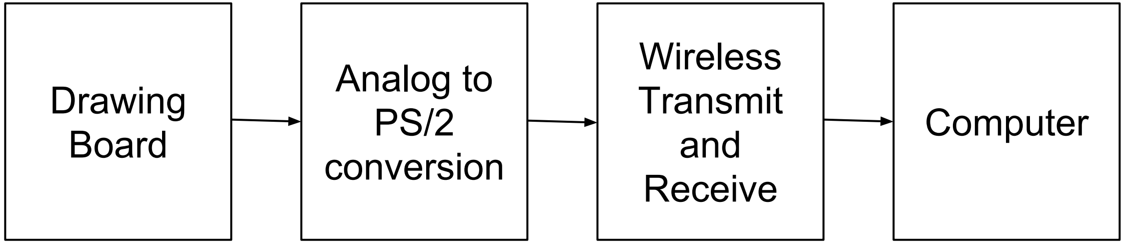
Analog Drawing Board

Henry Love and Nicholas Klugman

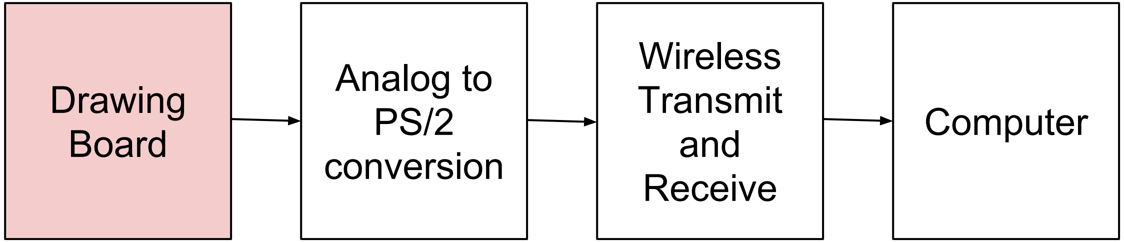
Block Diagram

3 Main Modules

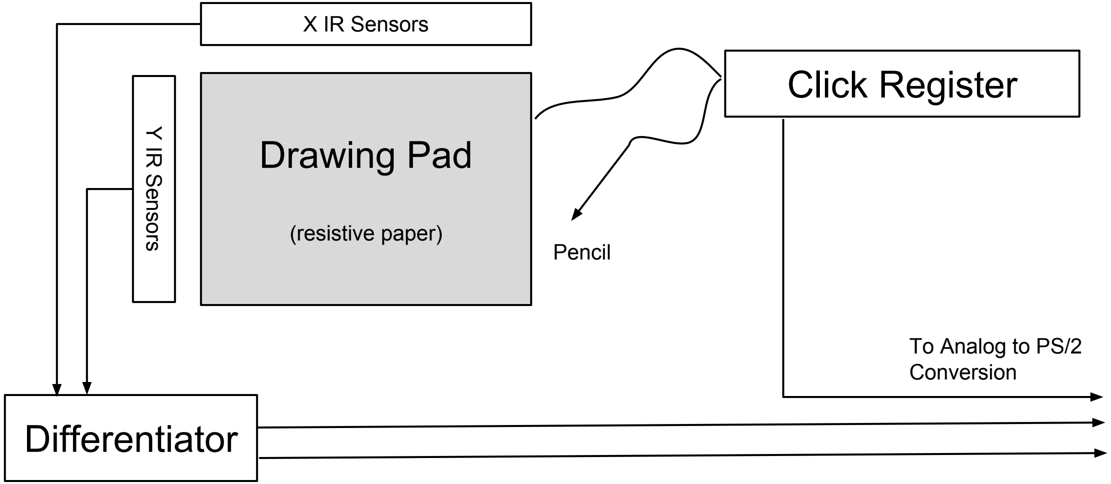
- Drawing Pad
  - IR Sensors
  - Resistive Paper
- Wireless Communication
  - Modulator
  - Encoding Scheme
- PS/2 Interface
  - Micro-Controller
  - Analog to Digital
  - Timing



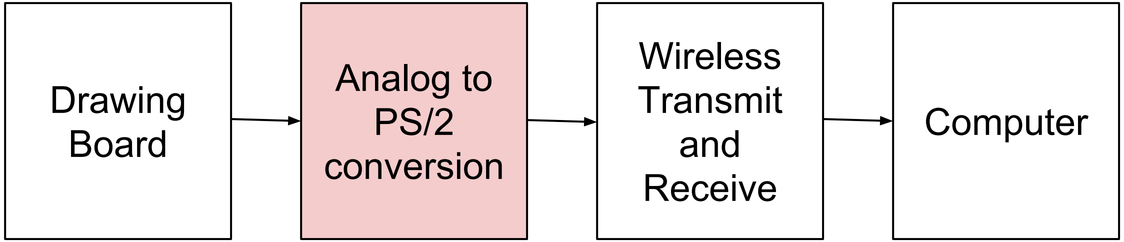
Drawing Board



- Resistive paper
- IR sensors that detect pencil
  - Still testing
- Blocking with arm poses a problem
- Put IR sensor on pencil?
- Put IR LED on pencil?



Analog to PS/2 Conversion

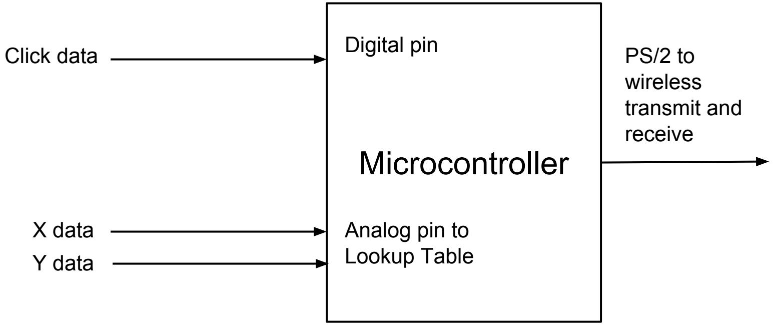

Drawing Board

Analog to PS/2 conversion

Wireless Transmit and Receive

Computer

- Takes analog signal from drawing board and converts it to PS/2
- ADC?
- May happen after signal transmission
- Current format requires two Microcontrollers
    - One for data conversion
    - One for interfacing with the computer


Click data

X data

Y data

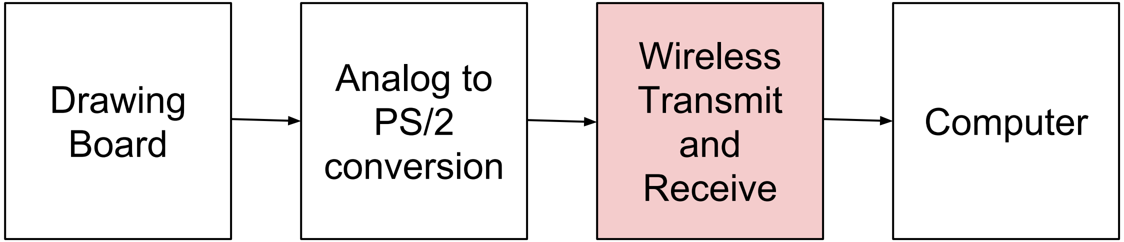
Microcontroller

Digital pin

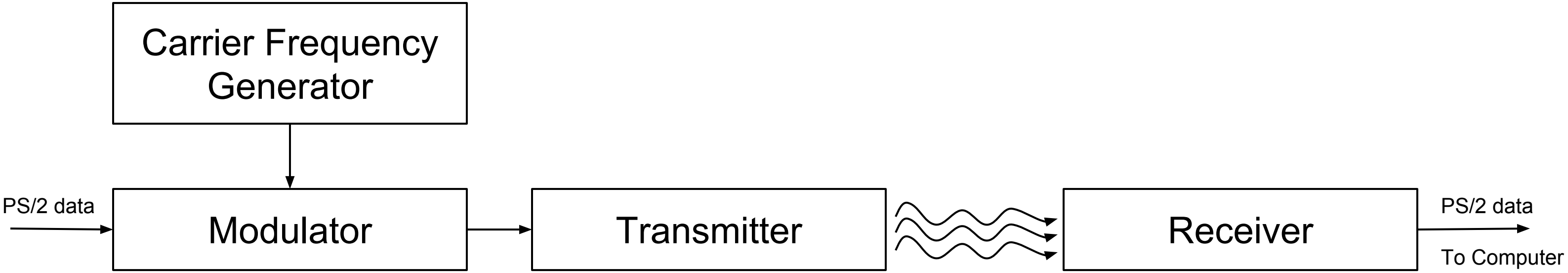
Analog pin to Lookup Table

PS/2 to wireless transmit and receive

Wireless Transmit & Receive

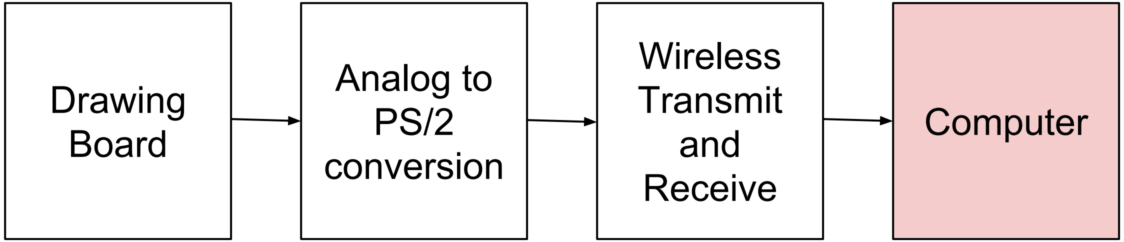


- Similar to Lab 1
- Allows the drawing board to be untethered
- 555 timer for carrier frequency generation
    - 7555 is CMOS equivalent
        - Will allow for frequencies up to 500kHz
        - 555 goes up to 100kHz reliably



Computer

- Microcomputer emulates mouse
- Talk between device and host
- Arduino code written that we can use and/or expand on

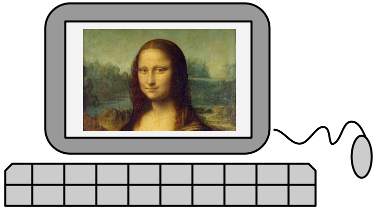


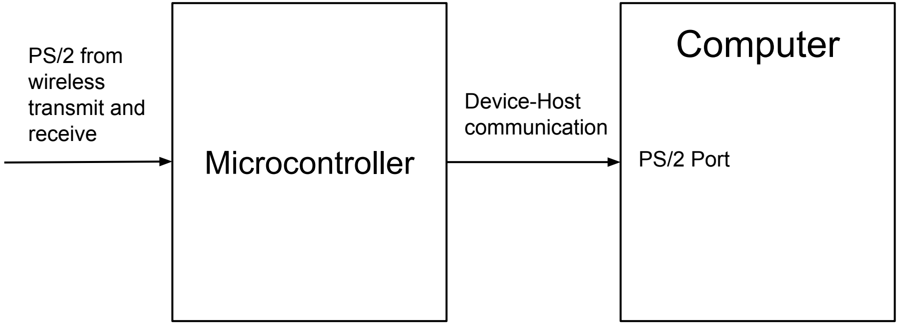
Drawing Board

Analog to PS/2 conversion

Wireless Transmit and Receive

Computer





PS/2 from wireless transmit and receive

Microcontroller

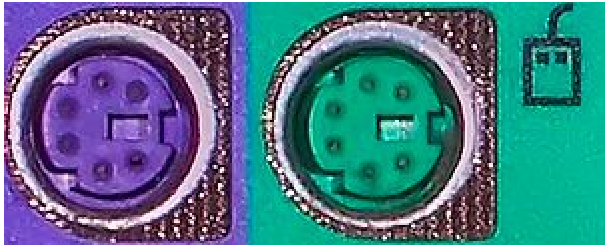
Device-Host communication

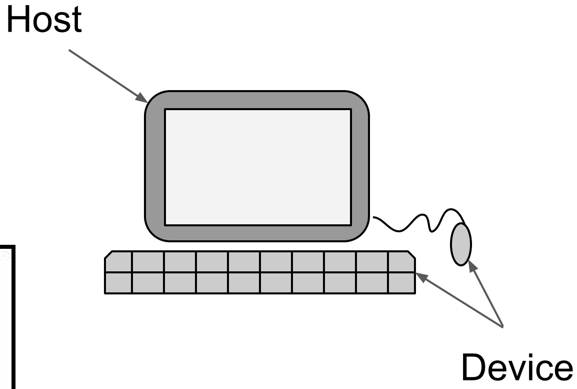
Computer

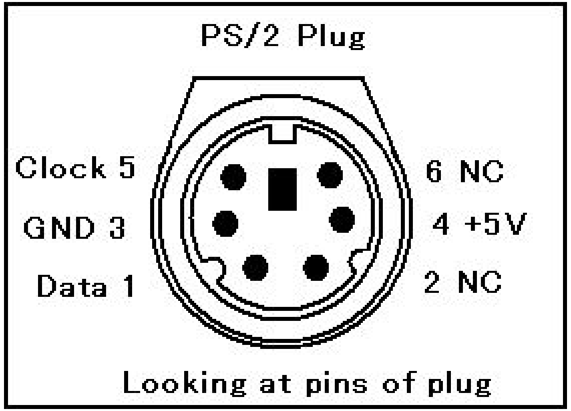
PS/2 Port

PS/2 Interface

- From IBM Personal System/2 (PS/2)
  - Getting replaced by USB
  - Simpler than USB
- Concerned with Mouse Protocol
- Maybe keyboard protocol, but that will be stretch goal
- Device sets clock
  - 10kHz - 16kHz







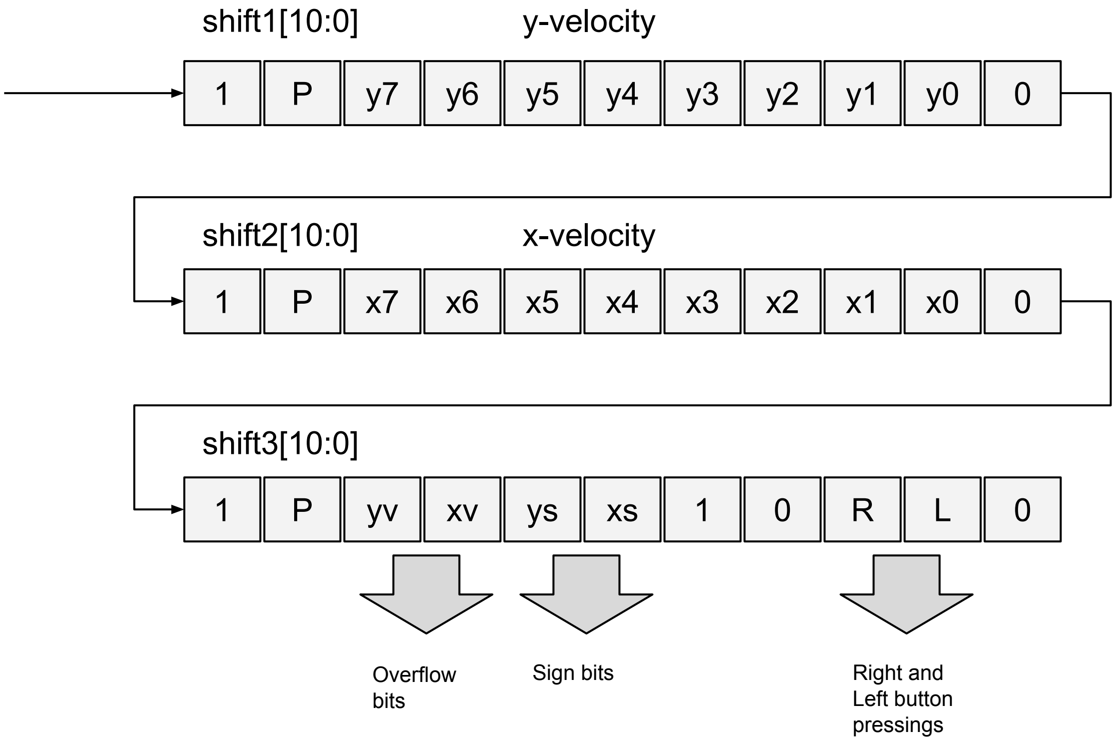
PS/2 Mouse

Four States of Operation

- Reset Mode
  - Initialization and self test
- Stream Mode
  - Default mode: transmits three data packets (in simplest form)
  - Host must transmit 0xF4 to the mouse to initiate data reporting
- Remote Mode
  - Host Request movement data packets
- Wrap Mode
  - Diagnostic

- For a double-button mouse with no scrolling, data sent in packets of 3 bytes
  - Mouse with scrolling wheel sends 4 byte packets

PS/2 Mouse



shift1[10:0] y-velocity

1	P	y7	y6	y5	y4	y3	y2	y1	y0	0

shift2[10:0] x-velocity

1	P	x7	x6	x5	x4	x3	x2	x1	x0	0

shift3[10:0]

1	P	yv	xv	ys	xs	1	0	R	L	0

Overflow bits

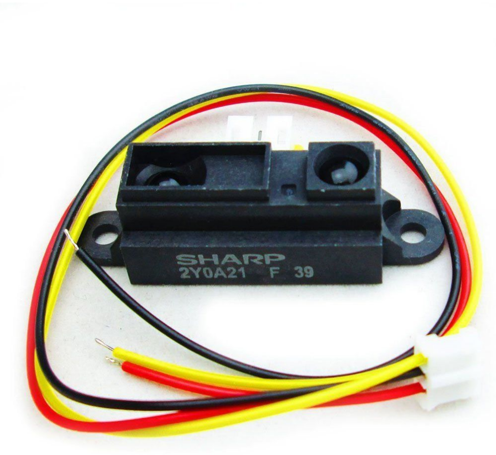
Sign bits

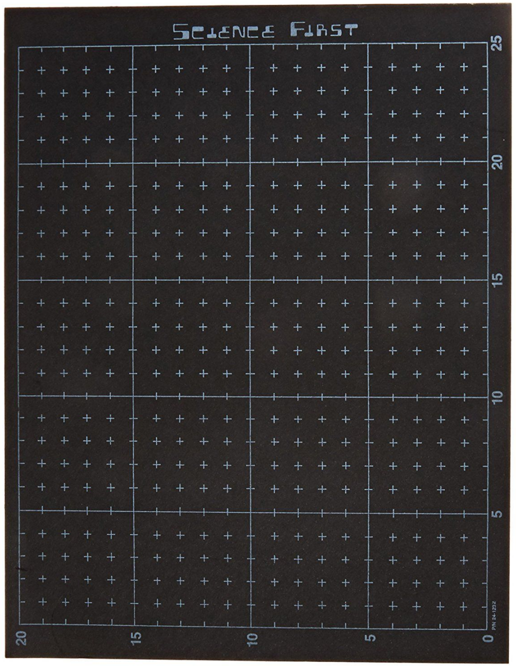
Right and Left button pressings

- Direction mouse is moving is determined by signed bits in shift3[10:0]
- Prepending sign bit to corresponding byte leads to 9 bit signed number for each velocity
  - Corresponds to data values that range from -256 to +255
- Moving too fast sets overflow bit
- Button press
  - R = 1 means right button pressed
  - L = 1 means left button pressed

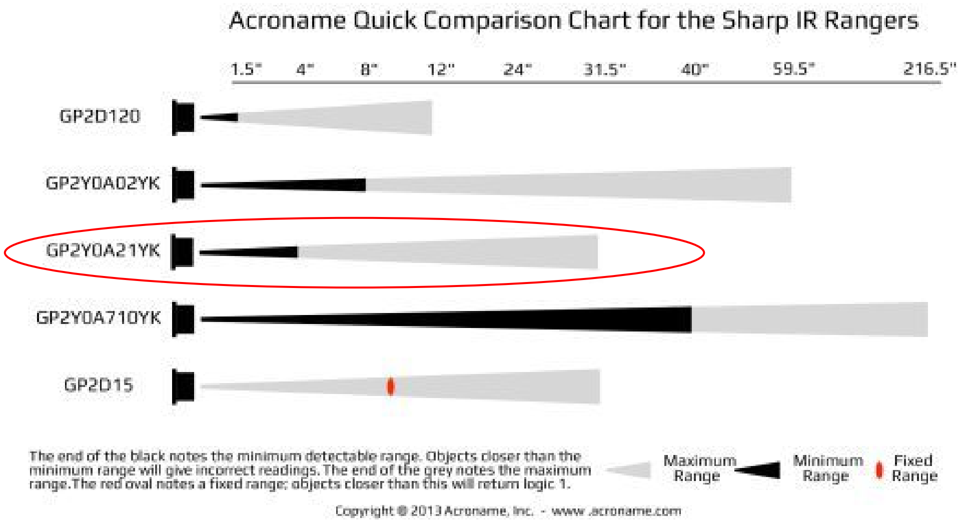
Materials

- Conductive paper
- Sharp IR Sensor GP2Y0A21YK0F
    - Range: 10 to 80cm
    - 5V at 35mA
- Microcontroller
    - Arduino
    - 8051(?)
    - 2051(?)





IR Sensor Comparison



The end of the black notes the minimum detectable range. Objects closer than the minimum range will give incorrect readings. The end of the grey notes the maximum range.The red oval notes a fixed range; objects closer than this will return logic 1.

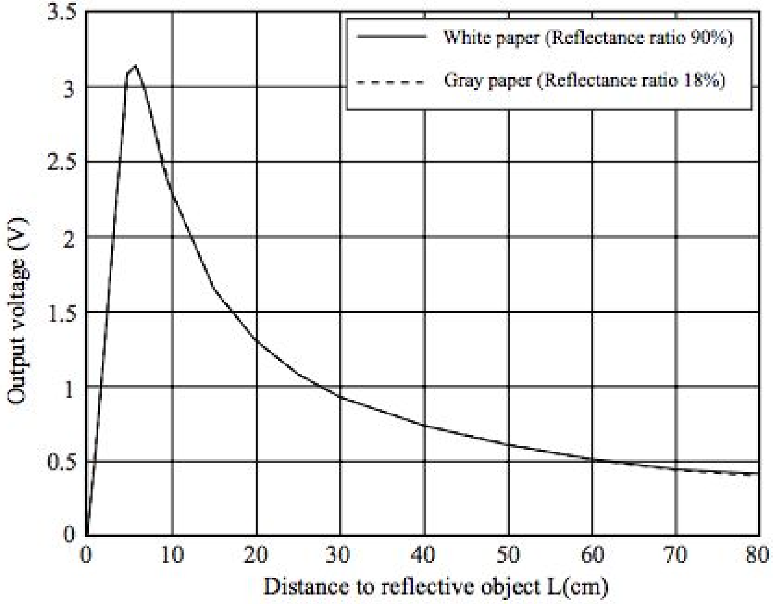
Copyright © 2013 Acroname, Inc. - www.acroname.com

Obscure

MODEL	OUTPUT	MIN. RANGE	MAX RANGE
GP2D120/GP2Y0A41	Analog	1.5"	11.8"
GP2Y0A02	Analog	8"	59"
GP2Y0A21	Analog	4"	30"
GP2Y0A710	Analog	36"	216"
~~GP2D15~~	~~Digital~~	~~9.5"~~	

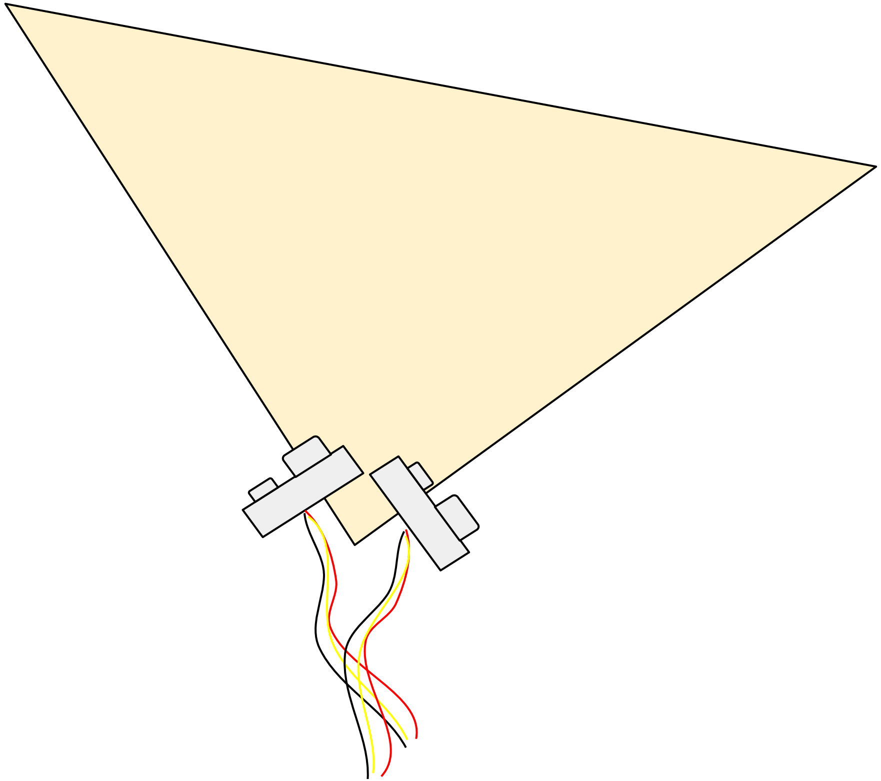
IR Sensor

- Output is in the form of an analog voltage
  - Nonlinear
- Output will be normalized
- Or we will operate in a region that appears linear



IR Sensor

- Measuring angle is quite small
- 4 IR sensors ordered
  - 2 for each axis
- 2 options:
  - Mount two 90 degrees to each other
  - Sweep the sensor
    - Slow



Schedule

Date	Goals
April 16 - 22	- Finalize Block Diagram - Order Materials - Begin initial testing
April 23 - 29	- Implement a simulated PS/2 mouse - Drawing Board touch and location working
April 30 - May 6	- Wireless circuitry - Stretch Goals - debugging
May 7 - 13	- Final debugging

Questions?

?