ARK

SDP18 | Team 21

CDR
3/21/2018

Team

Matteo Bolognese, EE
Hardware Guru
Device Design, Assembly
PCB Design, & Power

Chad Klinefelter, CSE
Backend Guru
Data Management &
App Development

Jackie Lagasse, CSE
UI/UX Guru
Augmented Reality &
App Development

Ethan Miller, EE
Algorithms Guru
Device Communication &
Sensor Interfacing

Problem Statement

- AR users are unable to incorporate extremities into an interactive application unless their device camera can directly face them
- There are no inertial sensing products for extremity tracking that can be used in AR applications

Solution: a separate foot attachment

- Sensor on foot can transmit data to phone, where it can then be acted upon

Our Vision

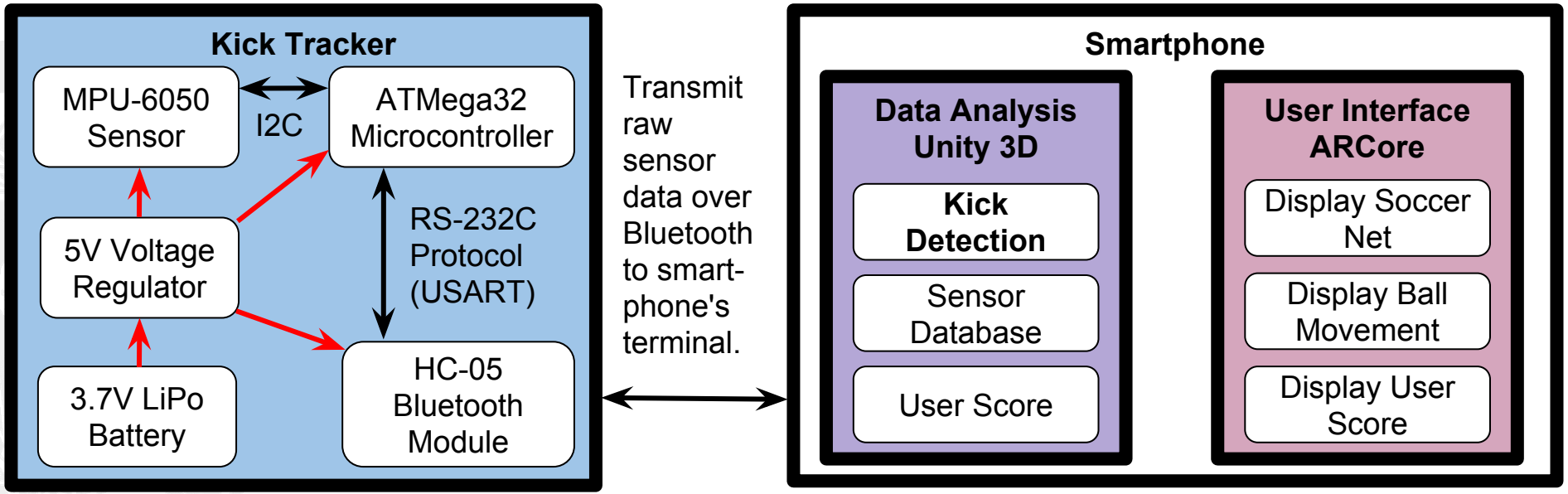
On demo day we plan to present the following:

- Bring user to an open space
- User wears headset and kick tracker
- Start ARK app on phone
- App displays virtual soccer ball and goal
- User kicks foot, observes movement of soccer ball

System Requirements

Requirement	Status
App connects to Bluetooth and begins game in less than 10 seconds	Met
App must determine user's kick speed and direction	Not met for all kick types
Ball must move with speed and direction proportional to user's foot	Met - Unity maps input vectors to ball as a rigid body
System delay < 300ms, ideally <100ms	Met - 18.7 Hz from sensor to Unity, 60 Hz refresh, 5 frames = 83ms delay
Maximum dimensions of device: 4 x 3 x 2 inches	Met in total size - Next iteration is much smaller
Maximum weight: 1 lb	Met - Weight 4.6oz
Minimum battery life: 5 hours	Met - Sources ~65mA during operation = 46hr of battery life

Block Diagram



All components in kick tracker are connected together in PCB.
Battery is rechargeable.

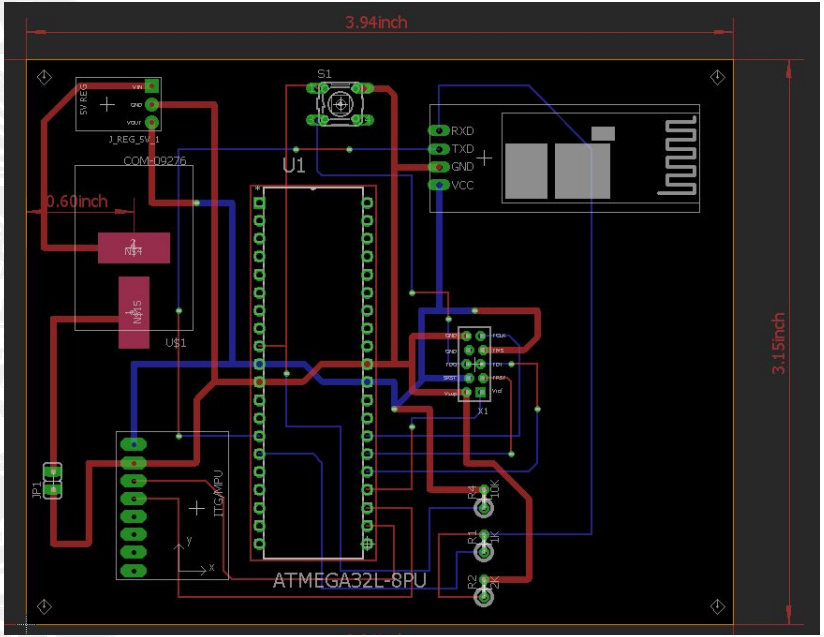
User data consists of user's score in this play through.

CDR Deliverables Status

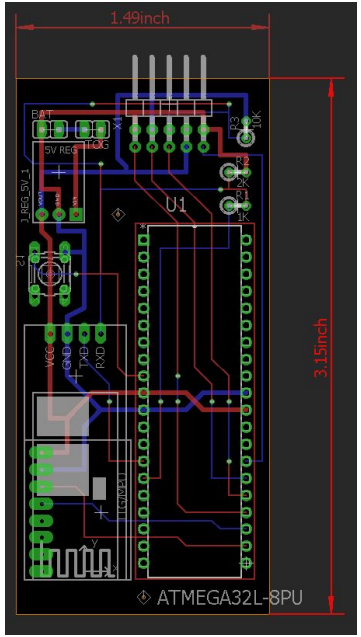
Deliverable	Member	Status	What's Next?
Design PCB, send for fabrication	Matteo	Completed	Revamp PCB for more efficient layout
Design foot mount structure and enclosure	Matteo	Completed	Revamp enclosure for PCB redesign
Backend of app must store sensor data	Chad	Completed	Refine data storage as kick model is finalized
Model data to analyze foot movement & perform kick accuracy measurement	Ethan	In Progress	Develop additional kick models
Augmented reality must show soccer goal and soccer ball	Jackie	Completed	Improve user interface, implement left / right foot calibration

PCB Layout

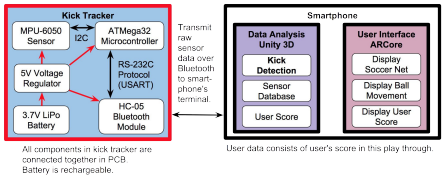
Old



New

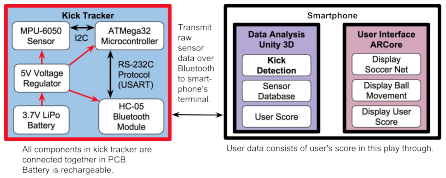


- First PCB design is complete and works properly
- New PCB layout is almost ⅓ size of current board
- Same width, significantly smaller length
- Better fit for shoes



Enclosure

- Current Dimensions:
  - Length - 3.75”
  - Width - 4.5”
  - Height - 1.5”
  - Thickness - 0.125”
- Material: ABS plastic
- New Case
  - ~⅓ size of current case (will fit smaller PCB)
  - Improved integrated strap holes

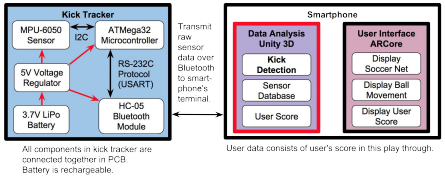


Sensor Data / Storage

- Caching recent most ~500 ms (9 samples) of data
  - Get threshold with reduced noise
- User delay: 83 ms
  - Data rate: 18.7 Hz (data update every 54 ms)
  - Frame rate: 60 Hz (frame update every 17 ms)
  - *Delay:* 4 frames for data arrival + 1 frame for processing = 83 ms

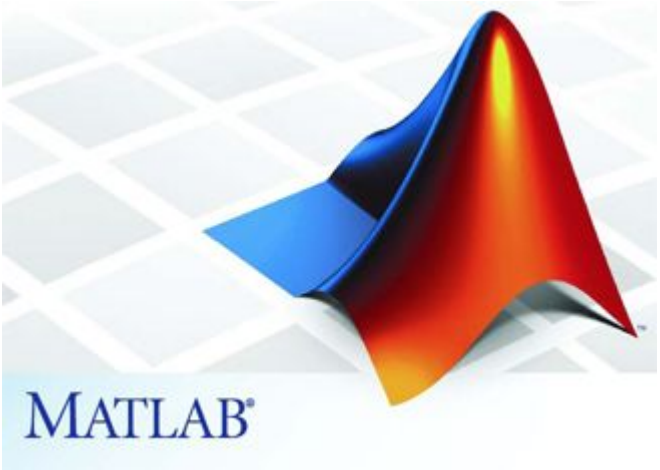
```
03-07 01:31:04.841  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
03-07 01:31:04.841  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : ARK LOG ********** Data: -0.10  0.02  1.19  -2.13  -0.18  -1.34
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : ARK LOG ********** Data: -0.11  0.01  1.18  -2.07  -0.18  -1.16
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : ARK LOG ********** Data: -0.11  0.02  1.18  -2.20  -0.24  -1.34
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : ARK LOG ********** Data: -0.11  0.02  1.17  -2.13  0.06  -1.34
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : ARK LOG ********** Data: -0.10  0.02  1.19  -2.01  -0.06  -1.34
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
```

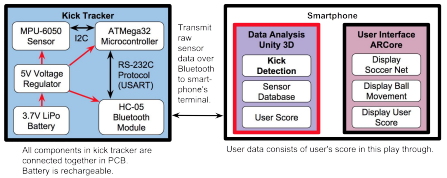
⇐ **Reading data samples from temp queue (in file)**



Kick Detection

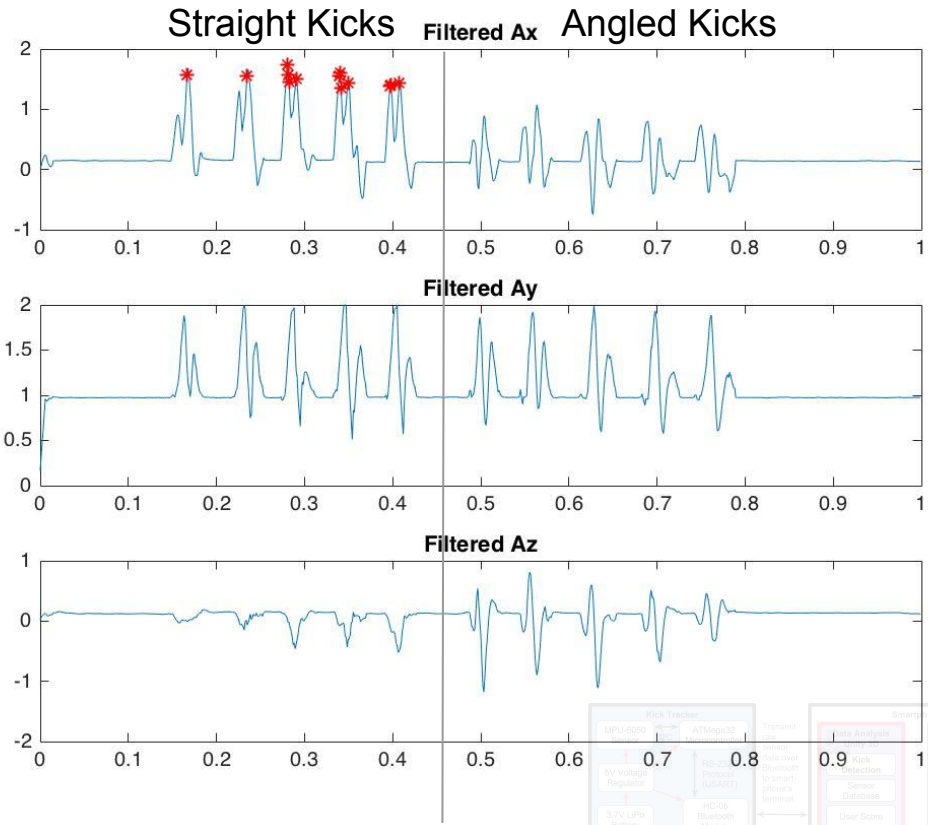
- Currently implement a threshold based kick detection for straight kicks
- Threshold is combination of all axes being above/below a certain point
- Will implement more kick detection algorithms using Matlab to extract features from time series





Current Measurements

- Raw time series is convolved with a window of the last 6 samples to reduce noise and peakiness
- Given a threshold for **each axis**, MATLAB scans the timeseries for all indices where a condition is met
  - In the case of **forward, straight** kicks (shown right), the established thresholds are >1.35, >1.25, and <0.3g for the X, Y, and Z axes respectively
  - Sometimes condition looks for above a threshold, sometimes below
- MATLAB then determines the intersection of all indices for each axis. If all 3 sets share an index, and event is determined to have occured.
- Shown right: 5 straight, forward kicks and 5 angled kicks (instep out)



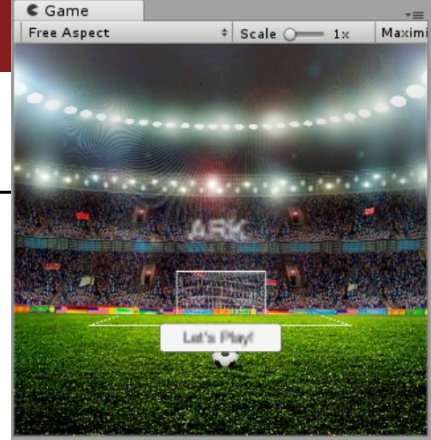
App Developments

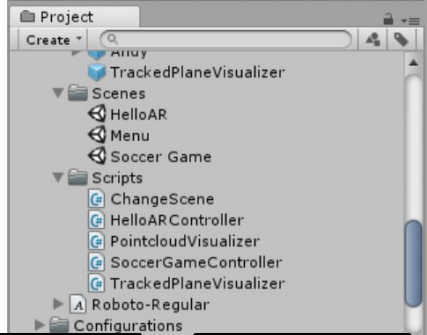
ARCore / Augmented Reality:

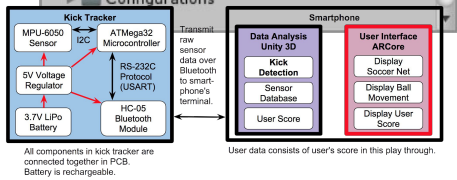
- ARCore finds a surface, spawns soccer goal and ball onto surface
- Stereoscopic (side by side camera) imaging adds depth when using the AR headset

Unity / Game Design:

- Use transparent surface to constrain ball movement
- Soccer ball moves based on sensor input
- Soccer ball is respawned on out of bounds, goal scored
- If a goal is scored, the current score is incremented







Demo

Budget & Cost Analysis: Production Cost

Items Per Device	#	Bulk Cost at Bulk Quantity	Distributor
ATMega32	1	$4.80 @ 100	Digikey
HC-05 Bluetooth	1	$2.75 @ 1	Alibaba
MPU6050 Inertial Sensor	1	$4.07 @ 1000	Digikey
Voltage Regulator	1	$3.49 @ 100	Pololu
Battery	1	$3.25 @ 4	Amazon
PCB & Case	1	$6.00	Estimation
Starlight Headset	1	$13.99 @ 1	Amazon
Total:		**$38.05**	

1. Device is only $24.06 if user has their own headset.

2. We have approximately $100 left of our $500 total prototyping budget.

FPR Deliverables

- **UI/UX:** Update UI to implement left and right foot calibration (**Jackie**)
- **Backend:** Finish implementing sensor data storage system (**Chad**)
- **Algorithms:** Implement algorithms for at least 2 different kick types (**Ethan**)
- **Hardware:** Assemble final case with new PCB (**Matteo**)

Team Member	Color Code
Matteo	
Jackie	
Chad	
Ethan	

Gantt Chart

		Start	Bench			Bench	CDR		Break	Bench			FDR	
Assignee	**Task**	**Jan. 22**	**Jan. 29**	**Feb. 5**	**Feb. 12**	**Feb. 19**	**Feb. 26**	**Mar. 5**	**Mar. 12**	**Mar. 19**	**Mar. 26**	**Apr. 2**	**Apr. 9**	**Apr. 16**
Matteo	Design, Order PCB													
Jackie	Develop App, Soccer Goal													
Chad	Develop App, Backend													
Ethan	Integrate Sensor w/Backend													
Matteo	Design Foot Mount, 3D Print													
Jackie	Develop App, Soccer Ball													
Chad	Develop App, UI													
Ethan	Assess Accuracy of App													
Matteo	Assemble PCB, Foot Mount													
Jackie	Develop App, Motion Accuracy													
Chad	Develop App, Motion Accuracy													
Matteo	Assess Accuracy of App													
All	Final Integration of System													

Thank You!

Questions?