Embedded Assignment: Tic-Tac-Toe

Due: 11:59PM October 19, 2016

Contents

1 Overview 1
2 Parts, Consumables and Hints 2
3 IoT Lab 2
4 Programming 2
5 Collaboration Policy 3
6 Documentation 3
7 Submission 3
8 Evaluation 3
9 Testing 3

1 Overview

Many of you have chosen to work with the Arduino platform for the courses final project, so this assignment will help you get familiarized with it. In this assignment, you will implement a Tic-Tac-Toe game. The project is divided in two parts: board design and software implementation. You will build your own tic-tac-toe game board using the parts listed in Section 2. On your board you should demonstrate the current game state and allow the human player to set her moves. All game control should be relayed to the Arduino platform. The Arduino program should start a game between a human player and the computer. You play against the computer until one of the players win or there is a draw or the board is reset. Your project should have some way of showing a win / loss / tie. In any of these cases, the game state should be cleared and a new game should be started.
2 Parts, Consumables and Hints

Assuming you have acquired your own Arduino platform and breadboard, each student/group is entitled to the parts listed below. To receive your material, please go to Prof. Reiss’s office (CIT room 403) and sign up for it. You can keep them for future use (e.g., your final project).

1. Electrical wire
2. Resistors
3. Button switches
4. RGB LEDs
5. Diodes

Note: neatness of your circuit will be part of your grade.

In designing your circuit and software, keep in mind potential use issues. For example, what if multiple buttons are pressed at once?

3 IoT Lab

You will have access to the IoT lab as a space to work on your project, located on the 8th floor of the sciences library. There will also be help sessions for you to optionally participate in. We have more information about these sessions once we have the times. If you wish, you can solder your assignment after breadboarding it. Soldering instructions will be provided during IoT lab help sessions. The solder-ready circuit boards will be provided by the course.

4 Programming

You should implement your tic-tac-toe logic using the Arduino IDE. You are supposed to handle I/O and game logic using the Arduino platform. You are free to implement your own game strategy for the CPU player, as long as it is capable of winning or drawing a game (i.e., it should never lose.) Details such as the initial player (which should be changeable), color assignment, winning detection, board reset, strategy selection (if there is more than one) are decisions that should be taken and documented by you. Clarification: you can use any method you deem appropriate for deciding initial player such as a button or an boolean flag in code. Most importantly, the CPU player should respond in real time, while the Arduino platform takes care of the game state. Everything Arduino-related can be found at [http://www.arduino.cc](http://www.arduino.cc) If you are not familiar with breadboards and wiring, an easy way to start is via a simulation environment. Open Electronics Lab ([http://123d.circuits.io/lab](http://123d.circuits.io/lab)) offers a free software solution you can use to prototype your ideas.
5 Collaboration Policy

Students can work individually or in pairs, however all the work must be the creation of you and your partner and not use an existing implementation for tic-tac-toe. If working in pairs, each member should have a deep understanding of the whole project, not just their part. Please use the mailing list or email the TAs if you have any questions.

6 Documentation

You should write a README file, documenting any bugs you have in your code, all the design decisions you took, and anything else you think the TAs should know about your project. Also be sure to document the instructions used to compile, install (if necessary) and run your program.

7 Submission

You should hand in your assignment by running /course/cs160/bin/cs160_handin ttt from your project directory. Please copy all the source code, documentation and the Makefile (if it exists) into your project directory, cd there, and run the handin script. Do NOT submit object or executable files and make sure your program compiles cleanly.

8 Evaluation

We will have interactive grading for this assignment. Your grading will be based not just on functionality, but also on the design, efficiency, commenting, and documentation. The interactive grading will take place two to fours days after the project is due; location and hour will be announced later. If you do not show to your interactive grading and you do not have a valid excuse, you will be penalized. We will post information necessary to schedule an interactive grading session on the mailing list. Be prepared to answer questions as to your design (both HW and SW) and implementation.

9 Testing

To make sure you will not have unpleasant surprises during the interactive grading session, make sure you have extensively tested your board and software before submitting your code. Imagine you are releasing your tic-tac-toe game to the market (disregard aesthetics and usability factors for now). How people would interact with it? How fast can you press the buttons? Are there any invalid inputs?