



Universität  
Siegen

FB 12: Elektrotechnik und Informatik

FG: Echtzeit Lernsysteme

Prof. Dr.-Ing. Kuhnert

---

# Einführung in die Hardware des LEGO Mindstorms

# Inhaltsverzeichnis

1 Einleitung.....	3
2 RCX.....	4
2.1 Innerer Aufbau.....	4
2.1.1 Zeitverwaltung.....	4
2.1.2 Speicheraufbau.....	5
2.1.3 A/D-Wandler.....	5
2.2 Benutzerinteraktion.....	6
2.2.1 IR-Schnittstelle.....	6
2.2.2 Bedienelemente.....	7
2.2.3 LCD-Display.....	7
2.3 Ausgänge.....	9
2.3.1 Motor.....	9
2.3.1.1 Pulsweitenmodulation.....	9
2.3.1.2 Anschließen der Motoren.....	10
2.4 Sensoren.....	11
2.4.1 Passive Sensoren.....	11
2.4.2 Aktive Sensoren.....	12
2.4.3 Sensormodus.....	12
2.4.3.1 Rohmodus.....	12
2.4.3.2 Logischer Modus.....	12
2.4.3.3 Flankenzählungsmodus.....	13
2.4.3.4 Impulszählungsmodus.....	13
2.4.3.5 Prozentmodus .....	14
2.4.3.6 Rotationsmodus.....	14
2.4.4 Sensortypen.....	14
2.4.4.1 Berührungssensor.....	14
2.4.4.2 Lichtsensor.....	14
2.4.4.3 Rotationssensor.....	15
2.4.5 Anschließen der Sensoren.....	15
2.5 Differential.....	16

## 1 Einleitung

In diesem Umdruck erfolgt die Einführung in die Hardware des RCX-Bausteins (engl.: Robotic Command Explorer), kurz genannt RCX. Dazu gehört nicht nur der interne sowie der externe Aufbau des RCX, sondern auch die Aktoren und Sensoren. Die Aktoren und Sensoren können bequem von außen über ein Verbindungskabel mit speziellen Kontaktplatten am RCX angeschlossen werden. Zusätzlich wird noch ein mechanisches Bauteil, nämlich das Differential vorgestellt. Dieses Bauteil nimmt eine wichtige Funktion für das geradeaus Fahren des Roboters ein.



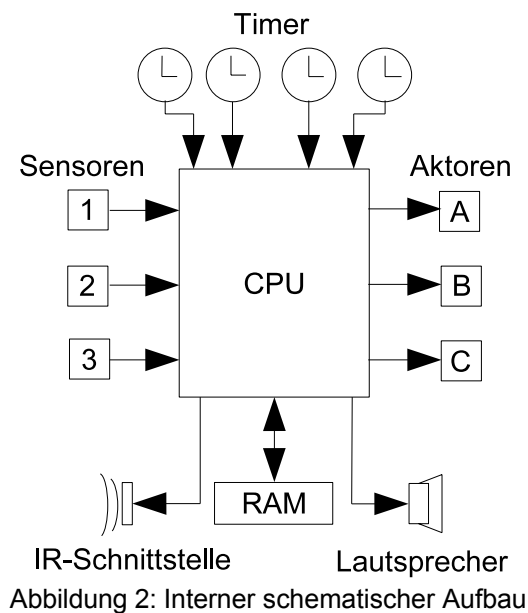
Abbildung 1: RCX mit angeschlossenen Sensoren und Motoren [MIN-04]

Zudem sollte noch erwähnt werden, dass der RCX ein so genanntes „embedded System“ ist. Deshalb werden die Programme für den RCX auf dem Personal Computer (PC) entwickelt („cross development“) und dann via Infrarot-Link zum RCX übertragen (engl.: downloaden).

## 2 RCX

Der RCX selbst basiert auf einem Mikroprozessor aus der Hitachi H8-Reihe. Dieser Mikroprozessor ist ein 8-Bit-Prozessor mit 16MHz. Er stellt dem Anwender eine serielle Ein-/Ausgabe, vier Timer und mehrere Analog/Digital- bzw. Digital/Analog-Wandler für diverse Steuerungsaufgaben zur Verfügung. Des Weiteren beinhaltet er einen 16KB großen internen ROM-Speicher, sowie einen 32KB großen statischen RAM-Speicher.

Der RCX verfügt über drei Eingänge für den Anschluss von Sensoren und drei Ausgänge für den Anschluss der Aktoren, sowie über einen Lautsprecher. Hinzu kommt noch eine Infrarot-Schnittstelle, um mit dem PC oder einem anderen RCX zu kommunizieren. Das LCD-Display und die vier Tasten dienen zur Benutzerinteraktion.



Die Spannungsversorgung des RCX erfolgt entweder über 6 AA-Batterien oder über einen Netzadapter. Um die Batterien zu schonen, schaltet sich der RCX nach einigen Minuten Inaktivität automatisch ab. Zwar steht diese Zeitspanne (Abschaltzeit) anfangs auf 15 Minuten, sollte aber durch die Programmiersoftware (z. B. Bricx Command Center) geändert werden, da er sich sonst auch während einer Programmausführung abschaltet.

### 2.1 Innerer Aufbau

Zu den wichtigsten internen Komponenten des RCX zählen die vier Timer, der RAM- und ROM-Speicher sowie die A/D-Wandler. Deshalb erfolgt in diesem Abschnitt eine genauere Vorstellung dieser Komponenten. Es wurde dabei ein besonderes Augenmerk auf die Ermittlung der Kennwerte des A/D-Wandlers gesetzt, da diese maßgebend für die Umformung eines Messwertes in einem für den RCX geeigneten, zu verarbeiteten Wert sind.

#### 2.1.1 Zeitverwaltung

Die vier Timer des RCX messen in Schritten von 100ms. Sie sind nur aktiv, wenn der RCX eingeschaltet ist und können jederzeit unabhängig voneinander abgefragt und zurückgesetzt werden. Ansonsten fangen sie nach ungefähr 55 Minuten wieder bei 0 an.

Zusätzlich zu den vier Timern besitzt der RCX eine Systemuhr, die sich die Anzahl der Minuten nach dem Einschalten merkt. Im Standardanzeigenmodus zeigt das LCD-Display die aktuelle Uhrzeit in Stunden und Minuten an.

### 2.1.2 Speicheraufbau

Der 16KB große interne ROM-Speicher ist mit einer rudimentären Betriebssystemsoftware (BIOS) vorprogrammiert. Der 32KB große statische RAM-Speicher wird größtenteils von der Firmware und diversen Systemparametern belegt. In diesem RAM-Speicher ist für den Benutzer ein 6KB großer „Benutzerspeicher“ reserviert. Dieser wiederum beinhaltet 5 „programm slots“, in denen maximal 5 Programme parallel abgespeichert werden können - also pro Slot ein Programm. Es kann jeweils nur ein Programm aktiv sein. Die jeweilige Slot-Nummer des aktiven Programms wird im LCD-Display angezeigt. Dadurch ergibt sich ein Speicheraufbau wie in Abbildung 3 gezeigt wird.

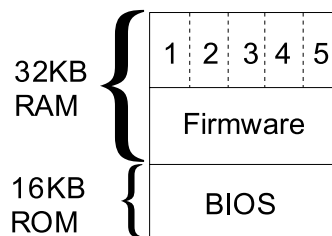


Abbildung 3: Speicheraufbau

### 2.1.3 A/D-Wandler

Ein Analog/Digital-Wandler (kurz: A/D-Wandler) oder auch A/D-Umsetzer genannt, wandelt ein analoges Signal in ein entsprechendes digitales Signal um. Bei den A/D-Wandlern im RCX handelt es sich jeweils um einen 10Bit-A/D-Wandler. Das bedeutet, dass der A/D-Wandler eine Auflösung von 10Bit besitzt. Daraus ergeben sich 1024 Spannungsstufen (Quantisierungseinheiten) für den 10Bit-A/D-Wandler nach Formel 1, wobei  $n$  die Anzahl der Bits (Auflösung) angibt.

$$N_{max} = 2^n = 2^{10} = 1024 \quad (1)$$

Ein Zähler liefert das Ausgangssignal des A/D-Wandlers. Dessen maximaler Zählerstand wird nach der Formel 2 berechnet.

$$Z_{max} = N_{max} - 1 = 1024 - 1 = 1023 \quad (2)$$

Daraus ergibt sich die folgende Zuordnung für den Anfangswert und den Bereichsendwert  $U_E$ :

- 0V  $\Rightarrow$  0
- 5V  $\Rightarrow$  1023

Eine Quantisierungseinheit bzw. eine Spannungsstufe des A/D-Wandlers beträgt nach folgender Formel 4,88mV.

$$\Delta U = U_{LSB} = \frac{U_E}{2^n} = \frac{5V}{2^{10}} = 4,88 \text{ mV} \quad (3)$$

Mit Hilfe dieser berechneten Werte wird die Quantisierung der Messspannung bzw. des Sensorwertes vorgenommen. Die Abbildung 4 zeigt das Prinzip der Quantisierung der Messspannung  $U_x$  für  $n=3$ .

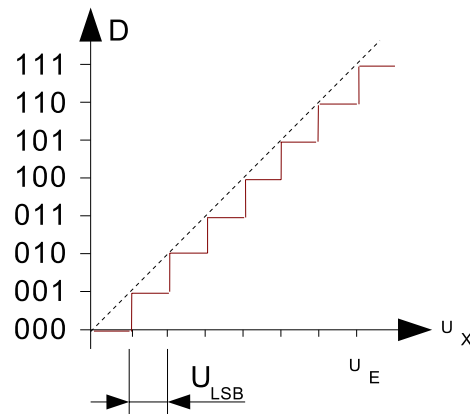


Abbildung 4: Prinzip der Quantisierung einer Messspannung für  $n=3$

## 2.2 Benutzerinteraktion

Auf der Oberseite des RCX sind vier Tasten sowie ein LCD-Display für die Benutzerinteraktion angeordnet. Damit kann der RCX nicht nur ein- und ausgeschaltet werden, sondern hier werden Programme ausgewählt und gestartet, und die Zustände der angeschlossenen Aktoren und Sensoren abgefragt. Zudem ist an der Vorderseite des RCX die IR-Schnittstelle für die Kommunikation mit einem PC oder einem anderen RCX angeordnet.

### 2.2.1 IR-Schnittstelle

Die Verbindung zwischen PC und RCX wird per Infrarot-Link (IR-Link) realisiert (siehe Abbildung 5). Am PC wird dazu der IR-Tower über den 9-poligen, seriellen Standardanschluss angeschlossen. Die Datenübertragungsrates beträgt 2400/4800 Baud. Die Reichweite des IR-Signals vom IR-Tower kann über den Schalter am IR-Tower eingestellt werden. Bei der Schalterstellung „Low“ hat das IR-Signal eine Reichweite von circa 60cm. Diese Einstellung eignet sich für die Programmübertragung. Im Gegensatz dazu, erreicht das IR-Signal bei der Schalterstellung „High“ eine Reichweite von circa 8m und somit eignet sich diese Einstellung für den Datenaustausch.

Aufgrund der Infrarotschnittstelle des RCX ergeben sich die drei folgenden Möglichkeiten bezüglich der Datenübertragungsrichtung:

1. PC  $\Rightarrow$  RCX
2. RCX  $\Rightarrow$  PC
3. RCX  $\Rightarrow$  RCX

Bei der 1. Möglichkeit kann die Firmware, oder ein eigenes Programm sowie einzelne Kommandos übertragen werden. Bei der 2. Möglichkeit können Werte vom Datentyp Integer, sowie Debugginginformationen übertragen werden. Dagegen können bei der 3. Möglichkeit nur Werte vom Datentyp Integer ausgetauscht werden.



Abbildung 5: IR-Link [MIN-04]

### 2.2.2 Bedienelemente

Der RCX besitzt vier Tasten für die Benutzerinteraktion. Diese vier Tasten sind die Steuertasten des RCX, deren Anordnung die Abbildung 6 zeigt. Mit Hilfe dieser vier Steuertasten kann der RCX gesteuert werden.

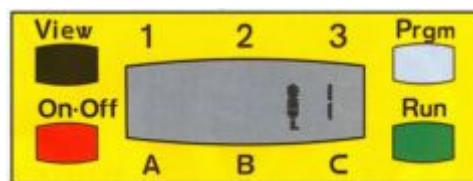


Abbildung 6: Bildausschnitt vom RCX [ART-04]

Die genaue Kennzeichnung, Funktion und Beschreibung der vier Steuertasten wird in der Tabelle 1 näher erläutert. Diese Tabelle gilt nur für die LEGO-Firmware und damit nur für Programmiersprachen, die auf dieser Firmware basieren.

<b>Kennzeichnung</b>	<b>Farbe</b>	<b>Bezeichnung</b>	<b>Funktion</b>	<b>Beschreibung</b>
On-Off	Rot	Ein-/Ausschalt-taste	Schaltet den RCX ein und aus	Die Programme bleiben im RCX gespeichert, solange sich die Batterien im RCX befinden
View	Schwarz	Ansicht	Ermöglicht dem Anwender, einen Ein- bzw. Ausgang zur Ansicht im Display auszuwählen	Ein Pfeil erscheint neben dem Ein- bzw. Ausgang, der im Display angezeigt wird
Prgm	Grau	Programm	Wählt ein bestimmtes Programm aus (1-5), das auf dem RCX ablaufen soll	Durch wiederholtes drücken der Taste wird die angezeigte Zahl jeweils um eins erhöht
Run	Grün	Start-/Stoptaste	Startet und beendet den Ablauf eines Programms auf dem RCX	Während das Programm abläuft, scheint die „kleine Person“ zu laufen

Tabelle 1: Tastenbelegung des RCX [GRO-98]

### 2.2.3 LCD-Display

Das LCD-Display dient zusammen mit den vier Tasten der Benutzerinteraktion. Es stellt nützliche Informationen wie Sensordaten und Konfiguration der Ausgänge zur Verfügung. Bei Auswahl eines Sensors können die Sensorwerte bis zu einer vierstelligen Zahl mit Vorzeichen angezeigt werden. Wenn kein Sensor ausgewählt wurde, wird mit der vierstelligen Zahl der aktuelle Wert der Systemuhr angezeigt. Die Programmnummer, die rechts im Display angeordnet ist, wird als einstellige Zahl angezeigt. In der Tabelle 2, für die das

gleiche bezüglich der LEGO-Firmware gilt wie für die Tabelle 1, wird die Bedeutung der einzelnen Symbole vom LCD-Display, das in der Abbildung 7 zu sehen ist, beschrieben.

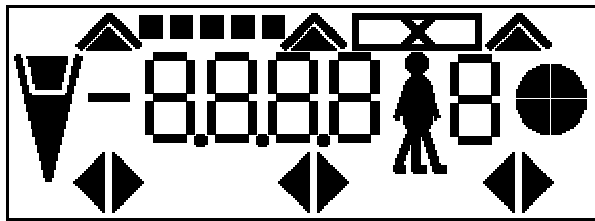


Abbildung 7: LCD-Display mit Symbolen [FHM-03]












Symbol	<i>Beschreibung</i>
	Batterieladung niedrig, ab 6,7V blinkt das Symbol
	RCX empfängt Daten vom IR-Link (Segmente laufen durch)
	Eingabeport ist aktiv
	Selektierter Sensoreingang (aktueller Wert wird angezeigt)
	IR-Link sendet oder empfängt Daten mit Empfindlichkeit „niedrig“
	IR-Link sendet oder empfängt Daten mit Empfindlichkeit „hoch“
	Sensorwert, Software-Uhr, etc.
	Zeigt an, dass der RCX eingeschaltet ist. „kleine Person“ beginnt beim Programmstart zu laufen
	Programmnummer 1-5
	Füllstand des Datalogs (Datenaufzeichnung), ¼ pro Segment
	Ausgabeport ist aktiv, Pfeilrichtung gibt die Drehrichtung des Motors an

Tabelle 2: Erklärung der Symbole des LCD-Displays [FHM-03]



## 2.3 Ausgänge

Der RCX besitzt die drei Ausgänge A, B und C mit jeweils einer Ausgangsspannung von 9V. An diesen Ausgängen kann als Aktor z. B. ein Getriebemotor oder eine Lampe angeschlossen werden.

### 2.3.1 Motor

Der Motor, der dem RIS-Baukasten beiliegt, ist ein 9V-Getriebemotor. Dieser besitzt ein internes Getriebe und ein Schwungrad. Er kann in den drei folgenden Modi betrieben werden: „ein“, „aus“ und „gleitend“. Die drei verschiedenen Modi werden anhand des Beispiels „angeschlossener Motor am Ausgang“ erklärt. Der Modus „ein“ bedeutet, dass der Motor eingeschaltet wird und somit läuft. Im Gegensatz dazu wird der Motor bei dem Modus „aus“ ausgeschaltet, also gestoppt. Beim Modus „gleitend“ wird der Motor nicht mehr mit Strom versorgt, kann sich aber noch frei drehen. Dieser Zustand entspricht dem Rollen eines Autos im Leerlauf.

Zusätzlich zu den drei Modi kann die Drehrichtung (im oder entgegen dem Uhrzeigersinn) des Motors eingestellt werden. Dies wird auch als Vorwärts- bzw. Rückwärtslauf bezeichnet.

Des Weiteren kann der Motor in verschiedenen Leistungsstufen betrieben werden. Es gibt acht Leistungsstufen (Ls) von 0-7. Da der RCX digitale Ausgänge besitzt, wird die Leistungssteuerung mittels der Pulsweitenmodulation (PWM), die oft auch als Pulsbreitenmodulation bezeichnet wird, durchgeführt. Die Pulsweitenmodulation funktioniert nur, weil die Getriebemotoren mit einem Schwungrad (Schwungmasse) ausgestattet sind. Dabei wirken sich die einzelnen Leistungsstufen nur unter Last aus.



Abbildung 8: 9V Getriebemotor [PHI-04]

#### 2.3.1.1 Pulsweitenmodulation

Am einfachsten lässt sich die Funktionsweise der Pulsweitenmodulation anhand eines Komparators erklären. An den beiden Eingängen des Komparators wird einmal eine Dreiecksspannung und zum anderen die Modulationsspannung PW angeschlossen. Der Komparator vergleicht die beiden Eingangsspannungen und gibt je nach Höhe der Modulationsspannung ein Ausgangssignal heraus, das proportional zur Modulationsspannung ist.

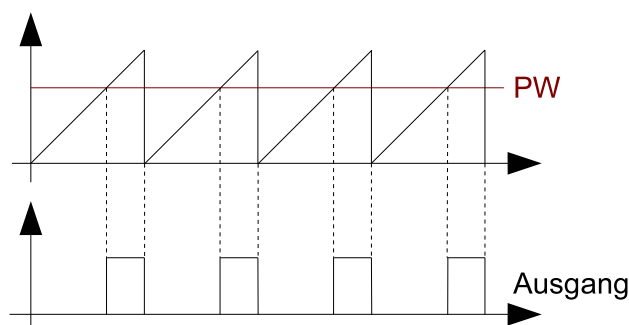


Abbildung 9: Kennlinien der Pulsweitenmodulation

Die Höhe der Modulationsspannung PW bestimmt die Impulsdauer des Ausgangssignals und somit den Tastgrad (siehe Formel 4).

$$\text{Tastgrad} = \frac{\text{Impulsdauer}}{\text{Periodendauer}} \quad (4)$$

Währenddessen ist die Frequenz und somit die Periodendauer konstant. Die Periodendauer beträgt beim RCX 8ms. Bei der niedrigsten Leistungsstufe also „Ls: 0“, beträgt die Impulsdauer 1ms. Da der Ausgang nur ein achtel der Zeit (Periodendauer) mit Spannung versorgt wird, ergibt sich daraus ein Tastgrad von 12,5%. Dagegen ergibt sich bei der höchsten Leistungsstufe also „Ls: 7“ einen Tastgrad von 100% (siehe Abbildung 10).

Zusammenfassend kann gesagt werden: Je größer die ausgewählte Leistungsstufe ist, desto länger der Impuls und um so größer der Tastgrad. Wobei pro Leistungsstufe eine Veränderung des Tastgrads um 12,5% vorgenommen wird.

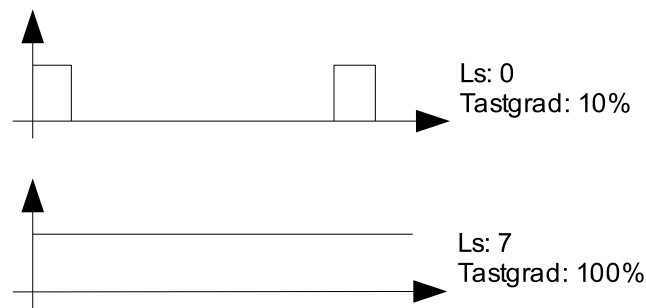


Abbildung 10: Pulsweitenmodulation beim RCX

### 2.3.1.2 Anschließen der Motoren

Der RCX und die Motoren werden mit einem Verbindungskabel verbunden, welche mit speziellen quadratischen Steckern ausgestattet sind. Durch entsprechendes drehen des Steckers erfolgt eine Drehrichtungs-umkehrung des Motors. Diese Tatsache beruht auf dem speziellen Aufbau des Steckers vom Verbindungskabel und der Kontaktplatte des RCX, die in der Abbildung 18 bzw. Abbildung 20 gezeigt werden. Daraus ergibt sich für die Abbildung 11 folgende Situation bezüglich des Anschließens und einer Änderung der Drehrichtung der Motoren: Wenn das Verbindungskabel am Ausgang A um 90° weiter nach rechts gedreht wird, bleibt die eingezeichnete Drehrichtung erhalten. Wenn stattdessen aber das Verbindungskabel um 90° weiter nach links gedreht wird, ändert sich die eingezeichnete Drehrichtung des Motors. Dabei wird vorausgesetzt, dass das Verbindungskabel am Motor so angeschlossen ist, dass das Kabelende von der Welle weg zeigt. Dadurch ändert sich die Drehrichtung der Motoren für die beiden folgenden Kombinationen nicht:

- Kabelende am RCX zeigt nach unten bzw. nach links
- Kabelende am RCX zeigt nach oben bzw. nach rechts

Die Abbildung 11 zeigt die Kombination, bei der die Motoren - eine entsprechende Programmierung durch die Software vorausgesetzt - in die richtige Richtung drehen. Die roten Pfeile geben die Drehrichtung der Motoren beim Vorwärtslauf an.



Abbildung 11: Anschließen der Motoren für die richtige Drehrichtung

## 2.4 Sensoren

Die Sensoren sind wohl der wichtigste Teil eines Roboters, sowie bei jeden anderen autonomen System auch. Nur mit Hilfe der Sensoren kann der Roboter auch seine Umwelt wahrnehmen und darauf reagieren. Aus diesen Grund werden die LEGO-Sensoren, die für die Laborversuche im Labor „Mobile Roboter“ gebraucht werden, in diesem Abschnitt noch näher erläutert. Es werden die drei folgenden Sensoren eingesetzt:

- Berührungssensor (passiv)
- Lichtsensor (aktiv)
- Rotationssensor (aktiv)

Der RCX besitzt für den Anschluss der LEGO-Sensoren drei Eingänge. Zudem unterscheidet er noch zwischen zwei verschiedenen Arten von Sensoren. Das sind zum einen die passiven und zum anderen die aktiven Sensoren.

Bevor die Sensoren richtig verwendet werden können, müssen sie über die Software konfiguriert werden. Dazu wird der Sensortyp und der Sensormodus konfiguriert. Der Sensortyp legt fest, wie der Sensor mit dem RCX interagiert. Im Gegensatz dazu legt der Sensormodus fest, wie die Messwerte des Sensors zu interpretieren sind.

Aus dem Messwert (Ergebniswert) des Sensors wird der Rohwert gebildet. Der Rohwert ist das direkte Ergebnis der A/D-Wandlung, welches aus dem tatsächlich gemessenen Spannungswert gebildet wird.

Zusätzlich dazu, stellt der RCX für alle Sensoren einen logischen (booleschen) Wert bereit. Der logische Wert wird aus der An/Aus-Entscheidung aufgrund des Rohwertes  $W_a$  (wahrer Wert) und des Steigungsparameters  $m$  bestimmt.

Aus dem Rohwert wird der so genannte verarbeitende Wert abhängig vom Sensormodus gebildet.

### 2.4.1 Passive Sensoren

Ein passiver Sensor besitzt einen sehr einfachen Aufbau, denn er besteht im allgemeinen aus einem Widerstand. Daher führt der RCX bei den passiven Sensoren zur Messwertbestimmung alle 3ms eine Widerstandsmessung mit einer Spannung von 5V aus. Dazu legt der RCX den Sensor über einen 10kOhm Widerstand an eine Spannung von 5V und bildet somit einen Spannungsteiler. Der A/D-Wandler konvertiert den Messwert, der sich aus dem Spannungsabfall am Widerstand ergibt, in eine 10Bit-Zahl (Rohwert). Für den Rohwert nach Formel 5 ergeben sich bei einem Spannungsabfall von 0V ein Rohwert von 0 und dementsprechend für 5V eine Rohwert von 1023.

### 2.4.2 Aktive Sensoren

Der RCX steuert die aktiven Sensoren genauso an, wie die passiven Sensoren. Der einzige Unterschied besteht darin, dass eine zusätzliche Spannungsversorgung von circa 8V periodisch über einem Transistor zugeschaltet wird. Für den RCX bedeutet das, dass der Sensor 3ms lang mit Spannung versorgt wird. Diese Phase wird als „Power Phase“ bezeichnet. Nach Ablauf der 3ms unterbricht der RCX dann die Spannungsversorgung für 0,1ms und liest den aktuellen Sensorwert aus. Diese Phase wird daher als „Read Phase“ bezeichnet. Da die Umschaltung zwischen der Power und Read Phase von der Firmware (Betriebssystem) gesteuert wird, können sie je nach Firmware variieren. Die hier angegebenen Zeiten beziehen sich auf die LEGO-Firmware „firm0328.lgo“. Zudem berechnet sich der Rohwert des A/D-Wandlers wie bei den passiven Sensoren nach Formel 5, da in diesem Zeitraum (Read Phase) die Spannungsversorgung von 8V nicht zugeschaltet ist.

$$W_a = \frac{U}{5V} \cdot 1023 \quad (5)$$

Aufgrund dessen, dass ein Sensor nur zwei Leitungen besitzt, ist eine spezielle interne Beschaltung des Sensors für die Realisierung der Power und Read Phase erforderlich. Da diese Beschaltung aber nicht zum weiteren Verständnis beiträgt, wird sie an dieser Stelle nicht vorgestellt.

### 2.4.3 Sensormodus

Der RCX besitzt acht verschiedene Sensormodi von denen einige nur für bestimmte Sensoren sinnvoll sind. Im folgenden Abschnitt werden die einzelnen Modi näher erläutert.

#### 2.4.3.1 Rohmodus

Beim Rohmodus wird der Rohwert dem verarbeiteten Wert gleichgesetzt. Da der RCX einen 10Bit A/D-Wandler besitzt, ergibt sich daraus eine Zahl zwischen 0 und 1023.

#### 2.4.3.2 Logischer Modus

Beim Logischen Modus wird der logische Wert weitergegeben. Dieser Modus ist am sinnvollsten in Verbindung mit einem Berührungssensor oder anderen Sensoren, die ebenfalls logische Werte nutzen. Wie oben schon mal erwähnt, wird der logische Wert aus der An/Aus-Entscheidung aufgrund des Rohwertes  $W_a$  (wahrer Wert) und des Steigungsparameters  $m$  bestimmt.

Der Steigungsparameter  $m$  reicht von 0 bis 31 und bestimmt die Konvertierung vom Rohwert in einen logischen Wert. Wenn der Steigungsparameter gleich Null gesetzt wird, ergibt sich eine Konvertierung wie sie in Tabelle 3 gezeigt wird.

<b>Bedingung</b>	<b>Logischer Wert</b>
$W_a > 562$	0
$W_a < 460$	1
$460 \leq W_a \leq 562$	unverändert

Tabelle 3: Logische Konvertierung für  $m$  gleich Null [DBL-01]

Damit bei leichten Schwankungen des Rohwertes, der logische Wert nicht ständig zwischen den zwei möglichen Zuständen hin und her geschaltet wird, wird eine Hysterese zur Reduktion der Schwankungen eingesetzt. Die Grenzpunkte - sprich Umschaltpunkte - der Hysterese liegen bei 45% und 55% der Eingangsspannung des Sensors (siehe Abbildung 12).

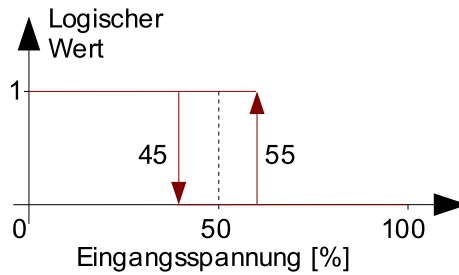


Abbildung 12: Hysterese

Wird der Steigungsparameter nicht gleich Null gesetzt, ergibt sich eine andere Konvertierung als oben beschrieben wurde. Jetzt wird bei jedem Auslesen des Sensors der vorige Rohwert mit dem aktuellen Rohwert verglichen. Es erfolgt also eine Differenzbildung; wenn dieser gebildete Absolutwert  $\Delta W$  kleiner als der Steigungsparameter  $m$  ist, dann wird der logische Wert nicht verändert. Wenn der gebildete Absolutwert aber größer als der Steigungsparameter ist, wird der logische Wert so gebildet, dass aus dem Ergebnis zu erkennen ist, ob der Rohwert zunahm (logischer Wert 0), oder abnahm (logischer Wert 1). Die Tabelle 4 zeigt die Konvertierung für  $m$  ungleich Null.

<b>Bedingung</b>	<b>Logischer Wert</b>
$\Delta W > m$	0
$\Delta W < -m$	1
$W_a > (1023 - m)$	0
$W_a < m$	1

Tabelle 4: Logische Konvertierung für  $m$  ungleich Null [DBL-01]

### 2.4.3.3 Flankenzählungsmodus

Beim Flankenzählungsmodus zählt der RCX die Zustandswechsel des logischen Wertes (Flanken). Dies bedeutet, dass der RCX einen Zähler mit dem Startwert Null immer um Eins erhöht, wenn ein Zustandswechsel des logischen Wertes auftritt; also von 0 zu 1 (steigende Flanke) oder von 1 zu 0 (fallende Flanke). Bei diesem Zählmodus ist zu beachten, dass er eine Totzeit von 300ms aufweist. Diese Zeit dient dazu, um Störsignale auszufiltern, wie z. B. das Entprellen eines Schalters.

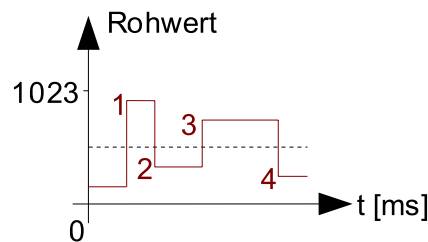


Abbildung 13: Flankenzählungsmodus

### 2.4.3.4 Impulzzählungsmodus

Beim Impulzzählungsmodus zählt der RCX die logische Übergänge von 0 auf 1 (steigende Flanke). Wobei hier auch eine Totzeit von 300ms wie beim Flankenzählungsmodus nach einen Wechsel angewandt wird.

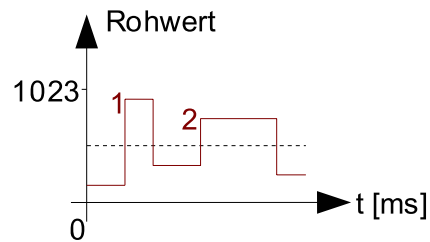


Abbildung 14: Impulzzählungsmodus

#### 2.4.3.5 Prozentmodus

Beim Prozentmodus erfolgt eine lineare Umrechnung vom Rohwert in einem Prozentwert, wobei ein Rohwert von 0 einen Prozentwert von 100 und ein Rohwert von 1023 einen Prozentwert von 0 entspricht.

#### 2.4.3.6 Rotationsmodus

Der Rotationsmodus verwendet einen Algorithmus, der die Ausgabe des Rotationssensors dekodiert. Dies sieht so aus, dass er die Winkelinkremente und -dekremente in Stufen von  $22,5^\circ$  zählt. Daraus folgt, dass ein Ergebniswert von 16 einer vollen Umdrehung also  $360^\circ$  entspricht.

### 2.4.4 Sensortypen

In diesem Abschnitt wird der Aufbau und die Funktion, sowie die Konfiguration der einzelnen Sensortypen näher erläutert.

#### 2.4.4.1 Berührungssensor

Der Berührungssensor ist der einfachste Sensor von LEGO und gehört zur Gruppe der passiven Sensoren, denn er besteht nur aus einem einfachen Taster (Schließer). Das bedeutet für die Funktion, dass - sobald der Schalter schließt - der Stromkreis dadurch geschlossen wird. Der Standardmodus für den Berührungssensor ist der Logische Modus, aber auch die Flanken- und die Impulzzählung können manchmal nützlich sein.



Abbildung 15: Berührungssensor

#### 2.4.4.2 Lichtsensor

Die wichtigsten Bauteile für die Funktion des Lichtsensors sind die rote Lumineszenz-Diode (Leuchtdiode oder kurz: LED) und der Fototransistor. Da diese Bauteile eine kontinuierliche Spannungsversorgung brauchen, gehört der Lichtsensor zur Gruppe der aktiven Sensoren. Die LED emittiert rotes Licht, dessen reflektierter Anteil mit dem Fototransistor gemessen wird. Natürlich, und das ist nicht immer von Nutzen, reagiert der Fototransistor auch auf das Umgebungslicht. Je nach Ladungszustand der Batterien und der Helligkeit des Hintergrundlichts können die gemessenen Werte schwanken. Um die Einflüsse des externen Lichts, z. B. bei der Linienverfolgung oder bei Erkennung von Oberflächen, soweit wie möglich zu minimieren sollte der Lichtsensor so nah wie möglich an der Zieloberfläche platziert werden. Zudem besitzt der Lichtsensor auch noch eine Empfindlichkeit gegenüber den reflektierten IR-Signalen von der IR-Schnittstelle des RCX.

Der Standardmodus für den Lichtsensor ist der Prozentmodus. Falls aber eine höhere Auflösung gebraucht wird empfiehlt es sich mit dem Rohmodus zu arbeiten.

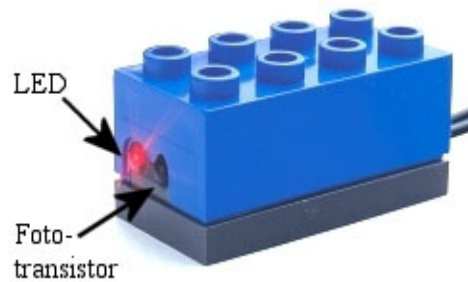


Abbildung 16: Lichtsensor

#### 2.4.4.3 Rotationssensor

Die wichtigsten Bauteile für die Funktion des Rotationssensors sind die zwei Lichtschranken und ein vierflügender Rotor, der das IR-Licht beim rotieren in einer bestimmten Abfolge unterbricht. Dadurch werden Widerstände ab- bzw. zugeschaltet, die dadurch definierten Werte werden letztendlich Schritten zugeordnet. Anhand der Abfolge dieser Schritte kann der RCX eine Drehrichtungsumkehrung erkennen. Der Rotationssensor gehört aufgrund seiner integrierten Schaltung zu den aktiven Sensoren. Er misst die Umdrehung einer Achse, die in ihn gesteckt wurde, mit einer Auflösung von  $22,5^\circ$ . Dadurch ergibt sich ein Sensorwert von 16 für eine volle Umdrehung.

Der Standardmodus des Rotationssensors ist der Rotationsmodus. Bei diesem Modus ist der Startwert Null. Bei Drehung der Achse erhöht bzw. erniedrigt sich der Wert jedoch - je nach Drehrichtung. Der Rotationssensor kann jederzeit durch einen speziellen Befehl auf den Startwert zurückgesetzt werden.



Abbildung 17: Rotationssensor [CMA-04]

#### 2.4.5 Anschließen der Sensoren

Der Berührungssensor ist der einzige Sensor von LEGO, der vom Benutzer mit einem Verbindungskabel an beiden Enden, also am RCX und am Sensor selbst, angeschlossen werden muss. Bei den anderen Sensoren, also beim Licht- und Rotationssensor ist das Kabel am Sensor bereits vormontiert. Zum besseren Verständnis werden erst einmal die 2x2 Kontaktplatten des RCX und des Verbindungskabels betrachtet. Bei der 2x2 Kontaktplatte sind jeweils zwei Kontakte zusammen geschaltet, wie es die Abbildung 18 zeigt.

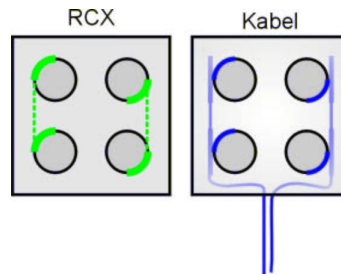


Abbildung 18: Interne Verdrahtung der Kontaktplatten vom RCX und Kabel [FHO-01]

Der Berührungssensor besteht aus einem 2x3 Baustein, der in der vorderen Hälfte des Bausteins die elektrischen Kontakte beinhaltet (siehe Abbildung 15). Deshalb stehen für diesen Sensor vier verschiedene Möglichkeiten bezüglich der Positionierung der Kontaktplatte des Verbindungskabels zur Verfügung (siehe Abbildung 19).

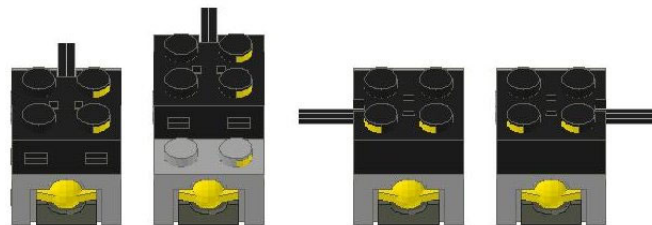


Abbildung 19: Anschlussmöglichkeiten des Berührungssensors

Beim Anschluss am RCX ist bezüglich der Orientierung der Kontaktplatte vom Verbindungskabel nichts besonderes zu beachten. Da die Kontaktplatten, wie es aus der Abbildung 18 hervorgeht, Viertelkontakte besitzen. Dabei wird vorausgesetzt, dass alle vier Kontakte der Kontaktplatten vom RCX und Verbindungskabel übereinander liegen. In der Abbildung 20 wird dabei gezeigt, welche Möglichkeiten bezüglich des Zusammensteckens der beiden Kontaktplatten bestehen. Dabei werden nur zwei der vier Möglichkeiten gezeigt, da die beiden anderen sich aus der Spiegelung der beiden gezeigten Möglichkeiten ergeben.

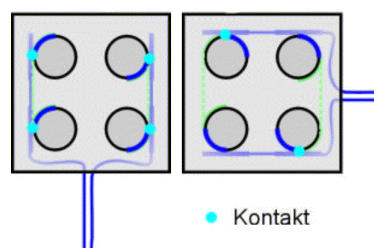


Abbildung 20: Anschlussmöglichkeiten des Verbindungskabels am RCX [FHO-01]

## 2.5 Differential

Das wichtigste und komplizierteste Bauteil im mechanischen Aufbau des Roboters ist das Differential. Es besteht aus einem Rahmen bzw. Ausgleichsgehäuse (Blau gekennzeichnet) und drei frei drehbaren Kegelrädern. Ein Kegelrad ist ein spezielles 12er-Zahnrad. Eines von den Kegelrädern (Grün gekennzeichnet) ist mittig am Rahmen gelagert und die beiden anderen (Orange gekennzeichnet) werden über von Außen aufgesteckte Achsen bewegt.



Das Differential hat bei diesem Roboter die Aufgabe die Drehzahldifferenz der beiden Achsen zu ermitteln, damit diese durch einen Rotationssensor, der über ein Zahnrad mit dem Differential verbunden ist, gemessen werden kann. Diese Messwerte werden hauptsächlich zur Regelung der Geradeausfahrt eingesetzt, können aber auch für die Kurvensteuerung bzw. für die Drehung des Roboters verwendet werden.

Aufgrund dieser Anforderungen darf sich das Ausgleichsgehäuse nur bei einer Drehzahldifferenz drehen. Damit das Ausgleichsgehäuse bei gleicher Drehzahl stehen bleibt, muss die Rotation an den beiden Seiten des Roboters gegensinnig orientiert sein. Sobald sich dann eine Drehzahldifferenz einstellt, fängt der Rahmen an, sich zu drehen.

Die genaue Funktionsweise des Differentials wird anhand der Abbildung 21 erklärt. Dabei wird der Effekt ausgenutzt, dass sich die Achsen gegensinnig drehen, wenn der Rahmen festgehalten wird. Aufgrund dieses Effektes bzw. den weiter oben beschriebenen Anforderungen, wurde der Aufbau des Roboters so gewählt, dass die äußeren Zahnräder von Außen gegensinnig gedreht werden. Das mittlere Zahnrad (Grünes Zahnrad) dreht sich mit der gleichen Geschwindigkeit, wie die anderen beiden (Orange Zahnräder) und wird deswegen in der aktuellen Position gehalten. Wenn die äußeren Geschwindigkeiten voneinander abweichen, weicht das mittlere Zahnrad seitlich aus, wodurch eine Rotation des Ausgleichsgehäuses verursacht wird. Zusammenfassend kann gesagt werden, dass die Rotation vom Ausgleichsgehäuse bei dieser gewählten Konfiguration eine Funktion der Drehzahldifferenz der äußeren Zahnräder bzw. deren Achsen ist. [ASR-02]

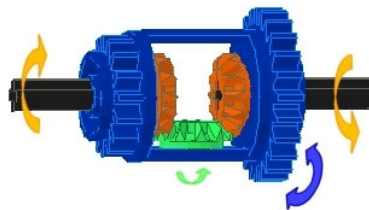


Abbildung 21: Differential bei sich gegensinnig drehenden Achsen

## Literaturverzeichnis

ART-04: ART DECOM, Stand: 12.07.2004,

<http://artdecom.mesh.de/projekte/werkzeuge/hardware/LEGO-hardware/hardware-von-LEGO-frameset.html>

ASR-02: Bauer & Reichel, Anti-Schlupf-Regelung, TU Ilmenau, Stand: 12.07.2004,

<http://www.theinf.tu-ilmenau.de/~nuetzel/mindstorms/asr2002>

CMA-04: Centre for Microcomputer Applications, Stand: 12.07.2004,

<http://www.cma.science.uva.nl/english/products/LEGOSaAe.html>

DBL-01: Dave Baum, Dave Baums LEGO MINDSTORMS Roboter, 2001, 1. Nachdruck, Galileo Press

FHM-03: Blockunterricht 3T, Prof. Dr. R. Schiedermeier, FH München, Stand: 22.06.2004,

<http://www.cs.fhm.edu/~schieder/block-03/index.html>

FHO-01: Prof. Dr. Michael Wülker, Skript zu Sensoren und Aktoren, FH Offenburg, Stand: 22.06.2004,

<http://mv.sirius.m.fh-offenburg.de/robotik/WS01Material/SensorenUndAktoren.pdf>

GRO-98: ROBOLAB - Das Grundlagenbuch (Lehrerhandbuch zur ROBOLAB 2.0 Software), 1998

MIN-04: LEGO Mindstorms, Stand: 22.06.2004, <http://mindstorms.lego.com/de/index.asp>

PHI-04: Stand: 12.07.2004, <http://www.philohome.com/sensors.htm>