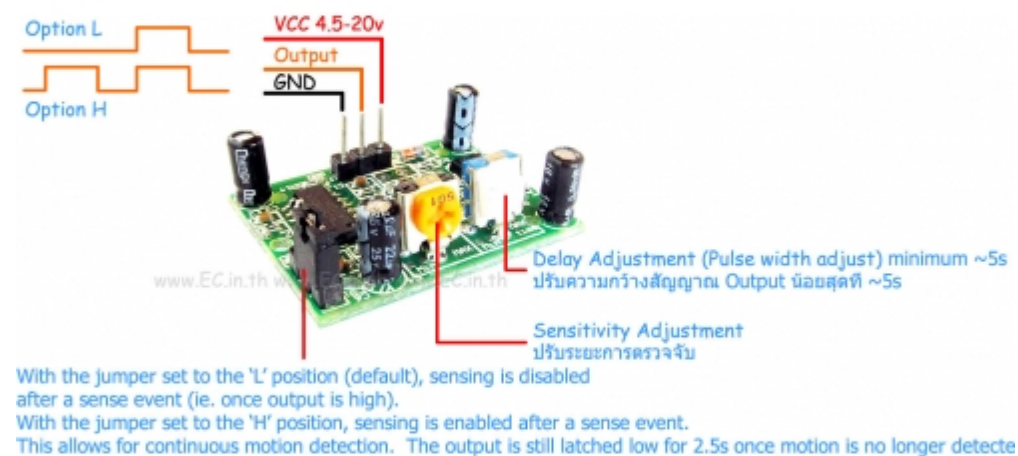


PIR Cidlo



Delay nastvit na minimum

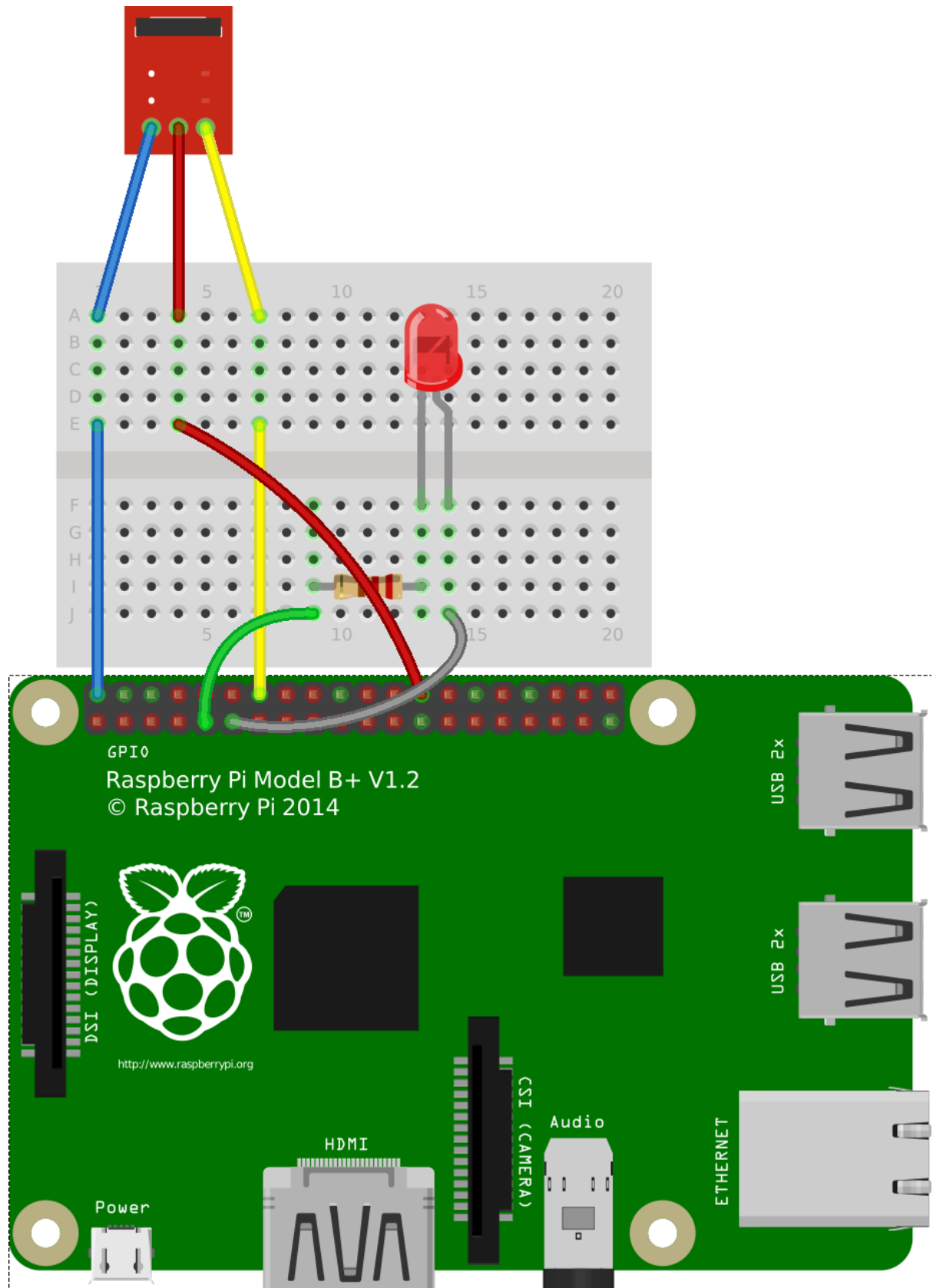
Citlivost na polovic

<http://www.raspberrypi-spy.co.uk/2013/02/cheap-pir-sensors-and-the-raspberry-pi-part-2/>

<http://www.cmprogrammers.com/post.php?data=2014-08-05+20:38:00>

wget <http://www.cmprogrammers.com/projects/documents/alarm.py>

wget <http://www.cmprogrammers.com/projects/documents/uz.sh>



fritzing

```

import RPi.GPIO as GPIO
import time

# Use BCM GPIO references
# instead of physical pin numbers
GPIO.setmode(GPIO.BCM)
# Define GPIO to use on Pi
GPIO_PIR = 7
GPIO_LED = 17

print "PIR Module Test (CTRL-C to exit)"
# Set pin as input
GPIO.setup(GPIO_PIR,GPIO.IN)      # Echo
GPIO.setup(GPIO_LED,GPIO.OUT)

Current_State = 0
Previous_State = 0

GPIO.output(GPIO_LED,True)
time.sleep(2)
GPIO.output(GPIO_LED,False)
try:
    print "Waiting for PIR to settle ..."
    # Loop until PIR output is 0
    while GPIO.input(GPIO_PIR)==1:
        Current_State = 0

    print " Ready"
    # Loop until users quits with CTRL-C
    while True :
        # Read PIR state
        Current_State = GPIO.input(GPIO_PIR)

        if Current_State==1 and Previous_State==0:
            # PIR is triggered
            print " Motion detected!"
            print time.ctime()
            GPIO.output(GPIO_LED,True)
            # Record previous state
            Previous_State=1
        elif Current_State==0 and Previous_State==1:
            # PIR has returned to ready state
            print " Ready"
            GPIO.output(GPIO_LED,False)
            Previous_State=0

        # Wait for 20 milliseconds
        time.sleep(0.02)
except KeyboardInterrupt:
    print " Quit. Cleaning up GPIOs"
    GPIO.cleanup()

```