Business from technology

I&C modelling in FinPSA software

Open PSA workshop 10.-11.12.2012
Tero Tyrväinen, Ilkka Niemelä (STUK)
VTT Technical Research Centre of Finland

I&C modelling in FinPSA

- I&C modelling feature of FinPSA is based on RELVEC algorithm.
- Models are built with success logic.
- Models are written in text files with simple and compact expressions.
- Fault trees can contain links to control tasks of I&C model and I&C model can include links to top events of fault trees.
- I&C models are automatically transformed into fault trees.

I&C Model Based on RELVEC Algorithm

RELVEC - A TOOL FOR CONTROL SYSTEM RELIABILITY ANALYSIS

Technical Research Centre of Finland, VTT,
Electrical Engineering Laboratory,
ESPOO,
FINLAND.

(Received for publication 7th July 1983)

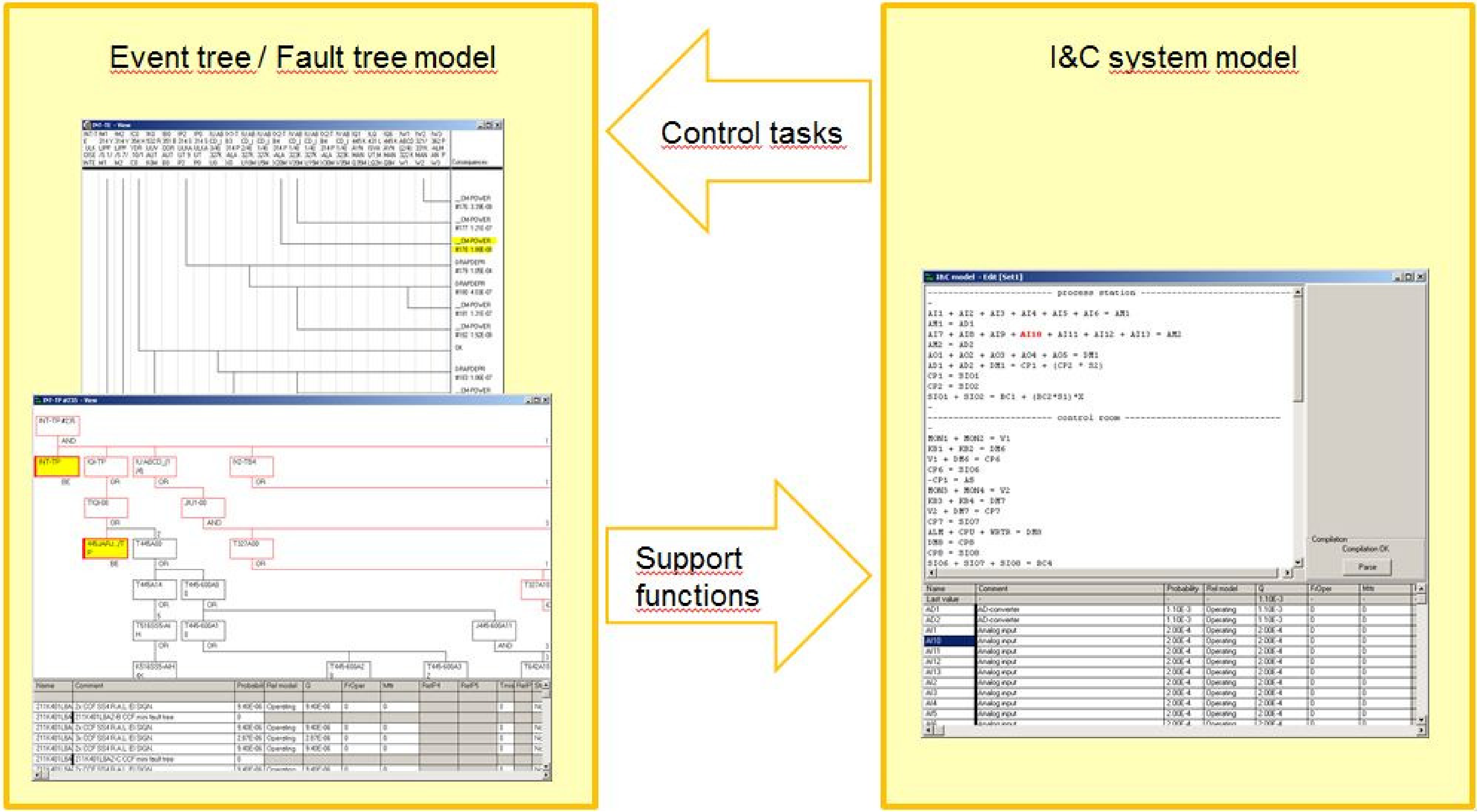
Background is communication matrix, expressed with vectors

RELVEC = **REL**iability **VEC**tors

Developed in 1980s for reliability analysis and design of distributed control systems:

- Oil refinery
- Paper mills
- Power plants
- Hazardous plants
- Satellite earth station
- Offshore
- PRA fault tree analysis

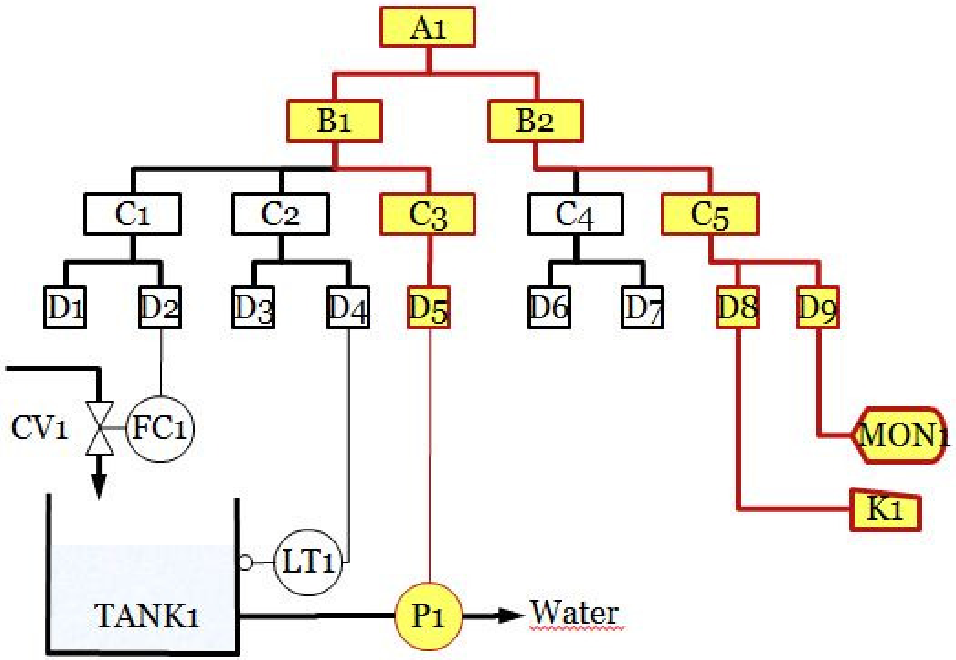
Isolation = Interface



Communication: Path Nets

Control tasks define functional entities inside the I&C system.

Path nets of control tasks are created and converted to fault trees.

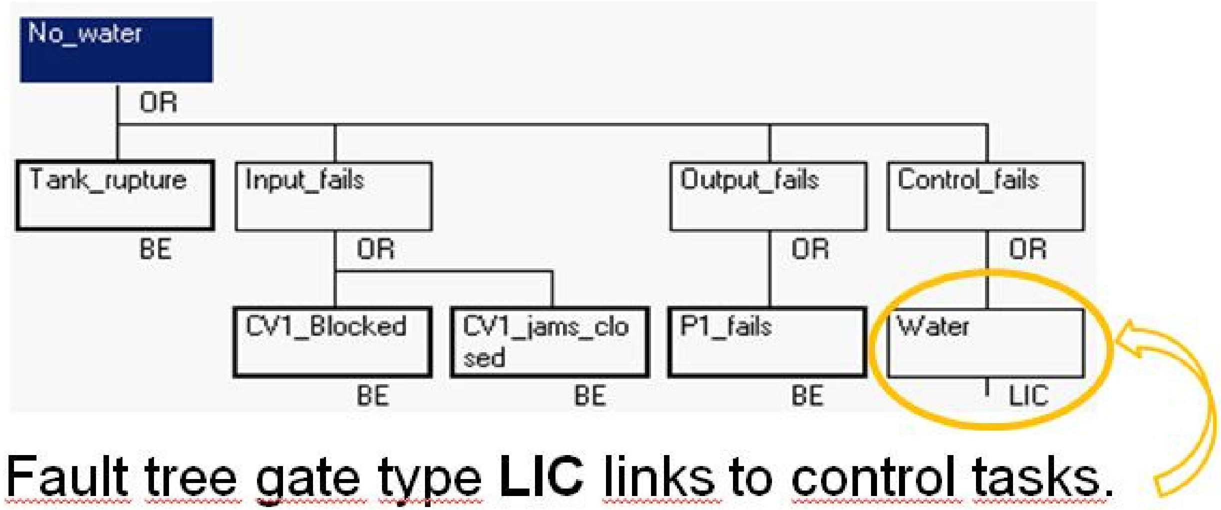


$ control system
D1 + D2 = C1
D3 + D4 = C2
D5 = C3
D6 + D7 = C4
D8 + D9 = C5
C1 + C2 + C3 = B1
C4 + C5 = B2
B1 + B2 = A1

$ interface to instrumentation
FC1 = D2
LT1 = D4
P1 = D5
K1 = D8
MON1 = D9

$ control tasks
Manual_P1 = MON1 * K1 * P1
L_TANK1 = FC1 * LT1
Water = P1 * L_TANK1 * CONTR
CONTR = MON1 * K1

Interface: Fault trees ↔ I&C Model



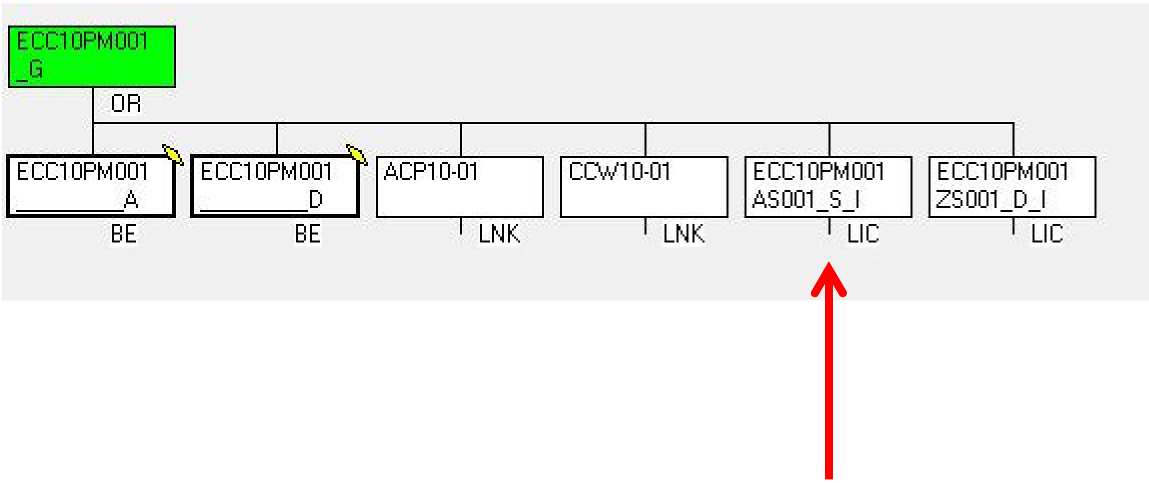
- Fault tree gate type **LIC** links to control tasks.
- Fault tree of the task is created and linked to LIC gate
  - Extremely fast routines, practically no delay in fault tree construction.
  - I&C fault tree is automatically created also when minimal cut set is double-clicked
- I&C model links to fault tree using fault tree name

  Rack01 = Pow01 * Rack01 * **DC24V_Tr_A**

Example PSA model

- Fictive and simplified nuclear power plant (BWR).
- 4-redundant safety systems.
- Fault trees for AC power system, component cooling water system, emergency core cooling system, emergency feedwater system, depressurasation valve system, residual heat removal system, service water system and main feedwater system.
- Event trees for large LOCA, loss of main feedwater, transient and loss-of-offsite power.
- I&C systems are modelled using the I&C modelling feature of FinPSA.

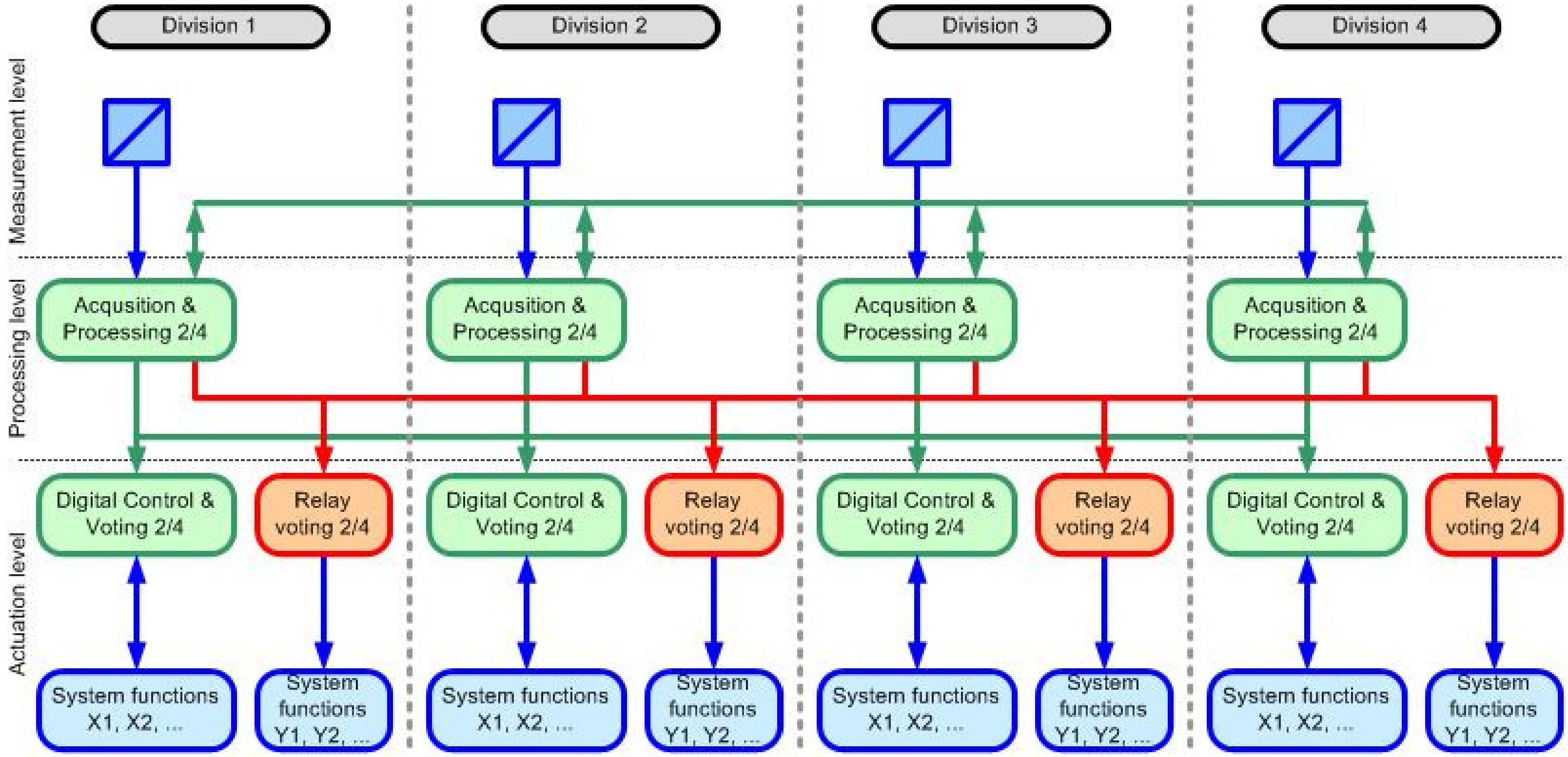
Fault tree of ECC system pump



A link to an I&C-model.

Representing the failure of the start signal.

I&C system


Division 1
Division 2
Division 3
Division 4
Measurement level
Processing level
Actuation level
Acqusition & Processing 2/4
Acqusition & Processing 2/4
Acqusition & Processing 2/4
Acqusition & Processing 2/4
Digital Control & Voting 2/4
Relay voting 2/4
Digital Control & Voting 2/4
Relay voting 2/4
Digital Control & Voting 2/4
Relay voting 2/4
Digital Control & Voting 2/4
Relay voting 2/4
System functions X1, X2, ...
System functions Y1, Y2, ...
System functions X1, X2, ...
System functions Y1, Y2, ...
System functions X1, X2, ...
System functions Y1, Y2, ...
System functions X1, X2, ...
System functions Y1, Y2, ...

ECC pump start signal I&C model

```
$ APU A
RPS10PU001I0002_4_I = RPS40PU001VL004_F_S * RPS14LLAPU1APU4_F_S
RPS10PU001I0002_3_I = RPS30PU001VL004_F_S * RPS13LLAPU1APU3_F_S
RPS10PU001I0002_2_I = RPS20PU001VL004_F_S * RPS12LLAPU1APU2_F_S
RPS10PU001I0002_1_I = RPS10PU001VL004_F_S
```

Functioning water level sensor and functioning communication link required.

Links to top events of fault trees.

```
$ APU B
RPS20PU001I0002_4_I = RPS40PU001VL004_F_S * RPS24LLAPU2APU4_F_S
RPS20PU001I0002_3_I = RPS30PU001VL004_F_S * RPS23LLAPU2APU3_F_S
RPS20PU001I0002_2_I = RPS20PU001VL004_F_S
RPS20PU001I0002_1_I = RPS10PU001VL004_F_S * RPS21LLAPU2APU1_F_S

$ APU C
RPS30PU001I0002_4_I = RPS40PU001VL004_F_S * RPS34LLAPU3APU4_F_S
RPS30PU001I0002_3_I = RPS30PU001VL004_F_S
RPS30PU001I0002_2_I = RPS20PU001VL004_F_S * RPS32LLAPU3APU2_F_S
RPS30PU001I0002_1_I = RPS10PU001VL004_F_S * RPS31LLAPU3APU1_F_S

$ APU D
RPS40PU001I0002_4_I = RPS40PU001VL004_F_S
RPS40PU001I0002_3_I = RPS30PU001VL004_F_S * RPS43LLAPU4APU3_F_S
RPS40PU001I0002_2_I = RPS20PU001VL004_F_S * RPS42LLAPU4APU2_F_S
RPS40PU001I0002_1_I = RPS10PU001VL004_F_S * RPS41LLAPU4APU1_F_S

$ APU 2/4 voting
RPS10PU001I0002_V_I = <2 RPS10PU001I0002_1_I + RPS10PU001I0002_2_I + RPS10PU001I0002_3_I + RPS10PU001I0002_4_I>
RPS20PU001I0002_V_I = <2 RPS20PU001I0002_1_I + RPS20PU001I0002_2_I + RPS20PU001I0002_3_I + RPS20PU001I0002_4_I>
RPS30PU001I0002_V_I = <2 RPS30PU001I0002_1_I + RPS30PU001I0002_2_I + RPS30PU001I0002_3_I + RPS30PU001I0002_4_I>
RPS40PU001I0002_V_I = <2 RPS40PU001I0002_1_I + RPS40PU001I0002_2_I + RPS40PU001I0002_3_I + RPS40PU001I0002_4_I>

$ Signal from APU
RPS10PU002EC001_4_I = RPS40PU001I0002_V_I * RPS41LLAPU4VU01_F_S
RPS10PU002EC001_3_I = RPS30PU001I0002_V_I * RPS31LLAPU3VU01_F_S
RPS10PU002EC001_2_I = RPS20PU001I0002_V_I * RPS21LLAPU2VU01_F_S
RPS10PU002EC001_1_I = RPS10PU001I0002_V_I * RPS10LLAPU1VU01_F_S
```

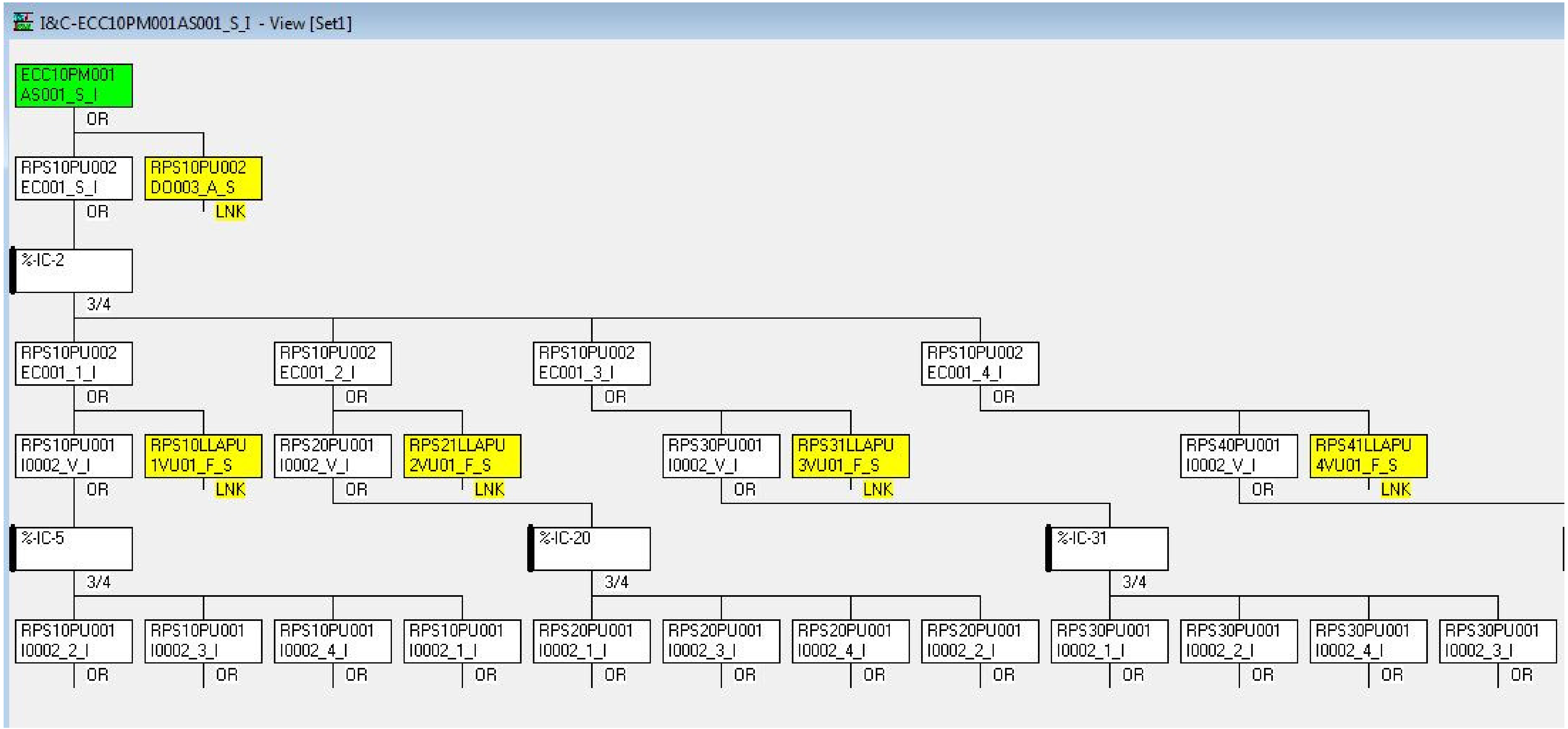
Positive voting result from APU and functioning communication link required.

```
$ Actuation 2/4 voting
RPS10PU002EC001_S_I = <2 RPS10PU002EC001_1_I + RPS10PU002EC001_2_I + RPS10PU002EC001_3_I + RPS10PU002EC001_4_I>

$ Start signal from DCV
ECC10PM001AS001_S_I = RPS10PU002EC001_S_I * RPS10PU002DO003_A_S
```

Positive voting result and functioning digital output module required.

I&C model is automatically transformed into fault tree



Fail-safe principles

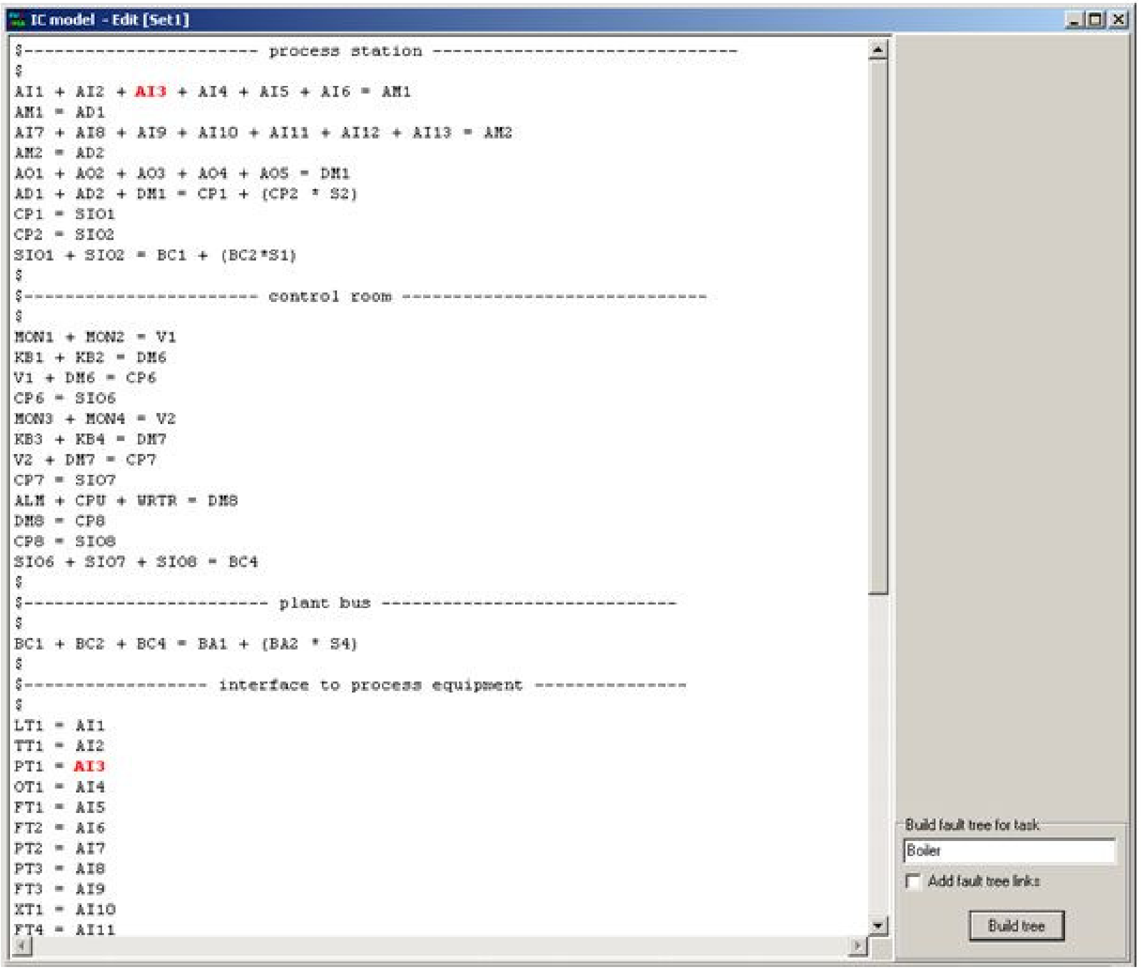
- How to handle detected failures in I&C system?
- When a failure is detected, the signal can be replaced by a default value of 0 or 1.
- Fixed binary variables can be used to control the fail-safe principle.

```
RPS10PU001I0002_4_I = RPS40PU001VL004_F_S * RPS14LLAPU1APU4_F_S * ((RPS40PU001VL004_E_S * RPS14LLAPU1APU4_E_S) + RPSC0PU001I0002_V_DE)
```

Undetected failures — Detected failures — Controlling variable

- If the first and second detected failures are handled differently, the modelling is much more challenging.

Implementation in FinPSA



Integrated to PRA data base:

- CCFs between I&C model parts
- Hazard Table mapping
  - Fires, floods, seismic…

Display and analyze fault tree of any control task

Natural import & export of model

Summary

- I&C model is isolated from PRA model via interface
  - Simple communication system model
  - Compact and computationally efficient representation
  - Not much new as such
- Isolation makes it possible to develop I&C modeling as a part of a full-scale PRA model
  - I&C model development can take its own course
- Text-based model offers freedom of future development
  - Expansion of modelling language
    - Dynamic properties

References

- Niemelä I. Isolation of I&C model from PRA fault tree model. PSAM11 proceedings, 2012.

VTT - 70 years of

technology for business and society