**MỤC LỤC**

[**DANH MỤC HÌNH ẢNH 3**](#_Toc41948965)

[**LỜI MỞ ĐẦU 4**](#_Toc41948990)

[**CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI. 5**](#_Toc41948991)

[**1.1 Lý do chọn đề tài. 5**](#_Toc41948992)

[**1.2 Giới thiệu tổng bộ. 5**](#_Toc41948993)

[**1.3 Nội dung nghiên cứu đề tài. 5**](#_Toc41948994)

[**CHƯƠNG 2: CƠ SỞ LÝ THUYẾT. 7**](#_Toc41948995)

[**2.1 Tổng quan về Arduino. 7**](#_Toc41948996)

[**2.1.1 Lịch sử phát triển. 7**](#_Toc41948997)

[**2.1.2 Giới thiệu chung về Arduino. 8**](#_Toc41948998)

[**2.1.3 Phần cứng của Arduino 9**](#_Toc41948999)

[**2.2 Cấu tạo của Arduino Uno R3. 11**](#_Toc41949000)

[**2.2.1 Giới thiệu. 11**](#_Toc41949001)

[**2.2.2 Thông số kĩ thuật. 12**](#_Toc41949002)

[**2.2.3 Vi điều khiển của Arduino uno R3. 12**](#_Toc41949003)

[**2.2.4 Năng lượng. 13**](#_Toc41949004)

[**2.2.5 Bộ nhớ. 13**](#_Toc41949005)

[**2.2.6 Cổng vào ra . 14**](#_Toc41949006)

[**2.3 Mạng Zigbee. 15**](#_Toc41949007)

[**2.3.1 Khái niệm về Zigbee. 16**](#_Toc41949008)

[**2.3.2 Các thành phần chính trong mạng Zigbee. 17**](#_Toc41949009)

[**2.3.3 Cấu trúc, liên kết mạng Zigbee. 18**](#_Toc41949010)

[**2.3.4 Ứng dụng của Zigbee. 19**](#_Toc41949011)

[**2.4 Module Zigbee CC2530. 19**](#_Toc41949012)

[**2.5 Cảm biến nhiệt độ, độ ẩm DHT 11. 20**](#_Toc41949013)

[**2.6 Các công cụ lập trình. 22**](#_Toc41949014)

[**2.6.1 Phần mềm lập trình Arduino. 22**](#_Toc41949015)

[**2.6.2 Phần mềm Visual Basic. 23**](#_Toc41949017)

[**CHƯƠNG 3: THIẾT KẾ VÀ THỬ NGHIỆM. 25**](#_Toc41949018)

[**3.1 Yêu cầu thiết bị. 25**](#_Toc41949019)

[**3.2 Thiết kế phần cứng. 26**](#_Toc41949020)

[**3.3 Thiết kế phần mềm. 27**](#_Toc41949021)

[**KẾT LUẬN 31**](#_Toc41949022)

[**TÀI LIỆU THAM KHẢO 32**](#_Toc41949023)

**DANH MỤC HÌNH ẢNH**

[Hình 2.1 những thành viên khởi sướng Arduino. 6](#_Toc41947984)

[Hình 2.2 Một mạch Arduino Uno chính thức với các mô tả về các cổng I/O. 9](#_Toc41947985)

[Hình 2.3 Ảnh Arduino Uno. 10](#_Toc41947986)

[Hình 2.4 Vi điều khiển của Arduino Uno R3. 11](#_Toc41947987)

[Hình 2.5 Các chân của Arduino Uno R3. 13](#_Toc41947988)

[Hình 2.6 Mạng Zigbee hình tổ ong. 15](#_Toc41947989)

[Hình 2.7 Ba loại mạng Zigbee chính. 17](#_Toc41947990)

[Hình 2.8 Module Zigbee CC 2530. 18](#_Toc41947991)

[Hình 2.9 Sơ đồ chân Module Zigbee CC 2530. 19](#_Toc41947992)

[Hình 2.10 Cảm biến nhiệt độ, độ ẩm DHT 11. 19](#_Toc41947993)

[Hình 2.11 Các chân của DHT 11. 20](#_Toc41947994)

[Hình 2.12 Cách nối chân DHT 11 vào board mạch Arduino UNO R3. 20](#_Toc41947995)

[Hình 2.13 Giao diện của Arduino IDE. 21](#_Toc41947996)

[Hình 2.14 Chức năng các công cụ. 22](#_Toc41947997)

[Hình 2.15 Vùng thông báo Arduino IDE. 22](#_Toc41947998)

[Hình 2.16 Giao diện chương trình Visual basic. 23](#_Toc41947999)

[Hình 3. 1 Một node truyền của hệ thống giám sát nhiệt độ độ ẩm sử dụng mạng Zigbee. 24](#_Toc41948000)

[Hình 3. 2 Một node nhận của hệ thống giám sát nhiệt độ độ ẩm sử dụng mạng Zigbee. 24](#_Toc41948001)

[Hình 3. 3 Sơ đồ khối hệ thống giám sát nhiệt độ độ ẩm môi trường sử dụng mạng Zigbee. 25](file:///C:\Users\User\Desktop\Student%20present%20topic\nhóm%201.docx#_Toc41948002)

[Hình 3. 4 Hoàn thành 2 node truyền và node nhận kết nối và hiển thị lên máy tính. 25](#_Toc41948003)

[Hình 3. 5 Lưu đồ thuật toán. 26](file:///C:\Users\User\Desktop\Student%20present%20topic\nhóm%201.docx#_Toc41948004)

[Hình 3. 6 Giao diện phần mềm thiết lập cấu hình Module Zigbee. 27](#_Toc41948005)

[Hình 3. 7 Thiết lập cấu hình cho Module Zigbee. 27](#_Toc41948006)

[Hình 3. 8 Giao diện chạy hiển thị trên máy tính. 30](#_Toc41948007)

# 

# **LỜI MỞ ĐẦU**

Ngày nay, sự phát triển mạnh mẽ của khoa học đời sống, cuộc sống của con người đã thay đổi ngày một tốt hơn, với những trang thiết bị hiện đại phục vụ trong công cuộc công nghiệp hóa, hiện đại hóa. Đặc biệt góp phần không nhỏ đó là ngành kĩ thuật điện – điện tử trong sự nghiệp xây dựng đất nước. Những thiết bị điện, điện tử được phát triển và ứng dụng rộng rãi trong đời sống hằng ngày. Từ những thời gian đầu phát triển vi điều khiển đã cho thấy sự ưu việt của nó và cho tới ngày nay tính ưu việt đó ngày càng được khẳng định thêm. Những thành tựu của nó đã có thể biến được những cái tưởng chừng như không thể thành những cái có thể, góp phần nâng cao đời sống vật chất và tinh thần cho con người.

Để góp phần làm sáng tỏ hiệu quả của những ứng dụng trong thực tế của môn vi điều khiển, sau một thời gian học tập được các thầy cô trong khoa giảng dạy về các kiến thức chuyên ngành, đồng thời được sự giúp đỡ nhiệt tình của các thầy cô trong khoa Điện - Điện tử, cùng với sự lỗ lực của bản thân, nhóm chúng em đã Thiết kế “ module giám sát nhiệt độ, độ ẩm môi trường ”nhưng do thời gian, kiến thức và kinh nghiệm của chúng em còn có hạn nên sẽ không thể tránh khỏi những sai sót. Chúng Em rất mong được sự giúp đỡ và tham khảo ý kiến của thầy cô và các bạn nhằm đóng góp phát triển thêm đề tài.

Chúng Em xin chân thành cảm ơn thầy Bùi Như Phong đã tận tình chỉ bảo và giúp đỡ Chúng em hoàn thành bài tập lớn này. Do kiến thức còn hạn chế trong quá trình thực hiện chúng em không tránh khỏi những sai sót kính mong quý thầy cô bỏ qua và giúp đỡ chúng em. Em rất mong được sự đóng góp của thầy cô và các bạn để nội dung đề tài này ngày càng hoàn thiện hơn.

# **CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI.**

* 1. **Lý do chọn đề tài.**

Ngày nay với những ứng dụng khoa học tiên tiến, cả thế giới của chúng ta ngày một hiện đại và văn minh hơn. Sự phát triển của ngành kĩ thuật điện tử đã tạo ra hàng loạt những thiết bị với sự chính xác cao, tốc độ nhanh, gọn nhẹ đã đóng vai trò vô cùng quan trọng trong cuộc sống của con người, là chìa khóa đi vào công nghệ hiện đại. Máy móc dần được thay thế sức lao động của con người, tự động hóa, điều khiển đóng vai trò vô cùng quan trọng trong công nghiệp, quản lí.

Điện tử đang trở thành một ngành khoa học đa nhiệm vụ. Bài toán kiểm soát nhiệt độ, độ ẩm đã và đang không ngừng đáp ứng được nhu cầu của các nghành nông- lâm- ngư nghiệp đến các nhu cầu trong cuộc sống. Một trong những ứng dụng quan trọng của công nghệ Điện tử là kỹ thuật điều khiển từ xa. Nó đã góp phần rất lớn trong việc điều khiển các thiết bị phục vụ nhu cầu sản xuất.

Chính vì vậy nhóm em đã chọn đề tài “ Thiết kế module giám sát nhiệt độ, độ ẩm môi trường sử dụng mạng Zigbee ”.

* 1. **Giới thiệu tổng bộ.**

Thiết kế module giám sát nhiệt độ, độ ẩm môi trường bằng cách sử dụng mạng Zigbee giúp cho ta dễ dàng quan sát, xử lý dữ liệu nhiệt độ, độ ẩm mà cảm biến gửi về qua môi trường mạng không dây.

Ta có thể nắm bắt được số liệu thông qua giao diện hiển thị trên máy tính được kết nối với module Zigbee thu dữ liệu từ các module Zigbee phát truyền về. Giao diện sẽ hiển thị số liệu, đồng thời biểu thị trên đồ thị để ta có thể dễ dàng nhìn nhận và quan sát sự chênh lệch nhiệt độ, độ ẩm.

* 1. **Nội dung nghiên cứu đề tài.**
* Tìm hiểu, nghiên cứu về mạng Zigbee.
* Nghiên cứu cấu trúc tổng quan của Arduino Uno R3.
* Xây dựng sơ đồ nguyên lý thuật toán chi tiết.
* Lập lưu đồ thuật toán và viết chương trình phần mềm.
* Thiết kế phần cứng.

# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.**

## 2.1 Tổng quan về Arduino.

### **2.1.1 Lịch sử phát triển.**

Arduino được khởi động vào năm 2005 như là một dự án dành cho sinh viên trại Interaction Design Institute Ivrea (Viện thiết kế tương tác Ivrea) tại Ivrea, Italy. Vào thời điểm đó các sinh viên sử dụng một "BASIC Stamp" (con tem Cơ Bản) có giá khoảng $100, xem như giá dành cho sinh viên. Massimo Banzi, một trong những người sáng lập, giảng dạy tại Ivrea. Cái tên "Arduino" đến từ một quán bar tại Ivrea, nơi một vài nhà sáng lập của dự án này thường xuyên gặp mặt. Bản thân quán bar này có được lấy tên là Arduino, Bá tước của Ivrea, và là vua của Italy từ năm 1002 đến 1014



Hình 2.1 những thành viên khởi sướng Arduino.

Lý thuyết phần cứng được đóng góp bởi một sinh viên người Colombia tên là Hernando Barragan. Sau khi nền tảng Wiring hoàn thành, các nhà nghiên cứu đã làm việc với nhau để giúp nó nhẹ hơn, rẻ hơn, và khả dụng đối với cộng đồng mã nguồn mở. Trường này cuối cùng bị đóng cửa, vì vậy các nhà nghiên cứu, một trong số đó là David Cuarlielles, đã phổ biến ý tưởng này.

Giá hiện tại của board mạch này dao động xung quanh $30 và được làm giả đến mức chỉ còn $9. Một mạch bắt chước đơn giản Arduino Mini Pro có lẽ được xuất phát từ Trung Quốc có giá rẻ hơn $4, đã trả phí bưu điện.

### **2.1.2 Giới thiệu chung về Arduino.**

Arduino thực sự đã gây sóng gió trên thị trường người dùng DIY (là những người tự chế ra sản phẩm của mình) trên toàn thế giới trong vài năm gần đây, gần giống với những gì Apple đã làm được trên thị trường thiết bị di động. Số lượng người dùng cực lớn và đa dạng với trình độ trải rộng từ bậc phổ thông lên đến đại học đã làm cho ngay cả những người tạo ra chúng phải ngạc nhiên về mức độ phổ biến.

Arduino là một board mạch vi xử lý, nhằm xây dựng các ứng dụng tương tác với nhau hoặc với môi trường được thuận lợi hơn. Phần cứng bao gồm một board mạch nguồn mở được thiết kế trên nền tảng vi xử lý AVR Atmel 8bit, hoặc ARM Atmel 32-bit. Những Model hiện tại được trang bị gồm 1 cổng giao tiếp USB, 6 chân đầu vào analog, 14 chân I/O kỹ thuật số tương thích với nhiều board mở rộng khác nhau.

Được giới thiệu vào năm 2005, Những nhà thiết kế của Arduino cố gắng mang đến một phương thức dễ dàng, không tốn kém cho những người yêu thích, sinh viên và giới chuyên nghiệp để tạo ra những thiết bị có khả năng tương tác với môi trường thông qua các cảm biến và các cơ cấu chấp hành. Những ví dụ phổ biến cho những người yêu thích mới bắt đầu bao gồm các robot đơn giản, điều khiển nhiệt độ và phát hiện chuyển động. Đi cùng với nó là một môi trường phát triển tích hợp (IDE) chạy trên các máy tính cá nhân thông thường và cho phép người dùng viết các chương trình cho Aduino bằng ngôn ngữ C hoặc C++.

Arduino là một nền tảng mà mọi thiết bị phần cứng đều được làm sẵn và chuẩn hóa, người dùng chỉ việc chọn những thứ mình cần, ráp lại là có thể chạy được. Arduino cung cấp cho bạn module điều khiển động cơ có sẵn, mạch điều khiển có sẵn, mạch thu phát sóng không dây có sẵn.

Một hệ thống Arduino có thể cung cấp cho ta rất nhiều sự tương tác với môi trường xung quanh với:

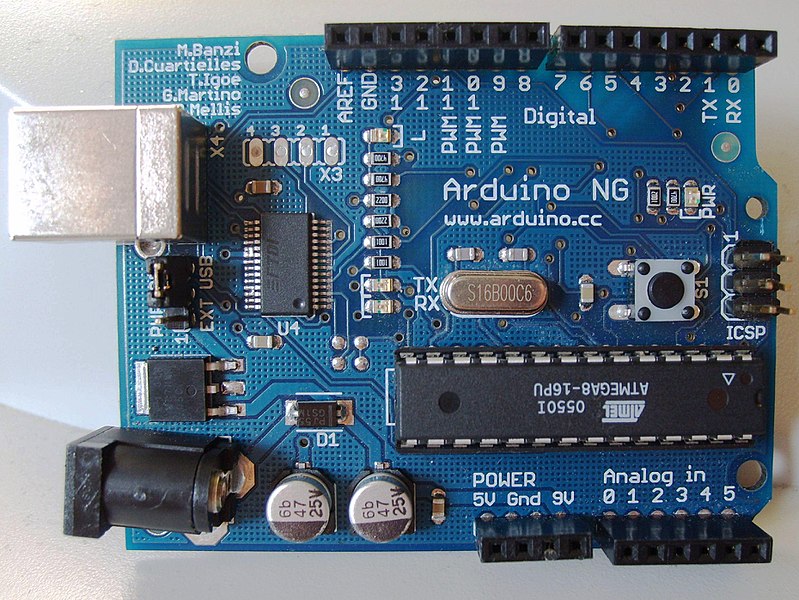
* Hệ thống cảm biến đa dạng về chủng loại (đo đạc nhiệt độ, độ ẩm, gia tốc, vận tốc, cường độ ánh sáng, màu sắc vật thể, lưu lượng nước, phát hiện chuyển động, phát hiện kim loại, khí độc,…),…
* Các thiết bị hiển thị (màn hình LCD, đèn LED,…).
* Các module chức năng (shield) hỗ trợ kêt nối có dây với các thiết bị khác hoặc các kết nối không dây thông dụng (3G, GPRS, Wifi, Bluetooth, 315/433Mhz, 2.4Ghz,…), …
* Định vị GPS, nhắn tin SMS,…

### **2.1.3 Phần cứng của Arduino**

Một mạch Arduino bao gồm một vi điều khiển AVR với nhiều linh kiện bổ sung giúp dễ dàng lập trình và có thể mở rộng với các mạch khác. Một khía cạnh quan trọng của Arduino là các kết nối tiêu chuẩn của nó, cho phép người dùng kết nối với CPU của board với các module thêm vào có thể dễ dàng chuyển đổi, được gọi là shield. Vài shield truyền thông với board Arduino trực tiếp thông qua các chân khách nhau, nhưng nhiều shield được định địa chỉ thông qua serial bus I²C-nhiều shield có thể được xếp chồng và sử dụng dưới dạng song song. Arduino chính thức thường sử dụng các dòng chip megaAVR, đặc biệt là ATmega8, ATmega168, ATmega328, ATmega1280, và ATmega2560.

Một vài các bộ vi xử lý khác cũng được sử dụng bởi các mạch Aquino tương thích. Hầu hết các mạch gồm một bộ điều chỉnh tuyến tính 5V và một thạch anh dao động 16 MHz (hoặc bộ cộng hưởng ceramic trong một vài biến thể), mặc dù một vài thiết kế như LilyPad chạy tại 8 MHz và bỏ qua bộ điều chỉnh điện áp onboard do hạn chế về kích cỡ thiết bị.

Một vi điều khiển Arduino cũng có thể được lập trình sẵn với một boot loader cho phép đơn giản là upload chương trình vào bộ nhớ flash on-chip, so với các thiết bị khác thường phải cần một bộ nạp bên ngoài. Điều này giúp cho việc sử dụng Arduino được trực tiếp hơn bằng cách cho phép sử dụng 1 máy tính gốc như là một bộ nạp chương trình.



Hình 2.2 Một mạch Arduino Uno chính thức với các mô tả về các cổng I/O.

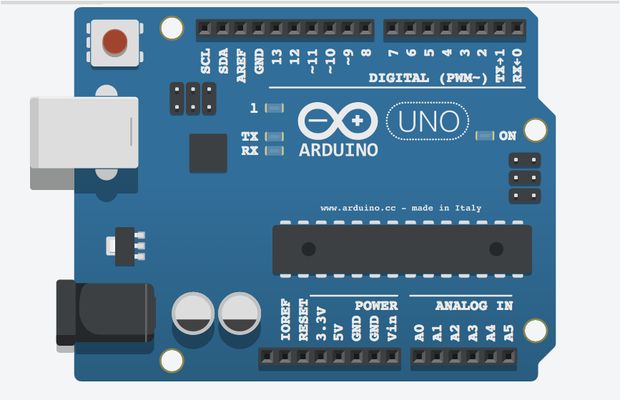
Board Arduino sẽ đưa ra hầu hết các chân I/O của vi điều khiển để sử dụng cho những mạch ngoài. Diecimila, Duemilanove, và bây giờ là Uno đưa ra 14 chân I/O kỹ thuật số, 6 trong số đó có thể tạo xung PWM (điều chế độ rộng xung) và 6 chân input analog, có thể được sử dụng như là 6 chân I/O số. Những chân này được thiết kế nằm phía trên mặt board, thông qua các header cái 0.10-inch (2.5 mm). Nhiều shield ứng dụng plug-in cũng được thương mại hóa. Các board Arduino Nano, và Arduino-compatible Bare Bones Board và Boarduino có thể cung cấp các chân header đực ở mặt trên của board dùng để cắm vào các breadboard.

Có nhiều biến thể như Arduino-compatible và Arduino-derived. Một vài trong số đó có chức năng tương đương với Arduino và có thể sử dụng để thay thế qua lại. Nhiều mở rộng cho Arduino được thực thiện bằng cách thêm vào các driver đầu ra, thường sử dụng trong các trường học để đơn giản hóa các cấu trúc của các 'con rệp' và các robot nhỏ. Những board khác thường tương đương về điện nhưng có thay đổi về hình dạng-đôi khi còn duy trì độ tương thích với các shield, đôi khi không. Vài biến thể sử dụng bộ vi xử lý hoàn toàn khác biệt, với các mức độ tương thích khác nhau.

## 2.2 Cấu tạo của Arduino Uno R3.

### **2.2.1 Giới thiệu.**

Nhắc tới lập trình hay nghiên cứu chế tạo bằng Arduino, dòng đầu tiên mà mọi người thường tìm hiểu là Arduino Uno và hiện tại đã phát triển đến thế hệ thứ 3 (R3). Nếu mà người mới tìm hiểu bạn nên nghiên cứu Arduino Uno R3 hơn là tiếp cận những dòng Arduino khác vì dòng Arduino Uno R3 rất dễ sử dụng đối với những người mới tiếp cận về lập trình.



Hình 2.3 Ảnh Arduino Uno.

Arduino Uno là 1 bo mạch thiết kế với bộ xử lý trung tâm là vi điểu khiển AVR Atmega328. Cấu tạo chính của Arduino Uno bao gồm các phần sau:

- Cổng USB: đây là loại cổng giao tiếp để ta upload code từ PC lên vi điều khiển. Đồng thời nó cũng là giao tiếp serial để truyền dữ liệu giữa vi điều khiển và máy tính.

- Jack nguồn: để chạy Arduino thỉ có thể lấy nguồn từ cổng USB ở trên, nhưng không phải lúc nào cũng có thể cắm với máy tính được . Lúc đó ta cần một nguồn từ 9V đến 12V.

- Có 14 chân vào/ra số đánh số thứ tự từ 0 đến 13, ngoài ra có một chân nối đất (GND) và một chân điện áp tham chiếu (AREF).

- Vi điều khiển AVR: đây là bộ xử lí trung tâm của toàn bo mạch. Với mỗi mẫu Arduino khác nhau thì con chip là khác nhau. Ở con Arduino Uno này thì sử dụng ATMega328.

### **2.2.2 Thông số kĩ thuật.**

xử lý: Atmega328

Điện áp hoạt động: 5V

Điện áp đầu vào: 7-12V

Điện áp đầu vào (Giới hạn): 6-20V

Chân vào/ra (I/O) số: 14 ( 6 chân có thể cho đầu ra PWM)

Chân vào tương tự: 6

Dòng điện trong mỗi chân I/O: 40mA

Dòng điện chân nguồn 3.3V: 50mA

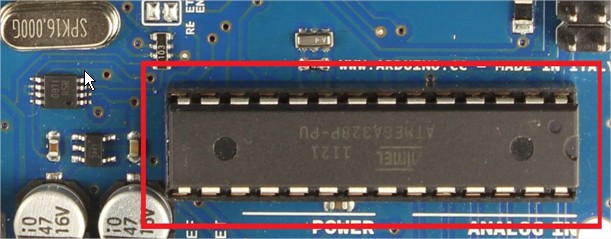
Bộ nhớ trong: 32 KB (ATmega328)

SRAM: 2 KB (ATmega328)

EEPROM: 1 KB (ATmega328)

Xung nhịp: 16MHz

### **2.2.3 Vi điều khiển của Arduino uno R3.**

[](http://k3.arduino.vn/img/2014/05/22/0/452_8121-1400736757-0--vdk.jpg)

Hình 2.4 Vi điều khiển của Arduino Uno R3.

Arduino UNO có thể sử dụng 3 vi điều khiển họ 8bit AVR là ATmega8, ATmega168, ATmega328. Bộ não này có thể xử lí những tác vụ đơn giản như điều khiển đèn LED nhấp nháy, xử lí tín hiệu cho xe điều khiển từ xa, làm một trạm đo nhiệt độ - độ ẩm và hiển thị lên màn hình LCD,…

### **2.2.4 Năng lượng.**

Arduino UNO có thể được cấp nguồn 5V thông qua cổng USB hoặc cấp nguồn ngoài với điện áp khuyên dùng là 7-12V DC và giới hạn là 6-20V. Thường thì cấp nguồn bằng pin vuông 9V là hợp lí nhất nếu bạn không có sẵn nguồn từ cổng USB. Nếu cấp nguồn vượt quá ngưỡng giới hạn trên, ta sẽ làm hỏng Arduino UNO.

Các chân năng lượng:

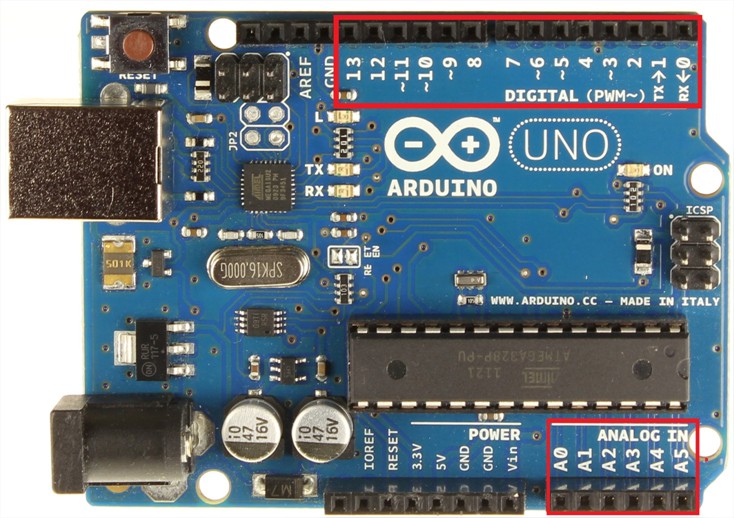
* GND (Ground): cực âm của nguồn điện cấp cho Arduino UNO. Khi bạn dùng các thiết bị sử dụng những nguồn điện riêng biệt thì những chân này phải được nối với nhau.
* 5V: cấp điện áp 5V đầu ra. Dòng tối đa cho phép ở chân này là 500mA.
* 3.3V: cấp điện áp 3.3V đầu ra. Dòng tối đa cho phép ở chân này là 50mA.
* Vin (Voltage Input): để cấp nguồn ngoài cho Arduino UNO, bạn nối cực dương của nguồn với chân này và cực âm của nguồn với chân GND.
* IOREF: điện áp hoạt động của vi điều khiển trên Arduino UNO có thể được đo ở chân này. Và dĩ nhiên nó luôn là 5V. Mặc dù vậy bạn không được lấy nguồn 5V từ chân này để sử dụng bởi chức năng của nó không phải là cấp nguồn.
* RESET: việc nhấn nút Reset trên board để reset vi điều khiển tương đương với việc chân RESET được nối với GND qua 1 điện trở 10KΩ.

### **2.2.5 Bộ nhớ.**

Vi điều khiển Atmega328 tiêu chuẩn cung cấp cho người dùng:

* 32KB bộ nhớ Flash: những đoạn lệnh bạn lập trình sẽ được lưu trữ trong bộ nhớ Flash của vi điều khiển. Thường thì sẽ có khoảng vài KB trong số này sẽ được dùng cho bootloader nhưng đừng lo, bạn hiếm khi nào cần quá 20KB bộ nhớ này đâu.
* 2KB cho SRAM (Static Random Access Memory): giá trị các biến bạn khai báo khi lập trình sẽ lưu ở đây. Bạn khai báo càng nhiều biến thì càng cần nhiều bộ nhớ RAM. Tuy vậy, thực sự thì cũng hiếm khi nào bộ nhớ RAM lại trở thành thứ mà bạn phải bận tâm. Khi mất điện, dữ liệu trên SRAM sẽ bị mất.
* 1KB cho EEPROM (Electrically Eraseble Programmable Read Only Memory): đây giống như một chiếc ổ cứng mini – nơi bạn có thể đọc và ghi dữ liệu của mình vào đây mà không phải lo bị mất khi cúp điện giống như dữ liệu trên SRAM.

### **2.2.6 Cổng vào ra** .

[](http://k3.arduino.vn/img/2014/05/25/0/467_8121-1401018414-0--input.jpg)

Hình 2.5 Các chân của Arduino Uno R3.

Arduino UNO có 14 chân digital dùng để đọc hoặc xuất tín hiệu. Chúng chỉ có 2 mức điện áp là 0V và 5V với dòng vào/ra tối đa trên mỗi chân là 40mA. Ở mỗi chân đều có các điện trở pull-up từ được cài đặt ngay trong vi điều khiển ATmega328 (mặc định thì các điện trở này không được kết nối).

Một số chân digital có các chức năng đặc biệt như sau:

* 2 chân Serial: 0 (RX) và 1 (TX): dùng để gửi (transmit – TX) và nhận (receive – RX) dữ liệu TTL Serial. Arduino Uno có thể giao tiếp với thiết bị khác thông qua 2 chân này. Kết nối bluetooth thường thấy nói nôm na chính là kết nối Serial không dây. Nếu không cần giao tiếp Serial, bạn không nên sử dụng 2 chân này nếu không cần thiết
* Chân PWM (~): 3, 5, 6, 9, 10, và 11: cho phép bạn xuất ra xung PWM với độ phân giải 8bit (giá trị từ 0 → 28-1 tương ứng với 0V → 5V) bằng hàm analogWrite(). Nói một cách đơn giản, bạn có thể điều chỉnh được điện áp ra ở chân này từ mức 0V đến 5V thay vì chỉ cố định ở mức 0V và 5V như những chân khác.
* Chân giao tiếp SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ngoài các chức năng thông thường, 4 chân này còn dùng để truyền phát dữ liệu bằng giao thức SPI với các thiết bị khác.
* LED 13: trên Arduino UNO có 1 đèn led màu cam (kí hiệu chữ L). Khi bấm nút Reset, bạn sẽ thấy đèn này nhấp nháy để báo hiệu. Nó được nối với chân số 13. Khi chân này được người dùng sử dụng, LED sẽ sáng.

Arduino UNO có 6 chân analog (A0 → A5) cung cấp độ phân giải tín hiệu 10bit (0 → 210-1) để đọc giá trị điện áp trong khoảng 0V → 5V. Với chân AREF trên board, bạn có thể để đưa vào điện áp tham chiếu khi sử dụng các chân analog. Tức là nếu bạn cấp điện áp 2.5V vào chân này thì bạn có thể dùng các chân analog để đo điện áp trong khoảng từ 0V → 2.5V với độ phân giải vẫn là 10bit.

Đặc biệt, Arduino UNO có 2 chân A4 (SDA) và A5 (SCL) hỗ trợ giao tiếp I2C/TWI với các thiết bị khác.

2.4 giao tiếp ethernet

2.4.1. ethernet là gì?

2.4.2. **cổng ethernet là gì?**

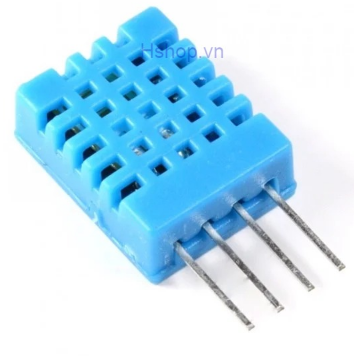
2.4.3 **các loại cáp ethernet phổ biến**

2.4.4 cách **ethernet hoạt động**

2.5 arduino ethernet shield

#hình ảnh thông số kỹ thuật

## 2.5 Cảm biến nhiệt độ, độ ẩm DHT 11.

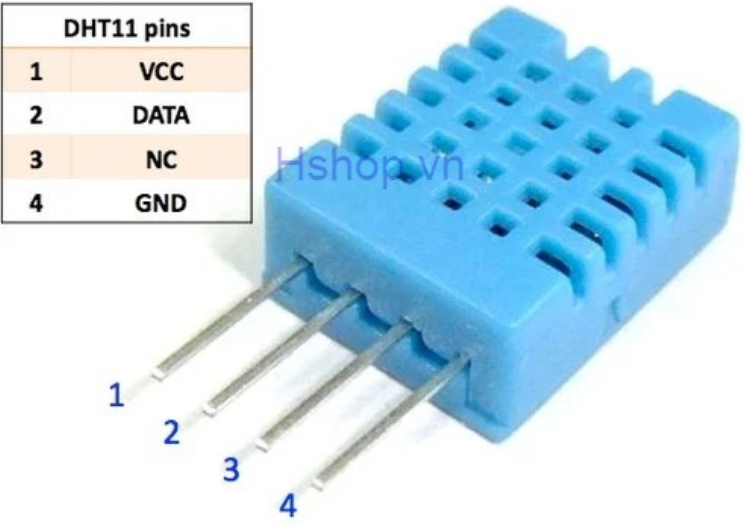


Hình 2.10 Cảm biến nhiệt độ, độ ẩm DHT 11.

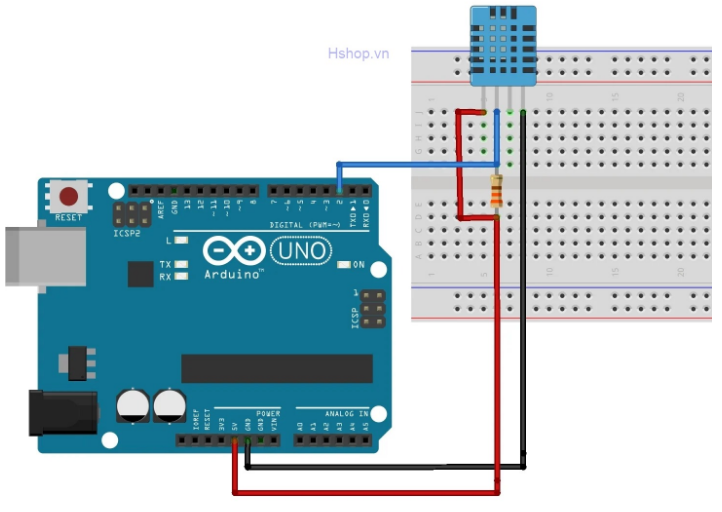
Cảm biến độ ẩm và nhiệt độ DHT11 Temperature Humidity Sensor là cảm biến rất thông dụng hiện nay vì chi phí rẻ và rất dễ lấy dữ liệu thông qua giao tiếp 1 wire (giao tiếp digital 1 dây truyền dữ liệu duy nhất). Bộ tiền xử lý tín hiệu tích hợp trong cảm biến giúp bạn có được dữ liệu chính xác mà không phải qua bất kỳ tính toán nào. So với cảm biến đời mới hơn là DHT22 thì DHT11 cho khoảng đo và độ chính xác kém hơn rất nhiều.

Thông số kỹ thuật:

* Nguồn: 3 -> 5 VDC.
* Dòng sử dụng: 2.5mA max (khi truyền dữ liệu).
* Đo tốt ở độ ẩm 2080%RH với sai số 5%.
* Đo tốt ở nhiệt độ 0 to 50°C sai số ±2°C.
* Tần số lấy mẫu tối đa 1Hz (1 giây 1 lần)
* Kích thước 15mm x 12mm x 5.5mm.
* 4 chân, khoảng cách chân 0.1''.



Hình 2.11 Các chân của DHT 11.



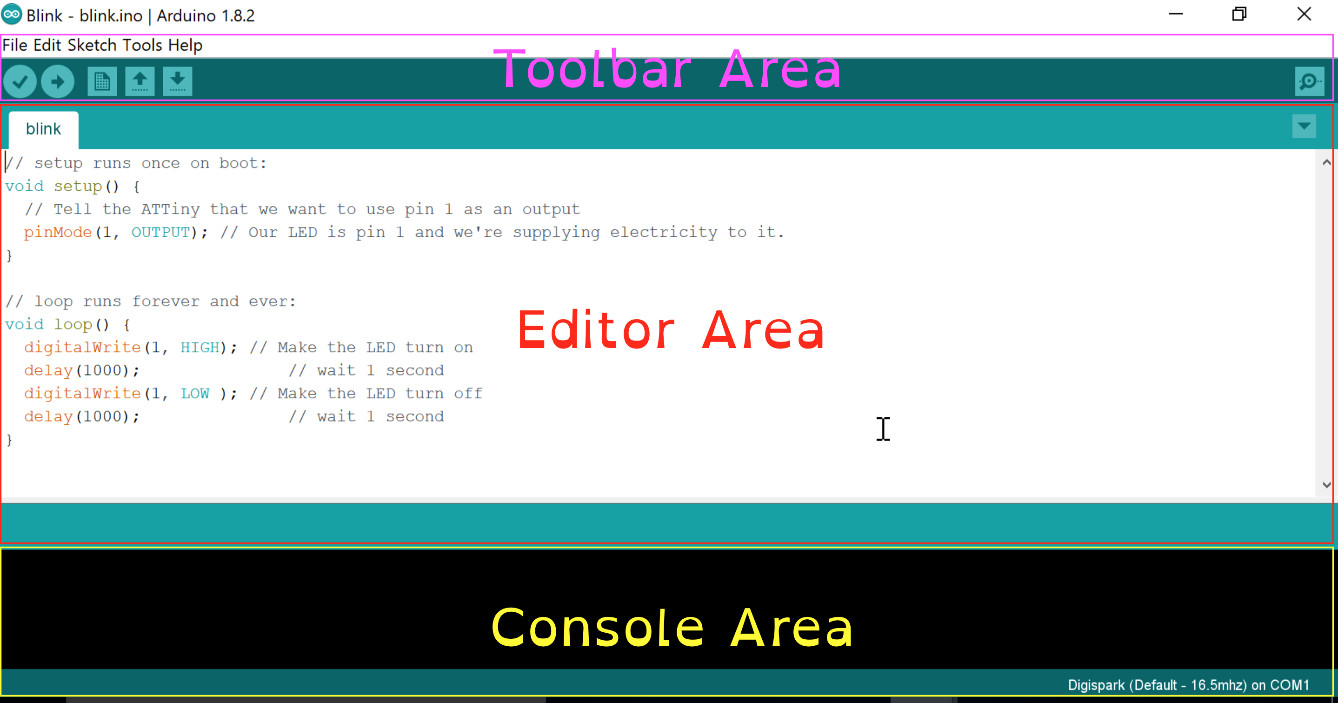
Hình 2.12 Cách nối chân DHT 11 vào board mạch Arduino UNO R3.

## 2.6 Các công cụ lập trình.

### **2.6.1 Phần mềm lập trình Arduino.**

Các thiết bị dựa trên nền tảng Arduino được lập trình bằng ngôn ngữ riêng. Ngôn ngữ này dựa trên ngôn ngữ Wiring được viết cho phần cứng nói chung. Và Wiring lại là một biến thể của C/C++. Một số người gọi nó là Wiring, một số khác thì gọi là C hay C/C++. Ta có thể gọi nó là “ngôn ngữ Arduino”. Ngôn ngữ Arduino bắt nguồn từ C/C++ phổ biến hiện nay do đó rất dễ học, dễ hiểu.

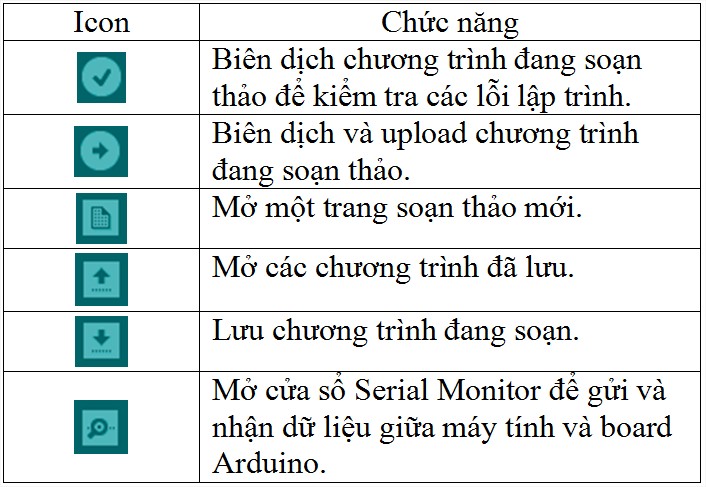
Để lập trình cho mạch Arduino, nhà phát triển cung cấp một môi trường lập trình Arduino được gọi là Arduino IDE (Intergrated Development Environment) như hình dưới đây:



Hình 2.13 Giao diện của Arduino IDE.

Như ta thấy ở trên hình giao diện của Arduino IDE có phần vùng chủ yếu là vùng lệnh, vùng viết chương trình và vùng thông báo.

* **Vùng lệnh:** Bao gồm các nút lệnh menu (File, Edit, Sketch, Tools, Help). Phía dưới là các icon cho phép sử dụng nhanh các chức năng thường dùng của IDE được miêu tả như sau:

[](http://k3.arduino.vn/img/2014/05/24/0/465_8121-1400927938-0--table.jpg)

Hình 2.14 Chức năng các công cụ.

### **Vùng viết chương trình:** Bạn sẽ viết các đoạn mã của mình tại đây.

* **Vùng thông báo (debug):** Những thông báo từ IDE sẽ được hiển thị tại đây.

[](http://k1.arduino.vn/img/2014/05/24/0/497_1231-1400928031-0--deb.jpg)

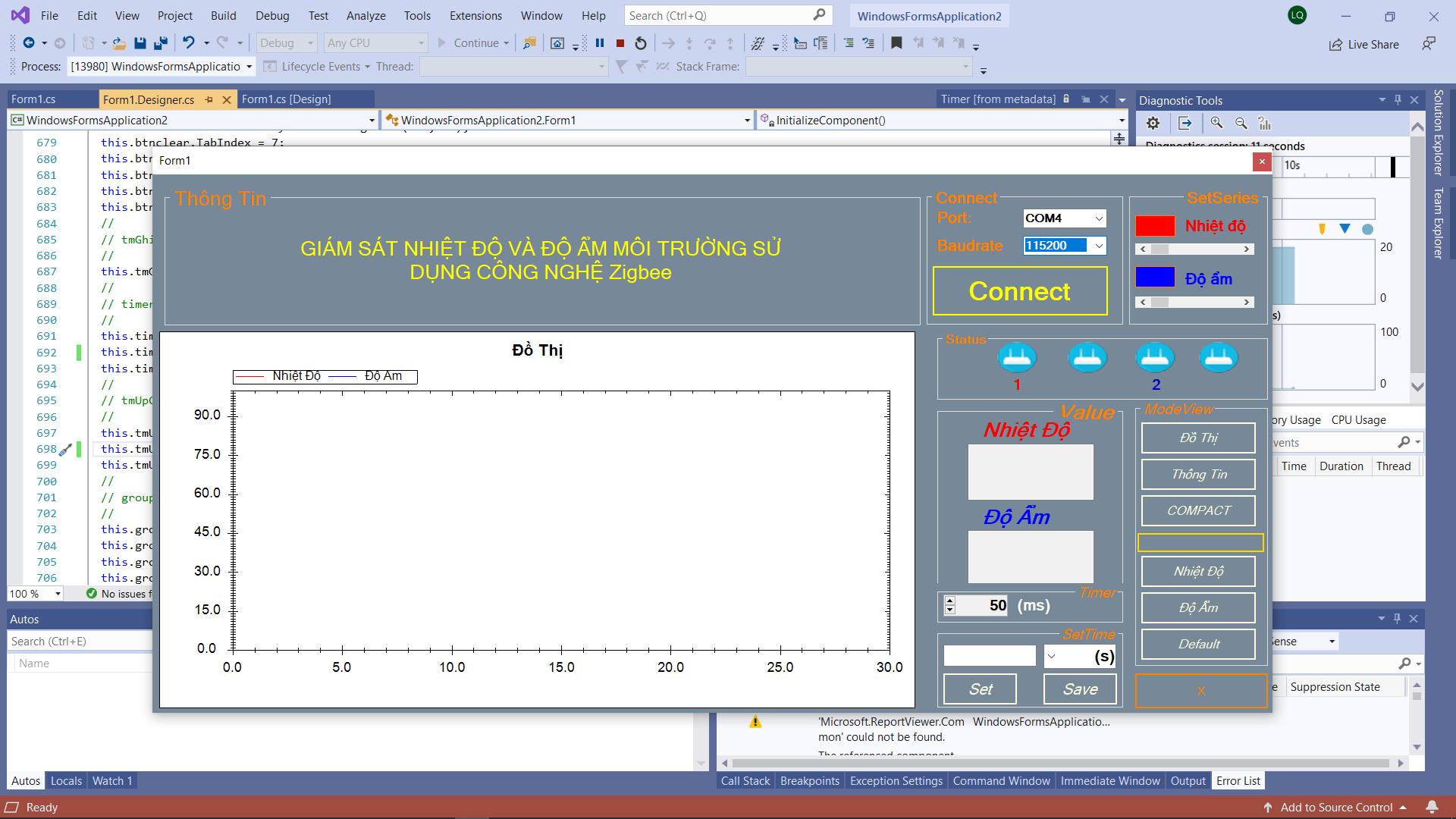
Hình 2.15 Vùng thông báo Arduino IDE.

**2.6.2 Phần mềm Visual Basic.**

Để có thể thiết kế được giao diện giúp hiển thị dữ liệu truyền qua máy tính, nhóm chúng em sử dụng phần mềm Visual Basic để làm phần giao diện giúp ta có thể xem dữ liệu, thông số truyền về.

Microsoft Visual Basic là ngôn ngữ lập trình cơ bản do Microsoft phát triển đồng thời là một môi trường phát triển tích hợp (IDE) sử dụng chuẩn COM. Visual Basic đặt nền móng phát triển của nhiều công nghệ và nền tảng khác của Microsoft, thích hợp với lập trình viên chưa có kinh nghiệm.

Về cơ bản, Microsoft Visual Basic là ngôn ngữ lập trình cũng như một môi trường phát triển tích hợp (viết tắt là IDE) sử dụng mô hình lập trình COM (Component Object Model), một chuẩn nhị phân của thành phần cấu tạo phần mềm.

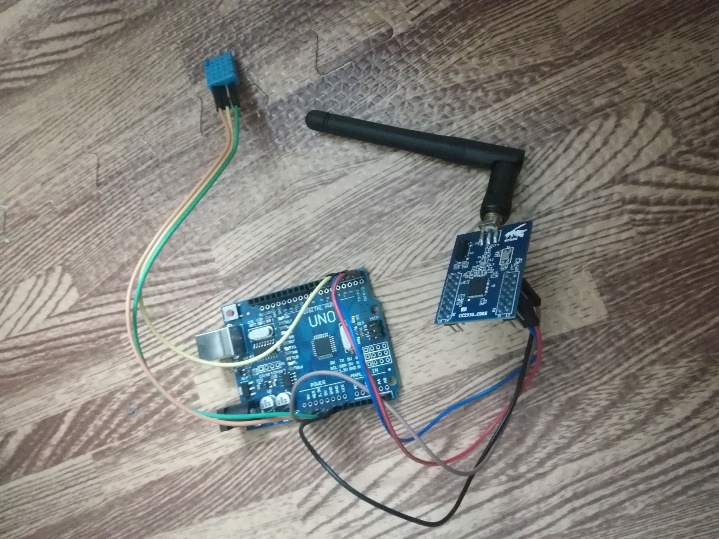


Hình 2 16 Giao diện chương trình Visual basic.

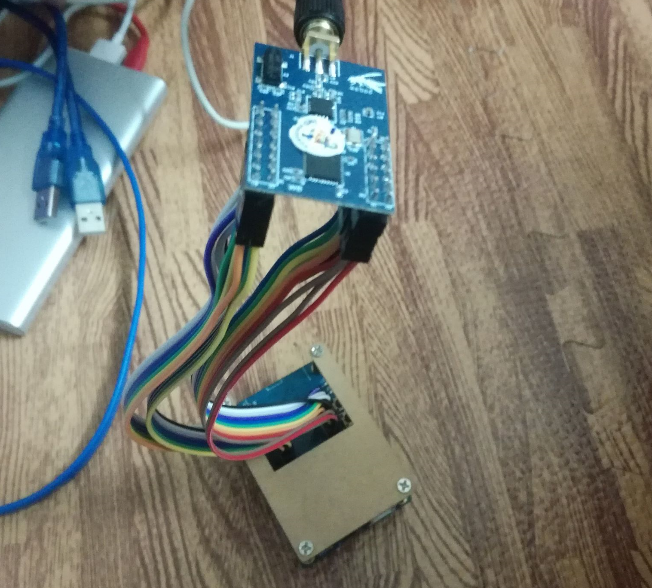
**CHƯƠNG 3: THIẾT KẾ VÀ THỬ NGHIỆM.**

**3.1 Yêu cầu thiết bị.**

Thiết kế module giám sát nhiệt độ độ ẩm môi trường sử dụng mạng Zigbee bằng cảm biến nhiệt độ độ ẩm DHT 11, board mạch Arduino và module Zigbee CC2530. Hệ thống được thiết kế gồm 2 node truyền và một node nhận.



Hình 3. 1 Một node truyền của hệ thống giám sát nhiệt độ độ ẩm sử dụng mạng Zigbee.



Hình 3. 2 Một node nhận của hệ thống giám sát nhiệt độ độ ẩm sử dụng mạng Zigbee.

Sơ đồ khối:

Hiển thị trên máy tính

Cảm biến

Module Zigbee thu

Module Zigbee phát

Module Zigbee phát

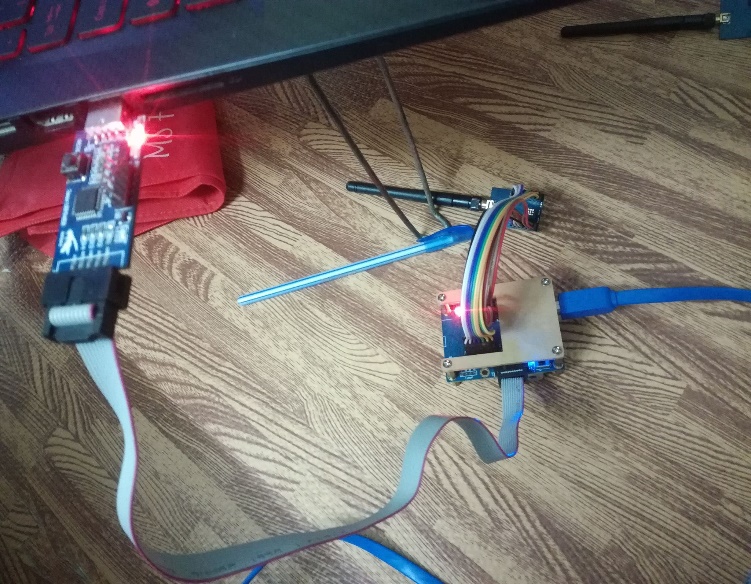
Cảm biến

Board Arduino UNO

Board Arduino UNO

Hình 3. 3 Sơ đồ khối hệ thống giám sát nhiệt độ độ ẩm môi trường sử dụng mạng Zigbee.

## C:\Users\User\Desktop\Student present topic\101457497_3636503776363717_1152937026618654720_n.jpg3.2 Thiết kế phần cứng.



Hình 3. 4 Hoàn thành 2 node truyền và node nhận kết nối và hiển thị lên máy tính.

Hiển thị lên máy tính

Arduino nhận dữ liệu truyền

Thiết lập cấu hình Coordinator

Dữ liệu truyền từ router

Thiết lập Arduino ,zigbee

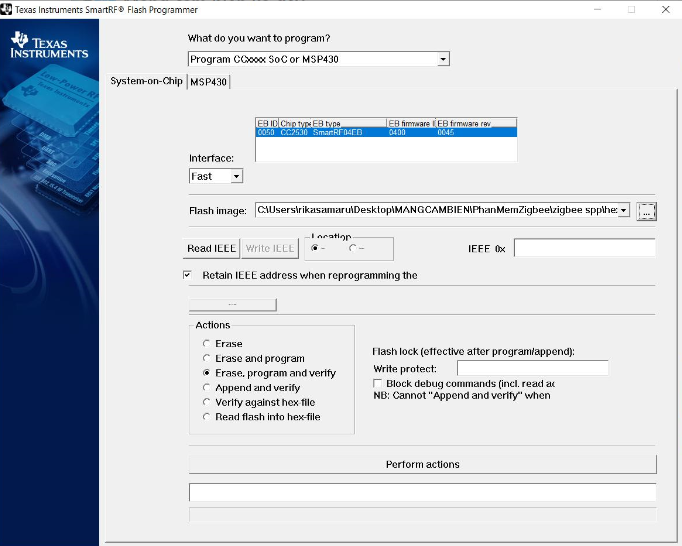
Sai

Đúng

Hình 3. 5 Lưu đồ thuật toán.

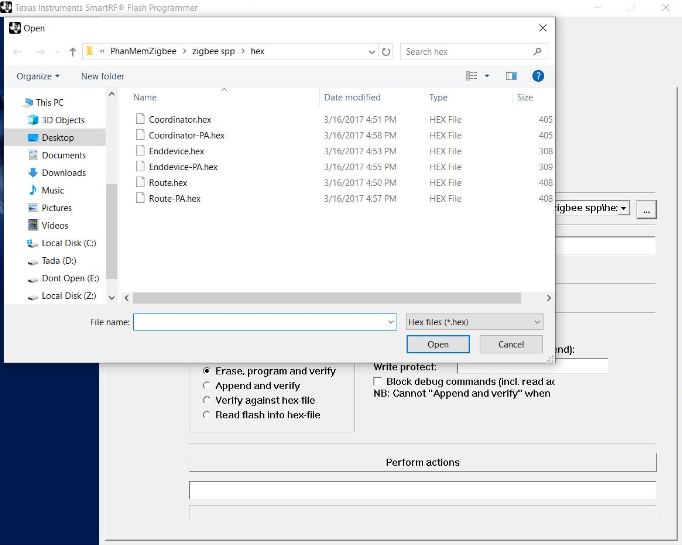
## 3.3 Thiết kế phần mềm.

Để làm hệ thống chạy đúng vai trò từng node, trước tiên ta phải thiết lập cấu hình cho từng module Zigbee ứng với chức năng cần dùng trong hệ thống. Hiện tại hệ thống mới chỉ làm có 2 node truyền và 1 node nhận, vậy nên ta sẽ kết nối từng module Zigbee với máy tình và cấu hình bằng phần mềm có sẵn cho con module Zigbee do hãng Weebee đưa kèm, dưới đây là giao diện phần mềm:



Hình 3. 6 Giao diện phần mềm thiết lập cấu hình Module Zigbee.

Ta cắm giao tiếp module Zigbee cc2530 với máy tính và nhận được thông tin đúng module Zigbee mà ta cần thiết lập, tiếp theo ta thiết lập chế độ Router hoặc Coodinator cho module Zigbee:



Hình 3. 7 Thiết lập cấu hình cho Module Zigbee.

Sau khi ta thiết lập cấu hình cho các module Zigbee xong, ta tiếp tục nạp code tiếp nhận giá trị nhiệt độ độ ẩm từ cảm biển cho các board mạch Arduino bằng phần mềm Arduino IDE:

#include <SoftwareSerial.h>

#include "DHT.h"

#define DHTPIN 2 // Digital pin connected to the DHT sensor

SoftwareSerial software\_Serial(4, 5); // RX, TX

#define DHTTYPE DHT11

//int i, count;

DHT dht(DHTPIN, DHTTYPE);

void setup() {

// Open serial communications and wait for port to open:

Serial.begin(115200);

while (!Serial) {

; // wait for serial port to connect. Needed for native USB port only

}

delay(2000);

Serial.println("Device Started.");

dht.begin();

}

void loop()

{

delay(2000);

// Reading temperature or humidity takes about 250 milliseconds!

// Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)

float h = dht.readHumidity();

// Read temperature as Celsius (the default)

float t = dht.readTemperature();

// Check if any reads failed and exit early (to try again).

if (isnan(h) || isnan(t)) {

Serial.println(F("Failed to read from DHT sensor!"));

return;

}

Serial.print("H");

Serial.print(h);

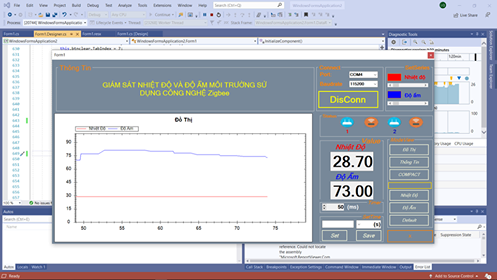
Serial.println();

delay(1000);

}

**3.3 Thực nghiệm và đánh giá.**

Hiện tại mạch đã chạy ổn định, các node truyền dữ liệu và hiển thị lên giao diện trên máy tính rõ ràng, liên tục được cập nhật. Giao diện hiển thị thêm đồ thị dễ năm bắt được nhiệt độ độ ẩm môi trường thay đổi liên tục trong thời gian thực. Tuy nhiên mạch hiện tại mới chỉ có 2 node, còn đơn giản, quy mô nhỏ. Dưới đây là hình chụp lại mạch chạy được:



Hình 3. 8 Giao diện chạy hiển thị trên máy tính.

# **KẾT LUẬN**

Sau một thời gian nghiên cứu và hoàn thành đề tài, chúng em đã nhận thấy mô hình đã hiệu quả được ở mức khá. Trong thời gian nghiên cứu và thực hiện đề tài, chúng em đã học hỏi và tìm hiểu thêm được nhiều kiến thức mới cũng như củng cố lại kiến thức đã học giúp hoàn thành đề tài này.

Đề tài cơ bản đã đáp ứng được hầu hết những yêu cầu đặt ra tuy nhiên cần được hoàn thiện, cải tiến và nghiên cứu thêm do mạch khá đơn giản chưa có gì cầu kì, phức tạp mạch mới chỉ dừng lại ở 2 node truyền, và tính ứng dụng chỉ ở quy mô nhỏ.

# **TÀI LIỆU THAM KHẢO**