2007-2008

Self-Scoring Cornhole

Senior Design Project

Our team (affectionately known as “America!”) is designing a new electronic scoring version of the American tailgating game, cornhole. We will apply our engineering education and innovative ideas to produce a cornhole set that utilizes some of today’s advanced technologies.

Paul Carlson
Kathryn Jannazo
Christina McCool
Chris Devitt
Tom Martin
Dan Wolff

4 October 2007

Table of Contents

Introduction

Cornhole is a favorite American pastime, in which players take turns pitching small bags filled with corn at a raised platform with a hole. This game can be played with as few as two people or as a tournament with many friends. Points are scored by either keeping a bag on the board or making a bag in the hole, a “Cornhole”.

The basic rules of cornhole allow individuals of all ages to enjoy this game. Two boxes will be set 30 feet apart from the center of hole to hole. Two teams are formed with either one or two players on each team. For singles play (one per team), opponents start and play from the same end. For doubles play (two per team), teammates should line up next to opposite boxes and on opposite sides. All eight bags begin on one end. The foul line is made by the front of the boxes and all throws must be made from behind this line. If a player breaks the foul line or throws out of turn, that teammate will lose a turn. The teams alternate throwing until all the bags are tossed. The team that scores the most points in a round starts throwing for the next round. If the teams are tied, the last team to win a round throws first. Any bag that touches the ground during a throw should be removed from the board and not counted. A team will score one point for any bag that is tossed on the board and three points for any bag in the hole. For each throwing round, the team with the most points wins the round and subtracts the other team’s points from theirs. This difference is added to the team’s score that won the round. The first team to 21 wins.

Cornhole is a game which most likely originated in Cincinnati, Ohio several decades ago. However, while it has been around since the late 20th century, it has only become popular in the last 8 years. Cornhole is similar to many other lawn games, but has a unique advantage as it can be transported to various locations. It is a fun competitive game which allows people of all ages participate in a competition to make the most “Cornholes”.

Problem Description

Our goal is to improve the existing game of cornhole by digitizing the scoring. Our game will have the intelligence to differentiate between teams, determine the score for the round, add it to the existing score, display it on an electronic scoreboard, and communicate it to the opposite scoreboard with wireless technology.

Proposed Solution

At a high level, we have several functional areas that need to be designed. Below is a short outline describing these…

- Construction
  - Build Set and Bean bags
  - Integrate 4 Force Sensors
- Sensing
  - Get input from Weight Scales
  - Programming different possible weight combos
- Scoring
  - Get input from sensors
  - Programming
- Displaying
  - Output to scoreboard
  - Circuit design for display
- Communication
  - Programming
  - RF link
- User Interface
  - RF within boards
  - Circuitry for keypads
  - Possible addition: Remote control user input
- Power
  - Power consumption
  - Drains: Microprocessor, Scoreboard/Display, Weight Sensor, RF

To accomplish this, some of the supplies we will need include…

- 2 playing boards
- Beanbags with slightly offset weights
- 4 Weight scales
- Keypad
    - Reset Button, on, off, score change
    - Keyboard to input names
    - Possible Remote Control
- Scoreboard with LEDs
- 2 Microprocessors
- RF capability to calculate the score

For the construction of the cornhole set we will be using the regulation size cornhole dimensions, a scoreboard that is on the cornhole board and plexiglas that covers the scoreboard. Each board must contain weight sensors that measure the amount of weight that is on the board at the end of each turn due to the bags. The sensing of the bags' weight is the means to keep score for each team.

The microcontroller is involved in most of the logical functions in this project. The microcontroller will use the output of the weight sensors to determine which team the bag belongs to. After the number and location of bags from each team has been determined, the microcontroller will calculate the new score by using a program to determine the score based on the locations of these bags. The microcontroller will then output this score to the scoreboard which will display the updated score. In addition, the microcontroller will use the RF link to transmit the updated score to the other board so both boards show the same score. We will need an RF transceiver in each of the boards to handle these communications and synchronizations.

Additionally, we need to use some form of input to allow for the user(s) to interact with the game. The user inputs information that affects certain parts of the system.

With all of these features, we need a way to provide power to the entire system (Microprocessor, Scoreboard/Display, Weight Sensor, RF). Using either a laptop battery, lithium ion battery, or some form of rechargeable power source are some options.

Demonstrated Features

The Scoreboard will keep track of both teams' scores during each round and, depending on our budget allocations, will be able to display team names above the respective scores.

Each board will need two weight sensors, one sensor measuring the entire weight of the bags that are on the board and one sensor measuring if any bags made it through the hole. Each team's bags will have a different weight so that there will be a different amount of weight on the board for each combination of bags that may land on the board. After sensing the total weight of the bags on the board, the sensor must send a signal, depending on what type of sensor is used, to the microprocessor that contains the data of how much weight is on the board. This can be done by voltage, current, frequency, etc.

In addition to taking the output from the weight sensors, the microcontroller will use this weight to determine the score, output this score to the display, output this score to the RF transmitter so it can be sent to the other board. This will be accomplished by programming the microcontroller to input and output certain signals.

The RF link will be designed to handle several key tasks. First, it will be used to synchronize the clocks of the microprocessors. One microprocessor can start and send a trigger to the other, synching the clocks. Secondly, it will be used to communicate the end of turn on one side and signal for the other side to begin. Finally, the RF link will be used to ensure the updated score from each round gets transmitted to the opposite display as well – keeping the scoreboards in sync.

The user must be able to turn the game on, start the game, restart the game, and be able to turn the game off. Since we're using weight sensors, the user will need input information to let the game know when all of the bags have been thrown by both teams. This input will let the game know that the round is over, and that it should calculate the new score. These functions above can be described as the "basic" functions that need to be present. Additional features besides these might be an input for team names, which are displayed, and the ability to let the user adjust the scores if there is a miss-score.

We will power all of the components using the battery. It will also be rechargeable for extended life of the set.

Available Technologies

Construction

The board will be made of regulation size ½ inch plywood and the Plexiglas will be ¼ inch thick and clear. For the scoreboard, we need a display that is visible outside from over 30 feet away. The scoreboard can either be premade or built by us. Currently we are planning on building it ourselves to help keep costs down.

Some of the available technologies are:

- **LED 7 Segment Display:** Made up of Bright Red LED's, comes with drive transistors and step-down resistors, have connectors for microcontroller input and is 180mm in length and 230mm in width.
- **Double 7 Segment Display:** This includes 4" digits, pre-mounted connectors, voltage regulators, capacitors and mounting sockets for a microcontroller.
- **Build our own LED Matrix:** Another possibility is for us to build our own LED circuit which we could tailor to the specific size, voltage and power consumption that we desire for the project.
- **Numerical LCD Display:** An LCD Digital Panel Meter would be Voltage Powered, 3 1/2 digit LCD Digital Panel Meters with adjustable Voltage Input

Weight Sensors

To determine the weight of the board after the bags, there must be a sensor that can be situated in the construction of the boards that will be able to detect all of the weight of the bags in a precise manner. An available technology to accomplish this is the use of load cells. Load cells are devices which produce an output signal proportional to the applied weight or force. There are several different kinds of load cell technologies that have different amount capacities, uses, and prices. A practical load cell for our project would have a capacity of anywhere between 10 kg to 15 kg depending on our construction of the boards and where we put the load cell in the board. A possible route to go would be using beam load cells that range from $90 - $150 based on the capacity and quality of the load cell. The beam load cells require a DC voltage ranging from 5 – 10 V and have 4 wires for input voltage and output signals.

Another available technology could involve using a simpler weight scale in combination with a photoelectric sensor that sensed the change of numbers on the display of the scale. The sensor would then send a signal in the same manner as the load cell.

Microcontroller

A microcontroller is a type of microprocessor which is focused on self-sufficiency and cost-effectiveness. The microcontroller will enable the cornhole set to have information input to it and output from it. The microcontroller we will be using is a 18(L)F4620. This microcontroller has various power managed modes and a flexible oscillator clock structure. In addition it has peripherals which include a high current sink/source, three programmable external interrupts, four input change interrupts, Capture/Compare/PWM modules, a Enhanced Capture/Compare/PWM module, a Master Synchronous Serial Port module, an Enhanced Addressable USART module, an Analog-to-Digital Converter, Dual analog comparators with input multiplexing, and Programmable 16-level High/Low-Voltage Detection module.

RF link

For this purpose, we will need two simple, cost-effective, and low-power RF transceivers. It will need to have a range of ~30 feet and should not interfere with existing RF signals (radio, cell-phone, etc). It does not require significant bandwidth or data rates.

Some technologies that are available include…

- **ZigBee** is based on the IEEE 802.15.4 standard and is targeted at RF applications that require a low data rate and long battery life.
- **Wibree** is designed for ultra low power consumption within a short range (10 meters / 30 ft) based around low-cost transceiver microchips in each device.
- **Bluetooth** is an industrial specification for wireless personal area networks.
- **EnOcean** uses self-powered modules (transmitters, receivers, transceivers, energy converter)

User Interface

There are many different ways to create a user interface. These include keyboards, keypads, mouse, switches, and touch screen technology. These input devices can come from a variety of manufacturers or can be made through the combination of devices. Each technology can possibly include all of the features listed above or just some of them. Multiple combinations of these technologies can also be used to get the needed features. Listed below are the combinations of available technologies with some of their features.

- Keyboards:
    a. Full: Computer keyboard style including letters, numbers, etc. This type of technology would allow for a complex interaction between the user and the game.

   b. Limited: Certain elements of a computer keyboard. This would limit the interaction between the user and the game, but still allow for most, if not all, of the features to be present.
   c. Custom: Using a pre-made-programmable keyboard or team made (almost from scratch) keyboard.
- Keypads:
   a. Number pad: This is the number pad on a computer keyboard with the numbers and some other programmable keys for other basic functions.
   b. Phone-like pad: This is like on a cell phone where there are numbers and with those numbers having specific letters (3 of them) that are assigned to that number. There would also be a few programmable keys for the basic operations of on, off, reset, and start.
   c. Custom: Using a pre-made-programmable keypad or team made (almost from scratch) keypad.
- Mouse: The display would need to be capable of displaying enough information to allow for this option.
   a. Roller: The user rolls the ball certain in directions moving the cursor on the display, and the user clicks a button to select the features that the cursor falls on.
   b. Computer mouse: The user moves the mouse in certain directions moving the cursor on the display, and the user clicks a button to select the features that the cursor falls on.
- Switches: Limited in function based on the number of switches used.
   a. Push: These would allow for the basic functions of turning the game on/off, restarting the game, and starting the game.
- Touch Screen Technology:
   a. Cost is the limiting factor in using this technology. However, it could be like the I-Phone or Ipod-Touch interface allowing the user to scroll through options and then select the function.

The available technology will be limited by cost, interface, their input/output types, power consumption, and availability. We would like to use a keypad like in phones. This would allow for all of the basic functions described above and possibly some of the additional features. However, switches may replace some of the basic functions.

Power

The following are examples of power supply technologies that we could use to power and recharge our power supply for our system:

- NIC9205 - 9.6V 160mAh NIMH - Nimh Cell Or Pack Of Cells

- 10 Slots Smart Bay Charger for 9V NiMH/ NiCd Battery (IC-109) CH-IC-109V
- Two Li-Ion 9V 400mAh Rechargeable Battery+ 0.3A Smart Charger (Plastic end) LI-9V400+CH
- LAP3005 - 10.8V 4500mAh LION - Laptop Battery
- Two-Cell Three-Cell & Four Cell Lithium-Ion or Lithium-Polymer Battery Protect
- Rechargeable Power Supply

Engineering Content

Figure 1: Overall system in terms of major functional blocks.

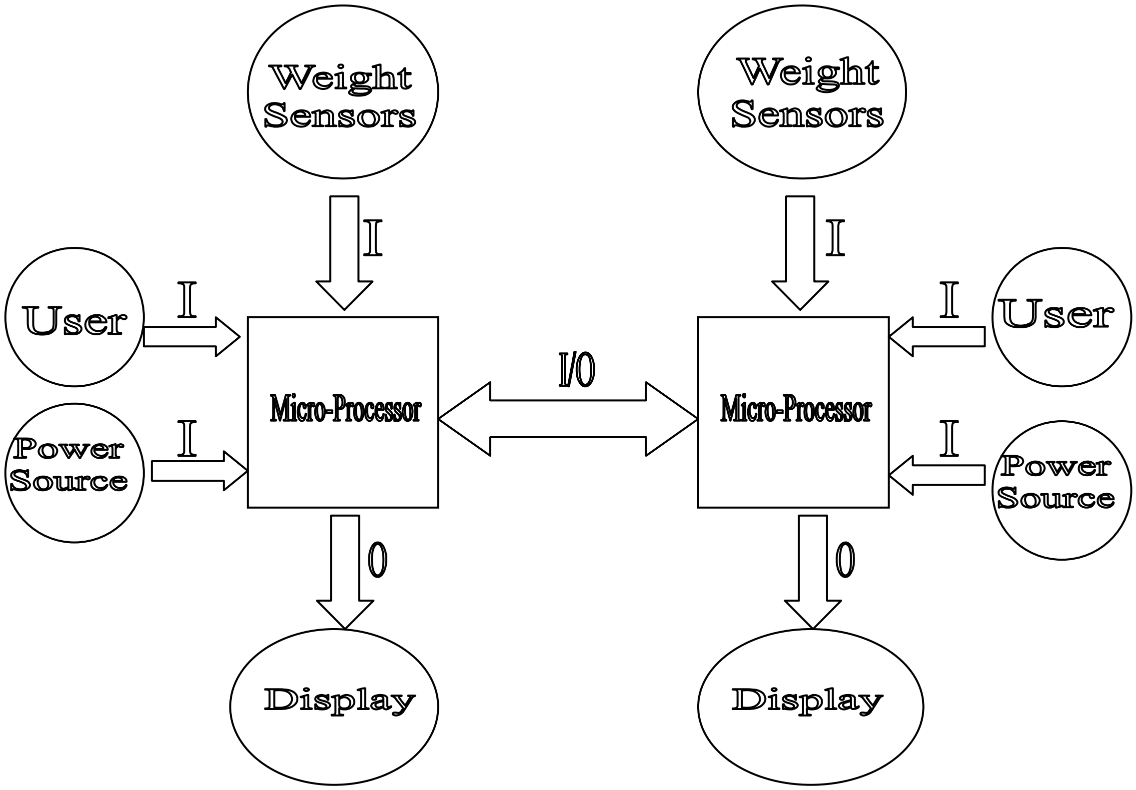

Figure 1 shows the overall system in terms of the major functional blocks. There are two boards in a Corn-hole game, two teams, and eight bags. Therefore, this system must have many inputs and outputs so that all of the features of the game can be maintained. We will have to engineer how to build the Corn-hole set with all of the components in them so that it still functions, is durable, is not overly heavy, and is as cost effective as possible. Designing, building, and testing the whole device will be done in parts because of the complexity of the device. Once all of the major functional blocks have been designed, built, and tested, we will be able to test the system as a whole. Described below is the engineering content. The content is divided into the major functional blocks, which need to be performed to design, build, and test the product.

1. Micro-Processor(s):
    a. Designing: There will be one on each board, and they are already designed and built for us. However, we have to engineer how to power them, allow them in interface with the user, output to the displays, get the input from the weight sensors, calculate the combinations of weights to determine the scores, store and update the scores, and interact with each other. The most complex part may be designing a way to have the two processors communicate with

each other. We will also have to engineer all of the programming needed to complete all of the features.

   b. Building: We will need to engineer how to build a board to incorporate all of the design features. This includes mounting and safety features for the components. We will have to engineer how all of the inputs and outputs are arranged and connected with the processors.
   c. Testing: Testing all of the features will be integral to the processors. Each program or script will have to be tested along with different inputs and outputs. The basic functions and possibly the additional features will also have to be tested. We will have to test if the environment will affect the communication between the processors, and if we can actually send signals back and forth effectively.

2. Weight Sensors:
   a. Designing: We will have to engineer how to interface the weight sensors output with the processors and how to mount them on the boards so they can actually give accurate weight. Choosing the type of weight sensor will also be important because they tend to be expensive.
   b. Building: They will have to be mounted in a way so that the force from the bags hitting the boards does not dislodge them. We will have to engineer how to take into account the weight already present and then the bags when they land on the board. Also, we will need to engineer how to incorporate a bag going through the hole.
   c. Testing: We will have to test the sensors to see if we can get enough accuracy to decide which team gets points, if they can withstand the bags hitting the board, and if we can actually interface the processors with them.
3. Display(s):
   a. Designing: Deciding what type of display that we can use will be important. If we cannot use a pre-made display then we will have to make a custom made, possibly LED, display. We will have to engineer a way to get the output from the processors to change the displays. Based on the type of display, we will have to engineer ways to get the team names, the scores, update the scores, and possibly other additional features. The displays have to be easily read even with a bright sun or lack there of.
   b. Building: If we have to create our display we will have to be able to show the scores and differentiate the teams. No matter what type of display(s) we choose to use, we will have to connect it (them) to the processors, mount the display(s), and power it (them).
   c. Testing: We will have to test if we can send signals from the processors to the display so that it shows the information correctly and test the display with environmental factors.

4. Power Source(s):
    a. Designing: We will have to engineer our own power source because most of the time when playing Corn-hole there is no easily accessible power outlet. Therefore, we have to use batteries of some sort that should be rechargeable. This may limit the types of batteries that we can use. We will have to engineer how to recharge the batteries and how the power is distributed between the elements of this system. It will be very important to choose the most effective batteries for our game especially if we cannot use rechargeable batteries. We might also have to engineer a way to use both a power outlet and a battery source just incase the user wants to play indoors with an outlet nearby. We will have to engineer a power distribution and safety system so as to maximize safely our usage of power.
    b. Building: We will have to build the mount and interconnects for the battery. This may also include an option for the use of a power outlet with a surge protector. We may have to build the circuit for the distribution of power.
    c. Testing: We will have to see how much power each element uses so that we can select the correct battery based on power requirements of the system. Minimizing the amount of power that each major functional block needs will be essential. We will have to test how long the batteries can last if our system is on.
5. User Interface(s):
    a. Designing: We will have to decide what type of user interface there is and how it will connect/interact with the processors. Choosing a specific type of output from the user interface will be important because the processors need to be able to process the information given.
    b. Building: We will need to be able to mount them to the boards, a separate component, or make them mobile possibly using RF. If we have to build them from scratch then we will have to build our own interface so that all of the features can be demonstrated. All of the "switches" either on a board or separately placed will need to be labeled to simplify the use for the user(s). They have to be connected to the processors either directly or through RF.
    c. Testing: We will have to test to see if we can communicate effectively with the processors. The user interface(s) must be able to send information to the processors so that the processors can accomplish the tasks given by the user.

The largest task is engineering a way to have all of the inputs/outputs to the microprocessors interact with each other. The processors may only be able to have a specific format of input and output creating a need to adapt the outputs and inputs of the other devices or have a "third-party"/converter between the major functional blocks. Information needs to be sent back and forth so that the game works correctly. Taking off-the-shelf items and adapting them for our own use will be a pretty complex engineering task. We will need to be able to power everything. Therefore, choosing and modifying each component will be essential in allowing us to use batteries. Also, since this is normally an outside game, we will have to take into account environmental factors that could affect our game operation. Once all of the major functional blocks are completed we will have to test each set of multiple components to make sure they are following the required features. Like all electronic games, this will be a highly intensive engineering input/output feat.

Conclusions

Cornhole is a very popular game today – from tailgates to backyards, it is a widely played game. We believe that we can combine this simple game with the technologies of the 21^{st} century to produce an exciting and revolutionary product.