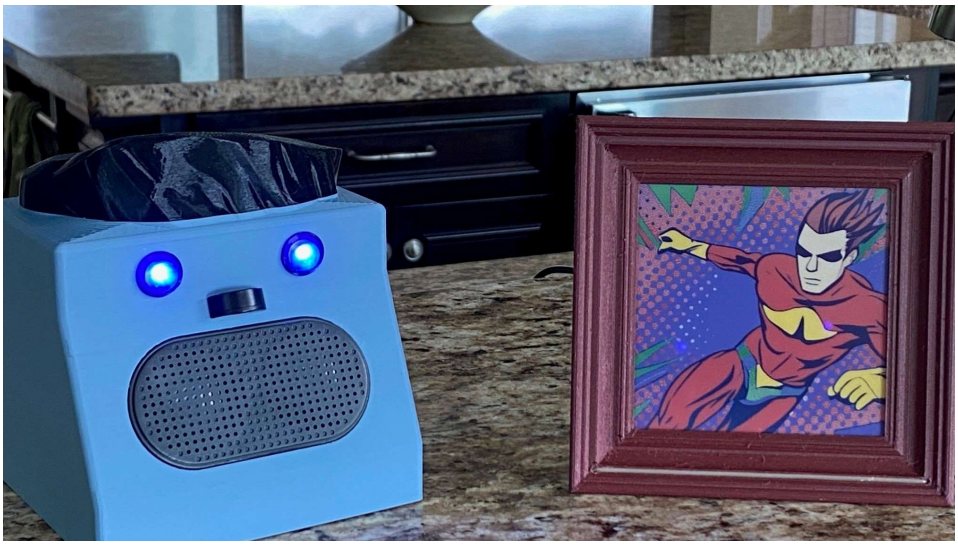
# DaVinci – The AI Artist



DaVinci is now an AI artist!  In addition to being a talking ChatGPT Virtual Assistant, DaVinci can now generate images using DALL-E and wirelessly send them to your AI E-Ink Art Frame for display based upon your voice command.

The multimodal setup uses PicoVoice Porcupine, Cobra and Leopard for wake word and voice detection and speech to text, OpenAI's GPT 3.5 and DALL-E for artificial intelligence responses and image generation, and Amazon Polly for text to speech.  The AI Art Frame wirelessly communicates with DaVinci using the MQTT protocol and serves as the MQTT broker.

A brief video demo is viewable here: https://youtu.be/GULQDLpVK0w

**Instructions for How to Make Your Own**

The parts list, python code, STLs and detailed instructions on how to build your own DaVinci ChatGPT Virtual Assistant and DALL-E Voice-Controlled Artificial Intelligence Art Frame are available on this GitHub site.  Changes to those instructions for the multimodal versions are below.

**DaVinci Multimodal Build**

If you have not already built your DaVinci, do so using the instructions here: https://github.com/DevMiser/DaVinci and verify that it is working properly.

To make DaVinci an AI artist, you simply change out the DaVinci.py code with the DaVinciTheArtist.py code in this repository and make the additional changes described below to enable the MQTT protocol.

**Download and Enable MQTT on DaVinci**

Open a terminal and enter the following commands in the following order (*note that mosquitto, unlike the annoying bug, is spelled with two t's everywhere it is used*):

sudo apt-get install mosquitto

sudo systemctl enable mosquitto.service

pip3 install paho-mqtt

We will make a few additional changes to DaVinci the AI artist, including loading the DaVinciTheArtist.py code, after we complete the AI Art Frame MQTT build.

**AI Art Frame MQTT Build**

The MQTT version of the AI Art Frame is simpler than the voice-controlled version because no voice detection hardware or software is needed. While the physical build for the AI Art Frame is mostly the same, the set up and code are different as described below.

If you have not already built your AI Art Frame, you should familiarize yourself with the instructions here: https://github.com/DevMiser/AI_Art_Frame. But for building the MQTT version of the AI Art Frame, you will not need or use the microphone, and you can use a less expensive Raspberry Pi 3 or Raspberry Pi Zero 2 instead of a Raspberry Pi 4. The rest of the physical aspect of the build is the same, but you should follow the instructions below for preparing your Raspberry Pi instead of the original instructions:

1. Edit the bashrc file on your Raspberry Pi as follows:

a. Open a terminal and enter the following command to open the bashrc file:

```
sudo nano ~/.bashrc
```

b. Scroll to the bottom of the file using your keyboard and add the following lines at the end (be certain to include the #s):

```
# sets a location where the Raspberry Pi OS and Python can look for
# executable/configuration files
export PATH="$HOME/.local/bin:$PATH"
```

c. Press the CTRL and X keys simultaneously on your keyboard, then press Y and then press Enter to save the revisions to the file.

d. Then enter the following command:

```
sudo reboot
```

This will reboot your Raspberry Pi. After the reboot is completed, log back in.]

2. Open a terminal and enter the following commands in the following order:

```
sudo apt update
```

sudo apt full-upgrade - *If asked if you want to continue, enter Y and then press Enter*

pip3 install --upgrade pip

pip3 install schedule

pip3 install inky[rpi,example-depends]

sudo apt-get install mosquitto - *If asked if you want to continue, enter Y and then press Enter*

sudo systemctl enable mosquitto.service

pip3 install paho-mqtt

sudo reboot – *this will reboot your Raspberry Pi; log back in after the reboot*.

3. Enable the SPI and I2C interfaces as follows:

a. Open a terminal and enter the following command:

sudo raspi-config

b. Using the arrows on your keyboard, scroll down to “3 Interface Options” and press Enter

c. Scroll down to “I4 SPI” and press Enter

d. The next screen will ask “Would you like the SPI interface to be enabled?” Select “Yes” and press Enter.

e. The next screen will state “The SPI interface in enabled”. Select “Ok” and press Enter.

f. Scroll down to “3 Interface Options” and press Enter

g. Scroll down to “I5 I2C” and press Enter

h. The next screen will ask “Would you like the ARM I2C interface to be enabled?” Select “Yes” and press Enter.

i. The next screen will state “The ARM I2C interface is enabled”. Select “Ok” and press Enter.

j. Scroll down to select “Finish” and press Enter

4. Obtain the IP Address of your AI Art Frame's Raspberry Pi by opening a terminal and entering the following command:

```
hostname -I
```

The IP address will be the first set of numbers shown – something like XXX.XXX.X.XXX. Write down the IP address because you will need it for changes to both devices.

5. Download the AIArtFrameDaVinciSubscriber.py program and associated files by opening a terminal and entering the following command:

```
git clone https://github.com/DevMiser/DaVinci_The_Artist.git
```

6. Modify AIArtFrameDaVinciSubscriber.py to include a password for communicating with DaVinci and to add the Art Frame's IP address by doing the following:

a. Open a terminal and enter the following commands:

```
cd /home/pi/DaVinci_The_Artist
```

b. Use your keyboard to scroll down close to the end of the program to the line that says:

```
client.username_pw_set("DaVinci", "PlaceYourPasswordHere")
```

Now modify that line to replace PlaceYourPasswordHere with a password of your own choosing. Write down the password because you will use it again later.

c. Use your keyboard to scroll down a little further to the line that says:

```
client.connect("PlaceYourArtFrameIPAddressHere", 1883, 60)
```

Now modify that line to replace PlaceYourArtFrameIPAddressHere with the IP address of your AI Art Frame's Raspberry Pi that you wrote down earlier.

d. Press the CTRL and X keys simultaneously on your keyboard, then press Y and then press Enter to save the revisions to the file.

**More Changes to DaVinci**

1. Now return to your DaVinci. Download the DaVinciTheArtist.py program and associated files by opening a terminal and entering the following command:

```
git clone https://github.com/DevMiser/DaVinci_The_Artist.git
```

2. Modify DaVinciTheArtist.py to include the secret OpenAI API key and your secret PicoVoice access key (these will be the same keys that you created when you originally set up your DaVinci), to add the password for communicating with the AI Art Frame and to add the Art Frame's IP address by doing the following:

a. Open a terminal and enter the following commands:

cd /home/pi/DaVinci_The_Artist

sudo nano DaVinciTheArtist.py

b. Use your keyboard to scroll down to the lines that say:

openai.api_key = "*put your secret API key between these quotation marks*"
pv_access_key= "*put your secret access key between these quotation marks*"

Now modify those lines to replace the portions that are italicized above with your secret OpenAI API key and your secret Picovoice access key, respectively.

c. Use your keyboard to scroll down close to the end of the program to the lines that say:

auth = {'username':"DaVinci", 'password':"PlaceYourPasswordHere"}
publish.single("AI-Art-Frame", "Clean", hostname="PlaceYourArtFrameIPAddressHere")

Now modify the first line to replace PlaceYourPasswordHere with the password that you wrote down earlier and modify the second line to replace PlaceYourArtFrameIPAddressHere with the IP address of your AI Art Frame's Raspberry Pi that you wrote down earlier.

d. Now scroll down further to where you will find the exact same lines again and modify them in the exact same way.

e. Press the CTRL and X keys simultaneously on your keyboard, then press Y and then press Enter to save the revisions to the file.

**More Changes to Both the DaVinci and the Art Frame**

You will use the following steps for both your DaVinci and your AI Art Frame. Perform the following steps on your DaVinci first:

1. Edit the mosquitto.conf file on your Raspberry Pi as follows:

a. Open a terminal and enter the following command to open the file:

sudo nano /etc/mosquitto/mosquitto.conf

b. Scroll to the bottom of the file using your keyboard and add the following lines at the end:

allow_anonymous true

listener 1883

c. Press the CTRL and X keys simultaneously on your keyboard, then press Y and then press Enter to save the revisions to the file.

d. Then enter the following command:

sudo reboot - *this will reboot your Raspberry Pi; log back in after the reboot*

2. Now return to your AI Art Frame and perform the exact same steps as in 1 immediately above.

**Run the Programs**

To run the programs, do the following:

On the AI Art Frame, open a terminal and enter the following commands:

cd /home/pi/DaVinci_The_Artist

python3 AIArtFrameDaVinciSubscriber.py

On the DaVinci, open a terminal and enter the following commands:

cd /home/pi/DaVinci_The_Artist

python3 DaVinciMultimodal.py

You can now speak to your DaVinci, and it will respond as normal. If you want DaVinci to generate an image and send it to your AI Art Frame, begin your request with any of the following:

| | | | |
|---|---|---|---|
| *Draw…* | *Create an image…* | *Create a drawing…* | *Make a painting…* |
| *Paint…* | *Create a painting…* | *Make an image…* | *Make a drawing…* |