

The Magician’s Assistant

Isayah A. Parent



28 November 2023

University Of maryland – College Park

ENEE408A Section 0101 - FALL 2023

Professor William Hawkins

# Table of Contents

|  |  |
| --- | --- |
| Honor Pledge + Approval | 2 |
| Executive Summary | 3 |
| Introduction | 4 |
| Goals and Design Overview | 5 |
| Realistic Constraints | 6 |
| Engineering Standards | 7 |
| Alternative Designs and Design Choices | 8 |
| Technical analysis for system and subsystems | 9 |
| Design validation for system and subsystems | 10 |
| Test Plan | 11 |
| Project planning and management | 12 |
| Postmortem | 13 |
| Salvage | 14 |
| Recycle | 15 |
| Conclusions | 16 |
| References | 17 |
|  |  |

# Honor Pledge + Approval

*“I, Isayah Parent, pledge on my honor that I have not given or received any unauthorized assistance on this assignment/examination.”*

As this project was entirely a solo endeavor, I take credit for all contributions made towards the Magician’s Assistant. This includes the hardware and software design, programming, testing, and data collection.

Signed : Isayah Parent

Executive Summary

* Goals
  + Initial Goal : A device capable of safely shuffling a deck of Magic the Gathering cards without bending, or defacing them in any way.
  + Updated Goal : A device focused on processing each card in the deck via Image to Text Artificial Intelligence, then packaging the scanned card data in a format useful for online deckbuilding.
* Paths
  + Interviewing avid trading card game players to find their qualms / ideas about playing trading card games in person as well as online
  + Splitting the device into a set of subsystems linked with various communication protocols
  + Designing the M.A. as if it were going on the market
* Decision Making
  + Considerations about overall size
  + Comparisons to what exists in the market now
  + Considering user experience and how the game works
* Prototyping
  + Materials + Programs used
  + Quick Explanation of why the shape / apparatus matter so much

Introduction

* Definitely echo what was said in the Executive Summary but with more detail
* Context
  + Explain my original inspiration for the Magician’s Assistant
  + Explain who the MA was intended for and what happens in a game
* Exigence
  + Briefly explain why trading card players care so much about their cards
  + Briefly explain what a “Magic Commander Deck” is and how that was the basis of the design
* Intended Outcome
  + Explain why I built as if I were taking this device to the market and how it compares to what exists

# Goals and Design Overview

* Explain the early shift in design goals
  + Why a scanner is much more useful than just a shuffler
* Main Design demands
  + Speed
  + Card Protection
  + Ease of Use in-game
* Immediate concerns / implementation questions I had at the start
* Block Diagram
* Subsystem Overview
  + What the systems are and how they communicate (high-level)
* Spec Sheet

Realistic Constraints

1. Economic – In its current design, the MA could be used to temporarily store a deck but only one at a time so I would want the device to be cheap so that people wouldn’t mind buying multiple for each deck.
2. Safety – The cards’ safety is honestly more of a concern than the users’. The device needs to be able to process cards without devaluing them at all
3. Safety 2 – Since the device is meant to be able to interface with the Archidekt API, user data needs to be protected
4. Manufacturability - The MA needs to be simple enough that its operation is simple which means that its mechanisms should also be simple enough to avoid the need to troubleshoot. Plus the device needs to be portable.
5. Social – In actual use, no one would want to lug an ugly machine to Games Night so the final design would need to be modern and hideaway all of the internals (for aesthetics and also to keep them safe)

# Engineering Standards

* Protocols used
* Software development practices
* <Not too sure about what to include in this section>

Alternative Designs and Design Choices

* Include Pugh chart from HW2
* Explain decisions behind I2C, SPI, UART, and the peripherals I chose
* Explain how Vegas card shufflers work and why I decided against that

# Technical Analysis for System and Subsystems

* Dive into clock speeds, PWM, camera registers, buffer sizes, libraries, specific hardware used
* Show CAD and photos of device
* Shortcomings
  + Bulky design for MK1
  + Forcing the design to be synchronous
  + Can’t save photo data in bulk
  + Requires WiFi
* Explain Block diagram / Data Flow
* Anticipated Errors
  + ESP and STM missing their handshake
  + Slow WiFi connection on ESP / Slow execution on STM making everything slow down (or timeout entirely)

# Design validation for system and subsystems

* Show photos from Arducam
* Show the ESP Web Server
* Show the Servos flicking cards
* (Hopefully) show successfully translated text from camera images and the resulting deck list on mobile device
* Show links between Arducam/STM, ESP/STM, Servos/STM, ESP/HTTP Interface

Test plan

* Testing
  + Field tests of other players trying to shuffle their decks manually vs using the MA
  + Post-Use survey / Feedback
  + Finding the average accuracy of card scans over a large dataset of MTG card images

Project planning and management

* Detail the timeline I had
* The order of implementation and why I chose that order
* Show plans from journal/sketches

J. Post mortem (discussion of failure to reach goals)

* from subsystem standpoint
  + Explain how specific systems could have been improved.
  + Bonus Features that I’ve thought of for Mark II
    - RFID Transfer
    - Saving Deck to Flash/SD Card
* from system standpoint
  + Explain how using an asynchronous design would be much faster
* from technology standpoint
  + The 2MP Arducam is very blurry and takes a lot of configuration to capture a clear enough photo
  + Servo “fingers” would benefit from an elevation changer so that the tip only touches one card
* from resource standpoint
  + On board flash and ESP flash are too small to hold all the photos

Salvage

* To meet reduced goals
  + I would want to replace the Arducam with an Arducam 5MP
  + Would need to re-organize the breadboards to save space and make a smaller chassis
  + Would replace the LED chain that I’m using for the camera flash by using an actually bright, pointed light
  + Would want to make the chassis totally closed and dark so that the camera can really focus
  + Would also want to make a dedicated Android App rather than connecting directly to the ESP’s IP

Recycle

* On a larger scale, the MA could definitely be used to sort through other kinds of documents like notebook pages or loose leaf papers
* The Design is also intentionally modular, so if need be, new subsystems could be attached fairly easily

Conclusions

* Brief recap of my data flow and what parts worked well together
* Creative approaches: My ESP <-> STM command “bus”, Communicator->Controller->Actuator paradigm
* Details unexpected challenges and how I would address them in Mark II
* Lessons Learned : New languages, new protocols, register controlling, PWM

References

* Will attach these as necessary