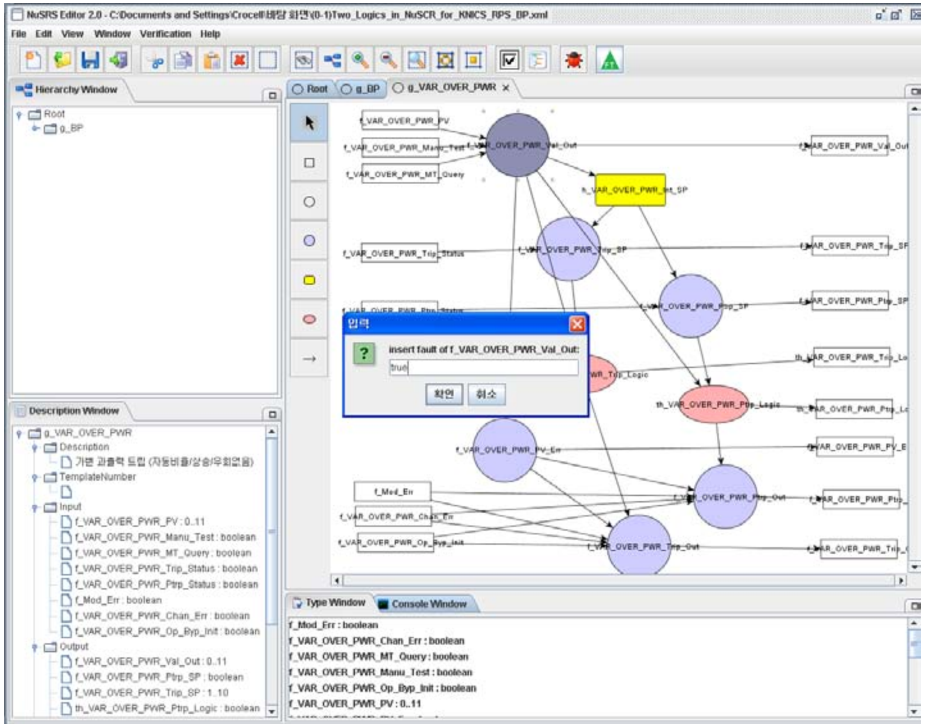
NuFTA : A CASE Tool for Automatic Software Fault Tree Analysis

Sanghyun Yun, Dong-Ah Lee, Junbeom Yoo

Division of Computer Science and Engineering, Konkuk University

Dependable Software Laboratory

NuFTA


NuSRS Editor 2.0 - C:\Documents and Settings\Crocel\바탕 화면\(0-1)Two_Logics_in_NuSCR_for_KNICS_RPS_BP.xml

입력

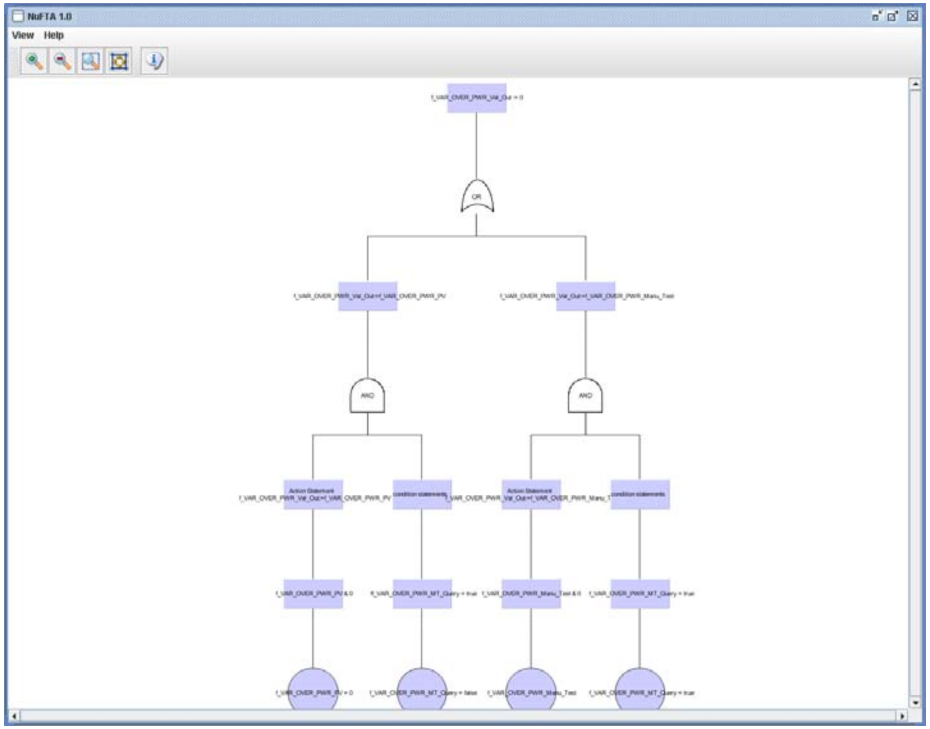
insert fault of f_VAR_OVER_PWR_Val_Out:

true

확인 취소

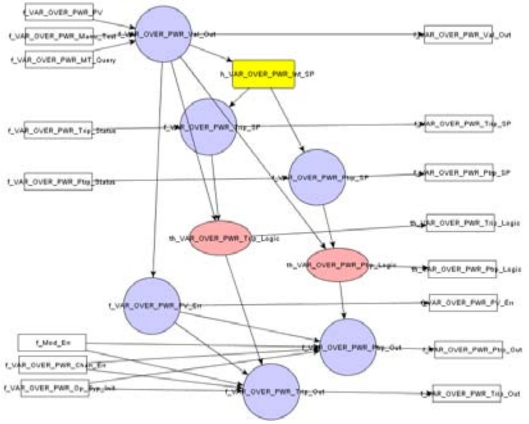
- g_VAR_OVER_PWR
  - Description
    - 가변 과출력 트립 (자동비율/상승/우회없음)
  - TemplateNumber
  - Input
    - f_VAR_OVER_PWR_PV : 0..11
    - f_VAR_OVER_PWR_Manu_Test : boolean
    - f_VAR_OVER_PWR_MT_Query : boolean
    - f_VAR_OVER_PWR_Trip_Status : boolean
    - f_VAR_OVER_PWR_Ptrp_Status : boolean
    - f_Mod_Err : boolean
    - f_VAR_OVER_PWR_Chan_Err : boolean
    - f_VAR_OVER_PWR_Op_Byp_Init : boolean
  - Output
    - f_VAR_OVER_PWR_Val_Out : 0..11
    - f_VAR_OVER_PWR_Ptrp_SP : boolean
    - f_VAR_OVER_PWR_Trip_SP : 1..10
    - th_VAR_OVER_PWR_Ptrp_Logic : boolean

```
f_Mod_Err : boolean
f_VAR_OVER_PWR_Chan_Err : boolean
f_VAR_OVER_PWR_MT_Query : boolean
f_VAR_OVER_PWR_Manu_Test : boolean
f_VAR_OVER_PWR_Op_Byp_Init : boolean
f_VAR_OVER_PWR_PV : 0..11
```

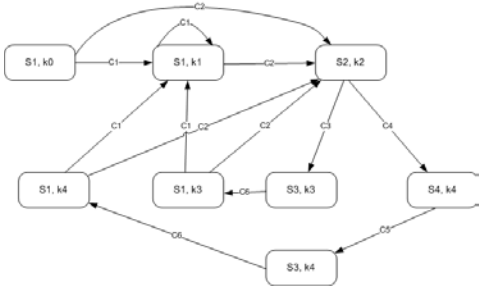


- A CASE tool for software fault tree analysis.
- Automatically generate and analyze software fault tree for an NuSCR output value.
- Combined with NuSRS 2.1(A tool supports NuSCR).

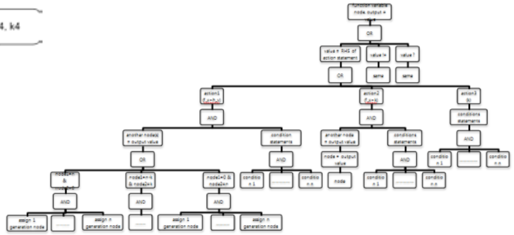
Automatic SFTA from NuSCR



NuSCR



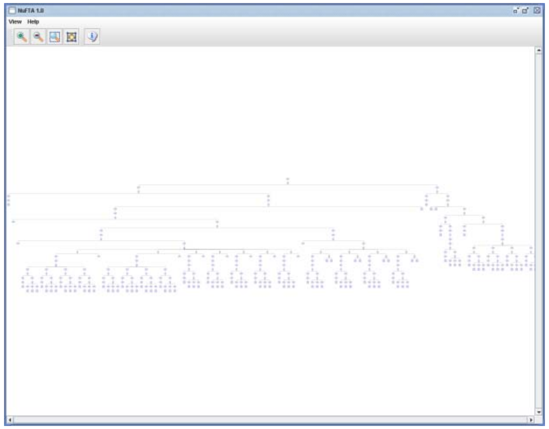
Expansion(FSM, TTS)



Templates for each model

Automatic Generation

- Backward analyze a failure using templates for each NuSCR model.

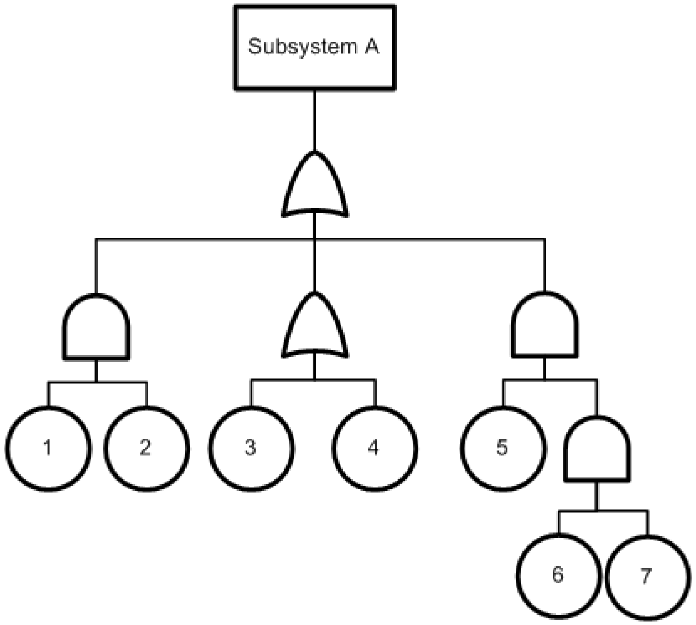


Mechanically generated SFT

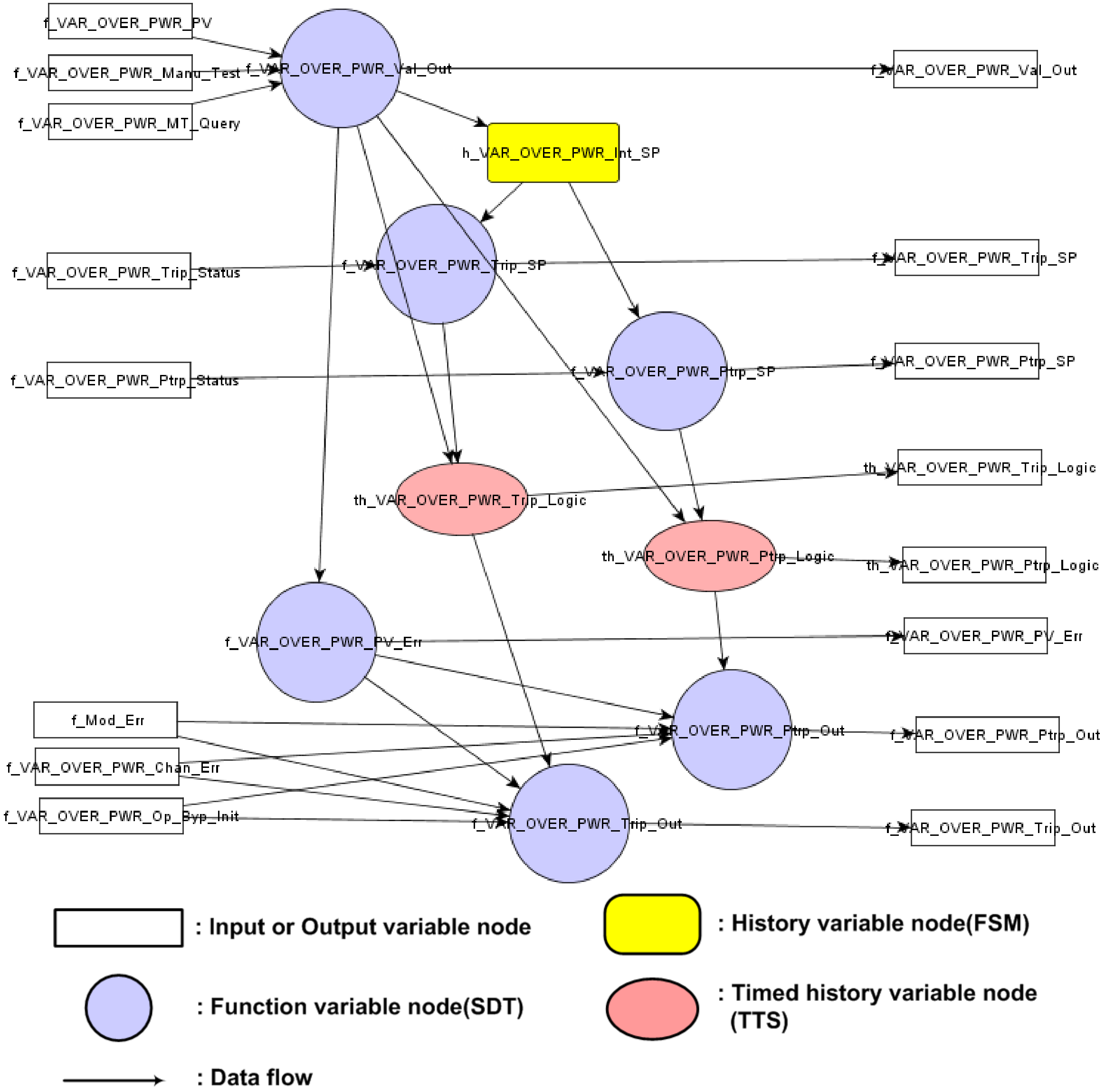
Software Fault Tree Analysis

- Manually construct a fault tree and analyze with
- Quality of FTA is depends on expert's knowledge and experience.

- Concernment : Software
  - No ware-out failure
  - Developer's logic



NuSCR

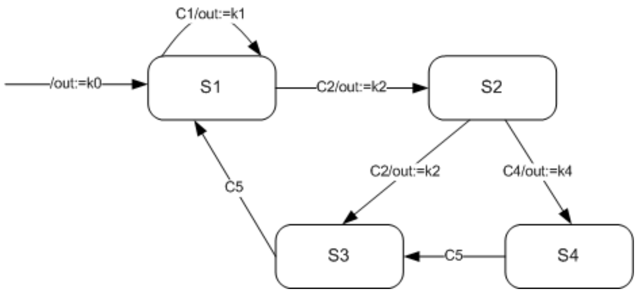


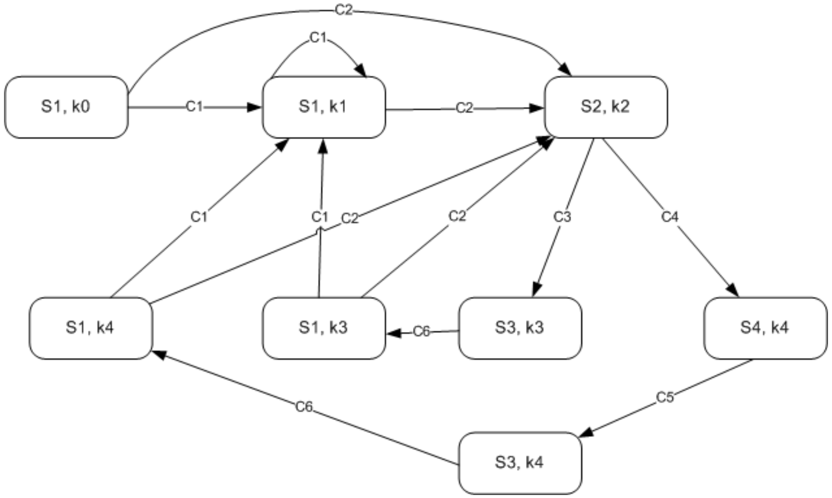
- Formal requirement specification
- Modified SCR for Reactor Protection System
- Three models
  - SDT
  - FSM
  - TTS

Expansion of FSM and TTS

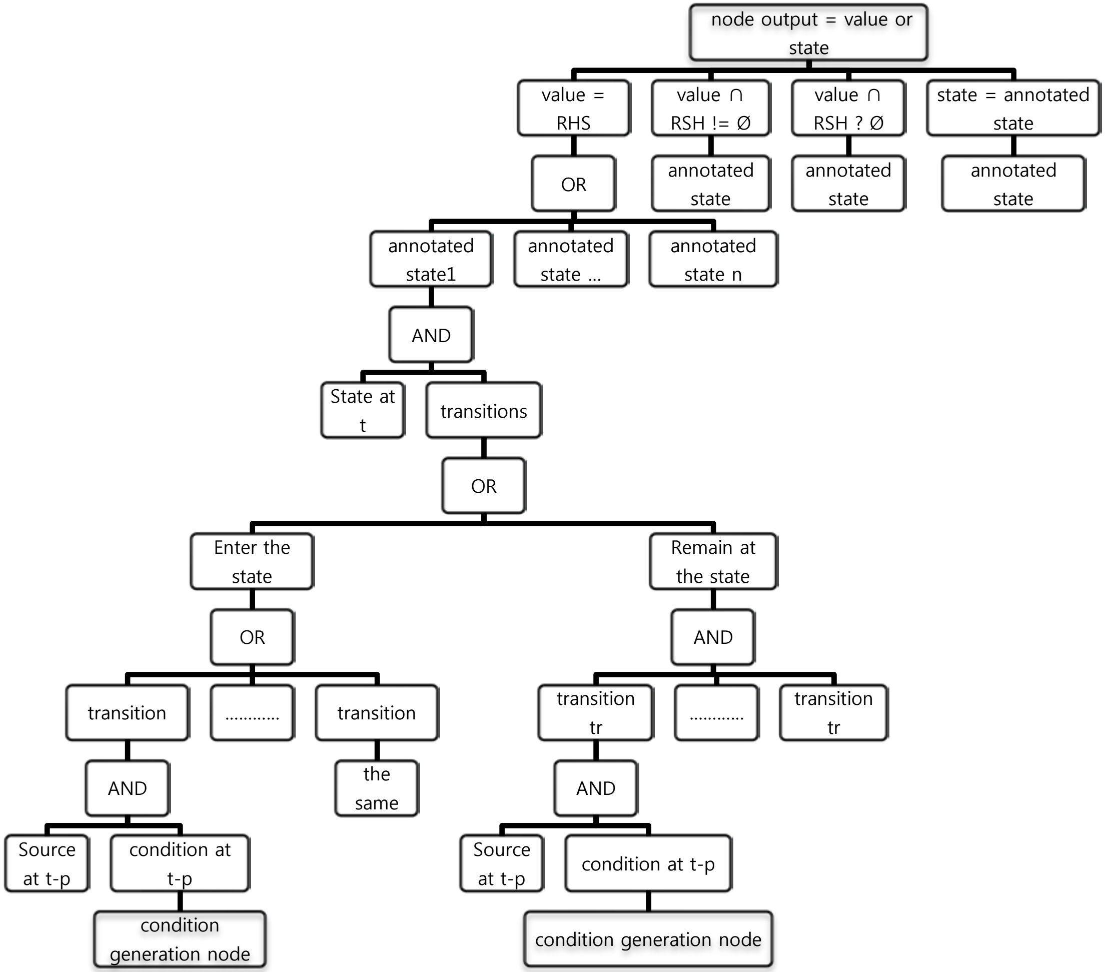
- FSM and TTS have states whose output value selected by previous state's output value and ingoing transition's assignment.
  - It is difficult analyze one state's total output value.

- Our solution : Annotated FSM and TTS
  - One state has previous state's name and output value
  - Reordered transitions which present new states' relation





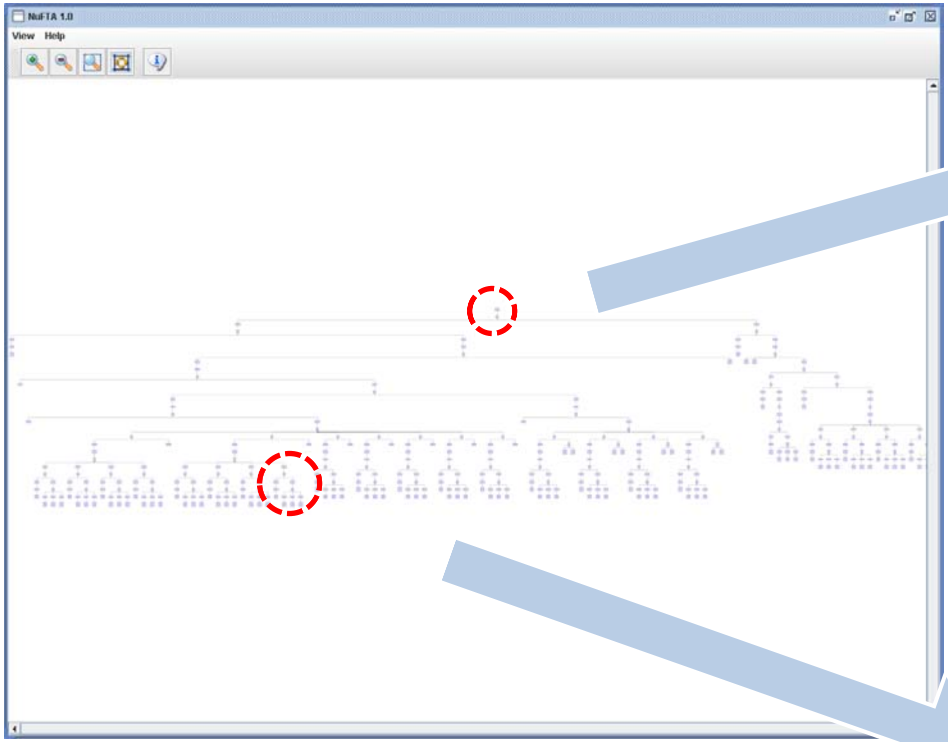
Templates



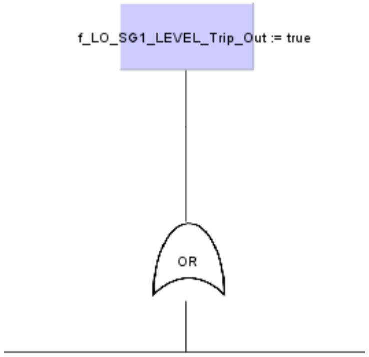
A template for FSM

- We use templates for mechanical generation of software fault tree.
- Templates support NuSCR models, respectively.

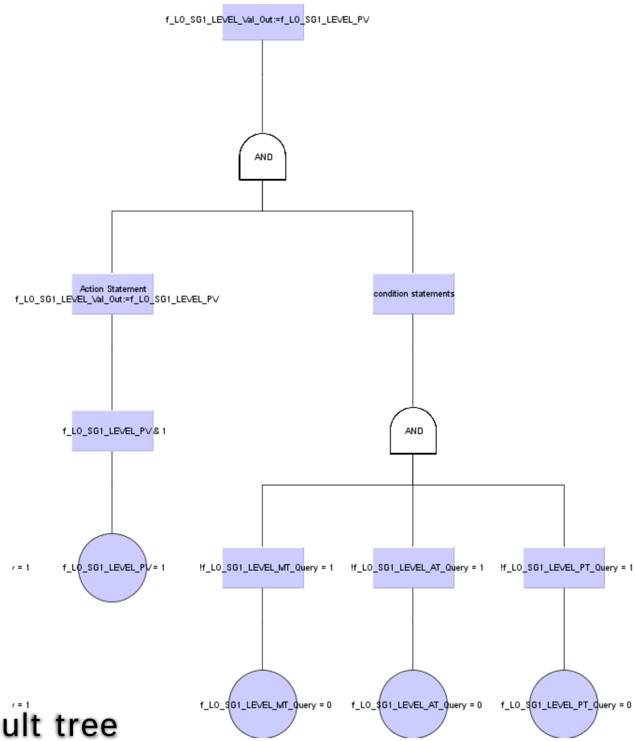
Generated SFT



A full software fault tree



Root node(failure)


f_LO_SG1_LEVEL_Val_Out:=f_LO_SG1_LEVEL_PV
AND
Action Statement
f_LO_SG1_LEVEL_Val_Out:=f_LO_SG1_LEVEL_PV
condition statements
f_LO_SG1_LEVEL_PV & 1
AND
f_LO_SG1_LEVEL_PV = 1
!f_LO_SG1_LEVEL_MT_Query = 1
!f_LO_SG1_LEVEL_AT_Query = 1
!f_LO_SG1_LEVEL_PT_Query = 1
f_LO_SG1_LEVEL_MT_Query = 0
f_LO_SG1_LEVEL_AT_Query = 0
f_LO_SG1_LEVEL_PT_Query = 0

Sub fault tree

Conclusion & Further work

- We propose a CASE tool which automatically analyze software fault tree from NuSCR formal requirement specification.

- For backward analyze output value's cause, we should consider all system's state.
  - It is difficult analyze a system's requirement specification which have large value.
  - NuFTA also need many time for that.

- We will interpret software fault tree to logical formula for use the result of analysis.