**Mail Code:**

import smtplib,ssl

from picamera import PiCamera

from time import sleep

from email.mime.multipart import MIMEMultipart

from email.mime.base import MIMEBase

from email.mime.text import MIMEText

from email.utils import formatdate

from email import encoders

camera = PiCamera()

camera.start\_preview()

sleep(5)

camera.capture('/home/pi/image.jpg') # image path set

sleep(5)

camera.stop\_preview()

def send\_an\_email():

toaddr = 'intruderalertingsystem@gmail.com' # To id

me = 'intruderalertingsystem@gmail.com' # your id

subject = "What's News" # Subject

msg = MIMEMultipart()

msg['Subject'] = subject

msg['From'] = me

msg['To'] = toaddr

msg.preamble = "test "

#msg.attach(MIMEText(text))

part = MIMEBase('application', "octet-stream")

part.set\_payload(open("image.jpg", "rb").read())

encoders.encode\_base64(part)

part.add\_header('Content-Disposition', 'attachment; filename="image.jpg"') # File name and format name

msg.attach(part)

try:

s = smtplib.SMTP('smtp.gmail.com', 587) # Protocol

s.ehlo()

s.starttls()

s.ehlo()

s.login(user = 'intruderalertingsystem@gmail.com', password = 'jarvis@123') # User id & password

#s.send\_message(msg)

s.sendmail(me, toaddr, msg.as\_string())

s.quit()

#except:

# print ("Error: unable to send email")

except SMTPException as error:

print ("Error") # Exception

send\_an\_email()

**Ultrasonic Code:**

**import RPi.GPIO as GPIO**

**import time**

**TRIG=21**

**ECHO=20**

**GPIO.setmode(GPIO.BCM)**

**while True:**

**print"distance measurement in progress"**

**GPIO.setup(TRIG,GPIO.OUT)**

**GPIO.setup(ECHO,GPIO.IN)**

**GPIO.output(TRIG,False)**

**print"waiting for sensor to settle"**

**time.sleep(0.2)**

**GPIO.output(TRIG,True)**

**time.sleep(0.00001)**

**GPIO.output(TRIG,False)**

**while GPIO.input(ECHO)==0:**

**pulse\_start=time.time()**

**while GPIO.input(ECHO)==1:**

**pulse\_end=time.time()**

**pulse\_duration=pulse\_end-pulse\_start**

**distance=pulse\_duration\*17150**

**distance=round(distance,2)**

**print"distance:",distance,"cm"**

**time.sleep(2)**

**CHEATS:**

**ssh** [**pi@192.168.0.10**](mailto:pi@192.168.0.10)

password:rpi@123

Sudo reboot

Sudo halt

Sudo python ok.py

Sudo yersinia -G

Sudo ettercap -G

Sudo ifconfig

git clone <https://github.com/An0nUD4Y/blackeye>

cd blackeye

sudo ./blackeye.sh

./ngrok http 8080

<http://127.0.0.1:4040/status>

raspistill -o image.jpg

Raspivid -o testvideo.h264 -t 10000

[intruderalertingsystem@gmail.com](mailto:intruderalertingsystem@gmail.com)

jarvis@123

cd RPi\_Cam\_Web\_Interface

./install.sh

To start this service:

./start,sh

To stop this service:

./stop,sh

Camera live feed:http://192.168.0.10/

**ESPEAK**

from subprocess import call

call(["espeak","-s140 -ven+18 -z","Hello From Mike"])

**Espeak With Ultrasonic Sensor**

import RPi.GPIO as GPIO

import time

from subprocess import call

TRIG=21

ECHO=20

GPIO.setmode(GPIO.BCM)

while True:

print("distance measurement in progress")

GPIO.setup(TRIG,GPIO.OUT)

GPIO.setup(ECHO,GPIO.IN)

GPIO.output(TRIG,False)

print("waiting for sensor to settle")

time.sleep(0.2)

GPIO.output(TRIG,True)

time.sleep(0.00001)

GPIO.output(TRIG,False)

while GPIO.input(ECHO)==0:

pulse\_start=time.time()

while GPIO.input(ECHO)==1:

pulse\_end=time.time()

pulse\_duration=pulse\_end-pulse\_start

distance=pulse\_duration\*17150

distance=round(distance,2)

print("distance:",distance,"cm")

time.sleep(2)

if distance<150:

call(["espeak","-s140 -ven+18 -z","Intruder Alert"])

**PLAYING AUDIO**

import pygame

pygame.mixer.init()

pygame.mixer.music.load("siren.wav")

pygame.mixer.music.play()

while pygame.mixer.music.get\_busy() == True:

continue

**Espeak and Siren With Ultrasonic Sensor**

import RPi.GPIO as GPIO

import time

from subprocess import call

import pygame

TRIG=21

ECHO=20

GPIO.setmode(GPIO.BCM)

while True:

print("distance measurement in progress")

GPIO.setup(TRIG,GPIO.OUT)

GPIO.setup(ECHO,GPIO.IN)

GPIO.output(TRIG,False)

print("waiting for sensor to settle")

time.sleep(0.2)

GPIO.output(TRIG,True)

time.sleep(0.00001)

GPIO.output(TRIG,False)

while GPIO.input(ECHO)==0:

pulse\_start=time.time()

while GPIO.input(ECHO)==1:

pulse\_end=time.time()

pulse\_duration=pulse\_end-pulse\_start

distance=pulse\_duration\*17150

distance=round(distance,2)

print("distance:",distance,"cm")

time.sleep(2)

if distance<150:

call(["espeak","-s140 -ven+18 -z","Intruder Alert"])

pygame.mixer.init()

pygame.mixer.music.load("siren.wav")

pygame.mixer.music.play()

while pygame.mixer.music.get\_busy() == True:

continue

**Ultrasonic Sensor and Mail Code:**

**import smtplib,ssl**

**from picamera import PiCamera**

**from time import sleep**

**from email.mime.multipart import MIMEMultipart**

**from email.mime.base import MIMEBase**

**from email.mime.text import MIMEText**

**from email.utils import formatdate**

**from email import encoders**

**import RPi.GPIO as GPIO**

**import time**

**TRIG=21**

**ECHO=20**

**GPIO.setmode(GPIO.BCM)**

**while True:**

**print("distance measurement in progress")**

**GPIO.setup(TRIG,GPIO.OUT)**

**GPIO.setup(ECHO,GPIO.IN)**

**GPIO.output(TRIG,False)**

**print("waiting for sensor to settle")**

**time.sleep(0.2)**

**GPIO.output(TRIG,True)**

**time.sleep(0.00001)**

**GPIO.output(TRIG,False)**

**while GPIO.input(ECHO)==0:**

**pulse\_start=time.time()**

**while GPIO.input(ECHO)==1:**

**pulse\_end=time.time()**

**pulse\_duration=pulse\_end-pulse\_start**

**distance=pulse\_duration\*17150**

**distance=round(distance,2)**

**print("distance:",distance,"cm")**

**time.sleep(2)**

**if distance<100:**

**camera = PiCamera()**

**camera.start\_preview()**

**sleep(5)**

**camera.capture('/home/pi/image.jpg') # image path set**

**sleep(5)**

**camera.stop\_preview()**

**def send\_an\_email():**

**toaddr = 'intruderalertingsystem@gmail.com' # To id**

**me = 'intruderalertingsystem@gmail.com' # your id**

**subject = "What's News" # Subject**

**msg = MIMEMultipart()**

**msg['Subject'] = subject**

**msg['From'] = me**

**msg['To'] = toaddr**

**msg.preamble = "test "**

**#msg.attach(MIMEText(text))**

**part = MIMEBase('application', "octet-stream")**

**part.set\_payload(open("image.jpg", "rb").read())**

**encoders.encode\_base64(part)**

**part.add\_header('Content-Disposition', 'attachment; filename="image.jpg"') # File name and format name**

**msg.attach(part)**

**try:**

**s = smtplib.SMTP('smtp.gmail.com', 587) # Protocol**

**s.ehlo()**

**s.starttls()**

**s.ehlo()**

**s.login(user = 'intruderalertingsystem@gmail.com', password = 'jarvis@123') # User id & password**

**#s.send\_message(msg)**

**s.sendmail(me, toaddr, msg.as\_string())**

**s.quit()**

**#except:**

**# print ("Error: unable to send email")**

**except SMTPException as error:**

**print ("Error") # Exception**

**send\_an\_email()**

**Print Fire:**

import RPi.GPIO as GPIO

import time

#GPIO SETUP

channel = 21

GPIO.setmode(GPIO.BCM)

GPIO.setup(channel, GPIO.IN)

def callback(channel):

print("flame detected")

GPIO.add\_event\_detect(channel, GPIO.BOTH, bouncetime=300) # let us know when the pin goes HIGH or LOW

GPIO.add\_event\_callback(channel, callback) # assign function to GPIO PIN, Run function on change

# infinite loop

while True:

time.sleep(1)

**Print Fire with Espeak:**

import RPi.GPIO as GPIO

import time

from subprocess import call

import pygame

#GPIO SETUP

channel = 20

GPIO.setmode(GPIO.BCM)

GPIO.setup(channel, GPIO.IN)

def callback(channel):

print("flame detected")

call(["espeak","-s140 -ven+18 -z","Fire Detected"])

GPIO.add\_event\_detect(channel, GPIO.BOTH, bouncetime=300) # let us know when the pin goes HIGH or LOW

GPIO.add\_event\_callback(channel, callback) # assign function to GPIO PIN, Run function on change

# infinite loop

while True:

time.sleep(1)

**Flame Ultrasonic Mail:**

**import smtplib,ssl**

**from picamera import PiCamera**

**from time import sleep**

**from email.mime.multipart import MIMEMultipart**

**from email.mime.base import MIMEBase**

**from email.mime.text import MIMEText**

**from email.utils import formatdate**

**from email import encoders**

**import RPi.GPIO as GPIO**

**from subprocess import call**

**import time**

**TRIG=21**

**ECHO=18**

**channel = 20**

**GPIO.setmode(GPIO.BCM)**

**while True:**

**GPIO.setup(channel, GPIO.IN)**

**def callback(channel):**

**print("flame detected")**

**call(["espeak","-s140 -ven+18 -z","Fire Detected"])**

**GPIO.add\_event\_detect(channel, GPIO.BOTH, bouncetime=300) # let us know when the pin goes HIGH or LOW**

**GPIO.add\_event\_callback(channel, callback) # assign function to GPIO PIN, Run function on change**

**distance=2**

**if distance<100:**

**camera = PiCamera()**

**camera.start\_preview()**

**sleep(5)**

**camera.capture('/home/pi/Downloads/image.jpg') # image path set**

**sleep(5)**

**camera.stop\_preview()**

**def send\_an\_email():**

**toaddr = 'intruderalertingsystem@gmail.com' # To id**

**me = 'intruderalertingsystem@gmail.com' # your id**

**subject = "What's News" # Subject**

**msg = MIMEMultipart()**

**msg['Subject'] = subject**

**msg['From'] = me**

**msg['To'] = toaddr**

**msg.preamble = "test "**

**#msg.attach(MIMEText(text))**

**part = MIMEBase('application', "octet-stream")**

**part.set\_payload(open("image.jpg", "rb").read())**

**encoders.encode\_base64(part)**

**part.add\_header('Content-Disposition', 'attachment; filename="image.jpg"') # File name and format name**

**msg.attach(part)**

**try:**

**s = smtplib.SMTP('smtp.gmail.com', 587) # Protocol**

**s.ehlo()**

**s.starttls()**

**s.ehlo()**

**s.login(user = 'intruderalertingsystem@gmail.com', password = 'jarvis@123') # User id & password**

**#s.send\_message(msg)**

**s.sendmail(me, toaddr, msg.as\_string())**

**s.quit()**

**#except:**

**# print ("Error: unable to send email")**

**except SMTPException as error:**

**print ("Error") # Exception**

**send\_an\_email()**

**import smtplib,ssl**

**from picamera import PiCamera**

**from time import sleep**

**from email.mime.multipart import MIMEMultipart**

**from email.mime.base import MIMEBase**

**from email.mime.text import MIMEText**

**from email.utils import formatdate**

**from email import encoders**

**import RPi.GPIO as GPIO**

**from subprocess import call**

**import time**

**TRIG=21**

**ECHO=18**

**channel = 20**

**GPIO.setmode(GPIO.BCM)**

**while True:**

**GPIO.setup(channel, GPIO.IN)**

**def callback(channel):**

**print("flame detected")**

**call(["espeak","-s140 -ven+18 -z","Fire Detected"])**

**GPIO.add\_event\_detect(channel, GPIO.BOTH, bouncetime=300) # let us know when the pin goes HIGH or LOW**

**GPIO.add\_event\_callback(channel, callback) # assign function to GPIO PIN, Run function on change**

**print("distance measurement in progress")**

**GPIO.setup(TRIG,GPIO.OUT)**

**GPIO.setup(ECHO,GPIO.IN)**

**GPIO.output(TRIG,False)**

**print("waiting for sensor to settle")**

**time.sleep(0.2)**

**GPIO.output(TRIG,True)**

**time.sleep(0.00001)**

**GPIO.output(TRIG,False)**

**while GPIO.input(ECHO)==0:**

**pulse\_start=time.time()**

**while GPIO.input(ECHO)==1:**

**pulse\_end=time.time()**

**pulse\_duration=pulse\_end-pulse\_start**

**distance=pulse\_duration\*17150**

**distance=round(distance,2)**

**print("distance:",distance,"cm")**

**time.sleep(2)**

**distance=2**

**if distance<100:**

**camera = PiCamera()**

**camera.start\_preview()**

**sleep(5)**

**camera.capture('/home/pi/Downloads/image.jpg') # image path set**

**sleep(5)**

**camera.stop\_preview()**

**def send\_an\_email():**

**toaddr = 'intruderalertingsystem@gmail.com' # To id**

**me = 'intruderalertingsystem@gmail.com' # your id**

**subject = "What's News" # Subject**

**msg = MIMEMultipart()**

**msg['Subject'] = subject**

**msg['From'] = me**

**msg['To'] = toaddr**

**msg.preamble = "test "**

**#msg.attach(MIMEText(text))**

**part = MIMEBase('application', "octet-stream")**

**part.set\_payload(open("image.jpg", "rb").read())**

**encoders.encode\_base64(part)**

**part.add\_header('Content-Disposition', 'attachment; filename="image.jpg"') # File name and format name**

**msg.attach(part)**

**try:**

**s = smtplib.SMTP('smtp.gmail.com', 587) # Protocol**

**s.ehlo()**

**s.starttls()**

**s.ehlo()**

**s.login(user = 'intruderalertingsystem@gmail.com', password = 'jarvis@123') # User id & password**

**#s.send\_message(msg)**

**s.sendmail(me, toaddr, msg.as\_string())**

**s.quit()**

**#except:**

**# print ("Error: unable to send email")**

**except SMTPException as error:**

**print ("Error") # Exception**

**send\_an\_email()**

**Complete:**

import smtplib,ssl

from picamera import PiCamera

from time import sleep

from email.mime.multipart import MIMEMultipart

from email.mime.base import MIMEBase

from email.mime.text import MIMEText

from email.utils import formatdate

from email import encoders

import RPi.GPIO as GPIO

from subprocess import call

import time

TRIG=21

ECHO=18

channel = 20

GPIO.setmode(GPIO.BCM)

while True:

GPIO.setup(channel, GPIO.IN)

def callback(channel):

print("flame detected")

call(["espeak","-s140 -ven+18 -z","Fire Detected"])

GPIO.add\_event\_detect(channel, GPIO.BOTH, bouncetime=300) # let us know when the pin goes HIGH or LOW

GPIO.add\_event\_callback(channel, callback) # assign function to GPIO PIN, Run function on change

print("distance measurement in progress")

GPIO.setup(TRIG,GPIO.OUT)

GPIO.setup(ECHO,GPIO.IN)

GPIO.output(TRIG,False)

print("waiting for sensor to settle")

time.sleep(0.2)

GPIO.output(TRIG,True)

time.sleep(0.00001)

GPIO.output(TRIG,False)

while GPIO.input(ECHO)==0:

pulse\_start=time.time()

while GPIO.input(ECHO)==1:

pulse\_end=time.time()

pulse\_duration=pulse\_end-pulse\_start

distance=pulse\_duration\*17150

distance=round(distance,2)

print("distance:",distance,"cm")

time.sleep(2)

if distance<100:

camera = PiCamera()

camera.start\_preview()

sleep(5)

camera.capture('/home/pi/Downloads/image.jpg') # image path set

sleep(5)

camera.stop\_preview()

def send\_an\_email():

toaddr = 'intruderalertingsystem@gmail.com' # To id

me = 'intruderalertingsystem@gmail.com' # your id

subject = "Intruder Alert" # Subject

msg = MIMEMultipart()

msg['Subject'] = subject

msg['From'] = me

msg['To'] = toaddr

msg.preamble = "test "

#msg.attach(MIMEText(text))

part = MIMEBase('application', "octet-stream")

part.set\_payload(open("image.jpg", "rb").read())

encoders.encode\_base64(part)

part.add\_header('Content-Disposition', 'attachment; filename="image.jpg"') # File name and format name

msg.attach(part)

try:

s = smtplib.SMTP('smtp.gmail.com', 587) # Protocol

s.ehlo()

s.starttls()

s.ehlo()

s.login(user = 'intruderalertingsystem@gmail.com', password = 'jarvis@123') # User id & password

#s.send\_message(msg)

s.sendmail(me, toaddr, msg.as\_string())

s.quit()

#except:

# print ("Error: unable to send email")

except SMTPException as error:

print ("Error") # Exception

send\_an\_email()

**FIRE ULTRASONICSENSOR Espeak Siren :**

import RPi.GPIO as GPIO

import time

from subprocess import call

import pygame

TRIG=21

ECHO=20

#GPIO SETUP

channel = 18

GPIO.setmode(GPIO.BCM)

GPIO.setup(channel, GPIO.IN)

GPIO.setwarnings(False)

while True:

print("distance measurement in progress")

GPIO.setup(TRIG,GPIO.OUT)

GPIO.setup(ECHO,GPIO.IN)

GPIO.output(TRIG,False)

print("waiting for sensor to settle")

time.sleep(0.2)

GPIO.output(TRIG,True)

time.sleep(0.00001)

GPIO.output(TRIG,False)

while GPIO.input(ECHO)==0:

pulse\_start=time.time()

while GPIO.input(ECHO)==1:

pulse\_end=time.time()

pulse\_duration=pulse\_end-pulse\_start

distance=pulse\_duration\*17150

distance=round(distance,2)

print("distance:",distance,"cm")

time.sleep(2)

if(GPIO.input(18)==False):

print("flame detected")

call(["espeak","-s140 -ven+18 -z","Fire Detected"])

if distance<50:

call(["espeak","-s140 -ven+18 -z","Intruder Alert"])

pygame.mixer.init()

pygame.mixer.music.load("siren.wav")

pygame.mixer.music.play()

while pygame.mixer.music.get\_busy() == True:

continue