Seuss Display Prototype

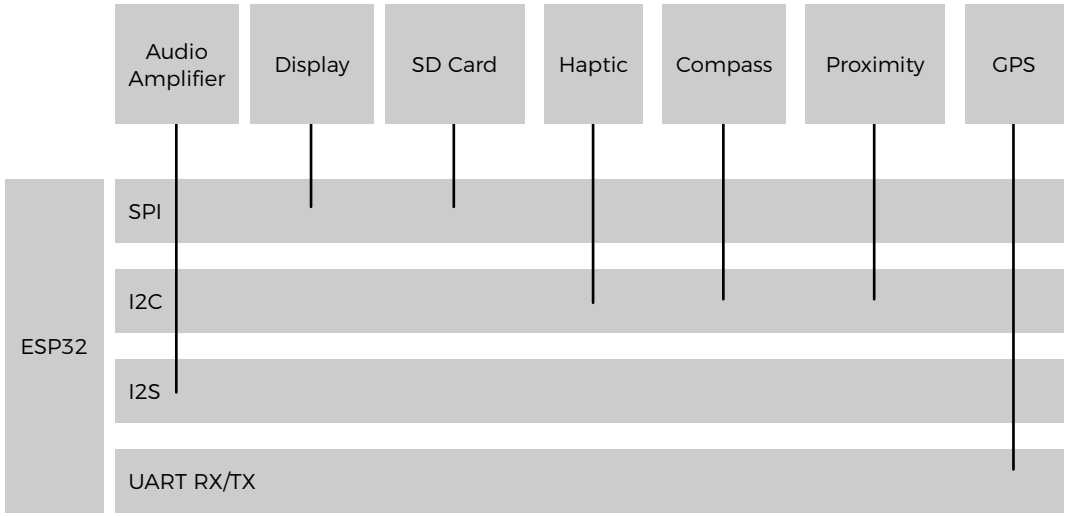
Version: March 24, 2021

fcohen@starlingwatch.com, (408) 364-5508

This document shows technical details of the Seuss display prototype. Our design goals:

1) Inexpensive off-the-shelf components
2) Provide an integration to run animated and musical shows
3) Expandable through over-the-air (OTA) updates
4) Publish the design under an open-source license so all Makers may benefit from it

Software driving the Seuss Display is at https://github.com/frankcohen/ReflectionsOS



Why Seuss, because Theodor Seuss Geisel was turned down 28 times before being published. Seuss Display is the second attempt for the Reflections project - Mesa Display being the first and failed attempt.

None of this is complicated. However, putting it all together was difficult and time consuming. For example, all but one GPIO pin is used in this design.

The design uses mostly breakout boards from Adafruit. They provide needed pull-up resisters and level-shifters.

Seuss Display Prototype 2021

Reflections Project
fcohen@starlingwatch.com
(408) 364-5508

fcohen@starlingwatch.com, (408) 364-5508

http://www.hiletgo.com/ProductDetail/1906566.html

ESP-WROOM-32 ESP32 ESP-32S Development Board 2.4GHz Dual-Mode WiFi + Bluetooth Dual Cores Microcontroller
The ESP32 integrated with Antenna switches, RF Balun, power amplifiers, low-noise amplifiers, filters, and management modules, and the entire solution occupies the least area of PCB. 2.4 GHz Wi-Fi plus Bluetooth dual-mode chip, with TSMC Ultra-low power consumption 40nm technology, power dissipation performance and RF performance is the best, safe and reliable, easy to extend to a variety of applications.

Espressif - based in Shanghai, China - makes ESP32. It's an open-source design.

Software drivers and utilities for ESP32 are at:
https://github.com/espressif/arduino-esp32
https://github.com/nodemcu/nodemcu-firmware/tree/dev-esp32

Notes on using ESP32 with Arduino IDE:

Follow these instructions to install ESP32 to Arduino IDE
https://github.com/espressif/arduino-esp32/blob/master/docs/arduino-ide/mac.md

Use Node32s as Board selection

To reference GPIO pins in code use just the number, for example "digitalWrite(13, HIGH)" sets GPIO13 high.

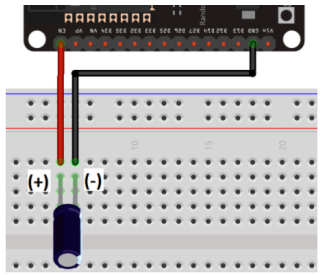
~/Library/Python/2.7/bin. I needed to patch esptool.py following https://github.com/espressif/arduino-esp32/issues/4408 and changing the +x perms too

I need to hold the boot button down to upload a sketch
https://randomnerdtutorials.com/solved-failed-to-connect-to-esp32-timed-out-waiting-for-packet-header/

ESP32 provides the easy-of-use of the Arduino environment (forums, experts, IDE), a 32-bit 2 core, and Bluetooth and Wifi networking.

ESP32 development boards provide a breadboard-friendly package with a USB programming interface and power supply.

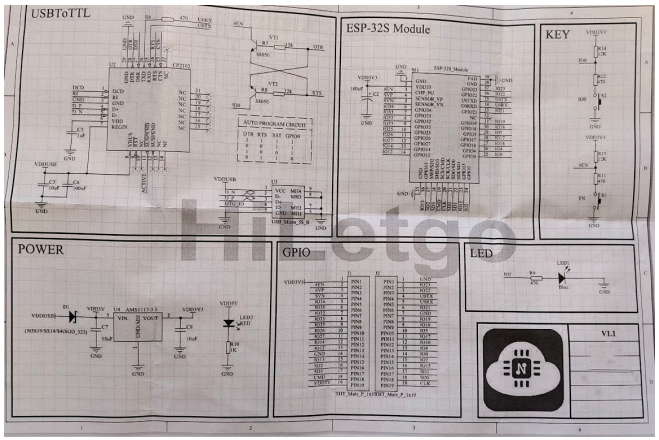
The downside of the development boards happens when people publish tutorials. Board change pinouts. For example, some boards provide access to ESP32's dual SPI bus pins, others do not.

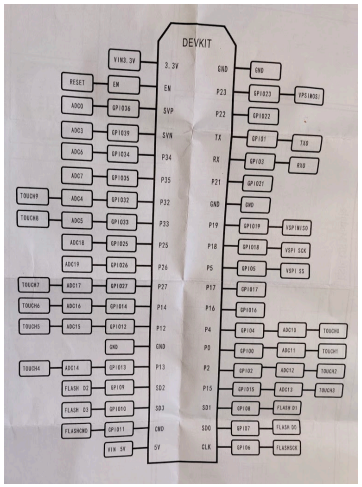


Note: electrolytic capacitors have polarity. The white/grey stripe indicates the negative lead.

If it works, then you can solder the **10 uF electrolytic capacitor** to the board. Since the EN and GND pins are far apart from each other, you can simply connect the capacitor between the EN and the GND of the ESP32 chip as shown in the schematic diagram below:

https://randomnerdtutorials.com/solved-failed-to-connect-to-esp32-timed-out-waiting-for-packet-header/





These pages came with the HiLetGo.com ESP32 Development Board

Wiring

Version: March 24, 2021

fcohen@starlingwatch.com, (408) 364-5508

		ESP32 Development Board			
		3.3v	19	GND	
	RESET	EN	18	GPIO 23	VSPI MOSI
input only	Left button	GPIO 36	17	GPIO 22	SCL I2C
input only	Right button	GPIO 39	16	GPIO 01	TX0
input only	Center button	GPIO 34	15	GPIO 03	RX0
input only	GPS TX	GPIO 35	14	GPIO 21	SDA I2C
	Display CS	GPIO 32	13	GND	
	Backlight LEDs	GPIO 33	12	GPIO 19	VSPI MISO
DAC1	Audio LRCL	GPIO 25	11	GPIO 18	VSPI SCK
DAC2	Audio BCLK	GPIO 26	10	GPIO 5	VSPI SS
	Audio DIN	GPIO 27	9	GPIO 17	Display RST
	HSPI SCK	GPIO 14	8	GPIO 16	Display DC
	HSPI MISO	GPIO 12	7	GPIO 4	Display SDCS
		GND	6	GPIO 0	Boot
	HSPI MOSI	GPIO 13	5	GPIO 2	LED
	Flash D2	GPIO 9	4	GPIO 15	HSPI SS
	Flash D3	GPIO 10	3	GPIO 8	Flash D1
	Flash CMD	GPIO 11	2	GPIO 7	Flash D0
		5V	1	GPIO 6	Flash SCK

Pins on ESP32 development board as seen from the top, 5V and GPIO 6 are closest to the USB connector

TX	GPS	GPS TX			
SPI	Display	CS	DC	SDCS	RST
I2S	Audio	LRCL	DIN	BCLK	
I2C	Compass	SDA	SCL		
I2C	Haptic motor				
I2C	Proximity sensor				

Breakout Boards

GPS https://www.adafruit.com/product/3133
Compass https://www.adafruit.com/product/2472
Display with SD Card https://www.adafruit.com/product/3787
Audio https://www.adafruit.com/product/3006
Haptic motor https://learn.adafruit.com/adafruit-drv2605-haptic-controller-breakout
Proximity sensor https://www.adafruit.com/product/466

This is the wiring guide. Use it as a top-of-board legend for all the components.

The most difficult part of this project was to keep track of the multiple uses of each pin. For example, GPIO 35 is input only. GPIO 2 looks normal but has an LED off of it.

In the end all of the available GPIOs are used. GPIO 14, 12, 13, 15, 4 are used by the internal Wifi system. 9, 10, 11, 6, 7, 8 are used by the Flash memory system. and 1, 3 are used by the USB adaptor.

I found conflicting tutorials, how-to guides, and references to data sheets (that turned out to be for different ESP32 development boards). One tutorial had a typo and kept referring to GPIO 20, which does not exist on the HiLetGo development board.

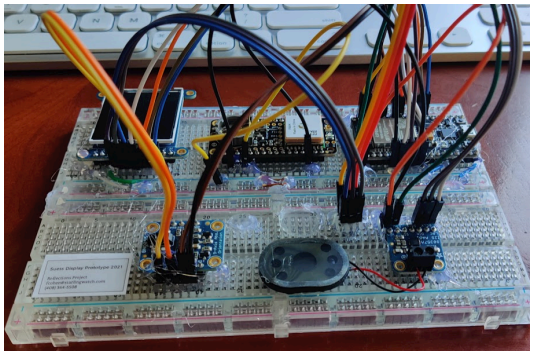
I also found I could not depend on translating tutorials and wiring guides for Arduino Uno and Arduino Mega to the ESP32 development boards.

Seuss uses the SD Card reader built into the Adafruit display breakout board - Display SDCS GPIO 4. I tried 2 SD Card reader boards and 2 SD cards. I could not get them to work. I have also read GPIO 4 is not available when using Wifi.

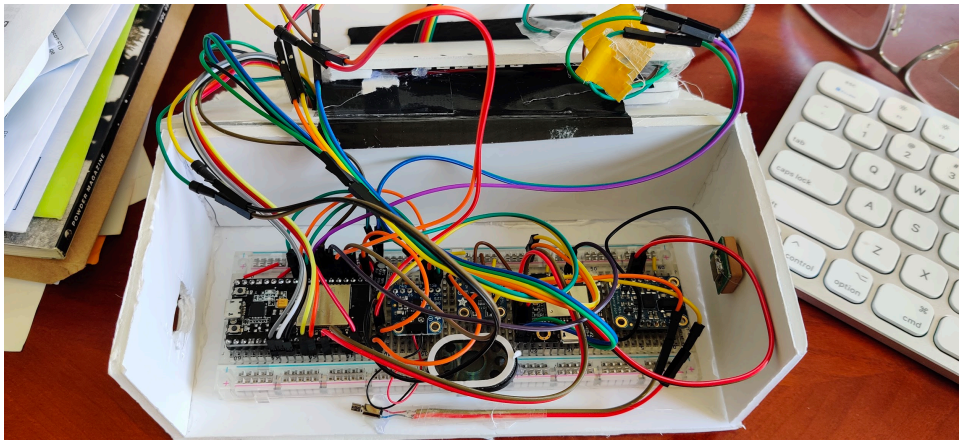
Many tutorials will reference pin numbers. While that works on Arduino Uno, the GPIO numbers are referenced here.

fcohen@starlingwatch.com, (408) 364-5508

Photo gallery during development



An early version, missing backlighting, haptic, proximity, buttons



Spring 2021 with all components installed, top of box opened in this photo.
The box top holds the buttons, display, and sensor

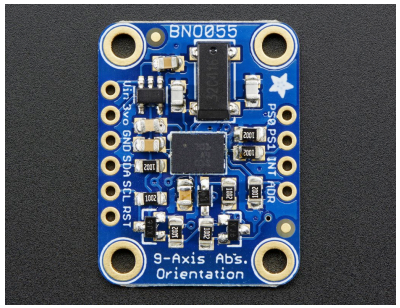
fcohen@starlingwatch.com, (408) 364-5508

MEMS accelerometer, magnetometer and gyroscope

Adafruit 9-DOF Absolute Orientation IMU Fusion Breakout - BNO055
https://www.adafruit.com/product/2472

Wiring guide at https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor/overview

https://github.com/adafruit/Adafruit_BNO055

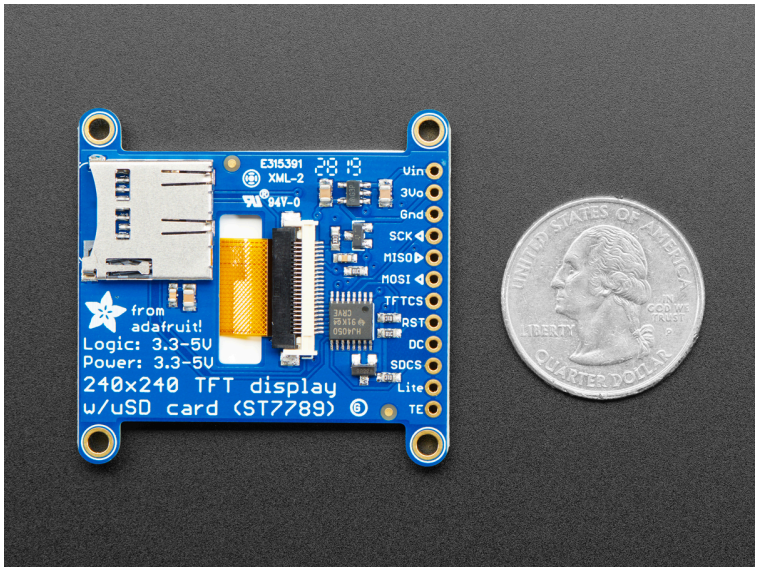


There are many accelerometers, compasses, and gyroscopes out there. And this board is relatively expensive. I decided on this one because it is all-in-one and requires only 2 I2C pins.

Display

Version: March 1, 2021

fcohen@starlingwatch.com, (408) 364-5508



https://www.adafruit.com/product/3787

I chose this display unit because it is fast - I can get 20 frames per second - and it includes an SD card reader on the breakout board. It uses SPI for both the display and SD card reader.

I am still confused by the difference between DC, TFTCS, and SDCS.

Use tutorial at https://www.instructables.com/Play-Video-With-ESP32/
Software at:
https://github.com/moononournation/RGB565_video

Uses Adafruit GFX library. Use examples from "Adafruit ST7735 and ST7789 Library" in the Arduino IDE examples drop-down menu


```
RAW/Uncompressed video playback library
https://github.com/cbm80amiga/ST7735_SDVideoPlayback

Video player
https://www.instructables.com/Play-Video-With-ESP32/
Works only with raw converted mpeg files, not rgb255 or anything else

#! /bin/bash
cd /Users/frankcohen/Desktop/ReflectionsExperiments/ReadyForTrial
for i in *.mp4; do /Applications/ffmpeg -i "$i" -vf
"fps=15,scale=-1:240:flags=lanczos,crop=240:in_h:(in_w-240)/2:0" -q:v
9 "/Users/frankcohen/Desktop/ReflectionsExperiments/Rendered/$
{i%.*}.mjpeg"; done
for i in *.mov; do /Applications/ffmpeg -i "$i" -vf
"fps=15,scale=-1:240:flags=lanczos,crop=240:in_h:(in_w-240)/2:0" -q:v
9 "/Users/frankcohen/Desktop/ReflectionsExperiments/Rendered/$
{i%.*}.mjpeg"; done
for i in *.wav; do ffmpeg -i "$i" -c:a libfdk_aac -b:a 8k "/Users/
frankcohen/Desktop/ReflectionsExperiments/Rendered/${i%.*}.m4a"; done
for i in *.m4a; do ffmpeg -i "$i" -c:a libfdk_aac -b:a 8k "/Users/
frankcohen/Desktop/ReflectionsExperiments/Rendered/${i%.*}.m4a"; done
```

fcohen@starlingwatch.com, (408) 364-5508

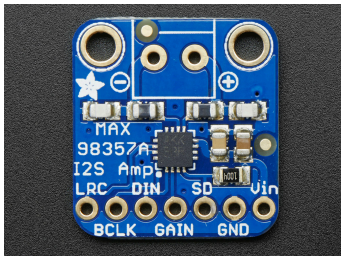
Startup sound by:
https://freesound.org/people/eardeer/sounds/385281/
https://freesound.org/people/tim.kahn/sounds/324059/

I built the start-up video using animation from
Nikita Kisan Rajiwade's project:
https://www.youtube.com/watch?v=WwKukI0f_Aw

I2S Sound Board

Version: March 3, 2021

fcohen@starlingwatch.com, (408) 364-5508



https://learn.adafruit.com/adafruit-max98357-i2s-class-d-mono-amp

```
libfdk_aac for converting wav to m4a (aac) is not installed by default in
ffmeg. I used these commands to install fumes with libfdk_aac on Mac OS X:

brew tap homebrew-ffmpeg/ffmpeg
brew install homebrew-ffmpeg/ffmpeg/ffmpeg --with-fdk-aac --HEAD

Then create an AAC audio file from an mp4 video
ffmpeg -i input.wav -c:a libfdk_aac -b:a 8k output.m4a
```

Use I2S audio library from Earle F. Philhower, III, earlephilhower@yahoo.com, for the WAV audio player library at https://github.com/earlephilhower/ESP8266Audio

Use ffmpeg to create AAC audio files from mp4 video file input: https://trac.ffmpeg.org/wiki/Encode/AAC

Tutorial on I2S at https://diyi0t.com/i2s-sound-tutorial-for-esp32/

MAX98357

The processor takes input files with:
sample rate of 44.1 kHz,
stereo format
16 bits
WAVE format

However, to play 240x240 MJPEG uncompressed files requires:
sample rate of 8000 kHz
16 bits
stereo format
WAV format
See experiments/Stream_all_videos_audio

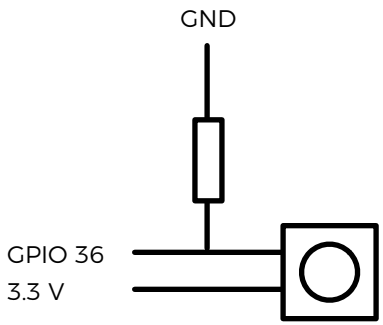
I use Audacity on a Mac Book Pro

Note: Connect the Gain pin on the MAX board to GND. Do not leave it open/floating.

Buttons

fcohen@starlingwatch.com, (408) 364-5508


```
#define leftButton 34
#define centerButton 39
#define rightButton 36
```



300 Ohm resistor

Temporary on switches

Seuss Display includes a 3 button interface for user control. Includes a left, middle, and right button.

See the buttons.ino sketch for an example of sensing three buttons

Note: the earlier versions of this circuit do not work and will cause a short.

fcohen@starlingwatch.com, (408) 364-5508

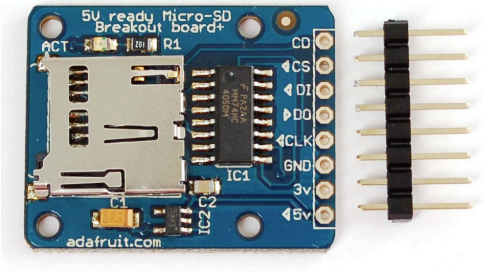
Seuss includes a haptic feedback system using the Adafruit DRV2605 breakout board. https://learn.adafruit.com/adafruit-drv2605-haptic-controller-breakout

Software is at: https://github.com/adafruit/Adafruit_DRV2605_Library

Failed Investigations

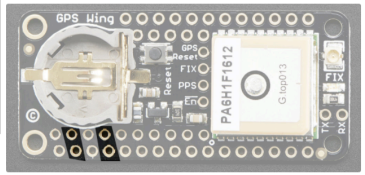
Version: March 1, 2021

fcohen@starlingwatch.com, (408) 364-5508



I tried, and failed, with SD card readers and the Adafruit VS1053 audio player.

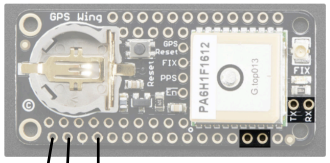
I used this library (got cardSize errors on Adafruit's branch):
https://github.com/greiman/SdFat

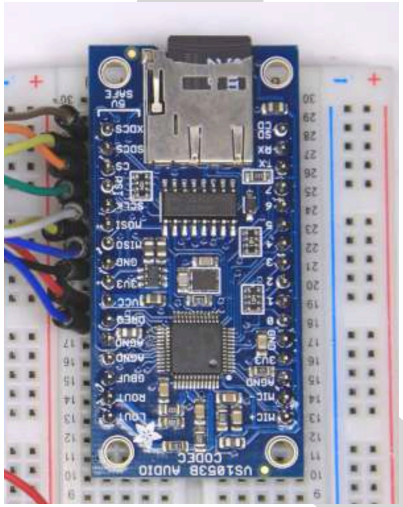


The GPS module runs from +3.3V power and uses 3V logic. the GPS is powered directly from the **3V** and **GND** pins on the bottom left of the Feather. Each Feather has a regulator to provide clean 3V power from USB or battery power.

https://learn.adafruit.com/adafruit-ultimate-gps-featherwing/pinouts

https://learn.adafruit.com/adafruit-vs1053-mp3-aac-ogg-midi-wav-play-and-record-codec-tutorial/software

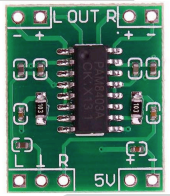


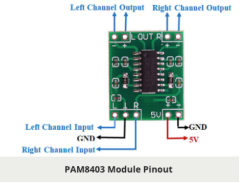


CS GPIO 16
XDCS-> GPIO 17
SDCS -> GPIO 32
DREQ -> GPIO 4
RST - GPIO25

CLK - GPIO18
MOSI - GPIO23
MISO - GPIO19

https://components101.com/modules/pam8403-stereo-audio-amplifier-module





https://www.amazon.com/DAOKI-Digital-Amplifier-PAM8403-Module/dp/B00XAGSZSK/ref=pd_yo_rr_rp_1