

Die Österreichische Tagung
zum Themenbereich des Entwurfs
integrierter Schaltungen und Systeme

11. Oktober 2008
Messezentrum Wien

Austrochip

OVE

 **IEEE**
AUSTRIA SECTION



Herausgegeben von
Peter Balog und Martin Horauer

Fachhochschule Technikum Wien
Fachbereich für Embedded Systems

Eine Serial ATA Interface Einheit für FPGAs

Herbert Nachtnebel
Institut für Computertechnik
Technische Universität Wien
Gußhausstraße 27-29/E384, 1040 Wien
herbert.nachtnebel@tuwien.ac.at

Klaus Gravogl
Oregano Systems
Design & Consulting GmbH
Phorusgasse 8, 1040 Wien
gravogl@oregano.at

Kurzfassung

Moderne, vom Anwender programmierbare, logische Bausteine (FPGAs, Field Programmable Gate Arrays) bieten heutzutage eine Komplexität die es erlaubt, komplexe Systeme mitsamt Prozessor und allen notwendigen Ein- und Ausgabeeinheiten in einem einzigem Baustein zu implementieren. Da moderne FPGAs meistens spezielle Hardwareblöcke zur seriellen Kommunikation integriert haben, stellt sich zunehmend die Frage, ob es nicht möglich wäre, FPGAs direkt mit modernen Festplatten mit dem ebenfalls seriellen Serial ATA Interface kommunizieren zu lassen, ohne dazwischen geschaltete, physikalische, integrierte Schaltkreise zu verwenden. In einer heuer an der TU-Wien abgeschlossenen Diplomarbeit wird diese Frage umfassend beantwortet.

1 Einleitung

Der Serial ATA (kurz SATA) Interface Standard [1] für Festplatten hat die Verbindungstechnik in der Computerindustrie revolutioniert, indem die parallele Busstruktur der fünf verschiedenen ATA/ATAPI Standards [2] mit deren 40/80 Busleitungen durch eine serielle Hochgeschwindigkeitsschnittstelle mit jeweils zwei Adempaaren pro Übertragungsrichtung ersetzt wurde. Damit konnte der Verkabelungsaufwand in modernen Computeranlagen entscheidend verringert werden. Interessant ist, dass dies nicht zu Lasten der Übertragungskapazität des Kanals erreicht wurde, da selbst der SATA 1.0 Standard bereits bis zu 150MByte pro Sekunde übertragen kann.

Die Serial ATA Verbindungstechnik folgt dem starken Trend in der Elektrotechnik, immer mehr parallele Bussysteme durch einfache, bitserielle Punkt-zu-Punkt Verbindungen zu ersetzen, da diese Kanäle leichter mit guten Übertragungseigenschaften bezüglich Frequenz- und Impedanzverlauf realisiert werden können. Man denke nur an die Infiniband, RocketIO oder PCI-Express Verbindungstechniken, allesamt serielle Übertragungsverfahren mit einer Datenrate bis zu 10 GBit pro Sekunde. Diese hohe Datenrate ist, diskret aufgebaut, außerhalb eines integrierten Schaltkreises, kaum mehr zu erreichen. Es ist daher nicht verwunderlich, dass viele Hersteller dazu übergegangen sind, spezielle Schaltungsblöcke, die Multi-Gigabit-Transceiver (MGTs), in ihre FPGAs zu implementieren, die die oben genannten Interface-Standards unterstützen. Im Wesentlichen ent-

halten diese Transceiver drei Schaltungsteile: PLLs zur Taktrückgewinnung, Leitungskodierer die den Bitstrom formen um damit die Übertragungseigenschaften zu verbessern und den Schaltungsaufwand für die Empfangsseite zu verringern, sowie Seriell-zu-Parallel Wandler, welche die hohen, seriellen Taktraten auf erträglichere Datenraten für die parallelen Datenwörter reduzieren, die dann intern weiterverarbeitet werden.

Prüft man die Datenblätter der aktuellen FPGAs verschiedener Hersteller mit eingebauten Multi-Gigabit-Transceivern so sieht man leider, dass der Serial ATA Standard defakto nie unterstützt wird. Dies ist insofern verwunderlich, da im Prinzip die MGTs durchaus kompatibel sein sollten. Die Signalisierung auf der Leitung ist differentiell nach dem LVDS (Low Voltage Differential Signaling) Standard, die Datenraten sind im unterstützten Bereich (SATA-1.0: 1.5 Gbit/s, SATA-2.0: 3.0 Gbit/s, Infiniband & RocketIO bis 10 Gbit/s) und der verwendete Leitungscode ist derselbe (8b/10b Kodierung, Originalveröffentlichung in [3]). Es drängt sich daher die Frage auf, welcher Teil der SATA Spezifikation von den Bausteinen nicht eingehalten werden kann und ob es nicht möglich ist, mit kleinen Anpassungen, möglicherweise mit einer externen Beschaltung, trotzdem die in den FPGAs eingebauten Transceiver zu verwenden. Dieser Frage ist einer der Autoren dieses Artikels, Klaus Gravogl, jüngst in seiner Diplomarbeit [4] nachgegangen. Seine Ergebnisse werden in diesem Artikel im folgenden kurz zusammengefasst.

Leser die mit der Materie vertraut sind werden einwenden, dass es sowieso gute, von der Industrie in umfangreichem Maße verwendete Bausteine gibt (im weiteren PHY genannt), mit denen man Serial ATA Festplatten ansprechen kann. Es ist daher ein bisschen widersinnig, mit einer externen Beschaltung FPGAs zwangsweise kompatibel zu machen, wenn es Standardbauteile für diesen Zweck gibt. Dem muss entgegen gehalten werden, dass die Verwendung dieser Bausteine auch Nachteile haben kann. Zum Ersten besitzen diese Bausteine erst wieder ein paralleles Interface und die Verbindungen zwischen dem FPGA und dem PHY belegen nicht nur kostbare Ein-/Ausgabe-Ressourcen, sondern sind vom elektronischem Standpunkt aus schwieriger zu handhaben, als die seriellen LVDS Leitungen. Und zweitens haben diese Bausteine den „Nachteil“, dass sie funktionieren. Möchte man gezielt Fehler in die Übertragung einbauen um beispielsweise die Konformität des angeschlossenen Geräts mit dem Standard zu überprüfen, so ist dies mit diesen Bauteilen einfach nicht möglich. Eine

direkte Lösung im FPGA ist daher durchaus sinnvoll, vor allem wenn ein ansonsten sowieso häufig brach liegender Multi-Gigabit-Transceiver einer guten Verwendungsmöglichkeit zugeführt werden kann.

2 Serial ATA Standard

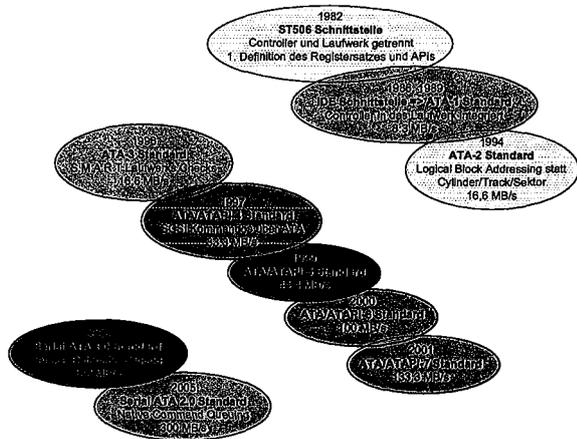


Abbildung 1: Geschichtliche Entwicklung der Festplatteninterfaces für Computer Systeme

Der Serial ATA Standard ist die neunte Inkarnation eines kostengünstigen, normierten Interfaces für die Kommunikation mit Festplatten in Computer Systemen, siehe Abbildung 1. Da in der Computertechnik die Kompatibilität der verschiedenen Generationen außerordentlich wichtig ist, finden sich selbst in der Serial ATA Norm einige Seltsamkeiten wieder, die bis auf die Anfänge der Festplatteninterfaces in den achtziger Jahren des vorigen Jahrhunderts zurück zu führen sind. Beispielsweise wird im Protokoll ein Registersatz definiert, der so schon in der IDE Schnittstelle – nachträglich als ATA-1 Standard bezeichnet – verwendet wurde. Da damals das Bussystem der Personal Computer 16 Bit breit war, bezieht sich der Registersatz ebenfalls auf 16 Bit breite Datenwörter. Die Übertragung der Daten auf den seriellen Datenleitungen wird bei Serial ATA aber immer mit 32 Bit Breite durchgeführt.

Der Serial ATA Standard definiert ein Referenzmodell zur Kommunikation zwischen den Kommunikationspartnern Host und Device. Der Host ist dabei jenes Gerät, das die Datübertragung steuert und Kommandos an das Device abgibt. Der Host entspricht daher dem Controller im Personal Computer und das Device der Festplatte des Computers.

Das Referenzmodell definiert vier Schichten: Bitübertragungsschicht (Physical Layer), Verbindungsschicht (Link Layer), Transportschicht (Transport Layer) und Applikationsschicht (Application Layer). In der Bitübertragungsschicht werden sowohl die elektrischen als auch die mechanischen Eigenschaften der Schnittstelle beschrieben. Die Verbindungsschicht und die Transportschicht nehmen den eigentlichen Kern des Interfaces auf. Die Host Link Statusmaschine der Verbindungsschicht kommuniziert über die elektrische, bitserielle Schnittstel-

le mit der korrespondierenden Statusmaschine im Device. Dabei werden immer komplette Nachrichten (Frames) zwischen den beiden Partnern ausgetauscht. Die Statusmaschinen der Transportschicht der Kommunikationspartner kommunizieren wiederum mit Hilfe dieser Nachrichten miteinander, um die Registerschnittstelle für die Applikationsschicht zu realisieren. Wird eines der Register im Host geändert, so wird eine Nachricht formuliert, in der diese Änderung vermerkt ist, und an das Device übertragen. Dieses reagiert entsprechend darauf und schickt anschließend eine Nachricht mit dem geänderten Inhalt zurück.

2.1 Physikalisches Interface

Als Transportmedium werden bei Serial ATA zwei Adernpaare verwendet, jeweils eines pro Übertragungsrichtung. Die Datenübertragung erfolgt differentiell mit einer Differenzspannung von 325 bis 600 mV bei einer nominellen Taktrate von 1500 MHz. Es werden geschirmte 50 Ohm Leitungen verwendet, die differentiell mit 100 Ohm zwischen den Leitungen abgeschlossen werden. Bis auf die erhöhte Spannung entsprechen diese Werte einer standardgemäßen LVDS Anwendung. Da die Anforderung an die Differenzspannung leicht erhöht ist, muss man bei direktem Anschluss an ein FPGA mit verringerten Leitungslängen rechnen.

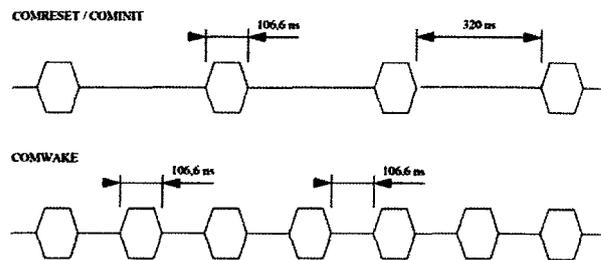


Abbildung 2: Serial-ATA Out-of-Band Signale

Als Besonderheit des physikalischen Interfaces stellen sich die so genannten Out-of-Band Signale heraus, siehe Abbildung 2. Diese sind zum Herstellen der Kommunikation nach dem Einschalten oder einem Reset der Verbindung zwingend vorgeschrieben und stellen das eigentliche Problem dar, weswegen die meisten FPGAs nicht mit Serial ATA kompatibel sind. Es gibt insgesamt drei verschiedene Out-Of-Band Signale, wobei zwei davon identisch sind, weil sie entweder nur vom Host oder vom Device ausgesendet werden. Bei jedem Signal werden 160 Bits eines bestimmten Bitmusters (die Align-Primitiven, siehe nächsten Abschnitt) übertragen, danach muss auf der Leitung die Differenzspannung für eine gewisse Zeit unter 25 mV abgesenkt werden. Jedes dieser Signale wird ein paar Mal wiederholt, um eine gute Erkennungsrate bei moderatem Schaltungsaufwand zu erreichen.

Das Abschalten der Differenztreiber ist bei den meisten FPGAs leider nicht möglich, da die im Empfänger eingebauten PLLs nicht vernünftig mit dem Verlust des Signals und damit dem Verlust der Synchronisierung umgehen können. Um dieses geforderte Verhalten zu

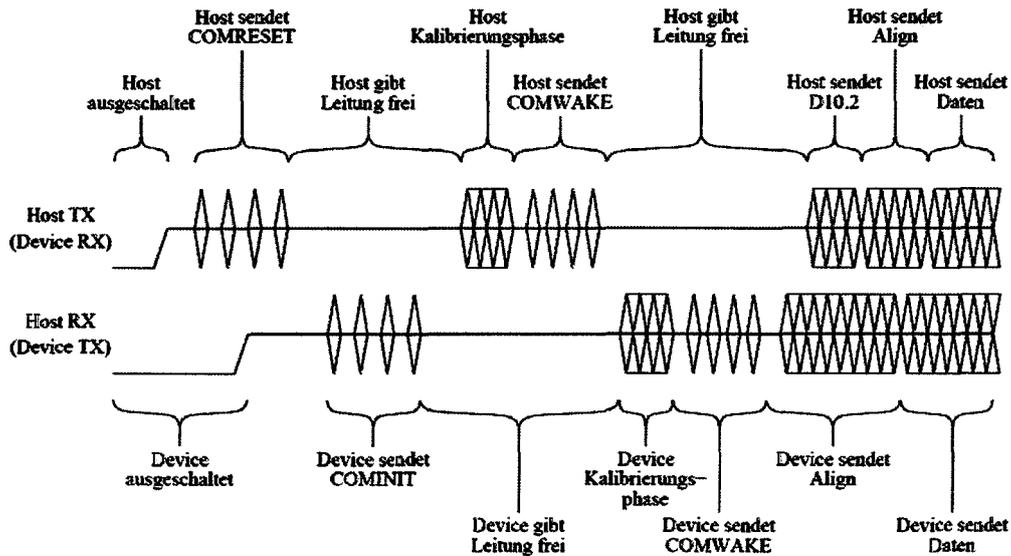


Abbildung 3: Initialisierungssequenz der Host↔Device Verbindung nach dem Einschalten

realisieren, muss man daher zu Tricks greifen. Dies ist der eigentliche Grund, wieso die meisten FPGAs nicht direkt mit Serial ATA Devices kompatibel sind.

2.2 Verbindungsschicht

Der Link Layer hat die Aufgabe, die Nachrichten, die an die gegenüberliegende Seite gesendet werden, richtig aufzubauen. Er versorgt die Frames mit den Start- und Ende-Kennzeichen, sichert die Datenübertragung mit Hilfe eines CRCs ab, löst Konflikte bei gleichzeitigem Senden und Empfangen von Nachrichten, steuert den Datenfluss und führt als wahrscheinlich wichtigste Aufgabe die Kodierung nach dem 8b/10b Verfahren durch.

Das 8b/10b Kodierungsverfahren verpackt jeweils 8 Bit Daten in ein 10 Bit Symbol in der Art, dass die Übertragungsleitung mittelwertfrei wird und es garantiert ist, dass nach maximal 5 Takten in jedem Fall eine Flanke im Signal zu finden ist. Diese Eigenschaft hilft den PLLs der Empfängerseite, den Sendetakt zurück zu gewinnen. Die versendeten Codewörter enthalten auch Leerwörter, in denen keine Nutzinformationen kodiert werden, sondern mit denen der Datenstrom gesteuert wird. Das wichtigste Kontrollwort ist dabei das so genannte Align-Primitiv, das spätestens alle 554 Datenwörter (ein Wort bezeichnet bei SATA immer 32 Bit) doppelt versendet werden muss. Dem Empfänger steht es frei, eines oder auch beide Align-Primitive aus dem Datenstrom zu entfernen, damit ein Ratenausgleich zwischen Sende- und Empfangstakt durchgeführt werden kann.

Das 8b/10b Kodierungsverfahren hat sich bei allen Gigabit Übertragungsverfahren durchgesetzt. Alle Multi-Gigabit-Transceiver besitzen daher eine Hardwareunterstützung für diesen Zweck. Darum ist es sinnvoll, die vorhandene Hardware der MGTs möglichst zu verwenden. Daher sollte der Kodierungsschritt bei der Implementation in einem FPGA nicht wie im Standard gefordert in der Verbindungsschicht realisiert werden, son-

dern in den physikalischen Layer verschoben werden. Glücklicherweise lässt sich dies ohne große Änderung an der Implementierung leicht durchführen.

2.3 Transport- und Applikationsschicht

In der Transportschicht werden die Nachrichten zur Kommunikation zusammengestellt. Leser die mit der Programmierung der Plattenschnittstelle von Personal Computern vertraut sind, können bei der Durchsicht der Serial ATA Norm leicht erkennen, dass die Nachrichten im Grunde den Registersatz der alten ST506 Schnittstelle direkt abbilden. Dieser Registersatz ist doppelt vorhanden, einmal im Device selbst und ein weiteres mal – als Schattenregister ausgeführt – im Host. Um Kommandos an die Platte zu senden werden diese Register programmiert, von der Transportschicht in eine Registernachricht umgesetzt und an das Device versendet. Nach erfolgreichem Abschluss der Operation versendet das Device den Status, ebenfalls in eine Registernachricht verpackt, an den Host zurück. Die Nutzdaten der verschiedenen Plattenoperationen werden dabei in so genannte Datennachrichten verpackt.

Die Applikationsschicht ist im Gegensatz zu den anderen Schichten in der Serial ATA Spezifikation nicht stark normiert und kann in großem Umfang an die Anforderungen der Applikation angepasst werden. Zwingend vorgeschrieben sind nur drei Register, die Status-, Fehler- und Kontrollregister, die für jeden Serial ATA Anschluss getrennt zur Verfügung gestellt werden müssen. Über das restliche Programmiermodell werden keine Aussagen getroffen, es gibt dafür aber Herstellernormen. Die bekannteste ist wohl die Advanced Host Controller Interface (AHCI) Spezifikation von Intel [5], der auch viele andere Hersteller von Chipsätzen für die Computerindustrie folgen. In diesem Standard ist die Registerschnittstelle eines Host Controllers ausführlich beschrieben.

3 FPGA Interface

Die Aufgabe, die in der Diplomarbeit von Klaus Gravogl zu lösen war, ist es mit Hilfe eines FPGAs Verbindung mit einer Serial ATA Festplatte aufzunehmen, ohne spezielle, für Serial ATA geeignete, physikalische Interfacechips zu Hilfe zu nehmen. Es sollte bewiesen werden, dass dies, zur Not mit einem minimalen externen Schaltungsaufwand, möglich ist. Dazu musste eine stabile Datenverbindung initialisiert werden.

Abbildung 3 verdeutlicht graphisch die Aufgabe, die realisiert werden musste. Nach dem Einschalten des Systems muss der Host das COMRESET Signal richtig erzeugen, danach auf der Verbindung die COMINIT Antwort des Devices abwarten und diese richtig erkennen. Danach wird mit Hilfe der COMWAKE Signale die Verbindung vom Host initiiert und nach einer weiteren Handshake-Phase als auch einer Kalibrierung der Verbindung, steht die Verbindung zwischen den Geräten zur Verfügung.

Die besondere Herausforderung war es nun, die COMRESET/COMINIT und COMWAKE Signale mit minimalem Aufwand ordnungsgemäß zu erzeugen, und darüber hinaus, im FPGA, richtig zu detektieren. Für den Testaufbau wurde ein Virtex-II Pro Development Board von Memec Design (DS-BD-2VP4/7-FG456 Rev. 5) verwendet. Im FPGA selbst wurde ein abgespeckter Serial ATA IP Core implementiert, bei dem nur die untersten zwei Schichten des Referenzmodells realisiert sind. Für den Beweis der Funktionsfähigkeit des Interfaces ist das ausreichend. In Abbildung 4 ist das Blockschaltbild der implementierten Lösung zu sehen.

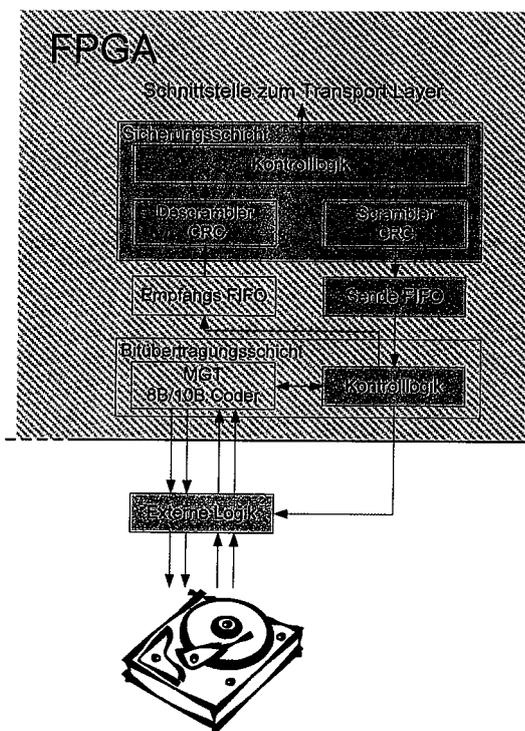


Abbildung 4: Blockschaltbild des realisierten Serial ATA IP Cores

3.1 Erzeugung der Out-of-Band Signale

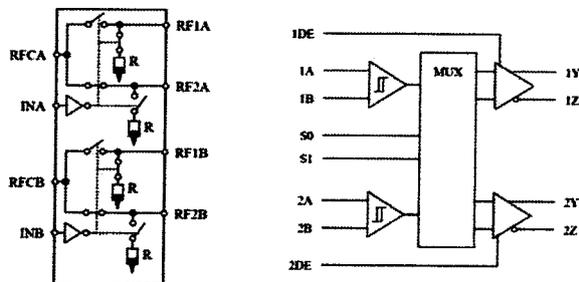
Die Multi-Gigabit-Transceiver der aktuellen FPGA Familien¹, unterstützen leider von sich aus nicht die Erzeugung der Out-of-Band Signale. Für die Erzeugung der Signale auf einem anderen Wege wurden daher insgesamt fünf verschiedene Lösungsmöglichkeiten untersucht. Zwei davon wären ohne eine externe Beschaltung durchführbar, die restlichen drei benötigen aber eine Logik in der Übertragungsleitung vom FPGA zur Festplatte:

- 1) Powerdown Modus des MGTs einschalten;
- 2) partielle Rekonfiguration des FPGAs im Betrieb;
- 3) Kurzschluss der Übertragungsleitung mit Hilfe eines FETs;
- 4) Realisierung mit Hilfe analoger LVDS Schalter, siehe Abbildung 5 links;
- 5) Realisierung mit einem digitalem LVDS Schalter, siehe Abbildung 5 rechts.

Jede einzelne Lösung wurde am Development Board implementiert und bezüglich Konformität mit dem Standard überprüft. Dabei zeigte es sich, dass die ersten beiden Lösungen leider nicht in der Lage waren, das geforderte Timing der Signale korrekt einzuhalten. Insbesondere die Anforderung innerhalb von 10 ns die Differenzspannung auf der Leitung unter 25 mV abzusenken, war damit nicht zu realisieren. Die dritte Lösung ist zwar prinzipiell funktionstüchtig hat aber den Nachteil, dass die Steuerspannung des Feldeffekttransistors größer als die Versorgungsspannung moderner FPGAs sein muss, um die Umschaltung rasch genug durchführen zu können. Sie ist daher mit einem nicht unerheblichen, zusätzlichen Schaltungsaufwand verbunden. Die letzten beiden Lösungen wurden letztendlich auf einer externen Leiterplatte implementiert (siehe Anhang) und konnten beide als funktionsfähig bewiesen werden.

3.2 Erkennung der Out-of-Band Signale

Messungen am Multi-Gigabit-Transceiver der Virtex 2 Pro Serie zeigten, dass die abgeschalteten Transceiver

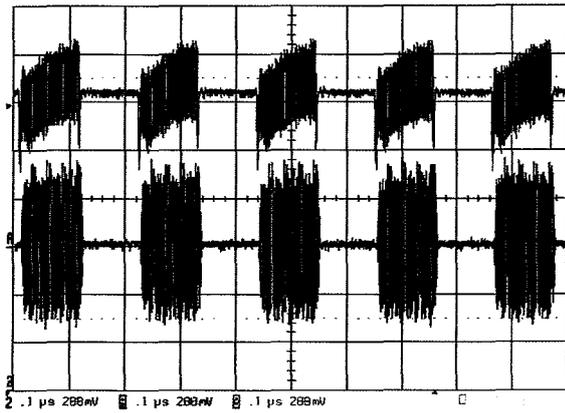


Analog Devices: ADG936

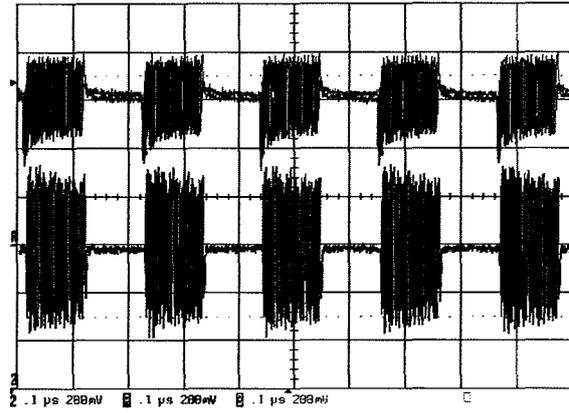
Texas Instruments: SN65LVDS122

Abbildung 5: analoge (links) und digitale (rechts) LVDS Schalter

¹ Untersucht wurden FPGAs von Xilinx (Virtex 2, Virtex 2 Pro, Virtex 4) sowie von Altera (Stratix, Stratix GX, Stratix 2, Stratix 2 GX)



analoger LVDS Schalter



digitaler LVDS Schalter

Abbildung 6: Messung der Aussendung von Out-of-Band Signaleburst mit Hilfe der verwendeten LVDS Schalter.

des Kommunikationspartners sofort detektiert wurden, indem das Not-In-Table Signal des MGTs im FPGA gesetzt wird. Es konnte weiters verifiziert werden, dass innerhalb des Timings der Out-of-Band Signale, die Empfangs-PLL auf den Sendetakt einschwingt und die Signale daher recht eindeutig erkennbar sind. Die Erkennung der Out-of-Band Signale kann daher zuverlässig über ein rein digitales Filter erfolgen, das den zeitlichen Verlauf der Signale auswertet, und muss daher nicht zwingend in Hardware realisiert werden.

4 Ergebnisse

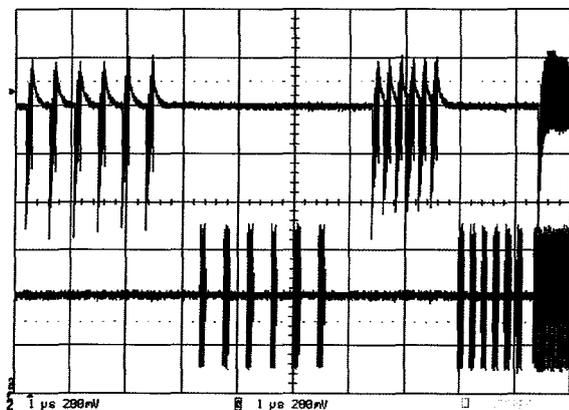
Für die Verifikation der verschiedenen Lösungsmöglichkeiten wurden einige Messungen am fertigen Versuchsaufbau durchgeführt. Abbildung 6 zeigt die Generierung der Out-of-Band Signale mit Hilfe der beiden LVDS Schalter Varianten. Links ist die analoge Version zu sehen, rechts die digitale. Die obere Hälfte der Oszilloskopbilder zeigt die beiden differentiellen Leitungen getrennt übereinander gelegt, die untere Hälfte zeigt die errechnete Differenzspannung. Durch notwendige kapazitive Kopplungen kommt es bei der analogen Version zur sichtbaren Dachschräge, die aber keine Probleme bei der Kommunikation verursacht, da sie beide Leitungen

des differentiellen Adernpaares gleich betrifft und durch die Differenzbildung zum Verschwinden gebracht wird.

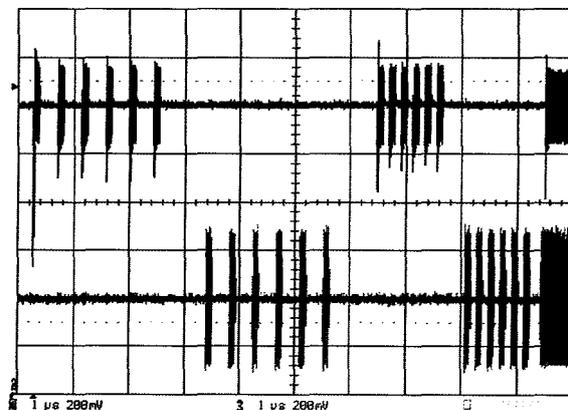
Abbildung 7 zeigt den Verbindungsaufbau der im FPGA realisierten Serial ATA Interface Einheit mit einer handelsüblichen angeschlossenen Festplatte. Bei der analogen Variante ist die kapazitive Kopplung wieder eindeutig erkennbar. Im Prinzip widerspricht die sichtbare Endladekennlinie der Serial ATA Norm, im Versuchsaufbau konnte aber kein Problem bei der Kommunikation mit der Festplatte festgestellt werden. Trotzdem sollte für eine möglichst gute Kompatibilität mit dem Serial ATA Standard, die digitale Version der externen Beschaltung bevorzugt werden.

5 Zusammenfassung

Im Rahmen der Diplomarbeit von Klaus Gravogl konnte gezeigt werden, dass es durchaus möglich ist, die in FPGAs integrierten Multi-Gigabit-Transceiver Blöcke mit einer kleinen externen Beschaltung so zu erweitern, dass mit ihnen eine Kommunikation mit Geräten mit Serial ATA Interface realisiert werden kann. Die im FPGA implementierte Interface Einheit ist weitgehend unabhängig vom Hersteller und kann als Basis für eine

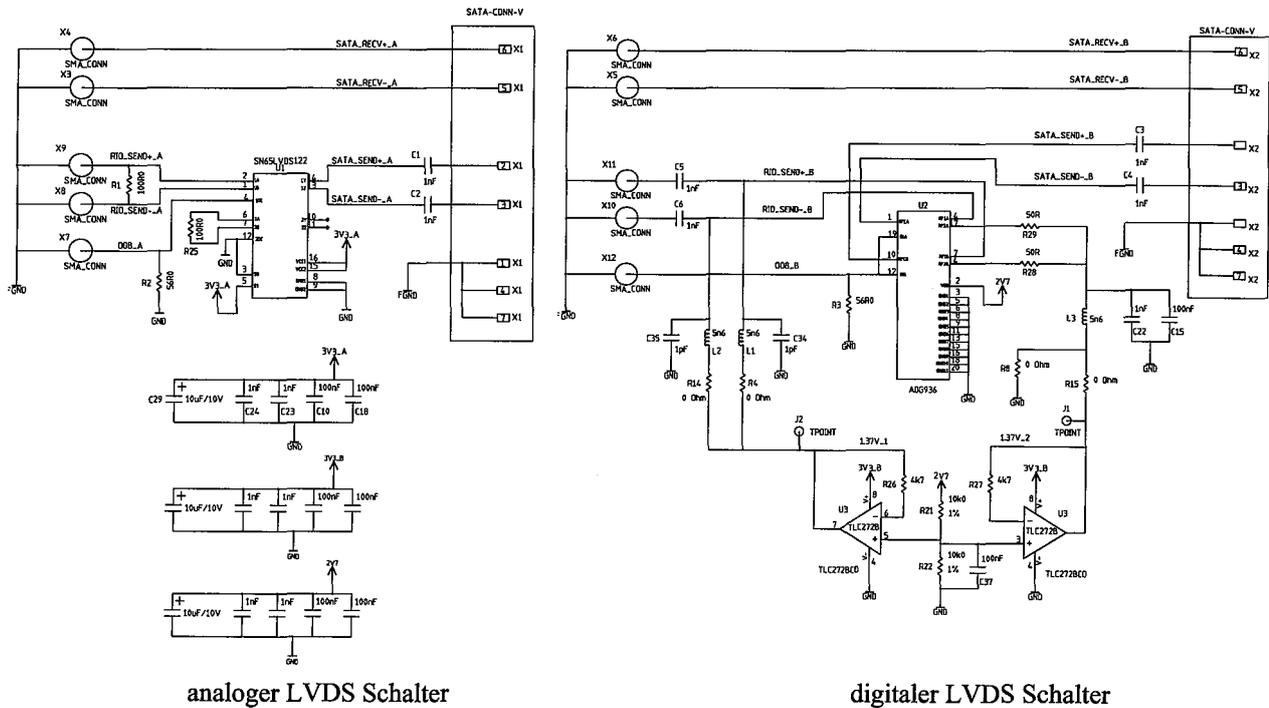


analoger LVDS Schalter



digitaler LVDS Schalter

Abbildung 7: Messung eines Verbindungsaufbaus zwischen FPGA und angeschlossener Festplatte



analoger LVDS Schalter

digitaler LVDS Schalter

Abbildung 8: Stromlaufplan der externen Leiterplatte

vollständige Implementierung des Serial ATA Standards dienen.

Die entwickelte Serial ATA Schnittstelle eröffnet einen breiten Anwendungsbereich. Neben der Verwendung zur normalen Kommunikation mit einer Festplatte ist es auch möglich, einen Konformitätstest der angeschlossenen Geräte zu realisieren. Dabei könnten gezielt Fehler im Timing und im Ablauf der Ansteuerung eingebracht werden und die Reaktion des Prüflings untersucht werden.

Die Defizite der aktuellen FPGA Serien sind auch den Herstellern bekannt. Neuere FPGA Familien bieten zunehmend eine direkte Unterstützung von Serial ATA in den Transceivern. Beispielsweise kann bei der Virtex 4 Serie der Ruhezustand auf der LVDS Leitung mit Hilfe eines Eingangssignals der MGTs realisiert werden. Leider entspricht die übrig bleibende Differenzspannung (40 mV) nicht ganz der Serial ATA Norm. Man müsste Praxistests mit dieser FPGA Familie durchführen, ob es dabei zu einem Problem mit den angeschlossenen Geräten kommt. Die neueste Lattice FPGA Familie, die LatticeSC sowie die LatticeSCM Linie unterstützen Serial ATA explizit laut Datenblatt. In Zukunft wird es daher möglich sein, auf die hier präsentierte externe Beschaltung zu verzichten.

Das beschriebene Serial ATA Interface bietet einige interessante Anwendungsgebiete. Beispielsweise ist es damit möglich, direkt RAID Controller im FPGA zu realisieren. Da sich mit FPGAs auch recht gut komplexe Bildverarbeitungssysteme realisieren lassen, könnte die gemeinsame Integration mit einem Massenspeicherzugriff eine neue Klasse an kostengünstigen, stromsparenden Bildaufzeichnungssystemen entstehen lassen. Anwendungen lassen sich leicht im Bereich der Sicherheitstechnik oder des Objektschutzes finden.

Anhang

Abbildung 8 zeigt den Stromlaufplan der realisierten externen Leiterplatte (aus Platzgründen wurde auf den Abdruck der Stromversorgung verzichtet). Der vom MGT kommende, zum Serial ATA Anschluss (SATA_SEND+/- Signale) führende, Sendepfad wird jeweils über einen der LVDS Schalter geführt. Die Empfangspfade (SATA_RECV+/- Signale) der beiden Lösungen werden ohne Unterbrechung direkt an einen MGT angeschlossen. Zum Aktivieren und Deaktivieren der LVDS Schalter wird jeweils ein Standard LVCMOS Signal vom FPGA verwendet (OOB_A Signal).

Literatur

- [1] Klaus-Peter Deyring (Editor): „Serial ATA: High Speed Serialized AT Attachment“, Revision 1.0a. Serial ATA Working Group (Hrsg.), 7. Januar 2003. <http://www.serialata.org>
- [2] John Masiewicz (Editor): „AT Attachment with Packet Interface Extension 7 (ATA/ATAPI-7 V1)“, Revision 4b. International Committee for Information Technology Standards (INCITS), Technical Committee T13 AT Attachment (Hrsg.), 21. April 2004. <http://www.t13.org>
- [3] A. X. Widmer, P. A. Franaszek: „A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code“. IBM Journal of Research and Development, Vol. 25, No. 5, S440-451, September 1983. <http://www.research.ibm.com/journal/rd/275/ibmrd2705D.pdf>
- [4] Klaus Gravogl: „Entwurf einer seriellen Hochgeschwindigkeitsschnittstelle für Festplatten – geeignet für den Einsatz in programmierbaren Schaltkreisen“. Diplomarbeit an der TU-Wien, Juni 2006.
- [5] Amber Huffman (Editor): „Serial ATA Advanced Host Controller Interface (AHCI)“, Revision 1.0. Intel Corporation (Hrsg.), April 2004. <http://www.intel.com/technology/serialata/ahci.htm>