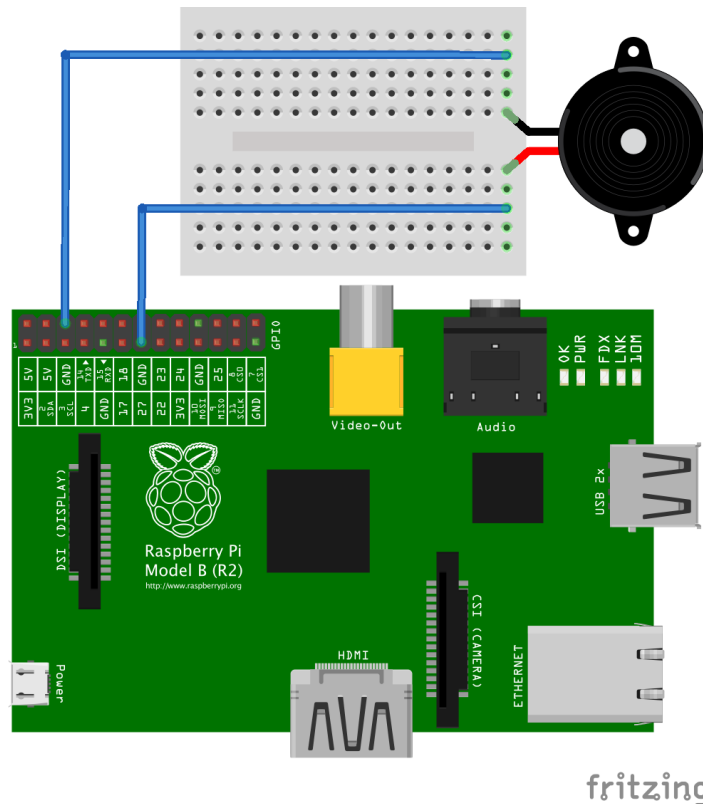
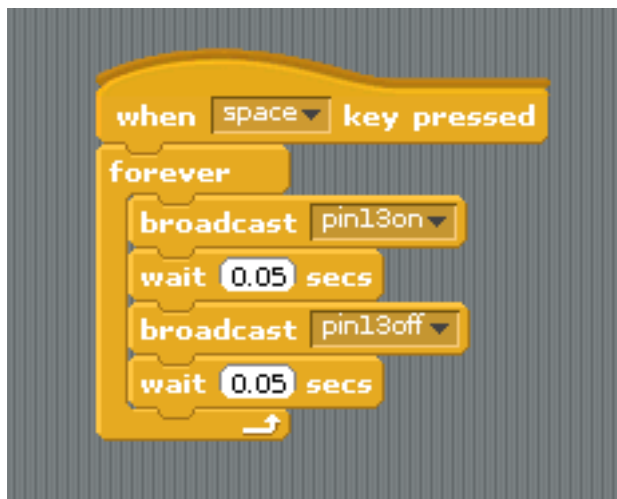
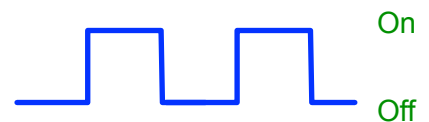


BUZZ BUZZER WITH SCRATCH



Simply turning on the buzzer like we did with the led will not work. This is a passive piezo buzzer so we need to feed it a square waveform. In other words, turn it on and off quickly.

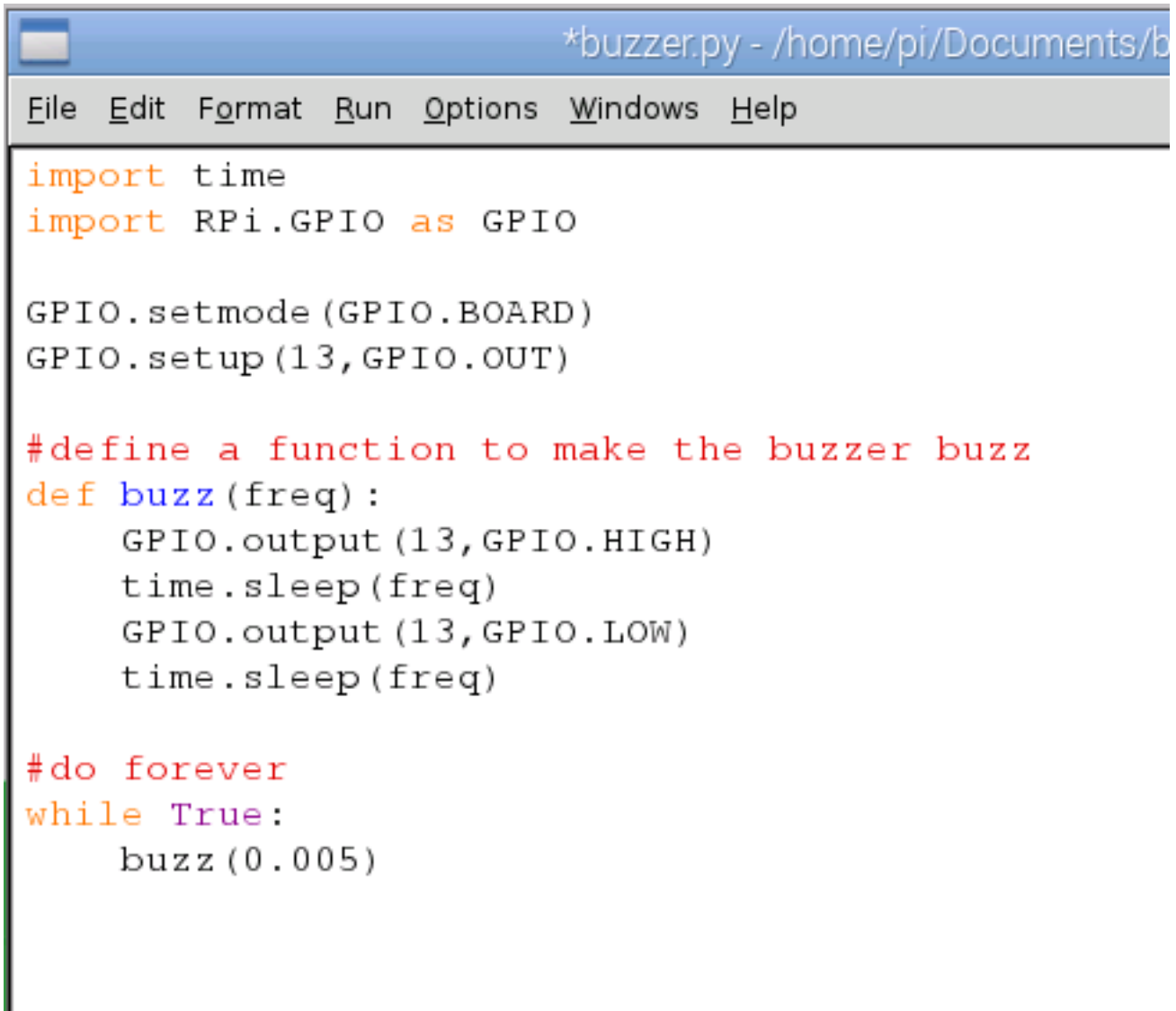


How does it sound?



BUZZ BUZZER WITH python™

Scratch is limited in how fast it can switch things.
Python can do much better!



```
import time
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BOARD)
GPIO.setup(13, GPIO.OUT)

#define a function to make the buzzer buzz
def buzz(freq):
    GPIO.output(13, GPIO.HIGH)
    time.sleep(freq)
    GPIO.output(13, GPIO.LOW)
    time.sleep(freq)

#do forever
while True:
    buzz(0.005)
```

Top Tip:
to stop the program type  +  together.

Try adjusting the value of freq that you
use with buzz to see how it affects the
sound.

Can you modify the buzz function so that you can
set how long the buzzers makes a noise?
Example: buzz(0.005,10) will make a sound for
10 seconds

BUZZ BUZZER WITH python™

Here are a couple of ways of making the buzzer sound for a certain period of time.

```
import time
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BOARD)
GPIO.setup(13,GPIO.OUT)

#define a function to make the buzzer buzz
def buzz(freq,dur):
    # calculate how many times we need to switch the buzzer
    repeat = dur/(2*freq)
    count = 0
    while count < repeat:
        GPIO.output(13,GPIO.HIGH)
        time.sleep(freq)
        GPIO.output(13,GPIO.LOW)
        time.sleep(freq)
        count = count + 1

buzz(0.005,10)
```

```
import time
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BOARD)
GPIO.setup(13,GPIO.OUT)

#define a function to make the buzzer buzz
def buzz(freq,dur):
    # calculate how many times we need to switch the buzzer
    repeat = dur/(2*freq)
    for count in range(int(repeat)):
        GPIO.output(13,GPIO.HIGH)
        time.sleep(freq)
        GPIO.output(13,GPIO.LOW)
        time.sleep(freq)

buzz(0.005,10)
```

Which one is best?

Which one is easiest to understand?