# Final Project Review

SDP Team 6

Sanghoon Lee
David Meschisen
Jori Platt
Minwo Wang

# The Team

Sanghoon Lee

- Microcontroller
- Camera
- IOIO

David Meschisen

- Projector/Display
- PCB
- Power System

Jori Platt

- Wink Detector
- Camera
- Casing

Minwo Wang

- Phone App
- IOIO
- PCB

# Intro

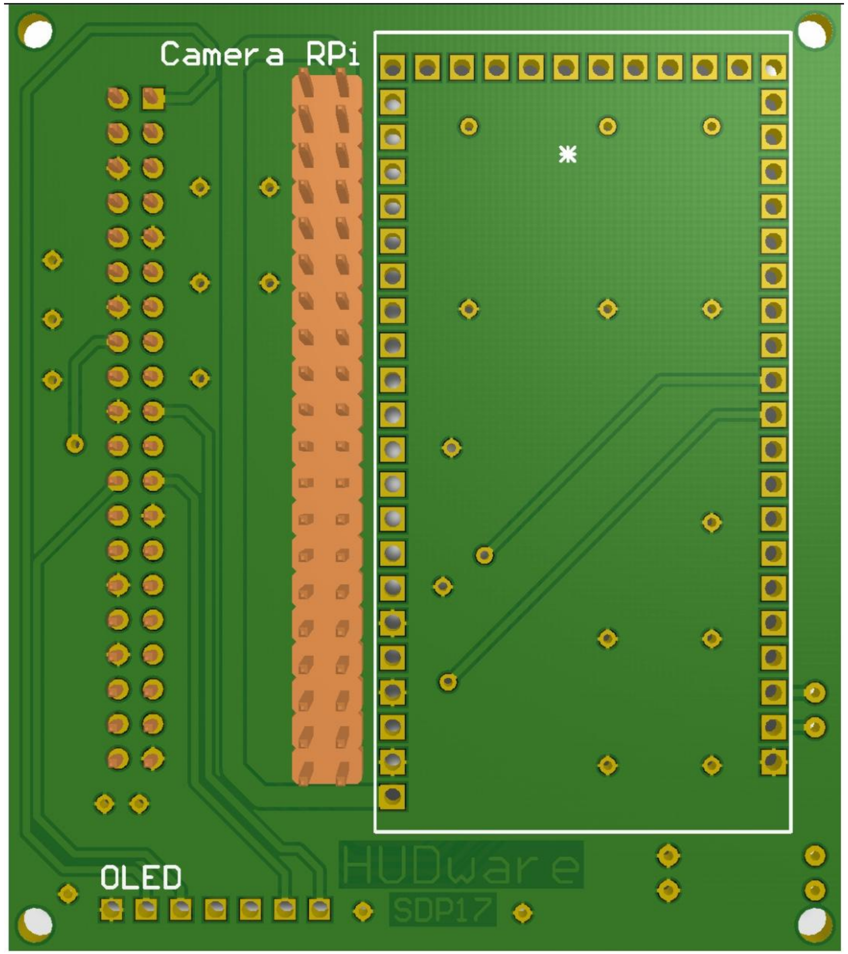
- A heads-up-display (HUD) attachment that connects a skier to their mobile device
- Uses wink detection to allow hands-free manipulation of the device

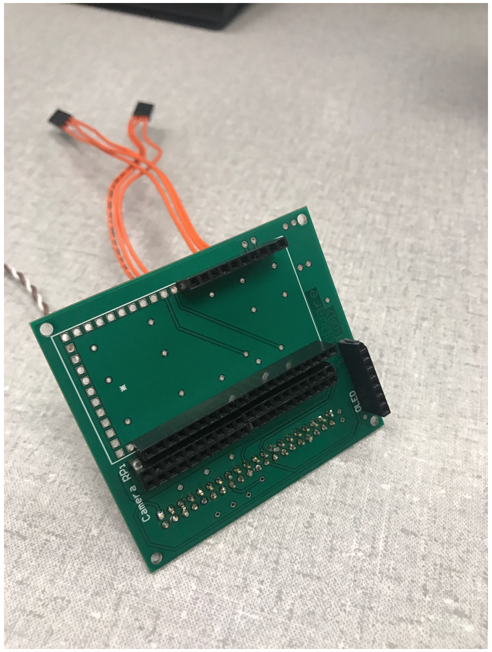
# Challenges

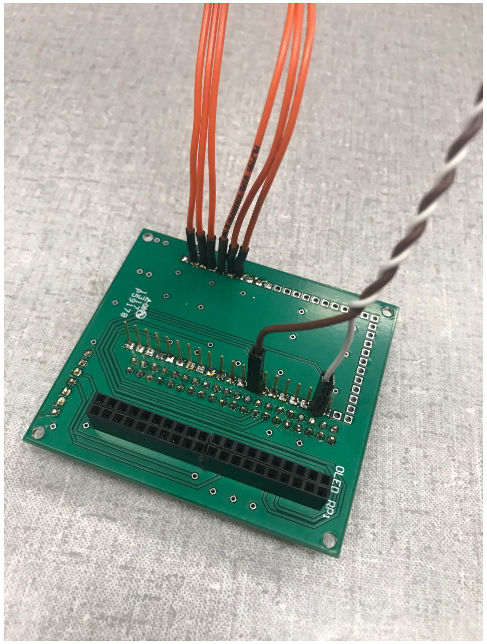
- Communication between devices
- Display information legibly and unobstructedly
- Accurately detect winks
- Control phone
- Collect relevant data from the phone application
- House components in a usable attachment

# Block Diagram

# PCB



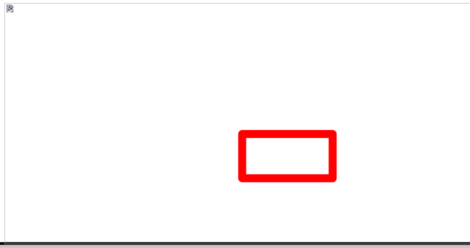




# Wink Detector

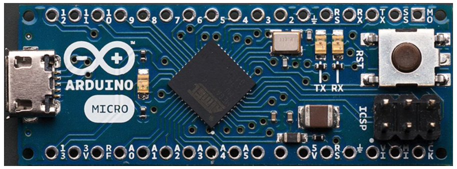
- Uses image processing techniques to detect if eyes are open or closed

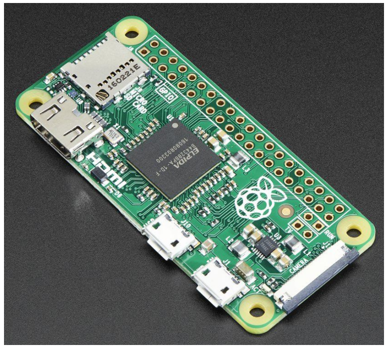
- Specifications
  - Speed (Requirement 2 fps)
    - 3.51 fps (timed and averaged between 50 images)
  - Accuracy (Requirement 90% images correctly identified)
    - 84.8% of images correctly identified (sampled from live stream and averaged between the 4 possible state results, n=23*4)



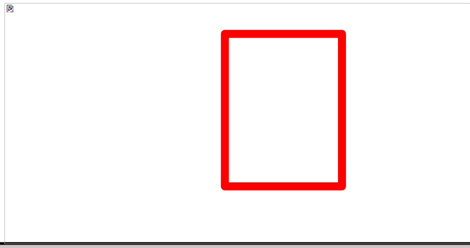
# Microcontroller

- Central processing units that relays information between Phone app and User
  - Phone app → Display
  - PiCamera → Phone app





- Specifications
  - MCU Temperature (Less than 80°C)
    - Average: 23°C (Arduino), 53°C (RPi)
  - Power Consumption (Less than 1W)
    - 5V @ 140mA (.7W), 5V @ 27mA (.135W)
  - Size (< 80mm x 80mm x 10mm)
  - Operational Temperature Range: (-17°C to 30°C)



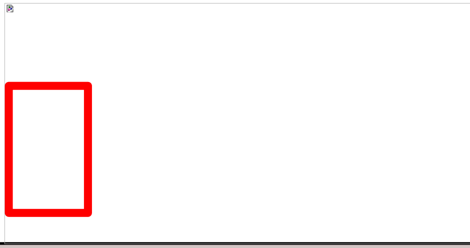
# Casing

- Plastic body that houses PCB, MCUs, IOIO, Camera, and Display
- Specifications
  - Houses relevant components
    - Met (Visual Inspection)
  - Operates in desired temperature range -20C to 30C
    - Tested at 25 C
  - Lightweight (100g)
    - Weighed 260g without MCUs
  - Unobstructed view
    - Met (Visual Inspection)

# Phone Application

- Collects data from phone:
  - SMS Text: user phone number, message content
  - GPS: speed, location
  - Music: play, pause, skip
  - Sensor Data

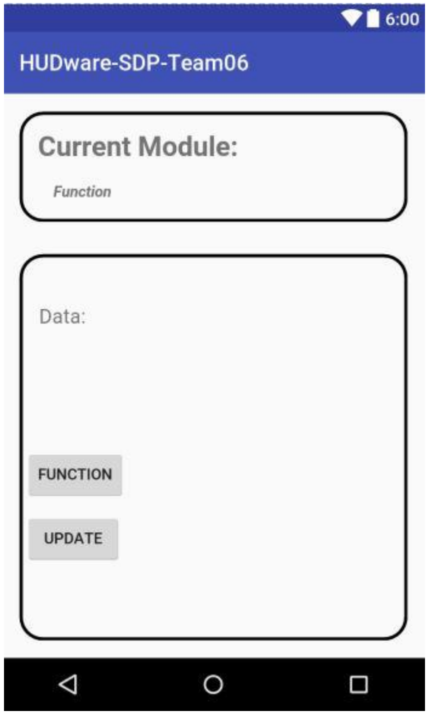
- Specifications
  - Gathers relevant information from phone
  - Easily maneuverable through GPIO signals
    - Manipulates music
    - Reads message data
    - Allows IOIO to control the phone



# Phone Application

- Function Implementation:
  - SMS Text: Broadcast Receiver
  - GPS Location: Google-Play-Service
  - GPS Speed: Haversine Formula
  - Music: MediaPlayer

- Testing Result:
  - Message can be shown on OLED Display
  - Accurate GPS location can be extracted
  - GPS Speed can be calculated
  - Songs can be restart, stop, and skip



Current Module:

*Function*

Data:

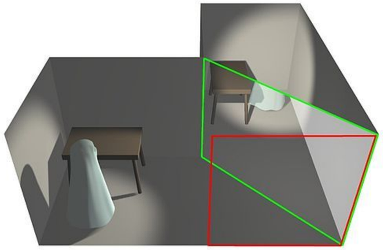
# Power Supply

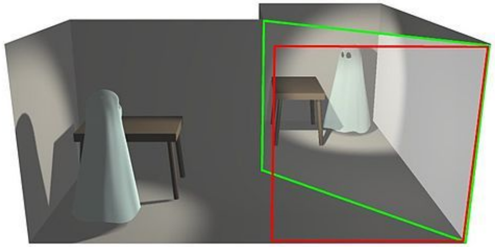
- External phone battery connected via USB cable to Raspberry Pi/Arduino Micro/IOIO

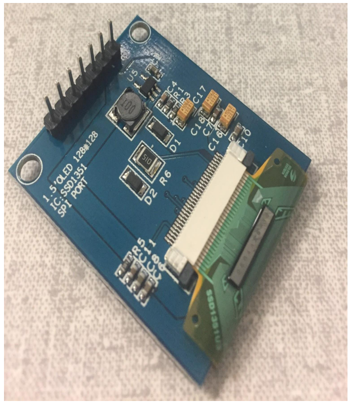
- Requirements
  - Powers whole system for 4 hours
    - Calculated to last 9.69 hours based on measured current draw

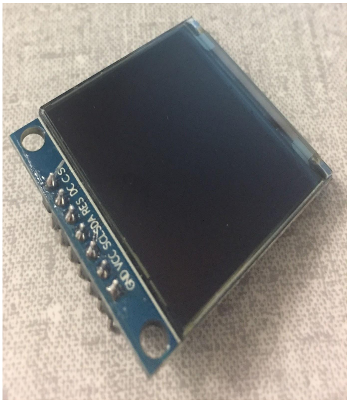
# Display

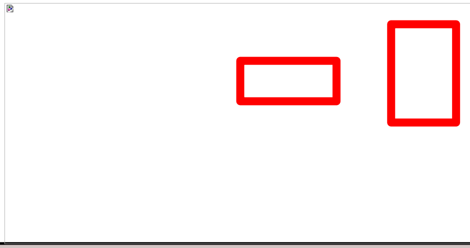
- SSD1351 OLED Display (128x128)
- Utilizes Pepper’s Ghost Effect
- 5 Pins connected to the Arduino Micro
  - 3 SPI pins
  - 2 User Configurable pins
- Uses Hardware SPI Bus on Arduino Micro











# Augmented Reality Display

Update:

- **Converted to Arduino Micro**
- **Added a 10cm FL Bi-Convex Lens**

Requirements:

- Focused & legible display (Image & Text) **USER TESTED**
- Minimum: 90 nits = 90 cd/m$^2$ **LIGHTMETER**
  - Target Range: 90 - 200 nits = 90 - 200 cd/m$^2$
- Operating temperature range: -20°C to 30°C **(-17 to 30)**
- Size: less than 50mm x 50mm **RULER**
  - OLED complies but optical path exceeds
- Power Consumption: less than 1W **VOLTMETER**

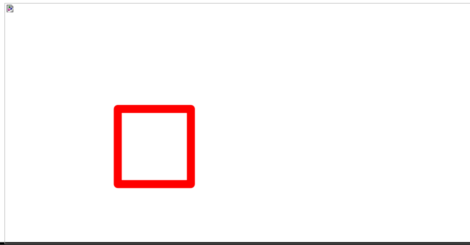
*The candela per square metre (cd/m2) is the derived SI unit of luminance. The unit is based on the candela, the SI unit of luminous intensity, and the square metre, the SI unit of area. Nit (nt) is a non-SI name also used for this unit (1 nt = 1 cd/m2). (Wikipedia)

UMassAmherst

# IOIO

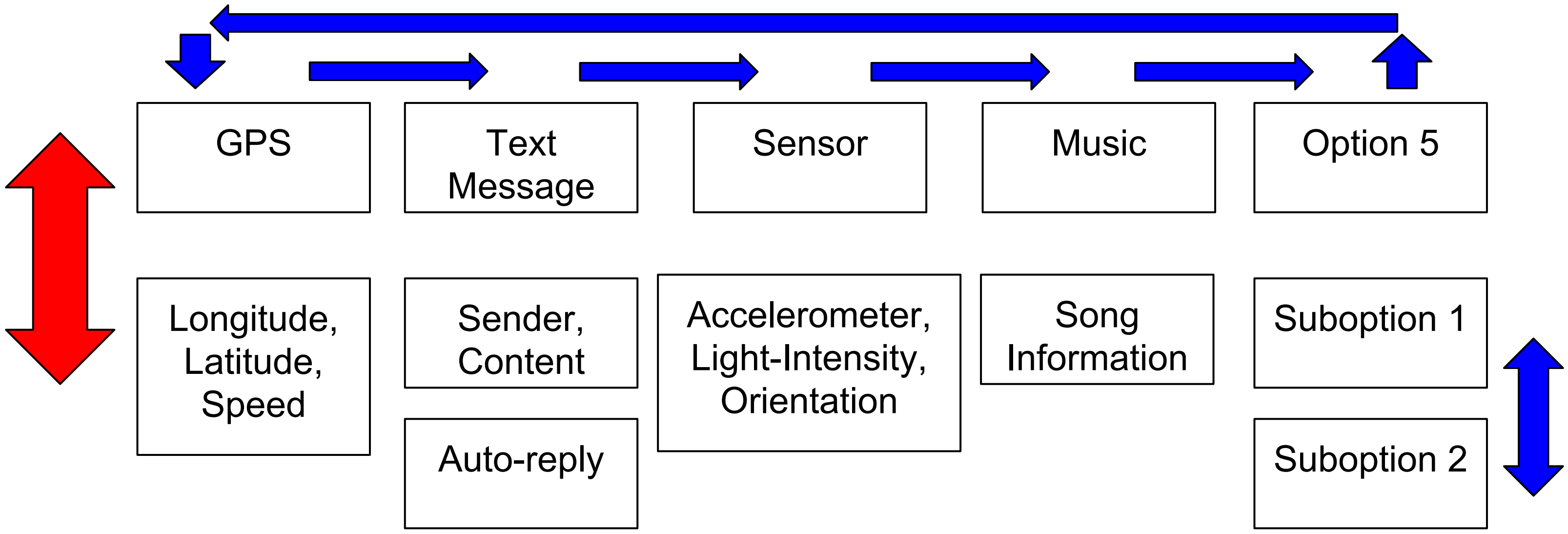
- Requirements:
  - Providing enough GPIO pins
  - Providing UART serial data communication
  - Establishing USB Connection with AOA Protocol

- Updates:
  - Merging Android App Life Cycle and IOIO looping mechanism by implementing IOIOLoopProvider
  - Using IOIOHelper to create IOIO Cycle(while loop)
  - High level software can send out low level hardware data, and also works in the other way



# Menu Controls

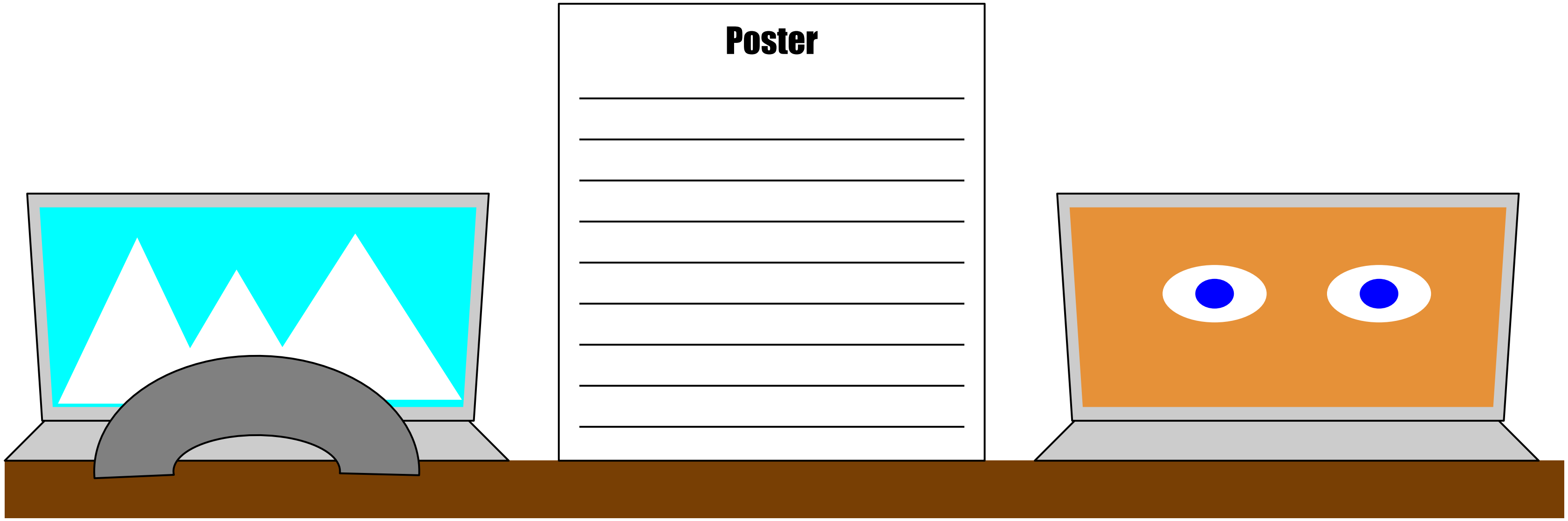
- 4 main functions
- left wink rotates between functions
- right wink selects/goes back to main options



# Deliverables

- PCB fabricated
- Full product assembled
    - All appropriate parts integrated in casing
    - Phone App
    - Projection clearly visible
    - Wink Detection allows for easy control

# Demo Day Setup



# Demo Day Presentation

- "30 second" Demo:
  - How HUDware can improve skiing experience
  - Try out system with POV video
- "2-5 minute" Demo:
  - How subsystems connect
  - How subsystems work
  - Try out system with POV video
- "10+ minute" Demo:
  - In depth description of subsystem processes
  - Try out system, receive text messages

# FPR Demonstration

- Try out HUDware
  - Wink Detection
  - Display
  - Phone App
    - Music
    - Message
    - GPS/Speed
    - Sensor Data

# Thank You

# Backup Slides

# Cost to Produce

| Price | Component |
|---|---|
| $80 | PCB |
| $5 | Raspberry Pi 2x |
| $40 | IOIO |
| $39 | Camera and Cable |
| $20 | Battery |
| $10 | Cable |
| $40 | OLED |
| $6 | Optical Lens |
| $12 | Fisheye Lens |
| <$1 | LEXAN Plastic Prism |
| $33 | Casing |
| $285 | Total |

# Budget Spent

| Price | Component |
|---|---|
| $49.49 | Zero, Camera, Camera Cable |
| $49.90 | Zero With Components |
| $10.99 | Fisheye Lens |
| $32.08 | Extended Camera Cable, USB Power Splitter |
| $49.59 | OLED,  USB OTG Cable |
| $45.67 | IOIO |
| $79.5 | PCB |
| $13.51 | 3D Print Bottom Test model |
| $66 | MIE 3D Print |
| | |
| $396.73 | Total |
| $103.27 | Remaining |

# Testing Methods

- 23 images were taken from a live stream where the subject was presenting one of the following situations
- An image was determined correctly identified if the code responded appropriately to the stimulus

| Image Type | % Correct | 95% Confidence Interval |
|---|---|---|
| Both Eyes Open | 100 | 100-100 |
| Left Wink | 87 | 73-100 |
| Right Wink | 65 | 46-84 |
| Both Eyes Closed | 87 | 73-100 |
| Combined | 85 | 77-92 |

# Phone Application--GPS

- Extract longitude and latitude data from User’s current position.
- Implementing GPS with Google-Play-Services
  - Internet has to be connected
  - Set minimum distance and time interval when the position is changed

# Phone Application--Speedometer

- Using Longitude and Latitude from GPS function
- Applying Harversine formula to calculate speed:

$$hav\left(\frac{d}{r}\right) = hav(\varphi_2 - \varphi_1) + \cos(\varphi_1)\cos(\varphi_2)\,hav(\lambda_2 - \lambda_1)$$

$$d = r \cdot hav^{-1}(h) = 2r \cdot \arcsin\left(\sqrt{hav\left(\frac{d}{r}\right)}\right)$$

$$d = 2r \cdot \arcsin\left(\sqrt{\sin^2\left(\frac{\varphi_2 - \varphi_1}{2}\right) + \cos(\varphi_1)\cos(\varphi_2)\sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right)$$

$d$ *is the distance between the two points*

$r$ *is the radius of the sphere*

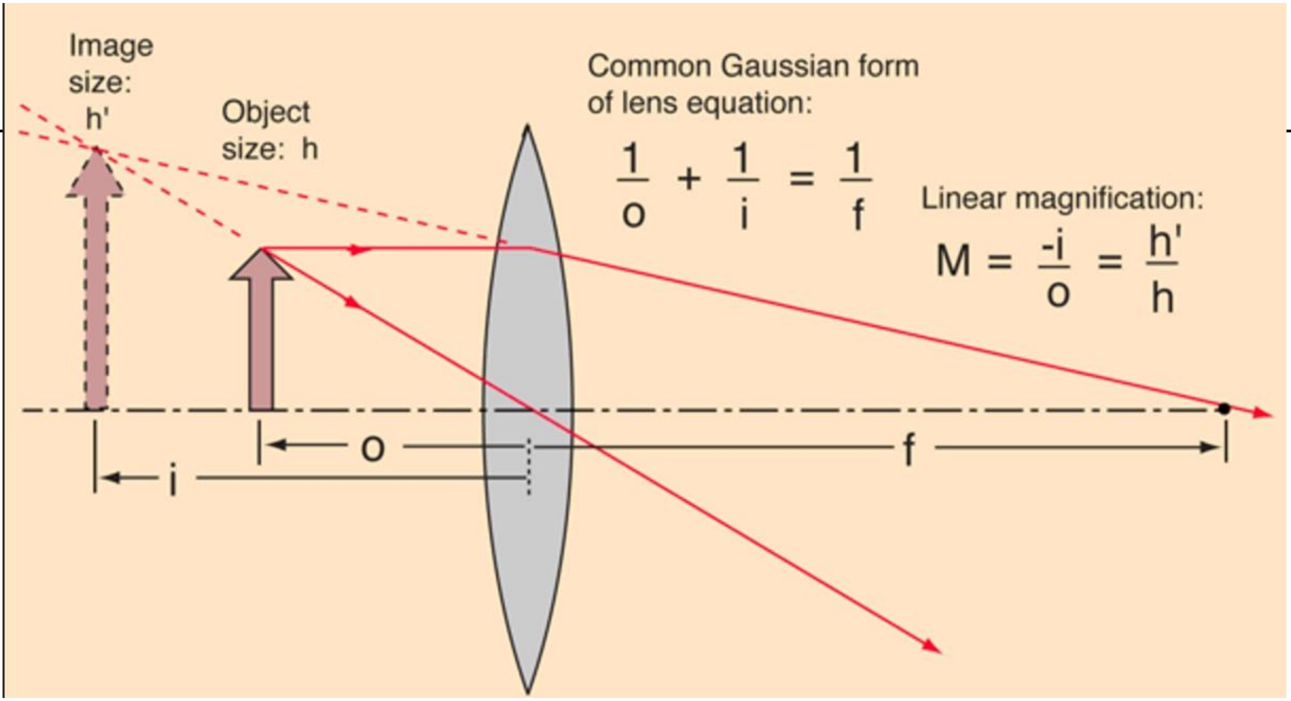
$\varphi_1, \varphi_2$: *latitude of point 1 and latitude of point 2*

$\lambda_1, \lambda_2$: *longitude of point 1 and longitude of point 2*

# Phone Application--Messaging

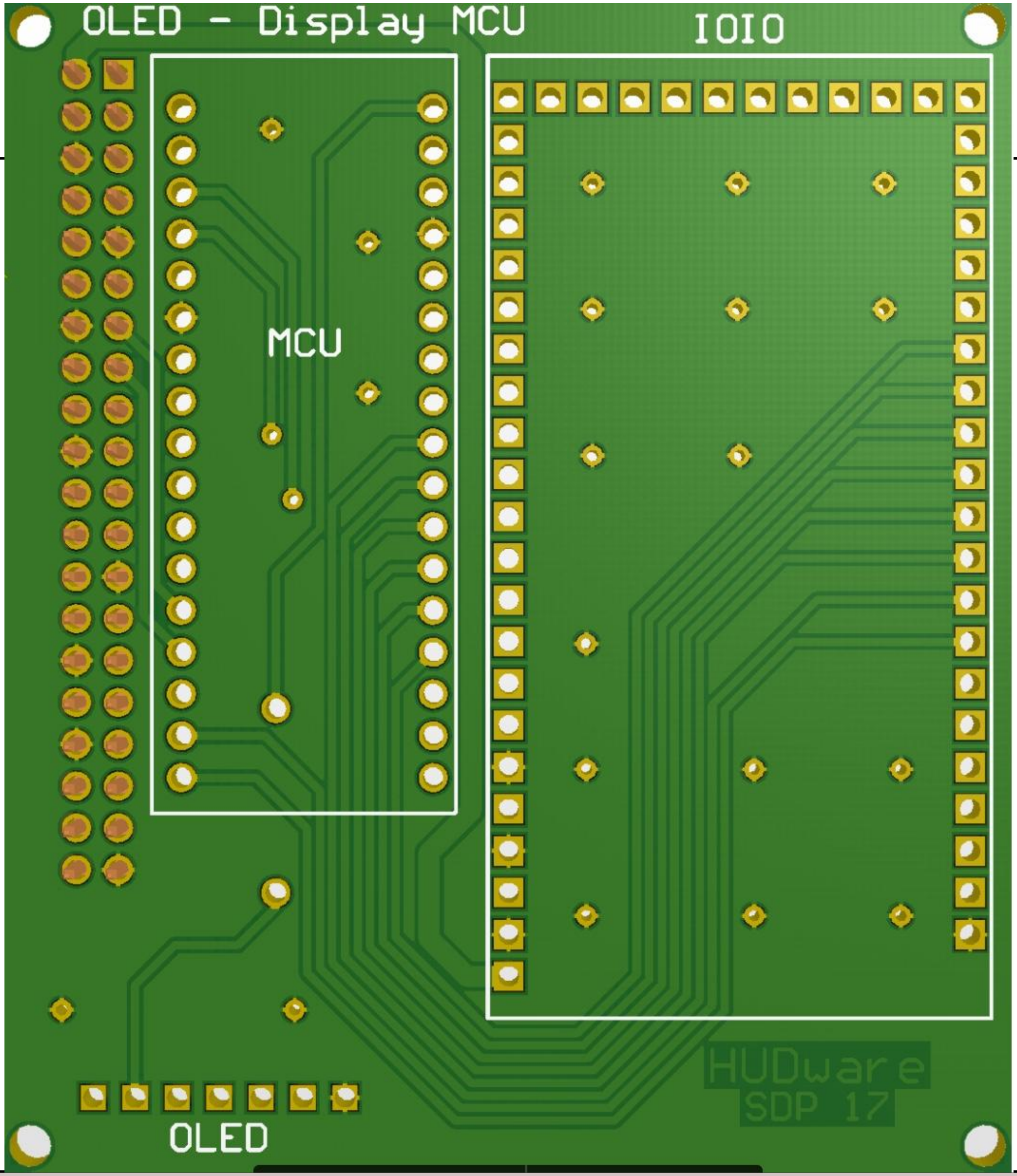
- Implement it by using Broadcast Receiver
- The content of information will be extracted and sent out through IOIO Communication
  - This has been tested for the real phone with SIM card

# Virtual Image



- Need Image to appear to be 25cm away from the back of the eye.
- 10cm FL lens placed 7.5cm from OLED generates a virtual image of 30cm

# New PCB Design


OLED - Display MCU
IOIO
MCU
OLED
HUDware
SDP 17