



Προγραμματισμός με Python
στο Raspberry Pi

LED και PWMLED



για το Code Club
Απρίλιος 2016

Σύλλογος Εκπαιδευτικών Πληροφορικής Χίου

LED και PWMLED: Ομοιότητες και διαφορές

LED

Δυο δυνατές καταστάσεις για το led:
ή θα ανάβει πλήρως ή θα είναι σβηστό.

Μόνο `led.on()` και `led.off()`

PWMLED

Ορίζουμε τη φωτεινότητα του led σε οποιαδήποτε τιμή μεταξύ 0 και 1.

Όχι μόνο `led.on()` και `led.off()`,
αλλά και `led.value = ...`

```
1 from gpiozero import LED, PWMLED
2 from time import sleep
3 led1 = LED(16)
4 led2 = PWMLED(24)
5 led1.on()
6 led2.on()
7 sleep(1)
8 led2.value = 0.5
9 sleep(1)
10 led1.off()
11 led2.value = 0
```

LED & PWMLED

$\Delta 1.1$

Γράψτε ένα πρόγραμμα που θα αναβοσβήνει ένα LED, αφού πρώτα ρωτήσει το χρήστη: **α)** πόσες φορές και **β)** με τι διάρκεια θα ανάβει και θα σβήνει.

```
1 from gpiozero import LED
2 from time import sleep
3 led = LED(16)
4 while True:                                # Για πάντα (μέχρι να πατηθεί Ctrl+C ή να βρεθεί κάποια break).
5     led.on()
6     sleep(0.5)                               # Περίμενε 0.5 sec.
7     led.off()
8     sleep(0.5)                               # Περίμενε 0.5 sec.
```



β) ... για πόση διάρκεια θα ανάβει και θα σβήνει.

```
1  from gpiozero import LED
2  from time import sleep
3  led = LED(16)
4  print ("Δώσε διάρκεια:")
5  t = float ( input() )
6  while True:
7      led.on()
8      sleep(t)
9      led.off()
10     sleep(t)
```

Μετατροπή του κειμένου σε αριθμό
και μάλιστα float για να δέχεται και
πραγματικές τιμές.

Περίμενε t δευτερόλεπτα.

Περίμενε t δευτερόλεπτα.

α) ... πόσες φορές θα ανάβει και θα σβήνει.

```
1  from gpiozero import LED
2  from time import sleep
3  led = LED(16)
4  print ("Δώσε διάρκεια:")
5  t = float ( input() )
6  ...
7  ...
8
9  ...
10 ...
11
12 print ("Τέλος")
```

Ρώτα το χρήστη τον αριθμό που επιθυμεί
και εκχώρησέ τον σε μια μεταβλητή, π.χ. n.

Επανάλαβε n φορές τις εντολές
για το άναψε-σβήσε του led.

LED & PWMLED

$\Delta 1.2$

Κάντε ένα LED να μοιάζει με ένα κερι που τρεμοπαίζει (το led ανάβει σε τυχαία ένταση και για τυχαίο χρονικό διάστημα).

```
1 from gpiozero import PWMLED
2 from time import sleep
3 led = PWMLED(16)
4 while True:
5     led.value = 0.1
6     sleep(0.2)
7     led.value = 0.5
8     sleep(0.2)
9     led.value = 1
10    sleep(0.2)
```

Η ιδιότητα **value** ενός PWMLED μας επιτρέπει
να ορίσουμε τη φωτεινότητά του (0 μέχρι 1).

```
1 from random import randint
2 z = randint(1, 6)
3 print (z)
```

Η randint παράγει έναν **ακέραιο** αριθμό
στο διάστημα που της καθορίζουμε.
Εδώ, στο κλειστό διάστημα [1,6].

LED & PWMLED

$\Delta 1.3$

Κάντε ένα led να αναβοσβήνει αργά, ώστε να μοιάζει σαν να αναπνέει.

```
1 from gpiozero import PWMLED
2 from time import sleep
3 led = PWMLED(16)
4
5 # Για πάντα:
6     # Ξεκίνα από φωτεινότητα μηδέν και αύξησε σταδιακά τη φωτεινότητα
7     # με μια πολύ μικρή καθυστέρηση (~0.05 sec) σε κάθε αλλαγή.
8     # Όταν φτάσεις στη φωτεινότητα ένα, περίμενε λίγο (~0.3 sec).
9     # Τώρα, ξεκίνα από φωτεινότητα ένα και μείωσε σταδιακά τη φωτεινότητα
10    # με μια πολύ μικρή καθυστέρηση σε κάθε αλλαγή, μέχρι αυτή να γίνει μηδέν.
11    # Όταν το led έχει σβήσει, περίμενε λίγο (~0.3 sec).
```

Παράρτημα

Ενδεικτικές λύσεις

Γράψτε ένα πρόγραμμα που θα αναβοσβήνει ένα LED, αφού πρώτα ρωτήσει το χρήστη: α) πόσες φορές και β) με τι διάρκεια θα ανάβει και θα σβήνει.

```
1  from gpiozero import LED
2  from time import sleep
3  led = LED(16)
4  print ("Δώσε διάρκεια:")
5  t = float ( input() )
6  print ("Πόσες φορές;")
7  n = int ( input() )
8  while n > 0:
9      led.on()
10     sleep(t)
11     led.off()
12     sleep(t)
13     n = n - 1
14  print ("Τέλος")
```

Εναλλακτική λύση της Δ1.1 με τη χρήση συνάρτησης.

```
1  from gpiozero import LED
2  from time import sleep
3  led = LED(16)
4  def blink (diarkeia, fores):
5      while fores > 0:
6          led.on()
7          sleep(diarkeia)
8          led.off()
9          sleep(diarkeia)
10         fores = fores - 1
11 t = float ( input("Δώσε διάρκεια: ") )
12 n = int ( input("Πόσες φορές; ") )
13 blink (t, n)
14 print ("Τέλος")
```

Κάντε ένα LED να μοιάζει με ένα κερι που τρεμοπαίζει (το led ανάβει σε τυχαία ένταση και για τυχαίο χρονικό διάστημα).

```
1  from gpiozero import PWMLED
2  from time import sleep
3  from random import randint
4  led = PWMLED(16)
5
6  while True:
7      brightness = randint(10, 100) / 100
8      time = randint(0, 10) / 100
9      led.value = brightness
10     sleep(time)
```

Δ1.3

Κάντε ένα led να αναβοσβήνει αργά, ώστε να μοιάζει σαν να αναπνέει.

```
1 from gpiozero import PWMLED
2 from time import sleep
3 led = PWMLED(16)
4 t1 = 0.01 # 0 χρόνος για κάθε αλλαγή φωτεινότητας
5 t2 = 0.2 # 0 χρόνος ανάμεσα στις αναπνοές
6 while True:
7     i = 0
8     while i <= 100:
9         led.value = i/100
10        i = i + 1
11        sleep(t1)
12    sleep(t2)
13    i = 1
14    while i >= 0:
15        led.value = i
16        i = i - 0.01
17        sleep(t1)
18    sleep(t2)
19    print ("Αναπνοή...")
```

Δ1.3

... ένα ακόμα led να αναβοσβήνει αντίστροφα από το άλλο.

```
1 from gpiozero import PWMLED
2 from time import sleep
3 l1 = PWMLED(16)
4 l2 = PWMLED(25)
5 t1 = 0.01
6 t2 = 0.2
7 while True:
8     i = 0
9     while i <= 100:
10         l1.value = i/100
11         l2.value = 1 - l1.value
12         i = i + 1
13         sleep(t1)
14     sleep(t2)
```

```
15 i = 1
16 while i >= 0:
17     l1.value = i
18     l2.value = 1 - l1.value
19     i = i - 0.01
20     sleep(t1)
21 sleep(t2)
22 print ("Αναπνοή...")
```




Το υλικό αυτό διατίθεται με άδεια Creative Commons Αναφορά Δημιουργού - Παρόμοια Διανομή 3.0.
<http://creativecommons.org/licenses/by-sa/3.0/gr/>

Η αναφορά σε αυτό θα πρέπει να γίνεται ως εξής:

*Προγραμματισμός με Python στο Raspberry Pi. Γιώργος Χατζηνικολάκης,
Σύλλογος Εκπαιδευτικών Πληροφορικής Χίου, Απρίλιος 2016.*