

*** Dieses Dokument wird noch bearbeitet und ist noch nicht vollständig. Stand: April 2019***

1 OpenSCAD

Die Programmierung ist stark an gängige Programmiersprachen wie Java oder C++ angelehnt. Bei diesen Sprachen ist ein Befehl immer nach dem Prinzip

```
Objekt.Methode(Parameter);
```

aufgebaut.

Da man in OpenSCAD immer nur ein Objekt hat, nämlich das zu druckende Bauteil, verzichtet man auf die Objektbezeichnung und schreibt immer nur

```
Methode(Parameter);
```

Die Zeichen `()` gehören immer zu einer Methode, auch wenn es keinen Parameter gibt. Genauso wird mit dem `;` gezeigt, dass der Befehl beendet ist. Sollte sich eine Methode über mehrere andere Methoden erstrecken, wird anstatt des `;` die `{}`-Klammern verwendet. In diesen Klammern befinden sich die neuen Methoden.

Für die Konstruktion mit OpenSCAD ist es wichtig, eine modulare Programmstruktur zu entwickeln. Mit *module* gibt man dem Programmteil einen Namen und kann ihn mehrfach wiederverwenden. Bei den gezeigten Beispielen sind das die Motorhalterungen und die Stützen.

2 Befehlsübersicht zu OpenSCAD

Bei geometrischen Objekten wie z.B. Würfel (*cube*) oder Zylinder (*cylinder*) gilt die Bezeichnung `([x,y,z])` als Parameter. Dabei sind es Längenangaben in *mm*. Es können auch Dezimalzahlen eingegeben werden, diese werden mit einem `.` und nicht mit einem `,` angegeben. Das `,` wird zur Trennung der Koordinaten verwendet. *x,y,z* sind die Achsen im Koordinatensystem, welches unten links zu sehen ist.

Befehl	Bedeutung
<code>module</code>	Der Befehl, gefolgt von einem Namen erstellt ein Modul, welches immer wieder aufgerufen werden kann. Bei NXC heißt es <i>sub</i> und bei Arduino <i>void</i> .
<code>cube([x,y,z])</code>	Es wird ein Würfel gezeichnet.
<code>cylinder([h=20, r=6])</code>	Es wird ein Zylinder mit der Höhe <i>h=20mm</i> und dem Radius <i>r=6mm</i> gezeichnet.
<code>translate([x,y,z])</code>	Das zu zeichnende Objekt beginnt an der Stelle <i>(x,y,z)</i> .
<code>rotate([0,0,45])</code>	Das zu zeichnende Objekt wird um 45° mit der <i>z</i> -Achse als Drehachse gedreht.
<code>difference()</code>	Das erste gezeichnete Objekt wird erstellt, alle folgenden Objekte werden aus dem ersten herausgeschnitten.
<code>//</code>	Kommentar, der im Programm nicht weiter beachtet wird.
<code>\$fn=50;</code>	Feinheit der Rundungen.