

MIM 2013
Saint Petersburg, Russia
2013.06.19 ~ 06.21



A Preliminary Report on Static Analysis of C Code for Nuclear Reactor Protection System

Jong-Hoon Lee , Eui-Sub Kim , Junbeom Yoo
KONKUK University, South Korea

Jang-Soo Lee

KAERI (Korea Atomic Energy Research Institute)

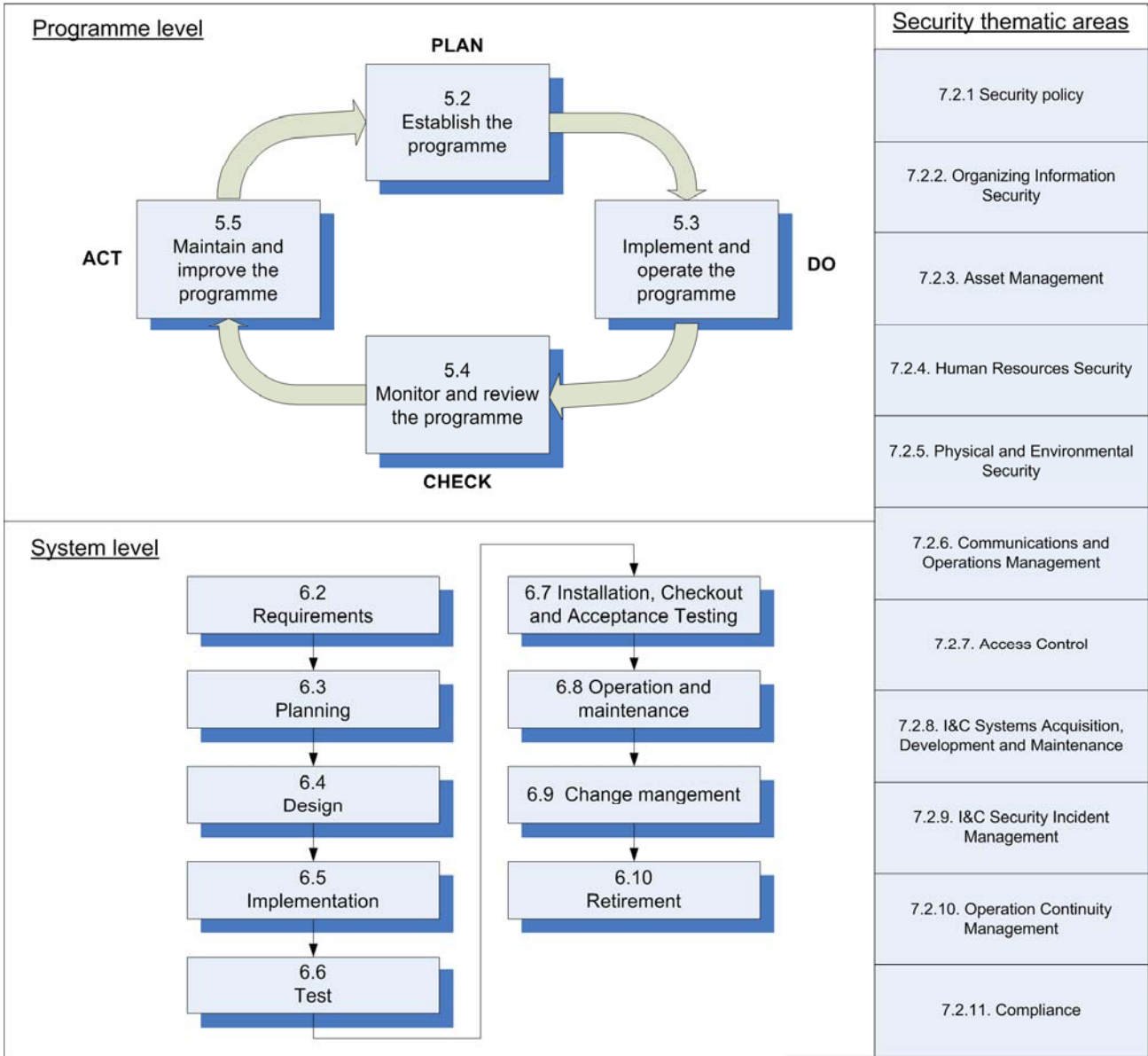
Cyber Security Standards for NPPs

<p>US NRC RG 5.71</p>	<p>Cyber Security Programs for Nuclear Facilities</p>
<p>IEC 61513</p>	<p>Nuclear Power Plants - I&C important to safety - General requirements for systems</p>
<p>IEC 60880</p>	<p>Nuclear Power Plants - I&C systems important to safety - Software aspects for computer-based systems performing category A functions</p>
<p>IEC 62645 (CD2)</p>	<p>Nuclear Power Plants - I&C systems - Requirements for security programmes for computer-based systems</p>

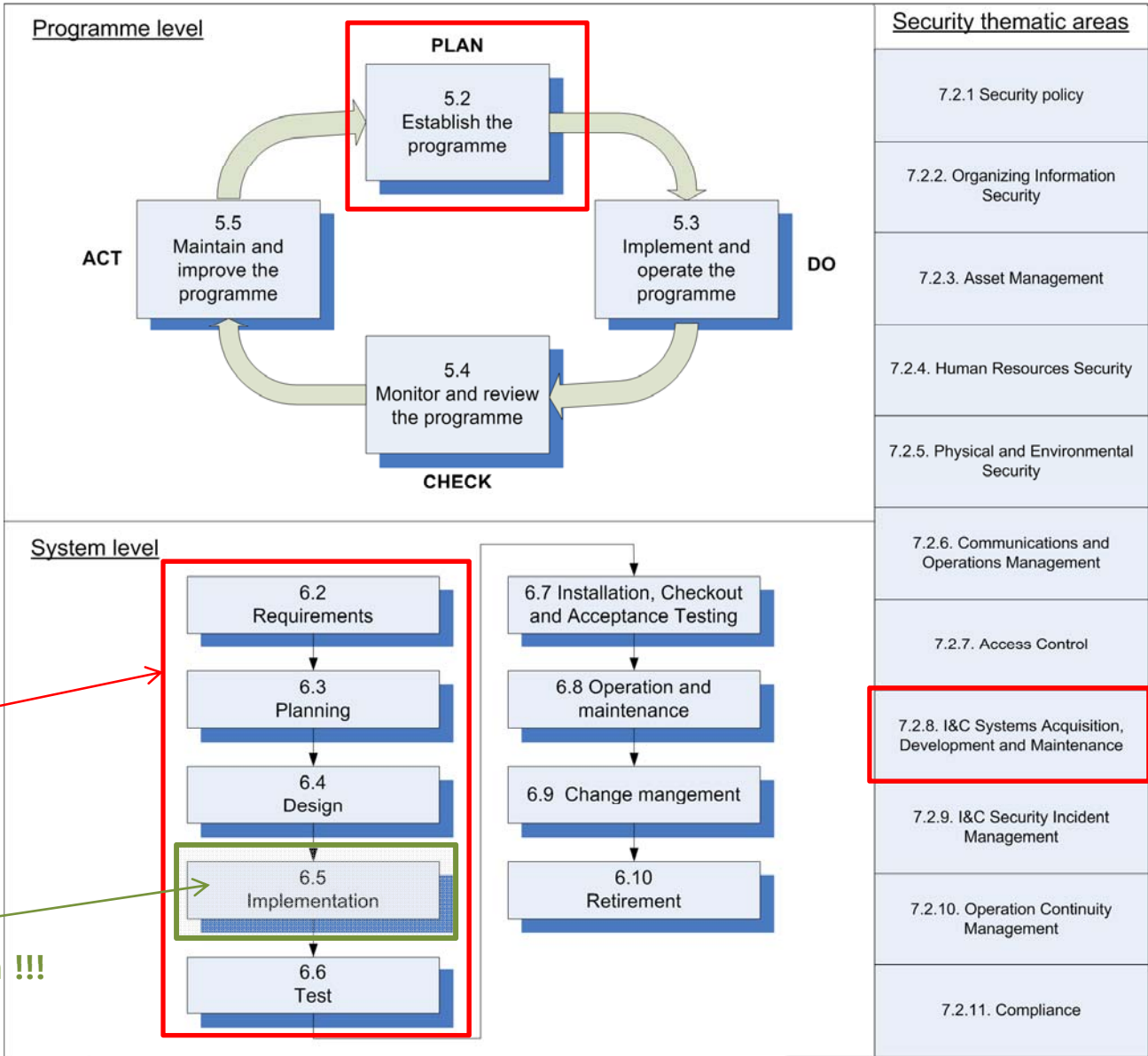
US NRC RG 5.71

Designers and developers for I&C systems shall have established and verified secure development methodologies in place throughout the development lifecycle of a system.

IEC 62645 (CD2)



IEC 62645 (CD2)



Our interest !!

This paper focuses on !!!

Secure Software Development Methodologies

SAFECode

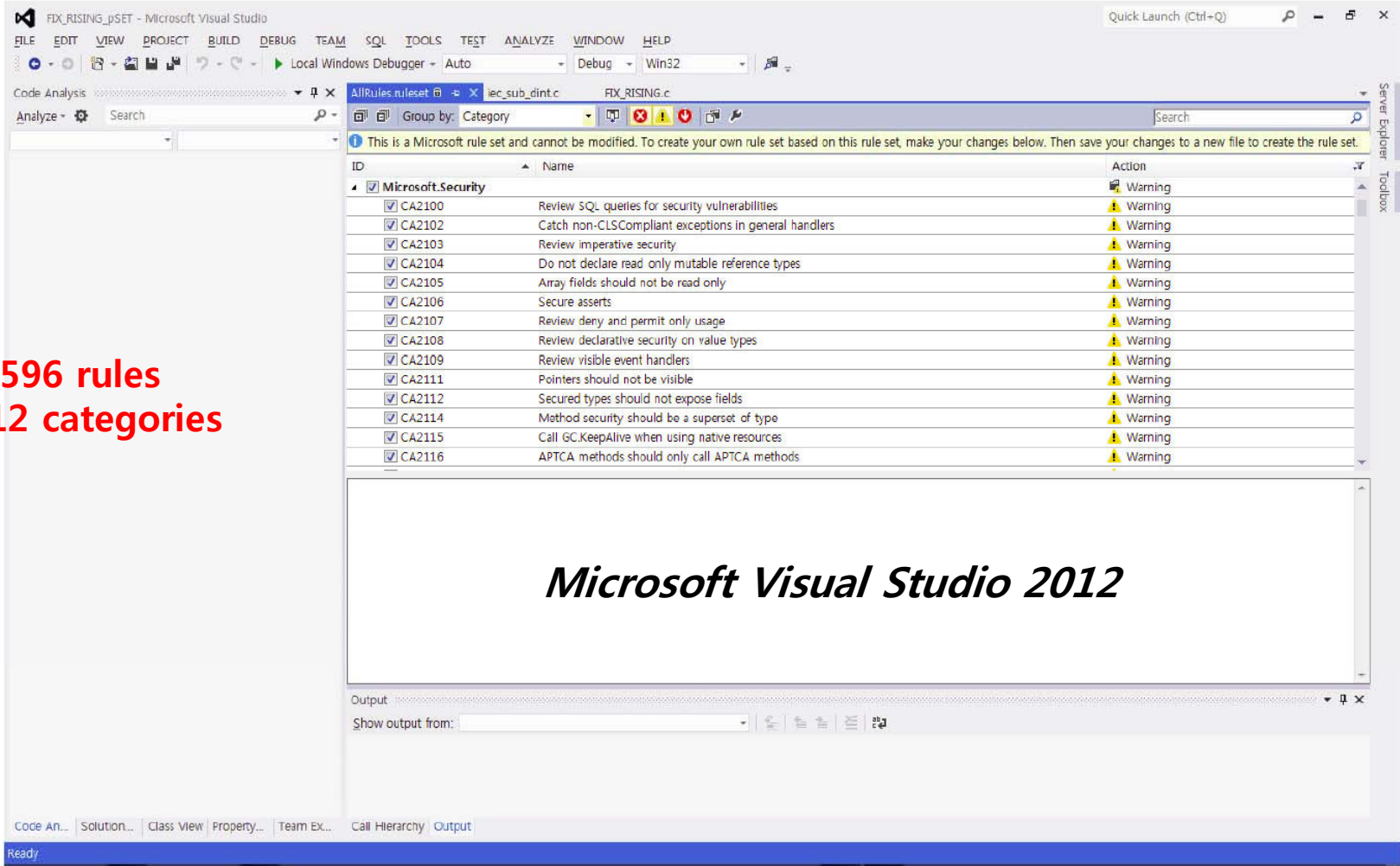
Widely-accepted practices should be followed throughout programming.

Use of **static** and dynamic **analysis code analysis tools** is highly recommended.

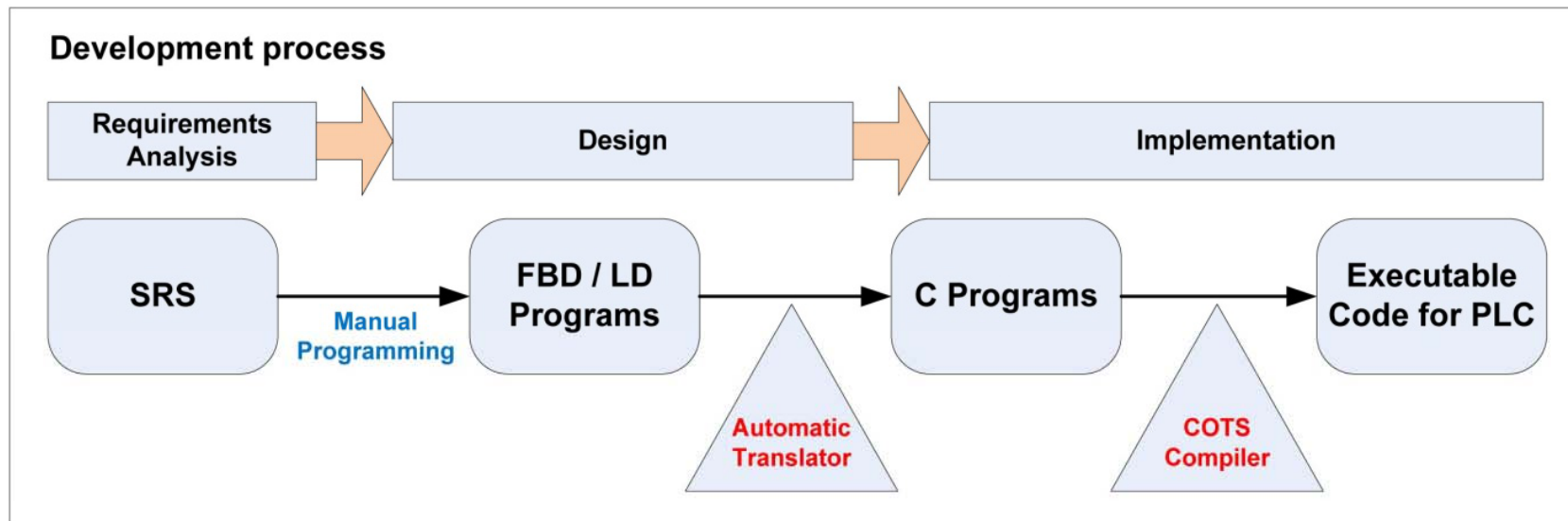
in the implementation phase

C Code Analysis Tool

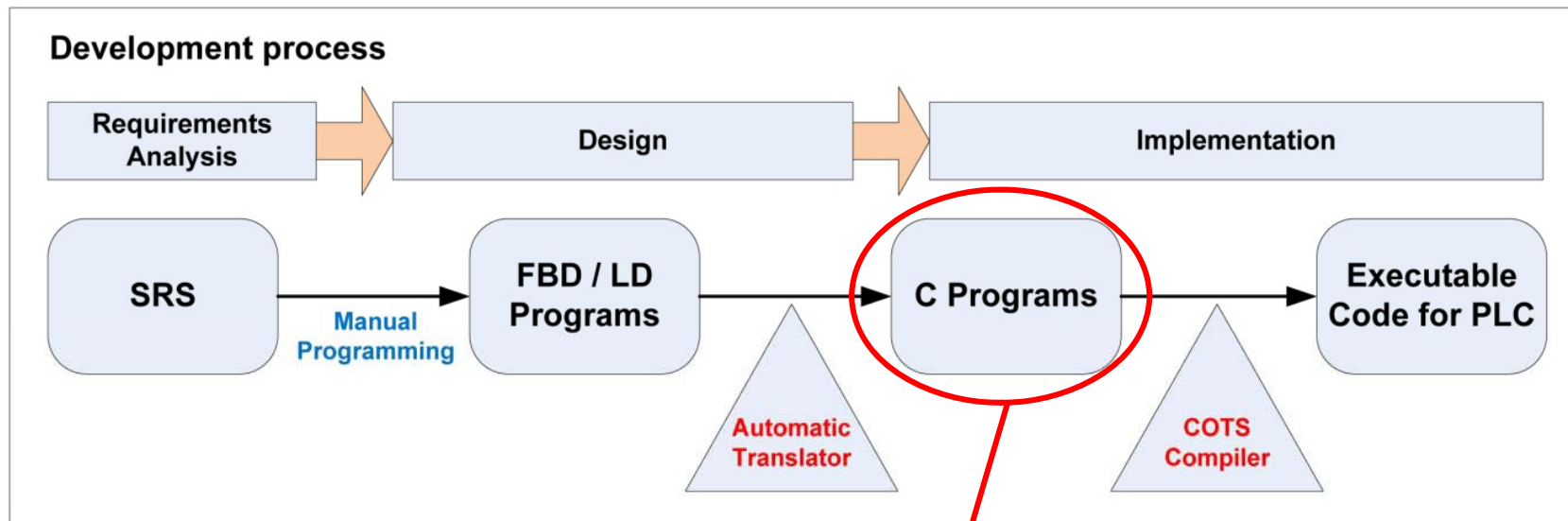
596 rules
of 12 categories



A Typical RPS SW Development Process



A Typical RPS SW Development Process



Our target of static code analysis

- A preliminary version of KNICS APR-1400 RPS BP
- PLC : *POSAFE-Q* PLC of POSCO ICT
- SW engineering tool : *pSET*
- Shutdown logic : fixed-setpoint rising trip

Static Analysis Result

C6001 Using uninitialized memory
Using uninitialized memory 'tmp':

Line	Explanation
89	'tmp' is not initialized
113	'tmp' is used, but may not have been initialized

More information
fix_rising.c (Line 113)
Warning

- C6281 Bitwise relation precedence
iec_sub_dint.c (Line 11)
- C6281 Bitwise relation precedence
iec_sub_dint.c (Line 11)
- C6281 Bitwise relation precedence
iec_sub_dint.c (Line 19)
- C6281 Bitwise relation precedence
iec_sub_dint.c (Line 19)

```

96  if(a_-->EN) {
97
98  GE_DINT(TRUE, a_-->PV_OUT, a_-->PTSP, &(ENO_1_FIX_RISING), &(OUT_1_FIX_RISING));
99
100 AND2_BOOL(TRUE, OUT_1_FIX_RISING, !(a_-->PTRIP_LOGIC), &(ENO_2_FIX_RISING), &(OUT_2_FIX_RISING));
101
102 ADD2_DINT(TRUE, a_-->PTRIP_CNT, 1, &(ENO_3_FIX_RISING), &(OUT_3_FIX_RISING));
103
104 SUB_DINT(TRUE, a_-->PTSP, a_-->PHYS, &(ENO_4_FIX_RISING), &(OUT_4_FIX_RISING));
105
106 SEL_DINT(TRUE, OUT_2_FIX_RISING, 0, OUT_3_FIX_RISING, &(ENO_5_FIX_RISING), &(OUT_5_FIX_RISING));
107
108 if(ENO_5_FIX_RISING) {
109     __tmp[0].__tdint = OUT_5_FIX_RISING;
110     a_-->PTRIP_CNT = __tmp[0].__tdint;
111 }
112
113 GE_DINT(TRUE, __tmp[0].__tdint, a_-->MAXCNT, &(ENO_6_FIX_RISING), &(OUT_6_FIX_RISING));
114
115 if(ENO_6_FIX_RISING) {
116     __tmp[1].__tbool = OUT_6_FIX_RISING;
117 }
118
119 SEL_BOOL(TRUE, __tmp[1].__tbool, a_-->PTRIP_LOGIC, TRUE, &(ENO_7_FIX_RISING), &(OUT_7_FIX_RISING));
120
121 if(ENO_7_FIX_RISING) {
122     __tmp[2].__tbool = OUT_7_FIX_RISING;
123 }
124
125 SEL_DINT(TRUE, __tmp[1].__tbool, a_-->PTSP, OUT_4_FIX_RISING, &(ENO_8_FIX_RISING), &(OUT_8_FIX_RISING));
126
127 if(ENO_8_FIX_RISING) {
128     __tmp[3].__tdint = OUT_8_FIX_RISING;
129 }
130
    
```

Output
Show output from: Build
iec_sub_dint.c(11): warning : C6281: Incorrect order of operations: relational operators have higher precedence than bitwise operators.
iec_sub_dint.c(19): warning : C6281: Incorrect order of operations: relational operators have higher precedence than bitwise operators.
=====
Rebuild All: 1 succeeded, 0 failed, 0 skipped

5 critical errors !!!

C6001 Using uninitialized memory

All Projects (5) All Results (5)

C6001 Using uninitialized memory
Using uninitialized memory '__tmp'.

Line	Explanation
89	'__tmp' is not initialized
113	'__tmp' is used, but may not have been initialized

[More information](#)

fix_rising.c (Line 113)
Warning Actions ▾

C6281 Bitwise relation precedence
iec_sub_dint.c (Line 11)

C6281 Bitwise relation precedence
iec_sub_dint.c (Line 11)

C6281 Bitwise relation precedence
iec_sub_dint.c (Line 19)

C6281 Bitwise relation precedence
iec_sub_dint.c (Line 19)

C6281 Bitwise relation precedence

All Projects (5) All Results (5)

C6001 Using uninitialized memory
fix_rising.c (Line 113)

C6281 Bitwise relation precedence
Incorrect order of operations: relational operators have higher precedence than bitwise operators.
iec_sub_dint.c (Line 11)
Warning Actions ▾

C6281 Bitwise relation precedence
iec_sub_dint.c (Line 11)

C6281 Bitwise relation precedence
iec_sub_dint.c (Line 19)

C6281 Bitwise relation precedence
iec_sub_dint.c (Line 19)

We found functional correctness (safety) - related errors.

We found no security-related error!

Splint

**More than 100 errors !!!
- Not categorized**

```

$ splint FIX_RISING.c
Splint 3.1.2 --- 28 Mar 2013

psettype.h:4:22: Type BOOL is probably meant as a boolean
                    type, but the boolean
                                type name is not set. Use -booltype BO
                                OL to set it.
    Use the -booltype, -boolfalse and -booltrue flags to cha
    nge the name of the
    default boolean type. (Use -likelybool to inhibit warnin
    g)
FIX_RISING.c: (in function FIX_RISING_)
FIX_RISING.c:96:5: Test expression for if not boolean, typ
e BOOL: a_->EN
    Test expression type is not boolean or int. (Use -predbo
    olint to inhibit
    warning)
FIX_RISING.c:98:2: Return value (type BOOL) ignored: GE_DI
NT(TRUE, a_...
    Result returned by function call is not used. If this is
    intended, can cost
    result to (void) to eliminate message. (Use -retvalother
    to inhibit warning)

    ...

FIX_RISING.c:61:6: Variable exported but not used outside
FIX_RISING:
                    OUT__24__FIX_RISING
FIX_RISING.c:62:6: Variable exported but not used outside
FIX_RISING:
                    END__25__FIX_RISING
FIX_RISING.c:63:6: Variable exported but not used outside
FIX_RISING:
                    OUT__25__FIX_RISING
FIX_RISING.c:64:6: Variable exported but not used outside
FIX_RISING:
                    END__26__FIX_RISING
FIX_RISING.c:65:6: Variable exported but not used outside
FIX_RISING:
                    OUT__26__FIX_RISING

Finished checking --- 103 code warnings

```

Lesson Learned

Find appropriate static code analysis tools!!

A number of tools are available

Different rules and categories

Lesson Learned

Consider OS and HW as well as safety/security rules!!

Most of tools assume the use of MS Windows and Linux

But, the RPS uses PLCs not PCs

Operating systems and HWs are different

Lesson Learned

Develop a secure development process from requirements!!

A systematic process is required

THANK YOU!!!
and Questions?

Jang-Soo Lee

Principal Researcher / Ph.D

KAERI (Korea Atomic Energy Research Institute)