



# TU Clausthal

Clausthal University of Technology

## **FPGA-Basisplattform für schnelle Regelungs- und Steuerungsaufgaben**

**A. v. Daake, C. Pelczar**

Technical Report Series

Fac3-11-01



Faculty of  
Mathematics/Computer Science  
and Mechanical Engineering  
Clausthal University of Technology

## Impressum

Publisher: Fakultät für Mathematik/Informatik und Maschinenbau,  
Technische Universität Clausthal  
Am Regenbogen 15, 38678 Clausthal-Zellerfeld, Germany

Editor-in-chief: Alfons Esderts

Technical editor: Martina Wächter

Contact: [martina.waechter@tu-clausthal.de](mailto:martina.waechter@tu-clausthal.de)

URL: <http://www.fakultaet3.tu-clausthal.de/forschung/technical-reports/>

ISSN: 1869-8018

## The Faculty of Mathematics/Computer Science and Mechanical Engineering Review Board

Prof. Dr. Frank Endres

Prof. Dr. Alfons Esderts

Prof. Dr. Stefan Hartmann

apl. Prof. Dr. Günter Kemnitz

Prof. Dr. Armin Lohrengel

Prof. Dr. Norbert Müller

Prof. Dr. Volker Wesling

Prof. Dr. Oliver Zirn

# FPGA-Basisplattform für schnelle Regelungs- und Steuerungsaufgaben

A. v. Daake  
C. Pelczar

TU Clausthal - Institut für Prozess- und Produktionsleittechnik (IPP)  
Arnold-Sommerfeld-Str. 1  
38678 Clausthal-Zellerfeld  
05323/727500

## Abstract

Aufgrund zunehmender Komplexität der Einsatzgebiete für FPGAs (Field Programmable Gate Array) am IPP (Institut für Prozess- und Produktionsleittechnik), sind am Markt erhältliche Evaluationsboards nicht mehr ausreichend. Diese bieten zwar teilweise ein großes Repertoire an direkt zur Verfügung stehender peripherer Beschaltung, sind jedoch in der Herausführung von I/O-Pins für die individuelle Nutzung begrenzt. Die Entwicklung eines Evaluationsboards am IPP geschieht mit der Zielsetzung, die auf der Platine direkt zur Verfügung stehende Peripherie den Ansprüchen anzupassen und mit eigenen Bibliotheken schnell und zuverlässig nutzbar zu machen. Darüber hinaus werden wesentlich mehr I/O-Pins des FPGAs auf universellen Highspeed-Steckverbindern für Ansteckplatinen zur Verfügung gestellt. Somit ist eine Basis geschaffen, auf welcher auch zukünftige Projekte aufbauen können.

# 1 Bisheriger Einsatz von FPGAs am IPP

## 1.1 Phasenstromregelung der VISMA

Ein Einsatzbereich der FPGA-Technologie ist der Phasenstromregler für die VISMA (Virtuelle Synchronmaschine). Die VISMA ist ein am IEE (Institut für Elektrische Energietechnik) entwickeltes leistungselektronisches System, welches sich im Netz wie eine elektromechanische Synchronmaschine verhält und für die Integration regenerativer Erzeuger, insbesondere in dezentralen Netzen, benutzt wird. Um Strom ins Netz einzuspeisen verwendet die VISMA einen Zweipunktregler. Der Regler war in der ersten Version als Analogregler ausgeführt. Nachteile der analogen Stromregelung sind Wärmedrift, nichtlineares Verhalten der elektronischen Bauteile, sowie ein hoher Kalibrierungsaufwand. Ein digitaler Phasenstromregler für diese Anwendung benötigt eine hohe Abtastrate des Stromsignals für alle drei Phasen des Netzes sowie schnelle Signalverarbeitung, um die Stabilität des Regelkreises zu garantieren. Auch schnelle Mikrocontroller, die mit Taktfrequenzen von über 200 MHz arbeiten, reichen hier nicht aus. Der Einsatz des FPGAs ermöglicht es, einen Phasenstromregler zu realisieren, der mit einer Abtastrate von 1 MHz parallel auf drei Phasen arbeitet. Darüber hinaus erleichtert der FPGA-Regler die Kalibrierung und Integration mit dem Mikrocontroller, auf welchem der VISMA-Algorithmus implementiert ist. Der Prototyp des FPGA-Phasenstromreglers wurde mit dem NanoBoard 3000 (von Altium Limited) und einem selbstentwickelten Daughterboard für Strom- und Spannungsmesswertaufnahme ausgeführt. Abbildung 1 zeigt die Architektur der VISMA. Die Algorithmen, die den Soll-Strom vorgeben, werden in einem Tricore Mikroprozessor erzeugt. Der Mikroprozessor kommuniziert mit dem FPGA über eine parallele Schnittstelle. Der Mikroprozessor kann über diese Schnittstelle den Sollstrom für den Phasenstromregler vorgeben, sowie Messwerte der Spannung und des Stromes, die im FPGA gespeichert sind, auslesen. Das FPGA-Board gibt anschließend die Gattersignale für den Umrichter vor. Im FPGA sind Phasenstromregler, Steuerlogik für die Analog-Digital-Wandler sowie Sicherheitsfunktionen implementiert. [1]

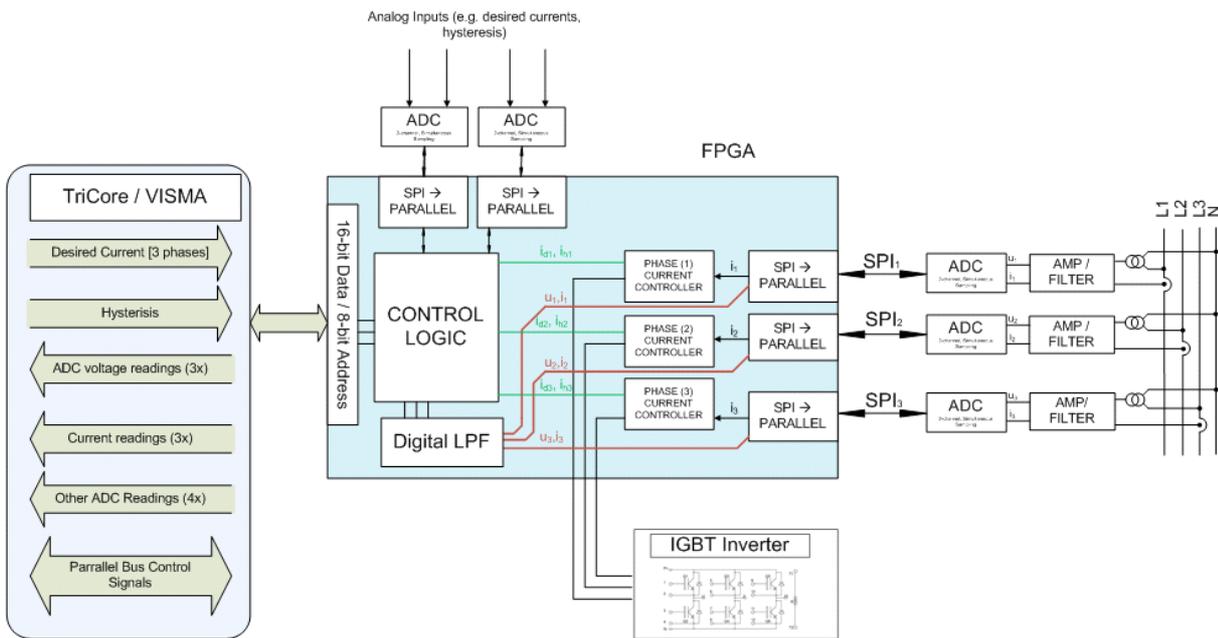


Abbildung 1. Architektur der VISMA mit FPGA Phasenstromregler

## 1.2 Motorsteuerung der Hexapod-Positioniereinheit

Ein weiteres Einsatzgebiet für die FPGA-Technologie ist die Regelung der Motorströme an Hexapoden. Eine in 6 Freiheitsgraden bewegliche Plattform wird bei dieser Kinematik über 6 Beine mit der Basis verbunden. Hybridschrittmotoren an jedem der Beine ermöglichen, deren Längen über durchlaufende Gewindespindeln einzustellen und somit komplexe Positionieraufgaben oder auch Bewegungen mit festem Anstellwinkel gegenüber der Bahnrichtung (z. B. für Schweißaufgaben) vorzunehmen. Zunächst wurden diese Motoren im Halbschrittbetrieb angesteuert, was zu erheblicher Vibration und Geräuschentwicklung führte. Die sinusförmige Regelung der Spulenströme führte zu einem sogenannten Mikroschrittbetrieb. Dieser wird mit 100 Unterschritten je Vollschritt ausgeführt. Hierzu werden vom FPGA Stromsollwertvorgaben für alle 12 Spulen (2 je Motor) über parallele SPI-Datenleitungen (Serial Peripheral Interface) zur Chopper-Stromreglung bereitgestellt. Die Abtastzeit für die Messung der Spulenströme beträgt jeweils 1MHz und wird, ebenfalls vollständig parallel, vom gleichen FPGA vorgenommen. Die zusätzliche Implementierung von 6 Glasmaßstäben zur Erfassung und Regelung der Ist-Beinlängen wird zu weiterem, parallelem Datenaufkommen führen, welches ebenfalls vom FPGA verarbeitet werden kann. Das bisher verwendete Evaluationsboard (NanoBoard 3000) stellt für diese Aufgabe jedoch nicht mehr genügend FPGA-I/O-Pins zur Verfügung. [2]

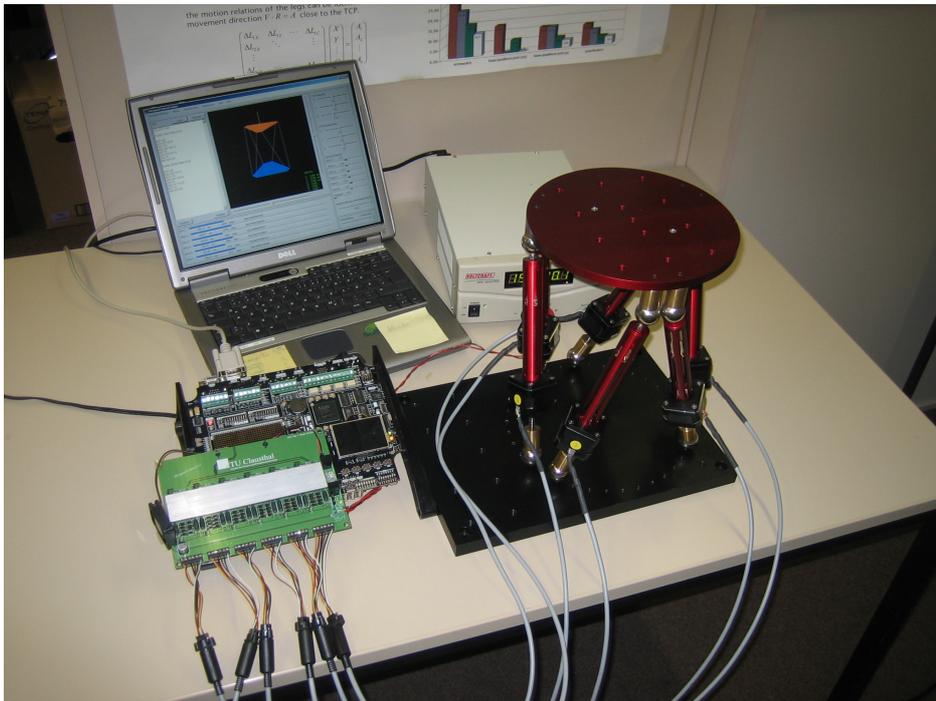


Abbildung 2. Hexapodansteuerung über FPGA-Entwicklungsboard

## 2 Wahl des FPGA

Auf dem FPGA-Board kommt ein „Spartan3 1400AN“ der Firma Xilinx zum Einsatz, welcher mit 1,4 M Gates sowie 502 User I/Os der leistungsstärkste der Spartan3-Familie ist, und zusätzlich bereits über internen Flash zur Konfiguration verfügt. Dieser FPGA kann mit Taktfrequenzen von bis zu 333 Mhz betrieben werden. Die Spartan3AN-Reihe verfügt über 5, in der Ausstattung gestaffelte, jedoch in den grundsätzlichen Eigenschaften gleiche FPGA-Typen. Dieses erlaubt bei der Entwicklung zunächst mit dem „kleinsten“ Vertreter (Spartan3 XC3S50AN) zu beginnen, welcher bereits für ca. 10 € erhältlich ist und aufgrund seiner TQG144-Package-Ausführung noch von Hand gelötet werden kann. An diesem werden zunächst grundsätzliche Beschaltung, sowie das Zusammenspiel mit vielfältiger Peripherie getestet. Anschließend, basierend auf den Testergebnissen, wird das endgültige Board für den Spartan3 XC3S1400AN gefertigt, dessen Bestückung aufgrund der GF676-Package-Ausführung (26 x 26 BGA (Ball-Grid-Array)) erheblich aufwändiger und dessen Beschaffung (mit ca. 80 €) teuer ist.

### 3 Grundlegende Beschaltung des FPGA

#### 3.1 JTAG-Schnittstelle (Joint Test Action Group, IEEE 1149.1)

Über eine JTAG-Schnittstelle werden FPGA sowie Konfigurations-Flash für Programmierung und Debugging zugänglich gemacht. Die JTAG-Kette stellt eine serielle Verbindung aller JTAG-Einheiten dar, wobei jeweils TDO (Test Data Output) von Element n an TDI (Test Data Input) von Element n+1 führt. Zusätzlich werden alle Elemente parallel mit TCK (Test Clock) und TMS (Test Mode Select Input) verbunden um Takt und Testmode vorzugeben:

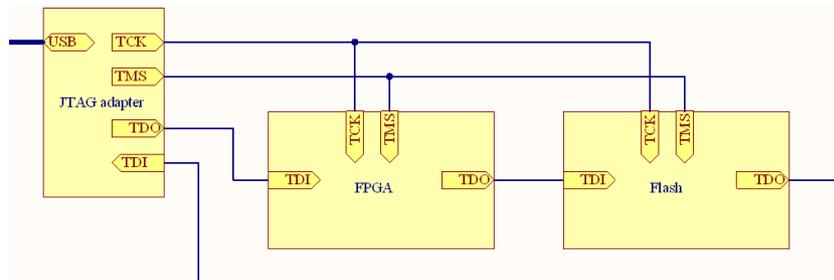


Abbildung 3. Organisation einer JTAG-Chain

Neben der JTAG-Hardchain ist eine zusätzliche JTAG-Softchain realisiert, welche an General Purpose I/O-Pins des FPGAs anliegt. Über diese besteht die Möglichkeit umfangreiche, selbsterstellte Debugfunktionen zu verwenden, welche auf dem FPGA implementiert werden und sich nach dessen Konfiguration aus dem Plattform-Flash bzw. über die JTAG-Hardchain wie eine eigene JTAG-Schnittstelle verhalten.

#### 3.2 FLASH

FPGAs enthalten in der Regel keinen persistenten Speicher. Bei jedem Einschalten ist daher eine neue Konfiguration aus einem persistenten Speicher notwendig (siehe Konfiguration 4). Die Spartan3-AN-Familie zeichnet sich unter anderem dadurch aus, dass ein, zur Speicherung der Konfigurationsdaten ausreichender Flash bereits im FPGA-IC integriert ist. Um mit beiden Varianten Erfahrungen zu sammeln, wird auf dem Board zusätzlich ein externer Plattform-Flash vorgesehen, welcher bei Bedarf bestückt werden kann.

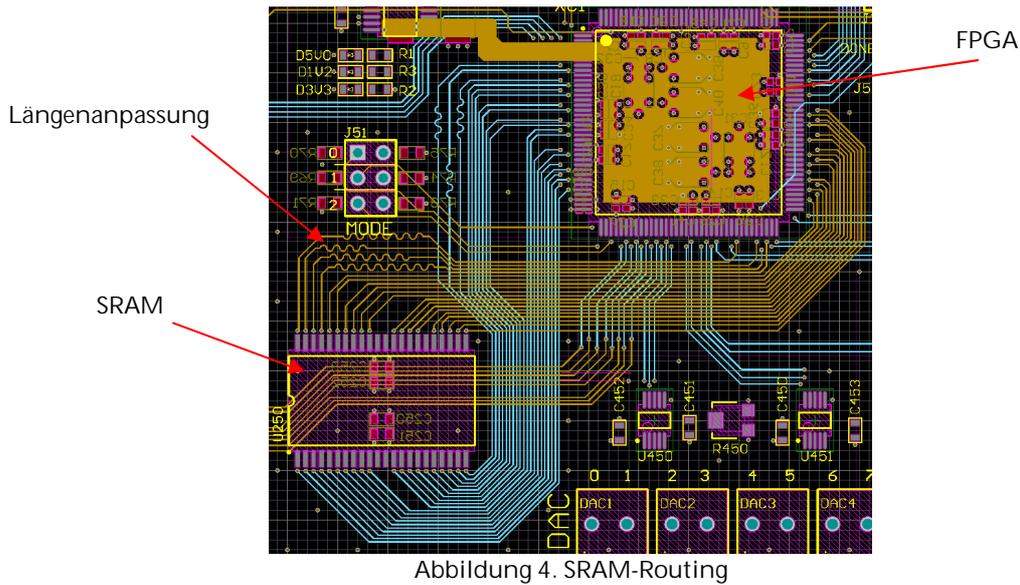
#### 3.3 SRAM

Um vom FPGA schnellen Zugriff auf Speicher zu haben, wird auf dem Board ein SRAM (Static Random Access Memory) platziert. Hohe Lese- und Schreibgeschwindigkeiten verursachen hohe Ansprüche an die Beschaffenheit von Adress- und Datenleiterbahnen, weshalb die spätere Platzierung auf individuellen Ansteckplatinen von Nachteil wäre. Reflexionen an Steckverbindern würden zu Signalstörungen führen. Bereits die Leiterbahnlänge begrenzt aufgrund von Signallaufzeiten die maximale Datenübertragungsfrequenz. Die minimale Zykluszeit  $t_{cyc}$  für das Lesen vom SRAM berechnet sich in diesem Fall aus der Summe folgender Zeiten (worst case):

- $t_{SU} < 2,4 \text{ ns}$  Setup-Zeit des FPGA
- $t_{SIG} = 63 \text{ ps / cm}$  Signallaufzeit der Leiterbahn (Typ FR4)
- $t_{AA} < 12 \text{ ns}$  Adress-Access Zeit des SRAM
- $t_{JIT} < 0,3 \text{ ns}$  Cycle-to-Cycle-Jitter des FPGA-Taktes

Eine Leiterbahnlänge von 20 cm führt somit z. B. zu einer minimalen Zykluszeit von  $t_{cyc} > t_{SU} + 2 \cdot t_{SIG} + t_{AA} + t_{JIT} = 17,22 \text{ ns}$ , also einer maximalen Datenübertragungsfrequenz von  $F_{cyc} < 58,07 \text{ MHz}$ . Bei Reduzierung der Leiterbahnlänge auf ein Viertel ließe sich diese Frequenz bereits um 7,16MHz erhöhen. Aus diesem Grund wird der SRAM nah am FPGA platziert und dem Routing der SRAM-Leiterbahnen hohe Priorität zugeordnet (siehe Abb. 4). Die größte Leiterbahnlänge kann somit auf 62 mm geroutet werden. Alle anderen Leiterbahnen werden bestmöglich an diese Länge angepasst, um den Zeitraum zu

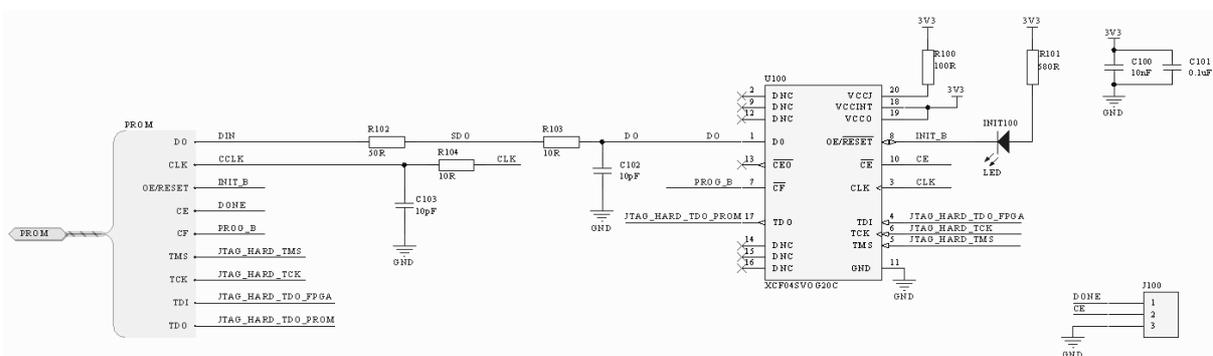
maximieren, in welchem ausschließlich gültige Signale anliegen. Hierzu kommen Mäanderstrukturen, vor allem aber sog. Pin-Swapping zum Einsatz.



Pin-Swapping erlaubt Verbindungen zum FPGA in Gruppen einzuteilen und diese jeweils innerhalb ihrer Gruppen während des Routingprozesses beliebig zu vertauschen. Somit können Überkreuzungen von Signalen reduziert und Leiterbahn­längen optimiert werden. Die endgültige Zuordnung, welche auch für die spätere Programmierung relevant ist, wird erst nach abgeschlossenem Routing anhand von Constraint-Files vorgenommen.

## 4 Konfiguration

Das Design eines FPGAs wird durch LUTs (Look Up Tables) sowie SRAM-Speicherzellen bestimmt, welche die logische Verschaltung von Eingangs- und Ausgangspins realisieren [3]. Diese Informationsspeicherung geschieht nicht persistent, muss also bei jedem Einschaltvorgang erneut vorgenommen werden. Diese sog. Konfiguration des FPGAs geschieht entweder anhand eines erneuten Downloads der Konfigurationsdaten über JTAG, oder aber automatisch aus einem persistenten Flash-PROM (Programmable Read Only Memory) in welchem diese zuvor (ebenfalls über die JTAG-Schnittstelle) hinterlegt wurden. Die Spartan3-AN-Familie verfügt hierzu über verschiedene „Modes“, welche anhand der Kodierung von 3 Mode-Pins gewählt werden können. Diese Pins werden auf dem Evaluationsboard auf Jumper herausgeführt, um auf Ground bzw. die entsprechende Versorgungsspannung „gezogen“ werden zu können. Je nach Mode kann die Konfiguration beim Bootvorgang z. B. selbstständig aus dem internen Flash-PROM oder einem angeschlossenen Plattform-Flash-PROM vorgenommen werden. Auch Mikrocontroller können zur Konfiguration eingesetzt werden. Neben dem internen Speicher ist auf dem Evaluationsboard zusätzlich ein externer Speicher zur Konfiguration vorgesehen. Dieser kann über JTAG beschrieben werden und ist mit den entsprechenden Konfigurationsleitungen des FPGAs verbunden, um dessen selbständige Konfiguration nach dem Einschalten aus diesem Speicher zu ermöglichen:



## 5 Ergebnis

Mit der Entwicklung des FPGA-Basisboards wurde eine Grundlage für aktuelle und zukünftige Projekte des IPP geschaffen, welche schnelle Regelungsaufgaben mit hoher Parallelität zu bearbeiten haben. Die Bereitstellung einer großen Anzahl von I/O-Pins auf Highspeed-Steckern erlaubt die volle Ausnutzung des Potentials der Parallelverarbeitung. Die Konfiguration des FPGAs kann über eine JTAG-Schnittstelle direkt vorgenommen werden, oder aus internem bzw. externem Plattformflash geschehen, welcher zuvor ebenfalls über JTAG programmiert werden kann. Eine Auswahl an Peripheriebausteinen auf dem Board ermöglicht den Einstieg in die FPGA-Konfiguration, bzw. das basale Debuggen anhand von Dipschaltern und LEDs. AD- und DA-Converter für einfache Anwendungen sind vorgesehen und können bei Bedarf bestückt werden. Darüber hinaus ist, aufgrund der hohen Anforderungen an dessen Leitungseigenschaften, ein SRAM-Chip dicht neben dem FPGA platziert. USB- und RS232-Schnittstellen ermöglichen schließlich die Kommunikation des Boards mit PC's und anderen Geräten.

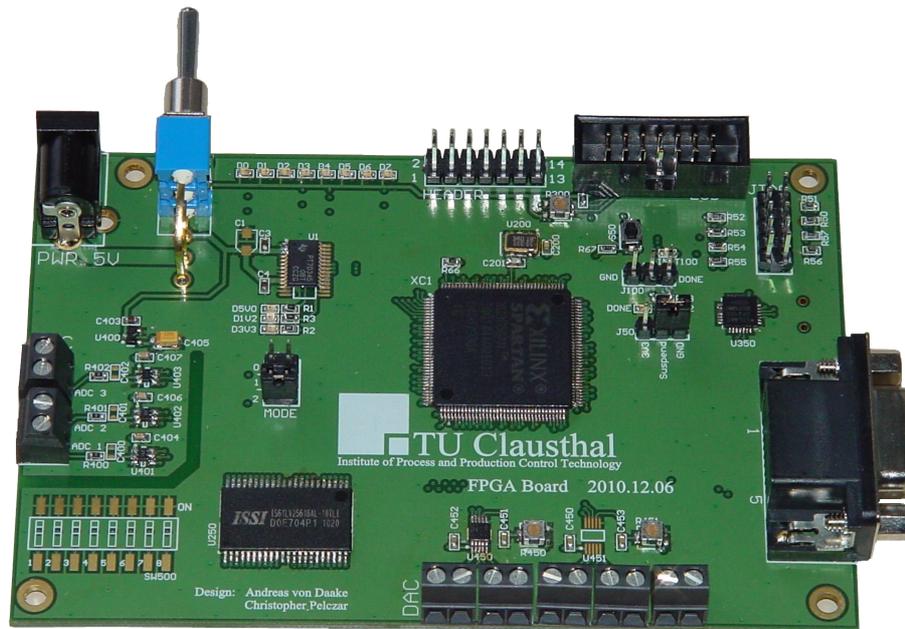


Abbildung 6: FPGA-Board (Spartan3 XC3S50AN)

## References

- [1] C. Pelczar, M. Stubbe, D. Turschner, O. Zirn, Mobile Virtual Synchronous Machine for V2G Applications, 17<sup>th</sup> ITS World Congress, Busan, 2010
- [2] A. v. Daake, C. Vetter, E. Böhm, O. Zirn, Error Classification and Visualisation for Hexapod Positioning Units, Proceedings of the 10<sup>th</sup> euspen International Conference, Delft, 2010
- [3] G. Kemnitz, Entwurf digitaler Schaltungen, Springer Verlag, 2011