Proposal für die Masterarbeit

Real-Time Signalübertragung sowie EEG-Markerversendung und automatisierte Ballrückführung

Eugen Nießen

Studiengang Systems Engineering
Masterarbeit im Projekt BRIO2

4. Januar 2010

Inhaltsverzeichnis

1 Anforderungen **3**
1.1 Ziele der Arbeit . . . 3

2 Arbeitspakete **4**
2.1 AP1: EEG-Marker und Mikrocontroller . . . 4
2.2 AP2: Echtzeitkommunikation . . . 5
2.3 AP3: Entwurf der Ballrückführung . . . 6
2.3.1 Vorgespannte Feder . . . 7
2.3.2 Raupenantrieb . . . 7
2.3.3 Schneckenantrieb . . . 8
2.3.4 Wippenantrieb . . . 8
2.4 AP4: Realisierung der Ballrückführung . . . 9

3 Zeit- und Kostenplan **9**
3.1 Zeitplan . . . 9
3.2 Kostenaufstellung . . . 10

1 Anforderungen und Motivation

Im Projekt Labyrinth2 dient ein modifiziertes Briolabyrinth® als Testumgebung zur Aufzeichnung verschiedener, für die Analyse des EEGs relevanter Daten. Die Modifikationen bestehen sowohl aus Sensorik, Motorik, als auch weiteren modularen Komponenten, wie dem Ballmagazin. Sie dienen der Beeinflussung des Spielverhaltens und der Kalibrierung der Testumgebung.

- Potentiometer messen permanent die Rotation der Spielachsen und liefern Signale anhand derer die Ausrichtung des Spielfeldes stets bekannt ist.
- Ein Piezo-Sensor befindet sich unterhalb des Spielbodens und registriert, wenn der Spielball durch ein Loch gefallen ist.
- Vier Schalter im Inneren Spiels dienen der Kalibrierung der Winkel der Spielfläche. Zudem registrieren sie starke Ausschläge der Spielfläche beim spielen.
- Taster unterhalb der beiden Drehachsen können ein Drücken des Spielers auf die Drehknöpfe registrieren.

Während eines laufenden Versuches sorgen diese Bauteile ständig für einen Signalfluss, welcher größtenteils über eine parallele Schnittstelle in eine NI-Karte[1] gelangt. Dort findet eine Datenerfassung statt, bevor die Daten im Computer ausgewertet werden. Die NI-Karte sorgt für eine bestimmte Latenz der Datenausgabe.
Die Erfassung der Verhaltensdaten geschieht über einen MS Windows®-Rechner auf dem Presentation®[2] installiert ist. Presentation® ist ein spezielles neurowissenschaftliches Programm zum steuern von Experimenten. Es liefert Daten über verschiedene Stimuli wie die Bewegung der Augen, fMRI, ERP und MEG. Laut des Herstellers ist Presentation® in der Lage Latenzen von unter einer Millisekunde bei der Messung der Stimuli zu erreichen[3].

Neben der sensorischen Komponenten befinden sich in der Testumgebung auch aktuatorische Bauteile, wie das Ballmagazin, welches sowohl manuell als auch über einen Servomotor Spielbälle auf das Spielfeld befördern kann. Bis zu 14 dieser Spielbälle müssen vor Spielbeginn ins Magazin eingefüllt werden. Das Spielen wird deutlich bequemer, da die Spielbälle nicht vom Spieler nach gelegt werden müssen. Zudem bietet das Ballmagazin, in Verbindung mit zwei Servomotoren, welche die Spielachsen steuern, die Möglichkeit einer Teleoperation des Spieles. Das Spiel kann auch vollständig ohne das Eingreifen des Menschen von einem Computer bedient werden, welcher somit die Möglichkeit bekommt das Spiel selbst spielen zu lernen. Die genutzten Lernverfahren benötigen oft sehr lange zum Lernen des Spiels. In einem solchen Fall reichen die gespeicherten Spielbälle im Magazin nicht aus, sodass ein Eingreifen des Menschen erforderlich ist.

1.1 Ziele der Arbeit

- AP1/AP2: Mikrocontroller und Echtzeitkommunikation
  Da der durch die NI-Karte verursachte Zeitversatz für die Qualität der Messergebnisse kritisch ist, besteht die erste Aufgabe der Arbeit darin, die NI-Karte vollständig zu ersetzen. Es soll eine schnellere Kommunikation anhand eines Mikrocontrollers realisiert werden (siehe Kap. 2.1 und Kap. 2.2).

[1] http://sine.ni.com/nips/cds/view/p/lang/en/nid/14114
[2] http://www.neurobs.com/
[3] http://www.neurobs.com/presentation/docs

- AP3/AP4: Entwurf und Realisierung der Ballrückführung
  Das Ersetzen des Ballmagazins durch eine automatisierte Lösung ist die zweite Anforderung der Arbeit. Mehrere denkbare Mechanismen werden dafür hinsichtlich ihrer Eignung untersucht und nach bestimmten Kriterien, wie Kosten, Aufwand und Integrierbarkeit in die Testumgebung, verglichen, um eine optimale Lösung zu finden (siehe Kap. 2.3 und Kap. 2.4).

2 Arbeitspakete

2.1 Schreiben der EEG-Marker durch einen Mikrocontroller

AP1 Auswahl des Mikrocontrollers, Multiplexprotokoll

Im der ersten Version des Brio©-Spiels werden alle sensorischen Signale in einer Box zusammengeführt und verschaltet. Sie werden im Anschluss über eine parallele Leitung mit der NI-Karte verbunden. Diese Architektur wird wie in der folgenden Abbildung (Abb. 2.1) durch einen Mikrocontroller ersetzt. Die Fähigkeit des Mikrocontrollers sowohl parallele als auch serielle

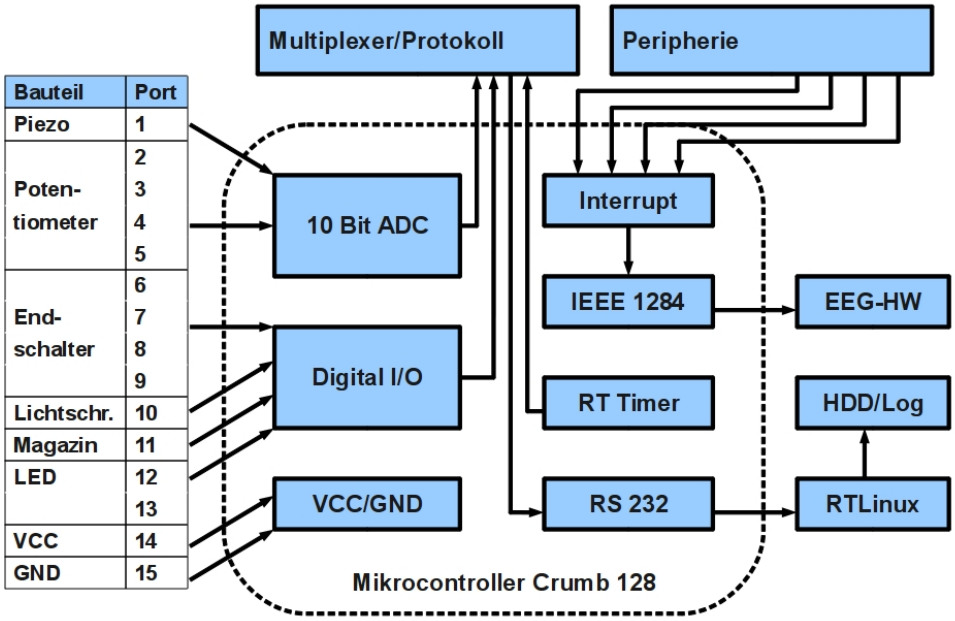


Abbildung 1: Konzept der Kommunikation

Signale übertragen zu können, ermöglicht eine Kommunikation über ein serielles Protokoll. Dieses Protokoll bedient sich des Prinzips der Multiplex-Datenübertragung[4] und gewährleistet eine sehr schnelle Übertragung der zeitkritischen Signale. Das Protokoll ist flexibel und überträgt die Signale nach Prioritäten bezüglich der Latenzen. Die serielle Signalübertragung eines geeigneten

[4] http://de.wikipedia.org/wiki/Multiplexing

Mikrocontrollers geschieht nahezu in Echtzeit. Der hier im Institut oft benutzter Mikrocontroller Crumb128[5] ist für diese Aufgabe gut geeignet und ist zudem sehr gut dokumentiert.

2.2 Aufbau der Kommunikation anhand eines Echtzeit-Kernels

AP2 Konzept der Kommunikation, Echtzeit-Kernel

Die zeitkritischen Signale der seriellen Übertragung werden als Marker direkt in die EEG-Datei geschrieben. Diese Marker dienen der Synchronisierung der Ereignisse der Testumgebung mit den korrespondierenden Ereignissen innerhalb des EEGs. Da das Schreiben des EEGs mit sehr geringer Latenz geschieht, ist es sehr wichtig, dass die Marker ähnlich schnell kommuniziert werden. Die Auswertung der Signale geschieht bisher auf einem Computer mit einem MS Windows© Betriebssystem. Dieses System erzeugt zum Teil sehr hohe Latenzen.
Die Datenerfassung wird auf ein Linux-basiertes System übertragen, welches durch eine offene Architektur des Kernels mit einem Echtzeit-Kernel-Patch[6] ausgestattet werden kann. Dieser Echtzeit-Kernel-Patch schafft eine separate, ausgelagerte Schnittstelle zum Prozessor, welche durch interne Interrupts oder andere Verzögerungen des Schedulers kaum beeinträchtigt wird. Das Prinzip ähnelt einem Beipass und ermöglicht ein Einlesen des seriellen Signals nahezu in Echtzeit.
Die folgende Abbildung (Abb. 2.2) zeigt das Prinzip der ausgelagerten Echtzeit-Prozesse[7]:

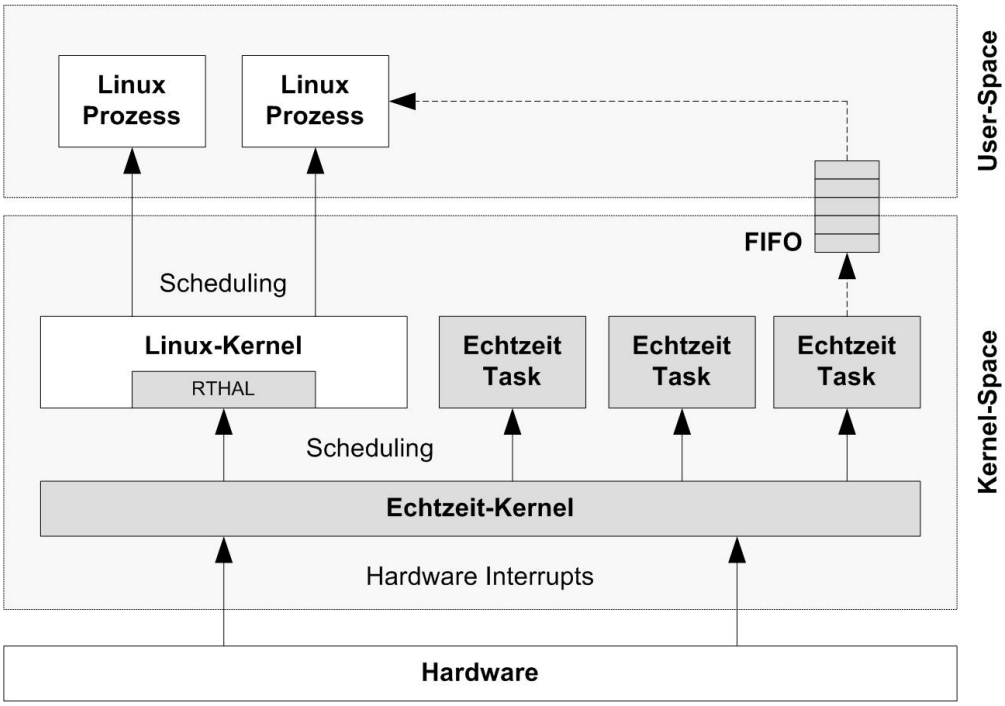


Abbildung 2: Konzept des Echtzeit-Kernels

[5] http://www.chip45.com/download/Infosheet_Crumb128_080715.pdf
[6] http://rt.wiki.kernel.org/index.php/CONFIG_PREEMPT_RT_Patch
[7] http://de.wikipedia.org/wiki/RTAI

2.3 Entwurf einer automatischen Rückführung des Spielballs

AP3 Konzepte der Transportarten, Auswahl nach Kriterien

Um die Ballrückführung vollständig automatisieren zu können, muss der Spielball den gesamten Weg von der ursprünglichen Ballöffnung bis hin zur Startposition der Spielfläche befördert werden. Dieses kann durch eine am Spieläusseren angebrachte, geneigte Röhre (Abb. 2.3) vollbracht werden.

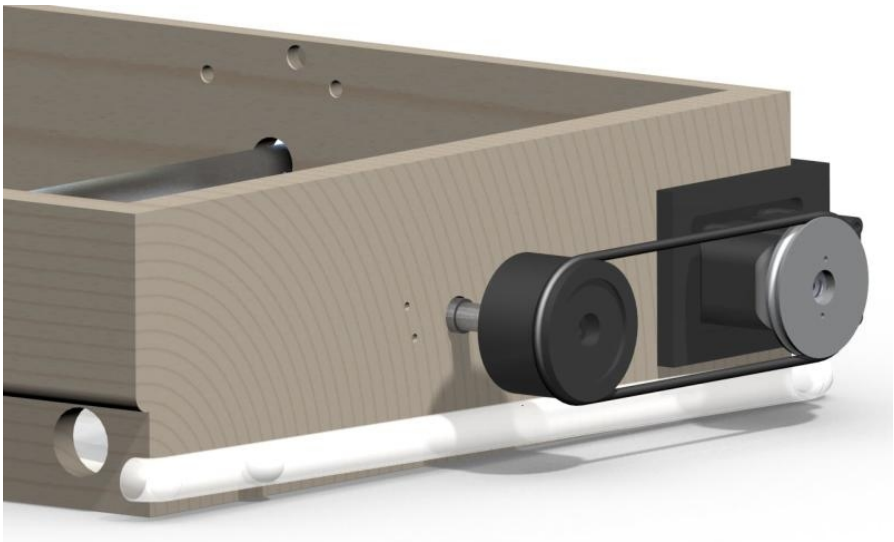


Abbildung 3: Konzept der Transportröhre

Alternativ kann jedoch auch der Innenboden des Spiels so verändert werden, dass der Ausgang des Spielballs bereits auf der gegenüberliegenden Seite liegt. So entfällt die Notwendigkeit der Röhre.

Ist der Spielball auf die benötigte Seite des Spiels transportiert worden, so muss er auf die Höhe der Spielfläche befördert werden. Für diese Funktionalität sind verschiedene Mechanismen denkbar. Berücksichtigt man die vom Spiel vorgegebenen Beschränkungen, so könnten die folgenden vier Methoden des Transports in Frage kommen (siehe Kap. 2.3.1,Kap. 2.3.2, Kap. 2.3.3 und Kap. 2.3.4).

2.3.1 Vorgespannte Feder

Dieser Mechanismus speichert kinetische Energie über eine gespannte Feder, welche bei Kontakt mit dem Spielball gelöst wird und ihn über ein Rohr oder eine Rampe auf die Startposition stößt. Das Prinzip ähnelt dem eines Flipper-Automaten (vgl. Abb. 2.3.1).

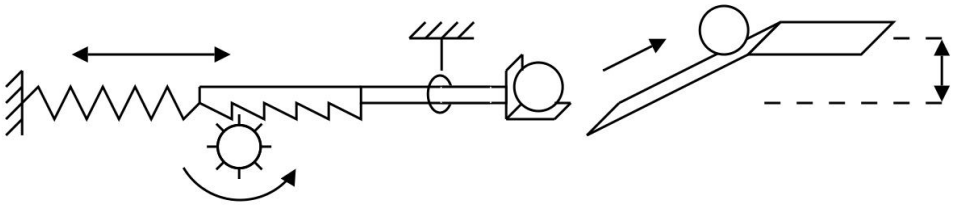


Abbildung 4: Transport über eine vorgespannte Feder

2.3.2 Raupenantrieb

Der Raupenantrieb transformiert eine rotatorische in eine translatorische Bewegung und befördert den Spielball ähnlich einer Rolltreppe (vgl. Abb. 2.3.2).

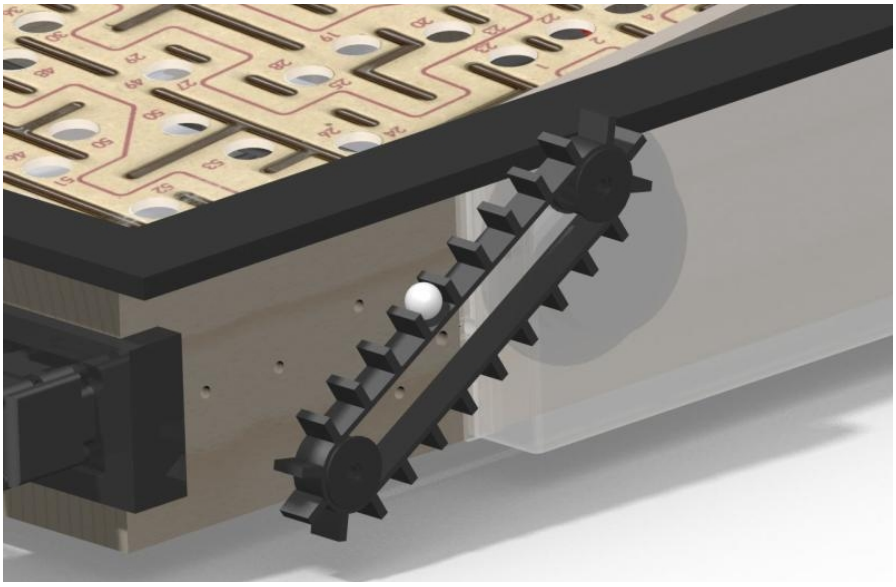
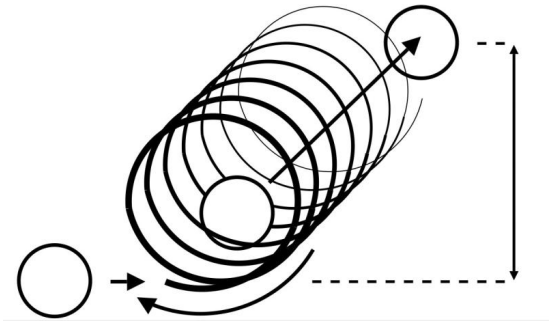


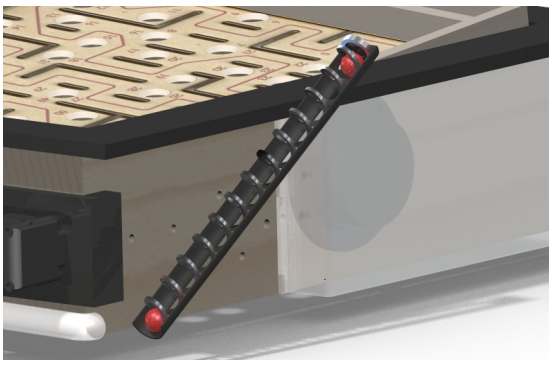
Abbildung 5: Transport über einen Raupenantrieb

2.3.3 Schneckenantrieb

Bei dem Schneckenantreib handelt es sich um eine rotierende Helix. Der Windungsabstand hat mindestens den Durchmesser des Spielballs und die Rotationsgeschwindigkeit ist so zu wählen, dass der Spielball in eine Aufwärtsbewegung versetzt wird (vgl. Abb. 6(a)).



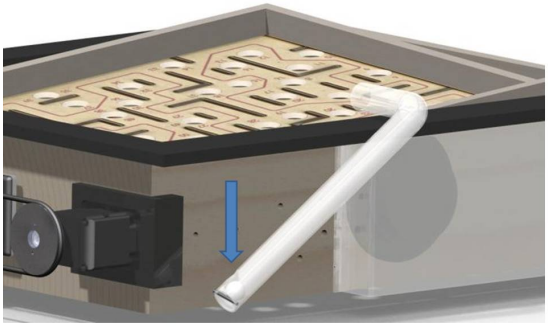
(a) Konzept des Schneckenantriebes



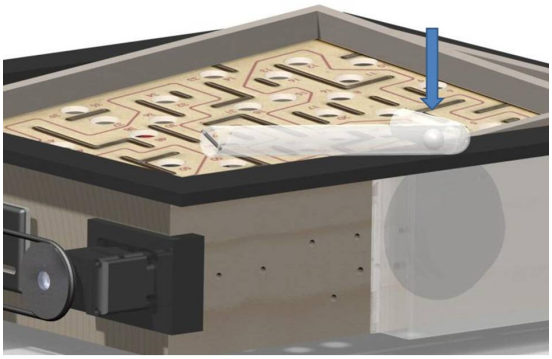
(b) Schneckenantrieb an der Testumgebung

2.3.4 Wippenantrieb

Bei dem Wippenantrieb handelt es sich um eine L-förmig gebogene Röhre. Rotiert man sie um die Achse des kürzeren Abschnittes der Röhre, so hebt sich der lange Abschnitt und der Spielball rollt durch die Röhre auf die Startposition (vgl. Abb. 6(c) und Abb. 6(d)).



(c) Transport über einen Wippenantrieb im Ausgangszustand



(d) Transport über einen Wippenantrieb im Endzustand

2.4 Entwurf einer automatischen Rückführung des Spielballs

AP4 Realisierung des Transports, Automatisierung

Der in der Evaluierung der Konzepte ausgewählte Mechanismus zum Transport des Spielballs wir in diesem Arbeitspaket integriert. Die Transportröhre (vgl. Abb. 2.3) wird mit dem gewählten Mechanismus kombiniert und eine Automatisierung des Ablaufs wird realisiert. Dazu wird notwendige Sensorik und Motorik integriert. Zu möglichen sensorischen Komponenten könnten Lichtschranken, Taster oder Piezo-Sensoren gehören und die Motorik kann durch Antriebe wie den Dynamixel Dx-117[8] realisiert werden. Dieser Servomotor wird in der Testumgebung bereits zum Antreiben der Spielachsen benutzt. Die Servomotoren können anhand der sogenannten „daisy chain“ sehr leicht miteinander verbunden werden. So wird die Bedienung erleichtert und der Aufwand verringert.

3 Zeitplan und Kostenaufstellung

3.1 Zeitplan

Der zeitliche Rahmen einer Masterarbeit umfasst sechs Monate. Die Reihenfolge der vier Arbeitspakete ist aus produktionstechnischen Gründen folgendermaßen gewählt worden (siehe Abb. 3.1 und Abb. 3.1):

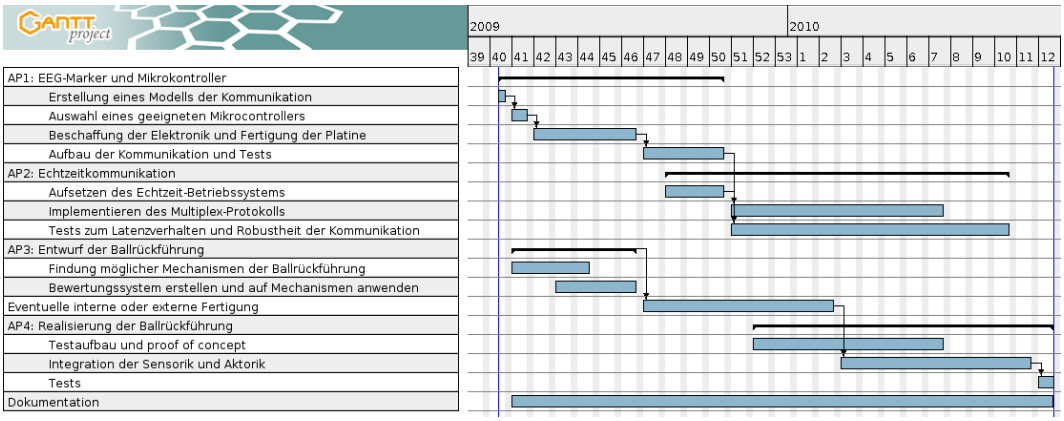


Abbildung 6: Zeitplanung anhand eines Gantt-Diagramms

[8] http://www.robotshop.ca/robotis-bioloid-dynamixel-dx-117.html

Vorgang	Anfang	Ende
AP1: EEG-Marker und Mikrokontroller	01.10.09	12.12.09
Erstellung eines Modells der Kommunikation	01.10.09	03.10.09
Auswahl eines geeigneten Mikrocontrollers	05.10.09	10.10.09
Beschaffung der Elektronik und Fertigung der Platine	12.10.09	14.11.09
Aufbau der Kommunikation und Tests	16.11.09	12.12.09
AP2: Echtzeitkommunikation	23.11.09	13.03.10
Aufsetzen des Echtzeit-Betriebssystems	23.11.09	12.12.09
Implementieren des Multiplex-Protokolls	14.12.09	20.02.10
Tests zum Latenzverhalten und Robustheit der Kommunikation	14.12.09	13.03.10
AP3: Entwurf der Ballrückführung	05.10.09	14.11.09
Findung möglicher Mechanismen der Ballrückführung	05.10.09	30.10.09
Bewertungssystem erstellen und auf Mechanismen anwenden	19.10.09	14.11.09
Eventuelle interne oder externe Fertigung	16.11.09	16.01.10
AP4: Realisierung der Ballrückführung	21.12.09	27.03.10
Testaufbau und proof of concept	21.12.09	20.02.10
Integration der Sensorik und Aktorik	18.01.10	20.03.10
Tests	22.03.10	27.03.10
Dokumentation	05.10.09	27.03.10

Abbildung 7: Aufgelisteter Ablauf der Arbeitspakete

Während des ersten Arbeitspaketes (vgl. Kap. 2.1) fällt die Entscheidung bezüglich des geeigneten Mikrocontrollers und der Architektur der Kommunikation. Somit können bereits während des ersten Arbeitspaketes die notwendigen Komponenten beschafft werden. Auch das dritte Arbeitspaket (vgl. Kap. 2.3) beginnt frühzeitig, da das Ergebnis für eine eventuelle Fertigung im Anschluss entscheidend ist. Sollte die Fertigung bestimmter Komponenten des automatisierten Ballmagazins von einem externen Unternehmen geschehen, so ist ein ausreichend großer Zeitraum dafür berechnet worden. Der Großteil der sechs Monate widmet sich dann zu gleichen Teilen dem zweiten und dem vierten Arbeitspaket.

3.2 Kostenaufstellung

Die anfallenden Kosten werden je nach verwendeter Methode des Transports des Spielballs variieren. Daher kann eine genaue Kostenaufstellung noch nicht erstellt werden.
Die folgenden elektronischen Komponenten könnten Verwendung finden, wobei nur die Position 1 (Crumb 128) definitiv benötigt wird.

	Komponente	Erläuterung	Kosten
1	Crumb128	Mikrocontroller / Kap. 2.1 und Kap. 2.2	ca. 40 Euro
(2)	Piezo-Sensor	Erschütterungssensor mit Messelektronik	10 bis 40 Euro
(3)	Lichtschranke	Balldetektion für den Transport des Spielballs	bis 50 Euro
(4)	Taster	Balldetektion für den Transport des Spielballs	ca. 3 Euro

Bei den vier verschiedenen Methoden des Transports können jedoch noch weitere Kosten entstehen, von denen mindestens eine Posten definitiv benötigt wird.

	Komponente	Erläuterung	Kosten
(5)	Servoantrieb	Modellbaumotor / Kap. 2.3.2 und Kap. 2.3.3	bis 30 Euro
(6)	Dynamixel Dx-117	Servoantrieb / Kap. 2.3.1 und Kap. 2.3.4	ca. 130 Euro
(7)	Federstahl	Material der Helix / Kap. 2.3.3	bis 5 Euro

Hinzu könnten noch geringe Kosten für Leitungen, Lochplatine, Stecker, Buchsen und Verbrauchsmaterialien (Lötzinn, POM usw.) anfallen.