Augmented Reality User Interface for NASA Spacesuits

University of Central Florida
Department of Electrical and Computer Engineering,
College of Optics and Photonics (CREOL)

Dr. Samuel Richie
Senior Design II

Group 6

David Brown	*Computer Engineering*
Yongsheng Xu	*Electrical Engineering*
Teodor Malendevych	*Photonic Science and Engineering*
Sammy Lee	*Photonic Science and Engineering*

Table of Contents

List of Figures

List of Tables

1. Executive Summary

Ever since man dreamt of going to space, there was a suit design created. In the first human spaceflight program in the United States, Project Mercury, a suit was designed to be worn only inside the spacecraft.The requirements for such a suit were to allow the wearer to withstand low pressure environments. NASA's second mission included activities outside of the spacecraft. A new suit had to be designed to perform these extravehicular activities (EVA). The spacesuit requirements for Gemini included Protection during emergency ejections, intra vehicular activities (IVA), EVAs, and Comfort. These suits were connected to the spacecraft to supply oxygen through a hose.

The previous two projects were set up for Apollo. In the Apollo Program, astronauts would be walking on the moon untethered to the space craft. New boots had to be designed such that they could walk on the moon without discomfort. A life support system as added to the suit in order to remove the necessity of the hose.

As space missions continue, the design of the space suit is changed to fit that particular mission. Currently NASA is pursuing the Artemis missions in which the first woman and next man are planned to explore the moon by 2024. NASA has issued a challenge in which the visual display systems integrate with some modern technology. Teams are tasked with creating spacesuit information displays in an augmented reality (AR) environment.

The requirements for the displays are used to help astronauts complete EVAs. These tasks include Navigation, EVA Systems States, and Geology Sampling. For navigation, the device must accurately guide the user to multiple EVA assets and designated geology excavation sites in real time. The EVA system states require the AR device to interact with a suit port to execute airlock activities and interaction with the suit to display vitals. The AR device must also help in science sampling and geology field notes.

Other considerations will be that of user interface (UI). Included in this is low light conditions due to the nature of the lunar south pole. High contrast conditions also must be considered. Easy controls to allow the user to complete EVA tasks. A requirement of a system tutorial is added to help new users navigate through the UI efficiently and effectively.

2. Project Description

The project description gives better insight into the requirements for the NASA suit project. Some of the requirements were given from NASA's competition page and requirements, others were decided upon by our group to add further functionality to the project.

2.1 Project Background

The Artemis program is a series of missions designed by NASA that will land American astronauts on the moon by 2024. The challenges that limited the Apollo missions still exist to this day. However, the last time we were on the moon was in December of 1972, and the technology has changed drastically since. NASA aims to utilize modern technology to tackle some of the challenges faced on the moon during the Apollo missions.

One of the newer requirements for next-generation spacesuits is a visual display system. The dynamic display system would enable the astronauts to complete their tasks and experiments more efficiently. Yet they have to keep track of their surroundings and what they are doing. That is where displays based on Augmented Reality (AR) shine. Augmented Reality allows the user to still view their environment enhanced with virtual display.

The display would utilize AR to assist the astronauts in navigating the terrain, taking field notes, interacting with lunar payloads, communication with crewmates, and keeping track of their health and vitals. There are many AR Head-Mounted Displays (HMDs) to choose from, which we detail in a further section.

Some other features that will be included are external sensors such as a wrist-mounted optical heart rate monitor, a pulse oximeter, and a temperature sensor. These sensors will measure the display the vitals of the astronaut as well as aid us in testing the HMD.

2.2 Requirements Specifications

The following sections detail the requirement specifications that we as well as NASA is imposing on this project. The NASA SUITS Challenge gave us several requirements for the project, but we are adding other requirements to enhance our device and its ability in the field.

2.2.1 General Requirements

- The HMD will display EVA task instructions.
- The astronaut shall be able to communicate with ground control at any time.
- In case of an interruption, the astronaut must be able to continue the task on hand.
- The user interface shall not permanently impede the astronaut's ability to perform.
- All hand gestures must be operable with EVA gloved hands (like heavy ski gloves).
- The HMD will include a tutorial for users new to the Hololens II and its features.

2.2.2 Navigation Requirements

- The astronaut should be able to locate themselves, their crew, the lander, and any geological points of interest at all times.
- The HMD should help the astronaut navigate from point A to point B
    - Variation A - The only data available will be analog: heading, elevation, and bearing.
    - Variation B - Available data will be in the form of modern-day Global Positioning System (GPS) data.
- The HMD should help the astronaut navigate back to the lander
    - Variation A - Only data available will be distance to lander, heading, and elevation.
    - Variation B - Data will come from ancillary localization systems.
- The HMD should find geological points of interest

2.2.3 EVA System State and Science Sampling Requirements

- The astronaut should be able to interface with a Display and Control Unit (DCU) and Umbilical Interface Assembly (UIA) during an EVA and suit preparation.
- The HMD must display all vital health and safety information to the astronaut.
- A caution and warning system must be implemented to inform the astronaut about spacesuit limits and anomalies.
- The astronaut must be able to access the status of the spacesuit anytime.
- The HMD must display science sampling instructions.
- The HMD must provide a method for taking field notes.
- The HMD will provide a method to capture images and video.

2.2.4 Peripheral Device Requirements

- Any external or additional device must be approved by NASA prior to the test week.
- The device shall communicate with the Head Mounted Display.
- All Tools must be operable with EVA gloved hands (like heavy ski gloves).
- An optical heart rate monitor that is designed with a EVA gloved hand in mind.
- The optical heart rate monitor will measure the heart rate of the user in real-time with ±2 beats per minute accuracy.
- A pulse oximeter designed such that it is not an impedance in the suit.
- The pulse oximeter will measure the oxygen saturation of the user with a maximum 5% error rate.
- A temperature sensor will be implemented to measure the temperature with a maximum 5% error rate.

2.3 Marketing and Engineering Requirements

The following figure details the tradeoff matrix where we identify the customer specifications and engineering constraints. This House of Quality helps the engineers in determining the main limitations with regards to certain aspects of the project. We evaluated how the needs of the project correlate with said limitations.

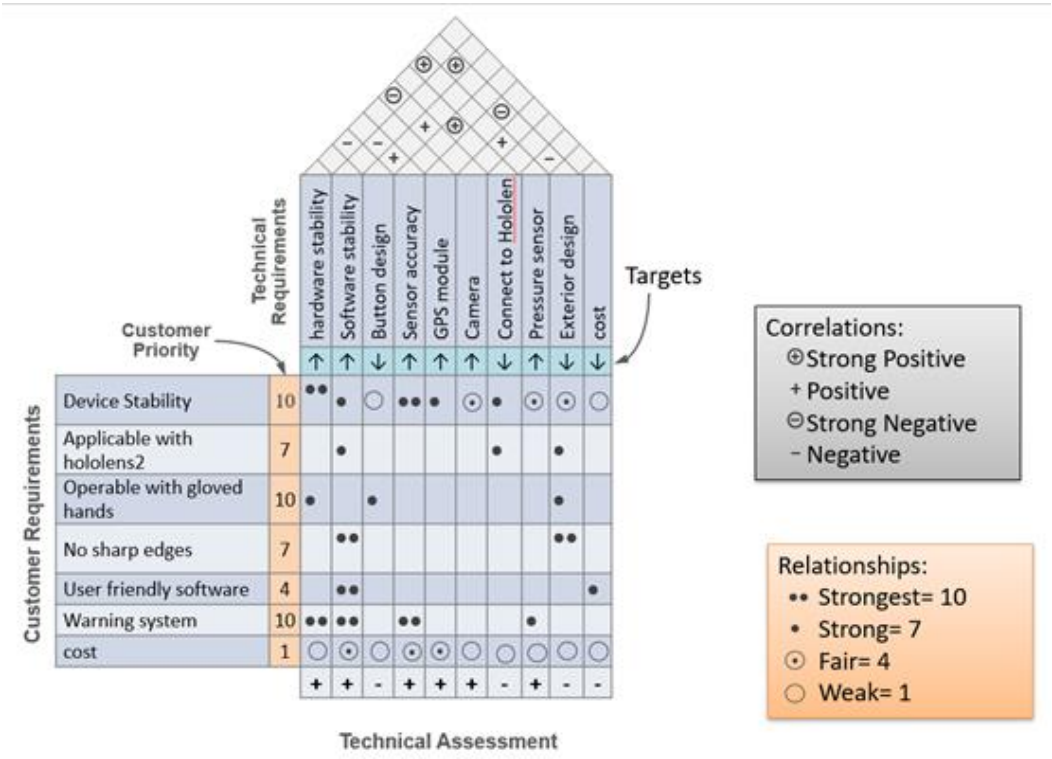


Figure 1: House of Quality

3. Relevant Research

The sections below indicate research that was conducted in order to gain more insight into that project's needs. Many technologies were researched and considered before the design of the project. Many of these topics are related to the NASA suits and Head Mounted Displays.

3.1 Artemis Mission

Back to the moon, this is a dream since Neil Alden Armstrong landed on the moon in 1969, limited by the budget and concern of safety, the project delayed for 50 years. With the Artemis program, NASA will land the first woman and next man on the Moon by 2024, using innovative technologies to explore more of the lunar surface than ever before. We will collaborate with our commercial and international partners and establish sustainable exploration by the end of the decade. Then, we will use what we learn on and around the Moon to take the next giant leap – sending astronauts to Mars.
As planned, this grand plan has three phases named as Artemis I, II, III.

<u>Artemis I:</u>

Artemis I will be used to test NASA's newest deep space exploration system: Space Launch System (SLS) rocket and the Orion spacecraft. In this series of complex missions, Artemis I will be an uncrewed flight to get data to demonstrate the commitment and capability. The flight will be launched on the world's most powerful rocket at Kennedy Space Center in Cape Canaveral, Florida, and travel thousands of miles beyond the moon.

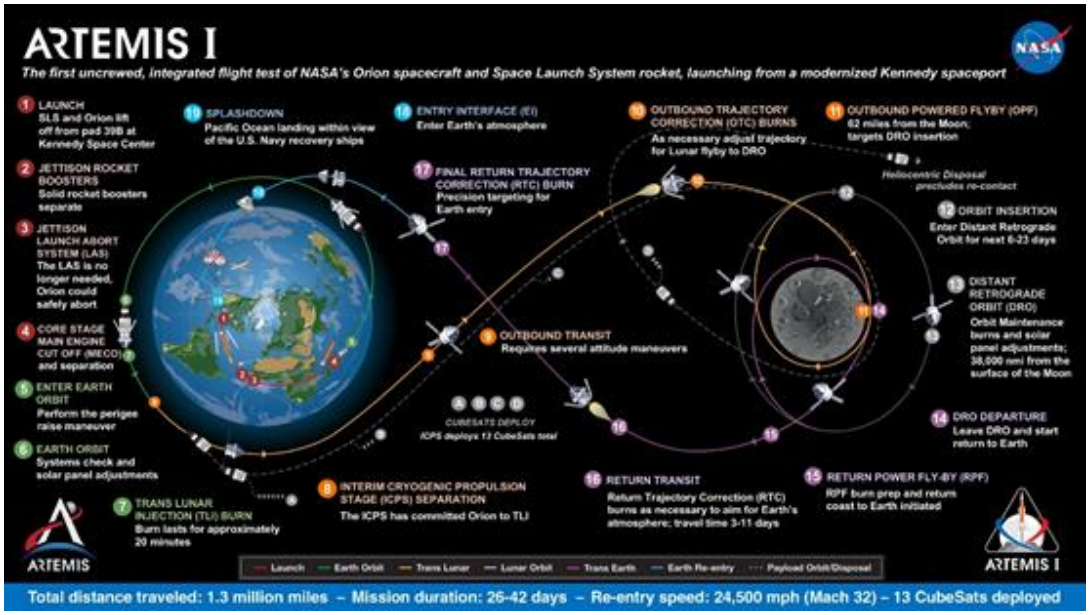


Figure 2: Artemis I flowchart (1)

Artemis II:

Artemis II is planned to launch in 2023, and it will be the first crewed flight that sends four astronauts to the moon in more than 50 years. With confidence gained by Artemis I, the Artemis II will board Orion spaceship atop the space launch system (SLS) for an approximate 10-day trip around the moon in a hybrid free return trajectory.

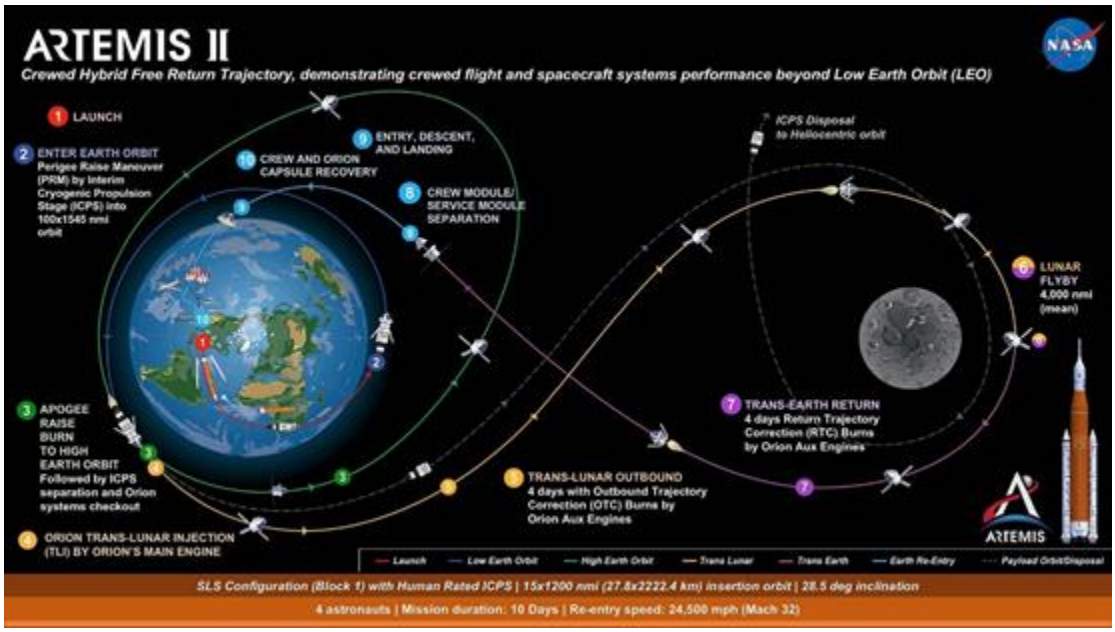


Figure 3: Artemis II flowchart (1)

Artemis III:

Artemis III will be the climax of the whole plan. In 2024, Orion and its crew of four will travel to the moon again, the difference is, this time, the first woman and next man will walk on the moon surface with the Human Landing System (HLS) to do research. According to plan, the rest two astronauts will stay on board the Orion.

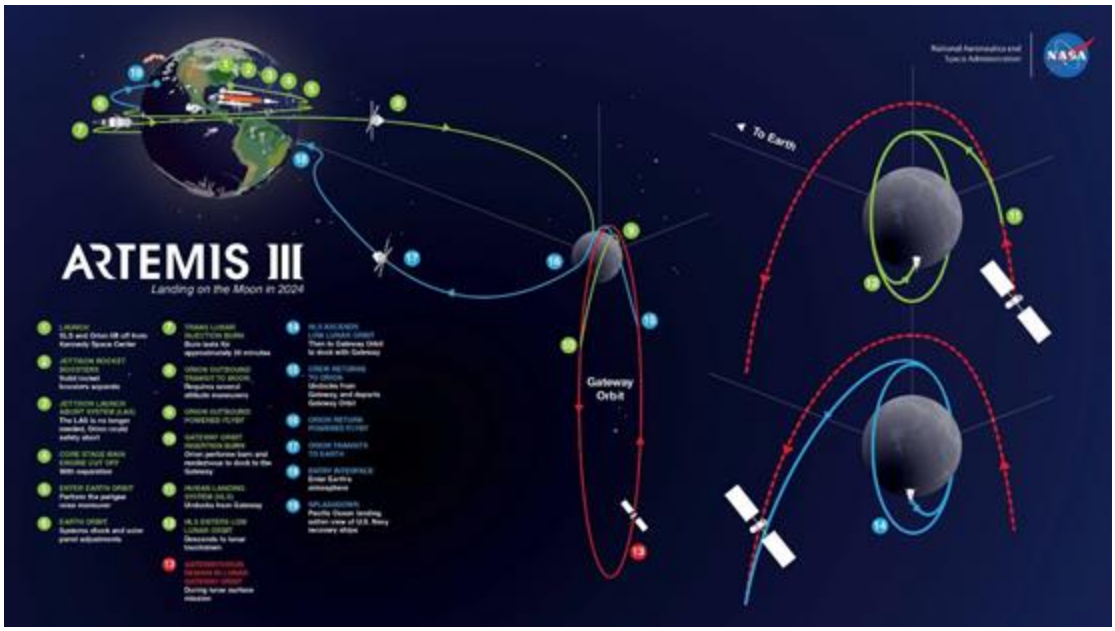


Figure 4: Artemis III flowchart (1)

3.2 Spacesuits

For the 2024 Artemis Mission, NASA astronauts will be wearing the Exploration Extravehicular Mobility Unit (xEMU). This spacesuit is an improvement over the previous Extravehicular Mobility Unit (EMU) that was used for spacewalks outside the International Space Station (ISS) (2). Often called an EVA suit, it kept the astronauts safe and healthy while they worked in hostile environments. The engineers of the suit had to combat extreme temperatures, radiation, reduced atmospheric pressure, micrometeoroids, etc. (3)

To resolve the issue of large temperature changes, the design team included a cooling garment in the suit. 300 feet of tubes filled with water are fitted into the spacesuits covering virtually all of the astronaut (2). The new xEMU suits will be able to withstand extreme temperatures of -250 to 250 degrees Celsius. (3) This cooling system will receive temperature information and compensate for the astronaut's safety and comfort.

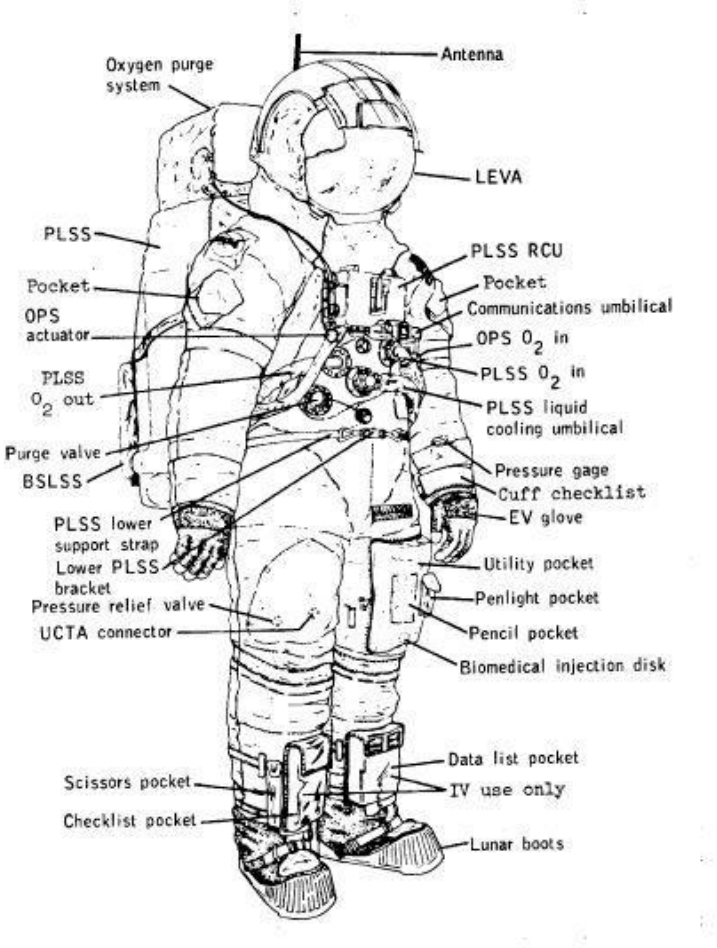


Figure 5: Apollo Mission EMU Schematics

The backpack attached to the suits is aptly named the Portable Life Support System (PLSS). The system supplies oxygen and water for breathing and cooling needs. It additionally pressurizes the suit to 3.9 psi. (4) One of the most important parts of the PLSS is the communication and telemetry module. The crew members can communicate with each other, the Lunar Module, and Mission Control. Suit telemetry data is communicated to the lunar module. Nine measures are communicated: Oxygen supply pressure, suit water inlet temperature, sublimator oxygen outlet pressure, suit pressure, feedwater pressure, suit water temperature rise, CO_2 partial pressure, and backpack battery current and voltage. (4) These measures are vital for the astronaut's survival on the moon. If any of these resources run low, an audible warning is initiated.

3.3 AR Technology

This section examines different types of current and some past devices which allowed AR perception. It examines how we chose which device to use and some of the technology that is used in the devices.

3.3.1 Different Realities

Altered perception is current in demand technology. Its use ranges from entertainment to education and many subjects in between. There are several types of altered perceptions, but the only three that will be discussed are Augmented Reality, Virtual Reality, and Mixed Reality (14).

Augmented Reality (AR) mixes our reality with digital data. It can be in the form of images, text, video, or a combination of them. It incorporates Machine Learning, Computer Vision, and Artificial Intelligence. It utilizes a camera and Inertial Measurement Unit to gather data of its surroundings. To analyze data, a point cloud and Spatial Mapping are formed to map out the location of the device with relation to its environment. Pattern Matching is incorporated into the software to help recognize objects in the environment. Pattern Matching utilizes Image recognition to help identify objects. Image recognition can be expanded to room or environment recognition to identify a 3-D space. With these tools, a digital object can be placed in a digital environment that seemingly interacts with one's physical environment (17).

Virtual reality (VR) is a virtual environment which can be interacted with. Unlike AR, the environment is completely computer generated. Its main objective is to immerse the user into the created environment through perception and other senses (15).

Mixed Reality (MR) is a combination of AR and VR. It combines the spatial awareness of AR and the virtual objects of VR. It blends the physical and digital world. An example of this is seen in some sci-fi movies. The protagonist will look at a map, and a holographic 3-D version of the area will appear (16).

3.3.2 Considered AR Devices

The following AR devices were considered for this project. The NASA SUITS Challenge did not limit us in the headset we choose. However, we do know that NASA has a predisposition towards Hololens and Hololens II and possibly Magic Leap One.

HoloLens 2

The features of the HoloLens 2 is that it has the highest field-of-view on the market with 43°. It includes a 29° vertical (5). For head tracking it uses 4 visible light cameras. It has an eye sensor which incorporates 2 infrared cameras.It senses depth with a 1-MP time of flight depth sensor. An accelerometer, gyroscope, and magnetometer are included and make up the inertial measurement unit. A built in camera that can capture 8-MP stills and 1080p30 videos is included in the head unit. For computing it uses a Qualcomm Snapdragon 850 Compute Platform. Which is supported with a holographic processing unit, which is a secondary processing unit that helps integrate real time data with its augmented counterparts. To the computing system 4GB of DRAM and 64 GB of storage are added to the system. It weighs 566g (8).

Magic Leap One

Magic Leap has a slightly lower field-of-view than Hololens at 40° and a vertical of 30° (6). It can capture videos and stills at 1080p with a resolution of 1440x1080 (10). For computing it uses a separate wearable device called the Lightpack. The Lightpack uses a Nvidia Parker Chip as its CPU. It includes a Nvidia Pascal for a GPU. It has 8 GB of Ram and 128 GB of storage. A haptic device is added for controls. This haptic device has 6 degrees of freedom (9). It is touch sensitive and has buttons.

Epson Moverio

The Epson Moverio has a field of view of 23° diagonally. It uses a 5MP camera to shoot 3D videos and take depth measurements. It makes use of an inertial measurement unit, a gps, and an ambient light sensor. It has wifi and bluetooth capabilities (11). It comes with a controller that controls selection and connection type. It makes use of an attachable microphone. The headpiece weighs ~300g and the controller weighs ~350g (12).

Google Glass Enterprise Edition 2

The Google glass is a portable AR system that can be attached to the glasses or goggles. The Google Glass uses a Qualcomm Quad Core, 1.7 GHz, 10nm. It uses android as its operating system. It includes 3GB LPDDR4 Ram and 32 GB of eMMC Flash storage. It has wifi and bluetooth capabilities. The Google glass includes a 8 megapixel camera. It weighs ~46g (13).

Focals by North

Focals by North uses a projection built into the side frame of the glasses. It uses a Qualcomm APQ8009w with a Arm Cortex (32-bit) at 1.09 GHz for computing and an accompanying Qualcomm Adreno 304 graphics processing unit. It connects to devices via bluetooth. It weighs ~80g.

Dream Glass 4K

This AR headset is made by DreamWorld and has the following specifications:

- Display: 90-degree FOV, 2.5K resolution, 60Hz.
- Tracking: Hand gesture recognition, 3 degree-of-freedom head tracking.
- Cameras: 1080p RGB camera w/microphone, IR camera
- Weight: ~200 g
- Software Development Kit: Unity-based SDK
- Cost: $599

Figure 6: Dream Glass's sleek design makes us question how much computing power the headset has. (79)

These specifications have been listed on their website. (79) However, we are somewhat skeptical especially with regards to their larger FOV compared to most competitors. We will be collaborating with the Institute of Simulation and Training at UCF to test this HMD before we consider this a sufficient back-up in the case of limited funding.

Selection

The HoloLens2 was chosen due to its capabilities and field of view. The Google Glass Enterprise Edition 2 seemed capable. Its performance was adequate for our needs. The main motivator behind not selecting the device was that the device only projects on the right side of the screen you choose to project it on. The Focals by North is less capable than the Google Glass. It also only projects on the right lens of the glasses frame. The Epson Moverio and the Magic Leap One were eliminated due to their use of peripheral controllers. The requirement for the challenge would be such that the peripheral device can be used with gloved hands. Out of these two devices, the only one that seemed to be able to fit that requirement was the Magic Leap One. The decision to completely remove the peripheral remote was decided. HoloLens 2's gesture controls can be used without need of a peripheral device. The only downfall of any helmet like device is that it may be uncomfortable to wear compared to the glasses.

As an alternative choice, the Dream Glass 4k was chosen based on availability and price. The Institute of Simulation and Training have allowed us to demo it in their lab. The demo is scheduled for 12/9/2020. The price of the Dream Glass is $600-$800 depending on whether we want the extra features or not.

The main concern with using this device is in its processing power. We plan to use real time image processing. The concern lies in the processing time of the algorithm used. This could create a lagging change which could make the image processing unusable in a moving environment. This could cause trouble to the astronaut. Take the example of using edge detection for holes or craters on the moon, if the detection lags behind too much, we could see the astronaut in the crater. If the processing speed is inadequate, we can still attach a secondary processing device like pi to do the image processing.

3.4 Microsoft Hololens II

The Microsoft HoloLens 2 is the second generation product to come from microsoft's initial expedition into the mixed-reality smartglass world. Hololens 2 attempts to build on the innovations of the first HoloLens by focusing on improvement in two main areas; user immersion within the mixed-reality virtual environment and comfortability which does not decay with time.

Microsoft added a few key innovations to the latest HoloLens design in order to improve user immersion. One of the biggest innovations was a changing from a LED display to a laser-based MEMs display which increased the diagonal field of view (FOV) from 34 degrees to 52 degree, with a resolution of 47 pixels per degree, or total 2K resolution for each eye. With this display, holographic images are encoded into lasers which are shot through a "waveguide" within the headset, reflecting off a chain of mirrors and relaying the image directly into the user's retina (Figure 7). The relay mirrors are composed of both "fast scan" and "slow scan" which scan at frequencies of 12000 Hz and 120 Hz, respectively. The fast scan mirror is responsible for creating the horizontal part of the image while the slow scan uses the laser to create the vertical. The image size can be adjusted by changing the angle of the reflective mirrors within the waveguide.

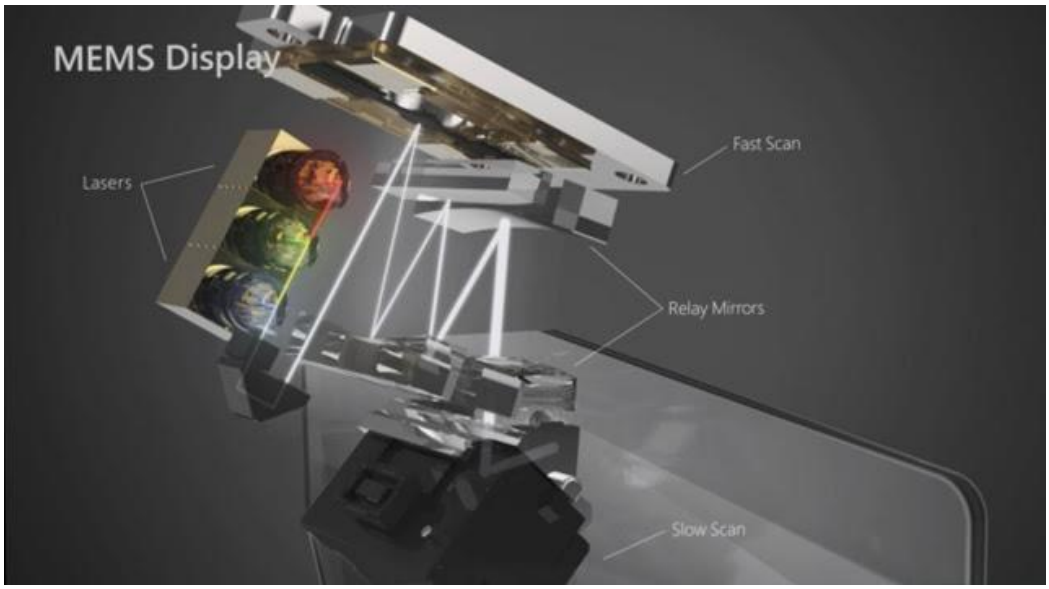


Figure 7: The MEMS Display of the Hololens II

The first HoloLens model used head tracking to orient the images within the user's field of view. Unfortunately, with just head tracking, images on the edges of the display could "clip" out of view since the HoloLens did not account for the degrees of rotational freedom afforded to the eye. The second generation HoloLens has a built-in monochromatic camera for tracking eye movement by illuminating the eye with infrared light and tracking the orientation of the iris through time. Addition of eye tracking to compliment head tracking improves the field of view drastically as well as increases user comfort through stabilizing the holographic images. Additionally, the headset uses this data for user authentication through iris recognition which Microsoft claims to have a false positive rate of 1 in a million.

Unlike most smartglass and mixed-reality headsets, the HoloLens does not require an external device for sensing the environment, as it comes with a built-in fourth generation Azure Kinect for both depth perception as well as spatial mapping. The Kinect scans the environment using two cameras (8MP stills, 1080p video) and phased time-of-flight analysis (TOF). Photons are emitted from the device, reflect off the surfaces in the field of view, and are absorbed by the headset to approximate the distance of all surfaces within the local landscape.

The built-in Kinect also records hand gestures which are processed and determined by the device's software. The first HoloLens was limited to sensing only one hand gesture, when the user "clicked" a virtual button. The second generation HoloLens, in contrast, understands a wide range of intuitive hand gestures such as clenching fists, pinching virtual objects, poking objects, etc. The emphasis on producing a device which allows for control through intuitive movement drastically increases user immersion by reducing time spent learning how to use the headset and reducing awkward unnatural movements while interacting with the virtually displayed landscape.

The combination of the aforementioned sensing mechanisms built into the HoloLens creates a weight and depth to the virtual objects displayed on the HoloLens screen. All of the sensor data is collected by the Qualcomm Snapdragon 850 CPU (ARM RISC architecture) and processed using artificial intelligence algorithms to tailor the device to a particular user. To boost the computational power, the headset also contains Microsoft's second generation "Holographic Processing Unit (HPU)" with 125Mb SRAM, 7x SIMD Fixed Point Processor (SFP) for 2D processing, 6x Floating Vector Processor (FVP) for 3D processing, and PCIe 2.0 x1 bus which communicates with the CPU at a bandwidth of 100 MB/s.

3.4.1 Microsoft Mixed Reality Software Development Environment

A few key pieces of software are required for application development within the Windows Mixed Reality Development environment. The HoloLens 2 runs a variant of the Windows 10 operating system which has been specifically modified for producing holographic images as well as implementing other key features of the device. As such, applications must be developed and prototyped on Windows 10 OS, using Visual Studio 2019 IDE and the Windows 10 software development kit (SDK).

While the choice of operating system and development platforms are limited to Microsoft's proprietary software, the engine responsible for creating the virtualized experience contains a few options, dependent on the goal of the application being developed. The most common platform for designing applications is the Unity engine. The Unity engine is a powerful 3D graphics engine commonly used in game design. The software is written in C++ and allows developers to design applications using C#. The open source mixed reality toolkit (MRTK) is used as an API to interface with the Windows 10 Holographic OS, wrapping the data produced by the Unity engine, and making system calls to Windows Holographic APIs (HolographicSpace API, OpenXR API). The end result is an immersive XR experience for the user application. The Microsoft XR APIs are extensive containing functionality for gazes, gestures, holographic rendering, spatial mapping, voice control, and motion control.

Two other options for producing XR applications are Buildwagon and Native OpenXR. Buildwagon is an online development platform specifically created to allow web developers to quickly develop and release HoloLens applications. The developer creates the application entirely using javascript code, which is then converted to C++ using Buildwagon's HoloBuild library, which makes API calls to Windows 10 Holographic API. This is the standard when it comes to developing web-based applications for the HoloLens. However, given the objectives of this project, this platform is unlikely to be useful.

Native OpenXR platform uses OpenXR to develop using DirectX 12 and C++ which is a standard created by Khronos group as a royalty-free, cross-platform way to develop XR. This depth and breadth of VR/XR/MR proprietary devices on the market created a situation in which developers were consistently forced to learn a new set of tools for each device causing a huge burden on the XR industry when it came to development time and money. To alleviate this, OpenXR was created, which has been given the stamp of approval by most of the big tech firms, who have agreed to follow the standard and create APIs to interact with the Native tools for that device.

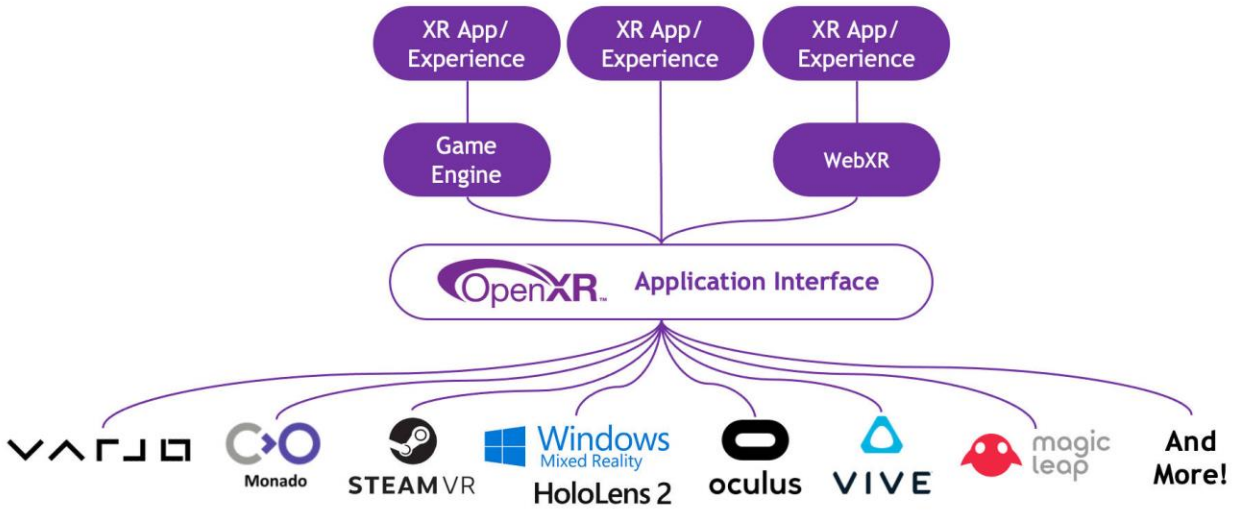


Figure 8: OpenXR API allows standardized XR development

In the case of the HoloLens 2, which contains firmware written in C++ and ARM assembly, an OpenXR API exists for interacting directly with the device. As such, developers who choose to develop using Native OpenXR have the ability to create their own middleware or framework for interacting with the headset through Windows Mixed Reality APIs.

The set of software development tools best suited to achieve the outlined objective appears to be Native OpenXR. This is especially true given our need for peripheral sensors, which must sense and interpret the local environment, box the data into an easily transmissible data format, and make API calls directly to the HoloLens OS. Luckily, our project goals do not require very detailed renderings of holographic images, but must respond and perform quickly in real-time. Thus, given the need to integrate external sensors, a set of tools which directly interacts with the device using low-level languages like C/C++ will be vital for performance.

On top of the performance benefits of using Native OpenXR compared to other platforms, it is now possible to access data from the HoloLens internal sensors using the recently developed "Research Mode" thanks to effort made within the Computer Vision community. As previously mentioned the HoloLens internal firmware is written with C++. Thus, to truly use the data granted to us by Microsoft and improve the quality of the application software for astronauts, we will need to use low-level languages like C/C++, because those are the languages used within the headset to make system calls to the OS as well as measure sensor data.

Finally, should we need higher-level functionality, such as Machine Learning or Computer Vision, we could integrate higher-level languages into our pipeline such as Python or Java. Python is the gold standard for machine learning thanks to the wealth of libraries which have been developed for data analytics and machine learning. If such an approach were needed, we could leverage the ability of languages required on the hardware side (C/C++) to listen to

communication through open sockets from external sensors, and add a small CPU such as a Raspberry Pi to alleviate the computation burden on the system. This would benefit our design two-fold by drastically increasing performance using low-level compiled languages while exporting a lot of the computation lifting onto an external CPU which becomes better through time.

3.5 Optical Heart Rate Monitoring and Pulse Oximetry

In order to measure the heart rate and oxygen saturation levels of the user, we will be designing a wrist-mounted optical heart rate monitor. These devices utilize photoplethysmography to measure vital health information. It is a relatively low-cost and non-invasive method. This technique shines short-wave low-intensity infrared light through the surface of the skin. The light is absorbed by bones, veins, arteries, and skin tissue. (17) Blood (both in veins and arteries) is the largest absorber of the infrared light. Arteries typically have more blood and are usually the target for photoplethysmography devices. Figure 9 describes the absorption of different components of the skin.

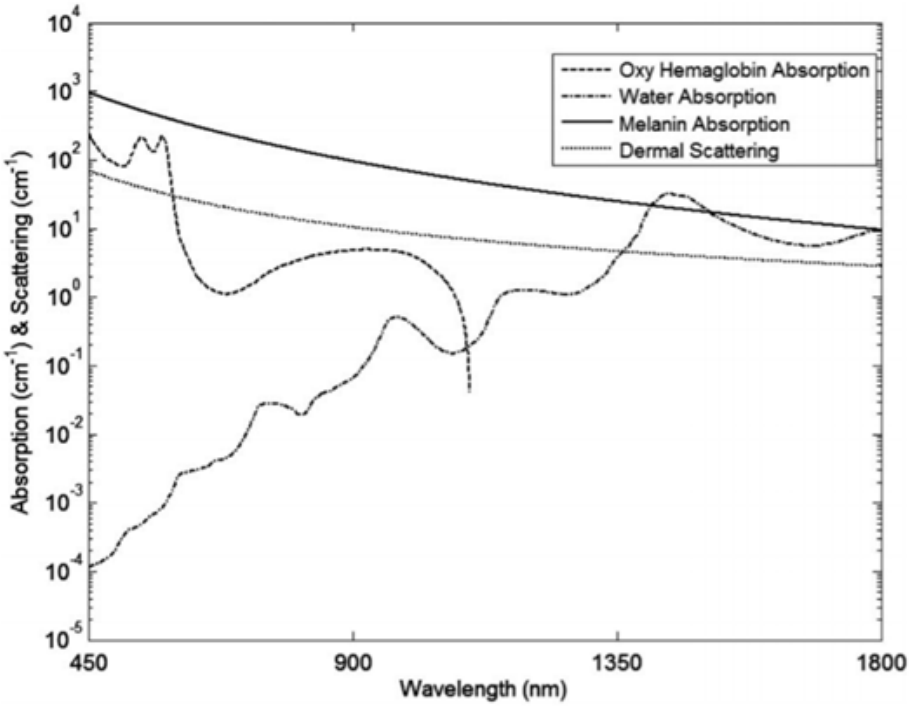


Figure 9: Absorption spectrum of Oxy Hemaglobin, Water, Melanin. Overall dermal scattering is plotted as well. (18)

As can be seen in Figure 9, there are two possible source wavelength ranges that are worth considering. The first range is 450 – 600 nm. This range contains the lowest water absorption and the highest hemoglobin and melanin absorption. We have two concerns with this range: The increased melanin absorption might result in virtually no light making it to the artery, the cost and scale of light sources in this range would make it harder to include in a wrist-mounted device. The second range, 800 – 1000 nm, contains a high hemoglobin absorption coefficient while absorption from melanin is lower than in the visible spectrum. Additionally, the near infrared (NIR) range is very attractive because it is a very economically efficient range in terms of LED and laser sources. However, the higher absorption coefficient of water in this range might result in sensitivity issues.

Figure 10 displays the process of photoplethysmography. The detected waveform has two parts. The DC steady state signal comes from the absorption of skin tissue as well as any other non-pulsating part of the epidermis. The AC part of the signal comes from the pulsating blood. By measuring the time difference between two crests of the AC signal, we can calculate the heart rate of the user.

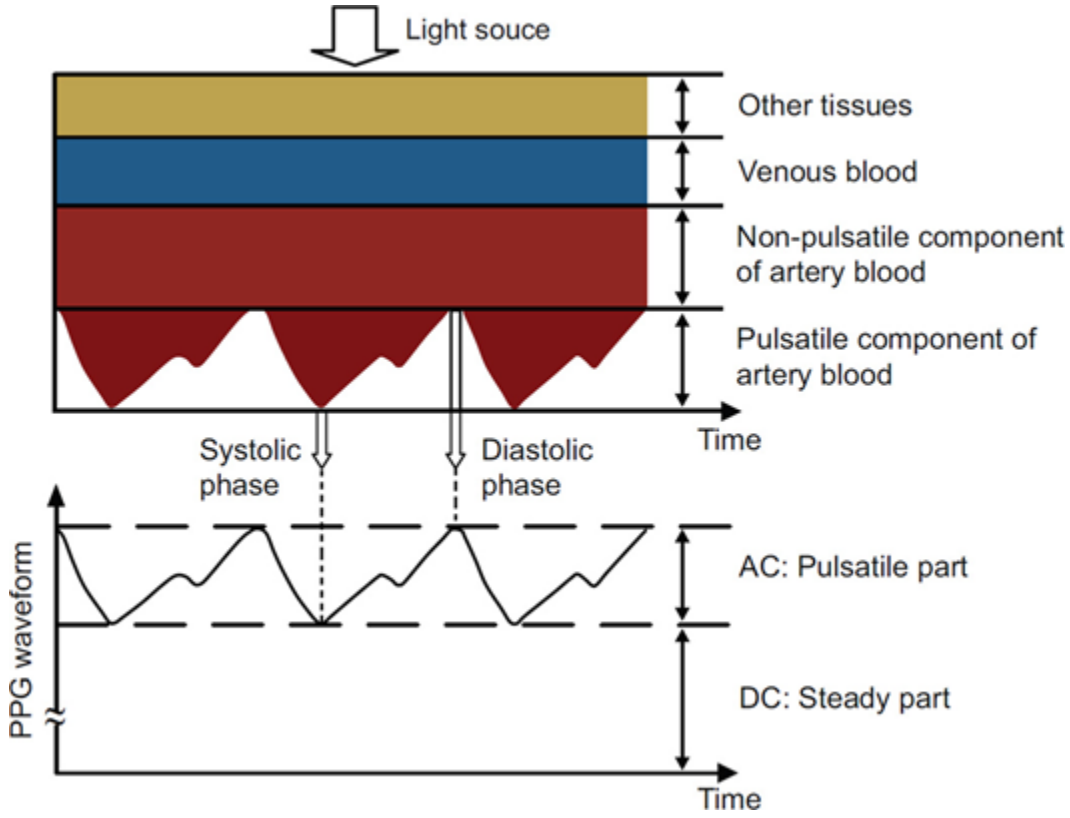


Figure 10: Useful diagram to demonstrate the AC and DC parts of a PPG signal.

Additionally, photoplethysmography can be utilized to measure the oxygen saturation (SpO_2) of the blood. SpO_2 is the ratio of oxygen-saturated versus total hemoglobin. The measurement of SpO_2, also known as pulse oximetry, is dependent on the absorption spectrum of oxyhemoglobin (HbO_2) and deoxyhemoglobin (Hb) in the visible and infrared spectral regions. Utilizing two LED sources in the sensitive 800-1000 range and one in the less sensitive 600-700 nm, we can measure two photoplethysmography signals as displayed in Figure 11.

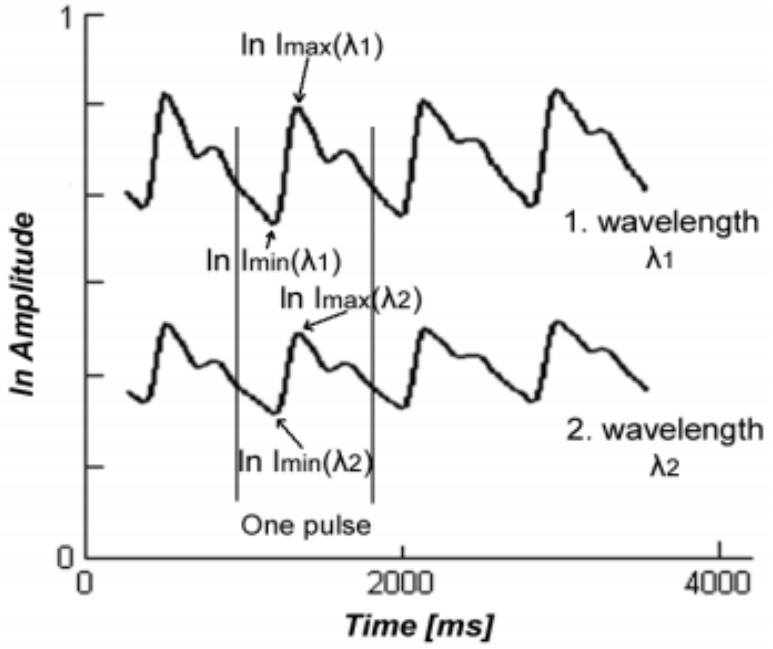


Figure 11: Reflected signal of two wavelengths measured by a photodiode (19)

From the max and min of both wavelength signal, we can calculate the value R from the following equation deduced from the Lambert-Beer Law:

$$R = \frac{\ln \frac{I_{max}(\lambda_1)}{I_{min}(\lambda_1)}}{\ln \frac{I_{max}(\lambda_2)}{I_{min}(\lambda_2)}}$$

Utilizing an off-the-shelf pulse oximeter, the ratio R can be mapped empirically to the oxygen saturation percentage. Modern pulse oximeters measure SpO_2 with less than 2% error and 1% resolution. Worse accuracy is also accepted by clinical staff due to 4% dispersion of the blood oxygen saturation level between healthy human subjects. (19) An example of a calibration curve of R vs SpO_2 can be shown in Figure 12.

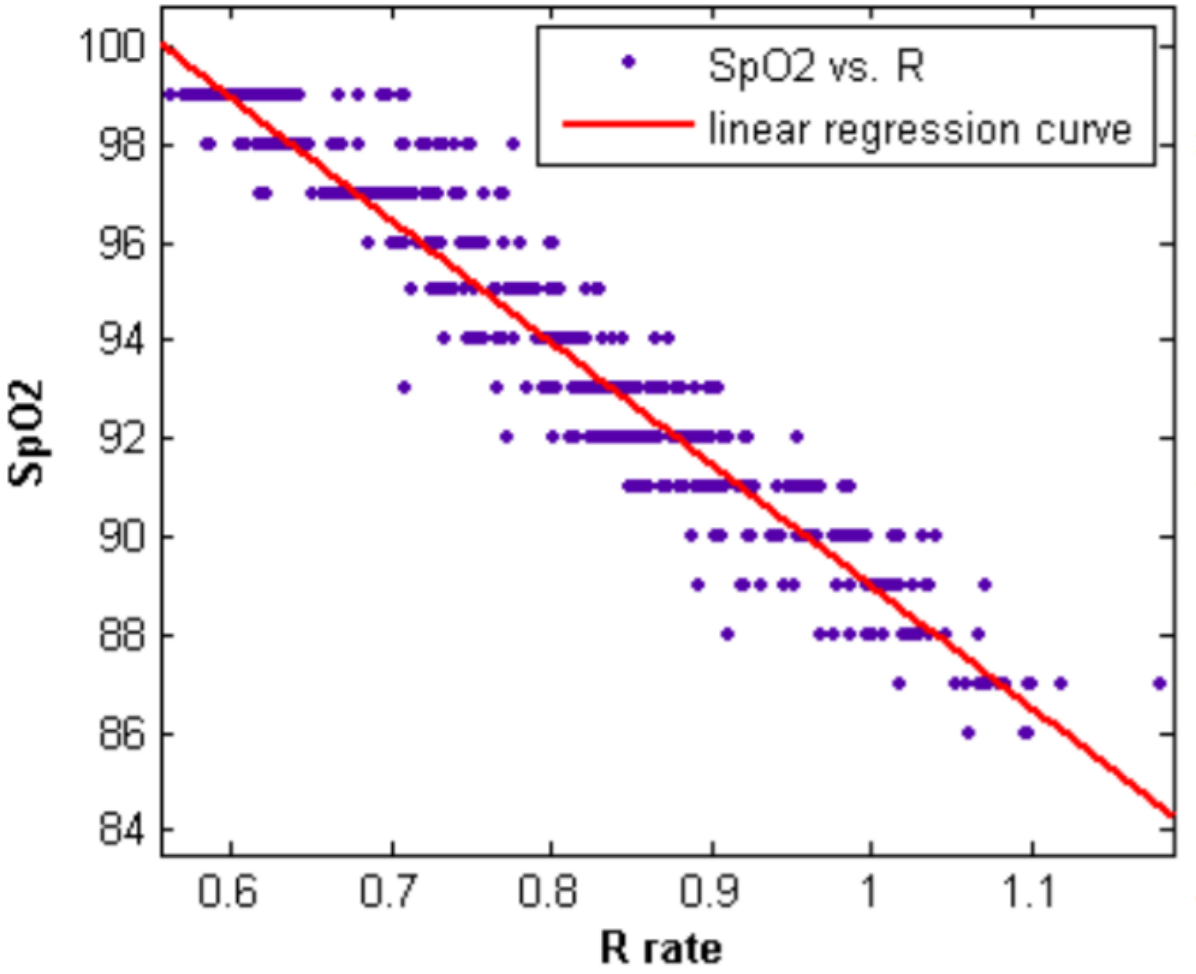


Figure 12: Calibration Curve of SpO2 vs R. The resolution of the pulse oximeter used is 1%. (19)

3.6 Image Processing and Enhancement

This section describes the theory of image processing. There will be several techniques listed below which will then be tested later in the document and compared. The basis of image processing is in manipulating pixel values to extract more information if possible.

Theory

For the lighting enhancement, the solutions were separated into two categories, hardware and software. The hardware solution came down to the choices of night-vision via image intensifier, IR vision, or LIDAR.The software solution is using image processing techniques. By using the components already in the HoloLens, data can be collected and manipulated to help the astronaut through image enhancement techniques.

These image enhancement techniques take the data and process it through different algorithms to try and enhance the image in a deliberate way. The techniques are separated into two fields, Spatial Domain Methods and Frequency Domain Methods (32). The Spatial Domain methods deal directly with the images pixels (33). The pixels are changed in order to create an enhancement. In the frequency domain, the fourier transform is utilized to change spatial data into frequency components. The image's frequency is manipulated and then returned into the spatial realm with an Inverse Fourier Transform.

Some Spatial domain techniques include gray-level transforms, histogram processing, and enhancement through arithmetic operations.

3.6.1 Gray Level Transforms

Gray transforms are one of the simpler image processing techniques. They are gray level because the theory only considers transforms from 0 to 255 pixel intensity. There is no consideration for color, which would be a 3x255 array at its simplest in RGB or CMY color space.

Image Negative

The image negative technique reverses the intensity levels of an image to produce an equivalent negative image. It is more suited for when there is white or gray detail intertwined into black regions. It is expressed through $s = L - 1 - r$, where s is the modified pixel values, L is the intensity levels, and r is original pixel values.

Log Transform

A log transform tries to suppress the higher image values and adds more weight to the lower values. This technique is used when the brightness of one spot can result in diminishing the values of every other spot by a relative scale. The transform is expressed through $s = c\ log(1 + r)$ where c is a constant, s is the modified pixel values, and r is the original pixel values.

Power-Law Transformations

The power-law transform gives adjust the values in a similar way like the log transform. It gives a family of curves which can be selected to adjust the final image. It is expressed through $s = cr^{\gamma}$ where s is the modified pixel value, c is a positive constant, r is the original pixel value, and γ is a positive constant (34).

3.6.2 Piecewise-Linear Transforms

The piecewise transforms act similarly to piecewise functions. They will hold something constant and can eliminate other pixel values in an area. They generally will use a threshold and can manipulate different planes of the image.

Gray Level Slicing

Gray level slicing, also known as intensity-level slicing, takes a part of the intensity range and examines it more closely. It can be used to emphasize a certain range of intensities or diminish that range. It works similar to a bandpass filter. (35)

Contrast Stretching

Contrast stretching works on the principle of picking two points on an input-output intensity curve and manipulating values in between. Let these locations points (r_1, s_1) and (r_2, s_2) control the shape of the transform. If $r_1 = s_1$ and $r_2 = s_2$, the change produces no change in the gray levels. If $r_1 = r_2, s_1 = 0$, and $s_2 = L - 1$ where L is the maximum intensity level a threshold function is created. Intermediate values between the two points produce a spread of gray values. Generally speaking $r_1 \leq r_2$ and $s_1 \leq s_2$. For contrast stretching $r_1 = r_{min}, s_1 = 0$, $r_2 = r_{max}$, $s_2 = L - 1$ where r_{min} denotes the minimum gray levels in the image and r_{max} denotes the maximum gray levels.

Bit-plane Slicing

Bit-plane slicing tries to isolate the main components of the image and emphasize it. It can keep the background unchanged or try to mute them so there is more contrast. In essence, the pixels of the object are intensified to bring greater contrast.

3.6.3 Histogram Processing

A histogram is a graph of the intensity levels in a discrete function. It's related to the probability of a certain value of appearing. The function is defined as $h(r_k) = n_k$. where $r_k$is the k_{th}intensity level in the interval [0,G], $n_k$is the number of pixels in the image whose intensity level is $r_k$and G is the size of the image class. A normalization can be obtained by dividing all elements in $h(r_k)$by the total number of pixels in the image.The normalization is expressed as $P(r_k) = h(r_k) / n = n_k / n$ for k=1,2,..L where L is the total possibly intensity levels in that range and n is the number of pixels.

Histogram Equalization

The equalization method increases the dynamic range of the gray-levels in order to cover the full range of grays. It achieves this equalization with a Cumulative Distribution Function (CDF) given a Probability Density Function (PDF). A histogram can be considered an approximation of a PDF of that image. The continuous case is expressed as $S = T(r) = \int_0^r P_r(w)dw$ where w is a dummy variable of integration. The output level is defined as $P_s(s) = \{1 \quad 0 \ll s \ll 1, 0 \quad otherwise\}$
For a discrete case
$S_k = T(r_k) = \sum_{j=1}^{k} P_r(r_j) = \sum_{j=1}^{k} n_{j/n}$, for k=1,2,....L (36)

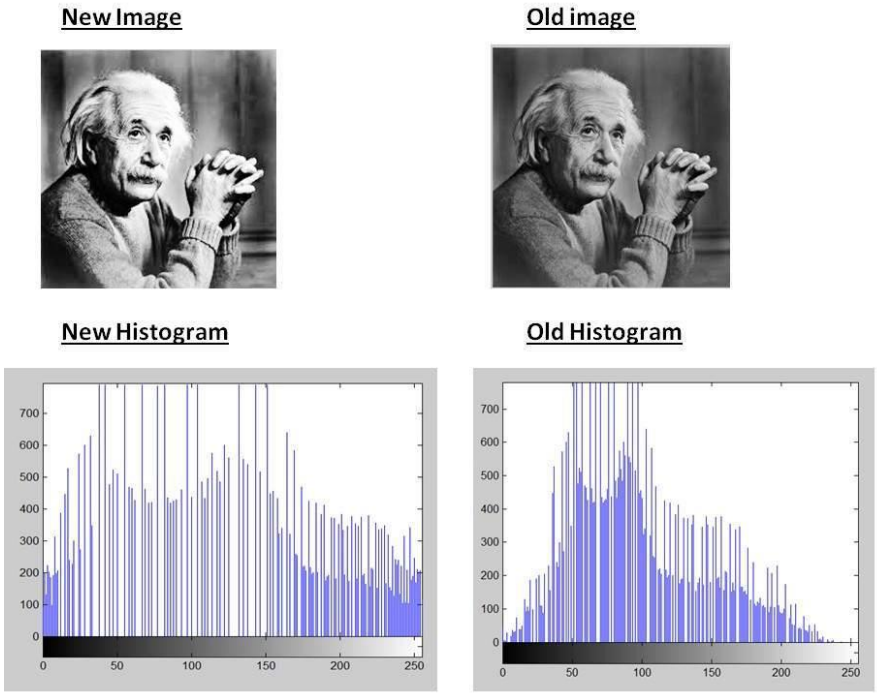


Figure 13: Example of histogram equalization

Histogram equalization works well with images that contain a distribution of pixel values with low variance. This technique struggles with images containing a highly variant pixel value distribution. The brightest and the darkest regions in the image will not see their contrast enhanced well.

Histogram Matching

Histogram matching uses the above techniques to try and match a histogram of one image to another histogram of another image. This technique modifies one image's contrast to the other (37).

Local Enhancement

Local enhancement uses the histogram techniques above but in a small localized area. This area is defined by a mask. The histogram of this mask is then taken and adjusted so that in other areas the pixel values remain unchanged. This can lead to higher contrast in an area with fairly similar values without completely changing all the images pixel values (38).

3.6.4 Contrast Limited Adaptive Histogram Equalization (CLAHE)

CLAHE is a derivative of Adaptive Histogram Equalization (AHE), which seeks to divide the image into multiple tiles and performs histogram equalization on each tile individually. AHE on its own has issues with noise amplification and CLAHE was designed as a solution.

In short, AHE transforms the values of each pixel based on the surrounding pixels. For each pixel, the AHE algorithm performs a histogram equalization of a sub-region of the image centered on the pixel and with a parameter of tile size given by the user. (80) The process is succinctly shown in Figure 14.

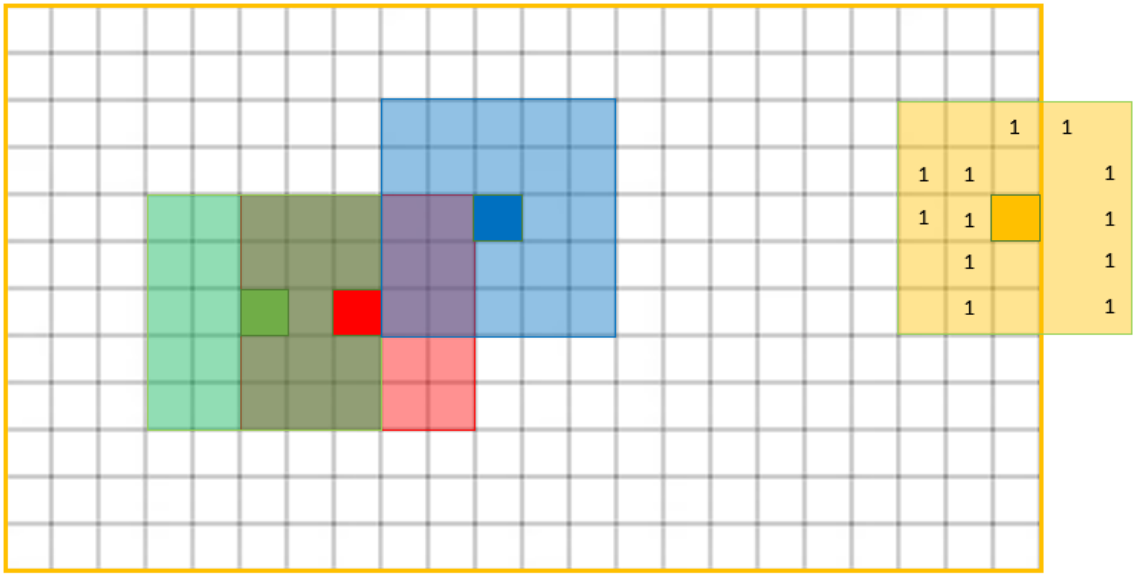


Figure 14: Diagram explaining how AHE and CLAHE performs histogram equalization on each pixel. In this example, the size of the region being equalized is 5x5 as specified by a user. The pixels on the edge are special cases. To perform proper histogram equalization, the image is mirrored with respect to the edge of the image, as shown by the binary orange pixel values above.

The need for CLAHE comes from noise amplification of regions with similar pixel values. With histogram equalization, these similar pixel values would redistribute to fill the entire range of the image. Artificial noise is therefore created in the image. (80) CLAHE presents a solution by introducing a parameter known as clip limit. Those problematic homogeneous regions will have a peaked histogram. Any part of the histogram above the clip limit will be redistributed equally into the rest of the histogram as seen in Figure 15.

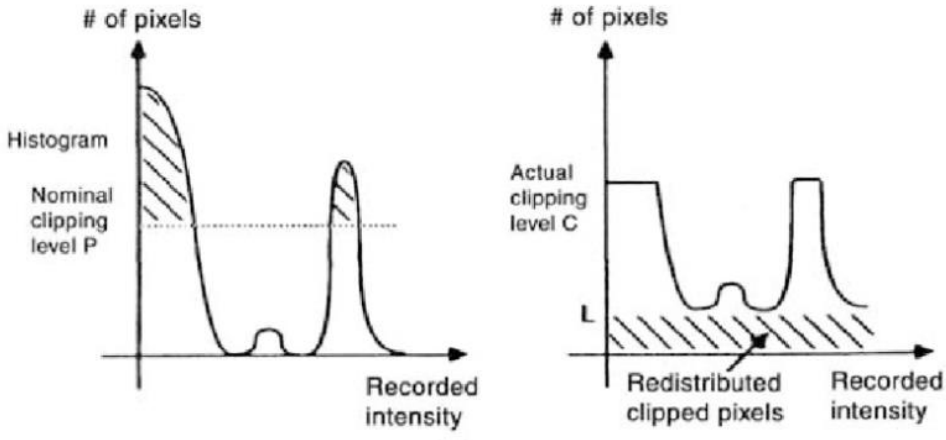


Figure 15: Clip Limit of CLAHE visualized. (81)

3.6.5 Frequency Domain Techniques

Low-pass filters

Low-pass filters are filters designed such that only the low part of the frequency are transferred. It cuts-off any frequency higher than a specific chosen frequency called the cutoff frequency. In imaging applications, the low pass filter acts like a blurring or smoothing filter. It averages out rapid changes in the intensity. It can be used to correct for noise. Two examples of these filters are the Butterworth low-pass filter and the Gaussian low-pass filter. The Butterworth filter is designed such that the transition after cutoff frequency is as little as possible.The transition is meant to be abrupt. The Gaussian filter tries to make the transition smoother. The Butterworth filter is described as $B(f) = 1/(1 + (f/f_c))^{2n}$ where f is the spatial frequency domain, $f_c$is the cutoff frequency, $n$is the order of the filter (39). The filter's response is seen in figure [X1]. As the order of the filter increases so does the steepness of the response. A Gaussian filter is defined by $H(u, v) = e^{-f^2/2*f_c^2}$, where f is the spatial frequency domain and $f_c$is the cutoff frequency (40). The response is seen in figure 13. As the cutoff frequency which is labeled $D_o$is increased, so is the slope and the range of allowed frequencies.

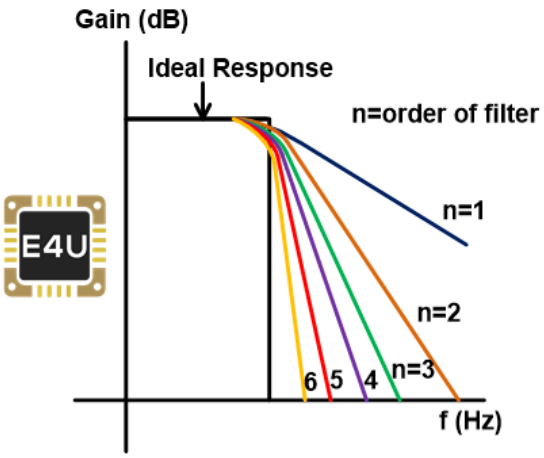


Figure 16: A graph comparing the ideal response to the Butterworth filters response. Taken from (41)

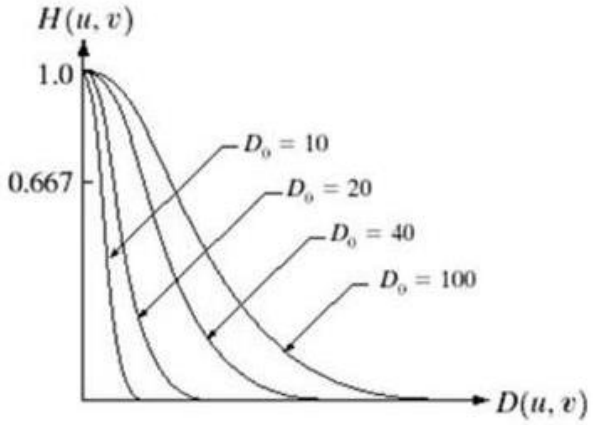


Figure 17: A graph of a Gaussian low-pass filters response. Taken from (42)

High-pass Filters

High pass filters are used to cut off low frequencies and only allow the higher frequencies to pass. In the spatial domain, this results in edge enhancement or edge detection. An ideal high-pass filter is described as $H(u,v) = 1$ if $f > f_c$and 0 if $f \leq f_c$. The response is shown in Figure 14 where $D(u,v)$is the bandwidth. It can be seen that at a certain point, the cutoff frequency, there is a transition into allowing a full transfer of everything past that point.

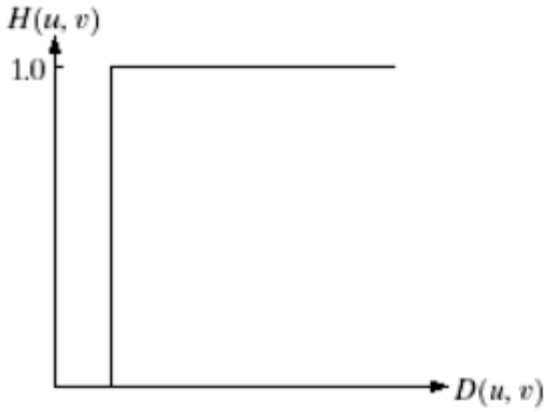


Figure 18: An idealized response of a high-pass filter. Taken from (43)

Similar to the low-pass filter, there is a Buttersworth and Gaussian version of both. Depending on the applications and the order of filters either filter will do better. In one study the Buttersworth filter did significantly better with a cutoff frequency of 100 and an order of 4 (44). The choice of filter will be decided based on calculation time, image improvement and usability.

Band-pass Filter

The band-pass filter retains only the frequencies in the pass band. It is similar to a combination of both the low-pass filter and the high-pass filter. It reduces the noise by attenuating the high frequencies. It enhances the edge by suppressing low frequencies. The response can be seen in Figure 19. The band pass filter can act like a mask in the spatial domain.

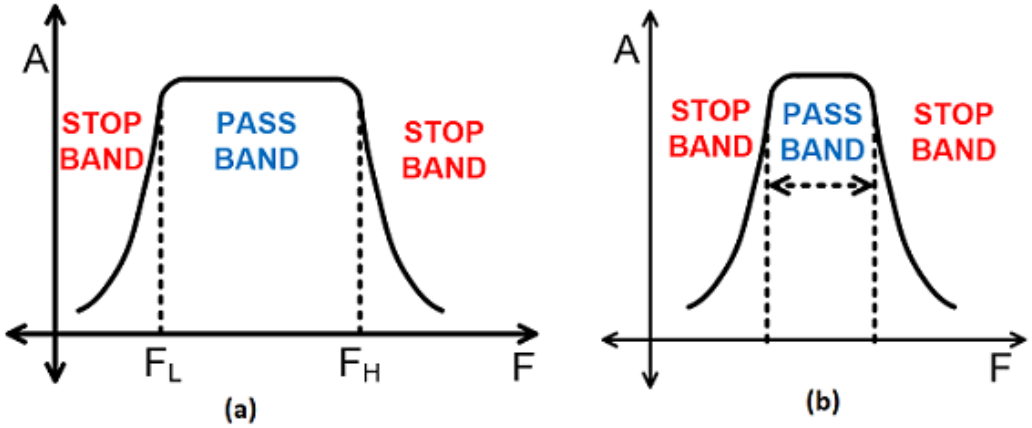


Figure 19: a) A wide band-pass filter b) A narrow band-pass filter Taken from (45)

For our illumination task, image processing was chosen based on its usability in most situations. The techniques will be tested and examined to determine which technique will be used. The concern with using image processing is in situations where there is no data to be modified. These situations occur when there is no light in the area or the light is too dim for a camera to detect. This will result in no information and thus no amplification or modification will be useful. In these situations an IR camera may be used instead of a CCD camera. These techniques will still work on IR data. The colors on the moon aren't very vibrant so the idea to use grayscale images was decided upon for image processing. This allows the algorithms used to deal with information in a 255x1 matrix instead of a 255x3. The use of grayscale will decrease processing time.

3.7 Wireless Communication

The need for external sensors to supplement the HoloLens to achieve the objectives for the UI outlined by the NASA SUITS challenge means we must implement communication channels between the assorted sensors (IR camera, temperature, optical heart rate monitor, night vision, etc) and the HoloLens. Our design will likely be constrained on how the data flows between the HoloLens and sensors given the restrictions and difficulty we will face trying to integrate our components through directly wiring. Moreover, it is possible our sensors, due to the constraints of the spacesuit, must communicate with the microprocessing unit wirelessly. Therefore, a detailed understanding of how wireless communication works and the potential use cases for different types of wireless communication are necessary.

Wireless Communication Theory

Computational networks can be subdivided into those which are connected physically (guided) using cables or those which are remotely connected (unguided) and communicate through propagation of electromagnetic waves. Movement of electrons through space generates electromagnetic waves which are composed of perpendicular sinusoidally oscillating magnetic and electric fields. The parameters used to measure these waves are frequency (f) which corresponds to the number of oscillations per second (1/s = 1 Hertz) and wavelength which is the distance between the maxima of two consecutive waves (1).

The electromagnetic spectrum is composed of multiple types of waveforms dependent on their frequency; radiowave (lowest energy/frequency), microwave, infrared, visible light, UV, X-ray, and gamma ray (highest energy/frequency). Each type of wave can be used to carry information and therefore is useful for wireless communication. However, radio waves are the most commonly used part of the spectrum for communication because they are easy to create, move long distances, move easily through objects, and do not have potential long-term health consequences with increasing exposure (1,2).

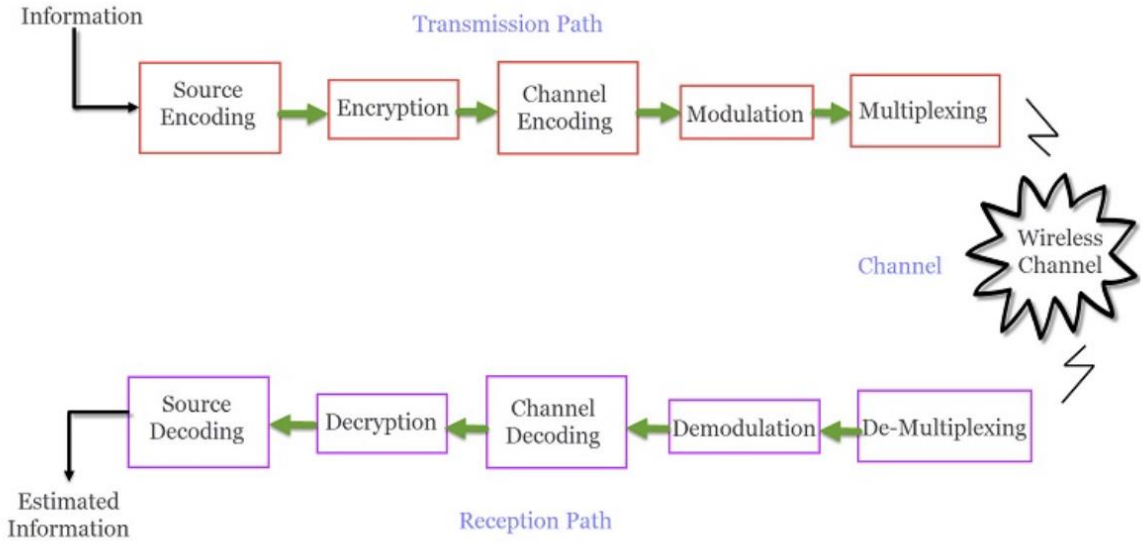


Figure 20: wireless communication network design (2)

Wireless communication networks are typically designed around three components; transmitter, channel, and receiver Figure 20. The transmitter receives information from a source, the information is encrypted, and converted into a suitable form for transmission through an antenna by amplitude, frequency or phase modulation. Finally, the signal is multiplexed using Frequency Division Multiplexing (FDM) or Time Division Multiplexing (TDM) for saving bandwidth across many signals. The electromagnetic wave propagates from the transmitter guided through the "channel" until it reaches the receiver. Given the unguided nature of the transmission of electromagnetic waves, it is important to note that the channel is often a huge source of distortion and noise due to interference between the environment (objects in the way) or other waves travelling through the channel at the same time. At the receiver end the signal is de-multiplexed, de-modulated, de-encrypted, decoded and finally read by the receiver.

The ease with which wireless communication allows data transfer has caused an explosion in the types of wireless communication dependent on usage (sending video, messages, voice, etc). A few of these systems are Television/Radio, Satellite, Mobile Telephone Systems, Infrared, Global Position System (GPS), Radio Frequency Identification (RFID), Radio Frequency (RF), Bluetooth, ZigBee, WLAN (WiFi). Wireless communication systems can be divided based on their electromagnetic wave frequency range (IR, Radio, etc), but also based on the direct information flow. For instance, a Full Duplex system allows consistent communication between both the transmitter and receiver while a Simplex system allows communication to

flow from transmitter to receiver only. A half-duplex system allows communication both directions, but only one direction at a time.

Wireless Communication Design Constraints

Considerations when designing (or integrating into a system) a wireless communication system include the infrastructure of the system (size, shape, cost, etc) and their ability to reliably transfer data (bandwidth, latency, data rate, interference, etc). The wireless communication network we integrate into our project is highly limited to only certain types of communication due to the project objectives. Obviously we cannot leverage the existing internet, cable, and phone infrastructure for our design which will be used on lunar missions. But another big consideration, given the lunar environment, is the potential for interference from electromagnetic waves from space. It is vital that communication channels between the sensor network and the HoloLens remain intact, as the astronaut's life could depend on accurate, error-free data.

Further considerations for choosing a design for communication between our sensor network and the HoloLens is the ability to reliably transfer data. A few important metrics are relevant when considering data transfer; bandwidth, data rate, and latency. The bandwidth of a component within a network refers to the range (difference) of upper and lower frequency values corresponding to a discrete signal. Higher bandwidth channels carry more data per second. Data rates are dependent on the bandwidth of the system. As bandwidth goes up, so do data rates, since more data can be sent across the network. Data rates are dependent on a few other factors as well such as receiver/transmitter distance, power, and antenna size.

Finally, latency, or the time between transmission of the signal and reception of the signal, must also be considered. Electromagnetic waves, in the vacuum of space, should travel at the speed of light. On Earth the propagation of EM waves from receiver to transmitter appears almost negligible, because the distances are small. However, as the distance between TX/RX increases, even the speed of light takes a significant amount of time to reach its destination. For instance, a wave traveling from Earth to Mars has a latency of 4-24 minutes, depending on where each celestial body is in it's relative orbit (3).

HoloLens Wireless Communication

Another limit to our wireless communication system design is the built in wireless connectivity of the HoloLens, which comes integrated with Bluetooth LE 5.0 and 802.11 2x2 WiFi for wireless connectivity. Bluetooth LE (5.0) is an IEEE 802.15.1 standard for wireless communication using radio waves (2.4 - 2.483GHz) and GFSK modulation (12). It is designed for transmitting data across a small range (10~ meters) at high data rates (~1Mb/s) with low

power consumption (11). Bluetooth Low Energy (BLE) standard can be used for months without recharged batteries with a power consumption of 0.1W when idle (6). Bluetooth is also a cheap option as a wireless communication module for a project.

WiFi (WLAN) is an IEEE standard (802.11) operating within radio wave frequency range of 2.4GHz. It has data rates which at worst double the data rate of Bluetooth. However, WiFi has a larger power consumption of approximately 1W when idle (6) compared to Bluetooth. It works by connecting an intermediary access point, a router, which acts as a stationary access point between a local wireless network (50-100 meters wide) and a much larger physically connected network. The router acts as a "middle man", receiving data from either the local or non-local side of the network and directing it to its specific address.

WiFi is a possible solution for the design of our communication network by creating a web server on a microcontroller and attaching a WiFi module. The benefits being a much higher potential data rate between the microcontroller and HoloLens and potential to form a much faster communication network (within 50-100 meters) when multiple astronauts are working outside the lunar lander. However, the benefits might be marginal at best given the requirement specifications, increased cost, and increased power consumption compared to other options which are just as viable.

Wireless Communication Between Sensors and Microprocessing Unit

The HoloLens design constrains our computer communication network slightly by requiring wireless communication between the microprocessing unit and HoloLens which must be WiFi or Bluetooth. However, when it comes to our communication between the microprocessing unit and sensors, we are much less limited.

For instance, one potential route for designing our wireless communication between sensors and the microprocessing unit is by using ZigBee. ZigBee was designed based on the way in which Honey Bees communicate, "dancing" to alert other Bees of potential pollen sources relative to their current position (5). ZigBee is an IEEE 802.15.4 standard for wireless communication using radio waves similar to Bluetooth. Like Bluetooth, it operates on the 2.4GHz frequency, but uses different modulation techniques (BPSK/QPSK) for transmitting the signal (12).

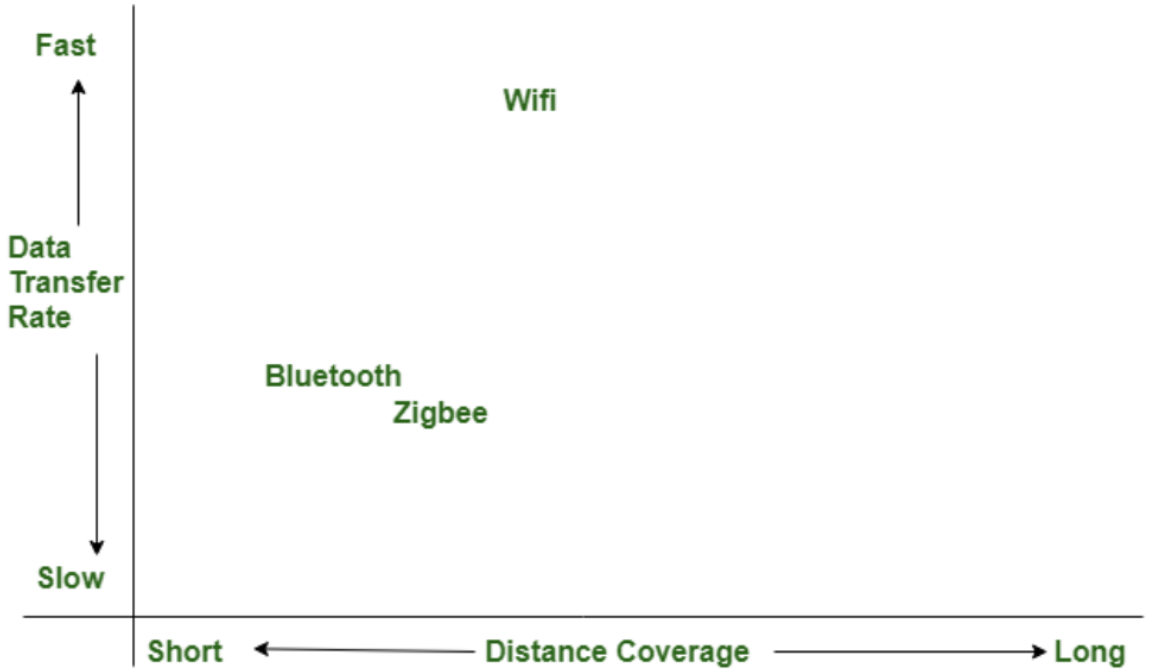


Figure 21: Comparison of data transfer rates and distance for three 2.4GHz wireless communication standards (7)

The distance ZigBee networks communicate exceed Bluetooth (Table 1), but have lower data transmission rates. Both standards use the same power consumption when idle, 0.1W (6). One benefit compared to Bluetooth, is the distance between "nodes" within a ZigBee network exceeds Bluetooth by an order of magnitude (10m for Bluetooth, 100m for ZigBee). So while the benefits, compared to Bluetooth seem negligible and ZigBee transmits data at lower rates, the ability of a ZigBee network to create communicative systems naturally between astronauts and rovers could prove beneficial.

	Bluetooth (BLE)	ZigBee	Wifi
Distance	10 meters	10-100 meters	50-100 meters
Data Rate	1 Mb/s (6)	250 Kb/s	2-54 Mb/s
Modulation	GFSK	BPSK/QPSK	QPSK
Power consumption (when idle)	0.1W	0.1W	1W
Frequency Range	2.4-2.483GHz	2.4GHz	2.4GHz-5GHz
Standard	IEEE 802.15.1	IEEE 802.15.4	IEEE 802.11

Table 1: Potential computer communication network standards

Due to the project constraints, the most likely avenues for creating communication between sensors, microcontroller, and the HoloLen are radio waves in the 2.4GHz frequency range. There are a few other options within the radio wave frequency range, especially when it comes to communication between sensors and the microcontroller. The United States has agreed upon a standard (1) where devices consuming power less than 1 watt (Figure 22) can use anything within the range of approximately 9MHz-5.86GHz (1).

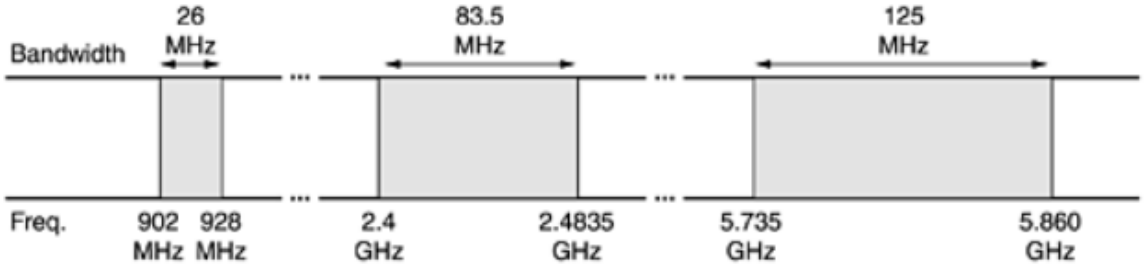


Figure 22: Industrial, scientific, medical (ISM) bands for wireless communication (1)

Serial Communication Protocols

Communication between the various sensors and microprocessing units must occur either in parallel (byte-by-byte) or serial (bit-by-bit). Given the possibility of multiple sensors and limited number of processing power, the best route is likely serial communication. There are many standards for communication serially depending on the structure of the sensor and microprocessor as a cohesive unit.

One of the most common schemes for serial communication between devices is Universal Asynchronous Receiver and Transmitter (UART). In this serial communication protocol, there is no shared clock pulse between the device and microcontroller (4). The transmitter controls voltage level on the shared data wire. When it sends a signal, it changes the voltage on data wire, which signals to the receiver to expect data. The data is transmitted on the rising edge of the clock and received on the next possible rising edge of the clock by the receiver. One potential problem using the UART communication protocol is the potential for oversampling. Basically, with oversampling, too many samples are taken by the receiver, which is functioning too high of a frequency compared to the transmitter. Still, this is one of the most common standards for wireless communication between devices.

Two other protocols are common to the sensor/microcontroller systems we plan to use; inter-integrated circuit (I2C) and serial peripheral interface (SPI). Both are synchronous and can transmit more data than UART using a leader/follower architecture, but in slightly different ways (4). The I2C protocol uses two wires and a bus topology to communicate. Two wires are used, one containing the serial clock (SCL) and another being the serial data (SDA). The "leader" device controls the SDA line, keeping it high until it wants to send data to devices along the line.

Serial peripheral interface (SPI) works similarly with a leader/follower architecture. However, unlike UART/I2C it is not a standard for communication amongst devices and as such should only be used when design isn't constricted by the use of devices with potentially changing and evolving formats. Regardless, it's a possible communication protocol with particular uses, and should be explored as a potential avenue for implementation within our design.

SPI works based on 4 communication wires and allows communication between all connected devices (full duplex). The four wires are as follows; master out/device in (MOSI), master in/device out (MISO), serial clock (SCLK), and chip select (CS; default to active low). With this communication protocol, like I2C, the leader device controls the clock signal with all other devices listening for data on the trailing edge of the clock. This protocol is great for daisy chaining devices together since the leader can send information within a single clock cycle. A comparison of all discussed serial communication protocols can be found in Table 2.

Serial Communication Standard	Clock Transmission	Average Data Rate	Number of Signal Wires
UART	Asynchronous	20kb/s	2
I2C	Synchronous	1Mb/s	2
SPI	Synchronous	25Mb/s	4*

**Can be more than 4 SPI signal wires when daisy-chaining*

Table 2: Comparison of common serial communication standards

4. Proposed Hardware Design

4.1 External Sensors

To aid the user in the completion of tasks and monitoring the user's vitals, we found it necessary to implement external sensors to provide additional information. These external sensors will transmit the data wirelessly to the HoloLens, where any computation necessary will be done. The data will then be displayed to the user in the corresponding UI screen.

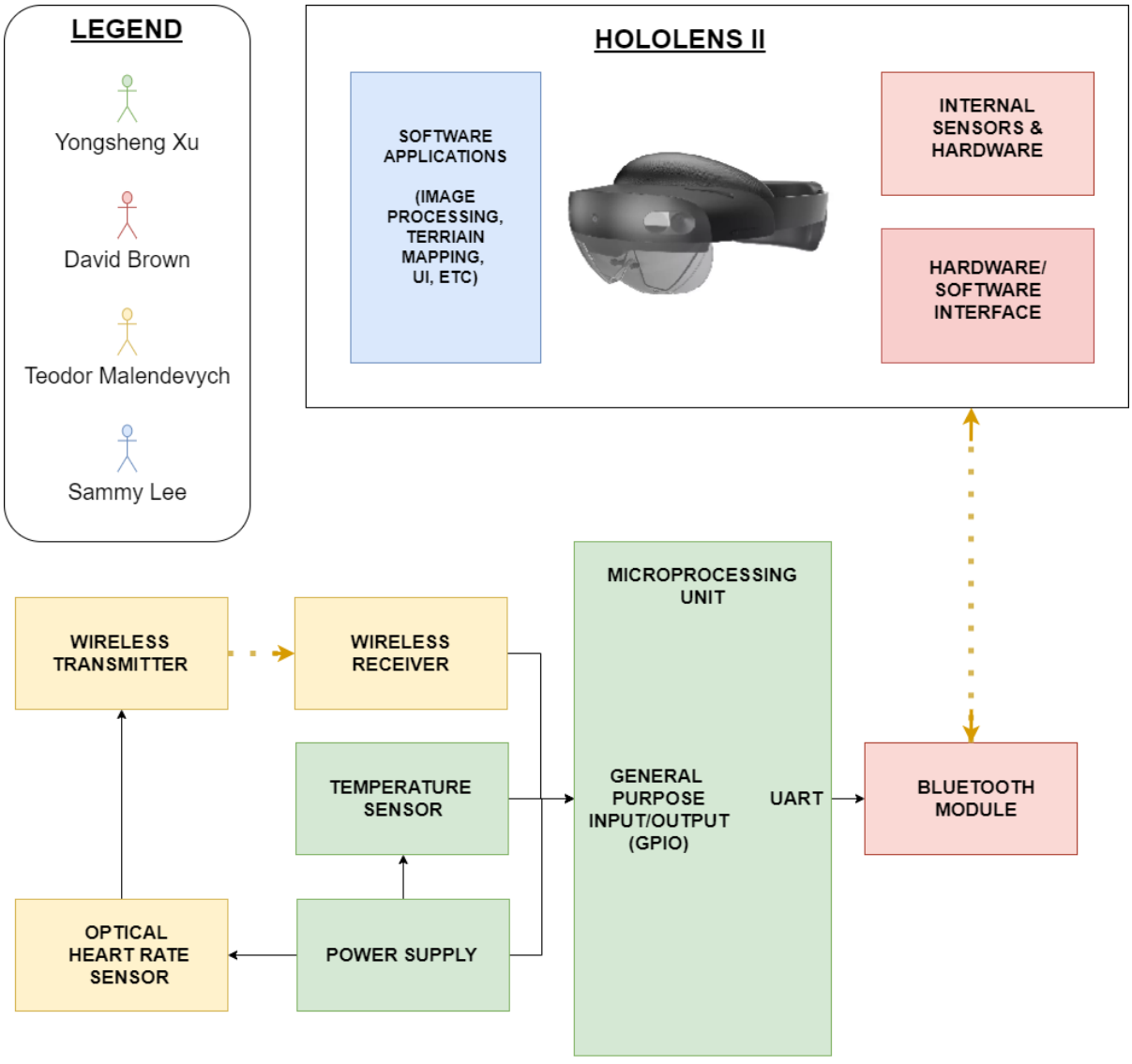


Figure 23: Hardware Block Diagram

The general design of our proposed hardware solution is outlined in Figure 23. The signals are either hardwired into the microprocessing unit through general purpose input/output pins (GPIO) or delivered to the microprocessing unit by radio wave transmission at 433 MHz. For instance, the optical heart rate sensor will deliver signals by transmitting from the sensor's location to a hardwired 433 MHz RF receiver, which decodes the message and sends the output to the GPIO pins of the microprocessing unit. The microprocessor will then process the inputs and send them, using UART, through a bluetooth module at approximately 2.4GHz. The transmitted signal will be received through the internal Bluetooth module built into the HoloLens II. The signals will be decoded, then processed, and then converted into the proper data format to be fed into the applications built into the HoloLens. A combination of the internal sensors and the internal sensors within the HoloLens will be processed by application software contained within the HoloLens harddrive.

4.1.1 Temperature Sensor

Temperature is an important measurement for astronauts on the moon for experimentation as well as monitoring hazardous conditions. A small temperature sensor will be purchased and utilized in testing the wireless communication capabilities of the HoloLens. The sensor’s real-time data will be displayed in the UI and accessible to the user at any time. An integrated temperature sensor will be included due to its small form factor. It can be housed with a small wireless chip to keep a low profile. The sensor chosen is Maxim DS18S20.

Maxim DS18S20 Features (61):

- 1-wire interface that requires only 1 port for communication
- 9 bit thermometer resolution
- Temperature conversion time of at most 750 ms
- Alarm command if the temperature is outside of programmed limits
- 3 pins
- Sensing temperature of -55°C to 125°C
- Accuracy of ±0.5°C
- Cost ~$5

4.1.2 Optical Heart Rate Monitor

Part Selection

IR LED: Vishay Semiconductors VSMY2943G, Emitter Wavelength: 940 nm, LxW: 5.8 mm by 2.3 mm, cost: $1
Red LED: Cree XLamp XP-E2 LED, Emitter Wavelength: 650-670 nm, LxW: 3.5 mm by 3.5 mm, cost: $1.88
Photodiodes: Osram LPT 80A Phototransistor, Spectral Range of Sensitivity: 450 - 1100 nm, LxW: 5 mm by 6 mm, cost: <$1
Filters: Edmund Optics Optical Cast Longpass IR Filter, Edmund Optics IR Cut-Off Filter, Costs: ~$40

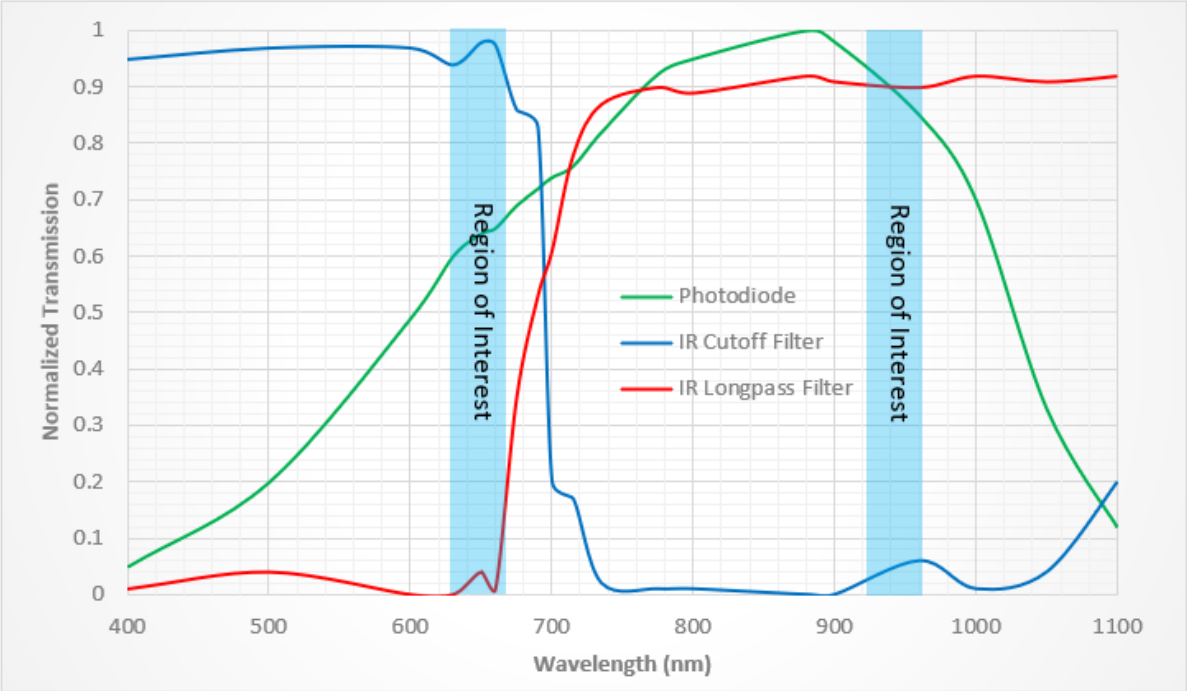


Figure 24: Photodiode and cut-off filters spectral sensitivity.

The LEDs and photodiode were chosen with the constraints of wavelength range, size, and cost. Multiples of these will be ordered in the case that some may fall victim to the stress of experimentation. The filters utilized for Design B will be ordered with larger sizes cut down to the size of the photodiode. Figure 24 details the spectral ranges of the filters and the photodiode.

Circuit Design A: Singular Photodiode

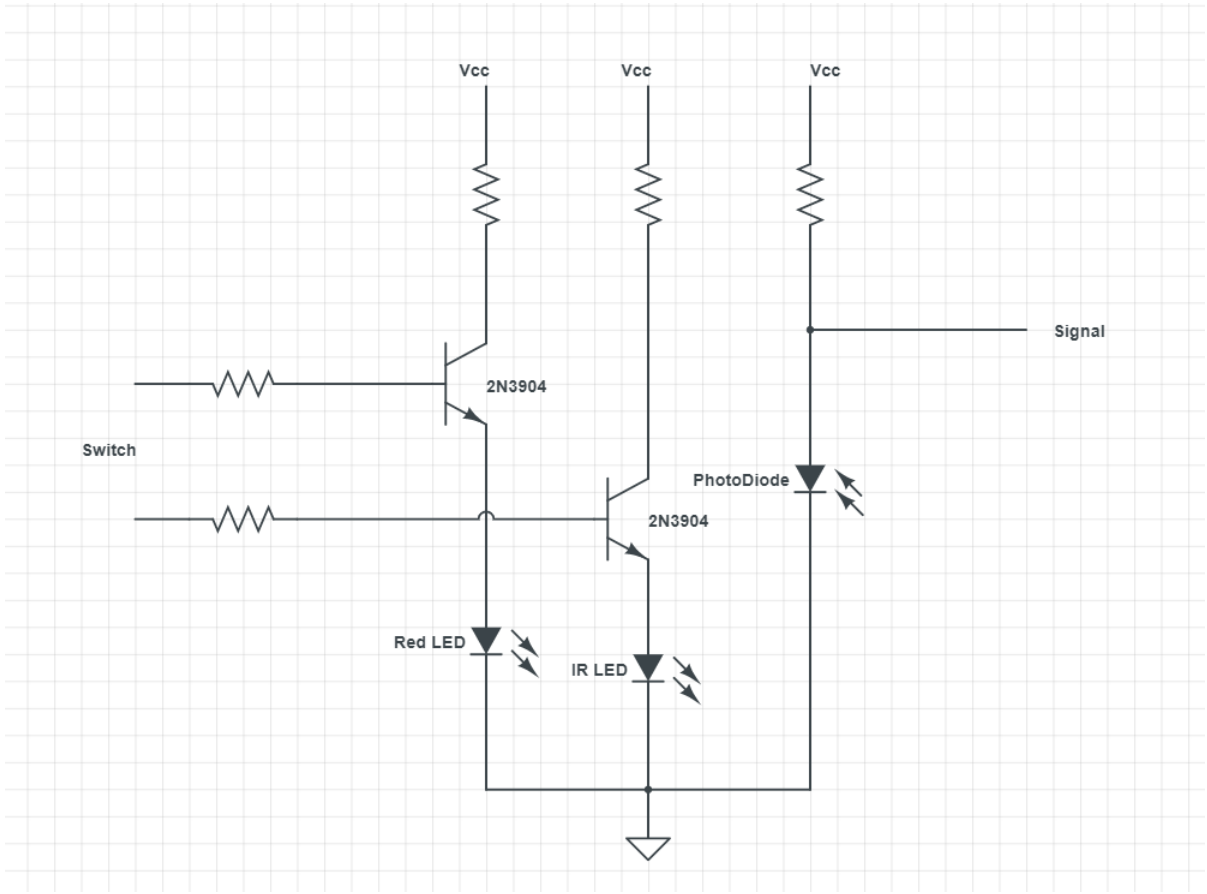


Figure 25: Circuit Design A for the Optical Heart Rate Monitor

This circuit would employ the Osram LPT 80A Photodiode sensitive to both required wavelengths. Due to the nature of having one photodiode, we will not be able to measure both the IR and the Red LED at the same time. The photodiode would not be able to distinguish the power of each distinctive wavelength of light. Therefore, periodic switching between the two LEDs will be required. The IR LED will remain on for the duration of the user wearing the device to continually measure the heart rate of the user. Once a minute, the IR LED will be turned off and the red LED turned on. Both signals will be recorded and analyzed to calculate the oxygen saturation of the user.

Circuit Design B: Double Photodiode

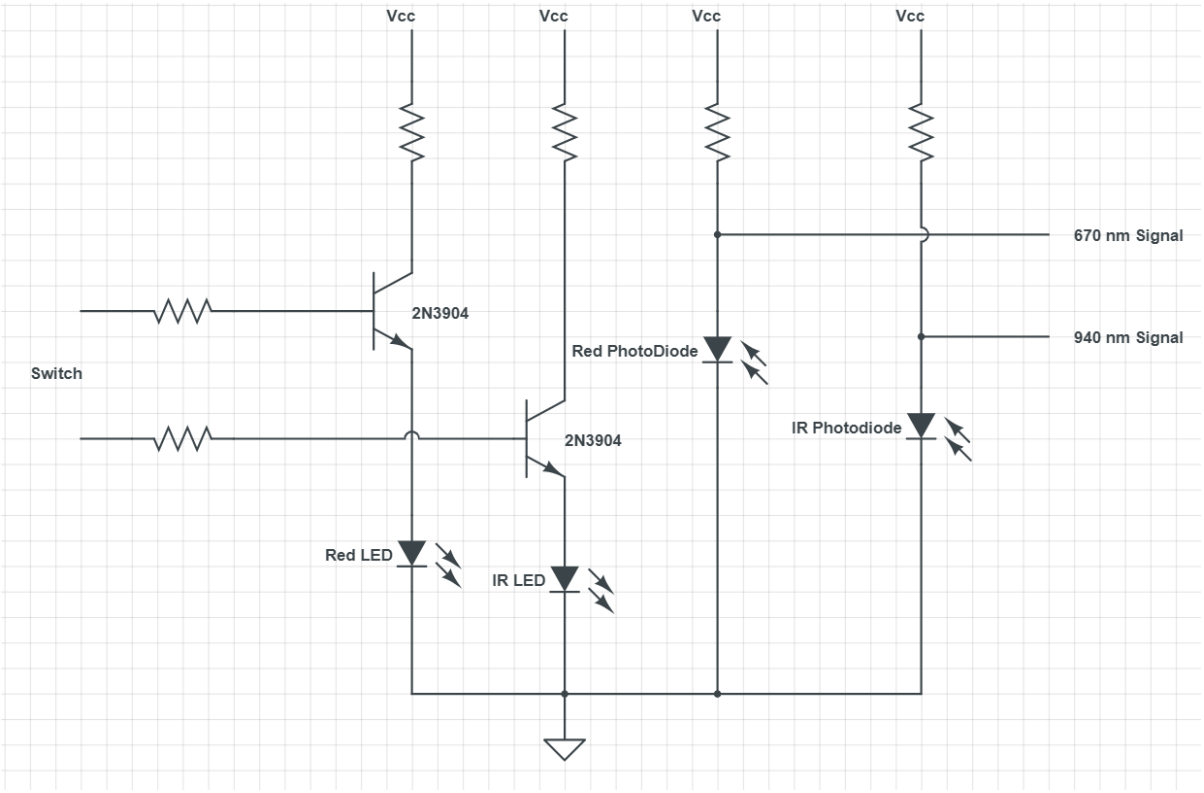


Figure 26: Circuit Design B for the Optical Heart Rate Monitor. This design adds a second photodiode to Design A with two separate signal outputs.

In the case that the single photodiode measurements are faulty, the circuit in Figure 26 will be built, employing two separate photodiodes. These photodiodes will have attached a filter, one IR longpass and one IR cut-off, for the 970 nm and 640 nm wavelengths respectively. This design will allow us to measure the reflected wavelengths simultaneously. Therefore, the data will reflect the heart rate and the oxygen saturation of the user simultaneously.

The size of the pulse oximetry circuit might be too bulky to wear on the wrist. In that case, the LEDs and photodiodes will be connected to the main circuit with header pins and jumper wires. The main circuit would reside further up the arm or the main telemetry unit on the spacesuit on the astronaut's chest.

Circuit Design C: Signal Processing

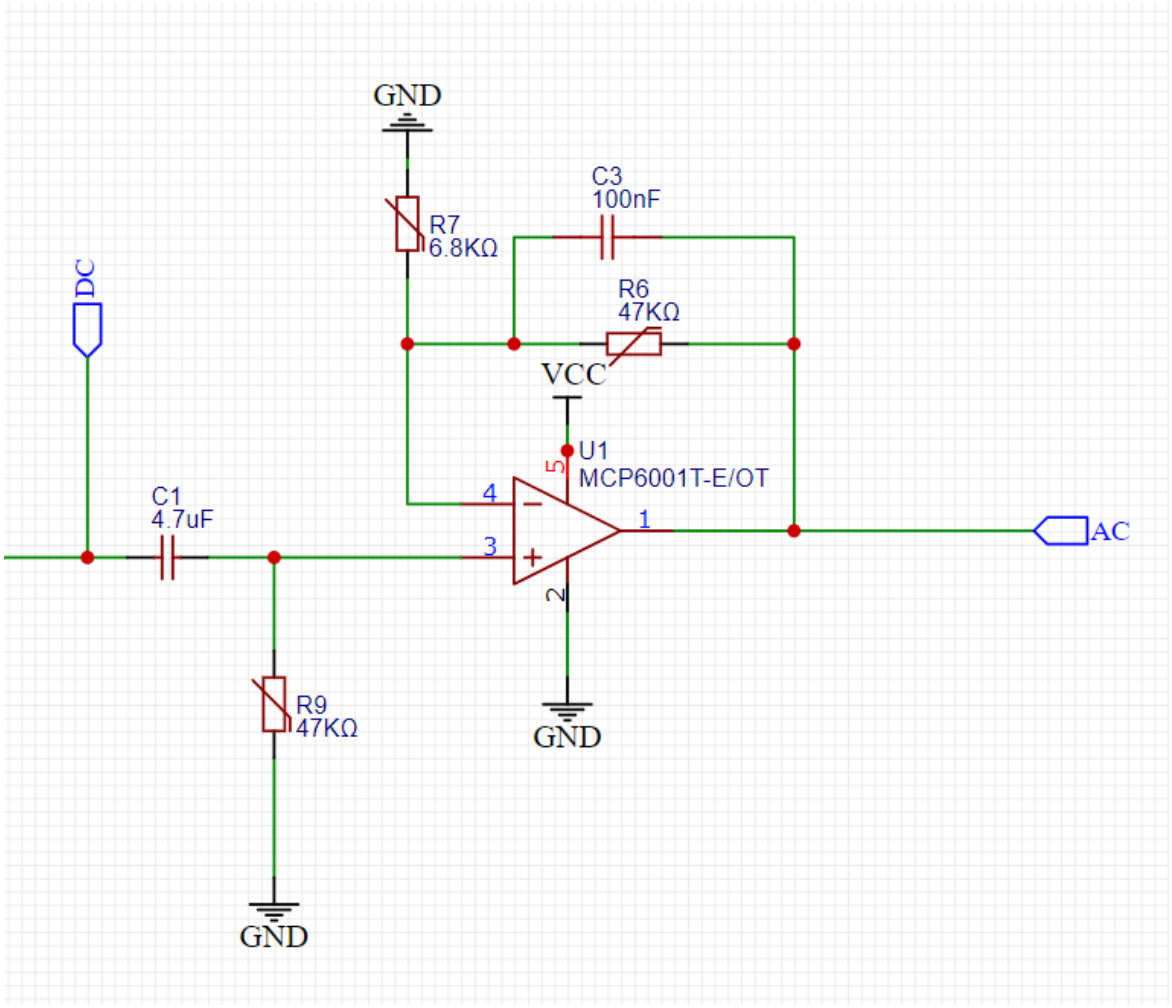


Figure 27: Signal Processing including a high-pass filter, low-pass filter, and a non-inverting amplifier.

We take two outputs out of the pulse oximeter: the non-processed signal out of the photodiode denoted in Figure 27 as DC, and a cleaned and amplified signal denoted as AC. The signal from the photodiode is processed through a passive high-pass filter with a cutoff frequency of 0.7 Hz, and through an active low-pass filter/non-inverting amplifier with a cutoff frequency of 33.8 Hz and a gain of ~8. These elements clean and amplify the signal, which is sent to be analyzed by the MCU. The AC signal will be utilized to determine the heart rate, while the DC signal will be utilized for oxygen saturation measurements.

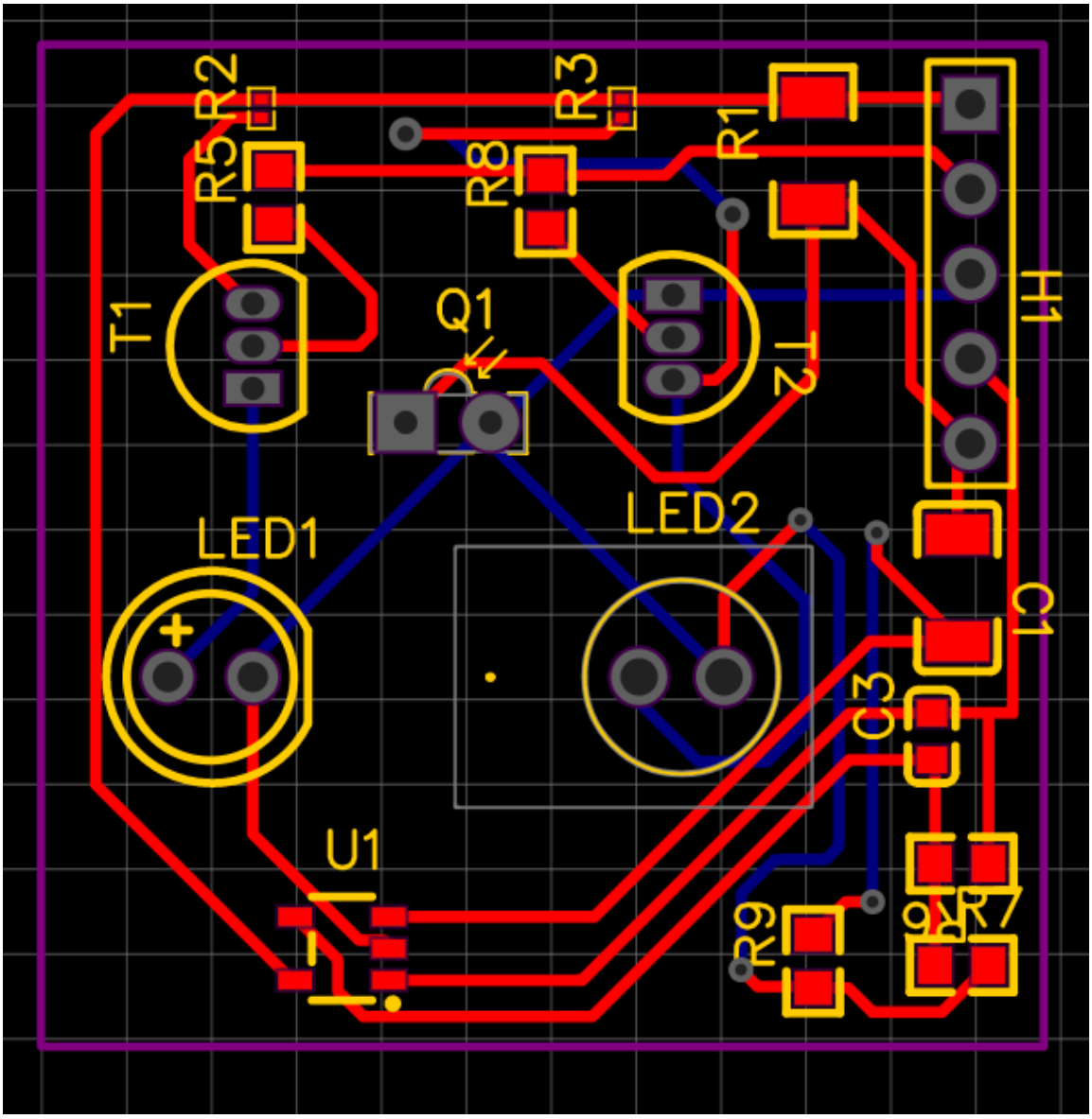


Figure 28: PCB design of pulse oximeter.

Figure 28 details the PCB design of the oximeter. The placement and orientation of the LEDs with relation to the photodiode is paramount in the acquisition in the signal. Due to the nature of the pulse oximeter, that distance is dependent on the person, how they will be wearing the wrist-mounted monitor, and any actions they will be performing such as standing still, walking, running, etc. Figure ZZ shows the 3-D view of our PCB.

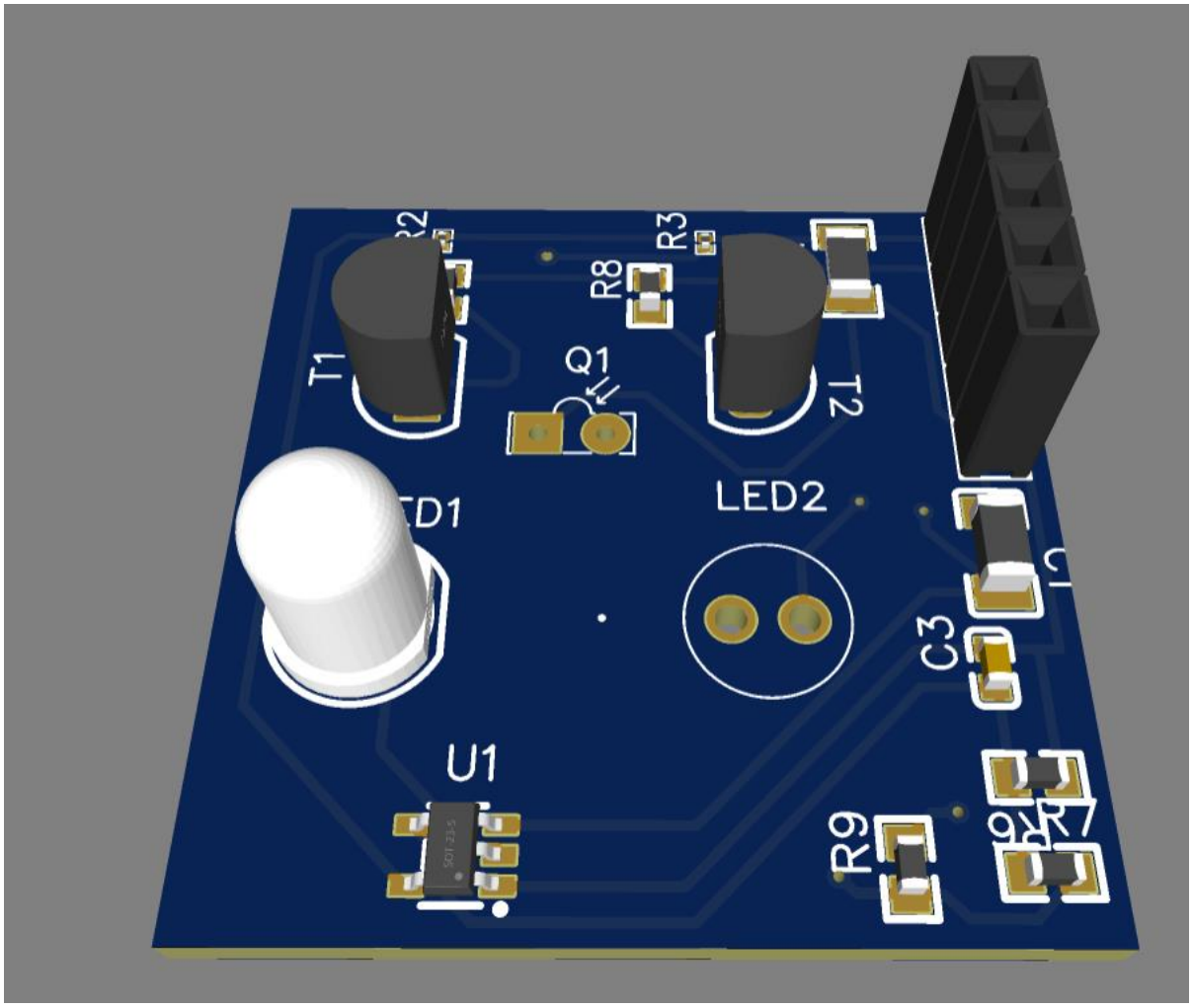


Figure 29: 3-D view of the pulse oximeter PCB. Only one LED (IR) is shown, but the red LED and the photodiode will be soldered by us.

4.2 Microcontroller Considerations

The microcontroller is the core part that controls whole system integration with modules, sensors within. In this project, several microcontrollers are considered to connect to PCB and integrate with modules and components.

4.2.1 MSP430FR6989

MSP430FR6989 microcontroller is an ultra-low-power FRAM platform that combines embedded FRAM technique and ultra-low-power system architecture. The development kit is easy to use including eZ-FET on-board Emulator, EnergyTrace Technology, 40-pin Booster Pack Plug-in Module Connector, MSP430FR6989 Microcontroller, Segmented LCD Display and two switch buttons.

Device features: (62)

- • 1.8-V to 3.6-V operation
- • 16-bit RISC architecture up to 16-MHz system clock and 8-MHz FRAM access
- • 128KB of nonvolatile FRAM
- • 100 μA/MHz active mode and 350 nA standby with RTC and 3.7-pF crystal
- • Certified ULPBench score of 109
- • 320-segment LCD controller
- • Extended Scan Interface
- • 16-channel 12-bit ADC
- • Comparator
- • Five Timers
- • Direct memory access
- • 256-bit AES
- • 83 GPIOs

4.2.2 Raspberry Pi Compute Module 4

Raspberry pi Compute module 4 is a small single-board computer. It is in a compact form factor for deeply embedded applications. It is famous for low cost and high portability. Bringing it to a smaller form factor suitable for integration into products. For generation 4, it includes a high-performance 64-bit Broadcom quad-core processor, wireless module, variant memory selection, and very good scalability.

Device features: (63)

- Portable dimension: 55 mm x 40 mm
- Broadcom BCM2711 quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- 1GB, 2GB, 4GB or 8GB LPDDR4 (depending on variant)
- Optional wireless LAN, 2.4GHz and 5.0GHz IEEE 802.11b/g/n/ac wireless, Bluetooth 5.0, BLE with onboard and external antenna options
- 1 × USB 2.0 interface
- PCIe Gen 2 x1 interface
- 28 GPIO signals
- Works from -20°C to +85°C

4.2.3 Atmega328p on Arduino

Arduino platform is an open-source, well-developed, and widely-used in many small projects, on the Arduino UNO, there is an Atmega328P-PU microcontroller, so we choose it as our microcontroller, it's a low-power CMOS 8-bit AVR microcontroller widely used in many projects. It has many digital communication peripherals: 1-UART, 2-SPI, 1-I2C, and several Capture/Compare/PWM Peripherals: 1 Input Capture, 1 CCP, 6 PWM. What's more , it's the same MCU used on Arduino UNO, after a process named "Burn the Bootloader" , we can use Arduino IDE to program with it using C/C++, then upload the program intro the chip and use it like Arduino. Since there are a lot of projects based on Arduino, so we choose our sensors based on existing projects, it can help us overcome challenges.

Device Features: (64)

- ATmega328P High Performance, Low Power AVR 8-bit Microcontroller
- 14 digital I/O pins (6 PWM output)
- 6 Analog Input pins
- 32 KB Flash Memory
- 2 KB SRAM
- 1 KB EEPROM
- 16 MHz Clock Speed
- 5 V Operating Voltage
- 7-12 V recommended Input Voltage (20V Max)

4.3 Power Supply

To maintain the microcontroller and all modules works perfectly, it's necessary to have a power system to output stable power to the system. Normally, there is a battery in a space suit, but it's not an available part we can get, so there will be a battery or power system to supply power. First of all, it's necessary to know the power requirements of each component. (for convince, microcontroller is considered as Raspberry Pi since it's needed the most power in the microcontroller considerations)

Component	Voltage	Current	Power
Microcontroller	5.1V	3A	15.3W
IR Camera	5V	1A	5W
Temperature Sensor	5V	/	Very Low
Optical Heart Rate Monitor	3.3V	/	Very Low
Bluetooth Module	3.6V	/	Very Low

Table 3: System Specific Power Requirements

Aftering searching power consumption for all components and modules needed for this project, simply adding up all powers needed, to be on the safe side, a battery system that can supply 25W power is necessary.
Normally, a power supply with 25W stable output is wall power adapter (AC Source), which is unrealistic to use on a space suit, and based on the known information, there is a battery on space suit, to replace it on earth or in this project, a fast charge usb-c power bank (DC Source) or a Li-ion Battery (DC Source) that can supply 25W power is more feasible.

4.4 Voltage Regulator

To make the power supply outputs a stable voltage and current, a Voltage Regulator is needed. Based on research, there are two types of voltage regulator, Line Voltage Regulator and adjustable switching DC to DC converter. For most projects, Linear Voltage Regulator is the first choice, but after researching, in some projects with high power requirement,

Linear Voltage Regulator:
For this project, the voltage input should be converted to 5V, then a standard 3-pin LM340 from TI is a good choice to power up the device.

Regulator Features:

- Output Current up to 1.5 A
- Available in Fixed 5-V, 12-V, and 15-V Options
- Output Voltage Tolerances of ±2% at TJ = 25°C (LM340A)
- 125 °C Maximum Operating Temperature
- Internal Thermal Overload, Short-Circuit and SOA Protection
- Output Capacitance Not Required for Stability

DC to DC Converter:
While the linear voltage regulator facing problems such as overheating, not working, then a DC to DC Converter is in consideration, after searching, TPS542A50 DC/DC Switching Regulator is a good choice.

Regulator Features:

- 4-18V wide input voltage range
- 0.5-5.5V wide output voltage range
- Integrated 9.1-mΩ and 2.6-mΩ MOSFETs support up to 15-A output current
- Switching frequency 400-2200 kHz
- –40 to +150°C operating junction temperature

4.5 Wireless Communication

Serial communication streams between all sensors, microprocessing units, and the HoloLens will require circuits which transmit data wireless across multiple different frequency channels at the same time. To accomplish this, multiple potential ICs and circuits have been researched which transmit and receive RF data.

433MHz RF wireless transmitter and receiver

The Generic 433MHz wireless RF module is a low cost, low power transmitter/receiver pair ideal for transmission of serial data.

Device specifications:

- Transmitter:
    - Transmission frequency: 433.05MHz - 434.79MHz
    - Operating voltage: 3.5-12VDC
    - Transfer rate: 10Kb/s (or less)
    - Pinout: Data (serial), VCC, GND.
    - Distance of signal transmission: 20-200 meters
    - Size: 19 x 18.9 mm
- Receiver:
    - Receiving frequency: 433.05MHz - 434.79MHz
    - Operating voltage: 5VDC
    - Receiving Sensitivity: -105dB
    - Distance of signal transmission: 20-200 meters
    - Size: 30 x 13.4 x 7mm

XBee ZigBee wireless communication module

The XBee ZigBee wireless communication module which transmits data further than Bluetooth but at a lower rate. It follows the IEEE 802.15.4 standard for RF transmission at 2.4GHz.

Device specifications:

- RF transceiver
- Low power consumption.
- Operating voltages: 1.8-3.6V
- Operating frequency: 2.4-2.483GHz
- Data rate 1.2 - 500 Kb/s

HC-05 Bluetooth module

The HC-05 Bluetooth module transmits data faster than ZigBee but at a much lower range. It follows the IEEE 802.15.1 standard for RF transmission at 2.4GHz.

Device Specifications:

- Support for multiple communication protocols: UART, USB, SPI
- BC417 Bluetooth IC
- Flash memory
- 3.3V regulator
- Operating Voltage: 4-6V
- RF Transformer and Antenna Trace
- 6 pins: VCC, GND, TX, RX, EN, STATE.
- 4 pins typically used for communication

nRF24L01 Transceiver Module

This wireless RF module operates at 2.4GHz. Both the transmitter and receiver are integrated into a single IC (compared to two with the Generic 433MHz module).

Device Specifications:

- RF transmitter/receiver IC
- Transmission frequency: 2.4GHz
- Data rates: 250kb/s, 1Mb/s, 2Mb/s
- Channel bandwidth of 1MHz
- Low power consumption
- SPI serial communication protocol
- Operating voltage: 1.9-3.6V

4.6 PCB Design

As designed, the PCB part will be divided into two parts, one includes our microcontroller, the other one includes all other wireless communication modules. For this project, Easyeda is used to design all circuit diagrams and PCB layouts, the reason is this software is free and more user friendly than some other softwares.

4.6.1 Wireless Communication PCB

For the second PCB with all wireless communication modules, the first step is to decide the specific component we need. For Generic 433MHz wireless RF module, we choose FS1000A transmitter, there is a switching transistor and a few passive components on it, when a logic HIGH applied to the data input, the oscillator runs and will produce a stable RF output wave at 433MHz, when a logic LOW applied to the data input, the oscillator will stops.

Port	Connection Port
DATA	D12
VCC	+5V
GND	GND

Table 4: FS1000A Transmitter Port Connect Instruction

For bluetooth modules, we choose HC-05 Bluetooth module, it's a multi-functional module that can provide two-way wireless functionality, we can use this module to communicate between any two devices with bluetooth functionality like an Arduino board or computer, the connection instruction is in figure below. For the Transceiver module, we choose nRF24L01 to be used, it uses 2.4 GHz band and operates with baud rate between 250 kbps and 2 Mbps, and it's range can reach up to 100 meters.

Port	Connection Port
BT-RX	D1
BT-TX	D0
GND	GND
VCC	+5V

Table 5: HC-05 Bluetooth Module Port Connect Instruction

For the Transceiver module, we choose nRF24L01 to be used, it uses 2.4 GHz band and operates with baud rate between 250 kbps and 2 Mbps, and it's range can reach up to 100 meters.

Port	Connection Port
GND	GND
CE	D9
SCK	D13
MISO	D12
VCC	+3.3V
CSN	D8
MOSI	D11

Table 6: nRF24L01 Transmitter Port Connect Instruction

For the wireless communication module, we choose Xbee Zigbee wireless communication module, it uses IEEE 802.15.4 standard which is for low power applications of radio frequency. In this project, Zigbee will be used as a transmitter and receiver by using serial communication to send and receive data.

Port	Connection Port
PWM0RSSI	D1
ASCADSDIOS	D0

Table 7: Xbee Zigbee Wireless Communication Module Port Connect Instruction

For the temperature sensor, we choose DS18S20 Digital Temperature sensor, it provides high accuracy (±0.5 °C) with fairly low cost, the operating voltage of the sensor is 3.0V to 5.0V, which works for the Arduino. In this project, we have multiple sensors to connected to Arduino, so this sensor has some advantages, it only requires one digital pin of the Arduino for communication which saves pin for the project, and has a unique 64-bit serial code, which can be connected to one Arduino pin in 1-Wire with other sensors and has no error code.

Port	Connection Port
GND	GND
DQ	D2
VCC	+5V

Table 8: DS18S20 Digital Temperature Sensor Port Connect Instruction

After searching all connect instructions for each module, it is necessary to draw a circuit diagram before making it as PCB. Since this PCB has several modules, and several of them use the same pin, it is better to list all pins needed, and use two connectors to connect all pins with all modules, for easier to read and connect.

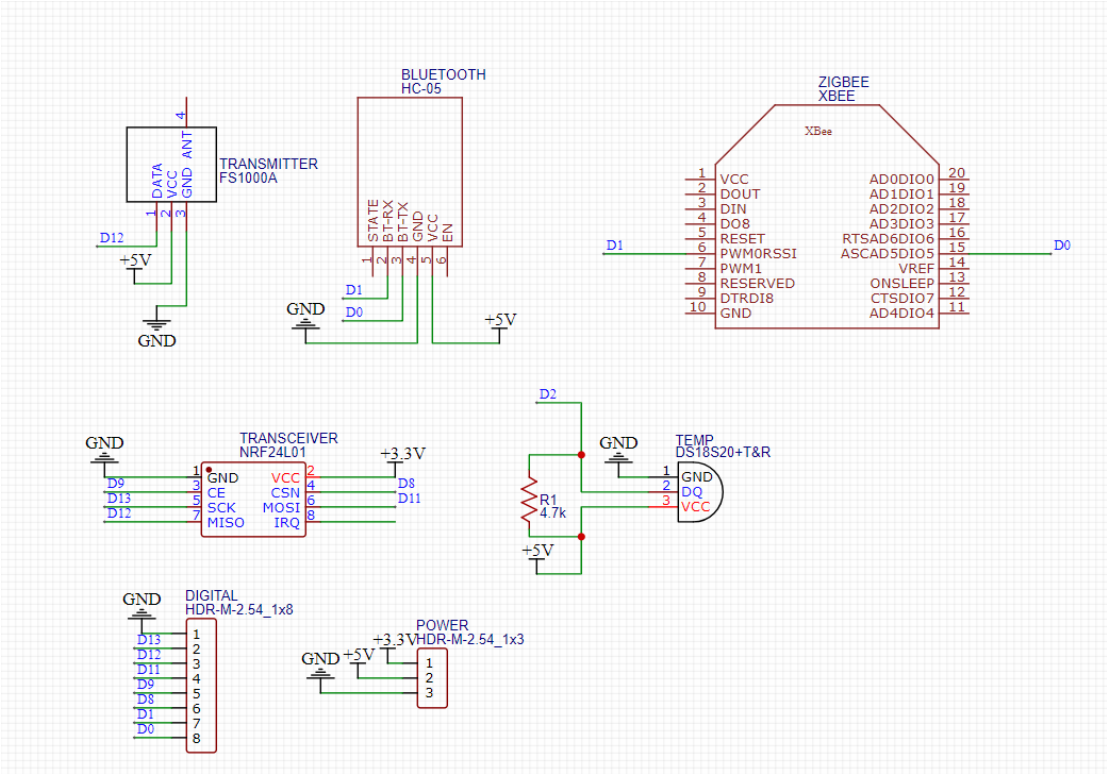


Figure 30: Wireless Communication PCB Circuit Design

While the circuit diagram is done, follow the guide routes to connect all pins, red wire is on the top layer, and blue wire is on the bottom layer. For this board, we get 2820 mil length and 2360 mil width, with all wires having 10 mil width. The following two figures illustrate the routes design and desired printed look for this board.

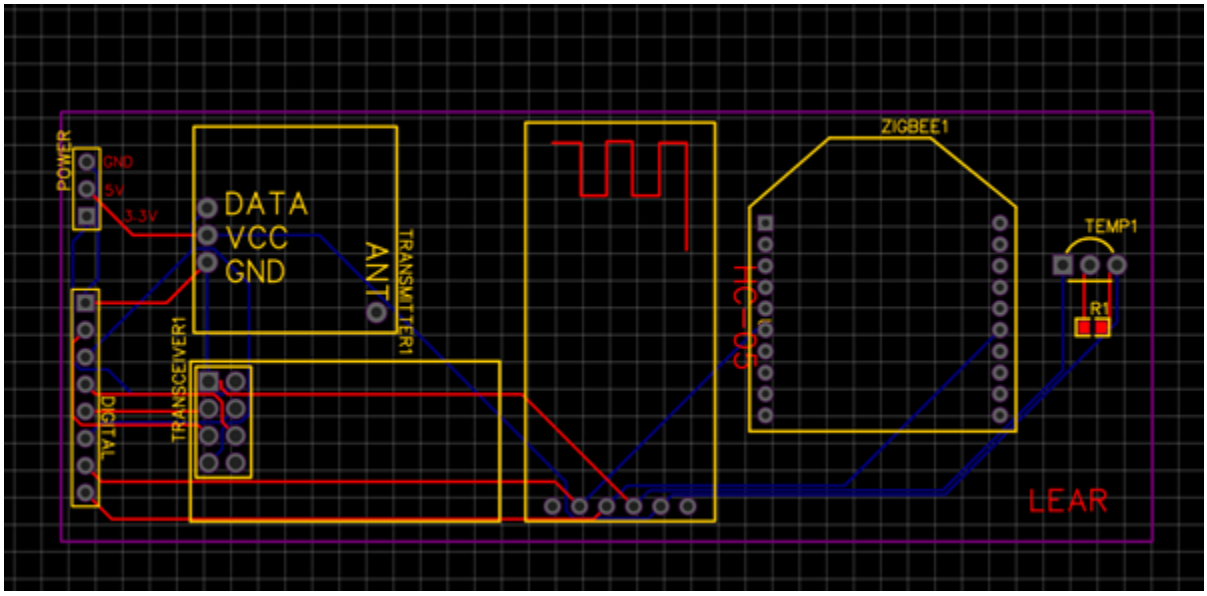


Figure 31: PCB design of Wireless Communication Module

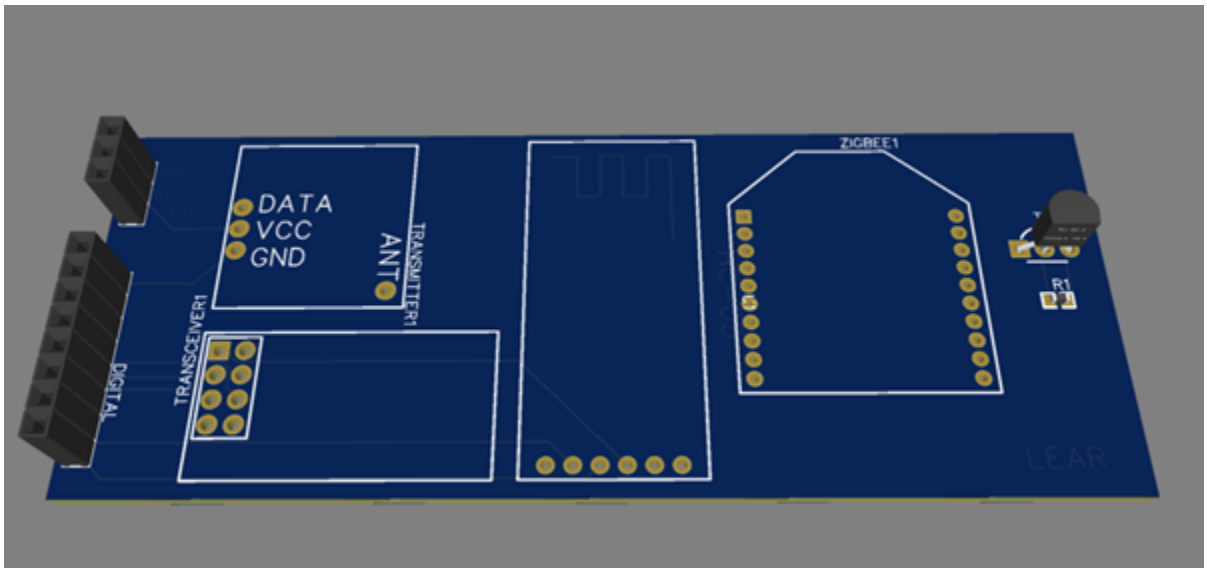


Figure 32: 3D View of Wireless Communication Module

4.6.2 Microcontroller PCB

In order to control the hardware system and connect to HoloLens, a microcontroller PCB is designed. On this PCB, an Atmega328p-pu is the heart, with 2 voltage regulators circuits built-in, LM7805 and LM317 for 5V and 3.3V respectively. In case to connect all pins and for future improvements, there are 32 pins in total with 6 analog pins built-in, so it can connect almost everything would use in a senior design project, for example, originally, the Bluetooth module should be connected to TX and RX pins, but it will be conflict to the oximeter, which is connected to analog pins, but there is another pins can be used, so we just change pins to transfer data. Also, an Atmega328p-pu processor is working on C/C++ environment, and its design for Arduino, to make the project easier, it's better to use Arduino IDE directly, so it's required to burn the bootloader on a brand new Atmega328p-pu before solder onto the PCB. After that, we can connect our microcontroller PCB with a USB to TTL converter with 5 pins: 3.3V, 5V, RX, TX, GND, then this module can use the Arduino IDE and all its libraries directly, and upload program into the microchip.

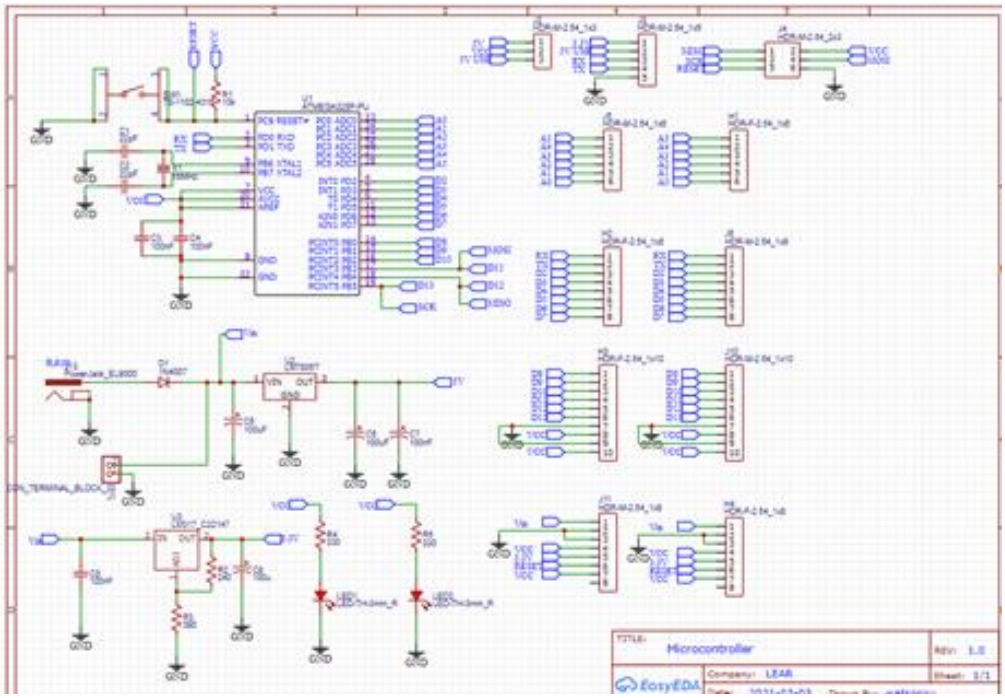


Figure 33: Microcontroller PCB Schematic

While the schematic is done, convert it to the PCB view in software, and follow the wires it listed, connect all of them, place components in a solder convenience order.

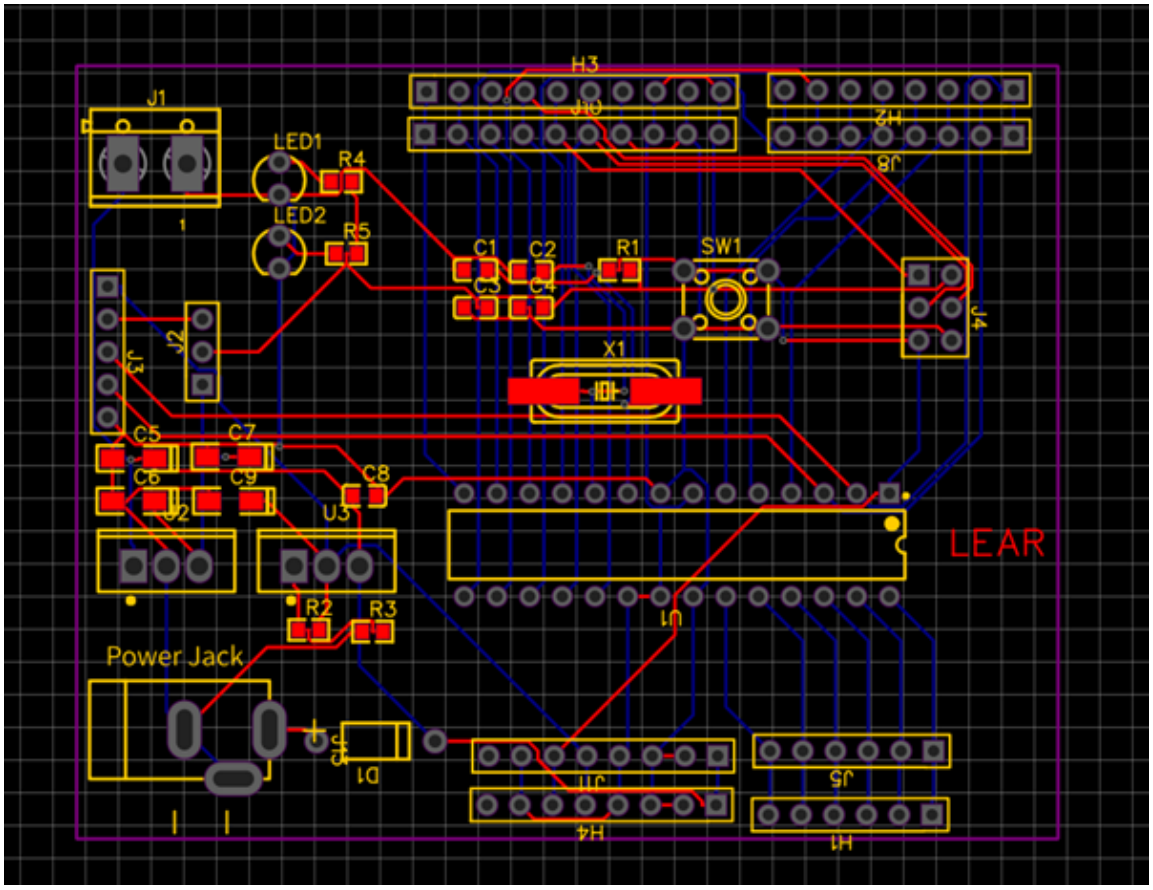


Figure 34: Microcontroller PCB 2D View

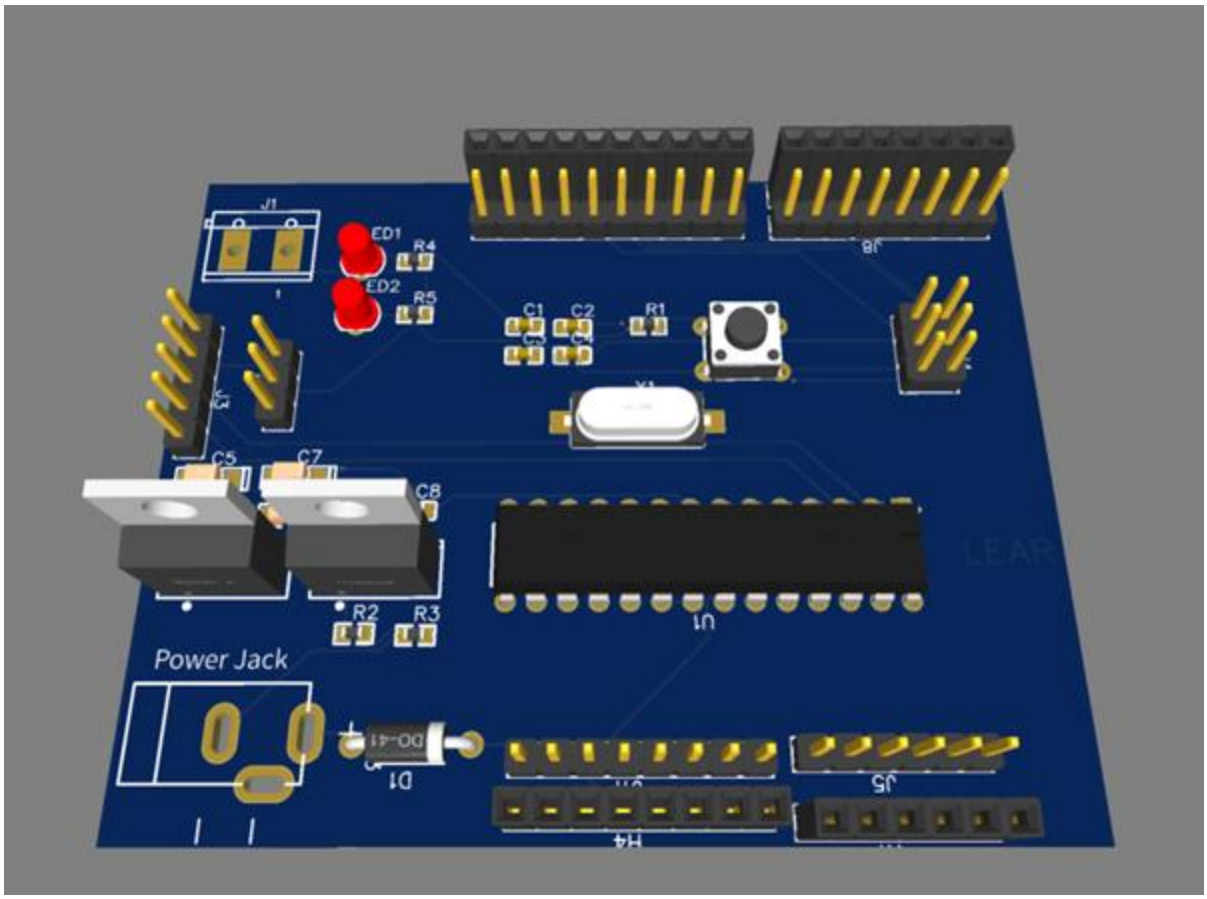


Figure 35: Microcontroller PCB 3D View

5. User Interface and Software Development

5.1 Use Cases

Given the large number of requirement specifications for completion of this project, and a large portion of those specifications being software, a basic set of use cases were derived to guide the design of our User Interface based on NASA's SUIT challenge main objectives. This allowed us to identify the main actors in the design of our system and the functionality the software must fulfill. We define the primary actors (left side of Figure 36) as the astronauts and the secondary actors (right side of Figure 36) as the HoloLens II, which responds to the astronauts explicit interactions with the UI or implicit interactions (through passive and continuous measurement of astronaut vitals using external sensor data).

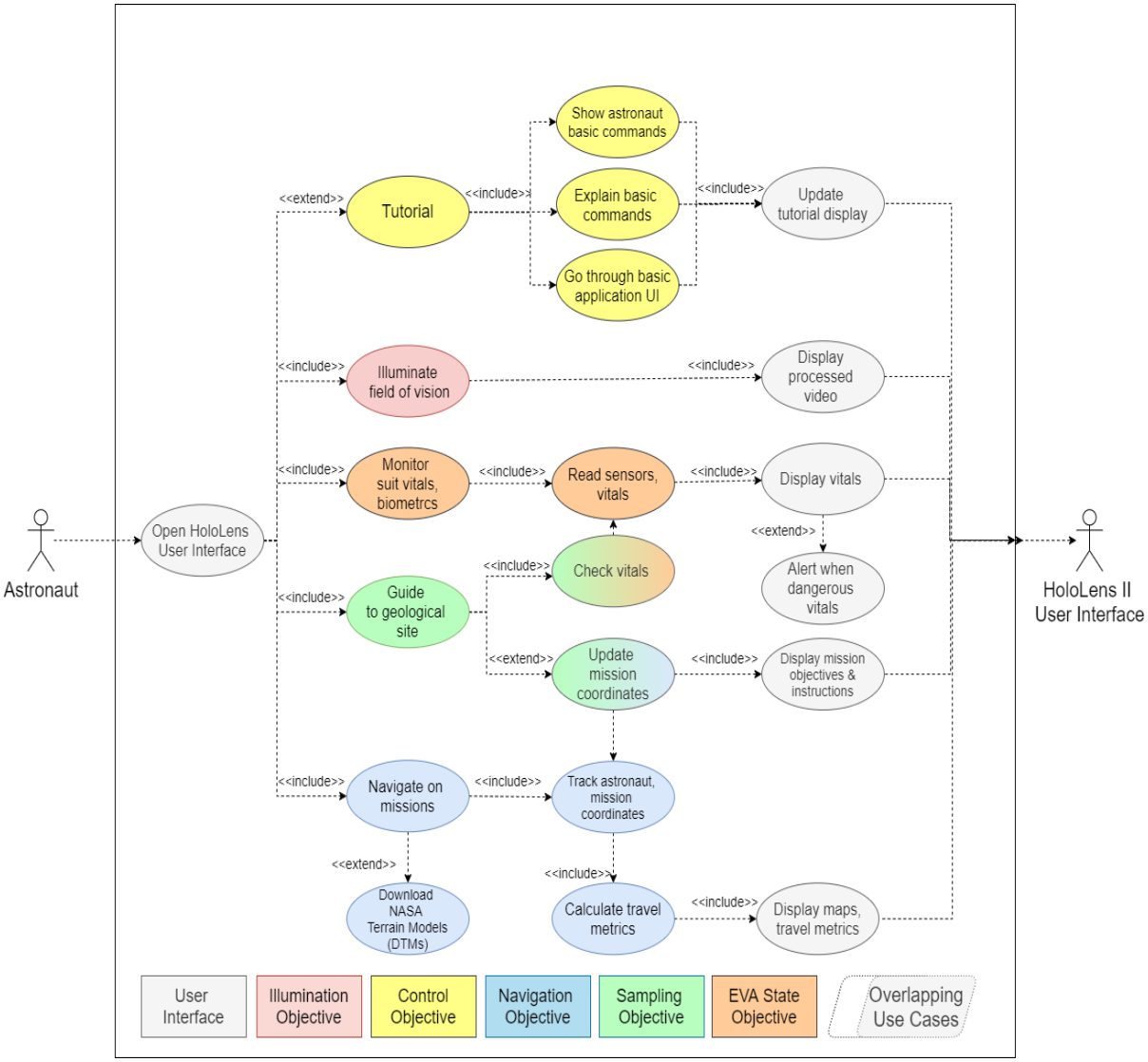


Figure 36: Use Case Diagram

NASA's challenge outlines five main objectives; 1) Control methods, 2) EVA System State, 3) Science Sampling, 4) Navigation, and 5) Illumination. Each of these is pictured as a separate use case in Figure 36. The 'Control methods' objective is to give the astronaut efficient ability to execute EVA tasks. They emphasize an importance in training the astronaut how to control the HoloLens II User Interface efficiently using a range of possible commands (gestures, voice, text, etc etc), so the most important use case for achieving this particular objective is through a tutorial. The tutorial is only executed if the astronaut is a new user or makes the choice to manually engage the tutorial. See Figure 36 (yellow) for specific details.

For the 'EVA System State' objective, the overarching goal is to monitor both astronaut and suit vitals (heart rate, suit oxygen levels, etc) in real-time. This is done passively using external sensor data, which is processed and displayed on the HoloLens UI. Under the condition that either suit or astronaut vitals approach a dangerous level, the HoloLens UI will alert the astronaut with a warning (Figure 36, orange).

The goal of the 'Illumination' objective is to illuminate the lunar terrain so the astronaut can safely navigate when conducting EVA tasks (Figure 36, red). When the astronaut enables this option, the video recorded from the HoloLens field of vision will be processed, contrasted in such a way to allow the astronaut to more clearly see terrain variance, and then projected onto the astronaut's field of vision.

The 'Navigation' objective helps the astronaut navigate the lunar terrain. This use case requires downloading an updated "NASA Digital Terrain Model (DTM)" to effectively map the lunar search (Figure 36, blue). The software tracks the astronauts coordinates relative to their starting position (lunar lander) and ending position (geological/mission coordinates) allowing effective monitoring of the astronauts position and helping them navigate. A set of maps, both 2D and 3D, will be rendered, updating the astronauts position in real-time, along with the quickest path to their destination. Lastly, estimated time of arrival, estimated speed, and distance will be calculated and displayed so the astronaut can make effective decisions as the missions evolve.

Finally, the last use case (and objective) is 'Science Sampling.' The goal of this objective is to allow the astronaut to efficiently collect samples from geological sites of interest on the lunar surface. To complete this objective, the astronaut must first download a set of documents including instructions for the mission, tasks for the mission, notes about the mission, mission coordinates, and set of tools necessary for collecting the samples. The set of documents must then be processed into the proper format for display on the HoloLens UI. This use case shares overlap with the EVA System State and Navigation use case for successful navigation to the geological site (Figure 36, green)

5.2 Design Description

A flowchart detailing the overall structure underpinning the software of our proposed SUITS User Interface (UI) can be located in Figure 37. There are six total applications, each responsible for fulfilling a primary objective (or use case; see Figure 36) of the SUITS Design Challenge. Differentiation of each application is based on the color coded legend. Application interfaces between two different applications are represented by two rhombus shaped boxes containing the communicating applications. Further detail on the logical structure of the Sampling, EVA State, and Illumination Applications can be found, in detail, in following sections of this proposal.

Also pictured is the communication protocol used to send data from our external sensors, which collect and send data in real-time through a microprocessing unit. The microprocessing unit then communicates wireless using bluetooth, sending the collected data to the HoloLens. This is represented in the diagram by the bluetooth symbol and the hardware/software interface. The data is collected by the HoloLens through a bluetooth socket, processed into a decipherable format, and fed into the application to allow the external environment to control the flow of information through the application.

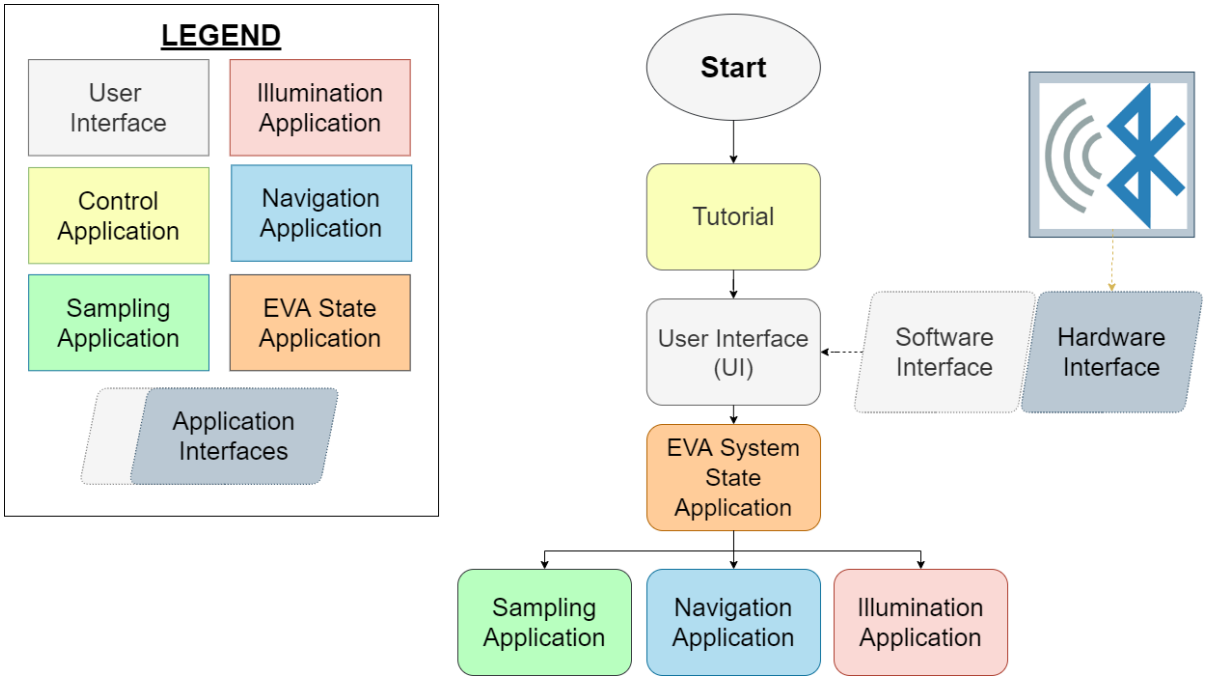


Figure 37: Software Design Diagram

5.2.2 EVA system state

The EVA System State application (Figure 37, orange blocks) runs at all times and monitors both astronaut and suit vitals (heart rate, suit oxygen levels, cooling water, etc) in real-time to maintain astronaut safety. These measurements are displayed on the UI in an unobtrusive manner to avoid restricting the astronauts field of vision or destroying astronaut concentration. If an abnormality is measured, the astronaut will be warned on the UI. If the abnormality does not correct itself or is compromising the health of the astronaut, the astronaut will be warned to abandon the mission and return to the lander.

When the astronaut marks coordinates for a mission, the application interfaces with the Sampling/Navigation applications to calculate approximate distance, travel time, and whether current suit vitals (oxygen, etc) will last the round-trip. The astronaut will be warned if the mission would be considered dangerous or taxing on resources.

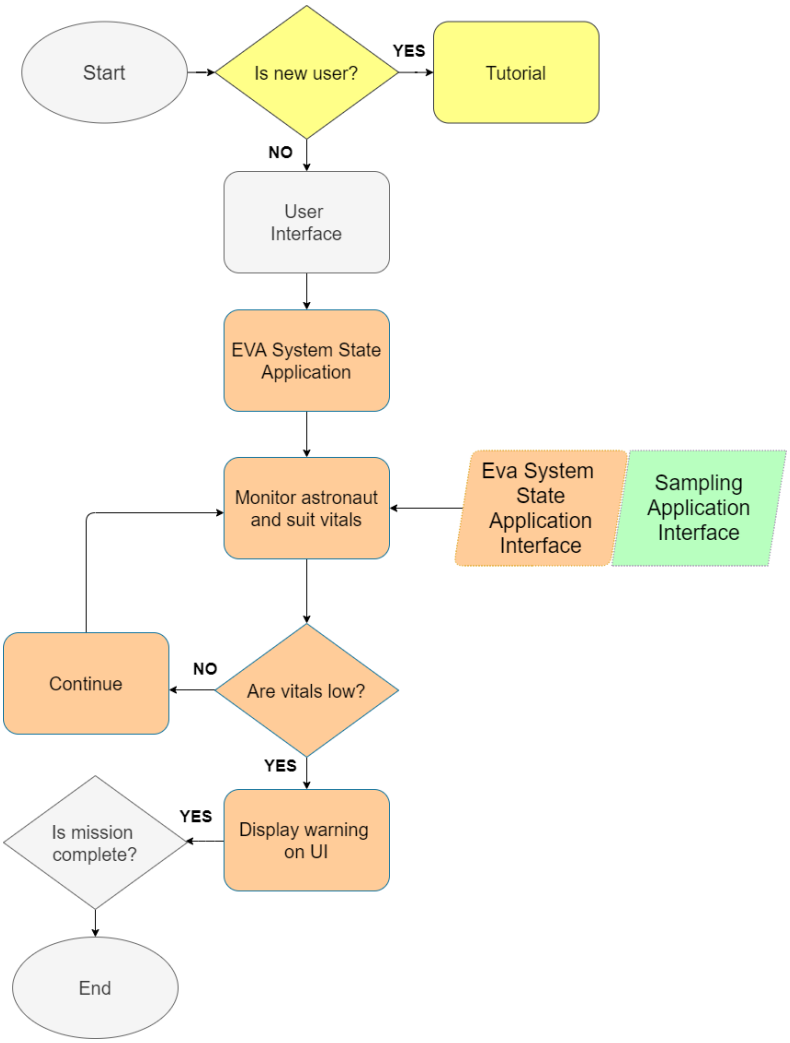


Figure 38: EVA System State Design Diagram

5.2.3 Illumination

Astronauts will have the ability to toggle on an ‘enhanced illumination mode’. Enabling the illumination mode will utilize our enhanced image processing techniques and night vision solutions to aid the astronaut in viewing low-light or high contrast environments. The enhanced image will be overlaid onto the visual field of the astronaut (Figure 39).

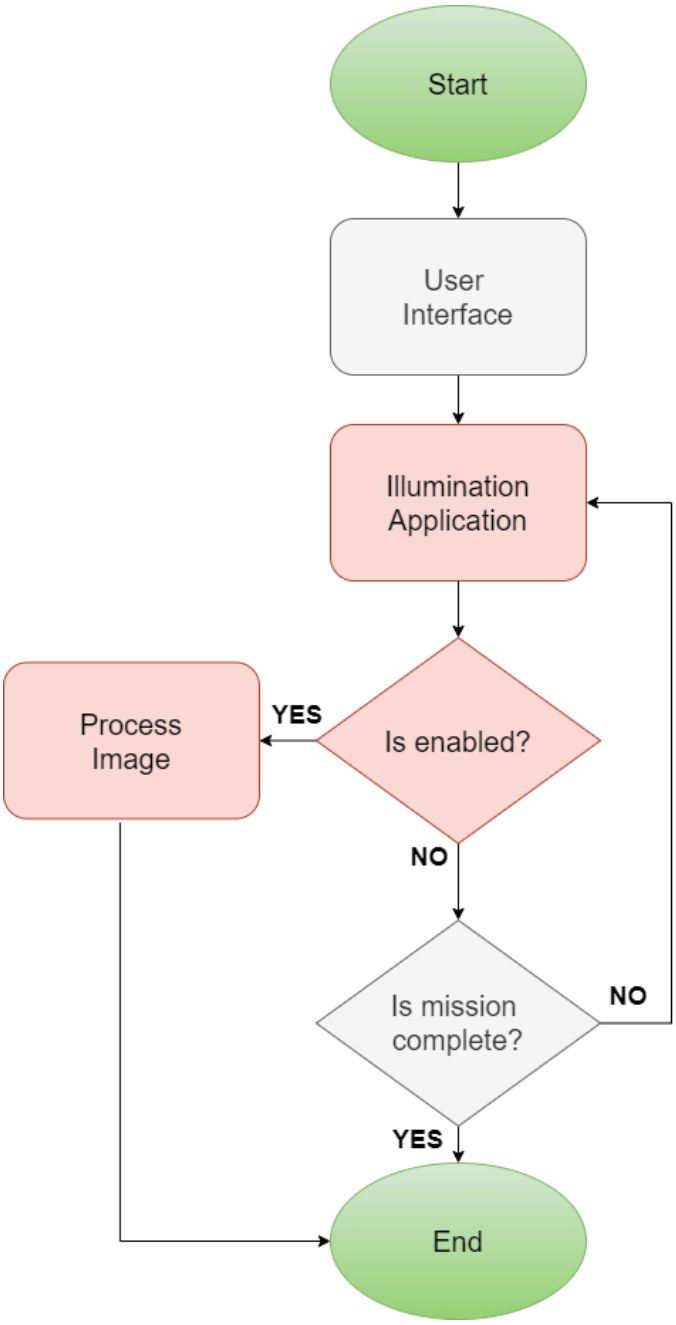


Figure 39: Illumination Design Diagram

5.2.4 Navigation

Astronauts will be guided to marked locations of interest using the Navigation application (Figure 40, blue). When in use, this application will first decipher if this is a new mission. If so, the application will download NASA Digital Terrain Models for the 2km - 10km range surrounding the coordinates (58). The distance and approximate travel time will be calculated along with a directional path from the astronaut's current location to the marked location, which will display on the UI as an overhead directional arrow for easy guidance. Lastly, the terrain models will be used to render a 2D and 3D topological map of the terrain surrounding the marked location. These maps can be referenced by the astronaut and viewed within the HoloLens 2 visual field.

When the astronaut starts their mission, a virtual "tether" will attach the astronaut to their starting coordinates, keeping track of the astronaut's directional movement through the terrain. This will act as a way for the astronaut to record discrepancies between the map renderings virtually compared to the actual terrain, a source of field notes for recording actual paths around the terrain, and an easy way for the astronaut to navigate back to the Lander. The aforementioned navigation metrics will be calculated and the UI will be updated until the astronaut completes their mission.

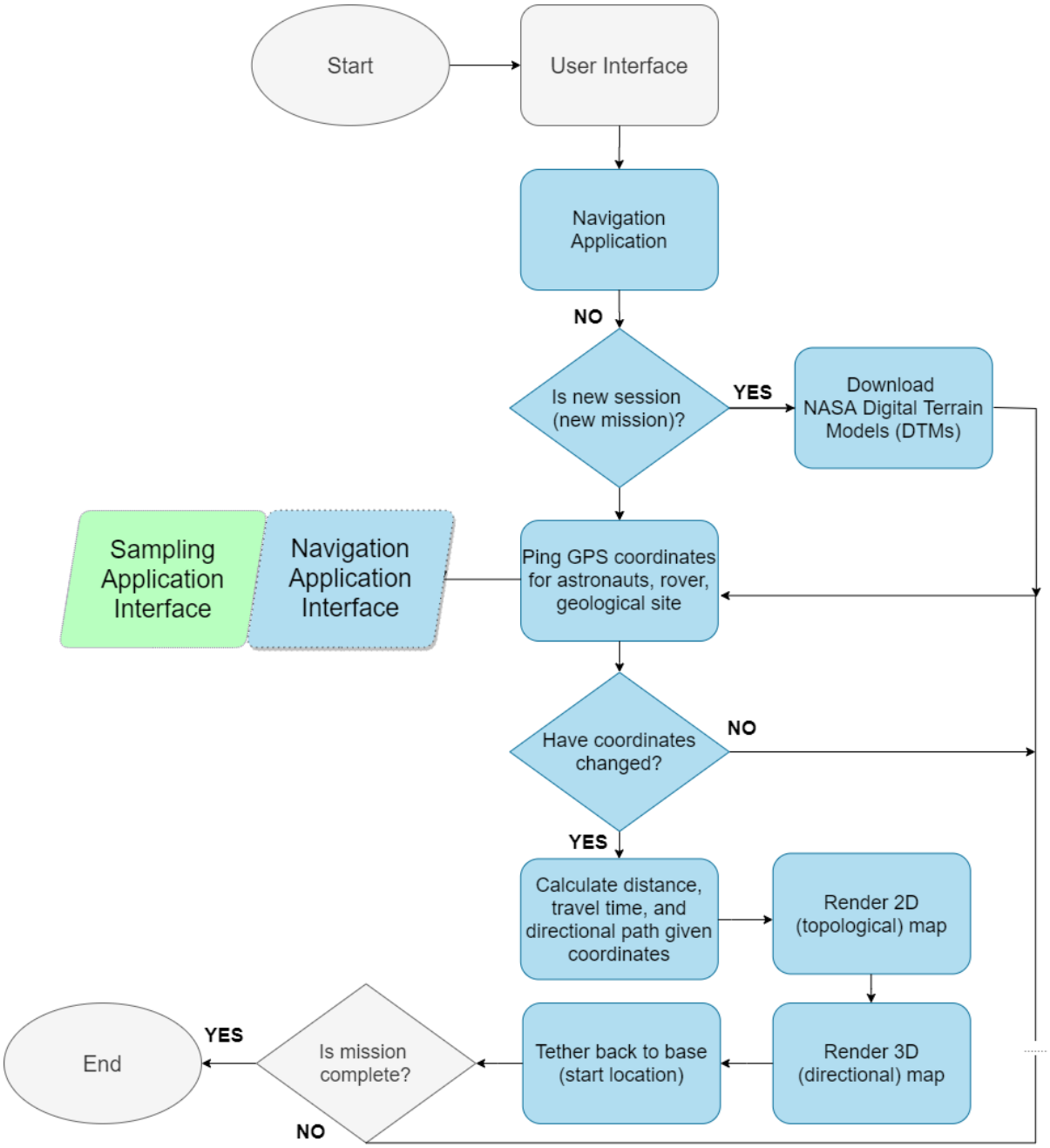


Figure 40: Navigation Design Diagram

5.2.5 Geological Sampling

Exploratory missions to geological sites of interest will be done using the Sampling application (Figure 41, green). Information pertaining to each individual site such as coordinates, tools to use, list of instructions for mission, field notes, and goals will be downloaded from mission control and listed within the UI. Once the astronaut has been briefed on the mission, they will precede to the geological site. The distance to the location, approximate time of travel, and rendered 2D/3D maps will be automatically displayed through an interface between the Sampling and Navigation applications.

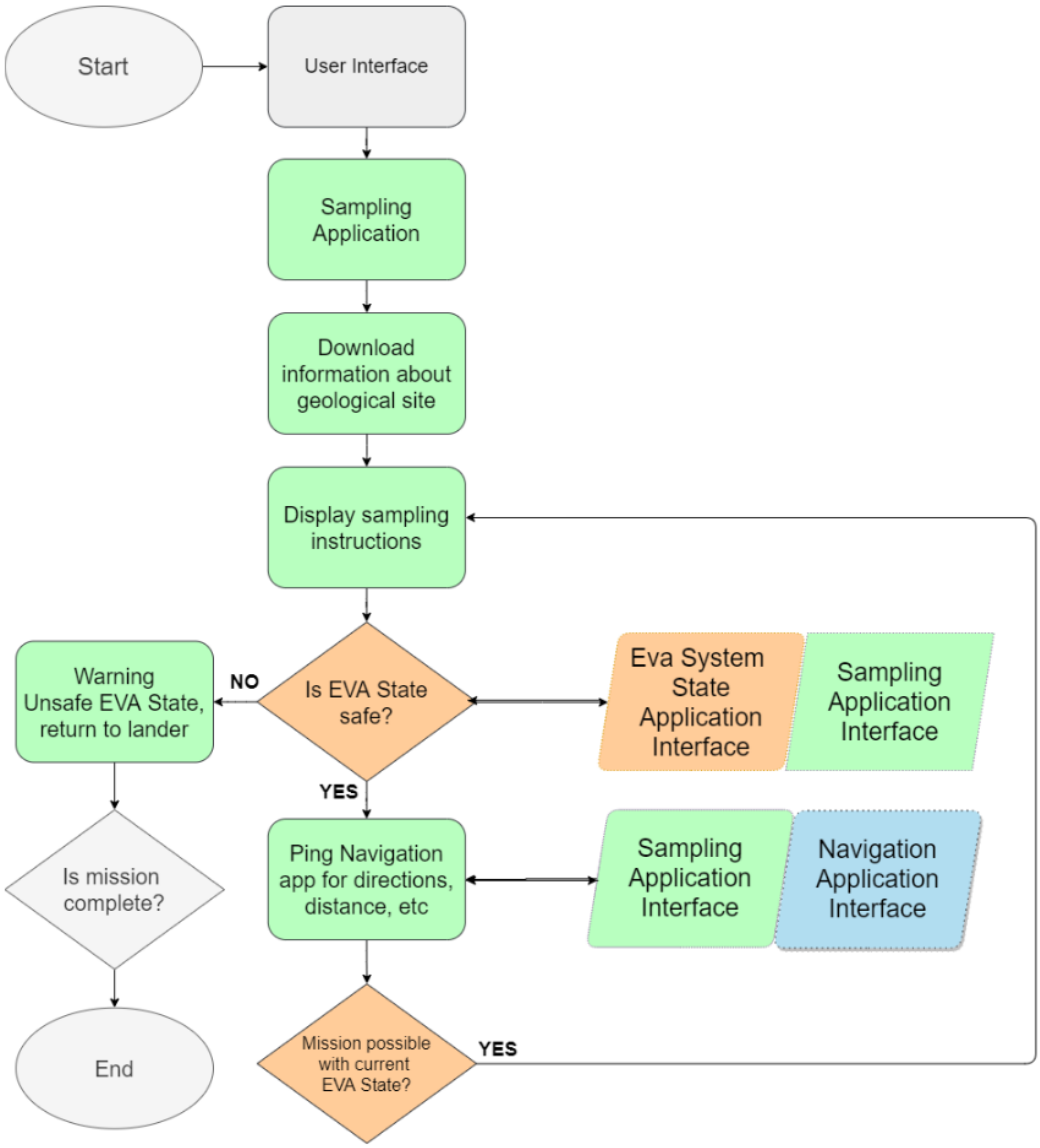


Figure 41: Sampling Design Diagram

5.2 Concept of Operations (CONOPs)

This section details the final product from the point of view of the user. The operations will use different overlays which can be swapped out using built-in voice commands or gesture controls. Each command will display different information. The overlays will each respond to distinct commands which are only available through that overlay. Some of these views will utilize built-in voice commands and gestures to navigate through the applications. A combination of commands, both gestures and voice, will be implemented with the purpose of maintaining astronaut freedom of movement (voice commands), along with mission control communication streams (gesture commands).

5.2.1 System Tutorial

The first time an astronaut engages the UI (Figure 38, yellow/gray), a tutorial will be launched with the goal of introducing the features of the UI as well as configuring the software for astronaut-specific (vision metrics, heart rate, name, etc) all of information will be saved to a local database for future reference.

Next, the astronaut will learn general navigation through the user interface, starting with how to control objects, menu icons, etc through hand gestures, voice, and eye movements. Once the astronaut is familiar with controlling their virtual environment, they will be introduced to the basic applications (navigation, illumination/lighting, EVA system state, sampling), their features, and basic information for how the applications work.

5.2.2 Control Methods

The applications will be controlled using voice commands, hand gestures, or a combination of both (59). Unique voice commands will be implemented to navigate through the applications.

The home screen of our application will allow voice controls to navigate through different areas. Each separate area will have their own distinct voice commands.The tables below show which key phrases will be used to navigate through the application.

Home

Key phrase	Action done
“Start navigation”,”Run navigation”	Opens the navigation overlay
“Start health”, “Run health”	Opens the vitals and health overlay
“Start task”, “Run task”	Opens the task overlay
“Close navigation”,”Stop navigation”	Closes the navigation overlay
“Close health”, “Stop health”	Closes the vitals and health overlay
“Close task”, “Stop task"	Closes the task overlay

Navigation

“New mission”	Updates the area to the current mission by downloading NASA Digital Terrain Models.
“Locate lander”, “Find lander”	Puts a marker and distance to where the lander is.
“Locate objective”, “Find objective”	Puts a marker on all objectives of the current mission
“Hide lander”,	Hides the marker for the lander
“Hide objective”	Hides the objectives.
“Select objective”	Opens list of objectives.

Lighting

“Start night vision”, “Run night vision”	Turns on night vision
“Start enhancement”, “Run enhancement”	Turns on enhanced vision
“Stop enhancement”, “Close enhancement”	Turns off enhanced vision
“Stop night vision”, “Close night vision”	Turns off night vision.

Task

“Task complete”, “Task finished”	Opens a task menu where the user can select the completed task
“Select task”, “Choose task”	Opens a task menu where the user can select task to complete
“Hide task”	Hides all task

Tutorial

“Start tutorial”	Starts the tutorial
“Next”	Moves to the next part of the tutorial
“Back”	Return to the previous part.

Table 9: selection of voice commands concerning the a) Home Screen, b) Navigation, c) Illumination,d) Task Menu, e) System Tutorial.

Some of the built-in gestures from HoloLens 2 will be used (60) The main gesture used will be the selection tool. The selection tool acts like a cursor. It can “click” the implemented button. There is a consideration of creating another gesture that allows the voice commands to be locked until said gesture is used.

5.2.3 Sample User Interface

We designed our first draft of the User Interface shown in the following figures. As we gain more experience with creating UI in Visual Studio and Unity and as we come across unanticipated issues, the design will change.

Figure 42 demonstrates that the user will only have important information concerning the condition of their spacesuit on the moon. The user should not have his vision impaired at all while doing tasks and therefore certain menus will only open on voice command activation.

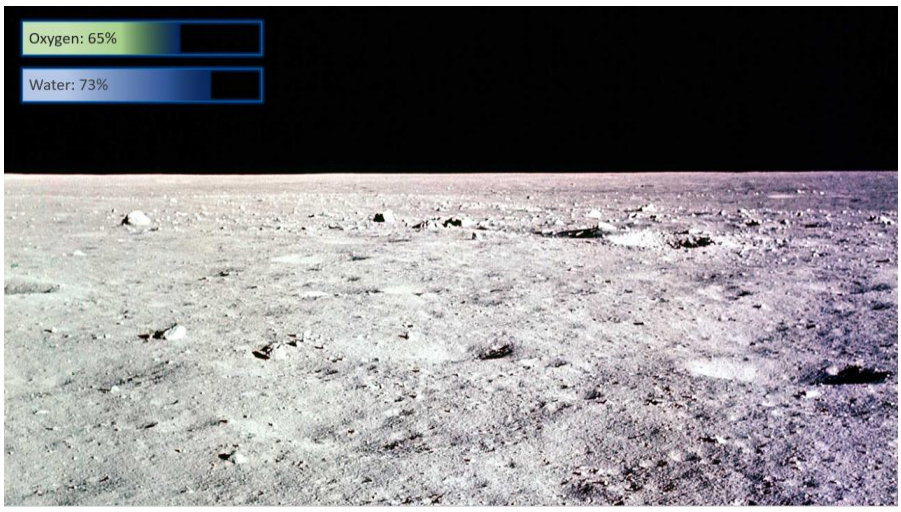


Figure 42: “Home Screen”

Figure 43 displays one of those menus: the navigation screen. The navigation screen will show the position of the user, any other users, lander position, and points of interest. The astronaut will be able to use his hand to zoom in and out of the map as he sees fit. At any point, the user will be able to add and delete waypoints to describe points of interest.

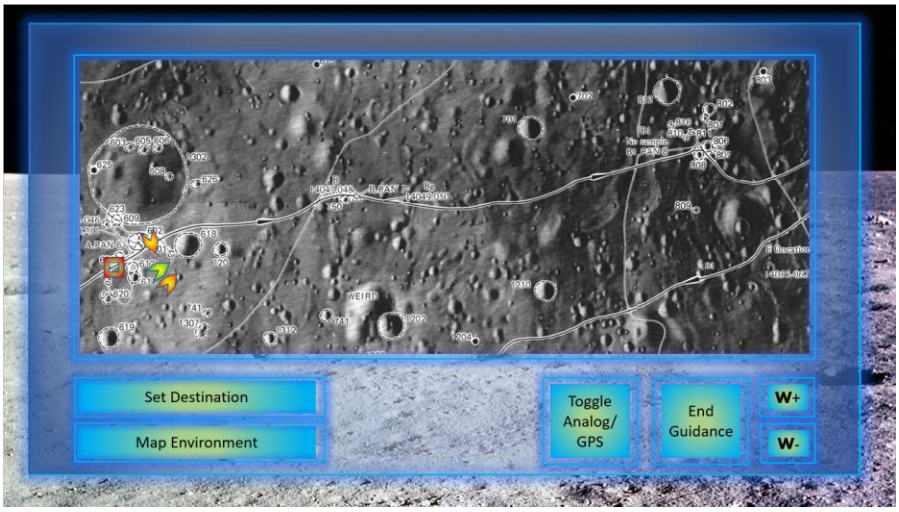


Figure 43: The Navigation Menu

Figure 44 shows the task instruction screen for when the astronaut embarks on a mission to geological sampling locations. A database will be created to store the instructions needed. The astronaut can edit instructions and review by Mission Control in the case that field conditions dictate a required change. From this menu, the user will be able to take pictures or video with voice commands as well as record his voice for note-taking.

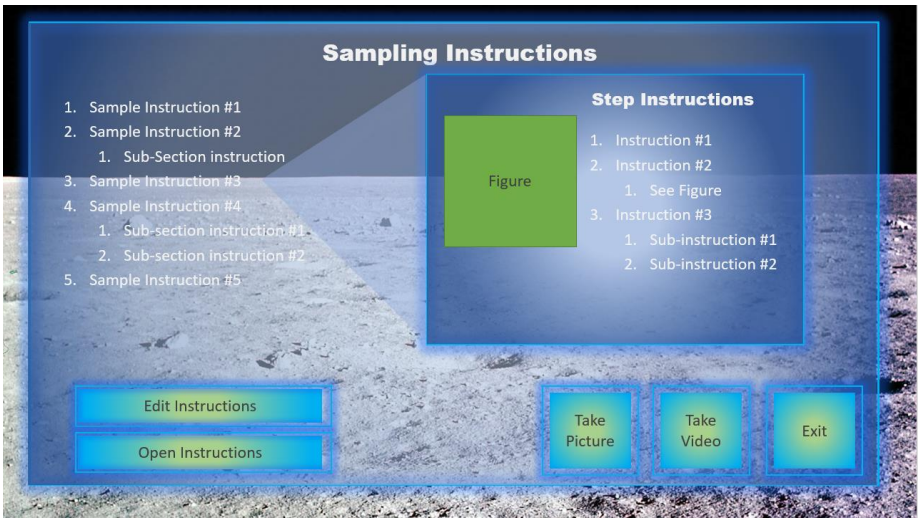


Figure 44: The Task Menu

5.3 Hardware Integration

Integration between the external sensors responsible for monitoring the astronaut health and suit vitals must be processed by the microcontroller and then wirelessly transmitted via a bluetooth socket to the HoloLens. Once received by the HoloLens, the data will be further processed, converted into a proper format, and displayed on the user interface in order to give the astronaut real-time updates.

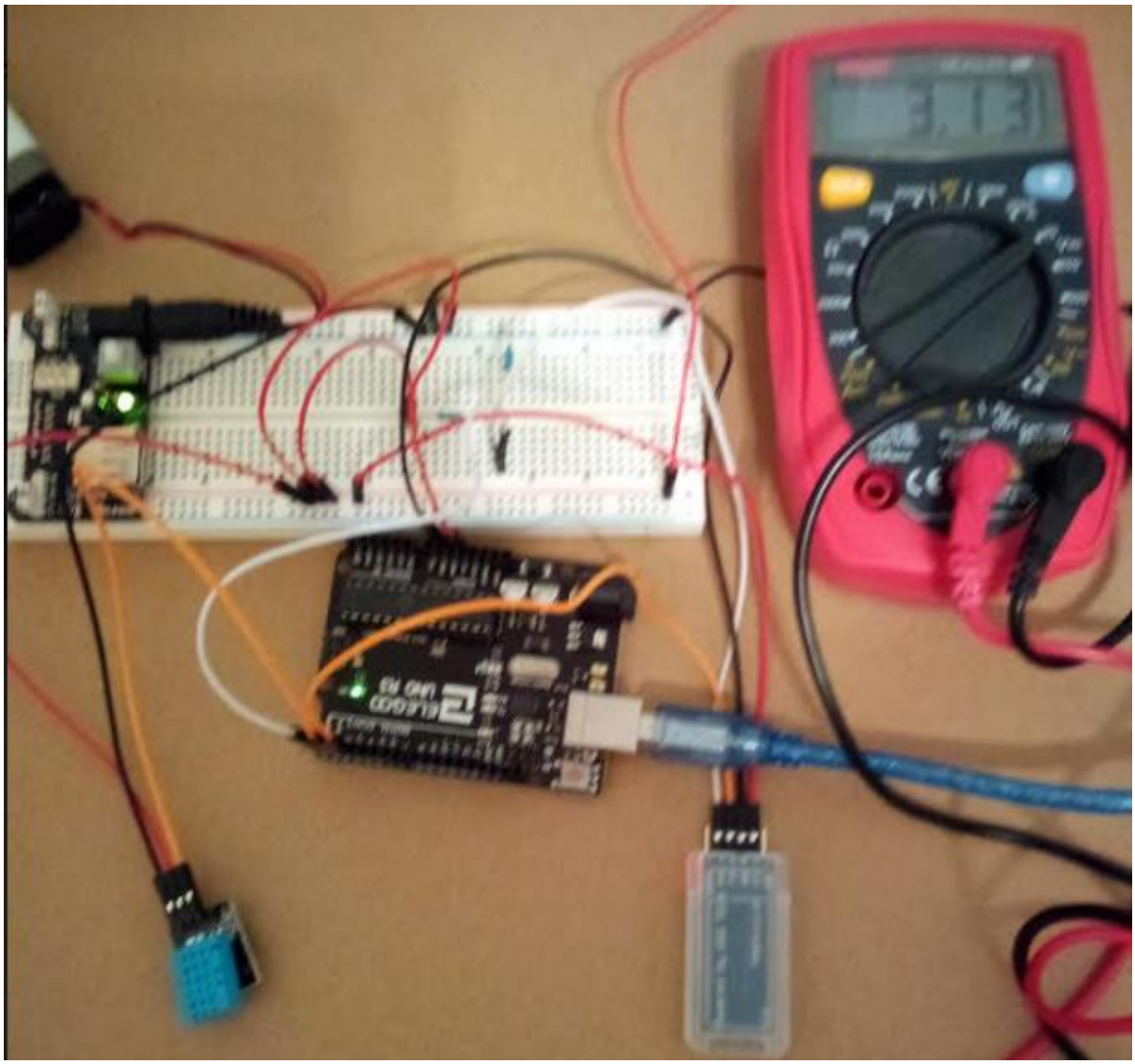


Figure 45: Microcontroller, temperature sensor, and bluetooth module testing setup

While the hardware design appears to be functional, we wanted to verify our design by prototyping the communication between a single temperature sensor, microcontroller, and a computer receiving serial data via the microcontroller. To this end we set out to set-up a basic prototype to demonstrate the integration of the external sensors with the software development environment. The HC-05 bluetooth module requires a working voltage of approximately 3-3.3V on the receiving pin (RX), but it must connect to the 5V transmit pin on the arduino. A voltage divider was set-up to avoid destroying the HC-05 module. The voltage seen from the HC-05 RX pin was measured using a digital multimeter and found to be within the proper range (Figure 45; 3.13~V).

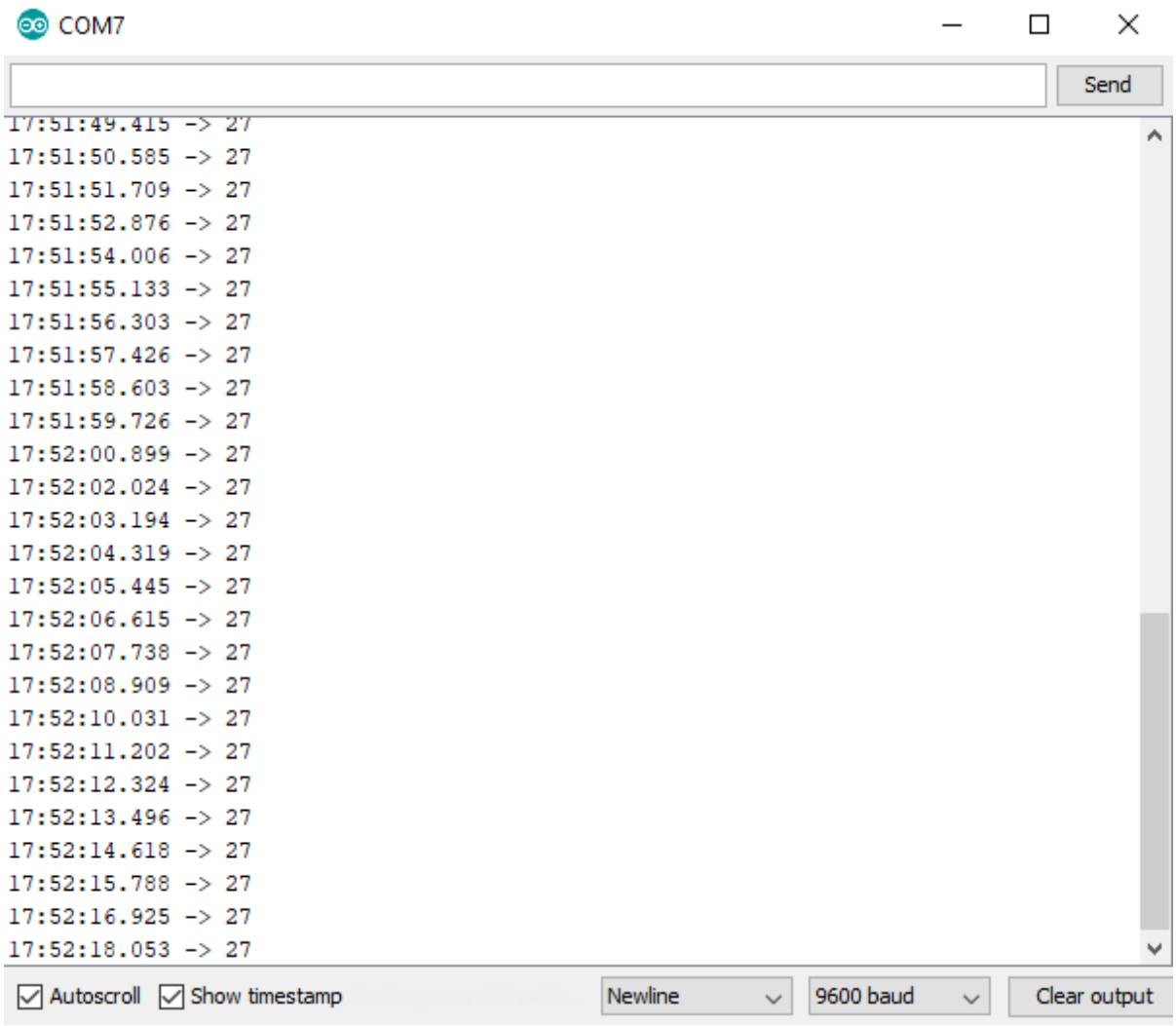


Figure 46: Serial communication between arduino, computer, and temperature sensor

Our circuit contained an Arduino microcontroller and thus we used the Integrated Development Environment (IDE) to program the board. The Arduino transmits data every second with a baud rate of 9600 for UART serial communication. The readout is pictured in Figure 46 with a timestamp to verify the Arduino was transmitting data at the programmed interval. The Arduino IDE was configured to send data through the computer's communication (COM) port 7.

After verifying the circuit worked as expected for serial transmission from the temperature sensor to computer via the arduino, we wanted to test whether we could produce the same data via the Visual Studio IDE using a C# console application. The reasoning being that the Microsoft HoloLens Mixed Reality Development Environment requires the Visual Studio IDE for development and debugging. Furthermore, C/C++ and C# are the only languages that can be used to produce applications within this development environment, excluding communication through sockets. Therefore, we wanted to verify that we could obtain serial data from a sensor via the microcontroller to validate our design of the software/hardware interface between the HoloLens and the external sensors.

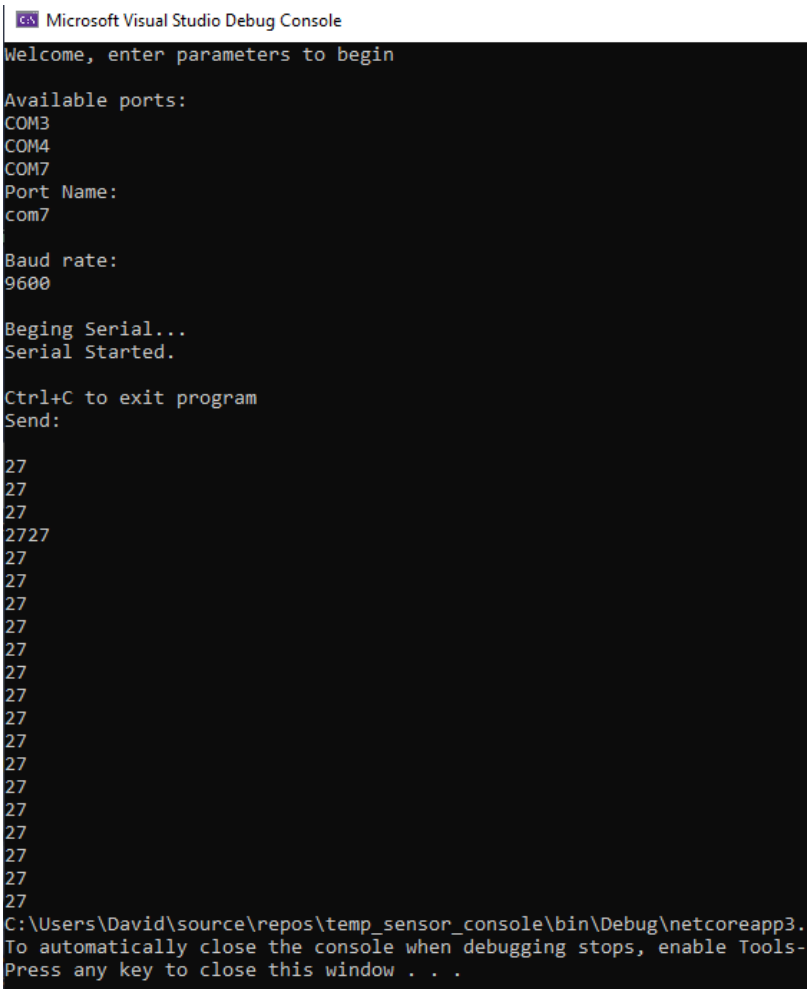


Figure 47: Serial communication (UART) between temperature sensor, microcontroller, and C# console application in Visual Studio

When the C# program runs, it first prints COM ports on the CPU which are available. Then asks the user to input the baud rate and the COM port to use for communication. The values of which are then printed on the console for verification by the user. Again, a baud rate of 9600 and COM port 7 were used. Figure 47 shows the result of the application, the serial communication proceeds as expected, printing the temperature reading on the screen every second, until the user kills the process.

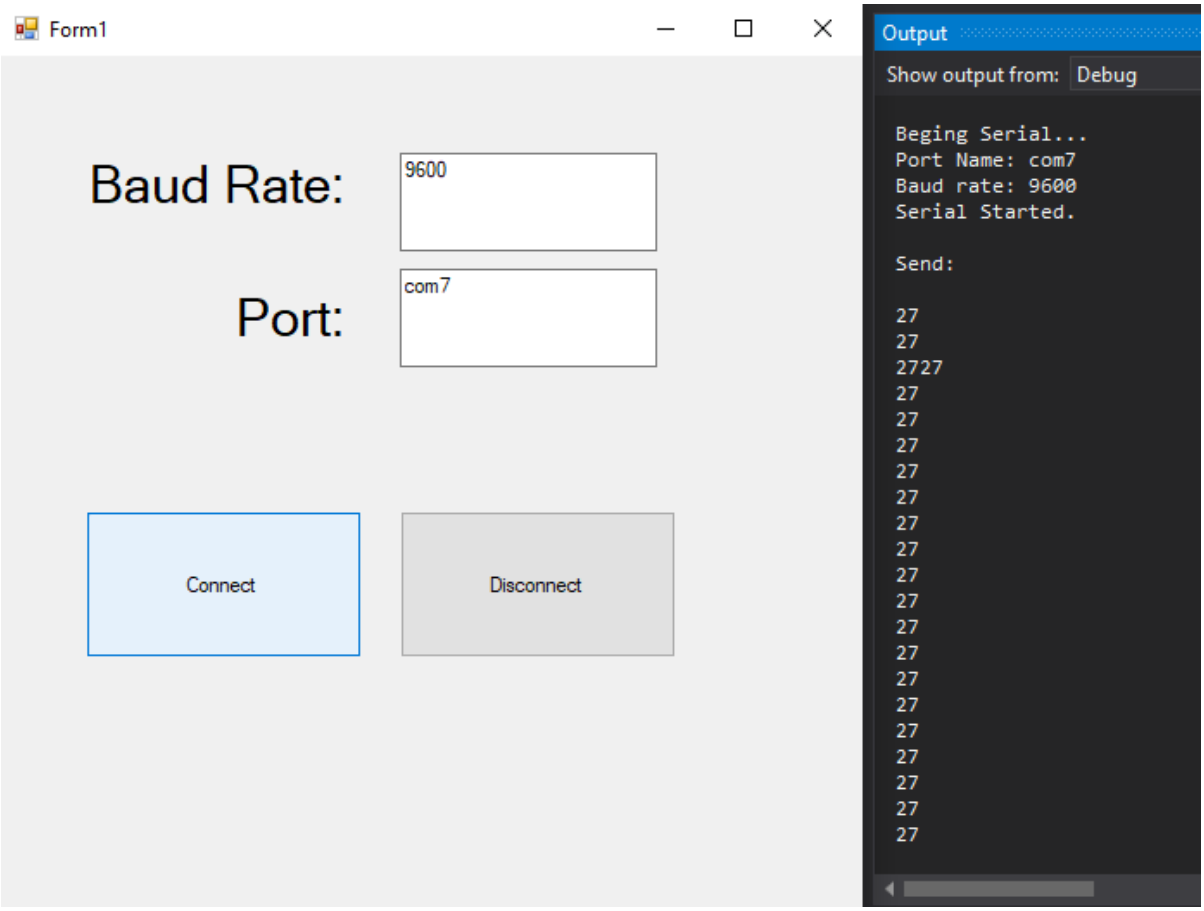


Figure 48: C# User Interface for connecting and disconnecting to the arduino serial communication port

Next, a simple user interface was built to simplify the process of testing serial communication between the external sensors and the software development environment (Figure 48). This functions similarly to the previous C# console script, asking the user for baud rate and a COM port, then connecting to the serial communication stream transmitted by the Arduino. The output from the connected data stream is printed onto the console.

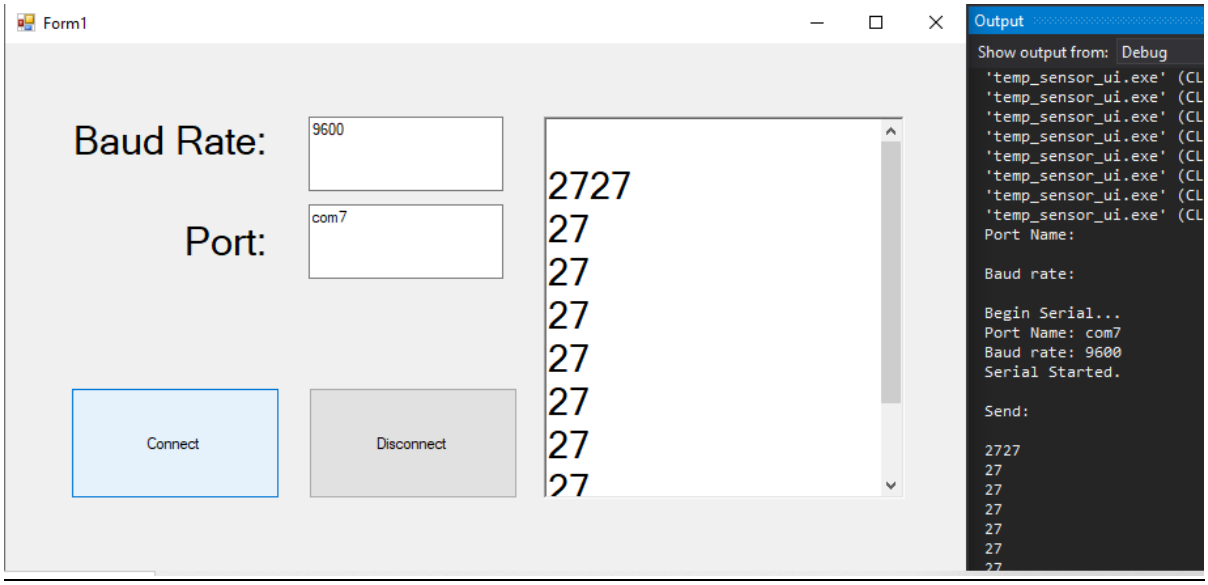


Figure 49: C# User Interface for printing out continuous serial port data stream

Lastly, a textbox was added to the UI which prints out the temperature data from the Arduino, appending new readings into the textbox as they occur. The output of which was verified by comparison to the output on the console (). While this likely doesn't parallel our exact implementation of our hardware components, it does demonstrate the feasibility of our design, since the HoloLens requires C# (or C/C++) for development. We can use this simple user interface within our HoloLens software for troubleshooting and testing our design from both the hardware and software side of this project.

6. Standards and Constraints

In this section, we discuss the industry standards that apply to our NASA SUITS Challenge project as well as the constraints including economic, time, environmental, health, safety, and sustainability.

6.1 Related Standards

For any engineering project, it is important to adhere to international engineering standards to ensure the project's marketability, security, and efficiency. Some of the most common organizations whose standards we will be discussing are IPC and IEEE.

IPC standards are important in the electronics industry, enabling "reliable, high-quality electronics by developing the trusted standards that drive the global electronics industry's success" (65). IPC contains over 300 standards developed by more than 3,000 industry professionals. We will be discussing their standards as it pertains to our electronic components.

IEEE standards are also of interest. Spearheaded by the IEEE-SA (Institute of Electrical and Electronics Engineers Standards Association), over 1100 standards have been created and 600 more are in development. (66) Many of their standards go beyond simple electronics and target computer networks and medicine. We will be discussing their standards as it pertains to our pulse oximeter and optical heart rate monitor.

6.1.1 PCB Design Standards

IPC-2221 standard is the foundation for designing Printed Circuit Board (PCB). The purpose of this standard is to establish design principles and recommendations that might be used in conjunction with the detailed requirements of a specific inter-coming structure standard. As illustrated in the document, PCB is a structure that is made of at least one layer of insulating material with electrical conductors installed with other electronic components. To make the PCB industrial more effective and standard, IPC-2221 published, providing distinct documents that focus on specific aspects of problems might come out in PCB design. Like PCB design procedure listed below.

PCB design steps:

- Create footprints of all the components and make a schematic by using a schematic capture program;
- Set design rules (such as size of the conductors and spacing between nets);
- Place and route all the parts;
- Pour ground planes connected to the ground nets;

- Run connection check and design rules check;
- Generate files for the board fabrication- usually it is Gerber files.

While in the design procedure, there also are important aspects users should pay attention to. For example, the width and the thickness of the conductors should follow the IPC-2152 Standard which illustrates that width and thickness is based on maximum allowable temperature rise. For the traces, users should not let traces melt during the condition of shorted circuit or surge current, which means there is an acceptable impedance should be found.

As technology develops, IPC-2221 is updating every few months and there are more IPC series standards relates to PCB design is coming out, the user should find what they need based on IPC's "Documentation Hierarchy":

- IPC-2222 Rigid organic printed board structure design
- IPC-2223 Flexible printed board structure design
- IPC-2224 Organic, PC card format, printed board structure design
- IPC-2225 Organic, MCM-L, printed board structure design
- IPC-2226 High Density Interconnect (HDI) structure design
- IPC-2227 Organic board design using discrete wiring

6.1.2 ISO/IEEE 11073

ISO/IEEE 11073 is an industry standard dealing with personal health and wellness devices and their communication to external computers. This standard targets devices such as glucose monitors, weighing scales, and pulse oximeters among others. The wrist optical heart rate monitor/pulse oximeter falls under ISO/IEEE 11073, specifically standard 10404.

One of the key issues of this standard is security. The pulse oximeter gathers personal data that should not be provided to anyone else but the user without the user's permission. In this project, the pulse oximeter will be transmitting the data wirelessly to the HoloLens II. To ensure that this personal data should not fall into the wrong hands, a secure encryption will be developed. Of course, while performing the missions tasked upon them, the astronauts will give permission to NASA Mission Control so that their vitals could be monitored remotely.

6.1.3 IEEE 802.15.4

IEEE 802.15.4 is a technical standard that defines the operation of low-rate wireless personal area networks (LR-WPANs), maintained by the IEEE 802.15 working group and defined in 2003.It is the basis for the Zigbee, ISA100.11a, WirelessHART, MiWi, 6LoWPAN, Thread and SNAP specifications. This standard intends to offer the fundamental lower network layers of a type of wireless personal area network (WPAN) which focuses on low-cost, low-speed ubiquitous communication between devices.

Key 802.15.4 features include

- real-time suitability by reservation of Guaranteed Time Slots (GTS),
- collision avoidance through CSMA/CA
- integrated support for secure communications.
- power management functions such as link speed/quality and energy detection.
- Support for time and data rate sensitive applications because of its ability to operate either as CSMA/CA or TDMA access modes. The TDMA mode of operation is supported via the GTS feature of the standard.
- IEEE 802.15.4-conformant devices may use one of three possible frequency bands for operation (868/915/2450 MHz).

6.2 Realistic Design Constraints

Any engineering project must undergo scrutiny with regards to realistic design constraints in every field. While all constraints affect each other, we will be discussing them separately to dissect what solutions can be provided to said constraints.

6.2.1 Economic and Time Constraints

Far and away, the largest constraint on this project is economic. AR technology is very expensive for high-end computational machines devised by companies such as Microsoft and Magic Leap. The Hololens 2 is the largest portion of our budget: $3500. Cheaper alternatives were researched and selected should there be lapses in funding. Our project is partially funded by the George Jackson Foundation and we are pursuing the Florida Space Grant as well to reduce the burden of our budget on our bank accounts. Many hardware components were also chosen due to their cheaper cost. In some cases, we had to sacrifice efficiency for cost.

Additionally, our project has quite a few time constraints. The NASA SUITS Challenge imposes a deadline of March for our Software and User Interface Design. If selected, the Hololens II will be sent to Houston for testing in their virtual laboratory the week of April 17th. Therefore, we have established a set of milestones described in a section below detailing our progress to a complete project by April.

6.2.2 Environmental, Social, Ethical, and Political Constraints

The NASA Artemis Mission end goal is to travel to the moon in 2024. Therefore this device has to be able to function on the moon. However, because the astronaut will be wearing the AR headset in his pressurized suit, the Hololens II will not have to endure the harshness of the environment on the moon. Navigation is the biggest component of our device impacted on the moon versus the earth. To tackle that issue, our headset will be able to navigate with both analog (heading, bearing) and gps inputs depending on what NASA needs. The device will itself not have any impact on the environment, but it will make it easier for astronauts to complete their tasks (geological sampling, etc.) on the moon. This project does not have any social, ethical or political constraints.

6.2.3 Health, and Safety Constraints

In terms of health and safety, our project has many constraints. In our testing procedures, we outline below how we will be abiding by the proper safety protocols with regards to the pandemic.

The device itself has health and safety constraints. The optical heart rate monitor has to be designed so that no electrical components touch and possibly damage the human skin. The intensity of the LEDs, while typically harmless, should be limited as well. The headset can have a negative effect on the eyes of the user. Many bright colors might induce epileptic seizures, which should be avoided. In addition, prolonged use may harm the user’s eyes. Each of these safety concerns will be presented to the user upon puting the HMD on and throughout the system tutorial.

6.2.4 Manufacturability and Sustainability Constraints

Given the end goal of this project is to help astronauts on the moon, the devices would be only manufactured a few times for the small number of astronauts that will be on the Astronaut mission. All devices used by astronauts also are designed with the astronaut in mind and need to be tailored to the specific person. Our project would not be mass produced.

The headset will be utilized on the surface of the moon, but inside a pressurized spacesuit. It will not be subject to the moon environment. However, we should consider the launch of the rocket itself. The launch will induce large vibrations on everything on board the rocket. Therefore, the hardware in the project should be able to withstand strong vibrations without creating any deviation of the device function.

7. Testing Procedures

This section details thoroughly the extent to which the various elements of our design will be tested. We will discuss the experimentation planned on our hardware as well as our software. The final design, if selected by the NASA SUITS Challenge team, will be tested in moon-like conditions and simulations at the Johnson Space Center.

7.1 Hardware Testing

Initial testing will focus on each hardware component separately prior to their combination. This extends to both the external sensors and the Hololens II headset itself. All circuits built will be first simulated in MultiSim prior to physical testing.

In the process of experimentation, human subjects may be used to assess the functionality of our device. Half of our subject pool will be experienced with the HoloLens 2 and know the features of the project. The second half will have no experience with the HMD. Due to the pandemic, we will also be following the proper safety protocols in prototype testing. It will be limited to a small number of people who will be required to stand six feet apart. We will mandate every participant to use facial coverings. Materials used will be cleaned with alcohol wipes between every test.

In terms of location, most of the hardware experiments will be done in the Senior Design laboratory in CREOL.

Optical Heart Rate Monitor

Preliminary Test: I-V Curve
To measure the efficacy of the IR and red LEDs, we replicated the datasheets I-V curve through a simple source-resistor-diode circuit. The diode current and voltage were measured utilizing the Digilent Analog Discovery 2 given to us for testing purposes. The results can be shown in Figure 49.

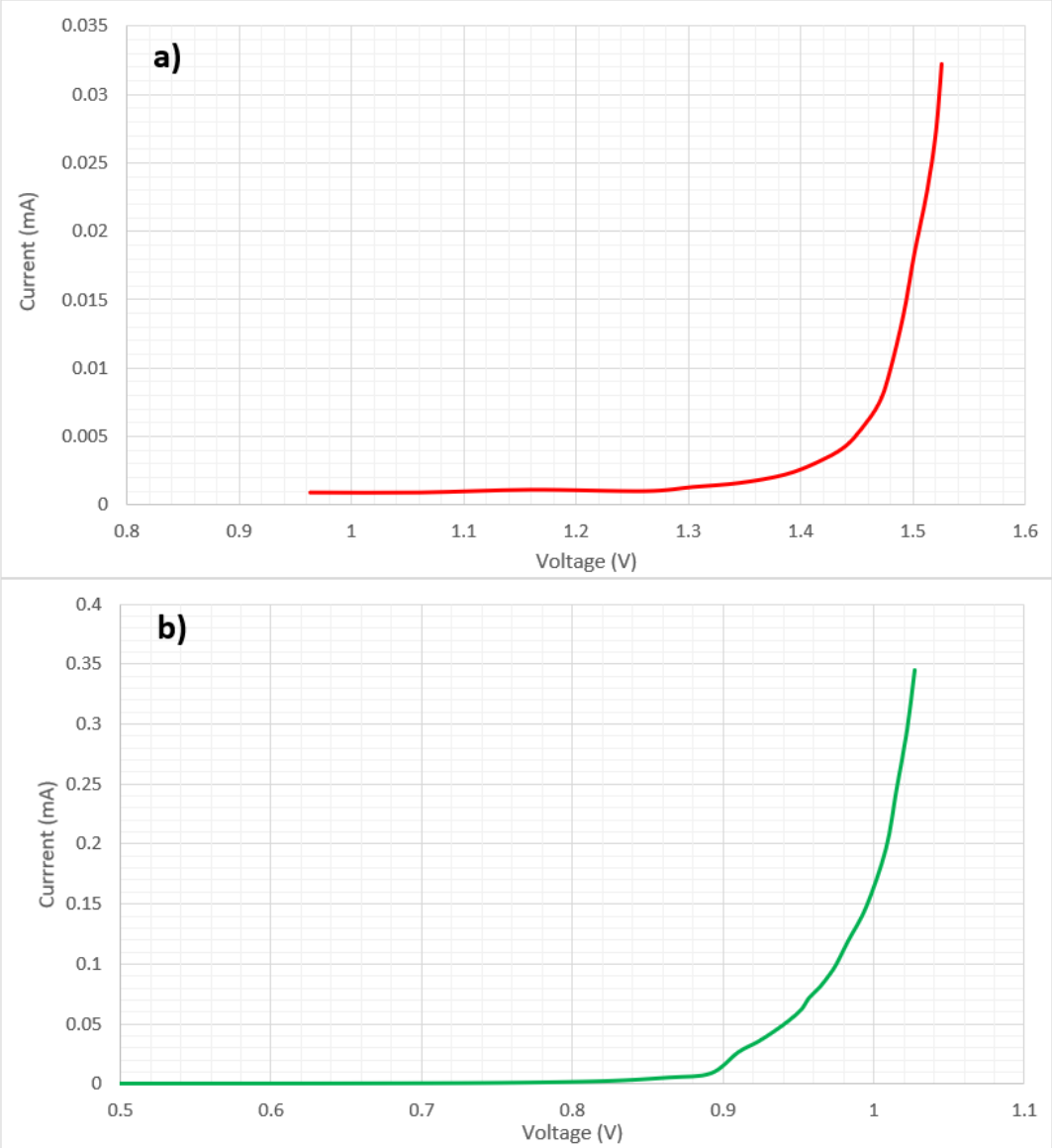


Figure 50: I-V Curve of the a) red and b) IR LED.

Breadboard Oximeter Testing

The circuit in Figure 25 was built on a breadboard. The first test was on a subject's finger as it is much easier to attain blood pulsary information from the finger than the wrist. The data was analyzed through an oscilloscope. Figure 50 shows the resulting waveform.

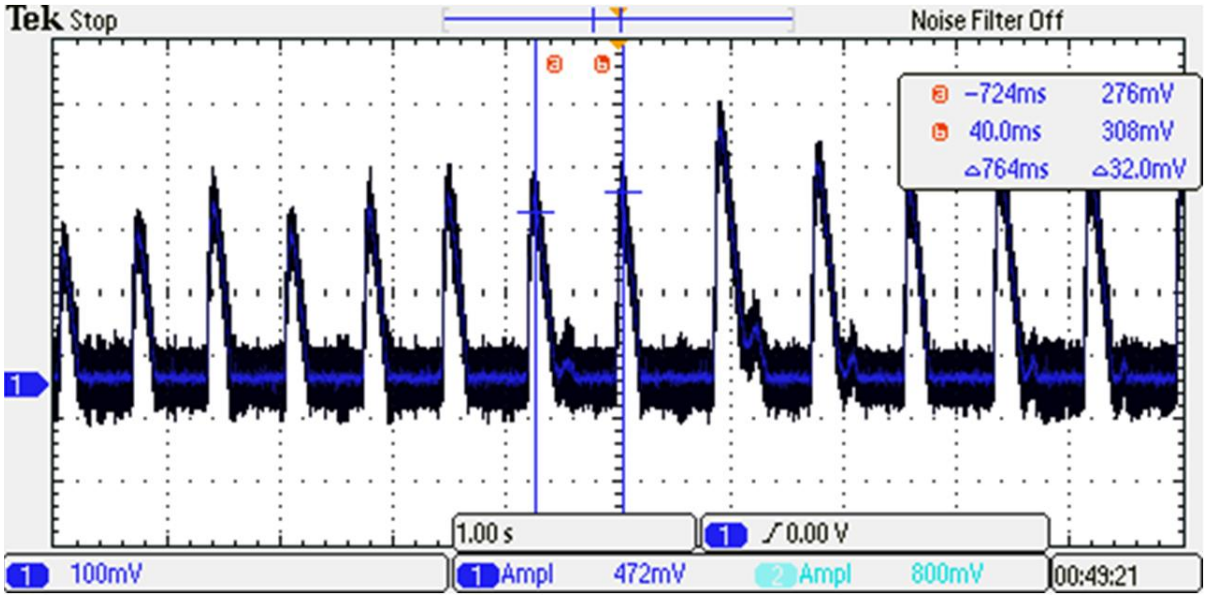


Figure 51: Waveform of breadboard circuit

The heart rate was determined using the following:

$$Heart\ Rate = \frac{60}{764\ ms} = 78.53\ BPM$$

The average resting heart rate of a healthy human rests between 60-100 beats per minute. To test the accuracy of our design, we tested it against an off-the-shelf pulse oximeter: 500DL Pro Series Pulse Oximeter. Table 10 details the accuracy results on one human subject:

Heart Rate Values	Sample Size	Average Error
70-75	6	2.63 %
76-80	31	2.58 %
81-85	19	4.55 %
86-90	26	2.27 %
91-95	18	2.49 %
96-100	2	4.05 %
101-105	1	4.76 %

Table 10: Error of our oximeter design based on heart rate values measured.

Once we can determine the heart rate and oxygen saturation reliably from the user's finger, we will move the LEDs and photodetector to the wrist and perform the same experimentation. Our goal is to reliably attain the vital information of the user with 5% error rate. After a desirable result is achieved through a breadboard circuit, Figure 29's PCB circuit will be created and tested. Primary testing will be on the finger and then the wrist.

At every point we will be testing to see whether the switching system of the optical heart rate monitor would contribute errors or provide misinformation to the user. In that case, we would repeat our breadboard testing with the circuit in Figure 26, the double photodiode circuit.

Power Source Testing

On the moon, the stability of the power system is super important, based on the known information, astronauts always go out of the space ship for 2 to 4 hours, which means our battery should at least supply energy for the HoloLens system steady for 2 hours. To perform the stability, we will connect the system together, turn on the IR camera as the astronauts will do on the moon and turn on the heart rate monitor for 10 minutes. If the system runs steady for 10 minutes, then turn it off, and check the battery usage, then we can get our system's theoretical maximum operating time. If the system cannot run steady for 10 minutes, then we might need a heat sink for the system or we should change our power supply system.

Alternative Headset Testing

On December 9th, 2020, we will be testing the Dream Glass 4K Headset at the Institute of Simulation and Training at UCF. The metrics we will be looking for and the questions we will be asking include:

- Ergonomics:
    - Does the headset impair the astronaut in their ability to do tasks.
- Software Development Kit:
    - Software Language
    - User Interface Interaction
    - Compatibility with Image Processing languages
- Processing Power
    - Ability to perform real-time complex image processing
    - Display real-time user interfaces
- Battery Life
- FOV
    - 90-degree FOV is claimed

7.2 Software Testing

The following are software and UI parts that we are planning on testing:

1. External Sensor Communication
   Test Protocol: We will be testing the wireless communication module that each external sensor will use to communicate data to the HoloLens.
   Metrics: The speed at which data is transmitted to the headset will be measured.

2. Voice commands and gesture control
   Test Protocol: We will test our voice command and gesture control software. As planned, the tester will speak commands or move his hands to find the menus he or she needs.
   Metrics: We will measure how effective the voice commands and gesture control are qualitatively.

3. Navigation
   Test Protocol: The user will utilize the HoloLens to guide him or her from point A to point B. The same user will also use conventional guidance technology such as Google Maps to guide him the same distance in a different direction.
   Metrics: We will be measuring the amount of time that the user takes in walking the guided path that either the HoloLens or the Google Maps provides him.

4. Illumination
   Test Protocol: The user will use the headset in a completely dark environment as well as one with high contrast between light and dark. He will attempt to navigate and complete basic tasks.
   Metrics: We will be measuring the amount of time that basic tasks are done in various lighting environments to determine the effectiveness of the IR camera and image processing of the headset.

5. Science Sampling
   Test Protocol: The user will carry out a preordained task using the AR Hololens as a guide. The user will not know anything about the task prior to performing it.
   Metrics: We will be measuring qualitatively how effective the user it is at performing his or her task and how he utilizes the information that the HMD provides them to aid them.

7.2.1 Image Processing

The image processing techniques were tested through python using libraries. The libraries used were cv2, skimage, and numpy. The techniques from the theory were implemented and used on a landscape picture of a mountain at night to simulate a low light situation. Some parts of the mountain are hard to discern, which attempts to fulfill our low contrast requirement.

The tests were run on Google's colaboratory notebook. The processing time is not indicative of any particular system and was generalized through using the notebook as a constant. This IDE was chosen due to its ability to change the code and share the code without downloading a file. It is cloud based. The photo used in the image was taken from a free to use image from eberhard grossgasteiger from pexels. The image is of a mountainous area at night which was picked to simulate our environment. The original image size is 3648x5472. This is much larger than the images that will be taken from the HMD. The reason for the large image file is to keep as much information possible to allow the imaging techniques to enhance contrast as much as possible. The image was changed to grayscale using a rgb to gray function and io.imread from skimage. It creates the image as a numpy array. This numpy array was then used as the main image for the test below until bit plane slicing.

From bit plane slicing, the cv2 library was used and the function cv2.imread was used. The change in how the image was read was due to an error caused by one the functions used for the technique. The cv2 version is used for the histogram equalization technique as well. To simplify the system and the data needed for libraries on the device, the computer vision library will be the main library installed onto the processing unit.

Negative Testing

The first technique implemented was creating a negative image. The negative image takes very little resources to create. It inverts the light values and the dark values. It's a useful technique when looking at a black background and a gray foreground. This might be useful in identifying some features that are harder to spot regularly.

Figure 52: A picture of the landscape(left) and its negative(right)

Log Transform Testing

The use of log transform was implemented using a function that was already defined in the skimage library. It takes a gain parameter from 0-1. The test was done in increments of 0.2 from 0.2 to 1. The 0 case was not used because the transform created an image with no information. We can see that the lower gain creates a darkened image compared to it at higher levels. This technique may be useful when we have much more light and need to enhance the dark spots. The function was also tested to see what happens if the parameter was over 1 and the input value of 2 was put in. In the grayscale case, there doesn't appear to be a change when the input value was set to above the gain parameter. In the rgb case, there is a distortion of color.

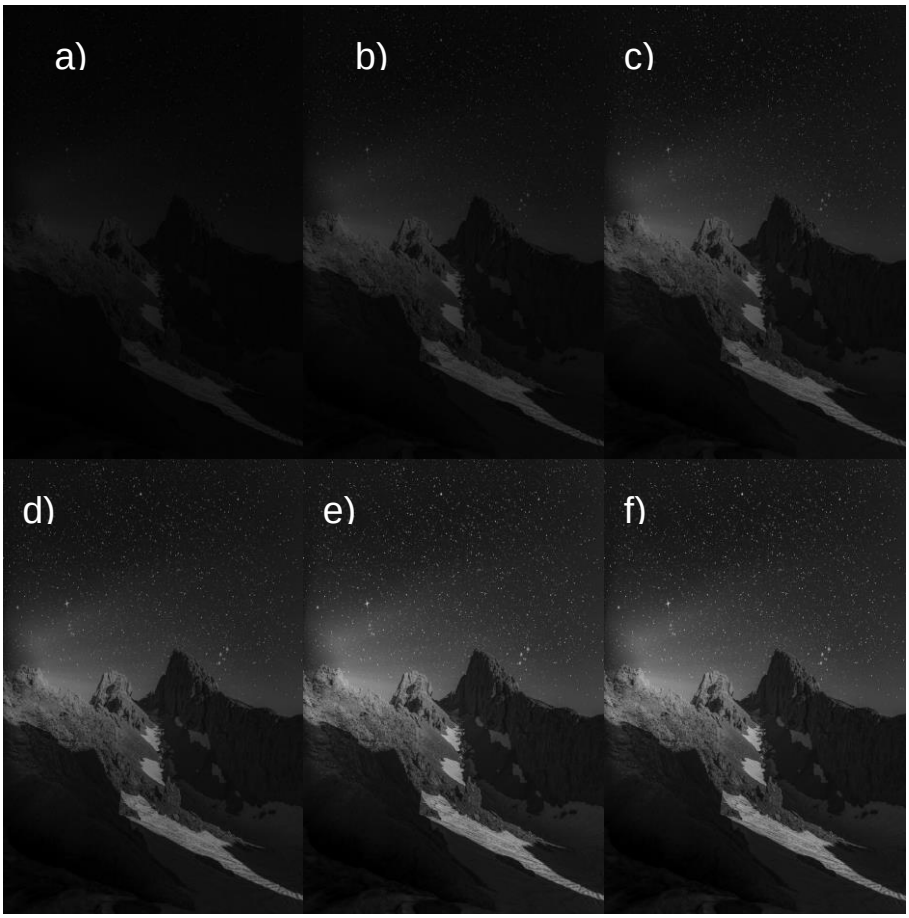


Figure 53:log transform set with gain parameter a) gain set to 0.2. b) gain set to 0.4. c) gain set to 0.6. d) gain set to 0.8. e) gain set to 1.0. f) gain set to 2.0

Inverse Log Testing

The log function also has an inverse parameter. The same experiment was tested with the same values to examine the differences between the regular log function and its inverse. When comparing the two we can see that the regular log function brightens up more quickly. In our test the inverse log was not as effective in this image. It might be more useful when we use the negative version of this image to create more white or an image with higher intensity values.

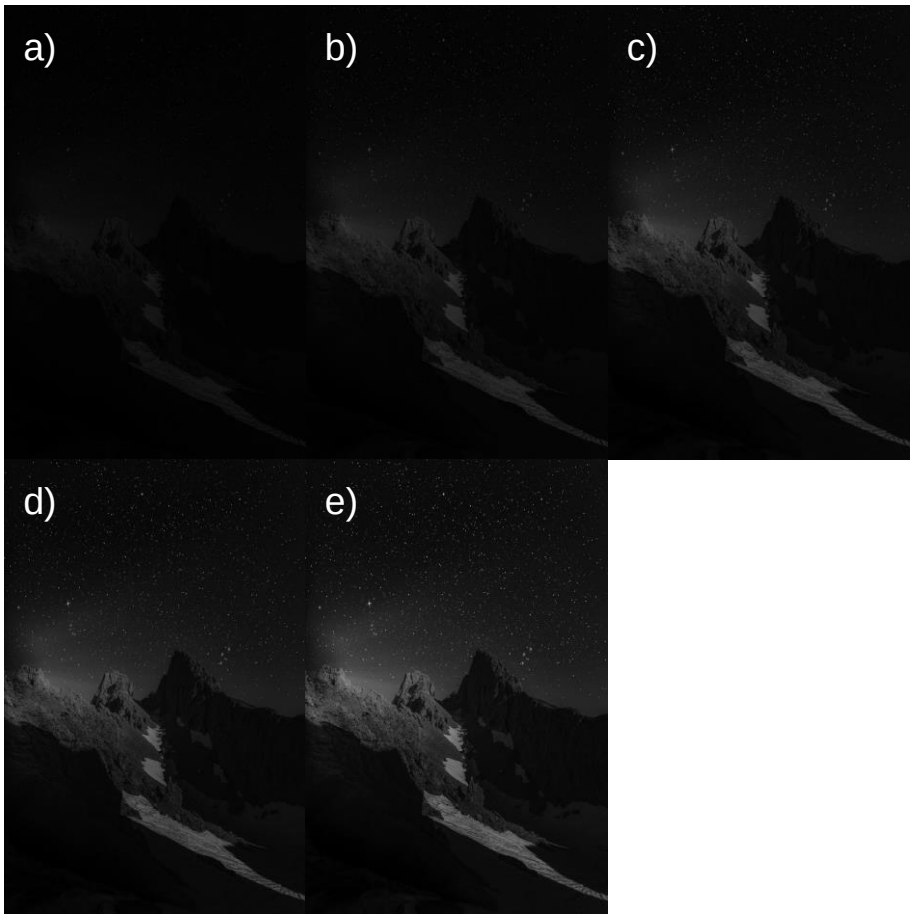


Figure 54: Inverse log transform with a changing gain parameter a) gain set to 0.2. b) gain set to 0.4. c) gain set to 0.6. d) gain set to 0.8. e) gain set to 1.0

Threshold Point Processing

The point processing technique was tested to see its capabilities in this environment. It was tested by creating a threshold value and changing any of the values below a threshold to black and any above threshold to white. This creates a binary image which gives a large amount of contrast. It is not a localized technique so the threshold has to be picked depending on which area of the image one wants to emphasize. The threshold level was slowly increased in increments of 50 starting at 50 and ending at 250. For our image, this technique doesn't seem useful. It had quite a bit of processing time because it uses loops to create a processed image. It processes each pixel individually which may create a lag when we use it for real time processing.

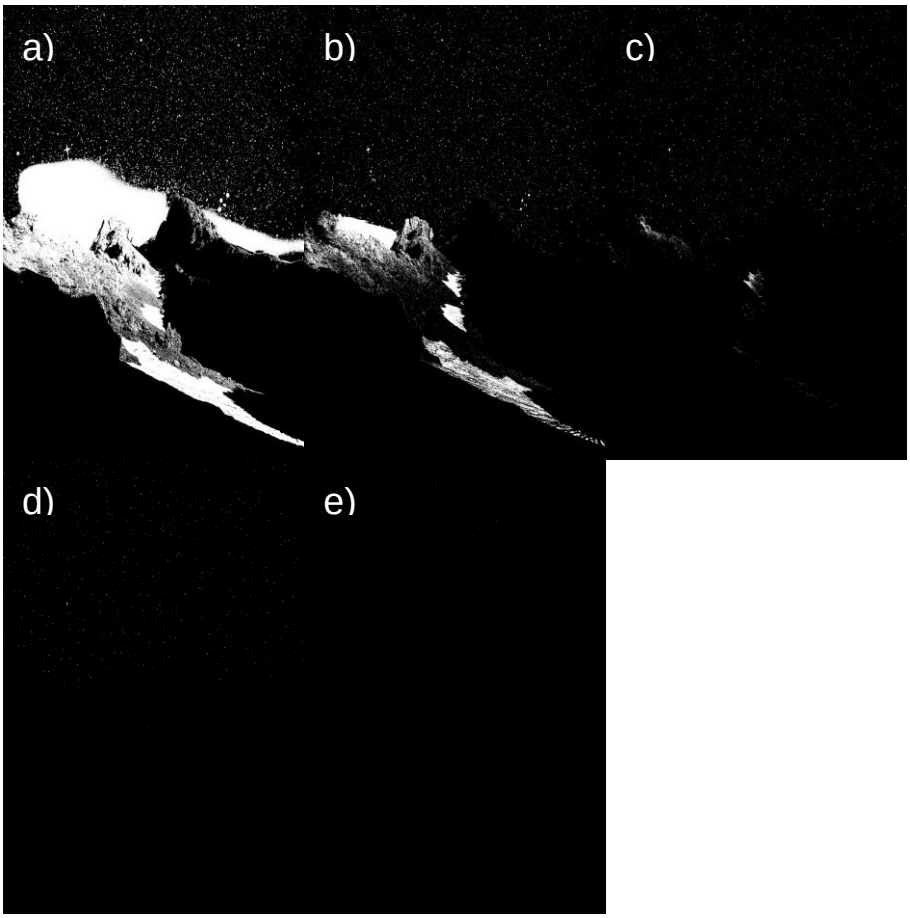


Figure 55:The processed image with variable threshold. a) threshold set to 50. b) threshold set to 100. c) threshold set to 150. d) threshold set to 200. e) threshold set to 250

Gray Level Slicing Testing

Gray level slicing was next tested. It uses a similar process to the point processing technique but uses two thresholds. The two points allow a level where the values in between are preserved. It may be harder to optimize due to having to choose two values. The two values have to create a clearer image for the environment. It also uses loops which can create a longer processing time. In our testing we can see how choosing different thresholds can create emphasis on different objects in the image. This technique would be useful assuming that the environment isn't changing too quickly such that the optimal thresholds could be found.

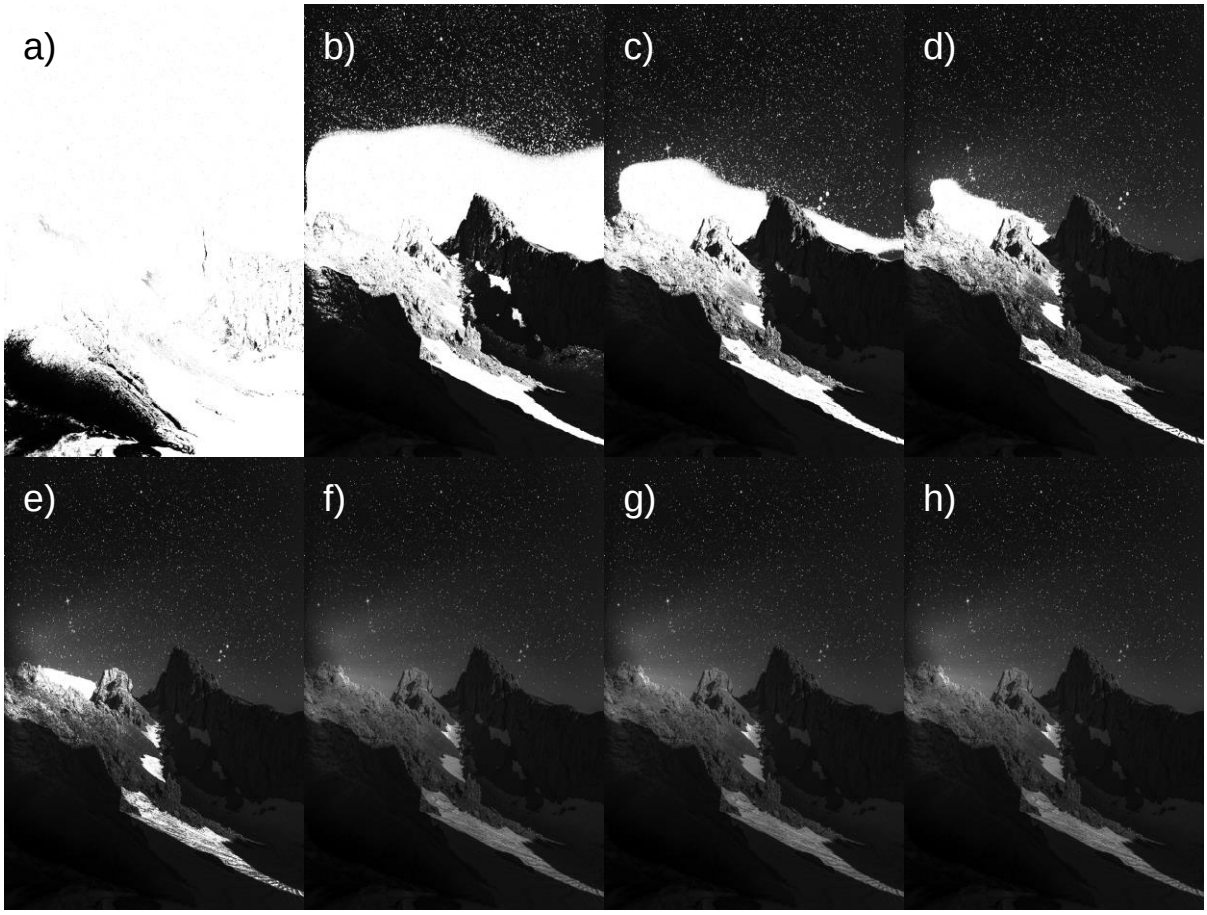


Figure 56: Gray level slicing with two thresholds labeled T1 and T2. a) T1=10 and T2=150. b) T1=30 and T2=150. c) T1=50 and T2=150. d) T1=70 and T2=150 e) T1=100 and T2=150. f) T1=150 and T2=170. g) T1=150 and T2=200. h) T1=150 and T2=250.

Contrast Stretching

Contrast stretching was tested, but there was no discernable difference between the unprocessed image and the processed image. It works better in an image where there is less contrast in the image. Which could be useful in an environment where most of the intensity values vary less.

Bit Plane Slicing Testing

Bit plane slicing was then tested and showed some characteristics similar to gray level slicing. They can isolate certain characteristics in the image. Since we are using bit planes it is far easier to decide which values should be chosen by inspection. This technique looks to be useful due to it quickly isolating different parts of the image.

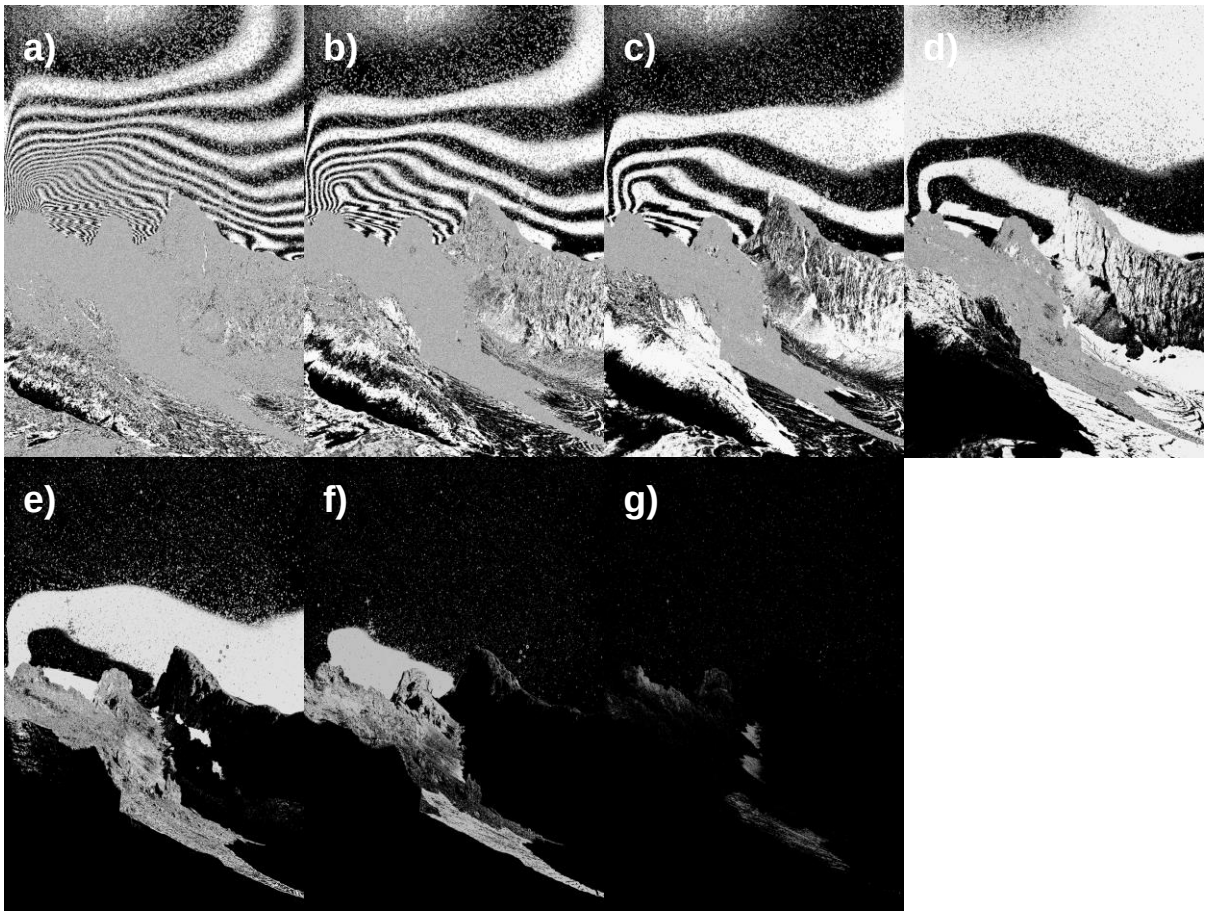


Figure 57: a) 2-bit plane slicing. b) 3-bit plane slicing. c) 4-bit plane slicing. d) 5-bit plane slicing. e) 6-bit plane slicing. f) 7-bit plane slicing. g) 8-bit plane slicing

Histogram Equalization Testing

Histogram equalization was tested and uses a library in cv2. It acts as a global transformation that normalizes the occurrence of any individual value. Looking at Figure 58 we can see how the equalization redistributed the values. The highest occurrence of a single value changed from around 4000 to 1400. This technique provided the most drastic change in lighting.

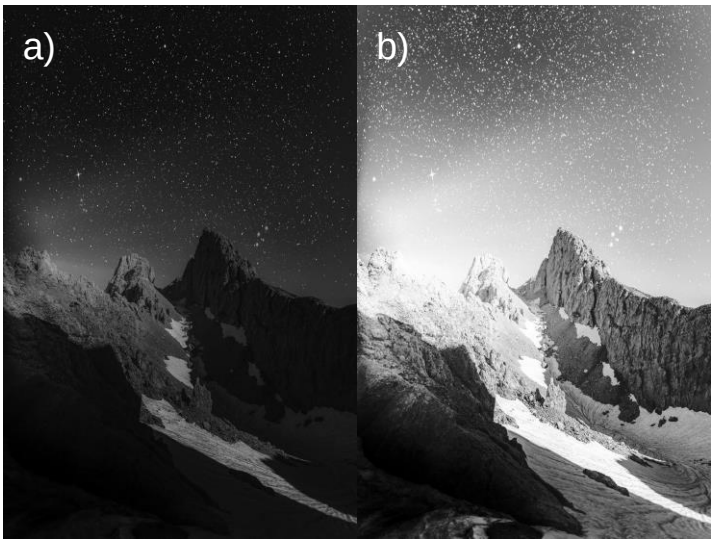


Figure 58: a) the base image. b) The equalized image

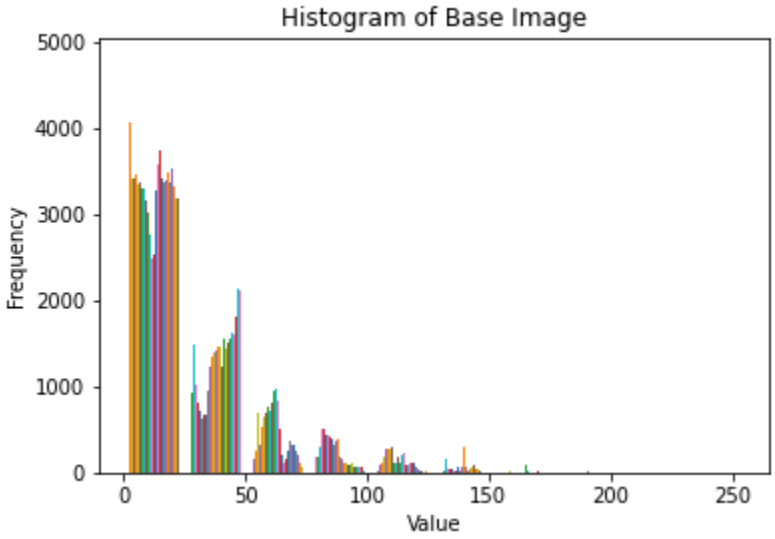


Figure 59: Histogram of the base image

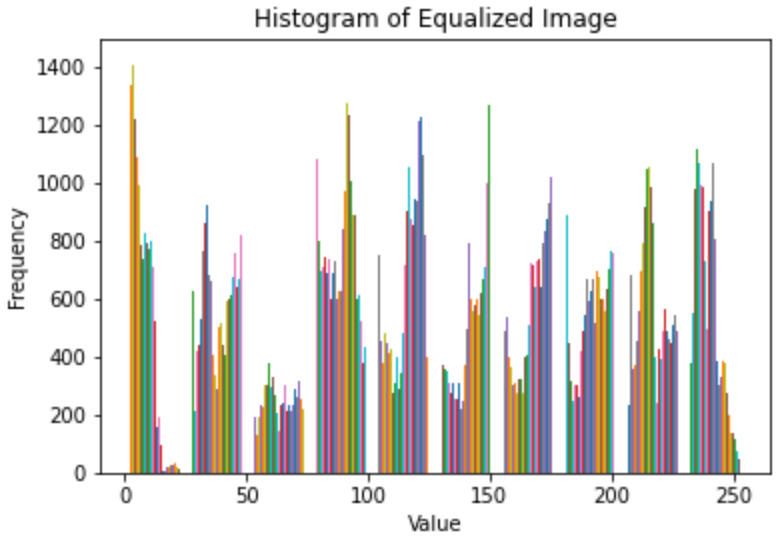


Figure 60: Histogram of the equalized image.

Contrast Limited Adaptive Histogram Equalization

From our experimentation, we believe that the CLAHE image processing algorithm might be one of the most useful ones. In Figure 61, we can see the results of testing in Google Colab's python IDE. We can clearly tell that the brighter and darker regions of the region have better contrast in the image. The image as a whole looks brighter and features are easier to identify. Increasing the tile size perimeter only marginally changed the image, up until we reached 64x64. We believe that this is dependent on the size of the image. In this case our image size was 228x329. A tile size of 64x64 would result in ~⅔ of the pixels being edge cases. This visibly distorts the image if the edge cases amount to more than 50% of the image, seen in the 64x64 case.

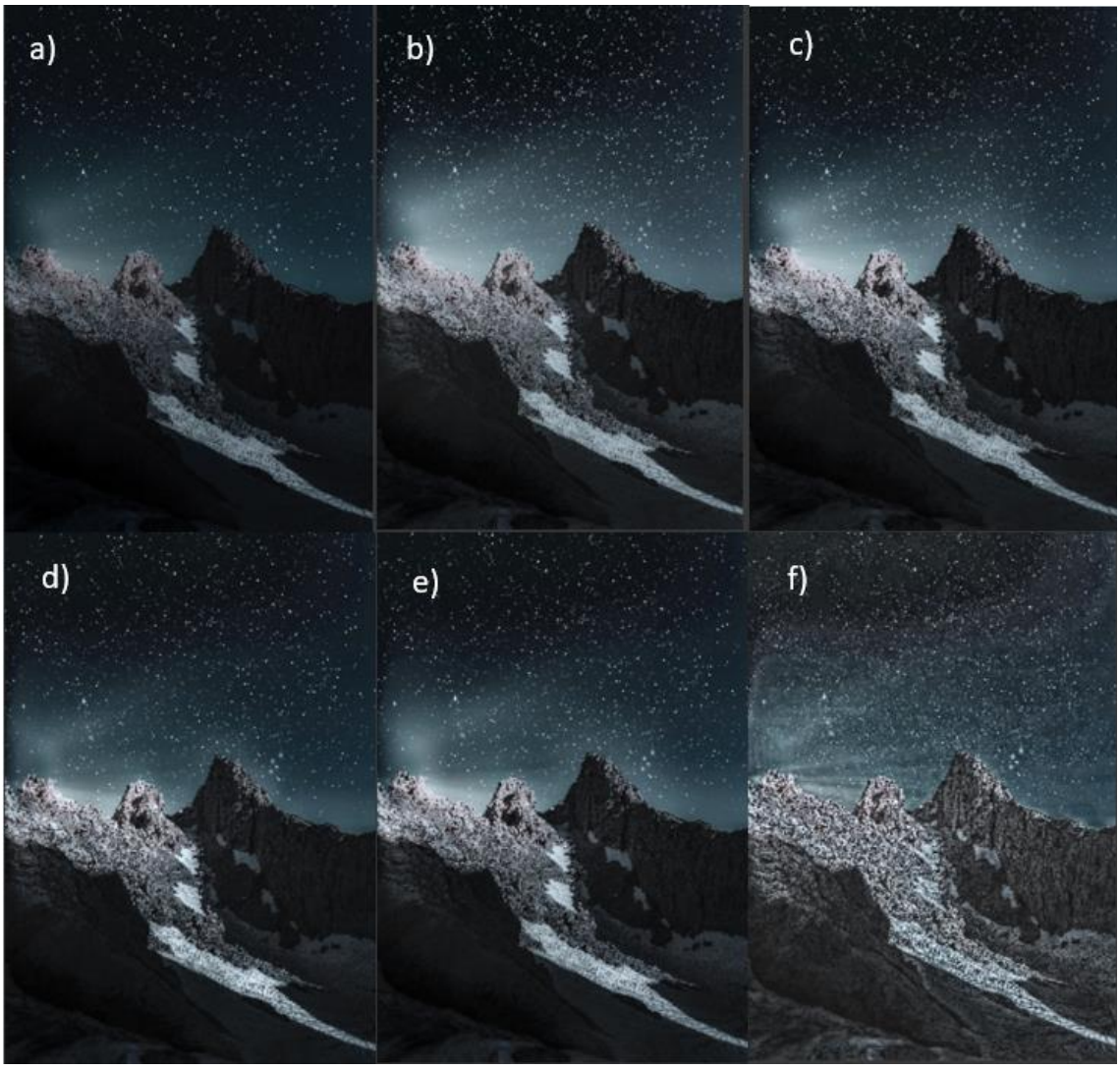


Figure 61: Results of CLAHE image processing on the a) original landscape image. The tile sizes used were b) 2x2, c) 4x4, d) 16x16, e) 32x32, f) 64x64.

Our main concern with this method is the intensity of the computation might be too heavy to measure in real-time. We did some basic runtime experiments to determine the computational dependence of the CLAHE algorithm based on the tile size parameter given, displayed in Figure 62. The CLAHE algorithm was implemented in Google Colab's python IDE. An internal timer was established to measure the amount of time for one image to be processed. Each data value in Figure 61 corresponds to an average of 100 runs.

Tile Length	Time (sec)
2	0.00168
4	0.00181
8	0.00179
12	0.00192
16	0.00196
20	0.00233
24	0.00243
28	0.00275
32	0.00301
40	0.00386
48	0.00472
56	0.00561
64	0.00688
80	0.00937
96	0.01256
112	0.01628
128	0.0206

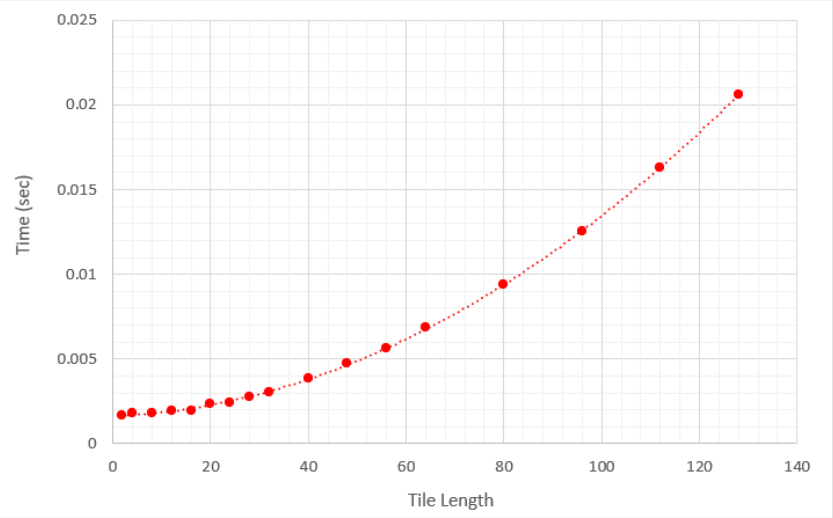


Figure 62: Runtime experimentation of the CLAHE algorithm. The clip limit was kept the same at an intensity value of 2.

These data, we realize, are not absolute. It is highly dependent on the processing system utilized and the additional tasks that the system was handling. However, we believe that useful information can be gleaned in the form of the relationship between tile size and run time. There is a very clear quadratic relationship between tile dimension size and image processing time signaling an $O(n^2)$ runtime of the algorithm.

Low-Pass and High-Pass Filters

Low-Pass and High-Pass filters are commonly used and well understood image processing techniques. Utilizing the Google Colab Python IDE and the OpenCV library, we mapped the effect of low-pass filters and high-pass filters in our image processing. The results can be seen in Figure 63.

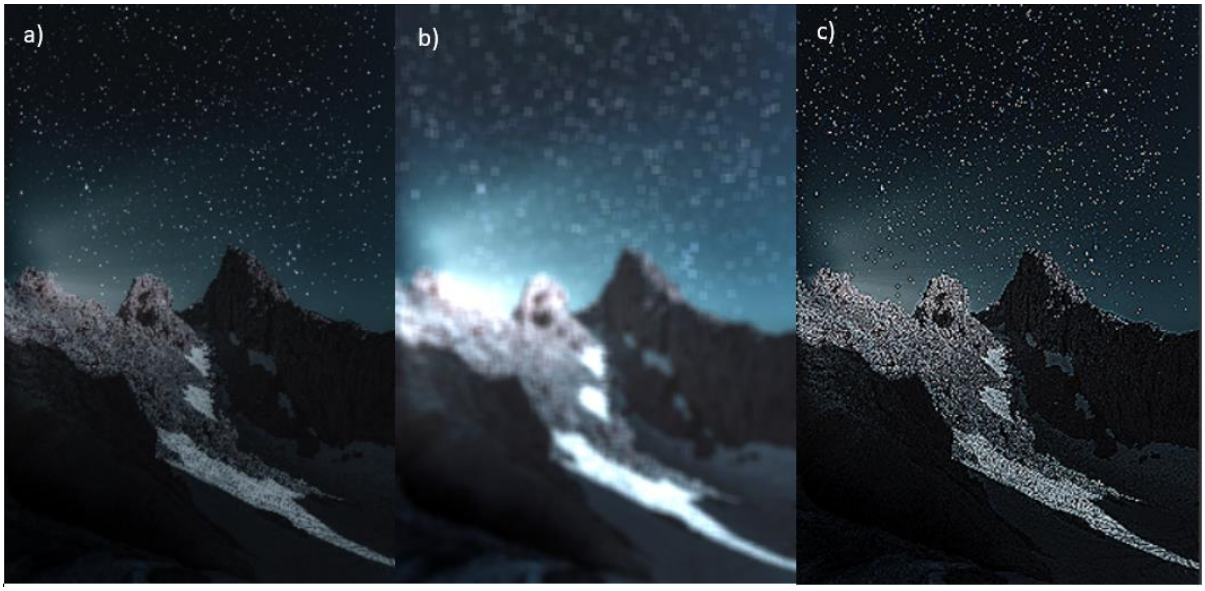


Figure 63: a) original image, b) low-pass filter implementation, c) high-pass filter implementation

Figure 63 shows fairly predictable results. Low-pass filters in image processing tend to blur or smooth the image, while high-pass filters tend to emphasize edges. However, information is lost in the process of filtering.

Real-time and Video Image Processing

The image processing techniques will be combined with a real time video feed to try and enhance the output video to the astronaut. The processed video will be then imposed onto the Glasses to act more like VR. . The video processing will most likely be using the computer vision library found in C# and python.

The algorithm chosen to enhance contrast was Contrast Limited Adaptive Histogram Equalization. It uses the principles that were explained earlier to create a contrast in a local neighborhood. It was originally implemented in Python using Google Colaboratory, which is similar to a Jupyter Notebook. This only worked for any videos that were saved onto the cloud. It had problems with access to a locally connected camera. To more easily implement the local camera, the environment changed to the IDE Spyder in the Anaconda environment. In this iteration of implementation, the capture device used was a built-in camera in a MacbookPro 2014. The library used for camera capture and processing was OpenCV. The only issue found in the python implementation is that there wasn't a great display option. The issue that the display caused was that we could not test in real-time whether or not the updated frames were being processed.

Another solution that was proposed was to take individual frames, and process it, and then recombine them to create a video feed. This method was successful in its implementation but

was ultimately too slow to be useful for the astronauts. For a small 5 second video at 30 fps, the processing time took around 2 minutes. This solution was ruled out due to the heavy processing time. The implementation was then moved to C# to become more native to the Microsoft Development Environment.

The environment chosen was Visual Studios 2019. It was chosen based on its interactivity with Unity and Windows. There wasn't a direct OpenCV library and the .net wrapper Emgu.CV was used. The challenges in changing languages was due to inexperience in the language. The developer on this processing had not programmed in C# before. The testing program used was created from a windows form and tested various different types of filters that were implemented into the Emgu.CV library. The two filters that were implemented were thresholding and histogram equalization techniques.

For thresholding, there were various implementations such as Binary, Inverse Binary, Otsu, and Truncation. Each of these would be fairly useful for applications where we want binary information to be either more clear or more contrasted. They are all global transform and would not be useful for a wide range of areas. Adaptive Thresholding was then implemented to create a localized area such that more information is kept. Through testing this implementation seemed only useful in machine learning or edge detection cases. Thus the histogram equalization techniques were implemented.

In the histogram equalization techniques, we found success with the implementation and the contrast it added to areas. It was tested in various settings and light levels. For lowlight levels, the tested environment was a living room with blackout curtains and lights turned off. In this setting there are some small amounts of light that still lit the room. In this situation, it was seen that there was an improvement in objects which could be recognized through the processed view. The result was seen in all light levels except for when there is no light or barely any light. This could also be seen as if there is simply a canvas of a single color or the light is so intense that the image values are at max value as well.

The concern of the runtimes were measured to well under the times needed to achieve 30 or 60 fps. The choice of 30 or 60 fps was chosen based on it being a digital standard and that the frames would be barred by the hardware instead of the software. The only main issue that arose was in its implementation into the HMD using the HoloLens' built in cameras. The permissions allowed on sensor data were not easily accessed.

Depth Mapping

The astronaut's HMD can be used to help the support crew to create a 3-D Map of their location. This uses two cameras. The two cameras will either take images or a video feed to create a 3-D mapping that can be sent back to the lander or a support to recreate the image

using a depth map and pose estimation. The only modification to our device is to add a secondary camera if there isn't one. The software only has to take a photo or multiple photos with both cameras at similar times. The data can then be sent back or collected to be uploaded at another time. A secondary place can do the processing and create a 3-D depth map or an interactive map using holograms. The astronaut becomes a data collector for future missions which can possibly create a full interactive moon environment in the future.

The process of creating this environment uses depth maps and image reconstruction techniques. The two cameras will create a depth map. The depth map gives enough information to create a reconstruction of the environment. This could be then used in unity to create holograms of the structures or environment taken such that the next set of astronauts will be better equipped for the terrain.

The concerns of creating such a program is that there may not be enough time to do this before our deadline. With all the other specifications in our documents, this is not a necessary one. It's usefulness is only seen in later missions. It will not add anything to the astronaut other than another task to be done. There is a possibility of taking photos of the environment automatically. This would mean that there would be a need to have a limit on how much data can be collected daily. The data limit would also have to be deleted from the device after each upload.

8. Additional Features

The following sensors are not critical to the function of the device, but are meant to add further usability or accuracy. As such, they will be added to the project if both time and funding allow it. They are not fundamental in any of the tasks required from NASA, but will enhance the information provided to the astronaut and mission control as well as enhance our testing procedures.

8.1 Gyroscope

A gyroscope is to be added to the system to be part of an inertial navigation system (INS). This system is going to be used as a secondary reference for the INS built in or as a primary INS if our HMD doesn't include one. The gyroscope will be used to measure directional changes of the astronaut.

The mechanical version of the gyroscope consists of gimbals, a frame, and spinning axis, and a rotor. The main theory behind how a gyroscope works is based on the resistance to change of the moving wheel. It stays parallel while all the other components move. Each gimbal has one dimension of freedom. They are connected in such a way that the inner gimbal's directional movement is orthogonal to the previous one. The frame allows the gimbals to be held in place and allowed to spin. A sensing system is added to read the angular displacement between two gimbals and change to an electro signal by way of potentiometer, resolvers or encoders. When designing a mechanical gyroscope careful consideration into friction must be considered. The system is meant to be as frictionless as possible. The friction will cause a measurement drift over time. To combat this drift, high precision bearings and special lubricants are used (67).

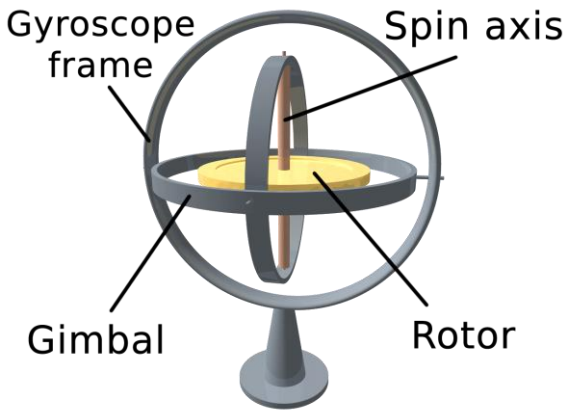


Figure 63: The working parts of a mechanically based gyroscope from (68).

The mechanical gyroscope wasn't chosen based on the considerations and carefulness of manufacturing. The items could be bought pre assembled or individually. The device would

have to be enclosed and mounted in an area where the astronaut would not be bothered. This type of gyroscope would also need a sensor connection to communicate with a computing device or a computing chip inside the unit itself. The device requires more maintenance then some of the other types of gyroscopes.

An optical gyroscope can be split into two types, a fiber-optic gyroscope and a ring laser gyroscope. The fiber optic gyroscope consists of a laser and optical fiber. It is based on the Sagnac effect. The Sagnac effect describes the phenomena where a light beam traveling against the rotation has a shorter optical path length than the other beam. This difference causes a phase shift and an interference pattern. This interference pattern can be used to describe the amount of rotational difference. The fiber can be coiled multiple times to increase the length of the fiber which increases the accuracy of the interferometer and strengthens the Sagnac effect (69).

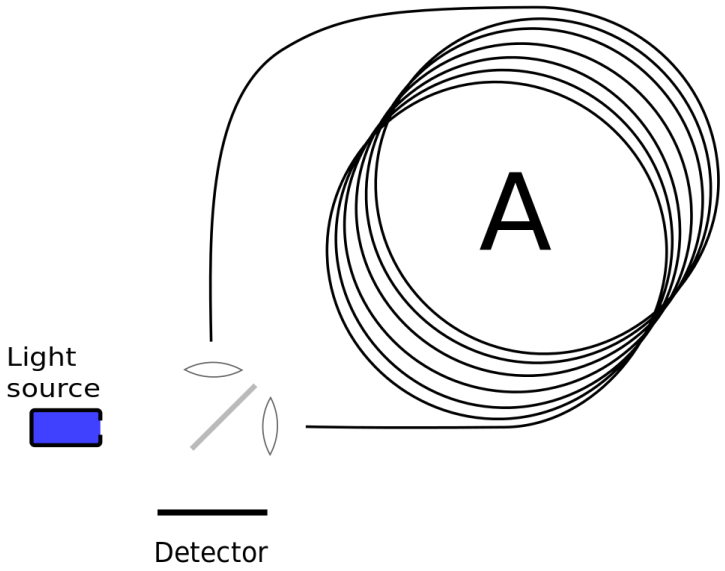


Figure 64 : A simplified design of a fiber optic gyroscope with A being our optical fiber. Taken from (70)

A ring laser gyroscope uses mirrors instead of fiber. It uses the Sagnac effect as well. The laser is split into two beams using a beam splitter and sent in opposite directions. This creates an interferometer. The mirrors are then set up to create directional changes in the beam. Small rotations will increase the path length difference and cause a change in the interference pattern (71).

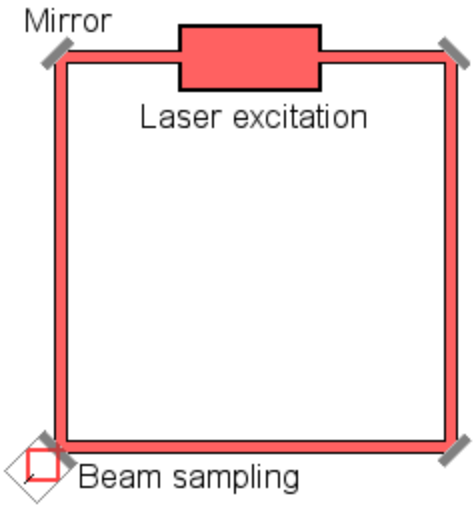


Figure 65: A figure of the working principles behind a ring laser gyroscope. Taken from (72)

The advantages of using an optical gyroscope is that there is no friction in the device for the us to consider, so they do not experience the calculation drift. They generally have a fairly long life span of mean failures times of 60,000. There are no moving parts, so the astronauts would not have to provide maintenance on the device. The choice between using either a ring or a fiber-optic gyroscope depends on our choice of portability compared to sensitivity. The ring laser gyroscope is much more compact and can possibly fit into our additional processing unit's housing. The fiber optic gyroscope can also fit, but it would sacrifice the amount of optical fiber that can be packaged in the housing. The ring laser has some problems with small rotational changes. There is a possibility that with small rotations there will be no detection. Forced dithering can counteract this problem.Creating two different polarizations in the ring laser and then using a polarizer to combine them can also solve this issue.

Another gyroscope in consideration is based on using microelectromechanical systems (MEMS). They are made using nano electrical systems and nanotechnology. It is similar to the vibrating structure gyroscope, also called the Coriolis vibratory gyroscope (73). The principle it uses is that vibrating objects vibrate in the same plane even if the supports rotate. The Coriolis effect exerts a force on the support which can be measured and used to determine the rate of rotation. Many MEMS are developed on integrated circuits with several gyroscopes. Having separate gyroscope sensors allows more dimensions of freedom.

The sensor is designed by having a mass tethered to a frame such that it can only move in one direction. The frame itself is also tethered in a way that is orthogonal to the motion of the mass. This allows the measurement of coriolis acceleration. The change is measured through a capacitance change. It's noted that if we were to design our own, the sensor could be placed anywhere that as long as the sensing axis is parallel to the rotational axis. The capacitance changes are on such a small scale that temperature, stress, external acceleration and noise must

be considered or neutralized. Vibrations from any of the manufacturing defects or from outside sources can also create a large discrepancy in the measurement. To counteract this resonators are placed in different directions equally to compensate for the vibrations (74). These MEMS gyroscopes are small enough and already integratable into our secondary processing device. There are chips that are fairly inexpensive and can be bought easily.

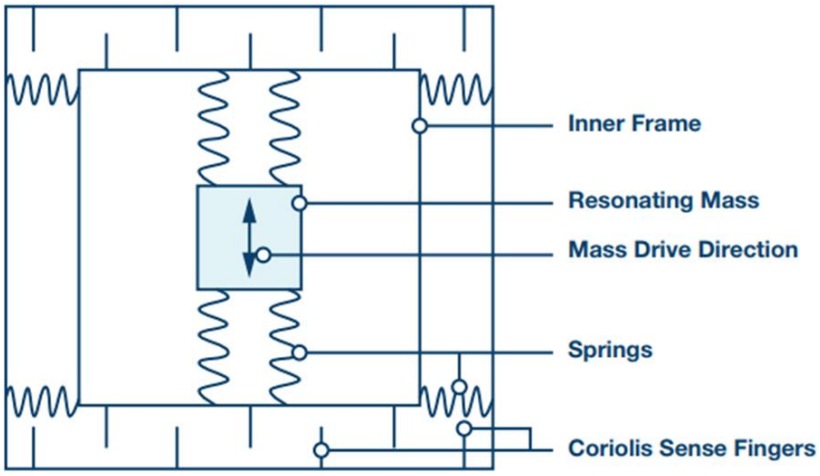


Figure 66: A diagram of the working principles of the Coriolis vibratory gyroscope. Taken from (74).

The gyroscope we plan to use is a fiber-optic gyroscope. The decision was made based on how simple it would be to implement for a proof of concept. The housing can be 3-D printed if necessary or possibly enclosed in the pi or arduino that can be used as a secondary processor. The second choice would be a ring laser gyroscope. The components aren't expensive and are within budget. These two gyroscopes were chosen based on what elements would affect them. The last option in place is the MEMS gyroscope since it would be an out of box option that only possibly needs calibration. This option will be used if time is insufficient in our competition and we need a working gyroscope. The ring laser gyroscope can also be bought and used if necessary. They are small enough that creating a housing for the device would not be difficult. The main issue that we suspect will be where exactly to put the device with consideration to the equipment that is already contained in the spacesuits. This IMS system is most likely a secondary device based on the HMD we've selected. The two HMD already contain an IMS and this secondary one can be used as a backup or secondary data reference.

The laser source was picked to be one at 633 nm. It is commonly used in commercial gyroscopes.The L365P5 from ThorLabs seems to fulfill our parameters of such a device. It is small and fairly inexpensive. The choice between fibers is between a single mode fiber and a multimode fiber.We looked into using a doped fiber to amplify the optical signal, but they are too expensive for our project. The two fibers are not expensive. For the single mode fiber, one option is the SM450 which has a cost per meter of about ~$8 and has a usable wavelength from

450 to 633 nm. The other option is the SM600 which has a usable range from 633 to 780 nm for around ~$6 per meter. The SM600 would most likely be chosen based on its range fitting our laser diode better. Including the beam splitter and mounts, the total cost of the device can be from $60 to around $100. The ring laser gyroscope will cost more to produce due to the cost of buying dielectric mirrors and some type of mounts or 3d printed housings. The MEMS gyroscope chip cost around $5 dollars for the device. It may be more sensible to use the chip considering time and budget constraints. The chip itself can easily attach to any premade board or a printed circuit.

8.2 Stress response sensor

This sensor uses perspiration to check on levels of stress. This can be used as a secondary check to reinforce the heart rate monitor.

8.2.1 Galvanic Skin Response (GSR)

The human nervous system interacts with the environment through a call-response mechanism, in which an external stimuli triggers a biological response. The brain causes this to occur unconsciously by release of cocktails of hormones. These produce signaling cascades within the autonomic nervous system giving rise to physiological changes in the order of milliseconds. Many changes occur physiologically, but for the purpose of monitoring astronaut stress, we will focus on one change in particular *electrodermal response (EDR)* of the autonomic nervous system (75).

When an external stimuli triggers a change in emotional state (emotional arousal), the autonomic nervous system signals for epidermal eccrine glands to release "sweat," a salt solution (0.3% NaCl, 75), which helps thermoregulate the body. A consequence of which is an electrodermal activity referred to as *galvanic skin response (GSR)*.

Note that emotional arousal (and thus GSR) trigger a response regardless of whether the stimuli is positive or negative. Therefore, this proxy measurement can not discern precisely whether the triggering stimuli is "stress (negative emotional arousal)," since not all emotional arousal falls under the category of "stress," but rather that something is triggering the autonomic nervous system, and this could be an environmental stress. Measuring GSR is a measure of the intensity of emotional arousal, some subset of which could be potential stress. Therefore, an aggregate of biometrics are likely necessary, if the goal is to monitor external stimuli or situations triggering "stress" while ignoring all other emotional arousal.

8.2.2 Measuring stress with GSR (GSR sensor)

Emotional arousal due to external stimuli releases sweat, a saline solution, which reduces the resistance of the epidermis to flow of electricity. The increase in sweat from the sweat glands on the skin thus can be used as a proxy for emotional arousal and activation of autonomic nervous system (3). This relationship can be used as a proxy for indirectly measuring emotional arousal (and thus stress) since as the amount of sweat (and thus activation of the autonomic nervous system) increases, the skin resistance falls proportionally.

The inverse relationship of GSR and skin resistance can be used to measure the response, however, it is more commonly measured using conductance, since skin conductance is proportional to changes in GSR (75, 77). Typically, electrodes are attached exosomatically to the hand since the hand contains the highest region of sweat glands per volume at 600 sweat glands per cm^2 (75). The most common electrodes contain Ag/AgCl connection since these are cheap and safe. It is common to attach ionic gel between electrode and skin to increase signal sensitivity (75). The generated signal feeds into a microcontroller or microprocessor and is usually sent wirelessly for analysis and processing (Figure 67).

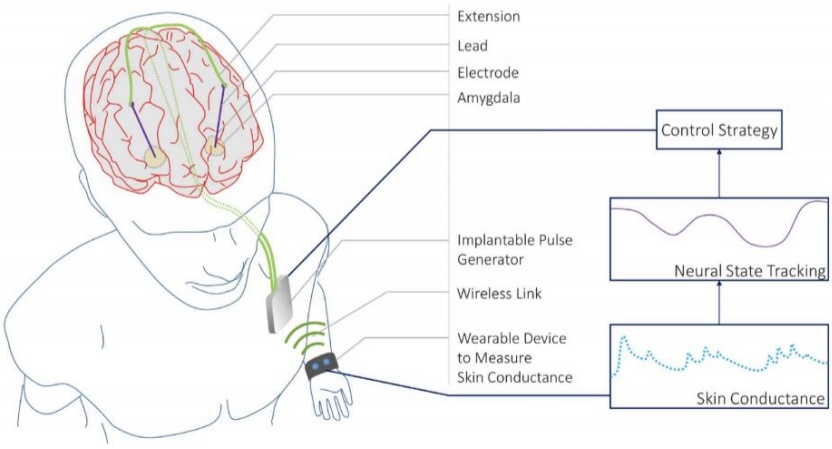


Figure 67: Generalized process for measuring emotional stimulus and wireless transmission *(76)*

The time-varying signal generated by measuring GSR is composed of two additive signals. The first of which is the “tonic” (baseline) signal. It is measured on the order of minutes and ranges from 10-50 uS (microSiemens). The second signal is referred to as “phasic” since it is the phasic, transient component of the overall signal, measured in seconds. This component is the one which varies due to external stimuli as a proxy for emotional arousal (75, 76, 77). Once the body adjusts (or the external stimuli disappears), the sweat dissipates, the conductance goes down, and the phasic component resolves to baseline. See Figure 68 for example signals generated by a standard GSR sensor.

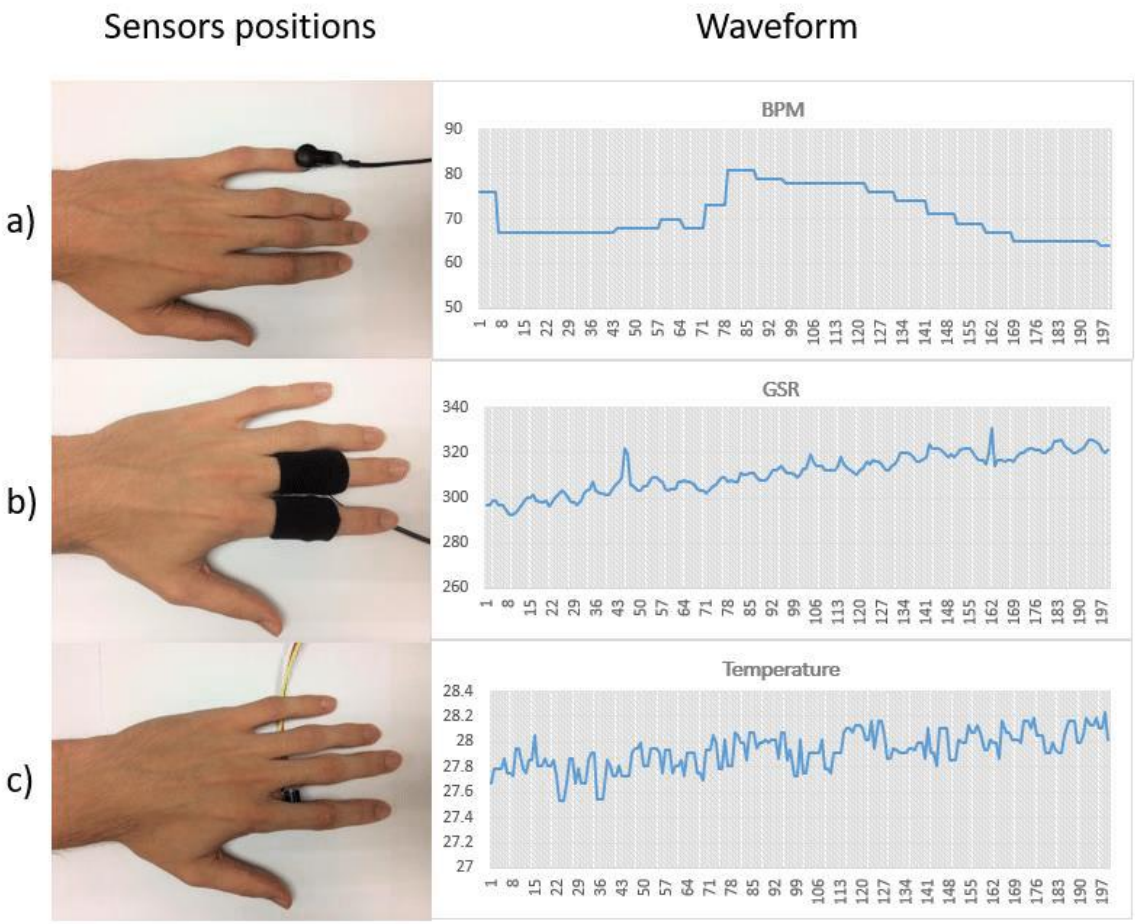


Figure 68: *Analyzing User Emotions via Physiology Signals (76)*

Available GSR sensors and their specifications

Seeed Grove GSR Sensor Module

Seeed Grove GSR sensor module is a cheap, open source sensor for measuring the galvanic skin response. Two electrodes are wrapped in velcro which can be attached to the fingers to obtain a real-time data stream. It operates at either 3.3V or 5V, using I2C communication to interface with the analog in pins on Arduino or Raspberry Pi microcontrollers. ADC resolution for this device is 10 bits and the sensitivity of the device can be adjusted with an onboard potentiometer.

Device Specifications:

- Input voltage: 3.3/5V DC
- Weight: 29g
- Two electrodes with velcro for attachment to fingers.
- Grove Cable (26AWG)
- Adjustable device sensitivity using onboard potentiometer
- Size: approx. 20 x 20mm
- ADC sampling rate：100Ks/s
- ADC accuracy/resolution: 10 bits
- Communication interface type: I2C
- Cost: $19.99

SMAKN Skin Sensor Module

SMAKN Skin Sensor module is a cheap, open source sensor for measuring the galvanic skin response receives a data stream continuously using SPI communication. It operates at 5V. ADC resolution for this device is 12 bits.

Device Specifications:

- Input voltage: 5V DC
- Weight: 28g
- Two electrodes with velcro for attachment to fingers.
- Size: approx. 32.3 x 17.8 x 6.7mm
- ADC sampling rate：100Ks/s
- ADC accuracy/resolution: 12 bits
- Communication interface type: SPI
- Cost: $19.99

8.2.3 GSR sensor design

The previously mentioned circuit modules are potential choices for our design. However, they do come with limitations. If we decide to use a microprocessor that differs from Arduino or Raspberry Pi (Texas Instrument's MSP4300, for example) then we might need to build our own GSR sensor. Also, if we decide to increase the analog-to-digital (ADC) conversion resolution size or use an electrode setup which differs from the one attached to the previously discussed modules, we will need to prototype our own circuit design.

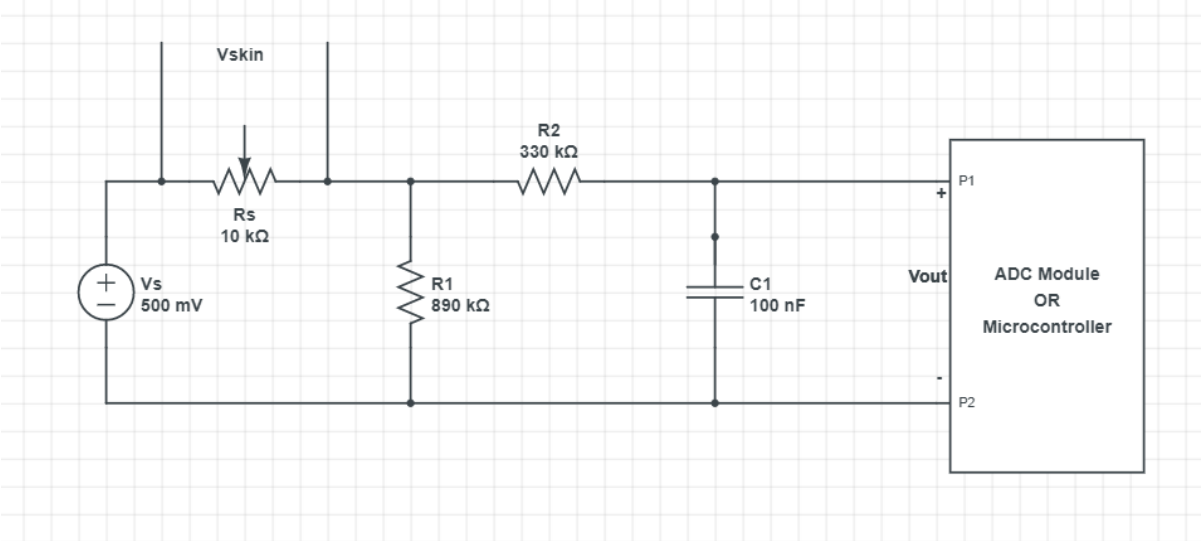


Figure 69: *Schematic for basic GSR sensor*

In general, a basic GSR sensor can be built using a few basic circuit components where the body acts as a variable resistor component in the range from 10kΩ - 10MΩ (77) and the current moving across the resistor is measured as a proxy for skin conductance (Figure 67, V_{skin}). For example, a basic GSR measurement can be obtained with a circuit containing a constant input voltage (0.5-1.5V) in the necessary range to measure GSR (77), electrodes which complete the circuit treating the skin like a variable resistor, and a constant resistor in series with the variable "skin" resistor. To increase the range of the input analog signal, the circuit can feed into an Analog-to-Digital (ADC) conversion IC or microcontroller. Additionally, a passive (or active) low pass filter is often used to smooth the data and reduce noise from high frequencies. Finally, if necessary, an Operational Amplifier can be built into the circuit to amplify the signal as well as filter noise from unwanted frequencies using high and low pass filtering.

8.3 Solar Radiation sensor

Measuring solar radiation would be very useful for the astronauts as well as mission control. There are two types of measurement devices for solar radiation: pyranometers and pyrheliometers. Pyrheliometers measure direct beam radiation from the sun and a small portion of the sky around the sun. Pyranometers are used to measure the total solar radiation, including any diffuse radiation. (82) While pyrheliometers would be our first choice, the cost and unavailability of the devices lead us to the pyranometers. However, because there is no atmosphere on the moon, there will be very minimal diffuse radiation allowing us to measure the direct beam radiation from the us. In effect, the pyranometer becomes a less sensitive pyrheliometer.

The sensor chosen for measuring solar radiation is the Davis 6450 Solar Radiation Sensor. The following table details the relevant specifications of the sensor:

Operating Temperature	-40° to +150° F (-40° to +65° C)
Transducer	Silicon photodiode
Spectral Response	400 to 1100 nanometers
Supplied Cable Length	2' (0.6 m)
Connector	Modular RJ-11
Housing Material	UV-resistant PVC plastic
Dimensions (Length x Width x Height)	2.00” x 2.75” x 2.25” (51 mm x 70 mm x 57 mm)
Weight	0.5 lbs. (226 g)
Cost	$167

Table 11: Relevant specifications for the Davis 6450 Solar Radiation Sensor.

8.4 Accelerometer

A secondary accelerometer may be developed to create an INS. This will work together with the gyroscope to give a distance and a direction to allow a sort of local navigation. This can become either the primary INS or a secondary one used for accuracy.

Introduction to accelerometers

An accelerometer is a device that measures the vibration or accelerations. When an object moves, there always is a force caused by vibration or acceleration (a change in motion), and the force will squeeze the piezoelectric material contained in the accelerometer sensor, conversely, it produces an electrical charge proportional to the force on it. In this situation, the charge is proportional to the force, and the mass is a constant, then the charge will be proportional to the acceleration, then the sensor will get some values to calculate the acceleration.

Accelerometer Types

Commonly, accelerometers can be divided into three types, compression mode accelerometers, shear mode accelerometers and capacitive accelerometers.

For compression mode accelerometers, it’s composed of a sensing element crystal , which

emits charge when subjected to compression force. The crystal is bonded to the mass of an internal induction element. This is wrapped in a suitable body--usually stainless steel, it can make the sensor have a strong meaning, it can withstand harsh environments. Then the sensor has an electrical connection, which can be a sealed cable or plug socket.

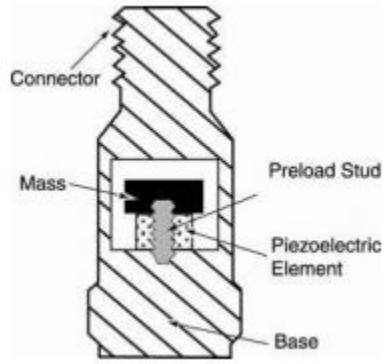


Figure 70: Schematic of compression mode accelerometer

For shear mode accelerometer, it also uses sensing crystals, which are connected between the central column and the seismic mass. Under the action of acceleration , mass will cause shear stress acting on the sensing crystal. Using piezoelectric sensors , applied stress on the crystal will produce proportional electrical output. The output is then transmitted to the built-in sensor signal conditioning circuit or directly to the electrical connector for charging the model type. Shear mode accelerometers use sensing crystals isolated from the base and shell, which means they perform very well in rejecting thermal transient and base bending effects. Their sizes are also smaller, which promotes the high-frequency response.

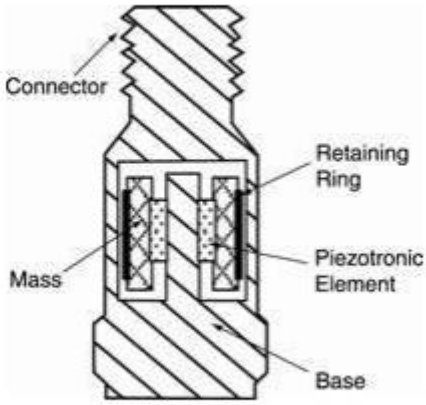


Figure 71: Schematic of shear mode accelerometer

For Capacitive Accelerometers, the sensor inside will perceive the change of capacitance with acceleration. The sensitive element of capacitive accelerometer is composed of two parallel plate capacitors, acting on the differential mode. These parallel capacitors operate in a bridge circuit with two fixed capacitors to change the peak voltage generated by the oscillator of the

sensor under acceleration. The Detection circuit will send the captured peak voltage into a summation amplifier to process the final output signal.

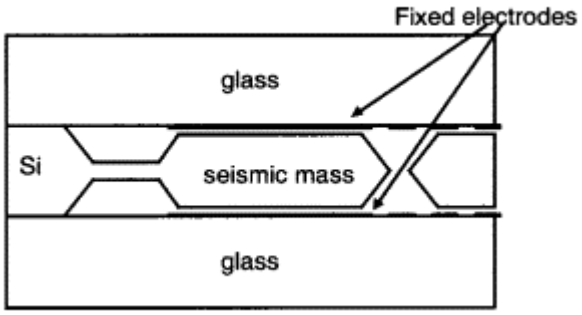


Figure 72: Schematic of capacitive accelerometers

Accelerometer Specifications

In order to have a sense on choosing accelerometers, it's necessary to know the specifications of the accelerometer, and there are a bunch of core specifications that need to be considered.

Measurement Range:
The acceleration level supported by the sensor output signal specification is generally expressed as ± g . This is the maximum acceleration that the part can measure and is accurately represented as an output. For example , the output of a ± 3 g accelerometer is linearly related to the acceleration up to ± 3 g. If accelerated at 4g, the output may be read as fail . Also, it's necessary to know the breakpoint is specified by the absolute maximum acceleration , not by the measurement range. A 4g acceleration won't break a ±3g accelerometer.

Sensitivity:
Sensitivity is the ratio of change in input to change in output signal, the input here defines acceleration. This defines the ideal linear relationship between acceleration and output (example in Figure 71, gray line). Sensitivity is specified under a specific power supply voltage, which is usually represented by the unit mV/g for the analog output accelerometer, and the unit LSB/g or mg/LSB for the digital output accelerometer. It is usually specified in a range (min, type, max) or as a typical graph and % deviation. For analog output sensors, the sensitivity is proportional to the supply voltage, for example, when the supply voltage doubles, the sensitivity doubles.

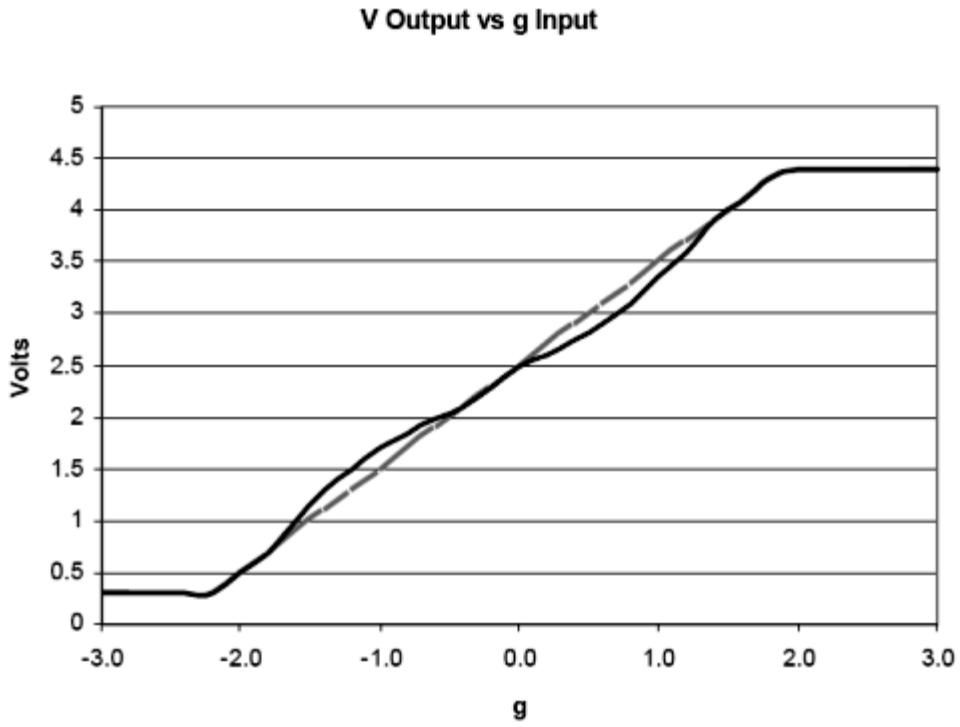


Figure 73: V output VS g Input

Temperature:
Temperature changes will influence the sensitivity, so there is a specification called temperature sensitivity, which is the voltage output per degree of the measured temperature. Also, temperature compensation is applied to the sensor to keep the output change within the specified range. Normally, a temperature range is -50℃ to 120℃ , and it's specified by the electronic microcircuit that can convert charge to low impedance output.

Available Accelerometer Consideration

ADXL377 Accelerometer

The ADXL377 is a small, low power, thin, 3-axis accelerometer with signal conditioned voltage outputs, it can measure the acceleration from motion, shock and vibration with a wide range of ±200 g. It is also user friendly, the user can choose the bandwidth of the accelerometer using the CX, CY, and CZ capacitors at the XOUT, YOUT, and ZOUT pins. With a range from 0.5 Hz to 1300 Hz in x and y axis, from 0.5 Hz to 1000 Hz in z axis.

Device specifications:

- 3-axis sensing
- ±200 g measuring range
- 3 mm × 3 mm × 1.45 mm LFCSP, small low profile package
- low power: 300 μA (typical)
- Single-supply operation: 1.8 V to 3.6 V

- Excellent temperature stability
- 10000 g high-shock survivability
- cost: $25.95

H3LIS331DL Triple Axis Accelerometer

The SparkFun H3LIS331DL triple axis accelerometer is a small, low power high-g accelerometer with I2C and SPI interface options. The module is good for grueling conditions that normal accelerometers can't suited for, it offers a wide adjustable output range of 100, 200, and 400 g, and an adjustable data rate from 0.5 Khz to 1 KHz. It can operate between 2.16V to 3.6V. And there is a low-power mode with low-voltage compatible IOs at 1.8V with a power usage of 10μA.

Device Specifications:

- wide supply voltage, 2.16V to 3.6V
- Low-voltage compatible IOs, 1.8V
- Ultra-low power consumption down to 10μA in low-power mode
- ±100g/±200g/±400g dynamically selectable full scales
- I2C & SPI digital output interface
- 10000g high-shock survivability
- cost: $11.95

9. Administrative Content

9.1 Milestones Discussion

This section details our schedule and deadlines through the duration of the project. Each of these milestones is vital in the completion of our design. As we acquire more information about challenge deadlines in the spring, we will be updating Table 12.

Number	Task	Start Date	End Date	Team Lead	Status
1	NASA SUITS Letter of Intent	Sept 8th	Sept 30th	Teo	Completed
2	NASA SUITS Proposal	Oct 1st	Oct 20th	Teo	Completed
3	Parts List	Sept 8th	Oct 20th	Yongsheng	Completed
4	Funding Period	Sept 20th	Jan 15th	Teo	In Progress
5	Order Parts	Oct 21st	Jan 15th	Sammy	In Progress
6	Final Documentation	Nov 14th	Nov 27th	David	In Progress
7	External Sensor Communication w/ HoloLens II	Jan 10th	Jan 31st	Yongsheng	Not Started
8	Controls Testing	Jan 10th	Jan 31st	Sammy	Not Started
9	User Interface Graphical Design	Jan 10th	Feb 7th	David	Not Started
10	Navigation Interface Development	Jan 10th	Feb 21st	David	Not Started
11	Task Interface Development	Feb 1st	Feb 21st	Teo	Not Started
12	External Sensor Development	Jan 10th	Mar 15th	Sammy/ Yongsheng	Not Started
13	Experimentation and Testing	Feb 1st	Mar 22nd	Teo	Not Started
14	Ship Device to Houston (if selected)		April	All	Not Started
15	Virtual Test Week	April	April	All	Not Started

Table 12: Project Schedule

9.2 Budget and Finances

Table 13 details the budget associated with our project. This is subject to change upon testing and experimentation. We have been partially funded by the George Jackson Foundation. We are also looking at multiple other funding sources, which include the NASA Florida Space Grant Consortium, private companies, and grant foundations. We are also prepared to self-fund a portion of this project between our team members.

Budget and Funding			
Item	Price/per unit	Quantity	Price/total
Microsoft Hololens II	$ 3,500.00	1	$ 3,500.00
PCB Board	$ 20.00	5	$ 100.00
Photosensor	$ 10.00	1	$ 10.00
Microcontroller	$ 30.00	1	$ 30.00
Temperature Sensor	$ 11.00	1	$ 11.00
Push Button	$ 1.00	10	$ 10.00
Wireless Communication Module	$ 9.00	1	$ 9.00
Pressure Sensor	$ 9.00	1	$ 9.00
IR Camera	$ 50.00	1	$ 50.00
LED Source	$ 5.00	2	$ 10.00
Wiring	$ 0.50	20	$ 10.00
Housing for External Sensors	$ 15.00	1	$ 15.00
Wristwatch Housing	$ 15.00	2	$ 30.00
LED Detector	$ 5.00	2	$ 10.00
Total (Estimated Range)			$ 3,804.00

Table 13: Project Budget

9.3 Statement of Rights

October 15, 2020

Dear NASA SUITS Program Administrator,

Whereas, David Hagan, Ph.D., Interim Dean and Professor, The College of Optics & Photonics, University of Central Florida (UCF) is designated as Faculty Advisor on a proposal entitled "Comprehensive External Sensor and Internal User Interface Design for the NASA SUITS Challenge" (Proposal) proposed by a team of higher education students from the UCF;

Whereas, in accordance with federal law, Dr. Hagan, as a UCF employee, has assigned rights to any intellectual property related to his field of employment or for which University Support was used to the University of Central Florida Research Foundation, Inc. (UCFRF), a direct support organization of the UCF, organized under the laws of the State of Florida, United States;

Now therefore, in support of the Proposal, UCFRF will and hereby does grant the U.S. Government a royalty-free, nonexclusive and irrevocable license to use, reproduce, distribute (including distribution by transmission) to the public, perform publicly, prepare derivative works, and display publicly, any technical data contributed by Dr. Hagan and contained in this proposal in whole or in part and in any manner for Federal purposes and to have or permit others to do so for Federal purposes only. Further, with respect to all computer software designated by NASA to be released as open source which is first produced or delivered under this proposal and subsequent collaboration, if selected, shall be delivered with unlimited and unrestricted rights, as between UCFRF and NASA, so as to permit further distribution as open source. For purposes of defining the rights in such computer software, "computer software", shall include source codes, object codes, executables, ancillary files, and any and all documentation related to any computer program or similar set of instructions delivered in association with this collaboration. In support of the proposal entitled "Comprehensive External Sensor and Internal User Interface Design for

the NASA SUITS Challenge" proposed by a team of higher education students from the University of Central Florida, UCFRF will and herby does grants the U.S. Government a nonexclusive, nontransferable, irrevocable, paid-up license to practice or have practiced for or

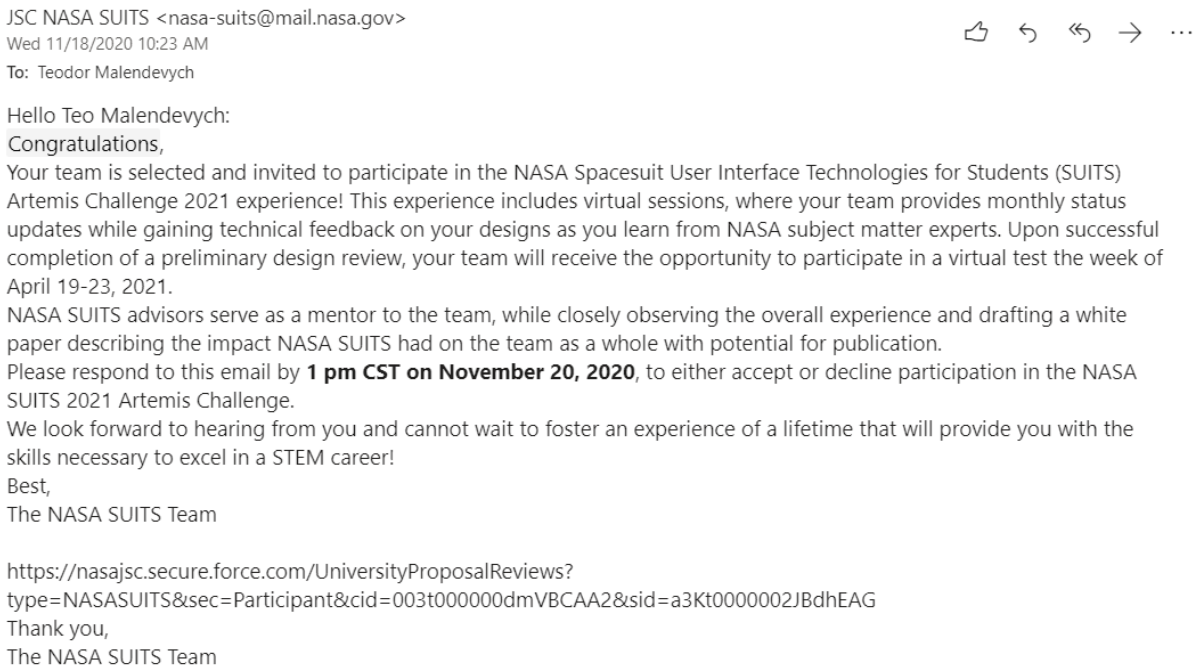
on behalf of the United Stated Government any invention developed by Dr. Hagan and described or made part of this proposal throughout the world.

Sincerely,

Digitally signed by Svetlana Shtrom
Date: 2020.10.15 08:51:59 -04'00'
Svetlana Shtrom, Ph.D.
Vice President of Technology Transfer
University of Central Florida Research Foundation, Inc.

9.4 Acceptance email from NASA SUITS Challenge

The following details the acceptance of our proposal for the NASA SUITS Challenge. We attended the selection ceremony and will be attending the virtual talk on 12/8/2020.



JSC NASA SUITS <nasa-suits@mail.nasa.gov>
Wed 11/18/2020 10:23 AM
To: Teodor Malendevych

Hello Teo Malendevych:
Congratulations,
Your team is selected and invited to participate in the NASA Spacesuit User Interface Technologies for Students (SUITS) Artemis Challenge 2021 experience! This experience includes virtual sessions, where your team provides monthly status updates while gaining technical feedback on your designs as you learn from NASA subject matter experts. Upon successful completion of a preliminary design review, your team will receive the opportunity to participate in a virtual test the week of April 19-23, 2021.
NASA SUITS advisors serve as a mentor to the team, while closely observing the overall experience and drafting a white paper describing the impact NASA SUITS had on the team as a whole with potential for publication.
Please respond to this email by **1 pm CST on November 20, 2020**, to either accept or decline participation in the NASA SUITS 2021 Artemis Challenge.
We look forward to hearing from you and cannot wait to foster an experience of a lifetime that will provide you with the skills necessary to excel in a STEM career!
Best,
The NASA SUITS Team

https://nasajsc.secure.force.com/UniversityProposalReviews?type=NASASUITS&sec=Participant&cid=003t000000dmVBCAA2&sid=a3Kt0000002JBdhEAG
Thank you,
The NASA SUITS Team

Figure 74: Email from nasa-suits@mail.nasa.gov

Appendix A. Technical References

1. Artemis. (11/05/2020). Retrieved November 13, 2020, from https://www.nasa.gov/specials/artemis/
2. Mahoney, E. (2019, October 04). Spacesuit Basics. Retrieved November 13, 2020, from https://www.nasa.gov/feature/spacewalk-spacesuit-basics
3. Mahoney, E. (2019, October 04). A Next Generation Spacesuit for the Artemis Generation of Astronauts. Retrieved November 13, 2020, from https://www.nasa.gov/feature/a-next-generation-spacesuit-for-the-artemis-generation-of-astronauts
4. Apollo Lunar Surface Journal. (11/05/2020). Retrieved November 13, 2020, from https://www.hq.nasa.gov/alsj/
5. Diaz, J. (2019, November 04). From Apple Glasses to Hololens 2: AR glasses you can buy now (and soon). Retrieved November 13, 2020, from https://www.tomsguide.com/reference/ar-glasses
6. Magic Leap 1. (11/05/2020). Retrieved November 13, 2020, from https://www.magicleap.com/en-us/magic-leap-1
7. Rousseau, S., & Steve Rousseau is the Features Editor at Digg. . (11/05/2020). It's Okay That Magic Leap Costs $2,300 And Isn't Mind-Blowingly Incredible. Retrieved November 13, 2020, from https://digg.com/2018/magic-leap-one-creators-edition
8. HoloLens 2-Overview, Features, and Specs: Microsoft HoloLens. (11/05/2020). Retrieved November 13, 2020, from https://www.microsoft.com/en-us/hololens/hardware
9. Feltham, J., Broadwell, J., & Baker, H. (2020, August 17). Magic Leap One Creator Edition Tech Specs Revealed. Retrieved November 13, 2020, from https://uploadvr.com/magic-leap-one-creator-edition-tech-specs-revealed/
10. Developer Portal: Magic Leap. (11/05/2020). Retrieved November 13, 2020, from https://developer.magicleap.com/en-us/learn/guides/design-why-magic-leap
11. BT-2000 - Technical Information - MOVERIO Pro. (11/05/2020). Retrieved November 13, 2020, from https://tech.moverio.epson.com/en/bt-2000/
12. Moverio Pro BT-2200 Smart Headset. (11/05/2020). Retrieved November 13, 2020, from https://epson.com/For-Work/Wearables/Smart-Glasses/Moverio-Pro-BT-2200-Smart-Headset/p/V11H853020
13. Tech Specs – Glass. (11/05/2020). Retrieved November 13, 2020, from https://www.google.com/glass/tech-specs/
14. Biswas, B. (2019, May 31). What Is Augmented Reality. Retrieved November 13, 2020, from https://towardsdatascience.com/fundamentals-of-augmented-reality-ar-89ce986f70f8
15. What is Virtual Reality? (11/05/2020). Retrieved November 13, 2020, from https://www.vrs.org.uk/virtual-reality/what-is-virtual-reality.html

16. BrandonBray. (11/05/2020). What is Mixed Reality? - Mixed Reality. Retrieved November 13, 2020, from https://docs.microsoft.com/en-us/windows/mixed-reality/discover/mixed-reality
17. Cheriyedath, S. (2019, February 27). Photoplethysmography (PPG). Retrieved November 13, 2020, from https://www.news-medical.net/health/Photoplethysmography-(PPG).aspx
18. https://www.osapublishing.org/DirectPDFAccess/E504C5A5-7C41-4819-A72D991D8A4ACBA0_333549/ao-54-35-10559.pdf?da=1&id=333549&seq=0&mobile=no
19. Stubán, N., & Masatsugu, N. (2008). Non-invasive calibration method for pulse oximeters. Periodica Polytechnica Electrical Engineering, 52(1-2), 91. doi:10.3311/pp.ee.2008-1-2.11.
20. Woodford, C. (2020, August 19). How does night vision work? Retrieved November 13, 2020, from https://www.explainthatstuff.com/hownightvisionworks.html
21. Woodford, Chris. (2007/2020) Night vision goggles. Retrieved from https://www.explainthatstuff.com/hownightvisionworks.html. [Accessed (11/05/2020)]
22. Carolyn Haley → Jul 13, John Snell → Jul 21, Jeicyn Heierbuernathee → Oct 10, Terry a witt → Dec 19, Kyle → Jan 08, & Virginia → Apr 18. (11/05/2020). Night Vision: How Animals See in the Dark: The Outside Story. Retrieved November 13, 2020, from https://northernwoodlands.org/outside_story/article/night-vision-how-animals-see
23. O'Connor, Jim.(February 28, 2020) Dark Adaptation of the Human Eye and the Value of Red Flashlights.Retrieved from https://www.nps.gov/articles/dark-adaptation-of-the-human-eye-and-the-value-of-red-flashlights.htm. [Accessed (11/5/2020)]
24. Paschotta,Rüdiger (October 2008) photocathodes. Retrieved fromhttps://www.rp-photonics.com/encyclopedia_cite.html?article=photocathodes[Accessed (11/5/2020)]
25. Abramowitz, Mortimer, Michael W. Davidson. (July 12, 2016) Photomultiplier Tubes.Retrieved from https://micro.magnet.fsu.edu/primer/digitalimaging/concepts/photomultipliers.html [Accessed (11/5/2020)]
26. Albert W. Hull, E. F. Hennelly and F. R. Elder, The Dynatron Detector -- a new heterodyne receiver for continuous and modulated waves, Proceedings of the Institute of Radio Engineers Vol. 10, No. 5 (Oct. 1922), pages 320-343
27. S. O. Flyckt and C. Marmonier, "Photomultiplier tubes: principles and applications", Philips Photonics, Brive, France (2002), http://wwwpv.infn.it/~debari/doc/Flyckt_Marmonier.pdf
28. Hamamatsu, "Photomultiplier tubes, basics and applications", https://www.hamamatsu.com/resources/pdf/etd/PMT_handbook_v3aE.pdf

29. Image Intensifiers. Retrieved from https://stanfordcomputeroptics.com/technology/image-intensifier/phosphor-screen.html [Accessed (11/5/2020)]
30. File:PhotoMultiplierTubeAndScintillator.svg. (n.d.). Retrieved November 13, 2020, from https://commons.wikimedia.org/wiki/File:PhotoMultiplierTubeAndScintillator.svg
31. Reina Wang, N. (2018, April 04). #2: Blackbody Radiation and the Photoelectric Effect. Retrieved November 13, 2020, from https://thisgirlreina.wordpress.com/2018/03/22/blackbody-radiation-and-the-photoelectric-effect/
32. Raman Maini, Himanshu Aggarwhal . A Comprehensive Review of Image Enhancement Techniques. Journal of Computing. Vol 2. Issue 3. March 2010. a from https://arxiv.org/ftp/arxiv/papers/1003/1003.4053.pdf. Retrieved November 13,2020
33. Bhabatosh Chanda , Dwijest Dutta Majumder.2002. *Digital Image Processing and Analysis*.
34. Rafael C. Gonzalez, Richard E. Woods. 2007. *Digital Image Processing, 3rd Edition*. Pearson.
35. A.C. Bovik, Digital Image Processing Course Notes, Dept. of Electrical Engineering, U. of Texas at Austin, 1995. Retrieved November 13,2020
36. Qadri Hamarsheh. Digital Image Processing Course Notes, Philadelphia University Jordan. From https://www.philadelphia.edu.jo/academics/qhamarsheh/uploads/Lecture_11_Histogram_Processing.pdf . Retrieved November 13,2020.
37. Ali Pourramezan Fard, Histogram Matching, Towards Data Science. From https://towardsdatascience.com/histogram-matching-ee3a67b4cbc1. Retrieved November 13, 2020
38. S. E. Umbaugh. 1998. *Computer Vision & Image Processing*. Prentice Hall PTR.
39. Sayed, I. S., & Ismail, S. S. (2020, April 1). *Comparison of Low-Pass Filters for SPECT Imaging*. International Journal of Biomedical Imaging. https://www.hindawi.com/journals/ijbi/2020/9239753/.
40. Jesus, J. Caban, 2010. Introduction to Data Science Course Notes University of Maryland, Baltimore County Retrieved November 05, 2020
41. Electrical4U. (2020, October 11). Butterworth Filter: What is it? (Design & Applications). Retrieved November 13, 2020, from https://www.electrical4u.com/butterworth-filter/
42. High Pass vs Low Pass Filters. (2020). Retrieved November 13, 2020, from https://www.tutorialspoint.com/dip/high_pass_vs_low_pass_filters.htm
43. B. P. Lathi, Signal Processing and Linear Systems. Crc Press. 2000.
44. Maharishi, M. (2014). “Image Sharpening By Gaussian And Butterworth High Pass Filter. November 13, 2020, http://dx.doi.org/10.13005/bpj/545

45. Electrical4U. (2020, October 11). Band Pass Filter: What is it? (Circuit, Design & Transfer Function). Retrieved November 13, 2020, from https://www.electrical4u.com/band-pass-filter/
46. TANENBAUM, A. S. (2019). *COMPUTER NETWORKS*. PEARSON.
47. Wireless communication: Intro, types, and applications. (2017, June 8). Retrieved November 13, 2020, from https://www.electronicshub.org/wireless-communication-introduction-types-applications/
48. Baird, D. (2020, October 05). Space Communications: 7 Things You Need to Know. Retrieved November 13, 2020, from https://www.nasa.gov/feature/goddard/2020/space-communications-7-things-you-need-to-know
49. Davies, J. (2013). MSP430 microcontroller basics. Retrieved November 13, 2020, from https://www.amazon.com/MSP430-Microcontroller-Basics-John-Davies/dp/0750682760
50. Barron, A., & Plath, J. (2017, December 01). The evolution of honey bee dance communication: A mechanistic perspective. Retrieved November 13, 2020, from https://jeb.biologists.org/content/220/23/4339
51. 3308610866. (2019, July 18). WiFi VS Bluetooth VS ZigBee:How to choose the appropriate connection protocol?: GearBest Blog. Retrieved November 13, 2020, from https://www.gearbest.com/blog/how-to/wifi-vs-bluetooth-vs-zigbeehow-to-choose-the-appropriate-connection-protocol-6905
52. Zigbee vs. Bluetooth: Choosing the Right Protocol for Your IoT Application. (2020, March 5). Retrieved November 13, 2020, from https://www.digi.com/blog/post/zigbee-vs-bluetooth-choosing-the-right-protocol
53. RF Wireless Technology. (2020). Retrieved November 13, 2020, from https://www.mouser.com/applications/rf-wireless-technology/
54. Admin. (2019, December 09). Different Types of Wireless Communication Technologies. Retrieved November 13, 2020, from https://www.typesnuses.com/different-types-wireless-communication-technologies/
55. 433MHz RF Sniffer. (2019, October 02). Retrieved November 13, 2020, from https://www.electroschematics.com/433mhz-rf-sniffer/
56. Ray, B. (2015, August 24). ZigBee Vs. Bluetooth: A Use Case With Range Calculations. Retrieved November 13, 2020, from https://www.link-labs.com/blog/zigbee-vs-bluetooth
57. MKS075Check out this Author's contributed articles., MKS075, & Check out this Author's contributed articles. (2019, November 18). Difference between Bluetooth and Zigbee. Retrieved November 13, 2020, from https://www.geeksforgeeks.org/difference-between-bluetooth-and-zigbee/

58. Beyer, R. A., Alexandrov, O., & Mcmichael, S. (2018). The Ames Stereo Pipeline: NASA's Open Source Software for Deriving and Processing Terrain Data. *Earth and Space Science, 5*(9), 537-548. doi:10.1029/2018ea000409
59. Motiv, Teresa. *Use your voice to operate HoloLens*. Microsoft. Retrieved October 13, 2020 https://docs.microsoft.com/en-us/HoloLens/HoloLens-cortana
60. Mamaylya. *HoloLens2 gestures for authoring and navigating in Dynamics 365 guide*. Microsoft. Retrieved October 13, 2020. https://docs.microsoft.com/en-us/dynamics365/mixed-reality/guides/authoring-gestures-hl2
61. DS18S20+ - Temperature Sensor IC, Open Drain, ± 0.5°C, -55 °C, 125 °C, TO-92, 3 Pins. (n.d.). Retrieved December 08, 2020, from https://www.newark.com/maxim-integrated-products/ds18s20/digital-thermometer-0-5-deg-c/dp/81Y9062
62. MSP430FR6989,TI. (2020). TI. https://www.ti.com/lit/ug/slau627a/slau627a.pdf?ts=1604816940404&ref_url=https%253A%252F%252Fwww.ti.com%252Ftool%252FMSP-EXP430FR6989
63. Raspberry Pi Control Module 4. (2020). Raspberry Pi. https://datasheets.raspberrypi.org/cm4/cm4-product-brief.pdf
64. Arduino Uno. (2020). Arduino. https://store.arduino.cc/usa/arduino-uno-rev3
65. IPC-2221 Standard. (2020). IPC Org. https://www.ipc.org/TOC/IPC-2221A.pdf
66. IEEE Announces Virtual Reality (VR) and Augmented Reality (AR) Standards Projects In Advance of Participation at Augmented World Expo. (n.d.). Retrieved December 08, 2020, from https://standards.ieee.org/news/2017/ieee_p2408.html
67. Inertial navigation systems analysis, Kenneth R. Britting, Wiley-Interscience, 1971.
68. Vieira, Lucas, "A 3D gyroscope rendered in POV Ray." October 4, 2006.
69. Lefèvre, Hervé (1993). *The Fiber-Optic Gyroscope*. ARTECH HOUSE, INC. ISBN 0-89006-537-3.
70. D. Mcfadden . "Fibre-optic-interferometer." February 10, 2010.
71. *Knowing Machines*, Donald MacKenzie, The MIT Press, (1991)
72. Cleonis. "Ring Laser Interferometer." August 19th, 2006.
73. IEEE Std 1431–2004 Coriolis Vibratory Gyroscopes.
74. Watson, Jeff. "MEMS Gyroscope Provides Precision Inertial Sensing in Harsh, High Temperature Environments." Retrieved from MEMS Gyroscope Provides Precision Inertial Sensing in Harsh, High Temperature Environments | Analog Devices Accessed 12/6/2020.
75. 27.1 INTRODUCTION. (n.d.). Retrieved December 07, 2020, from http://www.bem.fi/book/27/27.htm
76. How does a GSR sensor work? (2018, January 18). Retrieved December 07, 2020, from https://www.tobiipro.com/learn-and-support/learn/GSR-essentials/how-does-a-gsr-sensor-work/

77. Villarejo, M. V., Zapirain, B. G., & Zorrilla, A. M. (2012). A Stress Sensor Based on Galvanic Skin Response (GSR) Controlled by ZigBee. *Sensors, 12*(5), 6075-6101. doi:10.3390/s120506075
78. Wickramasuriya, D. S., Amin, M. R., & Faghih, R. T. (2019). Skin Conductance as a Viable Alternative for Closing the Deep Brain Stimulation Loop in Neuropsychiatric Disorders. *Frontiers in Neuroscience, 13*. doi:10.3389/fnins.2019.00780
79. Dream Glass 4K/4K Plus. (n.d.). Retrieved December 08, 2020, from https://www.dreamworldvision.com/dream-glass-4k-4k-plus
80. K. Zuiderveld: *Contrast Limited Adaptive Histogram Equalization*. In: P. Heckbert: *Graphics Gems IV*, Academic Press 1994, ISBN 0-12-336155-9
81. Pisano, Etta & Cole, Elodia & Hemminger, Bradley & Yaffe, Martin & Aylward, Stephen & Maidment, Andrew & Johnston, R. & Williams, Mark & Niklason, Loren & Conant, Emily & Fajardo, Laurie & Kopans, Daniel & Brown, Marylee & Pizer, Stephen. (2000). Image Processing Algorithms for Digital Mammography: A Pictorial Essay1. Radiographics : a review publication of the Radiological Society of North America, Inc. 20. 1479-91. 10.1148/radiographics.20.5.g00se311479.
82. 2.1 Available Solar Radiation and How It Is Measured. (n.d.). Retrieved December 08, 2020, from https://www.e-education.psu.edu/eme812/node/644
83. Machine Design. (2011, February 1). Retrieved December 07, 2020, from https://www.machinedesign.com/automation-iiot/sensors/article/21828324/introduction-to-accelerometers
84. Accelerometer Specifications - Quick Definitions. (n.d.). Retrieved December 07, 2020, from https://www.analog.com/en/products/landing-pages/001/accelerometer-specifications-definitions.html
85. ADXL377. (n.d.). Retrieved December 07, 2020, from https://www.analog.com/en/products/adxl377.html
86. SparkFun Triple Axis Accelerometer Breakout - H3LIS331DL. (n.d.). Retrieved December 07, 2020, from https://www.sparkfun.com/products/14480

Appendix B - Supplemental Figures & Tables

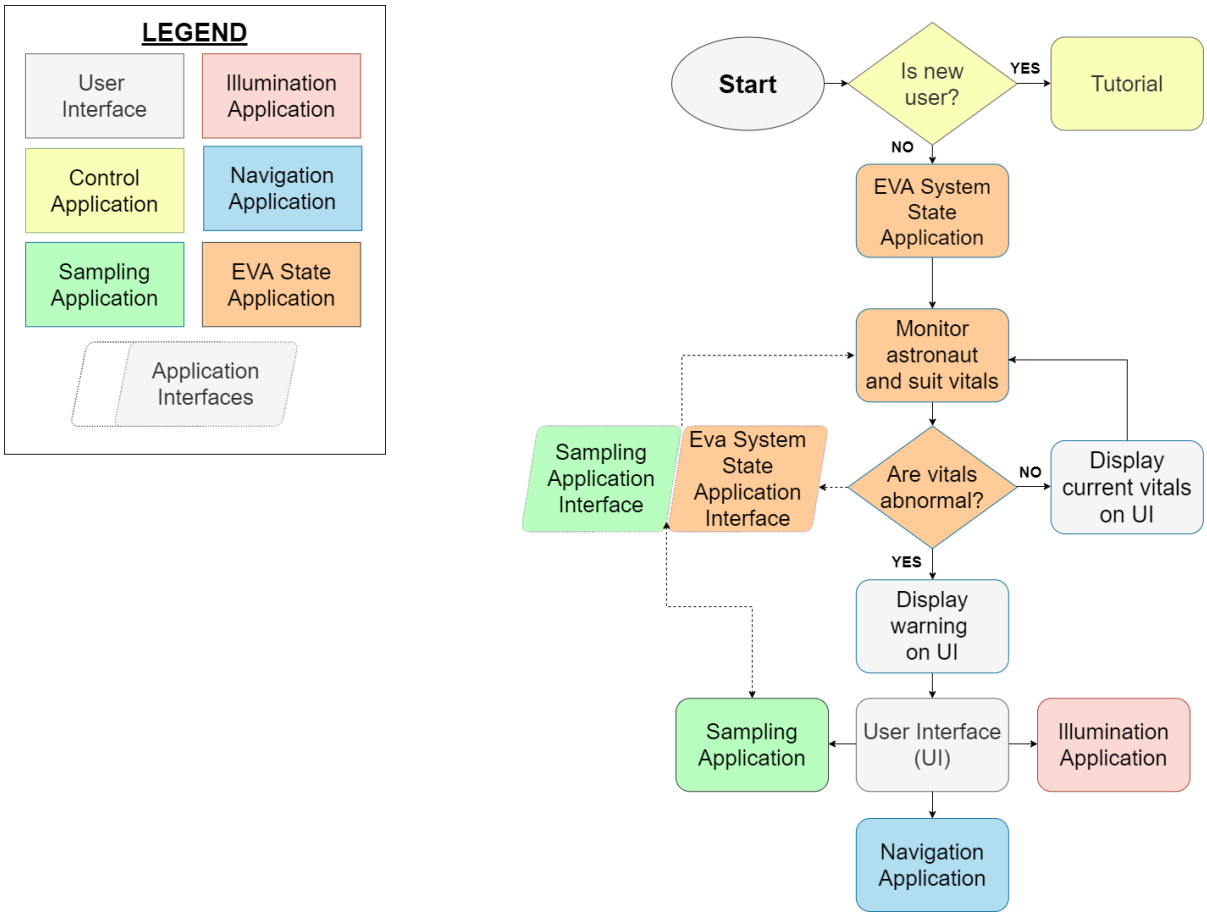


Figure A1: Software Design Diagram

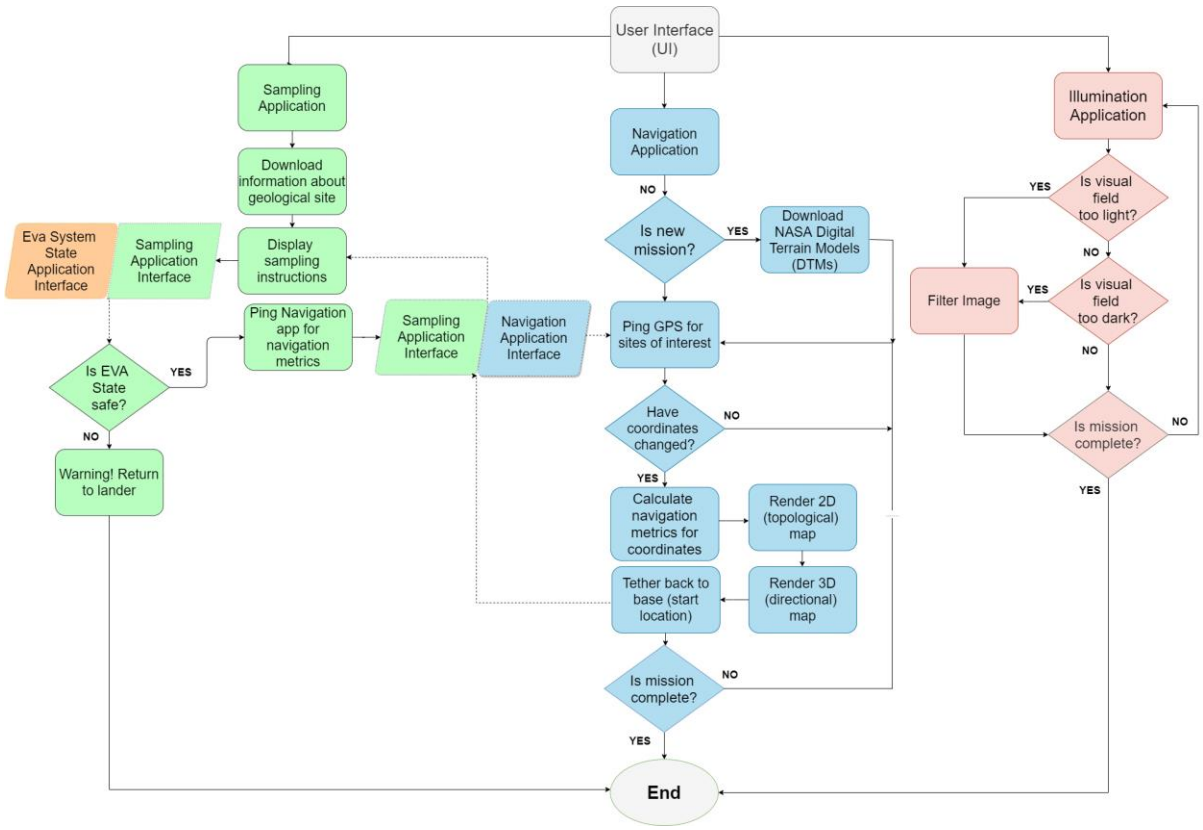


Figure A2: Software Design Diagram focusing on Sampling, Navigation, and Illumination Applications