

# Entwicklung einer ALPHA

## Architektur einer lokationssensitiven Puppe für die hydropneumatische Animation

### Diplomarbeit

eingereicht bei

Prof. Dr. Krömker  
Professur Graphische Datenverarbeitung  
Betreuer: Tobias Breiner

Fachbereich Informatik und Mathematik  
Johann Wolfgang Goethe-Universität  
Frankfurt am Main

Toni Perkovic

01.März 2006

## **Eidesstattliche Erklärung**

Hiermit erkläre ich an Eides statt, dass die vorliegende Diplomarbeit ohne unzulässige Hilfe und nur unter Verwendung der angegebenen Literatur angefertigt wurde.

Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Frankfurt am Main, den 01.März 2006

---

Toni Perkovic

# Inhaltsverzeichnis

<b>1 EINLEITUNG .....</b>	<b>1</b>
<b>2 TECHNISCHE GRUNDLAGEN UND STATE-OF-THE-ART -ANALYSE... 7</b>	
2.1 OPTISCHE SENSOREN .....	7
2.1.1 <i>Optischer Resolver</i> .....	7
2.1.2 <i>Optoencoder</i> .....	8
2.2 SPANNUNGSTEILER UND ANALOG/DIGITAL-WANDLER .....	10
2.2.1 <i>Spannungsteiler</i> .....	10
2.2.1 <i>Analog/Digital Wandler</i> .....	14
2.2.2 <i>Verschiedene Architekturen der Analog/Digital-Wandler</i> .....	17
2.2.2.1 <i>Parallelverfahren (Flash-Verfahren)</i> .....	17
2.2.2.2 <i>Wägeverfahren / iterative Verfahren (Sukzessive Approximation)</i> .....	19
2.2.2.3 <i>Zählverfahren</i> .....	22
2.2.3 <i>Signalberechnung mittels Potentiometer und Analog/Digital-Wandler</i> .....	23
2.3 STATE-OF-THE-ART .....	24
2.3.1 <i>Stop-Motion-Animation, Rotoscoping und Animatronics</i> .....	24
2.3.2 <i>Motion-Capturing</i> .....	25
2.3.2.1 <i>Grundlagen</i> .....	25
2.3.2.2 <i>Optisches Motion-Capturing</i> .....	26
2.3.2.3 <i>Prothetisches (mechanisches) Motion-Capturing</i> .....	27
2.3.2.4 <i>Magnetisches Motion-Capturing</i> .....	27
2.3.2.5 <i>Akustisches Motion-Capturing</i> .....	27
2.3.2.6 <i>Ein bestehendes System: Das „Motion Analysis's Eagle“</i> .....	27
2.3.3 <i>Dinosaur Input Device</i> .....	29
2.3.4 <i>Datenhandschuhe</i> .....	32
2.3.4.1 <i>Grundlagen</i> .....	32
2.3.4.2 <i>DataGlove und Dextrous Handmaster</i> .....	33
2.3.5 <i>Datenanzüge</i> .....	35
2.3.6 <i>Nachteile vorgestellter Systeme</i> .....	35
<b>3. ANFORDERUNGSANALYSE .....</b>	<b>39</b>
<b>4. EIGENES KONZEPT – DIE LOKATIONSENSITIVE PUPPE .....</b>	<b>41</b>
4.1 DIE GRUNDIDEE .....	41
4.2 DER PUPPEN-AUFBAU .....	42
4.2.1 <i>Grundlagen</i> .....	42
4.2.2 <i>Das Skelett</i> .....	43
4.2.3 <i>Die Gelenke</i> .....	45
4.2.4 <i>Die Signalübermittlung</i> .....	48
4.2.4.1 <i>Potentiometer</i> .....	48
4.2.4.3 <i>Analog/Digital-Wandler</i> .....	52
4.2.4.4 <i>Stromaufnahme und Spannungsversorgung</i> .....	54
4.2.4.5 <i>Parallelport</i> .....	56
4.2.4.6 <i>Die Kabel und die Platine</i> .....	57

4.3 DER TREIBER .....	60
4.3.1 Grundlagen .....	60
4.3.2 Schnittstellenspezifikation.....	65
4.4 DIE ANZEIGE .....	66
<b>5. EVALUATION DES KONZEPTS .....</b>	<b>67</b>
<b>6. AUSBLICK .....</b>	<b>69</b>
<b>7. ZUSAMMENFASSUNG.....</b>	<b>71</b>
<b>8. ANHANG .....</b>	<b>73</b>
A ABBILDUNGSVERZEICHNIS .....	73
B TABELLENVERZEICHNIS .....	75
C LITERATURVERZEICHNIS.....	77

# 1 Einleitung

Seit Beginn der Filmindustrie Anfang des 20. Jahrhunderts sind Menschen von bewegten Bildern fasziniert. Zunächst beschränkte sich das auf Stummfilme, da die Aufzeichnung des Tons noch nicht möglich war. Aber auch das Bild war nicht perfekt. Die Akteure schienen sich zu rasch und oft in derselben Szene ungleichmäßig schnell zu bewegen. Der Grund hierfür waren die Aufnahme- und Abspielgeschwindigkeit, welche nicht aufeinander abgestimmt waren. Das ist auch nicht verwunderlich, denn bei den ersten Kameras wurde die Filmrolle während der Aufzeichnung noch von Hand gedreht. Dadurch war es unmöglich eine stetige Geschwindigkeit zu halten.

Abgesehen davon, dass der Ton nicht aufgenommen wurde, waren die Filme nicht in Farbe, sondern in schwarz-weiß, was den optischen Effekt minderte. Dennoch war das Publikum begeistert. Denn Filme bedeuteten Unterhaltung auf eine neue, moderne Art und Weise und dies fesselte ungemein. Ein wichtiger Bestandteil, wenn auch damals sehr aufwendig, waren Spezialeffekte, mit denen sich imaginäre Szenen verwirklichen ließen. So auch beim 1933 gedrehten Film „King-Kong“. Selbst wenn die Effekte einem heutzutage weder realistisch noch authentisch vorkommen, so war der Film damals ein Meilenstein in der Film- und Spezialeffektentwicklung. Puppen wurden benutzt, welche im Stop-Motion-Verfahren (siehe Abschnitt 2.3.1) aufgenommen wurden. Um realistische Aufnahmen von einem, an Hochhäusern in New York kletternden, Riesenaffen zu erzielen, wurden Fotomontageszenen eingefügt. Für einige Szenen wurden sogar aufwendige Animatronics-Figuren (siehe Abschnitt 2.3.1) gebaut.



Abbildung 1.1: Szene aus „King Kong und die weiße Frau“ von 1933

Um den erwünschten und realistischen Effekt einer perfekten visuellen Illusion zu erzielen, schritt die Entwicklung wagemutiger und ausgefallener und dadurch durchaus erfolgreicher Animationstechniken rasant voran. Denn für die perfekte Illusion reichten bald nicht mehr einfache Stop-Motion-Verfahren oder die simple Überlagerung der Filmmaterialien für die Fotomontage. Als ein gutes Beispiel dient hier das Rotoscoping (siehe Abschnitt 2.3.1). Es wurde 2002 in einer abgewandelten Form auch beim zweiten Teil der Trilogie „Herr der Ringe“ benutzt. Doch seine Anfänge hat das Rotoscoping viel früher gehabt.

Bereits 1914 wurde das Rotoscoping in einer Zeichentrickfilmserie namens „Out Of The Inkwell“ angewendet. Selbst beim 1977, also über 60 Jahre später, gedrehten Film „Star Wars“ von George Lucas hat sich am grundsätzlichen Verfahren des Rotoscopings nicht geändert. In diesem Film wurden Lichtschwerter den Kampfszenen nachträglich hinzugefügt. Das Ergebnis war beeindruckend. Der Film war ein Publikumsmagnet und ein Meisterwerk der damaligen Spezialeffekte. [ILM96][ANI02]



Abbildung 1.2: Szene aus ‚Star Wars‘

Doch der größte Durchbruch der Animation kam mit voranschreitendem Computerzeitalter, Mitte der 80er Jahre. Zum ersten Mal wurde Motion-Capturing (siehe Abschnitt) verwirklicht. Die Anzahl der Animationstechniken und Methoden wuchs rasant. Digitale Handschuhe, so genannte Handmaster sind realisiert worden, welche Handbewegungen messen, und sie mit Hilfe des Computers anzeigen konnten. Weiterhin wurden mechanische Eingabepuppen mit Gelenksensoren entwickelt, um längst ausgestorbene Tiere darstellen zu können. Das Motion-Capturing gab es in sehr vielen Varianten der technischen Implementierung und Funktion. Und das Alles, um das Bedürfnis des Menschen zu befriedigen, optisch perfekt unterhalten worden zu sein.

Eine der wichtigsten Errungenschaften des Computerzeitalters waren 3D-Software-Entwicklungstools. Das waren Programme mit denen sich jede Bewegung, jede Form und jede Darstellung auf den Bildschirm zaubern ließ. Diese Werkzeuge mussten lange auf ihre Realisierung warten, weil sie unheimlich viel Rechenkraft brauchen.

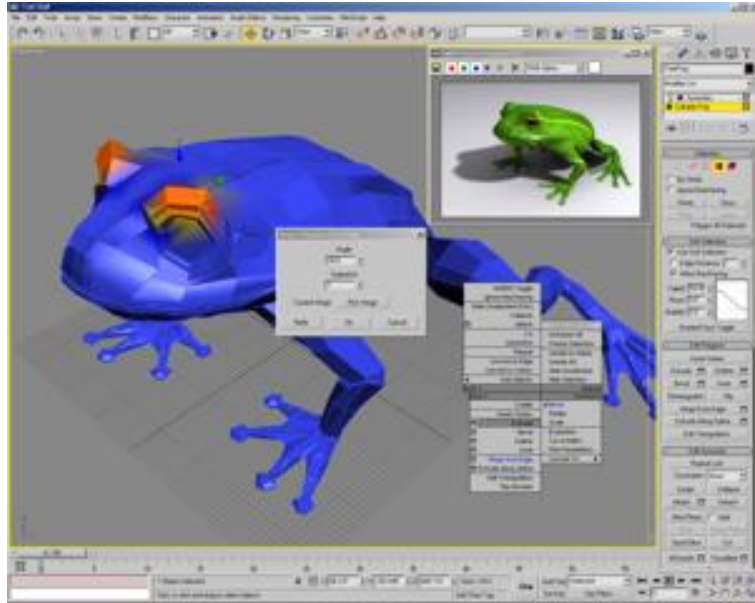


Abbildung 1.3: Darstellung eines Froschs mit 3D Studio Max

Nebenher wurde auch an der Entwicklung mechanischer Eingabegeräte weitergearbeitet. Im Zeitalter der mächtigen 3D-Software-Werkzeuge, wie 3D Studio Max oder Maya, erscheint es vielleicht nicht gerade fortschrittlich Geräte oder Puppen zu entwickeln, die die zu animierenden Körper darstellen sollen. Doch solche Eingabegeräte haben vielerlei Vorteile und sollen die Software unterstützen und nicht ersetzen.

Sie sind nämlich sehr leicht zu handhaben und sind perfekt dazu geeignet von Stop-Motion Animatoren benutzt zu werden. Dadurch kann die Animation, die mit der Puppe gemacht wurde, mit einem, durch die mit 3D-Software erstelltem Modell verknüpft werden.

Die Fortschritte in der Computertechnologie sind soweit, dass heutige Rechner die gewaltigen Eingabemengen einer aufgenommenen Animationsbewegung schnell bewältigen können. Sogar so schnell, dass sie in Echtzeit Computeranimationen darstellen können. Das ist vor allem wichtig, da Animatoren das Ergebnis möglichst schnell auswerten wollen. Auch die Renderer sind weiterentwickelt worden, denn echtzeitfähige Ray-Tracing-Renderer sind bereits entwickelt [RTD05].



Diese Arbeit beschäftigt sich mit verschiedenen Bewegungsaufzeichnungsmethoden für Animationen und wird die Möglichkeit der Implementierung einer präzisen, universellen und kostengünstigen Eingabepuppe zeigen.



## 2 Technische Grundlagen und State-of-the-Art - Analyse

### 2.1 Optische Sensoren

#### 2.1.1 Optischer Resolver

Optische Resolver gehören zu den Absolutwertgebern und sind Messgeräte für eine Weg- oder Winkelmessung. Das Herzstück eines einfachen optischen Resolvers ist eine runde Glasscheibe, die in mehrere konzentrische Spuren unterteilt ist. In jeder Spur sind Hell/Dunkel-Muster angebracht. Die Glasscheibe wird von einer Lichtquelle beleuchtet und hinter jeder Spur befindet sich ein Lichtsensor. Das gleichzeitige Auslesen der Spuren ergibt einen Binärcode, welcher dann einem bestimmten Winkel (oder Weg) zugeordnet werden kann. Der optische Resolver besteht aus einem sich drehenden Teil, an welchem die Glasscheibe befestigt ist und einem feststehendem Teil, an welchem die Sensoren und die Lichtquelle angebracht sind. Dreht man nun die Glasscheibe, so dass ein anderes Hell-Dunkel Muster die Spursensoren verdeckt, so ergibt sich ein anderer Binärcode.

Je mehr Spuren man benutzt, desto genauer ist natürlich die Ausgabe. Benutzt man vier Spuren, so kann man  $2^4 = 16$  Zustände kodieren. Bei 16 Spuren wären es schon  $2^{16} = 65536$  Zustände.

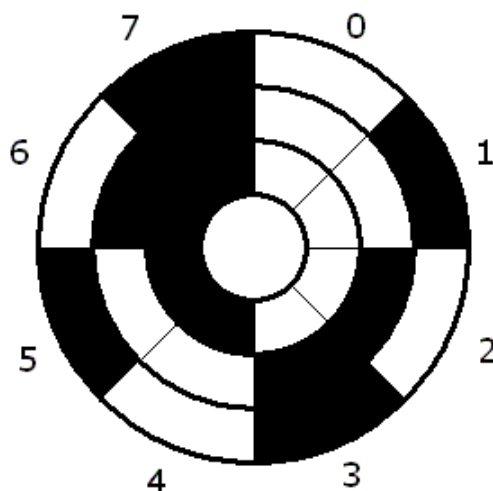


Abbildung 2.1: Glasscheibe eines optischen Resolvers mit 3 Spuren

Anstatt des Binärcodes kann man auch den Graycode benutzen. Beim Graycode ändert sich bei benachbarten Werten immer nur ein Bit. Damit sollen Fehlinterpretationen durch Zwischenzustände vermieden werden. [MEC03]

Dezimalzahl	Binärcode	Graycode
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

Tabelle 2.1: Beispiel des Graycodes

### 2.1.2 Optoencoder

Optoencoder gehören zu den inkrementellen Messverfahren. Sie haben nicht wie ein Spannungsteiler oder ein Resolver einen Absolutwert als Ausgabe, sondern geben beispielsweise bei Winkelmessungen ein Signal aus, wann immer sich der Winkel um einen bestimmten Wert erhöht oder erniedrigt hat.

Die wesentlichen Bauteile sind ein Lichtsignalgeber (meistens eine LED), ein Abtastgitter und zwei Photoelemente (Sensoren). Zusätzlich braucht man noch einen digitalen Zähler. Das Abtastgitter ist eine runde Drehscheibe und besitzt zwei Spuren, Spur A und Spur B. Diese Spuren besitzen in periodischen Abständen Löcher, die das Licht der LED auf die Photoelemente durchlassen. Die Öffnungen der Spur A und B sind zueinander um  $90^\circ$  phasenverschoben. Das macht es einem möglich zu erfahren, in welche Richtung man das Abtastgitter gedreht hat. Man verändert den Wert des Zählers ausschließlich bei einer aufsteigenden Flanke der Spur A. Ist zu diesem Zeitpunkt der Photosensor der Spur B auf High (Licht wird empfangen), so

zählt man hoch, ist er auf Low (kein Licht wird empfangen), dann zählt man herunter.

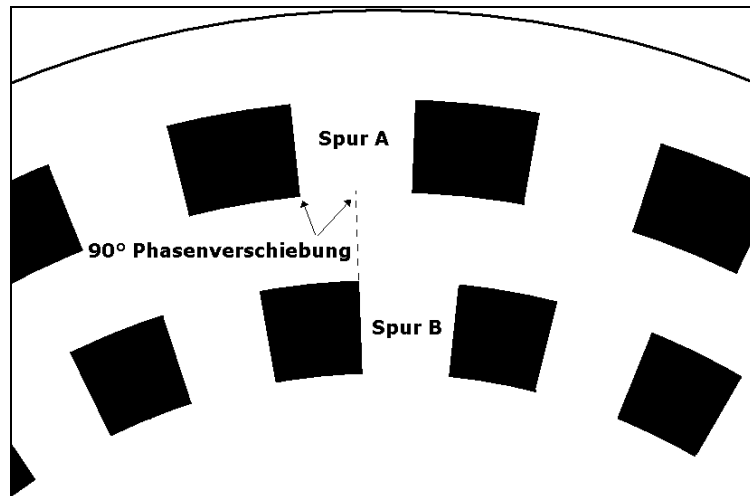


Abbildung 2.2: Ausschnitt der Drehscheibe eines Optocoders

Nimmt man als Vorwärts-Richtung eine Rechtsdrehung an, so ergibt sich folgendes Ausgangssignalmuster an den Photosensoren:

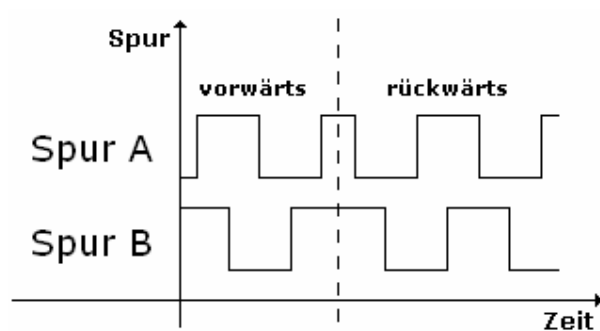


Abbildung 2.3: Signalveränderung beim Drehen der Glasscheibe

Durch das Auslesen des Zählers kann man somit immer bestimmen, um wie viel Grad man seit Beginn der Messung vorangeschritten ist. Den Zähler sollte man aus diesem Grund mit einem äußeren Schalter auf Null bringen können, um eine Mes-

sung starten zu können. Der Schalter kann entfallen, falls der Encoder eine zusätzliche Spur, eine sogenannte „Nullspur“ besitzt, die genau einmal pro Umdrehung ein „Loch“ hat. Wird ein Impuls der Nullspur registriert, wird der Zähler auf Null gesetzt. [MEC03]

## **2.2 Spannungsteiler und Analog/Digital-Wandler**

### **2.2.1 Spannungsteiler**

Ein Spannungsteiler ist eine elektrische Schaltung zur Teilung von Spannungen. Er besteht in der einfachsten Form aus zwei in Reihe geschalteten Widerständen  $R_1$  und  $R_2$ . Legt man nun eine Spannung  $U_1$  an diese Schaltung an, so kann man zwischen den beiden Widerständen eine abgesenkte Spannung  $U_2$  messen. Die Formel zur Berechnung dieser Spannung lässt sich wie folgt ableiten: Der Strom  $I$ , der beim Anlegen von  $U_1$  durch die Schaltung fließt, lässt sich sowohl durch  $U_1$  und den beiden Widerständen  $R_1$  und  $R_2$  beschreiben, als auch durch  $U_2$  und dem Widerstand  $R_2$  beschreiben. Damit kann man folgende Gleichungen aufstellen:

$$I = \frac{U_1}{R_1 + R_2} \quad (2.1)$$

$$I = \frac{U_2}{R_2} \quad (2.2)$$

Daraus folgert man:

$$\frac{U_2}{R_2} = \frac{U_1}{R_1 + R_2} \quad (2.3)$$

und schließlich:

$$U_2 = U_1 \cdot \frac{R_2}{R_1 + R_2} \quad (2.4)$$

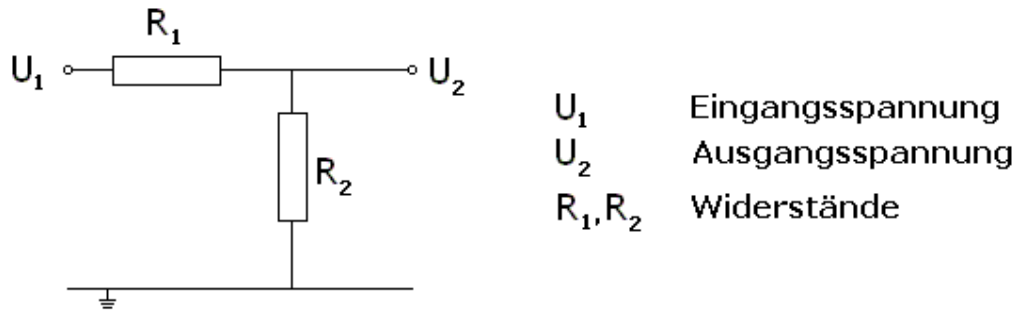


Abbildung 2.4: Schaltbild eines einfachen Spannungsteilers

Allgemein, für beliebig viele Widerstände, kann man folgendes Schaltbild aufstellen.

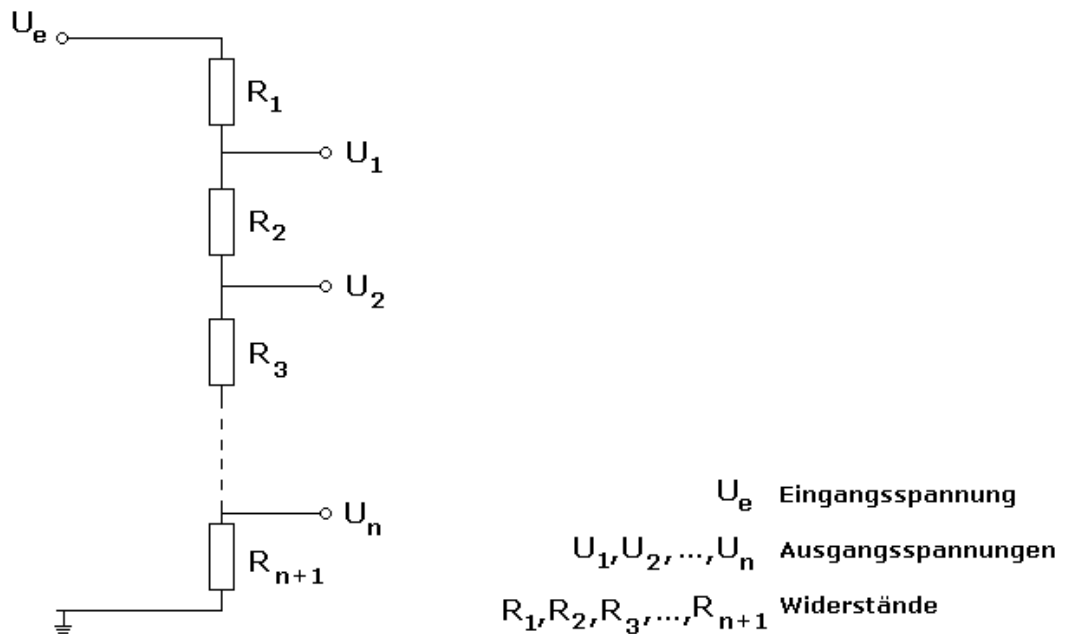


Abbildung 2.5: Schaltbild eines Spannungsteilers mit n Stufen

Analog zur obigen Berechnung, ergeben sich dann die Gleichungen:

$$U_1 = U_e \cdot \frac{R_2 + R_3 + \dots + R_n}{R_1 + R_2 + R_3 + \dots R_n} \quad (2.5)$$

$$U_2 = U_e \cdot \frac{R_3 + R_4 + \dots + R_n}{R_1 + R_2 + R_3 + \dots R_n} \quad (2.6)$$

$$U_n = U_e \cdot \frac{R_{n+1}}{R_1 + R_2 + R_3 + \dots R_n} \quad (2.7)$$

Einen Widerstand dieser Schaltung, kann man demnach mit der folgenden Gleichung herausfinden:

$$U_i = U_e \cdot \frac{R_{i+1} + R_{i+2} + \dots + R_n}{R_1 + R_2 + R_3 + \dots R_n} \quad (2.8)$$

[GEL05]

Nun kann man statt zwei Widerständen auch ein Potentiometer benutzen. Ein Potentiometer ist ein variabler Widerstand. Es besitzt drei Anschlüsse. Zwischen zwei Anschlüssen ist ein Widerstand geschaltet, an welchem mittels des dritten Anschlusses ein Abgriff erfolgt. Der Abgriffspunkt ist von außen einstellbar, so dass der Widerstand zwischen einem Anschluss und dem Abgriffsanschluss variabel ist.

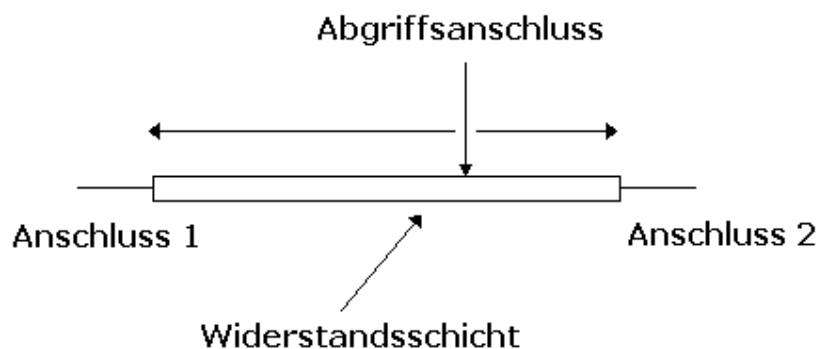


Abbildung 2.6: Potentiometeranschlüsse



Legt man nun wie im obigen Beispiel eine Spannung  $U_1$  an die äußeren Anschlüsse an (z.B.  $U_1$  an Anschluss 1, Masse an Anschluss 2), so kann man am Abgriff eine einstellbare Spannung  $U_2$  messen. Sei  $R_p$  der gesamte Widerstand zwischen den beiden äußeren Anschlüssen, und  $R_{ab}$  der Widerstand zwischen dem Abgriff und dem Anschluss 2. Dann gilt für die Spannung  $U_2$ :

$$U_2 = U_1 \cdot \frac{R_{ab}}{R_p} \quad (2.9)$$

Wie man sieht, lässt sich hiermit für  $U_2$  der gesamte Bereich zwischen 0 und  $U_1$  einstellen. Bewegt man nämlich den Abgriff so weit in Richtung Anschluss 2, dass der Widerstand  $R_{ab} = 0$  ist, so wird auch  $U_2 = 0$  sein. Stellt man andererseits  $R_{ab} = R_p$  ein, so wird  $U_2 = U_1$  sein.

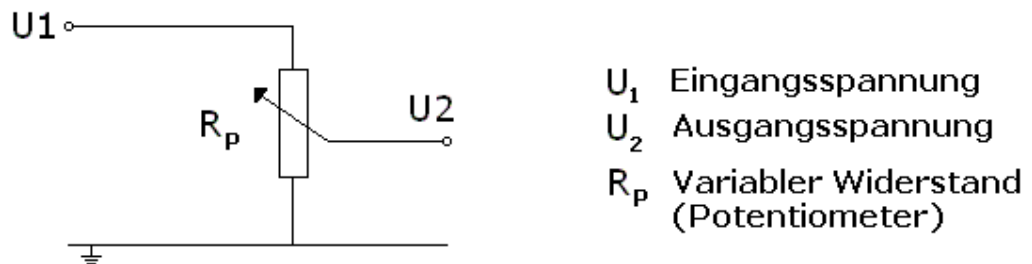


Abbildung 2.7: Schaltbild eines Potentiometers

Potentiometer gibt es in verschiedenen Bauformen. Die wichtigsten sind der Dreh- und der Schiebepotentiometer. Beim Drehpotentiometer dreht man einen Stift, welcher mit dem Abgriff verbunden ist. Der Drehwinkel ist meist zwischen  $100^\circ$  und  $300^\circ$ . Das Schiebepotentiometer besitzt auch einen Stift oder einen kleinen Hebel, welcher aber nicht gedreht wird, sondern links und rechts bewegt werden kann. Auch er ist mit dem Abgriff verbunden und steuert somit den Widerstand.

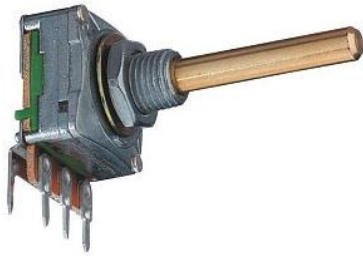


Abbildung 2.8: Drehpotentiometer



Abbildung 2.9: Schiebepotentiometer

Des Weiteren wird bei Potentiometern unterschieden, ob die Widerstandsänderung linear oder logarithmisch ist. Logarithmische Potentiometer kommen beispielsweise in Lautstärkereglern von Audioverstärkern zum Einsatz, weil dies aufgrund der Beschaffenheit des menschlichen Gehörs bessere Effekte erzielt.

### 2.2.1 Analog/Digital Wandler

Analog/Digital-Wandler, kurz A/D-Wandler, oder Analog-Digital-Converter (ADC) sind signalverarbeitende elektronische Schaltungen. Sie rechnen ein angelegtes analoges Signal, z.B. Spannung oder Strom in ein digitales Signal, eine binäre Zahl um. Dieses digitale Signal kann dann mit anderen digitalen Schaltungen weiterverarbeitet werden (z.B. für die Anzeige an einem Voltmeter-LCD). A/D Wandler gibt es in verschiedenen Wandlungsgeschwindigkeiten (von weniger als 1000 Wandlungen in der Sekunde bis mehr als 200 Millionen Wandlungen in der Sekunde). Die Geschwindigkeit hängt davon ab, welche Architektur der Chip benutzt, mit welcher maximalen Taktfrequenz er betrieben werden kann und ob die Ausgabe seriell oder parallel erfolgt. Die Wandlungsgeschwindigkeit ist meistens konstant, kann aber bei manchen Architekturtypen vom Wert des angelegten Signals abhängen.

Weiterhin gibt es die Unterteilung in Auflösungen. Die Umwandlung eines analogen Wertes, wie z.B. der Spannung, in einen diskreten digitalen Wert hat automatisch Rundungsfehler zur Folge. Die Auflösung des A/D Wandler beschreibt, wie viele Bits er für die Ausgabe erzeugt. Es gibt Wandler mit 6 bis hin zu 24 Ausgabebits. Ein 8-bit A/D Wandler hat als Ausgabe eine binäre Zahl zwischen 0 und 255 ( $2^8=256$  Ausgabewerte). Die Quantisierung erfolgt hier in Spannungsschritten. Die Anzahl

der Schritte bei dem 8-bit Wandler ist 255 (von 0 bis 255). Wenn nun eine Spannung zwischen 0 und 5 Volt gewandelt werden soll, dann sind die Spannungsschritte für den nächst höheren Wert des Wandlers ungefähr 0,0196V groß. Denn,

$$\text{Spannungsschrittgröße} = 5\text{V} : 255 \approx 0,0196\text{V} .$$

Das heißt, dass jede Spannung zwischen 0 und 0,0196 Volt als 0 ausgegeben wird. Für einen 24-bit Wandler gälte:

$$\text{Spannungsschrittgröße} = 5\text{V} : (2^{24} - 1) \approx 2,98 \cdot 10^{-7} .$$

Der 24-bit Wandler ist  $2^{16} = 65536$  Mal genauer als ein 8-bit Wandler. Je mehr Bits demnach ein A/D Wandler als Ergebnis bereitstellt, desto genauer ist seine Berechnung.

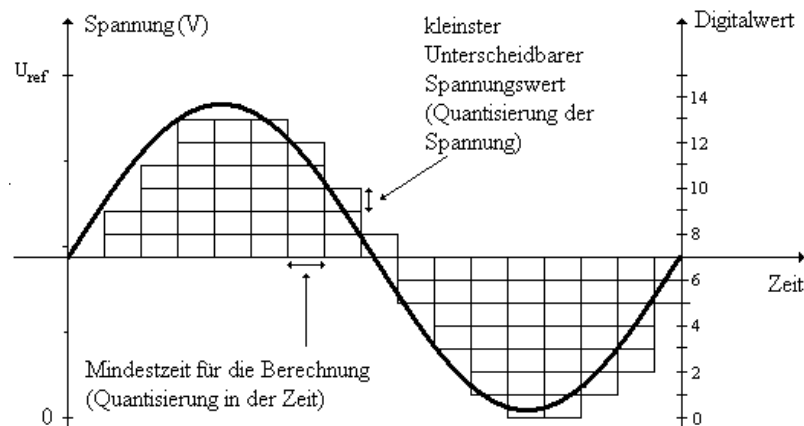


Abbildung 2.10: Darstellung der Quantisierung eines A/D Wandlers

Ein Analog/Digital Wandler quantisiert somit das analoge Signal sowohl in der Zeit, denn es kann nur ein Wert pro Wandlung berechnet werden, als auch in der Amplitude, weil die Ausgabe diskret erfolgt. Die Ausgabe ist demgemäß immer stufenförmig. Die Quantisierung in der Amplitude nennt man bei einem Spannungs-A/D-Wandler  $U_{\text{LSB}}$ , das heißt, dass dies die Spannung ist, die einen Wechsel des nieder-

wertigsten Bits (LSB, Lowest Significant Bit) hervorruft. Dann ist das Digitalwort D, das Ausgegeben wird:

$$D = U_E / U_{\text{LSB}} = U_E / U_{\text{Ref}} \cdot 2^n \quad (2.10)$$

Der Wandlungsvorgang des A/D-Wandlers besteht somit aus drei Schritten:

1. Zeitabtastung des Signals (Zeitquantisierung)
2. Quantisierung (Diskretisierung) der Analogsignale zu den Abtastzeitpunkten
3. Kodierung des ermittelten Amplitudenwertes.

Die resultierenden Codes sind meist Binärcodes, bei manchen A/D-Wandlern kommen auch komplementärer Binärcode, BCD-Code, Offset-Binär-Code, Zweier-Komplement-Code oder auch der Gray-Code zum Einsatz.

Eine weitere wichtige Kenngröße ist das Quantisierungsrauschen. Es entsteht durch den Quantisierungsfehler. Dazu kommt es durch eine sägezahnförmige „Fehlspannung“ beim Quantisieren. Man definiert dann ein Signal/Rausch-Verhältnis

$$S/N[\text{dB}] = 6,02 \cdot n + 10,8 \quad (2.11)$$

welches mit der Auflösung steigt.

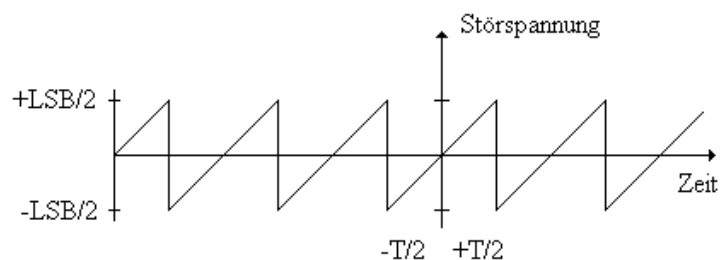


Abbildung 2.11: Übertragungskennlinie des A/D-Wandlers  
(Stufenbreite 1LSB für ideale Betriebsweise)

Wichtige Kenngrößen des A/D-Wandlers zusammengefasst: [ELE95]

1. die Auflösung in Bits
2. die Umsetzgeschwindigkeit, Anzahl Umsetzungen pro Zeiteinheit
3. die Umsetzzeit, Zeit für eine Umsetzung
4. das Quantisierungsrauschen

## 2.2.2 Verschiedene Architekturen der Analog/Digital-Wandler

Drei verschiedene Unterteilungen können bei den Architekturen von Analog/Digital-Wandlern unterschieden werden: [ELE95]

- Parallelverfahren (Flash-Verfahren)
- Wägeverfahren, iterative Verfahren (Sukzessive Approximation)
- Zählverfahren

### 2.2.2.1 Parallelverfahren (Flash-Verfahren)

Das Parallelverfahren berechnet alle Bits auf einmal. Dazu benötigt ein n-bit A/D-Wandler genau  $2^n - 1$  Komparatoren. Ein Komparator ist ein analoges elektronisches Bauteil, welches durch einen Operationsverstärker ohne Gegenkopplung realisiert wird. Zwei Eingänge (+ und -) und ein Ausgang stehen also zur Verfügung. Am Ausgang wird genau dann eine eins erzeugt, wenn die Spannung am „Plus“-Eingang mindestens so groß ist wie die Spannung am „Minus“-Eingang. Ansonsten liegt am Ausgang Null an. Dies kommt dadurch zustande, dass der Operationsverstärker mit sehr hoher Verstärkung betrieben wird. Somit nähert er sich bei positiver bzw. negativer Eingangsspannungsdifferenz der positiven bzw. negativen Spannungsversorgung an.

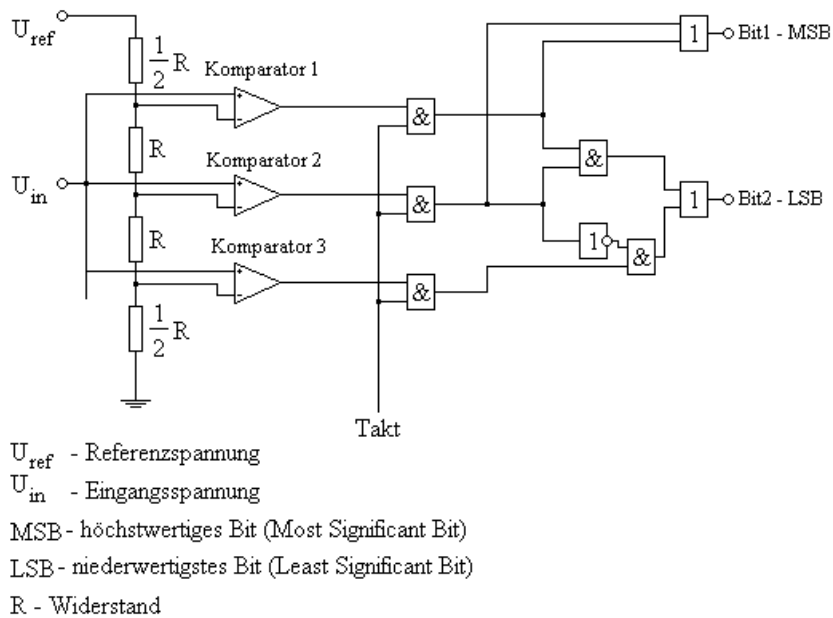


Abbildung 2.12: 2-bit A/D-Wandler mit Parallelverfahren (Blockschaltbild)

Die Komparatoren werden zueinander parallel an einen Spannungsteiler mit  $2^n$  Stufen geschaltet (so genanntes Flash-Prinzip). Der Spannungsteiler teilt die Referenzspannung in eine Stufe mehr ein, als es Komparatoren gibt. Jeder Komparator vergleicht eine anliegende Spannung im Spannungsteiler (eine Spannungsnormale) mit der Eingangsspannung. Der erste und der letzte Spannungsschritt sind nur halb so groß wie die anderen, um damit die Spannung auf-, bzw. abzurunden. Alles was unter der Spannung des letzten Gliedes im Spannungsteiler ist, wird auf 0 abgerundet und alles was über der Spannung des ersten Gliedes liegt, wird auf  $U_{ref}$  aufgerundet.

Je mehr sich die Eingangsspannung der Referenzspannung nähert, desto mehr Komparatoren schalten ihren Ausgang auf Eins. Zählt man jetzt mit einer simplen Additionslogik alle Ausgänge der Komparatoren zusammen, so hat man einen sehr genauen digitalen Wert der Eingangsspannung vorliegen.

Das Parallelverfahren erfordert  $2^n - 1$  Spannungsnormale für n Bits. Bei einem 12-bit Wandler wären das 4096 Spannungsnormale. In der Industrie ist das auch die obere

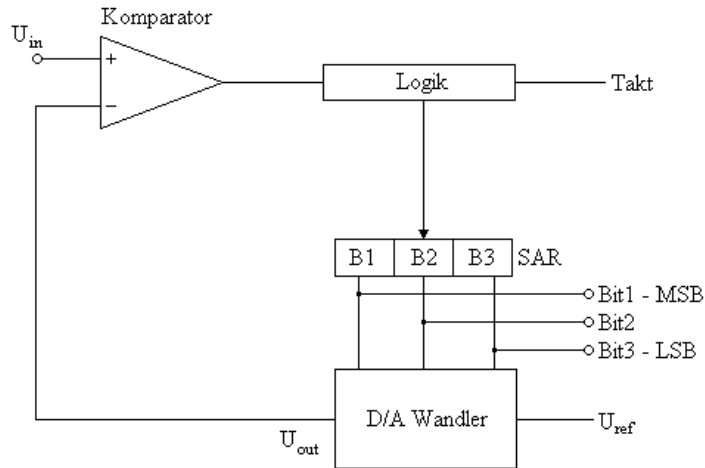
Auflösungsgrenze für Parallelwandler, weil jenseits davon die Chips zu teuer werden.

Der Aufwand sinkt, wenn man aus dem reinen Parallel-Wandler einen 2-Stufigen Parallel-Serien A/D-Wandler (Kaskaden-Wandler) anfertigt. Ein  $n$ -bit Kaskaden Wandler besteht aus  $2 \cdot n/2$ -bit Parallelwandlern. Dies funktioniert bei einem 8-Bit Wandler folgendermaßen: Zuerst berechnet der Erste die oberen 4 Bits, indem er die angelegte Eingangsspannung konvertiert. Die unteren 4 Bits, also die Verfeinerung, werden in darauf folgenden Taktzyklus errechnet. Dies geschieht indem für die ersten 4 Bits mithilfe eines D/A-Wandlers wieder eine Spannung ausgegeben wird, diese mit einem Subtrahierer von der Eingangsspannung abgezogen und mit dem zweiten  $n/2$ -bit A/D Wandler umgesetzt wird.

Abgesehen von den Kosten haben Parallelwandler auch einen hohen Leistungsverbrauch. Sie sind jedoch sehr schnell und werden in der Videobearbeitungs- und Kommunikationstechnik eingesetzt. [ELE95]

#### **2.2.2.2 Wägeverfahren / iterative Verfahren (Sukzessive Approximation)**

Bei diesem Verfahren kommt zusätzlich noch ein Digital/Analog Wandler zum Einsatz. Zunächst setzt man das Ausgaberegister auf null. Man setzt dann das höchstwertige Bit dieses Ausgaberegisters auf Eins und lässt den Digital/Analog-Wandler die dazugehörige Spannung ausgeben. Diese wird über einen Komparator mit der angelegten Spannung verglichen. Ist sie niedriger, so belässt man das Bit auf Eins, sonst setzt man es auf Null.



- $U_{in}$  - zu konvertierende Analoge Eingangsspannung
- $U_{ref}$  - Referenzspannung (Maximalspannung)
- $U_{out}$  - konvertierte Spannung aus dem D/A-Wandler
- MSB - höchstwertiges Bit (Most Significant Bit)
- LSB - niederwertigstes Bit (Least Significant Bit)
- SAR - Successive Approximation Register (Ausgaberegister)
- B1, B2, B3 - Bit 1 - 3, Ausgabebits des SAR

Abbildung 2.13: 3-bit A/D-Wandler mit Wägeverfahren (Blockschaltbild)

Was man hier gemacht hat, ist folgendes: Durch das Setzen des höchsten Bits wurde über den D/A-Wandler die halbe Referenzspannung erzeugt. Entscheidet man nun, dass die angelegte Spannung größer ist, so muss man kleinere Spannungen nicht mehr überprüfen. Ist sie kleiner, braucht man sich um größere nicht mehr zu kümmern. Man hat somit die möglichen Restspannungen halbiert und bleibt damit in linearer Laufzeit (zur Anzahl Bits).

Das Setzen des zweiten Bits erzeugt nun entweder  $\frac{1}{4} U_{ref}$  oder  $\frac{3}{4} U_{ref}$ , je nachdem ob man das erste Bit gelöscht oder gesetzt gelassen hat.



Macht man das nun für alle Bits (die vorher überprüften Bits dürfen nicht gelöscht werden), so hat man am Ende den Binärkode der angelegten Spannung. Mathematisch wird das ganze dann so ausgedrückt:

$$U(i) = U(i-1) - (-1)^{b_{i-1}} \cdot 2^{-i} \cdot U_{\text{ref}} \quad (2.12)$$

$$\text{mit } b_i = \begin{cases} 1 & \text{für } U_{\text{in}} > U(i); U(1) = U_{\text{ref}} / 2 \\ 0 & \text{sonst} \end{cases} \quad (2.13)$$

$U(i)$  ist dabei die Ausgangsspannung am D/A-Wandler nach Schritt  $i$ . Die Gleichung besagt, dass je nachdem ob die Spannung im vorherigen  $(i-1)$ -ten Schritt größer oder kleiner als die Eingangsspannung war, der neu berechnete Spannungsanteil ( $2^{-i} \cdot U_{\text{ref}}$ ) von der Ausgangsspannung des D/A-Wandlers abgezogen oder zu ihr hinzuaddiert wird.

Benutzt man statt der Halbierung der Spannungsnormale das Umladen von Kondensatoren (bei gleicher Kapazität eines hinzugeschalteten Kondensators halbiert sich die Spannung) so erhält man einen schaltungstechnisch sehr effizienten Wandler (Shannon-Rock-Dekoder).

Eine weitere Möglichkeit einen A/D-Wandler nach dem Wägeverfahren zu bauen, ist indem man das SAR durch einen vorwärts/rückwärts Zähler austauscht. Man erhält dann einen Nachlaufwandler (Tracking-ADC). Mithilfe des DA-Wandlers wird solange die kleinste Spannung aufsummiert (oder abgezogen), bis die Eingangsspannung erreicht wird. Es sind  $2^n - 1$  Schritte erforderlich und damit zählt der Nachlaufwandler eher zu den Zählverfahren. Vom Schaltungskonzept her zählt er aber zu den Wägeverfahren-Wandlern.

Wandler nach dem Wägeverfahren sind sehr verbreitet und stellen einen guten Kompromiss zwischen Schaltungsaufwand und Geschwindigkeit dar. Die üblichen Geschwindigkeiten sind zwischen 0,1 bis 1 MHz (oder auch mehr). [ELE95]

### 2.2.2.3 Zählverfahren

Ein Zählverfahren A/D-Wandler benutzt nur ein Referenznormal, welches so lange aufsummiert wird, bis die Eingangsspannung erreicht wird. Zählt man die Zahl der Schritte mit einem Binär-Zähler, so speichert dieser am Ende des Wandlungsvorgangs den gesuchten Digital-Wert. Der Vorteil dieses Verfahrens ist, dass nur ein Spannungsnormal gebraucht wird. Allerdings wird ein n-Bit-A/D-Wandler bis zu  $2^n - 1$  Schritte benötigen.

Ein wichtiger Vertreter des Zählverfahrens ist der Zweiflanken-A/D-Wandler (Dual-Slope-ADC), der zur Gruppe der Sägezahnumsetzer zählt. Ein Sägezahnumsetzer verwandelt das analoge Eingangssignal in eine proportionale Zwischengröße (Frequenz, Zeit). Diese setzt man dann mit Hilfe eines Zählers in ein Digitalwort um.

Der Zweiflanken-Wandler integriert zuerst die Eingangsspannung während einer Zeit  $t_1$ . Die Knotenspannung am Ausgang des Integrierers steigt damit auf:

	$u_A(t_1) = \frac{1}{RC} \int_0^{t_1} U_E dt = \frac{U_E t_1}{RC}$	(2.14)
--	--	--------

Zum Zeitpunkt  $t_1$  wird eine Referenzspannung mit umgekehrter Polarität angelegt. Auch der Zähler wird gestartet.  $u_A(t)$  geht nach einer Zeit  $t_x$  durch Null. Dann wird der Zähler gestoppt. Aus

$$u_A(t_1 + t_x) = u_A(t_1) - \frac{1}{RC} \int_{t_1}^{t_1+t_x} U_{Ref} dt = \frac{U_E t_1}{RC} - \frac{U_{Ref} t_x}{RC} = 0 \quad (2.15)$$

ergibt sich

$$t_x = t_1 \frac{U_E}{U_{\text{Ref}}} \quad (2.17)$$

Die Integrationszeit ergibt sich aus einer Zählfrequenz  $f_0$ :

$$t_1 = N_1 / f_0 \quad (2.18)$$

Somit entspricht die Zahl der gezählten Impulse:

$$N_x = f_0 \cdot t_x = f_0 \cdot \left(\frac{N_1}{f_0}\right) \cdot \left(\frac{U_E}{U_{\text{Ref}}}\right) = N_1 \cdot \left(\frac{U_E}{U_{\text{Ref}}}\right) \quad (2.19)$$

Dieses Verfahren hat viele Vorteile. Das Messergebnis ist beispielsweise unabhängig von der Integrationskonstante oder der Taktfrequenz. Selbst kurzzeitige Störungen haben einen geringen Einfluss. Dieses Verfahren kommt oft in der Messtechnik zum Einsatz (z.B.) in Digitalvoltmetern. Dort stört die geringe Wandlungsgeschwindigkeit nicht. [ELE95]

### **2.2.3 Signalberechnung mittels Potentiometer und Analog/Digital-Wandler**

Eine Kombination aus einem Potentiometer und einem Analog/Digital-Wandler bietet eine Möglichkeit, Weg oder Winkelmessungen durchzuführen und sie zu digitalisieren. Befestigt man beispielsweise ein Potentiometer an ein Gelenk, so wird sich die Spannung proportional zum Ausschlagwinkel des Gelenks bewegen. Diese Spannung kann mit einem A/D-Wandler zu einem digitalen Signal verarbeitet werden.

## **2.3 State-of-the-Art**

### **2.3.1 Stop-Motion-Animation, Rotoscoping und Animatronics**

Animation ist eine Aneinanderreihung von Bildern, welche, wenn sie genügend Schnell abgespielt werden, dem menschlichen Auge eine Bewegung vortäuschen. Möchte man Objekte animieren, welche sich von selbst gar nicht oder nicht realistisch genug bewegen können, kommt man in der klassischen Filmherstellung nicht um das Stop-Motion Verfahren herum.

Das Stop-Motion ist eine Technik, bei der das zu animierende Objekt bei jeder Änderung seiner Position, seiner Stellung oder seines Aussehens, von Hand verändert wird. Vom Objekt wird dann ein Schnappschuss gemacht und das Objekt wird neu eingestellt. Ein Nachteil des Verfahrens ist, dass bei den bewegten Bildern kein Motion-Blur (Bewegungsunschärfe) zu sehen ist. [DIG01]

Schwieriger stellt sich die Situation dar, wenn man Objekte darstellen möchte, die es eigentlich gar nicht gibt. Eine der Möglichkeiten wäre es diese Objekte zu zeichnen. Hierfür könnte sich das Rotoscoping eignen. Beim Rotoscoping wird auf ein bereits bestehendes aufgenommenes Bild das zu animierende Objekt hineingezeichnet. Dabei kann es sich um 2D-Animation handeln, die pro Bild mehrere Schichten braucht, auf welche dann beispielsweise der Hintergrund oder das animierte Objekt gezeichnet werden. Es kann sich aber auch um 3D-Animation handeln, bei welcher der Hintergrund zwar auch starr bleibt, aber mittels Computern ein sich bewegendes 3D-Modell eingefügt wird. Dieses Modell kann beispielsweise mit einer 3D-Software und seine Bewegung mit Motion-Capturing erstellt werden. [DIG01][ANI02]

Will man aber etwas sehr realistisch wirken lassen, ist diese Methode sehr aufwendig - wenn nicht gar unmöglich - zu realisieren. Eine andere Möglichkeit ist, solche Objekte oder Teile davon detailgetreu nachzubauen, und sie dann vor der Kamera mechanisch oder manuell zu bewegen. Diese Figuren nennt man Animatronics. [DIG01]

Der Aufwand diese Puppen zu bauen ist groß und die Animation oft schwierig. Die beste Alternative ist jedoch, solche Objekte grafisch mit einem Computer darzustellen.

Die Frage, die sich dabei stellt, ist, wie man sehr gelenkige Objekte (wie Tiere) realistisch animiert. Man kann unter Zuhilfenahme von 3D-Software natürlich auch jeden Schritt oder jedes Bild der Animation manipulieren, es gibt aber zwei Möglichkeiten um solche Bewegungen schneller und zum Teil realistischer zu digitalisieren.

## **2.3.2 Motion-Capturing**

### **2.3.2.1 Grundlagen**

Das Motion-Capturing ist eine fortschrittliche Animationstechnik zum Aufzeichnen von Bewegungen. Animatoren können dies mit Hilfe von Computern bewerkstelligen und es dann auf bereits vormodellierte Computercharaktere anwenden. Dies wird meist mit Menschen oder Tieren durchgeführt. Dabei werden an möglichst vielen für die Bewegung wichtigen Körperstellen Markierungspunkte angebracht. Diese Markierungspunkte werden dann während der Bewegung von einem Computer ausgewertet und daraus eine Animation erstellt.

Das Motion-Capturing kann auch dazu benutzt werden, Grundbewegungen zu erzeugen, welche dann mit Hilfe anderer Animationstechniken verbessert werden können. So werden zum Beispiel feine Finger- oder Gesichtanimationen erst später zu diesen primären Bewegungen hinzugefügt.

Es gibt nicht viele High-End-Systeme, die bereits mit allen Komponenten gekauft werden können. Diese Systeme beinhalten einen Computer mit genügend Anschlussmöglichkeiten für die Daten der Sensoren. Des Weiteren muss auch eine genügend große Datenbandbreite gegeben sein, um alle Daten mit einer angemessenen Geschwindigkeit auch verarbeiten zu können. Gute Motion-Capturing-Systeme werten gleichzeitig bis zu 70 Markierungspunkte aus, es gibt aber auch welche, die mit etwa einem Dutzend auskommen müssen.

Es gibt neuartige Motion-Capturing-Verfahren, die es erlauben, Objekte ohne Markierungspunkte zu animieren. Dabei wird ein Objekt von einer Kamera ausgewertet und die Umrisse erkannt. Diese Umrisse dienen dann bei der Nachbearbeitung als Ausgangspunkt für die Animation. [ART96]

Folgende Grund-Verfahren des Motion-Capturing unterscheidet [ART96]:

1. optisches Motion-Capturing
2. prothetisches Motion-Capturing
3. magnetisches Motion-Capturing
4. akustisches Motion-Capturing

[DVO05] erwähnt auch das Satelliten-Motion-Capturing, welches die Signale der Positionierungssysteme vom amerikanischen Navstar-GPS oder russischem GLO-NASS nutzt, um die Position der zu verfolgenden Objekte zu bestimmen. Auch eine weitere Unterteilung des optischen Motion-Capturing in optisch-photogrammetrische, optisch-stereoskopische und optisch-videobasierte Systeme wird vorgenommen. Diese Arbeit wird im Einzelnen die Unterscheidung der in [ART96] vorgestellten Systeme zeigen.

### **2.3.2.2 Optisches Motion-Capturing**

Werden die Markierungspunkte optisch ausgewertet, so spricht man vom optischen Motion-Capturing. Die Markierungspunkte sind dabei Licht reflektierende Stellen, welche entweder direkt auf den Körper aufgetragen werden, oder sich an Anzügen befinden, die dann dem zu animierenden Objekt übergezogen werden. Während der Aufnahme wird die Szene von Lichtquellen beleuchtet. Die Auswertung erfolgt hierbei mit Kameras, die ihre Bilder einem Computer liefern. Der Computer erkennt dabei zu jedem Zeitpunkt die Position der einzelnen Markierungen und kann damit eine Animation erstellen. [ART96]

Optisches Motion-Capturing ist in der Hinsicht praktisch, dass die aufzunehmenden Akteure eine große Bewegungsfreiheit besitzen. Auch für die Animation von mehr

als einem Objekt ist diese Methode sehr gut geeignet. Das größte Problem ist hierbei jedoch, dass die Reflektionspunkte während der Aufnahme zeitweise verdeckt sein können. Dieses Problem wird umgangen, indem man die Anzahl der Kameras erhöht. Das wiederum erhöht aber auch die Informationsmenge, die zu verarbeiten ist. [ART96]

### **2.3.2.3 Prothetisches (mechanisches) Motion-Capturing**

Bei dieser Variante werden Potentiometer an den Gelenken der Akteure benutzt. Hiermit kann also der Winkel der einzelnen zu messenden Gelenke übermittelt werden. Das größte Problem, das sich dabei ergibt, ist dass Potentiometervorrichtungen klobig sind und den Akteur in seiner Bewegungsfreiheit einschränken. Diese Systeme werden schon längere Zeit benutzt und zwar vor allem im medizinischen Bereich, wo der Bewegungsbereich einzelner Gelenke gemessen wird. Daher kommt auch der Name. Man könnte es aber auch als mechanisches Motion-Capturing bezeichnen. [ART96]

### **2.3.2.4 Magnetisches Motion-Capturing**

Setzt man die ganze Szene einem Magnetfeld aus, und misst die Position einzelner Körperstellen mit Magnetfeldsensoren, dann spricht man vom magnetischen Motion-Capturing. Dabei muss man aufpassen, dass die Messung nicht von metallischen Gegenständen gestört wird. Manche Systeme benutzen Kabelverbindungen von Sensoren zum Computer. Da sich die Sensoren direkt am Körper befinden, können die Kabel den Akteur bei der Bewegung stören. [ART96]

### **2.3.2.5 Akustisches Motion-Capturing**

Das akustische Motion-Capturing basiert auf akustischen Signalgebern, die mittels Schallwellen zu jedem Zeitpunkt ihre Position bestimmen können. Hierbei ist zu achten, dass die Oberflächen um die Szene herum keine Echos erzeugen, die das System und die Auswertung stören. [ART96]

### **2.3.2.6 Ein bestehendes System: Das „Motion Analysis's Eagle“**

Eines der neuesten Systeme von Motion Analysis Corporation, einer führenden Firma in Sachen Motion-Capturing, heißt Eagle Digital System. Dieses System besteht

aus Kameras, einem Hub für dessen Verbindung und einer Software zur Erfassung aller Daten. Es ist ein optisches Motion-Capturing-System.



Abbildung 2.14: Die Kamera des Eagle-Systems



Abbildung 2.15: Der Hub des Eagle-Systems

Das System ist Real-Time fähig. Das heißt, dass der Benutzer alle Bewegungen des Objekts sofort am Bildschirm verfolgen kann. Es können etwa 600 Millionen Pixel in einer Sekunde verarbeitet werden. Das sind 500 Frames/s bei einer Auflösung von 1280x1024. Verkleinert man die Auflösung auf 1280x512 oder 1280x256 so sind auch 1000 oder 2000 Frames/s möglich.

Die Signalaufnahme erfolgt in der Kamera, welche das Signal dann per TCP/IP an den Hub und den dahinter liegenden Rechner weiterleitet. Die Software der Kamera kann direkt aus dem Internet (ohne dass die Kamera abgebaut werden muss) aktualisiert werden.

Das System benutzt bis zu 8 Kameras, welche Problemlos in 2 mitgelieferten Taschen transportiert werden können. Das zu animierende Objekt wird mit Punkten im roten oder infraroten Bereich versehen.



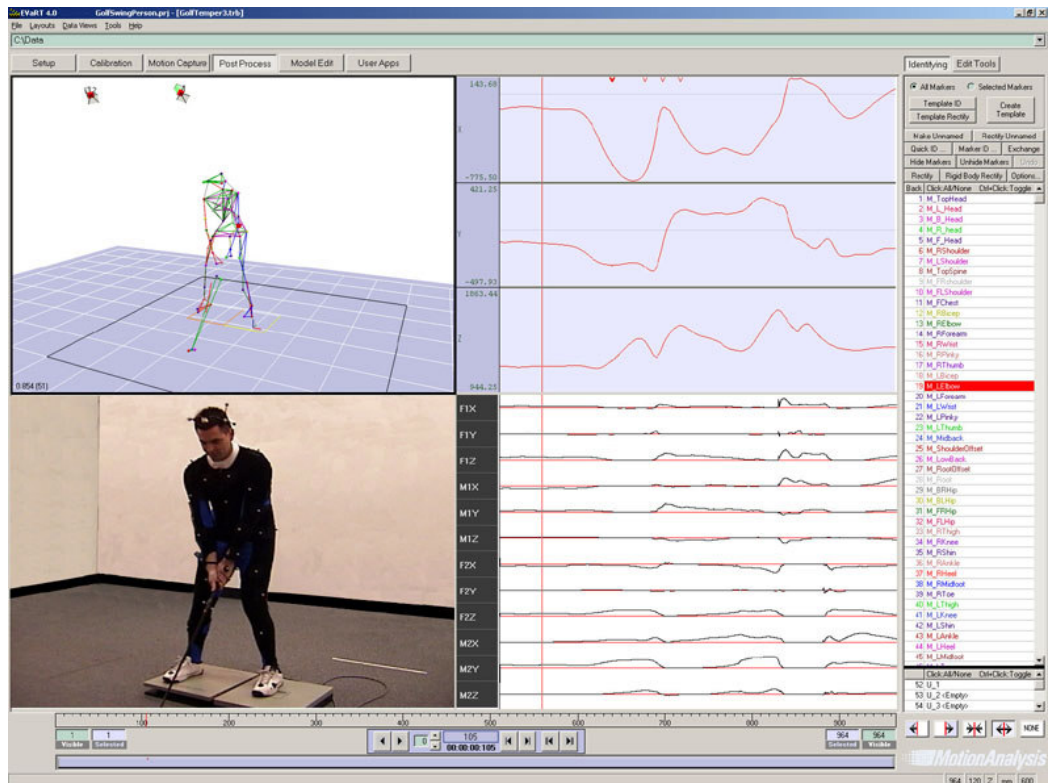


Abbildung 2.16: Die Darstellungssoftware des Eagle-Systems

Es werden von Motion Analysis auch weitere optische Motion-Capturing-Systeme angeboten, wie das Hawk oder das Falcon System. Auch eine weitere Version des Eagle, das Eagle4 mit einer 4 Megapixel-Kamera, steht zur Verfügung. [MOT06]

### 2.3.3 Dinosaur Input Device

Das Dinosaur Input Device (DID) wurde 1993 für den Film „Jurassic Park“ von Steven Spielberg entwickelt. Spielberg drehte einen Film über geklonte Dinosaurier, welche so realistisch wie möglich wirken sollten. Für die Animation der über 50 Szenen in denen Dinosaurier vorkamen, benutzte man verschiedene Techniken. Zum einen sollten erfahrene Stop-Motion Animatoren kleine Dinosaurier-Puppen benutzen, welche dann nachträglich in den Film eingebaut werden sollten. Zum anderen benutzte man 3D-Software, um mit ihrer Hilfe Dinosaurier zu kreieren und zu animieren. Der dritte Ansatz war der Einsatz lebensgroßer Animatronics-Figuren. Animatronics sind nachgebaute Kreaturen, welche lebensecht wirken sollen.

Bei der Entwicklung der Dinosaurier konnte man diese nicht durch Motion-Capturing animieren, weil es keine lebenden Dinosaurier gibt. Selbst vergleichbare Wesen existieren nicht. Und auch wenn es sie gäbe, wäre eine Instruierung durch einen menschlichen Regisseur eine zumindest problematische Angelegenheit.

Zu Beginn war es noch nicht klar, welcher der drei Ansätze am besten geeignet war, und somit wurden alle drei weiterverfolgt. Die Computergraphik-Animatoren sollten nur ein paar Szenen, die mit traditionellem Stop-Motion hergestellt wurden, verfeinern. Doch sie entschieden sich, eine Herdenszene komplett in Eigenregie zu entwickeln und sie Steven Spielberg zu zeigen. Spielberg war über die Authentizität erstaunt, und entschied, das Stop-Motion-Verfahren aus der Produktion herauszunehmen.

Bei einigen Szenen, die mit 3D-Software hergestellt wurden, kam es zum störenden Jerkiness. Jerkiness entsteht, wenn für eine Animation zu wenig Frames in der Sekunde (unter 12) benutzt werden. Das menschliche Auge kann dann die einzelnen Bilder wahrnehmen und die Animation wird ruckelig. Die Animatoren konnten das Problem zunächst nicht beheben, und so entwickelte man das Dinosaur Input Device in der Hoffnung das Problem zu lösen. [ILM96]

Das DID ist im Wesentlichen eine metallische Puppe mit Gelenken, die mit einem Computer verbunden ist. Sie repräsentiert das Skelett eines Dinosauriers. An den Gelenken sind Sensoren angebracht, welche den jeweiligen Winkel des Gelenks messen. Die Bewegung eines Gelenks wird direkt auf ein im Computer generiertes Wireframe-Objekt des Dinosauriers übertragen. Dieses Objekt wird dann später mit den Details wie Hautstruktur oder Farbe versehen. Auch das Hinzufügen der Beleuchtung und der Kameraeinstellung für die ganze Szene erfolgt später. Das DID wird also ausschließlich für das Übermitteln der Bewegung des Objekts an das 3D-Animationsprogramm benutzt.

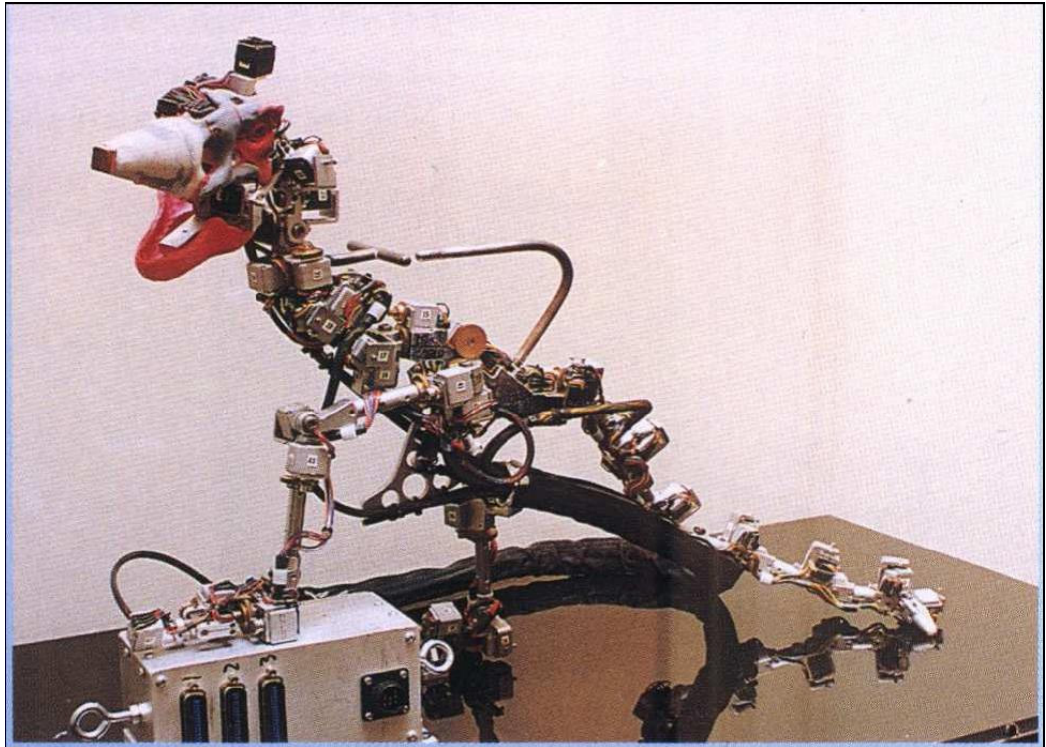


Abbildung 2.17: Dinosaur Input Device für einen Velociraptor

Der wesentliche Vorteil des DID gegenüber einer klassischen Stop-Motion-Puppe ist, dass die Animatoren sich zunächst nur um die Bewegung des Objekts kümmern müssen, da Beleuchtung und Kameraeinstellung später am Computer hinzugefügt werden. Auch das Fehlen der Körperdetails (das DID ist nur ein Skelett) ist von Vorteil für die Animatoren, weil sie nicht darauf achten müssen, irgendetwas vom Puppenkörper abzubrechen. Durch die Tatsache, dass die Animatoren die Gelenke direkt sehen, können sie, falls benötigt, ein Gelenk lockern oder fester machen, ohne lange danach zu suchen.

Die Puppe selbst ist ganz einfach aufgebaut. Die „Knochen“ bestehen aus Metall und an den Gelenken sind Optocodier angebracht. Es wurden Optocodier mit 255 Hell/Dunkel Übergängen benutzt. An den Gelenken selbst sind für zusätzliche Präzision Getriebeübersetzungen zu den Encodern eingebaut worden, so dass eine Umdrehung des Gelenks mehrere Umdrehungen des Sensors verursacht. Des Weiteren wurden auch beide Spuren (A und B) des Optocodiers ausgewertet, so dass die Auflösung auf 1024 Stufen vervierfacht werden konnte. Dies alles macht die Aus-

wertung der Bewegung sehr präzise. Eine Drehung um etwa ein Drittel Grad konnte so übermittelt werden.

Die Gelenke wurden so konzipiert, dass sie aus bis zu drei nebeneinander stehenden drehbaren Einheiten bestehen. So können sie auch mehr als einen Encoder drehen. Es konnten dadurch bis zu drei Freiheitsgrade eines Gelenks simuliert werden. Diese komplizierten Gelenke konnten aber aus Platzgründen nur am Hals und den Knöcheln realisiert werden. Kugelgelenke wurden nicht eingesetzt.

Insgesamt ist die Puppe etwa 30% größer als eine übliche Stop-Motion Puppe. Die Befürchtungen, dass die Animatoren mit der Größe nicht klar kommen, haben sich nicht bewahrheitet. [DID95]

Mit dem DID konnten nun die Stop-Motion Animatoren etwas anfangen. Sie konnten für die Szenen in denen Dinosaurier-Bewegungen benötigt waren, nun mit etwas arbeiten, was sie kannten. Später wurde das ursprüngliche Jerkiness-Problem von den Computer-Animatoren gelöst, aber das DID benutzte man trotzdem für etwa 15 Aufnahmen, in welchen insgesamt etwa 20 Dinosaurier vorkamen. [ILM96]

Später wurde die Animationsmethode Direct-Input-Device oder Digital-Input-Device bezeichnet. Es bezeichnet ganz allgemein jede Eingabe-Puppe, welche zum Zwecke der Stop-Motion-Animation ihre Stellung einem Computer mitteilen kann, mit welchem man diese Aufzeichnung später einen Computergraphikcharakter steuern kann. [DIG01]

## **2.3.4 Datenhandschuhe**

### **2.3.4.1 Grundlagen**

Der Mensch benutzt die Hand in fast jeder Interaktion mit seiner Umwelt. Aufgrund dessen wurde auch die Bewegungsaufnahme der menschlichen Hände erforscht. Das

Eingabemedium für die Digitalisierung dieser Bewegung ist der Datenhandschuh. Es benötigt die folgenden Informationen:

- Absolute Lage der Hand im Raum
- Ausrichtung der Hand
- Lage der Finger zueinander

Die Lage und die Ausrichtung der Hand werden mit einem, am Handrücken befestigten Sensor ermittelt. Für die wichtigste Aufgabe, die Position der Finger wurden verschiedene Methoden entwickelt. [VRE94] [VRP95]

#### **2.3.4.2 DataGlove und Dextrous Handmaster**

Das Problem der Fingerbewegung wurde beim DataGlove mit Glasfasern gelöst. Der Handschuh wurde im Auftrag der NASA entwickelt. Er besteht aus einem elastischen Stoff, der sich der Hand anpasst. Auf dem Handrücken und über die Finger verlaufen Glasfasern, die sich mit jeder Bewegung krümmen. An einem Ende der Faser ist eine Leuchtdiode befestigt, welche Licht hinein sendet. Am anderen Ende empfängt eine Fotodiode dieses Licht. Je mehr die Hand und dadurch die Glasfaser gekrümmt ist, desto mehr Licht tritt aus der Faser heraus und desto weniger kommt am Empfänger an. Dies wird alles von einem Prozessor am Handschuh ausgewertet und an einen Computer verschickt.

Die Messung erfolgt am jeweils obersten und untersten Gelenk eines Fingers und bei manchen Ausführungen wird auch die Spreizung der Finger gemessen. Der Vorteil dieses Handschuhs ist seine Anschmiegsamkeit.

Der Handschuh muss für jeden Benutzer neu kalibriert werden, da jeder Mensch andersförmige Hände hat. Außerdem wird bei dauerhafter Benutzung der Handschuh verrutschen, was ebenfalls eine Rekalibrierung nötig macht.

Es gab auch eine andere Version des DataGloves, den CyberGlove. Dieser hatte anstatt der empfindlichen Glasfasern, Metallbänder als Sensoren. Diese änderten ihren

Widerstand wenn sie verbogen wurden und so konnte die Position der Finger ermittelt werden. Der Handschuh war robuster und konnte leichter industriell hergestellt werden.

Es gibt noch viele andere Datenhandschuh-Implementierungen. Ein weiteres Beispiel wäre der Dextrous Handmaster. Dieser benutzt ein Exoskelett, also ein äußeres Skelett für die Bewegungsaufnahme. Dieses Skelett kann man mittels Bänder an die Fingerknochen befestigen.

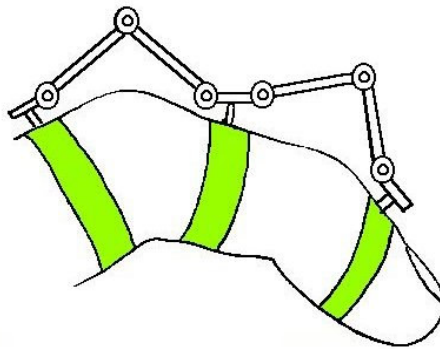


Abbildung 2.18: Funktionsweise des Exoskeletts eines Dextrous Handmaster

Bei jeder Bewegung der Finger bewegt sich das Skelett mit und übermittelt mit Hilfe von kleinen Magneten und Hall-Effekt-Sensoren ein Signal. Die Magneten erzeugen damit einen Spannungsabfall im Sensor, welcher weitergeleitet und ausgewertet wird.

Beide Systeme benutzen einen Polhemus-Positionssensor am Handrücken, welcher die Lage und die Ausrichtung der Hand misst. Dieser Sensor wurde von der Firma Polhemus-Navigational-Systems entwickelt und misst die Lage einer Spule im elektromagnetischen Feld. Drei zueinander orthogonale Spulen befinden sich jeweils im Sender und Empfänger. Jede der drei Senderspulen gibt nacheinander ein Signal ab, welches in den Empfängerspulen gemessen wird. Die Messung erfolgt aufgrund der Tatsache, dass ein größerer Strom in einer Spule induziert wird, wenn sie parallel zum induzierenden Magnetfeld ist. [VRE94][VRT94]

### **2.3.5 Datenanzüge**

Datenanzüge werden für die Ermittlung menschlicher Ganzkörperbewegung benutzt. Es gibt sie in verschiedenen Ausführungen. Sie werden meistens für das vorher beschriebene Motion-Capturing-Verfahren eingesetzt.

Der DataSuit-Anzug basiert auf der gleichen Technologie wie der DataGlove. Glasfaser sind am Anzug angebracht, welche durch ihre Streckung und Stauchung die Lichtdurchlässigkeit verändert. Mit Leuchtdioden und Fotosensoren wird indirekt auf eine bestimmte Körperstellung geschlossen. [VRE94]

Den Körperanzug im Film „Toys“, der speziell für den Film gebaut wurde, benutzte man für verschiedene Szenen. Beispielsweise saßen in einer Szene einige Generäle an einem Tisch. Jeder von ihnen trug einen der Datenanzüge. So war es möglich diese Szene nachträglich so zu animieren und anzuzeigen, als ob man mit speziellen Röntgenstrahlen ihre Skelette beobachten konnte. Im Körperanzug waren an den Gelenken Potentiometer angebracht, welche die Bewegung gemessen haben. Diese Art der Bewegungsmessung wird auch als mechanisches (prothetisches) Motion-Capturing bezeichnet. [DID95][MOR93]

### **2.3.6 Nachteile vorgestellter Systeme**

Die bisher vorgestellten Animationstechniken und –Werkzeuge besitzen alle ihre Vorteile und Schwächen. Es sollte vorher gut überlegt sein, was man animieren möchte, und welchem Zweck das dienen soll. Man muss sich sowohl die Frage stellen, wie realistisch das Endergebnis wirken muss, als auch ob der Aufwand und die benötigten Kosten für die Realisierung notwendig sind.

Das Motion-Capturing sollte nicht bei Animationen eingesetzt werden, die imaginäre Charaktere darstellen. Unter diese Kategorie fallen fast alle Zeichentrickfilme, sowie realistisch wirkende Filme mit erfundenen Wesen oder Objekten. Dies kommt dadurch zustande, dass selbst ein ungeübter Beobachter merkt, dass hinter einer, auf

Motion-Capturing basierender Animation, die Bewegung eines Menschen steuert. Dann wirkt jedoch ein erfundenes Wesen nicht mehr authentisch. Denn der Hintergedanke, der bei solchen Kreaturen dahinter steht, ist meist jener, dass sie in erfundenen Welten existieren, welche nicht, oder nicht vollständig, den Gesetzen unserer Physik gehorchen. Ein Mensch, der sich auf einer Motion-Capturing-Bühne bewegt, wird aber diesen Gesetzen unterliegen und er kann oft nicht die Aktionen durchführen, die sich ein Animator ausgedacht hat. Das bedeutet, dass das Motion-Capturing für die Darstellungen benutzt werden darf, bei denen eine zumutbare Bewegung von einem Menschen oder Tier verlangt werden kann.

Andererseits ist das Motion-Capturing für genau diese Szenen hervorragend. Denn es bietet einem die Möglichkeit, computergenerierte Menschen auch durch Menschenbewegungen zu animieren. Möchte man also virtuelle Welten benutzen in denen sich auch Menschen oder Tiere befinden, so bietet das Motion-Capturing-Verfahren eine sehr gute Möglichkeit diese Wesen auch zu animieren.

Das Dinosaur-Input-Device (DID) schließt eine Lücke des Motion-Capturing. Damit ist es möglich auch Wesen zu animieren, die nicht existieren. Die zu animierenden Dinosaurier können als Puppen nachgebaut werden, welche dann als Eingabegeräte dienen. Jede vorstellbare Bewegung ist zu realisieren. Dafür braucht man natürlich erfahrene Animatoren, die sich im Stop-Motion-Verfahren bewährt haben.

Der größte Nachteil des DID ist, dass es speziell für „Jurassic Park“ gebaut wurde. Des Weiteren wurden nur zwei Dinosaurier damit realisiert. Da der Ansatz nicht weiterverfolgt wurde, ist das DID als Eingabegerät nicht universell einsetzbar. Ein weiterer Nachteil ergibt sich auch als Konsequenz des ersten. Es ist nicht möglich, bestehende Puppen zu erweitern. Das kommt dadurch zustande, dass eine DID-Puppe in sich abgeschlossen ist. Weitere Gliedmaßen oder Gelenke können nicht einfach so ohne weiteres hinzugefügt werden.

Die Handmaster sind für ihr Einsatzgebiet sehr gut, es ist aber klar, dass sie nicht universell benutzt werden können. Sie sind für die feine Animation der Hände geeignet. Auch eine mixed-reality Umgebung ist mit ihnen zu realisieren.



Auch die Kosten müssen berücksichtigt werden. Egal, ob es sich um Eingabegeräte, wie das DID, Handmaster-Geräte oder ein Motion-Capturing-System handelt, die Hardware ist meistens mehrere Tausend bis zu hundert Tausend US-Dollar oder Euro teuer. Einige Datenhandschuhe befinden sich in einer Preisregion von mehreren Hundert US-Dollar. [FDT06] Optische Motion-Capturing Systeme befinden sich in einer Preisregion von 50000 bis zu 250000 US-Dollar. Magnetische Systeme sind etwas günstiger. Sie kosten zwischen 5000 und 150000 US-Dollar. Die Preise für Mechanische Systeme sind zwischen 5000 und 10000 US-Dollar [EXP06]. Das bedeutet, dass diese Geräte für private Anwender, oder für kleinere Animationsunternehmen nicht ohne weiteres erschwinglich sind.



### 3. Anforderungsanalyse

Ein neues Animationssystem, sollte möglichst universell, erweiterbar, präzise und kostengünstig sein.

Universell sollte es in der Hinsicht sein, dass möglichst viele Objekte damit animiert werden können. Das bedeutet, dass man das gleiche Gerät für die Animation verschiedener Kreaturen, Wesen oder sonstiger zu animierenden Objekte benutzt werden kann. Die Benutzbarkeit des Gerätes sollte nicht von einem bestimmten Ort, oder dem System an den es angeschlossen wird, abhängen.

Des Weiteren sollte das System erweiterbar sein. Das heißt, dass man die Eingabemöglichkeiten ohne viel Aufwand vergrößern kann. Soll das System beispielsweise den Winkel der Gelenke messen, so sollen mühelos weitere Gelenke angeschlossen werden können.

Die Messungen sollen präzise sein. Der Computer soll möglichst das einlesen, was auch vom Benutzer eingegeben wurde. Ist ein Gelenk in einer  $90^\circ$  Stellung, so sollen auch möglichst  $90^\circ$  eingelesen werden und nicht  $80^\circ$  oder  $100^\circ$ . Dies hängt natürlich größtenteils von der verwendeten Hardware, aber auch von der Software, welche die Eingabe auswertet, ab.

Nicht zuletzt sollte das System auch für einen kleinen Anwender finanzierbar sein. Das bedeutet, dass es sich nicht in einer Preisklasse von mehreren Tausend Dollar oder Euro befinden soll, sondern eher bei höchstens einigen Hundert bis maximal Tausend Euro.



## **4. Eigenes Konzept – Die lokationssensitive Puppe**

### **4.1 Die Grundidee**

Das hier vorgestellte Animationssystem sollte die obigen Anforderungen erfüllen. Es muss also universell, erweiterbar, präzise und darüber hinaus kostengünstig sein.

Um ein universelles System zu erhalten, muss ein gewisser Grad an Modularität und Erweiterbarkeit gegeben sein. Denn nur wenn es möglich ist ein System leicht und schnell zu erweitern, ist es möglich das Ganze auch für verschiedene Anwendungsgebiete zu benutzen.

Deswegen fällt die Entscheidung auf eine modulare, aus separaten Skelett-Teilen bestehende Puppe. Sie soll darüber hinaus Gelenke besitzen, dessen Stellung man messen kann. Die Skelett-Teile sind miteinander entweder starr, oder an diesen Gelenken beweglich, verbunden.

Um die Gelenkstellung auszuwerten, wird ein Verbund aus Potentiometer und A/D-Wandler benutzt. Dies ist die kostengünstigste Alternative. Denn Optoencoder und optische Resolver sind zwar genauer, jedoch auch teurer. Außerdem würde ein Optoencoder mit 256 Stufen insgesamt 10 Drähte haben. Das bedeutet, dass eine Puppe mit 16 Gelenken, insgesamt 160 Drähte führen würde. Die Handhabung der Puppe würde darunter leiden, denn ein erforderlicher Schutz der Drähte würde sehr viel Platz einnehmen.

Da Potentiometer einen nicht ganz konstanten Widerstand liefern und A/D-Wandler ein Rauschen besitzen, wird sich die Treibersoftware auch damit beschäftigen, den abgelesenen Wert zu glätten, um verwertbare Ergebnisse zu erzielen.

Die Skelett-Teile werden aus Holz bestehen, da es sich leicht bearbeiten lässt und damit einen schnellen Puppenaufbau ermöglicht. Holz ist außerdem leicht und kostengünstig. Die Gelenke werden mit Schrauben befestigt sein und zwar so, dass sie

sich trotzdem drehen lassen. Die Holzteile, welche verbunden sein müssen, an dessen Enden sich jedoch keine Gelenke befinden, werden fest miteinander verschraubt. Sollen Holzteile senkrecht aufeinander stehen, dann werden metallische Verbindungen benutzt.

## **4.2 Der Puppen-Aufbau**

### **4.2.1 Grundlagen**

Bei der vorgestellten Puppe handelt es sich vom Gerüstaufbau her, um ein dinosaurierartiges Wesen, vergleichbar beispielsweise mit einem Macroplata, einem Wasserdinosaurier. Die Beinbewegungen wären auch mit denen einer Wasserschildkröte vergleichbar.



Abbildung 4.1: Der Wasser-Dinosaurus Macroplata

Die Bewegungen sind bei diesem Modell stark vereinfacht. Die Gelenke lassen sich nur in einer Ebene bewegen.

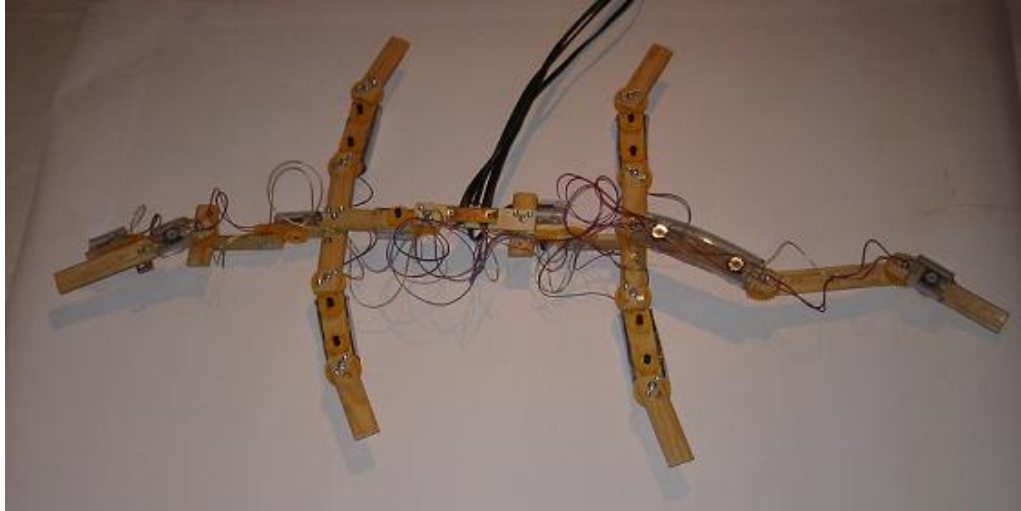


Abbildung 4.2: Die entwickelte lokationssensitive Puppe

#### 4.2.2 Das Skelett

Das Skelett ist von der Spitze des Kopfes bis zum Ende des Schwanzes 85 cm lang und an der breitesten Stelle (Hinterbeine) 44,7 cm breit. Es besteht aus Holz, da es ein leicht zu verarbeitendes Material mit einer geringen Dichte ist. Das macht die ganze Puppe zusätzlich leichter, als wenn sie aus Metall wäre. Zum Einsatz kommen kleine Holzbrettchen, welche je nach Bedarf zwischen 6,5 und 15 cm lang sind. Alle Brettchen besitzen eine Breite von 21 mm und eine Höhe von 5 mm.

Die Brettchen sind untereinander mit Schrauben verbunden. Ist die Verbindung ein Gelenk, so lassen sie sich dort drehen. An einer Verbindung, die kein Gelenk ist, sind je zwei Brettchen starr miteinander verbunden.

Da ein Gelenk in nur einer Ebene drehbar ist, ist es klar, dass es eine Möglichkeit geben muss waagrecht zu bewegende Puppenteile mit den senkrechten zu bewegenden zu verbinden. Dies wurde mit Hilfe von „L-Gliedern“ aus Metall gelöst, welche fest mit dem Holz verschraubt sind und die Möglichkeit des Ebenenwechsels bieten. Es finden sich genau 5 Ebenenwechsel auf der Puppe. Die hinterste Ebene ist die größte. Sie ist waagrecht. Das heißt, dass alle Gelenke sich in der waagerechten

Ebene drehen lassen. Sie beinhaltet 7 Gelenke. Es sind die 3 Schwanzgelenke und je 2 Gelenke auf den hinteren Beinen. Zur Mitte kommt es dann zum ersten Ebenenwechsel. Jetzt kommt das senkrechte Gelenk in hinteren Rumpf. Sofort danach kommt es wieder zum Ebenenwechsel zur zweitgrößten Ebene mit 5 Gelenken. Diese ist waagrecht. Zuerst kommt das vordere Rumpfgelenk und danach wieder zwei Beine, die Vorderbeine, mit zwei Gelenken pro Bein. Zum Kopf hin, kommt es wieder zum Ebenenwechsel, wo sich das senkrechte Halsgelenk befindet. Für das Kopfgelenk wird noch mal in die Waagerechte gewechselt. Das Kiefergelenk ist das letzte Gelenk. Es befindet sich in der Senkrechten und der Wechsel findet auf dem Kopf statt.

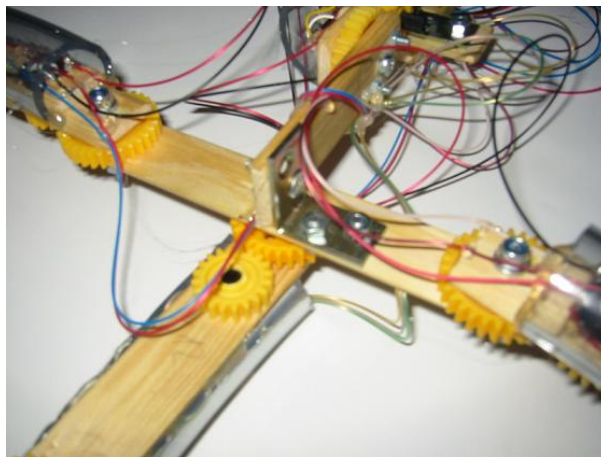


Abbildung 4.3: Das „L“-Verbindungsglied

Es sind also insgesamt 16 Gelenke, wovon sich 13 in der waagerechten und 3 in der senkrechten Position befinden.

Zwischen dem vorderen und hinteren Rumpfgelenk befindet sich ein 11 cm langes Brettchen, zu welchem alle Stromkabel führen und mit ins Holz verschraubten Metallplatten befestigt sind.

Zum Schutz der Potentiometer wurden auf diese Plastikschläuche geklebt. Das verhindert ein Berühren der Potentiometer und der Kabel bei der Bedienung der Puppe. Das bietet einen Schutz vor möglicher Beschädigung.



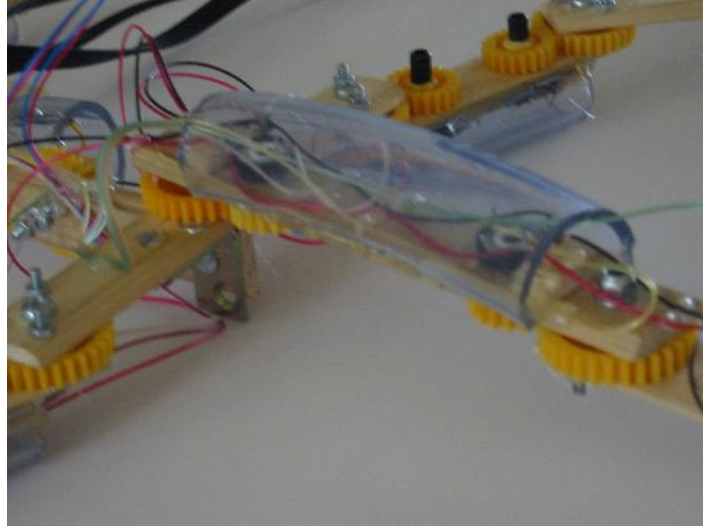


Abbildung 4.4: Der Schutzschlauch

### 4.2.3 Die Gelenke

Auf der Puppe sind insgesamt 16 Gelenke. Sie sind an verschiedenen Körperregionen angebracht und ihre Stellung wird von einem Potentiometer überwacht. Ein Gelenk besteht aus zwei mit einer Schraube verbundenen Hölzern. Beide besitzen ein Loch (6mm Durchmesser) an einem Ende, durch das die Schraube hindurchgeht und sie miteinander verbindet. Um die Schraube herum ist die Drehung frei durchführbar.

Um die Stellung durch das Potentiometer zu messen, sind verschiedene zusätzliche Bauteile am Gelenk angebracht. Zwischen den Hölzern befindet sich ein Plastikzahnrad mit einem Durchmesser von 3 cm. Das Zahnrad hat ein Loch in der Mitte und ist so angebracht, dass die vorher erwähnte Befestigungsschraube hindurchgehen kann. An einem Holz ist dieses Zahnrad fest durch zusätzliche Schrauben befestigt.

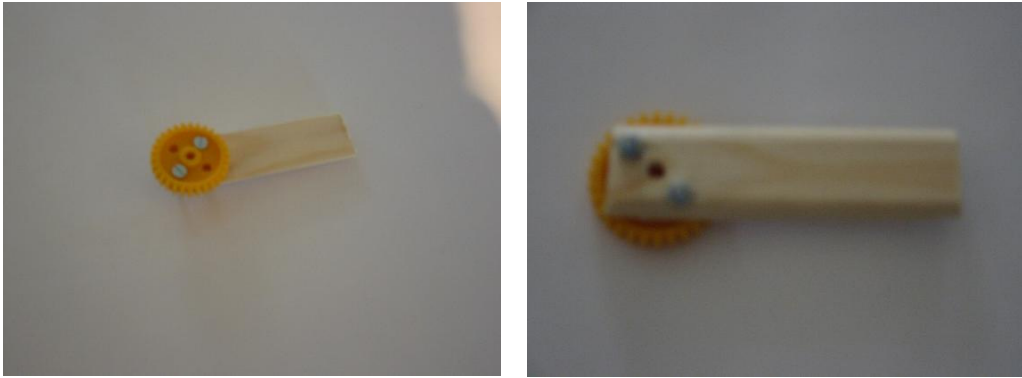


Abbildung 4.5: Der nichtsensorische Teil des Gelenks

Am anderen Holz ist in 2,5 cm Entfernung vom ersten Loch ein weiteres Loch (6mm Durchmesser) angebracht. Hierdurch geht ein runder Plastikstift (6mm Durchmesser und 15,7 mm oder 25,2 mm Länge), das an einem Ende fest mit einem Potentiometer verbunden ist und am anderen mit einem zweiten Zahnrad mit einem Durchmesser von 2 cm. So sitzen das Zahnrad und das Potentiometer eng anliegenden gegenüberliegenden Seiten des Brettchens. Das Potentiometer ist fest mit dem Brettchen verklebt. Zur besseren Haftung des Klebers und zusätzlicher Stabilisation des Potentiometers ist an diesem noch ein kleines Hölzchen (10mm x 21mm x 5mm) befestigt.

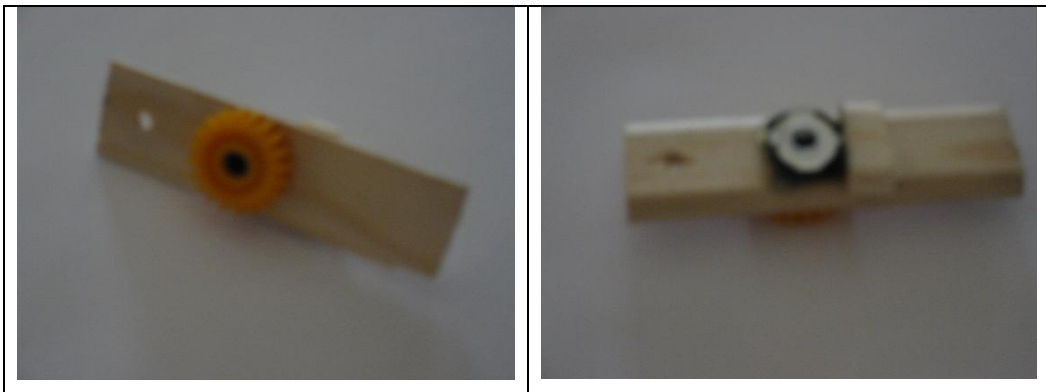


Abbildung 4.6: Der sensorische Teil des Gelenks mit Potentiometer

Beim Zusammenfügen der beiden Hölzer an der Befestigungsschraube gehen die Zähne der beiden Zahnräder lückenlos ineinander hinein. Da das eine Zahnrad fest mit einem Holz verbunden ist, verursacht eine Drehung des Gelenkes auch eine Drehung des zweiten Zahnrades und damit des Potentiometers.

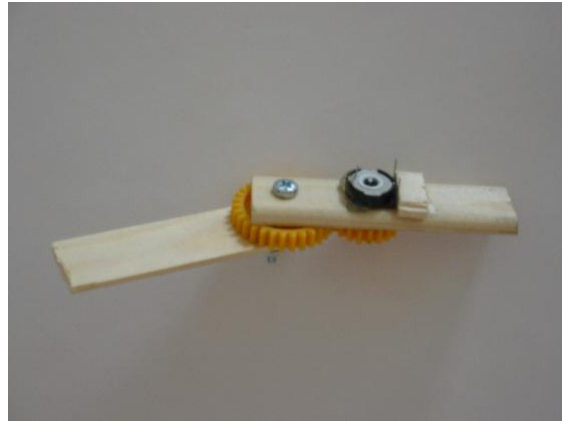


Abbildung 4.7: Das zusammengefügte Gelenk

Für die Befestigung der Gelenkschraube wurde eine Sicherheitsmutter verwendet. Am oberen Ende des Gewindes einer Sicherheitsmutter befindet sich ein Gummiüberzug, der verhindert, dass sich die Mutter durch die Drehung der Gelenke abschrauben kann. Zwischen der Mutter und dem Holz sind auch zwei Metallscheiben angebracht worden, die mit Vaseline geschmiert wurden. Dies verhindert zusätzlich eine Reibung der Mutter mit dem Holz und befestigt dadurch zusätzlich das Gelenk.

Gelenk	Position an der Puppe	Bewegungsebene
1	vorderer Rumpf	horizontal
2	rechtes Vorderbein (außen)	horizontal
3	rechtes Vorderbein (innen)	horizontal
4	Hals	vertikal
5	Kopf	horizontal
6	Kiefer	vertikal
7	linkes Vorderbein (innen)	horizontal
8	linkes Vorderbein (außen)	horizontal
9	hinterer Rumpf	vertikal
10	rechtes Hinterbein (außen)	horizontal
11	rechtes Hinterbein (innen)	horizontal
12	Schwanz (am Körper)	horizontal
13	Schwanz (Mitte)	horizontal
14	Schwanz (Spitze)	horizontal
15	linkes Hinterbein (innen)	horizontal
16	linkes Hinterbein (außen)	horizontal

Tabelle 4.1: Die Gelenke der lokationssensitiven Puppe

## 4.2.4 Die Signalübermittlung

Die Signalübermittlung basiert auf einem Spannungsabfall, welcher durch eine Potentiometerdrehung am Gelenk verursacht wird. Der Spannungsabfall wird über einen Analog-Multiplexer (Analog-Switch) an einen Analog-Digital-Wandler übertragen. Der digitale Wert aus dem A/D-Wandler wird an den Parallel-Port eines PCs angelegt und über einen Treiber ausgelesen. Die digitale Steuerung der Bauteile erfolgt ebenfalls über den Parallelport. Den Strom bezieht die Schaltung aus dem USB-Port des Computers.

### 4.2.4.1 Potentiometer

Da mit jedem Potentiometer die gesamte Spannung zum Abfall gebracht werden kann, ist es egal, wie groß der Maximal-Widerstand ist. Aus einigen Gründen muss man aber darauf achten, dass der Widerstand doch in einem bestimmten Bereich liegt.

Zum einen sollte der Widerstand nicht zu klein sein, damit der aufgenommene Strom nicht zu groß wird. Als Spannungsquelle wird der USB-Port benutzt. Dieser kann Maximal 500 mA bei 5V liefern. Da der A/D Wandler höchstens 3 mA verbraucht (Die Leckströme des A/D-Wandlers und des Analogmultiplexers werden vernachlässigt, da sie sich im  $\mu\text{A}$ -Bereich befinden), so kann der verbliebene Strom von 497 mA auf die 16 Potentiometer verteilt werden. Das bedeutet, dass durch jedes Potentiometer maximal 31,0625 mA fließen dürfen. Bei 5V führt das zu einem Widerstand von:

$$R_{\text{pot}} = \frac{5\text{V}}{31,0625\text{mA}} = 160,97\Omega \quad (4.1)$$

wobei  $R_{\text{pot}}$  der Maximal-Widerstand des Potentiometers ist. Dies ist die absolut unterste Grenze für diese Beschaltung.

Die obere Grenze wird durch den Leckstrom des A/D-Wandlers bestimmt. Bei einem Spannungsabfall an einem Spannungsteiler geht man davon aus, dass kein Verbraucher an die Schaltung angeschlossen ist. Dies ist aber nicht ganz korrekt. Durch einen A/D-Wandler fließt ein minimaler Strom. Diesen Strom nennt man Leckstrom und es

ist ein unerwünschter Strom, der durch Isolationen fließt. [LEM93] Er bewirkt, dass der Spannungsabfall nicht mehr korrekt ist. Dies kommt dadurch zustande, dass der Strom durch  $R_2$  nicht exakt dem Strom durch  $R_1$  entspricht (siehe Spannungsteiler, Gleichungen 2.1 und 2.2). Dieser Fehler muss möglichst klein gehalten werden. Ist der Maximalwiderstand des Potentiometers unter 1% des Widerstandes, den der A/D-Wandler bietet, so kann der Fehler vernachlässigt werden.

Der Leckstrom des A/D-Wandlers beträgt höchstens  $1\mu\text{A}$ . Dies ergibt einen Widerstand von  $5\text{M}\Omega$  bei  $5\text{V}$ . Da der Widerstand des Potentiometers nicht mehr als 1% betragen sollte, ist die obere Grenze  $50\text{k}\Omega$ .

Somit sollte ein Potentiometer gewählt werden, dessen Maximal-Widerstand nicht unter  $160,97\ \Omega$ , jedoch auch nicht über  $50000\ \Omega$  liegt. Hinzu kommt aber noch ein anderer Faktor.

Je größer der Maximal-Widerstand des Potentiometers, desto mehr neigt der momentan eingestellte Wert zu „flattern“. Das bedeutet, dass der Wert, der eigentlich konstant sein sollte, nicht konstant ist. Stellt man den Wert eines Potentiometers auf beispielsweise  $100\ \Omega$  ein, so wird der abgelesene Wert zwischen etwa  $99\ \Omega$  und  $101\ \Omega$  variieren.

Um nicht zu nah an der Minimalgrenze zu liegen, und auch nicht einen zu wechselhaften Widerstand zu bekommen wurde hier ein  $10\text{k}\Omega$  Widerstand benutzt. Er ist von der Firma Piher. Die Toleranz beträgt 20%. Das heißt, dass ihr tatsächlicher Maximal-Widerstand zwischen  $8\text{k}\Omega$  und  $12\text{k}\Omega$  liegt. Dies ist aber nicht schlimm, weil sich das auf die Berechnung überhaupt nicht auswirkt.



Abbildung 4.8: Das benutzte Potentiometer von Piher

Das flattern kann etwas unterdrückt werden, indem man zwischen Abgriff- und Masseanschluss des Potentiometers einen Kondensator zwischenschaltet. Dies ist aber hier nicht notwendig gewesen, da bei der Messung der Wert nicht um mehr, als 1 LSB des A/D-Wandlers variiert hat.

#### **4.2.4.2 Analog-Multiplexer**

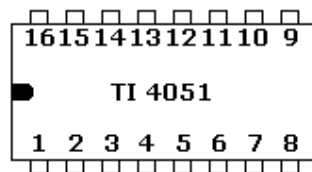
Ein Analog-Multiplexer (oder Analog-Switch) ist ein integrierter Schaltkreis, mit dem es möglich ist, aus einer Anzahl von Eingängen einen auszuwählen und an den Ausgang zu legen. Es handelt sich somit um ein Selektionsschaltnetz. Es besitzt mehrere Eingänge und je nach Ausführung einen oder mehrere Ausgänge. Es sind auch mehrere Steuereingänge vorhanden, mit denen es möglich ist auszuwählen, welcher Eingang zum Ausgang geschaltet wird.

Im Unterschied zum Digital-Multiplexer, welcher nur digitale Signale (also Low und High) durchleiten kann, arbeitet der Analog-Multiplexer mit analogen Signalen. Das heißt, dass er keinen definierten Spannungsbereich eines Low- oder High-Levels besitzt und theoretisch eine beliebige Spannung weiterleiten kann. Dieser Spannungsbereich ist meist aber doch durch die Versorgungsspannung beschränkt.

Der Analog-Multiplexer funktioniert nicht wie sein digitales Gegenstück mit Gattern, welcher je nach Ansteuerung und Eingangspegel ein High- oder Low-Signal an den Ausgang legt. Stattdessen sind es digital ansteuerbare Schalter, die eine echte elektrische Verbindung vom Ausgang zum Eingang herstellen. Deswegen kann er sowohl als Multiplexer, als auch als Demultiplexer benutzt werden. [LEM93]

Der hier verwendete Analog-Multiplexer ist der CD4051 von Texas-Instruments. Es ist ein 8:1 Multiplexer der Bausteingruppe TI-4051. Für die 16 Gelenke der Eingabepuppe werden 2 Stück gebraucht. Er besitzt 16 Pins. 8 Davon (Pin 1, 2, 4, 5, 12, 13, 14 und 15) sind die analogen Eingänge, von denen einer, je nach Belegung der digitalen Steuerports A, B und C (Pin 11, 10 und 9), mit dem Ausgang (Pin 3) verbunden ist. Zusätzlich gibt es noch einen digitalen Steuereingang Inhibit (Pin 6). Ist dieser auf High, so schaltet der Multiplexer keinen Eingang durch.

Die Stromversorgung erfolgt über die Anschlüsse  $V_{DD}$  (Pin 16),  $V_{SS}$  (Pin 8) und  $V_{EE}$  (Pin 7). Dabei ist die Differenz  $V_{DD} - V_{SS}$  der Bereich über den die Spannung am Eingang variieren kann und  $V_{DD} - V_{EE}$  die Differenz zwischen den Low- und High-Signalen. Beispielsweise ist bei  $V_{SS} = V_{EE} = 0V$  und  $V_{DD} = 5V$  der Bereich des analogen Signals 0V bis 5V, der Low-Pegel ist höchstens 1,5 V und der High-Pegel beträgt mindestens 3,5V.



Pin	Funktion
1	Eingang 4
2	Eingang 6
3	Ausgang
4	Eingang 7
5	Eingang 5
6	Steuereingang Inhibit
7	$V_{EE}$
8	$V_{SS}$
9	Steuereingang C
10	Steuereingang B
11	Steuereingang A
12	Eingang 3
13	Eingang 0
14	Eingang 1
15	Eingang 2
16	$V_{DD}$

Tabelle 4.2: Pinbelegung des TI-4051

Steuereingänge				Durchgeschalteter Eingang
Inhibit	A	B	C	
L	L	L	L	0
L	L	L	H	1
L	L	H	L	2
L	L	H	H	3
L	H	L	L	4
L	H	L	H	5
L	H	H	L	6
L	H	H	H	7
H	X	X	X	Keiner

**X - Don't Care**

**L - Low Signal**

**H - High Signal**

Tabelle 4.3: Steuereingang-Belegung für den TI-4051

[TIS06]

#### 4.2.4.3 Analog/Digital-Wandler

Der benutzte A/D Wandler ist ebenfalls von Texas-Instruments. Es ist der TLC 549. Das Wandlungsverfahren ist die Sukzessive Approximation und die Genauigkeit ist 8-bit. Der Wandler hat eine serielle Ausgabe. Beim Umwandeln wird eine angelegte Spannung mit zwei Referenzspannungen verglichen. Befindet sich die Spannung innerhalb der Referenzspannungen, so wird die Ausgabe ein digitaler Wert zwischen 0 und 255 sein. Ist die Eingangsspannung höher als die größere Referenzspannung, dann ist die Ausgabe stets 255 und umgekehrt ist sie 0, falls die Eingangsspannung niedriger ist als die kleinere Referenzspannung.

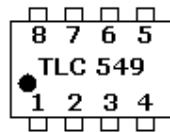


# TLC549



Abbildung 4.9: Der A/D-Wandler TLC549

Der Wandler besitzt 8 Pins. Pin 1 ist die größere Referenzspannung (Ref+), Pin 3 die kleinere (Ref-). Die zu messende Spannung wird an Pin 2 (Analog In) gelegt. Die Stromversorgung erfolgt über Pin 8 (VCC, Versorgungsspannung) und Pin 4 (GND, Masse). Der Wandler hat auch einen externen Taktgebereingang an Pin 7 (CLK) und einen „Chip Select“-Eingang an Pin 5 ( $\overline{CS}$ ). An Pin 6 (Data Out) werden nach der Berechnung die Daten vom A/D Wandler zur Verfügung gestellt.

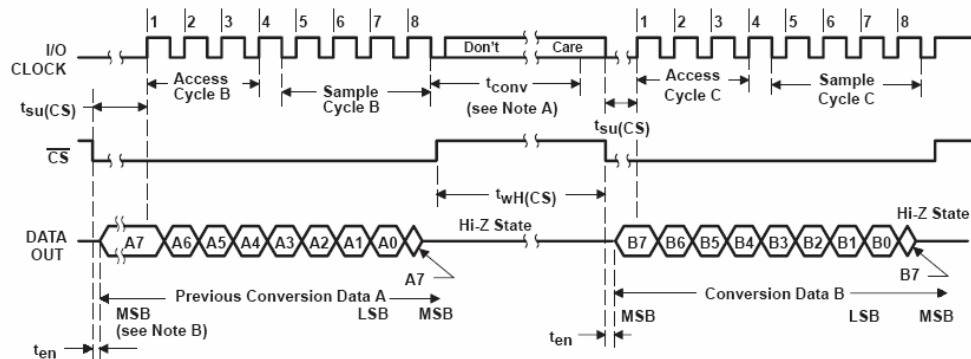


Pin	Funktion
1	Ref+
2	Analog In
3	Ref-
4	GND
5	$\overline{CS}$
6	Data Out
7	CLK
8	VCC

Tabelle 4.4: Pinbelegung des TLC549

Die Wandlung verläuft nach folgendem Schema: Nachdem die zu berechnende Spannung an Pin 2 angelegt wurde, müssen 8 Taktperioden mit einer Frequenz nicht höher als 1,1 MHz an Pin 7 gesendet werden. Während dieser Zeit muss  $\overline{CS}$  (Pin 5) auf Low sein. Jetzt fängt die Konvertierung des analogen Signals in ein digitales. Das  $\overline{CS}$  muss auf High gelegt werden. Nach 36 internen Taktzyklen (maximal  $17\mu s$ ) ist die Wandlung abgeschlossen und das höchstwertige Bit der Berechnung wird vom A/D-Wandler auf Pin 6 (Data Out) gelegt. Jetzt muss zum Lesen das  $\overline{CS}$  wieder auf

Low gesetzt werden und am Pin 7 (CLK) wieder der Takt angelegt werden. Mit jeder fallenden Taktflanke wird das jeweils nächste Bit auf Data (Pin 6) auslesbar. Hat man die 8 Bits herausgelesen, fängt die nächste Berechnung an. [TIA06]



- NOTES: A. The conversion cycle, which requires 36 internal system clock periods (17  $\mu$ s maximum), is initiated with the eighth I/O clock pulse trailing edge after CS goes low for the channel whose address exists in memory at the time.  
 B. The most significant bit (A7) is automatically placed on the DATA OUT bus after CS is brought low. The remaining seven bits (A6–A0) are clocked out on the first seven I/O clock falling edges. B7–B0 follows in the same manner.

Abbildung 4.10: Operationssequenz des A/D-Wandlers TLC 549

#### 4.2.4.4 Stromaufnahme und Spannungsversorgung

Die Potentiometer und die elektronischen Bauteile sind alle zueinander parallel geschaltet. Den Gesamtwiderstand einer Parallelschaltung berechnet man folgendermaßen:

	$\frac{1}{R_{\text{ges}}} = \frac{1}{R_1} + \frac{1}{R_2} + \dots + \frac{1}{R_n}$	(4.2)
--	--	-------

Dabei ist  $R_i$  der Widerstand eines einzelnen Verbrauchers und  $n$  die Anzahl der Verbraucher.

Die Widerstände der einzelnen Potentiometer sind:

Gelenk	Widerstand in $\Omega$
1	9830
2	9020
3	10230
4	10400
5	11360
6	8990
7	11550
8	9460
9	11400
10	8940
11	10140
12	10190
13	9550
14	11110
15	8500
16	9200

Tabelle 4.5: Widerstandswerte der benutzten Potentiometer

Der Gesamtwiderstand ist somit  $619,024 \Omega$ . Es ergibt sich bei  $5V$  ein Strom durch die Potentiometer von  $8,077 \text{ mA}$ . Der Verbraucherstrom des A/D-Wandlers ist  $3 \text{ mA}$  (die Leckströme des Wandlers und des Multiplexers werden hier vernachlässigt, da sie sich im  $\mu\text{A}$ -Bereich befinden). Dies führt zu einer benötigten Gesamtstromaufnahme von  $11,077 \text{ mA}$ .

Es ist theoretisch möglich, den Strom direkt aus dem Parallelport zu beziehen. Er könnte maximal  $24 \text{ mA}$  liefern bei einer Spannung von höchstens  $0,4 \text{ V}$  oder  $2,6 \text{ mA}$  bei mindestens  $2,4 \text{ V}$ . Abgesehen davon, dass der Parallelport nicht dazu gedacht wurde, Versorgungsspannung daraus zu beziehen, da er keine Stromanschlüsse besitzt, wäre das zu wenig. Man braucht  $5 \text{ V}$  (oder maximal  $10 \%$  weniger) und die

kann der Parallelport nicht mit Sicherheit liefern. Deswegen wurde hier ein separater Stromanschluss vom USB-Port benutzt. Er liefert 5V und maximal 500 mA. Am Testrechner wurde eine Spannung zwischen 4,79 V und 4,81 V gemessen. [EPP97]

#### 4.2.4.5 Parallelport

Der Parallelport eines üblichen PCs besitzt 25 Anschlusspins. 8 davon sind Masseleitungen. Die restlichen Pins sind in Datenpins (Data), Statuspins (Status) und Steuerpins (Control) unterteilt. Jede der drei Gruppen hat ein belegbares (Data) bzw. auslesbares (Status und Control) Register zum Steuern des Ausganges. Die genaue Belegung sieht wie folgt aus:

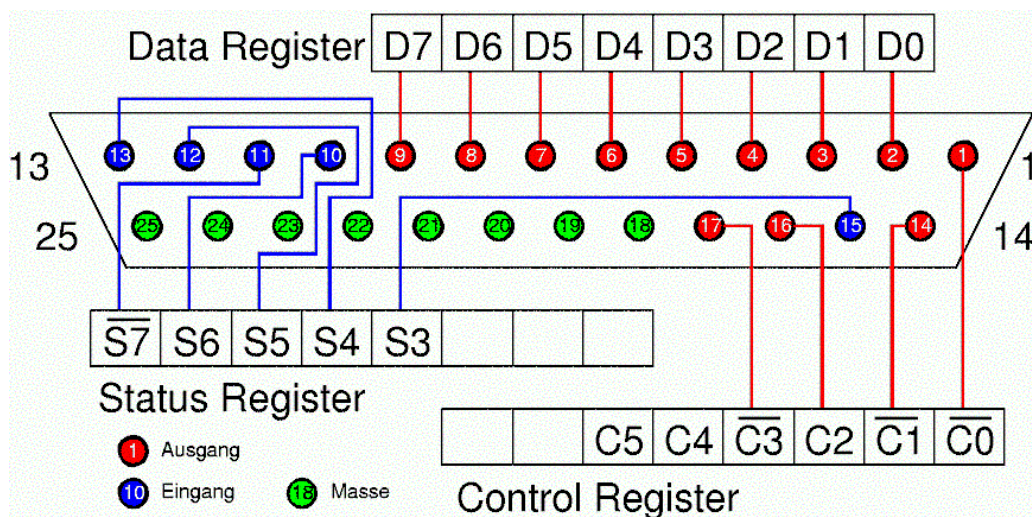


Abbildung 4.11: Pin-Belegung des Parallelports

Die Basisadresse des Parallelports ist üblicherweise 378 als Hexadezimalzahl, also 888 im Dezimalsystem. Über diese Adresse hat man Zugriff auf das Data-Register. Mit der Adresse 379 erreicht man das Status-Register und mit 37A das Control-Register.

Die Ansteuerung des Ports erfolgt mit den x86-Maschinenbefehlen <<IN Dest, Port>> bzw. <<OUT Port, Source>>. Bei der IN-Operation werden die Daten an der

Adresse Port ins Register Dest kopiert. Die OUT-Operation funktioniert genau umgekehrt. Es werden Daten aus dem Register Source auf die Adresse Port gelegt.

Seit Windows-NT/2000 und Windows-XP ist es nicht mehr möglich, diese Befehle aus einer beliebigen Anwendung aufzurufen. Denn bereits beim 286-Prozessor war es möglich, den Prozessor im so genannten „protected mode“ laufen zu lassen. Dies ermöglicht dem Prozessor primär auf einen viel größeren Speicherbereich zuzugreifen, als tatsächlich vorhanden ist, und erleichtert damit die Speicherverwaltung der Programme. Windows-NT/2000/XP unterstützen dies voll und ganz. Damit haben die Anwendungen einen „privilege level“. Er ist entweder Ring-0 oder Ring-3. Alle gewöhnlich programmierten Anwendungen laufen unter Ring-3 und sind damit im „User-Mode“. Sie haben unter Windows-NT/2000/XP keinen direkten Zugriff mehr auf die Portadressen, weil der Aufruf eines IN- oder OUT-Befehls eine Exception im Prozessor auslöst. Ring-0 Anwendungen, so genannte „Kernel-Anwendungen“, haben unbeschränkten Zugriff auf alle Ports und sind im „Kernel-Mode“.

Damit eine Anwendung das Zugriffsrecht Ring-0 bekommt, muss man sie unter diesen Betriebssystemen als „Kernel-Mode“-Anwendungen anmelden. Dies sind beispielsweise Systemtreiber. [PCH03][POR05]

#### 4.2.4.6 Die Kabel und die Platine

Eine Universal IC-Leiste dient als Platine zum Zusammenschalten der Analogmultiplexer, des A/D-Wandlers und der Leitungen vom Parallel- und USB-Port und der von der Puppe ankommenden Kabel. Die Abmessungen der Platine sind 60 mm x 160 mm x 1,5 mm und sie hat bereits vordefinierte Anschlussbereiche für IC-Chips.

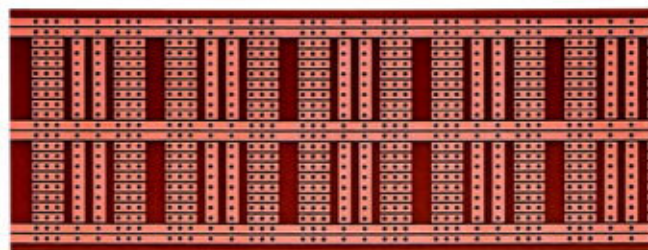


Abbildung 4.12: Die verwendete Universal IC-Leiste

Für den Parallelportanschluss wurde ein herkömmliches Druckerkabel benutzt. Das Ende mit der Centronics-Schnittstelle wurde abgeschnitten und die losen Einzeladern auf die Platine gelötet. Gleichermaßen wurde auch beim USB-Kabel vorgegangen. Es wurde nur die Seite mit dem Standard-A-Anschluss benutzt, welchen man auch an jeden USB-fähigen PC anschließen kann. Auf der anderen Seite wurde das Kabel aufgeschnitten und die Adern, die 5V und Masse führen, mit der Platine verlötet.

Für die Spannungsversorgung der Potentiometer auf der Puppe und für die Signalarückführung von den Potentiometern wurden Flachband-Telefonkabel benutzt. Je ein 8-Adriges Kabel wurde für die Potentiometer an den Gelenken 1-8 und den Gelenken 9-16 gebraucht. Die Versorgungsspannung wurde mit einem 4-adrigen Kabel von der Platine zur Puppe geführt, wovon aber nur zwei Adern (für 5V und Masse) tatsächlich gebraucht wurden. Diese 3 Verbindungskabel sind alle in der Mitte trennbar, da sie mit jeweils einer Western-Telefonkabelkupplung in der Mitte verbunden sind. An dieser Kupplung sind die Kabel ohne Probleme herausziehbar. Dies erleichtert den Transport und das Verpacken der gesamten Apparatur. Die Kabel sind beschriftet, um das spätere Verbinden zu erleichtern.

Auf der Puppe selbst führen dann beide Stromversorgungskabel von einem Potentiometer zum nächsten, an welche sie gelötet sind. Sie sind an verschiedenen Stellen an die Puppe verklebt um sie bei der Bedienung nicht herauszureisen. Die Abgriffe der Potentiometer sind mit je einer Leitung aus den 8-adrigen Kabeln verbunden und sind ebenfalls mit der Puppe verklebt.

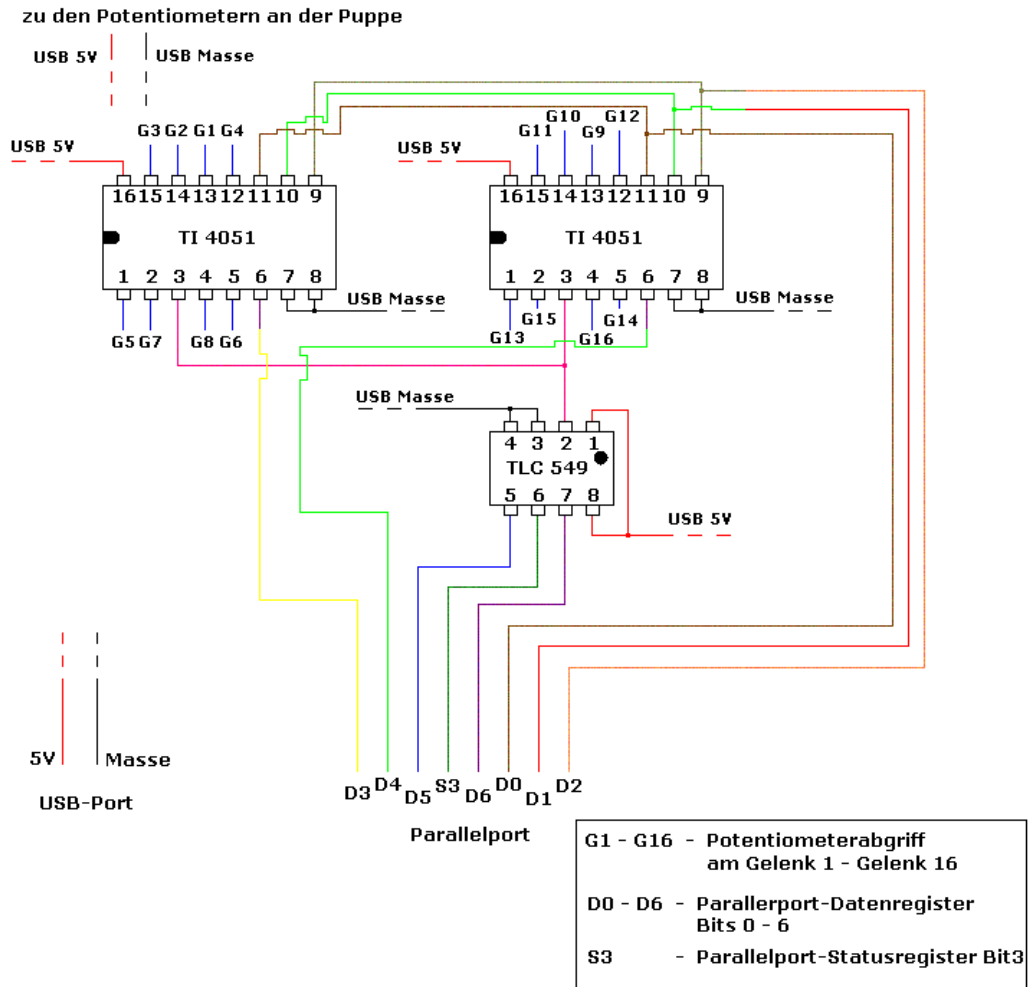


Abbildung 4.13: Das Schaltbild der benutzten Platine

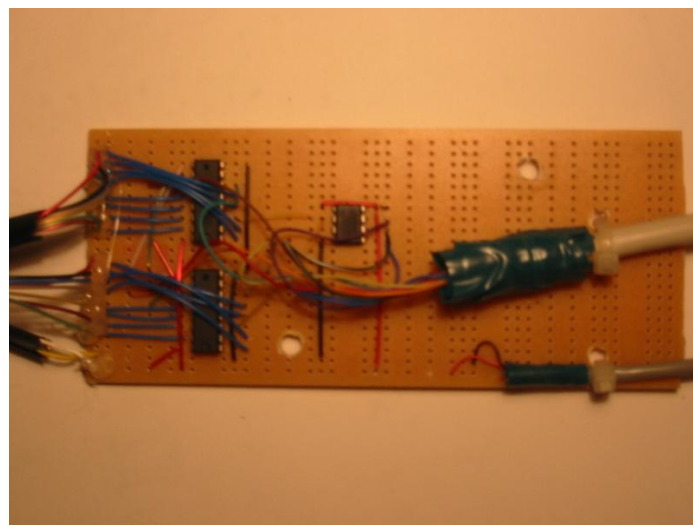


Abbildung 4.14: Die benutzte Platine

Auf dem Bild ist die Platine zu sehen. Die zwei größeren IC-Chips auf der linken Seite der Platine sind die TI-4051 Analog-Multiplexer. Der kleinere IC-Chip in der Mitte ist der TLC549 A/D-Wandler. Ganz links sind die Verbindungskabel zur Puppe zu sehen. Sie sind mit Heißkleber auf die Platine geklebt. Das dicke Kabel rechts ist das Kabel zum Parallelport. Das dünnere ist die Stromversorgung vom USB-Port. Die beiden Kabel sind mit Kabelbindern an der Platine befestigt.

Die beiden Löcher dienen der Befestigung der Platine an ein Schutzgehäuse. Sie wird mit einer Aluminiumplatte verschraubt. Oberhalb und unterhalb des Loches befinden sich Schutzringe aus Gummi. Die Befestigungsplatte wird dann an das Gehäuse gesteckt und verschraubt.

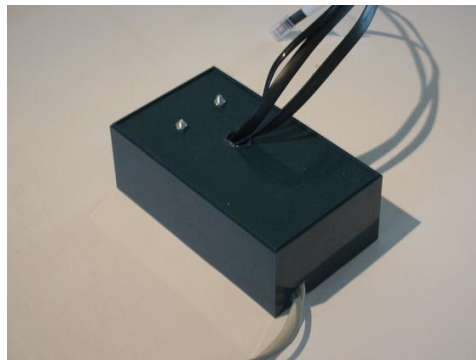


Abbildung 4.15: Das Gehäuse der Platine

## **4.3 Der Treiber**

### **4.3.1 Grundlagen**

Der Treiber ist als eine Windows-DLL (Dynamic Link Library) konzipiert. Er besteht aus zwei Teilen. Man könnte sie auch als zwei Schichten des Treibers betrachten. Die obere Schicht kommuniziert mit einer Anwendung, welche die Gelenkstellungen herauslesen möchte. Die untere Schicht ist in der Lage, auch unter Windows-NT/2000/XP-Systemen mit dem Parallelport zu kommunizieren.



Diese Port-Kommunikationsschicht ist ein für private Nutzungen frei zugänglicher Treiber. Es ist der Inpout32. Er besteht aus einer DLL-Datei, der Inpout32.dll, welche ins System-Verzeichnis von Windows kopiert wird. Dies ist meistens das „C:\Windows\System32“-Verzeichnis.

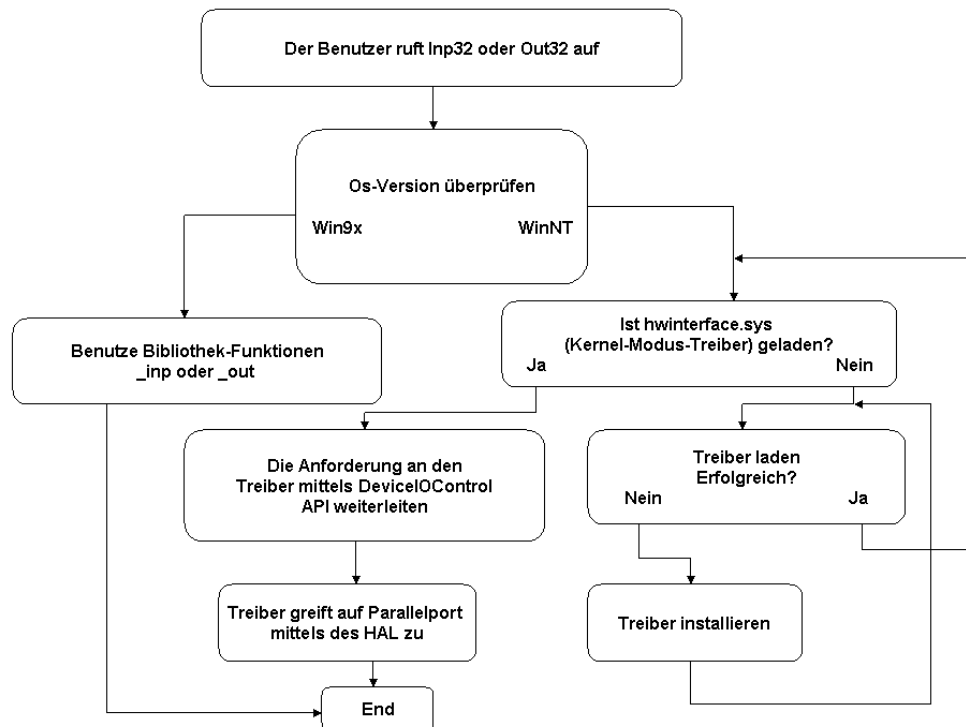


Abbildung 4.16: Flussdiagramm des Inpout-Treibers

Der Inpout32-Treiber bietet zwei sichtbare Funktionen, die Inp32 und die Out32. Diese ersetzen beim Programmieren seiner Anwendung die bekannten x86-Befehle in und out. Diese funktionieren nicht unter Windows-NT/2000/XP-Systemen. Der Treiber überprüft zuerst, ob das benutzte Betriebssystem Windows 95/98 oder Windows NT/2000/XP ist. Ist es Windows 9x, dann kann der Treiber ohne Probleme die Maschinenbefehle in und out benutzen, denn unter diesen Betriebssystemen gab es noch keine Einschränkungen. Ist es ein neueres Betriebssystem, dann wird zuerst überprüft, ob hwinterface.sys geladen ist, das ist der Kernel-Modus-Treiber, welcher die benötigten Rechte (Ring-0) besitzt, um auf den Port zuzugreifen. Falls er nicht

installiert oder geladen ist, dann wird er gegebenenfalls aus dem Binärcode des Inpout.dll extrahiert, ins „C:\Windows\System32\Drivers“-Verzeichnis kopiert und geladen. Falls er bereits geladen ist, dann werden an ihn die Anforderungen weitergeleitet und er ruft dann die benötigten HAL-Funktionen von Windows NT auf.

HAL steht für den „Hardware Abstraction Layer“ und bezeichnet eine mit Windows-NT eingeführte softwareseitige Hardware-Kommunikationsschicht. Diese dient dem Sperren von Hardwarekommunikation für nicht berechtigte Anwendungen.

Die Auswertung der gelesenen Daten und das Ansteuern der Bauelemente auf der Platine übernimmt die obere Schicht des Treibers. Sie wurde speziell für die vorher vorgestellte Platine geschrieben und heißt alphaDrv.dll. Diese Datei kann an einer beliebigen Stelle gespeichert sein. Wichtig ist, dass sie Zugriff auf die Inpout.dll hat. Dies ist gewährleistet, wenn sich die Inpout.dll im Systemverzeichnis befindet.

Der Treiber besteht aus 6 Funktionen:

Funktion	Beschreibung
DllMain( HANDLE hModule, DWORD ul_reason_for_call, LPVOID lpReserved )	<p>Diese Funktion wird sowohl aufgerufen, wenn die Dll mit der LoadLibrary()-Funktion aus der Windows-API aufgerufen wird, als auch wenn sie mittels FreeLibrary() verlassen wird.</p> <p>Sie lädt den Treiber Inpout32.dll und dessen Funktionen inp32() und out32(). Des Weiteren startet sie den Thread, welcher die Funktion run() ausführt.</p> <p>Beim verlassen der Dll wird der Inpout32-Treiber aus dem Speicher entfernt und der Thread beendet.</p>
getAngles(	Das ist die Schnittstelle des Treibers und die ein-

double * pAngArr)	zige Funktion auf die von außen zugegriffen werden kann. Sie kopiert die Daten aus dem internen Array aiAnglesArray, in dem die Winkelwerte gespeichert sind in das Array, dessen Adresse die Übergabevariable pAngArr ist.
computeAngles()	In dieser Funktion wird das Auslesen der A/D-Wandlers implementiert. Der Wert jedes einzelnen Gelenks wird nach dem Auslesen an die Funktion checkAndInsertValue() übergeben.
checkAndInsertValue( int index, short value)	Diese Funktion bekommt den aktuellen Wert value des Gelenks index und berechnet einen Mittelwert zusammen mit den letzten 5 ausgelesenen Werten. Des Weiteren wird hier auch die Hysterese implementiert. Der so ermittelte Wert wird im Array aiAnglesArrayBuff zwischengespeichert.
sleepMics( int mics)	Diese Funktion hält die Ausführung des Programms für mics µs an.
run(LPVOID Data)	Der in derDllMain() erstellte Thread ruft diese Funktion auf. Sie ruft in einer While-Schleife die Funktion computeAngles() auf, welche die Werte des A/D-Wandlers ermittelt. Aus dem Zwischenspeicher aiAnglesArrayBuffer werden die Werte in das aiAnglesArray kopiert. Die Kommunikation mit getAngles() wird mit den Variablen bThreadWorking und bValueRequest sichergestellt. Diese sind notwendig, damit getAngles nicht aus dem aiAnglesArray liest, während der Thread schreibt und umgekehrt.

Tabelle 4.6: Die Treiber-Funktionen

Aufgrund des Signalrauschens des A/D-Wandlers variiert das herausgelesene Signal manchmal um 1 LSB, obwohl die Gelenke nicht bewegt werden. Dies führt beim Herauslesen zu nichtkonstanten Werten, was beim Zeichnen die Figur an ihren Gelenken „flattern“ lässt, obwohl sie stillstehen müsste.

Die Funktion `checkAndInsertValue()` implementiert deswegen eine Mittelwertbildung und eine Hysterese, welche das variieren der Werte unterdrücken. Dies funktioniert folgendermaßen. Im Allgemeinen spricht man von Hysterese, falls eine Funktion einen anderen Verlauf bei steigenden, als bei fallenden Werten hat. Im unteren Beispiel beschränkt man sich auf einen Werte- und Definitionsbereich zwischen 0 und 2. Dies reicht aus, um die implementierte Hysterese zu verstehen.

Der A/D-Wandler liefert diskrete Werte. In diesem Beispiel werden die Werte 0 und 1 betrachtet. Ohne eine Hysterese würde eine Ausgabe des A/D-Wandlers direkt zu einem entsprechenden Winkel führen. Variiert aber bei stillstehenden Gelenken die Ausgabe des Wandlers um 1 LSB, so wird sich auch der herausgelesene Winkel ändern. Was nun gemacht wird ist folgendes. Ist die Ausgabe auf 1 so lässt man sie so lange auf 1, bis sie 0,5 oder 2 erreicht hat. Ist die Ausgabe auf 0, so steigt man nicht auf 1 hoch bis auch 1 erreicht wurde. Jetzt ergibt sich aber das folgende Problem. Ein A/D-Wandler kann keinen Wert zwischen 0 und 1 liefern, also auch nicht 0,5. Deswegen wird zuerst ein Mittelwert gebildet.

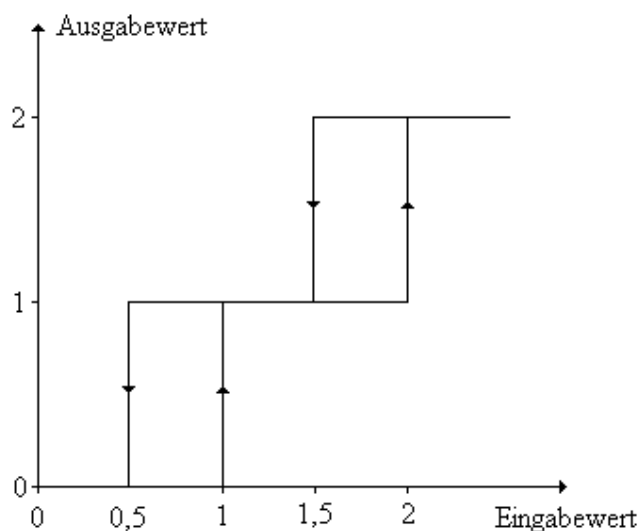


Abbildung 4.17: Hysteresekurve

Die Mittelwertbildung basiert auf der Speicherung der letzten 8 Werte, die vom A/D-Wandler berechnet wurden. Diese werden für jedes Gelenk in der Matrix

aaADValArrayHist [NR\_OF\_HISTORY][NR\_OF\_JOINTS]

gehalten. NR\_OF\_HISTORY ist eine Konstante, welche die Anzahl der zu benutzenden Werte für die Mittelwertbildung bestimmt. NR\_OF\_JOINTS bestimmt die Anzahl der zu überprüfenden Gelenke. Aus diesen 8 Werten wird der Mittelwert gebildet. Es wird geprüft, ob er sich im Bereich

$$(\text{Letzter Ausgabewert} - 0,5) \leq \text{Mittelwert} < \text{Letzter Ausgabewert} + 1$$

befindet. Ist er in diesem Bereich, so wird der alte Ausgabewert zum neuen Ausgabewert. Falls sich der Mittelwert aber außerhalb des Bereichs befindet, so erkennt man aus der Hystereseffunktion, dass sich dann auch die Ausgabe ändern muss. Damit entspricht die Vorgehensweise der oben gezeigten Hystereseffkurve.

Durch die Mittelwertbildung entsteht eine geringe Verzögerung der Ausgabe von höchstens 240 ms, diese fällt aber bei der Ausgabe nicht auf.

### 4.3.2 Schnittstellenspezifikation

Die Schnittstelle besteht aus der Dll-Datei AlphaDrv.dll und sie exportiert die folgende Funktion:

<b>void getAngles(double * pAngArr)</b>	
<i>Übergabevariablen</i>	
double * pAngArr	Pointer auf ein Array aus Double-Werten.  Das Array muss mindestens 16 Stellen besitzen. Diese werden von der Funktion mit Werten zwischen 0.0 und 180.0 gefüllt und entsprechen dem Winkel des ausgelesenen Gelenks.

Tabelle 4.7: Schnittstellenbeschreibung des Treibers

## 4.4 Die Anzeige

Für die Visualisierung der Puppenbewegungen wurde zusätzlich eine Anzeige-Software in Delphi geschrieben. Die Software liest die Gelenke mittels des Treibers aus und zeigt einen Blick auf die Puppe, von oben und von der Seite.

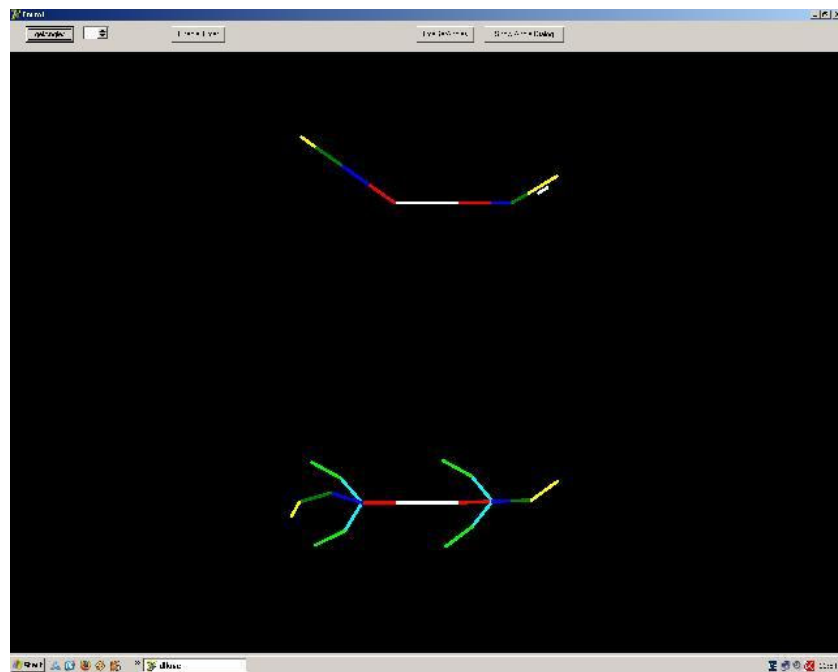
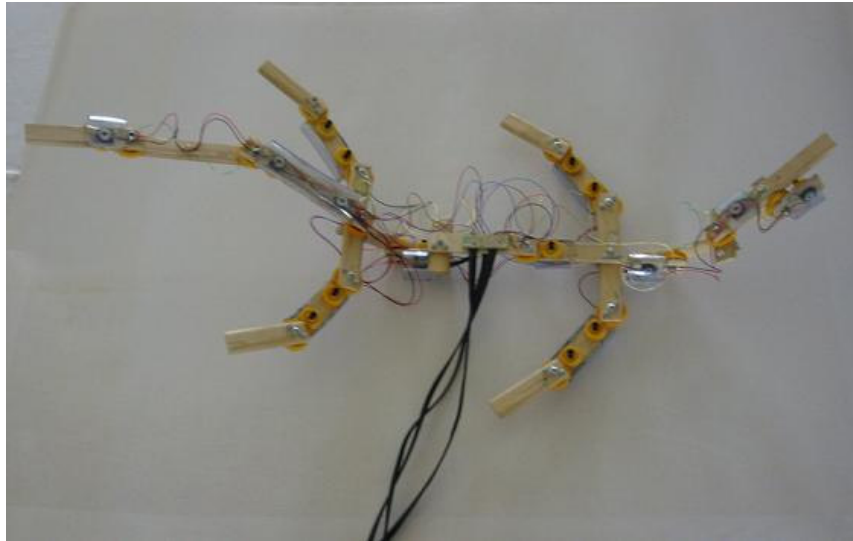


Abbildung 4.18: Die reale und die digitale Puppe

## 5. Evaluation des Konzepts

Das vorgestellte Animationssystem erfüllt vorrangig die Anforderungen universell und kostengünstig zu sein. Es wurde gezeigt, dass es möglich ist, ein System zu bauen, welches den Anforderungen eines Animateurs entspricht. Das System ist aber auch präzise genug, um damit wirkungsvolle Animationen durchzuführen. Die Universalität ist in der Hinsicht gegeben, dass die Animation sehr vieler verschiedener Formen und Objekte damit realisiert werden kann. Praktisch wäre es sogar möglich mit diesem System einen Datenhandschuh zu bauen.

Die bereits bestehende Platine kann für jede beliebige Form genutzt werden. Durch einige kleine Umbauten könnten auch mehr als 16 Gelenke eingelesen werden. Dazu muss ein (oder mehr) zusätzlicher 16:1 Analog-Multiplexer hinzugefügt werden. Momentan wird der Inhibit-Eingang der Multiplexer einzeln angesprochen und man braucht für die zwei Multiplexer auch zwei Leitungen. Dies kann auch verändert werden, indem ein Zusätzliches Bauteil die Signale aus dem Dezimalen-Code in einen „1 aus n“- Code, auch „One-Hot“ - Code genannt, umwandelt. Bei diesem Code ist immer nur eine Binär-Ziffer anders als alle anderen. Somit wird beispielsweise aus einer 3 eine 0100 und aus einer 4 eine 1000. Das würde es ermöglichen, mit zwei Leitungen bis zu 4 Multiplexer anzusprechen. Natürlich müsste man in diesem Fall die 1 mit der 0 ersetzen, weil die Multiplexer durch ein Low-Signal am Inhibit aktiviert werden. Ein solcher Chip ist der CY74FCT138AT von Texas-Instruments[TID06]. Der Treiber muss auch verändert und neu kompiliert werden, aber das nicht zeitaufwendig. Denn es muss nur eine neue Abfrage in die Funktion `computeAngles()` hinzugefügt werden, um einen neuen Analog-Multiplexer anzusteuern.

Ein gewisser Grad an Modularität ist erreicht worden, weil einzelne „Knochen“-Bauteile mehrmals verwendet werden oder sich stark ähneln. Dadurch ist es möglich, bestehende Gliedmaßen einfach zu erweitern, oder Gelenke zwischen zwei bereits bestehende einzufügen. Das Verfahren ist kein einfaches Einstecken im „Plug & Play“-System, jedoch pro Gelenk mit einem Aufwand, von nicht mehr als einer Stunde zu erledigen. Es muss ein passendes Brettchen erstellt und die dazugehörigen

Löcher gebohrt werden. Das große Zahnrad muss an ein Ende geschraubt und das kleinere muss präpariert und zusammen mit dem Potentiometer an das andere Ende eingesteckt werden. Jetzt muss das Potentiometer verklebt und verlötet werden. Dieses Verfahren ist bei jedem Gelenk identisch.

Dadurch, dass das Skelett aus Holz ist und leicht zu bearbeiten ist, ist es praktisch möglich jede Form daraus zu bauen. Obwohl sich die Gelenke an der vorgestellten Puppe nur in einer Ebene drehen lassen, so ist es kein Problem Kugelgelenke zu simulieren. Das wird erreicht indem man drei Gelenke in  $90^\circ$  zueinander verschobenen Ebenen nahe beieinander platziert und als ein Gelenk auswertet. Dies führt zu einem sehr universell einsetzbaren System.

Potentiometer sind eigentlich sehr unpräzise, da sie zum „Flattern“ neigen. Dazu kommt auch noch das Rauschen des A/D-Wandlers. Doch durch die benutzte Hysterese ist das System präzise genug, um damit sowohl einfache als auch aufwendige Animationen durchzuführen.

Der ausschlaggebende Vorteil ist der Preis. Die Bauteile dieser Puppe haben weniger als 90€ gekostet. Die Puppe ist dadurch deutlich billiger als alle anderen bestehenden Animationssysteme.

Weil das System für hydropneumatische Animationen benutzt werden soll, ist es auch realtime-fähig. Da der Treiber und die Platine alle Gelenke in einer Zeit von 30 Millisekunden auswerten, sind sogar bis zu 33,33 Frames/s für die Darstellung möglich.

Abschließend kann man zusammenfassen, dass das System die Anforderungen und Erwartungen erfüllt. Es ist universell, erweiterbar, präzise und kostengünstig.



## 6. Ausblick

Die vorgestellte Puppe kann in verschiedenen Gebieten Anwendung finden. Zum einen ist es mit ihr möglich jedes beliebige Objekt nachzubauen und zu animieren. Ein entsprechendes Programm kann dann über den Treiber die Winkelstellungen auslesen und das Skelett anzeigen. Die werte könnten aber auch für eine spätere Auswertung in einer Datei abgespeichert werden.

3D-Software Programme, wie das 3D-Studio-Max oder Maya unterstützen Motion-Capturing Dateien. In diesen Dateien ist eine komplette Aufzeichnung der Bewegung eines Objekts vorhanden. Ein Programm, das auf den Treiber zugreifen kann und die Bewegung aufzeichnet, könnte sich das zu Nutze machen und es dadurch möglich machen, das sich bewegende Objekt später nachträglich mit 3D-Software zu bearbeiten. 3D-Studio-Max unterstützt weiterhin Plug-Ins, welche auch direkt auf den Treiber zugreifen könnten.

Aufgrund der relativ kleinen der Auswertungszeit von 30ms ist es möglich real-time Wiedergaben der Bewegung zu zeigen. Die dadurch erreichbaren 33,33 Frames/s gestatten eine ruckelfreie Darstellung der Gelenkmanipulation. Möchte man eine VR-Umgebung mit diesem Systems bereichern, so ist dies notwendig. Eine Möglichkeit der Einbettung in VR- oder Mixed-Reality-Umgebungen, wäre es das System zu einem Datenhandschuh umzubauen. Eine andere wäre es ein mechanisches Motion-Capturing-System zu erstellen, welches die vorgestellten Komponenten direkt an die Gelenke befestigt, um so ihren gegenwärtigen Winkel zu messen. Dies wäre eine Art mechanisches Motion-Capturing.

Auch für hydropneumatische Animationen, könnte man die Apparatur nutzen. Körperaufbau und Bewegungsmuster von Lebewesen können dabei als hydropneumatische Vorgänge beschrieben werden. Einzelne Teile der Organismen unterliegen nach [DVO05] klassischen hydropneumatischen Gesetzen.

Werden die Glieder der Puppe an eine äußere Mechanik angeschlossen, welche eine hydropneumatische Bewegung an den Gelenken verursacht, so erreicht man eine Animation, welche sich den hydropneumatischen Gesetzen unterwirft. Diese Bewegung kann dann mittels der Puppe in Echtzeit angezeigt werden.

Weiterhin erklärt und implementiert [DVO05] eine neuartige 3D-Computer-Modellierungsweise, genannt „Quaoaring“-Technologie. Diese unterliegt mit Einklang zur Theorie ebenfalls der Hydropneumatik. Der entwickelte Algorithmus hält sich dabei strikt an die Theorie der zusammenhängenden pneumatischen Einheiten in einem Organismus und ermöglicht dadurch die Modellierung von biologischen Wesen, welche sich sehr realistisch verhalten.

Von den bei [DVO05] vorgestellten neun Einsatzgebieten, sind zwei auch für das Konzept der hier vorgestellten Puppe interessant. Zum einen kann „Quaoaring“ im Edutainment-Bereich benutzt werden. Die Entwicklung eines „Urmeeres“ ist geplant, welches ausgestorbene Tiere beherbergen soll. Ein Besucher kann in einem virtuellen U-Boot in dieses Meer hineintauchen und diese künstliche Welt bewundern. Die Puppe kann dazu dienen, einzelne Tiere interaktiv zu bewegen, um zusätzliche Unterhaltung zu bieten.

Aber auch im e-Learning-Bereich ist es möglich, die Puppe einzusetzen. Die „Quaoaring“-Technologie wird dabei computerdargestellte Einblicke in die Evolution und Entwicklung von Lebewesen ermöglichen und die lokationssensitive Puppe bietet die Möglichkeit diese Organismen zu bewegen.

Abschließend kann man zusammenfassen, dass das vorgestellte Animationssystem in vielen Bereichen der Animation benutzt werden kann und seinen Einsatz auch bei neu entwickelten computergesteuerten 3D-Modellierungs-Technologien findet.

## 7. Zusammenfassung

Die Diplomarbeit hat ihren Schwerpunkt bei der Entwicklung einer ALPHA, einer Architektur einer lokationssensitiven Puppe für hydropneumatische Animation. Es ist ein Animations-Eingabegerät, oder auch Eingabe-Puppe. Die Puppe soll ein Objekt repräsentieren, welches animiert werden soll. Ihre Gliedmaßen, welche aus Knochen und Gelenken bestehen, sind beweglich. Sie wird an einen Computer angeschlossen und die gegenwärtige Stellung ihrer Gelenke kann mittels eines ebenfalls entwickelten Treibers von diesem Computer eingelesen werden. Des Weiteren wird erklärt was eine hydropneumatische Animation ist.

Die Diplomarbeit weist die folgende Gliederung auf:

- Motivation
- Technische Grundlagen und State-Of-The-Art-Analyse
- Anforderungsanalyse
- Das eigene Konzept, die lokationssensitive Puppe (ALPHA)
- Evaluation
- Ausblick

Die Motivation beschäftigt sich mit der Entwicklung der Filmanimation und einiger Errungenschaften in der Laufbahn der Animationstechnologischen Entwicklung.

Die technischen Grundlagen beschränken sich auf die Funktionsweise von Messapparaturen, welche als Gelenkstellungssensoren fungieren können. Im Einzelnen sind das der optische Resolver, der Optoencoder und die Kombination von einem Drehpotentiometer und einem Analog/Digital-Wandler. Zur State-Of-The-Art-Analyse gehört die Erläuterung bereits entwickelter Stellungen- und Bewegungsmessender Technologien, wie das Stop-Motion-Verfahren, das Motion-Capturing, das Dinosaur-Input-Device und der Datenhandschuh. Eine Zusammenstellung der Defizite dieser Verfahren schließt dieses Kapitel ab.

Die Analyse der Anforderungen an ein zu entwickelndes System ist im Kapitel Anforderungsanalyse zu finden.

Zum eigenen Konzept gehört die gesamte Entwicklung einer lokationssensitiven Puppe. Die Puppe ist das Ebenbild des Skeletts eines zu animierendes Objekts. Sie besteht aus mehreren Gliedmaßen, welche an ihren Gelenken beweglich sind. Die Gelenkstellung wird von Potentiometern gemessen, dessen Signal ein A/D-Wandler empfängt. Die Umschaltung der einzelnen Messwerte erfolgt über Analog-Multiplexer. Die gesamte Steuerung der Bauteile und das Auslesen des A/D-Wandler werden durch einen Treiber über den Parallelport eines PCs gesteuert. Die Funktionsweise des Treibers und seine Implementierung werden ebenfalls in diesem Kapitel erläutert.

Im Kapitel Evaluation befindet sich eine Bewertung des Konzepts und der Erfüllung der Anforderungsanalyse.

Schließlich zeigt der Ausblick die Möglichkeiten der Anwendungen der Puppe und einen Blick in zukünftige Technologien.

## 8. Anhang

### A *Abbildungsverzeichnis*

Abbildung 1.1: Szene aus „King Kong und die weiße Frau“ von 1933.....	2
Abbildung 1.2: Szene aus ‚Star Wars‘ .....	3
Abbildung 1.3: Darstellung eines Froschs mit 3D Studio Max .....	4
Abbildung 2.1: Glasscheibe eines optischen Resolvers mit 3 Spuren .....	7
Abbildung 2.2: Ausschnitt der Drehscheibe eines Optoencoders .....	9
Abbildung 2.3: Signalveränderung beim Drehen der Glasscheibe .....	9
Abbildung 2.4: Schaltbild eines einfachen Spannungsteilers .....	11
Abbildung 2.5: Schaltbild eines Spannungsteilers mit n Stufen .....	11
Abbildung 2.6: Potentiometeranschlüsse .....	12
Abbildung 2.7: Schaltbild eines Potentiometers .....	13
Abbildung 2.8: Drehpotentiometer .....	14
Abbildung 2.9: Schiebepotentiometer.....	14
Abbildung 2.10: Darstellung der Quantisierung eines A/D Wandlers.....	15
Abbildung 2.11: Übertragungskennlinie des A/D-Wandlers .....	16
Abbildung 2.12: 2-bit A/D-Wandler mit Parallelverfahren (Blockschaltbild).....	18
Abbildung 2.13: 3-bit A/D-Wandler mit Wägeverfahren (Blockschaltbild) .....	20
Abbildung 2.14: Die Kamera des Eagle-Systems .....	28
Abbildung 2.15: Der Hub des Eagle-Systems.....	28
Abbildung 2.16: Die Darstellungssoftware des Eagle-Systems.....	29
Abbildung 2.17: Dinosaur Input Device für einen Velociraptor.....	31
Abbildung 2.18: Funktionsweise des Exoskeletts eines Dextrous Handmaster.....	34
Abbildung 4.1: Der Wasser-Dinosaurus Macroplata .....	42
Abbildung 4.2: Die entwickelte lokationssensitive Puppe.....	43
Abbildung 4.3: Das „L“-Verbindungsglied .....	44

Abbildung 4.4: Der Schutzschlauch.....	45
Abbildung 4.5: Der nichtsensorische Teil des Gelenks .....	46
Abbildung 4.6: Der sensorische Teil des Gelenks mit Potentiometer.....	46
Abbildung 4.7: Das zusammengefügte Gelenk.....	47
Abbildung 4.8: Das benutzte Potentiometer von Piher .....	49
Abbildung 4.9: Der A/D-Wandler TLC549 .....	53
Abbildung 4.10: Operationssequenz des A/D-Wandlers TLC 549.....	54
Abbildung 4.11: Pin-Belegung des Parallelports .....	56
Abbildung 4.12: Die verwendete Universal IC-Leiste .....	57
Abbildung 4.13: Das Schaltbild der benutzten Platine.....	59
Abbildung 4.14: Die benutzte Platine .....	59
Abbildung 4.15: Das Gehäuse der Platine .....	60
Abbildung 4.16: Flussdiagramm des Inpout-Treibers.....	61
Abbildung 4.17: Hysteresekurve.....	64
Abbildung 4.18: Die reale und die digitale Puppe .....	66

## ***B Tabellenverzeichnis***

Tabelle 2.1: Beispiel des Graycodes .....	8
Tabelle 4.1: Die Gelenke der lokationssensitiven Puppe .....	47
Tabelle 4.2: Pinbelegung des TI-4051 .....	51
Tabelle 4.3: Steuereingang-Belegung für den TI-4051 .....	52
Tabelle 4.4: Pinbelegung des TLC549 .....	53
Tabelle 4.5: Widerstandswerte der benutzten Potentiometer .....	55
Tabelle 4.6: Die Treiber-Funktionen .....	63
Tabelle 4.7: Schnittstellenbeschreibung des Treibers .....	65





## **C Literaturverzeichnis**

[ANI02] Lynn Pocock, Judson Rosebush: *The Computer Animator's Technical Handbook*, Morgan Kaufmann Publishers, 2002

[ART96] Isaac Victor Kerlow: *Computer Animation and Imaging*, John Wiley & Sons, 1996

[CHA98] Doug Kelly: *Character Animation In Depth*, The Coriolis Group Inc., 1998

[CPT05] Conrad Electronic, *Datenblatt PT 15 NV 10K* (Potentiometer), [http://www2.produktinfo.conrad.com/datenblaetter/425000-449999/432024-da-01-de-Einstellregler\\_PT\\_15\\_Nv.pdf](http://www2.produktinfo.conrad.com/datenblaetter/425000-449999/432024-da-01-de-Einstellregler_PT_15_Nv.pdf), Zugriff am 06.Januar 2005

[DID95] Brian Knep, Craig Hayes, Rick Sayre and Tom Williams: *Dinosaur Input Device*, CHI Paper, [http://www1.acm.org/sigs/sigchi/chi95/Electronic/documnts/papers/bk\\_bdy.htm](http://www1.acm.org/sigs/sigchi/chi95/Electronic/documnts/papers/bk_bdy.htm), 1995, Zugriff am 13.November 2005

[DIG01] Karen E. Goulekas: *Visual Effects in a Digital World*, Morgan Kaufmann, 2001

[DVO05] Tobias Breiner: *Dissertation über Dreidimensionale virtuelle Organismen*, Fachbereich Informatik und Mathematik, Johann Wolfgang Goethe-Universität in Frankfurt am Main, 2005

[ELE95] Reinhold Paul: *Elektronik und Elektrotechnik für Informatiker*, B.G. Teubner, 1995

[EPP97] Tomi Engdahl: *Powering Devices From PC Parallel Port* <http://www.epanorama.net/circuits/lptpower.html>, 1997, Zugriff am 05.Januar 2006

[EXP06] Kathleen Ruiz: *Experimental Motion Capture Systems Development*, <http://www.rpi.edu/~ruiz/research/research2/krmkmocap.htm>, Zugriff am 15. Januar 2006

[FDT06] *Preise für Trackingsysteme von 5DT*, <http://www.5dt.com/pricelist.html>, Zugriff am 15. Januar 2006

[GEL05] Horst Clausert, Gunther Wiesemann: *Grundgebiete der Elektrotechnik*, Oldenbourg Verlag, 2005

[ILM96] Mark Cotta Vaz, Patricia Rose Duignan: *Industrial Light & Magic - Into The Digital Realm*, Ballantine Books, 1996

[LEM93] Dieter Sautter, Hans Weinerth: *Lexikon Elektronik und Mikroelektronik*, VDI Verlag, 1993

[MEC03] Bodo Heimann, Wilfried Gerth, Karl Popp: *Mechatronik*, Carl Hanser Verlag, 2001/2003

[MOR93] Michelle Quinn: *Beyond The Valley Of The Morphs*, [http://www.wired.com/wired/archive/1.01/morphs\\_pr.html](http://www.wired.com/wired/archive/1.01/morphs_pr.html), 1993, Zugriff am 15. Januar 2006

[MOT06] Motion Analysis Corporation: *3D optical motion capture system Eagle*, <http://www.motionanalysis.com/applications/animation/film/eaglesystem.html>, Zugriff am 11. Februar 2006

[NOD06] *Preise für Trackingsysteme von noDNA*, [http://www.nodna.com/options\\_and\\_prices.231.0.html?&L=1](http://www.nodna.com/options_and_prices.231.0.html?&L=1), Zugriff am 15. Januar 2006

- [PAR99] Linux-Magazin: *Belegung des Parallellports*,  
[http://www.linux-magazin.de/Artikel/ausgabe/1999/10/IO/parport\\_b.gif](http://www.linux-magazin.de/Artikel/ausgabe/1999/10/IO/parport_b.gif),  
1999, Zugriff am 10. Februar 2006
- [PCH03] Hans Peter Messmer, Klaus Dembowski: *PC-Hardwarebuch*, Addison-Wesley, 2003
- [PCN02] Robert Bruce Thompson, Barbara Fritchman Thompson: *PC-Hardware in a Nutshell*, O'Reilly, 2002
- [POR05] *PortTalk - A Windows NT I/O Port Device Driver*  
<http://www.beyondlogic.org/porttalk/porttalk.htm>, Zugriff am 10. Februar 2005
- [RTD05] Toni Nemeč: *Diplomarbeit über die Entwicklung eines Echtzeit Raytracers*  
[http://www.gdv.informatik.uni-frankfurt.de/diplomarbeiten/pdf/Diplomarbeiten\\_final/Diplomarbeit\\_tnemeč.pdf](http://www.gdv.informatik.uni-frankfurt.de/diplomarbeiten/pdf/Diplomarbeiten_final/Diplomarbeit_tnemeč.pdf),  
2005
- [TIA06] Texas Instruments, *Datenblatt A/D Wandler TLC598*,  
<http://focus.ti.com/lit/ds/symlink/tlc549.pdf>, Zugriff am 14. Januar 2006
- [TIS06] Texas Instruments, *Datenblatt Analog Switch TI4051*,  
<http://focus.ti.com/lit/ds/symlink/cd4051b-q1.pdf>, Zugriff am 14. Januar 2006
- [TID06] Texas Instruments, *Datenblatt, „1 aus 8“ – Decoder*,  
<http://focus.ti.com/lit/ds/symlink/cy74fct138t.pdf>, Zugriff am 14. Januar 2006
- [VRE94] Sven Bormann: *Virtuelle Realität*, Addison Wesley Publishing Company, 1994
- [VRP95] Dimitris N. Chorafas, Heinrich Steinmann: *Virtual Reality*, Prentice Hall, 1995

[VRT94] Grigore Burdea, Philippe Coiffet: *Virtual Reality Technology*, John Wiley & Sons, 1994