**WEEK - 03 Program to interface ultrasonic and PIR sensors to RPi**

**PRELAB QUESTIONS - 03**

1. What is the range of ultrasonic sensors?

Ans:

Ultrasonic sensors can measure distance and detect the presence of an object without making physical contact. They do so by producing and monitoring an ultrasonic echo. Depending on the sensor and object properties, the effective range in air is between a few centimeters up to several meters. The ultrasonic sensor (or transducer) generates and emits ultrasonic pulses that are reflected back towards the sensor by an object that is within the field of view of the sensor. Ultrasonic sensors use sound waves above the 20 kHz range to detect objects in proximity, similar to how bats use echolocation to maneuver without colliding into obstacles.

2.What is the working principle of ultrasonic sensor?

Ans: Ultrasonic sensors emit short, high-frequency sound pulses at regular intervals. These propagate in the air at the velocity of sound. If they strike an object, then they reflected back as an echo signal to the sensor, which itself computes the distance to the target based on the time-span between emitting the signal and receiving the echo.

3.How does a PIR sensor work?

Ans: PIR (passive infrared) sensors utilise the detection of infrared that is radiated from all objects that emit heat. This type of emission is not visible to the human eye, but sensors that operate using infrared wavelengths can detect such activity. They are sometimes referred to as ‘motion-based detectors’, as they sense the presence of people, animals and objects through the movement of their infrared wavelengths.

They are referred to as “passive” due to the fact that no heat or energy is emitted by the sensor itself. It is also important to remember that PIR sensors detect the emission of infrared radiation, and *not*heat.

4. List the application areas of PIR sensor.

Ans:

PIR sensors find diverse applications in various industries:

1. Home and Business Security Systems: Integrated into security alarms for motion detection.

2. Automatic Lighting Systems: Utilized to trigger lights based on motion, enhancing energy efficiency.

3. Industrial Machinery Monitoring: Employed for predictive maintenance by monitoring vibrations in machinery.

4. Structural Health Monitoring: Used to assess vibrations in structures for safety and integrity.

5. What are the different types of motion sensors and how they work?

Ans: There are two widely used types of motion sensors: active ultrasonic and passive infrared (PIR).

Active ultrasonic sensors and passive infrared sensors are the two most common motion sensor technologies, both of which are known for their accuracy and reliability.

6. What is the sensitivity range of PIR?

Ans:

The sensitivity range of PIR sensors extends up to about 20 feet (6 meters), allowing them to effectively detect movement within this distance. This range makes PIR sensors suitable for a variety of residential and commercial applications.

Lab programs:

1)Write a program to Control PIR sensor.

Code:

import sys sys.path.append('/home/pi/Adafruit-Raspberry-Pi-Python-Code- legacy/Adafruit\_MCP230xx')

from Adafruit\_MCP230xx import Adafruit\_MCP230XX

import time

mcp=Adafruit\_MCP230XX(busnum=1,address=0x21,num\_gpios=16) mcp.config(0,mcp.INPUT)

mcp.pullup(0,1)

while True:

i=mcp.input(0)

time.sleep(1)

if i==1:

print "person detect"

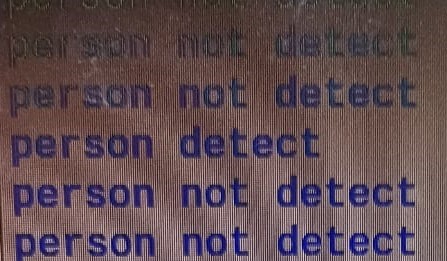
time.sleep(1)

if i==0:

print "person not detect"

time.sleep(1)

Output:





1. Write a program to Control Ultra Sconic sensor

Code:

import time, sys

import RPi.GPIO as GPIO

GPIO.setwarnings(False )

GPIO.setmode(GPIO.BOARD)

GPIO\_TRIGGER=16

GPIO\_ECHO = 18

GPIO.setup(GPIO\_TRIGGER, GPIO.OUT)

GPIO.setup(GPIO\_ECHO, GPIO.IN)

GPIO.output(GPIO\_TRIGGER, False)

i=1 time.sleep(0.5)

while(True):

GPIO.output(GPIO\_TRIGGER, True)

time.sleep(0.0001)

GPIO.output(GPIO\_TRIGGER, False)

while(GPIO.input(GPIO\_ECHO)==0):

Start = time.time()

while(GPIO.input(GPIO\_ECHO)==1):

Stop = time.time()

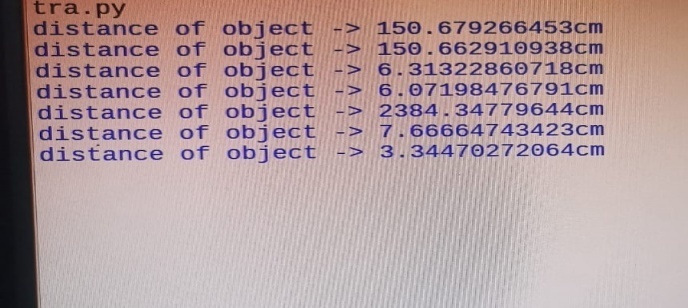
Elapsed = stop-start

distance=elapsed\*34300

distance=distance/2

print(“Distance of object -->“+str(distance)+”cm”)

time.sleep(5) 1+=1

**OUTPUT: **

****