

# ACTIBLES: DESIGN AND DEVELOPMENT OF A NEW PLATFORM FOR ACTIVE TANGIBLES

by

Brien G. East

Bachelor of Engineering, Ryerson, 2013

A thesis

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Masters of Applied Science

in the Program of

Electrical and Computer Engineering

Toronto, Ontario, Canada, 2017

©Brien G. East 2017



## **AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS**

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.





Actibles: Design and Development of a New Platform for Active Tangibles

Masters of Applied Science 2017

Brien G. East

Electrical and Computer Engineering

Ryerson University

## **Abstract**

In this thesis, we present a new kind of active tangible call an Actible. Actibles are an open source hardware/software platform for designing and implementing active tangibles for Tangible Embodied Interaction (TEI) applications. Web technologies and a smartwatch core are leveraged for ease of development, and enables the active tangibles to act coupled to a server, or act as their own server. This kind of connectivity enables the use of the Actibles independently or in combination with other devices and displays ubiquitously. We derived an expanded set of input and output interactions based on previous work on active tangibles, including tilting, shaking, neighboring, stacking, on-screen gestures, and integrated LED feedback. We describe the design and technical implementation of these interactions, and demonstrate it's use in a multitude of published example applications.



## Acknowledgements

I would like to thank Dr. Ali Mazalek and Dr. Dimitri Androutsos for their supervision and guidance through this process.

Secondly I would like to thank Dr. Jason Nolan and Dr. Jennifer Mactavish for their help and support in helping me get this degree finished.

Thirdly, I would like to thank Roozbeh Manshaei, Sean DeLong and Ahmed Arif for their contribution to the development, testing and employment of the Actible platform in their research.



## Dedication

I would like to dedicate this work to my Granddad, Maurice East. From humble beginnings, he went on to serve his country, conquer the skies, sail the seas and tame the Canadian North. He was intelligent, adventurous and entrepreneurial. He was a real gentleman and a true Canadian. He was an inspiration to everyone he encountered, and most certainly continues to be an inspiration to me.



Maurice East (March 5th 1924 - September 10th 2017)



# Contents

<i>Declaration</i> . . . . .	iii
<i>Abstract</i> . . . . .	v
<i>Acknowledgements</i> . . . . .	vii
<i>Dedication</i> . . . . .	ix
<i>List of Tables</i> . . . . .	xiii
<i>List of Figures</i> . . . . .	xv
<i>List of Appendices</i> . . . . .	xvii
<b>1 Introduction</b>	<b>1</b>
1.1 What are Actibles? . . . . .	1
1.2 Purpose and Scope of this Thesis . . . . .	3
1.3 Contributions . . . . .	3
1.4 Overview of this Thesis . . . . .	3
<b>2 Related Work</b>	<b>5</b>
2.1 Active Tangible Frameworks . . . . .	5
2.2 Active Tangible Platforms . . . . .	6
2.3 Tangible/Tabletop Sensing Techniques . . . . .	6
2.4 Active Tangible Applications . . . . .	7
<b>3 The Actibles Platform</b>	<b>9</b>
3.1 Design Goals . . . . .	9
3.2 Interactions . . . . .	10
<b>4 Technical Implementation</b>	<b>13</b>
4.1 Design Process and Methodology . . . . .	13
4.1.1 Smartphones as Active Tangibles . . . . .	13
4.1.2 NFC for Stacking Detection . . . . .	13
4.1.3 The Sparse Tangible Active Tangible . . . . .	14
4.1.4 Implementation of the Actible Platform . . . . .	14
4.2 Physical Design . . . . .	15

4.3	Hardware . . . . .	15
4.4	Software . . . . .	19
<b>5</b>	<b>Application Examples</b>	<b>21</b>
5.1	Active Pathways . . . . .	21
5.2	Sparse Tangibles . . . . .	23
5.3	Tangible mtDNA . . . . .	24
<b>6</b>	<b>Evaluation</b>	<b>27</b>
6.1	Evaluative Comparison . . . . .	27
6.1.1	Smartphones and Smartwatches . . . . .	27
6.1.2	Sifteo Cubes . . . . .	27
6.2	Performance Specifications . . . . .	29
<b>7</b>	<b>Discussion</b>	<b>31</b>
<b>8</b>	<b>Conclusion</b>	<b>33</b>
8.1	Future Work . . . . .	33
	<b>References</b>	<b>50</b>



# List of Tables

6.1	Performance and Interaction Comparison . . . . .	29
A.1	Bill of Materials . . . . .	43



# List of Figures

1.1	Actible . . . . .	2
3.1	The Tilt Interaction . . . . .	10
3.2	Grouped Interactions. . . . .	11
4.1	Sparse Tangible Active Tangible . . . . .	14
4.2	A Field Striped Actible . . . . .	15
4.3	Actible Exploded Assembly . . . . .	16
4.4	Major Electrical Components of an Actible . . . . .	17
5.1	Actibles in Active Pathways . . . . .	22
5.2	Actibles in Sparse Tangibles . . . . .	23
5.3	Actibles in mtDNA . . . . .	25
A.1	PCB Layout Design . . . . .	41



# List of Appendices

<b>Appendix A</b>	<b>Design Documentation</b>	<b>35</b>
A.1	Electrical Schematics . . . . .	36
A.1.1	Board Design . . . . .	41
A.1.2	Electrical Bill of Materials . . . . .	42
A.2	Mechanical Design . . . . .	44



# Chapter 1

## Introduction

### 1.1 What are Actibles?

Tangible, embedded and embodied interaction (TEI) is a growing area of human-computer interaction (HCI) research that employs physical objects, often referred to as "tangibles", as both controls and representations for underlying digital information and functions [1]. A large amount of research on tangible interaction employs passive (i.e., unpowered) tangibles that are tracked on display surfaces using fiducial markers and computer vision (e.g., [2]). More recently, there has been a growing interest in the use of active (i.e., powered) tangibles due to the additional interactive capabilities they provide. Drawing on previous work on active tokens [3, 4, 5, 6] and active tangibles [7], we define active tangibles as physical input-feedback devices that can be used independently or in conjunction with other hardware or software to tangibly augment interaction with integrated displays, sensing, or actuation capabilities. Applications that involve active tangibles include data querying or browsing (e.g., [7, 8, 9]), media content editing (e.g., [10]), scientific discovery (e.g., [7, 11, 12]), and other computational tasks. A particular benefit of active tangibles is the possibility for dynamic reconfiguration, allowing users to modify their associations to different data or functions over time [6].

Developing custom hardware and software to support active tangibles is a task that is undertaken by many TEI researchers, because of active tangible's usefulness and strengths. This development distracts researchers away from their main objective of designing the application and experimenting with relevant interactions. Consequentially, numerous researchers in the TEI community have utilized available or previously available off-the-shelf devices such as Sifteo Cubes or smartphones as active tangibles for use in research applications [7, 13, 14, 15, 16]. However smartphones have their limitations as active tangibles due to the limited interaction set and form factor, and Sifteo Cubes are no longer produced. Although inventories of Sifteo Cubes may be found, they are no longer supported. Further, Sifteo Cubes have various technical limitations when being used in TEI applications. Firstly, Sifteo Cubes are coupled with a custom USB base, and custom firmware must be implemented to enable bi-directional communication with a server. Bi-directional communication is an important requirement for TEI applications.



Figure 1.1: Actibles are a smartwatch-based active tangibles platform that incorporates additional sensors to detect stacking and neighboring, as well as LED feedback. The image shows an assembled Actible.

Secondly, Development on Sifteo Cubes is only supported in C++. This results in the maintenance of native codebases for every combination of bi-directional communication between displays and computers. Thirdly, despite Sifteo Cubes use in many TEI applications, it has limited display capabilities and no support for multi-touch or on-screen interactions, thereby limiting their usefulness as a general platform for experimenting with active tangibles in research.

We present a new kind of active tangible call an Actible. Actibles are an open source hardware/software platform for designing and implementing active tangibles for Tangible Embodied Interaction (TEI) applications (an Actible is shown in Figure 3.1). Web technologies and a smartwatch core are leveraged for ease of development, and enables the active tangibles to act coupled to a server. This allows the tangibles to be used independently or in combination with other devices and displays. An expanded set of input/output interactions are provided, including tilting, shaking, neighboring, stacking, on-screen gestures, and integrated LED feedback. As the platform is open source<sup>1</sup>, we enable designers to create their own cases, offering the possibility to integrate their own smartwatch and/or extend the platform with additional sensing capabilities. We contribute to the development of robust applications with active tangibles while reducing the development time of TEI projects, since more time can be spent developing applications, with less time spent developing hardware and the accompanying software.

In the following section we provide a review of related work. We then describe the design and

---

<sup>1</sup>Complete schematics and bill of materials and case design are available in Appendix A



technical implementation of Actibles, followed by several example applications. We conclude with a discussion and directions for future work.

## 1.2 Purpose and Scope of this Thesis

The benefit of intuitive tangible devices are apparent as surface computers become commonplace and are enlisted to solve more complex problems. The work presented in this thesis focuses on the necessity, benefit and development of an active tangible for use with surface computers with a variety of different example applications. The resulting device is an example of the types of interactions and features that are needed to achieve a level of intuition and fluidity required in the design of active tangibles.

## 1.3 Contributions

This work builds on previous frameworks and examples of active tangible development. This work proposes a set of design goals and interaction capabilities that we believe are required for a successful active tangible. An example of an active tangible was designed and tested in a variety of applications. This design was also made available to various researchers in the TEI field.

The work carried out by the author was primarily on the hardware development. All circuitry, case design, sensing technology implementation was solely completed by the author.

The author also participated in the development of various applications that utilized the Actible platform. These collaborative roles varied from assistance with hardware/software integration and assistance with user studies.

The following papers were published/co-published by the author on the Actible platform and its applications:

- Actibles: Open Source Active Tangibles [17]
- Tangible mtDNA: A Tangible Tabletop System for Exploring Genetic Mutations on Mitochondrial DNA Cancer Data [18]
- Extending the Design Space of Tangible Objects via Low-Resolution Edge Displays [19]
- Sparse Tangibles: Collaborative Exploration of Gene Networks using Active Tangibles and Interactive Tabletops [7]
- Exploring Genetic Mutations on Mitochondrial DNA Cancer Data with Interactive Tabletops and Active Tangibles [20]

## 1.4 Overview of this Thesis

The remainder of this thesis is outlined as follows: Chapter 2 reviews the latest relevant work in active tangible development; Chapter 3 outlines the necessary design goals and interaction capabilities that are

necessary for an active tangible we called an 'Actible'; Chapter 4 discusses the technical implementation of Chapter 3 in terms of physical design, hardware and software; Chapter 5 showcases three published application examples involving Actibles; Chapter 6 is a evaluative comparison and technical specification of the Actible; Chapter 7 is a discussion of the design and applications and finally Chapter 8 is the conclusion and projected future work on the Actible platform.

## Chapter 2

# Related Work

We review previous work on the conceptual positioning of active tangibles, general platforms for active tangibles, explorations of tangible/tabletop sensing techniques, and active tangible based applications.

### 2.1 Active Tangible Frameworks

Several researchers have proposed theoretical frameworks with a particular focus on active tangibles to assist in the design and development of tangible-based systems. Ullmer et al. [5] proposed a "token+constraint" framework that described a class of tangible interfaces that build on relationships between systems of physical tokens and constraints. They defined tokens as "discrete, spatially re-configurable physical objects that typically represent digital information" and constraints as "confining regions within which tokens can be placed". Similarly, Shaer et al. [4] noted that tangible interfaces can be described as a "set of relationships between physical objects and digital information." Their Token and Constraints (TAC) paradigm describes tangible interfaces in terms the relationships between tokens as graspable physical objects in a tangible system, constraints which limit their behavior, and variables as pieces of digital information or computational functions in an application. Both frameworks highlight the way in which physical manipulation of objects, possibly in combination with constraints such as a containing surface, can express digital syntax [14].

Other researchers have extended the active tangibles design space to provide frameworks or guidelines for active tangibles within specific application areas. For example, Valdes et al. [6] explored the use of gestural interaction with active tangibles for manipulating large data sets. They studied user expectations of a hybrid tangible and gestural language engaging this space. Based on their findings, they proposed a framework of user-defined gestures for interaction with active tangibles, including off-surface gestures performed in air, on-bezel, and while hovering over-surface. Pillias et al. [15] explored the use of Sifteo Cubes in different gaming contexts. Based on their findings, they provided a framework that game designers can use to exploit Sifteo Cubes' physical properties to transfer some of its game mechanics into the real world. For example, if a game requires a virtual world translation or rotation for its completion, Sifteo Cubes can be physically manipulated to complete the game, and thus they extend

the virtual world into the tangible one.

This prior work provides a basis for understanding the kinds of interactions that need to be detected and supported by any general-purpose active tangibles platform. These include: 1. physical manipulations on a surface or constraint, such as placement, translation, rotation, stacking, neighboring and removal; 2. physical manipulations independent of any constraints, such as neighboring and stacking; and 3. gestural manipulations, such as tilting and shaking.

## 2.2 Active Tangible Platforms

Current open platforms for active tangibles provide some, but not all of the interactive capabilities offered by the Actibles platform. Merrill et al. [21] developed Siftables, later commercialized as Sifteo Cubes. These compact cube-shaped devices include sensing, graphical display, and wireless communication capabilities via an accelerometer and IrDA transceivers. They can be physically manipulated, either individually or as a group, to interact with digital information and media. Sifteo Cubes can detect shaking, tilting, neighboring, and tapping, but do not support multi-touch or stacking interactions.

Le Goc et al. [22] developed miniature active tangibles called SmartTokens that are 28 mm in diameter. Their intention was to provide an open platform for embedding basic sensing capabilities within a small form-factor. SmartTokens support touch, motion sensing, and wireless communication with a coordinator, but do not include a graphical display. They also cannot detect shaking, flipping, tilting, stacking, or neighboring.

Mazalek and Arif [14] proposed the use of mobile devices as active tangibles. They argued that since most current mobile devices are touchscreen-based and come with a wide range of inertial sensors, they can be used as active tangibles in conjunction with interactive tabletops. In a tutorial, they demonstrated that these devices can detect shaking and tilting using the inertial sensors, multi-touch on the touchscreen, and neighboring when placed on the table. Their platform also does not support stacking, and must be on the table to detect adjacency.

## 2.3 Tangible/Tabletop Sensing Techniques

Researchers have been exploring different sensing approaches for tangible interactions with interactive tabletops. While typically not active (i.e., powered), several examples are worth mentioning because they use alternate methods to provide similar capabilities as some active tangibles, such as the display of information on the tangibles and stacking interactions.

Rekimoto et al. [23] developed DataTiles that are transparent acrylic square objects with embedded RFID tags. They used these tangibles in conjunction with a tabletop interface, where each tile represented an application-specific function and placing it on the table triggered its associated function. The interaction is constrained to a horizontal surface on which the tiles must be placed within predefined grid cells, but the system provides an expressive physical language for retrieving and manipulating digital content, and leverages the underlying display surface to display information on the tiles themselves.

In more recent work, Baudisch et al. [24] developed Lumino, tangible building blocks that allow interactive tables to track their 3D arrangements. This is achieved by augmenting each Lumino with a unique fiducial marker and a glass fiber bundle. When placed on a table, the glass fiber bundle can optically guide the light reflected off the markers in the higher levels down to the table for the system to identify their arrangements. Chan et al. [25] designed three different types of capacitive tangibles, CapStones, Zebra Dials and Zebra Sliders. Each consists of multiple parts, such as stackable blocks and tangible widgets with moving parts. These tangibles use an internal system of wires and connectors that causes the capacitance of the bottom-most block to reflect the entire assembly. While these tangibles can work with current consumer hardware, their functionality is limited to touch and/or three-dimensional arrangements (including stacking), dialing, and sliding. Höfler et al. [26] also explored stacking interactions for tabletop tangibles using fiducial marker based tracking on the table and a depth-sensing camera (Kinect) to detect stacks above the surface. They demonstrated their approach with a tower defense game on a tabletop, where stacking multiple cylinders increased the strength of the tower. In all of these examples, the tangibles function only when placed on the surface. They are not powered and do not contain sensors that would enable them to work independently of the tracking surface.

## 2.4 Active Tangible Applications

A number of researchers have appropriated Sifteo Cubes as active tangibles in their systems. For example, Grote et al. [11] developed a tangible tabletop interface called Eugenie to assist synthetic biologists in exploring large datasets through the construction of complex queries using Sifteo Cubes. They extended the interaction space of the Sifteo Cubes by building custom cases augmented with connectors on the left, right, and top. This allowed the cubes to interlock both horizontally and vertically, supporting stacking. A similar system called SynFlo [12] uses Sifteo Cubes to enable a playful bio-design activity to create digital biological creations. It uses Sifteo Cubes in groups of three, representing a collection of genes, a plasmid, and a bacterial cell. Rangoni et al. [16], on the other hand, used Sifteo Cubes as different widgets, such as zoom, writer, link, and stamp, etc., in a tabletop application for collaborative business modeling.

Many other researchers have designed novel active tangibles to fit the needs of a particular application or scenario. For example, Ullmer et al. [9] created two kinds of custom active tangibles, parameter wheels and parameter bars, for querying large databases, such as real-estate data, which is visualized on an adjacent display. The tangibles are tracked by and manipulated on custom physical constraints, which limits their portability. Zigelbaum et al. [10] encased handheld computers into custom designed plastic cases that resembled jigsaw pieces. Each tangible jigsaw piece represented a video clip, and these could be connected together in a sequence in order to construct video sequences consisting of multiple clips. While the jigsaw-puzzle form factor of the tangible pieces provided a clear physical affordance that suggested the ability to connect individual pieces together, it also limited the use of the tangible pieces for other purposes.

More recent examples of custom active tangibles include the work of Klum et al. [8] on Stackables,

which are active tangibles with a display and two control wheels. The researchers used Stackables in a query interface in which each tangible represented a query parameter. The system also enabled complex query construction by stacking multiple tangibles representing different query parameters. Interaction with these tangibles is limited to using the two control wheels. In another project, Arif et al. [7] used LG G Watch smartwatches as active tangibles in a tabletop system that allowed users to explore gene and protein networks. They built a custom case for the smartwatches for better usability, assimilation with the system, and to extend their interaction space. The cases employ spring-loaded pins and targets on the top and the bottom, allowing the smartwatches to communicate with each other when stacked. The cases also include arrays of LEDs surrounding the frames.

Some research on custom active tangibles has also explored actuation in the form of physical movement. For example, Pedersen and Hornbaek [27] designed Tangible Bots, motorized tangibles capable of moving and rotating on interactive tabletops. These devices support direct, indirect, individual, and group interactions, and can provide haptic feedback on correct and incorrect actions by simulating attraction and repulsion, respectively. While these devices can ignite novel ways of interactions, their functionality is limited to tabletops due to their bulky design and absence of a graphical display. Overall, these custom active tangibles provided affordances in the context of their corresponding applications, yet their functionality is usually not easily extensible to other systems and applications.

## Chapter 3

# The Actibles Platform

### 3.1 Design Goals

Based on prior research described in the previous section as well as our own past work with custom active tangibles, we formulated a set of design goals for a general-purpose active tangible platform. As the platform is open source, our design goals focus on ease of use and customization in order to enable designers to extend the platform’s capabilities and applications.

*G1: Easy transition from existing to new tangibles.* Many existing applications use custom or commercial tangibles. To ensure that the new active tangibles could be made to work with most of these existing systems, they must provide support for the most common tangible interactions, including shaking, flipping, tilting, neighboring, and tapping, as well as an integrated display.

*G2: Inspire new interaction approaches.* One of our aims is to inspire designers and developers to explore novel interaction approaches within their applications. Augmenting the device with a touchscreen, a tactile button, and a variety of sensors provides a rich set of potential data sources when developing custom interactions. Designers can combine data from the sensors, touchscreen, and tactile buttons to design new interaction techniques for the tangibles.

*G3: Accommodate individual and group interactions.* The new active tangibles must provide support for individual and group interactions for a complete tangible user experience. An individual interaction involves using one tangible to perform one operation, while a group interaction involves using multiple tangibles, such as stacking, bumping, etc.

*G4: Easy development.* It is important to provide a simple software development kit to encourage development for a novel system. Therefore, our goal is to use common Web technologies, such as HTML5, CSS, and JavaScript for software development, which not only eliminates the need for learning an entirely new platform that is exclusive to a system but also accommodates a transference of skills from prior Web development experiences. Web technologies also hold appeal for an active tangible platform, as the same technologies can be used to develop interfaces for other devices with which the active tangibles might be used (e.g., interactive tabletops), facilitating communication between different system components.

*G5: Feedback.* Apart from visual feedback on the touchscreen, the tangibles should enable alternative



Figure 3.1: The tilt interaction. The tilted edge gives a confirmation flash.

feedback methods, such as auditory, haptic/vibration, and visual feedback through LEDs.

*G6: Easy to use and customize.* The physical design of the active tangible must allow the user to easily grasp it with one hand (e.g., not too bulky or heavy, smoothed edges). The exterior of the device should also be visually appealing and customizable to fit the look-and-feel of particular systems.

## 3.2 Interactions

The basic interactive capabilities of the Actibles can be grouped into five categories. These categories include tangible interactions, on-screen interactions, physical controls, tabletop interactions, and visual feedback. These basic interactions are not mutually independent. Combinations of basic interactions can be implemented by developers to create unique interactions for specific applications. The many possible interactions make Actibles a robust and flexible platform for HCI applications.

### Tangible Interactions

The tangible interactions supported by the Actibles are derived from the common interactions found in prior research on active tangibles. As described in our design goals, these include both individual interactions (i.e., one tangible on its own) and group interactions (i.e., multiple tangibles used together).

Individual interactions include the following tangible gestural interactions [6, 28]: 1) Tilting – a tangible is tilted from side-to-side, for example to scroll through a list; 2) Shaking – a tangible is given a brief shake, for example to clear its contents; and 3) Flipping – a tangible is flipped over, for example to undo an action. Tilting and shaking interactions are especially familiar from previous active tangible based systems, such as those using Sifteo Cubes. For example, Eugenie [11] uses tilting as a means to browse through a database of items at the same level in a hierarchical database. An expressive use of



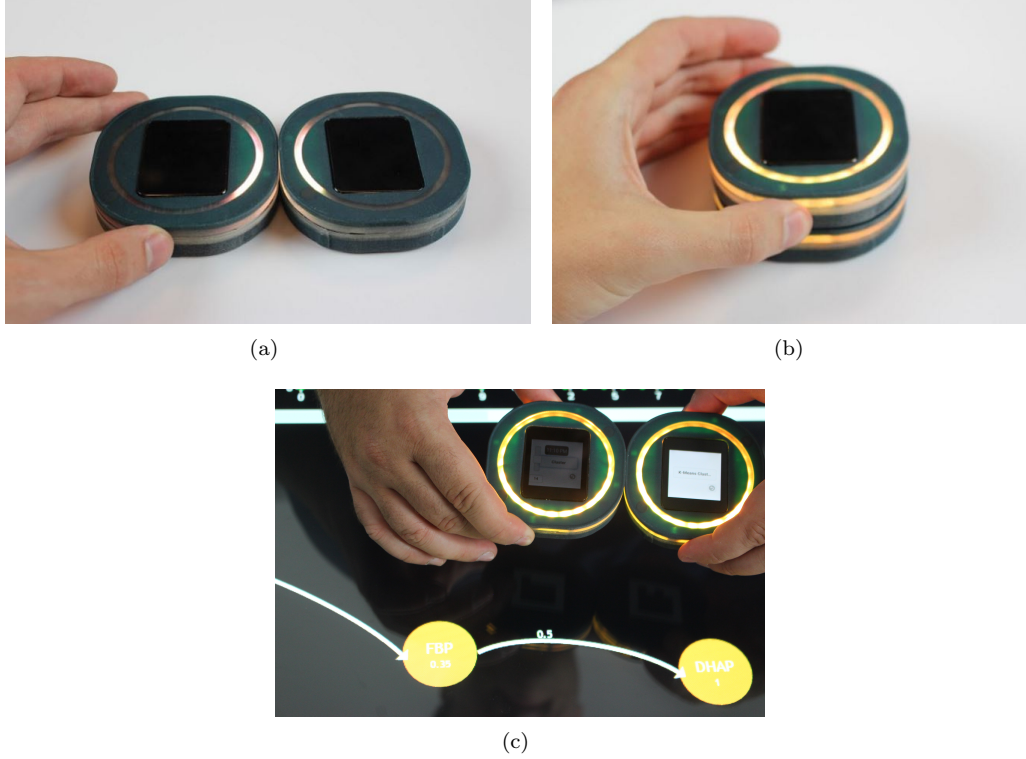


Figure 3.2: Grouped Interactions: (a) Neighboring configuration. The LEDs of the neighbored edges are activated for user feedback ; (b) Stacked configuration. Both Actibles activate their LED rings momentarily when a stacking event is detected; (c) A momentary bumping of two Actibles

shaking is seen in the SynFlo system [12], where two Sifteo Cubes are shaken together to transfer a gene into a plasmid.

Group interactions include: 1) Bumping – two tangibles are briefly touched together, for example to transfer content from one to the other; 2) Neighboring – two or more tangibles are touched together along particular edges, for example to create a relationship between them; and 3) Stacking – one tangible is placed on top of another, for example to filter contents associated with the lower tangible based on a parameter setting associated with the upper tangible. Neighboring and stacking interactions are shown in Figures 4.4(a) and 4.4(b). The neighboring interaction has been used extensively in active tangible applications. For example, horizontal neighboring is used to sequence together video clips in the Tangible Video Editor [10], while vertical neighboring is used to traverse a hierarchy in Eugenie [11]. Stacking has been used, for example, by Stackables [8] to construct queries to a literature database.

Since the Actibles leverage a smartwatch display, all commonly available on-screen manipulations are available to the user. These on-screen interactions include tap, double-tap, swipe and pinch/spread. Tap interactions are often used for selection, e.g., on Sifteo cube applications such as SynFlo [12] and Eugenie [11]. Swiping is familiar from mobile devices, e.g., for moving between pages.

### Physical Controls

The only physical control currently available on the Actible is the tactile button on the upper right corner. It is used for power switching, but can also be used for selection in an application context. For example, the button press could be used to confirm a selecting when tilting to scroll through a menu on the Actible screen.

### Visual Feedback

The Actibles provide two types of visual feedback. The smartwatch core contains an LCD display, which can be used to display menus and messages. An RGB LED ring is also used to provide non-graphical visual feedback. For example, when two Actibles are neighbored there is a confirmation flash along the neighbored edge of each Active (see Figure 4.4(a)), and when two Actibles are stacked there is a confirmation flash of the whole LED ring on both Actibles (see Figure 4.4(b)). Developers can also create custom LED feedback to provide confirmation of actions or indicate error conditions in their applications. For example, a repeated red flashing of the whole LED ring could indicate that a performed action is invalid.

### Tabletop Interactions

The Actibles are ideal for use in interactive tabletop applications. Common tangible tabletop interactions such as placing, moving, rotating and removing are facilitated by the interactive tabletop, and can be used to manipulate content display on the tabletop surface. Additionally, nearly all of the above-mentioned interactions can be performed on the Actibles while they are on an interactive tabletop. This includes neighboring, stacking, and button press, as well as all on-screen interactions and visual feedback.

## Chapter 4

# Technical Implementation

From a technical perspective, our goals were to create an active tangible that enables easy application development (G4), is easy to use and customize (G6), and provides a variety of input and feedback options (G2, G3, G5), including those that are common to most active tangible based applications (G1). These design goals led to a compact device with the following physical design, and hardware and software implementation.

### 4.1 Design Process and Methodology

The technical implementation of the design goals and requirements of the Actibles platform was an iterative process. Previous work by Arif et. al [7, 29] explored the use of smartwatches as active tangibles, as well as using smartwatches with custom cases as active tangibles .

#### 4.1.1 Smartphones as Active Tangibles

The use of smartphones as active tangibles was explored. Interaction applications included 'viewing and copying images, sharing playlists and receiving messages/calls on tabletop surfaces. While these applications were mostly carried out with the use of fiducial markers and multi-touch gestures, the authors encouraged the use of the accelerometer, gyroscope and other sensors to explore additional interactions. It was later determined that to implement interactions such as stacking and neighboring would be immensely difficult. It was found however that the use of a touchscreen, its use of web-based technologies, and its processing capabilities were ideal, and that future active tangible development should utilize the smartphone or other similar mobile technologies.

#### 4.1.2 NFC for Stacking Detection

For example, NFC sensors was considered for stacking detection. Previous work by Jain et. al. [30] explored the use of NFC in mobile devices for proximity detection and handshaking. Although NFC would be an ideal candidate for these types proximity detection, it would be impossible to implement for

stacking multiple devices. First, NFC is a type of Radio Frequency Identification (RFID) that uses special tags designed for a specific frequency (13.56 MHz). The proximity required for NFC communication is approximately 4-10 centimeters. Although this fact would make NFC an ideal candidate, multiple tags are not supported. This is because like RFID, NFC tags are passive devices that draw power from the phones signal. If multiple tags are present, there may not be sufficient signal for the tags to be activated. Another concern is that NFC does not detect proximity. Since stacking may occur with multiple devices, it would be impossible to determine the stacking order, assuming multiple tags could be read.

In order for the interaction capabilities to be to be user friendly and robust, it was determined that alternative hardware would have to be implemented such as proximity sensors.

### 4.1.3 The Sparse Tangible Active Tangible

While participants in a Sparse Tangible user study praised the tabletop system with active tangibles as a possible productivity enhancer, there were some suggestions brought up by the participants to enhance the experience. One complaint was a lack of filtering capability. In the Sparse Tangibles project, stacking was the only interaction that initiated a filtering event. The lack of neighboring capabilities in the Sparse Tangible active tangible was the cause, and including neighboring would expand the filtering capabilities of the active tangibles. Informally, the participants found it difficult to stack the active tangibles properly. This was due to two factors: first, stacking detection was implemented using pins. This required almost perfect alignment of the active tangibles. Secondly, the circular geometry of the active tangible did not facilitate stacking alignment. A circular shape was chosen because the active tangible could be used as a dial. However, this geometry did not allow for easy stacking alignment, because a small rotation would misalign the pins. It became clear that a new type of stacking detection method was needed along with a geometry that facilitated stacking, while maintaining a circular shape for dial manipulation.

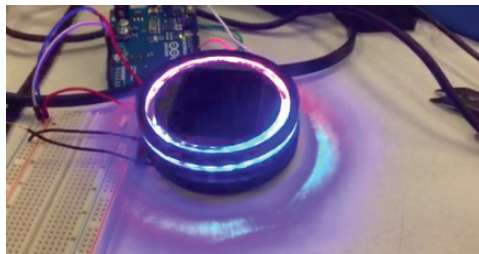


Figure 4.1: The first iteration of the active tangibles contained a smartwatch core, required pins for alignment and was circular in geometry.

### 4.1.4 Implementation of the Actible Platform

Based on the previous work, research and design iterations, a new active tangible was designed. All features, findings, limitations and characteristics from previous designs were implemented as a single device called the 'Actible'.

## 4.2 Physical Design

The Actible can be broken down into four main components, namely the case, the smartwatch, the circuit board, and the battery (see Figure 4.2). The case itself (Figure 4.3, parts B, D and F) serves to securely house the smartwatch, circuitboard and battery. The battery can be easily removed from the bottom of the case by opening the cantilevered snap fit panel (Figure 4.3, part F). The battery is also rechargeable via the USB port. The case is also designed to contain neodymium magnets and Hall effect sensors in a specific layout, in order to detect stacking, bumping and neighboring (as described in the following section).

In terms of physical form, the case is designed with a squircular geometry in order to facilitate bumping, neighboring and stacking alignment, while maintaining a rounded shape for dials, rotation and circular GUIs. This geometry also smoothly houses a circular arrangement of 24 RGB LEDs, which are visible through a translucent ring on the top and side of the case. The case is 3D printed on a Objet260 Connex3 printer out of a mixture of VeroCyan and TangoBlack plastic for the opaque portions and VeroClear plastic for the translucent portions. The assembly schematics for the case can be found in Appendix A.

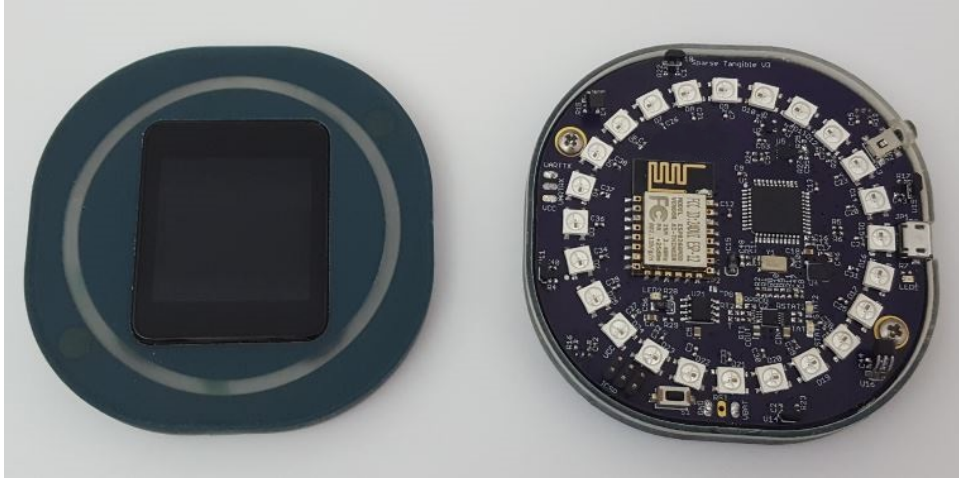


Figure 4.2: A 'field strip' unassembled view of an Actible sans smartwatch wiring. The top half contains the smartwatch, and the bottom half contains the circuit board and battery.

## 4.3 Hardware

The Actibles hardware consists of the smartwatch and custom circuitboard (Figure 4.3, parts A and C). The smartwatch used in the current implementation is the LG G Watch, which comes with a 280 x 280 resolution IPS LCD Display, 9 Axis sensors, 512 MB RAM, and 4GB internal storage. The smartwatch draws power and power/reset control from the circuit board. Solderable terminals for power and reset are located on the circuit board to connect many different types of smartwatches.

The custom board is what makes an Actible unique, and enables both smartwatch integration and interactive feedback. The custom board is a single board solution that is inserted into the case. It contains eight omnidirectional Hall effect sensors for neighboring and stacking detection. An ESP8266 WiFi module was installed for UDP or TCP/IP connectivity. LEDs are arranged in a circular pattern for user feedback. A standard lithium polymer smartphone battery is used to extended battery life for 10 hours of continuous use. The battery can be recharged via the USB port on the side of the Actible. This USB port can also be used to upgrade the Arduino firmware. Power supervisory circuitry was integrated for battery under voltage protection, tactile button shutdown initialization and reset. If the Actible is powered down, a quick push of the tactile button will power it on. When Actible is powered on, a three second push and hold will power it down. The full electrical schematic and bill of materials can be found in Appendix A.

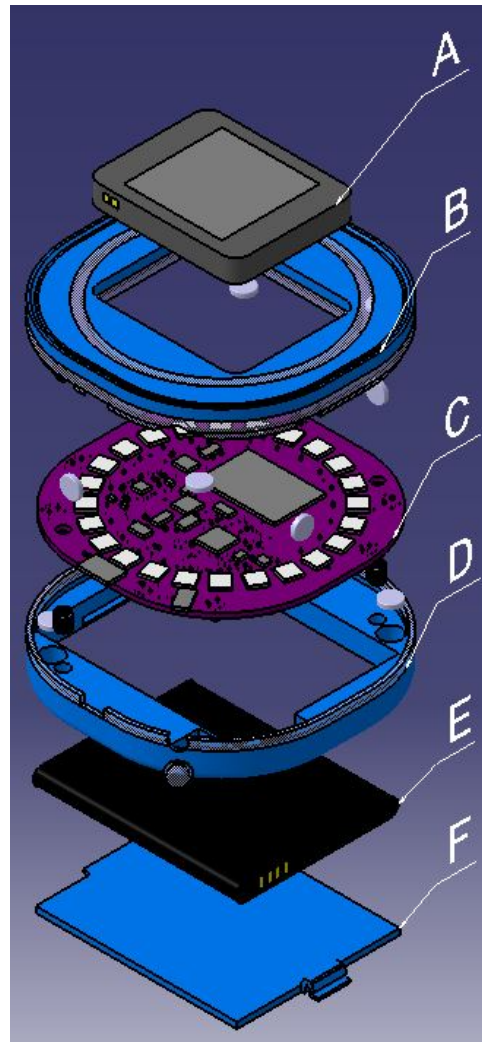


Figure 4.3: An exploded assembly view of the Actible showing: A) smartwatch, B) upper case, C) circuit board, D) lower case, E) smartphone battery, and F) battery cover.



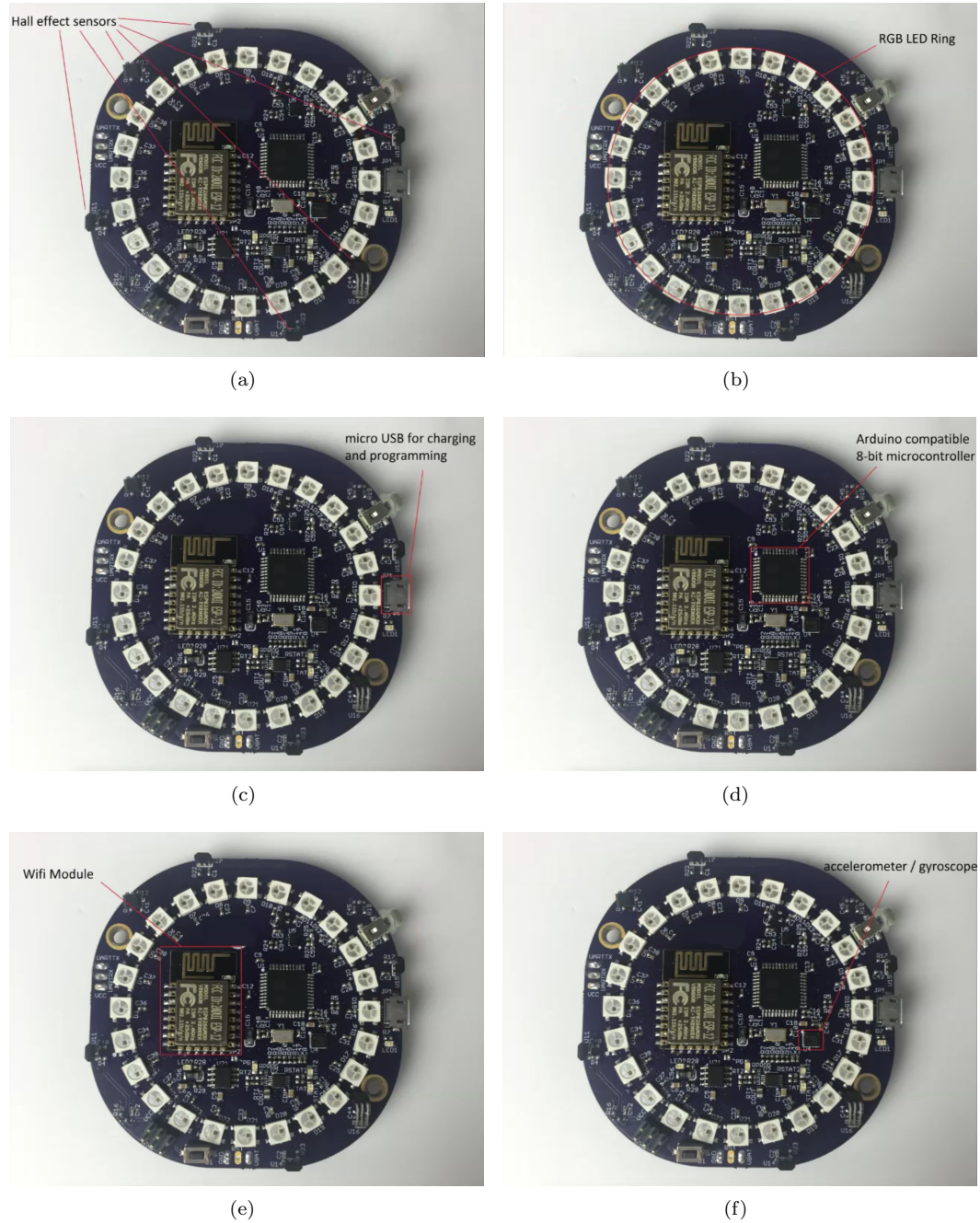


Figure 4.4: A set of four subfigures: (a) Hall effect sensor placement; (b) RGB LED ring; (c) Micro-B USB placement; (d) Microcontroller placement; (e) WIFI module placement and, (f) Accelerometer and Gyroscope Placement.

For the interactions describe above, the Actibles implement the following hardware solutions:

*Tangible Interactions.* The individual (i.e., single tangible) interactions of tilting, flipping, and shaking are enabled by an accelerometer/gyro IC controlled using an 8-bit Arduino compatible Atmel microcontroller. The group interactions of bumping, neighboring and stacking are enabled by neodymium magnets and Hall effect sensors. Bumping is a brief planar side contact of two Actibles, while neighboring is the planar side contact of two or more Actibles. This is accomplished both electronically and geometrically. Neodymium magnets are arranged in a staggered formation in the Actibles case to trip off an adjacent Actible’s Hall effect sensor, which are also placed on the circuit board in a staggered formation, each in an adjacent and opposite position to the magnets in its case. This magnet/Hall effect sensor orientation allows for any combination or tessellation of neighboring or bumping tangibles, and also has a high tolerance for misalignment. Stacking is accomplished using the same geometrical and electrical implementations as neighboring and bumping. The physical shape (squire geometry) of the Actibles allows for an intuitive alignment for bumping, neighboring and stacking.

*On-Screen Interactions.* The smartwatch’s capacitive touch capabilities are leveraged through the jQuery Mobile<sup>1</sup> JavaScript library, which can detect touch, swipe, and pinch/spread gestures among others. These gestures, once recognized by the Web application, are communicated via Socket.io<sup>2</sup> to a central Node.js<sup>3</sup> server, which handles the action.

*Physical Controls.* The power supervisory circuitry allows for the tactile button to act not only as a power button, but also as a tactile sensor. If the Actible is powered on, the PSU sends a signal to the microcontroller if the tactile power button is pushed for less than three seconds. This enables the power button to also act as a selector.

*Tabletop Interactions.* These are made possible by utilizing a set of standardized fiducial markers placed on the bottom of the Actibles. The Actibles can be used with different interactive tabletop platforms, such as ReacTIVision [2] and MultiTaction<sup>4</sup>. Our projects employ MultiTaction Cell Ultra-Thin Bezel (MT553UTB) displays and corresponding MultiTaction Codice markers. We leverage the TUIO protocol for multi-touch communication, which possesses JavaScript (and other language) libraries and allows all multi-touch events to be streamed directly to any device on the network.

*Visual Feedback.* Two types of visual feedback are available on the Actible. The main visual feedback is the LCD display of the embedded smartwatch. This display can be easily customized. If the smartwatch has a browser installed, it can display a GUI designed in HTML5/JavaScript. Android applications can also be used. A second type of visual feedback available is a 24 RGB LED ring. The LED ring is addressable by the on-board micro-controller. The LEDs are installed in a circular configuration, and can be used to provide non-graphical interaction feedback, such as flashes or simple animations in response to actions. For example, neighboring two Actibles results in a flash along the neighbored edge on each Actible (see Figure 4.4(a)), while stacking one tangible on top of another results in a whole-circle flash on both of the Actibles (see Figure 4.4(b)).

---

<sup>1</sup><https://jquerymobile.com>

<sup>2</sup><http://socket.io>

<sup>3</sup><https://nodejs.org/>

<sup>4</sup><https://www.multitaction.com/>



## 4.4 Software

The basic software required for the Actibles to function is firmware that enables WiFi communication and PSU control. This firmware is written in Arduino, although supports AVR. The Actibles firmware contains interface functions for the accelerometer, LED array and Hall effect sensors. Hardware interface functions are provided for WiFi communication, Hall effect sensor detection, accelerometer functions such as flip, tilt, and shake detection. Various RGB LED functions are also provided. All functions can be configured to send or receive messages to the server via WiFi, depending on whether the specific function is feedback or input. Other interactions that involve two or more tangibles, such as neighboring and stacking, require server side functions to receive Hall effect sensor detection messages from each Actible. The server then determines which tangibles are interacting by analyzing the Hall effect sensor message time stamps from each Actible.

A UDP or TCP/IP server can be implemented to interact with the Actibles if they are to be used in conjunction with an interactive tabletop or other interaction device. A separate UDP or TCP/IP server is not mandatory if one Actible is acting as a server for independent, off-table usage. The server can be implemented to send and receive commands to control the LEDs, power off the active tangibles, as well as for handshaking, timeouts, etc. Any custom communication protocol can be set up by the user.

A standard Android browser is utilized on the Actible's smartwatch to view custom Web based GUIs. These custom GUIs are usually menus, and are developed using HTML5, CSS and JavaScript. Menu manipulation on the smartwatch browser triggers events which send messages to the server. The server then coordinates various feedback messages to the Actibles.



## Chapter 5

# Application Examples

We chose two existing projects, Active Pathways [31, 32] and Sparse Tangibles [7], and one new project, Tangible mtDNA, to demonstrate how Actibles can be used in different application scenarios. All three applications make use of Actibles in combination with an interactive tabletop.

We chose these applications for a number of reasons. First, they use a wide range of tangible interactions, both on- and off-the-table, which makes them ideal for demonstrating the interaction space afforded by the Actibles platform. Second, the two existing projects made use of different kinds of active tangibles in their previous versions – Sifteo Cubes in the case of Active Pathways, and custom active tangibles in the case of Sparse Tangibles – allowing us to demonstrate how Actibles can replace other kinds of active tangibles. Finally, the third application, Tangible mtDNA, showcases an application that was built from scratch using Actibles as its platform for tangible tabletop interactions.

### 5.1 Active Pathways

Active Pathways is a tangible tabletop application designed for biochemical modeling. Using an interactive tabletop display surface and tangible interactions Active Pathways allows systems biologists to construct and simulate biochemical reaction networks, as well as to "fit" (i.e., match) their constructed models to experimental datasets by adjusting model parameters such as molecule concentrations and reactions speeds. The project has gone through several iterations. The first version, called Kinesthetic Pathways [32], focused on simulation alone, employing passive tangibles to manipulate parameters in the tabletop simulation (e.g., a dial for changing parameter values on the table) and to display different kinds of output charts (e.g., time-based graphs and error charts).

The second version, called Pathways Builder [31], allowed users to construct the biochemical reaction network and fit it to a selected set of experimental data using Sifteo cube based active tangibles. In this case, the active tangibles act as objects within the biochemical reaction network, such as molecules and reactions, as well as tools to manipulate these objects, such as a dial. Building a simple reaction network is as easy as picking up two active tangibles, navigating their on-screen menus to associate them with molecules, and touching them together to create a reaction between them (Figure 5.1). The Sifteo Cubes



Figure 5.1: The Actibles represent molecules in the Active Pathways system. Off-table neighboring interactions are used to construct reactions between molecules.

can then be placed on the table and rotated to adjust associated parameters, such as their concentration value. Shaking a Sifteo cube unbinds it from the object that it has been associated to (e.g., a molecule). The change from passive to active tangibles allowed us to extend the interactions beyond the tabletop, thus menu selection can be made off-the-table by tilting the active tangibles, and reactions can be created by touching two tangibles together in the air (see Figure 5.1).

From a technical perspective, the functionality of the Sifteo-based implementation was easily transitioned over to the Actibles-based implementation. Moreover, the transition allowed us to simplify the project codebase, as both the tabletop and active tangible interfaces are now implemented in JavaScript/HTML5 and there is no need to maintain a separate native codebase for the Sifteo Cubes. From a development perspective, this also allows us to test components of the Actibles interface within a Web browser running on desktop computer, without requiring a simulator program. The Actibles allowed us to implement an on-screen swipe gesture to scroll through menus (e.g., to select between molecules, enzymes and reactions), which is more familiar to users than tilting based on mobile device usage experience. The Actibles also allowed us to explore new custom off-the-table interactions, such as tilt-dialing (i.e., tilting an Actible in a circular motion in the air to mimic a dial), as well as the use of LED feedback to notify the user if a physical connection of the Actibles has occurred, and if that reaction is valid.



Figure 5.2: Actibles are used to browse gene interaction networks in the Sparse Tangibles system. Stacking one Actible on top of another filters the data associated with the bottom Actible based on the parameters set on the upper Actible.

## 5.2 Sparse Tangibles

Sparse Tangibles [7] is an interactive tabletop application that allows users to collaboratively explore gene and protein networks using active tangibles. It uses smartwatches as active tangibles to allow both on- and off-the-table interactions. It also uses a custom case for the smartwatches for better usability and to extend their interaction space. The case is augmented with spring-loaded pins on the top of the tangible and targets on the bottom to enable communication between the tangibles when stacked.

In Sparse Tangibles, users navigate on-screen menus to associate active tangibles to specific organisms or genes. The active tangibles can then be placed on the tabletop visualize available networks (for a selected organism) or available organisms with the given gene (for a selected gene). Sparse Tangibles also allows expressive query construction using tangibles interactions. To construct a query, users associate an active tangible to a particular filter and set its corresponding parameters by tapping and swiping on the screen. Users then stack this tangible onto organism or gene tangibles in order to express the query (Figure 5.2). Multiple filters can be stacked to express a more complex query consisting of a logical AND relationship. If the query is valid, the application displays the results on the tabletop; if not, an error message appears on the tangible display. The user can remove a query parameter by shaking the corresponding tangible.

We replaced the custom tangibles of Sparse Tangibles with Actibles. The previous tangibles were also based on a smartwatch and provided similar functionality, despite using a different sensing approach for detecting stacking. The updated version of Sparse Tangibles provides users with more expressive visual feedback using the RGB LEDs. These act as a non-graphical display to provide users with feedback

on: 1) An interaction or action (i.e., tilting, confirmation of stacking); 2) Valid and invalid interactions (including the cause of the error, when determinable); and 3) additional information (such as a parameter value). This further reduces the tangibles' dependency on the tabletop and the built-in display screen, increasing the interaction space of the application. The Actibles also provide additional interactions that were unavailable on the previous custom tangibles, such as neighboring. Sparse Tangibles currently supports query construction via stacking for the logical AND relationship only. We envision using the neighboring interaction to express different kinds of logical relationships for query construction.

### 5.3 Tangible mtDNA

Tangible mtDNA is a tangible tabletop system that allows multiple users to better understand mitochondrial DNA sequencing mutation data in breast cancer patients through collaborative interaction. The system uses a multi-touch tabletop that also supports tangible interaction. From the outset, the system has used the Actibles platform for tangible interactions, including stacking, shaking, and several on-screen gestures, both on- and off-the-table. These gestures include tapping, swiping, dragging, rotating, and pinching. Like Sparse Tangibles, users conduct queries with these interactions in order to navigate, manipulate and visualize large datasets. In the case of Tangible mtDNA, the dataset includes both DNA sequence and mutation data, as well as clinicopathological features (e.g., age, hemoglobin levels, etc.) of patients.

After sorting and filtering the data with the Actibles, users can compare their findings by stacking the Actibles and placing them on the table, creating an interactive visualization that combines each user's queries. Additionally, a mobile menu on the Actibles permits users to choose gene-based or patient-based clustering options and select the number of clusters with a slider (see Figure 5.3). Placing the Actibles on the tabletop displays both the mutation patterns and clustering results. Removing the Actibles from the tabletop hides the mutation patterns but maintains the results of clustering. Shaking an Actible clears any cached selections.

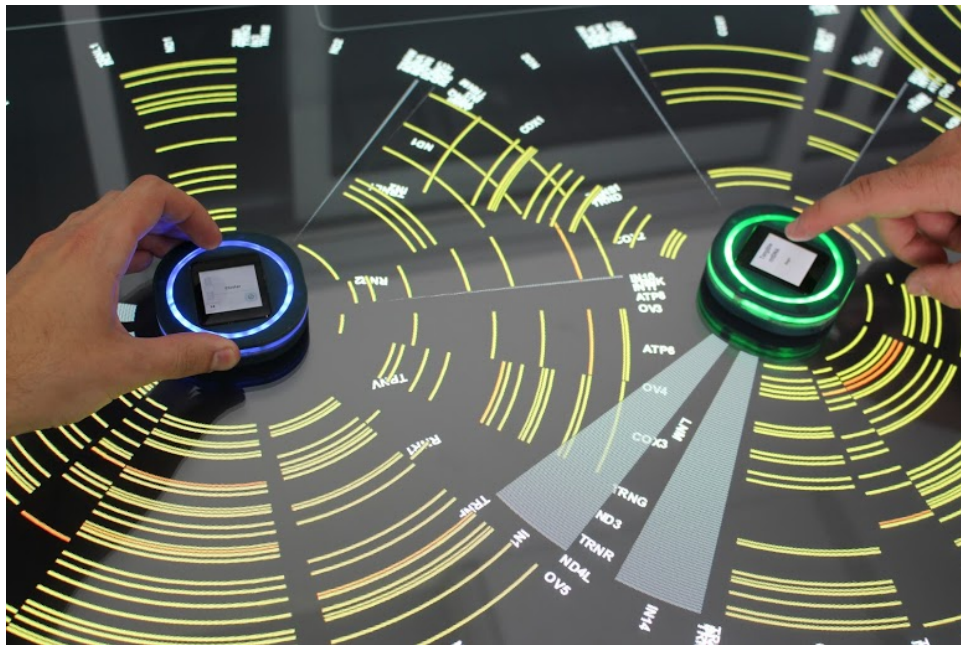


Figure 5.3: Actibles are used to explore mitochondrial DNA mutation data of breast cancer patients in relation to different clinicopathological features in the Tangible mtDNA system. Users navigate menus on the Actibles screen to select gene or patient based clustering methods.





## Chapter 6

# Evaluation

### 6.1 Evaluative Comparison

Actibles are custom active tangibles designed for TEI applications. As such, many design choices were made based on current hardware shortcomings of other devices commonly used in TEI applications (i.e. smartwatches, smart-phones and Sifteo Cubes). Therefore it is not surprising that the Actible has many advantages over off-the-shelf devices for TEI applications.

#### 6.1.1 Smartphones and Smartwatches

Smartphones and smartwatches are often used as active tangibles. The main challenge with these devices is that they do not have specialized hardware to detect proximity. Proximity features may be programmed utilizing the accelerometer or near field communication (NFC). However, this implementation would be difficult arduous at best and most likely unreliable for TEI applications. The main motivation behind the use of smartphones and smartwatches in TEI applications is their ability to facilitate on-screen and multi-touch interactions, ease of software application development and inherent processing capabilities. Also, a smartwatch's small form factor makes it ideal for applications with multiple active tangibles. This is why we selected a smartwatch as the Actible's core. The smartwatch's low battery life is not an issue since the smartwatch core is powered via the Actible's 2100 mAh battery.

By incorporating all of the above features, the Actible is able to host a wide variety of familiar on-screen and multi-touch interactions. The browser and android operating system allow for rapid software development. Lastly, the extended battery life allows an Actible to be powered on for longer when used in TEI applications.

#### 6.1.2 Sifteo Cubes

As stated previously, Sifteo Cubes require custom codebases to complete basic TEI tasks. GUI development on the Sifteo Cube is a similar problem. Battery life is low (approximately 4h on a single

AAA battery). Sifteo Cubes are not rechargeable. However one characteristic that makes Sifteo Cubes highly desirable as an active tangible, is its ability to sense other Sifteo Cubes when they are in close proximity (contacting). This feature was incorporated into the Actible's design. The Actible is also capable of detecting stacking events, whereas the Sifteo Cube is not. Sifteo Cubes require additional hardware to communicate with a computer, and its range is approximately 20 feet. Actibles utilize WIFI communication, therefore no additional hardware is needed. Lastly the battery life on a single Actible is approximately 10 hours, well above a Sifteo Cubes useful battery life.

One unique feature the Actible has is the customizable LED ring. This LED ring gives the TEI developer an extra set of feedback capabilities.

Because of the desirable features that were incorporated into the design of the Actible; its smartwatch core, extended battery life, LED feedback, along with custom hardware for proximity sensing borrowed from Sifteo Cubes, the Actible is truly a better hardware platform for active tangible applications.

## 6.2 Performance Specifications

Evaluative Comparison of Hardware Utilized for Active Tangibles				
Hardware Specifications and Performance				
Property	Actible	Sifteo Cube	Smartphone	Smartwatch
Dimensions (mm)	80x80x17.5	45x45x20	110mm screen avg. 2017	50mm screen avg. 2017
Weight (g)	420	100	150 avg. 2017	50 avg 2017
Geometry	Squircle	Square	rectangle	Varies
WIFI	Yes	NO	YES	YES
Bluetooth	NO	NO	YES	YES
Other Wireless	NO	YES, custom	YES (NFC)	NO
Battery Charging	YES	NO	YES	YES
Battery (mAh)	2100	1000 (1xAAA)	Varies	Varies
Power Consumption (mA)	200 avg.	250 avg.	Varies	Varies
Battery Life	10.5h	4.5h	Varies	Varies
Proximity Sensing	YES	YES	YES*	YES*
IMU	YES	YES	YES	YES
RGB LEDs	YES	NO	Varies (typically 1 status RGB LED)	Varies
Interactions				
Interaction	Actible	Sifteo Cube	Smartphone	Smartwatch
Neighboring	YES	YES	YES*	YES*
Stacking	YES	NO	YES*	NO*
Bumping	YES	YES	YES*	YES*
Tilting	YES	NO	YES	YES
Flipping	YES	YES	YES	YES
Shaking	YES	YES	YES	YES
Physical Controls	YES	NO	YES	YES
LED Feedback	YES	NO	YES	YES
Haptic Feedback	NO	NO	YES	YES

Table 6.1: Performance and Interaction Comparison

<sup>0\*</sup>These capabilities may possibly be imitated with significant software modifications and testing with current smartphone technology.



## Chapter 7

# Discussion

Despite their commercial failure, many researchers and DIY communities have appropriated Sifteos for use in a variety projects beyond gaming markets. The appropriation of Sifteos and other mass-market platforms like smartphones as active tangibles requires significant effort, demonstrating a need for active tangibles that are easy to use and easy to appropriate. We designed the Actibles platform to meet the above-mentioned goals, G1-G6, and to provide a flexible platform for developers and researchers who would like to incorporate active tangibles in their projects.

Furthermore, we designed the Actibles to incorporate the most desirable features of all available hardware that is currently employed as an active tangible in the TEI community. We combined the smartwatch’s software and display advantages with the sifteo’s proximity sensing capabilities to design a powerful and flexible active tangible platform for TEI applications.

Benefits of the Actibles include the broad range of interactions provided, including common active tangible interactions such as stacking, neighboring, tilting, flipping, as well as on-screen gestures such as tapping, swiping and pinching/spreading (G1 and G3). In addition to the integrated display screen, RGB LEDs also provide an alternate modality for visual feedback (G5). Moreover, the Actibles leverage Web technologies, and are thus easy to integrate with browser-based applications that run on other devices (G4). The diversity of sensors provided by the Actibles platform also encourages the exploration of new interactions, such as the tilt-dialing interaction implemented in Active Pathways (G2). We have also shown that Actibles can be used to transition projects that have used other kinds of active tangibles, such as Sifteo Cubes (G1) and that they provide a form factor that is easy to use in different kinds of tangible tabletop systems (G6). We have yet to test the creation of custom cases for the Actibles.

Despite these benefits, we have also identified some limitations of the current Actibles implementation. Although smartwatches allow for quick GUI and software development, they remain difficult to work with. Many smartwatch operating systems are custom, slimmed down versions of Android that encourage little firmware integration. Browsers are difficult to run, there is often no access to the sensor data, and the smartwatch is typically expected to be paired with a smartphone to provide much of the processing. Newer, higher performance smartwatch models may contain more robust versions of Android, independent of a smartphone. By making the Actibles platform open source we enable developers

to potentially integrate their own smartwatches. However, this requires some tedious integration tasks, such as disassembling the smartwatch and finding its reset pins in order to switch it on and off, as well as possible physical modifications and/or new case designs in order to integrate a new smartwatch into the Actibles.

## Chapter 8

# Conclusion

In this thesis, the Actibles platform was presented, an open source hardware/software platform that enables researchers and developers to explore the use of active tangibles in their applications. We described the design goals, interactions, and technical implementation of the platform, and demonstrated its use in three application scenarios.

### 8.1 Future Work

Future work will extend the Actibles design to create an Android and Arduino compatible version, as the single board solution is currently only Arduino compatible. We also aim to address the limitations of the smartwatch, by looking into development platforms that provide LCD display support and an ARM chipset that is Linux/Android compatible. In this way, we could maintain the support for Web technologies, while eliminating the need for a smartwatch. Our hope is that other HCI designers will be able to leverage the Actibles platform to explore and innovate new interactions, and that everyday users may turn to Actibles as a viable solution when interfacing with surface computer applications.



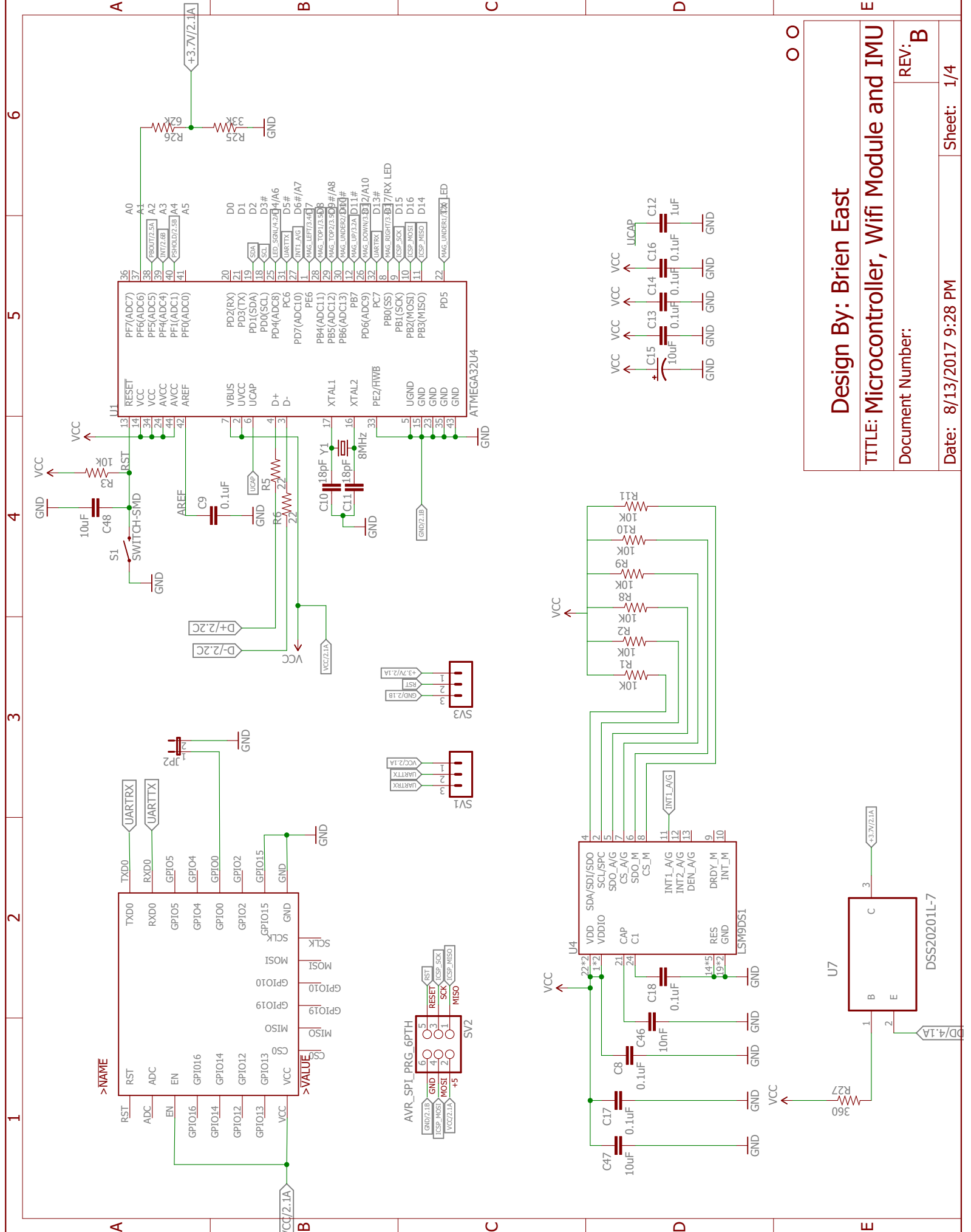


## Appendix A

# Design Documentation

*Included in Appendix A are the electrical and mechanical schematics for the Actible.*

## **A.1 Electrical Schematics**



Design By: Brien East

TITLE: Microcontroller, Wifi Module and IMU

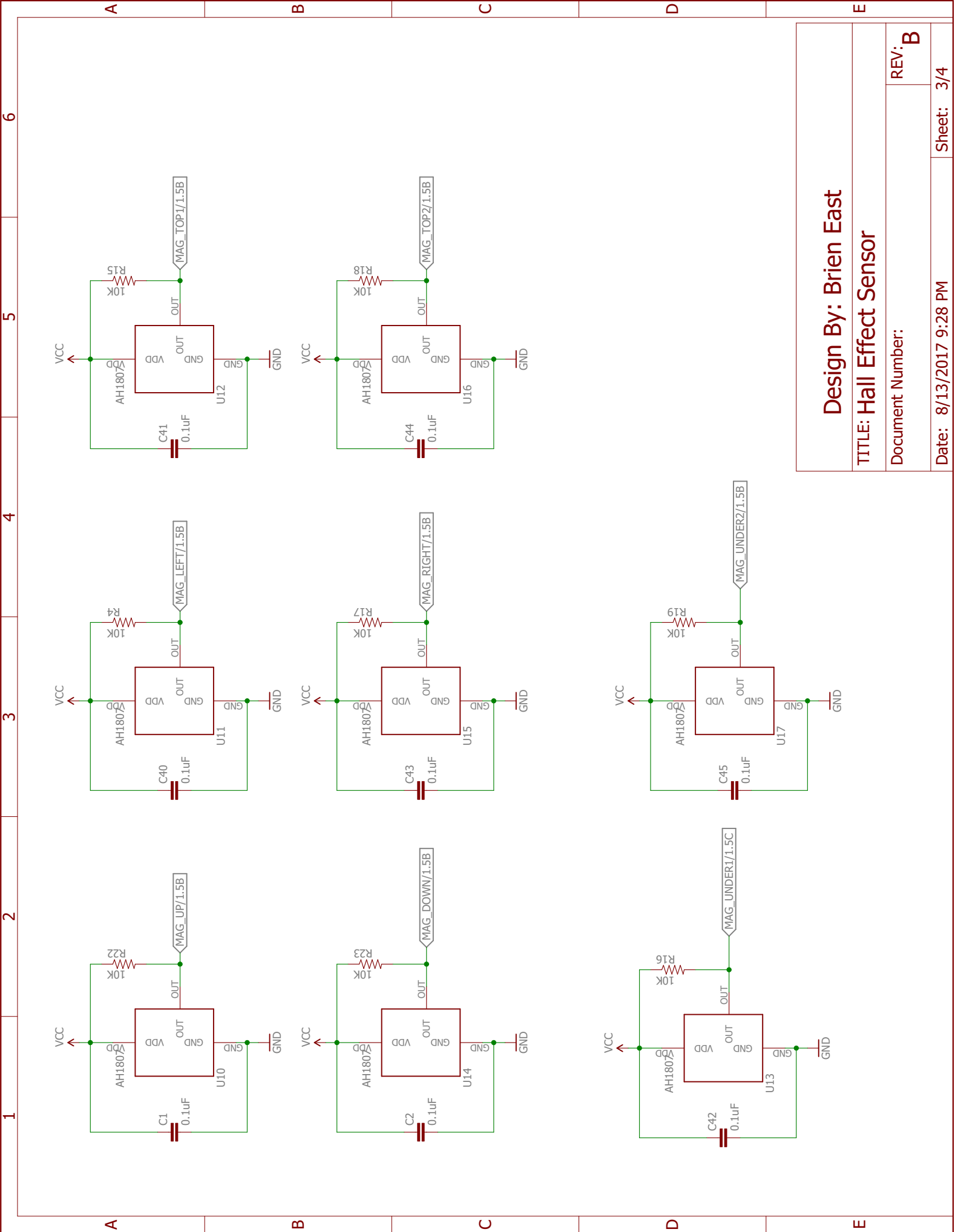
Document Number:

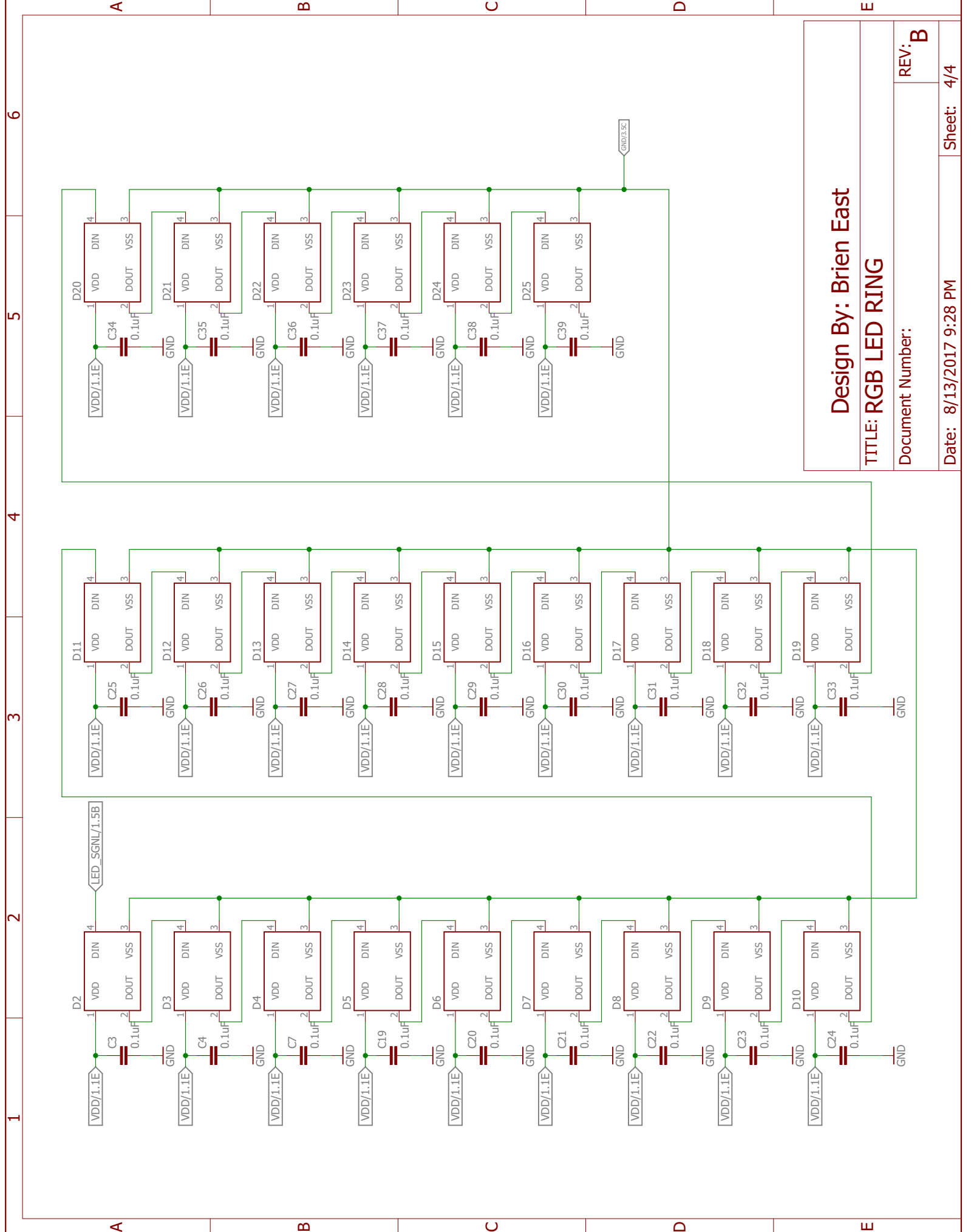
REV: B

Date: 8/13/2017 9:28 PM

Sheet: 1/4







Design By: Brien East

TITLE: RGB LED RING

Document Number:

REV: B

Date: 8/13/2017 9:28 PM

Sheet: 4/4

## A.1.1 Board Design

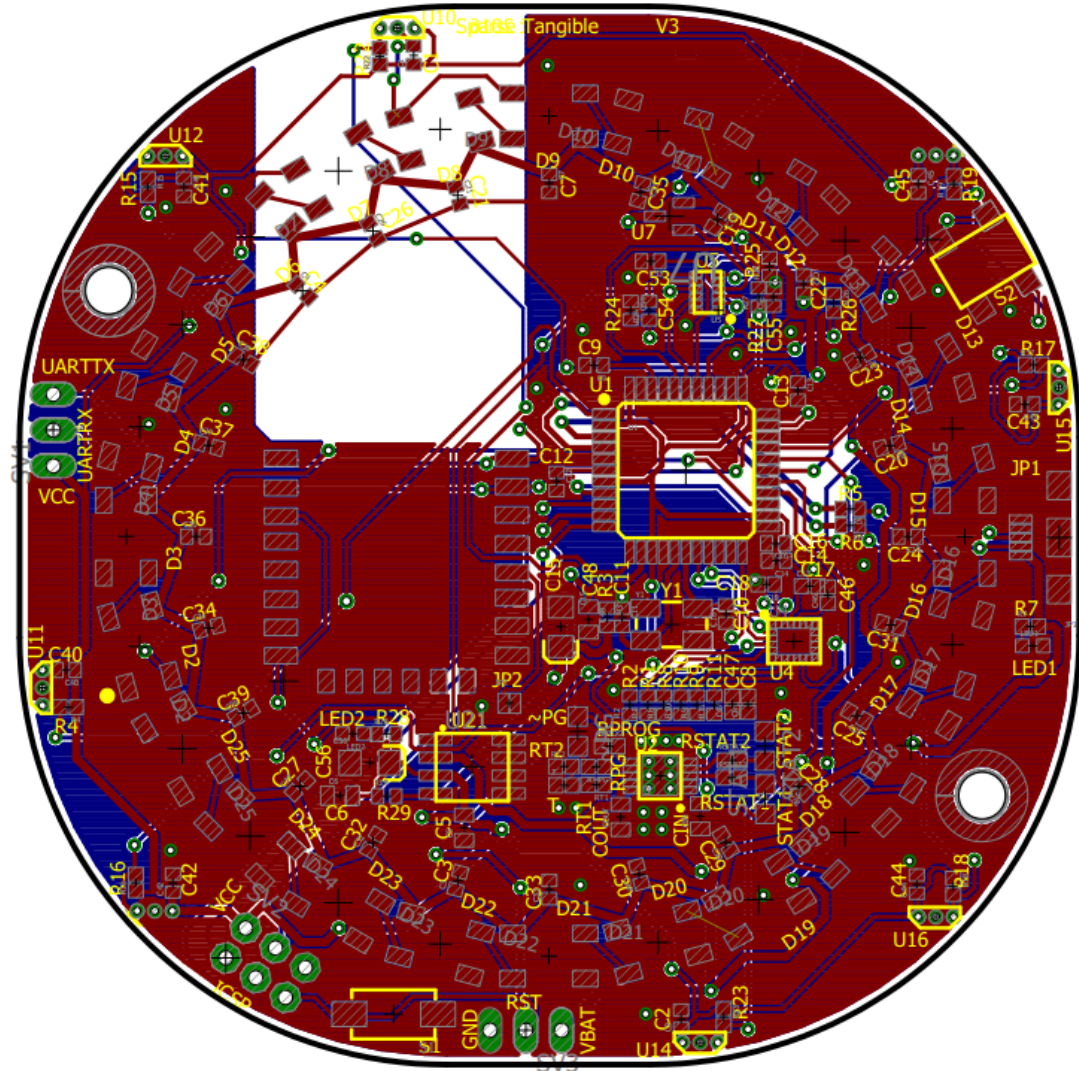


Figure A.1: PCB Layout Design

## A.1.2 Electrical Bill of Materials

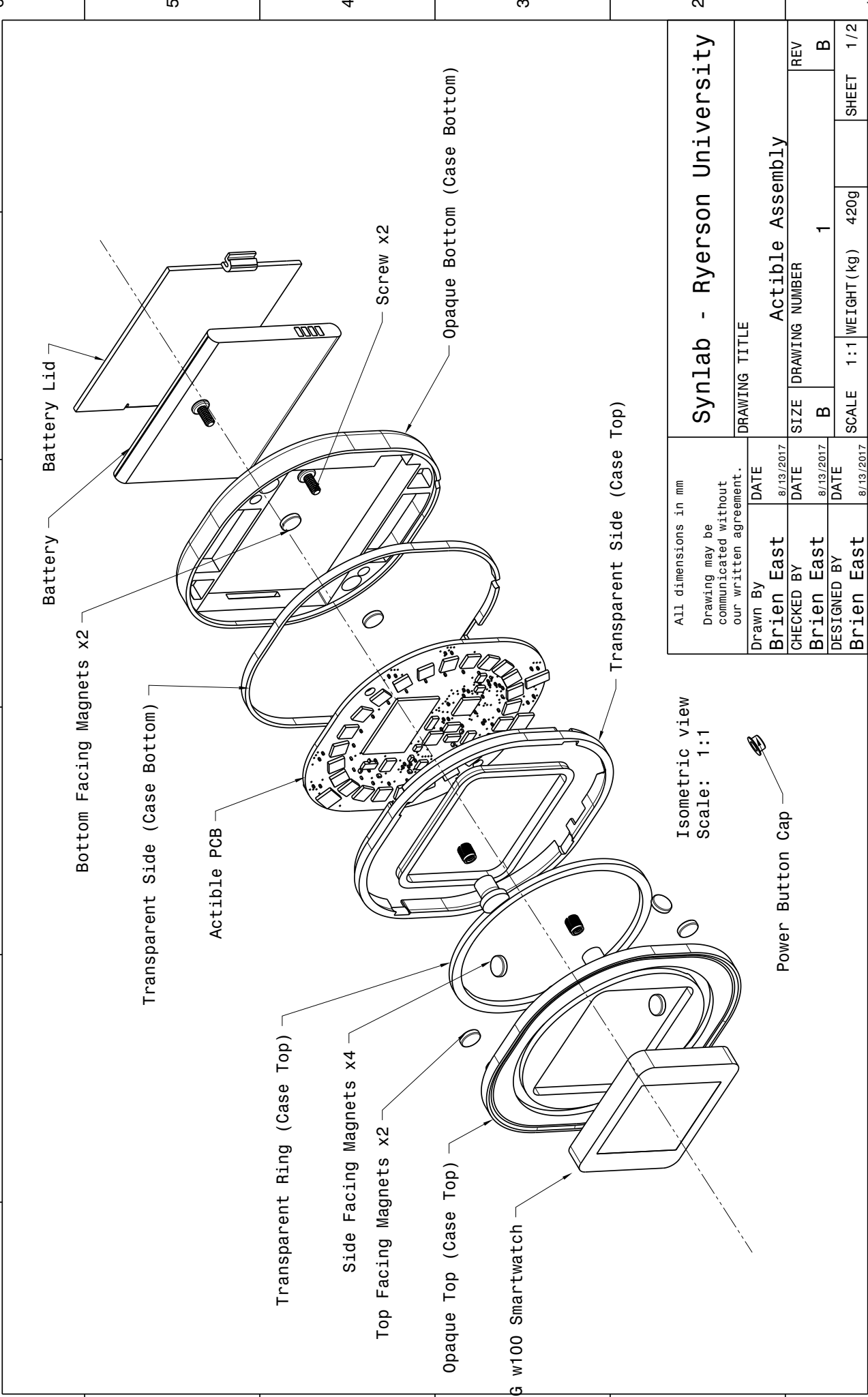
Bill of Materials			
Qty.	Part	Value	Pack
1		JUMPER-2SMD-NC	JP2
2		MA03-1	SV1, SV3
1		USB MICRO-B	JP1
24		WS2812B	D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, D16, D17, D18, D19, D20, D21, D22, D23, D24, D25
41	0.1uF	CAP0402-CAP	C1, C2, C3, C4, C7, C8, C9, C13, C14, C16, C17, C18, C19, C20, C21, C22, C23, C24, C25, C26, C27, C28, C29, C30, C31, C32, C33, C34, C35, C36, C37, C38, C39, C40, C41, C42, C43, C44, C45, C53, C54
1	0.1uF	CAP0805	C6
8	10K	RESISTOR0402	R4, R15, R16, R17, R18, R19, R22, R23
8	10K	RESISTOR0402	R1, R2, R8, R9, R10, R11, R24, R29
1	10k	RESISTOR0402	R3
1	10k/THERM	RESISTOR0402	T
1	10nF	CAP0402	C46
1	10uF	CAP0402	C47
3	10uF	CAP0805	C48, CIN, COUT
2	10uF	CAPPOL1206	C15, C56
3	150	RESISTOR0402	RSTAT1, RSTAT2, R PG
1	150K	RESISTOR0402	RT2
2	18pF	CAP0402	C10, C11
2	1K	RESISTOR0402	RPROG, RT1
1	1k	RESISTOR0402	R7
2	1uF	CAP0402	C12, C55
1	1uF	CAP0805	C5
2	22	RESISTOR0402	R5, R6
1	330	RESISTOR0402	R28
1	33k	RESISTOR0402	R25
1	360	RESISTOR0402	R27
1	BATTERY CONNector	4 PIN SMD	JP4
1	62k	RESISTOR0402	R26



1	8MHz	CRYSTAL5X3	Y1
8	AH1807	AH1807	U10, U11, U12, U13, U14, U15, U16, U17
1	ATMEGA32U	TQFP-44	U1
1	BLUE	LEDCHIPLED0805	STAT1
1	DSS20201L-7	DSS20201L-7	U7
1	ESP-12-E	ESP-12-E	U20
1	EVQ-PSQ02K	EVQ-PSQ02K	S2
1	GREEN	LEDCHIPLED0805	PG
1	LSM9DS1	LSM9DS1	U4
1	MCP73833	MCP73833	U2
1	MIC39101	MIC39101	U21
2	Power	LED0603	LED1, LED2
1	RED	LEDCHIPLED0805	STAT2
1	STM6601	STM6601	U5
1	SWITCH-SMD	SWITCH-MOMENTARY	TACTILE-SWITCH-1101NE S1

Table A.1: Bill of Materials

## **A.2 Mechanical Design**

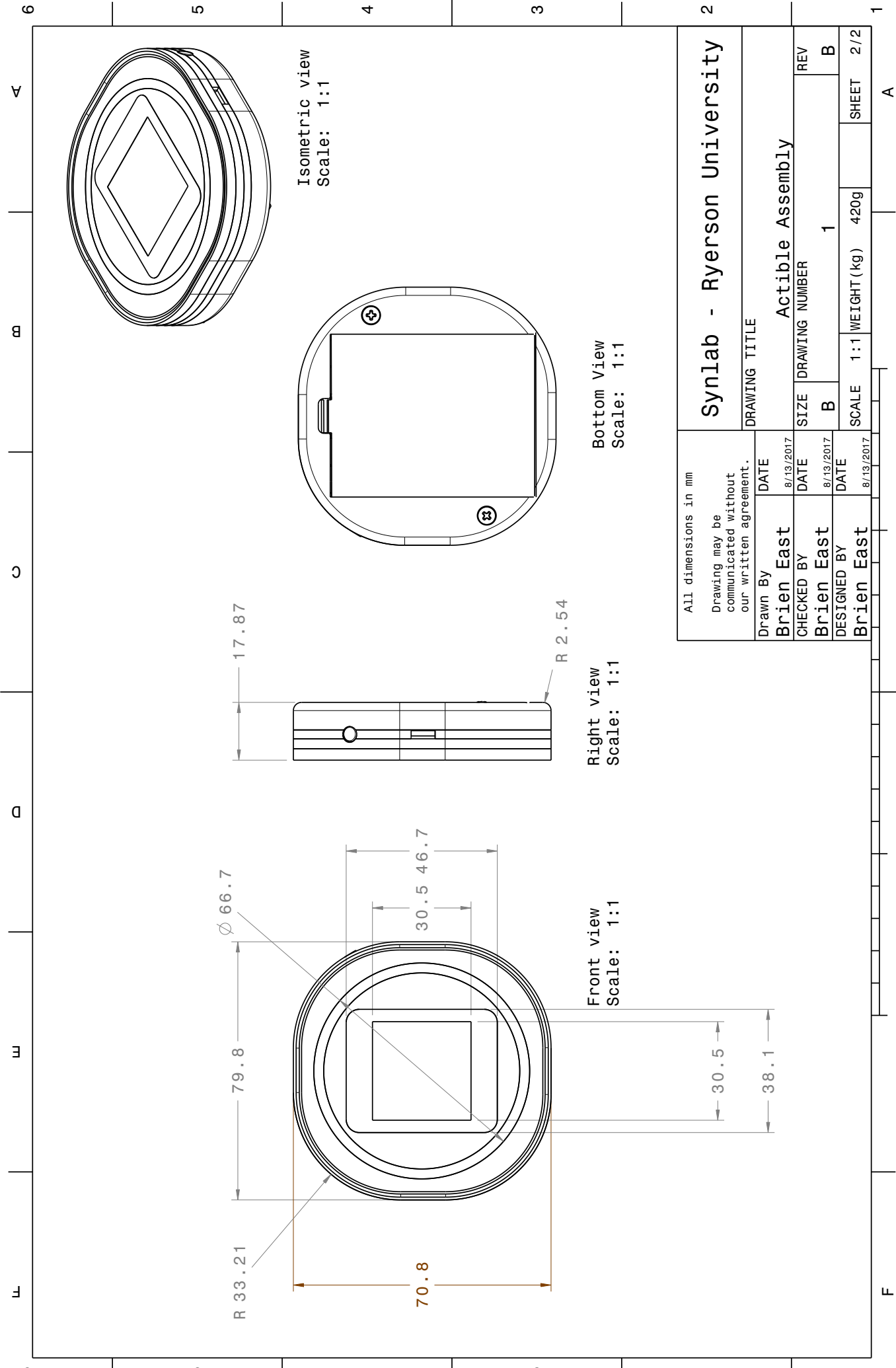


All dimensions in mm Drawing may be communicated without our written agreement.		DATE	8/13/2017
Drawn By	Brien East	CHECKED BY	Brien East
DESIGNED BY	Brien East	DATE	8/13/2017
Brien East		DATE	8/13/2017

Synlab - Ryerson University			
DRAWING TITLE			
Actible Assembly			
SIZE	DRAWING NUMBER	REV	
B	1	B	
SCALE	1:1	WEIGHT (kg)	420g
SHEET	1/2		

Isometric view  
Scale: 1:1

Power Button Cap



All dimensions in mm Drawing may be communicated without our written agreement.		Drawn By <b>Brien East</b>	DATE 8/13/2017
		CHECKED BY <b>Brien East</b>	DATE 8/13/2017
		DESIGNED BY <b>Brien East</b>	DATE 8/13/2017

Synlab - Ryerson University			
DRAWING TITLE			
Actible Assembly			
SIZE <b>B</b>	DRAWING NUMBER <b>1</b>	REV <b>B</b>	
SCALE <b>1:1</b>	WEIGHT (kg) <b>420g</b>	SHEET <b>2/2</b>	

# References

- [1] B. Ullmer and H. Ishii. Emerging frameworks for tangible user interfaces. *IBM Syst. J.*, 39(3-4):915–931, July 2000.
- [2] Martin Kaltenbrunner and Ross Bencina. reactivation: a computer-vision framework for table-based tangible interaction. In *Proceedings of the 1st international conference on Tangible and embedded interaction*, pages 69–74. ACM, 2007.
- [3] Ali Mazalek, Orit Shaer, Brygg Ullmer, Miriam K Konkel, et al. Tangible meets gestural: Gesture based interaction with active tokens. In *ACM CHI 2014 Workshop on Gesture-based Interaction Design, ACM CHI*, 2014.
- [4] Orit Shaer, Nancy Leland, Eduardo H. Calvillo-Gamez, and Robert J. Jacob. The tac paradigm: Specifying tangible user interfaces. *Personal Ubiquitous Comput.*, 8(5):359–369, September 2004.
- [5] Brygg Ullmer, Hiroshi Ishii, and Robert J. K. Jacob. Token+constraint systems for tangible interaction with digital information. *ACM Trans. Comput.-Hum. Interact.*, 12(1):81–118, March 2005.
- [6] Consuelo Valdes, Diana Eastman, Casey Grote, Shantanu Thatte, Orit Shaer, Ali Mazalek, Brygg Ullmer, and Miriam K. Konkel. Exploring the design space of gestural interaction with active tokens through user-defined gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14*, pages 4107–4116, New York, NY, USA, 2014. ACM.
- [7] Ahmed Sabbir Arif, Roozbeh Manshaei, Sean Delong, Brien East, Matthew Kyan, and Ali Mazalek. Sparse tangibles: Collaborative exploration of gene networks using active tangibles and interactive tabletops. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '16, pages 287–295, New York, NY, USA, 2016. ACM.
- [8] Stefanie Klum, Petra Isenberg, Ricardo Langner, Jean-Daniel Fekete, and Raimund Dachsel. Stackables: Combining tangibles for faceted browsing. In *Proceedings of the International Working Conference on Advanced Visual Interfaces, AVI '12*, pages 241–248, New York, NY, USA, 2012. ACM.
- [9] Brygg Ullmer, Hiroshi Ishii, and Robert JK Jacob. Tangible query interfaces: Physically constrained tokens for manipulating database queries. In *Proc. of INTERACT*, volume 3, pages 279–286, 2003.

- [10] Jamie Zigelbaum, Michael S. Horn, Orit Shaer, and Robert J. K. Jacob. The tangible video editor: Collaborative video editing with active tokens. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, TEI '07, pages 43–46, New York, NY, USA, 2007. ACM.
- [11] Casey Grote, Evan Segreto, Johanna Okerlund, Robert Kincaid, and Orit Shaer. Eugenie: Multi-touch and tangible interaction for bio-design. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '15, pages 217–224, New York, NY, USA, 2015. ACM.
- [12] Johanna Okerlund, Evan Segreto, Casey Grote, Lauren Westendorf, Anja Scholze, Romie Littrell, and Orit Shaer. Synflo: A tangible museum exhibit for exploring bio-design. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '16, pages 141–149, New York, NY, USA, 2016. ACM.
- [13] Seth Hunter, Jeevan Kalanithi, and David Merrill. Make a riddle and telestory: Designing children's applications for the siftables platform. In *Proceedings of the 9th International Conference on Interaction Design and Children*, IDC '10, pages 206–209, New York, NY, USA, 2010. ACM.
- [14] Ali Mazalek and Ahmed Sabbir Arif. Mobile-based tangible interaction techniques for shared displays. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*, pages 561–562. ACM, 2014.
- [15] Clément Pillias, Raphaël Robert-Bouchard, and Guillaume Levieux. Designing tangible video games: Lessons learned from the sifteo cubes. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*, CHI '14, pages 3163–3166, New York, NY, USA, 2014. ACM.
- [16] Yves Rangoni, Valérie Maquil, Eric Tobias, and Eric Ras. Implementing widgets using sifteo cubes for visual modelling on tangible user interfaces. In *Proceedings of the 2014 ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '14, pages 205–210, New York, NY, USA, 2014. ACM.
- [17] Brien East, Sean DeLong, Roozbeh Manshaei, Ahmed Arif, and Ali Mazalek. Actibles. In *Proceedings of the 2016 ACM on Interactive Surfaces and Spaces - ISS '16*. ACM Press, 2016.
- [18] Roozbeh Manshaei, Nauman Baig, Sean Delong, Shahin Khayyer, Brien East, and Ali Mazalek. Tangible mtDNA. In *Proceedings of the Tenth International Conference on Tangible, Embedded, and Embodied Interaction - TEI '17*. ACM Press, 2017.
- [19] Ahmed Sabbir Arif, Brien East, Sean DeLong, Roozbeh Manshaei, Apurva Gupta, Manasvi Lalwani, and Ali Mazalek. Extending the design space of tangible objects via low-resolution edge displays. In *Proceedings of the Tenth International Conference on Tangible, Embedded, and Embodied Interaction - TEI '17*. ACM Press, 2017.

- [20] Roozbeh Manshaei, Nauman Baig, Sean DeLong, Shahin Khayyer, Brien East, and Ali Mazalek. Exploring genetic mutations on mitochondrial DNA cancer data with interactive tabletop and active tangibles. In *Proceedings of the 2016 ACM on Interactive Surfaces and Spaces - ISS '16*. ACM Press, 2016.
- [21] David Merrill, Jeevan Kalanithi, and Pattie Maes. Siftables: Towards sensor network user interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction, TEI '07*, pages 75–78, New York, NY, USA, 2007. ACM.
- [22] Mathieu Le Goc, Pierre Dragicevic, Samuel Huron, Jeremy Boy, and Jean-Daniel Fekete. Smart-tokens: Embedding motion and grip sensing in small tangible objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, UIST '15*, pages 357–362, New York, NY, USA, 2015. ACM.
- [23] Jun Rekimoto, Brygg Ullmer, and Haruo Oba. Datatiles: A modular platform for mixed physical and graphical interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '01*, pages 269–276, New York, NY, USA, 2001. ACM.
- [24] Patrick Baudisch, Torsten Becker, and Frederik Rudeck. Lumino: Tangible blocks for tabletop computers based on glass fiber bundles. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '10*, pages 1165–1174, New York, NY, USA, 2010. ACM.
- [25] Liwei Chan, Stefanie Müller, Anne Roudaut, and Patrick Baudisch. Capstones and zebrawidgets: Sensing stacks of building blocks, dials and sliders on capacitive touch screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12*, pages 2189–2192, New York, NY, USA, 2012. ACM.
- [26] Simon Höfler, Alexander Müller, Johannes Luderschmidt, Nadia Haubner, Ralf Dörner, and Ulrich Schwanecke. Combining tangible and above the surface interaction. 2012.
- [27] Esben Warming Pedersen and Kasper Hornbæk. Tangible bots: Interaction with active tangibles in tabletop interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '11*, pages 2975–2984, New York, NY, USA, 2011. ACM.
- [28] Elise van den Hoven and Ali Mazalek. Grasping gestures: Gesturing with physical artifacts. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 25:255–271, 8 2011.
- [29] Ali Mazalek and Ahmed Sabbir Arif. Mobile-based tangible interaction techniques for shared displays. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services - MobileHCI '14*. ACM Press, 2014.
- [30] Garima Jain and Sanjeet Dahiya. NFC: Advantages, limits and future scope. *International Journal on Cybernetics & Informatics*, 4(4):1–12, aug 2015.

- 
- [31] Johanna Okerlund Casey Grote Brygg Ullmer Orit Shaer, Ali Mazalek. Tangible interaction with large data sets using active tokens. In *CHI 2015 Workshop on Exploring the Challenges of Making Data Physical*, 2015.
  - [32] Andy Wu, Jung-Bin Yim, Eric Caspary, Ali Mazalek, Sanjay Chandrasekharan, and Nancy J. Nersessian. Kinesthetic pathways: A tabletop visualization to support discovery in systems biology. In *Proceedings of the 8th ACM Conference on Creativity and Cognition*, C&#38;C '11, pages 21–30, New York, NY, USA, 2011. ACM.







