Solution House GAIO TECHNOLOGY

Model-based Dev. Tool

ISO 26262
IEC 61508
Certified

MC-Verifier

Integrated Model/Code Back-to-Back Test Tool

- **Back-to-Back (B2B) Test Tool for meeting ISO 26262 requirements**
- **Model-to-Model, Model-to-Software, Model-to-Target-Code B2B Testing**
- **Identify test error location (subsystem) on Simulink model**
- **Trace test errors over time on Simulink model**
- **Code Coverage (Statement, Branch, MC/DC)**

"MC-Verifier" is an integrated test tool for performing model-based development Back-to-Back testing. Debug and verify consistency between model, software, and target code implementation at a variety of development phases.

[List of Terms]	
MC= Model and Code	MIL = Model In the Loop
B2B = Back-to-Back	SIL = Software (C code) In the Loop
	PIL = Processor (Object code) In the Loop

Model, Software & Target Code B2B Testing, Evaluation & Reporting

MC-Verifier can perform B2B testing with models (MIL), software (SIL), & target code (PIL). Evaluate B2B test results to detect and report error locations. Import test cases created from other MBD tools for B2B testing.
Note: GAIO's own MPU simulator is built-in for target code testing. Test hardware is not required.

Identify Error Location on Simulink model

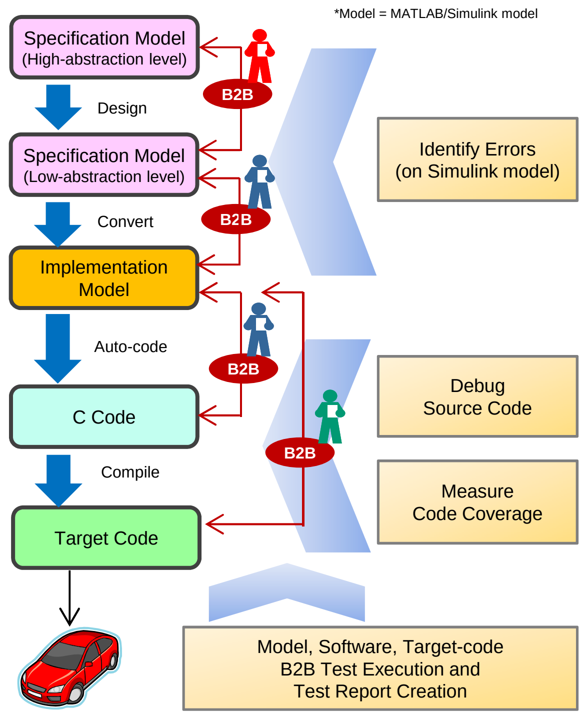
The output values of all model subsystem signals, and all code variables are recorded during testing. After testing, the model signal and code variable output values can be analyzed over time in order to locate errors.

The acceptable error range can be specified by percent. Subsystems that exceed the acceptable error range will be highlighted on the Simulink model in order to efficiently locate and correct errors. This feature for locating errors even works for models that include feedbacks.

Code Debugging Features

Code debugging features included with the MPU simulator can be used to analyze errors for model-to-target-code tests. Set breakpoints in the code, verifying changes in variable values, register values, memory values, etc.

MC-Verifier B2B Test Framework


*Model = MATLAB/Simulink model
Specification Model (High-abstraction level)
Design
Specification Model (Low-abstraction level)
Convert
Implementation Model
Auto-code
C Code
Compile
Target Code
B2B
B2B
B2B
B2B
Identify Errors (on Simulink model)
Debug Source Code
Measure Code Coverage
Model, Software, Target-code B2B Test Execution and Test Report Creation

MATLAB/Simulink Integration

MC-Verifier is integrated with MATLAB/Simulink. MC-Verifier features can be executed from the MATLAB/Simulink GUI, command-line or MATLAB scripts.

Code Coverage

Code coverage (statement, Branch, MC/DC) can be measured for model-to-target-code tests. Detect code structure issues through code coverage testing and fulfill ISO 26262 code coverage requirements.

Model-Based Development - B2B Test Use Cases

High-Abstraction Level & Low-Abstraction Level Specification Models

Perform B2B testing with the high-abstraction level and low-abstraction level specification models for verification. Evaluate behavior consistency between models when creating control specification.

Control System Designers
(MIL to MIL)
-Verify MATLAB/Simulink algorithm design
-Evaluate behavior consistency between models

Specification Model, Implementation Model & C Code

Perform B2B testing with the low-abstraction level specification model, implementation model and auto-generated code for verification. Detect C code generation related issues.

Code Developers
(MIL to MIL to SIL)
-Verify floating-point to fixed-point conversion
-Verify model modifications for code generation
-Detect auto-coder related issues

Implementation Model & Target Code

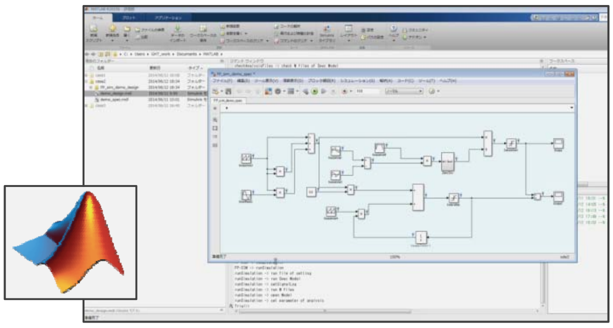
Perform B2B testing with the implementation model and the target code during the implementation phase to evaluate in a target environment as recommended by ISO 26262. Detect MPU structure, cross compiler optimization and other target code related issues.

Code Implementers / Testers
(MIL to PIL)
-Find MPU and compiler related issues, rounding errors
-Measure code coverage
-Create ISO 26262 compliant test reports

MC-Verifier Features (In development, actual features may differ)

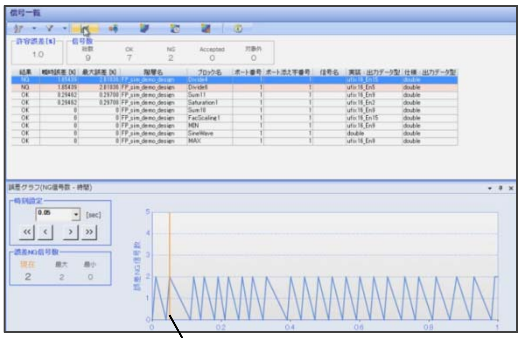
MATLAB/Simulink Integration

MC-Verifier is integrated with MATLAB/Simulink. MC-Verifier features such as model, software, target-code B2B testing and report generation can be executed from the MATLAB/Simulink GUI, command-line or MATLAB scripts.

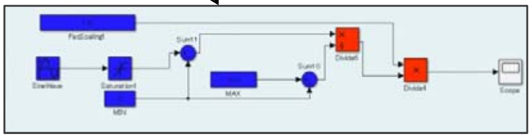


Analyze the Error Range

Evaluate the error range over time for model signal lines and code variable values. Set the allowed error range by percent, then graph and highlight items that exceed the allowed error range.



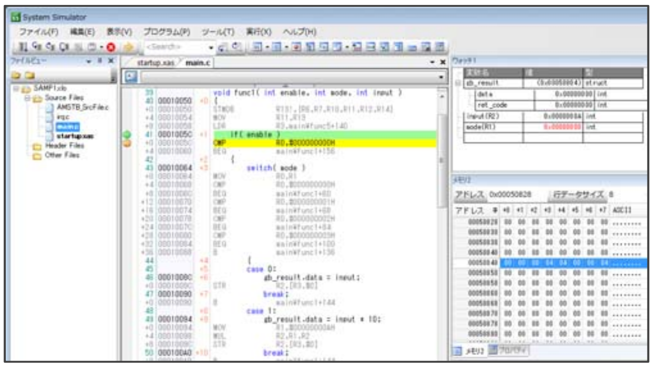
Select the time on the graph



Blocks are highlighted according to error range

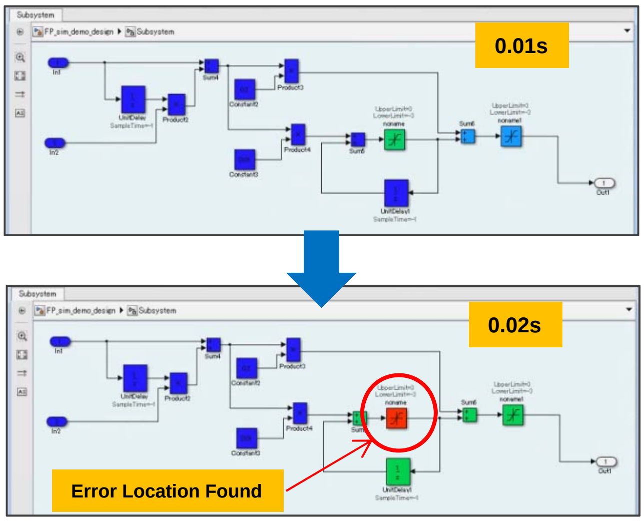
Code Debugging

Target code B2B tests are executed on GAIO's MPU simulator (Instruction Set Simulator). Test and debug the target code without needing actual hardware. Set breakpoints in the code, verifying changes in variable values, register values, memory values, etc.



Error Location Analysis

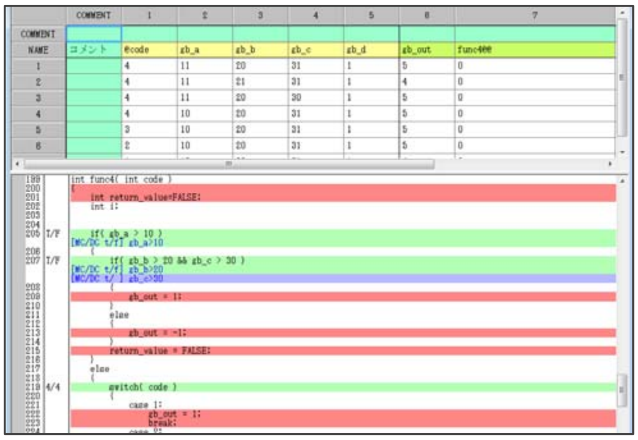
The output values of signal lines and code variables for the selected subsystem are recorded during testing. After testing, signal line and code variable values can be analyzed over time in order to locate errors. This feature for locating errors even works for models that include feedbacks.



Block highlighted colors indicate the degree of error
(From small to large: BLUE - GREEN - RED)
Use to locate the cause of errors when they first occur

Code Coverage

Code coverage (statement, Branch, MC/DC) can be measured for model-to-target-code tests. In this way, detect code structure issues through code coverage testing and fulfill ISO 26262 code coverage requirements.



	COMMENT	1	2	3	4	5	6	7
COMMENT								
NAME	コメント	@code	gb_a	gb_b	gb_c	gb_d	gb_out	func4@@
1		4	11	20	31	1	5	0
2		4	11	21	31	1	4	0
3		4	11	20	30	1	5	0
4		4	10	20	31	1	5	0
5		3	10	20	31	1	5	0
6		2	10	20	31	1	5	0

```
          int func4( int code )
          {
              int return_value=FALSE;
              int i;

205 T/F       if( gb_a > 10 )
          [MC/DC t/f] gb_a>10
              {
207 T/F           if( gb_b > 20 && gb_c > 30 )
          [MC/DC t/f] gb_b>20
          [MC/DC t/ ] gb_c>30
                  {
                      gb_out = 1;
                  }
                  else
                  {
                      gb_out = -1;
                  }
                  return_value = FALSE;
              }
              else
              {
219 4/4           switch( code )
                  {
                      case 1:
                          gb_out = 1;
                          break;
```

Test Reports

Test reports including B2B test results, tested subsystem/function list and code coverage results can be output to XML, HTML, CSV, XLS(X) formats. Easily create test reports needed for functional safety certification.

Import Test Cases from other MBD Tools

Test case data created from other MDB tools can be easily imported for B2B testing in CSV, M-file and other general formats.

GAIO TECHNOLOGY CO.,LTD.
Sales Division
Tennouzu First Tower 25F
2-2-4 Higashi-Shinagawa, Shinagawa-ku, Tokyo 140-0002 Japan
E-mail: info@gaio.co.jp WEB: http://www.gaio.com/

Feb. 2017