# Safety and Reliability of Embedded Systems - SRES (WS 19/20)
# Problem Set 5

**Problem 1: Fault Tree Analysis K out of N system**

You have to evaluate the failure of a 2 out of 3 system with the help of Fault Tree Analysis. A 2 out of 3 system consists of 3 components and for the system to be operating at least two out of these 3 components have to be operating. Pl ease consider the following events for your analysis:

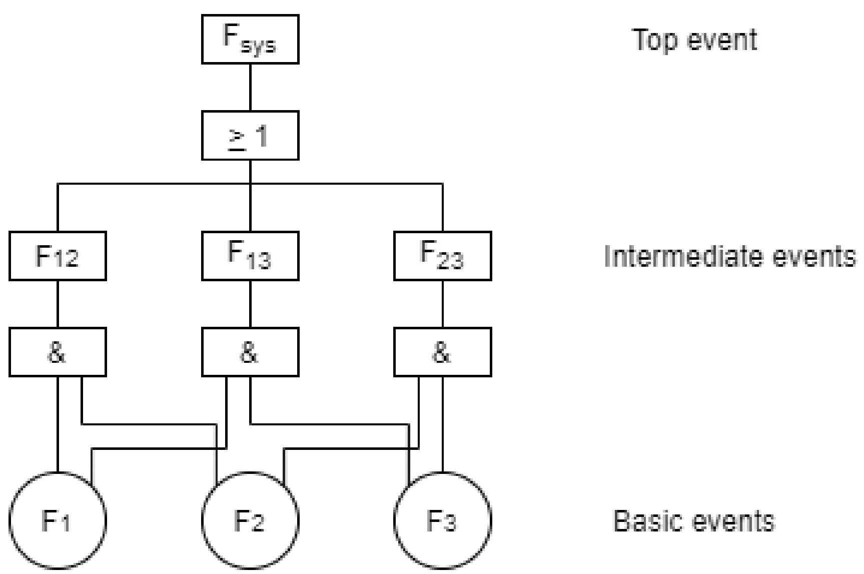
| | | | |
|---|---|---|---|
| Top event | $F_{sys}$ | 2 out of 3 system fails | $f_{sys}$ |
| Intermediate event | $F_{12}$, $F_{13}$, $F_{23}$ | Two components fail | $f_{12}$, $f_{13}$, $f_{23}$ |
| Basic event | $F_1$, $F_2$, $F_3$ | A single component fails | $f_1$, $f_2$, $f_3$ |

The probability of failure of each single component is assumed to be $f_c = 0.03$

a. Please draw the corresponding fault tree using the above event names.

-

Draw a fault tree for a 2 out of 3 system




b. Try to calculate the probability of failure of the system f sys by applying standard gate formulas known from lecture in a bottom up fashion.

-

Calculate the failure probability $f_{sys}$

$$f_1 = f_2 = f_3 = 0.03$$

$$f_{12} = f_1 * f_2 = 0.0009 = 9 * 10^{-4} = f_{13} = f_{23}$$

$$f_{sys} = f_{12} \vee f_{13} \vee f_{23}$$

$$f_{sys} = 1 - \prod_{i=1}^{3}(1 - P_i)$$

$$f_{sys} = 1 - (1 - 9 * 10^{-4})^3$$

$$f_{sys} \approx 0.002698 = 2.698 * 10^{-3}$$


Nazrul.arif93@gmail.com

c. Now determine the minimal cut sets and calculate an approximation for $f_{sys}$.

-

Calculate the MCSs and approximate $f_{sys}$

| **Recap**: | |
|---|---|
| Cut set | Set of basic events that trigger the top events |
| Minimal Cut Set (MCS) | Cut set where the removal of any element results in the top event no longer being triggered. |

MCSs: $\{F_1, F_2\},\ \{F_1, F_3\},\ \{F_2, F_3\}$

*Note:*

$\{F_1, F_2, F_3\}$ *is also a Cut Set but not Minimal Cut Set (MCS)*

Approximations: $f_{sys} \approx f_1 * f_2 + f_1 * f_3 + f_2 * f_3 = 0.0027 = 2.7 * 10^{-3}$

*Note:*

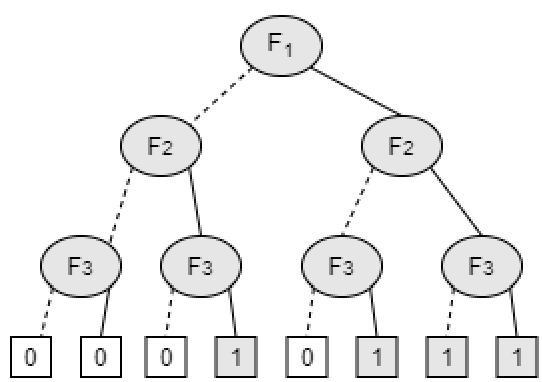
*Approximation:* $P\ (A \vee B) \approx P_A + P_B$

*Correct:* $P\ (A \vee B) \approx P_A + P_B - P_A \,.\, P_B$

d. Finally, draw a binary decision tree for f sys using the variable order $F_1 \rightarrow F_2 \rightarrow F_3$. Convert the tree e into a reduced ordered binary decision diagram (ROBDD). Annotate the diagram with probabilities and again calculate the availability $f_{sys}$.

-

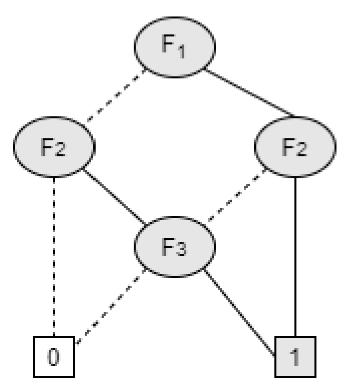
**Draw the BDD** for $f_{sys}$ (variable order: $F_1 > F_2 > F_3$ )




**Draw a ROBDD** (variable order: $F_1 > F_2 > F_3$ )

*Note:*

- *Identify isomorphic subgroups*
- *Eliminate redundant nodes and edges*



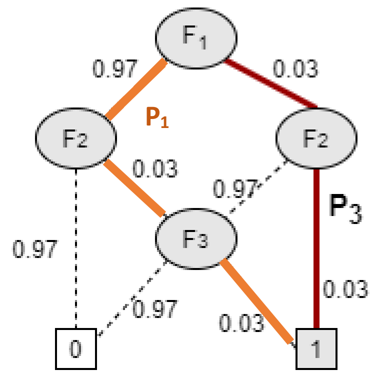
**Calculate** $f_{sys}$

$P_1 = 0.97 * 0.03 * 0.03$

$P_2 = 0.03 * 0.97 * 0.03$

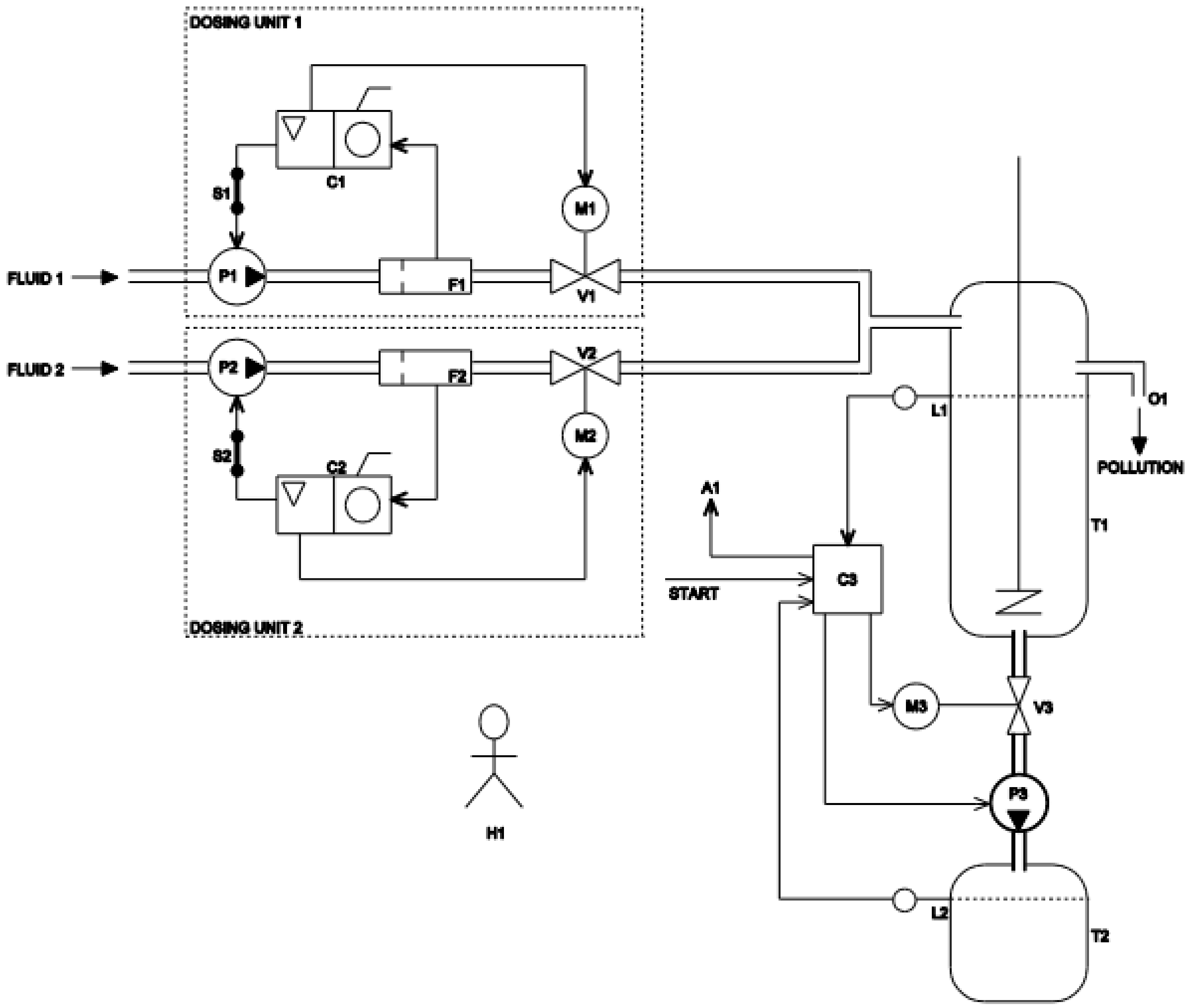
$P_3 = 0.03 * 0.03$

$f_{sys} = P_1 + P_2 + P_3 = 2.6 * 10^{-3}$



**Problem 2: Detailed – Fault Tree Analysis**

The purpose of the following plant is to mix FLUID 1 and FLUID 2 within tank T1. If a particular mixing time has elapsed, the operator H1 manually sends a START signal to controller C3, which in turn opens valve V3 via an electric motor M3 and initiates the filling process of tank T2. The mixed product is pumped into tank T2 using pump P3 until the level at level sensor L2 is reached. Pump P3 is switched off and valve V3 is closed again automatically by the controller.




In order to exactly dose the amount of fluids mixed, two dosing units (DOSING UNIT 1, DOSING UNIT 2) with mass flow meters (F1, F2) are used. Each flow meter sends electrical impulses to a software counter (C1, C2), using the impulse frequency as a figure for the amount of mass transported per unit time. If a pre-defined impulse count is reached, the software counter switches off the corresponding pump (P1, P2) and shuts the corresponding valve (V1, V2) via an electric motor (M1, M2). It is assumed here that if both dosing units work properly, the amount of mixed fluid within tank T1 is always below level L1.

In order to avoid overfilling, tank T1 is equipped with a safety mechanism. This mechanism is realized by a level sensor L1 built into the tank. If the corresponding level is reached, a signal is sent to controller C3, which immediately starts an emergency filling process of tank T2. In addition, the alarm A1 is triggered, indicating the hazardous condition to the operator. Before tank T2 is completely filled, the operator must switch off pumps P1 and P2 manually by interrupting the emergency switches S1 and S2.

If the safety mechanism fails, the mixed product might exit the tank via pipe O1, thereby causing pollution to the environment.

In the following, a fault tree analysis for the top-event POLLUTION should be performed. For this, please assume that the emergency switches S1 and S2 never fail and that all basic events are stochastically independent.

e) As a first step, draw a high-level fault tree for the top-event p (POLLUTION), thereby only considering the intermediate events d1 (DOSING UNIT 1 FAILS UNSAFE), d2 (DOSING UNIT 2 FAILS UNSAFE) and sm (SAFETY MECHANISM FAILS). *Determine the corresponding Boolean function*. *What does "dosing unit fails unsafe" mean in this example?*

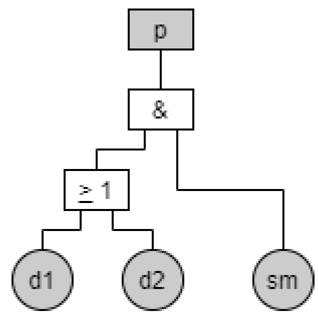
-

p: pollution

d1: dosing unit 1 fails unsafe

d2: dosing unit 2 fails unsafe

sm: safety mechanism fails




- *Determine the Boolean function*
  $p = (d1 \vee d2) \wedge sm$
- *When does a dosing unit fail unsafe?*
  If liquid leaks uncontrollably.

f) Refine the intermediate events d1 (DOSING UNIT 1 FAILS UNSAFE) and d2 (DOSING UNIT 2 FAILS UNSAFE) in separate fault tree diagrams. *Determine the corresponding Boolean functions*. *Eliminate repeated events* by restructuring your expressions and *redraw the diagrams accordingly*. For the dosing units we have two intermediate events, the pump not being switched off and the valve not being closed. The pump is dependent on the counter(c1/2) and the mass flow meter (f1/2). The valve is dependent on the motor(m1/2), the counter(c1/2), and a correct working valve(v1/2).

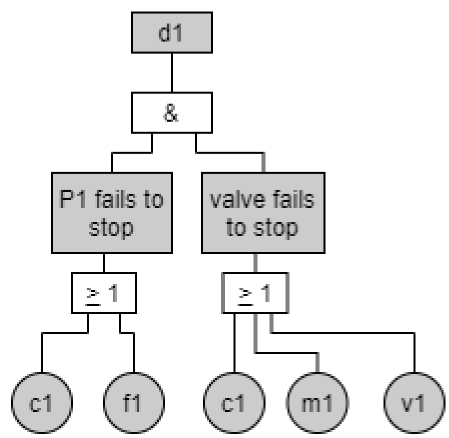
-

c1: software counter

f1: flow meter

m1: motor

v1: valve

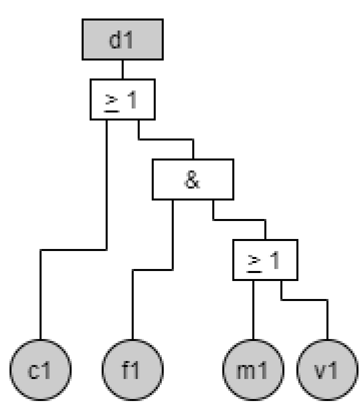



- *Determine the Boolean function*

  $\mathrm{d1} = \mathrm{P}\,(\text{pump fails to close}) \wedge \mathrm{P}\,(\text{valve fails to close})$

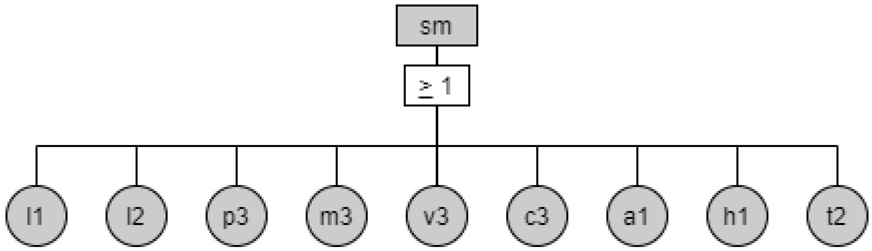
  $\mathrm{d1} = (c_1 \vee f_1) \wedge (c_1 \vee m_1 \vee v_1)$

  $\mathrm{d1} = c_1 \vee (f_1 \wedge (m_1 \vee v_1))$
- Eliminate repeated events
  Problem: not a tree anymore
- Instead: Construct new tree
  $\mathrm{d1} = c_1 \vee (f_1 \wedge (m_1 \vee v_1))$



g) Refine the intermediate event sm (SAFETY MECHANISM FAILS) in a separate fault tree diagram. *__Determine the corresponding Boolean function__*. *(Hint: the variable order in g) shows the involved events)*

-




l1, l2: level sensors
p3: pump
m3: motor
v3: value
c3: controller
a1: alarm
h1: human
t2: tank

Boolean expression: $sm = l1 \vee l2 \vee p3 \vee m3 \vee v3 \vee c3 \vee a1 \vee h1 \vee t2$

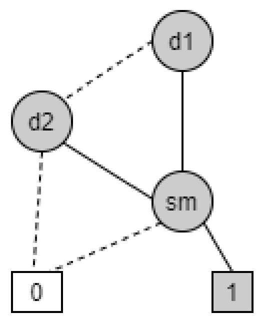
h) *__Develop the ROBDD__* for the Boolean expression *__for p as obtained in question__* e

a. using the variable order: d1 → d2 → sm

-

p: pollution
d1: dosing unit 1 fails unsafe
d2: dosing unit 2 fails unsafe
sm: safety mechanism fails

$p = (d1 \vee d2) \wedge sm$



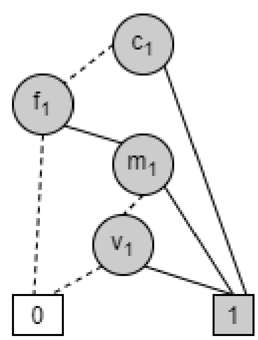

i) *__Develop the ROBDD__* for the Boolean expression *__for d1 as obtained in question__* f

b. using the variable order: c1 → f1 → m1 → v1

-

c1: software counter
f1: flow meter
m1: motor
v1: valve

d1 $= c_1 + \ f_1 \ . \ (\ m_1 + \ v_1))$




j) *__Develop the ROBDD__* for the Boolean expression *__for d2 as obtained in question f__*

c. using the variable order: c2 → f2 → m2 → v2

-

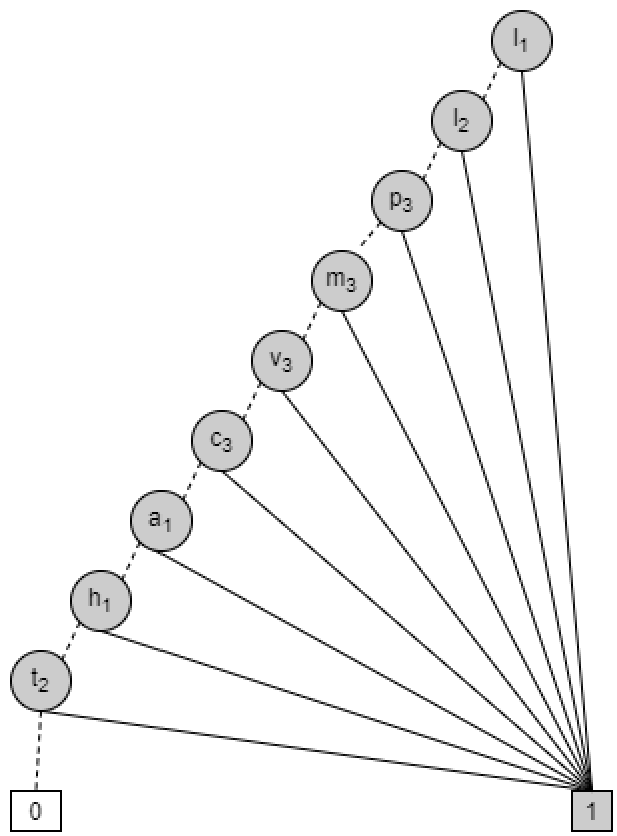
Same process as i

k) *Develop the ROBDD* for the Boolean expression *for sm as obtained in question q*

d. using the variable order: l1 → l2 → p3 → m3 → v3 → c3 → a1 → h1 → t2

-

| | | |
|---|---|---|
| l1, l2: level sensors | v3: value | h1: human |
| p3: pump | c3: controller | t2: tank |
| m3: motor | a1: alarm | |



l) Assuming the basic event probabilities as given in the table below, now *__calculate the probability for the top-event p__* (POLLUTION).

| | | |
|---|---|---|
| t2 | TANK T2 FULL | 0.1 |
| c1/2 | COUNTER C1/2 DEFECTIVE | 0.05 |
| a1 | ALARM A1 DEFECTIVE | 0.03 |
| h1 | OPERATOR H1 SLEEPS | 0.1 |
| m1/2 | MOTOR M1/2 DEFECTIVE | 0.06 |
| v1/2 | VALVE V1/2 STUCK OPEN | 0.08 |
| f1/2 | MASS FLOW METER F1/2 DEFECTIVE (NO IMPULSES) | 0.05 |
| l1/2 | LEVEL SENSOR L1/2 DEFECTIVE | 0.05 |
| p3 | PUMP P3 DEFECTIVE | 0.06 |
| m3 | MOTOR M3 DEFECTIVE | 0.06 |
| v3 | VALVE V3 STUCK CLOSED | 0.08 |
| c3 | CONTROLLER C3 DEFECTIVE | 0.05 |

-

$p = (d1 + d2) * sm$

$d1 = d2 = 0.056422 \ \approx (c1 + f1 * (m1 + v1))$

$sm = 0.45289$

So,

$p = (0.05642 * 0.45289) + (0.9435 * 0.05642 * 0.45289)$

$p = 0.049$