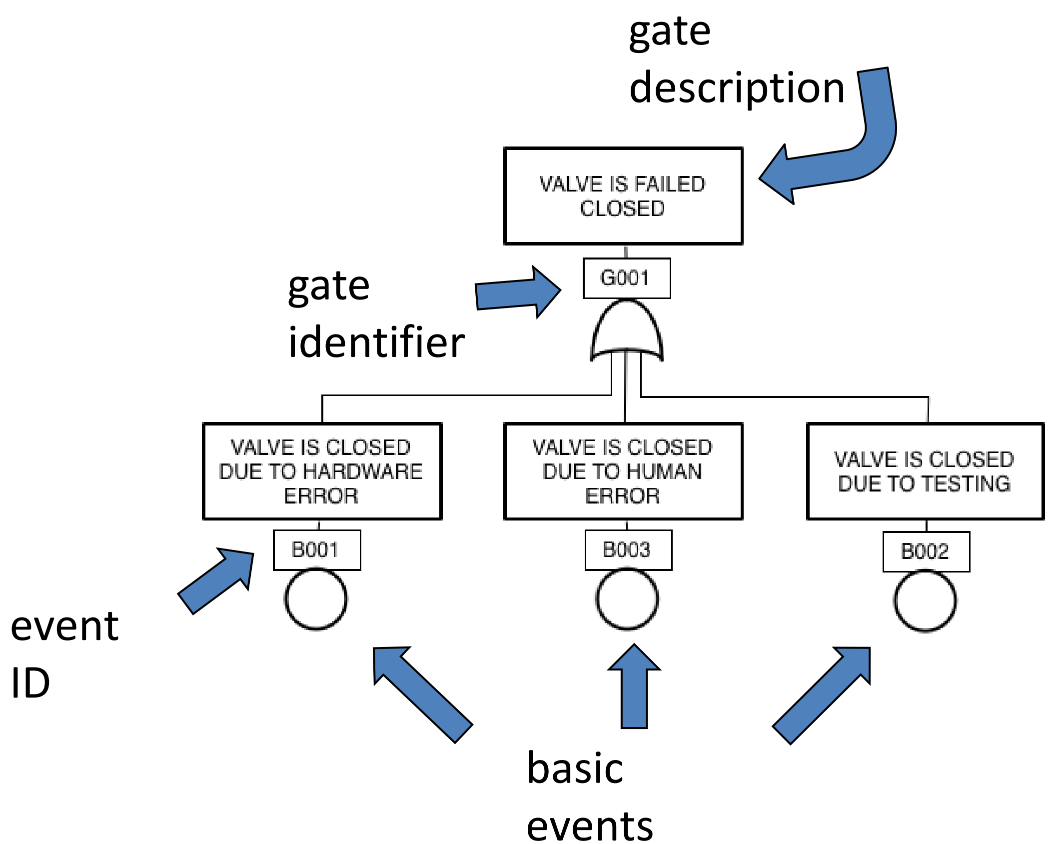
Failure probabilities

$$P_t = P_1(1-P_2) + (1-P_1)P_2 + P_1P_2$$
$$= P_1 - P_1P_2 + P_2 - P_1P_2 + P_1P_2$$
$$= P_1 + P_2 - P_1P_2$$

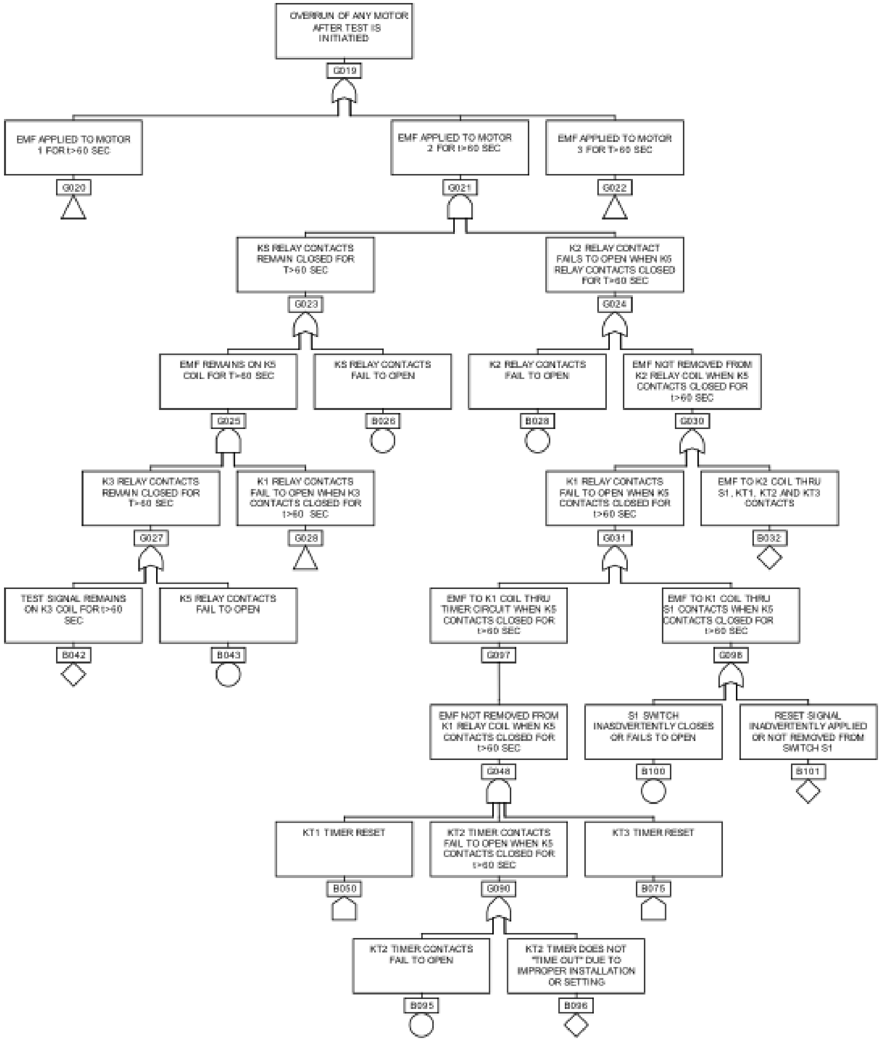
Gate	Failure probability	
OR	$P_t = P_1 + P_2 - (P_1P_2)$ P_t : total failure probability P_i: failure probability, event i	
AND Priority AND	$P_t = P_1P_2$	
XOR	$P_t = P_1 + P_2 - 2(P_1P_2)$	
Voting OR k-out-of-n	$P_t = P_1 + P_2 + P_3 - (P_1P_2) - (P_1P_3) - (P_2P_3) - (P_1P_2P_3)$ (2-out-of-3)	

FTA model

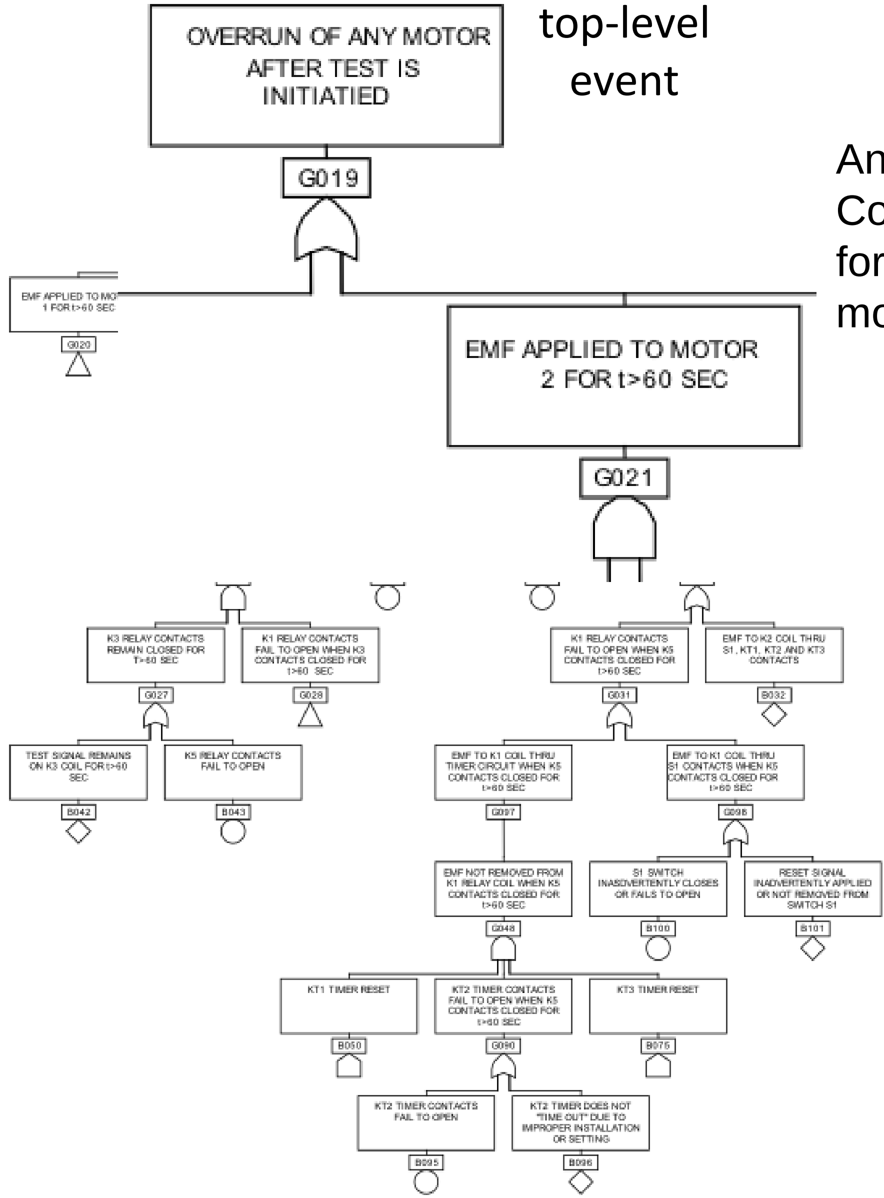
Gate & Events symbols and descriptions



FTA Example

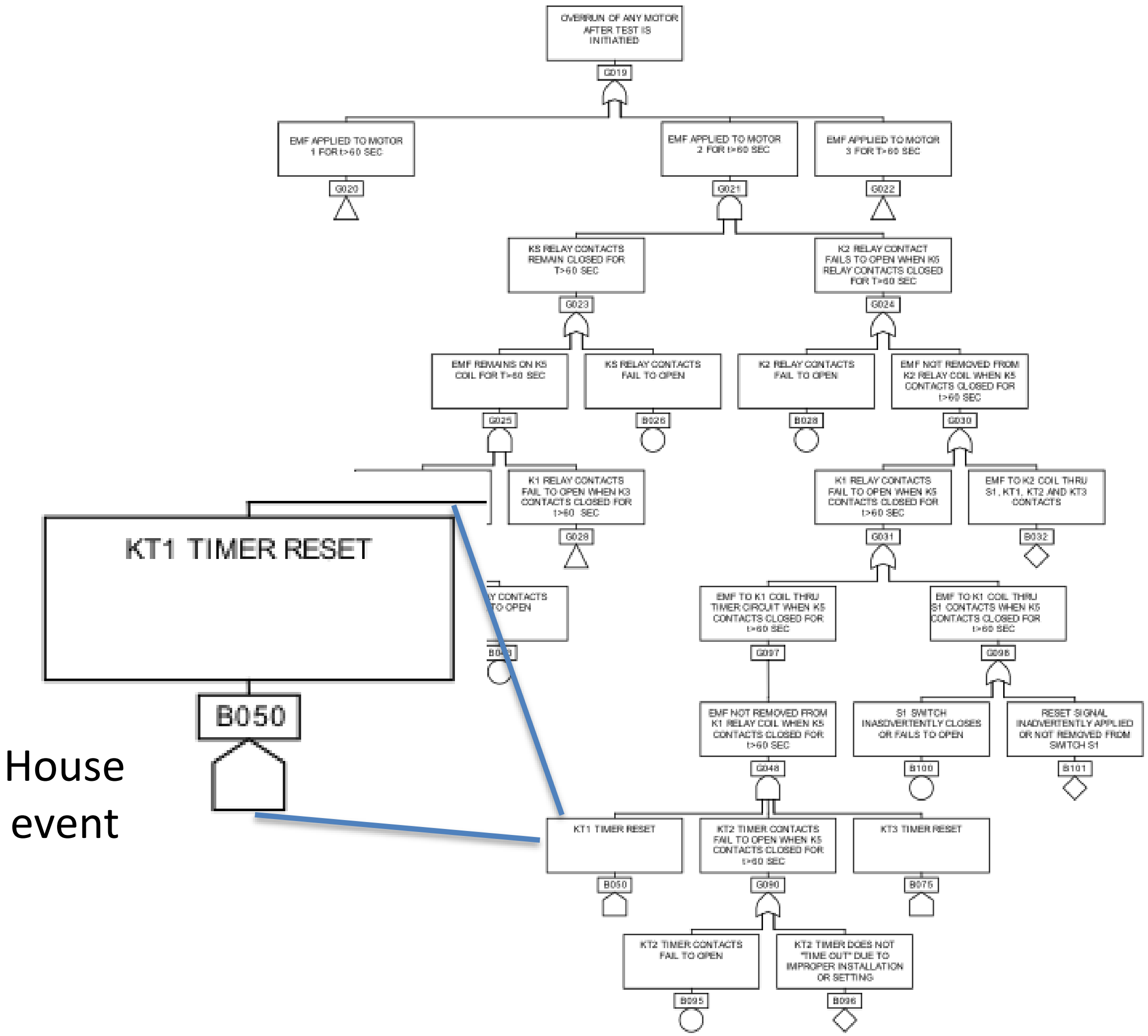

OVERRUN OF ANY MOTOR AFTER TEST IS INITIATED
G019
EMF APPLIED TO MOTOR 1 FOR t>60 SEC
G020
EMF APPLIED TO MOTOR 2 FOR t>60 SEC
G021
EMF APPLIED TO MOTOR 3 FOR T>60 SEC
G022
K5 RELAY CONTACTS REMAIN CLOSED FOR T>60 SEC
G023
K2 RELAY CONTACT FAILS TO OPEN WHEN K5 RELAY CONTACTS CLOSED FOR T>60 SEC
G024
EMF REMAINS ON K5 COIL FOR T>60 SEC
G025
K5 RELAY CONTACTS FAIL TO OPEN
B026
K2 RELAY CONTACTS FAIL TO OPEN
B028
EMF NOT REMOVED FROM K2 RELAY COIL WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G030
K3 RELAY CONTACTS REMAIN CLOSED FOR T>60 SEC
G027
K1 RELAY CONTACTS FAIL TO OPEN WHEN K3 CONTACTS CLOSED FOR t>60 SEC
G028
K1 RELAY CONTACTS FAIL TO OPEN WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G031
EMF TO K2 COIL THRU S1, KT1, KT2 AND KT3 CONTACTS
B032
TEST SIGNAL REMAINS ON K3 COIL FOR t>60 SEC
B042
K5 RELAY CONTACTS FAIL TO OPEN
B043
EMF TO K1 COIL THRU TIMER CIRCUIT WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G097
EMF TO K1 COIL THRU S1 CONTACTS WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G098
EMF NOT REMOVED FROM K1 RELAY COIL WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G048
S1 SWITCH INASDVERTENTLY CLOSES OR FAILS TO OPEN
B100
RESET SIGNAL INADVERTENTLY APPLIED OR NOT REMOVED FROM SWITCH S1
B101
KT1 TIMER RESET
B050
KT2 TIMER CONTACTS FAIL TO OPEN WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G090
KT3 TIMER RESET
B075
KT2 TIMER CONTACTS FAIL TO OPEN
B095
KT2 TIMER DOES NOT "TIME OUT" DUE TO IMPROPER INSTALLATION OR SETTING
B096

FTA Example

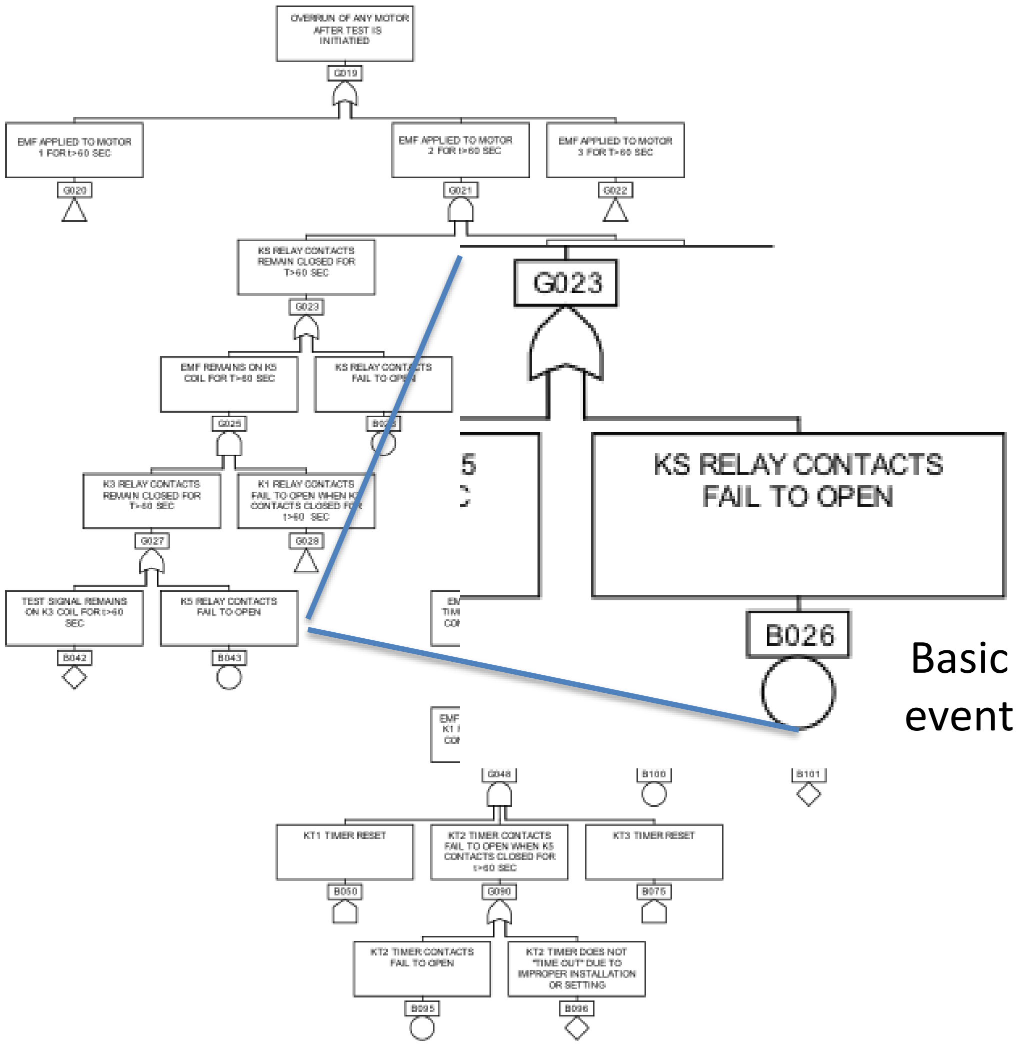


Any motor overruns if Counter-electromotive force is applied for more than 1 min

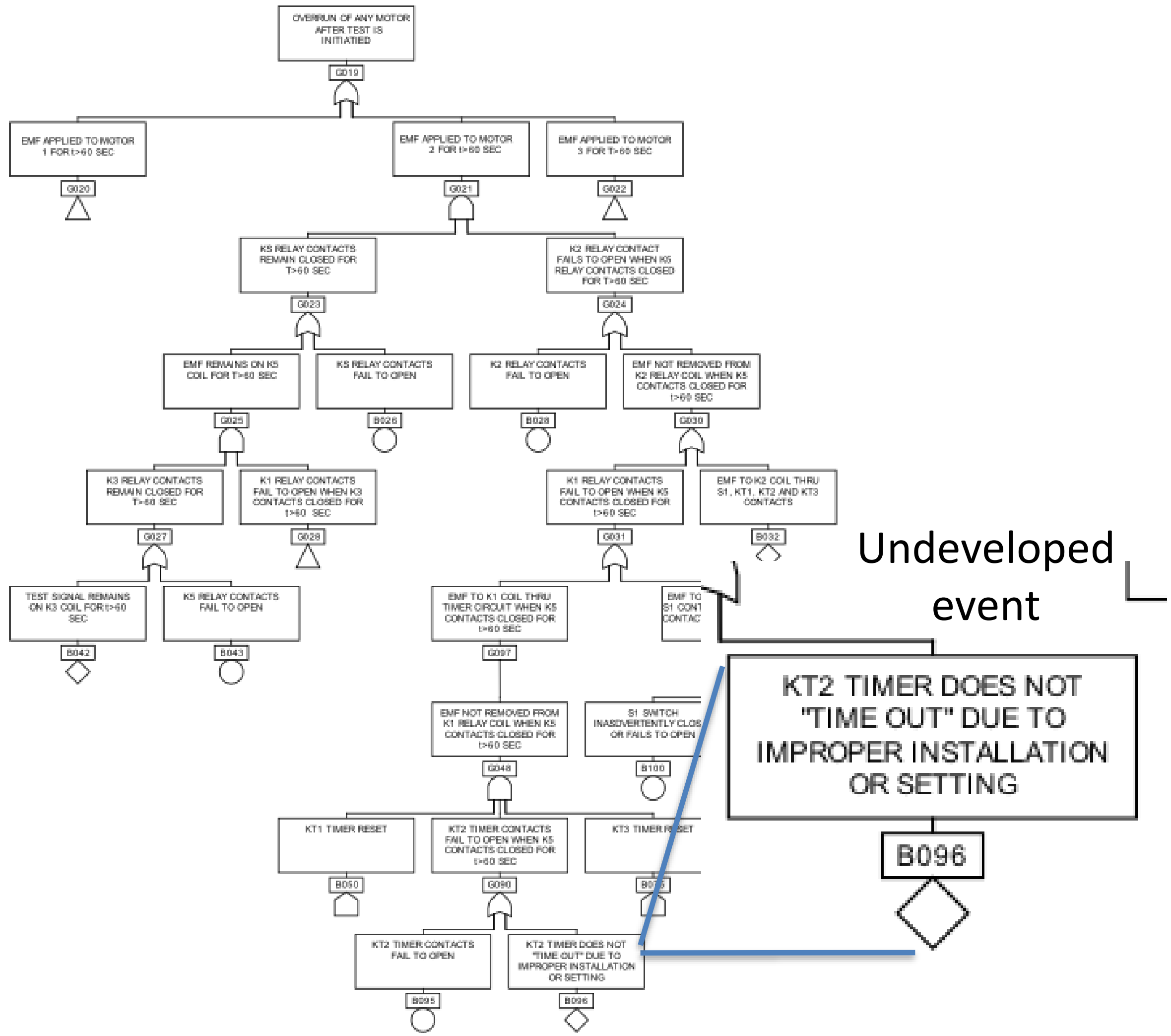
FTD Example


OVERRUN OF ANY MOTOR AFTER TEST IS INITIATIED
G019
EMF APPLIED TO MOTOR 1 FOR t>60 SEC
G020
EMF APPLIED TO MOTOR 2 FOR t>60 SEC
G021
EMF APPLIED TO MOTOR 3 FOR T>60 SEC
G022
K5 RELAY CONTACTS REMAIN CLOSED FOR T>60 SEC
G023
K2 RELAY CONTACT FAILS TO OPEN WHEN K5 RELAY CONTACTS CLOSED FOR T>60 SEC
G024
EMF REMAINS ON K5 COIL FOR T>60 SEC
G025
K5 RELAY CONTACTS FAIL TO OPEN
B026
K2 RELAY CONTACTS FAIL TO OPEN
B028
EMF NOT REMOVED FROM K2 RELAY COIL WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G030
K1 RELAY CONTACTS FAIL TO OPEN WHEN K3 CONTACTS CLOSED FOR t>60 SEC
G028
K1 RELAY CONTACTS FAIL TO OPEN WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G031
EMF TO K2 COIL THRU S1, KT1, KT2 AND KT3 CONTACTS
B032
KT1 TIMER RESET
B050
House event
EMF TO K1 COIL THRU TIMER CIRCUIT WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G097
EMF TO K1 COIL THRU S1 CONTACTS WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G098
EMF NOT REMOVED FROM K1 RELAY COIL WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G048
S1 SWITCH INADVERTENTLY CLOSES OR FAILS TO OPEN
B100
RESET SIGNAL INADVERTENTLY APPLIED OR NOT REMOVED FROM SWITCH S1
B101
KT1 TIMER RESET
B050
KT2 TIMER CONTACTS FAIL TO OPEN WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G090
KT3 TIMER RESET
B075
KT2 TIMER CONTACTS FAIL TO OPEN
B095
KT2 TIMER DOES NOT "TIME OUT" DUE TO IMPROPER INSTALLATION OR SETTING
B096

FTD Example


OVERRUN OF ANY MOTOR AFTER TEST IS INITIATED
G019
EMF APPLIED TO MOTOR 1 FOR t>60 SEC
G020
EMF APPLIED TO MOTOR 2 FOR t>60 SEC
G021
EMF APPLIED TO MOTOR 3 FOR T>60 SEC
G022
KS RELAY CONTACTS REMAIN CLOSED FOR T>60 SEC
G023
EMF REMAINS ON K5 COIL FOR T>60 SEC
G025
K3 RELAY CONTACTS REMAIN CLOSED FOR T>60 SEC
G027
TEST SIGNAL REMAINS ON K3 COIL FOR t>60 SEC
B042
K5 RELAY CONTACTS FAIL TO OPEN
B043
G028
G048
B100
B101
KT1 TIMER RESET
B050
KT2 TIMER CONTACTS FAIL TO OPEN WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G090
KT3 TIMER RESET
B075
KT2 TIMER CONTACTS FAIL TO OPEN
B095
KT2 TIMER DOES NOT "TIME OUT" DUE TO IMPROPER INSTALLATION OR SETTING
B096
G023
KS RELAY CONTACTS FAIL TO OPEN
B026
Basic event

FTD Example


OVERRUN OF ANY MOTOR AFTER TEST IS INITIATED
G019
EMF APPLIED TO MOTOR 1 FOR t>60 SEC
G020
EMF APPLIED TO MOTOR 2 FOR t>60 SEC
G021
EMF APPLIED TO MOTOR 3 FOR T>60 SEC
G022
K5 RELAY CONTACTS REMAIN CLOSED FOR T>60 SEC
G023
K2 RELAY CONTACT FAILS TO OPEN WHEN K5 RELAY CONTACTS CLOSED FOR T>60 SEC
G024
EMF REMAINS ON K5 COIL FOR T>60 SEC
G025
K5 RELAY CONTACTS FAIL TO OPEN
B026
K2 RELAY CONTACTS FAIL TO OPEN
B028
EMF NOT REMOVED FROM K2 RELAY COIL WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G030
K3 RELAY CONTACTS REMAIN CLOSED FOR T>60 SEC
G027
K1 RELAY CONTACTS FAIL TO OPEN WHEN K3 CONTACTS CLOSED FOR t>60 SEC
G028
K1 RELAY CONTACTS FAIL TO OPEN WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G031
EMF TO K2 COIL THRU S1, KT1, KT2 AND KT3 CONTACTS
B032
TEST SIGNAL REMAINS ON K3 COIL FOR t>60 SEC
B042
K5 RELAY CONTACTS FAIL TO OPEN
B043
EMF TO K1 COIL THRU TIMER CIRCUIT WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G097
EMF NOT REMOVED FROM K1 RELAY COIL WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G048
S1 SWITCH INADVERTENTLY CLOS OR FAILS TO OPEN
B100
KT1 TIMER RESET
B050
KT2 TIMER CONTACTS FAIL TO OPEN WHEN K5 CONTACTS CLOSED FOR t>60 SEC
G090
KT3 TIMER RESET
KT2 TIMER CONTACTS FAIL TO OPEN
B095
KT2 TIMER DOES NOT "TIME OUT" DUE TO IMPROPER INSTALLATION OR SETTING
B096
Undeveloped event
KT2 TIMER DOES NOT "TIME OUT" DUE TO IMPROPER INSTALLATION OR SETTING
B096

Fault tree construction

Define a TOP event in a clear and unambiguous way

- What
- Where
- When

What are the necessary, and sufficient events and conditions causing the TOP event?

Fault tree construction

Define a TOP event in a clear and unambiguous way

- What
- Where
- When

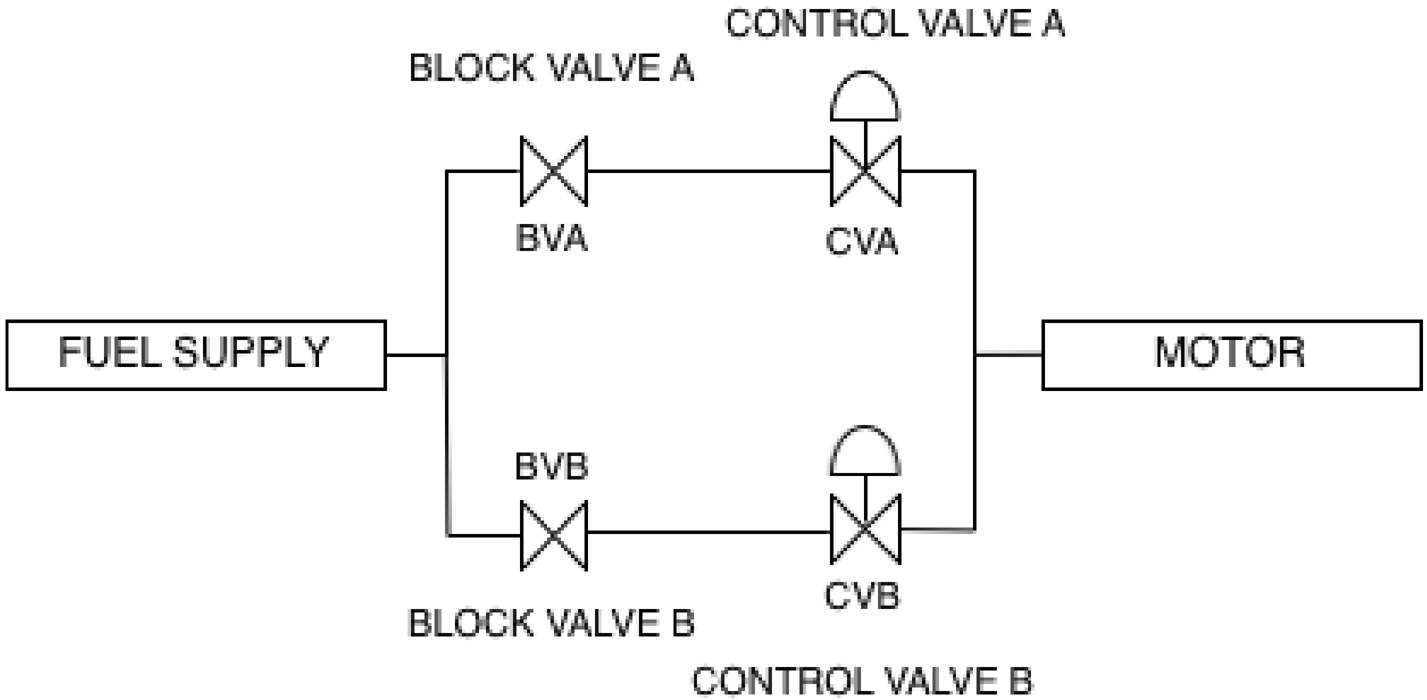
What are the necessary, and sufficient events and conditions causing the TOP event?

What has to be considered as a basic event?

- Independent events
- Events for which we have failure data

Example

Fuel system schematic



Fuel system faults

Example

Failures States

- No fuel flow when needed
- Fuel flow cannot be shut off not needed any more

Fuel system faults

Example

Failures States

- No fuel flow when needed
- Fuel flow cannot be shut off not needed any more

Component failures

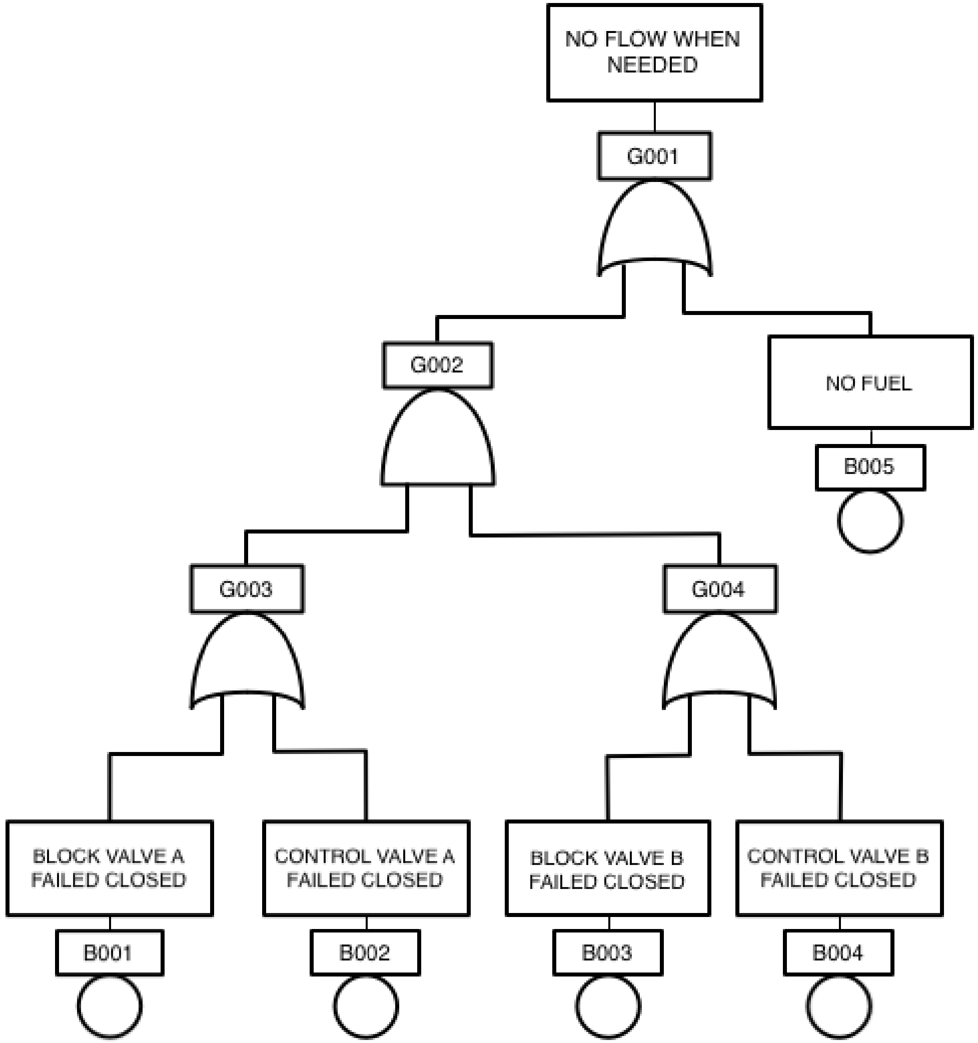
- Block Valve A fail open/fail close
- Block Valve B fail open/fail close
- Control Valve A fail open/fail close
- Control Valve B fail open/fail close

External failure

- No fuel available

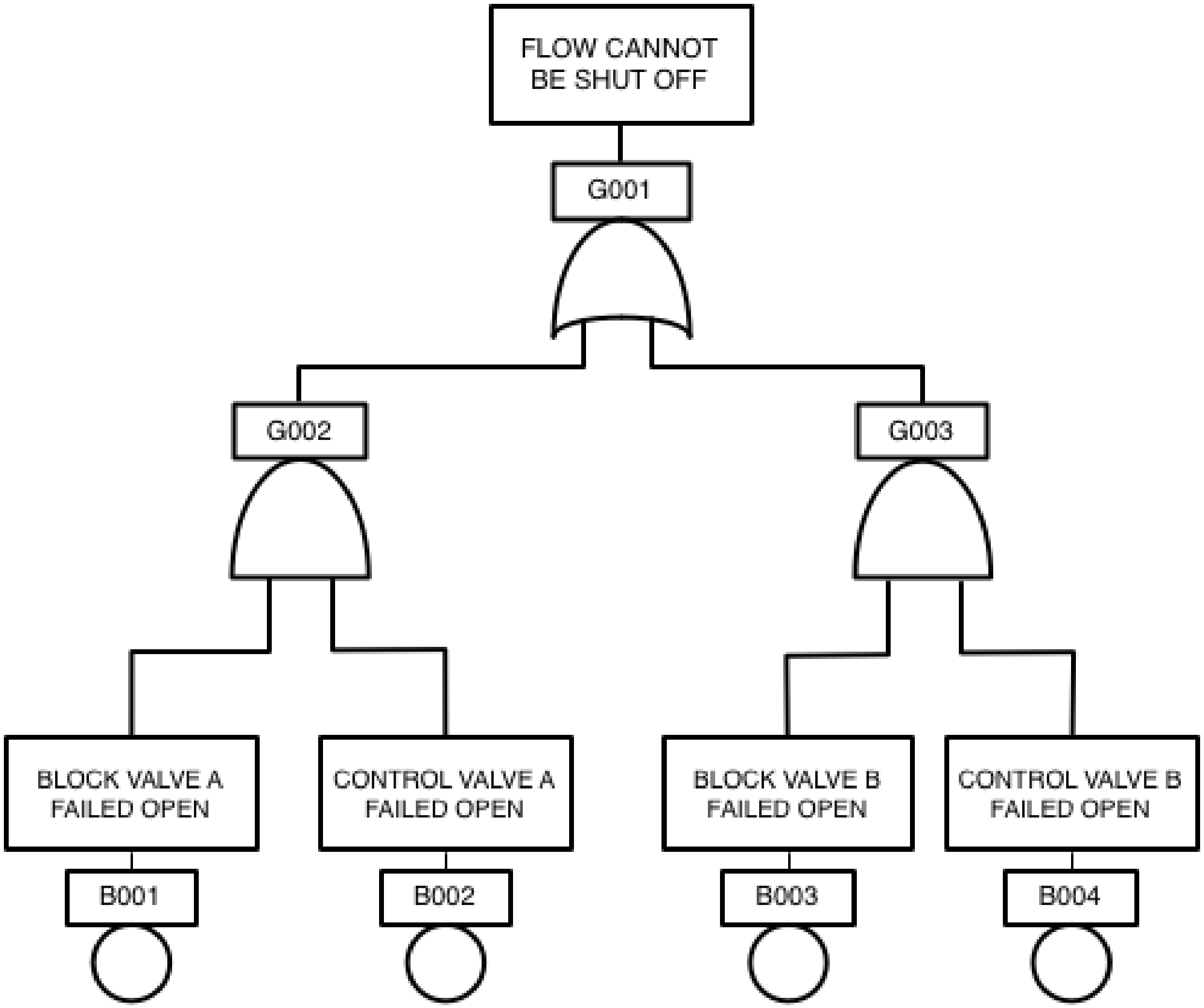
No fuel flow when needed

Example



Fuel flow cannot be shut off not needed any more

Example



Fault tree

Major characteristics:

- FT without repeated events (same event in input at different gates)
    - can be mapped onto RBDs
    - can be solved in linear time

Fault tree

Major characteristics:

- FT without repeated events (same event in input at different gates)
  - can be mapped onto RBDs
  - can be solved in linear time

- FT with repeated events
  - Theoretical complexity: exponential in the number of events

- Up to 100 components can still be solved …

How to exploit Fault Trees?

Analyze the causes leading to top events and identify the critical elements within the entire system

Identify the (sets of) basic elements that cause the top event

Cut set: definition and use

Given a fault tree, it is possible to derive **cut sets**

Cut set: a (sub)set of basic events, such that if they all occur, the top event will occur

Cut set: definition and use

Given a fault tree, it is possible to derive **cut sets**

Cut set: a (sub)set of basic events, such that if they all occur, the top event will occur

A cut set puts basic events into relation with the final outcome top set event

Minimal cut set: smallest set of basic events leading to the top event

Minimal cut sets

The set is minimal if all its events **must occur** to lead to the top-event

Each fault tree has a finite number of unique minimal cut sets

Minimal cut sets

The set is minimal if all its events **must occur** to lead to the top-event

Each fault tree has a finite number of unique minimal cut sets

The number of different basic events in a minimal cut set is called the **order of the cut set**

They identify all distinct ways a top event can occur w.r.t. basic events

Path set

A set of basic events whose **non** (simultaneous) **occurrence** guarantees that the top event does not occur

A minimal path set is one that cannot be reduced without loosing its status as a path set

Qualitative assessment

Qualitative assessment by investigating the minimal cut sets

- Ordering cut sets
- Ranking based on the type of basic events
  - Human error (most critical)
  - Failure of active equipment
  - Failure of passive equipment
- Highlight “small” minimal cut sets

Quantitative analysis

Cut sets are computed and failure probabilities are combined to get the top event probability

- Generate cut sets
- Apply failure data
- Compute probabilities
- Compute criticality measures

Instruments:

FT mathematics (Boolean algebra & probability)

FT approximation methods

Minimal cut sets: computation

The exact identification of the minimal cut sets may be a very hard task for complex FTs

Sub-optimal solutions may be identified via:

- Boolean reduction
- Bottom up reduction algorithms
- Binary Decision Diagrams
- Min Terms method (Shannon decomposition)
- Modularization methods
- Genetic algorithms

Safety Related Analysis (FMEA)

Functional Safety Analyses

- FMEA – Failure Mode and Effects Analysis
- FMEDA – Failure Modes, Effects and Diagnostic Analysis
- FMECA – Failure Mode, Effects and Criticality Analysis

Standards:

IEC 61508 – the generic functional safety standard for electrical and electronic (E/E) systems

ISO 26262 – automotive-specific

A. Nardi and A. Armato, "Functional safety methodologies for automotive applications," *2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, Irvine, CA, USA, 2017, pp. 970-975

Safety functions

- safety functions are what needs to be done to achieve the desired/required level of safety
- elements that are added to the system to mitigate the effects of a fault somewhere else in the system

Safety functions

- safety functions are what needs to be done to achieve the desired/required level of safety
- elements that are added to the system to mitigate the effects of a fault somewhere else in the system

They can be:

- "on demand" (or "low demand")
- "continuous" (or "high demand")   ***   automotive

Safety functions

- safety functions are what needs to be done to achieve the desired/required level of safety
- elements that are added to the system to mitigate the effects of a fault somewhere else in the system

They can be:

- “on demand” (or “low demand”)
- “continuous” (or “high demand”) *** automotive

Generally speaking:

- On demand: found in a protection system separated from the system-under-consideration
- Continuous: part of the system-under-consideration

Analysis: boundary, conditions, resolution

The physical boundaries of the system

which parts are included in the analysis

The initial conditions

what is the operating status when the (top) event occurs

The level of resolution

how detailed the analysis should be (components, faults, …)

Failure Mode and Effect Analysis (FMEA)

Progressively selects the individual components or functions within a system and investigates possible modes of failure

Failure Mode and Effect Analysis (FMEA)

Progressively selects the individual components or functions within a system and investigates possible modes of failure

Considers possible causes for each failure mode and assesses the likely consequences

Failure Mode and Effect Analysis (FMEA)

Progressively selects the individual components or functions within a system and investigates possible modes of failure

Considers possible causes for each failure mode and assesses the likely consequences

Effects of the failure are determined for the unit itself and for the complete system

Failure Mode and Effect Analysis (FMEA)

Progressively selects the individual components or functions within a system and investigates possible modes of failure

Considers possible causes for each failure mode and assesses the likely consequences

Effects of the failure are determined for the unit itself and for the complete system

Possible remedial actions are suggested

Goals

Often used at a functional level, early in the lifecycle

- Before any implementation/prototype is available

Goals

Often used at a functional level, early in the lifecycle

- Before any implementation/prototype is available

May be applied at several levels to refine the analysis

Goals

Often used at a functional level, early in the lifecycle

- Before any implementation/prototype is available

May be applied at several levels to refine the analysis

Used to provide input data for fault tree analysis

Analysis steps

FMEA

Four main steps

- System definition, its functions and components
- Failure modes identification, and their causes
- Effects identification (top events)
- Conclusions and recommendations

Analysis steps

Four main steps

- System definition, its functions and components
- Failure modes identification, and their causes
- Effects identification (top events)
- Conclusions and recommendations

For each failure mode, with a pre-defined scale:

- Severity
- Frequency
- Detection

Risk Priority Number = S•F•D

Failure mode assumptions

Value failure: The unit produces one or several erroneous results which are syntactically correct.

Timing failure: The value of a result is correct, but the result is delivered too late, or too early.

Omission failure: The unit stops producing results for some finite time, and then (after an internal recovery) re-starts to produce correct and timely results again.

Crash failure: The unit stops producing results and does not recover from the failure (the crash is observable).

Silent failure: The unit produces either no results at all, or results that can be identified as being incorrect by all other units (the silent failure is not observable).

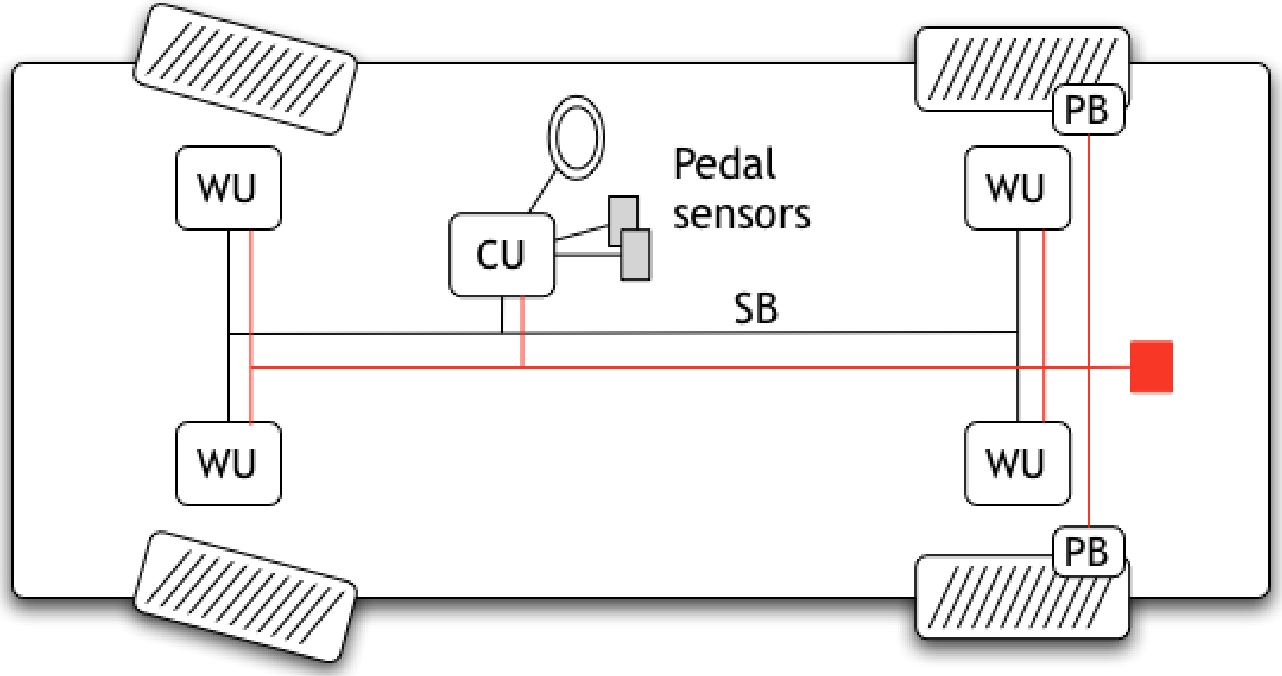
Report sheet

FMEA

Syetem: Performed by: Page: of
Ref. drawing no.: Date:

Description of unit			Description of Failure			Effect of Failure		Failure rate	Severity ranking	Risk reducing measures	Comments
Ref. No.	Function	Operational Mode	Failure mode	Failure Mechanisms	Detection of Failure	On Components in the subsystem	Ont the system function				

Example: X-by-wire system



WU: wheel unit
CU: central unit
SB: serial bus

PB: parking brake

Report

FMEA

Function	Class of Failure	Failure	Effects on the System	Severity
Acceleration	Omission	No accelearation available	Car eventually stops	Marginal
	Commission	Sudden acceleration	Car increases its speed rapidly	Critical
	Stuck	Constant acceleration	Car increases its speed	Critical
Retardation	Omission	No retardation possible	Car can't stop	Catastrophic
	Commission	Wheels lock	Car stops suddenly	Catastrophic
	Stuck	Constant retardation	Car continues to brake	Critical
Steering	Omission	No control of steering	Car looses stability	Catastrophic
	Commission	Steering when not requested	Car changes trajectory unintedended	Catastrophic
	Stuck	Car maintains a turning angle	Car continues on turning trajectory	Critical

References

Fault Tree Handbook

Marvin Rausand's Chapter on "System Analysis Fault Tree Analysis"

Arnljot Hoyland, Marvin Rausand, "System Reliability Theory", John Wiley & Sons, Inc., 1994

Federal Aviation Administration, "Guide to Reusable Launch and Reentry Vehicle Reliability Analysis", 2005

http://sharpe.pratt.duke.edu/

TOPIC QUESTIONS

How does the system react to the occurrence of a fault?

What are the most critical faults?

How reliable or available is the system?

TOPICS

Reliability/Availability estimation

\+ RBDs

\+ FTs

\+ FMEA