Embedded Development
iTEC 2017 Winter Edition

The final scope of this challenge is to have a working handheld Tetris game that adopts as many official Tetris rules as possible, built using an Arduino board, a color LCD display, push buttons and anything else you think would add to the usability and fun of playing Tetris.

For a list of tetris specifications and rules, please use online references, e.g.: https://harddrop.com/wiki/Tetris_Guideline & ttp://tetris.wikia.com/wiki/Tetris_Guideline.

The Hardware Kit that's available from us consists of the following items:
- Arduino Mega2560 (or Arduino Uno R3)
- LCD display: 128x128 pixels, SPI interface to ST7735 controller
- Level shifters (either x2 or x4 channels) for 8 channels
- MPU6050
- Push buttons
- Jumpers
- Breadboards
- Breadboard power supplies
- Protoboards
- Headers
- Resistors

The parts listed above should be sufficient to get a fully functional handheld tetris game, but you are free to add, modify, remove any components, or use any parts that you have brought yourselves, if you believe this will help you get better results.

The MPU6050 is a wild card that we threw in for this challenge, as a secondary form of input for the gameplay, be creative.

Milestones and points:

- Manually interfacing with the LCD to get color graphics: **10 points**
- Getting the LCD to display a Tetris-like game scene (Tetris grid, Tetris blocks, sidebar points, etc): **10 points**
- Adding the fixed periodical Tetris block spawning: **10 points**
- Adding push button block rotation and drop: **10 points**
- Adding correct block placement, line deletion and screen update, score system, game over and restart functionality: **20 points**
- Soldering the components to a protoboard for a "handheld" Tetris game (e.g. as an Arduino Uno form factor shield): **10 pointsz**
- Sticking to the Tetris official board rules: **15 points**
- Creativity: **15 points**

Points can be given partially based on your progress with a certain milestone.

Your progress will be recorded during the competition, and points will be partially awarded at these moments as well.

We prefer seeing a partial game that was manually implemented than a fully functional game that was made by another.
Because of this we might withhold points or not award points on a certain milestone based on your use of libraries and source code that was taken from the Internet and not written by you.

Use libraries to quickly validate that your hardware connections are correctly made, but try to make your own libraries, drivers and APIs afterwards based on your use case, as you will receive more points this way.
If you decide to use external source code to get more done, make sure you understand it first, and can prove this to the jury if questioned.

Good luck and have fun!