Die Grundlegende Funktionsweise unseres Programms ist sehr einfach. Wir berechnen über mehrere Formeln die Position einer LED im Bild, lesen die Farbe des entsprechenden Pixels aus und ändern die Farbe der dazu passenden LED entsprechend.

Zuerst wird die aktuelle Systemzeit mithilfe der Funktion time(), die die aktuelle Systemzeit ausgibt, die Anfangszeit() ermittelt. Wenn eine Umdrehung beendet ist, also wenn der Magnetschalter ausschlägt wird mithilfe der Endzeit () die Umlaufzeit berechnet:

Mit dieser rechnen wir nun die momentane Winkelgeschwindigkeit() aus:

Sobald der Magnetschalter nun nicht mehr auslöst, setzt eine Dauerschleife ein. Bei jeder Ausführung wird hier zuerst der Zeitabschnitt zwischen Anfang der Umdrehung und der momentanen Zeit berechnet () mithilfe der aktuellen Zeit und der Anfangszeit der Umdrehung nach der Formel

berechnet.

Die Winkelgeschwindigkeit kann man auch anders darstellen:

stellt man dies nun, wie oben angegeben um erhält man:

Da das entstehende Bild noch um 90°() gedreht ist subtrahieren wir hier noch diesen Wert.

Nun werden für jede LED die Koordinaten im Bild, die von jener widergespiegelt werden, nach den Grundformeln des Einheitskreises

und bestimmt.

Da wir für jede LED einen anderen Radius haben und dies hier nur für den Radius 1 multiplizieren wir diese noch mit dem Radius:

und

Nun wird das Zentrum des Koordinatensystems zum Mittelpunkt des Bildes verschoben. Dies funktioniert, indem man zur X-Koordinate die Hälfte der Breite und zur Y- Koordinate die Hälfte der Höhe des Bildes addieren:

Da man die einzelnen Pixel nur als ganze Zahlen auslesen kann wird dieses Ergebnis noch gerundet:

Im Programm sieht dies dann so aus:

x = int(round(cos(alpha) \* rad + breite)) *#Berechnung der X-Koordinate*

y = int(round(sin(alpha) \* rad + höhe)) *#Berechnung der Y Koordinate*

sin() und cos() sind Methoden der Python math Bibliothek und berrechnen den Sinus und Cosinus des angegebenen Winkels

Nun wird mit der PI(Python Image) Bibliothek der entsprechende Pixel ausgelesen.

Zuerst wird das Bild als zweidimensionaler Array in den Cache von Python geladen:

from PIL.Image import open

im = open("/home/pi/Desktop/JugendForscht/Stop.png")

pix = im.load()

Als nächstes wird der Array mit den entsprechenden Pixeln ausgelesen:

r,g,b, \_ = pix[x, y] *#auslesen eines Pixels*

Das berrechnen und Auslesen der Pixel

Die Variablen r, g, b werden nun durch die Methode bildAuslesen():

streifen.setPixelColor(Led\_NR, bildAuslesen(alpha, r))