HITACHI
Inspire the Next

12th International Workshop on the application of FPGAs in NPPs
Oct 14-16, 2019
Budapest, Hungary

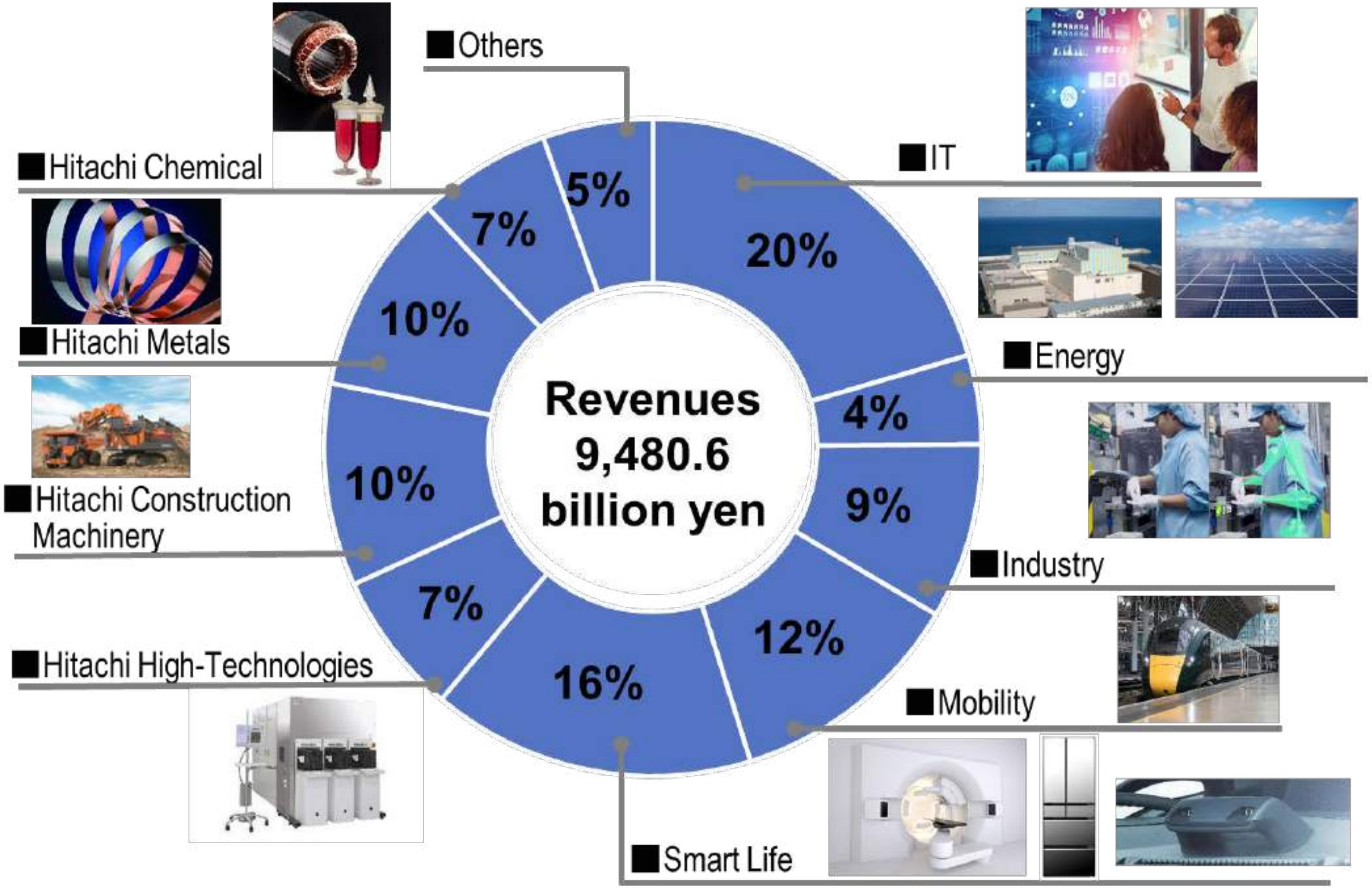
Class 1 Compliant Design and Verification Process for FPGA-Based I&C System

Oct 15, 2019

Satoshi Nishikawa, Junichi Kumagai, Takumi Uezono

Hitachi, Ltd.

Business Segment Constitution of Hitachi, Ltd. (FY2018)



HITATCHI
Inspire the Next

Contents

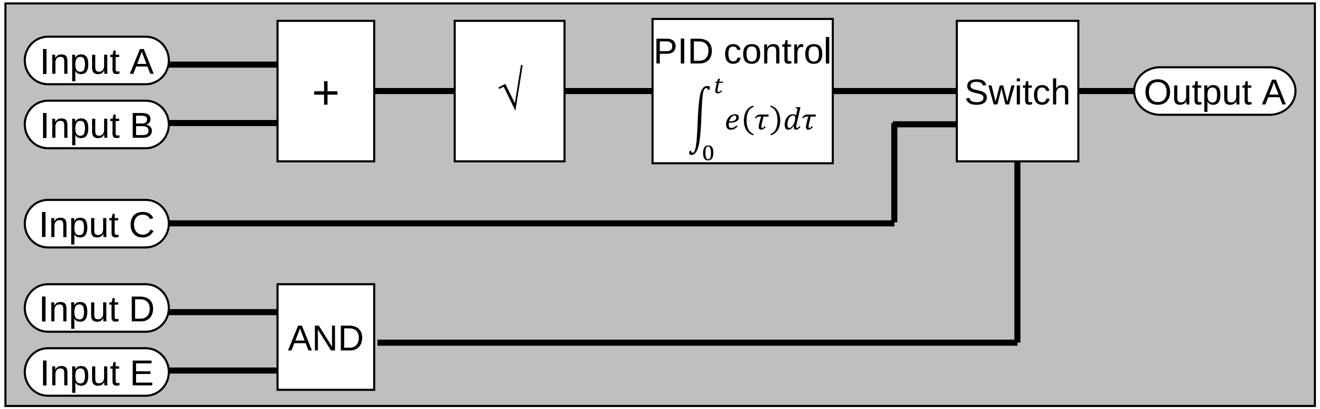
FPGA in Nuclear Power Plant Safety Application

HITACHI
Inspire the Next

- FPGA has many advantages
  - ➢ much simpler and less costly V&V process
  - ➢ resistance to cybersecurity threats
  - ➢ long term support by FPGA vendors
  - ➢ resilience to hardware obsolescence
- Flash-based FPGA vs SRAM-based FPGA
  - ➢ SRAM-based FPGA
    - highly integrated because area of SRAM cell is small
    - high-speed and low-power
    - vulnerable to noise such as radiation and it leads to soft error
  - ➢ Flash-based FPGA
    - cannot be highly integrated because are of Flash memory cell is large
    - generally limited speed and limited logical scale mounted on Flash-based FPGA
    - high resilience to radiation-induced soft error

Requirement for Nuclear Safety Applications

- Designed as Function Block Diagram (FBD)



- Some nuclear safety applications process a wide range of real number

  → Floating-point calculation is essential function for nuclear safety apps.

Floating-point Calculation Unit in Flash-based FPGA

HITACHI Inspire the Next

- Complexity of algorithm of floating-point calculation
  - ➢Difficult to verify
  - →solved by effectively applying formal and dynamic verification in combination[1]
- Consume large amount of FPGA resources
  - ➢Floating-point calculation units consume massive resources
  - ➢More than 900 calculation units are used to make an application
    - Application is implemented in FPGA as written in FBD in general FPGA development process
    - Flash-based FPGA has limited logical resources

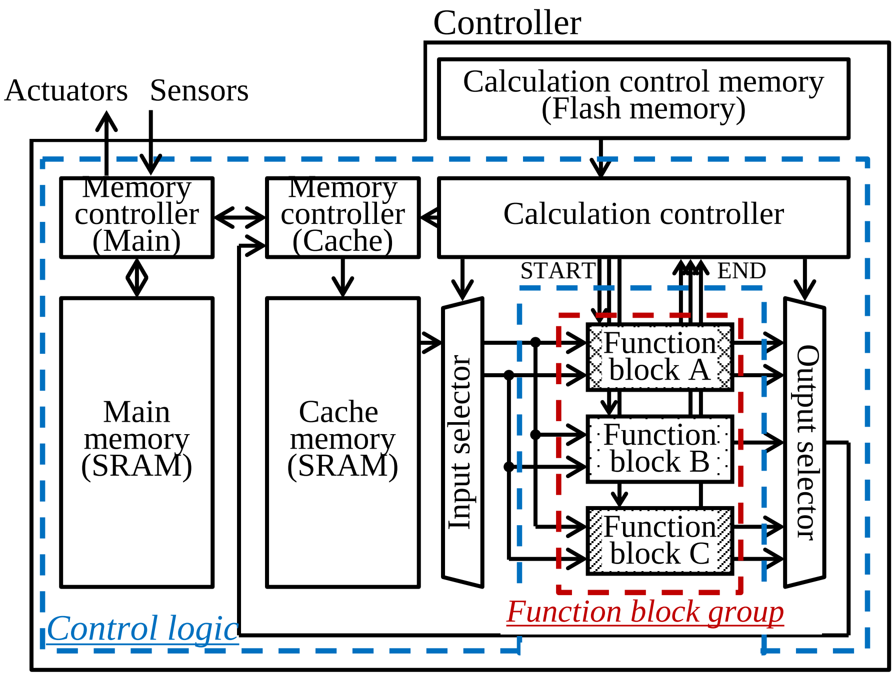
→ Area-efficient FPGA implementation method is required

[1] T Motoya, et.al, "Introduction of Class 1 FPGA Platform for the UK ABWR" 11th International Workshop on the application of FPGAs in NPPs

Resource Sharing Architecture for FPGA

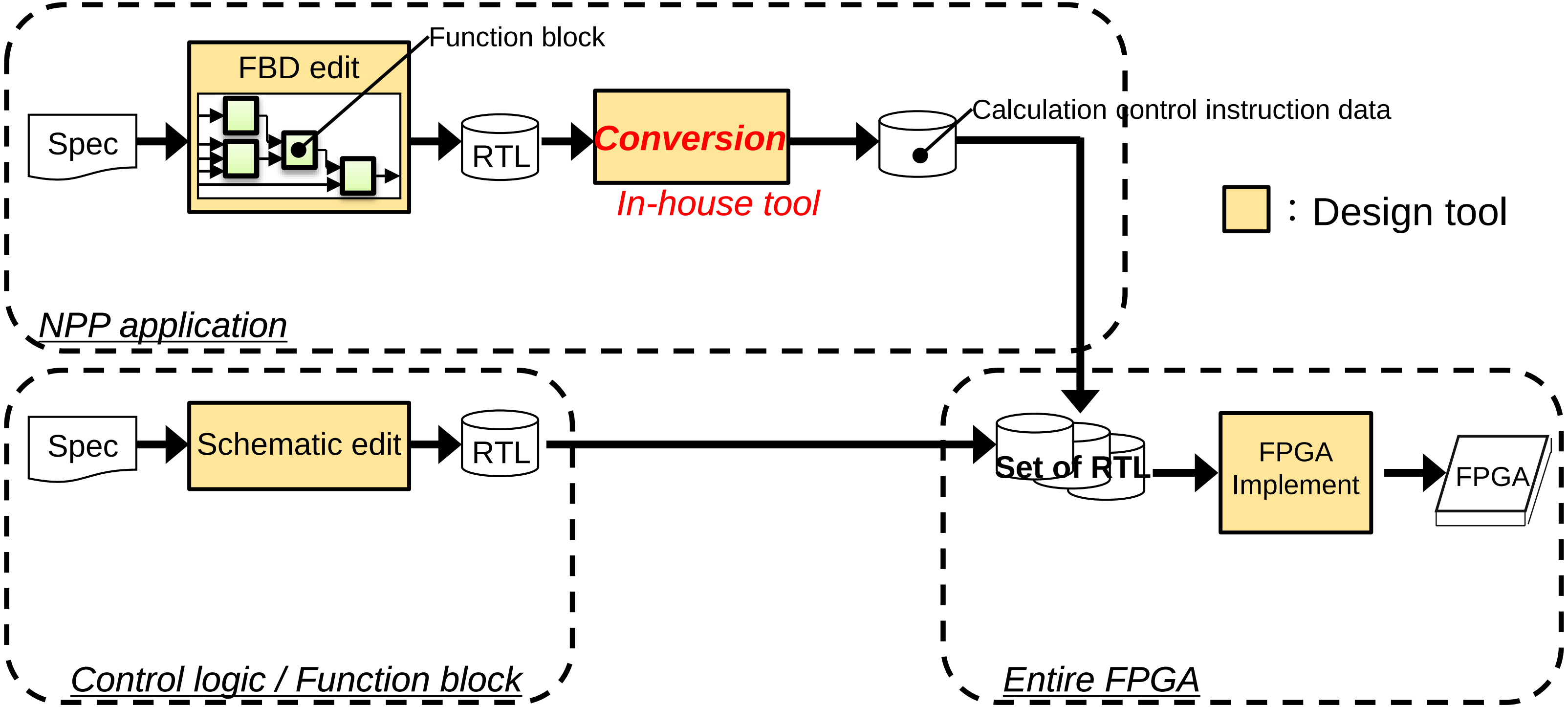
- Implement only one circuit for each function block in FBD application
  - Each function block operates based on time-sharing manner
  - Operate according to calculation control instruction in Flash memory generated from FBD application
  - Verify control logic by formal method

How to generate and verify calculation control instruction according to FBD apps.



Design Process for Resource Sharing Architecture in FPGA

HITACHI
Inspire the Next



FBD to Calculation Control Instruction Conversion

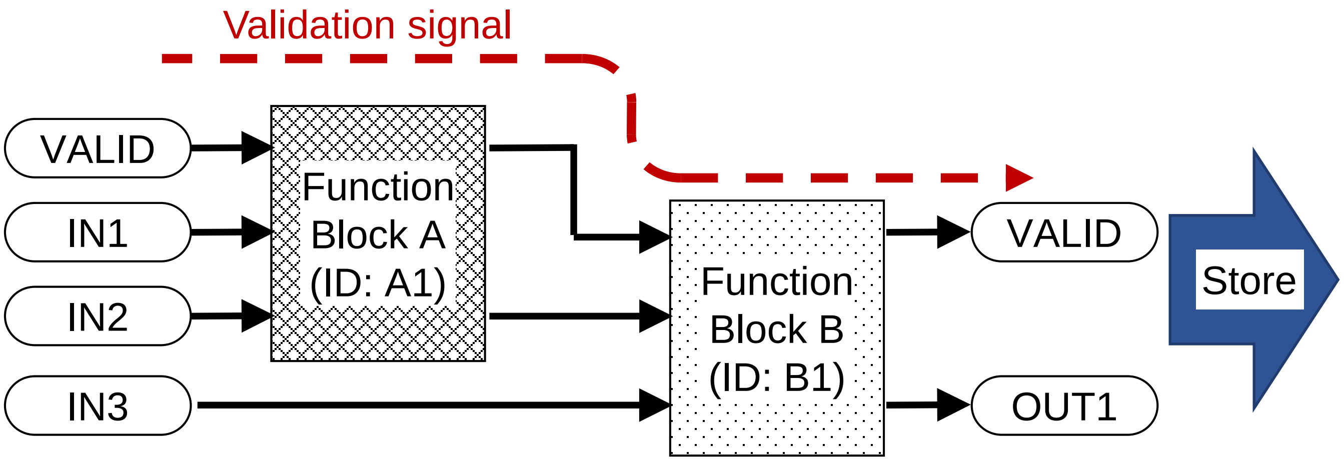
HITACHI
Inspire the Next

- Conversion steps
  1. Trace and collect the function block data
  2. Determine the cache memory addresses of the FBD input, output, intermediate signals
  3. Generate a calculation control instruction data which is stored in the flash memory

FBD to Calculation Control Instruction Conversion

Step 1: Trace and collect the function block data

- The conversion tool has to understand the calculation order of the function blocks
- Introduced the concept of "Validation signal"



Block ID	Calc. order	Block Type
A1	1	A
B1	2	B
C1	3	C
B2	4	B
…	…	…

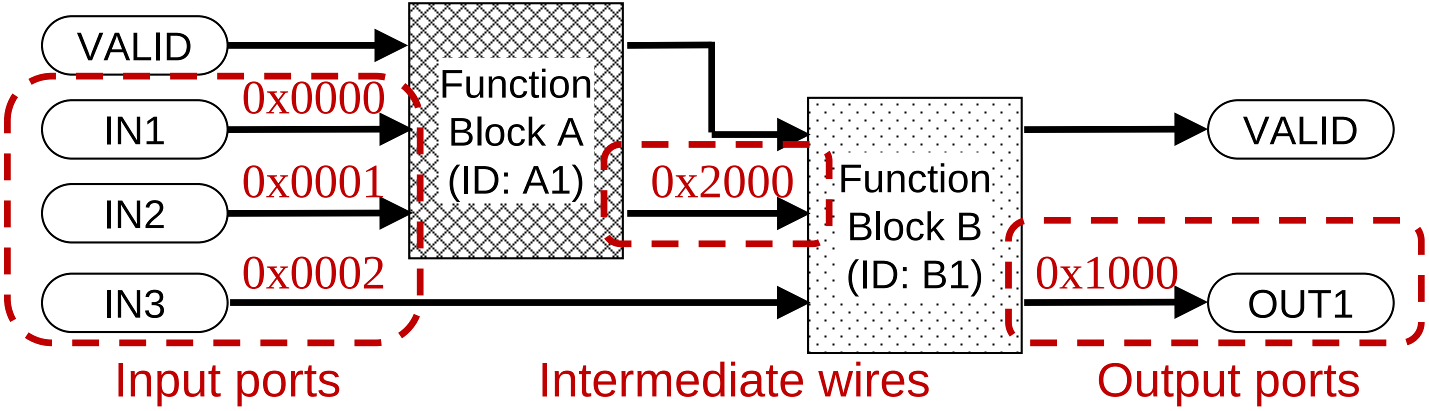
FBD to Calculation Control Instruction Conversion

Step 2: Determine the cache memory addresses of the FBD input, output, intermediate wires

- Address is assigned in address range determined for each type

Eg.

0x0000-0x0FFF for input ports, 0x1000-0x1FFF for output ports, 0x2000-0x2FFF for intermediate wires



FBD to Calculation Control Instruction Conversion

HITACHI
Inspire the Next

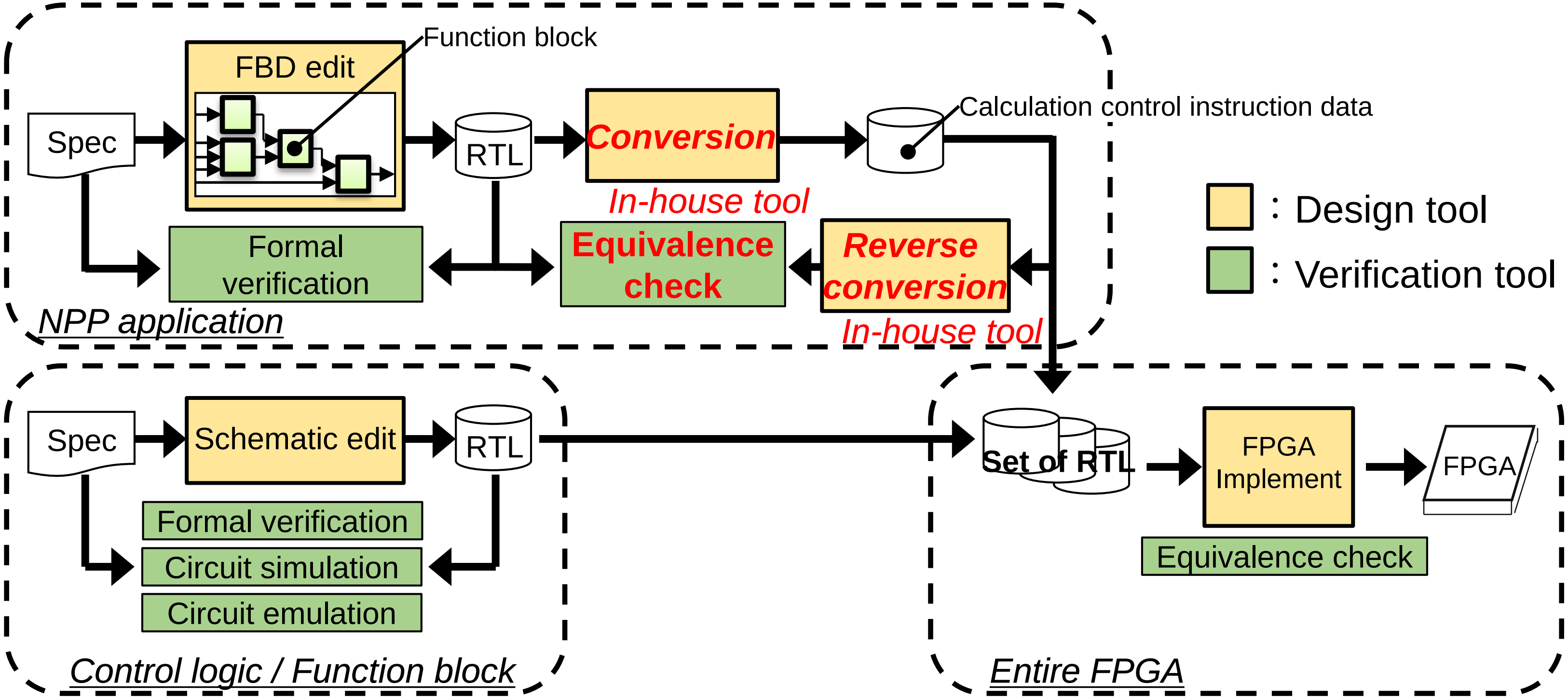
Step 3: Generate a calculation control instruction data which is stored in the flash memory

- Combine the data generated in Step 1 and Step 2

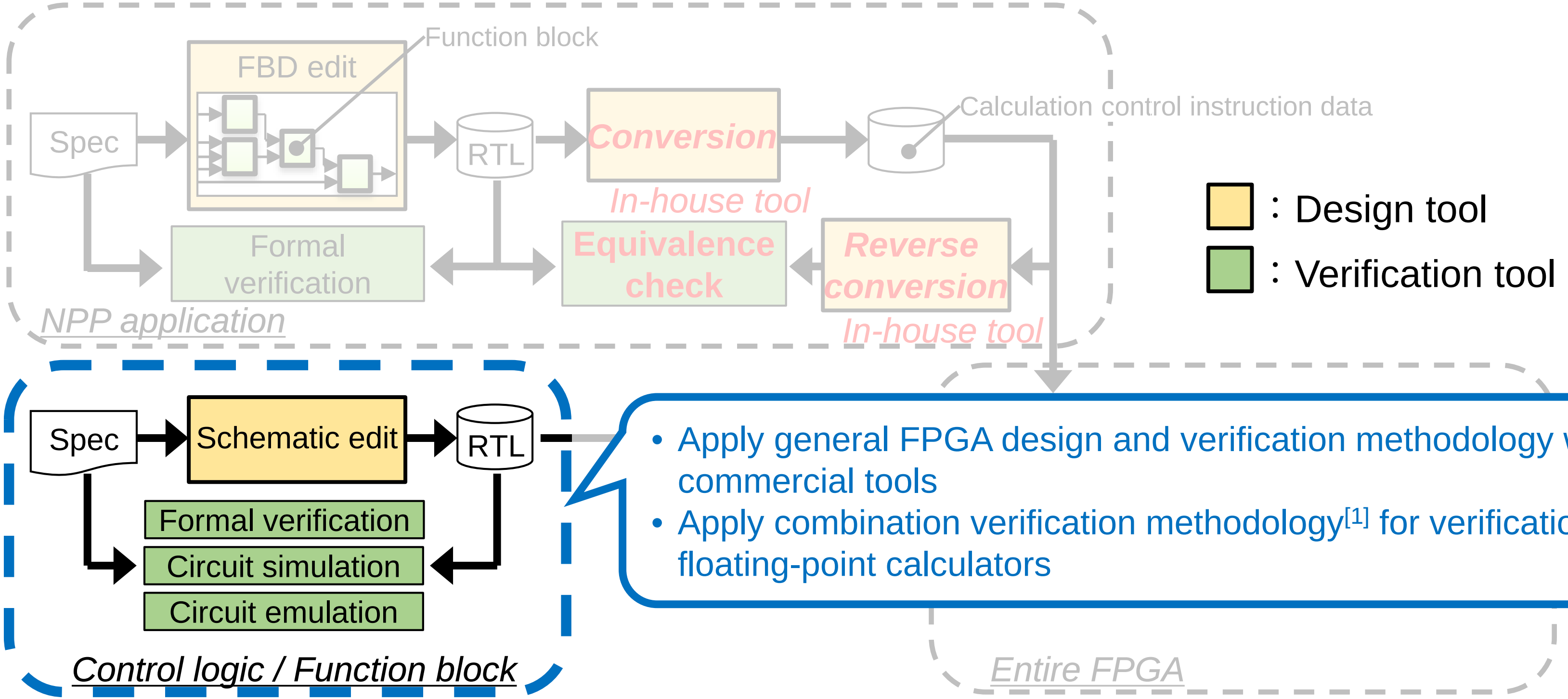
from Step 1			from Step 2	
Block ID	Calc. order	Block Type	Data address	
			Input	Output
A1	1	A	0x0001, 0x0002	0x2001, 0x2002
B1	2	B	0x2001, 0x0003	0x2003
C1	3	C	0x2003, 0x2002	0x1001, 0x2004
B2	4	B	0x2004, 0x0004	0x1002
…	…	…	…	…

Design and Verification Process for Resource Sharing Architecture in FPGA

HITACHI
Inspire the Next

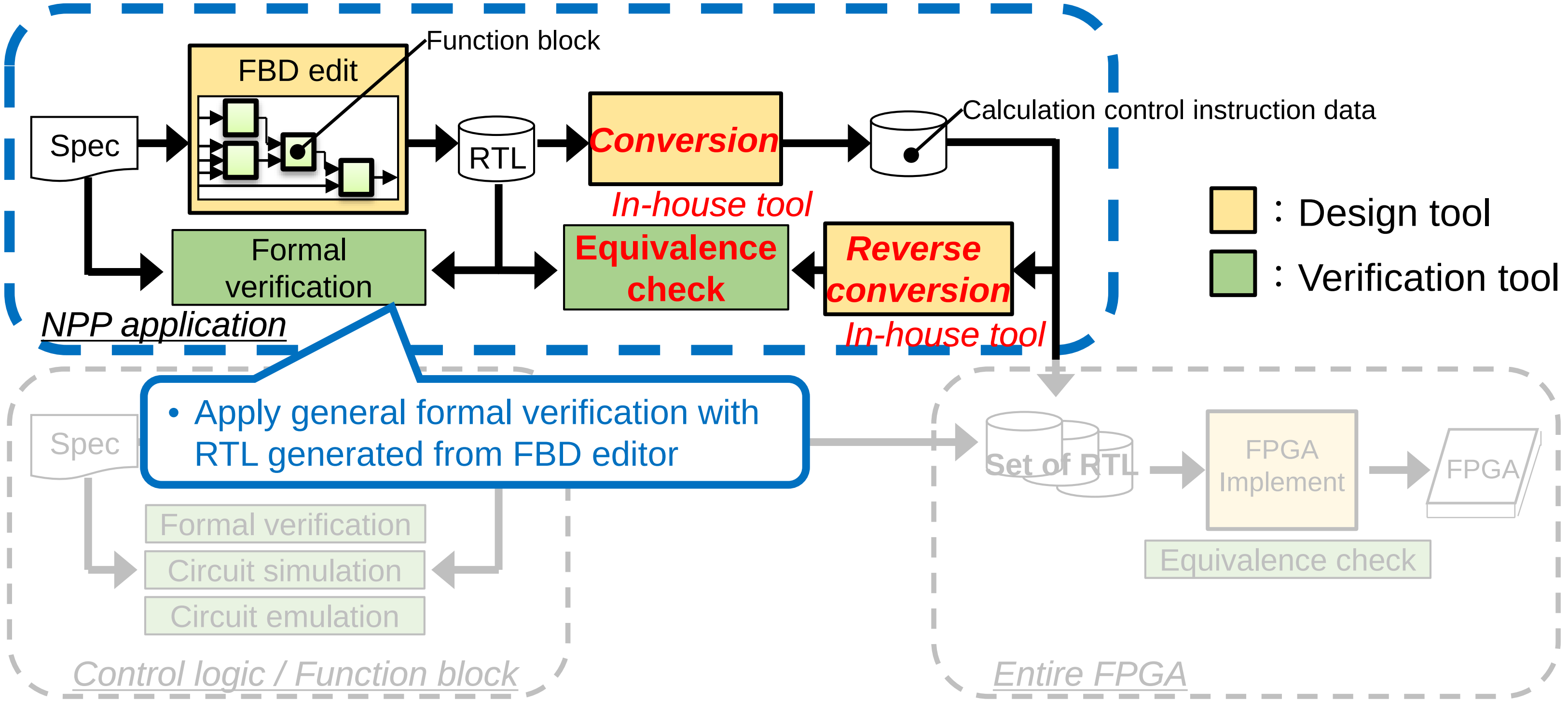


Design and Verification Process for Resource Sharing Architecture in FPGA



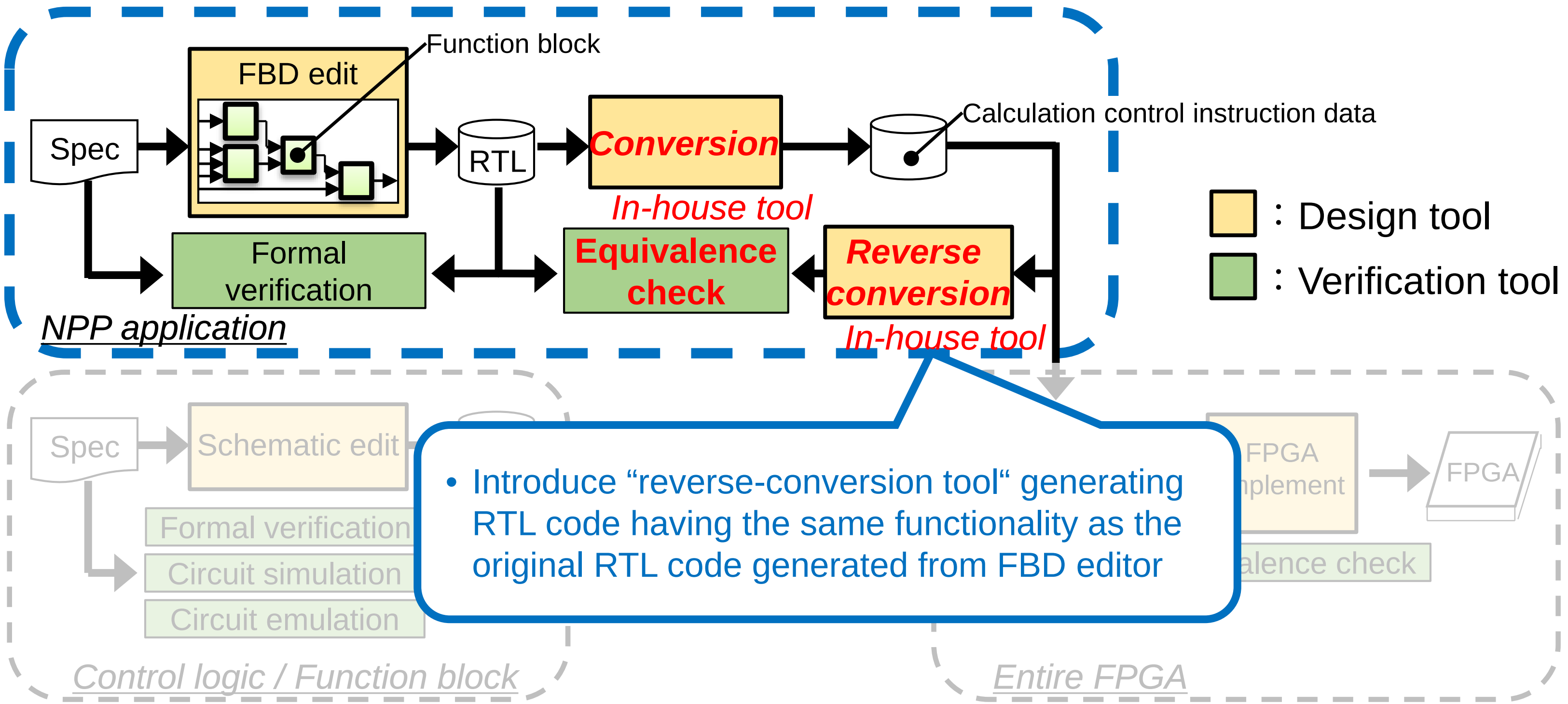
- Apply general FPGA design and verification methodology with commercial tools
- Apply combination verification methodology[1] for verification of floating-point calculators

Design and Verification Process for Resource Sharing Architecture in FPGA



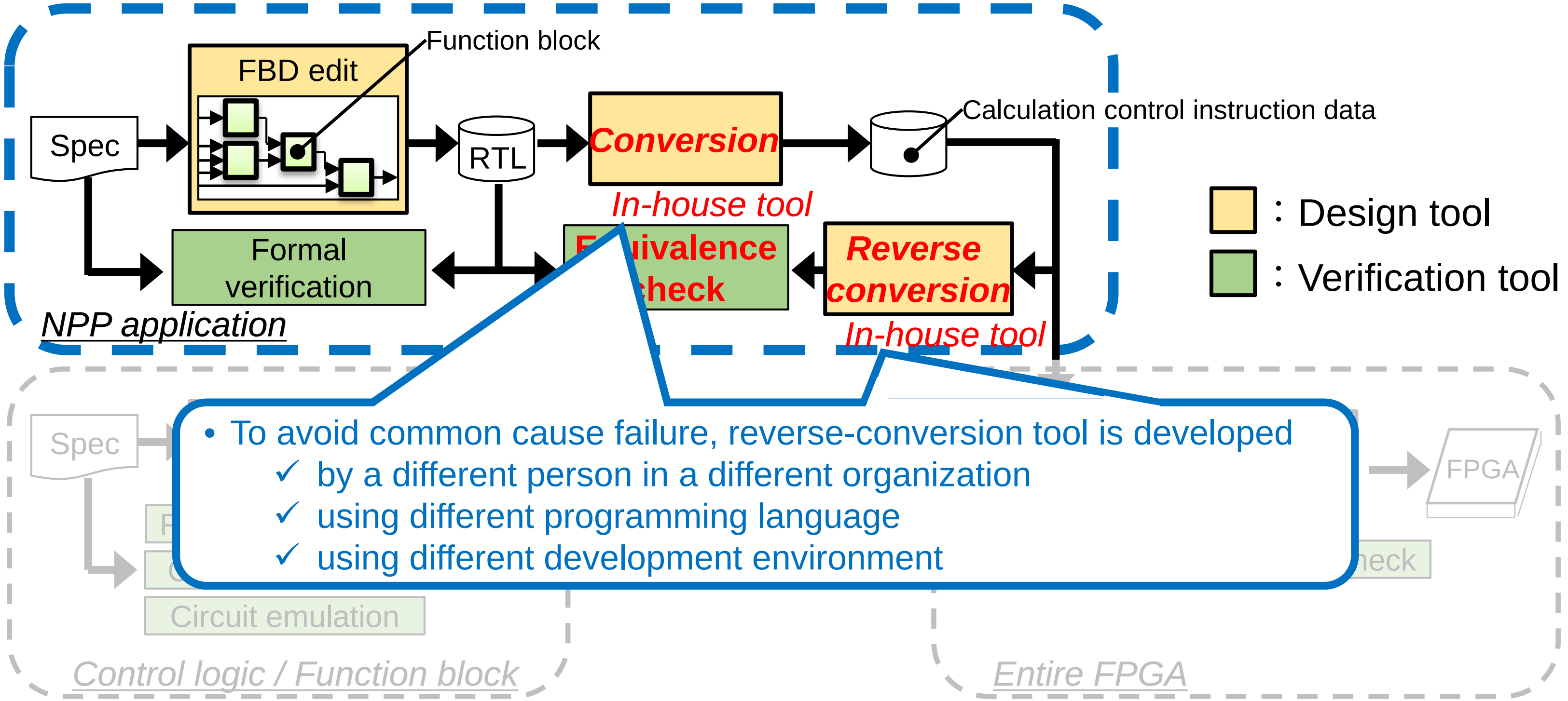
Design and Verification Process for Resource Sharing Architecture in FPGA

HITACHI
Inspire the Next



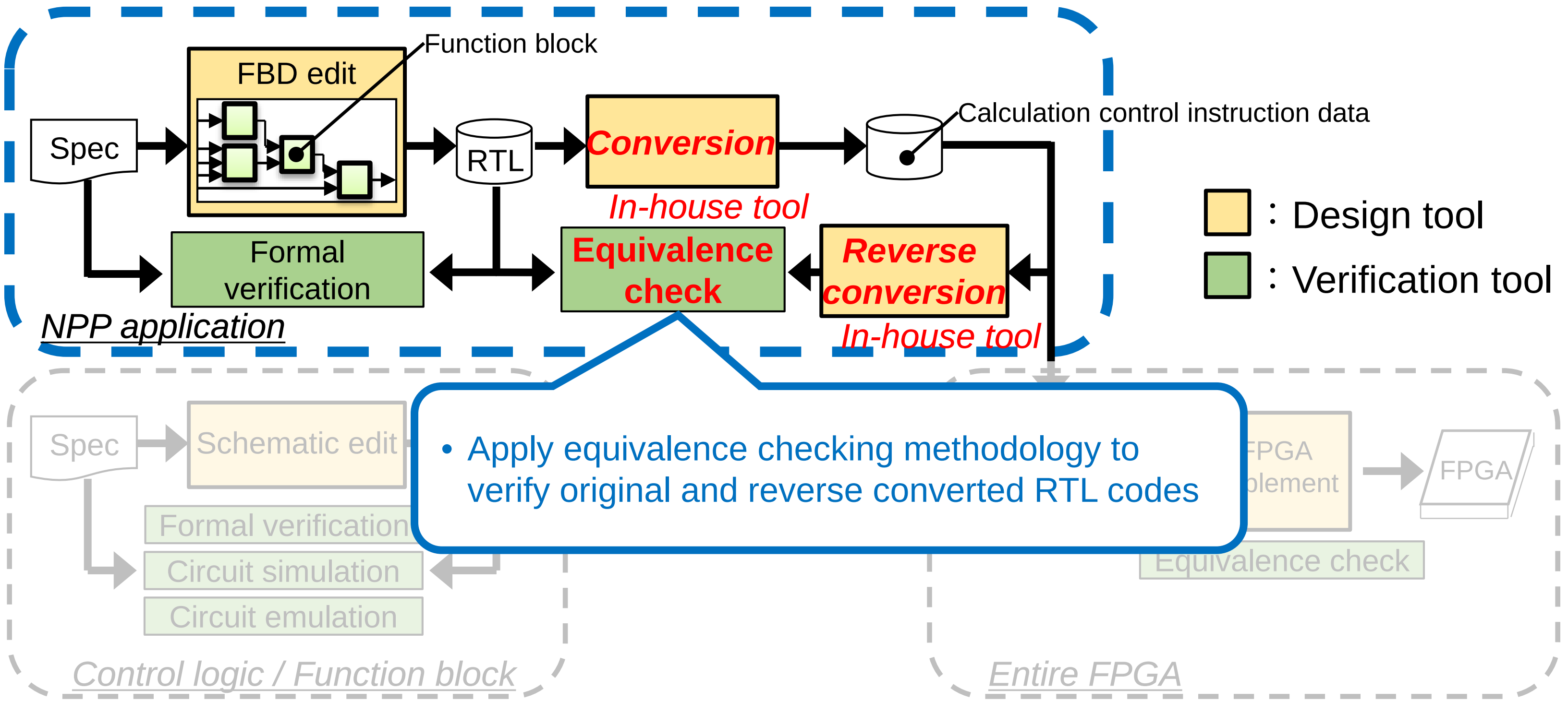
- Introduce "reverse-conversion tool" generating RTL code having the same functionality as the original RTL code generated from FBD editor

Design and Verification Process for Resource Sharing Architecture in FPGA



- To avoid common cause failure, reverse-conversion tool is developed
  - ✓ by a different person in a different organization
  - ✓ using different programming language
  - ✓ using different development environment

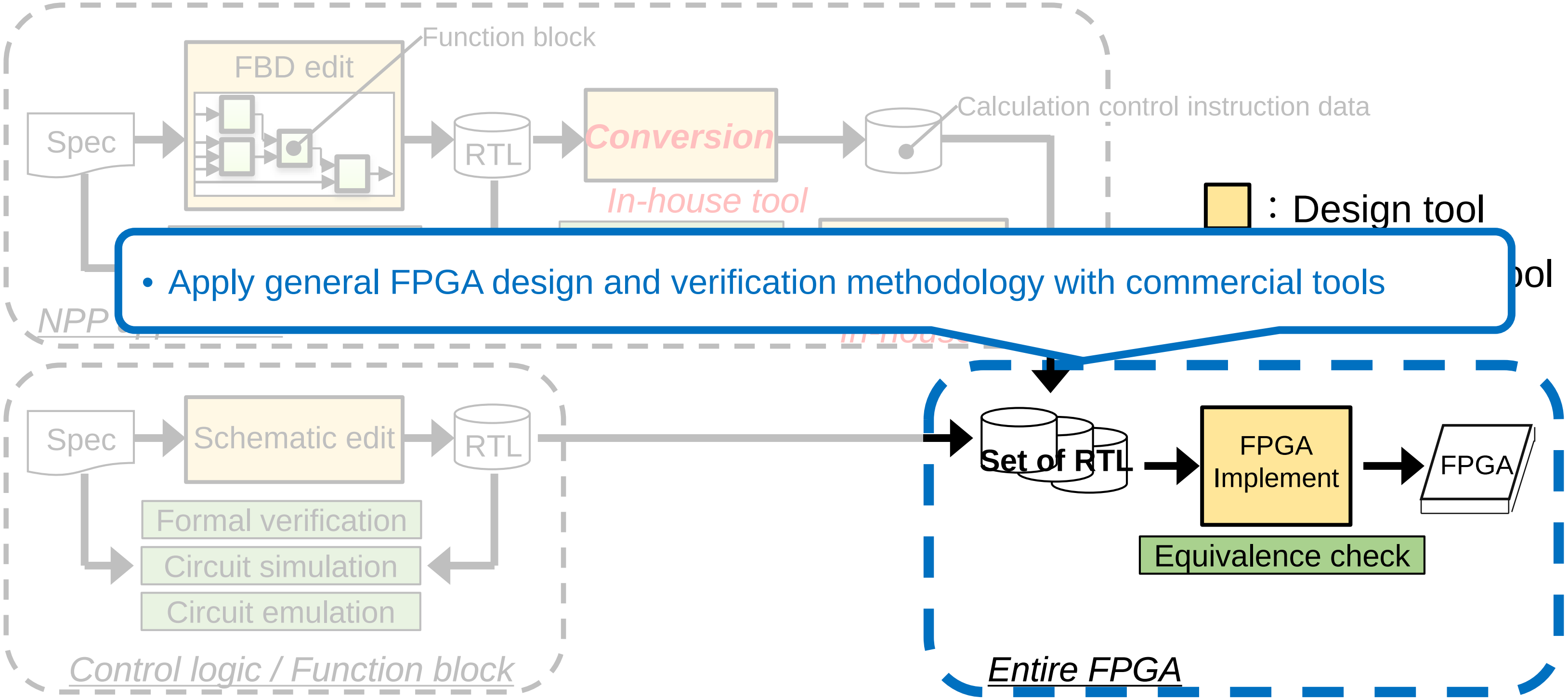
Design and Verification Process for Resource Sharing Architecture in FPGA



- Apply equivalence checking methodology to verify original and reverse converted RTL codes

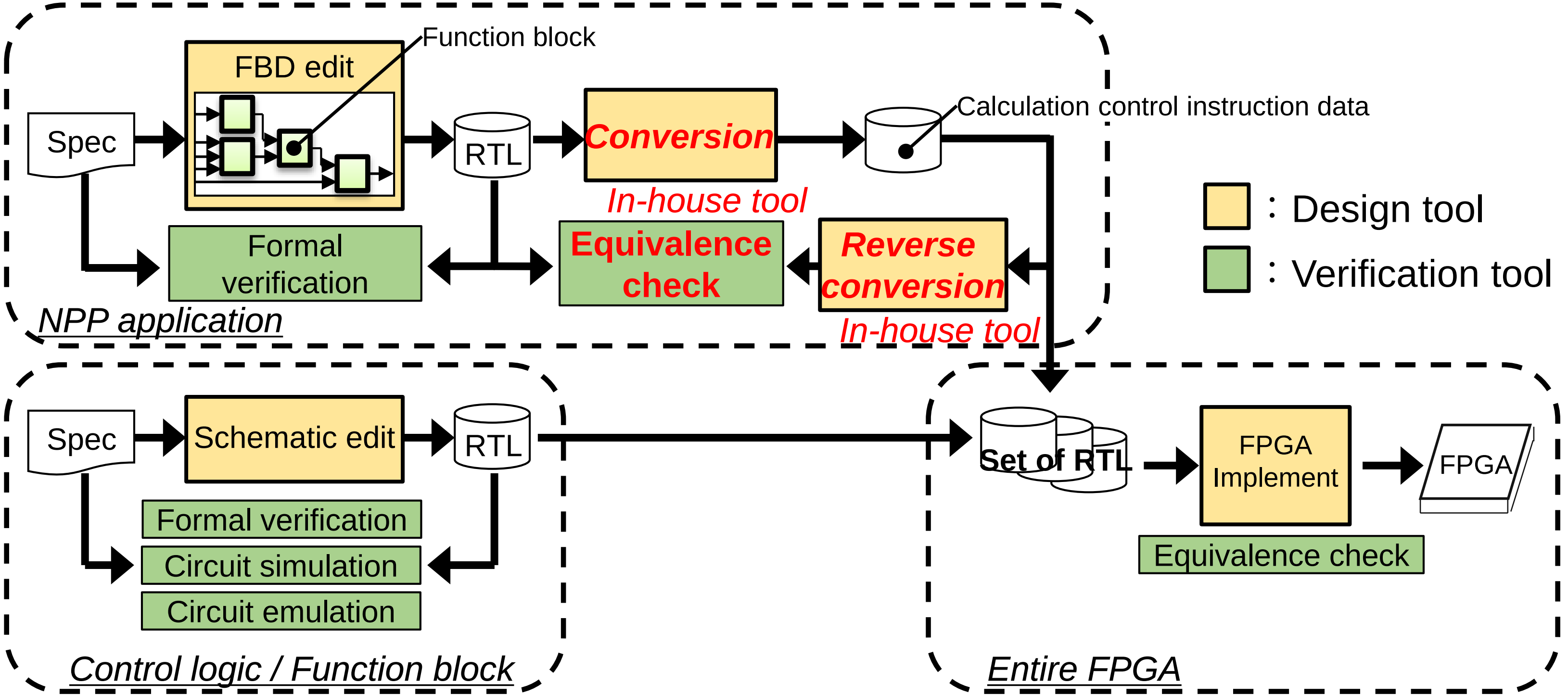
Design and Verification Process for Resource Sharing Architecture in FPGA

HITACHI
Inspire the Next



- Apply general FPGA design and verification methodology with commercial tools

Design and Verification Process for Resource Sharing Architecture in FPGA



Functional Safety Controller

- FPGA-based functional safety controller was realized by applying the resource sharing calculation architecture and the comprehensive verification process

- Design and verification process was accepted in accordance with SIL 4 compliant by TÜV Rheinland Industrie Service GmbH.

Conclusion

- We proposed hardware-resource-efficient and safe design and verification process for the FPGA-based functional safety controller
  - ➢Resource sharing calculation architecture on the FPGA
  - ➢Conversion tool generating calculation control data from FBD application
  - ➢Output data of conversion tool is verified by the following process
    - Output data is reversely-converted to the application in the FBD format
    - Equivalence checking between reversely-converted data and NPP application
- Our proposed design and verification process was accepted in accordance with SIL 4 compliant by the third party certification body, TÜV Rheinland.

HITACHI
Inspire the Next

END

Class 1 Compliant Design and Verification Process for FPGA-Based I&C System

Oct 15, 2019

Satoshi Nishikawa, Junichi Kumagai, Takumi Uezono

Hitachi, Ltd.

HITACHI
Inspire the Next