

Elektronische Kunst: Oszilloskop-Plotter in Aktion

Yeva Makarova, Elektrotechnik und Informationstechnik
Otto-von-Guericke-Universität Magdeburg

Zusammenfassung—Die folgende Arbeit beschreibt den Entwicklungsprozess eines Oszilloskop-Plotters. Der Prozess wurde so automatisiert, dass der Roboter mit Hilfe von zwei in MATLAB programmierten Motoren jede im x-y-Koordinatensystem darstellbare Figur auf dem Bildschirm des Oszilloskops zeichnen kann.

Schlagwörter—Kunst, LEGO Mindstorms, MATLAB, Oszilloskop-Plotter, Projektseminar.

I. EINLEITUNG

O SZILLOSKOPE werden zum Messen, Speichern und Aufzeichnen von elektrischen Signaldaten verwendet. Normalerweise werden sie in Laboratorien sowie in der Forschung und Entwicklung zur Überwachung oder Messung und Untersuchung elektrischer Signale eingesetzt. Aber wer sagt, dass sie nicht auch in der Kunst eingesetzt werden können?

In einem speziellen Zweikanal-Oszilloskopmodus, dem so genannten x-y-Modus, wird das Signal des einen Kanals zur Ablenkung des Strahls entlang der horizontalen x-Achse und das Signal des anderen Kanals zur Ablenkung des Strahls entlang der vertikalen y-Achse verwendet.

In diesem Artikel wird beschrieben, wie man mit Hilfe von zwei Potentiometern ein bestimmtes Bild auf dem Bildschirm des Oszilloskops darstellen kann. Das Prinzip des Zeichnens ist das gleiche wie bei dem bekannten Kinderspiel "Etch A Sketch": Durch Bewegen des linken Potentiometers wird eine horizontale Linie gezeichnet, durch Bewegen des rechten Potentiometers eine vertikale Linie. Durch Einstellen der Geschwindigkeit und der Drehrichtung der Potentiometer kann jede beliebige Form gezeichnet werden.

II. VORBETRACHTUNGEN

A. Idee und Beweggrund

Am Anfang stand die Idee, einen Roboter zu entwickeln, der auf dem Bildschirm eines Oszilloskops beliebige Figuren zeichnen kann, insbesondere das Nikolaushaus. Da das Oszilloskop zum Zeichnen im x-y-Modus mit zwei Potentiometern verwendet werden kann, können diese Potentiometer auch mit Motoren verbunden werden, die auf bestimmte Weise programmiert werden können. Dadurch sollte das Bild auf dem Oszilloskop von wesentlich besserer Qualität sein, als wenn es von Hand gezeichnet wird. Wenn es genug Zeit übrig wäre, könnte auch versucht werden, andere Formen zu zeichnen, z. B. einen Kreis.

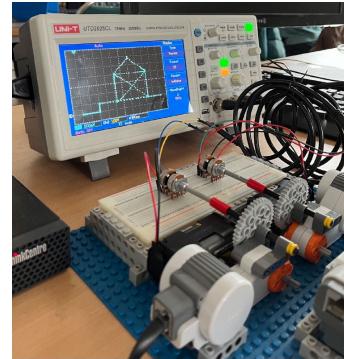


Abbildung 1. Komplette Roboterkonstruktion in Verbindung mit dem Oszilloskop

B. Bestandteile

Die Konstruktion des Roboters besteht aus folgenden Elementen (siehe Abbildung 1):

- 1) LEGO-Bauplatte
- 2) LEGO-Motor 2x
- 3) 8-er Zahnrad 2x
- 4) 40-er Zahnrad 2x
- 5) Potentiometer 2x
- 6) Batterie 2x
- 7) Breadboard
- 8) Drähte
- 9) LEGO-NXT-Stein
- 10) Oszilloskop
- 11) Andere LEGO-Bauteile

C. Code-Aufgabe

Ursprünglich sollte der Code die folgende Abfolge von Aktionen ausführen:

- 1) Kalibrierung der Motoren
- 2) Grenzen nach innen verschieben
- 3) Darstellung der Figur am Computer
- 4) Starten der Motoren und Zeichnen der Figur auf dem Bildschirm des Oszilloskops

III. TECHNISCHE UMSETZUNG

In diesem Abschnitt wird die technische Realisierung des Roboters beschrieben, die aus zwei Hauptteilen besteht: der Konstruktion und dem Programmcode. Ein vereinfachtes Programmflussdiagramm ist ebenfalls in Abbildung 4 dargestellt.

A. Aufbau

Die Konstruktion des Oszilloskop-Plotters besteht aus zwei Motoren, die jeweils über zwei Zahnräder mit einem der Potentiometer verbunden sind. Diese Zahnräder regeln gleichzeitig die Drehgeschwindigkeit der Potentiometer (siehe Abbildung 2). Das erste Zahnräder ist direkt mit dem Motor verbunden und hat daher die gleiche Geschwindigkeit. Das zweite Zahnräder befindet sich neben dem ersten Zahnräder und ist mit einem Potentiometer verbunden. Das Übersetzungsverhältnis zwischen den beiden Zahnrädern beträgt 5:1, d. h. das zweite Zahnräder dreht sich fünfmal langsamer. Das bedeutet, dass sich das Potentiometer langsamer dreht und somit besser gesteuert werden kann. Alle Zahnräder sind fest mit dem Motor verbunden.

Die Batterien mit einer Spannung von jeweils 6 V werden mit Hilfe eines Breadboards parallel zu den Potentiometern angeschlossen. Bei der Installation der Batterien ist darauf zu achten, dass keine Kurzschlüsse entstehen.

Die Drähte, die den Aufbau mit dem Oszilloskop verbinden, werden in ähnlicher Weise angeschlossen, jedoch so, dass der Widerstand des Potentiometers halbiert wird (siehe Abbildung 3). Das erste Potentiometer (Abb. 1 links) ist für die x-Achse, das zweite Potentiometer (Abb. 1 rechts) ist für die y-Achse zuständig. Beim Verstellen der Potentiometer ändert sich die Spannung. Das Oszilloskop misst die Spannung und zeigt sie als Diagramm auf dem Bildschirm im x-y-Modus an.

Die gesamte Konstruktion, einschließlich des LEGO-NXT-Steins und des Breadboards, wird auf der LEGO-Bauplatte befestigt.

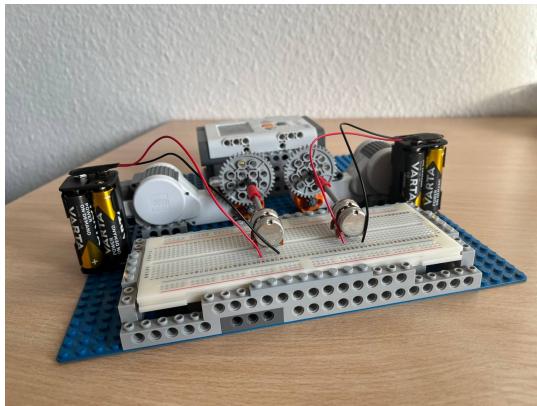


Abbildung 2. Konstruktion des Roboters, Ansicht von hinten

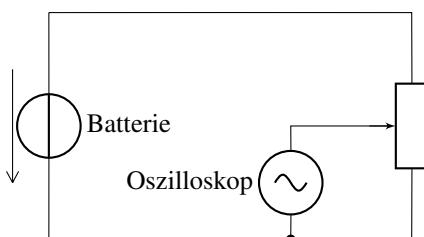


Abbildung 3. Elektrische Auslegung des Roboters

B. Programm

Zuerst muss das Programm die Anfangs- und Endpunkte kalibrieren, auf die die Potentiometer maximal eingestellt werden können. Dazu müssen die Motoren so programmiert werden, dass sie zunächst in eine Richtung bis zum Anschlag fahren und dann anhalten, ohne dass die Potentiometer aus dem Breadboard herauspringen. Dann müssen die Motoren ihre Position bestimmen und die Daten über den LEGO-NXT-Stein an den Computer senden und das Gleiche in die andere Richtung wiederholen.

Da die ermittelten Start- und Endpositionen der Motoren bei jedem Programmstart andere Koordinaten haben und eine vollständige Kalibrierung bei jedem Programmstart sehr zeitaufwendig ist, bestand die Notwendigkeit, diese Positionen schnell und allgemein zu ermitteln. Nach mehreren Kalibrierungsversuchen wurde festgestellt, dass der maximale Drehmomentbereich von Motor A bei etwa 1500 Drehmomenteinheiten und der von Motor B bei 1460 Drehmomenteinheiten liegt. Mit diesen Daten kalibriert der Code die Endpunkte wie folgt: Beide Motoren bewegen die Potentiometer nach rechts bis zum Anschlag und stoppen dann; danach werden die Endpunkte berechnet und im Code gespeichert.

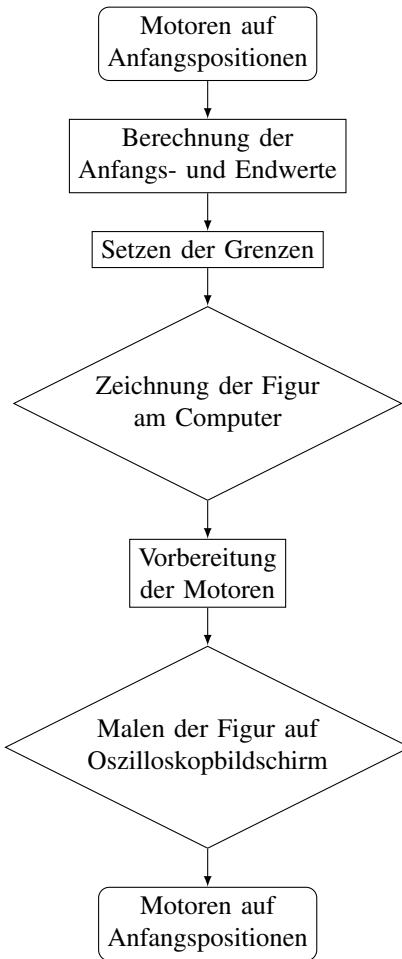


Abbildung 4. Vereinfachtes Flussdiagramm des Programms

Im nächsten Schritt wurde versucht, die erste Figur zu zeichnen, was jedoch nicht gelang. Es stellte sich heraus,

dass es Bereiche auf der Koordinatenebene des Oszilloskops gab, in denen die Figur nicht linear gezeichnet wurde. Wurde z. B. das Potentiometer in einem ausreichend großen Winkel gedreht, so wurde zunächst eine sehr kurze gerade Linie auf dem Oszilloskop gezeichnet. Dann, nach einer kleinen Drehung desselben Potentiometers, wurde eine um ein Vielfaches längere Linie gezeichnet, und dann, nach einer langen Drehung, wieder eine kurze Linie. Nach Untersuchung des Verhaltens der Oszilloskopkurve wurde ein linearer Bereich ermittelt, in dem das Bildverhalten vorhersagbar ist. Daher wird bei der Ausführung des Codes nach der Kalibrierung der Potentiometer der Arbeitsbereich des Koordinatensystems berechnet und seine Grenzen festgelegt. Anschließend wird der Lichtpunkt, mit dem die Figur auf dem Oszilloskop gezeichnet wird, an den Anfang des zulässigen linearen Bereichs verschoben.

Der nächste Schritt besteht darin, eine beliebige Form auszuwählen und sie im Koordinatensystem der x-Achse mit Maximalwerten von 6 in y-Richtung und 10 in x-Richtung zu definieren. Die Form wird dann in das Koordinatensystem des Oszilloskop-Plotters umgewandelt. MATLAB zeichnet dann die definierte Form zur Überprüfung auf den Computer und startet die Motoren.

Da die Motoren spiegelbildlich zueinander montiert wurden, musste der Softwarecode in gewisser Weise angepasst werden. Das heißt, um eine gerade Diagonale z. B. in einem Winkel von 45 Grad zu zeichnen, mussten die Motoren die gleiche Geschwindigkeit haben, aber in entgegengesetzter Richtung. Es gibt noch ein weiteres Problem: Um Diagonalen zu erzeugen, die nicht im Winkel von 45 Grad verlaufen, müssen die Motoren unterschiedliche Geschwindigkeiten haben, die voneinander und von der Position abhängen, von der sie kommen und zu der sie gehen sollen (siehe die Lösung im Anhang).

Sobald die gewünschte Figur auf dem Bildschirm des Oszilloskops angezeigt wird, setzt das Programm die Motoren auf ihre Startpunkte und ist bereit für den nächsten Lauf.

IV. ERGEBNISDISKUSSION

Das Ergebnis ist, dass nicht nur die Figur des Nikolaushauses (Abb. 5), wie ursprünglich geplant, sondern auch ein Stern (Abb. 6a) und ein Kreis (Abb. 6b) dargestellt werden können, was ein unbestreitbarer Erfolg ist. Dies zeigt, dass der Oszilloskop-Plotter in der Lage ist, jede im x-y-Koordinatensystem angegebene Form darzustellen.

Es gibt aber auch Probleme, die noch nicht vollständig gelöst sind. An einigen Stellen drehen sich die Motoren nicht vollständig, was dazu führen kann, dass die Figur nicht zusammenhängend ist. Und obwohl wir dieses Problem zunächst lösen konnten, indem wir die Motoren so programmierten, dass sie eine bestimmte Strecke über die eingestellte Position hinaus drehen, hat sich herausgestellt, dass sich die erforderliche Strecke bei jedem Einschalten des LEGO-NXT-Steins ändert.

V. ZUSAMMENFASSUNG UND FAZIT

In zwei Wochen wurde viel Arbeit geleistet und die meisten Fehler wurden behoben. Der Oszilloskop-Plotter funktioniert recht gut und ist bereits in der Lage, einige Formen recht schön und flüssig zu zeichnen. Mit mehr Zeit wäre es möglich

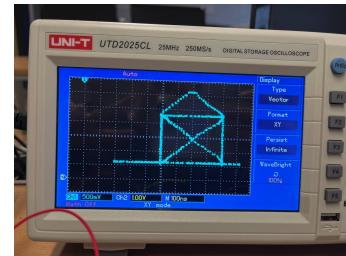


Abbildung 5. Haus des Nikolaus

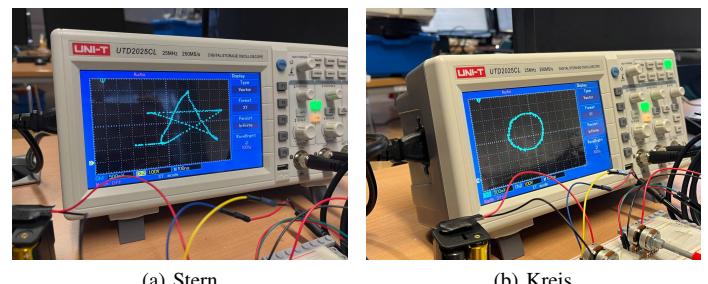


Abbildung 6. Andere Ergebnisse

gewesen, einige der Zeichnungen zu verbessern, indem der oben beschriebene Fehler behoben worden wäre.

ANHANG

Einstellung der Motordrehzahl in Abhängigkeit von der Anfangs- und Endposition für das Zeichnen von Diagonalen:

```
powerA=max([5,min([100,round(156.5902/149.5236*abs(tacho_limitA/tacho_limitB)*power)])]);
powerB=max([5,min([100,round(149.5236/156.5902*abs(tacho_limitB/tacho_limitA)*power)])]);

disp(['Power A: ',num2str(powerA)])
disp(['Power B: ',num2str(powerB)])

if powerA>powerB
    powerA=power;
else
    powerB=power;
end
```

LITERATURVERZEICHNIS

- [1] Wikipedia: Oszilloskop, <https://de.wikipedia.org/wiki/Oszilloskop>
Version: Januar 2024
- [2] Wikipedia: Etch A Sketch, https://en.wikipedia.org/wiki/Etch_A_Sketch
Version: Januar 2024