

MotorXP

Christian Brunner, Andreas Kölbl, Ricardo Krause, Bernd Krupinski, Andreas Lackner, Michael Schleinkofer, Franz Welker

January 10, 2017

Projekt Start

Projekt Start Phase



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Projekt Auftrag
- Projekt Plan
- Versionsverwaltung
- Kommunikation
- Dokumentenmanagement

Projekt Start

Projekt Auftrag



Projektauftrag - Gr. 3 - MotorXP

Projekttitel:	Entwicklung und Aufbau eines Motorexperimentierplatzes
Projektnummer:	Gruppe 3
Projektart:	Entwicklungsprojekt
Projektleiter/in:	-
Projektauftraggeber/in:	OTH Regensburg
Projektkunde(n):	Prof. Roth
Projektdauer:	Geplanter Beginn: 04.10.2016 Geplantes Ende: 28.01.2016
Ausgangssituation / Problembeschreibung:	<p>Die Studenten der Technischen Informatik müssen im 6./7. Semester ein Projekt im Rahmen der Vorlesung Datenverarbeitung in der Technik umsetzen.</p> <p>In diesem Projekt sollen im Studium erlernte Techniken und Fähigkeiten angewandt und vertieft werden.</p> <p>Das zu bearbeitende Projekt befasst sich mit der Konzeption, Implementierung und Integration eines Experimentierplatzes für BLDC-Motoren.</p> <p>Aktuell gibt es in der Fakultät IM keinen Experimentierplatz, welcher diese Anforderungen erfüllt.</p> <p>Eine weitere Verwendung des Experimentierplatzes für die Lehre ist gegebenenfalls angedacht</p>
Projektgesamtziel:	<p>Das Ziel des Projektes ist die Entwicklung und der Aufbau eines Motorexperimentierplatzes mit einem BLDC – Motor welcher mit verschiedenen Sensoren und Regelungen ausgestattet werden kann um variierende Versuchsaufbauten zu ermöglichen. Des Weiteren sollen die gewählten aufbauten als Simulation zur Verfügung stehen um Vergleiche zwischen den erdachten Konzepten und den Realen Aufbau zu erhalten. Darüber hinaus sollen die erfassten Sensordaten mittels einer Schnittstelle an einen PC gesendet und zusätzlich zu den Simulationen visualisiert werden. Es sollten teile des Projektes für die EM Vorlesung als Unterrichtsmaterial aufbereitet werden können.</p>

Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

Projekt Start

Projekt Auftrag



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

	Name	Dauer	Start	Ende	Ressourcen
1	<input checked="" type="checkbox"/> Projektvorbereitung	6 tage	04.10.16 ...	11.10.16 ...	
2	Projektauftrag erstellen	0,125 tage	04.10.16 0...	04.10.16 ...	Ricardo Krause
3	Projektplan erstellen	0,188 tage	04.10.16 0...	04.10.16 ...	Ricardo Krause[50%],Andreas Lackner[50%]
4	GitHub einrichten	0,125 tage	04.10.16 0...	04.10.16 ...	Ricardo Krause
5	VM Workspace einrichten	0,375 tage	04.10.16 0...	04.10.16 ...	Andreas Koelbl
6	Dropbox einrichten	0,125 tage	04.10.16 0...	04.10.16 ...	Ricardo Krause
7	Dokumentenvorlagen erstellen	0,5 tage	04.10.16 0...	04.10.16 ...	Ricardo Krause
8	Slack einrichten	0,25 tage	04.10.16 0...	04.10.16 ...	Ricardo Krause
9	Plakat erstellen	1 tag	11.10.16 0...	11.10.16 ...	BerndKrupinski
10	Logo erstellen	1 tag	11.10.16 0...	11.10.16 ...	BerndKrupinski
11	Simulationsumgebungen einrichten	0,25 tage	04.10.16 0...	04.10.16 ...	Franz Welker
12	<input checked="" type="checkbox"/> Analysephase	7,5 tage	04.10.16 ...	13.10.16 ...	
13	<input checked="" type="checkbox"/> Anforderungsanalyse erstellen	2 tage	04.10.16 ...	05.10.16 ...	
14	Motor anforderungen	2 tage	04.10.16 0...	05.10.16 ...	Andreas Koelbl;Christian Brunner
15	Sensor anforderungen	2 tage	04.10.16 0...	05.10.16 ...	Andreas Lackner;Christian Brunner
16	Regelung und Leitsystem anfo...	2 tage	04.10.16 0...	05.10.16 ...	BerndKrupinski
17	Kommunikations anforderungen	2 tage	04.10.16 0...	05.10.16 ...	MichaelSchleinkofer
18	Benutzeroberfläche anforderu...	2 tage	04.10.16 0...	05.10.16 ...	Ricardo Krause
19	Simulationsanforderungen	2 tage	04.10.16 0...	05.10.16 ...	Franz Welker
20	<input checked="" type="checkbox"/> Schnittstellen definieren	1,5 tage	12.10.16 ...	13.10.16 ...	
21	Motor	1 tag	12.10.16 0...	12.10.16 ...	Andreas Koelbl
22	Sensor	1 tag	12.10.16 0...	12.10.16 ...	Andreas Lackner
23	Regelung und Leitsystem	1 tag	12.10.16 0...	12.10.16 ...	BerndKrupinski
24	Kommunikation	1 tag	12.10.16 0...	12.10.16 ...	MichaelSchleinkofer
25	Benutzeroberfläche	1 tag	12.10.16 0...	12.10.16 ...	Ricardo Krause
26	Simulation	1,5 tage	12.10.16 0...	13.10.16 ...	Franz Welker
27	<input checked="" type="checkbox"/> Entwurfsphase	2 tage	13.10.16 ...	17.10.16 ...	
28	<input checked="" type="checkbox"/> Architektur entwurf	1 tag	13.10.16 ...	14.10.16 ...	
29	Motor	1 tag	13.10.16 1...	14.10.16 ...	Andreas Koelbl;Christian Brunner
30	Sensor	1 tag	13.10.16 1...	14.10.16 ...	Andreas Lackner;Christian Brunner
31	Regelung und Leitsystem	1 tag	13.10.16 1...	14.10.16 ...	BerndKrupinski
32	Kommunikation	1 tag	13.10.16 1...	14.10.16 ...	MichaelSchleinkofer
33	Benutzeroberfläche	1 tag	13.10.16 1...	14.10.16 ...	Ricardo Krause

Projekt Start

Github



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

The screenshot shows the GitHub interface for the repository `OTHRegensburgDT / dt_proj_ws16_17`. At the top, there's a search bar and navigation links for Pull requests, Issues, and Gist. Below the repository name, there are buttons for Unwatch (4), Star (0), and Fork (0). A navigation bar includes links for Code, Issues (5), Pull requests (0), Projects (0), Wiki, Pulse, Graphs, and Settings. The repository description states "No description or website provided." Below this, statistics show 170 commits, 9 branches, 0 releases, and 4 contributors. A section for recently pushed branches highlights the `Documentation` branch, pushed 41 minutes ago, with a "Compare & pull request" button. Below this are buttons for "Branch: master", "New pull request", "Create new file", "Upload files", "Find file", and "Clone or download". The commit history table lists the following commits:

Commit Hash	Commit Message	Time Ago
RK-003	REMOVE unused using	Latest commit 02978b3 17 days ago
com	Stability patch: Start of Frame for Sensordata message	18 days ago
common	Adds Sensor Interface and Hall Support	2 months ago
doc	ADD: Eclipse file templates	3 months ago
gui/MotrXPGUI	REMOVE unused using	17 days ago

Projekt Start

Slack

MotorXP

Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

DT_Proj_WS16/... ric687

★ STARRED

general

CHANNELS (10)

- # communication
- # documentation
- # github_info
- # gui_integration
- # motor_integration
- # random
- # sensor
- # simulation

DIRECT MESSAGES (8)

- slackbot

#general 7 | 0 | Company-wide announcements and w...

berndnk 5:40 PM
uploaded this image: [MotorXP_Title_Black.png](#)

MotorXP
Motor Experimentier Platz

berndnk 5:40 PM
Hier alles was ihr jemals wolltet

berndnk 5:40 PM
uploaded this image: [MotorXP_Title_White.png](#)

MotorXP
Motor Experimentier Platz

+ Message #general

Projekt Start

Dropbox



Projekt Start

Sensorik








Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

Name	Änderungsdatum	Typ	Größe
 0100_AktuelleVersion	04.01.2017 18:00	Dateiordner	
 0200_Projektplan	26.11.2016 22:01	Dateiordner	
 0300_Anforderungen	04.01.2017 18:02	Dateiordner	
 0400_Projekt_Vorbereitung	04.01.2017 18:01	Dateiordner	
 0500_Material_Sammlung	04.01.2017 09:48	Dateiordner	
 0600_Präsentationen	04.01.2017 18:01	Dateiordner	
 0700_Dokumentation	04.01.2017 09:48	Dateiordner	

Sensorik

Anforderungen



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Welche Daten brauchen wir?
 - Kommutierungszeitpunkt
 - Umdrehungsgeschwindigkeit
 - Drehwinkel
 - Temperatur
- Welche Sensoren stehen zur Verfügung?
 - Drei Hall-Sensoren
 - Inkrementalgeber
 - NTC Temperatursensor

Sensorik

Sensor Interface



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Kapselung in ein eigenes Softwaremodul
- Zugriff auf Sensorwerte über ein definiertes Interface
 - Geringer Integrationsaufwand
 - Gute Portierbarkeit

Sensor.h

```
├── Sensor_Hall.h
├── Sensor_QuadratureDecoder.h
└── Sensor_Temperature.h
```

Sensorik

Sensor Interface



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

```
void Sensor_Init();

void Sensor_StartAll(void);

void Sensor_StopAll(void);

void Sensor_SetDirection(MotorDirection_t direction);

Std_ReturnType Sensor_RegisterHallCallback(Sensor_HallCallbackType callback);

Std_ReturnType Sensor_GetCurrentHallPattern(Sensor_HallPattern_t* pattern);

Std_ReturnType Sensor_GetVelocity(double* velocity);

Std_ReturnType Sensor_GetAngle(double* angle);

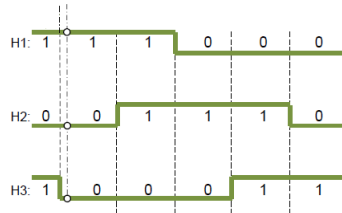
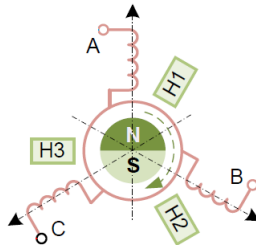
Std_ReturnType Sensor_GetTemperature(int* temperature);
```

Sensorik

Hall-Sensoren

MotorXP

Digitale Hall-Sensoren zur Messung von Magnetfeldern



Sensorik

Hall-Sensoren

MotorXP

Projekt Start

Sensorik

Kommunikation

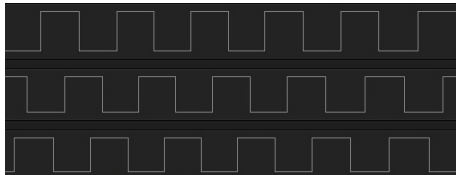
Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Messung des Hall-Patterns mit POSIF
- Zwei mögliche Events
 - Correct-Hall-Event
 - Wrong-Hall-Event
- Ermittlung des motorspezifischen Hall-Patterns



Sensorik

Inkrementalgeber



Projekt Start

Sensorik

Kommunikation

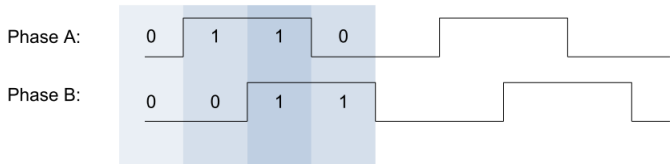
Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Messung Umdrehungsgeschwindigkeit
- Messung Drehwinkel der Welle
- Drei Signalleitungen
 - Indexleitung
 - Phase A
 - Phase B



- Mögliche Implementierungsstrategien
 - POSIF + CCU
 - Zwei CCU Slices

Sensorik

Temperatursensor



Projekt Start

Sensorik

Kommunikation

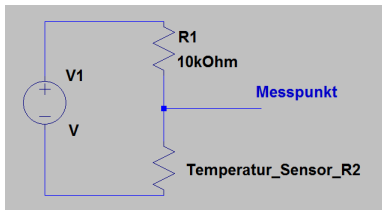
Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- NTC Widerstand
 - Sinkender Widerstand bei steigender Temperatur
- Temperaturermittlung durch Datenblatt
- Widerstand nicht direkt messbar
 - Messung durch Spannungsteiler



Sensorik

Temperatursensor



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

Berechnung des Widerstands

$$U_2 = \frac{U_{ges}}{R_1 + R_2} * R_2 \quad (1)$$

Umstellen auf R2 durch Äquivalenzumformung

$$R_2 = \frac{U_2 * R_1}{U_{ges} - U_2} \quad (2)$$

Sensorik

Ausblick



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Portierung auf anderen Controller
- Nutzung des Inkrementalsgebers als Basis für die Kommutierung

Kommunikation

Anforderungen



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Controller -> PC
 - Sensordaten
 - Wiederholt
 - Erweiterbarkeit
- PC -> Controller
 - Regelungsparameter
 - Sporadisch

Kommunikation

Entwurf



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Physical Layer
 - UART-Baustein des μ -Controllers via USB
 - DAVE APP zur Parametrierung
- Data Link Layer
 - Eigens definiertes Frame-Format



Kommunikation

Entwurf



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Restliche Layer
 - Keine Adressierung, da genau zwei Teilnehmer
 - Keine Sessions
 - Keine Flusskontrolle
- Payload: Protocol Buffer Nachricht
 - Flexibilität und Erweiterbarkeit
 - Performance

Kommunikation

Entwurf



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

● Sensordaten

```
//defining an entry of the data
    ↳ table
message DataEntry{
    uint32 SensorId = 1;
    double Data = 2;
}
//defining the real message
message SensorMsg{
    //Upcounting Nr
    uint64 SequenceNr = 1;
    //all Data
    repeated DataEntry
        ↳ DataTable = 2;
}
```

● Parameter

```
//defining the parameter message
message RegParams{
    uint32 target = 1;
    float paraP = 2;
    float paraI = 3;
    float paraD = 4;
    float tgtVal = 5;
}
```

Kommunikation

Implementierung



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Frameaufbau für Sensordaten erweitert



- PC: C#-Bibliothek
 - SerialPort-Objekt
- Controller: C-Funktionen
 - DAVE APP für UART
 - DAVE APP für CRC

Regulation & GUI Controls

Regulation



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Regeln des Motors über Sensor und Zielwerte
- GUI - Custom Controls

Regulation & GUI Controls

Regulation - PID Regler



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotorXP GUI

Test und Analyse

Simulation

```
#define REGULATION_P_REGULATE(crntValue, targetValue, Kp) Kp * (targetValue - crntValue)
#define REGULATION_I_REGULATE(crntValue, targetValue, regSumPtr, passedTime, Ki) \
    Ki * (*regSumPtr = passedTime * (targetValue - crntValue))
#define REGULATION_D_REGULATE(crntValue, targetValue, lastDifferencePtr, lastDifferenceValue, passedTime, Kd) \
    (Kd * (((*lastDifferencePtr = targetValue - crntValue) - lastDifferenceValue) / passedTime))

#define REGULATION_REGULATE_SINGLE(regulateVariablesPtr, passedTime, crntValue) \
    REGULATION_P_REGULATE(crntValue, regulateVariablesPtr->targetValue, regulateVariablesPtr->Kp) + \
    REGULATION_I_REGULATE(crntValue, regulateVariablesPtr->targetValue, &(regulateVariablesPtr->regSum), passedTime, regulateVariablesPtr->Ki) + \
    REGULATION_D_REGULATE(crntValue, regulateVariablesPtr->targetValue, &(regulateVariablesPtr->lastDifferenceValue), \
        regulateVariablesPtr->lastDifferenceValue, passedTime, regulateVariablesPtr->Kd);

struct Regulation_PidValues
{
    float targetValue; // the desired target value
    float Kp; // degree in how much the p regulator affects the output.
    float Ki; // degree in how much the i regulator affects the output.
    float Kd; // degree in how much the d regulator affects the output.

    float regSum; // for I regulator.
    float lastDifferenceValue; // for d regulator.
};
```


Regulation & GUI Controls

Regulation - Main loop

MotorXP

Projekt Start

Sensorik

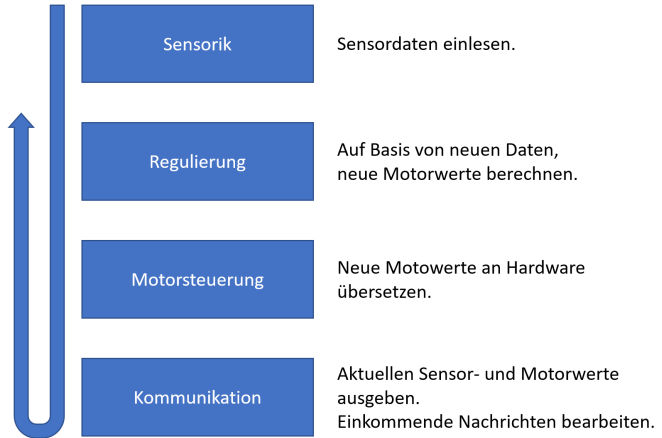
Kommunikation

Regulation & GUI
Controls

MotorXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

Gauge Control



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



```
<Canvas x:Name="PART_BackgroundCanvas"/>
<Canvas x:Name="PART_NeedleCanvas"/>
```

```
2 references | SendMsg, 46 days ago | 1 author, 1 change
private void DrawNeedle() {
    if (_needleCanvas == null) return;

    _needleCanvas.Children.Clear();

    var poly = new Polygon();
    var points = new PointCollection();

    var centerY = _needleCanvas.ActualHeight / 2;
    var centerX = _needleCanvas.ActualWidth / 2;
    // radius = the biggest we can fit into the canvas. So take width or height, whatever is smaller
    var outerRadius = (_backgroundCanvas.ActualWidth > _backgroundCanvas.ActualHeight
        ? _backgroundCanvas.ActualWidth
        : _backgroundCanvas.ActualHeight) / 2; // divide by two because we want the radius

    GaugeSize = outerRadius * 2 - 2; // set GaugeSize for TextFields
    var needleLength = outerRadius * NeedleLength;

    // draw needle like this >
    points.Add(new Point(centerX, centerY - NeedleWidth / 2));
    // draw back
    points.Add(new Point(centerX - NeedleBackLength, centerY));
    // lower part of arrow
    points.Add(new Point(centerX, centerY + NeedleWidth / 2));
    // needle top
    points.Add(new Point(centerX + needleLength, centerY));

    poly.Points = points;
    poly.Fill = NeedleColor;
    poly.Stroke = NeedleBorderBrush;
    poly.StrokeThickness = 1;

    _needleCanvas.Children.Add(poly);
}
```

Regulation & GUI Controls

Gauge Control - Drehfunktion



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

```
4 references | BerndIK, 48 days ago | 1 author, 1 change
private void RotateNeedle() {
    if (_needleCanvas == null) return;

    var percent = ((Value - MinValue) / (MaxValue - MinValue));
    var angle = MinAngle + (MaxAngle - MinAngle) * percent;
    angle -= Math.PI / 2; // adjust by 90 degrees (since we did the same thing in GetPoint, so that 0 = straight up)

    var centerY = _needleCanvas.ActualHeight / 2;
    var centerX = _needleCanvas.ActualWidth / 2;
    var rotateTransform = new RotateTransform(angle / Math.PI * 180, centerX, centerY);

    _needleCanvas.RenderTransform = rotateTransform;
}
```

Regulation & GUI Controls

LineChart Control



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



LineChart Control - Mehr Pixel als Sample 1

Simulation



Regulation & GUI Controls

LineChart Control - Mehr Pixel als Sample 2



Projekt Start

Sensorik

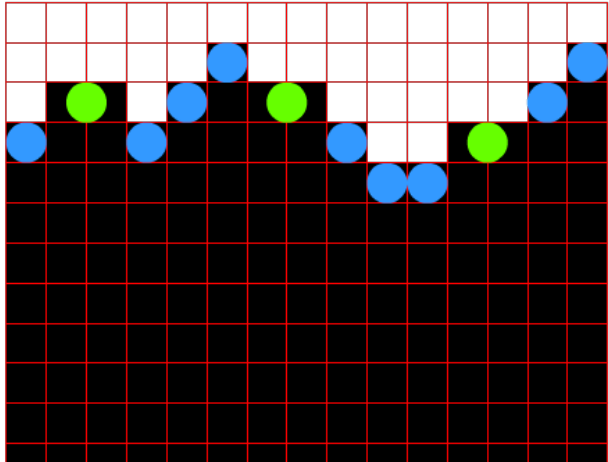
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

LineChart Control - Mehr Sample als Pixel

MotorXP

Projekt Start

Sensorik

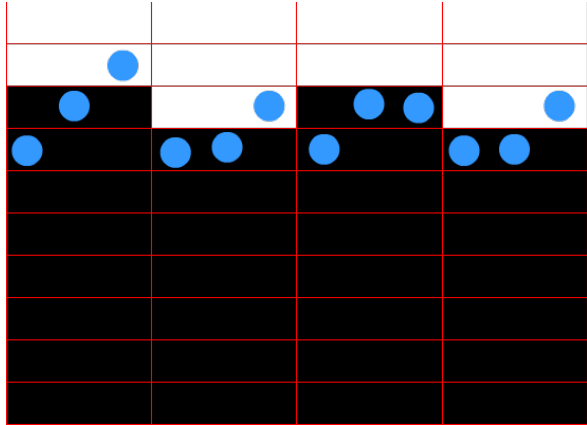
Kommunikation

Regulation & GUI
Controls

MotorXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

LineChart Control - Mittelwert Rechnung 1



Projekt Start

Sensorik

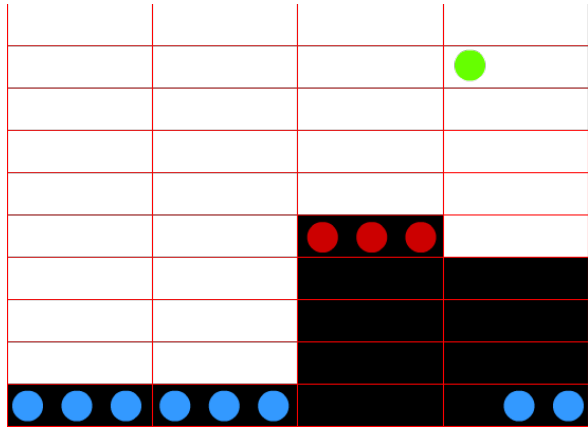
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

LineChart Control - Mittelwert Rechnung 2



Projekt Start

Sensorik

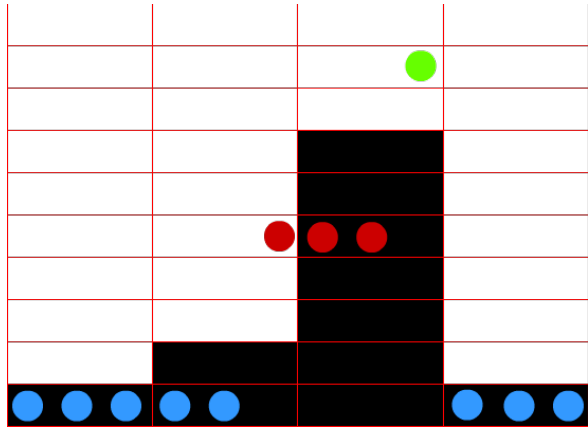
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

LineChart Control - Mittelwert Rechnung 3

MotorXP

Projekt Start

Sensorik

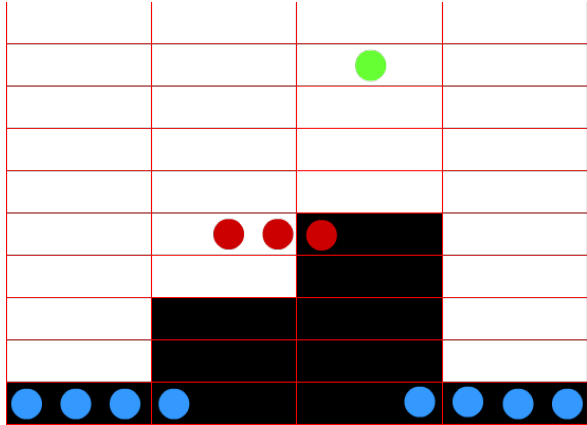
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

LineChart Control - Min/Max Rechnung 1

MotorXP

Projekt Start

Sensorik

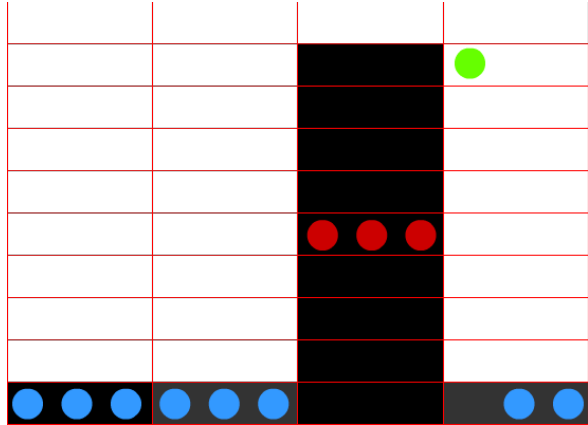
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

LineChart Control - Min/Max Rechnung 2

MotorXP

Projekt Start

Sensorik

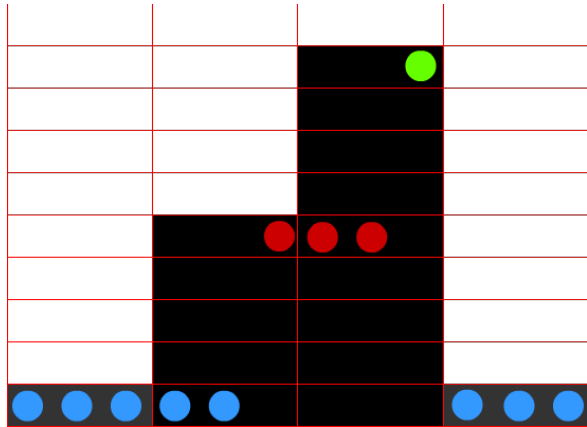
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

LineChart Control - Min/Max Rechnung 3

MotorXP

Projekt Start

Sensorik

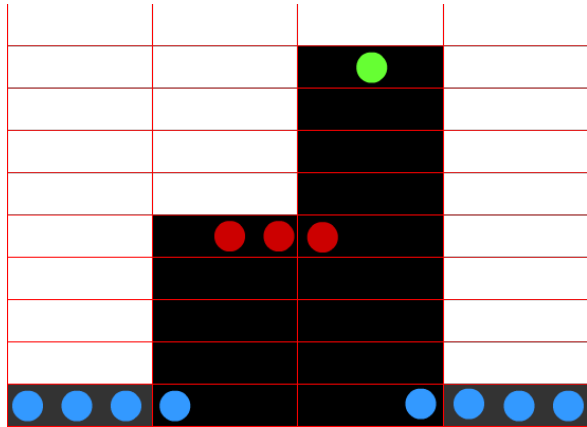
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

LineChart Control - Min/Max Rechnung 4

MotorXP

Projekt Start

Sensorik

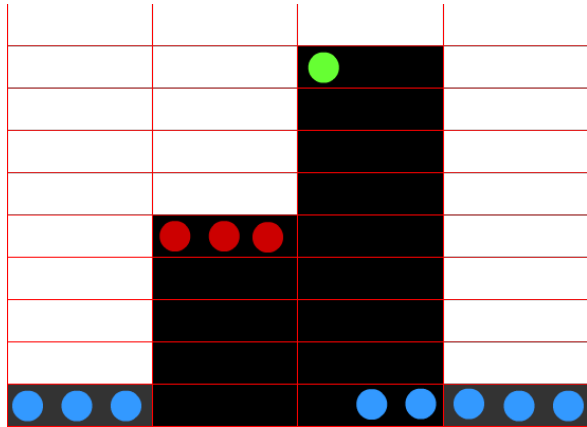
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

LineChart Control - Min/Max Rechnung 5

MotorXP

Projekt Start

Sensorik

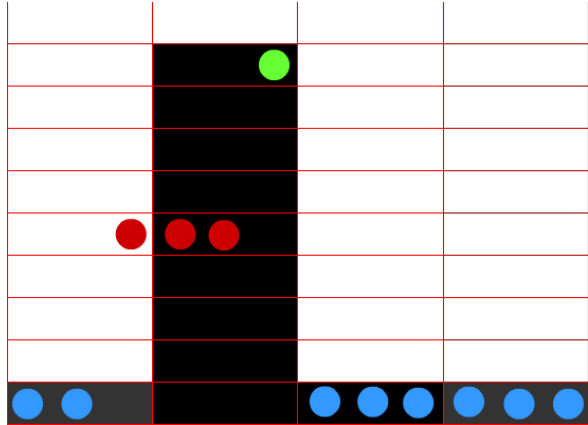
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Regulation & GUI Controls

Plakat

MotorXP

Projekt Start

Sensorik

Kommunikation

Regulation & GUI Controls

MotrXP GUI

Test und Analyse

Simulation



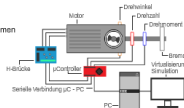
Das Projektziel ist die Entwicklung und der Aufbau eines Motorexperimentierplatzes für BLDC – Motoren.

Features

- Simulation mit Simulink
- Verschiedene Regelalgorithmen
- Serielle Kommunikation
- Parametervisualisierung

Messung

- Drehzahl
- Drehmoment
- Drehwinkel



Controller

- Infineon XMC
- Infineon XE167

Sensoren

- Hall Sensor
- Drehzahl Encoder
- Drehmomentsensor

Motorkit

- Texas Instruments DRV8301-HC-C2-KIT

Drehzahl Sensorik
Controller
Mikrochip
Motor
Regelung
Datenverarbeitung
Drehmoment



OSTBAYERISCHE
TECHNISCHE HOCHSCHULE
REGENSBURG

MotrXP GUI

Anforderungen



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

Funktionale Anforderungen:

- Anzeige der Sensordaten
- Regelung der Drehgeschwindigkeit
- Einstellung des PID Reglers

Nicht-Funktionale Anforderungen:

- Modulares erweiterbares System
- Modernes Metro Design

MotrXP GUI

Entwurf

MotorXP

Projekt Start

Sensorik

Kommunikation

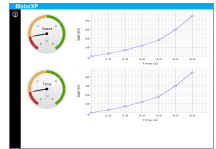
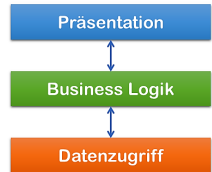
Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Drei Schichten Architektur
- Entwurfsmuster
- DatenStrukturen
- Mockup



MotrXP GUI

Implementierung



Projekt Start

Sensorik

Kommunikation

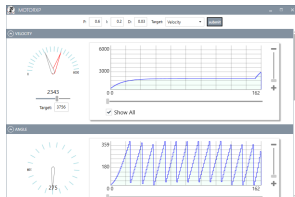
Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- MVVM-Light Framework
- MahApps Metro UI Toolkit
- Custom Controls



MotrXP GUI

Ausblick

MotorXP

Projekt Start

Sensorik

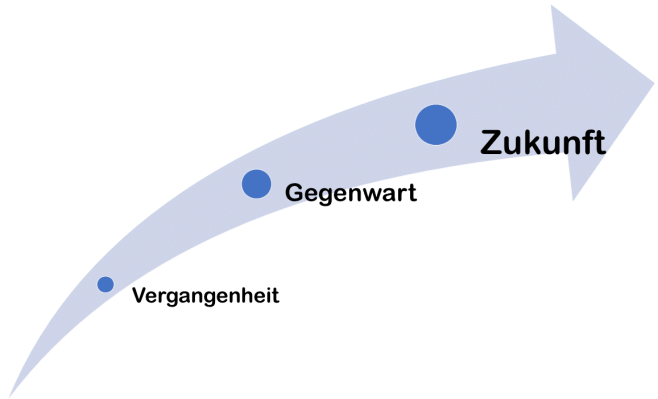
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Test und Analyse

Übersicht



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Spannungssignal
- Hall-Sensoren
- XMC 4700 Relax Kit
- Steuerung BLDC

Test und Analyse

Spannungsmessung Fremderregung

MotorXP

Projekt Start

Sensorik

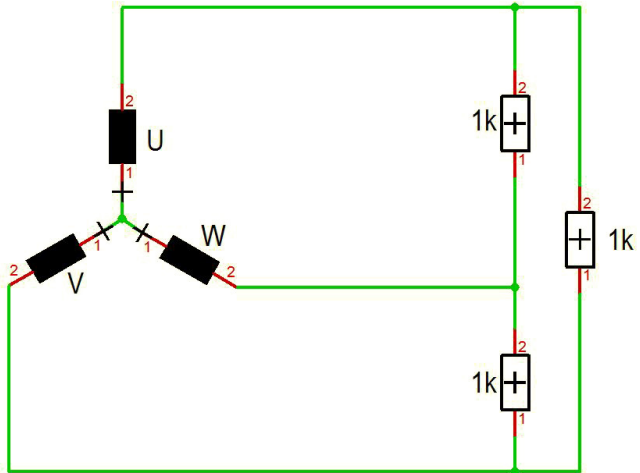
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Test und Analyse

Analyse Spannungsmessung Fremderregung

MotorXP

Projekt Start

Sensorik

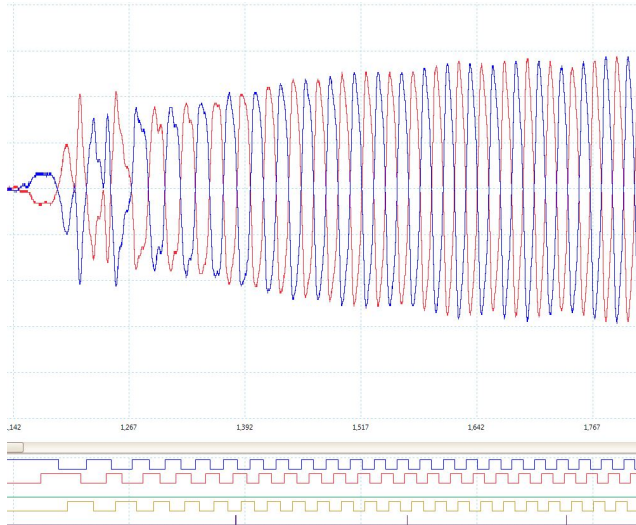
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Test und Analyse

Analyse Spannungsmessung Fremderregung mit Störsignalen

MotorXP

Projekt Start

Sensorik

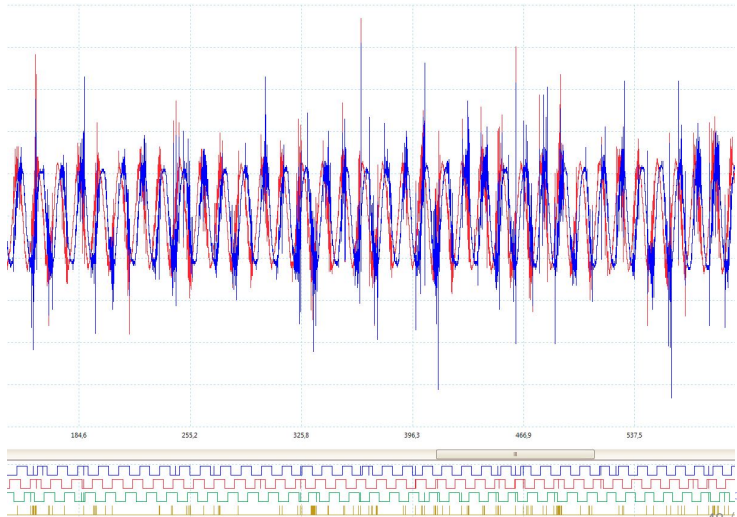
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Test und Analyse

Hall-Sensoren in Verbindung mit dem XMC 4700 Relax Kit 5V



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- XMC 4700 Relax Kit 5V
 - Vorbereitet für Arduino Shields (5V)
 - Pegelwandler (5V \leftrightarrow 3.3V)
 - Störempfindlich gegenüber längeren Messleitungen (\approx 20 cm)

Test und Analyse

Hall-Sensoren in Verbindung mit dem XMC 4700 Relax Kit 5V

MotorXP

Projekt Start

Sensorik

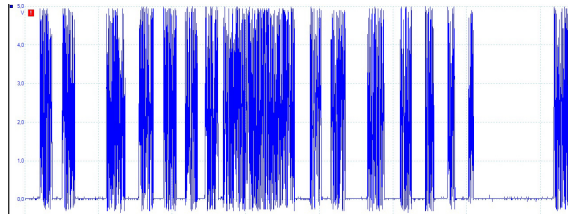
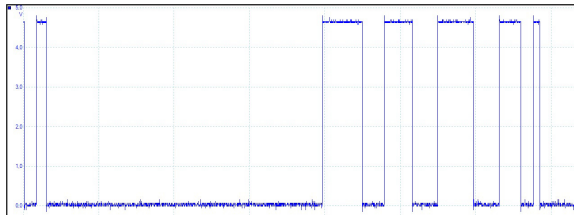
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Test und Analyse

Hall-Sensoren in Verbindung mit dem XMC 4700 Relax Kit 5V



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- XMC 4700 Relax Kit 5V
 - Vorbereitet für Arduino Shields (5V)
 - Pegelwandler (5V \leftrightarrow 3.3V)
 - Störempfindlich gegenüber längeren Messleitungen (\approx 20 cm)
- \rightarrow Messleitung verkürzen
- \rightarrow Pegelwandler austauschen entfernen

Test und Analyse

Steuerung BLDC - Schaltung

MotorXP

Projekt Start

Sensorik

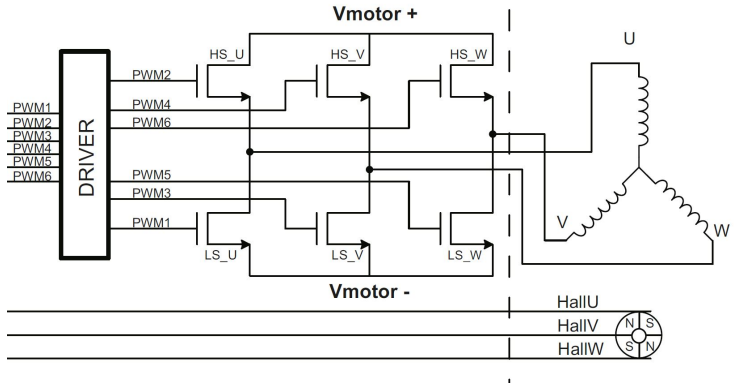
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Test und Analyse

Steuerung BLDC - Steuersignal



Projekt Start

Sensorik

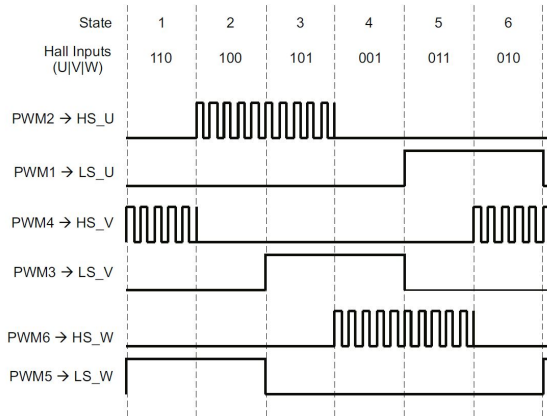
Kommunikation

Regulation & GUI Controls

MotrXP GUI

Test und Analyse

Simulation



Test und Analyse

Steuerung BLDC - Texas Instrument



Projekt Start

Sensorik

Kommunikation

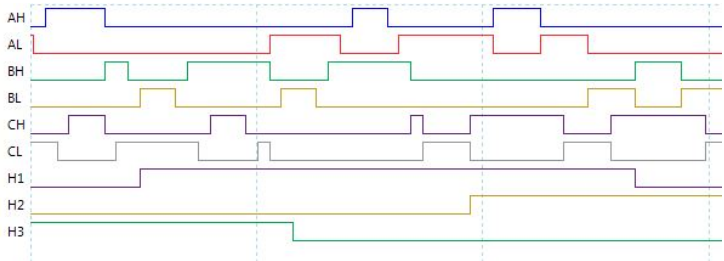
Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Abweichende Signale
- Zwei Ausgänge gleichzeitig HIGH



Simulation

Anforderungen



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Kommunikation mit GUI
- Simulation in Echtzeit
- Kommunikation mittels serieller Schnittstelle

Simulation

Entwurfsphase und Implementierung I

MotorXP

Projekt Start

Sensorik

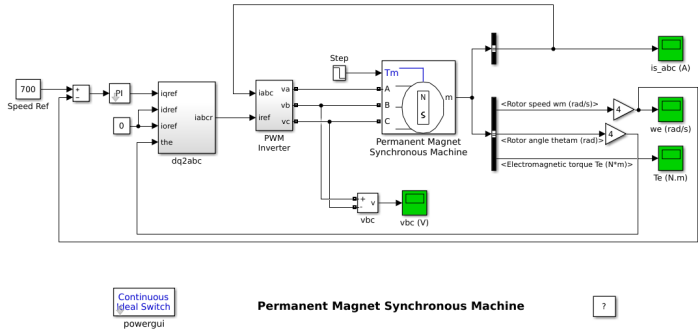
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Simulation

Analysephase II

MotorXP

Projekt Start

Sensorik

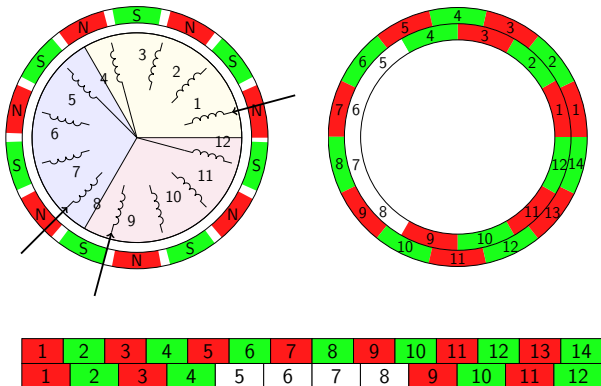
Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation



Simulation

Bewertung

MotorXP

Projekt Start

Sensorik

Kommunikation

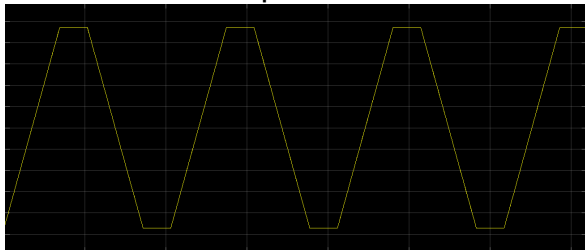
Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

- Resultierendes Spulenfeld



$$\sum_1^{28} V_{res_i} = \sum_1^{28} (V_i + R_i) = \sum_1^{28} V_i + \sum_1^{28} R_i$$

Simulation

Bewertung



Projekt Start

Sensorik

Kommunikation

Regulation & GUI
Controls

MotrXP GUI

Test und Analyse

Simulation

● Drehmoment

