**Threaded callbacks for GPIO inputs**

RPi.GPIO runs a second thread for callback functions. This means that callback functions can be run at the same time as your main program, in immediate response to an edge. For example:

**def** my\_callback(channel):

**print**('This is a edge event callback function!')

**print**('Edge detected on channel',channel)

**print**('This is run in a different thread to your main program')

GPIO.add\_event\_detect(channel, GPIO.RISING, callback=my\_callback) *# add rising edge detection on a channel*

...the rest of your program...

OR

GPIO.add\_event\_detect(channel, GPIO.RISING)

GPIO.add\_event\_callback(channel, my\_callback)

**Switch debounce**

You may notice that the callbacks are called more than once for each button press. This is as a result of what is known as 'switch bounce'. There are two ways of dealing with switch bounce:

* add a 0.1uF capacitor across your switch.
* software debouncing
* a combination of both

To debounce using software, add the bouncetime= parameter to a function where you specify a callback function. Bouncetime should be specified in milliseconds. For example:

*# add rising edge detection on a channel, ignoring further edges for 200ms for switch bounce handling*

GPIO.add\_event\_detect(channel, GPIO.RISING, callback=my\_callback, bouncetime=200)

or

GPIO.add\_event\_callback(channel, my\_callback, bouncetime=200)

**Remove event detection**

If for some reason, your program no longer wishes to detect edge events, it is possible to stop them:

GPIO.remove\_event\_detect(channel)

Zie voorbeeld volgende pagina!

Voorbeeld CALLBACK GPIO

from tkinter import \*

import RPi.GPIO as GPIO

window = Tk()

window.title ("Schakelaar volgen")

window.geometry("800x400")

schakelaar =18

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)

GPIO.setup(schakelaar, GPIO.IN, pull\_up\_down=GPIO.PUD\_UP)

# add callback to schakelaar

def my\_callback(pin\_nr):

global toestand\_nu

toestand\_nu = not toestand\_nu

print("callback called!", pin\_nr,toestand\_nu)

**GPIO.add\_event\_detect(schakelaar, GPIO.FALLING, callback=my\_callback, bouncetime=200 )**

vara=StringVar()

vara.set("test")

mijnLabel1= Label( window, textvariable=vara ,relief =GROOVE)

mijnLabel1.place(x=10,y=50)

toestand\_nu = True

toestand\_vorige = True

while True:

if toestand\_vorige != toestand\_nu:

vara.set(toestand\_nu)

toestand\_vorige=toestand\_nu

window.update()

‘’’

Opmerking :

GUI aanpassen in de callback zelf is een probleem!

Tkinter is single-threaded, and all access to widgets must be from the same thread.

‘’’