



A Custom Block-Based Language For The Modified ECS Curriculum

Matthew Bajzek, John Casey, Luke Mivshek, Tyler Much, Dr. Dennis Brylow



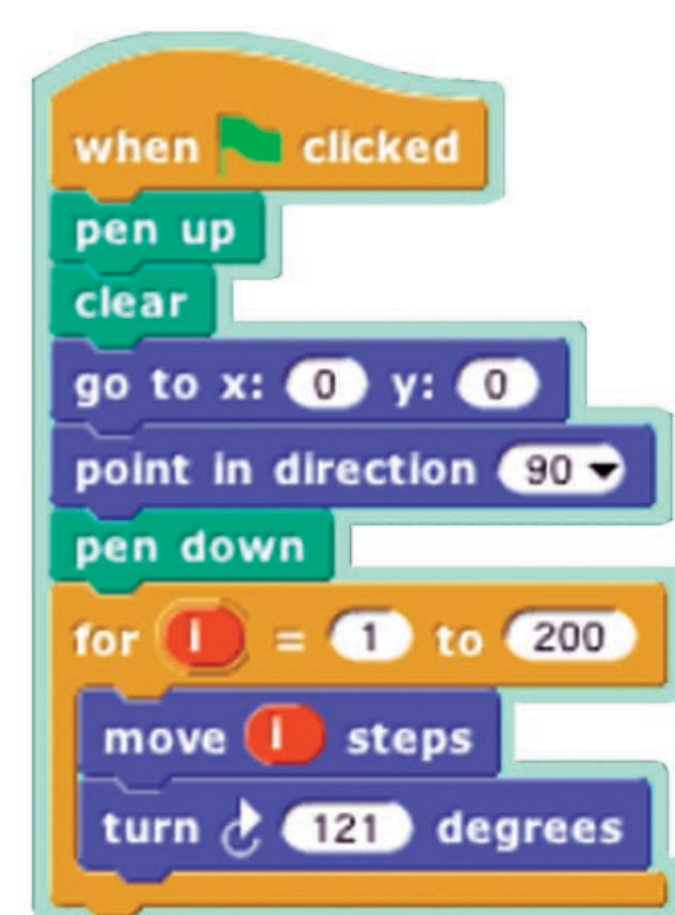
Motivation

The ECS (Exploring Computer Science) curriculum aims to spread quality computer science education to underrepresented portions of the high school population. Our team hopes to further this goal by creating an alternative to the course's final module.

Any tool presented to ECS students must be flexible enough to foster learning without overwhelming or intimidating them. To achieve this, we have created a custom block-based language.

Scratch

Scratch, a popular block-based programming language, is the foundation for ECS coding.



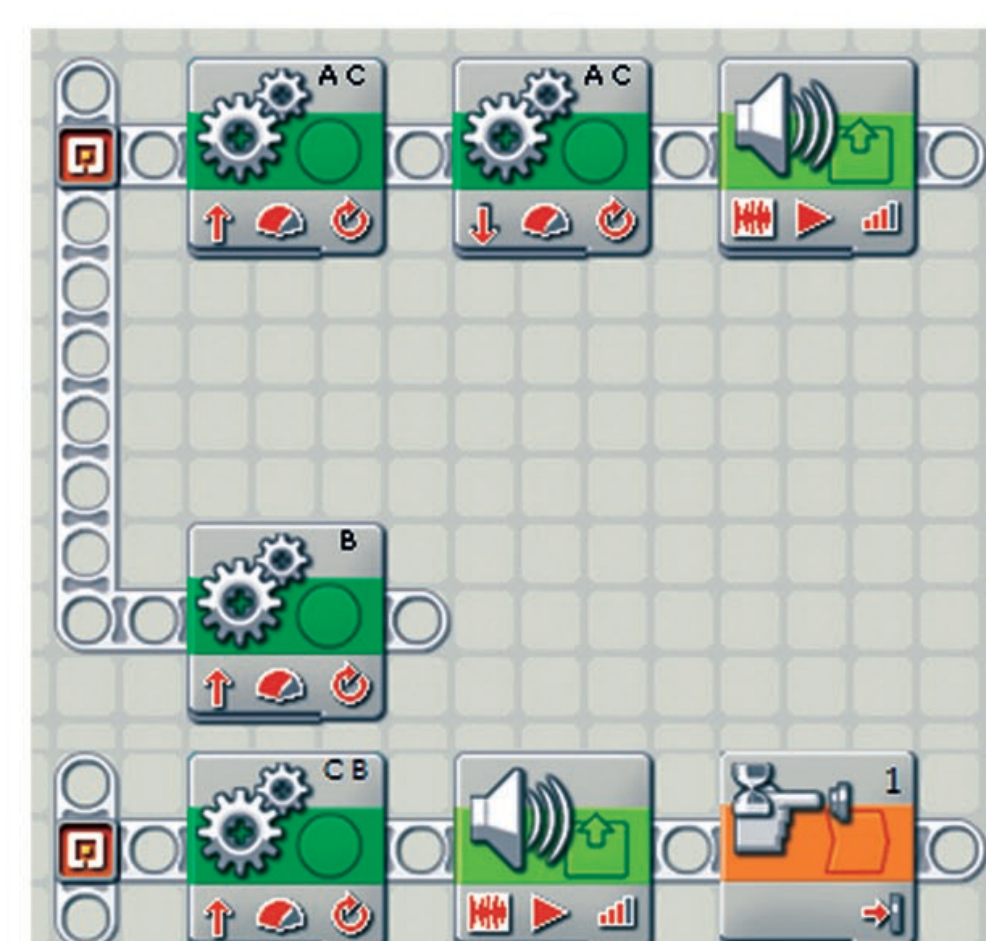
Students are taught to place code "blocks" via a simple drag-and-drop interface in order to form programs.

All Scratch code is written, compiled, and executed in the same window, generating basic animated output.

Mindstorms

Currently, the sixth module of ECS utilizes LEGO Mindstorms hardware and software to introduce students to robotics as an application of computing.

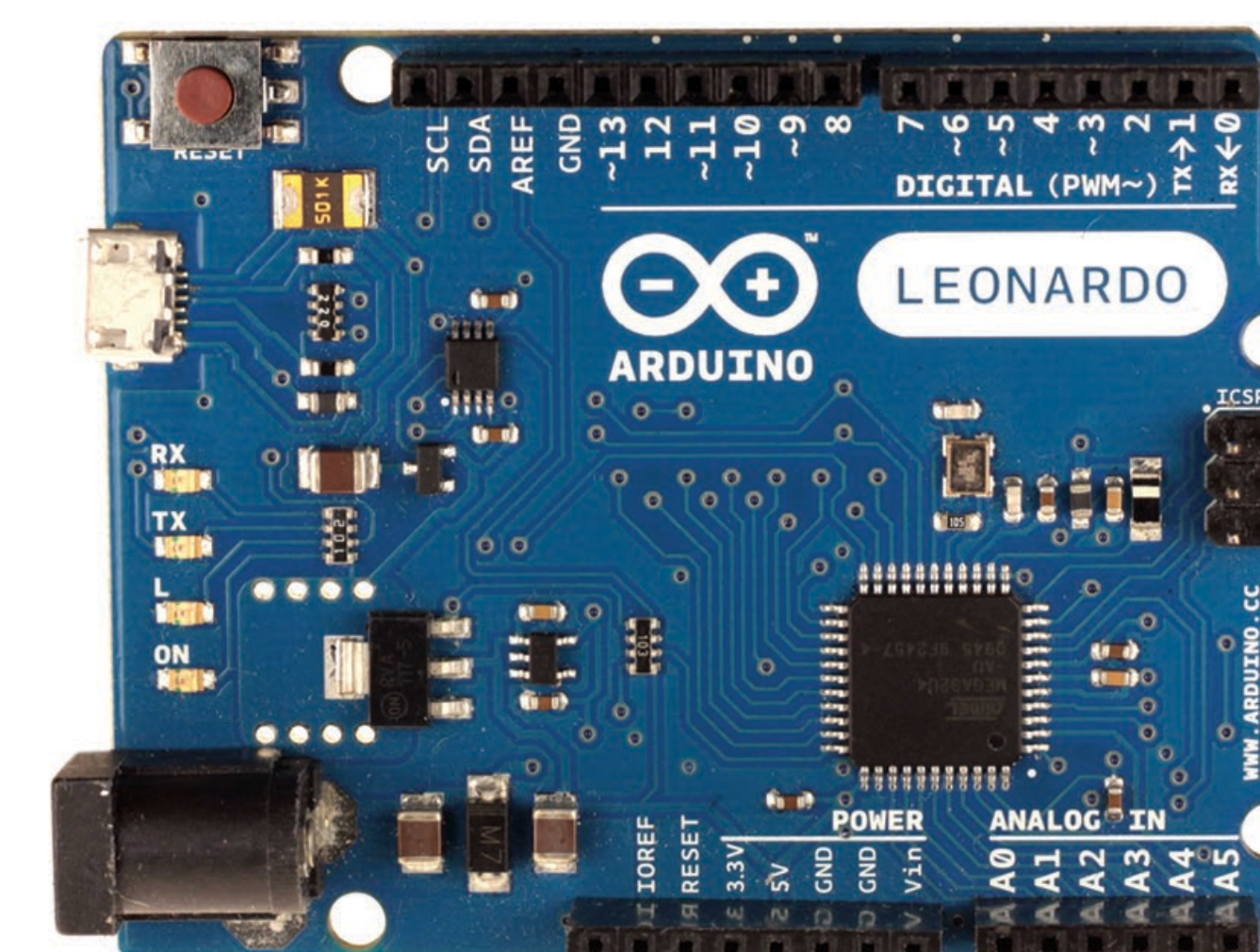
Much like Scratch animations, LMS robots are driven by code created using a custom toolkit, simplified to reduce complexity for new programmers.



Proposed Change

While ECS's current Mindstorms-based module 6 has been proven effective in the classroom, it is also prohibitively expensive for a significant number of schools and districts. In order to promote increased computer science learning opportunities for the students in these schools, a modified module 6 has been designed to remove cost as a barrier to entry.

At the center of the proposed change is our move to a new hardware platform, the Arduino Leonardo.



At only a fraction of the cost (under \$25 MSRP), the Leonardo can be afforded by far more schools than the traditional \$350 Mindstorms kit.

In order to leverage this new hardware, new software is necessary. This software must satisfy numerous criteria to make the change viable:

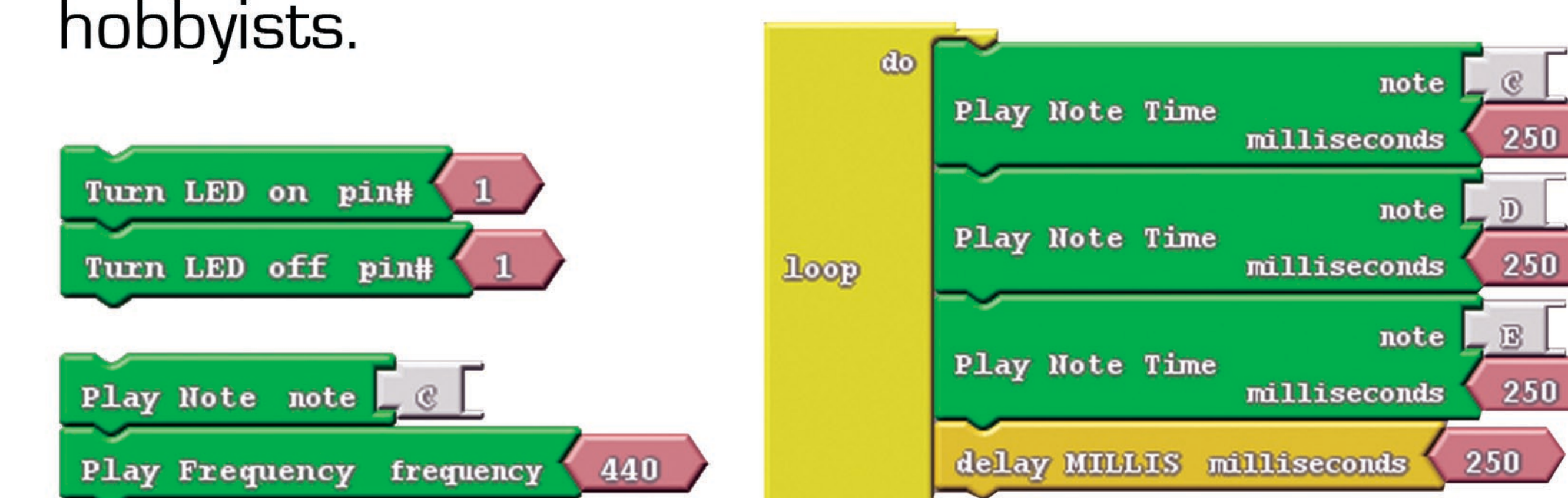
- Allow students to control Leonardo GPIO pins
- Consistent with student Scratch experience
- Simple to learn
- Easy to deploy on hardware
- Robust enough to allow experimentation
- Easily modifiable for future development
- Adaptable to a variety of projects

References

- <http://www.exploringcs.org/>
- <http://blog.ardublock.com/>
- <http://scratch.mit.edu/>
- <http://arduino.cc/en/Main/arduinoBoardLeonardo>
- <http://www.lego.com/en-us/mindstorms/>

Our Solution

In order to control the Arduino Leonardo's pins, we have created a modified version of Ardublocks, a popular program used by Arduino designers and hobbyists.



Our modifications allow control of peripherals pre-selected for the curriculum's projects, including LEDs, speakers, sensors, and photoresistors.

This custom block set has several distinct advantages beyond hardware cost reduction:

- Code is written almost exactly as in Scratch
- Programs are easy to read and debug
- Peripheral control is clear and simple
- Code is deployed via USB with Arduino's program
- More general, advanced blocks are included to allow advanced students more flexibility

Our block set is designed to be simple and easy for students to pick up and use.

We look forward to seeing what they create with it.

