TARS WIZARD Guide

A comprehensive code guide on how to deal with electronics and technical stuff.

Follow me on socials:

Instagram: https://www.instagram.com/wizard.py/

LinkedIn: https://www.linkedin.com/in/Dhruv-Goswami-24-/

Table of Contents

Raspberry Pi 5 connections 3

How to get AWS Polly API key (For TARS Voice):..... 4

How to get OpenAI API key: 5

How to get Weather API: 5

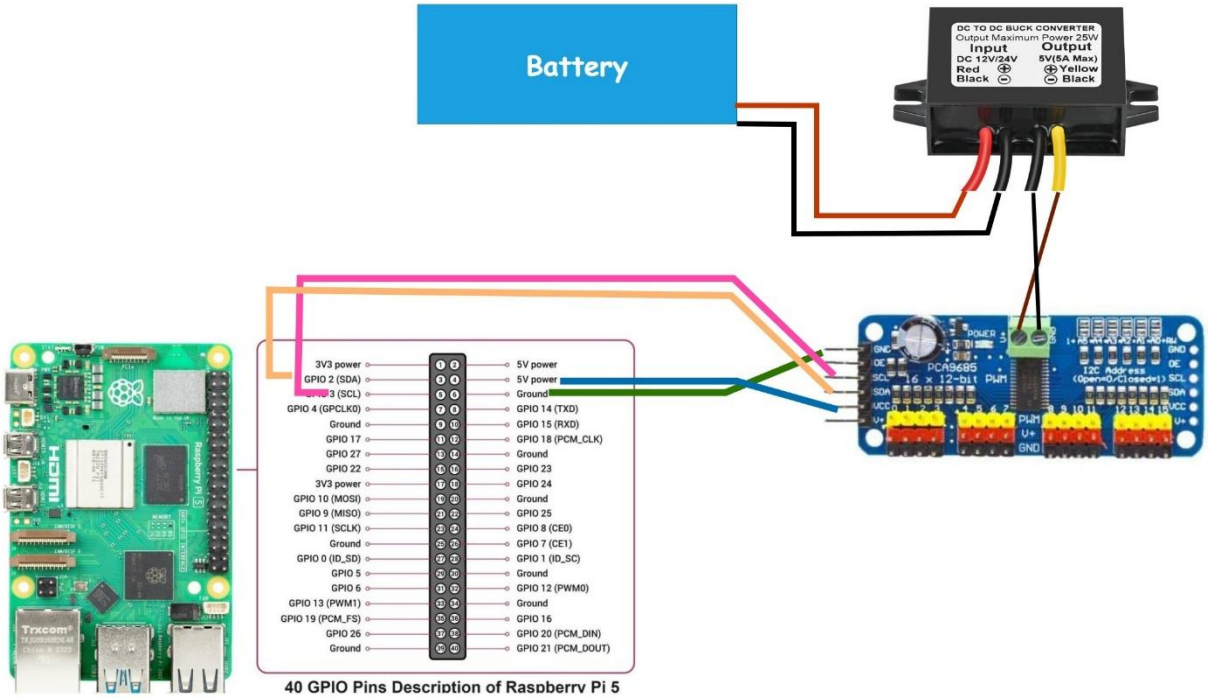
Libraries and Requirements to be installed:..... 6

How to Run the script on Raspberry Pi: 7

How to surpass ALSA logs (Bunch of gibberish appearing on screen when running script): 7

How to connect controller to raspberry pi (For movements using controller):..... 8

Raspberry Pi 5 connections



On this Servo Controller shown below only connect servos as follows:

Body servo on channel = 0

Left arm servo on channel = 3

Right arm servo on channel = 4

How to get AWS Polly API key (For TARS Voice):

Step-1: Sign up for AWS

1. Go to the AWS Management Console
2. If you don't have AWS account, click on Create and AWS account and follow the instructions to create one. If you already have one, sign in.

Step-2: Create an IAM User with Programmatic Access

1. In AWS management console, go it IAM services b searching for "IAM" in search bar.
2. In left hand menu, click "Users", then click "Add users" button.
3. Enter a username, e.g., "polly-user'
4. Under "Select AWS access type", check "Programmatic access".
5. Click Next: Permissions.

Step-3: Attach a policy to the user

1. On "Set permissions" page, choose "Attach policies directly".
2. Search for the "AmazonPollyFullAccess" policy and check the box next to it.
3. Click next: Tags (optional), Then click next: Review
4. Review the settings and click "Create user".

Step-4: Download Access Key and Secret Key

1. After user is created, you wil see the "Access ID" and "Secret access key".
2. Click Download .csv to save these keys securely. NOTE: This is the only time you can download the keys so keep them SAFE!!!!!

Now you can use those ACCESS KEY and SECRET KEY in the code to make TARS talk.

How to get OpenAI API key:

Step-1: Sign up for OpenAI

1. Go to OpenAI Platform
2. Sign up if you don't have an account or login if you already have an existing account. You can use your ChatGPT's credentials to Login no need to sign up.

Step-2: Get API Key

1. Go to Dashboard.
2. In left hand menu click on "API Keys".
3. Click on "Create new secret key" button which is on top right corner.
4. Write name of secret key, e.g., "TARS"
5. Select "Default project" or if you have any specific project already created for TARS, select that.
6. Select "All" in Permissions.
7. Finally click on "Create Secret Key".
8. Copy and paste this key somewhere secure and safe because this is the only time you will be able to see and access your API Key.

Now you can use this key in the code to ask any burning question to TARS.

How to get Weather API:

Step-1: Sign in or create an account

1. **Go to Weather API website**
2. **Sign in or create an account**

Step-2: Generate API Key

1. **Click on API Keys**
2. **Enter key name and click on "Generate button"**

Use this key on your code to access weather data of your current location.

Libraries and Requirements to be installed:

Use “spip” to install all these libraries and requirements using terminal on your Raspberry Pi

Things to install:

- **SpeechRecognition==3.10 # For speech-to-text functionality**
- **openai==0.27.8 # For ChatGPT API integration**
- **boto3==1.28.66 # For AWS services integration (e.g., Amazon Polly or S3)**
- **requests==2.31.0 # For making HTTP requests**
- **pydub==0.25.1 # For audio processing (e.g., playback and normalization)**

Some libraries like pydub and speechRecognition will require dependencies to be installed separately:

- FFmpeg for pydub
  sudo apt-get install ffmpeg

- ALSA and PortAudio for speechRecognition
  sudo apt-get install alsa-utils portaudio19-dev

Based on the imports on the code:

```
- import speech_recognition as sr
- import openai
- import boto3
- import requests
- from pydub import AudioSegment, effects
- import io
- import re
- from datetime import datetime
- from pydub.playback import play
- import threading
- import sys
- import time
- import mechanism  # Import the movement module
```

If there are any questions or issues seek support from forms page on the website.

How to Run the script on Raspberry Pi:

Step-1: Navigate to the project Directory and activate virtual environment

Source /your/folder/path/env/bin/activate

Step-3: Run the python script

python3 tars_main.py (With ALSA Logs)

OR

python3 tars_main.py 2>/dev/null (surpassing ALSA Logs)

NOTE: Before connecting Servos to servo arms for legs, run script and calibrate the servo such that it is set to 0 degree rotation. In this way the legs will be aligned properly.

How to surpass ALSA logs (Bunch of gibberish appearing on screen when running script):

Simply run the following in your terminal. This line will remove the gibberish and your terminal will look clean with only listening and responses on your screen.

COMMAND => python3 your_script.py 2> /dev/null

How to connect controller to raspberry pi (For movements using controller):

Step-1: Find your connected controller through terminal

- First connect controller via Bluetooth to raspberry pi
- Next open terminal and find your controller using following command:

  Command: ls /dev/inputs/event*

  This will give you list of all events (Wired or wireless devices) on your pi you will then have to manually select events using command: cat /dev/input/eventX

  Then read details if it says your controller name and test inputs by pressing buttons on controller, you should see random gibberish appear on button press means it is working.

  OR (Recommended method below for ease)

  Install evtest by: sudo apt install evtest OR pip install evtest

  Then use evtest: sudo evtest OR python -m evtest

  A list of events with name of connected devices will appear. Select your controller by typing event number, e.g., If “event13” is your controller then just type 13 and then check your inputs by pressing buttons.

- Now remember the event number, e.g., “event17” and cross check the event value on code and see if it matches. If it does not match simple change the event value to your corresponding value, e.g., if it is written “event17” on code and yours is “event2” change “event17” to “event2”.

- NOTE: The buttons might be different depending on your controller. Xbox, PS, 8 bit-do or other controllers all will have different input values. e.g., Pressing X on 8 bit-do controller will make tars move forward but pressing X on Xbox controller might make him turn right.