CATE
(Autonomous Cracker Dispenser)

Name: Mark Antilla
Date: December 10, 2002
TA: Uriel Rodriguez
Jason Plew
Instructor: A. A Arroyo

Table of Contents

I. Abstract

CATE is an autonomous agent that will serve as a caterer for parties. The robot will first dispense a pre-loaded stack of crackers onto a tray and then wander around a room asking people if they would like a cracker. While roaming around the room, CATE will demonstrate obstacle avoidance and person following.

II. Executive Summary

CATE is a robot that was designed to roam around a room and dispense crackers to people. The physical design of CATE consists of three parts: a mobile platform, a vertical loading stack, and a rotating tray. Crackers will be loaded from the stack onto empty positions on the tray, which are sensed by CdS cells mounted above the tray. With the tray full of crackers, CATE roams around a room looking for heat sources (assumed to be people) for whom it will serve crackers.

III. Introduction

A key ingredient to parties and other social gatherings are hors d'oeuvres for the guests to snack. Rather than have the host occupy him or herself with preparing these treats and constantly refilling a platter, it would be much more convenient for autonomous agent to tend to the matter. CATE will take over this task by automatically loading crackers onto a rotating circular platter from a vertical stack. Once this task is complete, it will roam around a room asking people if they would like a cracker. This paper discusses the physical designs of CATE (including actuator and sensor types), followed by a description of the behaviors it is capable of exhibiting.

IV. Integrated System

CATE is controlled by Atmel's ATmega163 microcontroller via Progressive Resources LLC's MegaAVR-Dev development board. This board will directly control all of the sensors, as well as the actuators for CATE.

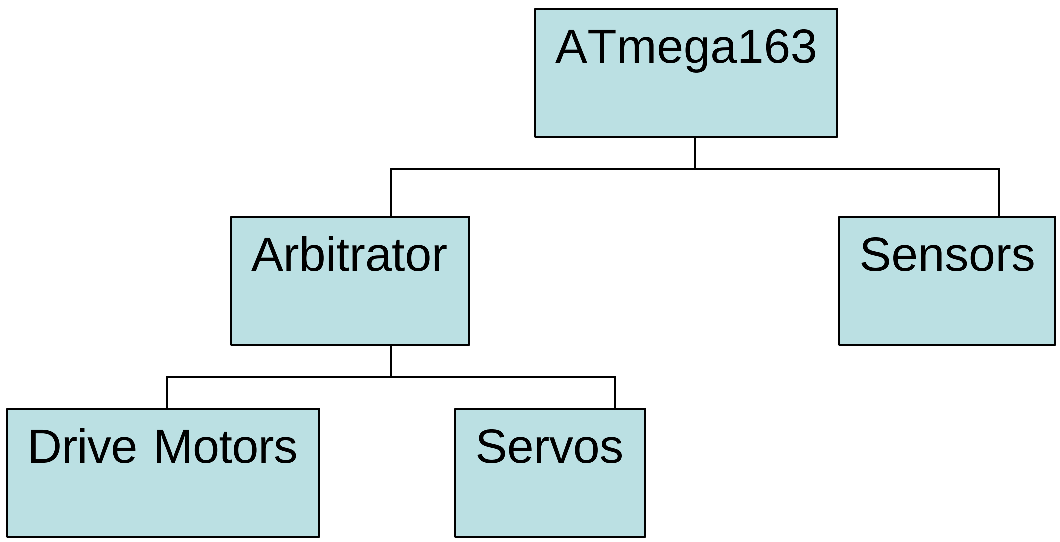

Figure 1: Block Diagram of Hardware

V. Mobile Platform

The platform for which CATE will use will be constructed from 1/8” aircraft plywood. The platform will consist of a base with two motor driven wheels and low friction post for balance. At the top of the base will be a horizontal, rotating platter (similar to a motor driven Lazy Susan). Above the platter will be a vertical shaft containing a stack of crackers. The shaft will have an opening at the top for the loading of the crackers and a partial opening at the bottom which will allow one cracker to be pushed out of the stack at a time. A side view of this can be seen in Figure 1.

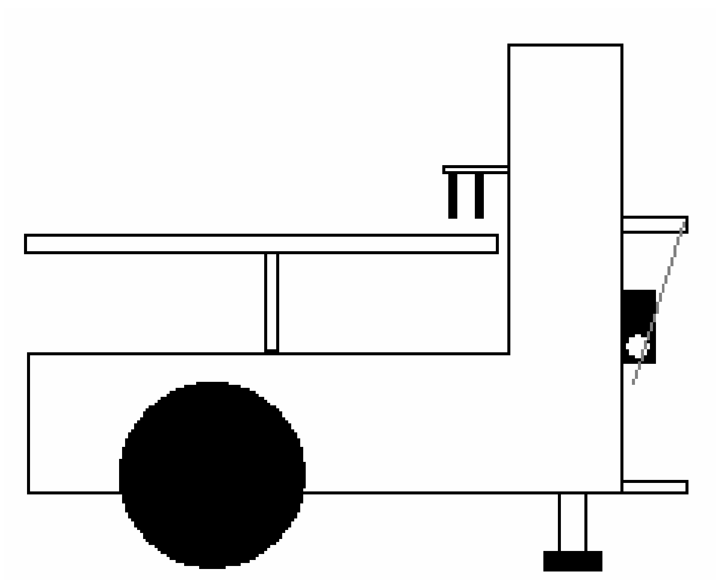

Figure 2 – Side view of CATE

VI. Actuation

Scope and Specifications

The purposes of the actuators are to: provide mobility for the platform, provide a means to rotate the tray, and to load a cracker onto the tray.

Mobility

The actuators chosen to drive the wheels of the platform were hacked Futaba servos. They were both mounted on the side of the platform and glued to 3.5" diameter model airplane wheels. Since they were hacked, they allowed for continuous rotation both forwards and backwards. Both servos were connected to output capture ports OC1 and OC2, respectively, on the microcontroller board. The ports generated a PWM signal that instructed the servos to rotate either forwards or backwards and at what speed.

Tray Rotation

A 12 V unipolar stepper motor was chosen as the actuator to rotate the tray. The reason for this is that the position of the motor (and thus the tray) could be controlled greatly. This would allow the tray to be divided up into eight equal sections, each to hold a single cracker. Also, a stepper motor allows continuous motion which provides a smoother appearance. A nylon gear was attached to the output shaft of the motor, which was placed next to a nylon gear of equal size attached to the shaft through the tray. The motor is connected to a Little Step-U stepper motor driver. This provides optoisolation as well as abstracts the motor control so that a single pulse can instruct the stepper motor to go to its next preconfigured position. Since a PWM signal is not necessary, the driver chip was attached to an output pin on Port D of the microcontroller.

Cracker Loading

An unhacked Expert servo was chosen as the actuator for the cracker loading arm. The servo's output shaft was attached to a bar, connected to a strip of wood (hand) at a pivot. The hand was bounded by wood so that it could only move horizontally through the mouth of the stack to push a cracker onto the tray. When the servo rotates, the bar forces the hand in or out of the stack opening. The servo was attached to the OC0 port of the microcontroller so that a PWM signal could control it.

VII. Sensors

Scope and Specifications

The purpose of CATE is to roam around a room and serve crackers to party guests. The purpose of the sensor suite is to provide environmental feedback to CATE. More specifically, the sensors allow the robot to effectively roam and seek out party guests. They also allow the tray of crackers to be reloaded when empty. The sensor suite consists of five groups of sensors:

Collision Avoidance – IR detectors

Collision Detection – Bump switches

Cracker Detection – CdS cells

Person / Activity Detection – Pyroelectric cell & CdS cell

Collision Avoidance

The collision avoidance sensor suite consists of four Sharp GP2D12 Distance Measuring Sensors from the Mark III Robot Store. Each sensor consists of an IR emitter and IR detector. The IR emitter modulates the IR signal at a frequency (approximately 40 kHz). The light signal bounces off an obstacle and is received by the IR detector. The time delay from the transmission to the reception of the signal is proportional to the distance the sensor is from the obstacle. This is converted to an analog voltage which can then be interpreted by the microcontroller. This type of sensor can be used by CATE to detect when an obstacle is near while there is still time to steer away. This theoretically, should prevent collisions from occurring.

The four IR sensors are placed in a configuration as seen in Figure 3 and connected to the microcontroller according to the schematic in Figure 4. The first sensor is aimed forward and connected to an analog port (Port A Pin 0). The second sensor is aimed backwards and connected to a second analog port (Port A Pin 1). The remaining two sensors are placed at a 90º angle in a cross-eyed pattern and connected to two more analog ports (Port A Pin 2 and Pin 3, respectively).

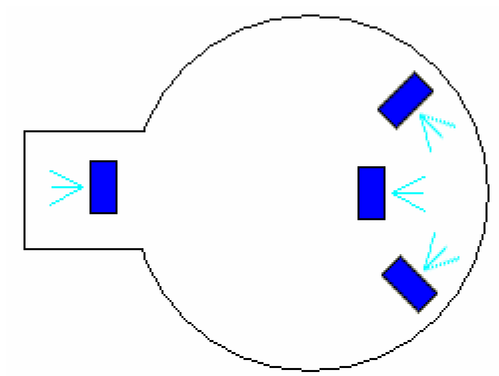

Figure 3: Bottom View of IR Sensor Placement

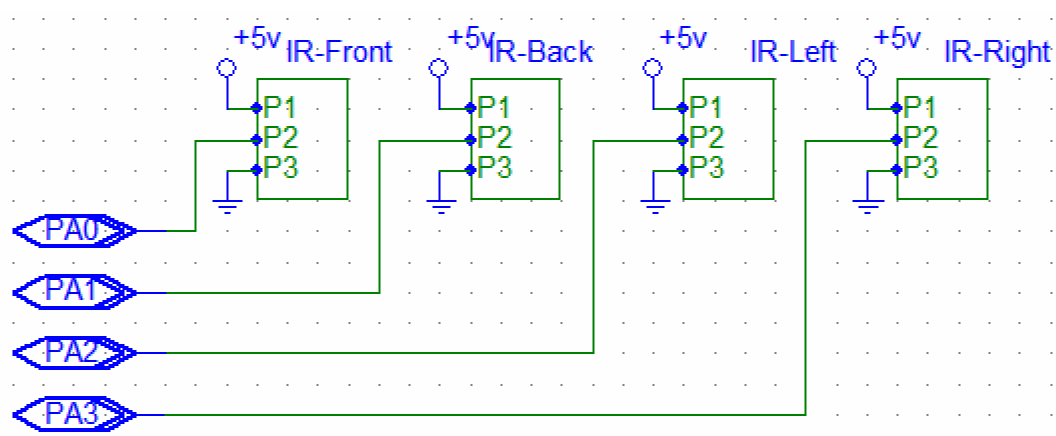

Figure 4: IR Sensor Schematic

One of the IR sensors was used to gather data on the output voltage at various distances as seen in Table 1. It should be noted that the Atmel ATmega163 processor has a 10-bit Analog-to-Digital conversion channel. For convince during the initial testing phase, the lower two bits were discarded so a 1-byte output could be displayed using the onboard LED's. The data is displayed in Table 1 and plotted in Figure 5.

Distance (inches)	Trial 1	Trial 2
2	129	128
3	117	118
4	117	115
5	112	111
6	108	107
7	104	102
8	102	100
12	96	96
24	91	90
36	91	90

Table 1: IR Sensor Output Voltages

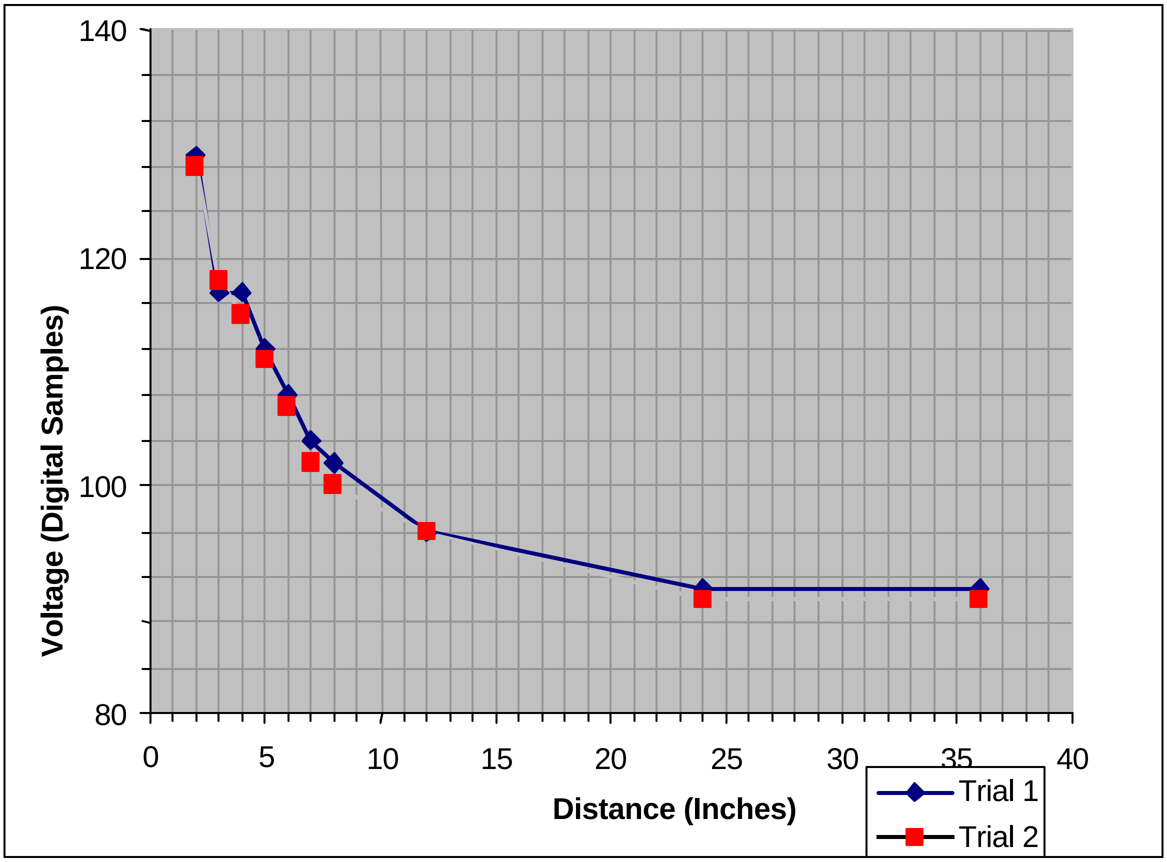

Figure 5: IR Sensor Output Voltages

Collision Detection

The collision detection suite consists of seven small normally open switches. Each switch is connected in series with a resistor. When one or more of the switches closes, a voltage divider network is created. Each resistor has a different resistance (in multiples of approximately 2). This is so that the output voltage connected to the analog input will be unique for each combination of switches. The purpose of the sensor suite is to detect when CATE has physically come in contact with an obstacle. The switches are attached to a bump skirt that surrounds the outside of the robot according to Figure 6. The schematic for the switches is shown in Figure 7.

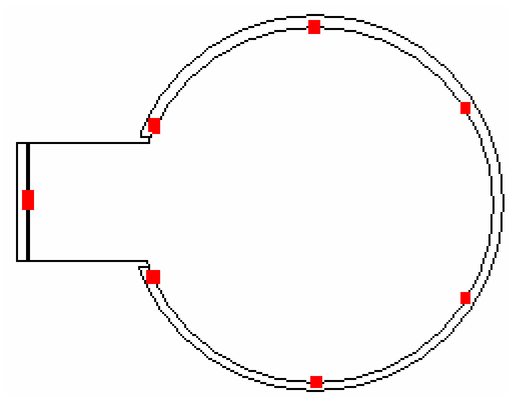

Figure 6: Cross Section Placement of Bump Switches on Bump Skirt

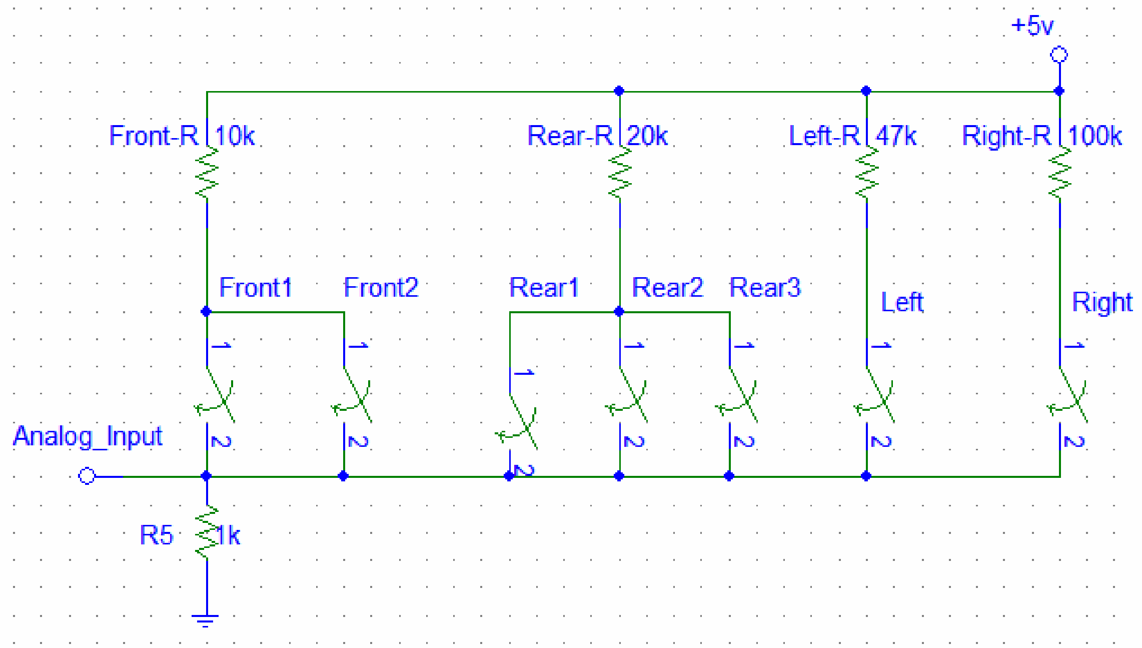

Figure 7: Bump Switch Senor Schematic

When one of the front switches is closed, the output voltage increases by 0.45 V. When one of the rear switches closes, the output voltage increases by 0.24 V. When the left switch closes, the output voltage increases by 0.1 V. When the right switch closes, the output voltage increases by 0.04 V. Therefore the output voltage (connected to Port A Pin 4) will range from 0.04 V – 0.83 V. All of the possible voltages are shown in Table 2.

Front	Rear	Left	Right	Output Voltage
open	open	open	open	0
open	open	open	closed	0.04
open	open	closed	open	0.1
open	open	closed	closed	0.14
open	closed	open	open	0.24
open	closed	open	closed	0.28
open	closed	closed	open	0.34
open	closed	closed	closed	0.38
closed	open	open	open	0.45
closed	open	open	closed	0.49
closed	open	closed	open	0.55
closed	open	closed	closed	0.59
closed	closed	open	open	0.69
closed	closed	open	closed	0.73
closed	closed	closed	open	0.79
closed	closed	closed	closed	0.83

Table 2: Output Voltages from Bump Switch Voltage Divider Network

Cracker Detection

The cracker detection suite consists of three collimated CdS cells coupled with three LEDs used to detect the presence of a cracker. A CdS cell is a device that varies in resistance inversely proportional to the amount of light absorbed. The cell has very low resistance in bright environments and very high resistance in dark ones. Since the cells are sensitive to all wavelengths of light, they are collimated with a piece of heat shrink tubing as seen in Figure 7. The tubing prevents ambient light from being absorbed by the cell and only allows light to come from a single direction.

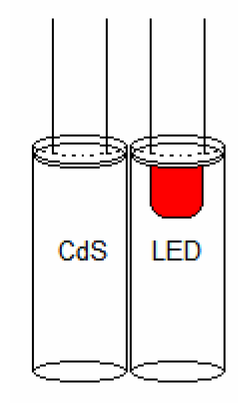

Figure 7: CdS Cell Collimation

When the collimated cells (coupled with the LED) are placed over a black material, very little light is reflected from the LED back to the CdS cell. Therefore, the cell will have a very high resistance (on the order of 4 MO). When the cells are above a white object (e.g. a cracker), much more light is reflected to the cell. This causes a smaller resistance of the cell (on the order of 220 kO). This sensor is used to detect when it is safe for a new cracker to be loaded onto the tray. Three CdS cells (collimated and coupled with a LED) are placed over the cracker loading area, as shown in Figure 8 and Figure 9.

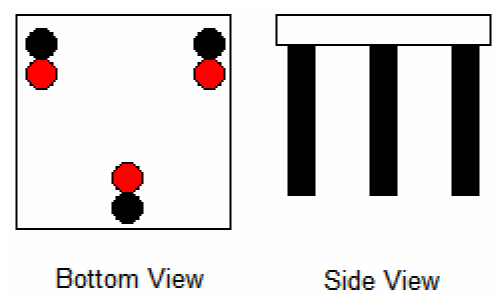

Figure 8: CdS Cell Placement

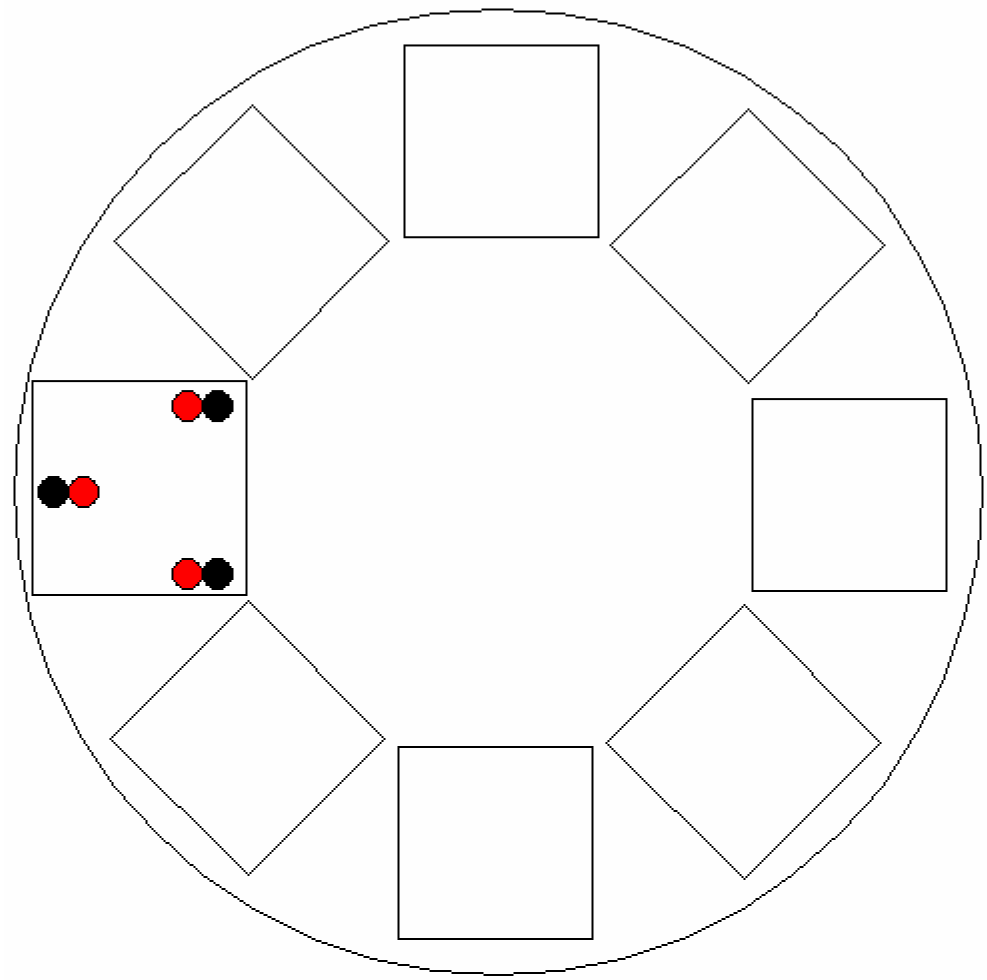

Figure 9: CdS Cell Placement Over Tray With Crackers

The circuits are assembled to the schematic in Figure 10. The three CdS cells are attached via diodes. Normally, the output of the circuit will be near 1 V. When one or more of the sensors detects the presence of a cracker (tray not ready to be loaded) then the output of the circuit to an analog input pin (Port A Pin 5) will increase to a voltage near 4 V.

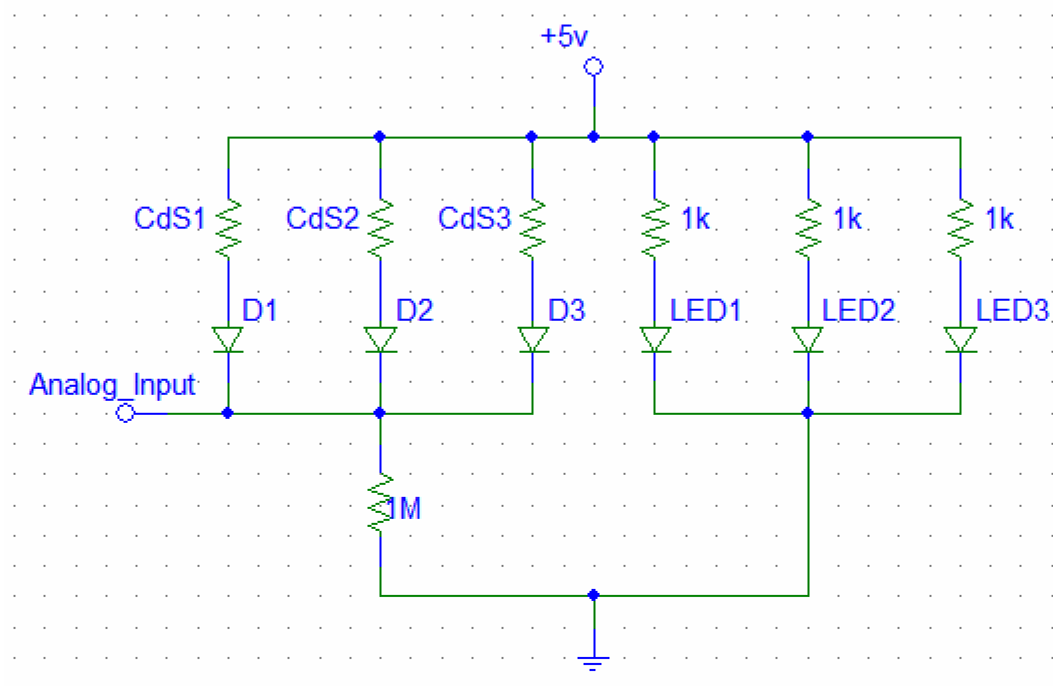

Figure 10: CdS Cell Cracker Detection Schematic

Person / Activity Detection

The person detection suite consists of a CdS cell and a pyroelectric sensor (Eltec 442-3 Pyroelectric Detector). As mentioned in the previous section, a CdS cell is a device whose resistance varies inversely proportionally with the amount of light absorbed. A pyroelectric sensor outputs a constant 2.5 V until it observes the movement of a heat source. If the heat source moves across the sensor's field of view to the left, the output voltage will decrease. Conversely, if a heat source moves across the sensor's field of view to the right, the output voltage will increase.

The CdS cell is mounted on the top of CATE as to measure the amount of ambient light. The pyroelectric sensor is mounted on the front of the robot angled towards the forward direction of movement. The sensor will be able to detect either: a heat source (person) moving in front of the robot or a heat source during a turn. The primary purpose of this sensor suite will be to determine if CATE should exhibit a “boredom” behavior when it does not sense any heat sources or if it should go into a power save / sleep mode when it does not sense any heat and the room lights are off. A schematic of this sensor suite is shown in Figure 30.

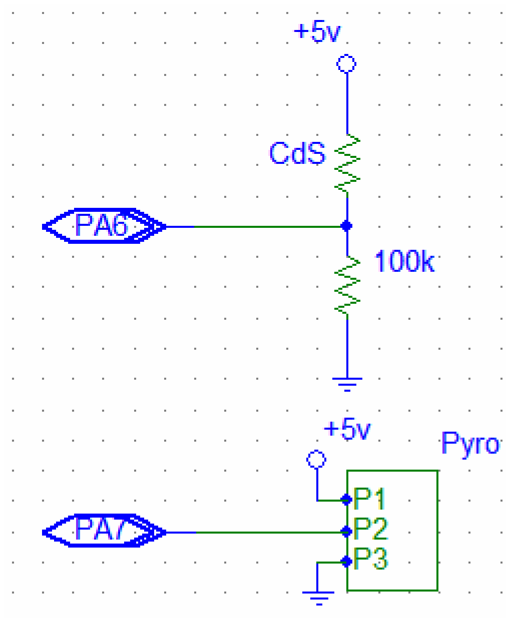

Figure 30: Schematics for Person / Activity Detection

The CdS cell was found to have an average resistance of 3.5 kO in a well lit room. It was also found to have an average resistance of 40 kO in a dark room. In a pitch black environment, the cell had a resistance greater than 20 MO. From this information, it can be concluded that a voltage on Port A Pin 6 that is above 4 V is most likely a bright room, where as anything below that is a dark room.

The pyroelectric sensor was had a standard output of 2.5 V when there is no moving heat source. When a hand moves quickly from the right to left side of the sensor's field of view, the voltage on Port A Pin 7 drops to 0.8 V.

Sensor Integration

The collision avoidance / detection sensor suites are used in conjunction for the basic behavior, which is roaming. The cracker detector sensor suite is used for the occasional behavior of reloading the tray with crackers. The person / activity detection sensor suite is used to control the transition into the "boredom" behavior. The pyroelectric sensor is also used to supplement the chosen direction of the roaming behavior.

Lessons Learned

I learned from my experiments that the sensors are delicate components and it is wise to not exceed their input current/voltage ratings. I accidentally connected the pyroelectric sensor to 120V AC for a brief instance. This caused a large spark and smoke came from the sensor. Afterwards, the output signal was constantly at 5V, regardless of the motion of a heat source.

VIII. Behaviors

Scope

CATE will exhibit four main types of behavior:

Roaming

Bored

Loading

Sleep

The intent of these behaviors is allow CATE to perform the task of dispensing and serving crackers in an adaptive environment such as a party with several people.

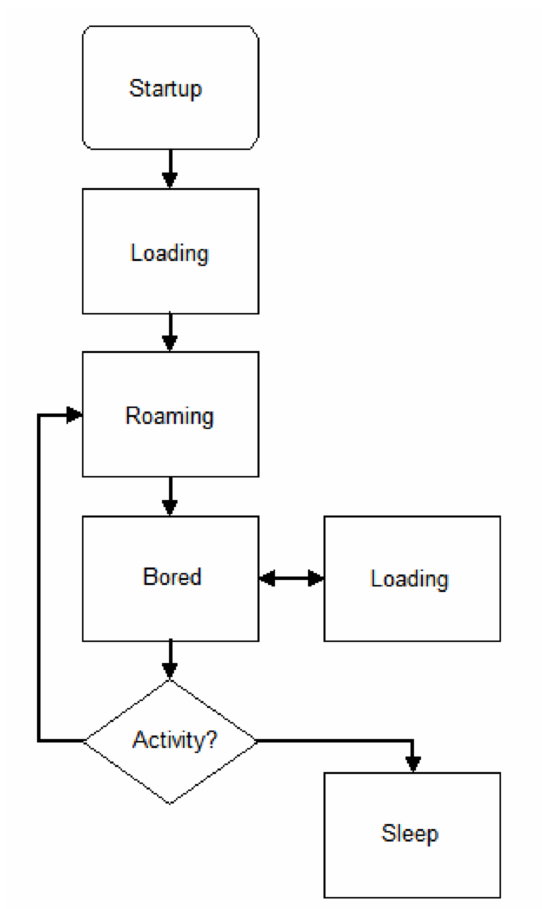

Figure 31 – Block Diagram of Behavior Transition

Roaming

The first behavior is Roaming. While in this mode, CATE will demonstrate basic obstacle avoidance. The IR sensors will detect when an obstacle is near and the robot will attempt to steer away from the obstacle. The bump switches will detect when a collision occurred and cause the robot to immediately turn in the opposite direction. This mode will be the default mode that CATE is in while serving crackers.

Bored

The second behavior is Bored. After a certain amount of time demonstrating the Roaming behavior, the robot will become "bored" with roaming and stop doing obstacle avoidance. It will then spin in a circle (direction is random), while looking for environmental activity. During this time, the pyorelectric sensor will attempt to detect the presence of a heat source. The process of spinning in a circle will simulate the effect of a moving heat source and generate a voltage depending on which direction it is spinning. Upon finding a heat source, CATE will stop spinning and revert back to Roaming in the direction of the source. The default time to remain in Roaming before becoming "bored" is 30 seconds. While in the Bored behavior, ambient light levels are also sampled. The maximum time to be in the bored state without finding a source is 30 seconds.

Loading

The Loading behavior will be executed initially when the robot first powers up and then during every other entry into the Bored state. During this behavior, the robot will not be mobile. The tray will move to each of the eight positions. For each position, CATE will attempt to detect if a cracker is present. This is done via the collimated CdS cells coupled with the LEDs above the loading position. If most of the light is detected by the cells, then it can be assumed that a cracker is present and it will move to the next position. Otherwise, it will assume the position is empty and attempt to load a cracker. After the loading attempt, it will attempt to detect if the load was successful. After three unsuccessful loading attempts, CATE will abort the loading activity and return to the next scheduled behavior.

Sleep

The final behavior is Sleep. If the robot exits from the Bored behavior two consecutive times without success, it will be assumed that there are no people in the general vicinity. It will then shut down for a period of a minute to conserve battery power. While in the Bored behavior, if the ambient light sensing CdS cell detects that there are no lights in the room, it will cause it to enter the Sleep mode for a period of 2 minutes. This is because it is assumed to be night and no people in the room for crackers to be served.

Lessons Learned

From experiments and observations of the behaviors it was learned that the set of behaviors can cause a trap. Sometimes CATE will be fooled into following a heat source other than human (such as a computer monitor). It will navigate in the direction of the heat source and upon reaching it will obstacle avoid away from it. When the time out has expired, it will navigate back to the source. This produces a patrolling like behavior where it repeats approximately the same path back in forth in the instance that there is only one heat source. Also, due to the time delay from when a heat source is sensed and the robot stops spinning, it will not navigate in the exact direction of the heat source, but rather 15 degrees past the source. Since an exact pin point of the source is not necessary, this is acceptable.

IX. Conclusion

In this project, I accomplished a simple autonomous robot that is capable of avoiding obstacles and loading a cracker onto a tray. It can also locate a heat source and navigate in its general direction.

Some areas of the project that I feel could use more work include the mechanical design and application of interaction sensors / actuators. The intent for the robot is to serve crackers to people, yet the serving tray is only eight inches off the ground. It is possible that it could be used on a table top, but "edge of the world" detection would then be necessary. The design is mainly circular (similar to a TJ style robot), but has a rectangular addition to the back. Also since the servo and loading arm potentially stick out three inches, the rear bump skirt was extended which makes the design awkward with it comes to turning. Perhaps a future design could be made where a different type of loading device is used that would allow for a more symmetrical design. Another area for improvement includes adding a device which asks a person if they would like a cracker. It could wait for a verbal response from the person, or navigate away upon a negative response. An improvement on the scope of the project could be an additional loading device that dispenses additional hors d'oeuvres such as slices of cheese to be place on the crackers.

X. Appendix A – Obstacle Avoidance Code

```
/* do basic obstacle avoidance according to the following algorithm:
    if bump switch pressed, then move in opposite direction.
    if no switch pressed, move away from IR detection unless the previous
     direction was backwards, then turn right so not stuck in a loop*/
void obstacle_avoid()
{
    while (1)
    {
        /* get updated sensor values */
        get_update_bump();
        get_update_IR();

        /* update direction based on bump switches */
        if (bump_switch == no_switches)
        {
            (direction == go_backwards)?(direction = turn_right):update_direction_IR();
        }
        else if (bump_switch == front_switch)
        {
            direction = go_backwards;
        }
        else if (bump_switch == right_switch)
        {
            direction = turn_left;
        }
        else if (bump_switch == left_switch)
        {
            direction = turn_right;
        }
        else if (bump_switch == rear_switch)
        {
            direction = go_forward;
        }
        else    // something is wrong since this should not be called, but go forward anyway
        {
            direction = go_forward;
        }

        /* change motor direction */
        set_update_motors();

        /* wait a second before choosing new direction */
        sleep(1);
    }
}
```

```c
/* update direction based IR sensors */
void update_driection_IR()
{
    if (center_ir == ir_very_near)
    {
        direction = go_backwards;
    }
    else if (center_ir == ir_near)
    {
        direction = turn_soft_right;
    }
    else if (right_ir == ir_very_near)
    {
        direction = turn_left;
    }
    else if (right_ir == ir_near)
    {
        direction = turn_soft_left;
    }
    else if (left_ir == ir_very_near)
    {
        direction = turn_right;
    }
    else if (left_ir == ir_near)
    {
        direction = turn_soft_right;
    }
    else if (rear_ir == ir_very_near)
    {
        direction = go_forward;
    }
    else if (rear_ir == ir_very_near)
    {
        direction = go_forward;
    }
    else  // no IR readings
    {
        direction = go_forward;
    }
}
```

```c
void get_update_bump()
{
    if ((BUMP_PORT & FRONT_BUMP_V) != 0
    {
        bump_switch = front_switch;
    }
    if ((BUMP_PORT & LEFT_BUMP_V) != 0
    {
        bump_switch = left_switch;
    }
    if ((BUMP_PORT & RIGHT_BUMP_V) != 0
    {
        bump_switch = right_switch;
    }
    if ((BUMP_PORT & REAR_BUMP_V) != 0
    {
        bump_switch = rear_switch;
    }
    else
    {
        bump_switch = no_switches;
    }
}
```

```
void get_update_IR()
{
    if (LEFT_IR_PORT < LEFT_IR_V_NEAR_THREASH)
    {
        left_ir = ir_very_near;
    }
    else if (LEFT_IR_PORT < LEFT_IR_NEAR_THREASH)
    {
        left_ir = ir_near;
    }
    else
    {
        left_ir = ir_far;
    }

    if (RIGHT_IR_PORT < RIGHT_IR_V_NEAR_THREASH)
    {
        right_ir = ir_very_near;
    }
    else if (RIGHT_IR_PORT < RIGHT_IR_NEAR_THREASH)
    {
        right_ir = ir_near;
    }
    else
    {
        right_ir = ir_far;
    }

    if (FRONT_IR_PORT < FRONT_IR_V_NEAR_THREASH)
    {
        front_ir = ir_very_near;
    }
    else if (FRONT_IR_PORT < FRONT_IR_NEAR_THREASH)
    {
        front_ir = ir_near;
    }
    else
    {
        front_ir = ir_far;
    }

    if (REAR_IR_PORT < REAR_IR_V_NEAR_THREASH)
    {
        rear_ir = ir_very_near;
    }
    else if (REAR_IR_PORT < REAR_IR_NEAR_THREASH)
    {
        rear_ir = ir_near;
    }
    else
    {
        rear_ir = ir_far;
    }
}
```

```c
void set_update_motors()
{
    if (direction == go_forward)
    {
        set_left_motor(LEFT_MOTOR_FULL_AHEAD_PWM);
        set_right_motor(RIGHT_MOTOR_FULL_AHEAD_PWM);
    }
    else if (direction == go_backwards)
    {
        set_left_motor(LEFT_MOTOR_FULL_BACK_PWM);
        set_right_motor(RIGHT_MOTOR_FULL_BACK_PWM);
    }
    else if (direction == turn_left)
    {
        set_left_motor(LEFT_MOTOR_HALF_BACK_PWM);
        set_right_motor(RIGHT_MOTOR_FULL_AHEAD_PWM);
    }
    else if (direction == turn_right)
    {
        set_left_motor(LEFT_MOTOR_FULL_AHEAD_PWM);
        set_right_motor(RIGHT_MOTOR_HALF_BACK_PWM);
    }
    else if (direction == turn_soft_left)
    {
        set_left_motor(LEFT_MOTOR_HALF_AHEAD_PWM);
        set_right_motor(RIGHT_MOTOR_FULL_AHEAD_PWM);
    }
    else if (direction == turn_soft_right)
    {
        set_left_motor(LEFT_MOTOR_FULL_AHEAD_PWM);
        set_right_motor(RIGHT_MOTOR_HALF_AHEAD_PWM);
    }
    else
    {
        set_left_motor(LEFT_MOTOR_STOP_PWM);
        set_right_motor(RIGHT_MOTOR_STOP_PWM);
    }
}
```