



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Ausarbeitungarbeit

Johann-Nikolaus Andreae

Konzeptperspektive

für die Sensorik- und Mikrocontrollerplattform
im Formula Student Rennwagen

Inhaltsverzeichnis

1 Einführung	3
2 Ziele	4
2.1 Entwicklungsziele des Formula Student Rennwagens	4
2.2 Erfordernisse für die Telemetrie	4
2.2.1 Fahrzeugverständnis erhöhen	5
2.2.2 Fahrverhalten verbessern	5
2.3 Randbedingungen für die Elektronik	5
2.3.1 Größe und Gewicht	6
2.3.2 Stromversorgung	6
2.3.3 Leitungslänge vom Sensor zur Auswertung	6
3 Ist-Zustand im aktuellen HAWKS-Rennwagen	8
3.1 Sensoren und Aktoren in Rennwagen	8
3.2 Aktuelles Sensor- / Aktor-Konzept	8
3.3 Einschränkungen durch die aktuelle Architektur	9
3.3.1 Hardwareaufwand	10
3.3.2 Softwarearchitektur	11
4 Lösungsansätze für die Beseitigung der Einschränkungen	12
4.1 Zwei statt einem Mikrocontroller	12
4.2 Multicore Mikrocontroller	13
4.3 Einsatz von FPGA statt Mikrocontroller	14
5 Ausblick	15
Literaturverzeichnis	16

1 Einführung

Die Elektronik spielt im Automobil eine immer stärkere Rolle, so auch im Formula Student Rennwagen. Die Elektronik übernimmt im Fahrzeug immer mehr Steueraufgaben. Diesen Anforderungen ist die aktuelle Elektronik im HAWKS-Rennwagen nicht mehr gewachsen. Ziel dieser Ausarbeitung ist es die Ziele die bei der Entwicklung des neuen Konzeptes beachtet werden müssen zu beschreiben und einen Lösungsansatz für die weitere vorgehensweise zu liefern.

Die Formula Student ist ein Wettbewerb für Studenten bei dem es darum geht einen Prototypen für Rennwagen für Hobbyrennfahrer zu entwickeln. Es gibt jährliche Wettbewerbe in verschiedenen Ländern. Das HAWKS-Racing Team¹ ist das Formula Student Team der HAW-Hamburg. Es entwickelt zur Zeit seine fünften Rennwagen.



Abbildung 1.1: HAWK08 beim Wettbewerb in Hockenheim

¹<http://www.hawksracing.de>

2 Ziele

Das oberste Ziel ist es den Wettbewerb zu gewinnen. Die übergeordneten Ziele gelten für den Rennwagen als ganzes. Diesem Zielen untergeordnet sind die Teilziele die sich daraus für die einzelnen Baugruppen ergeben. Hieraus ergeben sich die Ziele für die einzelnen Komponenten. Im folgenden werden wir uns mit den Zielen für die Telemetrie und der Fahrzeugelektronik beschäftigen.

2.1 Entwicklungsziele des Formula Student Rennwagens

Ziel der Formula Student ist es einen schnellen und kostengünstigen Rennwagen zu bauen. Hierbei sind die Regeln [SAE \(2008\)](#) einzuhalten. In den Regeln wird auch festgelegt in welchen Disziplinen der Wagen getestet wird. Eine ausführliche Beschreibung des Ablaufes findet sich in [Andreae \(2008\)](#).

Die Schwierigkeit der Konstruktion des Rennwagens besteht darin, mit den vorgegebenen Einschränkungen ein in allen Bereichen ausgewogenes Fahrzeug zu bauen. Hierbei sind folgende Kriterien ausschlaggebend.

- Geschwindigkeit / Beschleunigung
- Zuverlässige Konstruktion
- geringe Kosten
- Geschäftskonzept

Diese Kriterien lassen sich auch bei der Entwicklung eines PKWs wiederfinden wobei sie unterschiedlich stark bewertet werden.

2.2 Erfordernisse für die Telemetrie

Das Aufgabengebiet der Telemetrie ist zweigeteilt. Zum einen werden Daten über das Fahrzeug gesammelt, mit Hilfe derer das Verständnis des Fahrzeugs seitens der Techniker erhöht

wird. Zum anderen wird der Fahrer während der Fahrt unterstützt. Dies kann durch das Anzeigen von Informationen oder durch Steuerungsaufgaben im Fahrzeug geschehen.

Auch im PKW kann man diese beiden Bereiche wiederfinden. Die Sensoren die das Fahrzeugverständnis erhöhen sind zu großen Teil nur während der Erprobungsphase im Auto. In dem Serienfahrzeug erhöhen sie nur die Produktionskosten. Ein Rennwagen befindet sich immer in dieser Erprobungsphase. Er wird laufen weiterentwickelt und optimiert. Deshalb sind die Sensoren die das Fahrzeugverständnis erhöhen fester Bestandteil des Konzeptes.

2.2.1 Fahrzeugverständnis erhöhen

Die Auswertung der Messdaten durch die Fahrzeugtechniker, welche durch das Telemetriesystem gesammelt werden, findet zu zwei Zeitpunkten statt. Zu erst werden die Daten direkt während der Fahrt in der Livedatenanzeige überwacht. Später können die aufgezeichneten Daten für die genauere Analyse ausgewertet werden. Für die Auswertung der Messdaten ist es wichtig das diese synchron aufgenommen werden. So lassen sich die Daten der unterschiedlichen Sensoren einander zuordnen.

2.2.2 Fahrverhalten verbessern

Systeme die das Fahrverhalten verbessern sind besonders kritische Systeme. Sie greifen Aktive in den Wagen ein. Ein Ausfall oder Fehlfunktion dieser Systeme hat folgen für das Verhalten des Wagens. Systeme die in das Gas-, Brems- und Lenksystem des Wagens eingreifen (Drive by Wire) sind deshalb in den Regeln [SAE \(2008\)](#) nicht erlaubt. Bei der Entwicklung der Assistenzsysteme muss genau analysiert werden welche Folgen ein plötzlicher Ausfall haben könnte. Die meisten Fahrerassistenzsysteme stellen hohe Anforderungen an die Aktualität der Messwerte. Eine Analyse für die Antriebsschlupfregelung findet sich in [Kolbe \(2008b\)](#). Die Zeitanforderung anderer Assistenzsystem muss noch ermittelt werden.

2.3 Randbedingungen für die Elektronik

Bei der Entwicklung der Elektronik müssen die Ziele aus dem Gesamtkonzept des Rennwagens als auch der Telemetrie umgesetzt werden. Zusätzliche Randbedingung ergeben sich durch den Einsatzort: Den Formula Student Rennwagen.

2.3.1 Größe und Gewicht

An dem Ziel, Gewicht zu sparen ist auch die Elektronik beteiligt. Die Schwersten Komponenten sind die Lichtmaschine und der Akku gefolgt von den Leitungen. Aber auch das Gewicht der Platinen ist nicht zu vernachlässigen.

Von der Konstruktion des Wagens wird der zur Verfügung stehende Platz vorgegeben. Bei der Wahl des Platzes müssen Leitungslänge zum Sensor und Erreichbarkeit für die Wartung berücksichtigt werden.

Ein anderes Kriterium das die Größe von unten einschränkt ist die Produzierbarkeit. Um so kleiner die Platinen und Bauteile werden um so schwieriger sind sie zu fertigen bzw. zu verarbeiten und um so höher sind die Fertigungskosten. Diese sind vor allem bei kleinen Stückzahlen, wie im Prototypenbau vorkommen, hoch. Aus diesem Grund muss die Größe so gewählt werden das sich die Teile per Hand bestücken lassen um die Einrichtungskosten von Bestückungsmaschinen zu sparen.

2.3.2 Stromversorgung

Der Strom welcher für den Betrieb der Elektronik benötigt wird, wird aus dem Bordnetz des Formula Student Rennwagens bezogen. Dieses besteht wie auch beim PKW aus Lichtmaschine und Bleiakku. Um Gewicht zu sparen kann der Akku nicht so groß dimensioniert werden das er die alleinige Stromversorgung übernimmt. Die Lichtmaschine ist beim Kawasaki ZX600J gleichzeitig auch Anlasser, kann also nicht durch einen größeren Akku ersetzt werden.

Die Lichtmaschine kann (bei optimalem Lichtmaschinenregler) mit 1kW belastet werden. Bei der Wandlung von der Motorenergie in die elektrische gehen (laut [Reimann und Männel, 2008](#)) mindestens 35% verloren. Für den HAWKS-Rennwagen muss die Verlustleistung noch ermittelt werden. Die Leistung des Motor beträgt in der aktuellen Konfiguration 63kW. Diese Leistung soll möglichst auf die Straße übertragen werden und nicht für die Stromproduktion verwendet werden. Wenn die Lichtmaschine zur Stromerzeugung genutzt wird steigt die Leistungsaufnahme, dies führt zu einem höheren Treibstoffverbrauch. Diesen gilt in einem möglichst geringen Verhältnis zur Geschwindigkeit zu halten (siehe [Formula Student Germany e.V., 2008](#), 7.5 Fuel Efficiency Scoring).

2.3.3 Leitungslänge vom Sensor zur Auswertung

Um so länger die Leitungen zwischen Sensor und Signalauswertung sind, desto größer ist die Wahrscheinlichkeit, dass sie durch elektromagnetische Störungen verändert oder Zer-

stört werden. Bei der analogen Messdatenübertragung lassen sich diese Störungen in den Messdaten nur schwer erkennen. Bei digitalen Übertragungsverfahren kann man falsche Werte durch Prüfsummen erkennen.

3 Ist-Zustand im aktuellen HAWKS-Rennwagen

In diesem Kapitel wird der aktuelle Zustand im Formula Student-Rennwagen des HAWKS-Racing Teams betrachtet. Im ersten Abschnitt werden die im Fahrzeug integrierten Sensoren und Aktoren im aktuellen Zielkonzept betrachtet. Im weiteren gehe ich auf die aktuell verwendete Hardware ein. Zum Schluss werden die Probleme die im aktuellen Hardwarekonzept vorhanden sind behandelt.

3.1 Sensoren und Aktoren in Rennwagen

Die Sensoren und Aktoren sind je nach Aufgabengebiet über den ganzen Rennwagen verteilt (siehe Abb. 3.1). Viele Sensoren sind mehrfach im Fahrzeug vorhanden wie z.B. die Raddrehzahlsensoren für alle vier Räder.

Um herauszufinden welche Sensoren direkt im Wagen von den Aktoren ausgewertet werden und welche nur für die Analyse notwendig sind werde im Abb. 3.2 die Sensoren nach Datenverwertung hervorgehoben.

Die Verarbeitung der von Daten in einem kombinierten Sensor- / Aktor-Modul unterliegen nicht den Zeitanforderungen des Bussystems. Die Übertragung der Daten kann unabhängig von der internen Verarbeitung erfolgen. Müssen die Sensorwerte zu den Aktoren, aufgrund ihres Räumlichen Abstandes, über den Bus übertragen werden müssen unterliegen den Zeitanforderungen des Busse. Hier ist zu überprüfen ob das aktuelle Busskonzept diesen Anforderungen gewachsen ist.

3.2 Aktuelles Sensor- / Aktor-Konzept

Im aktuellen Konzept werden die Sensoren und Aktoren über einen Time Triggert CAN-Bus verbunden (siehe Abb. 3.3). Zur Anbindung der Motorsteuerung (ECU) wurde ein Gateway entwickelt was die RS232-Schnittstelle der Motorsteuerung auf den CAN-Bus übersetzt. Zur

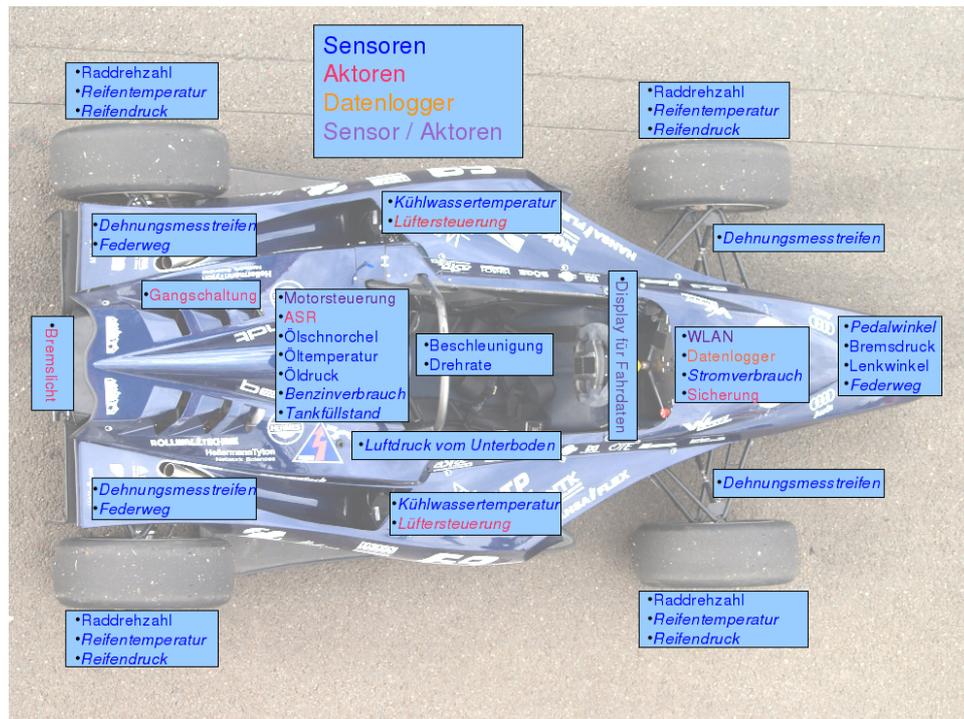


Abbildung 3.1: Position der Sensoren und Aktoren im Rennwagen

Datenauswertung werden die Daten per WLAN an die Basisstation gesendet. Zusätzlich werden die Daten vom Datenlogger auf einer SD-Karte aufgezeichnet. Dies ist nötig da die WLAN-Verbindung nicht unterbrechungsfrei sichergestellt werden kann und somit Lücken in der Aufzeichnung entstehen können.

In den Sensor- und Aktor-Modulen wird der AT90CAN128 von [Atmel \(2008\)](#) eingesetzt. Hierbei handelt es sich um einen 8-Bit RISC Prozessor mit 16MHz. Er verfügt über 53 I/O-Pins die teilweise mit zuschaltbaren Sonderfunktionen belegt sind. Er verfügt über 8 multiplexed A/D-Wandler-Eingänge, SPI- und I2C-Schnittstelle, 4 Timer (2 x 8bit, 2 x 16bit), CAN-Bus und viele andere Funktionen. Beim Stromverbrauch ist der AT90CAN mit 150mW recht sparsam. Bei Abnahme von mehr als 500 Stück kostet er um die 8€ pro Stück. Aufgrund seiner Bauform ist er noch gut mit der Hand zu löten.

3.3 Einschränkungen durch die aktuelle Architektur

Die aktuelle Architektur hat Einschränkungen sowohl auf der Hardwareseite als auch auf der Softwareseite.

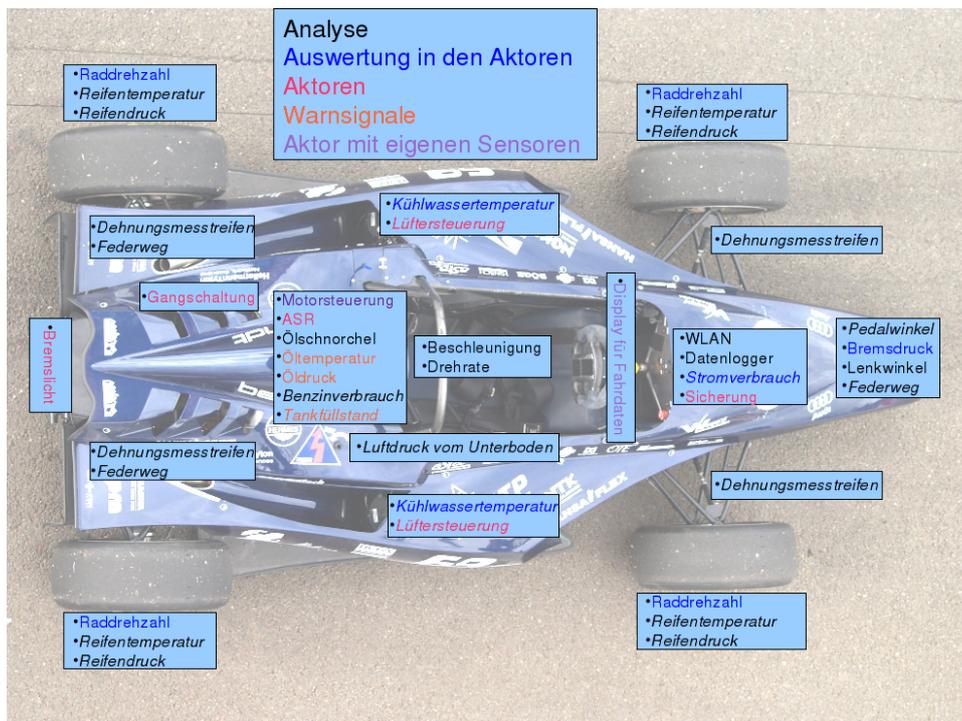


Abbildung 3.2: Verwendungszweck der Sensordaten des Rennwagens

3.3.1 Hardwareaufwand

Das aktuelle Konzept sieht für jedes Modul eine eigene Platine mit Spannungswandler, Mikrocontroller, CAN-Bus-Transceiver und Debugschnittstelle vor. Dieses Konzept ist in vielen Bereichen redundant ausgelegt, was für hohe Flexibilität im Fehlerfall sorgt. Bei Störungen in einem Modul wird es einfach abgeklemmt oder ausgetauscht. Andere Module sind nicht in ihrer Funktion beeinträchtigt (mit Ausnahme der Störungen die sich auf den Bus auswirken). Der Nachteil dieses Konzeptes ist der hohe Platzverbrauch was auch Gewicht bedeutet. 1 dm^2 FR4¹ wiegt ungefähr 20g. Im HAWK08 sind 6 dm^2 verbaut. Diese Fläche wird sich beim Ausbau zum Zielkonzept, bei beibehaltung der aktuellen Architektur in etwas verdoppeln. Dies entspricht der Platinenfläche die in einem Audi verbaut ist² und wiegt ohne Bauteile $\frac{1}{4}\text{ kg}$.

Im aktuellen Zielkonzept, wie in Abb. 3.1 dargestellt, sind schätzungsweise 90 Sensoren verbaut. Hierzu müssen 55 analoge Messwerte aufgenommen werden und 50 digitale I/O-Schnittstellen bedient werden. In Abhängigkeit mit der Auslastung der Mikrocontroller seitens der Software macht das min. 25 AVR8-Module.

¹Standard Material für Leiterplatten bestehend aus Epoxidharz und Glasfasergewebe

²Laut Aussage des Audi-Fahrwerkleiters

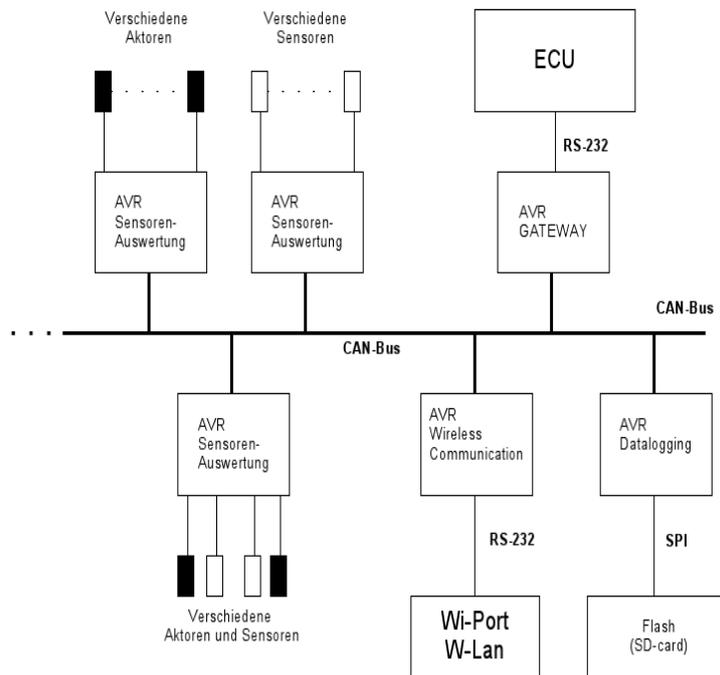


Abbildung 3.3: Sensoren und Aktoren am CAN-Bus(Quelle: [Schuckert, 2007](#))

3.3.2 Softwarearchitektur

Die Softwarearchitektur in den Sensor- und Aktor-Modulen ist TimeTriggert (vgl. [Schuckert, 2007](#)). D. h. der Scheduler gibt genau vor in welchem Zeitslot der Mikrocontroller welche Aufgabe erledigt. Es handelt sich um einen offline Scheduler, es wird also im vornherein in einer Tabelle festgelegt welcher Prozess wann wie lange arbeitet. Die Zeiten der einzelnen Module werden über den TimeTriggert CAN-Bus synchronisiert (vgl. [Haase, 2007](#)).

Diese TimeTriggert-Konzept hat den Vorteil das genau festgelegt ist wann welcher Sensor ausgelesen wird und wann die Daten auf dem Bus übertragen werden. Der Nachteil liegt in der Umsetzung auf der aktuellen Architektur. Damit die Prozesse nicht in ihrer Abarbeitung unterbrochen werden, ist im Mikrocontroller nur der Interrupt des Timers für den Scheduler zulässig. Weitere Interrupts bringen das TimeTriggert Konzept durcheinander. Dies hat zur Folge das kombinierte Sensor-Aktor-Module nicht auf interne Ereignisse reagieren können die nicht in das Zeitslotkonzept passen.

So kann z. B. das Strom-Sicherungsmodul seine Messwerte nicht auf den CAN-Bus übertragen, da es während der Übertragungszeit nicht auf Stromspitzen reagieren kann und somit die geforderte Reaktionszeit verletzt.

4 Lösungsansätze für die Beseitigung der Einschränkungen

Um die aktuellen Einschränkungen aufzulösen kann man einen schnelleren Prozessor einsetzen. Dies hatte zur Folge das die Zeit in der der Prozessor mit der Kommunikation beschäftigt ist kürzer wird. Es werden die Einschränkungen also nur minimiert nicht aber beseitigt. Bei einem optimalen Ansatz laufen die Kommunikation und die Erfassung der Messdaten parallel.

4.1 Zwei statt einem Mikrocontroller

Die einfachste Möglichkeit parallelität herzustellen, ist es zwei AVR8 statt einem einzusetzen. Der eine übernimmt die Kommunikation auf dem CAN-Bus, der andere die reiner Aktor arbeiten.

Vorteile

- Softwarearchitektur muss nur geringfügig verändert werden
- Hardware ist schon bekannt.

Nachteile

- Auch zwischen den beiden Mikrocontrollern muss es Kommunikation geben um z.B. die Konfiguration des Aktors verändern zu können
- Doppelter Stromverbrauch
- Doppelter Hardwareaufwand

Durch die nötige kommunikation zwischen den beiden AVR8s, gibt es auch in diesem System Zeiten wo der Aktor mit der Kommunikation beschäftigt ist.

4.2 Multicore Mikrocontroller

Multicore Prozessoren sind im SoC¹-Bereich selten. Der Propeller von Parallax verfügt über 8 Cores aber bis auf eine I2C Controller für einen externen Speicher über interne Hardwarekomponenten (siehe [Parallax, 2008](#)) und ist somit nicht für den automotive Bereich geeignet.

Der einzige Multicore Mikrocontroller der gefunden werden konnte ist der MPC5668G von Freescale. Er verfügt über zwei Prozessorecores sowie jedemenge Schnittstellen wie CAN-Bus, FlexRay, Ethernet, 36-Chanel AD-Wandler, 4 x I2C und vieles mehr (siehe [Freescale, 2008](#)).

Vorteile

- Fertiger Mikrocontroller
- Große Anzahl von Schnittstellen
- Für Automotive optimiert

Nachteile

- 208-Pin BGA Bauform. Teure (Prototypen kosten zwischen 1500 und 2000€je Modul) und zeitaufwendige Platinenentwicklung (siehe [Andreae, 2008](#)).
- Schwere Beschaffung. Es konnte kein Händler gefunden werden der den Chip im Programm hat.
- Unbekannter Preis
- noch kein Öffentliches Datenblatt
- Andere Prozessorarchitektur (PowerPC). Software muss angepasst werden.

Nach den verfügbaren Daten zu Urteilen handelt es sich bei diesem Chip zur Zeit um ein Vorserienprodukt.

Aufgrund seiner Bauform ist dieser Mikrocontroller für den HAWKS-Rennwagen nur attraktiv wenn er also System on Module² erhältlich ist. Hiermit ist aufgrund der Zielgruppe des Chips nicht zu rechnen.

¹System on Chip: Hardware Komponenten die bei einem Prozessor extern angeschlossen werden sind im Chip integriert

²Auf einem Modul aufgebautes System, welches neben dem Mikrocontroller auch den Speicher und die Spannungswandlung enthält.

4.3 Einsatz von FPGA statt Mikrocontroller

Beim Hardware Software Codesign werden Softwaremodule in die Hardware verlagert. Die interne Beschaltung eines FPGAs wird mit speziellen Programmiersprachen wie VHDL und Verilog programmiert. Die Beschaltung lädt der FPGA beim Booten aus einem externen Speicher.

Vorteile

- Softwaremodule lassen sich durch die Verlagerung in die Hardware parallelisieren und beschleunigen.
- Es müssen nur die wirklich benötigten Hardwaremodule mit Strom versorgt werden.
- Gute Verfügbarkeit
- Es sind Bauformen verfügbar die gerade noch per Hand zu löten sind.
- Große Anzahl von frei konfigurierbaren Pins

Nachteile

- 3 bis 4 mal so teuer wie Mikrocontroller
- Hardwaremodule müssen zum Teil selber entwickelt werden.
- Größere Anpassungen der Software durch Auslagerung in die Hardware nötig.
- Höherer Stromverbrauch als bei einem optimiert produzierten ASICs³

Um den höheren Stromverbrauch und die höheren Hardwarekosten zu kompensieren müssen mehrere Module zusammengefasst werden. Dies führt zu einer Veringerung der Redundanz, da bei einem Ausfall mehr Sensoren oder Aktoren betroffen sind. Das Zusammenfassen hat aber den Vorteil dass die Zahl der Platinen sinkt und die kompakter aufgebaut werden können.

Ein Nachteil der sich bei der Zusammenlegung von Sensoren die nur für die Auswertung benötigt werden mit den übrigen Sensoren und Aktoren ergibt, ist dass die Kosten für das Modul im Kostreport angegeben werden müssen (siehe [SAE, 2008](#), 4.3.7.4 Transponders, Data Acquisition, Video and Radio Systems) und somit den Fahrzeugpreis steigern. Dies muss bei der Zusammenlegung beachtet werden.

³ASIC steht für: **a**pplication **s**pecific **i**ntegrated **c**ircuit. Hierbei handelt es sich um eine elektronische Schaltung die in nach Kundenwunsch in eine IC integriert wurde.

5 Ausblick

Aus den Lösungsansetzen hat sich das FPGA-Konzept als das erfolgversprechenste herauskristallisiert. Im weiteren Projektverlauf ist der Einsatz von FPGAs als Lösung der aktuellen Einschränkungen zu überprüfen. Hierfür sind folgende Schritte notwendig:

- Analyse der Hardware / Softwarepartitionierung. Dies wird in [Kolbe \(2008a\)](#) behandelt.
- Entwicklung von Kriterien für die Zusammenlegung der Sensor und Aktormodule.
- Recherche über die Verfügbarkeit von fertigen IP-Cores¹.
- Implementierung eines Modules auf einem Evaluationsboard. Es sind zwei verschiedene in der Hochschule verfügbar, welche im weiteren genauer betrachtet werden müssen.
 - Nexus-2 von Digilent. Ein universelles FPGA-Board welches sich über 4 x 8 Pin und eine 40 Pin I/O-Schnittstelle gemäß den eigenen Bedürfnissen erweitern lässt.
 - XAS1600E von Xilinx. Ein Speziell für automotive Einsatzgebiete optimiertes Board.
- Entwicklung eines Prototypen zum Einbau in den HAWKS-Rennwagen.

¹Als intellectual property core werden im Chipdesign die Baupläne für die Hardwaremodule bezeichnet.

Literaturverzeichnis

- [Andrae 2008] ANDRAE, Johann-Nikolaus: *Das Lenkraddisplay eines Formula Student Rennwagens: Von der Analyse, über die Hardware- und Linuxtreiberentwicklung bis zum Prototypen*, Hochschule für Angewandte Wissenschaft Hamburg, Bachelorarbeit, August 2008. – URL http://opus.haw-hamburg.de/volltexte/2008/637/pdf/Andrae_Bachelorarbeit.pdf
- [Atmel 2008] Atmel: *Datasheet AT90CAN*. August 2008. – URL http://www.atmel.com/dyn/resources/prod_documents/doc7679.pdf
- [Formula Student Germany e.V. 2008] Formula Student Germany e.V.: *Formula Student Germany Rules 2009*. 2008. – URL http://www.formulastudent.de/uploads/media/FSG_RULES_2009_Version20081107_01.pdf
- [Freescale 2008] Freescale: *Fact Sheet MPC5668G*. Rev. 0. November 2008. – URL http://www.freescale.com/files/microcontrollers/doc/fact_sheet/MPC5668GFS.pdf
- [Haase 2007] HAASE, Sebastian: *Telemetrie im Formula Studente Rennwagen auf Basis von CAN-Bus, Datenspeicherung und Wireless LAN Technologie*, Hochschule für Angewandte Wissenschaft Hamburg, Bachelorarbeit, 2007
- [Kolbe 2008a] KOLBE, Felix: *Betrachtung von FPGA-Konzepten im Umfeld eines Formula Student Rennwagens*, Hochschule für Angewandte Wissenschaft Hamburg, Ausarbeitung, Dezember 2008
- [Kolbe 2008b] KOLBE, Felix: *Redundanzkonzept eines Time-Triggered Bussystems in einem Formula Student Rennwagen: Modellierung, Implementierung und Anwendung in der Antriebsschlupfregelung*, Hochschule für Angewandte Wissenschaft Hamburg, Bachelorarbeit, August 2008. – URL http://opus.haw-hamburg.de/volltexte/2008/638/pdf/Felix_Kolbe_Bachelorarbeit.pdf
- [Paralax 2008] Paralax: *Datasheet P8X32A*. Rev. 1.1. September 2008. – URL <http://www.parallax.com/Portals/0/Downloads/docs/prod/prop/PropellerDatasheet-v1.1.pdf>
- [Reimann und Männel 2008] REIMANN, Dipl.-Ing. W. ; MÄNNEL, Dipl.-Ing. R.: *Verbrauchseinfluss der elektrischen Energie im Fahrzeug*. In: *ATZ elektronik* (2008), Februar. – ISSN 1862-1791 70934

- [SAE 2008] SAE: *2009 Formula SAE® Rules*. 2008. – URL <http://students.sae.org/competitions/formulaseries/rules/rules.pdf>
- [Schuckert 2007] SCHUCKERT, Simon M.: *Microcontrollerbasierte Telemetrie und Echtzeitauswertung von Sensordaten im Formula Student Rennwagen*, Hochschule für Angewandte Wissenschaft Hamburg, Bachelorarbeit, 2007