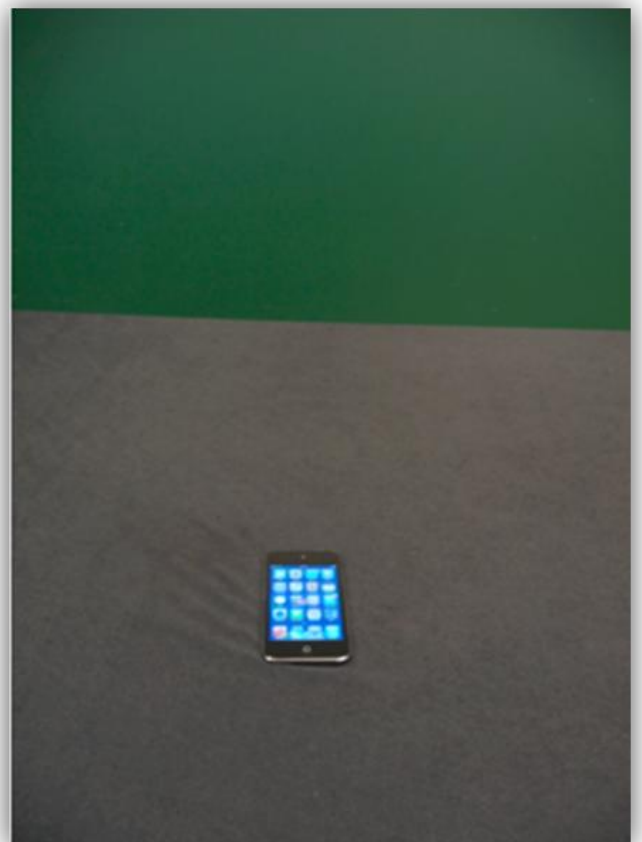


# Der iPod touch

## Eine Revolution im Unterricht



## Jugend forscht 2011

Florian Büttner und Maurice Wald

St. Michael-Gymnasium Bad Münstereifel

## ***Der iPod touch – eine Revolution im Unterricht***

### **1 Kurzfassung**

### **2 Einleitung**

### **3 Einsatz im Unterricht allgemein**

### **4 Einsatz im Physikunterricht**

#### **4.1 Physik-Apps**

#### **4.2 Experimente mit eingebauter Hardware**

#### **4.3 Experimente mit externer Hardware**

#### **4.4 Spannungs-Frequenz-Wandler**

### **5 Eigene Software**

#### **5.1 Einführung in das iOS SDK**

#### **5.2 Projekt: Eigenes Programm**

### **6 Ausblick**

### **7 Danksagung**

### **8 Literatur- und Linkliste**

#### **8.1 Literaturliste**

#### **8.2 Linkliste**

## 1 Kurzfassung

Technische Neuerungen haben oft die Welt verändert. Haben auch iPod touch und iPad das Potenzial den Unterricht zu revolutionieren? Wir glauben ja!

Um dies zu zeigen, haben wir für mehrere Fachbereiche eine Reihe kostenloser und käuflicher Apps zusammen gestellt, die im Unterricht gewinnbringend eingesetzt werden können. So erübrigt sich das Schleppen kiloschwerer Rucksäcke, weil die Schulbücher im iPod bzw. iPad als iBooks gespeichert sind. Aktuelle Updates – kein Problem. Auf weltweites Wissen zugreifen – durch das integrierte Internet eine Selbstverständlichkeit. Elendig nervige Software-Wartungen – durch Cloud Computing überflüssig. Besser Lernen mit Spaßfaktor.

Obwohl iPod touch und iPad in allen Fächern mit Gewinn eingesetzt werden können, legten wir unseren Schwerpunkt auf den Einsatz im Physikunterricht. Schon mit der eingebauten Hardware (z.B. Beschleunigungssensor, Mikrofon, Display, ...) und den vorhandenen Apps lassen sich zahlreiche Experimente durchführen (Akustik, Optik, Videoanalyse, ...). Eine Herausforderung für uns war es jedoch eine eigene externe Hardware zu entwickeln und hierzu passende Apps zu programmieren. Dazu bauten wir einen Spannungs-Frequenz-Wandler, so dass alle physikalischen Größen, die von Sensoren in Spannungen umgewandelt werden mit dem iPod erfasst werden können.

## 2 Einleitung

Lernen ohne Internet ist nicht mehr zeitgemäß, da es vollkommen unsinnig ist auf diesen großen Wissenspool ohne Not zu verzichten. Google, Wikipedia, Wolfram Alpha, ... sich selbst Wissen zu holen wird immer einfacher und immer wichtiger. Deshalb ist Recherchieren eine Fähigkeit, die man nicht früh genug lernen kann.

Wozu schleppen wir noch kiloschwere Rucksäcke, da alle Schulbücher als iBooks auf dem iPod bzw. iPad gespeichert werden können. Nervige Softwareprobleme gehören der Vergangenheit an, da durch die Einschränkungen von Apple nicht jeder Schüler ihre Funktion beeinträchtigen kann. Leerlaufzeiten durch Booten gehören auch der Vergangenheit an, da iPod und iPad keine Festplatte besitzen. Für uns Schüler ist Lernen mit diesen Geräten ein Lernen mit Spaßfaktor.

Obwohl iPod touch und iPad in allen Fächern mit Gewinn eingesetzt werden können, legen wir unseren Schwerpunkt auf den Einsatz im Physikunterricht. Schon mit der eingebauten Hardware (z.B. Beschleunigungssensor) und den vorhandenen Apps lassen sich zahlreiche Experimente durchführen. Eine Herausforderung für uns war es jedoch eine eigene externe Hardware zu entwickeln und hierzu passende Apps zu programmieren.

### 3 Einsatz im Unterricht allgemein

Wir haben eine Reihe kostenloser und käuflicher Apps zusammen gestellt, die im Unterricht gewinnbringend eingesetzt werden können. Hier einige Beispiele:

#### Sprachen

##### „Skype“

„Skype“, eine VoIP-Software, erlaubt es gratis über das Internet mit anderen Usern auf der ganzen Welt, teilweise auch über Video, zu telefonieren und zu chatten. Für unsere Arbeit dient sie in den Fremdsprachen zur besseren, kontinuierlicheren Korrespondenz zwischen verschiedenen Ländern, Sprachen und Kulturen und somit zu einem effektiven Lernen einer Sprache.



Abbildung 3.1 „Skype“

#### Geschichte, Sozialwissenschaft

##### Kamera



Abbildung 3.2 „Google Earth“

In den Gesellschaftswissenschaften lassen sich mit dem iPod Video-Interviews mit Hilfe der integrierten Kamera durchführen. Diese ist im Gegensatz zu den bisherigen Aufzeichnungsmethoden mobil und einfach zu bedienen.

##### Erdkunde

##### „Google Earth“

Mit „Google Earth“ kann der iPod-User Luft- und Sattelitenaufnahmen der ganzen Welt betrachten. Das ist für unser Projekt vor allem im Erdkundeunterricht sehr von Nutzen. Es werden keine Atlanten mehr gebraucht, man hat die ganze Welt in der Tasche.

# Jugend forscht 2011

## Der iPod touch – Eine Revolution im Unterricht

### Musik

In Kombination mit einer Dock-Station eignet sich der iPod wie kein anderes Gerät zur Wiedergabe von Musik.

### Sprachmemo

Die Sprachmemo-App erlaubt es mit dem integrierten oder einem externen Mikrofon vielfältige Dinge aufzunehmen. Im Musikunterricht kann man diese App zur Aufnahme von Musikstücken verwenden und diese so zu analysieren.



Abbildung 3.3 Sprachmemo



Abbildung 3.4 „iThoughts“

### Lehrer

#### „Teacher Tool“

Das „Teacher Tool“ ist eine Organisations-App, die dem Lehrer das Verwalten einer Klasse und die Unterrichtsplanung vereinfacht und teilweise abnimmt. Kursverwaltung, Bildsitzpläne, Notengebung, etc. machen diese App für den Lehrer äußerst nützlich.

#### „iThoughts“

Mit „iThoughts“ ist es möglich, Gedanken sortiert in Form einer Mindmap festzuhalten. Somit sorgt diese App für eine systematische Unterrichtsvorbereitung. Diese App ist nicht nur für den Lehrer sondern auch für die Schüler mit Gewinn einsetzbar.

### In allen Fachbereichen

#### „iBooks“

Eine revolutionäre App ist „iBooks“. Mit ihr kann man digitale Bücher downloaden und verwalten. Auch ist das Speichern von eigenen PDFs auf ihr möglich. Theoretisch können die „iBooks“ also die bisherigen Schulbücher ersetzen. Ihre Vorteile gegenüber den herkömmlichen Büchern sind, dass sie **viel leichter** und mobiler und dank Updates immer auf dem neusten Stand sind.

Diese wenigen Beispiele zeigen mit welchem großem Nutzen der iPod (iPad) in der Schule eingesetzt werden kann. Es würde den Rahmen dieser Arbeit sprengen, alle nützlichen Apps hier aufzuzählen.

## 4 Einsatz im Physikunterricht

### 4.1 Physik-Apps

Auch an dieser Stelle wollen wir nur einige Apps als Beispiele aufführen.

#### „Atom in a Box“

Mit „Atom in a Box“ kann man sehr leicht die quantenphysikalische Wahrscheinlichkeitsverteilung der Elektronen eines Atoms anhand eines 3D-Modells darstellen. Dies hilft den Schülern enorm eine Vorstellung von diesem abstrakten Atommodell zu verinnerlichen.

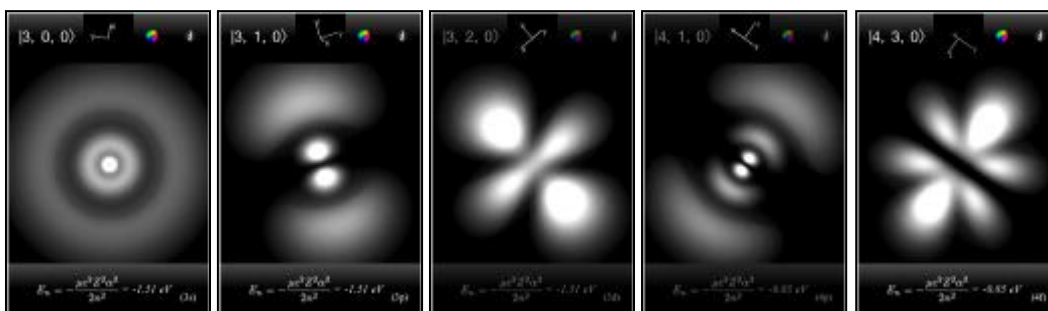


Abbildung 4.1 „Atom in a Box“

#### „Starmap Pro“



Abbildung 4.2 „Starmap Pro“



Abbildung 4.3 „Starmap Pro“ Jupiter

Diese App haben wir für die astronomischen Unterrichtseinheiten in der Physik gewählt. Sie ist den üblichen Sternenkarten haushoch überlegen, da sie sehr viel umfangreicher und interaktiver ist. „StarmapPro“ lässt sich nicht nur mit großem Gewinn im Unterricht selber einsetzen sondern auch bei abendlichen Feldbeobachtungen hilft es, sich am aktuellen Sternenhimmel optimal zu orientieren.

### Wissenschaftlicher Taschenrechner

Der kostenlose wissenschaftliche Taschenrechner, der alle Standardfunktionen aufweist, kann in der Mathematik und mehreren Naturwissenschaften eingesetzt werden.



Abbildung 4.4 Wissenschaftlicher Taschenrechner



Abbildung 4.5 „GraphCalculator“

### „GraphCalculator“

Der „GraphCalculator“, stellt zusätzlich noch Funktionen und Kurven grafisch dar.

## 4.2 Experimente mit eingebauter Hardware

### Kamera



Abbildung 4.6 Lichtstreuung und Überlagerung von Wellen wurden mit der iPod-Kamera aufgezeichnet

Eine Neuerung des iPod 4G sind die Kameras. Vor allem die HD-Kamera an der Rückseite des Geräts ist vielseitig innerhalb und außerhalb der Schule anwendbar (siehe Abb. 4.6).

### „Video Physics“

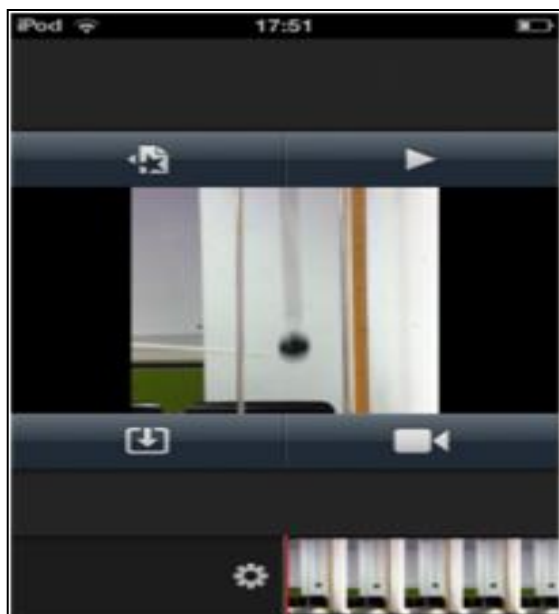


Abbildung 4.7 Federpendel

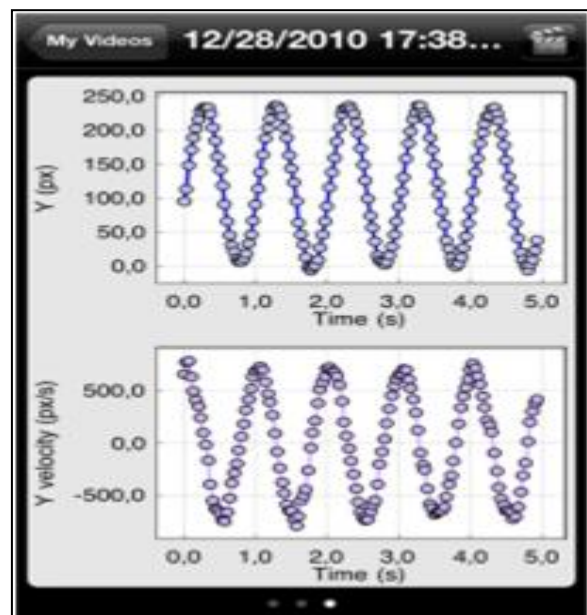


Abbildung 4.8 Versuchsergebnisse Federpendel

Jugend  forscht 2011  
Der iPod touch – Eine Revolution im Unterricht

VideoPhysics wird von „Vernier Software & Technology“ gratis zur Verfügung gestellt. Diese Software ermöglicht es, aufgezeichnete Videos physikalisch zu analysieren. Diese Videos hatten, dank der HD-Kamera des iPods, die bis zu 30 Bilder pro Sekunde aufnimmt, die entsprechende Qualität um ein ordentliches Versuchsergebnis zu erzielen. Die Ergebnisse aus den ausgewerteten Videos können auch grafisch dargestellt werden.

Mit Hilfe dieser App haben wir schließlich einige Versuche zum Thema Mechanik durchgeführt. Als erstes haben wir nicht nur einen inelastischen Stoß auf einer Luftkissenfahrbahn untersucht, sondern auch die Schwingung von Feder- und Fadenpendel.

## Display

### Kapazitives Display

Im Internet kann man nachlesen, dass der iPod touch ein kapazitives Display hat. Aber wie genau funktioniert das? Diese Frage lässt sich durch einen Versuch klären.

Also probierten wir alle möglichen Materialien, bis wir auf die Idee kamen, eine Möhre als Fingerersatz zu verwenden, da sie Wasser enthält, welches eine hohe Dielektrizitätszahl besitzt. Um diese These zu überprüfen benutzten wir also eine Möhre als Fingerersatz. Es funktionierte! Daraus folgerten wir, dass nicht die Leitfähigkeit, sondern die Dielektrizitätszahl eines Stoffes ausschlaggebend ist.



Abbildung 4.9 Kapazitives Display

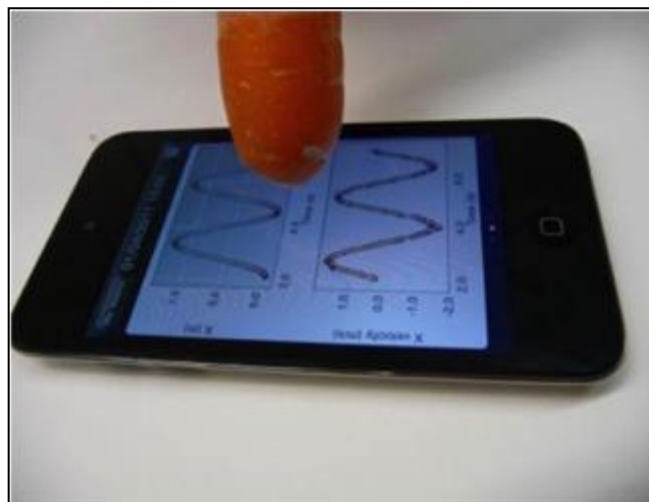


Abbildung 4.10 Kapazitives Display



Abbildung 4.11 Polarisationsfilter senkrecht

### Polarisiertes Licht

Eine weitere Besonderheit des Displays ist, dass es für den Physikunterricht eine Quelle für polarisiertes Licht darstellt. Diese Polarisation kann durch einen Polarisationsfilter je nach dessen Position unterschiedlich stark weggedreht werden. So kann das Display durch den Filter in einer bestimmten Position komplett verdunkelt werden. Nachdem wir das ausprobiert haben, legten wir ein Geodreieck und eine Folie zwischen das



Abbildung 4.12 Polarisationsfilter + Geodreieck

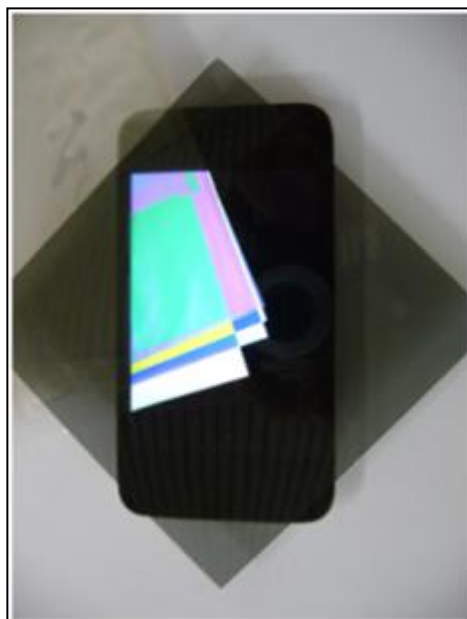


Abbildung 4.13 Polarisationsfilter + Folie

Display und den Polarisationsfilter. Da das Geodreieck bzw. die Folie die Polarisation leicht drehen sollten, müsste das Licht durch den Filter durchleuchten. Diese Theorie haben wir durch einen erfolgreichen Versuch bewiesen.

Durch diese Methoden können auch Dünnschliffe untersucht werden (*siehe Abb. 4.14*).

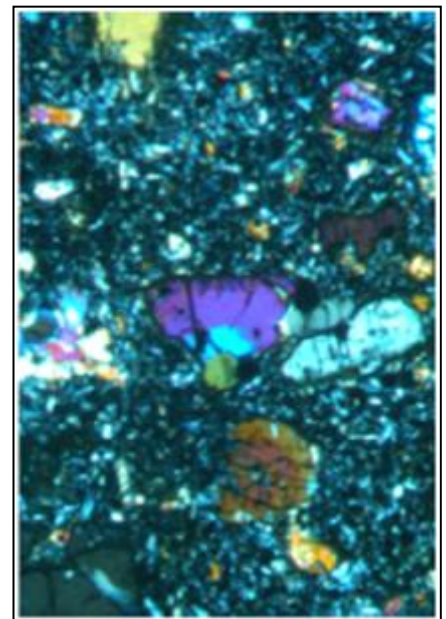


Abbildung 4.14 Dünnschliff Basalt

## Accelorometer

Aber auch der eingebaute Beschleunigungssensor ist von Nutzen. Mit ihm lassen sich nicht nur Beschleunigungen im Klassenraum sondern auch außerhalb z.B. auf der Achterbahn messen.

## Mikrofon

### Schwebung

Auch in der Akustik lässt sich der iPod für Versuche einsetzen. Wir haben z.B. mit der App „SignalScopePro“ eine Schwebung aufgezeichnet. Eine Schwebung entsteht, wenn sich zwei Töne ähnlicher Frequenzen gegenseitig überlagern und somit abwechselnd verstärken oder abschwächen. Die Versuchsergebnisse haben wir dann grafisch dar gestellt.

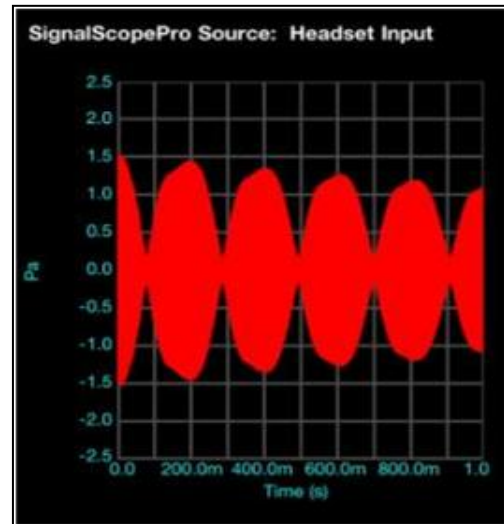
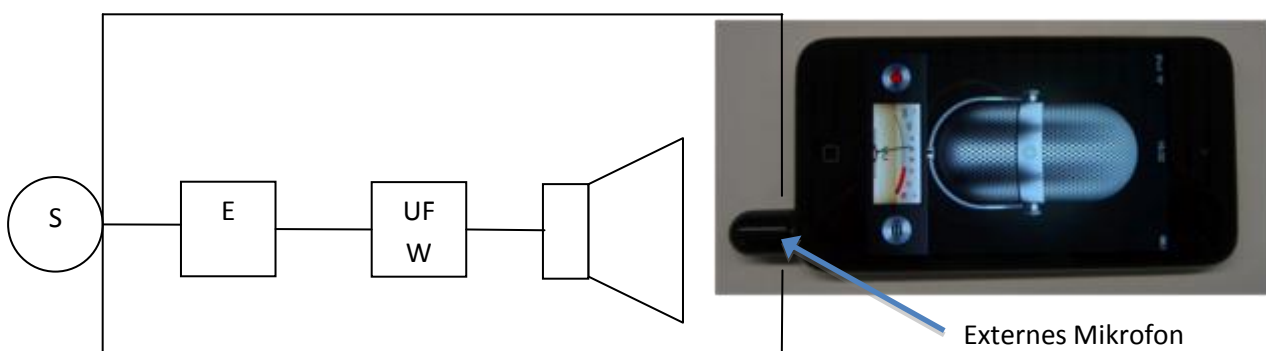


Abbildung 4.15 Schwebung

## 4.3 Experimente mit externer Hardware

Aber wir wollten nicht nur mit der eingebauten Hardware arbeiten. Deshalb beschlossen wir eine eigene externe Hardware zu bauen. Aber schon bei der Planung traten die ersten Probleme auf. Über welchen Eingang sollen wir gehen? Nach einigen Recherchen fanden wir heraus, dass Apple den Zugriff auf den Dock-Eingang sowie auf Bluetooth nur großen Firmen erlaubt. Die Lösung brachte uns der Line-In-Eingang. Dieser ist von Apple freigegeben. Um aber den hochempfindlichen Line-In-Eingang nicht zu gefährden, entschlossen wir uns zum Bau eines Akustikkopplers. D.h., der Sensor (S) wandelt die zu messende physikalische Größe über eine Elektronik (E) in eine Wechselspannung um. Der Spannung-Frequenz-Wandler (UFW) wandelt die Höhe der Wechselspannung in eine Frequenz um. D.h. je größer die Amplitude der Wechselspannung, desto höher ist die Frequenz. Diese Frequenz wird über einen Lautsprecher in ein akustische Signal umgewandelt, welches dann über ein externes Mikrofon in den iPod eingegeben wird.



4.16 Akustische Kopplung

#### 4.4 Spannungs-Frequenz-Wandler

Unser Spannungs-Frequenz-Wandler besteht aus einem Mikrocontroller (ATMEGA8). Dadurch haben wir die Möglichkeit ein Programm in C zu schreiben, welches die anliegende Spannung mit einer Referenzspannung vergleicht und auf einem dafür ausgelegten Pin eine der Spannung entsprechende Frequenz ausgibt. Das Programm wird über eine ISP („In-System-Programming“) Verbindung

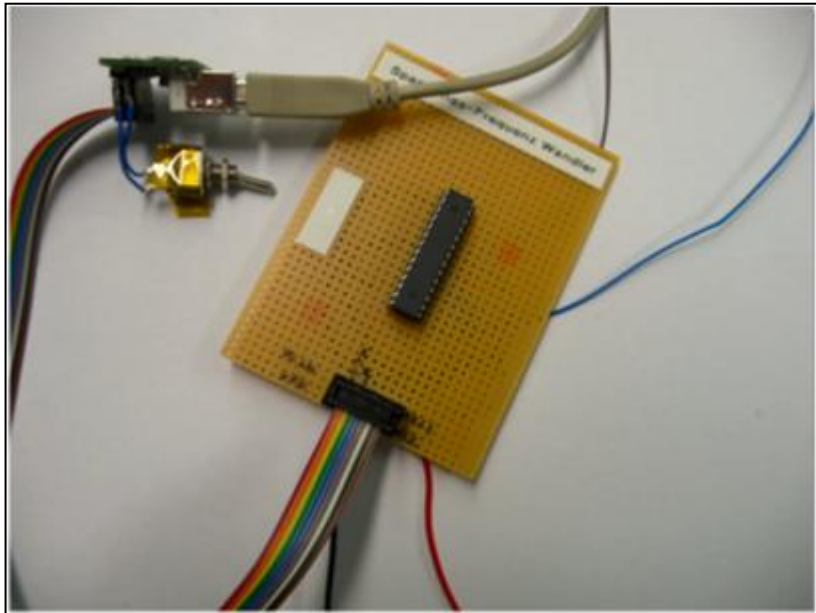


Abbildung 4.17 Spannungs-Frequenz-Wandler

aus dem Mikrocontroller geflasht. Sobald man den Chip mit Spannung versorgt, beginnt er automatisch das anliegende analoge Spannungssignal im internen Analog-Digital-Wandler in einen 16-Bit-Wert zu wandeln. Die Spannung lässt sich also mit einer sehr genauen Auflösung von  $1/65536$  der Referenzspannung messen. Dieser digitale Wert wird einer selbst bestimmbaren Frequenz zugeordnet und ausgegeben.

## 5 Eigene Software

### 5.1 Einführung in das iOS SDK

Das Software Development Kit für das Betriebssystem iOS, welches von iPhone, iPod touch, iPad & Co. genutzt wird. Es wird von Apple angeboten und ermöglicht das Programmieren von eigenen Apps. Mithilfe von Objective C, einer Erweiterung der Programmiersprache C um Sprachmittel für objektorientiertes Programmieren, lassen sich eigene Apps entwickeln. Das SDK setzt



Abbildung 5.1 Das iOS SDK

sich zusammen aus Xcode, mit welchem man den Code bzw. Programmtext für die Apps schreiben kann, dem Interface Builder, ein Programm zur Gestaltung der View der Apps, und dem iPhone Simulator (siehe Abb. 5.1). Diese Programme laufen ausschließlich auf Mac.

### 5.2 Eigenes Programm iDatenlogger

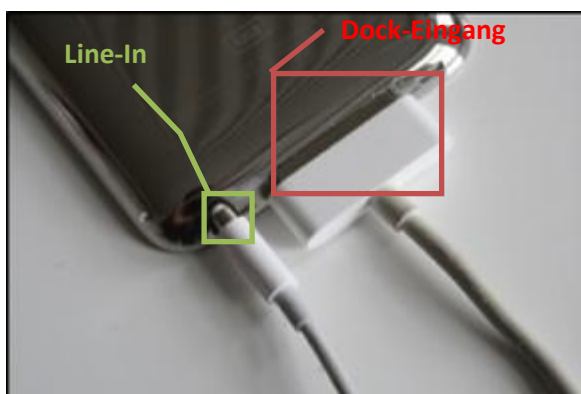


Abbildung 5.2 Die Schnittstellen des iPod Touch

#### Programmidee

Das Programm, welches wir benötigen, muss eine Schnittstelle des iPods nutzen können um Daten bzw. Messwerte in das Multimedia Gerät hinein zu bekommen. Als erstes denkt man da an den Dock-Stecker-Slot, welcher auch genutzt wird um den iPod über USB mit dem PC oder Mac zu verbinden um Daten, z.B. in Form von Musik zu übertragen. Wie oben schon oben erwähnt ist es nur großen Firmen und nicht zuletzt Apple selbst vorbehalten diesen Eingang zu nutzen. Da Apple auch

# Jugend forscht 2011

## Der iPod touch – Eine Revolution im Unterricht

Einschränkungen in der Nutzung von Bluetooth gibt, blieb uns nur noch der Line-In. Diese Schnittstelle wird üblicherweise genutzt, um Kopfhörer anzuschließen. Da fragt man sich, wie man dort Daten hineinbekommen soll, wo die Musik doch nur herausgegeben wird. Doch der Line-In verfügt auch über die Funktion ein externes Mikrofon anzuschließen, um so Audiosignale an den iPod zu senden. Somit standen die Grundanforderungen an unser Programm fest: Es muss Audiosignale aufzeichnen können. Anschließend soll es diese Daten auswerten und „entschlüsselt“, d.h., als Zahl oder in einem Diagramm darstellen können. Dies ist in der Tat keine leichte Aufgabe.

### **Umsetzung**

Um mit dem SDK programmieren zu können, muss man Objective C können. Da unser Projektthema schon seit länger feststand, hatten wir uns bereits mit Objective C und dem SDK auseinander gesetzt. Ein guter Einstieg war das Buch „Entwickeln mit dem iPhone SDK“. Wie schon erwähnt benötigt man ein Mac Book um zum das iOS SDK nutzen zu können. Wir entschieden uns nach einigen Überlegungen zum Kauf eines Mac Book 13“. Das iOS SDK ließ sich von der Apple-Homepage herunterladen und eine Entwicklergebühr von 90\$ im Jahr ermöglicht es die Apps auf dem iPod zu testen und zudem im App-Store zu vertreiben.

### **Eigene App „iDatenLogger“**

Nachdem nun die Zielsetzung feststand, den iPod touch über den Line-In als Datenlogger zu nutzen, gingen wir davon aus, dass die anzuschließenden Sensoren physikalische Messgrößen stets in eine Wechselspannung umwandeln sollten, da der Line-In das Aufzeichnen von Gleichspannungen nicht zulässt. Auf die aufgezeichnete Wechselspannung können wir wie folgt zugreifen: Wir erfassen die Amplituden (Lautstärke) des Wechselspannungssignals oder seine Frequenz. Da nicht jeder angeschlossene iPod und jedes externe Mikrofon gleich empfindlich ist, eignet sich die Erfassung der Lautstärke für eine exakte Messwertwiedergabe nicht. Wesentlich besser geeignet ist dagegen eine Frequenzanalyse.

Da es kein Programm gibt, welches Langzeitaufnahmen ermöglicht und somit als Datenlogger fungieren könnte, mussten wir dies selber schreiben.

Zu allererst musste unser eigenes Programm über den Line-In aufnehmen können. Die Klasse AVAudioRecorder ermöglicht es, ein Recorder-Objekt mit einer Ziel-URL, die das Ziel der Aufnahme sein soll und einige Einstellungen zu erzeugen, und mit der Aufnahme der Audiodatei zu beginnen. Jedoch ist es mit dieser Klasse, soweit ich das verstanden hatte, nur möglich mit dem internen Mikrofon aufzunehmen. Um für unser Projekt hilfreich zu sein, muss das Programm jedoch über eine an den Line-In angeschlossene Hardware aufnehmen können. Andere Apps wie z.B. „SignalScopePro“ erkennt sofort wenn ein externes Mikrofon angeschlossen wird und nimmt automatisch von diesem auf. Somit musste es möglich sein. Nach einigen Recherchen dann die Erleichterung: Wird die Klasse AVAudioRecorder verwendet, so schaltet der iPod automatisch auf Hardware am Line-In um ohne dass wir groß umprogrammieren mussten. Die Daten können jetzt mithilfe des AVAudioRecorders als Audiodatei aufgenommen und als PCM- oder Encoded-Format gespeichert werden.

Jugend  forscht 2011  
Der iPod touch – Eine Revolution im Unterricht

Nun musste auf die Lautstärkewerte, welche aufgenommen werden, zugegriffen werden können, um dann an die Frequenz zu kommen. Nach einer erfolglosen Recherche stießen wir auf einen Sample Code von der Apple-Homepage, welcher die Lautstärke von den Aufnahmen als sich füllende Balken darstellt. Es dauerte einige Zeit, bis wir die für uns wichtigen Code-Abschnitte gefunden und verstanden hatten. Ein Paar Codestellen übernahmen wir aus dem Sample Code und passten sie an unsere Bedürfnisse an. Mit „printf()“ ließen wir uns die Werte in die Konsole schreiben und konnten sie somit in Excel importieren um herauszufinden, wie die Lautstärkewerte gemessen werden. Die Sampling Rate der Aufnahme liegt bei 44100 Hz, d.h. es werden 44100 Lautstärkewerte pro Sekunde erfasst und geschrieben. In Excel ergaben die Werte einer Aufnahme von einer Stimmgabel mit 440 Hz dieses Diagramm (siehe Abb. 5.3 und 5.4).

Abbildung 5.3 440 Hz-Aufnahme mit unserer App dargestellt in Excel

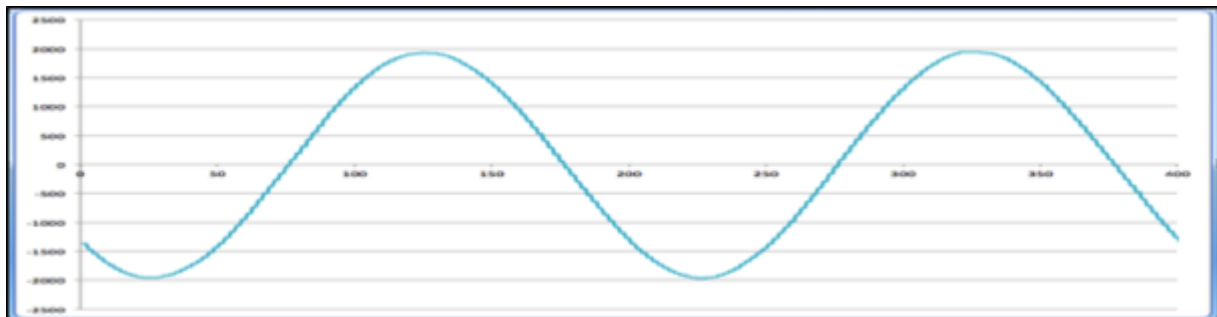
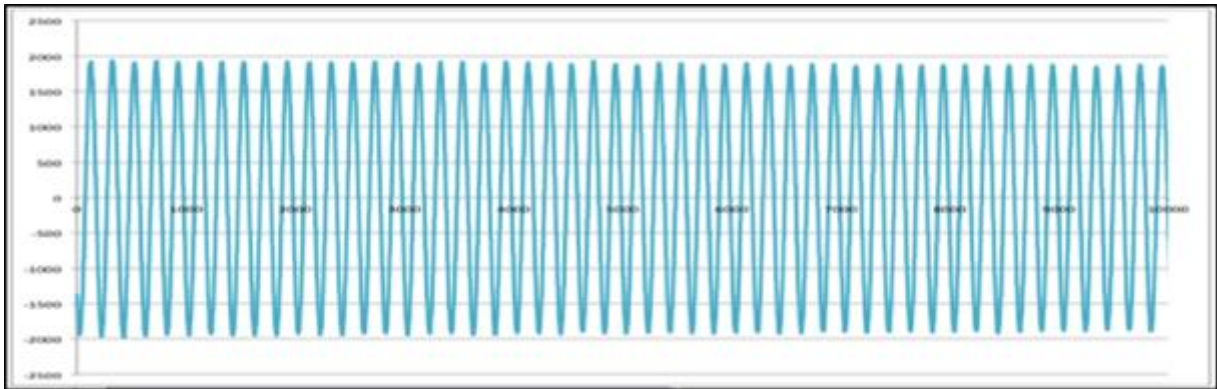


Abbildung 5.4 440 Hz-Aufnahme unserer App dargestellt in Excel (nah)

### Problem: Frequenz errechnen

Die Aufnahmen ermutigten uns die Frequenzen zu errechnen. Die erste Idee dazu war es, die Nulldurchgänge zu zählen und die Anzahl der Durchgänge zu halbieren um an die Frequenz der Aufnahme zu kommen. Um jedes Mal zu zählen, wenn die Lautstärke den Wert Null erreicht, sind die Messwerte nicht angemessen, da sie meist nicht genau den Wert null, sondern um null herum liegen.

Eine weitere Idee, die eigentlich idiotensicher war, war es, immer wenn der Wert vorher positiv war und dann negativ wurde einmal zu Zählen. Dies war mit Hilfe diesen Codes unserer Meinung nach möglich:

Jugend  forscht 2011  
Der iPod touch – Eine Revolution im Unterricht

```
short* audioDaten = (short*)inBuffer->mAudioData;
short pot = 0;
for(int i = 0; i < 44100; i++)
{
    if (
        ((audioDaten[i] > 0 ) and (positivBefore = 0))
    ) {
        pot = pot + 1;
        positivBefore = 1;
    }

    if (
        ((audioDaten[i] < 0 ) and (positivBefore = 1))
    ) {
        pot = pot + 1;
        positivBefore = 0;
    }
}
printf("%i Hz, ", pot);
```

In der ersten Zeile wird eine Variable vom Typ short namens „audioDaten“ erstellt, welche den Wert von „inBuffer->mAudioData“ in Form von short zugewiesen bekommt. Des weiteren wird eine weitere short Variable „pot“ und eine vom Typ double „positivBefore“ welche nur da ist um „ja“ (positivBefore = 1) oder nein (positivBefore = 0) anzugeben erstellt. Der Anfangswert wird auf 1 gesetzt damit sie einen Wert besitzt. Mit der nächsten Zeile wird nur bezweckt, dass alles was in der geschweiften Klammer steht für jeden Wert von „i“ zwischen 0 und 44100 durchgeführt wird. Dort prüft das Programm nun jeden Wert von „audioDaten“ durch. Wenn ein vorheriger Wert negativ war (positivBefore = 0) und der jetzige Wert größer als Null ist, wird „pot“ zu „pot“ + 1, also wenn in „pot“ vorher den Wert von null hatte sind wird er jetzt zu eins etc. Hat das Programm dies für alle Werte getan schreibt es diesen Wert in die Entwickler Konsole mit printf(), da das Programm erst im Simulator getestet werden musste. Die angeblichen Frequenzwerte lagen aber im Bereich von 11 kHz, was natürlich viel zu hoch ist, und sie veränderten sich dazu auch nicht merklich bei Veränderung der Frequenz.

Nach längerer Studie der Messwerte fiel dann auf, dass zwei aufeinanderfolgende Werte immer gleich sind. Nachdem wir die Kurve analysiert hatten, bestätigte sich unser Verdacht, dass über zwei Kanäle aufgenommen wird. Wir mussten uns eine andere Methode überlegen.

Eine zweite Idee, die uns vorher auch schon kam, aber wegen der Aufnahme über 2 Kanäle auch nicht funktionierte, war das Zählen der Peaks, d.h. der höchsten Werte einer Schwingung. Mithilfe des folgenden Codes war es uns dann möglich die Anzahl der Peaks zu zählen und somit die Frequenz zu analysieren. Der entscheidende Unterschied war jedoch, dass wir nur mit jedem zweiten Lautstärke-Wert arbeiteten, da wie oben bereits erwähnt zwei aufeinanderfolgende Lautstärken den selben Wert besaßen. Der folgende Code ist der wichtige Teil der Frequenzanalyse:


  
 Jugend forscht 2011
   
 Der iPod touch – Eine Revolution im Unterricht

```

short* audioDaten = (short*)inBuffer->mAudioData;
short pot = 0;
for(int i = 0; i < 44100; i++)
{
    if ((audioDaten[i] > audioDaten[i - 2]) and
        (audioDaten[i] > audioDaten[i + 2])) {
        pot = pot + 1;
    }
}
printf("%i Hz, ", pot);
  
```

Wie im ersten Codeauszug gibt es auch hier die Variablen „audioDaten“ und „pot“ sowie die for(){} Wiederholung von „i“ jedoch passiert hier in der geschweiften Klammer etwas anderes. Für jeden Messwert wird geprüft, ob er größer ist als der Wert zwei Positionen vor ihm („i“ – 2) und größer als der zwei nach ihm („i“ + 2). Ist dies der Fall, so wird „pot“ wieder zu „pot“ + 1.

Mehrere Experimente mit einer Stimmgabel von 440 Hz und einer mit 1600 Hz wiesen eine hohe Genauigkeit von nur wenigen Prozenten Abweichung auf (siehe Abb. 5.5). Und dies noch ganz ohne Filterung.

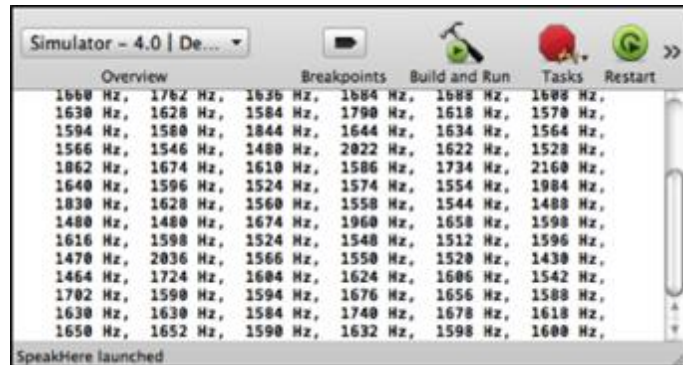


Abbildung 5.5 Frequenzwerte unseres Programms der Aufnahme des Signals einer Stimmgabel mit 1600 Hz



Abbildung 5.6 Die Benutzeroberfläche von iDatenlogger

Wir entwarfen noch ein schönes provisorisches Interface für unsere App (siehe Abb. 5.6), testeten sie im Simulator mit unserem externen Mikrofon wodurch eine noch genauere Aufnahme erfolgte. Mit unserer akustischen Kopplung in einer schalldichten Box wird das Ergebnis natürlich noch genauer. Mit fertiger Hardware ist dies die einzige App auf der Welt welche Spannungen und - mit Erweiterung - weitere Signale messen kann. Ein großer Erfolg für uns.

Während des Programmierens dieser App fielen uns zahlreiche weitere Ideen zur Weiterentwicklung des Programms ein, beispielsweise die grafische Darstellung der gemessenen Signale, für welche wir bereits eine Oberfläche programmiert hatten (siehe Abb. 5.7). Und das Messen mit weiteren Sensoren durch Einbau von Umrechnungsfaktoren wäre auch eine denkbare Modifikation.



Abbildung 5.7 Meine Oberfläche für die grafische

## 6 Ausblick

Wir sind überzeugt mit unserem Konzept den Unterricht revolutionieren zu können. Wir haben einen Einblick in die Vielzahl von käuflichen Apps und ihren Anwendungsbereich im Unterricht gegeben. Wie gesagt ist dies nur ein Teil des Mosaiks, eine Reihe weiterer Applikationen können für den Unterricht verwendet werden.

Unser Projekt für eine eigene Software und Hardware, die im Unterricht verwendet werden kann, hat schon weite Fortschritte erlebt, ist funktionsfähig und kann bis zum Wettbewerb noch weiter ausgearbeitet werden.

Der iPod hat das Potenzial den Unterricht zu revolutionieren.

## 7 Danksagung

An dieser Stelle bedanken wir uns herzlich bei Stefan Hück, für die freundliche Beantwortung von Fragen bezüglich der Hardware, Heiko Bura und Herrn Wagner für die Unterstützung bei der Entwicklung der Software, sowie nicht zuletzt vor allem bei unserem Betreuungslehrer Walter Stein.

## 8 Literatur- und Linkliste

### 8.1 Literaturliste

Bura, Heiko	„Physikversuche mit dem Handy“
Dudney, Bill u.a. (Demmig, Thomas: Deutsche Übersetzung)	„Entwickeln mit dem iPhone SDK“
Sadun, Erica	„Das große iPhone Entwicklerbuch: Rezepte für Anwendungsprogrammierung mit dem iPhone“

### 8.2 Linkliste

Apfeltalk	<a href="http://www.apfeltalk.de/forum/iphone-os-developer-f212.html">http://www.apfeltalk.de/forum/iphone-os-developer-f212.html</a>
Apple Developer	<a href="http://developer.apple.com/">http://developer.apple.com/</a>
iTunes	<a href="http://www.apple.com/de/itunes/">www.apple.com/de/itunes/</a>
Wikipedia	<a href="http://de.wikipedia.org/">de.wikipedia.org/</a>