

Mit dem PicoBoard ein berührungsloses Musikinstrument (Theremin) programmieren

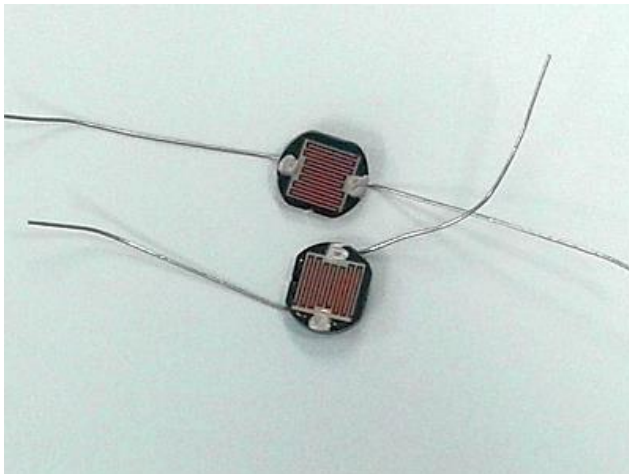
Ein Theremin ist elektronisches Musikinstrument, das berührungslos gespielt werden kann. Dabei beeinflusst die elektrische Kapazität des menschlichen Körpers das elektromagnetische Feld des Musikinstrumentes.

Alternativ kann man das auch mit Lichtsensoren durchführen. Dazu sollten zwei Lichtsensoren auf einem Stück Karton in einem Winkel von 90 Grad angebracht werden, damit die Lichtintensität unabhängig voneinander gemessen werden kann.

Auf einem PicoBoard befindet sich jedoch nur ein Lichtsensor.

Da das PicoBoard aber mit 4 Anschlussklemmen zur Messung von Widerständen versehen ist, können über diese die Lichtintensität gemessen werden.

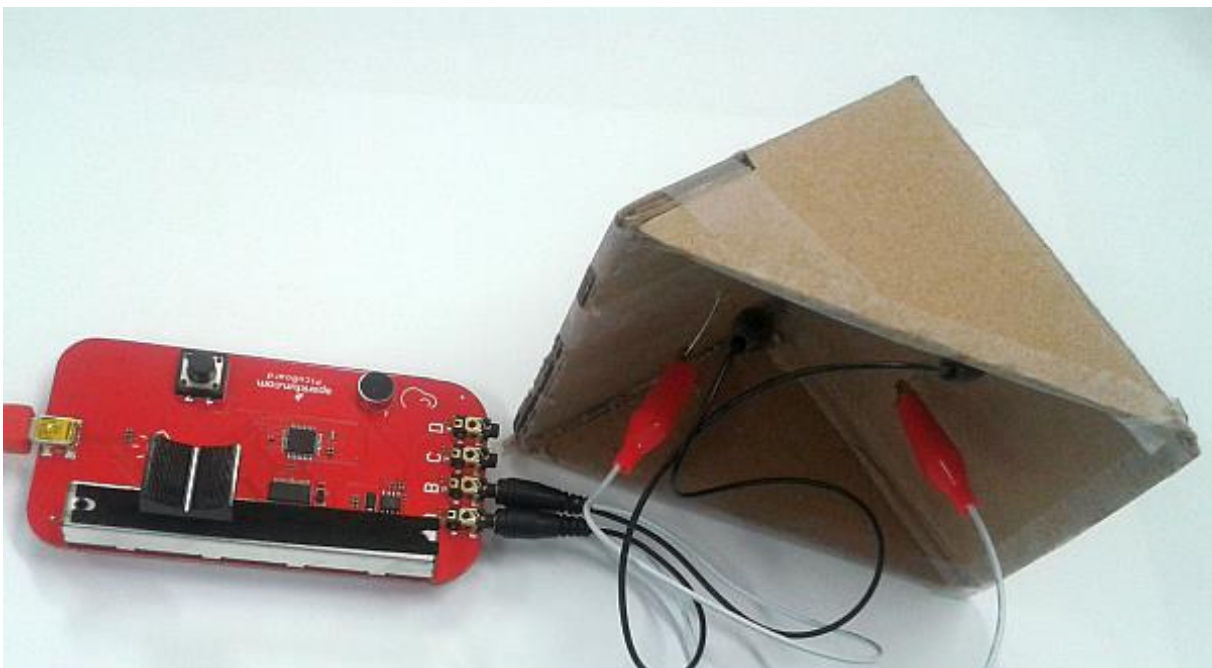
Dazu benötigt man zwei Fotowiderstände, die im Elektronikhandel ab ca. 1 Euro erworben werden können. Als vorteilhaft erweisen sich solche mit einem Durchmesser von ca. 1 cm, da diese dann auch für andere Projekte verwendet werden können.



Fotowiderstände: ca. 1 cm Durchmesser



Fotowiderstände außen auf dem Karton



Anschluss der Widerstandsklemmen an das PicoBoard

Scratch-Programmierung:

Wir verwenden hier die Scratch-Version 1.4, die auch auf dem Raspberry Pi vorinstalliert ist.

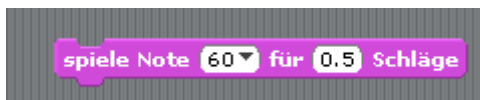


Zuerst setzen wir die Lautstärke auf 100 % und wählen das Musikinstrument Xylophon aus (Nr. 14).

ACHTUNG: Bei Verwendung der Scratch Version 2.0 ändert sich die Nummerierung der Musikinstrumente.

Unser Ziel ist es, einen Ton endlos abzuspielen, dessen Tonhöhe von der Lichtstärke abhängt, die auf den ersten Fotowiderstand trifft und die Länge des Tones soll von der Lichtstärke abhängen, die auf den zweiten Fotowiderstand trifft.

Wir wählen dazu folgenden Block aus:



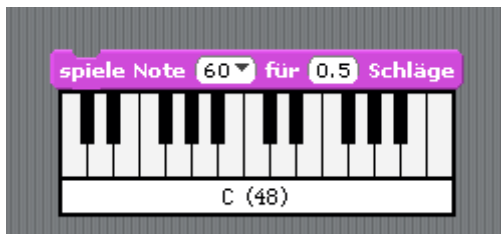
Die voreingestellten Werte ersetzen wir durch die Werte von 2 Variablen, die wir zuerst definieren.

Wir nennen sie „Tonhoehe“ und „Tonlaenge“.



In Scratch 1.4 haben wir 25 Noten zur Verfügung. Da die Sensoren Werte zwischen 0 und 100 angeben, müssen wir das anpassen.

Wir dividieren daher den Messwert durch 4 und addieren dazu den Wert der ersten Note, das ist 48 für die Note C.



Mit Hilfe von Operatoren (grüne Blöcke) kommen wir zu diesem Wert:

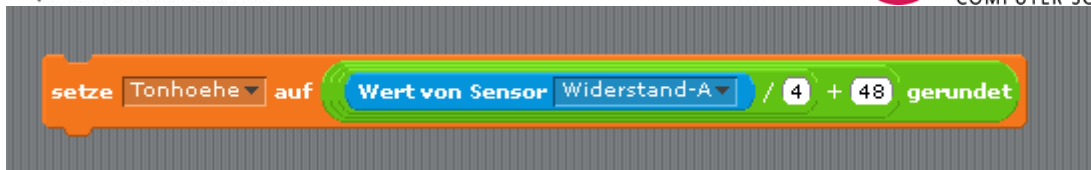


Diesen runden wir noch:



Danach weisen wir diesen Wert der Variablen „Tonhoehe“ zu:





Analog gehen wir für die Tonlänge vor.

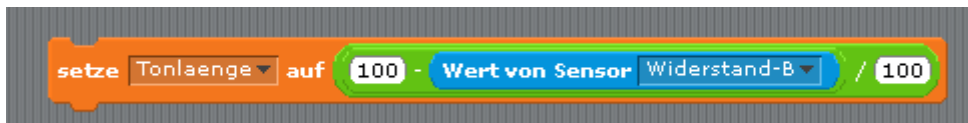
Damit aber der Ton bei weniger Licht kürzer gespielt wird, müssen wir den Sensorwert von 100 abziehen:



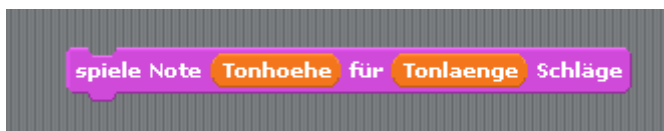
Jetzt könnte man diesen Wert noch durch 100 dividieren, um einen größeren Spielraum zu bekommen:



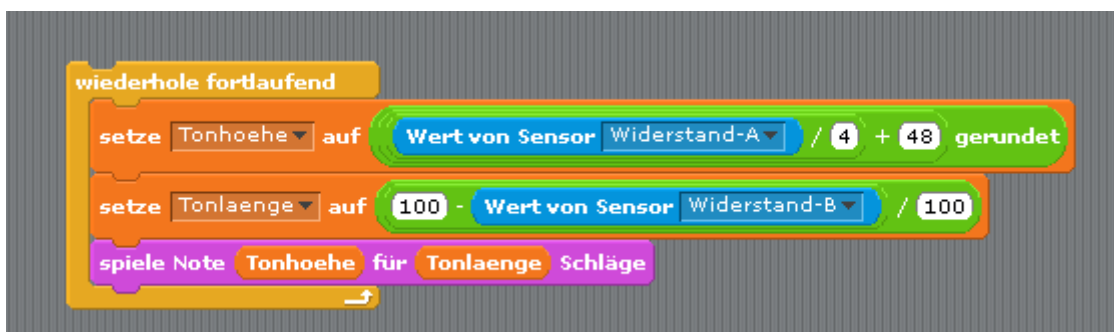
Dieser Wert wird jetzt der Variablen „Tonlaenge“ zugeordnet:



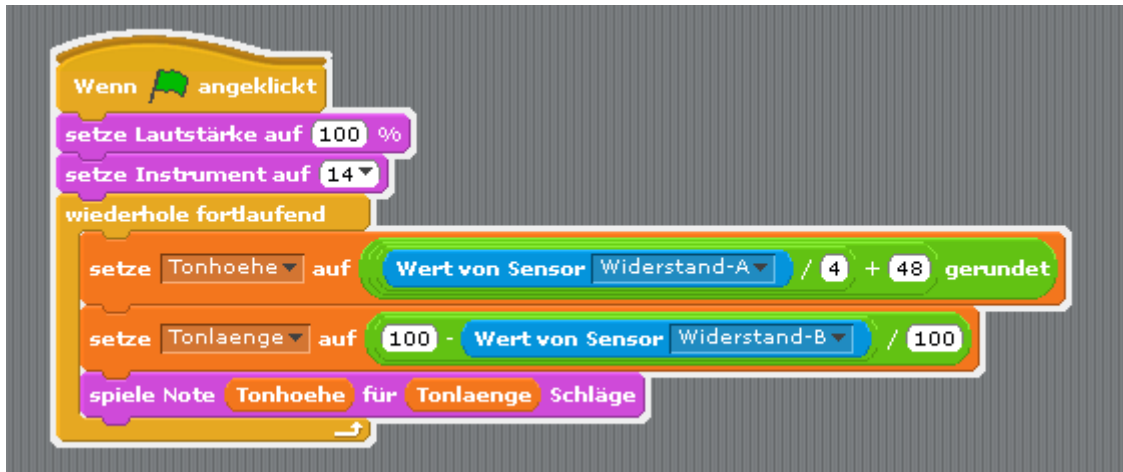
Nun können wir die beiden Variablen in unseren Block „Spiele Note ...“ einfügen:



Sensoren müssen endlos abgefragt werden, daher verwendet wird die Schleife „wiederhole fortlaufend“:



Damit ist unser Programm fertig:



Nun könnten wir weiter experimentieren. Wie klingt das mit anderen Instrumenten? Welche Instrumente sind dafür gut geeignet, welche schlecht?

Wie könnten wir die Empfindlichkeit der Fotowiderstände an die aktuellen Lichtverhältnisse anpassen?

Was passiert beim Abspielen im Dunklen unter der Verwendung einer Taschenlampe?

Dieses Beispiel entstand nach einer Idee der Warwick Volunteers von der University of Warwick:

<http://www2.warwick.ac.uk/fac/sci/wmg/about/outreach/resources>



