

Week 6

AIM: Write the python Flash an LED based on cron output (acts as an alarm)

Components Required:

Connecting pins

1K Ω resistor

Raspberry pi 3 B+ Model

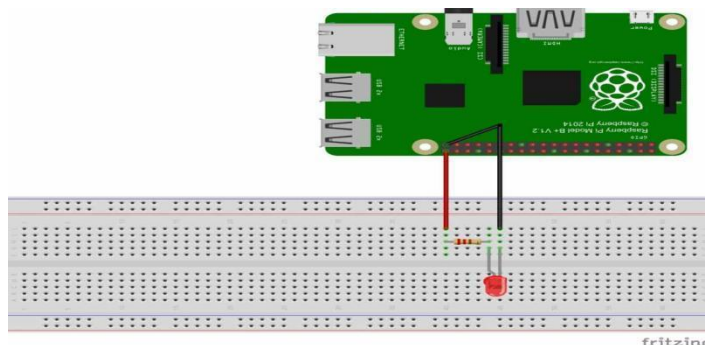
LED

Bread Board

HDMI Cable

Power Cable

Circuit Explanation:



As shown in the circuit diagram we are going to connect an LED between PIN11 and PIN6(GROUND) so to limit the current we are connecting a 1K Ω resistor in series with the LED.

Working Explanation:

Turn ON your PI and go to the desktop.

1. On the desktop, go the Start Menu and choose for the **PYTHON 3**, as shown in figure below.
2. After that, PYHON will run and you will see a window as shown in below figure.
3. After that, click on *New File* in *File* Menu, You will see a new Window open
4. Save this file as ***alarm.py*** on the desktop
5. After that write the program for ***alarm.py*** as given below and execute the program by clicking on “RUN” on ‘DEBUG’ option.

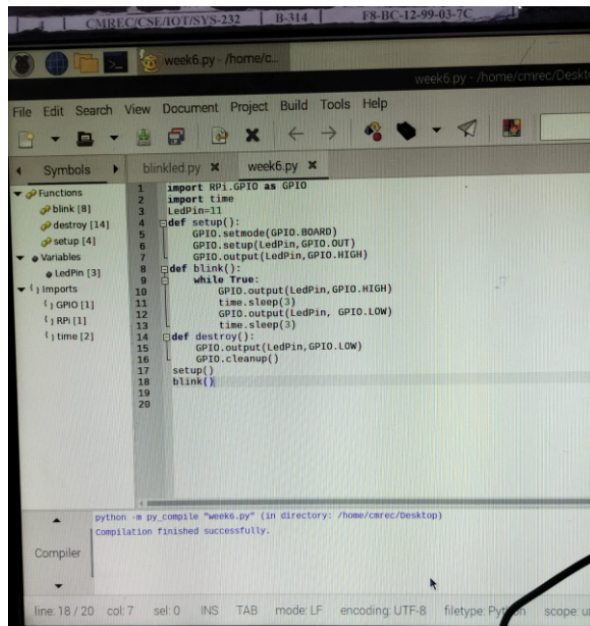
Code Explanation:

In the Python program, first we have imported two packages RPi.GPIO and Time. The package RPi.GPIO will help us in controlling the GPIO Pins of the Raspberry Pi.

The first important function of the RPi.GPIO Module is the `setmode()`. Using `GPIO.setmode()`, we can select either GPIO Numbering of the Pins or Physical Numbering. By using `GPIO.setmode(GPIO.BOARD)`, we are selecting the Physical Numbering Scheme.

Create two functions `setup()` and `blink()`. The next function is the `setup(Ledpin, GPIO.OUT)`. This function will allow us to set the pin as either input (`GPIO.IN`) or as output (`GPIO.OUT`). In the program, I have set the `ledPin` as output by using **`GPIO.setup(ledPin, GPIO.HIGH)`**. `setup()` After setting the LED Pin as OUTPUT, now we need to set the state of this OUTPUT i.e. HIGH (`GPIO.HIGH`) or LOW (`GPIO.LOW`). For this, we need to use the function `output(pin, state)`. So, in our program, we need to use **`GPIO.output(ledPin, GPIO.HIGH)`** for turning ON the LED and **`GPIO.output(ledPin, GPIO.LOW)`** for turning it OFF. Clean up GPIO.

Code:

A screenshot of a code editor window titled 'week6.py - /home/carec/Desktop'. The editor shows a Python script for controlling an LED. The script imports RPi.GPIO as GPIO and the time module. It defines a variable 'ledPin' as 11. The 'setup()' function sets the pin mode to GPIO.OUT and the initial state to GPIO.HIGH. The 'blink()' function is a loop that turns the LED on (GPIO.HIGH) for 3 seconds and then off (GPIO.LOW) for 3 seconds. The 'destroy()' function sets the pin mode to GPIO.LOW and cleans up the GPIO module. The script calls 'setup()', 'blink()', and 'destroy()' in sequence. The status bar at the bottom indicates 'line:18 / 20 col:7 sel:0 INS TAB mode:LF encoding:UTF-8 filetype:Python scope:un'.

```
1 import RPi.GPIO as GPIO
2 import time
3 ledPin=11
4 def setup():
5     GPIO.setmode(GPIO.BOARD)
6     GPIO.setup(ledPin, GPIO.OUT)
7     GPIO.output(ledPin, GPIO.HIGH)
8 def blink():
9     while True:
10        GPIO.output(ledPin, GPIO.HIGH)
11        time.sleep(3)
12        GPIO.output(ledPin, GPIO.LOW)
13        time.sleep(3)
14 def destroy():
15     GPIO.output(ledPin, GPIO.LOW)
16     GPIO.cleanup()
17 setup()
18 blink()
19
20
```

Output:

LED IS ON

LED IS OFF

