**IOT PROJECT (SG5233)**

**VIRTUAL PERSONAL ASSISSTANT**

**Objective:**

To integrate Amazon’s Echo (Alexa) with our Raspberry pi to control the GPIO pins and switch on/off the output device (relay)

***CODE TO INVOKE ALEXA (AMAZON ECHO)***

Create an account in developer.amazon.com and get the product ID, client ID and client secret key.

To download the Alexa kit,

cd Desktop

git clone https://github.com/alexa/alexa-avs-sample-app.git

cd ~/Desktop/alexa-avs-sample-app

nano automated\_install.sh

#A text editor opens where we enter our product ID, client ID and client secret key in the designated fields.

cd ~/Desktop/alexa-avs-sample-app

.automated\_install.sh

#This will configure our PI and install some extra software

**Run Alexa Service**

In a new terminal,

Cd ~/Desktop/alexa-avs-sample-app/samples

Cd companionService && npm start

#This starts the companion service and opens up a port to communicate with Amazon. Leave this window open.

In a new terminal,

Cd ~/Desktop/alexa-avs-sample-app/samples

Cd javaclient && mvn exec:exec

#A java app opens up with tokens to communicate with the Alexa service

In another terminal,

Cd ~/Desktop/alexa-avs-sample-app/samples

Cd wakeWordAgent/src && ./wakeWordAgent -e kitt\_ai

#Now our alexa service is up and running.

***CODE TO INTEGRATE AMAZON ECHO WITH RELAY***

# commands to install packages and python libraries.

sudo apt-get update && sudo apt-get upgrade -y

sudo apt-get install python 2.7 -dev python-dev python-pip

sudo pip install Flask flask-ask

# Download Linux ARM from <https://ngrok.com/download> and unzip the file

unzip /home/pi/ngrok-stable-linux-arm.zip

# Run ngrok

sudo ./ngrok http 5000

# Note down the forwarding URL since the URL changes each time we start this service

#Open a new terminal and create a python file named gpio\_control.py

Nano gpio\_control.py

#Paste the following code into it:

from flask import Flask

from flask\_ask import Ask, statement, convert\_errors

import RPi.GPIO as GPIO

import logging

GPIO.setmode(GPIO.BCM)

app = Flask(\_\_name\_\_)

ask = Ask (app, '/')

logging.getLogger("flask\_ask").setLevel(logging.DEBUG)

@ask.intent('GPIOControlIntent', mapping={'status': 'status', 'pin': 'pin'})

def gpio\_control(status, pin):

try:

pinNum = int(pin)

except Exception as e:

return statement('Pin number not valid.')

GPIO.setup(pinNum, GPIO.OUT)

if status in ['on', 'high']: GPIO.output(pinNum, GPIO.HIGH)

if status in ['off', 'low']: GPIO.output(pinNum, GPIO.LOW)

return statement('Turning pin {} {}'.format(pin, status))

#End of Code

#start the flask server

Sudo python gpio\_control.py

#Leave the terminal open and running. Make sure both the flask server and the ngrok service is running.

#Set up a new alexa skill in developer.amazon.com

#Paste this code in the intent schema field:

{

"intents": [{

"intent": "GPIOControlIntent",

"slots": [{

"name": "status",

"type": "GPIO\_CONTROL"

},

{

"name": "pin",

"type": "AMAZON.NUMBER"

}]

}]

}

#we can now test Alexa by saying:

‘Alexa, ask/tell raspberry pi to turn pin twenty-one (any pin that is connected) on/off’