

# XESCOE

| x-tendable environment for soundcard operated experiments |



## Benutzerhandbuch

- ▶ Umgang mit der Software
- ▶ Versuchsanleitungen
- ▶ Auswertung und Erklärung der Versuche
- ▶ Inkl. CD „XESCOE“

## Inhaltsverzeichnis

<b>Vorwort</b>	<b>1</b>
▶ Über XESCOE	1
<b>CD „XESCOE“</b>	<b>2</b>
▶ Inhalt der CD-ROM	2
▶ CD	2
<b>Software</b>	<b>3</b>
▶ Installation	3
▶ Versuchsdurchführungen	3
▶ Skripting zur Integration neuer Versuche	4
<b>Versuche</b>	<b>10</b>
▶ g-Versuch ( <i>Bestimmung der g-Konstanten</i> )	10
▶ Schallgeschwindigkeit	11
▶ WALITO ( <i>Lichtschraken an der Soundkarte</i> )	13
▶ Abhängigkeit von Kraft und Beschleunigung (+WALITO)	14

Das vorliegende Handbuch ist ein Produkt der Projektgruppe „Schulversuche mit der Soundkarte“ im Seminarkurs 2004/2005 „Computereinsatz im naturwissenschaftlichen Unterricht“ unter der Leitung von Herrn Schich am Friedrich-List-Gymnasium Reutlingen. Der Kurs befasste sich damit, neue Möglichkeiten zu finden, den Computer als Unterrichtsmedium ins Gymnasium zu integrieren. Hierzu fand sich unsere Gruppe, bestehend aus Niklas Reisser, Benedikt Rudolph und Oliver Stickel zusammen und wir wählten das Thema „Messwerterfassung mit der Soundkarte“, was soviel bedeutet wie die Erstellung von physikalischen Versuchen mit anschließender Auswertung an einem Computer, für uns aus. Die Schnittstelle, also den Übergang zwischen Versuchsaufbau und Computer stellt hierbei die Soundkarte dar. Das Ziel war es, in projektorientierter Arbeit ein Produkt zu erstellen, das sich anschließend im Physikunterricht auch wirklich einsetzen lässt und nicht, wie es an der Schule zu unserem Leidwesen so oft geschieht, ins Leere zu arbeiten. Unter diesem Leitgedanken entwickelten wir die Struktur unseres Projekts. Wir nahmen uns vor, ein Komplettpaket namens XESCOE – „x-tendable environment for soundcard operated experiments“, also sinngemäß „erweiterbare Arbeitsumgebung für Experimente auf Basis einer Soundkarte“ - zu erstellen. Diese sollte physikalische Versuche inklusive Anleitungen und Erklärungen, sowie ein komplett selbstgeschriebenes Messwerterfassungsprogramm für den PC umfassen. Diese Software sollte erweiterbar sein, so dass jeder interessierte Lehrer oder auch Schüler neue Versuche in die Umgebung integrieren kann. Damit ist gewährleistet, dass unsere Arbeit auch auf zukünftige Sicht nützlich sein kann und der Schule somit einen realen Nutzen bietet. Dieses Vorhaben haben wir umgesetzt und es liegt nun in Form dieses Handbuches, einer ausführlicheren Dokumentation und natürlich der beigelegten Software vor. Wir wünschen dem Benutzer hiermit viel Erfolg und genaue Ergebnisse bei der Arbeit mit XESCOE.

Niklas Reisser, Benedikt Rudolph, Oliver Stickel  
9.6.2005

Auf der CD finden sich zwei Unterordner:

- /Doku** Dieses Handbuch und die umfangreichere Dokumentation des Seminarprojektes als pdf - Dateien.
- /XESCOE** Die eigentliche Software. Im Unterordner /Sources findet sich der dazugehörige Quellcode.

Wenn das Setup nicht automatisch nach Einlegen der CD in Ihr Laufwerk aufgerufen wird, dann starten Sie die Setup-Routine *XESCOEsetup.exe* und folgen Sie den Anweisungen auf dem Bildschirm.

Zur Deinstallation rufen Sie entweder die entsprechende Option im Startmenü auf oder entfernen das Programm unter *Start - Einstellungen - Systemsteuerung - Software*, bzw. unter Windows XP unter *Start - Systemsteuerung - Software*.

Mit XESCOE lassen sich viele verschiedene Arten von Versuchen durchführen, das Grundprinzip ihrer Auswertung ist aber allen gleich. Bauen Sie den Versuch gemäß seiner Anweisungen auf und starten Sie dann ein beliebiges Programm zur Aufnahme von Audiodateien. XESCOE selbst wertet nur aufgezeichnete Audiodateien aus und führt diese Aufgabe nicht selbst durch. Unter Windows brauchen sie jedoch trotzdem kein zusätzliches Programm, da hier standardmäßig der *Audiorekorder* mitgeliefert wird. Sie finden ihn unter *Start - Programme - Zubehör - Audiorekorder* bzw. wenn Sie Windows XP benutzen unter *Start - Programme - Zubehör - Unterhaltungsmedien - Audiorekorder*. Bevor Sie mit der Aufzeichnung beginnen müssen Sie noch einige Einstellungen vornehmen bzw. überprüfen. Passen sie unter *Datei - Eigenschaften* die Qualität der Aufnahme ihren Wünschen an. Wählen Sie dazu unter *Formate* die Option *Aufnahmeformate* aus und klicken Sie auf *Jetzt konvertieren...* Wichtig ist hier, dass unter *Format PCM* ausgewählt ist. Sollten Sie sich nicht sicher sein, welche Einstellungen für Sie geeignet sind, dann wählen sie am Besten unter *Name* die Option *CD-Qualität* aus und bestätigen Sie zweimal mit *OK*. Danach müssen Sie überprüfen, ob der richtige Eingangskanal ausgewählt ist. Es gibt die Möglichkeit, den Versuchsaufbau an *Line-In* oder an *Mikrofon* anzuschließen, je nach Versuch ist es unterschiedlich was eher geeignet ist. Klicken Sie unter *Bearbeiten - Audioeigenschaften* im Bereich *Soundaufnahme* auf *Lautstärke*. Hier wählen Sie je nach Anschluss entweder *Eingang* oder

*Mikrofon* und stellen die Lautstärke entsprechend ein.

Führen Sie jetzt den Versuch durch und zeichnen Sie ihn dabei auf. Speichern Sie die Aufnahme danach in einem beliebigen Verzeichnis.

Starten Sie jetzt XESCOE entweder über das Startmenü, die Bildschirmverknüpfung oder direkt im Installationsverzeichnis.

Im ersten Schritt werden Sie gefragt, welchen Versuch Sie mit XESCOE durchführen wollen. Wählen Sie hier den entsprechenden Versuch aus, im Fenster *Versuchsbeschreibung* erhalten Sie nähere Informationen. Klicken Sie dann auf *Versuch starten*.

Auf der zweiten Seite werden Sie nach der aufgezeichneten Versuchsdatei gefragt. Geben Sie hier den Pfad an, an dem Sie Ihre Aufnahme vorher abgelegt haben. Hierzu werden im Programm selbst notwendige Informationen angezeigt, falls Sie nicht weiter wissen. Klicken Sie nach Eingabe des Dateipfades auf *Weiter*.

Jetzt öffnet sich Ihnen die Seite zur Versuchsauswertung. Diese sieht je nach Versuch unterschiedlich aus (s. nächster Abschnitt). Evtl. müssen hier noch weitere Eingaben gemacht werden, etwa die Länge einer gemessenen Strecke o.ä., dann wird ihnen das Ergebnis Ihres Versuchs präsentiert. Beenden Sie XESCOE, indem Sie auf den roten Button in der rechten oberen Bildschirmcke klicken.

Die dritte Seite von XESCOE soll für jeden Versuch anders aussehen und andere Ergebnisse liefern können. Daher gibt es die Möglichkeit, Aussehen und Funktionalität dieser Seite über ein sog. Skript zu steuern. Die Auflistung der verfügbaren Versuche auf der ersten Seite ist technisch gesehen eine Auflistung aller Skriptdateien, die sich im Unterverzeichnis *experiments* von XESCOE befinden. Ein Skript ist eine einfache Textdatei, die Zeile für Zeile von XESCOE ausgeführt wird. Variablen und mathematische Operatoren gibt es keine, nur Befehle. Jeder Befehl hat eine Liste von Parametern, die er verarbeitet. So würde man beispielsweise um Zahlen zu addieren, dem Additionsbefehl alle Summanden als Parameter übergeben. Jeder Befehl bildet einen

Rückgabewert, der wiederum einem anderen Befehl als Parameter übergeben werden kann. Um also z.B. eine Summe mit einer Zahl zu multiplizieren, erhält der Multiplikationsbefehl als einen Parameter den Additionsbefehl mit den beiden Summanden. So lassen sich beliebig tiefe Verschachtelungen realisieren. Die einzelnen Parameter werden durch spitze Klammern eingeschlossen.

*Beispiele:*

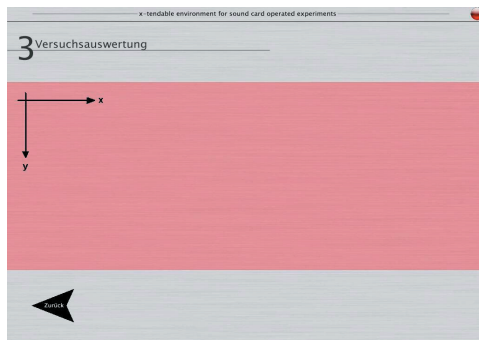
`add<2><5><1>` entspricht  $2 + 5 + 1$  und gibt 8 zurück  
`mul<add<2><5>><2>` entspricht  $(2 + 5) * 2$  und gibt 14 zurück

Für die Berechnungen werden natürlich die Wechselstellen der Signale gebraucht. Die Befehle orientieren sich dabei an den Stellen ohne Signal. Durch `findzerostart` wird der Anfang und durch `findzeroend` das Ende eines Abschnitts ohne Signal abgefragt. Dabei wird die Zeit ab Beginn der Audiodatei in Millisekunden zurückgegeben. Die Befehle erhalten zwei Parameter, der erste bestimmt den Kanal (muss auch bei Monoaufnahmen angegeben werden), der zweite die Nummer der entsprechenden Wechselstelle. Für beide Parameter gilt, dass die 0 den ersten Kanal bzw. die erste Wechselstelle angibt.

*Beispiel:*

`sub<findzeroend<0><0>><findzerostart<0><0>>` gibt die Dauer des ersten Abschnitts ohne Signal zurück.

Über Befehle lassen sich auch grafische Elemente auf dem Bildschirm ausgeben. Diese benötigen die Positionsangaben eines umgebenden Rechtecks.



Dabei muss beachtet werden, dass nicht die ganze dritte Seite zugänglich ist. Nur auf dem rot unterlegten Bereich (s. Bild) können Elemente dargestellt werden. Als Einheit werden keine Pixel verwendet, sondern, um von der Bildschirmauflösung unabhängig zu bleiben, relative Koordinaten,

wobei der rote Bereich 250 Einheiten breit und 100 Einheiten hoch ist. Die Koordinaten werden nach rechts und nach unten hin größer.

*Beispiel:*

`winin<125><50><80>` erzeugt ein Feld zur Texteingabe durch den Benutzer mit der oberen linken Ecke in der Mitte des roten Bereichs und einer Breite von 80 Einheiten. Da Eingabefelder immer einzeilig sind, ist die Höhe festgelegt und kann nicht verändert werden.

Um den geforderten Inhalt eines Textfelds klarszustellen gibt es auch noch einen alternativen Befehl für die Erzeugung von Textfeldern mit Namen `descrwinin`. Dieser nimmt als vierten Parameter zusätzlich eine Beschriftung entgegen die rechtsbündig unter dem Textfeld angezeigt wird.

Wird ein Textfeld wie im genannten Beispiel erzeugt, so wird ihm automatisch eine Nummer zugewiesen, die abhängig von der Anzahl der vorher erzeugten Textfelder ist. So erhält das im Skript als erstes benannte Feld die Nummer 0, das zweite die 1 usw. Über diese Nummer kann dann der Inhalt des Textfelds mithilfe des Befehls `getwinin` abgefragt werden.

*Beispiel:*

`mul<2><getwinin<0>>` nimmt den Inhalt des ersten Textfeldes mal 2.

Alle bisher beschriebenen Befehle werden sofort nach Erscheinen der dritten Programmseite aufgerufen. Es kann aber sein, dass bestimmte Befehle erst später ausgeführt werden sollen, da zum Beispiel ein Textfeld am Anfang immer leer ist und der Anwender erst Zeit braucht, die entsprechenden Daten einzugeben. Um dann eine Berechnung zu starten ist es am Besten, wenn er auf einen Button klicken kann. Im Skript kennzeichnet man einen Befehl, der erst beim Klicken eines Buttons ausgeführt wird dadurch, dass man ihn einem Befehlsblock zuordnet. Dies geschieht über den Befehl `block`. Wie bei den Textfeldern erhalten die Blocks in der Reihenfolge ihrer Benennung Identifikationsnummern. Ein Befehlsblock kann, wie der Name bereits sagt, mehrere Befehle enthalten.

In der Zeile unter dem `block`-Befehl muss eine Zahl stehen, die angibt, wieviele der folgenden Zeilen zum Block gehören sollen. Ein Button wird dann über den Befehl `button` erzeugt, der die Nummer des auszuführenden Blocks entgegennimmt.

*Beispiel:*

```
block
2
add<2><5>
mul<2><getwinin<0>>
button<20><10><Klick!><0>
```

erzeugt einen Button mit der Aufschrift „Klick!“ und ordnet ihm den erstgenannten Befehlsblock zu. Dieser besteht aus zwei Zeilen, nämlich einer Addition und einer Multiplikation mit der Abfrage eines Textfelds.

Für eine Versuchsauswertung fehlt jetzt lediglich noch die Möglichkeit, Text auf dem Bildschirm auszugeben. Der Befehl `textout` gibt einzeiligen Text mit festgelegter Maximalbreite aus.

*Beispiel:*

```
textout<20><10><80><Das Ergebnis ist ><add<2><3>>
Gibt „Das Ergebnis ist 5.00“ aus. Sollen Rechnungen in Text eingebaut werden, so muss dies über einen extra Parameter geschehen, ansonsten wird ein Befehl nicht ausgeführt, sondern ausgegeben.
```

Soll nicht nur eine einzelne Zeile, sondern ein ganzer Text ausgegeben werden, so steht der Befehl `winout` zur Verfügung. Er erzeugt ein Ausgabefenster ähnlich dem Versuchsbeschreibungsfenster auf der ersten Programmseite, allerdings mit variabler Überschrift. Der Inhalt des Ausgabefensters lässt sich formatieren. Ein „`§`“ im Text führt zu einem Zeilenumbruch, mit „`$`“, „`#`“ und „`|`“ wird der folgende Text kursiv, fett und fett-kursiv angezeigt. Ein Formatierungszeichen gilt bis zur nächsten Umstellung, bzw. bis zur Wiederholung des Zeichens, was den folgenden Text wieder in Normaldruck anzeigt.

*Beispiel:*

```
winout<20><10><100><70><Überschrift><Hier können
$beispielsweise #physikalische Erklärungen# angezeigt
werden.>
```

Erzeugt ein Ausgabefenster mit der Breite 100, der Höhe 70, der Betitelung „Überschrift“ und dem Inhalt „Hier können *beispielsweise physikalische Erklärungen* angezeigt werden.“

Außerdem gibt es die Möglichkeit zur Veranschaulichung der Berechnungen eine Visualisierung der WAVE-Daten anzuzeigen. Diese Aufgabe wird vom Befehl `visu` übernommen. Die ersten vier Parameter werden für Position, Breite und Höhe gebraucht, der vierte Parameter nimmt den Kanal entgegen, die letzten beiden Parameter geben den Start und das Ende der zu visualisierenden Daten innerhalb der Datei in Milisekunden an.

*Beispiel:*

```
visu<10><10><100><30><0><1000><5000>
Zeigt eine Visualisierung des ersten Kanals von Sekunde 1 bis 5 an.
```

### Referenz Skripting:

<code>add&lt;x&gt;&lt;y&gt;&lt;z&gt;</code>	gibt $x + y + z$ zurück
<code>sub&lt;x&gt;&lt;y&gt;&lt;z&gt;</code>	gibt $x - y - z$ zurück
<code>mul&lt;x&gt;&lt;y&gt;&lt;z&gt;</code>	gibt $x * y * z$ zurück
<code>div&lt;x&gt;&lt;y&gt;&lt;z&gt;</code>	gibt $(x / y) / z$ zurück
<code>quad&lt;x&gt;&lt;y&gt;</code>	gibt $x^y$ zurück
<code>sqrt&lt;x&gt;</code>	gibt die Wurzel aus $x$ zurück

`getzerostart<x><y>` gibt den Beginn des Abschnitts  $y$  ohne Signal auf Kanal  $x$  zurück

`getzeroend<x><y>` gibt das Ende des Abschnitts  $y$  ohne Signal auf Kanal  $x$  zurück

`textout<x><y><w><text><...>` erzeugt an der Stelle  $x | y$  ein Textausgabefeld ohne Hintergrund mit maximaler Breite  $w$  und gibt „text...“ aus

`winout<x><y><w><h><Titel><text><...>` erzeugt an der Stelle x | y ein Ausgabefenster mit der Breite w, der Höhe h, der Überschrift „Titel“ und dem Inhalt „text...“. Über „§“ erreicht man einen Zeilenumbruch, ansonsten wird automatisch bei Leerzeichen umgebrochen. „\$“ erzeugt kursiven, „#“ fetten und „|“ fett-kursiven Text.

`visu<x><y><w><h><c><a><b>` erzeugt an der Stelle x | y eine Visualisierung des Kanals c mit der Breite w und der Höhe h von Millisekunde a bis b.

`winin<x><y><w>` erzeugt an der Stelle x | y ein Texteingabefeld der Breite w und automatischer Identifikationsnummer

`descrwinin<x><y><w><Beschriftung>` wie `winin`, nur mit zusätzlicher Beschriftung

`getwinin<x>` gibt den Inhalt des Textfelds x zurück

`block`

`x`

benennt einen Befehlsblock mit den x folgenden Zeilen und automatischer Identifikationsnummer

`button<x><y><Titel><z>` erzeugt einen Button mit der Aufschrift „Titel“ und führt bei Klick den Befehlsblock z aus

`analbutton<x><y><z>` erzeugt einen Button mit der Aufschrift „Auswertung“ und führt bei Klick den Befehlsblock z aus

`descr<Beschreibung des Versuchs>` legt den Inhalt des Versuchsbeschreibungsfensters auf der ersten Programmseite fest. Der Inhalt lässt sich wie bei einem `winout` formatieren.

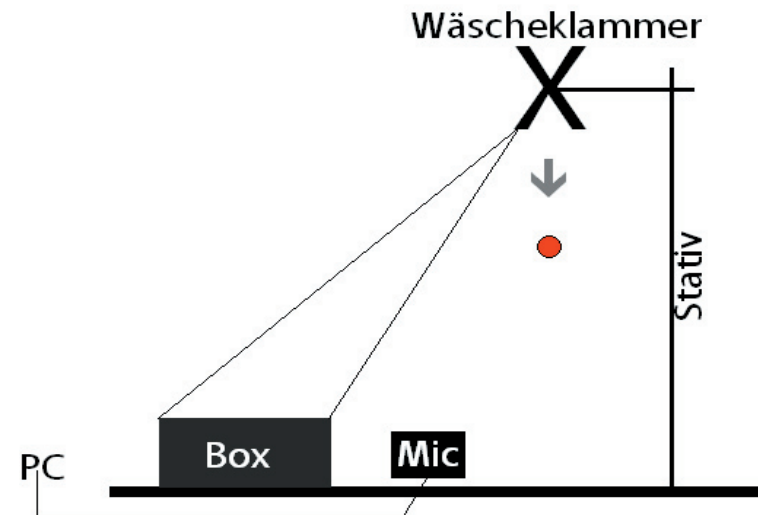
`setlim<x><y><z>` legt die Toleranzgrenze auf x (Werte von 0 - 100), die minimale Dauer ohne Unterbrechnungen für einen Abschnitt ohne Signal auf y und den minimalen Prozentanteil an Signalen oberhalb der Toleranzgrenze für einen Abschnitt mit Signal innerhalb der Zeit y auf z fest.

*Versuchsaufbau:*

Um den Versuch durchzuführen, brauchen Sie neben dem eigentlichen Versuch (die Batteriebox mit angeschlossener Wäscheklammer) noch folgende Teile:

- Einen Zollstock
- Verschiedene Stativteile: 1 Stange, 1 Tischhalterung, 1 Schraubklemme, 1 Kreuzverbindung
- Ein Mikrofon (liegen bei)
- Einen Computer mit Line In Anschluss, auf dem ein Programm zum aufnehmen von Wave-Dateien (z.B. Windows Audiorecorder) und XESCOE installiert ist.

Haben Sie die Teile zusammen, bauen sie den Versuch nach folgendem Schema auf:



Danach wird der Computer hochgefahren und der Audiorecorder gestartet. (Start – Programme - Zubehör – Unterhaltungsmedien - Audiorecorder) Testweise können Sie nun die Kugel in die Klammer stecken. Ertönt ein Summton, ist der Versuch einsatzbereit.



### Versuchsdurchführung:

1. Stellen Sie die Klammer auf die gewünschte Höhe ein und fixieren Sie diese dort.
2. Zuerst wird die Kugel in die Klammer gesteckt. Jetzt sollte ein Summton zu hören sein
3. Als nächstes muss die Audioaufnahme gestartet werden.
4. Jetzt folgt der eigentlich Versuch: Öffnen sie die Wäscheklammer, worauf die Kugel nach unten fällt und ein Aufschlageräusch verursacht.
5. Jetzt können Sie die Aufnahme stoppen, und die Datei speichern. (Näheres zum Umgang mit dem Audiorecorder entnehmen sie bitte dem „Software“ - Teil )

### Versuchsauswertung:

Zur Auswertung des Versuchs benutzen Sie bitte das Programm „XESCOE“. Dort finden Sie auch physikalische Hintergründe.

## Schallgeschwindigkeit

## Versuche

### Versuchsaufbau:

Für die Durchführung des Schallmessungsversuchs benötigen Sie folgende Teile:

- Den eigentlichen Versuch (eine Batteriebox mit 2 Mikrofonen)
- Eine Messleiste aus der Optik sowie zwei dazugehörige Wägelchen mit jeweils zwei Schrauben zum Fixieren.
- Eine Schallquelle (dem Versuch liegen „Knackenten“ bei)
- Einen Computer mit Line In Anschluss, auf dem ein Programm zum aufnehmen von Wave-Dateien (z.B. Windows Audiorecorder) und XESCOE installiert ist.
- Ein Lineal, Geodreieck oder etwas Ähnliches

Haben Sie alle Teile zusammen bauen sie den Versuch nach folgendem Schema auf:



Dazu stecken Sie die beiden Mikrofone in die Wägelchen und setzen diese auf der Leiste auf. Richtig ausgerichtet schließen diese ziemlich genau bündig mit der Kante des Wägelchens ab. Um das sicherzustellen benutzen sie bitte das Geodreieck. Dann fahren Sie den Computer hoch und starten den Audiorecorder.

### Versuchsdurchführung:

1. Als erstes stellen Sie bitte die gewünschte Entfernung zwischen den beiden Mikrofonen ein. Benutzen Sie dazu die Beschriftung der Leiste.
2. Dann starten Sie die Aufnahme im Audiorecorder. (Näheres zum Umgang mit dem Audiorecorder entnehmen sie bitte dem „Software“ - Teil)
3. Halten Sie nun die gewählte Geräuschquelle vor das erste Mikrofon und betätigen diese.
4. Stoppen sie die Aufnahme mit dem Audiorecorder und speichern sie die Datei ab.

### Versuchsauswertung:

Zur Auswertung des Versuchs benutzen Sie bitte das Programm „XESCOE“. Dort finden Sie auch physikalische Hintergründe.

Wie schon gesagt, ist der moderne Computer mit seiner präzisen internen Zeitintervallmessung ein optimales Messgerät. Bei etwas komplizierteren Versuchen wird nun aber der Wunsch nach anspruchsvolleren Inputgeräten als einem oder zwei Mikrofonen aufkommen, das Gerät der Wahl wäre hier natürlich eine Lichtschranke, da sie die Tür zu diversen neuen Versuchen öffnet. Im Prinzip ist das ein schöner Gedanke, nur stellt sich sehr bald ein elementares Problem: Eine Soundkarte ist ein Gerät zur Erfassung von - natürlich - Sound, also Wechselfrequenzsignalen bis ca. 20kHz. Eine Lichtschranke dagegen gibt eine quasistationäre Gleichspannung aus, die eine Soundkarte im Prinzip nicht verarbeiten kann.

Daraus ergibt sich die Notwendigkeit, ein Gerät zu bauen, welches in der Lage ist, die Outputsignale einer Lichtschranke zu verarbeiten und soundkartenverträgliche Wechselfrequenzsignale auszugeben. OStR. Klaus-Dieter Grüninger hat ein solches Gerät, den WALITO (Wandler Licht-Ton) konzipiert und eine Bauanleitung dazu auf dem Landesbildungsserver Baden-Württemberg veröffentlicht. Er hat uns auch beim Bau des WALITO unterstützt, weshalb wir ihm an dieser Stelle unseren ganz herzlichen Dank aussprechen wollen.

Prinzipiell arbeitet der WALITO nach einer „Modulationsmethode“. Sprich: Er erzeugt immer genau dann, wenn die Lichtschranke verdunkelt ist, über einen Generator ein Wechselfrequenzsignal, das an die Soundkarte geht. Es können maximal zwei Lichtschranken an den WALITO angeschlossen werden, welche dann über das Gerät auch gleich mit Strom versorgt werden. Der grundlegende Aufbau des Geräts ist: Lichtschranke - Invertierlogik - Timer - Soundkarte. Die Invertierlogik sorgt dafür, dass sowohl Lichtschranken, die bei Unterbrechung 0V, als auch solche, die dann 5V ausgeben, angeschlossen werden können. Beide Arten von Lichtschranken sind auf dem Markt erhältlich. Der Timer schwingt (erzeugt das gewünschte Wechselfrequenzsignal), sobald an seinem Eingang 5V anliegen, sind es 0V, ist er in Ruhe. Somit kann eine Soundkarte über einen ganz normalen Klinkestecker auf beiden Kanälen (eine Lichtschranke pro Kanal) angesteuert werden.

Zu S.30/31 Dorn-Bader: Physik 11

- Fragestellung: 1. Wie hängt die nötige Kraft  $F$  bei ein und demselben beschleunigten Fahrzeug von der gewünschten Beschleunigung  $a$  ab?  
2. Wie hängt die Kraft  $F$  vom Fahrzeug und seiner Beladung ab, wenn sich  $a$  nicht ändern soll?

### Zu Frage 1:

*Versuch 1:* a) Auf einer Schiene wird ein Wagen festgehalten, an dem eine beliebige Anzahl von Wägestücken der Masse  $m$  zieht. Auf dem Wagen wird die entsprechende Stange befestigt. Außerdem sind mithilfe von Stativen über der Schiene in beliebigem (nicht zu kleinem) Abstand 2 Lichtschranken angebracht (Siehe Abb. 1). Diese werden über den WALITO an die Soundkarte angeschlossen. Nach dem Aufzeichnen der Audio-datei muss in XESCOE nun das beschleunigende Gewicht in Gramm (wird intern in Newton umgerechnet) und die Strecke  $s$  zwischen den Lichtschranken in cm angegeben werden, anschließend kann der Versuch ausgewertet werden.



Abb. 1

b) Nun soll die wirkende Kraft geändert werden, das Gesamtgewicht aber konstant bleiben. Dazu wird eines der Wägestücke weggenommen und auf die Stange auf dem Wagen aufgesteckt (Abb. 2). So wird es weiterhin mitbeschleunigt ohne dass seine Gewichtskraft noch mitzieht. Anschließend erfolgt wiederum die Auswertung in XESCOE. Man kann diese Messung natürlich beliebig oft mit verschiedenen Gewichten und Strecken durchführen.





Abb. 2

Bsp.:

$F$ in N:	0,08	0,06	0,04	0,02
$a$ in m/s:	0,392	0,294	0,196	0,098

Wir sehen also: Bei gleich bleibender Masse ist bei  $n$ -facher Kraft die Beschleunigung ebenfalls  $n$ -mal so groß. Damit ist  $F \sim a$ .

### Zu Frage 2:

Wie hängt die Kraft  $F$  vom Fahrzeug und seiner Beladung ab, wenn sich  $a$  nicht ändern soll?

Diese Frage stellt sich deshalb, da z.B. Raketen auch im schwerelosen Raum zum Beschleunigen Kräfte benötigen, die von ihrer Beladung abhängen, obwohl diese dort nichts wiegt. Die Raketen sind also auch im Weltall träge.

Deshalb wollen wir mit folgendem Versuch überprüfen, ob ein Körper doppelter Masse für die gleiche Beschleunigung auch die doppelte Kraft benötigt, also doppelt so träge ist.

*Versuch 2:* Der Aufbau ist nahezu der selbe wie bei Versuch 1b), nur wird diesmal noch ein zweiter Wagen gleicher Masse an den ersten angehängt und gleichzeitig die Gewichte am Ende der Schnur verdoppelt (Siehe Abb. 3)



Abb. 3

Nach der Auswertung in XESCOE sehen wir, dass die Beschleunigung gleich bleibt wie in Versuch 1b).

Damit kommt man zu dem Schluss, dass die Kraft  $F$ , die man für eine bestimmte Beschleunigung braucht zu der Masse  $m$  des zu beschleunigenden Körpers proportional ist. Damit ist  $F \sim m$ .