

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



**Dr.-Ing. Antoun Khawaja**

**CEO KhawajaHealth**

[Antoun.Khawaja@KhawajaHealth.com](mailto:Antoun.Khawaja@KhawajaHealth.com)

[Antoun.Khawaja@Corpuls.com](mailto:Antoun.Khawaja@Corpuls.com)

27.06.2017, MATLAB EXPO 2017 München Deutschland

# Agenda

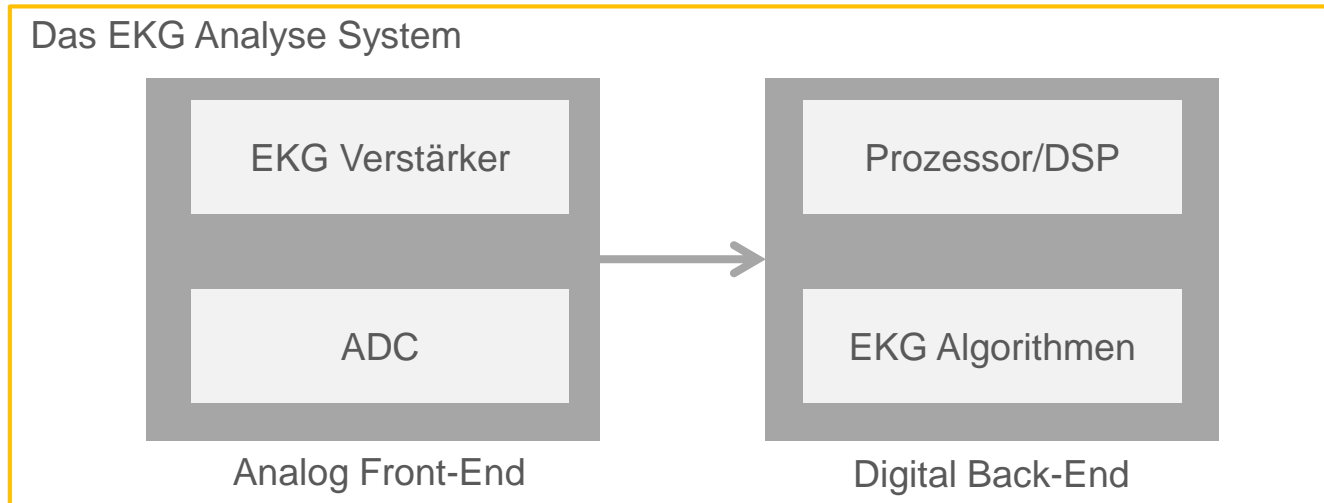
- Einführung
- Ziel der Arbeit
- Hintergrund Kardiologie
- Klassische und Modellbasierte Entwicklung
- Die gesamte Entwicklungsplattform
- Warum Modellbasiertes Design (MBD)? Sieben gute Gründe...
- Zusammenfassung & Take-Home Message

# Einführung

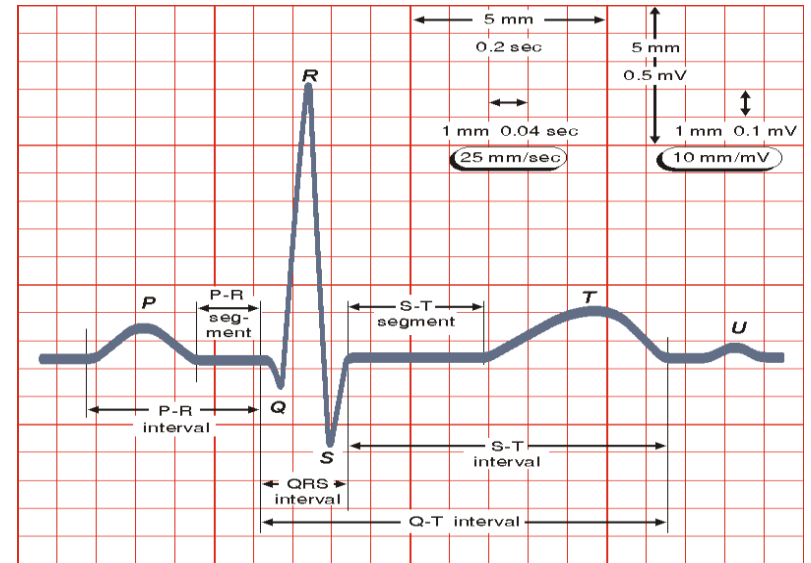
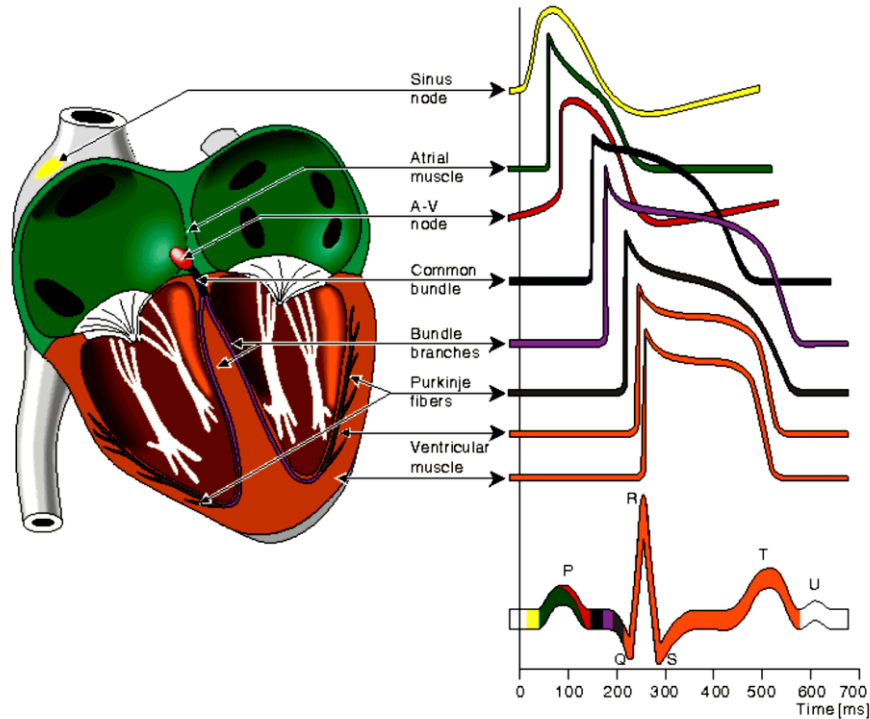
- Herz-Kreislauf Erkrankungen ist eine der häufigsten Todesursache
- Das Elektrokardiogramm (EKG) ist die zentrale Untersuchungsmethode für die Diagnostik und die Überwachung von Patienten mit Herz-Kreislauf-Erkrankungen, Herzrhythmusstörungen, oder chronischer Herzinsuffizienz.
- Kardiologische Medizingeräte 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

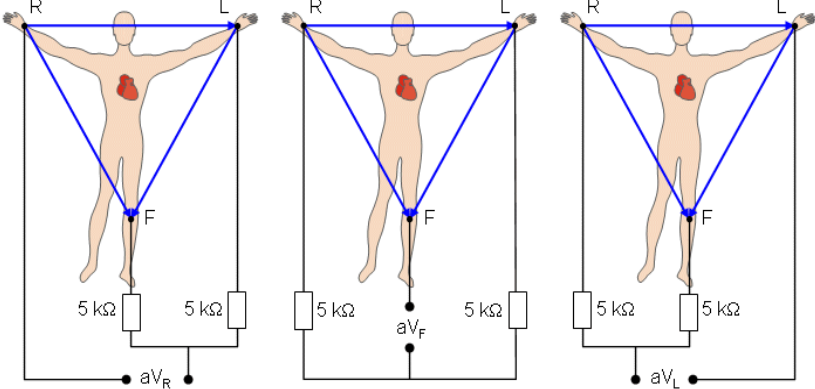
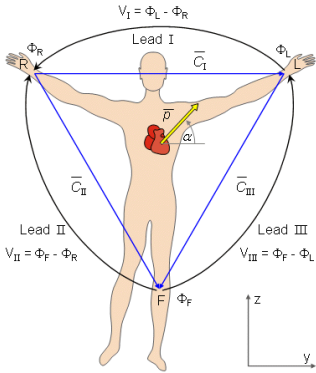
- Entwicklung eines Systems für die Aufnahme, Auswertung und automatische Diagnose verschiedener Arten von EKGs für unterschiedliche kardiologische Untersuchungen und Einsätze.



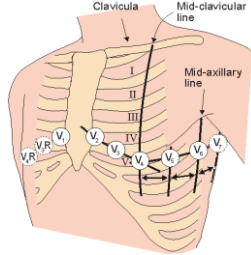
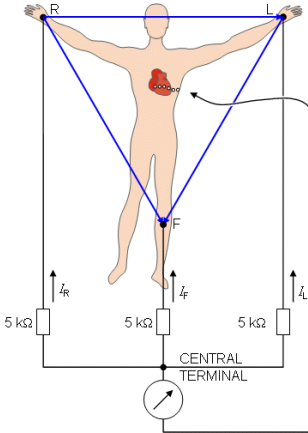
# Entstehung eines EKGs (Elektrokardiogramm)



# Konventionelles 12-Kanal-EKG „Standardableitungen“



Sechs Ableitungen an den Extremitäten nach Einthoven und Goldberger



Sechs Brustwandableitungen nach Wilson

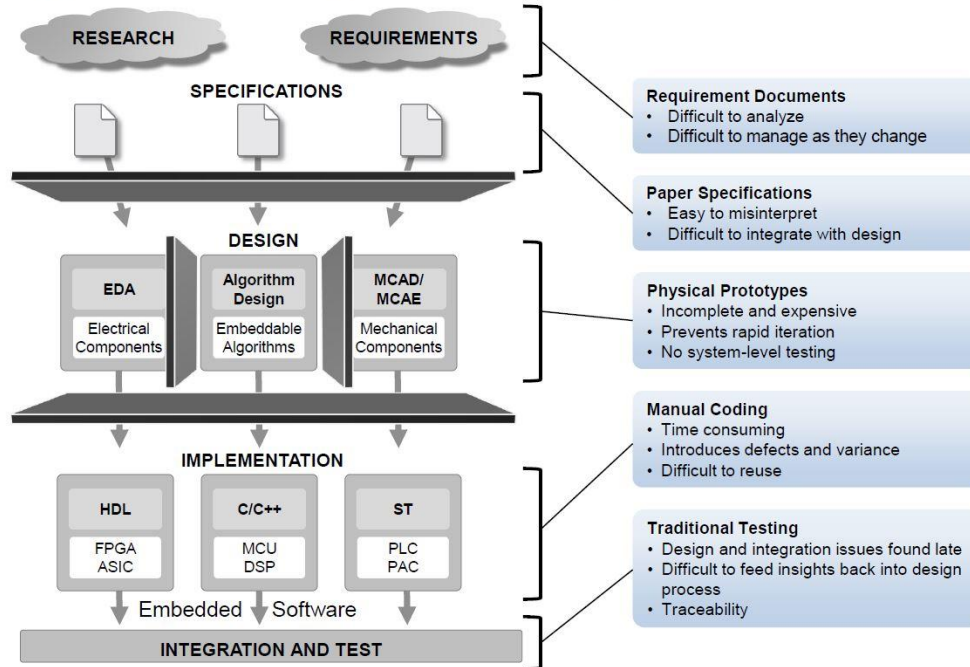
# EKG Anwendungs- und Analysearten

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



# Klassische Entwicklungsmethode

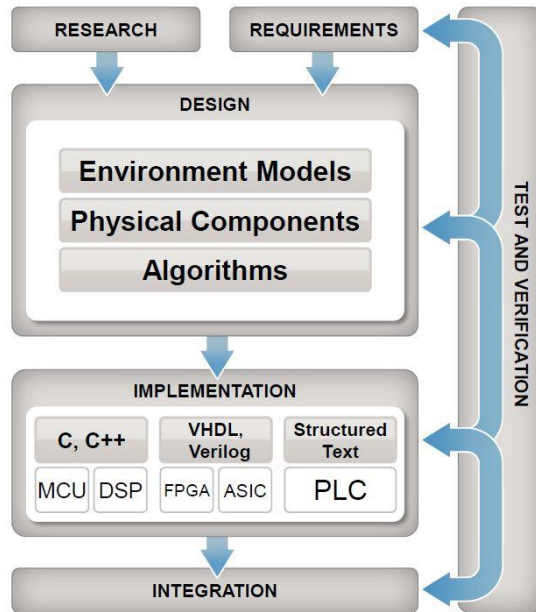
## Typical Challenges in Development





# Modellbasiertes Design (MBD) Methode

## Model-Based Design



Design as Executable Specification

Requirements Traceability

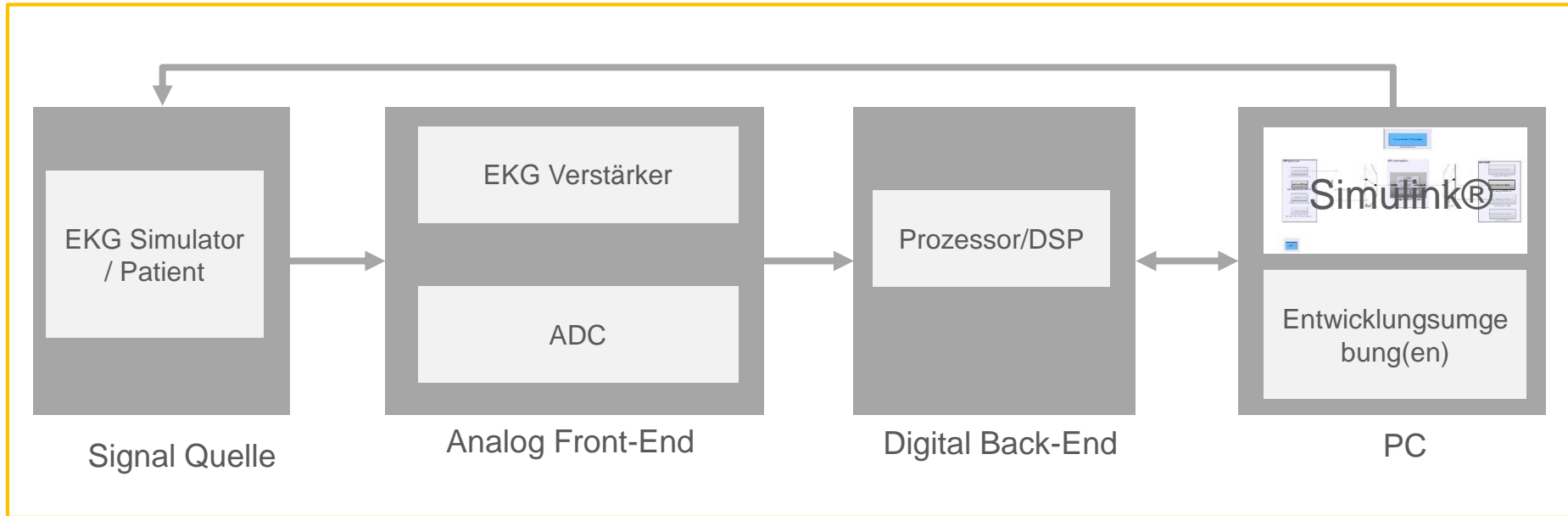
Continuous and early Verification

Document and Report Generation



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

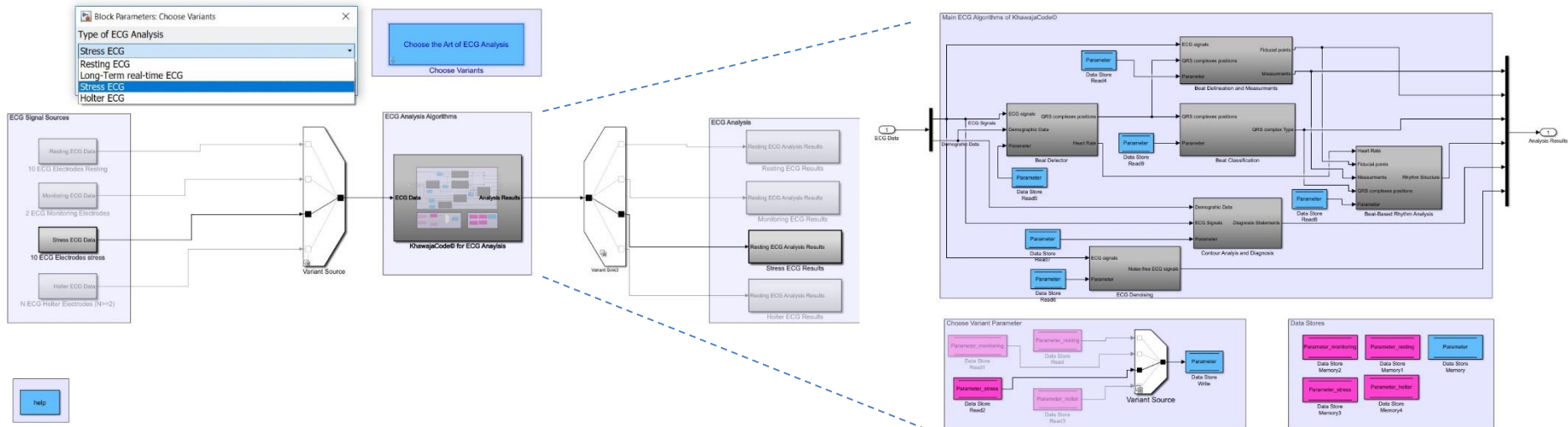
# Die gesamte Entwicklungsplattform



# Warum Modellbasiertes Design (MBD)?

## 7 gute Gründe aus eigener Erfahrung : 1. Komplexität meistern

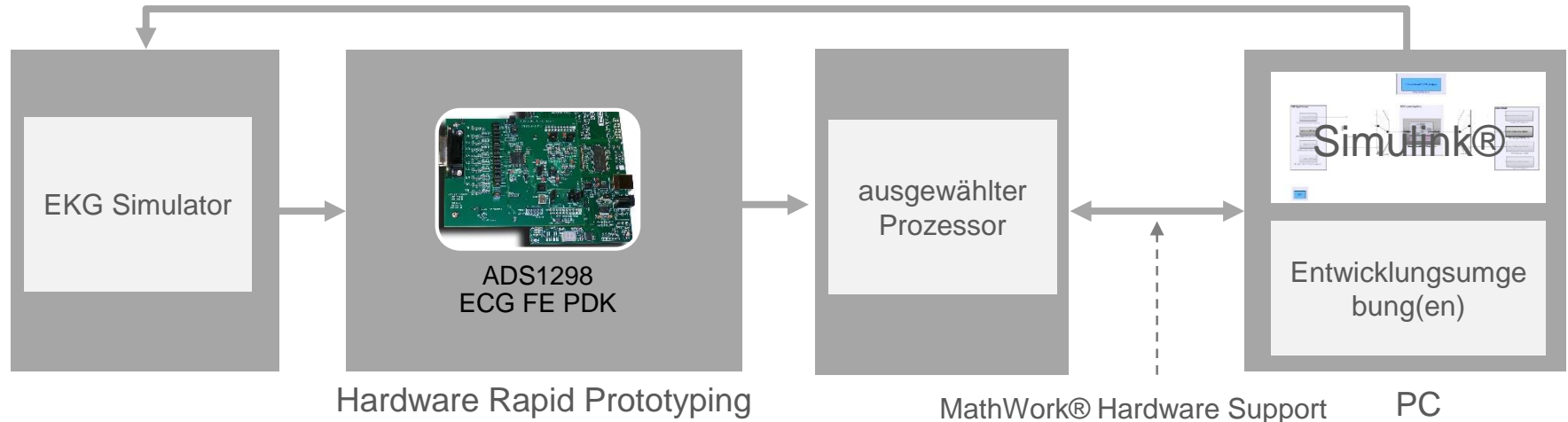
- Durch Nutzung von Simulink® **Variant Subsystems**
- Durch Nutzung von Simulink® **Object-Oriented Design** und **System Objects**
- Klare Software Architektur und Spezifikation durch Simulink® Modelle und Subsysteme
- Anwendung höherer Programmiersprachen durch **Simulink® MATLAB Function** Block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



# Warum Modellbasiertes Design (MBD)?

## 7 gute Gründe aus eigenen Erfahrung : 2. Rapid Prototyping

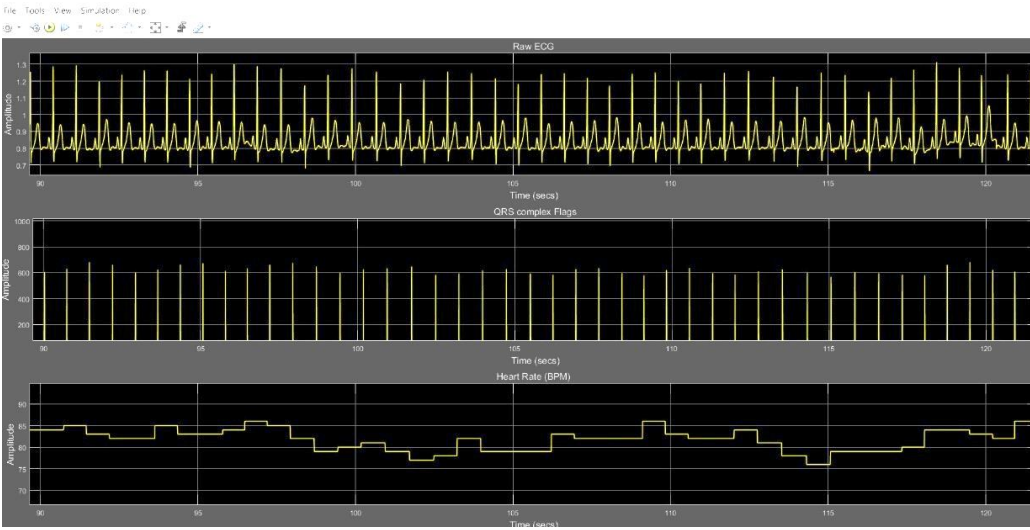
- Unterstützung für third-party hardware und Targets durch **MathWork® Hardware Support**
- Verkürzung der Software-Entwicklungszyklen über 60 %: Design Iterationen in Manntagen statt Mannmonate durchführen
- Automatische Codegenerierung für MATLAB/Simulink Modellen (Generisch und Target-spezifisch)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

**Requirements Traceability Report for KhawajaCodeTopModel**

Location: file:///E:/KhawajaCodeTopModel/requirements.html

**Inhaltsebenen:**

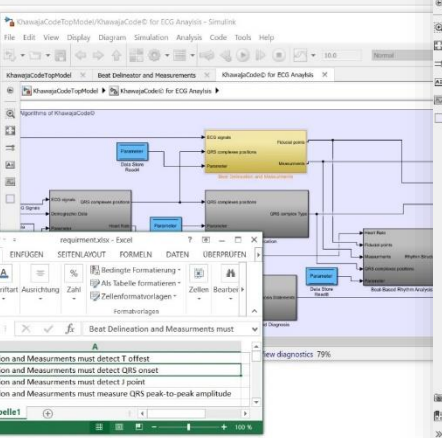
1. Model Information for "KhawajaCodeTopModel"
2. System - Beat Delineation and Measurements
3. System - Beat Delineation and Measurements

**Table Information for "KhawajaCodeTopModel"**

ID	Text	Source
1	Beat Delineation and Measurements must detect T offset	Req-001
2	Beat Delineation and Measurements must detect QRS onset	Req-002
3	Beat Delineation and Measurements must detect T point	Req-003
4	Beat Delineation and Measurements must measure QRS peak-to-peak amplitude	Req-004

**Table 1.1. KhawajaCodeTopModelKhawajaCode for ECG Analysis Beat Delineation and Measurements Requirements Traceability Data**

ReqID	Req Description	Req Target (element name and location ID)
1	Beat Delineation and Measurements must detect T offset	KhawajaCodeTopModelKhawajaCode for ECG Analysis Beat Delineation and Measurements Requirements Traceability Data
2	Beat Delineation and Measurements must detect QRS onset	KhawajaCodeTopModelKhawajaCode for ECG Analysis Beat Delineation and Measurements Requirements Traceability Data
3	Beat Delineation and Measurements must detect T point	KhawajaCodeTopModelKhawajaCode for ECG Analysis Beat Delineation and Measurements Requirements Traceability Data
4	Beat Delineation and Measurements must measure QRS peak-to-peak amplitude	KhawajaCodeTopModelKhawajaCode for ECG Analysis Beat Delineation and Measurements Requirements Traceability Data



**Code Generation Report**

Contents:

- Summary
- Subsystem Report
- Code Interface Report
- Traceability Report
- Static Code Metrics Report
- Code Replacements Report

**Generated Code**

- [-] Main file
  - ert\_main.c
- [-] Model files
  - Beat.c
  - Beat.h
  - Beat\_private.h
  - Beat\_types.h
- [-] Utility files (1)
  - Match Case

```
63 /*
64 */
65 * The example "main" function illustrates what is required by your
66 * application code to initialize, execute, and terminate the generated code.
67 * Attaching rt_OneStep to a root-clone block is target specific. This example
68 * illustrates how you do this relative to initializing the model.
69 */
70 int_1_main(int_1_argc, const char *arg1[])
71 {
72     /* Unused arguments */
73     (void)argc;
74     (void)arg1;
75
76     /* Initialize model */
77     Beat_initialize();
78
79     /* Attach rt_OneStep to a timer or interrupt service routine with
80      * period of 2 seconds (the model's base sample time) here. The
81      * call syntax for rt_OneStep is
82      *
83      * rt_OneStep();
84      */
85     printf("Warning: The simulation will run forever. "
86           "Generated RTW main won't simulate model step behavior. "
87           "To change this behavior select the 'MAT-file logging' option.\n");
88     fflush(NULL);
89     while (rtwGetErrorStatus(Beat_M) == (NULL)) {
90         /* perform other application tasks here */
91     }
92     /* Disable rt_OneStep() here */
93 }
94
```

# 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**

Model Advisor Report - KhawajaCodeTopModel/ Stress ECG Results

Model Advisor - KhawajaCodeTopModel/ Stress ECG Results

Model Advisor - KhawajaCodeTopModel/ Stress ECG Results

Verify model complies with modeling guidelines.

Legend

- Not Run
- Passed
- Failed
- Warning
- A warning this check triggers an update diagram.

Code Generation Advisor - KhawajaCodeTopModel/ codegeneration

Code Generation Advisor

Code Generation Objectives (System target file: art.tlc)

Available objectives

- MDIRA C.2012 guidelines
- PlaySpace

Selected objectives - prioritized

- Execution efficiency
- ROM efficiency
- RAM efficiency
- Traceability
- Safety precaution Debugging

Run Selected Checks

Show report after run

Report

Report: [\\_inprot\\_954.html](#) Save As...

Date/Time: 28-May-2017 18:18:11

Summary: ✔ Pass: 8 ✘ Fail: 3 ⚠ Warning: 6 ⏸ Not Run: 0

Tips

To process all enabled items in this folder and generate a new report, click "Run Selected Checks". Right-click to select or deselect all items in this folder.

To automatically display the report after processing, select "Show report after run".

To display the last report generated, click the "Report" path link.

For a list of all available actions, right-click an item in the Task Hierarchy.

To show or hide By Product folder, select or clear "Show By Product Folders" in the Settings > Preferences dialog box.

# Warum Modellbasiertes Design (MBD)?

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

### ➤ „Software in the Loop“ (SIL)

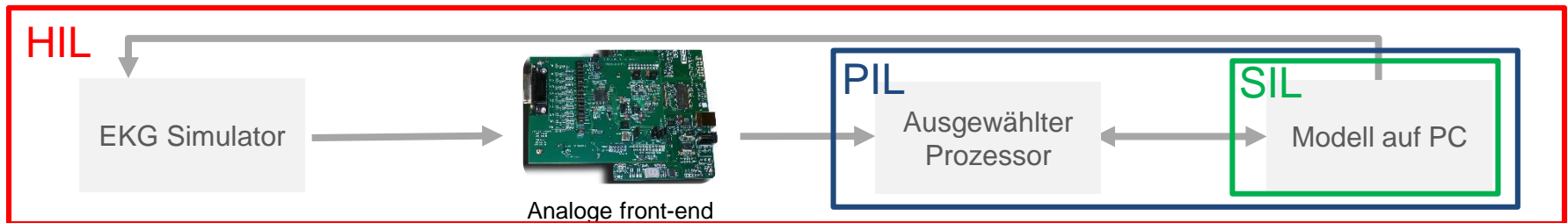
- Code aus dem Modell generieren und auf dem *PC* Ausführung
- Das Verhalten gegen das Simulink® Modell verifizieren

### ➤ „Processor in the Loop“ (PIL)

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

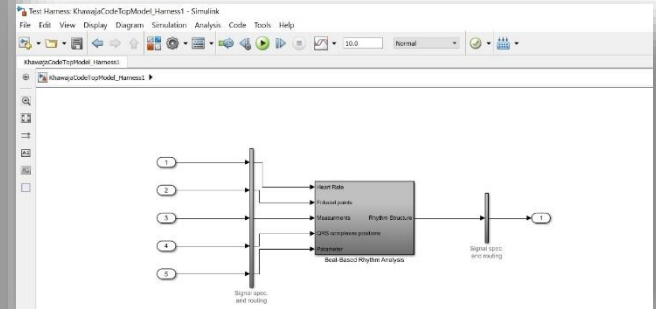
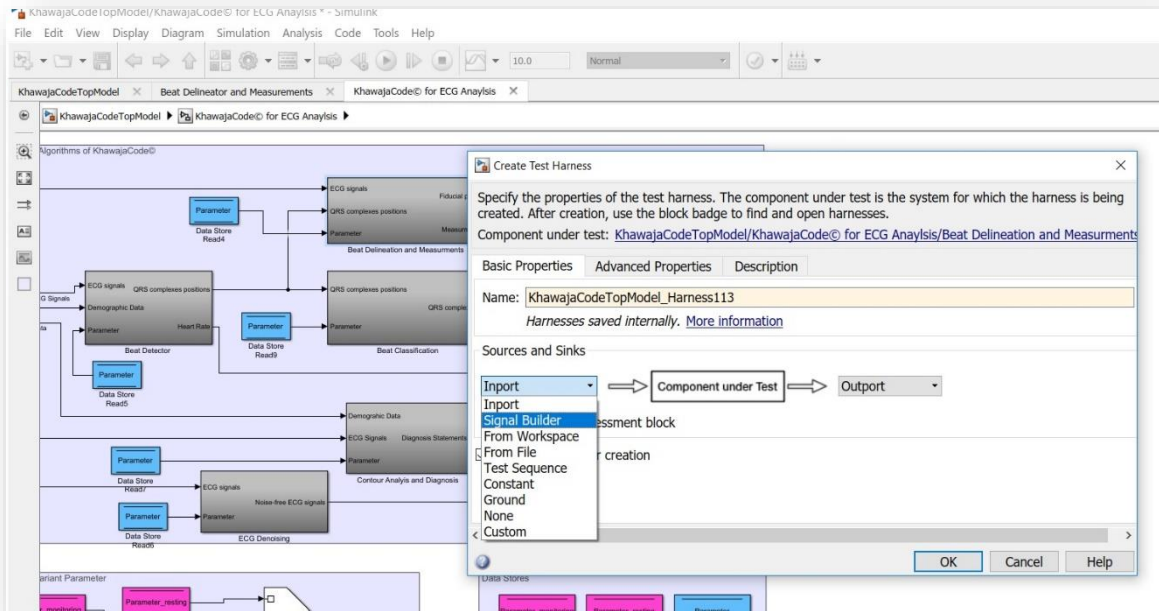




# Warum Modellbasiertes Design (MBD)?

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

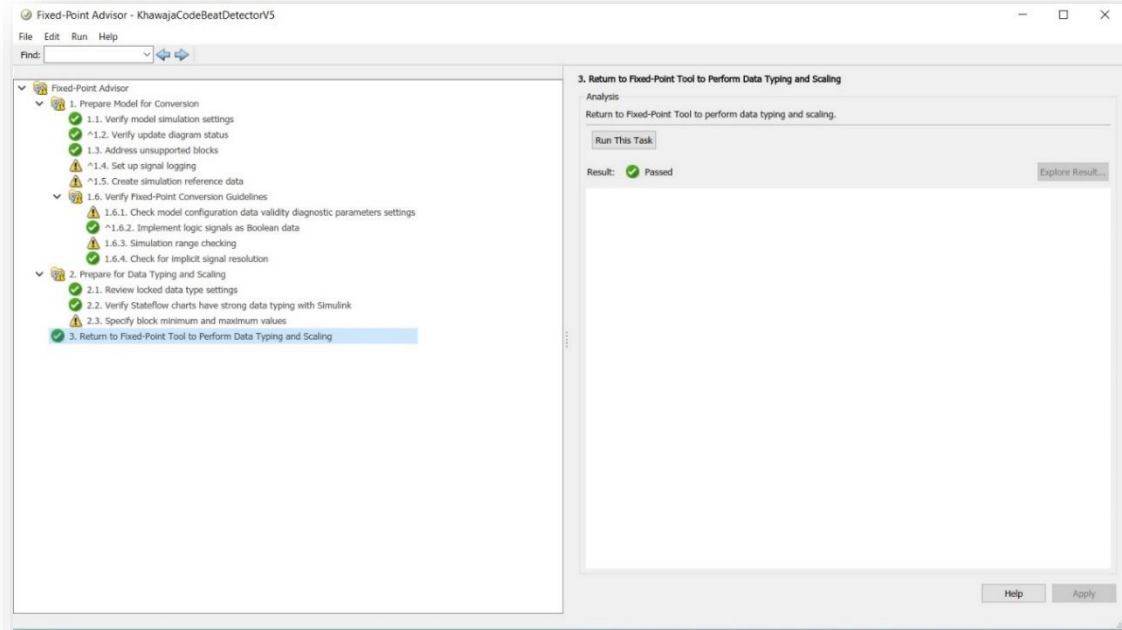
- Testen von Modellen und Subsystemen durch Testharnesse
- Testen durch Testharnesse ist jederzeit wiederholbar
- Nicht geprüfte Elemente in Modell werden mit Hilfe Simulink® **Model coverage** und **Code coverage** lokalisiert



# 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

certklic\_mbd\_ied2304

file:///C:/Program%20Files/MATLAB/R2017a/toolbox/qual/iec/certklic\_mbd\_ied2304.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	X	X	X		
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	X	X	X		
5.1.10 Supporting items to be controlled	X	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	X	X	X		

Model Advisor - KhawajaCodeTopModel\KhawajaCode@ for ECG Analysis

Modeling Standards for IEC 62304

- Check model object names
- Check configuration management data
- Display model metrics and complexity report
- Check for unconnected objects
- Check for root reports with missing properties
- Check for root reports with missing range definitions
- Check for root Outports with missing range definitions
- Check for blocks not recommended for CPU+ production code deployment
- Check usage of Stateflow constructs
- Check state machine type of Stateflow charts
- Check usage of Math Operations blocks
- Check usage of Signal Routing blocks
- Check usage of Logic and Bit Operations blocks
- Check usage of Ports and Subsystems blocks
- Check for inconsistent vector indexing methods
- Check for model objects that do not link to requirements
- Check for MATLAB Function interfaces with inherited properties
- Check MATLAB Function metrics
- Check MATLAB Code Analyzer messages
- Check MATLAB code for global variables
- Display bag reports for Embedded Coder
- Display bag reports for IEC Certification Kit
- Display bag reports for PolySpace Code Prover
- Display bag reports for PolySpace Bug Finder
- Display bag reports for Simulink Design Verifier
- Display bag reports for Simulink PLC Coder
- Display bag reports for Simulink Verification and Validation
- Display bag reports for Simulink Test

Modeling Standards for IEC 62304

Model Advisor

Analysis

Checks that facilitate designing and troubleshooting models, subsystems, and the corresponding generated code for applications to comply with IEC 62304

Run Selected Checks

Show report after run

Report

Generate Report... report\_451.html

Date/Time: Not Applicable















Summary: Pass: 0 Fail: 0 Warning: 0 Not Run: 28

Upgrade Advisor

Code Generation Advisor

Performance Advisor

# 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