

Software Requirements

• Accuracy:

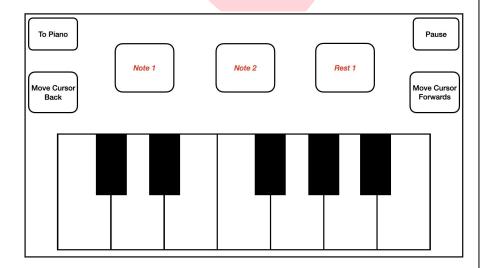
- 75% for eye-commands
- 95% for screen-section

• Latency:

- 500 ms for eye-commands
- 100 ms update for secondary UI

• Coverage:

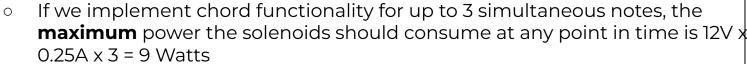
- One octave
- Two note lengths/one rest length
- Moving the cursor forward/back
- Accessibility:
 - Only eye-commands



Hardware Requirements

Use Case Requirement: Playing notes of composed piece Technical Design Requirements:

- Power regulator outputs
 - o 12V, 5A
 - 3.3V
- Solenoid power consumption
 - Each solenoid 12V, 250mA



• Solenoid duty cycle

o 100%



Why?

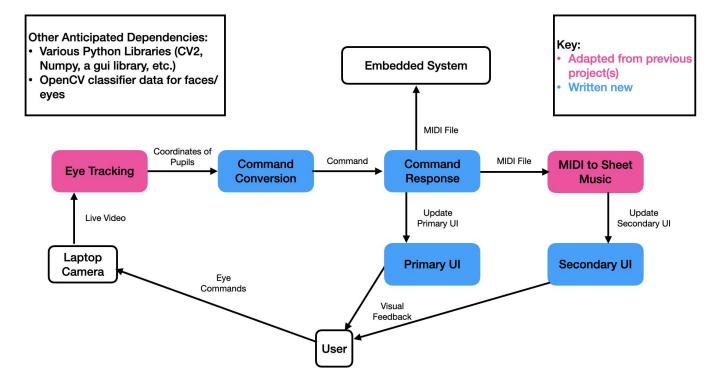
Health, Safety & Welfare:

• Therapeutic Hobby

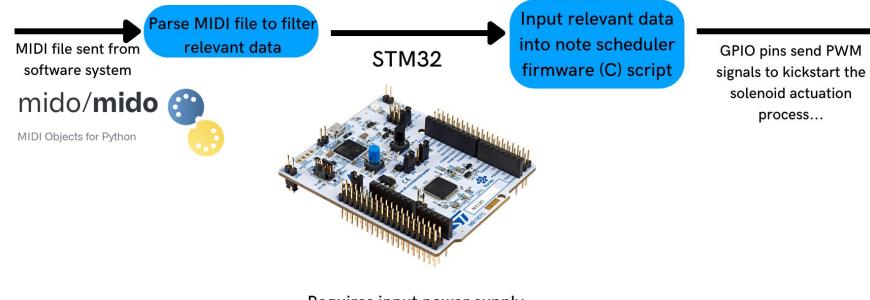
Social Factors:

- Creates an inclusive way to compose music Economic Factors:
 - 3D printed housing
 - Product's software utilizes computer hardware

Block Diagram: Software System

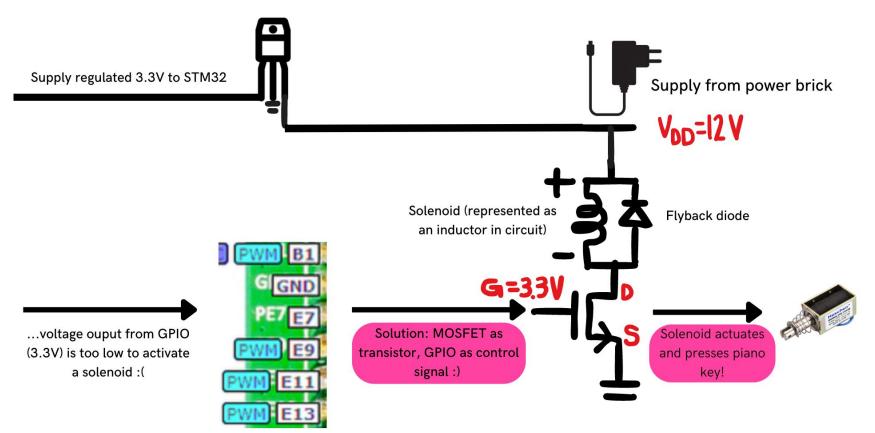


Block Diagram: Processing Path



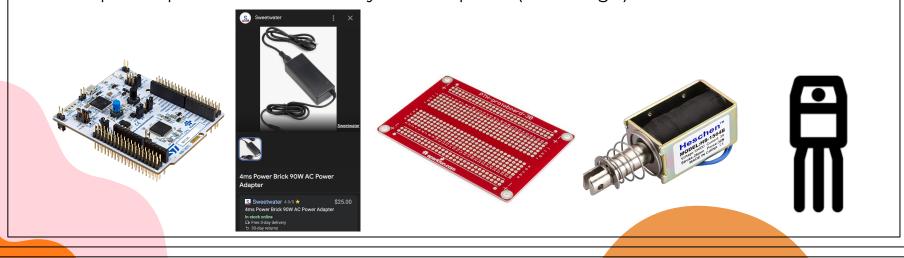
Requires input power supply of 3.3V

Block Diagram: Hardware Execution



Implementation Plan

- STM32 (already own)
- Power brick (purchase)
- Solderable breadboard (already own)
- Push-pull solenoids (purchase, currently experimenting with Adafruit 412)
- Linear voltage regulator (purchase)
- Simple 3D-printed case to hold system on piano (will design)



Formal Testing

• Standardized tests for **accuracy** and **latency**.

- Software
 - Test with multiple users.
 - Test with different patterns of eye-commands.
 - Video-record them performing the patterns for playback.
- Hardware
 - Design (manually) various MIDI input for the solenoid system.
 - Can also test power consumption here.

Informal Testing

- More informal spot-checks throughout semester.
 - Specifically, for eye-tracking.
- To recognize and fix bugs earlier.

Project Management: Upcoming Weeks

Week	Fiona	Shravya	Peter				
6	Code frontend & identify command coordinates	Test power management design Create MIDI conversion firmware	Test solenoid control design Implement eye-tracking for target sections				
7	Create code to map coordinates to commands	Implement UART to transmit data to STM32	Implement individual use calibration for eye-tracking.				
8	Code MIDI command responses Integrate MIDI to sheet music code w/ system	Integrate controllers & solenoids	Design/3D print solenoid case Implement eye-tracking for commands				
9	Code UI command responses	Integrate primary software & controller					

Project Management: The Bigger Picture

Group Member		v1w1v1w1w2w2v2v2v2	w2w2w2w2w2w2w2w3i0/	0/: 0/: 0/: 0/: 0/! 0/! 0/	// 0/10/10/10/10/10/10	10/10/10/10/2	רביו בצו בצו בצו בצו בצו בצו	rzurzJ/20/30/31/·1/:	G1/41/81/81/81/81	1/11/11/11/11/11/11/	11/11/11/11/21/21/21/2	1/21/21/21/21/21/21/21	13 21 21: 21: 21: 21: 2	11 21 21 21 21 21 21 12 12 12 12 12 12
	Deliverables													
All	Design Presentation													
All	Design Report													
All	Interim Demo													
All	Final Slides													
All	Final Report													
All	Final Demo													
All	Final Poster													
All	Final Video													
	Solenoid System - Hardware													
Peter	Research solenoid & solenoid control													
	Research power management for													
Shravya	solenoids													
Shravya	Create schematic for power management													
	Test schematic design of power													
Shravya	management													
Peter	Create schematic of solenoid control													
Peter	Test schemaic design of solenoid control													
All	Design review of schematics													
All	Submit first order of components													
Peter	Design solenoid case													
Peter	3D print solenoid case													
	Solenoid System - Firmware													
Shravya	Create MIDI conversion firmware													
	Eye Tracking Software													
Peter & Fiona	Research OpenCV options for eye-tracking	a												
	Start Implementing eye-tracking	5												
Fiona	software—identify eyes from a live video input													
Peter	Implement eye-tracking for recognizing target sections													
	Implement individual use calibration													
Peter	functionality Implement eye-tracking for recognizing													
Peter	eye-commands													
	Application													
Fiona	Code UI frontend and identify coordinate ranges of commands													
	Create functionality to map coordinates to													
Fiona	commands													
Fiona	Create command responses (UI updates)													
Fiona	Create command responses (MIDI updates)													
	Integrate MIDI to sheet music code with													
Fiona	system													
	Testing													
All	Test eye-tracking accuracy requirements													
All	Test physical accuracy requirements													
AI	Test response time requirements													
All	Test power consumption requirements													
All	Perform full-system integration test (eye-tracking to MIDI to solenoid)													
	Integration													
Shravya	Integrate controller & solenoids													
	Integrate primary device software &							•						
Shravya	controller													
Fiona	Integrate primary & secondary device													
AI	Integrate eye-tracking and MIDI system with STM32 for real-time playback													
	Other													
All	Slack					11111								