**System Design Document**

**For**

**Low-Cost Laser Communications**

Team members: Sean Huber, Dylan Koch, Cameron Martinez, Dominic Steiner, Matthew Simms, Jarrod Siglin

|  |  |
| --- | --- |
| Version/Author(s) | Date |
| 1.0/Jarrod Siglin, | 10/7/2022 |

TABLE OF contents

[1 INTRODUCTION 3](#_Toc83752298)

[1.1 Purpose and Scope 3](#_Toc83752299)

[1.2 Project Executive Summary 3](#_Toc83752300)

[1.2.1 System Overview 3](#_Toc83752301)

[1.2.2 Design Constraints 3](#_Toc83752302)

[1.2.3 Future Contingencies 4](#_Toc83752303)

[1.3 Document Organization 4](#_Toc83752304)

[1.4 Project References 4](#_Toc83752305)

[1.5 Glossary 4](#_Toc83752306)

[2 SYSTEM ARCHITECTURE 4](#_Toc83752307)

[2.1 System Hardware Architecture 4](#_Toc83752308)

[2.2 System Software Architecture 4](#_Toc83752309)

[2.3 Internal Communications Architecture 5](#_Toc83752310)

[3 HUMAN-MACHINE INTERFACE 5](#_Toc83752311)

[3.1 Inputs 5](#_Toc83752312)

[3.2 Outputs 5](#_Toc83752313)

[4 DETAILED DESIGN 5](#_Toc83752314)

[4.1 Hardware Detailed Design 5](#_Toc83752315)

[4.2 Software Detailed Design 5](#_Toc83752316)

[4.3 Internal Communications Detailed Design 5](#_Toc83752317)

[5 EXTERNAL INTERFACES 6](#_Toc83752318)

[5.1 Interface Architecture 6](#_Toc83752319)

[5.2 Interface Detailed Design 6](#_Toc83752320)

[6 SYSTEM INTEGRITY CONTROLS 6](#_Toc83752321)

System Design Document

# INTRODUCTION

## Purpose and Scope

The System Design Document describes the system requirements, operating environment, system and subsystem architecture, files and database design, input formats, output layouts, human-machine interfaces, detailed design, processing logic, and external interfaces for the Low-Cost Satellite Communications.

## Project Executive Summary

This section provides a description of the Low-Cost Satellite Communications project from a management perspective and an overview of the framework within which the conceptual system design was prepared.

### System Overview

The product is designed to be a low-cost alternative for optical communication system (OCS) satellites. Previous endeavors in the field can be fully utilized in order to lessen the financial burden of creating a new OCS along with using as many commercial-off-the-shelf components (COTS). Additionally, the project is geared more towards an educational mindset, with the long-term goals of commercial viability.

### Diagram Description automatically generated

**Figure1.2.1.1.1**: System overview basic architecture diagram.

1.2.1.1: Sensor: This is the sensor that takes in the data to be transmitted. Currently, this is a Raspberry Pi camera, however, it can be changed to be any number of other sensors.

1.2.1.2: OBC: This is the transmitter side Raspberry Pi.

1.2.1.3: Power System: This will be the portable power device to power the transmitter side of the system. It includes a rechargeable battery, a voltage converter for the various components which require varying input voltages, connectors mounted to the voltage converter compatible with connected devices battery output cables, and a battery charger.

1.2.1.4: Laser Driver: This is our laser driver board, the LD1100.

1.2.1.5: Laser: This is the laser diode, which will emit the laser.

1.2.1.6: Free Space: This is free space, or air, over which the laser will send data.

1.2.1.7: Detector: This is the laser detector. It detects if the laser is on or off and transmits it to the Raspberry Pi.

1.2.1.8: Pointing Mechanism: A device to automatically align the transmitter and receiver. This was removed from the scope of this semester.

1.2.1.9: OBC: This is the receiver side Raspberry Pi.

1.2.1.10: Display for demonstration: This is simply a display connected to the receiver Raspberry Pi over HDMI which will display the output video stream.

### Design Constraints

The project revolves around two major constraints, budget and size. The target for the budget is to keep all expenses under $1000 USD. To achieve this, commercial-over-the-counter systems (COTS) are to be used. The other major constraint, size, revolves around that this is to fit into a satellite and must be within 1U (10x10x10cm). By being compact and small, the size and weight limitations will allow for use on satellites. For previous satellites, total weight was less than 1.8 kg, meaning that this system should be less than that since housing for the communication system is needed as well. As for the power requirements, the transmitting laser will be operating at approximately 40 mA of power at 125mV, or about 5mW of power. The current batteries will support this, along with however much power the Raspberry Pi consumes, for about 5 hours using 3300mAh 18650 Lithium-Ion batteries.

### Future Contingencies

As of writing, the system is operational minus the optical transmission portion. The transmitting and receiving of video work on the Raspberry Pis, however, this is only over copper wire. The transmitter can produce a bit stream through OOK, but the laser detector portion of the receiver does not produce a measurable output voltage when connected to an oscilloscope. Also within the scope is a custom transmitter-side power system, which includes a battery charge circuit and voltage conversion. This is currently being designed and will be produced this semester. The optical pointing device is also in scope; however, it will likely be difficult and may not be able to be completed this semester.

## Document Organization

This section describes the organization of the Systems Design Document.

## Project References

None are currently applicable.

## Glossary

COTS – Commercial-off-the-shelf

LOS – Line of Sight

OCS – Optical Communications System

OOK – On-Off Key

RX – Receive

TX – Transmit

# SYSTEM ARCHITECTURE

This section describes the architecture of the system and/or subsystem(s) for the Low-Cost OCS Satellite project.

## System Hardware Architecture

As the final design choice has not been completed, variations in the hardware architecture are going to occur. The potential changes to the final design are listed above in section 1.2.3.

Raspberry Pi – Additional choice for the onboard computer.

ThorLabs – Optical Amplifiers and laser pointers.

Geekworm Power Management Board – Commercial battery hat manufactured for Pi board

Boost Converter Board – Converts 5.1V output of the power board to 9V for the laser driver

## System Software Architecture

Currently, the software being used to create the system is bash scripting. The mjpg-streamer library is a crucial component of the software, as it simplifies it considerably. The mjpg-streamer library can read from a camera device (currently the Raspberry Pi camera) and output to any number of different outputs. At this time, it is setup to output to a web server, as well as to a file. The web server is just to check and see what the video output looks like to compare with the video captured by the camera. The file output is input to the Raspberry Pi’s serial pin, and because everything in Linux is simply a file, this can be done by writing to the file for that pin. Since serial data is just a stream of bits, we can then connect this to the laser driver and act as though it is a wire. Once the receiving device receives the signal, it can then decode the serial signal back into a video stream, using the same mjpg-streamer library.

**Internal Communications Architecture**

The main communication system is laser communication between the transmitter and the receiver. This will use a serial UART interface to transfer data as a single bitstream from one computer to another.

# HUMAN-MACHINE INTERFACE

The limited human-machine interface is the input to the computer from the camera, which is then sent and displayed on the receiving computer on a screen.

## Inputs

The only input to the system is the live video feed from the camera on the transmitting computer. This then gets converted into a bitstream and sent to the receiving computer by way of a laser and laser receiver, respectively.

## Outputs

The output of the system will be the live video feed from the input, displayed on a screen, either directly off the receiving computer or streamed to a web server and viewed from the browser of another computer.

# DETAILED DESIGN

This section provides the information needed for a system development team to build and integrate the hardware components, code and integrate the software modules, and interconnect the hardware and software segments into a functional product.

## Hardware Detailed Design

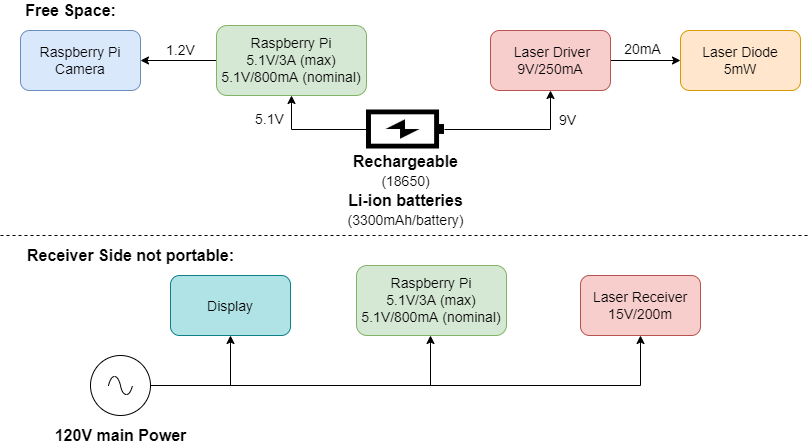
The hardware design follows as shown in Figure 4.1.1. The transmitting side of the system consists of a single board computer, the Raspberry Pi in this case, a camera, the laser diode, and all associated hardware, including the driver, modulation and power supply. The receiving side is somewhat the reverse of this, using the laser receiver, another Raspberry Pi, a power source, and a display. Whether the display is a physical screen attached to the Raspberry Pi or a web hosted live stream has yet to be determined, however, it will more likely be the latter. In the architecture diagram below (Figure 4.1.1), the blue arrows connecting each device represents a physical connection, for data and/or power. The square wave image represents a connection that implements the OOK method and is where the video data is transmitted. Any red arrows represent laser emission over free space.

## A picture containing text, device, gauge, meter Description automatically generated

**Figure 4.1.1**: Overall system hardware architecture.

### Power Supply Detail Design

The power supply consists of a battery hat manufactured by Geekworm. The battery hat is fed by two rechargeable Li-ion batteries that together provide the 5.1V needed by the Raspberry Pi board. The Pi board can supply the necessary power to the camera. The laser driver requires 9V, however. The conversion to 9V from 5.1V is accomplished using a boost converter mounted to a printed circuit board designed specifically for this application. The laser driver is connected to the 9V output of the boost converter and supplies power to the laser diode.



**Figure 4.1.1.1**: Power supply architecture.

The above figure, Figure 4.1.1.1 is a layout of the distribution of power from sources on the receiver and transmitter side to their respective components, separated by free space. Because the boost converter PCB was designed, some additional details are provided here. A screenshot of the opened board file is shown in Figure 4.1.1.2. The board was designed with dimensions that allow it to be mounted to the Raspberry Pi hat using the four bolt holes that match the bolt pattern of the hat and Raspberry Pi. A female-to-male 40-pin header was added to the board. This will plug into the male ends of the 40-pin header on the Pi hat so that the GPIO pins are still available for later use, if need be. A shunt capacitor and series inductor are included in the design before the 5V-to-9V converter. These work together to filter out noise that could otherwise get through to the driver on the other side of the converter and affect data transmission. The converter draws power from the Pi hat using pin 2, the same respective pin that the hat uses to supply power to the Raspberry Pi. Finally, power is delivered from the converter to the driver using a terminal block located on the right-hand side of the board in the screenshot.

### Diagram Description automatically generated

**Figure 4.1.1.2**: Screenshot of the boost converter PCB opened in KiCad.

## Software Detailed Design

The software for the system is rather simple from an overarching view, but complicated underneath. By making use of the mjpg-streamer library, the transmitting code can be reduced to a single line in a bash script. This script can then be run upon startup of the Raspberry Pi, or by a user starting it via the command prompt. There is no other software required for the laser part of the communication system, as the laser will simply be transmitting via on-off keying. On the receiving side, the complexity of the software has yet to be determined, however, once the initial setup and testing is complete, the software will theoretically be just as simple as the transmitting side. Unfortunately, not too many details on how the mjpg-streamer library works are not included in this document, as this is not fully understood by the team, but it does what is asked of it.

## Internal Communications Detailed Design

The internal communications of the system are just as simple as the software, in theory. The transmitting computer will send out the video stream which it receives from the camera as a bitstream to the serial pin. On the transmitting computer, the video data will be compressed using the standard MJPG compression rate (about 15:1) before being transmitted. The serial pin will then feed directly into the laser driver board as an enable/disable signal and will enable the laser when the signal is high and disable when it is low. The laser will then transmit this signal as light, which will be picked up by the laser receiver. The laser receiver will then decode and decompress the incoming video stream and output the video in a way viewable by the user.

## Diagram Description automatically generated

**Figure 4.3.1**: Internal Communications Diagram.

# EXTERNAL INTERFACES

This section will explore the external systems that will interact with the Low-Cost OCS.

## Interface Architecture

This system will not be interfacing with any external systems, as it is simply a communication link between two devices. This system can and will be able to function without the aid of the internet or any external services.

## Interface Detailed Design

Considering the air as the only external interface of the system, the only consideration that needs to be followed is one of safety. Lasers are dangerous tools and should be treated as such. This means that appropriate laser safety equipment should be worn when operating the system, and that care should be taken when using the system to prevent it from pointing in any unintended direction.

# SYSTEM INTEGRITY CONTROLS

Due to the nature of the project being geared towards educational use, and a lack of sensitive information being transmitted or received, the maximum level of control is simply physical access to the system. The project is using COTS and is intended to be built using previous works published in the field of OCS under educational or open-source licenses. As such there is no present need for security measures currently, but this is subject to change as the project evolves.