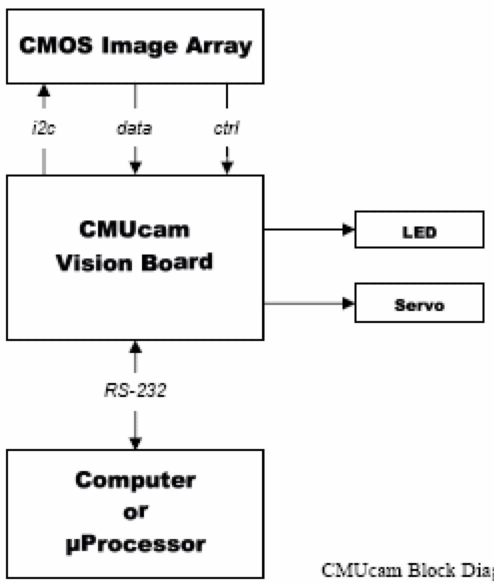
EEL5666
Kari Bowen
Dr. Arroyo
3/17/2005
Nerfherder Special Sensor Report

Table of Contents

CMUCam

Theory

The CMUCam is a simple camera with microprocessor which was created at Carnegie Mellon University. It has built in functions such as tracking an object of a certain color, finding the average color of a frame. It utilizes the SX28 microcontroller and an OV6620 Omnivision CMOS camera. It communicates in RS232 or TTL serial ports.



CMUcam Block Diagram

Objective

For Nerfherder, the CMUCam will be used to find the red target. Once it identifies the red target, it will calculate the distance from the target. As the robot moves, the CMUCam will continue to tell the distance from the target, until the robot determines it is close enough. The robot will then launch its projectile.

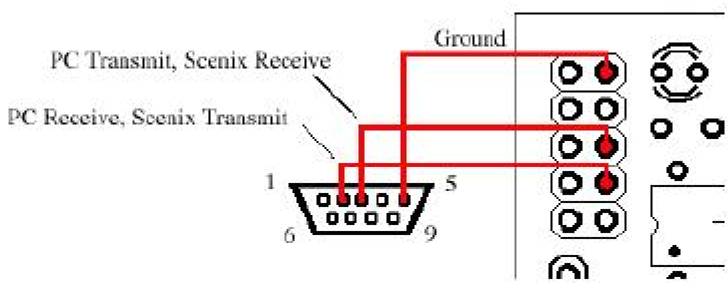
References

Vendor Name: Acroname
Part number: R140-CMUCAM-KIT
Price: $109
Contact Phone: 720.564.0373
Website: www.acroname.com
Email: info@acroname.com
Product manual: http://www-2.cs.cmu.edu/~cmucam/

Integration

The CMUCam will communicate with my MAVRIC-II board via RS232 communication. For initial testing purposes, it can also interface with a PC using the serial connection and a terminal program such as hyperterminal. Upon connection to the microprocessor, the commands and responses can be sent and received fairly easily.

The CMUCam uses only 3 pins on the serial connector: a ground pin, and two transmit/receive lines. See below. The serial connection can operate at 4 different baud rates: 115,200; 38,400; 19,200; and 9600 baud. It uses 8 bits with 1 stop bit and no parity or flow control.

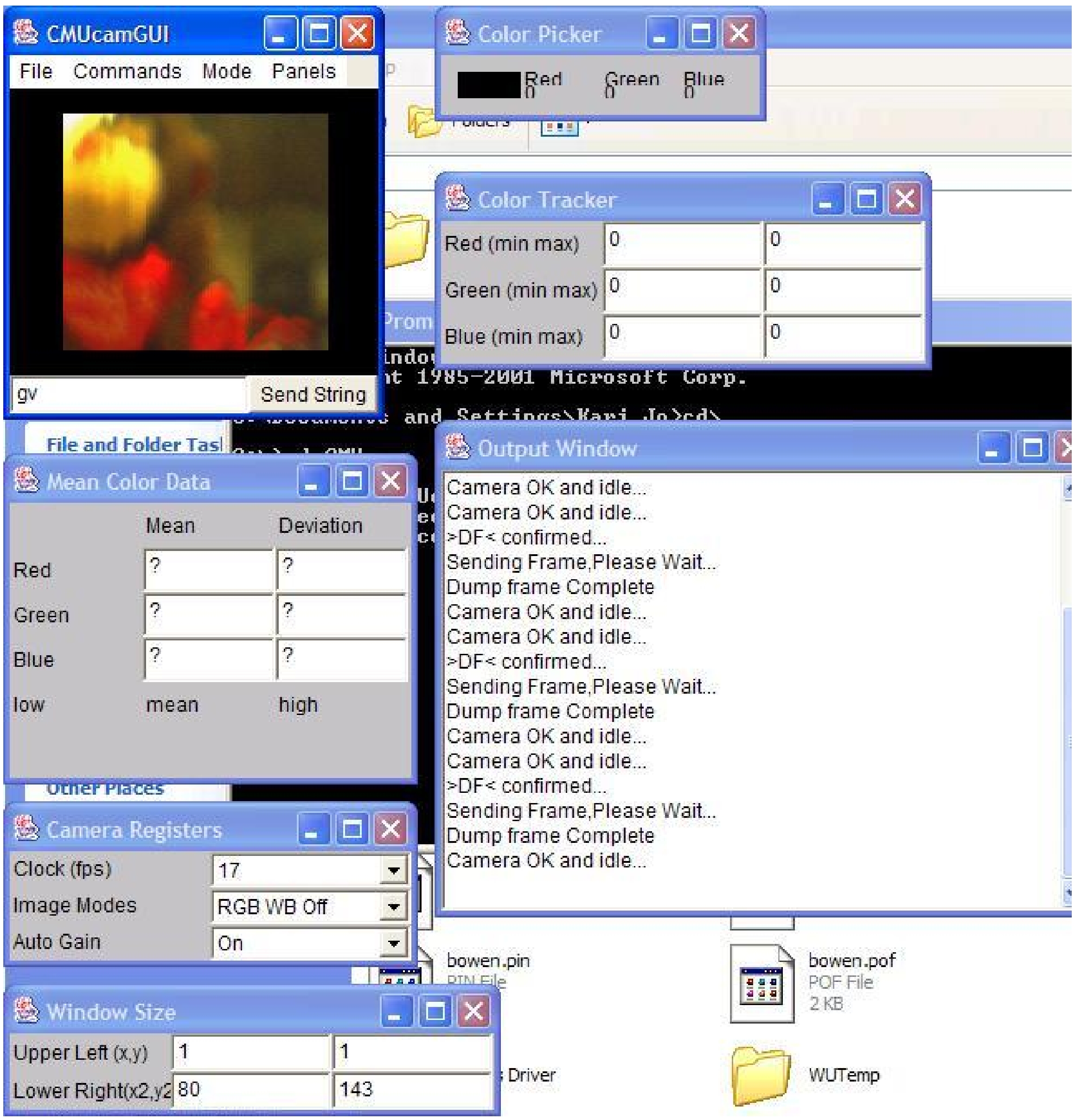


CMUCam Commands

Command	Action
/r	Set board into idle state
CR[reg1 value 1 [reg2 value2 … reg16 value16]]\r	Set internal register values
DF\r	dump frame onto serial port
DM value\r	sets delay before transmission
GM\r	get mean color value
GV\r	get current version of firmware
HM active\r	puts into half-horizontal resolution mode
II \r	uses servo port as digital input
L1 value\r	control the tracking light
LM active\r	turns on line mode
MM mode\r	sets middle mass mode
NF active\r	noise filter setting
PM mode\r	poll mode
RM bit_flags\r	raw serial transfer mode
RS \r	Reset
S1 position \r	set position of servo 1
SM value \r	enable switching mode of color tracking
SW [x y x2 y2] \r	set window size of camera
TC [Rmin Rmax Gmin Gmax Bmin Bmax]\r	track a color
TW \r	track the color in the central region of the currect window

Experimental Data

I have tested the CMUCam using the CMUCamGUI provided by the Carnegie Mellon website. The camera can capture images and dump frames. Commands can be used to find the colors at certain positions. The cam correctly identifies red, blue and green. See below screen shot for example of a dumped frame.



The camera is more sensitive to the color red than any other color. I found that usually a value of 120 or greater signaled a significant amount of red, whereas 100 or more would be a significant amount of blue or green.

Lessons Learned

One major lesson learned is to not expect a sensor to work right out of the box. I dragged my feet to finish soldering this kit together, and I have suffered for it! I had trouble making the serial cable as it was a snap-on device. I did not put the cable far enough into the end piece, so I was missing a connection on one of the wires. Upon trying to fix this, I broke part of the snap-on piece.

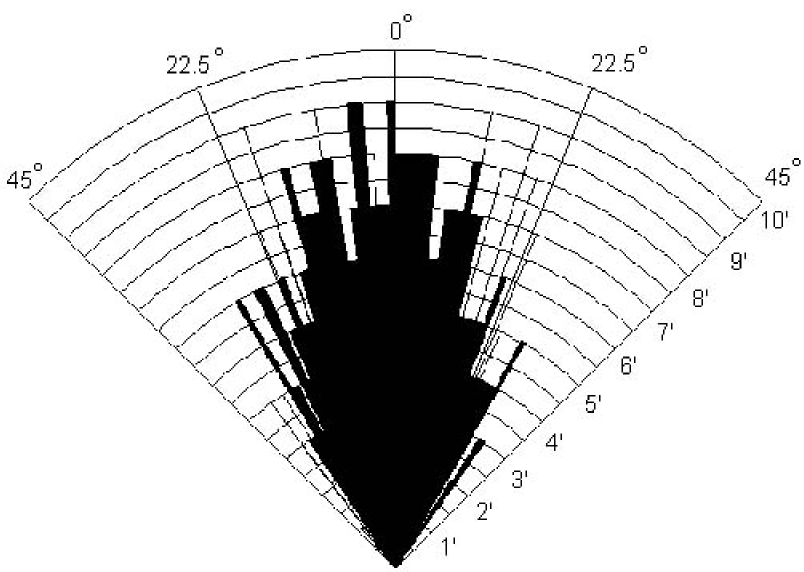
Once I fixed the cable, I plugged the serial connector into my laptop. I then turned on the camera and expected to see "CmuCam v. 1.12." Instead, I got garbage. After a few hours of attempting to debug this, I realized that the supply voltage for the camera should be 6-9v, not 5v. Even after fixing that, I still am not getting exactly the right output from the camera. This will, I'm sure, lead to many more hours of debugging.

Sensor Suite

Devantech SRF04 Sonar

Theory

The SRF04 sensor works on the principle of sonar. The sensor sends out a ping and waits for a response. The time it takes for the ping to return is an indication of how far away an object is. The SRF04 in particular has a range of 3cm to 3 m. The angles at which the SRF04 pings are shown below.



Objective

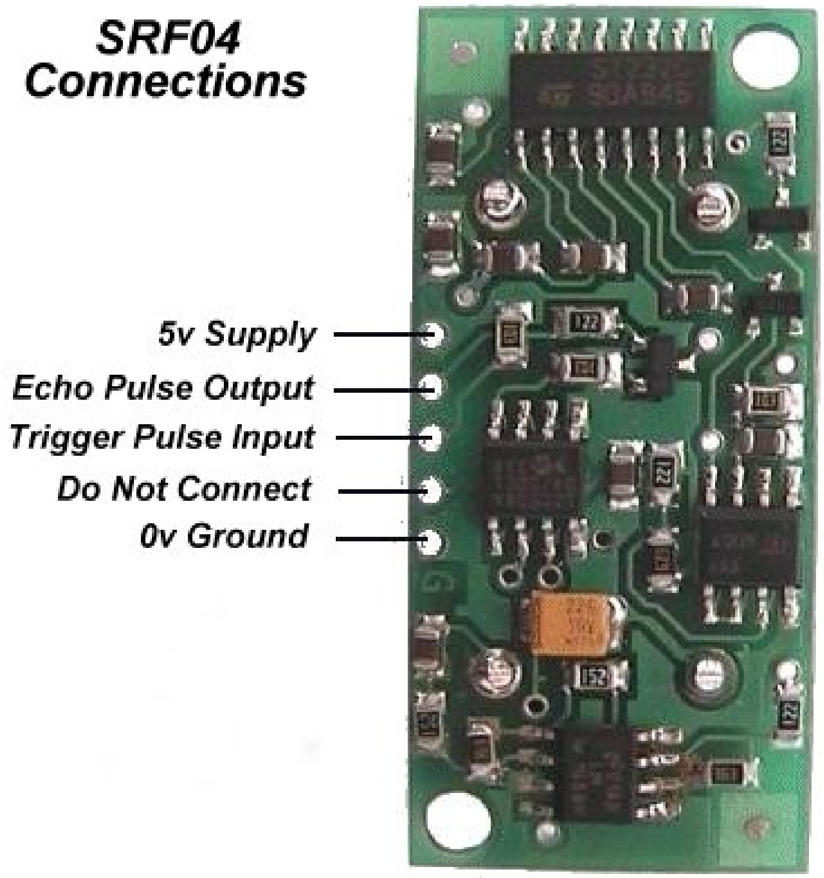
The SRF04 will be used on Nerfherder for obstacle avoidance. The sensors will ping continuously to determine how far away from any walls or objects Nerfherder is. If Nerfherder gets too close to an object (within some threshold that I will define), an obstacle avoidance routine will run in which Nerfherder will turn away from the obstacle(s) and then continue what is was doing.

References

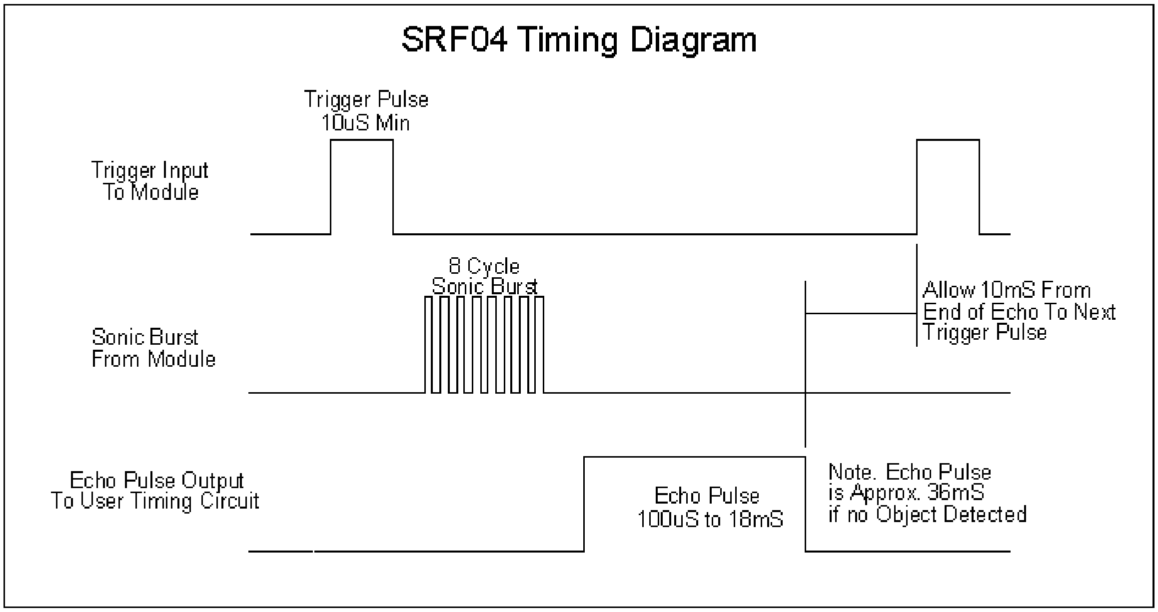
Vendor Name: Acroname
Part number: R93-SRF04
Price: $36 (however, I got them for free from a previous IMDL student).
Contact Phone: 720.564.0373
Website: www.acroname.com
Email: info@acroname.com

Integration

The sonar will be interfaced to the microprocessor via a digital port (for me, Port E and on the MAVRIC-II). The connections for the sensor are shown below.



The microprocessor will send out a pulse on the "trigger pulse input" line. This will enable the SRF04 to begin pinging. The microprocessor will then monitor the echo pulse output line to find when it goes low. A timer running will then determine what the distance is to the nearest object. The timing diagram is also shown below.

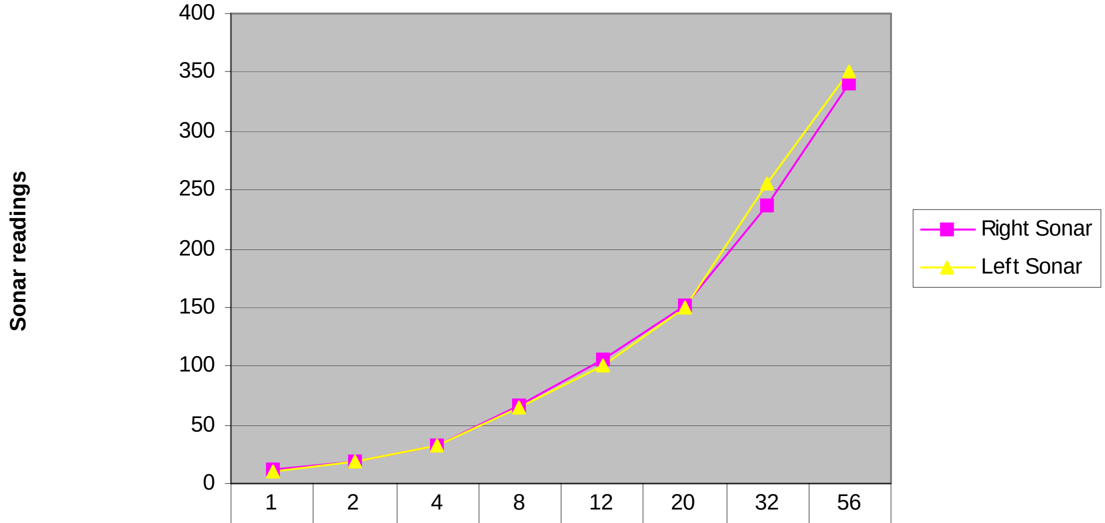


Experimental Results

The sonar performed at a roughly linear level at mid-range (1-4 feet). Above and below that threshold, there were slight variations. Data shown below.

Measured Distance (in inches)	Right Sonar	Left Sonar
1	12	10
2	18	18
4	32	32
8	66	65
12	105	100
20	152	150
32	237	256
56	341	351

Sonar Readings (Average)



	1	2	4	8	12	20	32	56
Right Sonar	12	18	32	66	105	152	237	341
Left Sonar	10	18	32	65	100	150	256	351

Distance in inches