

# Model-based Design für medizintechnische Anwendungen

Dr. Momme Winkelkemper



**DR. TÜRCK INGENIEURBÜRO**  
OPTIKENTWICKLUNG UND DATENANALYSE  
tuerck-optik.de



# Projektumfeld

**DR. TÜRCK**  
INGENIEURBÜRO 

Entwicklungsdienstleister

- Algorithmen
- Datenanalyse
- U.a. für Messgeräte-  
Entwicklung

# Projektumfeld

**DR. TÜRCK**  
INGENIEURBÜRO 

Entwicklungsdienstleister

- Algorithmen
- Datenanalyse
- U.a. für Messgeräte-Entwicklung

Auftrag:  
Algorithmus



Medizintechnik-Konzern

- Point-of-Care Messgerät
- Entwicklung nächste Generation
- Neuentwicklung des Mess-Algorithmus



# Projekthalt

## Point-of-Care Messgerät

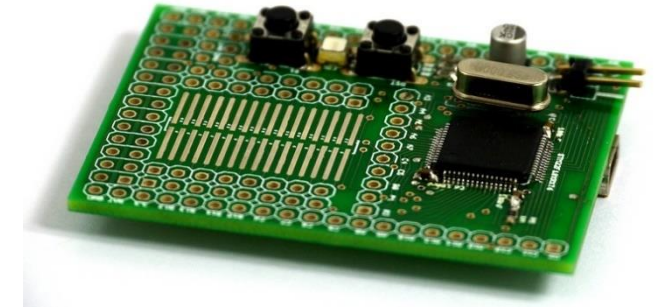
- Medizinisch-technische Analytik in Kundennähe:  
-> z.B. in Arztpraxis, beim Patienten o.ä.
- Häufig auf *Embedded Devices*

## Neuentwicklung Mess-Algorithmus

- Verbesserung Genauigkeit, Messgeschwindigkeit
- Stabilisierung gegen Störeinflüsse

## Randbedingungen

- IEC 62304: „*Medizingeräte-Software - Software-Lebenszyklus-Prozesse*“
- IEC 61508: „*Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme*“



# Projektumfeld

**DR. TÜRCK**  
INGENIEURBÜRO

Entwicklungsdienstleister

- Algorithmen
- Datenanalyse
- U.a. für Messgeräte-Entwicklung



Firmware-Entwicklung

Auftrag:  
Algorithmus

Ergebnis

Auftrag:  
Firmware



Medizintechnik-Konzern

- Point-of-Care Messgerät
- Entwicklung nächste Generation
- Neuentwicklung des Mess-Algorithmus



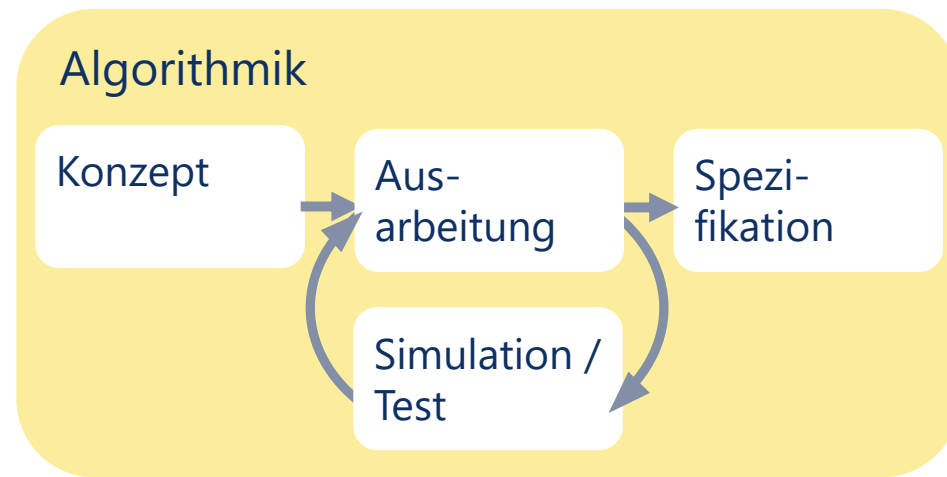
# Inhalt

1. Warum Model-Based Design?
2. Messaufgabe & Algorithmen Entwicklung
3. Umsetzung in Simulink
4. Zusammenfassung

# Inhalt

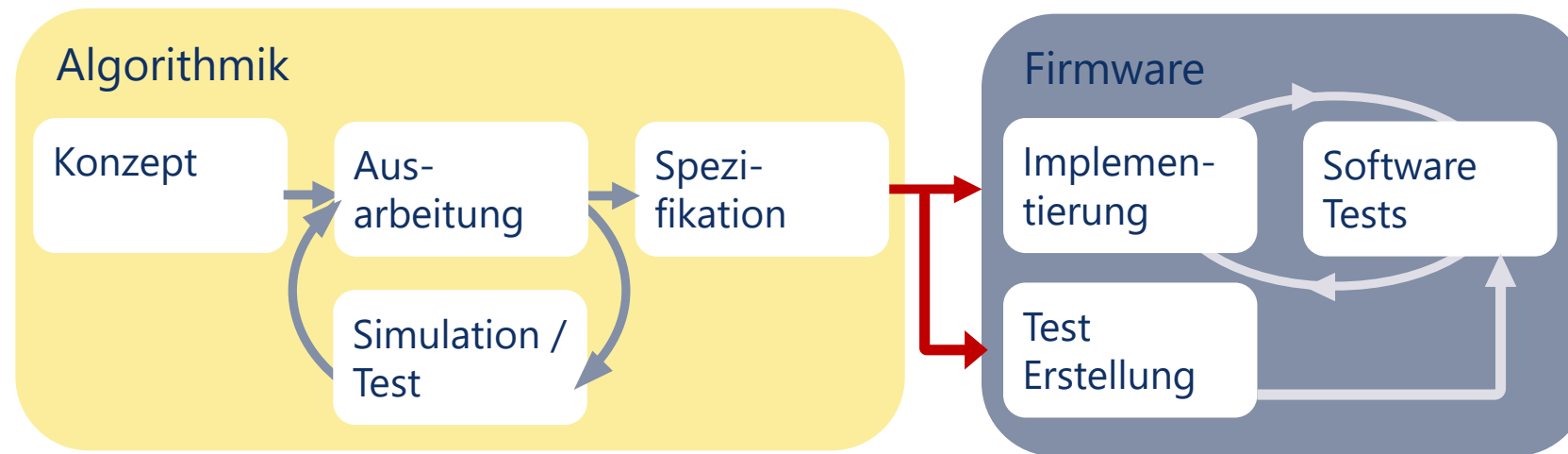
1. Warum Model-Based Design?
2. Messaufgabe & Algorithmen Entwicklung
3. Umsetzung in Simulink
4. Zusammenfassung

# Herkömmlicher Projektlauf

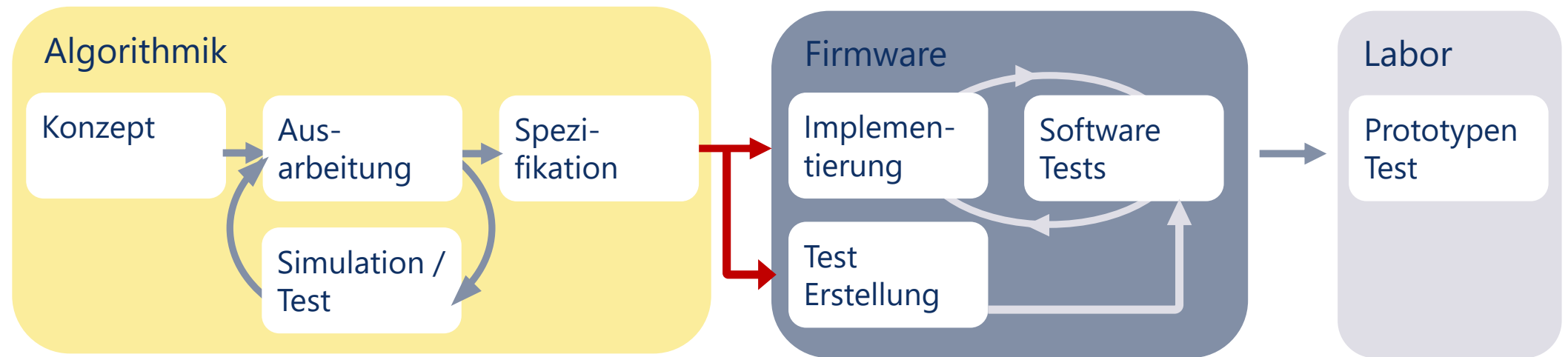




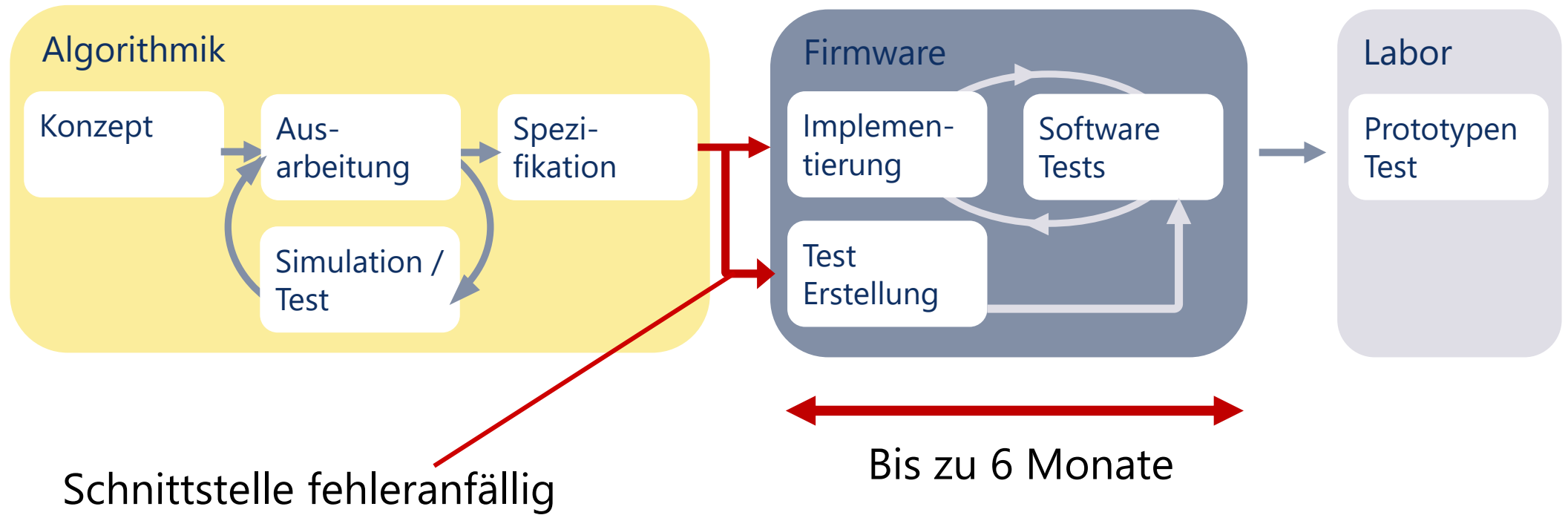
# Herkömmlicher Projektablauf



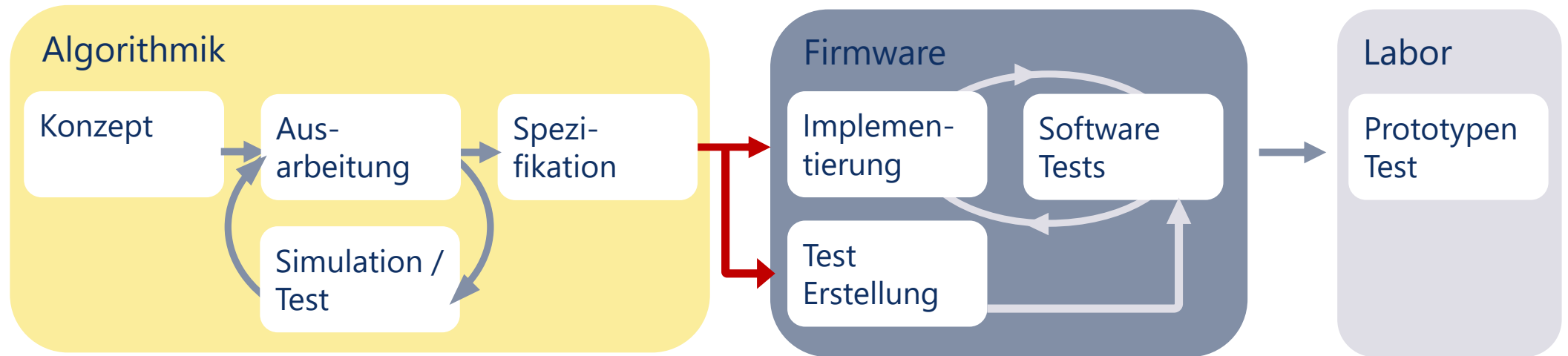
# Herkömmlicher Projektablauf



# Herkömmlicher Projektablauf

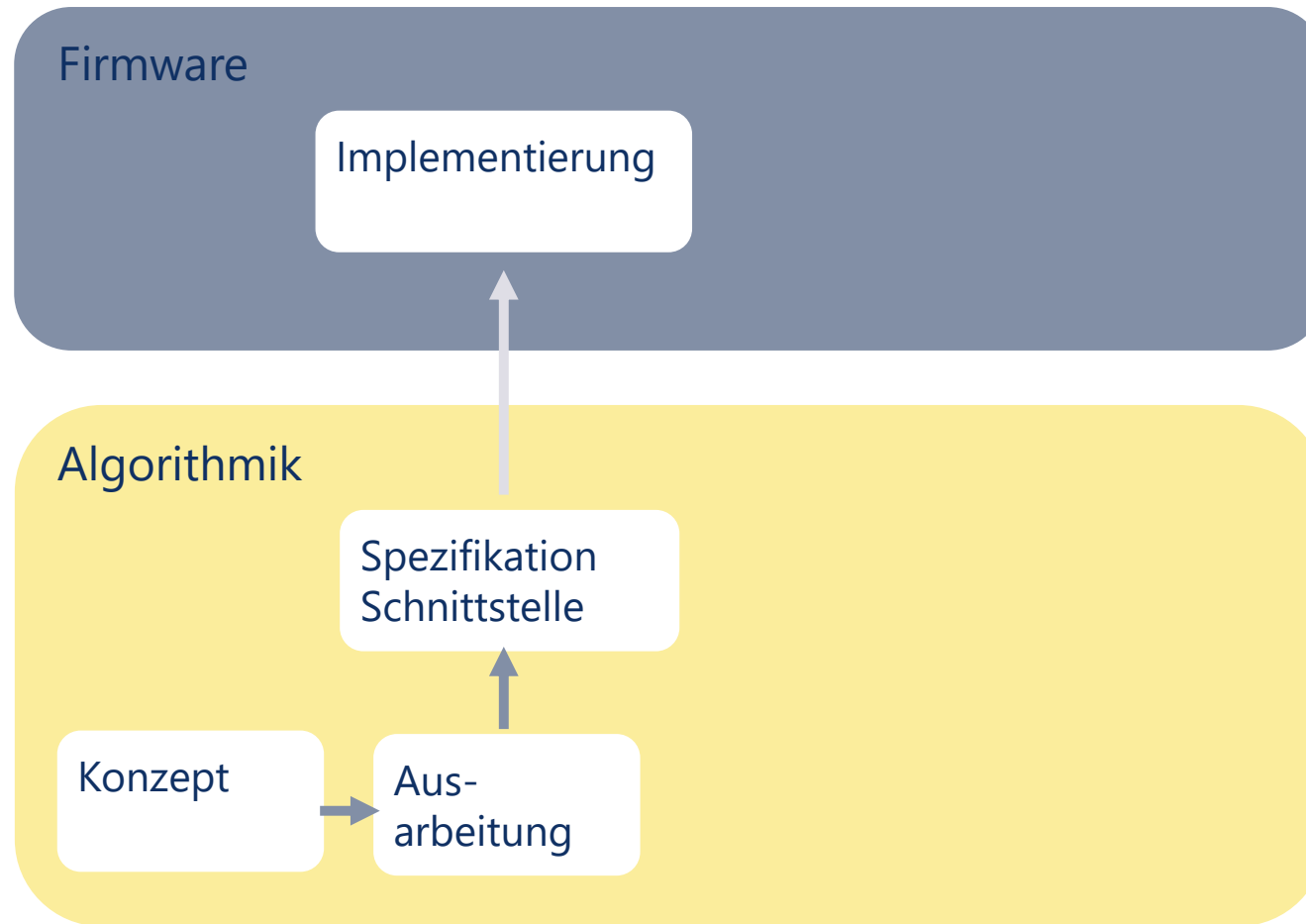


# Herkömmlicher Projektablauf



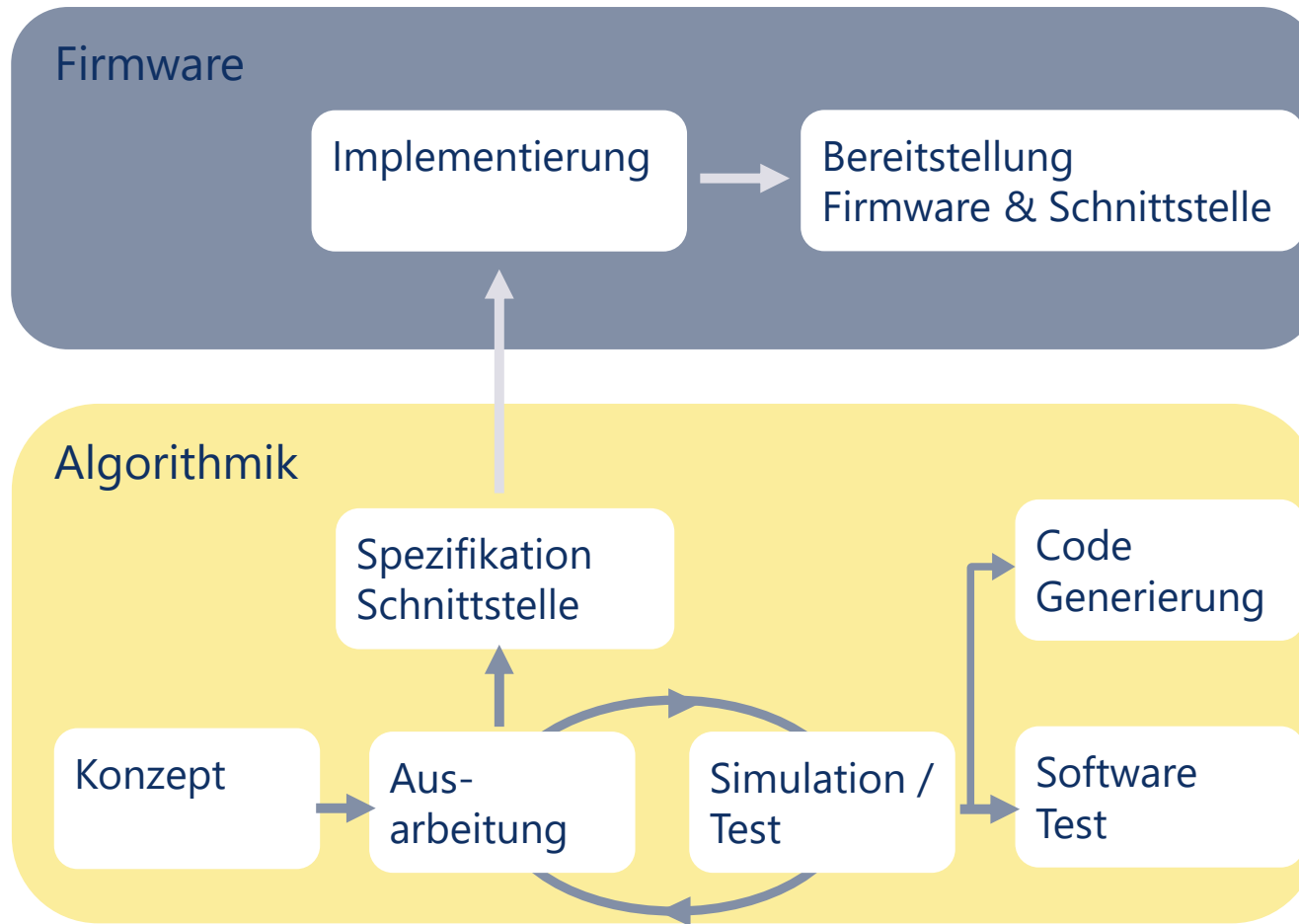
Konzept-Anpassungen aufwändig

# Model-based Design: Projektablauf



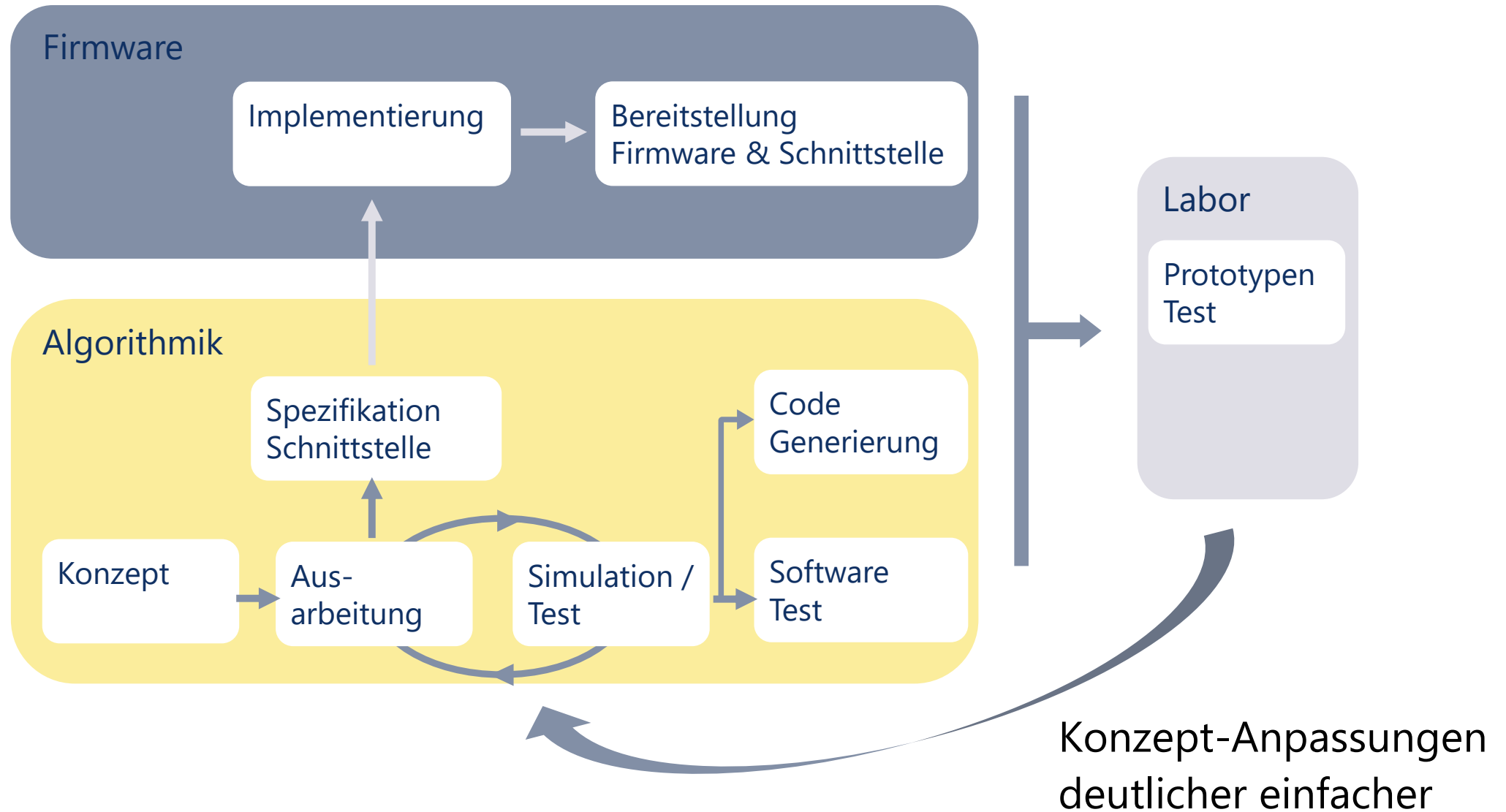
- Frühzeitige Definition der Schnittstelle

# Model-based Design: Projektablauf



- Firmware- und Algorithmen-Entwicklung arbeitet parallel
- Code- /Test-Erstellung aus Simulationsmodell

# Model-based Design: Projektablauf



# Model-based Design: Vorteile

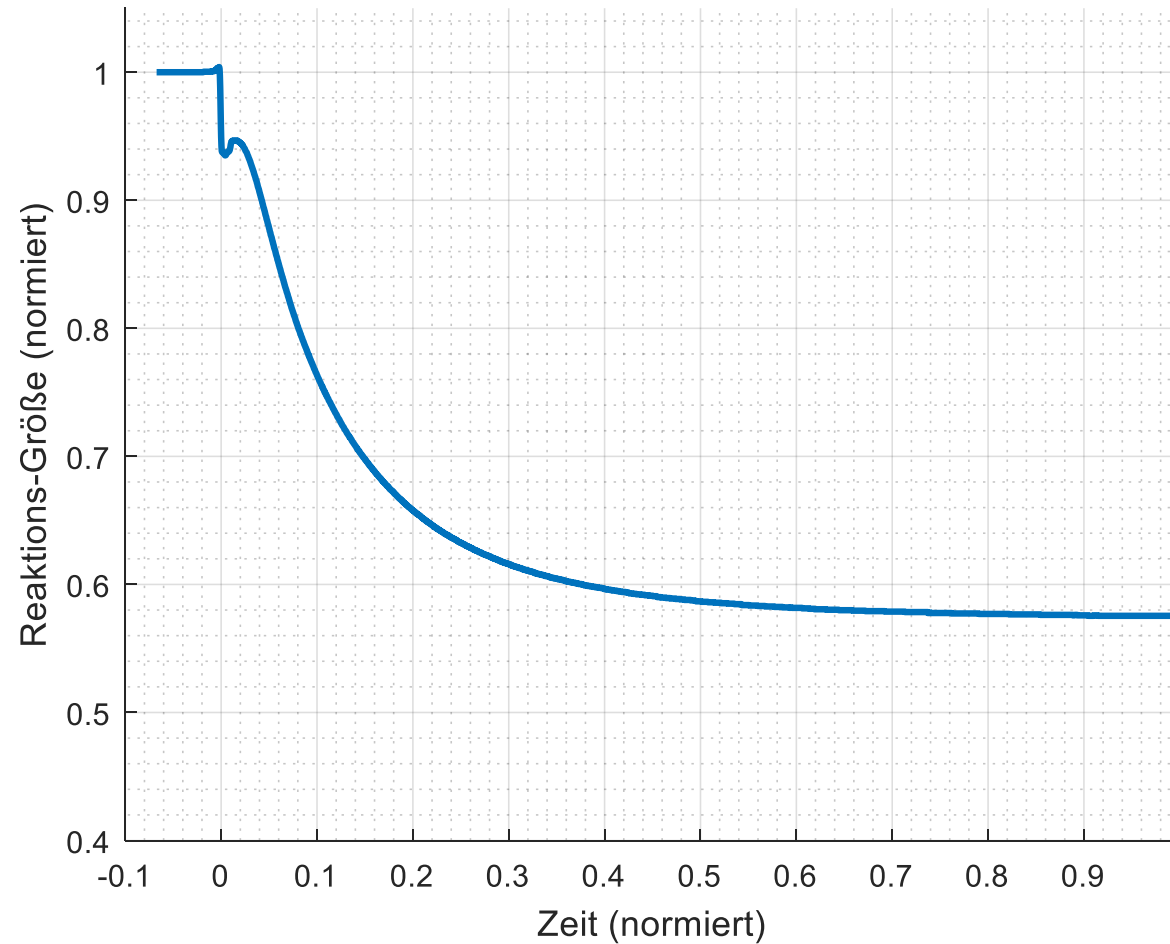
- **Schneller**
  - Paralleles Entwickeln, Automatische Code-Generierung
- **Weniger Fehleranfällig**
  - Keine Übergabe komplexer Spezifikationen
- **Agiler**
  - Nachträglicher Anpassung deutlich einfacher



# Inhalt

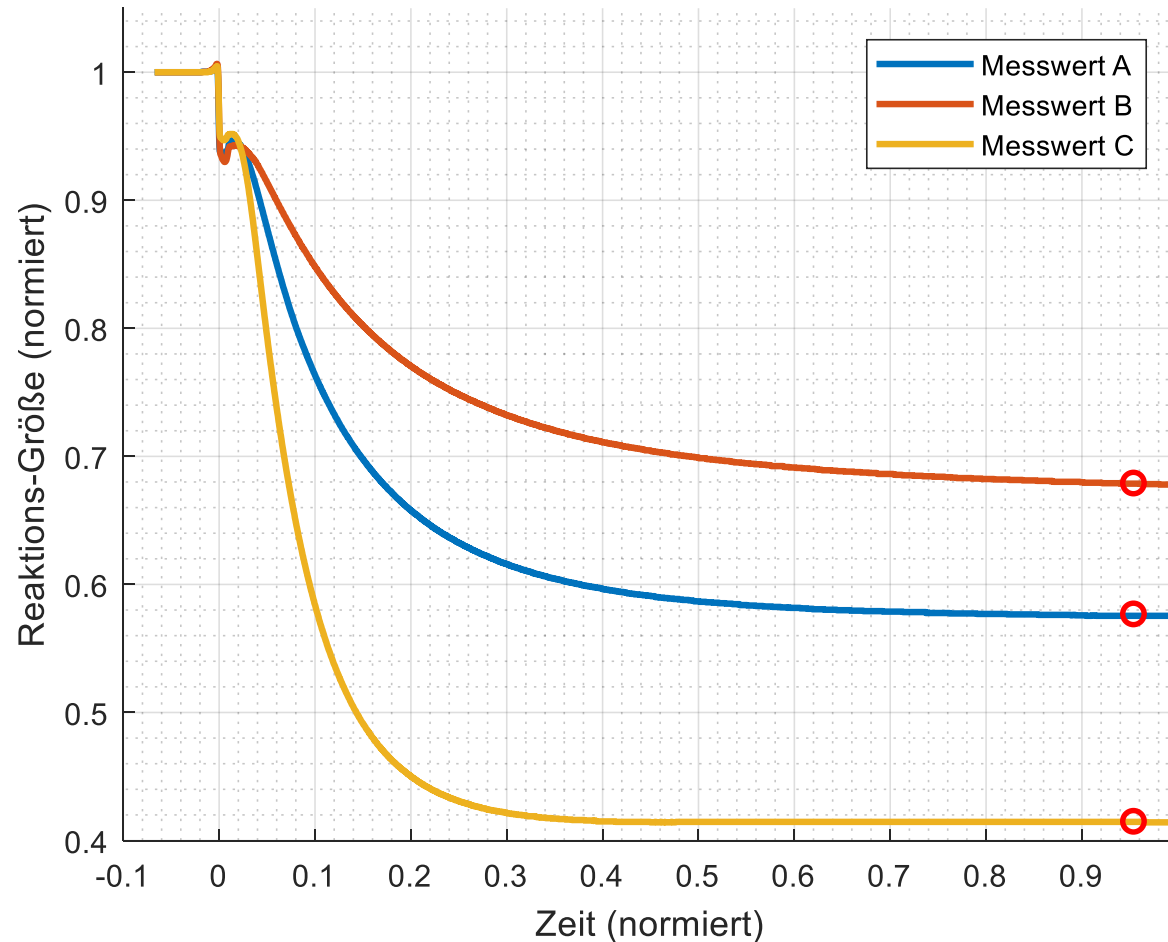
1. Warum Model-Based Design?
2. Messaufgabe & Algorithmen Entwicklung
3. Umsetzung in Simulink
4. Zusammenfassung

# Messaufgabe: Chemische Reaktion



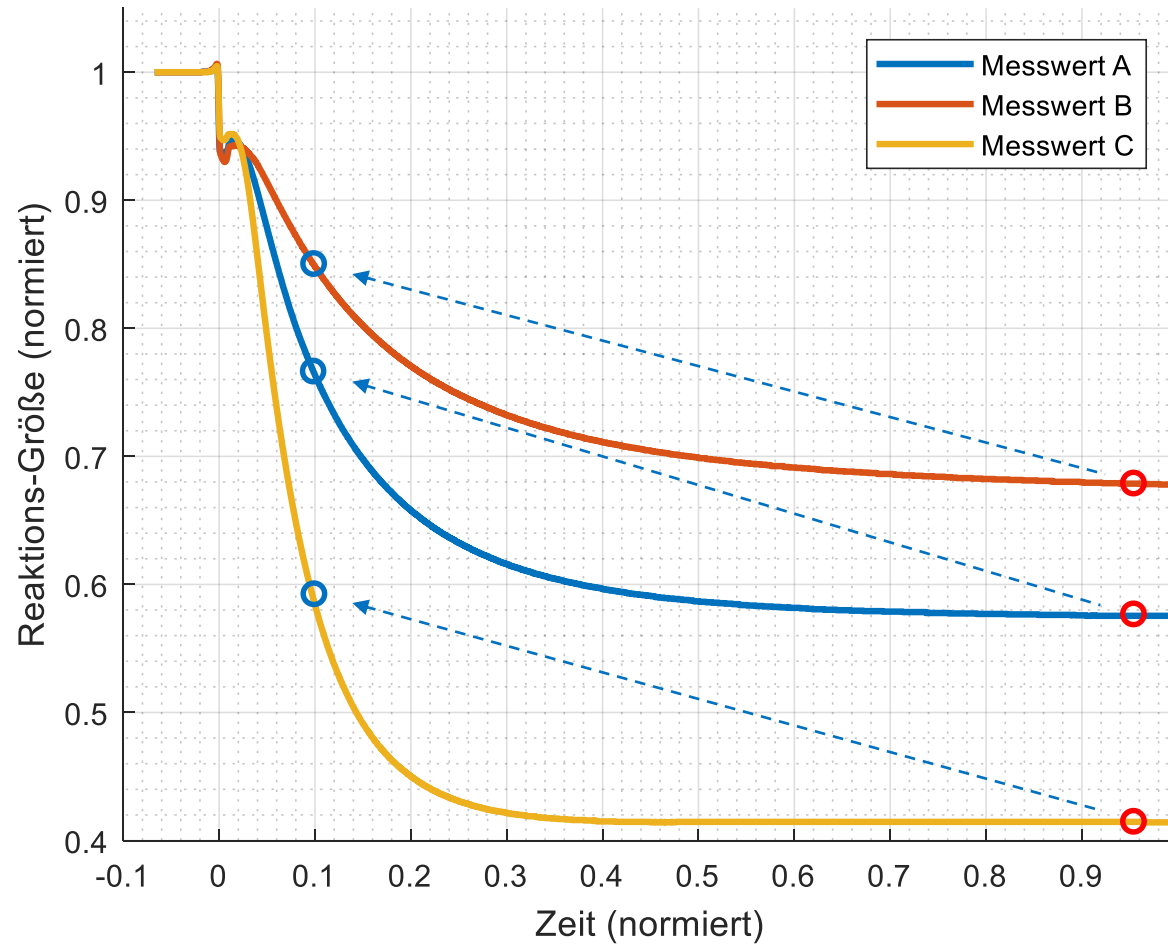
- Analyse chemischer Reaktion
- Verfolgung transienter Reaktionsgröße

# Messaufgabe: Chemische Reaktion



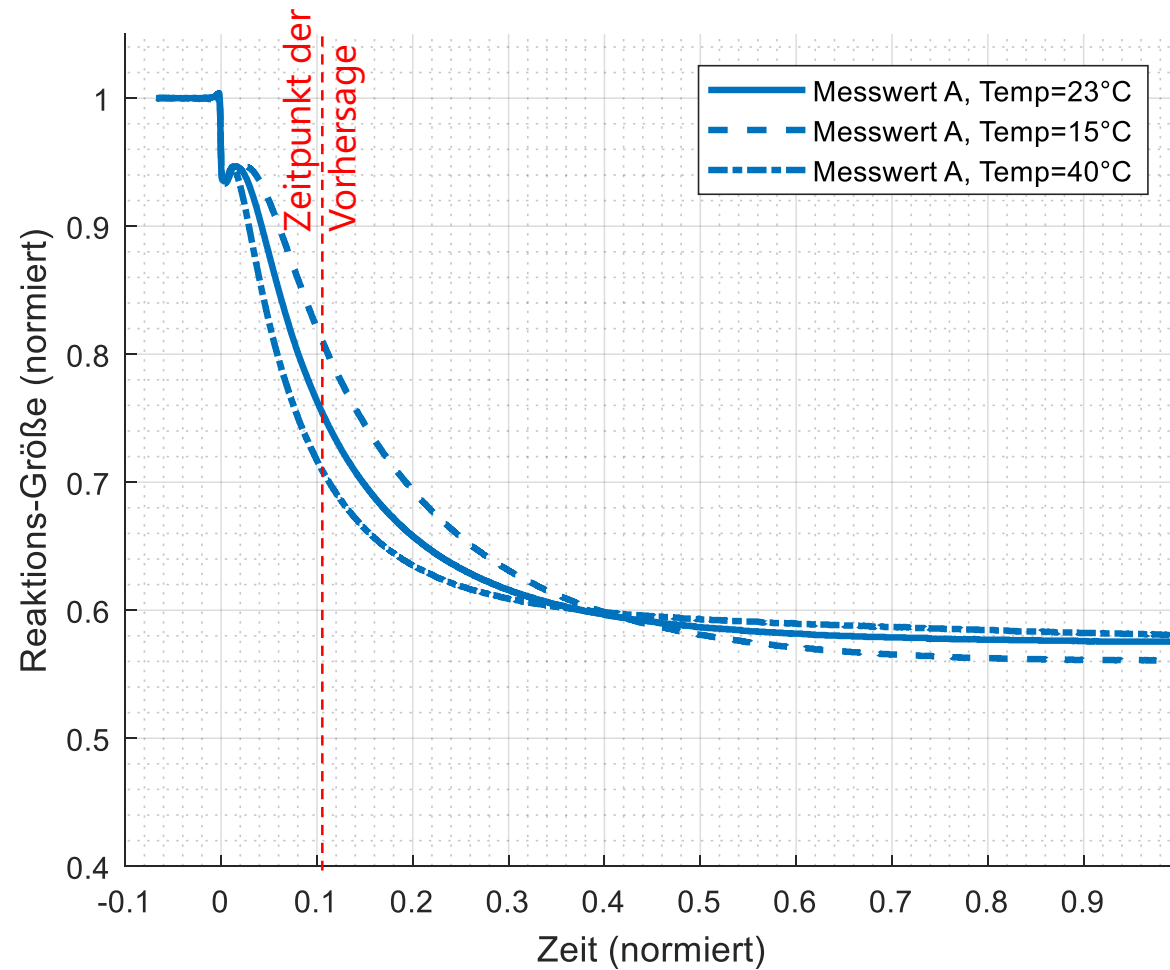
- Analyse chemischer Reaktion
- Verfolgung transienter Reaktionsgröße
- Messwert korreliert mit Endwert

# Messaufgabe: Beispiel



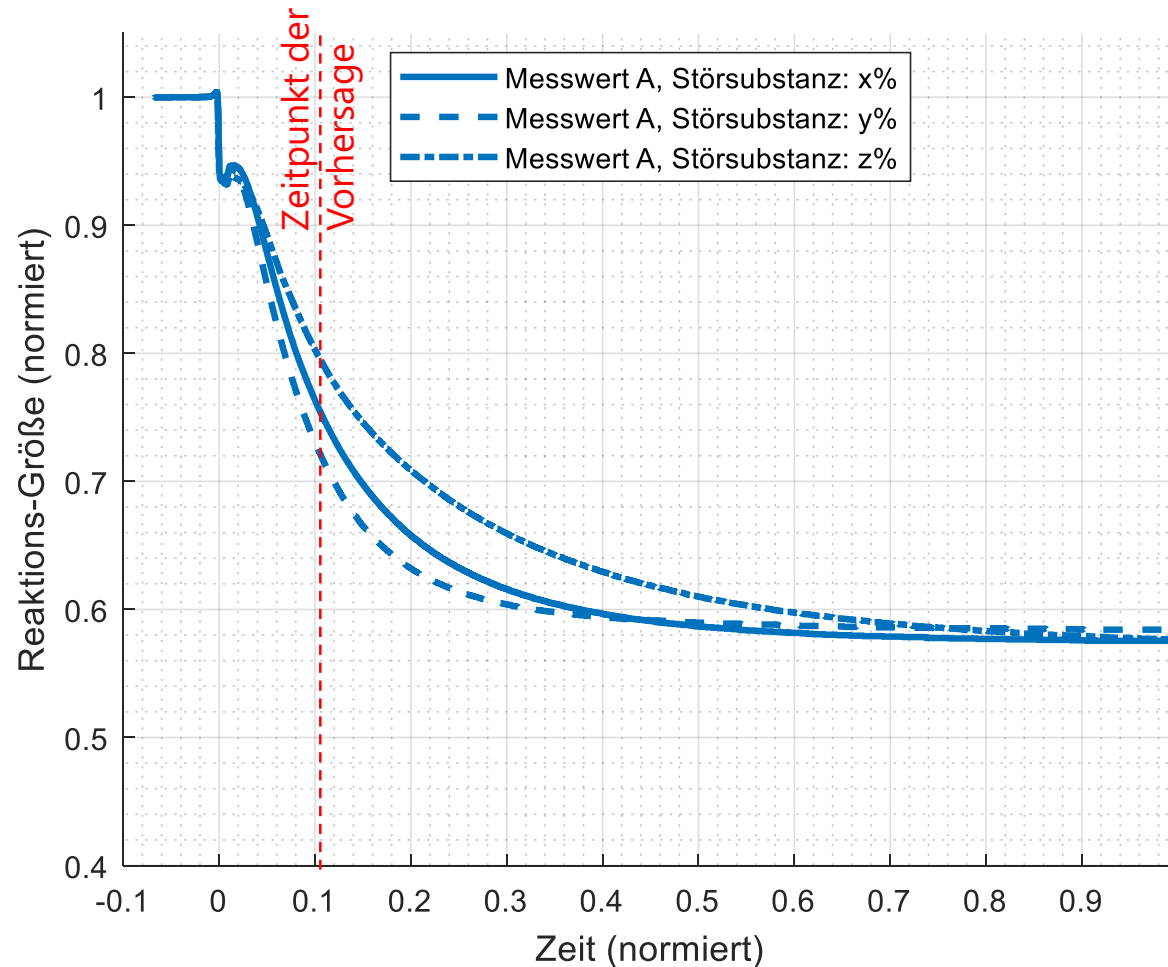
- Gewünscht: Beschleunigung der Messung
- Voraussetzung: Eindeutigkeit

# Temperatur-Einfluss



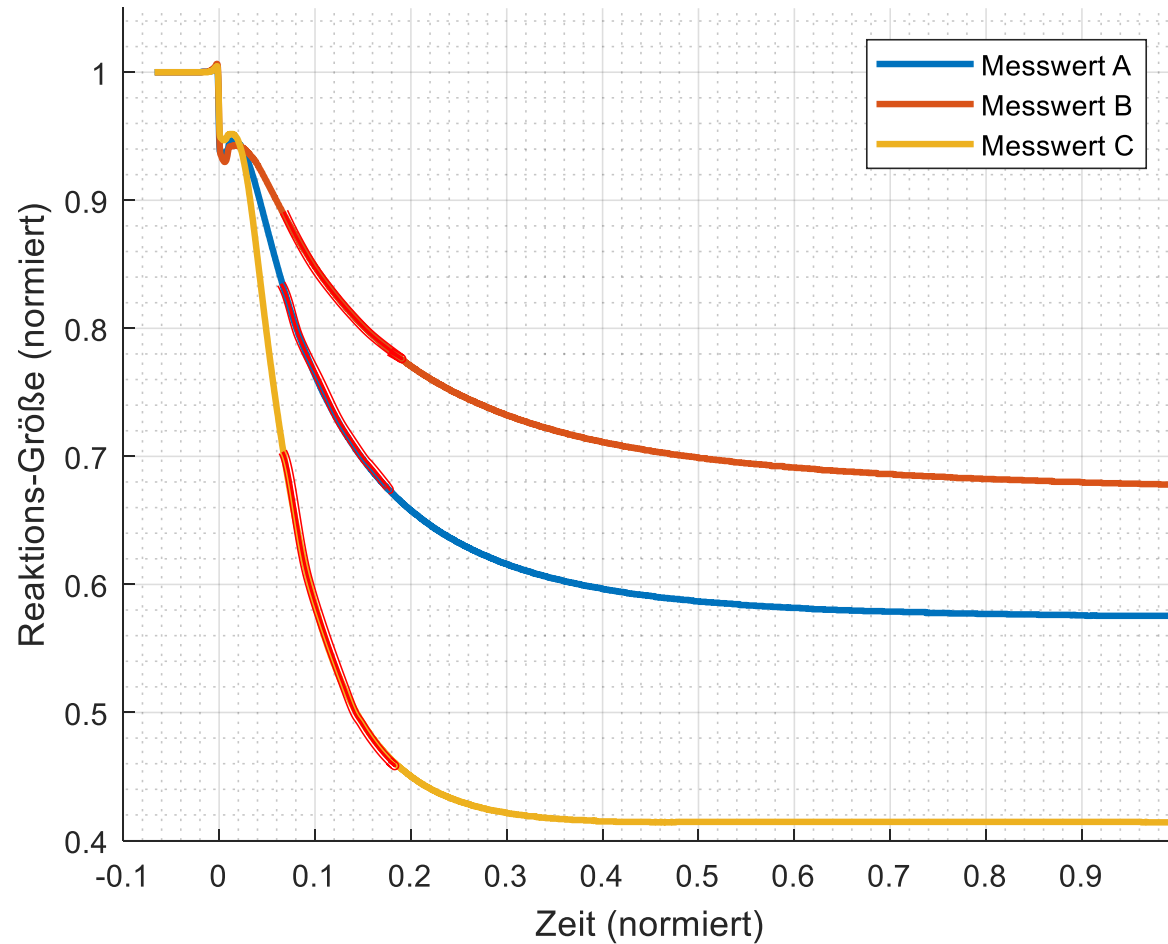
- Temperatur beeinflusst Kurvenform
- Gegenmaßnahme: Temperaturmessung

# Einfluss von Störsubstanzen



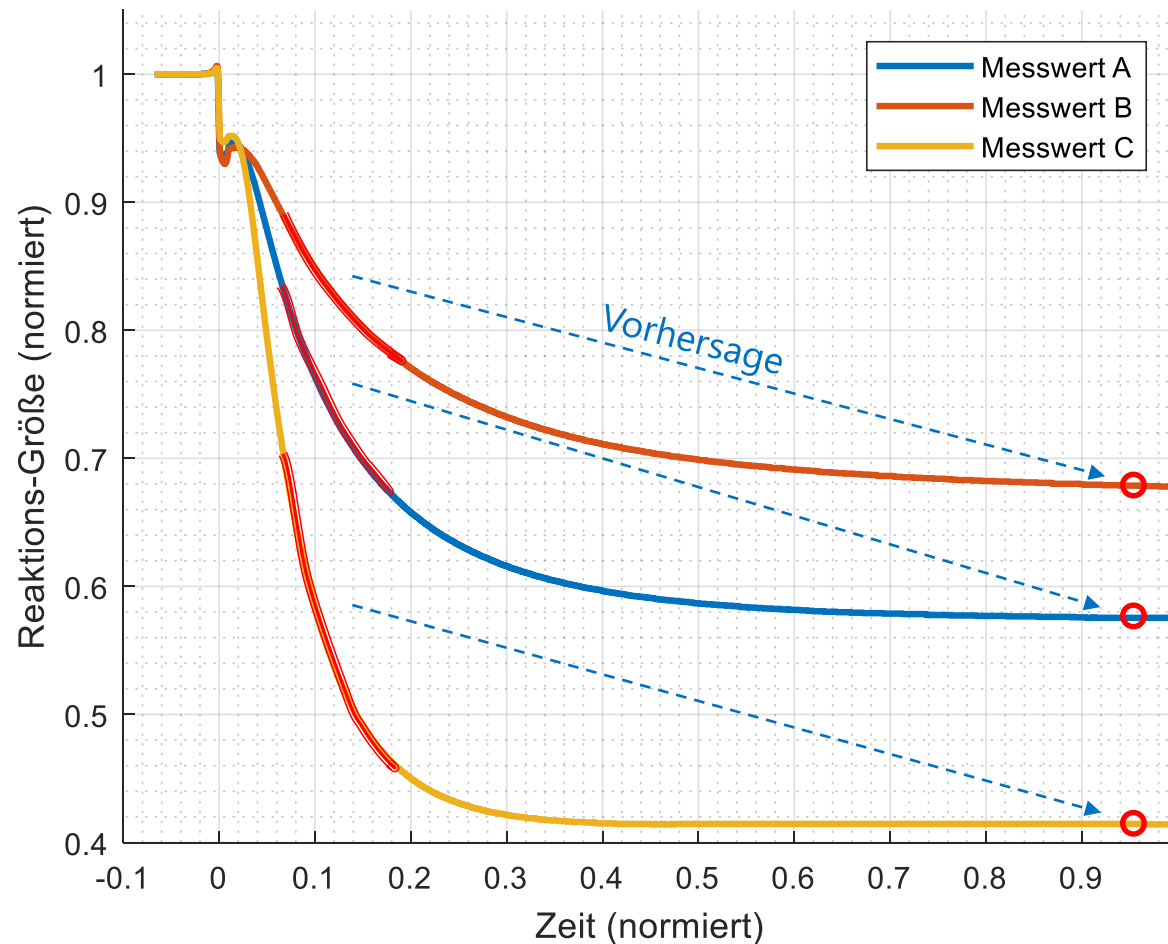
- Störsubstanzen beeinflussen Kurvenform
- Problem: Konzentration kann nicht gemessen werden

# Störsubstanzen: Kompensation



- Kontinuierliche Kurvenformanalyse
- Spezieller Exponentialansatz

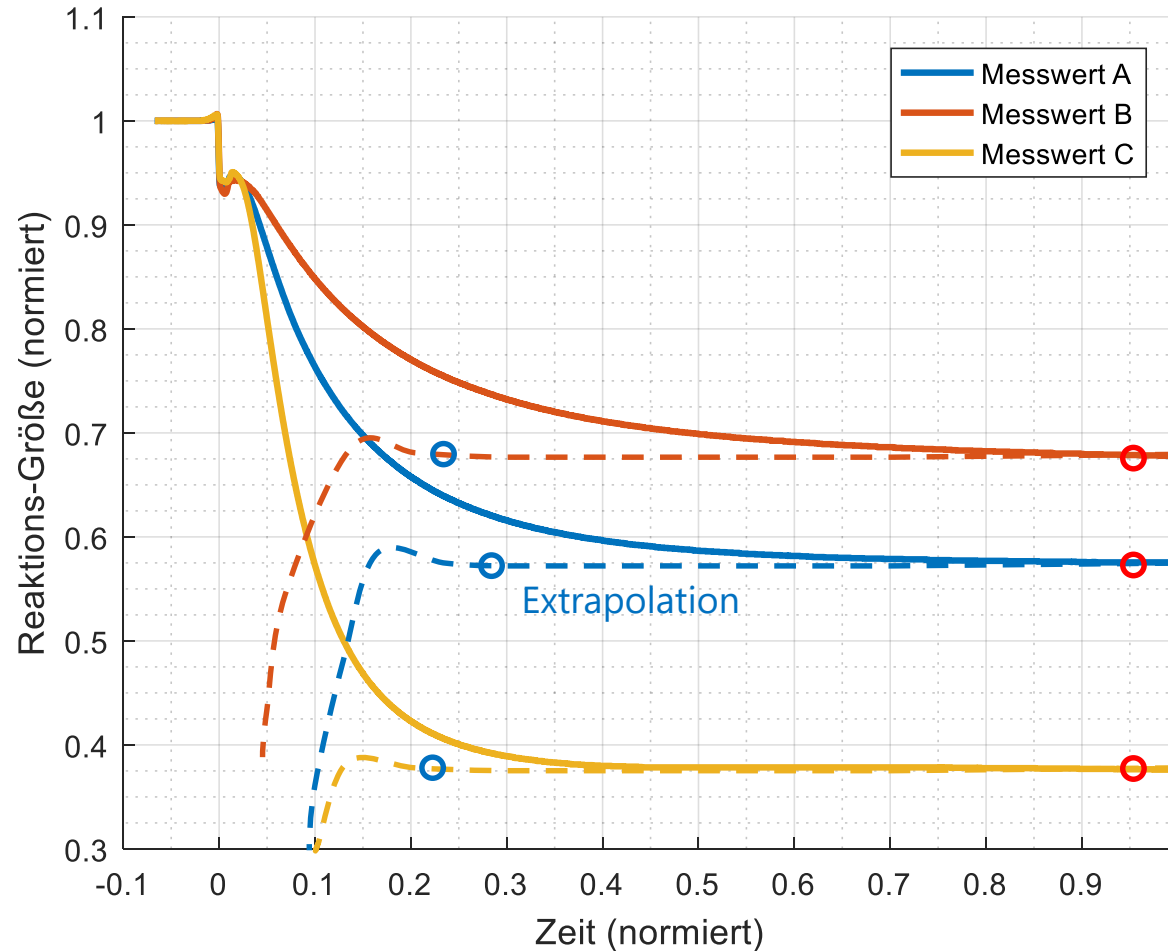
# Störsubstanzen: Kompensation



- Kontinuierliche Kurvenformanalyse
- Spezieller Exponentialansatz
- Vorhersage des Messwerts zu jedem Zeitpunkt

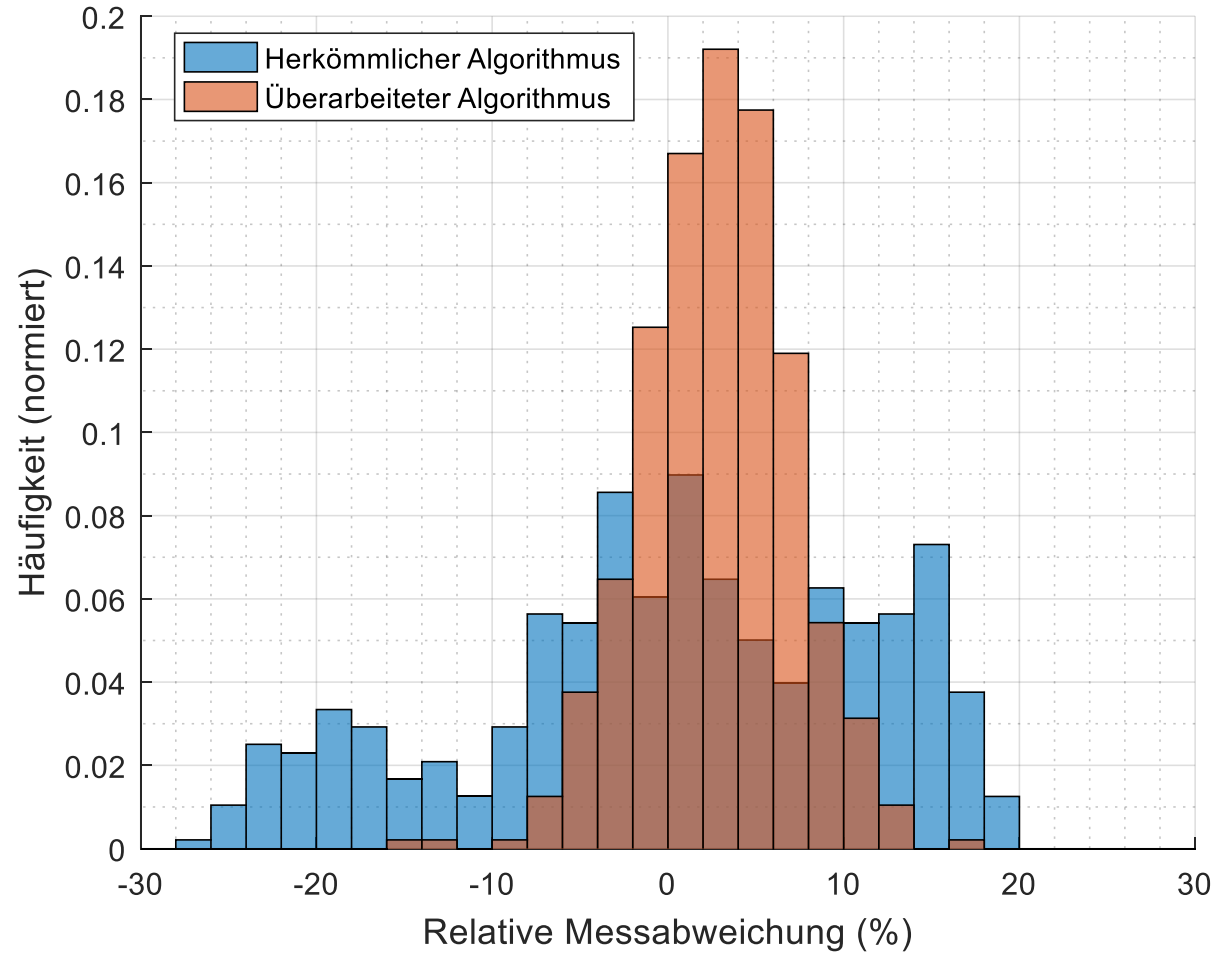


# Störsubstanzen: Kompensation



- Vorhersage zu jedem Zeitpunkt
- Konvergiert deutlich früher

# Störsubstanzen-Korrektur: Messabweichung



90% innerhalb -19% ... +15%

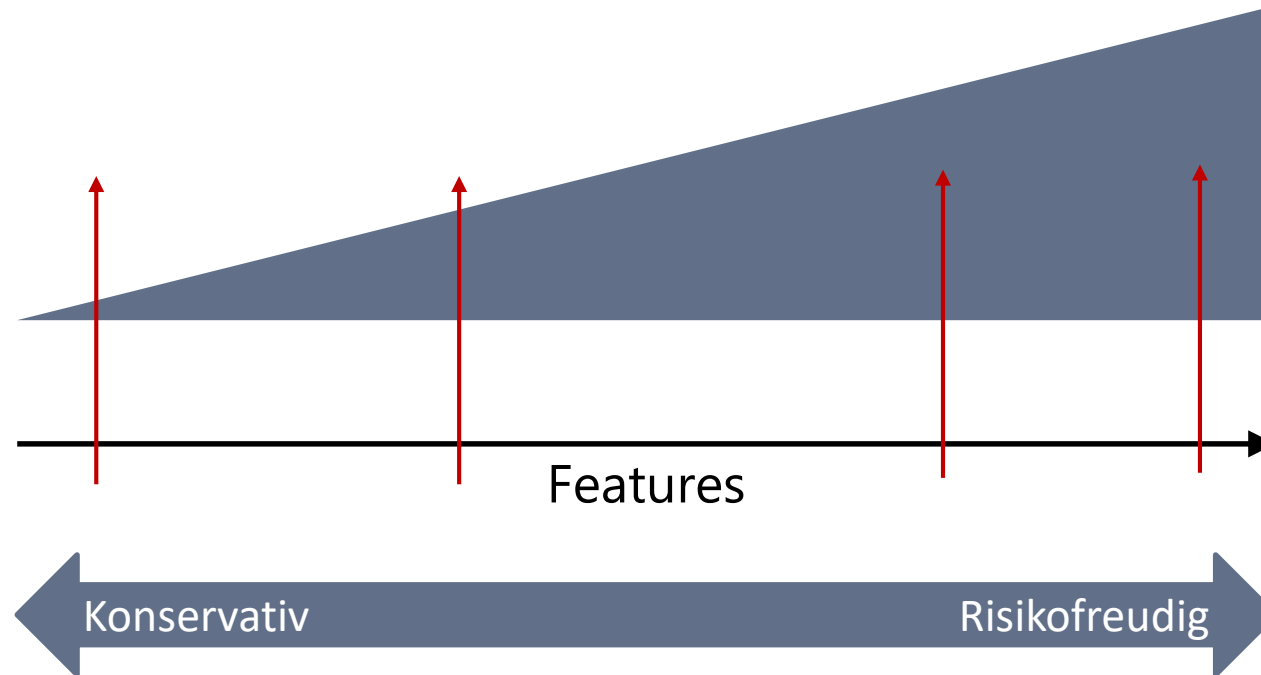


90% innerhalb -4% ... +9%

# Inhalt

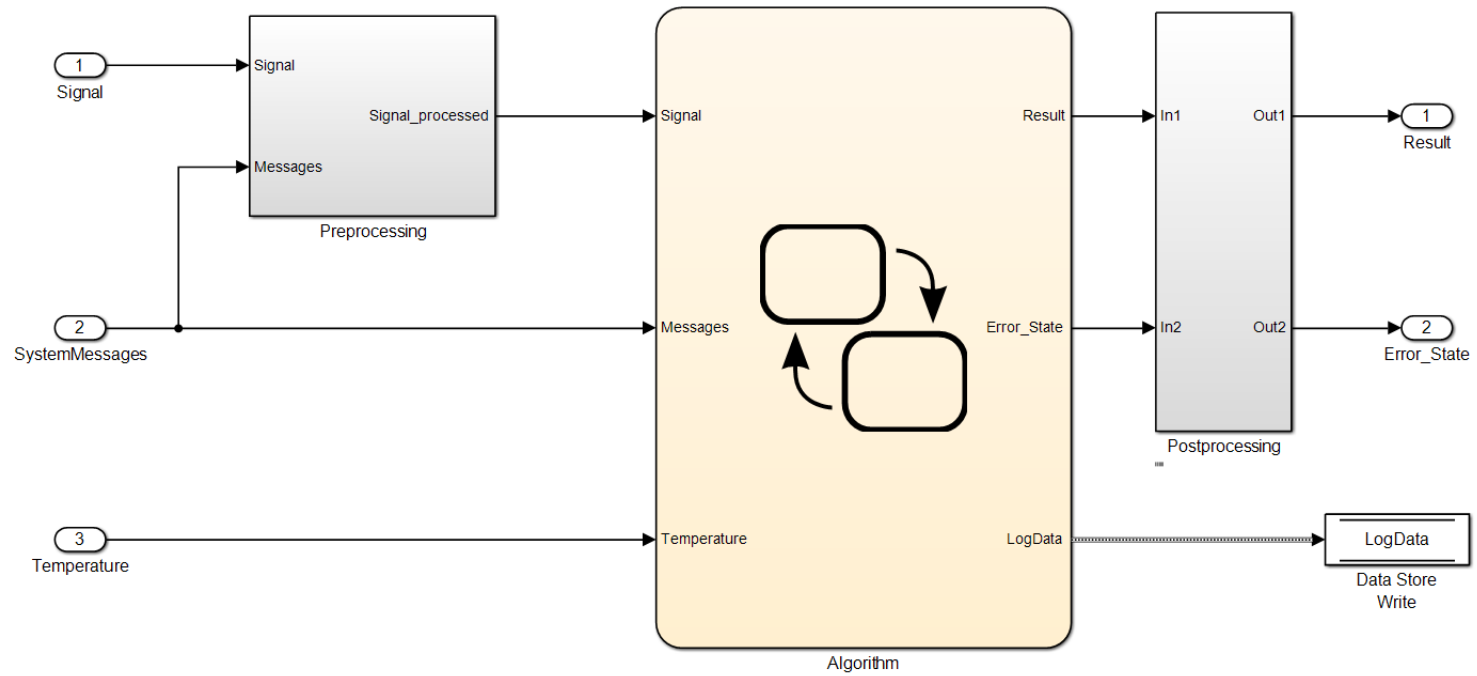
1. Warum Model-Based Design?
2. Messaufgabe & Algorithmen Entwicklung
3. Umsetzung in Simulink
4. Zusammenfassung

# Prototyp: Algorithmen Auswahl

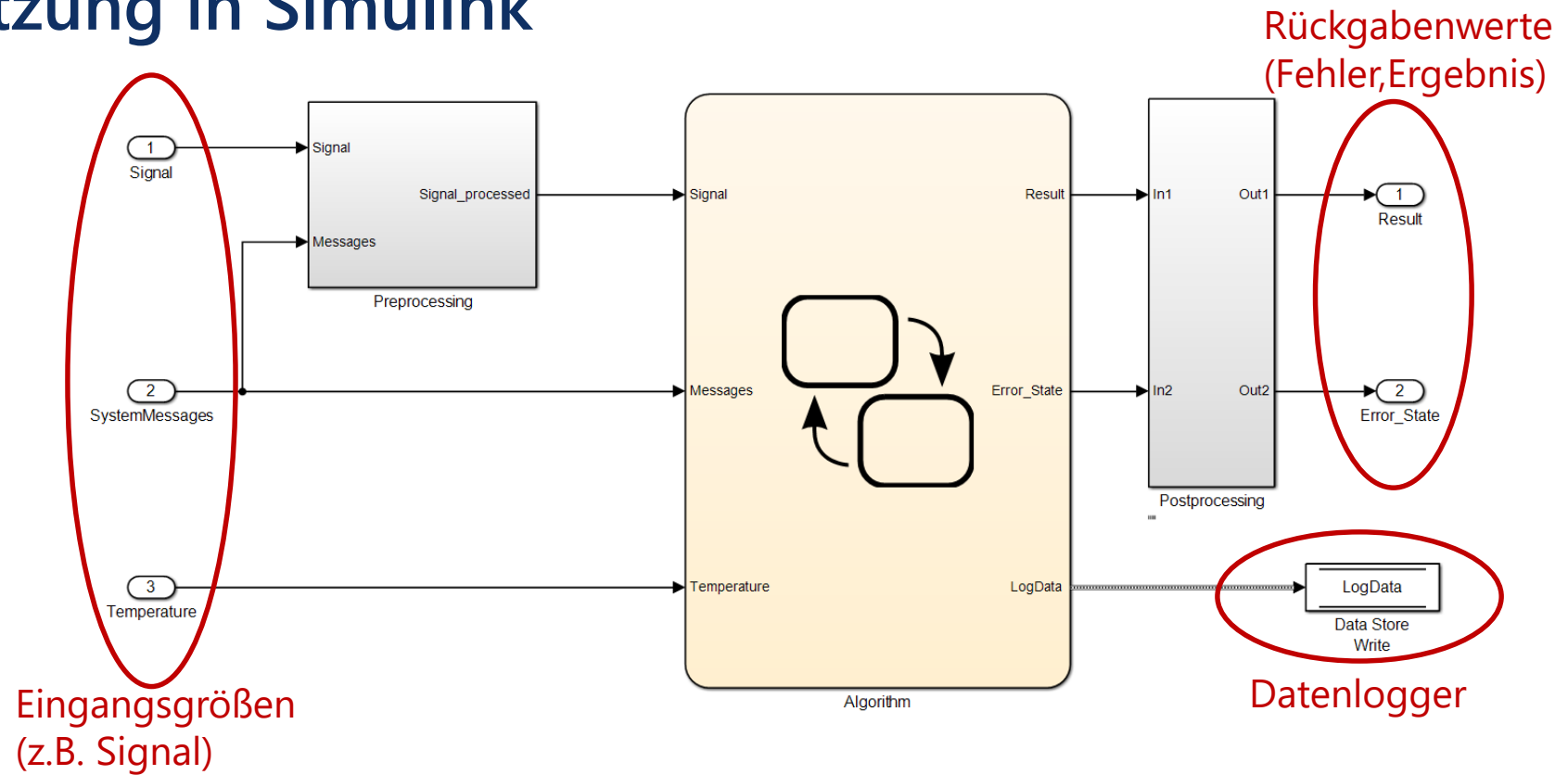


- 4 Varianten implementiert
  - Alle im Prototyp aktivierbar
- Entscheidung wird bewusst offen gehalten
  - Auswahl zu späterem Zeitpunkt problemlos möglich

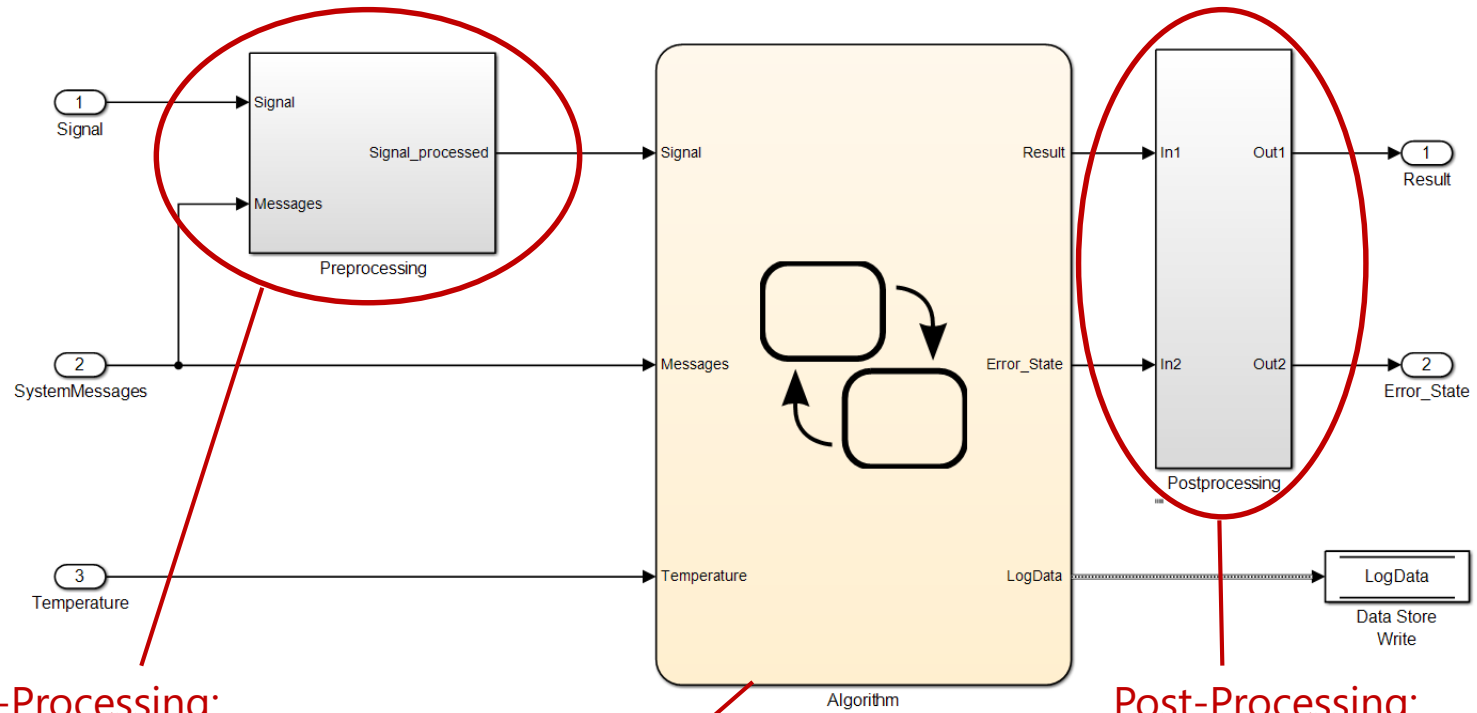
# Umsetzung in Simulink



# Umsetzung in Simulink



# Umsetzung in Simulink

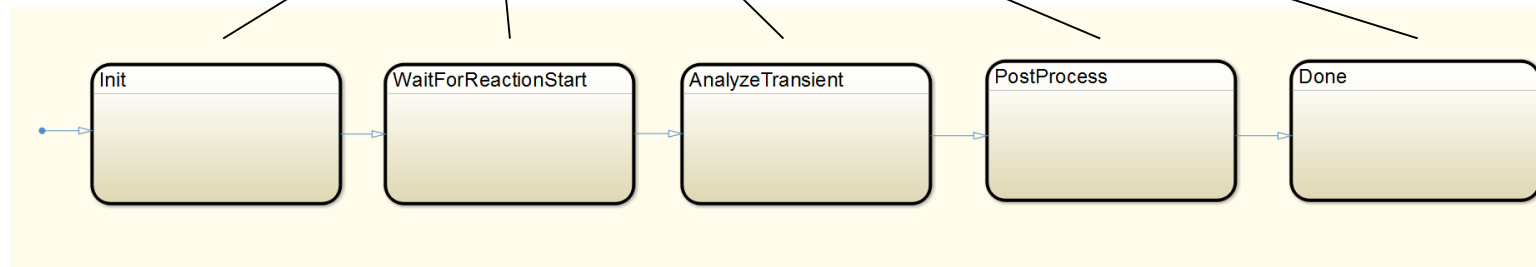
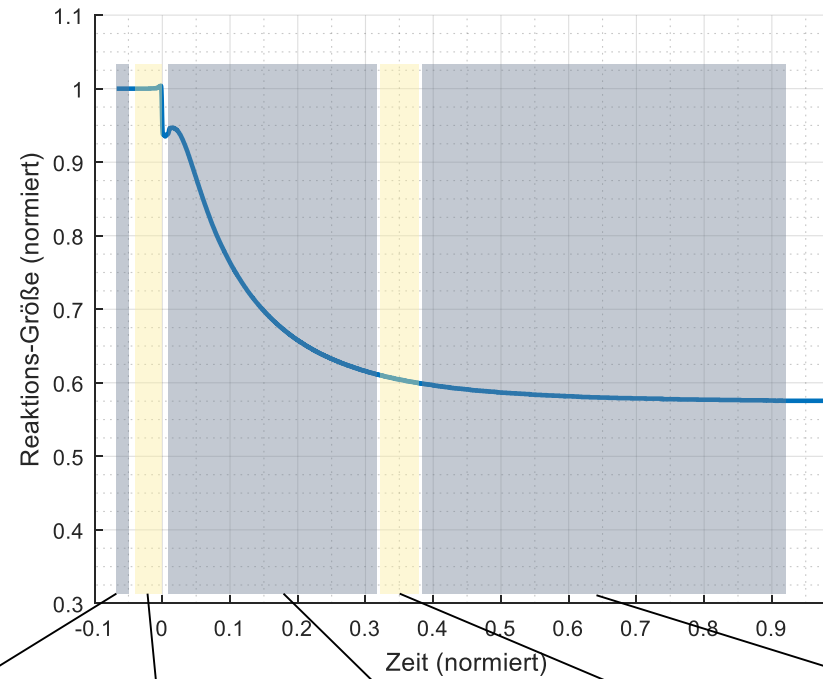


Pre-Processing:  
Signalaufbereitung für die  
Analyse

Hauptroutine:  
Analyse wurde in Stateflow  
realisiert

Post-Processing:  
Signalaufbereitung für  
Rückgabe an Host

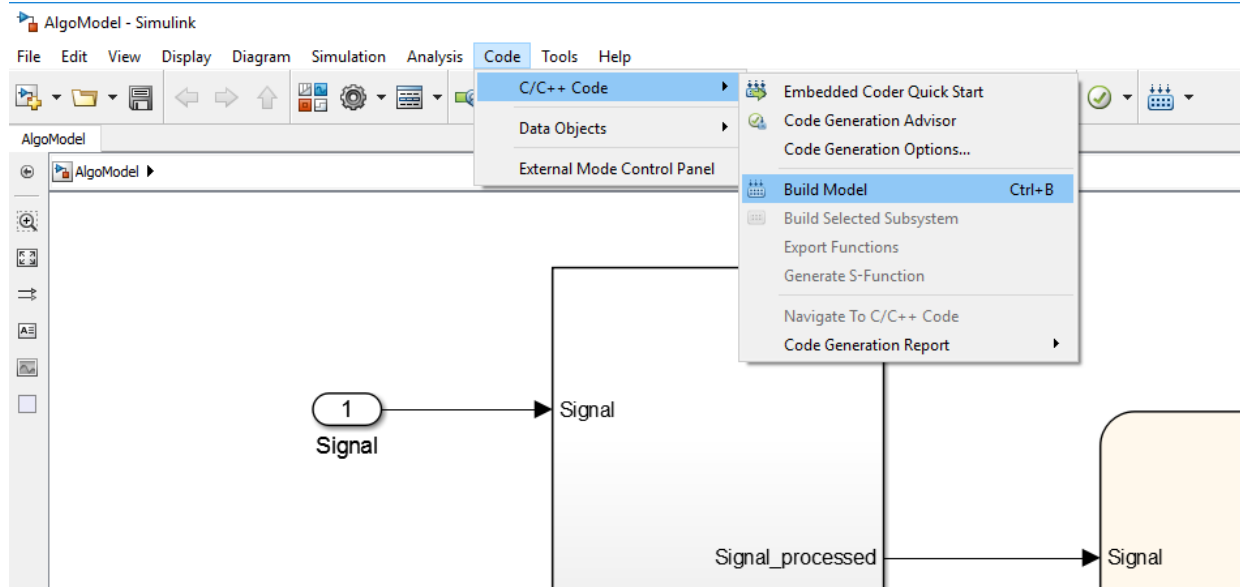
# Hauptroutine: Stateflow



- State-Machine zur Darstellung der Analyse-Phasen im Code



# Code Generierung



- Simulink Embedded Coder
- Spezielles Target
  - Von MathWorks für Kunden erstellt
- Reibungsloser Ablauf
  - Lauffähiger Prototyp vorhanden

# Inhalt

1. Warum Model-Based Design?
2. Messaufgabe & Algorithmen Entwicklung
3. Umsetzung in Simulink
4. Zusammenfassung

# Aktueller Projektstand

## Algorithmus

- für medizin-technisches Messgerät entwickelt
- Lauffähiger Prototyp mit **Simulink Embedded Coder** realisiert
  - Prinzipielle Funktionstests absolviert
  - Bereit für Labor-Tests

# Aktueller Projektstand

## Algorithmus

- für medizin-technisches Messgerät entwickelt
- Lauffähiger Prototyp mit **Simulink Embedded Coder** realisiert
  - Prinzipielle Funktionstests absolviert
  - Bereit für Labor-Tests

## Dokumentation

- **Simulink Report Generator** für Implementierung
- Allgemeine Entwicklungs-Dokumentation

# Aktueller Projektstand

## Algorithmus

- für medizin-technisches Messgerät entwickelt
- Lauffähiger Prototyp mit **Simulink Embedded Coder** realisiert
  - Prinzipielle Funktionstests absolviert
  - Bereit für Labor-Tests

## Dokumentation

- **Simulink Report Generator** für Implementierung
- Allgemeine Entwicklungs-Dokumentation

## Tests

- Erstellt mit **Simulink Test Manager**
- Grundlegende Modul- und Submodul-Tests vorhanden

# Erfahrungen mit Model-based Design

Entlastet von Fremdaufgaben

- Erstellung von Spezifikationen für die Implementierung
- Händische Implementierung des Algorithmus

# Erfahrungen mit Model-based Design

## Entlastet von Fremdaufgaben

- Erstellung von Spezifikationen für die Implementierung
- Händische Implementierung des Algorithmus

## Höhere Komplexität bei höherer Robustheit

- Erleichtert die Umsetzung komplexer Algorithmen
- Vermeidet Fehler bei Implementierung

# Erfahrungen mit Model-based Design

## Entlastet von Fremdaufgaben

- Erstellung von Spezifikationen für die Implementierung
- Händische Implementierung des Algorithmus

## Höhere Komplexität bei höherer Robustheit

- Erleichtert die Umsetzung komplexer Algorithmen
- Vermeidet Fehler bei Implementierung

## Gut für agilen Workflow

- Reduktion der Entwicklungszeit
- Flexibler bei Änderungen



Vielen Dank für Ihre Aufmerksamkeit!

Bildnachweis:

Folien 5&7: © Marvid / iStock

Folie 6: © bannosuke / fotolia

Folie 7: © Robert Kneschke / fotolia