

Bernburg
Dessau
Köthen



Hochschule Anhalt
Anhalt University of Applied Sciences



Fachbereich
Elektrotechnik, Maschinenbau
und Wirtschaftsingenieurwesen

BELEGARBEIT

Digitalisierung von Maschinen

Ashly Fonseca

Vorname Nachname #1

Informatik, Medien &
Spieleentwicklung, 2020, 5013757

Studiengang, Matrikel, Matrikelnummer

Jeremy Becker

Vorname Nachname #2

Informatik, Medien &
Spieleentwicklung, 2020, 5014725

Studiengang, Matrikel, Matrikelnummer

Thema:

RC-Car mit Gyrosteuerung

Prof. Dr. S. Twieg

Dozent

23.09.2022

Abgabe am

Selbstständigkeitserklärung

Hiermit erkläre/n ich/wir, dass die Arbeit selbständig verfasst, in gleicher oder ähnlicher Fassung noch nicht in einem anderen Studiengang als Prüfungsleistung vorgelegt wurde und keine anderen als die angegebenen Hilfsmittel und Quellen, einschließlich der angegebenen oder beschriebenen Software, verwendet wurden.

Köthen, 24.09.2024

Ort, Datum

Unterschrift/en der/des Studierenden

Inhaltsverzeichnis

1	Motivation und Zielsetzung	Fehler! Textmarke nicht definiert.
1.1	Einleitung in die Thematik	iv
1.2	Zielsetzung der Arbeit	iv
2	Theoretische Grundlagenbetrachtung Steuerungstechnik	vi
2.1	RC-Car	Fehler! Textmarke nicht definiert.
2.1.1	RC-Car, Aufbau	Fehler! Textmarke nicht definiert.
2.1.2	RC-Car, Programmcode	Fehler! Textmarke nicht definiert.
2.2	Gyro-Handschuh	Fehler! Textmarke nicht definiert.
2.2.1	Gyro-Handschuh, Aufbau	Fehler! Textmarke nicht definiert.
2.2.2	Gyro-Handschuh, Gyrosensor	Fehler! Textmarke nicht definiert.
2.2.3	Gyro-Handschuh, Programmcode	
2.3	MQTT	Fehler! Textmarke nicht definiert.
2.4	Verknüpfung der Komponenten	Fehler! Textmarke nicht definiert.
3	Projektanalyse	Fehler! Textmarke nicht definiert.
4	Zusammenfassung und Ausblick	Fehler! Textmarke nicht definiert.
	Symbol- und Indexverzeichnis	Fehler! Textmarke nicht definiert.
	Abbildungsverzeichnis	xiii
	Tabellenverzeichnis	Fehler! Textmarke nicht definiert.
	Literaturverzeichnis	xiii

1 Motivation und Zielsetzung

1.1 Einleitung in die Thematik

Das Patent für „Fahrzeug mit Gasmotorenbetrieb“, welches Carl Benz am 29. Januar 1886 erhielt, gilt als Geburtsurkunde für das Automobil. Die Mobilität des Menschen wird von hieran stetig revolutioniert und verbessert. Die Fortbewegung wurde schneller, energieeffizienter und sicherer. Das Auto ist in unserer heutigen Gesellschaft überall vertreten. Schnellere Autos führten zum Rennsport, sichere Autos zu neuer Kriegsführung und natürlich auch weniger Toten im normalen Straßenverkehr und Energieeffizienz zu neuen Motoren mit weniger Verbrauch oder gar neuen Antriebsmöglichkeiten. Mit der Entwicklung von Computern konnten auch Autos noch stark weiterentwickelt werden und bieten heutzutage ein breites Spektrum an Funktionen, Assistenten und sonstigen Spielereien, wo selbst die größten Fantasien der Anfangszeit nicht mit gerechnet hätten. Was vor 50 Jahren in Sci-Fi Serien zu sehen war, ist heute durchaus Realität. Mittlerweile sind wir an einem Punkt angelangt, wo der Mensch das Steuerrad verlässt und das Auto selber fahren lässt, wodurch der Begriff Automobil („Selbstbeweger“) seine volle Bedeutung erhält. Wir wollen uns in unserem Projekt auch weg vom Steuerrad bewegen und uns mit einer alternativen Steuerung über Handbewegungen beschäftigen. Diese Steuerungsart bietet den Vorteil, dass eine Hand zum Steuern des Autos vollkommen ausreicht. Bereits heute lassen sich somit Drohnen in der Luft bewegen oder Charaktere und Autos in Videospielen über Gyroskope in den Kontrollern. Mit dieser Technik ließen sich CGI-freie Filme produzieren, welche trotzdem futuristische Einblicke bringen.

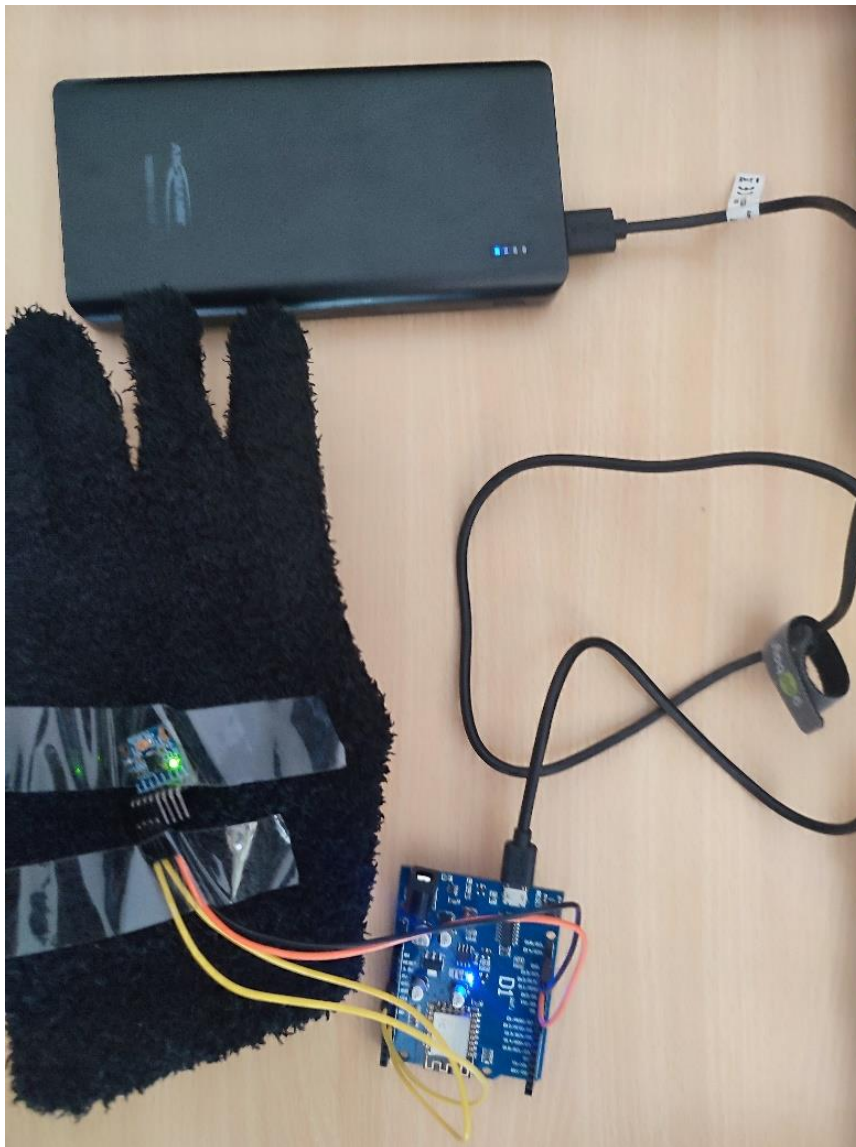
Neue Techniken erbringen auch neue Gefahren mit sich. In der Fahrschule hieß es, dass bei einem Wildunfall das Lenkrad fest in die Hand genommen werden müsste, da sonst die Räder abdriften könnten. Hier haben wir natürlich nicht die manuelle Kontrolle wie bei einem Lenkrad. Die Räder müssten volle Stabilität und Vollkommene Kontrolle gewährleisten. Zudem wäre es durchaus eine vollkommen ungewohnte Art der Steuerung und man müsste ggf. über eine Ablage für den Arm oder die Hand nachdenken, da die permanente Position der Hand, wie zum Beispiel auf längeren geraden Strecken zu Schmerzen führen kann. Es ließe sich beispielsweise mit einem flexiblen Pad erweitern und den Komfort erhöhen. Zudem könnte dieses Pad und die Position der Hand und entsprechend der Gyrosteuerung Notfallsysteme aktivieren. Ein Gyroskop (siehe 2.2.2) direkt am Körper bietet neue Möglichkeiten zum Personenschutz und Rettung.

1.2 Zielsetzung der Arbeit

In unserem Projekt wollen wir mit einem Handschuh, an welchem eine Gyrosteuerung befestigt ist, ein RC-Car bewegen können. Das remote-controlled Car, zu deutsch ferngesteuerte Auto, soll je nach Lage des Gyrosensors die Fahrtrichtung ändern. Dies bezieht sich sowohl auf vorwärts und rückwärts, als auch die Lenkung nach links oder rechts. Unser Projekt dient als Grundlagenforschung zur Gyrosteuerung von Autos im Modell-Maßstab. Wir wollen über MQTT Befehle von unserem Handschuh mit der Steuerung an das RC-Car senden und damit die Bewegung erreichen. Gegebenenfalls können über MQTT auch die Neigungswerte übergeben werden, wodurch eine Differenzierung in der Geschwindigkeit sowie dem Ausschlag beim Lenken gemacht

werden kann. Wie bereits erwähnt wollen wir über MQTT die Werte der Steuereinheit, unserem Gyrosensor, an das Auto senden. Der Arduino des Autos soll, entsprechend der gesendeten Werte, diese in Bewegung übersetzen, indem die Motoren durch die Flussrichtung des Stroms in die entsprechende Richtung bewegt werden.

Das in der Einführung genannte Pad kann hier mit einem Kissen simuliert werden, jedoch weitere aufgeführte Sicherheitsmaßnahmen für den Straßenverkehr lassen sich nicht in diesem Maßstab umsetzen. Eine Idee zur Erweiterung des Projektes wäre ein Gyrosensor, welcher einen Aufprall ermittelt und alles deaktiviert, bis das Auto neu gestartet wird.



2 Theoretische Grundlagenbetrachtung Steuerungstechnik

2.1 RC-Car

2.1.1 Aufbau

Wir arbeiten mit dem steuerbaren „ELEGOO UNO R3 Project Smart Robot Car Kit“. Innerhalb von 1,5 Stunden haben wir das RC-Car zusammengebaut. Hierfür wurden ein Akku, ein Arduino D1, vier DC-Motoren, ein L298N Motor Driver Board mit einer Vielzahl an Kabeln über Platten, Streben und Schrauben verbunden. Der Arduino ist mit dem Motor Driver verbunden und dieser mit den Motoren, welche die Räder antreiben. Der Akku betreibt alle elektrischen Bauteile, wie die Motoren, welche separat die Räder betreiben können. Der Arduino ist unser Board-Computer. Er empfängt alle, über MQTT gesendeten Daten und übersetzt diese für das Motor Driver Board. Dieses Board dient der Stromverteilung an die Motoren, da der Arduino nicht die nötige Spannung händeln kann. Auf dem Arduino läuft das Entsprechende Programm zur Übersetzung und Interpretation der empfangenen Daten. Unser RC-Car (Abb. 2.x) ist die Minimal-Version des vollständigen Autos (Abb. 2.x).



Abbildung 0.1: ELEGOO UNO R3 Project Smart Robot Car [Car]

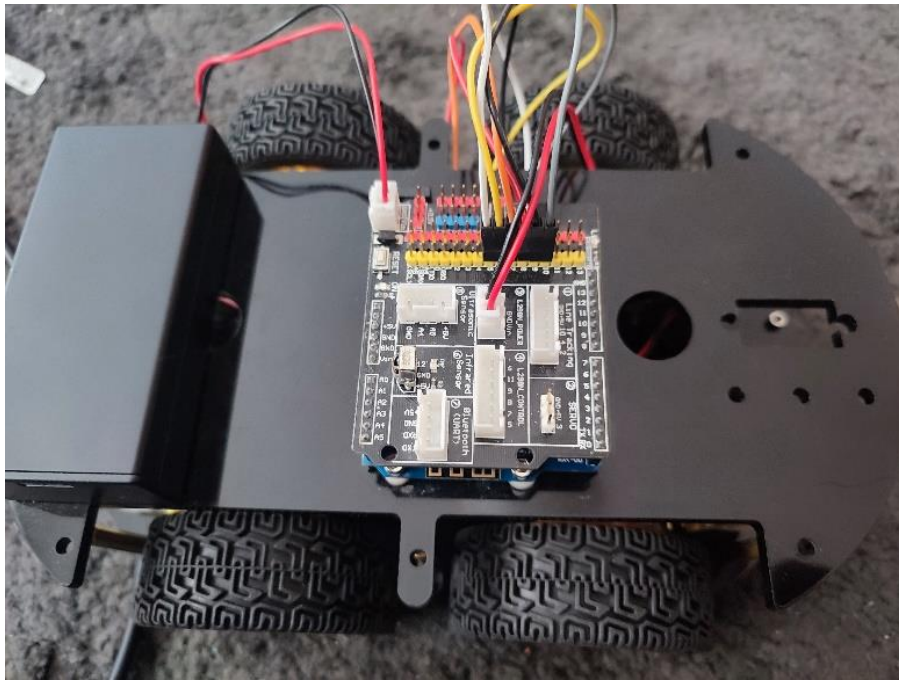


Abbildung 0.2: RC-Car

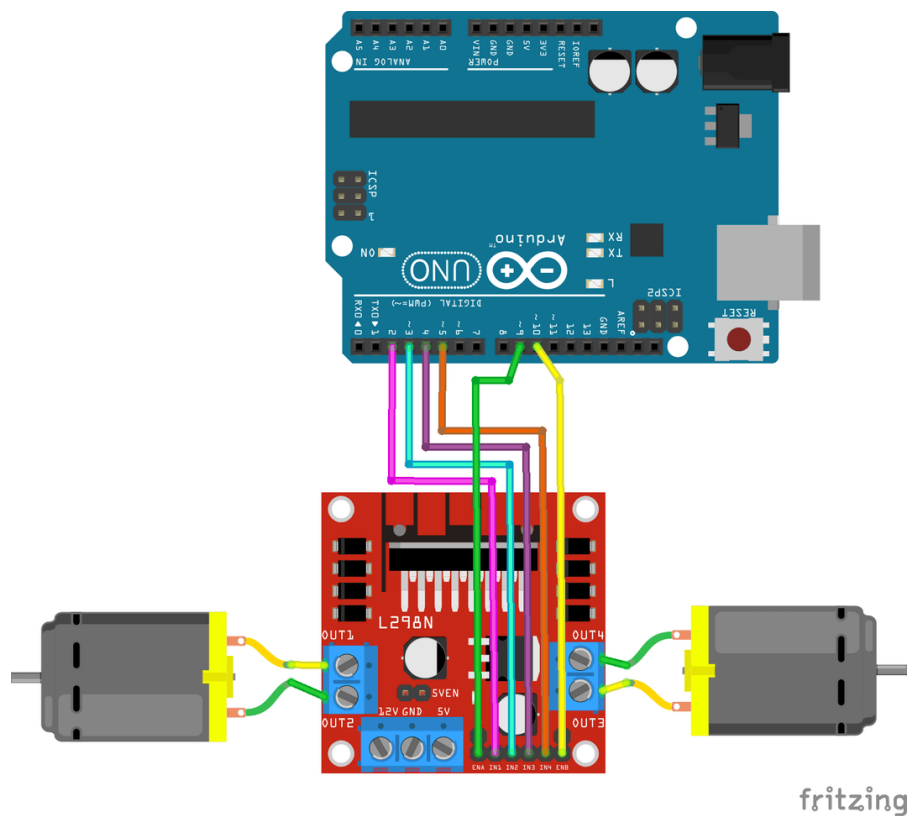


Abbildung 0.2: Aufbau der Steuerung des RC-Cars [Arduino1]

2.1.2 Programmcode

[Anhang: RC-Car]

Unser Arduino wird zunächst mit dem WLAN verbunden, indem die SSID und das Passwort im Quellcode eingerichtet wird. Es wird konstant überprüft, ob die WLAN-Verbindung bestehen ist. Sofern diese abbricht, wird er neu verbunden. Die Broker ID und der Broker Port wird eingestellt und die MQTT-Verbindung eingerichtet (siehe 2.3 MQTT). Nach erfolgreicher Verbindung mit MQTT wird der Arduino in einem Channel subscribed und überprüft, ob neue Werte empfangen wurden. Die Werte werden ausgelesen und entsprechend werden die Funktionen „goAhead()“, „goBack()“, „goLeft()“, „goRight()“ und „stopRobot()“ ausgeführt. Wurde „goAhead()“ oder „goBack()“ ausgeführt, wird nach einer Sekunde ohne neue Eingabe der Befehl „stopRobot()“ ausgeführt. Für „goLeft()“ und „goRight()“ jeweils nach 0,25 Sekunden.

Beim RC-Car werden die Pins den einzelnen Motoren zugewiesen, Output, und ENA sowie ENB werden die Geschwindigkeit der Motoren auf der rechten sowie linken Seite zugewiesen und fungieren als Input, beziehungsweise Prüfwert. Beispiel des „goAhead()“ werden IN2 und IN4 auf High eingestellt und IN1 und IN3 auf Low eingestellt. Dies resultiert darin, dass sich die Motoren auf der linken sowie rechten Seite nach vorne bewegen. Die Flussrichtung des Stroms entscheidet über die Laufrichtung der Motoren. Wären alle Einstellungen auf Low oder High würde es in Stillstand resultieren.

2.2 Gyro-Handschuh

2.2.1 Aufbau

Unser Gyro-Handschuh besteht aus zunächst einem handelsüblichen Handschuh sowie einem GY-521 Gyroskop, einem Arduino Uno und einem Akku. Das kompakte GY-521 Modul basiert auf dem MPU-6050 NEMS Chip, und vereint ein 3-Achsen Gyroskop und ein 3-Achsen Beschleunigungsmesser mit einem digitalen Bewegungs-Prozessor. Das Gyroskop, welcher relativ lose auf dem Handschuh befestigt ist, ist mit den Sender-Arduino verbunden. Alle Elemente werden mit einem Akku betrieben.

In unserem Ersten Ansatz zum „Handschuh“ wollten wir unsere Smartphones nutzen. Diese verfügen bereits seit einigen Jahren präzise Gyroskope. Leider konnten wir nicht an die Daten gelangen, welcher wir zum Steuern des RC-Cars gebraucht hätten. Zudem wäre die Übermittlung der Daten weitaus umständlicher gewesen, da die entsprechende App alle nötigen Berechtigungen von Android benötigt.

Das verbaute Gyroskop auf dem Handschuh bietet Erweiterungsmöglichkeiten zur Messung von Luftdruck, Himmelsrichtung, o.ä. Hierbei könnte man einen Ausblick auf Geschicklichkeitsspiele machen, wobei anhand der Himmelsrichtung das Auto gesteuert wird.

2.2.2 Gyrosensor

Ein Gyrosensor, Gyroskop oder auch Kreiselinstrument genannt, ist ein Lage- und Beschleunigungssensor. Er reagiert bereits auf kleine Veränderungen in seiner Lage, Beschleunigung und Drehbewegung. Das Prinzip des Sensors besteht auf Masseträgheit. Diese bezeichnet den Widerstand eines starren Körpers gegen eine Drehbeschleunigung. Es gibt verschiedene Funktionsweisen von Gyrosensoren. Bereits heute werden solche Sensoren in Autos verwendet, um in Navigationsgeräten die Richtungsänderung anzugeben. In anderen technischen Geräten, wie beispielsweise Smartuhren zur Erfassung von Schrittdaten oder in Mobilgeräten zur Lagebestimmung für beispielsweise Videospielen oder der Umstellung von Hoch- auf Querformat.

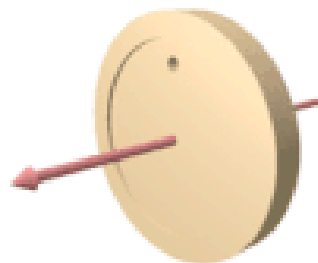


Abbildung 0.3: Kreiselinstrument in Bewegung (rot: Kreiselachse, grün: Achse äußere Kraft, blau: Achse Ergebnis) [Gyro1]

Gyrosensoren werden zunehmend als MEMS-Sensoren ausgeführt. „Micro Electro Mechanical Systems“ (vgl. NEMS, Nano) sind in unserer modernen Technik nicht wegzudenken. Die Umwandlung der physikalischen Größen basiert auf statischen, magnetischen, optischen, piezoelektrischen oder elektrothermischen Verfahren. Die Vorteile der MEMS-Sensoren liegen in der Kompaktheit, der Produktionstechnik und der Integration auf einem Chip.

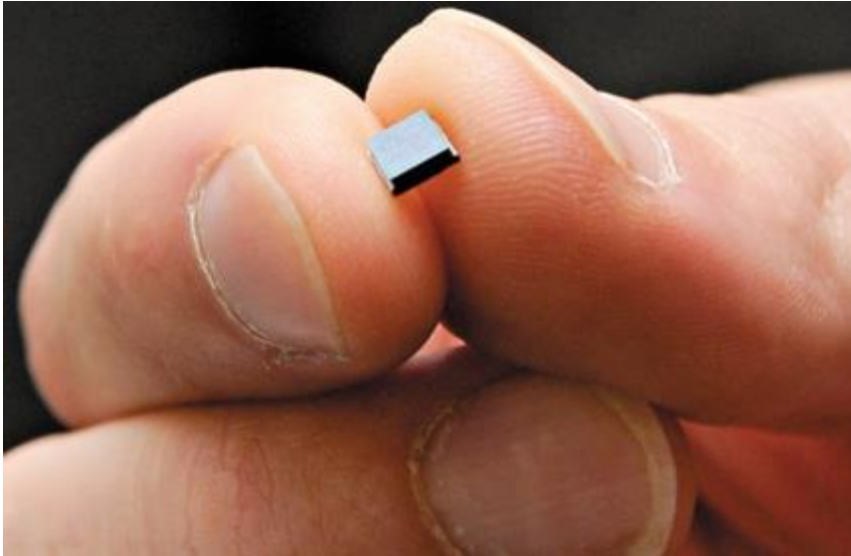


Abbildung 2.x: MEMS-Beschleunigungssensor [MEMS]

2.2.3 Programmcode

[Anhang: Handschuh]

Unser Arduino wird zunächst mit dem WLAN verbunden, indem die SSID und das Passwort im Quellcode eingerichtet werden. Es wird konstant überprüft, ob die WLAN-Verbindung bestehend ist. Sofern diese abbricht, wird er neu verbunden. Die Broker ID und der Broker Port wird eingestellt und die MQTT-Verbindung eingerichtet (siehe 2.3 MQTT). Nach erfolgreicher Verbindung mit MQTT wird der Arduino in einem Channel subscribed, wo er die Werte x,y und z der Gyrosteuerung übersetzt und sendet. Bei der Übersetzung wird geprüft, ob die Werte positiv oder negativ sind, abhängig dieser Werte wird das Auto vorwärts, rückwärts, links oder rechts fahren. Nach Auswertung der Werte werden diese als Character an das MQTT weitergeleitet – entsprechend f, b, l, r, s. Das „s“ ist ein Befehl, wenn die Neigung nicht ausreichend ist und kann als „stopp“ verstanden werden.

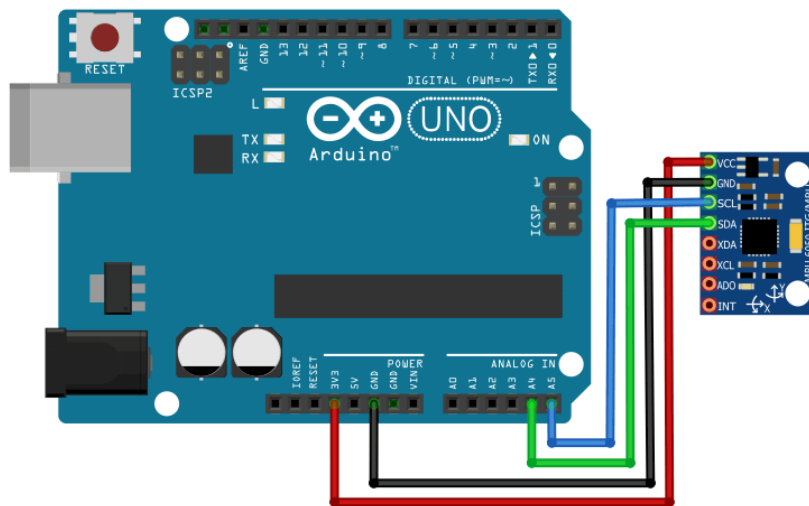


Abbildung 0.x: Aufbau der Gyro-Steuerung zum Arduino [Arduino2]

2.3 MQTT

Die grundlegende Erklärung wie MQTT funktioniert beginnt damit, dass wir einen Knotenpunkt besitzen, wie den Arduino mit WLAN-Modul, welcher Nutzdaten, wie Texte

an den Broker sendet. Dieser Arduino ist der Publisher und ist bei uns am Handschuh zu finden. Der Begriff Broker bezeichnet ein Architekturmuster zur Strukturierung von entfernten und nicht gekoppelten Komponenten. Im Finanzwesen bezeichnet dieser die Vermittlung von Handelsobjekten zwischen Anlegern und überwiegend der Börse. Übertragend in die Informationstechnik dient dieser zur Vermittlung von Nutzdaten zwischen entkoppelten Komponenten. Er ist eine Art Mittelpunkt-Server, welcher Nutzdaten in Themen speichert, wie z.B. unsere Gyroskopdaten. Ein weiterer Knotenpunkt, also bei uns das RC-Car, beziehungsweise dessen Arduino kann mit einer Subscription des Themas die Informationen vom Broker abrufen. Dieser Austausch erfolgt über das Internet, speziell bei den Arduinos über die WLAN-Module. Für wertvolle Daten wird häufig mit Tokens gearbeitet, welche die Daten verschlüsseln. Für unser Projekt ist dies jedoch nicht nötig.

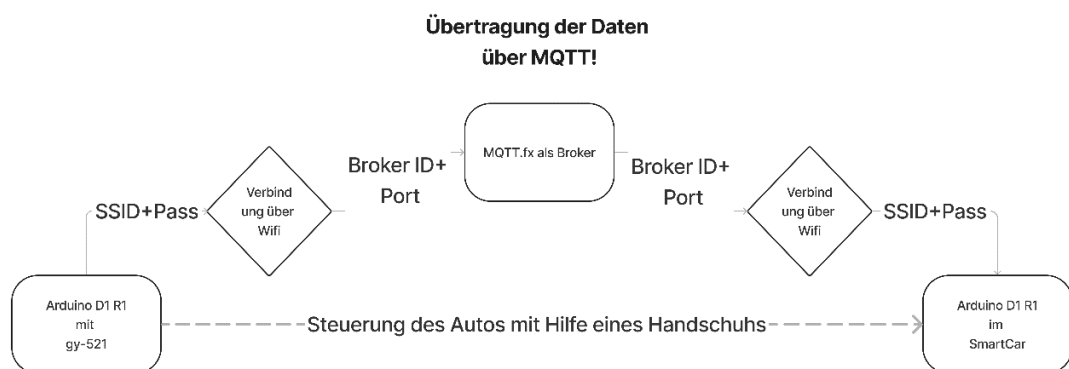


Abbildung 0.5: Funktionsweise des MQTT in unserer Anwendung [MQTT]

3 Projektanalyse

4 Zusammenfassung und Ausblick

Anhang

[RC-Car] RC-Car.txt

[Handschuh] Handschuh.txt

Abbildungsverzeichnis

Abbildung 0.1: Aufbau der Steuerung des RC-Cars

Abbildung 0.2: Kreiselinstrument in Bewegung (rot: Kreiselachse, grün: Achse äußere Kraft, blau: Achse Ergebnis)

Literaturverzeichnis

[Car] elegoo.com, ELEGOO UNO R3 Project Smart Robot Car Kit V 3.0 Plus, 2022
<https://www.elegoo.com/collections/robot-kits/products/elegoo-smart-robot-car-kit-v-3-0-plus>, letzter Zugriff 23.09.2022

[Gyro1] Wikipedia.org, Kreiselinstrument, 13. Juni 2022,
<https://de.wikipedia.org/wiki/Kreiselinstrument>, letzter Zugriff 23.09.2022

[MEMS] itwissen.info, MEMS-Sensor, 2022,
<https://www.itwissen.info/MEMS-Sensor-MEMS-sensor.html>, letzter Zugriff 23.09.2022
