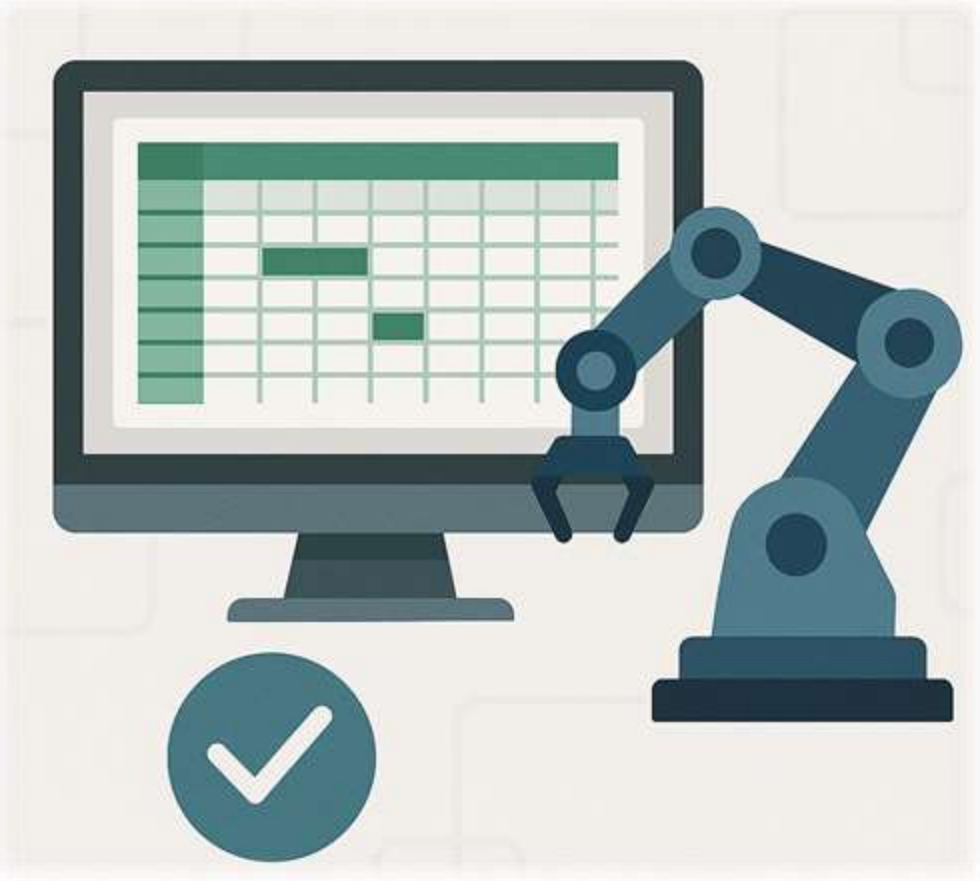
IFA
Institut für Arbeitsschutz der
Deutschen Gesetzlichen Unfallversicherung

SOFTEMA

A tool to assist in the development of safety-related application software for machines



6th PEROSH Research Conference in Manchester
Wednesday, September 10th 2025

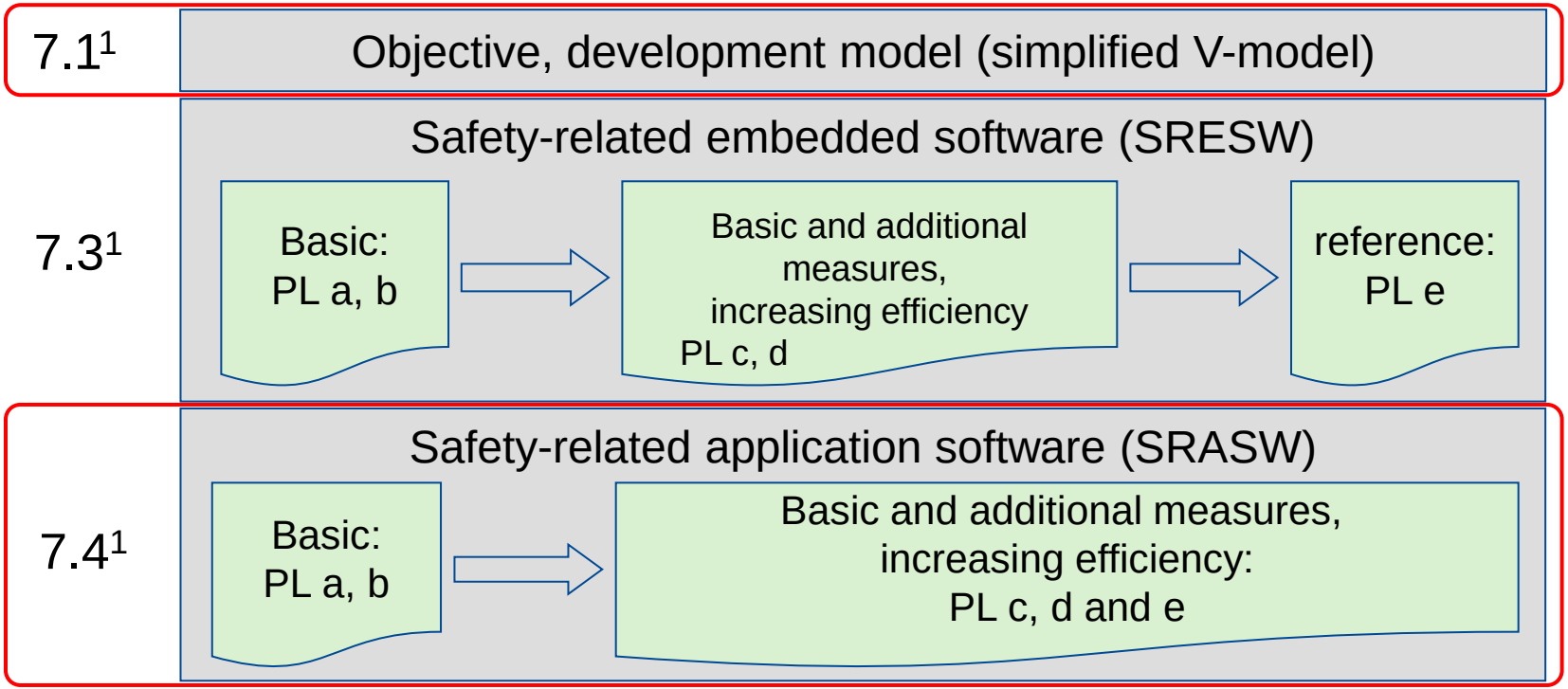
Albert Bohlscheid

Agenda

- Basic principles
  - Normative requirements
  - IFA matrix method
  - IFA Report 2/2016e
- SOFTEMA
- SOFTEMA code visualisation

ISO 13849-1: normative requirements

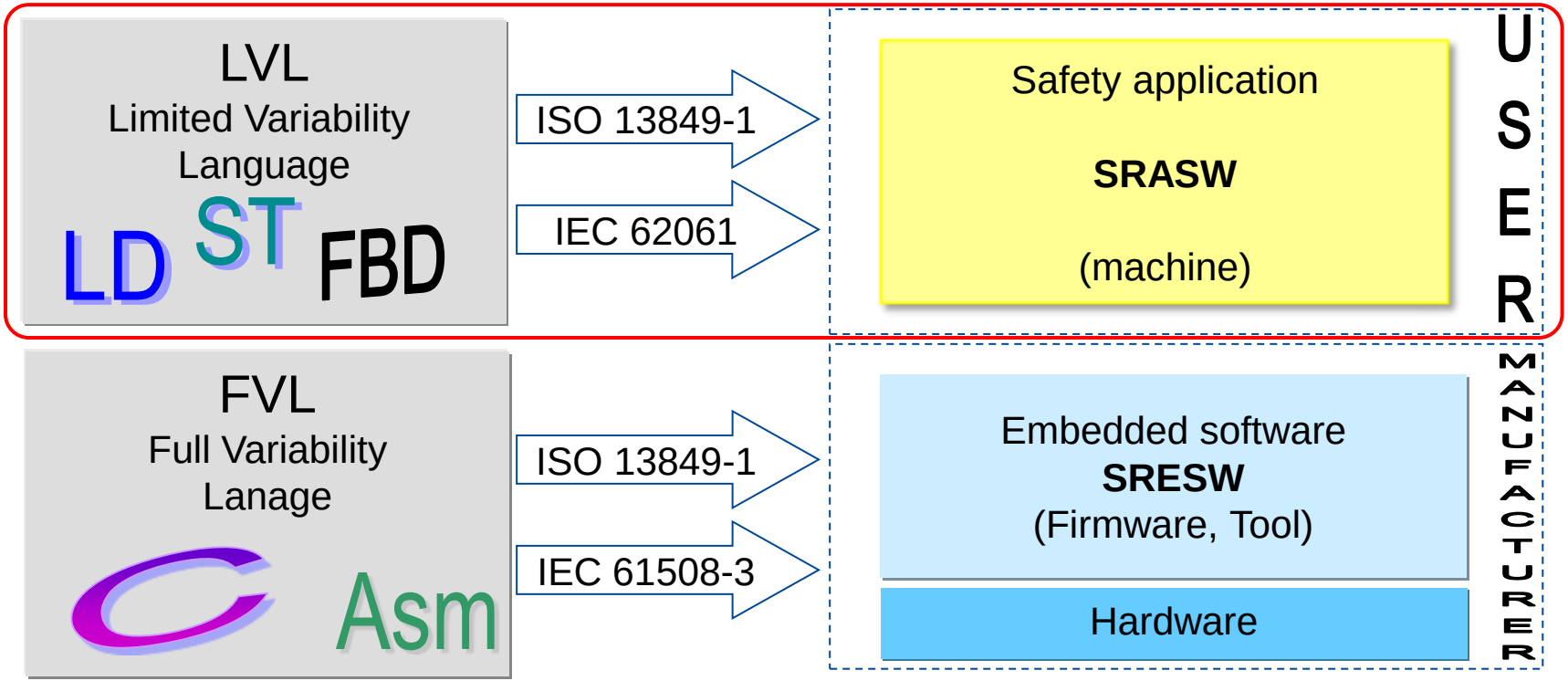
ISO 13849-1: normative requirements



⇒ IFA Report 2/2016e "application software", Chapter 3

[1] ISO 13849-1:2023

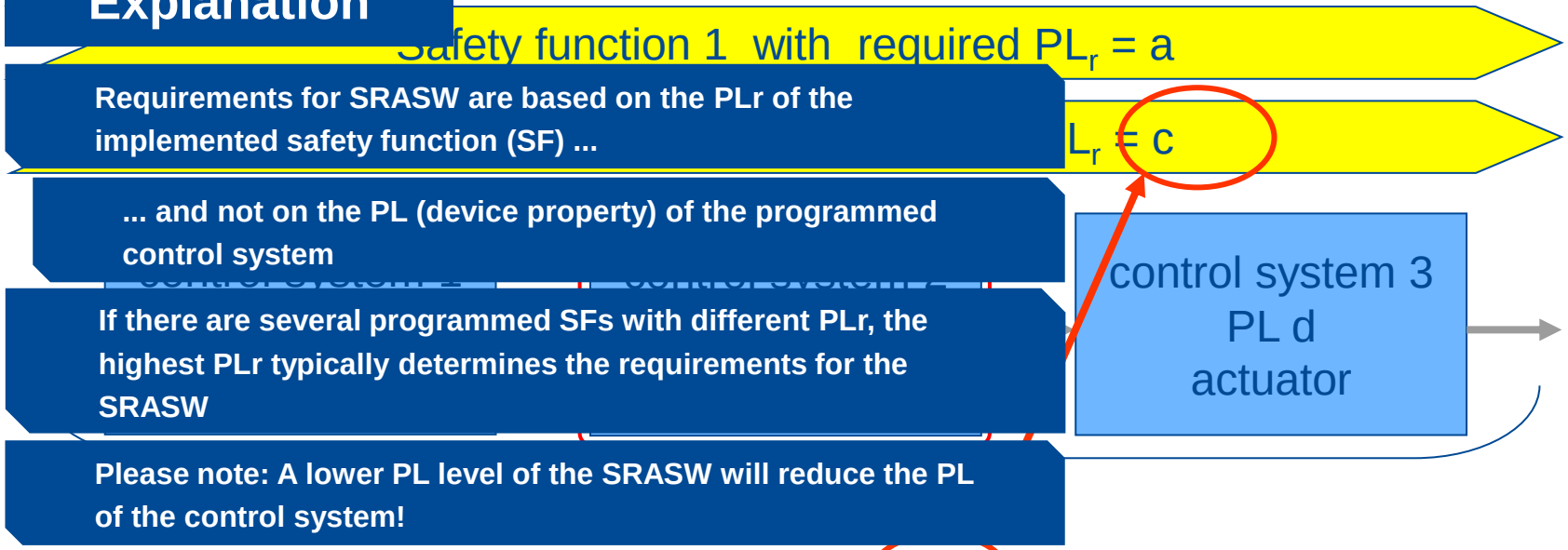
Difference between SRESW and SRASW



Implementation of safety functions

Explanation

Source: IFA Report 2/2016e "application software", Chapter 4



Requirements for SRASW are based on the PLr of the implemented safety function (SF) ...

... and not on the PL (device property) of the programmed control system

If there are several programmed SFs with different PLr, the highest PLr typically determines the requirements for the SRASW

Please note: A lower PL level of the SRASW will reduce the PL of the control system!

Resulting PL = d for hardware, maximum PL_r e = c for SRASW
Resulting PL = c for application software
=> Overall PL = c (for hardware and software)

Overview of techniques and measures

- Development model (V-model)
- Modular and structured programming
- Documentation of specifications and design
- Appropriate development activities following changes
- Functional test ...

Basic measures

- ... and extended functional test
- Specification of security requirements/safety functions
- Project management
- Programming guidelines
- Configuration management
- etc.

Additional measures

Most of these measures are implemented by SOFTEMA

Source: IFA Report 2/2016e "application software", Chapter 5

IFA Report 2/2016e "Application software for machines"

- Motivation for IFA publication
  - Hardly any patterns/examples, repeated requests from machine manufacturers
  - 13849-1 in 3rd edition; 13849-2 with software validation
  - Completion of the FP319 funding project (2011-2013)
- Contents
  - Description of the IFA matrix method
  - Additionally: error prevention measures, validation, etc.
  - Examples
- Available in English and German

SOFTEMA

The IFA matrix method

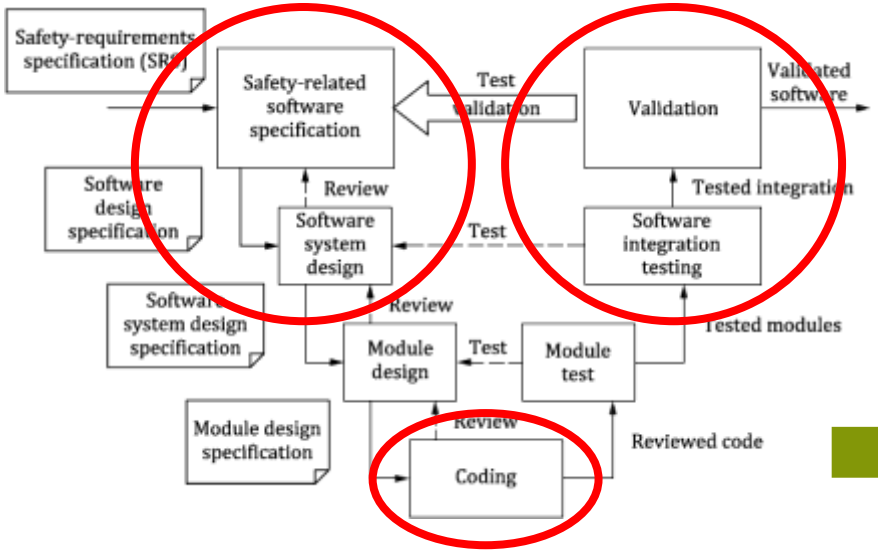
Basic requirements for the IFA matrix method

- PLC development environment (e.g. 61131) specifies software architecture
- 3-level software structure (input/logic/output)
- Purely Boolean logic

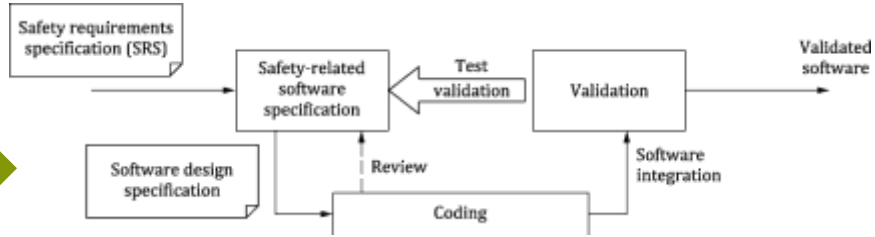
The V-model

Design Activities

Reviewing Activities



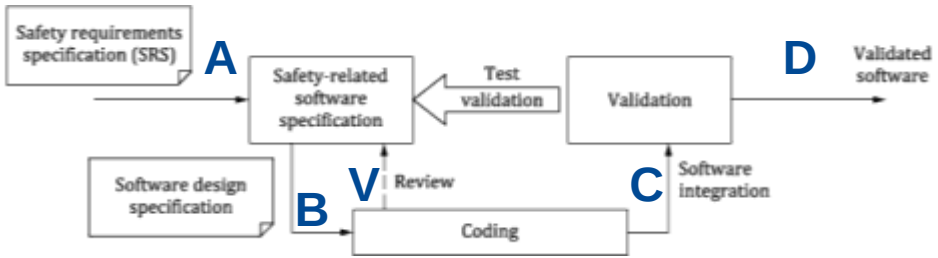
a) Simplified V-model of the software safety lifecycle[1]



b) Simplified V-model for software[1]

[1] ISO 13849-1:2023

Basic requirements for the IFA matrix method

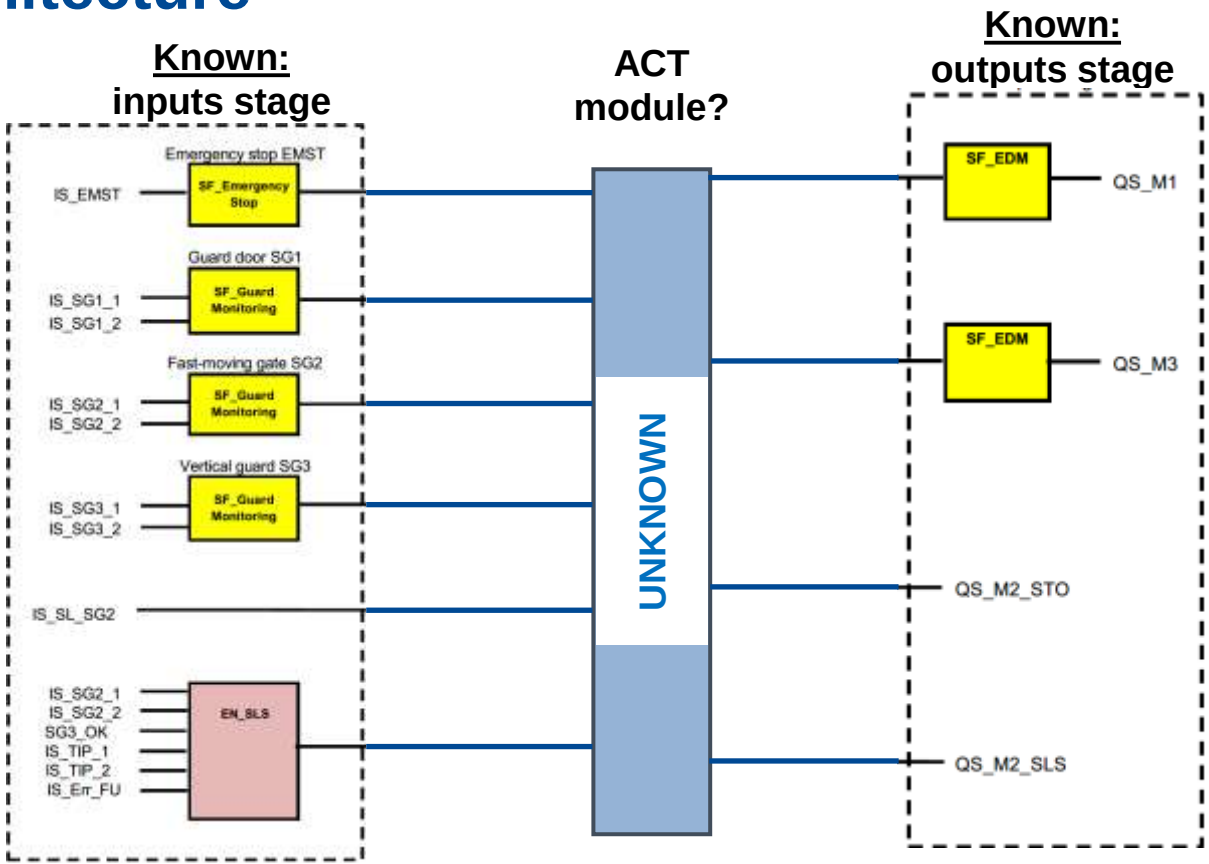


Abbreviation	document
A1	Specification of safety functions
A2.1	Specification of the hardware, system sketch
A2.2	Specification of the hardware, circuit diagram
A2.3	Specification of the hardware, system structure
A2.4	Specification of the hardware, I/O list
A3	Catalogue of measures for fault avoidance, tools and programming rules
A4	Requirements 13849-1
B1	Architecture of a safety program
B	Architecture of a standard program
B3	Modular architecture
B4	Safety-related software specification (cause-effect matrix)
B5	Program sketch
V1	Protocol of verification
C1	Protocol of code review
D1	Protocol of software validation

B3: Modular architecture

Note:

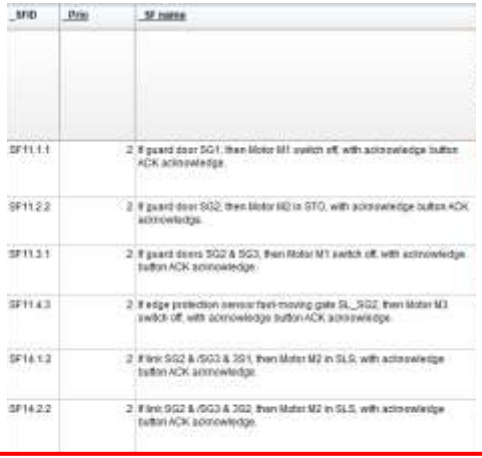
In SOFTEMA, the function blocks are stored as a list in Table B3: Modular architecture.



Certified library modules
Library modules developed in-house
Processing stage developed in-house

B4: Software specification (C&E matrix)

verification
validation



Additional
test lines

Names
date

C1: Code review protocol (expandable)

No	Description	_Reference sheet	Verification	_Comment	_Comment_Check
R1	Have the agreed error-prevention measures, tools and programming rules been adhered to?	A3 Measures	OK		
R2	Has the system structure of the hardware been implemented?	A2.3 System structure	OK		
R3	Has the interconnection of the I/O signals been implemented correctly?	A2.4 IO list	OK		
R4	Has the architecture of the safety programme been adhered to?		OK		
R5	Has the modular architecture been adhered to?	B3 Module architecture	OK		
R6	Has the software specification from the matrix been implemented?	B4 Matrix C+E	60 %		
€€€					
		Sum	83 %		
		Date	02.04.2014		
		Name	Johanna Dietz		
		Signature	1272993002		
		Date	30.08.2025		
		Check1	Willi Minmax		

D1: Software validation protocol (expandable)

No	Description	Reference sheet	Validation	Comment	Comment_Check
	Have the activities been carried out?				
V1	Validation of safety functions (D1)	A1 Safety functions	0 %		
V2	Validation of I/O check (D1)	A2.4 IO list	OK		
V3	Validation of normative requirements (D1)	A4 Requirements	97 %		
V4	Verification of the modular architecture (V1)	B3 Module architecture	OK		
V5	Verification of the matrix (V1)	B4 Matrix C+E	60 %		
V6	Matrix validation (D1)	B4 Matrix C+E	60 %		
V7	Verification code review	C1 Codereview	83 %		
V8	Verification of peripheral devices		OK		
V9	Testing the sensors		OK		
	Is the documentation complete?				
D1	Documents of the V-model from this Excel document		OK		
D2	PDF printout of all safety-relevant software incl. checksum		OK		
D3	PDF printout of the hardware configuration (with all settings) incl. checksum		OK		
D4	Archiving of the manuals for all system components		OK		
D5	PDF printout of the configuration of peripheral devices incl. checksums		OK		
D6	Manufacturer's acceptance regulations (e.g. parameterisation of safety components)		OK		
D7	Specifications to be complied with from C standards		OK		
D8	Specifications to be complied with from B standards		OK		
€€€					
		Sum	71 %		
		Date	04.04.2014		
		Name	Marcel Benus		
		Signature	1272993002		

Basic principles of SOFTEMA

- Implementation of the matrix method in “spreadsheet format"
- based on Excel files ""*.xlsx", but Excel is not required
- Per PLC/application programme: 1 SOFTEMA project = 1 Excel file
- Can be edited alternately with Excel/SOFTEMA
- Automation: modifications, formal verification, generate and print checklists + test plans, etc.
- Users with different roles/authorisations
- Cross-project master data: persons, documents, programming rules, function blocks

Advantages of SOFTEMA

- Works similarly to Excel, but can do more and is simply easier to use
- Saves time through automation
- Error detection
- Simply documentation, programme-controlled printing
- Simply modification
- Flexible support for validation and testing
- Logging/encryption

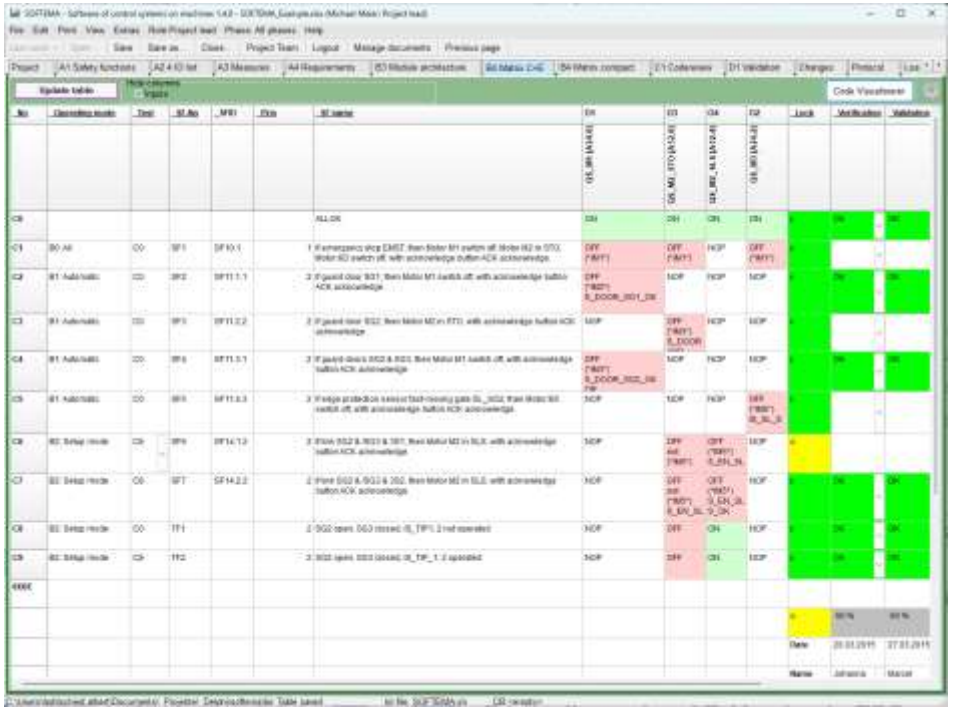
Summary

- Pragmatic method for processing a further simplified V-model
- Independent of control system, programming language and PLr
- Key points of the method are
  - Division of the software into inputs stage, logic, outputs stage
  - Specification of the logic using a cause and effect matrix
  - Higher test coverage with additional test cases
- Documentation as tables in Excel format
- Planning and testing always together in one table
- Software validation consists of inspections (verification, code review) and functional tests (I/O list, cause and effect matrix) + manufacturer-specific tests

SOFTEMA

Code visualisation

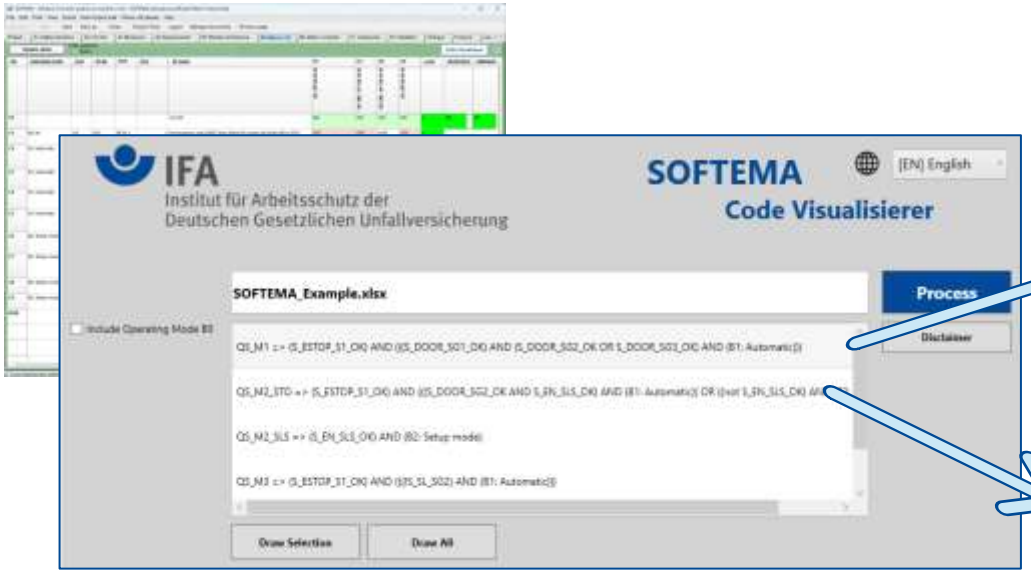
SOFTEMA code visualisation



- Manual transfer of the CE matrix to a development environment
- Potential problems:
  - Limited clarity
  - Increased susceptibility to errors
- Typical development languages with limited language scope:
- Structured text (ST)
- **Function block diagrams (FBD)**

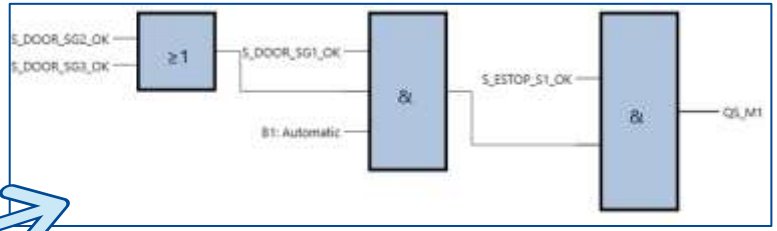
OBJECTIVE **Representation of the CE matrix as an FBD**

Working with the SOFTEMA code visualiser

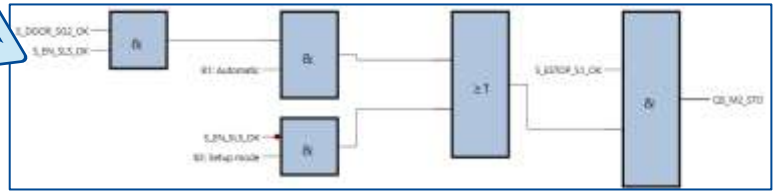


SOFTEMA Code Visualiser

Visualised FBD with output QS_M1



Visualised FBD with output QS_M2_STO



RESULT **CE matrix represented as FBD**

IFA
Institut für Arbeitsschutz der
Deutschen Gesetzlichen Unfallversicherung

Albert Bohlscheid
Research Officer
Topic: SOFTEMA

albert.bohlscheid@dguv.de

Thank you for your attention.