Morse Mentor
Design and Evaluation Report

Prepared by:
Omar Al-Ejel
Brian Epstein
Kanu Gaba
Justin Kirsh

Project Engineers

Las Tortugas

Prepared for:
Nikola Tesla
Little Toy Blue
Educational Toy Division

Project Dates:
12 March 2018 to 14 April 2018

Date Submitted:
17 April 2018

Executive Summary

Little Toy Blue has been interested in diving into the Educational Toy market since January, and our team was tasked with developing a micro-processor based toy prototype for a potential product. The CFO of Little Toy Blue supported the development of this prototype until April 14, and looks forward to seeing what may become a flagship product for the company. The purpose of this document is to evaluate our project's design process and the outcomes of our work.

Before thinking of toy ideas, our team explored the current educational toy market with visits to the Toys"R"Us and Learning Express toy stores located in Ann Arbor. At these stores, we conducted research on toys that are already available, in search of the most common input/output devices, what type of toys existed in the toy market, who appeared to be the targeted audiences, and what features had been successfully implemented. Understanding these key points allowed us to come up with an idea that is new and distinct. With that knowledge, we set out to create a toy that would be simple, inexpensive, and fun to play: the *Morse Mentor*.

At its core, the *Morse Mentor* is a toy with a Morse code button input that plays a Morse sound and gives the user feedback through a learning process. In order to construct a functional prototype that would serve our intended purpose, we decided to build three game modes that would provide users with a comprehensive Morse learning experience. The most critical learning tool in our prototype is the *Learn Mode*, which teaches the Morse alphabet. *Communication Mode* allows users to practice writing in Morse code with their friends wirelessly. Lastly, *Diary Mode* offers users a way to privately practice writing with Morse code.

Our prototype faced several developmental constraints. It had to be an educational microprocessor-based toy that used an Altera DE2-115 board and E100 processor. Because of this, we had to use input/output (I/O) devices that were compatible with this setup. We used a mouse as the user's method of input, as well as a VGA display and an audio port for output. XBee wireless modules allow for communication between boards across a distance and provide a multiplayer experience. The allotted development time spanned five weeks; this meant that we would need to maintain a steady pace throughout the development period in order to complete the prototype on time.

Our team is pleased to present a fully-functional prototype of the *Morse Mentor*, with three essential games modes that give users thorough experience with Morse code. The device uses auditory and visual stimuli to engage users, which we think will strengthen its appeal to the evolving toy market.

Given the *Morse Mentor* prototype's successful development and functionality, we believe that Little Toy Blue will choose to move forward with production of the final version of our toy this summer. We are prepared for any needed improvements like design modifications in preparation for production. Our proof of concept successfully demonstrates our vision for the final design of *Morse Mentor*.

Table of Contents

1 Introduction

Little Toy Blue's Educational Toy Division was offered funding by the company's Chief Financial Officer in order to pursue development of a new line of educational toys. Using this funding, our team has been given the task to conceptualize and develop an educational toy prototype that could be modified into a functional final product to be produced over the summer. Our team proposed the design idea for the *Morse Mentor* on the twelfth of March, and was authorized to progress with the development of our prototype. The purpose of this report is to provide an overview of the development of our project and self-evaluation of our work as a team.

Our toy, the *Morse Mentor*, is a microprocessor-based educational toy that allows our target audience of kids with ages nine to fifteen to learn, practice, and communicate in Morse code. Morse code is a system of transmitting information through a sequence of short and long signals referred to as *dits* and *dahs*, respectively. Letters and numbers each have specific Morse representations, and separate characters are distinguished by a pause in between.

The *Morse Mentor* offers three different game modes: *Learn Mode* for developing an initial understanding of Morse code, *Communication Mode* for allowing two users to wirelessly communicate with one another, and *Diary Mode* for practicing Morse code freely. Our toy prototype is comprised of speakers, a VGA display, a mouse, a DE2-115 board, and an XBee module for wireless communication with another *Morse Mentor*.

In the rest of this report, our team will discuss the original scope, design constraints, and motivations of our project. We will then provide an overview of our product design, the process behind the completion of this design, and an overall assessment of our design. We will then conclude with a summary of the information contained within this report and appendices for including other resources.

2 Project Background

After we, *Las Tortugas,* were hired by Little Toy Blue, we were given the opportunity to create a new microprocessor-based educational toy in order for Little Toy Blue to be able to enter a new market for educational toys, essentially expanding their consumer audience and raising profits. Upon approval of our proposed toy idea, the Chief Financial Officer offered project development funding until April 14, 2018 for the completion of a prototype of the final product's vision.

In order for Little Toy Blue to successfully enter a market that already has microprocessor-based educational toys, we had to produce a unique, immersive toy. To begin the brainstorming process for our toy proposal, our team explored Toys"R"Us and Learning Express in Ann Arbor, Michigan to evaluate the toys in the store. We were interested in which input/output devices were most common among the educational toys, what type of toys existed in the toy market, who appeared to be the targeted audiences, and what features had been successfully implemented.

After gathering and reviewing our data, we created a concept for a toy that would fit the company's criteria: interesting, fun, simple, and inexpensive. We call this toy the *Morse Mentor*.

Upon approval of our toy design on March 12, 2018, we set out to create a prototype of the toy that would showcase the necessary functionality to prove the success of a final, polished version. On April 14, 2018, various teams present their prototypes to Little Toy Blue and those that are selected by the company, are put into full-scale production during the summer of 2018.

2.1 Design Constraints

Our toy design was bound by several constraints, namely those that were technical or time-related. The main technical constraint placed upon the project was the requirement that our prototype for an educational microprocessor-based toy used the Altera DE2-115 board and an E100 CPU. We had to develop the prototype with this in mind, as it meant we were limited to using input and output devices that were available for use with the E100 processors.

Furthermore, we faced a funding deadline of April 14. This left us with approximately five weeks to complete our prototype. However, each team member still had commitments, such as work for other classes, that had to be taken into account. As a result, the amount of time that could be spent working on the project was limited. We would have to maintain a steady pace throughout the development period in order to provide a functional prototype by the deadline.

2.2 Motivation

Our goal to bring a unique educational toy to market led us to think deeply about the skills that consumers valued. Our market research showed that devices that taught concepts not available in the classroom were most successful. Members of our group were familiar with Morse, which helped us understand the potential of a device aimed at teaching the historic method of communication. Given our technical and organizational constraints, we proposed the *Morse Mentor*, with a focus on teaching Morse alphabet sequences through auditory and visual practice. The prospect of reviving a core communication system of the 19th century motivated the *Las Tortugas* team to take advantage of each group member's skills in hardware prototyping, graphic design, and multilingualism to create a successful prototype that would showcase the foundational features relevant to a final iteration of the *Morse Mentor.*

3 Design Overview

The *Morse Mentor's* final product design depended on the core features we wanted to include in our toy. We then limited those features to something feasible in the time given. The features we chose affected the input/output devices that we needed. Additionally, we had to envision a final product that would be appealing to children of our targeted age group of nine to fifteen year olds. The following section will discuss our final product design, the prototype design, and the reasoning behind each.

3.1 Product Design

A complete, production-ready *Morse Mentor* would be a textbook-sized portable device that contains a color LCD display, dual speakers, colored and labeled buttons, headphones, and a port for an included Morse key input. As shown in Figure 1, a Morse key is the historic means of inputting Morse; our hope to replicate such an input is part of our commitment to an immersive Morse communication experience.

Figure 1: A Morse Key. A user presses on the lever to signal a Morse input. (Poole, 2016)

The display in the center of the device should display contextualized information for each game mode with the purpose of allowing the user to visualize Morse dits and dahs. The buttons would be titled with each of the game modes: "Learn," "Communicate," and "Diary." Speakers would play Morse sounds at an authentic frequency of 750 Hz. Figure 2 shows a conceptualization of the final *Morse Mentor*.

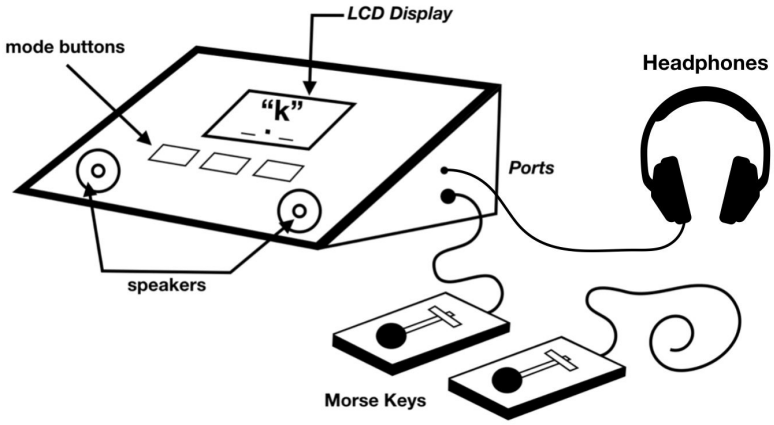


Figure 2: A mockup for a final iteration of the *Morse Mentor*.

3.2 Prototype Design

The *Morse Mentor* prototype is designed to demonstrate the essential features of the final product. This means that we emphasized functionality of game modes over the device's physical appeal. The device is made with a DE2-115 prototyping board, a computer mouse, an external speaker, headphones, and a VGA display. The prototype features all three game modes to be included in the final product; Learn Mode teaches the letters of the alphabet, Communication Mode allows users to wirelessly communicate with another *Morse Mentor*, and Diary Mode

provides a way to hear and see the Morse inputted by the user. The three game modes will be covered in more detail in section 3.3.

The *Morse Mentor* prototype does not feature a plastic enclosure to hide the circuit board and its circuit components. To switch between game modes, one must flip corresponding switches at the bottom edge of the DE2-115. Instead of using a 3D-printed Morse key for a more historically accurate input, the prototype gets its Morse code commands from a computer mouse's left-click button. Clicking down on the mouse button is very similar to pressing down on the lever of a Morse key, and most users are already familiar with computer mice and are comfortable clicking them. Figure 3 shows a photograph of the *Morse Mentor* prototype in the main menu screen. Figure 4 shows a labeled depiction of the same prototype.

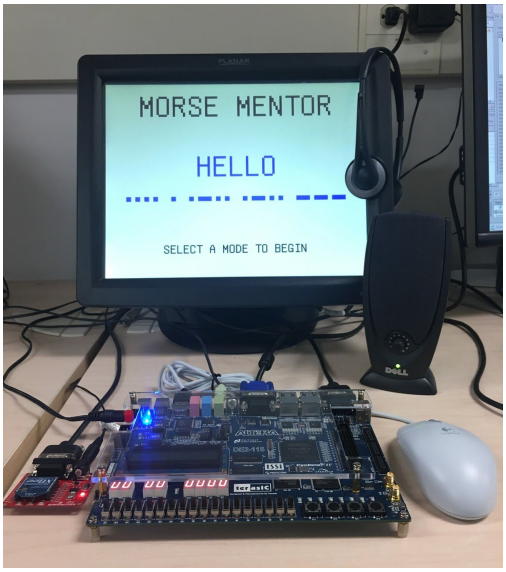


Figure 3: A photo of the *Morse Mentor* in the main menu.

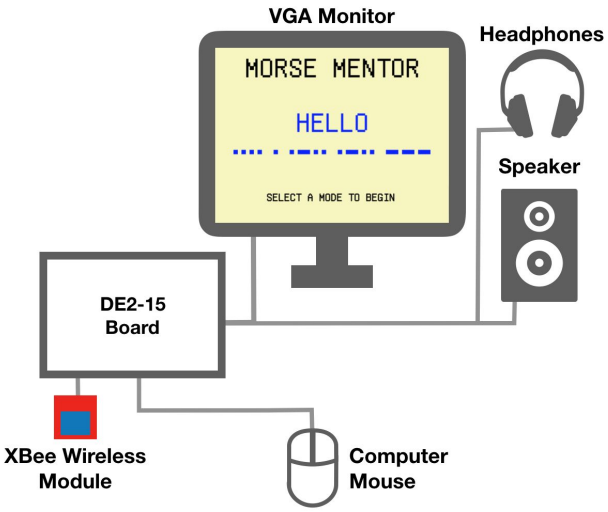


Figure 4: Labeled diagram of prototype and components in main menu.

Morse Mentor's audio is output through the DE2-115's line out port. Headphones or speakers can be connected to this port and are used to play both sample Morse audio and input by users throughout gameplay. Since the design of the prototype's visual interface is as crucial as its ability to play Morse, a bulk of the development time was put into intuitively displaying instructions, Morse feedback, and input boxes on screen. A friendly resizable font is used to display the name of the current device mode, and the highlighted titles make it easy to know the current game mode.

We also made use of XBee wireless modules for our Communication Mode, which featured a serial connection between two boards so that two users could converse in Morse code. Morse code was historically used to often transmit information over long distances, so we wanted to have a way for two users who are learning Morse code to be able to practice with each other even without being right beside one another. Our prototype also works with a serial cable

running between two *Morse Mentors*, but we wanted wireless functionality for a better and more authentic user experience.

3.3 Game Modes

The *Las Tortugas* team is proud of our accomplishments in the diversity of game modes written before the April 14th project deadline. Our goal to create game modes that allow users to thoroughly explore and practice Morse was made possible by our ability to aim for a modest target for the Learn, Communicate, and Diary Modes. The *Morse Mentor's* philosophy is that to rapidly learn a new method of communication, one must be exposed to examples of the language, tested on them, and then given the ability to practice with another learner as a corrective learning process. Figure 5 shows a screenshot of the main menu that greets the user once the prototype is powered.

Figure 5: The main menu screen, which greets the user in English and Morse, and prompts the user to select a game mode.

Learn Mode is the *Morse Mentor's* first step in bringing a user to Morse code mastery. As soon as the Learn Mode switch (switch 1) is pulled up, the user sees a prompt for entering the Morse for "A." As shown in Figure 6 (page 6), Learn Mode features a colorful title bar at the top of the screen, the letter of the English alphabet and Morse to be learned in the center, and a sequence of the alphabet that the user is expected to go through. The user clicks on the left button of the attached mouse to input Morse dits and dahs, and the *Morse Mentor* tracks the time taken between the press and release of the button to determine whether the user correctly understands the letter's Morse. If the user mistakenly inputs a signal that is too short or too long, Learn Mode highlights the central letter in red to indicate that the user should repeat that letter's Morse and follow the code on the display.

Whenever a new letter is taught by the *Morse Mentor*, the central letter on screen is updated, and the sound for that Morse is played on the speaker to give the user an accurate sense of the pacing to be used for inputting that letter. Figure 7 (page 6) shows how the progress characters "A" and "B" are highlighted in green to indicate successful understanding of the two letters' Morse code. After completing all 26 letters, the user receives a message of congratulations. *Morse Mentor*

then enters the quiz submode, in which the process is repeated on the same interface but without the assistance of Morse sound or code on screen. These two stages of Learn Mode work in conjunction to help the user memorize the Morse code sequences that make up each letter.

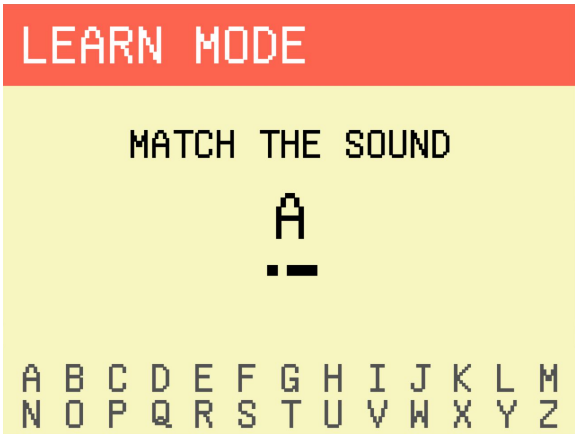


Figure 6: Learn Mode teaching Morse for the letter "A."

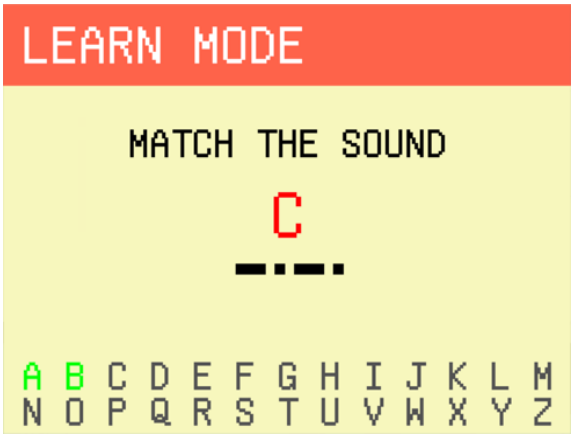


Figure 7: Red highlighting indicates incorrect input. Green indicates complete.

Once users have learned the alphabet's Morse code sequences, they will be able to test their skills in a fun and fast-paced Communication Mode. With two *Morse Mentor* prototypes, the attached XBee modules are used to wirelessly communicate Morse signals between devices without any noticeable delay. If a user presses a dit followed by two dahs, another user will hear that input and see it displayed on his or her screen, and vice versa. Figure 8 shows the screen of a user who received a message of 3 dits, 3 dahs, and 1 dit.

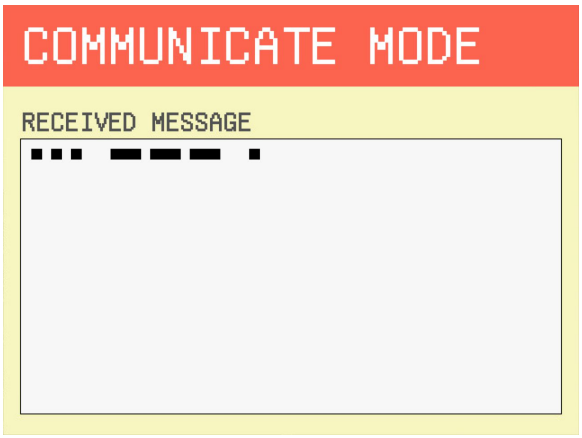


Figure 8: Communication Mode, showing a received message of 3 dits followed by 3 dahs and 1 dit.

Note that with in Communication Mode, it can become difficult to comprehend high frequency beeping sounds when they are simultaneously emitted from different devices. For this reason, the included headphones are useful in isolating Morse audio from a user's environment. In fact,

headphones add a touch of historical accuracy to the *Morse Mentor* experience, since telegraph operators of the 19th century wore headsets to focus on received Morse messages. Our team believes that Communication Mode motivates users to experiment with their new knowledge and develop an appreciation for a historic communication system.

If users do not have a friend with a *Morse Mentor*, they will still have the ability to practice their skills with Diary Mode. To reach this game mode, the user switches switch 3 up to be presented with a pseudo-text box as shown in Figure 9. When users press the mouse, they will be able to hear the 750 Hz frequency play from the plugged headphones or speaker. The team invested a large amount of time to have an accurate timing driver that is able to differentiate between a dit and dah, and then draw the corresponding symbol. Each Morse dit or dah on screen is drawn with an offset that allows the user to read their input from left to right, top to bottom. Note that the prototype is limited in how much information the user can input; if the bottom of the text box is reached, then the Morse on screen will be cleared for a new row of Morse. The final revision of the *Morse mentor* will also have the ability to save Diary entries onto an SD card for long-term storage.

Figure 9: Diary Mode, with an entered message that spans 4 rows

4 Design Process

Before submitting our toy proposal, we created a Gantt chart located in Appendix A to keep us on track throughout the project. This chart displayed what we would be working on, when we'd be working on it, and what parts of the project we could work on simultaneously. Due to the fact that we had about a month to complete a finalized prototype, we had to find a way to allocate our time so that the project's goals could reasonably be met. We decided that with each group member working about ten hours a week, for a total amount of forty hours per person, we would be able to successfully complete our project.

While continuing work on the prototype, however, our work plan had changed. These changes were attributed to problems that occurred causing certain tasks to get delayed, causing some to

even become impossible to complete within our deadline. For example, because the Learn Mode took longer than expected, we dropped one of our stretch goals of creating a 3D-printed enclosure. The Gantt chart on Appendix A displays the tasks we completed and those that we deferred till a later date.

4.1 Original Work Plan

When we first began the development of our prototype, we produced a comprehensive work plan to guide us towards its timely completion. To make it easier to manage our time, we divided this plan into four stages of completion: research, design, implementation, and testing.

In order to initiate the construction of our prototype, we first had to research the parts, drivers, and libraries that we would need to use. Our initial work plan included creating two Morse keys, which we had planned to develop to simulate an authentic Morse communication experience. In addition, the Learn Mode that we conceptualized was supposed to have two quiz modes, one for testing the user's knowledge of single letters and another for testing the user's ability to spell out words in Morse. The original goal was for our team to complete all of the necessary research in a total of three weeks.

After conducting most of the research that we needed, our team decided that the next stage of our development would be creating a detailed prototype design. Our plan was to design a toy with two game modes for both learning and communicating in Morse code. With some overlap with our research stage, we figured that we would need about four weeks to finish our prototype design.

We agreed that the next step in developing our prototype would be to implement our design, which would overlap quite a bit with our design and testing stages. Estimating that the implementation process would be a long one, we planned to work on this stage during the last five weeks leading up to the funding deadline.

Once we started completing basic tasks, we knew that we could begin our final testing stage. The easiest way to run all of our tests was to test individual components as we developed them, so we planned to start testing our prototype as soon as two weeks after the beginning of our work.

4.2 Work Completed

As a team, we surmised that we would need two DE2-115 boards, speakers and/or headphones for users to hear clicks, two VGA displays in order to individually learn Morse and also to communicate with another in Morse, and a serial connection between our two boards or two Xbee modules for our Communication Mode. We then wrote drivers for all of these input/output devices, ensuring that a wired serial connection would work on our prototype before we tried using a wireless connection. In addition, our team created two main libraries for use in our prototype: one for drawing letters on the VGA screen and another to determine if a dit or dah was written.

To start the implementation process, we wrote the necessary drivers and conducted small practical tests, created the library for drawing letters onto screen, and created the library that

determines if a dit or a dah was written. We then created an array that contained all of the Morse sound codes that we needed, another array that contained all of the letter bitmaps for displaying text onto the screen, and a basic program skeleton to display all the different game modes that our toy offers. All that we had left to do was to combine the different elements that we had already created into the Learn Mode, and then take this a step further by allowing for communication between two DE2-115 boards. With extra time before the deadline, we were able to implement our stretch goal of a Diary Mode for users to write freely, adding slight modifications to the Communication Mode to make this possible.

The Learn Mode was created to include a way to practice individual letters in Morse code with the help of audible Morse code letters and printed Morse on the VGA screen. The Learn Mode that we developed included a quiz mode for testing the user's knowledge of single letters in Morse. The other game mode that we developed was Communication Mode, which would allows for communication between two users so that they can send Morse code messages to one another. Specifically, we engineered our prototype so that the audible and visible Morse code from one user's clicks are sent to another user and vice versa. Our Diary Mode was designed using much of the same interface and programming from the Communication Mode.

As we began to complete project tasks, we tested our input accuracy because we wanted to be absolutely sure that users' clicks would be recognized as the dit or dah that they intended it to be. Concerning the drawing of Morse code onto the screen, we tested our implemented VGA drawing tools so that we could display the proper information to the user. Our team was also concerned that we would run into issues concerning data loss when we implemented our Communication Mode, so we tested our serial connection several times to see if it was functioning correctly.

4.3 Problems and Solutions

While we worked to produce a functional prototype, we encountered a couple of key challenges that halted our further work. These essentially pushed back the start of other tasks and altered the final scope of our working prototype. In this section, we will discuss the main problems that we had and how we solved them.

4.3.1 Limited Bandwidth

The largest problem we faced during the development of *Morse Mentor* was allowing the communication game mode to work wirelessly using the XBee modules. The boards were originally unable to communicate properly, which meant that mouse inputs by the other user would not be recognized correctly and sound would play or fail to at the wrong times. However, Communication Mode worked properly when data was transmitted through wired serial cables instead. We discovered that this problem was likely due to the XBee modules' limited bandwidth. Our original program would constantly send that board's left mouse button state through the serial port for every loop of the Communication Mode's main function. Because the program would traverse each loop very quickly, this method of sending data would exceed the XBee modules' bandwidth and prevent Communication Mode from functioning when using a wireless connection.

In order to solve this problem, we had to develop a program that would allow one board's left mouse button state to be known to the other board in real-time without going beyond the XBee modules' limit. The solution we came up with was to change the function so that it would only queue the left mouse button state to be sent through the serial port when it had changed since the last time it was sent. This way, the receiving board could just store the last received state and continue playing the speaker at the appropriate times because if it ever had to change it would just receive a signal saying to do so. This solution allowed Communication Mode to retain its function and with no noticeable delay. After implementing our solution, the same information describing the mouse state was transmitted between the boards but much more efficiently.

4.3.2 Proper Timing

Another major issue that we faced during development was correctly interpreting differences in timing for the purpose of distinguishing between dits, dahs, and spaces. In Morse code, there are typically only two possible states for the signal being used to transmit information. In the case of audio these states would usually be whether the sound is currently on or off, and it is the same for many other methods of communicating in Morse code, such as turning a flashlight on and off. Because of this, the only way to distinguish between dits and dahs is the timing. Dits act as the short signal whereas dahs are longer. Typically communication in Morse code has a person on the receiving end interpreting the signals being sent, but this is not the case in *Morse Mentor*'s single-player modes. Different people would have their own timing habits when communicating in Morse code, similar to how people can have different accents in speech, so we needed our toy to be able to account for these differences. In addition, we needed to implement these timings in the toy's Learn Mode when it would play example Morse code audio for the user. *Morse Mentor* would determine how to play the Morse code on the spot instead of simply playing a premade sound file from an SD card.

Our solution for interpreting the timing of the user's inputs as dits, dahs, and spaces was to use a reasonable length threshold. Inputs by the user that were shorter than our threshold would be interpreted as dits, while those that were greater than the threshold would be interpreted as dahs. This was essential for Learn Mode, in which the toy has to interpret whether the user's input is correct for each letter. This timing threshold was also used to properly display the received Morse code in Communication Mode as well as the entered Morse code in Diary Mode. In addition, we included a threshold that would automatically add in a space in the Diary and Communication Modes if the person inputting Morse code pauses for a long enough time. As stated before, the only thing distinguishing different inputs in Morse is the timing. The letter "E", for example, consists of one dit in Morse code, while the letter "S" consists of three. The reason why three "E"s can be told apart from one "S" is because of the pauses between each dit. We conveyed this by adding spaces in displayed Morse code if the delay between a user's inputs was greater than the chosen threshold.

Because the example Morse audio in Learn Mode had consistent timings due to being produced by the toy itself, we were able to hardcode particular lengths for dits, dahs, and pauses. We really wanted to create a function that would play the correct audio given the sequence of dits and dahs that made up a letter so that we could potentially play entire words in a similar manner without having to record them all beforehand. To accomplish this, the current time according to the clock

would be stored, and the speaker would play our Morse sound until a later time determined by adding the length of the current dit or dah to the previously stored time. After completing that dit or dah, *Morse Mentor* would use a similar method to stay silent for a short period of time before continuing to the next dit or dah in the sequence that represented that letter. This would continue until the letter's audio was done being played, or until the user clicked the mouse button, in which case that letter's example audio would immediately cease playing.

5 Design Assessment

Overall, our team was able to successfully deliver a prototype with the promised goals and functionality determined during the proposal. Although we weren't able to accomplish all our stretch goals, and although we made our quiz mode slightly easier, we believe that our prototype is able to showcase the functionality of our final toy.

Overtime, we were able to notice limitation of our prototype that would definitely affect the marketing of our final product. However, we believe that these drawbacks would be able to be fixed before production this summer. Discussed in the sections below are our design rationale, drawbacks, and the skills we gained from working on this project.

5.1 Design Rationale

We are confident that the *Morse Mentor's* will be a successful addition to Little Toy Blue's line of educational toys once consumers are allowed to make a judgement on the device's value. The success of the *Morse Mentor* is dependent on both the the market's ability to recognize its educational and entertainment value, and the feasibility of developing the device.

When the *Las Tortugas* team developed the *Morse Mentor*, we emphasized the importance of creating an educational toy that could actually teach users a new skill. Most toys marketed as educational are only tangentially related to a school-related subject, and focus on the entertaining aspect of the product. With the *Morse Mentor*, we followed a philosophy that to rapidly learn a new method of communication, one must be exposed to examples of the language, tested on them, and then given the ability to thoroughly practice in a rewarding experience. The progression through Learn Mode, its quiz submode, Communication Mode, and Diary Mode encompasses the experience with Morse code that we believe will allow consumers to appreciate modern means of communication.

The entertainment factor that comes with the *Morse Mentor* is supported by the fact that it uses auditory and visual stimuli to induce quick, calculated physical responses from users. In product demos, we were fascinated by student experimenters who felt the need to race through Learn Mode, absorbing each letter's morse sequence. Once shown the ability to communicate wirelessly, students experimenters smiled, and seemed to have an interest in encoding personalized messages to send to their peers. While the experiences of a dozen students interacting with the *Morse Mentor* may seem anecdotal, we believe that there is a strong likelihood that the average teenager will enjoy the device's diverse features.

The feasibility of the game modes and hardware used to make the *Morse Mentor* reduces the production risks involved with a new, unique product. Since the *Morse Mentor* is comprised of distinct game modes, each mode can act as a standalone feature, and is thus expendable in the case that not enough time is available to perfect the mode. This also means that new, separate modes can be added in the same way that one can install as many apps onto a phone as desired.

In regard to device hardware, we believe that we have maximized the benefits of the components used in the *Morse Mentor* prototype. Headphones, the XBee wireless module, and a simple mouse input are a simple set of input/output devices with relatively low costs. Not having to depend on expensive peripherals makes it easy to create a low-cost proof of concept. Now that we have demonstrated the functionality of the prototype, it is reasonable to discuss adaptations of the input/output devices that make them more integrated in the Morse experience. For example, the mouse will likely be replaced with a 3D printed Morse key, and the headphones will have the Little Toy Blue brand name embossed on the curved headband.

The ability to rapidly modify our code base is supported by the fact that the Diary Mode was made in less than one hour of work because its foundation was taken from Communication Mode. The abstraction used for repetitive processes, such as drawing text to screen, drawing morse to screen, playing morse, and loading bitmaps makes it easy to test changes to the *Morse Mentor* at the request of Little Toy Blue. The result is a streamlined development process that only increases the chance that the prototype will become a successful toy that elevates this company's rank among toy producers such as Mattel and Hasbro.

5.2 Design Drawbacks

Our prototype was created to show that we can develop the necessary functionality to demonstrate that a final product of the toy is feasible. Because of this, limitations and drawbacks are present within our final prototype.

One of our most important limitations is the range of the XBee modules. The *Morse Mentor*'s Communication Mode is to be played with any other user in possession of another *Morse Mentor*; however, the XBee modules that we utilized have a limited range of 300 feet. This means that in order for two players to be able to to communicate with each other in Morse code, they will be required to be in a close proximity of one another. The goal of our final product is to have the Communication Mode work through a WiFi connection, allowing any user with internet access to be able to find and communicate with any user of the *Morse Mentor* in the world.

Another drawback of our prototype is the portability. We were never able to successfully complete our stretch goal of 3D-printing an enclosure for the various input and output devices of our prototype. As a result, our prototype contained many different components with many wires, along with an extremely large and heavy VGA monitor, making the *Morse Mentor* virtually non-portable. However, for our final product, we plan on creating a complete enclosure to hold the wires, input/output devices, and the microprocessor and development board.

Because Morse code depends on timing, our toy needs a way to interpret the timing of a user's inputs. Different users likely have different habits or "dialects" for their inputs that might not be

major enough to confuse another person, but enough to mismatch the *Morse Mentor's* interpretation. When two humans are communicating in Morse code, they can more easily comprehend the received message with clues like sentence context that minimize the effect of mistakes in communication. In Learn and Diary Mode, the *Morse Mentor's* timing function is what attempts to differentiate dits from dahs. The timing thresholds distinguish dits from dahs and interpret input pauses as spaces, but it is possible that there are experienced Morse users who makes their dits and dahs significantly shorter than our lower bound for dah length. Our thresholds were determined by finding reasonable lengths for what constitutes a dit versus a dah, but for the final product, we would like to include a custom calibration option that gives users the ability to modify the threshold.

5.3 Skills Gained

Our work on the prototype has shown us the importance of Git in sharing our work and having it organized in one place that is accessible to everyone in the group. When one group member finished working on a specific task, he could easily upload his updated work so that others could work on it without conflicting edits. Git proved to be invaluable tool in the success of our project, and thanks to the experience we have gained with it while working on this project, we can put it to good use in future team projects.

Having written many drivers for our final prototype, we have significantly improved our code debugging skills. This is a crucial programming skill that is especially beneficial with assembly programming, which is nearly impossible to do right the first time. The ability to efficiently scan our code for errors and then quickly fix those errors allowed our prototype function successfully.

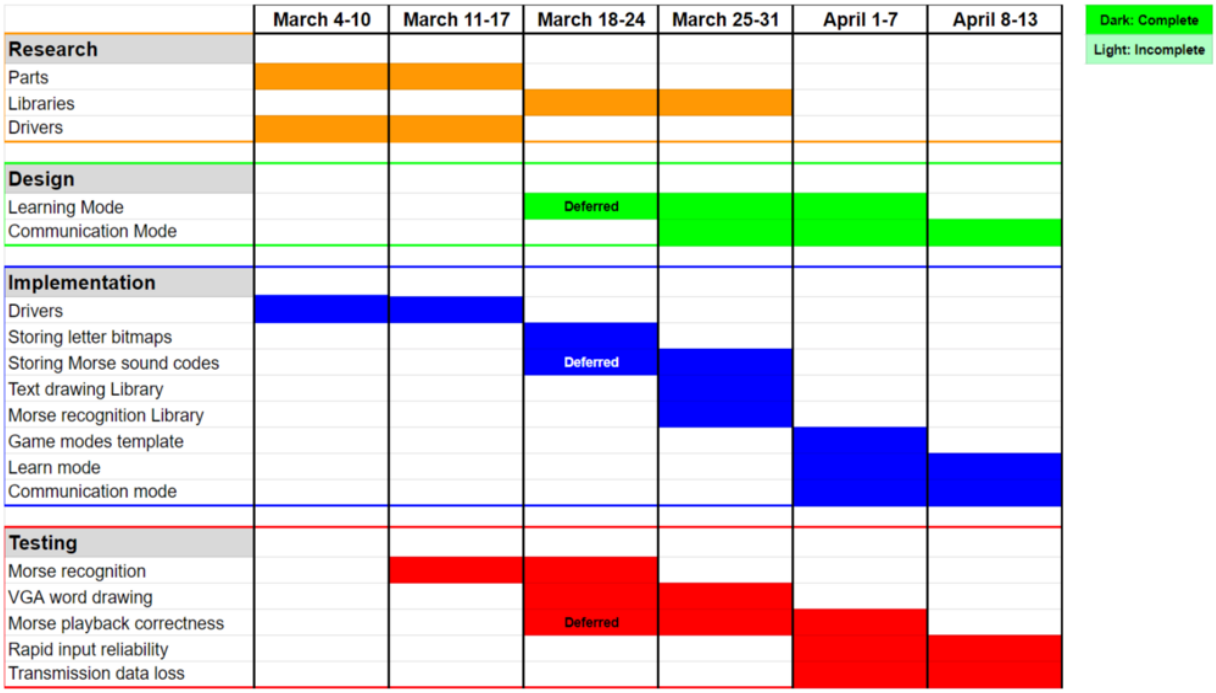
We have also gained extensive knowledge of using input/output devices in the production of our prototype. During the first stages of our project, we created drivers for the speakers, mouse, VGA display, and serial communication. Our work with these drivers gave us understanding of low-level functionality that will inform the way we perceive technology at a high level.

References

Poole, I. (2016, April 06). Morse Telegraph Key History & Development. Retrieved April 17, 2018, from https://www.electronics-notes.com/articles/history/morse-code-telegraph/morse-key-development.php

Appendix A: Gantt Chart

This appendix includes the final version of the project Gantt chart, updated April 14, 2018. Boxes with dark highlighting (now all of them) refer to complete subtasks. Boxes marked "deferred" pertain to subtasks that could not be completed at the planned time as a result of having to dedicate more time to other portions of the project.



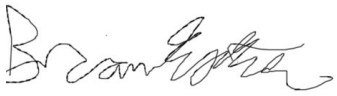
Appendix B: Honor Code Pledge

This appendix includes a signed University of Michigan Honor Code Pledge to testify that this project was completed without violation.

Honor Code Pledge

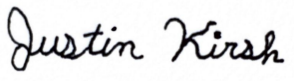
"We have neither given nor received unauthorized aid on this assignment, nor have we concealed any violations of the Honor Code."

Omar Al-Ejel



Brian Epstein

Kanu Gaba



Justin Kirsh

Name	Work Contributed	Total Percentage
Omar Al-Ejel	Executive Summary, Motivation, Design Overview, Product Design Prototype Design, Game Modes, Design rationale, images and figure descriptions, references	26%
Brian Epstein	Executive Summary, Design Constraints, Problems and Solutions, Introduction, Prototype Design, Design Drawbacks	24.33%
Kanu Gaba	Executive Summary, Background Information, Design Overview, Design Process, Problems and Solutions, Design Assessment, Design Drawbacks	24.33%
Justin Kirsh	Executive Summary, Introduction, Original Work Plan, Work Completed, Problems and Solutions, Design Process, Design Assessment, Skills Gained	24.33%