

Elektrifiziertes SUP

Stand-up-Paddle-Board mit ferngesteuerten Motoren

Als ich mit meinem nagelneuen Stand-up-Paddle-Board (SUP) – wie im Titelbild zu sehen – am frühen Morgen über den spiegelglatten Ossiacher See in Kärnten/Österreich gepaddelt bin, hat mich endgültig die Faszination des Stand-up-Paddelns gepackt und ich habe verstanden, warum diese Sportart DIE Trendsportart der letzten Jahre geworden ist. Es ist leicht zu erlernen, aufblasbare SUPs sind relativ leicht zu transportieren, und man trainiert die Armmuskeln und gleichzeitig den Gleichgewichtssinn. In einer stehenden Position beobachtet man sowohl die Unterwasserwelt als auch die Landschaft, während man durch das Wasser gleitet. Falls man allerdings gegen einen starken Wind paddeln muss, erfährt man sehr rasch, wie anstrengend das sein kann und wie schnell man an seine physischen Grenzen stößt. Daraus entstand die Idee, mein SUP mit einem alternativen Antrieb zu elektrifizieren.



Leserwettbewerb



*siehe Seite 112

Dr. Peter Tschulik

hat für seinen Beitrag zum Leserwettbewerb einen Gutscheincode* über 200,- Euro erhalten!

Konzept

In diesem Beitrag beschreibe ich meinen Weg zur Elektrifizierung meines SUPs (Stand-up-Paddle-Boards), wobei die Prämisse war, keine Änderungen am SUP selbst durchzuführen und dieses auch ohne elektrischen Antrieb weiterhin voll nutzen zu können. Inspiriert wurde ich von dem Projekt „RC Benchy Tugboat with drone motors“, bei dem ein 3D-gedrucktes Boot ein SUP ziehen kann [1].

Darauf aufbauend wollte ich eine kompaktere Version entwickeln, mit der ich das SUP auch gut lenken kann. Es gibt fertige SUP-Antriebe, die allerdings sehr teuer und auch nur für bestimmte SUP-Modelle verfügbar sind. Bei meinem Projekt gab es die folgenden technischen Hürden zu überwinden, die ich in diesem Beitrag beschreibe und die auch für andere Projekte interessant sein könnten.

- Ansteuerung von Brushless-Motoren über einen Mikrocontroller
- Realisierung einer Fernbedienung (Sender und Empfänger) mit ESP32-Boards und der ESP-now-Funktionalität
- Wasserdichter Einbau der Komponenten
- Einsatz der Deep-Sleep-Funktionalität für den batteriebetriebenen Sender
- Gestaltung einer ansprechenden Frontplatte für den Sender

Der elektrische SUP-Antrieb besteht aus den Teilen Antriebseinheit, Empfangsbox und Handsender, die ich im Folgenden beschreibe.

Hinweis: In diesem ELVjournal zeigen wir ab Seite 22 den detaillierten Einstieg in ESP-now.

Aufbau der Antriebseinheit

Beginnen möchte ich mit der Beschreibung der Antriebseinheit für das SUP. Hierzu habe ich mir eine Ersatzflosse besorgt, die ich - wie in [Bild 1](#) gezeigt - umgebaut habe. An die SUP-Flosse (im Bild stehend in der Mitte zu sehen) habe ich jeweils zwei Edelstahlwinkel aus dem Baumarkt im 90°-Winkel an beide Seiten der Flosse geschraubt und mit zwei geraden Winkeln verlängert. An die beiden Enden habe ich jeweils eine Motoreinheit (Bezugsquelle siehe [\[2\]](#)) mit jeweils drei Schrauben fixiert.



Bild 1: Antriebseinheit für das SUP

Mit den beiden Motoren kann eine Lenkung sehr leicht realisiert werden, indem jeweils ein Motor in die Gegenrichtung dreht. Aus optischen Gründen habe ich die ganze Konstruktion in der blauen Farbe der SUP-Flosse lackiert.

Bei den Motoren handelt es sich um bürstenlose Unterwassermotoren, die in einem turbinenartigen Gehäuse inklusive 4-Blatt-Propellern bereits montagefertig geliefert werden. Bei bürstenlosen Motoren handelt es sich um Gleichspannungsmotoren, die ohne Bürsten auskommen und sehr leistungsfähig und kompakt gebaut werden können. Sie kommen im Modellbau in Drohnen, Autos und Speed-Booten zum Einsatz. Nachteilig ist, dass diese Motoren über drei Anschlüsse verfügen und über eine spezielle Elektronik aufwendig angesteuert werden müssen. Laut Spezifikation kann der eingesetzte Motor mit 12 bis 24 Volt betrieben werden, er ist nur im Süßwasser einzusetzen. Die drei recht kurzen Anschlusskabel habe ich in den schwarzen Anschlussboxen zu einem 6-poligen Kabel zusammengefasst.

Die Anschlussboxen bestehen aus jeweils einer Gehäusenhälfte eines Kleingehäuses. Die Hälften mit den eingeleiteten und mit Schrumpfschlauch isolierten Verlängerungen habe ich anschließend mit Gießharz [\[3\]](#) vergossen. Alle Kabel und die Anschlussboxen sind mit Kabelbinder über zusätzliche Befestigungsbohrungen mit den Winkeln verbunden, damit kein Kabel lose herumliegt. [Bild 2](#) zeigt die Befestigung im Detail.

Das 6-polige Anschlusskabel habe ich mit einem mechanischen Schutz versehen, indem ich einen Duschschlauch auf die entsprechende Länge gekürzt und über das Kabel gestülpt habe. Die beiden Enden habe ich zusätzlich mit Silikon versiegelt. Eine Seite am Winkel des Antriebs habe ich mit einem Kabelbinder fixiert. An das andere Ende des Kabels kommt ein spritzwassergeschützter 8-poliger Stecker [\[4\]](#), wobei die Belegung zwar frei wählbar ist, allerdings mit dem Gegenstück zusammenpassen muss.



Bild 2: Rechter Teil der Antriebseinheit

Aufbau der Empfangsbox

Kommen wir zum Aufbau der Empfangsbox, die neben den Antriebsbatterien die Motorregler sowie das ESP-Empfangsboard enthält. Die Auswahl eines geeigneten Motorreglers gestaltete sich gar nicht so einfach, da dieser sowohl auf den eingesetzten Motor, den 4S-Akku sowie auf die Möglichkeit des Vor- und Rückwärtsantriebs ausgelegt sein muss. Meine Wahl fiel dann schlussendlich auf [5].

Dieser Brushless-Regler scheint sehr zuverlässig zu funktionieren und kann mit der beigefügten Bedienungsanleitung leicht programmiert werden. Das eingesetzte Gehäuse [6] erfüllt die Schutzklasse IP66 und bietet mit den Abmessungen 220 x 120 x 90 mm gerade genug Platz, um alle Komponenten unterzubringen.

Damit die beiden parallel geschalteten, recht schweren 4S-Akkus nicht im Gehäuse lose herumliegen, habe ich mir zuerst einmal eine Halterung nach der Konstruktionskizze (s. Bild 3) aus Aluminiumblech mit der Stichsäge ausgeschnitten.

An den gestrichelt eingezeichneten Linien wird das Blech wie folgt um 90 Grad gebogen: Mit einem scharfen Stanley-Messer werden die Biegekanten an der Außenseite leicht eingeritzt und dann in einem Schraubstock gebogen. Bild 4 zeigt die fertige Akkuhalterung.

In der Akkuhalterung unten sieht man drei Bohrungen, die zur Fixierung des Akkuhalters im Gehäuse über Abstandsrollen dienen. Die weißen Filzteile sind angebracht, um zu verhindern, dass die Akkus gegen die Schrauben scheuern. Die zwei Anschlüsse für die Akkus sind über XT60-Stecker realisiert und wurden auf eine Lochrasterplatine gelötet, die mit der Unterseite des Akkuhalters über zwei Schrauben verbunden ist.

Die vier schwarzen „Linien“ im Bild sind Kabelbinder, die die Motorregler an der Unterseite der Akkuhalterung fixieren, wie Bild 5 zeigt. Die Stromversorgungsleitungen der beiden Regler werden mit den XT60-Steckern verlötet, wobei die beiden Akkus parallel geschaltet sind.

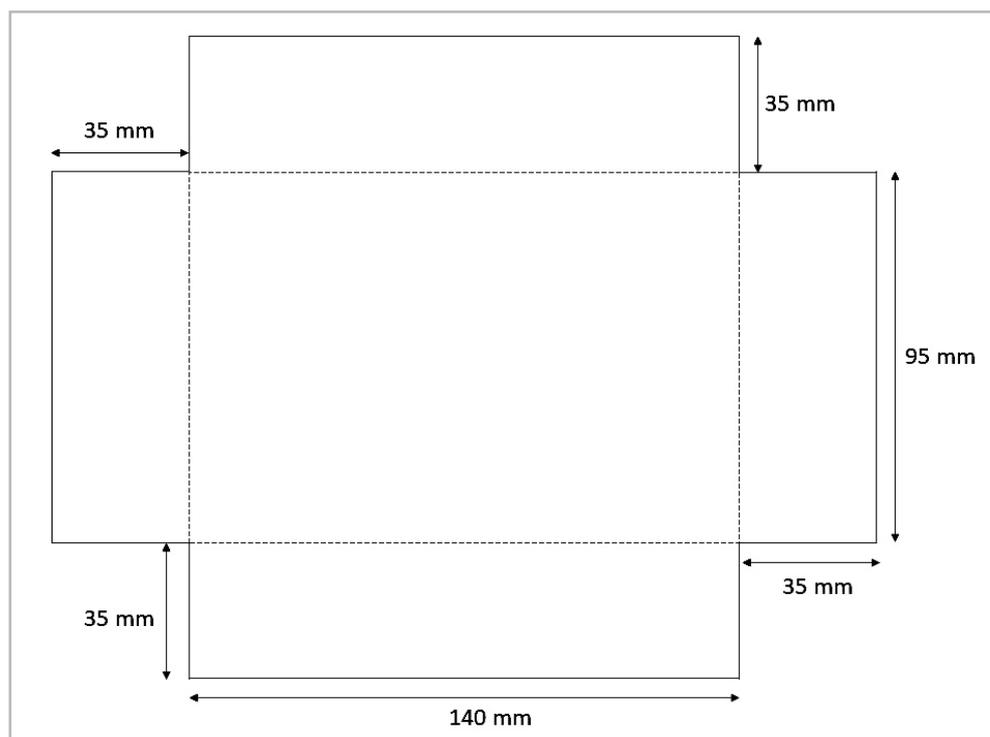


Bild 3:
Konstruktions-
skizze der Akku-
halterung

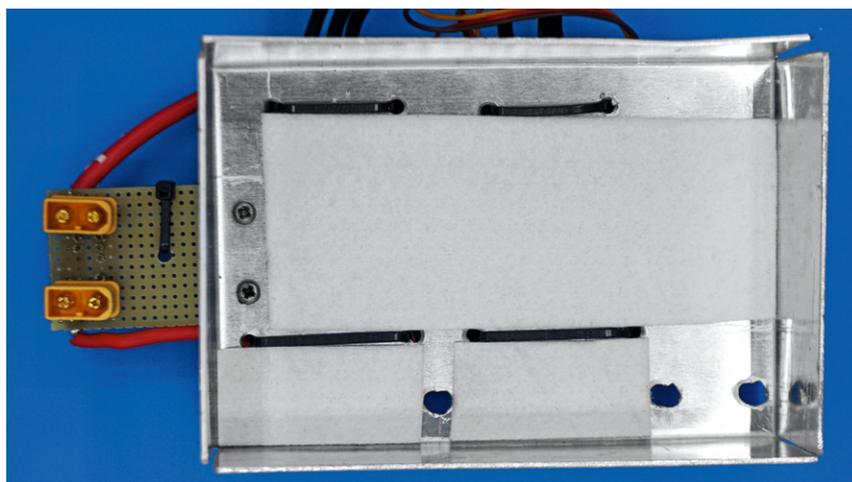


Bild 4: Fertig gebogene Akkuhalterung

Als Nächstes wird die 8-polige Motoranschlussbuchse [7] an einer geeigneten Stelle des Gehäuses montiert, indem ein Loch mit der entsprechenden Größe gebohrt wird, wie in Bild 6 rechts unten ersichtlich. Im Anschluss erfolgt die Verkabelung anhand des Schemas in Bild 7.

Als Mikrocontroller kommt die sehr kompakte und preisgünstige ESP C3 Stamp von M5Stack zum Einsatz [8]. Die ESP C3 Stamp wird über die BEC-Funktionalität eines der Motorregler versorgt, wobei die beiden Spannungsausgänge der beiden Motorregler auf keinen Fall miteinander verbunden werden dürfen.

Die ESP C3 Stamp wird über einen 4-poligen Stecker mit dem Rest der Schaltung verbunden. Der linke Motorregler wird mit GPIO 5 und der rechte Motorregler mit GPIO 6 verbunden. GND(Masse) und der 5-V-BEC-Ausgang eines Motorreglers werden mit den entsprechenden Pins der ESP C3 Stamp verbunden. Falls die ESP C3 Stamp über den USB-C-Stecker versorgt und programmiert wird, darf diese nicht mit dem Rest der Schaltung verbunden sein!

Die verkabelte ESP C3 Stamp mit dem zugehörigen Stecker ist in Bild 8 zu sehen.

Noch ein Hinweis zur Verkabelung der Motoren: Sollte sich einer der Motoren in die falsche Richtung drehen, so müssen nur beliebige zwei der drei Adern des betroffenen Motors vertauscht werden. Ist die Verkabelung abgeschlossen, wird der Akkuhalter mit dem Gehäuseunterteil über Abstandsrollen - wie in Bild 6 gezeigt - verschraubt.

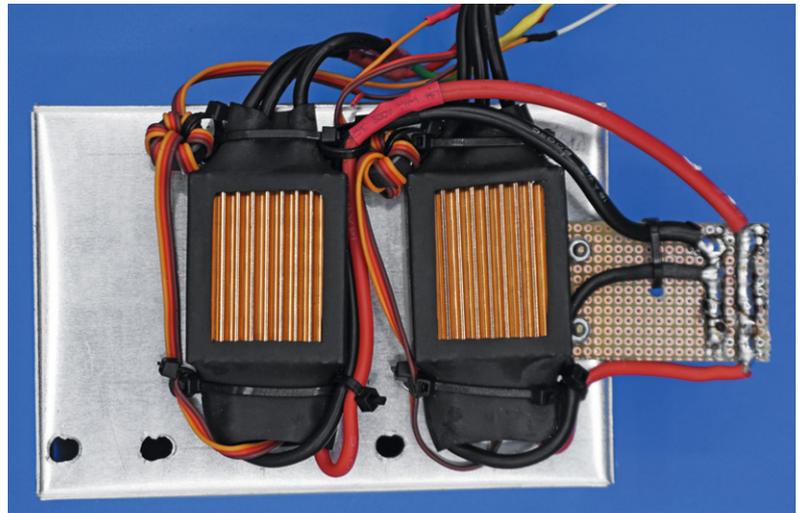


Bild 5: Montage der Motorregler an der Unterseite des Akkuhalters

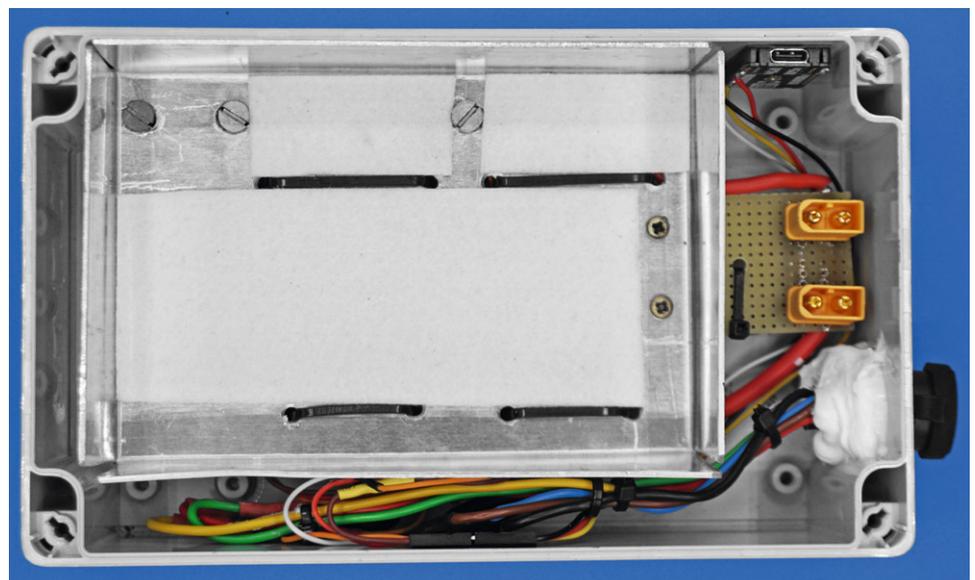


Bild 6: Montage der Motoranschlussbuchse und der anderen Komponenten

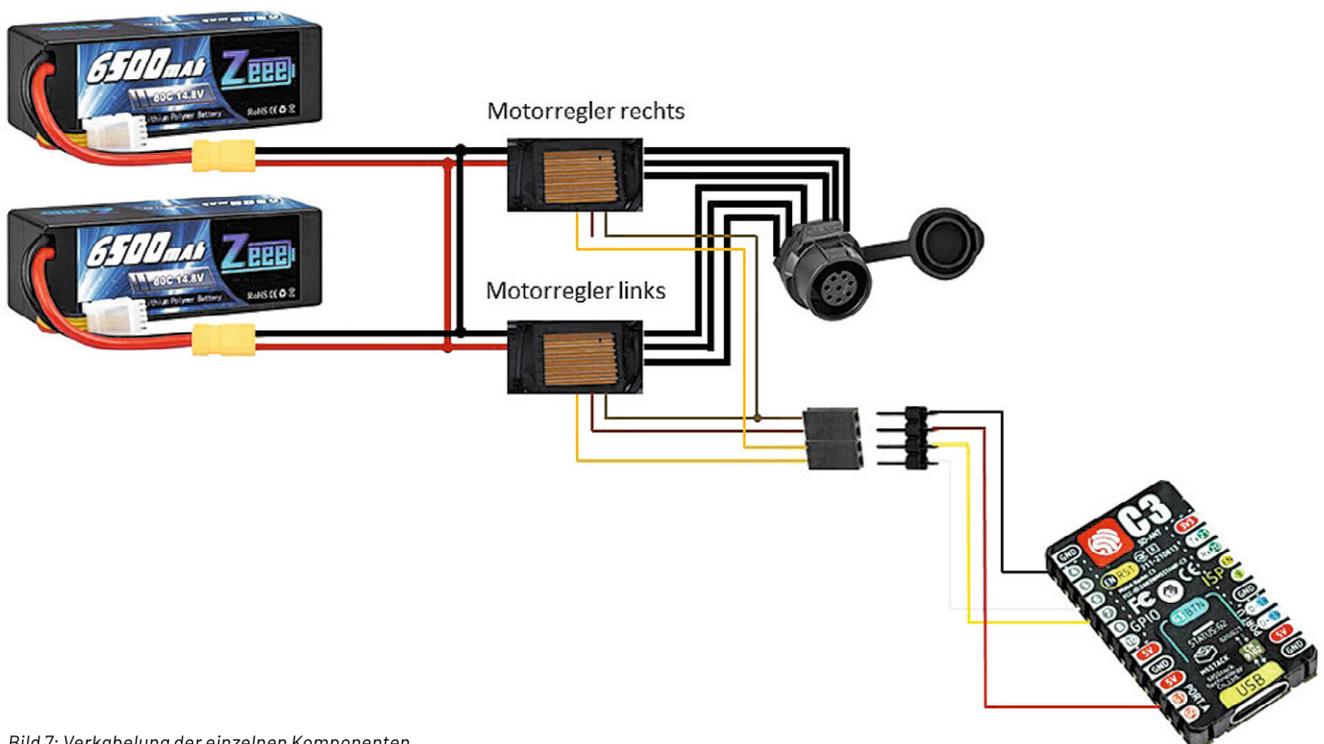


Bild 7: Verkabelung der einzelnen Komponenten

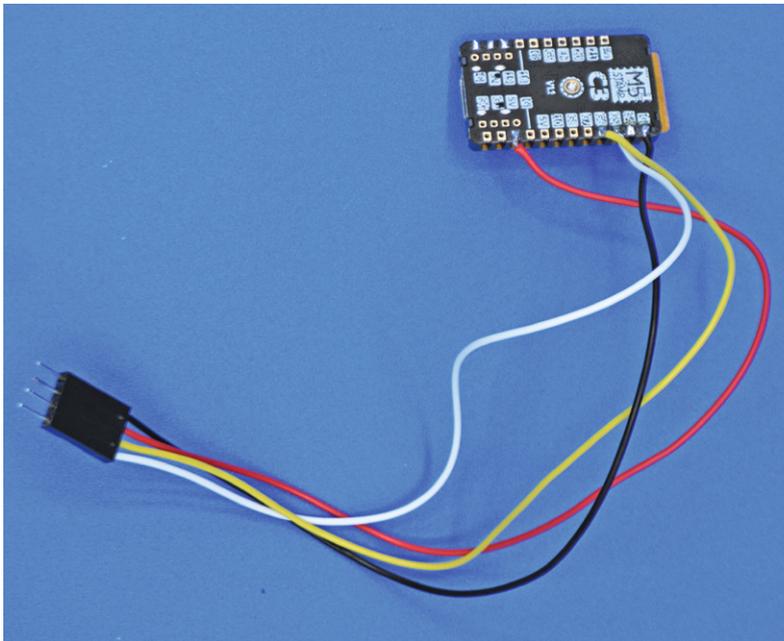


Bild 8: Verkablung der ESP C3 Stamp



Bild 9: Fertig aufgebaute Empfangsbox



Bild 10: Empfangsbox mit Klettband

Die ESP C3 Stamp wird mit Doppelklebeband an die Innenseite des Gehäuses geklebt. Nach einem ersten Funktionstest habe ich dann, wie in [Bild 6](#) gezeigt, die Motoranschlussbuchse mit Silikon zusätzlich abgedichtet.

[Bild 9](#) zeigt den vollständigen Aufbau inklusive der beiden 4S-Akkus [\[9\]](#). Die Akkus versprechen eine hohe Kapazität zu einem fairen Preis und einen sehr robusten Gehäuseaufbau.

Bei den beiden Motorcontrollern habe ich nach der beigefügten Bedienungsanleitung die Abschalt-schwelle für die 4S-Akkus eingestellt. Zum Einsatz kam eine Fernbedienung mit dem Empfänger eines Modellbauautos.

Nach der Verschraubung des Gehäuseoberteils stellt sich noch die Frage, wie die Empfangsbox am SUP montiert wird, damit diese leicht anbring- und entfernbar und sicher am SUP fixierbar ist. Ich habe dazu ein extra dickes Klettband verwendet [\[10\]](#). Zwei Stücke habe ich mit Montagekleber an die Gehäuse-Ober- und -Unterseite geklebt. Ein längeres Stück dient dazu, die Empfängerbox mit der Befestigungs-schleife des SUPs zu verbinden. [Bild 10](#) zeigt das fertige Gehäuse samt Klettband.

[Bild 11](#) und [Bild 12](#) schließlich zeigen die Montage des Antriebs und der Empfängerbox am SUP.

Damit ist die Beschreibung der Empfangsbox abgeschlossen und wir widmen uns dem Handsender.

Aufbau des Handsenders

Für den Handsender habe ich mit dem Programm „Eagle“ eine Platine entwickelt, die als Originaldatei oder als Bild unter [\[11\]](#) downloadbar ist. [Bild 13](#) zeigt den Schaltplan dazu.

Da der Handsender mit vier Standard-Micro-Batterien oder -Akkus betrieben werden soll, kommt ein sogenannter Buck-Boost-Konverter [\[12\]](#) zum Einsatz, der die Spannung auf 3,3 Volt sowohl absenken als auch anheben kann, um die Batterie optimal auszunutzen. Achten Sie darauf, die 3,3-V-Version zu bestellen, da es die Platine mit unterschiedlichen Ausgangsspannungen gibt.

Zentrales Element der Schaltung ist ein ESP32-Pico-Kit [\[13\]](#), das über viele Pins verfügt und leicht von verschiedenen Distributoren zu erwerben ist. An dem ESP32-Pico-Kit hängen über entsprechende Vorwiderstände fünf LEDs, die die Geschwindigkeitsstufe und -richtung anzeigen, und eine zweifarbige LED (rot/grün), die anzeigt, ob die Übertragung des Sendebefehls erfolgreich war. Diese zweifarbige LED hat nur zwei Pins, und die Farbe der LED wird durch Umpolung der Versorgungsspannung geändert. Schlussendlich hängen an weiteren Ports des ESP32-Pico-Kits fünf Taster, mit denen die Geschwindigkeit, die Richtung und Stopp ausgewählt werden können.



Bild 11: SUP-Montage der Antriebseinheit



Bild 12: SUP-Montage der Empfangsbox

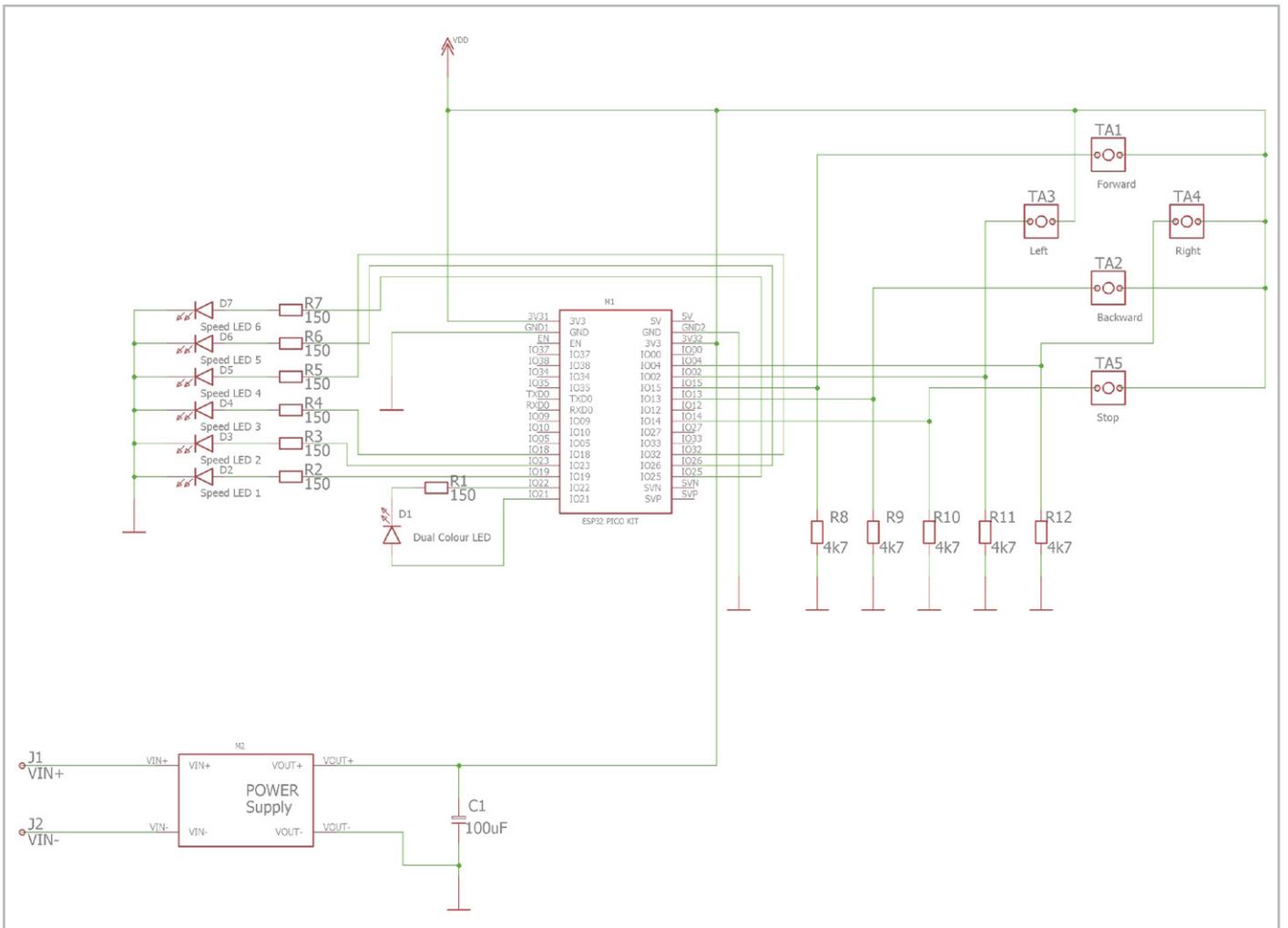


Bild 13: Schaltplan des Handsenders

Kommen wir zur Bestückung der Platine. Zuerst einmal müssen sechs Drahtbrücken verlötet werden, wie Bild 14 zeigt. Danach werden die Widerstände und der Elektrolytkondensator – auf die korrekte Polung achten! – bestückt.

Es folgt das ESP32-Pico-Kit-Modul, das wegen der Einbauhöhe möglichst plan zu bestücken ist. Das Buck-Boost-Modul wird verlötet, indem kurze Drahtstücke mit der Hauptplatine und dem Modul verlötet werden. Es folgen die Taster, die ebenfalls plan zu bestücken sind.

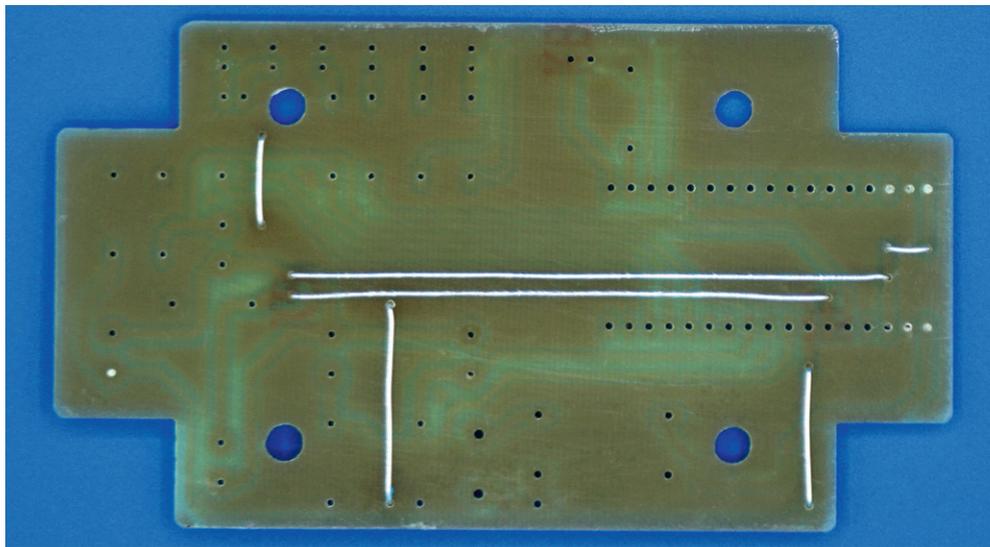


Bild 14: Platine mit bestückten Drahtbrücken

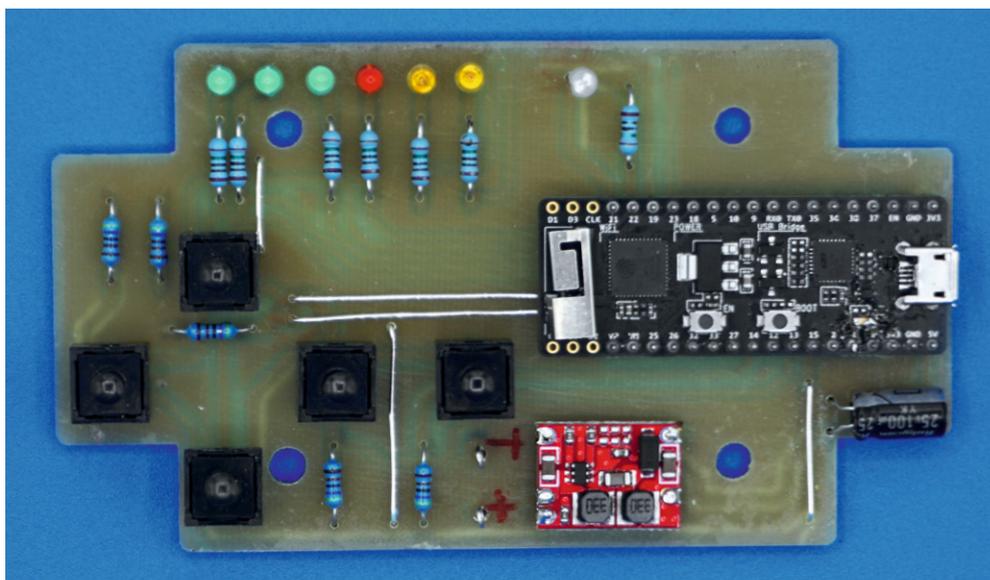


Bild 15: Fertig bestückte Senderplatine

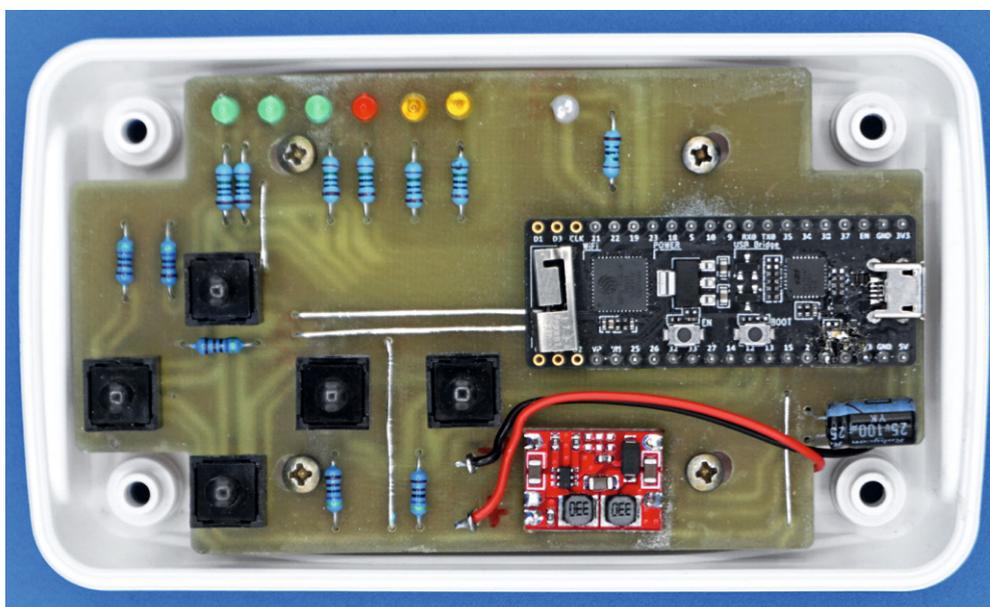


Bild 16: Montage der Platine im Gehäuse

Zuletzt werden die LEDs bestückt, die einen Abstand von 7 mm zur Platine aufweisen müssen. Drei grüne LEDs zeigen die drei Fahrstufen vorwärts an, eine rote LED die Stopp-Position und die zwei gelben LEDs die zwei Fahrstufen rückwärts an.

Die zweifarbige LED wird nach dem Trial-and-Error-Prinzip bestückt: Leuchtet sie falsch herum, so wird sie umgedreht oder die Software entsprechend angepasst! Bild 15 zeigt die fertig bestückte Platine.

Die Platine passt genau in ein spritzwassergeschütztes Gehäuse von Bopla [14], das auch über ein Batteriefach für vier Micro-Batterien (AAA) verfügt. Die entsprechenden Dichtungen sind extra zu bestellen. Mit den blauen Dichtungen und dem blauen Stoßschutz passt das Gehäuse perfekt zu meinem blauen SUP vom Titelbild.

Für den Einbau werden zuerst an den Batteriehalter zwei Drähte für Plus und Minus gelötet, die dann mit den Pins in der Nähe des Buck-Boost-Konverters verbunden werden. Danach wird die Platine mit vier Schrauben in das Gehäuse geschraubt, wie aus Bild 16 ersichtlich ist.

Für die Bearbeitung der Frontplatte habe ich eine Bohrschablone [11] angefertigt, die in Bild 17 zu sehen ist. Diese wird einfach im Maßstab 1:1 ausgedruckt, ausgeschnitten und auf die Frontplatte geklebt, bevor die Bohrungen angefertigt werden. Die Bohrungen werden zuerst angekört und anschließend gebohrt. Bild 18 zeigt das aufgesetzte Gehäuseoberteil.

Da die Tastenköpfe zu weit aus dem Gehäuse ragten, habe ich sie mit einem Dremel mit aufgesetzter Diamanttrennscheibe so gekürzt, dass sie weniger als einen 1 mm aus dem Gehäuse zeigen.

Jetzt fehlt noch eine ansehnliche Bedruckung der Frontplatte. Diese stelle ich wie folgt her: Zuerst drucke ich die Frontplatte [11] auf einem guten Farblaserdrucker im Maßstab 1:1 aus. Danach bringe ich an der Rückseite des Ausdrucks eine doppelseitige Klebefolie an, gute Erfahrungen habe ich mit [15] gemacht.

Die Aussparungen für die LEDs stanze ich mit einem 3-mm-Lochzahn aus, dann laminiere ich die Vorderseite mit einer matten Laminierfolie [16].

Danach schneide ich die Frontplatte aus, überprüfe nochmals die Größe und klebe sie passgenau auf das Gehäuse. Damit die Tasten bedienbar bleiben, muss an den Stellen der 10-mm-Löcher noch ein dünnes Stück Papier angebracht werden, sonst kleben die Tastenköpfe an und lassen sich nicht gut bedienen. Der Aufwand lohnt sich, und man erhält eine professionelle kratz- und wasserfeste Oberfläche.

Bild 19 zeigt das fertige Gehäuse inklusive der blauen Schutzabdeckung.

Das Gehäuse mit der Frontplatte sollte zwar spritzwasserfest sein, ein Untertauchen ins Wasser würde der Handsender aber wahrscheinlich nicht unbeschadet überstehen. Deshalb habe ich mir noch eine wasserdichte Tasche gekauft [17], die ich mir beim Stand-up-Paddeln umhängen kann und die absolut wasserdicht ist (Bild 20).

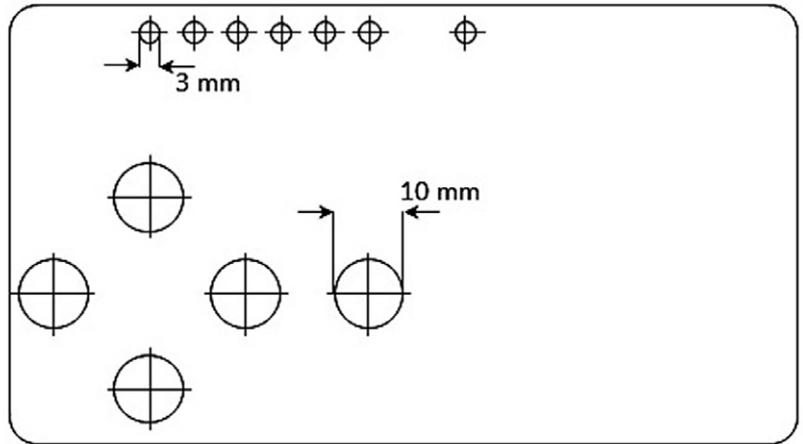


Bild 17: Bohrschablone für die Gehäusefront



Bild 18: Gehäusefront mit Bedienelementen

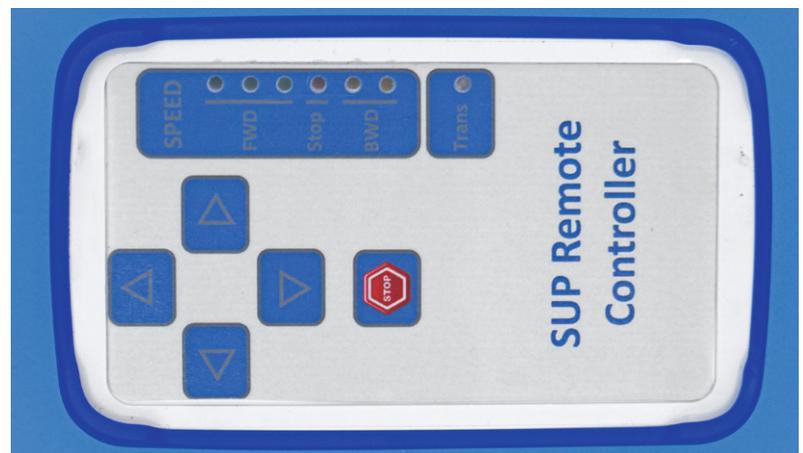


Bild 19: Fertiger Sender im Gehäuse eingebaut



Bild 20: Fertiger Handsender in wasserdichter Verpackung

Inbetriebnahme und Bedienung

Zunächst müssen wir für die ESP-now-Funktionalität die MAC-Adresse der ESP C3 Stamp herausfinden. Dazu müssen die Boards von M5 Stack in der Arduino-Umgebung verfügbar sein. Eine Anleitung dazu findet man unter [18]. Danach öffnet man den Sketch „SUP_ESP32_MAC.ino“, der unter [11] verfügbar ist, und überträgt ihn auf das Board.

Im „Serial Monitor“ wird dann die MAC-Adresse der ESP C3 Stamp angezeigt. Diese Adresse notiert man sich auf einem Zettel. Danach kann schon der eigentliche Sketch aufgespielt werden. Zuvor muss noch die Library „ESP32_C3_ISR_Servo“, die ebenfalls unter [11] verfügbar ist, installiert werden. Dies geschieht dadurch, dass das gesamte Verzeichnis in das „libraries“-Verzeichnis der Arduino-Umgebung kopiert wird.

Achtung: Die Library wurde geringfügig modifiziert, da sie bei der ESP C3 Stamp meckerte. Die Änderungen sind auch im Header des Sketches „SUP_ESP32_Receiver.ino“ beschrieben, der nun in der Arduino IDE geöffnet und auf die ESP C3 Stamp übertragen wird.

Noch ein weiterer Hinweis: Das Programm funktioniert nur mit ESP32-C3-Versionen und nicht mit normalen ESP32-Boards! Für ESP32-Boards muss eine andere Library für die Servoansteuerung eingesetzt werden.

Die Funktion des Sketches soll hier nur kurz beschrieben werden, da dieser ausführlich dokumentiert ist. In der Set-up-Routine werden die Servos initialisiert und in die Neutralpositionen gestellt, damit die Motoren ausgeschaltet bleiben. Nach einer Wartezeit, die die Motorcontroller benötigen, wird die ESP-now-Funktionalität initiiert und eine Interrupt-Routine definiert, die bei einem Empfang von Datenpaketen angesprungen wird. Das implementierte Protokoll wird ebenfalls im Header des Sketches beschrieben.

In der Hauptroutine wird das Protokoll abgearbeitet und die entsprechenden Befehle werden ausgeführt. ESP-now arbeitet wie eine Fernbedienung und ist eine einfache und leistungsfähige Möglichkeit, Daten zwischen zwei oder mehreren ESPs auszutauschen. Wer tiefer in die Funktionalität eintauchen möchte, dem empfehle ich als Einstieg das Tutorial unter [19]. Der große Vorteil von ESP-now ist, dass man sich weder um den Verbindungsaufbau noch um das Handshaking kümmern muss.

Inbetriebnahme des Senders

Kommen wir nun zur Inbetriebnahme des Senders. Zuerst ist das ESP32-Pico-Kit-Modul vom Rest der Schaltung über den 4-poligen Stecker – siehe [Bilder 7 und 8](#) – zu trennen, bevor es mittels USB-Kabel mit einem PC oder Laptop verbunden wird.

In der Arduino IDE öffnet man den unter [11] verfügbaren Sketch „SUP_ESP32_Sender.ino“. Als Board muss das „ESP32 Pico Kit“ ausgewählt werden. In Zeile 115 muss die MAC-Adresse des Empfängers unter „BROADCAST_ADDRESS“ eingetragen werden. Dann kann der Sketch auf das Board übertragen werden.

Die Funktion des Programms ist wie folgt: Beim Start wird nach einigen Initialisierungen ein Test aller LEDs durchgeführt, und danach sollte die rote Geschwindigkeits-LED leuchten, was einem Stopp entspricht. Ist der Empfänger eingeschaltet, so kann ein erster Funktionstest erfolgen. Man drückt die Taste „Forward“, und die Geschwindigkeitsstufe sollte auf die erste grüne LED springen für „langsam vorwärts“. Leuchtet die zweifarbige LED grün, so sollte – sofern die LED korrekt gepolt ist – der Befehl korrekt übertragen worden sein und die Motoren sollten sich langsam drehen.

Mit der Taste „Forward“ kann die Geschwindigkeit bis zum Maximum erhöht werden, mit der Taste „Backward“ kann die Geschwindigkeit zuerst bis zur Stopp-Position verringert werden, und danach können die zwei Geschwindigkeitsstufen für den Rückwärtsgang angewählt werden. Die „Stop“-Taste stoppt den Motor aus jeder Geschwindigkeitsstufe.

Drückt man in der Vorwärtsrichtung die „Left“- oder „Right“-Taste, wird eine Motorrichtung so umgekehrt, dass die entsprechende Richtung angesteuert wird. Die LEDs zeigen in diesem Fall die gewählte Vorwärts- und Rückwärtsgeschwindigkeit an. Eine Lenkbewegung wird durch ein neuerliches Drücken der „Left“- oder „Right“-Taste beendet, wobei dann die vorher gewählte Geschwindigkeitsstufe in Vorwärtsrichtung eingestellt wird. Die Lenkbewegung kann auch durch die „Stop“-Taste beendet werden, wodurch das SUP zum Stillstand kommt.

Achtung: Die Lenkung funktioniert nur in der Vorwärtsrichtung und ist in der Rückwärtsrichtung gesperrt.

Sollte einmal eine Übertragung über ESP-now fehlschlagen, wird automatisch die Stopp-Position aus Sicherheitsgründen eingestellt.

Eine Sonderfunktion gibt es noch: Wird in der Stopp-Position die „Stop“-Taste für mehr als drei Sekunden gedrückt, werden die Motorcontroller auf die Motoren hin kalibriert. Dazu werden zuerst die beiden Motoren von Stopp auf die maximale Geschwindigkeit hochgefahren und dann wieder bis zur Stopp-Position verringert, um danach das Gleiche in Rückwärtsrichtung durchzuführen.

Diese Kalibrierung sollte bei jeder Inbetriebnahme erfolgen, damit die Motoren den ganzen Regelbereich ausnutzen können. Dass dies entweder im Wasser oder mit Bedacht erfolgen soll, muss eigentlich nicht erwähnt werden. Die Motoren sind kein Spielzeug und haben eine erhebliche Kraft! Während des Kalibriervorgangs leuchten drei LEDs am Sender und die Tasten sind für die Dauer der Kalibrierung gesperrt.

Eine letzte Besonderheit gibt es noch: Da der ESP32 im aktiven WLAN-Betrieb, den ESP-now nutzt, bis zu 200 mA an Strom konsumiert, wären die vier Micro-Batterien bald leer. Eine Micro-Batterie bietet eine Kapazität zwischen 850 und 1200 mAh. Nimmt man daher eine Kapazität von vier Batterien von 4000 mAh an und dividiert diese durch 200 mA, so wären die Batterien in 20 Stunden leer!

Deshalb geht der ESP32 in den Deep-Sleep-Modus [20], wenn innerhalb von zehn Sekunden keine weitere Tastenbedienung erfolgt. In dieser Betriebsart ist der ESP32 bis auf die Real Time Clock (RTC) deaktiviert. Die Stromaufnahme sinkt bis auf 3 mA, was zu einer langen Batterielaufdauer von ca. 50 Tagen führt. Das ist zwar akzeptabel, aber die Micro-Batterien sollten nach einer Nutzung immer entfernt werden, oder man ergänzt einen Ein-/Ausschalter. Die 3 mA erreicht man allerdings nur, wenn die rote Power-LED des ESP32-Pico-Kit ausgelötet wird.

Noch ein paar Worte zu der erreichbaren Fahrzeit: Im Wasser verbrauchen die beiden Motoren in der niedrigsten Geschwindigkeitsstufe ca. 1 A, in der mittleren ca. 3 A und in der höchsten Stufe über 10 A. Nimmt man hier die Akkukapazität von 2x 6500 mAh, ergibt sich in der mittleren Geschwindigkeitsstufe eine Fahrzeit von ca. vier Stunden. In meinen ersten eingeschränkten Tests kam ich auf drei Stunden, was ich darauf zurückführe, dass die Akkus weniger Kapazität bieten als

angegeben und noch nicht die ersten Ladezyklen durchlaufen hatten.

Ich habe noch einige Erweiterungsideen wie beispielsweise eine Betriebsart, mit der Paddelschläge durch den elektrischen Antrieb nur verstärkt werden. Zudem konnten aufgrund der winterlichen Temperaturen, die beim Bau des SUP-Antriebs herrschten, noch nicht alle Praxistests absolviert werden.

Ich biete daher für interessierte Leser an, Verbesserungen, Fragen und Erfahrungen auszutauschen. Bei Interesse dazu bitte eine kurze Mail an peter.tschulik@chello.at senden.

Wenn Sie dieses Projekt nachbauen, steht einem sanften Dahingleiten auf dem SUP ohne Kraftanstrengung nichts mehr im Weg. **ELV**

i Weitere Infos

- [1] Projekt „RC Benchy Tugboat with drone Motors“:
<https://www.youtube.com/watch?v=mg0Hi9M1HmE> und <https://www.thingiverse.com/thing:5408501/makes>
- [2] Brushless-Motor: https://www.amazon.de/dp/B09BJSP5VZ?psc=1&ref=ppx_yo2ov_dt_b_product_details
- [3] Vergußmasse Elektronik: https://www.amazon.de/dp/B071SHRBP9?psc=1&ref=ppx_yo2ov_dt_b_product_details
- [4] Stecker:
<https://www.conrad.at/de/p/lumberg-0251-08-rundsteckverbinder-stecker-gerade-gesamtpolzahl-8-1-st-1784504.html>
- [5] Brushless-Regler: <https://www.ebay.at/itm/263273688662?var=562276517334>
- [6] Gehäuse für Empfänger: <https://www.conrad.at/de/p/tru-components-tc-9065672-universal-gehaeuse-200-x-120-x-90-abs-lichtgrau-ral-7035-1-st-2266418.html> →
- [7] Einbaubuchse:
<https://www.conrad.at/de/p/lumberg-0271-08-rundsteckverbinder-kupplung-einbau-gesamtpolzahl-8-1-st-1784496.html>
- [8] ESP C3 Stamp: <https://shop.m5stack.com/collections/m5-controllers/products/m5stamp-c3-5pcs>
- [9] 4S-Akku: https://www.amazon.de/gp/product/B07K460JP3/ref=ppx_yo_dt_b_asin_title_o05_s02?ie=UTF8&psc=1
- [10] Klettband: https://www.amazon.de/dp/B09FDMLV89?psc=1&ref=ppx_yo2ov_dt_b_product_details
- [11] Downloads: Artikel-Nr. 253477
- [12] Buck-Boost-Konverter:
https://www.amazon.de/gp/product/B09T2LQFYS/ref=ppx_yo_dt_b_asin_title_o00_s00?ie=UTF8&psc=1
- [13] <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/esp32/get-started-pico-kit.html>
- [14] Bopla-Gehäuse: <https://www.conrad.at/de/p/bopla-bop-500-be4mc-9016-hand-gehaeuse-130-x-75-x-26-abs-weiss-ral-9016-1-st-1607282.html> →
Boppla-Gehäuse mit Dichtung: <https://www.conrad.at/de/p/bopla-bop-500-di-5005-dichtung-tpe-geruchneutrales-spezialgummigemisch-blau-1-st-1607268.html> →
Schutzabdeckung: <https://www.conrad.at/de/p/bopla-bop-500-s-5005-schutzabdeckung-l-x-b-x-h-136-x-81-x-36-3-mm-tpe-geruchneutrales-spezialgummigemisch-blau-1-st-1607298.html> →
- [15] Doppelseitige Klebefolie:
https://www.amazon.de/gp/product/B07RFSTDW3/ref=ppx_yo_dt_b_asin_title_o06_s00?ie=UTF8&th=1
- [16] Matte Laminierfolie: https://www.amazon.de/Fellowes-5452101-Laminierfolien-Enhance-Mikron/dp/B000WL3PJU/ref=sr_1_3?__mk_de_DE=%C3%85M%C3%85C5%BD%C3%95%C3%91&crd=1US5HYK4SR6N9&keywords=fellowes+matt&qid=1676060532&srefix=followers+matt%2Caps%2C101&sr=8-3 →
- [17] Wasserdichte Tasche: https://www.amazon.de/dp/B0140Y9R46?psc=1&ref=ppx_yo2ov_dt_b_product_details
- [18] https://docs.m5stack.com/en/arduino/arduino_development
- [19] <https://randomnerdtutorials.com/esp-now-one-to-many-esp32-esp8266/>
- [20] <https://www.electrorules.com/esp32-deep-sleep-with-arduino-ide-and-wake-up-source>

Alle Infos finden Sie auch online unter: de.elv.com/elvjournal-links