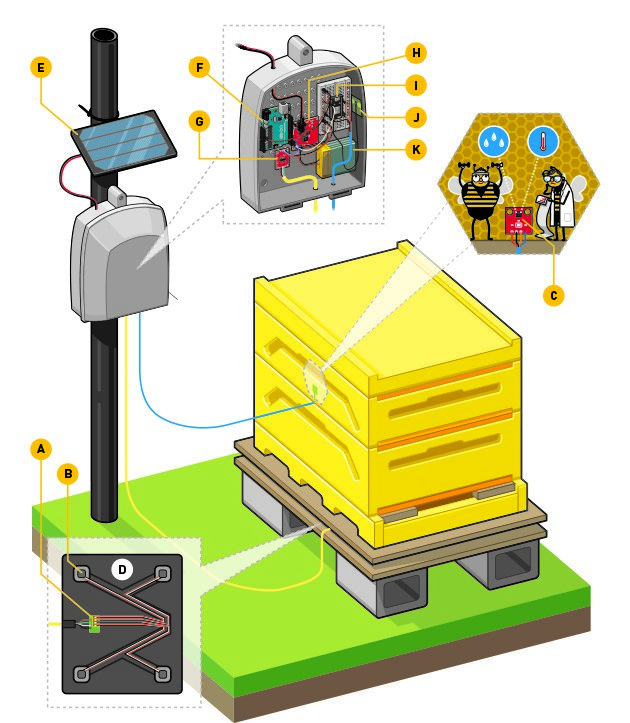
**Tài liệu hướng dẫn sử dụng BeeMonitor**



MỤC LỤC

[DANH MỤC HÌNH VẼ 5](#_Toc69115829)

[DANH MỤC BẢNG BIẾU 6](#_Toc69115830)

[GIỚI THIỆU CHUNG 7](#_Toc69115831)

[1. CHẾ TẠO THIẾT BỊ BEEMONITOR 8](#_Toc69115832)

[1.1 Phần cứng thiết bị BeeMonitor 8](#_Toc69115834)

[1.1.1 CPU 8](#_Toc69115835)

[1.1.2 OAM 8](#_Toc69115836)

[1.2 Tổng quan về modul cung cấp nguồn cho thiết bị 8](#_Toc69115837)

[1.2.1 Modul nguồn 8](#_Toc69115838)

[1.2.2 Nguồn điện thứ cấp 9](#_Toc69115839)

[1.3 Chế tạo vỏ 10](#_Toc69115840)

[1.3.1 Chi tiết khung thiết bị BeeMonitor 10](#_Toc69115841)

[1.3.2 Chi tiết thành phần thiết bị 10](#_Toc69115842)

[1.4 Chế tạo lõi 12](#_Toc69115843)

[1.4.1 ESP32 NODE-MCU-32S 12](#_Toc69115844)

[1.4.2 Module DHT21 14](#_Toc69115845)

[1.4.3 Module INMP441 : Module thu âm thanh 15](#_Toc69115846)

[1.4.4 Module HX711 và LoadCell 50kg : Cảm biến trọng lượng 16](#_Toc69115847)

[1.4.5 Mạch tự động Reset WatchDog 18](#_Toc69115848)

[1.4.6 Mạch nạp LM2596 18](#_Toc69115849)

[1.4.7 Mạch tiết kiệm Pin ( đang trong quá trình thử nghiệm) 18](#_Toc69115850)

[1.5 Thiết kế mạch in 19](#_Toc69115851)

[1.6 Kết luận chương 20](#_Toc69115852)

[2. THIẾT KẾ VÀ XÂY DỰNG THIẾT BỊ BEEMONITOR 21](#_Toc69115853)

[2.1 Xây dựng yêu cầu cần đạt của thiết bị 21](#_Toc69115855)

[2.2 Mô hình sử dụng của thiết bị thu thập thông tin môi trường 21](#_Toc69115856)

[2.3 Kiến trúc tổng thể của thiết bị thu thập thông tin môi trường 22](#_Toc69115857)

[2.4 Thiết kế và xây dựng phần mềm thiết bị thu thập thông tin môi trường 22](#_Toc69115858)

[2.4.1 Các thư viện phần mềm sử dụng cho thiết bị 22](#_Toc69115859)

[2.5 Kết luận chương 26](#_Toc69115860)

[3. TÀI LIỆU HƯỚNG DẪN LẮP ĐẶT VÀ SỬ DỤNG THIẾT BỊ 28](#_Toc69115861)

[3.1 Tài liệu hướng dẫn lắp đặt thiết bị 28](#_Toc69115863)

[3.2 Hướng dẫn sử dụng thiết bị 32](#_Toc69115864)

BẢNG CÁC TỪ VIẾT TẮT

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Từ viết tắt** | **Từ gốc** | **Tiếng Việt** |
|  | CPU | Center Processing Unit | Bộ xử lý trung tâm |
|  | 2G | 2nd - generation technology | Mạng di động thế hệ thứ 2 |
|  | 3G | Third - generation technology | Mạng di động thế hệ thứ 3 |
|  | IoT | Internet of Things | Kết nối vạn vật |
|  | GPIO | General-purpose input/output | Cổng ra vào vạn năng |
|  | ADC | Analog to Digital converter | Mạch chuyển đổi tương tự số |
|  | OAM | Operation adminnistration maintenace | Khai thác, quản lý, bảo dưỡng |
|  | CPU | Central Processing Unit | Bộ xử lý trung tâm |
|  | ROM | Read Only Memory | Bộ nhớ trong |
|  | DC | Direct Current | Dòng điện một chiều |
|  | DAC | Digital to Analog converter | Mạch chuyển đổi số tương tự |
|  | IOT | Internet of things | Internet vạn vật |
|  | I2S | Inter-IC Sound | Mạch tích hợp với thiết bị âm thanh điện tử |
|  | RAM | Random Access Memory | Bộ nhớ truy cập tạm thời |
|  | I2C | Inter Intergrated Circuit | Mạch tích hợp |
|  | IC | Intergrated Circuit | Con chip điện tử |
|  | SNR | Signal-to-Noise Ratio | Tỉ số tín hiệu trên nhiễu |

# DANH MỤC HÌNH VẼ

Hình 1 Mô hình giá đỡ BeeMonitor 10

Hình 2 Sơ đồ nguyên lý mạch BeeMonitor 12

Hình 3 Sơ đồ ESP32NODE-MCU-32S 13

Hình 4 : Sensor cảm biến nhiệt độ độ ẩm DHT21 14

Hình 5 Module BH1750FVI 15

Hình 6 LoadCell 50kg và HX711 16

Hình 7 Mạch cầu WheatStone 17

Hình 8 Mạch WatchDog 18

Hình 9 Mạch Save Battery (option) 19

Hình 10 Mạch in thiết bị BeeMonitor 19

Hình 11 Mô hình sử dụng của thiết bị BeeMonitor 21

Hình 12 Kiến trúc phần cứng của thiết bị thu thập thông tin môi trường 22

Hình 13: Sơ đồ hộp thiết bị BeeMonitor 28

Hình 14 : Sơ đồ kết nối DHT21 với ESP32 –NODE32-MCU 29

Hình 15Sơ đồ kết nối INMP441 với ESP32 –NODE32-MCU 30

Hình 16 Cách đấu cầu 4 LoadCell 50kg 30

Hình 17 : Kết nối Module HX711 với ESP32NODE-MCU-32S 30

Hình 18: Hộp in 3D đựng LoadCell 50Kg 31

Hình 19: Mạch tạo nguồn pin mặt trời 31

Hình 20 : Hình ảnh thực tế của hộp BeeMonitor 32

Hình 21: Ứng dụng EspTouch 33

DANH MỤC BẢNG BIẾU

Bảng 1 Bảng thông số ESP32NODE-MCU-32S 12

Bảng 2: Bảng thông số kỹ thuật DHT21 14

# **GIỚI THIỆU CHUNG**

Thiết bị BeeMonitor là thiết bị cho phép giám sát, kiểm tra thông số bên trong và bên ngoài của 1 tổ ong như Nhiệt độ, độ ẩm , cân nặng cùng với đo là âm thanh diễn ra trong tổ ong và sau đó gửi toàn bộ dữ liệu về Server thông báo cho người sử dụng để có thể giúp người nuôi ong có thể xử lý kịp thời nếu xảy ra sự cố .

Thiết bị BeeMonitor thường đặt ở ngoài trời nên cần những yêu cầu về mặt vật lý, hóa học.Ngoài ra, để thông tin thu thập được từ thiết bị nhanh chóng chuyển đến server để lưu trữ, xử lý một cách an toàn, tin cậy, thiết bị cần phải có kết nối tới server với độ tin cậy cao, nên khi thiết kế cần chú ý nhiều yêu cầu khác

1. **CHẾ TẠO THIẾT BỊ BEEMONITOR**

## Phần cứng thiết bị BeeMonitor

### CPU

CPU là bộ xử lý trung tâm, có trách nhiệm xử lý hầu hết dữ liệu, tác vụ của thu thập thông tin môi trường, điều khiển thiết bị đầu vào như phím giao tiếp, cũng như thiết bị đầu ra như màn hình. CPU là một tấm mạch rất nhỏ, bên trong chứa một tấm wafer silicon được bọc trong một con chip bằng gốm và gắn vào bảng mạch với tốc độ cao và hiệu suất rất lớn. Tốc độ CPU được đo bằng *hertz* (*Hz*) hay *gigahertz* (*Ghz*) giá trị này càng lớn thì CPU hoạt động càng nhanh.

Một *hertz (Hz*) là một dao động trong mỗi giây, còn một gigahertz là một tỷ dao động trong mỗi giây. Tuy nhiên, tốc độ của CPU không chỉ được đo lường bằng giá trị Hz hay *Ghz* vì CPU của mỗi hãng sẽ có những công nghệ cải thiện hiệu năng khác nhau nhằm làm tăng thông lượng dữ liệu cách mà họ mong muốn.

Module giao tiếp với hệ thống backend gồm: *2G/3G, Wifi/ IoT*.

### OAM

OAM viết tắt bởi cụm từ “Operation adminnistration maintenace” nghĩa là khai thác, quản lý và bảo dưỡng. Tập hợp các công việc nhằm bảo đảm và duy trì sự làm việc bình thường, an toàn của dự án theo quy định của thiết kế trong suốt quá trình khai thác sử dụng.

## Tổng quan về modul cung cấp nguồn cho thiết bị

### Modul nguồn

Modul nguồn là module có tích hợp các linh kiện điện tử phù hợp có nhiệm vụ biến đổi điện áp để tạo ra điện áp phù hợp với thiết bị điện và nhu cầu sử dụng. Mỗi một module nguồn chỉ có một khả năng cung cấp một dòng điện giới hạn nhất định cho tải để giữ được tính ổn định điện áp đầu ra, thông số này sẽ đặc trưng cho khả năng cung cấp dòng còn gọi là công suất module nguồn. Công suất module nguồn càng lớn thì kích thước càng to và giá thành cũng cao hơn.

### Nguồn điện thứ cấp

Nguồn điện thứ cấp hay còn gọi là pin sạc, có thể tái sử dụng nhiều lần bằng cách cắm điện và đặt vào bộ sạc để sạc lại.

Bộ nguồn là một thiết bị phần cứng cực kỳ quan trọng, nhằm cung cấp năng lượng điện cho toàn bộ hệ thống, và là một trong những thiết bị quyết định đến tuổi thọ, độ bền, sự ổn định của toàn bộ hệ thống phần cứng có trên thiết bị.

Các yêu cầu cần đạt của một bộ nguồn:

* Sự ổn định của điện áp đầu ra: điện áp không sai lệch quá *-5%* đến *+5%* so với điện áp danh định ghi trên bộ nguồn khi mà nguồn hoạt động đến công suất thiết kế,
* Điện áp đầu ra là bằng phẳng, không bi nhiễu,
* Hiệu suất làm việc cao, đạt trên *80%* (Công suất đầu ra/đầu vào đạt *> 80%),*
* Nguồn không gây ra từ trường, điện trường, nhiễu sang các bộ phận khác xung quang nó và phải chịu đựng được từ trường, điện trườn nhiễu từ các vật khác xung quang tác động đến nó,
* Trong quá trình hoạt động, bộ nguồn tỏa nhiệt ít và tiếng ồn tương đối nhỏ,
* Các dây nối đầu ra đa dạng, nhiều chuẩn chân cắm, được bọc dây gọn gàng và chống nhiễu,
* Đảm bảo hoạt động ổn định với công suất thiết kế trong một thời gian hoạt động dài.

## Chế tạo vỏ

### Chi tiết khung thiết bị BeeMonitor

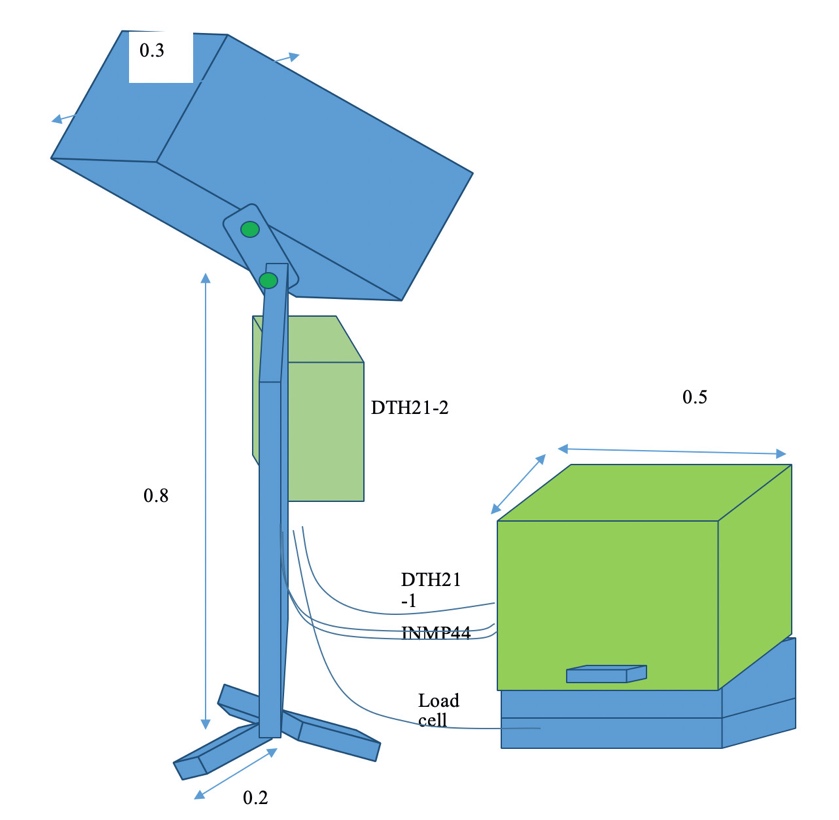
Với hình dạng tương ứng:

### Chi tiết thành phần thiết bị

Phần giá đỡ thiết bị BeeMonitor được chế tạo bằng inox (thép không gỉ).

* *Ưu điểm của Inox*
* Tuổi thọ thành phẩm lâu dài,
* Độ bóng cao,
* Không bị ăn mòn bởi hóa chất , chống oxi-hoá cao
* Ít khi hỏng, độ bền cao
* Dễ lau chùi do khó bị gỉ
* Không gây hại sức khỏe cho con người
* Khả năng chịu lực , chịu nhiệt cao, có khả năng làm việc ở môi trường khắc nhiệt

Mô hình giá đỡ BeeMonitor



Hình Mô hình giá đỡ BeeMonitor

Tấm giá đỡ này gồm có:

* Chi tiết tấm thu năng lượng mặt trời
* Chi tiết trục đỡ thiết bị
* Chi tiết chân đỡ thiết bị
* Chi tiết tấm gỗ đỡ tổ ong

1. **Chi tiết tấm thu năng lượng mặt trời**

* Tấm năng lượng mặt trời gồm 2 tấm có kích thước là 200*mm* x 300*mm*, trong đó:
* Chiều dài là 200*mm,*
* Chiều rộng là *300 mm*
* Có khả năng xoay 1800

1. **Chi tiết trục đỡ thiết bị**

* Trục đứng có chiều dài là 800*mm*, trong đó:
* Chiều dài là 800 *mm,*
* Trên trục có 2 vị trí bắt để lắp thiết bị ( vị trí đầu tiên cách đầu trục 50mm , vị trí thứ 2 cách vị trí đầu tiên 200mm)

1. **Chi tiết chân đỡ thiết bị**

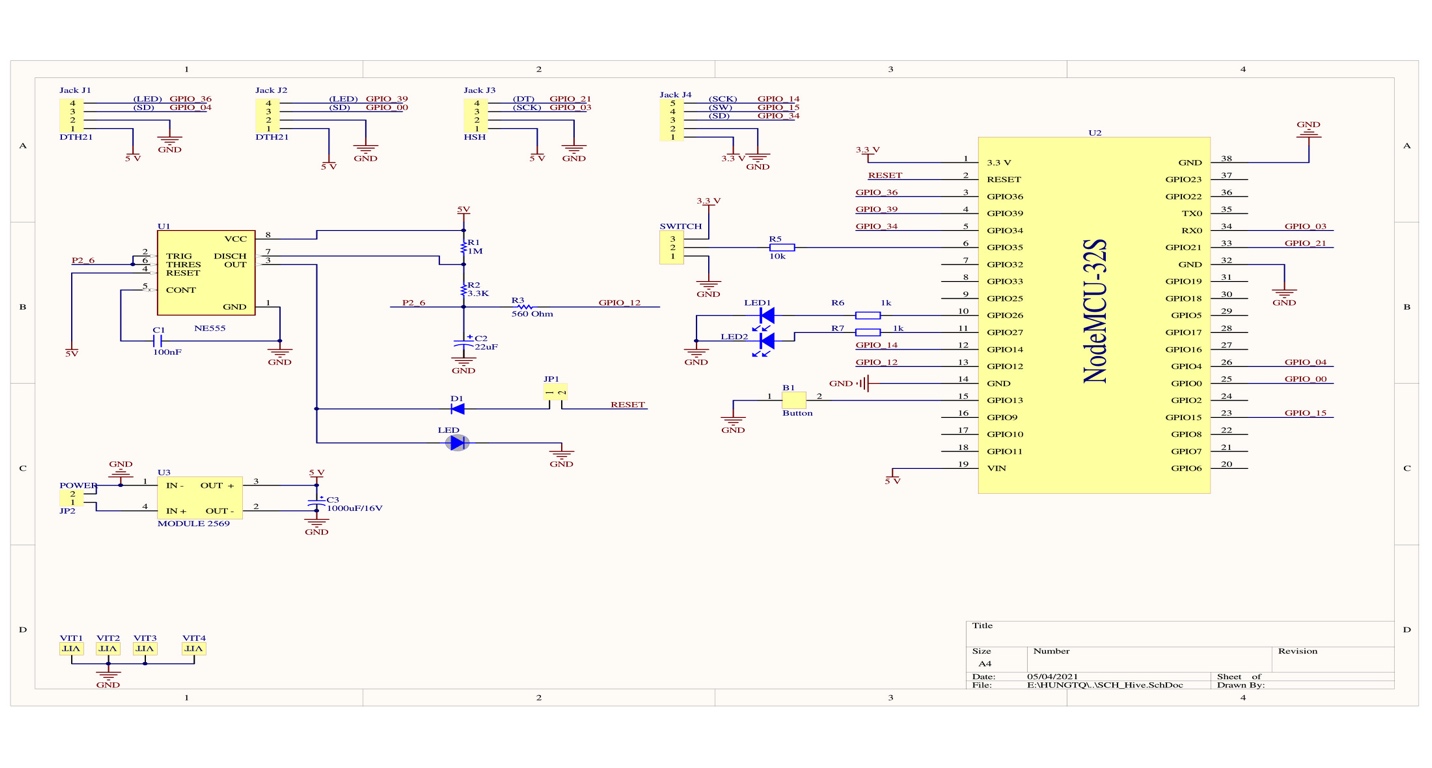
* Gồm 3 thanh có chiều dài 200mm, tạo thành kiềng 3 chân tạo thành chân đỡ cho thiết bị
* Chiều dài là 200 *mm,*
* Tiếp điểm của 3 thanh cách mặt đất 100mm

1. **Chi tiết tấm gỗ đỡ tổ ong**

* Bao gồm 2 tấm gỗ hình vuông có cạnh 500mm, trong đó
* Cạnh là 500 *mm,*
* Có khả năng chịu lực lớn ( tối đa 100kg)

## Chế tạo lõi

Sơ đồ nguyên lý mạch BeeMonitor như sau:



Hình Sơ đồ nguyên lý mạch BeeMonitor

Các công nghệ về cảm biến và thu thập dữ liệu gồm có là:

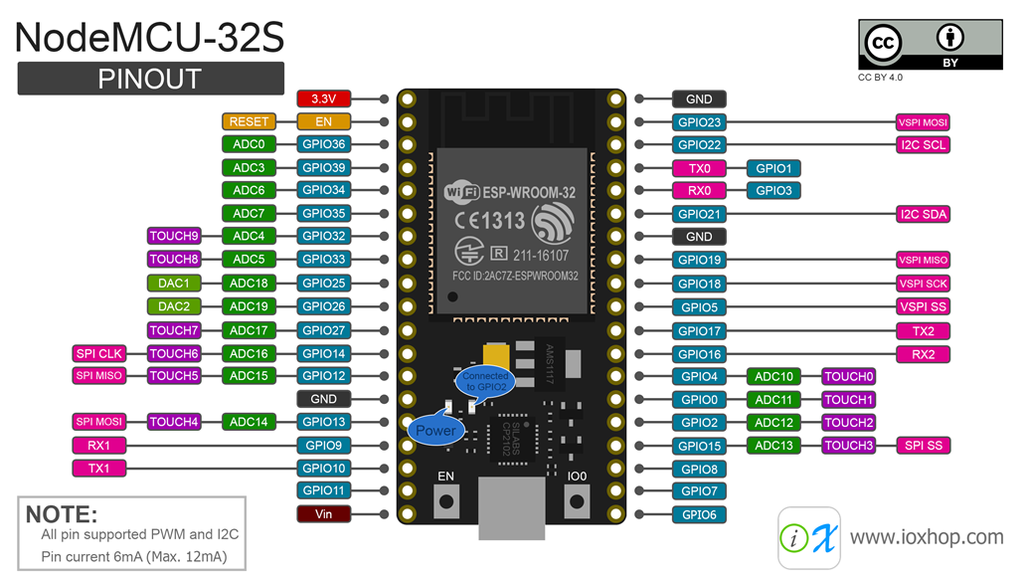
+ ESP32 NODE-MCU-32S kết nối với các module sau:

* Modul DHT21: module cảm biến nhiệt độ độ ẩm
* Module INMP441: module tiếp nhận âm thanh
* Module HX711 và LoadCell 50kg : Module cảm biến trọng lực
* Mạch nạp LM2596
* Mạch tự động reset thiết bị WatchDog
* Mạch Tiết kiệm Pin (option)

### ESP32 NODE-MCU-32S

Với những bạn thích thiết kế hệ thống điện tử, đặc biệt là hệ thống IoT thông minh thì đều biết đến ESP32.  Module ESP32 là dòng module wifi phiên bản mới nhất của hãng Espressif, các phiên bản cũ có tên là ESP . Với nhiều cải tiến về cả tính năng lẫn tốc độ xử lí, ESP32 wifi trở thành nên thông dụng trong các hệ thống IoT dù mới chỉ ra mắt vào đầu năm 2016.

Esp32 hỗ trợ thêm truyền nhận Blutooth, RAM nhiều hơn, Tốc độ xử lý nhanh hơn, số chân GPIO nhiều hơn, nhiều cổng giao tiếp hơn, nhiều chân PWM hơn, nhiều chân ADC hơn, tích hợp cả 3 loại cảm biến (nhiệt độ, hall, touch sensor)…



Hình Sơ đồ ESP32NODE-MCU-32S

Bảng Bảng thông số ESP32NODE-MCU-32S

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên** | **Giá trị** |
|  | IC chính | Wifi BLE SoC ESP32 ESP-WROOM-32 |
|  | Điện áp sử dụng | 2.2V-3.6V DC |
|  | Dòng điện sử dụng | 90mA |
|  | Nhân xử lý trung tâm | **ESP32-D0WDQ6 Dual-core low power Xtensa® 32-bit LX6 microprocessors.** |
|  | ROM | 448KBytes |
|  | SRAM | 520Kbytes |
|  | RTC SLOW | 8Kbytes SRAM |
|  | RTC FAST | 8Kbytes SRAM |
|  | WiFi | 802.11 b/g/n/d/e/i/k.r  (802.11n đến 150Mbps) |
|  | Bluetooth | v3.2 BR/EDR và chuẩn BLE |
|  | Security | WPA/WPA2/WPA2-Enterprise/WPS |
|  | Encryption | AES/RSA/ECC/SHA |
|  | Protocol | IPv4 , IPv6, SSL , TCP/UDP/HTTP/FTP/MQTT |
|  | Interfaces | SD-card, UART,SPI,SDIO,I2C,LED PWM, Motor PWM,I2S,IR,GPIO, capacitive, touch sensor, ADC , DAC , Hall sensor , temperature sensor |

### Module DHT21

Moduel cảm biến độ ẩm, nhiệt độ DHT21 AM2301 Temperature Humidity Sensor được sử dụng để đo độ ẩm và nhiệt độ của không khí, cảm biến có độ bền và độ ổn định cao, sử dụng giao tiếp 1-wire giao tiếp với Vi điều khiển dễ dàng chỉ với 1 dây tín hiệu duy nhất, thích hợp với các ứng dụng cần đo độ ẩm, nhiệt độ trong nông nghiệp, nhà thông minh,...



Hình : Sensor cảm biến nhiệt độ độ ẩm DHT21

Bảng : Bảng thông số kỹ thuật DHT21

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên** | **Giá trị** |
|  | Điện áp sử dụng | 3.3-5V DC |
|  | Dòng tiêu thụ | 300uA |
|  | Kích cỡ | *58.8x26.7x13.8(mm)* |
|  | Model | AM2301 |
|  | Độ phân giải chính xác | 0,1 |
|  | Khoảng đo | 0100% RH |
|  | Khoảng đo nhiệt độ | -40 ℃ ~ 80 ℃ |
|  | Đo lường chính xác độ ẩm | ± 3% RH |
|  | Đo lường chính xác nhiệt độ | ± 0.5 ℃ |

### Module INMP441 : Module thu âm thanh

Cảm biến âm thanh INMP441 I2S Omnidirectional Microphone được sử dụng như một Microphone đa hướng giúp thu âm thanh từ môi trường, chuyển thành tín hiệu số (Digital) và truyền tới Vi điều khiển qua giao tiếp I2S, thích hợp với các ứng dụng nhận dạng, xử lý âm thanh, điều khiển bằng giọng nói,.



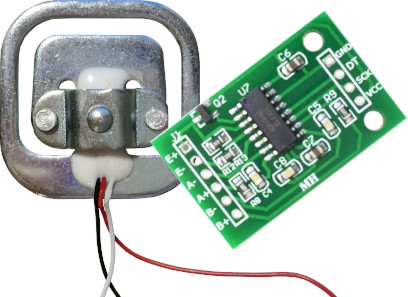
Hình Module BH1750FVI

* *Thông số kỹ thuật INMP441 gồm:*
* Điện áp sử dụng : 1.8 – 3.3V DC
* Sử dụng giao tiếp I2S
* Độ chính xác 24Bit Data
* Có chỉ số tín hiệu trên nhiễu SNR cao 61dBA
* Độ nhạy cao : -26dBFS
* Tần số âm thanh :60Hz – 15kHz
* Dòng tiêu thụ : 1.4mA

### Module HX711 và LoadCell 50kg : Cảm biến trọng lượng

Mạch chuyển đổi ADC 24bit Loadcell HX711 được sử dụng để đọc giá trị điện trở thay đổi từ cảm biến Loadcell (thường rất nhỏ không thể đọc trực tiếp bằng VĐK) với độ phân giải ADC 24bit và chuyển sang giao tiếp 2 dây (Clock và Data) để gửi dữ liệu về Vi điều khiển, thích hợp để sử dụng với Loadcell trong các ứng dụng đo cân nặng.

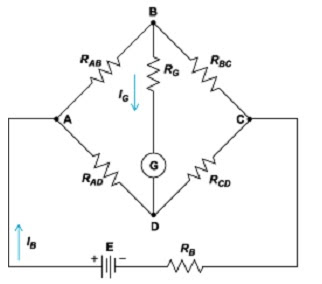
Cảm biến Loadcell 50Kg sử dụng để đo khối lượng của vật thể tối đa 50Kg, cảm biến bằng kim loại với thiết kế rất dễ lắp đặt, phù hợp với các ứng dụng cân điện tử, cảm biến khối lượng,...



Hình LoadCell 50kg và HX711

Nguyên lý hoạt động của LoadCell :

* + Load cell điện trở làm việc dựa vào nguyên lý áp lực - trở kháng. khi một tải trọng, lực, lực căng tác động lên cảm biến , trở kháng của nó thay đổi. Sự thay đổi trở kháng này sẽ đẫn đến sự thay đổi điện áp đầu ra khi một điện áp đầu vào được cấp.
  + Strain gauge là thành phần cấu tạo chính của loadcell, nó bao gồm một sợi dây kim loại mảnh đặt trên một tấm cách điện đàn hồi.
  + Một loadcell thường bao gồm các strain gauges được dán vào bề mặt của thân loadcell. Thân loadcell là một khối kim loại đàn hồi và tùy theo từng loại loadcell và mục đích sử dụng loadcell, thân loadcell được thiết kế có hình dạng đặc biệt khác nhau và chế tạo bằng vật liệu kim loại khác nhau (nhôm hợp kim, thép không gỉ, thép hợp kim).

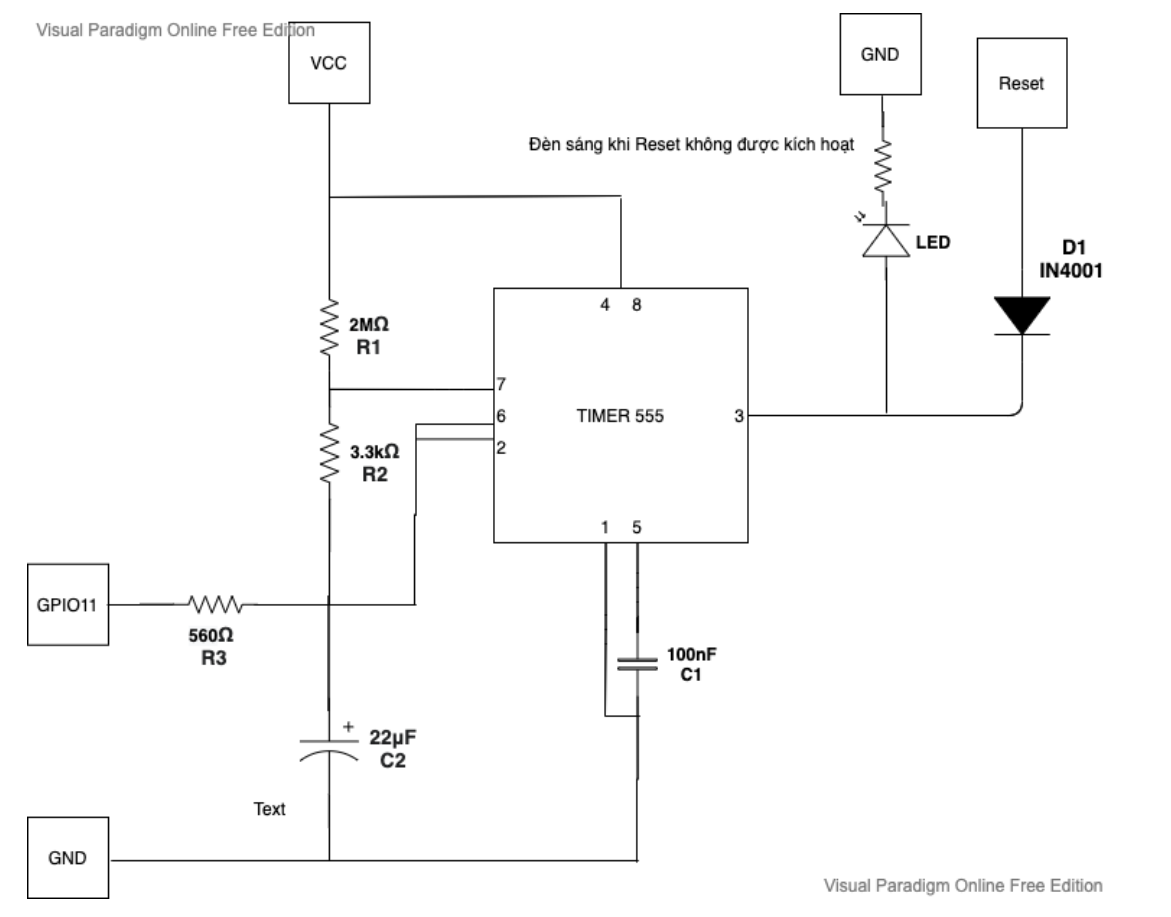


Hình Mạch cầu WheatStone

* + Bốn strain gauges được kết nối thành 1 cầu điện trở Wheatstone như hình trên và được dán vào bề mặt của thân loadcell. một điện áp kích thích - thường là 10 V được cấp vào mạch cầu ở 2 điểm A,C và chúng ta sẽ đo đc 1 điện áp đày ra ở 2 điểm B,D như hình vẽ. Tại trạng thái cân bằng (trạng thái không tải), điện áp tín hiệu ra là số không hoặc gần bằng không khi bốn điện trở được gắn phù hợp về giá trị. Khi có tải trọng hoặc lực tác động lên thân loadcell làm cho thân loadcell bị biến dạng (giãn hoặc nén), điều đó dẫn tới sự thay đổi chiều dài và tiết diện của các sợi kim loại của điện trở strain gauges dán trên thân loadcell dẫn đến một sự thay đổi giá trị của các điện trở strain gauges. Sự thay đổi này dẫn tới sự thay đổi trong điện áp đầu ra. Sự thay đổi điện áp đầu ra này thường rất bé (thường khoảng 20mV khi full tải) để có thể đo được và số hóa để tính toán thì cần phải khuếch đại tín hiệu mv này lên 0 - 5V hay 0 - 10V
* *Thông số kỹ thuật của HX711 gồm:*
* Nguồn cấp: 2.7-5V DC
* Dòng tiêu thụ : <1.5mA
* Tốc độ lấy mẫu : 10-80 SPS ( tuỳ chỉnh)
* Độ phân giải : 24bit ADC
* Độ phân giải điện áp : 40mV
* Kích thước : 38x21x10 mm

### Mạch tự động Reset WatchDog

Mạch này dùng để tự động khởi tạo lại board trong trường hợp CPU bị treo giữa chừng (30 sec khi không nhận được tín hiệu write từ chân GPIO 12)



Hình Mạch WatchDog

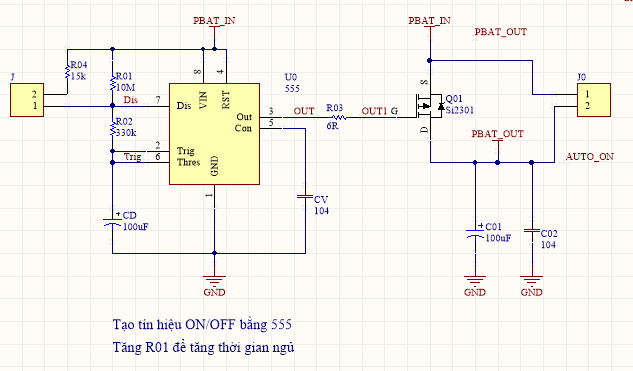
### Mạch nạp LM2596

Mạch giảm áp LM2596 là module giảm áp có khả năng điều chỉnh được dòng ra đến 3A. LM2596 là IC nguồn tích hợp đầy đủ bên trong. Tức là khi cấp nguồn 9v vào module, sau khi giảm áp ta có thể lấp được nguồn 3A < 9v...như 5V hay 3.3V.

* Thông số kĩ thuật :
* Module nguồn không sử dụng cách ly
* Nguồn đầu vào : 4-35V
* Nguồn đầu ra : 1-30V
* Dòng đầu ra max : 3A
* Kích thước mạch : 53mmx26mm
* Đầu vào : INPUT+,INPUT-
* Đầu ra : OUTPUT+, OUTPUT-

### Mạch tiết kiệm Pin ( đang trong quá trình thử nghiệm)

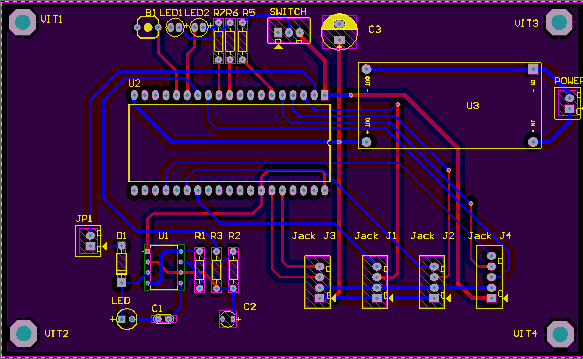
* + Khi cần board hoạt động ở chế độ tiết kiệm pin thì cần ngắt nguồn nuôi khi mạch không hoạt động
  + Mạch này dùng để tạo một cơ chế điều khiển sao cho mạch hoạt động trong 10 min và sau đó dừng hoạt động trong 30 min
  + Mạch sẽ lắp bên ngoài không được đưa vào mạch in



Hình Mạch Save Battery (option)

## Thiết kế mạch in

Mạch in được thiết kế theo khuôn mẫu dưới đây:



Hình Mạch in thiết bị BeeMonitor

## Kết luận chương

Trong chương này, nhóm thực hiện tập trung nghiên cứu, phân tích chi tiết về thiết bị thu thập thông tin môi trường gồm có phần cứng, modul cung cấp nguồn, bản thiết kế chế tạo lõi, vỏ, chi tiết giá đỡ, và thiết kế mạch in cho phù hợp với người dùng.

# **THIẾT KẾ VÀ XÂY DỰNG THIẾT BỊ BEEMONITOR**

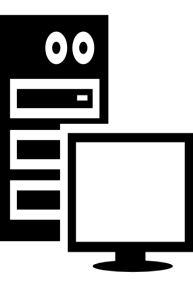
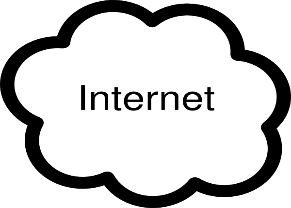
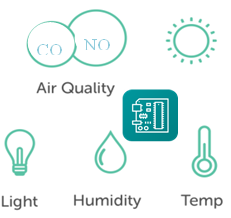
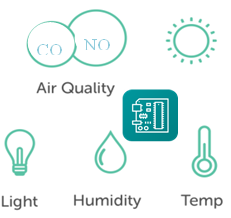
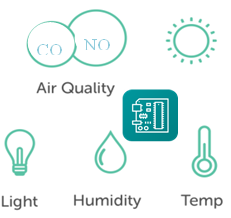


## Xây dựng yêu cầu cần đạt của thiết bị

Thu thập thông tin cho thiết bị cần đạt các yêu cầu sau:

* Đảm bảo các yêu cầu về mặt vật lý, hóa học, phù hợp với môi trường trong nhà, ngoài trời
* Đảm bảo kết nối an toàn, tin cậy để truyền thông tin tới server
* Đảm bảo độ sẵn sàng cao, không quá phụ thuộc vào nguồn điện bên ngoài
* Chất lượng, độ chính xác, độ ổn định của các sensor quan trắc chất lượng không khí tương đương với các thiết bị trên thế giới hiện nay.

## Mô hình sử dụng của thiết bị thu thập thông tin môi trường



Hình Mô hình sử dụng của thiết bị BeeMonitor

## Kiến trúc tổng thể của thiết bị thu thập thông tin môi trường

ESP32- wifi

DHT21 #1

Loadcell20kg

INMP441 mic

Nguồn dự phòng

watchdog

DHT21 #2

Hình Kiến trúc phần cứng của thiết bị thu thập thông tin môi trường

## Thiết kế và xây dựng phần mềm thiết bị thu thập thông tin môi trường

### Các thư viện phần mềm sử dụng cho thiết bị

|  |
| --- |
| 1. #include <WiFi.h> 2. #include <WebSocketsClient.h> 3. #include<WiFiMulti.h> 4. #include<ArduinoOTA.h> 5. #include <Arduino\_JSON.h> 6. #include "DHT.h" //temp and Hump 7. #include "HX711.h" //load cell 8. #include "driver/i2s.h" //sound 9. const char \*board\_code="B0001"; 10. #define AUTO\_WIFI 11. bool OTA\_MODE=false; 12. //---------------wifi---------------------------------------- 13. const char\* ssid = "CDIT"; 14. const char\* password = "Cdit@2020!"; 15. WiFiMulti WiFiMulti; 16. #define ESP\_NOW\_MAX\_DATA\_LEN 256 17. //-------------------------------------------- 18. WebSocketsClient webSocket; 19. //----------------Temp-hump defined--------------------- 20. #define DHT1PIN 4 21. #define DHT2PIN 0 22. #define DHTTYPE DHT21 23. DHT dht1(DHT1PIN, DHTTYPE); 24. DHT dht2(DHT2PIN, DHTTYPE); 25. //-----------------Load cell define---------------------- 26. const int LOADCELL\_DOUT\_PIN = 21; 27. const int LOADCELL\_SCK\_PIN = 3; 28. HX711 scale; 29. //------------------Sound define (port 14,15,34)------------ 30. const i2s\_port\_t I2S\_PORT = I2S\_NUM\_0; 31. #define SOUND\_CLOCK\_PIN 14 32. #define SOUND\_WS\_PIN 15 33. #define SOUND\_DATA\_PIN 34 34. int SOUND\_MONITOR\_ON =1; //defaut rec audio 35. //------------------------------------------------------------ 36. #define BLINK\_LED\_PIN 27 37. #define WATCHDOG\_PIN 13 |

*Hàm đọc dữ liệu âm thanh*

|  |
| --- |
| int readSound(int noise)  {  const i2s\_config\_t i2s\_config = {  .mode = (i2s\_mode\_t)(I2S\_MODE\_MASTER | I2S\_MODE\_RX),  .sample\_rate = 16000, // or 44100 if you like  .bits\_per\_sample = I2S\_BITS\_PER\_SAMPLE\_32BIT,  .channel\_format = I2S\_CHANNEL\_FMT\_ONLY\_LEFT, // Ground the L/R pin on the INMP441.  .communication\_format = i2s\_comm\_format\_t(I2S\_COMM\_FORMAT\_I2S | I2S\_COMM\_FORMAT\_I2S\_MSB),  .intr\_alloc\_flags = ESP\_INTR\_FLAG\_LEVEL1,  .dma\_buf\_count = 4,  .dma\_buf\_len = ESP\_NOW\_MAX\_DATA\_LEN \* 4,  .use\_apll = false,  .tx\_desc\_auto\_clear = false,  .fixed\_mclk = 0,  };  // The pin config as per the setup  const i2s\_pin\_config\_t pin\_config = {  .bck\_io\_num = 14, // Bit Clock.  .ws\_io\_num = 15, // Word Select aka left/right clock aka LRCL.  .data\_out\_num = I2S\_PIN\_NO\_CHANGE,//-1,  .data\_in\_num = 34, // Data-out of the mic. (someone used 23 on forums).  };  // Configuring the I2S driver and pins.  esp\_err\_t err = i2s\_driver\_install(I2S\_PORT, &i2s\_config, 0, NULL);  if (err != ESP\_OK){  Serial.printf("Failed installing driver: %d\n", err);  while (true);  }  err = i2s\_set\_pin(I2S\_PORT, &pin\_config);  if (err != ESP\_OK){  Serial.printf("Failed setting pin: %d\n", err);  while (true);  }  Serial.println("I2S driver installed.");  int32\_t rsp\_in[ESP\_NOW\_MAX\_DATA\_LEN];  i2s\_read(I2S\_NUM\_0, &rsp\_in, sizeof(rsp\_in), &bytes\_read, 1000);  cnt++;  //should add index here    webSocket.sendBIN((uint8\_t\*)rsp\_in,bytes\_read);  } |

*Hàm đọc dữ liệu nhiệt độ, độ ẩm, khối lượng*

|  |
| --- |
| //đo nhiệt độ , độ ẩm  float h = dht.readHumidity();  float t = dht.readTemperature();  //đo khối lượng |

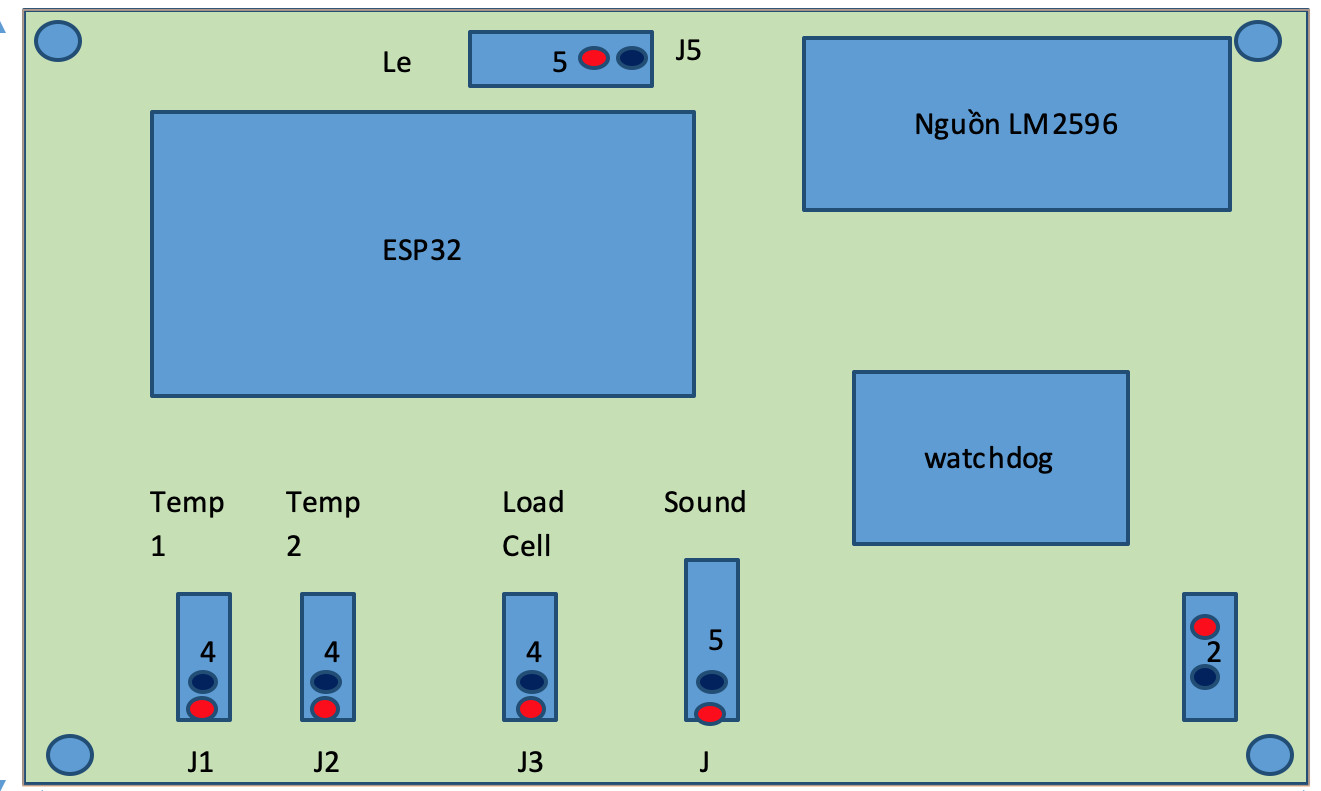
## 2.5 Kết luận chương

Trong chương này, nhóm thực hiện đã phác thảo bản thiết kế và xây dựng thiết bị theo mô hình đơn giản, dễ hiểu để người dùng sử dụng nhanh chóng và hiệu quả nhất.

# **3. TÀI LIỆU HƯỚNG DẪN LẮP ĐẶT VÀ SỬ DỤNG THIẾT BỊ**



**3.1** **Tài liệu hướng dẫn lắp đặt thiết bị**



Hình : Sơ đồ hộp thiết bị BeeMonitor

* Bước 1: Kết nối 2 thiết bị Sensor cảm biến nhiệt độ , độ ẩm DHT21 với ESP32



**DHT21**

VCC: đỏ: 5V

GND: đen

Data: Vàng, Treo 4K7 lên VCC (option)

XX: chân dự phòng khi dùng SHT21 thay DTH

**Jack J1 (4 pin)**

VCC 5V

GND

SD: GPIO04

XX: GPIO36

**Jack J2 (4 pin)**

VCC 5V

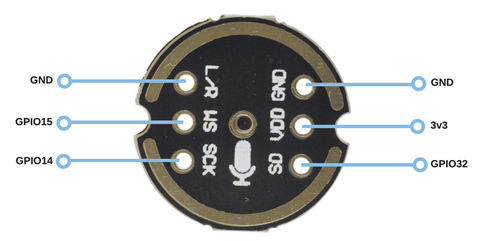
GND

SD: GPIO00

XX: GPIO39

Hình : Sơ đồ kết nối DHT21 với ESP32 –NODE32-MCU

* Bước 2 : Kết nối thiết bị thu âm audio INMP441 với ESP32



GPIO34

GPIO14

GPIO15

10K

**Jack J4 (5 pin)**

VCC 3.3V

GND

SD: GPIO34

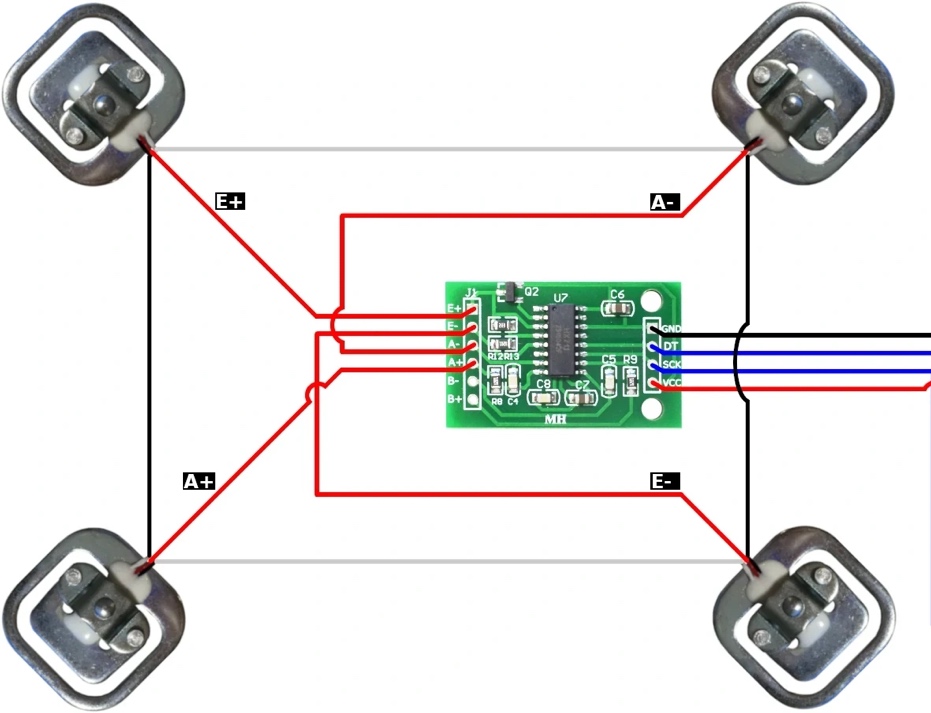
WS: GPIO15

SCK: GPIO14

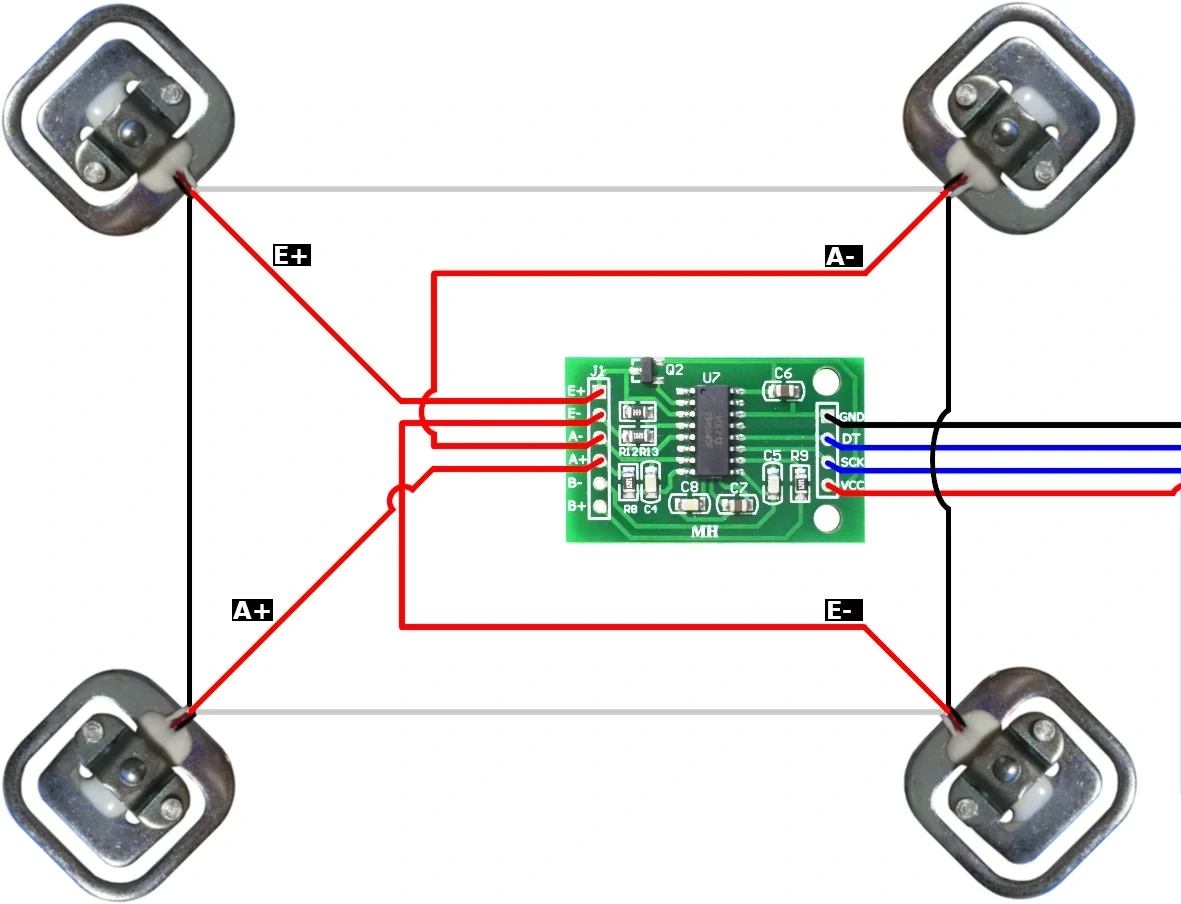
**INMP441**

Hình Sơ đồ kết nối INMP441 với ESP32 –NODE32-MCU

* Bước 3:Kết nối Module HX711 và LoadCell 50kg với ESP32-NODE32-MCU



Hình Cách đấu cầu 4 LoadCell 50kg



**Jack J3**

VCC 5V

GND

SCK: GPIO03

DT: GPIO21

Hình : Kết nối Module HX711 với ESP32NODE-MCU-32S

* Hiệu chỉnh cân :
* Giả sử có 1 chai nước nặng Km là tải trọng mẫu
* Giá trị đọc từ Load Cell khi không tải là V1
* Giá trị đọc từ Load Cell khi đặt vật mẫu lên là V2
* Công thức để tính khối lượng Vật thể bất kỳ sẽ là:

K=(Km-0)(x-V1)/(V2-V1)

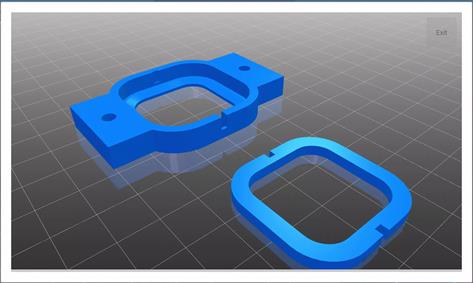
* Ví dụ Km=20g, V1=125000, V2=197570 thì

K=(250-0)(x-125000)/(197570-125000)

K=250(x-125000)/72570=0,00345x-1722

K=0.00345x-430

* Dùng công thức này để tính khói lượng (bằng gam) và gửi lên server
* Note: Các cảm bến trọng lực được đặt trong hộp để giữa với mặt gỗ ( tránh tiếp xúc giữa 2 mặt gỗ để ảnh hưởng đến sai số trong quá trình đo )



Hình : Hộp in 3D đựng LoadCell 50Kg

* Bước 4 : Kết nối mạch nạp với ESP32



Pin dự phòng

Mạch nạp LM2596

Từ pin mặt trời

Từ nguồn cấp

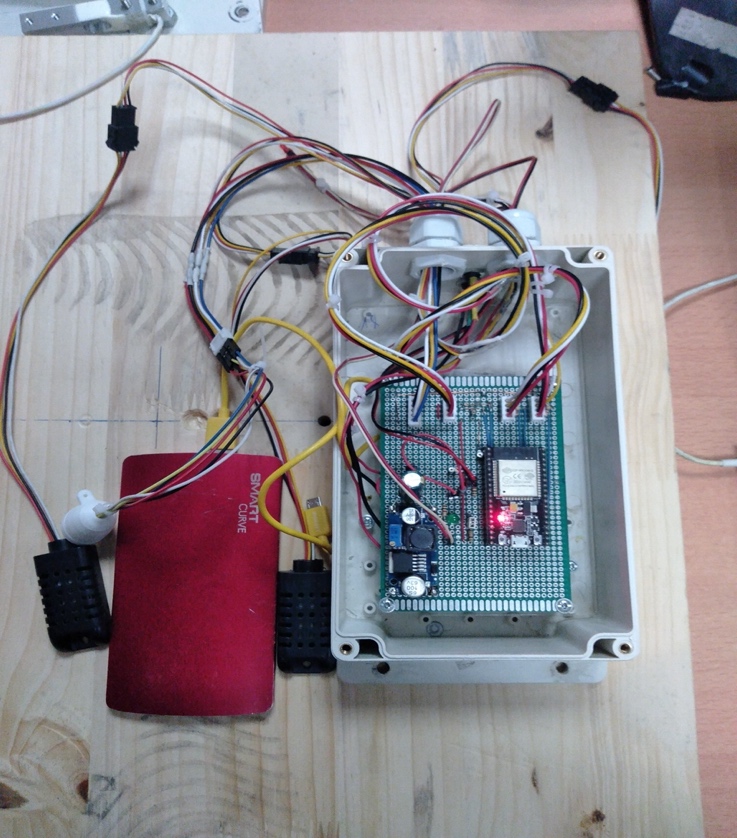
Tới ESP32

47MF,50V

Mạch nạp LM2596

Hình : Mạch tạo nguồn pin mặt trời

* Hình ảnh thực tế :



Hình : Hình ảnh thực tế của hộp BeeMonitor

## Hướng dẫn sử dụng thiết bị

* Cấp nguồn cho thiết bị

Thiết bị có thể chạy với nguồn đầu vào từ 4-35V. Chú ý giá trị nguồn đầu vào để tránh làm hư hại tới thiết bị.

* Quy trình hoạt động của thiết bị.

Sau khi nạp điện cho thiết bị sẽ thấy đèn LED bên trái sáng ( cho thấy thiết bị chưa kết nối đến WiFi). Sau khi kết nối tới WiFi , đèn LED bên trái tắt , đèn Led bên phải sẽ nháy chứng tỏ thiết bị đang hoạt động bình thường.Đến thời điểm này thiết bị đã và đang trao đổi dữ liệu thông tin với Server**.**

* Config WiFi

Địa chỉ WiFi mà thiết bị kết nối mặc định là một địa chỉ WiFi được gán cứng, cụ thể chỉ có thể thay đổi nếu nạp lại Chương trình cho thiết bị . Thế nhưng trong thiết bị chúng tôi đã cài sẵn chương SmartConfig WiFi cho phép người sử dụng có khả năng thay đổi Địa chỉ WiFi mà thiết bị có thể truy cập theo cách sau :

* Bước 1 : Sử dụng 1 thiết bị điện tử SmartPhone có khả năng chia sẻ WiFi. Sau đó cài đặt gán cứng cho thiết bị kết nối với WiFi được phát bởi SmartPhone.
* Bước 2 : Cài đặt ứng dụng SmartConfig : ESP touch



Hình : Ứng dụng EspTouch

* Bước 3 : Bấm nút trên hộp , nếu thấy đèn LED bên phải không còn nháy mà chuyển sang sáng liên tục thì thiết bị đã đưa vào trạng thái SmartConfig.
* Bước 4 : Vào ứng dụng ESP Touch: SmartConfig trên SmartPhone nhập password của WiFi mà thiết bị SmartPhone đang kết nối và ấn confirm để giúp thiết bị chuyển sang WiFi mà SmartPhone đang kết nối
* Bước 5 : Đèn Led phải sẽ quay lại trạng thái nháy chứng tỏ thiết bị chạy bình thường
* Mạch WatchDog
* Đây là một mạch hoạt động tự động trong trường hợp thiết bị bị treo đột ngột. Yêu cầu thiết bị điện tử phát WiFi cho mạch luôn phải để gần khi thiết bị BeeMonitor reset giúp thiết bị hoạt động bình thường phải có WiFi để giúp nó quay lại trạng thái bình thường.
* Chú ý nếu thấy giá trị khối lượng đưa ra bị sai lệch quá lớn do nhiễu có thể hiệu chỉnh lại cân theo công thức trong phần lắp đặt