Zak Rowland
CST471
Preliminary design discussion

1.1 What are you doing?

My project is a handheld Bluetooth OBD-II diagnostic tool. The system will consist of two primary modules, the OBD-II port module and the handheld touchscreen module, both of which contain Bluetooth transceivers for communication.

1.2 Why you are doing it?

I chose this project primarily because it includes hardware, software, and data transfer protocols that I want to learn how to use. This includes using a Raspberry Pi with Linux, Python for scripting, a touchscreen interface, Bluetooth to send and receive data, rechargeable battery circuitry, and interfacing with an OBD-II port / CAN bus. I also like working on cars, so I can use this tool at home and others can do the same.

1.3 What is it about your project that is unique?

It's not a secret that projects like this one have been done before. One example called "OBD-Pi" was done in 2014, but this one used a prebuilt OBD-II Bluetooth reader and an aftermarket stereo head unit to display the data. My solution will include a custom made OBD-II interfacing circuit and a standalone handheld touchscreen device.

https://www.instructables.com/OBD-Pi/

Another example is a project made for someone's 1997 BMW, however like the first example, this one uses a prebuilt OBD-II reader and it has a different interpreter (STN1110). This example also uses a Raspberry Pi, but the touchscreen interface and Raspberry Pi is embedded permanently into the dash with a custom housing. Again, my OBD-II circuit will be custom made and open source, and the Raspberry Pi will be inside a handheld, wireless, rechargeable, touchscreen device.

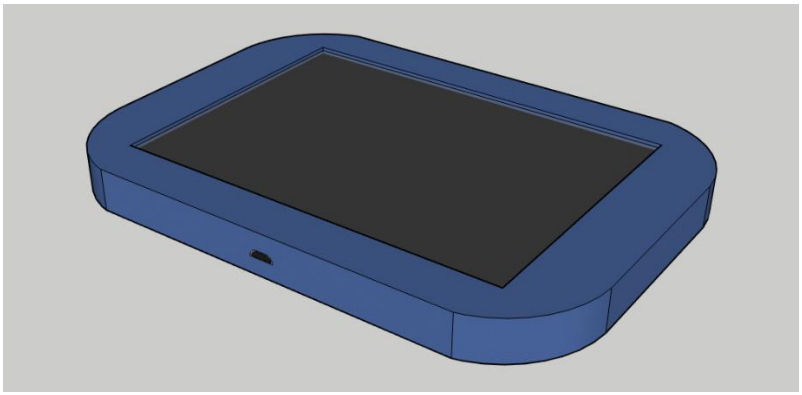
https://projects-raspberry.com/m3-pi-raspberry-pi-obd-ii-touchscreen-car-computer/

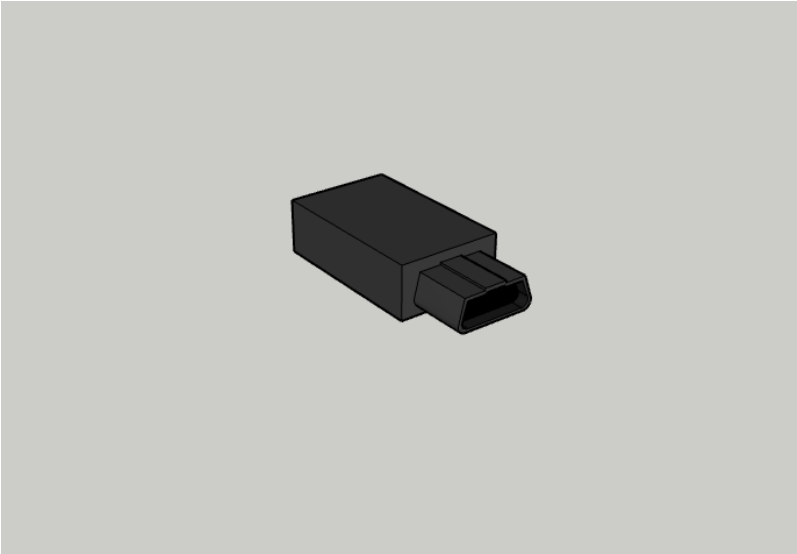
The existing commercially made options are cheap Bluetooth readers that connect to a phone app or expensive and bloated wireless solutions. My solution doesn't require a phone app, will be open source so features can be added, altered, or removed, and lower cost than the existing commercial wireless handheld solutions.

I wanted to also make the project as low cost as possible, however with limited lab equipment access, it is safer to go with parts that may cost more but will use high quality components and have good documentation.

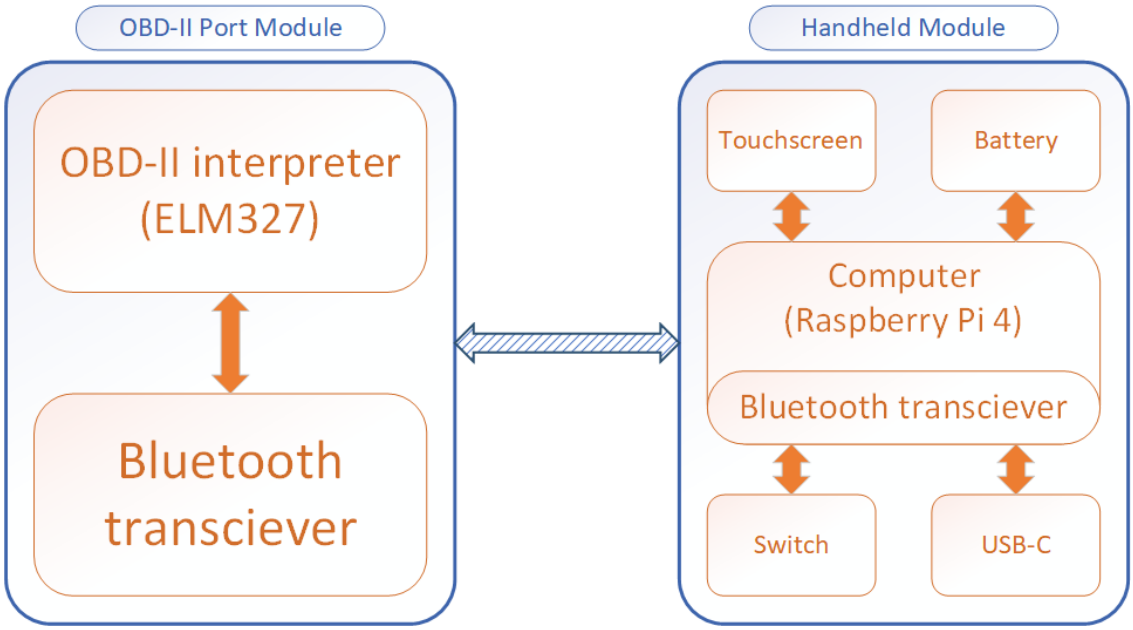
In short, my project is unique because the OBD-II reader will be built from scratch, the touchscreen interface is handheld and rechargeable, and all schematics and documentation will be open source.

2. A drawing of proposed final product





3. System block diagram



4. Three key requirements

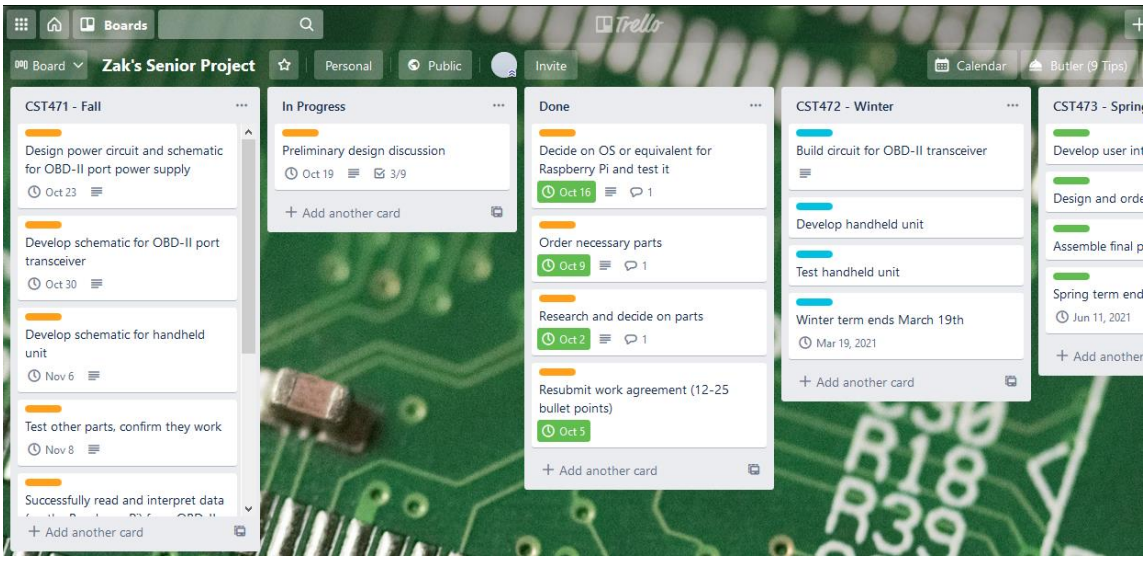
2. The system shall be able to read and clear diagnostic (trouble) codes.
    a. The user interface will display the diagnostic codes in list form with buttons to scroll up and down through the list.
        i. The list will display the diagnostic codes (e.g. P0011) only.

   ii. The user must touch one of the diagnostic codes to read the description or possible cause.
b. The user interface will provide a button to clear all diagnostic codes.

3. The system shall have the ability to read sensor data at minimum 30 times per second including speed, coolant temperature, and oil pressure.
   a. The user interface will display the data in decimal format.
      i. The option for digital gauges may be implemented.
   b. The data will be display *by default* in units of miles per hour for speed, Fahrenheit for temperature, and pounds per square inch for pressure.
      i. The option for metric units shall be implemented.
4. The system shall use Bluetooth 4.0 or greater for data transfer.
   a. The Bluetooth version shall be 4.0 or greater because previous versions do not support Bluetooth Low Energy (BLE.) The range of the connection is also improved with newer versions.

5. Estimated schedule (Gantt chart or other project management tool)

I am using Trello. I prefer the simplicity and flexibility over other schedule options.



6. Estimated NRE costs, prototyping and production costs (includes tools and software).

Estimated non-recurring engineering and prototyping costs		
Item	Cost	Total
Entry level computer engineer salary	$70,000	9-months for $52,500
(2) Breadboards	$2.50	$5
5V/3A+ USB-C power supply	$8	$8
Voltmeter/Ammeter	$20	$20
Oscilloscope	$250	$250
		$52,783

Estimated production costs		
Item	Cost	Total
Raspberry Pi 4 Model B 2GB	$35.00	$35.00
Elm Electronics ELM327 IC	$22.13	$22.13
OSOYOO 5” DSI display	$43.88	$43.88
Samsung 32GB MicroSD card	$7.99	$7.99
Bluetooth transceiver	$9.99	$9.99
MCP2551 (CAN transceiver)	$1.09	$1.09
Male J1962 Type A connector	$2.00	$2.00
4MHz crystal oscillator	$0.50	$0.50
7805 regulator	$0.50	$0.50
317L regulator	$1.00	$0.50
Rechargeable battery	$14.00	$14.00
Battery charger and protection	$7.00	$7.00
Assorted resistors	$3.00	$3.00
Assorted capacitors	$3.00	$3.00
Assorted diodes	$2.00	$2.00
Assorted transistors	$4.00	$4.00
		$156.58

7. Parts matrices and selection- be prepared to discuss.

Zak Rowland
CST471
Preliminary design discussion

Computer for handheld unit			
Name	Raspberry Pi 4 Model B	Raspberry Pi Zero W	BananaPi-M3
Price	$35	$10	$43
Processor	Broadcom BCM2711 (Quad core Cortex-A72 ARM v8), 64-bit, 1.5GHz	Broadcom BCM2835 (Single core ARM1176JZFS), 1GHz	Realtek RTD1395 (Quad core Cortex-A53), 64-bit
Storage	MicroSD	MicroSD	8GB eMMC + MicroSD
Memory	2GB LPDDR4-3200 SDRAM	512MB	1GB DDR4
Wireless	WiFi and Bluetooth 5.0	WiFi and Bluetooth 4.1	WiFi and Bluetooth 4.2
I/O	(2) USB 3.0, (2) USB 2.0, (2) Micro HDMI, DSI, CSI, 3.5mm A/V, Ethernet	Mini HDMI, USB OTG, CSI, Composite video	(4) USB 2.0, M.2, 3.5mm, HDMI, Ethernet
GPIO	40-pin	40-pin	28-pin
USB/Power	USB-C, 5V/3A+	Micro-USB, 5V/2A+	USB-C, 5V/2A+
Other	OpenGL ES 3.0 graphics		OpenGL ES 1.1/2.0 graphics

Zak Rowland
CST471
Preliminary design discussion

OBD-II to UART interpreter		
Name	ELM327	STN1110
Price	$21	$10
Operating voltage	4.5 – 5.5VDC	3 – 3.6VDC
Operating current	12mA	63mA
Power saver mode	0.15mA	<2mA
Operating power	60mW at 5V	207.9mW at 3.3V
Power saving power	0.75mW at 5V	6.6mW at 3.3V
Features	• Power control with standby mode • Serial interface • Automatically searches for correct protocol • Fully configurable with AT commands • Low power CMOS design • Very popular and well documented	• Fully compatible with the ELM327 (AT) command set • Extended (ST) command set • UART interface • Automatically searches for correct protocol • Large memory buffer (more RAM) • Voltage input for battery monitoring
Cons	• Costly • Many buggy fakes/clones if not purchased from manufacturer	• Less documentation and examples • Uses more power

Bluetooth Transceiver			
Name	HC-05	HC-06	HM-19
Price	$7.99	$7.39	$9.99
Bluetooth version	2.0	2.0	5.0
Operating voltage	3.6 – 6VDC	3.3VDC	3.6 – 6VDC
Logic voltage	3.3V	3.3V	3.3V
Modes	Master/slave	Slave only	Master/slave

8. Be prepared to answer questions about your project.

9. Be prepared to ask others questions about their project and point out possible flaws or problems in their idea.