

Autonomous Mobile Robots

Lecture 04: Sensors

Lecture is based on material from Robotic Explorations: A Hands-on Introduction to Engineering, Fred Martin, Prentice Hall, 2001.

Outline



- Sensor Interfacing
- Switch Sensors
- Light Sensor Circuits
- Resistive Position Sensors
- Reflective Optosensors
- Break-Beam Sensors
- Shaft Encoding

Homework #4

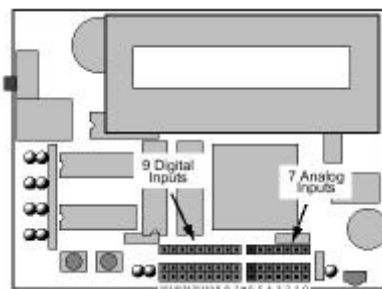
- **Sensors:** Read Chapter 3 of Robotic Explorations (textbook)

Copyright Prentice Hall, 2001

3

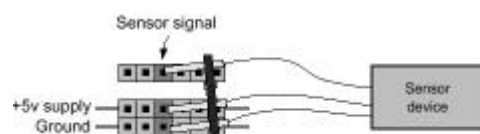
Sensor Interfacing

Handy Board's Sensor Input Banks



Handy Board has two banks for sensors:

- **Digital inputs**, numbered 15 to 7 on the left
- **Analog inputs**, numbered 6 to 0 on the right



Each sensor ports provides three signals to the sensor:

- **+5v power** - middle row
- **Ground** - lower row
- **Sensor signal line** - upper row

Not all sensors require +5v power, e.g., switches and photocells may be wired between sensor signal and ground lines

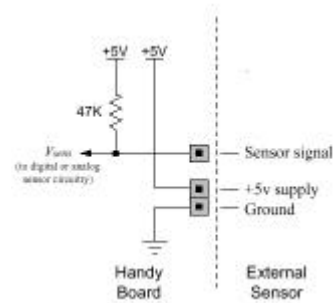
Copyright Prentice Hall, 2001

4

Sensor Interfacing

Handy Board's Sensor Input Circuitry

- Handy Board provides +5 volt and ground to the sensor, and expects the sensor's signal to be provided on the **"sensor signal" line**
- This signal line is "pulled up" to the +5 volt level with a 47K Ω resistor
 - Resistor gives the signal line a default value when nothing is plugged in
 - Also provides one half of the common **voltage divider circuit**
- The resulting signal of the circuit, V_{sens} , connects to either digital input circuitry (for the nine digital inputs) or analog input circuitry (for the seven analog inputs)



Copyright Prentice Hall, 2001

5

Sensor Interfacing

Digital Inputs

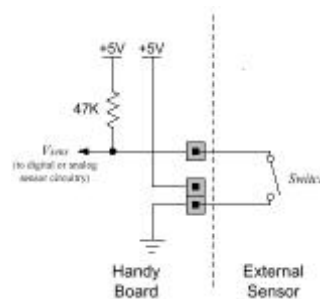
- Nine digital sensor ports connect to circuitry on the HB that interprets each sensor's V_{sens} voltage as a digital true/false

$V_{\text{sens}} > 2.5$ v, signal is **logic one**

$V_{\text{sens}} < 2.5$ v, signal is **logic zero**

switch state	V_{sens} voltage	hardware reading	digital() result
open – not pressed	5 volts	1	0 – false
closed – pressed	0 volts	0	1 – true

- To connect switch to digital input circuit:
 - Wire between the sensor signal line and ground
- "normally open"** switch
 - Switch is **released**: it is open, so there is no connection between the V_{sens} sensor line and ground. The 47K Ω pull-up resistor on the HB then provides the default value of +5v or logic one to the sensor input circuitry.
 - Switch is **pressed**: it connects the V_{sens} sensor line to ground, the zero volt level. Then the sensor input circuitry detects a logic zero reading.
- Switch reading is inverted in software: **digital()**



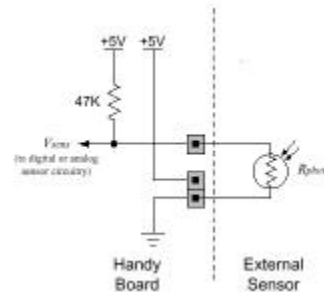
Copyright Prentice Hall, 2001

6

Sensor Interfacing

Analog Inputs

- HB's 7 analog inputs measure **continuously-varying** sensors
 - 0 to 5 volts are converted into 8-bit numbers 0 to 255 (decimal) (**A/D conversion**)
- How to wire a resistive device, such as a **photocell** element:
 - R_{photo} connects between the sensor signal line and ground
 - Photocell provides a variable resistance, which is balanced against the fixed $47K\Omega$ pull-up resistor
 - Two resistors form **voltage divider circuit**
- V_{sens} voltage at the center tap of the two resistors is proportional to the ratio of the two resistances.
 - $R_{photo} = 47K\Omega, V_{sens} = 2.5 \text{ v (exactly)}$
 - $R_{photo} \ll 47K\Omega, V_{sens} \approx \text{gnd}$
 - $R_{photo} \gg 47K\Omega, V_{sens} \approx +5 \text{ v}$
- Also possible to connect circuits that generate a voltage



Copyright Prentice Hall, 2001

7

Switch Sensors

- **Contact (touch) Sensing**
 - Switch sensors can be used to indicate when a mechanism has made physical contact with another object
 - e.g., it can trigger when a robot's body runs into a wall, or when a robot's gripper closes around a cube
- **Limit Sensing**
 - Related to simple contact sensing, a limit sensor detects when a mechanism has moved to the end of its range of travel, signaling to the control program that the motor should be turned off
- **Shaft Encoding**
 - As with past instances of shaft encoding, an axle may be fitted with a contact switch that clicks once per revolution. Software that counts the clicks can then determine the amount and speed of the axle's rotation.

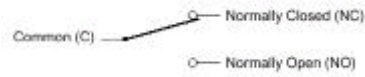


Various Switches

Copyright Prentice Hall, 2001

8

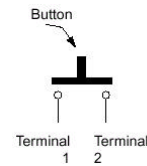
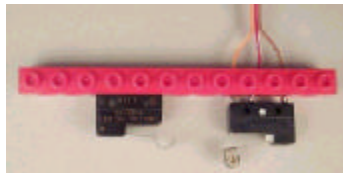
Switch Sensors



Microswitches typically have three terminals:

- “NO” (normally open)
- “NC” (normally closed)
- “C” (common)

Common terminal may connect to either of the other two terminals, depending on whether or not the switch is pressed. In the relaxed, un-pressed state, the common terminal is connected to the normally closed contact; when pressed, the common terminal moves to the normally open contact.



A **pushbutton switch** is simpler:

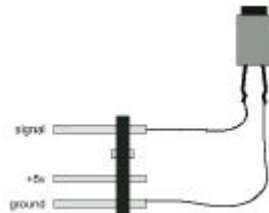
- Normally open pushbutton: when the switch is pressed, the two contacts are connected.
- Normally closed pushbuttons also exist, but these are less common.

Copyright Prentice Hall, 2001

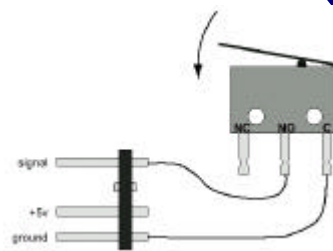
9

Switch Sensors

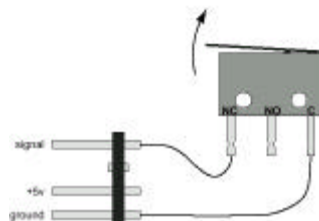
Switch Sensor Construction



Pushbutton Switch Wiring Diagram



Microswitch Normally Open Configuration



Microswitch Normally Closed Configuration

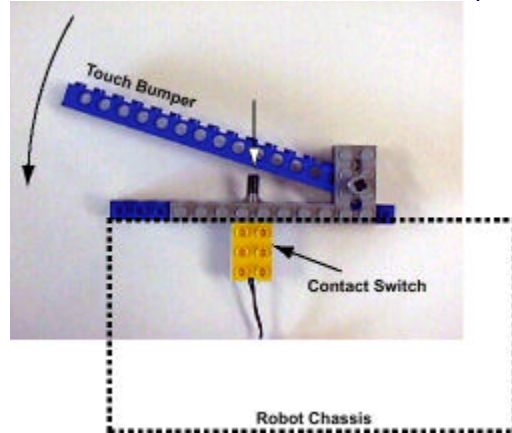
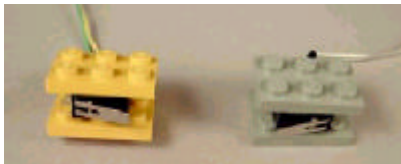
Copyright Prentice Hall, 2001

10

Switch Sensors

Switch Sensor Applications

Left- and Right-Hand Switch Construction



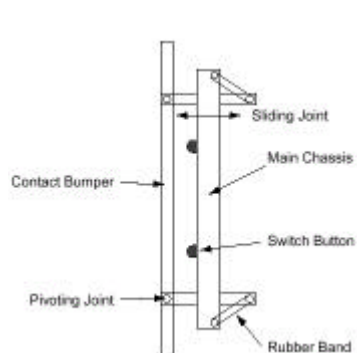
Design for a Simple Touch Bumper

Copyright Prentice Hall, 2001

11

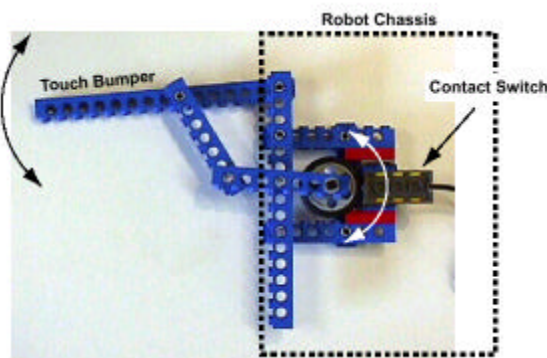
Switch Sensors

Switch Sensor Applications



HandyBug Bumper Design

- rotational and sliding pivot points allow the bumper to react to pressure from any forward direction



Design for Bi-Directional Touch Bumper

- can detect pressure from front or behind
- movement in either direction pushes levered arm away from contact sensor
- rubber bands pull arm back onto switch when pressure is released

Copyright Prentice Hall, 2001

12

Light Sensor Circuits

Single Photocell Circuit

- Photocell element is connected to the circuit ground and the HB's sensor input line via a voltage divider circuit

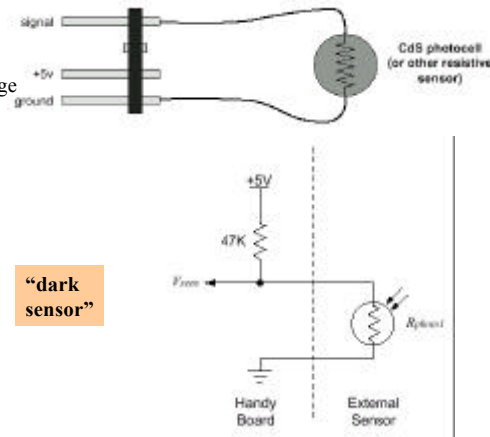
- V_{sens} , resulting sensor voltage, varies as to the ratio between $47K\Omega$ and R_{photo}

- When the photocell resistance is small (brightly illuminated), the $V_{\text{sens}} \approx 0\text{v}$
- When the photocell resistance is large (dark), $V_{\text{sens}} \approx +5\text{v}$
- Continuously varying range between extremes

- Sensor will report small values when brightly illuminated and large values in the dark

- May invert the sense of the readings from the HB's analog ports:

```
int light(int port) {return 255 - analog(port);}
```



Photocell Voltage Divider Circuit

Copyright Prentice Hall, 2001

13

Light Sensor Circuits

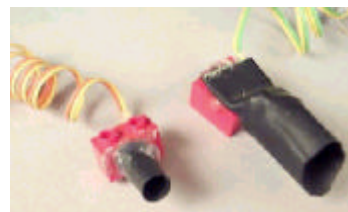
Single Photocell Circuit

- After building the photocell and test that it works: (port 0)

```
while (1) { printf("%d\n",
    analog(0));
    msleep(100L);
}
```

- Mount the photocell's leads through holes of a **LEGO Technic beam**, making a sensor device that can easily be positioned anywhere on the robot and subsequently reused

- If your photocell easily floods from ambient room light, then the next order of business is to build an **optical shield** to limit the amount of ambient light that is able to fall on the sensor



Photocell Sensors with Light Shields



Photocell Sensors Mounted on LEGO Technic Beam

Copyright Prentice Hall, 2001

14

Light Sensor Circuits

Differential Photocell Sensor

- Instead of comparing the single photocell to a fixed resistor value, the values of two photocells are compared to each other

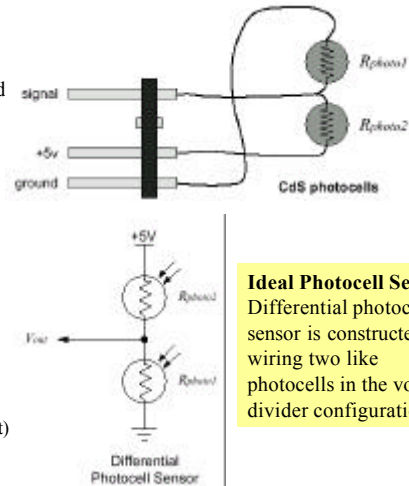
$$V_{out} = \frac{5R_{photo1}}{R_{photo1} + R_{photo2}}$$

- Differential sensor provides a signal that can be directly interpreted to indicate which side of the sensor is receiving more light, and by how much

$$R_{photo2} = R_{photo1}, V_{out} = 2.5 \text{ v}$$

$$R_{photo2} \ll R_{photo1}, V_{out} \approx +5 \text{ v (R2 more light)}$$

$$R_{photo2} \gg R_{photo1}, V_{out} \approx \text{gnd}$$



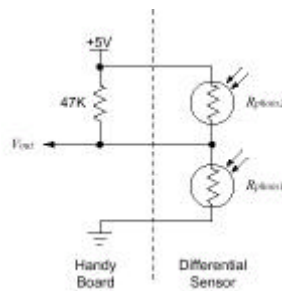
Ideal Photocell Sensor:
Differential photocell sensor is constructed by wiring two like photocells in the voltage divider configuration

Copyright Prentice Hall, 2001

15

Light Sensor Circuits

Differential Photocell Sensor



Actual Differential Photocell Sensor Schematic

Considerations:

- Use photocells with small dark resistance values, e.g., 10KΩ, Otherwise 47KΩ pull-up resistor on HB will bias sensor reading in the dark
- Mount a “nose” between two sensor elements to cast shadow on one element if there is a distinct source of light off to the side

Copyright Prentice Hall, 2001

16

Light Sensor Circuits

Differential Photocell Sensor

Program tests the value of the differential light sensor to decide which way to turn

If the value is less than 128, the program causes *HandyBug* to take a step to the left

Otherwise, *HandyBug* takes a step to the right

```
/* stepdiff.c - Light-Seeking Program for HandyBug */
int LEFT_MOTOR= 0;
int RIGHT_MOTOR= 3;
int DIFF_EYE= 0;
void main()
{
    while (1) {
        if (analog(DIFF_EYE) < 128) {
            /* turn to left */
            motor(RIGHT_MOTOR, 100); sleep(0.1); off(RIGHT_MOTOR);
        } else {
            /* turn to right */
            motor(LEFT_MOTOR, 100); sleep(0.1); off(LEFT_MOTOR);
        }
    }
}
```

Copyright Prentice Hall, 2001

17

Light Sensor Circuits

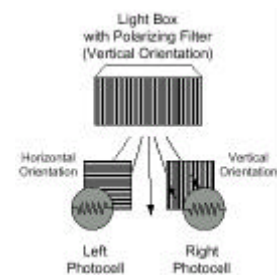
Polarized Light Seeking

- Two opposing goals consists of a light box with light filter; one has the **polarization filter** aligned **vertically**, while the other has it aligned **horizontally**

- If polarized light is passed through a filter at a right angle to the plane of polarization, it is completely blocked out
- At angles in between the 0 - 90 deg, light passes through proportional to the ratio of the polarization angle

- Using **differential sensor with polarized shields** makes it easy to tell if robot is pointed at a light beacon or not

- Sensor readings above the midpoint indicate readings from one beacon and readings below the midpoint indicate readings from the other



Robot employs a **pair of photocells**, one with a +45 deg rotation (right photocell) and one with a -45 deg rotation (left photocell). Depending on the polarization of the light source, either

- light will pass equally through both photocells' filters (no polarization)
- be blocked in the left and transmitted in the right (+45 deg polarization)
- be blocked in the right and transmitted in the left (-45deg polarization)

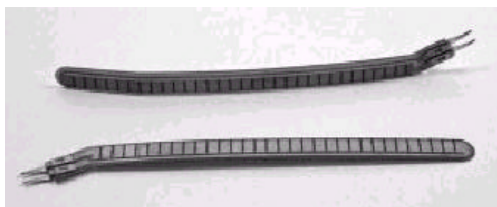
Copyright Prentice Hall, 2001

18

Resistive Position Sensors

Bend Sensors

- Useful for contact sensing and wall-tracking
- Electrically, the bend sensor is a simple resistance
 - As the plastic strip is bent (with the silver rectangles facing outward), the resistance increases
- To interface the bend sensor with the Handy Board, wire it as a single photocell
- Mechanically, the bend sensor is not terribly robust, and requires strong protection at its base, near the electrical contacts. Unless the sensor is well-protected from direct forces, it will fail over time.



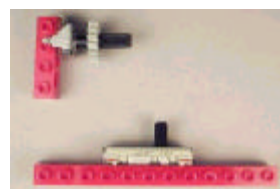
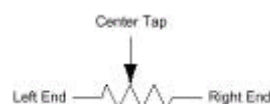
Copyright Prentice Hall, 2001

19

Resistive Position Sensors

Potentiometers

- Manually-controlled variable resistor, commonly used as volume/tone controls of stereos
- Mechanical varieties:
 - Linear and rotational styles - make position sensors for both sliding mechanisms and rotating shafts
 - Resistance between the **end taps** is fixed, but the resistance between either end tap and the **center swipe** varies based on the position of the swipe
- Electrical varieties:
 - **Linear taper** - linear relationship between position and resistance. Turn the pot 1/4 way, the resistance between the nearer end and the center is 1/4 of end-to-end resistance
 - **Audio taper** - logarithmic relationship between position and resistance. At one end, 1/4 turn would swipe over a small bit of total resistance range, while at the other end, 1/4 turn would be most of the range



Copyright Prentice Hall, 2001

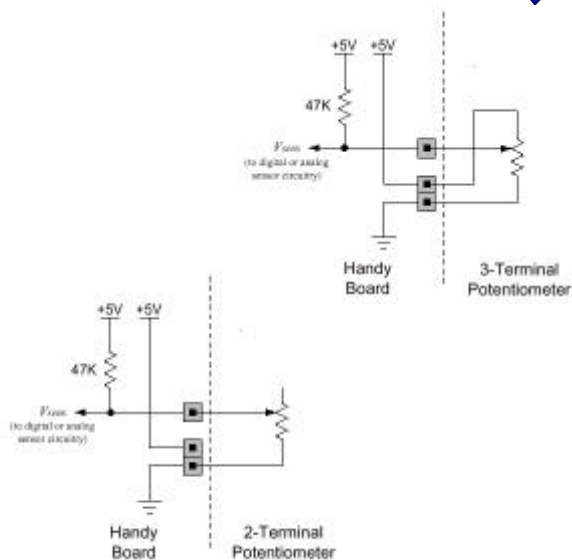
20

Resistive Position Sensors

Potentiometers

- 3-terminal potentiometer wiring works best when the effect of the Handy Board's 47K pull-up resistor is negligible; i.e., when the potentiometer resistance is **small** enough such that a 47K resistance in parallel with the pot's resistance has only a small effect

- 2-terminal potentiometer works best when the pot's value is **large** and the 47K pull-up resistor would be problematic in the 3-terminal wiring



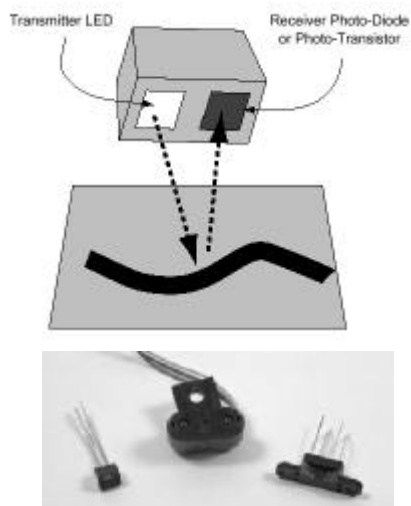
Copyright Prentice Hall, 2001

21

Reflective Optosensors

- **Active Sensor** - includes own source of quantity being detected
- Commercial ones use infrared light; include filter to pass infrared wavelengths while blocking visible light
- **Reflective optosensor** includes a source of light (emitter LED) and a light detector (photodiode or phototransistor)

- Arranged in a package so that light from emitter LED bounces off of an external object (e.g., the black line on a surface) and is reflected into the detector
- Depending on the reflectivity of the surface, more or less of the transmitted light is reflected into the detector
- Quantity of light is reported by the sensor



Copyright Prentice Hall, 2001

22

Reflective Optosensors

Applications

Object detection. Reflectance sensors may be used to measure the presence of an object in the sensor's field of view. In addition to simply detecting the presence of the object, the data from a reflectance sensor may be used to **indicate the object's distance** from the sensor. These readings are dependent on the reflectivity of the object, among other things—a highly reflective object that is farther away may yield a signal as strong as a less reflective object that is closer.

Surface feature detection. Reflective optosensors are great for detecting features painted, taped, or otherwise marked onto the floor. **Line-following** using a reflective sensor is a typical robot activity.

Wall tracking. Related to the object detection category, this application treats the wall as a continuous obstacle and uses the reflective sensor to indicate **distance from the wall**.

Rotational shaft encoding. Using a pie-shaped encoder wheel, the reflectance sensor can measure the rotation of a shaft (**angular position and velocity**).

Barcode decoding. Reflectance sensors can be used to decode information from barcode markers placed in the robot's environment.

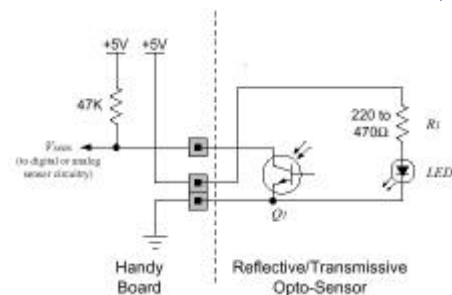
Copyright Prentice Hall, 2001

23

Reflective Optosensors

Interfacing

- Two components of the sensor, the **emitter** and **detector**, have logically separate circuits, though they are wired to the same connector plug
- The **emitter LED** (LED1), is wired to the Handy Board's +5V power supply through R1, the current-limiting resistor
 - R1's value can vary 220-470Ω, depending on how much brightness is desired from the emitter LED
- Detector Q1**, shown as a *phototransistor*, is wired between ground and the sensor signal line—just like a photocell



Reflectance Sensor Interface Diagram

Copyright Prentice Hall, 2001

24

Reflective Optosensors

Photocells vs. Phototransistors

How do you choose one type of device rather than the other?

- **Photocells** are easy to work with, because electrically they are just resistors, but their response time is slow compared to the photodiode or phototransistor's semiconductor junction. This means photocells are suitable for detecting levels of ambient light, or acting as break-beam sensors in low frequency applications (e.g., detecting when an object is between two fingers of a robot gripper).
- For applications such as shaft encoding, the rapid response time of the **photodiode or phototransistor** is required. Also, these devices are more sensitive to small levels of light, which allows the illumination source to be a simple LED element.

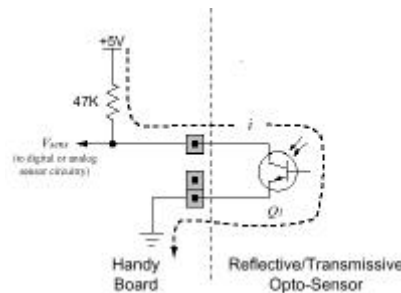
Copyright Prentice Hall, 2001

25

Reflective Optosensors

Interfacing

- Light-sensitive current source: the **more light** reaching the phototransistor, the more current passes through it
 - This creates a voltage drop in the 47K pull-up resistor on HB
 - This voltage drop is reflected in a **smaller voltage on the V_{sens} sensor** signal line, which has a level that is equal to 5 volts minus the 47K resistor's voltage drop
- Smaller values than 47K may be required to obtain good performance from the circuit
 - If transistor can typically generate currents ≥ 0.1 mA, then voltage drop across the pull-up resistor will be so high as to reduce V_{sens} to zero
 - Solution is to wire a smaller pull-up resistor with the sensor itself



The current, i , flowing through the Q_1 phototransistor is indicated by the dashed line.

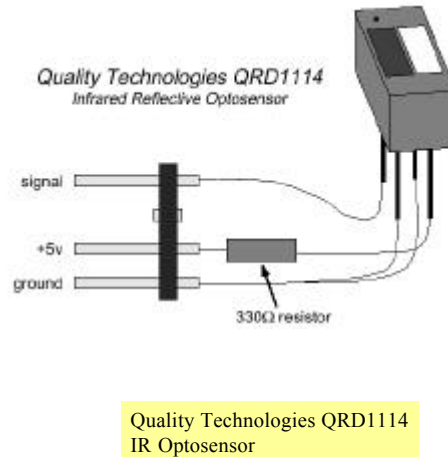
Copyright Prentice Hall, 2001

26

Reflective Optosensors

Building It

- LED emitter and detector phototransistor or photodiode are matched: at peak sensitivity of detector is at same wavelength of emissions of emitter
- Use infrared detector card to test IR light output
- Wiring
 - Detector transistor pulled high with HB internal 47K resistor
 - May have trouble figuring out which element is transistor and which is detector
 - Length of leads: longer +, shorter -
 - **Detector** connects to sensor signal line
 - **Emitter** LED connects through 330K Ω resistor to +5v supply (constantly on)



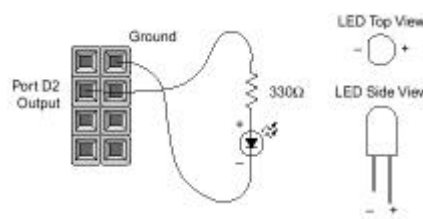
Copyright Prentice Hall, 2001

27

Reflective Optosensors

Correcting for Ambient Light

- How can a robot tell the difference between a stronger reflection and simply an increase in light in the robot's environment?
- Answer: switch a reflectance sensor's emitter light source on and off under software control
 - Take two light level readings, one with the emitter on, and one with the emitter off, then subtract away the ambient light levels
- Info about HB digital electronics:
 - Typical LED draws 5-20 mA
 - Typical processor digital output can supply 20-25 mA
 - So, a 68HC11 pin can drive 1-5 LEDs



Wiring an LED to bit 2 of Port D (Serial Peripheral Interface) Pin

```
int active_read(int port)
{
    int dark, light;          /* local variables */
    dark= analog(port);       /* reading with light off */
    bit_set(0x1009, 0b00000100); /* turn light on */
    light= analog(port);      /* reading with light on */
    bit_clear(0x1009, 0b00000100); /* turn light off */
    return dark - light;
}
```

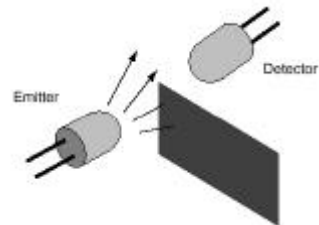
Copyright Prentice Hall, 2001

28

Break-Beam Sensors

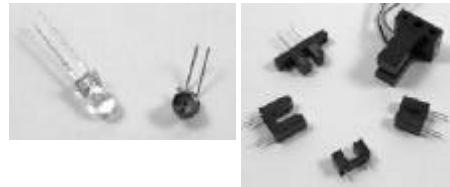
- Pairs of light-emitting and light-detecting components used in the break-beam configuration

- Light-emitting component aimed at a light-detecting component
- When opaque object comes between emitter and detector, the beam of light is occluded, and the output of the detector changes



- Any pair of compatible emitter–detector devices may be used:

- Incandescent flashlight bulbs and photocells
- Red LEDs and visible-light-sensitive phototransistors
- Infrared emitters and detectors



Discrete infrared LED, phototransistor and various commercial break-beam optosensors

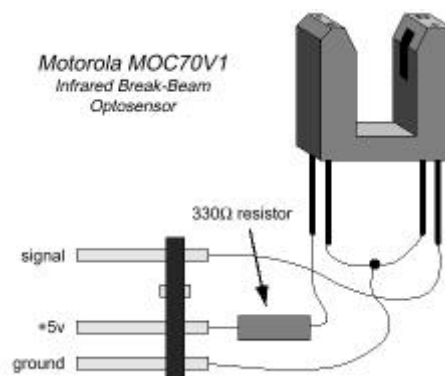
Copyright Prentice Hall, 2001

29

Break-Beam Sensors

Building

- For sensing objects between **larger gaps**, use discrete emitters and detectors
- Interface to HB the same as for the reflective optosensors
 - Emitter LED powered from HB +5v supply through dropping resistor
 - Detector phototransistor connected between sensor signal line and ground
 - Polarity is not indicated by length of device leads; look for + marking
- Brainstorm robotic applications for break-beam sensing
 - e.g., detecting something between fingers of a robotic gripper

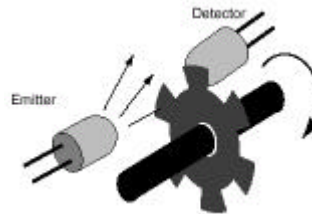


Copyright Prentice Hall, 2001

30

Shaft Encoding

- Use Break-Beam Sensors
- Shaft encoder measures the angular rotation of an axle, reporting position and/or velocity information
- Example: **speedometer**, which reports how fast the wheels are turning; **odometer**, which keeps track of the number of total rotations



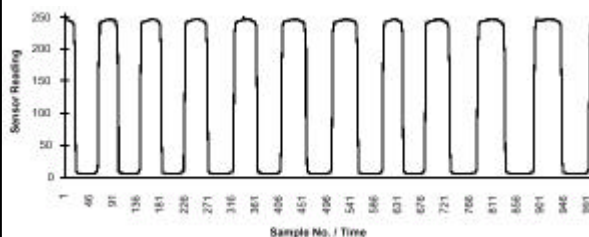
Single-Disk Shaft Encoder

A perforated disk is mounted on the shaft and placed between the emitter-detector pair. As the shaft rotates, the holes in the disk chop the light beam. Hardware and software connected to the detector keeps track of these light pulses, thereby monitoring the rotation of the shaft.

Copyright Prentice Hall, 2001

31

Shaft Encoding



Data from shaft encoder built from MOV70V1 break-beam sensor and pulley wheel:

The sensor data graph is a **nearly ideal square wave**. Using the standard HB analog input, which reports a sensor reading between 0 and 255, the sensor's output varies from a low of about 9 (about 0.18 volts) to a high of about 250 (4.9 volts) with a sharp edge between the transitions.

Other break-beam sensors yield a time graph that looks more like a sine wave.

This assembly uses the Motorola break-beam sensor with the medium pulley wheel as a photo-interrupter. After determining a position of the break-beam sensor that yielded good break and make transitions, the sensor was hot-glued into position along the LEGO beam.

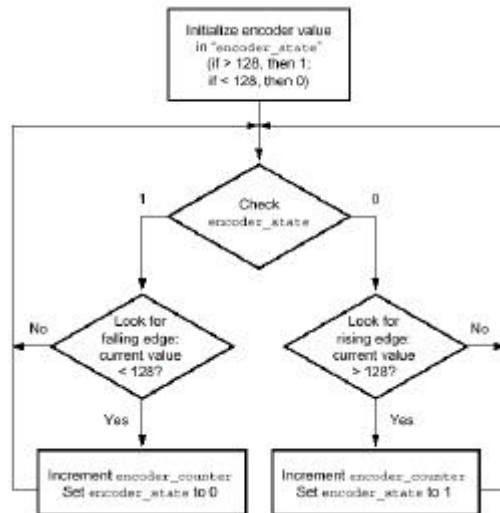
Copyright Prentice Hall, 2001

32

Shaft Encoding

Counting Encoder Clicks

- To make sense of data from a shaft encoder, install a routine that repeatedly checks the sensor value.
 - If the encoder wheel turns faster than the routine checks the sensor state, it will start missing transitions and lose track of the shaft's rotation
 - Solution: check midrange point
- Variables for algorithm:
 - encoder_state** - Keeps track of last encoder reading: 1 if high (above 128), 0 if low (below 128)
 - encoder_counter** - Keeps running total of encoder "clicks"



Copyright Prentice Hall, 2001

33

Shaft Encoding

Driver Software

- Machine language routine loaded into IC's underlying layer of direct 68HC11 code, with user interface - *IC binary* (ICB) files installed in interrupt structure of 68HC11
- Monitors shaft encoder values and calculates encoder steps and velocity needs **quickly and at regular intervals**
- HB's software libraries include set of routines for supporting shaft encoders for both position-counting and velocity measurement. For each analog input on HB, a pair of shaft encoder routines is provided. For each pair, there is a high-speed version and a low-speed version.
 - High speed version checks for transitions on the encoder sensor **1000 Hz**
 - Low speed version checks encoder at **250 Hz** (less of a processing load on the system)
 - Both versions calculate the velocity (position difference) measurement at about 16 Hz
- Once loaded into IC, the encoder routines are automatically active; no additional commands are needed to turn them on.
 - Each *encoder0_counts* variable (running total of transitions on encoder sensor) will automatically increment every time it senses a transition on its corresponding encoder sensor
 - The *encoder0_velocity* value (velocity measurement) is continuously updated

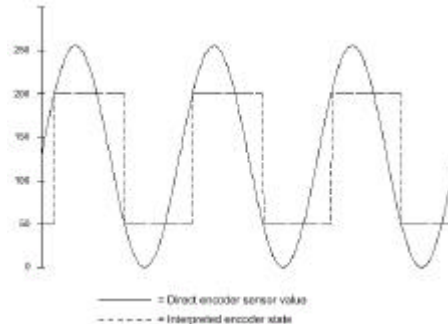
Copyright Prentice Hall, 2001

34

Shaft Encoding

Driver Software - Hysteresis

- Library routines use **two thresholds** to track changes in the raw encoder reading
 - Reading must rise above a high threshold value to be considered high, and then fall below a low threshold to be considered low
 - Square wave changes state either when encoder reading rises above 200, or falls below 50
 - When encoder value is between these two thresholds, square wave “waits” for value to reach threshold before changing state
- This algorithm implements a property known as **hysteresis**, in which an output signal depends on the input signal and its recent history



The sine wave represents the **raw encoder value**. The dashed square wave represents the **encoder state of the driver routine** — each transition on the square wave corresponds to an encoder count.

Copyright Prentice Hall, 2001

35

Shaft Encoding

Measuring Velocity

- Driver routines measure rotational velocity as well as position
 - Subtract difference in the position readings after an interval of time has elapsed
- Velocity readings can be useful for a variety of purposes
 - Robot that has an **un-powered trailer wheel** with a shaft encoder can easily tell whether it is moving or not by looking at encoder activity on the trailer wheel. If the robot is moving, the trailer wheel will be dragged along and will have a non-zero velocity. If the robot is stuck, whether or not its main drive wheels are turning, the trailer wheel will be still.
- Velocity information can be **combined with position information** to perform tasks like causing a robot to drive in the straight line, or rotate a certain number of degrees. These tasks are inherently unreliable because of mechanical factors like slippage of robot wheels on the floor and backlash in geartrains, but to a limited extent they can be performed with appropriate feedback from shaft encoders.

Copyright Prentice Hall, 2001

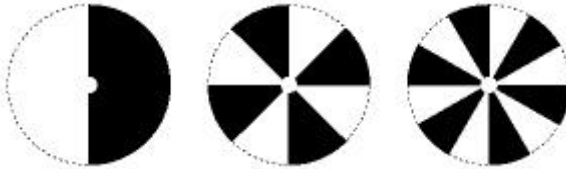
36

Shaft Encoding

Reflective Optosensors as Shaft Encoders

- It's possible to build shaft encoders by using a reflective optosensor to detect black and white markings on an encoder wheel

- Wheels can be used with any of the reflective optosensor devices, as long as the beam of light they generate is small enough to fit within the black and white pie-shaped markings



Copyright Prentice Hall, 2001

37

Shaft Encoding

Opto-Electronic Computer Mice

- Common **desktop mouse** uses shaft encoder technology to figure out how the mouse ball is turned

- Two slotted encoder wheels are mounted on shafts that are turned by the ball's movement

- On either side of each encoder wheel are the infrared emitter and detector pair

- Mice use **quadrature shaft encoding**, a technique that provides information about which way the shaft is turned (in addition to the total "encoder clicks")

- IR detector on each shaft actually has two elements, aligned so that as one element is being covered up by the leaf between the slots, the other is being exposed



Copyright Prentice Hall, 2001

38

Shaft Encoder Exercises

1. Build a shaft encoder using a break-beam optosensor and a perforated disk or LEGO pulley wheel. Verify the raw sensor performance—what values represent the light beam being broken vs. not broken?
2. Choose a suitable midpoint value for determining encoder transitions. Write a program in IC to implement the simple encoder counting algorithm presented in the flowchart. Use IC multi-tasking capability to display the encoder counter variable while the counting routine is running, and experiment with the encoder. Can you determine the performance limit of the algorithm in your implementation, in terms of counts per second? What is a fundamental problem with this implementation method?
3. Load a library shaft encoder routine and experiment with its performance. Capture raw data from the encoder. Based on the graph of raw encoder performance, choose suitable high and low threshold values. Explain your choices.
4. One limitation of current encoder routines, both the IC and library versions, is that they cannot determine which *direction* the shaft is rotating. Can you think of a different approach for determining the direction of rotation?
5. Implement the trailer wheel idea discussed in the text on your *HandyBug*. Write a program to make *HandyBug* drive around and stop, back up, and turn when the trailer wheel's velocity is 0. Can you think of other applications for knowing the robot's velocity, other than as a non-zero/zero (i.e., moving/not moving) quantity?
6. Instrument one of *HandyBug*'s drive wheels with an encoder, and write a program that attempts to maintain constant velocity on the drive wheel by varying the power level delivery to the motor. Experiment with the system by holding *HandyBug* in the air and applying pressure to the drive wheel. Is the system able to maintain the velocity? What happens if you suddenly remove the pressure?