University of Florida

Department of Electrical and Computer Engineering

EEL5666

Intelligence Machines Design Laboratory

Spring 2005

Written Report I

Instructor: Dr. Arroyo

Student: Jose D. Garcia

Date: 04/19/2005

Table of Contents

Abstract

Dr. J is an autonomous robot that simulates the actions of a basketball player. Dr. J wanders around in search of a basketball while avoiding any obstacles it may encounter. When Dr. J finds and locates the basketball it pulls up within close proximity to the ball and awaits a pass from someone. Once it receives the basketball it shoots the ball into the hoop scoring a point for its team. Dr. J also has a bump sensor that senses when the catapult is pushed down and locks it into place.

Executive Summary

Dr. J is an autonomous basketball player robot. It is based on an MAVERIC-IIB board with an Atmega128 processor. Dr. J will search and find the basketball with a CMUCAM camera. When it finds the ball, it will wait for a pass from its teammate (me). Dr J will sense the basketball when the catapult is down. When the ball is provided, Dr. J will use the SFR08 ultrasound sensor to measure the distance from its position to the basket. When the precise distance is reached, Dr. J will release the catapult resulting in shooting the ball into the hoop.

Introduction

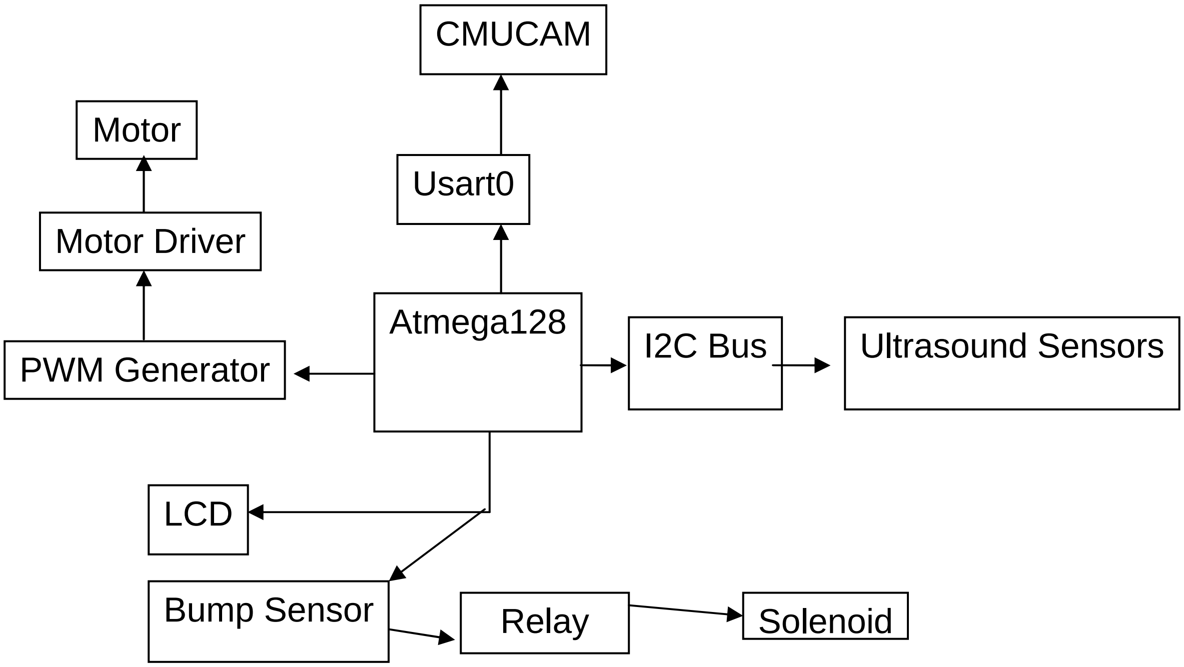
This report is a detailed description of my robot project "Dr J." Dr J will shoot a basketball into a basket from a certain distance. Dr. J will have sensors to measure the distance from its position to the actual basket. Also, Dr. J will have Ultrasound sensors that will allow it to avoid any obstacles that may prevent it from shooting the ball successfully into the basket. In addition, Dr J will search and find the basketball instead of being provided with one; for this task I will use a CMUCAM. This paper will present some approaches done by other students as well as my own approaches and improvements on the methods that have been previously carried out.

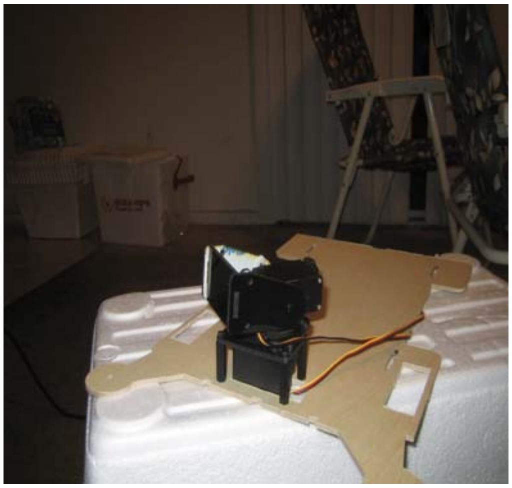
Integrating System

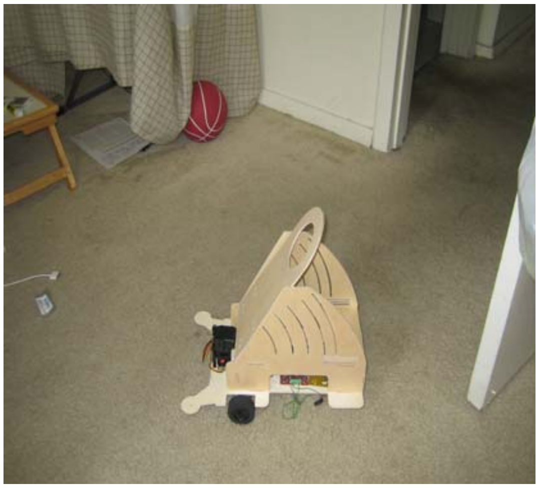
This section gives a description of the interfaced components of Dr J:

Dr. J will have different components to implement the following tasks:

- CMUCAM - a CMUCAM is used to track the color orange to find the basketball. The CMUCAM is going to be mounted in a Pan and Tilt to look around.
- Sonar Range Finder - two Daventech SFR08 ultrasound sensors are used to measure the distance from the actual robot to the basket to shoot the ball successfully.
- Shooting Mechanism – the shooting mechanism consists of two coils holding the catapult and the actual platform. When the catapult is pulled back, a bump sensor detects it using an external interrupt and it will lock the catapult with a pull solenoid.

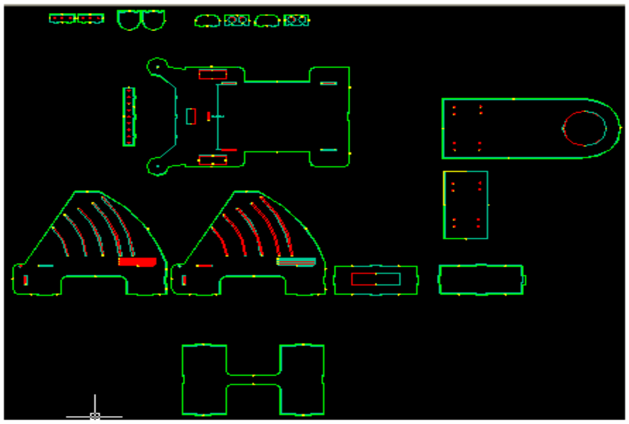






Mobile Platform

The platform consists of a base to hold all the components of the system, such as the microcontroller, power board, motor driver, etc; two side panels to hold the catapult; two holders for the ultrasound sensor and a square 1” by 3/4” inches to hold the CMUCAM to the Pan and Tilt.



Actuation

I used two Planetary Gear Motors for my robot.

Description:

Voltage = 12vdc

RPM = 148

Reduction = 27:1

Stall Torque = 73.13 oz-in (5.27 kg-cm)

Motor Type = 28mm OD RS-380

I decided to use DC motors over servos because motors have a longer life than servos and you need to hack them.

Sensors

CMUCAM

The special sensor of my robot is the CMUCAM developed by Carnegie-Mellon University. This camera is a low cost and low power device which is perfect for robotics. The CMUCAM is a SX28 microcontroller interfaced with an OV6620 Omnivision CMOS camera which it communicates with a microprocessor via a RS232 or a TTL serial port. This camera consists of many features such as, getting the mean color of an image, giving the center of an object, and automatic tracking of a color and it also drives a servo to track an object.

The camera by default reads in ASCII commands. Each command or instruction should be followed by a \r (ASCII 13). The camera sends back ACK\r or NCK\r depending if the instruction was received correctly or not. You can suppress this feature by writing a 1 to the second bit of the Raw Mode command. After each command is executed and the camera is ready for another command, it outputs a ":" character. Also, depending on the command sent to the camera and/or mode it is in, one of six different types of output packets is returned; with the exception of the DF command, which outputs data column by column. Both the transmit and receive lines have to be inverted so that the transmit line from the camera goes to the receive line on the microprocessor and vice versa.

Devantech SRF08 Range Finder

The communication with the SRF08 is via the I^{2}C Bus. The default address of the SRF08 is 0xE0. However, this address can be changed by the user to any of its addresses in memory (E0, E2, E4, E6, E8, EA, EC, EE, F0, F2, F4, F6, F8, FA, FC or FE). Therefore, one can use up to 16 sonars connected to the same I^{2}C Bus.

Registers

The SFR08 has 36 registers (from location 0 to 35). Only location 0, 1, 2 can be written to. Location 0 is the command register and Location 1 is the onboard light sensor. In addition, Location 2 and 3 are a 16-bit unsigned value that results from the latest ping where location 2 is the high-byte and location 3 is the low-byte. A value of zero indicates that no objects were found in its path.

Commands

There are three commands to initiate a ping (80 to 82); to return the results in inches, one have to write 0x50 to location 0, to return the results in centimeters, one have to write 0x51 to location 0 and to return the results in micro-seconds, one have to write 0x52 to location 0.

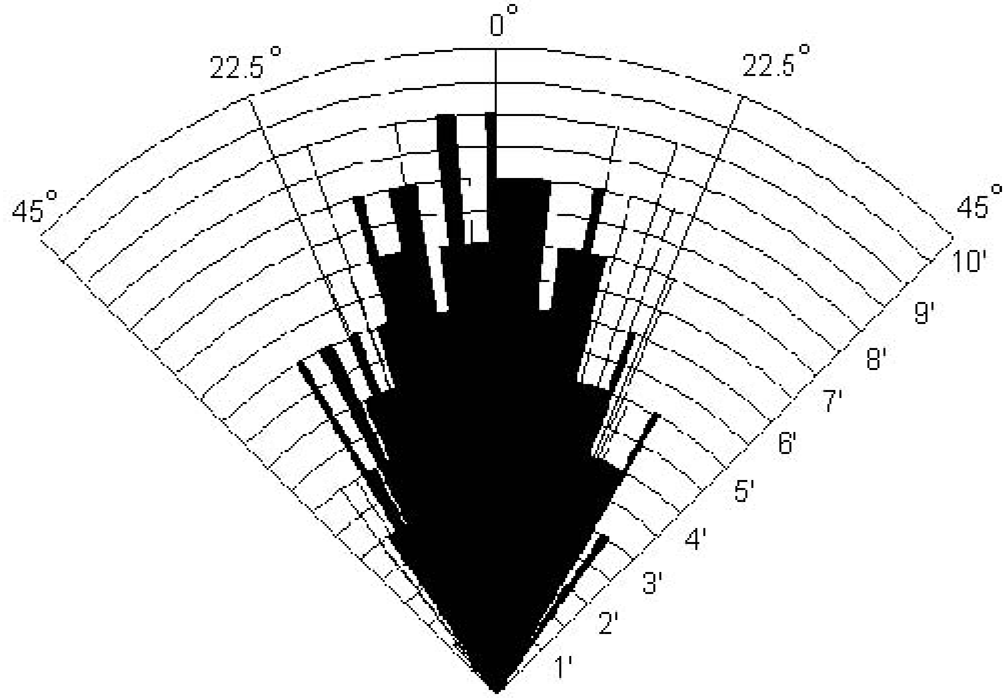
Changing the I^{2}C Bus Address

To change the address of I^{2}C of the SRF08, one must have only one sonar connected to the I^{2}C Bus. To change this address write the sequence commands (0xA0, 0xAA, 0xA5, “new address in hex”) to location 0. These commands must be sent in four separate write transaction in the correct sequence.

Theory

An ultrasound ping is released by the sensor and it determines the distance based on the time this ping takes to return back to the module. This distance is calculated given the speed of sound at sea level; the ultrasound sensor is accurate to about 0.5 inches. It ranges between 3 centimeters and 6 meters.

Beam Pattern



Behaviors

Dr. J has the following behaviors:

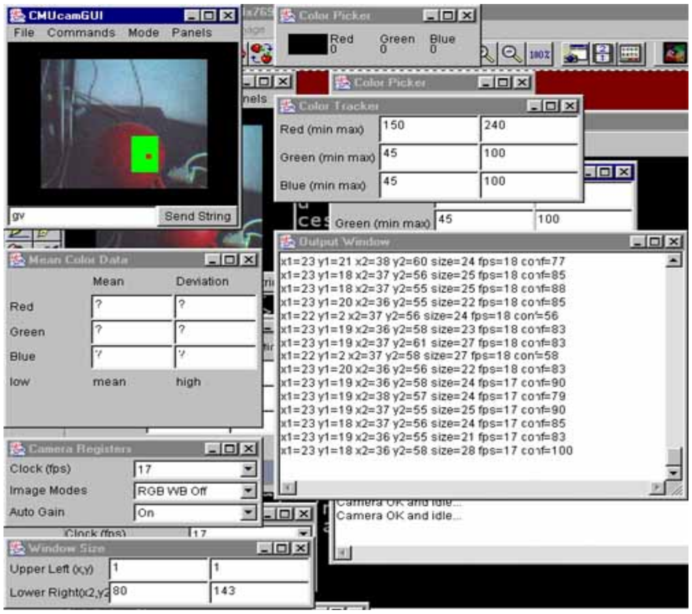
- When Dr. J boots up, it initializes all modules in the system.
- When it does not have the ball, it wanders around looking for the ball.
- When it finds the ball, Dr. J stays near the ball until the ball is provided.
- When it has the ball, it finds the basket and lines up with the hoop.
- When the robot finds the hoop, it will measure the distance and shoot the ball successfully in the hoop.

Experiments and Layouts

CMUCAM

The table below is data collected for tracking orange color from different distances with CMUCAM.

Inches Away	Confidence Value
3	236
6	212
10	192
12	178
16	156
18	143
20	136
24	2



gv

Red 0 Green 0 Blue 0

Red (min max)	150	240
Green (min max)	45	100
Blue (min max)	45	100

	Mean	Deviation
Red	?	?
Green	?	?
Blue	?	?
low	mean	high

Clock (fps): 17

Image Modes: RGB WB Off

Auto Gain: On

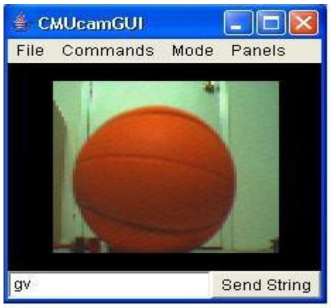
Upper Left (x,y)	1	1
Lower Right(x2,y2	80	143

```
x1=23 y1=21 x2=38 y2=60 size=24 fps=18 conf=77
x1=23 y1=18 x2=37 y2=56 size=25 fps=18 conf=85
x1=23 y1=18 x2=37 y2=55 size=25 fps=18 conf=88
x1=23 y1=20 x2=36 y2=55 size=22 fps=18 conf=85
x1=22 y1=2 x2=37 y2=56 size=24 fps=18 conf=56
x1=23 y1=19 x2=36 y2=58 size=23 fps=18 conf=83
x1=23 y1=19 x2=37 y2=61 size=27 fps=18 conf=83
x1=22 y1=2 x2=37 y2=58 size=27 fps=18 conf=58
x1=23 y1=20 x2=36 y2=56 size=22 fps=18 conf=83
x1=23 y1=19 x2=36 y2=58 size=24 fps=17 conf=90
x1=23 y1=19 x2=38 y2=57 size=24 fps=17 conf=79
x1=23 y1=19 x2=37 y2=55 size=25 fps=17 conf=90
x1=23 y1=18 x2=37 y2=56 size=24 fps=17 conf=85
x1=23 y1=19 x2=36 y2=55 size=21 fps=17 conf=83
x1=23 y1=18 x2=36 y2=58 size=28 fps=17 conf=100
```

```
Camera OK and idle...
Camera OK and idle...
```

The table below illustrates data collected for tracking the color orange from different distances with CMUCAM along with white LED's on object.

Inches Away	Confidence Value
3	254
6	251
9	242
12	236
24	230



gv

Below is a sample program to make LED blink at a baud rate of 115200 baud.

```
/***************************************************/
#include <avr/io.h>
#include <avr/interrupt.h>
#include <avr/signal.h>
#include <avr/twi.h>

#include <stdio.h>
#define BAUD_RR ((CPU_FREQ/(16L*9600L) - 1))
#define CPU_FREQ 16000000L  /* set to clock frequency in Hz */

#if CPU_FREQ == 16000000L
#define OCR_1MS 125
#elif CPU_FREQ == 14745600L
#define OCR_1MS 115
#endif



volatile uint16_t ms_count;

/*
 * ms_sleep() - delay for specified number of milliseconds
 */
void ms_sleep(uint16_t ms)
{
  TCNT0  = 0;
  ms_count = 0;
  while (ms_count != ms)
    ;
}
```

```
/*
 * millisecond counter interrupt vector
 */
SIGNAL(SIG_OUTPUT_COMPARE0)
{
  ms_count++;
}
/*
 * Initialize timer0 to use the main crystal clock and the output
 * compare interrupt feature to generate an interrupt approximately
 * once per millisecond to use as a general purpose time base.
 */
void init_timer0(void)
{
  TCCR0 = 0;
  TIFR  |= BV(OCIE0)|BV(TOIE0);
  TIMSK |= BV(TOIE0)|BV(OCIE0);        /* enable output compare interrupt */
  TCCR0  = BV(WGM01)|BV(CS02)|BV(CS00); /* CTC, prescale = 128 */
  TCNT0  = 0;
  OCR0   = OCR_1MS;                    /* match in aprox 1 ms */
}
int def_putc(char ch)
{
  /* output character to UART0 */
      while ((UCSR0A & BV(UDRE)) == 0);
      UDR0 = ch;
      return ch;
}
int main(void)
{
  init_timer0();
```

```
/* enable UART0 */
UBRR0H = (BAUD_RR >> 8) & 0xff;
UBRR0L = BAUD_RR & 0xff;
UCSR0B = BV(TXEN);    /* enable transmitter */

/* enable interrupts */
sei();

/* initialize stdio */
fdevopen(def_putc, NULL, 0);

/* set the I2C bit rate generator to 100 kb/s */
TWSR &= ~0x03;
TWBR  = 28;
TWCR |= BV(TWEN);

/* initiaze PORTA*/
DDRA = 0xff;

/* initialize LCD */
LCD_initialize();

Send_SentenceToLCD("I Love Vannessa");

while (1) {

          ms_sleep(1500);
  printf("L1 1\r");
          ms_sleep(1500);
```

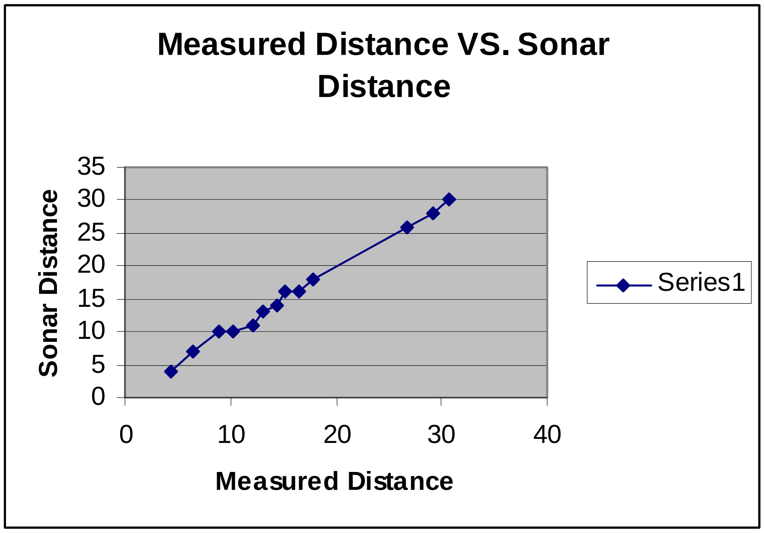
```
            printf("L1 0\r");
  }

}
/**********************************************************************/
```

Ultrasound Sensor

Experiment 1: Detection of an object with one single ultrasound sensor

Distance Measured in Inches	**Distance Measured in Centimeters**	**Distance Obtained from the Sonar**	**% Error**
1.75	4.4	4	9.100%
2.50	6.4	7	9.375%
3.50	8.9	10	12.360%
4	10.2	10	1.961%
4.75	12.1	11	9.100%
5.1	13	13	0%
5.65	14.4	14	2.780%
6	15.2	16	5.263%
6.50	16.5	16	3.125%
7	17.8	18	1.124%
10.5	26.7	26	2.62
11.5	29.2	28	4.120%
12.1	30.7	30	2.280%
12.7	32.2	30	6.90%



Note: it is a linear graph. Therefore, the sonar was giving accurate data.

Experiment 2: Detection of an object with two ultrasound sensor turned at a small angle (20 degrees) and separated about 8 centimeters. The distance measured is the distance from the center point between the two sensors.

Distance Measured in Inches	**Distance Measured in Centimeters**	**Distance Obtained from Sonar 1**	**Distance Obtained from Sonar 2**
1.75	4.4	4	3
2.50	6.4	7	6
3.50	8.9	10	10
4	10.2	10	10
4.75	12.1	11	16

5.1	13	13	11
5.65	14.4	14	12
6	15.2	16	17
6.50	16.5	16	18
7	17.8	18	18
10.5	26.7	26	22
11.5	29.2	28	25
12.1	30.7	30	28
12.7	32.2	30	29

Conclusion

My overall experience making this robot has been a great learning opportunity. I made several mistakes and was able to correct some of them. If I were to design and build this robot again I would have built the platform very early in the semester because that was the most time consuming portion of the robot. As far as enhancements go, I would have liked to construct an arm for Dr. J to be able to pick up and place the basketball in its catapult. Overall, it was a great learning experience and this project has helped me further realize the importance of good time management.

Documentation

CMUcam Vision Board User Manual.

http://www.seattlerobotics.com/cmucam.htm

Motor Driver User Manual

http://www.acroname.com/robotics/parts/R84-HS-322HD.html

Appendix A

Program Code: Please see attached file.

Vendor Information:

Devantech SRF08 Ultrasound Sensor

Acroname Inc.
4894 Sterling Drive
Boulder , CO 80301
www.acroname.com
Part #:R145-SRF08
$59.50 each

CMUCAM

Acroname Inc.
4894 Sterling Drive
Boulder , CO 80301
www.acroname.com
Part #:R140-CMUCAM-KIT
$109.0 each

Bump Switches

(Purchased from
Tsang-Wei (Dan)
Huang)
$3 for two.