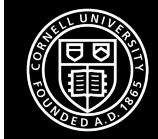
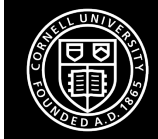


ECE 5725
Embedded Operating Systems
Lecture 12

Prof. Joseph F. Skovira

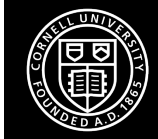


A few items

Lab2 Week 1

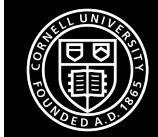
Lab 1 Report

Homework 2



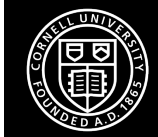
Python GPIO interrupts

```
import RPi.GPIO as GPIO
Import subprocess
GPIO.setmode(GPIO.BCM)

GPIO.setup(19, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(13, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(26, GPIO.IN, pull_up_down=GPIO.PUD_UP)
def GPIO19_callback(channel):
    print "falling edge detected on 19”
def GPIO13_callback(channel):
   cmd = 'echo “pause”’ ‘
   subprocess.check_output(cmd, shell=True)
# “ main” part of the program
GPIO.add_event_detect(19, GPIO.FALLING, callback=GPIO19_callback, bouncetime=300)
GPIO.add_event_detect(13, GPIO.FALLING, callback=GPIO13_callback, bouncetime=300)
 Try:
     print "Waiting for falling edge on port 26"
     GPIO.wait_for_edge(26, GPIO.FALLING)
     print ”Falling edge detected on port 26”
 except KeyboardInterrupt:
     GPIO.cleanup()       # clean up GPIO on CTRL+C exit
GPIO.cleanup()           # clean up GPIO on normal exit
```



Python GPIO interrupts

```
import RPi.GPIO as GPIO
Import subprocess
GPIO.setmode(GPIO.BCM)

GPIO.setup(19, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(13, GPIO.IN, pull_up_down=GPIO.PUD_UP)

def GPIO19_callback(channel):
    print "falling edge detected on 19”

def GPIO13_callback(channel):
   cmd = 'echo “pause”’ ‘
   subprocess.check_output(cmd, shell=True)

# “ main” part of the program
GPIO.add_event_detect(19, GPIO.FALLING, callback=GPIO19_callback, bouncetime=300)
GPIO.add_event_detect(13, GPIO.FALLING, callback=GPIO13_callback, bouncetime=300)

#  Background Processing
Code_Run = True
While CodeRun  # simple loop
   print “tick”
   time.sleep(1)

GPIO.cleanup()          # clean up GPIO on normal exit
```



Python GPIO interrupts

```
import RPi.GPIO as GPIO
Import subprocess
GPIO.setmode(GPIO.BCM)

GPIO.setup(19, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(13, GPIO.IN, pull_up_down=GPIO.PUD_UP)

def GPIO19_callback(channel):
    global CodeRun
    print "falling edge detected on 19 – quit!"
    CodeRun = False

def GPIO13_callback(channel):
   cmd = 'echo "pause"' '
   subprocess.check_output(cmd, shell=True)

# " main" part of the program
GPIO.add_event_detect(19, GPIO.FALLING, callback=GPIO19_callback, bouncetime=300)
GPIO.add_event_detect(13, GPIO.FALLING, callback=GPIO13_callback, bouncetime=300)

#  Background Processing
Code_Run = True
While CodeRun  # simple loop
    print "tick"
    time.sleep(1)

GPIO.cleanup()          # clean up GPIO on normal exit
```

PyGame Display Example

```
import pygame     # Import pygame graphics library
import os    # for OS calls

#   os.putenv('SDL_VIDEODRIVER', 'fbcon')   # Display on piTFT
#   os.putenv('SDL_FBDEV', '/dev/fb1')

pygame.init()

size = width, height = 320, 240
speed = [2,2]
black = 0, 0, 0

screen = pygame.display.set_mode(size)
ball = pygame.image.load("magic_ball.png")
ballrect = ball.get_rect()

while 1:
      ballrect = ballrect.move(speed)
      if ballrect.left < 0 or ballrect.right > width:
         speed[0] = -speed[0]
      if ballrect.top < 0 or ballrect.bottom > height:
         speed[1] = -speed[1]


      screen.fill(black)             # Erase the Work space
      screen.blit(ball, ballrect)    # Combine Ball surface with workspace surface
      pygame.display.flip()          # display workspace on screen
```

Lab2 Tips….

Multiple Button Hits

Use different Ball images

Ball Initial position

Image Flicker

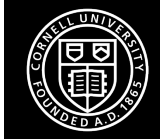
Bounce doesn't play on piTFT

```
sudo SDL_VIDEODRIVER=fbcon SDL_FBDEV=/dev/fb1 mplayer -vo sdl -framedrop bigbuckbunny320p.mp4
```

```
os.putenv('SDL_FBDEV', '/dev/fb1')
os.putenv('SDL_VIDEODRIVER', 'fbcon'')
```

Quit = Bail Out Button

Timeout Code



A few more Lab 2 items….

Bounce is too fast

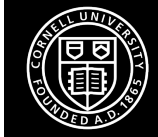
Ball off screen or stuck at edge…

Balls stick together on collision

Bounce pattern repeats

No Touch Bounce

Black on Black



Date and Time

Network Time Protocol(NTP)

Timesyncd