

Design- und Steuerungsaspekte humanoider Roboter

Dissertation

zur Erlangung des akademischen Grades

**Doktoringenieur
(Dr.-Ing.)**

von M.Sc. Artem Rudskyy
geb. am 13.11.1984 in Donetsk (Ukraine)

genehmigt durch die Fakultät für Elektrotechnik und Informationstechnik
der Otto-von-Guericke-Universität Magdeburg

Gutachter:

Univ.-Prof. Dr.-Ing. habil. Frank Palis

Prof. Dr.-Ing. Olexandr Khorkhordin

Promotionskolloquium am 11. April 2012

Abstrakt

Mobile anthropomorphe Schreitroboter finden immer häufiger Einsatz im industriellen Umfeld, in Serviceaufgaben, bei planetaren Erkundungen, im Haushalt u. ä. und werden zunehmend in die menschliche Gesellschaft eingegliedert. Trotz intensiver Forschungen auf dem Gebiet der Robotik, existiert eine Reihe von offenen Fragen, die die Entwicklung der robusten und flexiblen mechatronischen Konstruktion, der echtzeitfähigen Steuerungssysteme, energieoptimalen Bewegungsabläufen, sowie der akzeptablen Mensch-Roboter-Interaktion betreffen.

Vom besonderen Interesse für die Menschen sind die anthropomorphen zweibeinigen kontaktfreudigen Roboter. Einerseits eignen sich diese am besten für die menschliche Umgebung und finden eine erhöhte Akzeptanz in den Bereichen der Mitarbeit, Interaktion und Koexistenz mit den Menschen. Andererseits tragen diese zum besseren Verständnis über die menschliche Natur bzw. dem Wesen (Verhalten, Emotionen, Gehen, Laufen usw.) bei und ermöglichen folglich eine Verbesserung entsprechender Mechanismen, wie interaktiv gesteuerte Haushaltgeräte, medizinische Prothesen und Exoskelette.

Die Interaktion und Koexistenz der humanoiden Roboter mit den Menschen ist eine Aufgabe, die immer aktueller wird, weil in einer menschlichen Umgebung der Roboter nicht nur als Spielzeug sondern auch zunehmend als Helfer betrachtet wird. Um ein hohes Level menschlicher Akzeptanz zu erreichen, müssen sie unter anderem die intuitive Sprache der Emotionen und Gesten beherrschen. Diese nonverbale Sprache ist dem Menschen von Geburt an gegeben und bildet eine wesentliche Komponente der Interaktion. Die Implementierung eines Robotersteuerungssystems, welches durch sein Verhalten und seine Emotionen mit dem Mensch interagieren und damit die Roboterakzeptanz erreichen kann, ist einer der Schwerpunkte dieser Arbeit.

Die vorliegende Arbeit widmet sich ebenfalls der systematischen Entwicklung der mechatronischen Oberkörperstruktur des mobilen humanoiden Schreitroboters ROTTO sowie der Entwicklung seines Steuerungssystems zur Mensch-Roboter-Interaktion. Hier wird auf der

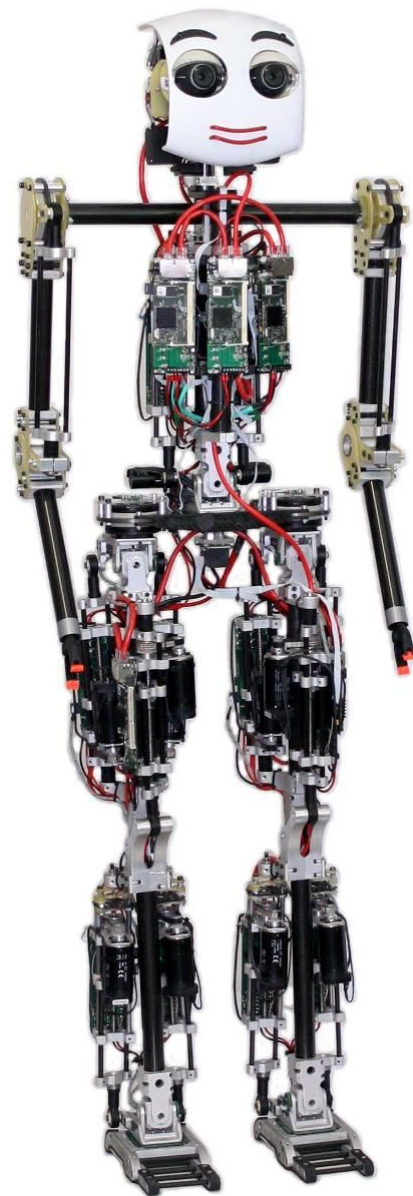


Bild 0.1 – Der mobile, humanoide, kontaktfreudige Roboter ROTTO

einen Seite eine besondere Softwarearchitektur vorgeschlagen. Sie ermöglicht eine effiziente und fehlerfreie Entwicklung, eine umfangreiche Modellierung und eine schnelle Inbetriebnahme der Robotersteuerung. Auf der anderen Seite wird eine Hardwarearchitektur eingesetzt, die flexible Technologieplattformen, effiziente Netzwerke und leistungsfähige Rechner umfasst. Diese Software- und Hardwarearchitekturen sind einer der Schwerpunkte dieser Arbeit, welche an der Implementierung einer Mensch-Roboter-Interaktion getestet werden kann.

Der entwickelte mobile, humanoide, kontaktfreudige Schreitroboter ROTTO ist im Bild 0.1 dargestellt. Die entworfene Roboterkonstruktion zeichnet sich durch eine einfache, aber dennoch für viele Aufgaben (die Mensch-Interaktion, die Fortbewegung bzw. Lokomotion u. ä.) ausreichende Kinematik mit optimalen Verhältnissen und größeren Arbeitsbereichen aus. Sowohl die mechanische Struktur als auch das Steuerungssystem des Roboters sind modular aufgebaut. Der entwickelte Prototyp des zweibeinigen Roboters ist somit eine solide Testplattform für verschiedene Steuerungs- und Regelungsaufgaben.

Zu den wichtigsten Ergebnissen der vorliegenden Arbeit gehört der experimentelle Nachweis der hohen Akzeptanz eines mechanischen Wesens -wie es der zweibeinige, mobile Schreitroboter ROTTO darstellt-, die dank der spezifisch gestalteten Außenansicht und dem entwickelten Emotionssteuerungssystem erreicht wird.

Danksagung

Die vorliegende Arbeit entstand während meiner Tätigkeit als Doktorand am Institut für Elektrische Energiesysteme der Otto-von-Guericke-Universität Magdeburg. Mein besonderer Dank gilt Herrn Prof. Dr.-Ing. habil. Frank Palis für die großzügige Unterstützung und das stete Interesse sowie für die intensive Betreuung dieser Arbeit. Prof. Palis hat sich ferner sehr stark für die persönlichen Interessen der Doktoranden eingesetzt.

Den Herrn Prof. Dr. Sc. techn. Ulrich Schmucker und Prof. Dr.-Ing. habil. Frank Palis möchte ich für die allseitige Unterstützung bei der Durchführung der Projekte und bei der Entwicklung des Roboterprototyps meinen besten Dank aussprechen.

Der innige Dank gilt Herrn Prof. Dr.-Ing. Kalaschnikow Viktor Iwanowitsch für die allseitige großzügige Unterstützung während meines Studiums und der anschließenden Forschungsarbeit.

Beim Herrn Dipl.-Ing. Hans-Jürgen Hilscher möchte ich mich verbindlichst für die enge Zusammenarbeit und großzügige Unterstützung bei der Realisierung nicht nur der Kommunikationssysteme sondern auch der neuen Automatisierungsplattformen bedanken.

Mein besonderer Dank gilt Herrn Dr.-Ing. habil. Anatoli Schneider für die praktischen Ratschläge, Dr.-Ing. Yuriy Zavgorodniy für die wissenschaftliche Unterstützung und fruchtbare Mitentwicklung der Konzepte, Dr.-Ing. Vadym Rusin für die nützlichen Bemerkungen und Ratschläge, M.Sc. Andriy Melnykov und M.Sc. Vadim Belous für die Mithilfe in der Entwicklung der mechanischen Roboterkonstruktion, M.Sc. Mykhaylo Konyev, Dipl.-Ing. Dipl.-Inf. Sergiy Dzantimirov und M.Sc. Andrej Telesh für die fachliche Hilfe, Dr.-Ing. Yuriy Tsepkovskiy für die Einführung in die digitale Bildverarbeitung. Bei allen Genannten bedanke ich mich für die langjährige und fruchtbare Zusammenarbeit, für die zahlreichen Diskussionen und in erster Linie für das freundliche Arbeitsklima.

Ich möchte mich ganz herzlich bei meiner lieben Frau Olga, meinen Eltern und meiner großen Familie für die stetige und geduldige Unterstützung während der Promotionszeit bedanken.

Schließlich möchte ich mich bei allen Mithelfern, die an meinen Versuchen mit dem Roboter ROTTO teilgenommen haben, bedanken.

Abkürzungsverzeichnis

ALU	<i>Eine arithmetisch-logische Einheit (engl. arithmetic logic unit)</i>
BFB	<i>Basic Function Blocks</i>
BFD	<i>engl. Binary File Description</i>
CAD	<i>Rechnergestützter Entwurf (engl. computer-aided design)</i>
CCD	<i>Ladungsgekoppeltes Bauteil (engl. Charge-coupled Device)</i>
CFB	<i>Composite Function Blocks</i>
CMOS	<i>Komplementärer Metall-Oxid-Halbleiter(engl. Complementary Metal Oxide Semiconductor)</i>
CORBA	<i>Allgemeine Architektur für Vermittler von Objekt-Anforderungen (engl. Common Object Request Broker Architecture)</i>
CPU	<i>Der Hauptprozessor (engl. central processing unit)</i>
DC-Motor	<i>Gleichstrommaschine</i>
DOF	<i>Freiheitsgrad (engl. Degree of Freedom)</i>
DSP	<i>Digitaler Signalprozessor</i>
EMV	<i>Elektromagnetische Verträglichkeit</i>
FPGA	<i>Programmierbarer Integrierter Schaltkreis (engl. Field Programmable Gate Array)</i>
GCC	<i>engl. GNU compiler collection</i>
GDB	<i>engl. GNU debugger</i>
HAL	<i>Hardwareabstraktionsebene (engl. Hardware Abstraction Layer)</i>
HiL	<i>engl. Hardware-in-the-Loop Simulation</i>
HW	<i>Hardware</i>
I/O	<i>Eingabe/Ausgabe (engl. Input/Output)</i>
I ² C	<i>engl. Inter-Integrated Circuit (serieller Datenbus)</i>
IDE	<i>Eine integrierte Entwicklungsumgebung (engl. integrated development environment)</i>
ISA	<i>engl. Industry Standard Architecture (ein Computerbus-Standard)</i>
LLVM	<i>engl. Low level virtual machine</i>
PC	<i>Persönlicher Computer (engl. Personal Computer)</i>
PCI	<i>engl. Peripheral Component Interconnect (ein Bus-Standard zur Verbindung von Peripheriegeräten)</i>
PWM	<i>Puls-Breite-Modulation (engl. Puls- Width Modulation)</i>
RCP	<i>engl. Rapid Control Prototyping</i>
SiL	<i>engl. Software-in-the-Loop Simulation</i>
SPI	<i>engl. Serial Peripheral Interface</i>
SPS	<i>Speicherprogrammierbare Steuerung (engl. Programmable Logic Controller - PLC)</i>
SW	<i>Software</i>
UART	<i>engl. Universal Asynchronous Receiver Transmitter</i>
USB	<i>engl. Universal Serial Bus (serielles Bussystem)</i>
WLAN	<i>Drahtloses lokales Netzwerk (engl. Wireless Local Area Network)</i>
xPEC	<i>Protocol Execution Controller</i>
xPIC	<i>Peripheral Interface Controller</i>

Bilderverzeichnis

Bild 0.1 – Der mobile, humanoide, kontaktfreudige Roboter ROTTO.....	II
Bild 1.1 – MIT-Roboters: COG (a) und Kismet (b).....	5
Bild 1.2 – MIT-Roboters: Merz (a) und Leonardo (b).....	6
Bild 1.3 – ASIMO (a), KOBIAN (b) und DB (c)	9
Bild 1.4 – KATARINA (a), SLAIR (b), SLAIR2 (c) und ANTON (d).....	13
Bild 2.1 – CAD Entwurf und reellen Außenansicht des Roboters ROTTO	17
Bild 2.2 – Mit steigender Ähnlichkeit zum Menschen steigt die Roboterakzeptanz, ausschließlich der Phase 3, aus der sich das Phänomen „unheimliches Tal“ ergibt: 1 – industrielle Roboter; 2 – soziale und anthropomorphe Roboter; 3 – „unheimliches Tal“: Roboter mit starker Menschähnlichkeit oder kranke Menschen; 4 – gesunde Menschen	19
Bild 2.3 – Universeller integrierter Linearantrieb	20
Bild 2.4 – Konstruktive Umsetzung des Hüftgelenks	21
Bild 2.5 – Konstruktive Umsetzung des Kniegelenks.....	21
Bild 2.6 – Konstruktive Umsetzung des Sprunggelenks (a) und des Fußes (b).....	22
Bild 2.7 – Konstruktive Umsetzung des Beckens	23
Bild 2.8 – CAD-Entwurf und Realisierung (rechts) des Roboteroberkörpers ROTTO	24
Bild 2.9 – Konstruktion des Kreuzbeingelenks: a – Körperebenen; b – Kinematik des Kreuzbeingelenks; c, d – CAD-Entwurf, Front- und Rückansicht (1 – Befestigungselementen, 2 – universeller Linearantrieb, 3 – Kohlefaserrohr bzw. Körperskelett, 4 – Schubstangen, 5 – Kugelgelenk).....	25
Bild 2.10 – Konstruktion des Brustgelenks: a – Kinematik des Kreuzbeingelenks; b – CAD-Entwurf (1 – Hals, 2 – magnetischer Winkelsensor, 3 – Kohlefaserrohr des Schlüsselbeines, 4 – Befestigungselement der Schulter, 5 – Getriebe, 6 – Synchronmotor, 7 – Befestigungselement für die Ankopplung des Kreuzbeingelenkes)	26
Bild 2.11 – Konstruktion des Schultergelenks: a – Kinematik; b – Kinematik zur Berechnung der Rückwärtstransformation; c – CAD-Entwurf (1 – Schulterlager, 2 – magnetischer Winkelsensor, 3 – Kohlefaserrohr der	

Schulter, 4 – Befestigungselement aus Glasfaserkunststoff, 5 – Schlüsselbein, 6 – Kugelgelenk, 7 – Linearantrieb, 8 – Kardangelenk)	27
Bild 2.12 – Ellbogengelenk: a – Kinematik; b – CAD-Entwurf (1 – Schulter, 2 – Halteschraube, 3 – Fixierungseinschnitt des Ellbogens, 4 – Kohlefaserrohr des Unterarmes, 5 – Hand).....	28
Bild 2.13 – Dreieckiger McCloud’s Designraum für die Robotergesichter: 1 – Dalek [56], 2 – R2D2 [57], 3 – DB [19], 4 – COG [21], 5 – Kismet [5], 6 – Infanoid [58], 7 – Nuvo [59], 8 – Kobian [62], 9 – Asimo [60], 10 – Repliee Q2 [64], 11 – K-Bot [63], 12 – Kaspar [27], 13 – Merz [6], 14 – Papero [61], 15 – ROTTO	29
Bild 2.14 – CAD-Entwurf und Realisierung des Kopfes des interaktiven Roboters ROTTO.....	30
Bild 2.15 – Halskreuzgelenk: a – Kinematik des Kreuzgelenkes; b – CAD-Entwurf (1 – Hals, 2 – Servoantriebe, 3 – Kopfkopplung, 4 – Zahnräder)	30
Bild 2.16 – Aufbau des menschlichen Auges (Bild aus [65]): 1 – innerer gerader Muskel, 2 – oberer gerader Muskel, 3 – oberer schräger Muskel, 4 – temporaler gerader Muskel, 5 – unterer gerader Muskel, 6 – unterer schräger Muskel	32
Bild 2.17 – Augengelenk: a – Antriebstruktur und Kinematik des Kugelgelenkes; b – CAD-Entwurf; c – Rückansicht (1 – Windenkopf, 2 – Gelenkpfanne, 3 – Gelenkkopf (Videokamera bzw. Auge), 4 – Vorspannfeder, 5 – hochfestes Polyethylen-Faser-Seil)	32
Bild 2.18 – Augenbraue: a – Kinematik; b – CAD-Entwurf (1 – Hebelwerk, 2 – Zugstange, 3 – Braue, 4 – Servoantrieb, 5 – Stift)	33
Bild 2.19 – Lippenkonstruktion: a – Lippenstruktur und Kinematik; b – CAD-Entwurf (1 – Servoantrieb, 2 – gekrümmter Metallstab, 3 – biegsames Silikon-Rohr bzw. Lippe)	34
Bild 2.20 – Augenlid: a – Kinematik; b – CAD-Entwurf (1 – Hebelwerk, 2 – Zugstange, 3 – Lid, 4 – Servoantrieb, 5 – Auge bzw. Videokamera).....	35
Bild 2.21 – Gesamter Kopfaufbau, CAD-Entwurf: a – dreidimensionale Frontansicht; b – dreidimensionale Rückansicht; c – seitliche Ansicht; d – Frontansicht (1 – Polyoxymethylen-Gerüst, 2 – Glasfaserkunststoff-Gerüst, 3 – zwei Ausgleichfedern, 4 – Hals, 5 – Halskreuzgelenk, 6 – Augenbrauen, 7 – Augenlider, 8 – Augegelenk, 9 – Lippen)	36
Bild 2.22 – Berechnung der Steife der Ausgleichfeder: a – seitlicher Ansicht mit den charakteristischen Punkten (A – Kopfdrehpunkt, B – Massenmittelpunkt des Kopfes, G_k – Kopfmasse, E und H – Federfixierungspunkte, M_k – Kippmoment); b – keine Kopfdrehung ($\beta = 0$) bzw. Ausgangsposition; c – der Kopf ist in der Sagittalebene um Winkel β geneigt.....	37
Bild 2.23 – Abhängigkeit der Ausgleichkraft F_f von der Federlänge	38
Bild 2.24 – Kinematische Struktur des Roboters ROTTO.....	40

Bild 2.25 – Abmessungen des Kopfes des Roboters ROTTO	41
Bild 3.1 – Informations- und Energieflüsse (IF und EF) in einem mechatronischen System	43
Bild 3.2 – Entwurfsschritte (V-Modell) für einen mobilen Roboter als mechatronisches System	45
Bild 3.3 – Werkzeugkette zur Entwicklung der komplexen Algorithmen für ein mechatronisches System.....	47
Bild 3.4 – Von einer Idee zum Produkt durch graphische Softwareentwicklung	48
Bild 3.5 – <i>4DIAC-DIE</i> , Aufbaubeispiel mit den Daten- und Ereignisflüssen.....	51
Bild 3.6 – IEC61499 für verteiltes Rechnen im mechatronischen Robotersystem.....	52
Bild 3.7 – Softwarearchitektur in der Softwareentwicklung.....	53
Bild 3.8 – Struktur und Werkzeuge des xPIC C-Compilers	54
Bild 3.9 – Eclipse-Entwicklungsumgebung: Entwicklungs- (links) und Debug-Ansichten (rechts und unten)	57
Bild 3.10 – Debug eines Prozesses.....	58
Bild 3.11 – Zentrale Steuerungsstruktur	59
Bild 3.12 – Dezentrale Steuerungsstruktur	61
Bild 3.13 – Erweiterte Hardware- und Steuerungsstruktur für den humanoiden Roboter ROTTO (Technologieplattformen für die Antriebe: links–NetX500, rechts–NetX10)	62
Bild 4.1 – Ebenen der Steuerung des Roboters ROTTO.....	66
Bild 4.2 – Allgemeine Struktur zum Aufbau der Steuerung	68
Bild 4.3 – Modell der Theory of Mind von Leslie	70
Bild 4.4 – Modell der Theory of Mind von Baron-Cohen	71
Bild 4.5 – Allgemeine Struktur der sozial-emotionalen Steuerung für den Roboter ROTTO.....	72
Bild 4.6 – Struktur der Stimuli in der sozial-emotionalen Robotersteuerung.....	73
Bild 4.7 – anregend-temporärer Ursprung des internen Stimulus.....	75
Bild 4.8 – anregend-kumulativer Ursprung des internen Stimulus	76
Bild 4.9 – Struktur des Verhaltens in der sozial-emotionalen Robotersteuerung	77
Bild 4.10 – Einheitliches Modul für eine Emotion	78

Bild 4.11 – Neutrales Gesicht (0%) und Emotion „Freude“ mit der mittleren (30% und 60%) und der ausgeprägten (100%) Intensität	78
Bild 4.12 – Entscheidungssystem und detaillierte Struktur der Emotionssteuerung für den Roboter ROTTO	81
Bild 4.13 – Das allgemeine strukturelle Schema des emotionalen Steuerungssystems für die Mensch-Roboter-Interaktion.....	83
Bild 4.14 – Das strukturelle Schema der Servoebene	83
Bild 4.15 – Das strukturelle Schema der Bildverarbeitung für den Roboter ROTTO	84
Bild 4.16 – Das Schema der Berechnung der mit einem Mensch verbundenen Ereignisse und Zustände	84
Bild 4.17 – Das Schema der Berechnung der mit einem Gegenstand verbundenen Ereignisse und Zustände.....	85
Bild 4.18 – Das Schema der Implementierung des Ereignisses „schnelle Annäherung“	86
Bild 4.19 – Erkennung der periodischen Bewegung.....	86
Bild 4.20 – Experiment zur Erkennung der periodischen Bewegung	87
Bild 4.21 – Realisierung der anregend-temporären internen Stimuli: strukturelles Schema (a), Verlauf beim Anregen des Stimulus (b), initiativer und interaktiver Stimuli ohne Erregung(c)	88
Bild 4.22 – Realisierung der anregend-kumulativen erholsamen Stimuli: strukturelles Schema (a), Verlauf bei der Anregung des Stimulus (b).....	89
Bild 4.23 – Das strukturelle Schema des Teilsystems für das Objektsuchen (a), System der Blickrichtung (b) und Winkelsollwerte (c).....	91
Bild 4.24 – Das strukturelle Schema des Teilsystems für das Objekverfolgen (a), PI-Regler mit der „anti-wind-up“ Korrektur (b), System zur Reglereinstellung (c) und Fokussierungsverlauf (d)	92
Bild 4.25 – Das strukturelle Schema des Teilsystems zeigt das Augenblinzeln (oben) und die erzeugten Signale (unten)	93
Bild 4.26 – Signale des Teilsystems der Atmung und der Lippenpositionen	93
Bild 4.27 – Implementierung des Entscheidungssystems	94
Bild 5.1 – Umfrage zum Versuch zur Emotionszuordnung (1 – Freude, 2 – Trauer, 3 – Schreck, 4 – Unzufriedenheit, 5 – Staunen, 6 – Zorn)	98
Bild 5.2 – Ergebnisse der Umfrage zur Emotionszuordnung (a – Freude, b – Staunen, c – Trauer, d – Unzufriedenheit, e – Schreck, f – Zorn)	99
Bild 5.3 – Mensch-Roboter-Interaktion	101

Bild 5.4 – Stimuli des Roboters während einer Mensch-Roboter-Interaktion.....	102
Bild 5.5 – Emotionen und das Verhalten des Roboters während einer Mensch-Roboter-Interaktion.....	103
Bild 5.6 – Umfrage zur Untersuchung der Akzeptanz des Roboters ROTTO.....	105
Bild 5.7 – Allgemeine Wahrnehmung des Roboters ROTTO	106
Bild 5.8 – Wahrnehmung des Roboterhaltens.....	107
Bild 5.9 – Deutlichkeit der Wünsche des Roboters	107
Bild 5.10 – Bewertung des Robotercharakters.....	108
Bild 5.11 – Emotionalität des Roboters	109
Bild 5.12 – Verständlichkeit der Roboteremotionen in der Interaktion.....	109
Bild 5.13 – Zugeordnetes Alter des Roboters	110

Tabellenverzeichnis

Tabelle 2.1 – Zuweisung der Lippenbewegungen zur Emotion.....	34
Tabelle 2.2 – Konstruktive Begrenzung der Arbeitsbereiche für die Oberkörpergelenke.....	41
Tabelle 4.1 – Einzelheiten des Aufbaues der Emotionen.....	89
Tabelle 4.2 – Ein Beispiel für eine Tabelle der aktiven Stimulussätze.....	94
Tabelle 4.3 – Ein Beispiel für die Datenbank des Wissens: die Tabellen der Zusammenhänge und der Aktionssätze	95

Inhaltsverzeichnis

1	Einleitung und Aufgabenstellung.....	1
1.1	Einführung.....	1
1.2	Herausforderungen in der Steuerung von Schreitrobotern	2
1.3	Stand der Technik.....	4
1.3.1	Kontaktfreudige Roboter.....	4
1.3.2	Anthropomorphe Roboter	7
1.3.3	Software- und Hardwarearchitektur in der Robotik.....	10
1.3.3.1	Kommunikative Roboter	10
1.3.3.2	Schreitroboter der Universität Magdeburg.....	11
1.4	Ziel und Aufbau der Arbeit	15
2	Die mechanische Konstruktion des Roboters ROTTO	17
2.1	Voraussetzungen	17
2.2	Unterkörper	19
2.2.1	Linearantrieb	19
2.2.2	Beine	20
2.2.3	Becken.....	22
2.3	Torso und Arme	23
2.3.1	Kreuzbeingelenk	24
2.3.2	Brustgelenk	26
2.3.3	Schultergelenk.....	27
2.3.4	Ellbogengelenk.....	28
2.4	Kopf.....	28
2.4.1	Differentiales Halskreuzgelenk	30
2.4.2	Augenkonstruktion	31
2.4.3	Augenbrauen	33
2.4.4	Lippenflexibilität.....	33
2.4.5	Augenlider.....	35
2.4.6	Gesamter Kopfaufbau	35
2.5	Technische Daten der Roboterkonstruktion	39
2.6	Zusammenfassung.....	42

3	Steuerungssystem des Roboters ROTTO	43
3.1	Struktur und Komponenten des Steuerungssystems.....	44
3.2	Aspekte der Softwareentwicklung für mobile Roboter.....	46
3.2.1	Graphische Softwareentwicklung	48
3.2.1.1	RCP-fähige Software.....	49
3.2.1.2	Automatisierungsstandard für verteiltes Rechnen	50
3.2.1.3	Softwarearchitektur im Entwicklungsprozess für den Roboter ROTTO.....	53
3.2.2	Hardwarespezifische Codeumwandlung	54
3.2.2.1	Front- und Backend des Codeumwandlers	55
3.2.2.2	Prozessorcode.....	55
3.2.2.3	Code-Bibliotheken	56
3.2.3	Fehlerbeseitigung	56
3.2.3.1	Debug im Programmcode	56
3.2.3.2	Debug eines Prozesses	58
3.3	Hardwarestruktur.....	58
3.3.1	Zentrale Steuerungsstruktur	59
3.3.2	Dezentrale Steuerungsstruktur	60
3.3.3	Architektur der Hardware und Steuerungsstruktur für den humanoiden Roboter ROTTO	61
3.4	Zusammenfassung.....	64
4	Aspekte der Mensch-Roboter-Interaktion.....	65
4.1	Aufgaben- und Strukturhierarchie der Robotersteuerung	65
4.1.1	Aufgabenspezifische Ebenen der Robotersteuerung.....	65
4.1.2	Allgemeiner Aufbau der Steuerung	67
4.2	Aufgaben der Mensch-Roboter-Interaktion	69
4.2.1	Theory of Mind	69
4.2.2	Steuerung für den Roboter ROTTO	72
4.2.2.1	Stimuli	73
4.2.2.2	Das Verhalten.....	76
4.2.2.3	Entscheidungssystem	80
4.3	Praktische Implementierung.....	82
4.3.1	Externe Stimuli.....	84
4.3.2	Interne Stimuli.....	87
4.3.3	Emotionales Verhalten	89
4.3.4	Bewusste Aktionen.....	91
4.3.5	Unbewusste Aktionen	93
4.3.6	Entscheidungssystem	94
4.4	Zusammenfassung.....	95

5	Experimentelle Ergebnisse mit dem humanoiden Roboter ROTTO	97
5.1	Untersuchung der Anwendbarkeit der Roboteremotionen	97
5.1.1	Beschreibung des Versuches zur Emotionszuordnung	97
5.1.2	Auswertung der Ergebnisse.....	99
5.2	Nachbildung des emotionalen Verhaltens und Akzeptanz der Mensch-Roboter-Interaktion	100
5.2.1	Beschreibung des Versuches zur Mensch-Roboter-Interaktion	101
5.2.2	Auswertung der Ergebnisse.....	106
5.3	Zusammenfassung	110
6	Zusammenfassung und Ausblick	111
6.1	Zusammenfassung	111
6.2	Ausblick	114
7	Datenbank des Wissen für den Roboter ROTTO	115
8	Literaturverzeichnis	119

Kapitel 1

Einleitung und Aufgabenstellung

1.1 Einführung

Seit längerer Zeit träumen Science-Fiction-Autoren von mechanischen Maschinen, die dem Menschen im Haushalt bei der schweren mühsamen Arbeit helfen könnten. Ihre gutes Kraftvermögen, große Mobilität und Ausdauer sind besonders wichtig für die Menschen. Seit Anfang des 20. Jahrhunderts nennt man diese Maschinen „Roboter“ – Mechanismen, die eine Arbeit ausführen können. Das ausgeprägtes Bedürfnis nach mechanischen Helfern in verschiedenen Industriegebieten, in der Medizin und im Haushalt weckt weltweit ein großes Interesse in einer Vielzahl von Forschungsgruppen.

Unter praktischen Gesichtspunkten betrachtet, zeigen Roboter eine sehr hohe Effizienz im industriellen Einsatz, dort wo gefährliche, schwere und monotone Aufgaben schnell und mit hoher Genauigkeit übernommen werden können. Monotone Alltagsarbeiten, wie Aufräumen, Putzen und Waschen, werden zunehmend von elektromechanischen Haushaltshelfern geleistet.

Die praktische Implementierung von frei beweglichen bzw. mobilen Robotern erfordert ein leistungsfähiges Steuerungssystem, das die Aufgaben der Bearbeitung visueller und akustischer Informationen, der koordinierten Steuerung mehrerer Gelenke und Körperteile, sowie des reaktiven Verhaltens bewältigt.

Rasante Entwicklungen im Bereich der Rechentechnik begünstigen die Fortschritte in der Robotertechnik. Die daraus entstandenen neuen Verfahren für die Objekterkennung und Objektlokalisierung, für die Algorithmen der räumlichen Bildverarbeitung und der komplexen Bewegungssteuerung bilden eine solide Basis für die Entwicklung von neuen Generationen intelligenter mobiler Roboter. Die Durchführung von anspruchsvollen Pflege- und Serviceaufgaben erfordert eine enge Integration des Roboters in das Umfeld des Menschen. Der Roboter soll dabei nicht nur die ihm aufgetragene Aufgabe lösen, sondern auch seine Umgebung hinreichend wahrnehmen und auf die Anweisungen bzw. Fragen des Menschen sinnvoll reagieren. Die Entwicklung geeigneter reaktiver Steuerungssysteme und die Untersuchung ihrer Eigenschaften sind von großer Bedeutung für die Qualität und für die Akzeptanz der mobilen Roboter.

Mit jeder neuen Robotergeneration lösen die Forscher komplexe mechatronische Aufgaben. Obwohl eine Vielzahl der technischen Ziele, wie z.B. leichte und zuverlässige Konstruktion, hohe Leistungsdichte und intelligente Steuerung in Vordergrund stehen, werden hierbei auch wichtige physiologische Aspekte deutlich. Die Erforschung dynamischer Gangarten von anthropomorphen Robotern dient dem tieferen Verständnis der menschlichen Gehbewegungen. Die bei der Entwicklung von mechatronischen Komponenten für die Roboterarme und -füße erarbeiteten Lösungsmethoden tragen zur Weiterentwicklung von medizinischen Prothesen

bei. Versuche zur künstlicher Sehkraft zeigen Lösungen auf zur Überwindung der Blindheit. Die Erforschung der Kommunikation und Interaktion zwischen Mensch und Roboter verspricht neue Erkenntnisse auf dem Gebiet der Psychologie.

Die Menschen führen komplexe koordinierte Bewegungen, dank der Vielzahl der Muskeln und Gelenke, aus, hierbei wird die Umgebung über visuelle, taktile, thermale u. a. Informationen wahrgenommen. Beispielsweise kommen beim Tastatureintippen, neben dem taktilen Gefühl, bis zu 100 Arm- und Handmuskeln zum Einsatz. Eine stabile Bewegung der Menschen wird durch gemeinsame Verwendung von ca. 400 Skelettmuskeln, dem Vestibularapparat und der Sehkraft ermöglicht. Die emotionale Ereignisreaktion mit einem von 55000 Gesichtsausdrücke erfolgt durch die koordinierte Bewegung von ca. 200 Kopfmuskeln [1].

Nach dem aktuellen Stand der Technik ist es kaum möglich, solch eine extreme Anzahl an Aktoren in einem Roboter einzusetzen. Deswegen wird meistens nur eine für die gestellte Aufgabe ausreichende Aktorenzahl gewählt. Eine durchschnittliche Konstruktion eines anthropomorphen Schreitroboters schließt bis zu fünfzig Aktoren und bis zu hundert Sensoren ein. Das Steuerungssystem eines Roboters soll in der Lage sein, alle Sensoreninformation über unterschiedliche Schnittstellen zu erfassen, eine große Anzahl von Aktoren zu steuern und komplexe Regelungsalgorithmen in der Echtzeit zu berechnen. Im Entwicklungsprozess ist es immer wichtig, einzelne Elemente mathematisch zu analysieren und das gesamte System zu simulieren. Die Entwicklung eines komplexen Automatisierungssystems, das ein Entwicklungsprozess von der mathematischen Analyse über Verifikation eines Simulationsmodells bis hin zur Integration in einen echtzeitfähigen, eingebetteten Hardwaremodul auf effiziente Weise begleitet, ist eine wichtige und aktuelle Aufgabe im Bereich der Automatisierung sowie der Robotertechnik.

1.2 Herausforderungen in der Steuerung von Schreitrobotern

Bereits seit einigen Jahrzehnten werden Industrieroboter erfolgreich für industrielle Automatisierungsaufgaben eingesetzt. Dabei sind die Anforderungen an Roboter Genauigkeit, Gewicht und Leistungsvermögen, die für die Durchführung spezifischen Aufgaben notwendig sind, in der Regel weitestgehend spezifiziert und festgehalten. Dagegen sind die Ansprüche an einen mobilen Roboter oft unklar und widersprüchlich. Es wird oft eine möglichst perfekte Balance zwischen der Leistung/Dynamik und der Mobilität/Gewicht angestrebt. Aus diesem Grund werden von vielen Forschungsgruppen ehrgeizig neue ausgeklügelte Lösungen für mechatronische Konstruktion des mobilen Roboters gesucht. Um höhere Leistungswerte zu erreichen, ist die Bauart des mobilen Roboters dabei an das Anwendungsfeld angepasst. Die Suche nach einem optimalen Verhältnis von Leistung zu Gewicht, bedingt die Neuentwicklungen auf den Gebieten der elektrischen, pneumatischen und hydraulischen Antriebstechnik, die die speziellen Anforderungen der mobilen Roboter erfüllen. Die existierenden industriellen Antriebs- und Steuerungssysteme sind in einem mobilen Roboter wegen des erhöhten Gewichts und der Größe kaum einsetzbar.

Bei den modernen Steuerungssysteme der industriellen Roboter liegt der Aufgabenschwerpunkt in der Bahnplanung und möglichst genauen Bahnverfolgung. Die unterlagerten Antriebsregelungs- bzw. Antriebssteuerungssysteme sind mittlerweile durchgängig standardisiert und bieten eine umfangreiche Funktionalität. So können heute bereits sehr flexibel die Automatisierungsaufgaben je nach Anforderung, sowohl nach dem zentralen (Intelligenz in der Steuerung) als auch nach dem dezentralen (Intelligenz lokal im Antrieb) Ansatz realisiert werden. Die überlagerten Steuerungsebenen, die die Aufgaben der Koordination, der Handha-

bung und der Prozesssteuerung mehrerer Robotersysteme übernehmen, werden meistens mit der SPS Technik realisiert.

Die Steuerungssysteme der mobilen Roboter sind zwar ähnlich aufgebaut wie die der industriellen Roboter, doch ihre Lösungsmöglichkeiten umfassen ein deutlich breiteres Aufgabenspektrum. Die Bahnplanung und Bahnverfolgung wird bei mobilen Robotern um spezielle Verfahren zur Gangplanung bzw. Gangstabilisierung erweitert und stellt aktuell ein der wichtigsten Forschungsschwerpunkte dar. Aber nicht nur das unterschiedliche Laufverhalten und die dynamische Stabilisierung des Roboterkörpers, sondern auch die emotionale Interaktion und plausible Kommunikation mit Menschen sind von großem wissenschaftlichen Interesse und immenser Bedeutung und bieten ein breites Forschungsfeld. Andererseits stellen die unteren Schichten des Steuerungssystems, wie z.B. die Antriebsregelung mit den gut erforschten stabilen Regelungsalgorithmen, sowie die einfache Berechnung der Kinematik für die Roboter-gelenke, zunehmend weniger ein wissenschaftliches Interesse dar. Dennoch verbrauchen die Entwickler der mobilen Roboter erfahrungsgemäß viel Zeit für die aufgabenspezifische Auslegung und Anpassung von zahlreichen Algorithmen der unteren Ebene, wie beispielweise die Sensorsignalerfassung und die Verarbeitung, Antriebsregelung und Kommunikation.

Die moderne Entwicklungsmethodik erlaubt noch vor der hardwarenahen Inbetriebnahme von komplexen Steuerungsalgorithmen die Teile oder sogar das komplette System mit Hilfe der mathematischen Modellierung zu analysieren. Die Ergebnisse der Simulation werden für die Anpassung bzw. die Korrektur der zahlreichen Parameter verwendet. Die Simulation erfolgt in einfacher, benutzerfreundlicher, grafischer Oberfläche, auf der alle Systemteile übersichtlich miteinander verbunden sind und die Modellstruktur klar definiert ist. Erst in den weiteren Phasen der Entwicklung wird anhand dieses mathematischen Modells der Programmcode in einer Programmiersprache für einen eingebetteten Prozessor entworfen. Dieses Verfahren erfordert bei manueller Durchführung eine große Konzentration, ist zeitraubend und fehleranfällig. Deswegen zeichnet sich hier ein deutlicher Trend ab in Richtung der Entwicklung und Verwendung von Werkzeugen für die automatische Codegenerierung. Diese Werkzeuge übernehmen die Aufgabe aus einem entworfenen, simulierten und ausgewerteten mathematischen Modell einen effizienten und optimalen Programmcode automatisch für die Hardware des Zielsystems zu generieren. Die Entwicklungsumgebung soll eine parallele Entwicklung, Modellierung und Fehlerbeseitigung während des Systementwurfs gewährleisten.

Der mobile Roboter nimmt die Welt mit verschiedenen Sensoren wahr. Beim Gehen, für die Kontakterfassung und Berechnung eines Massenmittelpunktes werden Kraftsensoren in den Roboterfüßen eingesetzt. In den Algorithmen der Körperstabilisierung finden Gleichgewichtssensoren, deren Rolle ein elektronischer Gyroskope-Beschleunigungsmesser übernimmt, Anwendung. Den größten Teil der Umgebungsinformation nimmt der Roboter mit Hilfe von Video- und Audiosensoren wahr, ähnlich wie die Menschen. Darüber hinaus werden zahlreiche Sensoren eingesetzt, um die korrekte Funktion der Roboteraktoren zu gewährleisten. Bedingt durch die Ausführungsart, spezifische Messeigenschaften und andere Faktoren, ist jeder Sensor mit einer bestimmten Datenerfassungsschnittstelle (SPI, I²C, USB, Ethernet usw.) ausgerüstet. Die Hardwareplattform des eingebetteten Steuerungssystems soll in der Lage sein, möglichst mehrere standardisierte Schnittstellen zu unterstützen. Die Qualität und Eigenschaften der Algorithmen eines Steuerungssystems sind oft stark von der Dauer der Sensordatenerfassung, sowie der Geschwindigkeit und Synchronität der Datenübertragung zwischen verschiedenen Hardwaremodulen des Systems abhängig. Daher muss das eingebettete Steuerungssystem in der Lage sein, nicht nur mehrere Datenerfassungsschnittstellen bereitzustellen, sondern auch die Algorithmen sehr deterministisch zu berechnen und einen echtzeitfähigen Datenaustausch zwischen Hardwaremodulen zu gewährleisten.

In der langjährigen Robotergeschichte haben die Forscher viele mobile Roboter inklusive verschiedener Aktorarten, Sensorensätze und Steuerungsstrukturen entworfen, damit die unterschiedlichen Bereiche untersucht werden können (z.B. die Natur des menschlichen/tierischen Gehens zu begreifen). Es wurden Roboter gebaut, die direkt oder indirekt auf den Menschen und seine Umgebung wirken können. Dennoch haben erst im letzten Jahrzehnt die Forscher an eine Koexistenz von Roboter und Menschen gedacht und angefangen in dieser Richtung zu forschen.

Es ist offensichtlich, dass ein anthropomorpher mobiler Roboter als ein komplexes mechanisches und elektronisches System nicht nur rein technisch als nichtlineares Steuerungsobjekt der Regelungstechnik betrachtet wird, sondern auch für die Erforschung der zahlreichen soziologischen und psychologischen Aspekte von großem Interesse für die Forscher ist. Während des vergangenen Jahrzehnts ist eine breite Reihe von anthropomorphen und kontaktfreudigen Roboter entstanden, die eine friedliche Koexistenz in menschlicher Umgebung und im Leben anstreben. Aus diesem Grund soll ein mobiler Roboter eine durchdachte Außenansicht und einen Aufbau aufweisen, welche völlig der jeweiligen Roboteraufgabe entsprechen und vom Menschen akzeptiert werden. Der Mensch nimmt ein mechatronisches Wesen anfänglich stets durch seine Außenform wahr. Deswegen ist ein Kopf mit einem kontaktfreudigen Gesicht für die Gestaltung des anthropomorphen Roboters besonders wichtig. Vor allem weil der Erkennungsvorgang des Gegenübers bei dem Menschen beim Gesicht beginnt. Das Robotergesicht beeinflusst dabei die psychologische und emotionale Umgangslaune der Menschen und übernimmt wichtige repräsentative Funktionen. Ausgerüstet mit zusätzlichen Sensoren, wie Videokameras und Mikrofonen, und unterstützt durch intelligente Verarbeitungsalgorithmen, kann der Roboter deutlich mehr Umgebungsinformationen wahrnehmen. Das große Ziel dieser Zusatzfunktionen ist die interaktive Kommunikation zwischen Mensch und Roboter. Nur auf diese Weise kann ein Roboter sinnvoll in eine menschliche Umgebung integriert und vom Menschen akzeptiert werden.

1.3 Stand der Technik

Die Realisierung der interaktiven Kommunikation zwischen einem Mensch und einem Roboter stellt die Robotertechnik vor eine große Anzahl komplexer Aufgaben. Diese Aufgaben reichen von der Ansteuerung der mechanischen Konstruktion bis hin zur Entwicklung der Roboterintelligenz. Wie soll ein Mensch einen Roboter wahrnehmen; welches Kommunikationsvermögen soll ein Roboter für den intuitiven Umgang mit den Menschen und der Umgebung aufweisen; soll das Robotergesicht dem menschlichen Gesicht ähnlich sein oder ist ein vereinfachtes Gesicht angenehmer für die Wahrnehmung; welche Motivation veranlasst einen Roboter zu bestimmten Verhaltensweisen und was kann er von dem menschlichen Charakter beherrschen? Diese Fragen mit emotionalem und sozialem Hintergrund stellen sich während der Entwicklung eines "kontaktfreudigen" Roboters.

Der nachfolgende Abschnitt beschäftigt sich mit der detaillierten Analyse des Standes der Technik auf dem Gebiet der Entwicklung von kontaktfreudigen und mobilen Robotern sowie der Auswertung von Experimenten zu emotionalen und sozialen Aspekten solcher Roboter.

1.3.1 Kontaktfreudige Roboter

Welche Eigenschaften und welche emotionalen und sozialen Fähigkeiten erlauben einem mechatronischen Wesen, dem Roboter, sich einfach und eng in der Kommunikation und dem Arbeits- und Lebensraum des Menschen zu integrieren? Die ersten bedeutenden Ergebnisse wurden vom Begründer der mobilen anthropomorphen Roboter an der japanischen Universität

Waseda veröffentlicht. Darin wurde eine steigende Akzeptanz des Robotergerichtetes je nach Ähnlichkeit zum menschlichen Gesicht untersucht. In seinen Studien hat Masahiro Mori das Phänomen vom sogenannten „Uncanny Valley“ (dt.: „Unheimliches Tal“) [2] beschrieben. Darin sinkt die Akzeptanz ab einer bestimmten Stufe des Anthropomorphismus schlagartig und steigt erst ab einem bestimmten, sehr hohen Niveau wieder an. Bis zu dieser Stufe nehmen die Menschen die Roboter wahr, nach dieser Stufe wird beim Menschen ein negatives Gefühl hervorgerufen.

In den wissenschaftlichen Studien der Forschungsgruppe “Computer Science and Artificial Intelligence Laboratory” von Massachusetts technologischen Institut (MIT) [3] wurden Versuche auf dem Gebiet der interaktiven Kommunikation zwischen Robotern und Menschen durchgeführt. Im COG-Projekt auf der gleichnamigen Roboterplattform (Torso, Arme und Kopf - Bild 1.1.a) wurden die Intelligenztheorien der Psychologie von Leslie und Baron-Cohen [4] untersucht. Entsprechend den Intelligenztheorien wurden im Roboter eine Motivation des Verhaltens, kausale Motivationen und Emotionen (eng: motivations, drives and emotions) implementiert. Besonders große Aufmerksamkeit wurde dabei auf das Augenkontaktsystem gerichtet. Dieses wurde der Entwicklung der visuellen Umgebungswahrnehmung beim Kleinkind (von der Geburt an bis zum 18. Monat) gegenübergestellt. Ein der wichtigsten Ergebnisse dieser Arbeit war die Begründung und der Beweis der besonderen Wichtigkeit des Augenkontaktes während der Interaktion eines Roboters mit dem Menschen.

Die Entwicklung von kontaktfreudigen sozialen Robotern wurde im MIT mit dem Roboter Kismet [5] fortgesetzt. Dieses Robotersystem war in der Lage mit den Menschen nicht nur verbal sondern auch durch einen Gesichtsausdruck und seine Verhaltensweisen emotional zu kommunizieren. Die Fähigkeiten der Gesichtserkennung und des Augenkontaktes wurden um den Stimulus für die Interaktion mit dem Menschen, den Stimulus für das Spiel mit bunten Spielzeugen und eine weitere Gewöhnung erweitert. Durch den Einsatz eines verbalen Systems für Schallzeichen- und Spracheerkennung und Spracherzeugung (die meisten Schallarten sind einfachste Wörter bzw. Kleinkindmurmeln) wurden folgende zwei Aufgaben gelöst: die Erkennung der Aufmerksamkeit einer Person und die Bestimmung des Launeniveaus aus der Gesprächspartnerstimme. Das Äußere des Roboters Kismet (bestehend aus Kopf und Hals) erinnert an einen kleinen Haushund mit langen Schlappohren und einem breit wirkendem Gesicht (Bild 1.1.b). Hier hat Kismet viele Gesichtsteile, die auch für das menschliche Gesicht charakteristisch sind. Diese Merkmale verstärken die positiven emotionalen und psychologischen Wahrnehmungsfaktoren des Roboters auf die Menschen. Im Projekt Kismet ist letztendlich zusammengefasst, dass die Stimuli in der Interaktion zwischen einem emotionalen Roboter und einem Menschen eine bedeutende Rolle spielen.

Im nachfolgend vorgestellten Projekt wird im MIT ein sozialer Roboter namens Merz [6] entworfen. Der Roboterkopf hat eine vereinfachte Gesichtskonstruktion und Außenansicht, Gesichtsteile sowie die Kinematik des Roboters sind dem menschlichen Gesicht nachempfunden (Bild 1.2.a). Zusätzlich zu den Interaktions- und Spielstimuli, ist das Steuerungssystem

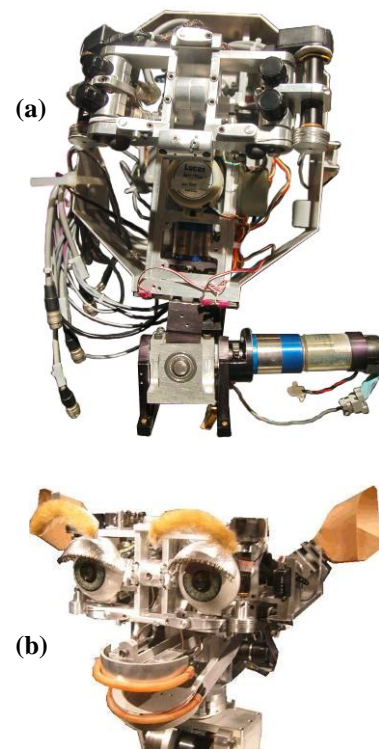


Bild 1.1 – MIT-Roboters:
COG (a) und Kismet (b)

um die Wiedererkennung eines Menschen erweitert worden. Das Erkennungssystem speichert eine Videoreihe mit dem Gesicht eines Gesprächspartners, segmentiert diese Information und erfasst Merkmale, die mit der akustischen Information (Gesprächspartnerstimme) zusammengesetzt werden. Diese visuell-akustische Information wird für jeden Gesprächsteilnehmer gesammelt und in einer Datenbank gespeichert. Diese gesammelte Information ist das Roboter Gedächtnis, in dem alle früher getroffenen Personen gespeichert sind. In der Merz-Forschung wird beschrieben, dass das Lernen während einer Interaktion ein unabdingbarer Teil eines sozialen Wesens ist. Dieses Erkenntnis wurde durch eine Versuchsreihe in [6] nachgewiesen, in welcher die Roboter-Interaktion für manche Personen sehr einer menschlichen Interaktion bzw. einem Gespräch ähnelte.

Die Weiterentwicklung des kontaktfreudigen Roboters für die Mensch-Roboter-Interaktion wurde mit dem Roboter Leonardo [7] fortgesetzt. Der Roboter ist ein vollständiges mechatronisches Wesen mit allen Körperteilen (Arme, Beine, Rumpf und Kopf - Bild 1.2.b) und hat insgesamt 65 Freiheitsgrade (DOF). Solche Konstruktionsvollständigkeit erlaubt dem Roboter mit dem Menschen nicht nur mit Hilfe von den Gesichtsemotionen, sondern auch durch die Körper- und Gliedmaßenbewegungen, zu kommunizieren. Der Forschungsschwerpunkt des Roboters Leonardo liegt in der Fähigkeit eines künstlichen Wesens dem Lehrer bzw. Gesprächspartner sinnvolle Emotionen sowie deutliche Handgesten und Körperbewegungen zu zeigen. Diese zusätzliche Information erleichtert den Interaktionsprozess für den Menschen.

Die Untersuchungen verschiedener Lernmethoden für interaktive Roboter ergeben ein breites Forschungsfeld. An der technischen Universität Tokio wurde ein Roboterkopf [8] (vollständiges Frauengesicht) gebaut, der während der Interaktion mit einer Person einen eigenen Charakter und eine eigene Persönlichkeit gewinnen kann. Hier wurde die Reinforcement Learning Methode für das Erlernen der gebührenden Reaktionen auf menschliches Verhalten durch einen Roboter eingesetzt. In der Arbeit [9] ist zusammengefasst, dass nicht das Äußere des Roboterkopfes sondern die Beweglichkeit der Gesichtspartien während eines Dialogs von entscheidender Bedeutung ist. Die Constructive Induction Methode [10] wird für das Erlernen von Verhaltensweisen, Mimik und Dialogführungsmethoden angewendet.

Im Lernprozess soll der Lehrer den Zustand der lernenden Person bzw. des Roboters verstehen. Der Lernende soll schnell und präzise auf die Anweisungen reagieren und eindeutige, verständliche Rückmeldungen geben. Diese Rückkopplung begründet somit die Wichtigkeit des Ausdrucks von Emotionen beim interaktiven Roboter, da sie dem Lehrer den Grad der Stoffbeherrschung und des -verständnisses übermitteln. Zu diesen Ergebnissen kamen die Forscher der Technischen Fakultät an der Universität Bielefeld [11].

Für die qualitative Erlernung eines Themas oder Objektes soll der Lehrende emotional fähig und stimuliert sein und seine Aufmerksamkeit muss auf dieses Thema konzentriert werden. Eine Aufmerksamkeitsfokussierung im Dialog ist ein Teil der Umgebungswahrnehmung und trägt zur Verhaltensdynamik eines Lernprozesses bei. Bei Interessenverlust, der Aufmerk-

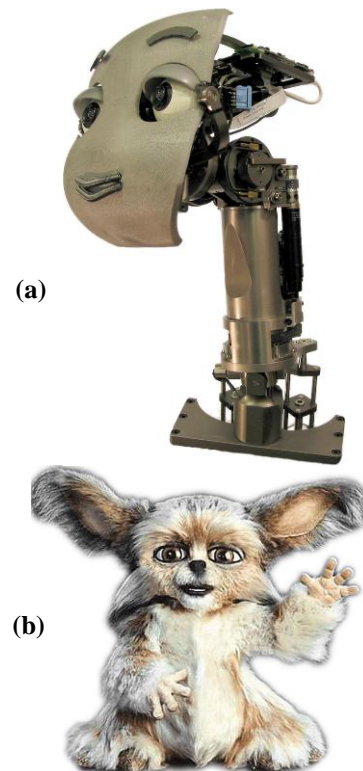


Bild 1.2 – MIT-Roboters:
Merz (a) und Leonardo (b)

samkeits- und Konzentrationsabnahme eines Lernenden bemerkt der Lehrer kleine Verhaltensänderungen (z.B. Kopfdrehungen zur Seite, falsche Blickrichtung usw.). Solche Besonderheiten wurden an der Technischen Fakultät der Universität Bielefeld an der Roboterplattform BIRON [12] erforscht. Ein bedeutendes Ergebnis dieser Forschung ist die Nachweisbarkeit der vom Roboter bekundeten Aufmerksamkeit während eines Dialogs.

Der mobile anthropomorphe Roboter ROTTO (Bild 2.1) wurde nach dem Vorbild eines 12-jährigen Kindes entwickelt und ist 1,3m groß. Der Roboter wurde als eine Plattform zur Erforschung der sogenannten ballistischen und dynamischen Schreitalgorithmien entwickelt. In den Zwischenphasen der Roboterentwicklung wurde der Roboter ROTTO mehrmals auf Messen und Universitätsveranstaltungen (Hannover Messe, Lange Nacht der Wissenschaft, Tag der offenen Universitätstür usw.) ausgestellt, als er weder einen Kopf noch Arme in der Konstruktion des Oberkörpers besaß [48]. Dieser unvollständige Aufbau des Roboters hatte zur Folge, dass die Besucher kein großes Interesse an dem Roboter zeigten. Daraus ist ersichtlich, wie wichtig eine vollständige Roboterkonstruktion mit Kopf (und einem emotionalem Gesicht) und Armen für die psychologische Akzeptanz ist.

Durch die steigende mechanische bzw. steuerungstechnische Komplexität der mobilen Roboter werden zunehmend neue Anforderungen an die kommunikativen Fähigkeiten eines Roboters gestellt: er soll den Menschen nicht nur akustisch sondern auch durch die Gestik verstehen und sein Verhalten, Motivation und Emotionen klar zeigen. Solche Fähigkeiten sind von großer Bedeutung für die effiziente Interaktion eines Roboters mit den Menschen. Daraus lassen sich zusammenfassend folgende Eigenschaften des sozial-emozionalen Roboters ableiten:

- Das äußere Erscheinungsbild des Roboters, sein ganzer Habitus und das entsprechend der zu bearbeitenden Aufgabe gewähltes Gesicht legen fest, ob dem Roboter innerhalb der menschlichen Gesellschaft mit Akzeptanz oder Antipathie begegnet wird.
- Für den erfolgreichen Dialog ist der visuelle Augenkontakt mit dem Gesprächspartner wichtig.
- Der Einsatz der Stimuli für eine Interaktion, ein Spiel und Lernen bedingt das voraus erwartete und unbekanntes Roboterverhalten. Auf diese Weise wird dieses Verhalten nicht als vorprogrammiert (welches eine Abneigung erregen kann), sondern als akzeptabel und natürlich wahrgenommen.
- Die Lernfähigkeit eines Roboters. Sie erlaubt dem Roboter neue Fähigkeiten zu erlernen und hierdurch den eigenen Charakter und eigenes Verhalten an den aktuellen menschlichen Kreis anzupassen.
- Förderung von Aufmerksamkeit und Konzentration bei Dialogen und Lernprozessen.

1.3.2 Anthropomorphe Roboter

Bei der Wahl zur Rolle der helfenden Hand des Menschen steht der anthropomorphe Roboter an erster Stelle. Die sind auf Art und Weise der Menschen konstruiert und verfügen über die Flexibilität, Manövrierfähigkeit und Beweglichkeit des menschlichen Körpers. Im Vergleich zu menschlichen Aktor- und Sensorsystemen sind die des Roboters weniger komplex. Der Mensch kann Temperatur, Berührung und das Kratzen mit jedem Hautteil fühlen und erkennen, aber ein Roboter kann diese nur mit den dazu definierten Sensorflächen. Andererseits kommen besondere Sensoren zum Einsatz, die für Menschen unspezifisch sind. Ein Radar und ein Laserabstandsmesser ermöglichen dem Roboter die genaue Bestimmung seiner Positi-

on und die mehrdimensionalen Fußkraftsensoren erlauben die Berechnung des Massenmittelpunktes und Null-Momentpunktes. Es ist kein Sonderfall einen Roboter mit mehreren (drei oder vier) Videokameras für detaillierte die Umgebungswahrnehmung auszurüsten. Die große Muskelanzahl und ihre außergewöhnliche Eigenschaften sind in den anthropomorphen Roboter durch eine begrenzte Anzahl von Aktoren ersetzt. Die muskelartigen Antriebeigenschaften versuchen die Entwickler durch eine besondere mechanische Aufbauweise und komplizierte Steuerungen zu erreichen. Darüber hinaus stehen energieoptimale Steuerungen und der Vergleich des Energiebedarfs des Menschen mit dem des Roboters im Mittelpunkt des Interesses. Im Folgenden werden die anthropomorphen Roboter seitens der Sensorik/Aktorik, der Steuerung und der menschlichen Wahrnehmung bzw. Akzeptanz analysiert.

Der japanische Roboter ASIMO wurde hauptsächlich als Helfer des Menschen entworfen und wurde mit verschiedenen Aufgaben betraut (z.B. in [13] als Empfangsdame). Die gewünschte Beweglichkeit des Roboters wird durch den Einsatz der 34 Servomotoren [14] mit dem „*Harmonic Drive*“ Getriebe [15] realisiert. Jede Motorachse ist mit einem Positionssensor für die Motorregelung und Berechnung des Roboterzustands ausgerüstet. In der ZMP-gesteuerten Lokomotion von ASIMO werden die Informationen von sechsdimensionalen Kraftsensoren (in jedem Fuß einer), von dem im Körper montierten Gyroskope und dem Beschleunigungssensor ausgewertet. Die Orientierung in der Umgebung verläuft über das Stereokamerasystem und über die Mikrophone, mit denen der Roboterkopf ausgerüstet ist

Das Steuerungssystem ist auf die Ethernet-verbundene Reihe von PC's verteilt, wo jeder Rechner eine eigene Aufgabe (Vision, Töne, Regelung) berechnet. Per WLAN ist der ASIMO mit dem Server verbunden, auf dem die Umgebungsdaten für die lokale Orientierung und die Gesichtsdatenbank für die Menschenwiedererkennung gespeichert sind. Auf dem Server wird ein Dialog-Manager, der den Roboter in Interaktionen steuert, ausgeführt. Der Server ist mit dem Internet verbunden, durch das ASIMO auf die gewünschten Datenbanken zugreifen kann und andererseits kann des Fachpersonal den Roboter beobachten und steuern.

Der ASIMO Roboter hat kein spezifisches Gesicht, es hat eine ikonische Gestalt und ist zu keiner emotionalen Interaktion fähig. Dessen ungeachtet hat ASIMO eine große Akzeptanz dank seiner vollständigen, an den Menschen angelehnten, anthropomorphen Konstruktion, den ausführlichen Gesten und dem Verhalten.

Der Roboter KOBIAN [16] ist fähig die Emotionen nicht nur mit dem Gesicht sondern auch mit dem ganzen Körper und Körperbewegungen auszudrücken. Der Körper wurde vom Roboter WABIAN-2 [17] übernommen. Beide Roboter sind für die Simulation der menschlichen Lokomotion und für medizinische Einsätze in der physikalischen Therapie konstruiert.

Für eine hohe Roboterbeweglichkeit ist der Körper mit 41 Antrieben ausgerüstet. Überdies ist der Kopf bzw. das Gesicht mit sieben Antrieben für emotionales Verhalten gestaltet. Die Körpergelenke werden mit DC-Motoren durch „*Harmonic Drive*“ oder Zahnriemen-Getriebe angetrieben. Der dem ASIMO ähnliche Sensorensatz ist mit zusätzlichen Kraftsensoren versehen. Jeder Arm ist mit einem sechsdimensionalen Kraftsensor für Stoß- und Zugaufgaben und jede Hand mit 14 „*FlexiForce*“ Punkt-Kraftsensoren [18] für Greifaufgaben ausgerüstet.

Das Steuerungssystem wird auf dem „*Pentium M*“ PC unter dem Echtzeitbetriebssystem *QNX* ausgeführt. Die Peripherie (Motorensteuerung, Sensoren, Kommunikation usw.) ist per PCI Schnittstelle am PC angeschlossen. Eine externe Station kann per WLAN auf dem Roboter-PC für die Steuerungs- und Beobachtungszwecke zugreifen.

Neben dem vollständigen Körper hat KOBIAN ein spezifisch gestaltetes Gesicht für die Emotionsdarstellung. In Verbindung mit dem Stereokamerasystem, das aus den zwei CMOS-USB-Videokameras (in jedem Auge) besteht, stellt KOBIAN ein vollständiges mechatronisches System dar, das eine gute Bewertung und hohe Akzeptanz in der emotionalen Interaktion erreicht.

In einem Gemeinschaftsprojekt von amerikanischen und japanischen Forschern wurde der Roboter DB [19] entworfen. Diese Arbeiten sind auf folgenden Aufgaben fokussiert: der Aufbau der Bewegungsbahn, das Roboterlernen von einer Demonstration, okulomotorische Steuerung und interaktives Verhalten [20].

Die Besonderheit dieses Roboters liegt im Einsatz von hydraulischen Aktoren: insgesamt 30 Hydroantrieben sind für die Roboterbewegungen zuständig. Der Kopf ist mit dem komplexen Stereokamerasystem ausgerüstet. Es besteht aus vier Kameras (zwei pro Auge). Eine Kamera hat einen weiten Schwenkwinkel und die andere ein schmales Sichtfeld (Fovea). Das gesamte Steuerungssystem ist auf sechs vernetzten „PowerPC“ Rechner unter dem Echtzeitbetriebssystem *VxWorks* realisiert.

Der Roboter DB zeichnet sich durch sein Verhalten, das er vom Menschen lernt, aus. Dank seiner Lernfähigkeit kann er in die menschliche Umgebung einfach integriert werden.

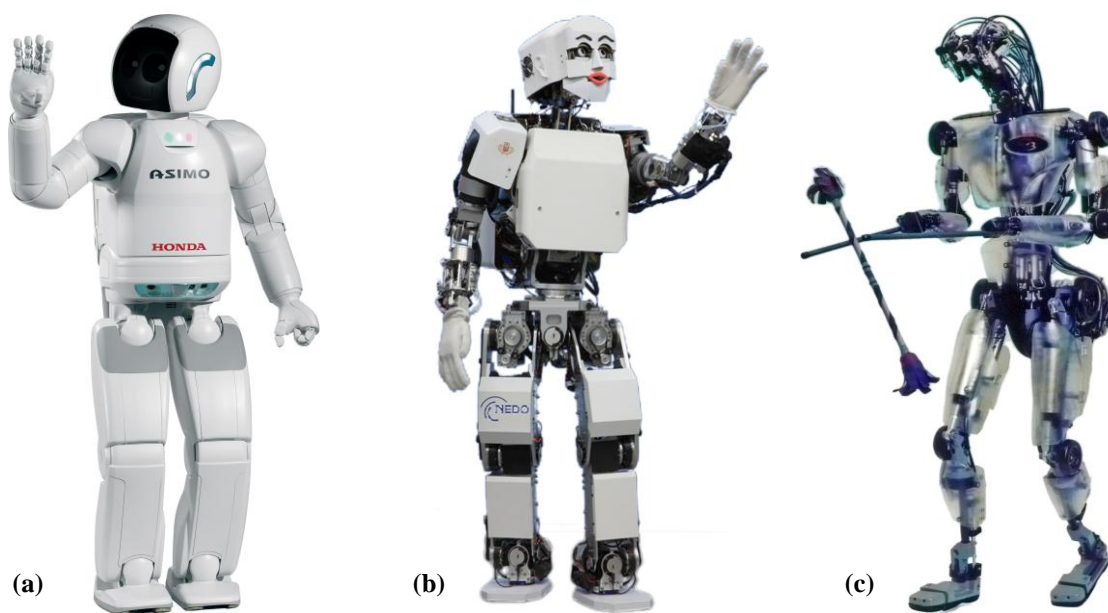


Bild 1.3 – ASIMO (a), KOBIAN (b) und DB (c)

Durch die Analyse der drei diskutierten anthropomorphen Roboter lassen sich die wichtigsten Eigenschaften humanoider Roboter zusammenzufassen:

- große Anzahl von Aktoren und Sensoren;
- äußerst komplexes Steuerungssystem, das auf mehreren vernetzten Rechnern mit einem Echtzeitbetriebssystem läuft;

- sicherer Datenaustausch zwischen den Aktoren, Sensoren und der Steuerung für einen ganzheitlichen Informationsfluss;
- für eine gute Bewertung und hohe Akzeptanz in der menschlichen Umgebung sind für einen Roboter ein vollständig anthropomorpher Aufbau sowie menschliches, emotionales Verhalten und eine gute Lernfähigkeit von großer Bedeutung.

1.3.3 Software- und Hardwarearchitektur in der Robotik

Die Entwicklung eines mechanisches Wesens, das zur Kommunikation die für den Menschen intuitive Sprache der Gesten und Emotionen benutzt, setzt den Einsatz von leistungsfähigen, exakt strukturierten Steuerungssystemen voraus. Für die Erkennung von Objekten in der Umgebung, von Gesichtern und Gesichtsausdrücken sind Umgebungssensoren von großer Bedeutung. Der Einsatz von Videokameras, Mikrofonen, Ultraschallradaren und anderer Umgebungssensoren erfordert komplexe Algorithmen zur Bild- und Signalbearbeitung, die teilweise sehr rechenintensiv sind. Für den genauen und qualitativen Ausdruck verschiedener Emotionen, Gesten und Verhaltensmustern muss ein sozialer mobiler Roboter über zahlreiche aktive Freiheitsgrade bzw. Gelenke verfügen. Darüber hinaus ist eine kompakte eingebettete Steuerungshardware von Vorteil, damit die Mobilität des Roboters nicht beeinträchtigt wird. Unter Berücksichtigung solcher Aspekte (Steuerungsstruktur, Sensoren, Aktoren, Hardware und Software) werden in diesem Unterkapitel verschiedene Entwicklungen betrachtet.

1.3.3.1 Kommunikative Roboter

Der soziale Roboter COG [21], der aus dem Kopf, Torso und den Armen besteht, wurde im MIT entworfen. Seine hohe Beweglichkeit wurde durch die Ansteuerung von insgesamt 22 Elektromotoren erreicht [22]. Insgesamt vier Videokameras dienen der visuellen Wahrnehmung der Roboterumgebung, wobei zwei Kameras mit einem Weitwinkelobjektiv unbeweglich in der Gesichtsmitte eingebaut sind. Zwei andere Kameras mit schmalen Sichtfeld (Fovea) sind in die beweglichen Augäpfel integriert. Für die Funktion des Gleichgewichtsapparats ist ein Gyroskop vorgesehen. Die Eingabe- bzw. die Ausgabe der Audio- und Videosignalen erfolgt über eine digitale PCI/ISA Schnittstelle. Das Multitasking-System der Robotersteuerung besteht aus dem Mikrokontrollernetzwerk und vier industriellen PCs, die unter dem QNX Betriebssystem laufen und mit Hilfe eines Ethernet Netzwerks verbunden sind. Das Multitasking-System umfasst die Ebenen der Signalbearbeitung, der Verhaltens- und Emotionsteuerung und wird mittels „multithreading subset of Common Lisp“ realisiert. Die Ebene der Roboter motorik (Motorensteuerung) besteht aus einem Motorola-Mikrokontrollernetzwerk und einer speziell für den COG entwickelten Elektronik.

Der soziale Roboter Kismet [5] von MIT hat emotionale Fähigkeiten für die Interaktion. Die Roboterkonstruktion besteht nur aus dem Hals und dem Kopf mit vielen aktiven Gesichtsteilen (Lippen, Augenbrauen, Ohren), die von insgesamt 19 Elektromotoren [23] angetrieben werden. Der Aufbau des Videosystems ist dem Roboter COG ähnlich. Das Multitasking-System für die höheren Steuerungsebenen wird in „multithreading subset of Common Lisp“ und verteilte Berechnungen mit Hilfe von CORBA (eine Spezifikation [24] für verteilte Anwendungen) implementiert. Die Bildverarbeitung, wie z.B. die Gesichtserkennung und Berechnung von Algorithmen der Verhaltenssteuerung, erfolgt verteilt auf neun industriellen PCs mit QNX Betriebssystem. Für die Aufgaben der Spracherkennung und Spracherzeugung wird ein PC mit Windows NT eingesetzt. Alle Rechner sind mit Hilfe eines Ethernet Netzwerks miteinander verbunden. Die Ebene der Roboter motorik von Kismet ist dem Roboter COG nachempfunden.

Zur Erforschung des ungesteuerten Lernens wurde im MIT der kontaktfreudige Roboter Merz [6] entwickelt, der nicht nur emotionale Fähigkeiten für die Interaktion hat, sondern mit einem System für die Menschenwiedererkennung ausgestattet wurde. Der Roboter (die gesamte Roboterkonstruktion) besteht aus dem Hals und dem Kopf, der ein stark vereinfachtes Gesicht hat. Der Roboterkopf ist mit 11 Servoantrieben für die Gesichtsmotorik und zwei Antrieben mit serieller Elastizität für die Halsmotorik ausgerüstet. Zwei kleine „Dragonfly“ Videokameras von „Point Grey“ mit FireWire Schnittstelle [25] wurden in die beweglichen Augäpfel integriert. Diese dienen der Erfassung der visuellen Umgebungsinformationen. Höhere Ebenen des Steuerungssystems sind mit Hilfe von einem Rechnernetzwerk aus sechs PCs realisiert, wobei jeder Rechner die Berechnung je einer bestimmten Aufgabe übernimmt. Das Multitasking-System besteht aus L/MARS -Softwarekomponenten [26], in welcher L eine allgemeine Lisp-basierte Programmiersprache ist. MARS ist eine spezifische Programmiersprache, die in L eingebettet ist und zur Programmierung paralleler Berechnungen eingesetzt wird. Zur Steuerung der Roboterantriebe wurde eine spezielle Steuerungselektronik auf DSP-Basis entwickelt.

In zahlreichen Studien wurden die emotionalen Aspekte in der Mensch-Roboter-Interaktion auch mit Hilfe von vereinfachten mechanischen und elektronischen Robotersystemen untersucht. Auf diese Weise ist in der Universität von Hertfordshire (England) ein Roboter KASPAR [27] zur Untersuchung der emotionalen Kommunikationsaspekte entstanden. Die Roboterkonstruktion besteht aus dem Torso, den Armen und dem Kopf, der das Gesicht mit einer menschenähnlichen Außenansicht aufweist. Die Projektaufgabe war die Entwicklung einer Mensch-Maschine Schnittstelle, die durch die Verwendung von Mimik und Emotionen, die Kommunikation mit dem Menschen intuitiv und natürlich gestaltet. Die Videoinformation wurde mit Hilfe von „ultra-miniatur Videokameras“ erfasst.

In der technischen Universität Tokyo (Japan) wurde ein Roboter entwickelt, der einen eigenen Charakter während eines Lernprozesses entwickeln kann [8]. In die Augen des Roboters sind zwei kompakte CCD-Kameras integriert. Zur Ansteuerung der einzelnen Freiheitsgrade des Roboterkopfes wurden 18 kleine pneumatische Aktoren unmittelbar in die Kopfkonstruktion eingebaut.

In [9] ist ein Roboter für die Untersuchung des Lernverhaltens und der emotionalen Interaktion entworfen. Zur Ansteuerung der Gesichtsausdrücke wurde in der vereinfachten Kopfkonstruktion RC-Servoantriebe von Hitec eingesetzt. Die Antriebssteuerung basiert auf den günstigen Mikrokontrollern „Basic Stamp“, die mit einer PC-Steuerung verbunden sind. USB-Videokameras, Sonare und Lichtsensoren wurden als Umgebungssensoren genutzt. Alle Steuerungsalgorithmen wurden in folgende Programmiersprachen entworfen: Visual C++, Visual Basic, Lisp und Prolog.

In der Universität von Pisa (Italien) ist im Projekt FACE (Facial Automaton for Conveying Emotions) ein vollständiger anthropomorpher biometrischer Roboterkopf entstanden [28]. Großer Wert wurde auf eine detaillierte Konstruktion des Gesichts gelegt, um die realitätsnahe Abbildung zahlreicher Emotionen zu erreichen. In der Gesichtsmotorik des Roboters sind dielektrische Elastomeraktoren eingesetzt.

1.3.3.2 Schreitroboter der Universität Magdeburg

Die Robotertechnik in der Universität Magdeburg hat eine vieljährige Geschichte in der Entwicklung von mobilen Schreitrobotern [29]. In den vergangenen Jahren wurde eine Serie aus

vier sechsbeinigen Schreitrobotern (KATHARINA, SLAIR, SLAIR2 und ANTON) entwickelt, um verschiedene Teilsaspekte des Roboterentwurfs zu beleuchten. Dabei ist ein Konzept zum Aufbau eines mehrstufigen Steuerungssystems entstanden, das das Kernsystem des zweibeinigen mobilen anthropomorphen Roboters ROTTO bildet [48].

Der mobile Roboter KATHARINA (Bild 1.4.a) [30] hat sechs Beine, jedes verfügt über drei aktive Freiheitsgrade. Alle 18 Freiheitsgrade des Roboters werden durch Gleichstrommotoren angetrieben. In jedem Fuß ist ein Drei-Komponenten Kraftsensor eingebaut, für die Erfassung der Kraftinformation im Kontaktpunkt. Der Roboter ist mit zwei Videokameras für eine Erkennung der Umgebung ausgerüstet. Das gesamte Robotersteuerungssystem wird in einem eingebetteten DSP Prozessor berechnet, sodass der Roboter autonom die Bewegungsabläufe anhand von einfachen Bewegungsbefehlen berechnen und ausführen kann. Die Befehle zur Bewegungsrichtung und Geschwindigkeit werden von einem Joystick über eine serielle Schnittstelle zum Roboter übertragen. Der Roboter KATHARINA wurde primär als eine flexible Plattform zur Untersuchung verschiedener Kontaktaufgaben, wie z.B. adaptive Nachgiebigkeit, aktive Verteilung der Reaktionskräfte, automatische Anpassung an eine unregelmäßige Oberfläche, und zahlreiche Serviceaufgaben [31], verwendet. Für die Echtzeitsimulation des Roboterhaltens, die Benutzerinteraktion und Ergebnisvisualisierung ist ein interaktiver visueller Simulator [32] entwickelt worden.

Der sechsbeinige Roboter SLAIR (Bild 1.4.b) [33] stellt eine Weiterentwicklung des Roboters KATHARINA dar, wobei der Schwerpunkt auf einer modularen und günstigeren Konstruktion gelegt wurde, die aus drei identischen Körpermodulen besteht. Dank der zusätzlichen aktiven Gelenke zwischen den einzelnen Körpermodulen, weist der Roboter eine hohe Beweglichkeit auf. Der Roboter SLAIR wird mit Hilfe von insgesamt 20 lagegesteuerten Servomotoren angetrieben. Drei-Komponenten Kraftsensoren in jedem Roboterfuß, Zwei-Achs Gyroskope und Sensor der Nahnavigation dienen der Erfassung von Umgebungsinformationen. Das Robotersteuerungssystem wurde in drei Ebenen strukturiert: "Action", "Primitive" und "Servo" Ebene. Die beiden oberen Steuerungsebenen, sogenannte "Action" und "Primitive", sind mit der Steuerung des Roboterhaltens beschäftigt und sind auf einen externen Steuerungsrechner verlagert. Die Antriebssteuerungsebene „Servo“ wird auf den lokal im Roboterkörper integrierten DSP-Prozessoren ausgeführt und umfasst die Geschwindigkeits- und Lageregelung sowie Überwachung aller Roboterantriebe. Die Datenübertragung zwischen dem Roboter und dem Steuerungsrechner erfolgt über eine serielle RS232-Schnittstelle. Mit SLAIR wurden verschiedene Gangarten und die Algorithmen der kraftgesteuerten Anpassung an unregelmäßige Oberflächen untersucht.

Der sechsbeinige mobile Roboter SLAIR2 (Bild 1.4.c) [33] wurde als Nachfolger von SLAIR ebenfalls modular gebaut, wobei der Schwerpunkt auf die Optimierung der Leistungsfähigkeit des Roboters gelegt wurde. Die mechanische Konstruktion des Roboters ist angelehnt an eine Leichtbauart. Einzelne Robotergelenke wurden neu strukturiert und eine bessere Kraftausnutzung durch den Einsatz von Differentialgelenk-Mechanismen erreicht. Die eingesetzten Sensoren und das Steuerungssystem sind dem SLAIR ähnlich, für die Datenübertragung zum externen Steuerungsrechner wurde jedoch eine fehlertolerante serielle RS485-Schnittstelle eingesetzt [35]. Diese Schnittstelle erlaubt es, die Berechnungsabstastzeit im Robotersystem auf unter 10ms zu reduzieren, was für die Regelungsqualität von Kontaktaufgaben von großer Bedeutung ist. Das gesamte Robotersteuerungssystem wurde in MATLAB/Simulink [36] entworfen und wird in Echtzeit auf einem Rechner unter dem Betriebssystem Windows XP [37] ausgeführt. Der Roboter SLAIR2 wurde bei der Untersuchung von komplexen Gangarten und der Entwicklung von Algorithmen zur Anpassung an unbekannte Oberflächen sowie Hindernisüberwindungen und Kletteraufgaben erfolgreich eingesetzt.

Bei der Entwicklung des sechsbeinigen mobilen Roboters ANTON (Bild 1.4.d) wurde ein neues Konzept zur Auslegung des Steuerungssystems gewählt, welches die beschleunigte Entwicklung von Steuerungs- und Regelalgorithmen eines komplexen Robotersystems erlaubt. Der Roboter ANTON verfügt über 24 Freiheitsgrade, die mit Gleichstrommotoren angetrieben werden [38]. Die integrierte Robotersensorik ähnelt der im Roboter SLAIR2 eingesetzten Komponenten und wurde mit zwei Videokameras im Roboterkopf vervollständigt. Die Besonderheit des Steuerungskonzeptes von ANTON liegt darin, dass alle drei Ebenen des Steuerungssystems an einem externen Steuerungsrechner berechnet werden, der unter einem Echtzeitbetriebssystem xPC [39] läuft. Die Datenübertragung zwischen dem Roboter und dem Steuerungsrechner erfolgt über ein 100MBit-Ethernet Netzwerk mit Hilfe eines echtzeitfähigen industriellen Protokolls EtherCAT [40]. Auf diese Weise wurde eine Berechnungsabtastzeit von unter 1ms erreicht. Die Steuerungselektronik in jedem Robotermodul besteht dabei jeweils aus einem Kommunikationsprozessor NetX50 von Hilscher [41] für die echtzeitfähige Ethernet-Kommunikation und einer Xilinx FPGA [42] für die Datenerfassung und Generierung der PWM Signale für die Leistungstreiber der Antriebe. Der Roboter ANTON diente der Entwicklung von Algorithmen für die automatische Anpassung an unbekannte und komplexe Oberflächen, der Evaluierung verschiedenartige Gangarten sowie der Untersuchung unbekannter Umgebungen [43].

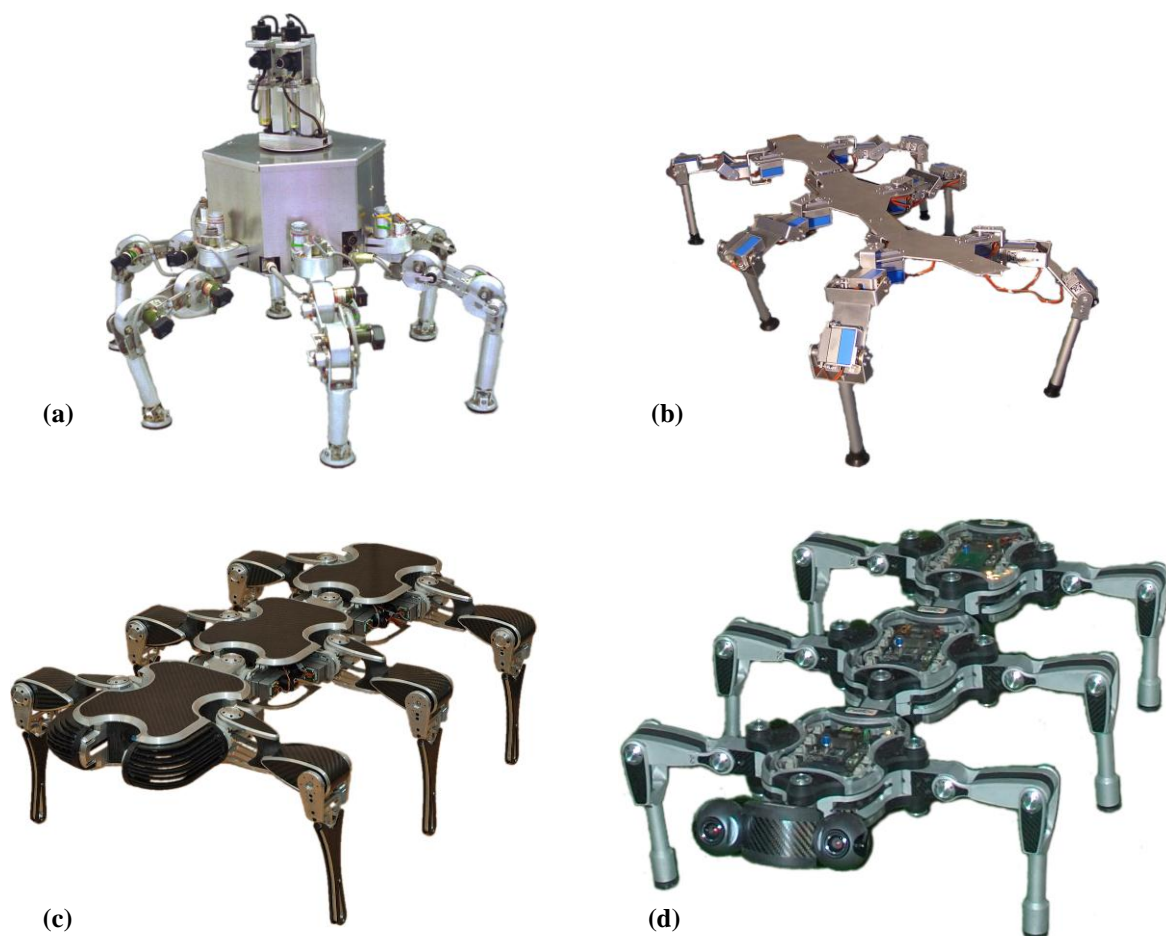


Bild 1.4 – KATARINA (a), SLAIR (b), SLAIR2 (c) und ANTON (d)

Die jüngste Entwicklung der Robotlab Gruppe [29] ist ein anthropomorpher Roboter ROTTO. Der Roboter ist modular aufgebaut. Jedes mechatronische Antriebsmodul ist als ein Linearantrieb realisiert und beinhaltet neben dem Motor auch die komplette Steuerungselektronik [44, 45, 46]. Das Modul ist mit Hilfe eines EtherCAT Netzwerks an ein übergelagertes Steuerungssystem angebunden, führt aber dennoch - zusätzlich zur Geschwindigkeits- und Positionsregelung des Synchronmotors - auch die Aufgaben der Datenerfassung von Peripheriesensoren (Kraftsensoren, Gyroskopen usw.) aus. Kleine Automatisierungsaufgaben, wie z.B. die Initialisierung und Arbeitsraumüberwachung, sowie sichere Abschaltung des Antriebes bei einer Störung werden ebenfalls im Modul autark abgewickelt [47]. Insgesamt sind 17 Module in die Roboterkonstruktion integriert und kommunizieren mit einem zentralen Steuerungsrechner über ein Ethernet-Netzwerk. Der zentrale Steuerungsrechner übernimmt die Berechnung sowohl der höheren Ebenen der Robotersteuerung (Ganggenerierung und Koordination, Erhaltung der Stabilität des Roboters uvm.) als auch der mittleren Ebenen der Antriebssteuerung (Positions-, Kraftregelung) berechnet. Auf dem Roboter ROTTO werden ballistisches Laufverhalten, Erhaltung der Roboterstabilität während der Bewegung [48], dynamisches Laufen mit Hilfe von positionsgeregelten [49, 50] sowie mit Hilfe von kraftgesteuerten Antrieben mit serieller Elastizität [51] erforscht.

Die hier aufgeführten Forschungsarbeiten zeigen deutlich, dass zahlreiche Forschungsgruppen die Interaktion zwischen dem Menschen und dem Roboter in den Mittelpunkt der Untersuchungen stellen. Eine effiziente Integration des Roboters ins menschliche Umfeld bedingt die Entwicklung hochkomplexer mechatronischer Konstruktionen, die eine große Anzahl von Sensoren, Aktoren und Steuerungselektronik in einem begrenzten Raum zusammenführen. Die Mobilität des Roboters hängt im besonderen Maße sowohl von der Effizienz der eingesetzten Energieversorgung als auch von der Steuerungs- und Regelungstechnik ab und stellt in einigen Fällen eine schwer erreichbare Aufgabe dar. Aufgrund der speziellen Größen-, Gewichts-, und Leistungsanforderungen an alle Roboterkomponenten müssen die Entwickler oft auf fertige industrielle Lösungen verzichten und eine eigene optimale Lösung suchen. Der Fortschritt in der Leistungselektronik, Rechentechnik, Steuerungsalgorithmen, Leichtbau bietet dabei neue Ansätze auf dem Weg zu modularen, hocheffizienten und leistungsfähigen Robotersystemen. Auf diesem Weg sind noch zahlreiche komplexe Aufgaben im Aufbau eines Robotersteuerungssystems bzw. Automatisierungssystems zu lösen, die insbesondere folgende kritische und wichtige Eigenschaften berücksichtigen:

- hoch optimierte Antriebe in Hinblick auf das Leistung-Gewichts-Verhältnis
- optimierte und integrierte Leistungselektronik mit intelligenter multifunktionaler Ansteuerung
- hohe Rechenleistung des eingebetteten Steuerungssystems. Effiziente und automatisierte Verteilung der Rechenaufgaben zwischen den Steuerungsmodulen.
- Echtzeitfähigkeit in den Berechnungen und in der Datenübertragung
- Einfache und intuitive Werkzeuge für die schnelle Entwicklung des Steuerungssystems eines Roboters

1.4 Ziel und Aufbau der Arbeit

Der hier beschriebene Stand der Forschung macht ersichtlich, dass die Entwicklung eines menschenähnlichen Roboters, der fähig ist, mit den Menschen durch intuitive Sprache der Emotionen und Gesten zu kommunizieren, die Lösung zahlreicher komplexer Aufgaben erfordert. Diese Aufgaben erstrecken sich von der Entwicklung der Außengestalt des Roboters bis hin zur Entwicklung der Algorithmen der Mensch-Roboter-Interaktion. Das Ziel, dass der Roboter seine eigene Umgebung wahrnimmt und mit den Menschen auf eine natürliche Weise interagiert, bedarf einer systematischen Lösung. Der Einsatz und die Auswertung mehrerer komplexer Sensoren und die Notwendigkeit einer präzisen Steuerung der Aktoren macht ein solches Robotersystem zu einem Beispiel eines anspruchsvollen Automatisierungssystems. Methodologisch gesehen, soll die Realisierung des Steuerungssystems für den mobilen Roboter eine gewisse Ähnlichkeit zur Realisierung des Steuerungssystems einer komplexen Bearbeitungsmaschine aufweisen. Dabei sind die Methoden der schnellen Entwicklung, effizienten Analyse, Optimierung bzw. der Evaluierung der Eigenschaften der Systemkomponenten, wie beispielweise graphische Programmierung, Modellierung, Rapid Prototyping u. a. von großer Bedeutung. Durch Auswahl und Einsatz einer modularen und performanten Softwarearchitektur, sollen die komplexen Aufgaben der Mensch-Roboter-Interaktion realisierbar sein.

Wissenschaftliches Ziel und Aufgabenstellung dieser Arbeit präsentiert die Entwicklung einer Software (SW) und Hardware (HW) Architektur, die auf ein mobiles anthropomorphes Robotersystem optimiert ist. Der besondere Wert dieser SW und HW Architektur soll in einer Modularität, Echtzeitfähigkeit, hohen Leistungsfähigkeit und Effizienz liegen. Eine modulare Architektur bietet deutliche Vorteile durch die Austauschbarkeit und Wiederverwendbarkeit der Komponenten. Die Echtzeitfähigkeit der Systemkomponenten stellt die Voraussetzung bei der Steuerung der zeitkritischen Prozesse dar. Bedingt durch die Mobilität und Autonomie des Robotersystems wird von der SW und HW Architektur die höchste Performanz und Energieeffizienz gefordert.

Zum Nachweis der vorgeschlagenen SW und HW Architektur, werden folgende technische Ziele definiert:

- Entwurf und Implementierung von mechanischen Konstruktionen des Oberkörpers (Kopf, Torso, Arme) für den zweibeinigen mobilen Roboter ROTTO
- Entwurf und Implementierung der Algorithmen der Mensch-Roboter-Interaktion in vorgeschlagener SW und HW Architektur
- Performante und optimierte automatische Abbildung graphischer Interaktionsmodelle auf eine HW Architektur – modulare echtzeitfähige Technologieplattform für Automatisierung der Steuerungsaufgaben.

Die Arbeit ist folgendermaßen gegliedert. Im Kapitel 1 wird zunächst die Problematik der Entwicklung und des Aufbaues einer Klasse der kontaktfreudigen und mobilen Roboter erläutert. In der Folge wird eine Analyse der bekannten Lösungen und Roboterprototypen auf dem Gebiet mobiler und unbeweglicher Roboter vorgestellt. Das Kapitel gibt einen

Überblick über die existierenden Konzepte zur Robotersteuerung für nahe Mensch-Roboter-Interaktionen, sowie auch einen Überblick über die bekannten Werkzeuge und Hardware- und Softwarearchitekturen zur Entwicklung der Steuerungsalgorithmen für mobile Roboter.

Im Kapitel 2 wird die mechatronische Entwicklung der Konstruktion des mobilen Roboters ROTTO präsentiert. Verschiedene Aspekte der Auslegung und des Entwurfs einzelner wichtiger Komponenten werden darin für den Prototyp zweibeiniger Roboter beschrieben.

Das Kapitel 3 gibt einen detaillierten Einblick in die Hardware- und Softwarearchitekturen, auf denen das Steuerungssystem des Roboters basiert. Der Schwerpunkt liegt dabei sowohl auf der schnellen, effizienten und fehlerfreien graphischen Entwicklung und softwareseitigen Wiederverwendung des Programmcodes, als auch auf der Hardwaremodularität des Systems und auf der Anwendung der RCP-Software bei der Entwicklung und Inbetriebnahme.

Das Kapitel 4 widmet sich dem Ziel der Darstellung von Aufgabenebenen für den Roboter ROTTO und der Lösung von einer der Aufgaben, und zwar der Mensch-Roboter-Interaktion. Ausgehend von den Modellen der menschlichen Weltwahrnehmung (von Geburt an bis zum 18. Monat) wird das Konzept der Emotionssteuerung erklärt und auf die komplexeren Mechanismen des Roboterhaltens erweitert. Weiterhin wird ein Emotionssteuerungssystem mit Einsatz der vorgeschlagenen Hardware- und Softwarearchitekturen entwickelt und implementiert.

Das fünfte Kapitel ist der experimentellen Erprobung des entwickelten Emotionssteuerungssystems in realen Mensch-Roboter-Interaktionen und der Untersuchung der Akzeptanz des Roboters ROTTO in Interaktion gewidmet.

Die Zusammenfassung der Ergebnisse dieser Arbeit erfolgt im Kapitel 6.

Kapitel 2

Die mechanische Konstruktion des Roboters ROTTO

2.1 Voraussetzungen

Der in der Otto-von-Guericke-Universität Magdeburg entwickelte anthropomorphe mobile Roboter ROTTO ist im nachfolgenden Bild 2.1 dargestellt.

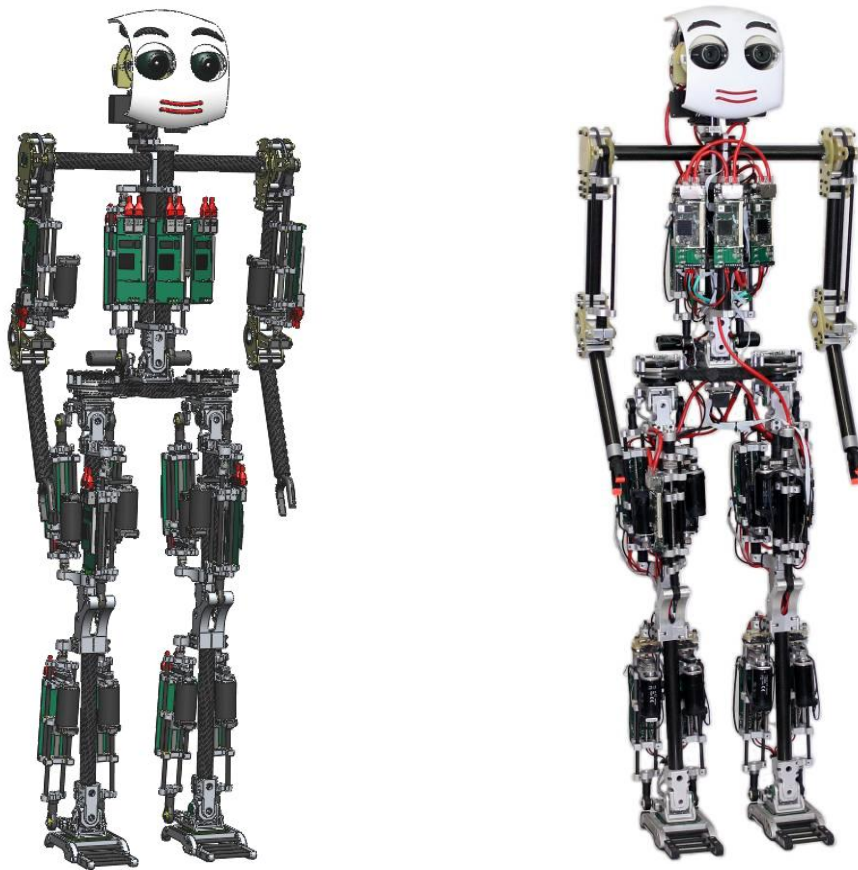


Bild 2.1 – CAD Entwurf und realen Außenansicht des Roboters ROTTO

Die mechanische Konstruktion des Roboters ROTTO besteht aus mehreren Elementen, die spezifisch für das anthropomorphe Wesen sind (Beine, Arme, Körper und Kopf). Der Entwurf und die Umsetzung der mechanischen Konstruktion des Unterkörpers (Hüftgelenk, Kniegelenk, Sprunggelenk und Füße) des anthropomorphen mobilen Roboters ROTTO ist in [48] in Zusammenhang mit der Entwicklung der Laufalgorithmen beschrieben. Im nachfolgenden Unterkapitel 2.2 werden diese Teilkomponenten zwecks Vollständigkeit beschrieben. Dieses

Kapitel umfasst den Entwurf und die Auslegung von mechanischen Konstruktionen für den Oberkörper und insbesondere des Roboterkopfes des anthropomorphen Roboters ROTTO. Diese Komponenten sind im besonderen Maße wichtig für die Entwicklung der Algorithmen zur Mensch-Roboter-Interaktion.

Der Entwurf eines Roboteroberkörpers wird zweistufig angegangen. Der Entwurf des Torso und der Arme ergibt die erste Stufe und verfolgt drei wichtige Aspekte: der erste Aspekt des Entwurfes beschäftigt sich mit der Vervollständigung der Roboterkonstruktion um wesentliche Komponenten und Funktionen, die für die Interaktion mit Menschen wichtig sind. Der zweite Aspekt weist die Nutzung des Torso in [49, 50] als eine Energiequelle aus, die ein zusätzliches Drehmoment für das energiegesteuerte Gehen bereitstellen kann. Der dritte Aspekt beschreibt die Nutzung der Roboterarme im Zusammenspiel mit dem Brustgelenk, um ein Lenkmoment um die vertikale Achse des Roboters zu erzeugen. Dieser Lenkmoment soll zur Kompensierung des während des dynamischen Gehen entstehenden bohrenden Drehmomentes genutzt werden [48, 51].

Der Entwurf des Roboterkopfes stellt die zweite Stufe bei der Entwicklung der mechanischen Konstruktion des Oberkörpers dar. Diese Stufe soll zwei weitere bedeutende Aspekte in der Konstruktion des anthropomorphen Roboters verfolgen. Im Roboterkopf sollen alle Umgebungssensoren (Videokameras, Mikrofonen usw.) integriert werden, die die notwendige Information für die Orientierung des Roboters in einer komplexen Umgebung bereitstellen. Beispielsweise, soll der Roboter in der Lage sein, während des Gehens seine Bewegungsrichtung durch Erkennung von visuellen Umgebungsmerkmalen (z.B. Hindernis) oder Schallzeichen (z.B. Warnsignal) zu korrigieren. Der zweite Aspekt legt seinen Schwerpunkt auf das Roboter Gesicht, das eine intuitive sozial-emotionale Schnittstelle für eine Mensch-Roboter Interaktion bildet.

Basierend auf der Analyse der Ergebnisse bzw. der Erfahrungen anderer Forschungsgruppen [4 – 23, 27, 28, 54] wird die Entwicklung des anthropomorphen Roboters ROTTO in zwei Bereiche aufgeteilt: der Entwurf der mechanischen Roboterkonstruktion und die Entwicklung des Steuerungssystems. Dieses Kapitel widmet sich der Entwicklung mechanischen Konstruktion des Roboters ROTTO.

Die mechanische Konstruktion eines interaktiven Roboters soll in der Lage sein, eine bestimmte Reihe von leicht erkennbaren Bewegungen, die in einem Dialog unterstützend wirken, durchzuführen. Beispielsweise kann der Roboter die Aufmerksamkeit einer Person durch eine Körperwendung in ihre Richtung, die Armenbewegungen und ein Lächeln auf dem Gesicht auf sich lenken. Einen bedeutenden emotionale Wert hat das äußere Erscheinungsbild des Roboters. Dieses hat einen großen psychologischen Einfluss auf den Interaktionspartner. Aus der Studie [2] geht eine wichtige Schlussfolgerung hervor (Bild 2.2), dass bei der Entwicklung einer Roboteraußenansicht das Phänomen „unheimliches Tal“ berücksichtigt werden muss.

Das Phänomen "unheimliches Tal" besagt, dass eine übermäßig äußerliche Ähnlichkeit eines Roboters mit dem Menschen (Bild 2.2) eine negative Wahrnehmung des Roboters bewirkt. Darüber hinaus spielt die Beweglichkeit des Roboters eine wichtige Rolle, deren Amplitude größere Werte erreicht (Bild 2.2). Deshalb leistet die Roboterbeweglichkeit einen großen Beitrag zu der emotionalen Akzeptanz eines mechanischen Wesens. Aufgrund dieser Fakten werden die zusätzlichen Freiheitsgrade im Torso und in den Armen des Roboters ROTTO nicht nur ausschließlich zur Realisierung von Bewegungsaufgaben sondern auch in der Mensch-Roboter Interaktion verwendet.

Der kinematische Aufbau des Oberkörpers des Roboters ROTTO wird in den Unterkapiteln 2.3 und 2.4. näher erklärt. Bei der Auslegung der Rotationsachsen sind nur die für eine menschenähnliche Bewegung unbedingt notwendige Gelenke berücksichtigt. Der Roboter verfügt somit über insgesamt 33 Freiheitsgrade, zehn davon sind in den Beinen, zwei im Becken, sieben im Torso und in den Armen und 14 im Roboterkopf integriert. Die Besonderheit der Konstruktion bilden die modularen positions- bzw. kraftgeregelten Linearantriebe. Detaillierter Aufbau und Anordnung dieser Antriebe sind in [48], [51] beschrieben. Im Rahmen der vorliegenden Arbeit wurde die Konstruktion des Torso, der Arme und des Kopfes anhand der zahlreichen CAD-Modelle und Simulationen untersucht und optimiert.

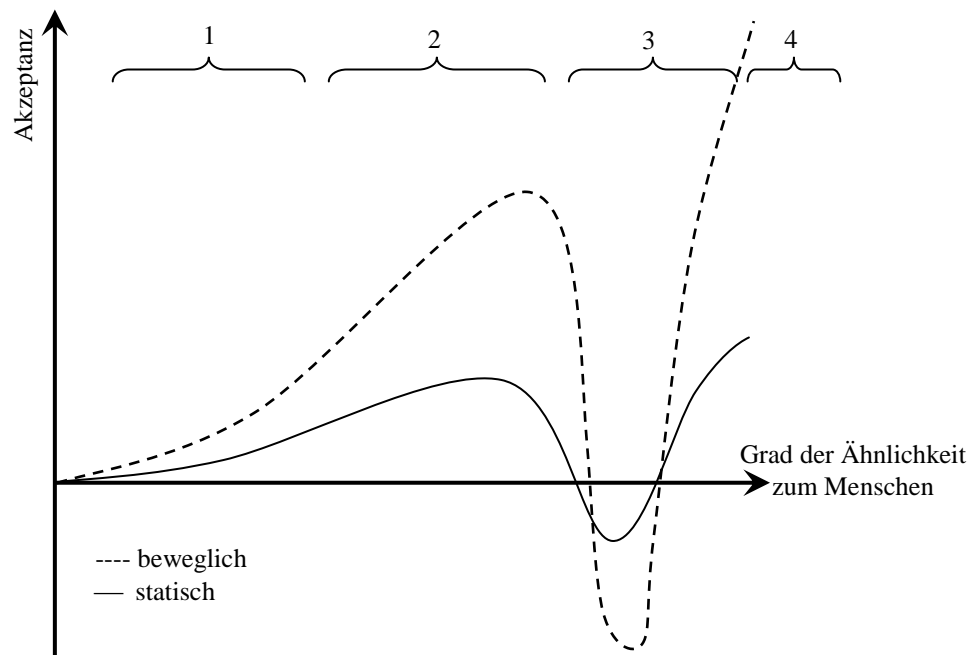


Bild 2.2 – Mit steigender Ähnlichkeit zum Menschen steigt die Roboterakzeptanz, ausschließlich der Phase 3, aus der sich das Phänomen „unheimliches Tal“ ergibt: 1 – industrielle Roboter; 2 – soziale und anthropomorphe Roboter; 3 – „unheimliches Tal“: Roboter mit starker Menschähnlichkeit oder kranke Menschen; 4 – gesunde Menschen

2.2 Unterkörper

Der Entwurf des Unterkörpers für den zweibeinigen Roboter ROTTO wurde in [48] beschrieben. Zur vollständigen Präsentation der Roboterkonstruktion wird im anschließenden Unterkapitel die konstruktive Umsetzung des Unterkörpers vorgestellt.

2.2.1 Linearantrieb

Der Roboter weist eine modulare Struktur auf, die durch die Verwendung von modularen Antriebsmodulen (Bild 2.3), und Leichtbaukonstruktionen gekennzeichnet ist. Der eingesetzte Antriebsmodul präsentiert einen universellen Linearantrieb, der lagegeregt [44] bzw. kraftgeregt [46] betrieben wird.

In dem Linearantrieb treibt ein leistungsfähiger bürstenloser Motor von Maxon [52] mittels einer Zahnriemenstufe ein Kugelgewindetrieb an. Die lineare Bewegung der Kugelmutter wird mit Hilfe von steifen Kohlefaserstäben, die gleichzeitig als Führungen benutzt werden, ausgeführt. Die Rotorposition des Motors wird mittels Inkrementalgeber mit hoher Auflösung

erfasst. Zwei magnetische Endschalter sind zur Überwachung des Arbeitsbereichs in die Konstruktion integriert. Ein integriertes Elektronikmodul übernimmt die Regelung bzw. Überwachung des Antriebs, die Datenerfassung der Sensoren und Datenaustausch per Ethernet-Netzwerk. Für die Kraftregelung wird der Linearantrieb zusätzlich mit einem seriellen elastischen Element (Kohlenfaserfeder) und einem Kraftsensor ausgerüstet. Die komplette Integration des Motors, des Getriebes, der Sensoren und der elektronischen Steuerung in einem mechatronischen Modul, bietet deutliche Vorteile, durch ein geringes Bauvolumen des Antriebsmoduls, bessere dynamische Eigenschaften, hohe Leistungsdichte und bessere Effizienz, hohe Zuverlässigkeit durch bürstenlose Motoren und die modulare Bauweise, sowie in den integrierten Überwachungs- und Diagnosefunktionen.

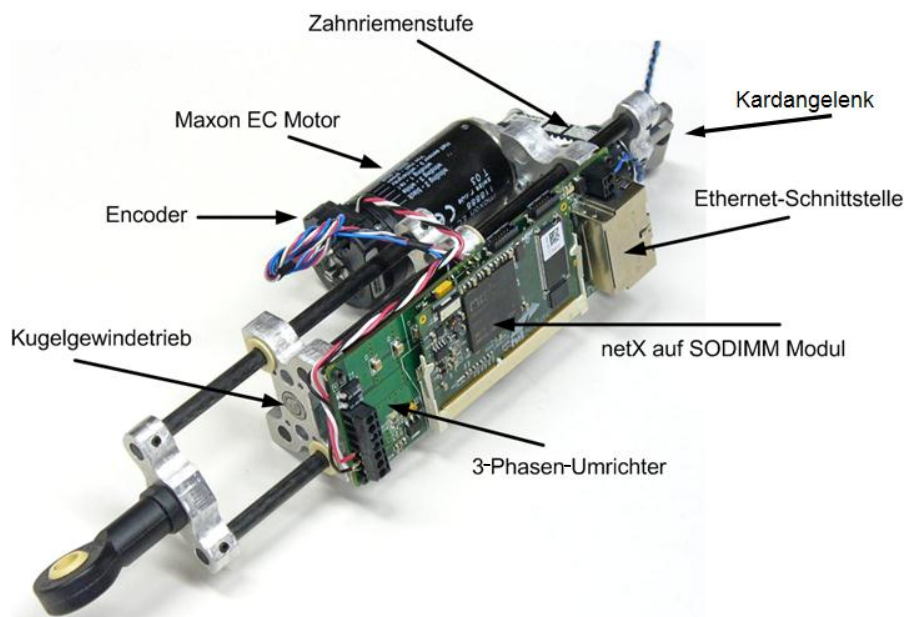


Bild 2.3 – Universeller integrierter Linearantrieb

2.2.2 Beine

Jedes Bein des zweibeinigen Roboters verfügt über drei Gelenke: Hüftgelenk, Kniegelenk und Sprunggelenk. Im Bild 2.4 ist die konstruktive Umsetzung des Hüftgelenks für den zweibeinigen Roboter ROTTO veranschaulicht. Die Konstruktion des Hüftgelenks wird ähnlich dem Sprunggelenk und dem Kreuzbeingelenk in Form einer parallelen Kinematik realisiert. Zwei Linearantriebe setzen mittels Schubstangen und Kugelgelenken das Roboterbein in Bewegung.

Der Konstruktion des Kniegelenks wurde ebenfalls ein Linearantrieb zu Grunde gelegt, der mittels Schubstange die Unterschenkel in Bewegung setzt. Der Linearantrieb ist ähnlich wie die anderen Antriebe von einer Seite mittels eines kompakten Universalgelenks an dem tragenden Kohlefaserrohr des Oberschenkels befestigt. An der anderen Seite überträgt die Schubstange die Kraft des Linearantriebes an ein Befestigungsbauteil des Unterschenkels. Die Kniegelenkposition wird mit Hilfe eines magnetischen Winkelgebers kontaktfrei erfasst und in digitaler Form durch das Steuerungssystem gelesen. Bild 2.5 veranschaulicht übersichtlich die konstruktive Umsetzung des Kniegelenks des zweibeinigen Roboters ROTTO.

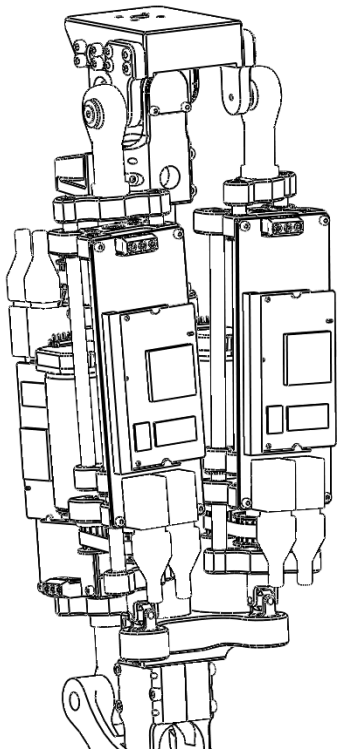


Bild 2.4 – Konstruktive Umsetzung des Hüftgelenks

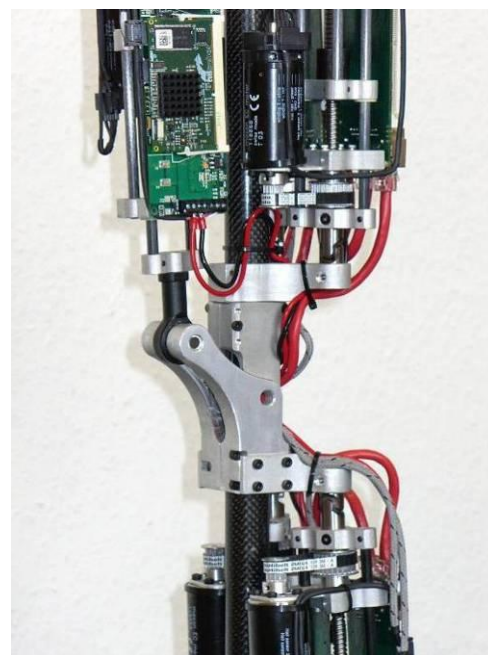
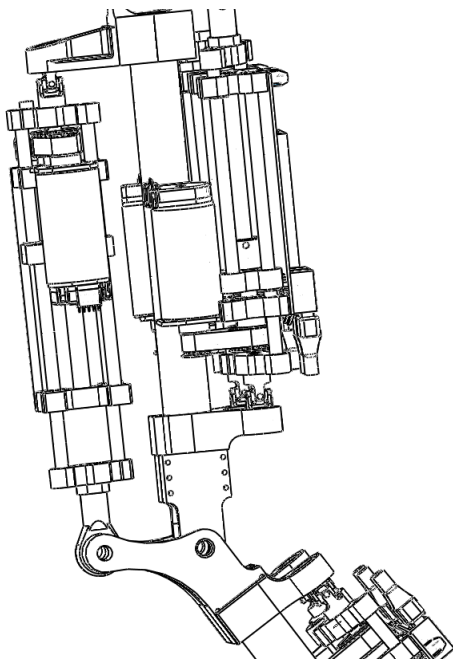


Bild 2.5 – Konstruktive Umsetzung des Kniegelenks

Die parallele Kinematik bestimmt hier, wie auch beim Hüftgelenk, die Ansteuerung des Sprunggelenkes. Die Position desselben wird dabei durch zwei magnetische Winkelgeber berührungslos erfasst und durch eine digitale Schnittstelle in das Steuerungssystem eingelesen.

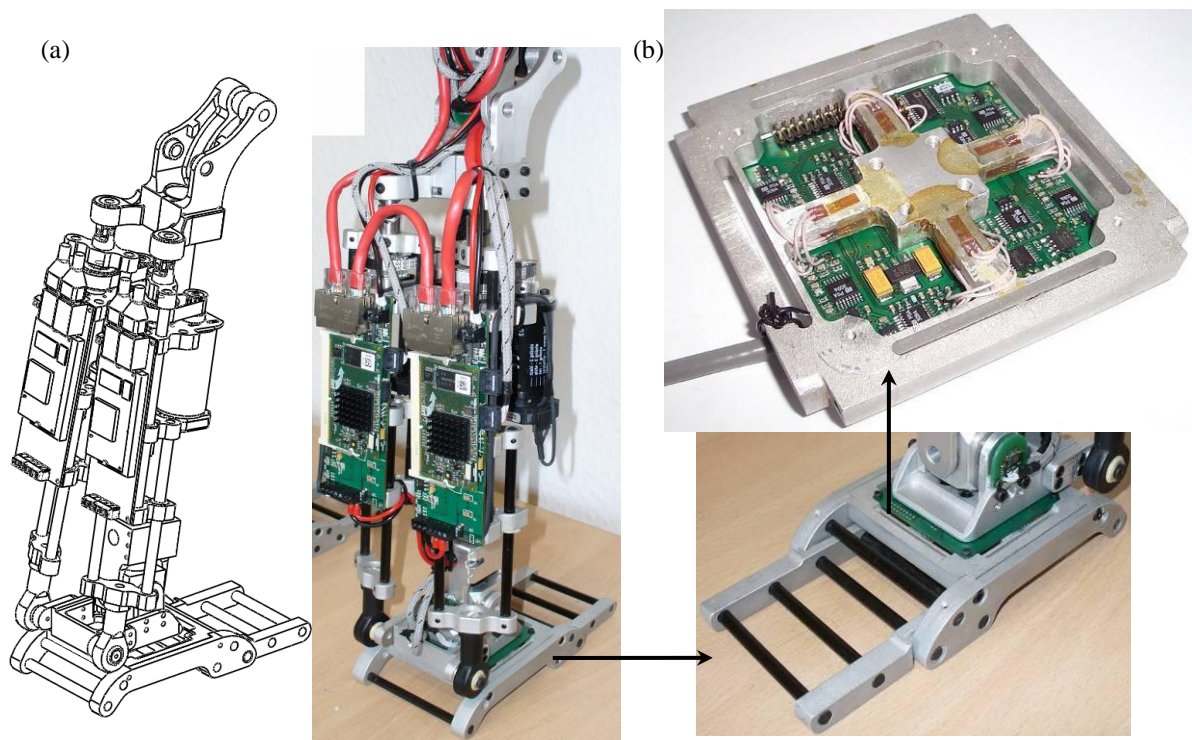


Bild 2.6 – Konstruktive Umsetzung des Sprunggelenks (a) und des Fußes (b)

Der Roboterfuß ist, während der Bewegung, der größten Beschleunigung im System ausgesetzt, deshalb wird der Fuß mit möglichst wenig Gewicht konstruiert. Gleichzeitig ist der Fuß stabil genug konstruiert, um das gesamte Gewicht des Roboters und wesentliche Anstoßkräfte aufzunehmen. Zur Messung der auf den Fuß einwirkenden Kräfte und Momente wurde ein 6-Komponenten Kraft-/Momentsensor in die Fußkonstruktion integriert. Bild 2.6 zeigt die konstruktive Umsetzung des Sprunggelenks, des Fußes und des Kraftsensors des anthropomorphen Roboters ROTTO.

2.2.3 Becken

Die Konstruktion des Beckens wurde stark auf das Gewicht ausgerichtet. Zur Reduzierung des Eigengewichts und Erhöhung der Steifigkeit, wurde das Roboterbecken aus mehreren zusammengeklebten flachen und leichten Kohlefaserteilen konstruiert. Aufgrund des kleinen Bauvolumens ist der bereits beschriebene Linearantrieb in der Konstruktion des Beckens nicht einsetzbar. Die neue Konstruktion des Linearantriebs wurde auf die Anforderungen bezüglich des Bauvolumens und Funktion ausgelegt und in die Beckenkonstruktion integriert. Dabei wird die vertikale Rotationsachse des Hüftgelenks samt Beinbefestigungsfläche von einem Linearantrieb bewegt. Die lineare Antriebsbewegung wird durch einen bürstenlosen Motor von Maxon mittels einer Zahnriemenstufe und einem Kugelgewindetriebe realisiert. Die lineare Bewegung der Kugelmutter wird mit Hilfe von zwei Hebeln in eine Rotation der im Kunststofflager (Igus [53]) integrierten Beinbefestigungsfläche umgewandelt. Zwei flache Kohlefasern erlauben dem Linearantrieb eine spiellose Lenkung in der horizontalen

Ebene. Die Rotation des Beins wird durch zwei magnetische Winkelgeber erfasst und von zwei optischen Endschaltern und Halteschrauben vor unerlaubten Drehwinkeln geschützt.

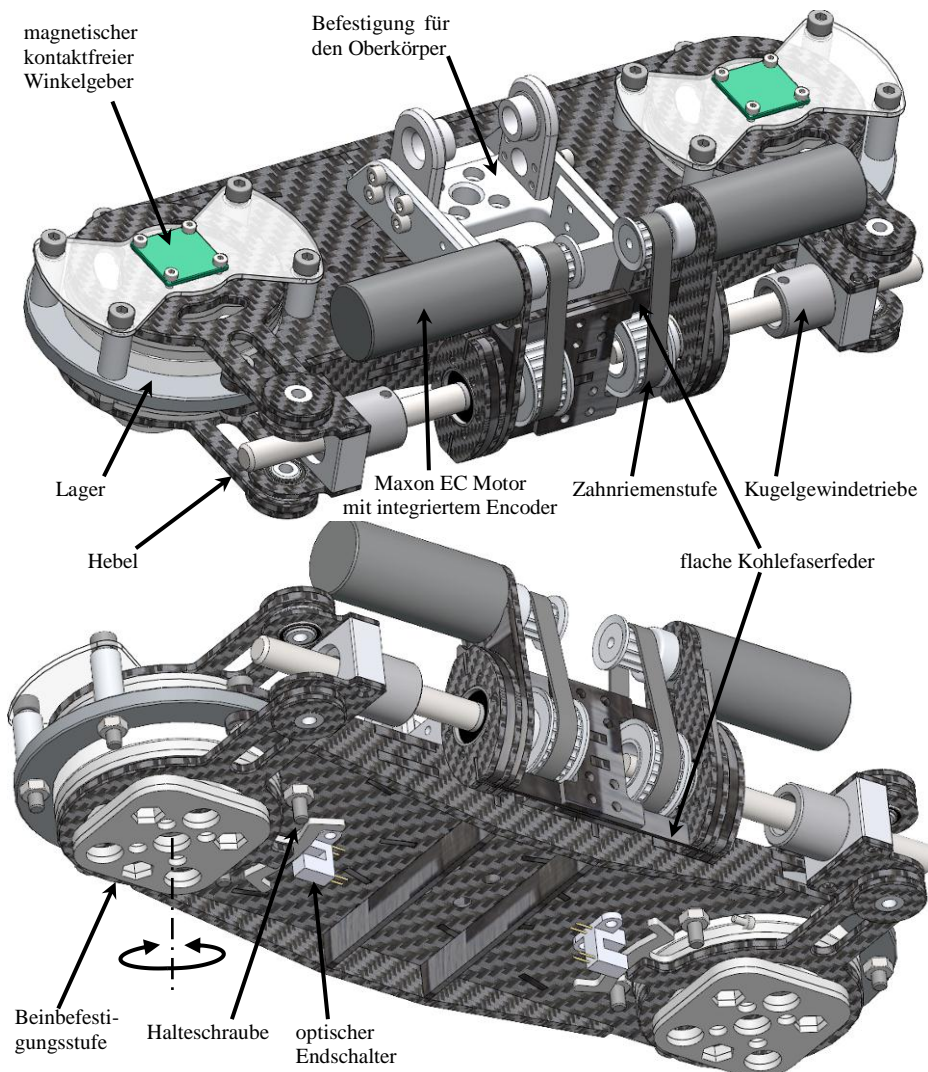


Bild 2.7 – Konstruktive Umsetzung des Beckens

2.3 Torso und Arme

Die tragende Skelettstruktur des Torsos und der Arme wird als Leichtbau-Konstruktion aufgeführt. Diese besteht aus einem steifen und leichten Kohlefaserrohr und mehreren auf sein Gewicht optimierten, aus Aluminium gefrästen Befestigungselementen.

Bild 2.8 präsentiert den prinzipiellen Gesamtaufbau des Oberkörpers des mobilen Roboters ROTTO als CAD-Entwurf sowie die aktuelle Realisierung. Zur Steigerung der Effizienz wurde eine parallelkinematische Gelenkstruktur am Kreuzbeingelenk gewählt. Durch zahlreiche Entwicklungsiterationen, in denen die Anzahl der Teile, der Kraftfluss, sowie das Gewicht jeder Einheit optimiert wurden, wurde ein geringes Gewicht des Roboters und gleichzeitig eine ausreichende Robustheit und Steifigkeit erzielt.

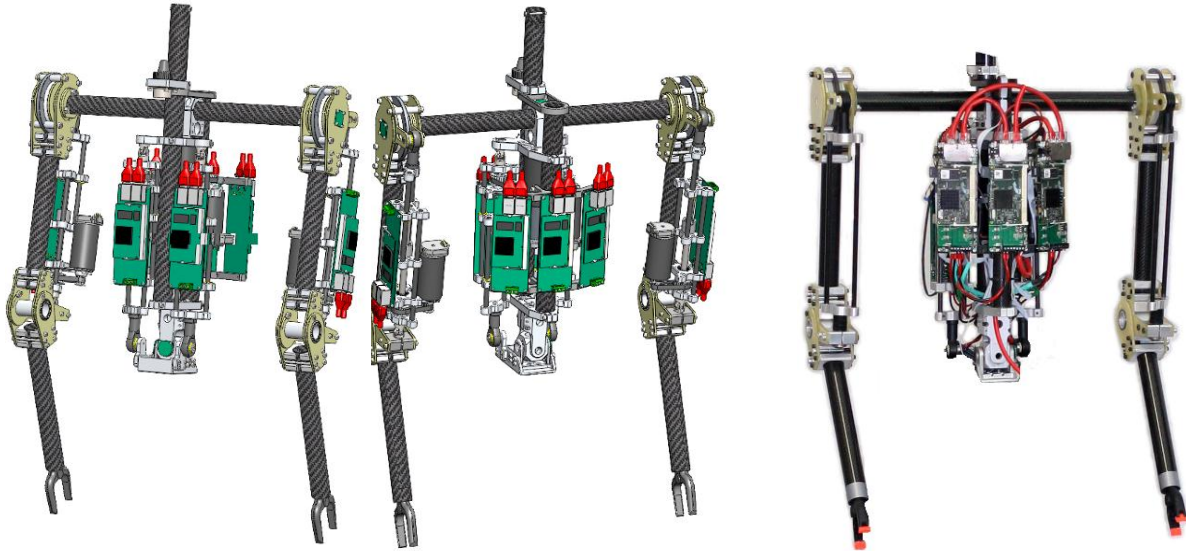


Bild 2.8 – CAD-Entwurf und Realisierung (rechts) des Roboteroberkörpers ROTTO

2.3.1 Kreuzbeingelenk

Das Kreuzbeingelenk erlaubt die Neigungen des Oberkörpers in der Sagittal- und Frontalebene (die Achsen X und Z im Bild 2.9). Die Konstruktion des Kreuzbeingelenks ist in Form einer parallelen Kinematik ausgeführt. Zwei Linearantriebe setzen mittels Schubstangen und Kugelgelenken den Roboterkörper in Bewegung. Die Bewegung des Kreuzbeingelenks in sagittaler Ebene (α_s) erfolgt durch die synchrone Ansteuerung der Linearantriebe in der gleichen Richtung (F_{1O} , F_{2O} oder F_{1U} , F_{2U}). Die Bewegung der frontalen Ebene (α_f) ist durch die synchrone Ansteuerung der Linearantriebe in entgegengesetzter Richtung (F_{1O} , F_{2U} oder F_{1U} , F_{2O}) möglich.

Bild 2.9.b veranschaulicht das kinematische Modell des Kreuzbeingelenks mit zwei Freiheitsgraden des zweibeinigen Roboters ROTTO. Die Rückwärtstransformation bzw. inverse Kinematik ergibt sich aus der Berechnung der Längen x_l und x_r anhand der bekannten Gelenkwinkel α_s und α_f . Die Längen x_l und x_r werden durch den Abstand zwischen den entsprechenden Punkten O_l , U_l und O_r , U_r und deren räumlichen Koordinaten bestimmt:

$$O_l = \begin{bmatrix} O_{lX} \\ O_{lY} \\ O_{lZ} \\ 1 \end{bmatrix}, \quad O_r = \begin{bmatrix} O_{rX} \\ O_{rY} \\ O_{rZ} \\ 1 \end{bmatrix}, \quad U_l = \begin{bmatrix} U_{lX} \\ U_{lY} \\ U_{lZ} \\ 1 \end{bmatrix}, \quad U_r = \begin{bmatrix} U_{rX} \\ U_{rY} \\ U_{rZ} \\ 1 \end{bmatrix} \quad (2.9)$$

$$x_l = \sqrt{(O_{lX} - U_{lX})^2 + (O_{lY} - U_{lY})^2 + (O_{lZ} - U_{lZ})^2} \quad (2.10)$$

$$x_r = \sqrt{(O_{rX} - U_{rX})^2 + (O_{rY} - U_{rY})^2 + (O_{rZ} - U_{rZ})^2}$$

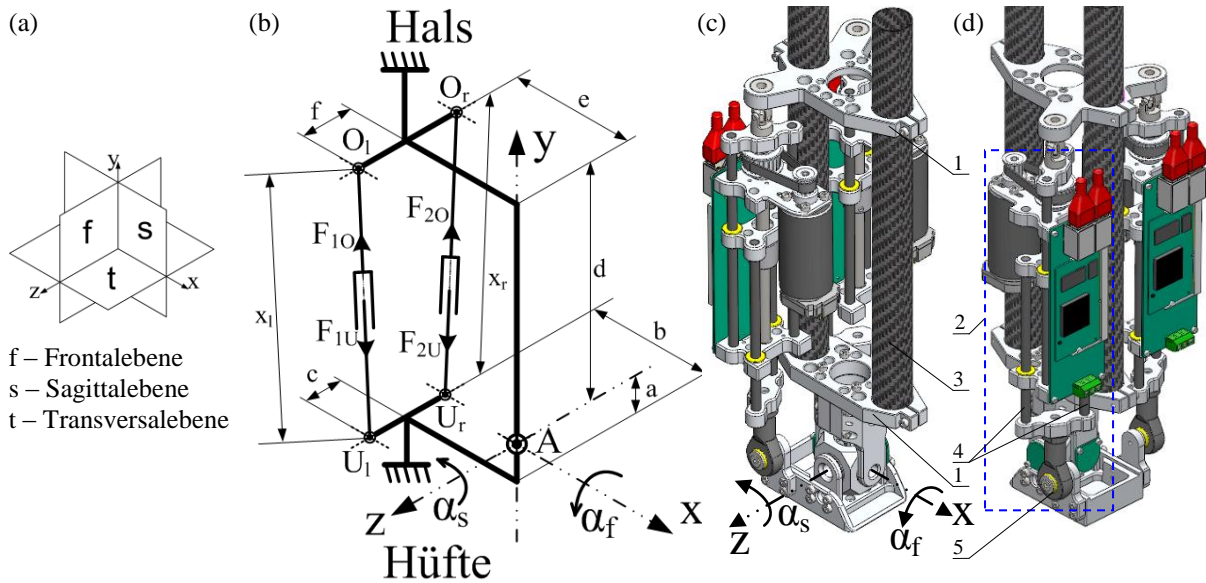


Bild 2.9 – Konstruktion des Kreuzbeingelenks: a – Körperebenen; b – Kinematik des Kreuzbeingelenks; c, d – CAD-Entwurf, Front- und Rückansicht (1 – Befestigungselementen, 2 – universeller Linearantrieb, 3 – Kohlefaserrohr bzw. Körperskelett, 4 – Schubstangen, 5 – Kugelgelenk)

Das Bezugskoordinatensystem wird im Punkt A mit den Koordinaten $[0 \ 0 \ 0 \ 1]^T$ gewählt. Die entsprechenden Koordinaten der Punkte O_1 , U_1 und O_r , U_r werden mit Hilfe von Transformationen ermittelt.

Die translatorischen Transformationen für die einzelnen Punkte (von A nach U_1 und von A nach U_r) werden auf folgende Weise beschrieben:

$$T_{U_1}^A = \begin{bmatrix} 1 & 0 & 0 & -b \\ 0 & 1 & 0 & -a \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad T_{U_r}^A = \begin{bmatrix} 1 & 0 & 0 & -b \\ 0 & 1 & 0 & -a \\ 0 & 0 & 1 & -c \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

Anschließend für die Koordinaten der Punkte U_1 und U_r gilt:

$$U_1 = T_{U_1}^A \cdot A, \quad U_r = T_{U_r}^A \cdot A \quad (2.2)$$

Im Punkt A werden die entsprechenden Rotationen auf den Winkeln α_f und α_s um die Achsen X und Z durch folgende Rotationsmatrizen berechnet:

$$R_{\alpha_f}^A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_f) & -\sin(\alpha_f) & 0 \\ 0 & \sin(\alpha_f) & \cos(\alpha_f) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad R_{\alpha_s}^A = \begin{bmatrix} \cos(\alpha_s) & \sin(\alpha_s) & 0 & 0 \\ -\sin(\alpha_s) & \cos(\alpha_s) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

Die translatorischen Transformationen für die einzelnen Punkte (von A nach O_l und von A nach O_r) werden auf folgende Weise errechnet:

$$T_{O_l}^{\alpha_s} = \begin{bmatrix} 1 & 0 & 0 & -e \\ 0 & 1 & 0 & d \\ 0 & 0 & 1 & f \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad T_{O_r}^{\alpha_s} = \begin{bmatrix} 1 & 0 & 0 & -e \\ 0 & 1 & 0 & d \\ 0 & 0 & 1 & -f \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2 \quad .)$$

Mit Hilfe der vorgestellten Transformationen gilt für die Koordinaten der Punkte O_l und O_r folgende Gleichung:

$$O_l = R_{\alpha_f}^A \cdot R_{\alpha_s}^{\alpha_f} \cdot T_{O_l}^{\alpha_s} \cdot A, \quad O_r = R_{\alpha_f}^A \cdot R_{\alpha_s}^{\alpha_f} \cdot T_{O_r}^{\alpha_s} \cdot A \quad (2 \quad .)$$

2.3.2 Brustgelenk

Das Brustgelenk lässt die Roboterschultern mit den Armen in der Transversalebene um die vertikale Achse Y rotieren. Durch das angetriebene Moment des Brustgelenks, wird das beim dynamischen Gehen entstandene Drehmoment kompensiert. Die Konstruktion des Brustgelenks wird mit einer räumlichen Optimierung des Körperskelettes ausgeführt (Bild 2.10). Der Synchronmotor mit dem Getriebe ist zwischen zwei Kohlefaserrohren des Körperskelettes mittels Befestigungselemente an der Schulter fixiert. Direkt am Getriebe ist das Kohlefaserrohr des Schlüsselbeines befestigt.

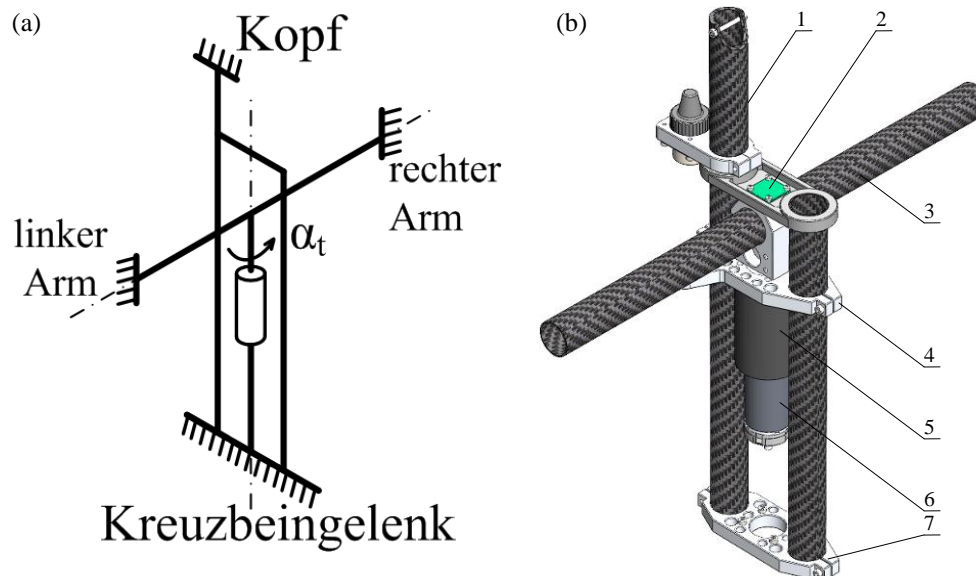


Bild 2.10 – Konstruktion des Brustgelenks: a – Kinematik des Kreuzbeingelenks; b – CAD-Entwurf (1 – Hals, 2 – magnetischer Winkelsensor, 3 – Kohlefaserrohr des Schlüsselbeines, 4 – Befestigungselement der Schulter, 5 – Getriebe, 6 – Synchronmotor, 7 – Befestigungselement für die Ankopplung des Kreuzbeingelenkes)

Der Drehwinkel α_t des Brustgelenks kann von der Rotorposition des Motors durch den Umrechnungsfaktor der Getriebe abgeleitet werden.

2.3.3 Schultergelenk

Das in Bild 2.11 dargestellte Schultergelenk ist das einzige aktive Gelenk im Roboterarm.

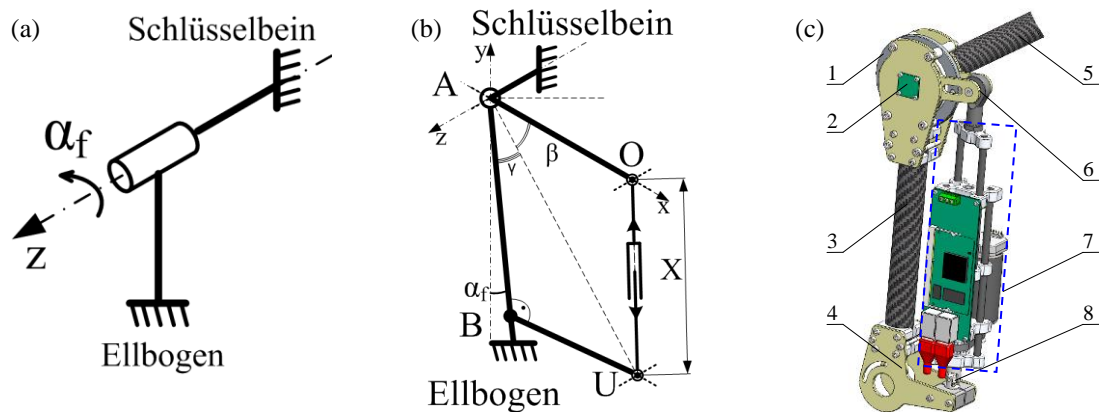


Bild 2.11 – Konstruktion des Schultergelenks: a – Kinematik; b – Kinematik zur Berechnung der Rückwärtstransformation; c – CAD-Entwurf (1 – Schulterlager, 2 – magnetischer Winkelsensor, 3 – Kohlefaserrohr der Schulter, 4 – Befestigungselement aus Glasfaserkunststoff, 5 – Schlüsselbein, 6 – Kugelgelenk, 7 – Linearantrieb, 8 – Kardangelen)

Der Arm ist in der Frontalebene mittels Schulterlager beweglich gestaltet und ist mit einem Fixierungselement auf dem Schlüsselbein befestigt. Der universelle Linearantrieb setzt mittels Kardangelenks, Schubstangen und Kugelgelenken den Roboterarm in Bewegung. Zur mechanischen Ankopplung des Linearantriebes werden robuste und leichtgewichtige Befestigungselemente aus Glasfaserkunststoff eingesetzt.

Die Rückwärtstransformation für das Schultergelenk ergibt sich aus der Berechnung der Länge X anhand des bekannten Gelenkwinkels α_f . Die Länge X wird aus dem Dreieck ΔAOU errechnet:

$$X = \sqrt{AO^2 + AU^2 - 2 \cdot AO \cdot AU \cdot \cos(\beta)} \quad (2.1)$$

$$\beta = \frac{\pi}{2} - \gamma - \alpha_f$$

Die Hypotenuse AU und der Winkel γ werden aus dem Dreieck ΔAUB bestimmt:

$$AU = \sqrt{AB^2 + BU^2} \quad (2.2)$$

$$\gamma = \arctg\left(\frac{BU}{AB}\right)$$

2.3.4 Ellbogengelenk

Da die Manipulation- bzw. Handhabungsaufgaben mit Hilfe der Roboterarme keinen Schwerpunkt der Arbeit darstellen, wurde das Ellbogengelenk passiv bzw. nicht angetrieben konstruiert. Die vereinfachte Konstruktion der Arme wird im Bild 2.12 gezeigt. Das Ellbogengelenk erlaubt dem Arm in einer vorbestimmten Position mittels einer Halteschraube innerhalb des Fixierungseinschnittes zu fixieren. Damit kann das Trägheitsmoment des Armes während der Untersuchung der Robotersteuerung variiert werden. Für solche Versuche wird die Roboterhand mit einem bestimmten Gewicht beschwert.

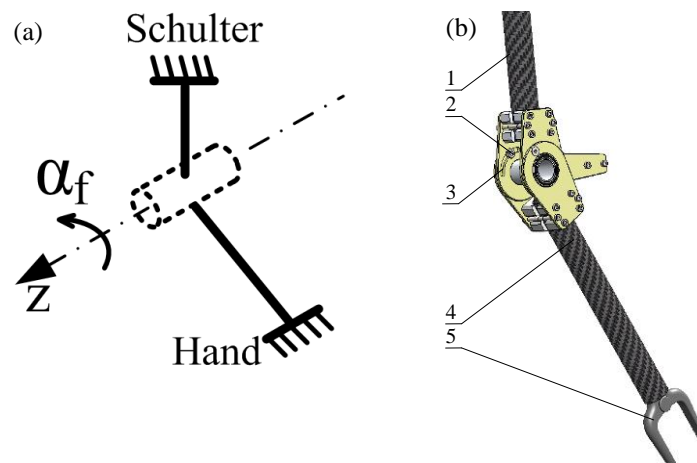


Bild 2.12 – Ellbogengelenk: a – Kinematik; b – CAD-Entwurf (1 – Schulter, 2 – Halteschraube, 3 – Fixierungseinschnitt des Ellbogens, 4 – Kohlefaserrohr des Unterarmes, 5 – Hand)

2.4 Kopf

Wie in vorigem Kapitel bereits angedeutet, wird das äußere Erscheinungsbild eines interaktiven Roboters als eines der wichtigsten Aspekte für die Mensch-Roboter-Interaktion angesehen. Das menschliche Gesicht verfügt über ein großes Repertoire an Mimiken, Gestiken und kann dadurch Emotionen ausdrucksstark wiedergeben. Obwohl die menschlichen Gesichter sich stark unterscheiden, ist die Erkennung der Emotionen zwischen den Menschen fast immer erfolgreich.

Bei der Suche nach der Gesichtsgestaltung kann z.B. die Gesichtsdarstellung in Zeichentrickfilmen nachverfolgt werden. Hier werden die Gesichter in drei Arten klassifiziert: realitätsnah, ikonisch und abstrakt. Diese Klassifizierung hat Scott McCloud [55] in einem Gesichtsdreieck zusammengefasst. Bild 2.13 veranschaulicht diese Gesichtsraum-Darstellung. In der linken Ecke sind realitätsvolle detaillierte Gesichter, auf denen eine Person erkennbar ist, platziert. Im Verlauf zur rechten Ecke werden die Gesichter immer ikonischer und die Detaillierung, die Gesichtsrealität und die Persönlichkeit gehen dabei verloren. Es bleiben lediglich die Gesichtszüge, die für eine deutliche Informationsübergabe bzw. Emotionsausdrücke erforderlich sind. In der oberen Ecke werden die Gesichter abstrakt, die Aufmerksamkeit eines Zuschauers wird nicht auf die Gesichtsbewegung, sondern auf das Vorhandensein einer gesichtsnahen Abstraktion konzentriert.

Dieser Gesichtsraum wurde in [27] bei der Gestaltung des Robotergesichtes berücksichtigt. Im Bild 2.13 wird eine Roboterreihe im Gestaltungsraum vorgestellt.

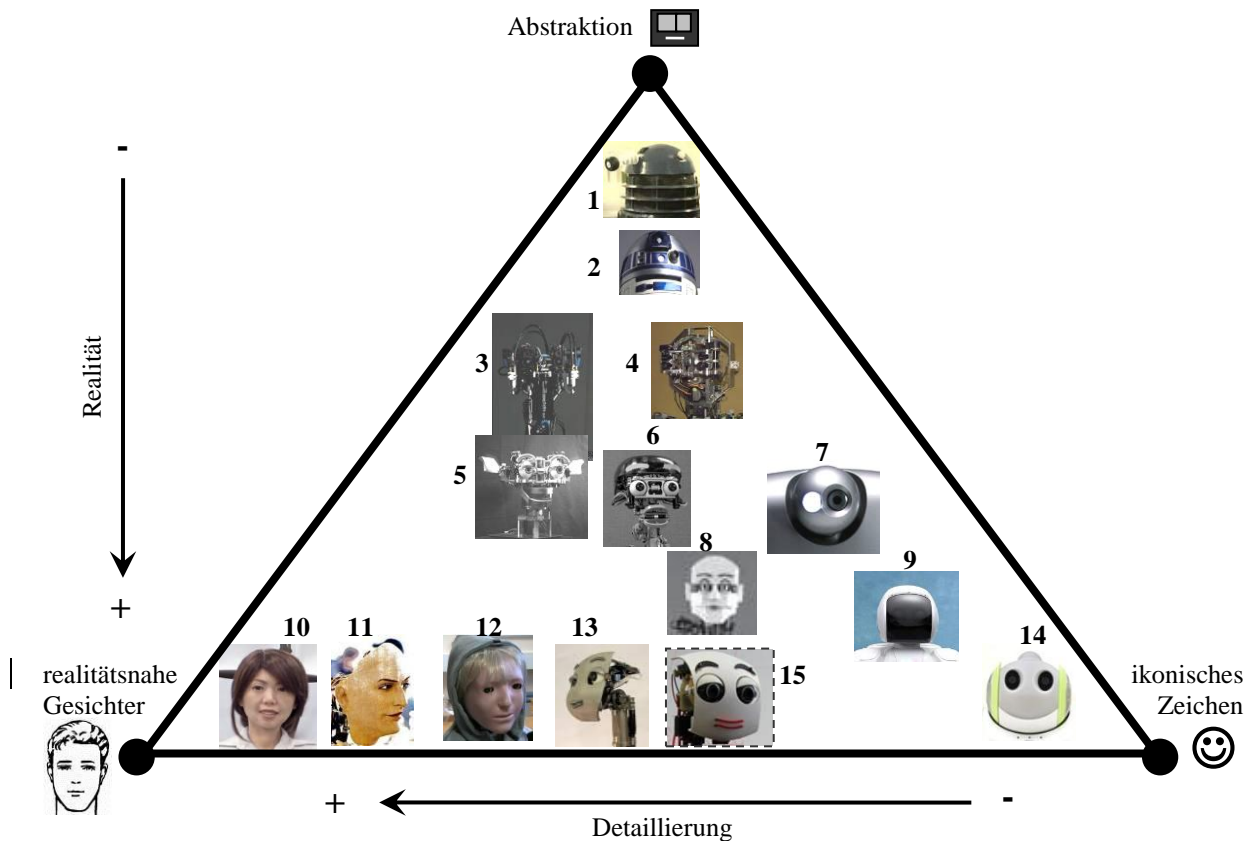


Bild 2.13 – Dreieckiger McCloud’s Designraum für die Robotergesichter: 1 – Dalek [56], 2 – R2D2 [57], 3 – DB [19], 4 – COG [21], 5 – Kismet [5], 6 – Infantoid [58], 7 – Nuvo [59], 8 – Kobian [62], 9 – Asimo [60], 10 - Repliee Q2 [64], 11 – K-Bot [63], 12 – Kaspar [27], 13 – Merz [6], 14 – Papero [61], 15 – ROTTO

Die Robotergestaltung in jedem Extremfall bzw. in jeder Ecke des Dreiecks bietet eigene Vor- und Nachteile. Beispielsweise gibt es keine Gefahr mit der abstrakten Gestaltung in das „unheimliches Tal“ (Bild 2.2 – statisch, Anfang des ersten Bereiches) zu geraten, weil es hier ein abstraktes (nicht menschenähnliches) Äußere gibt. Ungünstig sind hier die stark begrenzte emotionale Kommunikation und keine vorhandene Persönlichkeit. In der ikonischen Gestaltung ist das attraktive und einfache Äußere von Vorteil. Das „unheimliche Tal“ ist noch nicht erreichbar (Bild 2.2 – Ende des ersten, Anfang des zweiten Bereiches). Nachteilig wirken immer noch die schwache Persönlichkeit und begrenzte emotionale Kommunikation. Die realitätsnahe Darstellung präsentiert eine gute physikalische Anwesenheit, verschiedene komplizierte Emotionen sind für die visuelle Rückkopplung in einem Dialog einsetzbar. Diese Darstellung enthält eine hohe Persönlichkeit – eine reelle Person kann durch den realitätsvollen bzw. realistisch gestalteten Roboter mit hoher Glaubwürdigkeit wiedergegeben werden. Vom Nachteil sind hier die Konstruktionskomplexität und die Gefahr in das „unheimliches Tal“ zu gelangen (Bild 2.2 – Ende des zweiten, Anfang des dritten Bereiches).

Basierend auf der vorgestellten Analyse des McCloud’s Designraums für die Robotergesichter und auf den Anforderungen für einen interaktiven Roboter, wurde der ROTTO mit der Gesichtsgestaltung zwischen dem realitätsnahen und ikonischen Äußeren (Bild 2.14) entworfen. Im McCloud’s Designraum, in Bild 2.13, wird ROTTO unter Nummer 15 platziert. Dieser Platz fasst die Vorteile der realitätsnahen und ikonischen Gesichter zusammen: gute Emotionendarstellung, attraktive und einfache Außenansicht, sicher vom „unheimlichen Tal“ entfernt.



Bild 2.14 – CAD-Entwurf und Realisierung des Kopfes des interaktiven Roboters ROTTO

Die Konstruktion des Roboterkopfes zeigt einen menschenähnlichen Aufbau mit entsprechenden Gesichtsteilen, wie Augen, Augenbrauen, Augenlider und Lippen. Der Roboterkopf besitzt insgesamt 14 Freiheitsgrade, die mit Hilfe von 14 Servoantrieben angetrieben werden. Die Konstruktion des Roboterkopfes wird in den nachfolgenden Unterkapiteln detailliert betrachtet.

2.4.1 Differentiales Halskreuzgelenk

Das Halsgelenk ist ein Verbindungsglied zwischen dem Robotertorso und dem Kopf. Das Gelenk ermöglicht die gleichzeitigen Kopfrotationen in transversaler und sagittaler Ebene (Bild 2.15).

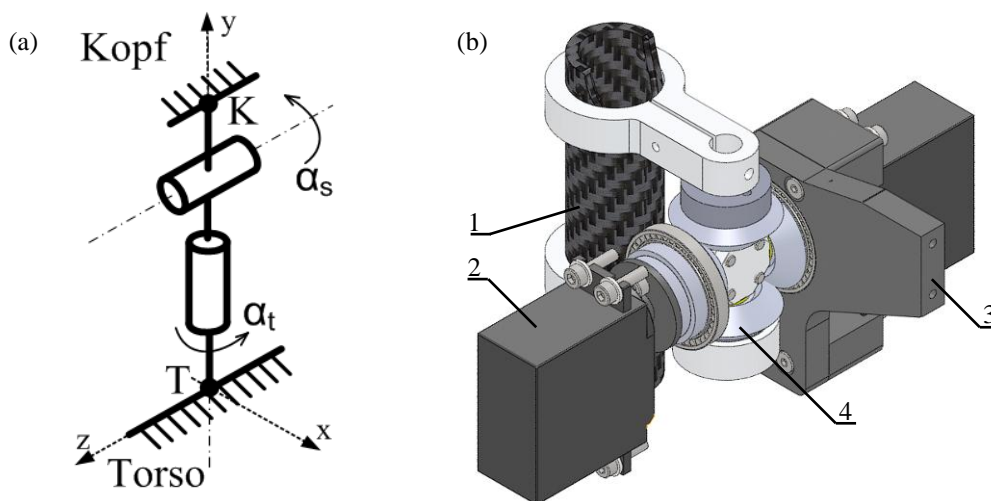


Bild 2.15 – Halskreuzgelenk: a – Kinematik des Kreuzgelenkes; b – CAD-Entwurf (1 – Hals, 2 – Servoantriebe, 3 - Kopfkopplung, 4 – Zahnräder)

Der Konstruktion des Halsgelenkes wurde ein differentiales Kreuzgelenk zu Grunde gelegt, das mittels zweier Servoantriebe den Roboterkopf in Bewegung setzt. Dank der differentialen Gelenkkonstruktion nimmt jeder Antrieb nur die Hälfte der Gesamtlast auf und kann entsprechend kleiner ausgewählt werden. Nach dem kinematischen Schema kann die räumliche Koordinatentransformation zwischen den Punkten K und T mit Hilfe von Formel 2.10 und Rota-

tionsmatrizen 2.11 berechnet werden. Auf diese Weise wird der Richtungsvektor des Roboterkörpers in Bezug auf den Robotertorso ermittelt.

$$K = \begin{bmatrix} x_K \\ y_K \\ z_K \\ 1 \end{bmatrix}, T = \begin{bmatrix} x_T \\ y_T \\ z_T \\ 1 \end{bmatrix}, K = R_{\alpha_t}^T \cdot R_{\alpha_s} \cdot T \quad (2.1)$$

$$R_{\alpha_t}^T = \begin{bmatrix} \cos(\alpha_t) & 0 & -\sin(\alpha_t) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(\alpha_t) & 0 & \cos(\alpha_t) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, R_{\alpha_s} = \begin{bmatrix} \cos(\alpha_s) & \sin(\alpha_s) & 0 & 0 \\ -\sin(\alpha_s) & \cos(\alpha_s) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

2.4.2 Augenkonstruktion

Bei der Konstruktion des Roboterauges wurde die Anatomie des menschlichen Auges untersucht. Der in [65] detailliert beschriebene Aufbau des Auges und der Augenmuskeln sind im Bild 2.16 veranschaulicht. Innerer und temporaler geraden Muskeln sind in der Konstruktion des Auges mit dem horizontalen Windenkopf d_t , Seil und der Vorspannfeder E_t (Bild 2.17) realisiert. Ähnlicher Aufbau wurde in der Sagittalebene übernommen, wobei die oberen und unteren geraden Muskeln durch den Windenkopf d_s und die Vorspannfeder E_s realisiert sind. Zwecks der Vereinfachung der mechanischen Augekonstruktion wurde auf die Realisierung der Funktion der schrägen Augenmuskeln bewusst verzichtet. Die ausgewählte Lösung resultiert in einer einfachen kompakten und gewichtsoptimierten Konstruktion (Bild 2.17).

Der Konstruktion des Auges wurde ein Kugelgelenk zu Grunde gelegt, das drei Freiheitsgrade α_t , α_s und α_f , (Bild 2.17) besitzt. Ein kugelförmiger Kamerakörper stellt den Gelenkkopf dar. Die Gelenkpfanne ist aus dem Fluorkunststoff mit einem kleinen Reibungsfaktor gefräst. Das Auge ist in sagittaler und transversaler Ebene angetrieben, die frontale Ebene ist frei beweglich. Die Augenkugel ist mit den Windenköpfen mittels vorgespannter (Vorspannfedern E_s , E_t) hochfester Polyethylen-Faser-Seile verbunden. Die Seile sind konzentrisch an der Augenkugel angekoppelt, sodass der Ablaufdurchmesser d_k bei einem Seilablauf bzw. einer Augenbewegung konstant bleibt. Dank der Seilvorspannung wird in der Neutralposition des Auges ($\alpha_t=0$ und $\alpha_s=0$) die entstandene frontale Augedrehung minimiert. Darüber hinaus wird, dank dieser Vorspannung, für jede angetriebene Achse nur ein Servoantrieb benötigt. Jeder Servoantrieb treibt einen Windenkopf an, auf dem das Seil jeweils aufgewickelt wird.

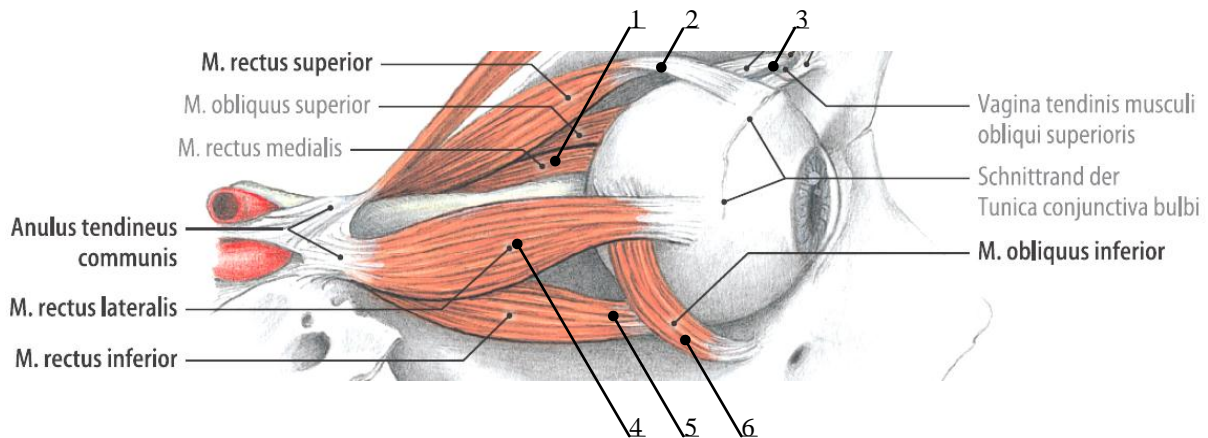


Bild 2.16 – Aufbau des menschlichen Auges (Bild aus [65]): 1 – innerer gerader Muskel, 2 – oberer gerader Muskel, 3 – oberer schräger Muskel, 4 – temporaler gerader Muskel, 5 – unterer gerader Muskel, 6 – unterer schräger Muskel

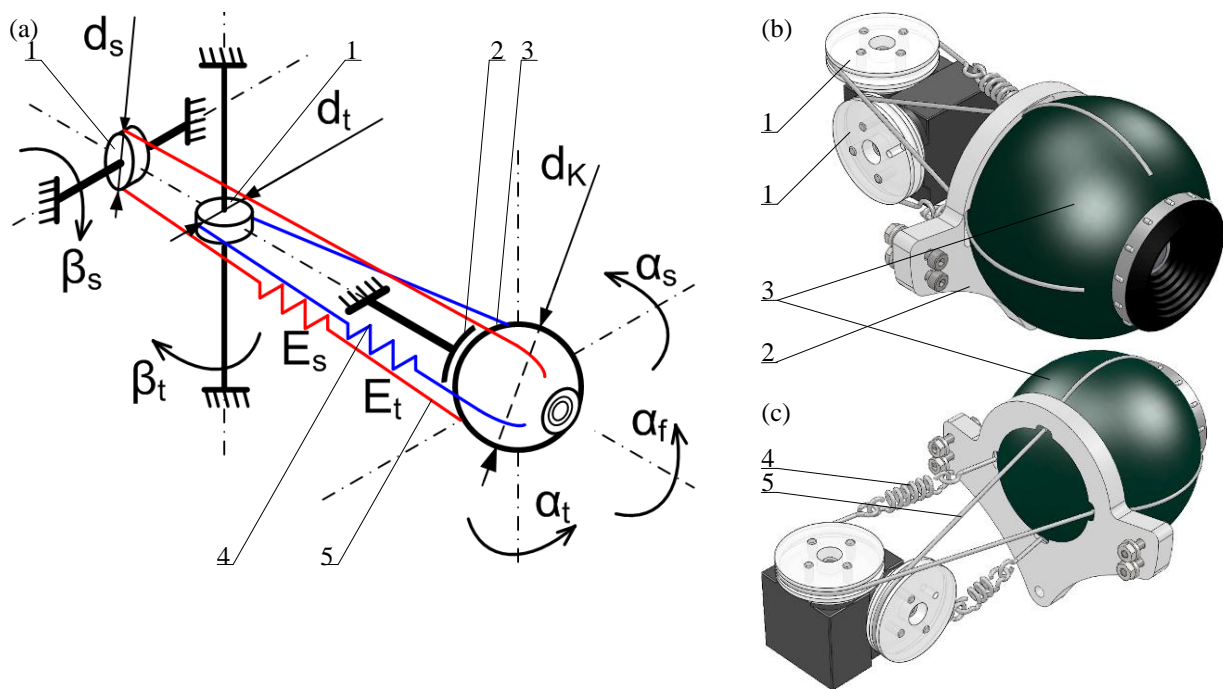


Bild 2.17 – Augengelenk: a – Antriebstruktur und Kinematik des Kugelgelenkes; b – CAD-Entwurf; c – Rückansicht (1 – Windenkopf, 2 – Gelenkpfanne, 3 – Gelenkkopf (Videokamera bzw. Auge), 4 – Vorspannfeder, 5 – hochfestes Polyethylen-Faser-Seil)

Die räumlichen Augendrehungen werden durch die Formeln 2.12 und 2.13 unter der Annahme, dass der Ablaufdurchmesser d_K in jeder beliebigen Augenposition stets konstant bleibt, ermittelt. Diese Annahme hat keinen Funktionseinfluss, weil die Augendrehungen bei eingesetzter Kopfsteuerung in kleinen Winkeln (bis $\pm 10^\circ$) verläuft, vereinfacht die Berechnung dennoch wesentlich.

$$\alpha_t = \frac{d_t}{d_K} \cdot \beta_t \quad (2.1)$$

$$\alpha_s = \frac{d_s}{d_K} \cdot \beta_s \quad (2.1)$$

2.4.3 Augenbrauen

Die Augenbrauen vervollständigen das Robotergesicht sowie die ausdrückenden Emotionen. Bild 2.18 veranschaulicht die Augenbrauekonstruktion. Die auf dem Stift befestigte Braue wird mittels einer, mit dem Servoantrieb durch ein Hebelwerk verbundenen, Zugstange betrieben. Die konstruktive Lösung wurde entsprechend der Integrationsanforderung der in den Roboterkopf ausgewählt.

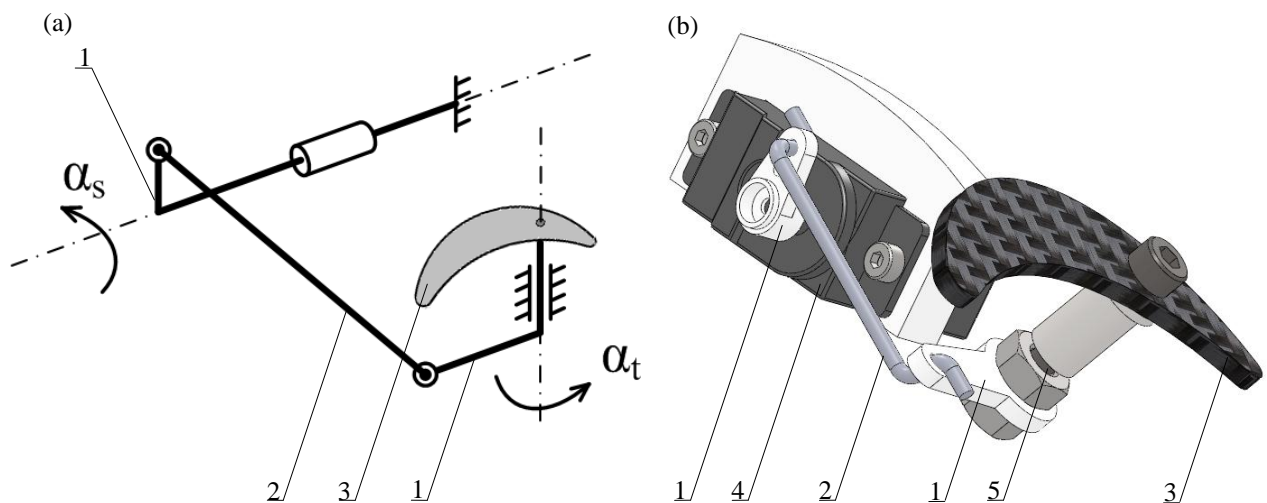


Bild 2.18 – Augenbraue: a – Kinematik; b – CAD-Entwurf (1 – Hebelwerk, 2 – Zugstange, 3 – Braue, 4 – Servoantrieb, 5 – Stift)

2.4.4 Lippenflexibilität

In einer emotionsgeladenen Interaktion nehmen die Lippen eine wichtige kommunikative Rolle ein. Auf einem vereinfachten Robotergesicht fallen außerdem die Lippen stark ins Auge und geben dadurch ein breites Feld an Emotionen wieder. Für eine qualitative Darstellung der Emotionen werden die Roboterlippen hochflexibel konstruiert. Die Lippen sind mit Hilfe von vier Getriebe in frontaler Ebene ($\alpha_{f1} - \alpha_{f4}$) betrieben, um eine Vielzahl der menschlichen Lippenbewegungen nachahmen. Bild 2.19 veranschaulicht die Lippenkonstruktion, die sich aus den zwei biegsamen roten Silikon-Röhrchen zusammensetzt, welche, durch die gebogenen Stäbe, direkt mit den Antrieben verbunden sind.

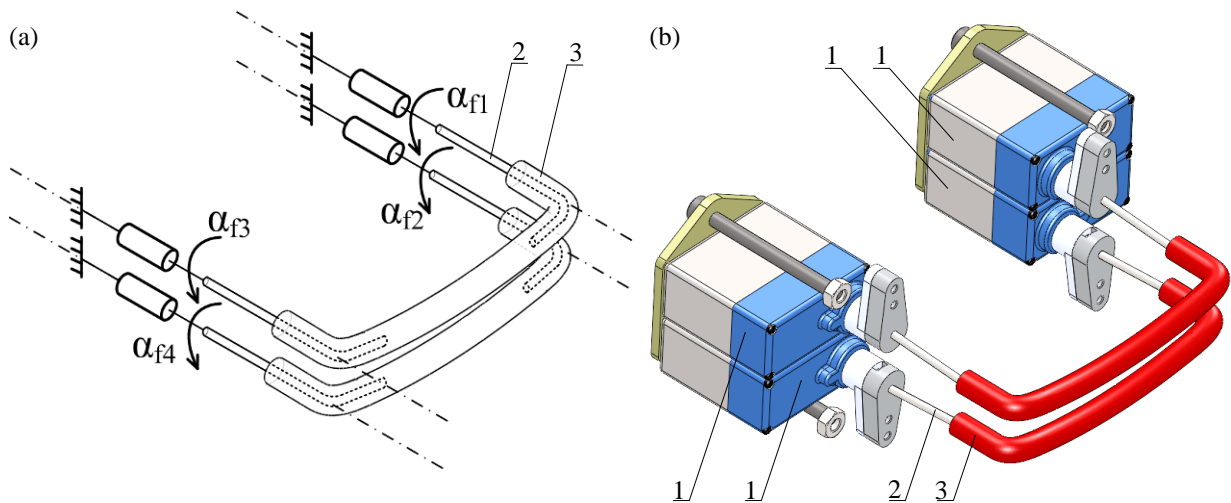


Bild 2.19 – Lippenkonstruktion: a – Lippenstruktur und Kinematik; b – CAD-Entwurf (1 – Servoantrieb, 2 – gekrümmter Metallstab, 3 – biegsames Silikon-Rohr bzw. Lippe)

Bei einer Motorrotation werden die flexiblen Silikon-Lippen durch die starren Stäbe bewegt. Durch die synchrone Ansteuerung der vier Stäbe, können die flexiblen Silikon-Lippen verschiedene emotionale Ausdrücke nachahmen. In der Tabelle 2.1 sind die Stabdrehungen bzw. Lippenbewegungen zur entsprechenden Grundemotion zugewiesen.

Tabelle 2.1 – Zuweisung der Lippenbewegungen zur Emotion

Stabdrehung Winkel: α_{f1} α_{f2} α_{f3} α_{f4}	Lippenbewegung	entsprechende Emotion
$\alpha_{f1} = 0$ $\alpha_{f2} = 0$ $\alpha_{f3} = 0$ $\alpha_{f4} = 0$		Neutralität
$\alpha_{f1} = +\alpha_1$ $\alpha_{f2} = +\alpha_1$ $\alpha_{f3} = -\alpha_1$ $\alpha_{f4} = -\alpha_1$		Freude
$\alpha_{f1} = -\alpha_2$ $\alpha_{f2} = -\alpha_2$ $\alpha_{f3} = +\alpha_2$ $\alpha_{f4} = +\alpha_2$		Trauer
$\alpha_{f1} = -\alpha_3$ $\alpha_{f2} = +\alpha_3$ $\alpha_{f3} = +\alpha_3$ $\alpha_{f4} = -\alpha_3$		Erstaunen
$\alpha_{f1} = +\alpha_4$ $\alpha_{f2} = +\alpha_4$ $\alpha_{f3} = +\alpha_4$ $\alpha_{f4} = +\alpha_4$		Unzufriedenheit

2.4.5 Augenlider

Das Äußere des Roboterkopfes wurde mit den Augenlidern komplettiert. Der Einsatz der Augenlidern ermöglicht die Darstellung einer Reihe von zusätzlichen Emotionsausdrücke, wie Zublinzeln, Einschlafen, Müdigkeit usw. Die Augenlider sind ein wichtiger Gesichtsteil, der eine Akzeptanz des Robotergesichtes verstärkt. Die Augenlider wurden aus Kunststoff gefertigt und sind in sagittaler Ebene beweglich. Bild 2.20 veranschaulicht diese Augenlidkonstruktion. Jedes Augenlid ist ähnlich, wie die Augenbrauen, mit Hilfe eines Servoantriebs inklusive durch das Hebelwerk befestigter Zugstange vorangetrieben.

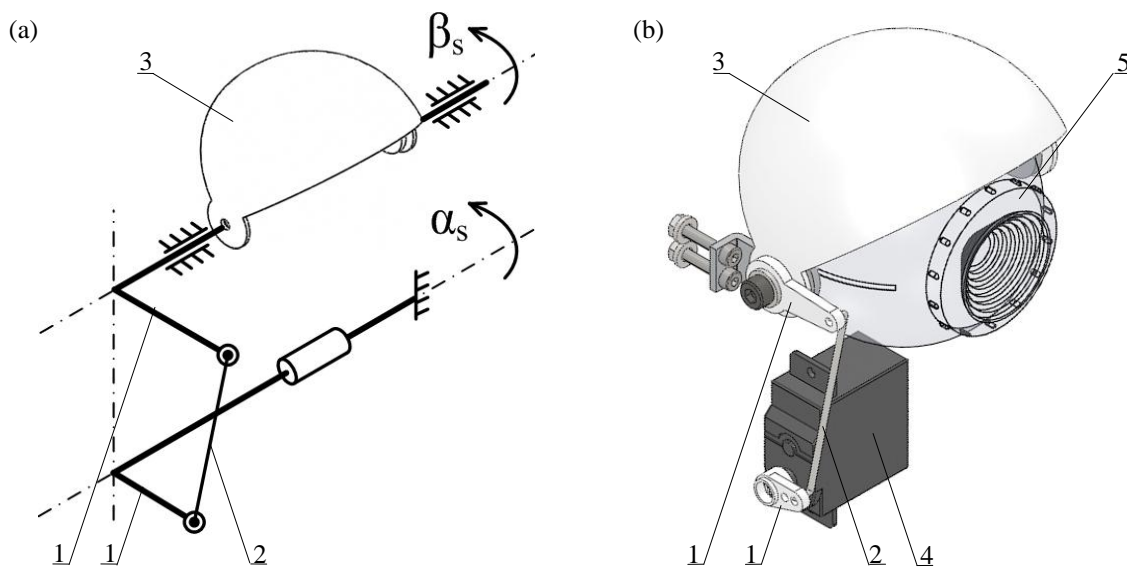


Bild 2.20 – Augenlid: a – Kinematik; b – CAD-Entwurf (1 – Hebelwerk, 2 – Zugstange, 3 – Lid, 4 – Servoantrieb, 5 – Auge bzw. Videokamera)

2.4.6 Gesamter Kopfaufbau

Die oben beschriebenen Kopfteile wurden auf einer gewichtsarmen Gerüstkonstruktion aus dem Glasfaser- und Polyoxymethylenkunststoff zusammengebaut. Die im Bild 2.14 aufgezeigte Gesichtsplatte, die aus dem Kunststoff mittels „rapid prototyping“ Verfahren gefertigt wurde, ist für die Außenansicht verantwortlich. Das äußere Erscheinungsbild des Roboters wurde, nach den bekannten Kriterien, möglichst freundlich, einfach und angenehm für die menschliche Wahrnehmung gestaltet. In Bild 2.14 versteckt die Gesichtsplatte den gesamten Kopfaufbau, deshalb veranschaulicht das Bild 2.21 die Details der Kopfkonstruktion ohne Gesichtsplatte.

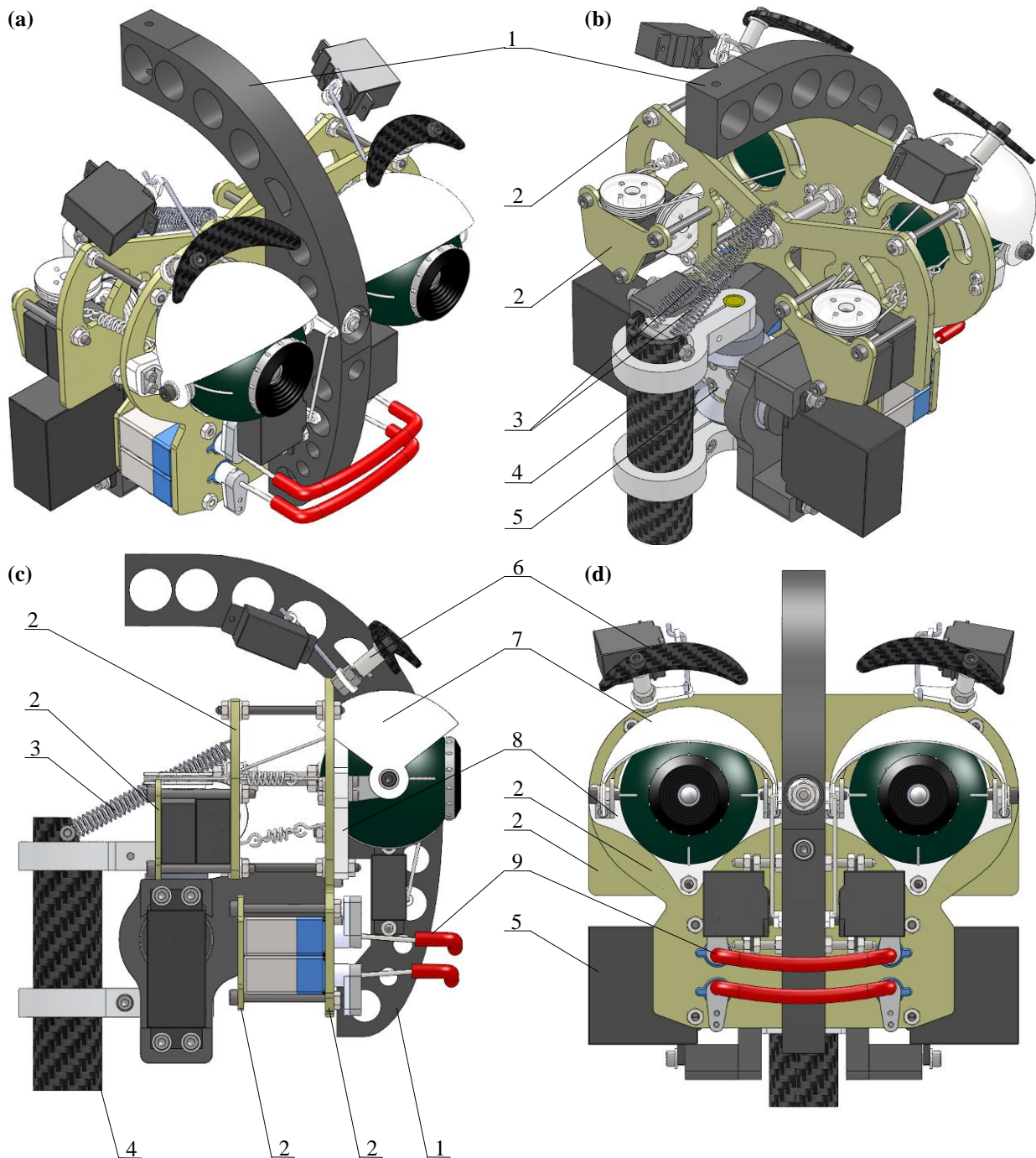


Bild 2.21 – Gesamter Kopfaufbau, CAD-Entwurf: a – dreidimensionale Frontansicht; b – dreidimensionale Rückansicht; c – seitliche Ansicht; d – Frontansicht (1 – Polyoxymethylen-Gerüst, 2 – Glasfaserkunststoff-Gerüst, 3 – zwei Ausgleichfedern, 4 – Hals, 5 – Halskreuzgelenk, 6 – Augenbrauen, 7 – Augenlider, 8 – Augegenlenk, 9 – Lippen)

Das Halsgelenk ist direkt mit dem Hals verbunden, wobei sich der Kopf in Bezug auf die Halsachse nach vorne neigt. Diese Methode bewirkt ein sogenanntes statisches Kippmoment M_k (Bild 2.22.a), das durch die Antriebe im Halsgelenk (Bild 2.15.b-2) kompensiert werden soll. Der Einsatz einer Ausgleichsfeder lässt die effiziente Kompensierung des Kippmomentes zu. Die Antriebe können dadurch entsprechend kleiner dimensioniert werden.

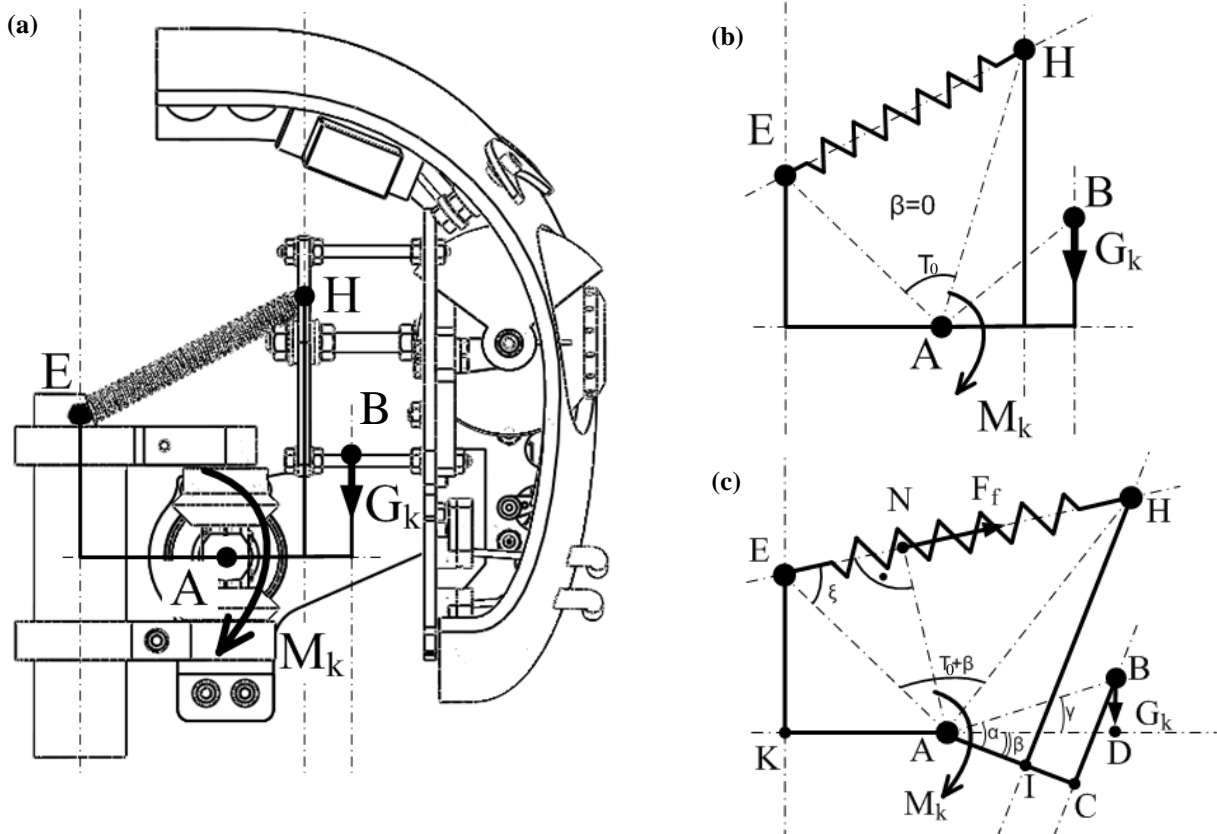


Bild 2.22 – Berechnung der Steife der Ausgleichfeder: a – seitlicher Ansicht mit den charakteristischen Punkten (A – Kopfdrehpunkt, B – Massenmittelpunkt des Kopfes, G_k – Kopfmasse, E und H – Federfixierungspunkte, M_k – Kippmoment); b – keine Kopfdrehung ($\beta = 0$) bzw. Ausgangsposition; c – der Kopf ist in der Sagittalebene um Winkel β geneigt

Eine Ausgleichsfedersteife wird mit Hilfe der Zeichnung in Bild 2.22.c bestimmt. Es wird nach einer Abhängigkeit der Ausgleichskraft F_f von der Federlänge EH gesucht. Zuerst wird die Abhängigkeit des Kippmomentes vom Winkel β aus ΔABC und ΔABD festgestellt:

$$M_k(\beta) = G_k \cdot g \cdot AD = G_k \cdot g \cdot \sqrt{AC^2 + BC^2} \cdot \cos\left(\arctg\left(\frac{BC}{AC}\right) - \beta\right) \quad (2.1)$$

Danach hängt die Federlänge EH vom Winkel β aus ΔEHA folgendermaßen ab:

$$EH(\beta) = \sqrt{EA^2 + BA^2 - 2 \cdot EA \cdot EB \cdot \cos(\tau_0 + \beta)} \quad , \quad (2.1)$$

$$\tau_0 = \pi - \arctg\left(\frac{EK}{AK}\right) - \arctg\left(\frac{HL}{AL}\right)$$

Die Ausgleichskraft F_f dahingegen hängt vom Winkel β aus ΔEHA ab:

$$F_f(\beta) = \frac{M_k}{AN} = \frac{M_k}{EA \cdot \frac{HA}{EH} \cdot \sin(\tau + \beta)} \quad (2.1)$$

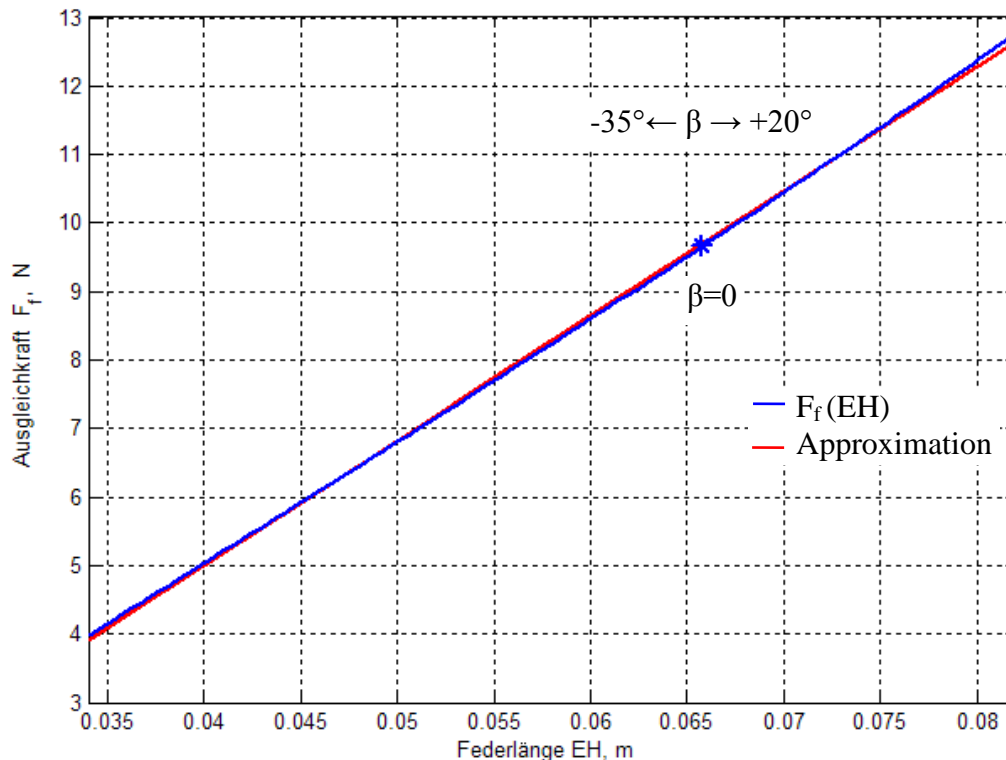


Bild 2.23 –Abhängigkeit der Ausgleichkraft F_f von der Federlänge

Hier wird die Steifigkeit der Ausgleichfeder für die Kopfdrehung in der sagittalen Ebene berechnet. Die Kopfdrehung in der transversalen Ebene wird durch eine kleine Änderung der Federlänge verursacht, deswegen wurde auf die transversale Ebene bei der Federberechnung verzichtet.

Für die zulässige Kopfdrehung β von -35° bis $+20^\circ$ Grad und die entsprechenden geometrischen Abmessungen, wurde die Abhängigkeit der Ausgleichskraft F_f von der Federlänge EH mit den Gleichungen 2.15 und 2.16 ermittelt und in Bild 2.23 (blau) veranschaulicht. Diese Kurve wurde durch eine Gerade mit Hilfe der Methode der kleinsten Quadrate approximiert und im Bild 2.23 rot hervorgehoben. Die Neigung ($k_f = 182\text{N/m}$) der Gerade entspricht der gewünschten Steifigkeit der Ausgleichsfeder. In der Ausgangsposition ($\beta=0$) soll die Feder mit einer Kraft von 9.7N bei einer Federlänge von 0.066m wirken, was einer freien Federlänge von 0.013m entspricht.

Entsprechend der berechneten Federeigenschaften werden zwei Ausgleichsfedern parallel (Bild 2.21.b-3) einmontiert, wobei die Feder eine Steife von 96N/m . Diese beide parallel leisten die gesamte Federsteifigkeit von 192N/m . Mit der ausreichenden Steifigkeit sind die eingesetzten Federn im unbelasteten Zustand $0,025\text{m}$ lang. Bei einer Länge von 0.066m üben beide Federn zusammen eine Kraft von 8.2N aus. Sie kompensieren größtenteils das Kippmoment und entlasten die Halsantriebe in der Ausgangsposition fast vollständig.

2.5 Technische Daten der Roboterkonstruktion

In diesem Kapitel werden die einzelnen technischen Details der Konstruktion des zweibeinigen Roboters ROTTO zusammengefasst.

Das Bild 2.24 veranschaulicht die gesamte kinematische Struktur des Roboters ROTTO. Die aus [48] übernommene kinematische Struktur des Unterkörpers bzw. der Roboterbeine und der Hüfte wird durch die kinematische Struktur des Oberkörpers vervollständigt. Die zwölf Freiheitsgrade in den Roboterbeinen sind für die Verwirklichung verschiedener Laufgänge verantwortlich. Die sieben Freiheitsgrade im Torso können sowohl für die Realisierung der Roboterbewegung als auch teilweise zur Vervollständigung der emotionalen Roboterfunktionalität eingesetzt werden. Mit den vierzehn Freiheitsgraden im Kopf wird eine hohe emotionale Flexibilität des Robotergesichtes erreicht.

Aus der vorgestellten Kinematik und den CAD-Entwürfen der Körper und Kopfteile des Roboters, ist es nachvollziehbar, dass jedes Gelenk in einem Bewegungs- bzw. Drehungsbereich begrenzt ist. Diese Begrenzungen sind in der Tabelle 2.2 zusammengefasst und werden bei der Erschließung der Steuerungsalgorithmen berücksichtigt.

Das Bild 2.25 veranschaulicht die äußeren Kopfabmessungen. Die Höhe des Kopfes (0,181m) beträgt 14,2% der gesamten Robotergröße (1,271m). Dieses Verhältnis entspricht der Proportion des menschlichen Körpers (im Alter von 12 Jahren – bis 14%, von 25 Jahren – bis 16%) [66]. Bezogen auf die Massenverteilung, beträgt das Gewicht des Roboterkopfes 1416g, was 8,8% des gesamten Robotergewichtes (16078g) ergibt. Im Vergleich zu der mittleren Massenverteilung des Menschen, die $1/14$ ($\approx 7,1\%$) des gesamten Körpergewichts beträgt [67], hat der Roboter, dank der grundlegenden Gewichtsoptimierung, bei der Entwicklung und vereinfachten Körperkonstruktion ein geringfügig anderes Gewichtsverhältnis. Die Konstruktion des Roboterkopfes bietet außerdem reichlich Potenzial zur weiteren Gewichtsoptimierung.

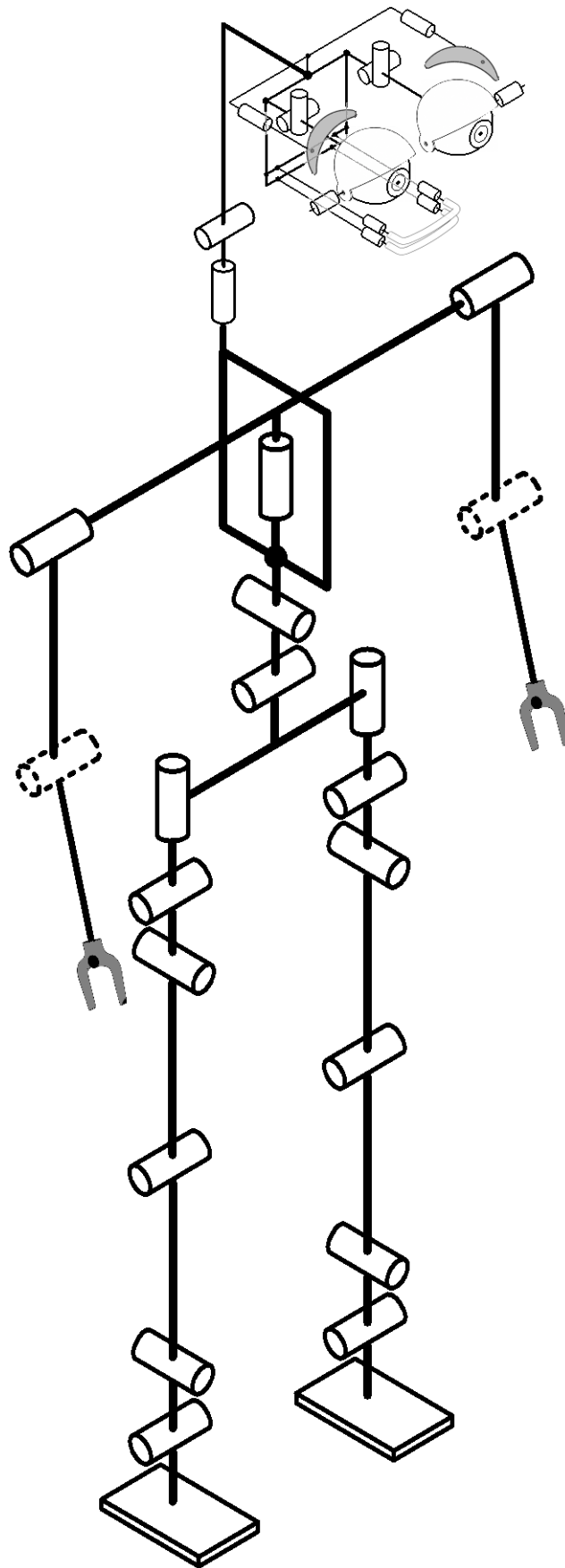


Bild 2.24 – Kinematische Struktur des Roboters ROTTO

Tabelle 2.2 – Konstruktive Begrenzung der Arbeitsbereiche für die Oberkörpergelenke

Gelenk	Ebene					
	Frontal, α_f		Sagittal, α_s		Transversal, α_t	
	min	max	min	max	min	max
Kreuzbeingelenk	-50° bei $\alpha_s = 0^\circ$	20° bei $\alpha_s = 0^\circ$	-35° bei $\alpha_f = 0^\circ$	35° bei $\alpha_f = 0^\circ$	–	–
Brustgelenk	–	–	–	–	-25°	25°
Schultergelenk	-37°	15°	–	–	–	–
Ellbogengelenk	0°	43°	–	–	–	–
Halskreuzgelenk	–	–	-20° bei $\alpha_t = 0^\circ$	35° bei $\alpha_t = 0^\circ$	-30° bei $\alpha_s = 0^\circ$	30° bei $\alpha_s = 0^\circ$
Augengelenk	-30° bei $\alpha_t = 0^\circ$	30° bei $\alpha_t = 0^\circ$	–	–	-30° bei $\alpha_f = 0^\circ$	30° bei $\alpha_f = 0^\circ$
Augenbrauen	–	–	–	–	$\beta_t = -15^\circ$	$\beta_t = 20^\circ$
Lippen	-20°	20°	–	–	–	–
Augenlider	–	–	-10°	50°	–	–

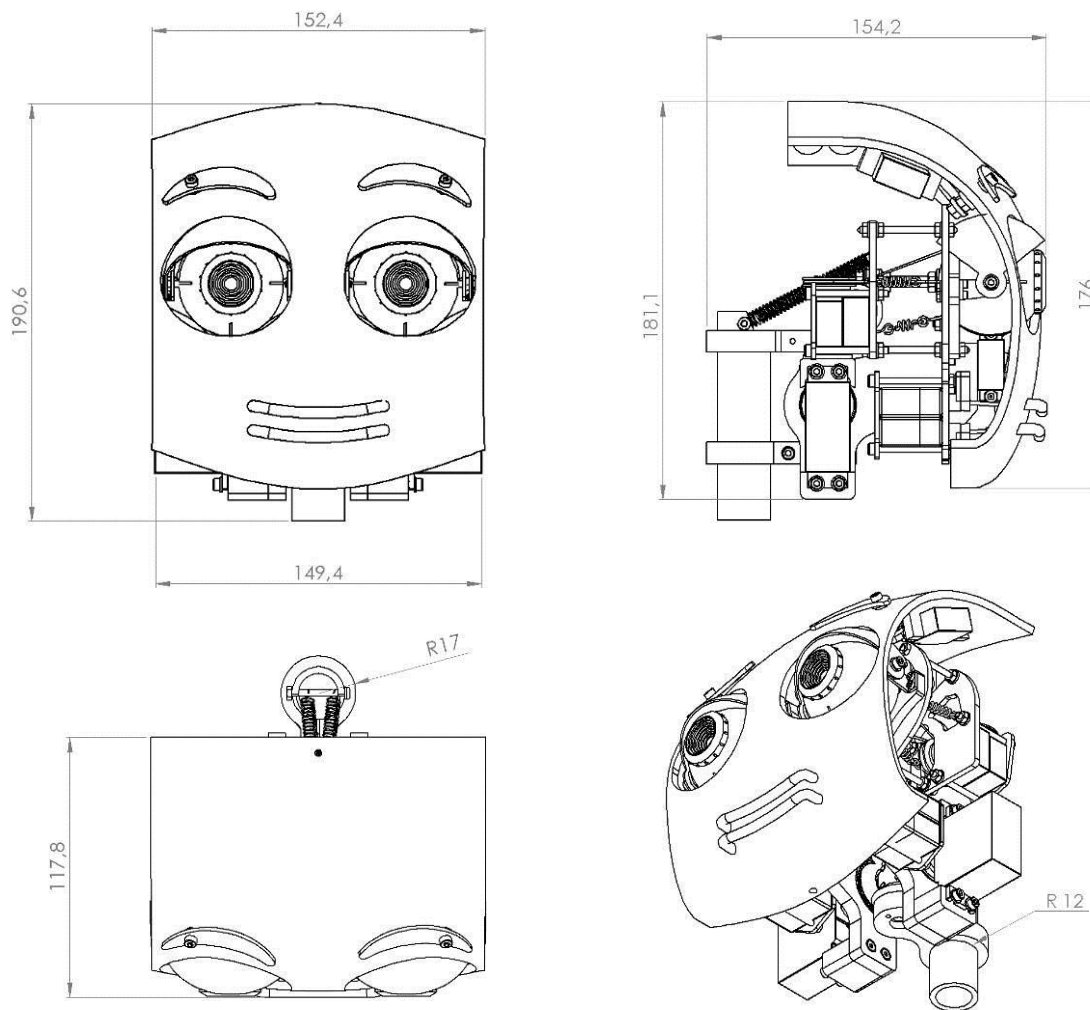


Bild 2.25 – Abmessungen des Kopfes des Roboters ROTTO

2.6 Zusammenfassung

In diesem Kapitel werden zunächst die Grundlagen zum Entwurf des Roboterkopfes und des Oberkörpers vorgestellt. Die diskutierten Anforderungen an das Äußere und die Beweglichkeit des Roboters wurden bei der Entwicklung des hier beschriebenen zweibeinigen, mobilen Roboters ROTTO angewendet. Bei dem Entwurf des mechatronischen Systems werden moderne Rechen- und Softwaretools in allen Bereichen der Systementwicklung stark miteinbezogen.

Im Laufe des Entwicklungsprozesses werden für die einzelnen Roboterkomponenten sowohl konstruktive Anforderungen, wie beispielweise die einfache Fertigung, eine hohe Modularität und Erweiterbarkeit, als auch funktionelle Anforderungen, wie Energie- und Materialeffizienz oder Maximierung des Arbeitsbereiches, festgelegt. Diese Anforderungen konnten auch eingehalten werden.

Diese Faktoren ermöglichen somit innerhalb kürzester Zeit die Entwicklung eines Roboters bis zu einem Laborprototyp voranzubringen und eine Reihe der Versuche auf dem Gebiet der Roboter-Mensch-Interaktion durchzuführen.

Aus funktioneller Sicht ist zu erwähnen, dass die entworfene Roboterkonstruktion sich durch einfache, aber dennoch nicht nur für Lokomotionsaufgaben sondern auch für Aufgaben der emotionalen Interaktion ausreichende, Kinematik mit optimalen Verhältnissen der Körperteile und größeren Arbeitsbereichen kennzeichnen lässt. Die anthropomorphe Platzierung der Antriebseinheiten in die Beine und den Roboterkörper wurde mit der Kompaktbauweise des Kopfes, mit den raumsparenden und leistungsfähigen Servoantrieb, vervollständigt. Die im Roboterkörper eingesetzten elektrischen Antriebe sind sehr effizient in Bezug auf Kosten, Energieverbrauch und Leistungsdichte ausgelegt. Die integrierte Sensorik erfasst alle notwendigen Informationen, wie Körperkonfiguration, Orientierung im Gravitationsfeld, mechanische Kontaktkräfte, sowie stereoskopische Videoinformationen, die für die Erforschung der gestellten Aufgaben notwendig sind.

Hiemit konnte verdeutlicht werden, dass der zweibeinige, anthropomorphe, interaktive Roboter ROTTO, dank des iterativen und systematischen Entwicklungsansatzes, in allen Bereichen optimal entwickelt wurde. Der Entwurf des Roboters orientiert sich an den anthropomorphen Verhältnissen und ist in der Lage Aufgaben, wie intuitive Roboterinteraktionen mit Menschen oder das hochdynamische Gehen und Laufen, durchzuführen.

Kapitel 3

Steuerungssystem des Roboters ROTTO

Mobile humanoide Roboter stellen komplexe mechatronische Systeme mit mehreren aktiven Gelenken dar und werden meistens über ein strukturiertes, eingebettetes System gesteuert. Die Hauptaufgabe solcher Steuerungssysteme ist eine einheitliche Organisation der Informations- und Energieflüsse im gesamten Robotersystem (Bild 3.1, links). Unter dem Begriff mechatronisches System wird in den mobilen Robotern die Vereinigung von Mechanik, Elektronik (Hardware) und Informatik (Software) verstanden (Bild 3.1, rechts).

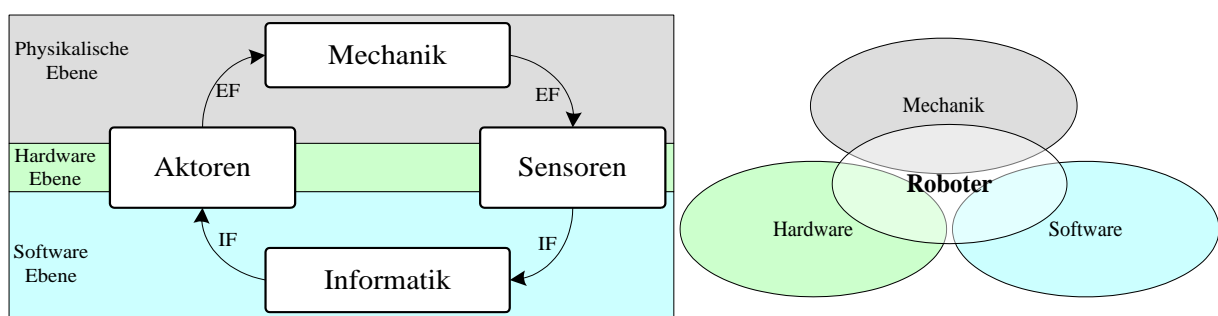


Bild 3.1 – Informations- und Energieflüsse (IF und EF) in einem mechatronischen System

Die direkten Aufgaben der Steuerungssysteme sind die Erfassung und Anpassung der Sensordaten, Digitalisierung und Vorbereitung dieser Signale für die Steuerungsalgorithmen sowie die Übertragung der berechneten Steuerungssignale zu den Aktoren. Diese Aufgaben koordiniert und in Echtzeit durchzuführen ist eine der besonderen Anforderungen an diese Systeme.

In den Phasen der Entwicklung, Implementierung, Tests und Inbetriebnahme eines komplexen technischen Systems sind die entsprechenden Entwicklungswerkzeuge von großer Bedeutung. Dabei wird von den Entwicklungswerkzeugen und vom Steuerungssystem die Fähigkeit sowohl zur hybriden Simulation (auch Target-in-the-Loop (TiL) Simulation genannt) als auch zum Rapid Control Prototyping (RCP) gefordert.

Diese Phasen, sowie die Entwicklung und der Aufbau der Hardware- und Softwarearchitektur eines echtzeitfähigen flexiblen Steuerungssystems für den humanoiden Roboter ROTTO mit der Fähigkeit zur hybriden Simulation und RCP werden in diesem Kapitel erläutert.

3.1 Struktur und Komponenten des Steuerungssystems

Das Ziel, eine optimale Struktur für das Steuerungssystem von zweibeinigen, humanoiden mobilen Robotern zu erarbeiten, setzt die Lösung verschiedener Teilaufgaben voraus. Eine der Fragen bezieht sich auf die schnelle und effiziente Entwicklung des Steuerungssystems, das dem strukturellen Aufbau der Software und der Steuerungshardware angepasst ist. Die nachfolgenden Diskussionen um die Komponenten des Steuerungssystems berücksichtigen das V-Modell in der Entwicklung des mobilen Roboters.

Die Entwicklung des Roboters ROTTO als eines mechatronischen Systems wird angelehnt an ein V-Modell [79] (die VDI Richtlinie 2206 - „Entwicklungsmethodik für mechatronische Systeme“) betrachtet, das in [48] für die Entwicklung von mobilen Robotern erweitert ist. Ein prinzipielles Vorgehensmodell der VDI Richtlinie 2206 besteht aus folgenden Elementen:

1. Mikrozyklus, der die Problemlösungszyklen beinhaltet:

- Lösungssuche durch Analyse und Synthese der Teilschritte
- Vergleich von Ist- und Sollzustand
- Bewertung, Entscheidung
- Planung

2. Makrozyklus in Form eines V-Modells

- Logische Abfolge von Teilschritten
- Anforderungen
- Systementwurf
- Domänenspezifischer Entwurf
- Systemintegration
- Eigenschaftsabsicherung (Verifikation, Validierung)
- Modellbildung (flankierend)
- Produkte: Laborprototyp, Funktionsprototyp, vorserienreife Roboter

3. Prozessbausteine für wiederkehrende Arbeitsschritte

- Wiederkehrende Prozessbausteine
- Systementwurf, Modellbildung, Bauelemente-Entwurf, Integration usw.

Die wichtigsten Entwurfsschritte für die Entwicklung von mobilen Robotern veranschaulicht das aus [48] abgeleitete und erweiterte Entwicklungsdiagramm im Bild 3.2. Der Reifegrad des Roboters nimmt von einem zum nächsten Entwicklungsschritt stufenweise zu. Die einzelnen Phasen bestehen jedoch aus vielen Iterationen, die im gegebenen V-Modell nicht eingezeichnet sind.

Im V-Modell werden drei bedeutende Reifegrade der Entwicklung der mobilen Robotern betont: Laborprototyp, Funktionsprototyp und vorserienreifer Roboter. Um den ersten Reifegrad zu erreichen, erfolgt der Systementwurf bis zum Laborprototyp, in dem erste Wirkprinzipien, Lösungselemente und Grobdimensionierungen eingesetzt und erste Funktionsuntersuchungen durchgeführt werden. Der nächste Reifegrad des Roboters wird durch die Systemintegration bis zum Funktionsprototyp erreicht, in der die Weiterentwicklung, Feindimensionierung und Integration der verteilten Komponenten, Leistungsmessungen, Standard-Schnittstellen vollzogen wird. Zur Erreichung des Endreifegrades werden die Systemtests bis zum fertigen, vorserienreifen Roboter durchgeführt, in dem die Fertigungstechnik, Standardisierungen, weitere,

modulare Integrationsstufen, Kapselung und die Feldtests berücksichtigt sind. Bei der für Forschungszwecke ausgerichteten Entwicklung eines mobilen Roboters ist der Zustand des Laborprototyps in der Regel ausreichend, um die gewünschte Theorie bzw. Ansatz zu prüfen.

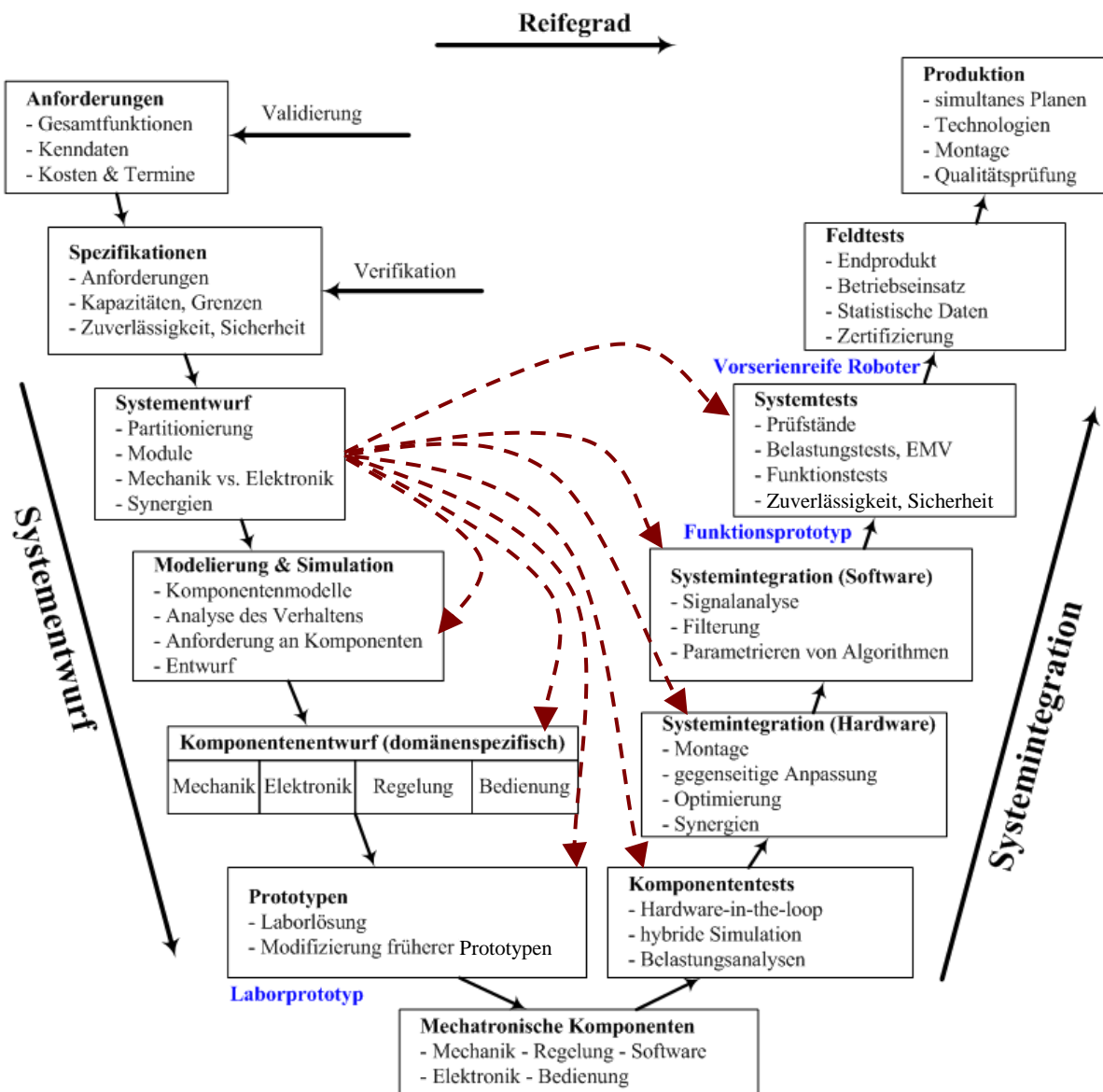


Bild 3.2 – Entwurfsschritte (V-Modell) für einen mobilen Roboter als mechatronisches System

Bei der Entwicklung von mobilen Robotern als komplexe mechatronische Systeme werden zuerst die Anforderungen und Spezifikationen definiert. Danach, in der Phase des Systementwurfs, werden die mechatronischen Komponenten – Robotermechanik und Steuerungssystem erarbeitet. Das Steuerungssystem beinhaltet bestimmte softwaremäßige Algorithmen, die auf einer eingebetteten Technologieplattform (Hardware) ausgeführt werden. Der Einsatz der effizienten Werkzeugkette, die die Entwicklung, Modellierung, hybride Simulation und die Verifikation der Softwarealgorithmen erlauben und somit die einfache, schnelle und effiziente Anpassung an die Technologieplattform zur Verfügung stellen, führt zur drastischen Verkürzung oder sogar zur automatischen Durchführung der einzelnen Entwurfsphasen (im Bild 3.2 mit roten Strichpfeilen dargestellt). Die nachfolgenden Unterkapitel widmen sich den

für mobilen Roboter ROTTO entwickelten Softwarewerkzeugen sowie der Hardware-Technologieplattform, die die effiziente Entwicklung und schnelle Erarbeitung der Entwurfsphasen ermöglichen.

3.2 Aspekte der Softwareentwicklung für mobile Roboter

Um eine effiziente und schnelle Entwicklung der komplexen Software-Algorithmen für einen komplexen humanoiden Roboter zu ermöglichen, sind folgende Teilaspekte der Softwareentwicklung zu berücksichtigen:

- 1 Die Notwendigkeit der einfachen und benutzerfreundlichen Werkzeugkette (Entwicklungsumgebung) ist für die Softwareentwicklung zu berücksichtigen.
- 2 Die Werkzeugkette soll den Entwicklungsprozess von der ersten Idee (Bild 3.3.a-b) bis zur Erarbeitung der Algorithmenmodule möglichst übersichtlich darstellen.
- 3 Neben der übersichtlichen Darstellung ist eine Möglichkeit der mathematischen Modellierung erforderlich, um die Algorithmen in der Simulation zu verifizieren und überprüfen (Bild 3.3.b-c).
- 4 Für erweiterte und tiefere Tests bzw. Verifikationen soll die Werkzeugkette hardwaregebundene, hybride Simulationen, wie beispielsweise Hardware-in-the-Loop (HiL), Software-in-the-Loop (SiL) bzw. die Teile der Rapid Control Prototyping (RCP) (Bild 3.3.d), ermöglichen.
- 5 Der Begriff Rapid Control Prototyping beschreibt hierbei ein Verfahren, bei dem die in der Werkzeugkette entwickelten Algorithmen in den hardwarespezifischen Code „per Mausklick“ umgewandelt und auf die eingebettete Roboterhardware heruntergeladen wird (Bild 3.3.e).
- 6 Die Algorithmen sollen in einem eingebetteten System bzw. einer Technologieplattform mit dem streng definierten Rechenzyklus bzw. in Echtzeit berechnet werden (Bild 3.3.f).
- 7 Im Unterschied zu modernen, leistungsfähigen Desktop-PCs ist die Rechenleistung der Ziel-Prozessoren eines eingebetteten Systems stark begrenzt. Deswegen soll der Machinencode während der Umwandlung für die spezifische Hardware stark optimiert werden (Bild 3.3.e). Dazu wurden moderne Codeumwandler (Compiler) zum Einsatz ausgewählt.
- 8 Während der Entwicklung der Algorithmen ist die Möglichkeit der einfachen Fehlerbeseitigung (Debugging) in allen Entwicklungsebenen von großer Bedeutung. Sowohl die Fehlerbeseitigung im Programmcode als auch die eines Prozesses sind signifikant für den weiteren Entwicklungsverlauf (Bild 3.3.g).
- 9 In den mobilen Robotern werden die Algorithmen oft auf mehreren Ziel-Prozessoren verteilt (verteilt Rechnen, eng: distributed computing), um damit die Rechenleistung des gesamten Rechensystems zu erhöhen bzw. auszubalancieren und zeitkritische Algorithmen mit sehr kleinem Rechenzyklen (bis ins Mikrosekunden-Bereich) zu berechnen. Darüber hinaus existieren die oben genannten Echtzeitanforderungen, die den Einsatz der echtzeitfähigen Datenübertragungsnetzwerke (Real-Time Kommunikation) erfordern (Bild 3.3.h).

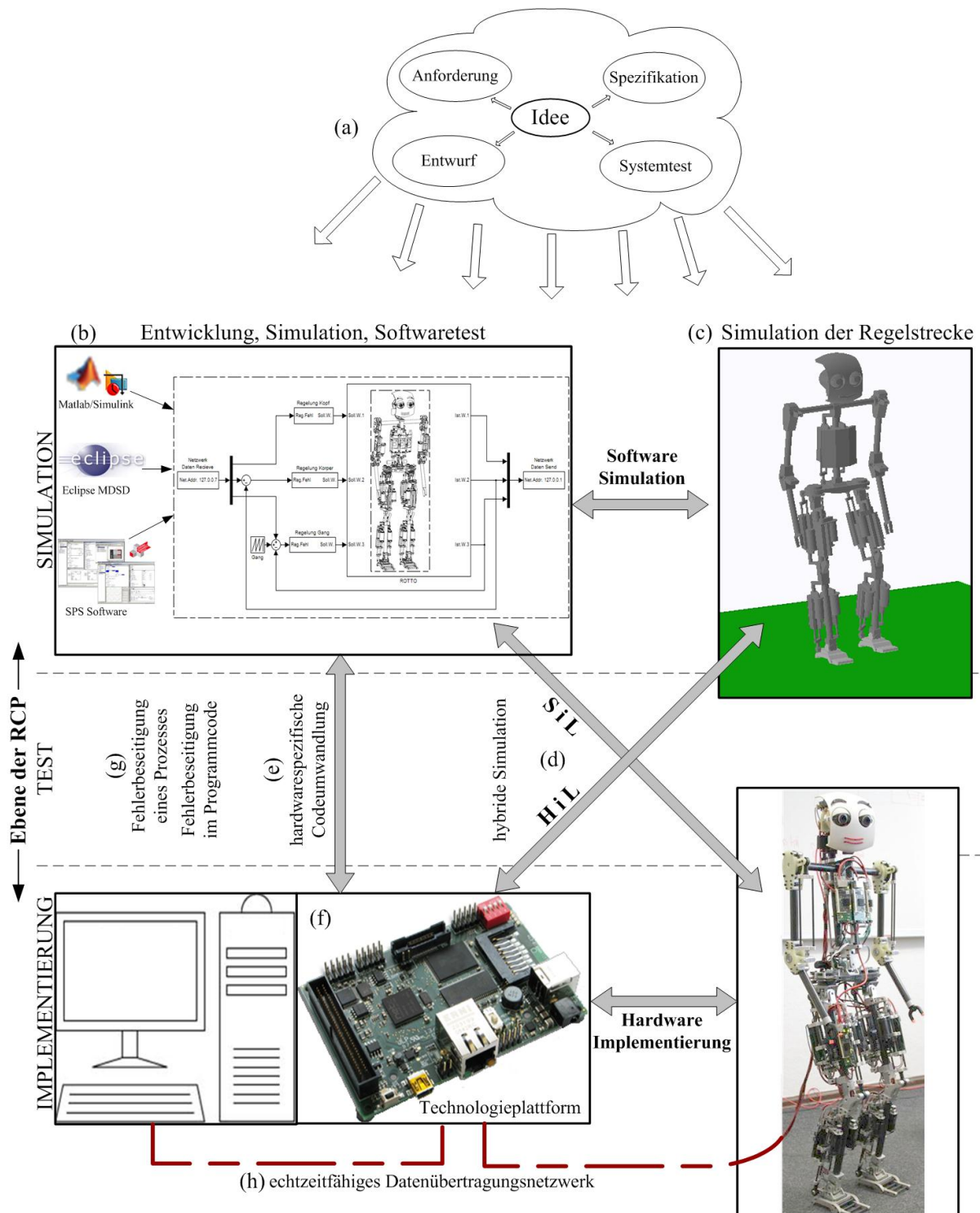


Bild 3.3 – Werkzeugkette zur Entwicklung der komplexen Algorithmen für ein mechatrisches System

Während der Entwicklung der mobilen Schreitroboter in [48] wurden die benannten Anforderungen fast erreicht. Es wurde eine Technologieplattform NetX500 [80] in Verbindung mit

dem xPC-Target [39] eingesetzt. Dabei ist beim NetX500 die Unterstützung vieler Datenübertragungsnetzwerke, einschließlich echtzeitfähiger Netzwerke, von großem Vorteil. Die einzige Schwierigkeit für den breiten Einsatz dieser Technologieplattform im mechatronischen Aufgabenbereich liegt im Fehlen der hardware-spezifischen Codeumwandlung für den Protocol Execution Controller (xPEC – Communication and User Task channel). Es gibt keine Unterstützung für höhere Programmiersprachen, deshalb ist der Programmwurf nur mit der hardware-spezifischen Assemblersprache möglich.

Dieser Mangel wurde in der neuen Technologieplattform NetX10 [81] behoben. Jene Technologieplattform bietet einen neuen Peripheral Interface Controller (xPIC) an, der vom Anwender frei programmierbar ist. Der Autor dieser Arbeit hat an der Entwicklung der Controllerhardware, des ALU-Befehlssatzes und der Software-Werkzeugkette teilgenommen. Der xPIC wurde für komplizierte Automatisierungs-, Steuerungs- und Regelungszwecke entworfen. Die oben erwähnten neun Anforderungen an die Entwicklung der mechatronischen Systeme wurden in der Technologieplattform NetX10/xPIC realisiert und während der Entwicklung des Steuerungssystems des Roboters ROTTO für die Mensch-Roboter-Interaktion eingesetzt.

3.2.1 Graphische Softwareentwicklung

In der Beschreibung zum Bild 3.3 wurde darauf hingewiesen, dass die schnelle Realisierung einer Idee zum Endprodukt oder Forschungsergebnis von großer Bedeutung ist. Damit sollen Zeit und Aufwandskosten gespart werden. Die Untersuchung und Implementierung sollen so ähnlich einfach sein, wie eine Idee auf dem Papierblatt zu zeichnen. Hier sind die Werkzeuge zur graphischen Softwareentwicklung mit den RCP-Eigenschaften einzusetzen.

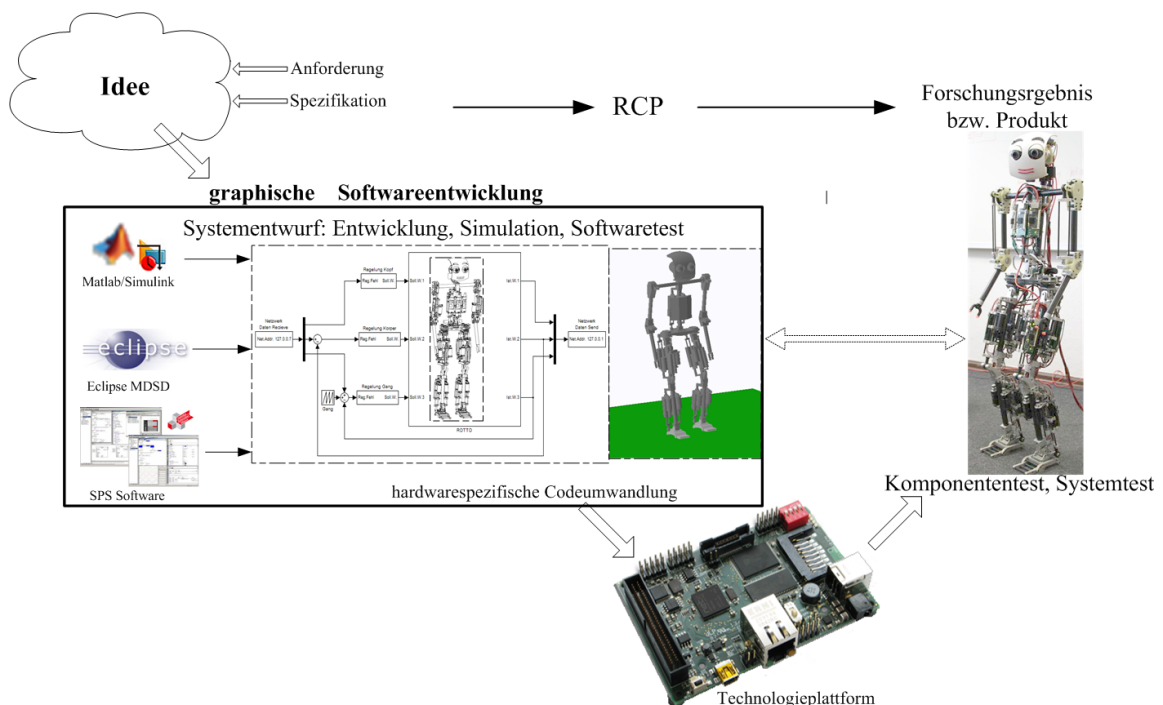


Bild 3.4 – Von einer Idee zum Produkt durch graphische Softwareentwicklung

Das Bild 3.4 veranschaulicht den gesamten Prozess des Entwurfes von ROTTO, in dem die graphische Softwareentwicklung in den Mittelpunkt des Entwurfsprozesses gerückt wird.

Dieser Ansatz erlaubt eine erhebliche Verkürzung der Entwicklungszeit auf dem Weg zum fertigen Roboterprototyp.

3.2.1.1 RCP-fähige Software

Der Begriff „Rapid Control Prototyping“ (RCP) bedeutet eine rechnergestützte Entwurfsmethode zur Steuerungs- und Regelungsentwicklung. Die RCP-fähige Software ermöglicht die Erfassung der Entwurfsmethode in die folgenden typischen Entwurfschritte [82]:

- dynamische Beschreibung des zu automatisierenden Objektes,
- dessen Modellbildung,
- die Steuerungs- und Regelungsentwürfe im erstellten Modell,
- die Umsetzung des Steuerungs- und Regelungsentwürfe auf der Technologieplattform und
- die Erprobung und der Vergleich der Lösung einer reinen Simulation mit der am realen Objekt.

Die gegenwärtige RCP-Software vereinigt diese fünf Schritte in einer Entwicklungsumgebung. Die Modellbildung und der Steuerungs- und Regelungsentwurf erfolgen sowohl in der analogen als auch in der diskreten Form, was die vollständige Abbildung (Simulation) der Prozesse in einer digitalen Technologieplattform erlaubt. Die Erprobung und der Vergleich der Simulations- mit den realen Ergebnissen erfolgt mit bequemen graphischen Werkzeugen, die die gewünschten Daten übersichtlich als Abhängigkeiten, Verläufe, Diagramme, Tabellen u.ä. darstellen können. Nachfolgend werden die verbreiteten RCP-Software Werkzeuge vorgestellt und die Auswahl der Werkzeuge für die Entwicklung des Roboters ROTTO begründet.

MATLAB/Simulink von MathWorks [36] ist ein sehr umfangreiches Softwarepaket, das durch zahlreiche umfangreiche Toolboxen (Werkzeuge) für die Simulation, für die numerische Mathematik, Systemidentifikation, Reglerentwurf und das RCP geeignet ist. *Simulink* erlaubt die hierarchische Modellierung mit Hilfe grafischer Blöcke. Dabei stellt *Simulink* einen Grundsatz an diskreten und kontinuierlichen Schaltungsblöcken zur Verfügung. Die zusätzliche Integration des eigenen Codes ist mittels M-Funktionen (in *Matlab*-Sprache) und S-Funktionen (C/C++/Ada/Fortran) möglich. Der Datenaustausch zwischen den Blöcken wird grafisch über Verbindungslinien realisiert. Auf diese Weise wird das Modell eines Systems erstellt. Das kann weiter innerhalb *Simulink* mit verschiedenen Lösungsverfahren (sog. "Solver") simuliert werden. Es werden alle Zahltypen (Integer-, Gleit- und Festkomma) und auch skalierte Festkommatypen in der Simulation und Codegenerierung unterstützt. Aus dem erstellten und simulierten Modell kann der Programmcode mit Hilfe von der *Real-Time-Workshop*-Toolbox erzeugt werden. Die Codegenerierung ist mit Konfigurationsdateien für verschiedene Zielsprachen anpassbar (*target language compiler*). Mit der *xPC-Toolbox* lässt sich der vom *Simulink*-Modell generierte Code auf dem Durchschnitts-PC als Echtzeitplattform austesten. Dieses Verfahren ist an einen beliebigen Zielprozessor (Technologieplattform) anpassbar.

Ein anderes Softwareprodukt dieser Reihe ist der *Advanced Simulation and Control Engineering Tool* (ASCET) von ETAS Group [83]. Dies ist ein kommerzielles und weit verbreitetes Werkzeug im Automobilbereich, das für die Simulation, Modellierung, RCP und Seriercodegenerierung für die Steuergeräte im Fahrzeug eingesetzt wird. Solche Steuergeräte sind verantwortlich für Funktionen wie Steuerung oder Regelung der Verbrennungsmotoren, Hybrid-

antriebe, Automatikgetriebe, Antiblockiersysteme, Fahrstabilitätssysteme u.ä. Eine Modelltransformation ist von und nach *MATLAB/Simulink* ebenfalls möglich.

Freie Software *Scilab/Scicos(Xcos)* von INRIA [84] und Scilab-Konsortium verfügt ebenfalls über sehr umfangreiches Softwarepaket für die numerische Mathematik mit einem ähnlichen Konzept und einer ähnlichen Syntax wie *MATLAB* und ist für die Simulation, die Systemidentifikation und das RCP geeignet [85]. Das Programmpaket *Xcos* (ehemals *Scicos*) ist an die *Simulink* angelehnt und dient der graphischen Simulation und Modellierung komplexer dynamischer Systeme. Es ist ebenfalls erweiterungsfähig, durch definierte Schnittstellen oder durch frei programmierbare *Toolboxen*. Eine Modelltransformation ist von *MATLAB/Simulink* nach *Scilab* mit Hilfe eines Konverters möglich.

CAMEL-View TestRig von iXtronics [86] ist ein kommerzielles Softwareprodukt, das die objektorientierte Modellbildung mechatronischer Systeme durch grafische Blöcke ermöglicht. Für die umfangreiche Modellbildung stehen Grundblöcke aus der Regelungstechnik, der Mehrkörpersystemdynamik, der Hydraulik sowie der Hardwareanbindung zur Verfügung. Der eigene C-Code ist ebenfalls in das System integrierbar. Die Beschreibung und Darstellung der Modellstruktur erfolgt mit Hilfe von gerichteten und ungerichteten Verbindungen. Neben den Signalflüssen (Eingangs-/Ausgangskopplungen) stehen physikalische Kopplungen (wie mechanische oder hydraulische Kopplungen) zur Verfügung. Es gibt eine konfigurierbare Experimentierumgebung, mit deren Hilfe die Simulationsmodelle auf den Hardware-Targets ohne zusätzliche Softwarewerkzeuge für die HiL-Anwendungen eingesetzt werden können. Die Generierung von S-Functions für *Simulink* und die Generierung von ANSI C-Code zur Simulation wird im eigenständigen Modus unterstützt.

Das kommerzielle grafische Programmiersystem *LabVIEW* von *National Instruments* [87] bietet Einfachheit in der Programmierung der parallelen Abläufe. Die Programmierung erfolgt mit der graphischen G-Programmiersprache nach dem Signalfluss-Modell. *LabVIEW* eignet sich besonders gut zur Datenerfassung und Datenverarbeitung. *LabVIEW*-Programme beinhalten zwei Komponenten: das Vornpanel enthält die Benutzerschnittstelle und das Blockdiagramm den graphischen Programmcode. Ein spezieller Compiler wandelt die graphische G-Programmiersprache in die ausführbare Anwendung um. Dadurch ist die mit anderen Hochsprachen vergleichbare Leistungsfähigkeit erreichbar.

Es existiert eine Reihe von Echtzeitsimulationsplattformen, die sich an genannten Softwareprodukten anbinden lassen. *Modelica* [88] ist eine objektorientierte Beschreibungssprache für physikalische Simulationen. Die Entwicklungsumgebungen *Dymola*, *MathModelica* und *SimulationX* [89] basieren auf der *Modelica*. Außerdem wird *Modelica* auch in Verbindung mit dem *Simulink* für RCP eingesetzt. *dSPACE* [90] bietet integrierte Hardware und Software-Lösungen für die Anbindung von *MATLAB* an verschiedene Versuchsstände an. *SCALE-RT* [91] ist eine unkommerzielle Simulationsplattform in Echtzeit für Linux-basierten PC welche sich an *MATLAB/Simulink*, *Scilab/Scicos* und *SimulationX* anbinden lässt.

Alle oben beschriebene RCP-Software Werkzeuge umfassen die gewünschten Entwurfschritte für ein mechatronisches System. Für die Entwicklung des Roboters *ROTTA* wurde sich mit der nachfolgenden Begründung für die *MATLAB/Simulink* entschieden. Die *Robots-Lab* Gruppe kann auf eine mehrjährige Erfahrung sowohl mit der hybriden Simulation mit dieser Software als auch mit der Anpassung der Modell-Codeumwandlung an die Technologieplattform *NetX500* [92] (Vorgänger von *NetX10*) zurückgreifen.

3.2.1.2 Automatisierungsstandard für verteiltes Rechnen

Die Steuerungs- und Regelungsaufgaben in komplexen mechatronischen Systemen (wie beispielsweise mobile Roboter) werden oft auf mehrere Ziel-Prozessoren aufgeteilt. Dadurch wird eine funktionelle Verteilung und Skalierung der Rechenleistung für die zeitkritischen oder rechenaufwendigen Aufgabenerreicht. Eine manuelle Verteilung der Rechenaufgaben auf mehrere Hardwarekomponenten bringt zusätzliche Fehlerquellen mit und ist mit einem zusätzlichem Zeitaufwand verbunden. Aus diesem Grund werden zum Aufbau des verteilten Rechnensystems die vorher bekannten Techniken bzw. Softwaremethoden eingesetzt. In der klassifizierten Übersicht in [93] sind mehrere komponentenbasierte Softwarepakete zum parallelen oder verteilten Rechnen aufgeführt.

Wenn ein mobiler Roboter aus dem Blickwinkel der komplexen Automatisierungsplattform betrachtet wird, sind die existierenden industriellen Standards zu evaluieren. Einer davon ist der IEC61499 [94]. Dieser Standard definiert eine offene Architektur für das verteilte Rechnen und die Automatisierung und zeigt ein allgemeines Modell für die verteilten Systeme. Dieses Modell stellt die Kommunikationsnetzwerke als Umgebung für die eingebetteten Systeme, Ressourcen (Sensoren/Aktoren) und Anwendungen dar. IEC61499 ist Programmiersprachenunabhängig, da es nur die Prinzipien und die Aufbauweise definiert.

Eine Entwicklung der Anwendung für verteiltes Rechnen nach IEC61499 erfolgt in graphischer Form, wo die Blöcke miteinander nach dem Signalfluss-Modell verbunden werden (Bild 3.5). Es stehen zwei Blocktypen zur Verfügung: *Basic Function Blocks* (BFB) und *Composite Function Blocks* (CFB). Der BFB ist die kleinste Einheit, die den einfachsten Operationen (mathematische, logische Operationen) dient. Der CFB ist ein System, das mehrere BFB und CFB beinhalten kann, in dem die modulare Lösung je einer Aufgabe möglich ist. Zwischen den Blöcken sind zwei Informationsflüsse vorgesehen. Die verschiedenen Signale, die den bestimmten physikalischen Werten oder Zuständen entsprechen, werden durch einen Datenfluss (Ein-/Ausgangsdaten) abgebildet (Bild 3.5, blaue Pfeile). In jedem Block wird ein eigener Algorithmus berechnet, der die Eingangsdaten für seine Berechnung nimmt und die Ausgangsdaten als Ergebnis zur Verfügung stellt (Bild 3.5, rote Pfeile). Wenn die Berechnung abgeschlossen ist, gibt der Block ein Ausgangsereignis an, welches als Eingangsereignis für den anderen Block genutzt werden kann. Jeder Block startet die Berechnung mit einem eigenen Eingangsereignis.

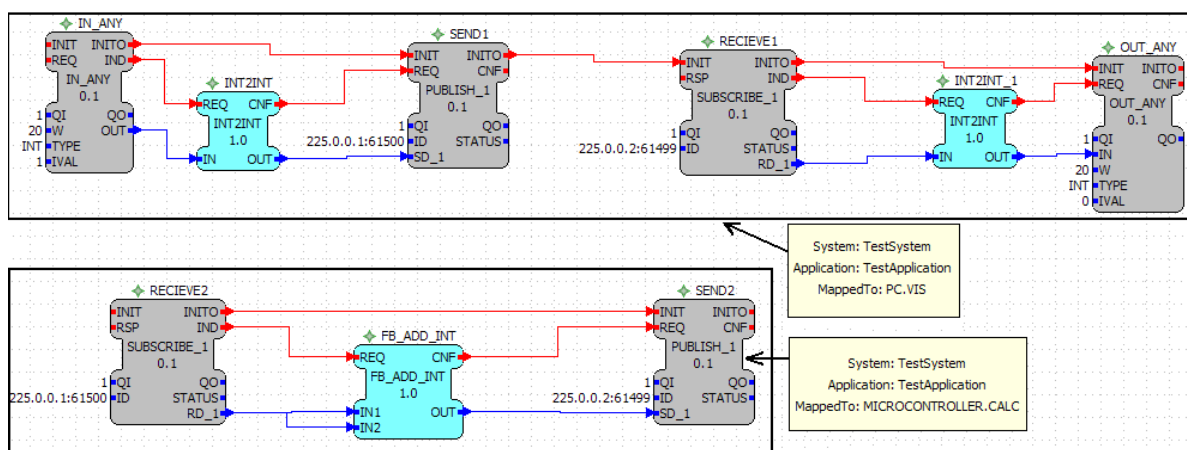


Bild 3.5 – 4DIAC-DIE, Aufbaubeispiel mit den Daten- und Ereignisflüssen

Bild 3.5 veranschaulicht ein Aufbaubeispiel in der nichtkommerziellen, IEC61499-basierten Entwicklungsumgebung *4DIAC-IDE* [95], in dem ein System in zwei Hardwaremodule „PC.Visionation“ und „Microcontroller.Calculation“ aufgeteilt wird. Der Datenaustausch erfolgt per Ethernet-Netzwerk, indem jedem Hardwaremodul eine Netzwerkadresse (ID) zugewiesen wird.

Eine Einsatzmöglichkeit des Automatisierungsstandards IEC61499 für verteiltes Rechnen im mechatronischen System wird im Bild 3.6 erläutert.

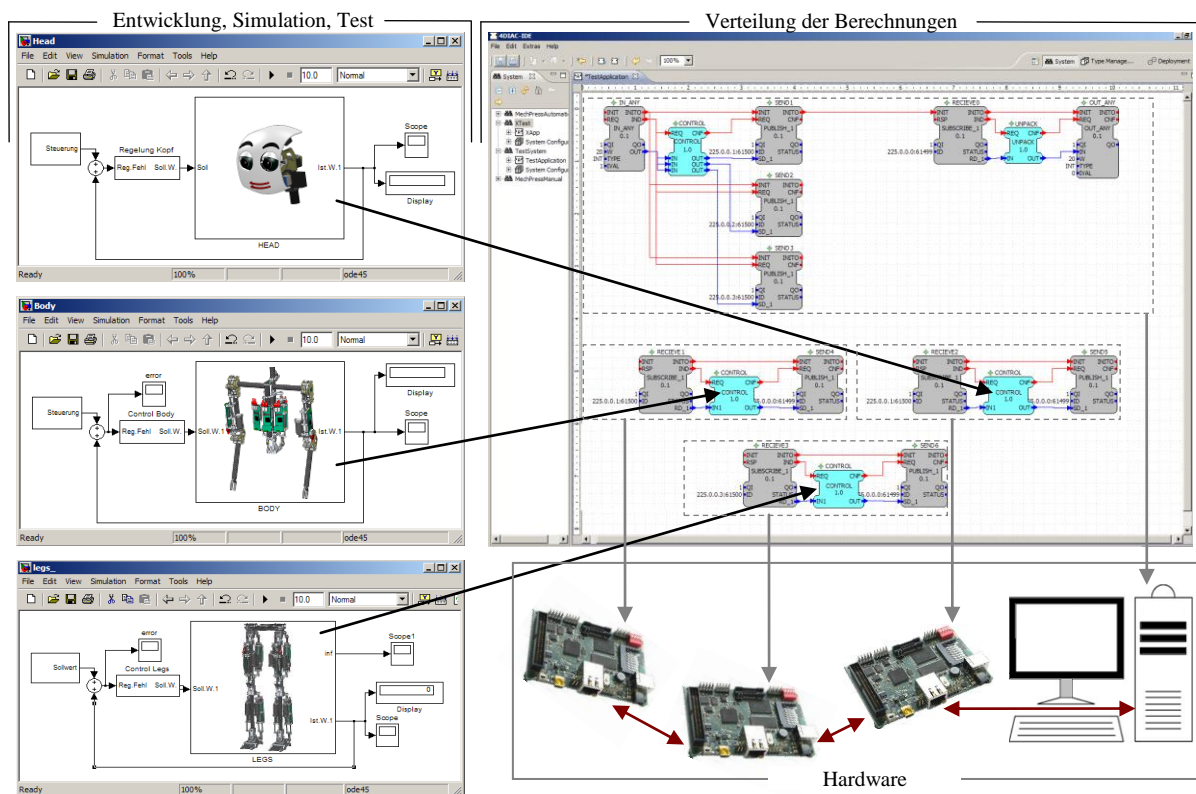


Bild 3.6 – IEC61499 für verteiltes Rechnen im mechatronischen Robotersystem

Der Entwurf erfolgt in drei Schritten. Im ersten Schritt (Bild 3.6, links) werden die einzelnen Modelle für jeden Bestandteil eines mechatronischen Robotersystems in der RCP-fähigen Software (z.B. Matlab/Simulink) graphisch entworfen. Die physikalischen Eigenschaften der entwickelten Modelle werden verifiziert. Danach werden die benötigten Regelkreise und gewünschten Steuerungsalgorithmen implementiert und simuliert. Zum vollständigen Test der entwickelten Steuerung/Regelung sind an diesem Schritt hybride Simulationstechniken (SiL und HiL) einsetzbar.

Im zweiten Schritt wird die Implementierung des Standards IEC61499 in *4DIAC-IDE* für verteiltes Rechnen verwendet. Jede entworfene und getestete Regelung und Steuerung wird in einen CFB-Block des Standards IEC61499 umgewandelt. Danach, im dritten Schritt, wird das gesamte Steuerungssystem aus einzelnen CFB-Blöcken aufgebaut (Bild 3.6, rechts oben), und jedem Block eine eigene Hardware zugeteilt. Alle Hardwareteile sind untereinander per Kommunikationsnetzwerk verbunden.

Der Vorteil der dargestellten Einsatzmöglichkeit des Standards IEC61499 (Implementierung in *4DIAC-IDE*) liegt bei der RCP-fähigen Software in der Integrierung des verteilten Rechnens.

3.2.1.3 Softwarearchitektur im Entwicklungsprozess für den Roboter ROTTO

Bei der Entwicklung der Steuerungs- und Regelungsalgorithmen für den humanoiden Roboter ROTTO wird primär die RCP-fähige Software *Matlab/Simulink* eingesetzt. Diese Programmier- und Simulationsumgebung ist in der Lage, aus einem grafisch entwickelten Programm bzw. Modell den Programmquellcode in eine höhere Programmiersprache zu transformieren. Die komplexeren Algorithmen für den Roboter ROTTO, die grafisch schwer und aufwendig darzustellen sind, werden direkt in Textform in *Eclipse-IDE* (*Integrierte Entwicklungsumgebung*) [96] entworfen. Die SPS-spezifische Softwarestrukturen sind mit einer SPS Software implementierbar. Auch eine moderne SPS Software (z.B. [97]) lässt sich an den Ziel-Prozessor, durch den Einsatz von Code-Umwandlern (eng: Compiler), für eine höhere Programmiersprache anpassen. Die genannten Werkzeuge bilden die erste Schicht der Softwarearchitektur (Bild 3.7), und sind in der Lage aus einem Modell den Code in einer höheren Programmiersprache zu generieren.

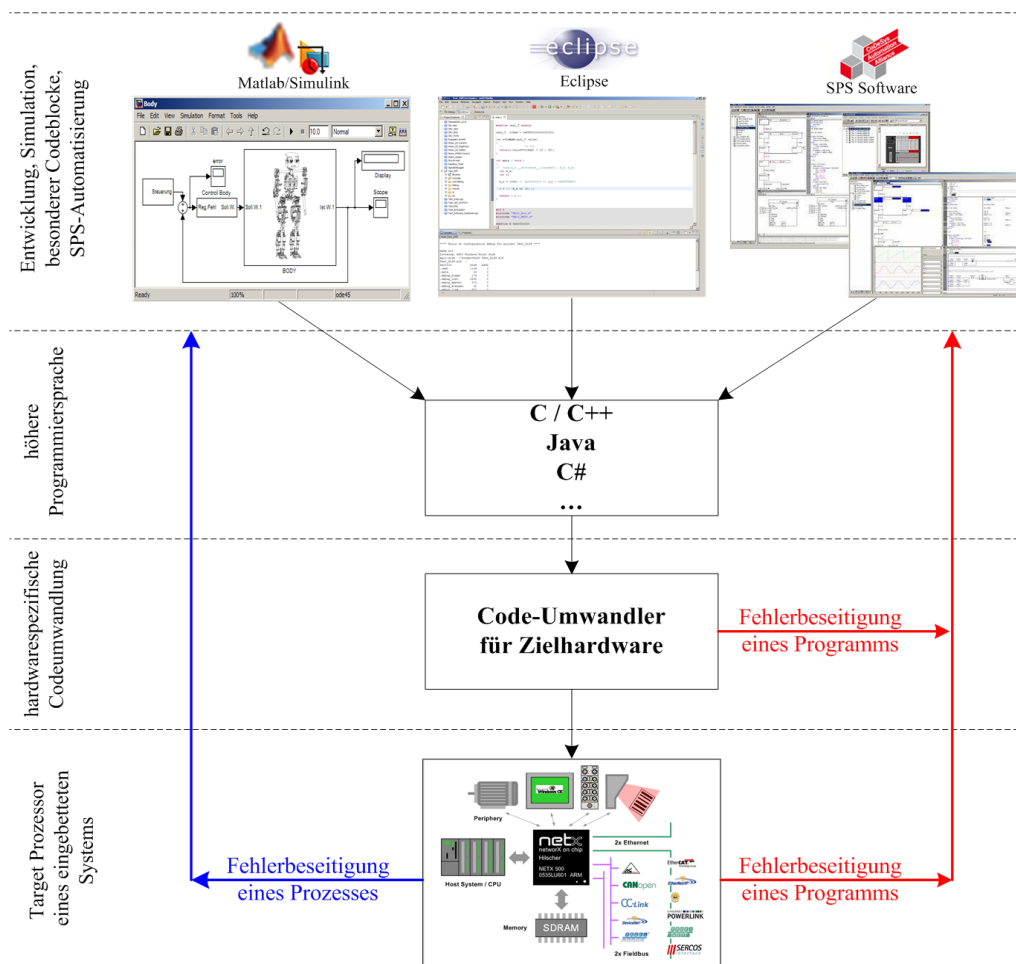


Bild 3.7 – Softwarearchitektur in der Softwareentwicklung

Die höhere Programmiersprache ergibt sich in der zweiten Schicht. Die dritte Schicht beinhaltet den Code-Umwandler für die Zielhardware, der aus einer höheren Programmiersprache einen spezifischen Maschinenbefehlfluss erzeugt. Höhere Programmiersprachen bieten die Möglichkeit, mit einfachen standardisierten Sprachstrukturen (class, union, struct u. a.) Programme zu schreiben. Dabei profitiert der Entwickler von der starken Abstrahierung der Be-

sonderheiten der Zielhardware-Architektur (Stackaufbau, Parameterübergabe beim Funktionsaufruf, Interrupts u. a.).

Der Code-Umwandler wandelt den Code der höheren Programmiersprache in den ausführbaren Code für den Ziel-Prozessor eines eingebetteten Systems bzw. einer Technologieplattform um. Auf der Ebene der Zielhardware sind verschiedene Möglichkeiten zur Fehlerfindung und Fehlerbeseitigung vorhanden, die tief in dem Code-Umwandler und der Entwicklungswerkzeuge (Monitoring, Debug) integriert sind. Die hardware-spezifische Codeumwandlung und Fehlerbeseitigung werden in den nachfolgenden Unterkapiteln beschrieben.

3.2.2 Hardware-spezifische Codeumwandlung

Das Ziel der hardware-spezifischen Codeumwandlung (eng: compiling) ist der Code einer höheren Programmiersprache in den hoch optimierten, ausführbaren Code für den Ziel-Prozessor einer Technologieplattform umzuwandeln. Wie oben dargestellt, konzentriert sich die vorliegende Arbeit dabei auf die neue Technologieplattform NetX10. Die erforderliche Anpassung der Werkzeugkette für die Codeumwandlung beim Prozessor NetX10 wird im Nachfolgenden beschrieben.

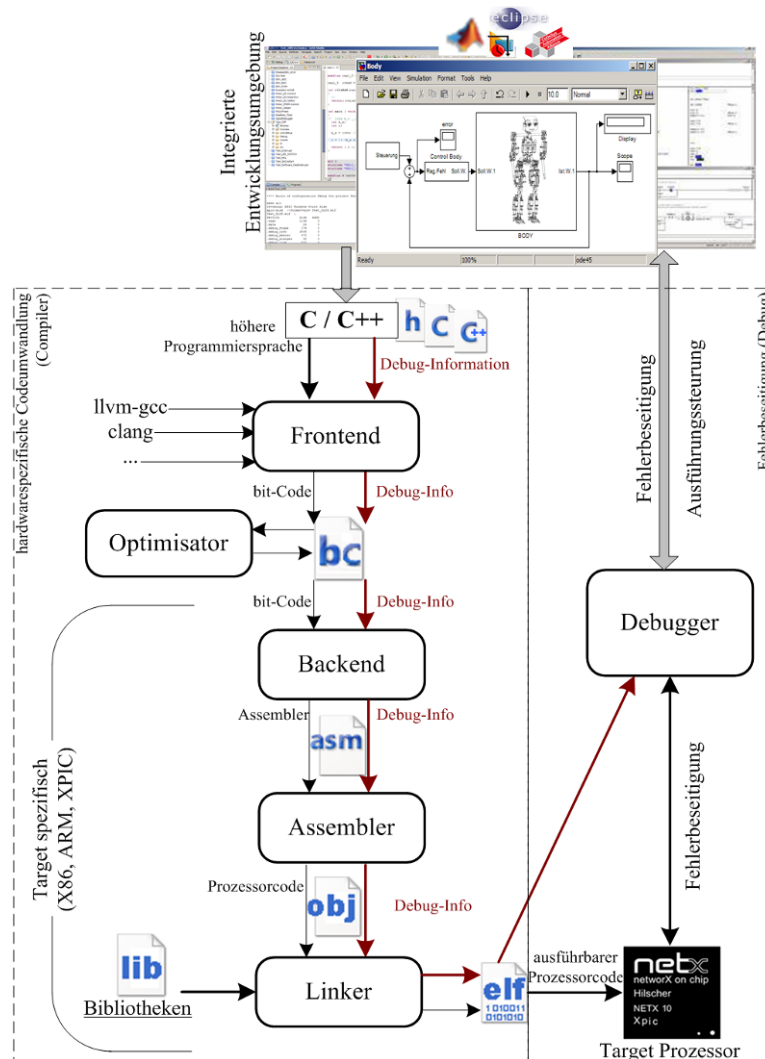


Bild 3.8 – Struktur und Werkzeuge des xPIC C-Compilers

Wegen des weitverbreiteten Einsatzes der C/C++ Programmiersprachen für die Hardwareprogrammierung, ist der GNU C-Codeumwandler (compiler) [98] für die Portierung auf die xPIC-Architektur gewählt worden. Die Qualität des generierten Prozessorcodes und die Fehlerbeseitigungsmöglichkeiten sind von der gesamten Struktur eines Compilers abhängig. Die angestrebte Struktur des xPIC-Compilers wird im Bild 3.8 veranschaulicht.

Die Struktur des xPIC-Compilers besteht aus mehreren Software-Werkzeugen: dem Frontend, der Optimisatoren, dem Backend, dem Assembler und dem Linker. In der Fehlerbeseitigung (eng: debugging) wird ein zusätzliches Werkzeug – der Debugger – eingesetzt, der mit dem Compiler nah verbunden ist. Jede dieser Komponenten wurde auf die xPIC Architektur portiert bzw. angepasst, um eine nahtlose Werkzeugkette zu bilden.

3.2.2.1 Front- und Backend des Codeumwandlers

Front- und Backend bilden die Bestandteile des Compilers für die C-Programmiersprache. Beide Komponenten wurden vom LLVM-Projekt [99] übernommen. Der Frontend *LLVM-GCC* ist ein modifizierter GNU-GCC-Compiler, der in der Lage ist, einen plattformunabhängigen Code (Bitcode – .bc) zu generieren. Gleichzeitig ermöglicht der Frontend den Einsatz aller Einstellungsoptionen des GNU-GCC-Compilers. Als Frontend kann nicht nur der *llvm-gcc* C-Compiler sondern auch der *Clang* (eigene llvm C-Compiler des LLVM Projektes [100]) und andere Werkzeuge verwendet werden. Sie erzeugen alle einen Zwischencode bzw. Bitcode, für den die Optimisatoren verschiedene Optimierungen durchführen können. Anschließend kommt der Backend zum Einsatz.

Der Backend übersetzt den vom Frontend generierten Bitcode in den xPIC-Prozessor-spezifischen Assembler-Code (Assembler – .s, .asm). Die Portierung des Backends ergibt den ersten Schritt in der Anpassung des C-Compilers auf die xPIC-Architektur. Hier werden alle Besonderheiten der xPIC Architektur, wie z.B. der Aufbau des Stacks, die Funktionsaufrufe oder die Rücksprünge, betrachtet. Der Backend erzeugt architektur-spezifische Assemblerdateien. Die Generierung der Objektdateien vom LLVM ist zurzeit in der Entwicklungsphase, deshalb wurde im Rahmen dieser Arbeit für den neuen netX10/xPIC Prozessor das Binutils-Projekt [101] herangezogen. Der existierende Backend unterstützt bereits folgende Architekturen: Alpha, ARM, CellSPU, IA64, Mips, MSIL, PowerPC, Sparc, X86, X86_64, XCore, PIC-16. Der Backend-Quellcode für diese Architekturen dient der Grundlage für die Portierung auf eine neue Prozessor Architektur.

3.2.2.2 Prozessorcode

Der Assembler-Code wird mit dem Assembler-Werkzeug in die Objektdateien (.obj und .o) umgewandelt. Hier wird der adressenlose Prozessorcode mit der Verschiebungsinformation erzeugt und in definierte Sektionen unterteilt. Der zweite Schritt für die Anpassung des C-Compilers besteht in der Festlegung eines binären Formates für die prozessor-spezifischen Objektdateien (Binary File Description – BFD). Dieses entwickelte Format wird für alle xPIC-spezifischen Bibliotheken und ausführbaren ELF-Dateien verwendet. Mit dem definierten BFD-Format lassen sich direkte Codeumsetzer (Assembler) und indirekte Codeumsetzer (Disassembler) anpassen.

Der dritte Schritt ist die Portierung des Linkers, der den ladbaren und ausführbaren XPIC-Prozessorcode (.elf) aus allen Objektdateien und den Bibliotheken zusammensetzt. Hier werden architektur-spezifische Codeumsetzungen (Relocation) und Codeumwandlungen (Longjump, Farload/Farstore) vorgenommen. Es wird eine default-Steuerungsdatei für den

Linker (Link Deskription Script – .ld) definiert, in der alle Speicheradressen, ladbaren Sektionen und definierten Symbole vorgegeben sind.

Assembler, Linker und Werkzeugsatz für die Manipulationen mit den adressenlosen Objektdateien, den Bibliotheken und ausführbaren Anwendungen wurden vom Binutils-Projekt übernommen und für den xPIC-Prozessor portiert.

3.2.2.3 Code-Bibliotheken

Die vorgefertigten Bibliotheken bieten eine flexible Möglichkeit zur Wiederverwendung von oft gebrauchten abgeschlossenen Funktionen und dienen der Aufwandsreduzierung bei der Programm-Entwicklung. Die Einsatznotwendigkeit der Bibliotheken für den xPIC-Prozessor besteht außerdem daher, da nicht alle mathematischen, trigonometrischen und algorithmischen Funktionen in der Prozessor-Hardware direkt unterstützt werden. Diese oft verwendeten Funktionen werden mit Hilfe von Softwaremodulen realisiert und in die Bibliotheken zusammengefasst. Im Bild 3.8 (unten links) sind die Bibliotheken in der Struktur des xPIC C-Compilers dargestellt.

Ein Teil der mathematischen bzw. trigonometrischen Funktionen werden mit Hilfe von des sogenannten *Coordinate Rotation Digital Computer* (CORDIC) auf der Technologieplattform NetX10 berechnet. netX10 verfügt über ein spezifisch entworfenes CORDIC-Hardwaremodul [102]. Das CORDIC-Modul benötigt dazu eine besondere Art der Ansteuerung (Aufladen der Koeffizienten, Umrechnung des Zahlenformats usw.). Diese Ansteuerung wird in einer separaten Bibliothek realisiert.

Die Hardware-Abstrahierungsschicht (Hardware Abstraction Layer - HAL) für alle Hardwaremodule der Technologieplattform wird in der separaten Bibliothek veranschaulicht. Die HAL ermöglicht dem Nutzer eine universelle Schnittstelle für den Zugriff auf die hardware-spezifischen Funktionen (Ein- / Ausgänge, PWM-Signale, Analog- und Encodereingänge).

3.2.3 Fehlerbeseitigung

Eine der wichtigsten Eigenschaften der Werkzeugkette liegt in der Bereitstellung der Unterstützung zur einfachen Fehlerfindung bzw. Fehlerbeseitigung (Debugging). Im Bild 3.7 sind zwei Fehlerbeseitigungstypen aufgeführt: Fehlerbeseitigung im Programm und Fehlerbeseitigung im Prozess. Ersteres ermöglicht die schrittweise Ausführung eines Programms und die Beobachtung der Zustände des Prozessors, der Speicher und der Register. Auf diese Weise können die Implementierungsfehler im Programmcode hardwarenah gefunden und korrigiert werden. Die Fehlerbeseitigung im Prozess ist Voraussetzung für die Beobachtung bzw. Beeinflussung des Verlauf der gewünschten Prozessvariablen während der bestimmten Ausführungszeit. Beide Methoden dienen der Erhöhung der Qualität des Gesamtsystems sowie der Minimierung des Entwicklungsaufwandes.

3.2.3.1 Debug im Programmcode

Der Frontend generiert die notwendige Debug-Information (Bild 3.8, rote Pfeile – *Debug-Info*), die durch den Backend, den Assembler und den Linker propagiert wird. Der Linker teilt diese Debug-Information in getrennte Sektionen innerhalb einer ausführbaren ELF-Datei bzw. Anwendung auf. Die Debug-Information wird über alle Programmzeilen, definierten Variablen und genutzten Funktionen in die C-Quelldateien eingetragen. Der Debugger nutzt diese Information zur richtigen Steuerung des xPIC-Prozessors und zur hardwarenahen Fehlerbesei-

tigung. Die Fehlerbeseitigungssoftware (eng: debugger) ist ebenfalls an das GNU-Projekt *GDB* [103] angelehnt, da GDB mit zahlreichen Compiler-Komponenten kompatibel ist.

Der GDB-Debugger ist in der Lage die Ausführung eines Programms im Prozessor zu steuern. Mit Hilfe des Debuggers kann das Programm von einer gewünschten Codezeile im Prozessor schrittweise ausgeführt und die Zustände des Prozessors, der Speicher und der Register beobachtet werden.

Der GDB-Debugger lässt sich über zwei Schnittstellen steuern. In der ersten akzeptiert er die Benutzerbefehle aus der Kommandozeile. Dabei kann der erfahrene Benutzer mit dem vollen Befehlssatz die gesamte Leistung des Debuggers nutzen, um komplexe Programmfehler zu finden. Für den schnellen Fehlerbeseitigungsprozess (z.B. zur Untersuchung der Variablen, des Speichers und der Prozessorregister) steht eine weitere Schnittstelle zur Verfügung, mit ihrer Hilfe kann der Debugger an eine integrierte Entwicklungsumgebung angekoppelt werden. Der GDB-Debugger für den xPIC-Prozessor wurde an die Eclipse-Entwicklungsumgebung [96] angebunden. Das gesamte Benutzerinterface der Eclipse xPIC Entwicklungsumgebung wird im Bild 3.9 veranschaulicht.

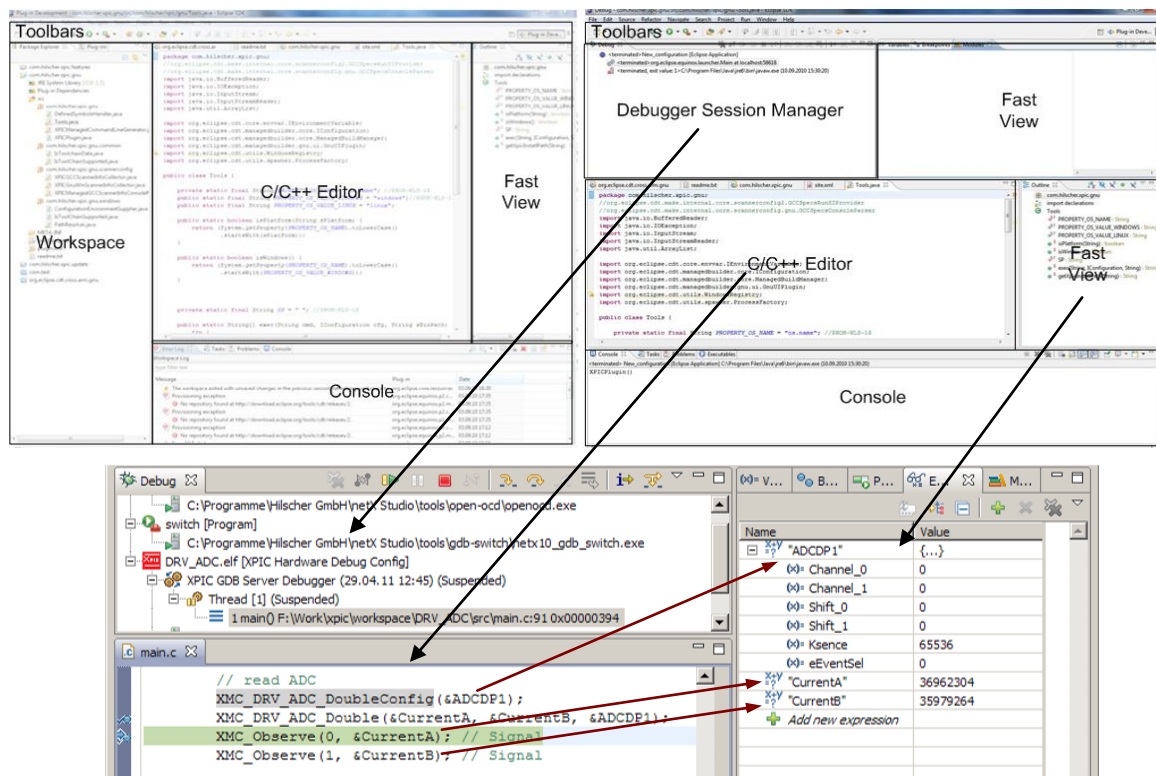


Bild 3.9 – Eclipse-Entwicklungsumgebung: Entwicklungs- (links) und Debug-Ansichten (rechts und unten)

Einerseits erlaubt die Eclipse-Entwicklungsumgebung eine Steuerung von der prozessor-spezifischen Codeumwandlung während der C/C++ Codeentwicklung (Entwicklungsansicht von links). Andererseits wird der Debugger während der Fehlerbeseitigung gesteuert (Debug-Ansicht von rechts und unten). Aus dem „Fast View“-Fenster sind die Werte der gewünschten Variablen, der Inhalt des spezifizierten Speicherbereichs und die Werte der Prozessorregister übersichtlich.

3.2.3.2 Debug eines Prozesses

Unter der Bezeichnung „Debug eines Prozesses“ wird eine Möglichkeit verstanden, den Verlauf der gewünschten Prozessvariablen während der bestimmten Zeit zu registrieren. Die Registrierung erfolgt in dem spezifischen Speicherbereich. Die Prozessdaten aus diesem Bereich werden zur weiteren Analyse in Form eines Diagramms oder von Kurven grafisch bereitgestellt. Das Bild 3.10 veranschaulicht den Debug eines Prozesses bzw. eine Prozessbeobachtung.

Abhängig von der Registrierungszeit wird die Prozessbeobachtung in zwei Betriebsarten aufgeteilt: in die kontinuierliche und die echtzeitfähige. In der kontinuierlichen Registrierung wird eine gewählte Variable bzw. mehrere Variablen in den spezifischen Speicherbereich in jedem Berechnungszyklus eingetragen (fette Pfeile im Bild 3.10). Mit einem Befehl wird dieser Speicherbereich ausgelesen und zu dem Entwicklungsrechner (Bild 3.10, rechts) zur Analyse übertragen. Je nach Anforderungen kann solche Beobachtung der Prozessvariablen von wenigen Millisekunden bis zu mehreren Tagen andauern.

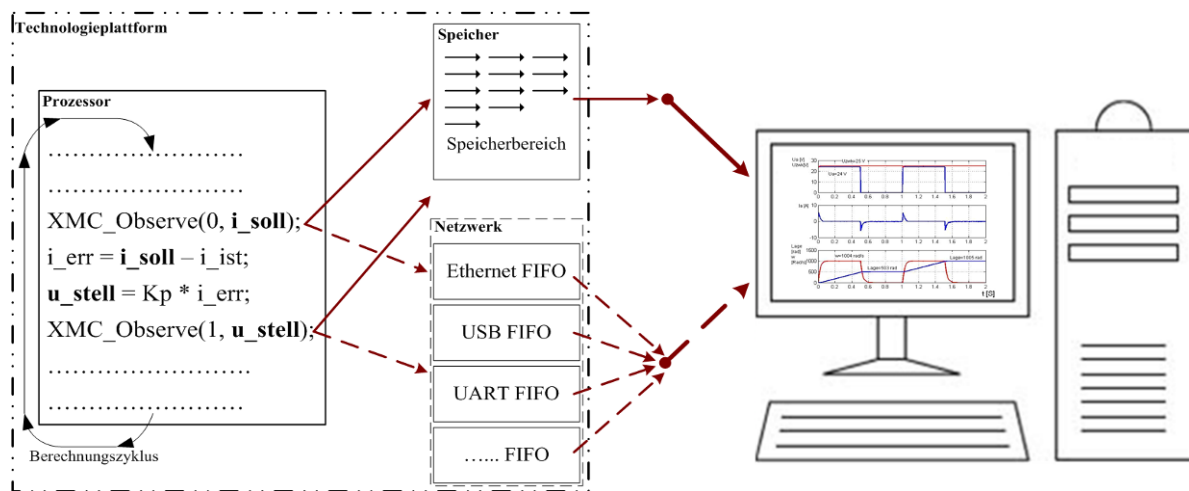


Bild 3.10 – Debug eines Prozesses

Bei der echtzeitfähigen Registrierung werden die Kommunikationsfähigkeiten der Technologieplattform genutzt. Die gewählte Variable wird direkt im Sendepuffer (FIFO) eines Datenübertragungsmoduls an jedem Berechnungszyklus gespeichert. Sobald der Sendepuffer voll ist, werden die Pufferdaten zum Entwicklerrechner übertragen. Dort werden die Daten sofort grafisch dargestellt und für weitere Analysen bereitgestellt. Je nach Größe des Sendepuffers können die Variablenwerte von geringen (über UART, SPI, USB Schnittstelle) bis großen Datenmengen (über USB, Ethernet Schnittstelle) beobachtet werden. Der Vorteil dieser Prozessbeobachtung liegt in der echtzeitgetreuen Prozessüberwachung. Darüber hinaus können die graphisch dargestellten Daten am Entwicklerrechner gespeichert und später (wie im kontinuierlichen Fall) für weitere Analysen verwendet werden.

3.3 Hardwarestruktur

Das Ziel, eine optimale Hardwarestruktur für das Steuerungssystem von humanoiden mobilen Robotern ROTTO zu erarbeiten, ist mit verschiedenen Schritten verbunden. Eine der wichtigsten Fragen bezieht sich auf den strukturellen Aufbau der Hardwarearchitektur (zentral oder dezentral).

Die Entscheidung über den Einsatz des zentralen bzw. dezentralen Aufbaus hängt von mehreren einzelnen Faktoren ab. Die wichtigste Rolle zur Entscheidungsfindung spielt der Determinismusgrad. Je höher die Anforderungen an die Echtzeitfähigkeit des Steuerungssystems sind, desto mehr verlagern sich die zentralen Steuerungsstrukturen hin zu den dezentralen Anordnungen. Gesamtkosten, Einfachheit der Inbetriebnahme und Wartung sind weitere wichtige Faktoren. In folgendem Unterkapiteln werden beide Steuerungsstrukturen diskutiert und ausgewählte Struktur für den humanoiden Robotern ROTTO dargestellt.

3.3.1 Zentrale Steuerungsstruktur

Bei der zentralen Architektur übernimmt ein leistungsfähiger zentraler Prozessor (oft ein PC) alle Aufgaben, wie beispielweise E/A, SPS und Bewegungssteuerung (Bild 3.11). Die Rechenleistung muss daher, im Vergleich zu dezentralen Prozessoren, deutlich höher sein. Vorteilhaft stellt sich eine einzige CPU dar, die keine Verteilung der Rechenaufgaben benötigt.

In der Softwareentwicklung ist eine Strukturierung und Modularisierung der Software, aus Gründen der Wiederverwendbarkeit und Wartung, möglich. Hier helfen auch die graphische Softwareentwicklung und Automatisierungsstandards (z.B. IEC 61499) mit objektorientierten Ansätzen. Da es nur ein Steuerungsprogramm gibt, fallen die Entwicklung und die Fehlerbehebung auf einer Plattform einfacher aus.

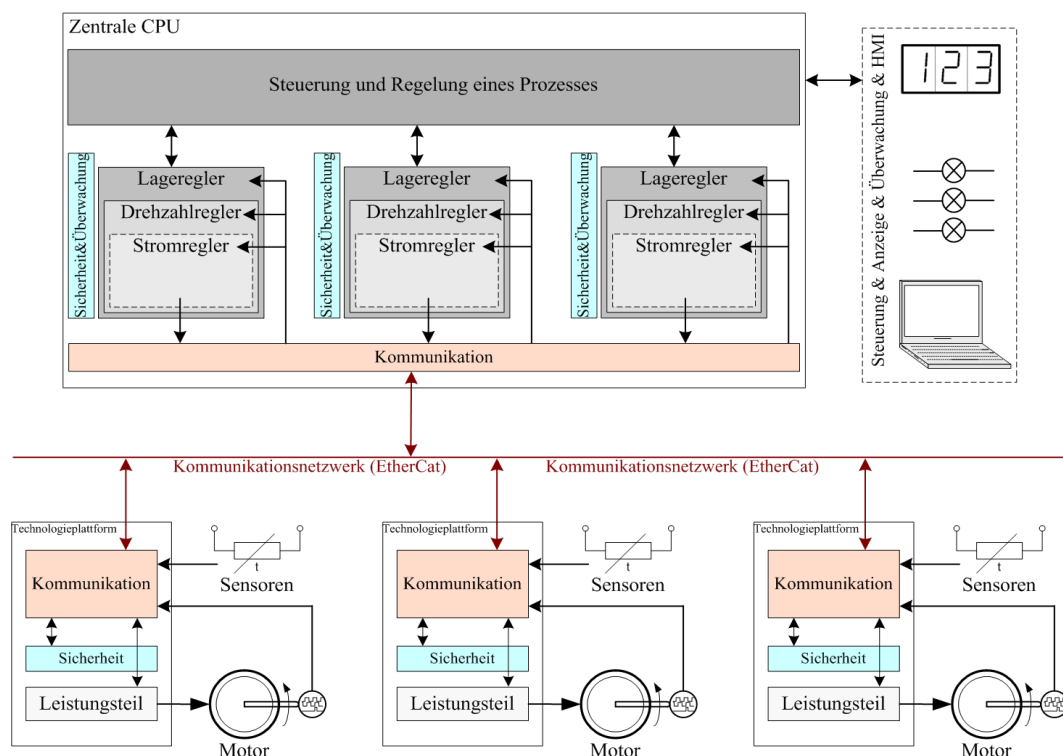


Bild 3.11 – Zentrale Steuerungsstruktur

Da für den humanoiden Roboter in in einer zentralen Steuerungsstruktur die Motorensteuerung im zentralen PC gerechnet werden muss, steigen die Echtzeitanforderungen an den Feldbus. Wird auf einen Feldbus für die E/A, Motorsteuerung u.ä. bestanden, dann kommen, neben der gestiegenen Anzahl von Knoten, noch die zeitlichen Anforderungen hinzu. Bei einer

Lageregelung auf dem PC kann oft nicht mehr als eine minimale Zykluszeit von 3–4 ms geleistet werden. Dabei erreichen die konventionelle Feldbussen schnell ihre Grenzen. Die Motorensteuerung auf einer zentralen CPU bringt aber auch viele Vorteile mit sich. Da die Lageregelung, die Kopplung und die Synchronisierung der Achsen allein durch die Softwareanteile im PC errechnet werden, können die Antriebseinheiten nur eine geringe Funktionalität besitzen (in manchen Fällen muss lediglich die Stromregelung implementiert werden), und damit auch einfach und kosteneffizient aufgebaut werden. Die synchronisierte Antriebe reagieren sehr schnell und die lästige Kopplung der Antriebe untereinander über spezielle Busse entfällt.

Der zentrale Rechner nimmt in der Regel die Rolle des Feldbusmaster in dem Kommunikationsnetzwerk. Aufgrund der großen Anzahl von Slaves (im Bild – Technologieplattformen) und Prozessdaten, ist es schwierig die Echtzeitanforderungen der Regelung bzw. der Steuerung einzuhalten, wenn die konventionellen Feldbusse verwendet werden. Daher werden für solche Aufgaben zunehmend die echtzeitfähigen Ethernet-basierten Feldbusse eingesetzt.

Die Leistungsfähigkeit der modernen Rechner und der nahezu unbegrenzt vorhandene Speicher ermöglichen die kompliziertesten Tabellenkopplungen zwischen den Antriebseinheiten.

Bild 3.11 veranschaulicht den Signalflussplan einer EtherCAT-basierten zentralisierten Motorregelung, die in einem sechsbeinigen Schreitroboter ANTON implementiert ist [38]. In diesem Konzept beinhaltet das Antriebmodul lediglich die Datenerfassung der Sensoren und die Ansteuerung der Leistungstransistoren im Umrichter. Die überlagerten Geschwindigkeits- bzw. Lageregelkreise sind über den Real-Time-Ethernet-Bus EtherCAT [104] angeschlossen und die entsprechenden Motorenregler, sowie die gesamte Robotersteuerung, werden in einer zentralen PC-Steuerung errechnet.

3.3.2 Dezentrale Steuerungsstruktur

Die Anzahl der eingesetzten Technologieplattformen wird bei den verteilten Steuerungsarchitekturen durch die Anzahl der Module bestimmt. Jedes Modul besitzt eine Prozesssteuerung und eine Antriebsregelung (Bild 3.12). Neben den verteilten Steuerungen gibt es in der Regel noch einen zentralen Master-CPU, der die zentralen Aufgaben, wie Management, Steuerung, Überwachung, Statistik und HMI übernimmt. Die dezentralen Technologieplattformen haben die Aufgabe, ihre speziellen Module zu automatisieren. Meistens benötigen sie weder Visualisierung noch Bedienung, da diese Funktionen im gesamten System vom Master-CPU übernommen werden.

Der Umfang der Steuerungsaufgabe in den dezentralen Technologieplattformen ist, im Vergleich zur gesamten Automatisierungsaufgabe, gering, deshalb können hier relativ leistungsschwache und damit preisgünstige Technologieplattformen eingesetzt werden. Diese müssen mit der Master-CPU über einen schnellen Bus kommunizieren. Meist wird hier ein Ethernet-basiertes Netzwerk eingesetzt und der Datenaustausch erfolgt in der Regel über zyklisch auszutauschende Netzwerkdaten. Beim Start und Stopp des gesamten Systems müssen die Technologieplattformen mit dem Master-CPU synchronisiert werden.

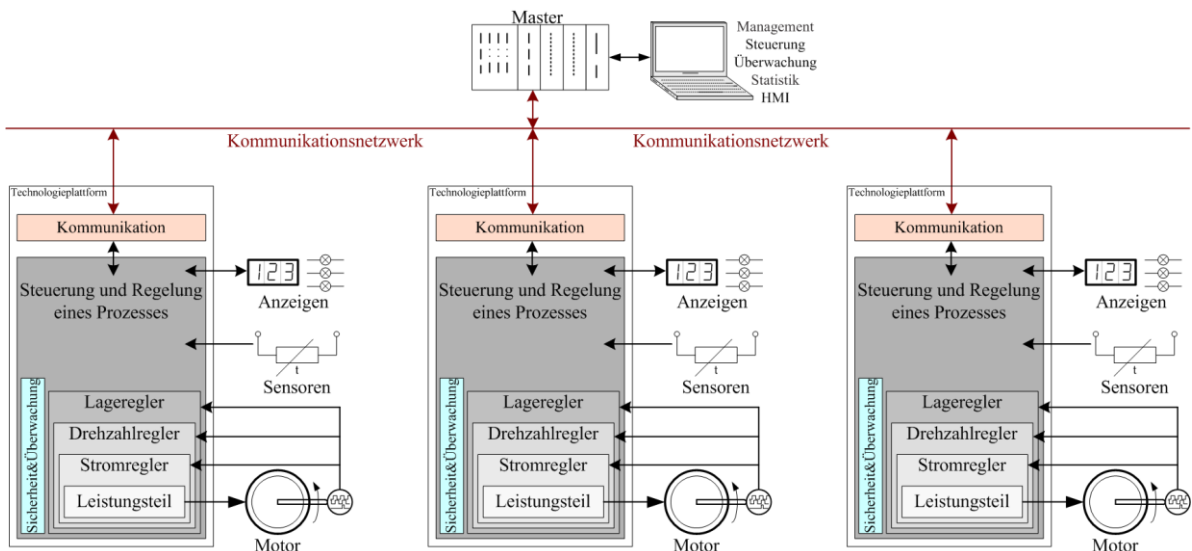


Bild 3.12 – Dezentrale Steuerungsstruktur

Die auf den dezentralen Technologieplattformen laufenden SPS-Programme sind meist nicht sehr aufwändig, da die zu automatisierende Teilaufgabe einfach ist. In der Regel reichen wenige Zeitebenen bzw. wenige Tasks. Ein erhöhter Aufwand ist auf der Seite der Master-CPU für die Kommunikationsprojektierung und Programmierung notwendig. Die Verwaltung und Verteilung der SPS-Programme auf die einzelnen Steuerungen ist nicht zu unterschätzen, daher ist die Software zum verteilten Rechnen hier anwendbar.

Die E/As und Feldbustopologien können auf den dezentralen Technologieplattformen entsprechend einfach gestaltet werden. Es sind nur relativ wenige I/O-Signale zu erfassen, da nur eine begrenzte Aufgabe automatisiert werden muss. Die Diagnose und Konfiguration der E/As und des Feldbusses müssen entsprechend für jede Technologieplattform einzeln und individuell durchgeführt werden.

Die Steuerung mehrerer Motoren durch eine einzelne dezentrale Technologieplattform ist, aufgrund ihrer geringen Leistungsfähigkeit, problematisch. Müssen mehrere Motoren bewegt und synchronisiert werden, kann dadurch auf den Einsatz von intelligenten und damit teuren Technologieplattformen nicht verzichtet werden. Die Motorenregelung (Strom, Drehzahl, Lage usw.) erfolgt auf den Technologieplattformen dezentral. Die Synchronisierung zwischen den Modulen ist durch den Einsatz der echtzeitfähigen Netzwerke möglich.

3.3.3 Architektur der Hardware und Steuerungsstruktur für den humanoiden Roboter ROTTO

Die Komplexität und Vielfalt der Steuerungsaufgaben für einen humanoiden Roboter erfordern den Einsatz sowohl dezentraler als auch zentraler Steuerungsarchitekturen. Bedingt durch die entsprechenden Vorteile bzw. Nachteile beider Ansätze ist oft eine Mischung der Elemente der zentralen bzw. dezentralen Architektur notwendig. Da in beiden Architekturen die Kommunikationstechnik eine große Rolle spielt, ist eine flexible Unterstützung der Kommunikationstechnik sowie der Steuerungstechnik, auf einer Technologieplattform wie NetX, von großem Vorteil.

Die Verfügbarkeit leistungsfähiger und preiswerter Technologieplattformen mit breiten Automatisierungsfähigkeiten und mit echtzeitfähigen Kommunikationsbausteinen in Verbindung

mit der modernen Leistungselektronik führt zu neuen Konzepten von intelligenten, elektrischen Antrieben für mechatronische Robotersysteme. Die immer leistungsfähigere und zuverlässigere Kommunikation beeinflusst den Fortschritt von sogenannten vernetzten, intelligenten Antrieben, deren Vorteile insbesondere bei der Entwicklung humanoider mobiler Roboter zum Vorschein kommen.

Mit solchen vernetzten, intelligenten Antrieben wurde der ROTTO ausgerüstet. Bild 3.13 veranschaulicht die, in [48] beschriebene und in vorliegender Arbeit durch die neueste Technologieplattform NetX10 erweiterte, hybride Hardware- und Steuerungsstruktur für den humanoiden Roboter. Die Netzwerkstruktur ist linear gebaut, was eine große Flexibilität für weitere Erweiterungen bietet. Darüber hinaus kann die Topologie des Systems von einer linearen Struktur umgewandelt werden in eine baumähnliche Struktur, mit Hilfe von Ethernet Switch-Modulen.

Die Auswahl der geeigneten Hardwarestruktur wird durch mehrere Faktoren beeinflusst. Ein zentraler Echtzeitrechner übernimmt einen großen Teil des Steuerungssystems, um die Entwicklung und die Modifikation der Algorithmen zu vereinfachen. Dazu wird eine moderne Bustechnik für die effiziente Vernetzung der Einzelkomponenten verwendet, zur Verringerung des Verdrahtungsaufwands und der Ausfallraten. Ein Echtzeit (Real-Time) Ethernet Bus bietet die Realisierung von verschiedenen Sicherheitskonzepten. Der Ausfall eines Teilnehmers (Antriebs oder Sensors) kann z.B. eine sofortige sichere Abschaltung oder sichere Begrenzung der Geschwindigkeit bzw. des Drehmoments bewirken. Es gibt breite Erweiterungsmöglichkeiten, denn alle Teilnehmer sind an einen Bus angeschlossen. Es ist möglich zusätzliche Rechner in einer Kommunikationsstruktur anzubinden, wenn die Rechenleistung nicht ausreichend ist oder die Realisierung weiterer komplexer Funktionen gewünscht wird. Außerdem kann der zentrale Steuerungsrechner weitere Busstränge auf die gleiche Weise anbinden.

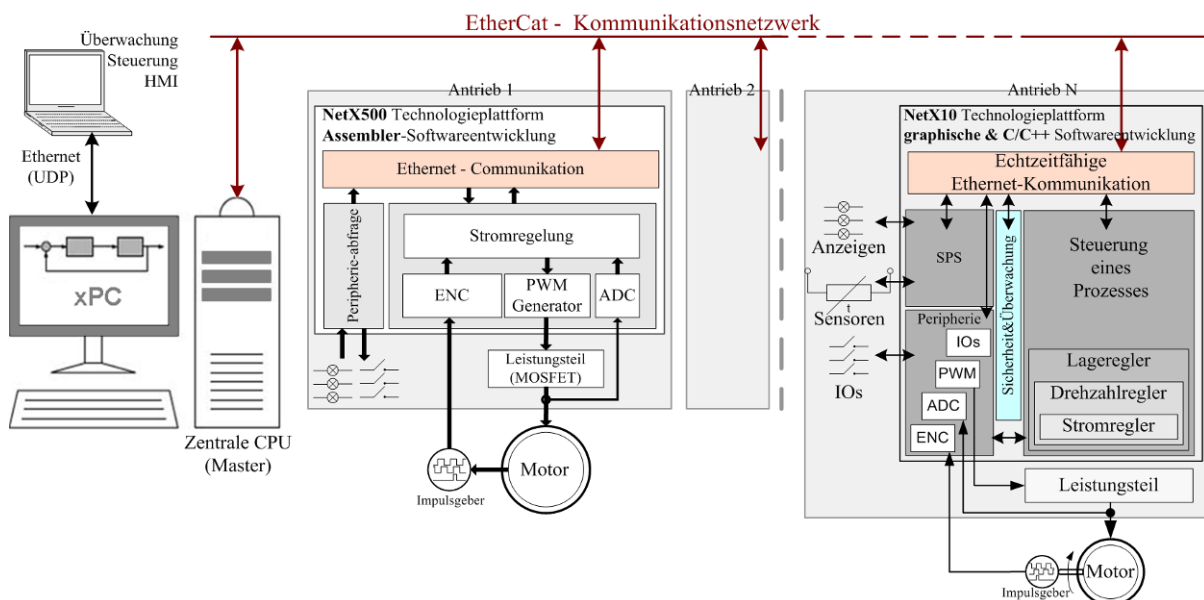


Bild 3.13 – Erweiterte Hardware- und Steuerungsstruktur für den humanoiden Roboter ROTTO (Technologieplattformen für die Antriebe: links–NetX500, rechts–NetX10)

Der größte Vorteil dieses Konzeptes liegt in der Möglichkeit der genauen Synchronisation aller Teilnehmer am Bus, durch die Eigenschaften des echtzeitfähigen Netzwerks und die deterministische Funktion des zentralen Rechners (CPU) mit einem Echtzeitbetriebssystem.

Es wurden mehrere Varianten des Betriebssystems, wie RTAI Linux [105], QNX Neutrino [106], Windows/RTX [107] und xPC-Target [39], bei der Entwicklung des Schreitroboters untersucht. Bedingt durch die große Erfahrung in der RCP-fähigen Entwicklungssoftware Matlab, die einfache Bedienung und native Integration mit einer aktuell primären Entwicklungsumgebung Matlab/Simulink, und auch durch die hervorragenden Echtzeiteigenschaften wurden das xPC-Target als Betriebssystem ausgewählt. Die Integration des xPC-Targets in die Matlab/Simulink ermöglicht u.a. eine schnelle Umsetzung und direkte Tests des entwickelten Steuerungssystems, was den Entwicklungs- und Inbetriebnahmeprozess erheblich beschleunigt.

Als zentraler Steuerungsrechner für humanoiden Roboter ROTTO dient ein moderner Standard-PC (AMD Athlon x64 3800+). Für die Ankopplung des Rechners an die Roboterhardware über einen Echtzeit-Ethernet Bus, wurde eine cifX-Master Erweiterungskarte [108] in das xPC-Target System integriert. Sie führt eine Echtzeitkommunikation mit mehreren Busteilnehmern (hier Roboterantriebe und Sensoren) mit einer Busgeschwindigkeit von 100 MBit/s aus. Der ausgewählte Rechner schafft die Voraussetzung, die Abtastzeit des Steuerungssystems deutlich unter 1 Millisekunde einzustellen. Dabei erfolgt im Zentralrechner, neben der Berechnung aller Komponenten des Steuerungssystems, auch die Berechnung der Geschwindigkeits- und Lageregelkreise für alle Antriebe im System (mit Technologieplattform NetX500), wie es im Bild 3.11 illustriert ist. Ein solches Konzept ermöglicht eine enorme Flexibilität bei der Entwicklung und Inbetriebnahme des Steuerungssystems.

Im Bild 3.13 sind zwei dezentralisierte Technologieplattformen aufgeführt. NetX500 ist ein effizienter Kommunikationsprozessor, der vor allem Feldbusse und echtzeitfähige Busse unterstützt. Darüber hinaus ist er für die Aufgaben der Visualisierung (HMI) und der Bewegungssteuerung anwendbar. Die Technologieplattform NetX500 lässt sich sehr einfach, durch die zur Verfügung stehende API, in mechatronisches System für sichere und stark determinierte Datenübertragung mit einer Busgeschwindigkeit von 100 MBit/s anschließen. Für die Motorensteuerung sind spezifische Hardwaremodule vorhanden (ADC, Encoder, PWM, E/A u.a.), die bei der Steuerung eines Synchronmotors im linearen Antrieb [45 – 47] für den ROTTO eingesetzt werden. Als nachteilig für die Implementierung der Motorensteuerung auf diese Technologieplattform, hat sich die Programmierung der Coprozessoren (XPEC's) in Assembler erwiesen. Die Assembler-Programmierung verlängert den Entwicklungsprozess und erschwert die Fehlerbeseitigung. Aus diesem Grund wurde nur die Stromregelung auf dieser Technologieplattform realisiert (s. Bild 3.13).

Mit der Übernahme aller Vorteile der Technologieplattform NetX500, hat die neueste Technologieplattform NetX10 mit dem XPIC Prozessor die o.g. Mängel aufgehoben. Während des Kooperationsprojektes [109] sind viele Hardwaremodule (ADC, Encoder, PWM, SPI u. a.) hinsichtlich der hohen technischen Anforderungen überarbeitet und eine neue Softwareinfrastruktur entworfen worden. Die Softwarestruktur ist oben im Unterkapitel 3.2.1.3 beschrieben. Sie erlaubt sowohl eine einfache, grafische und C/C++ Programmentwicklung, eine vollständige Modellierung und RCP, als auch eine umfangreiche Fehlerbeseitigung während der Tests und der Inbetriebnahme. Dadurch wurde die Möglichkeit geschaffen, die übergeordneten Geschwindigkeits- und Lageregelkreise unmittelbar in der Technologieplattform zu implementieren. Darüber hinaus gibt es die Möglichkeit, eine zusätzliche SPS-Laufzeitsystem bzw. Steuerung eines Prozesses auf der gleichen Plattform zu realisieren. Die gesamte Lösung ist im Bild 3.13 mit dem linken Antriebmodul veranschaulicht.

3.4 Zusammenfassung

Im ersten Teil des vorliegenden Kapitels wurde die Technik zur Entwicklung des komplizierten Steuerungssystems eines mechatronischen Systems vorgestellt. Die Entwicklung erfolgt nach den Entwurfsschritten des V-Modells. Das Steuerungssystem wurde in zwei Stufen, der Steuerungsalgorithmen und der Steuerungshardware, betrachtet.

Anschließend wurde die Softwareentwicklung (bzw. Steuerungsalgorithmen) für das mechatronische System mit den wichtigsten Anforderungen an die einzelnen Komponenten dargestellt. Diese Anforderungen wurden der Entwicklung eines Konzeptes zugeführt, welches die hybride (graphische / C/C++ / SPS) Softwareentwicklung mit den RCP-Werkzeuge, die Möglichkeit zur Verteilung der Rechenaufgaben, effiziente hardwarespezifische Codeumwandlung und umfangreiche Fehlerbeseitigung in den Programmen und Prozessen beinhaltet.

Zum Schluss wurde die Frage zum zentralen und dezentralen Aufbau der Hardware und des Steuerungssystems im Detail diskutiert und dabei die gängigen Aufbauarten der vernetzten Antriebe dargestellt. Anschließend wurden die Strukturen des Steuerungssystem des humanoiden Roboters ROTTO gezeigt und ihre Komponenten, wie zentraler Rechner, echtzeitfähige Kommunikationsbusse, eine existierende und eine neuentwickelte erweiterte Technologieplattformen (NetX500 und NetX10), ausführlich präsentiert.

Zu den wichtigsten Eigenschaften des entwickelten Konzeptes zum Software- und Hardwareaufbau des Steuerungssystems gehören die hohe Flexibilität und Erweiterbarkeit des verwendeten, echtzeitfähigen Kommunikationssystems, die schnelle Entwicklung und die Inbetriebnahme durch eine grafische Softwareentwicklung und den RCP, die breite Wiederverwendbarkeit des entwickelten Programmcodes und eine greifende Fehlerbeseitigung in den Programmen und Prozessen. Die Verwendung des gewählten leistungsfähigen Kommunikationssystems und die effizienten Softwarewerkzeuge, ermöglicht die schnelle Realisierung beliebiger Regelungskonzepte. Dabei kann eine flexible zentral/dezentrale Architektur ausgewählt werden, in der sowohl alle Teile des Steuerungssystems auf den zentralen Rechner ausgelagert und die Regelungsschleifen über den Bus geschlossen sind als auch alle Regelungsaufgaben dezentral auf die Technologieplattformen verlagert sind und auf dem zentralen Rechner eine allgemeine Steuerung, eine Überwachung und ein HMI realisiert wird.

Der große praktische Nutzen liegt in der Entwicklung und Realisierung der neuen Technologieplattform (Hardware) und ihr unterstützendes Softwarekonzept bzw. ihre Softwarewerkzeuge. Die neue Technologieplattform unterstützt die effiziente, echtzeitfähige Kommunikation und besitzt spezifische Rechenstrukturen für die Implementierung der Regelung, Steuerung, Automatisierung, HMI usw. Diese Eigenschaften haben eine schnelle und effiziente Erweiterung des humanoiden Roboters ROTTO, um zusätzliche komplexe Funktionen der Mensch-Roboter-Interaktionen, ermöglicht.

Kapitel 4

Aspekte der Mensch-Roboter-Interaktion

Der anthropomorphe Roboter ROTTO besteht aus einem, komplexen mechatronischen System. Seine elektromechanische Konstruktion ist vielgliedrig und hochflexibel und sollte über eine entsprechende Steuerung verfügen. In diesem Kapitel wird zuerst eine Gliederung der Robotersteuerung, bezogen auf die mechanische Struktur sowie auf die Aufgabenebene, präsentiert. Danach wird auf die einzelnen aufgabenspezifischen Elemente der Robotersteuerung eingegangen. Ein besonderer Schwerpunkt dieser Arbeit ist die Entwicklung des Kommunikationsverhaltens des Roboters. Dessen steuerungsspezifische Aspekte werden im Unterkapitel 4.2 beschrieben.

4.1 Aufgaben- und Strukturhierarchie der Robotersteuerung

Der Roboter ROTTO wurde anthropomorph aufgebaut. Dies betrifft nicht nur die unteren Extremitäten sondern auch die oberen Extremitäten (Torso, Kopf sowie die Proportionen zwischen den einzelnen Robotergliedern). Der Roboter sollte der Untersuchung mehrerer Bewegungs- sowie Kommunikationsaufgaben dienen. Um dies zu ermöglichen, wurde für den Roboter ein Steuerungssystem entwickelt, das eine einheitliche Herangehensweise an allen Steuerungsebenen gewährleistet. Ähnliche komplexe, einheitliche Steuerungssysteme wurden in [13, 16] behandelt.

4.1.1 Aufgabenspezifische Ebenen der Robotersteuerung

Eine übersichtliche Darstellung der Steuerungsstruktur, sowie ihre Verteilung in unterschiedliche Steuerungsebenen entsprechend den Roboteraufgaben, ist im Bild 4.1 zu sehen. Die erste Steuerungsebene befasst sich mit den Kontaktaufgaben sowie den Aufgaben des Gleichgewichthaltens bzw. der Ausbalancierung. Eine detaillierte und ausführliche Beschreibung der Kontaktaufgaben und ihren Lösungen ist in [31] und [48] dargelegt. Da die Ermittlung der Kontaktkräfte einen entscheidenden Faktor für die Implementierung der grundlegenden Bewegungsaufgaben und insbesondere für die Berechnung von Kriterien des Gleichgewichtes [48] darstellt, ist jeder Roboterfuß mit einem sechsdimensionalen Kraftsensor ausgerüstet.

Die zweite Steuerungsebene ist übergeordnet und befasst sich mit den Aufgaben der Fortbewegung des Roboters: dem statischen und dynamischen Gehen. Eine detaillierte und ausführliche Beschreibung des statischen Gehens, untersucht am Roboter ROTTO, wurde in [48, 43] veröffentlicht. Das dynamische Gehen des Roboters wurde ebenfalls intensiv untersucht. Zuerst wurde in [48] die ballistische Roboterbewegung, wobei das mechanische System sich ausschließlich entlang den energieeffizienten und natürlichen Flug- bzw. Bewegungsbahnen bewegt, detailliert betrachtet. Anschließend wurde in [68, 51] ein anderer Ansatz zur Implementierung des dynamischen Gehens beleuchtet, in dem die Antriebe mit serieller Elastizität

für die Theorie der virtuellen Kräfte eingesetzt wurden. Virtuelle Kräfte wirken auf das mechanische System und bringen dieses, unter Einhalten eines gewissen Kriteriums, das oft der Energie des mechanischen Systems entspricht, in Bewegung. Das Steuerungssystem formt die virtuellen Kräfte in reellen Antriebsmomenten um. Zu diesem Zweck sollte und ist der Roboter zusätzlich zur Roboterfußkraftsensoren mit einem Gleichgewichtsapparat ausgestattet. Der Gleichgewichtsapparat beinhaltet ein dreidimensionales Gyroskop und einen dreidimensionalen Beschleunigungsmesser.

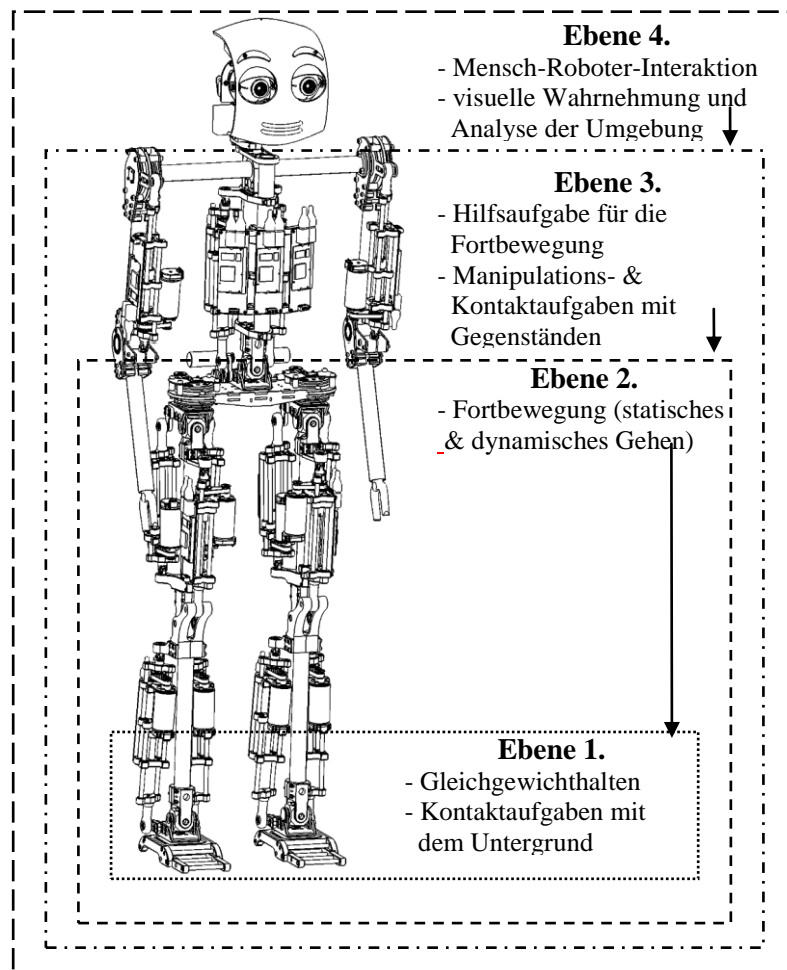


Bild 4.1 – Ebenen der Steuerung des Roboters ROTTO

Die dritte Steuerungsebene ist übergeordnet und befasst sich mit den Aufgaben der Manipulation, der mechanischen Interaktion mit der Umgebung mittels der Hände sowie der Bewegung des Oberkörpers, zwecks der Implementierung einiger Bewegungsmuster aus dem Bereich der Roboter-Mensch-Kommunikation. Dafür sind nicht nur das anthropomorphe Aussehen des Roboters, sondern auch die Bewegungen des Oberkörpers während eines Dialoges (Drehen in Richtung des Gesprächspartners etc), aufgrund der erhöhten Akzeptanz wichtig. Auf dieser Ebene wird außerdem das dynamische Gehen, durch die Bewegung des Oberkörpers, vervollständigt [48, 49, 50, 51]. Beispielsweise wird ein Kompensationsmoment, durch die entsprechende Regelung der Brust- und Schultergelenke, erzeugt, wenn ein Drehmoment im Fußgelenk, wegen des begrenzten Stützpolygons, während des Rollens des Roboterfußes

über den Fußrand zu groß wird. Auf diese Weise wird der dynamische Bereich des Gehens erweitert. Einige Untersuchungen zu diesem Thema sind in [51] beschrieben.

Als Helfer des Menschen, kann der Roboter z.B. im Haushalt beim Putzen, Waschen, Kochen usw. verschiedene Manipulationsaufgaben durchführen. Hier kommen die Roboterarme und die Umgebungssensoren, wie Videokameras, zum Einsatz. Die Beschreibung und die Forschung anhand der Manipulationsaufgaben für die Roboter wurden in [69] detailliert beschrieben. Eine Weiterentwicklung sowie Verbesserung der Roboterarme von ROTTO liegt außerhalb des Bearbeitungsbereiches der vorliegenden Arbeit und ist für die nachfolgenden Forschungsaktivitäten vorgesehen.

Die vierte und gleichzeitig oberste Ebene befasst sich mit Aufgaben, wie der Echtzeitanpassung der Roboterbewegung mittels visueller Sensoren sowie der Mensch-Roboter-Interaktion. Während der Fortbewegung des Roboters werden viele Umgebungsparameter im mathematischen Bewegungsmodell entweder nicht berücksichtigt oder vereinfacht. Dazu gehören nicht-ideale Böden, äußere Kräfte und Einwirkungen (Wind, Stoß) usw. Infolgedessen kann der Roboter seine Laufbahn bzw. Ziele verlieren. Der Mensch kann solche Ablenkungen, aufgrund der Information der Sensoren, wie Gleichgewichtsapparat und visuelle Sensorik, ausregeln. Roboter ROTTO ist ebenfalls mit visuellen Sensoren (Videokameras) ausgestattet und kann demnach seine Laufbahn beschreiten und seine Ziele anvisieren. Die Untersuchung dieser Aufgabe am Roboter ROTTO liegt ebenfalls nicht im Erarbeitungsbereich dieser Arbeit und soll in späteren Forschungen untersucht werden.

Die Fähigkeit zur Mensch-Roboter-Interaktion erhöht die Akzeptanz eines Roboters und erweitert seinen Einsatzbereich. Die Konstruktion des Roboters ROTTO erlaubt ebenfalls solche Art der Interaktion. Die Konstruktion seines Gesichtes beinhaltet viele spezifische Freiheitsgrade und Elemente und ermöglicht demzufolge die emotionale Kommunikation mit Menschen (s. Unterkapitel 4.2).

Die oben diskutierten Aufgabenebenen sind partiell aufgeteilt, weil in vielen Fällen die Ebenen nah zusammengebunden sind und eine strenge Aufteilung kaum möglich ist. Komplizierte Aufgaben beanspruchen die Zusammenarbeit mehrerer Steuerungsebenen, wodurch deren Grenzen verfließen.

4.1.2 Allgemeiner Aufbau der Steuerung

Im Rahmen dieser Arbeit wurde eine Vielzahl von Konzepten zum Aufbau des Steuerungssystems für den anthropomorphen Roboter untersucht. Das Ziel der Untersuchungen lag in der Entdeckung allgemeiner Strukturkomponenten sowie deren Wechselwirkungen. Die in [13, 16, 19, 48] eingesetzten Konzepte zum Aufbau der Robotersteuerung lassen sich in einer allgemeinen Struktur (s. Bild 4.2) aufgeteilt in drei Ebenen vereinheitlichen.

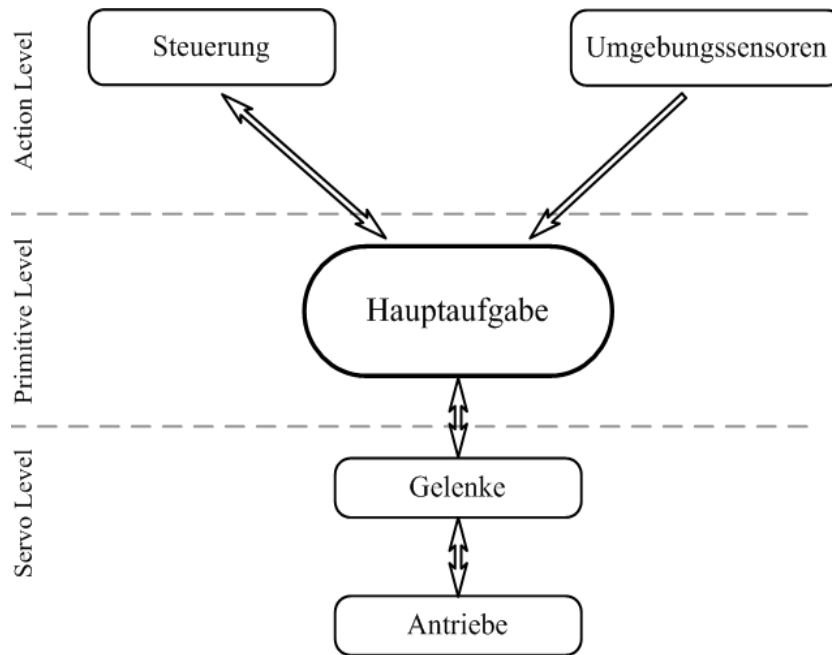


Bild 4.2 – Allgemeine Struktur zum Aufbau der Steuerung

Alle Roboterarbeiten äußern sich letztendlich in den koordinierten Bewegungen der Roboterantriebe. Die dafür zuständige Steuerungsebene *Servo Level* teilt sich in Substrukturen *Antriebe* und *Gelenke*. Die Substruktur *Antriebe* umfasst die mechatronischen Antriebskomponenten mit der entsprechenden Sensorik und Aktorik, die Steuerungs- und Regelalgorithmen sowie ihre softwaretechnische Abstraktion für die oberen Ebenen. Diese Ebene gewährleistet die notwendige Hardwareunabhängigkeit von den verwendeten Komponenten und die Modularisierung der motorischen und sensorischen Funktionen im Roboter. Die Substruktur *Gelenke* beinhaltet die abhängigen Freiheitsgrade sowie die entsprechenden Koordinatentransformationen in den separaten Robotergelenken.

Auf der Steuerungsebene *Primitive Level* werden die Algorithmen zur Lösung der Basisaufgaben platziert. Diese bilden die Bausteine komplexerer Roboterarbeiten und können separat untersucht und optimiert werden. Diese Schicht besitzt ein hohes Grad an Autonomie und erlaubt die parallele Abwicklung mehrerer Roboterarbeiten.

Die Steuerungsebene *Action Level* versorgt die untergeordneten Schichten mit Umgebungsinformationen und den Befehlen (Sollwertvorgabe, Stimuli, u.a.) vom Steuerungssystem bzw. Operator. Unter der Substruktur *Umgebungssensoren* wird nicht nur die Informationserfassung aus der Umgebung sondern auch die anschließende Bearbeitung dieser Information verstanden. Zu diesen Sensoren zählen: Fußkraftsensoren, Gyroskope, Beschleunigungssensor, Videokameras, Radars, Mikrofonen, u. a.. Die Informationsbearbeitung umfasst die Berechnungen, wie die Berechnung des Zero-Moment-Points aus den Kräften und Momenten des Gleichgewichtspunkts, die Berechnung der Körperneigungswinkel und die Laufrichtung aus den Daten des Gyroskops und des Beschleunigungssensors sowie die Erfassung der Ziellage aus den Kamerabildern u.a.

Jede Aufgabe soll ein Ziel erreichen, daher wird eine bestimmte sinnvolle Sollwertvorgabe benötigt. Ob dies die Fernsteuerung eines Operators, die vorbestimmte Bewegungstrajektorie, oder die Stimuli der Mensch-Roboter-Interaktion ist, wird die Unterstruktur *Steuerung* deutlich machen.

Die vorgeschlagene Struktur schafft die Voraussetzung für eine einfache Modularisierung und damit auch einfache Entwicklung und Programmierung der einzelnen Komponenten für das komplexe Steuerungssystem. Es bietet außerdem eine gewisse Autonomie und ermöglicht damit eine weitgehend rückwirkungsfreie Modifizierung der einzelnen Teile des Gesamtsystems.

Die vorliegende Arbeit widmet sich in erster Linie dem strukturierten Aufbau der Steuerung für den Oberkörper des anthropomorphen Roboters ROTTO, als Beispiel eines komplexen Automatisierungssystems. Der Roboter Oberkörper beinhaltet mehrere Aufgaben, wie die Emotionssteuerung, kompensatorische Aufgaben und die Fortbewegung zum Ziel (mit einer Abgleichung mittels Videosignals), welche durch den Einsatz der analysierten Steuerungsstrukturen in den folgenden Unterkapiteln gelöst werden.

4.2 Aufgaben der Mensch-Roboter-Interaktion

Roboter werden zunehmend nutzbringender und finden mehr Akzeptanz in der menschlichen Gesellschaft. Daraus entsteht der Bedarf zur Mensch-Roboter-Interaktion, welche für die Menschen natürlich und einfach werden soll. Emotionen, Gesten, Gesichtsausdrücke u.ä. sind die üblichen Informationsträger in einem menschlichen, alltäglichen Gespräch. Deren Beherrschung bringt deshalb den Roboter auf neue höhere Ebene der Mensch-Roboter-Interaktion und erhöht seine „zwischenmenschliche“ Akzeptanz.

Ein der Hauptziele dieser Arbeit liegt in der Befähigung des Roboters ROTTO zur Mensch-Roboter-Interaktion und insbesondere in der Entwicklung und Implementierung von intuitiven und naturrellen Emotionen und Gesten, wie Interessiertheit oder Gleichgültigkeit, Freude oder Trauer, Sympathie oder Schreck.

4.2.1 Theory of Mind

Um die Emotionssteuerung für den Roboter ROTTO zu entwickeln, wurde zuerst auf die menschliche Fähigkeit der Empathie, d.h. eine Annahme über die Meinungen, Absichten, Bedürfnisse, Erwartungen in anderen Personen vornehmen zu können und diese in der eigenen Person wieder zu erkennen, eingegangen bzw. auf die Theory of Mind. Die zwei populären Theorien [4] zeigen die Vernunftentwicklung eines Kindes. Auf Grund der sozialen Fähigkeiten des Kindes, die in der Theory of Mind veranschaulicht sind, werden die Theorieteile auf die Emotionssteuerung für den Roboter ROTTO projiziert, um ähnliche Fähigkeiten beim Roboter zu erreichen.

Die Struktur der ersten Theorie - der Theorie von Leslie [70] - veranschaulicht das Bild 4.3. Laut Leslie ist die Umgebung, basierend auf ihrer Kausalstruktur, in drei Typen der Ereignisse unterteilt: *mechanical*, *actional* und *attitudinal* Aktionen. Leslie vermutet eine evolutionäre Herkunft der selbständigen Module, die diese Aktionen behandeln.

Das *Theory of Body* Modul (*ToBY*) behandelt alle Ereignisse mit mechanischem Grund, die sich mit den mechanischen Gesetzen beschreiben lassen. Das *ToBY* Modul verwirklicht bei einem Kind das Begreifen eines physikalischen Objektes. Das Ziel dieses Moduls ist die Beschreibung der Welt und der Umgebung in die Begriffe der *Mechanik* für die eweiligen physikalischen Objekte und Prozesse. Bei Menschen nimmt *ToBY* die Information mittels zweier visueller Quellen wahr: der einfachen dreidimensionalen *bewegungsbasierten* Objektdarstellung und der dreidimensionalen Objektdarstellung des hochentwickelten *kognitiven und visu-*

ellen Systems. *ToBY* veranschaulicht das raumzeitliche Gefühl eines Ereignisses, welches bei einem Kind angeboren oder während ersten sechs Lebensmonaten schnell entwickelt ist.

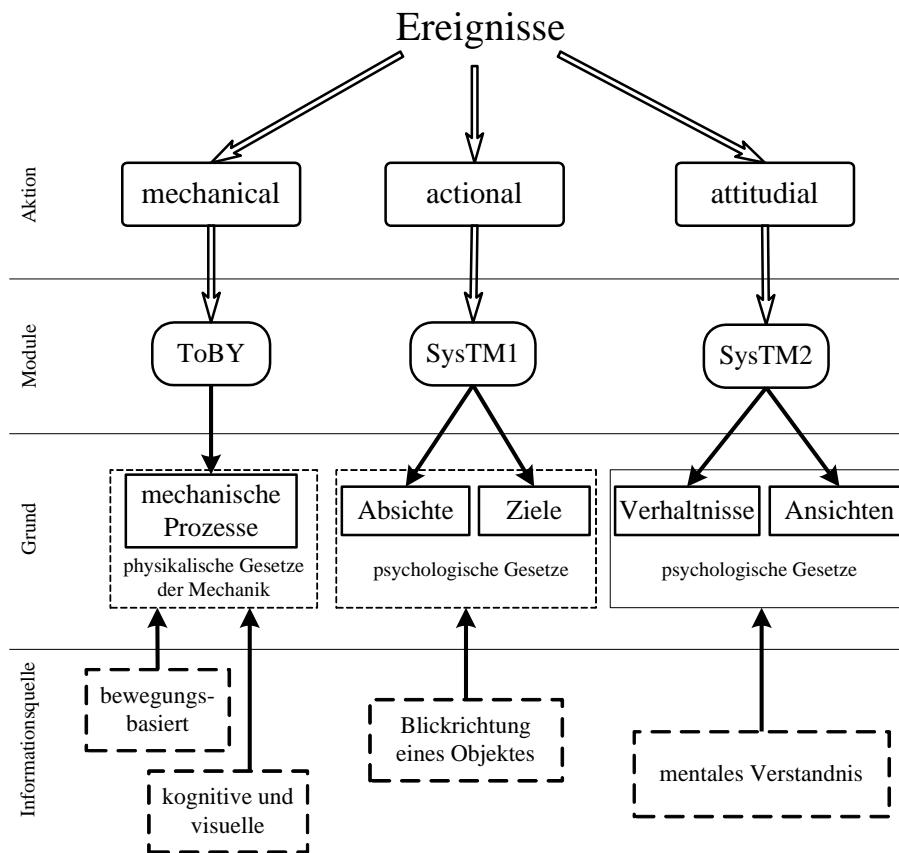


Bild 4.3 – Modell der Theory of Mind von Leslie

Das zweite Modul in Leslie's Theory of Mind ist das *System-1* (*System Theory of Mind – SysTM1*), welches die Ereignisse in Hinblick auf die Auswirkungen der Aktionen (Absichten und Zielen) eines Wesens interpretiert. Das *SysTM1* Modul greift die psychologischen Gesetze auf, nach denen der Mensch sich richtet und die mit Actionaktivität tätig sind. *SysTM1* beinhaltet die Objekte und ihre zweckgebundenen Aktionen, die zum Erreichen eines Ziels vorgenommen werden müssen. Die Grundauffassung der Aktionen, wie beispielweise die Vermeidung eines Vorganges oder seine Annäherung, wird in *SysTM1* aufgezeigt. Dieses System der Ermittlung von Absichten und Zielen bei Kindern eignet sich ab dem sechsten Lebensmonat zudem nimmt *SysTM1* die Information häufig, aus der Perspektive eines Interaktionspartners wahr.

Das dritte von Leslie's Modulen ist *System-2* (*SysTM2*), welches die Ereignisse interpretiert, unter Betrachtung der Meinungen bzw. Haltungen (*attitudes*), Verhältnisse und Ansichten eines Wesens Das *SysTM2* Modul umfasst die psychologischen Gesetze, die durch die Verhältnisaktivität tätig werden. *SysTM2* dient der Darstellung der Ansichten und die Beeinflussung der psychischen Zustände durch zweckgebundenes Verhalten. Dieses System entwickelt sich bei Kindern sukzessiv innerhalb des 18. bis 48. Lebensmonat. *SysTM2* verwendet die Metarepresentation, die die Wahrheit eines Ereignisses in Kombination mit dem mentalen Verständnis dieses Ereignisses (die Sinnesorgane können hierbei falsche Informationen aufnehmen) wahrzunehmen erlaubt. Das *SysTM2* System gibt einem Mensch zu verstehen, dass andere Menschen solche Ansichten haben können, die mit den eigenen Ansichten oder mit dem beobachtenden Ereignis nicht übereinstimmen.

Im Unterschied zu Leslie hat die Theory of Mind von Baron-Cohen [71] zwei Informationsquellen. Laut der Baron-Cohen's Theorie wird die Welt mit einem Komponentensatz („Mindreading system“) wahrgenommen, der aus vier verschiedenen Modulen (Bild 4.4) besteht.

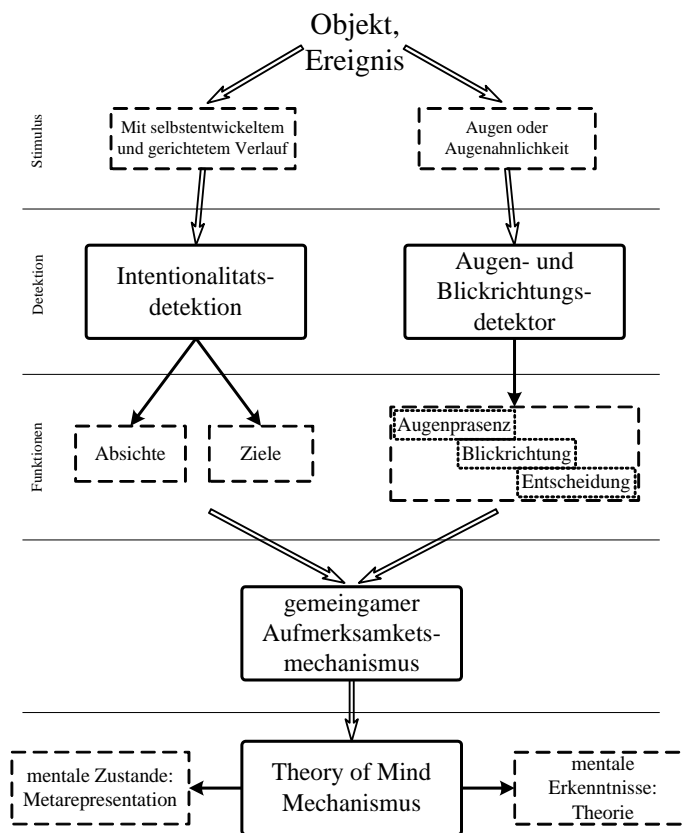


Bild 4.4 – Modell der Theory of Mind von Baron-Cohen

Das erste Modul behandelt die *selbstentwicklungsbasierte* Stimuli mit den Begriffen des mentalen Willenszustands von *Absichten und Zielen*. Dieser Modul ist als *Intentionalitätsdetektion (ID)* bezeichnet, der die Grundlage der Prozessentwicklung von der Vermeidung eines Vorganges oder Annäherung an dem repräsentiert. Das *ID* Modul basiert sich nur auf den Stimuli mit dem *selbstentwickelten* Verlauf und unterscheidet die Stimuli bzw. Initiatoren eines Lebewesens (Agent) von den Stimuli eines leblosen Objektes. Laut Baron-Cohen ist der *ID* eine angeborene Fähigkeit des Kindes.

Das zweite Modul beinhaltet nur die Stimuli in Form der Augen und Augenähnlichkeiten, und ist fähig die Blickrichtung zu ermitteln. Dieses Modul wird als *Augen- und Blickrichtungsdetektor (ABD)* bezeichnet und hat drei basische Funktionen. Die erste sucht nach augenähnlichen Stimuli im visuellen Raum (z.B. Augen einer Person) und determiniert die *Augenpräsenz*. Die zweite bestimmt die Blickrichtung der gefundenen Augen, und die dritte Funktion des *ABD* interpretiert die Augenblickrichtung in eine Information für die Wahrnehmung. Information wird auf Weise durch „er sieht mich“ oder „er schauet nicht auf mich“ repräsentiert. Der *ABD* entwickelt sich bei einem Kind bis zum neunten Lebensmonat.

Der *gemeinsame Aufmerksamkeitsmechanismus (GAM)* bildet das dritte Modul. Dieses nutzt die Repräsentationen von *ID* und *ABD* Modulen und konstruiert eine ternäre Repräsentation auf der Art und Weise wie zwei Akteure integriert werden: „Ulrich sieht: ich sehe eine Frau“. Der *GAM* ermöglicht es dem Agenten zu begreifen, dass er und ein anderer Agent zum selben

wahrnehmbaren Ereignis oder Objekt gehören. Dieser Zustand der *gemeinsamen Aufmerksamkeit* mündet in der Nachvollziehung einer Repräsentation (*ID*) innerhalb einer anderen Repräsentation (*ABD*). *GAM* ist fähig die Ausgangsrepräsentation vom ersten Modul am Eingang des zweiten Moduls überzuleiten, so dass die Blickrichtung als Willenszustand der *Ziele (Absichte)* interpretiert werden kann. Diese Möglichkeit gestattet dem *GAM* die Bildung der komplexen Repräsentation eines Ereignisses. Der *GAM* entwickelt sich bei einem Kind zwischen dem 9. und 18. Lebensmonat.

Das letzte Modul ist das *Theory of Mind Mechanismus (ToMM)*. Dieses repräsentiert die epistemischen mentalen Zustände in anderen Agenten oder Objekten, damit eigene Erkenntnisse der mentalen Zustände in kohärenter Gesamtheit bzw. in nutzbarer Theorie verbunden werden. Die ternäre Repräsentation von *GAM* wird in die *Metarepräsentation* umgewandelt. Die Entwicklung von *GAM* fängt bei einem Kind ab dem 18. Lebensmonat an und endet vollständig im 48. Lebensmonat.

Beide Theorien zeigen komplizierte, mentale Zusammenhänge eines Lebewesens, das in unterschiedlichen Situationen auf unterschiedliche Ereignisse und Objekte reagiert. In folgendem Unterkapitel werden beide Theorien zur Auslegung der Mensch-Roboter-Interaktion am Roboter ROTTO angewandt.

4.2.2 Steuerung für den Roboter ROTTO

Laut der Theorien of Mind wird die Umgebung des Lebewesens in verschiedene Ereignisse und Prozesse unterteilt. Diese werden in der Struktur der Robotersteuerung im Bild 4.5 als *externe Stimuli* bezeichnet, welche einen unmittelbaren Einfluss auf das Roboterverhalten haben. Genau wie bei einem Lebewesen, wird die Robotersteuerung von internen Ereignissen stimuliert. Die *internen Stimuli* soll die Absichten und die Ziele des Roboters (wie *SysTMI* Modul beim Leslie und *Detektionsebene* beim Baron-Cohen) zum Ausdruck bringen.

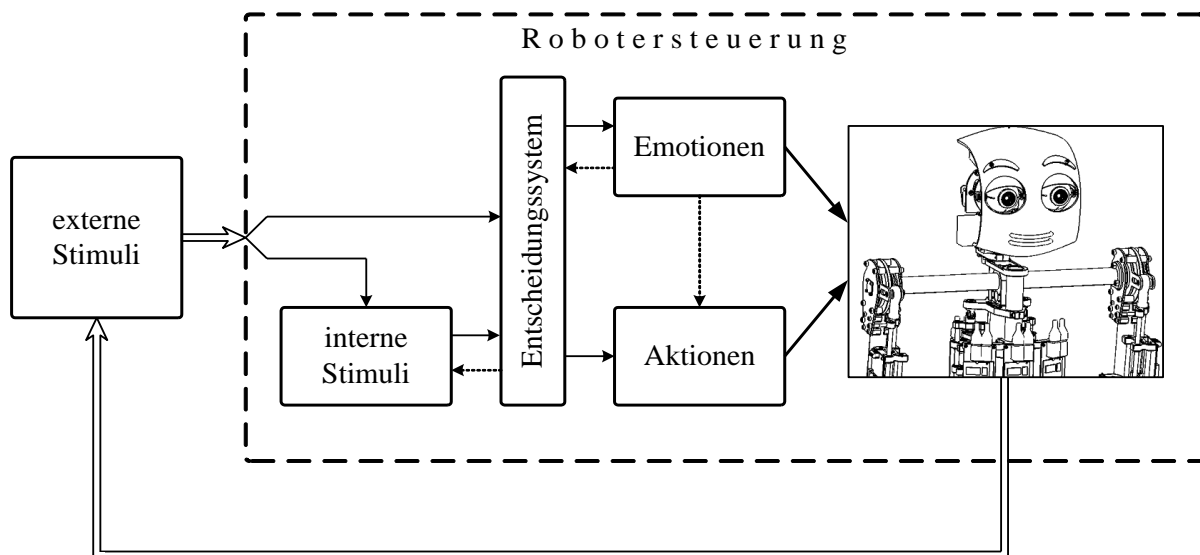


Bild 4.5 – Allgemeine Struktur der sozial-emotionalen Steuerung für den Roboter ROTTO

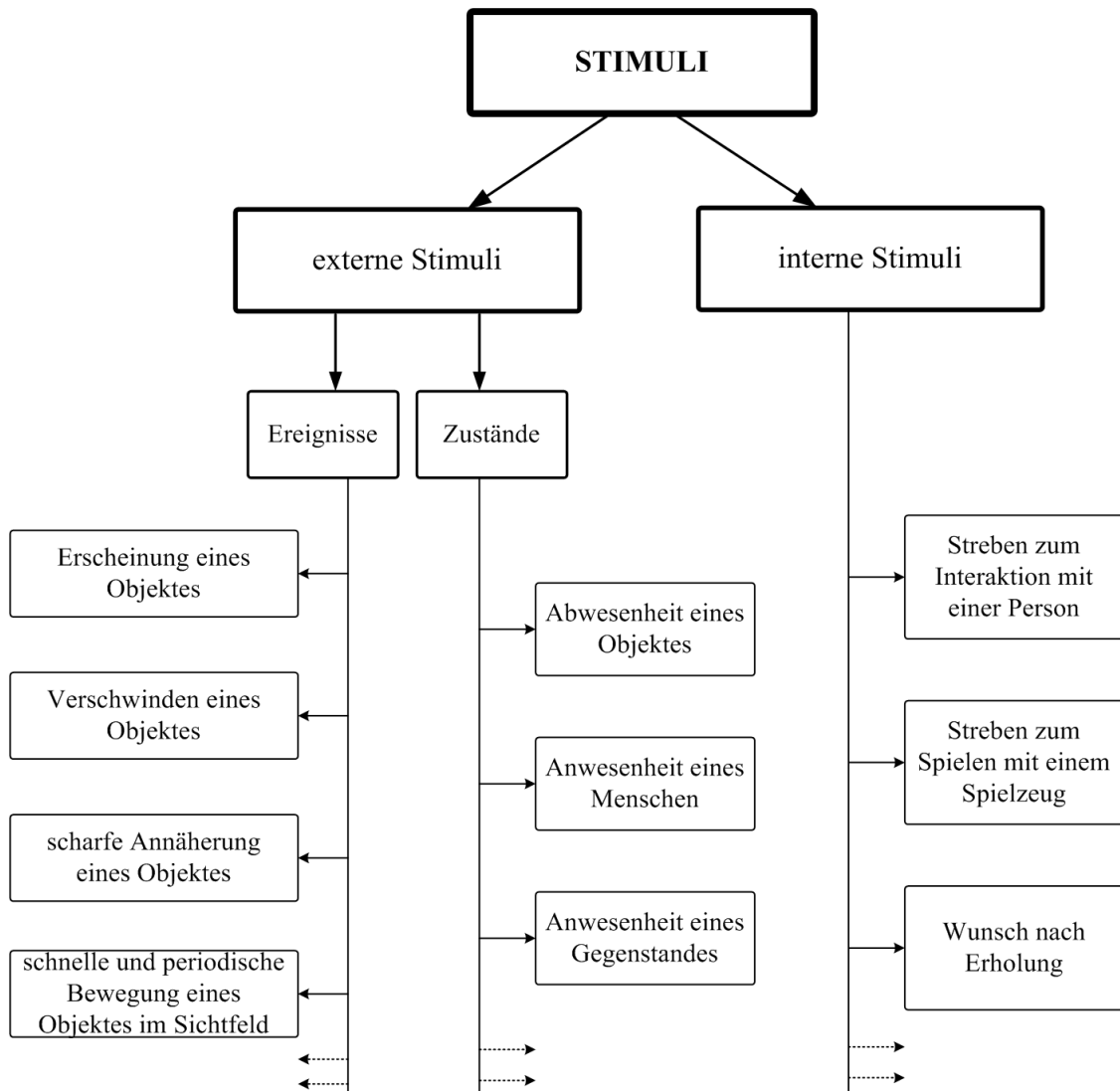


Bild 4.6 – Struktur der Stimuli in der sozial-emotionalen Robotersteuerung

Das Verhalten des Roboters besteht aus *Emotionen* und *Aktionen*. Es bezeichnet alle physikalischen, bewussten und unbewussten Bewegungen, die als Antworten auf die Stimuli entstehen. Eine Verbindung zwischen der Stimuli und dem Roboterverhalten ist im Entscheidungssystem, dem Kern der kommunikationsfähigen Steuerung, ausgeführt, weil hier alle Wechselwirkungen zwischen den Stimuli und dem Roboterverhalten (als Charakter oder Persönlichkeit) impliziert sind.

4.2.2.1 Stimuli

Das Bild 4.6 veranschaulicht ausführlich die Motivationsquellen, die das Roboterverhalten unmittelbar beeinflussen und die als *Stimuli* bezeichnet werden. Wie oben erwähnt, werden die *Stimuli* auf die *externe* und die *interne* aufgeteilt. Durch die externen Stimuli, nimmt die Emotionssteuerung des Roboters ROTTO die Welt in physikalischen Ereignisse und Zustände (wie der *ToBY* Modul beim Leslie und der Intensitätsdetektor beim Baron-Cohen) wahr. Im Gegensatz zu den externen, stellen die internen Stimuli den eigenen Zustand, wie z.B. Absichten und Ziele, dar und ergeben daher ein Selbstmotivationsystem.

Der Auswahl der externen Stimuli für die Robotersteuerung beruht auf den Erfahrungen aus den Untersuchungen der Aufmerksamkeit bei Säuglingen und Kleinkindern. Viele Forschungsaktivitäten haben sich dieser Fragestellung gewidmet: Welche Eigenschaften eines Objektes erregen ein natürliches Interesse beim Kind? Newson sagt, dass bewegliche, buntfarbige und laut klingelnde Gegenstände meistens die Aufmerksamkeit des Kleinkindes erregen [72]. Laut Trevarthen [73] sind Kleinkinder fähig, die bunten Farben (wie rot, grün, blau und gelb) zu unterscheiden. Besonders interessant ist die rote Farbe, weil die der Farbe der Mutterlippen ähnelt. Die Kleinkinder sind besonders empfindlich für Bewegungen und schenken demnach den beweglichen Spielzeugen im Unterschied zu den einfach buntfarbigen viel mehr Aufmerksamkeit. Darüber hinaus erregt eine schnelle und periodische Bewegung eines Spielzeuges ein besonders starkes Interesse. Nach dem ersten Lebensmonat beginnt die Entwicklung der Tiefenwahrnehmung [74]. Die Intensität eines Stimulus kann stark oder schwach für ein Kleinkind sein. Es versucht diese Intensität mittels der eigenen Reaktion zu regeln. Beispielsweise kann eine laute Stimme einen Säugling zum Weinen bringen, er drückt die Augen zu und erzeugt ein Signal, damit die Stimme gesenkt wird [75].

Die Ereignisse und Prozesse, die beim Kleinkind das Interesse erregen, bilden die *externen Stimuli* für die Emotionssteuerung. *Externe Stimuli* sind gekennzeichnet durch den Doppelsprung. Einerseits werden die schnellen oder plötzlichen Veränderungen („Ereignisse“) in der wahrnehmbaren Umgebung abgesondert. Zu diesen *Ereignissen* zählen verschiedene physikalische Vorgänge mit den Objekten. Unter einem *Objekt* wird eine Person (ihr Gesicht) oder ein bunter farbiger Gegenstand (Spielzeug) verstanden (hier in der Emotionssteuerung für den Roboter ROTTO).

Im Bild 4.6 werden vier externe Grundereignisse hervorgehoben, die im Fokus der Emotionssteuerung stehen. Die *Erscheinung eines Objektes* wird über den Anfang der neuen Interaktionsphase signalisiert und die Aufmerksamkeit des Roboters damit aktiviert. Beispielsweise kann eine Person zum Roboter kommen, ihm gegenüber stehen und eine Antwort bzw. Reaktion erwartet. Gleichzeitig ist dies ein Ereignisstimulus für die Steuerung, die die entsprechende Aktion hervorruft. Wenn nach der Interaktion das Objekt aus dem Robotersichtfeld verschwindet, ist diese Phase genauso wichtig, wie seine *Erscheinung*. Das *Verschwinden eines Objektes* signalisiert das Ende oder die Unterbrechung der Interaktionsphase. Die Steuerung soll abhängig vom internen Zustand (bzw. Stimuli) entscheiden, ob der Roboter entweder dem verschwundenen Objekt keine Aufmerksamkeit mehr schenkt oder das Objekt zurückt.

Weitere *Ereignisse* werden aus den Forschungen der Aufmerksamkeit bei Kleinkindern übernommen. Auf diese Weise werden die scharfe Annäherung und die schnelle periodische Bewegung eines Objektes im Robotersichtfeld in Betracht gezogen. Die beiden Vorgänge sind für das Kleinkinderspiel charakteristisch und werden in die Steuerung eingezogen.

Wie oben dargestellt, sind diese vier beschriebenen *Ereignisse aus der externen Stimuli* die Grundlage der Umgebungswahrnehmung für die Emotionssteuerung des Roboters. Dank des modularen Aufbaus der Steuerung können zusätzliche *Ereignisse* addiert werden.

Die zweite Variation *der externen Stimuli* wird als „Zustände“ bezeichnet und beinhaltet die aktuelle Umgebung nach einem *Ereignis*. Nach einer *Erscheinung* eines Objektes kommt der Stimulus „Anwesenheit“ zum Tragen. Zum Beispiel kommt der Ereignisstimulus (*Erscheinung einer Person*) nach der Erscheinung einer Person im Robotersichtfeld zustande. Danach hört dieser Ereignisstimulus auf und es entsteht simultan der Stimulus *Anwesenheit eines Menschen*. Dieser regt den Roboter zum entsprechenden Verhalten an. Ebenso führt die Erscheinung eines farbigen Gegenstandes zur Entstehung des Stimulus *Anwesenheit eines Ge-*

genstandes inklusive nachfolgendem Roboterverhalten. Falls der Roboter keine Person oder keinen farbigen Gegenstand sieht, entsteht der Stimulus *Abwesenheit eines Objektes*. Dieser kann den Roboter zur Suche nach einem Objekt anregen. Darüber hinaus sind einige *Ereignisse* und *Zustände* stark miteinander verbunden. Die *externen Stimuli* bilden daher ein Zusammenspiel der *Ereignisse* mit den *Zuständen*.

Die rechte Abzweigung *der Stimuli* im Bild 4.6 sind die internen Stimuli. Diese stellen einen internen Roboterzustand bzw. Roboterwünsche dar. Für die Emotionssteuerung des Roboters ROTTO sind folgende einfache interne Grundstimuli gewählt: *interaktiv*, *initiativ* und *erhol-sam*.

Interaktive und initiative Stimuli ändern sich in der Intensität und mit der Zeit (s. Bild 4.7). Der Stimulus dieser Art besitzt sowohl *externe* als auch *interne Anregungen*. Einerseits ändert er seinen *Erregungsgrad* gemäß der entsprechenden externen Anregung - dies könnten ein oder mehrere externe Stimuli sein - andererseits beruhigt sich der *Erregungsgrad* im Laufe der Zeit mittels der internen Anregung. *Die interne Anregung* symbolisiert eine Begrenzung des *Erregungsgrades* des Stimulus. Diese Begrenzung hängt vom *Erregungsgrad* und der *Zeit* ab. Bei hoher oder niedriger Erregung ändert sich die Begrenzung kontinuierlich bis zum Gleichgewicht und beliebige externe Anregungen sind unfähig diese Begrenzung zu überschreiten. Wenn die Begrenzung das Gleichgewicht erreicht hat, startet der Prozess in umgekehrter Richtung und die *interne Anregung* erweitert ihre Begrenzungsebene bis hin zur hohen und niedrigen Erregung. Auf diese Weise wird der Gewöhnungseffekt eines Stimulus realisiert.

Durch *den interaktiven Stimulus* strebt ROTTO nach der Interaktion mit einer Person, da dieser beim Roboter sein Interesse zum Dialog erregt. Bei der *Anwesenheit einer Person* nimmt *der interaktiver Stimulus* positive Erregungen auf und kann das positive emotionale Verhalten hervorrufen. Beim plötzlichen Verschwinden dieser Person bleibt die Erregung positiv und eine Suche nach dieser Person kann hervorrufen werden. Laut des anregend-temporären Ursprunges (bzw. Intensität-Zeit-Abhängigkeit) gewöhnt sich der *interaktive Stimulus* nach einer Zeit an die externe Anregung (*Anwesenheit einer Person*) und es wird keine Person weiter wahrgenommen. Nach dem Ablauf der Zeit erneuert der *interaktiver Stimulus* jedoch seine Fähigkeit zur Wahrnehmung. Dieser ist dem Augen- und Blickrichtungsdetektor des Baron-Cohens ähnlich (Bild 4.4).

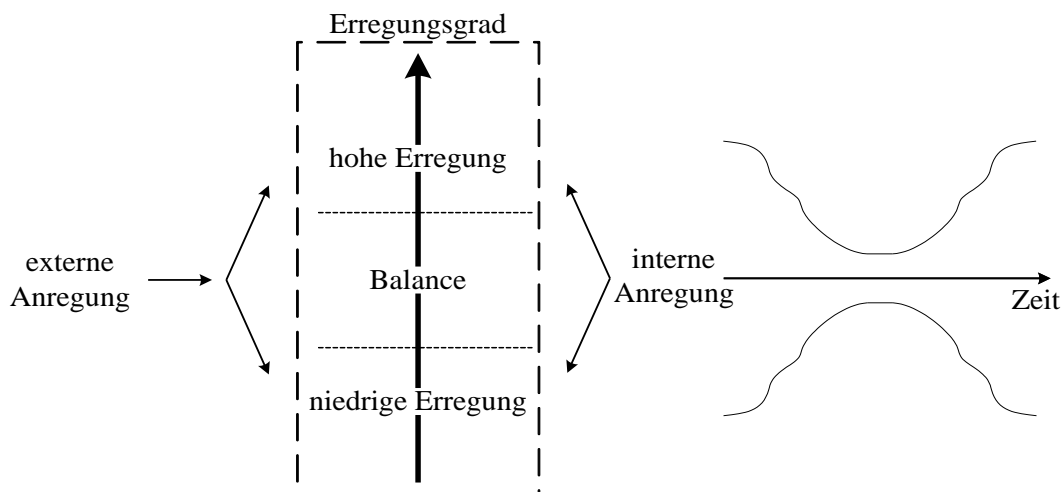


Bild 4.7 – anregend-temporärer Ursprung des internen Stimulus

Der initiative Stimulus wird im ROTTO nach dem Spiel mit einem Spielzeug aktiviert. Er erregt beim Roboter ein Interesse nach den bunten farbigen Gegenständen. *Der initiative Stimulus* besitzt ebenfalls einen anregend-temporären Ursprung und ähnelt daher dem Verhalten des *interaktiven Stimulus*.

Der erholsame Stimulus zeigt die Müdigkeit des Roboters an. Nachdem ROTTO im Laufe der Zeit ununterbrochen stark stimuliert wurde, strebt er durch diesen Stimulus nach Erholung. *Der erholsame Stimulus* hat einen *anregend-kumulativen Ursprung*, der in Bild 4.8 schematisch dargestellt ist. Sofern eine oder mehrere *externe Anregungen* wirken, wächst der *Erregungsgrad*, d.h. die Erregung wird akkumuliert. Je mehr Anregungen gegeben sind und je höher ihre Intensität ist, desto schneller wächst der Erregungsgrad. Dieser wird auf einer gewissen Ebene (hoher Erregung) begrenzt. Andererseits existiert eine Gegenwirkung, die den Erregungsgrad kontinuierlich minimiert. Die Intensität dieser Minimierung hängt von den *internen Anregungen* ab. Der Erregungsgrad kann bis zur *Nullebene* gesenkt werden. Die gesamte Erregung umfasst dabei die gesamte Einwirkung der externen und internen Anregungen. *Externe Anregungen* für den *erholsamen Stimulus* sind: die Roboteraktivität (*Aktionen*) und externe Belastungen (*externe Stimuli*) und sind daher die Müdigkeitsfaktoren. Interne Anregungen sind Faktoren, die für die Erholungsqualität verantwortlich sind. Für Menschen können dafür die Erholungsweise (z.B. Schlafen oder Stillsitzen), gesundheitlicher Zustand (gesund oder krank), Tageszeit (Tag oder Nacht) usw. aufgezählt werden. Einige von ihnen können für den *erholsamen Stimulus* hinsichtlich des Roboters modifiziert werden.

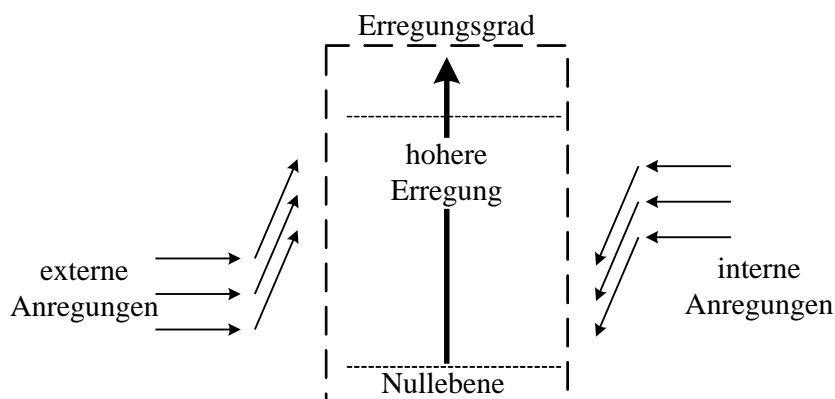


Bild 4.8 – anregend-kumulativer Ursprung des internen Stimulus

4.2.2.2 Das Verhalten

Das Verhalten eines Roboters ist ein wichtiger Teil der Mensch-Roboter-Interaktion. Der Roboter zeigt durch sein Verhalten den eigenen internen Zustand und eigene Ziele und Absichten. Durch ein natürliches Verhalten wird eine hohe Akzeptanz des Roboters erreicht. In der Studie von Masahiro Mori [2] über die sozialen Roboter Kismet, Merz und Leonardo [5, 6, 7] von MIT wird betont, dass das Verhalten mit der kontinuierlichen Bewegung den Roboter auf ein hohes Niveau der Akzeptanz bringt. Auf diese Weise wurde ein komplexes System des Verhaltens für ROTTO entworfen, das nicht nur Emotionen sondern auch unbewusste Bewegungen, die für das Lebewesen natürlich sind, beinhaltet.

Das Verhaltenssystem für den Roboter ROTTO wird in drei Gruppen unterteilt: Emotionen, bewusste und unbewusste Aktionen (bzw. Bewegungen). Das Bild 4.9 veranschaulicht jede Verhaltensgruppe und ihre jeweilige Struktur.

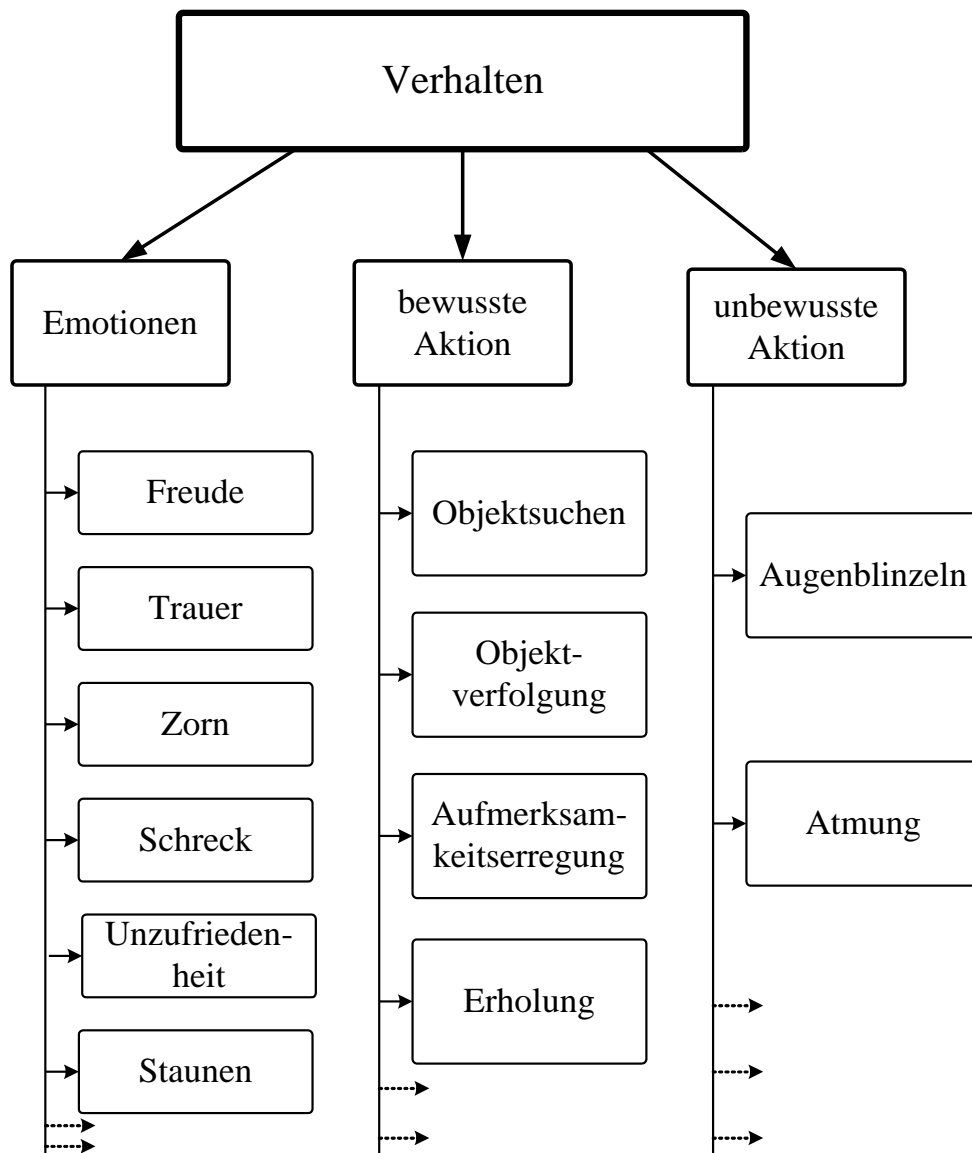


Bild 4.9 – Struktur des Verhaltens in der sozial-emotionalen Robotersteuerung

Die linke Abzweigung der Verhaltensstruktur sind Emotionen. Das emotionale Verhalten ist mit der Generierung und Darstellung bestimmter, auf dem Roboter ersichtlichen (Gesicht und Körper) Emotionen tätig. Für das emotionale Verhalten von ROTTO wurden „Grundemotionen“ gewählt. Obwohl in der wissenschaftlichen Literatur vielfältige menschliche Emotionen genannt werden, werden folgende Grundemotionen einstimmig aufgezählt: *Freude, Trauer, Zorn, Schreck, Unzufriedenheit* und *Staunen* [76, 77]. Als zusätzliche Emotionen werden Scham und Verachtung aufgeführt, aber die werden für die emotionale Steuerung vom ROTTO in Rahmen dieser Arbeit nicht mit eingezogen.

Jede dargestellte Emotion bildet ein einheitliches Modul. Er ist für die mechanischen Bewegungen der eigenen Emotion verantwortlich. Die Module sind unabhängig voneinander. Dank dieses modularen Aufbaus, gibt es eine Möglichkeit die zusätzlichen Emotionen in die Struktur der Emotionen einzuschließen. Jedes Modul wird durch seine Intensität (Bild 4.10) gesteuert. Dabei kann eine Emotion mit unterschiedlicher Stärke, die von der Intensität abhängt, veranschaulicht werden. Abhängig von der Intensität und dem eigenen Emotionstyp generiert das Modul die entsprechende Bewegung in Form der Koordinaten, die dem *Servo Level* als Sollwerte dienen.

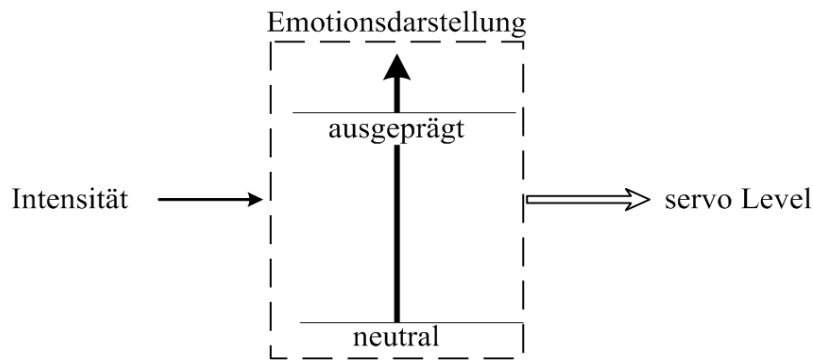


Bild 4.10 – Einheitliches Modul für eine Emotion

In Bild 4.11 wird veranschaulicht, wie die Emotion „Freude“ (in drei Intensitäten) auf dem neutralen Robotergesicht zustande kommt.



Bild 4.11 – Neutrales Gesicht (0%) und Emotion „Freude“ mit der mittleren (30% und 60%) und der ausgeprägten (100%) Intensität

Das Entscheidungssystem ist fähig mehrere Emotionen gleichzeitig zu generieren. Diese werden dazu zusammengemischt und infolgedessen ist ein komplexer Gesichtsausdruck erreichbar. Um die kontinuierliche Bewegung des Robotergesichtes zu gewährleisten, schwankt das Emotionssystem zwischen den vorhandenen Emotionen in einem engen Rahmen. Wenn das Entscheidungssystem nur eine Emotion zum Anzeigen erzeugt, wird ihre Intensität nur ein wenig schwanken. Das Verhaltenssystem wird zu den Emotionen anderer Aktionen hinzugefügt, um dadurch ein akzeptables Roboterverhalten zu erreichen.

Die mittlere Abzweigung der Verhaltensstruktur bezeichnet die *bewussten Aktionen*. Diese sind stets zweckgerichtet und durch die bestimmten Stimuli erregbar. Für jede bewusste Aktion ist ein unabhängiges Modul verantwortlich. Dieser realisiert die Bewegungen (bzw. generiert Sollwerte für Servo Level), die der gewünschten Aktion entsprechen und einem Mensch klar verständlich sind. Für den Roboter ROTTO wurden solche *bewussten Aktionen* ausgewählt, die dem Mensch absolut verständlich sind und die im Dialog zwischen der Mutter und ihrem dreimonatigen Kind zu beobachten sind. In [74] sind fünf Phasen der sozialen Interaktion beschrieben: *Initiation*, *gegenseitige Orientierung*, *Begrüßung*, *Spielphase* und *Abzug*.

In der Phase der *Initiation* will die Mutter ihr Kind zum Kontakt einladen. Sie platziert dazu ihr eigenes Gesicht vor dem Kind, bewegt sich intensiv und zeigt dem Kind ein buntes Spiel-

zeug, um seine Aufmerksamkeit zu erregen. Diese Phase kann das Kind durch murmeln oder schreien initiieren. In der folgenden Phase der *gegenseitigen Orientierung* wird die Aufmerksamkeit der Mutter und des Kindes aufeinander gerichtet. Das Kind nimmt seine Mutter wahr und ist für die folgende Interaktion bereit. In der *Begrüßung* ist daher die Kindesaufmerksamkeit gänzlich der Mutter gewidmet, welche die ihre Aktionen zur Aufmerksamkeitserregung reduziert und sie anlächelt dem Kind. Weiterhin folgt die *Spielphase*, in der beide leise miteinander Sprechen oder ein Spielzeug berühren. In der anschließenden Phase (*Abzug*) lenkt einer der beiden die Aufmerksamkeit um auf einen anderen Gegenstand oder eine neue Person - der andere bleibt jedoch für weitere Interaktionen bereit. Hier ist entweder ein neuer Versuch der *Initiation* möglich oder die Interaktion ist abzuschließen.

Zu den bewussten Aktionen von ROTTO gehören die ersten zwei Phasen: das „*Objektsuchen*“ und die *Aufmerksamkeitserregung* (die *Initiation* und die *gegenseitige Orientierung*). Die dritte Phase (*Begrüßung*) und der größte Teil der vierten (*Spielphase*) werden dem emotionalen Verhalten zugeordnet. Die *Objektverfolgung* wird ebenfalls der vierten Phase zugeordnet. In der *Abzugsphase* kommt schließlich die *Erholung* zum Einsatz.

Jede bewusste Aktion ist durch eigene Bewegungen gekennzeichnet. Bei der hoch erregten Anwesenheitsstimulus wird der Roboter nach einem Objekt oder einer Person suchen: der Kopf und die Augen drehen sich in die Seiten zusammen, um damit das Suchfeld zu erweitern und das gewünschte Objekt darin herauszufiltern. Eine Person (oder ein buntes Objekt) ist gefunden worden, nun wird die *Aufmerksamkeitserregung* aktiviert: der Roboter lädt die Person bzw. das Objekt zur Interaktion ein, er versucht ihre Aufmerksamkeit auf sich zu lenken, mittels eines auf sie gerichteten Blickes, einer Tonähnlichkeit und eines emotionale Lächelns. Diese Phase verläuft fließend in die Phase der *Begrüßung*, in der ROTTO durch die Emotionen interagiert wird. Die dritte Phase ist durch eine relativ kleine Laufzeit gekennzeichnet. In der nachfolgenden *Spielphase* (Interaktion) kommen die Emotionen zum Einsatz. ROTTO wird darin der eigene interne Zustand präsentiert und somit die Wünsche und Absichten emotional ausgedrückt. Darüber hinaus wird die *Objektverfolgung* (Beobachtung) aktiviert, um den Interaktionspartnerverlust zu verhindern. Wegen einer Bewegung des Interaktionspartners kann, der Roboter ihn aus dem Sichtfeld verlieren, deshalb müssen der Kopf und die Augen immer in Richtung des Dialogpartners ausgerichtet werden. Während der intensiven Interaktion (bzw. *Spielphase*) wird die Erregung des *erholsamen Stimulus* erhöht und beim Roboter wird die Müdigkeit durch langsame Bewegungen und der verringerten Intensität der Emotionen ersichtlich. Der Roboter geht in die *Abzugsphase* über, er strebt nach dem Erholungsverhalten, in dem er die Augen schließt, den Kopf senkt, und während der *Erholung* keine bewusste Bewegung vollzieht.

Wie oben erläutert, hat die bewusste Aktion eine modulare Struktur, in der jede dargestellte Aktion ein einheitliches Modul bildet. Er ist für alle Bewegungen, die zu dieser Aktion gehören, verantwortlich. Die Module sind unabhängig voneinander und die Aktion eines Moduls kann fließend in die Aktion eines anderen Moduls übergehen. Die gleichzeitige Anwesenheit der Aktionen von mehreren Modulen kann zum unbegreiflichen und zu komplexen Verhalten führen und ist für den ROTTO nicht vorgesehen.

Die rechte Abzweigung in der Verhaltensstruktur (Bild 4.9) bildet die *unbewussten Aktionen*. Ihre Bestimmung ist es, zusätzliche besondere Bewegungen im Roboterverhalten einzubringen. Die Menschen und Tiere vollziehen reflektorische, unbewusste Bewegungen, die jeweils an ihrem Körper beobachtbar und ablesbar sind. Beispielweise ist die Bewegung der Augenlider beim Augenblinzeln ersichtlich und bei der Atmung bewegen sich der Brustkorb und teilweise die Lippen. Diese unbewussten Bewegungen sind stets mit einem Lebewesen ver-

bunden und können ROTTO dadurch dazu verhelfen, zu den Lebewesen zugeordnet zu werden. Damit wird versucht, die Roboterakzeptanz bei den Menschen zu erhöhen.

Die Struktur der *unbewussten Aktionen* ist auch modular gebaut. Jeder Modul funktioniert unabhängig nicht nur von anderen Modulen sondern auch von den Stimuli und vom Entscheidungssystem.

4.2.2.3 Entscheidungssystem

Das Entscheidungssystem ist ein Verbindungsglied zwischen den Stimuli und dem Verhalten des Roboters ROTTO. Es vereinigt alle einzelnen Komponenten für eine einheitliche emotionale Steuerung. Das Entscheidungssystem, die Stimuli und das Verhalten des Roboters stellen seinen Charakter zusammen. Aber der Kern ist das Entscheidungssystem, weil das zeigt, welche Bewegungen der Roboter bei dem entsprechenden Ereignis machen sollte.

Bild 4.12 veranschaulicht den Aufbau des Entscheidungssystems und die detaillierte Struktur der Emotionssteuerung für den Roboter ROTTO. Die von links dargestellte Stimuli und von rechts, das Verhalten, wurden zuvor bereits diskutiert und sind durch das Entscheidungssystem verbunden. Dessen Aufbau erfolgt mittels dreier zweckbestimmter Tabellen.

Die erste Tabelle (Bild 4.12, links – *Tabelle der Stimulussätze*) trägt in jeder Zeile die Kombination der Stimuli, die in einem (bestimmten) Zeitpunkt auf den ROTTO wirken können. Beispielweise wird dem Roboter während einer Interaktion ein buntes Objekt gezeigt und ihm schnell angenähert. Hier sind der interne, *initiative* Stimulus (iIT, s. Anhang A für die Abkürzungen), der Zustand „Anwesenheit eines Gegenstandes“ (zAG) und das Ereignis „scharfe Annäherung eines Objektes“ (eSO) vorhanden, die in einer Zeile der ersten Tabelle platziert sind. Auf diese Weise werden die anderen Kombinationen der Stimuli in die *Tabelle der Stimulussatz* eingetragen.

Die zweite Tabelle (Bild 4.12, rechts – *Tabelle der Aktionssätze*) beinhaltet in jeder Zeile die Kombination der Emotionen und der Aktionen, die einheitlich erregt werden können. Im zuvor aufgeführten Beispiel wurde die Kombination der Stimuli (eSO, zAG, iIT) gewählt. Ihre entsprechende Reaktion erregt die Emotion „Schreck“ (ES) mit dem gleichzeitigen bewussten Verhalten „Objektverfolgung“ (BOV). Solche logischen Kombinationen der emotionalen Reaktionen und der Aktionen, die von bestimmten Stimuli abhängen, bilden die *Tabelle der Aktionssatz* des Entscheidungssystems.

In der dritten Tabelle (Bild 4.12, mittlere – *Tabelle der Zusammenhänge*) wird in jeder Zeile jeweils eine vorher definierte Zeile der ersten Tabelle mit einer bestimmten Zeile der zweiten Tabelle bzw. der Einfluss der anwesenden Stimuli auf das Roboterverhalten kombiniert.

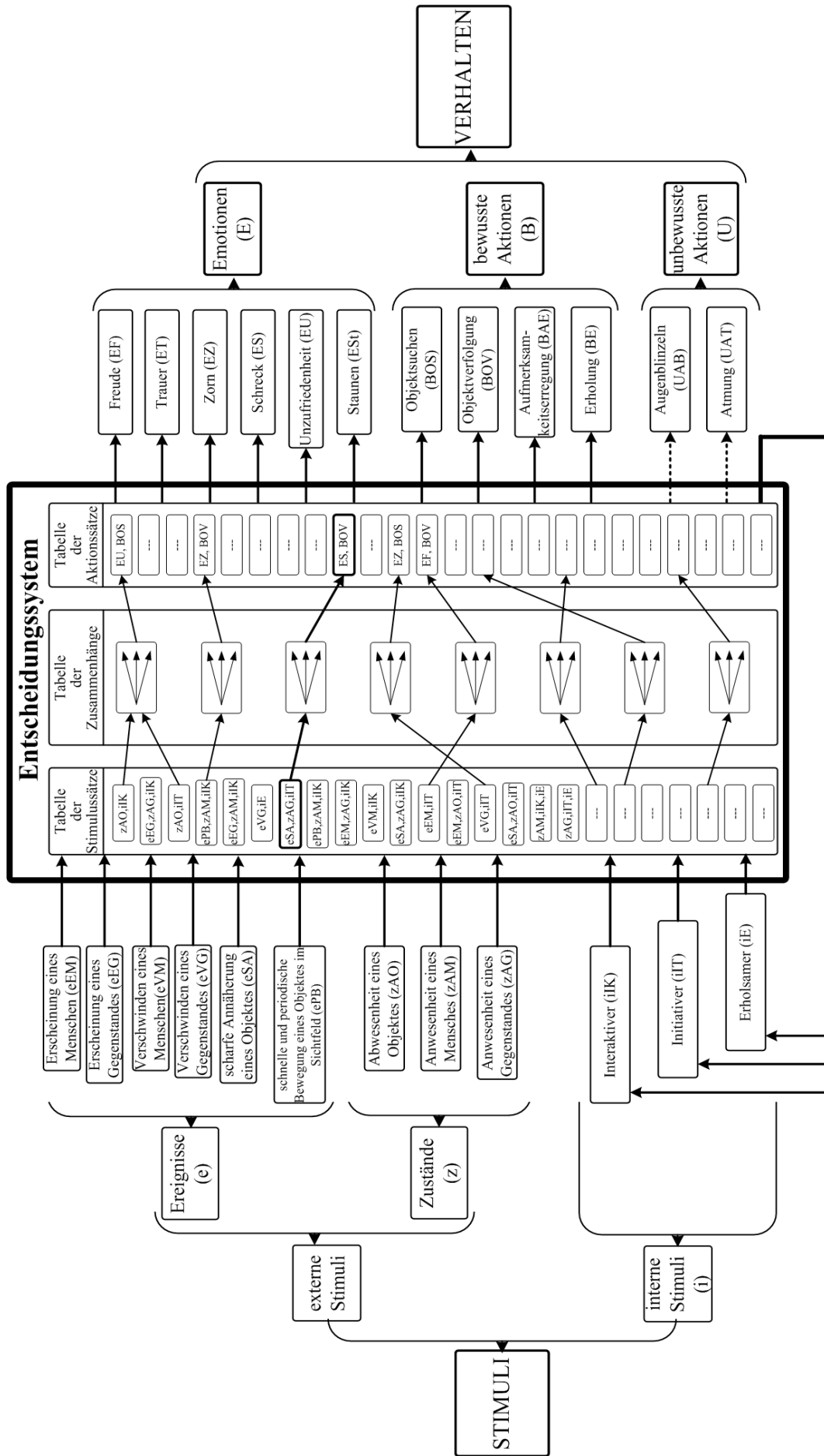


Bild 4.12 – Entscheidungssystem und detaillierte Struktur der Emotionssteuerung für den Roboter ROTTO

Die Modularität des Entscheidungssystems wird tabellarisch realisiert. Mit dem Anschluss eines neuen Stimulusmoduls, werden zusätzliche Zeilen (für einen neuen Stimulus) zur *Tabelle der Stimulussätze* addiert. Wenn eine neue Emotion oder Aktion an den Roboter angeschlossen wird, erweitert sich die *Tabelle der Aktionssätze* durch einen neuen Verhaltenssatz. Ebenso wird in beiden Fällen die *Tabelle der Zusammenhänge* durch eine neue „Stimulus-Verhaltens“-Verbindung ergänzt.

Die Reaktionszeit des Roboters ROTTO auf ein Ereignis hängt von der Schnelligkeit des Entscheidungssystems ab. Seine Arbeitsweise besteht in der zyklischen Überprüfung jeder Zeile aus der *Tabelle der Zusammenhänge* in Hinblick auf die Übereinstimmung von der jetzt wirkenden Stimuli mit den Stimuli dieser Zeile (mittels der *Tabelle der Stimulussätze*). Falls diese Übereinstimmung getroffen ist, wird der entsprechende Satz der Roboteraktionen (aus der *Tabelle der Aktionssätze*) aktiviert. Die Schnelligkeit des Entscheidungssystems des Roboters ROTTO entspricht der menschlichen Reaktionszeit auf optische Signale [78] und beträgt daher 0.2s. Das bedeutet, dass der Roboter ROTTO fünfmal pro Sekunde die Ereignisse in der Umgebung wahrnehmen kann.

4.3 Praktische Implementierung

In diesem Unterkapitel wird die Implementierung des emotionalen Steuerungssystems für die Mensch-Roboter-Interaktion detailliert dargelegt. Alle Systemteile sind in dieser Arbeit mit Hilfe von einer RCP-fähigen Software „MATLAB/Simulink“ und der C/C++ Programmiersprachen implementiert. Um die Allgemeingültigkeit der Implementierung beizubehalten, werden alle Strukturteile des Steuerungssystems schematisch in graphischer Form dargestellt.

Das Bild 4.13 veranschaulicht die allgemeine Struktur des Steuerungssystems von ROTTO für die Mensch-Roboter-Interaktion. Sie besteht aus drei Ebenen. In der ersten sind die Algorithmen zur Bildverarbeitung implementiert, wie z.B. die Erkennung von Gesichtern, bunten Gegenständen sowie schnellen Bewegungen. Sie werden mit den Programmiersprachen C/C++ auf der Basis der freien „Computer Vision“ Bibliothek „OpenCV“ als eine separate Windows-Anwendung realisiert.

Die zweite Ebene (Systemkern) widmet sich den Stimuli, dem Roboterverhalten und dem Entscheidungssystem, die zusammen den Kern der emotionalen Steuerung bilden. In dieser Arbeit werden mit dem „MATLAB/Simulink“ implementiert. Beide Ebenen werden am Host-PC unter „WindowsXP“ ausgeführt. Der Echtzeitbetrieb mit der Abtastzeit (bzw. Schrittzeit) von 10ms (und 200ms für das Entscheidungssystem) wird durch die Werkzeuge von MATLAB erreicht. Falls kleinere Abtastzeiten gefordert werden, kann das System mit einem echtzeitfähigen Betriebssystem, bspw. xPC-Target, aufgebaut werden.

Die dritte Ebene ist die Servoebene und beschäftigt sich mit der Motorensteuerung. Diese bekommt von der zweiten Steuerungsebene die Sollwertpositionen (im Grad) für alle Servoantriebe. Jeder Sollwert wird aus dem Grad in die PWM Pulsdauer errechnet und danach der entsprechenden Ansteuerung bzw. dem Hardwaremodul übergeben. Diese Servoebene ist im Bild 4.14 strukturell ersichtlicht und in einem separaten „MATLAB/Simulink“-Modell aufgebaut. Dieses Modell mit der Servoebene lässt sich durch den Einsatz des „RTW NetX10-Targets“ per Mausklick zuerst in die C-Programmiersprache und danach in den Prozessorcode umwandeln und direkt die Technologieplattform NetX10 (XPIC) programmieren. Die Servoebene ist mit den anderen Ebenen per Ethernet-Netzwerk für den Datenaustausch verbunden und dadurch synchronisiert.

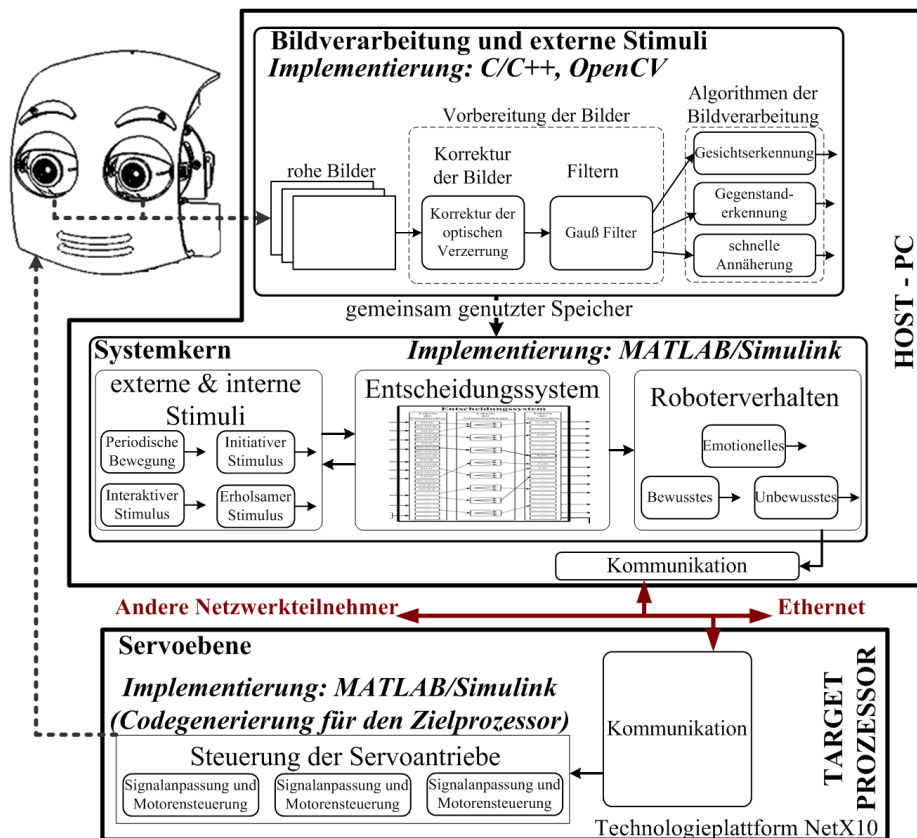


Bild 4.13 – Das allgemeine strukturelle Schema des emotionalen Steuerungssystems für die Mensch-Roboter-Interaktion

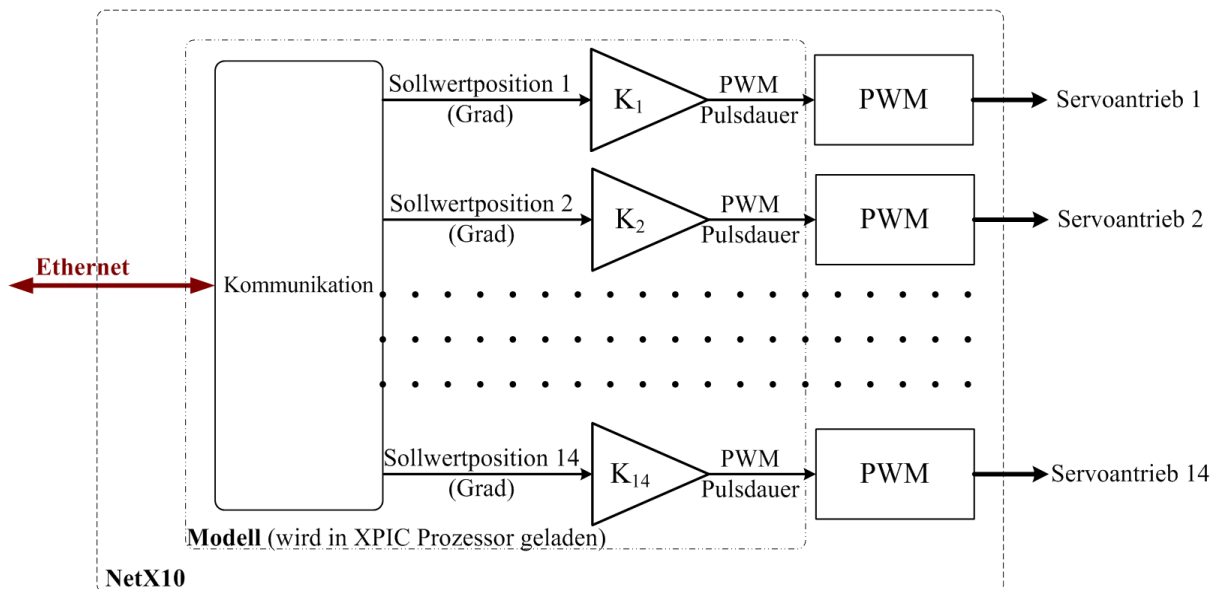


Bild 4.14 – Das strukturelle Schema der Servoebene

4.3.1 Externe Stimuli

Alle externen Stimuli sind mit der Änderung in der Umgebung verbunden. Der Roboter nimmt sie mit einem Kamerasystem und der nachfolgenden Bildverarbeitung wahr. Als Kamerasystem wurden zwei USB-Videokameras C300 von Logitech [110] verwendet, deren Bauform der Kopfkonstruktion gut angepasst ist (s. Kapitel 2.4.2). Die Bildverarbeitung wird mit Hilfe von der freien „Computer Vision“ Bibliothek OpenCV [111] realisiert. Das allgemeine Schema der implementierten Bildverarbeitung veranschaulicht das Bild 4.15.

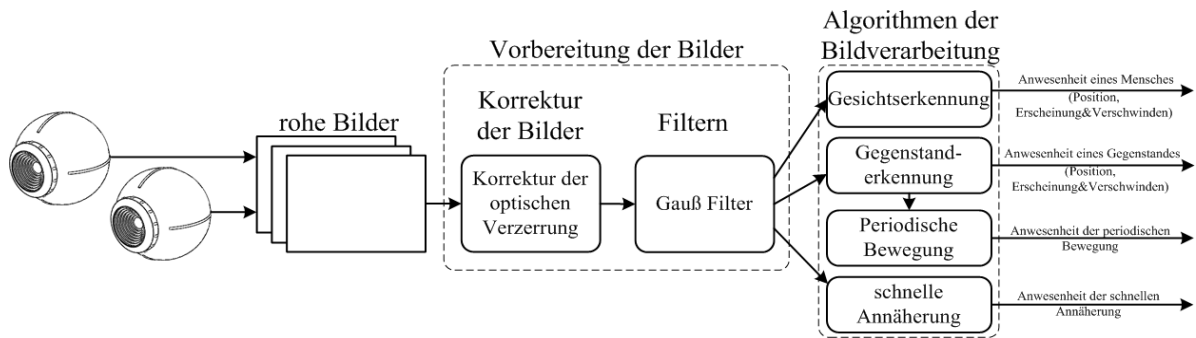


Bild 4.15 – Das strukturelle Schema der Bildverarbeitung für den Roboter ROTTO

Die Bildverarbeitung wird in zwei Stufen durchgeführt. Zuerst werden die Rohdatenbilder für die nachfolgenden Algorithmen vorbereitet. In jedem Bild wird die optische Verzerrung der Linse korrigiert und die hochfrequenten Geräusche mit dem Gauß-Filter filtriert. Dieses Verfahren wurde detailliert in [112] dargestellt. Weiterhin, in der zweiten Stufe, kommen die spezifischen Algorithmen der Bildverarbeitung zum Einsatz. Hier werden die oben beschriebenen Ereignisse und Zustände der Roboterumgebung aus den vorbereiteten Bildern extrahiert.

Für die Bestimmung der mit einem Mensch verbundenen Ereignisse und Zustände (Bild 4.16: Erscheinung, Anwesenheit und Verschwinden eines Menschen) wird zuerst die Gesichtserkennung eingesetzt. Die Gesichtserkennung ist in der *OpenCV* Bibliothek mit Hilfe des *Haar-Merkmale*-Detektors [114] realisiert. Er sucht im Videobild nach den Gesichtern und stellt als Ergebnis die Größe und Bildkoordinaten der gefundenen Gesichter zur Verfügung. Da der Roboter seine Aufmerksamkeit nur einem Objekt bzw. Gesicht widmen kann, wird aus den vielen Gesichtern das größte genommen.

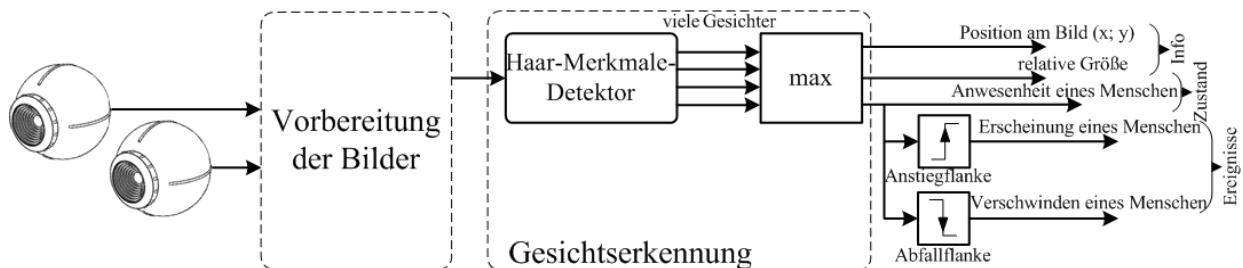


Bild 4.16 – Das Schema der Berechnung der mit einem Mensch verbundenen Ereignisse und Zustände

Wenn ein Mensch im Robotersichtfeld erscheint, wird sein Gesicht detektiert und werden die Gesichtsgröße und Koordinaten werden berechnet. Darüber hinaus wird der Zustand - die „Anwesenheit eines Menschen“ - hergestellt. Die Anstiegflanke dieses Zustandes signalisiert die Erscheinung eines Menschen im Sichtfeld und ein entsprechendes Ereignis. Die Gesichts-

koordinaten werden nun für die Objektverfolgung eingesetzt. Wenn der Mensch das Sichtfeld verlässt, fällt der Zustand und seine Abfallflanke signalisiert das Verschwinden eines Menschen.

Die Bestimmung der mit einem Gegenstand verbundenen Ereignisse und Zustände (Bild 4.16: Erscheinung, Anwesenheit und Verschwinden eines Gegenstandes) ist ähnlich aufgebaut. Der Unterschied liegt im Algorithmus der Gegenstandserkennung.

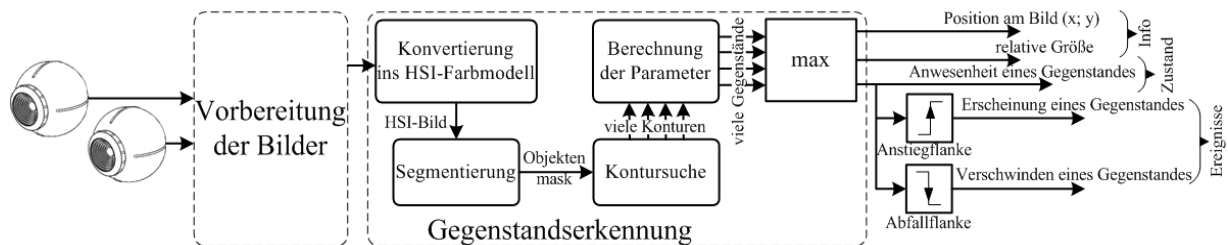


Bild 4.17 – Das Schema der Berechnung der mit einem Gegenstand verbundenen Ereignisse und Zustände

Die Gegenstandserkennung beruht auf der Farberkennung (ähnlich wie in [113]) und beginnt mit der Konvertierung der Videobilder aus dem RGB in das HSI-Farbmodell (H – Farbton, S – Farbsättigung, I – Hellwerte, eng: hue, saturation, intensity). Dann wird das HSI-Farbbild auf drei Räume (gelb, blau, grün) segmentiert. Im Vergleich zur RGB, bietet die HSI-Farbsegmentierung eine bequeme Anpassung an die jeweilige Farbe, ihre Helligkeit und ihre Sättigung. Nach der Farbsegmentierung sind die schwarz-weißen Masken der farbigen Gegenstände verfügbar. Zur Berechnung der Kenndaten jeder Kontur, werden die Größe und die Koordinaten der passenden Ellipse herangezogen. Für die Erlangung der Roboter Aufmerksamkeit wird nur ein Gegenstand gefördert, deswegen wird aus den gefundenen farbigen Gegenständen der größte entnommen. Der Aufbau des Ereignisses (Anwesenheit eines Gegenstandes) und die Zustände (Erscheinung und Verschwinden eines Gegenstandes) sind den zuvor beschriebenen, mit einem Gesicht verbundenen Ereignissen und Zuständen sehr ähnlich.

ROTTA kann die schnelle Annäherung eines Objektes wahrnehmen, um auf seine ungewünschten Position bzw. Entfernung entsprechend zu reagieren. Nähert sich beispielweise eine Person dem Roboter zu nah, dann ist es unmöglich, das Gesicht (oder einen Gegenstand) zu erkennen, da es entweder zu groß ist oder sich nicht vollständig im Sichtfeld des Roboters befindet. Der Roboter soll dabei emotional (z.B. durch Unzufriedenheit) zeigen, dass die Person an ihn falsch bzw. zu nah herantreten ist.

Der Implementierung des Ereignisses „schnelle Annäherung“ liegt der „Motion detection“ des Roboters „Kismet“ [115] zugrunde und wird mit einer Modifizierung am Bild 4.18 veranschaulicht. Zuerst wird das Intensitätsfeld als eine Differenz zwischen zwei Bildern einer Videoreihe berechnet. Für den „Kismet“ wurde dieses Intensitätsfeld als „Motion field“ betrachtet. Für ROTTA wird dazu die Summe der Intensitätsmodule von allen Pixeln aus dem Intensitätsfeld zusammenrechnet und ergibt somit den allgemeinen Intensitätswert. Bei der schnellen Annäherung eines Objektes zum Roboter hin, ändert sich sehr schnell die Bildintensität. Ihr entsprechender allgemeiner Intensitätswert erreicht hohe Werte. Sobald er den Grenzwert G überschreitet, wird das Ereignis „schnelle Annäherung“ aktiv. Die Grenze zwischen der langsamen und der schnellen Annäherung wird durch den Grenzwert definiert.

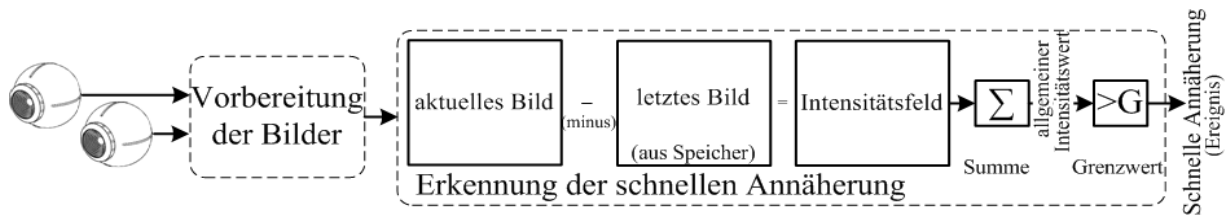


Bild 4.18 – Das Schema der Implementierung des Ereignisses „schnelle Annäherung“

Das letzte für den ROTTO definierte Ereignis ist die schnelle periodische Bewegung in seinem Sichtfeld. Eine Person kann die Roboteraufmerksamkeit mittels eines schwenkenden, bunten Gegenstandes erregen, daher wird als Informationsquelle für dieses Ereignis die Gegenstandposition (von „Anwesenheit eines Gegenstandes“) ausgewählt. Das Erkennungsschema der periodischen Bewegung ist aus dem Bild 4.19.a ersichtlich. Für die Erkennung der Schwingungen, werden die Schwenkungsfrequenzen mit einem Wert von zwei bis vier Herz angenommen, die mittels eines Bandpasses mit einer Frequenz von 3Hz (die Filterbeschreibung – 4.1 und die Frequenzkennlinie – Bild 4.19.b) herausgefiltert werden. Danach wird die Amplitude der Schwenkungen mittels des Moduls, des Tiefpasses (mit der Grenzfrequenz von 1Hz, Bild 4.19.c) und eines Grenzwertes bestimmt.

$$F_f(t) = \frac{T_n^2 \cdot s^2 + 2 \cdot d_n \cdot T_n + 1}{(T_{dn1}^2 \cdot s^2 + 2 \cdot d_{dn1} \cdot T_{dn1} + 1) \cdot (T_{dn2}^2 \cdot s^2 + 2 \cdot d_{dn2} \cdot T_{dn2} + 1)},$$

wo

$$\begin{aligned}
 F_n &= 2\text{Hz}, & F_{dn1} &= 3\text{Hz}, & F_{dn2} &= 3\text{Hz} & (4) \\
 T_n &= \frac{1}{2 \cdot \pi \cdot F_n}, & T_{dn1} &= \frac{1}{2 \cdot \pi \cdot F_{dn1}}, & T_{dn2} &= \frac{1}{2 \cdot \pi \cdot F_{dn2}} \\
 d_n &= 1, & d_{dn1} &= 0.9, & d_{dn2} &= 0.9
 \end{aligned}$$

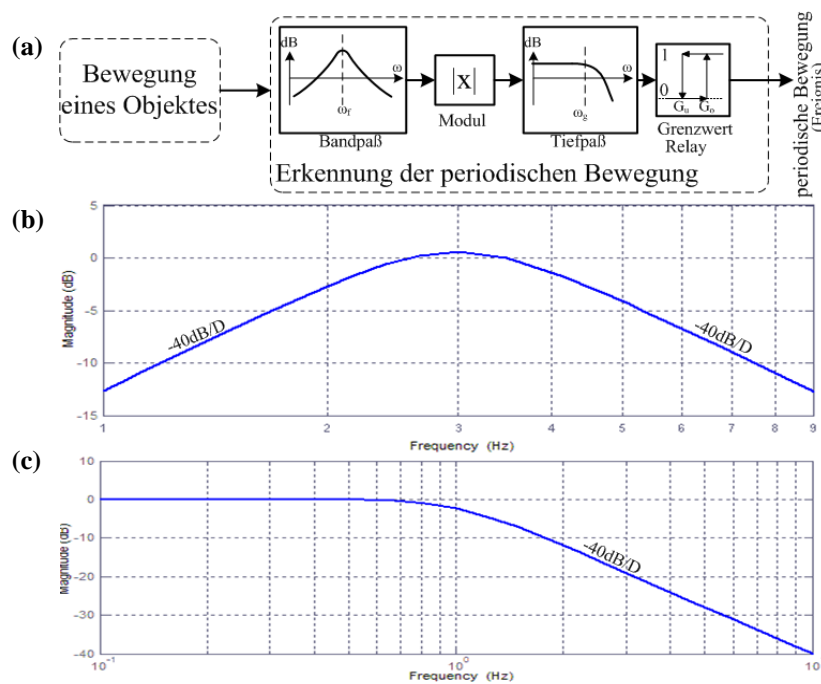


Bild 4.19 – Erkennung der periodischen Bewegung

Im Bild 4.20 wird das Experiment zur Erkennung der periodischen Bewegung gezeigt, in dem vor dem Roboter gesicht ein gelber Gegenstand periodisch bewegt wurde (Gegenstandposition - Bild 4.20.a).

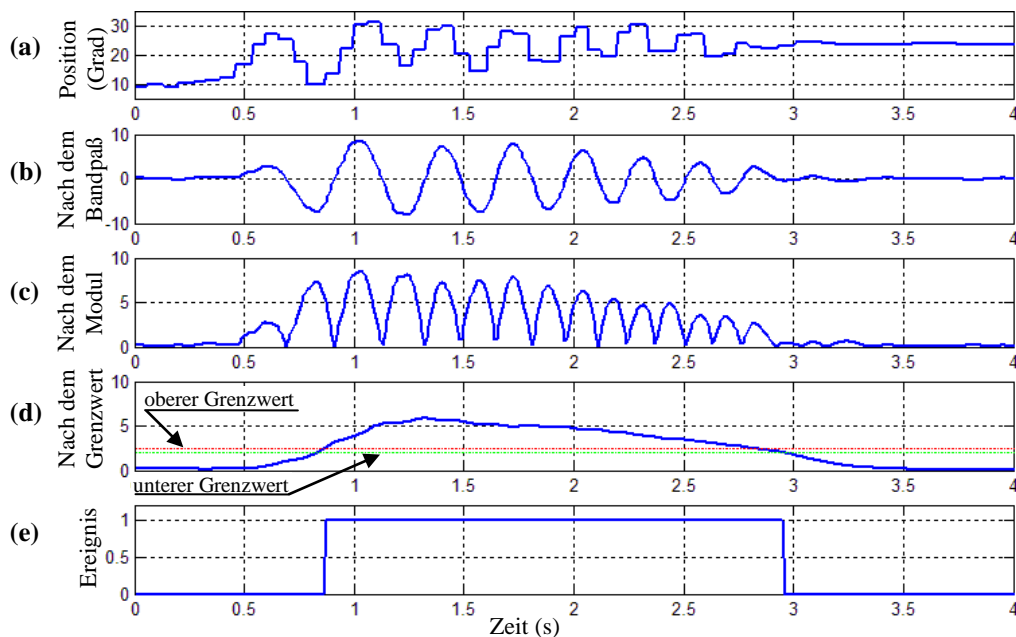


Bild 4.20 – Experiment zur Erkennung der periodischen Bewegung

Nach dem Bandpaßfilter sind die Schwingungen mit der gewünschten Frequenz gut erkennbar (Bild 4.20.b). Der Amplitudenverlauf (Bild 4.20.d), der proportional zur Schwingungsamplitude verläuft, wird nach dem Modul (Bild 4.20.c) und dem Tiefpassfilter ermittelt. Letztendlich wird der Amplitudenverlauf in Hinblick auf das Überschreiten des oberen Grenzwertes ($G_o=2,5$) hin geprüft. Dieses Überschreitungssignal (Bild 4.20.e) signalisiert die Erkennung der periodischen Bewegung. Das Signal sinkt im Frequenzbereich, sobald der Amplitudenverlauf bis unter den unteren Grenzwert ($G_u=2,3$) gesunken ist. Diese zweistufige Grenze erlaubt die kurzzeitige Reaktion bei einer kleineren Schwingamplitude zu vermeiden. Die beiden Grenzwerte wurden experimentell ermittelt.

4.3.2 Interne Stimuli

Wie zuvor beschrieben, strebt der Roboter intern nach einer Interaktion durch interne Stimuli. Als Beispiel wurden drei Stimuli für den ROTTO ausgewählt: der interaktive, der initiative und der erholsame. Der interaktive Stimulus verfolgt das Ziel einen Mensch in der Umgebung zu finden und mit ihm einen Dialog zu starten. Ziel des initiativen Stimulus ist es, einen Gegenstand (Spielzeug) zu finden, und durch ihn mit einem Mensch in Interaktion zu treten. Der erholsame Stimulus ist ein integraler Wert der Anwesenheit der anderen Stimuli (bzw. Roboteraktivitäten).

Die ersten zwei Stimuli (interaktiver und initiativer) werden gleichermaßen realisiert, d.h. Intensität bzw. der Angrad des Stimulus ergibt sich aus einer Zufallszahl (maximal 5% von der maximalen Intensität) und der Summe der Anregungen (Integralwert) (Bild 4.21.a).

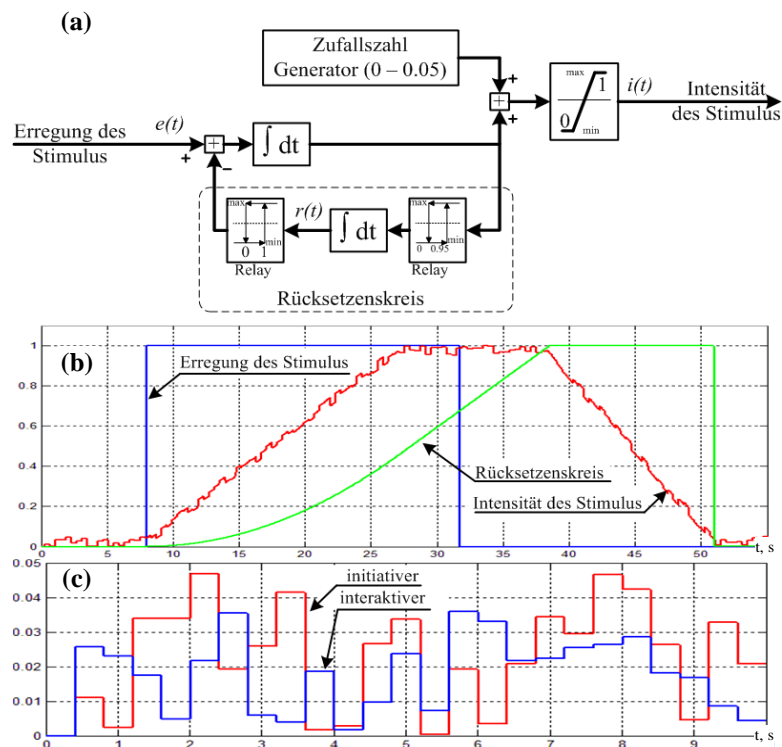


Bild 4.21 – Realisierung der anregend-temporären internen Stimuli: strukturelles Schema (a), Verlauf beim Anregen des Stimulus (b), initiativer und interaktiver Stimuli ohne Erregung(c)

Der Stimulus beruhigt sich durch den Rücksetzenskreis und ist von der aktuellen Intensität abhängig, d.h. je stärker die Intensität ist, desto früher wird die Beruhigung aktiviert. Das Beispiel des Stimulusverlaufs, in welchem der Stimulus maximal erregt wurde, wird im Bild 4.21.b veranschaulicht. Nach der 38. Sekunde hat sich ROTTO mit dem Rücksetzenssignal beruhigt. Das Bild 4.21.c zeigt gleichzeitig beide Stimuli ohne Erregung. Dank der Zufallszahlgeneratoren dominiert in jeder Zeit nur einer der beiden Stimuli. Deswegen wird der Roboter nur auf das Objekt reagieren (auf einen Menschen oder einen Gegenstand), das dem dominierenden Stimulus entspricht.

Der erholsame Stimulus ist ein integraler Wert aus den Intensitäten der interaktiven und initiativen Stimuli (Bild 4.22.a). Er verdeutlicht die „Müdigkeit“ des Roboters.

Es wird eine minimale Anregung ($C \leq 0.5$) aktiviert, in welcher der erholsame Stimulus nicht ansteigt (keine Ermüdung, Bild 4.22.b, bis zur 10. Sekunde). Danach wird er weiter angeregt und die Intensität steigt (Bild 4.22.b, 10 – 51s.) solange die Anregung existiert. Wenn der erholsame Stimulus eine hohe Erregung erreicht (den Wert von 1.0), generiert das Entscheidungssystem das Signal zur Erholung. Der Roboter verinnerlicht in diesem Zeitrahmen das bewusste Verhalten „Erholung“. Das Signal zur Erholung wirkt im erholsamen Stimulus anderen Anregungen entgegen, die Erregung wird auf die Nullebene gesenkt. Nach der Erholung steht der ROTTO nun für neue Interaktionen bereit.

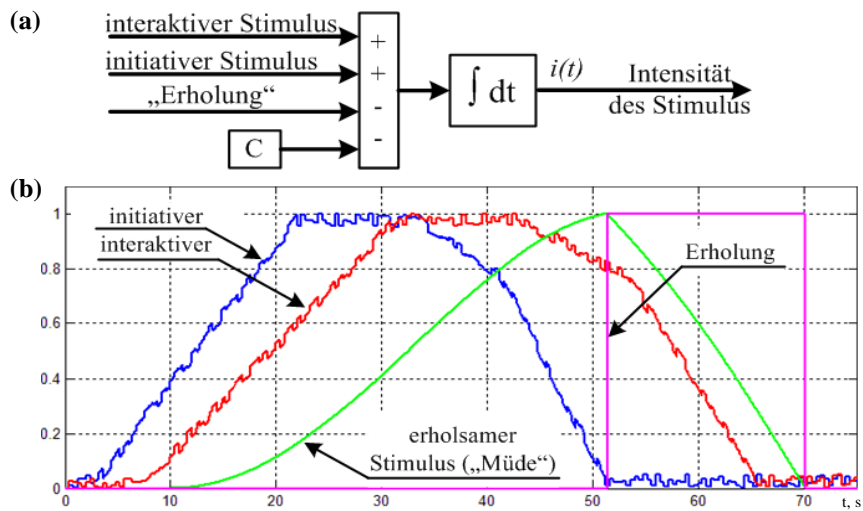







Bild 4.22 – Realisierung der anregend-kumulativen erholsamen Stimuli: strukturelles Schema (a), Verlauf bei der Anregung des Stimulus (b)

4.3.3 Emotionales Verhalten

Wie im Vorigen Kapitel erklärt, setzt sich das emotionale Verhalten hauptsächlich aus den verschiedenen Gesichtsausdrücken zusammen. Sie erlauben dem Roboter sowohl die eigene Reaktion auf ein Ereignis auszuprägen als auch den Interaktions- bzw. Dialogpartner zu beeinflussen. Die Gesichtsausdrücke bzw. die Emotionen werden im Robotergesicht durch das koordinierte Zusammenspiel der lagegeregelten Servoantriebe ersichtlich. Darauf ist jede Emotion durch die Achsenpositionen gekennzeichnet. In der Tabelle 4.1 werden diese Positionen jeder Emotion der hundertprozentigen Intensität zugewiesen und mit dem neutralen Gesicht verglichen.

Tabelle 4.1 – Einzelheiten des Aufbaus der Emotionen

Emotion	Achsenpositionen, grad								Bemerkung	Abbildung	
	Lippen				Augenbrauen		Augenlider				
	α_{f1}	α_{f2}	α_{f3}	α_{f4}	α_{t1}	α_{t2}	α_{s1}	α_{s2}			
Neutrale	0	0	0	0	0	0	0	0	0	Anfangspositionen	
Freude	10	15	-10	-15	-15	15	5	5	---		

Trauer	-10	-15	10	15	15	-15	-10	-10	Der Kopf neigt sich um 4Grad	
Zorn	10	-10	-10	10	40	-40	-20	-20	----	
Schreck	-30	0	30	0	-10	10	15	15	Der Kopf hebt sich um 5Grad, die Augen neigen sich um 5Grad	
Unzufriedenheit	10	-15	10	10	20	20	-15	-15	---	
Staunen	-10	15	10	-15	-40	40	-10	10	--	

Die Emotionen sind eine der wichtigsten Bestandteile in der Roboterinteraktion. Er zeigt seinen Zustand mittels einer oder mehrerer Emotionen. Der Mensch ist fähig mehrere Emotionen gleichzeitig auszudrücken. Der ROTTO hat diese Fähigkeit eingeschränkt erhalten: das Entscheidungssystem kann alle Emotionen aktivieren, aber nur die zwei intensivsten können zur Motorik gemischt weitergeleitet werden. Darüber hinaus werden ihre Intensitäten in Laufe der Zeit schwanken, um die kontinuierliche Gesichtsbewegung zu erreichen, d.h. ist eine Emotion aktiv, wird ihre Intensität verändert (durch Schwanken).

4.3.4 Bewusste Aktionen

Jede bewusste Aktion verfolgt ein bestimmtes Ziel. Wenn es kein Interaktionsobjekt gibt, wird der Roboter nach diesem suchen (Objektsuchen). Sobald eins gefunden wurde, wird die Aufmerksamkeit der Roboter für die nachfolgenden Interaktionen erregt (Aufmerksamkeitserregung). Während der Kontaktphase wird ROTTO seine Aufmerksamkeit auf das Objekt fokussieren (Objektverfolgen). Nach dem intensiven Interaktionsvorgang wird er müde (erholbarer Stimulus) und erreicht die Phase der Erholung, die dem Partner signalisiert, dass die Interaktion beendet wird. In diesem Unterkapitel wird die Realisierung der genannten Aktionen in ihren Einzelheiten betrachtet.

Der initiative und der interaktive Stimuli bewegen den Roboter zur Interaktion. Wenn kein Interaktionsobjekt ersichtlich ist, kommt es zum Einsatz des Objektsuchen. Dabei dreht der Roboter die Augen ungeordnet an die Seiten und fixiert sie kurzfristig. Das ist eine für den Menschen spezifische Augenbewegung, die sich aus den Fixierungen und sprunghaftigen Drehungen ergibt [116]. Zur Erweiterung des Sichtfeldes wird der Kopf mitgedreht.

Das Bild 4.23.a veranschaulicht in einem strukturellen Schema das Teilsystem für das Objektsuchen, die aus den zwei Zufallszahlengeneratoren und aus dem System der Blickrichtung besteht. Das System der Blickrichtung (Bild 4.23.b) übernimmt die gemeinsame Steuerung der Augen und des Kopfes. Es erlaubt den Augen sich in einem bestimmten Bereich ($\pm 15^\circ$) unabhängig (vom Kopf) zu bewegen (Bild 4.23.c, 0.6–1.5s). Beim Überschreiten dieses Bereiches bewegen sich der Kopf und die Augen hinsichtlich dessen, dass die Augen sich wieder in die Anfangsposition (0°) zurückdrehen können. Die Blickrichtung wird in diesem Fall durch die Kopfdrehung realisiert (Bild 4.23.c, 1.5–3.7s), was den Augen den vollen Bewegungsbereich gewährleistet. Beim großen Sollwert für die Blickrichtung wird die Kopfdrehung begrenzt und die Zurückdrehung der Augen wird unterbunden (Bild 4.23.c, 3.7–5s). Die Abhängigkeit k zeigt das Verhältnis zwischen den Kopf- und den Augendrehungen in Hinblick auf die gleiche Blickrichtung.

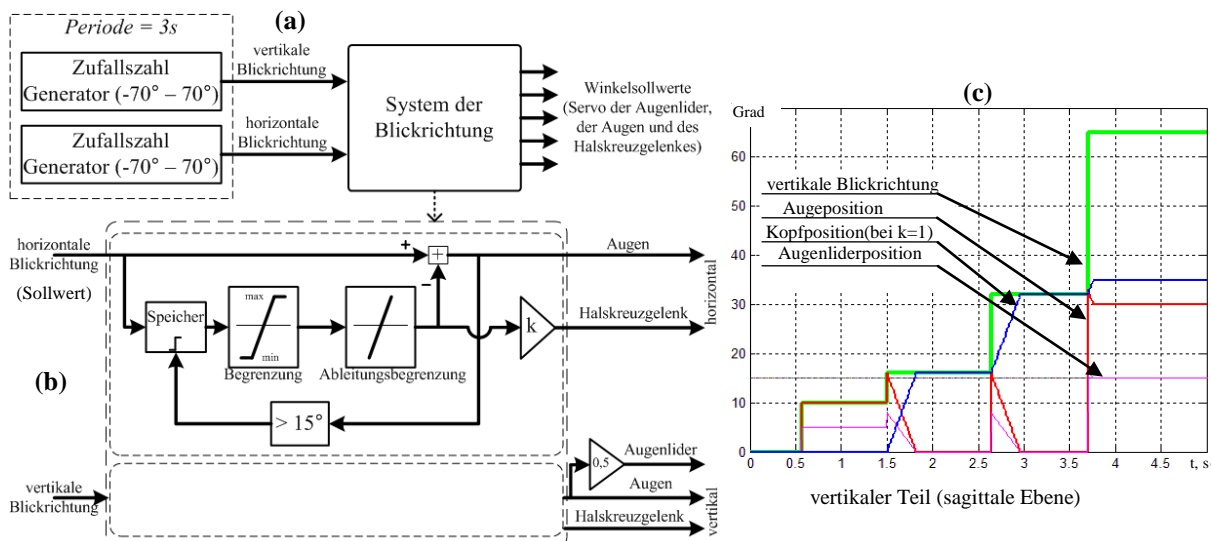


Bild 4.23 – Das strukturelle Schema des Teilsystems für das Objektsuchen (a), System der Blickrichtung (b) und Winkelsollwerte (c)

Im Unterschied zu den amerikanischen kontaktfreudigen Robotern (Kismet, Merz usw.) wurde der Roboter ROTTO nicht mit einem System zur Spracherzeugung ausgerüstet. Deswegen wird die Aufmerksamkeit der zu kontaktierenden Person mittels der schnellen Lippenbewegungen (eine Nachbildung der Lippenbewegungen bei dem Reden) erregt.

Während der Kontaktphase fokussiert ROTTO seine Aufmerksamkeit auf das Interaktionsobjekt (ein Mensch oder ein Gegenstand), d.h. der Roboterblick wird stetig an diesem gerichtet. Hier kommt das Teilsystem für das Objektverfolgen (Bild 4.24.a) zum Einsatz. Das Teilsystem ist so eingerichtet, dass der Blick des Roboters sich mittels einer Regelung auf das Objekt fokussiert. Dies besteht aus zwei gleichen PI-Reglern, die für die vertikale und die horizontale Blickrichtung zuständig sind, und aus dem System der Blickrichtung (dasselbe für das Objektsuchen). Für die Begrenzung des integralen Teiles des PI-Reglers ist der modifizierte Regler mit dem „anti-wind-up“ Verfahren [117] geeignet (Bild 4.24.b). Das Regelungsobjekt (lageregelter Servoantrieb und das Auge bzw. die Videokamera) wurde als PT_1 -Glied mit der Zeitkonstante von 0.1s ausgewählt. Die Objektposition wurde zu jeder 1/15s (Videobildfrequenz 15 Bilder/s) ermittelt. Der geschlossene Regelkreis ergibt das PT_2 -Glied, für welches die Dämpfung $d=1$ und Zeitkonstante $T=0.15s$ bei der Reglereinstellung festgesetzt wurden.

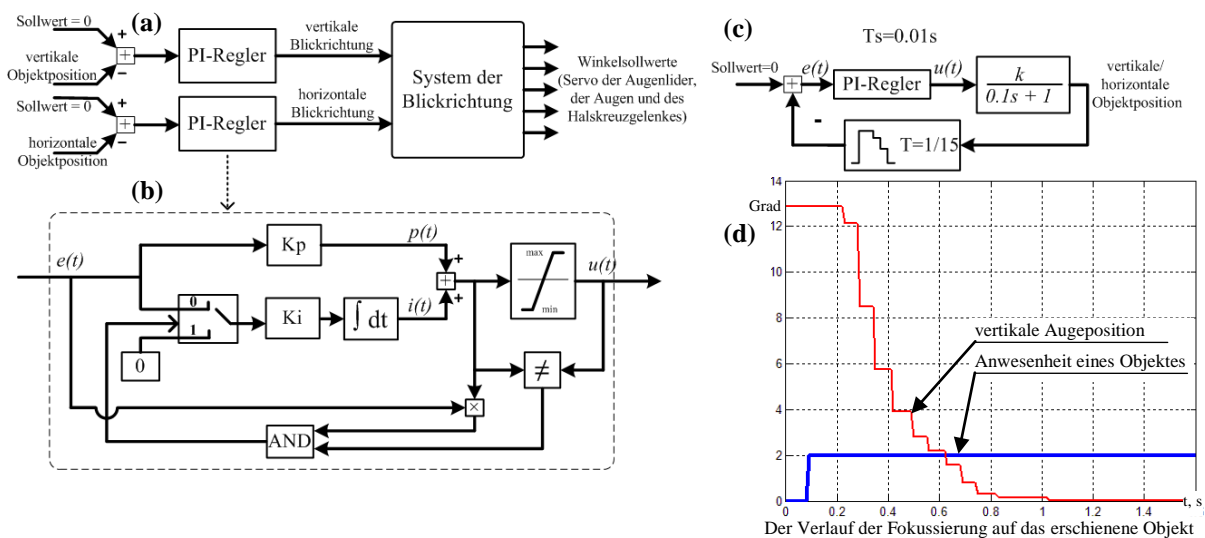


Bild 4.24 – Das strukturelle Schema des Teilsystems für das Objektverfolgen (a), PI-Regler mit der „anti-wind-up“ Korrektur (b), System zur Reglereinstellung (c) und Fokussierungsverlauf (d)

Nach der intensiven Interaktion (hoch erregter interaktiver und/oder initiativer Stimuli) beendet der Roboter den Dialog. Er zeigt seine Müdigkeit durch die verzögerten Bewegungen und der beschleunigten Atmung (s. Unterkapitel 4.3.5, Atmung) und der hoch erregte erholsame Stimulus drängt den Roboter zur Erholung. Während der Erholung bleibt ROTTO jedoch in Bewegung: zuerst kippt er seinen Kopf nach unten und schließt die Augenlider – dieser Vorgang ist der Erholungsanfang. Während des Erholungsvorgangs (Schlafen) bewegt ROTTO ein bisschen die Augenbrauen, die Lider und die Lippen ganz zufällig und während der kleinen Zeitabschnitte. In dieser Zeit sinkt der Grad der „Müdigkeit“ und sobald die Nullebene erreicht ist, wacht der Roboter auf, d.h. er öffnet seine Augenlider und hebt den Kopf. ROTTO ist nun wieder zur neuen Interaktionen bereit.

4.3.5 Unbewusste Aktionen

In der Rolle der unbewussten Aktionen sind für den Roboter ROTTO das Augenblinzeln und eine Nachbildung der Atmung implementiert. Diese Aktionen ahmen teilweise die menschlichen Reflexe nach.

Beim Augenblinzeln werden die Augenlider schnell geschlossen und wieder geöffnet. Dieser Vorgang wiederholt sich alle 10s. bis 15s mit einer veränderlichen Periode, die aus der Grundzeit (10s) und der Zufallszeit (0-5s) besteht (Bild 4.25, Schema des Teilsystems). Während des Augenblinzeln wird das Blickfeld der Augenlider geschlossen und die Videoinformation geht verloren. Um diese im Steuerungssystem zu berücksichtigen, wird ein Signal „Augen zu“ generiert. Es stoppt alle Algorithmen, die die Videoinformation nutzen. Nach dem Augenblinzeln wird dieses Signal abgeschaltet und die Algorithmen werden weiter frei ausgeführt (Bild 4.25, unten).

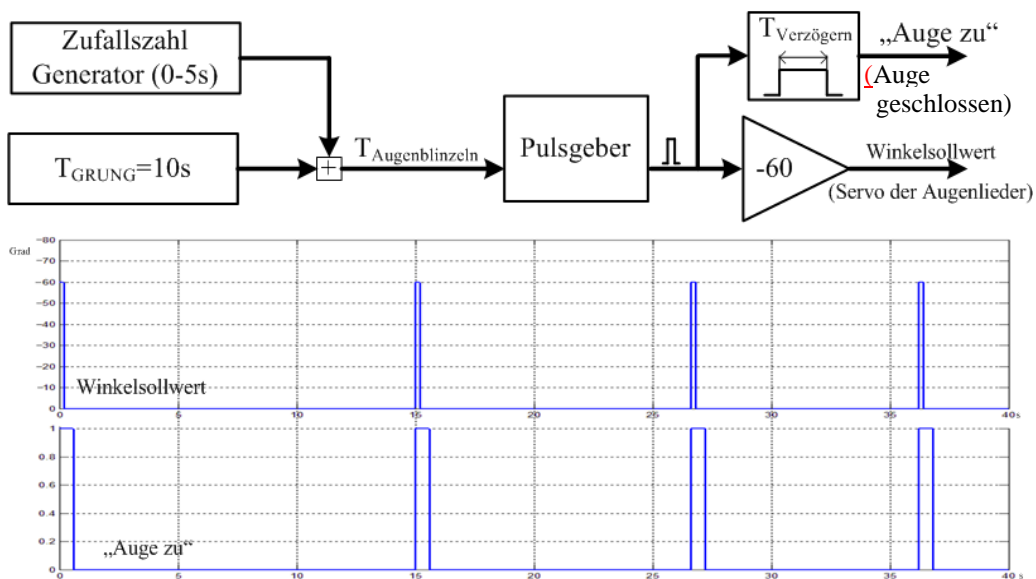


Bild 4.25 – Das strukturelle Schema des Teilsystems zeigt das Augenblinzeln (oben) und die erzeugten Signale (unten)

Die Atmung im Roboter Gesicht wird durch Lippenbewegungen nachgebildet. Der Positionsgenerator generiert die halbsinoidalen Winkelsollwerte (Bild 4.26) für die Lippendrehungen mit einer Periode, die eine Grundfrequenz besitzt und teilweise von der „Müdigkeit“ des Roboters abhängt. Je höher der Grad der Müdigkeit ausfällt, desto geringer ist die Periode. Damit wird eine Ermüdung des Roboters signalisiert.

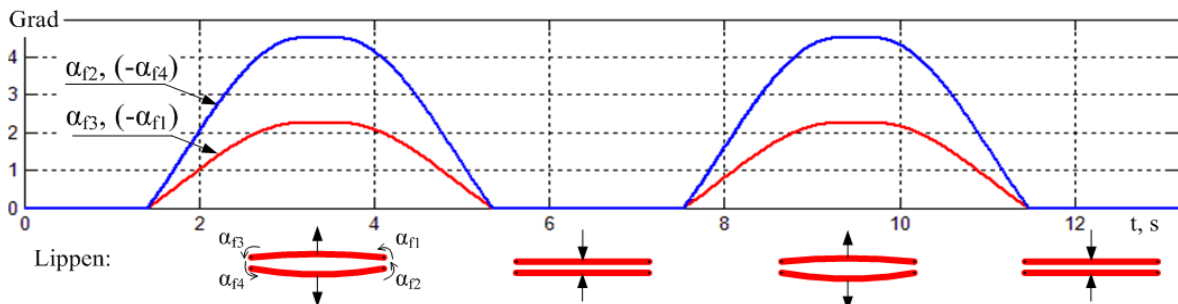


Bild 4.26 – Signale des Teilsystems der Atmung und der Lippenpositionen

4.3.6 Entscheidungssystem

Wie es im Unterkapitel 4.2.2.3 dargestellt wurde, soll das Entscheidungssystem die aktuelle Stimuli (externe und interne) bewusst mit dem Roboterverhalten verbinden. Der Roboter soll für den Menschen verständlich auf die vorgesehenen Ereignisse reagieren und seinen Zustand deutlich verdeutlichen. Für diese Art der „Intelligenz“ ist das Entscheidungssystem zuständig. Es wurde tabellarisch, mit der Möglichkeit zur Erweiterung und Vervollständigung, aufgebaut und ist universell anwendbar.

Das Entscheidungssystem wird in der MATLAB-Programmiersprache realisiert, da sie eine bequeme und einfache Operation mit den Matrizen, Vektoren und Datenbanken erlaubt. Das Entscheidungssystem besteht aus den drei Tabellen im Bild 4.27. In der ersten Tabelle werden alle zu dieser Zeit aktiven Stimuli an jedem neuen Rechenakt zusammengestellt. Diese Tabelle enthält zwei Zeilen, erste zeigt, welche Stimulussätze aktiv sind, und zweite signalisiert ihre Intensität. Die Tabelle 4.2 (s. Bild 4.12 für die Abkürzungen) veranschaulicht ein Beispiel, in dem der Zustand „Abwesenheit eines Objektes“ und die internen „inretaktiven“ (mit der Intensität von 0.05) und „initiativen“ (mit der Intensität von 0.03) Stimuli aktiv sind.

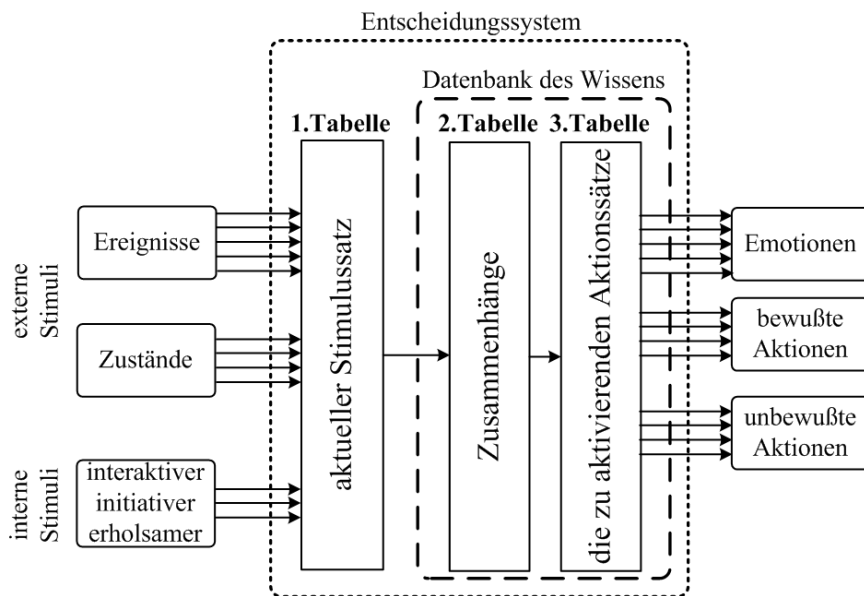


Bild 4.27 – Implementierung des Entscheidungssystems

Tabelle 4.2 – Ein Beispiel für eine Tabelle der aktiven Stimulussätze

	externe Stimuli									interne Stimuli		
	Ereignisse						Zustände			iIK	iT	iK
Stimuli	eEM	eEG	eVM	eVG	eSA	ePB	zAO	zAM	zAG			
aktiv	0	0	0	0	0	0	1	0	0	1	1	0
Intensität	0	0	0	0	0	0	0	0	0	0.05	0.03	0

Die Tabellen der Zusammenhänge und der Aktionssätze bilden eine Wissensdatenbank. Sie ist direkt für das bewusste Roboterverhalten zuständig. Jede Zeile in der Tabelle der Zusammenhänge weist auf die Zeile der Aktionssatz-Tabelle hin, die aufzeigt welches Roboterverhalten aktiviert werden soll. Beide Tabellen werden erfahrungsgemäß in einer Textdatei (auf Grund des menschlichen Verhaltens) abgebildet.

Aus Tabelle 4.3 ist ein Beispiel für solch eine Textdatei ersichtlich. Rechts sind die Zeilen der Aktionssatz-Tabellen abgebildet und links die entsprechende Zeilen der Zusammenhang-Tabelle. Jeder Aktionssatz von rechts zeigt, welche Aktionen und im welche Maße sie aktiviert werden sollen, wenn die derzeit aktiven Stimuli (1.Tabelle) und der Zusammenhang (links) zusammenpassen.

Beispielsweise verdeutlicht die erste Zeile die Zunahme der Intensität der Emotion „Trauer“ (ET) um 0.01 und die Aktivierung des Objektsuchens (BOS), wenn kein Objekt im Robotersichtfeld erkannt wird und die interaktiven und initiativen Stimuli (iIK, iT) aktiv sind.

Der vollständige Inhalt der Textdatei mit der Datenbank des Wissens für den Roboter ROTTO wird im Anhang A angegeben.

Tabelle 4.3 – Ein Beispiel für die Datenbank des Wissens: die Tabellen der Zusammenhänge und der Aktionssätze

Teil	Zusammenhänge (2. Tabelle)	Trennung	Aktionssätze (3. Tabelle)
Inhalt	zAO iIK iT	->	ET +0.01 BOS
	eEM iIK	->	EF +0.02 BOV
		

4.4 Zusammenfassung

In diesem Kapitel wurden die Aufgaben der Steuerungsebenen für den anthropomorphen Roboter ROTTO diskutiert. Ein besonderes Augenmerk wurde auf die Aufgabe der Mensch-Roboter-Interaktion gelegt. Diese wird ausführlich mit dem Einsatz der „Theory of Mind“ dargelegt. Aufgrund der menschlichen Wahrnehmung seiner Umgebung, am Beispiel von Kleinkindern und Säuglingen, werden die definierten Ereignisse und Zustände in der Umgebung und die Stimuli des Roboters ausgewählt und für die emotionale Steuerung des Roboters ROTTO eingesetzt.

Das Ziel der Untersuchung war die Implementierung des Roboterhaltens, das maximal dem menschlichen Verhalten ähnlich sein kann. Es sollte auch dem Interaktionspartner deutlich der interne Roboterzustand (seine Ziele) gezeigt werden. Deswegen wurden die Grundemotionen und Gesten für die Mensch-Roboter-Interaktion ausgewählt, die dem Mensch von Kindheit an bekannt sind. Darüber hinaus sollte der Roboter auf die externen Anregungen bewusst reagieren. Es werden alle auf den Roboter wirkenden Anregungen als Stimuli bezeichnet, da sie den Roboter zur Aktion stimulieren. Sie sind in zwei Kategorien untergliedert: in die interne und externe. Die Roboterziele werden durch die internen Stimuli definiert. Das Roboterverhalten hängt davon ab, in welchem Maße die externen Stimuli den internen Anforderungen entsprechen. Es existiert eine direkte Abhängigkeit des Verhaltens von den wirkenden Stimuli. Diese wird mit einem universellen Entscheidungssystem realisiert, welches eine Wissensdatenbank besitzt, die die direkten Zusammenhänge zwischen den Stimulussätzen und den Aktionssätze beschreibt. Diese Wissensdatenbank bildet der Charakter des Roboters, da sie das Roboterverhalten aufzeigt.

Der modulare Aufbau des Entscheidungssystems lässt die Wissensdatenbank in Form einer Textdatei konfigurieren. Für den Roboter ROTTO wurde diese Datei erfahrungsgemäß, auf Grund des Wissens über die menschliche Reaktion und über das Verhalten bei den bestimm-

ten Anregungen, erstellt. Darüber hinaus sieht die Universalität des Entscheidungssystems folgende Möglichkeit vor: die Robotersteuerung kann diese Datenbank (Datei) selbstständig durch die neuen Zusammenhänge ergänzen und der Roboter kann neues Verhalten beherrschen bzw. lernen. In diesem Fall sollte der Roboter lernfähig sein, damit er die richtigen Zusammenhänge zwischen den Stimuli und dem eigenen Verhalten bestimmen kann. Aber solche Roboterlernfähigkeit liegt außerhalb der hier vorliegenden Arbeit und ist für weitere Forschungen vorgesehen.

Bei der Implementierung der emotionalen Steuerung für den Roboter ROTTO werden, die im Kapitel 3 beschriebenen, Software und Hardware eingesetzt. Jedes Teilsystem (s. Bild 4.12 und Bild 4.13) wurde in der Entwicklungsumgebung MATLAB/Simulink aufgebaut und in der Simulation analysiert. Die besonderen Teile, wie beispielsweise die Bildverarbeitung, Gesichtserkennung und Farbsegmentierung, wurden mit dem Einsatz der C/C++ Programmiersprachen als einzelne Anwendungen implementiert. Die Hardwareseite des Steuerungssystems wurde in zwei Teile aufgeteilt: den Host-Rechner bzw. den zentralen Rechner und die Technologieplattform NetX10. Auf dem Host laufen zwei Steuerungsebenen ab: die Bildverarbeitung und der Systemkern. Die Technologieplattform NetX10 übernimmt die Aufgaben der Servoebene, in der die Anpassung der Führungsgrößen und die Motorensteuerung implementiert sind. Der Datenaustausch zwischen dem Host-Rechner und der Technologieplattform erfolgt per Ethernet-Netzwerk (wie alle Abtriebsmodule im Roboter). Das Ethernet bildet ein einheitliches Informationsnetzwerk des Steuerungssystems, das sich mit den zusätzlichen Technologieplattformen einwandfrei erweitern lässt. Die eingesetzte Software und Hardware erlaubt die Kombination komplexen Aufgaben (wie die Bildverarbeitung) mit den echtzeitkritischen Aufgaben (wie die Motorensteuerung). Die graphische Softwareentwicklung, welche die Schritte zur Simulation und Analyse beinhaltet, hat innerhalb kürzerer Zeit zur Entwicklung der emotionalen Robotersteuerung beigetragen.

Die Experimente und ihre Ergebnisse zum Einsatz der emotionalen Robotersteuerung in einer Mensch-Roboter-Interaktion, werden für den entwickelten, kontaktfreudigen Roboter ROTTO im Kapitel 5 erläutert.

Kapitel 5

Experimentelle Ergebnisse mit dem humanoiden Roboter ROTTO

In diesem Kapitel werden die experimentellen Ergebnisse aus einer Reihe von Versuchen dargestellt und analysiert. Der kontaktfreudige Roboter ROTTO verdeutlicht seinen Zustand, seine Ziele und Wünsche durch emotionales Verhalten, in dem seine Emotionen an erster Stelle stehen. Aus diesem Grund wird zuerst untersucht, ob die Darstellung und Wiedergabe der Emotionen vom Roboter ROTTO klar und deutlich in der Anwendung in der Mensch-Roboter-Interaktion umgesetzt werden.

5.1 Untersuchung der Anwendbarkeit der Roboteremotionen

Die unterschiedlichen Emotionsarten wurden bereits im Abschnitt 4.2.2.2 diskutiert. Sechs davon wurden als Grundemotionen für den humanoiden Roboter ROTTO ausgewählt. Im Abschnitt 4.3.3 wurden diese Emotionen mit der ausführlichen Beschreibung eingefügt und ihre Darstellung auf dem Robotergesicht in der Tabelle 4.1 gezeigt. Diese Implementierung basiert auf der persönlichen Vorstellung der Roboteremotionen und auf der gesammelten Erfahrung des Entwicklers bzw. des Autors dieser Arbeit. Aus diesem Grund wird zuerst untersucht, ob die implementierten Emotionen überhaupt für die Menschen erkennbar sind und ob sie für die vollständige Mensch-Roboter-Interaktion anwendbar sind.

5.1.1 Beschreibung des Versuches zur Emotionszuordnung

Das Ziel des Versuches lag in der Verdeutlichung der emotionalen Ausdrücke und Darstellung des Roboters ROTTO. Zu diesem Zweck wurde eine Umfrage im Internet organisiert.

Den Teilnehmer wurden sechs Fotos des Robotergesichtes mit verschiedenen Emotionen vorgelegt. Mit jedem der sechs Fotos soll eine bestimmte Emotion dargestellt werden. Jeder Teilnehmer musste die seiner Meinung nach zutreffende Emotion dem jeweiligen Foto zuordnen.

Der Fragebogen wurde folgendermaßen gestaltet (s. Bild 5.1): zuerst wurde in einer kurzen Einleitung über den entwickelten humanoiden Roboter ROTTO berichtet, danach wurde der Umfrageverlauf ausführlich beschrieben und letztendlich wurden die Fotos des Robotergesichtes mit verschiedenen Emotionen präsentiert. Zu jedem Foto wurde die Multiple-Choice-Frage gestellt, welche Emotion der Roboter zeigt, mit folgenden zu wählenden Antworten: Trauer, Unzufriedenheit, Zorn, Freude, Schreck, Staunen. Wenn der Teilnehmer unsicher in der Zuordnung einer Emotion war, so konnte er die Antwort „Schwer zu sagen“ ankreuzen.

Einführung:

Im Rahmen meiner Arbeit wurde ein anthropomorpher (menschenähnlicher) Roboternamens ROTTO entwickelt. Eine der Aufgaben ist die Implementierung der Emotionen, die vom Menschen intuitiv ausgedrückt werden. Sein Verhalten basiert auf den typischen emotionalen Reaktionen auf verschiedene Ereignisse. Durch die Mimik soll der Roboter ROTTO seine eigenen emotionalen Regungen zeigen.

Beschreibung:

Es werden sechs Fotos des Robotergesichtes mit verschiedenen Emotionen dargestellt. Mit jedem Foto des Roboters soll eine entsprechende Emotion ausgedrückt werden. Ordnen Sie die Ihrer Meinung nach zutreffende Emotion dem jeweiligen Foto zu.

Emotionaler Roboter ROTTO:







	<p>(1)</p> <p>Welche Emotion zeigt der Roboter?</p> <ul style="list-style-type: none"><input type="radio"/> Trauer<input type="radio"/> Unzufriedenheit<input type="radio"/> Zorn<input type="radio"/> Freude<input type="radio"/> Schreck<input type="radio"/> Staunen<input type="radio"/> Schwer zu sagen		<p>(2)</p> <p>Welche Emotion zeigt der Roboter?</p> <ul style="list-style-type: none"><input type="radio"/> Trauer<input type="radio"/> Freude<input type="radio"/> Zorn<input type="radio"/> Staunen<input type="radio"/> Schreck<input type="radio"/> Unzufriedenheit<input type="radio"/> Schwer zu sagen
	<p>(3)</p> <p>Welche Emotion zeigt der Roboter?</p> <ul style="list-style-type: none"><input type="radio"/> Schreck<input type="radio"/> Trauer<input type="radio"/> Unzufriedenheit<input type="radio"/> Zorn<input type="radio"/> Staunen<input type="radio"/> Freude<input type="radio"/> Schwer zu sagen		<p>(4)</p> <p>Welche Emotion zeigt der Roboter?</p> <ul style="list-style-type: none"><input type="radio"/> Freude<input type="radio"/> Trauer<input type="radio"/> Unzufriedenheit<input type="radio"/> Zorn<input type="radio"/> Schreck<input type="radio"/> Staunen<input type="radio"/> Schwer zu sagen
	<p>(5)</p> <p>Welche Emotion zeigt der Roboter?</p> <ul style="list-style-type: none"><input type="radio"/> Unzufriedenheit<input type="radio"/> Freude<input type="radio"/> Staunen<input type="radio"/> Trauer<input type="radio"/> Zorn<input type="radio"/> Schreck<input type="radio"/> Schwer zu sagen		<p>(6)</p> <p>Welche Emotion zeigt der Roboter?</p> <ul style="list-style-type: none"><input type="radio"/> Zorn<input type="radio"/> Trauer<input type="radio"/> Schreck<input type="radio"/> Freude<input type="radio"/> Staunen<input type="radio"/> Unzufriedenheit<input type="radio"/> Schwer zu sagen
<p>...</p> <p>Danke für Ihre Teilnahme</p> <p style="text-align: right;">Weiter >>></p>			

Bild 5.1 – Umfrage zum Versuch zur Emotionszuordnung (1 – Freude, 2 – Trauer, 3 – Schreck, 4 – Unzufriedenheit, 5 – Staunen, 6 – Zorn)

5.1.2 Auswertung der Ergebnisse

Für die Umfrage wurden 130 Teilnehmer innerhalb einer Woche befragt. Die Ergebnisse daraus werden im Bild 5.2 veranschaulicht. Es ist zu erkennen, dass alle Roboteremotionen in den meisten Fällen entsprechend der Idee des Entwicklers zugeordnet wurden. Einige der Emotionen wurden fast durchweg richtig (Bild 5.2.a/b/c) interpretiert, andere Emotionen haben einige Befragten zweideutig verstanden (Bild 5.2.d/e/f).

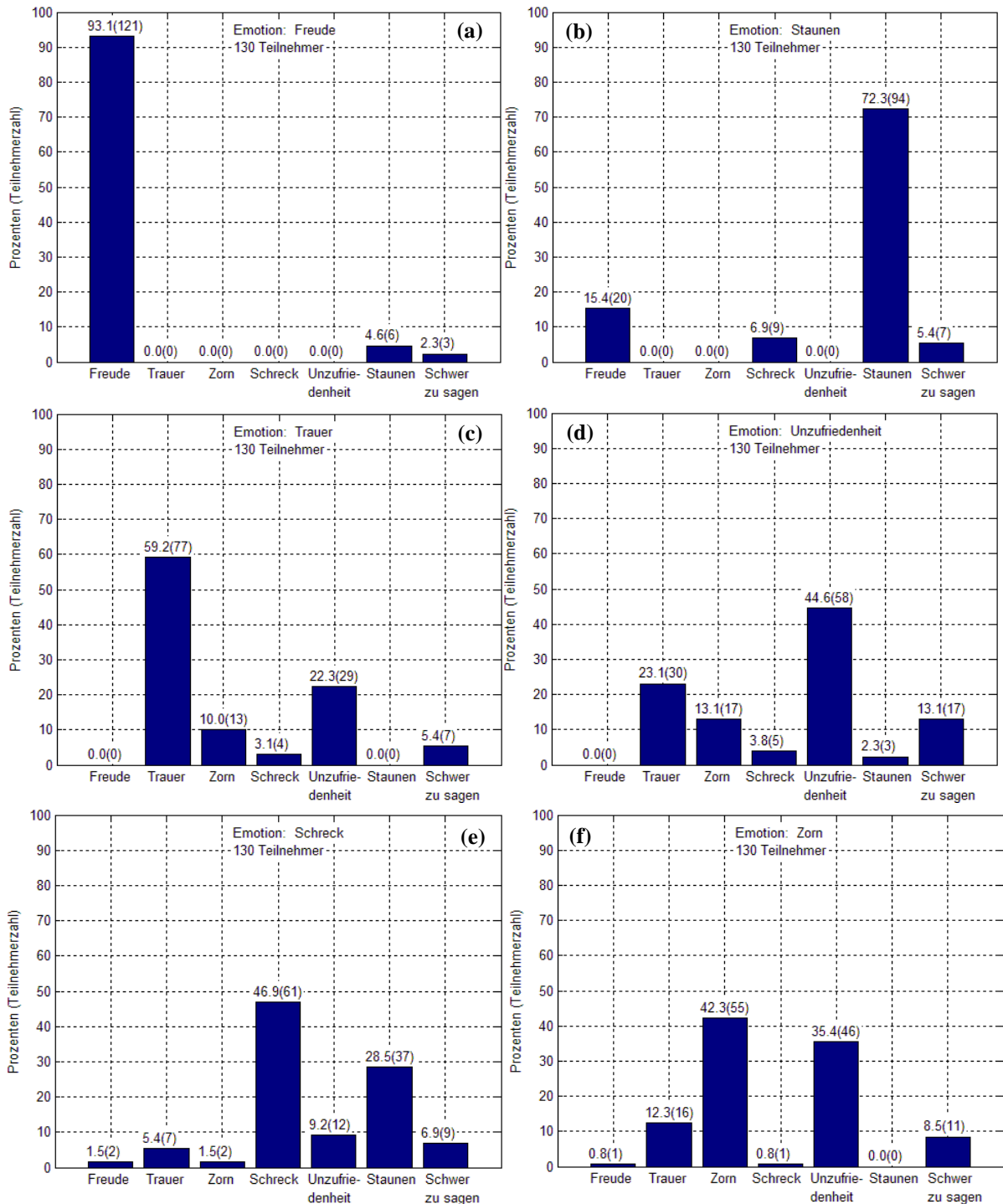


Bild 5.2 – Ergebnisse der Umfrage zur Emotionszuordnung (a – Freude, b – Staunen, c – Trauer, d – Unzufriedenheit, e – Schreck, f – Zorn)

Die Emotionen „Freude“ und „Staunen“ wurden von den meisten Teilnehmern richtig (93,1% und 72,3%) zugeordnet. Trotzdem ist im Bild 5.2.a/b erkennbar, dass die Befragten die beiden Emotionen selten miteinander verwechselt haben. Beide Emotionen werden als positiv bewertet [118] und haben ein gutes Gefühl bei den Teilnehmern hervorgerufen. Aus diesem Grund ist diese Verwechslung nachvollziehbar.

Die Befragten haben die Emotionen „Trauer“ und „Unzufriedenheit“ (Bild 5.2.c/d) „richtig“ zugeordnet, obwohl die Richtigkeit schlechter ist (59,2% und 44,6% entsprechend). Diese zwei Emotionen wurden von den Teilnehmern häufiger vertauscht, weil sie ebenfalls zu den negativen Emotionen gezählt werden. Die dritte negative Roboteremotion „Zorn“ wurde in beiden Emotionen „falsch“ gedeutet (bei „Trauer“ – 10,0% und für „Unzufriedenheit“ – 23,1%). Die negativen Emotionen zeigen einen negativen Roboterzustand und werden deshalb seltener verwechselt.

Die Emotion „Schreck“ wurde in 46,9% Fällen „richtig“ zugeordnet und wurde meistens mit „Staunen“ verwechselt (28,5%). Der Grund dafür ist eine Ähnlichkeit der Emotionen in den statischen Fotos. Bei der Emotion „Schreck“ wird der Kopf angehoben (s. Bemerkungen in der Tabelle 4.1), was in den statischen Bildern nicht eindeutig zu erkennen ist (s. Bild 5.1-3/5).

Die Emotion „Zorn“ wurde mehrmals mit „Unzufriedenheit“ (35,4%) und „Trauer“ (12,3%) vertauscht. Der Grund dafür ist dasselbe wie der bereits beschriebene. Nichtsdestotrotz wurde für die Emotion „Zorn“ 42,3% „richtige“ Zuordnungen gezählt und somit lässt sich erschließen, dass diese Emotion gut erkennbar ist.

Zusammenfassend ist in diesem Versuch zu betonen, dass die für den Roboter ROTTO entwickelten Emotionen von den Menschen meist richtig gedeutet werden und demnach in der Mensch-Roboter-Interaktion akzeptabel einsetzbar sind. Es ist weiterhin zu erwähnen, dass die in der Umfrage angebotenen Emotionen manchmal zweideutig und ungenau zugeordnet wurden. Zum Einen passiert dies aufgrund der gleichen Ausrichtung einiger Emotionen (positiv/negativ), zweitens wurden einige Emotionen, die eine zusätzliche Kopfneigung oder Kopfhebung beinhalten, in den Bildern unvollständig dargestellt, weswegen ein Teil der emotionalen Information verloren ging. Drittens kann das bewegliche Gesicht im Vergleich zum statischen mehr emotionale Informationen ausdrücken. Deswegen wird das Robotergesicht in der realen Interaktion die Emotionen wesentlich deutlicher darstellen müssen. Aufgrund der konstruktiven Merkmale, ist das Robotergesicht eher auf eine positive Außenansicht (rote Lippen, rundliches Gesicht, große Augen usw.) ausgerichtet. Das hat zur Folge, dass bei der Darstellung der negativen Emotionen gewisse Schwierigkeiten auftreten.

5.2 Nachbildung des emotionalen Verhaltens und Akzeptanz der Mensch-Roboter-Interaktion

Basierend auf den untersuchten Emotionen wurde in diesem Unterkapitel eine echte Mensch-Roboter-Interaktion nachgebildet. Der Zweck ist es die Roboterakzeptanz während der Interaktion mit einer Person zu untersuchen.

5.2.1 Beschreibung des Versuches zur Mensch-Roboter-Interaktion

Am Versuch haben insgesamt 12 Personen teilgenommen. Zuerst wurde jedem Teilnehmer über den Roboter ROTTO und seinen emotionalen Verhaltensweisen berichtet und sie wurden gebeten, auf dieses emotionalen Roboterverhalten zu achten. Danach hat jeder Teilnehmer eine oder wahlweise mehr Phasen der Interaktion mit dem emotionalen Roboter durchgeführt. Eine Interaktionsphase dauert bis an, bis der Roboter den Zustand „Erholung“ erreicht.

Der Versuch beginnt mit der Situation, dass sich eine Person, mit einem Spielzeug in der Hand, dem Roboter gegenüber steht. Der Roboter befindet sich am Anfang im Zustand „Erholung“ und wacht innerhalb der ersten fünf Sekunden auf. Mit dieser Situation werden sämtliche Versuche der Mensch-Roboter-Interaktion gestartet (Bild 5.3).



Bild 5.3 – Mensch-Roboter-Interaktion

Während der Interaktionsphase kann die Person sich bewegen, um die Aufmerksamkeit des Roboters auf sich zu ziehen. Der Roboter ROTTO erkennt das menschliche Gesicht und reagiert auf dieses. Ebenso kann die Roboteraufmerksamkeit auf einen bunten Gegenstand bzw. Spielzeug (z.B. gelbe Maus) und seine Bewegungen gelenkt werden. Hierbei sind Roboterverhalten und die emotionalen Reaktionen des Roboters auf verschiedene Ereignisse deutlich erkennbar. Die beschriebene Interaktion wird an einem realen Beispiel detailliert erklärt.

Zwecks ausführlicher Beschreibung des Versuches wurden die Roboterzustände mit Hilfe des zuvor beschriebenen „Debug eines Prozesses“ (s. Unterkapitel 3.2.3.2) innerhalb solcher Interaktionsphase aufgenommen. Sie werden für die detaillierte Analyse in Bild 5.4 und Bild 5.5 präsentiert.

Nach dem Erwachen waren die interaktiven (iIK) und initiativen (iT) Stimuli gleichermaßen aktiv, deshalb hat der ROTTO nach einem Menschen oder einem Gegenstand (BOS – Abschnitt **a**) gesucht. Wenn ein Mensch (ein Gesicht) vom Roboter erkannt wurde (eEM, zAM – **b**), hat der Roboter darüber (ESt) gestaunt und demnach versucht die menschliche Aufmerksamkeit (BAE) auf sich zu lenken. Der interaktive Stimulus wurde angeregt (IIK→iIK – **b**, **c**, **d**). ROTTO hat seine Aufmerksamkeit dem Menschen (BOV1-Mensch) gewidmet und hat

sich (EF), so lange der Mensch im Sichtfeld war (b, c), gefreut. Im Abschnitt d hat ROTTO den Menschen aus dem Sichtfeld verloren. Daraufhin wurde er zuerst zornig und danach traurig (e). Er hat erneut nach dem Menschen gesucht (e), weil der initiative Stimulus (iIK) stark angeregt wurde.

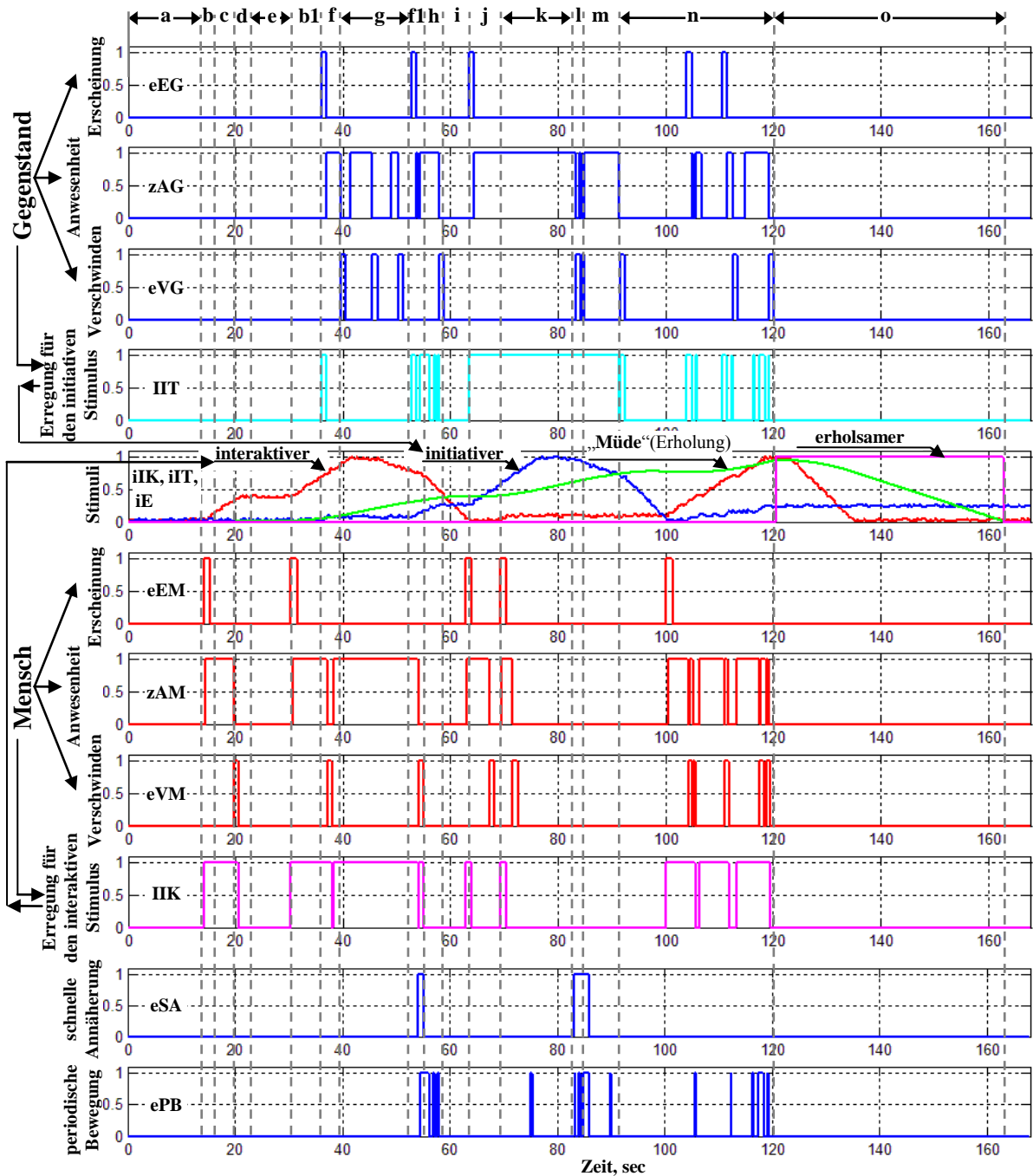


Bild 5.4 – Stimuli des Roboters während einer Mensch-Roboter-Interaktion

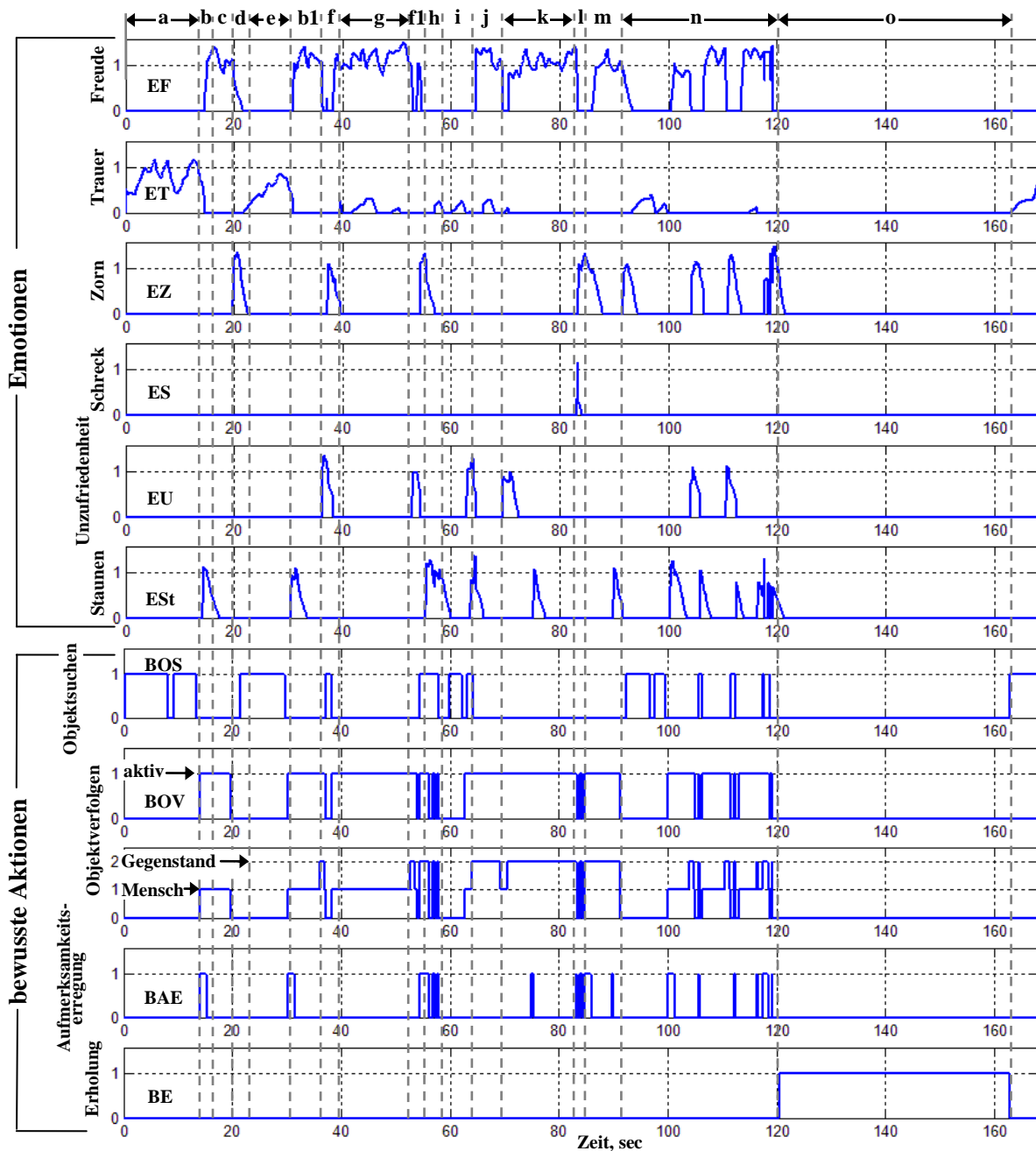


Bild 5.5 – Emotionen und das Verhalten des Roboters während einer Mensch-Roboter-Interaktion

Im Abschnitt **b1** wurde ein menschliches Gesicht wieder entdeckt (ähnlich dem Abschnitt **b**). Das menschliche Gesicht bleibt so lange fokussiert im Robotersichtfeld, bis dem Roboter ein Spielzeug gezeigt wurde (eEG, zAG – **f** und **f1**). Daraufhin fokussierte der Roboter dieses Spielzeug (BOV2-Gegenstand) und wurde unzufrieden (EU), da der interaktive Stimulus derzeit noch dominierte. Aus diesem Grund wendet sich der Roboter anschließend wieder dem Menschen zu (BOV1 – **g**).

Im Abschnitt **h** hat der Teilnehmer dem Roboter das schwingende Spielzeug (ePB) gezeigt, um seine Aufmerksamkeit darauf zu lenken. ROTTO hat über das Spielzeug gestaunt (ESt) und sein initiativer Stimulus (iIT→iIT) wurde angeregt. Entsprechend seinen anregend-temporären Ursprung (s. Unterkapitel 4.2.2.1) wurde der interaktive Stimulus unterdrückt. Bereits im Abschnitt **i** hat der initiative Stimulus dominiert, weshalb der Roboter nach dem Spielzeug gesucht hat (BOS) und mit der Anwesenheit des Menschen unzufrieden war (eEM, EU – **i**, am Ende). Der Roboter fokussiert sich auf das wieder entdeckte Spielzeug (BOV2 – **j**) und bringt dabei seine Freude zum Ausdruck (EF). Am Anfang des Abschnittes **k** hat ROTTO den Menschen gesehen und daraufhin seine Abneigung dagegen gezeigt (EU). Dann wurde der Abstand vom Spielzeug zum Roboter verringert und das Spielzeug wurde vor seinem Sichtfeld rasch bewegt (**l**). Der Roboter hat sich zuerst erschrocken (ES), danach aber wieder das Spielzeug fokussiert (**m**).

Im Abschnitt **n** hat wieder der interaktive Stimulus (iIK) dominiert, deswegen war ROTTO mit dem Menschen gegenüber freundlich gestimmt und dem Spielzeug dahingegen abgeneigt. Im Abschnitt **o** wurde der Roboter „müde“. Der für die Erholung verantwortliche Stimulus (iE) wurde sehr stark angeregt und der Roboter erreichte dadurch den Zustand „Erholung“ (Schlafen). Damit wurde eine Interaktionsphase abgeschlossen.

Wenn ROTTO eingeschlafen ist und der Teilnehmer um keine neuen Interaktionsphase gebeten hat, wurde der Versuch beendet. Abschließend wurde dem Probanden der Fragebogen zur Untersuchung der Akzeptanz des Roboters ROTTO gereicht. Die Gestaltung des Fragebogens wird im Bild 5.6 illustriert.

Zuerst wurde nach dem Teilnehmernamen gefragt und das Thema der Untersuchung genannt. Der Hauptteil der Umfrage besteht aus sieben Fragen. Jede Frage widmet sich verschiedenen Seiten der menschlichen Wahrnehmung des Roboters (bzw. den Emotionen des Roboters, seines Verhaltens und seines Charakters). Ist sein Verhalten und sein Charakter dem Menschen interessant, bewusst und deutlich gemacht worden? Welchen Charakter besitzt er? Ist der Roboter der Meinung des Teilnehmers nach emotional? Sind seine Emotionen klar zu deuten? In der letzten Frage sollte der Roboter in Bezug auf seine Verhaltensweisen und Emotionen mit dem Verhalten eines Menschen im Kleinkindalter verglichen werden. Damit wird geprüft, ob das Roboterverhalten mit der bei der Entwicklung ausgewählten Jahresbasis übereinstimmt.

Ihr Name: _____

Befragung zur Untersuchung der Akzeptanz des Roboters ROTTO

1. Finden Sie den Roboter und sein Verhalten interessant?

Nein					Ja
0	1	2	3	4	5

2. Ist das Verhalten des Roboters bewusst gemacht worden?

Nein					Ja
0	1	2	3	4	5

3. Wie deutlich werden die Wünsche des Roboters gezeigt? (Bzw. er will einen Menschen oder ein Spielzeug sehen)

undeutlich					sehr deutlich
0	1	2	3	4	5

4. Ist der Roboter freundlich, neutral oder ungnädig?

freundlich		neutral		ungnädig
+5	+2,5	0	-2,5	-5

5. Wie emotional ist der Roboter?

Zu wenig	wenig	normal	mehr	überflüssig
0	2,5	5	2,5	0

6. Wie verständlich sind die vom Roboter ausgedrückten Emotionen?

unverständlich		neutral		verständlich
-5	-2,5	0	2,5	5

7. Welchem Alter des Kindes könnten Sie den Roboter nach seinem Verhalten entsprechend zuordnen?

Jung				Alt
wenige Monate	1 Jahr	2 Jahre	3 Jahre	mehr

Bild 5.6 – Umfrage zur Untersuchung der Akzeptanz des Roboters ROTTO

5.2.2 Auswertung der Ergebnisse

In der Auswertung der Versuchsergebnisse werden alle Fragen der Umfrage zur Untersuchung der Akzeptanz des Roboters ROTTO erklärt und diskutiert. Die Statistik der Umfrage wurde zusammengefasst und zur Analyse graphisch dargestellt.

Frage: *Finden Sie den Roboter und sein Verhalten interessant?*

In der ersten Frage sollte bestimmt werden, ob ein Mensch den Roboter wahrnimmt. Die Mehrheit der Teilnehmer hat das Verhalten des Roboters als interessant bewertet (67%). Aus Bild 5.7 ist ersichtlich, dass nur einige Stimmen das Roboterverhalten als wenig interessant betrachtet haben (33%). Dieses Ergebnis weckt die Hoffnung, dass die allgemeine Wahrnehmung des Roboters ROTTO sich auf einem hohen Niveau befindet.

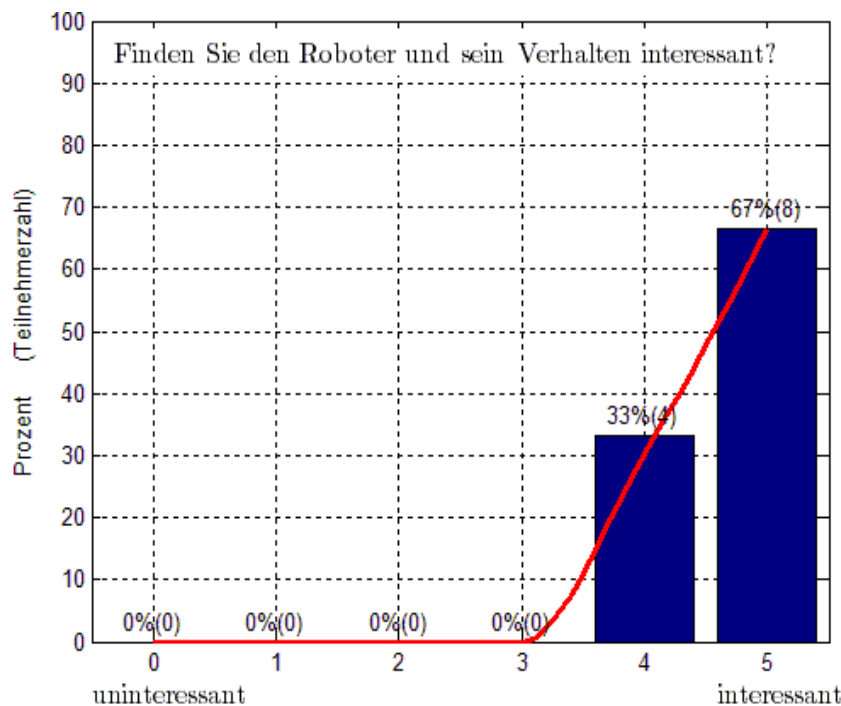


Bild 5.7 – Allgemeine Wahrnehmung des Roboters ROTTO

Frage: *Ist das Verhalten des Roboters bewusst gemacht worden?*

Mit dieser Frage sollte beantwortet werden, ob ein Mensch im Roboter ein bewusstes Verhaltensweise erkennen kann. 8% der Teilnehmer haben entschieden, dass das Roboterverhalten kaum als bewusst (bzw. neutral) zu erkennen ist. 17% sind der Meinung, dass es deutlich als erkennbar ist. Die Mehrheit (75% der Teilnehmer) hat die Noten 3 und 4 (von max. 5) für bewusstes Roboterverhalten vergeben und waren der Meinung, dass fast alle Aktionen von ROTTO zweckgebunden sind. Aus diesen Werten lässt sich zusammenzufassen, dass die Akzeptanz des Roboterverhaltens hoch ist, weil die Menschen die Roboteraktionen deutlich verstehen.

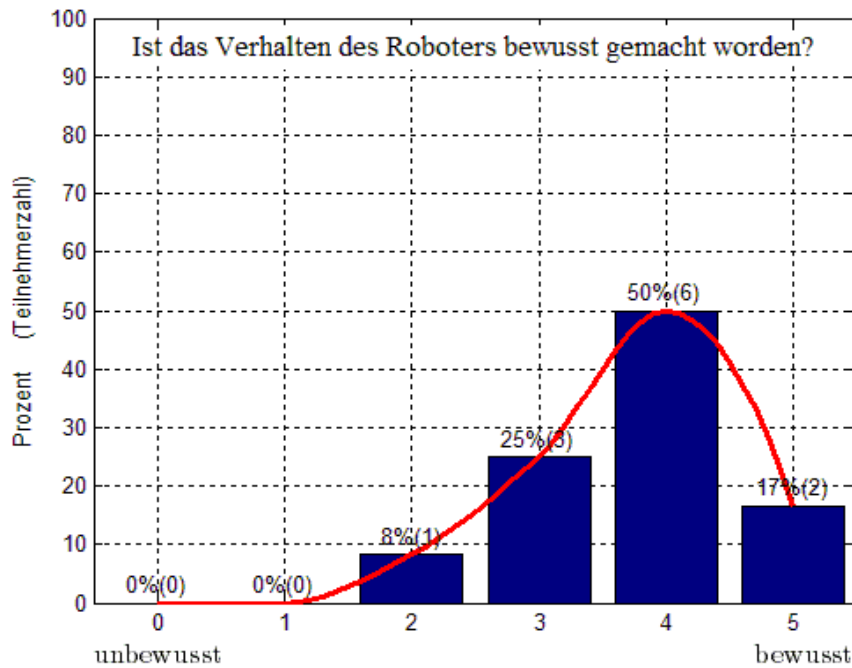


Bild 5.8 – Wahrnehmung des Roboterverhaltens

Frage: *Wie deutlich werden die Wünsche des Roboters gezeigt?*

Diese Frage sollte zeigen, wie klar ROTTO durch sein Verhalten seine eigene Ziele und Wünsche dem Dialogpartner gegenüber ausdrücken kann. Die Frage, ob dem Dialogpartner deutlich gemacht wurde, dass der Roboter entweder ihn selbst sehen wollte oder nach dem Spielzeug gesucht hat. Bild 5.9 zeigt, dass nur für 8% der Teilnehmer die Roboterwünsche wenig deutlich waren (2 Punkten von 5). Andere Stimmen bewerten den Ausdruck der Wünsche als deutlich (3 – 5 Punkte). Die Auswertung ergab, dass die Bewertung „fast deutlich“ (4 Punkten) am häufigsten gewählt wurde. Sie bedeutet, dass der Roboter die Wünsche meistens deutlich durch sein Verhalten und seine Emotionen ausdrücken konnte.

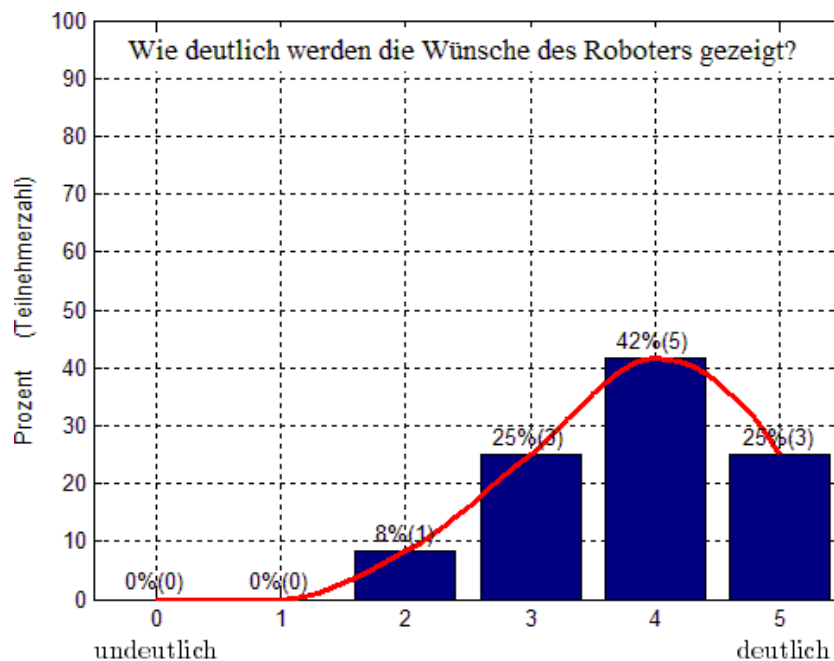


Bild 5.9 – Deutlichkeit der Wünsche des Roboters

Frage: *Ist der Roboter freundlich, neutral oder ungnädig?*

Mit dieser Frage galt es zu bestimmen, ob der Mensch im Roboter ein Charakter sieht und welcher Charakter (positiv / negativ) dabei erkannt wurde. Das Bild 5.10 zeigt, dass 8% der Teilnehmer keinen Charakter im Roboter entdeckt haben, er war für sie neutral. Nach der Meinung anderen 17% Teilnehmern war ROTTO genug freundlich. Die Mehrheit (75% der Teilnehmer) ist der Meinung, dass der Roboter freundlich ist. Der Grund dafür liegt einerseits in der positiv gestalteten Außenansicht des Robotergesichtes und andererseits in der Datenbank des Wissens (s. Anlage A). Sie wurde so entworfen, dass der Roboter sich meistens über das gewünschte Objekt freut, wenn er es sieht (Bild 5.5 – Emotion „Freude“). Aus diesem Ergebnis lässt sich zusammenzufassen, dass die Menschen ROTTO bei der nahen Interaktion nicht nur als interessant sondern auch als freundlich bewerten.

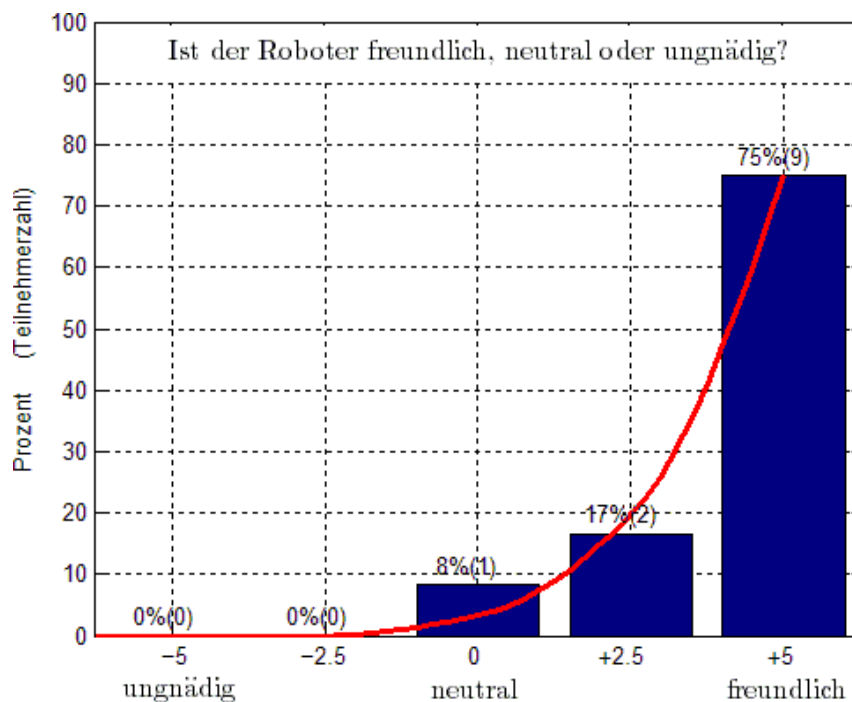


Bild 5.10 – Bewertung des Robotercharakters

Frage: *Wie emotional ist der Roboter?*

Diese Frage befasst sich mit der Untersuchung, wie stark die Roboteremotionen für die menschliche Wahrnehmung erkennbar sind. Bild 5.11 zeigt, dass auf 8% der Teilnehmer ROTTO wenig emotional wirkt. Weitere 8% der Teilnehmer bewerten seine Emotionen als überflüssig. Die häufigsten Bewertungen erhielt ROTTOs Emotionalität zwischen den Werten „normal“ und „mehr“. Diese Bewertung zeigt, dass solche Emotionalität für die Mensch-Roboter-Interaktion ausreichend (und manchmal fast überflüssig bzw. unnötig) ist.

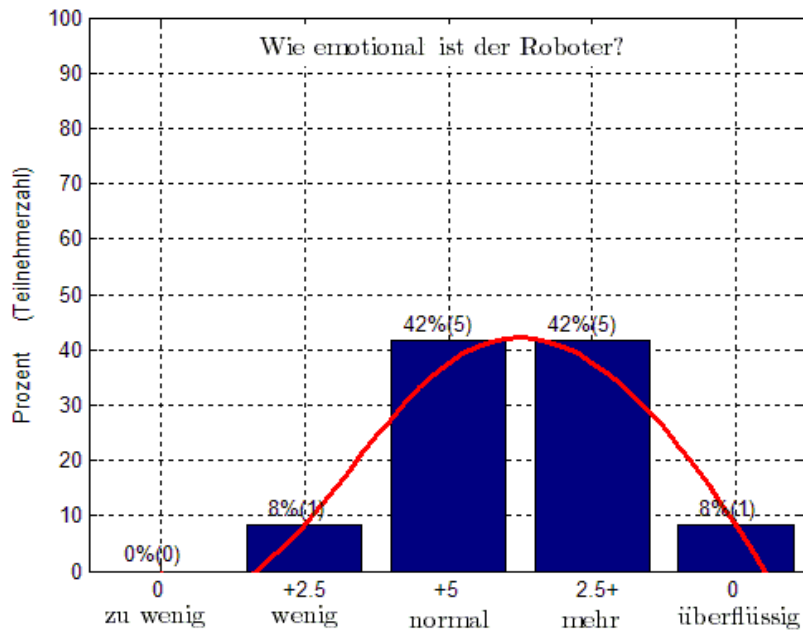


Bild 5.11 – Emotionalität des Roboters

Frage: *Wie verständlich sind die vom Roboter ausgedrückten Emotionen?*

Diese Frage wiederholt teilweise den ersten Versuch zur Untersuchung der Roboteremotionen und zeigt, ob die emotionale Reaktion des Roboters während der Interaktion verständlich und adäquat war. Bild 5.12 zeigt, dass nur 8% der Teilnehmer daran gezweifelt haben. Für sie waren die Emotionen wenig verständlich und wurden mit anderen vertauscht (auch wie im ersten Versuch). Für die anderen Teilnehmer war der Roboter emotional klar und deutlich. Hier ist zu bemerken, dass, in der realen Roboter-Interaktion, der Mensch verstehen kann, ob der Roboter mit seinem Verhalten die Erwartungen erfüllt. Das Ergebnis zeigt, dass der Roboter auf die von den Teilnehmern getätigten Aktionen verständlich und adäquat reagiert hat.

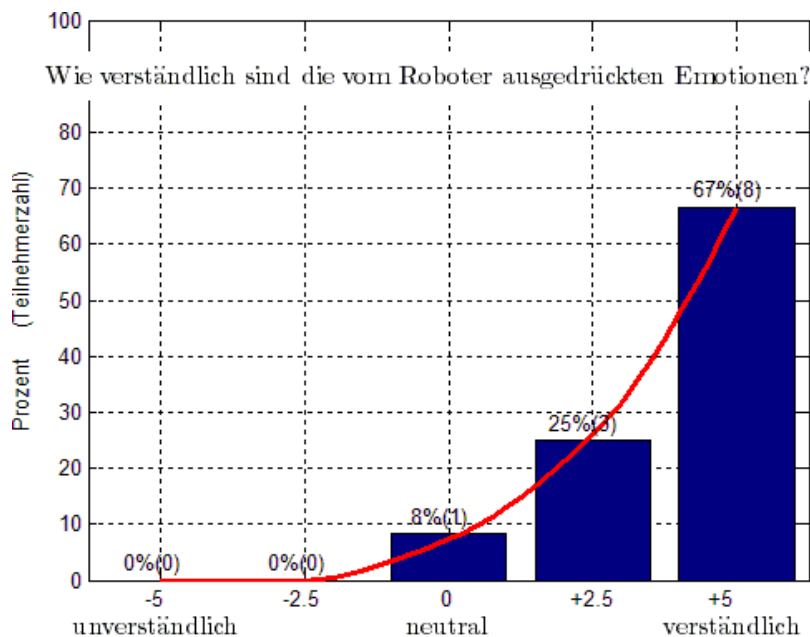


Bild 5.12 – Verständlichkeit der Roboteremotionen in der Interaktion

Frage: *Welchem Alter des Kindes könnten Sie den Roboter nach seinem Verhalten entsprechend zuordnen?*

Laut der oben diskutierten Theory of Mind (s. Unterkapitel 4.2.1) wurde der Roboter als emotional und im Verhalten als 1,5-jähriges Kind (18 Monate) eingestuft. Mit der letzten Frage wurde geprüft, ob für die Teilnehmer diese Altersangabe leicht ersichtlich war. Das Bild 5.13 zeigt, dass die Stimmenmehrheit sich zwischen den Altersangaben „1 Jahr“ und „2 Jahre“ (42% und 25%) befindet und der Höchstwert liegt beim vorgesehenen Altersbereich. Dieses Ergebnis zeigt, dass die Teilnehmer im Verhalten des Roboters das Verhalten eines 1,5-jährigen Kindes erkannt haben. Es entspricht vollständig dem implementierten Verhalten des Roboters.

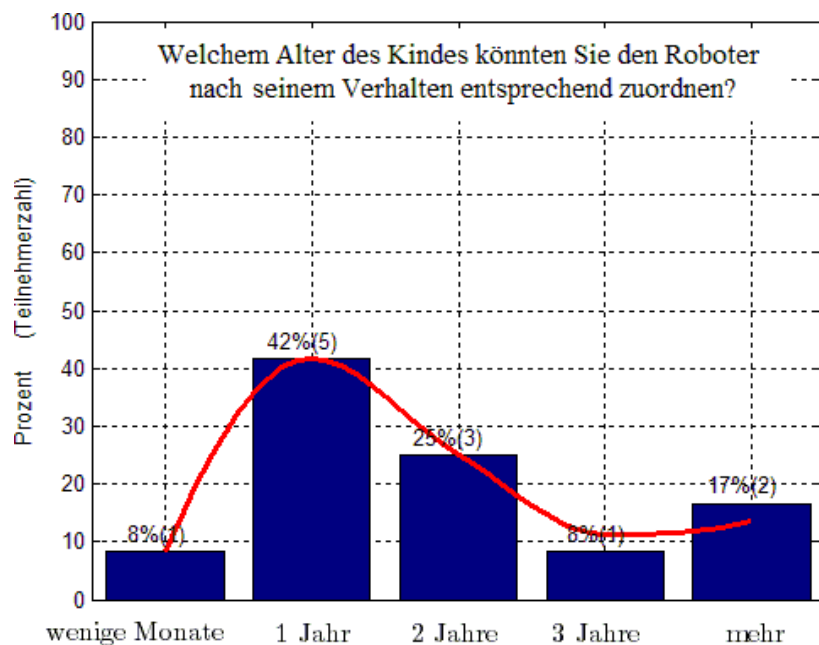


Bild 5.13 – Zugeordnetes Alter des Roboters

Zusammenfassend ist in diesem Versuch zu betonen, dass alle Teilnehmer des Versuches sowohl das Verhalten und den Charakter des Roboters ROTTO als auch seine Emotionen gut akzeptiert haben. Dafür sprechen die dargestellten Ergebnisse der Umfrage, in welchen jede Frage die Bestandteile der Roboterakzeptanz berührt hat.

5.3 Zusammenfassung

Die dargestellten experimentellen Untersuchungen zeigen anhand einer realen Mensch-Roboter-Interaktion, dass der Roboter ROTTO seine Emotionen deutlich ausdrücken kann und diese von den meisten Menschen eindeutig akzeptiert werden.

Das für den Roboter ROTTO entwickelte Emotionssteuerungssystem basiert auf dem universalen Entscheidungssystem und wurde in diversen Interaktionsexperimenten erprobt. Das Roboterverhalten ist für die Menschen akzeptabel und für eine nahe Mensch-Roboter-Interaktion gut geeignet. Das Konzept im entwickelten Entscheidungssystem lässt sich in einen beliebigen Robotercharakter bzw. das gewählte Verhalten implementieren. Die zusätzlichen „Gefühle“ (wie z.B. Gehör, Tastgefühl u.a.) lassen sich auf einfache Weise in das Verhaltenssteuerungssystem des Roboters integrieren.

Kapitel 6

Zusammenfassung und Ausblick

6.1 Zusammenfassung

Die vorliegende Arbeit unterliegt dem Schwerpunkt der der Konstruktion und Steuerung des mobilen anthropomorphen emotionalen Roboters ROTTO und befasst sich mit dem systematischen Ansatz zu seiner Entwicklung unter dem Einsatz der vorgeschlagenen Architekturen für die Software und Hardware.

Aus funktioneller Sicht ist zu erwähnen, dass die entworfene Roboterkonstruktion sich durch eine einfache und eine, sowohl für die Aufgaben der Mensch-Roboter-Interaktion als auch für die statischen und dynamischen Lokomotionsaufgaben ausreichende, Kinematik mit optimalen Verhältnissen und größeren Arbeitsbereichen auszeichnet. Die Arbeitsweise und Platzierung der Antriebseinheiten ist sehr anthropomorph gehalten. Die elektrischen Antriebe sind in Bezug auf Kosten, Energieverbrauch und Leistungsdichte effizient ausgelegt. Die integrierte Sensorik erfasst alle für die definierten Aufgaben (z.B. Interaktion und Lokomotion) notwendigen Informationen, wie beispielsweise Umgebungsmerkmale (Videokameras), Körperkonfiguration, künstliches Gleichgewicht und Bodenreaktionskräfte. Dank des systematischen, iterativen Entwicklungsansatzes lässt sich behaupten, dass der Roboter ROTTO, von allen Seiten optimal entwickelt worden ist. Er folgt im Großen und Ganzen den anthropomorphen Verhältnissen und ist in der Lage die den Menschen nahen Aufgaben, wie Interaktion, dynamisches Gehen, Laufen u.ä., durchzuführen.

Für den Entwurf des anthropomorphen Roboters als komplexes mechatronisches System, wurde aus verschiedenen Methoden das V-Entwurfsmodell gewählt und mit den neuen Ansätzen, zur Beschleunigung des Entwurfprozesses, erweitert. Diese Ansätze sind durch die neuentwickelten Technologieplattform (ASIC mit mehreren Prozessorkernen und mit den spezifischen Hardwaremodulen) möglich geworden. Die Technologieplattform bildet einen wichtigen und untrennbaren Teil der Hardwarearchitektur des Roboters. Die entsprechende Softwarearchitektur besteht aus zwei Teilen. Die Architektur nutzt „Rapid Control Prototyping“-fähige Softwarewerkzeuge, die sowohl eine schnelle grafische Entwicklung eines Steuerungssystems als auch seine effiziente Modellierung, Verifizierung und Fehlerbeseitigung erlaubt. Auf der anderen Seite wurde die Softwarearchitektur durch die im Rahmen dieser Arbeit entworfenen Werkzeuge, zur schnellen und qualitativen Codegenerierung, für die genannte Technologieplattform erweitert. Das vorgestellte Verfahren hat es möglich gemacht, die Roboterentwicklung mit dem Emotionssteuerungssystem innerhalb kurzer Zeit bis zu einem Funktionsprototyp voranzubringen.

Neben dem systematischen Entwurf des Roboters ROTTO wurde in der vorgeschlagenen Softwarearchitektur ein hierarchisches modulares Steuerungssystem entwickelt, das sich als sehr effizient für einen zweibeinigen, emotionalen, mobilen Roboter erwies. Die vorgeschlagene technische Lösung zum Aufbau des Steuerungssystems basiert auf den vernetzten leistungsfähigen Recheneinheiten mit einem zentralen Rechner (Host). Zu den wichtigsten Eigenschaften dieses Konzepts gehören: eine einfache und effiziente Entwicklung, hohe Flexibilität und Erweiterbarkeit, sowie ein leistungsfähiges Kommunikationssystem zwischen den einzelnen Einheiten. Die Verwendung einer Technologieplattform mit einem Kommunikationssystem ermöglichte die Realisierung neuer Regelungskonzepte, bei dem alle zeitkritischen (weniger als 1ms, z.B. Stromregelkreise und Sicherheit) Komponenten des Steuerungs- bzw. Regelungssystems auf den Technologieplattformen bzw. dezentralen Recheneinheiten und die weniger kritischen (1ms und mehr) Komponenten auf den Zentralrechner verlagert worden sind. Die aufgabenspezifischen Regelkreise wurden über dem Kommunikationsnetzwerk geschlossen, wobei der Zykluszeitraum nicht größer ist als 1ms. Das resultiert in einem exzellenten Synchronismus der betroffenen Antriebe und einer schnellen Realisierung und Prüfung verschiedener Steuerungs- und Regelungskonzepte. Die dargestellte Inbetriebnahme des Emotionssteuerungssystems und dessen quantitative Untersuchungen, insbesondere für die Mensch-Roboter-Interaktion, beweisen die hohe Hochwertigkeit der gesamten Hardware- und Softwarearchitekturen zur Entwicklung des Steuerungssystems sowie dessen einzelner Komponenten.

Zur Bewertung der vorgeschlagenen Konzepte wurden die verschiedenen Aufgaben des Roboters ROTTO dargestellt und eine der Aufgaben - die emotionale Interaktion mit den Menschen - für die Implementierung gewählt. Die Aufgabe der Mensch-Roboter-Interaktion stellt sich als ebenso wichtig dar, wie z.B. die dynamische Roboterfortbewegung, weil der Roboter mit den Menschen mittels der ihnen von Kindheit an bekannten Sprache der Emotionen und Gesten kommunizieren und damit in einer menschlichen Umgebung akzeptiert werden konnte.

Die effiziente Mensch-Roboter-Interaktion ist durch das in dieser Arbeit entworfene Emotionssteuerungssystem des Roboters ROTTO möglich geworden. Basierend auf den Untersuchungen im Gebiet der Psychologie („Theory of Mind“) wurden alle Aktionen in der Roboterumgebung auf die Ereignisse und Zustände aufgeteilt und als externe Stimuli des Roboters bezeichnet. Ebenso wurden die internen Stimuli, die die Wünsche und Ziele des Roboters bestimmen, implementiert. Die externen und internen Stimuli bilden die Erregungen für den Roboter. Andererseits kann ROTTO mit seinem Verhalten und Emotionen auf die Umgebung bzw. den Interaktionspartner einwirken bzw. durch das emotionale Verhalten seine Wünsche und Ziele verdeutlichen. Der Kern des Emotionssteuerungssystems bildet ein universelles Entscheidungssystem, das das Roboterverhalten den aktiven Erregungen anpasst. Das Entscheidungssystem wird in einfacher Sprache beschrieben und bildet die Datenbank des Wissens für den Roboter ROTTO. Der Roboter verhält sich dieser Datenbank entsprechend, welche die Merkmale seines Charakters umfasst. Sie kann an neue Erregungen und Verhaltensweise schnell und einfach angepasst werden. Diese Eigenschaft verdeutlicht die Effizienz und die Universalität des entworfenen Entscheidungssystems und des gesamten Emotionssteuerungssystems des Roboters ROTTO.

Die in der vorliegenden Arbeit dargestellten Konzepte und Verfahren zur Mensch-Roboter-Interaktion, wurden in simulativen und praktischen Experimenten untersucht. Hier wurde zuerst die Deutlichkeit der vom Roboter ausgedrückten Emotionen untersucht und als Zweites die Roboterakzeptanz in der realen Mensch-Roboter-Interaktion nachgewiesen.

Der entwickelte Prototyp des zweibeinigen mobilen Roboters verkörpert somit zukünftig eine solide Testplattform, sowohl für die Entwicklung fortgeschrittener Algorithmen der Mensch-Roboter-Interaktion, die die Methoden des Selbstlernens und der Menschenwiedererkennung verinnerlicht, als auch für die Manipulationsaufgaben (Arme und Hände), für verschiedene Steuerungs- und Regelungsstrategien zur Fortbewegung und Lokomotion.

6.2 Ausblick

Die vorliegende Arbeit behandelt schwerpunktmäßig die Problemstellungen der konstruktiven Gestaltung des humanoiden mobilen Schreitroboters ROTTO und seine Steuerungsaspekte. Durch die Entwicklung und Verwendung optimierter Systemkomponenten, ist der vorgestellte humanoide Roboter ROTTO den anderen aktuellen Prototypen weltweit in vielerlei Hinsicht überlegen. Damit stellt ROTTO eine solide Grundlage für die Entwicklung zukünftiger leistungsfähiger Laufroboter dar.

Bei der Entwicklung des Oberkörpers für den humanoiden kontaktfreudigen Roboter ROTTO wurde eine Analyse der menschlichen Wahrnehmung des mechanischen Wesens hinsichtlich der Außenansicht bzw. des Gesichtes durchgeführt. Es werden drei Arten des Roboter Gesichtes unterscheiden: das abstrakte, realitätsnahe und ikonische Roboter Gesicht. ROTTO wurde mit einem mittleren, ikonisch-realitätsvollem Gesicht gestaltet. Darüber hinaus wurde ein großer Teil der Aufmerksamkeit auf die Beweglichkeit der Gesichtsteilen gerichtet. Dadurch wurde die hohe Akzeptanz in einer nahen Mensch-Roboter-Interaktion erreicht, was die Ergebnisse der durchgeführten Versuche bestätigt haben. Für die weitere Roboterintegration in die menschliche Umgebung können sowohl das emotionale Roboter Verhalten, durch die neue Gesichtsteile, erweitert als auch neue Roboterfähigkeiten, wie das Selbstlernen und Menschenwiedererkennung, implementiert werden.

Da sich humanoide, mobile, kontaktfreudige Roboter am besten für die menschliche Umgebung eignen und eine erhöhte Akzeptanz im menschlichen Umfeld erfahren, wird mit Hochdruck an der Entwicklung solcher Roboter weltweit gearbeitet. Das trägt zum besseren Verständnis von menschlichen Aktionen, wie Interaktion, Gehen, Laufen u.ä. bei und ermöglicht die Verbesserung besonderer Mechanismen, wie die Hausanlagen mit einer intuitiven Bedienung, medizinischen Prothese und Exoskeletten. Dennoch ist es, für den Einsatz solcher Roboter in einer Umgebung zusammen mit Menschen, notwendig eine weitere Entwicklung und Erforschung der Intelligenz, der Fähigkeit zu sprechen und die menschliche Sprache zu verstehen, sowie der Fähigkeit sich mit den dem Menschen bekannten emotionalen Ausdrucksweisen und Körpergesten erklären zu können und sie zu verstehen. Diese Kommunikationsformen und Gestik, wie beispielsweise die Bedienungsweise für ein Gerät bzw. eine Anlage, ist den Menschen von Geburt an bekannt, weswegen sie in unserem Alltagsleben als intuitive Steuerung der Haushaltsgeräte eingesetzt werden könnte.

Anhang A

Datenbank des Wissen für den Roboter ROTTO

A.1 Bemerkungen

„nur_diese“	Ein Hilfsoperator, der nur die Anwesenheit der angegebenen Erregungen berücksichtigt, obwohl andere Erregungen angeregt werden können.
„//“	Ein Hilfsoperator, der eine Kommentarzeile kennzeichnet.
„iK<iT“	Der initiative Stimulus dominiert
„iK>iT“	Der interaktive Stimulus dominiert

A.2 Abkürzungen

A.2.1 Externe Stimuli, Ereignisse (sie werden mit dem Präfix „e“ markiert)

„eEM“	Erscheinung eines Menschen
„eEG“	Erscheinung eines Gegenstandes
„eVM“	Verswinden eines Menschen
„eVG“	Verswinden eines Gegenstandes
„eSA“	Scharfe Annäherung eines Objektes (bzw. eines Menschen oder eines Gegenstandes)
„ePB“	Schnelle und periodische Bewegung eines Objektes im Robotersichtfeld

A.2.2 Externe Stimuli, Zustände (sie werden mit dem Präfix „z“ markiert)

„zAO“	Abwesenheit eines Objektes
„zAM“	Anwesenheit eines Menschen
„zAG“	Anwesenheit eines Gegenstandes

A.2.3 Interne Stimuli (sie werden mit dem Präfix „i“ markiert)

„iK“	Interaktiver Stimulus (Interaktion mit einem Mensch)
„iT“	Initiativer Stimulus (Interaktion mit einem Gegenstand)
„iE“	Erholsamer Stimulus

A.2.4 Emotionen (sie werden mit dem Präfix „E“ markiert)

„EF“	Freude
„ET“	Trauer
„EZ“	Zorn
„ES“	Schreck
„EU“	Unzufriedenheit
„ESt“	Staunen

A.2.5 Bewusste Aktionen (sie werden mit dem Präfix „B“ markiert)

„BO“	Suche nach einem Objekt (Objektsuche)
„BV“	Verfolgung eines Objektes (Objektverfolgung)
„BAE“	Erregung der Aufmerksamkeit beim Mensch (Aufmerksamkeitserregung)
„BE“	Erholung

A.2.5 Unbewusste Aktionen (sie werden mit dem Präfix „U“ markiert)

„UAB“ Augenblinzeln
„UAT“ Atmung

A.2.6 Erregung der internen Stimuli (sie werden mit dem Präfix „I“ markiert)

„IIK“ Für den interaktiven Stimulus
„IIT“ Für den initiativen Stimulus
„IE“ Für den erholsamen Stimulus

A.3 Entwickelte Datenbank des Wissens (mit Erklärungen)

// Kein Objekt im Sichtfeld (zAO). Die Suche nach einem wird gestartet:
nur_diese zAO -> BOS ET 0.06

// Ein Gegenstand ist im Sichtfeld anwesend (zAG).
// Wenn der initiative Stimulus dominiert ($iIK < iIT$), fokussiert der Roboter sich auf dem
// Gegenstand (BOV 2), freut er sich (EF) und es wird der initiative Stimulus erregt (IIT 1):
nur_diese zAG $iIK < iIT$ -> BOV 2 EF 0.2 IIT 1
// sonst (der interaktive Stimulus dominiert $iIK > iIT$) sucht der Roboter weiter nach einem
// Mensch (BOS) mit kleinem Trauer (ET 0.06):
nur_diese zAG $iIK > iIT$ -> BOS ET 0.06

// Ein Mensch ist im Sichtfeld anwesend (zAM).
// Wenn der interaktive Stimulus dominiert ($iIK > iIT$), fokussiert der Roboter sich auf dem
// Mensch (BOV 1), freut er sich (EF) und es wird der interaktive Stimulus erregt (IIK 1):
nur_diese zAM $iIK > iIT$ -> BOV 1 EF 0.2 IIK 1
// sonst (der initiative Stimulus dominiert $iIK < iIT$) sucht der Roboter weiter nach einem
// Gegenstand (BOS) mit kleinem Trauer (ET 0.06):
nur_diese zAM $iIK < iIT$ -> BOS ET 0.06

// Ein Mensch und ein Gegenstand sind im Sichtfeld anwesend (zAM zAG).
// Wenn der interaktive Stimulus dominiert ($iIK > iIT$), fokussiert der Roboter sich auf dem
// Mensch (BOV 1), keine Objektsuche (BOS -1), freut er sich (EF) und es wird der interaktive
// Stimulus erregt (IIK 1):
nur_diese zAM zAG $iIK > iIT$ -> BOV 1 BOS -1 EF 0.2 IIK 1
// sonst (der initiative Stimulus dominiert $iIK < iIT$), fokussiert der Roboter sich auf dem
// Gegenstand (BOV 2), keine Objektsuche (BOS -1), freut er sich (EF) und es wird der
// initiative Stimulus erregt (IIT 1):
nur_diese zAM zAG $iIK < iIT$ -> BOV 2 BOS -1 EF 0.2 IIT 1

// Ein Gegenstand wird im Sichtfeld erscheint (eEG).
// Wenn der interaktive Stimulus dominiert ($iIK > iIT$), fokussiert der Roboter sich auf dem
// Gegenstand (BOV 2), zeigt er seine Unzufriedenheit (EU 0.5) [weil der interaktive
// Stimulus einen Mensch sehen will], keine Freude (EF -0.6) [falls sie erregt wurde] und
// trotzdem wird der initiative Stimulus erregt (IIT 1) [auf diese Weise ist es möglich die
// Aufmerksamkeit des Roboters auf das Gegenstand umzufokussieren]:
nur_diese eEG $iIK > iIT$ -> BOV 2 EU 0.5 EF -0.6 IIT 1
// sonst (der initiative Stimulus dominiert $iIK < iIT$), fokussiert der Roboter sich auf dem
// Gegenstand (BOV 2), staunt er über das Gegenstand (Est 0.4) und es wird der initiative

// Stimulus erregt (IIT 1):

nur_diese eEG iIK<iIT -> BOV 2 ESt 0.4 IIT 1

// Ein Mensch wird im Sichtfeld erscheint (eEM).

// Wenn der initiative Stimulus dominiert (iIK<iIT), fokussiert der Roboter sich auf dem
// Mensch (BOV 1), zeigt er seine Unzufriedenheit (EU 0.5) [weil der initiative Stimulus ein
// Gegenstand sehen will], keine Freude (EF -0.5) [falls sie erregt wurde] und trotzdem wird
// der interaktive Stimulus erregt (IIK 1) [auf diese Weise ist es möglich die Aufmerksamkeit
// des Roboters auf den Mensch umzufokussieren]:

nur_diese eEM iIK<iIT -> BOV 1 EU 0.5 EF -0.5 IIK 1

// sonst (der interaktive Stimulus dominiert iIK>iIT), fokussiert der Roboter sich auf dem
// Mensch (BOV 1), versucht er die Aufmerksamkeit dieses Menschen zu Erregen (BAE),
// staunt er über den Mensch (ESt 0.4) und es wird der interaktive Stimulus erregt (IIK 1):

nur_diese eEM iIK>iIT -> BOV 1 BAE ESt 0.4 IIK 1

// Ein Gegenstand wird plötzlich aus dem Robotersichtfeld verschwindet (eVG).

// Bei dem dominierenden initiativen Stimulus (iIK<iIT) wird der Roboter böse (EZ 0.4) [weil
// ein Spielzeug „beim Kind“ weggenommen wurde] und es wird der initiative Stimulus erregt
// (IIT 1):

nur_diese eVG iIK<iIT -> EZ 0.4 IIT 1

// Bei dem dominierenden interaktiven Stimulus (iIK>iIT) zeigt der Roboter keine Reaktion
// auf das Verschwinden eines Gegenstandes:

// nur_diese eVG iIK>iIT ->

// Ein Mensch wird plötzlich aus dem Robotersichtfeld verschwindet (eVM).

// Bei dem dominierenden interaktiven Stimulus (iIK>iIT) wird der Roboter böse (EZ 0.4)
// [weil er weiter den Mensch sehen will] und es wird der interaktiven Stimulus erregt
// (IIK 1):

nur_diese eVM iIK>iIT -> EZ 0.4 IIK 1

// Bei dem dominierenden initiativen Stimulus (iIK<iIT) zeigt der Roboter keine Reaktion
// auf das Verschwinden eines Menschen:

// nur_diese eVM iIK<iIT ->

// Die Aufmerksamkeit des Roboters ist es Möglich auf das Gegenstand zu fokussieren. Es soll
// im Robotersichtfeld periodisch bewegt werden (ePB). Unabhängig vom in derzeit
// dominierenden Stimulus), fokussiert der Roboter sich auf dem Gegenstand (BOV 2), erregt
// die Aufmerksamkeit im Antwort (BAE), staunt er über das Gegenstand (ESt 0.4) und es wird
// der initiative Stimulus erregt (IIT 1):

nur_diese ePB -> BOV 2 BAE ESt 0.4 IIT 1

// Ein Objekt wird zu dem Roboter schnell annähert (eSA).

// Das Objekt ist ein Mensch (zAM) oder ein Gegenstand (zAG). Es soll beim Roboter eine
// negative Reaktion (Schreck ES) hervorrufen. Hierbei werden alle anderen Emotionen
// beruhigt (EF, ET, EZ, EU, ESt -0.5) :

nur_diese eSA zAG -> ES 0.5 EF -0.5 ET -0.5 EZ -0.5 EU -0.5 ESt -0.5

nur_diese eSA zAM -> ES 0.5 EF -0.5 ET -0.5 EZ -0.5 EU -0.5 ESt -0.5

// Ein Gegenstand (zAG) wird zu dem Roboter schnell annähert (eSA) und periodisch bewegt
// (ePB). Solche unangenehme Bewegung vor dem Robotergesicht hervorruft sein Zorn (EZ)
// und beruhigt andere Emotionen (EF, ET, EU, ESt -0.5, ES -1):

nur_diese eSA ePB zAG -> EZ 1 EF -0.5 ET -0.5 ES -1 EU -0.5 ESt -0.5

*// Wenn der erholsame Stimulus erregt wird, bedeutet es eine intensive oder lange Interaktion.
// Die internen Stimuli des Roboters sind hoch Erregt und deswegen kommt es die Erholung
// (BE). Hierbei wird die Wirkung der ersten Befehlzeile (nur_diese zAO -> BOS ET 0.06)
// ausgeschlossen (BOS -1 ET -0.06):*

nur_diese iE -> BE 1 BOS -1 ET -0.06

*// Wenn es kein Objekt und kein Ereignis im Sichtfeld des Roboters gibt, wird der emotionalen
Zustand des Roboters langsam nach einer Ruhe gestrebt (alle Emotionen -0.5):*

nur_diese -> EF -0.05 ET -0.05 EZ -0.05 ES -0.05 EU -0.05 ESt -0.05

Literaturverzeichnis

- 1 Terzopoulos D, Waters K. "Analysis and synthesis of facial image sequences using physical and anatomical models", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.15, No.6, p. 569–79. 1993
- 2 Masahiro Mori. Bukimi No Tani (The Uncanny Valley). Energy 7(4), 1970, S.33–35.
- 3 <http://www.csail.mit.edu/>
- 4 Scassellati, Brian "Theory of Mind for a Humanoid Robot", First IEEE/RSJ International Conference on Humanoid Robotics, September, 2000
- 5 Breazeal, C. "Sociable Machines: Expressive Social Exchange Between Humans and Robots". Sc.D. dissertation, Department of Electrical Engineering and Computer Science, MIT, 2000
- 6 Aryananda, Lijin. "A few days of a robot's life in the human's world: toward incremental individual recognition" Ph.D. Thesis, MIT CSAIL, 2007
- 7 Matthew Roberts Berlin. „Understanding the Embodied Teacher: Nonverbal Cues for Sociable Robot Learning" Ph.D. Thesis. MIT. Boston, 2008
- 8 Iida, F., Tabata, M., Hara, F. "Generating personality character in a face robot through interaction with human", Proceedings of the 7th IEEE International Workshop on Robot and Human Communication, ROMAN 1998, pp. 481-486
- 9 Marek Perkowski, Tsutomu Sasao, Atsumu Iseno, Uland Wong, Mikhail Pivtoraiko, Michele Folgheraiter, Martin Lukac, David Ng, Miranda Fix, Karl Kuchs "Use of Machine Learning based on Constructive Induction in Dialogs with Robotic Heads", Proceedings of the ICORR-2003, 22-25 April 2003
- 10 Mehra, P., Rendell, L. A., Wah, B. W., "Principled Constructive Induction", in Proceeding of the 11th International Joint Conference on Artificial Intelligence, pp. 651-656, 1989
- 11 Wenzel Svojanovsky. Darstellung von Emotionen als Repräsentation von Dialogerfolg. Diplomarbeit. Karlsruhe, 2006
- 12 Sebastian Lang, „Multimodale Aufmerksamkeitssteuerung für einen mobilen Roboter“, Ph.D. Thesis. Universität Bielefeld, 2005

- 13 Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, "The intelligent asimo: System overview and integration," in Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'02), 2002, pp. 2478–2483
- 14 <http://asimo.honda.com/asimo-specs>
- 15 www.harmonic-drive.com
- 16 N. Endo, K. Endo, M. Zecca, and A. Takanishi, "Modular Design of Emotion Expression Humanoid Robots KOBIAN", 18th CISM-IFTOMM Symposium on Robot Design, Dynamics, and Control ROMANSY 2010, Udine, Italy, 2010
- 17 Ogura Y., Shimomura K., Kondo H., Morishima A., Okubo T., Momoki S., Lim H., Takanishi A., "Development of a New Humanoid Robot WABIAN-2", in Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA), Orlando, Florida, May 2006
www.takanishi.mech.waseda.ac.jp/top/research/wabian/index.htm
www.takanishi.mech.waseda.ac.jp/top/research/kobian/index.htm
- 18 Standard FlexiForce Sensors for Force Measurement
www.tekscan.com/flexible-force-sensors
- 19 S. Kotosaka, T. Shibata, S. Schaal, "Humanoid Robot DB", in Proceedings of the International Conference on Machine Automation (ICMA), 2000, pp. 279 – 290
- 20 Christopher G. Atkeson, Joshua G. Hale, Frank Pollick, and Marcia Riley, Mitsuo Kawato (ATR Human Information Processing Research Laboratories); Shinya Kotosaka, Stefan Schaal, Tomohiro Shibata, Gaurav Tevatia, Ales Ude, and Sethu Vijayakumar, Erato Kawato (Dynamic Brain Project), "Using Humanoid Robots to Study Human Behavior", IEEE INTELLIGENT SYSTEMS, july/august 2000 IEEE, pp. 46 – 56
- 21 Brooks, Rodney, Cynthia Breazeal, Matthew Marjanovic, Brian Scassellati and Matthew Williamson, "The Cog Project: Building a Humanoid Robot", in C. Nehaniv, ed., Computation for Metaphors, Analogy and Agents, Vol. 1562 of Springer Lecture Notes in Artificial Intelligence, Springer-Verlag, 1998
- 22 Fitzpatrick, Paul, Giorgio Metta, Lorenzo Natale, Sajit Rao and Giulio Sandini, "What am I Doing? Initial Steps Towards Artificial Cognition", IEEE International Conference on Robotics and Automation (ICRA), Taipei, Taiwan, May 12 - 17, 2003
- 23 Breazeal, C. and Scassellati, B., "A context-dependent attention system for a social robot". In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI99). Stockholm, Sweden, 1146-1151, 1999
- 24 www.corba.org
- 25 www.ptgrey.com
- 26 R. A. Brooks and C. Rosenberg, "L - a common lisp for embedded systems", Association of Lisp Users Meeting and Workshop, 1995

- 27 Mike Blow, Kerstin Dautenhahn, Andrew Appleby, Chrystopher L. Nehaniv, David Lee, Adaptive Systems Research Group, "The Art of Designing Robot Faces – Dimensions for Human-Robot Interaction", Proceedings of the 1st conference on Human-robot interaction (HRI'06), Salt Lake City, Utah, USA, March 2-3, 2006
- 28 Giovanni Pioggia; Arti Ahluwalia; Federico Carpi; Andrea Marchetti; Marcello Ferro; Walter Rocchia; Danilo De Rossi, "FACE: facial automaton for conveying emotions", Applied Bionics and Biomechanics, Volume 1, pp. 91 – 100, Issue 2, June 2004
- 29 www.uni-magdeburg.de/ieat/robotslab/
- 30 A. Schneider, U. Schmucker, "Force Sensing for Multi-legged Walking Robots: Theory and Experiments – Part 1: Overview and Force Sensing", Mobile Robots, Moving Intelligence, pp. 447-470, ARS/pIV, Germany, December 2006
- 31 V. Rusin, „Adaptive Regelung von Robotersystemen in Kontaktaufgaben“, Ph.D. Thesis. Magdeburg, 2007
- 32 U. Schmucker, A. Schneider, V. Rusin. Interactive Virtual Simulator (IVS) of Six-Legged Robot "Katharina", Int. Conf. on Climbing and Walking Robots, CLAWAR'03, Italy, pp. 327-331, 2003.
- 33 F. Palis, V. Rusin, U. Schmucker, A. Schneider, Y. Zavgorodniy, „Adaptive multi-legged robot with articulated body“, Int. Symp. In Adaptive Motion in Animals and Machines, AMAM'05, Ilmenau, Germany, 2005
- 34 Palis F, Rusin V., Konyev M., and Zavgorodniy Y., "SLAIR 2 - Six Legged Autonomous Intelligent Robot", Thematic Issue ONPU 2006, Odessa, Ukraine.
- 35 Konyev M., Palis F, Rusin V., and Zavgorodniy Y., "Control System of Six Legged Autonomous Intelligent Robot", 7th Conference on Mobile Robots and Competitions, April 2007, Paderne, Portugal.
- 36 www.mathworks.com/products/simulink/
- 37 www.microsoft.com/windowsxp
- 38 Konyev M., Palis F., Zavgorodniy Y., Melnikov A., Rudskyy A., Telesh A., "Walking Robot "ANTON": Design, Simulation, Experiments", 11th Int. Conf. on Climbing and Walking Robots, CLAWAR'08, Sept.2008, Coimbra, Portugal.
- 39 www.mathworks.com/products/xpctarget/
- 40 EtherCAT – Ethernet for Control Automation Technology,
www.ethercat.org
- 41 www.hilscher.com, Hilscher GmbH, Hattersheim, Germany
- 42 www.xilinx.com
- 43 Mykhaylo Konyev, "Contact Processing within Walking Robots' Design", Ph.D. Thesis, Magdeburg, submitted in 2011

- 44 Palis F., Zavgorodniy Y., Konyev M., Telesh A., Melnykov A., Rudskyy A. Linear Drive For Anthropomorphous Robot "ROTTTO" (In russisch), Vestnik NTU "KPI", 2008
- 45 Konyev M., Melnykov A., Palis F., Rudskyy A. Linear Drive With Serial Elastic Actuator for Anthropomorphous Robot "ROTTTO" (In russisch), Vestnik NTU "KPI", 2009
- 46 A. Melnykov, M. Konyev, F.Palis, U. Schmucker, Robotslab Team. Linear elastic actuator of a biped robot "ROTTTO". 13th Int. Conf. on Climbing and Walking Robots, CLAWAR'10, August 2010, Nagoya, JP
- 47 M. Konyev, F.Palis, Y. Zavgorodniy, A. Melnykov, A. Rudskyy, A. Telesh, U. Schmucker. Low-level control system of a new biped robot "ROTTTO". 12th Int. Conf. on Climbing and Walking Robots, CLAWAR'09, Sept.2009, Istanbul, Turkey
- 48 Yuriy Zavgorodniy, „Konstruktion und Steuerung von Schreitrobotern mit ballistischem Laufverhalten“, Ph.D. Thesis, Magdeburg, 2009
- 49 A. Telesh, F.Palis, A. Rudskyy, A. Melnykov, T. Dynnik, M. Konyev, U. Schmucker, „Energy control for design of oscillations of byped in the frontal plane“, Accepted to: 14th Int. Conf. on Climbing and Walking Robots, CLAWAR'11, September 2011, Paris, France
- 50 A. Telesh, F.Palis, A. Rudskyy, A. Melnykov, T. Dynnik, M. Konyev, U. Schmucker, „Limit cycles walking in sagital plane based on the total energy control and virtual constraints“, Accepted to: 14th Int. Conf. on Climbing and Walking Robots, CLAWAR'11, September 2011, Paris, France
- 51 Andriy Melnykov, „Steuerung des zweibeinigen Schreitroboters mit elastisch gekoppelten Antrieben“, Magdeburg, Abgabe in 2011
- 52 www.maxonmotor.com
- 53 www.igus.com
- 54 Edsinger, Aaron and Una-May O'Reilly "Designing a Humanoid Robot Face to Fulfill Social Contracts", ROMAN-2000, Fall 2000
- 55 S. McCloud, "Understanding Comics: The Invisible Art", Harper Collins Publishers, Inc., 1993.
- 56 www.bbc.co.uk
- 57 www.starwars.com
- 58 Hideki Kozima, Cocoro Nakagawa, Hiroyuki Yano. Using Robots for the Study of Human Social Development. Developmental Robotics. A 2005 AAI Spring Symposium, March 21-23
- 59 www.nuvo.jp
- 60 www.asimo.honda.com

- 61 www.nec.co.jp/products/robot/
- 62 N. Endo, S. Momoki, M. Zecca, K. Itoh, A. Takanishi. Design and evaluation of the new head for the whole-body emotion expression humanoid robot KOBIAN. *Gerontechnology* 2008; 7(2):99.
- 63 “The man who mistook his girlfriend for a robot”, *Popular Science*, September, 2003, pp. 60-70
- 64 Minato T., Shimada M., Itakura S., Lee K., Ishiguro H, “Does gaze reveal the human likeness of an android?”, *The 4nd International Conference on Development and Learning*, pp. 106 – 111, 2005
- 65 Bernhard N. Tillmann, „Atlas der Anatomie des Menschen“, 2. überarbeitete Auflage, Springer Medizin Verlag Heidelberg 2004, 2010
- 66 Michael Schünke, „Funktionelle Anatomie- Topographie und Funktion des Bewegungssystems“, Georg Thieme Verlag, 2000
- 67 Joseph von Gerlach, „Handbuch der speciellen Anatomie des Menschen in topographischer Behandlung: mit besonderer Rücksicht auf die Bedürfnisse der ärztlichen Thätigkeit“, Oldenbourg, 1891
- 68 Jerry Pratt, Peter Dilworth, Gill Pratt, “Virtual Model Control of a Bipedal Walking Robot”, In *Proceedings of the 1997 International Conference on Robotics and Automation, ICRA '97*
- 69 Aaron Ladd Edsinger, „Robot Manipulation in Human Environments“, Ph.D. Thesis, Massachusetts Institute of Technology, January 2007
- 70 Alan M. Leslie, “Spatiotemporal continuity and the perception of causality in infants”, *Perception*, volume 13, pp. 287–305, 1984
- 71 Simon Baron-Cohen, „Mindblindness“, MIT Press, 1995
- 72 Newson, J., “The growth of shared understandings between infant and caregiver”, in M. Bullowa, ed., “Before Speech”, Cambridge University Press, pp. 207–222, 1979
- 73 Trevarthen C., “Communication and cooperation in early infancy: a description of primary intersubjectivity”, in M. Bullowa, ed., “Before Speech”, Cambridge University Press, pp. 321–348, 1979
- 74 Tronick E., Als H., Adamson, L., “Structure of early Face-to-Face Communicative Interactions”, in M. Bullowa, ed., “Before Speech”, Cambridge University Press, pp. 349–370, 1979
- 75 Brazelton T., “Evidence of Communication in Neonatal Behavior Assessment”, in M. Bullowa, ed., “Before Speech”, Cambridge University Press, pp. 79 – 88, 1979
- 76 James A. Russell, „What does facial expression mean?“, In James A. Russell and Jose Miguel Fernandez-Dols, editors, “The Psychology of Facial Expression”, chapter 1, Cambridge University Press, Cambridge, UK, 1997

- 77 Paul Ekman, "Are there basic emotions?", in "Psychological Review" №99 (3), pp. 550–553, 1992
- 78 Rainer Wollny, „Bewegungswissenschaft: ein Lehrbuch in 12 Lektionen“, 2007 by Meyer & Meyer Verlag, Aachen 2. Auflage 2010, p.141
- 79 VDI 2206, „Entwicklungsmethodik für mechatronische Systeme“, Berlin: Beuth Verlag, 2004
- 80 "The Insider's Guide to netX. Hilscher GmbH",
http://www.hilscher.com/files/netx_insiders_guide.pdf
- 81 „netX 10 Netzwerk Controller“,
http://de.hilscher.com/products_details_hardware.html?p_id=P_4ce145a5983e6
- 82 Dirk Abel, Alexander Bollig, "Rapid control prototyping“, 1. Auflage. Springer, Berlin 2006
- 83 Advanced Simulation and Control Engineering Tool (ASCET),
www.etas.com/ascet
- 84 Institut national de recherche en informatique et en automatique (INRIA), France,
www.inria.fr
- 85 Jean-Marie Zogg, „Arbeiten mit Scilab und Scicos. Scilab für numerische Berechnungen, Scicos für grafische Simulationen“, HTW-Verlag, 5.2007
- 86 Computer Aided Mechatronic Laboratory - Virtual Engineering Workbench (CAM-eL-View), www.ixtronics.com
- 87 <http://www.ni.com/labview/>
- 88 www.modelica.org/
- 89
 - www.3ds.com/products/catia/portfolio/dymola
 - www.mathcore.com/products/mathmodelica
 - www.simulationx.com
- 90 Digital Signal Processing and Control Engineering,
www.dspace.com
- 91 Scalable simulation software based on open-source,
www.scale-rt.com
- 92 Sergey Dzhantimirov, "Entwicklung einer Matlab/Simulink Werkzeugkette für die Unterstützung der Softwareentwicklung für eingebettete Systeme", Diplomarbeit, Magdeburg, Mai 2007
- 93 Ivica Crnković, Séverine Sentilles, Aneta Vulgarakis, Michel Chaudron, "A Classification Framework for Component Models", submitted for IEEE Transactions on Software Engineering, 2008

- 94 The new Standard in Automation,
www.iec61499.de
- 95 4DIAC – Open Source for Distributed Industrial Automation,
www.fordiac.org
- 96 Eclipse - an open development platform. Eclipse Foundation,
www.eclipse.org
- 97 3S – Smart Software Solutions
www.3s-software.com
- 98 The GNU Compiler Collection
<http://gcc.gnu.org>
- 99 LLVM – Low Level Virtual Machine. The LLVM Compiler Infrastructure,
<http://llvm.org>
- 100 Clang: a C language family frontend for LLVM,
<http://clang.llvm.org/>
- 101 The GNU Binutils – collection of binary tools,
www.gnu.org/software/binutils
- 102 Markus Hagen, “Implementierung eines ASIC-Cores und einer Software-Bibliothek zur Berechnung transzendenter Funktionen mittels CORDIC”, TU Berlin Fachgebiet Elektronik und medizinische Signalverarbeitung, Berlin, Februar 2010
- 103 GDB: The GNU Project Debugger
www.gnu.org/software/gdb
- 104 EtherCAT Technology Group
www.ethercat.org
- 105 RTAI - Real Time Application Interface
www.rtai.org
- 106 QNX realtime operating system
www.qnx.com/products/neutrino-rtos
- 107 RTX – Real-Time Extension for Windows
www.intervalzero.com
- 108 Communication Interface PCI Real-Time-Ethernet
http://hilscher.com/products_details_hardware.html?p_id=P_44fc100b6495a

Hilscher Gesellschaft für Systemautomation mbH, „Benutzerhandbuch. cifX-Karten Real-Time-Ethernet. Installation, Bedienung und Hardwarebeschreibung“
www.hilscher.com/files_manuals/CIFX-RE_usermanual_de.pdf
- 109 PRO INNO II Kooperationsprojekt, „iNET – Integrierte Technologieplattform für Communication & Control & Motion & Visualization“, Magdeburg, 2011

- 110 www.logitech.com
- 111 OpenCV – Open Source Computer Vision
<http://opencv.willowgarage.com/wiki>
- 112 Artem Rudskyy, „Entwicklung eines Stereokamerasystems zur Objekterkennung und Lokalisierung für einen mobilen Roboter“, Master’s Thesis, Magdeburg 2007, pp. 8–10 und 38–41
- 113 Artem Rudskyy, „Entwicklung eines Stereokamerasystems zur Objekterkennung und Lokalisierung für einen mobilen Roboter“, Master’s Thesis, Magdeburg 2007, pp. 18–20 und 46–48
- 114 ▪Paul Viola and Michael J. Jones, “Rapid Object Detection using a Boosted Cascade of Simple Features”, IEEE CVPR, 2001
▪Rainer Lienhart and Jochen Maydt, “An Extended Set of Haar-like Features for Rapid Object Detection”, IEEE ICIP 2002, Vol. 1, pp. 900-903, Sep. 2002.
- 115 Breazeal, C. "Sociable Machines: Expressive Social Exchange Between Humans and Robots". Sc.D. dissertation, Department of Electrical Engineering and Computer Science, MIT, 2000: Motion Saliency Feature Maps, p. 70
- 116 Jarbus A. L., “Rolle der Augenbewegung im Sehprozess“, Nauka 1965 (in russisch: Ярбус А. Л., “ Роль движений глаз в процессе зрения” , „Наука“ 1965)
- 117 A. Visioli, „Modified anti-windup scheme for PID controllers”, IEE Control Theory and Applications, 150(1):49—54, January 2003
- 118 Paul Ekman, “Gefühle lesen: Wie Sie Emotionen erkennen und richtig interpretieren”, Spektrum Akademischer Verlag, März 2007