



FACHHOCHSCHUL-BACHELORSTUDIENGANG

Automatisierungstechnik

Schnittstellenoptimierung des RoboCup Rescue Robots

Betreuung der Masterarbeit

DI(FH) Edlinger Raimund

Inhalt

1. Einleitung	3
2. Systemübersicht	4
2.1 IP Adresse Übersicht	5
3. GUI	5
3.1 GUI zur Steuerung des Robots	5
3.2 GUI für Kamera Server	6
3.3 GUI für Kamera Client	6
4. EMX Development Board	7
4.1 Kommunikation Operator -> EMX	8
4.1.1 Space Pilot Pro	9
4.1.2 Heartbeat Signal	11
5. FLIR Wärmebildkamera	11
5.1 RTSP [Rfc]	12
5.1.1 Protokoll-Richtlinien.....	12
5.2 SDP [Rfc1]	14
5.3 RTP/RTCP [Rfc2] [Rfc3]	14
6. Mini ITX Board	14
6.1 USB Cam.....	15
6.2 DirectShow	16
6.2.1 Architektur	16
7. 3D Kamera Efactor PMD3D - O3D200	17
8. Erkenntnisse	19
9. Literaturverzeichnis	19

Abbildungsverzeichnis

<i>Abbildung 1.1: Rescue Roboter</i>	3
<i>Abbildung 2.1: Systemübersicht</i>	4
<i>Abbildung 3.1: GUI für Steuerung</i>	6
<i>Abbildung 3.2: GUI für Kamera Server</i>	6
<i>Abbildung 3.3: GUI für Kamera Client</i>	7
<i>Abbildung 4.1: EMX Development Board²</i>	7
<i>Abbildung 4.2: Übersicht Kommunikation Operator – EMX</i>	9
<i>Abbildung 4.3: Space Pilot Pro³</i>	10
<i>Abbildung 5.1: Übersicht Kommunikation Operator – FLIR Wärmebildkamera</i>	12
<i>Abbildung 6.1: Übersicht Kommunikation Operator – Mini ITX Board</i>	15
<i>Abbildung 6.2: Logitech Sphere⁶</i>	16
<i>Abbildung 7.1: O3D200 Kamera⁷</i>	18
<i>Abbildung 7.2: Übersicht Kommunikation Operator – 3D Kamera</i>	18

1. Einleitung

Im Zuge der ersten Masterarbeit wurde am Rescue Roboter, siehe *Abbildung 1.1: Rescue Roboter*, eine Umstrukturierung der Hardware durchgeführt. Es wurde passend zur neuen Hardware die Softwaretreiber für die Steuerung des Roboters entwickelt. Des Weiteren wurde ein Treiber für die vorhandene 3D - Vision Kamera¹ entwickelt. Ziel der Masterarbeit 2 war es, die Kommunikation der vorhandenen Kamerasysteme zu optimieren bzw. die Audio und Video Streams dieser Systeme zu vereinen und auf einer geeigneten Oberfläche zu visualisieren. Dabei sollte darauf geachtet werden, ein einheitliches System zu entwickeln, was ein einfaches hinzufügen bzw. entfernen von Komponenten erleichtern soll.



Abbildung 1.1: Rescue Roboter

¹ <http://www.ifm.com/ifmat/web/dsfs!O3D200.html>

2. Systemübersicht

Abbildung 2.1 veranschaulicht die wichtigsten Komponenten. Man sieht, dass verschiedene Systeme verwendet werden, die im Verlauf der Arbeit detaillierter erklärt werden.

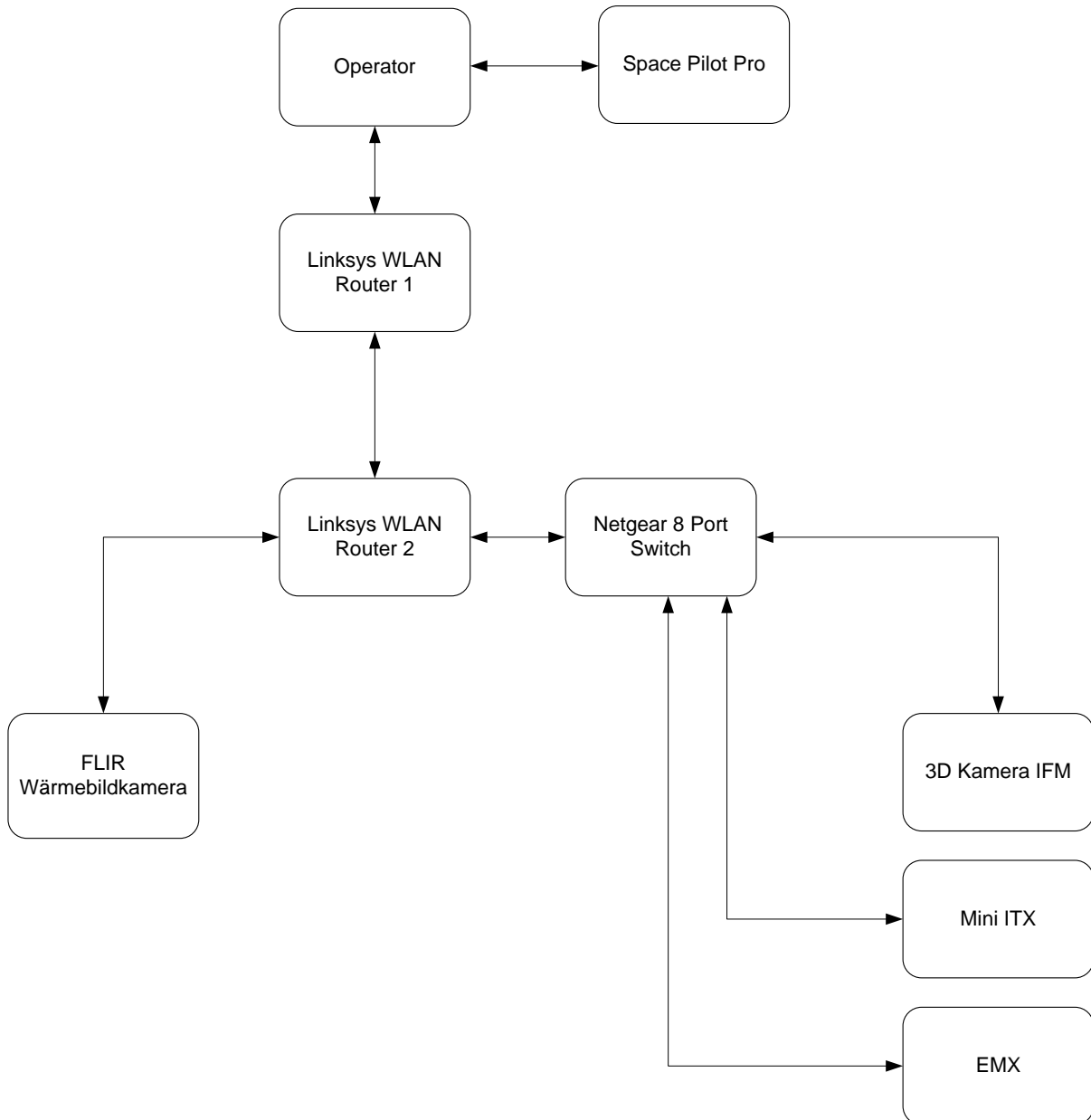


Abbildung 2.1: Systemübersicht

2.1 IP Adresse Übersicht

Im Folgenden werden die IP Adressen sämtlicher Ethernetgeräte aufgelistet. Man beachte hierbei, dass der Netgear Switch keine IP Adresse besitzt, sondern den WLAN Router 2 lediglich um Ethernet Ports erweitert.

	IP
Operator	192.168.1.3
Linksys WLAN Router 1	192.168.1.2
Linksys WLAN Router 2	192.168.1.1
Flir A320 Infrarotkamera	192.168.1.5
3D Kamera IFM	192.168.1.50
Mini ITX	192.168.1.100
Embedded Master EMX	192.168.1.200

3. GUI

Die GUI wurde mittels C# und Visual Studio 2010 erstellt. Es wurden drei Projekte erstellt.

3.1 GUI zur Steuerung des Robots

Mit dieser GUI ist es möglich die Steuerung des Roboters zu übernehmen. Dabei ist darauf zu achten, dass vor dem Start der Anwendung die Treiber für die Space Pilot 3D Maus gestartet werden, sowie eine korrekte Ethernet Verbindung zum EMX Board besteht. Mit dem Button Ping kann getestet werden, ob das Embedded Master EMX Board im Netzwerk verfügbar ist. Beim Klicken auf den Button Start wird die Kommunikation zum Board aufgebaut und es können Steuerbefehle gesendet und verarbeitet werden. Mit dem Button Flipper Ref wird eine Referenzfahrt der Flipper gestartet. Somit kann im Programm genau verfolgt werden, wo sich die Flipper befinden. Set Speed setzt die Maximalgeschwindigkeit der Servoantriebe, siehe *Abbildung 3.1: GUI für Steuerung*.

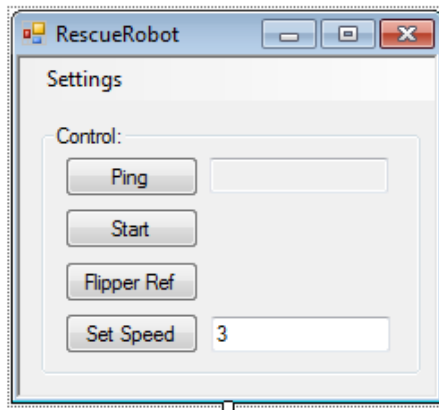


Abbildung 3.1: GUI für Steuerung

3.2 GUI für Kamera Server

Im Kamera Server werden die angeschlossenen USB Webcams erfasst und für das streamen vorbereitet. Beim Klick auf den Button Start kann mit der GUI Kamera Client das Bild gestreamt werden. Beim Klick auf den Button Stop wird die Kamerasitzung beendet und der Server geschlossen.

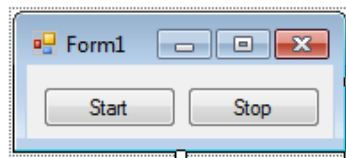


Abbildung 3.2: GUI für Kamera Server

3.3 GUI für Kamera Client

Im Kamera, siehe *Abbildung 3.3: GUI für Kamera Client*, Client kann mittels IP Adresse und Port die Verbindung zum Server aufgebaut werden. Verfügt die angeschlossene Kamera über Funktionen wie Pan oder Tilt, können diese mittels des Steuerkreuzes im Programm verändert werden. Beim Klicken auf den Button Stop wird die Verbindung beendet.

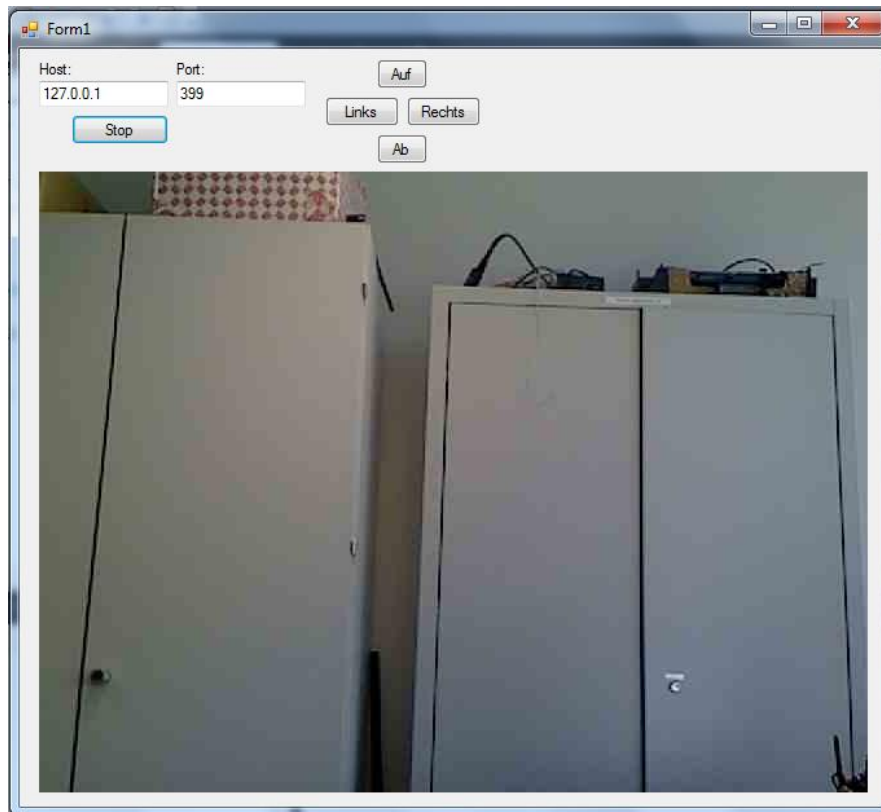


Abbildung 3.3: GUI für Kamera Client

4. EMX Development Board

Das Embedded Masters EMX Development Board² ist einfach zu implementieren und kann mittels Visual Studio 2010 und C# programmiert werden. Des weiteren gibt es eine Vielzahl an fertigen Bibliotheken für die Hardwareansteuerung.

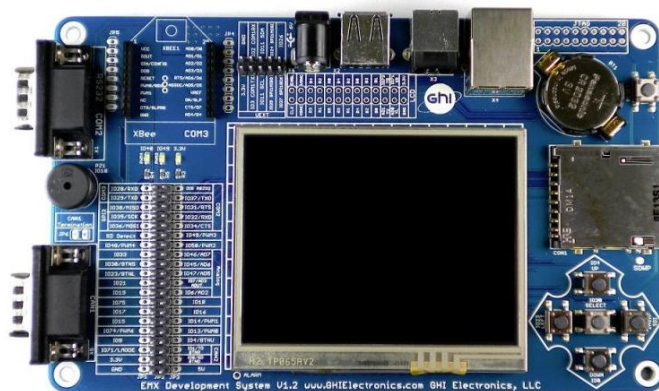


Abbildung 4.1: EMX Development Board²

² <http://www.ghielectronics.com/catalog/product/129>

Die wichtigsten Funktionen:

- .NET Micro Framework 4.1
- 72 MHz 32-bit ARM 7 Prozessor
- 320 x 240 3.5" TFT Display with touch screen
- 2 SPI Master Bus (8/16bit)
- I2C Interface
- 7 analog Inputs
- 1 analog Output
- 2 CAN Interfaces
- uvm.

4.1 Kommunikation Operator -> EMX

Das EMX Modul wird primär zur Steuerung des Rescue Roboters verwendet. Die Steuerbefehle werden vom Operator über die beiden Linksys WLAN Router und dem Netgear Switch an das EMX Board gesendet. Dort werden die eingehenden Befehle in die entsprechenden CAN Kommandos umgewandelt und mittels CAN Bus an die Servosteuerungen gesendet. Die zweite Aufgabe des EMX Boards ist es, Hardwareseitig diverse Module anzusteuern wie zum Beispiel die Endlagenschalter für die Flipperreferenz, die Berechnung des CO² Gehalts in der Luft und die Spannungsüberwachung der Akkuzellen zum Schutz gegen Tiefentladung.

Die Steuerbefehle werden von einer Space Pilot Pro 3D Maus mittels TCP und Port 666 vom Operator zum EMX Board übermittelt. Des weiteren wurde zum Schutz des Roboters ein Heartbeat Schnittstelle implementiert. Die Kommunikation hierfür erfolgt über UDP Port 666.

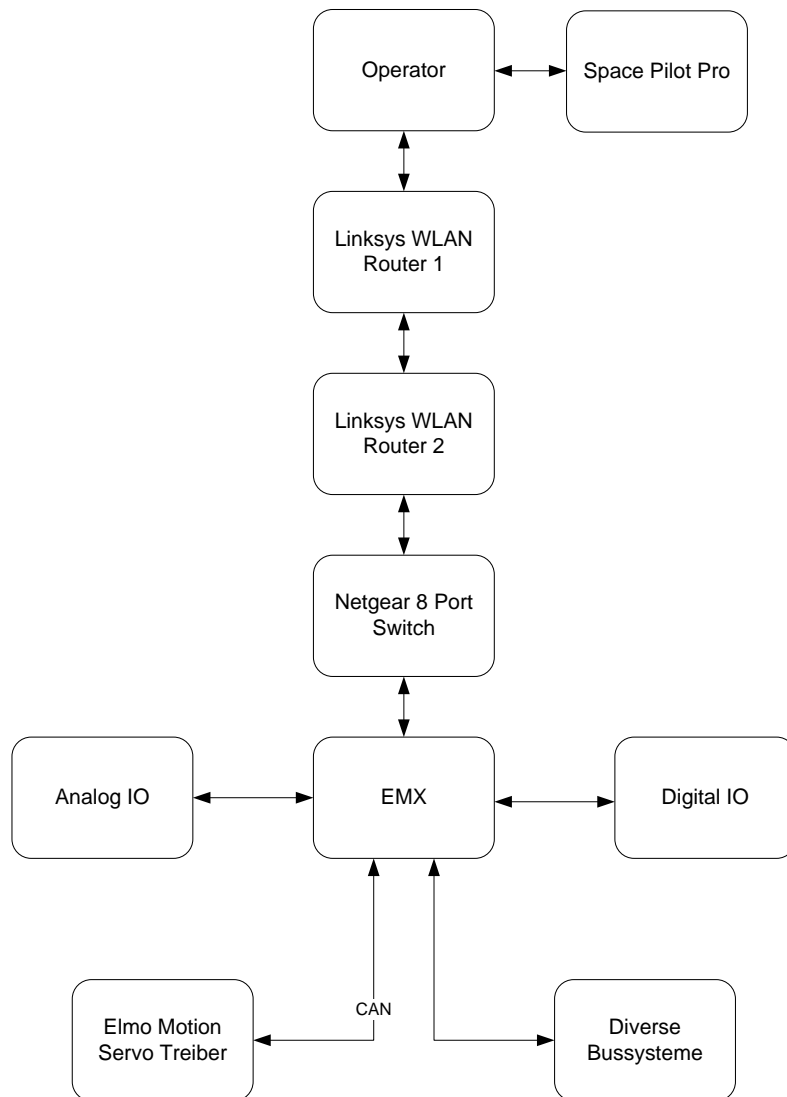


Abbildung 4.2: Übersicht Kommunikation Operator – EMX

4.1.1 Space Pilot Pro

Um die Kontrolle des Roboters zu erleichtern, wird eine 3D Maus zur Steuerung verwendet. Die Space Pilot Pro³ wird überwiegend in der 3D Konstruktion verwendet. Sie kann jedoch für eigene Anwendungen mittels C# programmiert⁴ werden.

³ <http://www.3dconnexion.com/products/spacepilot-pro.html>

⁴ <http://www.3dconnexion.com/service/software-developer.html>



Abbildung 4.3: Space Pilot Pro3

Folgende Bewegungen wurden implementiert:

- Translation Y Richtung
Vorwärts oder rückwärts fahren.
- Translation Z Richtung
Bewegung der 4 Flipper nach oben oder unten.
- Rotation
Rotation um die eigene Achse
- Flipper Referenz
Referenzfahrt der Flipper
- Set Speed
Festlegen der Geschwindigkeit der Servoantriebe

Für die Datenübertragung werden die .Net enthaltenen Sockets verwendet. Um ein Socket Objekt zu instanzieren, müssen 3 Argumente übergeben werden.

- Die AddressFamily spezifiziert das Adressierungsschema das eine Instanz der Socket Klasse benutzen kann. Das kann z.B. InterNetwork sein, welches das

IPv4 implementiert oder InterNetworkV6 für IPv6 oder Unix für Unix Adressen.

- Als zweiter Parameter wird der SocketType übergeben. Die häufig verwendeten Typen sind die Stream und Datagram Sockets. Aber auch Raw Sockets kommen oft zum Einsatz.
- Der dritte und letzte Parameter ist ProtocolType und spezifiziert den verwendeten Protokolltyp. Mögliche Argumente sind unter anderem das TCP, UDP oder IP.

Die Steuerbefehle werden zu Paketen zu je 15 Bytes verpackt und an das Embedded Master Board gesendet.

4.1.2 Heartbeat Signal

Um im laufenden Betrieb die Sicherheit zu erhöhen, wurde ein Heartbeat Signal implementiert. Hier werden im Abstand von einer Sekunde mittels UDP eine Nachricht an das Embedded Master Modul gesendet. Bricht nun die Verbindung zwischen Benutzer und Roboter unerwartet ab, bleibt das Signal aus und der Roboter beendet die laufende Bewegung. Somit ist es nicht möglich, dass sich der Roboter unkontrolliert fortbewegt.

5. FLIR Wärmebildkamera

Für die Verletztendetektion wird eine FLIR Wärmebildkamera verwendet. Dies ermöglicht eine einfache Erkennung und Auswertung von Temperaturszenen. Die A320 Wärmebildkamera unterstützt das RTSP Protokoll das im folgenden erklärt wird.

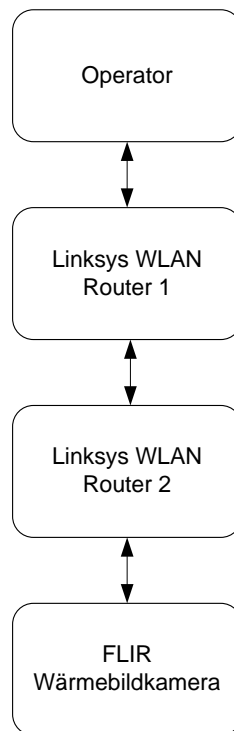


Abbildung 5.1: Übersicht Kommunikation Operator – FLIR Wärmebildkamera

5.1 RTSP [Rfc]

Das Real Time Streaming Protocol (RTSP) ist ein Netzwerk Protokoll für den Einsatz in Unterhaltungs-und Kommunikationssystemen. Das Protokoll ist für die Festlegung und Kontrolle zwischen den Endpunkten zuständig. Die Übertragung von Streaming-Daten selbst ist nicht die Aufgabe des RTSP-Protokolls.

5.1.1 Protokoll-Richtlinien

Während in mancher Hinsicht ähnlich zu HTTP, definiert RTSP Steuersequenzen, die nützlich bei der Kontrolle der Multimedia-Wiedergabe sind. Anders als http hat RTSP gewisse Zustände. Wenn es die Situation erfordert, wird eine Kennung verwendet, um gleichzeitige Sessions zu verfolgen. RTSP verwendet TCP zur Aufrechterhaltung der Verbindung und während die meisten RTSP Nachrichten vom Client zum Server gesendet werden, werden einige Befehle in die andere Richtung verschickt. Der Standard Port für das RTSP Protokoll ist 554 für UDP und TCP. Nachfolgend die wichtigsten Befehle:

- **OPTIONS**
Eine OPTIONS-Anfrage gibt zurück, welche Arten von Anfrage der Server akzeptiert.

- **DESCRIBE**
Eine DESCRIBE Anfrage enthält eine RTSP-URL (rtsp://...), und die Art der Daten, die gehandhabt werden kann. Diese Antwort enthält die Eigenschaften des Multimediadatenstroms, in der Regel im Session Description Protocol (SDP)-Format. Unter anderem werden auch die einzelnen Media-Streams aufgelistet, die mit den einzelnen URLs kontrolliert werden können. Typischerweise gibt es einen Media-Stream, jeweils für Audio und für Video.

- **SETUP**
Eine SETUP Anfrage legt fest, wie die einzelnen Media-Streams transportiert werden müssen. Sie enthält die Media-Stream-URL und einen Transport-Spezifizierer. Dies muss geschehen, bevor eine Play Anfrage gesendet wird. Der Transport-Spezifizierer umfasst in der Regel einen lokalen Port für den Empfang von RTP-Daten (Audio oder Video), und eine andere für RTCP Daten.

- **PLAY**
Eine PLAY Anfrage verursacht das Abspielen einer oder aller Media-Streams.

- **PAUSE**
Eine PAUSE Anfrage haltet vorübergehend eine oder alle Media-Streams an, mit PLAY wird die Wiedergabe fortgesetzt.

- RECORD

Die RECORD Anfrage kann verwendet werden, um am Server den Stream zu speichern.

- TEARDOWN

Eine TEARDOWN Anfrage wird verwendet, um die Sitzung zu beenden. Es stoppt alle Medien-Streams und löscht alle Session-Daten auf dem Server.

5.2 SDP [Rfc1]

Das Session Description Protocol (SDP) ist ein Format zur Beschreibung von Streaming Media Initialisierungsparametern. SDP ist für die Beschreibung von Sitzungsankündigungen, Sitzungseinladungen und Parameterverhandlungen bestimmt. SDP liefert die Medien nicht selbst, sondern ist für die Verhandlungen zwischen den Endpunkten der Medien, das Format und alle damit verbundenen Eigenschaften zuständig.

5.3 RTP/RTCP [Rfc2] [Rfc3]

Das Real-Time Transport Protocol (RTP) definiert ein einheitliches Paketformat zur Übermittlung von Audio und Video über IP-Netzwerke. RTP wird in Verbindung mit dem Real-Time Control Protocol (RTCP) verwendet. Während RTP die Medien-Streams transportiert (z.B. Audio und Video), ist RTCP für die Überwachung von Statistiken Quality of Service (QoS) zuständig.

6. Mini ITX Board

Auf dem Rescue Robot werden verschiedenste Kamerasysteme verwendet. Ziel des Mini ITX Boards ist es diverse Kamerasysteme, die über keine Ethernet Schnittstelle verfügen, zusammenzufassen und die Audio und Videodaten an den Operator zu übermitteln. So wird zum Beispiel eine einfache USB Webcam verwendet um eine onboard view des Ro-

boters zu haben. Das Streamen der Audio- und Videodaten erfolgt über TCP und Port 399. Die Steuerbefehle werden über UDP und Port 399 versendet. Als Framework wird das Microsoft eigene DirectShow⁵ verwendet, was im folgenden weiter erklärt wird.

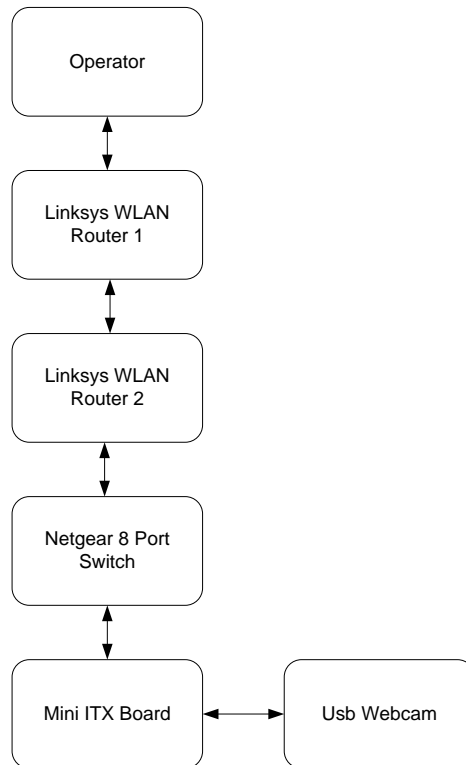


Abbildung 6.1: Übersicht Kommunikation Operator – Mini ITX Board

6.1 USB Cam

Für Testzwecke wurde eine Logitech Sphere⁶ verwendet. Sie verfügt über eine Auflösung von 1280 x 960. Außerdem unterstützt sie Funktionen wie Pan, Tilt und Zoom, was eine Rundumsicht im Raum möglich macht.

⁵ <http://msdn.microsoft.com/en-us/library/dd375454%28v=vs.85%29.aspx>

⁶ <http://www.logitech.com/en-in/webcam-communications/webcams/devices/3480>



Abbildung 6.2: Logitech Sphere⁶

6.2 DirectShow

Ziel war es, dass jede Art von Usb Webcam an das ITX Board angeschlossen werden kann und über diese dann Audio und Video zu streamen. Dies kann einfach mit dem Microsoft eigenem Framework DirectShow realisiert werden. DirectShow ist ein Multimedia-Framework für Software-Entwickler, mit dem man verschiedene Operationen mit Mediendateien oder Medienstreams durchführen kann.

6.2.1 Architektur

DirectShow unterteilt komplexe Multimedia Aufgaben in eine Folge von grundlegenden Bearbeitungsschritte auch als Filter bezeichnet. Jeder Filter hat Eingangs und/oder Output-Pins, welche verwendet werden, um den Filter mit anderen Filter zu verbinden. Es gibt drei Arten von Filtern:

- Source Filter

Dieser bildet den Grundfilter der Daten. Zum Beispiel lesen der Rohdaten aus einer Medien Datei.

- Transform Filter

Dieser bildet die Transformation der Daten. Zum Beispiel ist eine Transformation das Hinzufügen von Text.

- Renderer Filter

Diese erzeugt die Daten. Zum Beispiel sendet er das Audiosignal an die Soundkarte.

Während des Rendering-Prozesses sucht der Filter Graph in der Windows-Registrierung nach registrierten Filtern und baut deren Graphen auf. DirectShow stellt außerdem Funktionen zur Verfügung die es ermöglichen die bereits genannten Funktionen wie Pan Tilt und Zoom zu steuern.

7. 3D Kamera Efector PMD3D - O3D200⁷

Der efector pmd 3d ist ein industrieller 3D-Sensor, der auf einem Blick Objekte räumlich bewertet. Die Lichtlaufzeitmessung und die Auswertung sind zusammen auf einem Sensor-Chip integriert. Jeder Bildpunkt dieser Chip-Matrix wertet den Abstand zum Objekt aus. Das Abbild des Objektes auf der Chip-Matrix und die zugehörigen Abstandswerte entsprechen einem 3D-Bild. Diese Technologie ermöglicht die detaillierte Bewertung verschiedener Zustände des Objektes oder der Szene über das räumliche Erfassen von Volumen oder Abstand.

Die pmd 3d Kamera erfasst Szenen und Objekte auf einen Blick in ihren räumlichen Dimensionen. Im Gegensatz zu Laserscannern kommt sie ohne bewegliche Komponenten aus und ist dadurch robust und verschleißfrei. Neben dem 3D-Abstandsbild liefert die Kamera ein Graubild der Szene. Die Kombination dieser Bilder bietet Integratoren die Möglichkeit der freien Programmierung applikationsspezifischer Anwendungen mittels eines Software Development Kits.

⁷ <http://www.ifm.com/ifmat/web/dsfs!O3D200.html>



Abbildung 7.1: O3D200 Kamera7

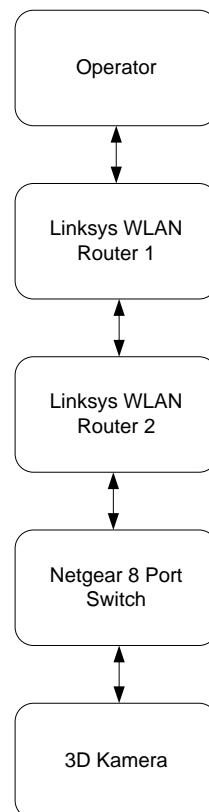


Abbildung 7.2: Übersicht Kommunikation Operator – 3D Kamera

Im ersten Schritt wurde für die 3D Kamera ein Softwaretreiber für die Ansteuerung programmiert. Die Kommunikation vom Operator zur 3D Kamera erfolgt über TCP und Port 8080. Ziel der 3D Kamera wird es sein, den Rescue Roboter vor Kollisionen zu schützen, bzw. im Folgeschritt einen autonomen Ausgleich der Flipperstellung während der Fahrt zu gewährleisten.

8. Erkenntnisse

Der Rescue Robot beinhaltet eine Fülle an Kommunikationsarten. Ziel soll es sein, ein modulares System aufzubauen, dass es dem Anwender ermöglicht, diverse Komponenten hinzuzufügen, auszutauschen oder zu entfernen. Des weiteren sollte der programmierte Code einfach zu verstehen und zu warten sein. Dies kann nur geschehen, in dem man sich an gewisse Kommunikationsstandards hält. In diesem Semester wurde ein erster Schritt in diese Richtung getätigt und ein Standard implementiert, der die Audio und Video Daten auf effektivste Weise streamt. Ziel wird es sein, diesen Standard auf die noch nicht implementierten Systeme zu erweitern und so ein einheitliches System zu schaffen.

9. Literaturverzeichnis

[Rfc] Rfc2326 - Real Time Streaming Protocol (RTSP). [Online] <http://tools.ietf.org/html/rfc2326>.

[Rfc2] Rfc3550 - RTP: A Transport Protocol for Real-Time Applications. [Online] <http://tools.ietf.org/html/rfc3550>.

[Rfc4] Rfc3556 - Session Description Protocol (SDP) Bandwidth Modifiers. [Online] <http://tools.ietf.org/html/rfc3556>.

[Rfc3] Rfc3605 - Real Time Control Protocol (RTCP). [Online] <http://tools.ietf.org/html/rfc3605>.

[Rfc1] Rfc4566 - SDP: Session Description Protocol. [Online] <http://tools.ietf.org/html/rfc4566>.