Portable Microscope

Austin Bryant - Computer Scientist
Adam Bush - Computer Engineer
Cayla Gill - Electrical Engineer
Hannah Pierson - Optics and Photonics Engineer

Motivation

- Scientific learning tool
  - Educators, Students
  - Incorporate smart devices
- Encourage involvement in STEM
  - Get people excited about the scientific world
- Ease of access
  - Untethered
  - Multiple people can view display at once

Overview

It will be a battery operated microscope that can be used without a stand to explore the surrounding world.
The microscope will transmit the images via Wi-Fi to an Android app. The user will be able to apply filters, add tags, and save photos of the images in the app.

Goals and Objectives

- Portable and lightweight
- Wireless
- Able to resolve an onion cell
- Imaging in visible
- App-enabled

Specifications

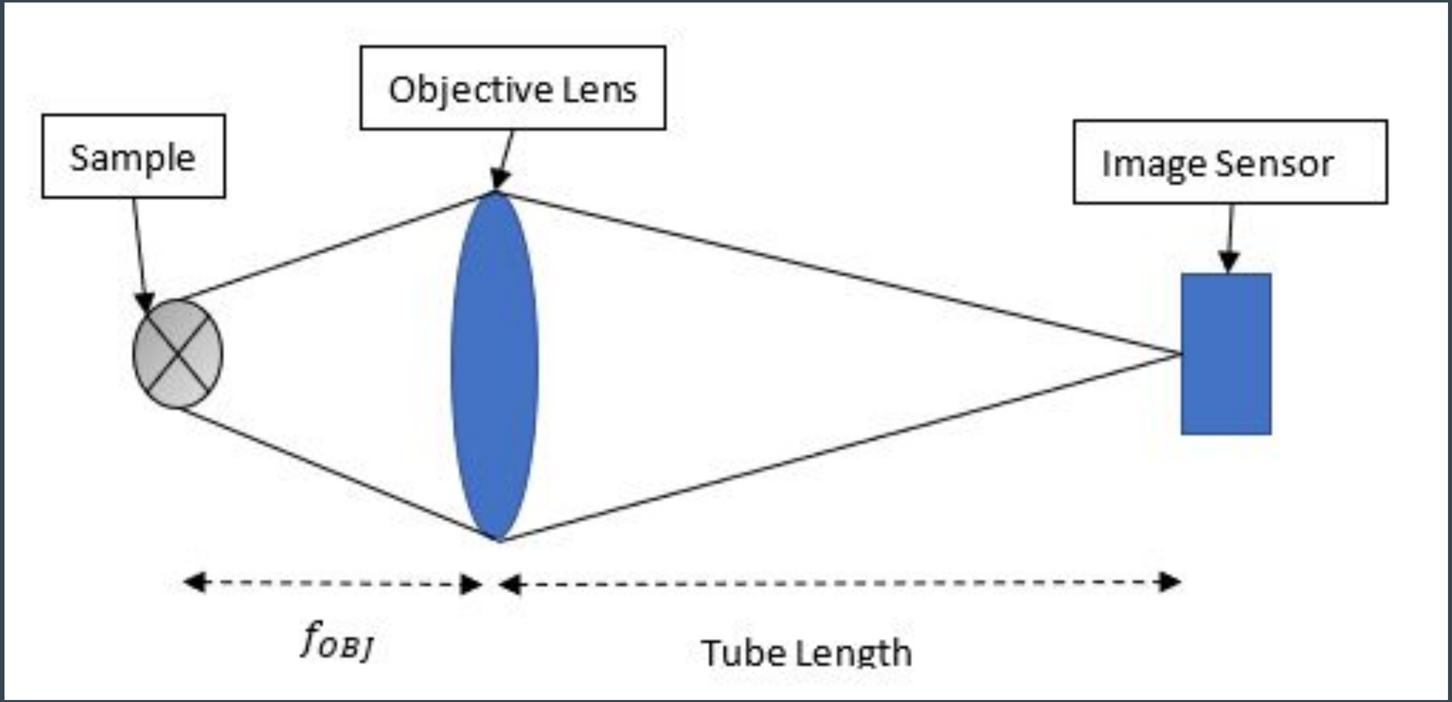
- Under 10 lbs
- Transmitting at 10 FPS
- Able to resolve at 120 um
- Delay of less than 1 second

Hardware Designs

Optical Design Overview

- Sensor
  - OV5647
- Objective Lens
  - DIN Standard
- Body with moveable parts
  - Custom built
- Light Source
  - White LEDs, and 850 nm LEDs

How It Works



Sensor

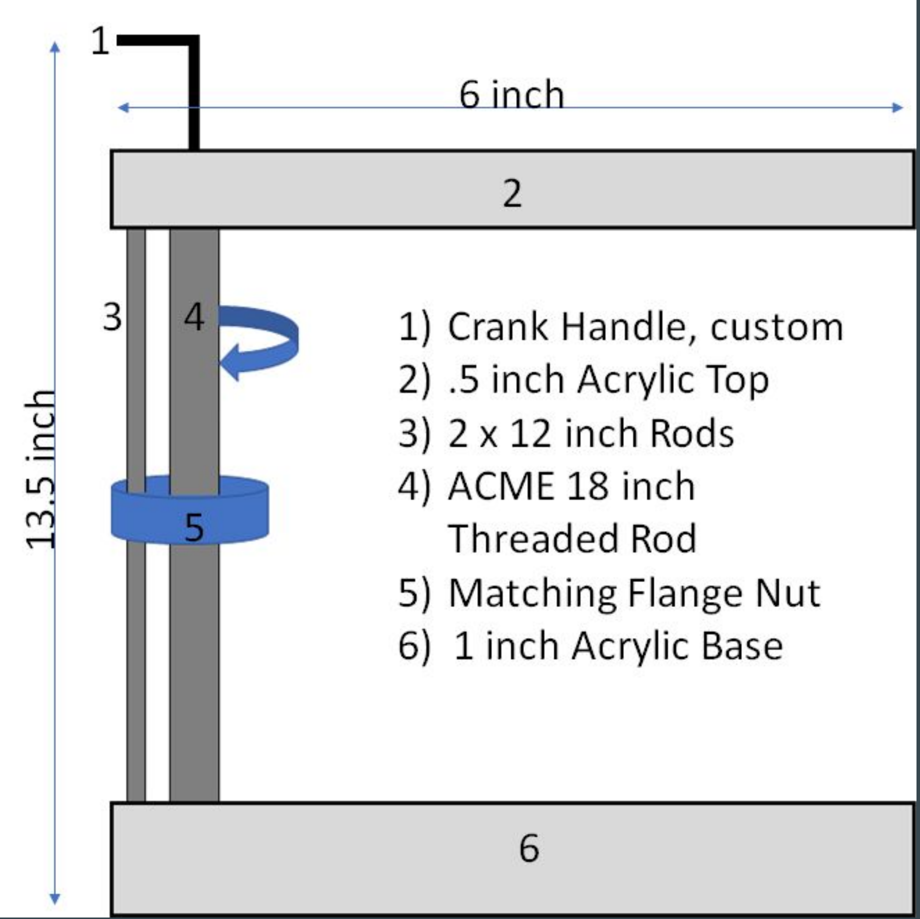
Raspberry PI Camera Module
1.4um pixel size
5 megapixel
2592*1944 pixel array
Up to 90 fps

Objective Lens

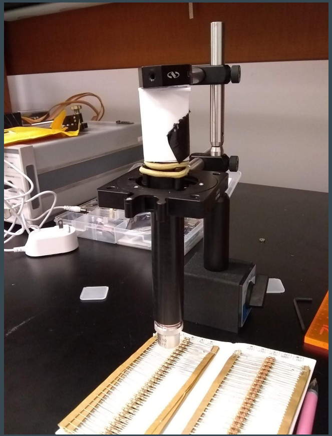
Pros and Cons of different Objective Lenses

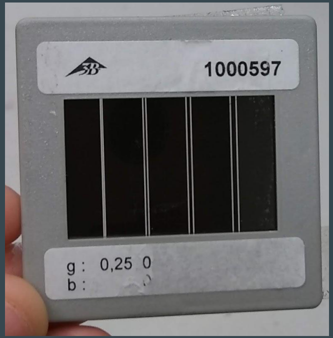
DIN Standard	Infinity Corrected
One Component	Requires additional Lens
$20	$140
Fixed Focal Length	Fixed Focal Length
Difficult to Focus	Easy to Focus
Standard Tube Length	Tube Length Varies based on Brand

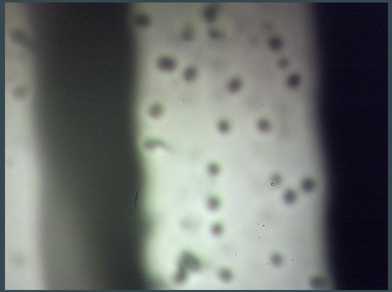
Optomechanical Design Moving Elements



Optical Testing So Far...







Power Supply Design

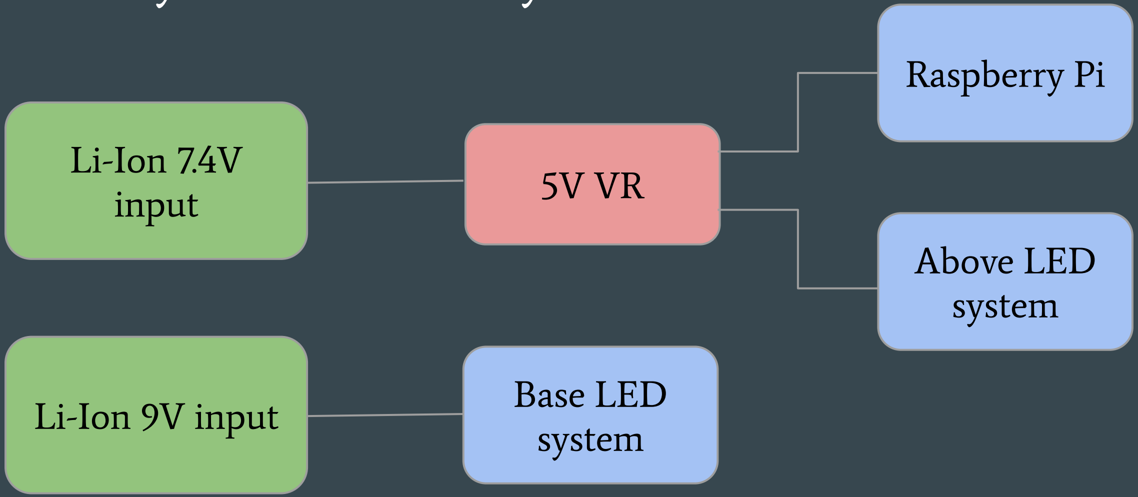
- Power supply constraints
  - No tethering
  - Portability
  - Low power system
  - Compact size
  - Lightweight
- Batteries work around all constraints

Power Supply Design

- Lithium Ion Batteries Selected
  - High energy density
  - Low self-discharge
  - Durable
  - Lightweight
  - Compact
  - 2 - Lithium Ion with 7.4V combined output

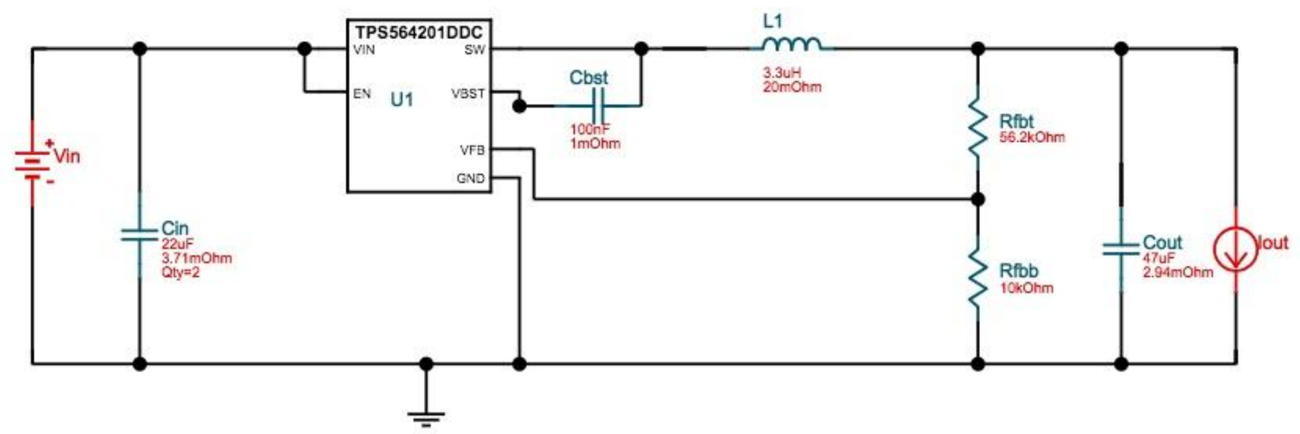
Power Supply Block Diagram

- 2 - 3.7V Li-Ion Batteries
- Voltage regulation needed to protect components
  - One 5V regulator
- 9V Li-Ion Battery for base LED system

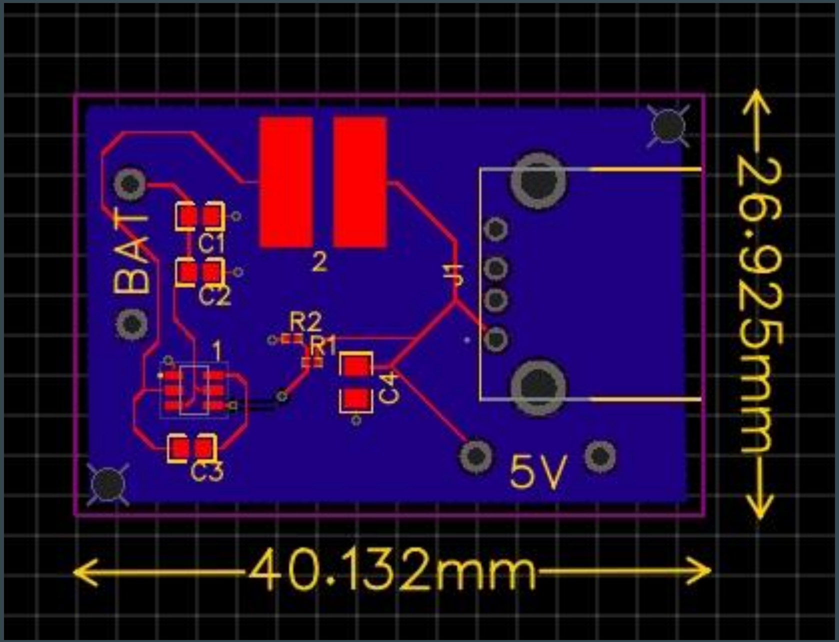


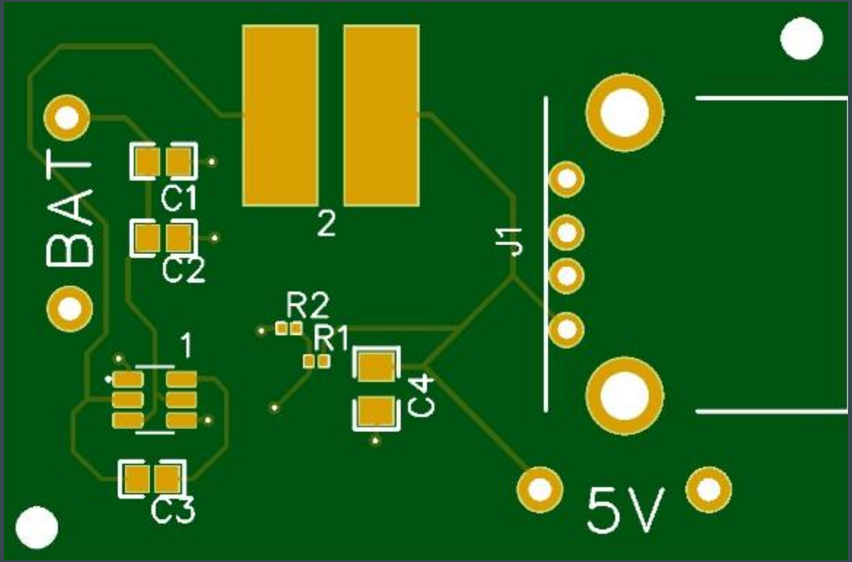
Voltage Regulation Designs

- TI Webench Designs
- 5V Regulator needed for Raspberry Pi and above lighting system



Voltage Regulation Design

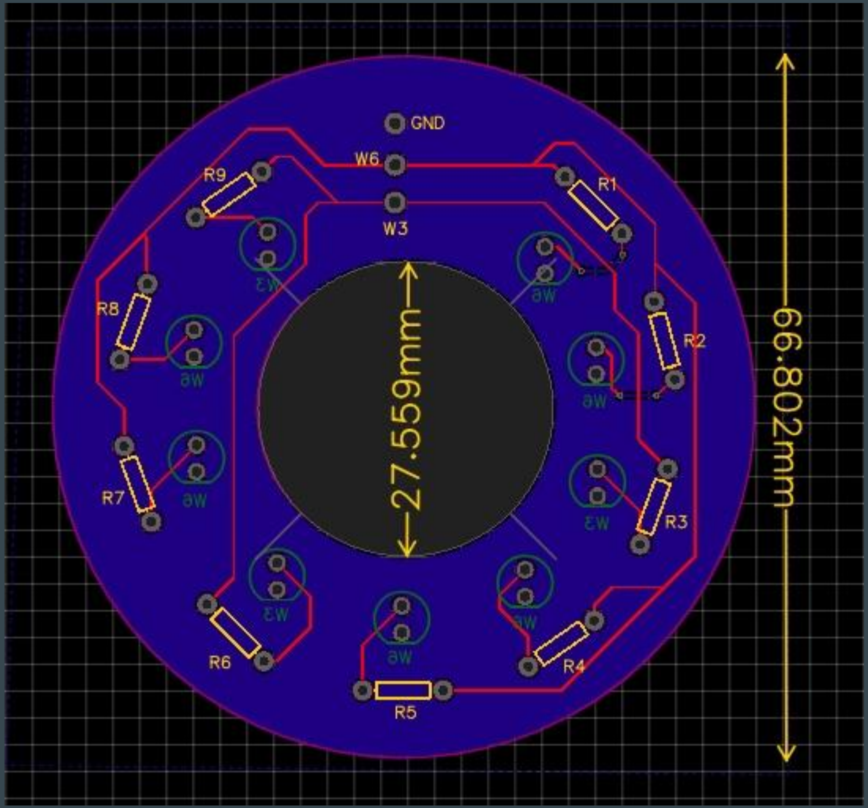




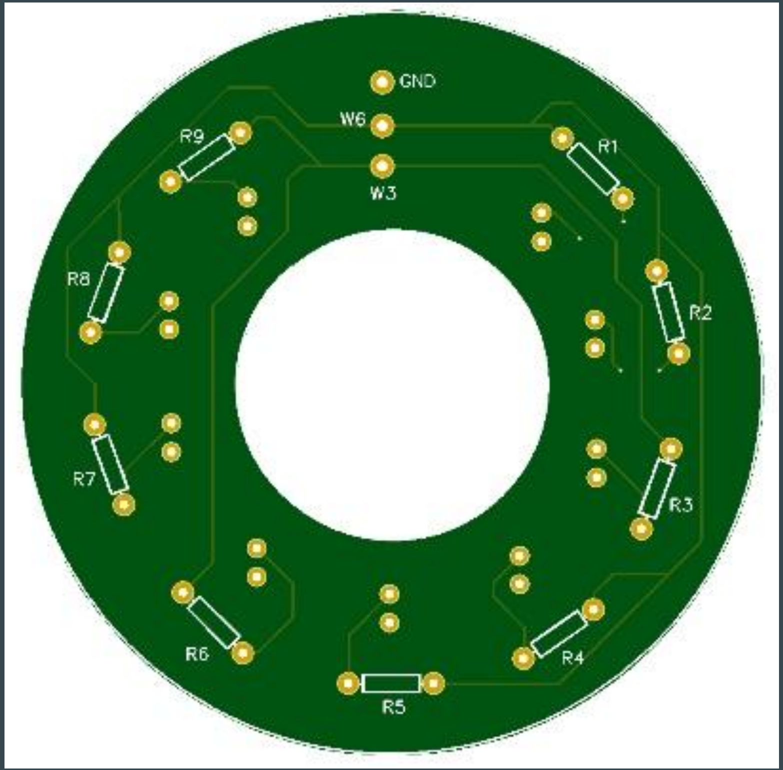
Above LED PCB Design

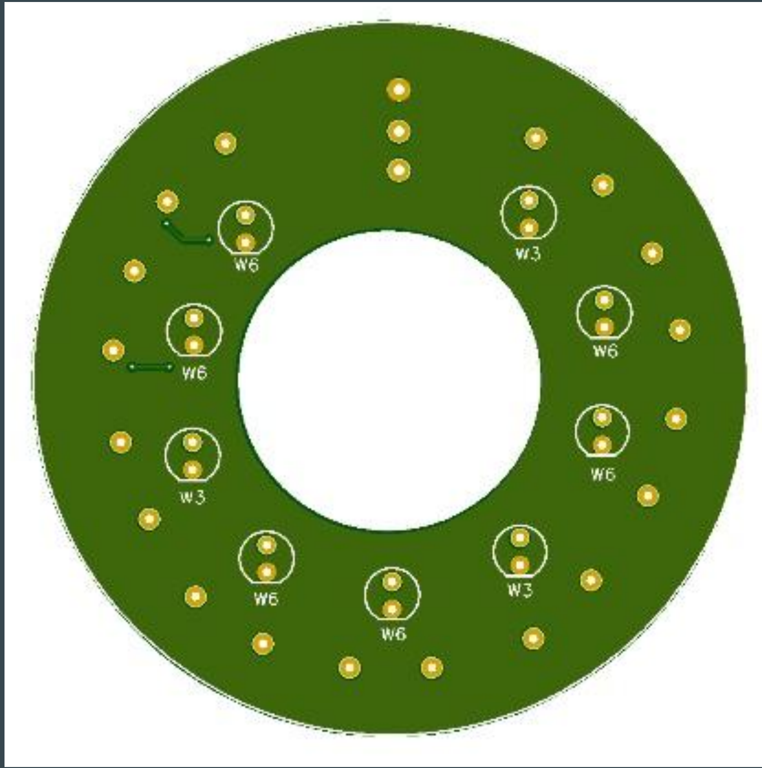
- Upper light adds illumination from above
- On-off-on switch allows option for 3 or 6 diffused LED illumination
- Lights will face towards observed sample
- Able to be adjusted depending on lens used

Above LED PCB Design



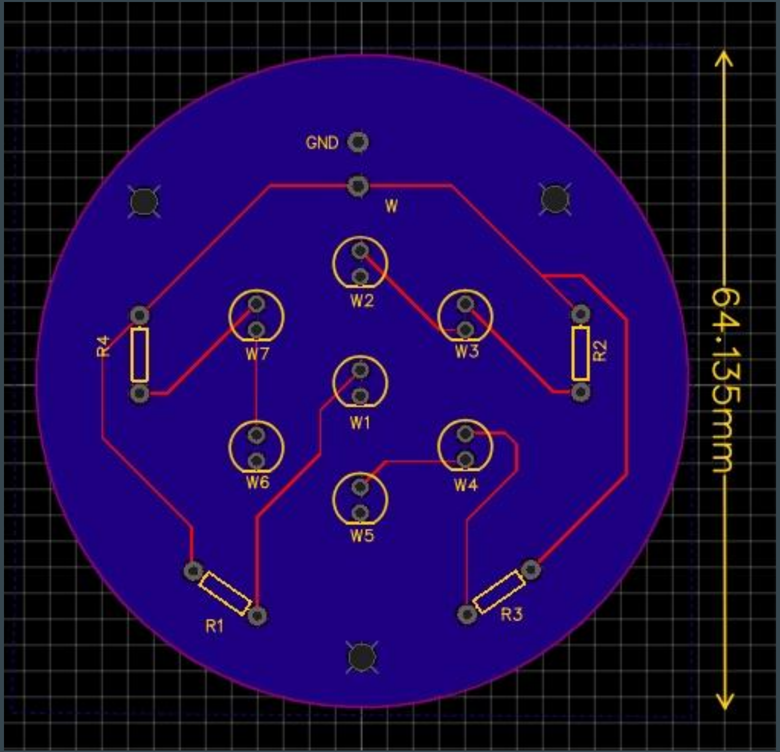
Above LED PCB Design



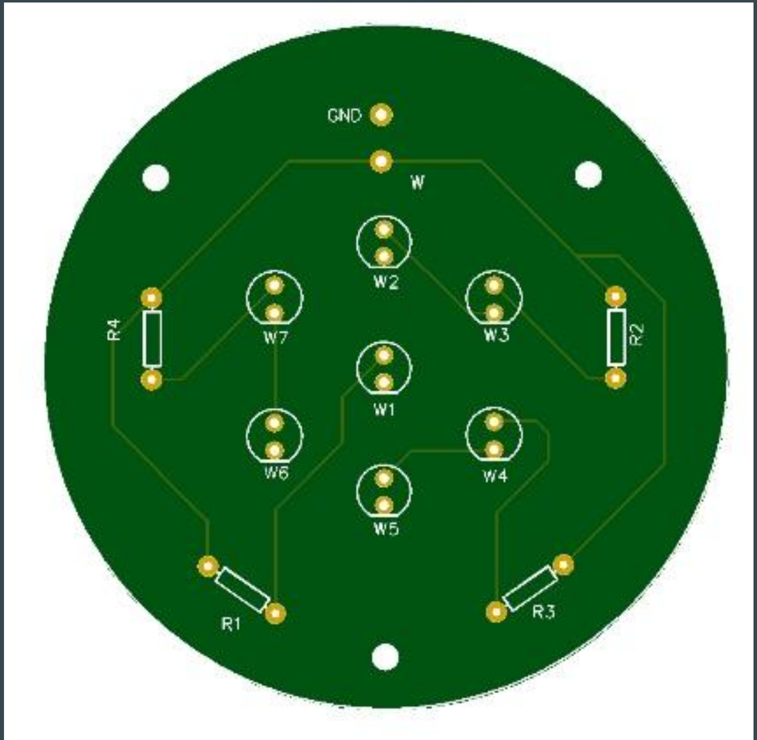


Base LED PCB Design

- Used for illuminating slide samples from above
- Seven diffused LEDs powered by a 9V Li-Ion battery
- Removable disc with diffusion filter for more dispersed light



Base LED PCB Design



Software Designs

Raspberry Pi Image Acquisition

- Python script that grabs images from OV5647 camera module
- Socket connection is established
  - JPEG images are sent continuously over socket connection
  - ~40 FPS is possible using this method
- Script is started as a service when the Pi boots up
  - Allows the script to be run headlessly

Mobile Application Overview

- Android Application
  - Written in Java
  - SQLite, OpenCV
- Connect, Stream, Modify, Store
- Image Processing, Data Storage/Manipulation

Mobile Application Wi-Fi

- Wireless Access Point (WAP) hosted on Raspberry Pi
    - Range & stability benefits
- Why WAP over Wi-Fi direct?
    - Could not reliably get devices connected
    - Connection process is much simpler for the user

Mobile Application Wifi (Continued)

- Data received from Raspberry Pi
  - Sent via TCP socket connection
    - Static IP address, port number
  - Blocks of bytes are received
  - Bytes stored and encoded into JPEG format

User Interface Design

- Goal was simple and intuitive
  - Keep user experience in mind
  - Not overbearing
- Use-case
  - Classrooms, hobbyists
- Emphasis on launch to stream time
  - Core functionality is streaming

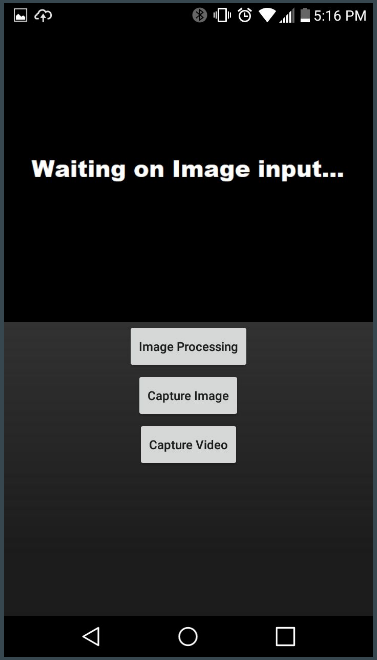


Image Processing

- OpenCV4 Android SDK
  - Massive Image Processing Library
    - Gaussian Blur, Edge Detection, etc.
- Performed directly on media files
  - Displayed in ImageView
- High level of granularity
  - Processing parameters can be fine-tuned

Data Storage

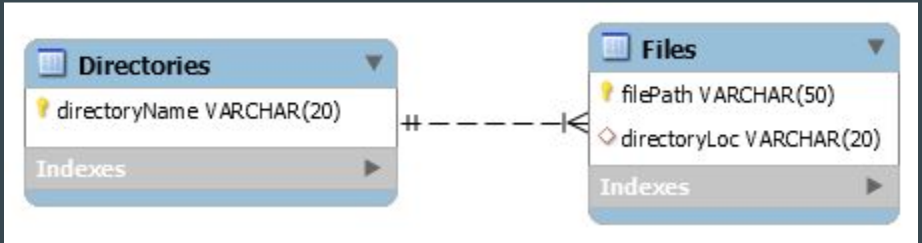
- Captured media files must be stored somewhere
  - Allow user to choose location and categorize as needed
- Device's external storage will be used
  - On Android OS, internal storage is only usable by app
  - User might want to access media outside of app
  - Files won't be deleted when the app is uninstalled

SQLite Data Storage

- SQLite
  - Relational Database Management System
  - Self-contained, no remote server needed
- Why SQLite?
  - Tagging allows for users to access files without having to view the entire filesystem
  - Files are already unique to each user
  - Remote database introduces unnecessary overhead

SQLite Schema Breakdown

- References to file paths will be stored
  - All files must have parent directory
  - If none specified, default is "Unfiled"
  - Dynamically updated when queried



Administrative Content

Work Distribution

	Optical System	PCB Design	Power Design	Body Construction	Mobile App	Image Processing
Cayla EE		P	P	S		
Hannah PSE	P			P		
Austin CS					P	P

Research Costs

Product	Price
Pocket Microscope	$14
USB Microscope	$32
Raspberry Pi accessories	$26
Raspberry Pi	$43
Camera Module	$45
PCBs	$60
Circuit Parts	$45
5mm white LEDs	$5
Ribbon Cable	$6
Pi Case	$8
Optical Design	$88
Objective Lenses	$48
Li-Ion Batteries (3 sets)	$43
Charger	$18
USB cable	$8
Acrylic	Donated by CREOL machine shop
Delrin	Donated by CREOL machine shop
Sample Slides	$31
Android Program	$25
Battery Packs	$15
Total	**$560**

Production Costs

Product	Price
Raspberry Pi	$43
Camera Module	$15
PCBs	$30
Circuit Parts	$15
5mm white LEDs	$5
Ribbon Cable	$6
Pi Case	$8
Optical Design	$88
Objective Lenses	$48
Li-Ion Batteries	$14
Charger	$18
USB cable	$8
Application	$5
Acrylic	$20
Delrin	$5
Total	**$328 per unit**

Issues

- Electrical
  - Working with new components and custom footprints
  - Overall PCB design software
- Optical
  - Mechanical Design, weeble wobble
  - Getting a suitable test sensor
- Computer Science
  - Designing UI so that it is intuitive but still looks *FABULOUS*

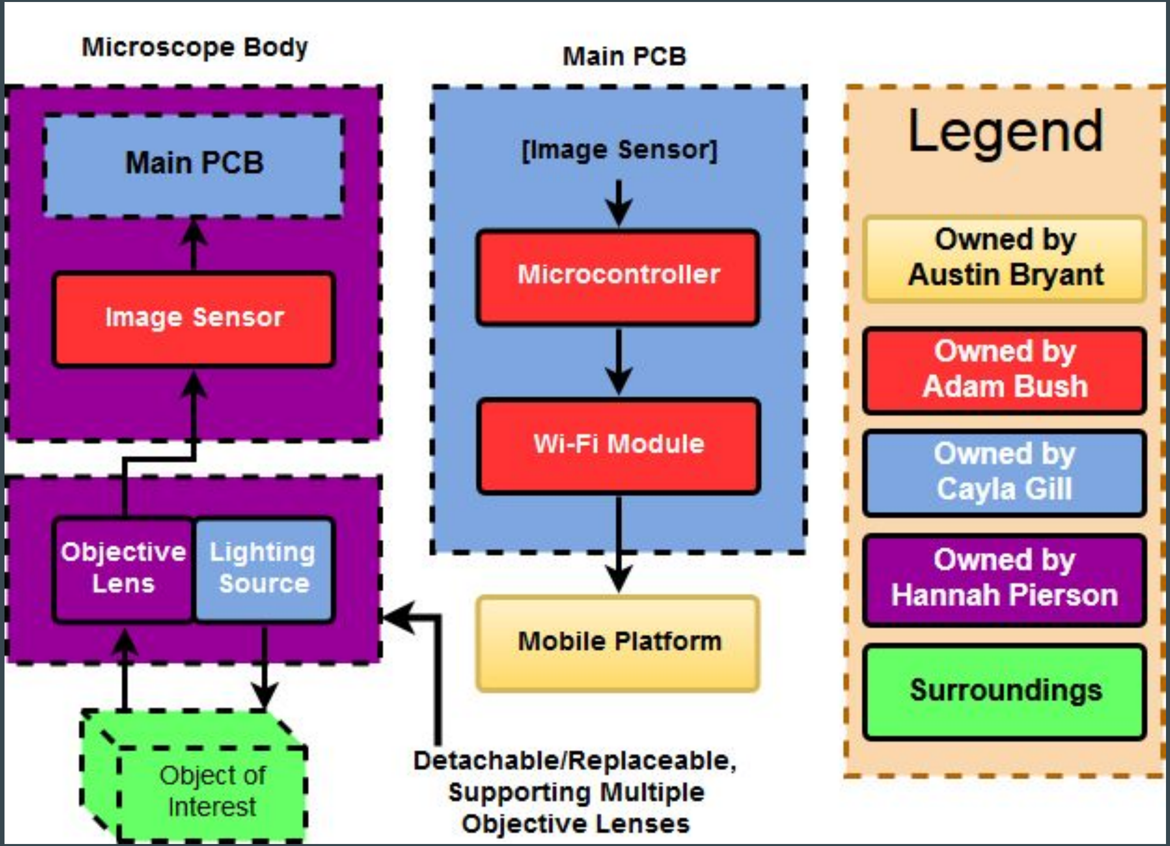
Upcoming

- Electrical
  - Working with wire connections to ensure stable wiring
- Optical
  - Building the optomechanical system and more testing
- Computer Science
  - Implementing SQLite/Storage functionality

Questions?

ADAM

Hardware Block Diagram



Software Block Diagram

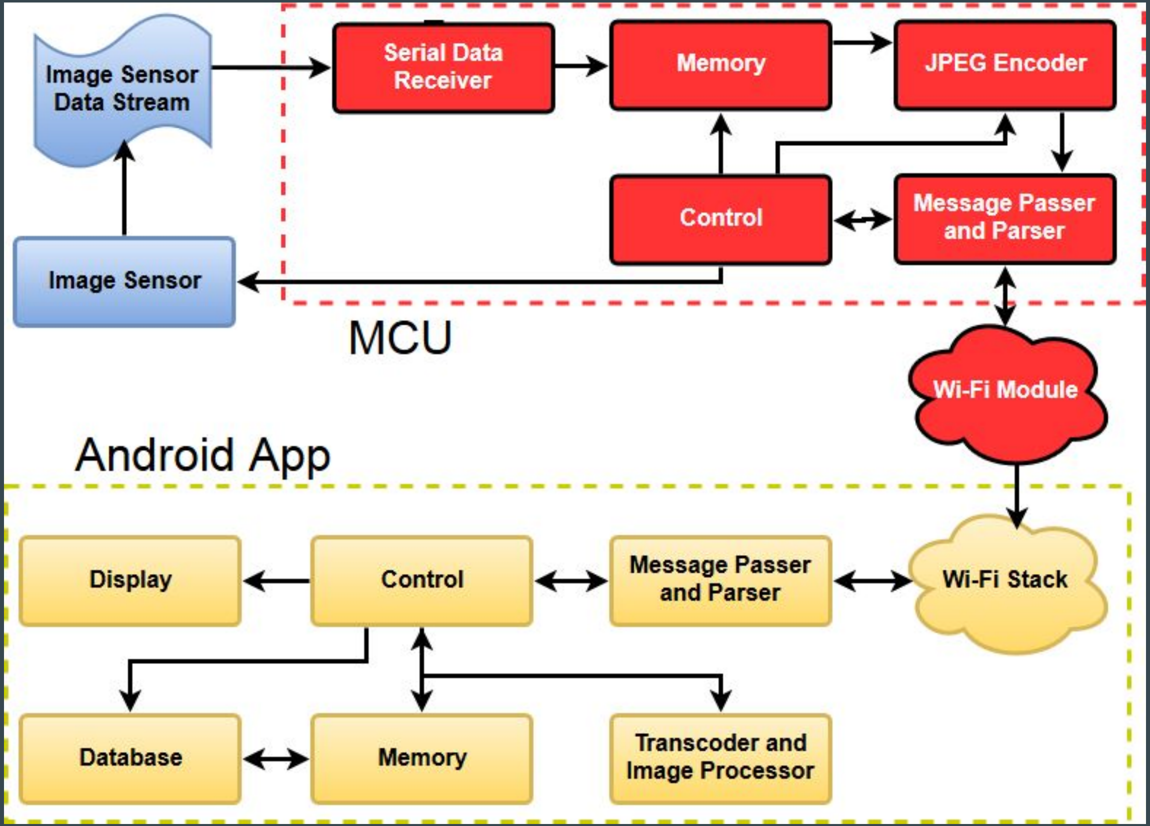


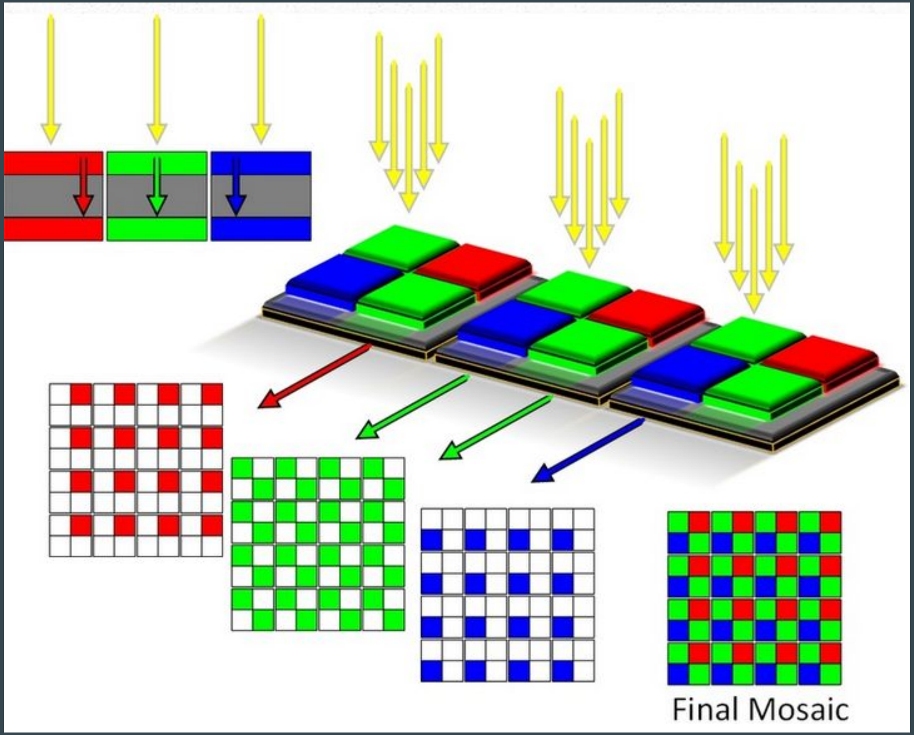
Image Sensor - AR0237IR

- Capable of sensing near-IR light
- 12-bit, parallel data output (synchronous)
- 30FPS at its maximum 1080P (with parallel output)

One other similar sensor was found on the market, the OV9738, but it had a 10-bit output (1/4th the output range of our choice)

Image Sensor - AR0237IR

Standard Sensor Matrix [ref 1]

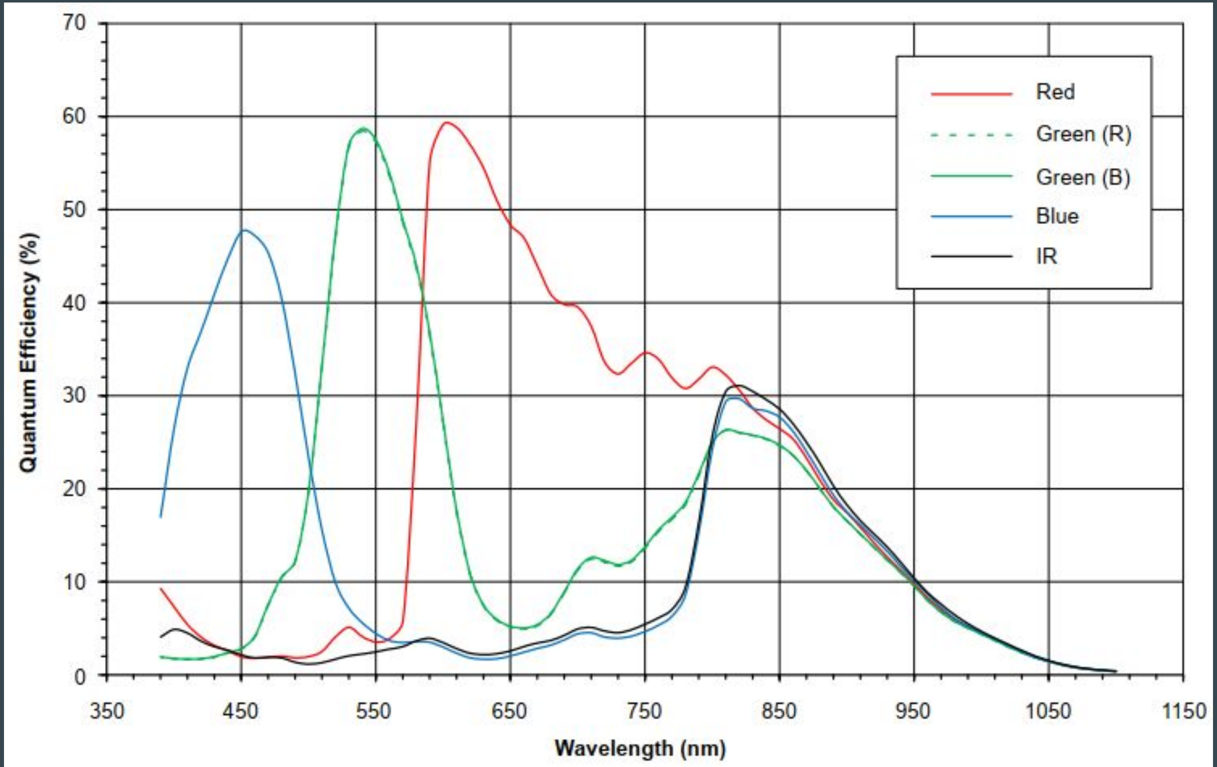


AR0237IR Matrix [ref 2]

B	G	R	G	B	G	R	G
G	IR	G	IR	G	IR	G	IR
R	G	B	G	R	G	B	G
G	IR	G	IR	G	IR	G	IR

Image Sensor - AR0237IR

Sensor Sensitivity [ref 2]



STM32H753ZI - Microcontroller

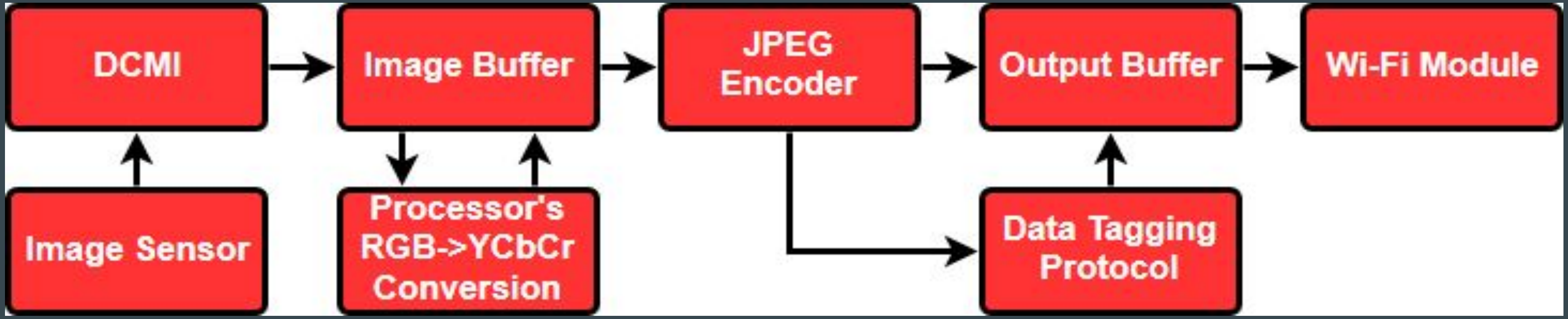
- 400MHz superscalar ARM Cortex-M7 core
- 1MB of RAM with further 2MB R/W Flash memory
- Embedded JPEG codec
- Direct Camera to Memory Interface
- Hardware Abstraction Layer library providing implementation assistance for most required tasks
- Supported, inexpensive development board, the NUCLEO-H743ZI

GS2101M Wi-Fi Module

- Dual-core ARM Cortex-M3, 120MHz, ~500kB RAM each
- 4-bit serial data interface (as SDIO slave)
- TCP throughput tests (Module to Receiver) > 1MB/s
  - Note: calculated throughput need of approx. 700kB/s
- Supports Wi-Fi Direct and (currently unused, needed for a realized product release) encryption

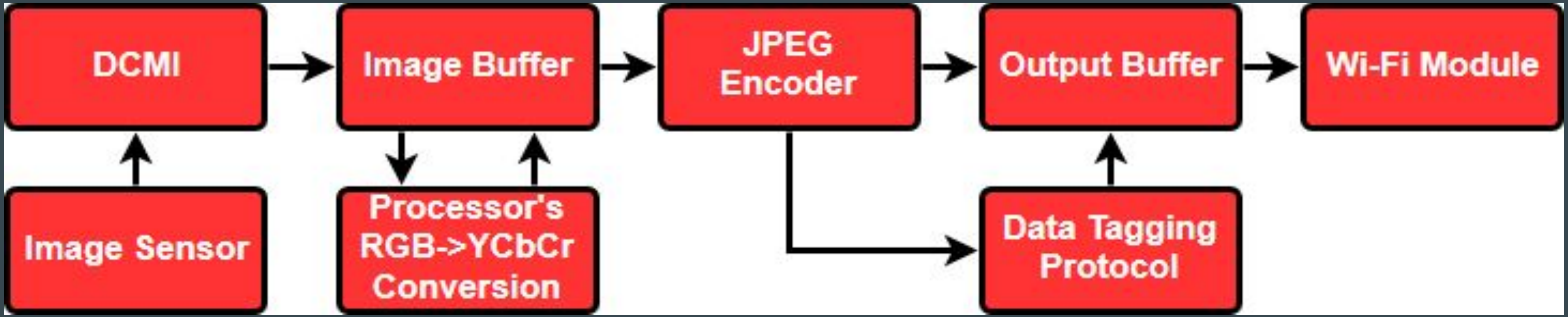
MCU Program Flow (Part 1)

1. DCMI transfers incoming image to 512kB buffer
2. DMA transfers in-memory image to processor's tightly-coupled memory (block by block)
3. Processor converts RGB to YCbCr (for JPEG step)
4. DMA returns data to JPEG encoder buffer



MCU Program Flow (Part 2)

5. Processor triggers JPEG encoding, which takes ~4ms
6. DMA transfers converted image to output buffer (2x128kB)
7. Processor adds header (for mobile application) and initiates transfer to Wi-Fi module



Wi-Fi Module Setup and Use

- Largely developed by configuring manufacturer’s SDK
- Programmed through UART by host microcontroller
- Dual-Interface configured through human-readable AT Commands sent by UART
- Wi-Fi connection initiated by prompt
- Data transfer initiated by UART prompt, carried out over 4-lane serial (SDIO) interface
- Further development and use of the module’s limited processing capabilities is not planned

References

1. "Analogue Video Capture", https://www.causewaysecuritysolutions.com/analogue-video-capture.html
2. AR0237 series product overview