# Projekt: Druckgesteuertes Theremin

## Force Sensing Resistor

Als Input für unser „*Druckgesteuertes Theremin***“** benutzen wir zwei „Force Sensing Resistors“ (FSR) oder auf Deutsch Kraftdrucksensoren. Diese Sensoren sind Grundsätzlich nichts anderes als Widerstände, die bei Druck bzw. Krafteinwirkung ihren Widerstand ändern.

An die Sensoren wird eine Spannung angelegt und durch die Änderung des Widerstandes, kann man eine Spannungsschwankung messen. Mit diesen Werten werden später die Frequenzen und Lautstärken der MIDI-Noten ermittelt.

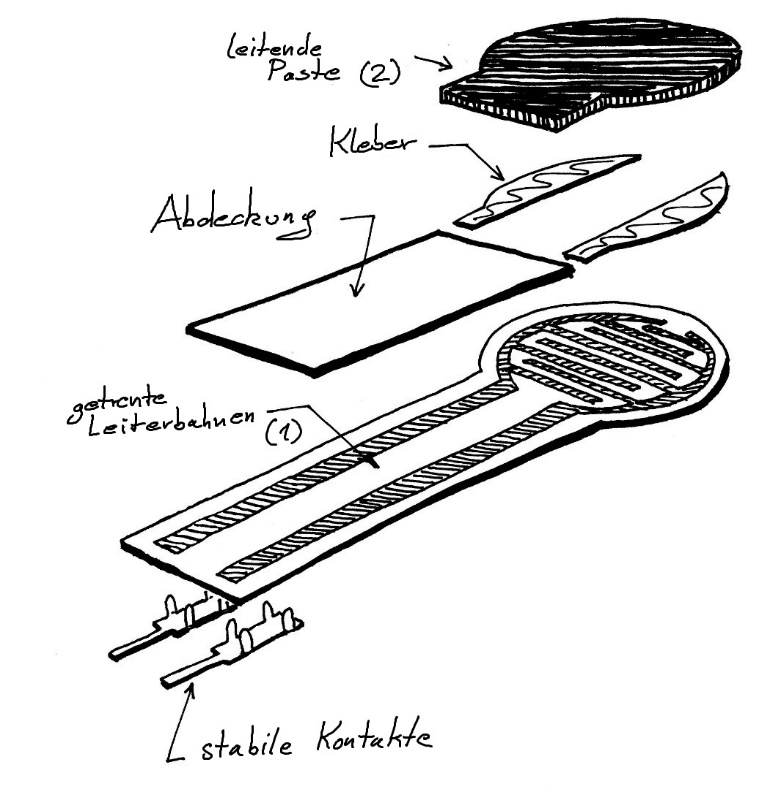
Die Kraftdrucksensoren sind grundlegend aus zwei Folien aufgebaut (siehe Abbildung 1). 

Abbildung : Schichten eines Kraftdrucksensors

Auf der ersten Folie befinden sich zwei gitterförmige Leiterbahnen, welche miteinander keinen Kontakt haben. Die zwei Bahnen haben jeweils am anderen Ende einen Anschluss, mit dem man den Sensor an das Steckbrett anschließen kann.

Auf der zweiten Folie befindet sich eine graphithaltige, halbleitende Paste. Diese Folie ist auf die andere Folie geklebt.

Wenn nun Kraft auf den FSR ausgeübt wird, verbindet die Graphitpaste die zwei elektrisch leitenden Gitter. Je mehr Druck nun ausgeübt wird, desto besser sind die Gitter miteinander verbunden und desto geringer wird der Widerstand des Sensors.

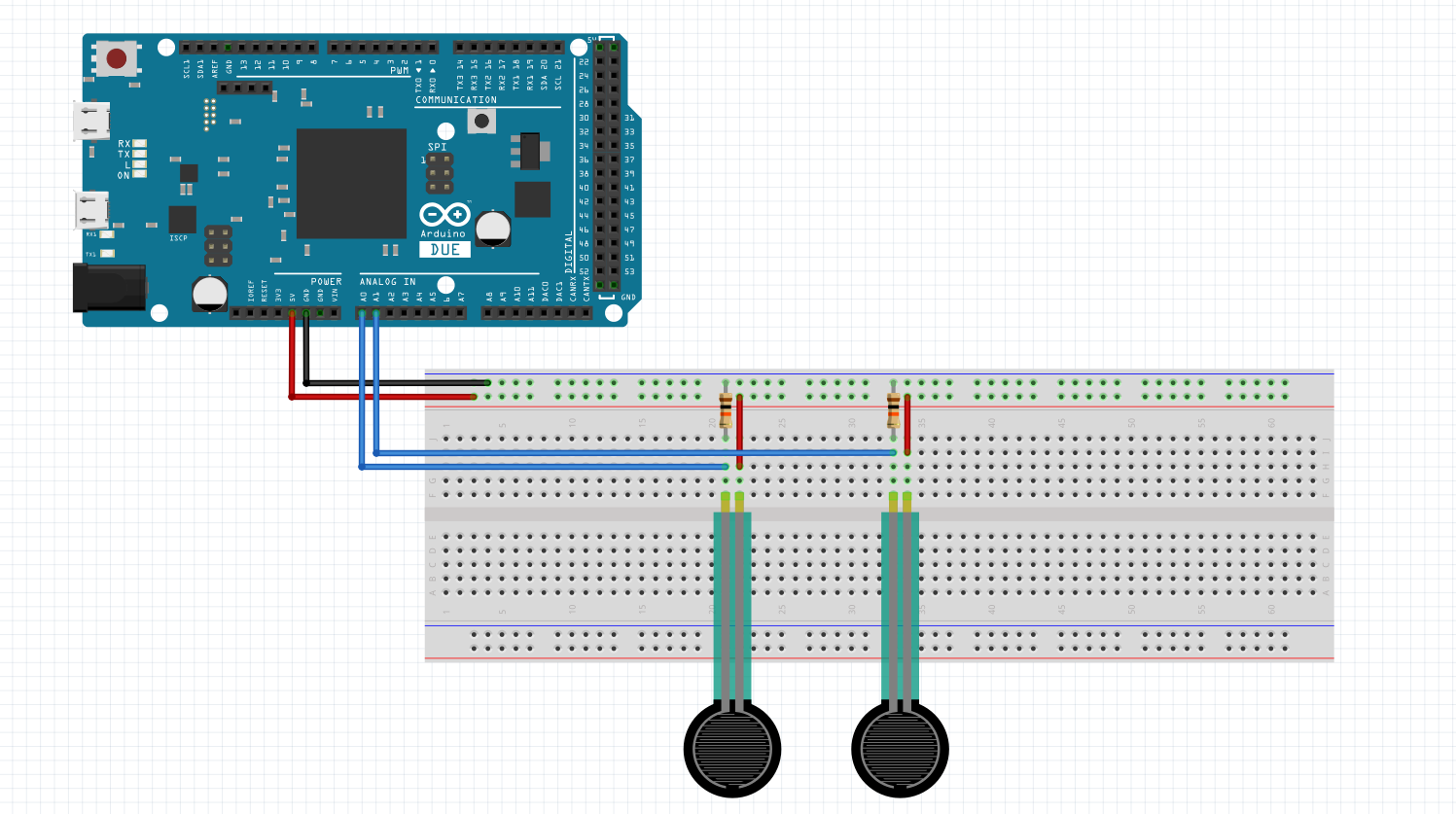
****

Abbildung 2

# Arduino

Mit Hilfe des Arduino wird nun eine Spannung angelegt. Wenn sich der Widerstand des Sensors ändert, kann der Arduino über einen analogen Pin die Spannungsänderung messen (siehe Abbildung 2).

Die meisten Arduino Boards können an ihren Analogen Eingängen lediglich Werte bis zu 10 Bit Auflösung ablesen. Das Arduino Due Board dagegen kann bis zu 12 Bit Werte lesen, dadurch wird eine vier Mal so hohe Auflösung erreicht.

## Planänderung

Eigentlich wollten wir direkt auf dem Arduino Midi Events erzeugen und ein natives MIDI-USB-Endgerät imitieren. Dies wäre auch in der Theorie möglich gewesen, jedoch hätten wir dazu die Firmware des ATmega16U2 Chips überschreiben müssen. Das ist zwar eigentlich überhaupt kein Problem, solange denn die nötige Hardware dafür vorhanden ist. Um die Firmware des USB Controllers zu überschreiben wird leider ein zweites Arduino Board benötigt.   
Von der Hardware ganz abgesehen, gibt es dabei auch noch ein anders Problem: Nach dem Ändern der Software des ATmega16U2 Chips lässt sich der Arduino nicht mehr wie gewohnt neu programmieren, denn genau dafür war die Firmware zuständig, die wir überschrieben haben.

Als Alternativlösung werden die Daten, nachdem sie ausgelesen wurden, als reine Textdaten über die USB Verbindung an einen angeschlossen PC gesendet. Hier müssen sie von einem speziellen Programm verarbeitet werden, dass dann an Stelle des Arduinos MIDI Signale erzeugt und an ein beliebiges MIDI Output Gerät sendet.