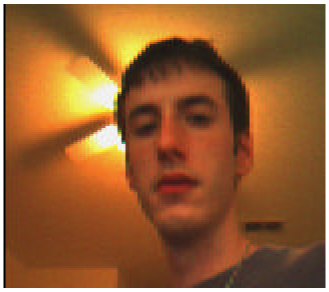
University of Florida
Department of Electrical and Computer Engineering
EEL 5666
Intelligent Machines Design Laboratory

Sensor Report

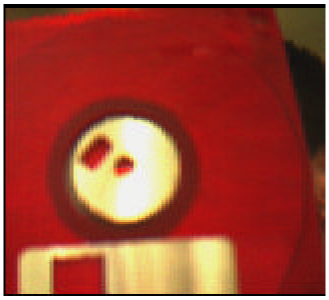
04/20/03
Jordan T. Wood

CMUcam Vision Camera

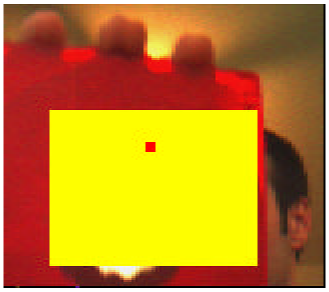
. The color tracking was done with a CMUcam digital camera. The CMUcam is a CMOS camera that is controlled with a SX-28 microcontroller. Images information is transmitted serially to the main microcontroller. The camera works by taking the RGB values of the tracked item. It then, determines the center of mass of the tacked item. The camera then sends mass Y, mass X, to the main microcontroller. The main microcontroller then uses this information to control the motor movement to keep on center of the tracked item. Sending command to the camera is done serially. This mean that you must you ASCII characters to tell camera what to do. For example to track a color, you send the command "TC" following by the color RGB min and max, "TC 20 50 20 50 10 90", will follow the color blue. The camera in return sends Y, X, coordinates. Since the camera viewing size is 80 X143 pixels I can deter mind if the color is center, left, or right by taking the x value. If it is 0 the it is left of center, if it is 80 then it is right of center, and if it is 40 then it is center. Here is some test images.



Picture of me taken by CMUcam



A Item to track



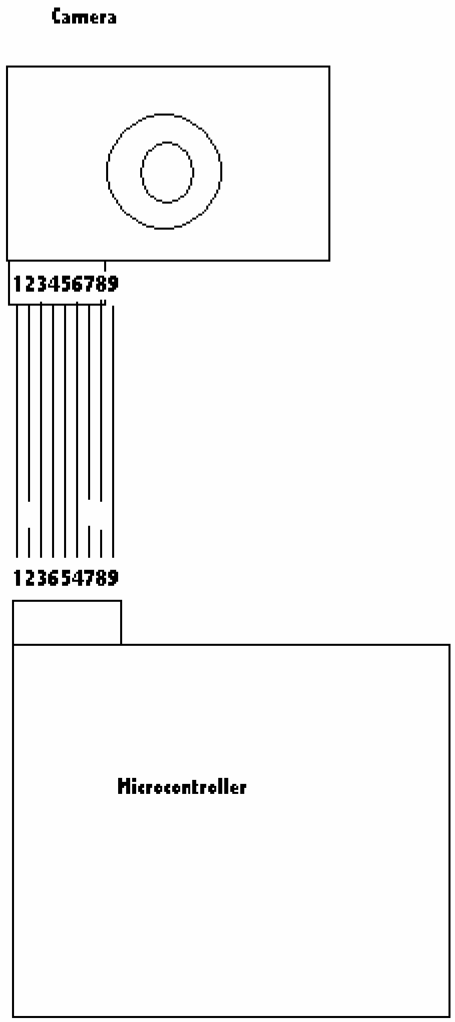
CMUcam Tracking the color

The camera was set a 38,400 baud rate, which give great refresh rate. If you use a Atmel MegaAVR 323 microcontroller there is some modification to connect the camera serially. On the Progressive board the TX pins and RX are need to be flipped. Also I had to cut unneeded pins 2, 6, and 7, because the UART would not set the RX pin high if I did not. When ordering a CMUcam, make sure that it has a IR coating on the lens, and that it has a wide view lens. If you do not want to build the camera order for Seattle Robotics (http://www.seattlerobotics.com/cmucam.htm.)

Here is example of some code.

```
#include <io.h>
#include <sig-avr.h>
#include <stdlib.h>
#include <interrupt.h>
#include <progmem.h>
typedef unsigned char u08;
typedef unsigned int u8;
typedef         char s08;
typedef unsigned short u16;
typedef         short s16;
#include "uart.h"
#include "delay.h"
volatile u8 temp;
volatile u8 i=0;
volatile u8 cmudat[9];

//Initialize UART
//'baud' is the baud register divider
void uartinit(void)
{
  /* Set baud rate */
  outp(0x98,UCSRB);
  outp(0x86,UCSRC);
  outp(0x00,UBRRH);
  outp(0x09,UBRRL);
  /* Enable Receiver and Transmitter */

        // 1 stop bit
```

```
}

//Send a single byte of data
//'data' is the byte sent
void uarttransmit(unsigned char data)
{
  /* Wait for empty transmit buffer */
  while (!(UCSRA & (1<<UDRE))){}
  /* Put data into buffer, sends the data */
  outp(data,UDR);
}



//Send a given EOS terminated string
void uartstring(unsigned char * myStringIn)
{
  unsigned char *myString = myStringIn;
  unsigned char ch1;
  unsigned char gotNULL = 0;
  ch1= *myString++;
  while(!gotNULL){
    uarttransmit(ch1);
    ch1 = *myString++;
    if(ch1 == '\r'){
      gotNULL = 1;
      uarttransmit(ch1);
      }

    }
}



SIGNAL(SIG_UART_RECV){
        temp=inp(UDR);
        if(temp != 0x3A){
                if(temp == 0x20 || i==9){
                i=0;
                }
                else if(i==0){
                        if(temp == 0xFF){
                                cmudat[i]=temp;
                                i++;
                        }
                }
                else if(temp !=0x20 && i<9){
                cmudat[i]=temp;
                i++;
                }
        }

}
/*void blink(void){
        uartstring("L1 1\r");
        delay(1500);
        delay(1500);
        delay(15000);
        uartstring("L1 0\r");
        delay(15000);
        delay(1500);
        delay(1500);

        }*/
int main(void){                                         //using the camera
        outp(0xFF,DDRC);
        delay(10000);
        uartinit();                                     //setup of UART
        delay(10000);
        uartstring("PM 1\r");
        delay(1000);
        uartstring("RM 3\r");
        delay(1000);
        sei();

        while(1){
        uartstring("TC 30 50 30 45 10 30\r");           //tracking a color
        outp(cmudat[2],PORTC);                          //storing responds
        delay(50000);
        }
}
```



Wiring diagram for the CMUcamera