Ha Tran

EE104

05/07/2023

Lab 8 Documentation

In this documentation, I will go over the OpenAI File Q&A which the information from the files we keep in the directory will be used by the OpenAI chat bot to answer questions about those files. The second one is about OpenAI Web Crawl Q&A. The last one is hardware part, it is about Traffic Controller.

1)For the OpenAI File Q&A, I have to set up an OpenAI key based on the web was provided on the instruction and then, open Powershell window to install all package. Besides that, I also have to set up Pinecone API key and pip install openai. The assignment requires me should create file as “.env” include:

OPEN\_API\_KEY = "YOUR\_API\_KEY"

YOUR OPENAI ORG KEY = "YOUR\_OPENAI\_ORG\_KEY"

PINECONE\_API\_KEY="YOUR\_PINECONE\_API\_KEY"

PINECONE\_ENVIRONMENT="Your\_environment"

Next, I have to open two separate Powershell windows and do both Server and Client setup in parellel to save time.

Create a file text include all the questions and answers for each questions as shown below:

Text

Description automatically generated

Open http://localhost:3000 with your browser to see the app, then type the question:

Graphical user interface, text, application, email

Description automatically generated

2) For the OpenAI Web Crawl Q&A, I have to set up an OpenAI key based on the web was provided on the instruction and then, open Powershell window to install all package. Afterward, I need to visit their API keys page to retrieve their API key and create a folder. Once the folder is created, the user needs to create a ".env" file and type "OPEN\_API\_KEY = 'YOUR\_API\_KEY'" into it, then save the file as "all type" with file name ".env".

Graphical user interface, application, Word

Description automatically generated

Next, I should run "pip install -r requirements.txt" to install all necessary packages. Afterward, the user should open the file "web-qa.py" and replace the domain and URL with their own. The user can then run the file and crawl the website to build the CSV file.

Graphical user interface, text, application, email

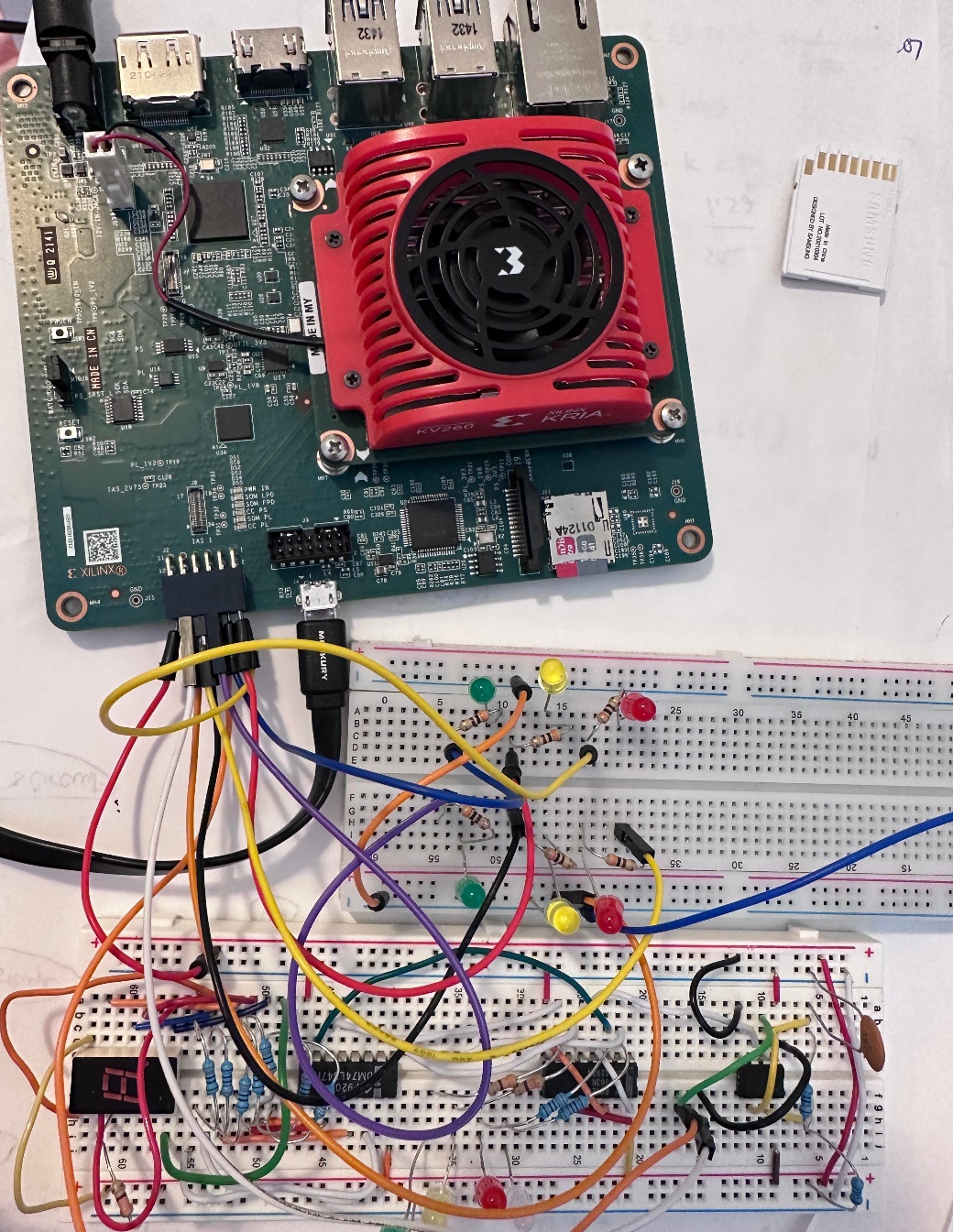
Description automatically generated

A screenshot of a computer

Description automatically generated

3) Traffic Controller:

For the circuit, we replaced the whole state diagram part with the python and the Kria board. And you can clearly see that there are two ways on the breadboard.



For the software contribution, first, we set the whole circuit to first state, and the code is as below:

# 1st state

led4.write(1)

led3.write(1)

Then we wrote the second state to fourth state:

# 2nd state to 3rd state - When passenger press the button

from time import sleep

DELAY = 1.5

led4.write(0)

led3.write(1)

led5.write(1)

sleep(DELAY)

led5.write(0)

led6.write(1)

sleep(DELAY)

led3.write(0)

led1.write(1)

CountDown.write(1)

# 4th state

sleep(5)

led1.write(0)

led2.write(1)

CountDown.write(0)

sleep(DELAY)

led2.write(0)

led3.write(1)

sleep(DELAY)

led4.write(1)

led3.write(1)

led6.write(0)

From the code above, if we run it, it is like when the pedestrian presses the crossroad button. The traffic light on the car side will turn from green to yellow then to red, and the pedestrian side’s light will turn green. And the 7 segments will start counting down, and after a while, the light on the pedestrian side will start to turn yellow and red.

OpenAI FileQ&A and Web Crawl Q&A Youtube link: <https://youtu.be/-5D5L8o9sO8>

Our Hardware Traffic Controller link: <https://www.youtube.com/watch?v=dyUuvyeEQLA>