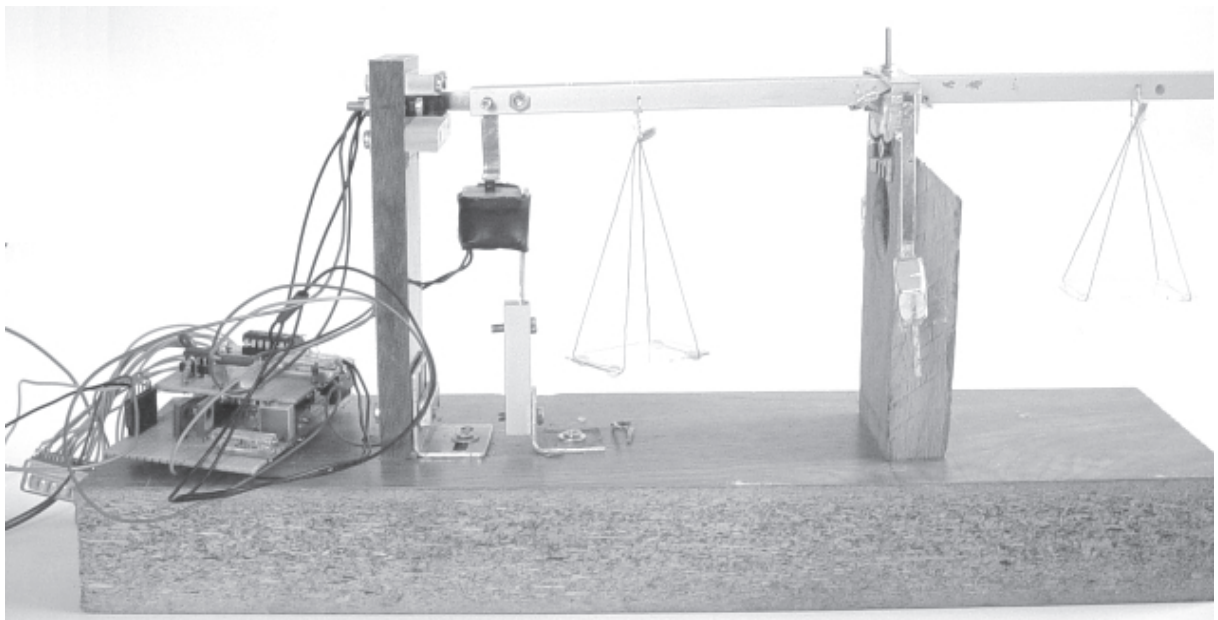


Computergesteuerte Präzisionswaage mit 12-Bit D-A- Wandler



Wettbewerb "Jugend Forscht" 2003

Sebastian Hellberg (14 Jahre)

**Arbeitsgemeinschaft "Jugend Forscht"
des Christian-Gymnasiums Hermannsburg
Leitung: StD Thomas Biedermann**

Inhaltsverzeichnis

1.	Einleitung	3
2.	Vom Gewicht zum digitalisierten Wert	3
2.1	Herkömmliche Waagen	3
2.2	Prinzip meiner Waage	4
2.2.1	Erzeugen der Gegenkraft	4
2.2.2	Erkennen des Gleichgewichtszustandes	4
2.2.3	Digitalisierung per Computer	4
3.	Praktische Realisierung	5
3.1	Aufbau der Mechanik	5
3.2	Schaltungsaufbau	6
3.2.1	DA-Wandler HDG-0807	6
3.2.2	DA-Wandler TI-5680V	6
3.2.3	Leistungsverstärker	7
3.2.4	Lichtschrakenabfrage	7
3.2.5	Computeranschluss	8
3.3	Computer-Programm	8
3.4	Funktionstest	9
4.	Messergebnisse	10
4.1	Nullpunktstabilität	10
4.2	Erster Test mit Kunststoffröhrchen	10
4.3	Überprüfung mit Apotheker-Gewichten	11
4.4	Überprüfung mit Reißzwecken	12
5.	Folgerungen aus den Ergebnissen	13
5.1	Verbesserung der Linearität	13
5.2	Absolute Genauigkeit	13
5.3	Weitere Verbesserungsmöglichkeiten	14
6.	Literaturverzeichnis	14
7.	Danksagung	14
Anhang:		
A1	Programmlisting (Hauptprogramm)	15
A2	Programmlisting (Bit-Test-Programm)	17
A3	Schaltplan der Waagenelektronik	18

1. Einleitung

Ich bin eigentlich durch eine andere Jugend-Forscht-Gruppe, bei der ich auch mitmache, auf die Idee dieses Projektes gekommen. Wir benötigten dort eine sehr genaue Waage, aber leider hat es zeitlich nicht mehr gepasst, eine eigene Waage zu bauen. Also habe ich gesagt, dass ich die Konstruktion dieser Waage als Einzelprojekt weiterführe.

Eine Waage dient dazu, so genau wie möglich das Gewicht bzw. die Masse eines Objektes zu bestimmen. Die Masse eines Objektes wird durch Vergleich mit einer bekannten Masse bestimmt, während das Gewicht, das ortsabhängig ist, z.B. durch die Kraft gemessen werden kann, mit der sie einen Körper verformt. Moderne elektronische Waagen verwenden das gleiche Prinzip, dabei wird als verformbarer Körper häufig ein Piezokristall verwendet.

Ich habe mir zum Ziel gesetzt, eine einfache Balkenwaage zu bauen, bei der die Vergleichskraft aber nicht durch Massestücke oder eine Federkraft entsteht, sondern durch die Anziehungskraft einer Strom durchflossenen Spule. Sie soll außerdem nicht mit einer herkömmlichen mechanischen oder digitalen Anzeige ausgestattet sein, sondern komplett über einen PC gesteuert werden. Das hat den Vorteil, dass die Waage ohne viel aufwändige zusätzliche Elektronik sehr genau angesteuert und ausgelesen werden kann. Der Messbereich soll im Grammbereich liegen mit einer Auflösung in der Größenordnung von einigen Milligramm.

2. Vom Gewicht zum digitalisierten Wert

2.1 Herkömmliche Waagen

Bei den meisten mechanischen Waagen (Küchen-, Personenwaagen) wird durch das zu wiegende Objekt eine Feder verformt. Über ein Getriebe wird dabei ein Zeiger über eine Skala geführt, an der man das **Gewicht** ablesen kann.

Bei Balkenwaagen, dazu gehören z.B. Apothekerwaagen oder die Kartoffelwaage (oder auch Dezimalwaage der Landwirte) wird durch Auflegen oder Verschieben von Massestücken auf einem Hebelarm eine Gegenkraft erzeugt. Wenn sich die Waage im Gleichgewicht befindet, kann die **Masse** des Objektes anhand der aufgelegten Gewichtsstücke ermittelt oder an der Skala des Hebelarms abgelesen werden.

Eine Besonderheit ist die Briefwaage, hier wird durch die Masse des Objektes ein Gewichtsstück seitlich ausgelenkt, bis ein Gleichgewicht entstanden ist. Auch hier wird die **Masse** bestimmt.

Elektronische Waagen verformen in der Regel einen Piezokristall, die dabei entstehende Spannung wird gemessen und der Wert in eine digitale Anzeige umgewandelt. Diese Waagen messen wiederum das **Gewicht** des Objektes.

Grundsätzlich gilt: Beim Vergleich mit einer Vergleichsmasse wird die **Masse**, bei Aufwenden einer Gegenkraft das **Gewicht** eines Objektes bestimmt.

2.2 Prinzip meiner Waage

Äußerlich sieht meine Waage aus wie eine Balkenwaage. Jedoch wird das Gleichgewicht durch Erzeugen einer Gegenkraft hergestellt, also messe ich das Gewicht der Objekte. Das hat den Nachteil, dass meine Waage eigentlich an jedem neuen Aufstellungsort geeicht werden müsste, wenn ich die Masse angeben will.

2.2.1 Erzeugen der Gegenkraft

Statt einer Feder verwende ich einen Elektromagneten, der auf einen Permanentmagneten eine Kraft ausübt, wenn durch die Spule ein Strom fließt. Dabei gehe ich davon aus, dass der Zusammenhang zwischen Strom und Kraft proportional ist. Dies gilt nur, wenn das Ferrit-Material meines Permanentmagneten sich nicht durch das magnetische Feld der Spule beeinflussen lässt, wie es z.B. bei Stahl passieren kann (magnetische Hysterese wegen der Permanenz des Materials).

Damit durch die Spule ein Strom fließt, muss sie an eine Spannungsquelle angeschlossen werden. Der Strom durch die Spule ergibt sich dann nach der Gleichung

$$I = \frac{U}{R} \quad (1)$$

wobei U die Spannung der Spannungsquelle, R der ohmsche Innenwiderstand der Spule und I der Strom durch die Spule ist. Solange sich die Spule nicht erwärmt, bleibt der Widerstand konstant.

Für meine Zwecke benötige ich also eine Gleichspannungsquelle, die sich sehr genau einstellen lässt. Dafür bietet sich ein Digital-Analog-Wandler (DA-Wandler) an, dessen Ausgangsspannung digital eingestellt werden kann. Um den benötigten Strom für die Spule zu erhalten, wird diese Spannung von einem gegengekoppelten Operationsverstärker mit einer Transistor-Leistungsendstufe verstärkt. Die Gegenkoppelung greift dabei die Spannung direkt an der Spule ab, wodurch Nichtlinearitäten der Transistoren ausgeglichen werden.

2.2.2 Erkennen des Gleichgewichtszustandes

Die Spannung an der Spule wird solange verändert, bis der Waagebalken im Gleichgewicht steht, das erkennt man daran, dass er sich in seiner Ruhelage befindet. Um dies zu erkennen, benutze ich eine Lichtschranke. Normalerweise kann diese nur eine Auskunft darüber geben, ob der Lichtweg unterbrochen ist oder nicht, wobei das Fenster 1 .. 2 mm groß sein kann, was für meine Zwecke zuviel ist. Ich nutze deswegen die Tatsache aus, dass der als Empfänger dienende Phototransistor einen Strom liefert, der sich von hell nach dunkel nicht schlagartig, sondern kontinuierlich ändert. Eine als Komparator angelegte Verstärkerschaltung erlaubt es mir damit, dieses Fenster auf Bruchteile eines Millimeters zu verringern. Die Auswertung dieses Signals liefert mir eine Aussage, ob der Balken zu hoch, zu tief oder genau richtig steht.

2.2.3 Digitalisierung per Computer

Die Ansteuerung eines DA-Wandlers erfolgt über einzelne Bits. Bei dem von mir in der endgültigen Fassung verwendeten Wandlers werden 12 Bit benötigt - entsprechend einer Auflösung von 1 : 4096. Zum Setzen dieser Bits verwende ich den Parallelport eines Computers, wodurch sich die

Ansteuerung des Wandlers vom elektronischen Aufwand her stark vereinfacht, da so die meiste Arbeit von einem Programm erledigt werden kann. Über diesen Port kann ich auch den Gleichgewichtszustand abfragen. Die Anzeige des ermittelten Wertes erfolgt ebenfalls per Programm auf dem Bildschirm, was einer digitalen Anzeige entspricht.

3. Praktische Realisierung

Zur praktischen Umsetzung der Idee sind drei Aufgaben zu bewältigen. Ich benötige einen hinreichend präzisen mechanischen Aufbau des eigentlichen Waagesystems, eine für den Anschluss an den Parallelport geeignete Schaltung zur Ansteuerung des DA-Wandlers sowie zur Abfrage der Lichtschranke sowie ein Programm zur Steuerung der Waage und der Anzeige des Messwerte.

3.1 Aufbau der Mechanik

Die Mechanik habe ich soweit wie möglich recht einfach gehalten. Sie besteht aus einem zweiseitigen symmetrischen Waagebalken, der mit zwei Kerben auf einer Schneide aufliegt und sich somit reibungsarm und fast ohne Seitenspiel auf- und abbewegen kann. An jedem Arm dieses Balkens ist in jeweils gleichem Abstand vom Drehpunkt eine Waagschale aus Glas aufgehängt, eine für den zu wiegenden Gegenstand, die andere für Tariergewichte. Eine schematische Darstellung dieses Aufbaus findet sich in Abb. 1, ein Bild des Schneidenlagers in Bild 1.



Bild 1: Schneidenlager mit Auslegern

An der Seite des Waagebalkens, wo die Waagschale für die Tariergewichte angebracht ist, befindet sich ein senkrecht angeordneter quaderförmiger keramischer Permanent-Magnet und eine Metallzunge für die Lichtschranke zur Positionserkennung. Der Magnet taucht in eine auf der Basisplatte

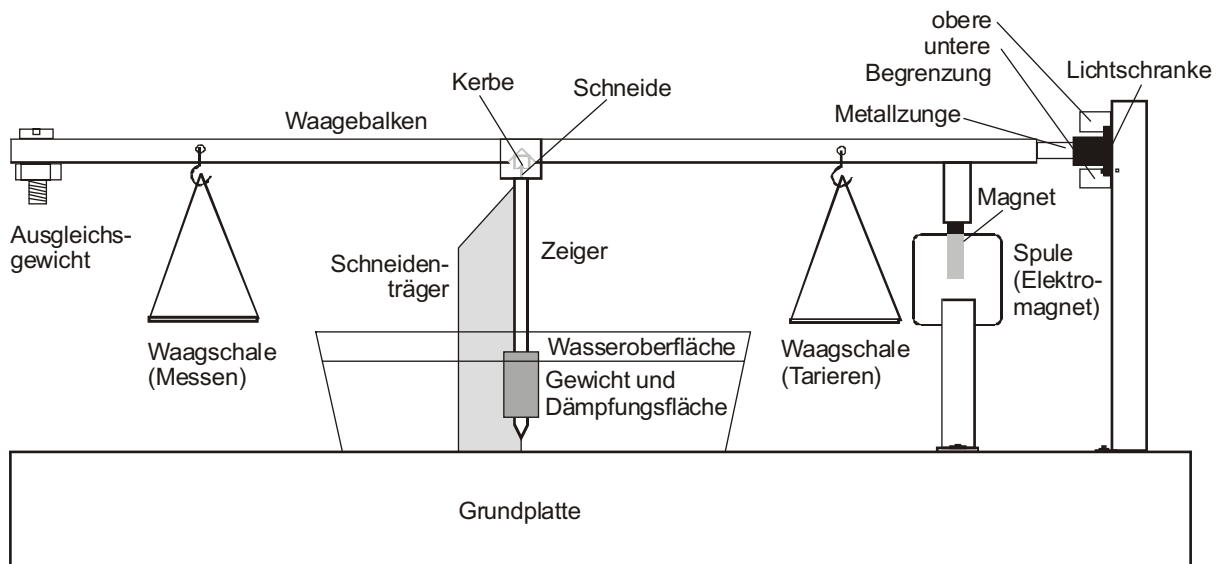


Abb. 1: Schematische Darstellung des mechanischen Aufbaus der Waage

feststehende eisenlose Spule mit sehr vielen Windungen ein. Die Metallzunge bewegt sich gleichzeitig mit dem Waagebalken innerhalb einer Gabellichtschränke auf und ab.

In der Mitte des Waagebalkens sind nach unten zwei gleich lange senkrechte Ausleger mit Gewichten angebaut, um den Schwerpunkt unter die Aufhängung zu verlagern und somit aus dem labilen ein stabiles Gleichgewicht zu machen. Im labilen Gleichgewicht ist es unmöglich, die Waage auszurufen, da der Waagebalken immer zu einer Seite hin wegkippen würde. An dem Ende des Waagebalkens, der dem Magneten gegenüber liegt, ist außerdem noch eine Metallschraube mit Mutter als Gegengewicht zu dem Magneten und der Metallzunge angebracht.

Die beiden Gewichtsstücke an den Auslegern tauchen in zwei Schalen ein, die mit Wasser gefüllt sind. Da die Waage sehr leichtgängig gelagert ist, neigt sie dazu, bei einer Auslenkung auf- und abzupendeln. Durch das Eintauchen der Gewichtsstücke in das Wasser wird diese Schwingung stark genug gedämpft, sodass nach hinreichend kurzer Zeit der Gleichgewichtszustand erreicht und das Gewicht abgelesen werden kann.

3.2 Schaltungsaufbau

3.2.1 DA-Wandler HDG-0807

Die allerersten Versuche habe ich mit einem Digital-Analog-Wandler vom Typ HDG-0807 gemacht. Die Schaltung dazu ist sehr einfach, sie besteht nur aus dem IC, einem Potentiometer und einem Entkoppelungskondensator. Um die Schaltung auszuprobieren, habe ich ein Programm in Microsoft Qbasic geschrieben, mit dem man die Bits D0-D7 sowie die Bits STROBE, AUTOFEED, INIT und SELECT am Parallelport einzeln setzen kann. Das Listing des Programmes ist im Anhang zu finden.

Leider hat diese Schaltung nicht wie erwartet funktioniert, weil der HDG-0807 ein 8-Bit Echtzeit-Videowandler ist, und nicht, wie ich dachte, ein normaler 11-Bit DA-Wandler. Die Datenbits D0 - D7 liefern zwar eine mit dem Zahlenwert linear steigende Spannung (Bildaten), allerdings beginnt diese nicht bei 0 Volt, sondern bei ca. 0,8 Volt. Zwei weitere Bit dienen zur Erzeugung der Signalpegel für die Horizontal- und Vertikalaustastung eines FBAS-Signals, während eine High-Flanke am dritten Bit die Übernahme der Daten in einen internen Speicher bewirkt. Da wir den Baustein aus einer „Ausschlachtplatine“ entnommen hatten und kein Datenblatt vorlag, konnten wir die Eingänge nur anhand der vorliegenden Beschaltung identifizieren, was zu obiger Fehlannahme führte.

3.2.2 DA-Wandler TI-5680V

Der nächste Versuch mit einem DA-Wandler vom Typ TI-5680V (ebenfalls aus einer „Ausschlachtplatine“) war dann erfolgreich, allerdings ist die Schaltung nicht ganz so einfach wie die erste. Hier handelt es sich um einen 12-Bit-Wandler, der allerdings 3 Versorgungsspannungen benötigt. Zwei Potis dienen zum Einstellen des Nullpunktes und der Maximalspannung, damit ist das Einstellen nicht so einfach wie bei der Schaltung mit dem HDG-0807. Dieser Wandler liefert eine Ausgangsspannung im Bereich von -5,000 bis +5,000 Volt, wobei das höchste Bit die Polarität bestimmt.

3.2.3 Leistungsverstärker

Da der Ausgang des DA-Wandler keine großen Ströme liefern kann, brauchte ich aber noch einen Leistungsverstärker für die Ansteuerung der Spule. Da die Kennlinie von Transistoren nicht linear verläuft, verwende ich sie in Zusammenhang mit einem OP als Regelverstärker. Damit ist der Leistungsverstärker zur Ansteuerung der Spule recht einfach gehalten: Er besteht aus dem Universal-Operationsverstärker $\mu A741$, zwei Transistoren BD139 (PNP) und BD140 (NPN), einem Widerstand zur Rückkoppelung und einem Potentiometer zum Einstellen der Schleifenverstärkung. Je nach Polarität der Spannung am Ausgang des OP's liefert entweder der obere oder der untere Transistor den Strom, der durch die Spule fließen soll. Der Abgriff der Spannung für die Gegenkoppelung direkt am Spulenanschluss stellt sicher, dass die Schaltung an der Spule bis auf einen konstanten Faktor k immer genau der Spannung entspricht, die vom DA-Wandler geliefert wird. Die Verstärkung ist notwendig, da die Spule im Bereich von -12 bis +12 Volt betrieben wird, während der Wandler ja nur von -5 bis +5 Volt arbeitet. Für die Spannung an der Spule gilt somit:

$$U_{Sp} = k \cdot U_{DA} \quad \text{mit} \quad k = \frac{12V}{5V} = 2,4 \quad (2)$$

Bei der hier für den OP gewählten Schaltung gilt

$$k = \frac{R_g}{R_v} \quad (3)$$

dabei ist R_g (P6, s. Schaltplan im Anhang) der Gegenkoppelungswiderstand von der Spule zum invertierenden Eingang des OP und R_v (R13) der Ableitwiderstand gegen Masse. Zum genauen Einstellen der richtigen Schleifenverstärkung ist R_g als 20-Gang-Feintrimmpotentiometer ausgeführt. Der gelötete Aufbau der Platine ist in Bild 2 zu sehen.

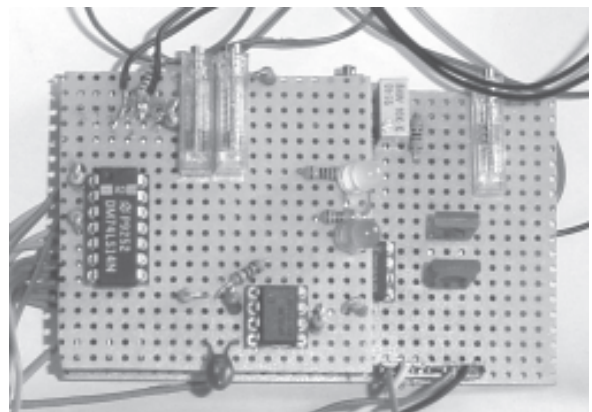


Bild 2: Platinenaufbau der Schaltung

3.2.4 Lichtschrankenabfrage

Die Schaltung zur Positionsrückmeldung besteht aus zwei OP's (MC1458), zwei TTL-Invertern (74LS14), ein paar Widerständen, Dioden und Potis. Der erste OP dient als Komparator, er vergleicht die durch den Strom des Phototransistors am Widerstand R5 erzeugte Spannung mit einer vorgegebenen Spannung, die mit P4 eingestellt werden kann. Auf Grund seiner Verstärkung, die mit P5 eingestellt werden kann, ergibt sich innerhalb der Lichtschranke ein kleiner Bereich, in dem die Ausgangsspannung des OP nahezu linear von -10 bis +10 Volt verläuft. Der zweite OP ist als invertierender Verstärker mit der Verstärkung 1 geschaltet, er liefert die gleiche Spannung, jedoch mit entgegengesetzter Polarität.

Die Ausgänge der beiden OP's gehen über einen Spannungsteiler und eine Schutzdiode (TTL-Pegelanpassung) auf die Eingänge zweier Inverter mit Schmitt-Trigger-Eingängen, die sehr präzise

bei Überschreiten einer Spannung von 2,4 Volt ihren Ausgang auf Low und erst bei Unterschreiten einer Spannung von 1,5 Volt wieder auf High schalten.

Wenn beide OP's eine Spannung in der Nähe von 0 Volt liefern (Gleichgewichtszustand der Waage), sind folglich beide Ausgänge High, in den anderen beiden möglichen Fällen zeigt der auf Low stehende Ausgang an, ob der Waagebalken zu hoch oder zu niedrig ist. Es können nie beide Ausgänge gleichzeitig Low sein.

Im Bild 3 sieht man die Anordnung von der Spule (unten rechts im Bild) mit dem in sie eintauchenden Permanentmagneten, der an einem Ausleger am Waagebalken befestigt ist sowie zwischen den beiden Plexiglasblöcken als mechanische Begrenzer die Lichtschranke mit der Unterbrecherzunge des Waagebalkens.

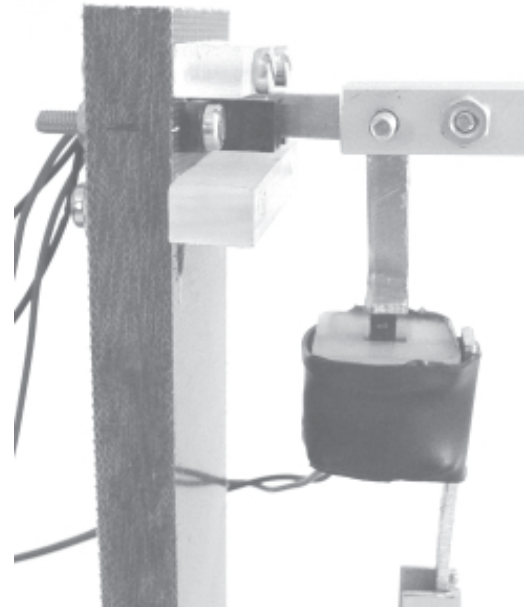


Bild 3: Spule und Lichtschrankensensor

3.2.5 Computeranschluss

Für den Anschluss an den PC verwende ich den Parallel- oder Druckerport des Rechners. Er stellt mit seinen acht Datenleitungen, vier frei programmierbaren abgehenden Steuerleitungen sowie drei eingehenden Statusabfrageleitungen die ideale Schnittstelle für meine Zwecke dar. Da die Leitungen alle mit TTL-Pegeln arbeiten, kann ich sowohl den DA-Wandler als auch die beiden Ausgänge der Lichtschrankenabfrage direkt an diesen Port anschließen.

3.3 Computer-Programm

Zur Erstellung des Programmes arbeite ich unter DOS mit dem Basic-Interpreter Microsoft QBasic. Dieses Programm ist zwar schon recht alt, aber für meine Zwecke voll ausreichend und auf jedem Rechner nach Compilierung lauffähig.

Abb. 2 zeigt den Programmplan in Form eines Struktogramms. Zu Beginn wird *Wert* auf 4095 gesetzt und die Invertierungsmaske für die Portausgabe gesetzt. Das eigentliche Programm besteht aus einer Schleife, die durch einen beliebigen Tastendruck beendet wird. Innerhalb dieser Schleife wird geprüft, ob die Lichtschranke zu hoch oder zu tief steht. Ist das der Fall, wird *Wert* für den DA-Wandler entsprechend erhöht oder erniedrigt. Anschließend erfolgt unabhängig vom Lichtschrankenstatus die Ausgabe des aktuellen Wertes von *Wert*.

Hauptprogramm

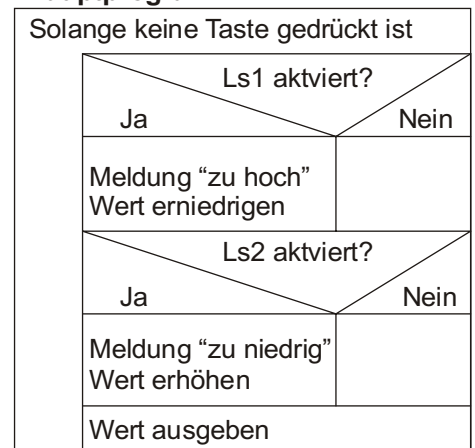


Abb. 2: Struktogramm des Hauptprogrammes

Zum Erhöhen oder Erniedrigen von *Wert* muss zuvor geprüft werden, ob der Maximal- oder Minimalwert bereits erreicht ist, da *Wert* nur Werte im Bereich von 0 .. 4095 annehmen darf. Die entsprechenden Struktogramme zeigen die Abb. 3 und 4.

Zur Ausgabe des Wertes muss berücksichtigt werden, dass die Bits D8, D9 und D11 (also das 9., 10. und 12. Bit) durch den LPT-Port invertiert werden. Dazu verwende ich eine XOR-Verknüpfung (eXclusive OR), die bitweise diejenigen Bits der Zahl invertiert, die in der XOR-Maske zuvor festgelegt wurden. Da die Zugriffe auf die Ports nur Byte-weise erfolgen können, werden aus dem umgewandelten Bitmuster von *Wert* die unteren 8 Bit in *Wert1* und die oberen 4 Bit in *Wert2* gespeichert und anschließend an die beiden Port-adressen ausgegeben. Vor Beendigung der Routine muss die Bitinvertierung von *Wert* wieder rückgängig gemacht werden, da *Wert* als Zählvariable verwendet wird.

Das vollständige Programm sowie der Quellcode für ein Testprogramm, mit dem ich die Kommunikation zwischen Programm und Waagenelektronik bitweise testen konnte, finden sich im Anhang.

3.4 Funktionstest

Zuerst habe ich grundsätzlich alle Schaltungen mit einem Digitalvoltmeter bzw. einem Oszilloskop / Speicheroszilloskop durchgetestet. Dann habe ich die Verbindung zum PC hergestellt und das jeweilige Test-Programm gestartet.

Zur Überprüfung der Lichtschrankenelektronik habe ich einen lichtundurchlässigen Gegenstand in die Lichtschranke gehalten. Zur Kontrolle der Spulensteuerung habe ich die Spannung an der Spule mit einem Digitalvoltmeter und dem Oszilloskop kontrolliert, letzteres ist nötig, da OP-Schaltungen manchmal zum Schwingen neigen, was in diesem Fall sehr unangenehm wäre. Nachdem Mechanik, Elektronik und Programm wie gewünscht funktionierten, konnte ich die Waage mit Eichgewichten eichen und erste Messungen zur Zuverlässigkeit der Waage durchführen.

Wert erniedrigen

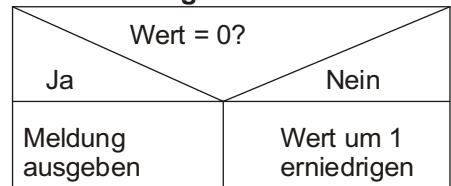


Abb. 4: Struktogramm für „Wert erniedrigen“

Wert erhöhen

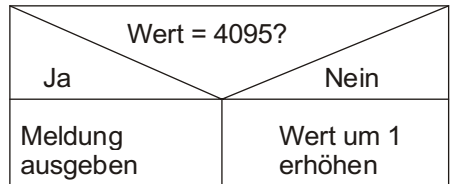


Abb. 3: Struktogramm für „Wert erhöhen“

Wert ausgeben

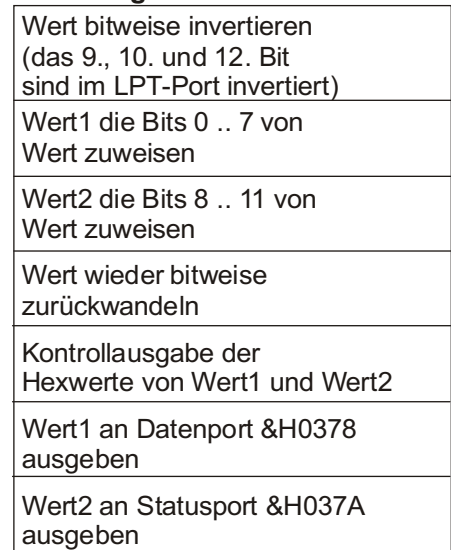


Abb. 5: Struktogramm für „Wert ausgeben“

4. Messergebnisse

4.1 Nullpunktstabilität

Die Waage wird so austariert, dass der Waagebalken sich beim Startwert von *Wert* gerade noch am oberen Anschlag befindet. Anschließend wird das Programm gestartet. Der Zähler wird nun automatisch so weit verändert, bis die Ruheposition erreicht ist. Dieser Wert wird als Nullwert gespeichert. In meinem ersten Versuch habe

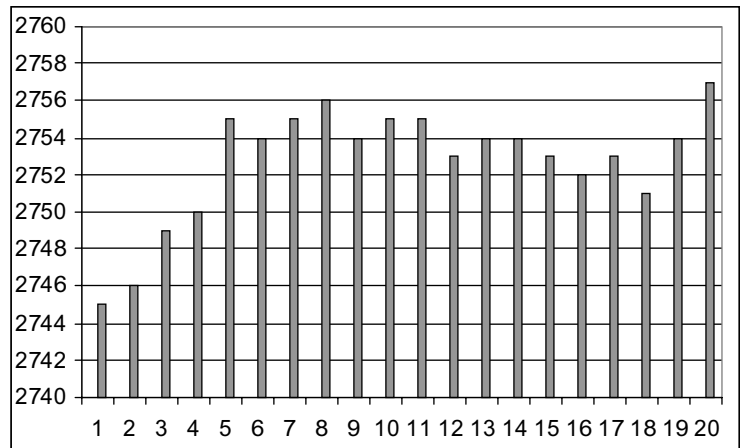


Abb. 6: Nullpunktstabilität der elektronischen Waage

ich allerdings nicht berücksichtigt, dass bei einem Anfangswert von 4095 bereits ein Spulenstrom fließt, der auf den Magneten eine nach oben gerichtete Kraft ausübt, deshalb sind meine Werte um ca. 1500 Einheiten nach unten verschoben.

Das Diagramm nach Abb. 6 zeigt den zeitlichen Verlauf von *Wert* über einen Zeitraum von 300 Sekunden. Man kann gut erkennen, dass es etwa 60 Sekunden dauert, bis sich die Waage eingependelt hat. Danach schwanken die Werte nur noch um +/- 2 Zählerwerte. Da der bei Digitalsteuerungen unvermeidliche Digitalisierungsfehler bereits +/- 1 Zähler bewirkt, ist diese ein recht gutes Ergebnis.

4.2 Erster Test mit Kunststoffröhrchen

Bei meinen ersten Messungen habe ich zuerst, wie in 4.1 beschrieben, die Waage austariert und das Programm gestartet, sodass die Ruheposition erreicht ist. Dies habe ich etwa 10 mal gemacht, bis sich die Waage eingependelt hat, außerdem wollte ich damit sichergehen, dass die Waage, ihre Elektronik und das Programm richtig funktionierten. Der so ermittelte Wert lag etwa bei 3500. Dann habe ich bei laufendem Programm ein kleines Gewichtsstück (Platinen-Abstandshalter aus Kunststoff, Gewicht: 0,46 Gramm) aufgelegt und gewartet, bis der Gleichgewichtszustand wieder erreicht war. Nachdem das geschehen war, habe ich mir den Wert notiert und ein weiteres gleichartiges Gewichtsstück aufgelegt. Ich konnte bis zu 4 dieser Gewichtsstücke auflegen, ohne dass das Programm eine Fehlermeldung ausgab. Dem Diagramm in Abb. 7 kann man entnehmen, dass die Werte bereits angenähert auf einer Geraden liegen.

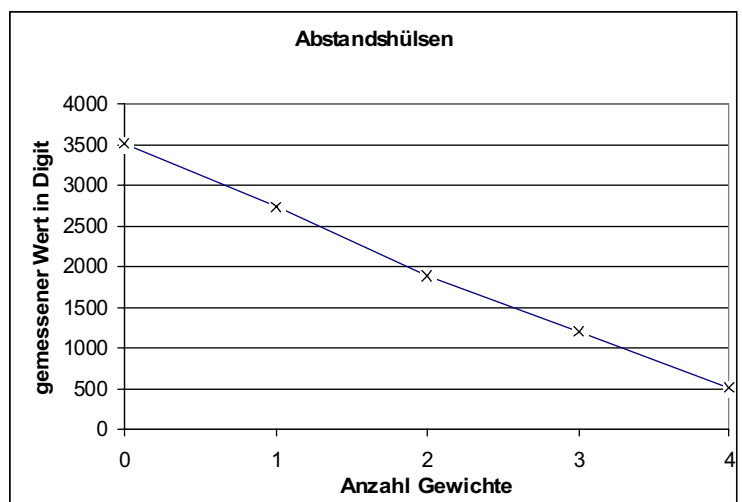


Abb. 7: Erster Versuch zur Gewichtsbestimmung

4.3 Überprüfung mit Apotheker-Gewichten

Um mit der Waage ein Gewicht angeben zu können, muss bekannt sein, welcher Messwert zu welchem Gewicht gehört. Um das zu überprüfen, benötigt man Gewichte mit bekannter Masse. Dazu hat mir die Hubertus-Apothek e Hermannsburg für einen Tag einen geeichten Gewichtssatz ausgeliehen, mit dem ich die nachfolgend beschriebene Messung ausgeführt habe.

Zunächst wurde die Waage tariert und der entsprechende Messwert notiert. Diese Tariierung habe ich vier Mal wiederholt. Dann wurde ein Gewicht auf die Messwaagschale aufgelegt und ebenfalls vier Mal der Anzeigewert bestimmt und notiert. Diese Messung führte ich für Gewichte von 200, 500, 1000, 1500 und 2000 mg durch.

Zur Auswertung habe ich das Programm MS Excel verwendet. Damit wurde für jede Messung zunächst der Mittelwert gebildet. Mit diesen berechnete ich die Differenz des Messwertes zum Tariierungsmesswert. Die obere Graphik in Abb. 8 zeigt den Zusammenhang zwischen den so berechneten Werten und dem Gewicht. Die Messpunkte liegen recht gut auf einer Geraden. Als Trendlinie liefert Excel eine Gleichung der Funktion

$$y = 1,664x + 42,9 \quad (4)$$

darin ist x das Gewicht in Milligramm und y der zu erwartende Messwert. Um eine Gleichung für das Gewicht zu erhalten, muss ich diese Gleichung nach x auflösen, dabei erhält man

$$x = \frac{y - 42,9}{1,664} \quad (5)$$

Zu einem Messwert von z.B. 1739 gehört demnach ein Gewicht von

$$x = \frac{1739 - 42,9}{1,664} = 1019$$

als 1019 mg. Dieses stimmt recht gut mit dem tatsächlichen Gewicht von 1000 mg überein.

Wenn die Krafterzeugung der Waage linear mit dem Wert des Zählers zusammenhängt, sollte diese Gleichung für jedes Gewicht im Messbereich der Waage anwendbar sein. Zur Überprüfung habe

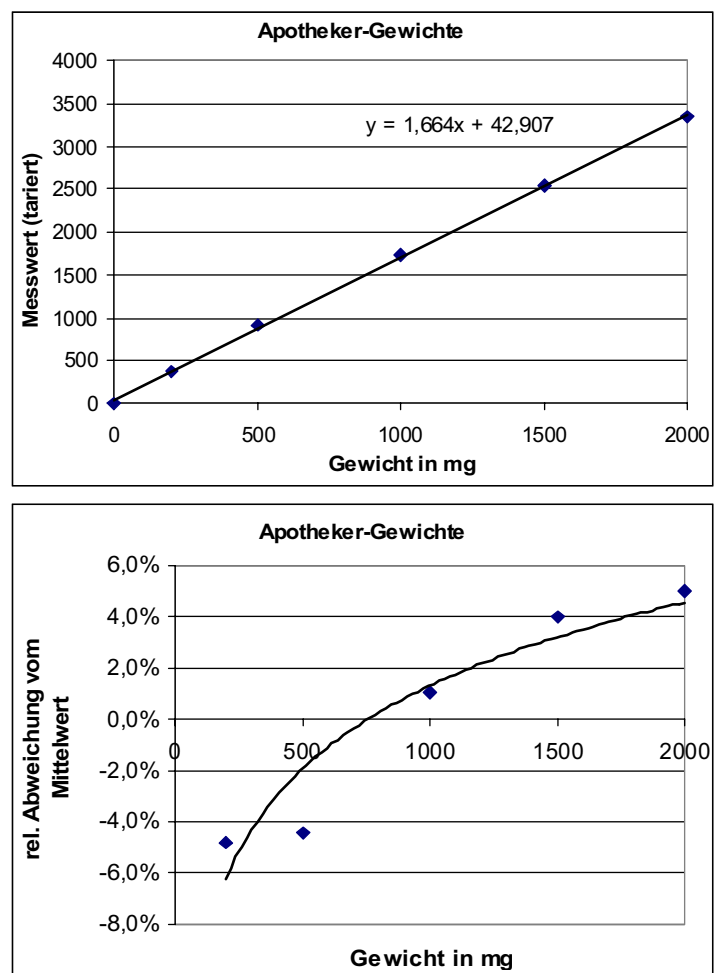


Abb. 8: Auswertung der Eichmessung mit Apotheker-Gewichten

ich für jede Messung den Quotienten von Messwert und Gewicht bestimmt und dafür einen Mittelwert berechnet. Dann habe ich die prozentuale Abweichung des jeweils berechneten Quotienten von diesem Mittelwert bestimmt. Das Ergebnis ist im unteren Graphen von Abb. 8 wiedergegeben. Bei einer idealen Waage hätte ich eine Gerade auf der x-Achse bekommen müssen. Man kann allerdings deutlich eine Abweichung von diesem idealen Verlauf sehen, der noch nicht einmal einen linearen Zusammenhang erkennen lässt.

Deswegen muss ich meine ursprüngliche Annahme zurücknehmen, dass die durch die vom DA-Wandler erzeugte Spannung proportional zur erzeugten Gegenkraft ist. Ich vermute, dass die Magnetkraft des Permanentmagneten sich doch von der Magnetkraft der Spule beeinflussen lässt. Mir fehlen die physikalischen Kenntnisse, um diesen Vorgang genauer beschreiben zu können.

4.4 Überprüfung mit Reißzwecken

Um weitere Messungen vornehmen zu können, habe ich mir einen Satz von Reißzwecken zusammengestellt, die ich gereinigt habe. Ich gehe davon aus, dass alle Reißzwecken das gleiche Gewicht haben.

Mit diesen Reißzwecken als Gewichtsstücke habe ich die gleiche Messung gemacht wie mit den Apothekergewichten, allerdings mit jeweils fünf Einzelmessungen für die Mittelwertbildung. Die Messwerte habe ich genauso ausgewertet wie in 4.3 beschrieben, sie sind in Abb. 9 wiedergegeben.

Auch hier liegen die Messwerte auf einer Geraden, allerdings mit einer anderen Funktion für die Trendlinie, weil auf der x-Achse nun die Anzahl angegeben ist. Mit Glg. (5) kann ich aber zu jedem Messwert das zugehörige Gewicht berechnen, so erhält man für 9 Reißzwecken

$$x = \frac{3499 - 42,9}{1,664} = 2077$$

also 2077 mg, damit erhalte ich für das Gewicht einer Reißzwecke

$$m = 231 \text{ mg}$$

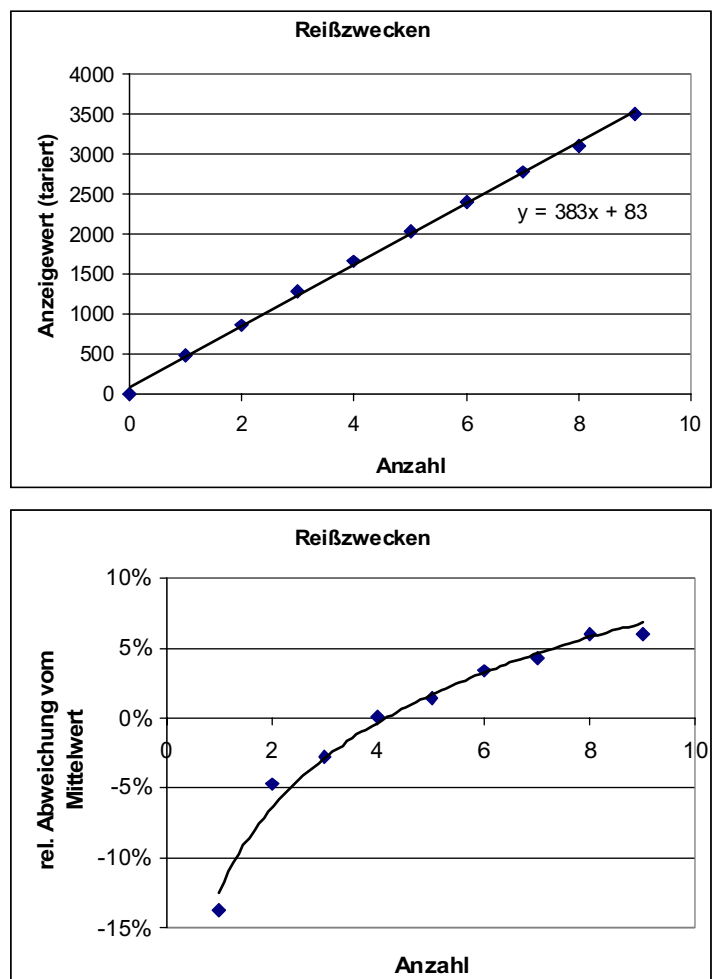


Abb. 9: Auswertung mit Reißzwecken als Gewichtsstücke

Nehme ich den Messwert 471 für einer Reißzwecke, erhalte ich daraus

$$x = \frac{471 - 42,9}{1,664} = 257$$

also 257 mg. Wegen der Nichtlinearität meiner Waage weicht aber der Wert bei 9 um ca. 5% nach oben, bei 2 um ca. 6% nach unten ab (laut Trendlinie), was einen Gesamtfehler von etwa 11% ausmacht. Und tatsächlich weicht meine Berechnung für 2 Gewichte um 11% von der Messung mit 9 Gewichten ab.

Das heißt, dass ich in meinem Programm das Messergebnis entsprechend der prozentualen Abweichung, die jedem Messwert (Variable *Wert* im Programm) zugeordnet werden kann, korrigieren muss. Dafür spricht auch, dass die beiden Abweichungskurven in Abb. 8 und 9 (jeweils unterer Graph) nahezu den gleichen Verlauf haben.

5. Folgerungen aus den Ergebnissen

Es ist mir gelungen, eine Waage zu bauen, die sehr empfindlich im Bereich von 0 bis 2 Gramm auf Gewichtsänderungen im Milligrammbereich reagiert. Die Gewinnung eines Messwertes ist allerdings deutlich zeitaufwändiger als bei kommerziellen Waagen, weil ich vor jeder Wägung die Waage zunächst „warmlaufen“ lassen und anschließend tarieren muss, außerdem dauert es ohne Mittelwertbildung zwischen 10 s und 100 s, bis die Waage eingependelt ist. Wenn ich zur Erhöhung der Messgenauigkeit mehrere Messungen durchführe, verlängert sich die Messzeit entsprechend.

5.1 Verbesserung der Linearität

Leider hat sich gezeigt, dass die vom Programm derzeit gelieferten Messwerte nicht die geforderte Linearität aufweisen, die für eine „echte“ Wägung mit einem auf Milligramm genauen Ergebnis notwendig ist. Dies kann ich aber z.B. dadurch beheben, dass ich in mein Programm eine Korrektur einbaue, die zu jedem Messwert einen Korrekturfaktor verwendet, den ich aber noch mit bekannten Gewichten ermitteln muss. Für andere zu messende Gewichte muss ich dann entsprechend interpolieren, da ich nicht für jeden der fast 4000 möglichen Messwerte einen eigenen Korrekturfaktor bestimmen kann.

5.2 Absolute Genauigkeit

Ein Digit meines Messwertes entspricht einer Gewichtsänderung von ca. 0,5 mg. Versuche mit einem Gewicht und sehr vielen Einzelmessungen (bis zu 80) haben gezeigt, dass nach ca. 60 Messungen die Werte nur noch maximal um +/- 4 Digits schwanken. Damit hätte meine Waage eine Messgenauigkeit von +/- 2 mg, was noch um den Faktor 5 besser ist als die Genauigkeit der Präzisionswaage, die uns in der AG zur Verfügung steht (Messgenauigkeit: 10 mg), allerdings ist diese mit einer Messzeit von weniger als einer Sekunde erheblich schneller.

5.3 Weitere Verbesserungsmöglichkeiten

Durch Verwendung eines schnelleren Rechners (zur Zeit verwende ich einen 286SX mit 8 MHz Taktfrequenz) könnte man die Messzeit verkürzen.

Wenn ich statt des Permanentmagneten einen Eisenkern aus Dynamoblech verwende, könnte sich die Linearität steigern lassen, allerdings würde dabei der Messbereich erheblich kleiner werden, da die magnetische Kraft allein durch das magnetische Feld der Spule erzeugt werden müsste. Dem könnte man entgegenwirken, indem man eine andere Spule benutzt, die bei gleicher Spannung ein stärkeres magnetisches Feld erzeugt.

Zur Zeit ist das Programm darauf ausgelegt, eine möglichst große Kontrolle über das Verhalten der Waage auszuüben. Für einen Messeinsatz sollte es für den Bediener noch erheblich vereinfacht und in seinem Ablauf stärker automatisiert werden (Tarierung, Mittelwertbildung, Datenspeicherung).

6. Literaturverzeichnis

- [1] E. Böhmer, „Elemente der angewandten Elektronik“, Vieweg
- [2] „ttl – Taschenbuch“, Franzis-Verlag
- [3] „Operationsverstärker - Datenbuch“, Franzis-Verlag
- [4] StD H. Lindner, Dr.-Ing. H. Brauer, Dr.-Ing. C. Lehmann:
„Elektrotechnik - Elektronik: Grundlagen u. Begriffe“, VEB Fachbuchverlag Leipzig 1988
- [5] Datenblatt des D-A-Wandlers TI-5680V der Firma Texas Instruments
- [6] Handbuch zum Schneider GW-Basic
- [7] W. Wang: „Programmieren für Dummies“, mitp-Verlag 2000

7. Danksagung

Ich bedanke mich bei meinem Betreuungslehrer, Herrn StD Thomas Biedermann, für seine große Geduld und Zeiteinsatz. Außerdem bedanke ich mich bei der gesamten Familie Biedermann, die mich an langen Forschungstagen ertragen und gut gepflegt hat. Mein Dank gilt auch dem Apotheker Peter Vogel, der mir seine Eichgewichte zur Verfügung gestellt hat. Auch der Förderstiftung „Jugend Forscht“ gilt mein besonderer Dank, da sie das Experimentieren erst möglich gemacht hat. Last but not least bedanke ich mich bei meinen Eltern für ihre Geduld und Taxifahrerei.

DANKE!!!

A1 Programmlisting (Hauptprogramm)

```
50 xorval = 256 + 512 + 2048: status% = 0
55 ende = 0
60 tara = 0: faktor = 1000 / 1706
74 Rem langsamer Rechner:
75     waitcnt% = 0: maxwait% = 1000: waitinc% = 5
76 Rem schneller Rechner:
77 Rem waitcnt% = 0: maxwait% = 20000: waitinc% = 1
100 Cls
110 wert = 4095
112 LOCATE 1, 1
114 Print "Befehle: <T>arieren    <M>essen    <+/-> Wert manuell „ndern    <Q>uitt"
125 While INKEY$ <> "": Wend
150 GoSub 400: Rem Portausgabe

200 Rem Hauptprogramm
205 While ende = 0
207     LOCATE 25, 1: Print "Taste... ";
210     c$ = INKEY$
220     If c$ = "t" Or c$ = "T" Then GoSub 2000: Rem Tarieren
225     If c$ = "m" Or c$ = "M" Then GoSub 3000: Rem Messen
230     If c$ = "+" Then GoSub 1000: Rem Manuell erh"hen
235     If c$ = "-" Then GoSub 1000: Rem Manuell erniedrigen
240     If c$ = "q" Or c$ = "Q" Then ende = 1: Rem Programm beenden
250     If c$ = "s" Or c$ = "S" Then
251         wert = 4095: GoSub 400
252         waitcnt% = 0
253     End If
290     c$ = ""
295 Wend

298 Rem Hauptprogramm beenden
299 End

300 Rem Wert um 1 erniedrigen
301     If wert = 0 Then
302         LOCATE 25, 10
304         Print "Fehler: Wert zu klein!!";
310     Else
312         wert = wert - 1
318     End If
319 Return

350 Rem Wert um 1 erh"hen
351     If wert = 4095 Then
352         LOCATE 25, 10
354         Print "Fehler: Wert zu groá!!";
360     Else
362         wert = wert + 1
368     End If
369 Return

400 Rem Portausgabe
408     wert = wert Xor xorval
410     wert1 = wert And 255
420     wert2% = (wert And &HF00) / 256
430     wert = wert Xor xorval
440     LOCATE 25, 40
442     Print Hex$(wert2%); " "; Hex$(wert1); "    "; wert; "    ";
444     OUT &H378, wert1
446     OUT &H37A, wert2%
448     For i = 0 To waitcnt%: Next i
450 Return
```

Programmlisting (Hauptprogramm - Fortsetzung)

```
1000 Rem Steuerung über Tastatur
1020 LOCATE 25, 1: Print "Manuell "; c$; " ";
1040 If c$ = "+" Then wert = wert + 1
1050 If c$ = "-" Then wert = wert - 1
1060 GoSub 400
1065 LOCATE 25, 10
1067 status% = INP(&H379)
1070 If status% = 88 Then Print "Waage am unteren Anschlag ";
1080 If status% = 248 Then Print "Waage am oberen Anschlag ";
1090 If status% = 120 Then Print "Waage im Gleichgewicht ";
1095 Return
1200 Rem Messvorgang
1205 c$ = ""
1210 While c$ = ""
1220 c$ = INKEY$
1222 If c$ = "0" Then
1224 waitcnt% = 0
1226 c$ = ""
1228 End If
1229 status% = INP(&H379)
1230 If status% = 248 Then
1231 LOCATE 25, 10
1232 Print "Waage am oberen Anschlag";
1234 GoSub 300
1235 GoSub 400
1236 LOCATE 25, 10
1238 Print " ";
1240 End If
1250 If status% = 88 Then
1251 LOCATE 25, 10
1252 Print "Waage am unteren Anschlag";
1253 If waitcnt% < maxwait% Then waitcnt% = waitcnt% + maxwait%
1254 GoSub 350
1255 GoSub 400
1256 LOCATE 25, 10
1258 Print " ";
1260 End If
1265 If status% = 120 Then
1267 If waitcnt > 0 Then
1270 waitcnt% = waitcnt% - 1
1275 End If
1280 End If
1298 Wend
1299 Return

2000 Rem Tarieren
2007 LOCATE 25, 1: Print "Tarieren..";
2010 GoSub 1200
2020 tara = wert
2099 Return

3000 Rem Messen
3007 LOCATE 25, 1: Print "Messen... ";
3010 GoSub 1200
3020 ergebnis = (tara - wert) / faktor
3030 LOCATE 20, 10
3040 Print "Ermitteltes Gewicht in mg: "; ergebnis; " ";
3050 Return
```

A2 Programmlisting (Bit-Test-Programm)

```
10 Cls
20 Dim a(12)
30 For i = 1 To 12
34 a(i) = 0
40 Next i
90 x% = 0: z = 1
100 c$ = INKEY$
101 If c$ = " " Then End
110 If (c$ = "1") Then xp = 1: GoSub 500
120 If (c$ = "2") Then xp = 2: GoSub 500
130 If (c$ = "3") Then xp = 3: GoSub 500
140 If (c$ = "4") Then xp = 4: GoSub 500
150 If (c$ = "5") Then xp = 5: GoSub 500
160 If (c$ = "6") Then xp = 6: GoSub 500
170 If (c$ = "7") Then xp = 7: GoSub 500
180 If (c$ = "8") Then xp = 8: GoSub 500
190 If (c$ = "9") Then xp = 9: GoSub 500
200 If (c$ = "a") Then xp = 10: GoSub 500
210 If (c$ = "b") Then xp = 11: GoSub 500
220 If (c$ = "c") Then xp = 12: GoSub 500
230 If (c$ = "+") Then GoSub 800
240 If (c$ = "-") Then GoSub 900
250 If (c$ = "s") Then GoSub 1000
270 x = 0
271 If a(1) = 1 Then x = x + 1
272 If a(2) = 1 Then x = x + 2
273 If a(3) = 1 Then x = x + 4
274 If a(4) = 1 Then x = x + 8
275 If a(5) = 1 Then x = x + 16
276 If a(6) = 1 Then x = x + 32
277 If a(7) = 1 Then x = x + 64
278 If a(8) = 1 Then x = x + 128
279 If a(9) = 0 Then x = x + 256
280 If a(10) = 0 Then x = x + 512
281 If a(11) = 1 Then x = x + 1024
282 If a(12) = 0 Then x = x + 2048
290 GoSub 400
298 LOCATE 1, 1
299 For i = 1 To 12: Print a(i);: Next i: Print " "; Hex$(x),

300 GoTo 100

400 wert1 = x And 255
405 wert2% = (x And &HF00) / 256
410 Rem PRINT HEX$(wert2%); " "; HEX$(wert1)
420 OUT &H378, wert1
430 OUT &H37A, wert2%
450 Return

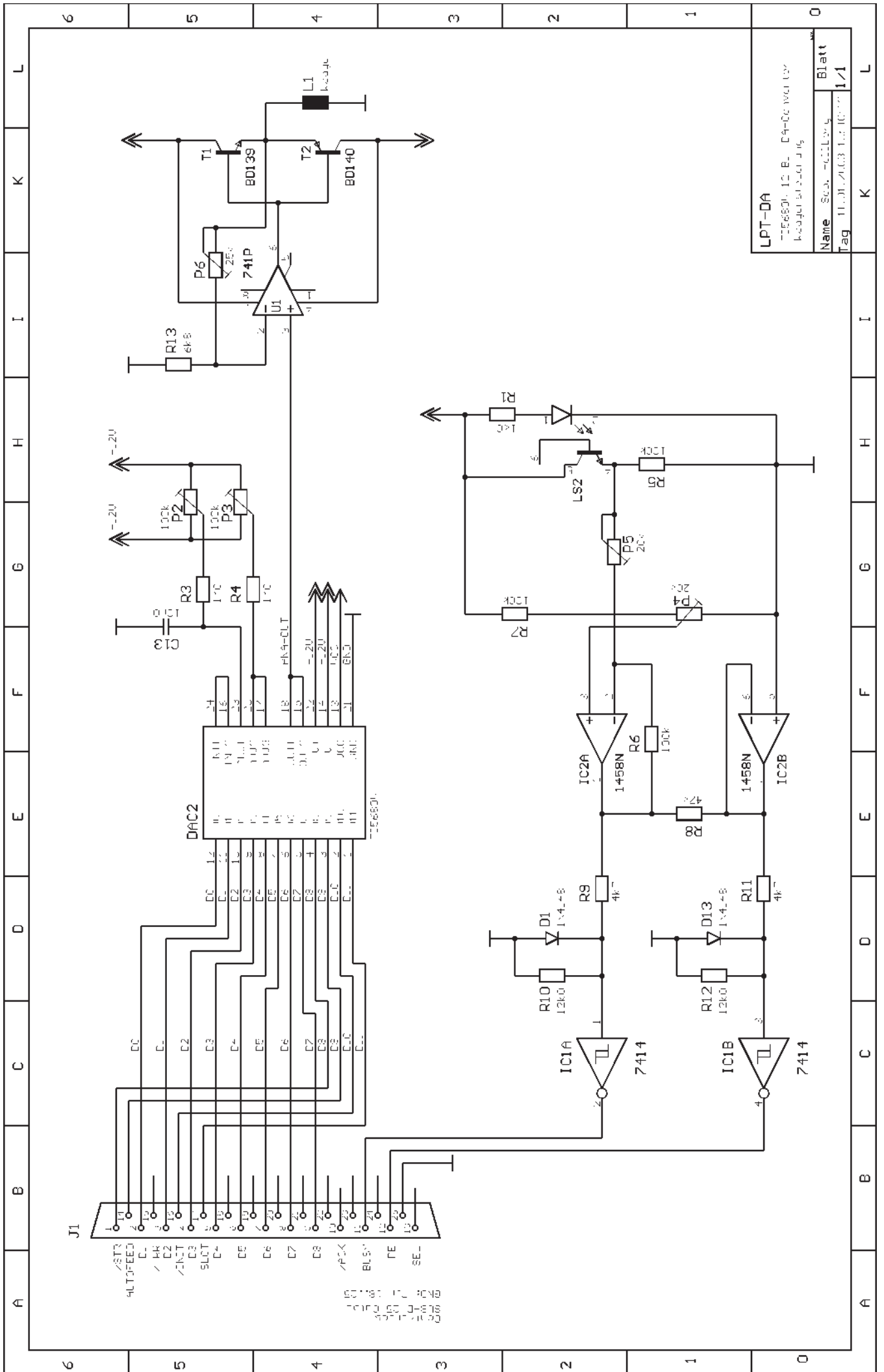
500 If a(xp) = 1 Then a(xp) = 0 Else a(xp) = 1
540 Return

800 For i = 1 To 12
810 a(i) = 1
820 Next i
830 Return

900 For i = 1 To 12
910 a(i) = 0
920 Next i
930 Return

1000 x = 0
1005 xp = 256 + 512 + 2048
1010 While INKEY$ = ""
1020 x = x Xor xp
1030 GoSub 400
1040 x = x Xor xp
1050 x = x + 1
1060 Wend
1070 Return
```

A3 Schaltplan der Waagenelektronik



LPT-DA	
TTC6801, IC 1458N, BD139, BD140, Kosag	
Name	Scale-Elektronik
Tag	11.01.2003 14:10:00
Blatt	1/1