

Open Robotic Board

Firmware

Specification

Thomas Breuer

Hochschule Bonn-Rhein-Sieg

Datum: 12.11.2023

1 Anwendungszweck

Diese Dokument beschreibt die Firmware für das OpenRoboticBoard (ORB).

Das ORB ist ein Mikrocontroller-Modul zur Steuerung von Modell-Robotern. Das Modul verfügt über folgende Anschlüsse:

- Spannungsversorgung (DC, $U_{\max} = 9V$, $I_{\max} = 5A$)
- M1 bis M4: Motor-Ports für Gleichstrommotoren ($I_{\max} = 1A$), optional mit Encoder
- S1 bis S4: Sensor-Ports mit je 2 analogen und 2 digitalen Kanälen
- D1 und D2: Digitale Input-Ports
- Servo 1 und Servo 2: Ports für Modellbau-Servos (5V/500mA)

Als User-Interface stehen 2 Tasten sowie 3 LEDs zur Verfügung. Für zukünftige Erweiterungen sind ein I2C-Port vorgesehen.

Zur Kommunikation sind ein USB-Port (Micro-B) sowie ein Bluetooth-Modul vorhanden.

Die Firmware lässt sich über USB flashen. Zu Debug-Zwecken ist der SWD-Port des Mikrocontrollers zugänglich.

2 ORB-Firmware

Die Firmware des ORB übernimmt folgende Aufgaben:

- Regelung der Motoren an den Ports M1 bis M4. Die Regelung kann in verschiedenen Modi betrieben werden, beispielsweise Drehzahlregelung oder Positionierung.
- Steuerung von Modellbau-Servos an den Ports Servo1 und Servo2.
- Konfiguration und Erfassen der Sensoren an den Ports S1 bis S4. Es können digitale, analoge, I2C oder UART-Sensoren angeschlossen werden.
- Erfassung der digitalen Eingänge D1 und D2.
- Kommunikation mit einem externen Rechnersystem über USB oder Bluetooth
- Download einer Anwendungssoftware, die dann als Thread gestartet werden kann (lokale Steuerung). Die Firmware stellt der Anwendungssoftware die Zugriffsfunktionen als API bereit.

2.1 Aktorik

2.1.1 Motor

Die Firmware kann die Motoren an den Ports M1 bis M4 in 4 Betriebsarten betreiben:

- Einstellen einer variablen Spannung im Bereich +/-Versorgungsspannung
- Kurzschlussbremse: Durch Kurzschluss der Motoranschlüsse werden diese passiv gebremst
- Drehzahlregelung mit PI-Regler, nur für Encoder-Motoren. Die Regelparameter und Encoderauflösung lassen sich per Software einstellen, s.d. unterschiedliche Motortypen verwendet werden können.
- Positionierung. Der Drehzahlregelung wird eine Positionsregelung überlagert, s.d. eine Zielposition angefahren werden kann. Die Drehzahl wird während des Anfahrens und Abstoppens rampenförmig angepasst und auf eine Maximaldrehzahl begrenzt. Rampensteilheit (Beschleunigung) und Maximaldrehzahl sind konfigurierbar.

Die Drehzahlregelung und Positionierung ermöglichen einen synchronen Betrieb mehrerer Motoren, intern findet jedoch kein Abgleich unter den Motoren statt.

2.1.2 Servo

Die Ports Servo1 und Servo2 sind für Modellbau-Servos geeignet. Die Firmware erzeugt ein PWM-Steuersignal mit einer variablen Pulsbreite von 1-2ms und einer Periodendauer von 16 ms.

Weiterhin kann eingestellt werden, wie schnell die Pulsbreite vom bisherigen Wert an einen neuen Einstellwert angepasst wird. Das Steuersignal kann auch ausgeschaltet (=Ground-Pegel) werden.

2.2 Sensorik

2.2.1 Digital

Der Digitalwert an den Pins 5 und 6 der Sensor-Ports S1 bis S4 kann unabhängig vom verwendeten Sensortyp jederzeit eingelesen werden.

2.2.2 Analog

Der Analogwert an den Pins 1 und 2 der Sensor-Ports S1 bis S4 kann jederzeit unabhängig vom verwendeten Sensortyp jederzeit eingelesen werden

2.2.3 I2C

Die Firmware unterstützt den I2C-Bus an den Pins 5 und 6 der Sensor-Ports. Softwareseitig wird u.a. das Protokoll der NXT-Ultraschallsensoren unterstützt. *Weitere Sensortypen sind z.Zt. in Planung.*

2.2.4 UART

Die Pins 5 und 6 können als UART-Schnittstelle betrieben werden. Die Firmware unterstützt das Protokoll der EV3-Sensoren.

2.2.5 Digitaleingänge

Die zusätzlichen Digitaleingänge D1 und D2 können erfasst werden.

2.3 Kommunikation

Zur Kommunikation mit der Firmware kann wahlweise USB oder Bluetooth verwendet werden. Die beiden Kommunikationsschnittstellen unterstützen:

- Download der Anwendungssoftware zur lokalen Steuerung
- Konfiguration der Motoren und Sensoren
- Steuerung der Motoren und Modellbau-Servos
- Abfrage der Motor- und Sensordaten
- Abfrage der Anzeigetexte
- Übertragung Tastencodes
- Systemkonfiguration

2.4 Betriebsarten

Die Firmware ermöglicht den Betrieb des ORB in verschiedenen Modi:

2.4.1 Monitoring

In dieser Betriebsart können Daten vom ORB erfasst, Systemeinstellungen vorgenommen und die Anwendungssoftware heruntergeladen werden.

2.4.2 Lokale Steuerung

Die lokale Steuerung ermöglicht einen Standalone-Betrieb des Roboters, der so ohne weitere externe Rechensysteme agieren kann.

Die Anwendungssoftware zur lokalen Steuerung kann mit den Tasten "Btn1" oder "Btn2" gestartet werden. Je nach Taste, wird der Anwendungssoftware ein unterschiedlicher Startparameter übergeben. Falls die lokale Steuerung aktiv ist, wird dies durch die LED1 bzw. LED2 angezeigt.

Taste	Startparameter	Anzeige
Btn1	0	LED1 (grün)
Btn2	1	LED2 (orange)

Durch einen erneuten Tastendruck auf "Btn1" oder "Btn2" wird die Anwendungssoftware abgebrochen.

Die Anwendungssoftware hat vollen Zugriff auf die Konfiguration und Messdaten der Motoren und Sensoren. Weiterhin kann die Anwendungssoftware Kalibrationsdaten persistent speichern.

Die Anwendungssoftware kann Textausgaben erzeugen und Tastaturcodes einlesen, sofern ein entsprechender Monitor angeschlossen ist. Der ORB-Monitor stellt damit das User-Interface der Anwendung dar.

2.4.3 Remote-Steuerung

In dieser Betriebsart wird der ORB durch ein externes Rechnersystem (Smartphone, Tablet, PC) gesteuert. Dazu lassen sich Aktoren und Sensoren über die Kommunikationsschnittstelle konfigurieren und die Motor- bzw. Sensordaten abfragen.

2.5 Systemkonfiguration

2.5.1 Batterieüberwachung

Mit der LED3 (rot) wird der Zustand der Spannungsversorgung angezeigt:

Spannung V_{CC}	Anzeige LED3 (rot)
Spannung OK $V_{OK} \leq V_{CC}$	blinkt mit kurzen Unterbrechungen ---- _ ---- _
Spannung kritisch $V_{low} < V_{CC} < V_{OK}$	blinkt gleichmäßig --- _ _ _ _
Spannung zu klein $V_{CC} \leq V_{low}$	blinkt schnell _ _ _ _ _

Die Spannungsschwellen lassen sich einstellen, siehe Kapitel Kommunikationsschnittstelle. Die Anzeige ist während eines Downloads der Anwendungssoftware oder einem Update der Firmware nicht gültig.

2.5.2 ORB-Name

Es kann ein Name für ein ORB vergeben werden, der dauerhaft im Flash gespeichert wird. Der Name kann abgefragt werden.

2.5.3 Versionsnummer

Die Versionsnummer der Firmware sowie die Hardwareversion des Boards können abgefragt werden, siehe Kapitel Kommunikationsschnittstelle.

Die Hardwareversion ist diejenige Version, von der die Firmware ausgeht. Daher Vorsicht bei der Installation der Firmware, da diese keine Möglichkeit hat, die tatsächliche Hardwareversion zu prüfen

3 Application Programming Interface (ORB-API)

Die ORB-API stellt der Anwendungssoftware eine Klasse zum Zugriff auf den ORB bereit.

Dokumentation der API siehe: `Doc/ORB-API.html`