

# Lab Manual for Embedded System Design

## Lab No. 13

### Raspberry PI Hardware Applications

---

#### Objectives

*Design and Interface Raspberry-PI Hardware Applications.  
Design Traffic Lights and Pulse Width Modulation (PWM)  
circuits.*

---

# LAB # 13

## Raspberry PI Hardware Applications

### Introduction

#### Raspberry – Pi GPIO Controller:

- The R-Pi GPIO connector actually has a number of different types of connection on them. There are:
- True GPIO (General Purpose Input Output) pins that you can use to turn LEDs on and off etc.
- I2C interface pins that allow you to connect hardware modules with just two control pins
- SPI interface with SPI devices, a similar concept to I2C but a different standard
- Serial Rx and Tx pins for communication with serial peripherals
- In addition, some of the pins can be used for PWM (pulse Width Modulation) for power control and another type of pulse generation for controlling servo motors called PPM (Pulse Position Modulation).

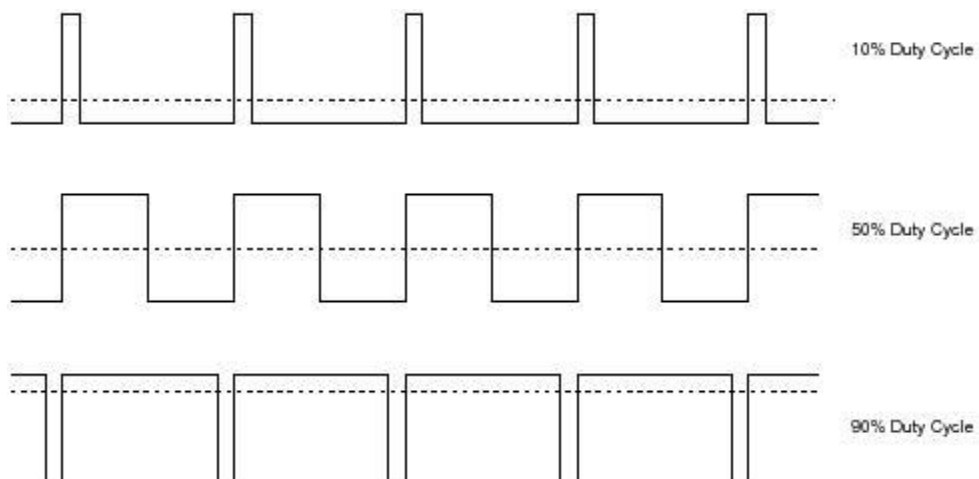
**Raspberry Pi 3 GPIO Header**

Pin#	NAME		NAME	Pin#
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1 , I <sup>2</sup> C)		DC Power 5v	04
05	GPIO03 (SCL1 , I <sup>2</sup> C)		Ground	06
07	GPIO04 (GPIO_GCLK)		(TXD0) GPIO14	08
09	Ground		(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)		Ground	14
15	GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23	16
17	3.3v DC Power		(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)		Ground	20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08	24
25	Ground		(SPI_CE1_N) GPIO07	26
27	ID_SD (I <sup>2</sup> C ID EEPROM)		(I <sup>2</sup> C ID EEPROM) ID_SC	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12	32
33	GPIO13		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40

Fig. 9.1: GPIO Inputs

### PWM:

Pulse width modulation (PWM) is a powerful technique for controlling analog circuits with a microprocessor's digital outputs.



## Time Boxing

Activity Name	Activity Time	Total Time
Login Systems + Setting up Raspberry PI Environment	3 mints + 5 mints	8 mints
Walk through Theory & Tasks	60 mints	60 mints
Implement Tasks	80 mints	80 mints
Evaluation Time	30 mints	30 mints
Total Duration		178 mints

## Objectives

1. Design and Interface Raspberry-PI Hardware Applications.
2. Design Traffic Lights and Pulse Width Modulation (PWM) circuits.

## Lab Tasks/Practical Work

1. In this task we will write a simple Python script in LXTerminal to glow LED using GPIO 7 of Raspberry-Pi.

```
import RPi.GPIO as GPIO
GPIO.setmode (GPIO.BOARD)
GPIO.setup (7, GPIO.OUT)
GPIO.output (7, True)
```

2. In this task we will write a simple Python script in LXTerminal to blink single LED using a GPIO 7 of Raspberry-Pi.

```
import RPi.GPIO as GPIO
import time
GPIO.setmode (GPIO.BOARD)
GPIO.setup (7, GPIO.OUT)
While True:
    GPIO.output (7, True)
    time.sleep (0.5)
    GPIO.output (7, False)
    time.sleep (0.5)
```

3. In this task we will write a simple Python script to control LED using GPIOs as PWM application.

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setup(7, GPIO.OUT)
pwm_led = GPIO.PWM(7, 500)
pwm_led.start(100)
while True:
    duty_s = input("Enter Brightness (0 to 100):")
    duty = int(duty_s)
    pwm_led.ChangeDutyCycle(duty)
```

4. In this task we will write a simple Python script for Traffic control signals using various GPIOs.



```
import RPi.GPIO as GPIO
import time
GPIO.cleanup()
```

```
GPIO.setmode(GPIO.BOARD)
GPIO.setup(3,GPIO.OUT)
GPIO.setup(5,GPIO.OUT)
GPIO.setup(7,GPIO.OUT)
GPIO.setwarnings(False)
while True:
    GPIO.output(7,GPIO.HIGH)
    time.sleep(5)
    GPIO.output(5,GPIO.HIGH)
    time.sleep(2)
    GPIO.output(7,GPIO.LOW)
    GPIO.output(5,GPIO.LOW)
    GPIO.output(3,GPIO.HIGH)
    time.sleep(10)
    GPIO.output(3,GPIO.LOW)
    GPIO.output(5,GPIO.HIGH)
```

4. Write a simple Python script to drive dc motors in both directions.