EEL5666C IMDL Spring 2006

Student: Andrew Joseph

Alarm-o-bot

TAs: Adam Barnett, Sara Keen
Instructor: A.A. Arroyo

Final Report
April 25, 2006

Table of Contents

	Page
Abstract	**3**
Executive Summary	**3**
Introduction	**3**
Integrated System	**4**
Mobile Platform	**5**
Actuations	**5**
Sensors	**6**
Behaviors	**13**
Results	**13**
Problems	**14**
Conclusion	**14**
References	**14**
Appendix	**15**

Abstract

Have you ever had trouble waking up in the morning? It's even worse when you realize that somehow you hit the snooze button in a daze and just kept right on sleeping. Well Alarm-o-bot is for you. When your alarm goes off, Alarm-o-bot goes and finds a place to hide and you must actually get up out of bed to find it if you want to turn the alarm off or hit snooze. If you hit snooze, Alarm-o-bot will again go off and hide. This will make it harder for people to go back to sleep once waking up. In addition, the alarm will not go off if there is not someone sleeping in the bed, preventing unnecessary searching for Alarm-o-bot.

Executive Summary

Alarm-o-bot is a completely autonomous robot with the purpose of waking up users by forcing them to get out of bed to snooze their alarm clock. The alarm clock remains on the table so that the sound is still in the user's ear when it goes off.

The Mavric-IB is the core of the robot. It contains the popular Atmega128 micro controller. WinAVR and the PonyProg program were used to program the C/C++ code and to down the C program to the board.

Alarm-o-bot uses two hacked continuous rotating servos for moving around. To sense the world around the robot, 4 different sensors were used. IR sensors are used to detect nearby objects and avoid them, a bump sensor is used to act as a snooze button, CDS cells are used to detect light variation in the environment, and a RF transmitter/receiver pair is used to communicate between the alarm clock and moving robot portion.

The purpose of this report is to document all the steps taken into the creation of Alarm-o-bot with the aim to have it reproduced, if desired, by anyone that reads this report.

Introduction

If you have a job that requires you to get up early in the morning, then you are most assuredly familiar with an alarm clock. It is hard to imagine waking up without one. For all its benefits however, alarm clocks still have their faults. Once we get used to hitting the snooze button, it can become a subconscious activity that prevents us from actually ever having to wake up. At that point it becomes ineffective in achieving its purpose. Alarm-o-bot is a prototype solution which aims to help alleviate the problem by making you get up out of bed in order to snooze or turn off the alarm clock.

This paper will start out by describing the organization of the system that constitutes Alarm-o-bot and describe each function on a high level. Then the mobile platform will be explained, followed by actuation and the sensors. Finally the behaviors are expanded upon along with the results obtained from testing the system. Final remarks and documentation conclude the paper.

Integrated System

Alarm-o-bot is controlled by the Atmega128 micro-controller on a Mavric-IB development board. Features of the board are 128K-program flash, 4K static RAM, 4K EEPROM, dual level shifted UARTs, RS485, I2C, up to 53 I/O pins, and 16MHz clock.

The robot will be equipped with an LCD display for feedback information and continuous rotating servos for locomotion. There will also be about 2 IR sensors (GP2D12) for object detection/avoidance, 3 CDS cells for light variance detection, 1 bump switches for a snooze button, and a RF TX/RX pair for wireless communication between alarm clock and Alarm-o-bot.

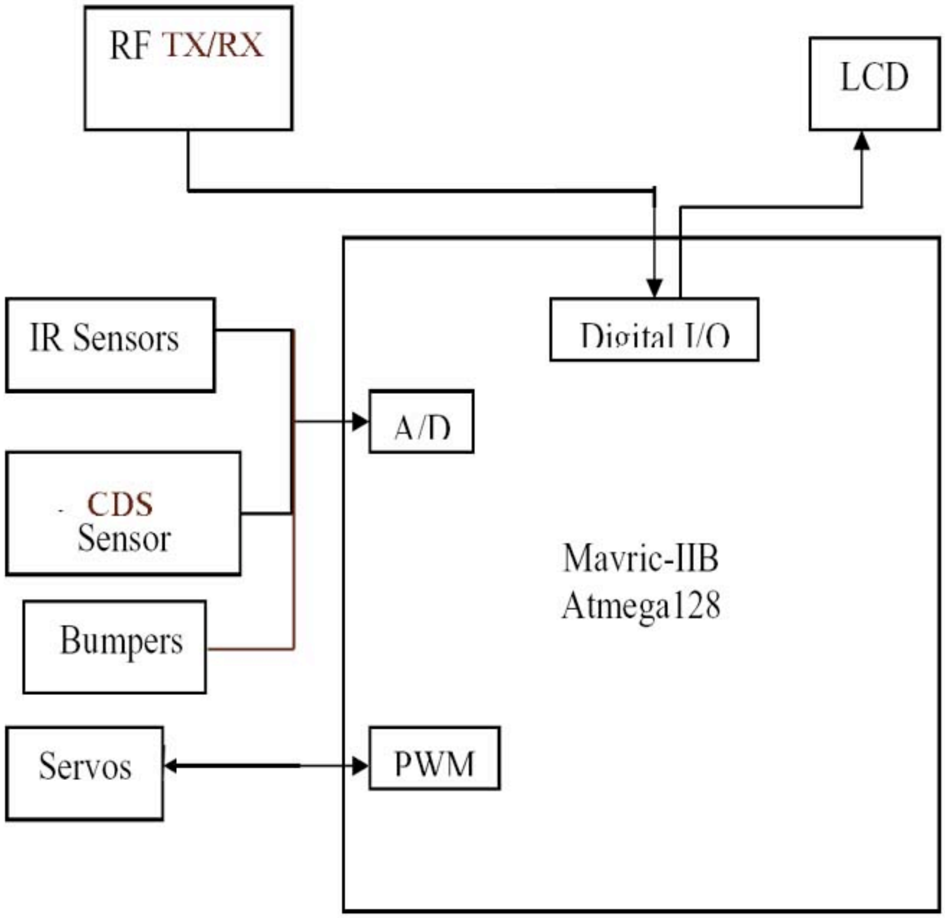

Figure 1: Hardware Organization

\

Mobile Platform

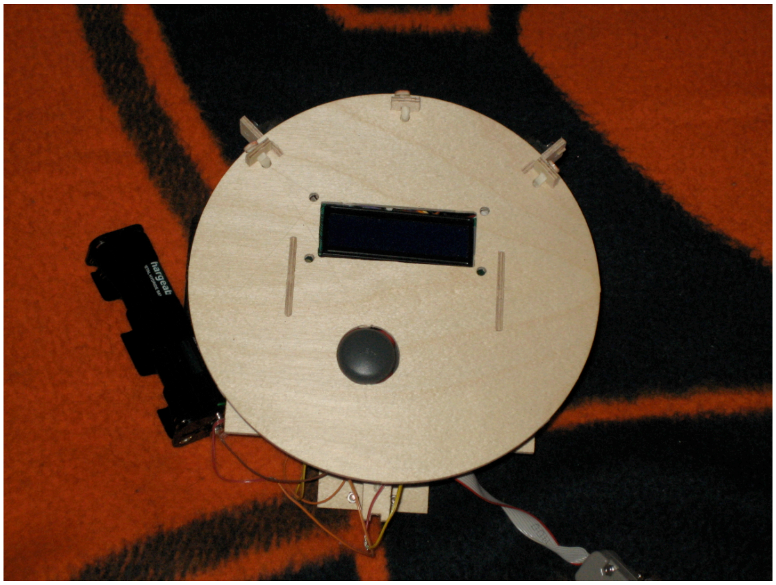

Figure 2: Alarm-o-bot top view

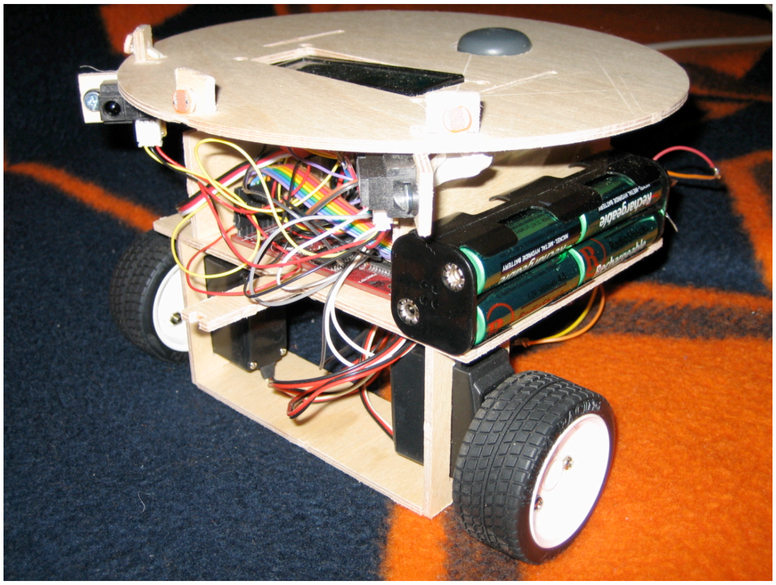

Figure 3: Alarm-o-bot side view

Actuation

The R174-CONT-RO-SERVO is a continuous rotating servo (hacked) which is used in Alarm-o-bot as locomotion. The Parallax Continuous Rotation servo is ideal for robotic products that need a geared wheel drive or other projects that require a 360 degree rotation geared motor. The Parallax Continuous Rotation servo output gear shaft is a standard Futaba configuration. This servo has a JR/Hitec connector which mates directly with a 0.1 inch 3-pin header.

Dimensions:	40.5 x 37.9 x 19.7 mm
Weight:	45.0 g
Output Torque:	3.4 kg-cm
Operating Speed at 4.8v	0.23 sec/60 degrees
Power Consumption:	6.0v/12mA at idle

Table 1: Servo Characteristics

Sensors

The four types of sensors I am using are bump switches, CDS cells, RF transmitters/receivers, and IR sensors.

IR Sensors

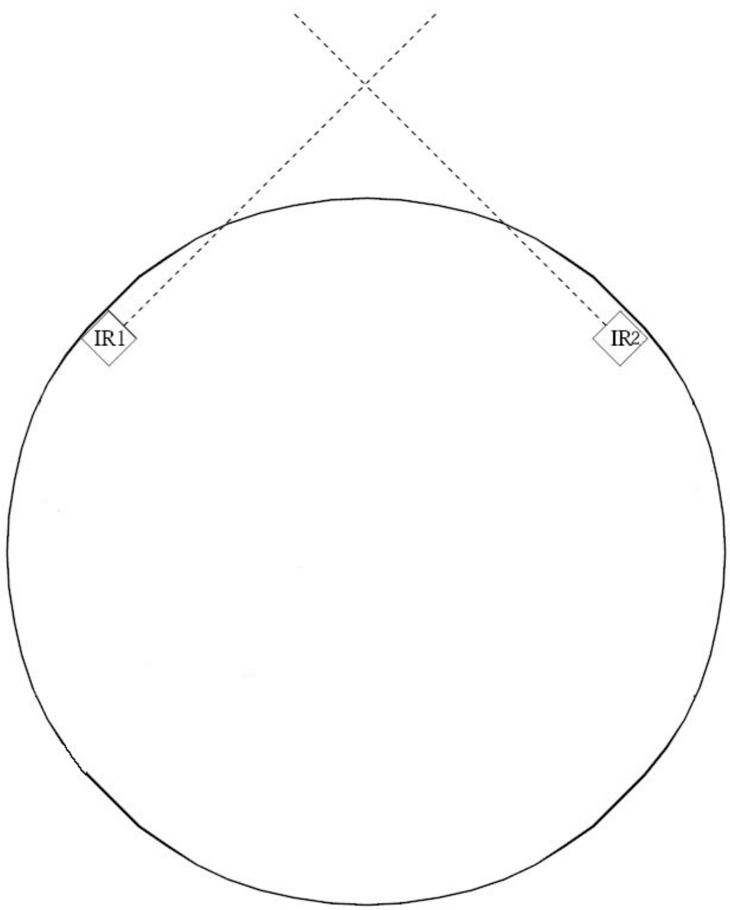

Figure 4: IR Sensor arrangement on Alarm-o-bot platform

Two IR sensors will be located on the wood platform as shown above. For this particular project GP2D12 IR sensors are used as the basic distance sensor. Each IR sensor is attached to the Mavric ib board through dedicated A/D ports. The sensor has an emitter which sends out an IR pulse and a receiver which records the reflected pulse. Depending on the received pulse, the sensor can approximate the distance of the obstacle. The distance is reported as an analog voltage with a range of 10cm to 80cm.

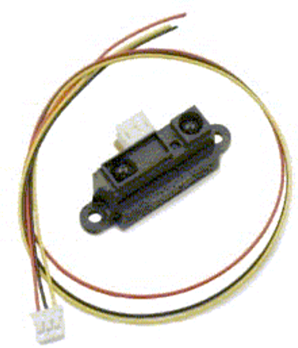

Figure 5: GP2D12 IR Sensor

From each port, an ADC value is attained through converting the voltage on the pin. This number ranges from about 500 (approximately 2.4V) which is within several inches of the sensor, to about 50 (about 0.4V) which is essentially out of detection range. This ADC value is a unitless number that corresponds to the distance an object is away from the sensor. The higher the number, the closer an object is to the IR sensors. A quick conversion (good estimation) leaves us with the distance from the pin voltage: Distance = 27*(voltage)^-1.1.

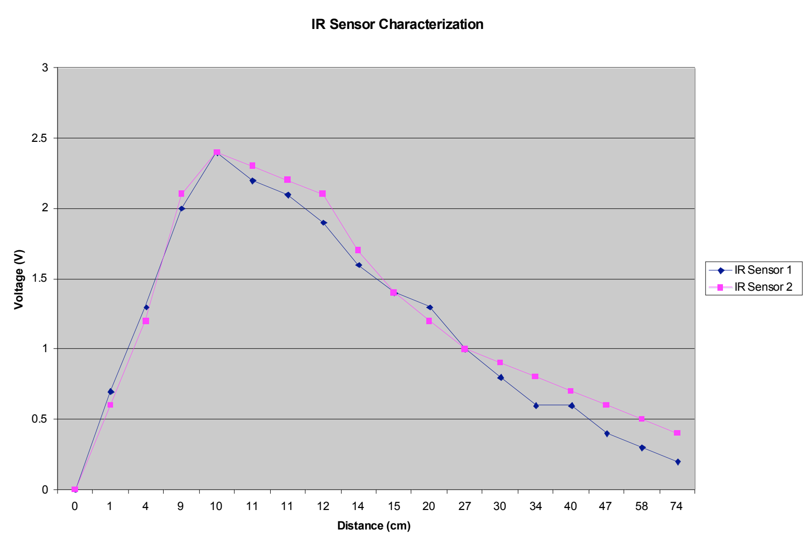

Figure 6: IR Sensor Characterization

From Figure 6, we can see that the two IR sensors are very close to each other as far as calibration is concerned. This allows for accurate comparable threshold values for both sensors. Alarm-o-bot will be self calibrating. The user will be able to power on the robot with it placed 10 cm from a corner in a room and from that value, the program will calibrate the threshold to be used the rest of the time it is powered on.

CDS Sensors

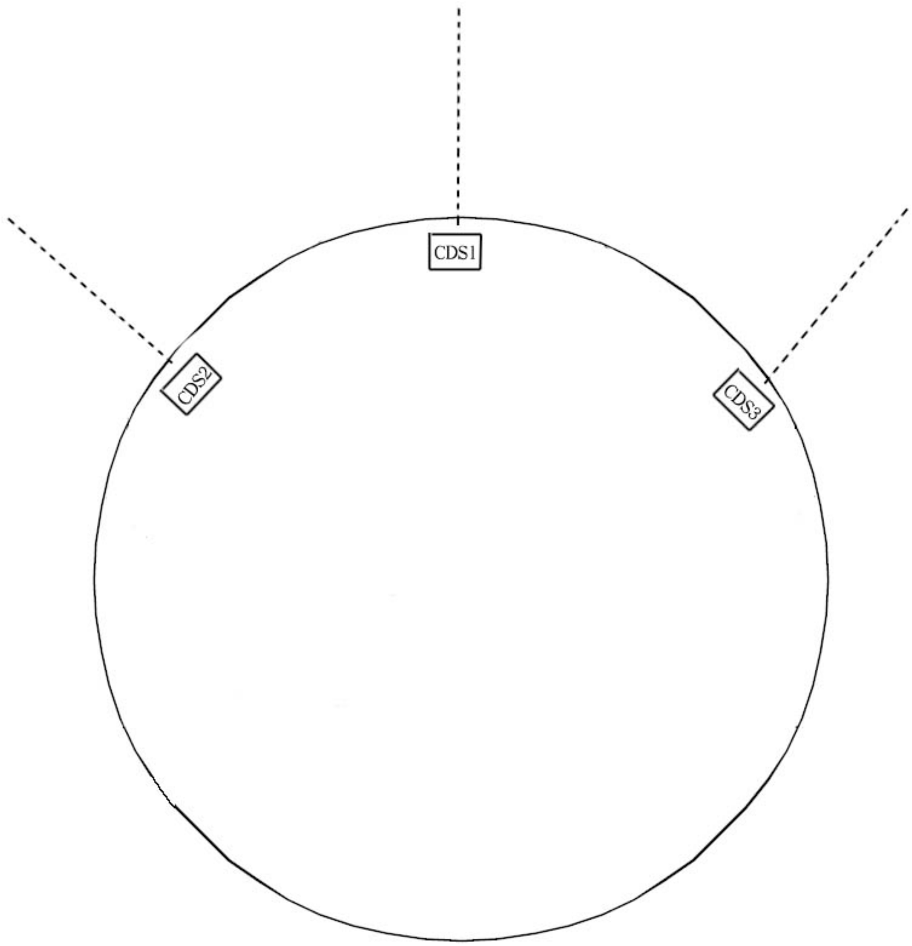

Figure 7: CDS Sensor arrangement on Alarm-o-bot platform

As shown in the figure above, there are 3 CDS sensors placed at the front of the platform. These sensors are used to determine the darkest area in a room that the robot should move into.

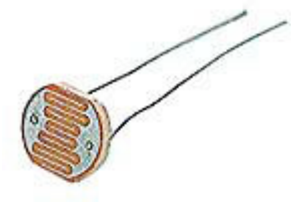

Figure 8: CDS Sensor

Sometimes referred to as photoresistors, CDS cells are used to measure the intensity of light. Light striking the surface of the photocell causes a decrease in resistance, while darkness produces a higher resistance. CDS photocells are best when used indoors, but they do have applications in extreme environments like bright sunlight or total darkness. In direct sunlight the resistance is very low. In total darkness the resistance is very high.

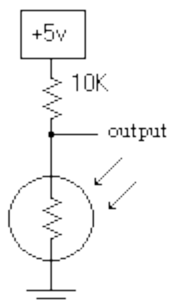

Figure 9: CDS Sensor voltage divider circuit

As shown above, a voltage divider is used on each CDS cell to create an output voltage which characterizes the light intensity at a given spot. This output voltage is attached to an A/D port on the Mavric ib board as with the IR sensors. Because each of the CDS cells used has an average resistance of 10k in what is deemed as average light intensity, a 10K resistor was chosen as the other portion for the voltage divider. This way, we can get a full voltage swing from 0V to 5V. Of note, because the robot only moves to the darkest place, no calibration is needed. Simple comparison of the 3 CDS sensors provides enough data for moving in the darkest direction regardless of environment. Which ever sensor gives the highest voltage(lower ADC value) is considered the darkest portion of the room.

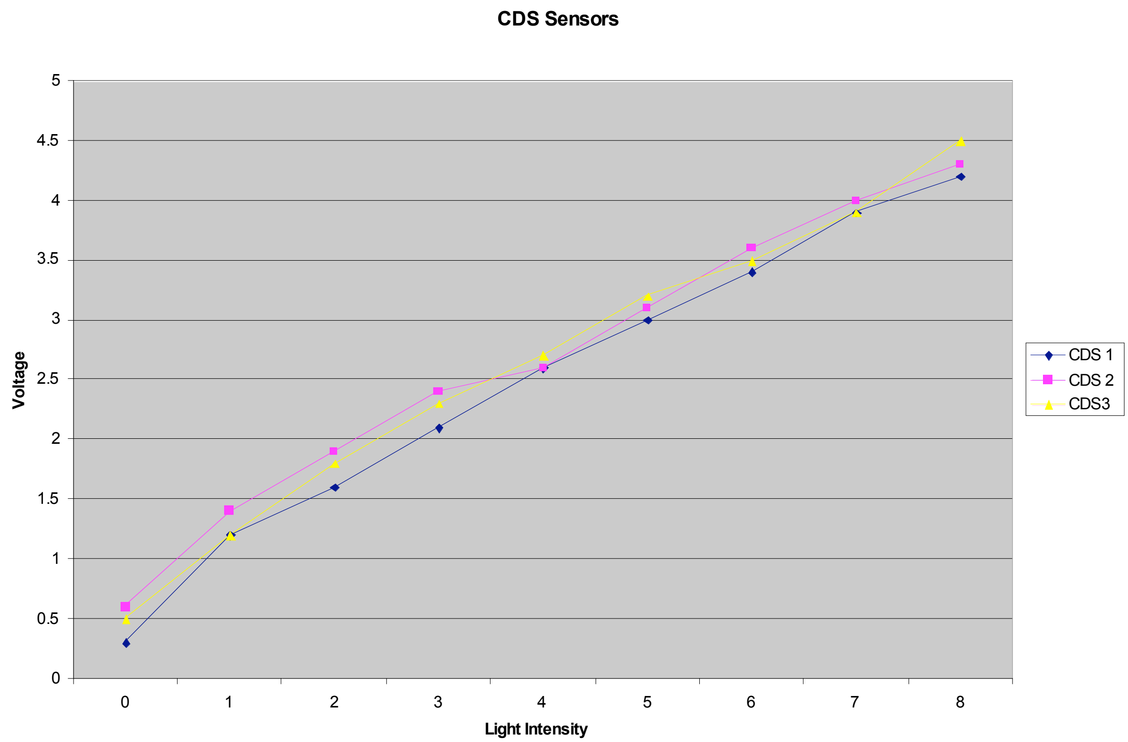

Figure 10: CDS Sensor characterization

Figure 10 data was obtained by placing the CDS cells 3 feet away from a light source controlled by a dimming switch. From on to off is 180 degrees. I tried as best I could to accurately divide this into 8 segments to define as going from full darkness to full brightness. All three CDS sensors were then used to measure voltage from the corresponding A/D ports. As can be seen, the sensors for whatever they are worth, are consistent within themselves which is exactly what I need for my comparison of darkness in the three directions each sensor points to.

RF Transmitter/Receiver

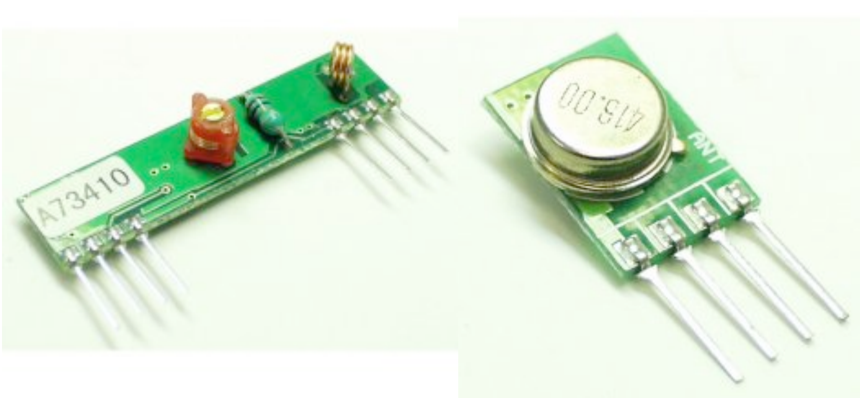

Figure 11: RF Receiver and Transmitter circuitry

Alarm-o-bot uses RF transceivers to communicate between the moving portion and the still portion that sits near your bedside. The transceivers operate at 433 MHz send data serially through TX pins and receive data through RX pins. The data is sent by way of a simple encoding and a parity bit for error checking.

The wireless transmitter/receiver RF-KLP modules have up to 500 ft range in open space. The receiver is operated at 5V and the transmitter operates anywhere from 2-12V. The higher the voltage, the greater the range. In this particular application, 5V is used to power the transmitter just because that is the easiest voltage available straight from the board itself.

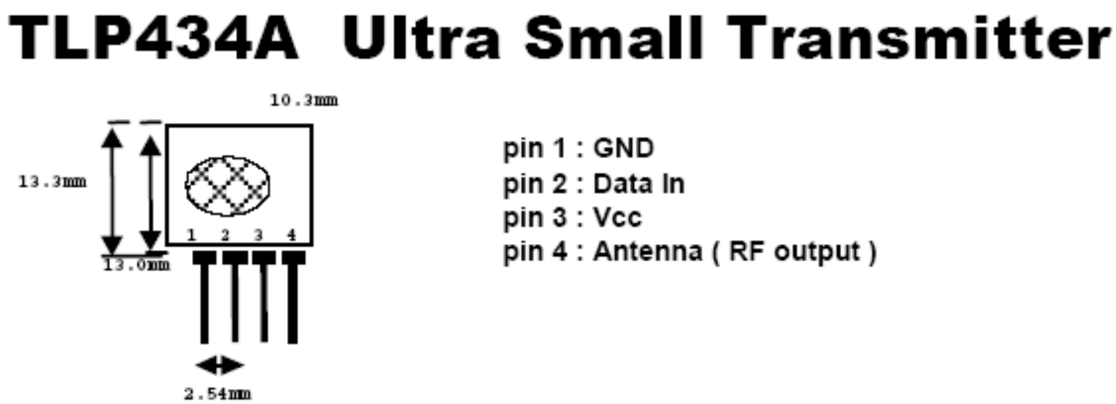

Figure 12: Transmitter dimensions and pinout

What the transmitter 'sees' on its data pin is what the receiver outputs on its data pin. Some configuration is required in the UART module on a Mavric ib, before a wireless data connection can be made. Data rates are limited to 2400bps. There are 433.92Mhz, 418Mhz and 315Mhz versions available, of which the 433.92MHz and 315MHz ones are used. Two frequencies of operation were choosen so that you can constantly transmit and receive data simultaneously.

This ASK transmitter module with an output of up to 8mW depending on power supply voltage. This receiver has a sensitivity of 3uV. It operates from 4.5 to 5.5 volts-DC and has both linear and digital outputs. The typical sensitivity is -103dbm and the typical current consumption is 3.5mA for 5V operation voltage.

Because there is so much noise in the environment, an initial sequence will be sent before any data is sent. This way, the microcontroller knows that it is receiving data and there is no false reads.

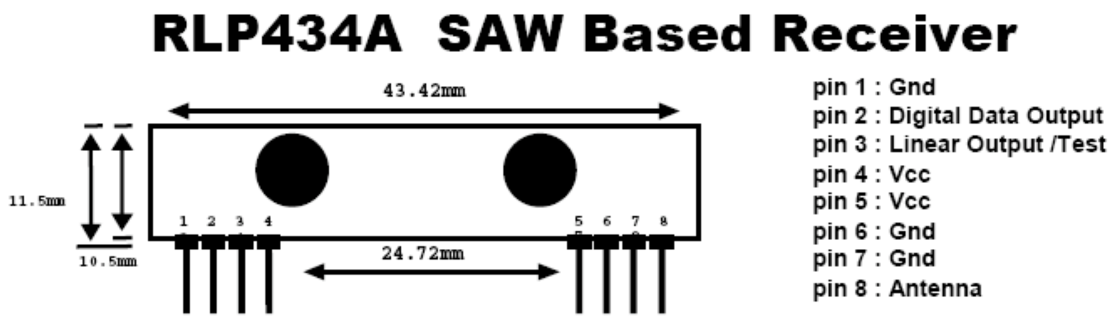

Figure 13: Receiver dimensions and pinout

The receiver is able to data from pin 8 (the input) and output it as digital data. For Alarm-o-bot, there is the main Mavric ib board plus an additional Atmega8 based board mr8. Each microcontroller has a receiver and transmitter operating at different frequencies. This allows for simultaneous communication between the boards (unlike the transceiver approach).

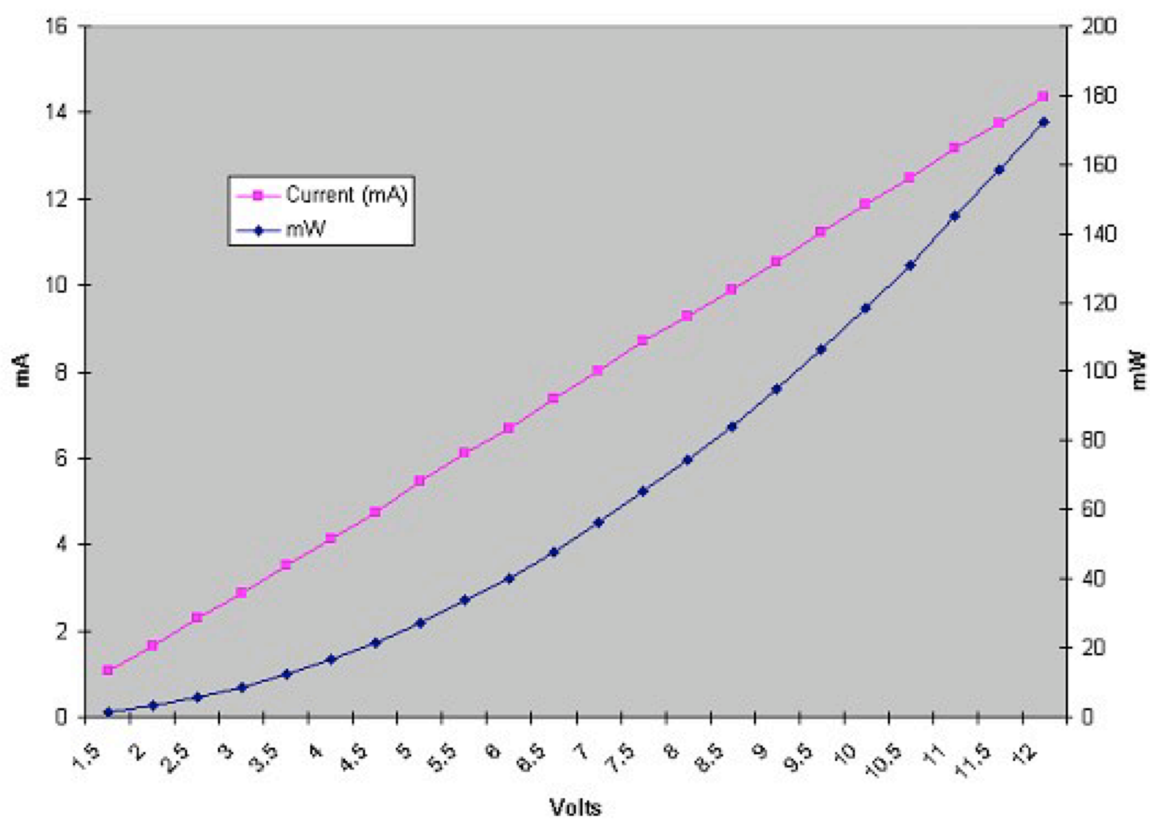

Figure 14: Transmitter power curve

The transmitter power curve is shown below with the input voltage ranging from 1.5V to 12V. For the transmitter, a 9 inch wire in antenna will be used as suggested by the datasheet.

Bump Sensor

Last and certainly least, is the bump sensor that I will be using. Unlike many robots, the bump sensor I am using is used as a button. In particular is the snooze button that must be hit for the alarm clock to shut off. Any push on the button changes the value on the A/D pin and signifies that the snooze is hit.

Figure 15: Bump Switch

Behaviors

1. When the alarm clock goes off, you must hit the snooze button on Alarm-o-bot.
2. After the first snooze (and each successive snooze), Alarm-o-bot hides itself based on low light intensity
3. The snooze button can only be hit on Alarm-o-bot (on/off switch remains on alarm clock)

Experimental Layout and Results

IR Sensors:
Scope: Testing Functionality
Specifications: None
Objectives: Obtain Data and/or Info on IR Sensors
Data:

IR1 (Voltage)	IR2 (Voltage)	Distance (cm)
0	0	0
0.7	0.6	1
1.3	1.2	4
2	2.1	9
2.4	2.4	10
2.2	2.3	11
2.1	2.2	11
1.9	2.1	12
1.6	1.7	14

1.4	1.4	15
1.3	1.2	20
1	1	27
0.8	0.9	30
0.6	0.8	34
0.6	0.7	40
0.4	0.6	47
0.3	0.5	58
0.2	0.4	74

CDS Sensors:

Scope: Testing Functionality

Specifications: None

Objectives: Obtain Data and/or Info on CDS Sensors

Data:

Light Intensity	CDS1(Voltage)	CDS2(Voltage)	CDS3(Voltage)
0	0.3	0.6	0.5
1	1.2	1.4	1.2
2	1.6	1.9	1.8
3	2.1	2.4	2.3
4	2.6	2.6	2.7
5	3	3.1	3.2
6	3.4	3.6	3.5
7	3.9	4	3.9
8	4.2	4.3	4.5

Problems

Sometimes light intensity variation is minimal preventing a true hiding function. Also, the alarm clock alarm is very soft and lacks any functionality.

Conclusion

Alarm-o-bot achieved most goals set forth and provided experience in building circuits/integrating electronics. For future considerations I want to be able to increase hiding parameters, add random hide function (so doesn't hide in same place), stronger platform. Alarm-o-bot avoids obstacles very well. The snooze button works well (RF signals communicate effectively through most barriers).

References

Code Credits:

www.bdmicro.com/code/

Steven Pickles