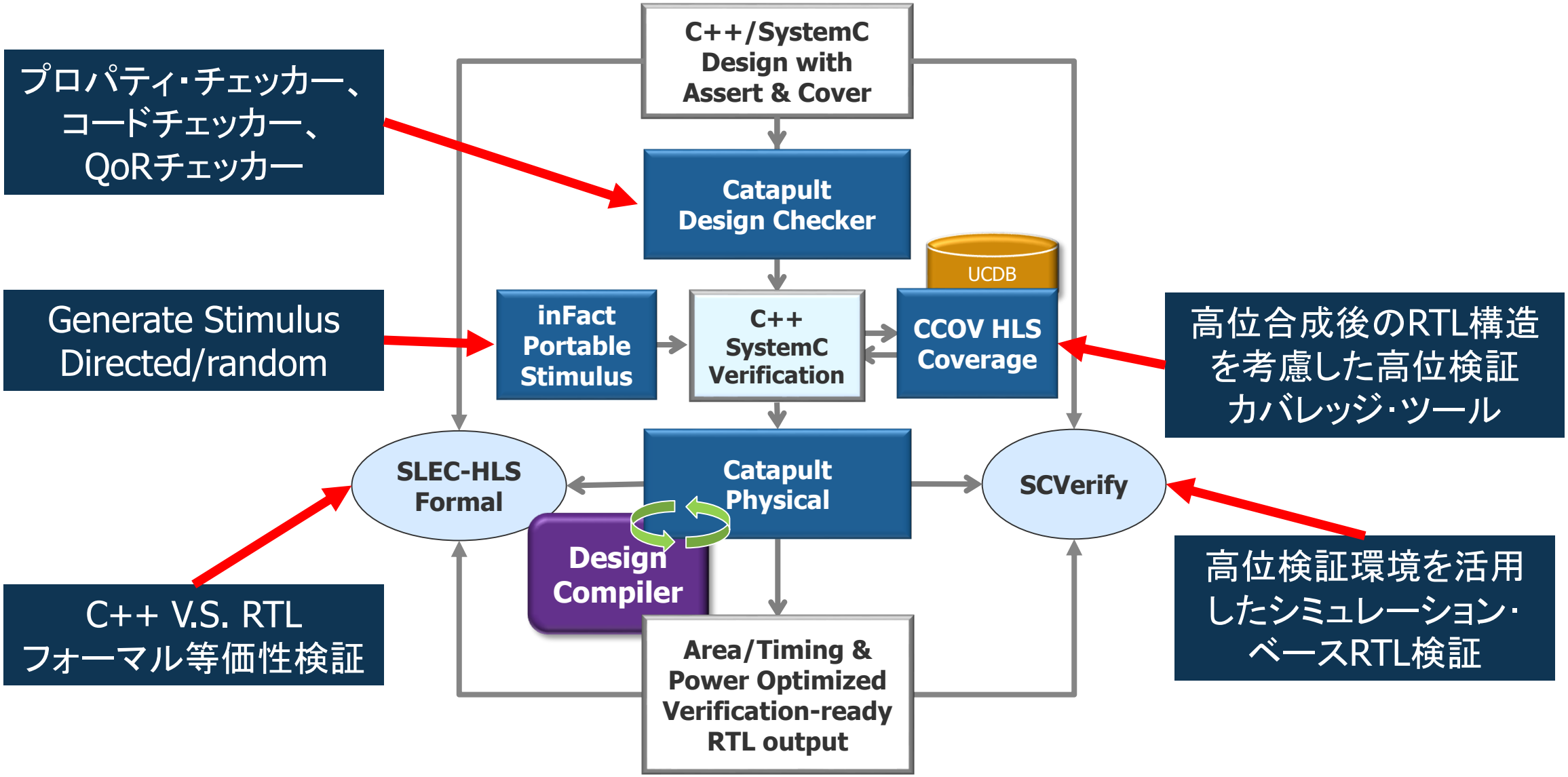
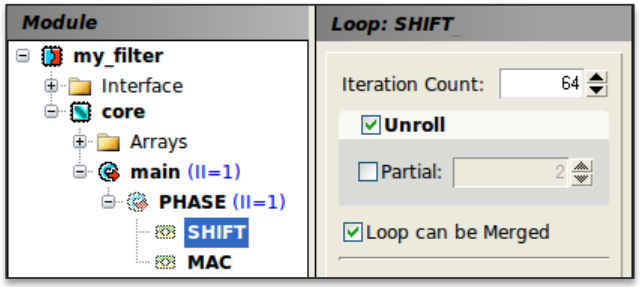

プロパティ・チェッカー、
コードチェッカー、
QoRチェッカー
Generate Stimulus
Directed/random
C++ V.S. RTL
フォーマル等価性検証
C++/SystemC
Design with
Assert & Cover
Catapult
Design Checker
inFact
Portable
Stimulus
C++
SystemC
Verification
UCDB
CCOV HLS
Coverage
高位合成後のRTL構造
を考慮した高位検証
カバレッジ・ツール
SLEC-HLS
Formal
Catapult
Physical
Design
Compiler
SCVerify
高位検証環境を活用
したシミュレーション・
ベースRTL検証
Area/Timing &
Power Optimized
Verification-ready
RTL output

```
int mac(
  char data[N],
  char coef[N]
) {
  int accum=0;
  for (int i=0; i<N; i++)
    accum += data[i] * coef[i];
  return accum;
}
```

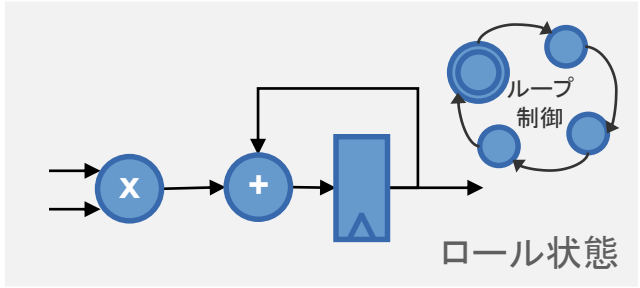
C/C++, SystemC に対応

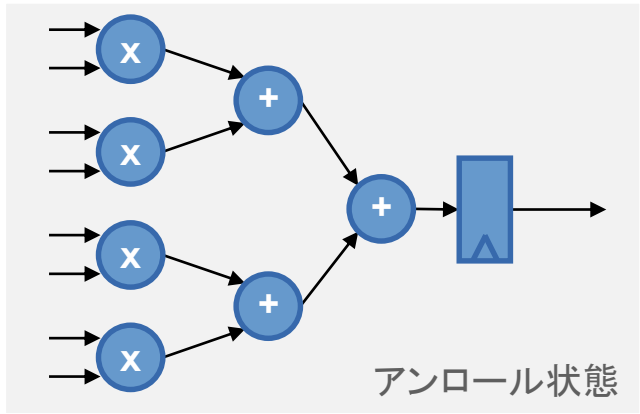
高位モデル

並列性の制御

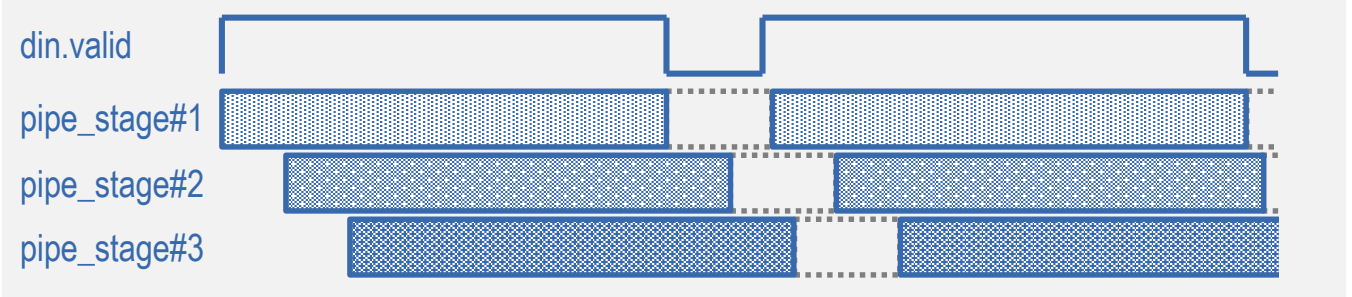


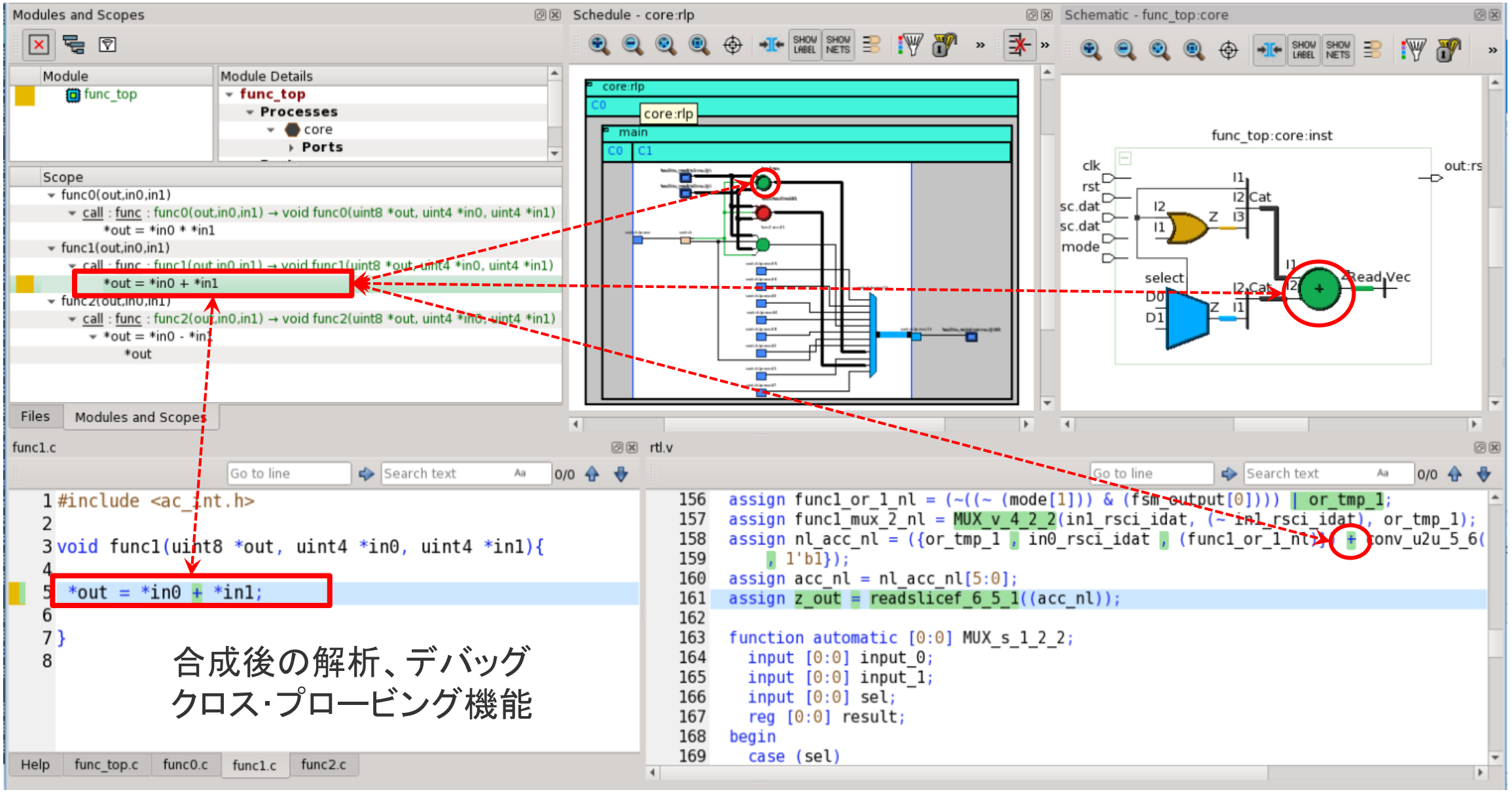
並列展開





パイプライン並列化




Modules and Scopes
Module
func_top
Module Details
func_top
Processes
core
Ports
Scope
func0(out,in0,in1)
call : func : func0(out,in0,in1) → void func0(uint8 *out, uint4 *in0, uint4 *in1)
*out = *in0 * *in1
func1(out,in0,in1)
*out = *in0 + *in1
func2(out,in0,in1)
*out = *in0 - *in1
*out
Files
Modules and Scopes
Schedule - core:rlp
core:rlp
C0
main
C0 C1
Schematic - func_top:core
func_top:core:inst
clk
rst
sc.dat
sc.dat
mode
select
D0
D1
I1
I2
Cat
Z
I3
out:rs
Read_Vec
func1.c
Go to line
Search text
0/0
1 #include <ac_int.h>
2
3 void func1(uint8 *out, uint4 *in0, uint4 *in1){
4
5 *out = *in0 + *in1;
6
7 }
8
Help
func_top.c
func0.c
func1.c
func2.c
rtl.v
156 assign func1_or_1_nl = (~((~ (mode[1])) & (fsm_output[0]))) | or_tmp_1;
157 assign func1_mux_2_nl = MUX_v_4_2_2(in1_rsci_idat, (~ in1_rsci_idat), or_tmp_1);
158 assign nl_acc_nl = ({or_tmp_1 , in0_rsci_idat , (func1_or_1_nl)}
159 , 1'b1});
160 assign acc_nl = nl_acc_nl[5:0];
161 assign z_out = readslicef_6_5_1((acc_nl));
162
163 function automatic [0:0] MUX_s_1_2_2;
164 input [0:0] input_0;
165 input [0:0] input_1;
166 input [0:0] sel;
167 reg [0:0] result;
168 begin
169 case (sel)
合成後の解析、デバッグ
クロス･プロービング機能