

BE1010 – Fall 2004

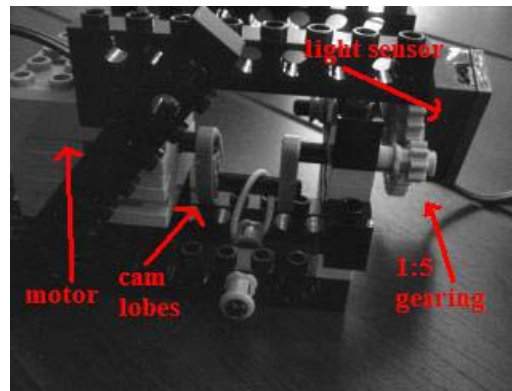
Prof. Mohamad Hassoun, WSU

Quiz # 10 (25 minutes. Open book/webpage)

Solve only one problem

- I. (10 points) Sketch and describe the operation of a stationary robot employing the step motor discussed in class last week. Your design should allow the RCX to control the position of a Lego Light Sensor such that it takes ambient light readings along five equally spaced angles, over a range of 360 degrees; i.e., the sensor takes measurements at 0° , 72° , 144° , 216° , and 288° . DO NOT employ a rotation sensor. **BONUS (5 points):** write and NQC program that controls the position of the Light Sensor, saves the five measurements in an *array*, and finally, display the average ambient light reading on the RCX's LCD.

```
task main()
{
  int i, x, avg, light[4];
  SetSensorType(SENSOR_1, SENSOR_TYPE_LIGHT);
  i=0;
  while(i < 4)
  {
    light[i]=SENSOR_1;
    OnFwd(OUT_A);
    Wait(100);
    Float(OUT_A);
    Wait(100);
    i=i+1;
  }
  Wait(100);
  x=0;
  for(i=0; i<4; i=i+1)
  {
    x=x+light[i];
  }
  avg=x/5;
  SetUserDisplay(avg,0);
  until(false);
}
```



- II. (10 points) Assume an RCX with a Light Sensor connected to its input port #1. The Light Sensor is mounted on the RCX. You are to hold the RCX and move it around (slowly, for a period of 5 seconds) in a room so as to sample the light intensities around the room. Write an NQC program that will save 50 light sensor readings in an *array*. Then, your program should compute and display (continuously on the LCD, one after the other) computed maximum, average, and minimum light intensities.

```
task main()
{
  int i, x, avg, max, min, light[49];
  SetSensorType(SENSOR_1, SENSOR_TYPE_LIGHT);
  i=0;
  ClearTimer (0);
  while(Timer(0) < 50)
  {
    light[i]=SENSOR_1;
    Wait(10);
    i=i+1;
  }
  Wait(100);
  x=0;
  for(i=0; i<49; i=i+1)
  {
    x=x+light[i];
  }
  avg=x/50;
  min=light[0];
  for(i=0; i<49; i=i+1)
  {
    if (light[i] < min)
    {
      min=light[i];
    }
  }
  max=light[0];
  for(i=0; i<49; i=i+1)
  {
    if (light[i] > max)
    {
      max=light[i];
    }
  }
  while(true)
  {
    SetUserDisplay(max, 0);
    Wait(200);
    SetUserDisplay(avg, 0);
    Wait(200);
    SetUserDisplay(min, 0);
    Wait(200);
  }
}
```