Report to the Certificate

Z10 16 11 55460 008

Software Component

Code Generator SCADE Suite KCG 6.6

Manufacturer:

Esterel Technologies
14 & 15, Place Georges Pompidou
78180 Montigny-le-Bretonneux
- France -

Report No.: EM90205C
Revision 1.1 of December 5th, 2016

Test Body:
TÜV SÜD Rail GmbH
Rail Automation
Barthstr. 16
D-80339 München

Certification Body:
TÜV SÜD Product Service GmbH
Ridlerstraße 65
D-80339 München

Table of Contents page

List of tables page

List of figures page

Revision history

Rev.	Date	Author	Modification / Description
1.0	Nov. 17th, 2016	U. Kremer	Initial, certification of KCG 6.6 based on the certification of KCG 6.4 (report EE86102C, rev. 1.0)
1.1	Dec. 5th, 2016	U. Kremer	Update of references in table no. 3

Table 1: Revision history

1 Purpose and Scope

The Code Generator SCADE Suite KCG 6.6 has been developed by Esterel Technologies using the implementation language OCAML. The input notation from the predecessor version was extended by merging block diagrams and safe state machines besides the introduction of improvements such as array types and operators.

The Code Generator SCADE Suite KCG 6.6 is an incremental release of SCADE Suite KCG 6.4.

This Report to the Certificate is a set of the results of all steps made during certification of Code Generator SCADE Suite KCG 6.6. It is based on the requirements in chapter 4.1 and the documents listed in chapter 4.2.

TÜV SÜD Rail GmbH has been contracted by Esterel Technologies in October 14th, 2015 for certification of the update to Code Generator SCADE Suite KCG 6.6. TÜV SÜD Rail GmbH project number is 717511669.

2 Product

2.1 Overview

The Code Generator SCADE Suite KCG 6.6 is an automatic code generator. It takes as input a formal SCADE model and generates C or Ada source code.

The Code Generator SCADE Suite KCG 6.6 can be used with the SCADE[1] Suite integrated development environment, developed by Esterel Technologies. SCADE Suite is used to develop applications using the SCADE language. Its purpose is to develop control software.

SCADE Suite is not part of the assessment.

[1] SCADE - Safety Critical Application Development Environment

2.2 Code Generation Chain of KCG 6

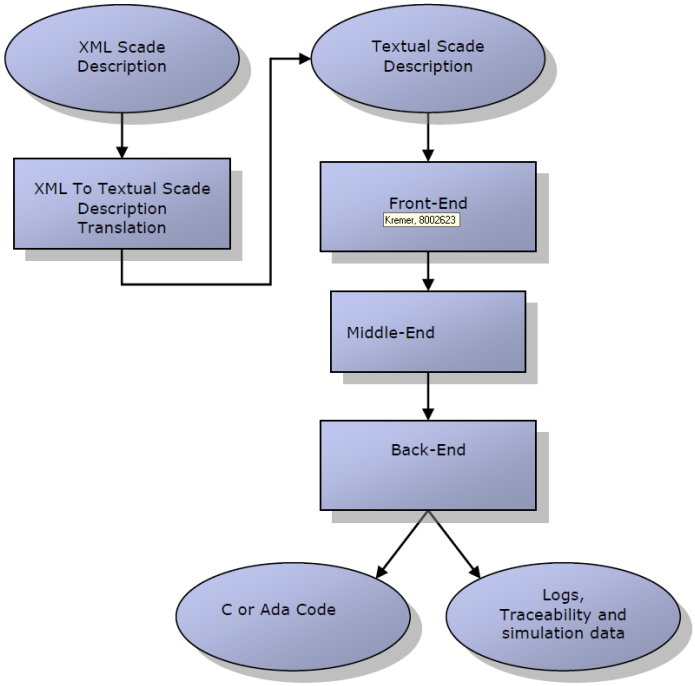

Figure 1: KCG 6 code generation

- **XML to Textual step:** the XML format is translated into the corresponding SCADE textual format, keeping only mandatory information for code generation and discarding information required for drawings (such as position, color, …).
  - **Front-End**: loads the input files and does all the checks
  - **Loading step:** the SCADE input files are loaded by the tool and basic syntax checks are performed.
  - **Static semantic checks step:** the model is analyzed (type checks, consistency of declarations, check of model semantics wrt language semantics…).
- **Middle-End:** performs internal transformations
  - **Normalization step:** model is transformed to ease next steps. The transformation is performed using rewriting techniques which ensure the consistency of the model and keep its semantics unchanged.
  - **Optimization step:** during this step, dataflow equations are optimized.
- **Back-End:** prepares for output and generates code
  - **Sequentialization step:** this step performs the scheduling of the dataflow so that it can be finally translated into an imperative-style language such as C or Ada.

- **Optimization step:** during this step, scheduled equations are optimized, as well as memory variables.
- **Code generation:** this is the last step where the output C or Ada code is generated.

2.3 Safety Properties

2.3.1 Input language – Scade Language

The SCADE language is designed for modeling real-time applications. It has a textual notation and a graphical notation, with a direct mapping between both notations.

The properties of the SCADE language and of the generated code and the KCG development process ensure that the generated code complies with model at source level. Therefore, module testing of the generated code can be alleviated.

In addition to previous versions KCG 6 introduces state machines in the hierarchy of nodes. Each node can contain equations and state machines. Equations are mainly used to express the data parts of a model. The Scade 6 state machines are hierarchical and allow parallelism. They allow to design complex control parts of a model. Equations and state machines can be freely mixed, at any level.

- The input model has a formal (precise and unambiguous) definition. Its meaning is completely accurate and is formally defined by the SCADE language semantics.
- Code Generator SCADE Suite KCG 6.6 implements the SCADE language and must respect its determinism: any specific input sequence will always produce the same output sequence.
- Code Generator SCADE Suite KCG 6.6 itself is deterministic. A given model will always produce the same output C/Ada code provided that the code generation options are fixed.

2.3.2 Output Language – C/Ada code

Independently from the choice of the code generation options, the generated C/Ada code has the following properties:

- The C code is portable: it is ISO-C compliant.
- The Ada code is portable: it is SPARK 95 compliant.
- The C/Ada code structure reflects the model architecture for data-flow parts.
- The C/Ada code behaviour complies with the model semantics.
- The C/Ada code is readable and traceable to the input SCADE model through the use of corresponding names, specific comments and traceability file.
- For control-flow parts traceability between state names and C/Ada code is ensured.

- Memory allocation is fully static (no dynamic memory allocation).
- Recursions are avoided.
- Bounded loops are allowed, since they use static values known at SCADE code generation time.
- Execution time is bounded.

3 Identification

The identification of the certified Code Generator SCADE Suite KCG 6.6 revision is given in the table below:

KCG	Software Data	Executable	Checksum
6.6	Integrated executable code for Win. 7 SP1 (64 bit) and Win. 8.1 (64 bit)	KCG66.exe (build i19)	BD1D03B5784A77EA0FDF810F7DADF520

Table 2: Identification

4 Certification

4.1 Basis of Certification

The Code Generator SCADE Suite KCG 6.6 is certified according to the regulations and standards listed in chapter 4.3 of this document. This comprises the successful completion of the following test segments:

- Functional safety:
  - Fault avoidance
- Product-related quality management and product care

4.2 Certification Documentation

The certification is based on the following reports:

KCG	Report	ID	Rev.	Author
6.6	Technical Report	EE74781T	1.5	TÜV SÜD Rail
	Assessment Report	EE82740G	1.4	TÜV SÜD Rail
	Safety Case Report	KCG-TR-083	A-i2	Esterel Technologies

Table 3: Certification Documentation

The technical documentation and the documentation of the tests executed are deposited at the test body.

The certification of the Code Generator SCADE Suite KCG 6.6 according to the regulations and standards listed in chapter 4.3 verifies successful completion of the assessment and the product-related Quality Assurance in Manufacture and Product Development.

4.3 Standards and Guidelines

Reference	Title
IEC 61508 (part 1, 3, 4):2010, 2nd Edition	Functional Safety of electrical/electronic/programmable electronic safety-related systems (requirement: SIL3)
EN 50128:2011	Railway Applications - Communication, signalling and processing systems – Software for railway control and protection systems
ISO 26262-6:2011	Road vehicles - Functional safety - Part 6: Product development at the software level
ISO 26262-8:2011	Road vehicles - Functional safety - Part 8: Supporting processes

Table 4: Application specific and functional safety standards

5 Overall Results

5.1 Functional Safety

The tests and analyses performed by Esterel Technologies have shown that Code Generator SCADE Suite KCG 6.6 complies with the testing criteria specified in chapter 4.3 for SIL 3 according to IEC 61508, SIL3/4 according to EN 50128 and ASIL D according to ISO 26262.

Lifecycle Phase	Title
Lifecycle Issues and Documentation	• Software planning documents • S/W requirements documents • S/W design documents • S/W module documents • Source code & documentation • S/W test reports • S/W validation/integration test report • S/W assessment report • S/W maintenance records
Software requirement specification	• Natural language • Formal method partly used
Architecture	• Semi-formal methods • Structured methods • Modelling • Defensive and failure assertion programming • Fully defined interface
Support Tools and Programming Language	• Subset of C with coding standards • OCAML subset and coding rules [KCG 6.x] • Usage of static code analyzing tools • Tools: increased confidence from use resp. qualified as verification tools acc. DO-178B Level A • Translator: increased confidence from use • Configuration management
Detailed Design	• Structured methods • Modelling • Modular approach o Software module size limit o Information hiding/encapsulation o One entry/one exit point in subroutines and functions o Fully defined interface • Design and coding standards • Analysable programs

Lifecycle Phase	Title

Software module testing and integration	• Static analysis
	  o Control flow analysis
	  o Data flow analysis
	  o Walkthroughs/design reviews
	• Dynamic analysis and testing
	  o Test case execution from boundary value analysis
	  o Equivalence classes and input partition testing
	  o Structure-based testing: 100% MC/DC
	• Data recording and analysis
	• Functional and black box testing
	• Performance testing
	• Interface testing
	• Software error effect analysis (SEEA)
	• Metrics
	• Traceability matrix

Software validation	• Functional and black box testing • Simulation/modelling o Performance modelling (in sense of complex input structures) • Structure-based testing: 100% of all applicable requirements
Modification	• Impact analysis • Reverify changed software module • Reverify affected software modules • Revalidate complete system dependent of impact analysis • SW configuration management • Data recording and analysis
Software Assessment Techniques	• Checklists • Static software analysis • Dynamic software analysis
Quality insurance	• Accreditation according to EN ISO 9001

Table 5: Lifecycle Phases

5.2 Product Specific Quality Assurance and Control

The applied plans, standards and guidelines listed in the Safety Cases (see chapter 4.2) guarantee that Code Generator SCADE Suite KCG 6.6 is developed in a safe manner. In addition, Esterel Technologies has been certified according to ISO 9001.

The most important guidelines of Best Practices are:

- Guidelines for Software Engineering
- Guidelines for Customer Support
- Guidelines for Change Control of Software Products
- Guidelines for Managing Project Documentation
- Guidelines for Employees Training

- Guidelines for Audits

As part of the certification process TÜV Product Service also performs a procedure that is tailored to the assessed product in order to assess the consistency of product quality while accounting for product modifications and their identifiability (follow-up service).

5.3 Conclusion

Safety of **Scade-language** means that the language is scientifically proven to be completely accurate and formally defined. Safe State Machines, formerly expressed by Esterel-language, and Block Diagrams were merged into SCADE 6 language by the extension of LUSTRE-language.

Safety of **generated C/Ada code** means that unsafe C/Ada language constructs are explicitly excluded. It contains no dynamic memory allocation, no pointer arithmetic, and the only loops are bounded loops over delay buffers and over bounded arrays. It also means that the generated code behaviour complies with the model semantics.

Safety of **Code generator KCG** means that the behaviour of the generated C/Ada Code complies with the model semantics of the SCADE model. To avoid deviations between SCADE model and generated C/Ada Code, the development process of Esterel Technologies underlies the requirements of safety related software standards with respect to fault avoidance.

Safety of **implementation** means that the development tools used for KCG are reliable. The use of ML language and compiler has been assessed according to IEC 61508 SIL3 and EN 50128 SIL3/4. The assessment summary, provided in the technical report, concludes that the ML language and compiler with their restrictions of use are fit for the purpose of developing KCG.

Safety of **KCG application** requires that design rules of SCADE-language are applied.

Safety of **KCG integrated development environment** requires consciousness of the application developer.

Safety of **KCG execution** requires that the impact of non safety related execution platforms needs to be considered.

5.3.1 Limits

While the Code Generator SCADE Suite KCG 6.6 is suitable for the realization of safety functionality, it does not support the implementation of hardware integrity functionality. The SCADE language is function oriented and not hardware oriented.

Due to the absence of hardware integrity measures inside the model, the generated safety function/model needs to be embedded into a safety layer.

The responsibility for correctly calling the generated C/Ada code is on the user. The correct behaviour of the runtime environment shall be verified by appropriate means according to the relevant standard.

5.3.2 Summary

With respect to the given conditions of use listed in chapter 6 below the Code Generator SCADE Suite KCG 6.6 is considered to be compliant with SIL 3 according to IEC 61508-3, SIL 3/4 according to EN 50128 and ASIL D according to ISO 26262-8. Its usage for the development of safety critical applications up to SIL 3, resp. SIL3/4 and ASIL D is considered to be in accordance with the reference basis.

6 Conditions of Certification

/C1/ The safety related conditions for use which have been identified or referred inside the safety case (see chapter 4.2) have to be considered for the use of the Code Generator SCADE Suite KCG 6.6 as well as the conditions for use which have been given inside the user documentation of Code Generator SCADE Suite KCG 6.6.

/C2/ The updates of the Safety Status report issued during maintenance by Esterel Technologies have to be considered for the use of the Code Generator SCADE Suite KCG 6.6.

/C3/ For each application a specific software assessment depending on the safety integrity level of the application shall be carried out.

7 Certificate Number

This report adds technical details and implementation conditions required for the application of the Code Generator SCADE Suite KCG 6.6 to the certificate:

Z10 16 11 55460 008

TÜV SÜD Rail GmbH
Rail Automation

Peter Weiß