SIT210: Embedded Systems Development

Task 5.1P RPi - GPIO Basics

Raspberry Pi platform provides variety of functionalities through 40 pins available on the board. From these, 28 are General Purpose Input Output (GPIO) pins. We will be using these pins in this task and for our developments to use many capabilities of RPi.

Hardware Required

Raspberry Pi, Keyboard, Mouse, LEDs, Breadboard, jumper wires

Software Required

Noob software installed on the Raspberry Pi

Pre-requisites: You must do the following before this task

1) Complete previous Raspberry Pi task 4.1P

Task Objective

In this task, you will learn the basics of Python coding for Raspberry Pi using the GPIO library. While this task is focusing on Python, you can use any programming language you are most comfortable with (and supported by RPi platform) to perform this task and future tasks incorporating a RPi.

Steps:

1) Build a basic breadboard interface with the Raspberry Pi that allows you to turn on or off an LED by connecting the terminals of the LED to a GPIO pin on RPi (refer to GPIO map below to find a suitable pin)
2) Turn on your RPi, open a terminal and install GPIO (if not already) using the following command: sudo apt-get install rpi.gpio
3) Create a python file using a text editor, such as nano (in your home folder or any other folder) and name it appropriately, here we call it blinker: nano blinker.py
4) In the python file you created, put in the import statement: import RPi.GPIO as GPIO
5) Declare pin number standard you will be using: GPIO.setmode(GPIO.BOARD)
6) Set the pin mode for the GPIO pin you have connected your LED, for example, pin 10: GPIO.setup(10, GPIO.OUT)
7) Turn the LED on by issuing command: GPIO.output(10, GPIO.HIGH)
8) Add delay:

```
include time
time.sleep(0.25)
```

9) Turn off LED: GPIO.output(10, GPIO.LOW)
10) Repeat 7 to 8 by putting in a while loop, and clean up your code as below:

```
import time
```

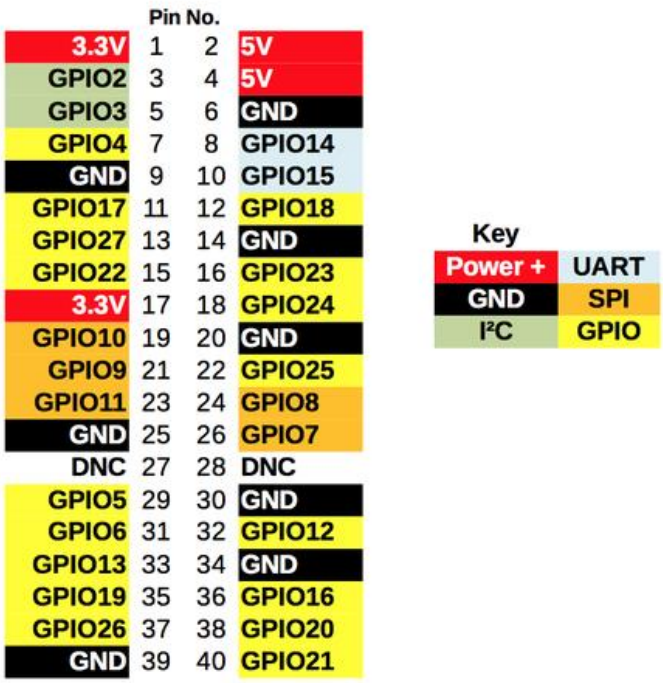
```
try:
    while 1:
        GPIO.output(10, GPIO.HIGH)
        time.sleep(0.25)
except KeyboardInterrupt:
    GPIO.cleanup()
```

11) Save the file, run the code using *sudo* command (you might need to just use *python* command): *sudo python blinker.py*

Task Submission Details

Q1: Take a video of your RPi with LED blinking. Briefly describe your circuit board, your code and how you have developed it in the video. Upload your video to the website of your choice, and put the link below.

RPi GPIO map:



Remember, anytime you submit a task to OnTrack, it is a good practice to check the status of any existing tasks, and the future tasks you are expected to complete. If you have got feedback on previous tasks, you may need to fix and resubmit some of your work. You want to check out why, so that you can learn from this and make it faster and easier to accomplish later work to the required standard.