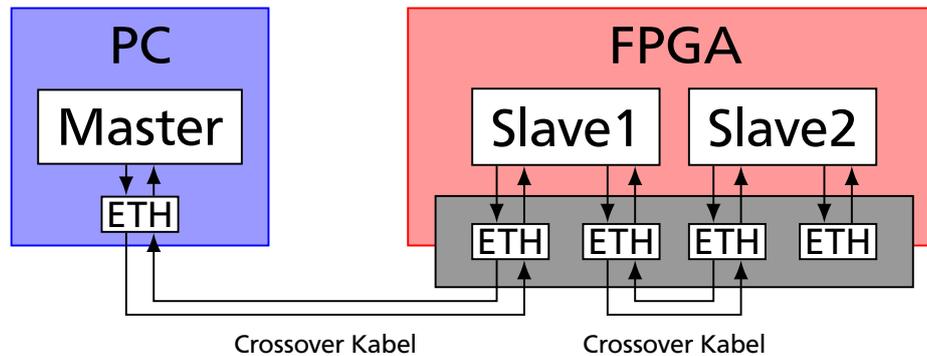


# Bachelorarbeit:

## Hardware-Beschleunigung von Teilen des EtherCAT Slave Stacks



Das auf Ethernet basierende *EtherCAT* Protokoll zeichnet sich durch eine hohe zeitliche Präzision aus und wird in Automatisierungsanwendungen zur Kommunikation zwischen verteilten Reglern eingesetzt. Dabei werden idR. lineare Topologien aus einem Master und mehreren Slaves gebildet. Dabei benötigen die Slaves jeweils zwei Ethernet Ports (ETH), um Pakete mit möglichst niedriger Latenz durch das gesamte Netzwerk zu leiten. Jeder Slave entnimmt dem Paket die für ihn bestimmten Daten bzw. fügt seine eigenen Daten hinzu, während das Paket von Slave zu Slave weitergereicht wird. Der letzte Slave schickt das modifizierte Paket entweder direkt (über einen zweiten ETH Port am Master) oder indirekt (über den ETH Rückkanal) zurück an den Master. Der Protokoll-Stack der EtherCAT Slaves muss für minimale Latenz und Jitter in Hardware implementiert werden.

Als Vorbereitung für spätere Forschungsarbeiten an Hardware-beschleunigten EtherCAT Systemen soll zunächst die grundlegende Funktionalität für einer linearen EtherCAT Topologie in Verilog implementiert werden. Dazu wird ein Xilinx Zynq basiertes Zedboard per FMC-Adapter um vier Ethernet Ports erweitert, wodurch zwei Slaves nachgebildet werden können. Eine PC-Software (SOEM: <http://openethercatsociety.github.io/>) dient als einfacher EtherCAT Master zur Detektion aller angeschlossenen Slaves, sowie zum Senden und Empfangen von Prozessdaten.

Folgende konkrete Aufgaben sind im Rahmen einer Bachelorarbeit zu realisieren:

- Literaturübersicht zu kommerziellen und wissenschaftlichen HDL-Implementierungen von EtherCAT Slaves.
- Inbetriebnahme der Ethernetports (<https://opsero.com/product/ethernet-fmc>) am Zedboard. Dafür können Referenzdesigns adaptiert werden, die unter <https://github.com/fpgadeveloper/ethernet-fmc-axi-eth> verfügbar sind.
- Verilog-Implementierung folgender Bestandteile des EtherCAT Device Protokolls:
  - Topologie-Scan und Konfiguration
  - Fieldbus-Memory-Management-Unit (FMMU)
  - Zyklischer Datenaustausch (Lesen, Schreiben, Weiterleiten)
  - Automatisches Öffnen und Schließen der Übertragungsleitung je nach Status des Trägersignals

- 
- Robustheit gegenüber alternativen Ethernet Paketen
  - Optional: Zeitsynchronisation mittels Distributed Clocks

Dabei sind folgende Randbedingungen zu beachten:

- Die Slaves-Stacks sollen keine Informationen direkt über die FPGA-Fabric oder gemeinsame Speicher austauschen.
  - Das EtherCAT Automation Protokoll sowie aufgesetzte Mailbox-Protokolle müssen nicht implementiert werden.
  - Die EtherCAT-Telegramme werden direkt im Ethernet-Frames, und nicht in UDP- bzw. TCP-Telegrammen versendet.
  - Die Anwendungsschicht über dem Slave-Protokoll kann möglichst einfach gehalten werden. Die Schnittstelle zwischen Anwendung und Protokoll ist aber angemessen zu strukturieren und zu kommentieren.
  - Der Implementierungsumfang kann auf die zum Betrieb einer linearen Topologie notwendigen Bestandteile beschränkt werden.
  - Die einfache Erweiterbarkeit um zusätzliche EtherCAT Funktionalität ist von zentraler Bedeutung.
- Funktionale Verifikation der Slave Implementierungen mittels SOEM für zwei Slaves in einer linearen Topologie. Die Reihenfolge der Slaves in der linearen Topologie soll dabei prinzipiell ohne Rekonfiguration des FPGAs möglich sein.
  - Evaluation des Ressourcen-Bedarfs sowie der IO-Latenz eines Slaves.

Der Arbeitsfortschritt ist durch regelmäßiges Einpflegen der Quelldateien in ein Versionsverwaltungssystem (Git) nachzuweisen. Die Quelldateien müssen angemessen dokumentiert und strukturiert werden, um spätere Weiterentwicklungen zu vereinfachen.

Benötigte Kenntnisse:

- Verilog
- Grundlagen von Netzwerkprotokollen

Empfohlene Literatur:

- [https://www.ethercat.org/pdf/ethercat\\_d.pdf](https://www.ethercat.org/pdf/ethercat_d.pdf)
- [https://download.beckhoff.com/download/document/io/ethercat-development-products/ethercat\\_esc\\_datasheet\\_sec1\\_technology\\_2i2.pdf](https://download.beckhoff.com/download/document/io/ethercat-development-products/ethercat_esc_datasheet_sec1_technology_2i2.pdf)
- [https://download.beckhoff.com/download/Document/io/ethercat-development-products/ethercat\\_esc\\_datasheet\\_sec2\\_registers\\_2i7.pdf](https://download.beckhoff.com/download/Document/io/ethercat-development-products/ethercat_esc_datasheet_sec2_registers_2i7.pdf)

Bei Interesse melden Sie sich bei: Andreas Engel (engel@esa.informatik.tu-darmstadt.de), S2|02 E106

---