

Team C5: Design Proposal

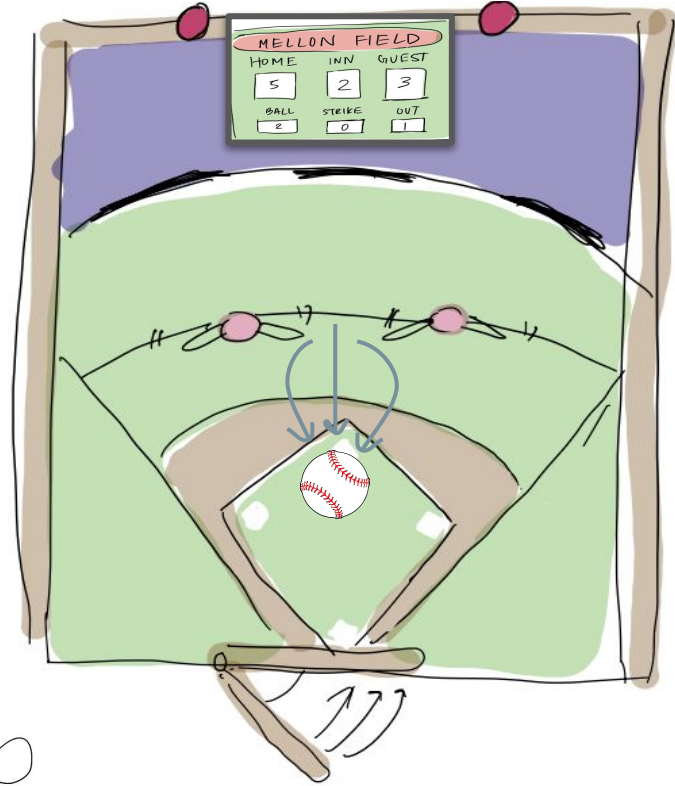
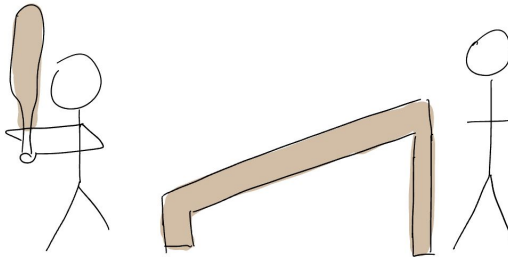
Aiden Magee, **Bing Bhakdibhumi**, Vivian Wang

Two Player Baseball Arcade Game

Modern entertainment is often single-player & screen-bound



We seek to fuse *nostalgic* feel and modern tech to make a social, immersive, and engaging experience for family, friends, or just anyone looking to have fun.



Design Requirements

Social Immersive Engaging

1. **Two-player gameplay.**
2. **Portability:** ~3 ft x 2 ft gameboard. Free, physical plastic bat. <10 min setup time
3. **Gantry motion:** XY, ~3 ft x 2 ft range, ~450 mm/s
4. **Responsiveness:** < 100 ms typical
 - a. Bat swing detection-to-flipper trigger latency (IMU sensing + wireless + actuator command).
 - b. Pitcher input-to-gantry response (UI to motion command).
5. **Gantry-Ball coupling:**
 - a. During pitch: 95% of the time the ball remains coupled to the gantry on the pitch.
 - b. Decoupling: 90% of hits feature a “clean” redirection (no visible interference from pitch carriage mech.)
6. **Pitch repeatability:** >95% two pitches of the same type will land within 5mm of each other.
7. **Player Feedback:** Haptics on bat and audio + LEDs to provide hit and pitch feedback.
8. **Electrical/Mechanical Errors:** <5% over 100 trials. (e.g. flipper not actuating on swing, ball getting stuck, etc.)
9. **Satisfaction:** Average user enjoyability rating of at least 8 out of 10.

Solution Approach

Core gameplay aspects:

- **Novelty** -> swing mechanics, physical gameplay
- Minimal latency for immersive experience
- Flashy visuals

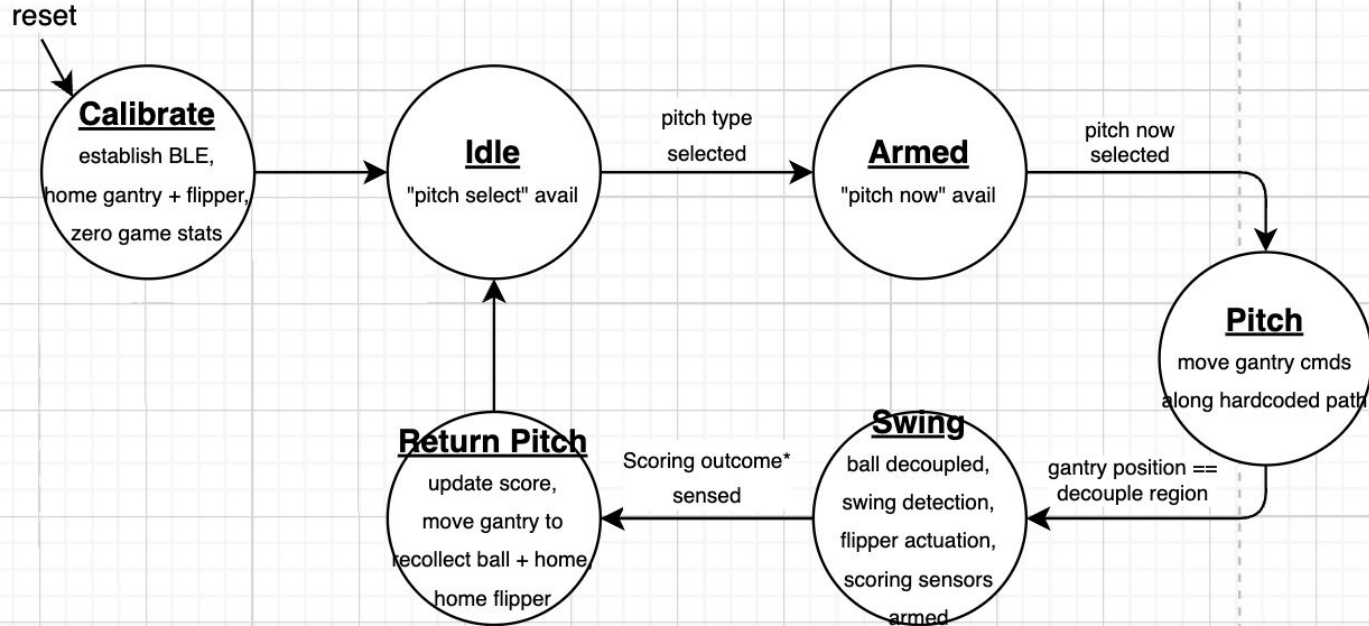


Cultural, **Health**, and **Safety** considerations:

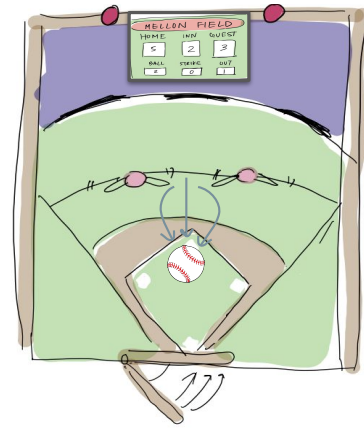
- Recreating a nostalgic feel of mechanical arcade games
- Capturing the joy of swinging a bat
- Being aesthetically pleasing



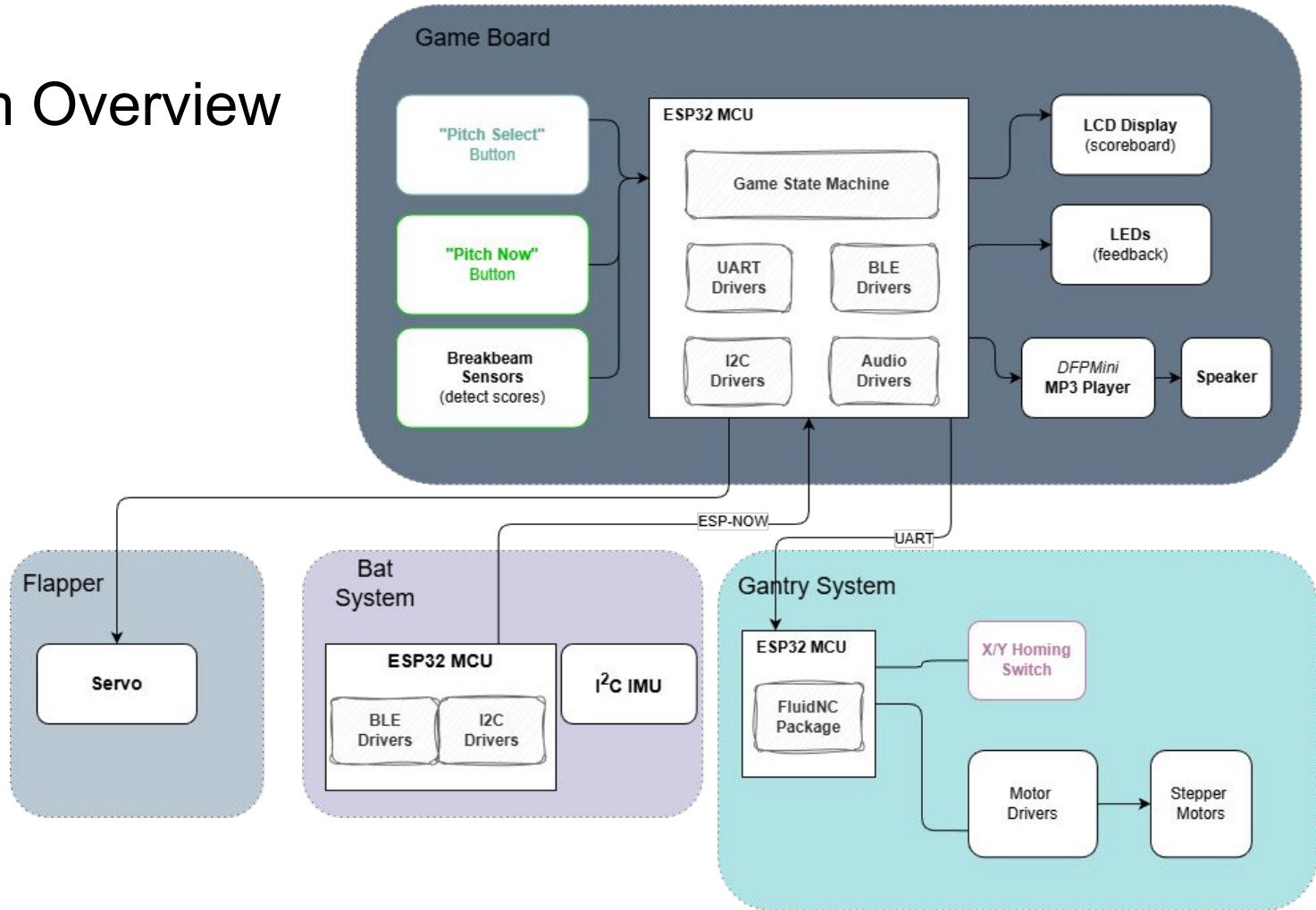
System Overview



*note, out / fouls are grouped as scoring outcomes here



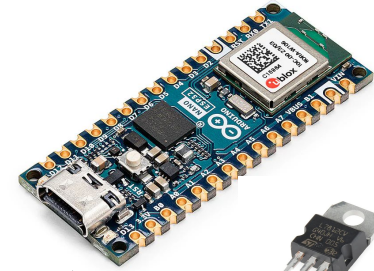
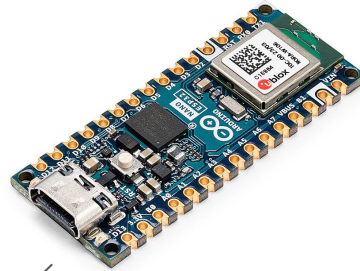
System Overview



Implementation - Paddle/Bat

The bat communicates with another ESP32 microcontroller connected to a servo unit on the game board.

ESP-NOW

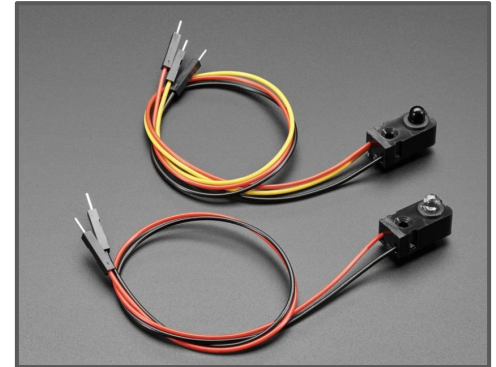
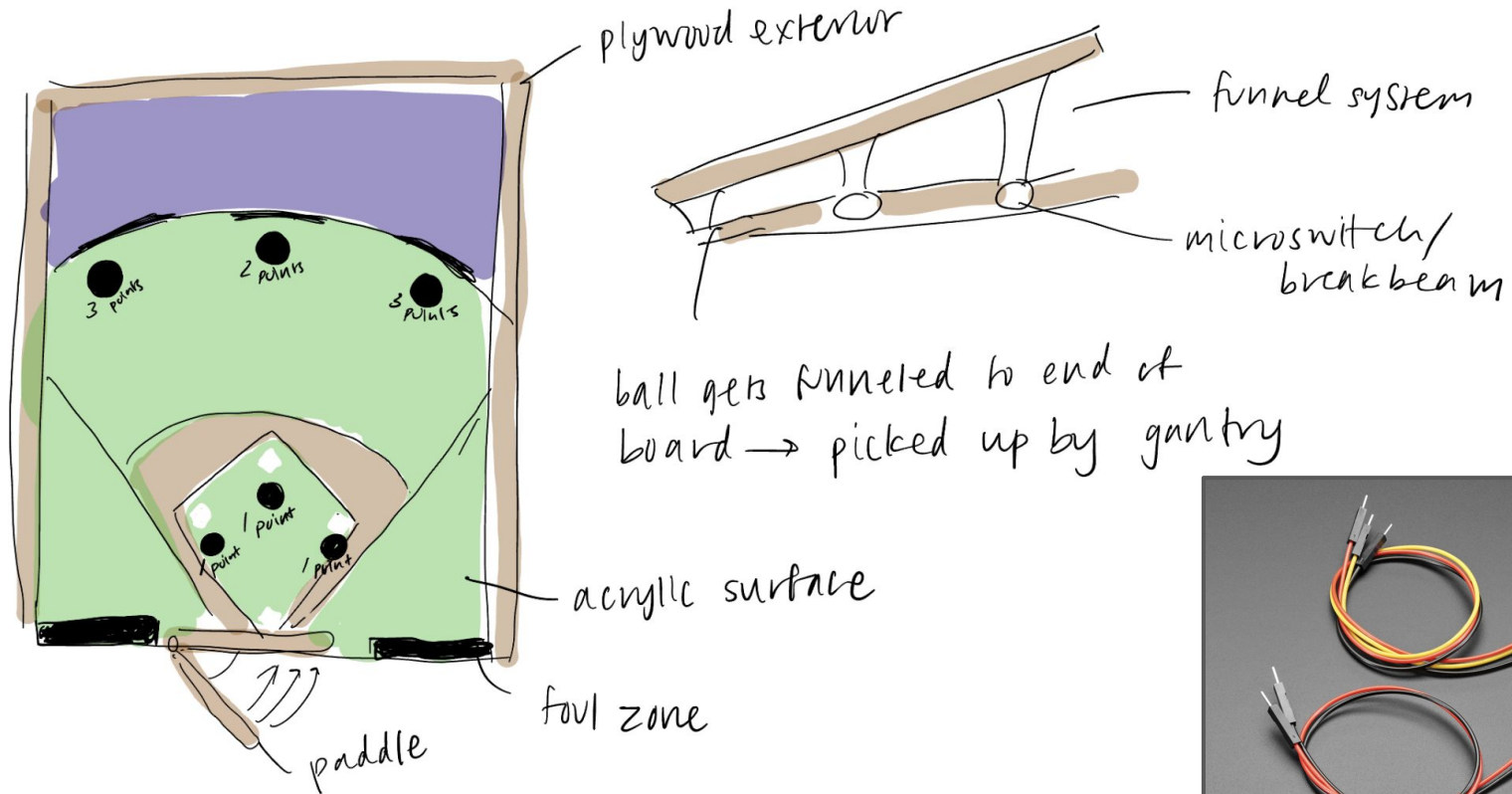


Batting Components

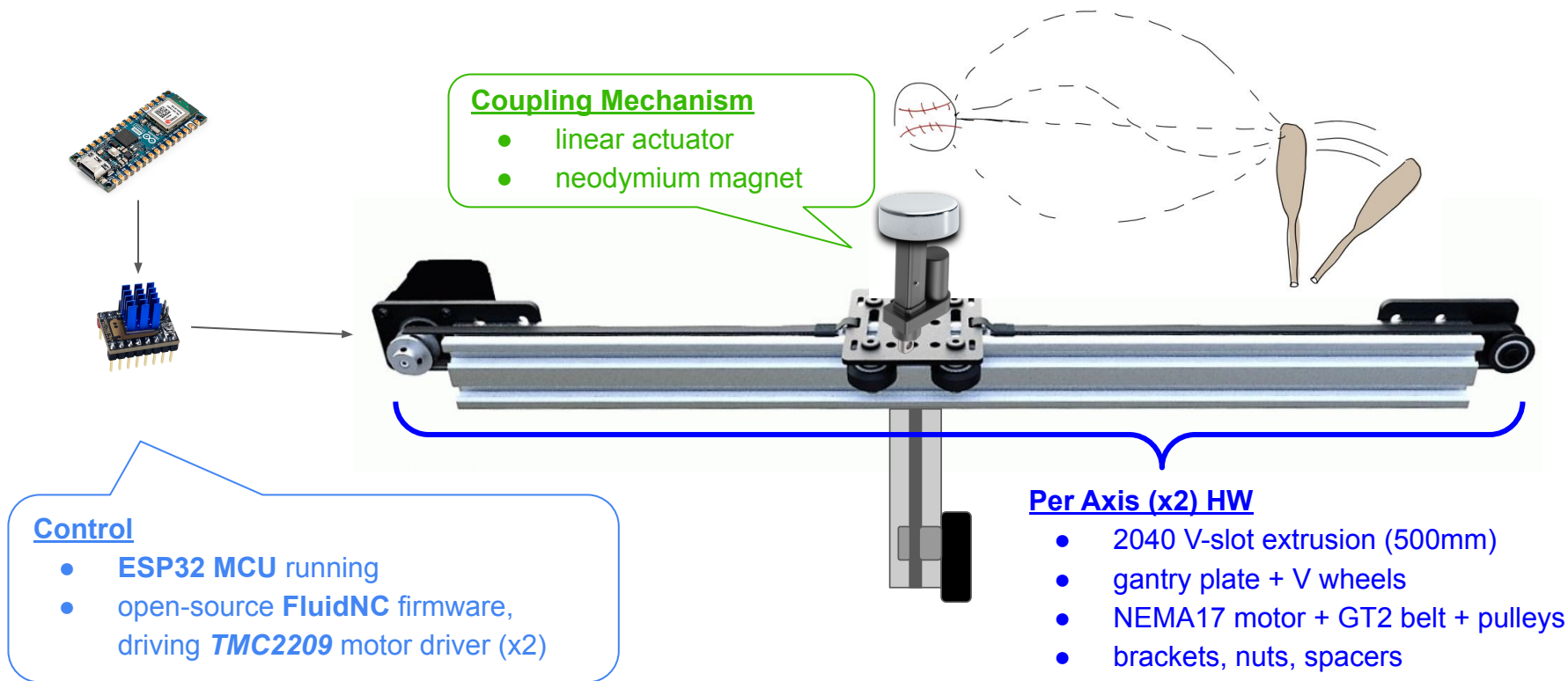
- Arduino Nano ESP32
- BNO085 9-DoF IMU
- 9V battery with regulator
- Plastic baseball bat



Implementation - Game Board



Implementation - Pitch



Potential Technical Concerns

Pitching:

- **Speed of pitch delivery:**
 - If pitches are too slow, the game becomes too predictable.
- **Magnetic decoupling of the ball:**
 - The ball must strictly couple with the gantry during the pitch, and cleanly decouple once hit.

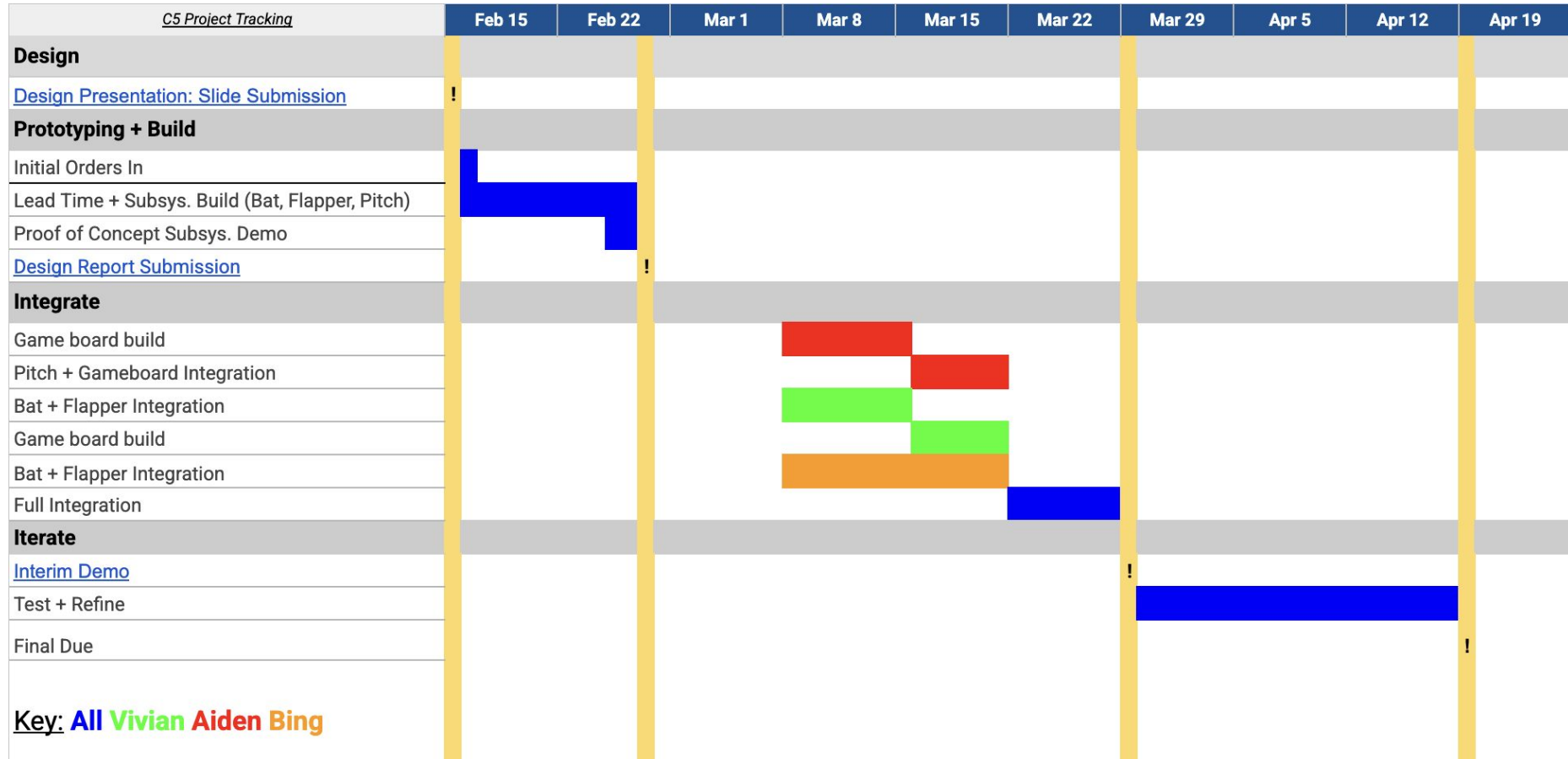
Hitting:

- **Latency of hitting actuation:**
 - The swing of the bat and the actuation of the flapper should be almost seamless.
- **Erroneous swings due to random bat movements:**
 - The flapper should only be triggered by an actual swinging motion.

Testing/Verification

- **Gantry motion accuracy:** Record XY positions and compare error versus a calibrated grid (<5 mm)
- **Coupling tests:** execute 50 pitches of each type at various speeds and record any decouples or slips.
- **Plate decouple tests:** execute 30 plate arrivals of each pitch type (50% hit), recording any noticeable magnetically interfered ball redirection.
- **Scoring accuracy:** route puck into each lane/pocket 20+ times and verify the scoring logic is correct 99% of the time.
- **Gameplay latency:** Pitch/swing actuation is less than 100 ms on average.
- **Fault handling:** induce common failures, such as a stuck ball or missed pitches, and verify timeout and reset behavior.

Project Management



“to fuse *nostalgic* feel and *modern tech* to make a

social, immersive, and engaging

experience for family, friends, or just anyone looking to have *fun*.”

