MUzECS: Extending the Standard Firmata Protocol

Omokolade Hunpatin, Ryan Thomas, and Dr. Dennis Brylow

Marquette University | Mathematics, Statistics, and Computer Science

Motivation

We wanted to improve the MUzECS platform, which was made to be a cost effective supplement to the final module of The Exploring Computer Science (ECS) curriculum. The Exploring Computer Science (ECS) curriculum has been proven to be effective in retaining and teaching underrepresented groups in Computer Science. We have analyzed the problems of the previous MUzECS platform, and changed the MUzECS platform in order to solve them.

Previous Work

The initial version of the MUzECS platform worked well in the pilot schools, and didn't have any significant problems associated with it. It used Ardublock, which is a block based IDE running on Java. Ardublock was able to translate MUzECS blocks to Arduino code, which would then run on an Arduino Uno or Adruino Leanardo.

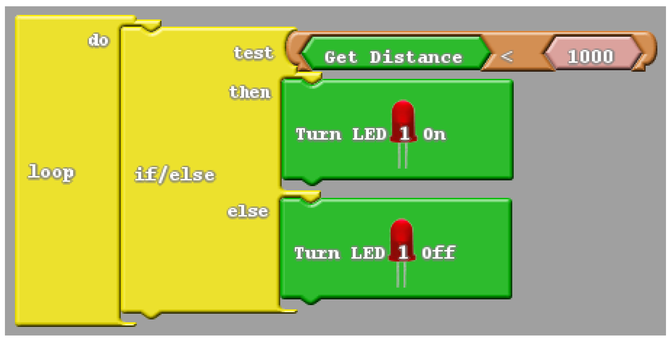

Figure 1: Example MUzECS dialect in Ardublock

Recently, the MUzECS platform was changed in order to accommodate schools in Wisconsin switching to Chromebooks. This is because Chromebooks can only run Chrome, and therefore Ardublock wouldn't work on them. MUzECS switched to a cloud based solution where the blocks were translated to Arduino code and then the Arduino code was compiled on our own servers and uploaded to the client's Arduino.

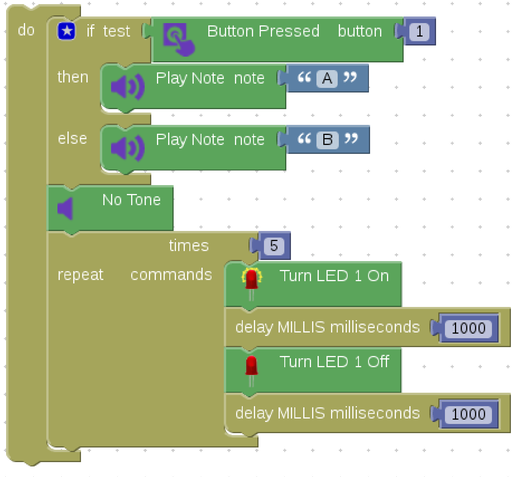

Figure 2: Example of program in MUzECS dialect in Chrome based MUzECS platform

Figure 3: Arduino Leonardo with MUzECS shield

Acknowledgements

Some students on the MUzECS team were supported by the National Science Foundation, grants CNS-1339392, and ACI 1461264. We would like to thank our pilot teachers and students. We owe special thanks to Robert Juranitch and Gail Chapman. MUzECS and MUzECS on Chrome are created from Ardublock and Blocky, respectfully.

Removing Server Side Compilation

The Chrome based version of MUzECS relied heavily on compiling arduino code on the server side, so that Chromebooks could use our platform. The server had load balancing problems, and later couldn't be relied on to handle the scale of the increasing demand for the MUzECS platform. Through the use of Johnny Five and Standard Firmata we were able to remove server side compilation of the translated Arduino code. The blocks are translated to Javascript, which communicates with the Standard Firmata running on the Arduino following the Standard Firmata protocol.

We had to extend the Standard Firmata in order to add support for the custom distance senors on the MUzECS shield. The Johnny Five framework and the Standard Firmata didn't support communication with our custom distance sensor. We made a Distance Request SYSEX message, and extended the Standard Firmata so it would activate the distance sensor in response. The Standard Firmata would then send the integer value of the distance back to the client in 7 bit chunks.

```
<SYSEX_MESSAGE> ::= <START_SYSEX> <SYSEX_COMMAND> <DATA>* <END_SYSEX>

<START_SYSEX> ::= 0xF0

<SYSEX_COMMAND> ::= 0x00-0x7F

<DATA> ::= 0x00-0x7F

<END_SYSEX> ::= 0xF7
```

Figure 4: A BNF grammar of a typical SYSEX message

References

- Bajzek, Matthew, et al. "MUzECS: Embedded blocks for exploring computer science." Blocks and Beyond Workshop (Blocks and Beyond), 2015 IEEE. IEEE, 2015.
- Brylow, Dennis. "MUzECS: Block-based Arduino Programming for Exploring Computer Science." Proceedings of the 47th ACM Technical Symposium on Computing Science Education. ACM, 2016.