|  |
| --- |
| import RPi.GPIO as GPIO |
|  |  | import time |
|  |  |  |
|  |  | class buzzer: |
|  |  | def run(self,pin,delay): |
|  |  | GPIO.setwarnings(False) |
|  |  | GPIO.setmode(GPIO.BOARD) |
|  |  | GPIO.setup(pin,GPIO.OUT) |
|  |  | GPIO.setwarnings(False) |
|  |  | for i in range(0,2): |
|  |  | time.sleep(1) |
|  |  | GPIO.output(pin,0) |
|  |  | time.sleep(delay) |
|  |  | GPIO.output(pin,1) |
|  |  | time.sleep(delay) |
|  |  | GPIO.output(pin,0) |
|  |  |  |
|  |  | def controllRunOff(self,pin): |
|  |  | GPIO.setwarnings(False) |
|  |  | GPIO.setmode(GPIO.BOARD) |
|  |  | GPIO.setup(pin,GPIO.OUT) |
|  |  | GPIO.setwarnings(False) |
|  |  | GPIO.output(pin,0) |
|  |  |  |
|  |  | def controllRunOn(self,pin): |
|  |  | GPIO.setwarnings(False) |
|  |  | GPIO.setmode(GPIO.BOARD) |
|  |  | GPIO.setup(pin,GPIO.OUT) |
|  |  | GPIO.setwarnings(False) |
|  |  | GPIO.output(pin,1) |
|  |  |  |
|  |  |  |
|  |  | #l = buzzer() |
|  |  | #l.run(36,0.5) |