

Entwicklung von Medizinischen Algorithmen für die Kardiologie mit Hilfe Simulink® und Modellbasiertes Design



Dr.-Ing. Antoun Khawaja

CEO KhawajaHealth

Antoun.Khawaja@KhawajaHealth.com

Antoun.Khawaja@Corpuls.com

27.06.2017, MATLAB EXPO 2017 München Deutschland

Agenda

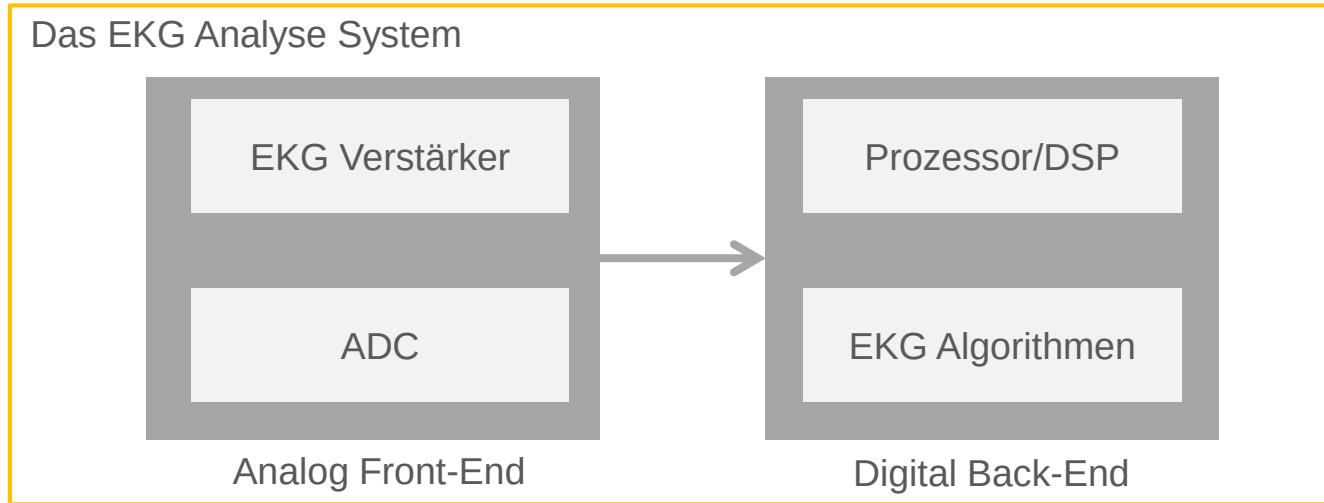
- Einführung
- Ziel der Arbeit
- Hintergrund Kardiologie
- Klassische und Modellbasierte Entwicklung
- Die gesamte Entwicklungsplattform
- Warum Modellbasiertes Design (MBD)? Sieben gute Gründe...
- Zusammenfassung & Take-Home Message

Einführung

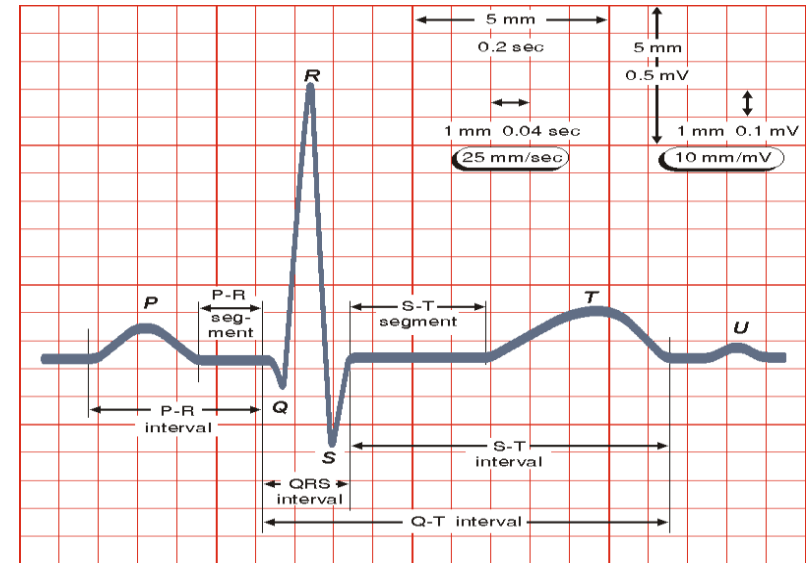
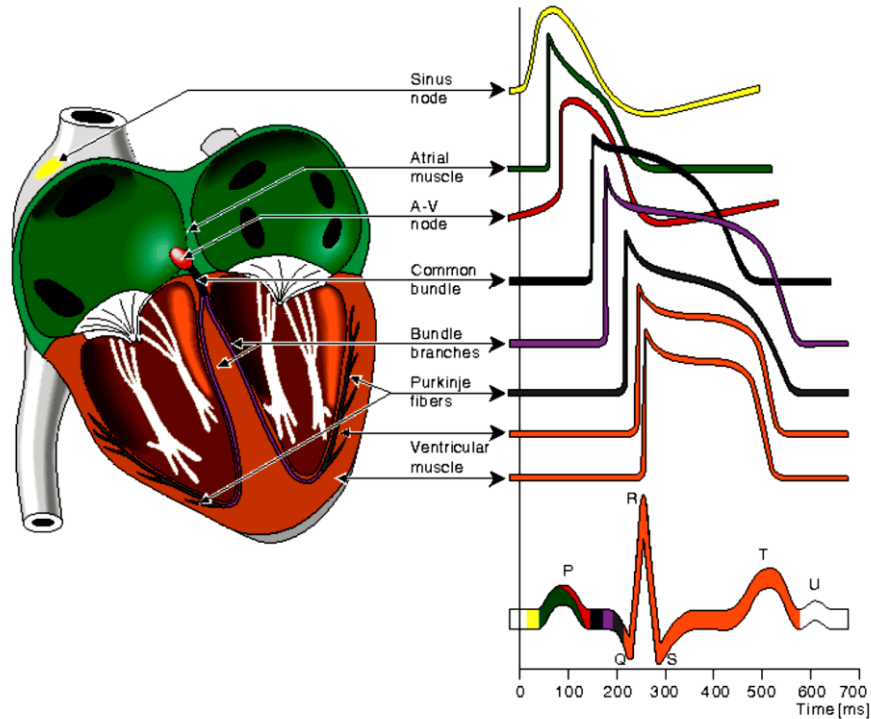
- Herz-Kreislauf Erkrankungen ist eine der häufigsten Todesursache
- Das Elektrokardiogramm (EKG) ist die zentrale Untersuchungsmethode für die Diagnostik und die Überwachung von Patienten mit Herz-Kreislauf-Erkrankungen, Herzrhythmusstörungen, oder chronischer Herzinsuffizienz.
- Kardiologische Medizingeräte haben oft komplexe Diagnose-Algorithmen, die für bestimmte Hardware maßgeschneidert und nach klassischen Entwicklungsmethoden, manuell und sehr aufwendig, erstellt und getestet sind.

Ziel der Arbeit

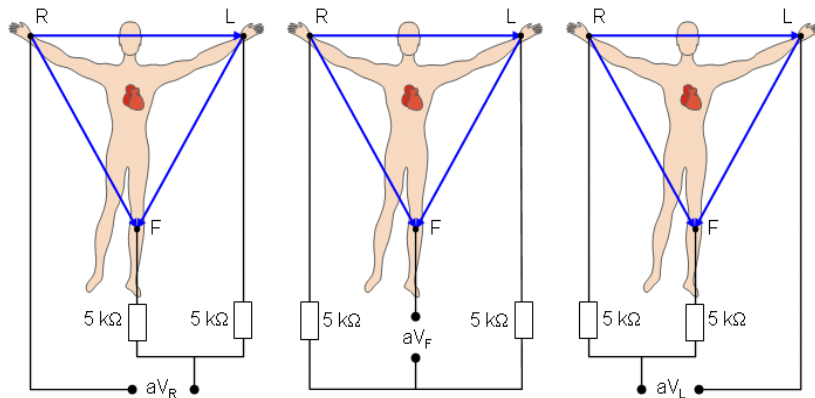
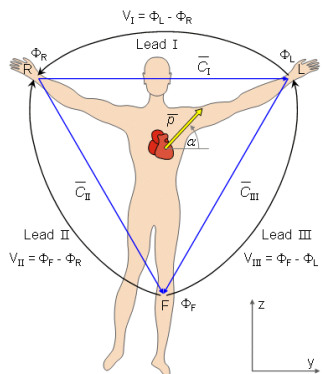
- Entwicklung eines Systems für die Aufnahme, Auswertung und automatische Diagnose verschiedener Arten von EKGs für unterschiedliche kardiologische Untersuchungen und Einsätze.



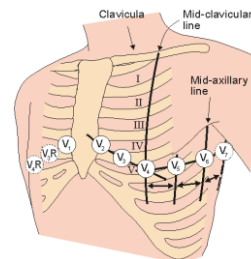
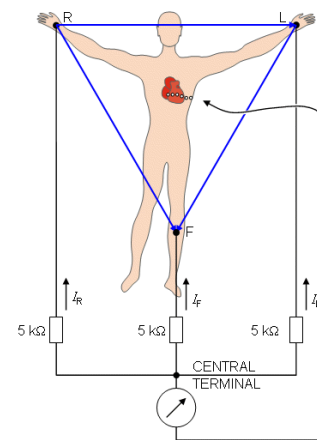
Entstehung eines EKGs (Elektrokardiogramm)



Konventionelles 12-Kanal-EKG „Standardableitungen“



Sechs Ableitungen an den Extremitäten nach Einthoven und Goldberger



Sechs Brustwandableitungen nach Wilson

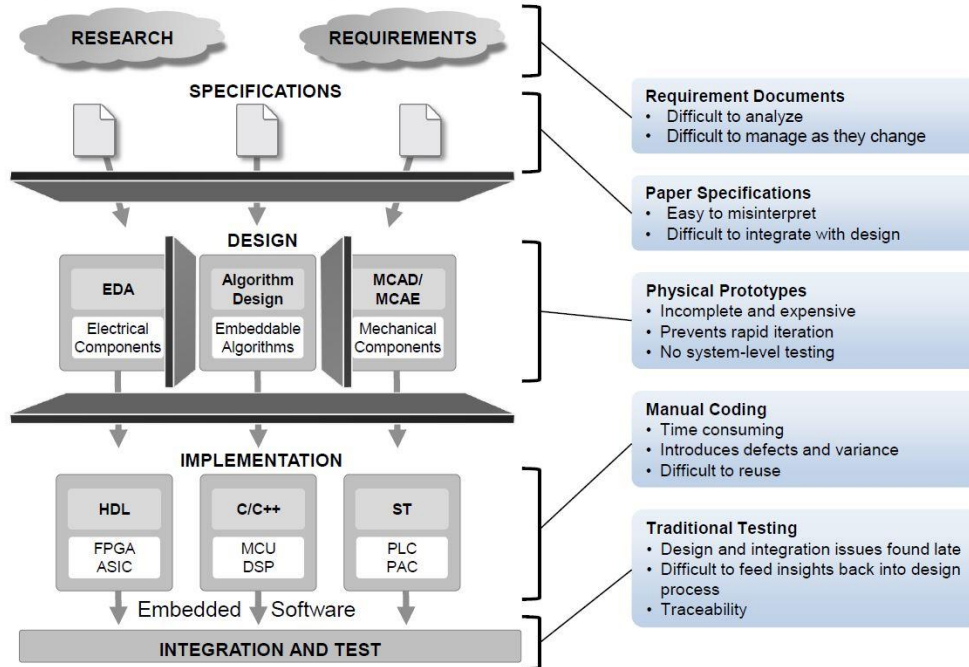
EKG Anwendungs- und Analysearten

- Ruhe-EKG
- Langzeit-EKG Monitoring
- Belastungs-EKG
- Fetales EKG
- Telemetrie EKG
- Homecare EKG
- Notfall EKG



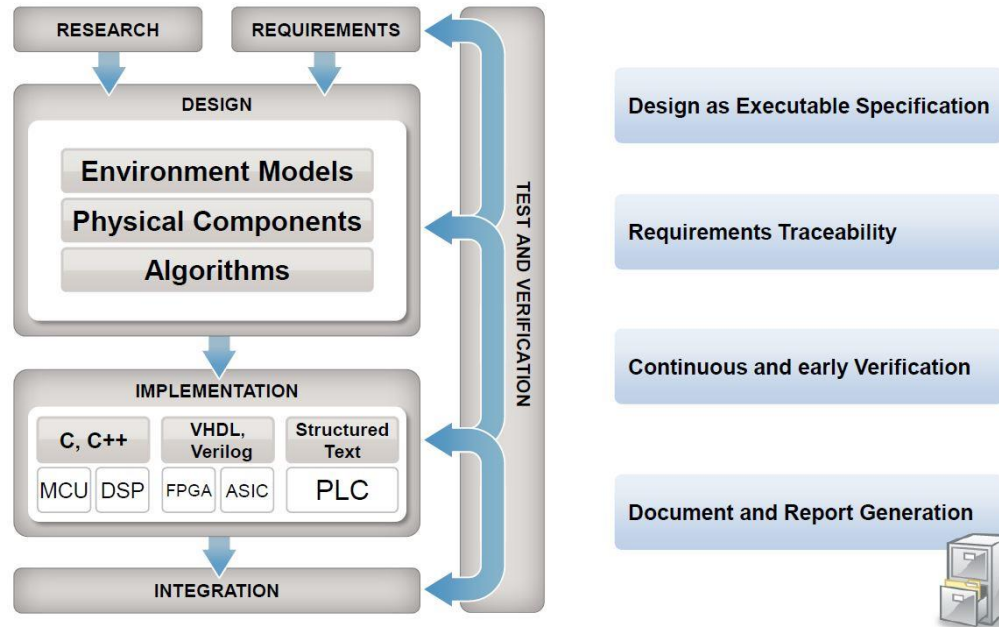
Klassische Entwicklungsmethode

Typical Challenges in Development



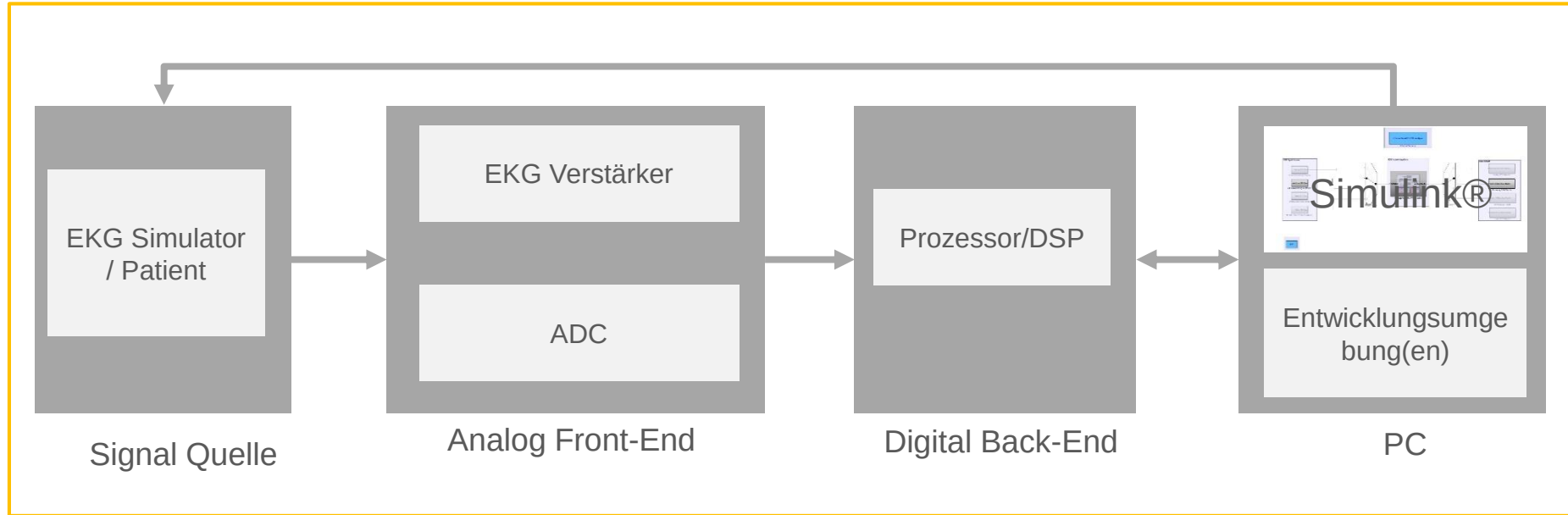
Modellbasiertes Design (MBD) Methode

Model-Based Design



“Model-Based Design is transforming the way engineers and scientists work by moving design tasks from the lab and field to the desktop.”

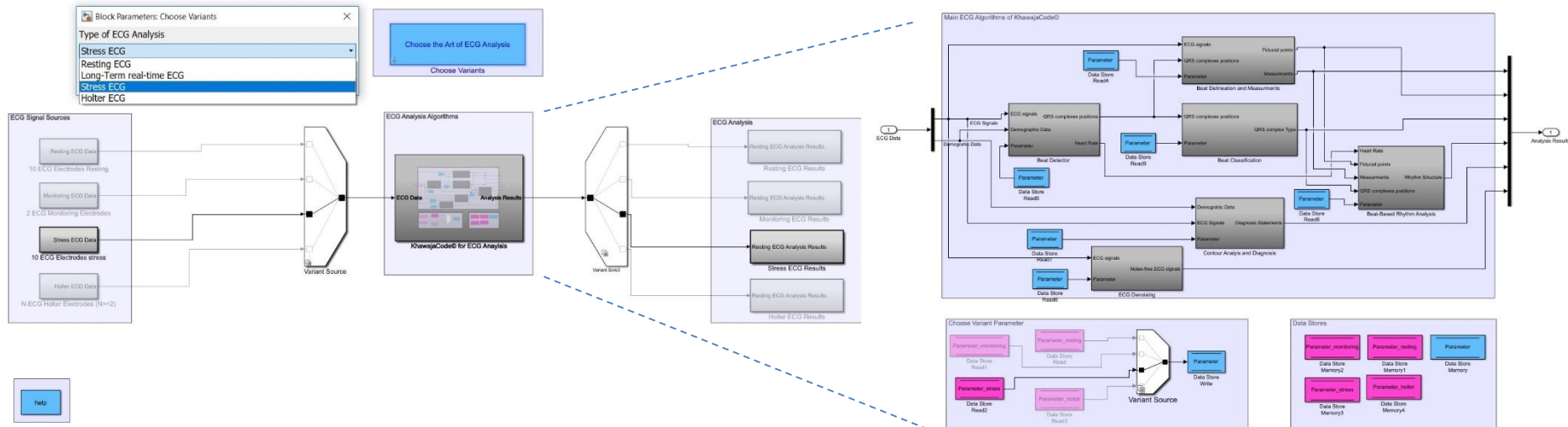
Die gesamte Entwicklungsplattform



Warum Modellbasiertes Design (MBD)?

7 gute Gründe aus eigener Erfahrung : 1. Komplexität meistern

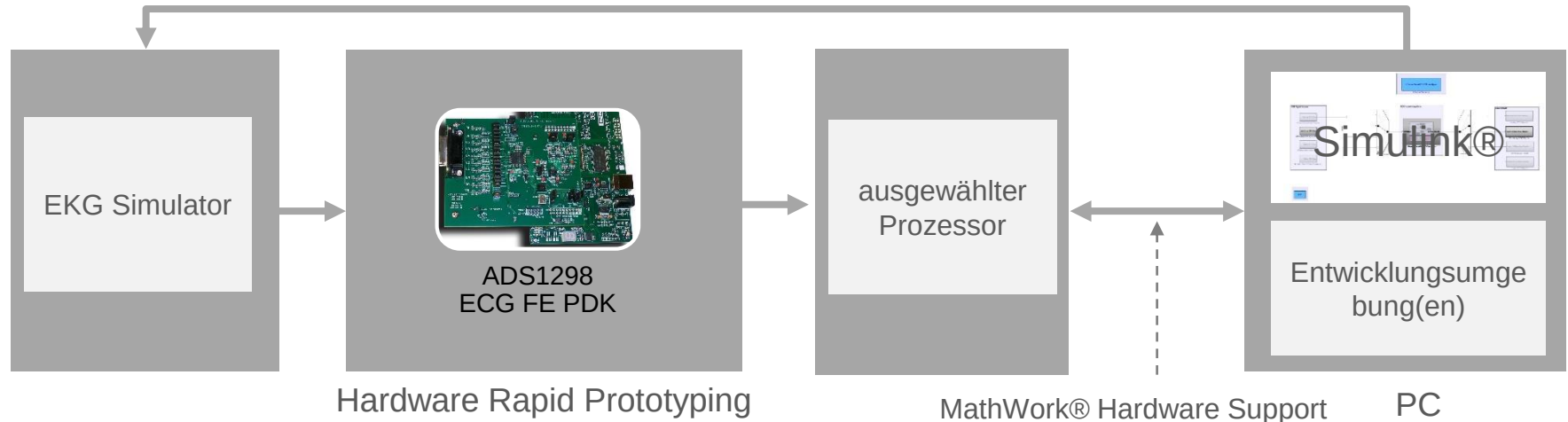
- Durch Nutzung von Simulink® **Variant Subsystems**
- Durch Nutzung von Simulink® **Object-Oriented Design** und **System Objects**
- Klare Software Architektur und Spezifikation durch Simulink® Modelle und Subsysteme
- Anwendung höherer Programmiersprachen durch **Simulink® MATLAB Function** Block
- Anwendung von Legacy Codes durch **Simulink® S-Function** Block



Warum Modellbasiertes Design (MBD)?

7 gute Gründe aus eigenen Erfahrung : 2. Rapid Prototyping

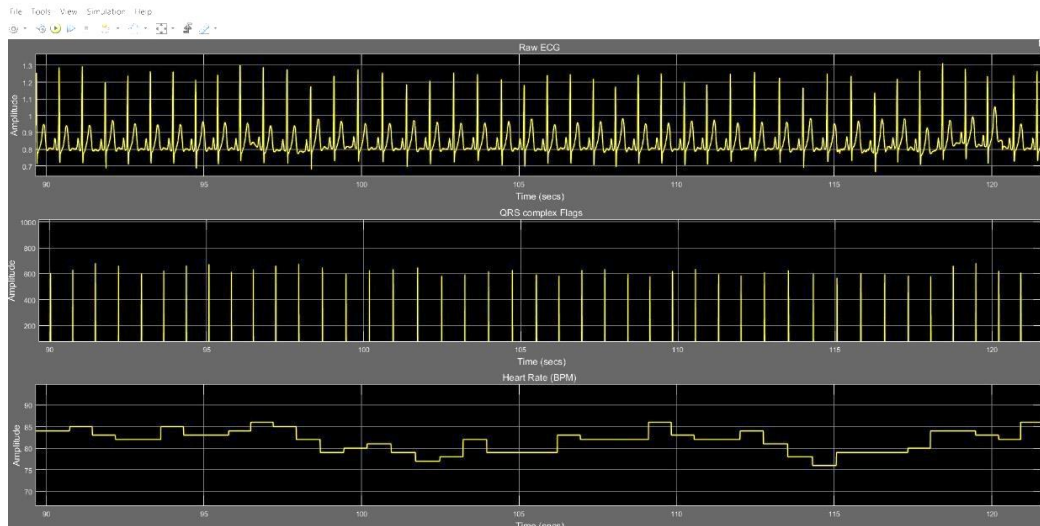
- Unterstützung für third-party hardware und Targets durch **MathWork® Hardware Support**
- Verkürzung der Software-Entwicklungszyklen über 60 %: Design Iterationen in Manntagen statt Mannmonate durchführen
- Automatische Codegenerierung für MATLAB/Simulink Modellen (Generisch und Target-spezifisch)
- Mehr Focus auf Innovationen



Warum Modellbasiertes Design (MBD)?

7 gute Gründe aus eigenen Erfahrung : 3. Hohe Effizienzsteigerung

- Simulation, Optimierung und Testen von Algorithmen vor der Implementierung
- Entwicklungskosten reduzieren durch frühzeitige Fehlererkennung
- Aufwände für die gesamte Entwicklung, Änderungsmanagement, Wartung und Dokumentationen erheblich reduzieren



Warum Modellbasiertes Design (MBD)?

7 gute Gründe aus eigenen Erfahrung : 4. Rückverfolgbarkeit

- Rückverfolgbarkeit des gesamten Produktlebenslaufs
- Rückverfolgbarkeit zwischen dem Modell und den übergeordneten Anforderungen
- Requirements Management Interface zur Rückverfolgung von Modellobjekten und generiertem Code zu den Anforderungsdokumenten

The screenshot displays the KhawajaCodeTopModel software interface, which is used for model-based design and code generation. The interface is divided into several panes:

- Requirements Traceability Report for KhawajaCodeTopModel:** This pane shows a hierarchical view of requirements, including model information, traceability, and system requirements.
- Model Information for "KhawajaCodeTopModel":** This pane provides details about the model, including its name, version, and the user who created it.
- System - Beat Delineation and Measurements:** This pane shows a diagram of the system, including components like "Beat Delineation and Measurements" and "QRS complex position".
- Code Generation Report:** This pane shows the generated code, including the "Main file" (ert_main.c), "Model files" (Beat.c, Beat.h, Beat_private.h, Beat_types.h), and "Utility files" (1).

The interface also includes a "Simulink" pane at the bottom, which shows the Simulink model and its associated code.

Warum Modellbasiertes Design (MBD)?

7 gute Gründe aus eigenen Erfahrung : 5. Entwicklungsbegleitende V&V

- Mehr Sicherheit und Funktionale Sicherheit bei Risiken trotz wachsende Komplexität durch kontinuierliche Tests
- Iterative Modell Verifikation mit Hilfe Simulink® **Model Advisor**
- Iterative Optimierung der Codegenerierung mit Hilfe Simulink® **Code Generation Advisor**

The image displays three overlapping windows from the Simulink environment, illustrating the Model Advisor and Code Generation Advisor tools.

- Model Advisor - KhawajaCodeTopModel:Stress ECG Results:** This window shows a summary of model checks. The 'Run Summary' indicates 151 Passed, 0 Failed, and 0 Warning. The 'Filter checks' pane on the left lists various checks, including '1.1 Simulink', '1.2 Simulink Coder', '1.3 Embedded Coder', '1.4 Simulink Verification and Validation', '1.5 EC Certification Kit', '1.6 Managing Data Store Memory Blocks', '1.7 Managing Library Links and Variables', '1.8 Migrating to Simplified Initialization mode', '1.9 Model Metrics', '1.10 Model Referencing', '1.11 Modeling Standards', '1.12 MathWorks Automotive Advisory Board Checks', '1.13 Model Metrics', '1.14 Requirements Consistency', and '1.15 EC Certification Kit'. The 'Model Advisor' pane on the right provides a detailed view of the 'Verify model complies with modeling guidelines' check, including a list of tips and a legend for the status icons (Not Run, Passed, Failed, Warning).
- Code Generation Advisor - KhawajaCodeTopModel_codegenation:** This window shows the configuration and results of code generation. The 'Code Generation Advisor' pane on the left lists various checks, including '1. Check model configuration settings against code generation objectives', '2. Identify unconnected lines, input ports, and output ports', '3. Check for optimal bus virtuality', '4. Check Data Store Memory blocks for multitasking, strong typing, and shadowing issues', '5. Identify black output signals with continuous sample time and non-floating point data type', '6. Check the hardware implementation', '7. Identify questionable software environment specifications', '8. Identify questionable code instrumentation (data I/O)', '9. Check for blocks that have constraints on tunable parameters', '10. Identify questionable subsystem settings', '11. Identify blocks that generate expensive rounding code', '12. Identify questionable fixed-point operators', '13. Identify blocks using one-based indexing', '14. Identify loop table blocks that generate expensive out-of-range checking code', '15. Check output types of logic blocks', and '16. Identify blocks that generate expensive fixed-point and saturation code'. The 'Code Generation Objectives' pane on the right shows the selected objectives, including 'MDIRA C.2012 guidelines' and 'Polyspace'. The 'Run Selected Checks' button is visible at the bottom.

Warum Modellbasiertes Design (MBD)?

7 gute Gründe aus eigenen Erfahrung : 5. Entwicklungsbegleitende V&V

➤ „Software in the Loop“ (SIL)

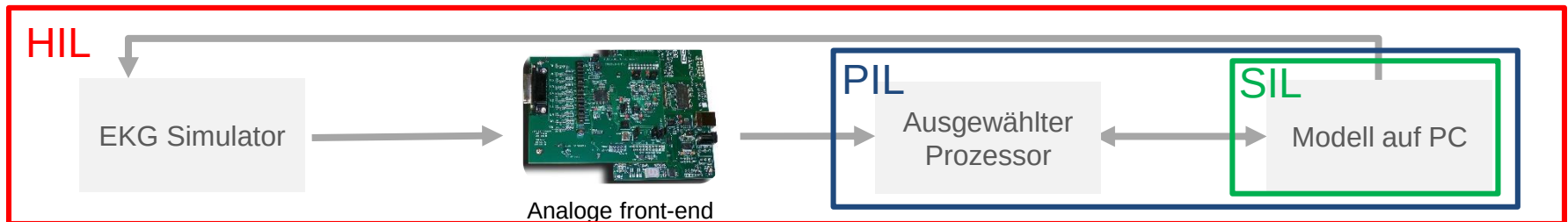
- Code aus dem Modell generieren und auf dem *PC* Ausführung
- Das Verhalten gegen das Simulink® Modell verifizieren

➤ „Processor in the Loop“ (PIL)

- Code aus dem Modell generieren und auf dem *Zielformat (Prozessor)* Ausführung
- Das Verhalten gegen SIL verifizieren: anderes Verhalten als bei SIL möglich
- Zielformat und zugehörige Toolchains notwendig

➤ „Hardware in the Loop“ (HIL)

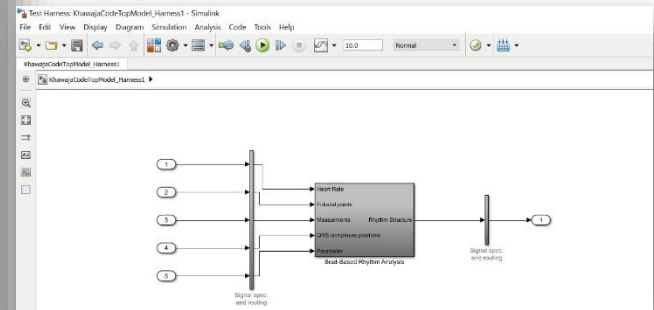
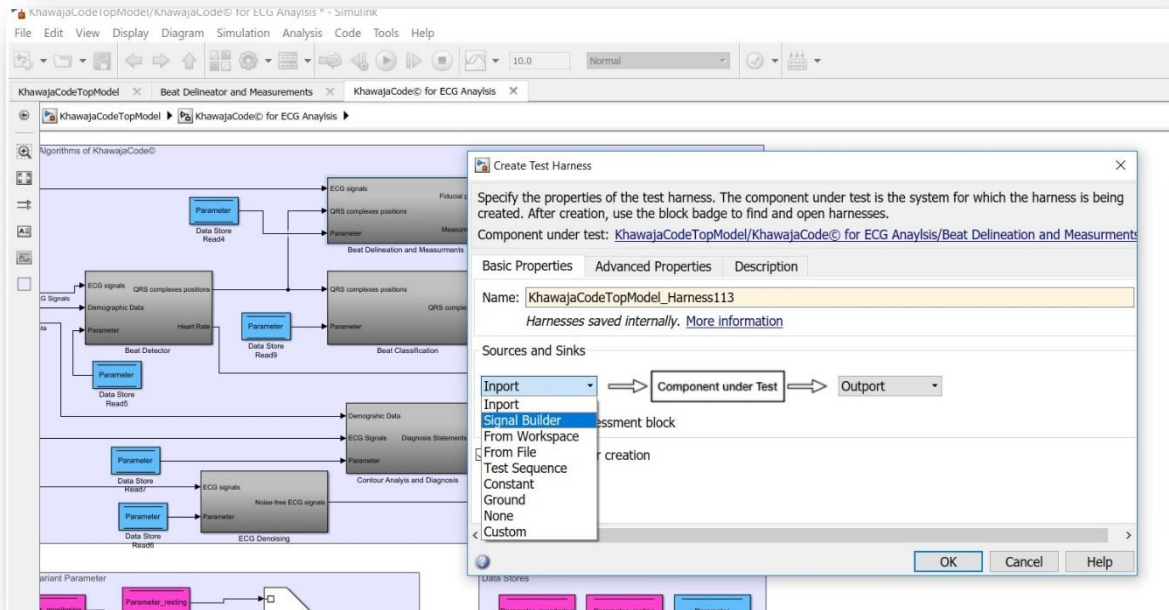
- Code aus dem Modell generieren und auf dem kompletten *System* Ausführung
- Das Verhalten gegen PIL verifizieren: anderes Verhalten als bei PIL möglich wegen Analoge Hardware Teile
- Komplette Zielhardware, Patienten Simulator und zugehörige Toolchains notwendig



Warum Modellbasiertes Design (MBD)?

7 gute Gründe aus eigenen Erfahrung : 5. Entwicklungsbegleitende V&V

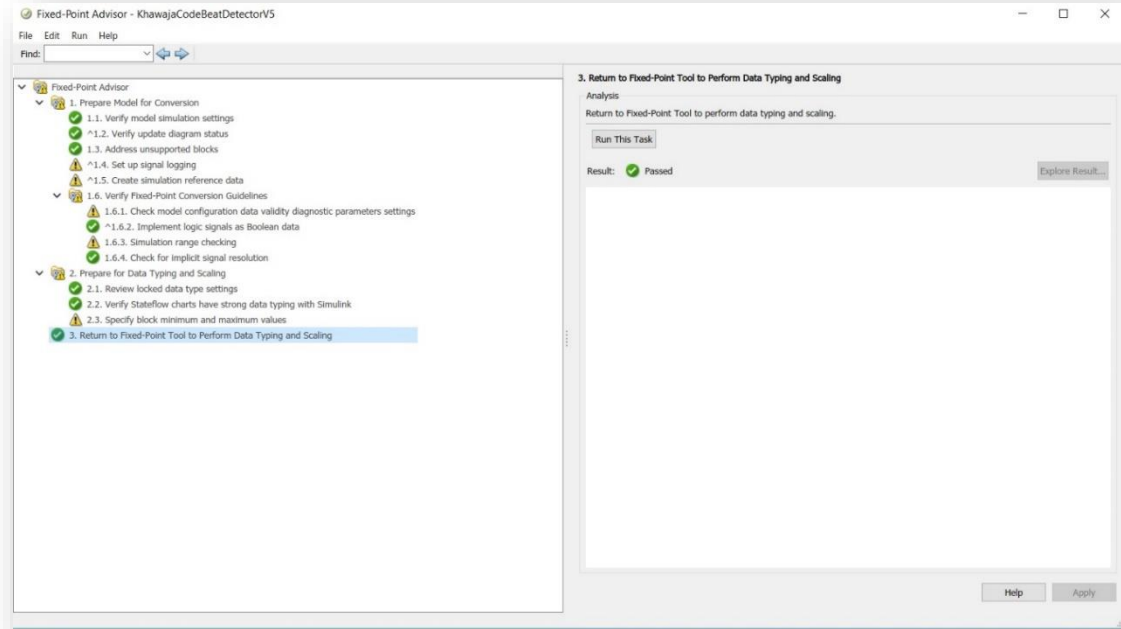
- Testen von Modellen und Subsystemen durch Testharnesse
- Testen durch Testharnesse ist jederzeit wiederholbar
- Nicht geprüfte Elemente in Modell werden mit Hilfe Simulink® **Model coverage** und **Code coverage** lokalisiert



Warum Modellbasiertes Design (MBD)?

7 gute Gründe aus eigenen Erfahrung : 6. Unterstützung Festkomma-Arithmetik

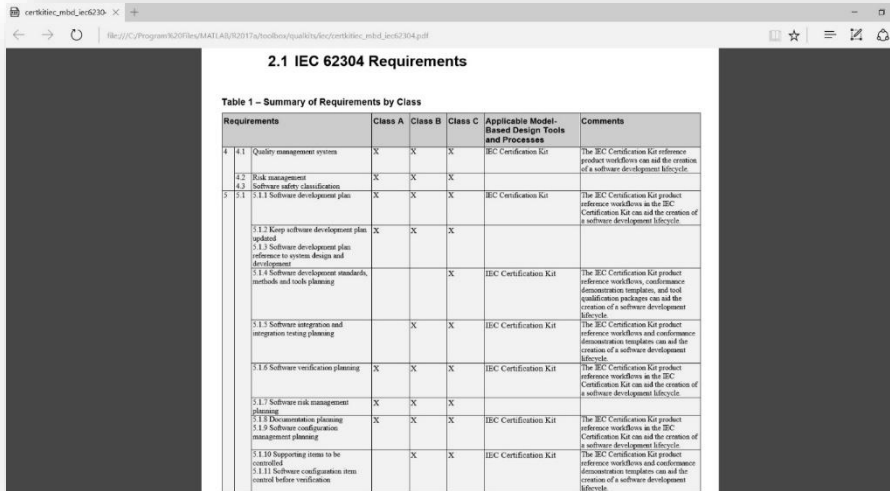
- Festkomma-Datentypen in MATLAB, Simulink und Stateflow
- Automatisierte Umwandlung für das Modell von Gleitkomma zu Festkomma
- Optimale Vorbereitung auf die Automatisierte Umwandlung durch Simulink® **Fixed-Point Advisor**
- Unterstützung Codegenerierung



Warum Modellbasiertes Design (MBD)?

7 gute Gründe aus eigenen Erfahrung : 7. Hohe Qualität

- Hohe Qualität trotz wachsende Anforderungen an funktionale Sicherheit
- Von-TÜV-SÜD-genehmigte Referenz-Workflow für die IEC 62304 (Norm für Medizinische Software-Lebenszyklus)
- Automatische Berichte Generierung und dadurch weniger Dokumentationsaufwand und mehr Zeit für Innovationen
- Erstellung Entwicklungs- und Zulassungsbezogene Berichte jederzeit möglich



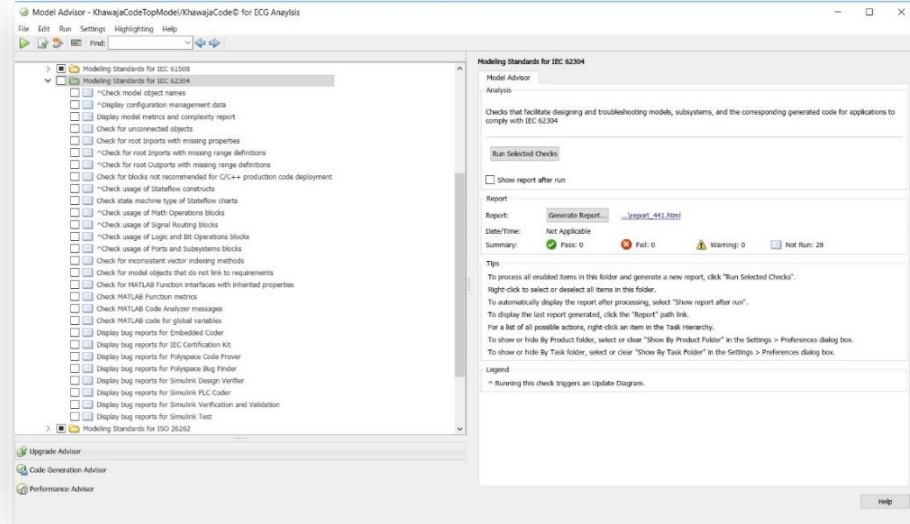
certklicc_mbd_iec62304

file:///C:/Program%20Files/MATLAB/R2017a/toolbox/quality/iec/certklicc_mbd_iec62304.pdf

2.1 IEC 62304 Requirements

Table 1 – Summary of Requirements by Class

Requirements	Class A	Class B	Class C	Applicable Model-Based Design Tools and Processes	Comments
4.1 Quality management system	X	X	X	IEC Certification Kit	The IEC Certification Kit reference product workflows can aid the creation of a software development lifecycle.
4.2 Risk management	X	X	X		
4.3 Software safety classification	X	X	X		
5.1.1 Software development plan	X	X	X	IEC Certification Kit	The IEC Certification Kit product reference workflows in the IEC Certification Kit can aid the creation of a software development lifecycle.
5.1.2 Keep software development plan updated	X	X	X		
5.1.3 Software development plan reference to system design and development					
5.1.4 Software development standards, methods and tool planning			X	IEC Certification Kit	The IEC Certification Kit product reference workflows, confidence demonstration templates, and tool qualification packages can aid the creation of a software development lifecycle.
5.1.5 Software integration and integration testing planning		X	X	IEC Certification Kit	The IEC Certification Kit product reference workflows and confidence demonstration templates can aid the creation of a software development lifecycle.
5.1.6 Software verification planning	X	X	X	IEC Certification Kit	The IEC Certification Kit product reference workflows in the IEC Certification Kit can aid the creation of a software development lifecycle.
5.1.7 Software risk management planning	X	X	X		
5.1.8 Documentation planning	X	X	X	IEC Certification Kit	The IEC Certification Kit product reference workflows in the IEC Certification Kit can aid the creation of a software development lifecycle.
5.1.9 Software configuration management planning					
5.1.10 Supporting items to be controlled		X	X	IEC Certification Kit	The IEC Certification Kit product reference workflows and confidence demonstration templates can aid the creation of a software development lifecycle.
5.1.11 Software configuration item control before verification					



Zusammenfassung & Take-Home Message

		Klassische Entwicklungsmethode	MBD mit Simulink®
Komplexität meistern			
Rapid Prototyping			
Hohe Effizienzsteigerung			
Rückverfolgbarkeit	*		
Entwicklungsbegleitende V & V	*		
Unterstützung Festkomma Arithmetik	*		
Hohe Qualität	*		

* Mit sehr hohe Aufwand verbunden

Klassische Entwicklungsmethode ist **OLD** Standard

Entwicklung von Medizinischen Algorithmen für die Kardiologie mit Hilfe Simulink® und Modellbasiertes Design

Danke für die Aufmerksamkeit

Fragen?



Dr.-Ing. Antoun Khawaja

CEO KhawajaHealth

27.06.2017, MATLAB EXPO 2017 München Deutschland