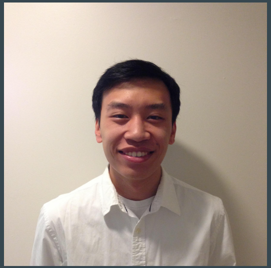
MDR
Pothole Tracker

Muhammad Mir . Daniel Chin . Mike Catalano . Bill Quigg
Advisor: Professor Ciesielski

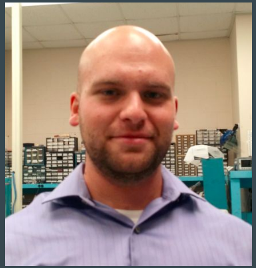
Pothole Tracker

Team 5

Muhammad Mir
CSE



Daniel Chin
CSE



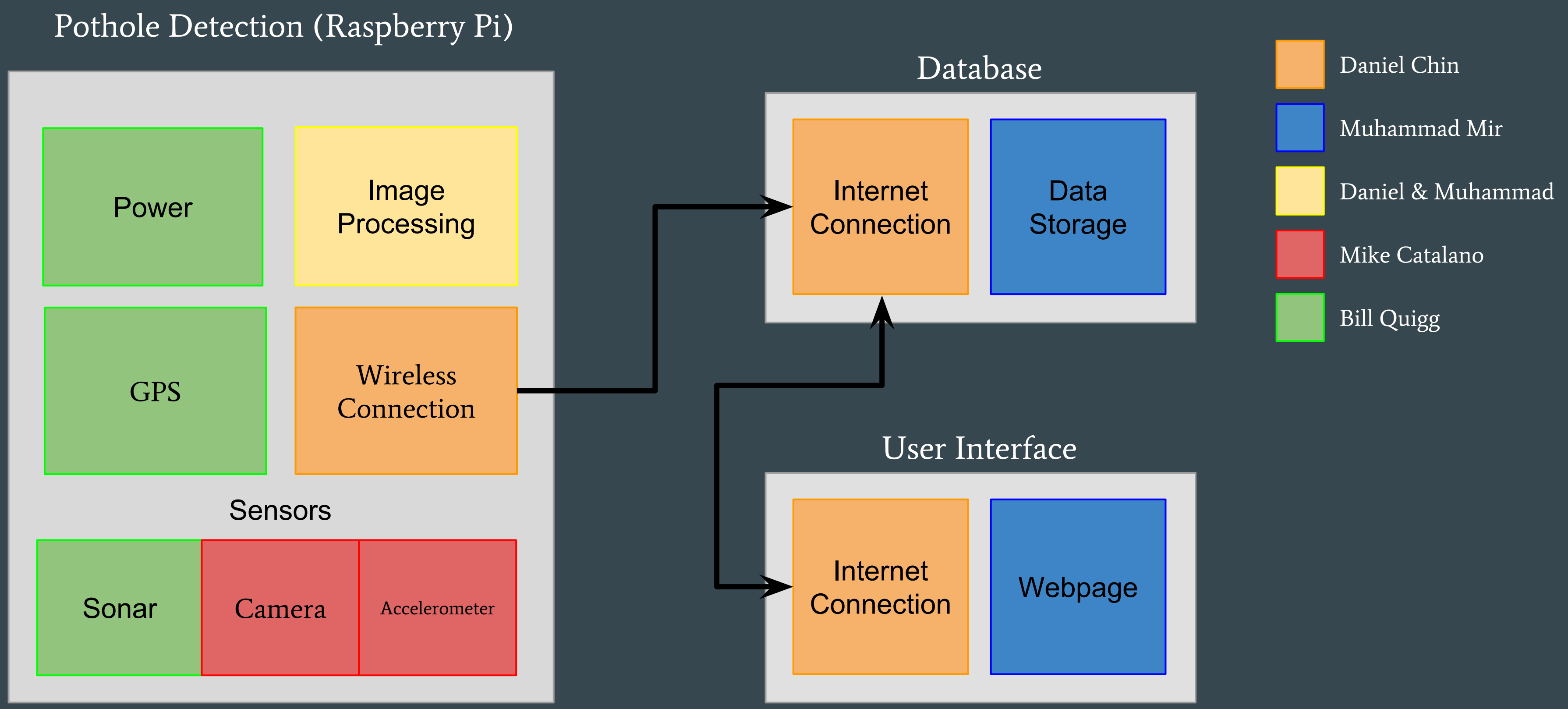
Mike Catalano
EE

Bill Quigg
EE

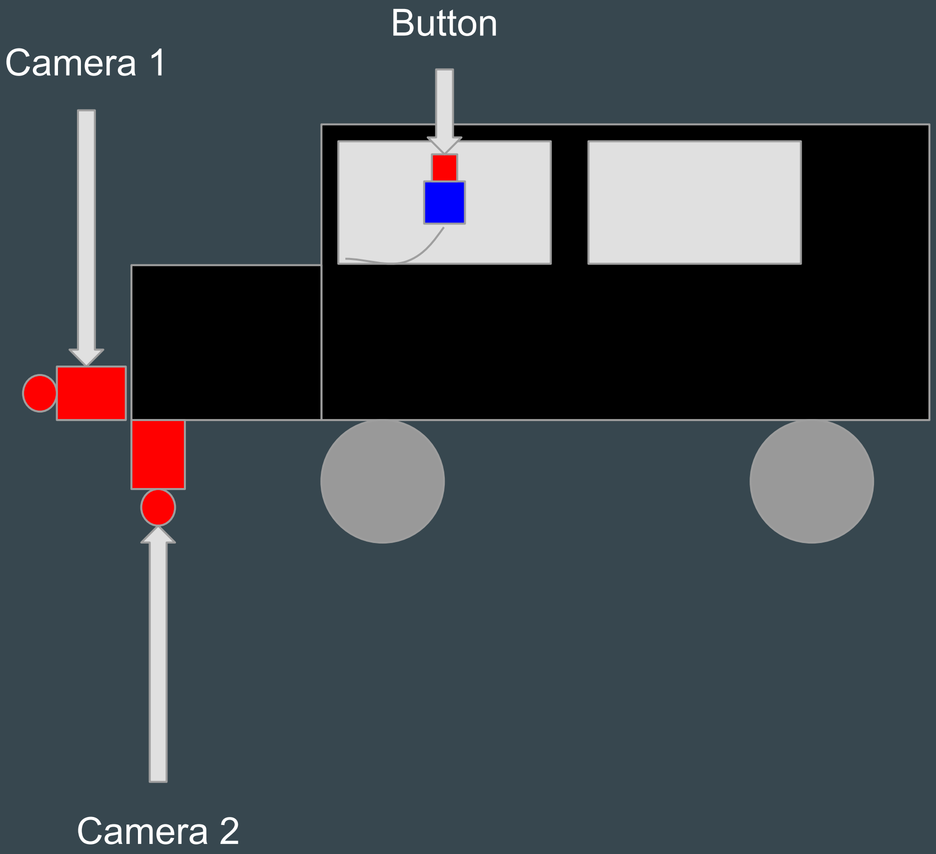
Why are Potholes a Problem?

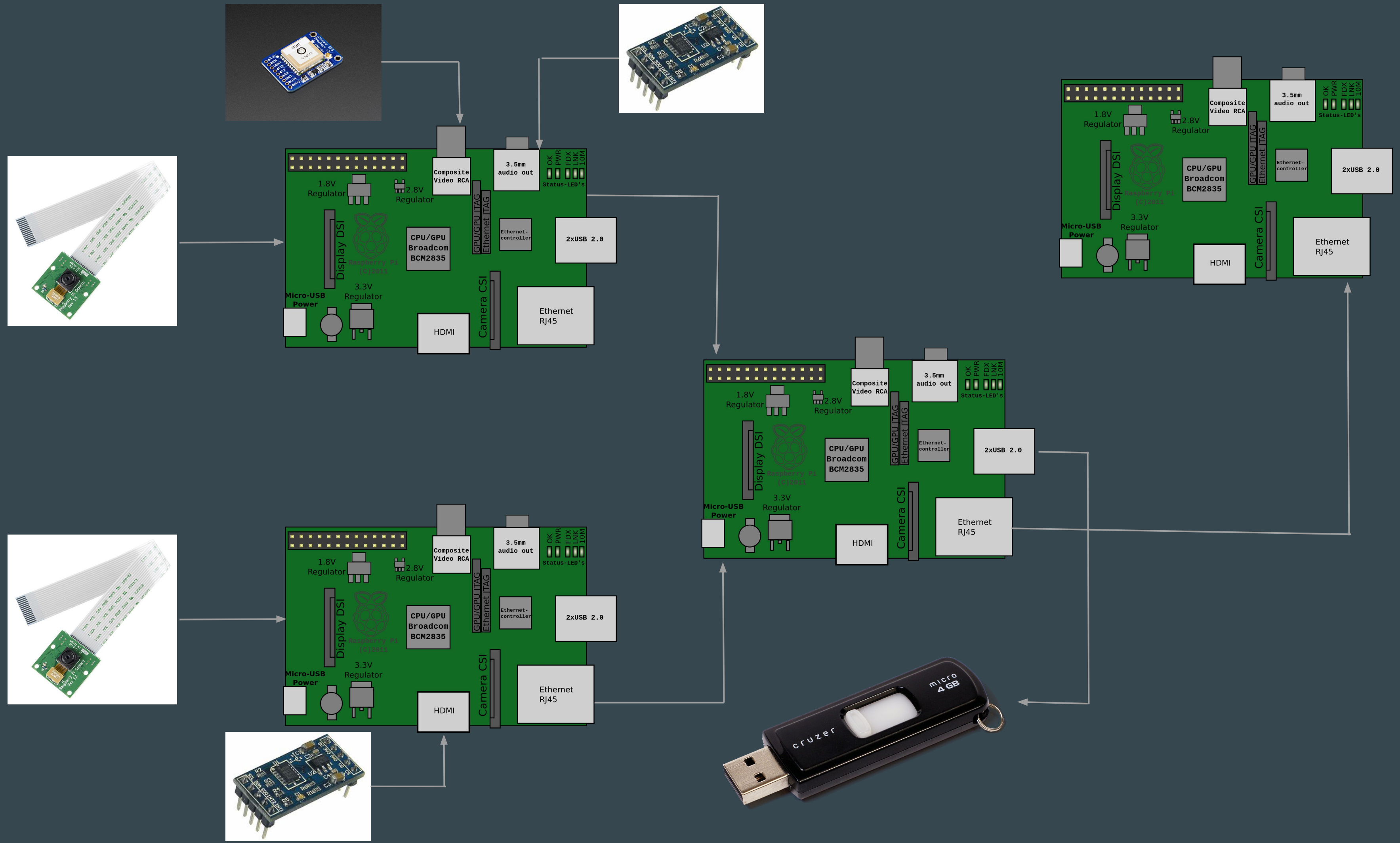
- Damaging to cars
- According to AAA pothole damage costs drivers $6.4 billion a year
- Repairs can range from $50 (wheel alignment) - $500 (alloy wheel)
- Can average about $2000 in repairs over life span of a car

Block Diagram (Old)



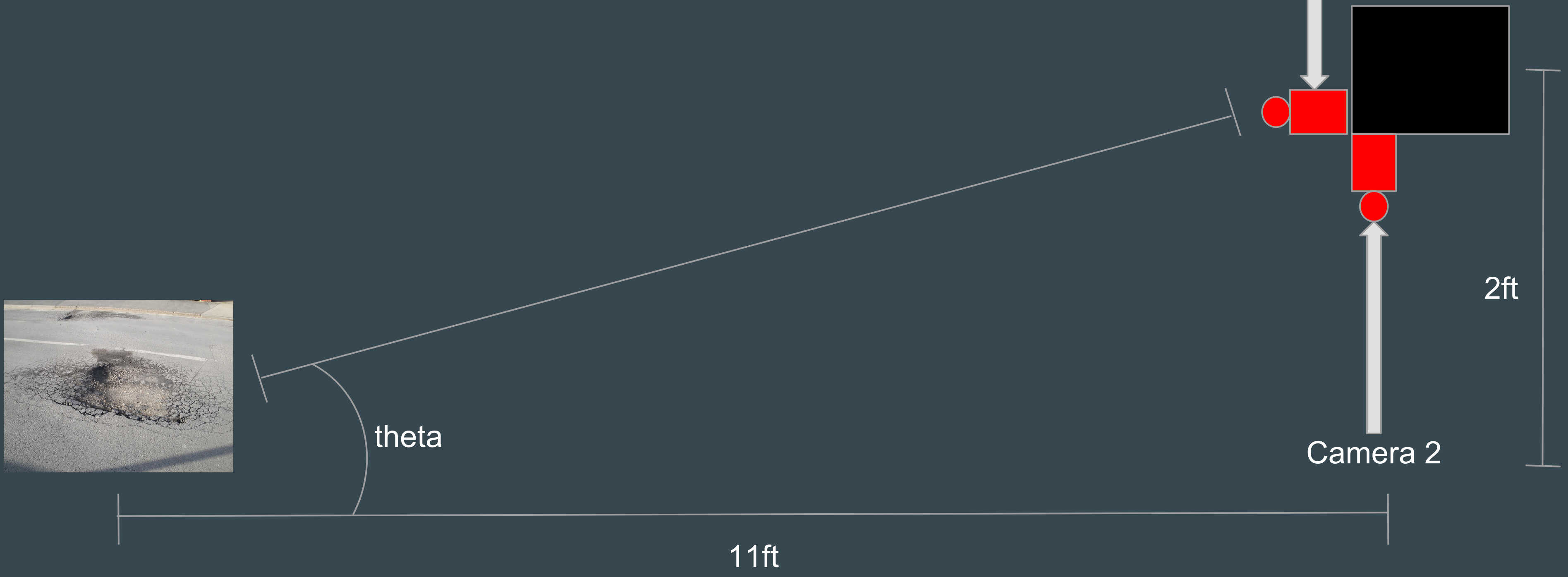
A New Design




Composite Video RCA
3.5mm audio out
OK PWR FDX LNK 10M
Status-LED's
1.8V Regulator
2.8V Regulator
Display DSI
CPU/GPU Broadcom BCM2835
GPU/GPU JTAG
Ethernet JTAG
Ethernet-controller
2xUSB 2.0
3.3V Regulator
Micro-USB Power
HDMI
Camera CSI
Ethernet RJ45
cruzer
micro 4GB

Camera Setup

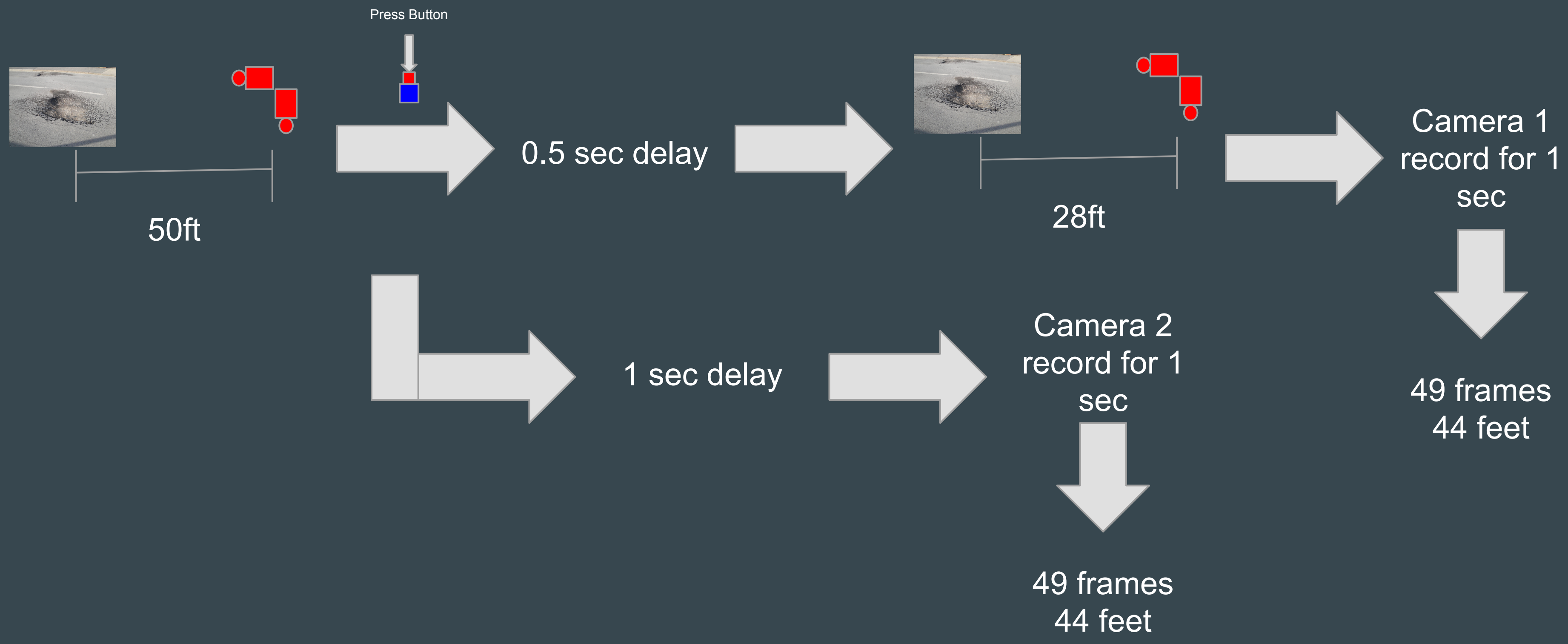
- 30 mph, 44 feet per second
- theta = tan^-1(2/11) = 10.53 degrees
- Camera video = 1 - 49 fps



Camera Considerations

Camera 2ft vantage point directly above start of crosswalk, visible edge is 11 ft away, camera angle is approx 11°

Scenario


Press Button
50ft
0.5 sec delay
28ft
Camera 1 record for 1 sec
49 frames 44 feet
1 sec delay
Camera 2 record for 1 sec
49 frames 44 feet

NAS(Network Attached Storage)

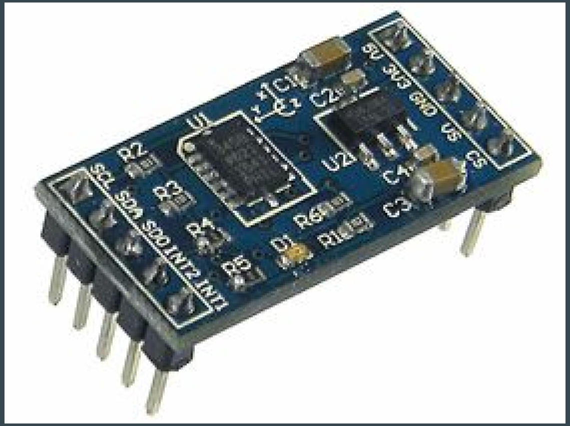
- Problem: Multiple boards need to have universal storage for images
- Have two camera boards and image processing board connect to a fourth board via ethernet/usb
- One ethernet vs. four usb on each board
- Possible use of switch

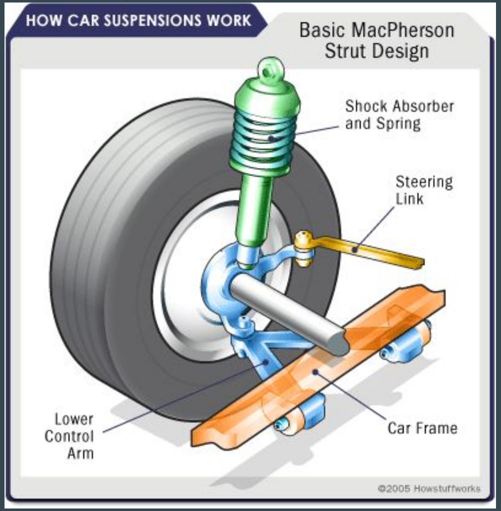
The Database/Webpage

- Manually choose when to send data to database
  - Vehicle will be parked and will be in an area with a good WiFi signal.

Accelerometers

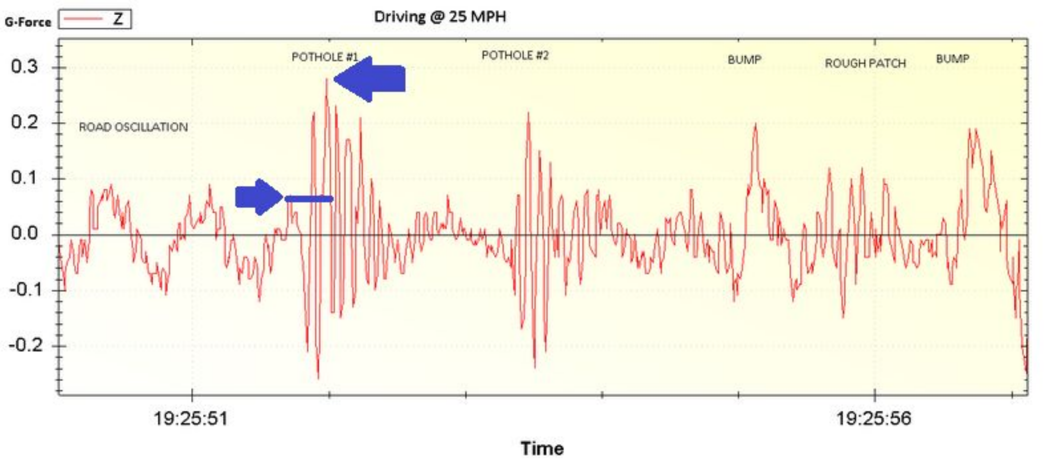
- Accelerometers collect force data as the operator drives over a pothole
- Two accelerometers are used to increase operator's chances of hitting a pothole
- Accelerometers must be mounted to a section of the car's suspension that remains static upon impact with a pothole (frame, control arm)





Accelerometers cont.

- Depth of pothole can be extrapolated from data
  - A car's tire is briefly in free-fall when the operator drives over the pothole
  - kinematics equation dictating free-fall:
    - $\Delta z = 0.5at^2$
    - Note the peak acceleration readings ($a = 0.28g$) where $g = 9.81m/s^2$
    - Note the duration of time over which the car begins free fall until it experiences max accel.
    - $t \cong 0.25s$
- $\Delta z = 0.5*(0.28*9.81m/s^2)*(0.25s)^2$
  - $\Delta z = 0.086m$ or 3.4"
- Filtering will provide more accurate results (12Hz)
- BPF where 12Hz < f < ?



Synchronizing Components

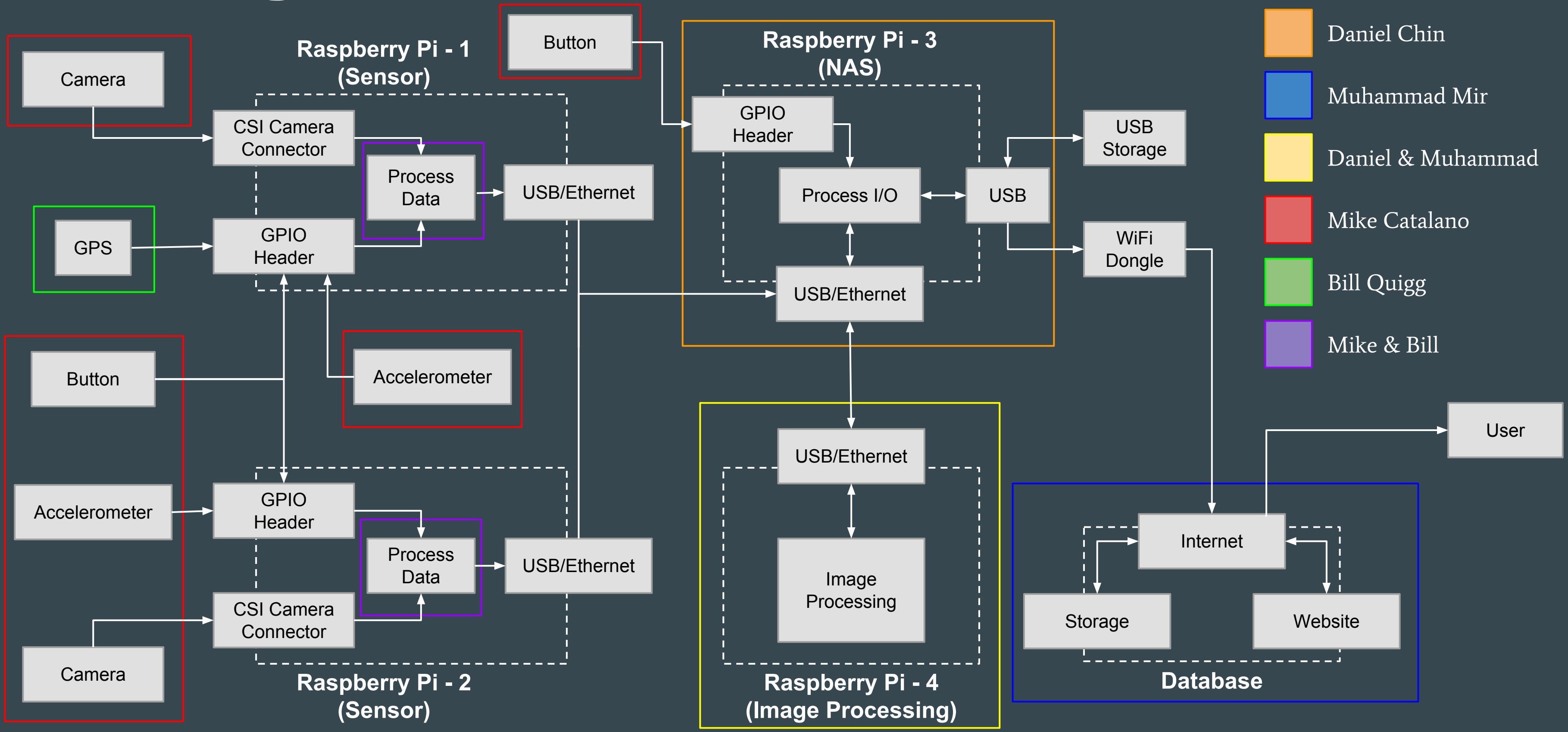
Time based communication between GPS, camera and accelerometer

- Time stamp all data recorded
- Time stamping camera stills using ExifRead Python library
- GPS has built in timestamp component
- Time stamp accelerometer data
- Algorithm:

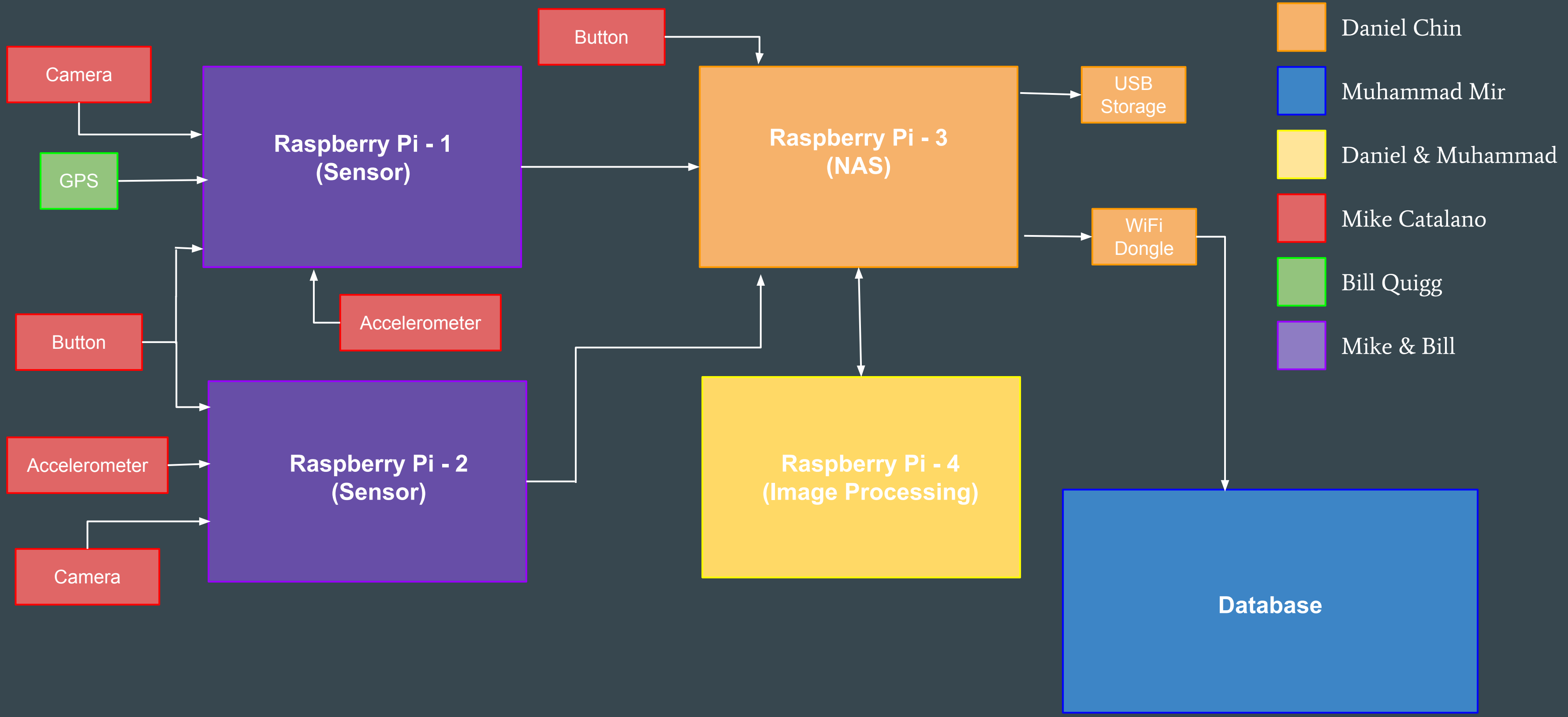
min | GPS time - camera time | = matched pair of GPS and camera

min | GPS time - Accelerometer time | = paired GPS and accelermoter data

Block Diagram (New)



TEST BLOCK DIAGRAM



MDR Deliverables

- Processed image and algorithms
- Parts (Raspberry Pi, wireless, gps, camera, accelerometer)
- Database for storing pothole specs will be set up

Image Processing/Algorithms: Muhammad

- Canny Edge detection
- Five step process
  a. Apply Gaussian Filter
  b. Find Gradient Intensity
  c. Trace edges based on pixel magnitude and orientation
  d. Apply double thresholds
  e. Get rid of weak edges

Image Processing/Algorithms: Daniel

- Sobel Operator
  - Goes through each pixel (except the edges)
  - Applies X and Y Gradient kernel
  - Computes the gradient magnitude
  - Normalize gradient magnitude to fit within 0 - 255
  - Applies the normalized gradient to the picture

-1	0	1
-2	0	2
-1	0	1

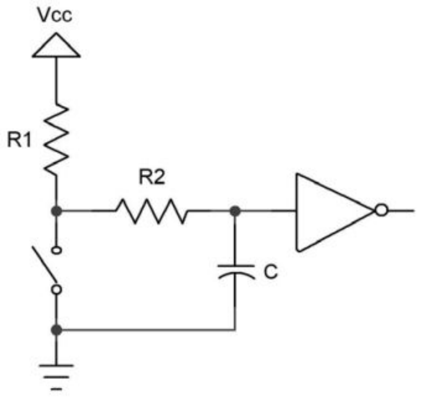
X - Gradient Kernel

-1	-2	-1
0	0	0
1	2	1

Y - Gradient Kernel

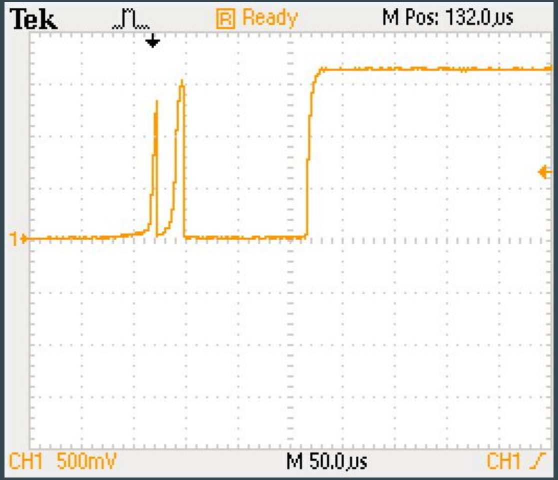
Button

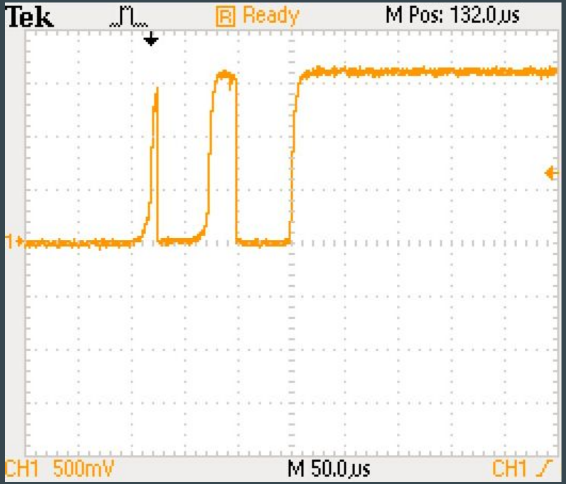
- Mechanical switch that allows driver to activate cameras and GPS
- Switch debouncing utilized to ensure reliable logic switching
    - Both hardware and software techniques exist
    - Hardware implementation chosen in order to maximize processor resources
    - Debouncing circuit: RC Debouncer with Schmitt Trigger



Switch Debouncing

- Ideally, the button is pushed/released and the logic level will change states once
- However, unnoticeable vibrations within the button cause logic levels to “bounce”
- The user inputs one button push but the processor recognizes several pushes



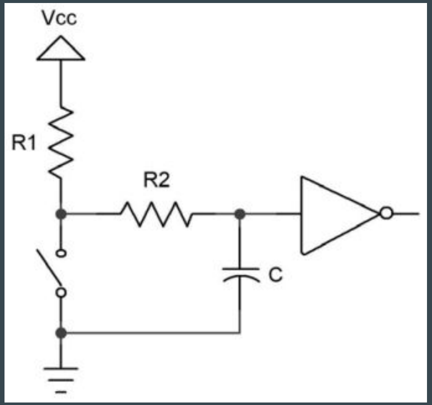


Switch Debouncing - Design

$$V_{cap} = V_{final}(1 - e^{\frac{-t}{RC}})$$

$$R = \frac{-t}{C\ln(1 - \frac{V_{cap}}{V_{final}})}$$

- Two cases to consider:
    a. button release (charging)
    b. button push (discharging)
- When the switch is not pressed, V_{cap} charges to V_{cc}. If the voltage across C = V_{cc}, the output of the inverter is Low.
- $V_{final} = V_{cc} = 5V$
- $V_{cap} = V_{T-}$ = threshold voltage for a logical Low = 0.8V
- Worst case bounce time, t = 175μs
- Selecting C = 0.1μF
- $R = R_1 + R_2 = 10037\Omega \cong 10k\Omega$

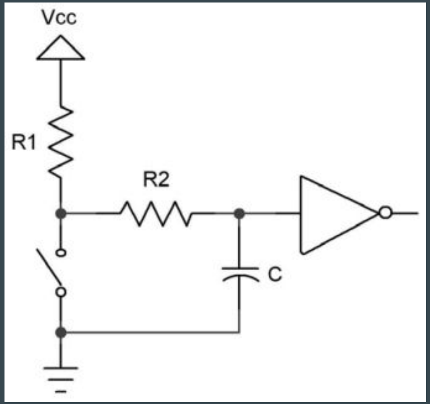


Switch Debouncing - Design cont.

$$V_{cap} = V_{initial}(e^{\frac{-t}{RC}})$$

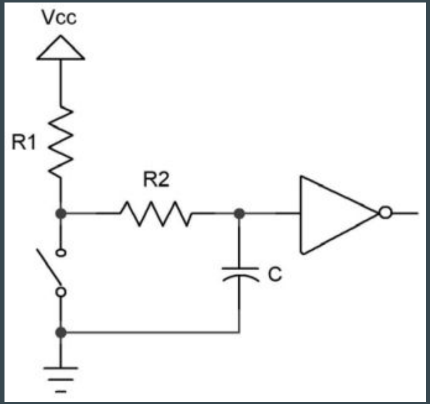
$$R = \frac{-t}{C\ln(\frac{V_{cap}}{V_{initial}})}$$

- When the switch is pressed, V_{cap} discharges to 0V. If the voltage across C = 0V, the output of the inverter is High.
- $V_{initial} = V_{cc} = 5V$
- C = 0.1μF
- $V_{cap} = V_{T+}$ = threshold voltage for a logical High = 1.6V
- t = 175μs
- Solving for $R_2 = 1536\Omega \cong 1.5k\Omega$



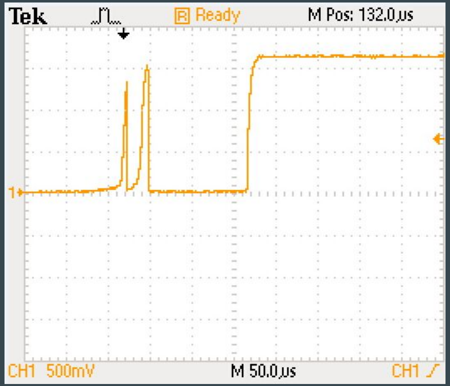
Switch Debouncing - Design cont.

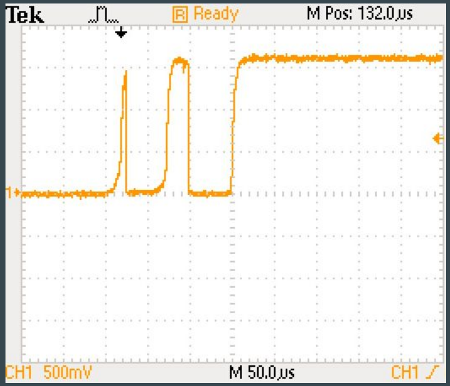
- Previous two slides yielded the following equations:
  - $R_1 + R_2 = 10k\Omega$ and $R_2 = 1.5k\Omega$
  - ⇨ $R_1 = 8.5k\Omega$
- Calculated component values:
  - $R_1 = 8.5k\Omega \cong 8.3k\Omega$
  - $R_2 = 1.5k\Omega$
  - $C = 0.1\mu F$
  - SN74LS14N Schmitt Trigger



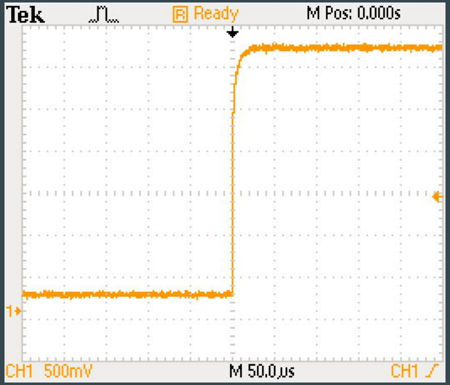
Switch Debouncing - Final Design

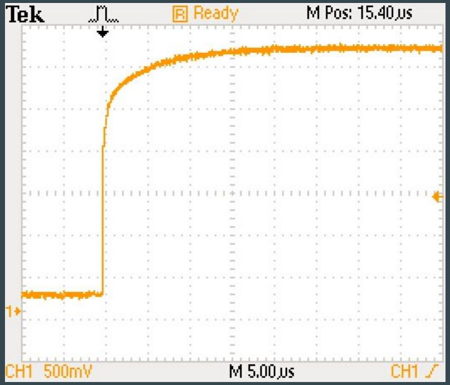
Before debounce:





After debounce:



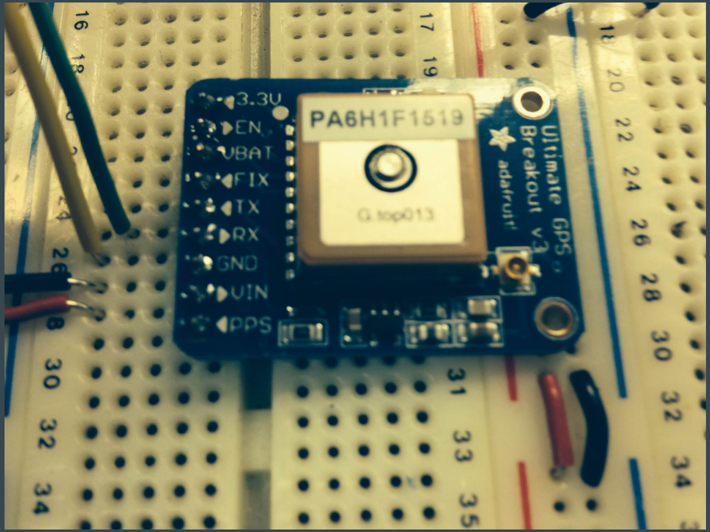


GPS Setup

Solder header to the Breakout Board

Configured Raspberry Pi to use UART Port instead of standard USB to TTL cable

Installed update software via an ethernet cable to achieve functionality



GPS Demonstration

When inside the SDP lab the GPS can not contact enough satellites to function

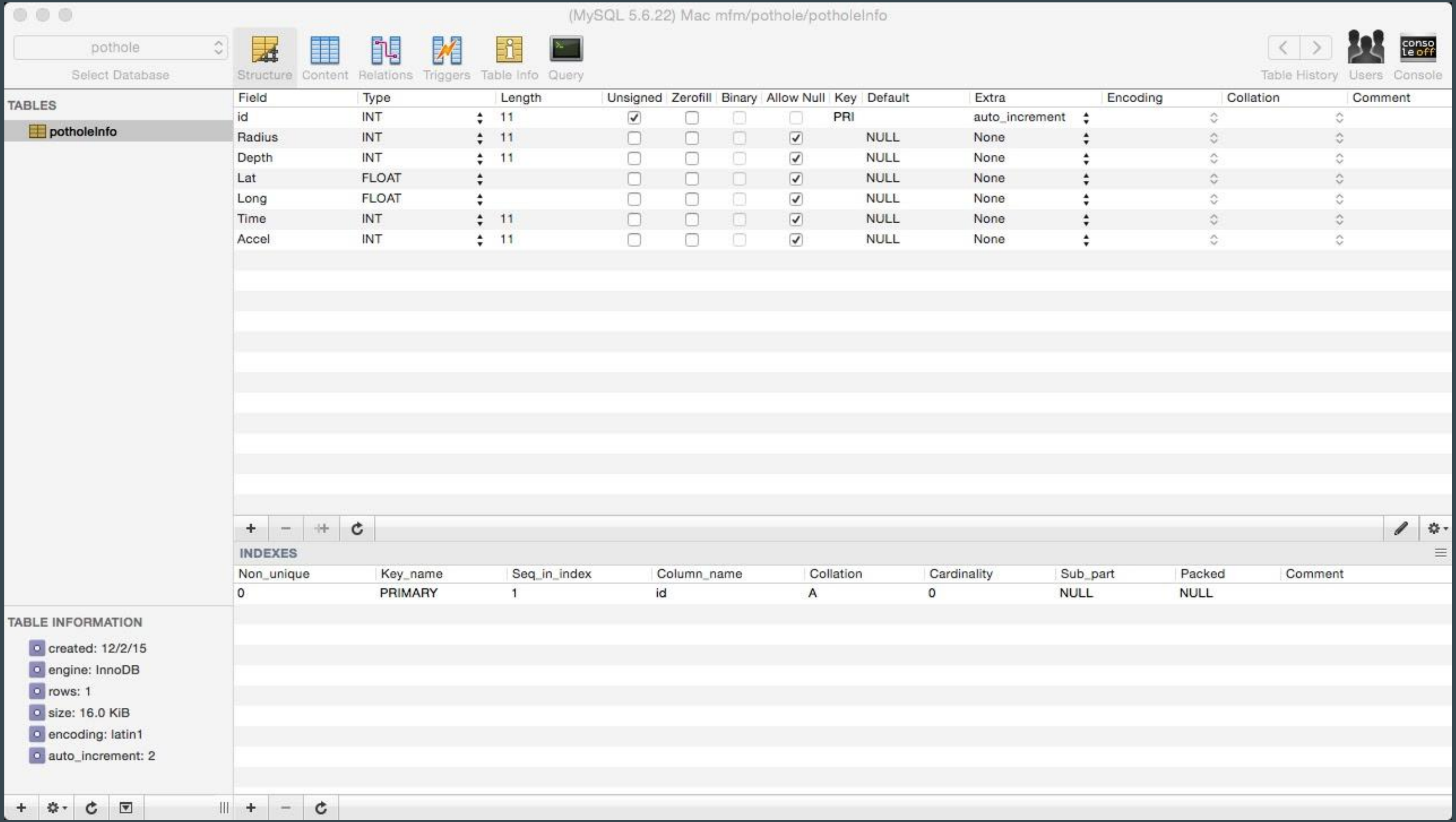
The red light beeps at a frequency of 1Hz when searching for satellites

The red light beeps once every 15 seconds when in contact with satellites

In person look at beeping at 1Hz

Video of GPS flashing once every 15 seconds

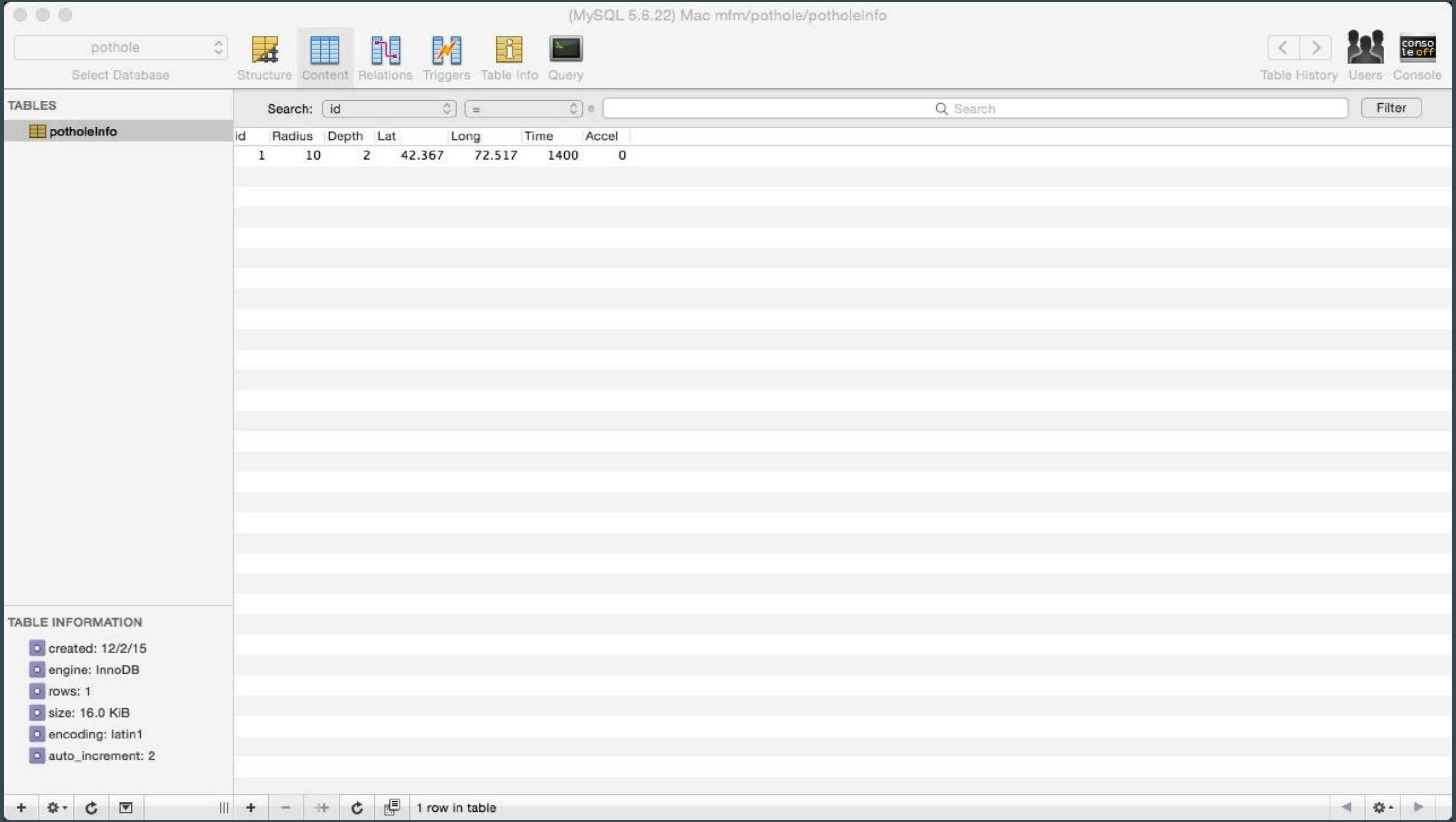
Database



Field	Type	Length	Unsigned	Zerofill	Binary	Allow Null	Key	Default	Extra	Encoding	Collation	Comment
id	INT	11	✓				PRI		auto_increment			
Radius	INT	11				✓		NULL	None			
Depth	INT	11				✓		NULL	None			
Lat	FLOAT					✓		NULL	None			
Long	FLOAT					✓		NULL	None			
Time	INT	11				✓		NULL	None			
Accel	INT	11				✓		NULL	None			

Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Comment
0	PRIMARY	1	id	A	0	NULL	NULL	

Database Cont.



CDR Deliverables

- Muhammad
  a. Create webpage displaying map and pothole locations
- Dan
  a. Setup NAS and send/receive data to/from usb storage
- Dan and Muhammad
  a. Improve image processing
  b. Be able to calculate depth and size of pothole
- Mike
  a. Mount accelerometers and apply filter circuitry
- Bill
  a. One timestamping format for all forms of data in the system
  b. GPS records location exactly 1.7 seconds after button is pushed
- Mike & Bill
  a. Design enclosure for mounting to front of vehicle