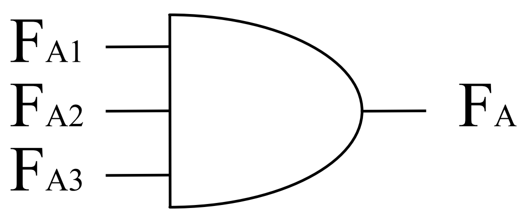
Fault Trees

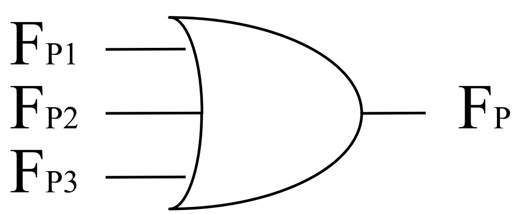
- Fault Trees
  - dual of Reliability Block Diagram
  - logic failure diagram
  - think in terms of logic where
    - 0 = operating, 1 = failed
- AND Gate
  - all inputs must fail for the gate to fail
- OR Gate
  - any input failure causes the gate to fail
- k-of-n Gate
  - k or more input failures cause gate to fail

e.g. Triplex Bus Guardian

- Active mode
  - M_1 and M_2 and M_3 fail =>
  - AND Gate

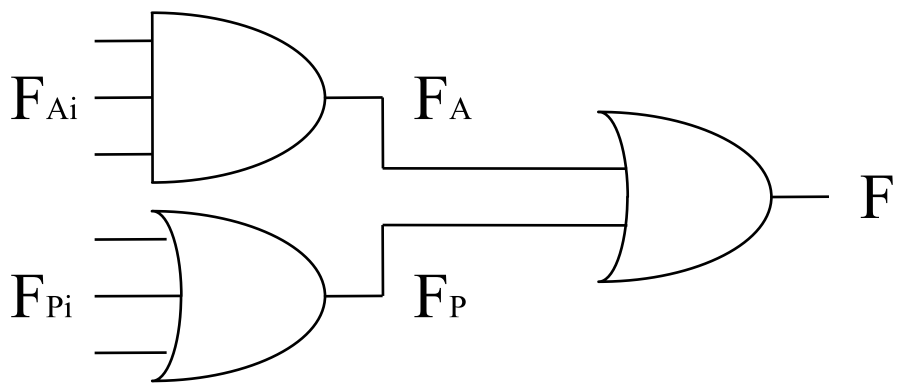


- Passive Mode
  - "cutoff" with any single unit failure =>
  - OR Gate



e.g. Triplex Bus Guardian

- Total Failure
  - caused by either active or passive mode



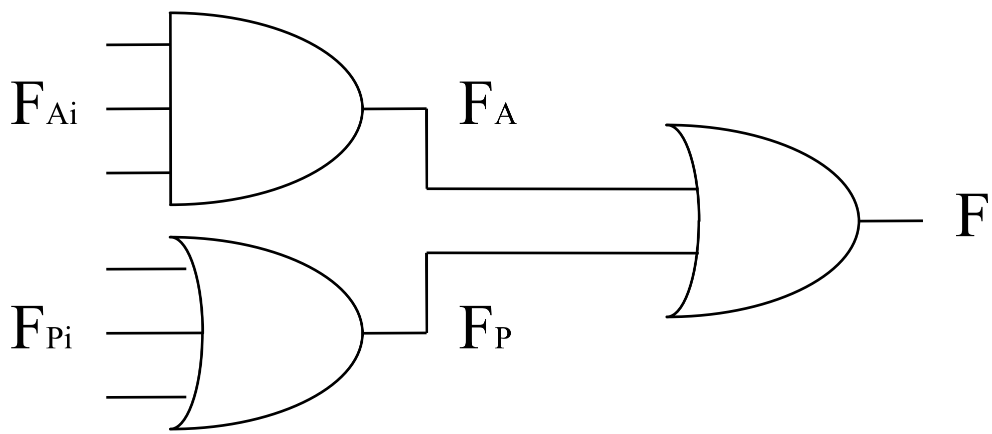
e.g. Triplex Bus Guardian

- How can one use the fault tree effectively to isolate those parts of the system that need reliability considerations?

e.g. Triplex Bus Guardian

- Combined fault model

$$Q(1000h) = 0.9851243 \cdot 10^{-6}$$



$$Q(1000h) = 0.295545 \cdot 10^{-1}$$

Examples

- Simple Passive TMR (no diagnosis)
  - RBD = (2 of 3): 2 operable => System operable
  - F-Tree = (2 of 3): 2 failed => System failed
- Simple TMR with *Benign* failures
  - RBD = (1 of 3): 1 operable => System operable
  - F-Tree = (3 of 3): 3 failed => System failed
- Summary
  - Parallel => AND
  - Series => OR
  - K-of-N => (n-k+1 of n)

SHARPE

- SYMBOLIC HIERARCHICAL AUTOMATED RELIABILITY AND PERFORMANCE EVALUATOR
- SHARPE provides a specification language and analysis algorithms for the following model types:
  - reliability block diagrams
  - fault trees
  - reliability graphs
  - series-parallel acyclic directed graphs
  - product-form queuing networks
  - Markov and semi-Markov chains
  - generalized stochastic Petri nets

Analysis using SHARPE

- we will be using Mobius this year, but here is a glimpse into what SHARPE looks like.
- SHARPE and SPNP are available to us with a license from Duke University, if anybody is interested.

- Below are three different SHARPE programs and output. The first two examples don't show all the details of the programs.

Bus Guardian (Active)

```
* SYSTEM: TRIPLEX BUS GUARDIAN -- ACTIVE FAILURE MODE
* MODEL: RELIABILITY BLOCK DIAGRAM
* -- Model Definition: block name, components, connectivity --
*
block bus_gd_act
comp z exp(lamact)
parallel z3 z z z
end

* Bind Values to Variable Names
*
bind
lamact 1.0*10^-5
end
```

```
* -- Calculate CDF for System Failure
*
cdf(bus_gd_act)

* -- Evaluate CDF at Specified Points
*
eval(bus_gd_act) 9   11   1
eval(bus_gd_act) 90  110  10
eval(bus_gd_act) 900 1100 100

end
```

Bus Guardian (Active)

CDF for system bus_gd_act:

```
   1.0000e+00 t(  0) exp( 0.0000e+00 t)
 + -3.0000e+00 t(  0) exp(-1.0000e-05 t)
 +  3.0000e+00 t(  0) exp(-2.0000e-05 t)
 + -1.0000e+00 t(  0) exp(-3.0000e-05 t)
```

mean: 1.8333e+05
variance: 1.3611e+10

----

system bus_gd_act

t	F(t)
9.0000 e+00	0.0000 e+00
1.0000 e+01	0.0000 e+00
1.1000 e+01	0.0000 e+00

system bus_gd_act

t	F(t)
9.0000 e+01	0.0000 e+00
1.0000 e+02	0.0000 e+00
1.1000 e+02	1.3288 e-09

----

system bus_gd_act

t	F(t)
9.0000 e+02	7.1923 e-07
1.0000 e+03	9.8512 e-07
1.1000 e+03	1.3092 e-06

Bus Guardian (Passive)

```
* SYSTEM: TRIPLEX BUS GUARDIAN -- PASSIVE FAILURE MODE
* MODEL: RELIABILITY BLOCK DIAGRAM
* -- Model Definition: block name, components, connectivity --
*
block bus_gd_pas
comp z exp(lampas)
series z3 z z z
end

* -- Bind Values to Variable Names --
*
bind
lampas 1.0*10^-5
end

* -- Calculate CDF for System Failure --
*
cdf(bus_gd_pas)

* -- Evaluate CDF at Specified Points --
*
eval(bus_gd_pas)   1   5   2
eval(bus_gd_pas)  10  50  20
eval(bus_gd_pas) 100 500 200

end
```

Bus Guardian (Passive)

CDF for system bus_gd_pas:

1.0000e+00 t(0) exp(0.0000e+00 t)
+ -1.0000e+00 t(0) exp(-3.0000e-05 t)

mean: 3.3333e+04
variance: 1.1111e+09

----

system bus_gd_pas

t	F(t)
1.0000 e+00	3.0000 e-05
3.0000 e+00	8.9996 e-05
5.0000 e+00	1.4999 e-04

system bus_gd_pas

t	F(t)
1.0000 e+01	2.9996 e-04
3.0000 e+01	8.9960 e-04
5.0000 e+01	1.4989 e-03

----

system bus_gd_pas

t	F(t)
1.0000 e+02	2.9955 e-03
3.0000 e+02	8.9596 e-03
5.0000 e+02	1.4888 e-02

SYSTEM: TRIPLEX BUS GUARDIAN -- ACTIVE FAILURE MODE
MODEL: RELIABILITY BLOCK DIAGRAM

```
block bus_gd_act
comp z exp(lamact)
parallel z3 z z z
end
```

This is the RBD defined as 3 parallel modules

```
block bus_gd_act3
comp z exp(lamact3)
end
```

Now I pretend this is the same as using 1/MTTF (calculated for a parallel system) in a simple 1 module expression.

```
bind
lamact 1.0*10^-5
lamact3 1/(1.8333*10^5)
end
```

Bind Values to Variable Names

```
cdf(bus_gd_act)
cdf(bus_gd_act3)
```

Calculate CDF for System Failure

```
eval(bus_gd_act) 900 1100 100
eval(bus_gd_act3) 900 1100 100
end
```

Evaluate CDF at Specified Points. Even though the MTTF are the same, the CDFs are different.

CDF for system bus_gd_act:

```
  1.0000e+00 t(  0) exp( 0.0000e+00 t)
+ -3.0000e+00 t(  0) exp(-1.0000e-05 t)
+  3.0000e+00 t(  0) exp(-2.0000e-05 t)
+ -1.0000e+00 t(  0) exp(-3.0000e-05 t)
```

mean: 1.8333e+05
variance: 1.3611e+10

CDF for system bus_gd_act3:

```
  1.0000e+00 t(  0) exp( 0.0000e+00 t)
+ -1.0000e+00 t(  0) exp(-5.4546e-06 t)
```

mean: 1.8333e+05
variance: 3.3610e+10

system bus_gd_act

t	F(t)
9.0000 e+02	7.1923 e-07
1.0000 e+03	9.8512 e-07
1.1000 e+03	1.3092 e-06

----

system bus_gd_act3

t	F(t)
9.0000 e+02	4.8971 e-03
1.0000 e+03	5.4398 e-03
1.1000 e+03	5.9821 e-03