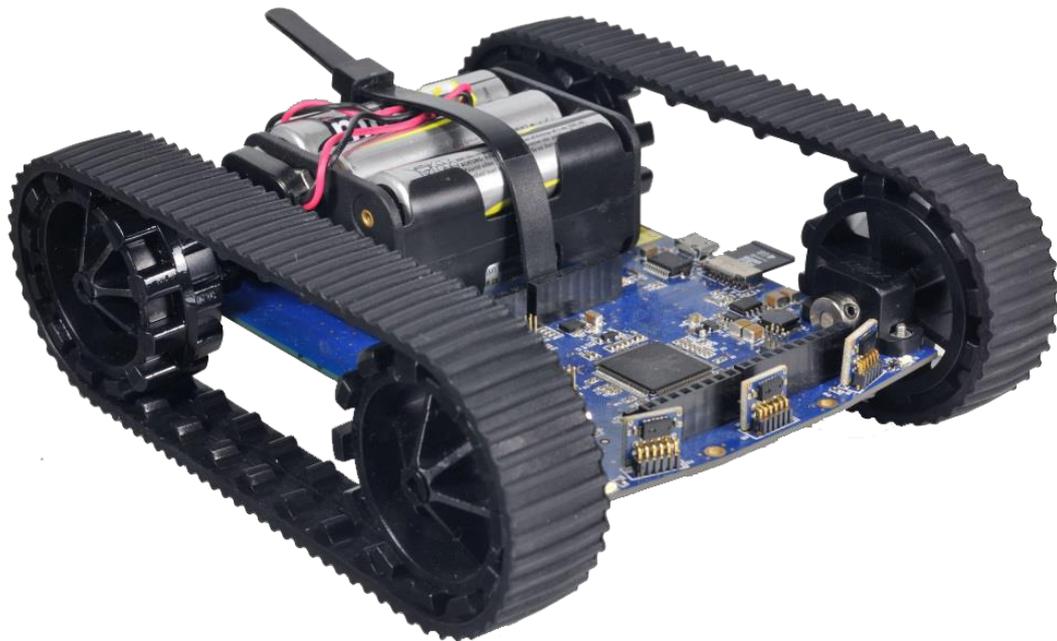


Bedienungsanleitung



Marvin IoT Robot

(c) 2015-2017 AREXX Engineering und JM³ Engineering

www.arexx.com

Neueste Updates sind auf jm3-engineering.com erhältlich!

<p>Impressum</p> <p>© 2016-2017 AREXX Engineering</p> <p>Nevistraat 16 8013 RS Zwolle The Netherlands</p> <p>Tel.: +31 (0) 38 454 2028 Fax.: +31 (0) 38 452 4482</p>	<p>Diese Bedienungsanleitung ist urheberrechtlich geschützt. Der Inhalt darf ohne vorherige schriftliche Zustimmung des Herausgebers auch nicht teilweise kopiert oder übernommen werden!</p> <p>Änderungen an Produktspezifikationen und Lieferumfang vorbehalten. Der Inhalt dieser Bedienungsanleitung kann jederzeit ohne vorherige Ankündigung geändert werden.</p> <p>Neue Versionen dieser Anleitung erhalten Sie kostenlos auf http://www.arexx.com/</p>
<p>„Marvin“ ist lizenziert und „iRP“ ist ein eingetragenes Warenzeichen von „JM3 Engineering“. „AREXX“ ist eine eingetragenes Warenzeichen von AREXX Engineering. Alle anderen Warenzeichen stehen im Besitz ihrer jeweiligen Eigentümer. Wir sind nicht verantwortlich für den Inhalt von externen Webseiten, auf die in dieser Anleitung verlinkt wird!</p>	
<p>Hinweise zur beschränkten Garantie und Haftung</p> <p>Die Gewährleistung von AREXX Engineering beschränkt sich auf Austausch oder Reparatur des Roboters und seines Zubehörs innerhalb der gesetzlichen Gewährleistungsfrist bei nachweislichen Produktionsfehlern, wie mechanischer Beschädigung und fehlender oder falscher Bestückung elektronischer Bauteile, ausgenommen aller über Steckverbinder/Socket angeschlossenen Komponenten.</p> <p>Es besteht keine Haftbarkeit für Schäden, die unmittelbar durch, oder in Folge der Anwendung des Roboters entstehen. Unberührt davon bleiben Ansprüche, die auf unabdingbaren gesetzlichen Vorschriften zur Produkthaftung beruhen.</p> <p>Sobald Sie irreversible Veränderungen (z.B. Anlöten von weiteren Bauteilen, Bohren von Löchern etc.) am Roboter oder seinem Zubehör vornehmen oder der Roboter Schaden infolge von Nichtbeachtung dieser Anleitung nimmt, erlischt jeglicher Garantieanspruch!</p> <p>Es kann nicht garantiert werden, dass die mitgelieferte Software individuellen Ansprüchen genügt oder komplett unterbrechungs- und fehlerfrei arbeiten kann. Weiterhin ist die Software beliebig veränderbar und wird vom Anwender in das Gerät geladen. Daher trägt der Anwender das gesamte Risiko bezüglich der Qualität und der Leistungsfähigkeit des Gerätes inklusive aller Software. Bitte beachten Sie auch die entsprechenden Lizenzvereinbarungen auf der SD-Karte bzw. Info-Box im Browser!</p> <p>WICHTIG! - Vor dem ersten Gebrauch dieses Roboters lesen Sie bitte zuerst die Gebrauchsanleitung aufmerksam durch! Sie erklärt die richtige Handhabung und informiert Sie über mögliche Gefahren. Zudem enthält sie wichtige Informationen, die nicht allen Benutzern bekannt sein dürften.</p>	

Sicherheitshinweise

- Prüfen Sie die Polung von der Spannung.
- Halten Sie die Elektronik stets trocken. Wenn das Gerät einmal nass geworden ist, entfernen Sie sofort die Batterien oder die Stromversorgung.
- Bei längerem Nichtgebrauch die Batterien entfernen bzw. die Stromversorgung trennen.
- Bevor Sie das Modul in Betrieb nehmen, prüfen Sie stets seinen Zustand sowie auch den der Kabel.
- Wenn Sie meinen, dass das Gerät nicht länger sicher betrieben werden kann, trennen Sie es sofort von der Stromversorgung und stellen Sie sicher, dass es nicht unabsichtlich benutzt werden kann.
- Befragen Sie einen Fachmann, wenn sie sich der Bedienung, Sicherheit oder Anschluss des Moduls nicht sicher sind.
- Niemals das Modul in Räumlichkeiten oder unter ungünstigen Bedingungen betreiben.
- Das Modul besitzt hochempfindliche Bauteile. Elektronische Bauteile sind sehr gegen statische Elektrizität empfindlich. Fassen Sie das Modul nur an den Rändern an und vermeiden Sie direkten Kontakt mit den Bauteilen auf der Platine.

Normaler Gebrauch

Dieses Produkt wurde als Experimentierplattform für alle an Robotik interessierten Personen entwickelt. Das Hauptziel besteht darin zu lernen, wie man das Gerät in C/C++-Sprache programmieren kann. Das Gerät ist kein Spielzeug! Es eignet sich nicht für Kinder unter 14 Jahren.

Das Gerät ist nur für den Gebrauch in Innenräumen bestimmt. Es darf weder feucht noch nass werden. Bitte achten Sie auch auf Kondenswasser, das sich bei einem Wechsel von einem kalten in einen warmen Raum entwickeln kann. Warten Sie eine Weile, bis sich das Gerät an die neuen Umgebungsbedingungen angepasst hat, bevor Sie es in Betrieb nehmen. Jede andere Einsatzart als oben beschrieben kann zu Schäden und Risiken wie Kurzschluss, Brand, Stromschlag usw. führen. Eine Verwendung des gesamten Produkts ist nur in geschlossenen, trockenen Innenräumen erlaubt. Das Produkt darf nicht feucht oder nass werden.

Eine andere Verwendung als zuvor beschrieben führt zu Beschädigungen dieses Produkts, außerdem ist dies mit Gefahren wie z.B. Kurzschluss, Brand, elektrischer Schlag etc. verbunden.

Laser Sicherheitshinweise Proxi Sensor VL6180X 80/87 DocID026171 Rev 7

Der VL6180X Proximity Sensor beinhaltet eine Laserlichtquelle und die zugehörige Ansteuer- schaltung. Die Laserlichtausgangsleistung wurde so festgelegt das diese immer unter allen angemessenen und erdenklichen Bedingungen den Sicherheitsgrenzen eines Class 1 Lasers entspricht. Dies beinhaltet auch Einzelfehler gemäß IEC 60 825-1:2007. Die Laserlichtleistung bleibt in den spezifizierten Grenzen so lange die von ST Microelectronics empfohlenen Einstellungen und Betriebsbedingungen wie im Datenblatt spezifiziert eingehalten werden.

Die Laser Ausgangsleistung darf unter keinen Umständen erhöht und keine optischen Linsen zur Bündelung verwendet werden!

Es wird zur Sicherheit empfohlen nicht in die Laserlichtquelle zu schauen.



Complies with 21 CFR 1040.10 and 1040.11 except for deviations pursuant to Laser Notice No.50, dated June 24, 2007.

Inhaltsverzeichnis

1	Einleitung	2
2	Anleitung	4
2.1	Marvin Roboter Hardware	4
2.2	Marvin Roboter Ausstattung und Zubehör	5
2.2.1	Ausstattungsmerkmale	5
2.2.2	Erfassungsbereich der Proximity-Sensoren	6
2.2.3	Einbau zusätzlicher Proximity-Sensoren (Hinten)	7
2.3	Inbetriebnahme	7
2.3.1	Dokumentation und Software	7
2.3.2	iRP WebIDE	8
2.3.3	iRP mit dem PC/Tablet verbinden	9
2.3.4	Status LEDs	12
2.4	Kalibrierung der Sensoren und stellen der Systemzeit	12
2.4.1	RTC - Systemzeit stellen	12
2.4.2	Kalibrierung des 3D Kompass	13
2.4.3	Kalibrierung des 3D Gyroskops	15
2.4.4	Aktivierung der Proximity Sensoren	15
2.4.5	Kalibrierung der Abstandssensoren	15
2.5	Kommandozeileninterface (CLI)	17
3	Programmieren mit iRP	19
3.1	Einführung in iRP	19
3.1.1	Grundlegende Bedienung	19
3.1.2	iRP Hilfe	21
3.1.3	Mein erstes Programm	21
3.1.4	Programm ausführen	22
3.1.5	Programm laden und speichern	22
3.1.6	Programmfehler beheben (Debugging)	22
3.1.7	Programmcode (Quelltext) anzeigen	23
4	JM3 Robot-Tool 2.0	24
4.1	Linux	24
4.2	Windows 7	24
4.3	Laden von eigenen Programmen die unter C/C++ erstellt wurden	25
4.3.1	Upload (Marvin Applikation)	25
4.3.2	Upgrade Firmware (Bootloader)	26
4.4	Terminal Fenster	27
4.5	Firmware und iRP Micro SD-Karten update	27
5	Option: C/C++ Software	29
5.1	Software Paket für Marvin	29
5.1.1	Toolchain	30
5.1.2	Software-Library	30
5.1.3	Marvin Funktionstasten	30
5.2	Demo Software Beschreibung	30
6	Technische Daten	31

6.1	Abmessungen und Gewicht	31
6.2	Spannungsversorgung und Strombedarf	31
6.3	Versorgungsspannungen Experimentierboard	31
6.4	Weitere Daten	31
7	Schaltungsdetails	32
7.1	Arduino Extension Board	32
7.2	Pin Mapping TM4C129EKCPDT	34
7.3	PCB Print	36
	Abbildungsverzeichnis	37
	Tabellenverzeichnis	37

Marvin der IoT Roboter

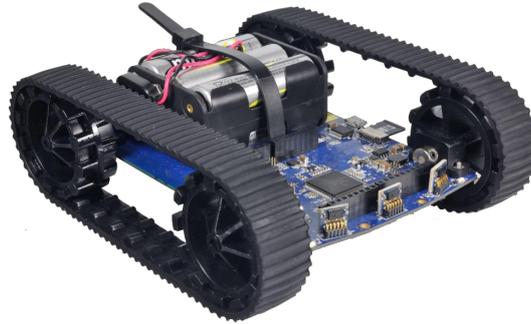


Abbildung 0.1 Marvin Roboter

*Marvin - iRP WebIDE - für Browser auf PC, Notebook und Tablet *1) *2)*



Abbildung 0.2 Welcome-Screen

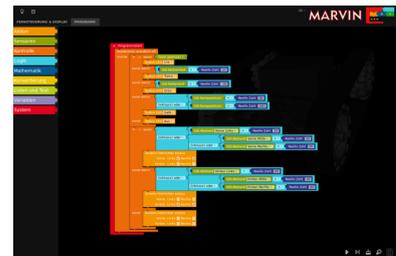


Abbildung 0.3 Program-Screen

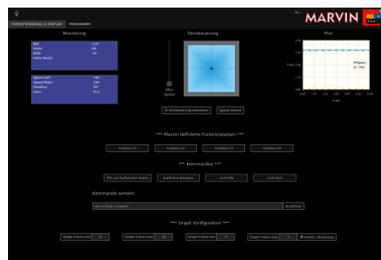


Abbildung 0.4 Remote Control und Status Display

*1) Apple iPad oder MacBook kann Programme nur auf dem Roboter, nicht lokal, speichern.

*2) Firefox, Google Chrome (PC/Notebook/Samsung Galaxy Tab A) und Safari (iPad Pro/MacBook Air) sind getestet – Internet Explorer oder Edge werden nicht unterstützt.

1 Einleitung

Der JM3 IoT-Roboter, genannt Marvin, besticht durch seine Ausstattung und Leistungsfähigkeit. Hauptbestandteile sind der TIVA TM C- Mikrokontroller mit ARM Cortex-M4F und 512KByte Flash, 256KByte SRAM, 6KB EEPROM und dem CC3100 WI-FI ® Networkcontroller der die Standards 802.11 b/g/n mit bis zu 16Mbit/s Datenrate, Multi-Connection, TCP und UDP unterstützt. Optimal angepasste PCB-Antenne, Infrastruktur- & Ad-Hoc Mode und eine Reichweite > 25m unter normalen Bedingungen.

Außerdem ist der Marvin mit High-Tech Sensorik, ein 9D Gyro / Kompass, und Proximity- Sensoren, die nahezu unabhängig vom Reflektionsgrad der Oberfläche arbeiten, ausgestattet. Diese können auch die Umgebungshelligkeit messen. Eine Batterie gepufferte Real-Time-Clock und der Arduino kompatible Erweiterungs-Header gehören ebenfalls zur Ausstattung. Es können hier weitere Sensorik (GPS oder Barometer-Modul), eigene H/W oder andere Arduino Shields (Software muss hier selbst entwickelt werden) angeschlossen werden.

Die kraftvollen Mikro-Getriebemotoren mit hochauflösender Odometrie garantieren eine hohe Geschwindigkeit des Roboters mit seinem Gummikettenantrieb. Die Spannungsversorgung erfolgt über 6 Standard AA Batterien oder NiMH Akkus.

Ein virtuelles Display kann auf im Browser (Firefox, Google Chrome, Safari) auf einem PC, Mac oder Tablet PC dargestellt werden. Außerdem kann der Marvin Roboter ferngesteuert werden.

Die grafische Programmieroberfläche „Marvin – iRP“ ermöglicht den Einstieg in die Welt der Programmierung auf einfache Weise – die insbesondere für Schüler und Jugendliche gedacht ist, da man nicht erst eine Programmiersprache erlernen und eine Software Entwicklungsumgebung installieren muss.

Die Option C/C++ ist für fortgeschrittene Programmierer und Experten gedacht. Die Programmierung erfolgt unter C/C++ mit FreeRTOS und einer kompletten Library aller Hardware- Komponenten (virtuelles Display, Taste, UART, SPI, I2C, DMA, ADC und Timer etc.). Für das Laden des Hex-Codes wird das JM3 –Robot-Tool für Linux und Windows bereitgestellt.

Marvin bietet alle Möglichkeiten einen Roboter zu programmieren und zu steuern, entweder über die grafische iRP Oberfläche oder aber mit C++.

Zubehör (nicht Enthalten)

- AA Batterien (Ultra Power) oder Akkus NiMH z.B. Ansmann HR06 Typ 2700
- Ladegerät MW3310HC / 1 A Ladestrom Einstellung
- USB-Kabel (Micro-B / Type A)

Sensor Modul Erweiterungen:

Extension Board (Arduino kompatibel, iRP programmierbar)

- Interface mit Buchse für Barometer oder GPS Modul vorbereitet
- für eigene Hardware-Erweiterungen

2 Anleitung

2.1 Marvin Roboter Hardware

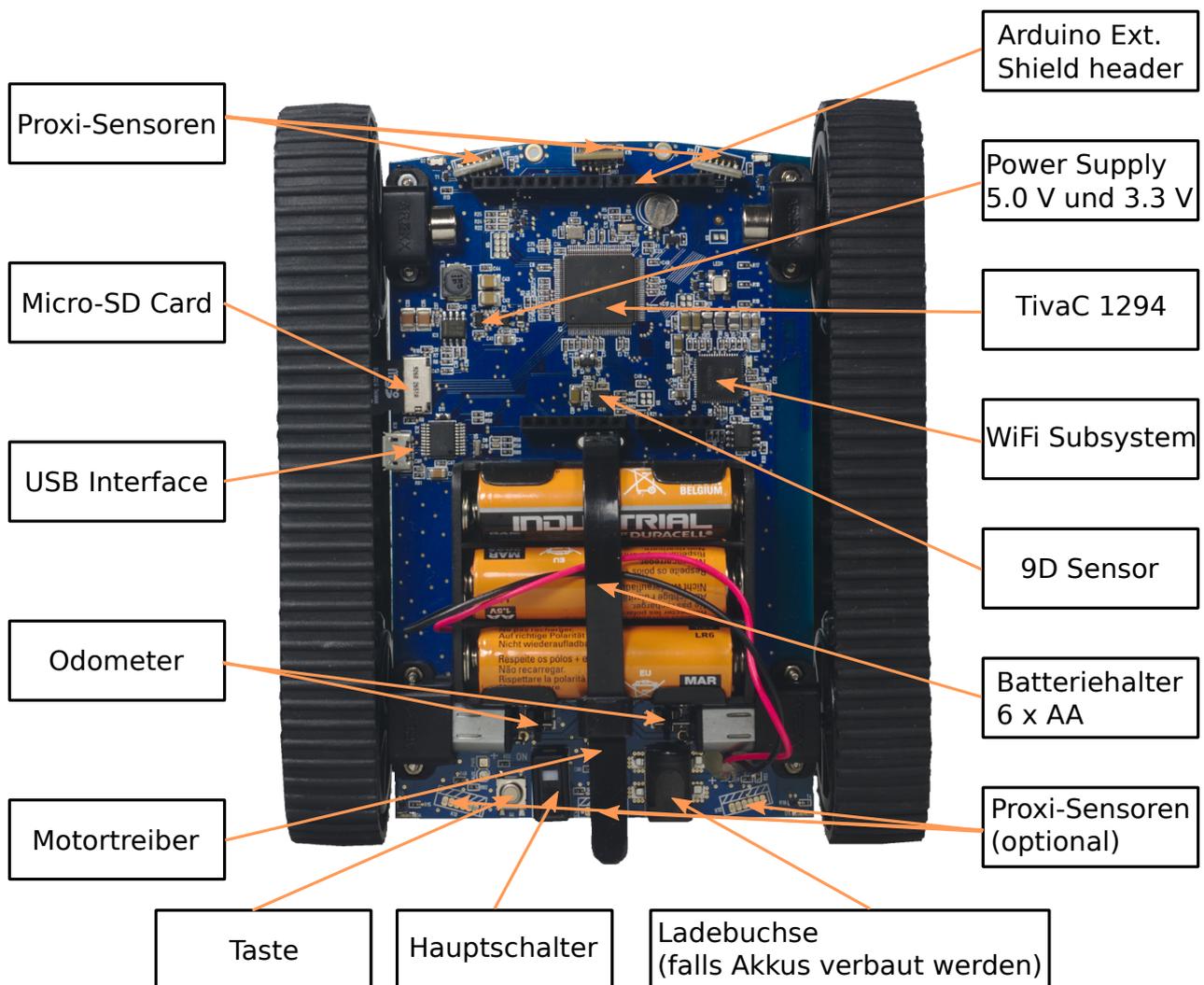


Abbildung 2.1 Marvin Roboter Hardware

Hinweis: Am bestem fassen Sie den Marvin-Roboter am Batteriefach oder an den Ketten an. Außerdem sollte man es vermeiden direkt auf die Elektronik (Mikrokontroller usw.) zu fassen (Elektrostatische Entladung - ESD!).

2.2 Marvin Roboter Ausstattung und Zubehör

2.2.1 Ausstattungsmerkmale

Der IoT Roboter stellt viele neue Möglichkeiten zu Verfügung:

- TIVA TM C- Mikrokontroller ARM Cortex-M4F Core (Floating Point) mit 512KByte Flash, 256KByte SRAM, 6KB EEPROM, CC3100 WI-FI® 802.11 b/g/n – bis zu 16 Mbps, optimal angepasste PCB-Antenne, Infrastruktur- und Ad-Hoc Mode, Flash-ROM 8Mbit
- Virtuelles Display auf PC oder Tablet-PC.
- 3x Proximity / Ambient Light Sensor (Time of Flight) vorne mittig
 - rechts, links vorne
 - optional nachrüstbar: + 3 x Hinten (rechts, links, Mitte)
- 9D - Gyro und Kompass Sensor
- RTC mit Puffer-Akku (lädt automatisch während dem Betrieb)
- 1 x RGB Status LED
- Status LEDs für Wi-Fi®
- 2 x LEDs USB (Rx, Tx)
- 2 x Head-Scheinwerfer (weiß) und 2x Rücklichter (Rot)
- USB-Programmierung / Wi-Fi Konfiguration (Micro USB Stecker)
- Micro SD-Card Reader
- SMD-Taste
- Arduino Erweiterungsstecker - separate UART, SPI, I2C; 6 ADC, bis zu 8 GPIOs
- Interrupt fähig, bis zu 4 PWM Kanäle für Servos
- 6 Zellen Spannungsversorgung für AA-Batterien oder Akkus für lange Betriebszeiten
- Schiebeschalter und AUX- Batterie Eingang
- Schaltregler für hohe Effizienz der Stromversorgung
- Kraftvolle Mikro-Getriebemotoren mit hochauflösender Odometrie für hohe Geschwindigkeit
- Gummikettenantrieb
- Anwendungsbeispiele für Marvin – iRP und Fernsteuerung über Firefox oder Google Chrome Browser (PC, Notebook oder Tablet (Android))
- Robot Programmier-Tool (unterstützt alle AREXX Roboter) und USB-Treiber für Linux und Windows 7
- C++ Software Entwicklungspaket (GCC ARM Compiler (Linux), C++ Software Library, Anwendungsbeispiele)
- Micro SD Karte (Roboter Dokumentation, iRP Software und Firmware: RobotTool, Beispiel Programme)

2.2.2 Erfassungsbereich der Proximity-Sensoren

Die Sensoren haben einen Erfassungsbereich (FoV) von 25° und einer Reichweite von ca. 30 cm. Distanzen kleiner als 10 cm werden als ‚Null‘ angezeigt. Damit ist sichergestellt dass der Roboter den Hindernissen noch gut ausweichen kann ohne rückwärts zu fahren zu müssen.

Die beiden äußeren Sensoren sind mit $+15^\circ$ bzw. -15° zur Roboter-Nullachse verdreht um eine breiteren Erfassungswinkel zu erzielen. Die Darstellung des resultierenden Erfassungsbereiches zeigt ein leichtes Überlappen der einzelnen Sensoren. Hierdurch wird sichergestellt dass alle Hindernisse im Fahrweg erkannt werden. Analog gilt das Bild auch für die optionalen Sensoren für hinten.

FoV Proximity-Sensoren (Darstellung mit allen drei Sensoren):

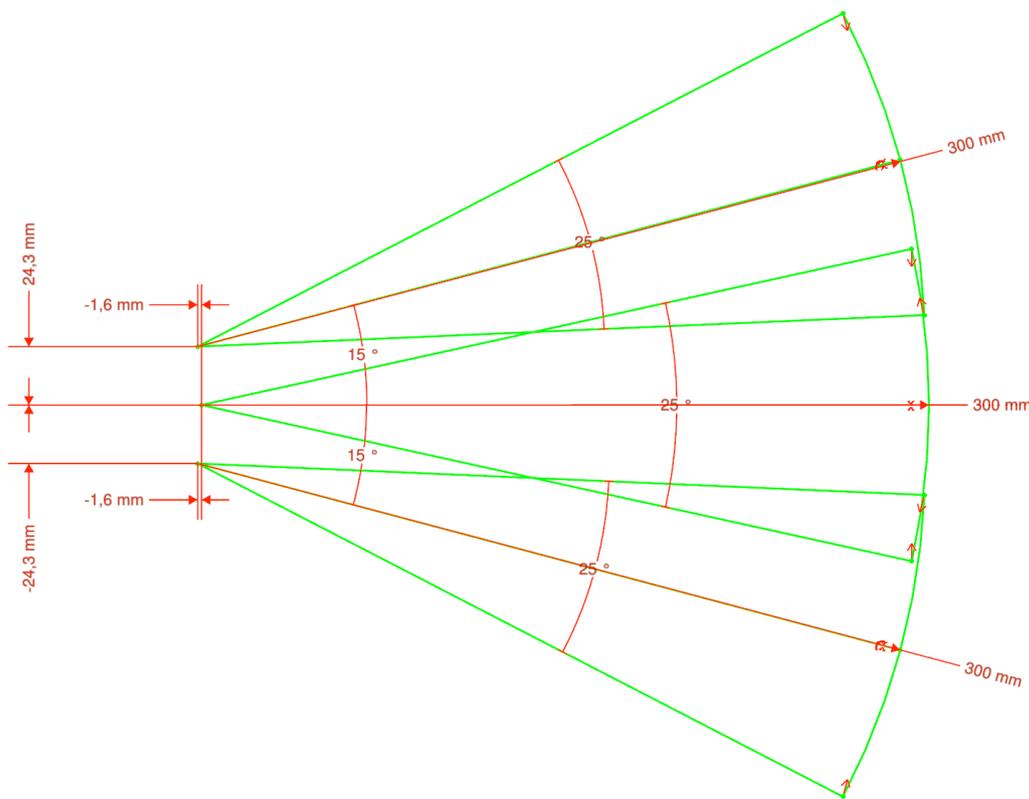


Abbildung 2.2 Field of View der vorderen Proxi Sensoren

2.2.3 Einbau zusätzlicher Proximity-Sensoren (Hinten)

Die Sensoren haben ein Rastermaß von 1,27mm – also halb so viel wie üblich. Deshalb sollte man einen LötKolben mit feiner Spitze und einer Leistung von ca. 50 W Leistung besitzen.

Achten Sie auf senkrechten Sitz der Sensoren und das sie nicht schief stehen. Es wird empfohlen zunächst nur einen Pin anlöten, dann den Sensor auszurichten und dann erst alle anderen Pins anlöten. Die Details zum aktivieren der Sensoren finden Sie in ??.

Hinweis: Vergessen Sie nicht die Sensoren noch per Software aktivieren!

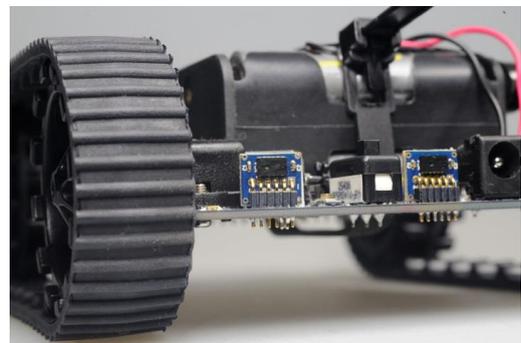
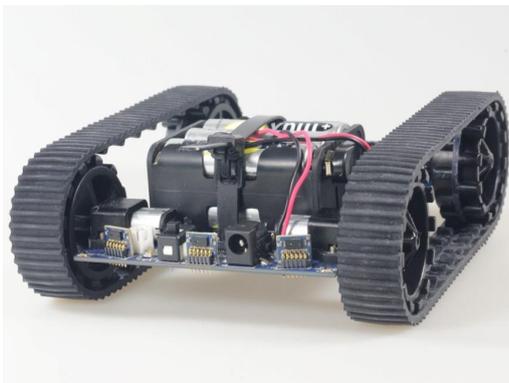


Abbildung 2.3 Marvin mit hinteren Proximity-Sensoren

Hinweis: Am besten üben Sie das Löten vorher an einer Lochraster - Leiterplatte und seien Sie sparsam mit dem Lötzinn. Zu viel Zinn kann Kurzschlüsse zwischen den PINs verursachen!

2.3 Inbetriebnahme

2.3.1 Dokumentation und Software

Die komplette Dokumentation und die Marvin – iRP Web IDE Software befinden sich auf der beigelegten Micro-SD Karte und kann über das Marvin Web-Interface heruntergeladen werden und befindet sich dann im entsprechen Order für Downloads des Computers.

Die SD Karte verwendet das von Linux bekannte ext4-Dateisystem und kann nicht von Windows PCs oder MacBooks ohne spezielle Hilfsprogramme gelesen werden. Eventuelle Updates werden über Datenträgerabbilder verteilt, die mit einem Hilfsprogramm auf die Micro SD-Karte geschrieben werden. Dazu ist ein passender Micro SD-Kartenadapter auf USB beigelegt.

Hinweis: Im Auslieferungszustand ist der Roboter auf den WiFi AdHoc-Mode eingestellt damit sie direkt mit einem PC mit Wifi-Interface kommunizieren können (z.B. Notebook, Tablet etc.).

Es gibt eine Vielzahl von Micro SD Karten die unterschiedliche Eigenschaften besitzen und die im Marvin eventuell nicht ordnungsgemäß funktionieren. SD-XC Karten sowie SanDisk Ultra Karten werden generell nicht unterstützt!

Marvin Software updates bestehen aus dem Datenträgerabbild mit der iRP Software. Außerdem ist darin auch die passende Firmware (hex-file) enthalten.

Die Versionen von iRP Software und Firmware sind immer identisch!

2.3.2 iRP WebIDE

Die Web Interface Software und die Marvin Dokumentation befinden sich schon auf der beigelegten Micro SD Karte. Falls die Micro SD-Karte nicht schon im Roboter eingesteckt ist, tun sie dies nun vorsichtig.



Abbildung 2.4 Marvin mit eingesetzter SD Karte

Hinweis: Die Micro SD-Karte muss in der Regel nicht aus dem Roboter entnommen werden. Sie ist geschützt verbaut damit sie vor Beschädigungen geschützt ist. Jedoch sollte man den Roboter nicht so anfassen das die Micro SD-Karte beschädigt wird!

2.3.3 iRP mit dem PC/Tablet verbinden

Es gibt generell zwei unterschiedliche Betriebsmodi wie der Marvin Roboter mit dem PC/Notebook über WiFi verbunden werden kann:

- WiFi - Infrastruktur Mode: Marvin mit einem Netzwerk verbinden (WiFi-Router oder Access-Point)
- WiFi - AdHoc Mode: hier können sie den Marvin direkt mit dem Notebook oder Tablet-PC verbinden.
 - SSID: Marvin
 - IP: 192.168.1.1
 - Passwort: IoT-Robot

Hinweis: **Groß- / Kleinschreibung beachten!**
Nur im AdHoc Mode ist die SSID des Marvin Roboters sichtbar, da ja im Infrastruktur-Mode sich der Marvin zu Ihrem Router verbindet und diese seine eigene SSID hat.

Bitte löschen Sie den Browser Cache falls der iRP Bildschirm nicht richtig geladen wird!

2.3.3.1 WiFi - AdHoc Mode

Dieser Modus ist bei Auslieferung des Roboters eingestellt. Er stellt die schnellste Möglichkeit dar den Roboter zu programmieren oder zu steuern.

Um den Roboter mit dem PC/Notebook zu verbinden führen Sie folgende Schritte aus:

1. PC/Notebook mit WiFi Interface einschalten
2. Roboter einschalten - die grüne WiFi Status LED leuchtet
3. Am PC/Notebook im WiFi Menü nach Funknetzwerken suchen - hier sollte der SSID Eintrag „Marvin“ zu sehen sein.
4. Koppeln Sie den Roboter mit dem PC indem sie das Passwort eingeben und die Verbindung aktivieren.
5. Browser öffnen und die IP 192.168.1.1 eingeben. Sie sollten nun Abbildung 2.5 im Browser sehen. Das war es schon – Sie können sich jetzt mit dem Menü vertraut machen und ein erstes kleines Programm laden und ausführen!
6. Laden Sie das Beispielprogramm „Lauflicht“
7. Sie sollten jetzt die Zustands LED verschieden Farben leuchten sehen die jeweils nach 1sec wechseln.

2.3.3.2 WiFi - Infrastruktur Mode

Dieser Modus muss zunächst am Roboter mit dem JM3 Robot Tool konfiguriert werden und am WiFi Router angemeldet werden. Der Marvin unterstützt DHCP – d.h. eine IP-Adresse wird automatisch zugewiesen.

Hinweis: Falls nötig - ändern Sie die Firewall-Einstellungen damit der Marvin Roboter sich mit dem Netzwerk verbinden kann!

Bitte löschen Sie den Browser Cache falls der iRP Bildschirm nicht richtig geladen wird!

Hier der Ablauf zu Umstellung auf Infrastruktur-Mode:

1. Installieren Sie das JM3-Robot Tool (Installation - siehe Kapitel 4)
2. Verbinden Sie den Roboter mit dem Rechner (Micro USB Kabel)
3. Öffnen Sie das Robot Tool, schalten Sie den Roboter Ein und stellen Sie eine Verbindung her.
4. Gehen Sie wie folgt vor:
 - a) Roboter über USB verbinden
 - b) Wechseln sie in das Terminal Fenster (Tab)
 - c) Senden Sie folgende Kommandos an den Roboter:
 - wlan DEFAULT - „Enter“
 - Schalten Sie den Roboter AUS und EIN wenn es im Fenster angezeigt wird
 - wlan STA addprofile „SSID-Ihres Routers“ „Passwort Ihres Routers“ - „Enter“

Hinweis: Achten Sie darauf das beim Absenden des „addprofile“ Befehls die Terminal History ausgeschaltet ist. Ansonsten wird Ihr Passwort im Klartext auf dem PC gespeichert!!!

Hinweis: Weitere Informationen über die Kommandos sowie die über die Eingabe von Leerzeichen finden Sie in Abschnitt 2.5

- wlan STA MODE - „Enter“
- Schalten Sie den Roboter AUS und EIN wenn es im Fenster angezeigt wird

d) Überprüfen Sie die zugewiesene IP Adresse (DHCP Router) für den Marvin Roboter

- wlan getIP - „Enter“

Die aktuelle IP Adresse wird nun im Terminalfenster angezeigt.

5. Danach öffnen Sie den Browser und geben die IP Adresse ein, zum Beispiel: 192.168.1.120
Sie sollten nun Abbildung 2.5 im Browser sehen.

Das war es schon – Sie können sich jetzt mit dem Menü vertraut machen und ein erstes kleines Programm laden und ausführen!

6. Laden Sie das Beispielprogramm „Lauflicht“

7. Sie sollten jetzt die Zustands LED verschieden Farben leuchten sehen die jeweils nach 1 sec wechseln.

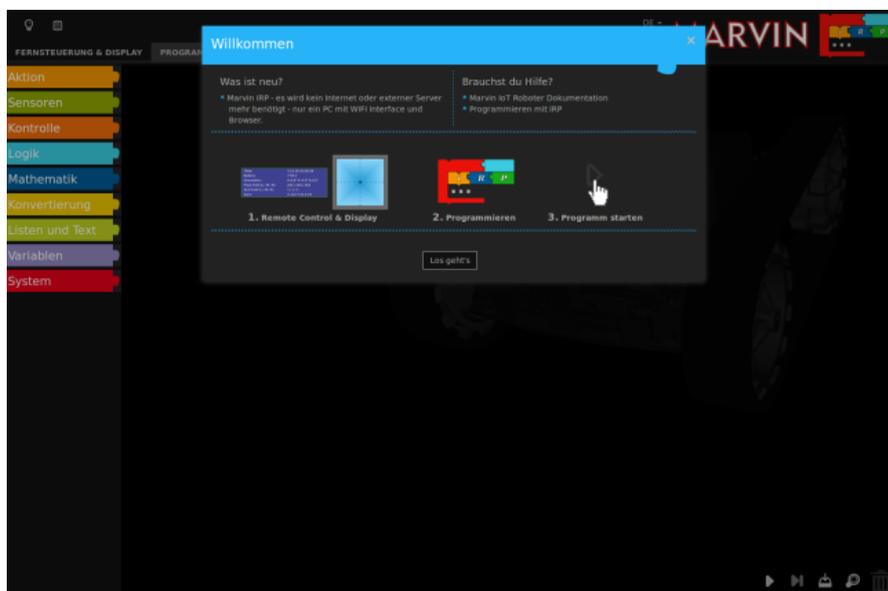


Abbildung 2.5 Marvin WebIDE Willkommensbildschirm

2.3.4 Status LEDs

Zur Zustandsanzeige des Roboters (RGB-LED):

- Blau - normale Funktion
- Lila (blinken) - Batteriespannung niedrig
- Gelb - keine SD Karte oder Funktionsbeeinträchtigung, Hinweise auf USB Interface.
- Grün - beschäftigt (SD recovery)
- Blau/Grün (blinken) - Keine Firmware
- Gelb/Rot (blinken) - Programmiert Firmware
- Rot - Funktionsstörung - bitte ein und ausschalten

Zur Zustandsanzeige des WIFI Subsystems gibt es zusätzlich drei LEDs:

- Rot - Verbindungsproblem mit WiFi Netzwerk
- Gelb (blinken) - Daten werden übertragen
- Gelb (dauer) - WiFi Verbindung hängt
- Grün - mit WiFi Netzwerk verbunden, dies ist im AdHoc Modus immer der Fall und ist unabhängig von der HTTP Verbindung.

2.4 Kalibrierung der Sensoren und stellen der Systemzeit

Der Marvin Roboter unterstützt das Kalibrieren von Kompass, Gyro (Drehratensensor) und den Abstandssensoren. Diese Funktionen werden im Fenster Fernsteuerung und Display (siehe auch Unterabschnitt 3.1.1) mit den Kommando-Buttons bzw. über das Kommando-Interface eingeleitet (Abbildung 2.6). Außerdem kann die Real Time Clock des Roboters mit der Systemzeit des PC/Notebooks synchronisiert werden.

Hinweis: Die Kalibrierung ist normalerweise nur ein einziges mal notwendig, dabei werden alle aufgenommenen Kalibrierdaten im Roboter gespeichert.

2.4.1 RTC - Systemzeit stellen

Das stellen der Real Time Clock im Marvin Roboter erfolgt durch ein drücken des Buttons „RTC Stellen“. Damit ist das stellen der Uhr abgeschlossen.



Abbildung 2.6 Kommandointerface

Hinweis: Die Uhrzeit läuft nun auch bei ausgeschaltetem Marvin für mehrere Monate weiter. Durch einschalten des Roboters wird die Batterie wieder aufgeladen.

2.4.2 Kalibrierung des 3D Kompass

Im folgenden ist die Prozedur zur Kalibrierung des 3D Kompasses beschrieben. Die Ausrichtung nach NWSE wird automatisch erkannt. Es muss nur zum starten/stoppen eine Taste gedrückt bzw. ein Kommando ausgeführt werden.

Hinweis: Die Hinweise auf die Himmelsrichtung erleichtern eine vollständige Kalibrierung - sind aber nicht zwingend erforderlich. Wichtig ist es einen vollständigen Kreis zu durchlaufen!

2.4.2.1 Schritt 1 - Vorbereitung

Schalten sie den Marvin Roboter an und verbinden Sie den Roboter mit dem Web-Interface im Browser. Nehmen sie einen Kompass und ermitteln die Himmelsrichtungen für Nord, Ost, Süd und West – merken sie sich die Punkte.

2.4.2.2 Schritt 2 - Kalibriermodus einschalten

Gehen sie zur Fernsteuer- und Display Seite und klicken sie den Button, „Kalibriere Kompass“ an.

Im Bild sehen sie die gedachten Achsen bezogen auf den Marvin. Die x-Achse ist rot, die y-Achse grün und die z-Achse blau dargestellt. Gelb ist der Inklinationsvektor dessen Winkel nicht genau bekannt ist und vom Standort auf der Erde abhängt. Liegen der Inklinationsvektor und die x-Achse auf einer Linie kann man das Maximum an Messwert erwarten. Dies ist für eine gute Kalibrierung notwendig.

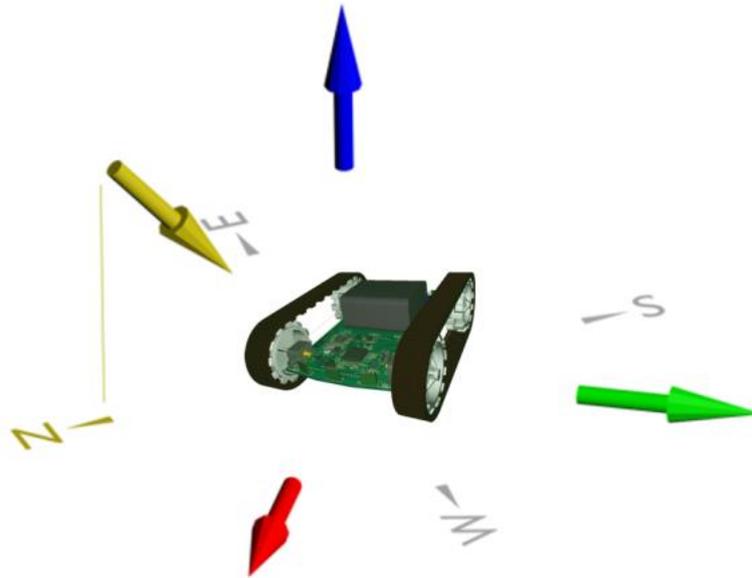


Abbildung 2.7 Himmelsrichtungen und Inklinationsvektor

2.4.2.3 Schritt 3 - Ablauf

- Halten sie dem Roboter nun nach Norden und kippen ihn jeweils um 90° nach oben und dann nach unten.
- Halten sie dem Roboter ungefähr horizontal in Richtung Norden und kippen ihn jeweils um 90° nach links und dann nach rechts.
- Wiederholen sie dies analog für Ost, Süd und West bzw. einen vollständigen Kreis - möglichst auch mit Zwischenwerten - was die Genauigkeit erhöht.

Hinweis: **Kippen sie den Marvin während der Kalibrierung nicht zu schnell !**

2.4.2.4 Schritt 4 - Abschliessen der Kalibrierung

Nachdem Sie die Prozedur durchgeführt haben beenden Sie den Kalibriermodus durch Klick auf den entsprechenden Button („Kalibrierung Abgeschlossen“) und überprüfen Sie das Ergebnis indem Sie den Roboter auf die entsprechende und bekannte Himmelsrichtung ausrichten.

2.4.3 Kalibrierung des 3D Gyroskops

Das Kalibrieren des Gyros erfolgt durch ein drücken des Buttons „Kalibriere Gyro Offset“. Im Button wird nun „Kalibriere - bitte warten“ angezeigt - dies erfolgt solange bis der Vorgang abgeschlossen ist.

Hinweis: Die Kalibrierung läuft selbstständig ab. Dabei darf natürlich der Roboter nicht bewegt werden.

2.4.4 Aktivierung der Proximity Sensoren

Das aktivieren oder deaktivieren der Proximity Sensoren erfolgt mittels Kommandozeilenbefehl - dem „conf“ Befehl. Details sind in Tabelle 2.1. Um zum Beispiel den Sensor in der Mitte hinten zu aktivieren, geben Sie in die Kommandozeile folgendes ein:

- conf add RM „Enter“ und anschliessend
- conf save RM „Enter“

Damit hätten Sie den Sensor RM (Hinten Mitte) hinzugefügt und die Konfiguration dauerhaft gespeichert und der Sensor ist betriebsbereit.

2.4.5 Kalibrierung der Abstandssensoren

Im Folgendem wird die Kalibrierung der Abstandssensoren beschrieben. Hierzu muss ein iRP Hilfsprogramm ausgeführt und die Messwerte für jeden Sensor notiert werden. Außerdem benötigt man ein Lineal mit 30 cm bis 50 cm Länge. Anschliessend werden die Werte mit dem Kommando-Interface Abbildung 2.6 in den Roboter geschrieben und dort gespeichert.

Hinweis: Die Kalibrierung der Abstandssensoren ist nicht zwingend erforderlich und stellt eine Option dar genauere Messergebnisse zu erhalten. Anfänger sollten das erst einmal nicht tun.

2.4.5.1 Schritt 1 - Vorbereitung

Schalten sie den Marvin Roboter an und verbinden Sie den Roboter mit dem Web-Interface im Browser.

- Legen sie das Lineal vor den jeweiligen Sensor, so dass die gedachte Sichtlinie des Sensors mit dem Lineal deckungsgleich sind.
- Starten Sie das iRP Programm „Abstandssensorkalibrierung“
- Gehen sie zur Fernsteuer- und Display Seite.

Im unteren Display Abbildung 2.8 werden die Abstandsrohwerter angezeigt:

- FM (Vorne Mitte) 0 - 255
- FL (Vorne links) 0 - 255
- FR (Vorne rechts) 0 - 255
- RM (Hinten Mitte) 0 - 255
- RL (Hinten links) 0 - 255
- RR (Hinten rechts) 0 - 255

Hinweis: Sie können die weiße Marvin Box zur Abstandsmessung verwenden!

2.4.5.2 Schritt 2 - Ablauf

Messen Sie nun den jeweiligen minimal und maximal Wert des Erfassungsbereiches aus.

- Dazu nähern Sie das Hindernis (Marvin Box) von ca. 35 cm aus an bis sich im Display der Rohwert auf einen kleineren Wert als 255 einstellt.
- Suchen sie den exakten Punkt und schreiben Sie den Rohwert und den gemessenen Abstand am Lineal auf.
- Nähern Sie das Hindernis (Marvin Box) weiter an bis sich im Display der Rohwert nicht mehr ändert (Minimalwert).
- Suchen sie den exakten Punkt und schreiben Sie den Rohwert und den gemessenen Abstand am Lineal auf.

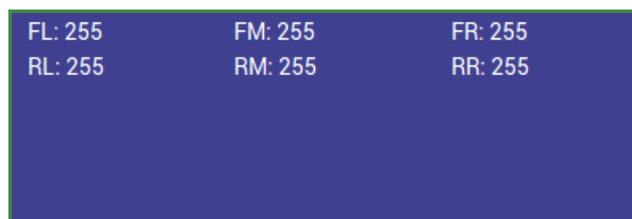


Abbildung 2.8 Abstandssensor Hilfsprogramm Bildschirm

Geben Sie in die Kommandozeile folgendes ein:

- `cal prox FM 23 13.5 300 170 „Enter“`

In diesem Beispiel würde der Sensor (FM) Front Mid mit dem unteren Rohmesswert von „23“, dem gemessenen Abstand „13.5“ und dem gemessenen Abstand „300“ mit dem oberen Rohmesswert von „170“ zur Kalibrierung verwendet werden.

Hinweis: **Alle Abstände sind in „mm“ einzugeben!**

2.4.5.3 Schritt 3 - Überprüfen der Kalibrierung

Nachdem Sie die Prozedur durchgeführt überprüfen Sie nun das Ergebnis für den linke, rechten und mittig sitzenden Sensor. Bei:

- ca. 13 cm muss 0
- ca. 30 cm muss 30
- > 30 cm muss 255 angezeigt werden.

2.5 Kommandozeileninterface (CLI)

Im Tabelle 2.1 sind alle Kommandos mit Syntax beschrieben die zur Verfügung stehen.

Tabelle 2.1 CLI Kommandos

Kommando	Beschreibung
cal cmps	Aktiviert/Deaktiviert Kompass Kalibriermodus.
cal gyro [--state]	Kalibriert Gyro Offsets. Wenn der Parameter „-state“ angegeben ist wird der aktuelle Fortschritt ausgegeben.
cal prox „sensor“ „xNear“ „mmNear“ „xFar“ „mmFar“	Kalibriert Sensor: FL/FM/FR/RL/RM/RR.
conf „cmd“ „sensor“	Konfiguriert die Proxi Sensoren: FL/FM/FR/RL/RM/RR. Kommandos (cmd): „add“ „del“ „read“ „save“
light „on/off“	Schaltet Fahrlichter an/aus.
setrtc „H_M_S_W_D_M_Y“	Setzte Marvin Uhrzeit.
wlan AP ssid „SSID“	Setzt SSID im AdHoc Modus auf „SSID“.
wlan AP passwd „passwd“	Setzt Passwort im AdHoc Modus auf „passwd“.
wlan AP txpwr „pwr“	Setzt Sendeleistung im AdHoc Modus auf „pwr“. „pwr“ ist eine Zahl zwischen 0 und 15. 0 ist maximale Sendeleistung. Maximale Effizienz ist bei „pwr“ = 4.
wlan AP channel „ch“	Setzt den WLAN Kanal im AdHoc Modus. „ch“ ist eine Zahl zwischen 0 und 13. „ch“ = 0 für automatische Kanalauswahl.
wlan AP MODE	Schaltet Marvin in den AdHoc Modus.
wlan STA addprofile „SSID“ „passwd“	Registriert das Netzwerk mit der SSID „SSID“ und dem Passwort „passwd“. In Zukunft wird automatisch versucht sich mit diesem Netzwerk zu Verbinden.
wlan STA delprofile „ID“	Löscht das WLAN Profile mit der ID „ID“. Die ID -1 löscht alle Profile.
wlan STA txpwr „pwr“	Setzt Sendeleistung im Infrastruktur Modus auf „pwr“. „pwr“ ist eine Zahl zwischen 0 und 15. 0 ist maximale Sendeleistung. Maximale Effizienz ist bei „pwr“ = 4.
wlan STA ipcfg „DHCP“ „IP“ „mask“ „gw“ „dns“	Aktiviert im Infrastrukturmodus DHCP oder setzt eine statische IP-Adresse. „IP“, „mask“, „gw“ und „dns“ sind IPv4 Adressen.
wlan STA MODE	Schaltet Marvin in den Infrastruktur Modus.
wlan STA SCAN	Scant nach verfügbaren WLAN Netzwerken.
wlan DEFAULT	Setzt das WLAN Modul zurück auf Werkseinstellungen.
wlan getIP	Gibt die Aktuelle IP Adresse aus.
version	Gibt die Firmware Versionsnummer aus.
security updatekey „key.der“	Kopiert HTTPS Schlüssel von der SD-Karte in das WiFi Modul.
security updatecert „cert.der“	Kopiert HTTPS Zertifikat von der SD-Karte in das WiFi Modul.
security httpasswd „user:passwd“	Setzte Web Server Benutzername und Passwort.

Die „security“ Befehle sind nur im C++ Modus unter Linux verfügbar.

3 Programmieren mit iRP

Die grafische Programmiersprache Marvin - iRP ist einfach zu erlernen und bedarf keiner Vorkenntnisse einer Programmiersprache wie z.B. C/C++.

Die verschiedenen Funktionsblöcke erlauben es auf einer logischen Ebene Programme zu erstellen und auszuführen.

3.1 Einführung in iRP

3.1.1 Grundlegende Bedienung

Die Bedienung von PC und Browser wird vorausgesetzt.

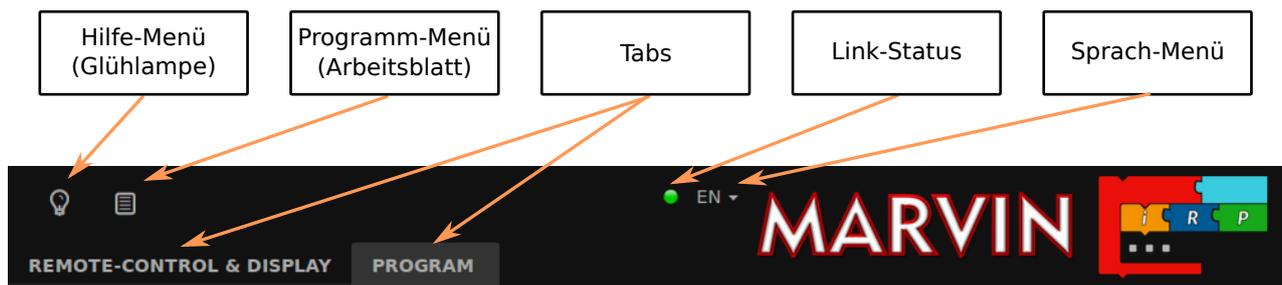


Abbildung 3.1 Marvin WebIDE Menüleiste

Der Link-Indikator befindet sich links neben dem Sprach-Menü.

- Ein grünes Licht zeigt eine gute Verbindung an.
- Ein rotes Licht zeigt eine kurzfristige Unterbrechung oder Benutzung der gesamten Bandbreite zum Download von Dokumenten von der SD-Karte an.
- Falls eine längere Störung vorliegt wird eine Meldung (Pop-up) auf dem Bildschirm angezeigt.

3.1.1.1 Das Spracheinstellungs-Symbol

- Die Sprache können Sie einfach über das Sprachauswahlfeld einstellen.
- Außerdem sieht man unten rechts die Zoom und Pan Buttons.

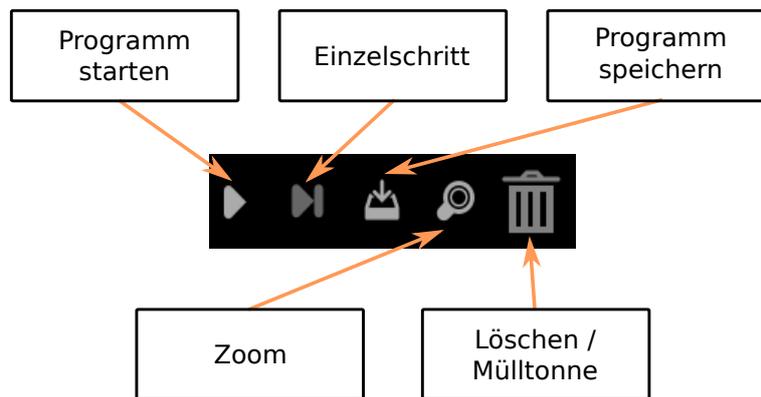


Abbildung 3.2 Marvin WebIDE Programmbuttons

Hinweis: Weitere Sprachen wie Spanisch, Italienisch, Französisch und Chinesisch (vereinfacht) werden unterstützt. Die Online-Hilfe ist neben Deutsch zur Zeit nur in Englisch verfügbar.

3.1.1.2 Das Glühlampen-Symbol

- Hier finden Sie die Hilfefunktion zu den iRP Blöcken mit kurzen Erklärungen.
- Hinweise zur Dokumentation und Software.
- Allgemeine Hinweise zum Programm (Info) und können sich den Startbildschirm nochmal anzeigen lassen.

3.1.1.3 Das Arbeitsblatt-Symbol

- Hier können Sie ihre Programme bearbeiten, z.B. laden, speichern etc..
- Marvin SD-Karten Programme (Menue zum laden und speichern)
- Außerdem können Sie hier den iRP-Modus (Beginner, Fortgeschrittener, Experte) einstellen. Die dunkelgraue Farbe zeigt die Auswahl an.

3.1.1.4 Fernsteuerung und Display Tab

- Hier gibt es zwei Displays, einen Fernsteuerfeld und ein Plotter-Bereich.
- Die zwei virtuellen Displays unterteilen sich in einen festen Bereich in dem Systemwerte angezeigt werden (z.B. die Batteriespannung) und ein Benutzer Bereich (User Display) auf. Hier kann man selbst Werte aus seinem Programm heraus anzeigen.

- Außerdem kann man hier Einstellungen vornehmen und Kommandos senden (siehe Abschnitt 2.5).

3.1.1.5 Programm Tab

- Hier können Sie ihre Programme bearbeiten, z.B. laden, speichern etc. .
- Außerdem finden sie hier iRP-Modus (Beginner, Fortgeschrittener, Experte) einstellen.

3.1.2 iRP Hilfe

Eine Hilfe zu den iRP Blöcken ist immer einfach im Browser aufrufbar.

- Klicken auf das Glühlampensymbol (Bild - oben links gezeigt) und dann auf Hilfe oder
- Klicken Sie auf das Fragezeichensymbol am rechten Bildschirmrand.

Die Hilfe scrollt automatisch immer zum gerade verwendeten Block (Abbildung 3.3).

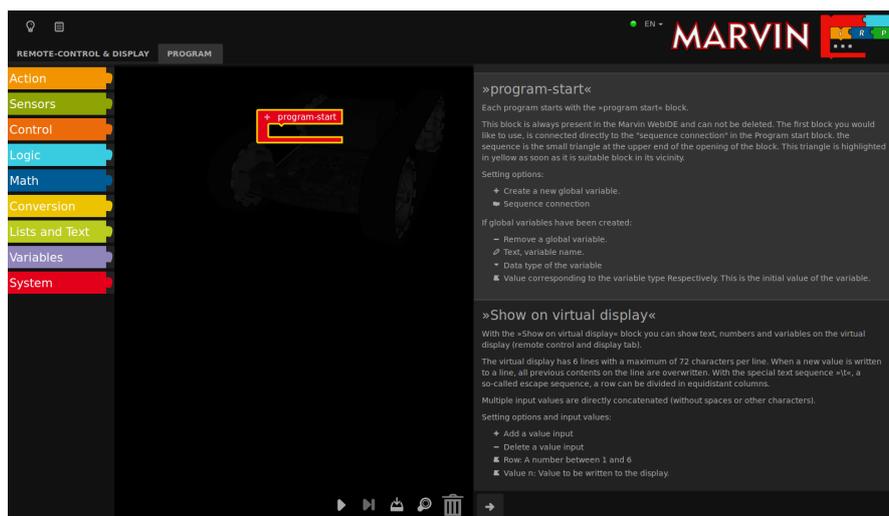


Abbildung 3.3 Der Marvin WebIDE Helpbrowser

3.1.3 Mein erstes Programm

- Ein neues Programm wird einfach aus den iRP Blöcken zusammen gefügt.
- Passen Blöcke logisch nicht zusammen kann man sie nicht an den anderen Block anhängen. Im Beispiel passt der Block „Reelle Zahl“ logisch nicht zu einer „Ganzen Zahl“ (Abbildung 3.4).



Abbildung 3.4 Nicht passende Blocktypen in iRP

3.1.4 Programm ausführen

- Ein Programm kann man über das Menü im Programm Tab oder den Start Button erfolgen.
- Neben dem Start Button befindet sich von ein Step Button (Schrittweise Ausführung des Programms bis zum Haltepunkt (BreakPoint), der Speicher Button, die Zoom Funktion und die Mülltonne – für nicht mehr benötigte Blöcke.

3.1.5 Programm laden und speichern

- Hier können Sie ihre Programme laden oder speichern. Der Speicherort kann wie gewohnt ausgewählt werden.
- Interessant ist auch die Möglichkeit schon erstellte Programmteile einzufügen. So kann man aus verschiedenen Programm-Modulen ein umfangreicheres Programm erstellen - siehe Beispiel: „Ein Quadrat nach Kompass fahren“.

3.1.6 Programmfehler beheben (Debugging)

Tritt bei der Programmerstellung ein Fehler auf so wird ein Hinweis gegeben und der entsprechende Block hervorgehoben (siehe Abbildung 3.5)

Hinweis: Ein sehr nützliches Feature ist die Möglichkeit Breakpoints zu setzen um das Programm anzuhalten und einen interessanten Wert oder Zustand im virtuellen Display anzuzeigen. Das vereinfacht das Debuggen sehr da so leicht interne Zustände der Software dargestellt werden können!

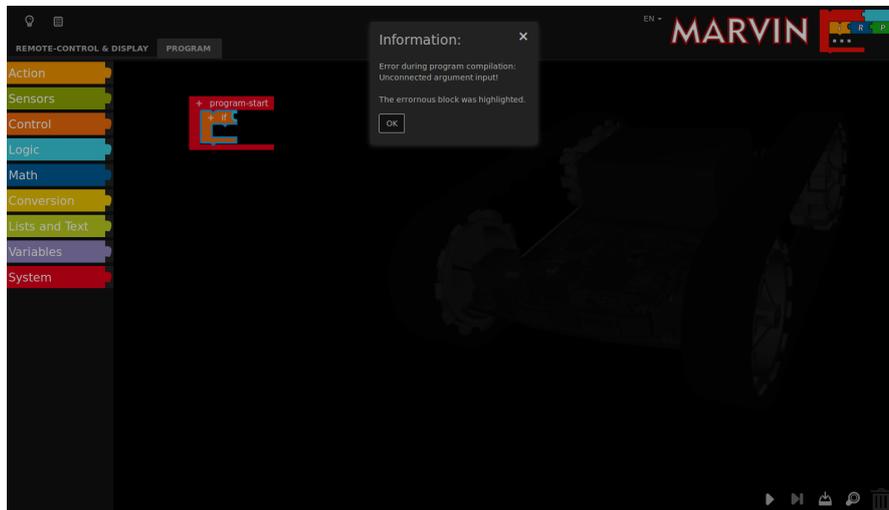


Abbildung 3.5 Beispiel für einen Compiler Fehler

3.1.7 Programmcode (Quelltext) anzeigen

Zum weiteren Debuggen kann es hilfreich sein sich den Quelltext anzusehen – ist aber eher etwas für Fortgeschrittene und Experten. Die Auswahl erfolgt über Programm Tab Quelltext - dabei sind Ausgaben als C++ Source, C++ Header and Assembler möglich (Abbildung 3.6).

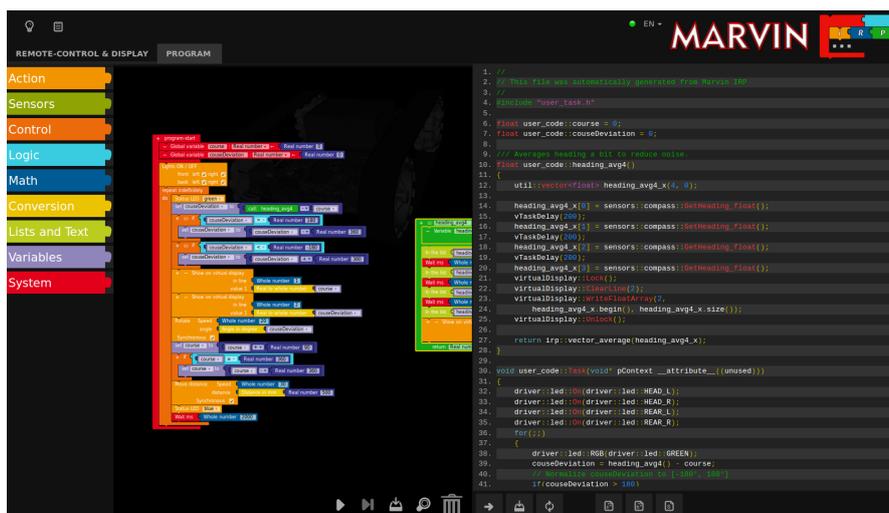


Abbildung 3.6 Beispiel für generierten C++ Code

4 JM3 Robot-Tool 2.0

4.1 Linux

- Kopieren sie das JM3 Robot Tool in einen Ordner und führen sie das Programm „launch_robotTool.sh“ aus!

Hinweis: **Details siehe [install.txt](#)**

- Klick auf das „Add Robot“ Icon - geben sie den Namen (frei wählbar), den Hostnamen z.B. den verwendeten USB Port ein und gehen zum nächsten Schritt.
- Wählen sie den Roboter-Typ, z.B. „Marvin“, aus. Den richtigen USB Port können sie im System nachsehen - üblich ist der USB Port „/dev/ttyACM0“.
Als Interface muss „UART“ mit einer Baudrate „0“ (default) eingestellt werden.
- Klick OK
- Klick auf „Add File“ um das hex-file mit dem neuen Programm auszuwählen. Die Suche geht einfach über den Pfad mit Select File. Wählen sie das hex-file aus dass sie in den Mikro Controller laden möchten. Als weiteren Schritt noch den Typ, z.B. „Marvin“, aus.
- Klick OK.
- Wähle der Roboter und das Programm in den Listen Links aus.
- Klick „Upload program“ (oben in der Toolbar)
- Klick auf „Save“ (oben in der Toolbar) falls Sie die angelegten Roboter und Programme speichern wollen.

4.2 Windows 7

- Kopieren sie das JM3 Robot Tool in einen Ordner und führen sie die Datei „robottool.exe“ aus.
- Klick auf das „Add Robot“ Icon - geben sie den Namen (frei wählbar), den Hostnamen, den USB Port z.B. „COM5“ ein und gehen zum nächsten Schritt.
- Wählen sie den Roboter-Typ, z.B. „Marvin“, und als Serial Port „UART“ aus. Die Baudrate sollte auf „0“ stehen (default). Unter Windows können Sie den richtigen COM Port im Device Manager nachsehen.

- Klick OK
- Klick auf „Add File“ um das hex-file mit dem neuen Programm auszuwählen. Die Suche geht einfach über den Pfad mit Select File. Wählen sie das hex-file aus dass sie in den Micro Controller laden möchten. Als weiteren Schritt noch den Typ, z.B. „Marvin“, aus.
- Klick OK.
- Wähle der Roboter und das Programm in den Listen links aus.
- Klick „Upload program“ (oben in der Toolbar)
- Klick auf „Save“ (oben in der Toolbar) falls Sie die angelegten Roboter und Programme speichern wollen.

4.3 Laden von eigenen Programmen die unter C/C++ erstellt wurden

Um ein selbst geschriebenes Programm (hex-file) in den Mikro-Kontroller zu laden muss vorher das **JM3 Robot Tool** installiert worden sein (Abbildung 4.1).

4.3.1 Upload (Marvin Applikation)

- Starten sie das **JM3 Robot Tool** Es wird davon ausgegangen, dass das Robot Tool schon wie vorher beschrieben vorbereitet ist.
- Klick auf „Add File“ um das hex-file mit dem neuen Programm auszuwählen. Die Suche geht einfach über den Pfad mit „Select File“. Wählen sie das hex-file aus dass sie in den Mikrokontroller laden möchten. Als weiteren Schritt wählen Sie noch den Typ, z.B. „Marvin“, aus.
- Klick OK.
- Wählen sie nun den Roboter-Eintrag aus der Liste aus z.B. „Marvin“ und das hex-File aus indem sie es nacheinander anklicken (werden leicht hellgrau hervorgehoben) und klicken sie auf „Upload“.
- Klick auf „Save“ (oben in der Toolbar) falls Sie die angelegten Roboter und Programme speichern wollen.

4.3.2 Upgrade Firmware (Bootloader)

Die Firmware des Marvin Roboters erlaubt es auch den Bootloader auf eine neue Version zu flashen. Dazu sind folgende Schritte notwendig.

- Neues Bootloader (hex-File) wie oben beschrieben auswählen.
- Klick auf „Upgrade Firmware“ führt den Update aus.

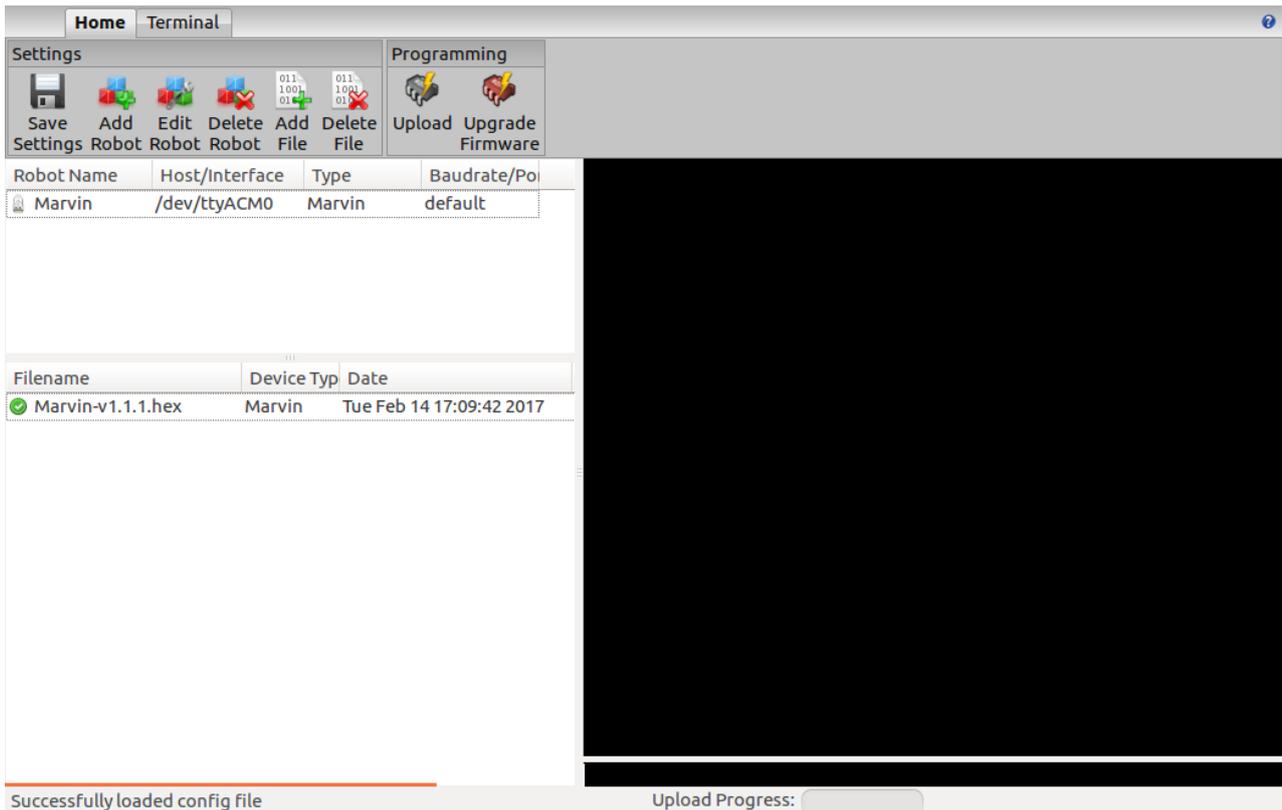


Abbildung 4.1 Das JM3 Robot-Tool

4.4 Terminal Fenster

Das Terminal im JM3 RobotTool erlaubt es serielle Daten zu empfangen und zu senden. Außerdem kann es Daten aufzeichnen und in eine Datei speichern. Verschiedene Einstellmöglichkeiten stehen zu Verfügung „Settings“:

- Verbindung mit Roboter herstellen „Connect“ erlaubt es Daten zu empfangen.
- Verbindung mit Roboter trennen „Disconnect“.
- Daten zum Roboter senden „Rechte Seite - unteres Fenster“ - Kommandos können hier jederzeit eingegeben werden. Bitte Zeilenende Zeichen auf „LF“ (Linefeed) einstellen.
- Das Programm unterstützt „undo“ mit den Pfeiltasten (up und down) und „refresh“ mit F5.

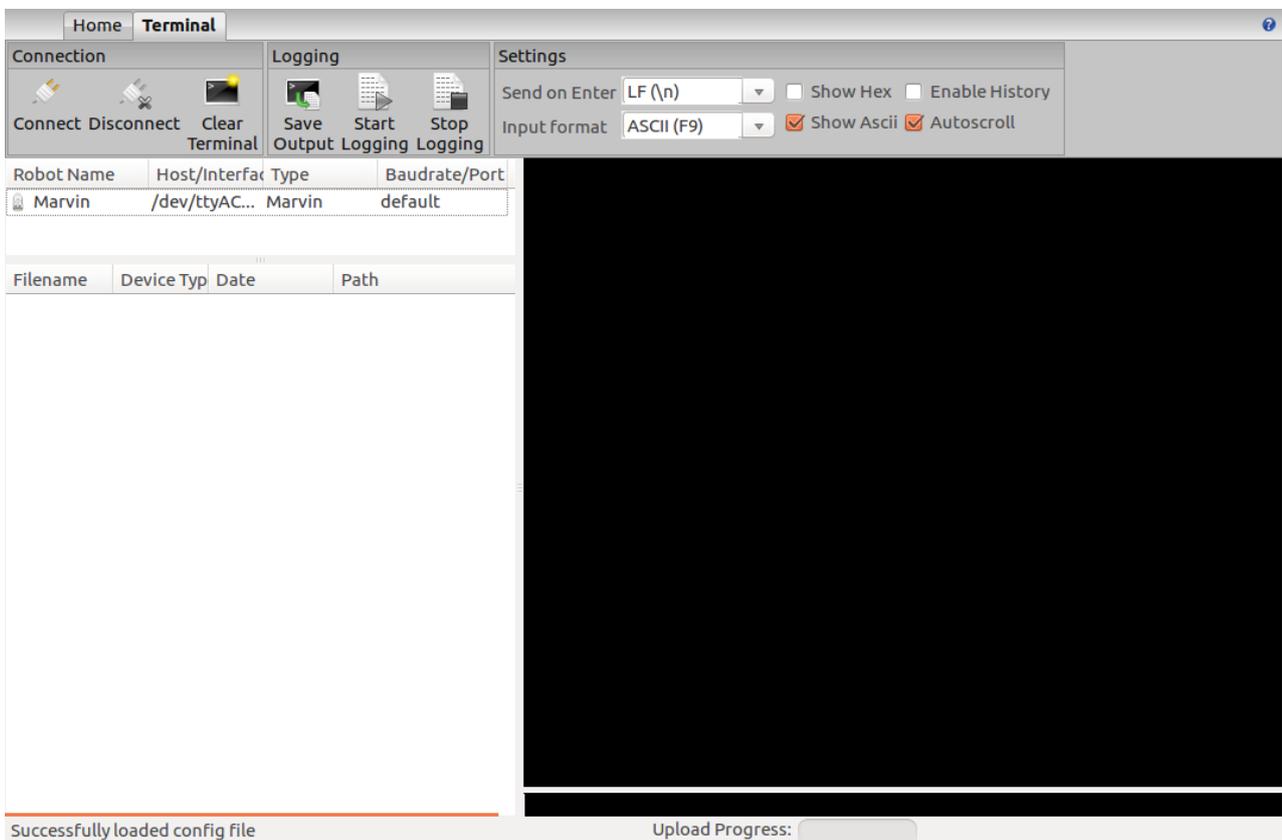


Abbildung 4.2 Das JM3 Robot-Tool - Terminal Fenster

4.5 Firmware und iRP Micro SD-Karten update

Im folgenden wird die Vorgehensweise eines Software Update beschrieben.

Firmware update:

Das Firmware update erfolgt mit dem JM3 Robot Tool. Details sind in Kapitel 4 beschrieben.

- Stellen Sie eine Verbindung mit dem PC/Notebook mit einem USB Kabel (Mirco-B / TypeA) her.
- Starten Sie das JM3 Robot Tool und wählen sie die „Firmware.hex“ Datei aus.
- Schalten Sie den Marvin Roboter ein und verbinden sich mit den PC/Notebook.
- Starten Sie den mit eine Click auf das „Upload Icon“.

Micro SD-Karten update (Windows):

Das SD-Karten Abbild wird mit dem Tool „win32DiskImager“ durchgeführt.

- Starten Sie das Programm und wählen das neue SD-Karten Abbild aus.
- Starten Sie den Sie den Abbild-Update.
- Warten Sie bis das Abbild vollständig kopiert wurde.
- Bitte Laufwerk „auswerfen“.

Micro SD-Karten Update (Ubuntu-Linux):

Das SD-Karten Abbild wird mit einem Click auf die Abbild-Datei (*.img) gestartet. Danach geht ein Fenster „Laufwerksabbild wiederherstellen“ auf.

- Wählen Sie Ihre Micro SD-Karte aus.
- Starten Sie den Abbild-Update mit „Wiederherstellung starten ...“
- Warten Sie bis das Abbild vollständig kopiert wurde.
- Bitte Laufwerk „auswerfen“ (sollte automatisch geschehen).

Hinweis: **Firmware und iRP SD-Karten Abbild gehören immer zusammen!**
Nach einem Update immer den Browser Cache löschen!

In anderen Linux Distributionen kann der Expertenbefehl „dd“ zum Imageschreiben verwendet werden.

5 Option: C/C++ Software

Die C++ Programmiersprache ist aus der C-Sprache hervorgegangen und stellt eine Erweiterung und Verbesserung von C dar.

Alles im allen die modernere Sprache die einen bessere Lesbarkeit des Codes und eine deutlich besseren Schutz vor Seiteneffekten bietet (e.g. enums / namespaces anstelle von unübersichtlichen #define Anweisungen). Assembler- und C-Programmteile können leicht mit eingebunden werden.

Spezielle Vorteile und Reduzierung von Program-Code wird bei der Verwendung von mehreren Instanzen einen H/W Treiber erreicht – z.B. zwei UART Treiber. Eine leichtere Portierbarkeit des entwickelten Programmcodes stellt einen weiteren Vorteil dar.

Die verwendete Extended Embedded C++ Implementation verbietet die folgenden C++ Features die Sie evtl. von PC-Programmierung her kennen – welche aber nicht sinnvoll für Embedded Systeme sind:

- RTTI
- Exceptions

Die zur Erstellung von Programmen benötigten Libraries (H/W Treiber etc.) und ein paar Beispielprogramme werden mit dem Paket geliefert.

Hinweis: Diese Option ist für Fortgeschrittene und Experten gedacht.

C/C++ Softwareentwicklung und das Realtime Operating System FreeRTOS sind nicht für Beginner geeignet!

Das Wissen zum aufsetzen der gesamten Linux basierten Toolchain und GCC Compiler inklusive deren Konfiguration wird vorausgesetzt.

5.1 Software Paket für Marvin

Das Marvin Software Paket besteht aus der Web-Interface Software, den Betriebssystem (FreeRTOS) mit API Funktionen und einer Library mit allen Hardware Treibern wie z.B. das virtuelle Display, ADC, Timer, I2C, UART, LED und Keypad. Für weitere Informationen und Möglichkeiten lesen sie bitte in den Beschreibungen der Module und in den entsprechenden Datenblättern nach!

5.1.1 Toolchain

Als Compiler muss der GCC für ARM (arm-none-eabi-gcc) unter Linux verwendet werden. Ausserdem werden die folgenden Pakete benötigt: make, newlib, arm-none-eabi-gcc und python. Es kann ein beliebiger Editor für den Source-Code verwendet werden.

Die Programmierung erfolgt über den USB Anschluss und dem **JM3 Robot Tool**. Ihr Programm kann damit schnell und effektiv in den TIVA C geladen werden. Wer den vollen Zugriff auf den Mikrokontroller benötigt braucht noch ein TIVA JTAG Interface (z.B.: TM4C1294 „Connected-Launch-Pad“ EK-TM4C1294XL). Außerdem muss man den 8 PIN JTAG Header (RM1.27) selbst einlöten und ein Verbindungskabel erstellen.

5.1.2 Software-Library

Die Software-Library enthält auch Funktionen wie Kompass, Neigungsmesser, Real Time Clock, Motorsteuerung und ein User Interface für die Konfiguration der WiFi Schnittstelle.

Das Webinterface kann verschiedene Werte auf dem virtuellen Display im Browser anzeigen.

Alle aus iRP bekannten Funktionen (Blöcke) und weitere sind in der Library verfügbar. C++ Code der mit iRP generiert wurde kann exportiert werden und in ein C++-Projekt übernommen werden.

Dies ist auch einer Erleichterung für den Umstieg von iRP auf C++.

5.1.3 Marvin Funktionstasten

Die Marvin Funktionstasten im Tab Fernsteuerung und Display Abbildung 5.1 können nur mit eigenen C++ Programmen verwendet werden.

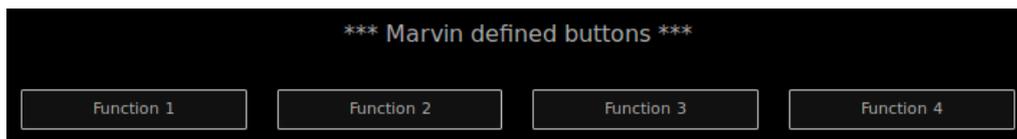


Abbildung 5.1 Marvin Funktionstasten Im Fernsteuer und Display Tab

5.2 Demo Software Beschreibung

Die Demoprogramme von iRP können exportiert werden und im ihren C++ Programm verwendet werden. Als virtuelles Display kann weiterhin die iRP Web-IDE - „Fernsteuerung und Display“ verwendet werden.

6 Technische Daten

6.1 Abmessungen und Gewicht

Breite:	125 mm
Länge:	148 mm
Höhe:	50 mm
Gewicht:	195 g (ohne Batterien)

6.2 Spannungsversorgung und Strombedarf

VCC = 8.4 V ± 5% => 6 AA Zellen

Hinweis: Das absolute Maximum liegt bei 10.0 V !!!

ICC_{AVR} = 160mA +40.0 mA / -20.0 mA => ohne Arduino Extension

Hinweis: **Batterielaufzeit:**
ca. 5 h (Fahrbetrieb - Motoren 50%) ca. 10 h bei Programmierung mit iRP

6.3 Versorgungsspannungen Experimentierboard

VDD_3V3:	I _{out,max}	≤	50 mA
VDD_5V0:	I _{out,max}	≤	50 mA
VBat_M (VSS):	I _{out,max}	≤	200 mA

6.4 Weitere Daten

Alle anderen Daten sind aus den folgenden IC Datenblättern zu entnehmen!

Controller:	Texas Instruments	TM4C1294KCPDT
Motor Driver:	Texas Instruments	DRV8833CPWP
9D-Sensor:	ST Microelectronics	LSM9DS1TR
Proxi-Sensor:	ST Microelectronics	VL6180X

7 Schaltungsdetails

7.1 Arduino Extension Board

Arduino Shields werden hier von den H/W Anforderungen unterstützt. Alle gängigen Schnittstellen wie I2C, SPI, UART ADC und GPIO bzw. Verschiedene Timer-Ausgänge ermöglichen die Frequenzausgabe oder eine PWM Generierung z.B. für Servo-Ansteuerung. Zur Messung von Frequenzen und Tast-Verhältnissen steht eine Input-Capture-Funktion zur Verfügung.

Tabelle 7.1 Pinout des Arduino kompatiblen Steckers

Pin 1	=	IOREF	Pin 17	=	IO_1
Pin 2	=	RESET	Pin 18	=	IO_2
Pin 3	=	VDD_3V3	Pin 19	=	IO_3
Pin 4	=	VDD_5V0	Pin 20	=	IO_4
Pin 5	=	GND	Pin 21	=	IO_5
Pin 6	=	GND	Pin 22	=	IO_6
Pin 7	=	VSS (V_Bat)	Pin 23	=	IO_7
Pin 8	=	ADC_X5	Pin 24	=	CS_X
Pin 9	=	ADC_X4	Pin 25	=	MOSI_X
Pin 10	=	ADC_X3	Pin 26	=	MISO_X
Pin 11	=	ADC_X2	Pin 27	=	SCK_X
Pin 12	=	ADC_X1	Pin 28	=	GND
Pin 13	=	ADC_X0	Pin 29	=	NC
Pin 14	=	RX_X	Pin 30	=	SDA_X
Pin 15	=	TX_X	Pin 31	=	SCL_X
Pin 16	=	IO_0			

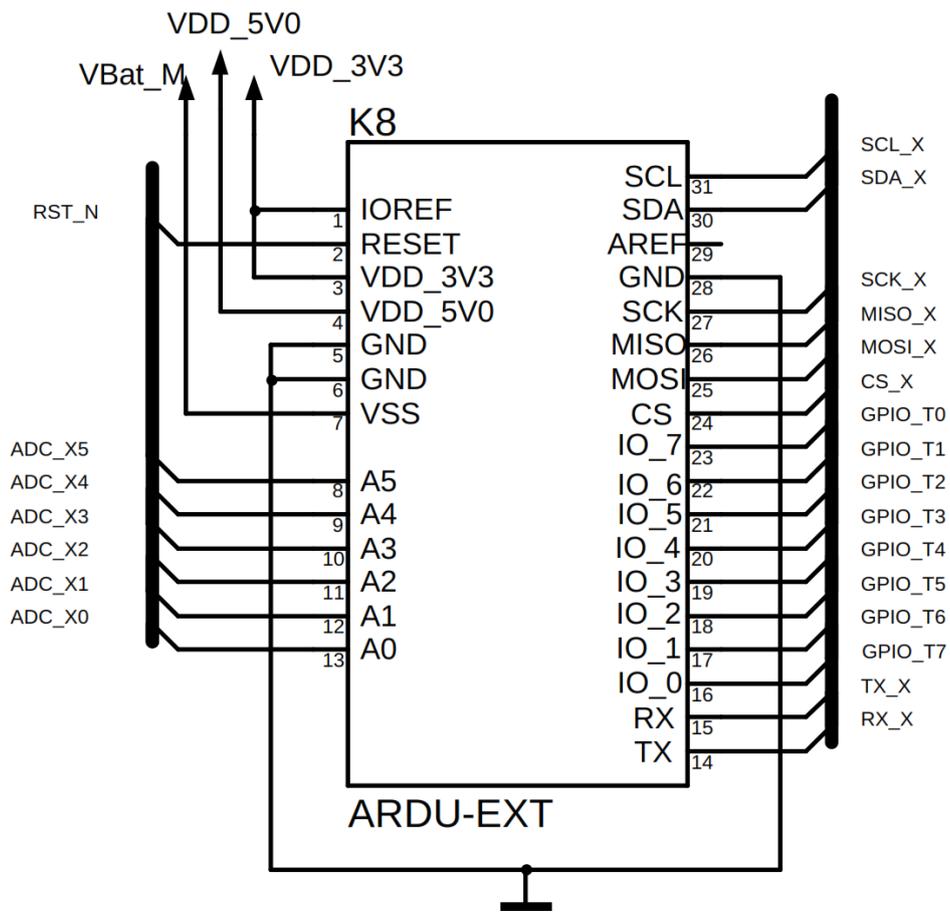


Abbildung 7.1 Marvin Arduino Stecker - Schaltplan

7.2 Pin Mapping TM4C129EKCPDT

Tabelle 7.2 TM4C129EKCPDT Pin Mapping

Pin	Name	Function	Signal	Pin	Name	Function	Signal
1	PD0	SSI2DAT1	MISO_SD	33	PA0	U0RX	RX_X
2	PD1	SSI2DAT0	MOSI_SD	34	PA1	U0TX	TX_X
3	PD2	SSI2FSS	CS_SD	35	PA2	SSI0CLK	SCK_X
4	PD3	SSI2CLK	SCK_SD	36	PA3	SSI0FSS	CS_X
5	PQ0	SSI3CLK	WIFI_SPI_CLK	37	PA4	SSI0DAT0	MOSI_X
6	PQ1	SSI3FSS	WIFI_SPI_CS	38	PA5	SSI0DAT1	MISO_X
7	VDD			39	VDD		
8	VDDA			40	PA6	GPIO	REAR_R
9	VREFA+			41	PA7	GPIO	REAR_L
10	GNDA			42	PF0	M0PWM0	Motor_L - AINT1
11	PQ2	SSI3DAT0	WIFI_SPI_MOSI	43	PF1	M0PWM1	Motor_L - AINT2
12	PE3	AIN0	ADC_X3	44	PF2	M0PWM2	Motor_R - BIN1
13	PE2	AIN1	ADC_X4	45	PF3	M0PWM3	Motor_R - BIN2
14	PE1	AIN2	ADC_X2	46	PF4	M0FAULT0	nFault
15	PE0	AIN3	ADC_X5	47	VDD		
16	VDD			48	GND		
17	GND			49	PG0	I2C1SCL	SCL_PRX
18	PK0	GPIO	SD_Present	50	PG1	I2C1SDA	SDA_PRX
19	PK1	GPIO	INT_2_A/G	51	VDD		
20	PK2	GPIO	INT_1_A/G	52	VDD		
21	PK3	GPIO	NC	53	EN0RXIN		NC
22	PC7	GPIO	INT_M	54	EN0RXIP		NC
23	PC6	GPIO	DRDY_M	55	GND		
24	PC5	GPIO	WIFI_RST	56	EN0TXIN		NC
25	PC4	GPIO	WIFI_HOST_INT	57	EN0TXIP		NC
26	VDD			58	GND		
27	PQ3	SSI3DAT1	WIFI_SPI_MISO	59	RBIAS		
28	VDD			60	PK7	I2C4SDA	SDA_9D
29	PH0	GPIO	PROX_RL_CS	61	PK6	I2C4SCL	SCL_9D
30	PH1	GPIO	PROX_RM_CS	62	PK5	GPIO	SWITCH
31	PH2	GPIO	PROX_RR_CS	63	PK4	GPIO	CHARGE
32	PH3	GPIO	NC	64	WAKE_N		

Pin	Name	Function	Signal
65	HIB_N		
66	XOSC0		
67	XOSC1		
68	VBAT		
69	VDD		
70	RST_N		
71	PM7	TSCCP1	GPIO_T7
72	PM6	TSCCP0	GPIO_T6
73	PM5	GPIO	GPIO_T5
74	PM4	GPIO	GPIO_T4
75	PM3	T3CCP1	GPIO_T3
76	PM2	T3CCP0	GPIO_T2
77	PM1	T2CCP1	GPIO_T1
78	PM0	T2CCP0	GPIO_T0
79	VDD		
80	GND		
81	PL0	I2C2SDA	SDA_X
82	PL1	I2C2SCL	SCL_X
83	PL2	GPIO	LED3_B
84	PL3	GPIO	LED2_G
85	PL4	GPIO	LED1_R
86	PL5	GPIO	WIFI_LED_GREEN
87	VDDC		
88	OSC0		
89	OSC1		
90	VDD		
91	PB2	GPIO	HEAD_R
92	PB3	GPIO	HEAD_L
93	PL7	GPIO	WIFI_LED_YELLOW
94	PL6	GPIO	WIFI_LED_RED
95	PB0	GPIO	ODO_L
96	PB1	GPIO	ODO_R

Pin	Name	Function	Signal
97	TDO	TDO	TDO
98	TDI	TDI	TDI
99	TMS	TMD	TMS
100	TCK	TCK	TCK
101	VDD		
102	PQ4	GPIO	nHIB
103	PP2	GPIO	PROX_FR_CS
104	PP3	GPIO	PROX_FM_CS
105	PP4	GPIO	PROX_FL_CS
106	PP5	GPIO	NC
107	PN0	GPIO	INT_FL
108	PN1	GPIO	INT_FM
109	PN2	GPIO	INT_FR
110	PN3	GPIO	INT_RL
111	PN4	GPIO	INT_RM
112	PN5	GPIO	INT_RR
113	VDD		
114	GND		
115	VDDC		
116	PJ0	U3RX	USB_TX
117	PJ1	U3TX	USB_RX
118	PP0	GPIO	NC
119	PP1	GPIO	NC
120	PB5	GPIO	NC
121	PB4	AIN10	ADC_BAT
122	VDD		
123	PE4	AIN9	ADC_ML
124	PE5	AIN8	ADC_MR
125	PD4	AIN7	NC
126	PD5	AIN6	ADC_BBAT
127	PD6	AIN5	ADC_X0
128	WD7	AIN4	ADC_X1

7.3 PCB Print

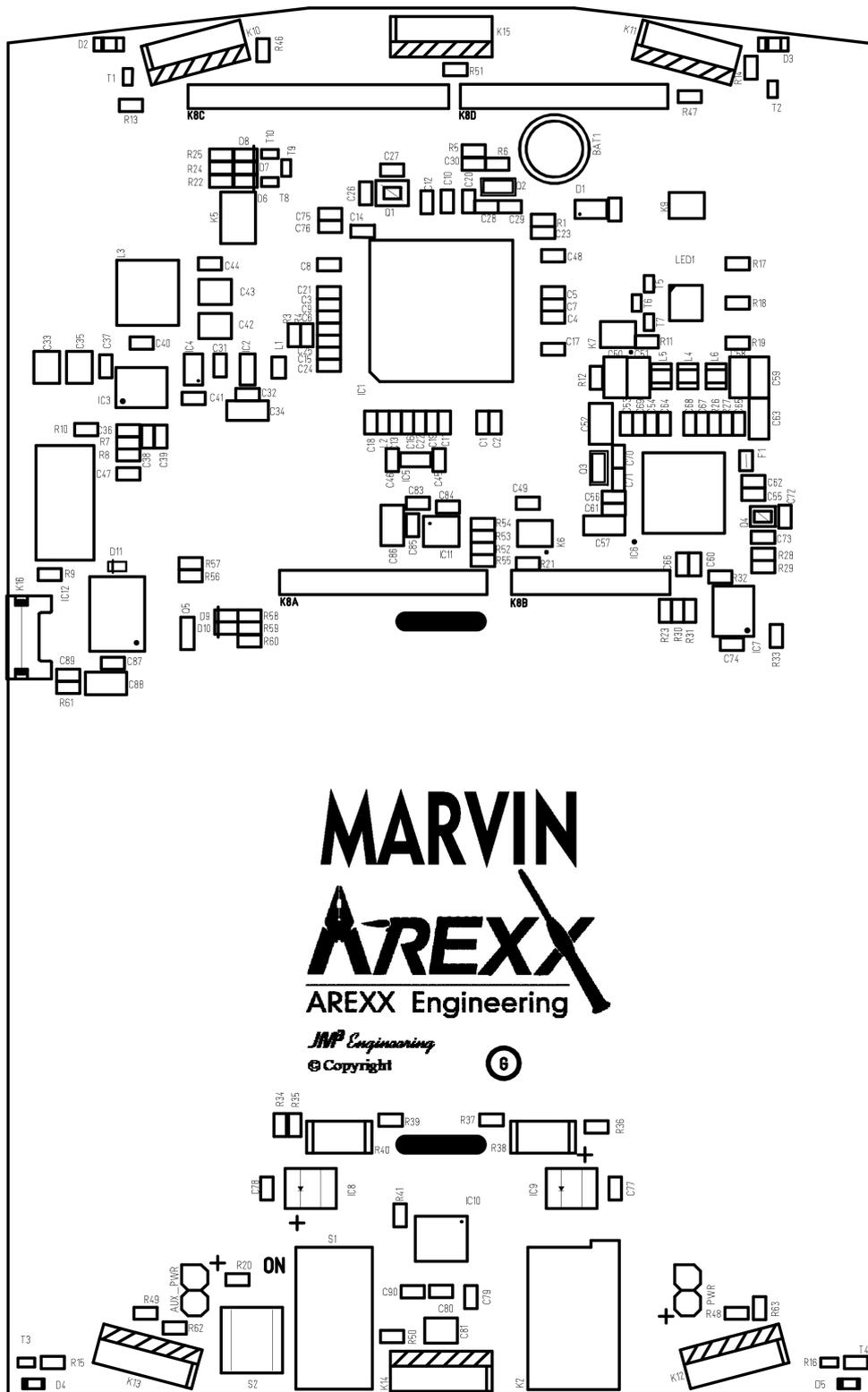


Abbildung 7.2 Marvin PCB Print

Abbildungsverzeichnis

0.1	Marvin Roboter	1
0.2	Welcome-Screen	1
0.3	Program-Screen	1
0.4	Remote Control und Status Display	1
2.1	Marvin Roboter Hardware	4
2.2	Field of View der vorderen Proxi Sensoren	6
2.3	Marvin mit hinteren Proximity-Sensoren	7
2.4	Marvin mit eingesetzter SD Karte	8
2.5	Marvin WebIDE Willkommensbildschirm	11
2.6	Kommandointerface	13
2.7	Himmelsrichtungen und Inklinationsvektor	14
2.8	Abstandssensor Hilfsprogramm Bildschirm	16
3.1	Marvin WebIDE Menüleiste	19
3.2	Marvin WebIDE Programmbuttons	20
3.3	Der Marvin WebIDE Helpbrowser	21
3.4	Nicht passende Blocktypen in iRP	22
3.5	Beispiel für einen Compiler Fehler	23
3.6	Beispiel für generierten C++ Code	23
4.1	Das JM3 Robot-Tool	26
4.2	Das JM3 Robot-Tool - Terminal Fenster	27
5.1	Marvin Funktionstasten Im Fernsteuer und Display Tab	30
7.1	Marvin Arduino Stecker - Schaltplan	33
7.2	Marvin PCB Print	36

Tabellenverzeichnis

2.1	CLI Kommandos	18
7.1	Pinout des Arduino kompatiblen Steckers	32
7.2	TM4C129EKCPDT Pin Mapping	34