**Erstellung eines interdisziplinaren Lehr-Lern-Konzeptes für MINT und beispielhafte Umsetzung in der Automationstechnik**

**STUDIENARBEIT (T3100)**

im Studiengang Elektrotechnik – Infotronik

an der Dualen Hochschule Baden-Württemberg Mosbach

von

**Jana Konrad**

Abgabedatum 00.12.2022

Matrikelnummer, Kurs 3450731, ET20B

Dualer Partner Robert Bosch GmbH Bamberg

Betreuer der Studienarbeit Prof. Dr. Christian Kuhn

**Gender-Erklärung**

Aus Gründen der besseren Lesbarkeit wird in dieser Projektarbeit die Sprachform des generischen Maskulinums angewandt. Es wird an dieser Stelle darauf hingewiesen, dass die ausschließliche Verwendung der männlichen Form geschlechtsunabhängig verstanden werden soll.

**Ehrenwörtliche Erklärung**

Ich versichere hiermit, dass ich meine Projektarbeit T3\_3100 mit dem Thema: „Erstellung eines interdisziplinares Lehr-Lern-Konzept für MINT und beispielhafte Umsetzung in der Automationstechnik“ selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Ich versichere zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ort, Datum Unterschrift

Zusatzvermerk

Aufgrund des internationalen Interesses an Big Data, Data Mining und Data Analytics sind viele Fachbegriffe rein im Englischen definiert. Eine Übersetzung ins Deutsche erfolgt meist wörtlich und ist somit nicht immer in der Lage den Kern des Wortes optimal abzubilden. Ebenfalls kann durch die wörtliche Übersetzung ins Deutsche die Lesbarkeit des Textes darunter leiden. Aus diesen zwei Gründen werden deutsche und englische Fachbegriffe in folgender Arbeit als Synonym verwendet. Dieser Vermerk greift ausschließlich bei englischen Begriffen, welche ebenfalls in wissenschaftlich belastbaren deutschsprachigen Quellen verwendet werden, im allgemeinen deutschen Sprachgebrauch üblich sind oder durch firmeninterne Definitionen als Synonym oder Abkürzung verwendet werden. Ebenfalls muss bei erster Verwendung des englischen Begriffs die deutsche Entsprechung klar gekennzeichnet sein. Dies kann durch eine Nennung des deutschen Gegenstücks in der Fußnote erfolgen, aber auch die Kennzeichnung im Fließtext ist möglich. In beiden Fällen muss der Bezug zum englischen Begriff klar erkennbar sein.

Kurzfassung

Text

Abstract

Text

Inhaltsverzeichnis

[Abbildungsverzeichnis VI](#_Toc119703806)

[Tabellenverzeichnis VI](#_Toc119703807)

[Abkürzungsverzeichnis VII](#_Toc119703808)

[1 Einleitung 1](#_Toc119703809)

[2 Aufgabenstellung und Zielsetzung der Studienarbeit 2](#_Toc119703810)

[3 Aufbau des Projektes für die Studierenden 3](#_Toc119703811)

[3.1 Projektzielsetzung der Studierenden 3](#_Toc119703812)

[3.2 Rahmenbedingungen für die Studierenden 4](#_Toc119703813)

[3.3 Verwendete Hardware: Arduino und Braccio 5](#_Toc119703814)

[3.4 Beispiellösung der Programmierung und des Hardwareaufbaus 15](#_Toc119703815)

[4 Didaktische Lehrmittel 15](#_Toc119703816)

[4.1 Anforderungen an didaktische Konzepte für moderne Lehrformen 15](#_Toc119703817)

[4.2 Selektierung der Lehrmittel für die Zielgruppe 15](#_Toc119703818)

[5 Erstellung eines didaktischen Lehr- und Lernkonzeptes 15](#_Toc119703819)

[6 Praktische Umsetzung mithilfe des Konzeptes 15](#_Toc119703820)

[7 Dokumentation und Hilfestellung für den Projekteinsatz 16](#_Toc119703821)

[8 Kritische Reflexion und Bewertung 16](#_Toc119703822)

[9 Literaturverzeichnis 17](#_Toc119703823)

# Abbildungsverzeichnis

[Abbildung 1: Arduino UNO R3 - Board 5](#_Toc119703788)

[Abbildung 2: Aufbau Projekt-Board 9](#_Toc119703789)

[Abbildung 3: Projekt-Board mit Arduino und Steckbrett 10](#_Toc119703790)

[Abbildung 4: Montagebeispiele 11](#_Toc119703791)

[Abbildung 5: Tinkerkit Braccio Sevro-Shield 12](#_Toc119703792)

[Abbildung 7: Tinkerkit Braccio Servomotoren 13](#_Toc119703793)

[Abbildung 6: Tinkerkit Braccio Roboterarm - Bewegungspunkte 14](#_Toc119703794)

# Tabellenverzeichnis

[Tabelle 1: Arduino UNO R3 - Komponentenübersicht 6](#_Toc119703801)

[Tabelle 2: Arduino Student Kit - Zubehör 9](#_Toc119703802)

[Tabelle 3: Zuordnungstabelle - Servo und Bewegungspunkte 15](#_Toc119703803)

# Abkürzungsverzeichnis

DHBW *Duale Hochschule Baden-Württemberg*

ECC *Education Competence Centern*

EdCoN *Education Competence Network*

LED *light-emitting diode*

# Einleitung

*‚Wir behalten von unsern Studien am Ende doch nur das, was wir praktisch anwenden.‘*

*Von Johann Wolfgang von Goethe*

Dieses Zitat von Johann Wolfgang von Goethe spiegelt die Relevanz der Praxisnähe zur Theorie innerhalb eines Studiums wider. Mithilfe der praktischen Anwendungen kann Studierenden die Theorie nähergebracht und erklärt werden, wobei vor allem in technischen Studiengängen eine gewisse Praxisnähe mithilfe verschieden eingesetzter Technik realisierbar ist. Hierzu muss an Hochschulen und Universitäten die entsprechenden Mittel geschaffen werden, um Projekte mit den Studierenden umsetzen zu können. Dabei kann es aufgrund von Kosten oder der fehlenden Organisationsstruktur für die digitale Transformation zu Verzögerungen in der Umsetzung kommen.

Innerhalb der Dualen Hochschule Baden-Württemberg existiert ein Gesamtprojekt zur Beschleunigung der digitalen Transformation der Lehre. Dieses Projekt trägt den Namen „Education Competence Network“ beziehungsweise EdCoN und wird durch die Stiftung „Innovation in der Hochschullehre“ gefördert. Mithilfe der Stiftung ist es der Dualen Hochschule möglich die vorliegende Digitalisierungsstrategie und die gemachten Erfahrungen aus der Pandemie Zeit aufzuarbeiten und umzusetzen. Hierbei werden durch ein Team aus sogenannten professoralen Themenpaten der Hochschule verschiedene Konzepte zu Innovationsthemen der Lehre unter Einbezug der Studierenden entwickelt.

Die einzelnen Teams an allen Standorten der Dualen Hochschule Baden-Württemberg (DHBW) heißen Education Competence Centern (ECC) und werden auch als sogenannte Lehr-Lern-Labore bezeichnet. Jeder ECC beschäftigt sich grundsätzlich mit einem Hauptthema in der Lehre. Das Thema des ECC3 handelt von kollaborativen digitalen Szenarien des Forschenden Lernens. Bei diesen Szenarien sollen Studierende die Fähigkeit erlangen, eine selbstständige Analyse eines komplexen Sachverhalts durchführen und eigenständig unbekannte Aufgabenstellungen bearbeiten zu können. Auch das agile Arbeiten in Teams, kritisches Hinterfragen von Daten und das Treffen von kritischen Entscheidungen soll mittels des kollaborativen Forschenden Lernen den Studierenden nähergebracht werden. Dabei sind die Ziele zukunftsgerichtet, um die Studierenden auf zukünftige Sachverhalte im Berufsleben vorzubereiten und die notwendigen Fähigkeiten bereits während des Studiums zu fördern.

Im Zuge des Themas des ECC3 soll mithilfe dieser Studienarbeit eine Erstellung eines interdisziplinaren Lehr-Lern-Konzeptes für MINT und eine beispielhafte Umsetzung in der Automationstechnik erfolgen.

# Aufgabenstellung und Zielsetzung der Studienarbeit

Das Ziel der Studienarbeit ist es ein konkretes Beispiel für projekt- und problembasiertes Lernen im Team für einen beispielhaftgewählten Anwendungsfall zu konzeptionieren und umzusetzen. Der gewählte Anwendungsfall behandelt das in der Industrie vorhandene Pick-And-Place-Problem, welches im Rahmen der zu konzeptionierenden Teamaufgabe mithilfe des Arduino Student Kits und des Tinkerkit Braccio Roboterarmes realisiert.

Um diesen Anwendungsfall in der Praxis um zu setzen, wird im Rahmen dieser Studienarbeit eine konkrete Realisierung der Programmierung und ein entsprechender Aufbau der Hardware abgebildet. Einerseits dient dies für den unterrichtenden Dozenten als Beispiellösung. Aufgrund dessen ist es für die Studierenden möglich kreative Ansätze zu finden und flexible Lösungswege zu wählen. Somit fördert das Lernen mittels teambasierter Problemlösung den didaktischen Effekt und die Flexibilität und Kreativität der Studierenden.

Andererseits kann mithilfe der erstellten Beispiellösung den Studierenden eine Hilfestellung gewährleistet werden. Durch die Freiheit der Umsetzung dieser Teamaufgabe kann es dazu kommen, dass beispielsweise Komplikationen auftreten. Hierbei kann eine Gruppe von Studierenden Probleme in der Programmierung oder beim Aufbau der Hardware haben. In diesem Fall ist es möglich die Studierenden zu unterstützen und somit den Dozenten zu autorisieren Ratschläge zur Problemlösung oder sporadische Unterstützung leisten zu können.

# Aufbau des Projektes für die Studierenden

Das Projekt der Studierenden besteht darin eine funktionsfähige Hardware aufzubauen und eine entsprechende Programmierung dieser Hardware durchzuführen. Dabei wird den Studierenden eine konkrete Zielsetzung genannt, welche im darauffolgenden Abschnitt näher erläutert wird. Wichtig ist es, dass das Projekt viele Umsetzungsfreiheiten bietet, wobei eine festgelegte Hardware genutzt werden muss. So soll es möglich sein, dass die Studierenden in Teams viele Ideen und Lösungsansätze sammeln, um dann am Ende die verschiedensten Herangehensweisen und Umsetzungsmöglichkeiten präsentieren zu können.

## Projektzielsetzung der Studierenden

Im Vordergrund des Projektes liegt das Pick-And-Place-Problem, welches mithilfe des Tinkerkit Braccio Roboterarmes gelöst werden soll. Jedoch muss bei der Projektzielsetzung zwischen den technischen Umsetzungszielen und den didaktischen Zielen unterschieden werden.

Das projektbasierte Lösen dieses exemplarisch gewählten Anwendungsfalles soll einerseits die Programmier- und Hardwarekenntnisse der Studierenden fördern. Andererseits soll das Projekt eine Möglichkeit zur Weiterentwicklung der Teamfähigkeit und Problemlösefähigkeit bieten.

Die Studierenden bekommen eine Aufgabenstellung, welche flexibel gelöst werden kann, um Ihnen die Möglichkeit zu geben, Ihre Kompetenzen eigenständig weiterzuentwickeln. Die Grundidee ist es drei Modi für den Roboterarm zu programmieren. Dabei soll der erste Modus das zugrundeliegende Pick-And-Place-Problem lösen, indem der Roboter einen automatisierten Ablauf durchläuft. Die Einzelteile, welche bewegt und auf andere Teile gesteckt werden sollen, müssen vorab platziert werden, sodass der Roboter nach Aktivierung des Modus eine automatisierte Routine ablaufen lässt, ohne dabei zusätzliche Hilfestellungen zu bekommen oder beeinflusst zu werden. Der zweite Modus soll eine manuelle Steuerung des Roboters ermöglichen, damit dieser wie ein Greifarm fungiert. Dabei können die Studierenden die Hardware freiwählen, mithilfe derer die manuelle Steuerung realisiert wird. Der dritte Modus….

Wichtig ist es, dass die Studierenden bei der Realisierung der Modi eine Statusanzeige mittels verschieden farbiger LEDs (light-emitting diode) integrieren.

## Rahmenbedingungen für die Studierenden

Bei der Umsetzung des Projektes soll es den Studierenden möglich sein, viele eigenständige Entscheidungen treffen zu können und die eigene Kreativität in Absprache mit dem Team einfließen zu lassen. Dementsprechend dienen die Rahmenbedingungen für das Projekt zur Festlegung der grundlegenden Basis, mithilfe derer jedes Team die konkrete Umsetzung frei gestalten kann. Als Arbeitsgrundlage bezüglich der Hardwarerealisierung dient das Arudino Student Kit und der Tinkerkit Braccio Roboterarm. Die darin vorhandenen Teile sind verpflichtend zu verwenden, jedoch kann der Hardwareaufbau flexibel gewählt werden. Die einzelnen Hardwarekits, sowie dazugehörigen Komponenten werden im folgenden Kapitel näher erläutert.

Bezüglich der Programmierung sind prinzipiell keine Grenzen gesetzt. Auch hier dürfen die Studierenden ihre Kreativität und lösungsorientiertes Denken unter Beweis stellen. Aufgrund der zur Verfügung stehenden Hardware ist die zu nutzende Programmiersprache festgelegt. Der Arduino kann über die Arduino IDE mithilfe von C++ programmiert werden. Durch die fast vollständige Abwärtskompatibilität von der Programmiersprache C++ zu C kann das Arduino-Programm ebenfalls mit C-Code erstellt werden. Jedoch kann die hier verwendete Programmiersprache auch als Arduino spezifische Sprache bezeichnet werden, da es bei der Verwendung von C++ beziehungsweise C kleine Abweichungen gibt, welche aufgrund der Gegebenheit des Arduinos notwendig sind und die Programmierung des Boards vereinfachen sollen. Mithilfe welches Frameworks der Arduino programmiert wird, steht den Studierenden frei zur Wahl. Dies begründet sich darin, dass so eigene Entscheidungen aufgrund von bisherigen Erfahrungen mit den Frameworks getroffen werden können und die Framework-Auswahl bezüglich der Programmierqualität keine Auswirkungen hat.

Des Weiteren soll das Projekt im Rahmen einer Teamarbeit stattfinden, weshalb je nach Kursgröße ein Team aus 3 bis 4 Studierenden besteht. Hierbei sollten primär dreier Gruppen gebildet werden und nur in Ausnahmefällen vierer Gruppen.

## Verwendete Hardware: Arduino und Braccio

Um das Projekt durchzuführen, steht den Studierenden die, im Folgenden beschriebene, Hardware zur Verfügung. Hierbei sind die Hauptkomponenten das Arduino Student Kit und der Tinkerkit Roboterarm Braccio.

Innerhalb des Arduino Student Kits ist das Arduino UNO R3 Board enthalten. Mithilfe der erstellten Software kann die Hardware auf der Platine angesteuert und im Mikrocontroller des Arduino-Boards gespeichert werden. Durch dieses Mikrocontroller-Entwicklungsboard können verschiedene Schaltkreise angesteuert und angeschlossen werden.

In der folgenden Abbildung ist der beigefügte Arduino UNO R3 abgebildet, wobei der Aufbau des Boards mit Buchstaben markiert ist.

Ein Bild, das Text, Elektronik enthält.

Automatisch generierte Beschreibung

Abbildung 1: Arduino UNO R3 - Board

In der untenstehenden Tabelle werden die entsprechenden Markierungen beziehungsweise die markierten Komponenten aus der Abbildung des Arduino UNO R3 beschrieben. Dies dient zum Allgemeinen Verständnis des Boards.

| Bezeichnung | Komponente | Beschreibung |
| --- | --- | --- |
| A | Reset-Knopf | Setzt den Mikrocontroller zurück. |
| B | TX und RX LEDs | Diese LEDs zeigen die Kommunikation zwischen Arduino UNO R3-Board und Computer an. Sie flackern, wenn Kommunikation stattfindet. |
| C | Digitale Pins | Diese Pins können mit digitalen Ein- oder Ausgabegeräten verbunden werden. Pins mit dem ~-Symbol können auch mit analogen Geräten verwendet werden. |
| D | Power LED | Zeigt an, dass der Arduino UNO R3 mit Strom versorgt wird. |
| E | ATmega Mikrocontroller | Der Chip, der den Sketch speichert und ausführt. |
| F | Analoge Pins | Pins zum Anschluss von Geräten, die analoge Signale zurückgeben. |
| G | Ground und 5V Pins | Über diese Pins können Stromkreise mit Masse und 5 Volt versorgt werden. |
| H | Stromanschluss | Darüber kann der Arduino UNO R3 mit Strom versorgt werden, wenn er nicht an einen USB-Anschluss angeschlossen ist. Der Arduino UNO R3 arbeitet mit Spannungen von 7-12 Volt. |
| I | USB-Anschluss | Dient zur Stromversorgung, zum Hochladen von Programmen und zur Kommunikation mit dem Computer. |

Tabelle : Arduino UNO R3 - Komponentenübersicht

Zusätzlich sind innerhalb des Student Kits weitere elektronische Komponenten und ein ergänzendes Steckbrett. Dieses Brett dient als Hilfsmittel, um verschiedene elektronische Schaltkreise lötfrei aufbauen zu können. Ebenfalls sind für diese Steckverbindungen bereits zugeschnittene und gebogene Drähte im Kit enthalten. Auch eine 9 Volt Batterie und ein entsprechender Batterieanschluss sind beigelegt. Somit kann der Arduino Uno R3 über einen Anschluss mithilfe des erwähnten Steckbrettes betrieben werden. Die weiteren elektronischen Komponenten werden in der folgenden Tabelle aufgelistet.

| Komponente | Anzahl | Beschreibung |
| --- | --- | --- |
| Kondensator | 2 x 100 **μ**F | Ein Kondensator speichert die elektrische Energie eines Stromkreises und kann dies wieder abgeben. Der Kondensator wird aufgeladen, wenn die Spannung im Stromkreis höher ist als die im Kondensator gespeicherte. Somit fließt Strom in den Kondensator. Ist die Spannung im Stromkreis niedriger, wird die gespeicherte Ladung freigegeben und den Kondensator entladen. Kondensatoren werden primär als Ausgleich bei Spannungsänderungen genutzt. |
| Drahtbrücke | 100 | Die einzelnen Drahtbrücken dienen dazu, das Steckbrett mit dem Arduino Uno R3 Board und die Komponenten auf dem Steckbrett untereinander zu verbinden. |
| Leuchtdiode | 20  (5 Grüne,  5 Blaue,  5 Gelbe,  5 Rote) | Leuchtdioden sind eine spezielle Art von Dioden, die aufleuchtet, sobald Strom hindurchfließt. Dioden lassen den Strom nur in eine Richtung fließen. |
| Piezo-Summer | 1 | Mithilfe dieses elektrischen Bauteiles können Vibrationen erkannt und Geräusche beziehungsweise Melodien erzeugt werden. |
| Fototransistor | 1 | Je nach einfallender Lichtmenge steuert der Transistor den elektrischen Strom. |
| Potentiometer | 2 | Hierbei handelt es sich um einen variablen Widerstand mit drei Anschlüssen, wobei zwei der Drähte mit den Enden eines festen Widerstands verbunden sind. Der Anschluss in der Mitte ist mit einem Schleifkontakt verbunden, wobei sich dieser über den Widerstand bewegt. Damit wird der Widerstand in zwei Teile geteilt. Ein Potentiometer dient zum Einstellen der Spannung in einem Stromkreis. Ein Beispiel ist der Lautstärkeregler eines Radios. |
| Stromkabel | 2 | Primär werden diese Überbrückungsleitungen dazu verwendet das Steckbrett mit Spannung und Masse des Arduino UNO R3 Boards zu verbinden. |
| Tastschalter | 5 | Beim Drücken dieses Schalters wird der Stromkreis geschlossen und beim Loslassen wieder geöffnet. Mithilfe des Arduino UNO R3 Boards kann der Taster als Eingabegerät genutzt und somit die Ein- und Aus-Signale erkannt werden. |
| Widerstand | 14  (5 x 220 Ω,  5 x 560 Ω,  1 x 1 kΩ,  2 x 4,7 kΩ,  1 x 10 kΩ) | Ein Widerstand behindert den Stromfluss einer Schaltung. Somit wird zusätzlich die Spannung und der Strom des Schaltkreises geändert. Die farbigen Streifen auf dem Widerstand stehen für den Widerstandswert und die dazugehörige Toleranz. |
| Servomotor | 1  (Typ:  SM-S2309S) | Hierbei handelt es sich um ein Getriebemotor, der sich um 180 Grad drehen und über elektrische Impulse durch den Arduino UNO angesteuert werden kann. |
| USB-Kabel | 1 | Über dieses Kabel kann das Arduino UNO R3 Board mit einem Computer verbunden und kompilierte Programme auf das Board übertragen werden. Zusätzlich wird der Arduino über das Kabel mit Strom versorgt. |

Tabelle : Arduino Student Kit - Zubehör

Außerdem gibt es für das Student Kit ein Projekt-Board, welches den Arduino UNO R3 mit dem Steckbrett zu einer Hardware zusammenfasst. Mit diesem vorgeschnittenen Projekt-Board auf Kunststoffbasis soll das Erstellen von Schaltkreisen erleichtert werden, da das Steckbrett und der Mikrocontroller auf dem Board nahe zusammen liegen. In der folgenden Abbildung ist die Zusammensetzung des Projekt-Boards in einzelnen Schritten dargestellt.

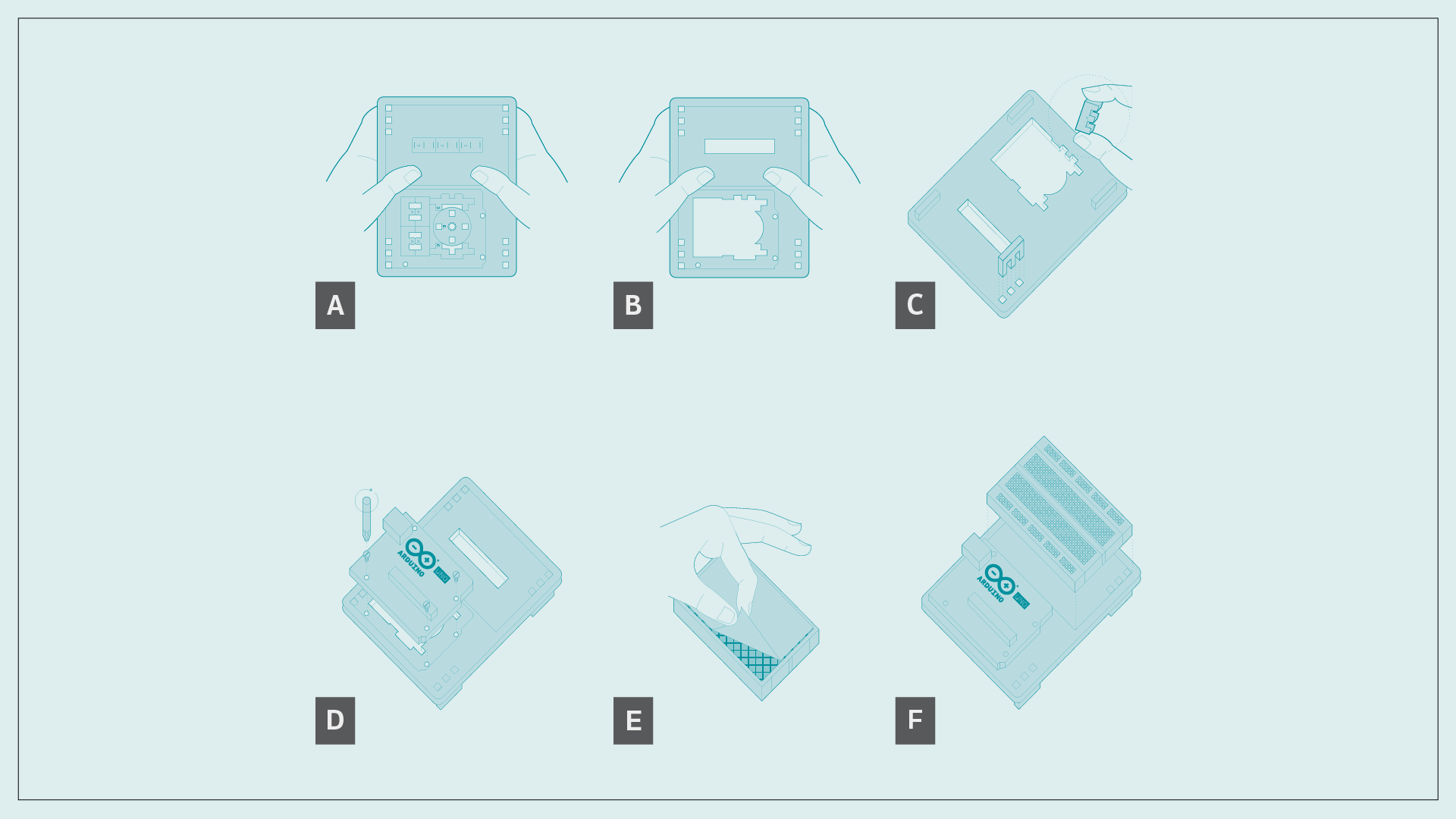


Abbildung : Aufbau Projekt-Board

Der, mit ‚A‘ gekennzeichnete, Schritt bildet das Projekt-Board im Originalzustand ab. Im folgenden Schritt ‚B‘ sind die Kleinteile herausgetrennt, welche anschließend wieder in das Kit zurückgelegt werden soll, da diese Teile für verschiedene Projekte in den online verfügbaren Arduino-Kursen vorgesehen sind. Die dabei entfernt Teile werden mit verschiedenen Buchstaben gekennzeichnet. Der darauffolgende Schritt ‚C‘ benötigt vier der Kleinteile, welche mit ‚A‘ markiert sind. Werden diese Teile in die Löcher an den Ecken des Projekt-Boards befestigt, entstehen Füße für das Board, um einen Abstand zwischen Board und Tisch zu kreieren. Im Schritt ‚D‘ wird das Arduino UNO R3 Board mit den beigefügten drei Schrauben an dem Projekt-Board befestigt. Dabei sollen die Schrauben durch die Arduino Platine und die Basisplatte beziehungsweise das Projekt-Board geführt werden und mithilfe der drei mitgelieferten Muttern können die Schrauben gesichert werden. Hierbei ist zu beachten, dass die Muttern nicht zu fest angezogen werden sollen. Die folgenden zwei Schritte ‚E‘ und ‚F‘ beschäftigen sich mit der Befestigung des Steckbrettes. Hierzu muss die Schutzfolie auf der Rückseite des Steckbrettes entfernt werden, sodass das Steckbrett auf der Basisplatte neben den Arduino UNO R3 geklebt werden kann. Das Steckbrett ist so auszurichten, dass sich das Loch ‚1a‘ in der Nähe des Reset-Knopfes auf dem Arduino befindet. Das fertige Projekt-Board sieht entsprechend der folgenden Abbildung aus.

Ein Bild, das Text, Elektronik, Screenshot enthält.

Automatisch generierte Beschreibung

Abbildung : Projekt-Board mit Arduino und Steckbrett

Nachdem das Arduino Student Kit ausführlich dargelegt wurde, wird auf den folgenden Seiten der Roboterarm Braccio des Tinkerkit erklärt.

Der Braccio ist ein voll funktionstüchtiger Roboterarm, welcher für den Tischeinsatz entwickelt und als Montagesatz geliefert wird. Aufgrund dessen kann der Roboterarm flexibel eingesetzt und angepasst werden. Wie in der folgenden Abbildung dargestellt ist, kann der Aufsatz des Roboterarmes beispielsweise durch eine Kamera, ein Smartphone oder ein Solarmodul ersetzt und die Länge des Armes angepasst werden.

Ein Bild, das LEGO, Himmel, Spielzeug enthält.

Automatisch generierte Beschreibung

Abbildung : Montagebeispiele

Die maximale Höhe, die der Roboterarm erreichen kann, sind 52 Zentimeter, jedoch kann dieser in einer Distanz von maximal 80 Zentimeter arbeiten. Die Basis hat einen Durchmesser von 14 Zentimeter und die Greifzange kann bis zu 9 Zentimeter weit geöffnet werden. Innerhalb einer Distanz von 32 Zentimeter kann der Roboterarm ein Gewicht von etwa 150 Gramm transportieren. Das maximale Hebegewicht beträgt 400 Gramm bei minimaler Distanz-Konfiguration des Roboterarmes.

Außerdem wird eine Arduino Platine und das mitgelieferte Braccio Servo-Shield benötigt, um den Roboterarm betreiben zu können. Das Shield ermöglicht es, dass das Arduino Board auf die einzelnen Servos zugreifen kann und der Roboterarm somit programmiert und angesteuert werden kann. Hierzu benötigt das Shield eine Versorgungsspannung von 5 Volt. Wie in der folgenden Abbildung dargestellt ist, wird das Braccio Servo-Shield auf den Arduino gesteckt, wobei das Shield zusätzliche Pins für die Ansteuerung der Servos bietet.

Ein Bild, das LEGO, Spielzeug, ausgestaltet enthält.

Automatisch generierte Beschreibung

Abbildung : Tinkerkit Braccio Sevro-Shield

Die einzelnen Steckverbindungen zu den Motoren sind mit dem Buchstaben ‚M‘ und einer Zahl, aufsteigend von ‚1‘, gekennzeichnet. Auch auf dem Shield ist markiert, welcher der Motoren an die jeweilige Verbindung angeschlossen werden soll. Beim Anschließen der Motoren ist es wichtig, wie diese gesteckt werden, da die einzelnen Kabel, wie die Abbildung zeigt, die Spannungsversorgung, das Signal und die Erdung der einzelnen Motoren mit den entsprechenden Pins des Shields verbinden. Die zwei kleinen Plastikausbuchtung an der Steckverbindung der Motoren dienen beim Verbinden des Motors und des Shields zur Orientierung. Diese zwei Ausbuchtungen an der Motorsteckverbindung muss die Plastikerhöhung des Shields rechts und links umschließen, damit die drei Kabel und entsprechenden Pins richtig angeschlossen sind.

Die Buchse vorne am Braccio Servo-Shield dient zum Anschluss der Spannungsversorgung. Hierzu ist im Arduino Braccio Tinkerkit ein entsprechendes Kabel mit Netzteil enthalten.

Da der Roboterarm als Montagesatz im Tinkerkit geliefert wird, ist dieser in seine einzelnen Komponenten aufgeteilt. Im Kit sind 21 Plastikteile und 6 Servomotoren enthalten. Mithilfe der beigefügten Schrauben, Muttern und des magnetischen Schraubendrehers ist es möglich den Roboterarm Schritt für Schritt, wie in der beigelegten Anleitung erklärt, aufzubauen.

Die in der untenstehenden Abbildung gezeigten Servomotoren sind die zwei Typen, welche im Tinkerkit mitgeliefert werden.

Ein Bild, das Adapter enthält.

Automatisch generierte Beschreibung

Abbildung : Tinkerkit Braccio Servomotoren

Der linke Motor ist vom Typ „SR 418“ und der rechte Servo ist vom Typ „SR 312“. Hierbei ist zu berücksichtigen, dass 4 Servos desselben Typs des links abgebildeten Motors und zwei des rechts abgebildeten Motors im Tinkerkit enthalten sind. Die zwei kleineren Motoren werden für die Bewegung der Greifzange benötigt, wobei einer diese öffnen, beziehungsweise schließen und der andere Servo die Zange um die eigene Achse drehen kann.

Des Weiteren sind innerhalb des sogenannten Quick Start Guides beziehungsweise der Anleitung drei Beispielprogramme aufgelistet, welche über die ArduinoIDE aufgerufen und auf das Arduino Board geladen werden können.

Mithilfe der Bibliothek „Servo.h“ ist es möglich mit dem Arduino Board einzelne Servomotoren anzusteuern. Hierbei kann das integrierte Getriebe oder eine Welle präzise gesteuert werden, wobei die Welle bei Standardservos im Winkel von 0 bis 180 Grad variiert werden kann. Auch die Geschwindigkeit der Wellendrehung kann bei Servos angepasst werden, sofern es sich um Servos mit kontinuierlicher Drehung handelt. Diese Bibliothek unterstützt maximal 12 Motoren auf dem Großteil der Arduino Boards und maximal 48 Motoren auf dem Arduino Mega. Die andere Bibliothek, welche für die Steuerung des Roboterarmes benötigt wird, heißt „Braccio.h“ und ermöglicht die Steuerung der einzelnen Gelenkpunkte des Tinkerkit Braccio Roboterarmes mithilfe vereinfachter Aufrufe.

Für die Programmierung des Roboterarmes ist es wichtig zu wissen, wie der aufgebaute Braccio Roboterarm aussieht und welche die Bewegungsmöglichkeiten der einzelnen Servos existieren. Hierfür werden die Bewegungspunkte spezifisch benannt und der entsprechenden Nummer auf dem Verbindungsstecker der Motoren zugeordnet. In der folgenden Abbildung ist der Tinkerkit Braccio Roboterarm aufgebaut und die einzelnen Bewegungspunkte markiert.

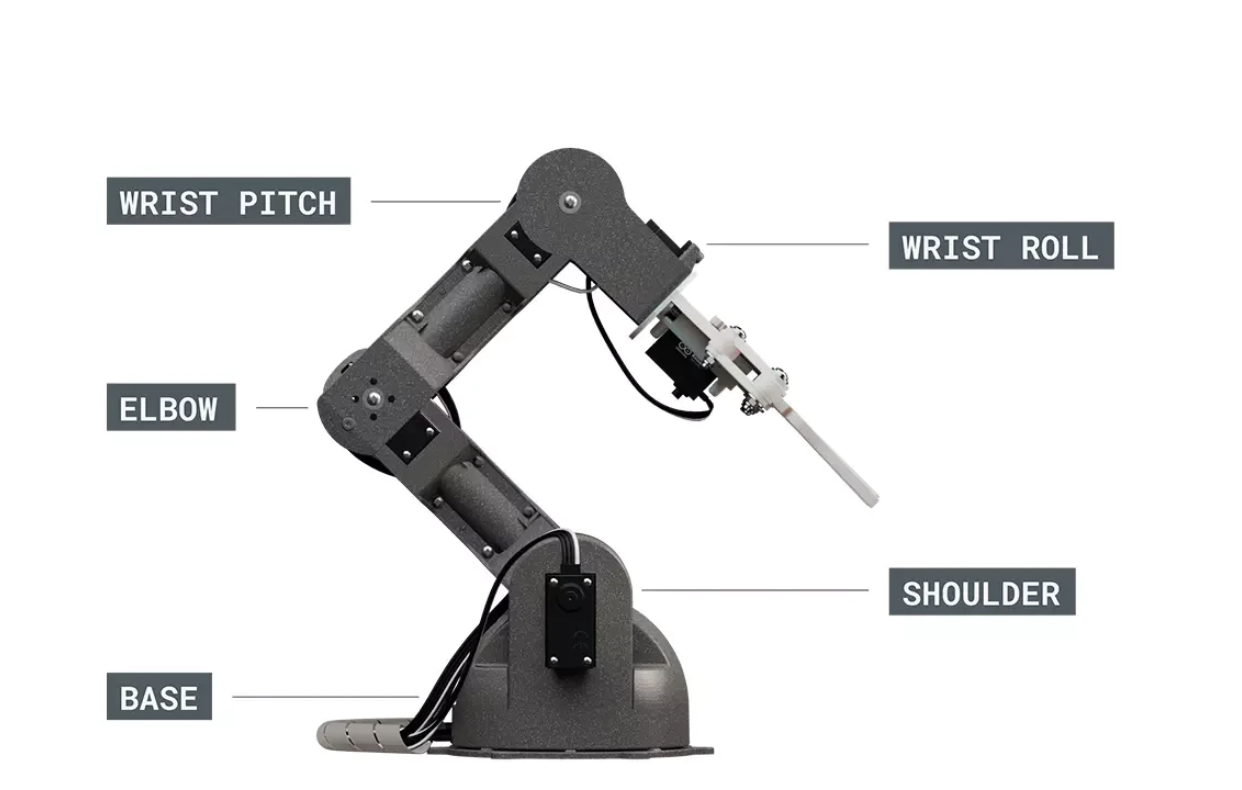


Abbildung : Tinkerkit Braccio Roboterarm - Bewegungspunkte

Der Roboterarm besitzt wie in der Abbildung dargestellt 5 Bewegungspunkte. Mithilfe derer die Positionierung und somit die einzelnen Servos gesteuert werden. Des Weiteren gibt es noch einen Motor an der Zange des Roboters, welcher, wie bereits erwähnt, die Greifzange öffnet und schließt. Dieser ist in der Abbildung nicht markiert, jedoch ist der Servo bei diesem Aufbau erkennbar. Für die Nutzung der Servomotoren ist die folgende Zuordnungstabelle wichtig, da mit dieser ersichtlich ist, welcher Motor an den einzelnen Bewegungspunkten platziert ist und wie diese bewegt werden können.

|  |  |  |  |
| --- | --- | --- | --- |
| Motor | Bewegungspunkt (englisch) | Bewegungspunkt (deutsch) | Bewegungsmöglichkeit |
| M1 | Base | Basis | 0 bis 180 Grad |
| M2 | Shoulder | Schulter | 15 bis 165 Grad |
| M3 | Elbow | Ellbogen | 0 bis 180 Grad |
| M4 | Wrist pitch | Handgelenksneigung | 0 bis 180 Grad |
| M5 | Wrist roll | Handgelenksrotation | 0 bis 180 Grad |
| M6 | Gripper | Greifzange | 10 bis 73 Grad |

Tabelle : Zuordnungstabelle - Servo und Bewegungspunkte

## Beispiellösung der Programmierung und des Hardwareaufbaus

Text

# Didaktische Lehrmittel

Text

## Anforderungen an didaktische Konzepte für moderne Lehrformen

Text (Stichworte: Projektbasiertes Lernen, Peer-to-Peer Learning, seminaristisches Lernen, forschendes Lernen)

## Selektierung der Lehrmittel für die Zielgruppe

Text

# Erstellung eines didaktischen Lehr- und Lernkonzeptes

Text

# Praktische Umsetzung mithilfe des Konzeptes

Text

# Dokumentation und Hilfestellung für den Projekteinsatz

Text

# Kritische Reflexion und Bewertung

Text

# Literaturverzeichnis

**Im aktuellen Dokument sind keine Quellen vorhanden.**