



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

# Bachelorarbeit

Jasper Salathé

**Integration einer CMOS-Kamera mit parallelem Interface in ein  
System-on-Chip basiertes Spurführungssystem**

*Fakultät Technik und Informatik  
Studiendepartment Informatik*

*Faculty of Engineering and Computer Science  
Department of Computer Science*

Jasper Salathé

**Integration einer CMOS-Kamera mit parallelem Interface in ein  
System-on-Chip basiertes Spurführungssystem**

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung  
im Studiengang Technische Informatik  
am Department Informatik  
der Fakultät Technik und Informatik  
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr.-Ing. Bernd Schwarz  
Zweitgutachter: Prof. Dr. rer. nat. Michael Schäfers

Eingereicht am: 30. August 2012

**Jasper Salathé**

**Thema der Arbeit**

Integration einer CMOS-Kamera mit parallelem Interface in ein System-on-Chip basiertes Spurführungssystem

**Stichworte**

CMOS-Kamera, 24B7525A, MT9V022 Bildsensor, Active-Pixel Sensor, System-on-Chip FPGA, MicroBlaze, I2C, IIC, Two Wire Interface, Kamera-Konfiguration, paralleles Kamerainterface, FrameGrabber, VGA-Interface, VHDL

**Kurzzusammenfassung**

Diese Arbeit behandelt die Integration einer CMOS-Kamera mit parallelem Interface in ein System-on-Chip basiertes Spurführungssystem. Eine Übersicht über die Funktionsweise von Active-Pixel CMOS-Kameras im Allgemeinen und im Speziellen anhand der 24B7525A Kamera wird gegeben. Eine Konfiguration wird hergeleitet und ein Interface zur softwareseitigen Konfiguration der Kamera durch einen MicroBlaze wird entwickelt. Aus dem Datenstrom der Kamera wird ein Bildsignal nach VGA-Standard generiert. Zum Speichern eines Bildes aus dem laufenden Datenstrom wird ein FrameGrabber eingesetzt.

**Jasper Salathé**

**Title of the paper**

Integration of a CMOS Imaging Sensor with a Parallel Interface into a System on Chip based Lane Following System

**Keywords**

CMOS-Camera, 24B7525A, MT9V022 Imaging Sensor, Active Pixel Sensor, System on Chip FPGA, MicroBlaze, I2C, IIC, Two Wire Interface, Image Sensor Configuration, Parallel Image Sensor Interface, Frame Grabber, VGA Interface, VHDL

**Abstract**

This thesis deals with the integration of a CMOS Imaging Sensor with a Parallel Interface into a System on Chip based Lane Following System. An overview about Active Pixel CMOS Imaging Sensors in general and about the 24B7525A Imaging Sensor in particular is provided. A configuration for the 24B7525A Imaging Sensor is deduced and a software interface to configure the Imaging Sensor by a MicroBlaze is developed. A standard VGA signal is generated from the image data provided by the Image Sensor with a VGA-Controller. To store a frame from the image data a FrameGrabber is used.

# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>1</b>
<b>2. Übersicht zum SOC-Kamera-System</b>	<b>4</b>
2.1. Hardware-Komponenten . . . . .	6
2.2. Dual-MicroBlaze System . . . . .	8
2.3. Bildverarbeitungs pipeline . . . . .	9
2.4. Regelung der Aktorik . . . . .	11
<b>3. CMOS-Bildsensoren</b>	<b>12</b>
3.1. Aufbau eines Active Pixel CMOS-Bildsensors . . . . .	14
3.2. Integrierte Funktionen des MT9V022-Bildsensors . . . . .	17
3.3. Parametrisierungsgrößen des MT9V022-Bildsensors . . . . .	20
<b>4. Konfiguration der Videology 24B7525A CMOS-Kamera über den I<sup>2</sup>C-Bus</b>	<b>30</b>
4.1. Elektrische Grundlage des I <sup>2</sup> C-Bus . . . . .	31
4.2. I <sup>2</sup> C-Bus Protokoll . . . . .	31
4.3. FPGA-Interface des MT9V022 Bildsensors an den I <sup>2</sup> C-Bus . . . . .	35
4.4. I <sup>2</sup> C-Bus Interface des MicroBlaze . . . . .	35
4.5. Checkliste zum Integrationsvorgang . . . . .	37
<b>5. Parametrisierungssoftware für den MT9V022 Bildsensor</b>	<b>40</b>
5.1. Register zur Parametrisierung der Kamera . . . . .	40
5.2. Parametersatz für den Betrieb der Kamera . . . . .	40
5.3. Softwareinterface zur Konfiguration der Kamera . . . . .	43
<b>6. Kamera-Integration in das System-on-Chip basierte Bildverarbeitungssystem</b>	<b>46</b>
6.1. Einbindung des Dateninterfaces der Kamera . . . . .	49
6.2. Timing nach VGA-Standard . . . . .	51
6.3. Einhalten des Timings auf Kameraseite . . . . .	54
6.4. Kamera-spezifische Parametrisierung des VGA-Interfaces . . . . .	55
6.5. Anpassung eines FrameGrabbers an die Pixelrate der Kamera . . . . .	60
<b>7. Analyse der durch die 24B7525A-Kamera gelieferten Bilddaten</b>	<b>64</b>
7.1. Linsenverzeichnung der Objektive . . . . .	64
7.2. Auswirkungen der nichtlinearen Digitalisierung des MT9V022 Bildsensors . . . . .	68

<b>8. Weitere Entwicklungsschritte</b>	<b>70</b>
<b>9. Zusammenfassung</b>	<b>72</b>
<b>A. Anhang</b>	<b>73</b>
A.1. Register des MT9V022 Bildsensors . . . . .	73
A.2. Konfigurationswerte für die Register des MT9V022 Bildsensors . . . . .	98
A.3. Listing des Sourcecodes . . . . .	101
<b>Literatur</b>	<b>102</b>
<b>Tabellenverzeichnis</b>	<b>106</b>
<b>Abbildungsverzeichnis</b>	<b>107</b>

# 1. Einleitung

Integrierte Mikroprozessorsysteme sind heute als zentrale Recheneinheiten in Smartphones, digitalen Bildaufnahmeggeräten und Navigationsgeräten nicht wegzudenken [5]. Sie übernehmen dort Steuer- und Regelaufgaben jeglicher Art, verarbeiten Nutzereingaben und bereiten Informationen für den Anwender auf. Ein weiteres Beispiel ist die Verwendung im Automobil. In den heute erhältlichen Automobilen sorgt eine ganze Reihe von Mikroprozessoren für einen verbesserten Fahrkomfort, mehr Sicherheit, einen reduzierten Energieverbrauch sowie eine multimediale Unterhaltung der Insassen. So gibt es beispielsweise zur Realisierung eines Antiblockiersystems oder zur optimalen Regelung des Motors eingebettete Mikroprozessoren. [8]

Dabei wird es bei einer immer größeren Anzahl von integrierten Mikroprozessorsystemen wichtig, die Kosten für diese Mikroprozessorsysteme und somit die sich für das Produkt ergebenden Folgekosten gering zu halten. Einerseits kann dies durch die Verwendung von in großer Stückzahl hergestellten Komponenten geschehen. Andererseits spielt die Anzahl der verbauten Komponenten in einem System eine Rolle. Durch die Integration vieler Bestandteile eines Systems in eine geringere Anzahl von Chips ergibt sich ein geringerer Fertigungsaufwand bei der Bestückung der Platinen, aufgrund der geringeren Anzahl von zu routenden Signalen eine allgemein geringere Komplexität der Platinen. Weiterhin ergibt sich ein geringerer Entwicklungsaufwand, da bestimmte Probleme schon beim Erstellen der Chips gelöst wurden. [24, S. 11 ff.]

Dabei entsteht ein System auf einem Chip (System-on-Chip), das ein Mikroprozessorsystem mit flüchtigen und nichtflüchtigen Speichern, Interruptcontroller und vielfältigen IO-Modulen enthält [26]. Bei ausreichender zu erwartenden Stückzahlen wird gegebenenfalls ein an die spezialisierten Anforderungen eines Produktes angepasster Chip (ASIC; Application-Specific Integrated Circuit) entwickelt, ein Prozess, der viel Zeit, spezialisiertes Wissen und Ressourcen benötigt [20]. Können die Entwicklungskosten nicht durch eine entsprechend hohe Stückzahl gedeckt werden, so stellen konfigurierbare Logikschaltungen wie CPLDs (Complex Programmable Logic Device) oder FPGAs (Field Programmable Gate Array) eine Alternative dar [24, S. 12]. Zwar ist der Stückpreis dieser Logikbausteine im Vergleich zu den in großer Stückzahl hergestellten ASICs relativ hoch, unterbietet diesen jedoch bei geringeren Stückzahlen [20]. Abgesehen von ihrer Rolle in Kleinserien stellen gerade FPGAs eine Plattform zum Testen von in der Entwicklung

befindlichen ASICs dar. Durch FPGAs werden dabei entwickelte ASIC-Designs ohne die teure Produktion des Chips auf einer Fertigungsstraße simuliert und getestet [21].

Im Rahmen des Projektes Fahrassistenz- und Autonome Systeme (FAUST) [12] entstehen an der HAW-Hamburg Systeme, die in der Lage sind, ein Fahrzeug autonom mit Hilfe einer Fahrspurerkennung auf einer Straße fahren zu lassen, dabei Kreuzungen zu erkennen und gegebenenfalls anderen Fahrzeugen die Vorfahrt zu gewähren. Weitere Disziplinen, wie das Überholen eines anderen Fahrzeuges sowie das Einparken, gehören dabei auch zum Funktionsumfang [9]. Ein Großteil der entwickelten Systeme verwendet Prozessoren der x86 Architektur, um die nötige Bildverarbeitung zu bewältigen, und bestehen entsprechend aus einem vollwertigen PC-System mit angeschlossenen Mikrocontrollern zur Einbindung der Hardwarekomponenten.

Die vorliegende Arbeit befasst sich mit einem SAV (System on Chip Autonomous Vehicle), das bewusst auf den Einsatz solcher PC-Komponenten verzichtet und die rechenintensive Bildverarbeitung mit Hilfe von Hardwarebeschleunigern realisiert. Diese Hardwarebeschleunigerkomponenten sind als Register Transfer Logic (RTL) in einem FPGA realisiert. Zusätzlich wurden auf dem FPGA weitere Komponenten zur Geschwindigkeitsregelung, zur Ansteuerung und zum Auslesen der Sensorik sowie zur Ansteuerung der Fahrzeugfunktionen realisiert. Auf den Aufbau dieses Systems wird in Kapitel 2 genauer eingegangen. Das System entstand im Rahmen mehrerer Abschlussarbeiten, die vorliegende Arbeit baut auf diesen auf. Bisher nutzte es zur Bilderfassung eine Sony FCB-PV10 CCD Kamera mit 30 Bildern pro Sekunde bei einer Auflösung von 640x480 Pixeln in Farbe. Da zur Fahrspurerkennung jedoch lediglich der Helligkeitswert genutzt wird und jeder zweite übermittelte Pixel ein Chrominanzwert und somit eine Farbinformation ist, sinkt die Menge der verwertbaren Daten pro Bild. Hinzu kommt, dass die Kamera lediglich über einen Blickwinkel von 46° verfügt. Hierdurch kann maximal eine einzelne Fahrspurmarkierung erfasst werden und gerät entsprechend schnell aus dem Blickwinkel. Das Objektiv ist weiterhin fest an der Kamera fixiert und nicht wechselbar.

Um eine Bildrate von 60 Bildern pro Sekunde zu erhalten und das Objektiv der Kamera wechselbar zu gestalten, befasst sich diese Arbeit mit der Integration der 24B7252A-Kamera des Unternehmens Videology. Sie erlaubt eine Auflösung von bis zu 752x480 Pixeln und liefert ein monochromes Bild mit einer Auflösung der Helligkeitswerte von 10 Bit. Zur Bilderfassung verwendet sie einen in CMOS-Technologie gefertigten MT9V022-Bildsensor der 2008 von der Firma Micron ausgegliederten Bildsensoren-Sparte Aptina.

Die Integration dieser Kamera in das vorhandene Spurführungssystem ist Inhalt dieser Arbeit. Es wurden eine fehlertolerante Prozeduren zur Parametrisierung der Kamera über den I<sup>2</sup>C-Bus und ein Parametersatz entwickelt. Weiterhin setzt diese Arbeit die Anforderungen durch die neue Kamera an die Bildverarbeitungspipeline von der Anbindung der Kamera über das Kamerainter-

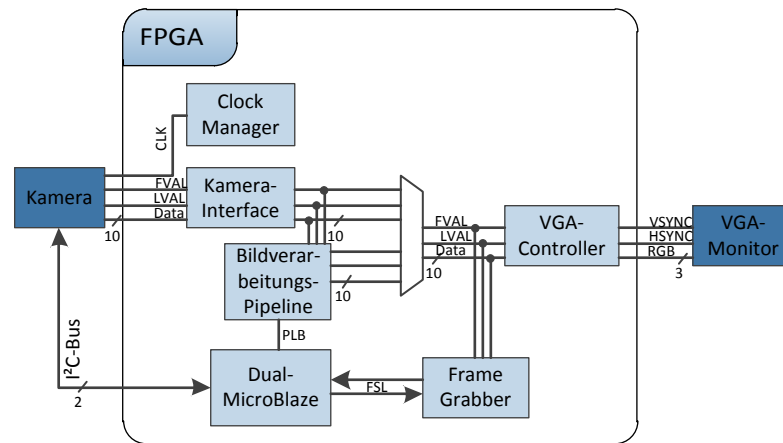


Abbildung 1.1.: in dieser Arbeit erstelltes System

face bis zum VGA-Interface zur Ausgabe auf einem Bildschirm nach VGA-Standard und dem FrameGrabber zum Abgreifen und Abspeichern der Bilddaten aus dem laufenden Bildstrom um. Dies ist in Abbildung 1.1 verdeutlicht.

Im Folgenden wird zuerst ein Überblick über das Gesamtsystem geschaffen (siehe Kapitel 2), danach wird ein Einblick in die Technologie der CMOS-Bildsensoren im Allgemeinen und speziell in den verwendeten MT9V022-Bildsensor gegeben (siehe Kapitel 3). Es folgt eine Darstellung der Konfiguration des Bildsensors über den I<sup>2</sup>C-Bus (siehe Kapitel 4) sowie der entwickelten Konfiguration und Konfigurationssoftware (siehe Kapitel 5). Weiterhin wird die Integration des neuen Bildsensors in das Bildverarbeitungssystem beschrieben (Kapitel 6). Abschließend wird eine Analyse der durch den neuen Bildsensors gelieferten Bilddaten und der vorhandenen Objektiv durchgeführt (siehe Kapitel 7), weitere Entwicklungsschritte vorgeschlagen (siehe Kapitel 8) sowie die Arbeit zusammengefasst (siehe Kapitel 9).



## 2. Übersicht zum SOC-Kamera-System

Das Spurführungssystem lässt sich in mehrere Komponenten sowohl auf logischer als auch auf der Ebene der Hardware unterteilen. Diese einzelnen Komponenten werden in diesem Kapitel detailliert beschrieben.

Das System wurde von den Ergebnissen der Masterarbeit von Marco Kirschke übernommen und angepasst. In seiner Arbeit sind die Komponenten dokumentiert. [16]

Die Hardware wird durch das Dual-MicroBlaze System initialisiert. Über den SAV-Connector kann das Dual-MicroBlaze System auf die Regelung der Aktoren und die Bildverarbeitungspipeline Einfluss nehmen und aktuelle Betriebsparameter auslesen. Zusätzlich kann über den FrameGrabber aus dem laufenden Bildstrom der Bildverarbeitungspipeline ein aktuelles Bild zur Analyse ausgelesen und an den PC weitergeleitet werden. Über das VGA-Interface können aktuelle Bilder aus der Bildverarbeitungspipeline auf einem Monitor mit einem VGA-Eingang ausgegeben werden. Dies ist gerade während der Entwicklung neuer Komponenten wichtig, da so eine visuelle Kontrolle der Ergebnisse stattfinden kann. Es ist aber auch bei der Konfiguration und Analyse des Systems hilfreich.

Neben der angeschlossenen Hardware lässt sich das System grob in mehrere Komponenten unterteilen:

- Bildverarbeitungskette, zuständig für die Gewinnung von Lenkdaten aus den Kamerabildern.
- Dual-MicroBlaze System, zuständig für die Parametrisierung und Überwachung der Hardwarebeschleuniger und -komponenten sowie für die Analyse der Laserscannerdaten. Übernimmt auch die Kommunikation mit einem PC zur externen Kontrolle und Überwachung.
- Regelung der Aktorik, zuständig für die Geschwindigkeitsregelung und Ansteuerung der Aktorik über PWM-Signale.

Diese Komponenten sind in Abbildung 2.1 grob dargestellt.

Im Rahmen dieser Arbeit wurde das Kamerainterface, der FrameGrabber, das VGA-Interface sowie das Dual MicroBlaze System angepasst.

## 2. Übersicht zum SOC-Kamera-System

---

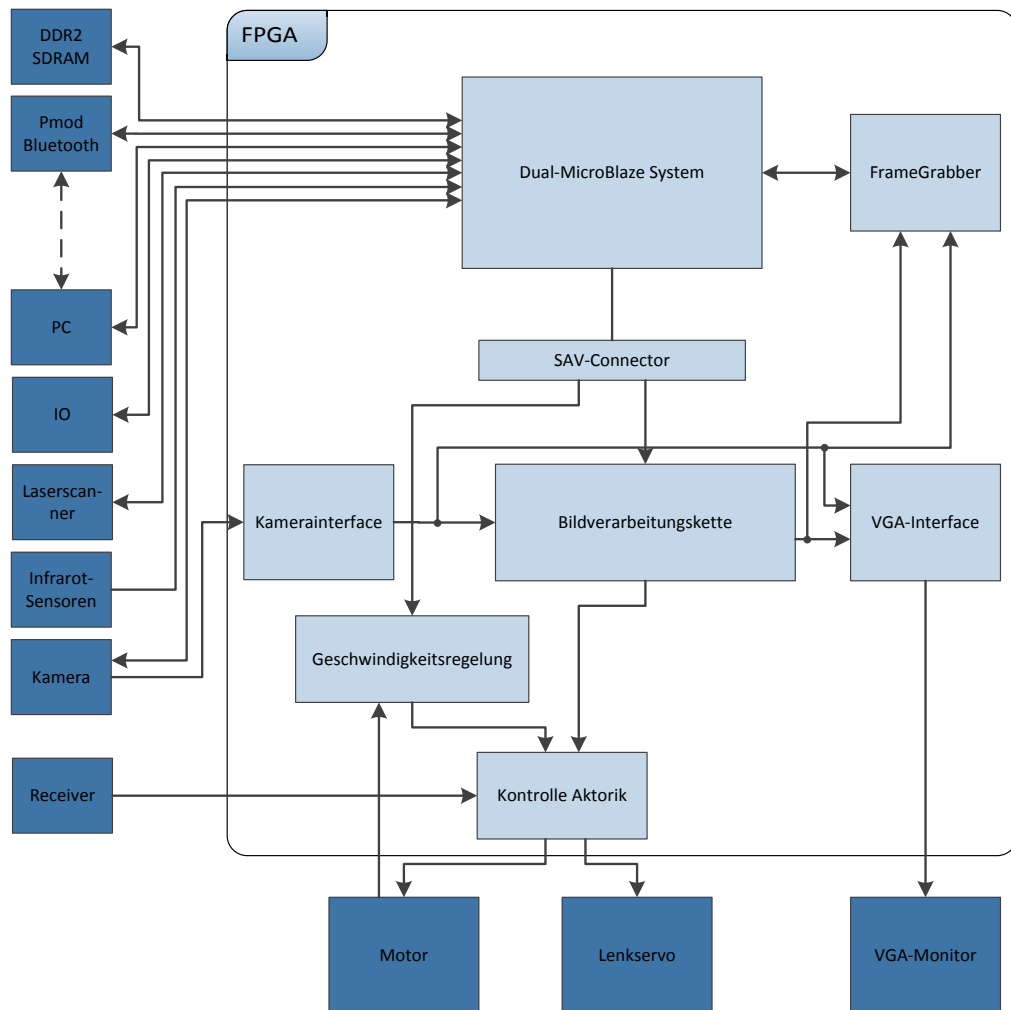


Abbildung 2.1.: grobe Systemübersicht über die Komponenten des SOC-Kamera-Systems

## 2.1. Hardware-Komponenten

Als zentrale Plattform dient ein Spartan-3A DSP Starter Kit mit einem Spartan-3A DSP FPGA in einem fg676-Gehäuse der Firma Xilinx. Er ist in der Version mit 1,8 Millionen Gattern verbaut, womit 16640 Slices, 1512 kBit Block RAM, 8 Digital Clock Manager (DCM), 519 I/O's und 84 DSP48A realisiert wurden. DSP48A sind für Bild- und Signalverarbeitungsfunktionalitäten besonders wichtig, da sie jeweils einen 18 Bit breiten Multiplizierer und einen 48 Bit breiten Addierer, der auch als Akkumulator verwendet werden kann, beinhalten. Diese sind zwar auch mit Slices zu realisieren. Dabei treten jedoch durch die Architektur dieser FPGA-Elemente signifikant höhere Laufzeiten auf. [31, 30, S. 373]

Das Spartan-3A DSP Starter Kit Board verfügt weiterhin über 8 vom FPGA ansteuerbare LED's, 4 Druckknöpfe und 8 DIP-Schalter. Als nicht flüchtigen Speicher verfügt es jeweils über einen seriell und einen parallel ansteuerbaren Flash-Baustein und als flüchtigen Speicher 128 MByte DDR2-SDRAM, zum Beispiel als System Speicher von MicroBlaze-Mikrokontrollern.

Hinzu kommen diverse Interfaces [32]:

- 10/100/1000 Ethernet PHY
- JTAP-Programmier- und Konfigurationsinterface
- RS232 Schnittstelle mit Levelshiftern
- VGA-Interface mit einem Widerstandsnetz zur Signalgenerierung
- diverse Pfostenleisten
- Extension-Header zum Herausführen weiterer I/O's
- Pmod-Header für Erweiterungsmodule

Das System verfügt über eine Reihe von extern angeschlossenen Sensoren zur Erfassung der Umgebung, Aktorik zur Fortbewegung sowie weitere extern angeschlossene Module. Extern angeschlossen sind:

- 24B7525A-Kamera des Unternehmens Videology zur Bilddatenerfassung
- gegebenfalls ein VGA-Monitor zum Anzeigen der aktuellen Bilddaten der Bildverarbeitungspipeline
- Pmod-Bluetooth Modul des Unternehmens Digilent zur drahtlosen Kommunikation mit dem Kontroll-PC

## 2. Übersicht zum SOC-Kamera-System

---

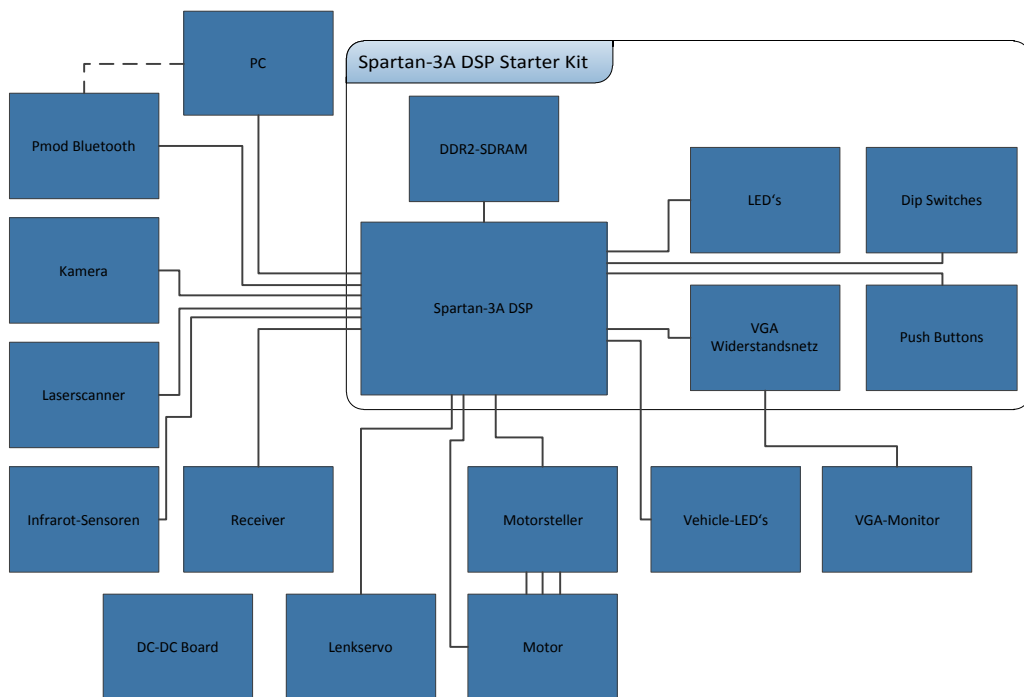


Abbildung 2.2.: Übersicht über die Hardwarekomponenten des Systems

- Verstärker-Board zum Ansteuern der Beleuchtungs-LED's des Fahrzeuges
- Motorsteller zum Ansteuern des bürstenlosen Antriebsmotors
- bürstenloser Antriebsmotor mit Hall-Sensoren zum Ermitteln der aktuellen Drehzahl und somit der Geschwindigkeit durch den FPGA
- Lenkservo
- Modellbauempfänger zum ferngesteuerten Eingriff während der Fahrt
- Infrarotsensoren zur seitlichen Abstandsmessung
- Laserscanner zur Hinderniserkennung in Fahrtrichtung
- DC-DC-Wandler Board zum Bereitstellen einer stabilen Betriebsspannungen für die einzelnen Komponenten aus mitgeführten Akkumulatoren

Die einzelnen Hardwarekomponenten sind in [Abbildung 2.2](#) dargestellt.

Die vom Unternehmen Videology hergestellte 24B7525A-Kamera besteht aus einem MT9V022-Bildsensor von Aptina, aus diversen Bausteinen zur geregelten Spannungsversorgung des Bildsensors, einem Digital-Analog-Wandler zum Bereitstellen eines analogen Bildsignals sowie einem 27 MHz Oszillator zum Erzeugen der Systemfrequenz für den Bildsensor. [25, S. 4] Da die Signale der Kamera lediglich über einen Molex-Stecker mit 30 Kontakten auf eine Größe von circa 3 mm x 6 mm herausgeleitet werden, wurde eine Adapterplatine zum Herausführen der Signale auf eine Pfostenleiste verwendet.

### 2.2. Dual-MicroBlaze System

Zur Konfiguration und Überwachung der einzelnen Komponenten sind auf dem FPGA zwei MicroBlaze-Prozessoren instantiiert und an das System angeschlossen. An den PLB (Processor Local Bus) des ersten Prozessors ist ein Großteil der Schnittstellen angebunden, da der zweite Prozessor lediglich für das Auslesen und Verarbeiten der Daten des Laserscanners verwendet werden soll. Diese Schnittstellen enthalten das I<sup>2</sup>C-Bus Interface zur Konfiguration der Kamera, diverse GPIO-Schnittstellen beispielsweise zum Ansteuern der Fahrzeugbeleuchtung und zwei UART-Schnittstellen zur Kommunikation mit dem angeschlossenen PC. An eine dieser Schnittstellen ist das Pmod Bluetooth-Modul angeschlossen. Die Konfigurations- und Überwachungsschnittstelle der Bildverarbeitungspipeline wird über den SAV-Connector zur Verfügung gestellt. Jeder der Prozessoren verfügt über einen eigenen 32 kByte großen Block-RAM und einen Anbindung an den 128 MByte großen DDR2-SDRAM. Ein MPMC (Multi-Port Memory Controller [27]) übernimmt das Ansteuern des DDR2-SDRAM. Die Speicher sind jeweils mit zwei Bussen an die Prozessoren angebunden, einer für Daten und einer für Instruktionen. Dies entspricht der Harvard-Architektur. Da es sich bei dem DDR2-SDRAM um keinen Speicher mit doppelten Zugriffsschnittstellen handelt, wird der Zugriff bei jedem Prozessor jeweils durch 8 kByte Cache am Daten- und Instruktionsbus beschleunigt. Grundsätzlich handelt es sich bei der Einbindung des DDR2-SDRAM allerdings um keine vollwertige Harvard-Architektur, da keine getrennte Anbindung des Daten- und Instruktionsspeichers vorhanden ist. Zur Synchronisation des Zugriffs auf gemeinsame Speicherbereiche im DDR2-SDRAM verfügen die beiden Prozessoren über eine Anbindung an einen Hardware-Mutex. Hinzu kommt ein FSL zwischen den beiden Prozessoren zum schnellen Datenaustausch. Zusätzlich ist an den ersten Prozessor zur Erfassung von Bild-daten im laufenden Betrieb der FrameGrabber über einen FSL angeschlossen. Um Programme einzuspielen und den Programmfluss zur Laufzeit zu beeinflussen und somit ein Debugging zur Verfügung zu stellen, wurde an beide Prozessoren ein MDM-Modul (MicroBlaze Debug Module [28]) angeschlossen. Da auf beiden Prozessoren der von der Firma Xilinx entwickelte XilKernel

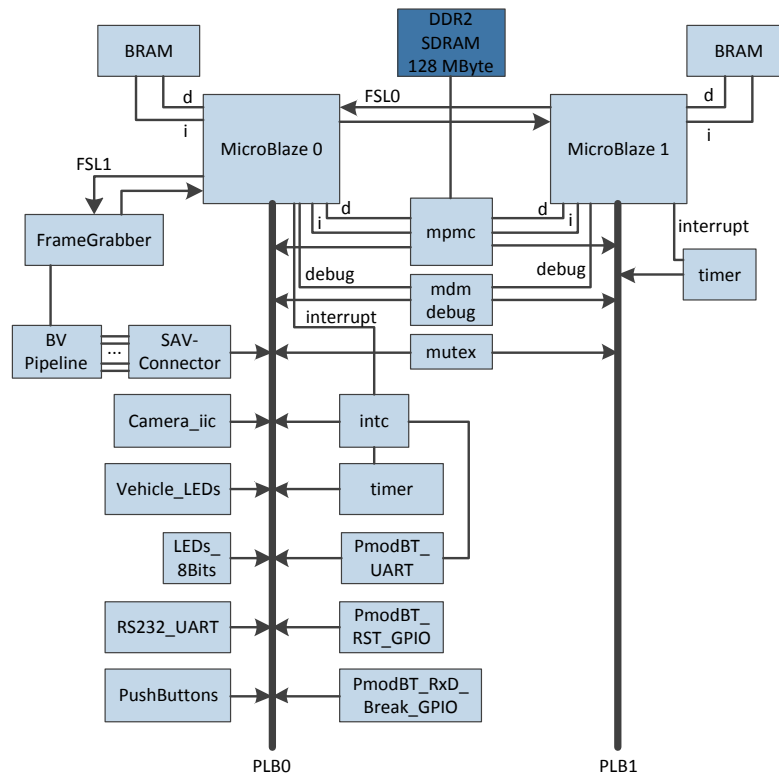


Abbildung 2.3.: Dual-MicroBlaze System mit angebotenen Komponenten

zum Einsatz kommt und dieser einen Timer benötigt, wurde an beide Prozessoren ein Timer angeschlossen. Um das Verarbeiten mehrerer Interruptquellen durch den ersten Prozessor zu realisieren, wurde ein Interruptcontroller integriert. Abbildung 2.3 gibt eine Übersicht über das aktuelle Dual-MicroBlaze System.

### 2.3. Bildverarbeitungspipeline

Die Verarbeitung des Bilddatenstroms und die Extraktion eines Lenkwinkels übernimmt die Bildverarbeitungspipeline. Sie wird mit dem Pixeltakt des MT9V022 Bildsensors betrieben. In einigen Bereichen werden Vielfache und Bruchteile des Pixeltaktes benötigt. Diese Taktraten werden durch einen Clock-Manager zur Verfügung gestellt. Der Bildverarbeitungsstrang ist in Abbildung 2.4 schematisch dargestellt. Die von der Kamera kommenden Daten werden im Kamera-Interface durch Registerketten synchronisiert und an die Bildverarbeitungskette weitergegeben.

In der ersten Stufe der Bildverarbeitung wird eine Binarisierung, eine Erosion sowie eine projek-

## 2. Übersicht zum SOC-Kamera-System

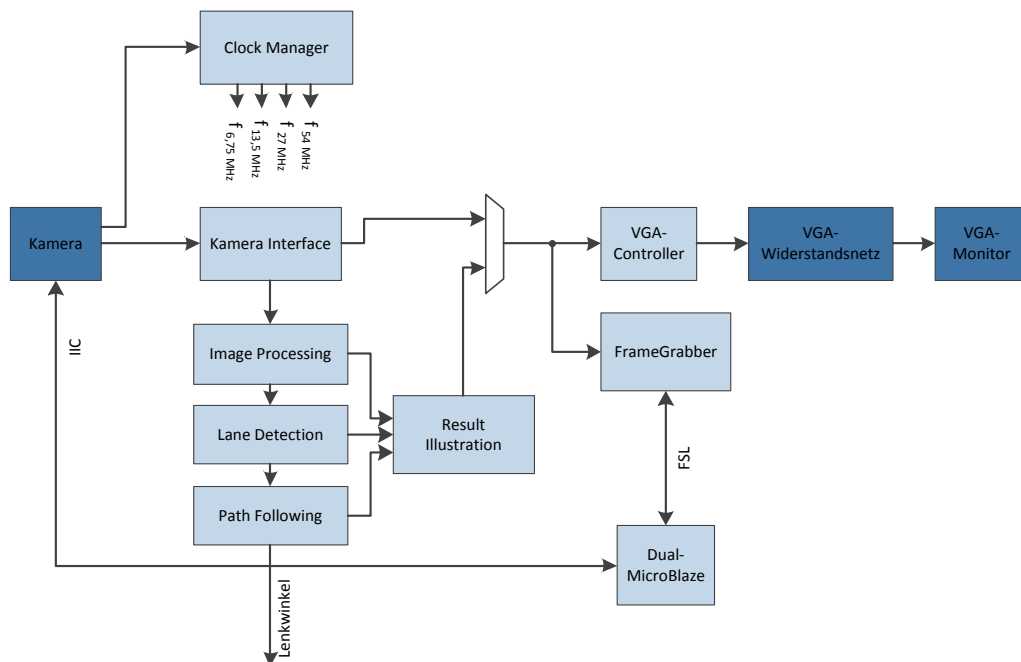


Abbildung 2.4.: Übersicht über die Bildverarbeitungspipeline

tive Transformation der Bilddaten vorgenommen. In Abbildung 2.4 ist sie mit Image Processing beschriftet. Die zweite Stufe der Bildverarbeitung übernimmt die Extraktion der Fahrbahnmarkierung durch eine in einer Region of Interest (ROI) durchgeführten Hugh-Transformation. Diese Stufe ist in Abbildung 2.4 mit Lane Detection beschriftet. In der letzten Stufe wird ein Look Ahead Point (LAP) und daraus der Lenkwinkel berechnet. Der ermittelte Lenkwinkel wird dann über den PWM-Generator an die Aktorik weitergeleitet. Die letzte Stufe ist in Abbildung 2.4 mit Path Following beschriftet.

Um die jeweiligen Ergebnisse der einzelnen Stufen visuell zu überprüfen, existiert ein Result Illustration Block, der die Ergebnisdaten zur Ausgabe auf einem VGA-Monitor vorbereitet.

Über einen Multiplexer können wahlweise direkt die Bilddaten der Kamera oder die der Result Illustration an den VGA-Controller und den FrameGrabber weitergegeben werden. Der VGA-Controller generiert die für den VGA-Standard nötigen HSYNC- und VSYNC-Signale aus den Frame- und Linevalid-Signalen und wandelt die bis jetzt monochromen Bilddaten in entsprechende RGB-Bilddaten um. Ein Widerstandsnetz wandelt die digitalen RGB-Werte entsprechend in die analogen Signale um.

Durch den FrameGrabber kann das aktuell durch den VGA-Controller ausgegebene Bild vom Dual-MicroBlaze System abgespeichert werden. Er sendet nach dem Erhalten eines Startkom-

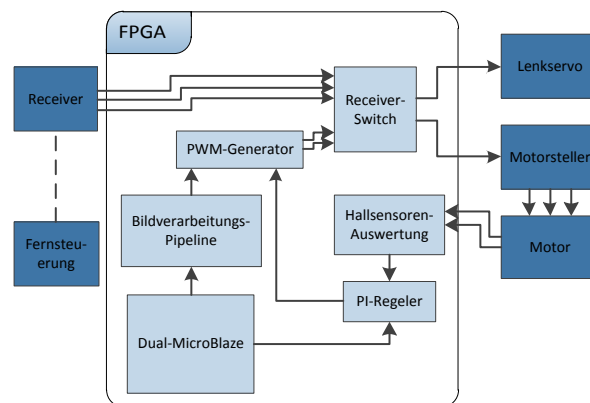


Abbildung 2.5.: Übersicht über die Ansteuerung der Aktorik

mandos über den FSL (Fast Simplex Link) die Bilddaten des nächsten Vollbildes an einen der MicroBlaze-Prozessoren. Dieser legt die Daten im RAM ab und sendet sie gegebenenfalls an den Kontroll-PC weiter.

### 2.4. Regelung der Aktorik

Die Regelung der Aktorik besteht aus einer digitalen Schaltung zur Regelung der Geschwindigkeit sowie einer digitalen Schaltung zum Generieren des bei Modellbaukomponenten üblichen PWM-Steuersignals für Motorsteller und Servos. Diese Schaltungen wurden auf dem FPGA integriert. Zur Erfassung der aktuellen Geschwindigkeit wurde ein bürstenloser Motor mit Hall-Sensoren verwendet. Aus der Vorgabe für die Geschwindigkeit und der aktuell gemessenen Geschwindigkeit generiert ein PI-Regler einen Steuerwert für den Motorsteller. Dieser Wert und der Wert für den Lenkwinkel werden dann von der PWM-Schaltung in ein im Modellbau übliches PWM-Steuersignal umgewandelt.

Um einen sicheren Betrieb des Fahrzeuges zu gewährleisten, wurde ein zusätzlicher Steuerkanal der Funkfernsteuerung genutzt. So kann die Steuerung des Fahrzeuges über einen Schalter an das System übergeben werden. Hierfür verfügt das System über einen im Modellbau üblichen Empfänger für den Empfang des Funksignals einer 2,4 GHz Fernsteueranlage. Zwei Kanäle werden für die Lenkfunktionen verwendet, einer zum Aktivieren des autonomen Betriebes. Das PWM-Signal für das Schalten des autonomen Betriebes wird vom Receiver-Switch ausgewertet. Entsprechend werden die Steuersignale des Empfängers oder der Fahrzeugregelung an die Aktorik weitergeleitet. In [Abbildung 2.5](#) ist dies dargestellt.



### 3. CMOS-Bildsensoren

CMOS-Bildsensoren (Complementary Metal Oxide Semiconductor) haben sich mit der breiter werdenden Verfügbarkeit von CMOS-Fertigungstechniken zu einer Alternativtechnologie zu den CCD-Bildsensoren (Charge Coupled Devices) entwickelt. Durch die CMOS-Fertigungstechnik werden auf dem Sensorchip für jeden einzelnen Pixelsensor eine eigene kleine Signalverarbeitungsschaltung integriert. Diese werden dann aufgrund der aktiven Signalaufbereitung direkt am Pixelsensor als Active Pixel bezeichnet. Die Idee, jeden einzelnen Pixel mit einer eigenen Signalverarbeitungsschaltung auszustatten, bestand zwar schon länger, war jedoch aufgrund von fehlenden Technologien noch nicht effizient umsetzbar. [17]

Im Folgenden wird eine grobe Übersicht über den Unterschied von CMOS- zu CCD-Bildsensoren gegeben. In den Abschnitten dieses Kapitels werden die grundlegenden Bestandteile eines Active Pixel CMOS-Bildsensors beschrieben und die in dem verwendeten MT9V022-Bildsensor integrierten Funktionen erläutert. Der letzte Abschnitt behandelt die herausgeführten Parametrisierungsschnittstellen des MT9V022-Bildsensors.

Ein wesentlicher Unterschied zwischen diesen beiden Sensorarten ist, dass die Pixel bei CMOS-Sensoren im Gegensatz zu den CCD-Sensoren nicht durch Ladungsverschiebung auf der Chipoberfläche ausgelesen werden. Stattdessen verfügt jeder Pixel eines CMOS-Bildsensors über eine Schaltung, durch die das gemessene Bildsignal auf eine einer bestimmten Anzahl von Pixeln gemeinsame Busleitung ausgegeben wird. Diese Schaltung wird durch ein digitales Enable gesteuert, das durch eine Art Adressdecoder, beziehungsweise eine Selektionsschaltung, erzeugt wird. Hierdurch erfolgt bei entsprechender Architektur des Adressdecoders ein beliebiger Zugriff auf die Pixel. Dies steht im Gegensatz zu den CCD-Bildsensoren. Darüber hinaus lassen sich auch nur bestimmten Bereiche des Sensors auslesen. Auch dies ist mit CCD-Bildsensoren architekturbedingt nicht realisierbar. Der grobe Aufbau der beiden Sensorarten sowie der Signalweg beim Auslesen eines Pixels sind in Abbildung 3.1 dargestellt. [17]

Ein weiterer Vorteil der CMOS-Bildsensoren ist, dass kein Blooming auftritt, da dieser Effekt aus der Ladungsverschiebung auf der Chipoberfläche der CCD-Bildsensoren resultiert.

Auch bei der Produktion unterscheiden sich die Bildsensoren. Während bei CCD-Bildsensoren zwei Chips, einer für den Bildsensor und einer für dessen Ansteuerung, miteinander verbunden

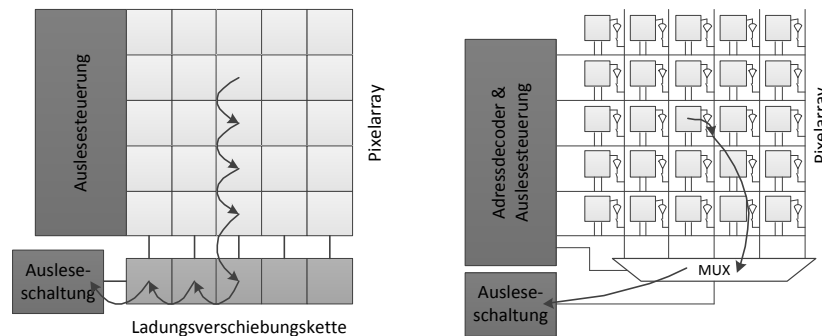


Abbildung 3.1.: grober Aufbau CCD-Bildsensor (links) und CMOS-Bildsensor (rechts); die Pfeile markieren den Signalweg beim Auslesen

werden, bestehen CMOS-Bildsensoren aus lediglich einem Chip. Auf diesem sind analoge und digitale Signalverarbeitungs- und Steuerungskomponenten, Analog-Digital-Wandler, Dateninterfaces sowie nahezu jede heute in CMOS implementierbare Schaltung zu verwirklicht. Dadurch ergeben sich teilweise komplexe System-on-Chip Architekturen. [17]

Weiterhin lassen CMOS-Bildsensoren höhere Bildwiederholungsraten und eine flexiblere Anpassung dieser durch die Reduktion des ausgelesenen Bereiches zu. Auch dies lässt die Architektur der CCD-Bildsensoren nicht zu. [17]

Allerdings haben CMOS-Bildsensoren gegenüber CCD-Bildsensoren nicht nur Vorteile. So wirkt sich die durch das Implementieren von Schaltungen in den bildaktiven Bereich des Sensors verkleinerte Fläche zum Messen von Photonen negativ auf die Lichtempfindlichkeit aus. Dem kann durch den Einsatz von Mikrolinsen entgegengewirkt werden. Dies verursacht jedoch einen höheren Produktionsaufwand. Aus den Schaltungen direkt im Umfeld der Pixelsensoren ergibt sich auch eine weitere Störgröße, weshalb das Bild von CMOS-Sensoren allgemein von schlechterer Qualität ist. Zu dieser Verschlechterung der Bildqualität tragen auch Fertigungstoleranzen bei, die in den Signalverarbeitungsschaltungen der einzelnen aktiven Pixel geringe Unterschiede in deren elektrischen Eigenschaften hervorrufen. [17]

Um diesen Nachteil von CMOS-Bildsensoren auszugleichen, verfügen diese über Schaltungen zur Korrektur des Schwarzwertes, zur Verbesserung des Signal-Rausch-Verhältnisses sowie zum Unterdrücken von Pixelrauschen.

### 3.1. Aufbau eines Active Pixel CMOS-Bildsensors

Die Erfassung digitaler Bilddaten geschieht durch richtungsabhängiges Messen eines eintreffenden Photonenstromes. Dabei werden die aus den unterschiedlichen Eingangsstrahlen eintreffenden Photonen durch optische Komponenten auf eine Detektorenfläche mit zweidimensional angeordneten Sensorelementen, den Bildsensor, gelenkt. Die einzelnen Sensorelemente werden als Pixel bezeichnet. Die Konvertierung des Photonenstroms in ein elektrisches Signal übernehmen dabei Photodioden. Sie verschieben abhängig von der Anzahl der eintreffenden Photonen eine bestimmte Ladungsmenge. [7, S. 190 ff.]

Dabei gibt es zwei Arten der Signalerfassung: Zum einen das Laden eines Kondensators durch den Betrieb der Diode im Photoelementbetrieb, zum anderen das Entladen eines mit einer definierten Spannung vorgeladenen Kondensators über die Diode in Sperrrichtung im Photodetektorenbetrieb. [7, S. 209 ff.]

Dabei wird der benötigte Kondensator zum Teil für einen MOSFET (Metal Oxide Semiconductor Field Effect Transistor) verwendet. Der Sourceanschluss dieses MOSFETs ist an die Versorgungsspannung angeschlossen. Über den MOSFET kann die von Source nach Drain abfallende Spannung entsprechend ausgewertet werden. Dieser Vorgang ist nicht destruktiv, der Sensor kann mehrmals Ausgelesen werden. Durch den Transistor kann je nach Dimensionierung auch eine Vorverstärkung des Signals geschehen. Ein weiterer Feldeffekttransistor sorgt für den Reset des Kondensators in einen definierten Zustand, ein Letzter das wahlweise Anlegen des gemessenen Signals an eine mehreren Pixeln gemeinsame Ausleseleitung mit angeschlossener Ausleseelektronik. Bei dieser Schaltung handelt es sich um eine einfache Version eines aktiven Pixels (Active Pixel), er wird als Source Follower per Detector Readout (SFD) bezeichnet. Sie ist in Abbildung 3.2 bei der Markierung Active-Pixel dargestellt. [22, S. 311 ff.] Das Prinzip anderer Active-Pixel-Schaltungen beinhaltet dabei immer mindestens diese Komponenten. Variiert wird beispielsweise die Vorverstärkerschaltung.

Im Wesentlichen verfügt jede Active-Pixel-Schaltung über einen Reset-Eingang (Reset) zum Zurücksetzen der gesammelten Signalinformationen, einen Enable-Eingang (Enable) zum Anlegen des gemessenen Signals am Ausgang sowie einen Ausgang (Bus) mit einem Vorverstärker. Die Bildinformationen werden in den Pixeln über einen bestimmten Zeitraum gesammelt (Integration oder auch Belichtung). [22, S. 311 ff.]

Begrenzt wird die Menge der maximal integrierbaren Bildinformation durch die Größe des Kondensators im Pixel. Ist der Kondensator entladen, so können weitere eintreffende Photonen das Bildsignal nicht mehr beeinflussen. Es befindet sich im Bereich der Sättigung. Daraus ergibt sich eine maximal messbare Größe  $Q_{sat}$ . Die minimal messbare Größe ergibt sich aus dem

maximalen Störsignal des Bildsensors  $NEI$ . Aus dem Verhältnis dieser beiden Größen ergibt sich der sogenannte dynamische Messbereich des Bildsensors (Dynamic Range; DR) [22, S. 337]:

$$DR = \frac{Q_{sat}}{NEI} \quad (3.1)$$

Dabei ist zu beachten, dass am gesamten Bildsensor nur eine Belichtungszeit verwendet werden kann. Hierbei kann es vorkommen, dass in einem Bild die minimal und maximal zu messenden Helligkeitswerte so weit auseinander liegen, dass sie nicht mehr durch den dynamischen Bereich des Bildsensors abbildbar sind. In diesem Fall befinden sich einige Pixel im Sättigungsbereich, andere erfassen dagegen ein kaum messbares und von Störungen überlagertes Signal. [22, S. 337] Zu diesem Zweck ist es unter bestimmten Umständen nötig, den dynamischen Bereich des Bildsensors zu erweitern (High Dynamic Range; HDR). Dies kann unter anderem durch eine spezielle Beschaltung der Reset-Schaltung realisiert werden. Dabei wird die Resetspannung des Reset-Transistors während des Resets nicht mehr konstant gehalten, sondern variiert. So wird zum initialen Reset die volle Spannung angelegt. Danach wird die Spannung aber nicht komplett abgeschaltet, sondern verbleibt auf einem Pegel, der etwas geringer als das initiale Resetsignal ist. Die Spannung am Reset-Eingang wird stufenweise immer weiter verringert, bis sie kurz vor dem Ende der Belichtungszeit den minimalen Wert erreicht. Durch dieses Vorgehen erfolgt die Sättigung bei maximaler Belichtung durch die am Reset-Signal eingestellten Werten schrittweise früher. [10, 13]

Dieser Vorgang ist in Abbildung 3.3 mit unterschiedlichen Photonenflussdichten dargestellt. Bei maximalem Photonenfluss wird der Sensor bis zur Sättigung belichtet, während bei moderatem Photonenfluss durch die vorzeitige Sättigung am Ende der Belichtung keine vollständige Sättigung eintritt. Für den minimalen Photonenfluss steht dabei eine lange Belichtungszeit zur Verfügung. Dabei ergibt sich der in Abbildung 3.4 dargestellte Zusammenhang zwischen Photonenfluss und Ausgangssignal. Zu erkennen sind mehrere lineare Abschnitte, die aus den unterschiedlichen Spannungspegeln am Reset-Signal und deren zeitlicher Länge resultieren. [13, S. 29]

Die Bilddaten werden mit drei Komponenten aus den einzelnen Active-Pixel Schaltungen ausgelesen. Diese Komponenten sind in dem in Abbildung 3.1 abgebildeten CMOS-Bildsensor dargestellt.

Zum einen eine Kodierungsschaltung, die die Ausgabe der entsprechenden Pixel durch deren Enables freigibt. Diese kann als Adressdecoder wie beispielsweise bei RAM realisiert sein, aber auch als einfache Schieberegisterkette (siehe Abbildung 3.2 Auslesesteuerung). [22, S. 287 ff.]

Weiterhin eine Ausleseschaltung, die die Signale der einzelnen Pixel aufbereitet sowie je nach Bedarf einen Analog-Digital-Wandler beinhaltet. In diesen Schaltungen kann auch eine einstellbare

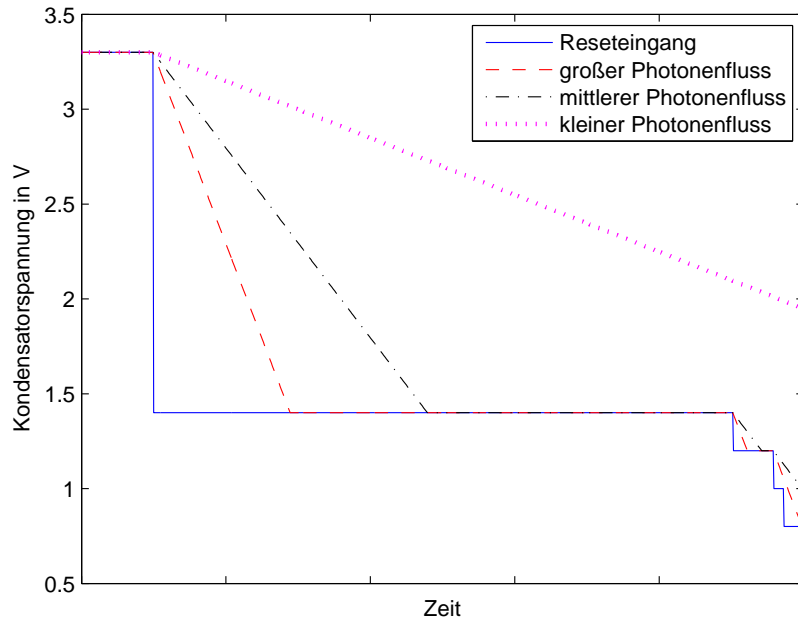


Abbildung 3.3.: Verlauf des Reset-Signals und der integrierten Spannung bei unterschiedlicher Photonendichte im HDR-Modus

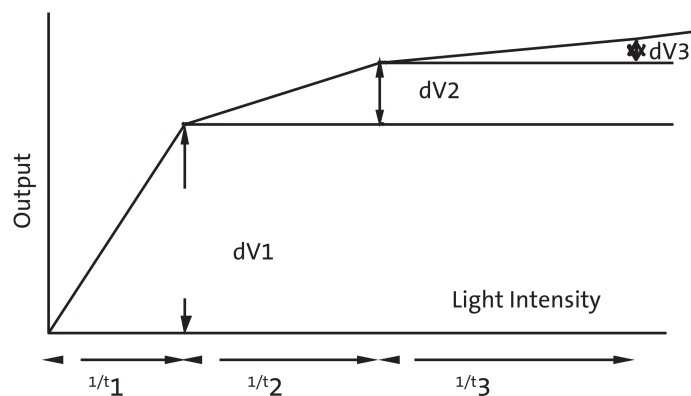


Abbildung 3.4.: Ausgangssignal im Verhältnis zum Photonfluss im HDR-Modus des MT9V022-Bildsensors; es ergeben sich mehrere lineare Abschnitte, abhängig von den Zeiten  $t_n$  und den Spannungsstufen  $dV_n$  [4, S. 26]

Verstärkung des Signales realisiert sein. Die Ausleseschaltung kann zum gleichzeitigen Auslesen mehrerer Pixel mehrfach ausgeführt werden (siehe Abbildung 3.2 Auslesschaltung). [22, S. 287 ff.]

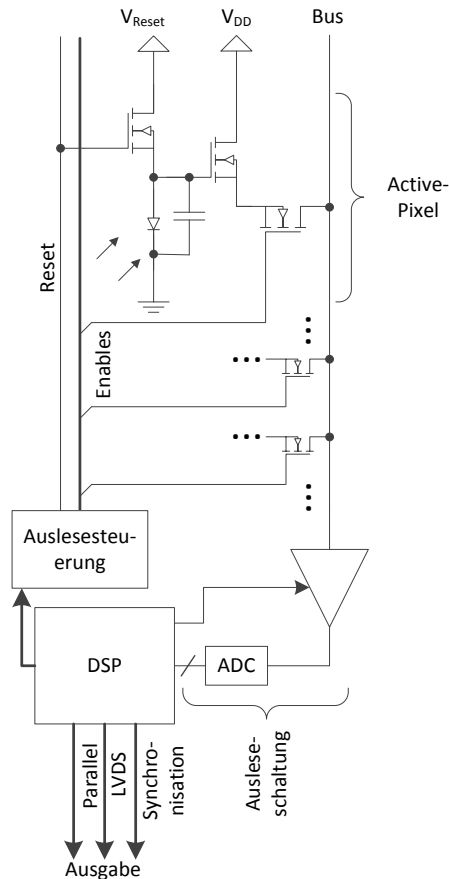


Abbildung 3.2.: Active-Pixel mit Leitungsnetz zum Auslesen (Bus) und Ansteuern (Enables, Reset), Ausleseschaltung, Auslesesteuerung sowie DSP zur Signalaufbereitung und Generierung der Ausgabe-signale [22, S. 287, 22, S. 312, 17]

Um die von den Pixeln gemessenen Signale zu der Ausleseelektronik zu transportieren wird überdies ein Leitungsnetz verwendet, das die einzelnen Pixel einer oder mehreren Gruppen mit der Ausleseelektronik verbindet (in Abbildung 3.2 mit Bus markiert). [22, S. 287 ff.]

Zur korrekten Ansteuerung der Active-Pixel-Sensoren ist eine Schaltung zum Generieren der Reset- sowie weiterer Signale vorhanden. Zusätzlich werden auch Schaltungen zur Korrektur der Fertigungsungenauigkeiten zwischen den einzelnen Pixeln integriert (in Abbildung 3.2 in Ausleseelektronik enthalten).

Zum Steuern dieser Schaltungen sowie zum Steuern anderer Parameter wird auf dem Chip des Bildsensors bei Bedarf ein Digitaler Signalprozessor (DSP) integrieren. Dieser kann auch das Bereitstellen einer Konfigurationsschnittstelle übernehmen. Zusätzlich zu diesen Funktionen wird auf dem Chip des Bildsensors meist eine Einheit zur Generierung des Synchronisationssignals sowie bei Bedarf eine serielle Datenschnittstelle wie zum Beispiel LVDS bereitgestellt. [4]

### 3.2. Integrierte Funktionen des MT9V022-Bildsensors

Der MT9V022-Bildsensor lässt sich in folgende Funktionsblöcke unterteilen (siehe Abbildung 3.5):

- Active-Pixel Array zum Erfassen der Bilddaten
- analoge Ausleseschaltung zum Auslesen und Aufbereiten der Bildinformationen
- 10 Bit Analog-Digital-Wandler zur Digitalisierung der Analogen Signale

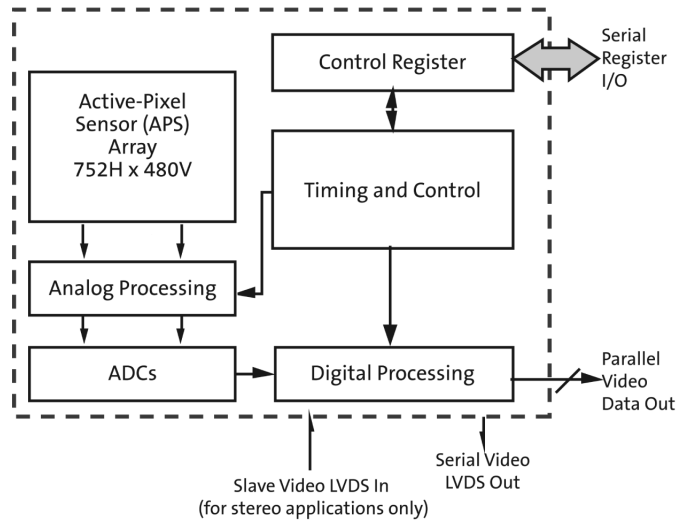


Abbildung 3.5.: MT9V022-Bildsensoren; Active-Pixel Array zur Bilddatenerfassung, analoge Ausleseschaltung (Analog Processing), 10 Bit Analog-Digital-Wandler (ADCs), Auslesesteuerung (Timing and Control) zum Ansteuern des Sensors, DSP (Digital Processing) zum Aufbereiten des Bildsignals und zum Generieren der Ausgabe sowie I<sup>2</sup>C-Businterface mit Konfigurationsregistern (Control Register) zur Konfiguration [4, S. 6]

- Auslesesteuerung zum Generieren der Enables und des Reset-Signals, zur Steuerung der analogen Ausleseschaltung; beinhaltet die Schaltungen zur automatischen Korrektur der Bildinformationen
- DSP zum Aufbereiten des Bildsignals sowie zum Erzeugen der Synchronisationssignale und des LVDS-Datenstromes
- Registerschnittstelle zum Konfigurieren des Bildsensors über ein I<sup>2</sup>C-Businterface

Um den MT9V022-Bildsensor in ein System einzubinden, werden die Bilddaten sowohl über ein paralleles als auch ein serielles Interface nach LVDS-Standard bereitgestellt. Zusätzlich ist in der 24B7525A-Kamera ein Digital-Analog-Wandler zum Erzeugen eines analogen Bildsignals vorhanden [25]. Das parallele Interface besteht aus 10 Datenleitungen und 2 Leitungen für die Synchronisationssignale (Line- und Framevalid). Das serielle LVDS-Interface ist in Form zweier differentieller Leitungen herausgeführt. Der LVDS-Datenstrom beinhaltet die 10 Datenbits sowie die Synchronisationssignale. Zum das Eintakten der Daten durch den Empfänger wird der Pixeltakt auf einer eigenen Leitung herausgegeben. Ein Hardware-Reset des Bildsensors lässt sich über einen Reset-Eingang durchführen. Der Bildsensor verfügt über einen Energiesparmodus, der über einen Eingang gesteuert werden kann. Dieser wird jedoch nicht verwendet, entspre-

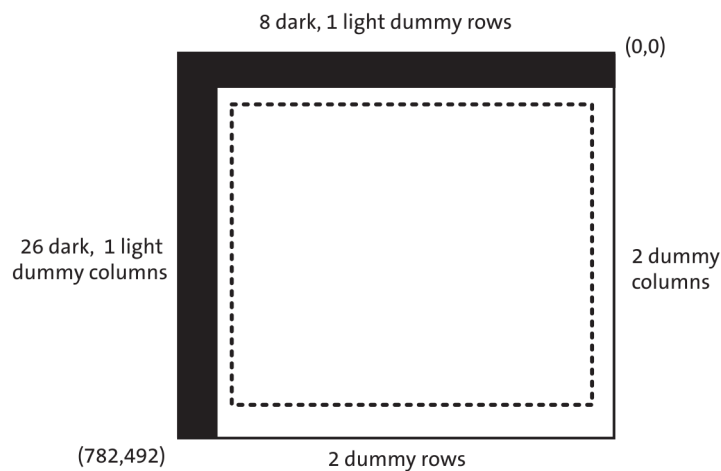


Abbildung 3.6.: Active-Pixel Sensor Array des MT9V022 Bildsensors mit optisch geschwärtzten und ungenutzten Pixeln [4, S. 10]

chend wird auch nicht weiter auf ihn eingegangen. Zur Konfiguration der Parameter wurde ein Kontrollregister-Funktionsblock implementiert, der die Steuergrößen für den Sensor beinhaltet und diese über ein I<sup>2</sup>C-Bus Interface konfigurierbar macht. [4]

Zum Erfassen der Bilddaten verfügt der MT9V022-Bildsensor über einen Active-Pixel Sensoren Array. Insgesamt hat der Sensoren Array 782 x 492 Pixel, wovon allerdings am linken Rand 26 Spalten und am oberem Rand 8 Zeilen geschwärtzt sind. Diese geschwärtzten Pixel werden zur Ermittlung des Kalibrierwertes zum Schwarzabgleich verwendet. Weiterhin ist die aktiv genutzte Sensorfläche am oberen und linken Rand von einer Reihe und am unteren und rechten Rand von zwei Reihen nicht genutzter Dummy-Pixeln umgeben. Durch die Verwendung von Dummy-Pixeln ist jeder aktiv genutzte Pixel gleichmäßig von anderen Pixeln umgeben, wodurch am Rand auftretende Unregelmäßigkeiten vermieden werden. [4, S. 10] Diese Anordnung der Pixel ist in Abbildung 3.6 dargestellt.

Um die Bilddaten aus dem Pixel-Array zu verarbeiten, verfügt der MT9V022 Bildsensor über eine analoge Ausleseschaltung (siehe Abbildung 3.11). Durch diese wird dem Signal ein zuvor ermittelter Korrekturwert zum Schwarzabgleich hinzugefügt (Offset Correction Voltage) sowie einen Verstärkungsfaktor für das Signal eingestellt (Gain Selection).

Hinzu kommt ein 10 Bit Analog-Digital-Umsetzer (ADC), um das Signal zu digitalisieren. [4, S. 23] Dieser kann auch auf einen nicht linearen Modus gestellt werden, in dem mit 12 Bit quantisierte Daten auf 10 Bit abgebildet werden. Bei geringer Lichtintensität sind die Quantisierungsschritte kleiner, wodurch es zu geringeren Quantisierungsfehlern kommt. Im Bereich der hohen Lichtintensität sind die Quantisierungsschritte größer, da hier der Quantisierungsfehler im



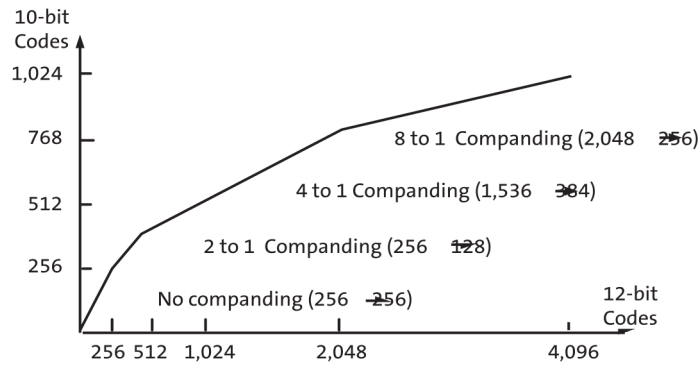


Abbildung 3.7.: Nichtlinearer Verlauf der 12- auf 10-Bit Umsetzung des ADCs im MT9V022 Bildsensor [4, S. 28]

Vergleich zum Gesamtwert gering ist. [4, 27 f.] Dieser nichtlineare Verlauf ist in Abbildung 3.7 dargestellt.

Der analogen Ausleseschaltung und dem Analog-Digital-Umsetzer schließt sich eine digitale Signalverarbeitungsschaltung an. Sie generiert einerseits die Bildsynchronisationssignale und das LVDS-Signal. Andererseits lässt sich mit ihnen eine Anpassen der Belichtungszeit und des Verstärkungsfaktors an die aktuellen Konditionen vornehmen. Hierzu wird aus den laufenden Bilddaten ein Histogramm und schließlich ein aktueller durchschnittlicher Grauwert berechnet. Anhand dessen und dem vorgegebenen anzustrebenden Grauwert wird eine neue Belichtungszeit (Automatic Exposure Control; AEC) und ein neuer Verstärkungsfaktor (Automatic Gain Control; AGC) berechnet. Die neu berechneten Werte werden dann noch durch einen einfachen und konfigurierbaren Tiefpassfilter gefiltert. Für die beiden Kontrolleinheiten können jeweils Maximalwerte angegeben werden, Minimalwerte sind fest eingestellt. [4, S. 32] AGC und AEC sind in Abbildung 3.8 dargestellt. Auch gegebenenfalls konfiguriertes Pixelbinning, also die gemittelte Ausgabe mehrerer Pixelwerte, wird hier umgesetzt.

Die Steuerung des Belichtungsvorgangs und des Auslesens des Active-Pixel Sensor Arrays übernimmt der Timing and Control-Funktionsblock. Er setzt die durch den Anwender oder durch die automatische Kontrolleinheit vorgegebene Belichtungszeit um und steuert die HDR-Signalerfassung. Weiterhin lässt sich hier auch die Ausleserichtung sowie ein nur partielles Auslesen des Pixel-Arrays konfigurieren.

### 3.3. Parametrisierungsgrößen des MT9V022-Bildsensors

Zur Parametrisierung des MT9V022-Bildsensors sind im Folgenden beschriebene Größen über Register herausgeführt. Diese lassen sich in mehrere Gruppen unterteilen:

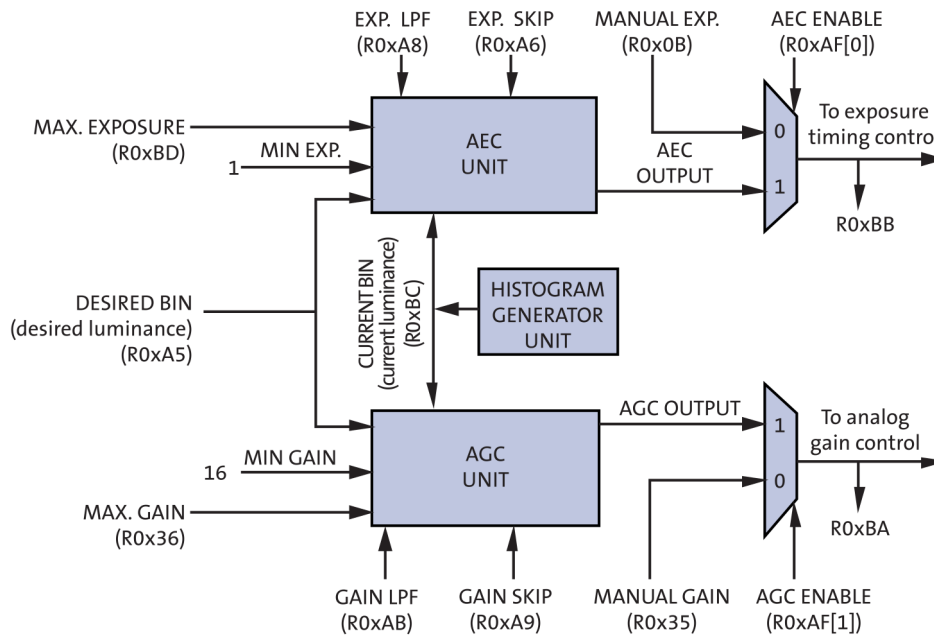


Abbildung 3.8.: AGC- und AEC-Einheiten zur automatischen Verstärkungsfaktor- und Belichtungszeitkontrolle des MT9V022 Bildsensors [4, S. 32]

- Belichtungskontrolle zum automatischen oder manuellen Anpassen der Belichtung an die gegebenen Verhältnisse
- Verstärkungsfaktorkontrolle zum automatischen oder manuellen Anpassen des Verstärkungsfaktors wenn die Belichtung den maximalen Wert erreicht hat
- Schwarzwertkorrektur und Bildrauschunterdrückung zum Beseitigen Effekten durch von Fertigungstoleranzen und Störungen
- Bildauslesesteuerung zum Beeinflussen des Timings, des Bildausschnittes sowie der Art des Auslesens
- Analog-Digital-Umsetzer (Referenzspannung und nichtlineare Quantisierung)
- Higy Dynamic Range zur Erweiterung des dynamischen Bereiches

### 3.3.1. Belichtungskontrolle

Die Belichtungszeit kann einerseits direkt angegeben und andererseits automatisch aus den laufenden Bilddaten ermittelt werden.

Die Angabe der Belichtungszeit erfolgt in der Anzahl der Zeilen, über die die Bildinformationen

integriert werden. Sie ist also von der angegebenen Anzahl der Zeilen  $N_{int}$ , der konfigurierten Anzahl der Pixel pro Zeile  $IMG_{WIDTH}$ , der konfigurierten Anzahl der Pixel zwischen den Zeilen (Horizontal Blanking)  $IMG_{HBLANK}$  sowie vom Systemtakt  $f$  abhängig. Aus diesen Größen ergibt sich die Belichtungszeit  $t_{int}$ :

$$t_{int} = N_{int} * (IMG_{WIDTH} + IMG_{HBLANK}) * \frac{1}{f} \quad (3.2)$$

Dabei ist zu beachten, dass eine Belichtungszeit, die größer als die Zeit eines Einzelbildes ist, die Zeit des Einzelbildes verlängern kann. Hierdurch reduziert sich also bei hoher Belichtungszeit die Bildwiederholungsrate.

Für die automatische Berechnung der Belichtungszeit (Automatic Exposure Control; AEC) wird durch den Bildsensor aus den laufenden Pixeldaten ein Histogramm über eine konfigurierbare Anzahl von Pixeln gebildet. Aus den Histogrammwerten  $h(g)$  für die einzelnen Grauwerte  $g$  wird ein im Mittel  $\mu$  erreichter Pixelwert errechnet, aus dem dann in Kombination mit dem konfigurierbaren angestrebten mittleren Pixelwert eine neue Belichtungszeit berechnet wird. Dieser ermittelte Wert wird auch zur Verstärkungsfaktorsteuerung verwendet. [4]

$$\mu = \frac{\sum_{i=0}^n g_i * h(g_i)}{\sum_{i=0}^n h(g_i)} \quad (3.3)$$

Um ein Schwingen dieses Regelkreises zu verhindern, verfügt der Bildsensor über einen einfach konfigurierbaren Tiefpassfilter sowie eine Verzögerung der Übernahme der neuen Werte. Dieser ist in Abbildung 3.9 dargestellt. [25]

Damit die Bildrate konstant gehalten werden kann und nicht durch die automatische Belichtungszeitkontrolle beeinflusst wird, lässt sich ein maximaler Belichtungszeitwert festlegen. Werden die Grenzwerte für die Belichtungssteuerung erreicht, so wird der angestrebte mittlere Grauwert mit Hilfe des Verstärkungsfaktors angepasst.

Diese Komponente ist in Abbildung 3.8 dargestellt.

#### 3.3.2. Verstärkungsfaktorkontrolle

Die gesamten Bilddaten werden konfigurierbar analog verstärkt. Zusätzlich kann für 25 getrennte Bildbereiche ein digitaler Verstärkungsfaktor getrennt angegeben werden.

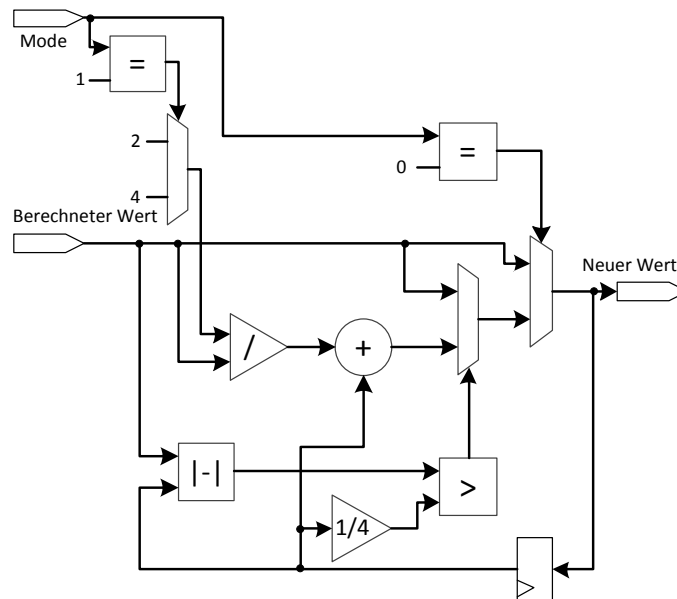


Abbildung 3.9.: Tiefpass der MT9V022 AEC- und AGC-Schaltungen

### Analoger Verstärkungsfaktor

Der analoge Verstärker setzt Verstärkungen um den Faktor 1 bis 4 um. Die Schrittweite des konfigurierbaren Wertes beträgt 0,0625.

Ähnlich wie bei der Belichtungszeit kann die analoge Verstärkung automatisch geregelt (Automatic Gain Control AGC) oder manuell konfiguriert werden. Dabei wird der durch die Gleichung 3.3 ermittelte aktuelle Mittelwert sowie der angestrebte Mittelwert verwendet. Kann der aktuelle Mittelwert durch die Belichtungszeitkontrolle nicht mehr ausreichend dem angestrebten Mittelwert angenähert werden, so wird dies durch den Verstärkungsfaktor realisiert. Um auch bei diesem Regelkreis ein Schwingen zu verhindern, verfügt der Bildsensor über einen einfachen konfigurierbaren Tiefpassfilter sowie eine Verzögerung der Übernahme der neuen Werte. Dies entspricht dem in Abbildung 3.9 dargestellten Schema.

Des Weiteren kann auch hier ein maximaler Wert des Verstärkungsfaktors angegeben werden.

### Digitaler Verstärkungswert

Zusätzlich zur analogen Verstärkung kann durch den integrierten DSP eine digitale Verstärkung vorgenommen werden. Diese kann für 25 Bereiche angegeben werden. Die Bereiche können durch 6 vertikale und 6 horizontale Geraden definiert werden. Beispielhaft ist dies in Abbildung

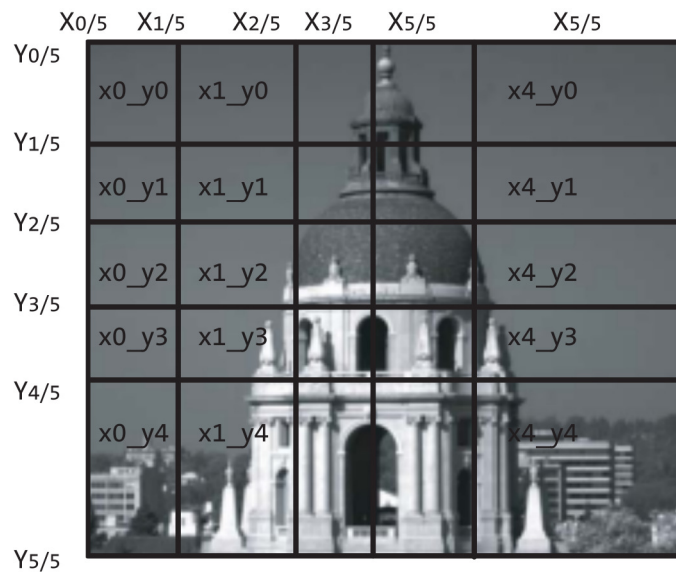


Abbildung 3.10.: Aufteilung des MT9V022-Bildsensors in 25 Bereiche [4, S. 29]

3.10 dargestellt. Für jeden Bereich kann eine individuelle Verstärkung um den Faktor 0 bis 3,75 konfiguriert werden. Die minimale Schrittweite des Faktors beträgt 0,25.

Weiterhin kann den einzelnen Bereichen eine Gewichtung für Einflussnahme auf die AGC- und AEC-Algorithmen zugewiesen werden.

### 3.3.3. Schwarzwertkorrektur

Zum Abgleichen des Schwarzwertes verfügt der MT9V022-Bildsensor über mehrere optisch geschwärzte Referenzpixel (siehe Abbildung 3.6). Zwei Zeilen am oberem Rand werden zur Ermittlung des aktuellen Korrekturwertes verwendet. Die gemessenen Werte dieser Pixel werden gemittelt und um zufällige Störgrößen zu eliminieren, über eine konfigurierbare Anzahl von Bildern durch eine Mittelung gefiltert. Dies entspricht einem FIR-Filter Tiefpass (Finite Impulse Respond). Weicht dieser gefilterte Wert von einem konfigurierbaren minimalen beziehungsweise maximalen Schwellwert ab, so wird die Schwarzwertkorrekturschaltung um einen konfigurierbaren Wert erhöht beziehungsweise verringert. Dies wird durch die in Abbildung 3.11 dargestellte Schaltung mit der Offset Correction Voltage umgesetzt. Die Spannung für die Schwarzwertkorrektur kann bei Bedarf auch manuell konfiguriert werden. Der Wertebereich erstreckt sich von -64 mV bis 63,5 mV in 0,5 mV Schritten.

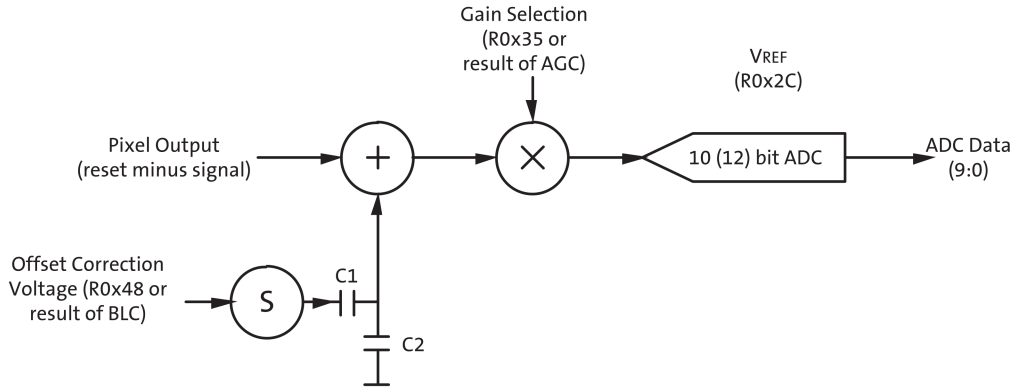


Abbildung 3.11.: Schaltung zur Schwarzwertkorrektur an der analogen Ausleseschaltung des MT9V022-Bildsensors [4, S. 30]

### 3.3.4. Bildrauschunterdrückung

Um zeilenweise Störgrößen zu eliminieren, kann der digitalisierten Wert jeder Zeile um einen individuell ermittelten Wert korrigiert werden. Zu diesem Zweck werden die optisch geschwärzten Pixel am linken Rand des Pixel-Arrays verwendet (siehe Abbildung 3.6). Die Werte dieser Pixel werden gemittelt ( $Pix_{Avg}$ ), wobei die minimalen und maximalen Werte nicht verwendet werden, und die jeweiligen Pixelwerte ( $ADC_{Val}$ ) werden wie folgt korrigiert:

$$Pix_{val} = ADC_{Val} - Pix_{Avg} + Pix_{Const} \quad (3.4)$$

Die Größe  $Pix_{Const}$  kann global für alle Pixel konfiguriert werden. Zusätzlich kann die Anzahl der verwendeten Pixel  $Pix_{Anz}$  im Bereich von 2 bis 18 konfiguriert werden. Außerdem lässt sich die Zeile des ersten zu verwendenden Pixels  $Pix_{Col}$  definieren. Dabei muss beachtet werden, dass optisch geschwärzte Pixel nur in den Zeilen 759 bis 776 vorhanden sind. Um den Zugriff auf nicht vorhandene beziehungsweise optisch nicht geschwärzte Pixel zu vermeiden, muss gelten:

$$758 < Pix_{Col} + Pix_{Anz} < 777 \quad (3.5)$$

Die Bildrauschunterdrückung kann bei Bedarf deaktiviert werden. Weiterhin ist das Verzögern des errechneten Korrekturwertes um ein Bild konfigurierbar. [4]

### 3.3.5. Bildauslesesteuerung

Die Bildausgabe kann in drei Parametergruppen beeinflusst werden:

- Bildausschnitt: Bereich der ausgegebenen Pixel, Ausleserichtung
- Auslesemethode: Intelaced-Modus, mehrere Pixel gemittelt auslesen
- Timing: Größe der nichtgenutzten Datenbereiche zwischen den Zeilen und Bildern

Diese werden im Folgenden beschrieben.

#### **Bildausschnitt**

Zur Ausgabe wird ein rechteckiger Bildausschnitt definiert werden. Für diesen kann eine beliebige Größe und Position angegeben werden. Dieser Ausschnitt muss sich lediglich im aktiven Sensorbereich befinden. Um das Auslesen des Bildausschnittes gegebenenfalls an vorhandene Gegebenheiten anzupassen, kann die Richtung des Auslesens getrennt sowohl in horizontaler als auch in vertikaler Richtung invertiert werden. Der ausgelesene Bildausschnitt lässt sich auf diese Weise spiegeln. Bei Bedarf lässt sich das Ausgeben der optisch geschwärzten Pixel am Rand, beispielsweise zum manuellen Berechnen eines Schwarzwertkorrekturwertes, aktivieren. Die schwarzen Zeilen und Spalten werden dann vor den normalen Bilddaten ausgegeben, unabhängig von der aktuellen Position des Bildausschnittes. [4]

#### **Auslesemethoden**

Um die Datenrate zu reduzieren und die Bildrate zu erhöhen, kann die Auflösung stufenweise in horizontaler beziehungsweise vertikaler Richtung reduziert werden. Hierzu werden jeweils zwei beziehungsweise vier Pixel gemittelt. Jedoch erhöht nur eine Reduzierung der Auflösung in der Vertikalen die effektive Bildrate, während eine Reduzierung in der Horizontalen lediglich die Datenrate reduziert.

Weiterhin kann eine Ausgabe der Bilddaten im Interlaced-Modus aktiviert werden. Dabei wird entweder das abwechselnde Auslesen der beiden Halbbilder oder das wiederholte Auslesen eines der beiden Halbbilder konfiguriert.

Der MT9V022-Bildsensor verfügt über einen simultanen und sequentiellen Auslesemodus. Im sequentiellen Modus wird der Bildsensor belichtet und danach ausgelesen. Ist der simultane Modus konfiguriert, so werden die Pixeldaten in einem Bereich des Bildsensors ausgelesen, während in anderen Bereichen noch oder schon wieder die Belichtung stattfindet. Hierdurch sind bei hohen Bildraten höhere Belichtungszeiten als im sequentiellen Modus umzusetzen. Allerdings kann es bei schnellen Bewegungen im Bild durch den untereinander verschobenen Belichtungszeitraum der einzelnen Zeilen zur Verzerrung der Bildinformationen kommen. [4]

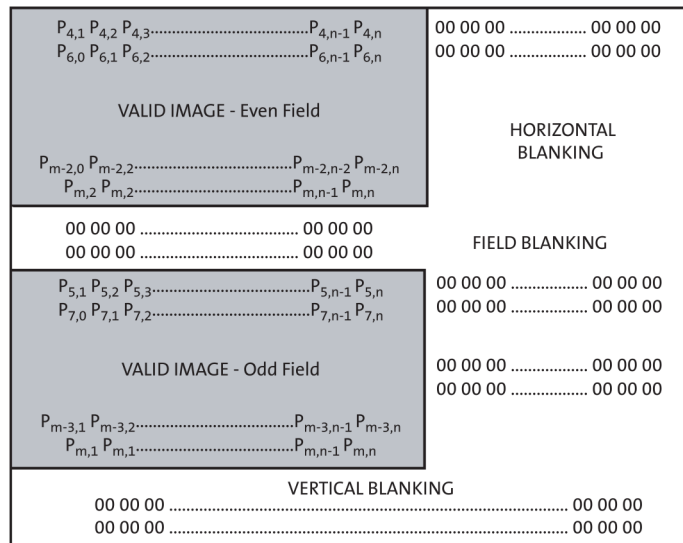


Abbildung 3.12.: zur Ausgabe genutzter (grau), horizontaler und vertikaler informationsfreier Bereich (weiß) bei der Bildausgabe des MT9V022-Bildsensors [4, S. 36]

### Timing

Um Standards wie beispielsweise dem VGA-Bildsignal im Bezug auf das Timing gerecht werden zu können, kann zwischen den gültigen Bilddaten ein freier Bereich konfiguriert werden. Weiterhin benötigt der MT9V022-Bildsensor eine Mindestgröße dieser informationsfreien Bereiche für interne Zwecke. Dieser kann zwischen den einzelnen Zeilen, also in der Horizontalen, und zwischen den einzelnen Bildern beziehungsweise Halbbildern, also in der Vertikalen, konfiguriert werden. In Abbildung 3.12 ist die Ausgabe zweier Halbbilder dargestellt. Die grauen Bereiche stellen gültige Bildinformationen dar, die weißen informationsfreie Bereiche. Erkennbar ist weiterhin, dass in vertikaler Richtung sowohl die Höhe des informationsfreien Bereiches zwischen den beiden Halbbildern als auch die Höhe zwischen dem alten und dem neuen Bild definierbar sind. Der vertikale informationsfreie Bereich muss dabei mindestens vier Zeilen betragen und der horizontale informationsfreie Bereich mindestens 43 Pixel. [4]

### 3.3.6. Analog-Digital-Umsetzer

Der Analog-Digital-Umsetzer zum Digitalisieren der Bildinformationen lässt sich durch zwei Parameter konfigurieren. Einerseits lässt sich die Referenzspannung im Bereich von 1,0 V bis 2,1 V stufenweise konfigurieren. Andererseits lässt sich ein nicht linearer Digitalisierungsmodus konfigurieren, dessen minimale Auflösung einem 12-Bit Analog-Digital-Umsetzer entspricht.



Die minimale Auflösung liegt dabei im minimalen Wertebereich. Je größer der Wert wird, desto größer wird auch die Auflösung. Der Verlauf dieser Kurve ist in Abbildung 3.7 dargestellt. Durch diesen nichtlinearen Verlauf kann der zum gemessenen Signal relative Quantisierungsfehler  $F_{relativ}$  über den gesamten Wertebereich minimal gehalten werden, da für den relativen Fehler gilt:

$$F_{relativ} = \frac{F_{absolut}}{Wert_{gemessen}} \quad (3.6)$$

Bei linearem Verlauf der Auflösung wird der Quantisierungsfehler entsprechend beim minimalen Signal am größten. [4]

#### 3.3.7. High Dynamic Range

Die im MT9V022-Bildsensor integrierte Schaltung zum Erweitern des dynamischen Messbereiches (HDR) erlaubt die Definition einer Kurve für die zu verwendende Resetspannung werden. Diese ergibt sich aus drei Zeitwerten und vier Spannungen, die stufenweise umgesetzt werden. Die Zeitwerte  $t_2$  und  $t_3$  können einerseits frei in Abhängigkeit zur gesamten Belichtungszeit  $t_{int}$  nach folgender Formel durch den Wert  $ratio$  definiert werden:

$$t = t_{int} * 0,5^{ratio} \quad (3.7)$$

Der Zeitwert  $t_1$  ergibt sich aus den ersten beiden Werten sowie der gesamten Belichtungszeit:

$$t_1 = t_{int} - t_2 - t_3 \quad (3.8)$$

Andererseits lassen sich auch absolute Werte als Anzahl von Zeilen angeben. Dabei sind  $t_1$  und  $t_2$  frei konfigurierbar, während sich  $t_3$  aus den anderen Werten ergibt:

$$\begin{aligned} t_1 &= ShutterWidth1 \\ t_2 &= ShutterWidth2 - ShutterWidth1 \\ t_3 &= t_{int} - ShutterWidth2 \end{aligned} \quad (3.9)$$

Die Spannungen  $V1$  bis  $V4$  lassen sich in Schritten von 62,5 mV in unterschiedlichen Bereichen konfigurieren. Der Verlauf der Resetspannungen ist in 3.13 dargestellt. Der initiale Reset erfolgt mit der Versorgungsspannung von 3,3 V. Danach wird die Spannung für  $t_1$  auf  $V1$  eingestellt, gefolgt von  $V2$  für  $t_2$  und  $V3$  für  $t_3$ , bis sie schließlich endgültig auf  $V4$  reduziert wird. [4]

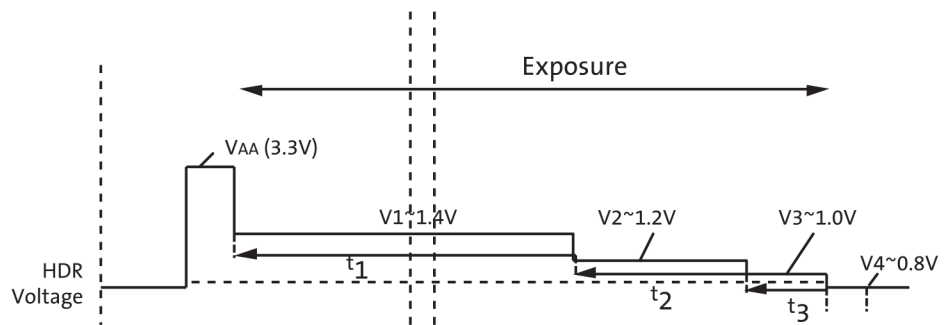


Abbildung 3.13.: Verlauf der Resetspannung im HDR-Modus des MT9V022-Bildsensors [4, S. 26]

## 4. Konfiguration der Videology 24B7525A CMOS-Kamera über den I<sup>2</sup>C-Bus

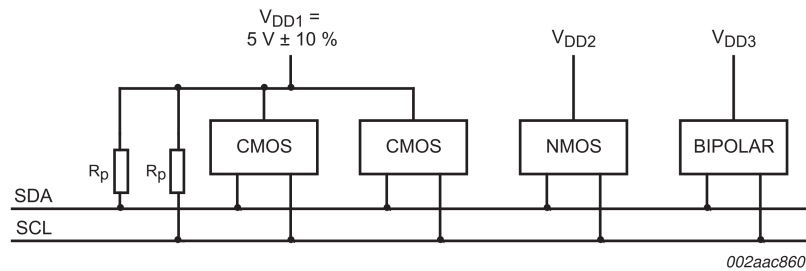
Der I<sup>2</sup>C-Bus wurde von Philips Semiconductors, heute NXP Semiconductors, 1982 initial spezifiziert. Aktuell befindet sich diese Spezifikation in der 4. Version [19]. Dieser Spezifikation sind auch die folgenden Informationen entnommen. Dementsprechend wird im Folgenden der Übersichtlichkeit halber nur bei Informationen, die einem anderen Dokument entstammen, eine Quelle angegeben.

Der in der Spezifikation definierte Bus dient zum Herstellen einer seriellen Verbindung zwischen mehreren Komponenten auf einer Platine mit nur zwei elektrischen Leitungen und wenigen weiteren Komponenten. Die benötigten Schaltungen werden im ersten Abschnitt dieses Kapitels beschrieben. Der I<sup>2</sup>C-Bus wurde zur Konfiguration von weniger intelligenten Komponenten durch intelligentere Komponenten, wie zum Beispiel Mikrocontroller, konzipiert.

Das I<sup>2</sup>C-Bus Protokoll ist im zweiten Abschnitt umfangreich beschrieben. Im dritten Abschnitt dieses Kapitels wird die Anbindung des I<sup>2</sup>C-Bus Interfaces der Videology 24B7525A-Kamera an den Spartan 3A DSP FPGA beschrieben sowie im vierten Abschnitt die für die Anbindung an das Dual-MicroBlaze verwendete Komponente. Im fünften Abschnitt werden Hinweise zur Integration dieses Businterfaces zusammengefasst.

Den wesentlichen Vorteil bei der Verwendung des I<sup>2</sup>C-Busses stellt die Tatsache dar, dass aufgrund der Tatsache, dass lediglich 2 Signale geroutet werden müssen, ein einfacheres Layout der Platine entsteht. Die geringe Anzahl der Signale wirkt sich entsprechend auch bei den IC's positiv aus, da weniger Pins herausgeführt werden müssen und sich somit kompaktere Komponenten entwickelt werden. Abgesehen davon ist die Anzahl der für den I<sup>2</sup>C-Bus benötigten zusätzlichen Komponenten direkt auf der Platine mit lediglich zwei Pull-Up Widerständen sehr gering. Diese Faktoren wirken sich entsprechend positiv auf die Produktionskosten aus.

Zu erwähnen ist, dass diverse Hersteller in ihren Dokumentationen ein Two-Wire-Interface (TWI) beschreiben. Die dort spezifizierten Interfaces gleichen den Spezifikationen des I<sup>2</sup>C-Busses bis ins Detail und sind als äquivalent zu betrachten. Zu nennen sind dabei beispielhaft die Hersteller Micron beziehungsweise Aptina Imaging [4, S. 15] und Atmel [6, S. 169]. Die Dokumentation zu dem in der Videology 24B7525A-Kamera verwendeten MT9V022-Bildsensor stellt dabei ein



V<sub>DD2</sub>, V<sub>DD3</sub> are device-dependent (for example, 12 V).

Abbildung 4.1.: Mehrere I<sup>2</sup>C-Busteilnehmer, auch mit unterschiedlichen Versorgungsspannungen [19, S. 8]

gutes Beispiel dar; in der durch die Firma Videology gelieferten Dokumentation wird das serielle Interface zur Konfiguration als I<sup>2</sup>C-Bus bezeichnet [25], während in der Dokumentation der Firma Aptina von TWI die Rede ist.

#### 4.1. Elektrische Grundlage des I<sup>2</sup>C-Bus

In der Spezifikation ist eine einfache elektrische Grundlage für den I<sup>2</sup>C-Bus definiert. Es handelt sich dabei lediglich um zwei Leitungen mit jeweils einem Pull-Up Widerstand. An diesen beiden Leitungen ist jeder im Bus befindliche IC angebunden. Jeder IC kann die sich im Ruhezustand auf dem High-Pegel befindliche Leitung über eine Open-Drain bzw. Open-Collector Schaltung eine der Leitungen auf den Low-Pegel zu ziehen. Dadurch entsteht ein Wired-AND. Dabei stellt eine der Leitungen die Taktleitung (SCL) und die andere die Datenleitung (SDA) dar. Eine Chip-Select-Leitung für jeden IC ist nicht nötig, da die Auswahl des anzusprechenden IC's über das I<sup>2</sup>C-Bus Protokoll geschieht. In Abbildung 4.1 ist dies verdeutlicht.

#### 4.2. I<sup>2</sup>C-Bus Protokoll

Der I<sup>2</sup>C-Bus ist Multi-Master fähig, da er über Arbitrierungsmechanismen und eine Kollisionserkennung verfügt. Es ist ein Adressraum definiert, der im einfachen Modus 7 Bit und im erweiterten Modus 10 Bit breit ist. Die beiden Adressierungsarten sind dabei untereinander kompatibel, ein 10 Bit fähiger Master kann sowohl einen 7 Bit als auch einen 10 Bit Slave adressieren. Sie können gemeinsam an einem Bus betrieben werden.

In den folgenden Abschnitten wird auf die Bus Arbitrierung eingegangen, der Aufbau der Da-

tenübertragung beschrieben, eine Übersicht über die reservierten Adressen gegeben und auf die Übertragungsraten eingegangen.

##### 4.2.1. Bus Arbitrierung

Das I<sup>2</sup>C-Bus Protokoll sieht vor jeglicher Kommunikation die Arbitrierung des Busses durch den Master vor. Dies geschieht mit Hilfe einer START-Kondition. Des Weiteren muss der Master nach dem Ende einer Übertragung mit einer STOP-Kondition den Bus wieder freigeben. Zusätzlich gibt es noch eine REPEATED-START-Kondition. Diese wird verwendet, wenn zwar die Übertragung beendet werden soll, der Bus danach aber immer noch arbitriert sein soll.

Um eine START-Kondition zu erzeugen, zieht der jeweilige Master die SDA-Leitung auf Low. Voraussetzung dafür ist, dass die SDA-Leitung und SCL-Leitung vorher auf High waren, der Bus also nicht arbitriert war. Danach beginnt der Master mit seiner Übertragung. Stellt er fest, dass ein anderer Master die SDA-Leitung auf Low zieht, so hat eine Kollision stattgefunden und er hat die Arbitrierung über den Bus verloren. Daraus folgt, dass ein Master nicht in der Lage ist eine Kollision festzustellen, wenn der konkurrierende Master dieselben Daten sendet. Dies muss insbesondere beachtet werden, wenn durch den Zugriff ein Event ausgelöst wird, da in diesem Fall anstelle von zwei Events lediglich eines ausgelöst wird.

Um eine STOP-Kondition zu erzeugen, hört der jeweilige Master nach dem letzten Acknowledge Bit auf, die SDA-Leitung auf Low zu ziehen. Dabei ruht die SCL-Leitung auf ihrem High-Pegel. Die START- und STOP-Konditionen sind in der Abbildung 4.2 dargestellt. Um eine REPEATED-START-Kondition zu erzeugen, sendet der jeweilige Master erneut eine START-Kondition, ohne vorher eine STOP-Kondition gesendet zu haben. Durch die REPEATED-START-Kondition kann eine Übertragung beendet werden, ohne dabei die Arbitrierung des Busses zu verlieren.

##### 4.2.2. Datenübertragung

Ist der Bus erfolgreich arbitriert, so kann der Master mit der seriellen Übertragung der Daten beginnen. Der Master beginnt die Übertragung indem er die SCL-Leitung auf Low zieht. Danach legt er das erste Bit an und gibt die SCL-Leitung frei. Der angesprochene Slave zieht gegebenenfalls die SCL-Leitung weiter auf Low. Hierdurch kann der Slave das Anliegen eines validen Datenbits verlängern, um zum Beispiel Betriebsbereitschaft herzustellen. Liegt an der SCL-Leitung nun ein High-Pegel an, so liegt an der SDA-Leitung ein valides Datenbit an. Nach einer der Übertragungsfrequenz entsprechenden Zeit beginnt die Übertragung des nächsten Bits auf die selbe Weise.

Die Datenübertragung beginnt mit dem MSB und endet mit dem LSB. Die ersten 7 Bit stellen die

Adresse des anzusprechenden Slave-IC's dar. Handelt es sich dabei um eine 10 Bit Adresse, so enthalten die ersten 5 Bit den Wert '11110'. Die folgenden 2 Bit enthalten die beiden am höchsten gewerteten Bits der Adresse.

Nach den ersten 7 Bits folgt das  $R/\overline{W}$  Bit. Es zeigt an, ob es sich bei der aktuellen Übertragung um einen schreibenden ('0') oder lesenden ('1') Zugriff handelt.

Darauf folgt ein Acknowledge-Bit, in dem der Slave den Empfang der Daten quittiert. In diesem Zeitraum zieht der Master die SDA-Leitung nicht auf Low, führt aber die Taktung der SCL-Leitung fort. Wird ein Slave von der gesendeten Adresse angesprochen, so zieht er die SDA-Leitung nun auf Low. Zu beachten ist, dass ein solches Acknowledge-Bit nach jedem übertragenden Byte eingefügt wird. Dabei bestätigt der Empfänger der Daten immer den Empfang der Daten und seine weitere Empfangsbereitschaft. Beim Senden einer 10 Bit Adresse folgen die letzten 8 Bit der Adresse.

Die Daten werden danach wie die Adresse übertragen. Das SCL-Signal wird weiterhin vom Master generiert und kann bei Bedarf vom Slave verlängert werden. Das SDA-Signal wird vom Sender der Daten generiert und ein Datenbyte jeweils vom Empfänger bestätigt. Will der Master nur eine bestimmte Anzahl von Bytes empfangen, so zieht er nach dem letzten benötigten Byte das Acknowledge Bit nicht auf Low und beendet somit die Übertragung. In Abbildung 4.2 ist die Übertragungssequenz dargestellt. Nach dem Acknowledge des ersten Bytes ist zu erkennen, wie der Slave die Übertragung durch Low-halten der SCL-Leitung verzögert.

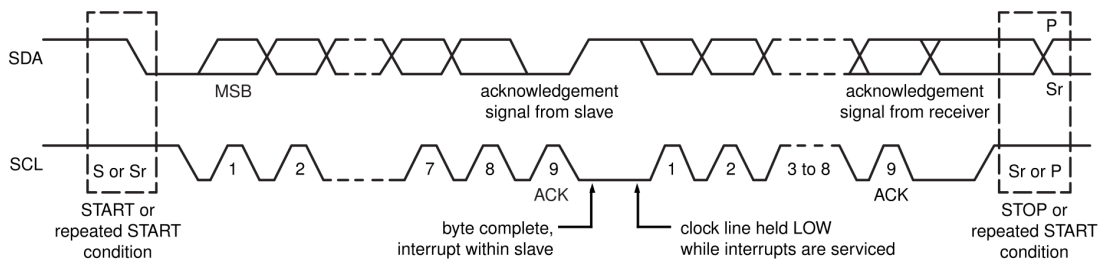


Abbildung 4.2.: Übertragung auf dem I<sup>2</sup>C-Bus mit Verzögerung der Übertragung durch den Slave nach dem Acknowledge des ersten Bytes sowie START- und STOP-Konditionen [19, S. 10]

#### 4.2.3. Reservierte Adressen

Zur Umsetzung spezieller Funktionen existieren reservierte Adressen. Ob diese seitens der Hersteller tatsächlich implementiert ist, ist den jeweiligen Dokumentationen zu entnehmen.

In der Tabelle 4.1 sind die reservierten Adressen aufgelistet.

Adresse	R/ $\overline{W}$ -Bit	Beschreibung
0000 000	0	general call Address
0000 000	1	START Byte
0000 001	X	CBUS Address
0000 010	X	reserved
0000 011	X	reserved
0000 1XX	X	Hs-mode Master code
1111 1XX	X	reserved
1111 0XX	X	10-Bit Addressing

Tabelle 4.1.: Reservierte Adressen des I<sup>2</sup>C-Busses [19, S. 17]

Dabei ist zu beachten, dass dies die reservierten Adressen im Standard-Mode, Fast-Mode und Fast-Mode Plus sind. Im Ultra-Fast Mode weichen sie ab [19, S. 29].

#### 4.2.4. Übertragungsraten

Der I<sup>2</sup>C-Bus erlaubt unterschiedlichen Übertragungsraten. Die maximale Übertragungsrate resultiert aus der Übertragungsrate des langsamsten Busteilnehmers und den Eigenschaften der Busleitungen. Da die Leitungen des Busses in jedem Fall parasitäre Kapazitäten besitzen und diese bei einem Übergang von Low nach High über den Pull-Up Widerstand geladen werden müssen, ergibt sich hierdurch gegebenenfalls eine maximale Taktrate. Dies kann zwar durch ein entsprechendes Layout der Leiterbahnen auf der Platine verringert, aber nie vollständig unterbunden werden. Dabei gilt:

$$\tau_{Pull-Up} = R_{Pull-Up} * C_{Leitung} \quad (4.1)$$

Hinzu kommt gegebenenfalls ein Schutzwiderstand an den Open-Collector Schaltungen der IC's. Dieser muss unter Umständen verwendet werden, da die Leitungskapazität vom Übergang von High auf Low über einen Transistor entladen werden muss. Dabei ergibt sich:

$$\tau_{Schutz} = R_{Schutz} * C_{Leitung} \quad (4.2)$$

Daraus ergibt sich dann die maximale theoretisch erreichbare Frequenz:

$$f_{max} = \frac{1}{\tau_{Pull-Up} + \tau_{Schutz}} \quad (4.3)$$

Modus	Übertragungsrate
Standard-Mode	100 kbit/s
Fast-Mode	400 kbit/s
Fast-Mode Plus	1 Mbit/s
High-Speed Mode	3,4 Mbit/s
Ultra Fast-Mode	5 Mbit/s (Unidirektional)

Tabelle 4.2.: Übertragungsraten I<sup>2</sup>C-Bus [19, S. 35]

Die I<sup>2</sup>C-Bus Spezifikation definiert mehrere Modi mit unterschiedlichen maximalen Übertragungsraten. Diese sind in der Tabelle 4.2 aufgelistet.

### 4.3. FPGA-Interface des MT9V022 Bildsensors an den I<sup>2</sup>C-Bus

Das I<sup>2</sup>C-Bus Interface wurde weitestgehend wie vorgesehen [4, S. 9] an den FPGA angebunden. Da in den Spartan 3A DSP FPGA für jeden I/O Port ein Pull-Up Widerstand hinzu geschaltet werden kann, wurden diese verwendet. [30, S. 332, 31, S. 12] Dass die Pull-Up Widerstände der Ports einen Wert von etwa 11 k $\Omega$  haben, hat sich jedoch nicht als problematisch erwiesen. Die internen Pull-Up Widerstände werden mit Hilfe der UCF-Datei konfiguriert. Bei den entsprechenden Constraints muss dazu lediglich PULLUP angefügt werden. Eine vollständige Darstellung der elektrischen Anbindung samt Dateninterface ist in Kapitel 6.1 beschrieben.

### 4.4. I<sup>2</sup>C-Bus Interface des MicroBlaze

Xilinx stellt mit dem XPS IIC Interface IP ein I<sup>2</sup>C-Bus Interface bereit. [33] Dieses wurde mit entsprechenden Parametern in der Version 2.03.a verwendet.

Laut dem Hersteller des Bildsensors beträgt die maximale Taktrate des implementierten I<sup>2</sup>C-Bus Interfaces 400 kHz. [3, S. 1] Jedoch hat sich lediglich eine Taktrate von 5 kHz als zuverlässig erwiesen. Bei einer höheren Taktrate kam es bei der Erprobung der Konfiguration wiederholt zu fehlerhaften Übertragungen. Eine Zuverlässigkeit der Konfiguration ist jedoch von essentieller Bedeutung, da die fehlerhafte Übertragung eines einzelnen Bits den Bildsensor beispielsweise in einen Schlafmodus versetzen kann. Zwar können die Werte nach jedem Schreiben überprüft werden. Allerdings würde dies bei einer höheren Frequenz zu einem mehrfach wiederholten Schreiben der Werte führen. Die Anzahl der Schreibvorgänge ist jedoch gering zu halten, da davon auszugehen ist, dass der Bildsensor die Konfigurationswerte intern auf einem Speicher hält, der nur eine begrenzte Anzahl von Schreibzyklen zur Verfügung stellt. Des Weiteren wird der Konfigurationsvorgang nur beim Start des Systems durchgeführt und unterliegt somit keinen



zeitkritischen Vorgaben.

Versuche während der Entwicklung der Konfigurationssoftware haben weiterhin ergeben, dass eine Pause von mindestens 20 ms zwischen den einzelnen Datenübertragungen nötig ist. Wird diese Pause nicht eingehalten, so reagiert das I<sup>2</sup>C-Bus Interface der Kamera nicht mehr auf weitere Übertragungen.

Daraus ergeben sich die in Tabelle 4.3 angegebenen Parameter für den Xilinx XPS IIC Interface IP. In der I<sup>2</sup>C-Bus Spezifikation [19, S. 47] ist vermerkt, dass im Fast-Mode Betrieb eine Filterung

Parameter	Wert
PLBv46 Bus Clock Frequency	AUTO
Output Frequency of SCL Signal	5000
Use 10-bit Address	7-bit Address
Width of GPIO	1
Width of glitches removed on SCL input	5
Width of glitches removed on SDA input	5

Tabelle 4.3.: Konfigurationsparameter für das Xilinx XPS IIC Interface IP

der SCL und SDA Signale nötig ist. So sollen Störimpulse mit bis zu 50 ns Länge heraus gefiltert werden. Dies wird im XPS IIC Interface IP durch die beiden Parameter 'Width of glitches removed on SCL input' und 'Width of glitches removed on SDA input' konfiguriert. Die maximale Breite der entfernten Störimpulse wird hierbei in Taktperioden angegeben. [33, S. 3] Zwar wird das I<sup>2</sup>C-Bus Interface nicht im Fast-Mode betrieben, jedoch erhöht eine Filterung des Signals die Stabilität des Interfaces im Betrieb.

Die I<sup>2</sup>C-Bus Adresse des Bildsensors ist über die Eingänge S\_CTRL\_ADR0 und S\_CTRL\_ADR1 konfigurierbar. Die Adressen sind mit 0x90, 0x98, 0xB0 und 0xB8 angegeben. [4, S. 16] Da an der Kamera jedoch keine Komponenten erkennbar sind, die eine solche Konfiguration erlauben, und diesbezüglich auch nichts dokumentiert ist, ist von einer fest konfigurierten Adresse auszugehen. [25] Um herauszufinden, welche Adresse durch den Bildsensor belegt ist, kann für jede konfigurierbare Adresse ein Lesezugriff begonnen werden. Anhand des nach der Adresse anliegenden Acknowledge-Signals lässt sich so ermitteln, ob die Adresse belegt ist.

Bei genauerem Betrachten der oben genannten Adressen fällt jedoch auf, dass sie nicht durch den Wertebereich einer 7 Bit Integer Zahl darstellbar sind ( $0x90=144 > 127$ ). Daraus lässt sich schließen, dass die letzte binäre Stelle der angegebenen Adressen das  $R/\overline{W}$ -Bit darstellt. Das Software-Interface des Xilinx XPS IIC Interface IP fordert die Adressen jedoch ohne das  $R/\overline{W}$ -Bit. Dementsprechend ergab das Scannen des Busses, dass der Bildsensor auf die Adresse 0x5C, also die von Aptina angegebene Adresse 0xB8, reagiert.

Zur Validierung wurde ein Lesezugriff mit einem PicoScop-3204 Oszillographen aufgezeichnet

(siehe Abbildung 4.3). Die senkrechten Linien markieren die START-, REPEATED-START- und STOP-Konditionen. Mit den Messsonden wurde das SCL- und SDA-Signal abgegriffen.

Bei der ersten Übertragung wird mit den ersten 7 Bit die Adresse 0x5C übermittelt. Danach wird mit dem  $R/\overline{W}$ -Bit signalisiert, dass es sich bei dieser Übertragung um einen schreibenden Zugriff handelt, gefolgt vom Acknowledge des Bildsensors. Darauf folgt die Nummer des Registers mit 0xBA im Datenteil, gefolgt von einem Acknowledge.

Die Pause zwischen den Übertragungen wurde für diese Abbildung der Übersicht halber verringert.

Darauf folgt eine weitere Übertragung an die Adresse 0x5C, in der das  $R/\overline{W}$ -Bit einen lesenden Zugriff signalisiert. Nach dem Acknowledge folgen 2 Byte Daten, die vom Bildsensor ausgegeben wurden. Das erste Byte 0x00 ist gefolgt von einem Acknowledge, während nach dem zweiten Byte 0x10 die Übertragung mit einem Not-Acknowledge beendet wird.

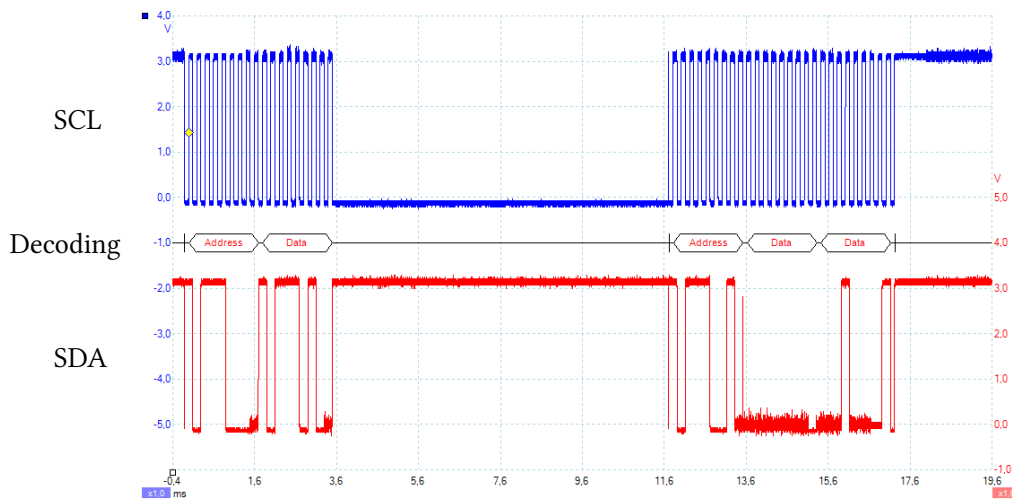


Abbildung 4.3.: Lesen des Registers 0xBA des MT9V022-Bildsensors (Adresse 0x5C) über den I<sup>2</sup>C-Bus; erste Übertragung enthält Adresse und Register, die zweite Übertragung die Adresse und die gelesenen Daten (0x0010) ; senkrechte Markierungen: START-, REPEATED-START und STOP-Konditionen (PicoScop-3204)

### 4.5. Checkliste zum Integrationsvorgang

Der Vorgang der Integration einer Komponente mit I<sup>2</sup>C-Bus Interface war Bestandteil mehrerer vorhergehender Arbeiten. Dabei ergaben sich unvorhergesehene und undokumentierte Eigenschaften der Komponenten, die den Integrationsprozess in die Länge zogen. Diese sind hier zusammengefasst und um weitere Hinweise ergänzt.

### **Busleitung kurz und Anzahl der Steckkontakte gering halten**

Jeder Steckkontakt stellt eine Stelle dar, an der es gegebenenfalls einen erhöhten Leitungswiderstand oder eine Unterbrechung der Leitung geben kann. Dementsprechend ist deren Anzahl gering zu halten. Eine lange Busleitung führt zudem zu einer erhöhten Kapazität und einem erhöhten Widerstand. Der Bus stellt durch den Pull-Up Widerstand und die Leitungskapazität ein RC-Glied dar. Dies ist in Kapitel 4.2.4 genauer beschrieben.

### **Bidirektionale Levelshifter für unterschiedliche Betriebsspannungen**

Werden die an den Bus angeschlossenen Komponenten mit unterschiedlichen Betriebsspannungen betrieben, so wird der Bus in Bereiche unterteilt. Die Bereiche richten sich nach dem von den Komponenten verwendeten Buspegeln. Zwischen den einzelnen Abschnitten werden Levelshifter verwendet. Sie sorgen für eine bidirektionale Übertragung der Signale über die Grenzen der unterschiedlichen Bereiche hinaus. Dies ist in der Spezifikation des I<sup>2</sup>C-Busses weiter beschrieben. [19, S. 42 f.]

### **Minimale Taktrate**

Da sich der Bus wie ein RC-Glied verhält (siehe Kapitel 4.2.4), sollte anfangs mit einer kleinen Busfrequenz gearbeitet werden. Wie aus dieser Arbeit hervorgeht, kann diese auch wesentlich unter der spezifizierten maximalen Busfrequenz liegen. Bei erfolgreichen Funktionstests kann die Busfrequenz dann schrittweise erhöht werden.

### **Für erste Tests statische Register lesen**

Oftmals ist in der Dokumentation der an den Bus angeschlossenen Komponenten ein Register mit statischem Inhalt angegeben. Register, die die aktuelle Chip- oder Softwareversion enthalten, haben beispielsweise einen bekannten, statischen Inhalt. Durch diese Faktoren können die gelesenen Werte validiert werden.

### **I<sup>2</sup>C-Bus Adresse suchen**

Ist die I<sup>2</sup>C-Bus Adresse einer Komponente unbekannt, so kann sie durch ein Brute-Force-Verfahren geraten werden. Dies bietet sich aufgrund des relativ kleinen Adressraumes an. Das Vorgehen ist in Kapitel 4.4 beschrieben.

### **Gegebenenfalls auf REPEATED-START-Kondition verzichten**

Einige Komponenten sind nicht in der Lage die REPEATED-START-Kondition korrekt zu verarbeiten. Dies war in der referenzierten Arbeit auch nicht in der Dokumentation der Komponente vermerkt, führte in diesem Fall aber zu einem fehlerhaften Verhalten der Komponente. Auf diese kann, solange an den I<sup>2</sup>C-Bus nicht mehrere zu synchronisierende Master angeschlossen sind, problemlos verzichtet werden. Die REPEATED-START-Kondition wird durch eine STOP-Kondition gefolgt von einer START-Kondition ersetzt. [15, S. 80]

### **Pausen zwischen den Übertragungen**

Angeschlossene Komponenten benötigen gegebenenfalls Zeit, um die übermittelten Kommandos zu verarbeiten. Das Einhalten von Pausen zwischen den einzelnen Übertragungen führte nicht nur in dieser Arbeit zu einer wesentlich besseren Stabilität des Gesamtsystems, sondern auch im Rahmen der Bachelorarbeit von Edris Sahak. [23, S. 23]

### **Gesperrter Schreibzugriff auf Register**

Um ein ungewolltes Ändern eines Registerwertes zu verhindern, verfügen Komponenten eventuell über ein Register zum Sperren des Schreibzugriffes. Der Schreibzugriff auf die Register muss mit dem Schreiben eines bestimmten Wertes auf dieses Register wieder freigegeben werden. [4, S. 19]

## 5. Parametrisierungssoftware für den MT9V022 Bildsensor

Die umfangreichen Konfigurationsmöglichkeiten des MT9V022-Bildsensors wurden analysiert und eine für den Betrieb mit dem Spurführungssystem geeignete Konfiguration ausgearbeitet. Um diese in den Bildsensor zu laden, wurde ein Softwareinterface zur Konfiguration des Bildsensors erstellt. Dieses wird im Folgenden beschrieben.

### 5.1. Register zur Parametrisierung der Kamera

Die Dokumentation der Register zur Konfiguration des MT9V022-Bildsensors [4] sind seitens der Hersteller des Bildsensors und des Kameraboards 24B7525A [25] unvollständig. Um die Informationen zu vervollständigen, wurde eine Dokumentation zu einem softwareseitigen Treiber der Firma Analog Devices hinzugezogen [1]. Zur Übersicht wurde eine möglichst vollständige Liste der Register erstellt. Sie ist im Anhang A.1 dieser Arbeit angefügt.

### 5.2. Parametersatz für den Betrieb der Kamera

Aus der Analyse der vorhandenen Kameraregister wurde ein geeigneter Parametersatz für den Betrieb erstellt. Eine Liste der daraus resultierenden Werte ist dem Anhang A.2 zu entnehmen. Zu beachten ist, dass die Ausgabe eines Testpatterns durch die Kamera aktiviert werden kann. Dies ist in dieser Konfiguration deaktiviert.

Zuerst wird auf die Bildausschnitts- und Timingkontrolle eingegangen und danach die Belichtungs- und Verstärkungsfaktorkontrolle behandelt. In den darauf folgenden Abschnitten wird die Konfiguration des Analog-Digital-Umsetzers, der Bildrauschunterdrückung und Schwarzwertkorrektur hergeleitet. Der letzte Abschnitt befasst sich mit der Konfiguration der Ausgabesignale.

Bez.	Wert	Spannung
V1	29	3,375 V
V2	24	2,0625 V
V3	21	1,875 V
V4	4	0,8125 V

Tabelle 5.1.: Reset-Spannungswerte für den HDR-Modus

### **Bildausschnitts- und Timingkontrolle**

Für die vorhandene Bildverarbeitungskette wird aktuell eine Auflösung von 640 x 480 Pixeln verwendet. Daraus und aus den Anforderungen bezüglich des Timings ergeben sich die Werte für die Bildausschnitts- und Timingkontrolle. Auf das Zustandekommen des Timings wird in Kapitel 6.3 genauer eingegangen.

### **Belichtungs- und Verstärkungskontrolle**

AEC und AGC sind zur automatischen Anpassung der Belichtungszeit und des Verstärkungsfaktors aktiviert. Dabei wurde jeweils der Tiefpass auf die langsamste Anpassung der Werte bei einer Verzögerung der Übernahme der Werte von 2 Bildern eingestellt. Als angestrebter Wert für die gemittelte Helligkeit des Bildes hat sich ein Wert von 58 ergeben. Der Wert für die aktuelle gemittelte Helligkeit wird aus 44000 Pixeln berechnet. Die maximale Belichtungszeit wurde mit 512 Zeilen so gewählt, dass sie die Bildrate nicht reduziert. Dies ergibt sich aus dem in Kapitel 6.3 entwickelten Timing. Der maximale analoge Verstärkungsfaktor ist der maximale verfügbare Wert von 63, was einer 3,9375 fachen Verstärkung entspricht. Der digitale Verstärkungsfaktor für die einzelnen Bereiche wurde auf 1 eingestellt. Die Einflussnahme der einzelnen Bereiche in die AEC- und AGC-Algorithmen wurde auf den maximalen Wert von 15 eingestellt. Die Bereiche sind gleichmäßig über die volle Auflösung des Bildsensors verteilt, jedoch irrelevant, da für alle Bereiche dieselben Werte eingetragen wurden.

Da der vorrangige Betriebsraum des Fahrzeuges aktuell aus geschlossenen und fensterlosen Räumen besteht, ist der HDR-Modus deaktiviert, jedoch sind in der Konfiguration zur einfachen Aktivierung bereits Werte eingetragen. Aufgrund der automatischen Belichtungszeitkontrolle ist eine automatische Anpassung der Eckpunkte der Reset-Spannungskurve aktiviert. Die Reset-Spannungen sind in Tabelle 5.1 aufgelistet.

Die Anti-Eclipse-Schaltung wurde deaktiviert.

### **Analog-Digital-Umsetzer**

Die Auswirkungen auf die Binarisierung bei einem Aktivieren der nichtlinearen Quantisierung des Analog-Digital-Wandlers wird in Kapitel 7.2 erläutert. Daraus geht hervor, dass ein Aktivieren bei gegebenen Lichtumständen zu keinen verbesserten Werte führt. Ob ein Aktivieren in besonders dunklen Räumen vorteilhaft ist, muss gegebenenfalls ermittelt werden. Dabei muss jedoch beachtet werden, dass dies zu einem verstärkten Messen von Rauschen im unteren Wertebereich führt. Als Referenzspannung für die Digitalisierung wurde der vom Hersteller vorgeschlagene Wert von 1,4 V eingestellt. [25, S. 15]

### **Schwarzwertkorrektur und Bildrauschunterdrückung**

Die Schwarzwertkorrektur ist aktiviert und es ist eine Filterung des gemittelten Schwarzwertes über 4 Bilder konfiguriert. Sowohl der untere als auch der obere Schwellwert für die Schwarzwertkorrektur ist auf einen Wert von 8 eingestellt. Als Schrittgröße für die Korrektur ist der kleinste Wert von 1 festgelegt. Dies entspricht einem halben LSB des Analog-Digital-Umsetzers der Pixelausleseschaltung.

Die Bildrauschunterdrückung wurde aktiviert und es wurden 10 optisch geschwärzte Pixel zur Ermittlung des Zeilenrauschens konfiguriert. Als erster für die Bildrauschunterdrückung zu verwendeter Pixel wurde ein gültiger Wert von 759 eingestellt. Die Konstante, die zu allen Pixeln addiert werden soll, wurde auf einen Wert von 0 eingestellt.

### **Konfiguration der Ausgabesignale**

Die parallele Schnittstelle des Bildsensors wurde aktiviert, der Sensor ist in jedem Fall Master, die Stereoskopiefunktion wird nicht verwendet und die Funktion zur Aufnahme von Einzelbildern wurde deaktiviert. Weiterhin wird der Bildsensor zur Verlängerung der Belichtungszeit im simultanen Modus ausgelesen und defekte Pixel werden korrigiert. Eine Ausgabe der Bilddaten im Zeilensprungverfahren (Interlacing) wurde nicht aktiviert.

Die Reduktion der Auflösung durch Mittelung von mehreren Pixeln wurde nicht aktiviert, da die Datenrate problemlos verarbeitet werden kann und eine Bildfrequenz von 60 Bildern pro Sekunde ausreichend ist. Die Invertierung der Auslesereihenfolge ist nicht nötig, da die Kamera entsprechend montiert wurde. Optisch geschwärzte Pixel werden nicht mit ausgelesen, da sie nicht benötigt werden. Der Modus zur Aufnahme eines Bildes alle 5 Minuten wurde deaktiviert. Zur Messung der Belichtungszeit kann die Ausgabe des Exposure-Signals aktiviert werden. Es wurde jedoch nicht aktiviert, da es nicht verwendet wird und gegebenenfalls lediglich Störungen auf den anderen Leitungen und einen erhöhten Energieverbrauch verursacht.

Frame- und Linevalid-Signale werden normal erzeugt, da sie in dieser Form benötigt werden. Die Ausgabe von NTSC-Synchronisationssignalen anstelle der Frame- und Linevalid-Signale wurde entsprechend deaktiviert.

Die serielle Ausgabe der Bilddaten nach dem LVDS-Standard wird nicht verwendet. Dementsprechend sind alle Schaltungen zum Generieren dieses Signals abgeschaltet. Auch die LVDS-Empfangsschaltung, die für die Stereoskopiefunktion benötigt wird, wurde deaktiviert.

### 5.3. Softwareinterface zur Konfiguration der Kamera

Da der MT9V022 Bildsensor die in den Registern konfigurierten Werte nach einer Unterbrechung der Spannungsversorgung durch Standardwerte ersetzt, wurde ein Interface zur Konfiguration des Bildsensors seitens der auf dem MicroBlaze laufenden Software entwickelt. Dabei wurde besonders darauf geachtet, dass die vorgegebene Konfiguration trotz eventueller Übertragungsfehler des I<sup>2</sup>C-Busses fehlerfrei in den Sensor eingespielt werden kann.

Zur Kommunikation mit dem MT9V022 Bildsensor wurde das durch die Firma Xilinx bereitgestellte XPS IIC Interface IP verwendet. Die Firma Xilinx stellt zu diesem IP weiterhin ein Softwareinterface bereit. Das Softwareinterface stellt im Wesentlichen Methoden zum Lesen und Schreiben auf den I<sup>2</sup>C-Bus bereit. Diese wurden verwendet und sind in der entsprechenden Dokumentation der Firma Xilinx beschrieben.

Weiterhin hat es sich als nötig erwiesen, zwischen den einzelnen I<sup>2</sup>C-Bus Paketen ein bestimmtes Timeout einzuhalten, da das I<sup>2</sup>C-Interface andernfalls nicht mehr reagiert. Um diesen Timeout zu umzusetzen, wird eine Funktion zum Abwarten einer definierten Zeit benötigt. Diese muss bei einer Portierung auf ein anderes System in folgender Form bereitgestellt werden.

```
1 void sleep(u32 timeMs);
```

Die Angabe der Zeit erfolgt in Millisekunden.

Das implementierte Softwareinterface stellt grundlegende Funktionen zur Initialisierung der benötigten Hardware, zum Lese- und Schreibzugriff auf die Register und zum softwareseitigen Reset des Bildsensors sowie Makros mit allen wichtigen Konstanten bereit. Weiterhin bietet es Funktionen und Datenstrukturen zur Konfiguration und Kontrolle auf einer höheren Ebene.

Ein vollständiges Listing ist im Anhang [A.3](#) angefügt.

#### 5.3.1. Low-Level Funktionalitäten

Um die einzelnen Registeradressen auf Softwareseite für den Anwender des Interfaces flexibel verfügbar zu machen, wurden alle Registeradressen als Makros mit dem Präfix CAMERA\\_REG\\_



in das Listing eingefügt. Der allgemeine Rückgabewert der Funktionen ist, sofern nicht anders beschrieben, `XST_SUCCESS` bei Erfolg und `XST_FAILURE` bei einem Fehler. Im Folgenden sind die einfachen Funktionen des Interface aufgelistet:

- `u32 cameraInit()`: Setzt das XPS IIC Interface IP zurück und initialisiert es.
- `u32 cameraSwReset(u8 id)`: Führt einen Softwarereset auf dem Bildsensor mithilfe des Reset Registers durch. Es muss die I<sup>2</sup>C-Bus Adresse des Bildsensors übergeben werden.
- `u32 cameraReadReg(u8 id, u8 reg, u16* value)`: Liest den Inhalt des Register `reg` des Bildsensors `id` in den Speicherbereich `value`.
- `u32 cameraWriteReg(u8 id, u8 reg, u16 value, u8 isChecked)`: Schreibt den Wert `value` in das Register `reg` des Bildsensors `id`. Durch `isChecked` kann mit `CAMERA_WRITE_CHECKED` beziehungsweise `CAMERA_WRITE_UNCHECKED` angegeben werden, ob der Schreibvorgang überprüft und bis zu einer Anzahl von `REG_WRITE_RETRY` mal wiederholt werden soll. Dabei wird vor jedem Schreibvorgang auf ein Register geprüft, ob der im Register schon vorhandene Wert dem zu schreibenden Wert entspricht. Da die Speichertechnologie der Register unbekannt ist und es sich gegebenenfalls um einen FLASH-artigen Speicher handelt, verhindert dies eine unnötige Abnutzung der Speicherzellen. Zudem werden, um eventuelle Lesefehler auszuschließen, vor jedem Schreibversuch eine Anzahl von `REG_WRITE_RETRY` Leseversuche durchgeführt.

### 5.3.2. High-Level Funktionalitäten

Zur einfachen Konfiguration des Bildsensors wurden Funktionen und Datenstrukturen auf einer höheren Abstraktionsebene erstellt. Sie umfassen Funktionen zum Laden der Registerkonfiguration in den Speicher und zum Schreiben einer Registerkonfiguration in den Bildsensor. Zusätzlich sind Hilfsfunktionen zur übersichtlichen Ausgabe der Registernamen und -werte auf der Standardtextausgabe vorhanden. Um diese Funktionalitäten zu realisieren, wurden zwei Datenstrukturen erstellt:

- `cameraRegValue`: Beinhaltet jeweils die 8 Bit Registeradresse `reg` und den zugeordneten 16 Bit Wert `value`.
- `cameraRegName`: Beinhaltet jeweils die 8 Bit Registeradresse `reg` und den zugeordneten Zeiger `name` auf eine Zeichenkette, die den Namen des Registers enthält.

Eine Konfiguration besteht immer aus einem Array über Elemente des Typs `cameraRegValue`. Folgende Funktionen stehen zur Verfügung:

- `u32 cameraGetDefaultConfiguration(cameraRegValue* regValues, u32 size)`: Diese Funktion schreibt die ausgearbeitete Standardkonfiguration der Kamera in einen vor dem Aufruf allokierten Speicherbereich `regValues` und gibt die Anzahl der geschriebenen Elemente zurück. Wird als Speicherbereich `NULL` übergeben oder mit `size` eine zu kleine Größe des Speicherbereiches angegeben, werden keine Werte geschrieben. Lediglich die Anzahl der Elemente, die geschrieben werden würden, zurückgegeben. Anhand dieses Rückgabewertes kann dann ein entsprechend großer Speicherbereich allokiert werden. Die Größe des benötigten Speicherbereiches ergibt sich aus Rückgabewert \* `sizeof(cameraRegValue)`.
- `u32 cameraWriteConfiguration(u8 id, cameraRegValue* regValues, u32 size, u8 isChecked)`: Diese Funktion schreibt eine Konfiguration `regValues` aus `size` Registern auf den Bildsensor mit der I<sup>2</sup>C-Busadresse `id`. Die Verwendung des Parameters `isChecked` entspricht der Funktion `cameraWriteReg()`. Scheitert das Schreiben eines Registers, so wird der Vorgang abgebrochen und der Fehler gemeldet.
- `u32 cameraListAvailableToSTUDIO()`: Diese Funktion schreibt alle an den I<sup>2</sup>C-Bus angeschlossenen Komponenten und den Inhalt des Registers `0x00` auf die Standardausgabe. Das Register `0x00` des Bildsensors beinhaltet die Chipversion und somit je nach Version die Werte `0x1311` oder `0x1313`. [2, S. 2] Dabei werden alle 128 I<sup>2</sup>C-Busadressen durch den Versuch eines Lesezugriffs auf das Register `0x00` überprüft. Wird eine der Adressen mit einem Acknowledge bestätigt, so ist unter dieser Adresse eine Komponente ansprechbar. Die Funktion gibt die Anzahl der gefundenen Komponenten zurück.
- `char* cameraGetRegisterName(u8 reg)`: Diese Funktion löst die Registeradresse `reg` in einen leserlichen Namen auf. Zu diesem Zweck wurde ein Array mit Elementen des Typs `cameraRegName` angelegt. In ihm sind sämtliche Register mit den zugehörigen Namen abgelegt. Um eine effiziente Suche in diesem Array umzusetzen, wurden die Elemente nach der Registeradresse sortiert abgelegt. Durch eine binäre Suche in diesem Array wird der Aufwand zur Auflösung ohne weiteren Speicherbedarf gering gehalten. Wurde die Registeradresse `reg` nicht gefunden, so wird `NULL` zurückgegeben, ansonsten ein Zeiger auf die entsprechende Zeichenkette.
- `void cameraDumpRegisterValuesToSTUDIO(u8 id)`: Diese Funktion list alle Registerwerte aus dem Bildsensor aus und stellt eine leserliche Ausgabe auf der Standardausgabe zur Verfügung.

## 6. Kamera-Integration in das System-on-Chip basierte Bildverarbeitungssystem

Ausgangspunkt für die Integration der 24B7525A-Kamera in das Bildverarbeitungssystem war die von Marco Kirschke in einer vorherigen Arbeit entwickelte Konfiguration aus Komponenten. [16] Die Anpassungen des bestehenden Systems werden in diesem Kapitel beschrieben. In den folgenden Abschnitten wird die Einbindung des Dateninterfaces der neuen Kamera behandelt, danach ein Timing nach VGA-Standard erläutert und die Realisierung dieses Timings seitens der Kamera hergeleitet. Darauf folgt eine Beschreibung des neuen VGA-Interfaces und des FrameGrabber.

Die ursprüngliche Bildverarbeitungskette ist in Abbildung 6.1 dargestellt. Da die vorher verwendete CCD-Kamera ein Farbbild lieferte, wurden über das 8 Bit breite parallele Dateninterface abwechselnd ein Helligkeits- und einer der beiden Chrominanzwerte bei einer Datenrate von 27 MHz übermittelt. Daraus ergab sich eine Pixelrate von 13,5 MHz. Zusätzlich wurden die Synchronisationssignale nicht direkt über Leitungen herangeführt, sondern aus in den Bilddatenstrom eingebetteten Daten generiert. Zur Ausgabe auf einem Monitor mit VGA-Interface wurden aus den Helligkeits- und Chrominanz-Werten RGB-Werte berechnet. Um eine Bildrate von 60 Hz zu erreichen, wurde in der CCD-Kamera der Interlaced-Modus aktiviert. Durch den Interlaced-Modus hatte das Bild jedoch lediglich eine Höhe von 240 Pixeln, was nicht dem VGA-Standard entspricht. Dementsprechend musste vor der Darstellung auf dem VGA-Monitor eine Zeilenverdopplung durchgeführt werden. Erst danach war die für die Ausgabe nach VGA-Standard nötige Pixelfrequenz erreicht. [16]

Um einen verwertbaren Pixelstrom zu erhalten, wurden also erst die Synchronisationssignale aus den Daten gewonnen, danach fand ein Demultiplexing der Daten in die einzelnen Chrominanz- und Helligkeitswerte statt. Die Helligkeitswerte wurden nun der Bildverarbeitungskette zur Verfügung gestellt. Zur Ausgabe durch das VGA-Interface beziehungsweise dem FrameGrabber wurde danach, um zu jedem Helligkeitswert einen der beiden erforderlichen Chrominanzwerte zuzuordnen, noch ein Upsampling der Daten vorgenommen. Danach fand eine Umrechnung in das RGB-Format statt und die Daten wurden über einen Multiplexer an das VGA-Interface und den FrameGrabber weitergegeben. Im VGA-Interface wurden die Zeilen schließlich verdoppelt

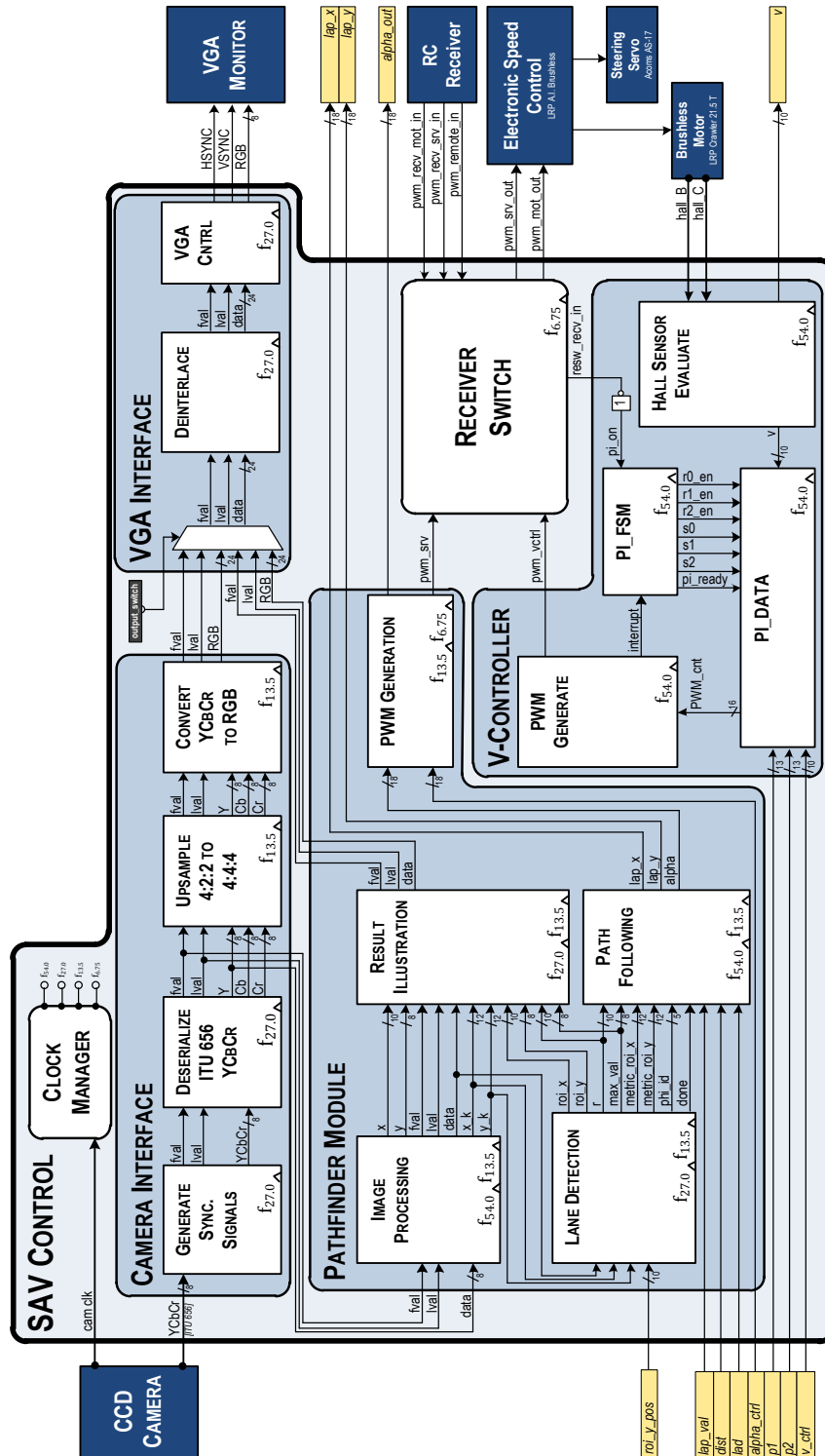


Abbildung 6.1.: Ursprüngliche SAV-Control Block [16, S. 48]

## 6. Kamera-Integration in das System-on-Chip basierte Bildverarbeitungssystem

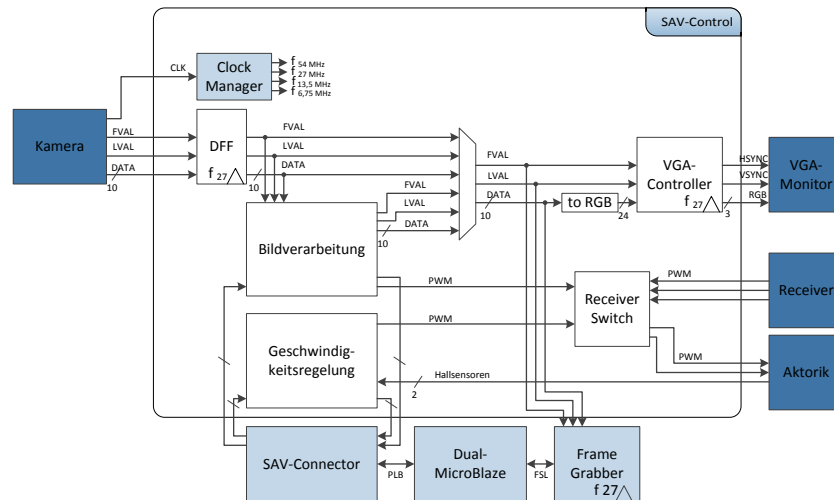


Abbildung 6.2.: Angepasster SAV-Control Block mit angeschlossenen Komponenten

und die Synchronisationssignale generiert. [16]

Der in der 24B7525A-Kamera verbaute MT9V022-Bildsensor liefert 10 Bit Helligkeitswerte ohne Farbinformationen bei einer Pixelrate von 27 MHz. Da dies in einer Bildrate von 60 Hz resultiert, ist der Interlaced-Modus zur Ausgabe nach VGA-Standard nicht nötig. Zudem werden die Synchronisationssignale der Kamera mit zusätzlichen Leitungen direkt in das Bildverarbeitungssystem hereingeführt. Dementsprechend fallen die einzelnen Komponenten des Kamerainterfaces größtenteils weg. Die Umrechnung in RGB-Werte wurde zudem in das VGA-Interface verschoben. Dies wurde nötig, da der FrameGrabber an die höhere Pixelfrequenz angepasst werden musste. Dementsprechend erhalten FrameGrabber und VGA-Interface lediglich 10 Bit Helligkeitswerte und die Synchronisationsdaten des Kamera-Interface beziehungsweise der Result-Illustration der Bildverarbeitungskette. Bedingt durch die höhere Pixelfrequenz ist eine Zeilenverdopplung im VGA-Interface nicht mehr nötig und wurde entfernt.

Die Anpassung der Bildverarbeitungskette an das breitere Datenformat und die höhere Pixelfrequenz des Bildsensors wurde noch nicht vorgenommen, da dies den Umfang dieser Arbeit zu sehr ausgeweitet hätte. Anzupassen ist weiterhin die Ausgabe der Result-Illustration auf 10 Bit Helligkeitswerte.

Das Ergebnis der Integration ist im Anhang A.3 zu finden. Eine Strukturübersicht über den angepassten SAV-Connector ist Abbildung 6.2 zu entnehmen.

## 6.1. Einbindung des Dateninterfaces der Kamera

Da die 24B8525A-Kamera von Videology lediglich über einen stark miniaturisierten Molex-Stecker in SMD-Bauweise verfügt, wurde eine Adapterplatine zum Herausführen der benötigten Signale auf eine Pfostensteckerleiste verwendet. Der Molex-Stecker verfügt über 30 Kontakte, die Pfostensteckerleiste aber lediglich über 24 Kontakte. Dies ergibt sich aus 2 nicht verwendeten Kontakten und einem vierfachen Herausführen des Massesignals auf dem Molex-Stecker. Zusätzlich wurde auf ein Herausführen des analogen Videosignals auf der Adapterplatine verzichtet und die Spannungsversorgung von den Datenleitungen getrennt. Diese Adapterplatine entstand im Rahmen vorheriger Projektaktivitäten.

Da Steckkontakte häufig eine nur unsaubere Verbindung darstellen und gegebenenfalls Verbindungsprobleme verursachen, wurde darauf geachtet, die Datenleitungen der Kamera direkt an den FPGA anzubinden. Dementsprechend wurden sie nicht über das angeschlossene Expansionsboard, sondern direkt über die nicht verwendete SAM-Pfostensteckerleiste (SystemACE Module) geleitet. [32, S. 38] Zudem verfügt nur die SAM-Pfostensteckerleiste über genügend Kontakte sowie einen dedizierten Takteingang. Durch die Verwendung dieses spezialisierten Ports des Spartan-3A DSP FPGA's wird eine Weiterleitung über Signalpfade mit minimalem Clock-Skew innerhalb des FPGA genutzt.

Die 24B8525A-Kamera besteht im Wesentlichen aus einem MT9V022-Bildsensor, der auf einem Board zusammen mit Elementen zur stabilisierten Spannungsversorgung, einer Fassung für die Aufnahme eines Objektivs und einem Oszillator installiert ist. Der Oszillator stellt die Taktrate von 27 MHz für den Bildsensor bereit. [25, S. 4] Dies wurde durch Messungen am Oszilloskop bestätigt (siehe Abbildung 6.3). CH1 stellt das PIXCLK-Signal dar, es ist eine Taktrate von 27,022 MHz angegeben.

Um den Hardwarereset und -standby des Bildsensors aktivieren zu können, wurden die Reset- und Standby-Signale der Kamera mit Hilfe von I/O Ports angebunden. Da für das Reset-Signal ein Pull-Up Widerstand vorgeschlagen wird, wurde dieser für den entsprechenden Port mit dem Constraint PULLUP in der UCF-Datei aktiviert.

Die Signale für den I<sup>2</sup>C-Bus wurden wie in Kapitel 4.3 beschrieben angebunden.

Das durch den Oszillator generierte Taktsignal wird durch den Bildsensor um 180° phasenverschoben und als PIXCLK-Signal wieder herausgeführt. [4, S. 41] Da das PIXCLK-Signal im FPGA zum Betrieb der Bildverarbeitungspipeline verwendet werden soll, wird es über den dedizierten Takteingang der SAM-Pfostensteckerleiste in den FPGA hineingeführt. Das PIXCLK-Signal stellt zwar das Taktsignal für die parallelen Pixeldaten des Bildsensors dar, jedoch lagen am FPGA zu den steigenden Taktflanken teilweise noch keine validen Pixeldaten an. Aus vorliegenden

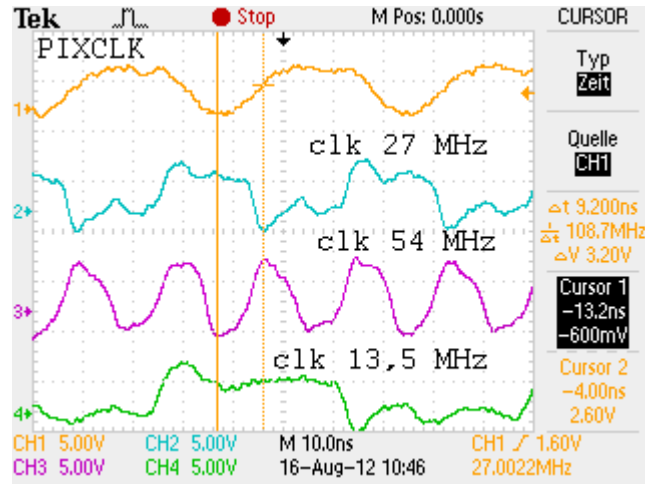


Abbildung 6.3.: Oszillogramm der generierten Takte; PIXCLK des MT9V022-Bildsensors; generierte 27 MHz, 54 MHz und 13,5 MHz Taktsignale (Tektronix TDS 2024C; 200 MHz; 2 GS/s)

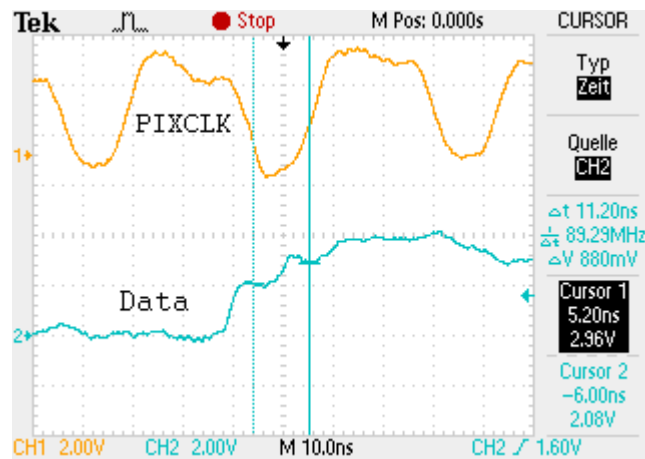


Abbildung 6.4.: Oszillogramm der Datenausgabe des MT9V022-Bildsensors; PIXCLK- und Datensignal (Tektronix TDS 2024C; 200 MHz; 2 GS/s)

FPGA		Mol.		Pfoften-Leiste				Mol.		FPGA	
SAM	PAD	#	#	Bezeichnung	Bezeichnung	#	#	PAD	SAM		
D12	Y25	3	1	DOUT0	DOUT1	2	5	Y24	D11		
D10	T20	7	3	DOUT2	DOUT3	4	9	U21	D9		
D8	Y23	11	5	DOUT4	DOUT5	6	13	Y22	D7		
D6	U18	15	7	DOUT6	DOUT7	8	17	U19	D5		
D4	AA25	4	9	DOUT8	DOUT9	10	6	AA24	D3		
D2	U20	20	11	FRAME	LINE	12	18	V21	D1		
N/C	N/C	19	13	SYSCLK	PIX_CLK	14	10	AA14	CLK		
D14	T17	12	15	STANDBY	RESET	16	8	T18	D13		
N/C	N/C	14	17	SER_O+ (LVDS)	SER_O- (LVDS)	18	16	N/C	N/C		
N/C	N/C	25	19	ERROR	LED_OUT	20	23	N/C	N/C		
A2	AB23	28	21	I2C CLK	I2C DATA	22	27	AB26	A1		
N/C	N/C	24, 26	23	N/C	GND	24	1, 2, 21, 22	GND	GND		

Tabelle 6.1.: Pinout des Dateninterfaces der 24B8525A-Kamera

Oszillogrammen (siehe Abbildung 6.4) ist zu entnehmen, dass das Datensignal bei steigender Taktflanke des PIXCLK-Signals noch keinen endgültigen Wert angenommen hat. Dies führt bei einer Verwendung der Daten ohne ein Phasenverschieben des Taktsignals zu stellenweise gekippten Bits in den einzelnen Pixelwerten. Deshalb wurde das PIXCLK-Signal im FPGA durch einen DCM, der auch für das Bereitstellen der von der Bildverarbeitungskette benötigten Taktraten genutzt wird, um 90° phasenverschoben. Die durch den DCM generierten Taktsignale sind in Abbildung 6.3 dargestellt.

Sämtliche Signale des parallelen digitalen Datenausgangs des Bildsensors wurden über normale I/O Ports in den FPGA hineingeführt. Dazu zählen die 10 DOUT-Leitungen sowie die LINE\_VALID- und FRAME\_VALID-Leitungen. Eine Masse-Verbindung wurde hergestellt.

Daraus ergibt sich das in Tabelle 6.1 dargestellt Pinout (siehe auch Abbildung 6.5).

## 6.2. Timing nach VGA-Standard

Mit dem VGA-Videosignal wurde ursprünglich eine einfache Schnittstelle zur Ansteuerung von Röhrenmonitoren geschaffen. Dabei wurde davon ausgegangen, dass ein Röhrenmonitor mit je einem Elektronenstrahl für rote, grüne und blaue Farbe existiert. Dieser wird mittels elektrische Magnete am Rand der Röhre durch die Lorentzkraft in horizontaler beziehungsweise vertikaler Richtung abgelenkt. Durch die Elektronik des Röhrenmonitors werden die Elektronenstrahlen Zeile für Zeile von links oben nach rechts unten über eine fluoreszierende Mattscheibe geführt. Trifft ein Elektronenstrahl auf die Mattscheibe, so beginnt sie für den Betrachter sichtbar zu



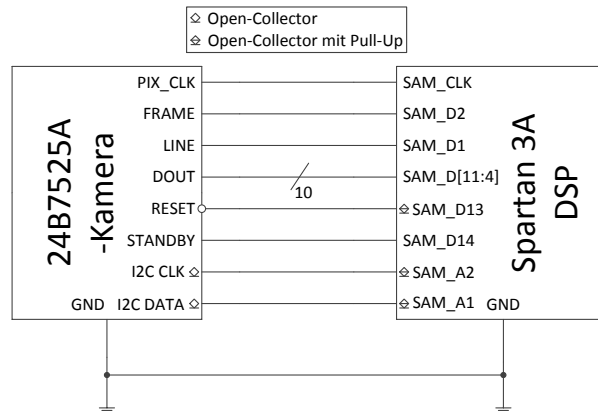


Abbildung 6.5.: Anbindung der Kamera an den FPGA

leuchten. Nach jeder Zeile findet ein Zeilenrücklauf statt, der die Elektronenstrahlen wieder an den linken Bildrand bewegt. Nach jedem kompletten Bild erfolgt ein Rücklauf von unten nach oben. [11]

Das VGA-Videosignal definiert 5 Signalleitungen. Drei zum analogen Übertragen der Farbwerte für die einzelnen Elektronenstrahlen und jeweils eines zum Signalisieren eines Rücklaufes in horizontaler beziehungsweise vertikaler Richtung. Diese beiden Signale beinhalten also die Synchronisationssignale und werden mit HSYNC und VSYNC bezeichnet. Eine gewisse Zeit vor dem Anfang jeden Bildes und jeder Zeile fällt dabei das entsprechende Synchronisationssignal für eine bestimmte Zeit auf den Low-Pegel. Danach steigt es bis zum nächsten Puls wieder auf den High-Pegel an, es handelt sich also um ein Active-Low-Signal. [11]

Die 5 Signalleitungen werden über einen 15-poligen SUB-D Stecker verbunden. Zu jedem der analogen Signale führt dieser ein eigenes Massesignal und ferner ein Massesignal für die digitalen Signale. [11]

Zum Generieren eines funktionierenden VGA-Signals muss weiterhin ein bestimmtes Timing eingehalten werden. Dies unterscheidet sich je nach Auflösung und Bildfrequenz. Die vorgegebene Auflösung von 480 x 640 Pixeln entspricht der VGA-Standardauflösung. Sie sieht weiterhin eine Bildwiederholfrequenz von etwa 60 Hz vor. Definiert sind auch die Zeiten, wie lang die Synchronisationspulse für die horizontalen und vertikalen Synchronisationssignale sein müssen. Weiterhin ist die Zeit zwischen zwei Synchronisationspulsen definiert. Der Zeitraum, in dem Bilddaten dargestellt werden, ist relativ zu den Synchronisationspulsen definiert. Daraus ergeben sich die Synchronisationssignale nach dem in Abbildung 6.6 abgebildeten Schema. Diese Zeiten

## 6. Kamera-Integration in das System-on-Chip basierte Bildverarbeitungssystem

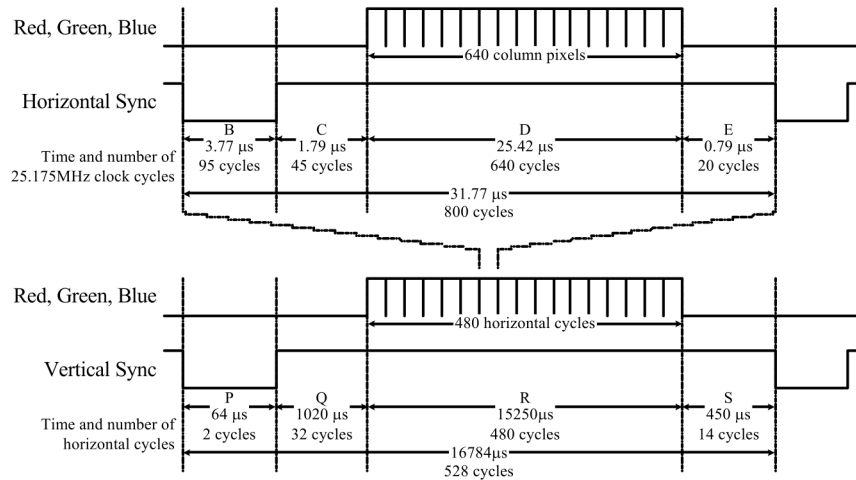


Abbildung 6.6.: Vertikale und horizontale VGA-Synchronisationssignale; Pulsbreite (B und P), Backporch (C und Q), Frontporch (E und S) und aktiver Bilddatenbereich (D und R) sind definiert [14]

	Puls	Backporch	Bilddaten	Frontporch	
	B	C	D	E	$\Sigma$
<b>Zeit</b>	3,77 $\mu$ s	1,79 $\mu$ s	25,42 $\mu$ s	0,79 $\mu$ s	31,77 $\mu$ s
<b>Takte</b>	95	45	640	20	800
	P	Q	R	S	$\Sigma$
<b>Zeit</b>	64 $\mu$ s	1020 $\mu$ s	15250 $\mu$ s	450 $\mu$ s	16784 $\mu$ s
<b>Zeilen</b>	2	32	480	14	528
<b>Takte</b>	1600	25600	384000	11200	422400

Tabelle 6.2.: Vorgabe VGA-Timinggrößen (25,175 MHz) [14]

sind in Tabelle 6.2 aufgelistet. [14]

Daraus folgt die vorgesehene Pixelfrequenz:

$$f_{Pixel} = \frac{N_{PixelproZeile}}{D} = \frac{640}{25,42\mu s} = 25,175 MHz \quad (6.1)$$

Da die Pixeldaten von der Kamera jedoch mit einer Pixelfrequenz von 27 MHz geliefert werden, bedarf es weiterer Anpassungen. Eine Option wäre das Puffern des Bildes in einem Speicher. Der Speicherbedarf ergibt sich aus der Gesamtzahl der Pixel und der benötigten Speichertiefe. Um ein Bild zu puffern werden entsprechend 3072000 Bit beziehungsweise 384000 Byte benötigt. Dies würde als BlockRAM realisiert einen wesentlichen Anteil der auf dem FPGA verfügbaren Ressourcen verbrauchen. Alternativ ließe sich der Puffer im angebundenen DDR2-SDRAM

	<b>Puls</b>	<b>Backporch</b>	<b>Bilddaten</b>	<b>Frontporch</b>	
	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	$\Sigma$
<b>Zeit</b>	3,85 $\mu s$	1,93 $\mu s$	23,7 $\mu s$	2,52 $\mu s$	32 $\mu s$
<b>Takte</b>	104	52	640	68	864
	<b>P</b>	<b>Q</b>	<b>R</b>	<b>S</b>	$\Sigma$
<b>Zeit</b>	64 $\mu s$	928 $\mu s$	15360 $\mu s$	352 $\mu s$	16704 $\mu s$
<b>Zeilen</b>	2	29	480	11	522
<b>Takte</b>	1728	25056	414720	9504 <sup>1</sup>	451008

Tabelle 6.3.: Angepasste VGA-Timinggrößen (27 MHz)

ablegen. Dieser wird jedoch zum Einen schon durch das Dual-MicroBlaze-System genutzt und die Nutzung als Bildpuffer würde dessen Performance bei den Speicherzugriffen merklich reduzieren. Zum Anderen lässt sich aufgrund der konkurrierenden Zugriffe des Dual-MicroBlaze-Systems keine Bandbreite für das Schreiben und Lesen der Pufferdaten garantieren. Hinzu kommt der umfangreiche Entwicklungsaufwand für eine Schaltung dieser Art, da beispielsweise ein Zugriff auf die Interfaces des Speichercontrollers in simultan lesender und schreibender Funktion stattfinden muss.

Eine andere Option stellt die Beibehaltung des vorhandenen Pixeltaktes bei Anpassung der Zählerwerte dar. Dabei wird die Dauer einer Zeile und eines Bildes nicht verändert. Lediglich die Zeit, in der aktive Pixel in einer Zeile übermittelt werden, wird verkürzt. Dementsprechend ergibt sich das in Tabelle 6.3 aufgeführte Timing.

### 6.3. Einhalten des Timings auf Kameraseite

Um das Timing zur erfolgreichen Generierung eines VGA-Videosignals einhalten zu können, muss seitens der Kamera ein Pixelstrom mit entsprechend großen Blanking-Feldern generiert werden. Die Bildgröße ist mit 480 x 640 Pixeln vorgegeben. Die Startwerte für das ausgegebene Bildfenster des Bildsensors beeinflussen das Timing nicht. Sie sind, solange das Bildfenster im Bereich existierender Pixel bleibt, frei wählbar. Für das Timing der Kameraausgabe ergibt sich laut Dokumentation folgendes Verhältnis [4, S. 13 f.]:

$$\begin{aligned}
 t_{Zeile} &= t_{Fensterbreite} + t_{hor.Blanking} \\
 t_{Bild} &= t_{Zeile} * (N_{Fensterhöhe} + N_{vert.Blanking}) + \frac{4}{f_{Pixelfrequenz}}
 \end{aligned}
 \tag{6.2}$$

<sup>1</sup>Dieser Wert vergrößert sich, wie im nächsten Abschnitt beschrieben, um 4.

Daraus ergibt sich ein horizontales Blanking von 224 Pixeln und ein vertikales Blanking von 42 Zeilen.

Weiterhin ist in der Dokumentation vermerkt, dass das Frame-Valid Signal am Bildende immer 23 Pixeltakte nach dem Line-Valid Signal von High- auf Low-Pegel fällt. Dementsprechend geht das Frame-Valid Signal am Bildanfang  $N_{\text{Fensterbreite}} - 23$  Pixeltakte vor dem Line-Valid Signal von Low- auf High-Pegel.

Da die Zeit eines Bildes, wie aus Gleichung 6.2 hervorgeht, immer um 4 Pixeltakte länger als der durch die Angabe der Bildgröße und des Blankings erwartete Wert ist, kann das VGA-Timing nicht exakt eingehalten werden. Dies lässt sich auch nicht durch eine Berücksichtigung in den Werten ausgleichen, da das vertikale Blanking in Zeilen angegeben wird. Dementsprechend verlängert sich die im vorherigen Abschnitt ermittelte Frontporch-Zeit um 4 Pixeltakte.

### 6.4. Kamera-spezifische Parametrisierung des VGA-Interfaces

Zur Erzeugung der benötigten Synchronisationssignale für das VGA-Videosignal wurde eine vorhandene VGA-Controller Entity aus der Arbeit von Ramin Jeyrani-Mameghani modifiziert. [15] Dieser ist über Generics weitgehend an das vorhandene Synchronisationssignal im Frame-/Linevalid-Format anpassbar. Es wird dabei jeweils für die Generierung des horizontalen und vertikalen Synchronisationssignals die Länge des Frontporches, die Länge des Backporches und die Pulsbreite angegeben. Zusätzlich lässt sich eine nicht gleichzeitige Flanke des Frame- und Linevalid-Signals ausgleichen. Die Werte für das horizontale Synchronisationssignal werden in einzelnen Pixeltakten und die für das vertikale Synchronisationssignal werden in Zeilen angegeben. Dementsprechend benötigt der VGA-Controller noch die Information, aus wie vielen Pixeltakten sich eine Zeile zusammensetzt. [15]

Da die VGA-Synchronisationssignale jeweils vor dem Beginn der dazugehörigen Frame-/Linevalid-Signale ausgegeben werden, werden sie relativ zu den fallenden Flanken der Frame-/Linevalid-Signale generiert. Die fallenden Flanken werden jeweils durch einen Zustandsautomaten erkannt und ein Enable für den entsprechenden Zähler aktiviert. Der jeweilige Zähler beginnt dann von 0 an zu zählen. Nach der konfigurierten Frontporch-Zeit sorgen zwei Komparatoren dafür, dass beim dazugehörigen Synchronisationssignal der Synchronisationspuls beginnt und nach der Pulsbreite wieder endet. Für den Zähler des vertikalen Synchronisationssignales wird ein zusätzliches Enable durch einen weiteren Zähler mit einem Komparator einmal pro Zeile generiert. Im Zeitraum des Vertikalen Blankings wird das horizontale Synchronisationssignal weiter aufrecht erhalten. Vor der Ausgabe werden die Signale inklusive der durchgereichten Farbsignale mit einer Registerkette abgetaktet. [15]

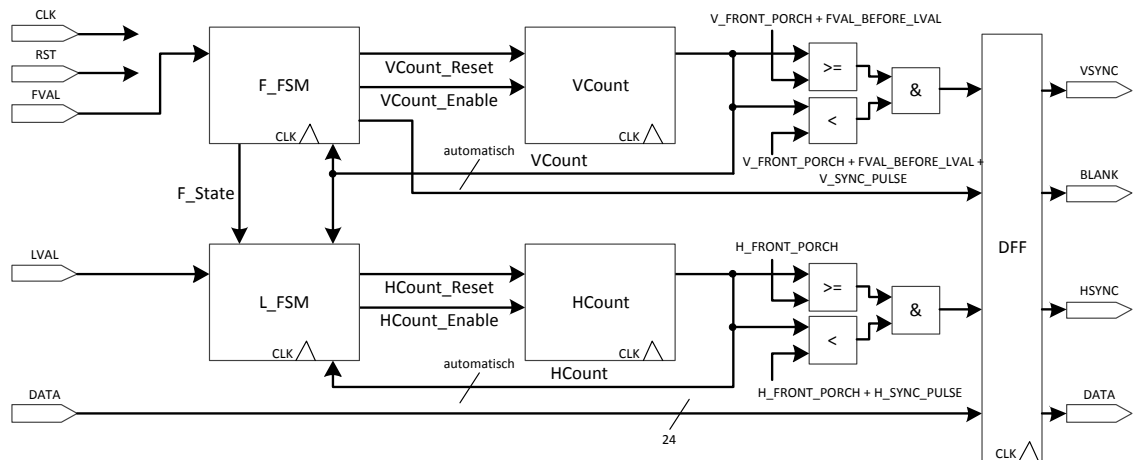


Abbildung 6.7.: Interner Aufbau VGA-Controller

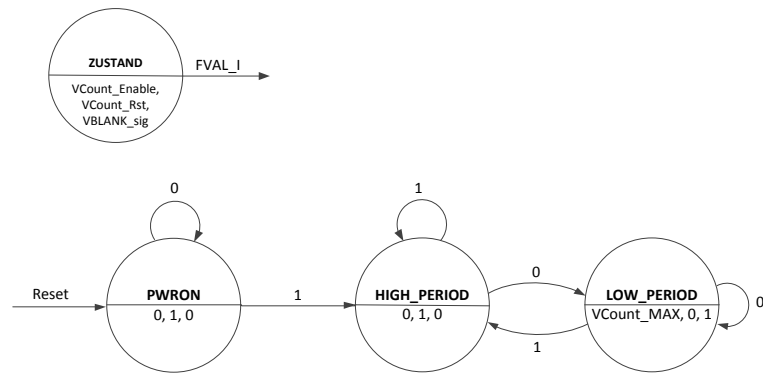
Der von Jeyrani-Mameghani erstellte VGA-Controller lässt entsprechend nicht zu, die Angaben zur Erzeugung der vertikalen Synchronisationssignale pixeltaktgenau anzugeben. Dies wird jedoch aufgrund des durch die Kamera vorgegebenen Timings benötigt. So ist beispielsweise die in den vorherigen Abschnitten errechnete Frontporch-Zeit nicht restlos durch die Gesamttaktzahl der Pixel pro Zeile teilbar. Dies resultiert aus dem in Gleichung 6.2 addierten Wert von 4 Pixeltakten. Hinzu kommt der ebenfalls im vorherigen Abschnitt aufgeführte Versatz der Frame- und Linevalid-Signale von 23 Pixeltakten zueinander. Da dieser auch nur in Zeilen angegeben werden kann, wurde der VGA-Controller entsprechend angepasst.

Der VGA-Controller wurde auf die Angabe der vertikalen Synchronisationssignale in Zeilen umgestellt. Dementsprechend fällt der Frequenzteiler für den vertikalen Zähler weg und der vertikale Zähler wird entsprechend breiter. Die aktuellen Generics und Ports sind in Tabelle 6.4 zusammen mit den für das vorgesehene Timing benötigten Werte aufgelistet. Der interne Aufbau des VGA-Controllers ist in Abbildung 6.7 dargestellt. Die Zustandsautomaten zum Erkennen der fallenden Flanken sind in Abbildung 6.8 genauer dargestellt.

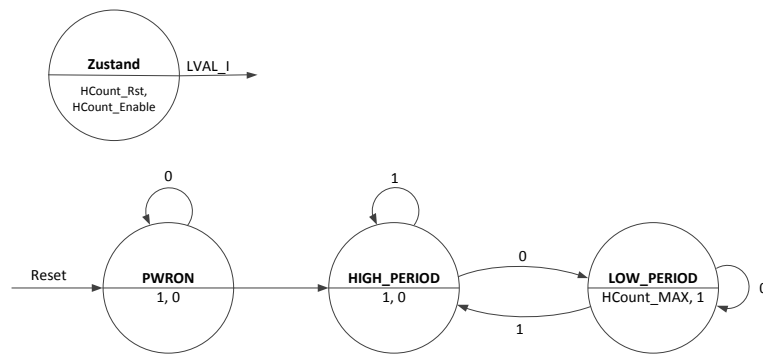
Zur Verifikation des VGA-Controllers durch eine Simulation wurde eine Testbench mit Stimuligeneratoren für zwei Testfälle erstellt. Dabei wird der Einfachheit halber ein Synchronisationssignal für ein Bild einer Größe von 64 x 48 Pixeln mit einem Blanking von 16 Pixeln beziehungsweise 4 Zeilen generiert. Im ersten Testfall werden das Frame- und Linevalid-Signal ohne Versatz und im zweiten Testfall mit 10 Pixeltakten Versatz erzeugt. Durch eine TCL-Skriptdatei wird ein Reset-Signal und ein Takt mit einer 10 ns Periode generiert. Das horizontale Synchronisationssi-

Name	Wert/Typ	Beschreibung
<b>Generics</b>		
H_PIXEL	640 Pixeltakte	Pixel pro Zeile
H_FRONT_PORCH	68 Pixeltakte	Länge Frontporch
H_SYNC_PULSE	104 Pixeltakte	Breite Synchronisationspuls
H_BACK_PORCH	52 Pixeltakte	Länge Backporch
H_SYNC_POLARITY	'0' → Active High	Active High oder Active Low
FVAL_BEFORE_LVAL	201 Pixeltakte	Versatz Frame-/Linevalid
V_FRONT_PORCH	9508 Pixeltakte	Länge Frontporch
V_SYNC_PULSE	1728 Pixeltakte	Breite Synchronisationspuls
V_BACK_PORCH	25056 Pixeltakte	Länge Backporch
V_SYNC_POLARITY	'0' → Active High	Active High oder Active Low
<b>Ports</b>		
CLK_I	in std_logic	Pixeltakteingang
RST_I	in std_logic	asynchroner Reset
LVAL_I	in std_logic	Eingang Linevalid
FVAL_I	in std_logic	Eingang Framevalid
RED_I	in std_logicvector(7 downto 0)	Dateneingang Rot
GRN_I	in std_logicvector(7 downto 0)	Dateneingang Grün
BLU_I	in std_logicvector(7 downto 0)	Dateneingang Blau
RED_O	out std_logicvector(7 downto 0)	Datenausgang Rot
GRN_O	out std_logicvector(7 downto 0)	Datenausgang Grün
BLU_O	out std_logicvector(7 downto 0)	Datenausgang Blau
BLANK_O	out std_logic	Ausgang Blanking-Signal
HSYNC_O	out std_logic	Ausgang hor. Synchronisation
VSYNC_O	out std_logic	Ausgang vert. Synchronisation

Tabelle 6.4.: Ports und Generics der VGA-Controller Entity



(a) Zustandsautomat für Framevalid-Signal



(b) Zustandsautomat für Linevalid-Signal

Abbildung 6.8.: Zustandsautomaten des VGA-Controllers [15]

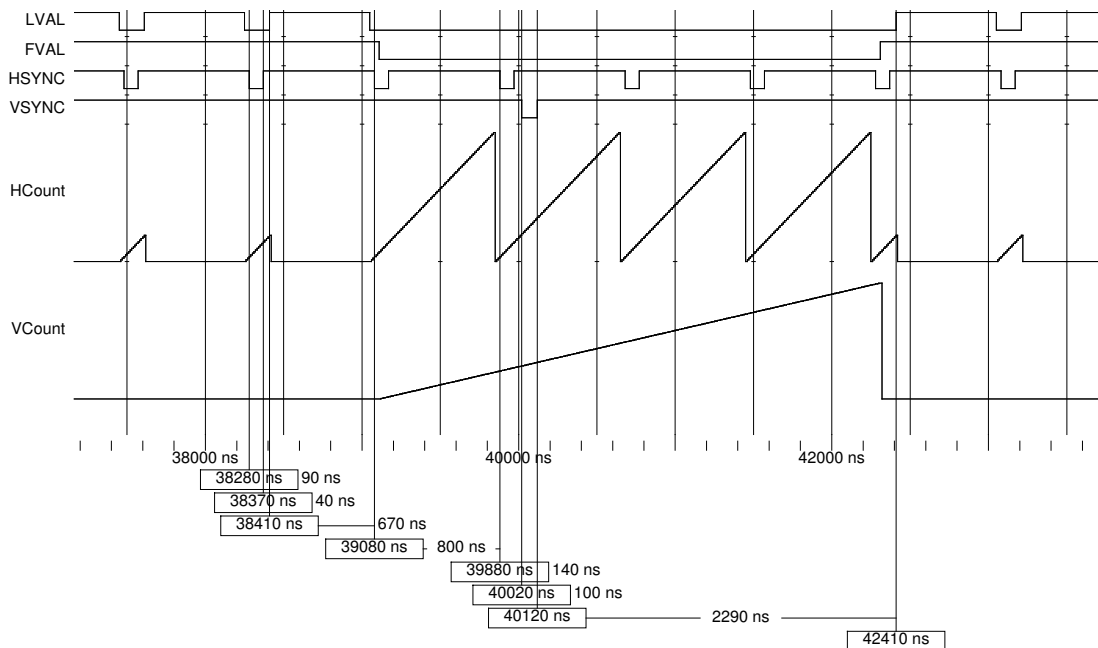


Abbildung 6.9.: Simulationsergebnis der Testbench des VGA-Controllers

gnal soll mit einer Breite von 9 Takten und einem Backporch von 5 Takten erzeugt werden. Das vertikale Synchronisationssignal soll mit einer Breite von 10 Takten und einem Backporch von 230 Takten erzeugt werden. Der VGA-Controller wird jeweils mit entsprechenden Generics instanziiert. Daraus ergibt sich das in Abbildung 6.9 als Ausschnitt abgebildete Simulationsergebnis. Abgebildet ist das Ende des ersten und der Anfang des zweiten Bildes. Die ersten drei markierten Stellen auf der Zeitskala zeigen die Pulsweite und den Backporch des HSYNC-Signales. Gemessen wurden für die Pulsweite 90 ns, also 9 Takte, und für den Backporch 40 ns, also 4 Takte. Die beiden darauf folgenden markierten Stellen zeigen die Periode des HSYNC-Signales während der Blankingzeit. Hier wurden 800 ns, also 80 Takte, gemessen. Eine Zeile hat eine Länge von  $64 + 16 = 80$  Pixeltakten. Die letzten drei markierten Stellen zeigen die Pulsweite und den Backporch des VSYNC-Signales. Für das VSYNC-Signal muss der Backporch vom Ende des VSYNC-Pulses bis zum Anfang der Übertragung der Bilddaten, also bis zur ersten LVAL-Flanke gemessen werden. Gemessen wurde eine Pulsbreite von 100 ns, also 10 Takten, und ein Backporch von 2290 ns, also 229 Takten. Dabei ist zu beachten, dass die HSYNC- und VSYNC-Signale im Vergleich zu den LVAL- und FVAL-Signalen um einen Takt verzögert und die Messungen von den HSYNC-/VSYNC-Synchronisationssignalen zu den FVAL-/LVAL-Synchronisationssignalen um 10 ns kürzer sind. Dementsprechend ergeben sich für die Backporches 5 beziehungsweise 230 Takte.



Die geforderten Werte sind somit eingehalten worden und die Verifikation des generierten Signals ist erfolgreich abgeschlossen.

Die Umsetzung der Grauwerte in RGB-Werte findet vor der Übergabe an den VGA-Controller statt. Hierfür werden die obersten 4 Bit des Grauwertes jeweils an die einzelnen Farbwerte des RGB-Signals übergeben. Das analoge VGA-Signal wird über ein auf dem FPGA-Board vorhandenes Widerstandsnetz erzeugt [32, S. 21].

Der Quellcode jeweils für die Testbench und den VGA-Controller ist dem Anhang A.3 zu entnehmen.

### 6.5. Anpassung eines FrameGrabbers an die Pixelrate der Kamera

Der ursprüngliche FrameGrabber wird mit einer Pixelrate von 13,5 MHz bei 8 Bit pro Pixel betrieben. In diesen 8 Bit sind jeweils 3 Bit für den roten, 3 Bit für den grünen und 2 Bit für den blauen Wert untergebracht. Die Daten werden durch ein Schieberegister zu 32 Bit breiten Werten zusammengesetzt, die über den FSL-FIFO an den angeschlossenen MicroBlaze übergeben werden. Um eine Übertragung zu starten, wird ein beliebiger Wert durch den angeschlossenen MicroBlaze über den FSL-FIFO an den FrameGrabber übergeben. Dadurch wird im FrameGrabber ein Zustandsautomat ausgelöst, der den durch den MicroBlaze übermittelten Wert aus dem FSL-FIFO entnimmt und den Anfang eines neuen Bildes abwartet. Sobald der Anfang des nächsten Bildes erkannt wird, wird mit dem Übermitteln der Daten begonnen. Der MicroBlaze wartet auf den Beginn der Übermittlung der Daten und legt diese im angebundenen DDR2-SDRAM ab.

Da durch die neue Kamera die Pixelfrequenz und somit die Datenrate verdoppelt wird, ist der MicroBlaze nicht mehr in der Lage, die Daten schnell genug im Speicher abzulegen. Dadurch kommt es bei der ursprünglichen Konfiguration des FrameGrabbers zum Überlauf des FSL-FIFO, wodurch ein Teil der Daten verloren gehen. Um dies zu umgehen gibt es mehrere Optionen. Zum Einen lassen sich die Daten über eine DMA-Schnittstelle (Direct Memory Access) direkt im Speicher ablegen. Dies hätte vor allem zum Vorteil, dass die vergleichsweise relativ langsame Anbindung der Caches des MicroBlaze über das XCL-Interface (Xilinx CacheLink) umgangen wird. Diese hat zwar auf den ersten Blick mit einer Datenrate von bis zu 225 MBytes/s [27, S. 219] eine mehr als ausreichende Bandbreite. Jedoch muss der MicroBlaze nicht nur Speicherzugriffe durchführen, sondern muss beispielsweise auch noch Befehle zur Adressberechnung und zum Durchlaufen einer Schleife abarbeiten. Durch eine Anbindung einer DMA-Schnittstelle über ein NPI (Native Port Interface) des Speichercontrollers lassen sich Datenraten von bis zu 813 MByte/s realisieren. [27, S. 213] Bei der Verwendung dieses Interfaces besteht also auch bei einem konkurrierenden Zugriff von MicroBlaze und FrameGrabber-DMA-Schnittstelle ein weiter

Spielraum. Allerdings ergibt sich aus dem Implementieren einer DMA-Schnittstelle ein massiver Entwicklungsaufwand, der im Rahmen dieser Arbeit nicht umzusetzen war. Zudem benötigt diese Option zur Konfiguration einer Zieladresse im Speicher eine Anbindung an den PLB. Daraus resultiert dementsprechend ein wesentlich größerer Umfang an benötigten Logikressourcen des FPGA, der in Bezug auf die nur geringen Nachteile der im Folgenden beschriebenen Optionen nicht zu rechtfertigen ist.

Eine weitere Option stellt eine hardwareseitige Komprimierung der Daten vor der Übermittlung an den MicroBlaze dar. Dafür ist es nötig, ein in Hardware effizient zu realisierendes Komprimierungsverfahren zu implementieren. Komprimierungsverfahren wie RLE (Run-Length Encoding) sind zwar einfach zu realisieren, kommen jedoch nicht in Frage, da sie beispielsweise bereits bei einem leichten Bildrauschen im unteren Wertebereich nicht mehr in der Lage sind, die Datenrate zu reduzieren. Dies ist durch das Komprimieren folgender 8 Bit Pixeldaten zu zeigen:

1; 2; 1; 3; 2; 2; 3; 2; 3; 4; 3; 4; 4; 5; 4; 3; 5

Daraus wird:

1 x 1; 1 x 2; 1 x 1; 1 x 3; 2 x 2; 1 x 3; 1 x 2; 1 x 3; 1 x 4; 1 x 3; 2 x 4; 1 x 5; 1 x 4; 1 x 3; 1 x 5

Aus 17 Byte werden durch ein RLE 30 Byte. Da es gerade bei CMOS-Bildsensoren aufgrund der Architektur oftmals zu einem Bildrauschen im unteren Wertebereich kommt, würde ein RLE nicht zu einer zuverlässigen Verringerung der Datenrate führen. Die Ermittlung und Implementierung eines geeigneten Komprimierungsverfahrens stellt weiterhin einen Zeitaufwand dar, der im Rahmen dieser Arbeit nicht umzusetzen war. Zusätzlich ist auch hier der Verbrauch an Logikressourcen im Bezug auf die nur geringen Nachteile der folgenden Option kaum zu rechtfertigen.

Die präferierte Option stellt eine simple Reduktion der Datenrate durch ein Verkleinern der Bitbreite der einzelnen Werte dar. Schon das Verwenden von 5 x 6 Bit Werten pro FIFO-Slot des FSL reicht aus, um die benötigte Datenrate, im Vergleich zu 4 x 8 Bit Werten pro FIFO-Slot, um 20% zu reduzieren. Diese Reduktion verringert die Anzahl der darstellbaren Werte zwar um 75%, jedoch sind die Bilddaten dabei durchaus noch zu Dokumentations- und Analysezwecken verwendbar.

Die Daten werden mit Hilfe einer vierstufigen Schieberegisterkette zusammengeführt und mit zwei weiteren Bits zum Auffüllen und dem aktuell anliegenden Datum an die Eingangsdatenleitung des FSL gelegt. Ein Zustandsautomat steuert dabei das Erkennen einer Anforderung von Daten und das Erfassen und Übermitteln der Daten. Der Zustandsautomat verfügt über 4 Zustände. Den Zustand `Idle`, in dem er auf eine Datenanforderung auf den eingehenden FSL wartet und diese bei einer anliegenden Datenübertragung auf dem eingehenden FSL liest und in den Zustand `WaitForVBlank` über geht. Dieser wartet auf das Ende des aktuellen Bildes und leitet in

den Zustand `WaitForFrame` über. `WaitForFrame` wartet auf den Beginn des nächsten Bildes und leitet in den Zustand `Transmit` über. In diesem werden die Daten, durch einen Zähler gesteuert, alle 5 Pixeltakte in den FSL geschrieben. Nach dem Ende einer Zeile wird das Schieberegister komplett gefüllt, damit keine Daten verloren gehen. Danach werden sie in den FSL geschrieben. Wenn die letzte Zeile und somit das Bild zu Ende ist, findet ein Übergang in den Zustand `Idle` statt.

Durch den angeschlossenen MicroBlaze wird bei Bedarf eine Übertragung durch das Schreiben eines Wertes in den FSL zum FrameGrabber gestartet. Daraufhin liest der MicroBlaze die Daten mit Hilfe der blockierenden Lesebefehle aus dem FSL. Wird nicht blockierend gelesen, so droht ein Unterlaufen des FIFO-Puffers. Vor dem Starten der Übertragung sollte der FSL von eventuellen im FIFO-Puffer verbliebenen Daten bereinigt werden. Beim Empfang der Daten auf der Seite des MicroBlaze ist es von essentieller Bedeutung, die Daten effizient und performant im Speicher abzulegen. Zu diesem Zweck werden aktivierte Daten- und Instruktionscaches benötigt. Eine Größe von 8 KByte hat sich bei einer Taktfrequenz von 62,5 MHz des MicroBlaze als ausreichend erwiesen. Hinzu kommt, dass ein wesentlicher Anteil der Instruktionen dem Lesen aus dem FSL und dem Schreiben in den Speicher zugeteilt werden sollte. Die Berechnung der aktuellen Speicheradresse und das Prüfen der Abbruchbedingung der Schleife sollte wenig Instruktionen beanspruchen. Zudem sollten die benötigten Instruktionen komplett im Instruktionscache abzulegen sein. Folgender Quellcode hat sich als performant genug erwiesen:

```
1  while ((u32) buff < stopAddr){
2      getfslx(*buff, 1, FSL_DEFAULT);
3      buff++;
4      getfslx(*buff, 1, FSL_DEFAULT);
5      buff++;
6      getfslx(*buff, 1, FSL_DEFAULT);
7      buff++;
8      getfslx(*buff, 1, FSL_DEFAULT);
9      buff++;
10     getfslx(*buff, 1, FSL_DEFAULT);
11     buff++;
12     getfslx(*buff, 1, FSL_DEFAULT);
13     buff++;
14     getfslx(*buff, 1, FSL_DEFAULT);
15     buff++;
16     getfslx(*buff, 1, FSL_DEFAULT);
17     buff++;
18 }
```

Dabei werden pro Schleifendurchlauf jeweils 8 Lesezugriffe auf den FSL durchgeführt. Hierdurch wird der Overhead durch die Schleife minimalisiert. Die Variable `buff` wird mit dem Beginn und die Variable `stopAddr` mit dem Ende des Speicherbereiches initialisiert. Durch diese Vorgehensweise muss die Anzahl der zu lesenden FIFO-Slots restlos durch 8 teilbar sein. Das Makro `getfslx(*buff, 1, FSL_DEFAULT)` liest einen Wert aus dem FSL mit dem Index 1 in den Speicherbereich bei `*buff`. Die Konstante `FSL_DEFAULT` bedeutet, dass ein blockierender Zugriff stattfinden soll. [29, S. 7 ff.]

Nach dem Lesen werden die 6 Bit breiten Werte wieder zu 8 Bit ausgeweitet und gefolgt von einem gültigen BMP-Header auf der seriellen Schnittstelle ausgegeben. Die ausgegebenen Daten können dann direkt als Datei abgespeichert werden.

## 7. Analyse der durch die 24B7525A-Kamera gelieferten Bilddaten

Um die in dieser Arbeit erstellte Konfiguration zu bestätigen, werden im Folgenden von der Kamera aufgenommene Bilder analysiert. Dabei wird die Linsenverzeichnung der einzelnen Objektivs und die Auswirkungen des Aktivierens des nicht linearen Modus des Analog-Digital-Umsetzers betrachtet.

### 7.1. Linsenverzeichnung der Objektivs

Für die Kamera sind mehrere wechselbare Objektivs vorhanden. Sie verfügen über unterschiedliche Blickwinkel im Bereich von etwa  $90^\circ$  bis  $160^\circ$ . Sie sind in Tabelle 7.1 aufgelistet. Das letzte Objektiv ist nicht genau bekannt, der Winkel wurde aus den Testbildern abgeleitet. Die Linsen werden mit einem M12 Gewinde aufgeschraubt und eingestellt.

Um die Auswirkungen der Linsenverzeichnung nach der projektiven Transformation abschätzen zu können, wurde diese in Matlab integriert. Hierdurch lassen sich insbesondere Auswirkungen auf die in der Bildverarbeitungskette darauf folgende Hough-Transformation herleiten. Die Koordinaten für den jeweiligen Pixel des Zielbildes  $\begin{pmatrix} x_z \\ y_z \end{pmatrix}$  im Quellbild  $\begin{pmatrix} x_q \\ y_q \end{pmatrix}$  ergeben sich aus [16, S. 31]:

$$\begin{aligned} x_q &= \frac{b_{11}x_z + b_{12}y_z + b_{13}}{b_{31}x_z + b_{32}y_z + 1} \\ y_q &= \frac{b_{21}x_z + b_{22}y_z + b_{23}}{b_{31}x_z + b_{32}y_z + 1} \end{aligned} \tag{7.1}$$

Bezeichnung	Blickwinkel
32S2120N	$160^\circ$
32S2520N	$140^\circ$
DSL213A	$106^\circ$
Unbekannt	ca. $90^\circ$

Tabelle 7.1.: Liste der Kameraobjektivs

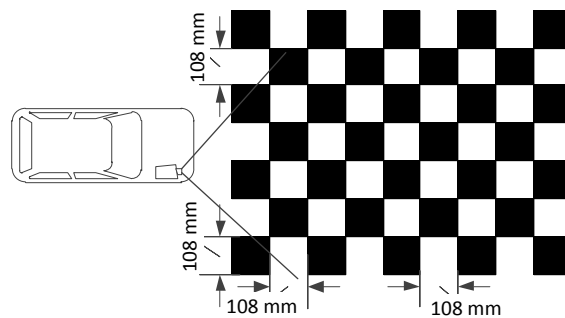


Abbildung 7.1.: Messaufbau zur Analyse der Linsenverzerrung

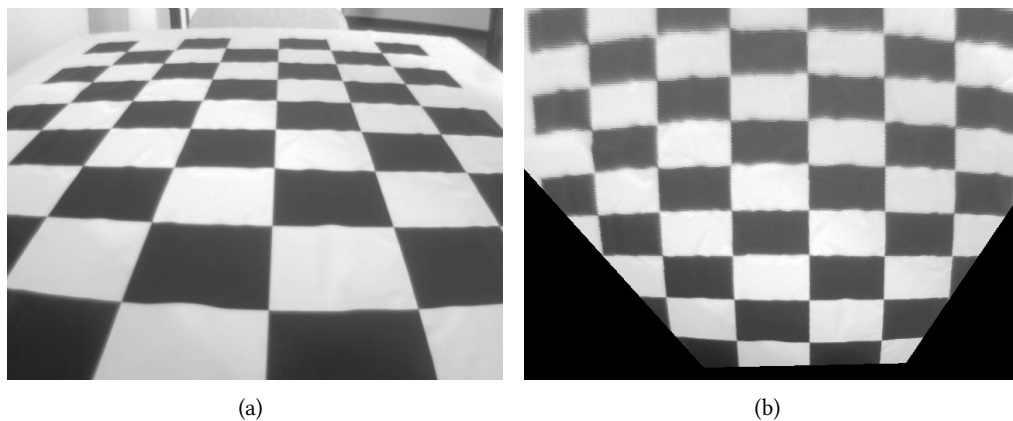


Abbildung 7.2.: Originalbild (a) und transformiertes Bild (b) des unbekannten Objektivs

Die Unbekannten  $b_n$  wurden mit einem linearen Gleichungssystem für jedes Objektiv ermittelt. Anzumerken ist, dass diese Parameter nicht in der Bildverarbeitungskette des Fahrzeugs zum Einsatz kommen können, da sie speziell für die Darstellungen in dieser Arbeit ermittelt wurden. Zur Aufnahme der Bilddaten wurde die Kamera auf dem Fahrzeug montiert und auf eine Fläche mit schwarzen und weißen Quadraten gerichtet. Dies ist in [Abbildung 7.1](#) skizziert.

Die [Abbildungen 7.2 bis 7.5](#) zeigen sowohl das original aufgenommene als auch das perspektivisch transformierte Bild der Kamera mit den unterschiedlichen Objektiven.

Das unbekannte Objektiv (siehe [Abbildung 7.2](#)) mit etwa  $90^\circ$  Blickwinkel zeigt kaum Verzerrungen in der Horizontalen. Lediglich im oberem Bereich des Bildes treten starke Krümmungen in der Vertikalen auf. Diese sind jedoch für die Hough-Transformation nicht relevant, da Fahrbahnmarkierungen aus Sicht der Kamera vorwiegend in horizontaler Richtung verlaufen.

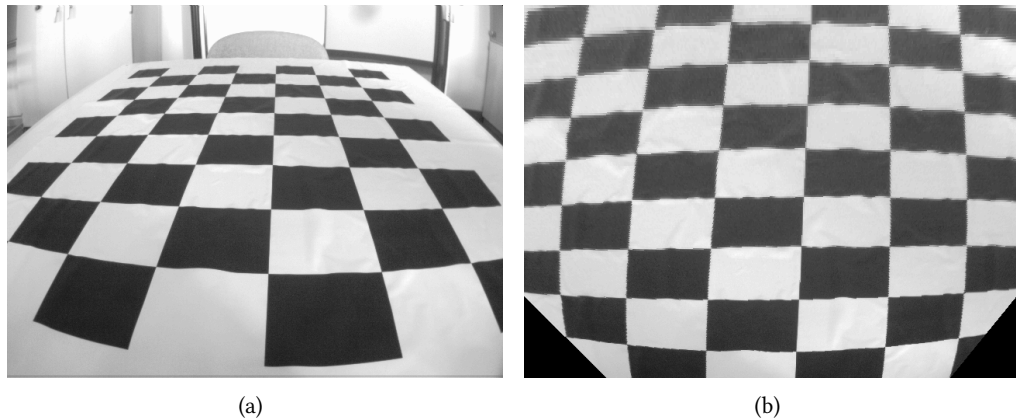


Abbildung 7.3.: Originalbild (a) und transformiertes Bild (b) des DSL213A Objektivs

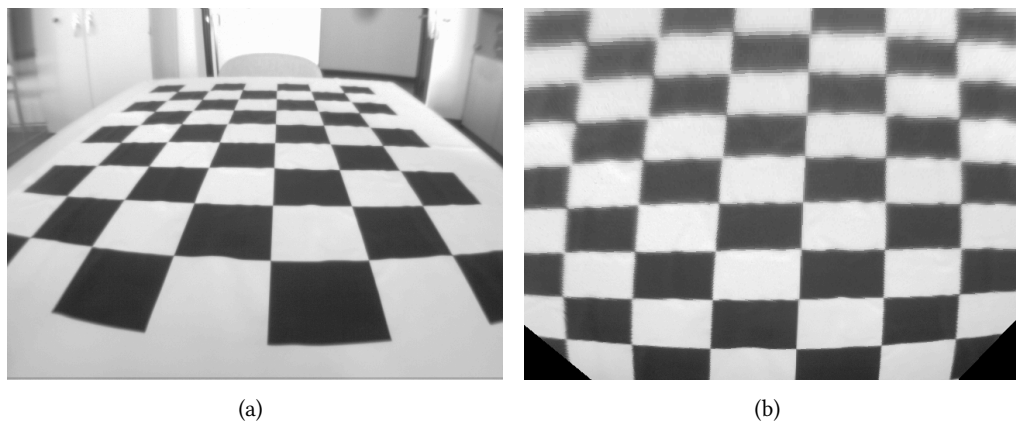


Abbildung 7.4.: Originalbild (a) und transformiertes Bild (b) des 32S2520N Objektivs

Das DSL213A Objektiv (siehe Abbildung 7.3) liefert im Vergleich zu den anderen Objektiven ein Bild mit sichtbar schärferen Kanten. Allerdings ist die Verzeichnung gerade in der unteren Bildhälfte in der Horizontalen trotz eines nur etwas weiteren Blickwinkels wesentlich größer. In den beiden untersten Reihe von Quadraten sind deutliche Winkelabweichungen der in der Realität gerade verlaufenden Linien zu erkennen. Auch in den darüber liegenden Reihen tritt dieser Effekt auf. Da sich die ROI vorrangig in diesem Bereich befindet, ist davon auszugehen, dass die Hough-Transformation Ergebnisse mit Winkelabweichungen liefert. Wie stark sich dies jedoch auf das Fahrverhalten auswirkt, ist gegebenenfalls in der Praxis genauer zu ermitteln. Das 32S2520N Objektiv (siehe Abbildung 7.4) zeigt trotz eines weiteren Blickwinkels in der Horizontalen eine etwas geringe Verzeichnung als das DSL213A Objektiv. In der Vertikalen ist

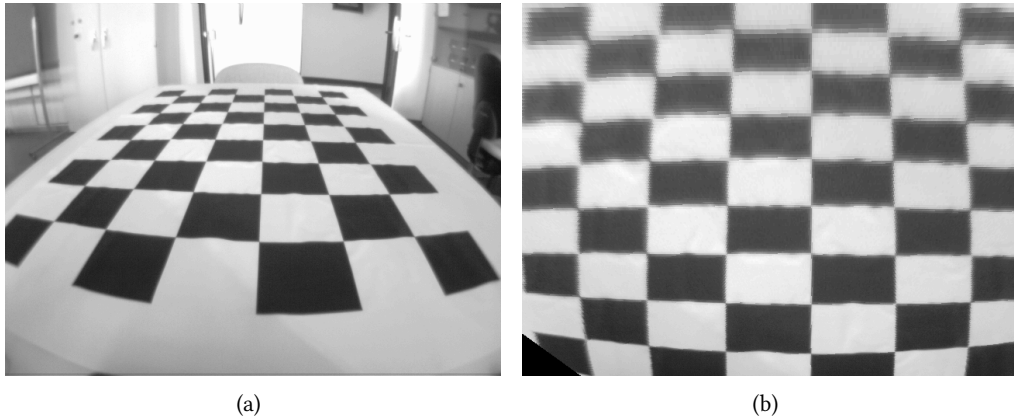


Abbildung 7.5.: Originalbild (a) und transformiertes Bild (b) des 32S2120N Objektivs

die Verzeichnung etwas stärker als bei den DSL213A Objektiv ausgeprägt. Dementsprechend ist auch bei diesem Objektiv mit einer Winkelabweichung der durch die Hough-Transformation ermittelten Werte zu rechnen. Die dritte bis sechste Reihe von unten zeigt in der Horizontalen nahezu gerade Kanten.

Das 32S2120N Objektiv (siehe Abbildung 7.5) zeigt mit dem weitesten Blickwinkel auch die stärkste Verzeichnung, sowohl vertikal als auch horizontal. Die horizontalen Kanten oberhalb der zwei untersten Reihen und unterhalb der obersten drei Reihen sind nahezu gerade.

Durch die Hough-Transformation wird ein Winkel im Intervall von  $-18^\circ$  bis  $18^\circ$  in  $2^\circ$  Schritten bestimmt [16, S. 84]. Eine Winkelabweichung wie sie beispielsweise in der untersten Reihe der Quadrate im Bild des 32S2120N Objektivs an den Rändern erkennbar ist, stellt unter diesen Umständen eine wesentliche Winkelabweichung dar. Zu erkennen ist weiterhin, dass es bei den weit winkelligen Objektiven 32S2120N und 32S2520N jeweils einen Bereich gibt, in dem die Kanten nahezu gerade verlaufen und somit die durch die Hough-Transformation ermittelten Winkel kaum Abweichungen aufweisen werden. Wird die ROI in den besagten Bereichen gehalten, so wird eine für das Objektiv minimale Winkelabweichung erreicht. Eine Verwendung der Objektive 32S2120N, 32S2520N sowie des unbekanntes Objektivs ist also unter Beachtung der genannten Bedingungen zu realisieren.



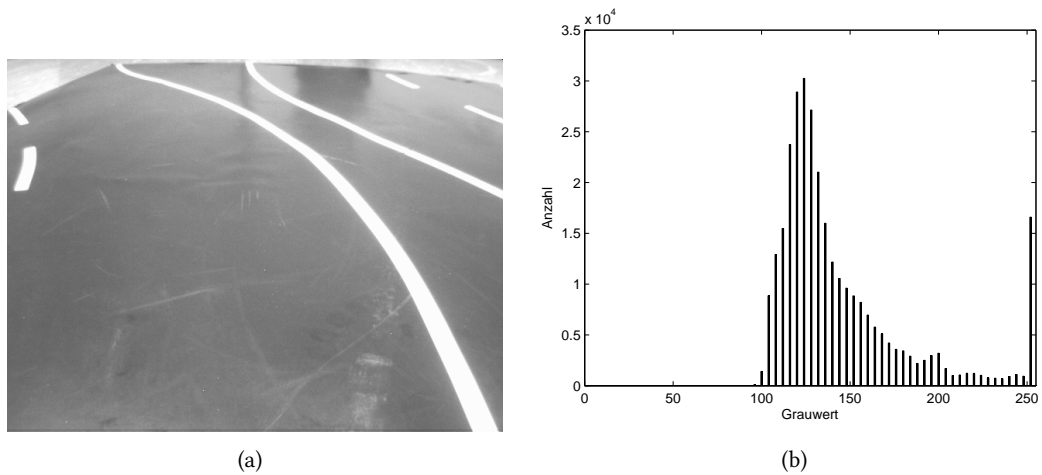


Abbildung 7.6.: Bild (a) und Histogramm (b) mit aktivierter nichtlinearer Quantisierung des ADCs im MT9V022 Bildsensor

## 7.2. Auswirkungen der nichtlinearen Digitalisierung des MT9V022 Bildsensors

Ein Aktivieren der nichtlinearen Digitalisierung des MT9V022 Bildsensors wirkt sich auf die adaptive Binarisierung der Bildverarbeitungspipeline aus. Für eine Binarisierung wird davon ausgegangen, dass das Histogramm des zu binarisierenden Bildes aus zwei Häufungen besteht. Zwischen den beiden Häufungen befindet sich der optimale Schwellwert. [18] Diese Häufungen sollten sich im Histogramm deutlich zeigen. Durch die nichtlineare Quantisierung des im MT9V022 Bildsensor vorhandenen Analog-Digital-Umsetzers wird das Histogramm im oberen Wertebereich gestaucht und im unteren Wertebereich gestreckt. Dementsprechend fallen die Häufungen im gestreckten Bereich schwächer ausgeprägt und im gestauchten Bereich stärker ausgeprägt aus.

Zum Ermitteln eines aussagekräftigen Histogramms wurde auf einer Fläche mit einer den Regularien des Carolo-Cups [9] entsprechenden Fahrspurmarkierung jeweils ein Bild mit Histogramm mit (Abbildung 7.6) und ohne (Abbildung 7.6) einer Aktivierung der nichtlinearen Quantisierung erstellt.

Bei aktivierter nichtlinearer Quantisierung des ADCs ist das Histogramm im oberen Wertebereich wie erwartet gestaucht. Zusätzliche Bildinformationen in den unteren Wertebereichen konnten nicht gewonnen werden. Dementsprechend sind die beiden Häufungen lediglich näher beieinander als bei einer deaktivierten nichtlinearer Quantisierung. Daraus lässt sich schlussfolgern,

## 7. Analyse der durch die 24B7525A-Kamera gelieferten Bilddaten

---

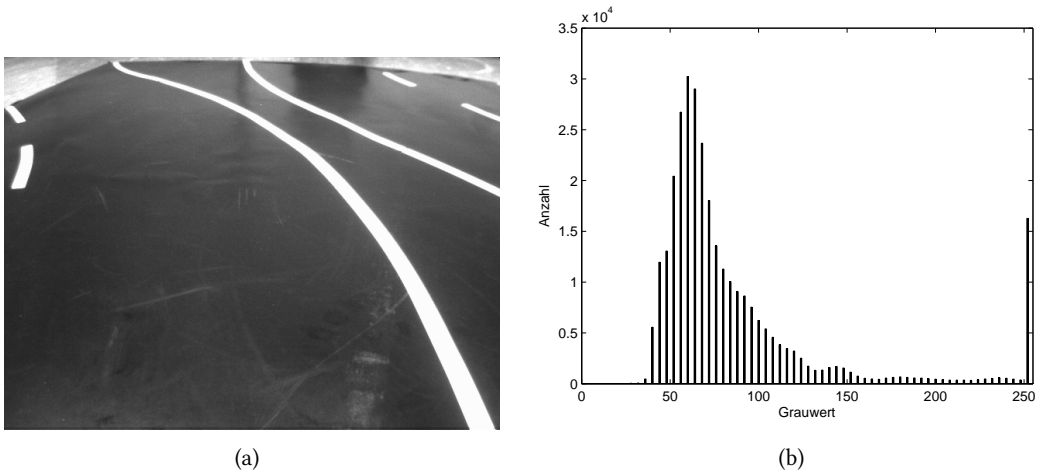


Abbildung 7.7.: Bild (a) und Histogramm (b) ohne aktivierter nichtlinearer Quantisierung des ADCs im MT9V022 Bildsensor

dass ein Aktivieren der nichtlinearen Quantisierung keine Verbesserung bei der Binarisierung, sondern eine Verschlechterung mit sich bringt.

Die Bilder wurden in einem Raum ohne Fenster und mit angeschalteter Deckenbeleuchtung aufgenommen. Wie sich die nichtlineare Quantisierung bei einer dunkleren Umgebung auswirkt, ist gegebenenfalls gesondert zu ermitteln.

## 8. Weitere Entwicklungsschritte

Für das in einer Reihe von Arbeiten entstandene Spurführungssystem sind noch weitere Entwicklungsschritte vorgesehen. Diese werden im Folgenden aufgeführt.

### **Anpassung der Bildverarbeitungspipeline an die 24B7525A-Kamera abschließen**

Wie schon in den vorherigen Kapiteln angemerkt, ist die Bildverarbeitungspipeline des Spurführungssystems noch nicht an die neue Kamera angepasst.

Da die Kamera nun 10 Bit breite anstelle von 8 Bit breiten Grauwerten liefert, muss die Binarisierung angepasst werden. Weiterhin muss diese auch an die neue Kamera angepasst werden, da das ausgegebene Bild sich von dem der davor verwendeten Kamera unterscheidet.

Es ist zu prüfen, ob durch die Verdopplung der Pixelfrequenz die Taktrate in den übertakteten Bereichen der Bildverarbeitungspipeline realisierbar ist. Da die Bildverarbeitungspipeline in einigen Bereichen um den Faktor 4 übertaktet ist, wird eine Taktrate von 108 MHz in einigen Bereichen benötigt.

Da die Bildausgabeelemente nun 10 Bit breite Grauwerte erwarten, muss der Result-Illustration Block der Bildverarbeitungspipeline angepasst werden.

Aufgrund des veränderten Bildbereiches müssen die Parameter der projektiven Transformation der Bildverarbeitungspipeline angepasst werden.

### **Bildverarbeitungspipeline auf volle Bildbreite umstellen**

Aktuell wird von der Bildverarbeitungspipeline lediglich ein Teil der Bildsensorenfläche genutzt. Dies wurde vorerst umgesetzt, um dem VGA-Standard zur Ausgabe auf einem angeschlossenen Bildschirm gerecht zu werden. Dies reduziert jedoch den Blickwinkel der Kamera. Um die integrierte Kamera optimal auszunutzen, ist also das Auslesen der vollen Breite von 752 Pixeln des Bildsensors zu realisieren. Aus diese Erweiterung des Sichtbereiches resultiert eine verbesserte Fahrspurerkennung, da die Fahrspur nicht so schnell aus dem Blickwinkel der Kamera gerät.

### **Umstieg auf Zynq EPP**

Mit der Mitte dieses Jahres erschienenen Zynq-7000 Extensible Processor Platform der Firma Xilinx ist ein leistungsfähiger Mikrocontroller mit integriertem FPGA auf den Markt gekommen. Der zusammen mit einem bis zu 350000 Logikzellen umfassenden FPGA auf einem Chip realisierte Dual ARM Cortex-A9 hat eine Taktfrequenz von bis zu 800 MHz und stellt somit, im Vergleich zu den aktuell vorhandenen MicroBlaze mit 62,5 MHz, massiv gesteigerte Rechenkapazitäten bei gleichzeitig flexibel modellierbaren Hardwarebeschleunigern zur Verfügung. Denkbar ist beispielsweise die Realisierung von durch Hardwarebeschleunigern nur ressourcenintensiv umsetzbaren Operationen, wie der Hough-Transformation, auf den Prozessoren.

## 9. Zusammenfassung

In dieser Arbeit wurde eine Übersicht über in CMOS-Technologie gefertigte Bildsensoren im Bezug auf den Aufbau und die sich daraus ergebenden Parametergrößen zur Konfiguration und Beeinflussung der Bilderfassung gegeben. Dies wurde im Allgemeinen und im Speziellen für den in der 24B7525A-Kamera eingesetzten MT9V022-Bildsensor der Firma Aptina ausgeführt.

Die Kamera wurde in die vorhandenen Komponenten des SAV eingebunden. Hierzu wurde das Interface der Kamera an den Spartan 3A DSP FPGA angeschlossen. Die durch die Kamera gelieferten Bilddaten wurden für die folgenden Schritte der Bildverarbeitungskette im FPGA bereitgestellt. Zur Kontrolle und Analyse der Bilddaten wurden diese an den VGA-Controller und den FrameGrabber weitergeleitet. Der VGA-Controller wurde an das veränderte Timing der Kamera angepasst. Da die gesteigerte Datenrate der 24B7525A-Kamera nicht durch das an den FrameGrabber angeschlossene Dual-MicroBlaze System verarbeitet werden konnte, wurde die Datenerfassung im FrameGrabber angepasst.

Für den Einsatz der Kamera im SAV wurde ein Parametersatz entwickelt. Dabei wurden die Timingvoraussetzungen für eine Ausgabe auf einem VGA-Monitor berücksichtigt. Der Parametersatz sieht eine weitestgehend automatische Anpassung der Kameraparameter vor. Um die Auswirkungen der nichtlinearen Quantisierung der Bilddaten durch den Analog-Digital-Umsetzer abzuschätzen, wurde diesbezüglich eine Analyse durchgeführt. Diese ergab, dass durch die nichtlineare Quantisierung keine zusätzlichen Bilddaten erfasst werden.

Zur Konfiguration der Kamera mit dem entwickelten Parametersatz wurde das I<sup>2</sup>C-Bus Interface der Kamera an das Dual-MicroBlaze System angebunden. Um die Kamera zuverlässig zu konfigurieren, wurde ein Softwareinterface entwickelt, das ein Überprüfen der eingespielten Konfiguration durchführt.

Die vorhandenen Objektive für die Kamera wurden im Bezug auf die Linsenverzeichnung analysiert. Zur Abschätzung der Auswirkungen auf die Hough-Transformation wurden auf die aufgenommenen Bilddaten eine projektive Transformation angewandt. Hierzu wurde für jedes Objektiv ein Parametersatz errechnet. Dabei ergab sich, dass die Linsenverzeichnung bei einem steigenden Blickwinkel allgemein stärker wird. Abweichend dazu zeigte das DSL213A-Objektiv eine überaus starke Linsenverzeichnung.

# A. Anhang

## A.1. Register des MT9V022 Bildsensors

Im Folgenden wird eine vollständige Liste der bekannten Register des MT9V022 Bildsensors aufgeführt. Jedes Register ist im Allgemeinen 16 Bit breit, jedoch sind gegebenenfalls nicht alle Bits belegt. Nicht aufgeführte Bits sollten beim Schreiben auf 0 gesetzt werden. Der Zugriff auf die Register findet über das I<sup>2</sup>C-Bus Interface statt.

### A.1.1. Belichtungssteuerung

#### 0x0B: Total Shutter Width

Über dieses Register wird bei manueller Konfiguration die Belichtungszeit angegeben. Sie wird als Anzahl von Zeilen, über die die Belichtung erfolgt, angegeben. Zu beachten ist, dass eine zu lange Belichtungszeit die Bildrate reduzieren kann.

Bits	Name	Beschreibung
[14:0]	Total Shutter Width	Anzahl der Zeilen, über die die Belichtung erfolgt

**Registernummer:** 0x0B

**Registername:** Total Shutter Width

**Zugriff:** W

**Wert:** -

#### 0xA6: AEC Update Frequency

Über dieses Register wird eine verzögerte Übernahme der durch die automatische Belichtungs-kontrolle errechneten Werte konfiguriert.

Bits	Name	Beschreibung
[3:0]	Exp Skip Frame	Anzahl der Bilder, nach der die Übernahme der Werte erfolgt

**Registernummer:** 0xA6  
**Registername:** AEC Update Frequency  
**Zugriff:** W  
**Wert:** 0x0002

**0xA8: AEC Low Pass Filter**

Über dieses Register wird ein konfigurierbarer Tiefpass für die Werte der automatischen Belichtungskontrolle konfiguriert.

Der Tiefpassfilter verfügt über drei Modi:

- 0: Die neue errechneten Werte werden direkt übernommen.
- 1: Wenn der neu errechnete Wert um mehr als ein Viertel vom aktuellen Wert abweicht, wird er direkt übernommen, andernfalls wird lediglich die Hälfte der Differenz zum neuem Wert hinzugefügt.
- 2: Wenn der neu errechnete Wert um mehr als ein Viertel vom aktuellen Wert abweicht, wird er direkt übernommen, andernfalls wird lediglich ein Viertel der Differenz zum neuem Wert hinzugefügt.

Durch diese Option kann ein Schwingen des errechneten Belichtungswertes verhindert werden.

Bits	Name	Beschreibung
[1:0]	Exp LPF	Modus des AEC Tiefpassfilters

**Registernummer:** 0xA8  
**Registername:** AEC Low Pass Filter  
**Zugriff:** W  
**Wert:** 0x0002

**0xBB: AEC Exposure Output**

Über dieses Register wird die aktuell durch die automatische Belichtungskontrolle gesetzte Belichtungszeit in Zeilen zur Verfügung gestellt.

Bits	Name	Beschreibung
[15:0]	AEC Exposure	aktuell durch die automatische Belichtungskontrolle eingestellter Wert

**Registernummer:** 0xBB  
**Registername:** AEC Exposure Output  
**Zugriff:** R  
**Wert:** -

#### 0xBD: Maximum Total Shutter Width

Über dieses Register kann die maximale Belichtungszeit, die durch die automatische Belichtungszeitkontrolle eingestellt werden kann, konfiguriert werden. Dabei ist zu beachten, dass eine zu hohe maximale Belichtungszeit zu einer verringerten Bildrate führen kann. Die Angabe der Belichtungszeit erfolgt in Zeilen.

Bits	Name	Beschreibung
[15:0]	Maximum Total Shutter Width	maximale Belichtungszeit bei automatischer Belichtungszeitkontrolle in Zeilen

**Registernummer:** 0xBD  
**Registername:** Maximum Total Shutter Width  
**Zugriff:** W  
**Wert:** 0x0200

#### 0xAF: AEC/AGC Enable

Über dieses Register wird die automatische Belichtungs- und Verstärkungsfaktorkontrolle aktiviert.

Bits	Name	Beschreibung
[0]	AEC Enable	Enable der automatischen Belichtungskontrolle
[1]	AGC Enable	Enable der automatischen Verstärkungsfaktorkontrolle

**Registernummer:** 0xAF  
**Registername:** AEC/AGC Enable  
**Zugriff:** W  
**Wert:** 0x0003

#### 0xB0: AEC/AGC Pixel Count

Über dieses Register wird die Anzahl der Pixel, über die der aktuelle mittlere Grauwert berechnet wird, konfiguriert.



Bits	Name	Beschreibung
[15:0]	Pixel Count	Anzahl der Pixel, die zur Ermittlung des aktuellen mittleren Grauwertes verwendet werden

**Registernummer:** 0xB0

**Registername:** AEC/AGC Pixel Count

**Zugriff:** W

**Wert:** 0xABE0

#### 0xBC: AEC/AGC Current Bin

Über dieses Register wird der zuletzt für die Belichtungs- und Verstärkungsfaktorkontrolle ermittelte mittlere Grauwert des Bildes zur Verfügung gestellt.

Bits	Name	Beschreibung
[5:0]	Current Bin	Aktueller mittlerer Grauwert

**Registernummer:** 0xBC

**Registername:** AEC/AGC Current Bin

**Zugriff:** R

**Wert:** -

#### 0xA5: AEC/AGC Desired Bin

Über dieses Register wird ein angestrebter mittlerer Grauwert für die automatische Belichtungs- und Verstärkungsfaktorkontrolle konfiguriert.

Bits	Name	Beschreibung
[5:0]	Desired Bin	Angestrebter mittlerer Grauwert

**Registernummer:** 0xA5

**Registername:** AEC/AGC Desired Bin

**Zugriff:** W

**Wert:** 0x003A

#### 0xBE: AGC/AEC Bin Difference Threshold

Über dieses Register wird die maximale Abweichung des ermittelten vom angestrebten mittleren Grauwertes für die automatische Belichtungs- und Verstärkungsfaktorkontrolle konfiguriert.

Bits	Name	Beschreibung
[7:0]	Bin Difference Threshold	maximale Abweichung

**Registernummer:** 0xBE  
**Registername:** AGC/AEC Bin Difference Threshold  
**Zugriff:** W  
**Wert:** 0x0014

### A.1.2. Verstärkungsfaktorsteuerung

#### 0x35: Analog Gain

Über dieses Register wird der analoge Verstärkungsfaktor manuell konfiguriert. Um den analogen Verstärkungsfaktor über dieses Register zu konfigurieren, muss die automatische Verstärkungsfaktorkontrolle deaktiviert sein. Der analoge Verstärkungsfaktor ist in einem Bereich einer einfachen bis vierfachen Verstärkung konfigurierbar. Das niederwertigste Bit hat eine Wertigkeit von 0,0625, womit der Wert 4 binäre Nachkommastellen hat. Der Wertebereich beträgt 16 bis 63, was einem Verstärkungsfaktor von 1 bis 3,9375 entspricht.

Bits	Name	Beschreibung
[6:0]	Analog Gain	manuell konfigurierter analoger Verstärkungsfaktor

**Registernummer:** 0x35  
**Registername:** Analog Gain  
**Zugriff:** W  
**Wert:** -

#### 0x36: Maximum Analog Gain

Über dieses Register wird bei automatischer Verstärkungsfaktorkontrolle der maximale Verstärkungsfaktor konfiguriert. Zu beachten ist, dass der eingetragene Wert nicht den maximalen Verstärkungsfaktor übersteigen darf (Siehe Register [0x35: Analog Gain](#) für den Wertebereich).

Bits	Name	Beschreibung
[6:0]	Maximum Analog Gain	maximaler automatisch ermittelter analoger Verstärkungsfaktor

**Registernummer:** 0x36  
**Registername:** Maximum Analog Gain  
**Zugriff:** W  
**Wert:** 0x003F

### 0x80-0x98: Tiled Digital Gain

Über diese Register wird für jeweils einen der 25 konfigurierbaren Bereiche der digitale Verstärkungsfaktor sowie die jeweilige Gewichtung bei den AEC- und AGC-Algorithmen konfiguriert. Der digitale Verstärkungsfaktor hat einen Wertebereich von 0 bis 3,75. Die Wertigkeit des niederwertigsten Bits beträgt 0,25, womit der Wert zwei binäre Nachkommastellen hat. Über die Wertigkeit des Gewichtungsfaktors sind keine Angaben vorhanden.

Bits	Name	Beschreibung
[3:0]	Tile Gain	digitaler Verstärkungsfaktor des Bereiches
[7:4]	Sample Wight	Gewichtungsfaktor des Bereiches bei den AEC- und AGC-Algorithmen

**Registernummer:** 0x80-0x98

**Registername:** Tiled Digital Gain

**Zugriff:** W

**Wert:** alle 0x00F4

### 0x99-0x9E: Digital Tile Coordinate X-Direction

Über diese Register wird die Position der horizontalen Geraden zur Unterteilung des Bildes in die einzelnen Bereiche konfiguriert. Es existieren 6 Register für 6 Geraden. Konfiguriert wird der Schnittpunkt der Geraden mit der X-Achse in Pixeln. Die Aufteilung des Bildes in die einzelnen Bereiche ist in Abbildung 3.10 dargestellt.

Bits	Name	Beschreibung
[9:0]	Xn	Schnittpunkt der Geraden mit der X-Achse

**Registernummer:** 0x99-0x9E

**Registername:** Digital Tile Coordinate X-Direction

**Zugriff:** W

**Wert:** 0x0000, 0x0096, 0x012C, 0x0258, 0x02F0

### 0x9F-0xA4: Digital Tile Coordinate Y-Direction

Über diese Register wird die Position der vertikalen Geraden zur Unterteilung des Bildes in die einzelnen Bereiche konfiguriert. Es existieren 6 Register für 6 Geraden. Konfiguriert wird der Schnittpunkt der Geraden mit der Y-Achse in Pixeln. Die Aufteilung des Bildes in die einzelnen Bereiche ist in Abbildung 3.10 dargestellt.

Bits	Name	Beschreibung
[9:0]	Yn	Schnittpunkt der Geraden mit der Y-Achse

**Registernummer:** 0x9F-0xA4  
**Registername:** Digital Tile Coordinate Y-Direction  
**Zugriff:** W  
**Wert:** 0x0000, 0x0060, 0x00C0, 0x0120, 0x0180, 0x01E0

#### 0xA9: AGC Update Frequency

Über dieses Register wird eine verzögerte Übernahme der durch die automatische Verstärkungsfaktorkontrolle errechneten Werte konfiguriert.

Bits	Name	Beschreibung
[1:0]	Gain Skip Frame	Anzahl der Bilder, nach der die Übernahme der Werte erfolgt

**Registernummer:** 0xA9  
**Registername:** AGC Update Frequency  
**Zugriff:** W  
**Wert:** 0x0002

#### 0xAB: AGC Low Pass Filter

Über dieses Register wird ein konfigurierbarer Tiefpass für die Werte der automatischen Verstärkungsfaktorkontrolle konfiguriert.

Der Tiefpassfilter verfügt über drei Modi:

- 0: Die neue errechneten Werte werden direkt übernommen.
- 1: Wenn der neu errechnete Wert um mehr als ein Viertel vom aktuellen Wert abweicht, wird er direkt übernommen, andernfalls wird lediglich die Hälfte der Differenz zum neuem Wert hinzugefügt.
- 2: Wenn der neu errechnete Wert um mehr als ein Viertel vom aktuellen Wert abweicht, wird er direkt übernommen, andernfalls wird lediglich ein Viertel der Differenz zum neuem Wert hinzugefügt.

Durch diese Option kann ein Schwingen des errechneten Verstärkungsfaktorwertes verhindert werden.

Bits	Name	Beschreibung
[1:0]	Gain LPF	Modus des AGC Tiefpassfilters

**Registernummer:** 0xAB  
**Registername:** AGC Low Pass Filter  
**Zugriff:** W  
**Wert:** 0x0002

#### 0xBA: AGC Gain Output

Über dieses Register wird der aktuell durch die automatische Verstärkungsfaktorkontrolle gesetzte analoge Verstärkungsfaktor in Zeilen zur Verfügung gestellt.

Bits	Name	Beschreibung
[6:0]	AGC Exposure	aktuell durch die automatische Verstärkungsfaktorkontrolle eingestellter Wert

**Registernummer:** 0xBA  
**Registername:** AGC Gain Output  
**Zugriff:** R  
**Wert:** -

### A.1.3. Schwarzwertkorrektur

#### 0x42: Frame Dark Average

Dieses Register enthält den durch den Algorithmus für die Schwarzwertkorrektur ermittelten aktuellen Schwarzwert. Er wird aus den optisch geschwärzten Pixeln berechnet.

Bits	Name	Beschreibung
[7:0]	Frame Dark Average	aktuell ermittelter Schwarzwert

**Registernummer:** 0x42  
**Registername:** Frame Dark Average  
**Zugriff:** R  
**Wert:** -

#### 0x46: Dark Average Thresholds

Über dieses Register werden die Schwellwerte des Schwarzwertes für den Algorithmus zur Schwarzwertkorrektur konfiguriert.

Bits	Name	Beschreibung
[7:0]	Lower Thresh- hold	unterer Schwellwert
[15:8]	Upper Thresh- hold	oberer Schwellwert

**Registernummer:** 0x46

**Registername:** Dark Average Thresholds

**Zugriff:** W

**Wert:** 0x0808

#### 0x47: Black Level Calibration Control

Über dieses Register kann der Algorithmus zur Schwarzwertkorrektur konfiguriert werden.

Bits	Name	Beschreibung
[0]	Manual Overri- de	konfiguriert ein manuelles Überschreiben des Korrekturwertes mit Register <a href="#">0x48: Black Level Calibration Value</a>
[7:5]	Frames to avera- ge over	Anzahl der Bilder, über die der Schwarzwert gemittelt wird

**Registernummer:** 0x47

**Registername:** Black Level Calibration Control

**Zugriff:** W

**Wert:** 0x0080

#### 0x48: Black Level Calibration Value

Wird in Register [0x47: Black Level Calibration Control](#) das manuelle Überschreiben des Korrekturwertes aktiviert, so wird der Wert im Zweierkomplement in diesem Register konfiguriert. Ist die automatische Schwarzwertkorrektur aktiviert, so wird in diesem Register der aktuelle Korrekturwert bereitgestellt.

Bits	Name	Beschreibung
[7:0]	Black Level Cali- bration Value	vorzeichenbehafteter Korrekturwert

**Registernummer:** 0x48

**Registername:** Black Level Calibration Value

**Zugriff:** R/W

**Wert:** -

### 0x4C: Black Level Calibration Value Step Size

Über dieses Register kann die Schrittgröße der Änderung des Korrekturwertes beim Überschreiten der Schwellwerte angegeben werden.

Bits	Name	Beschreibung
[4:0]	Step Size of Calibration Value	Wert, um den der Korrekturwert jeweils angepasst wird

**Registernummer:** 0x4C

**Registername:** Black Level Calibration Value Step Size

**Zugriff:** W

**Wert:** 0x0001

### A.1.4. Bildrauschunterdrückung

#### 0x70: Row Noise Correction Control 1

Über dieses Register wird die Row Noise Correction aktiviert und die Anzahl der Pixel am linken Sensorrand, die zur Ermittlung des Korrekturwertes herangezogen werden, konfiguriert. Die Anzahl der Pixel wird nicht direkt, sondern wie in folgender Tabelle aufgelistet angegeben.

Wert	Pixel
0	2
1	4
2	6
4	10
8	18

Sind lediglich zwei Pixel konfiguriert, so werden der minimale und der maximale Wert entfernt und es sind keine Werte mehr vorhanden. Die Row Noise Correction ist somit deaktiviert. Zusätzlich lässt sich aktivieren, dass der gemittelte Schwarzwert der optisch geschwärzten Zeile für die Schwarzwertkorrektur von jedem Pixel subtrahiert wird.

Bits	Name	Beschreibung
[0:3]	Number of Dark Pixels	Anzahl Pixel
[4]	Enable Noise Correction	Row Noise Correction aktivieren
[11]	Use Black Level Average	Subtrahieren des gemittelten Schwarzwertes aus der Schwarzwertkorrektur aktivieren

**Registernummer:** 0x70  
**Registername:** Row Noise Correction Control 1  
**Zugriff:** W  
**Wert:** 0x0024

#### 0x72: Row Noise Constant

Über diese Register wird die Konstante, die zu allen Pixeln bei der Row Noise Correction addiert wird, angegeben.

Bits	Name	Beschreibung
[7:0]	Row Noise Constant	Konstante

**Registernummer:** 0x72  
**Registername:** Row Noise Constant  
**Zugriff:** W  
**Wert:** 0x0000

#### 0x73: Row Noise Correction Control 2

Über dieses Register lässt sich die Startspalte der zu verwendenden optisch geschwärzten Pixel angeben.

Bits	Name	Beschreibung
[9:0]	Dark Start Column Address	Startspalte der zu verwendenden Pixel

**Registernummer:** 0x73  
**Registername:** Row Noise Correction Control 2  
**Zugriff:** W  
**Wert:** 0x02F7

### A.1.5. Bildauslesesteuerung

#### 0x01: Column Start

Über dieses Register wird die erste Spalte des ausgegebenen Bildbereiches konfiguriert.

Bits	Name	Beschreibung
[9:0]	Column Start	erste ausgegebene Spalte



**Registernummer:** 0x01  
**Registername:** Column Start  
**Zugriff:** W  
**Wert:** 0x0038

#### 0x02: Row Start

Über dieses Register wird die erste Zeile des ausgegebenen Bildbereiches konfiguriert.

Bits	Name	Beschreibung
[8:0]	Row Start	erste ausgegebene Zeile

**Registernummer:** 0x02  
**Registername:** Row Start  
**Zugriff:** W  
**Wert:** 0x0004

#### 0x03: Window Height

Über dieses Register wird die Höhe des ausgegebenen Bildbereiches konfiguriert.

Bits	Name	Beschreibung
[8:0]	Window Height	Höhe des ausgegebenen Bildbereiches

**Registernummer:** 0x03  
**Registername:** Window Height  
**Zugriff:** W  
**Wert:** 0x01E0

#### 0x04: Window Width

Über dieses Register wird die Breite des ausgegebenen Bildbereiches konfiguriert.

Bits	Name	Beschreibung
[9:0]	Window Width	Breite des ausgegebenen Bildbereiches

**Registernummer:** 0x04  
**Registername:** Window Width  
**Zugriff:** W  
**Wert:** 0x0280

### 0x05: Horizontal Blanking

Über dieses Register wird die Pause zwischen den Zeilen während des Auslesens konfiguriert.

Der Wert wird in Spalten angegeben.

Bits	Name	Beschreibung
[9:0]	Horizontal Blanking	Länge der Pause zwischen den Zeilen in Spalten

**Registernummer:** 0x05

**Registername:** Horizontal Blanking

**Zugriff:** W

**Wert:** 0x00E0

### 0x06: Vertical Blanking

Über dieses Register wird die Pause zwischen den Bildern während des Auslesens konfiguriert.

Der Wert wird in Zeilen angegeben.

Bits	Name	Beschreibung
[14:0]	Vertical Blanking	Länge der Pause zwischen den Bildern in Zeilen

**Registernummer:** 0x06

**Registername:** Vertical Blanking

**Zugriff:** W

**Wert:** 0x002A

### 0x07: Chip Control

Über dieses Register wird der Interlaced-Modus, die parallele Datenschnittstelle, die simultane Belichtung und die Pixelkorrektur aktiviert. Die Bits zum kontrollieren des Interlaced-Modus lassen mehrere Modi zu:

- 0: kein Interlacing
- 1: ungültig
- 2: Interlacing aktiv, beide Halbbilder werden ausgegeben; gerade Zeilen werden zuerst ausgegeben
- 3: Interlacing aktiv, nur ein Halbbild wird ausgegeben

Weiterhin wird über dieses Register der Stereoskopiemodus konfiguriert.

Bits	Name	Beschreibung
[2:0]	Scan Mode	Modus des Interlacing
[3]	Sensor Master/Slave Mode	aktiviert den Master-Mode des Sensors
[4]	Sensor Snapshot Mode	Funktion ist unbekannt
[5]	Stereoscopy Mode	aktiviert den Stereoskopiemodus
[6]	Stereoscopic Master/Slave Mode	konfiguriert, ob der Sensor im Stereoskopiemodus Master oder Slave darstellt
[7]	Parallel Output Enable	aktiviert die Datenausgabe über die parallele Schnittstelle
[8]	Simultaneous/Sequential Mode	aktiviert das Belichten des Bildsensors während des Auslesens
[9]	Defect Pixel Correction Enable	aktiviert die Korrektur defekter Pixel

**Registernummer:** 0x07

**Registername:** Chip Control

**Zugriff:** W

**Wert:** 0x0388

#### 0xBF: Field Vertical Blank

Über dieses Register wird die Pause zwischen zwei Halbbildern im Interlaced-Modus während des Auslesens konfiguriert.

Bits	Name	Beschreibung
[8:0]	Field Vertical Blank	Länge der Pause zwischen zwei Halbbildern in Zeilen

**Registernummer:** 0xBF

**Registername:** Field Vertical Blank

**Zugriff:** W

**Wert:** -

**0x0D: Read Mode**

Über dieses Register wird ein Reduzieren der Daten durch Auflösungsreduktion konfiguriert. Bei der Reduktion der Auflösung bedeutet eine 0 keine Reduktion, eine 1 die Halbierung der Auflösung und eine 2 das Vierteln der Auflösung. Weiterhin wird konfiguriert, ob die Ausleserichtung invertiert werden soll und ob die optisch geschwärzten Pixel mit ausgelesen werden sollen.

Bits	Name	Beschreibung
[1:0]	Row Bin	Auflösungsreduktion in vertikaler Richtung
[3:2]	Column Bin	Auflösungsreduktion in horizontaler Richtung
[4]	Row Flip	Bild von unten nach oben auslesen
[5]	Column Flip	Bild von rechts nach links auslesen
[6]	Show Dark Rows	optisch geschwärzte Zeilen ausgeben
[7]	Show Dark Columns	optisch geschwärzte Spalten ausgeben

**Registernummer:** 0x0D

**Registername:** Read Mode

**Zugriff:** W

**Wert:** 0x0300

**0x0E: Monitor Mode**

Über dieses Register wird der Überwachungsmodus aktiviert. Ist er aktiviert, so nimmt der Bildsensor alle fünf Minuten eine in Register [A.1.5](#) konfigurierbare Anzahl von Bildern auf.

Bits	Name	Beschreibung
[0]	Monitor Mode Enable	Aktiviert den Überwachungsmodus

**Registernummer:** 0x0E

**Registername:** Monitor Mode

**Zugriff:** W

**Wert:** 0x0000

**0xC0: Monitor Mode Capture Control**

Über dieses Register wird die Anzahl der aufzunehmenden Bilder im Überwachungsmodus konfiguriert. Gegebenenfalls sind mehrere Bilder zu konfigurieren, da der Sensor die Belichtungszeit und den Verstärkungsfaktor erst der Umgebung anpassen muss.

Bits	Name	Beschreibung
[7:0]	Image Capture Number	Anzahl der aufzunehmenden Bilder im Überwachungsmodus

**Registernummer:** 0xC0

**Registername:** Monitor Mode Capture Control

**Zugriff:** W

**Wert:** -

#### 0x74: Pixel Clock & Sync Control

Über dieses Register wird ein Invertieren des Pixeltaktausganges sowie der Frame- und Linevalid-Signale konfiguriert.

Bits	Name	Beschreibung
[0]	Invert Line Valid	Linevalid-Signal invertieren
[1]	Invert Frame Valid	Framevalid-Signal invertieren
[2]	Xor Line Valid	wenn 'Continuous Line Valid' aktiviert wird das Linevalid-Signal mit dem Framevalid-Signal XOR-verknüpft ausgeben
[3]	Continuous Line Valid	Linevalid-Signal auch während der Pause zwischen zwei Bildern generieren
[4]	Invert Pixel Clock	Pixeltakt invertiert (um 180° phasenverschoben) ausgeben

**Registernummer:** 0x74

**Registername:** Pixel Clock & Sync Control

**Zugriff:** W

**Wert:** 0x0000

#### 0xC3: NTSC Frame Valid Control

Über dieses Register wird eine Ausgabe der Synchronisationssignale nach NTSC konfiguriert.

Bits	Name	Beschreibung
[0]	Extended Frame Valid	aktiviert eine Verlängerung des Framevalid-Signals um eine halbe Zeile nach dem Interlaced-Halbbild mit den ungeraden Zeilen
[1]	Replace FV/LV with Ped/Sync	

**Registernummer:** 0xC3  
**Registername:** NTSC Frame Valid Control  
**Zugriff:** W  
**Wert:** 0x0000

#### 0xC4: NTSC Horizontal Blanking

Über dieses Register wird die Ausgabe eines horizontalen NTSC-Synchronisationssignals konfiguriert.

Bits	Name	Beschreibung
[7:0]	Front Porch Width	Länge des Frontporch in Pixeltakten
[15:8]	Sync Width	Pulsbreite des Synchronisationssignales in Pixeltakten

**Registernummer:** 0xC4  
**Registername:** NTSC Horizontal Blanking  
**Zugriff:** W  
**Wert:** -

#### 0xC5: NTSC Vertikal Blanking

Über dieses Register wird die Ausgabe eines vertikalen NTSC-Synchronisationssignals konfiguriert.

Bits	Name	Beschreibung
[7:0]	Equalizing Pulse Width	Pulsweite des Synchronisationssignals in Pixeltakten
[15:8]	Vertical Serration Width	Pulsweite des Synchronisationssignals in Pixeltakten

**Registernummer:** 0xC5  
**Registername:** NTSC Vertikal Blanking  
**Zugriff:** W  
**Wert:** -

### A.1.6. Analog-Digital-Umsetzer

#### 0x1C: ADC Resolution Control

Über dieses Register wird die nichtlineare Quantisierung des Analog-Digital-Umsetzers aktiviert. Gültige Werte sind:

- 2: 10 Bit; lineare Quantisierung
- 3: 12 Bit; nicht lineare Quantisierung

Bits	Name	Beschreibung
[1:0]	ADC Mode	Modus des Analog-Digital-Umsetzers

**Registernummer:** 0x1C  
**Registername:** ADC Resolution Control  
**Zugriff:** W  
**Wert:** 0x0002

#### 0x2C: VREF ADC Control

Über dieses Register wird die Referenzspannung des Analog-Digital-Umsetzers konfiguriert. Die Spannung kann durch folgende Werte konfiguriert werden:

Wert	Spannung
0	1,0 V
1	1,1 V
2	1,2 V
3	1,3 V
4	1,4 V (default)
5	1,5 V
6	1,6 V
7	2,1 V

Bits	Name	Beschreibung
[2:0]	VREF_ADC Voltage Level	Referenzspannung nach obiger Tabelle

**Registernummer:** 0x2C  
**Registername:** VREF ADC Control  
**Zugriff:** W  
**Wert:** 0x0004

### A.1.7. High Dynamic Range

#### 0x0F: Pixel Operation Mode

Über dieses Register wird der HDR-Modus aktiviert. Weiterhin wird konfiguriert, ob im sequentiellen Modus nur die Hälfte der Zeit des Bildes oder die Zeit des gesamten Bildes zur Belichtung

verwendet wird. Abgesehen davon wird hier konfiguriert, ob es sich um die Farbversion des Bildsensors handelt.

Bits	Name	Beschreibung
[2]	Colour/Mono	wenn aktiv, dann handelt es sich um die Farbversion des Sensors
[6]	High Dynamic Range	HDR-Modus aktivieren
[7]	Enable Extended Exposure	Verwendung der gesamten Zeit des Bildes zur Belichtung aktivieren

**Registernummer:** 0x0F

**Registername:** Pixel Operation Mode

**Zugriff:** W

**Wert:** 0x0011

#### 0x31-0x34: V1-V4

Über diese Register werden die Spannungen der HDR-Steuerung an den einzelnen Eckpunkten konfiguriert. Der konfigurierbare Wertebereich reicht in 62,5 mV Schritten von 0,5625 V bis 2,5 V. (siehe Kapitel 3.3.7)

Bits	Name	Beschreibung
[4:0]	Vn Voltage Level	Spannung zum jeweiligen Zeitpunkt

**Registernummer:** 0x31-0x34

**Registername:** V1-V4

**Zugriff:** W

**Wert:** 0x001D, 0x0018, 0x0015, 0x0004

#### 0x08+0x09: Shutter Width 1+2

Über diese Register wird gegebenenfalls ein fester Zeitpunkt für die Eckpunkte des Spannungsverlaufes der HDR-Steuerung konfiguriert. (siehe Kapitel 3.3.7)

Bits	Name	Beschreibung
[14:0]	Shutter Width n	Zeitpunkt für die jeweilige Spannung in Zeilen

**Registernummer:** 0x08+0x09

**Registername:** Shutter Width 1+2

**Zugriff:** W

**Wert:** 0x01BB, 0x01D9



### 0x0A: Shutter Width Control

Über dieses Register wird die Berechnung der Zeitpunkte für die Eckpunkte des Spannungsverlaufes der HDR-Steuerung konfiguriert. Es kann entweder ein relativer oder ein absoluter Zeitpunkt mit den Registern [0x08+0x09: Shutter Width 1+2](#) angegeben werden. (siehe Kapitel [3.3.7](#))

Bits	Name	Beschreibung
[3:0]	T2 Ratio	
[7:4]	T3 Ratio	
[8]	Exposure Knee Auto Adjust Enable	relativen Zeitpunkt aktivieren
[9]	Single Knee Enable	Spannungsverlauf besteht aus nur einem Eckpunkt wenn aktiviert

**Registernummer:** 0x0A

**Registername:** Shutter Width Control

**Zugriff:** W

**Wert:** 0x0164

### A.1.8. LVDS

#### 0xB1: LVDS Master Control

Über dieses Register wird der LVDS-Funktionsblock aktiviert, eine Taktquelle für das Schieberegister eingestellt und eine Testausgabe konfiguriert.

Bits	Name	Beschreibung
[0]	PLL Bypass	LVDS_BYPASS_CLK-Eingang als Taktquelle für Schieberegister nutzen
[1]	LVDS- Powerdown	schaltet den LVDS-Funktionsblock ab
[2]	PLL Test Mode	Taktquelle für Schieberegister ist Systemtakt
[3]	LVDS Test Mode	an LVDS-Ausgang wird Rechteckspannung ausgegeben

**Registernummer:** 0xB1

**Registername:** LVDS Master Control

**Zugriff:** W

**Wert:** 0x0002

#### 0xB2: LVDS Shiftclock Control

Über dieses Register wird eine Verzögerung für die LVDS-Taktquelle des Schieberegisters und ein Deaktivieren konfiguriert.

Bits	Name	Beschreibung
[2:0]	Shiftclock Delay	Verzögerung des Taktes für das Schieberegister
[4]	Shiftclock Powerdown	schaltet die Taktquelle ab

**Registernummer:** 0xB2

**Registername:** LVDS Shiftclock Control

**Zugriff:** W

**Wert:** 0x0010

#### 0xB3: LVDS Data Control

Über dieses Register wird eine Verzögerung für die LVDS Empfangsschaltung des Stereoskopie-modus und ein Deaktivieren konfiguriert.

Bits	Name	Beschreibung
[2:0]	Data Delay	Verzögerung des Taktes für das Schieberegister
[4]	Data Receiver Powerdown	schaltet die Empfangsschaltung ab

**Registernummer:** 0xB3

**Registername:** LVDS Data Control

**Zugriff:** W

**Wert:** 0x0010

#### 0xB4: LVDS Latency

Lediglich der Registername ist bekannt. Die Funktionen sind nicht dokumentiert.

**Registernummer:** 0xB4

**Registername:** LVDS Latency

**Zugriff:** -

**Wert:** -

#### 0xB5: LVDS Internal Sync

Über dieses Register wird die Ausgabe von Synchronisationsmusters im LVDS-Datenstrom aktiviert.

Bits	Name	Beschreibung
[0]	LVDS Internal Sync Enable	Sensor gibt abgesehen von Startbit nur 0 aus

**Registernummer:** 0xB5

**Registername:** LVDS Internal Sync

**Zugriff:** W

**Wert:** -

#### 0xB6: LVDS Payload Control

Über dieses Register wird die Ausgabe von 10 Bit Werte aktiviert. Werden 10 Bit Werte ausgegeben, so werden die Frame- und Linevaliddaten ersetzt.

Bits	Name	Beschreibung
[0]	Use 10-bit Pixel Enable	10 Bit pro Pixel werden über LVDS ausgegeben, wenn aktiviert

**Registernummer:** 0xB6

**Registername:** LVDS Payload Control

**Zugriff:** W

**Wert:** -

#### 0xB7: Stereoscapy Error Control

Über dieses Register wird die Ausgabe von Fehlern im Stereoskopiemodus auf der ERROR-Ausgang aktiviert.

Bits	Name	Beschreibung
[0]	Enable Stereo Error Detect	Aktiviert die Fehlererkennung
[1]	Enable Stereo Error Sticky Flag	Funktion nicht genau bekannt
[2]	Clear Stereo Error Flag	Löscht die Fehlerausgabe

**Registernummer:** 0xB7

**Registername:** Stereoscapy Error Control

**Zugriff:** W

**Wert:** -

### 0xB8: Stereoscapy Error Falgs

Lediglich der Registername ist bekannt. Die Funktionen sind nicht dokumentiert.

**Registernummer:** 0xB8  
**Registername:** Stereoscapy Error Falgs  
**Zugriff:** -  
**Wert:** -

### 0xB9: LVDS Data Output

Lediglich der Registername ist bekannt. Die Funktionen sind nicht dokumentiert.

**Registernummer:** 0xB9  
**Registername:** LVDS Data Output  
**Zugriff:** -  
**Wert:** -

## A.1.9. Allgemeine Register

### 0x00: Chip Version

Dieses Register enthält einen von der Chipversion abhängigen Wert.

Version	Wert
Rev2	0x1311
Rev3	0x1313

**Registernummer:** 0x00  
**Registername:** Chip Version  
**Zugriff:** R  
**Wert:** 0x1313

### 0x0C: Reset

Über dieses Register kann ein Softwarereset durchgeführt werden.

Bits	Name	Beschreibung
[0]	Soft Reset	Reset des Bildsensors; Registerwerte bleiben erhalten
[1]	Auto Block Soft Reset	Reset der AEC- und AGC-Blöcke

**Registernummer:** 0x0C  
**Registername:** Reset  
**Zugriff:** W  
**Wert:** -

**0x1B: LEDOUT Control**

Über dieses Register wird konfiguriert, was am LED\_OUT-Ausgang ausgegeben werden soll.

Bits	Name	Beschreibung
[0]	Disable LED_OUT	Deaktiviert den LED_OUT-Ausganges
[1]	Invert LED_OUT	invertieren des LED_OUT-Ausganges

**Registernummer:** 0x1B  
**Registername:** LEDOUT Control  
**Zugriff:** W  
**Wert:** 0x0000

**0x7F: Test Pattern**

Über dieses Register kann die Ausgabe eines Testpatterns durch den Bildsensor aktiviert werden. Zusätzlich kann das I<sup>2</sup>C-Bus Interface getestet werden.

Bits	Name	Beschreibung
[9:0]	TWI Test Data	Testdaten zum Testen des I <sup>2</sup> C-Bus Interface
[10]	Use TWI Test Data	aktiviert die Ausgabe der Testdaten des I <sup>2</sup> C-Bus Interface
[12:11]	Grey Shade Test Pattern	aktiviert die Ausgabe von Grauübergänge in vertikaler, horizontaler oder diagonaler Richtung ausgegeben
[13]	Test Enable	Testausgabe aktivieren
[14]	Flip TWI Test Data	invertieren der Testdaten des I <sup>2</sup> C-Bus Interface

**Registernummer:** 0x7F  
**Registername:** Test Pattern  
**Zugriff:** W  
**Wert:** 0x0000

### 0xC1: Temperature

Über dieses Register wird der Wert des On-Chip Temperatursensors zur Verfügung gestellt. Die genaue Bedeutung dieses Wertes ist nicht bekannt.

**Registernummer:** 0xC1  
**Registername:** Temperature  
**Zugriff:** R  
**Wert:** -

### 0xC2: Analog Controls

Über dieses Register wird die Anti-Eclipse-Schaltung konfiguriert.

Bits	Name	Beschreibung
[7]	Anti-Eclipse Enable	aktiviert die Anti-Eclipse-Schaltung
[13:11]	V_rst_lim Voltage Level	Referenzspannung für Anti-Eclipse-Schaltung; ergibt sich aus $Wert * 50mV + 1,95V$

**Registernummer:** 0xC2  
**Registername:** Analog Controls  
**Zugriff:** W  
**Wert:** 0x0840

### 0xFE: Register Lock

Über dieses Register wird das Verändern der Register durch I<sup>2</sup>C-Bus Zugriffe deaktiviert. Das Schreiben des Wertes 0xDEAD aktiviert und der Wert 0xBEEF deaktiviert den Schreibschutz.

Bits	Name	Beschreibung
[15:0]	Register Lock	aktivieren/deaktivieren des Schreibschutzes

**Registernummer:** 0xFE  
**Registername:** Register Lock  
**Zugriff:** W  
**Wert:** 0xBEEF

## A.2. Konfigurationswerte für die Register des MT9V022 Bildsensors

Im Folgenden befindet sich eine Tabelle mit der erstellten Registerkonfiguration.

Registernummer	Registername	Zugriff	Wert
0x00	Chip Version	R	0x1313
0x01	Column Start	W	0x0038
0x02	Row Start	W	0x0004
0x03	Window Height	W	0x01E0
0x04	Window Width	W	0x0280
0x05	Horizontal Blanking	W	0x00E0
0x06	Vertical Blanking	W	0x002A
0x07	Chip Control	W	0x0388
0x08+0x09	Shutter Width 1+2	W	0x01BB, 0x01D9
0x0A	Shutter Width Control	W	0x0164
0x0B	Total Shutter Width	W	-
0x0C	Reset	W	-
0x0D	Read Mode	W	0x0300
0x0E	Monitor Mode	W	0x0000
0x0F	Pixel Operation Mode	W	0x0011
0x1B	LEDOUT Control	W	0x0000
0x1C	ADC Resolution Control	W	0x0002
0x2C	VREF ADC Control	W	0x0004
0x31-0x34	V1-V4	W	0x001D, 0x0018, 0x0015, 0x0004
0x35	Analog Gain	W	-
0x36	Maximum Analog Gain	W	0x003F
0x42	Frame Dark Average	R	-
0x46	Dark Average Thresholds	W	0x0808
0x47	Black Level Calibration Control	W	0x0080
0x48	Black Level Calibration Value	R/W	-

Registernummer	Registername	Zugriff	Wert
0x4C	Black Level Calibration Value Step Size	W	0x0001
0x70	Row Noise Correction Control 1	W	0x0024
0x72	Row Noise Constant	W	0x0000
0x73	Row Noise Correction Control 2	W	0x02F7
0x74	Pixel Clock & Sync Control	W	0x0000
0x7F	Test Pattern	W	0x0000
0x80-0x98	Tiled Digital Gain	W	alle 0x00F4
0x99-0x9E	Digital Tile Coordinate X-Direction	W	0x0000, 0x0096, 0x012C, 0x0258, 0x02F0
0x9F-0xA4	Digital Tile Coordinate Y-Direction	W	0x0000, 0x0060, 0x00C0, 0x0120, 0x0180, 0x01E0
0xA5	AEC/AGC Desired Bin	W	0x003A
0xA6	AEC Update Frequency	W	0x0002
0xA8	AEC Low Pass Filter	W	0x0002
0xA9	AGC Update Frequency	W	0x0002
0xAB	AGC Low Pass Filter	W	0x0002
0xAF	AEC/AGC Enable	W	0x0003
0xB0	AEC/AGC Pixel Count	W	0xABE0
0xB1	LVDS Master Control	W	0x0002
0xB2	LVDS Shiftclock Control	W	0x0010
0xB3	LVDS Data Control	W	0x0010
0xB4	LVDS Latency	-	-
0xB5	LVDS Internal Sync	W	-
0xB6	LVDS Payload Control	W	-
0xB7	Stereoscopy Error Control	W	-



Registernummer	Registername	Zugriff	Wert
0xB8	Stereoscopy Error Falgs	-	-
0xB9	LVDS Data Output	-	-
0xBA	AGC Gain Output	R	-
0xBB	AEC Exposure Output	R	-
0xBC	AEC/AGC Current Bin	R	-
0xBD	Maximum Total Shutter Width	W	0x0200
0xBE	AGC/AEC Bin Difference Threshold	W	0x0014
0xBF	Field Vertical Blank	W	-
0xC0	Monitor Mode Capture Control	W	-
0xC1	Temperature	R	-
0xC2	Analog Controls	W	0x0840
0xC3	NTSC Frame Valid Control	W	0x0000
0xC4	NTSC Horizontal Blanking	W	-
0xC5	NTSC Vertikal Blanking	W	-
0xFE	Register Lock	W	0xBEEF

Tabelle A.1.: Registerwerte für den MT9V022-Bildsensor

### A.3. Listing des Sourcecodes

Der Sourcecode ist in Form eines Datenträgers angefügt. Die folgenden Ordner sind enthalten:

- **hw/FrameGrabber/** Quellcode des FrameGrabber
- **hw/VGA-Controller/** Quellcode des VGA-Controllers mit Testbench
- **hw/pcores/** IP's für die Verwendung im Xilinx EDK
- **sw/interface/** Interface zur Konfiguration der Kamera
- **sw/MB0/** Software für den MicroBlaze 0 des SAV (MicroBlaze 1 wurde nicht modifiziert)
- **matlab/** Matlab-Skripte zur Ergebnisanalyse
- **XPS\_SAV\_CTRL/** Xilinx EDK Projektverzeichnis des SAV

In den einzelnen Ordnern befinden sich gegebenenfalls Textdateien zur weiteren Erläuterung.

# Literatur

- [1] Analog Devices Inc. *ADI\_MT9V022 Device Driver*. Online Quelle, abgerufen am 13.06.2012 um 18:00. Jan. 2006. URL: [http://ez.analog.com/servlet/JiveServlet/download/4924-1689/adi\\_mt9v022.pdf](http://ez.analog.com/servlet/JiveServlet/download/4924-1689/adi_mt9v022.pdf).
- [2] Aptina Imaging Corporation. *1/3-Inch Wide-VGA CMOS Digital Image Sensor, MT9V022, Silicon Revision 3 Errata*. Rev. D. Online Quelle, abgerufen am 13.06.2012 um 16:50. Aug. 2010. URL: <http://www.aplina.com/assets/downloadDocument.do?id=736>.
- [3] Aptina Imaging Corporation. *1/3-Inch Wide-VGA CMOS Digital Image Sensor, MT9V022, Timing Specifications Addendum*. Rev. C. Online Quelle, abgerufen am 13.06.2012 um 16:58. Sep. 2010. URL: <http://www.aplina.com/assets/downloadDocument.do?id=660>.
- [4] Aptina Imaging Corporation. *1/3-Inch Wide-VGA CMOS Digital Image Sensor, MT9V022*. Rev. H. Online Quelle, abgerufen am 05.06.2012 um 16:55. Juni 2010. URL: <http://www.aplina.com/assets/downloadDocument.do?id=498>.
- [5] F. Arakawa u. a. „An embedded processor core for consumer appliances with 2.8GFLOPS and 36M polygons/s FPU“. In: *Solid-State Circuits Conference, 2004. Digest of Technical Papers. ISSCC. 2004 IEEE International*. 2004, S. 334 –531 Vol.1. DOI: [10.1109/ISSCC.2004.1332730](https://doi.org/10.1109/ISSCC.2004.1332730).
- [6] Atmel Corporation. *ATmega32 Documentation*. 2503Q-02/11. Online Quelle, abgerufen am 23.07.2012 um 15:00. Feb. 2011. URL: <http://www.atmel.com/Images/doc2503.pdf>.
- [7] Wolfgang Bludau. *Halbleiter-Optoelektronik : die physikalischen Grundlagen der LED's, Diodenlaser und pn-Photodioden*. Reihe Informationstechnik, Nachrichtentechnik. 2. Auflage. München und Wien: Carl Hanser Verlag München Wien, 1995.
- [8] Bundesministerium für Bildung und Forschung. *Allgemeine Hinweise zur BMBF-Förderung von Embedded Systems*. URL: <http://www.pt-it.pt-dlr.de/de/1852.php>.
- [9] *Carolo-Cup Regelwerk 2012*. Online Quelle, abgerufen am 23.08.2012 um 8:00. Juni 2011. URL: <http://www.carolo-cup.de/uploads/media/Regelwerk.pdf>.

- [10] Steven Decker u. a. „A 256 x 256 CMOS Imaging Array with Wide Dynamic Range Pixels and Column-Parallel Digital Output“. In: *IEEE Journal of Solid-State Circuits* 33.12 (Dez. 1998). Online Quelle, abgerufen aus dem Netz der HAW am 03.08.2012 um 13:10. URL: <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=735551&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel4%2F4%2F15831%2F00735551.pdf%3Farnumber%3D735551>.
- [11] Digilent Inc. *Nexys2 Reference Manual*. Online Quelle, abgerufen am 13.06.2012 um 21:50. Juli 2011. URL: [http://digilentinc.com/Data/Products/NEXYS2/Nexys2\\_rm.pdf](http://digilentinc.com/Data/Products/NEXYS2/Nexys2_rm.pdf).
- [12] FAUST Fahrerassistenz- und Autonome Systeme. URL: <http://www.informatik.haw-hamburg.de/faust.html>.
- [13] Abbas El Gamal. *High Dynamic Range Image Sensors*. Im Rahmen der International Solid-State Circuits Conference 2002 gehaltenes Tutorial. Online Quelle, abgerufen aus dem Netz der HAW am 01.08.2012 um 16:40. 2002. URL: [http://www-isl.stanford.edu/~abbas/group/papers\\_and\\_pub/isscc02\\_tutorial.pdf](http://www-isl.stanford.edu/~abbas/group/papers_and_pub/isscc02_tutorial.pdf).
- [14] Encho Hwang. *Build A VGA Monitor Controller*. Online Quelle, abgerufen am 13.08.2012 um 18:10. Nov. 2001. URL: <http://faculty.lasierra.edu/~ehwang/public/mypublications/VGA%20Monitor%20Controller.pdf>.
- [15] Ramin Jeyrani-Mameghani. „SoC-basierte Erprobungsplattform für CCD-Kameras mit Channel Link Interface“. Diplomarbeit. Hochschule für Angewandte Wissenschaften Hamburg, Sep. 2009.
- [16] Marco Kirschke. „FPGA-basierte MPSoC-Plattform zur Integration eines Antikollisionssystems in die Fahrspurführung eines autonomen Fahrzeugs“. Masterarbeit. Hochschule für Angewandte Wissenschaften Hamburg, März 2012.
- [17] Dave Litwiller. „CCD vs. CMOS: Facts and Fiction“. In: *Photonics Spectra* (Jan. 2001). Online Quelle, abgerufen am 28.07.2012 um 14:55. URL: [www.dalsa.com/public/corp/Photonics\\_Spectra\\_CCDvsCMOS\\_Litwiller.pdf](http://www.dalsa.com/public/corp/Photonics_Spectra_CCDvsCMOS_Litwiller.pdf).
- [18] Andreas Meisel. *Vorlesung Robot Vision*. 2012. URL: [http://www.informatik.haw-hamburg.de/wp\\_robot\\_vision.html](http://www.informatik.haw-hamburg.de/wp_robot_vision.html).
- [19] NXP Semiconductors. *I<sup>2</sup>C-bus specification and user manual*. Rev. 4. Online Quelle, abgerufen am 23.07.2012 um 12:15. Feb. 2012. URL: [http://www.nxp.com/documents/user\\_manual/UM10204.pdf](http://www.nxp.com/documents/user_manual/UM10204.pdf).

- [20] Hee Kong Phoon, M. Yap und Chuan Khye Chai. „A Highly Compatible Architecture Design for Optimum FPGA to Structured-ASIC Migration“. In: *Semiconductor Electronics, 2006. ICSE '06. IEEE International Conference on*. 2006, S. 506–510. DOI: [10.1109/SMELEC.2006.3811114](https://doi.org/10.1109/SMELEC.2006.3811114). URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4266664>.
- [21] C.R.W. Reinbrecht, J.L. Da Silva und E.E. Fabris. „Applying in education an FPGA-based methodology to prototype ASIC soft cores and test ICs“. In: *Programmable Logic (SPL), 2012 VIII Southern Conference on*. 2012, S. 1–5. DOI: [10.1109/SPL.2012.6211803](https://doi.org/10.1109/SPL.2012.6211803). URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6211803>.
- [22] William D. Rogatto, Joseph S. Accetta und David L. Shumaker, Hrsg. *Electro-Optical Components*. Bd. 3. The Infrared and Electro-Optical Systems Handbook. SPIE Optical Engineering Press, 1993.
- [23] Edris Sahak. „SoC-basierte partielle Rekonfiguration einer modularisierten Bildverarbeitungs pipeline“. Bachelorarbeit. Hochschule für Angewandte Wissenschaften Hamburg, Aug. 2011.
- [24] Rudolf Sautter, Hrsg. *Leiterplatten mit oberflächenmontierten Bauelementen : Technik oberflächenmontierter Bauelemente (SMD) ; Entwerfen, Fertigen, Bestücken, Löten von Leiterplatten mit oberflächenmontierten elektrischen und elektronischen Bauelementen*. 1. Auflage. Vogel-Buchverlag Würzburg, 1988.
- [25] Videology Imaging Solutions Inc. *Application Note 24B752XA Wide VGA B&W CMOS Board Camera*. Revision C. Online Quelle, abgerufen am 25.07.2012 um 13:50. Nov. 2008. URL: <http://www.videologyinc.com/media/products/application%20notes/APN-24B752XA.pdf>.
- [26] Thomas Wollinger, Jorge Guajardo und Christof Paar. „Security on FPGAs: State-of-the-art implementations and attacks“. In: *ACM Trans. Embed. Comput. Syst.* 3.3 (Aug. 2004), S. 534–574. ISSN: 1539-9087. DOI: [10.1145/1015047.1015052](https://doi.org/10.1145/1015047.1015052). URL: <http://doi.acm.org/10.1145/1015047.1015052>.
- [27] Xilinx Inc. *LogiCORE IP Multi-Port Memory Controller (MPMC) (v6.03.a)*. Aus dem Xilinx EDK Version 13.4. März 2011.
- [28] Xilinx Inc. *MicroBlaze Debug Module (MDM) (v2.00.b)*. Aus dem Xilinx EDK Version 13.4. März 2011.

- [29] Xilinx Inc. *OS and Libraries Document Collection*. 13.4. Aus dem Xilinx EDK Version 13.4, sowie online abgerufen am 17.08.2012 um 15:15. Juli 2012. URL: [http://www.xilinx.com/support/documentation/sw\\_manuals/xilinx13\\_4/oslib\\_rm.pdf](http://www.xilinx.com/support/documentation/sw_manuals/xilinx13_4/oslib_rm.pdf).
- [30] Xilinx Inc. *Spartan-3 Generation FPGA User Guide*. v1.8. Online Quelle, abgerufen am 25.07.2012 um 8:20. Juni 2011. URL: [http://www.xilinx.com/support/documentation/user\\_guides/ug331.pdf](http://www.xilinx.com/support/documentation/user_guides/ug331.pdf).
- [31] Xilinx Inc. *Spartan-3A DSP FPGA Family Data Sheet*. v3.0. Online Quelle, abgerufen am 25.07.2012 um 8:15. Okt. 2010. URL: [http://www.xilinx.com/support/documentation/data\\_sheets/ds610.pdf](http://www.xilinx.com/support/documentation/data_sheets/ds610.pdf).
- [32] Xilinx Inc. *Spartan-3A DSP Starter Platform User Guide*. v1.1. Online Quelle, abgerufen am 25.07.2012 um 8:30. Jan. 2009. URL: [http://www.xilinx.com/support/documentation/boards\\_and\\_kits/ug454\\_sp3a\\_dsp\\_start\\_ug.pdf](http://www.xilinx.com/support/documentation/boards_and_kits/ug454_sp3a_dsp_start_ug.pdf).
- [33] Xilinx Inc. *XPI IIC Bus Interface v2.03a*. 1.11. Aus dem Xilinx EDK Version 13.4. Juni 2011.

# Tabellenverzeichnis

4.1.	Reservierte Adressen des I <sup>2</sup> C-Busses [19, S. 17]	34
4.2.	Übertragungsraten I <sup>2</sup> C-Bus [19, S. 35]	35
4.3.	Konfigurationsparameter für das Xilinx XPS IIC Interface IP	36
5.1.	Reset-Spannungswerte für den HDR-Modus	41
6.1.	Pinout des Dateninterfaces der 24B8525A-Kamera	51
6.2.	Vorgabe VGA-Timinggrößen (25,175 MHz) [14]	53
6.3.	Angepasste VGA-Timinggrößen (27 MHz)	54
6.4.	Ports und Generics der VGA-Controller Entity	57
7.1.	Liste der Kameraobjektive	64
A.1.	Registerwerte für den MT9V022-Bildsensor	100

# Abbildungsverzeichnis

1.1.	in dieser Arbeit erstelltes System . . . . .	3
2.1.	grobe Systemübersicht über die Komponenten des SOC-Kamera-Systems . . . . .	5
2.2.	Übersicht über die Hardwarekomponenten des Systems . . . . .	7
2.3.	Dual-MicroBlaze System mit angebotenen Komponenten . . . . .	9
2.4.	Übersicht über die Bildverarbeitungspipeline . . . . .	10
2.5.	Übersicht über die Ansteuerung der Aktorik . . . . .	11
3.1.	grober Aufbau CCD-Bildsensor (links) und CMOS-Bildsensor (rechts); die Pfeile markieren den Signalweg beim Auslesen . . . . .	13
3.3.	Verlauf des Reset-Signals und der integrierten Spannung bei unterschiedlicher Photonenflussdichte im HDR-Modus . . . . .	16
3.4.	Ausgangssignal im Verhältnis zum Photonenfluss im HDR-Modus des MT9V022-Bildsensors; es ergeben sich mehrere lineare Abschnitte, abhängig von den Zeiten $t_n$ und den Spannungsstufen $dV_n$ [4, S. 26] . . . . .	16
3.2.	Active-Pixel mit Leitungsnetz zum Auslesen (Bus) und Ansteuern (Enables, Reset), Ausleseschaltung, Auslesesteuerung sowie DSP zur Signalaufbereitung und Generierung der Ausgabesignale [22, S. 287, 22, S. 312, 17] . . . . .	17
3.5.	MT9V022-Bildsensors; Active-Pixel Array zur Bilddatenerfassung, analoge Ausleseschaltung (Analog Processing), 10 Bit Analog-Digital-Wandler (ADCs), Auslesesteuerung (Timing and Control) zum Ansteuern des Sensors, DSP (Digital Processing) zum Aufbereiten des Bildsignals und zum Generieren der Ausgabe sowie I <sup>2</sup> C-Businterface mit Konfigurationsregistern (Control Register) zur Konfiguration [4, S. 6] . . . . .	18
3.6.	Active-Pixel Sensor Array des MT9V022 Bildsensors mit optisch geschwärzten und ungenutzten Pixeln [4, S. 10] . . . . .	19
3.7.	Nichtlinearer Verlauf der 12- auf 10-Bit Umsetzung des ADCs im MT9V022 Bildsensor [4, S. 28] . . . . .	20



3.8.	AGC- und AEC-Einheiten zur automatischen Verstärkungsfaktor- und Belichtungszeitkontrolle des MT9V022 Bildsensors [4, S. 32] . . . . .	21
3.9.	Tiefpass der MT9V022 AEC- und AGC-Schaltungen . . . . .	23
3.10.	Aufteilung des MT9V022-Bildsensors in 25 Bereiche [4, S. 29] . . . . .	24
3.11.	Schaltung zur Schwarzwertkorrektur an der analogen Ausleseschaltung des MT9V022-Bildsensors [4, S. 30] . . . . .	25
3.12.	zur Ausgabe genutzter (grau), horizontaler und vertikaler informationsfreier Bereich (weiß) bei der Bildausgabe des MT9V022-Bildsensors [4, S. 36] . . . . .	27
3.13.	Verlauf der Resetspannung im HDR-Modus des MT9V022-Bildsensors [4, S. 26] .	29
4.1.	Mehrere I <sup>2</sup> C-Busteilnehmer, auch mit unterschiedlichen Versorgungsspannungen [19, S. 8] . . . . .	31
4.2.	Übertragung auf dem I <sup>2</sup> C-Bus mit Verzögerung der Übertragung durch den Slave nach dem Acknowledge des ersten Bytes sowie START- und STOP-Konditionen [19, S. 10] . . . . .	33
4.3.	Lesen des Registers 0xBA des MT9V022-Bildsensors (Adresse 0x5C) über den I <sup>2</sup> C-Bus; erste Übertragung enthält Adresse und Register, die zweite Übertragung die Adresse und die gelesenen Daten (0x0010) ; senkrechte Markierungen: START-, REPEATED-START und STOP-Konditionen (PicoScop-3204) . . . . .	37
6.1.	Ursprüngliche SAV-Control Block [16, S. 48] . . . . .	47
6.2.	Angepasster SAV-Control Block mit angeschlossenen Komponenten . . . . .	48
6.3.	Oszillogramm der generierten Takte; PIXCLK des MT9V022-Bildsensors; generierte 27 MHz, 54 MHz und 13,5 MHz Taktsignale (Tektronix TDS 2024C; 200 MHz; 2 GS/s) . . . . .	50
6.4.	Oszillogramm der Datenausgabe des MT9V022-Bildsensors; PIXCLK- und Datensignal (Tektronix TDS 2024C; 200 MHz; 2 GS/s) . . . . .	50
6.5.	Anbindung der Kamera an den FPGA . . . . .	52
6.6.	Vertikale und horizontale VGA-Synchronisationssignale; Pulsbreite (B und P), Backporch (C und Q), Frontporch (E und S) und aktiver Bilddatenbereich (D und R) sind definiert [14] . . . . .	53
6.7.	Interner Aufbau VGA-Controller . . . . .	56
6.8.	Zustandsautomaten des VGA-Controllers [15] . . . . .	58
6.9.	Simulationsergebnis der Testbench des VGA-Controllers . . . . .	59
7.1.	Messaufbau zur Analyse der Linsenverzeichnung . . . . .	65

7.2.	Originalbild (a) und transformiertes Bild (b) des unbekanntes Objektivs . . . . .	65
7.3.	Originalbild (a) und transformiertes Bild (b) des DSL213A Objektivs . . . . .	66
7.4.	Originalbild (a) und transformiertes Bild (b) des 32S2520N Objektivs . . . . .	66
7.5.	Originalbild (a) und transformiertes Bild (b) des 32S2120N Objektivs . . . . .	67
7.6.	Bild (a) und Histogramm (b) mit aktivierter nichtlinearer Quantisierung des ADCs im MT9V022 Bildsensor . . . . .	68
7.7.	Bild (a) und Histogramm (b) ohne aktivierter nichtlinearer Quantisierung des ADCs im MT9V022 Bildsensor . . . . .	69

# Glossar

ADC	Analog Digital Converter
AEC	Automatic Exposure Control
AGC	Automatic Gain Control
ASIC	Application-Specific Integrated Circuit
CCD	Charge Coupled Devices
CMOS	Complementary Metal Oxide Semiconductor
CPLD	Complex Programmable Logic Device
DCM	Digital Clock Manager
DMA	Direct Memory Access
DR	Dynamic Range
DSP	Digital Signal Processor
FAUST	Fahrassistentz- und Autonome Systeme
FIR	Finite Impulse Respond
FPGA	Fiel Programmable Gate Array
FSL	Fast Simplex Link
HDR	High Dynamic Range
I <sup>2</sup> C-Bus	Inter-IC-Bus
IC	Integrated Circuit
IP	Intelectual Property

LAP	Look Ahead Point
LVDS	Low Voltage Differential Standard
MDM	MicroBlaze Debug Module
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
MPMC	Multi-Port Memory Controller
NPI	Native Port Interface
PLB	Processor Local Bus
RLE	Run-Length Encoding
ROI	Region of Interest
RTL	Register Transfer Logic
SAM	SystemACE Module
SAV	System on Chip Autonomous Vehicle
SFD	Source Follower per Detector Readout
SOC	System-on-Chip
TWI	Two-Wire-Interface
XCL	Xilinx CacheLink

*Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst  
und nur die angegebenen Hilfsmittel benutzt habe.*

Hamburg, 30. August 2012

---

Jasper Salathé