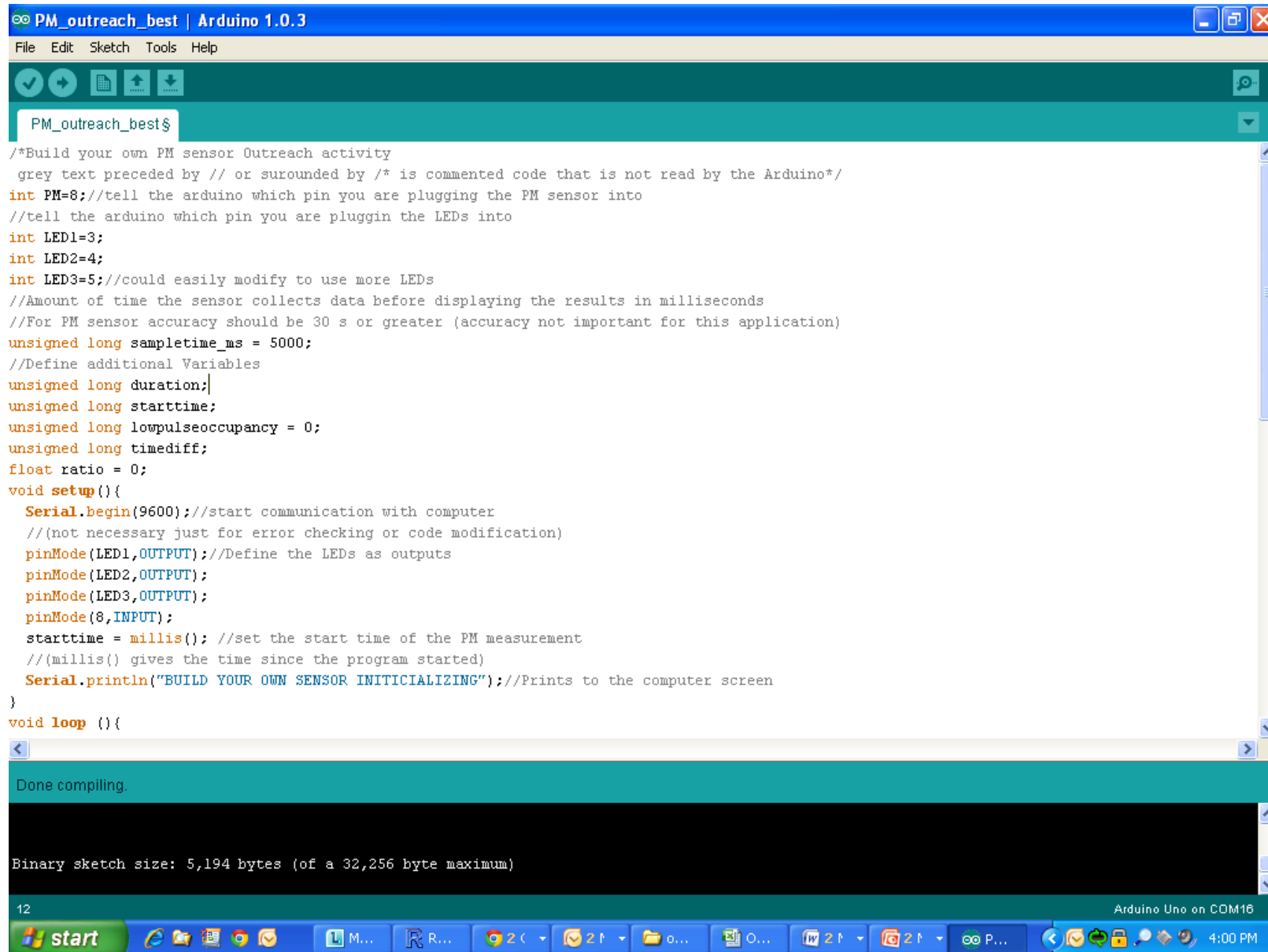


## Arduino Code for “Build Your Own Particle Sensor” Activity

### Arduino Code – Page 1



```
PM_outreach_best | Arduino 1.0.3
File Edit Sketch Tools Help

PM_outreach_best$
/*Build your own PM sensor Outreach activity
 grey text preceded by // or surrounded by /* is commented code that is not read by the Arduino*/
int PM=8;//tell the arduino which pin you are plugging the PM sensor into
//tell the arduino which pin you are pluggin the LEDs into
int LED1=3;
int LED2=4;
int LED3=5;//could easily modify to use more LEDs
//Amount of time the sensor collects data before displaying the results in milliseconds
//For PM sensor accuracy should be 30 s or greater (accuracy not important for this application)
unsigned long sampletime_ms = 5000;
//Define additional Variables
unsigned long duration;
unsigned long starttime;
unsigned long lowpulseoccupancy = 0;
unsigned long timediff;
float ratio = 0;
void setup(){
  Serial.begin(9600);//start communication with computer
  //(not necessary just for error checking or code modification)
  pinMode(LED1,OUTPUT);//Define the LEDs as outputs
  pinMode(LED2,OUTPUT);
  pinMode(LED3,OUTPUT);
  pinMode(8,INPUT);
  starttime = millis(); //set the start time of the PM measurement
  //(millis() gives the time since the program started)
  Serial.println("BUILD YOUR OWN SENSOR INITIALIZING");//Prints to the computer screen
}
void loop (){

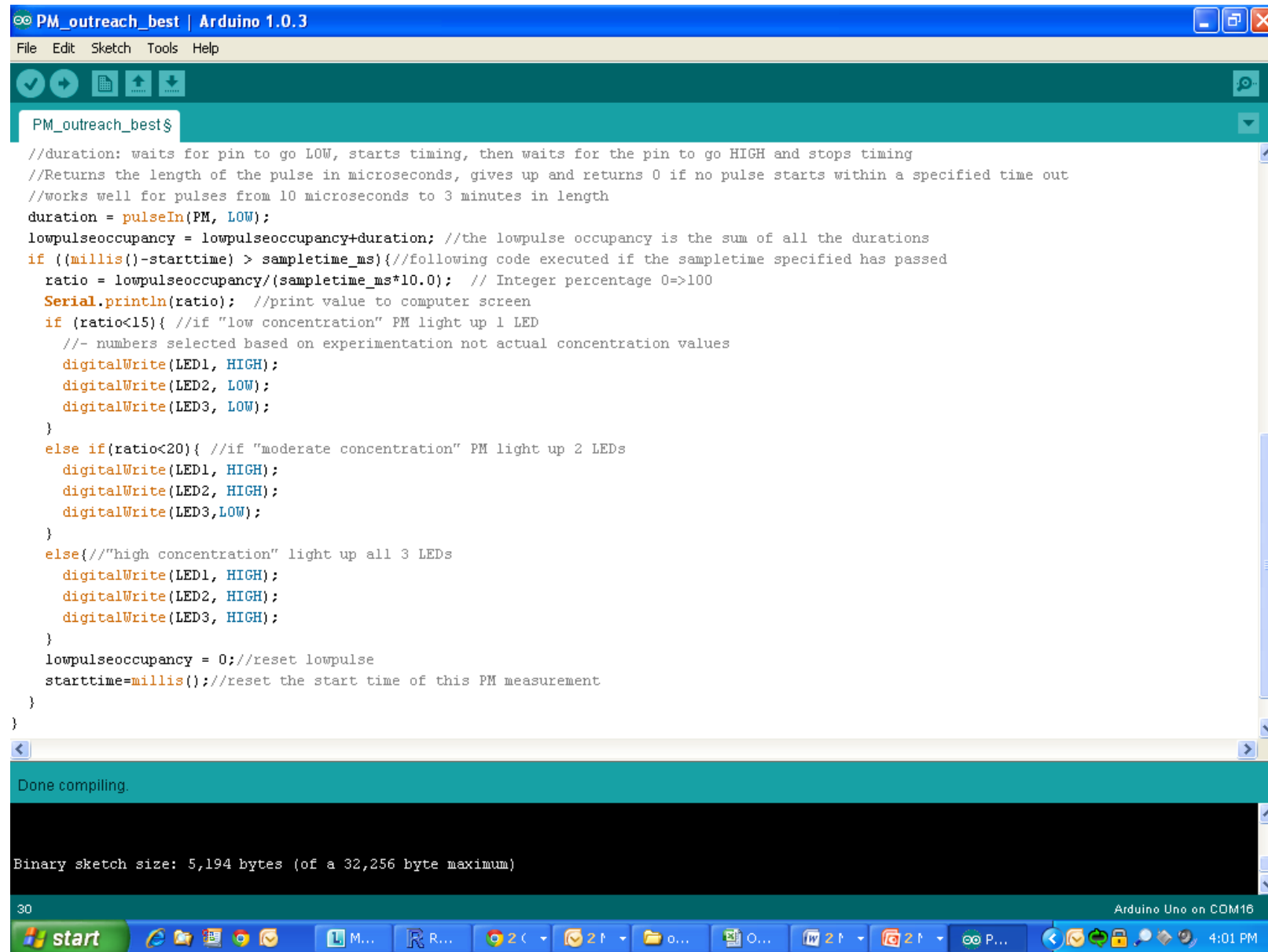
Done compiling.

Binary sketch size: 5,194 bytes (of a 32,256 byte maximum)

12 Arduino Uno on COM16
start M... R... 2 C 2 P O... W 2 P 2 P P... 4:00 PM
```

## Arduino Code for “Build Your Own Particle Sensor” Activity

### Arduino Code – Page 2



```
PM_outreach_best | Arduino 1.0.3
File Edit Sketch Tools Help

PM_outreach_best$
//duration: waits for pin to go LOW, starts timing, then waits for the pin to go HIGH and stops timing
//Returns the length of the pulse in microseconds, gives up and returns 0 if no pulse starts within a specified time out
//works well for pulses from 10 microseconds to 3 minutes in length
duration = pulseIn(PM, LOW);
lowpulseoccupancy = lowpulseoccupancy+duration; //the lowpulse occupancy is the sum of all the durations
if ((millis()-starttime) > sampletime_ms){//following code executed if the sampletime specified has passed
  ratio = lowpulseoccupancy/(sampletime_ms*10.0); // Integer percentage 0=>100
  Serial.println(ratio); //print value to computer screen
  if (ratio<15){ //if "low concentration" PM light up 1 LED
    //- numbers selected based on experimentation not actual concentration values
    digitalWrite(LED1, HIGH);
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, LOW);
  }
  else if(ratio<20){ //if "moderate concentration" PM light up 2 LEDs
    digitalWrite(LED1, HIGH);
    digitalWrite(LED2, HIGH);
    digitalWrite(LED3,LOW);
  }
  else{//"high concentration" light up all 3 LEDs
    digitalWrite(LED1, HIGH);
    digitalWrite(LED2, HIGH);
    digitalWrite(LED3, HIGH);
  }
  lowpulseoccupancy = 0;//reset lowpulse
  starttime=millis();//reset the start time of this PM measurement
}
}
```

Done compiling.

Binary sketch size: 5,194 bytes (of a 32,256 byte maximum)

30 Arduino Uno on COM16

start M... R... 2 C 2 P O... W 2 P 2 P P... 4:01 PM