

Robert Mille Rapid Control Prototyping
eines ASA-Controllers mit
MATLAB PLC Coder

ASA ist die Abkürzung von *Antisystem-Approach*. Dieser Begriff wurde von Dr. S. Zacher eingeführt und zum Entwurf eines neuartigen modellbasierten Reglers angewendet.

Ein ASA-Regler unterscheidet sich von konventionellen Kompensationsreglern dadurch, dass dafür keine Reziproke des Strecken-Modells gebildet werden muss. Das schwierig zu realisierende inverse Streckenmodell entfällt und wird durch eine Kreisschaltung ersetzt, so dass die Regelstrecke vollständig kompensiert wird, ohne dabei die differenzierenden Anteile im Regelkreis zu generieren.

Das Buch stellt eine Weiterentwicklung des ASA-Reglers dar, bei dem es z. Z. keine fertigen Bausteine gibt, sodass eine Entwicklung in einer Spezialsoftware unabdingbar ist. Die Schnittstelle zum Steuerungssystem wird hierbei durch den Einsatz des PLC Coders optimiert und somit wird der Entwicklungsprozess einhergehend beschleunigt. Der mit dem PLC Coder generierte ST-Code wird im Buch beim Prozessleitsystem ABB Freelance in einem Beispiel einer Füllstandsregelung implementiert, um die Funktionalität des ASA-Reglers zu testen.

Das Buch entstand aus der Master-Arbeit des Verfassers während seines aus sechs Semestern bestehenden Fernstudiums Master of Science mit der Vertiefung Automatisierung an der Hochschule Darmstadt, Fachbereich Elektrotechnik und Informationstechnik.

Das Buch richtet sich an Ingenieure, die in der Praxis mit Regelungssystemen arbeiten, an Studierende der Hoch- und Fachhochschulen sowie an Schüler der Technikausbildung aus den Fachbereichen Elektrotechnik und Automatisierungstechnik, die bereits grundlegende Kenntnisse in der Regelungstechnik und fundierte Vorkenntnisse in der Programmierung haben.



ISBN 978-3-937638-28-7



9 783937 638287

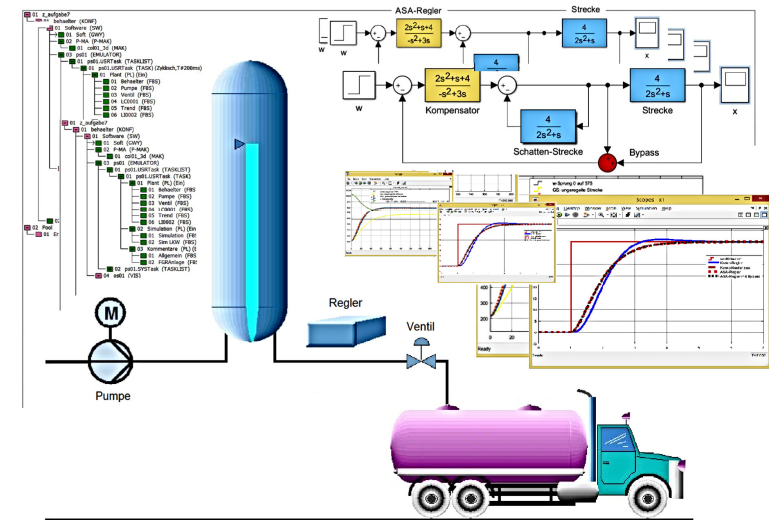
R. Mille

RCP eines ASA-Controller mit MATLAB PLC Coder

N

Robert Mille

RAPID CONTROL PROTOTYPING
eines ASA-Controllers mit
MATLAB PLC Coder



Robert Mille
Rapid Control Prototyping
eines ASA-Controllers mit
MATLAB PLC Coder



Inhaltsverzeichnis

1	Einleitung	7
2	Rapid Control Prototyping	9
3	Grundlagen der modellbasierten Regelung	13
3.1	Der Kompensationsregler	13
3.2	Antisystem-Approach	15
3.2.1	ASA-Regler mit Vorfilter	21
3.2.2	ASA-Regler mit Kompensator	23
4	MATLAB Simulink PLC Coder	25
5	Aufgabenstellung	29
6	Beschreibung des realen Systems	31
6.1	Funktionelle Beschreibung des Systems	32
6.2	Beschreibung des Systems im ABB Freelance Emulator	32
7	Modellbildung des Systems	35
7.1	Beschreibung des Modellierungszieles	35
7.2	Auswahl der Modellannahmen	35
7.3	Verbale Beschreibung der Regelstrecke	36
7.4	Aufstellung des Blockschaltbildes	36
7.5	Aufstellung der Modellgleichung	37
7.6	Modellvalidierung	41
8	Entwurf eines konventionellen Reglers	43
8.1	Entwicklung und Untersuchung eines PD-Reglers	43
8.2	Entwurf eines PD-Reglers durch einen vorgefertigten Freelance Baustein	47
8.3	Entwurf eines PD-Reglers durch einen selbstgestellten Freelance Baustein mit ST Code unter Verwendung des MATLAB Simulink PLC Coders	51

9	Modellgestützte Entwurfsverfahren	61
9.1	Entwicklung und Untersuchung eines modellbasierten Reglers nach dem Kompensationsprinzip .	61
9.2	Entwurf eines modellbasierten Reglers nach dem Kompensationsprinzip durch einen selbsterstellten Freelance Baustein mit ST Code unter Verwendung des MATLAB Simulink PLC Coders	66
9.3	Entwicklung und Untersuchung eines modellbasierten Reglers nach dem Antisystem-Approach .	74
9.4	Entwurf eines modellbasierten Reglers nach dem Antisystem-Approach durch einen selbsterstellten Freelance Baustein mit ST Code unter Verwendung des MATLAB Simulink PLC Coders	78
10	Vergleich zwischen konventioneller Regelung und ASA-Regelung	87
10.1	Vergleich der Blockschaltbilder und Sprungantworten von den Kompensationsreglern und den ASA-Reglern	87
10.2	Vergleich des Führungs- und Störverhalten der Kompensationsregler und ASA-Regler	89
10.3	Vergleich zwischen PD-Regler, Kompensationsregler und ASA-Regler mit Bypass als Split-Range-Regelung mit Begrenzern	92
10.4	Vergleich zwischen dem Z-transformierten PD-Regler, Kompensationsregler und ASA-Regler mit Bypass als Split-Range-Regelung mit Begrenzern	94
11	Schlussbetrachtung und weiterführende Forschung	97
12	Abbildungsverzeichnis	101
13	Literaturverzeichnis	107

1 Einleitung

Die elektronischen Systeme sind heutzutage allgegenwärtig und nicht mehr aus unserem Alltag wegzudenken. Dabei lief die Ausprägung der Elektronik bis zum heutigen Stand der Technik eindrucksvollerweise gerade einmal in einem halben Jahrhundert ab. Aber nicht nur die Technik selbst, sondern auch die einhergehenden Entwicklungsmethoden haben sich parallel dazu weiterentwickelt, denn die Life-Cycles und die Time to Market sind dementsgegen rasant verkürzt. Es ist daher logisch, dass Schnittstellen zwischen Systemen entweder möglichst gering gehalten werden müssen, oder in einem entsprechenden Maße zu optimieren sind. Damit ist gemeint, dass die Trennung zwischen Entwicklungssystem und tatsächlicher Steuerung zwar durchaus als sinnvoll zu erachten ist, der Austausch zwischen diesen beiden hingegen als eine Art Medienbruch zu sehen ist, der durch geeignete Methoden die Systeme stärker miteinander verschmelzen soll. Speziell beim Rapid Control Prototyping ist ein iterativ ablaufender Entwicklungsprozess mit Simulationen und Tests zu jedem Zeitpunkt gewünscht. Für diesen Ansatz ist es unerlässlich, eine „durchgängige Werkzeugkette“ [AB13] bereitzustellen. The MathWorks Inc. hat mit dem PLC Coder ein Zusatzwerkzeug für MATLAB Simulink entwickelt, welches genau zu diesem Thema eine Lösung bietet. Das Programm dient als Schnittstelle zwischen dem Entwicklungswerkzeug MATLAB Simulink und einem Steuerungssystem, welches die Programmierung von strukturiertem Text gemäß Richtlinie IEC 61131-3 erlaubt. Mit dieser Verfahrensweise können in MATLAB Simulink Prozesse modelliert und Regler entsprechend entworfen werden, um sie danach mit Hilfe des PLC Coders in strukturierten Text umzuwandeln und in einer Steuerung oder einem Leitsystem zu integrieren.

Üblicherweise gibt es die grundlegenden Regler bereits in den meisten Systemen als vorgefertigte Algorithmen, im Falle eines speziellen Reglers muss man diese Berechnungen neu verfassen. Bei dem von Prof. Dr. Zacher entwickelten ASA-Regler handelt es sich um einen speziellen modellbasierten Regler, der unter Einbezug des Streckenmodells eine Übertragungsfunktion bildet.

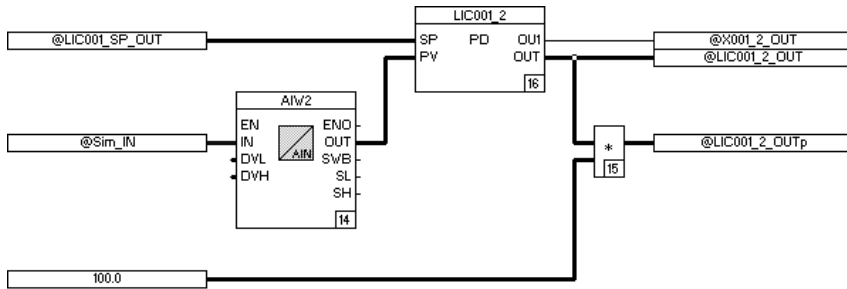


Abbildung 8.14: Konfiguration des neu erstellten PD-Reglers im FBS-Blatt LIC001

wurde in dem Bedienbild der Anlage ein Auswahlschalter hinzugefügt. Abbildung 8.15 zeigt das neu gestaltete Anlagenbild, das im Fortlauf der Arbeit stets um die neu erstellten Regler erweitert werden kann.

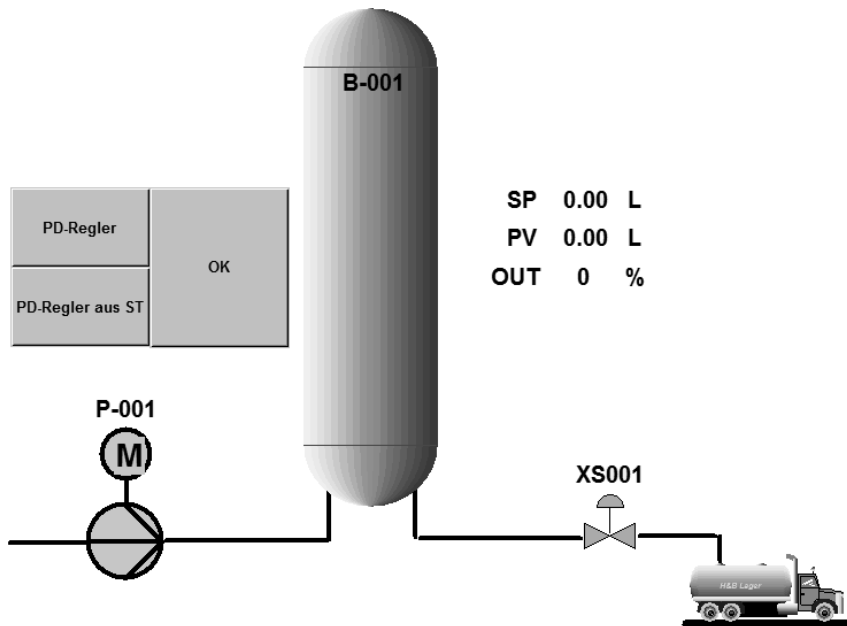


Abbildung 8.15: Bedienbild mit Auswahlschalter der konfigurierten Regler

9 Modellgestützte Entwurfsverfahren

Die in Kapitel 3 vorgestellten Grundlagen der modellbasierten Softwareentwicklung sollen nachfolgend Anwendung finden. Zuerst wird der Kompensationsregler und danach der ASA-Regler betrachtet.

9.1 Entwicklung und Untersuchung eines modellbasierten Reglers nach dem Kompensationsprinzip

In Kapitel 3.1 wurde die Vorgehensweise zur Entwicklung eines Kompensationsregler erläutert. Im ersten Ansatz wird das gewünschte Verhalten in der Übertragungsfunktion $G_M(s)$ definiert. Es wird $G_M(s)$ mit einem PT2-Verhalten mit aperiodischem Verlauf gewählt, Formel (9.1). Das Verhalten der gewünschten Übertragungsfunktion als Reaktion auf einen Eingangssprung ist in Abbildung 9.1 dargestellt.

$$G_M(s) = \frac{1}{(0.5s + 1)^2} = \frac{1}{0.25s^2 + s + 1} \quad (9.1)$$

Zusätzlich wird die reziproke Strecke gebildet und in den Kompensationsregler integriert. Sowohl für die Simulation als auch für die Realisierung in der Anlage ist die reziproke Strecke problematisch und nicht realisierbar, weil die meisten Strecken ein höheres Nennerpolynom besitzen, welches im Kehrwert zu einem hohen differentiellen Anteil führt. Insbesondere bei großen Zeitkonstanten im I-Anteil einer Strecke entsteht in der Reziproken ein großer D-Anteil.

$$\frac{1}{G_S(s)} = \frac{s(1 + sT_1)}{K_{IS}}$$

9.3 Entwicklung und Untersuchung eines modellbasierten Reglers nach dem Antisystem-Approach

Bereits in Kapitel 3.2 wurde die Theorie und die Herleitung des ASA-Reglers beschrieben. In dem folgenden Kapitel wird die Theorie angewandt, indem zunächst der ASA-Regler für den Behälter B-001 sowohl entwickelt als auch in der Simulation getestet und anschließend über den MATLAB Simulink PLC Coder für den Einsatz in Freelance umgewandelt wird. Durch geschickten Einsatz der Kreisschaltungen wurde gezeigt, dass der ASA Regler eine vollständige Kompensation liefert, mit dem Nachteil, dass am Ausgang des Regelkreises das Signal $x(t) + y(t)$ entsteht. Für das Absondern des reinen x-Signals wurden zwei Methoden vorgestellt, nämlich die des Vorfilters und die des Kompensators. Letztere erzielt einen entscheidenden Vorteil durch die Integration des Kompensators in den geschlossenen Regelkreis und der damit verbundenen Möglichkeit mit der Schattenstrecke zusammengefasst werden zu können – die Gründe hierfür werden im weiteren Verlauf beschrieben. Ferner wurde zuvor bei der Herleitung von $G_K(s)$ festgehalten, dass sich dieser als Kreisschaltung darstellen lässt und demgemäß die Berechnung nach Formel (3.9) auf Seite 24 grundlegend nicht notwendig ist. Durch eine Simulation mit MATLAB werden jedoch Probleme mit nichtverzögerten Rückkopplungen deutlich, sodass auf die Darstellung als Kreisschaltung verzichtet wird und der Kompensator entsprechend mathematisch berechnet wird¹.

Das gewünschte Verhalten $G_M(s)$ und die Strecke $G_S(s)$ sind identisch mit den vorherigen Übertragungsfunktionen, sie werden hingegen für die Berechnung in allgemeiner Darstellungsweise eingesetzt.

$$G_M(s) = \frac{K_{PS}}{(1 + sT_w)^2}$$

¹In MATLAB sind die nichtverzögerten Rückkopplungen beschrieben als algebraic loops. Es gibt einige Lösungsansätze in der MATLAB Dokumentation [The16] und in [Zac16b].

10 Vergleich zwischen konventioneller Regelung und ASA-Regelung

Im abschließenden Kapitel werden die entwickelten Regler untereinander hinsichtlich ihrer Schnelligkeit und Genauigkeit verglichen. Für die Analyse wurden vier gesonderte MATLAB Simulink Modelle aufgestellt mit der Absicht, das Führungs- und Störverhalten, den Unterschied unter Nutzung der Split-Range-Methode mit Begrenzern und die Auswirkungen der Z-Transformierten Reglerübertragungsfunktionen umfangreich zu beleuchten.

10.1 Vergleich der Blockschaltbilder und Sprungantworten von den Kompensationsreglern und den ASA-Reglern

In Abbildung 10.1 sind jeweils zwei Varianten der modellbasierten Regler aufgeführt, wobei die erste die Standardausführung und die zweite die erweiterte, zusammengefasste Reglerstruktur präsentiert. Der Kompensationsregler an oberster Stelle besteht aus dem gewünschten Verhalten und der Bilanz aus reziproker Strecke und Streckenmodell, allerdings sind bei der inversen Strecke zusätzliche Zeitkonstanten im Nenner der Übertragungsfunktion hinzugefügt, da die Simulation andernfalls nicht ausführbar ist. Die Erweiterung zeigt einen zusammengefassten Kompensationsregler ohne die Notwendigkeit zusätzliche Verzögerungen zu implementieren.

Die ASA-Regler haben einen differierenden Aufbau mit einer Vorwärtskopplung bzw. mit einem Kurzschluss der Strecke. In der

10.4 Vergleich zwischen dem Z-transformierten PD-Regler, Kompensationsregler und ASA-Regler mit Bypass als Split-Range-Regelung mit Begrenzern

Schließlich sind in Abbildung 10.9 die drei Regler in Subsysteme verpackt und Z-transformiert, um einen Vergleich der digitalen Regelung zu ermöglichen. Innerhalb der jeweiligen Subsysteme befinden sich die Regler, die als Vorbereitung zur ST Codegenerierung Z-transformiert wurden. Der Kurvenverlauf in Abbildung 10.10 zeigt, dass die drei Regler den Sollwert unterschiedlich schnell erreichen. In dem Auszug aus Abbildung 10.11 ist hin-

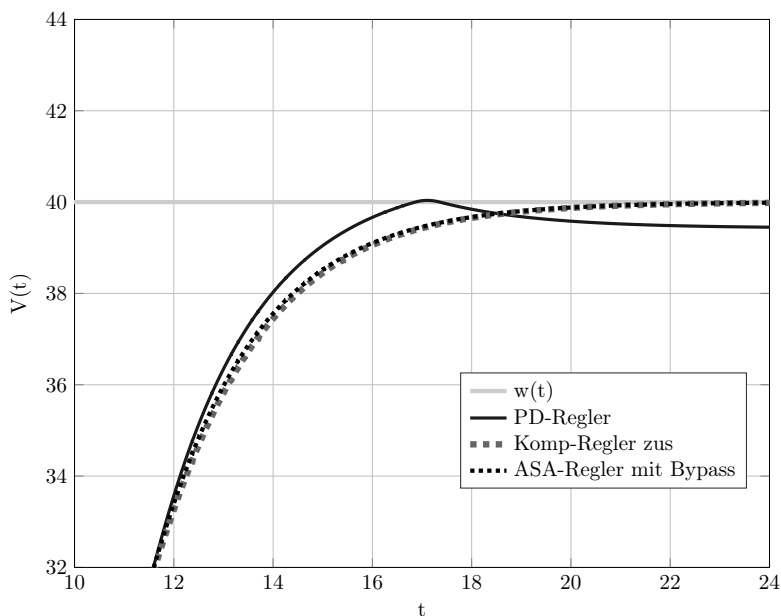


Abbildung 10.8: Auszug der Sprungantworten von PD-Regler, Kompensationsregler und ASA-Regler mit Bypass als Split-Range-Regelung mit Begrenzern

11 Schlussbetrachtung und weiterführende Forschung

In den vorangestellten Kapiteln wurde der Entwurf und die Implementierung der von Prof. Dr. Zacher entwickelten Antisystem-Approach-Regelung für eine Füllstandsregelung in der Methode des Rapid Control Prototyping untersucht.

Die Modellierung des Systems ergab ein integrierendes Streckenverhalten mit der Besonderheit einer aufgeteilten Regeldifferenz, die durch zwei Stellglieder geregelt werden kann. Beide Stellglieder sind darüberhinaus in ihrem Stellwert begrenzt. Die Pumpe P-001 gilt dabei als regelbar, während das Ventil nur binär arbeitet.

Nach Aufstellung der Systemgleichung wurde zunächst ein konventioneller Regler für die Regelung des Behälterfüllstandes ausgelegt. Für das integrierende Streckenverhalten eignete sich ein PD-Regler zur Erreichung der Regelungsvorgaben, da durch ihn keine Überschwinger und eine kleine Ausregelzeit erreicht werden konnten. Meist sind die konventionellen Regler bereits als fertige Bausteine in Steuerungssystemen vorhanden, sodass diese nur mit den Signalen verknüpft werden müssen. Aus diesem Grund wurde daraufhin in Freelance ein PD-Regler nach den Erkenntnissen aus dem MATLAB Simulink Modell für die Regelung des Systems eingesetzt. Durch die Aufteilung der Regeldifferenz war dennoch zusätzliche Konfiguration in Freelance erforderlich. Das Ergebnis der Regelung in Freelance wurde ohne signifikante bleibende Regeldifferenz aufgezeichnet.

Um gleichzeitig den Ablauf des Rapid Control Prototyping in Verbindung mit dem ST Coder zu prüfen, wurde der PD-Regler in MATLAB Simulink in strukturierten Text umgewandelt. Zuvor wurde das Modell in MATLAB entsprechend der Anforderungen des PLC Coders abgeändert – der PD-Regler musste diskretisiert in ein Subsystem als „atomic unit“ verpackt werden.

Für den Einsatz des Codes in Freelance musste eine neue Bausteinklasse erzeugt werden, in dem der ST Code platziert wurde.

13 Literaturverzeichnis

- [AB13] ABEL, Dirk ; BOLLIG, Alexander: *Rapid Control Prototyping - Methoden und Anwendungen*. Berlin Heidelberg : Springer-Verlag, 2013
- [ABB16] ABB: *Freelance Engineering Hilfe*. 2016
- [ABRW09] ANGERMANN, Anne ; BEUSCHEL, Michael ; RAU, Martin ; WOHLFARTH, Ulrich: *MATLAB® - Simulink® - Stateflow® - Grundlagen, Toolboxen, Beispiele*. 6. Auflage. München : Oldenbourg, 2009
- [JT09] JOHN, Karl-Heinz ; TIEGELKAMP, Michael ; VDI (Hrsg.): *SPS-Programmierung mit IEC 1131-3 - Konzepte und Programmiersprachen, Anforderungen an Programmiersysteme, Entscheidungshilfen*. 4. neubearbeitete Auflage. Berlin Heidelberg : Springer-Verlag, 2009
- [Lun08] LUNZE, Prof. Dr.-Ing J.: *Regelungstechnik 1 - Systemtheoretische Grundlagen Analyse und Entwurf einschleifiger Regelungen*. 7. Auflage. Berlin Heidelberg : Springer-Verlag, 2008
- [Sam14] SAMAL, Erwin: *Grundriss der praktischen Regelungstechnik - bearbeitet von Dirk Fabian und Christian Spieker*. 22. Auflage. München : De Gruyter Oldenbourg, 2014
- [Sch07] SCHULZ, Gerd: *Regelungstechnik 1 - Lineare und Nichtlineare Regelung, Rechnergestützter Reglerentwurf*. 3. Auflage. München : Oldenbourg, 2007
- [Sch08] SCHULZ, Gerd: *Regelungstechnik 2 - Mehrgrößenregelung, Digitale Regelungstechnik, Fuzzy-Regelung*. 2. Auflage. München : Oldenbourg, 2008

- [The] THE MATHWORKS: *Visualizing Real-Time Simulations*.
<http://de.mathworks.com/products/3d-animation/features.html#visualizing-real-time-simulations>. – [Online; Stand 25. September 2016]
- [The15] THE MATHWORKS: *Simulink[®] PLC Coder[™] User's Guide*. 2015
- [The16] THE MATHWORKS: *Simulink[®] Documentation*. Version: 2016.
<https://de.mathworks.com/help/simulink/>. – [Online; Stand 01. Oktober 2016]
- [Zac12] ZACHER, Prof. Dr. S.: *Regelungstechnik Aufgaben - Lineare, Zweipunkt- und digitale Regelung*. 3. Auflage. Wiesbaden : Dr. Serge Zacher Verlag, 2012
- [Zac15a] ZACHER, Prof. Dr. S.: Automation-Letter Nr. 5 - Model Discretizer von MATLAB - Hinweise zur Digitalisierung von analogen Simulink-Modellen. (2015). – Veröffentlicht auf www.zacher-automation.de
- [Zac15b] ZACHER, Prof. Dr. S.: *MBSE - Leitfaden zur Lehrveranstaltung Modellbasierte Softwareentwicklung*. 2015
- [Zac15c] ZACHER, Prof. Dr. S.: Software-in-the-Loop und Prozessor-in-the-Loop Simulationen mit MATLAB[®]/ Simulink[®]. 2015. – Forschungsbericht
- [Zac16a] ZACHER, Prof. Dr. S. ; ZACHER, Verlag D. (Hrsg.): *ASA-Control - Antisystem-Approach for Closed Loop Control*. Verlag Dr. Zacher, 2016
- [Zac16b] ZACHER, Prof. Dr. S.: Automation-Letter Nr. 15 - MATLAB-Meldung Algebraic loop bei Simulation eines modellbasierten ASA-Reglers. (2016). – Veröffentlicht auf www.zacher-automation.de
- [Zac16c] ZACHER, Prof. Dr. S.: Automation-Letter Nr. 16 - Das gewünschte Verhalten - Zum Entwurf von modellbasierten Reglern. (2016). – Veröffentlicht auf www.zacher-automation.de

- [Zac16d] ZACHER, Prof. Dr. S.: Automation-Letter Nr. 25 - ASA-Regler für I-Strecke - Hinweise zur Projektarbeit ASA-Regler für OSLO-Modell von ABB. (2016). – Veröffentlicht auf www.zacher-automation.de
- [Zac16e] ZACHER, Prof. Dr. S.: Automation-Letter Nr. 26 - PLC Coder von MATLAB für PLS Freelance von ABB. (2016). – Veröffentlicht auf www.zacher-automation.de
- [Zac16f] ZACHER, Prof. Dr. S.: Automation-Letter Nr. 29 - ASA-Regler mit Bypass - Modellbasierter Kompensationsregler mit einer überbrückten Regelstrecke. (2016). – Veröffentlicht auf www.zacher-automation.de
- [Zac16g] ZACHER, Prof. Dr. S.: Automation-Letter Nr. 6 - Die unwirksame Regelstrecke - Eine Studie über Kompensationsstrukturen für die modellbasierte Regelung. (2016). – Veröffentlicht auf www.zacher-automation.de
- [Zac16h] ZACHER, Prof. Dr. S.: Automation-Letter Nr. 7 - Regelung mit der Schatten-Strecke - Ein modellbasierter Regler mit Doppelstrecke nach dem ASA-Prinzip. (2016). – Veröffentlicht auf www.zacher-automation.de
- [Zac16i] ZACHER, Prof. Dr. S.: Automation-Letter Nr. 8 - ASA-Implementierung - Optionen eines modellbasierten Reglers nach dem Antisystem-Approach. (2016). – Veröffentlicht auf www.zacher-automation.de
- [ZR14] ZACHER, Serge ; REUTER, Manfred: *Regelungstechnik für Ingenieure - Analyse, Simulation und Entwurf von Regelkreisen*. 14. Auflage. Wiesbaden : Springer-Verlag, 2014
- [ZS10] ZACHER, Prof. Dr. S. ; SAEED, Waqar: Design of multivariable control systems using Antisystem-Approach - Internationaler Vergleich 2013. FH Technikum Wien : AALE, 2010