

/Based of the wiring code at <http://wiring.org.co/learning/basics/humiditytemperaturesht15.html>

```
int SHT_clockPin = 3; // pin used for clock
int SHT_dataPin = 2; // pin used for data

void setup(){
  Serial.begin(9600); // open serial at 9600 bps
}

void loop(){
  //these can take a bit to get the values (100ms or so)
  float temperature = getTemperature();
  float humidity = getHumidity();

  Serial.print(temperature);
  Serial.print(" | ");
  Serial.println(humidity);
}

float getTemperature(){
  //Return Temperature in Celsius
  SHT_sendCommand(B00000011, SHT_dataPin, SHT_clockPin);
  SHT_waitForResult(SHT_dataPin);

  int val = SHT_getData(SHT_dataPin, SHT_clockPin);
  SHT_skipCrc(SHT_dataPin, SHT_clockPin);
  return (float)val * 0.01 - 40; //convert to celsius
}

float getHumidity(){
  //Return Relative Humidity
  SHT_sendCommand(B00000101, SHT_dataPin, SHT_clockPin);
  SHT_waitForResult(SHT_dataPin);
  int val = SHT_getData(SHT_dataPin, SHT_clockPin);
  SHT_skipCrc(SHT_dataPin, SHT_clockPin);
  return -4.0 + 0.0405 * val + -0.0000028 * val * val;
}

void SHT_sendCommand(int command, int dataPin, int clockPin){
  // send a command to the SHTx sensor
  // transmission start
  pinMode(dataPin, OUTPUT);
  pinMode(clockPin, OUTPUT);
  digitalWrite(dataPin, HIGH);
  digitalWrite(clockPin, HIGH);
  digitalWrite(dataPin, LOW);
}
```

```

digitalWrite(clockPin, LOW);
digitalWrite(clockPin, HIGH);
digitalWrite(dataPin, HIGH);
digitalWrite(clockPin, LOW);

// shift out the command (the 3 MSB are address and must be 000, the last 5 bits are the command)
shiftOut(dataPin, clockPin, MSBFIRST, command);

// verify we get the right ACK
digitalWrite(clockPin, HIGH);
pinMode(dataPin, INPUT);

if (digitalRead(dataPin)) Serial.println("ACK error 0");
digitalWrite(clockPin, LOW);
if (!digitalRead(dataPin)) Serial.println("ACK error 1");
}

```

```

void SHT_waitForResult(int dataPin){
// wait for the SHTx answer
pinMode(dataPin, INPUT);

int ack; //acknowledgement

//need to wait up to 2 seconds for the value
for (int i = 0; i < 1000; ++i){
  delay(2);
  ack = digitalRead(dataPin);
  if (ack == LOW) break;
}

if (ack == HIGH) Serial.println("ACK error 2");
}

```

```

int SHT_getData(int dataPin, int clockPin){
// get data from the SHTx sensor

// get the MSB (most significant bits)
pinMode(dataPin, INPUT);
pinMode(clockPin, OUTPUT);
byte MSB = shiftIn(dataPin, clockPin, MSBFIRST);

// send the required ACK
pinMode(dataPin, OUTPUT);
digitalWrite(dataPin, HIGH);
digitalWrite(dataPin, LOW);
digitalWrite(clockPin, HIGH);
digitalWrite(clockPin, LOW);

// get the LSB (less significant bits)
pinMode(dataPin, INPUT);
byte LSB = shiftIn(dataPin, clockPin, MSBFIRST);
return ((MSB << 8) | LSB); //combine bits
}

```

```
void SHT_skipCrc(int dataPin, int clockPin){  
  // skip CRC data from the SHTx sensor  
  pinMode(dataPin, OUTPUT);  
  pinMode(clockPin, OUTPUT);  
  digitalWrite(dataPin, HIGH);  
  digitalWrite(clockPin, HIGH);  
  digitalWrite(clockPin, LOW);  
}
```