Building Computational Thinking Skills via Exergames

By

Alka Harriger
Professor
Purdue University

Bradley Harriger
Professor
Purdue University

Susan Flynn
Instructor
College of Charleston

Abstract

Teaching Engineering Concepts to Harness Future Innovators and Technologists (TECHFIT) was an NSF-funded science, technology, engineering, and math (STEM) project (DRL-1312215) (Harriger B. , Harriger, Flynn, & Flynn, 2013) that included a professional development (PD) program for teachers and an afterschool program for students. Curriculum and Assessment Design to Study the Development of Motivation and Computational Thinking for Middle School Students across Three Learning Contexts is an NSF-funded research project (DRL-1640178) (Harriger A. , Harriger, Parker, & Li, 2016) that examines the impact of delivering the TECHFIT curriculum to middle school students in three different contexts: afterschool program, in-school class, core class module. Thus far, the new project has deployed TECHFIT using the first two contexts, both of which use the entire TECHFIT curriculum. The goal of the TECHFIT curriculum is to spark interest in STEM and computational thinking (CT) in middle school students. The curriculum employs two computer programming tools as well as physical computing to introduce participants to STEM and CT. It also includes use of brain blasts to engage participants in a wide variety of physical activity throughout the instruction as well as to enrich their imaginations with different ways to make movement fun. This paper focuses on the process of exergame development using TECHFIT tools as a way to support CT skills development. The process is illustrated using a complete example from inception to a picture of teachers testing the working, physical exergame.

Keywords: computational thinking, STEM, exergame

TECHFIT is now in its sixth year of implementation. The TECHFIT approach involves having individual and teams of middle school teachers successfully complete a week-long, professional development (PD) workshop through which they acquire the essential skills that contribute to the task of innovating an exergame. An exergame is a technology-supported fitness game that can educate/train, assess performance, and/or entertain the player(s). Teachers demonstrate their understanding of concepts and tools taught in the PD by implementing their own functional exergame invention by the fifth day. Then during the subsequent school year, the teachers teach their students the same skills they learned in the PD, so their students can innovate their own exergames. Successful completion of the curriculum addresses the majority of standards for technological literacy identified by International Technology and Engineering Educators Association (2020).

This paper will walk the reader through the process of creating an exergame that will employ externally-wired input and output components. Included are several screenshots to illustrate artifacts from each step in the exergame development process. Instructions on using the built-in simulator to verify accurate functionality of the exergame are included. The program code for the exergame created is provided in the appendix. Links and instructions for downloading the free software to create the illustrated exergame, along with an opportunity for the reader to acquire the needed technology at no cost are shared. Although specific tools are used in the TECHFIT curriculum, the approach and process should be applicable to other tools.

Background

As of December 2018, the TECHFIT team has delivered 10 professional development (PD) programs to 65 teams of middle school teachers from 13 different states: Indiana, South Carolina, Ohio, Washington, Tennessee, Wisconsin, Michigan, Virginia, North Carolina, New

York, Kentucky, Illinois, and Florida. Thus far, over 125 middle school teachers and over 2000 middle school students have successfully completed TECHFIT programs.

TECHFIT program content includes computer programming, engineering design, electrical and mechanical engineering technology, game design, exercise science, and fitness, all with a common goal of innovating an exergame. The objective is to show participants how STEM and computing skills can be used to solve big, societal problems. In the case of TECHFIT, the problem being addressed is physical inactivity (Sween, et al., 2014), and by extension obesity.

Both groups, teachers and students, learn how to design and develop physical, technology-supported fitness games (exergames) that occupy a large space on a gym floor and allow multiple players to play and/or compete. The process of learning how to innovate exergames teaches essential STEM skills using a team-focused approach, but more importantly sparks enthusiasm for computing, engineering, and technology.

Computer programming concepts are taught using two tools, Scratch (Lifelong Kindergarten Group, 2018) and NanoNavigator (CNET, 2018). Scratch is a block programming tool that TECHFIT uses to create animations to explain visually how to play the exergame. There is considerable literature on Scratch, so this paper will focus on using NanoNavigator, which is a flowchart programming tool that is used to develop the program logic for the physical exergame.

In addition to learning computer programming through these two tools, everyone gains hands-on experience with physical computing through provided instruction on wiring electronic components to a microcontroller—the Nanoline (Phoenix Contact, 2018) controller by Phoenix Contact, which is shown in Figure 1. It should be noted that the controller was meant to be used in industrial manufacturing applications, so its design is robust enough to be nearly indestructible, unlike comparable educational tools that may be used to also develop exergames such as Arduino.

The curriculum also intersperses fun, physical activities called brain blasts throughout regular instruction. Some brain blasts last five minutes, while others may last twenty minutes or more. The integration of the shorter brain blasts help keep minds alert, resulting in better retention (Justin, 2013). The collection of all brain blasts also provide ideas for activities to be included in the exergames that will ultimately be invented by the participants. Once the physical exergames are built, they may be used to conduct some of these brain blasts.

Each teacher team receives a technology toolkit that includes the Nanoline microcontroller, wires, push buttons, lights, sensors, other related technology and documentation. The toolkit components are acquired through generous donations from Phoenix Contact, Balluff, Automation Direct, as well as purchases made by the project team leaders. Although each toolkit is valued at over $4500, each teacher team receives a complete toolkit at no cost to them with the understanding that they will fully implement TECHFIT at their schools.

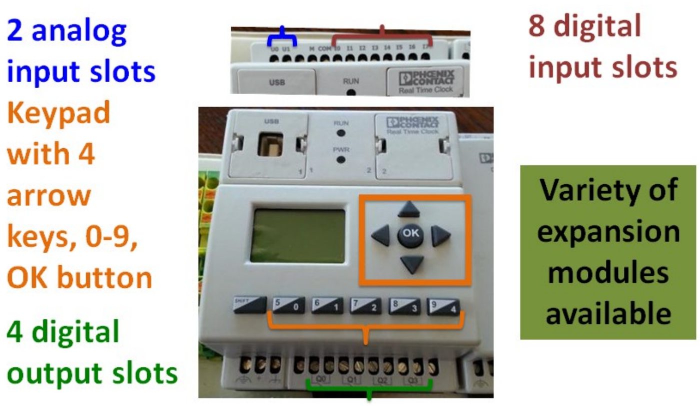


Figure 1. Phoenix Contact's Nanoline microcontroller

During the six-day TECHFIT PD, the majority of the instructional content is covered in the first three days. Daily homework gives the teachers opportunities for additional practice and to demonstrate their understanding. Questions from the homework are reviewed at the start of the next day. To further demonstrate understanding of these concepts, each teacher team is required to successfully implement their own exergame invention by the fifth day of the PD. On the sixth day, each team adapts the provided curriculum to their school's unique needs and circumstances, and presents their ready-to-implement plans to the rest of the group for feedback.

During the subsequent school year, each teacher team is expected to deliver a 30+ hour afterschool program or STEM class at their school. The content is the same as what they learned in the summer PD, but their students fortunately have more time to digest the content gradually. The student teams are also expected to innovate their own, large-group exergame using the same technology toolkit.

The TECHFIT Showcase

In early December, a showcase event has been hosted by Purdue University to give the top teams an opportunity to present their learning experiences and demonstrate their exergame to a public audience as well as to an invited panel of judges. The judges evaluate all presentations using a web application to help select the showcase champion. All student teams are recognized for noteworthy elements of their presentation/exergame, and the showcase winner is presented with the championship trophy.

In the first four years of TECHFIT, all participating schools were invited to the showcase; however, in 2017, there were too many teams to hold a one-day showcase. For this reason, a video showcase was instituted as a way to select up to the top eight teams most likely to have a competitive exergame project.

The video showcase occurs in early November, so that there is enough time for invited teams to make travel arrangements. Teams not selected are encouraged to attend as observers to take advantage of the educational opportunity of seeing the breadth of exergame inventions created with the same toolkit materials and educational curriculum.

The short video that each team is expected to submit to the video showcase must address all of the following:

1. Identifies teams and status of work completed by each.
2. Describes the exergame (some teams set up their game without technology and explained how/what technology will be used).
3. Shares the Scratch animation that explains the team's vision for the exergame.
4. Shows some aspect of the physical exergame that is working.
5. Includes a viable plan (who is doing what with a timetable) that convinces project leaders that they will be finished and ready to compete at the final showcase event.

The video showcase includes a public voting opportunity, so all teams are encouraged to spread the word about the video showcase to their stakeholders. The team with the most public votes receives an automatic invitation to the final showcase event; however, this public vote is more than a simple popularity contest. In order to vote, the voter must assign up to five (5) points to a minimum of three (3) teams for the votes to be counted. The total votes are visible only at the beginning of the voting period, but become hidden on the last two days. Based on Google Recaptcha Analytics from the 2017 video showcase, the TECHFIT website was visited nearly 4000 times in four selected days of voting.

A judging panel is used to select the remaining showcase competitors. They use a web application that includes criteria consistent with the final showcase, but takes into account that teams will have almost a month to finish their exergame.

Exergame Development Process

The target group is middle school, so it is important to instill good programming habits from the start. The exergame development process taught to teachers is designed to aid in the development of an exergame using the provided software, nanoNavigator, and the physical Nanoline controller. This process is illustrated in Figure 2 and explained in further detail through a step-by-step example that follows.

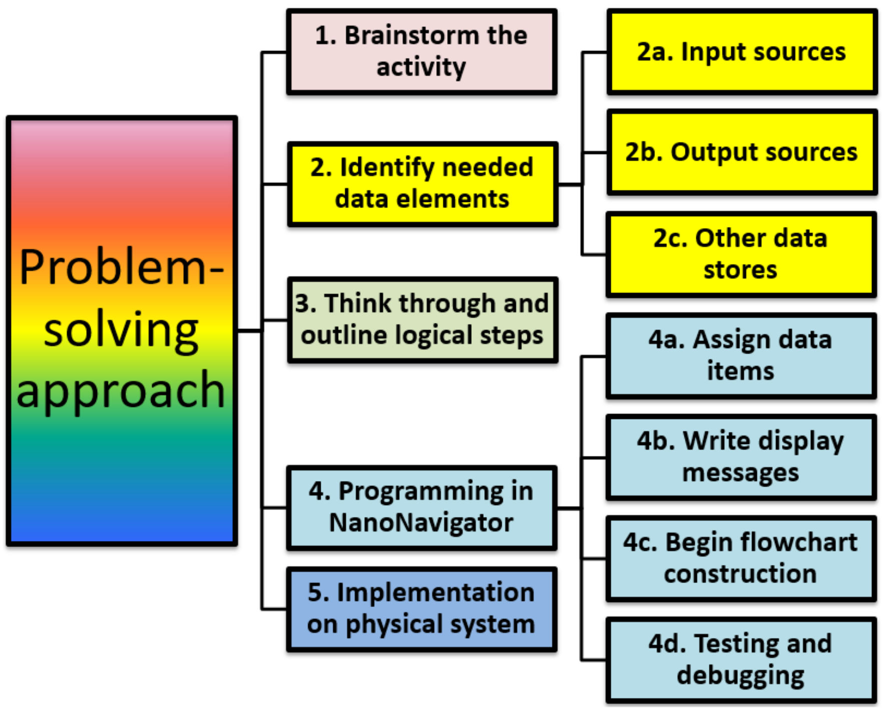


Figure 2. TECHFIT's exergame development process

The process is introduced to the teachers before any programming is done. Then the TECHFIT leaders facilitate development of a simple exergame to illustrate each step of the process. The artifacts produced during the process are shared as resources for the teachers. Refer to Figure 3 to view an artifact of step 2 in the process (identify needed data elements). It shows the notes that were made immediately after brainstorming a jumping game to be created.

Given that a physical system will be created, teachers are taught to plan the types of electronic components that would be needed to build the exergame. Anything that provides information to the exergame system, such as input from keypad keys or movement detected by a sensor, are categorized as input. Anything that the exergame system provides to the player, such as turning a light on or displaying a message on the screen of the operator panel, are considered output. Any other data used in the process of running the exergame are also given a descriptive name. All data items needed by the exergame are categorized into three columns: Input, Output, and Other Data Stores, as shown in Figure 3. This approach simplifies development of the flowchart program in a later step.

Input	Output	Other Data Stores
StartButton JumpSensor NextPlayer	Instructions JUMP DetectLight StopBuzzer GameStats	JumpTimer JumpCounter

Figure 3. List of data items for simple jumping exergame

Figure 4, an artifact of step 3 (think through and outline logical steps), depicts the logical outline that describes how the jumping exergame works.

1. Show Instructions
2. When player presses start button
    a. Reset jump timer
    b. Reset jump counter
    c. Show JUMP message
    d. Is there still time to play?
        i. When jump is detected
            1. Increment counter
            2. Pulse the detect light
    e. Sound the stop buzzer
    f. Stop the timer
    g. Show game stats
    h. Return to start when the next player button is pressed

Figure 4. Logical outline of simple jumping exergame

The logical outline was developed through an extended group discussion with the participants. It helps show how the logic for a simple exergame can get complex quickly when every detail needs to be written down.

Once all participants have practiced all of the steps many times and are ready to begin work on the final exergame team project, it is very likely that the groups that develop the flowchart program are not the same ones who will be constructing the physical, wired system. For this reason, it is imperative that all groups agree in advance regarding how specific data stores on the Nanoline controller will be used, particularly the externally wired input and output components. Therefore, before opening up NanoNavigator, remind participants that they are creating a physical exergame using a specific controller, the Nanoline microcontroller that was previously shown in Figure 1. Have everyone agree on where each of the external components would be wired to the controller. Return to the data item table shown in Figure 3, and mark the specific slot labels where the external input/output components will be wired. The resulting table is shown in Figure 5.

Input		Output		Other Data Stores	
I-0	StartButton	MI-0	Instructions	TC-0	JumpTimer
I-1	JumpSensor	MI-1	JUMP	TC-1	JumpCounter
I-2	NextPlayer	Q-0	DetectLight		
		Q-1	StopBuzzer		
		MI-2	GameStats		

Figure 5. Updated data items list with Nanoline locations for simple jumping exergame

Before programming in nanoNavigator, the types of blocks available within the tool are reviewed. These blocks are shown in Figure 6. When developing a flowchart, the Control, Math, Move, and Message blocks have exactly one entry point and one exit point. The Decision, Compare, and Wait blocks have exactly one entry point, just like the other blocks, but two exit points (true, false) because the content of the block is a Boolean expression. The exit path that will be followed depends upon the value of the Boolean expression at that particular point during execution. The Notes block is really not part of the flowchart logic, but it is useful to document sections of the logic to explain their function in English.

Figure 6. Control blocks in nanoNavigator

After opening NanoNavigator and configuring it to match the controller and basic setup to be used, the flowchart editor page appears with the starting terminal symbol at the top. However, before beginning to develop the flowchart, the data items to be used by the program should be named descriptively. To name the data items, use the notes from Figure 5 to match the Nanoline's names to descriptive labels that will make it easier to follow the program logic.

Once the data items have been named, the text of messages for the operator panel screen should be constructed. The message can contain text in either a 4x20 grid or a 2x10 grid, where each cell in the grid can hold a single character. If the message should display the content of a data store, then 8 contiguous cells on the same line are needed. This requires the programmer to be very concise in wording the messages.

Multiple messages could be used, but additional steps in the logic are needed to define how to move from one message to the next. During the initial learning stage, single messages are encouraged, so the attention can be placed on the logic. Figure 7 shows sample messages for the simple jumping exergame.

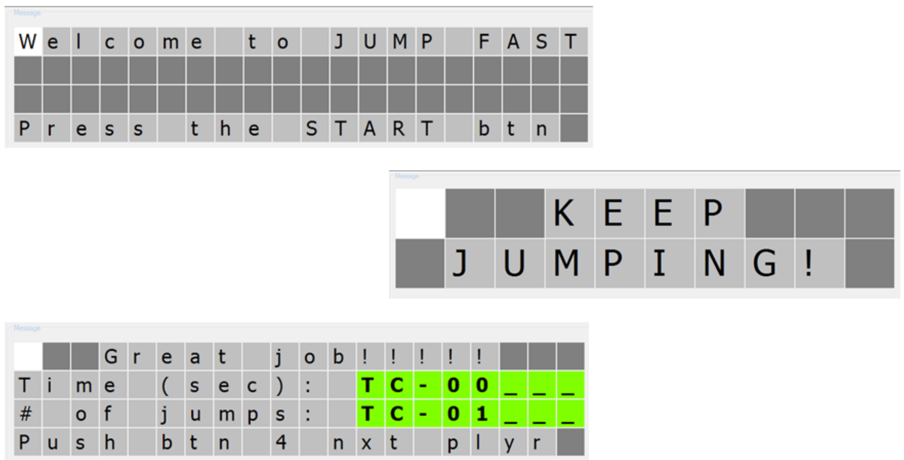


Figure 7. Predefined messages for the simple jumping exergame

Next, the group steps through the logical outline to construct the corresponding flowchart. Once it is finished, the built-in simulator is used to test the accuracy of the program. Figure 8 has a snapshot of the simulator during a test run.

Once the program works as expected, the program is downloaded to the wired controller while still connected to the computer. This allows the programmer to watch the program run alongside the simulator to verify that the data stores are changing as expected while the program runs. Figure 9 shows one of the teacher teams testing their physical jumping exergame.

Appendix A shows the entire flowchart to allow the reader to view the logical flow of control. Appendix B has four pages that contain quarter sections of the program from top to bottom to allow the reader to see and be able to read the content inside each program block.

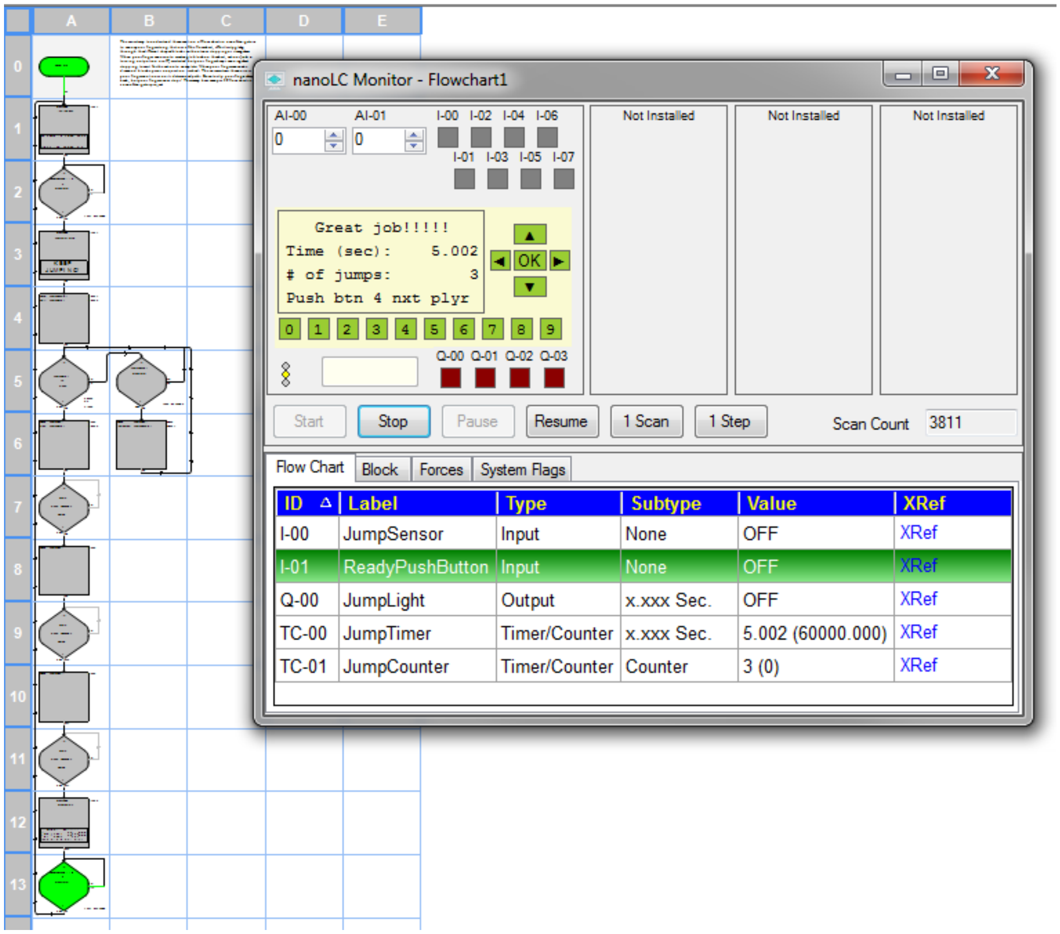


Figure 8. Testing the flowchart program using the simulator

Figure 9. Testing the physical exergame on the wired controller

One Way to Acquire the Technology

The nanoNavigator software used to program the microcontroller requires a Windows-compatible computer. The download is available at no cost from the Phoenix Contact website (https://www.phoenixcontact.com). It is also available from CNET at https://download.cnet.com/NanoNavigator/3000-2383_4-75330441.html. The software is an effective tool for getting students to express their logic visually in a way that can be understood by others. The built-in simulator is also very effective in helping students test their program logic and troubleshoot unexpected behavior of their program.

The greatest feeling of success happens once students are able to download the flowchart program to their wired, physical system and see it run. However, the cost of a basic system that includes the microcontroller, display interface, programming interface, power supply, power cable, jumpers, terminal blocks, ferrules, tools, wires, and input/output components can cost several hundred dollars. Learning proper wiring also takes some time.

Fortunately, for teachers who are interested in creating an opportunity for their students to gain practical experience with flowchart programming and physical computing, Phoenix Contact sponsored a free contest annually called the Nanoline Contest (Phoenix Contact, 2019). The contest targeted high school students, but there were several middle school teams that participated over the years, including one that won first place. In 2021, they plan to launch another student automation contest, but continue to make the Nanoline kit available to people who contact them.

If the new contest follows the process of the Nanoline contest, registration generally happens in the summer months. To register, each team of up to six students with an adult mentor submits their idea for creating an automated Nanoline controller system. Upon acceptance of a proposed project idea by the team, a basic toolkit and a generous gift card are shipped to the mentor. Teams may use the gift card to purchase sensors, wood, or any materials to aid them in the process of building their project. Teams must pass several checkpoints to remain in the contest. By January, teams that convince the contest managers of their ability to have a working system by the regional contest date in early February are then invited to compete.

Each region selects national qualifiers who will have two weeks to potentially improve their project based on judges' feedback. Phoenix Contest arranges travel to Harrisburg, Pennsylvania for all national qualifiers. At the national contest, the top overall project team receives an all-expense paid trip to Orlando, Florida. The second and third place teams receive cash awards.

Some tech-ed teachers have used the contest as an integral capstone element of their class. If they have more than six students, they have multiple teams submit project ideas, and each team received their own toolkit and giftcard. Other teachers (or even parents) have helped groups of students who want to participate in the contest doing work outside of school hours.

The greatest barrier for new teams has been getting started with the seemingly foreign technology and programming. However, Phoenix Contest also offers a free, jump-start workshop that is an abbreviated version of the week-long TECHFIT PD to introduce new students and mentors to the technology and nanoNavigator programming.

Conclusion

Exergaming is a fun, appealing activity for children and adults alike. Teaching both groups how they can create their own exergame equips them with essential STEM and CT skills and the knowledge that these skills can be applied to innovating useful products. This formula has helped the TECHFIT team spark enthusiasm for STEM and CT in over 125 teachers and over 2000 students while encouraging fitness to keep minds active.

References

CNET. (2018). *NanoNavigator Download*. (CNET) Retrieved December 18, 2020, from https://download.cnet.com/NanoNavigator/3000-2383_4-75330441.html

Harriger, A., Harriger, B., Parker, L., & Li, W. (2016, September 9). Curriculum and Assessment Design to Study the Development of Motivation and Computational Thinking for Middle School Students across Three Learning Contexts. Arlington, Virginia: National Science Foundation. Retrieved December 18, 2020, from https://www.nsf.gov/awardsearch/showAward?AWD_ID=1640178

Harriger, B., Harriger, A., Flynn, M., & Flynn, S. (2013). Teaching Engineering Concepts to Harness Future Innovators and Technologists. Arlington, VA: National Science Foundation. Retrieved December 18, 2020, from https://www.nsf.gov/awardsearch/showAward?AWD_ID=1312215

International Technology and Engineering Educators Association. (2020). *Standards for technological and engineering literacy: The role of technology and engineering in STEM education.* Reston, VA: International Technology and Engineering Educators Association. Retrieved December 18, 2020, from https://www.iteea.org/STEL.aspx

Justin, R. (2013, July). Why Do I Think Better after I Exercise? *24*(3). Retrieved December 18, 2020, from https://www.scientificamerican.com/article/why-do-you-think-better-after-walk-exercise/

Lifelong Kindergarten Group. (2018). *Scratch - Imagine, Program, Share*. (MIT Media Lab) Retrieved February 28, 2019, from https://scratch.mit.edu/

Phoenix Contact. (2018). *Simple Control: Nanoline*. (Phoenix Contact) Retrieved December 18, 2020, from https://www.phoenixcontact.com/nanoline

Phoenix Contact. (2019). *Nanoline Contest*. (Phoenix Contact) Retrieved December 18, 2020, from http://www.phoenixcontact.com/nanolinecontest

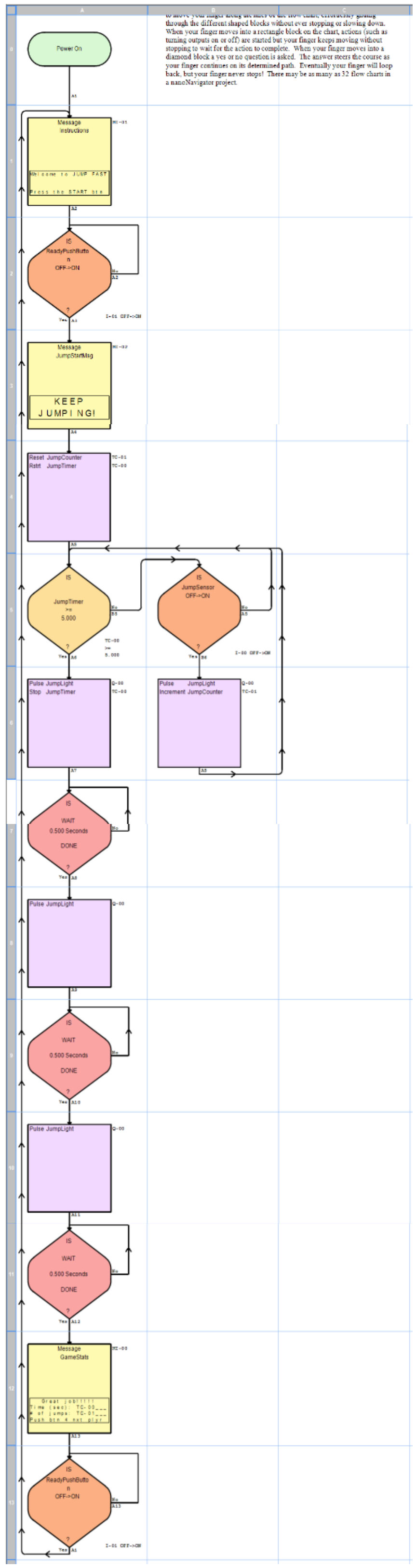
Sween, J., Wallington, S. F., Sheppard, V., Taylor, T., Llanos, A. A., & Adams-Campbell, L. L. (2014, May). The Role of Exergaming in Improving Physical Activity: A Review. *Journal of Physical Activity & Health, 11*(4), 864-870. Retrieved December 18, 2020, from http://dx.doi.org/10.1123/jpah.2011-0425

Authors' Note

TECHFIT was supported by two grants from the National Science Foundation, #DRL-1312215 and #DRL-1640178. Any findings and opinions expressed in this article are those of the authors and do not necessarily reflect the views of the National Science Foundation.

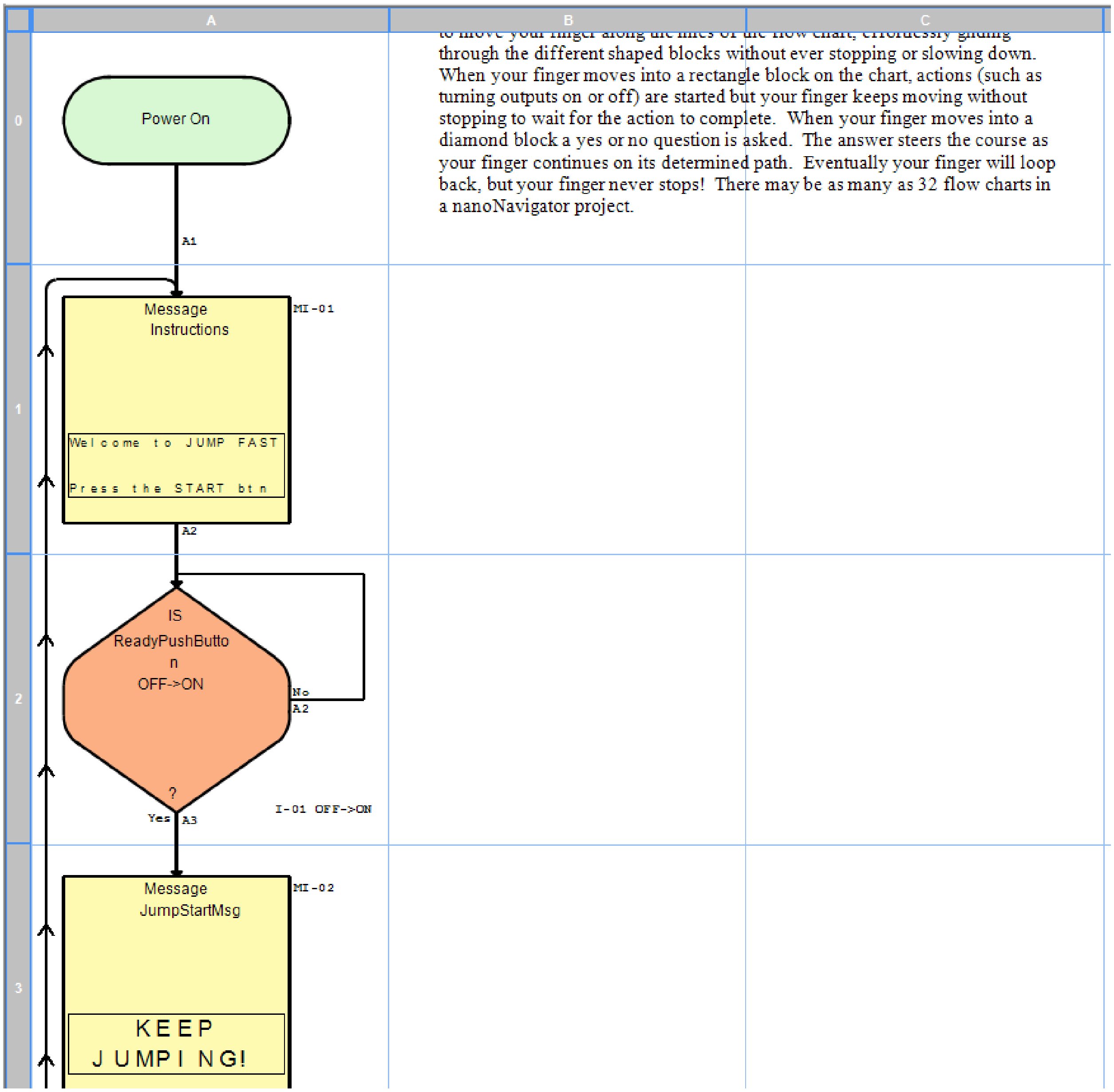
Appendix A

The whole flowchart program appears below to show the overall logical flow.

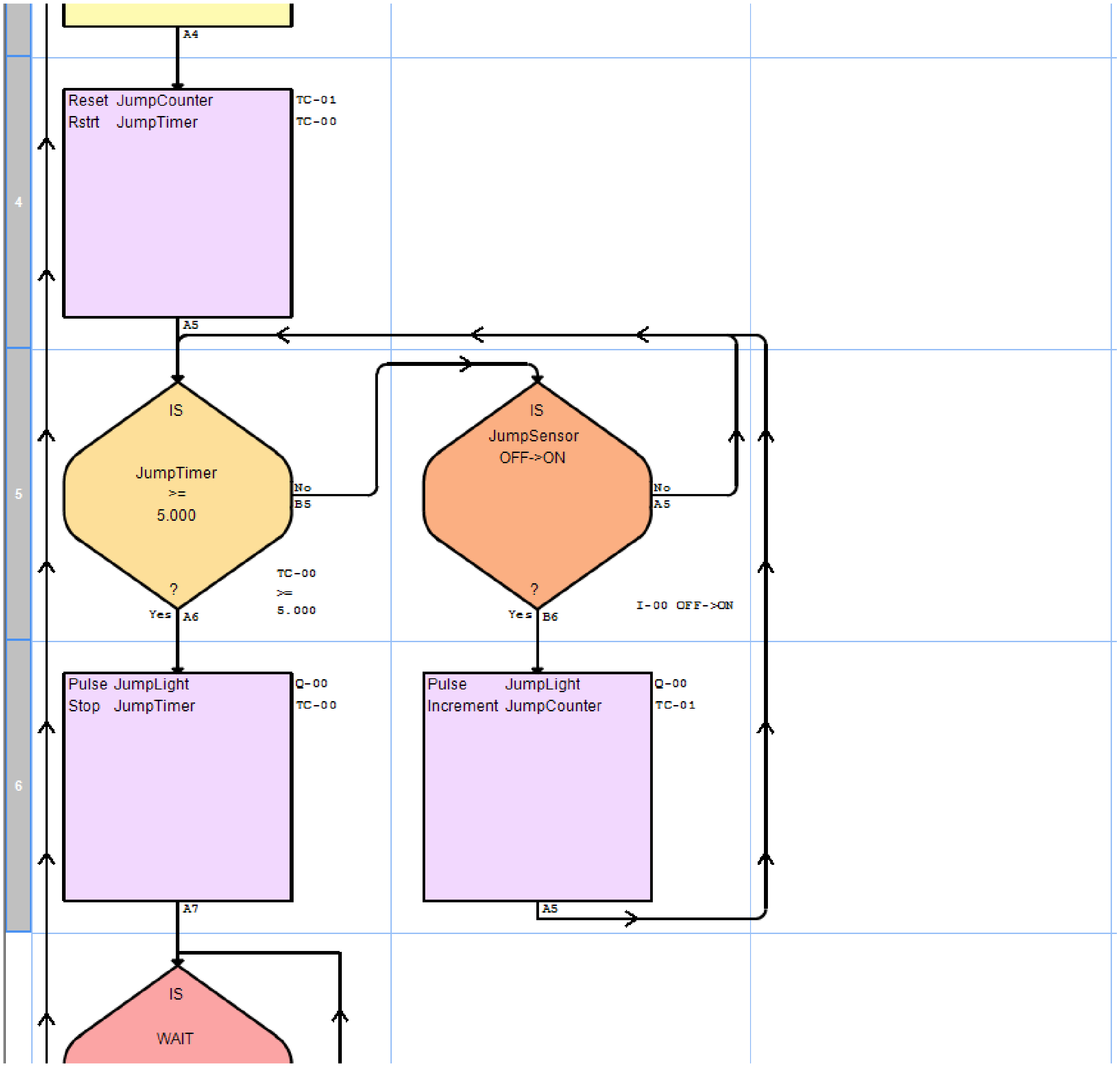


Appendix B

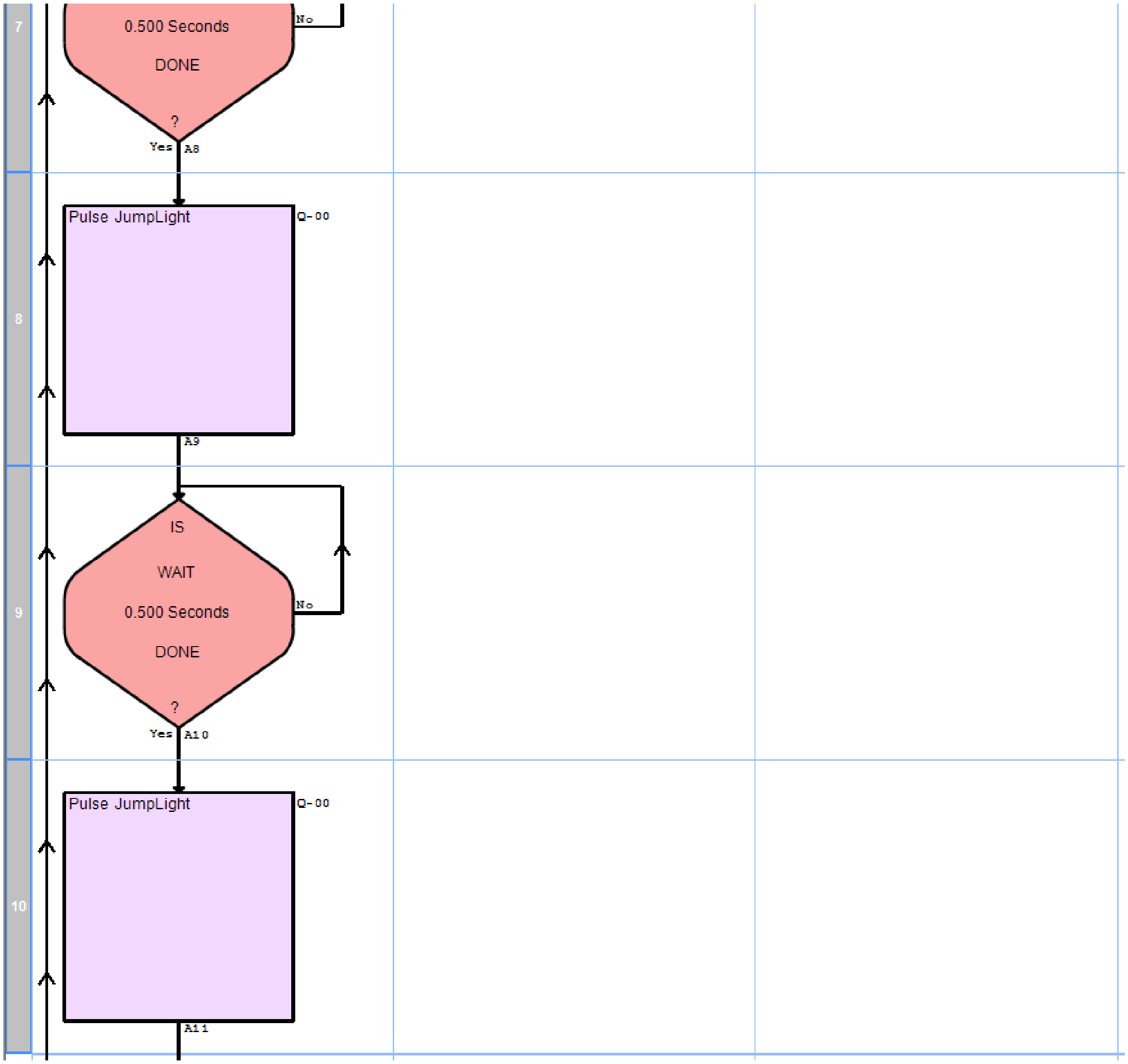
The following four pages show a quarter segment of the flowchart program from top to bottom such that the text inside each block is more easily read.



Appendix B, continued



Appendix B, continued



Appendix B, continued