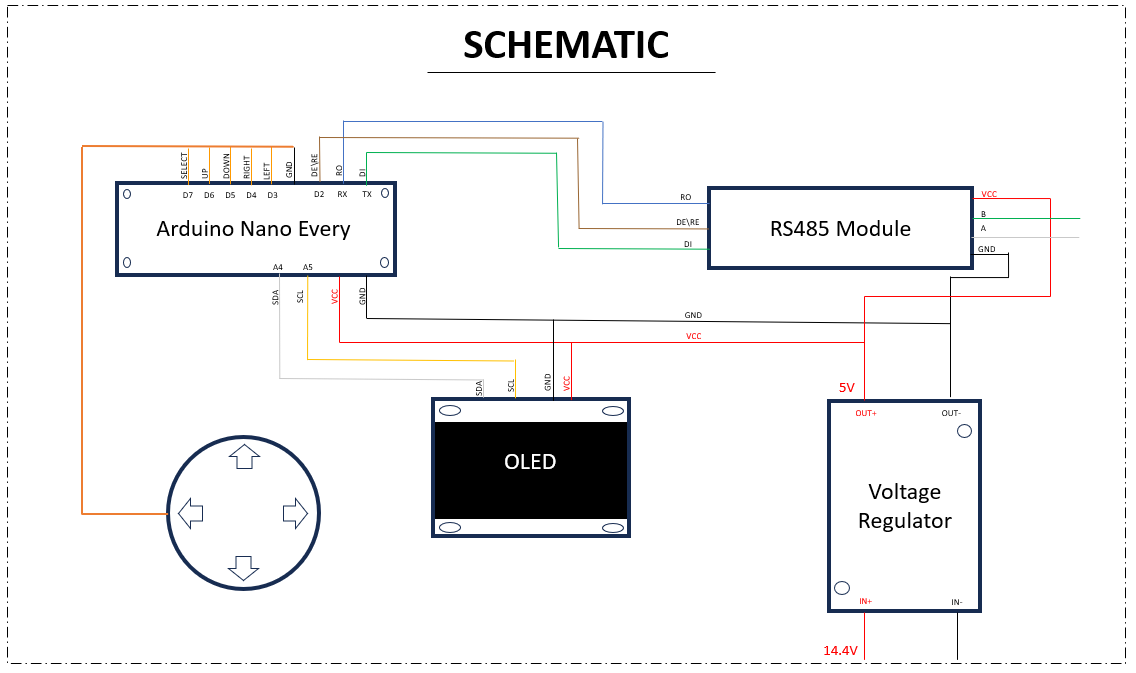
|  |  |
| --- | --- |
|  |  |
|  |  |

# Description

This device is designed to act as a replacement for the OEM DreamChip camera control software. Previously, field crews would need a computer, source of power and a cable kit to change video settings on DreamChip cameras using the control software using a Windows PC. This was inconvenient, expensive, and risky. Given how often crew members would need to change video settings in the field, a better solution was required.

# Electrical Design

When planning the project, there were a couple components that were essential. There needed to be a display for the menu and a microcontroller. From there, the RS485 was added since the camera and the microcontroller communicated via different protocols. The biggest obstacle was how to power both the microcontroller and the camera since



## RS485 Adapter

When communicating with another component, in this case a camera, it is important for the components to communicate with the same serial interface. Since the Arduino uses RS232 and the DreamChip uses RS485 as an interface, there needed to be a transceiver module between the microcontroller and the camera connection. For this project, a MAX485 TTL UART to RS485 Module was used. It is controlled by the UART interface and allows for serial data to be transmitted in both directions at high speeds. The following pins were used on the module:

* Supply Voltage (VCC)
  + Connected to the Voltage Regulator OUT+
* Ground (GND)
  + Connected to the Voltage Regulator OUT-
* Driver Input (DI)
  + Connected to the TX pin of the Arduino
* Receiver Output (RO):
  + Connected to the RX pin of the Arduino
* Driver Output/Receiver Input (A, B):
  + Soldered to a Hirose Connector that connects to the DreamChip
* Control Input Voltage (RE, DE):
  + These pins were combined using solder to have one enable voltage. This allowed there to be only one digital pin from the Arduino the enable read or write to the camera.
  + For example, the pin is set to HIGH when a command is being sent to the camera and LOW when the microcontroller reads the camera’s response.

## Voltage Regulator Module

## Microcontroller

## OLED Display

Originally, when choosing a display for the device, an LCD was chosen for the menu. However, there were multiple complications. In comparison with an OLED, the LCD drew more power, had less design functionality, and had slower refresh rates. After opting for the OLED, the original OLED chosen was a SSD1306 0.96-Inch Blue I2C 128x64 OLED Display Module. This display was chosen due to its functionality with the Arduino Nano Every microcontroller. Upon further inspection and testing with the OLED, it was noticeably too small. If this device is supposed to be seen by a variety of people, visibility is important. Thus, the 0.96-inch display was switched out for a 1.3-inch SH1106 I2C 128x64 display module.

The OLED uses the I2C protocol which is a two-wire data transfer bus that uses serial data and serial clock to transfer information. The OLED has four pins: VCC, GND, SDA, and SCL. The VCC and GND are connected to 5V and ground respectively. The SDA and SCL pins are connected to Arduino’s A4 and A5 pins that are specified for those pins.

As for the microcontroller’s communication with the OLED, the Arduino IDE was used in conjunction with the Adafruit GFX 1.11.7 and SH110x 2.1.9 libraries.

## Joystick

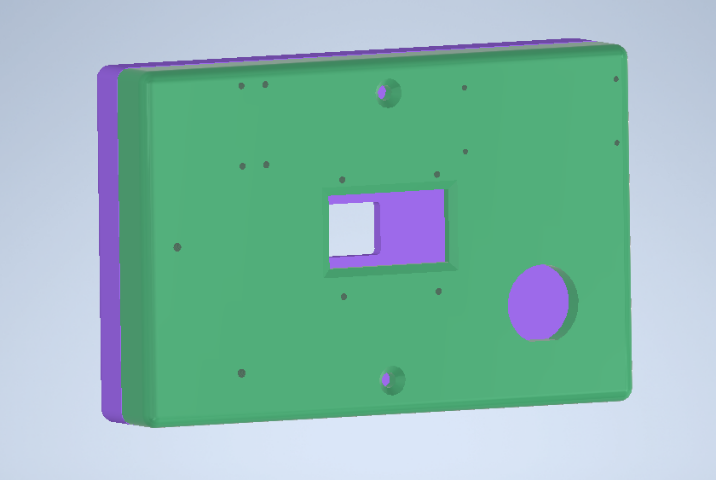
The Ruffy Controls HS1-6 ‘mini-joystick’ was selected as the primary user input device for a few reasons:

* It was already used in some existing designs and was readily available in stock.
* It is easy to mount and wire to the microcontroller.
* It is mechanically robust and resistant to abuse.
* It had more than enough controls for all the user interface options required.

# Mechanical Design

One of the main focuses of this project was to make a small controller that can easily change the state of the cameras without needing to carry a laptop and manually plug in the camera. When designing the physical case, it was important to measure the exact size of each component and how to fit them together efficiently. As for the software used for the design, AutoDesk Inventor was used (Figure.1). This program made it efficient to make changes to the design as obstacles were encountered. For instance, after the design was made and printed, there was an issue when it came to the lack of strain relief for the cable connected to the camera. If left without a strain relief addition, if the controller operator were to pull the cable too harshly, it could possibly damage the circuit within the device. Thus, a path was made for the cable upon entering the controller that constricts its movement.

There were multiple prototypes that moved around the OLED and joystick, but it made the most sense to have the display at the center of the screen and the joystick to the right. If the person who operates this controller is left-handed it may make sense to switch up the location of the joystick and Voltage Regulator (see Figure.2 for the layout).

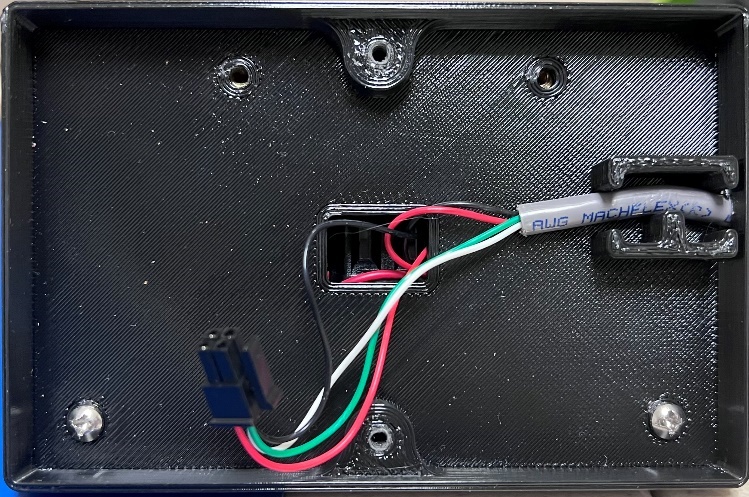


*Figure.1*

A close-up of a circuit board

Description automatically generated

*Figure.2*



Strain Relief

*Figure.3*

## Battery Choice

The decision to use a Gold Mount battery connector was made for several reasons:

* Gold Mount batteries are ubiquitous in broadcast settings and are easily available in any environment this device is likely to be used in.
* They have a nominal voltage that is well within the requirements of this project to power any DreamChip camera with the 6-pin Hirose Connector. In particular, the 4k mini line of cameras requires a minimum of 9v in. (Other larger cameras may be compatible with this device but may need an adapter harness)

# Software Design

## Passthrough Mode

The default way to control the DreamChip is to connect it with a USB RS485 Converter that connects directly to the computer. Commands can be sent through a command line or a GUI, like ProVideo.

Arduino MegaAVR Libraries 1.8.8

Adafruit GFX 1.11.7

Adafruit SH110x 2.1.9

Adafruit BusIO 1.14.2

AceButton 1.10.1

# Bill of Materials

* Arduino Nano Every
* ANMBEST MAX485 RS485 Transceiver Module
* DIYmall 1.3-in Blue OLED I2C Serial 128x64 Screen
* Ruffy Controls HS1-6 ‘mini-joystick’
* Hirose Connected
* 4 Conductor Cable
* GoldMount Battery