Categories of Robots and their Hardware Components

Click to add Text

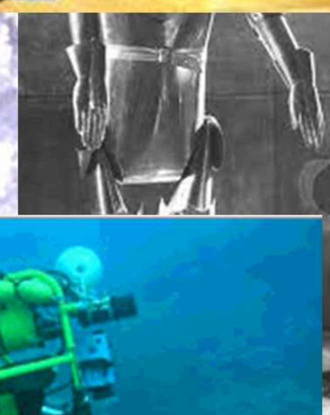
Martin Jagersand

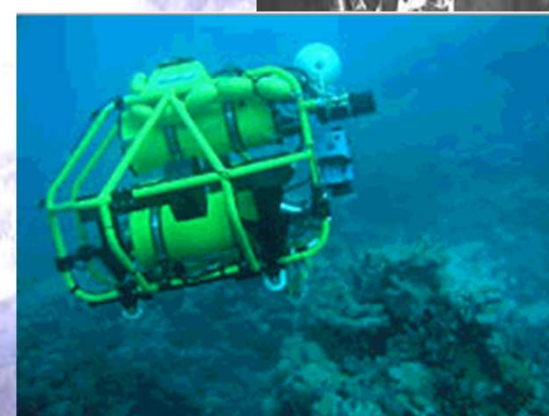
Robot?

ext

Robot?

ext





How do we categorize these robots?

- What they can do?

  Most robots can move things (but varying distances)

- What sensors they have?

  We can generally equip any robot with any type of sensor.

- How they move

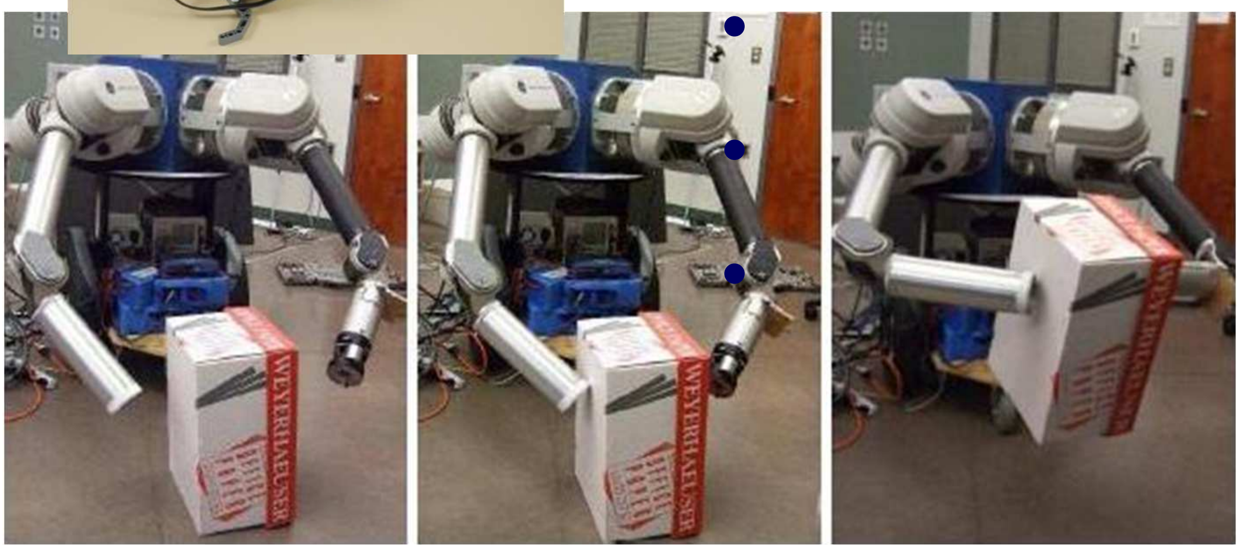
  Yes! The motion properties of an arm is different from a mobile robot/car and a UAV/helicopter

Mobile Robots (ground)

- Moves on 2D ground surface
- Needs just 2 motors
- Inexpensive
- Easy to model and control
- Large range of motion
- Hard to exactly localize
- Cannot generally pick things up and manipulate them

Robot arms and hands (linkages)

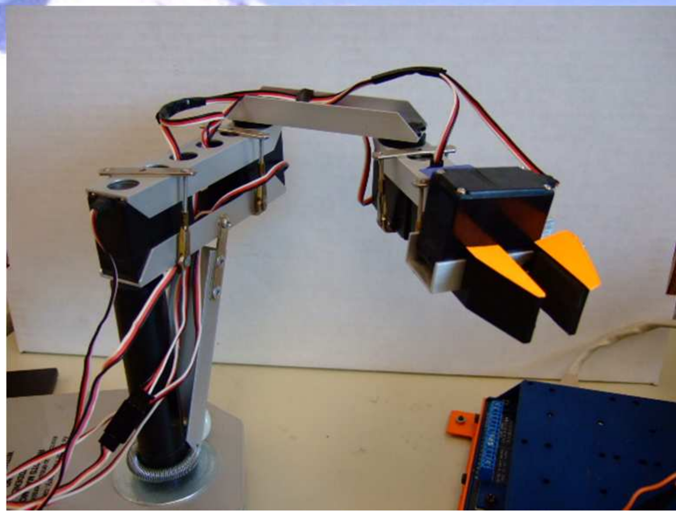
- Moves in 6DOF (3D position, 3D orientation)
- Min 6 motors in linkage
- Perfect localization
- Expensive
- Easy to control
- Manipulates!

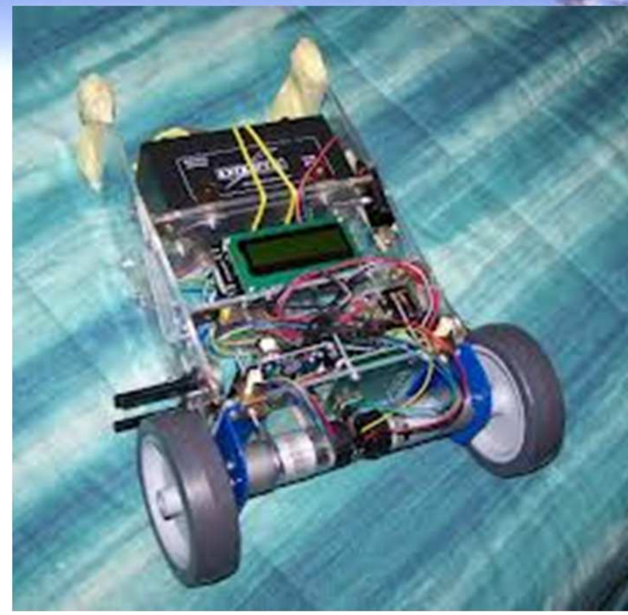


Free-flying robots
Aerial and Underwater

- Moves in 6DOF (3D position, 3D orientation)
- Generally underactuated (4DOF)
- Hard to model and control
- User error = Crash and break
- Hard to exactly localize
- RC heli inexpesive,
- Military UAV expensive

Parts of a Robot



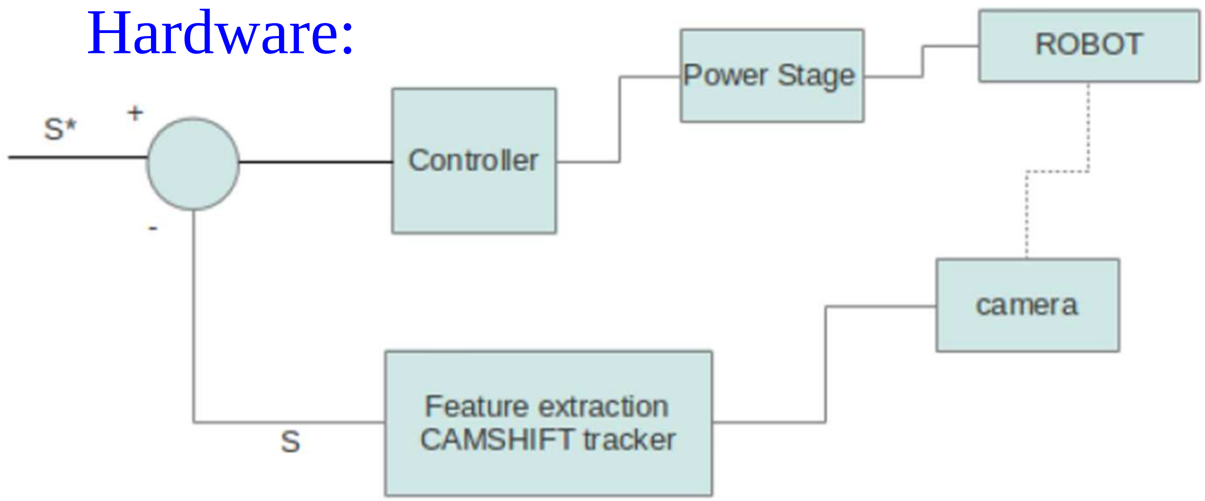


- Motors
- Motor controllers
- Transmission
- Linkages
- Sensors
- Computer

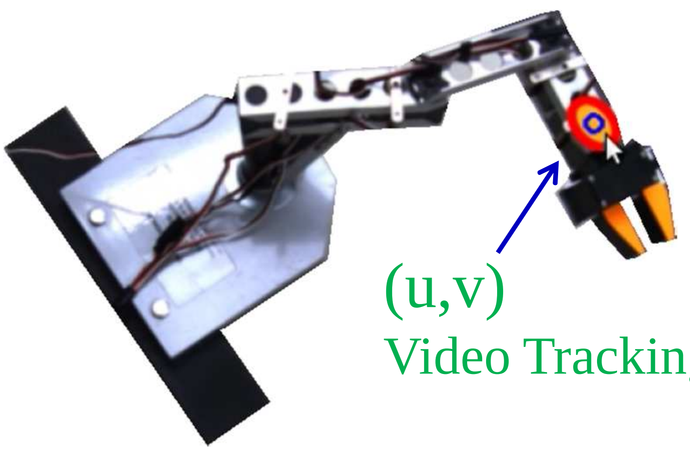
Example Software and Hardware Block Diagram

One of our robots in **Western Canadian Robot Games** May 2012

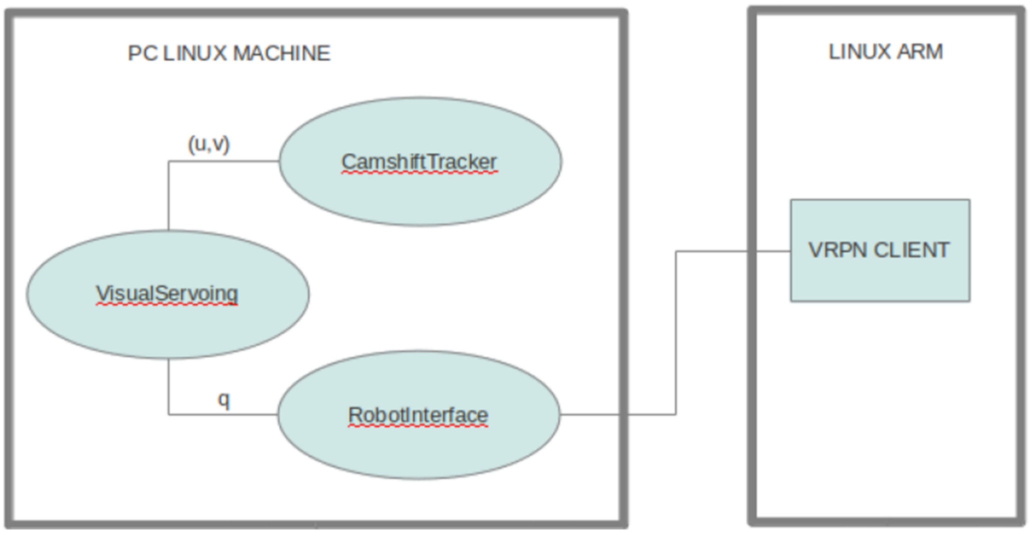
Hardware:



Software: Python and ROS processes



(u,v)
Video Tracking



Motors

- Regular brushed DC motors

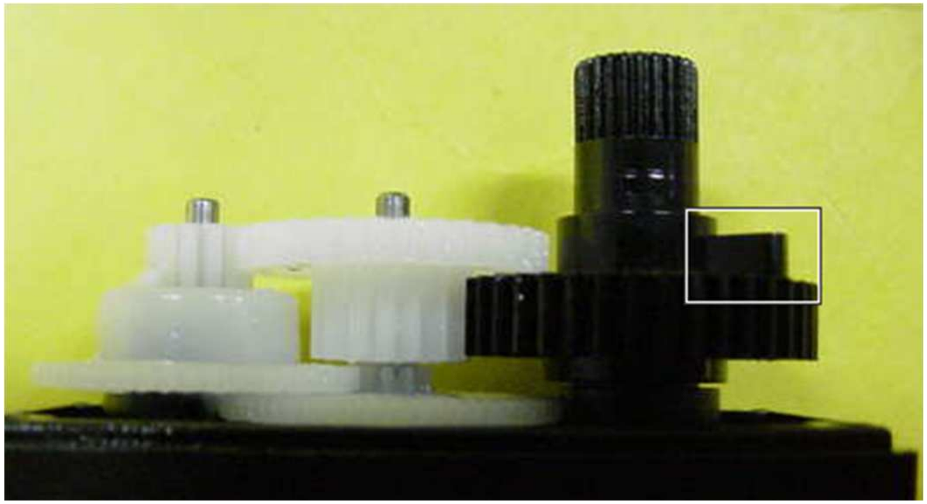
- RC servo motors

- BLDC motors

RC Servo Motors

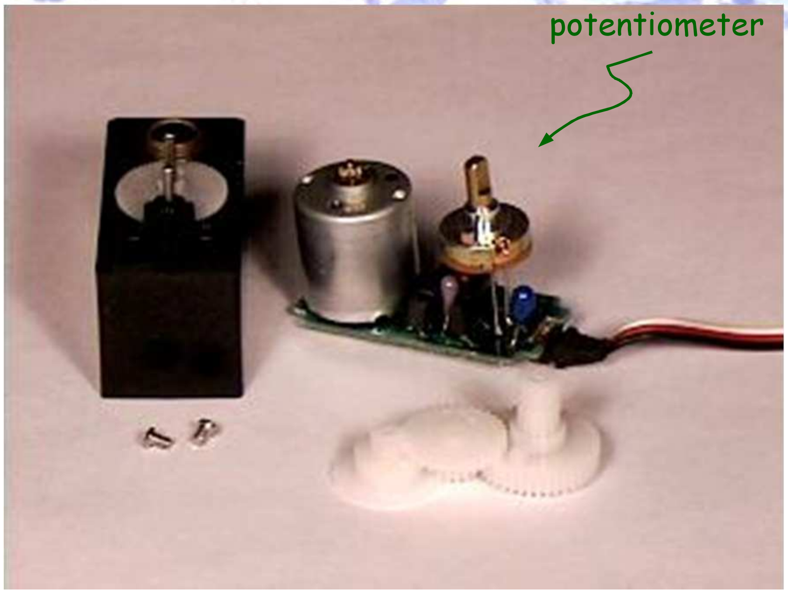
- Is a brushed motor+ gearbox+control electronics
- Does one revolution

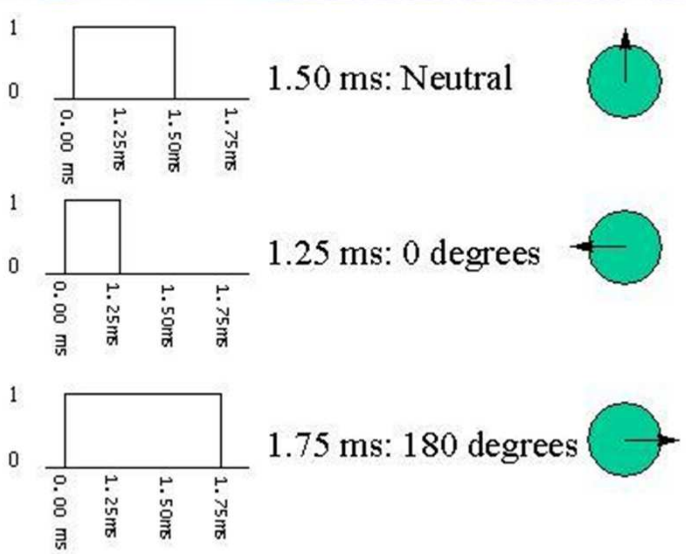
- Can modify for more by removing electronics and cutting stop tab



http://www.instructables.com/id/How-to-modify-a-servo-motor-for-continuous-rotatio

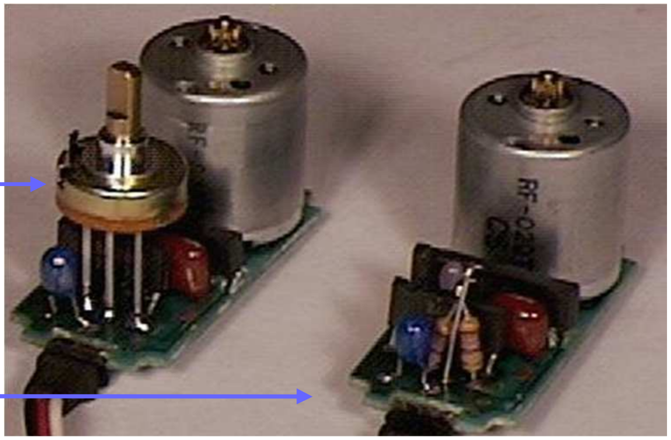
RC Servo motors





Direct position control in response to the width of a regularly sent pulse.

A potentiometer is used to determine the motor shaft angle.

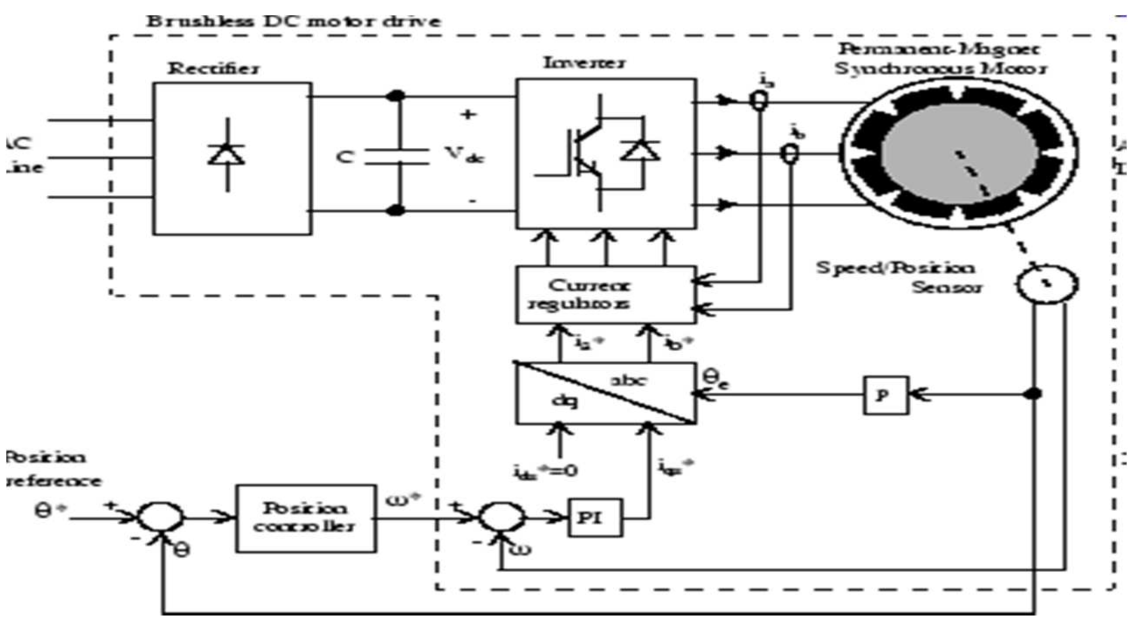


modified to run continuously

BLDC Motors

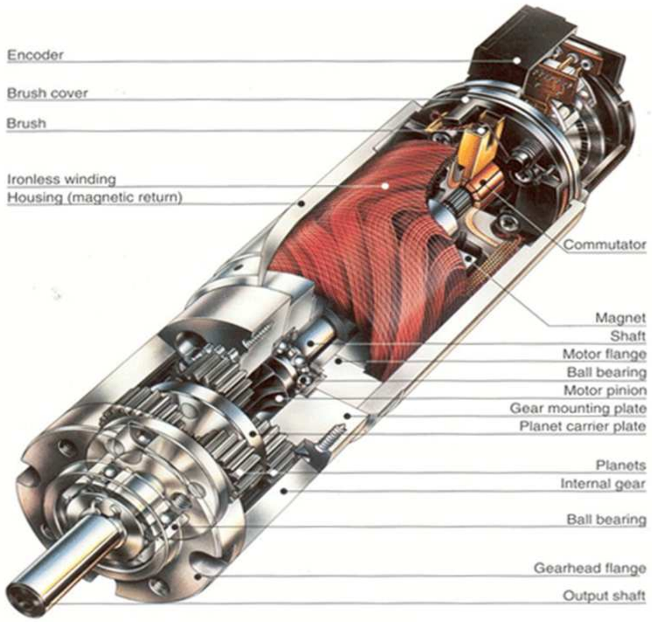
- Inexpensive
- Powerful
- Precise
- Used in newer commercial arms

- But:
- Need 3Phase controller

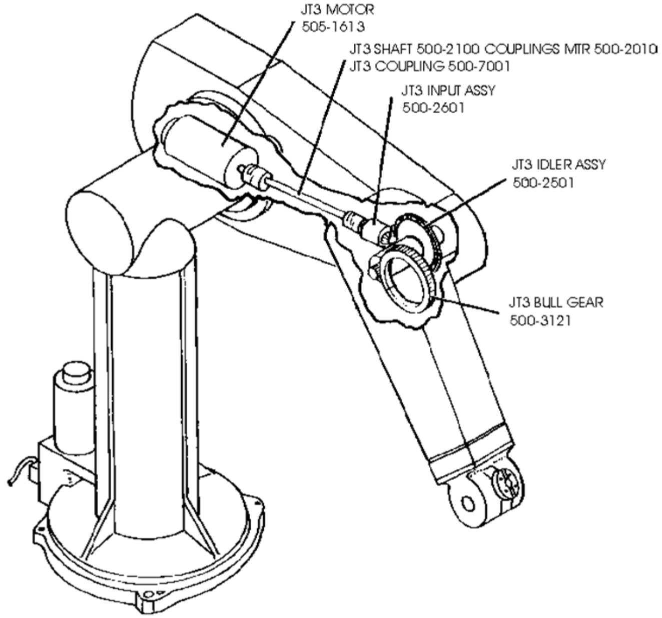


Transmission

- Gears
- Most common in industrial robots
  - Friction and backlash/deadzone



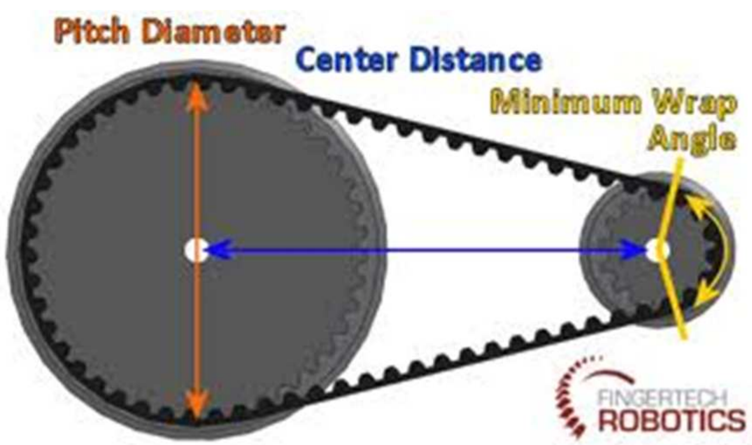
Geared servo motor

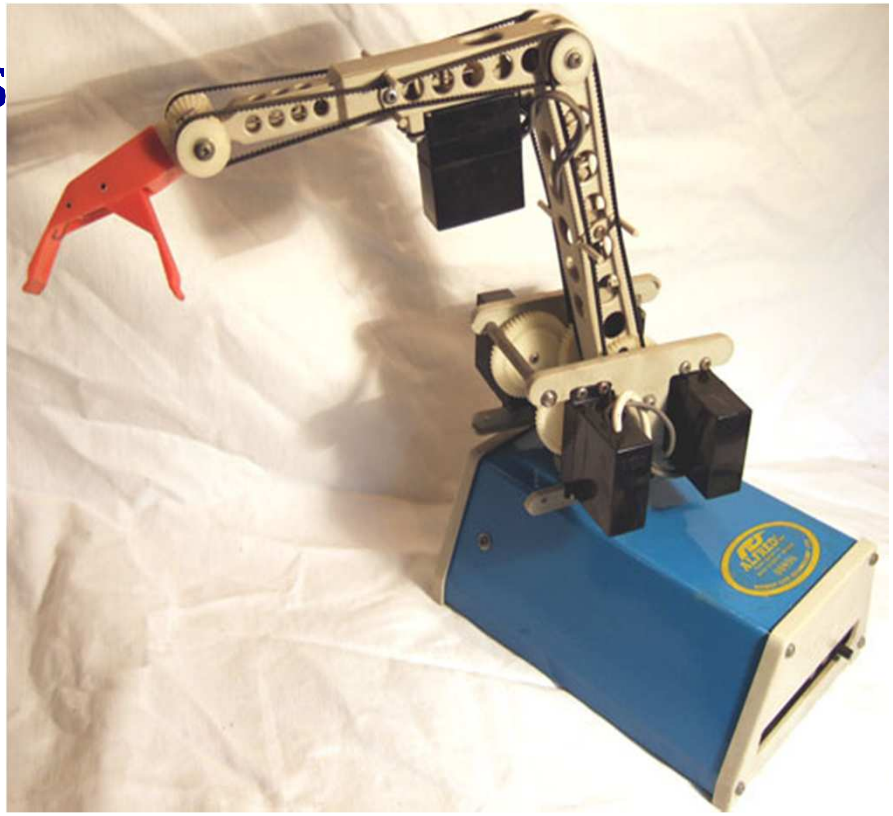


Puma 560 joint 3 gears and shafts

Transmission

- Timing/toothed belts
- Common in one-off robots
- Less stiff

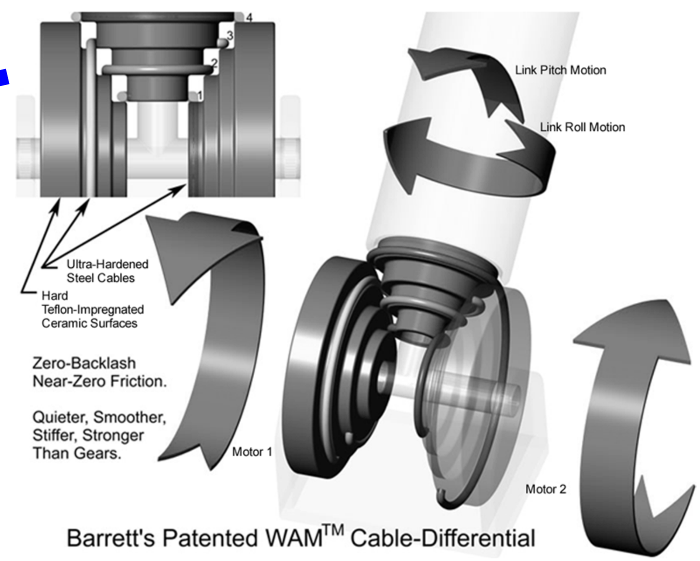




Beebcontrol “Alfred”

Transmission

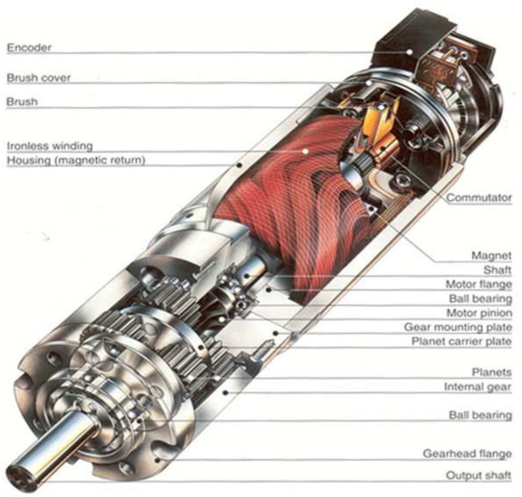
- Steel cables/tendons
- Both light and stiff
- Wear quicker

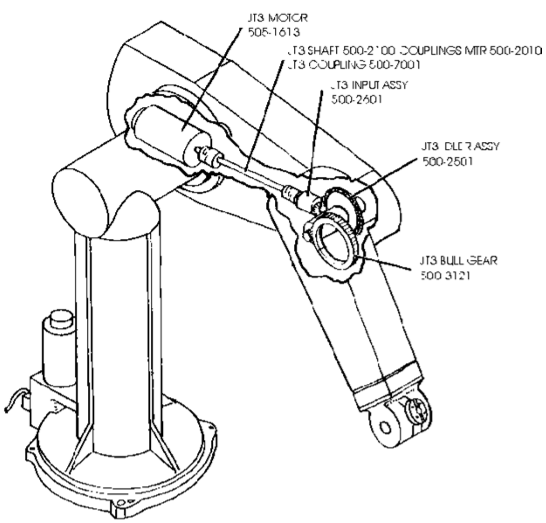


WAM: cables connect motors in base to joints in arm

Transmission

- Gears





Cables/tendons

- Belts

Beebcontrol “Alfred”

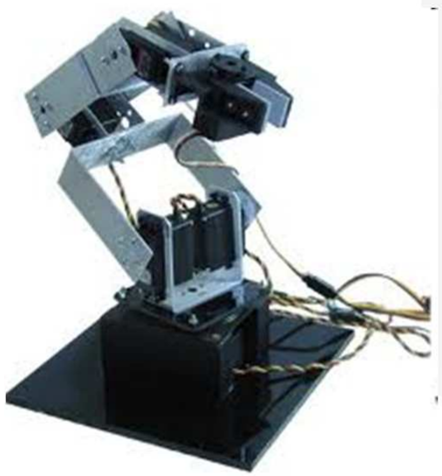
Linkage configuration

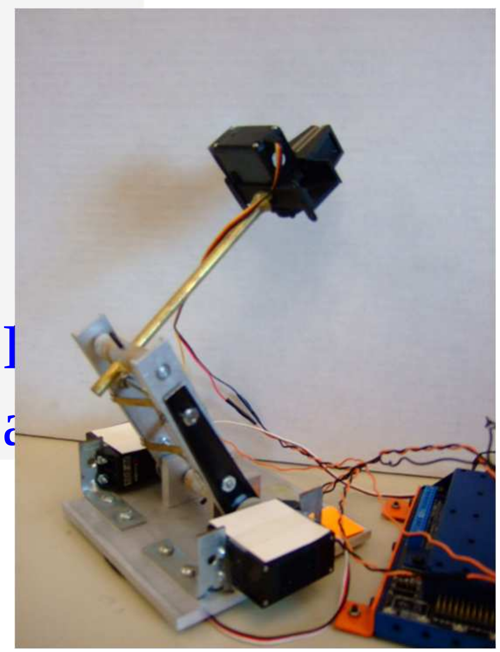
- Motors serially in arm
- Each motor carries the weight of previous
- Heavy

Motors at base

Lightweight and faster

More complex transmission



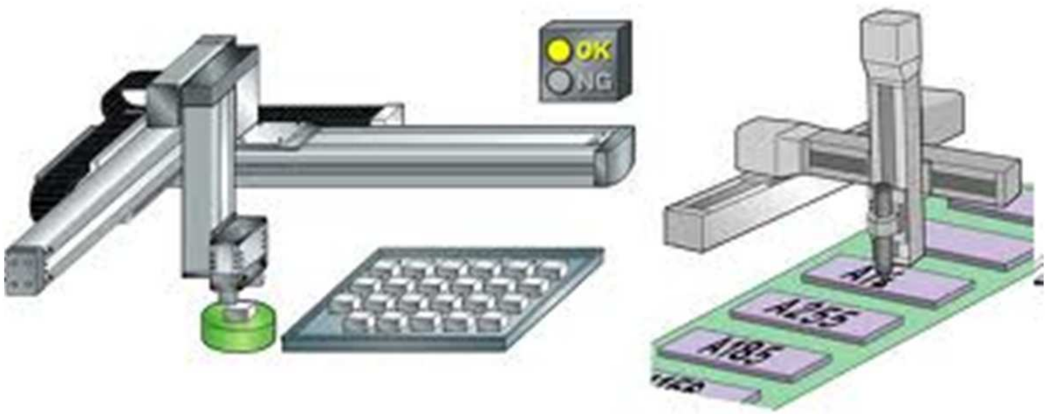


Linkage Material

- Aluminum profiles

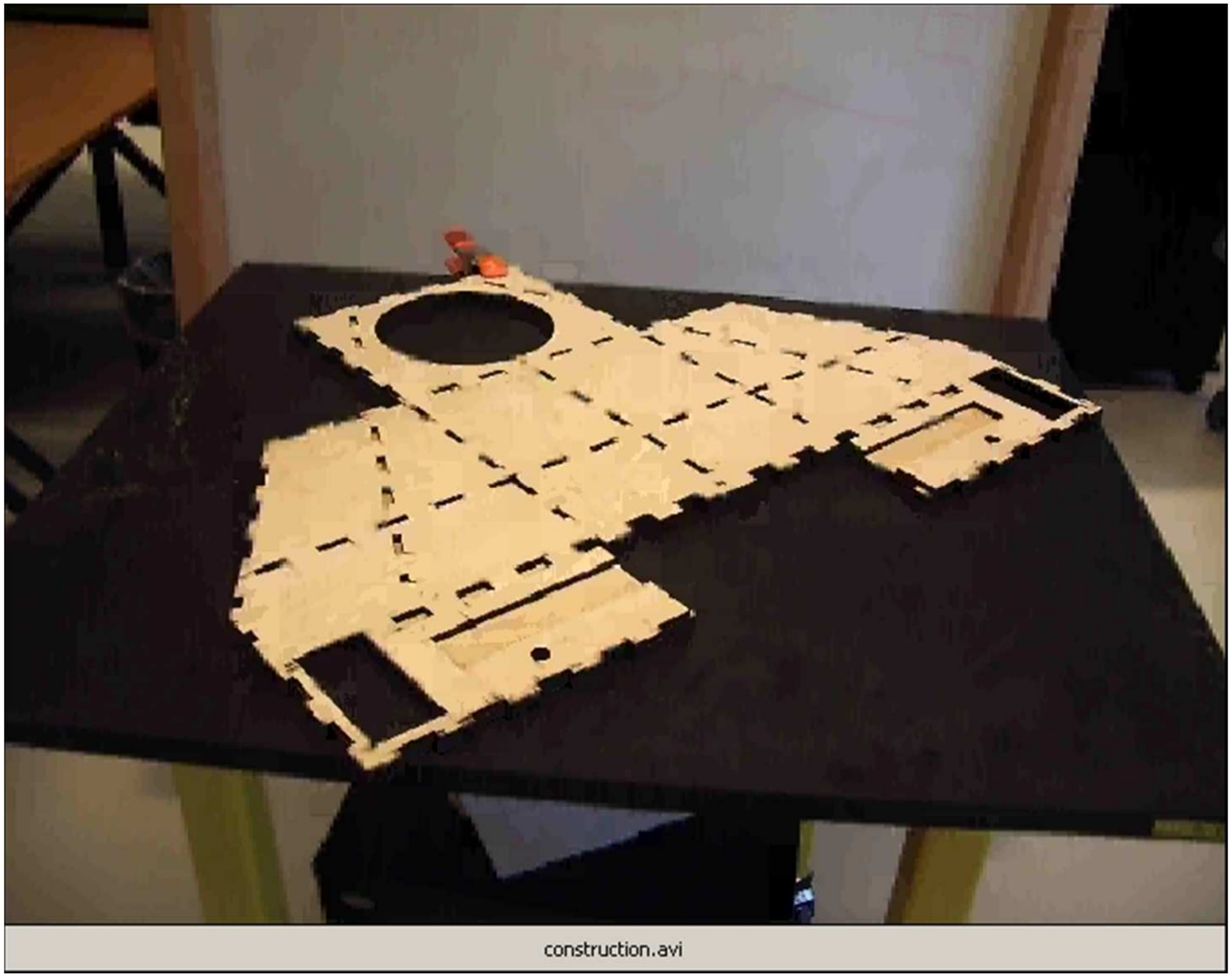
Linkage Material

- Aluminum profiles



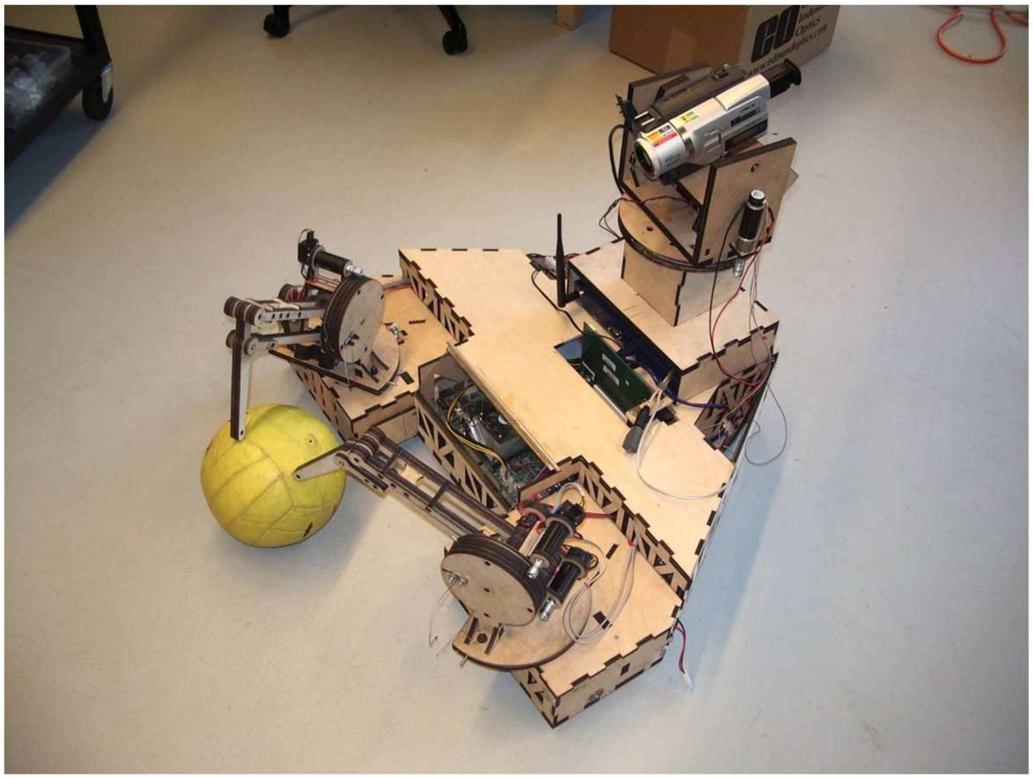
Linkage Material

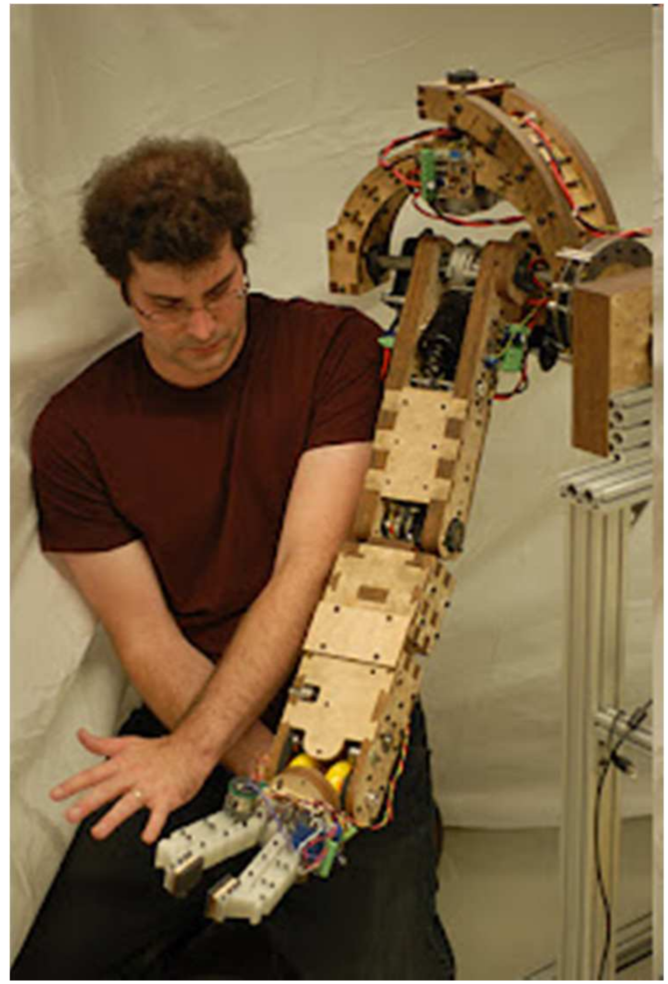
- Wood



Linkage Material

- Wood

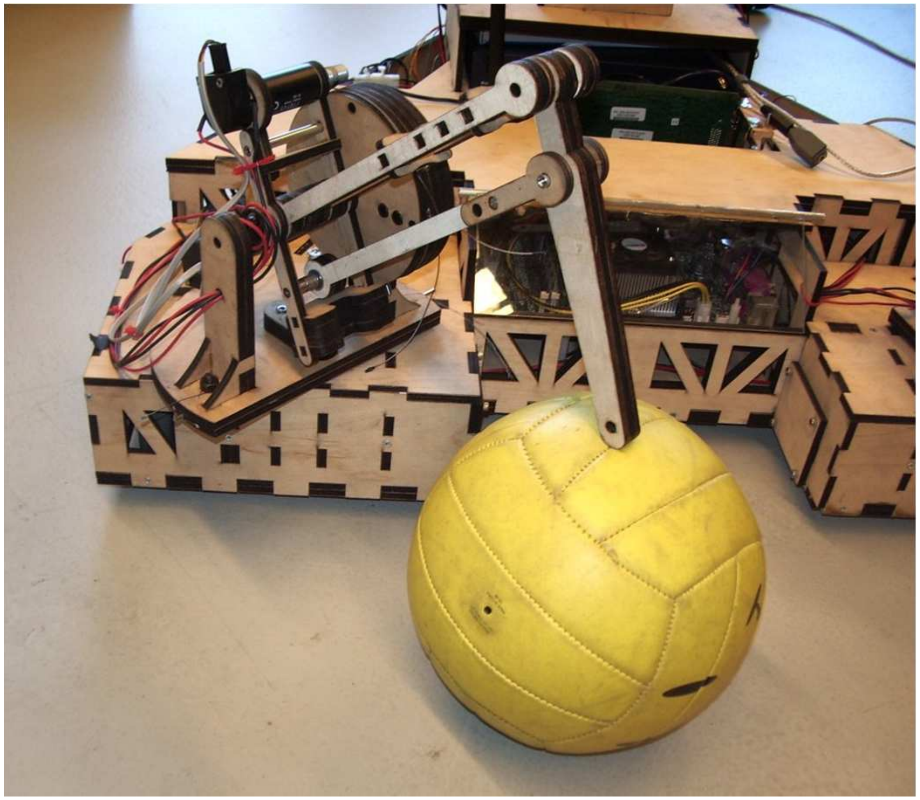


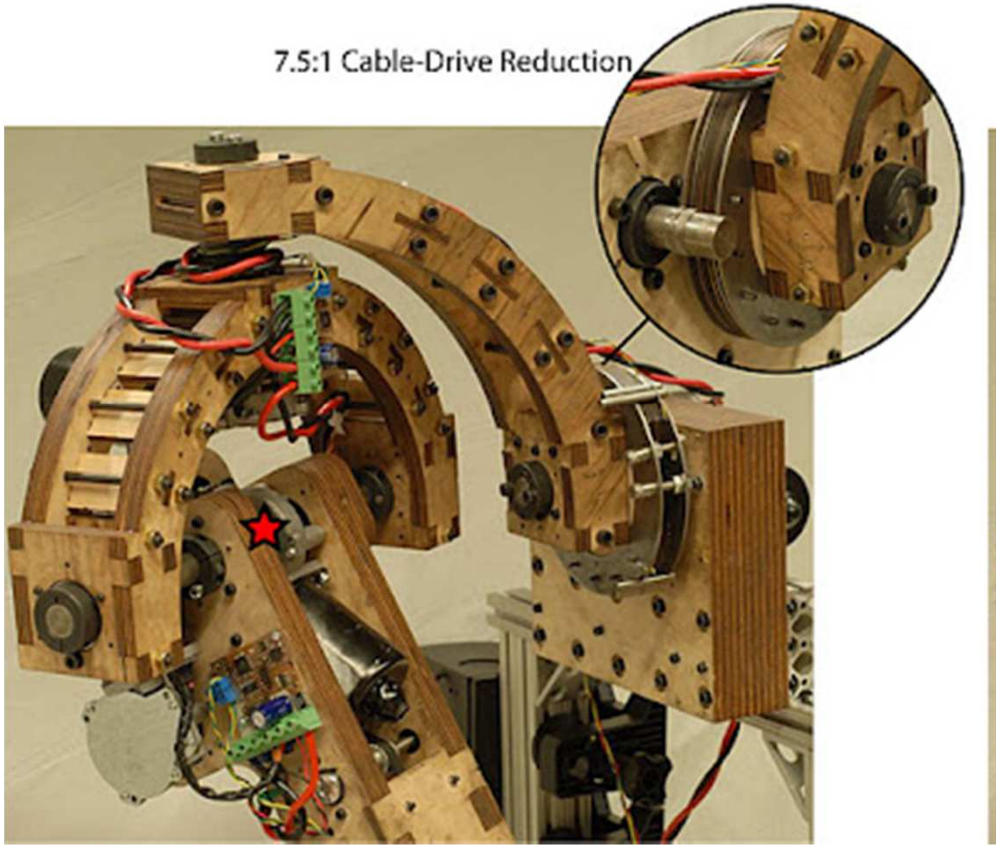


M. Quigley, R. Brewer, A. Y. Ng, K Salisbury, Stanford University

Linkage Material

- Wood





Linkage Material

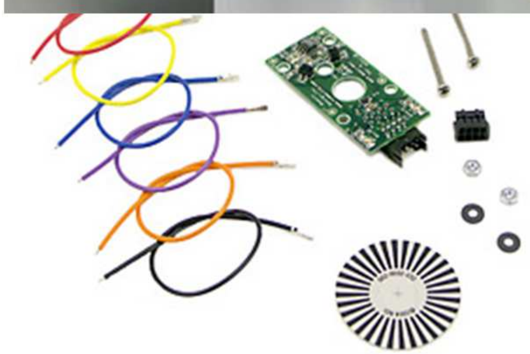
- Custom engineered from many materials

Intuitive surgical Da Vinci

Position/joint angle encoders

1. Optical

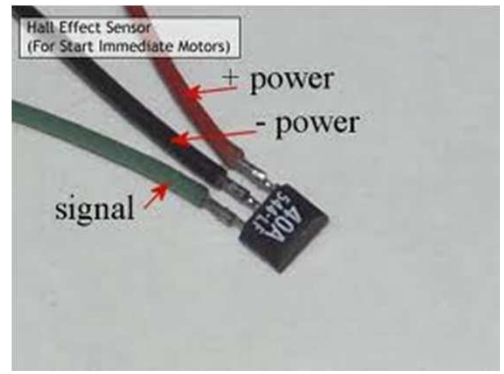
- Can be at motor or at joint
- Easy to make

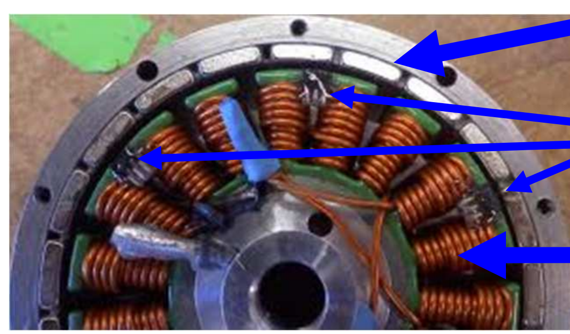


(3. Potentiometer)

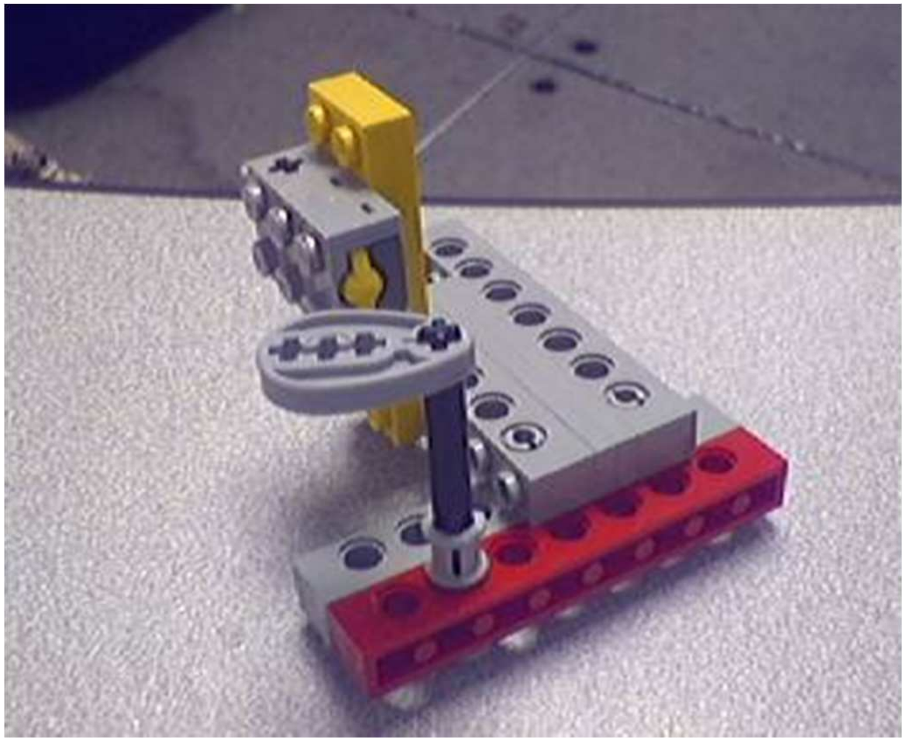
2. Hall effect magnetic

Embedded in motor





Physically counting rotations



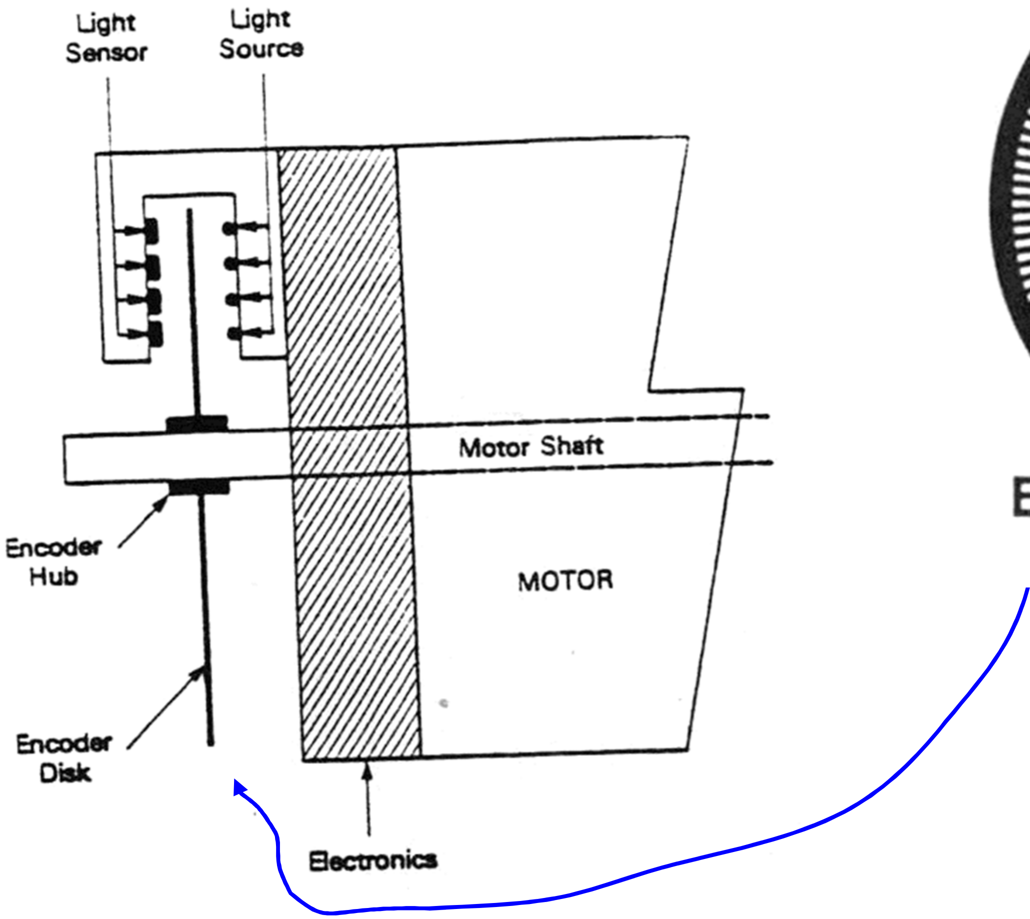
llk.media.mit.edu/projects/cricket/doc/LEGO-touchsensors.html

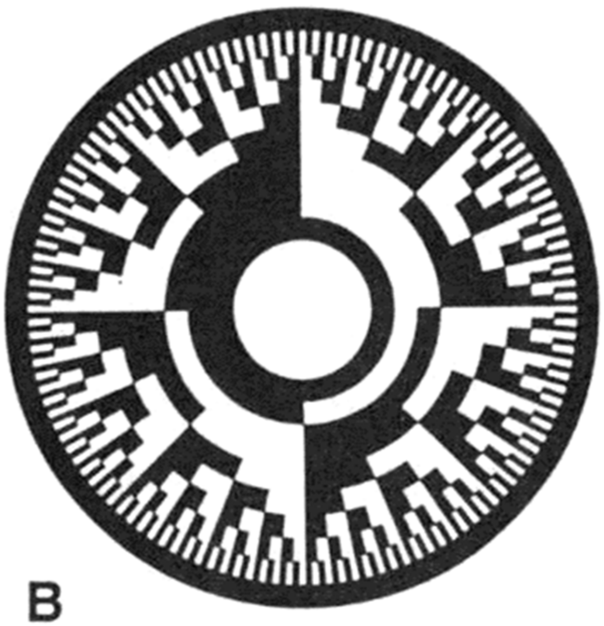
You can also use the touch sensor as a rotation sensor. Attach a cam to the rotating axle, and then position the LEGO sensor so that it is hit by the cam as the axle rotates. Counting the number of hits, combined with the wheel radius, can give you a distance reading. Measuring the time between hits can give you a speed measurement. Note that this only works well for axles that don't rotate very fast.

limited resolution… ?

Optical Encoders 1: *absolute*

- Detecting motor shaft orientation





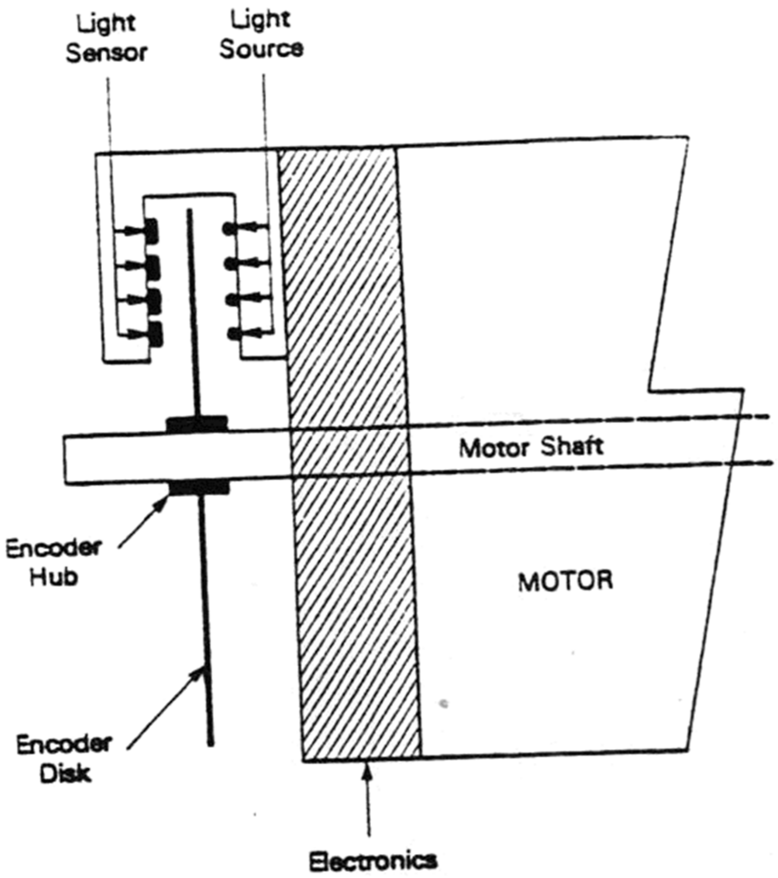
Binary encoding of shaft rotation

via light patterns

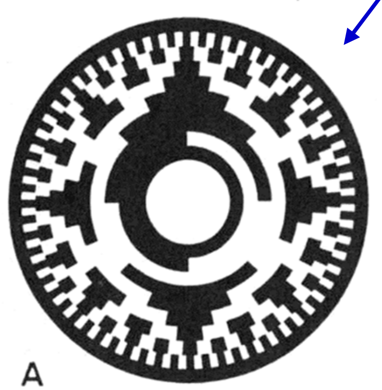
potential problems?

Optical Encoders 1: *absolute*

- Detecting motor shaft orientation



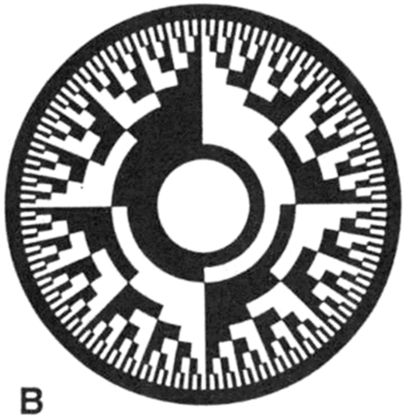
Alternative encodings are also popular… !

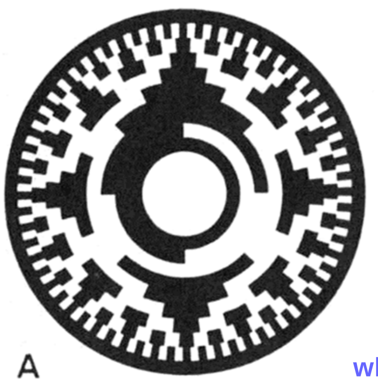


Gray Code

#	Binary	
0	0	000
1	1	001
2	10	011
3	11	010
4	100	110
5	101	111
6	110	101
7	111	100
8	1000	?
9	1001	?

What is important about each of these transitions?


B


A

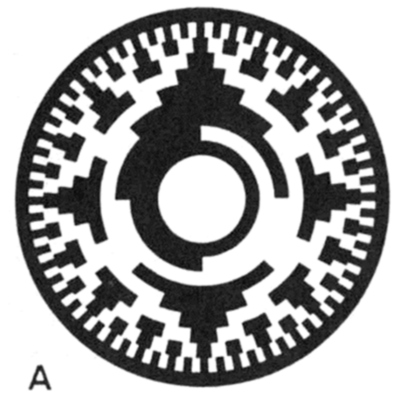
what comes next?

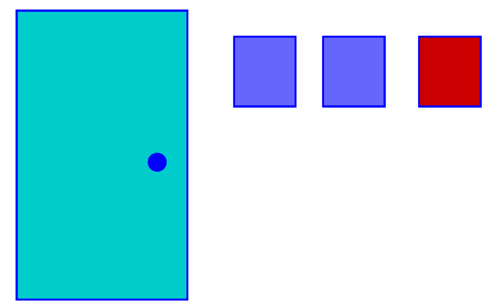
other applications?

Applications?

#	Binary	Gray
0	0	000
1	1	001
2	10	011
3	11	010
4	100	110
5	101	111
6	110	101
7	111	100
8	1000	1100
9	1001	1101

neighboring representations differ **by only 1 bit**





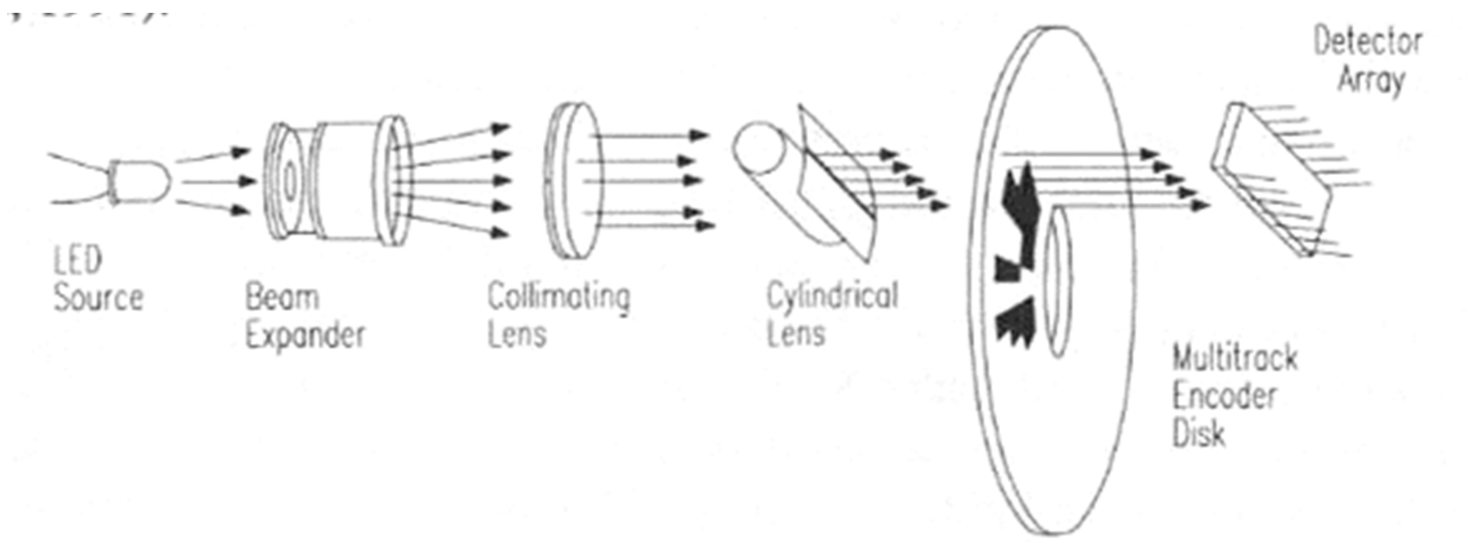
with important applications…

not found even in *The Official Duke Nukem 3d Strategies and Secrets ($1.94)*

wires?

Absolute Optical Encoders

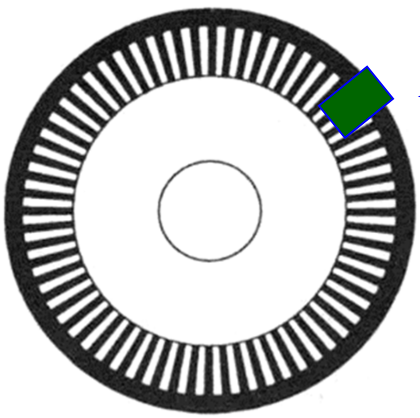
- Complexity of distinguishing many different states --
  high resolution is expensive!

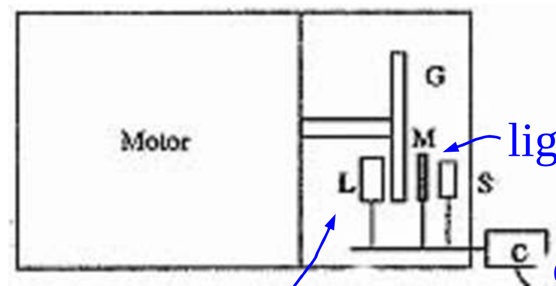


something simpler ?

Relative Optical Encoders

- Track position changes

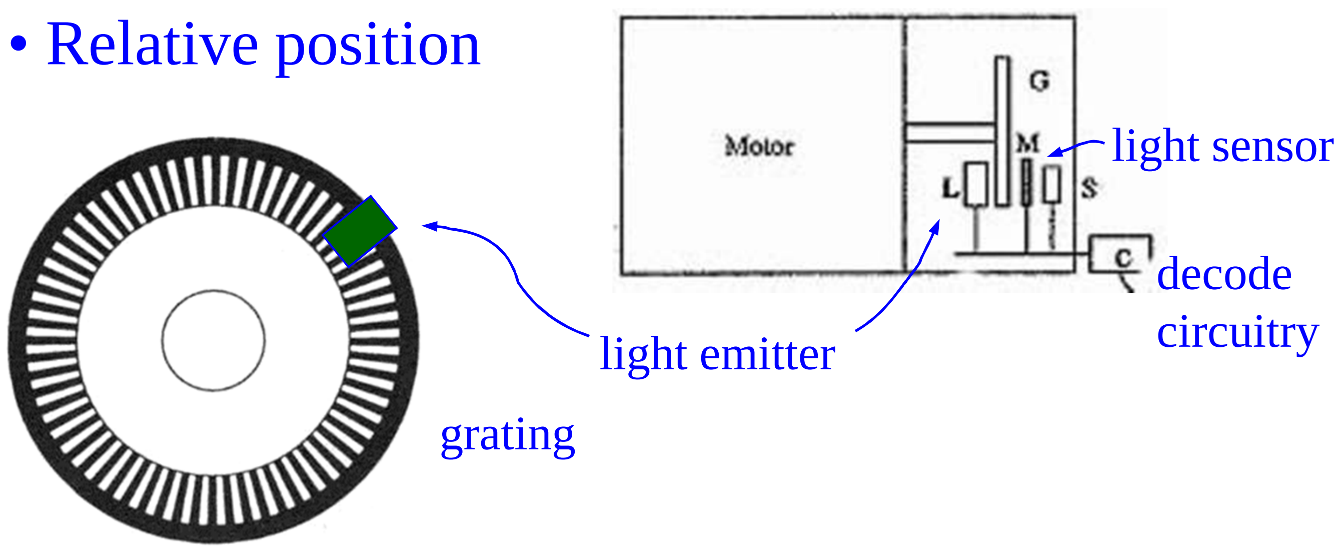




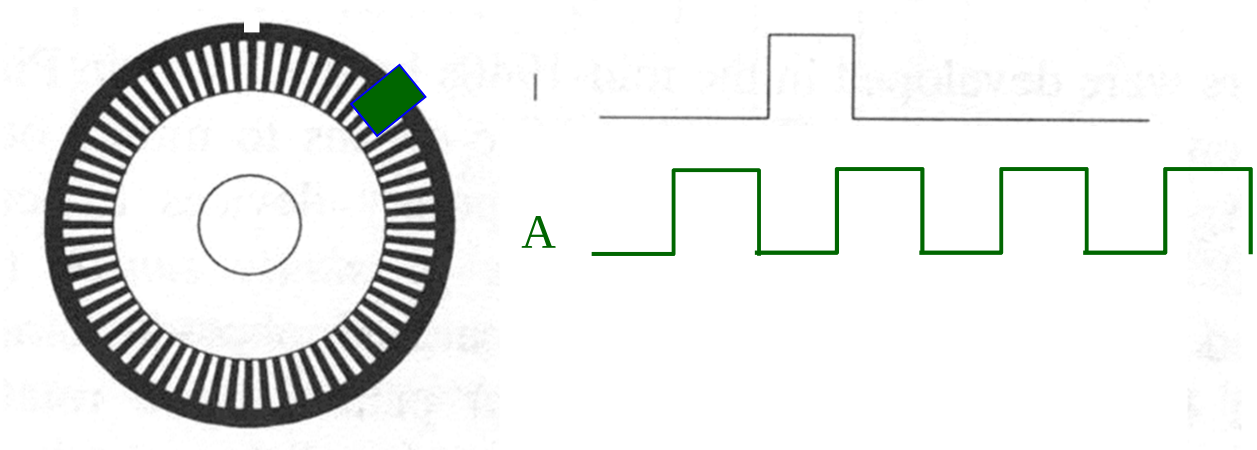
- calibration ?
- resolution ?
- direction ?

Relative Optical Encoders

- Relative position

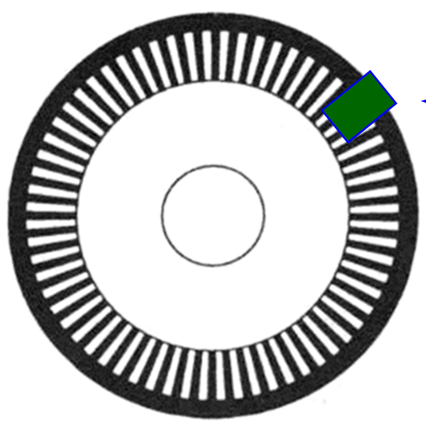


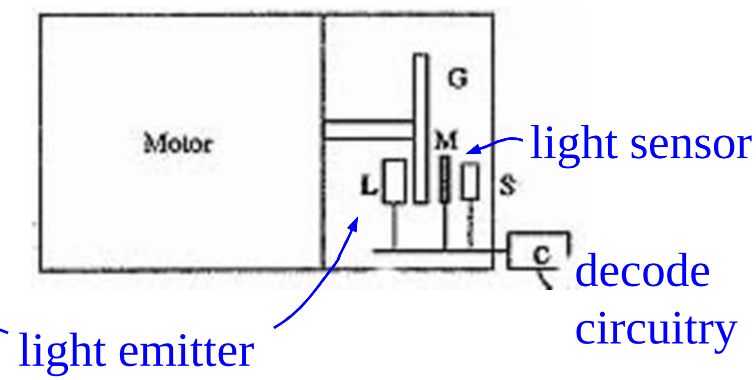
- calibration ?
- resolution ?
- direction ?



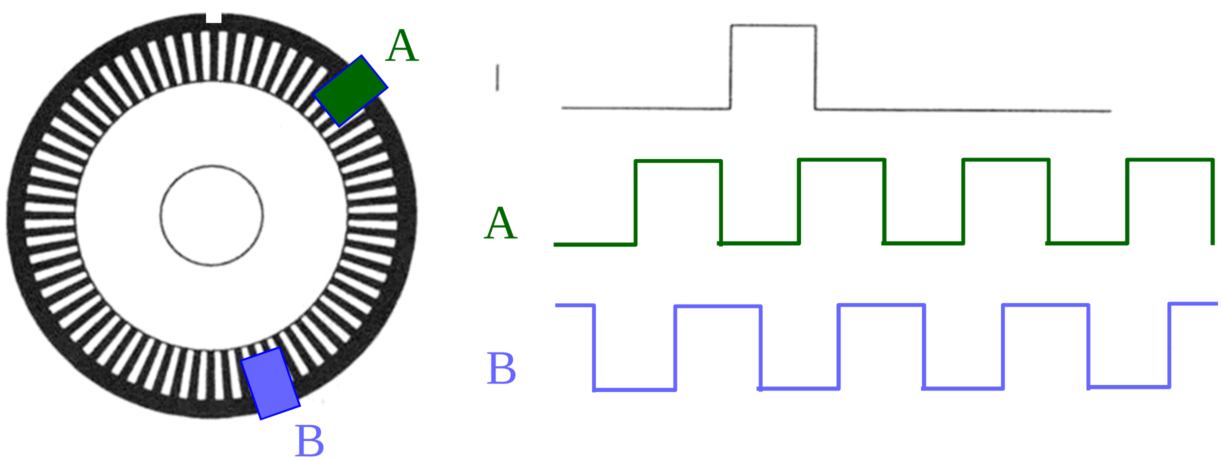
Relative Optical Encoders

- Relative position





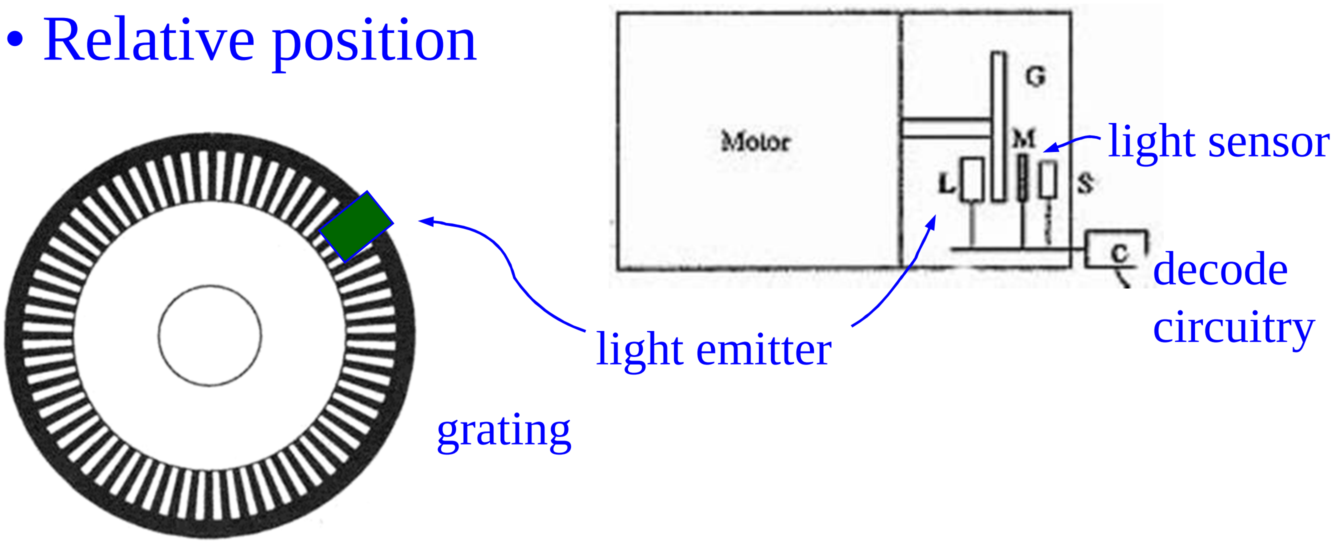
- calibration ?
- resolution ?
- direction ?



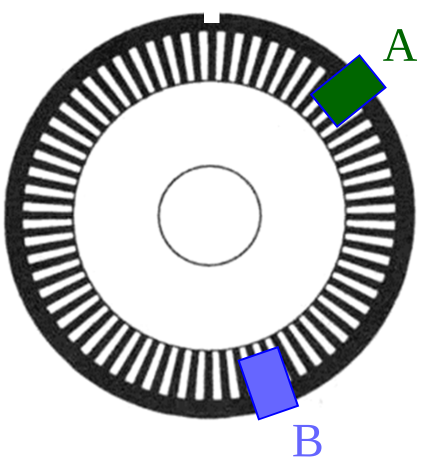
A vs. B

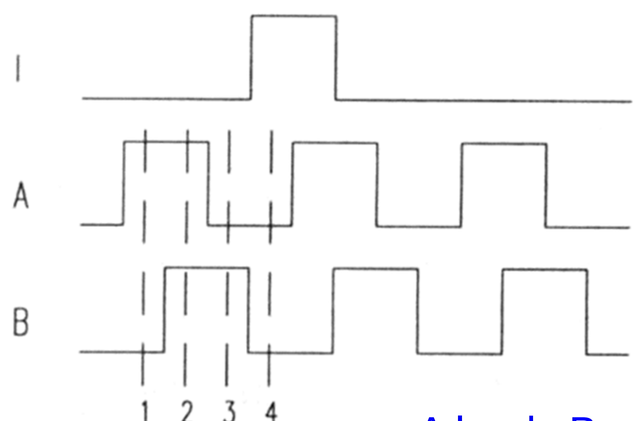
Quadrature encoding

- Relative position



- calibration ?
- resolution ?
- direction ?





quadrature encoding

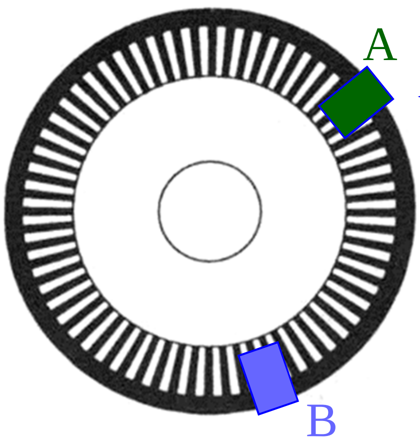
A leads B

State	Ch A	Ch B
S_1	High	Low
S_2	High	High
S_3	Low	High
S_4	Low	Low

Why not add more lines?

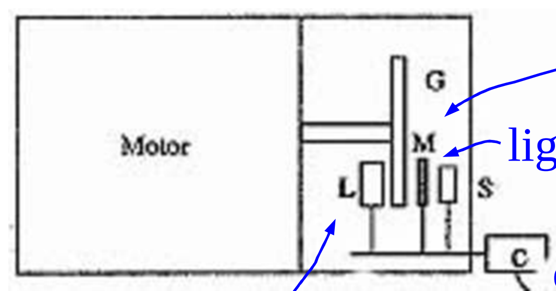
Relative Optical Encoders

- Relative position



light emitter

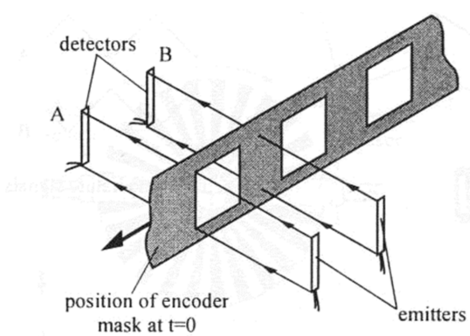
grating

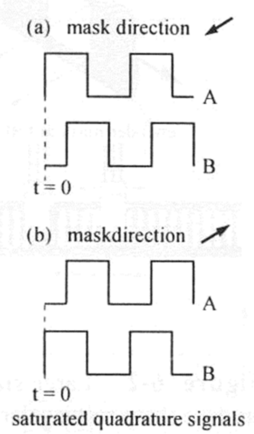


mask/diffuser

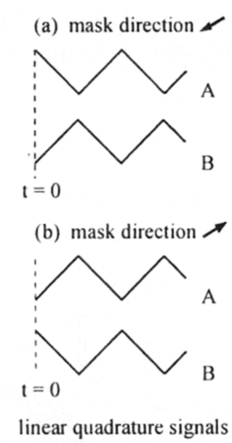
light sensor

decode circuitry





Ideal



Real

A diffuser tends to smooth these signals

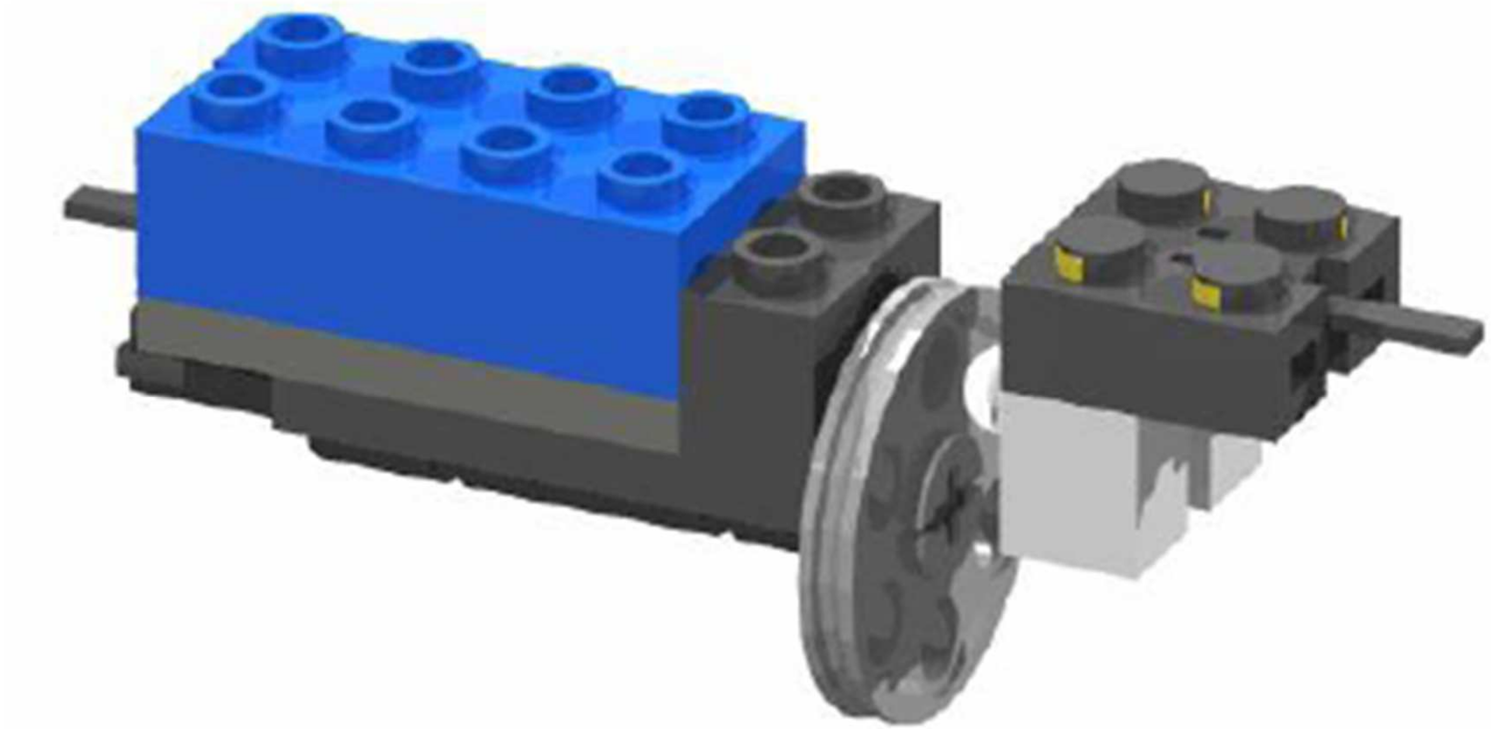
Two thresholds are used: ***hysteresis***

Another $40 word – free with the purchase of both "proprioceptive" and "exteroceptive"!

From proprio- to exteroceptivity!

First, a note on building materials...

Optical Encoders … in Lego!



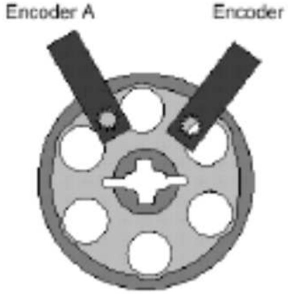
from the FIRST Lego League… only a *first* take

Course Name: Digital Control in Mechatronics
Course Number: EML 3804\EAS 3404

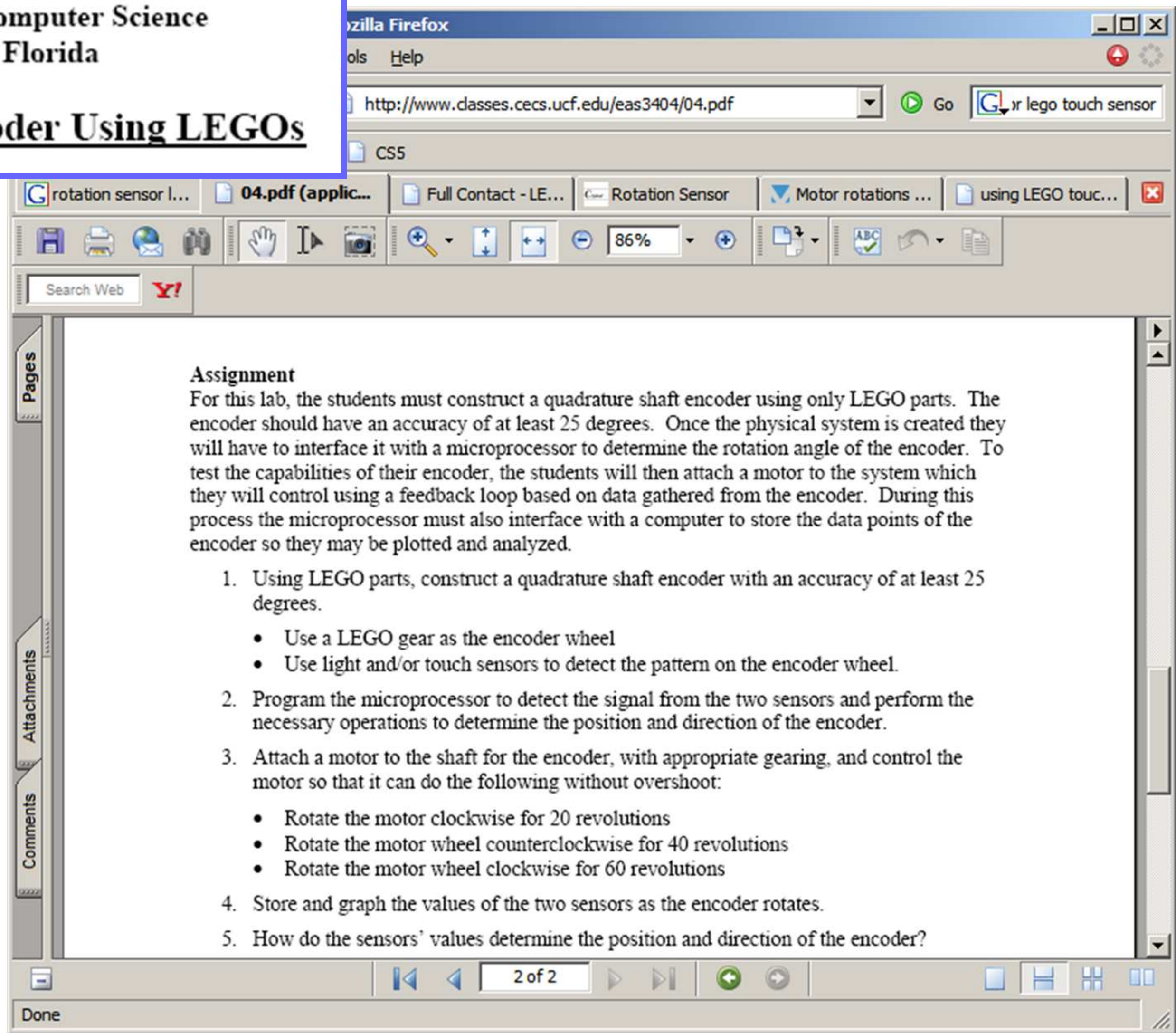
Alexander Leonessa
College of Engineering and Computer Science
University of Central Florida

<u>Design of a Quadrature Encoder Using LEGOs</u>

Lego-based quadrature encoding!



A LEGO pulley wheel may be used with two break-beam optosensors to build a quadrature encoder. The two optosensors must be placed so that they are 90 degrees out of phase in reading the position of the wheel. In the diagram, the "A" encoder is fully blocked, while the "B" encoder is in the transition between being blocked and being open.



Assignment

For this lab, the students must construct a quadrature shaft encoder using only LEGO parts. The encoder should have an accuracy of at least 25 degrees. Once the physical system is created they will have to interface it with a microprocessor to determine the rotation angle of the encoder. To test the capabilities of their encoder, the students will then attach a motor to the system which they will control using a feedback loop based on data gathered from the encoder. During this process the microprocessor must also interface with a computer to store the data points of the encoder so they may be plotted and analyzed.

1. Using LEGO parts, construct a quadrature shaft encoder with an accuracy of at least 25 degrees.
   - Use a LEGO gear as the encoder wheel
   - Use light and/or touch sensors to detect the pattern on the encoder wheel.
2. Program the microprocessor to detect the signal from the two sensors and perform the necessary operations to determine the position and direction of the encoder.
3. Attach a motor to the shaft for the encoder, with appropriate gearing, and control the motor so that it can do the following without overshoot:
   - Rotate the motor clockwise for 20 revolutions
   - Rotate the motor wheel counterclockwise for 40 revolutions
   - Rotate the motor wheel clockwise for 60 revolutions
4. Store and graph the values of the two sensors as the encoder rotates.
5. How do the sensors' values determine the position and direction of the encoder?

http://maven.smith.edu/~anevin/Robotics/FinalProject/Lecture05-AdvancedSensors.pdf

Computers

- Usually embedded microcontrollers
  - PIC
  - Arduino
  - Rasberry Pi
- Can use regular PC
  - Laptop
  - Mini-PC
  - Server (e.g. in PR2)
  - Off-board computing

Arduino

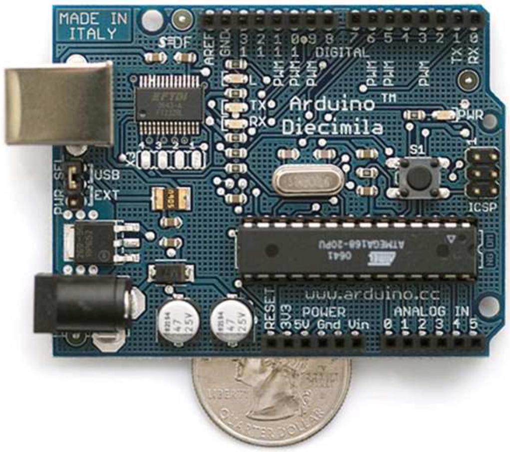
USB-based microcontroller interface board to:

- servomotors
- lots of sensors (not cameras)
- really anything ("low" bandwidth)

Comes with its own software (processing)

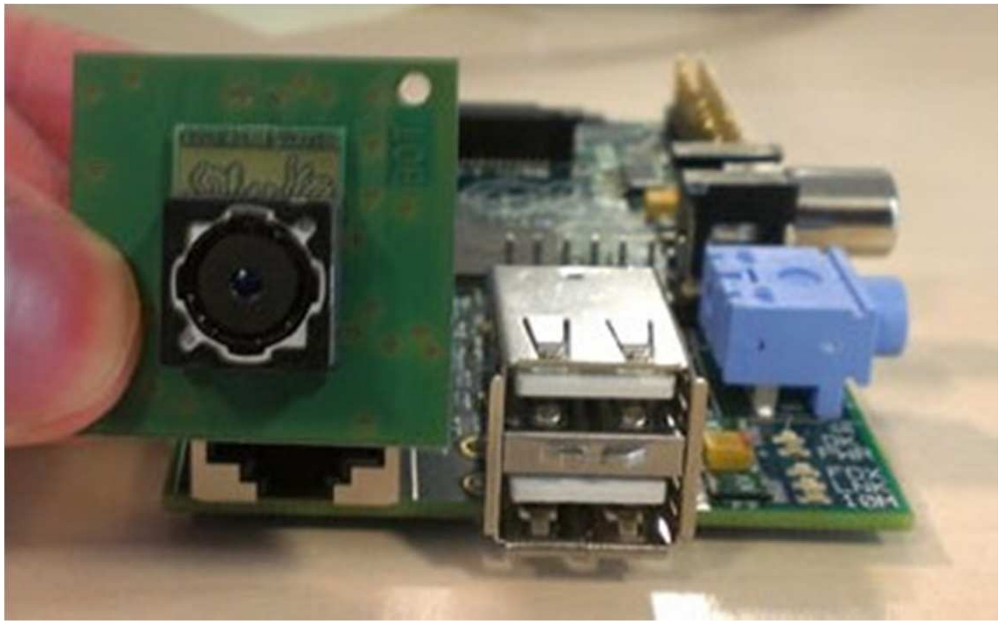
Plays well with serial communications

- can control it from any language
  - supported under Windows, Mac OX, Linux…



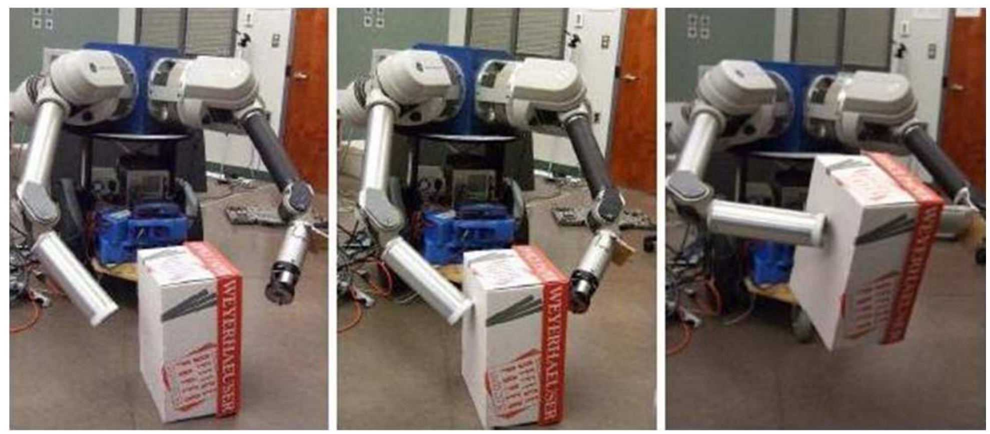
Raspberry-Pi

- $25-35
- 1.x GHz ARM w. FPU
- GPU
- Linux
- Like a 500Mhz PC
- Uses 2W power
- GPIO ports to connect:
  - Sensors
  - Motors
  - Camera!!
- Runs video processing!
- OpenCV

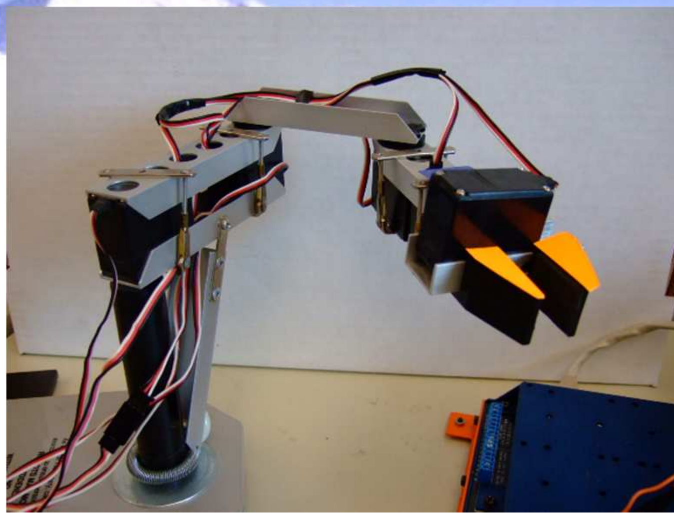


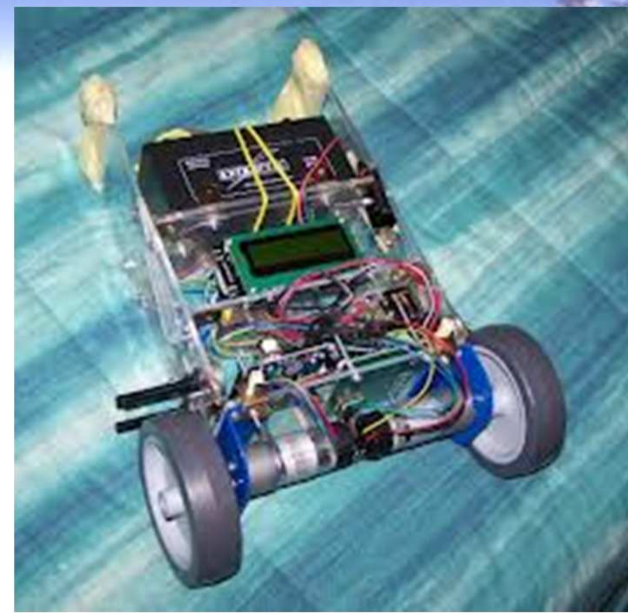
Robot categories summary

- Mobile ground robots
  - Inexpensive, easy
  - 2DOF
- Robot Arms Hands
  - Precise, expensive 6+ DOF
  - Manipulates objects
- Aerial/underwater
  - Challenging dynamics and control
  - UAV: Limited power and payload



Summary: Parts of a Robot





- Motors
- Motor controllers
- Transmission
- Linkages
- Sensors
- Computer

Upcoming:

- Kinematics: How do we model the relationship between:
  - how motors, joints and wheels turn (axle rotations: radians)
  - and how the robot moves (translation : x,y,z millimetre and rot)
- Control Paradigms: Reactive/Subsumption vs SPA
  - Simple local reactive control
  - Potential field motion control.