**SOFTWARE VERSION DESCRIPTION DOCUMENT**

House Rules

University of Notre Dame

Electrical Engineering

Senior Design 2011

**TABLE OF CONTENTS**

1. Project Software Needs
2. Division of Labor
3. Detailed Functional/Implementation Description
	1. Player Interface Unit
	2. Embedded Intelligence
4. Macroscopic Analysis
5. **Project Software Needs**

The need for the automated beverage pong table to have precise and user-interactive software arises from a two-fold demand. First off, it was decided fairly early in the design phase of the project that “made-cup” sensing technology would not be incorporated into our solution. This was primarily due to the fact that the hardware and game-piece specialization necessary to accomplish this task was much more difficult than our time and money constraints allowed. Thus, a different solution to game-tracking was necessary. The team decided that the best way to do this was by incorporating an external intelligence device that allowed the users to interact with the game and manually keep track of cups made. This external hardware/software solution is now referred to as the Player Interface Unit (PIU).

The second prong of the demand for significant investment in software development was the need to create a communication and translation link between the PIU and the stepper motors and their drivers. The solution developed to tackle this demand was on-board microprocessor firmware running on Arduino Deumilanove development boards. This hardware/software combination would handle the decoding of the movement commands sent from the PIU as well as the distribution of motor commands to the motors of the different axes.

Both of the software aspects of the two solutions will be discussed in detail herein.

1. **Division of Labor**

One of the toughest things regarding the overall completion of the team’s project was determining how to structure the necessary software needs. After a lot of design work and upfront thought, the software was divided up as described below.

The team decided to divide the software into two portions. Because of the processing requirements, it was decided to develop most of the software and carry out most of the operations on the PIU, an external computer. This PIU software is written in C# and will be packaged into a graphical-based application that will run on any machine. This C# program running on the PIU will perform the following tasks:

1. Display to the user the current cup arrangement and game statistics.
2. Allow the user indicate a made cup.
3. Create instruction set objects that have X and Y coordinate waypoints and magnetic latching information.
4. Pass correctly formatted instruction sets to embedded intelligence via a Virtual COM Port USB interface.

Because of the specialized interface board the team will use between the Arduino and stepper motors, the stepper motor control software will be implemented on the Arduino development board. This software will be written in a proprietary variant of C whose documentation can be seen on arduino.cc. This on-board firmware will perform the following tasks:

1. Receive instructions from the PIU via USB Interface.
2. Send necessary digital pulses to control stepper motors.
3. Send necessary feedback back to PIU.