University of Florida

Department of Electrical and Computer Engineering

EEL5666

Intelligence Machines Design Laboratory

Spring 2005

Sensor Report

Instructor: Dr. Arroyo

Student: Jose D. Garcia

Date: 04/19/2005

Abstract

The purpose of this report is to present a detailed description of my sensors utilized in my robot project as well as testing data results collected from experiments.

Introduction

One of the most important factors of this project is to accurately measure distance. In order to be able to shoot the basketball successfully into the hoop, my robot has to measure the distance from its position to the basket and to align precisely with the hoop. To carry this task I decided to use the Devantech SRF08 Ultrasonic Range Finder. The SRF08 is a high performance sensor that uses sonar to detect objects. I chose the Ultrasound sensors because it is the most accurate measuring device to measure distance; it is much more accurate and it has twice the range (3 centimeters to 6 meters) of the Infrared Range(IR) sensor.

Also, my robot will search and find the basketball instead of being provided with one; for this task I will use a CMUCAM. The CMUCAM can be used to track or monitor colors. In my project, I will use the camera to track the color “Orange”. However, one of the most common problems of using the CMUCAM is that the camera might see the color of an object differently in different light conditions. In addition, to avoid that my robot gets out of bound, I use photocells to track dark lines (side lines) on court.

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 includes has many features such as it gets the mean color of an image, it gives the center of an object, and it can automatically track a color and driver 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.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.

EXPERIMENTS

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 */
```

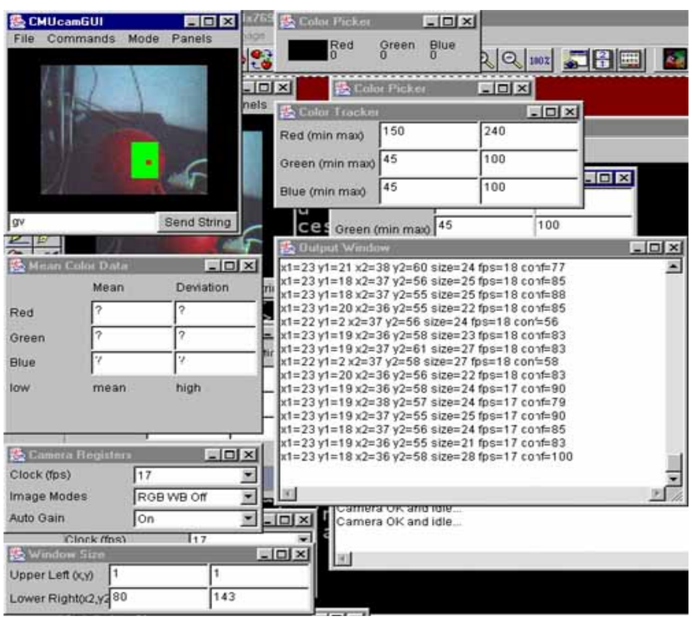
```
#if CPU_FREQ == 16000000L
#define OCR_1MS 125
#elif CPU_FREQ == 14745600L
#define OCR_1MS 115
#endif



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");
  }

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

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



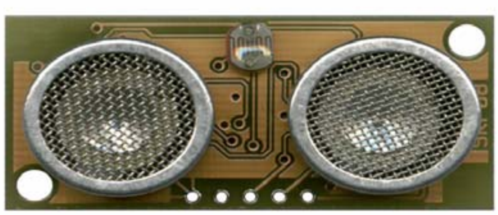
The below table is data collected for tracking orange color from different distances with CMUCAM along with white LED’s on object.

The table below is data collected for tracking orange color 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

Devantech SRF08 Range Finder

The communication with the SRF08 is via the I^2C 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 sonar's connected to the same I^2C 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.

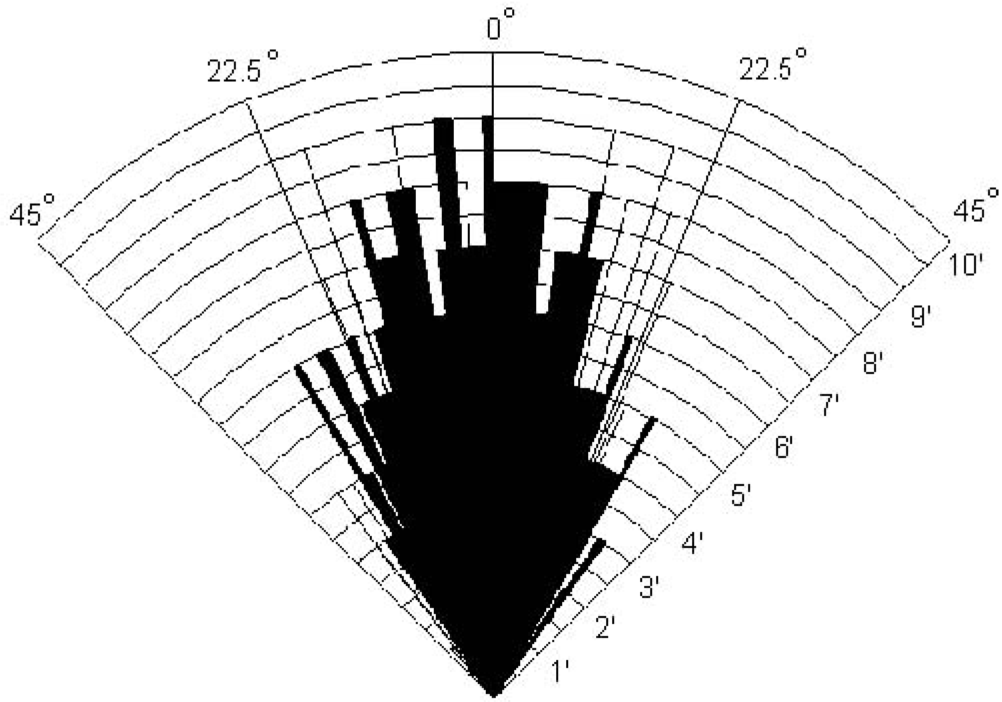
Changing the I^2C Bus Address

To change the address of I^2C of the SRF08, one must have only one sonar connected to the I^2C 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 inch . It ranges between 3 centimeters and 6 meters.

Beam Pattern

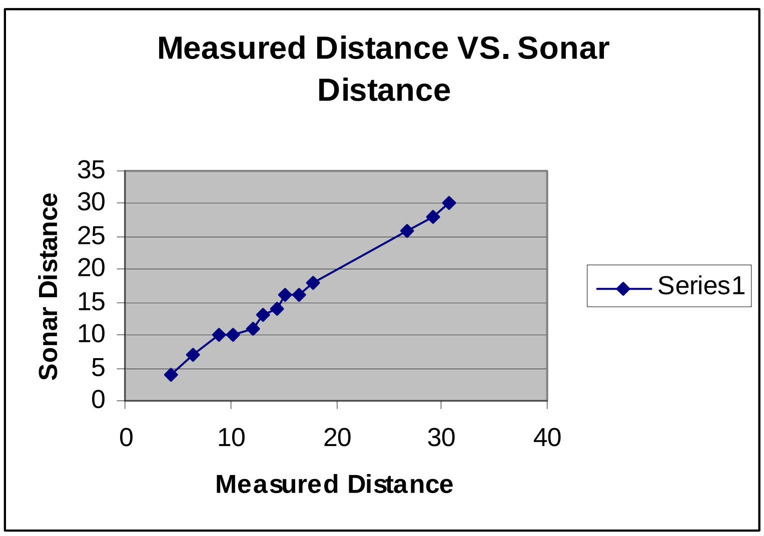


Experiment and Results

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 an small angle (20 degress) 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

The CMUCAM is a very versatile for simple task of tracking and sorting objects of different colors. I was successful in using the Track Color command to center in on a color with middle mass coordinate mx1 and my2. the SRF08 Ultrasound sensor is the best choice to measure distance and implement obstacle avoidance. Also, I utilize these sensors to align the robot with the basket. I believe the Ultrasound sensor can provide Dr.

J with the accurate distance measurements to do obstacle avoidance and aligning itself with the basket.

Appendix A:
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.