



# Integrating Circuit Playground with the MUzECS Software



Gustavo Gratacós (University of Puerto Rico, Rio Piedras), Sam Olson (University of Wisconsin, Eau Claire), and Dr. Dennis Brylow (Marquette University)

## Background

MUzECS is an educational platform made to teach computer science to high school students. Among its highlights are:

- Students interact with a board containing a speaker, buttons, and other peripherals
- The use of music as a learning tool
- It is part of a curriculum in schools all over Wisconsin
- The newest version works offline and in Chromebooks

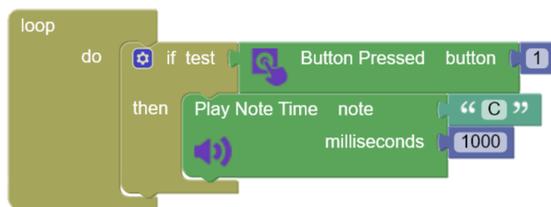


Figure 1: An example program in the MUzECS software.

The MUzECS software does not require students to upload programs to the board. Instead, student-made programs communicate to the board via the Firmata protocol.

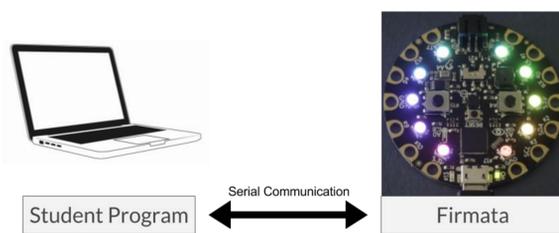


Figure 3: Laptop communicates to board using Firmata.

Recently, a board called Circuit Playground was released.

- It is aimed for computer science education
- It has more peripherals than the MUzECS board
- It is significantly less pricey than the MUzECS board

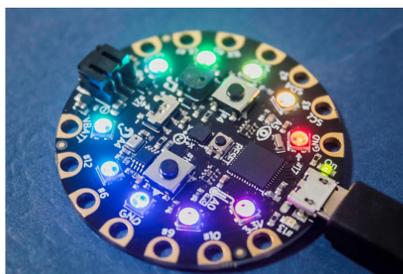


Figure 2: The Circuit Playground board.

The purpose of this research is to combine the Circuit Playground with the MUzECS software. This would provide a very inexpensive board, and the convenience of MUzECS's offline functionality. This would be beneficial for schools with few resources.

## Uploading Firmata

To make the MUzECS software upload Firmata to the Circuit Playground, we needed to:

1. Identify the board by searching for its product ID
2. Reset the board to bootloader mode, this allows us to upload software to the board
3. After the device is restarted, the board needs to be identified again with a different product ID
4. Upload Firmata to the board

## Sensors

Some of the board's sensors can be difficult to abstract in a way that students find them intuitive. For instance, the accelerometer, a sensor that detects the board's acceleration in the X, Y, and Z axis, requires you to:

- Send two bytes requesting an accelerometer reading.
- Wait for a 27-byte response that represents the reading.

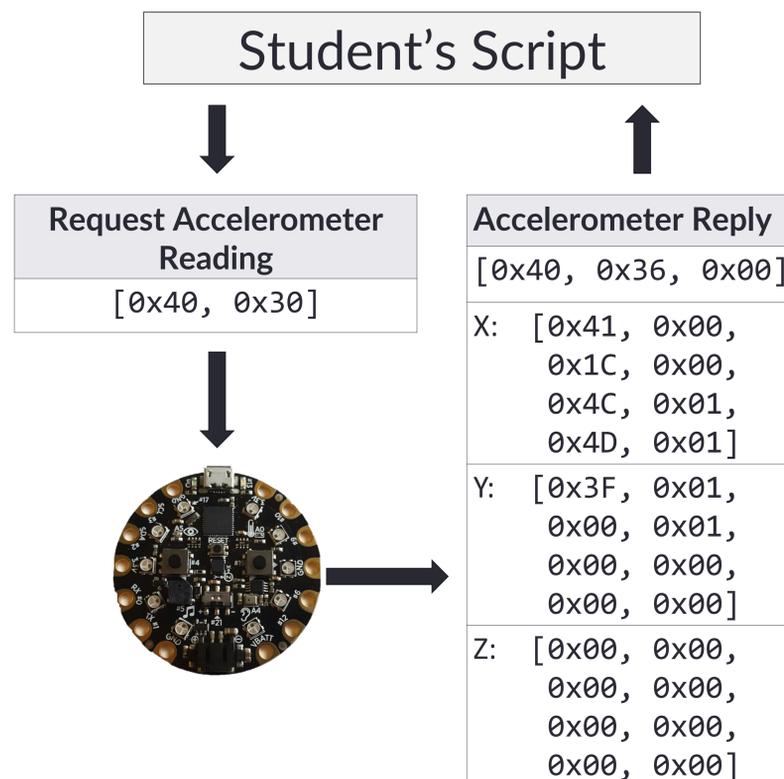


Figure 3: After sending a request to the Circuit Playground, the board sends back a reply with 24 bytes of data.

If a student wanted to use the accelerometer, they'd only need to worry about using a block like this:



## Results

1. We made the Circuit Playground board compatible with the MUzECS software.
2. We designed blocks for students to program each of the Circuit Playground's controls.

**Importance:** This will enable schools with little to no internet connectivity to use the inexpensive Circuit Playground in their curriculum. Maintaining the relevance of MUzECS in computer science schools can be very beneficial. Not only to provide computer science education to schools, but also to have a platform for computer science education research to be carried out.

## Future Work

When programming boards, there are two routes people can take:

- Event driven programming using a high level language
- Procedural programming with C or a similar language

*Does one route prepare students better for computer science?*

*On a similar note, using empirical data, how can the MUzECS software be improved on the basis of learnability?*

Finally, updating MUzECS to have new boards and cutting edge software will maintain its relevance and introduce us to more possible research questions regarding computer science education.

## Acknowledgments

This work was made possible by the National Science Foundation's funding through the NSF Award ACI-1461264. We would like to thank Dennis Brylow for mentoring this project. Furthermore, we would like to acknowledge both Dennis Brylow and Kim Factor for directing and carrying out the Marquette MSCS REU.

## References

- Bajzek, Matthew, Heather Bort, Omokolade Hunpatin, Luke Mivshek, Tyler Much, Casey Ohare, and Dennis Brylow. "MUzECS: Embedded blocks for exploring computer science." *2015 IEEE Blocks and Beyond Workshop (Blocks and Beyond)*(2015): n. pag. Web.
- Bau, David, Jeff Gray, Caitlin Kelleher, Josh Sheldon, and Franklyn Turbak. "Learnable programming." *Communications of the ACM* 60.6 (2017): 72-80. Web.
- Bruce, Kim B. "Controversy on how to teach CS 1: a discussion on the SIGCSE-members mailing list." (2004): n. pag. *ACM Digital Library*. Web. 21 July 2017.
- Hunpatin, Omokolade, Casey O'Hare, Ryan Thomas, and Dennis Brylow. "A Browser-based IDE for the MUzECS Platform." *Proceedings of the 22nd International Conference on Distributed Multimedia Systems*(2016): n. pag. Web.