EEL 5666C INTELLIGENCE MACHINE DESIGN LABORATORY

Final Report

Spring 2005
Clifford Pannella
Robot: "Dice"

TABLE OF CONTENTS

ABSTRACT

"Dice" is a craps playing robot. It is capable of searching out dice, playing a simplified game, and figuring out how well it turned out in its current session.

From start-up, Dice will attempt to search out any stray dice within the rolling area one by one. The value of each die, the side facing upward, will then be read. For this simplified version of craps, the many of the formal rules of the game will be sidestepped, both for simplicity and demonstration purposes.

The robot will keep tabs as to how many times it has won or lost in its current round of "gambling". After a set amount of rolls, he will display his final "bankroll", and sit idly.

EXECUTIVE SUMMARY

Dice's objective is to be interactive in a session at the craps table. Though his original design had him as an autonomous agent, he will rely on dice being rolled by a human hand. After pausing for this roll, the robot will scour the area searching for each die one at a time, using infrared sensors as a guide. When finding one, Dice will position itself over the die, placing it in its housing area. The housing area holds the special sensors which will read the upward facing values. The first die will be positioned in the nape of the V-shaped area, and the second will be adjacent to this. The placement of the dice is everything: if the special sensors move slightly, or if the dice are even as little as 2mm offset, the incorrect readings can occur. Thus it is pertinent that Dice's actuation be pinpoint.

Once the microcontroller receives confirmation that the dice are in place, it will go through its preset algorithm to present the user with the proper value, via the LCD. For this simplified version of craps, a win is achieved with a roll of 2, 3, 4, 5, 10, 11, or 12, while 6-9 takes a loss. These values give a 44.4% chance of winning – not improbable, but not favorable either, they were meant to exude a Vegas appeal. After a successful read, the LCD will update Dice's bankroll, and the robot will backup, allowing the user to pick up the dice and roll again. After three rolls, Dice has completed the current betting round, and will end the demonstration.

INTRODUCTION

Gambling is one of the biggest vices of the general public. For most people the thrill of the big win overshadows many of the ramifications that come along with gambling such as debt, helplessness, and depression.
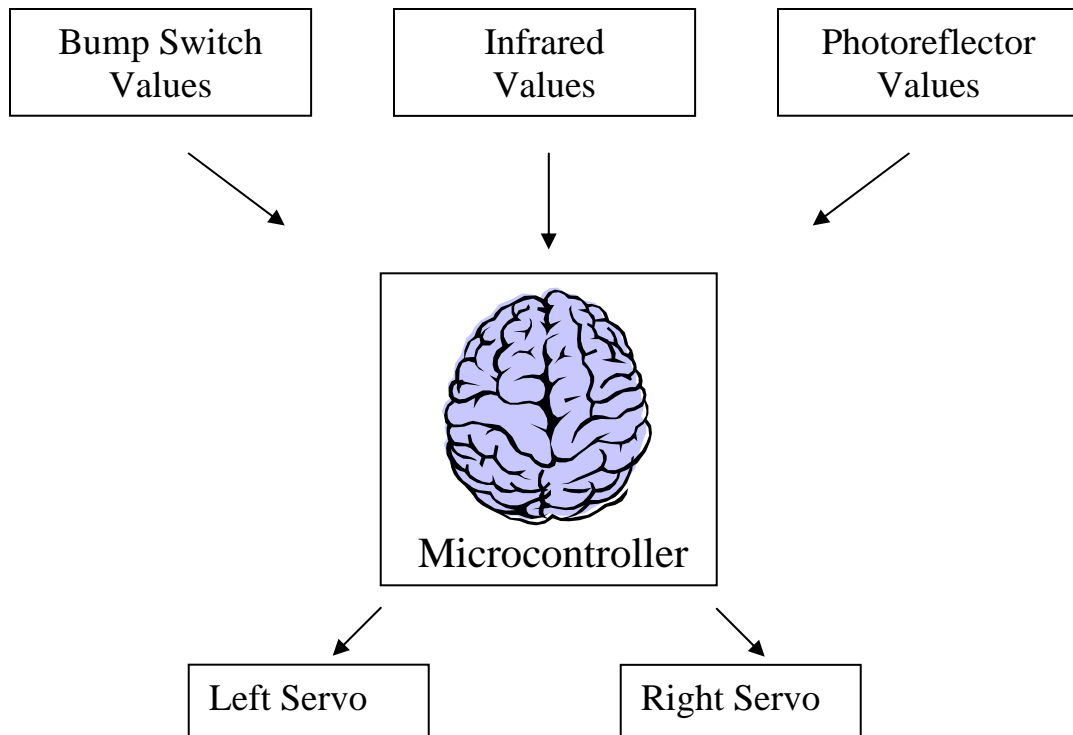
Fortunately, there is an alternative. With the recent explosion of television coverage on gambling, it is easy to see that people don't mind watching other people risk their funds. By watching others gamble, many avoid the financial and psychological dangers of betting their own money. Thus having a personal robot to play an entertaining game of dice can not only be relaxing, but also beneficial. The rollercoaster-like ups and downs of the gaming world can be accomplished without the fear of a direct consequence, and an enjoyable time may be achieved in the meantime.

Hence this project's purpose: an autonomous craps-playing robot. Win or lose, the robot feels nothing, and the viewer is entertained.

This paper will describe in sections how Dice will come together, laying out the specifics of parts as well as subsystems. The necessary movement and detection algorithms will be touched on, and the entire project will be clearly presented to the reader.

INTEGRATED SYSTEM

Dice is controlled by an ATMEGA128 Microcontroller on a MAVRIC-II board. This serves as the brain of the robot and makes all of Dice's decisions. These decisions are based on a variety of sensors, which are onboard the platform along with the microcontroller. These sensors constantly interact with the brain, much like the human sensory system interacts with a human brain. Bump and infrared sensors (IR) take in data for obstacle avoidance, and IR is also used to seek out and find dice that may be in the vicinity. Special sensors populated with photoreflectors sit underneath the robot to check if the dice have reached their "housing area", and then read these values. All of this information is relayed to the "brain" via voltages: bump sensors and photoreflectors both have a boolean output of 0V or 5V, while an analog to digital voltage conversion is necessary for IR, as they give a variable output. The microcontroller processes all of this information, and based on what is communicated to it, sends specific values to a pair of servos which control the movement of Dice. Figure 1 on the following page has a low-level block diagram showing the communication that takes place with Dice.

EEL5666C – INTELLIGENCE MACHINE DESIGN LABORATORY   CLIFFORD PANNELLA
INSTRUCTOR: DR. A. A. ARROYO   TAs: WILLIAM DUBEL, STEVEN PICKLES
Page 7 of 22

FIGURE 1

Unfortunately, the servos have no way of communicating back to the microcontroller if for some reason they receive a value and are unable to carry it out. This is why it is pertinent to have a full, competent sensor suite along with structured algorithms which take necessary precautions.

MOBILE PLATFORM

The robot must to be able to freely move back and forth as well as side-to-side to both

determine the position of the dice and approach them for value-reading. A platform that

accomplishes this needs to be small enough to avoid the obstacles presented to it, but

large enough to fit all necessary sensors, actuation devices, wiring, etc. I experimented

with a few different designs, but Figure 2 shows my final layout in Autocad:
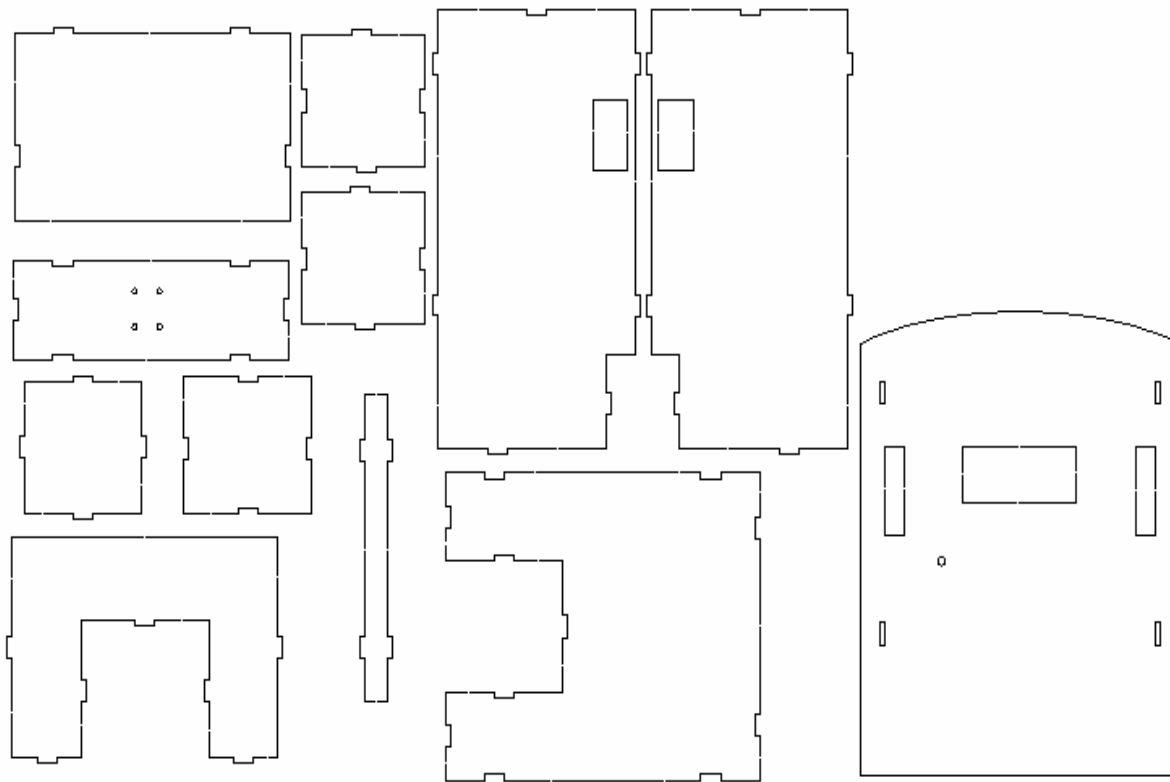
FIGURE 2

This design gave me large enough storage to be able to hold all onboard components, as

well as the necessary cutouts for servos, the Liquid Crystal Display (LCD), etc. It also

gave me a suitable housing area for the dice. Since it is difficult to tell what these pieces

are used for when drawn individually, a fully assembled picture is included in Figure 3:

FIGURE 3

The platform came out as good as I had hoped, but an embarrassing lesson was learned

along the way. LISTEN TO THE TA'S!! I was told numerous times to model my

Autocad design in three dimensions so as to make sure that everything would fit together

well. I failed to do this for my first design and was quite perturbed when everything was

printed out via the T-Tech machine and did not fit together. However, after going back

home and putting together a 3D design, I adjusted my measurements and turned in a

successful drawing.

ACTUATION

Dice moves around using two servos purchased from Acroname that were bought already modified for continuous rotation. I purchased these because I did not trust my ability to "hack" a servo and didn't want to be stuck with broken parts already paid for. They were more expensive than normal servos but I still found them to be a good value. A moderate speed can be achieved with their 0.15 speed through 60 degree rotation, and torque was not an issue as they could easily move my platform at 44 oz-in.

Attached to these servos are two 2 and ¾" inch diameter wheels also purchased from Acroname. These provided a large enough circumference for Dice to cover decent ground when traveling. Since I still needed an additional point of contact for a stable base, I looked for a caster to hold up the back end. This proved to be a difficult task as local stores turned up unfruitful searches. Luckily Ryan Moreau had a ball caster that he was not using and I was able to purchase it off of him. Figure 4 shows a close-up picture of the caster:

FIGURE 4

SENSORS

Bump Switches:

SCOPE:

Bump switches are very simple, convenient devices that give off a signal whenever they are activated (pressed).

THEORY:

The simplest of the sensors, the only necessary hookup for a bump switch is to attach an input pin to a voltage (5V). The corresponding output pin can then be attached to an I/O pin on the microcontroller. When the switch is hit by an object, the button is depressed, and the internal circuit connects the two pins, outputting 5V. Otherwise, a short circuit is created, and 0V is outputted.

OBJECTIVES:

The two bump sensors are placed on the back of Dice to determine if the platform has backed into something. Since there are no IR sensors covering the rear, something else was needed, and bump sensors provided an easy, inexpensive alternative.

REFERENCES:

Obtained at the IMDL laboratory

Price: Free

SHARP GP2D120 Infrared Sensor

SCOPE:

The ability of these infrared sensors is two-fold. First, they are the backbone of the obstacle avoidance routine that Dice uses. Second, they are used to detect whether there are dice on the floor that are accessible to the platform.

THEORY:

The sensor has an approximate 30 degree viewing window. An IR beam is sent out and reflects off the nearest object, if any. The beam is returned from this object and a signal processing circuit calculates an output voltage. The GP2D120 sensor differs from the standard GP2D12 by the fact that it has special lenses which give it a shorter detection range, which is conducive to better short-range readings (important for detecting dice). Figures 5 and 6 on the next page depict the block circuit diagram and performance graphs, respectively.

OBJECTIVES:

Two sets of IR's are used: one at dice level, one above it. If the IR sensors above dice level detect a close object, an obstacle has been reached and needs to be avoided. If the sensors at dice level detect something that is closer than what the top IRs read, than a die has been detected and it may be stored in the housing area for value-reading.

REFERENCES:

Vendor Address: www.acroname.com

Part Number: R146-GP2D120

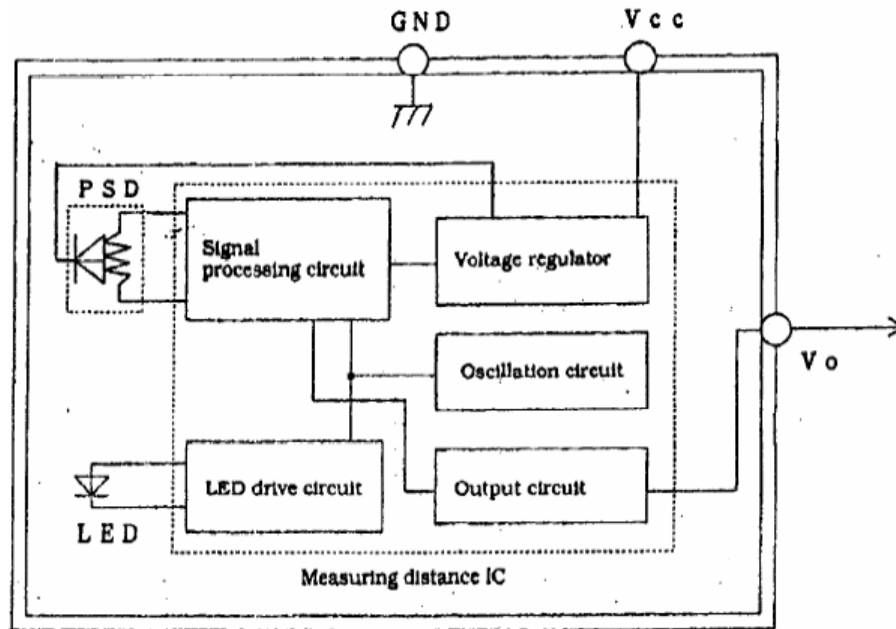Phone: 720-564-0373

Price: $12.50

FIGURE 5

OBJECT DISTANCE (cm)

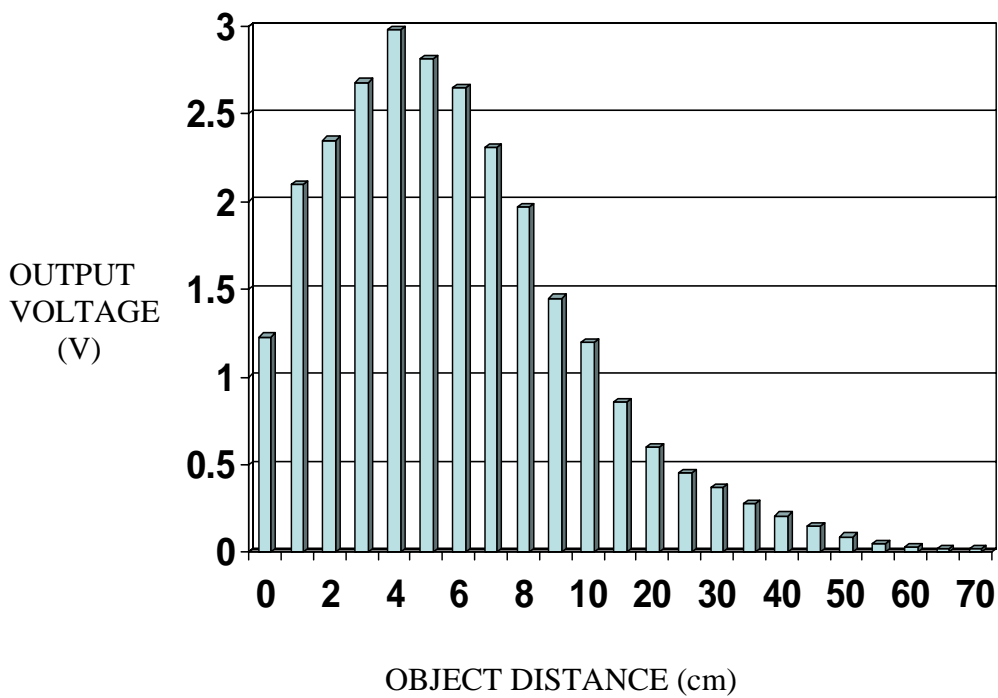**FIGURE 6**

Hamamatsu P5587 Photoreflectors

SCOPE:

The P5587 is used to detect the presence of white or black on the dice. The small size

(approximately 4mm x 4mm) creates the opportunity to have numerous photoreflectors

on one face of a die, so that a value may be read.

THEORY

The P5587 has 5 pins in its package. When a white object is placed above a

photoreflector in its readable distance (approximately 0.5mm-6mm), the output pin gives

off 5V. Otherwise, 0V is the output. A capacitance between power and ground and a

pair of resistors is necessary for the internal circuit to work correctly. Figures 7 and 8 on

the following page depict the circuit schematic and performance graphs, respectively.

OBJECTIVE

Setting up a circuit for the proper reading of each die proved to be a difficult task. With

Eagle software I attempted to make a printed circuit board (PCB) with 5 P5587's on it,

which is the minimum amount to be able to correctly read a face of a die (see the

Behaviors section for more). However, the trace auto-router could not route the board

fully because of the lack of room between photoreflectors (5 reflectors within a ½" x ½"

area). Wire-wrapping was used to connect the parts that could not be electrically

connected with traces. Each output pin was then connected to an I/O pin on the

microcontroller, and an algorithm was written to figure out which value was present

given the series of readings.

REFERENCES

Vendor Address: www.acroname.com

Part Number: R64-P5587
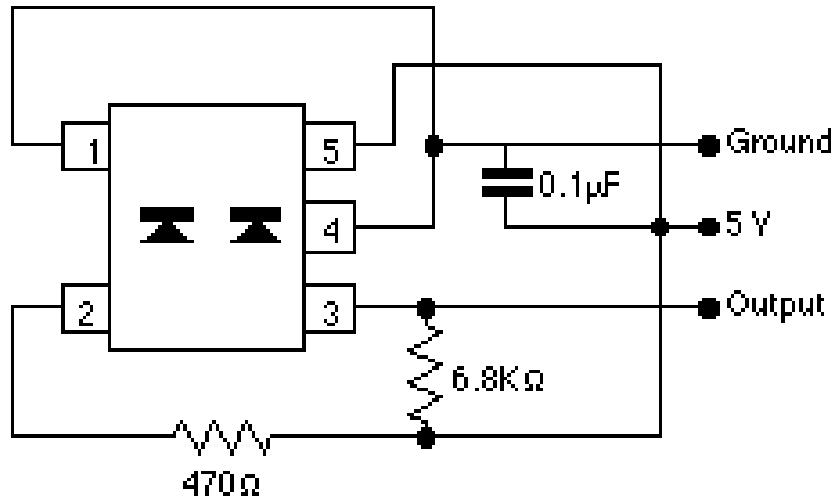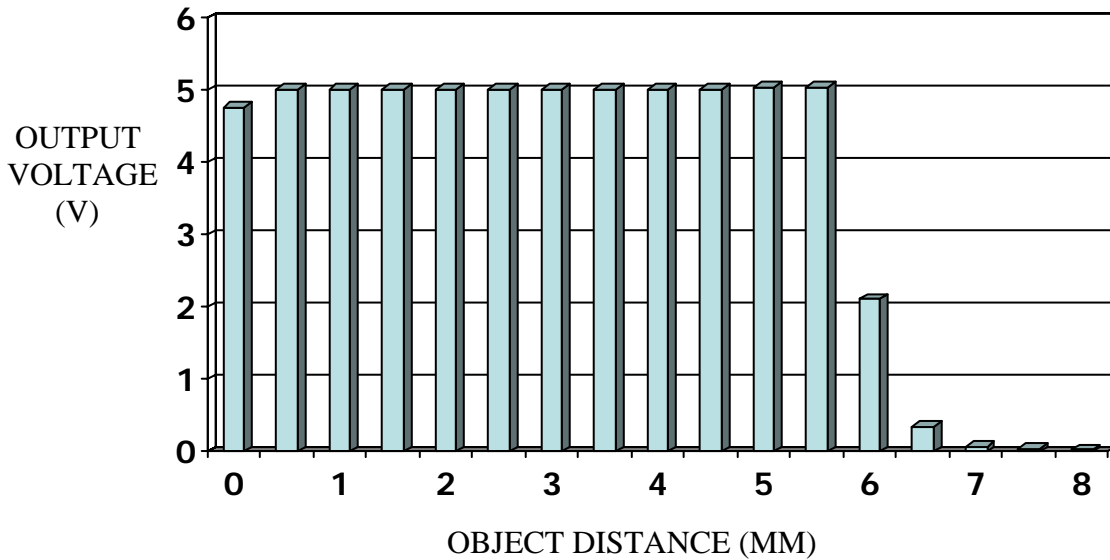
Phone: 720-564-0373

Price: $2.20

FIGURE 7

FIGURE 8

BEHAVIORS

Dice has three main behaviors: obstacle avoidance, dice locating, and value reading. Each of these had its own little quirks that made it difficult to accomplish.

Avoiding obstacles with infrared and bump sensors was not too difficult. The tricky part is figuring out how close to get to these obstacles – staying away is easy, but to traverse the entire playing area, the robot needs to be able to move around freely, and can't worry about a wall it sees 3 feet away. Another thing that needed to be overcome was the output of the IR sensors – they have a peak voltage at a certain distance, and both inside and outside this distance it is possible to have the same output, meaning without strong software the robot might think an object is a foot away instead of an inch and a half

Locating the dice was easier to prepare for than it was to actually perform. Using a system of vertical IR sensors, the microcontroller could tell whether an obstacle was at dice level, or if it was just a general object. If both sensors vertically arranged gave close to the same voltage output, than it must avoid this obstacle. However, if the bottom IR at dice level is reading a voltage that the upper sensor is not, a die has been found and needs to be housed correctly.

Reading the values of the dice was the most difficult of these three behaviors. After painstaking hours of using EAGLE software to build a PCB for my special sensor, I was finally able to print out a design, but still had wiring to do since the software could not auto-route the entire board (due to the close proximity of the photoreflectors). Even once

the PCB was fully functional, the placement of the board was key; it was very important

that I position it perfectly. Only after I verified the correct functionality of this single

board was I then able to print out a PCB for the second die – which needed to be a mirror

image of the first in order to correctly fit. The algorithm for reading the dice was

tweaked several times to lessen the amount of necessary photoreflectors, this information

is available in the sensor report.

EXPERIMENTAL LAYOUT AND RESULTS

Each subsystem of the robot required separate experimentation and testing to ensure

proper functioning. A lack of such rigorous research would most likely result in the

failure of the final product, due to an accumulation of small errors. Such an outcome is

nearly impossible to debug without again testing each subsystem. Therefore this analysis

was done at every step, no matter how trivial. Once each individual part was verified to

work, trials were constructed to ensure that the systems could be integrated together to

produce a successful product.

Extensive experimentation was done in regards to searching for the dice. I had to make

sure that the vertical alignment of the infrared sensors could differentiate between a

normal object in its path and a die that would need to be "picked up". I did a little testing

in software in changing the amount of times per second the analog to digital converter

read the IR output, because these values get averaged together and I wanted a more

precise measurement. I didn't want the robot to be in the middle of a turn and miss a die

right in front of it because the average voltage output was not specific enough. The speed

of the servos was slowed down in order for the sensors to better sense its surroundings

For reading the dice, software was not a problem, but hardware was the issue. I

purchased a temporary spray adhesive so that I could hold the PCB's down for small

amounts of time to test the success of the location. Only after triple checking the final

layout did I use a stronger substance. The angle of the "V" used to trap the dice was

made at right about 90 degrees to ensure a proper, stable cornering. Several test runs

were performed at this stage to see at what point the dice were best able to be read. This

is the reason why I added some minor servo movement to properly have the dice flush

against the sides, directly underneath the special sensors.

Some minor testing that was done included how the infrared sensors and photoreflectors

performed under different lighting conditions with different objects. This proved not to

be much of an issue. My main concern was if the special sensor would be able to read a

correct value given the dark atmosphere beneath the robot. Due to their well-designed

internal circuitry, however, photoreflectors do not rely on external light to base their

readings. Even in a dark, enclosed area, the readings were consistent to what I found

during normal tests. This made it much easier because I did not have to add any more

components such as internal LED's to ensure a fully functional robot.

CONCLUSION

This course has been a consistent learning experience for me. It has put me through valuable real-world activities such as project planning, budgeting time and money efficiently, and communicating with others to better achieve goals. Though I did not accomplish all my original goals for the term, I now have a better understanding of all of the above, and will actually take something concrete out of the course. I truly believe that you could fail IMDL and still learn more than if you coasted your way to an A in another class.

Some things that I originally planned to have functioning did not pan out. One was a chip stack attached to the back of the robot, which would serve as the current bankroll. This was quickly realized to be wishful thinking. The main thing that did not happen was a way to roll the dice. I was extremely disappointed in failing in this aspect, as nothing came to fruition in the closing weeks. I attempted numerous methods, including a high speed servo and a solenoid, but none would function well enough.

If I had to take the course over I would try to plan out my semester more thoroughly from the outset. The deadlines provided in the syllabus were good gauges, but their lack of enforcement sometimes led to a more complacent mindset. I would have started working on a dice-rolling mechanism earlier, and probably add some sort of speaker on the robot which would provide audio feedback based on the results of the roll. All-in-all, however, the course was extremely beneficial and I would recommend it to anyone with creativity and a proper background in engineering.

DOCUMENTATION

I would like to specifically thank the following sources: Steven Pickles for his tireless

efforts and patience when I bombarded him with multiple questions, William Dubel for

his superior knowledge of EAGLE software, Sara Keen for some minor software help,

Ryan Moreau for providing some crucial hardware, the people at Acroname for being

extremely cooperative, and Radio Shack and Electronics Plus for supplying parts locally.

APPENDICES

All source code is included on the CD turned in at the end of the course.