

# SCHULE konkret



swch.ch

szBulle und Weiterbildung Schweiz

2/19

ICT

Medien und Informatik – konkrete  
Umsetzung im Unterricht  
Sonic Pi – Live Coding im Unterricht  
Kartenspiel online erstellen  
Luft und Boden digital erleben  
Geschichten schreiben von analog  
bis digital



# Sonic Pi – Live Coding im Unterricht

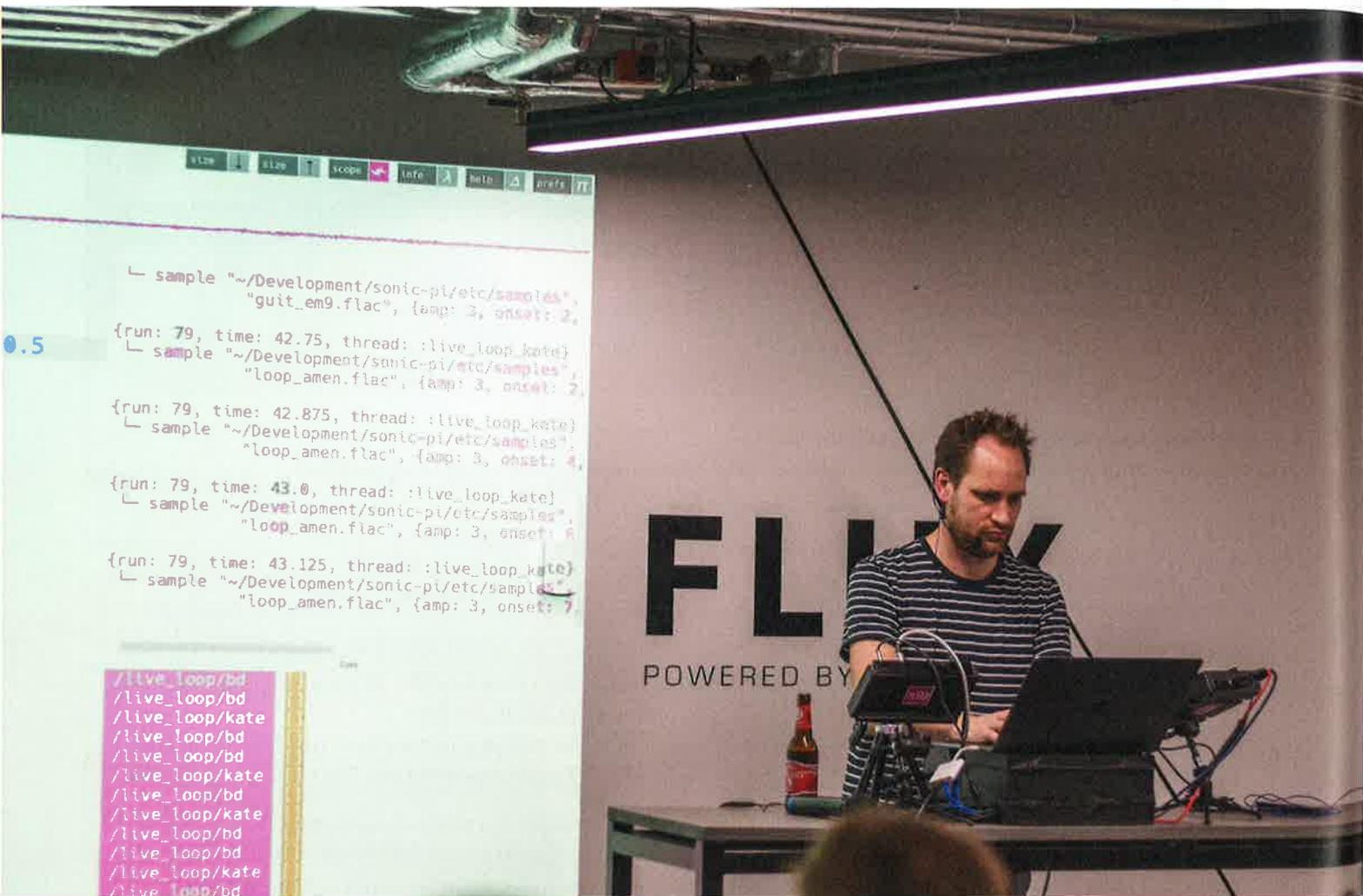
Text: Florence Weber / we

Fotos: Sam Aaron, siehe Nachweise



Die Schülerinnen und Schüler...

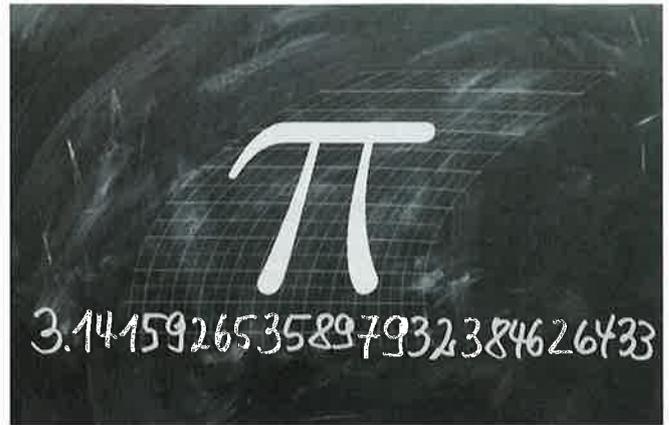
- » können selbstentdeckte Lösungswege für einfache Probleme in Form von korrekten Computerprogrammen mit Schleifen, bedingten Anweisungen und Parametern formulieren.
- » können selbstentwickelte Algorithmen in Form von korrekten Computerprogrammen mit Variablen und Unterprogrammen formulieren.



## Begeisterung für's Coden

Die Bildungslandschaft ist sich in weiten Teilen einig, dass Medien und Informatik schon in der Volksschule ein integraler Bestandteil sein muss. Beispielsweise hat sich dies im neuen Lehrplan 21 der Deutschschweizer Kantone im Modullehrplan Medien und Informatik niedergeschlagen. Hingegen nicht einig ist man sich über «Wie wollen wir Schülerinnen und Schüler denn überhaupt dafür begeistern?». Es scheiden sich die Geister, wenn es um die Vermittlung von Informatikkonzepten und Grundlagen geht:

Von Tabellenkalkulationen über Schildkröten, bis hin zu fahrenden Kleinstrobotern. Sonic Pi beschreitet einen anderen Weg. Das Programm verwandelt geschriebenen Code in Musik und wird so zu einem neuartigen Musikinstrument. Es ermöglicht Schülerinnen und Schülern, ihre eigene Musik und somit ihren eigenen Stil zu programmieren. Das Musikmagazin Rolling Stone beschreibt den Sonic Pi – Auftritt am Moogfest USA, 2016: «... it truly seemed less like a performance and more like an invitation to code your own adventure.»



### Leistungsstark aber simpel

Ein System, welches einen hohen Anspruch an Professionalität hat, ist oftmals schwierig zu erlernen, eignet sich somit nur sehr beschränkt für die Schule. Sonic Pi wurde deshalb in enger Kollaboration mit Lehrpersonen und in unzähligen Unterrichtsstunden mit Schülerinnen und Schülern entwickelt. Einfachheit im Design des Programms und dessen Bedienung war der Hauptanspruch in der Entwicklung. Ein zehnjähriges Kind kann es verstehen und bedienen. Für den Einstieg in die Sonic Pi-Welt braucht es lediglich zweier sehr einfacher Anweisungen. Die **play**-Anweisung, welche verschiedene Noten spielt und die **sleep**-Anweisung, welche die Wartezeit bis zur nächsten Note definiert.

Mit **sample** wird die nächste Sonic Pi-Ebene erschlossen, welche aufgenommene Sounddateien (Samples) abspielen kann. Dies können ganze Drumloops oder einfach auch Geräusche sein. Natürlich können diese Sounddateien auch während dem Abspielen manipuliert werden. Zusätzlich stehen der Coderin oder dem Coder auch Studio-Effekte wie zum Beispiel Hall oder Verzerrung zur Verfügung. Für eine Gitarre mit Hall braucht es drei Zeilen Code:

```
with_fx :reverb do
  sample :guit_harmonics
end
```

### Modullehrplan: Medien und Informatik

Die allermeisten Informatik-Kompetenzen des Modullehrplans Medien und Informatik sowie weitere, auch komplexere, Informatikkonzepte, lassen sich mit Sonic Pi aufzeigen. Die Stärke von Sonic Pi liegt im Gegensatz zu anderen Umgebungen im auditiven Zugang, welcher stark an Empfindungen geknüpft ist. Eine eindeutige Zuteilung von Sonic Pi in Fächer wie Musik oder Informatik und Mathematik ist von vornherein ausgeschlossen, da die direkte, auditive Rückmeldung sowie danach die Gleichwertigkeit des Empfundenen mit dem geschriebenen Code ganz im Vordergrund steht, was wiederum motivierend wirken kann. Nebst den bereits erwähnten Anweisungen können Schleifen, Listen, Variablen, bedingte Anweisungen sowie Funktionen einfach programmiert werden.

### Sonic Pi in der Schulklasse einsetzen

Musik ist Teil der Lebenswelt der Jugendlichen. Sonic Pi eignet sich also hervorragend als Einführung in eine Programmiersprache und vermittelt die wichtigsten Grundelemente des Programmierens. Das Programm kann unter [sonic-pi.net](http://sonic-pi.net) für OS X, Windows und Linux kostenlos heruntergeladen werden. Nebst den integrierten Tutorials existieren weitere, frei downloadbare Anleitungen wie zum Beispiel «Code music with Sonic Pi – Sonic Pi Essentials» von Sam Aaron, dem Entwickler von Sonic Pi.

Für weitere Informationen: [sonicpi.weebly.com](http://sonicpi.weebly.com)

### Erfahrungsbericht – für Lehrpersonen

Im Rahmen einer Projektwoche führte der Lehrer Reto Morgenthaler des Oberstufenzentrums Täuffelen im Mai 2018 eine Gruppe von 15 Knaben in Sonic Pi ein. Das Ziel: Am Ende der Woche vor Publikum live Musik zu programmieren. Das Projekt wurde vor Ort unterstützt von Christian Dietz und Nico Steinbach von der PHBern.

Die Woche mit dem abschliessenden «Live Coding» vor Publikum aus Schule und Dorf war ein Erfolg für das Leitungsteam, die Jugendlichen und auch für das Publikum. Hier sind die wichtigsten Erkenntnisse von Reto Morgenthaler, welche hoffentlich für andere Live-Coding-Projekte mit Jugendlichen hilfreich sein können:

- Für alle Jugendlichen wenn möglich einen eigenen Computer zur Verfügung haben.
- Einen kräftigen Verstärker mit Lautsprecher und einem langen Kabel auftreiben, damit all die coolen Sounds genug Volumen erhalten.
- Kurze Inputs geben und die Jugendlichen viel üben lassen.
- Zwischenresultate regelmässig präsentieren und Feedback geben für gegenseitige Inspiration.
- Nicht erwarten, dass die Jugendlichen mit Sonic Pi so arbeiten, wie man es selbst tun würde. Wichtig ist, offen zu sein für ihre eigene Art der Programmnutzung.
- Als Lehrperson idealerweise nicht mehr als sieben Jugendliche betreuen. Für grössere Gruppen sind mehrere Betreuende ideal.



- Falls man als Lehrperson Fragen nicht beantworten kann: Auf die Programm-Tutorial oder die Sonic-Pi-Webseite verweisen. Dort können sich die Schülerinnen und Schüler die Anleitungen ansehen und sie Schritt für Schritt nachmachen.
- Nach den ersten Gehversuchen die Jugendlichen ein Live Coding nach einer Kriterienliste vorbereiten lassen, das sie zuerst erarbeiten und dann üben.  
Beispielsweise:
  - 1 – 3 Minuten
  - Die Aktivität ist am Beamer sichtbar
  - Eigene Elemente verwenden
  - Einen Aufbau planen: Anfang, Hauptteil, Ende
- Die Jugendlichen zu diesen Kriterien regelmässig Feedback geben und ihnen ebenfalls Rückmeldungen zukommen lassen.
- Die Jugendlichen ermutigen, eigene Samples herzustellen. Sound-Samples von Youtube herunterladen und mit Audacity ([www.audacityteam.org](http://www.audacityteam.org)) zuschneiden.
- Die Beamer-Projektion bei der Vorführung so gross wie möglich machen.
- Das Live Coding wenn möglich so organisieren, dass jede und jeder Jugendliche auf dem «Präsentations-Computer» einen eigenen Programm-Puffer zugeteilt hat, in den sie oder er den Code vorgängig kopieren kann.
- Vor dem Auftritt vor Publikum auftreten: Dem Publikum kurz erklären, wieso Programmieren in der Schule wichtig ist. Eine Einführung in die Arbeitsweise beim Live Coding – insbesondere zu den Live Loops – hilft den Zuschauenden, die Aktivität am Beamer zu verstehen.

In der Youtube-Playlist ([tinyurl.com/sonic-oszt18](https://tinyurl.com/sonic-oszt18)) finden sich fünf Beiträge von 15 Jugendlichen. Diese besuchen die 8. oder 9. Klasse. Die Beiträge sind in fünf Halbtagen Arbeit mit Sonic Pi entstanden. Einige der präsentierten Elemente entstanden mit Hilfe der beiden Unterstützer der Pädagogischen Hochschule Bern.

Happy live coding!

Mit bestem Dank an Reto Morgenthaler für den Erfahrungsbericht und die Genehmigung zum Abdruck in dieser Zeitschrift!

### Bei Fragen

Im Forum der Sonic Pi Community ([in-thread.sonic-pi.net](http://in-thread.sonic-pi.net)) kann man sich kostenlos registrieren und die Fragen den Mitgliedern stellen. Antworten sind garantiert! Unter den Mitgliedern befinden sich von neuen schüchternen Sonic Pi Fans bis zu absoluten Sonic Pi Cracks alle, und helfen, wo sie können.

Wenn Sie eine persönliche und individuelle Betreuung wünschen, melden Sie sich bei Florence Weber, Redaktorin SCHULEkonkret. Die Anfragen werden, auch unter der Mithilfe von Christian Dietz vom Bereich Medien und Informatik, Institut für Weiterbildung und Medienbildung IWM an der PHBern, beantwortet.  
Mailadresse: [florence.weber@swch.ch](mailto:florence.weber@swch.ch)

## SONIC PI – Vier praktische Komponenten zum Gelingen des Programmierens

- Ladet SONIC PI am Computer oder am Tablet herunter.
- Lest die folgenden Anweisungen. Es sind vier grundlegende Komponenten, die euch erste Programmierversuche erleichtern.
- Macht die angegebenen Anleitungen nach. Was funktioniert auf Anhieb? Was nicht? Tauscht euch in Gruppen aus.
- Für weitere Anweisungen könnt ihr die Website [sonicpi.weebly.com](http://sonicpi.weebly.com) checken.
- Lernt das Programm kennen und probiert aus.
- Stellt einen eigenen Beitrag (alleine oder in kleinen Gruppen) zusammen und spielt ihn der ganzen Klasse vor. Optional: Mit Beamer-Übertragung!

**1. Schleifen:** Mittels den zwei einfachen Anweisungen **play** und **sleep** lassen sich schon etliche bekannte aber auch selbst erfundene Melodien abbilden. Damit nicht meterlanger Code geschrieben werden muss, bietet Sonic Pi eine einfach verständliche Schleife, analog den Wiederholungszeichen in der musikalischen Schreibweise, an.

```
2.times do
  play 60
  sleep 1
  play 62
  sleep 1
  play 64
  sleep 1
  play 60
  sleep 1
end
```

Alles, was zwischen **do** und **end** steht, wird wiederholt. Im Beispiel oben sind es die ersten zwei Takte von Bruder Jakob. Natürlich kann zwischen **do** und **end** auch anderer Code sowie mehrere Schleifen hintereinandergeschrieben werden:

```
5.times do
  sample :bd_haus
  sleep 0.5
end

3.times do
  sample :drum_cymbal_open
  sleep 1
end
```

**2. Listen:** Das Beispiel von Bruder Jakob lässt sich auch mit einem Ringbuffer (Ring-Liste) darstellen:

```
ton = 30
60.times do
  play ton
  sleep 0.06125
  ton = ton + 1
end
```

Bei jedem Durchgang in der Schleife wird die nächste Zahl der Klammer zurückgegeben und somit abgespielt. **tick** zählt einfach bei jedem Aufruf eins weiter. Beim fünften Aufruf wird wieder die erste Zahl (60) ausgegeben; deshalb Ring-Liste.

**3. Variablen:** Anstelle einer Liste mit definierten Werten kann der Computer natürlich auch die Zahlen selber berechnen. Dies geschieht am besten mit einer frei zu definierenden Variable. Im Beispiel unten wird diese **ton** genannt und am Ende der Schleife jeweils um eins erhöht.

```
rythmus = (ring 1, 0, 1, 0, 1, 0, 1, 1)
32.times do
  sample :bd_haus, amp: rythmus.tick
  sleep 0.5
end
```

Die Kombination von Schleifen, Listen und Variablen erlaubt auch, rhythmische Muster darzustellen:

```
loop do
  if one_in 2
    sample :drum_cowbell
  else
    sample :drum_snare_hard
  end
  sleep 0.125
end
```

Die Lautstärke des Pauken-Samples **:bd\_haus** wird bei jedem Durchgang durch die Variable **rythmus** definiert. **amp:** steht für Verstärker (amplifier), was bei **0** einer musikalischen Pause entspricht.

**4. Bedingte Anweisungen:** Oftmals will man etwas nur unter bestimmten Umständen hörbar machen, oder dies auch dem Zufall überlassen.

```
8.times do
  play (ring 60, 62, 64, 60).tick
  sleep 1
end
```

**loop** ist die Endlosschleife, welche erst mit dem Klicken des Stop Buttons beendet wird. In dieser Endlosschleife wird mit **one\_in 2** mit einer 50 zu 50 prozentigen Wahrscheinlichkeit entweder der Sample **:drum\_cowbell** oder der Sample **:drum\_snare\_hard** abgespielt.



Sonic Pi

Die Anweisungen stammen aus dem Artikel «Sonic Pi – Live Coding in der Bildung» von Sam Aaron. Der Originalartikel ist hier zu finden: [www.sonic-pi.net](http://www.sonic-pi.net). Aus dem Englischen übersetzt von Christian Dietz, Nico Steinbach und Nando Stöcklin, 2017. Besten Dank für die Genehmigung zum Abdruck!