

# Z.E.D. – Die Autonom Fahrende Schubkarre

Johann Gräfenhain, Titus Florin

Semesterprojekt • Studiengang Informatik B.Sc. • Fachbereich Informatik und Medien • 15.01.2026

## Aufgabenstellung

Das Ziel des Projekts war es, einen autonomen, intelligenten Roboter mit realem Anwendungszweck zu entwickeln. In diesem Rahmen haben wir einen Transportroboter entwickelt (Abb. 1), welcher einen durch künstliche Landmarken bestimmten Weg autonom abfährt. Dafür haben wir drei Zwischenziele definiert:

- das Halten des Kurses zwischen den Landmarken
- das Bestimmen der relativen Position zur Landmarke
- das Planen und Durchführen eines Manövers zur Korrektur des Kurses nach Überqueren einer Landmarke

Das praktische Einsatzgebiet des Roboters ist vielseitig. Er könnte z.B. in der Gartenarbeit eingesetzt werden. In größerem Maßstab ist auch der Einsatz auf Baustellen denkbar.

## Technische Umsetzung

Der Roboter hat eine Ackermann-Hinterrad-Lenkung und vorne eine Antriebsachse mit Differential. Dieser Aufbau schränkt ihn zwar in der Bewegung ein (es ist keine Rotation auf der Stelle möglich), erlaubt es jedoch höhere Lasten zu transportieren.

Insgesamt wurden vier Motoren im Roboter verbaut (2x Antriebsmotoren, 1x Servolenkung, 1x Kippmotor). Für die Sensorik haben wir uns auf ein Minimum beschränkt (2x Optokoppler, 1x ICM-20948 9-DoF IMU). Die zwei optischen Sensoren sind voreinander an der Front des Fahrzeuges montiert und dienen zur Erkennung der Landmarken, sowie der Bestimmung der Geschwindigkeit beim Überqueren der Landmarken. Die IMU dient zur Lagebestimmung und wird in Zusammenhang mit einem PID-Regler verwendet, um den Kurs zwischen den Landmarken zu halten. Dies führte zu einer hervorragenden Verbesserung beim geradeaus Fahren, selbst bei einer reinen P-Regler-Konfiguration.

Zur Steuerung wurde ein ESP32-Dualcore-Mikrokontroller (ESP32 NodeMCU Dev Kit C v2) eingesetzt (Abb. 3), wobei ein Kern für das Auslesen und Verarbeiten der Sensoren mit einer konstanten Frequenz von 200 Hz zuständig ist und der andere für das Planen und Durchführen der Aktionen.

## Landmarken

Für die Landmarken haben wir das bekannte Konzept der Z-Shaped Landmarken [1, pp. 179-180], welche für eine ähnliche Anwendung entwickelt wurden, erweitert. Diese können verwendet werden, um mit minimaler Sensorik die relative x-Abweichung beim Überqueren zu bestimmen. Wir haben sie um eine doppelte Diagonale erweitert (Abb. 2), um zusätzlich den relativen Winkel bestimmen zu können.

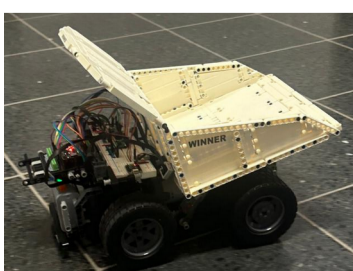


Abb. 1: Z.E.D.

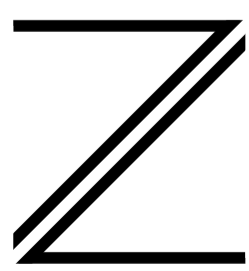


Abb. 2: Landmarke

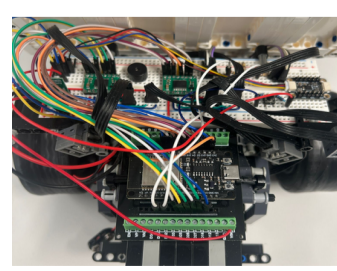


Abb. 3: Elektronik

## Manöver

Der Weg des Roboters wird dynamisch anhand der überquerten Landmarken bestimmt. Nach dem Überqueren einer Landmarke wird ein Korrektur-Manöver durchgeführt, welches den Roboter wieder zentriert auf die Gerade senkrecht zur letzten Linie der Landmarke bewegt (siehe Abb. 4). Durch das gezielte Rotieren einer Landmarke, können Kurven des Wegs beschrieben werden. Dabei ist jedoch zu beachten, dass die Korrektur des Kurses nur innerhalb einer Abweichung von weniger als  $\pm 45^\circ$  möglich ist.

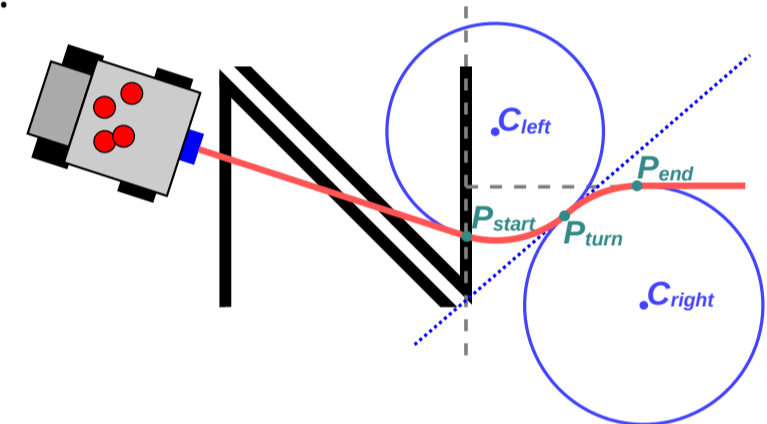


Abb. 4: Beispiel eines Manövers

Da der LEGO-Servo zur Lenkung nur in diskreten Schritten arbeitet, haben wir ein Manöver entwickelt, welches auf zwei aufeinander folgenden Kreisbögen basiert (Abb. 4). Die Radien der beiden Kreisbögen können variieren, müssen aber bekannt sein. Bei einer entsprechenden Mindestgröße der Radien kann auf diese Art jede mögliche Abweichung beim Überqueren der Landmarken korrigiert werden. Je nach Abweichung variiert dabei die Länge des benötigten Wegs, weshalb die nächste Landmarke mit ausreichendem Abstand platziert werden muss.

## Grenzen des Designs

Ein mögliches Problem, welches mit diesem Design für praktische Anwendungen auftreten kann, ist, dass der markierte Weg nicht bidirektional verwendet werden kann. Um den Roboter also wieder an seinen Ursprung fahren zu lassen, muss ein „Kreis“ mit den Landmarken erzeugt werden. Ein weiteres Problem ist, dass sich die Grenzen der maximalen x-Abweichung und der Winkelabweichung gegenseitig einschränken. Je größer die x-Abweichung ist, desto kleiner darf die maximale Winkelabweichung sein und umgekehrt.

## Ergebnis

Im Rahmen dieses Projekts konnten wir einen funktionierenden Prototypen erstellen, welcher die Landmarken in etwa 95% der Fälle korrekt erkennt und ein Korrekturmanöver plant. Bei der Durchführung des Manövers kommt es jedoch noch in ca. zwei Drittel der Versuche zu Fehlern.

## Quellen

[1] J. Borenstein, H. R. Everett, and L. Feng, „Where am I? Sensors and Methods for Mobile Robot Positioning“, April 1996.