

# TRUYỀN NHẬN DỮ LIỆU BẰNG ZIGBEE

## 1. Tổng quan về Zigbee:

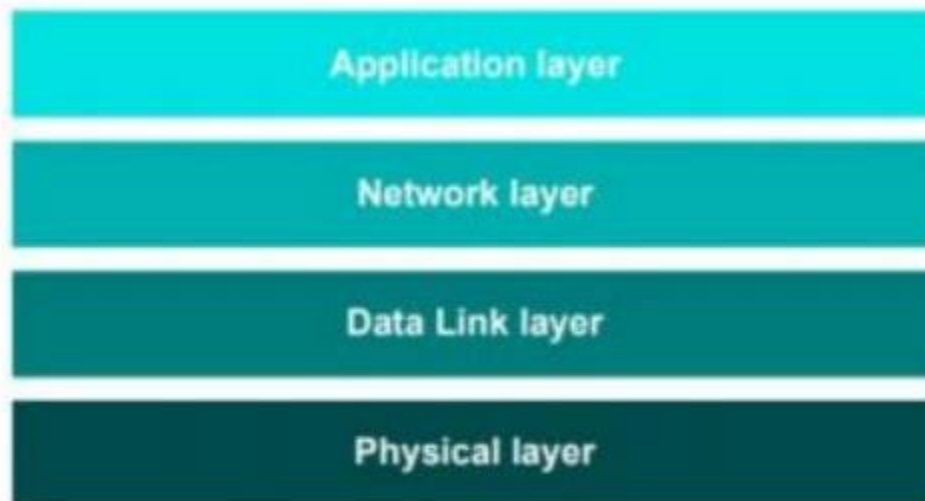
Mạng Zigbee là một công nghệ không dây đã được ra đời cách đây hơn một thập kỷ. Nó được xây dựng dựa trên **tiêu chuẩn 802.15.4 của IEEE**. Là một giải pháp thay thế cho mạng wifi và bluetooth trên một số ứng dụng. Trong đó sẽ bao gồm các thiết bị sử dụng ít năng lượng. Chẳng hạn như bóng đèn thông minh và công tắc thông minh có thể kết nối thông qua mạng Zigbee nếu như chúng được thiết kế sử dụng cùng 1 chuẩn kết nối.

Công nghệ Zigbee không có tập trung vào các điểm kết nối quá nhiều. Zigbee sẽ hoạt động tốt khi truyền dữ liệu giữa hai thiết bị có công suất cao khác nhau trong phạm vi ngắn. Trong thực tế, nếu dùng quá nhiều các thiết bị sử dụng mạng Zigbee trong hệ thống smarthome, vẫn sẽ có thể xảy ra trường hợp một số thiết bị không thể kết nối. Bởi vì có quá nhiều thiết bị muốn kết nối để tương tác với bộ điều khiển trung tâm. Do đó, bạn cần sử dụng các thiết bị có kết nối không dây để giúp tối giản hệ thống.

## 2. Kiến trúc về mạng Zigbee:

- **Tổng quan về kiến trúc mạng:**

Kiến trúc mạng Zigbee được chia thành 4 tầng chính : Application layer, Network layer, Data Link layer và Physical Layer.



Application Layer: chứa các ứng dụng chạy trong 1 node mạng. Cung cấp cho thiết bị các tính năng của nó – ví dụ như là ứng dụng chuyển đổi input thành tín hiệu số, chuyển đổi tín hiệu số sang output. 1 Node có thể chạy nhiều ứng dụng – VD: cảm biến đo môi trường có thể bao gồm các sensor: temp, hump,ats,..

Network Layer: Cung cấp các chức năng của Zigbee PRO và giao diện ứng dụng đến IEEE 802.15.4. Tầng này liên quan đến cấu trúc mạng và multi-hop routing

Data Link Layer: Được cung cấp bởi chuẩn IEEE 802.15.4 và chịu trách nhiệm trong việc quản lý địa chỉ - ví dụ cho 1 data được truyền ra ngoài, nó sẽ quyết định xem địa chỉ của data sẽ được đi tới, hoặc như 1 data được nhận về, nó sẽ phải biết được data được truyền từ đâu. Nó cũng chịu trách nhiệm lắp ráp các gói dữ liệu (data packets) hoặc khung (frames) để truyền hoặc phân tách các khung nhận được.

Physical Layer: Được cung cấp bởi chuẩn IEEE 802.15.4, liên quan tới giao diện truyền dẫn vật lý (sử dụng radio), trao đổi bit dữ liệu với lớp Data link.

### **3. Cấu trúc mạng Zigbee:**

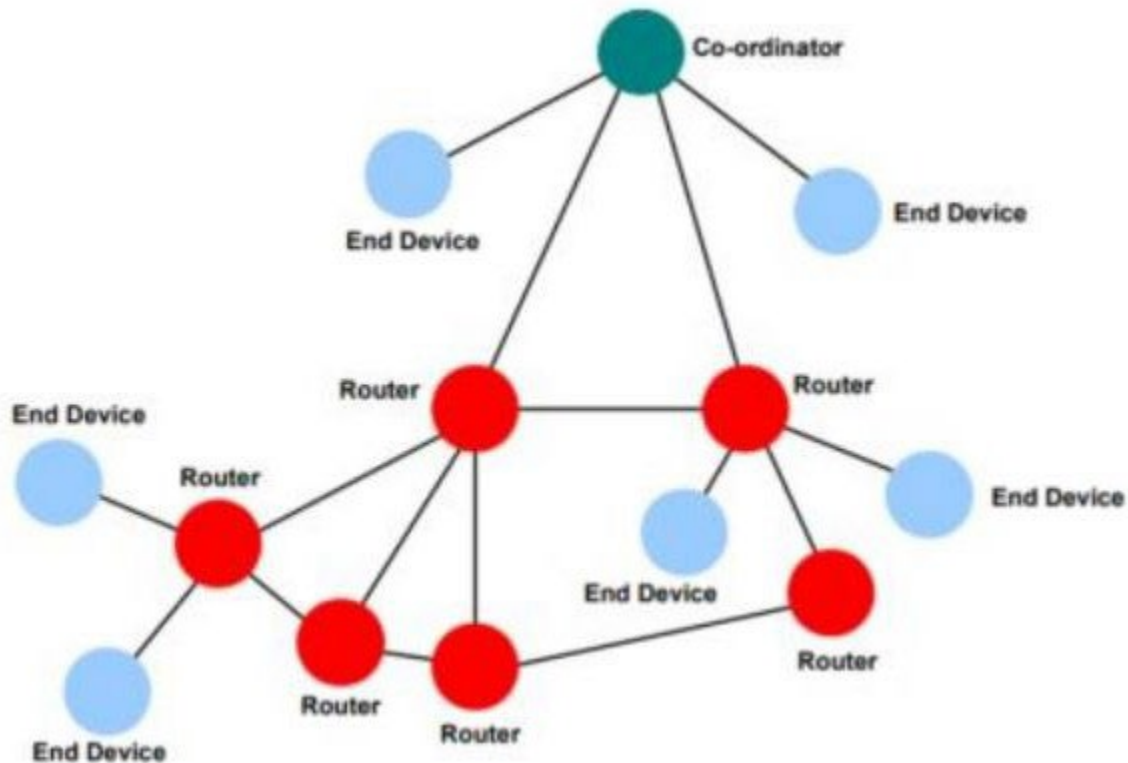
Bao gồm 3 node chính: Co-ordinator, Router, End-device. Zigbee được thiết kế để thực hiện giao thức Mesh-network.

Mesh network bao gồm Co-ordinator, Routers và End-devices. Coordinator sẽ liên kết với các Routers và End-devices – coi như là children của nó. Router có thể liên kết với các Router khác và End devices-children. Mọi quan hệ giữa các node phải được tuân thủ theo các luật:

- Coordinator và Router có thể có children, và do đó nó có thể là parent.
- Router có thể làm được cả parent lẫn child.
- End Devices không có children.

Luật giao tiếp trong mesh-network:

- End Device chỉ có thể giao tiếp trực tiếp với parent của nó – Router có thể giao tiếp trực tiếp với children, parent của nó và với các Router,Coordinator trong mạng.
- Coordinator có thể giao tiếp trực tiếp với children của nó và với các Router khác trong mạng.



#### 4. Neighbor Table:

Neighbour table trong các Routing node (Router hoặc Co-ordinator) sẽ giữ thông tin về các node giao tiếp trực tiếp với nó:

- Phần đầu tiên trong bản sẽ chứa các thông tin về các node làm parent của nó
- Phần tiếp theo trong bảng giữ thông tin về các thiết bị con của nó.
- Phần cuối cùng của bảng giữ thông tin về các node có thể giao tiếp trực tiếp với nó (không phải parent cũng không phải child) Neighbour table size mặc định được đề là 26 – đây là độ rộng size tối thiểu được yêu cầu bởi Zigbee-Compliant Platform. Size có thể dễ dàng thay đổi thông qua Active Neighbour Table Size, nhưng khi tăng table-size thì sẽ sử dụng nhiều RAM hơn. Hơn nữa, nếu tăng Neighbour size > 26 thì packet sẽ bị dài ra (vì 1 packet có thể chứa tối đa là 26 neighbour), do đó, bản tin sẽ được truyền 2 lần cho những packet này, 2 phần đầu của bảng neighbor, dành cho các thiết bị parent và child, chúng được lưu ở 1 bảng phụ trong EEPROM, bảng phụ này không được lớn hơn 2/3 bảng neighbor. Bảng phụ này chứa các mục con, kích thước của bảng phụ xác định số lượng children mà thiết bị được phép có. Số lượng children là nhỏ hơn 1 so với độ rộng của bảng phụ

## 5. Network Identify

Hệ thống mạng zigbee yêu cầu có 1 tên nhận dạng độc lập. Điều này cho phép có thể tồn tại nhiều mạng Zigbee cùng hoạt động trong 1 dải tần.

Zigbee sử dụng 2 loại nhận diện:

- PAN ID: sử dụng giá trị 16-bit, sử dụng trong giao tiếp trong mạng để các node biết được chúng đang ở trong cùng 1 mạng. Giá trị PAN ID được chọn random bởi Co-ordinator khi khởi tạo network. Khi các nodes các join mạng, nó sẽ học giá trị PAN-ID và sử dụng nó để giao tiếp trong mạng.
- Extended PAN ID: sử dụng giá trị 64-bit, sử dụng để tạo mạng và sau đó có thể sửa đổi mạng – nếu cần. Giá trị này có thể được cài đặt trước bởi người dùng chạy trong Co-ordinator. Hoặc thay vào đó, giá trị này có thể đặt trước là 0, khi đó Co-ordinator sẽ lựa chọn 1 địa chỉ Ex-PANID nhất định cho nó khi khởi tạo mạng điều này thì đảm bảo Ex- PANID không bị trùng so với những mạng khác.

Khi Router hoặc End-device lần đầu tìm mạng để xin vào, nó sẽ sử dụng Ex-PANID trong 2 cách:

- Nếu Ex-PANID được đặt sẵn bởi người dùng cho Router hoặc End-device, Node sẽ tự vào mạng với giá trị Ex-PANID có sẵn
- Nếu Ex-PANID chưa được đặt sẵn cho Router hoặc End-device, sau khi vào mạng, nó sẽ học Ex-PANID của mạng và sẽ sử dụng nó trong việc rejoin network nếu trong 1 vài trường hợp nó không thể liên kết được với mạng đó (rơi vào trạng thái orphaned).

## 5. Network Creation Khởi tạo mạng Zigbee

Quá trình tạo mạng của (Co-Ordinator): Co-ordinator là node có khả năng bắt đầu tạo 1 mạng mới. Nó phải là node đầu tiên trong mạng được hình thành, để tạo mạng, Coordinator hoạt động theo các bước:

- Set Ex-PANID và Coordinator address: Cài đặt địa chỉ Ex-PanID cho mạng dựa theo tầng app của Coordinator (nếu giá trị trong Coordinator được mặc định đặt là 0, nó sẽ tự tìm và cài đặt cho mình 1 địa chỉ 64-bit riêng). Sau đó, nó sẽ tự động cấu hình địa chỉ mạng cho nó là 0x0000
- Select radio channel: Coordinator thực hiện việc quét năng lượng (Energy Detected Scan) để tìm ra 1 kênh phù hợp.

Set PAN ID: Sau khi radio channel được chọn, Co-ordinator sẽ lựa chọn 1 địa chỉ 16- bit cho PAN ID network. Để làm được điều này, nó sẽ phải nghe các luồng mạng khác đang

tồn tại và xác định các địa chỉ PAN ID của các luồng mạng này. Và để tránh xung đột xảy ra, Co-ordinator sẽ chọn 1 định danh PAN-ID không sử dụng những định danh đang tồn tại

Receive join requests from other devices: Bây giờ, Coordinator có thể nhận yêu cầu xin gia nhập mạng từ Router và End devices.

## **6. Tham gia mạng (Routers And End-Devices)**

Search for network: node mới đầu tiên sẽ quét các kênh trong dải tần liên quan để tìm mạng. Có thể có nhiều mạng Zigbee đang hoạt động, đôi khi có thể nhiều mạng trong cùng 1 kênh. Nhưng việc lựa chọn mạng có thể được định trước do người dùng bằng việc định nghĩa trước địa chỉ Extended PAN ID cho node đó. Nếu không nó sẽ lựa chọn những mạng có năng lượng lớn và đang cho phép gia nhập mạng.

Select Parent: Node sẽ lựa chọn parent cho chính nó bằng việc nghe hoạt động trong mạng. Parent được node lựa chọn sẽ là những parent có giá trị depth nhỏ nhất trong mạng, nghĩa là parent đó đang có đường route gần Co-ordinator nhất.

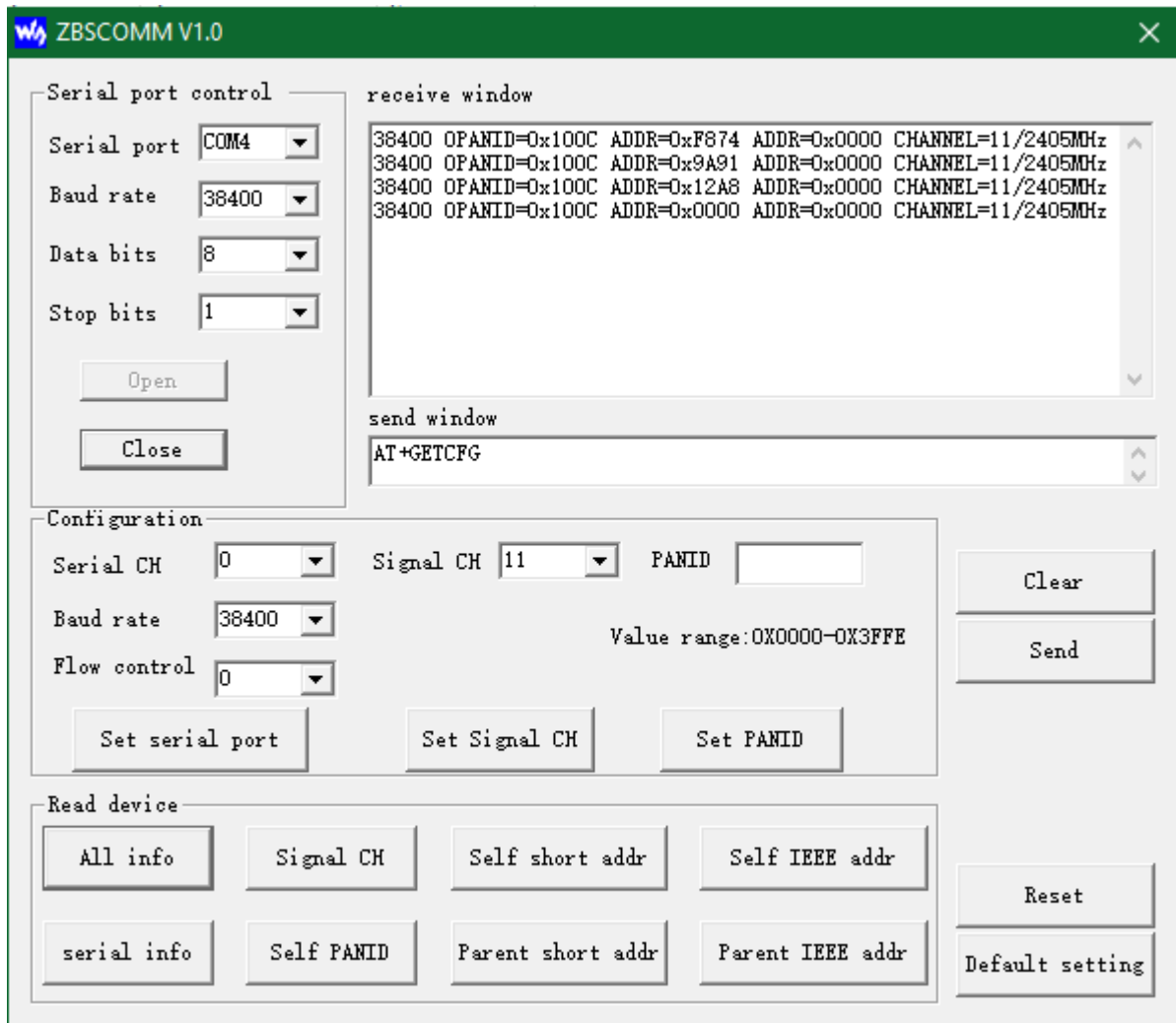
Request joining: Node sau đó sẽ gửi message cho parent của nó, yêu cầu xin gia nhập mạng

Receive Response: Sau khi gửi yêu cầu gia nhập, node sẽ đợi phản hồi từ parent của nó, để xác định node có phải là thiết bị được phép vào mạng hay không và parent có đang permit join hay không. Thì parent sẽ hỏi Trust-Center (trong trường hợp parent không phải là TC)

- Nếu các tiêu chí trên được thỏa mãn thì parent sẽ cho phép node tham gia mạng và thêm vào bảng child của nó.
- Trong trường hợp parent không chấp nhận node là child, 1 tin nhắn hủy bỏ sẽ được gửi đến node, lúc đó node sẽ phải đi tìm 1 parent khác (hoặc 1 mạng khác)

## 7. Thực hành cấu hình mạng Zigbee

Mục tiêu: Cấu hình một mạng Zigbee bao gồm hai thành phần là 1 thiết bị Coordinator và 3 thiết bị Routers, trong đó Coordinator quản lý các thiết bị trong mạng Zigbee, và thực hiện truyền nhận dữ liệu giữa các thiết bị trong mạng Zigbee.



Cấu hình thành công có thể thấy được có 3 Router với addr lần lượt là 0xF874, 0x9A91 và 0x12A8, 1 Coordinator có addr là 0x0000

- Module sử dụng: Zigbee CC2530 mẫu giống hình dưới



Mô-đun không dây CC2530 ZigBee là giải pháp hệ thống trên chip (SOC) thực sự cho các ứng dụng IEEE 802.15.4, mô-đun có thể được sử dụng rộng rãi trong lĩnh vực truyền thông không dây khoảng cách ngắn, với mức tiêu thụ điện năng thấp, âm lượng nhỏ, khả năng chống nhiễu mạnh mẽ.

Mô-đun ZigBee này dựa trên chip TI CC2530F256, nó có tất cả các tính năng của giao thức ZigBee. Khác với các loại mô-đun ZigBee khác, CC2530 khởi chạy mô-đun ZigBee tự tạo, sử dụng năng lượng, người dùng không cần hiểu sự phức tạp của giao thức ZigBee, tất cả phần xử lý giao thức ZigBee, trong mô-đun ZigBee tự động, người dùng chỉ cần truyền dữ liệu thông qua cổng nối tiếp, hiện đang có trên thị trường cách đơn giản nhất để sử dụng các ứng dụng ZigBee.

Thông số kỹ thuật:

- Tần số: 24MHz
- Flash: 256KB
- SRAM: 8kB
- GPIO: 21
- SPI: 1
- UART: 1
- ADC: 12-bit, 8 channel
- Min VCC: 2 VDC
- Max VCC: 3.6 VDC
- Chế độ chờ: 0.4uA
- Thời gian đánh thức chế độ chờ: 500us
- Nhiệt độ hoạt động: -40~125°C
- RF 6LoWPAN/802.15.4 MAC
- CPU Core: 8051
- Bảo mật: 128-bit AES
- Anten kết nối: có
- Dòng RX: 24mA
- Tần số băng thông: 2394~2507MHz

### 7.1. Bước 1: Cài đặt Firmware cho Zigbee

**Firmware** là thuật ngữ nói về loại chương trình máy tính với khả năng kiểm soát các phần cứng của thiết bị điện tử ở mức cơ bản. Hoặc, hiểu đơn giản firmware là phần mềm hỗ trợ kiểm soát phần cứng.

Đầu tiên để cài Firmware cho Zigbee các bạn nên cài các file theo đường link này sau: [https://github.com/dungph1029/embedded\\_ArduinoUnoR3/tree/main/CCLoader-main](https://github.com/dungph1029/embedded_ArduinoUnoR3/tree/main/CCLoader-main)

Ở trong link có một file ino là Ccloader.ino, truy cập vào file này. Trong code Ccloader.ino thì bạn chỉ cần chú ý về các chân ở chỗ dòng này



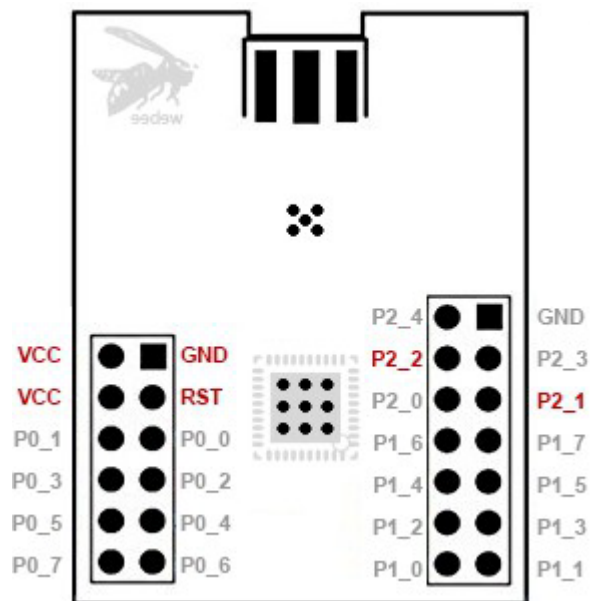
```

85
86 // Debug control pins & the indicate LED
87 int DD = D5;
88 int DC = D6;
89 int RESET = D7;
90 int LED = D4;
91

```

Khuyến nghị nên dùng board của esp32/8266 hoặc là wemos thay vì Arduino Uno để đỡ lỗi mất Firmware.

Sơ đồ chân của Zigbee CC2530:

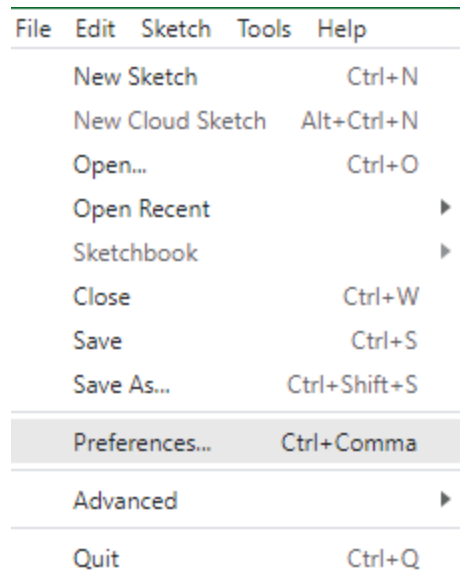


Các bạn nối dây Zigbee và Esp theo đúng chân file Ccloader.ino trong đó chân Zigbee là P2\_2 được định nghĩa là chân Debug\_Clock (DC) và P2\_1 là Debug\_Data(DD).

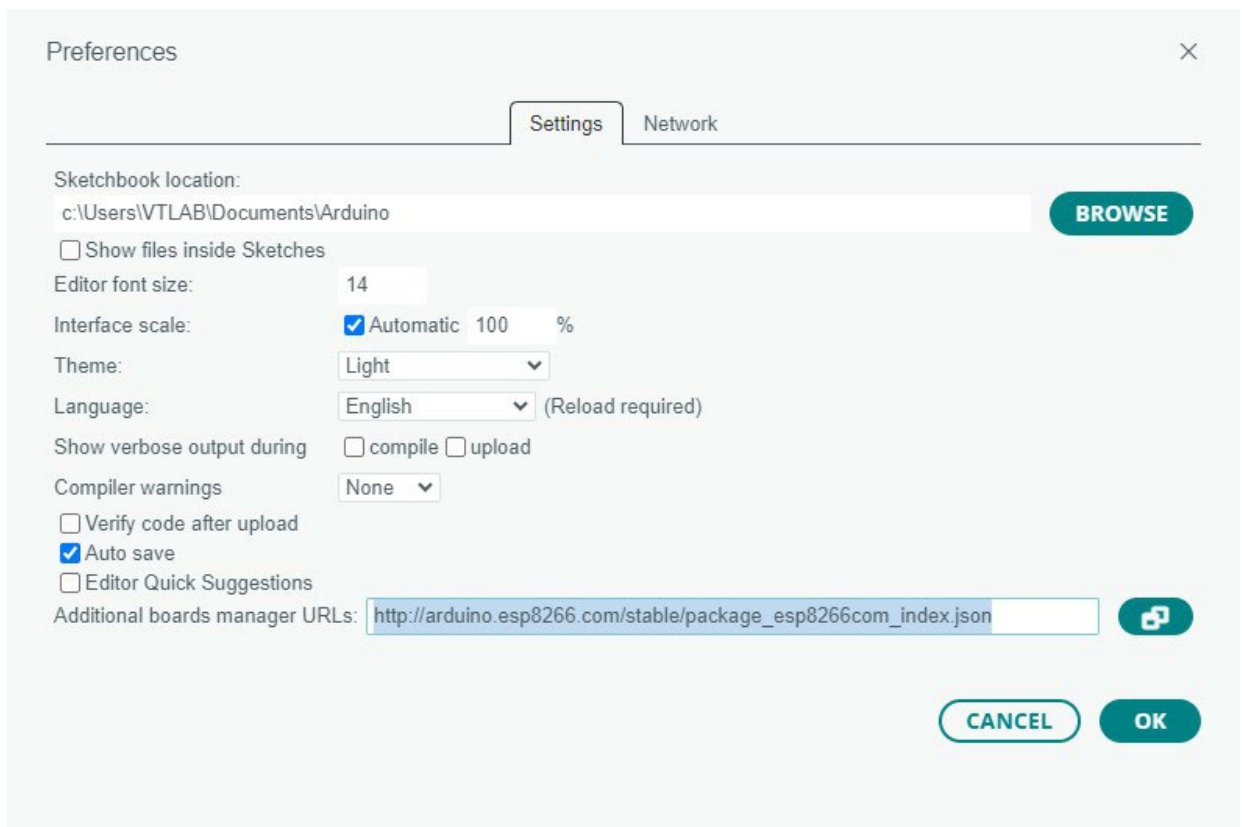
Lưu ý: Board Esp/Wemos phải nạp sẵn cái code của file Ccloader.ino giống như các bạn nạp code cho Arduino Uno

Cách nạp code cho ESP/WeMos sau:

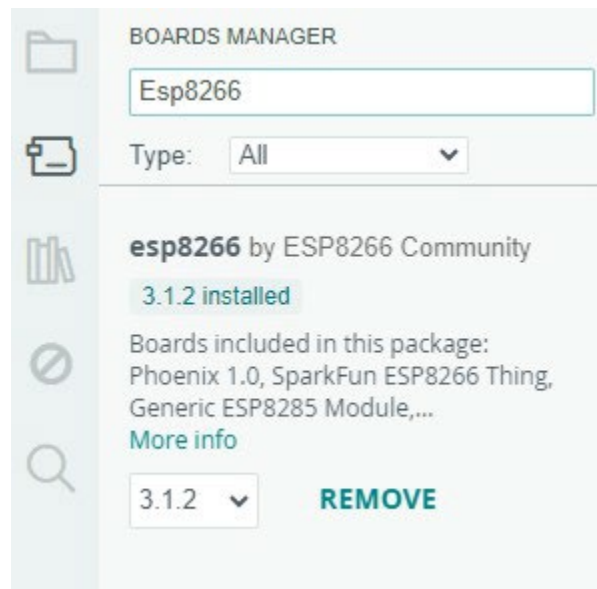
Bước 1: Truy cập Arduino IDE vào File rồi Preference



Bước 2: Copy đường dẫn này và Paste vào chỗ giống trong hình  
[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)

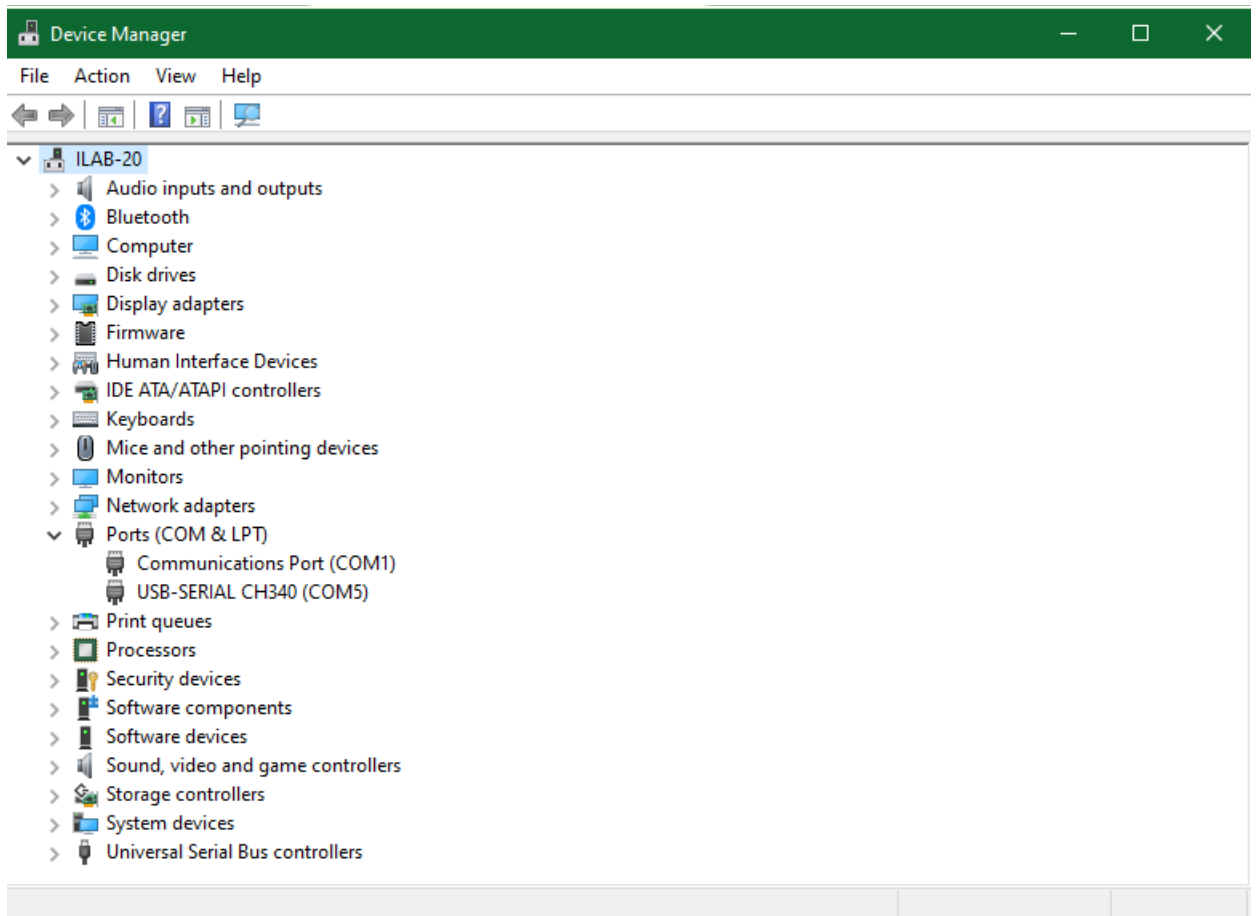


Ấn Ok xong Vào Board Manager ở góc trái xong tìm kiếm “Esp8266” và ấn Install



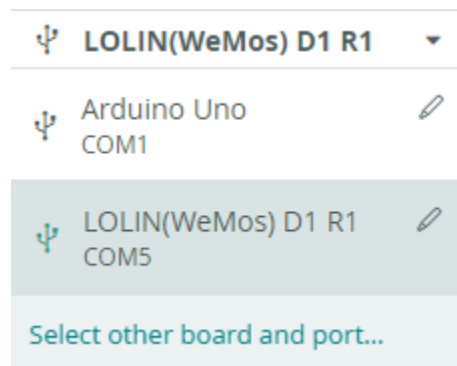
Vậy là xong giờ các bạn có thể nạp Ccloader.ino vào con Esp/Wemos

Lưu ý: Trước khi nạp nhớ chọn cổng COM đúng với cổng ESP/Wemos đã, Cách xem cổng COM bạn vào “Device Manager”

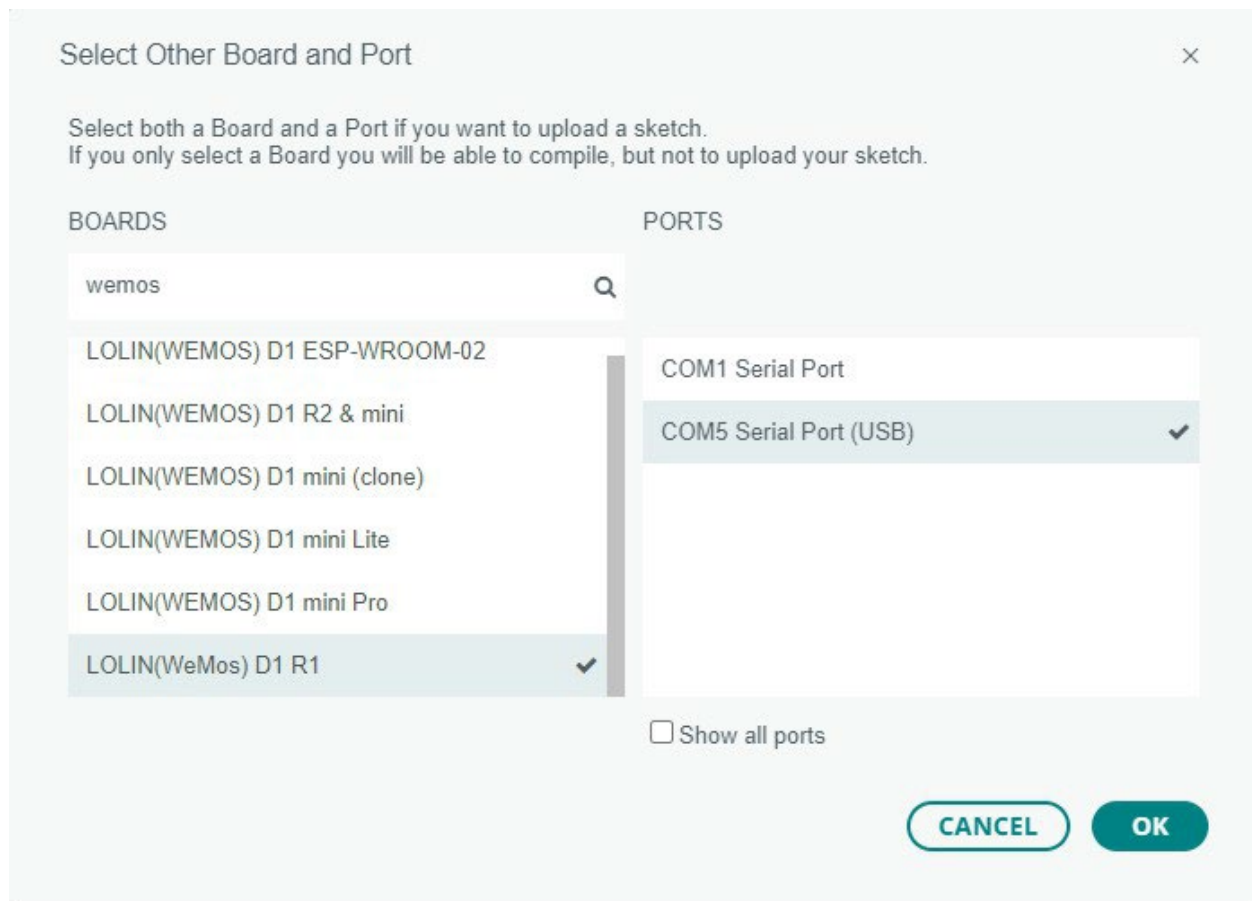


Như trên hình mình đang dùng COM5 thông thường Esp/Wemos nó thường nhận driver là CH340

Vào Chọn Cổng COM và chọn “Select other board and port”.

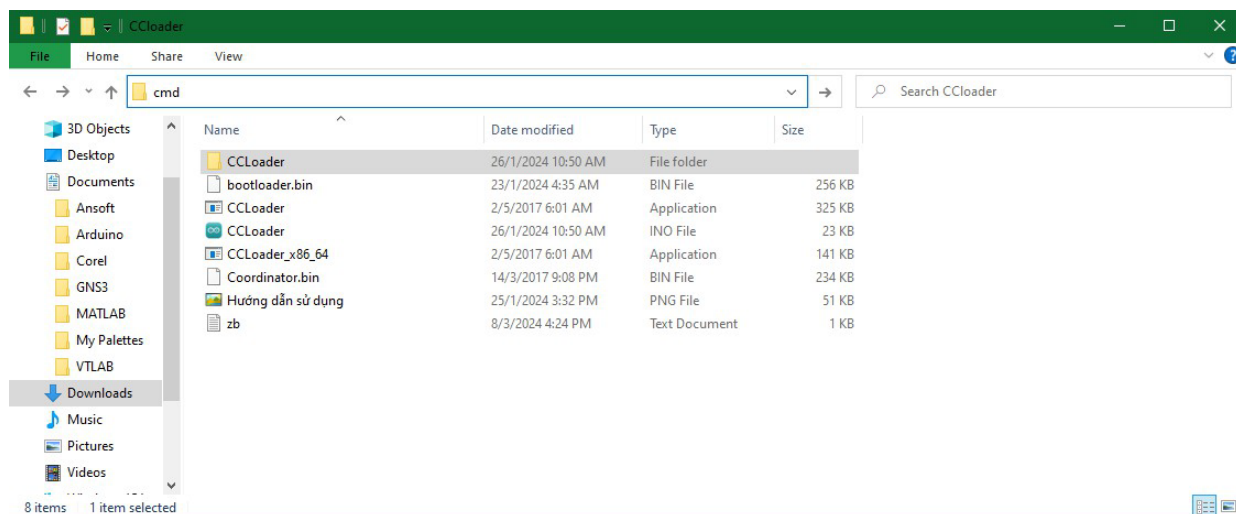


Nhập tên Board các bạn đang cắm nếu các bạn dùng dòng ESP8266 thì là Node MCU 1.0 hoặc 0.9 còn nếu là WeMos thì vẫn là WeMos



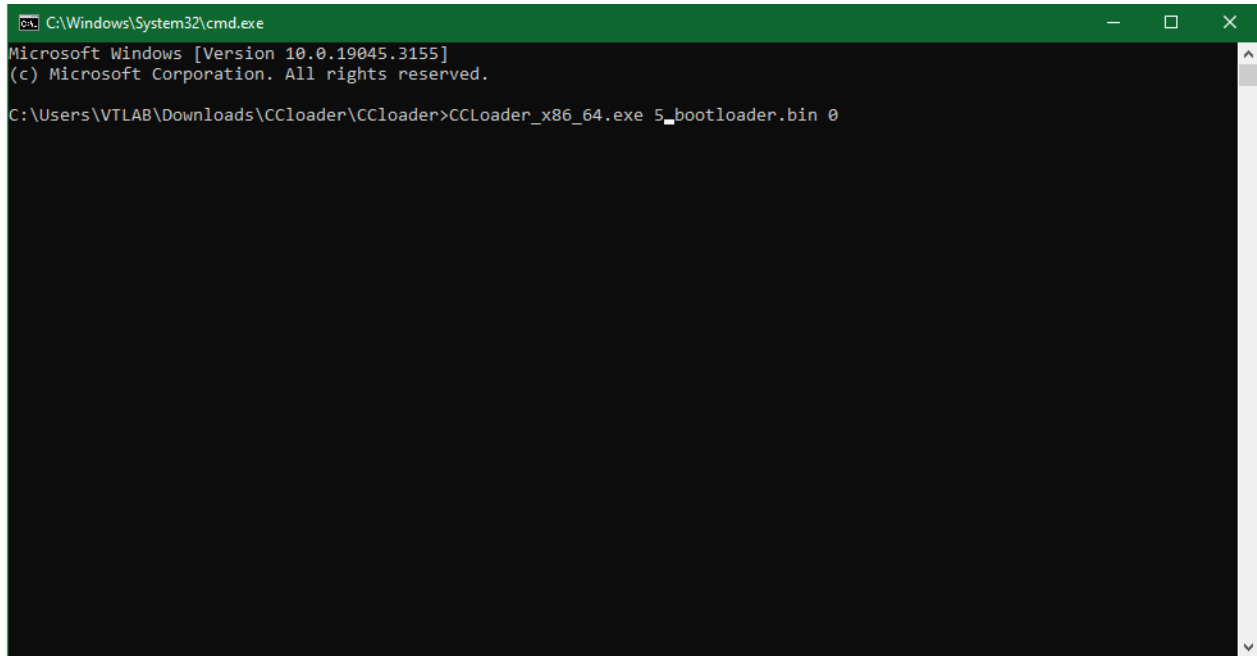
Vậy là xong. Các bạn nạp code file Ccloader.ino với Board ESP hoặc là Wemos

Giờ các bạn vào lại file Ccloader gõ cmd vào mục đường dẫn file



Ấn Enter nó ra cửa sổ của cmd, giờ bạn paste cái này vào: CCLoader\_x86\_64.exe 10  
bootloader.bin 0

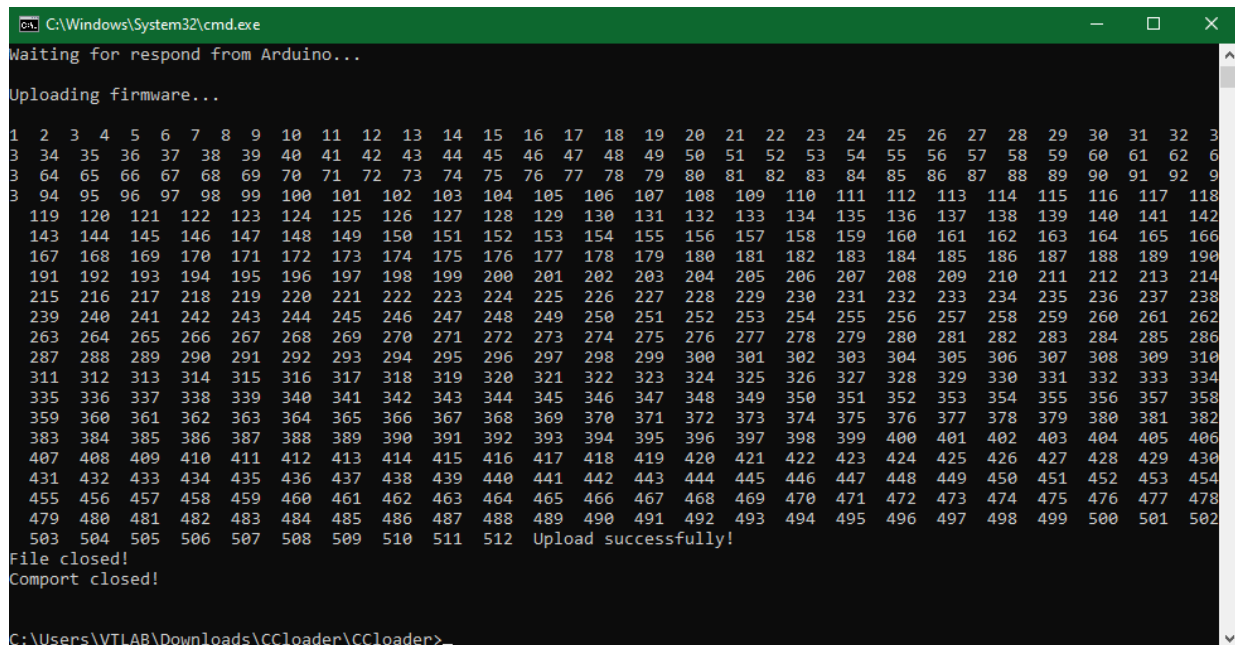
Trong đó “10” là cổng COM bạn đang cắm ESP/Wemos vào máy tính  
Hình ở dưới cửa sổ cmd mình đang paste nó là COM5 nên nó sẽ thế này.



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\VTLAB\Downloads\CCloader\CCloader>CCLoader_x86_64.exe 5_bootloader.bin 0
```

Ấn Enter nó sẽ trông như thế này:



```
C:\Windows\System32\cmd.exe
Waiting for respond from Arduino...

Uploading firmware...

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
3 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63
3 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93
3 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118
119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142
143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166
167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190
191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214
215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238
239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262
263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286
287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310
311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334
335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358
359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382
383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406
407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430
431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454
455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478
479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502
503 504 505 506 507 508 509 510 511 512 Upload successfully!
File closed!
Comport closed!

C:\Users\VTLAB\Downloads\CCloader\CCloader>
```

Lưu ý: Nhớ cắm dây ESP/Wemos với lại Zigbee khi Paste vào CMD (Cách cắm ở trang thứ 8)

## 7.2. Bước 2: Cấu hình Mạng cho Zigbee

Theo lý thuyết, mạng Zigbee có 3 node chính: Coordinator, Router, End-device.

Zigbee được thiết kế để thực hiện giao thức Mesh-network.

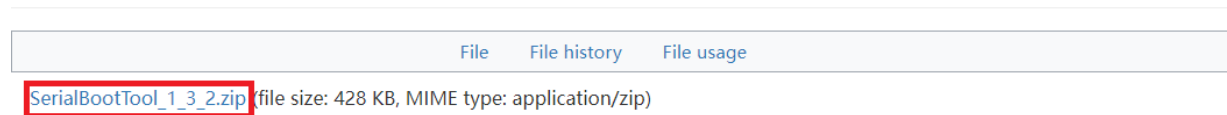
Để chọn node mình muốn trước tiên các bạn cài đặt các phần mềm này đã:

SerialBootTool dùng để cài đặt node mạng các bạn muốn link:

[https://www.waveshare.com/wiki/File:SerialBootTool\\_1\\_3\\_2.zip](https://www.waveshare.com/wiki/File:SerialBootTool_1_3_2.zip)

Ấn vào cái khung đỏ để Download

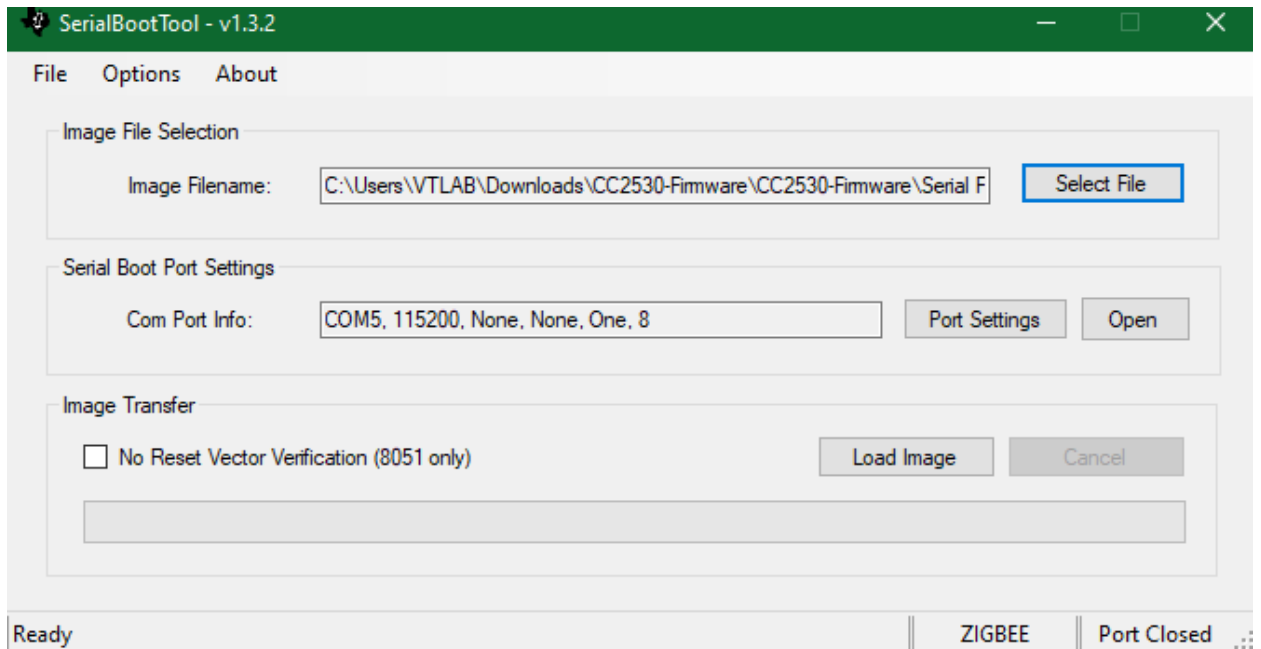
File: **SerialBootTool 1 3 2.zip**



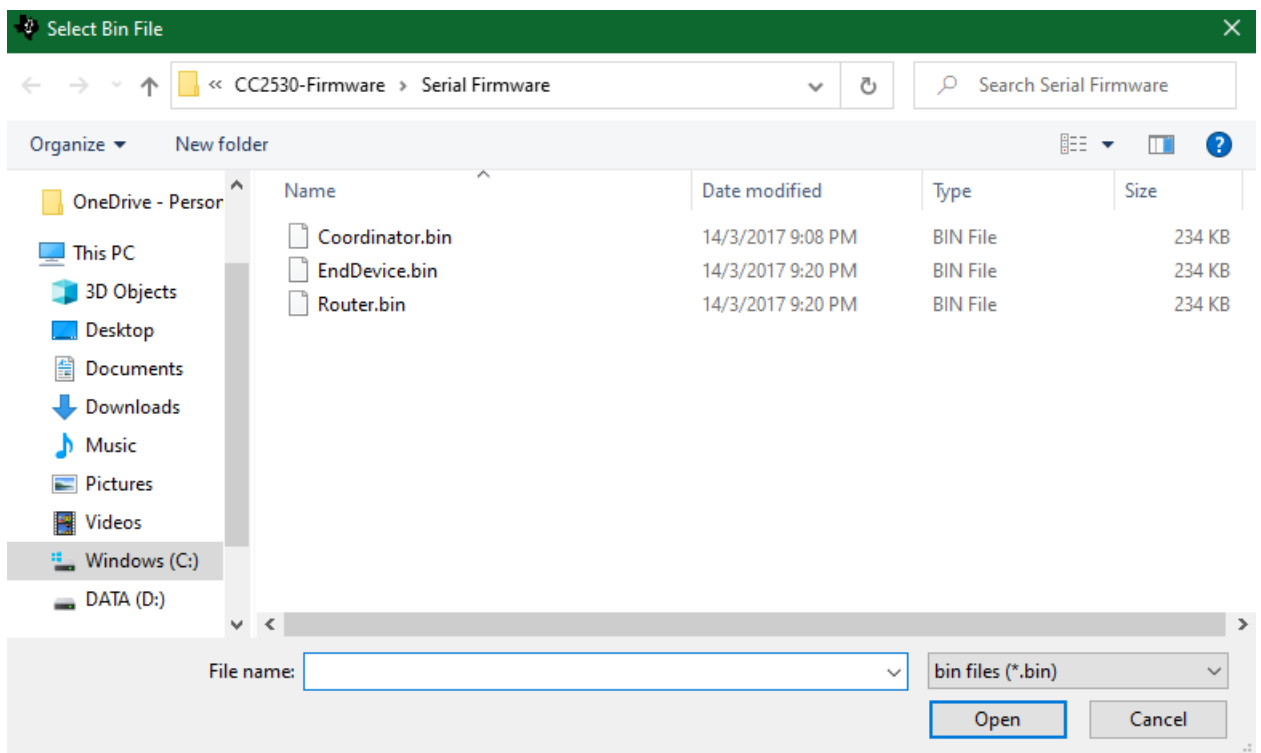
File .bin để cấu hình mạng link:

<https://www.waveshare.com/wiki/File:Core2530-B-Serial-Firmware.7z> Cách tải cũng giống cái SerialBootTool

Giờ hãy mở SerialBootTool lên tại Image File Selection bạn chọn Select file và trở tới file cấu hình mạng, nhớ giải nén file zip ra để có được file .bin. Ví dụ mình sẽ khai báo node mạng mình định cài đặt là Coordinator vì thông thường khi bắt đầu một mạng Zigbee thì sẽ là khai báo Coordinator.



Màn hình giao diện của nó



Còn đây là file chứa cấu hình node mạng

Sau khi các bạn chọn file .bin xong thì các bạn chọn cổng COM mà các bạn cắm USB TTL vào máy tính. Các bạn có thể check cổng COM qua “Device Manager”. Cổng COM của USB TTL thường dùng chip CP210x nên Port của nó sẽ có tên là CP210x

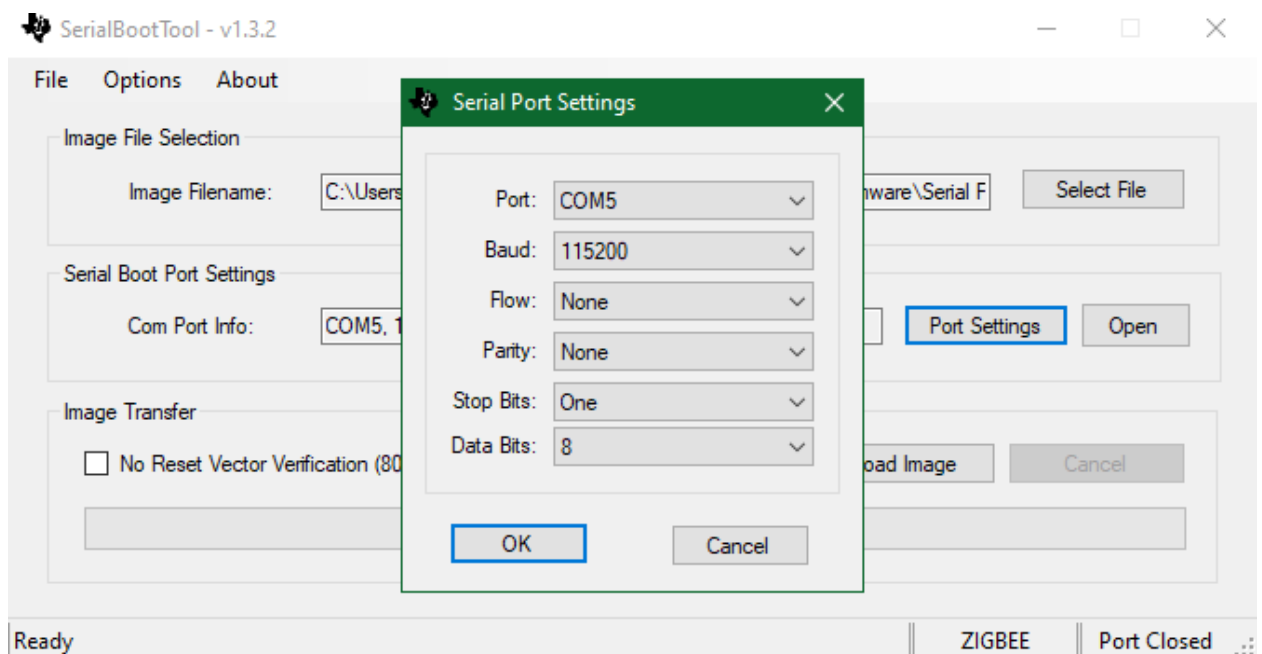


Cách cắm USB TTL giao tiếp với Zigbee như sau. USB TTL được định là một bộ chuyển đổi USB sang UART. Nó được dùng để giao tiếp và lập trình nhiều thiết bị như Node MCU, Lora, Resbery Pi

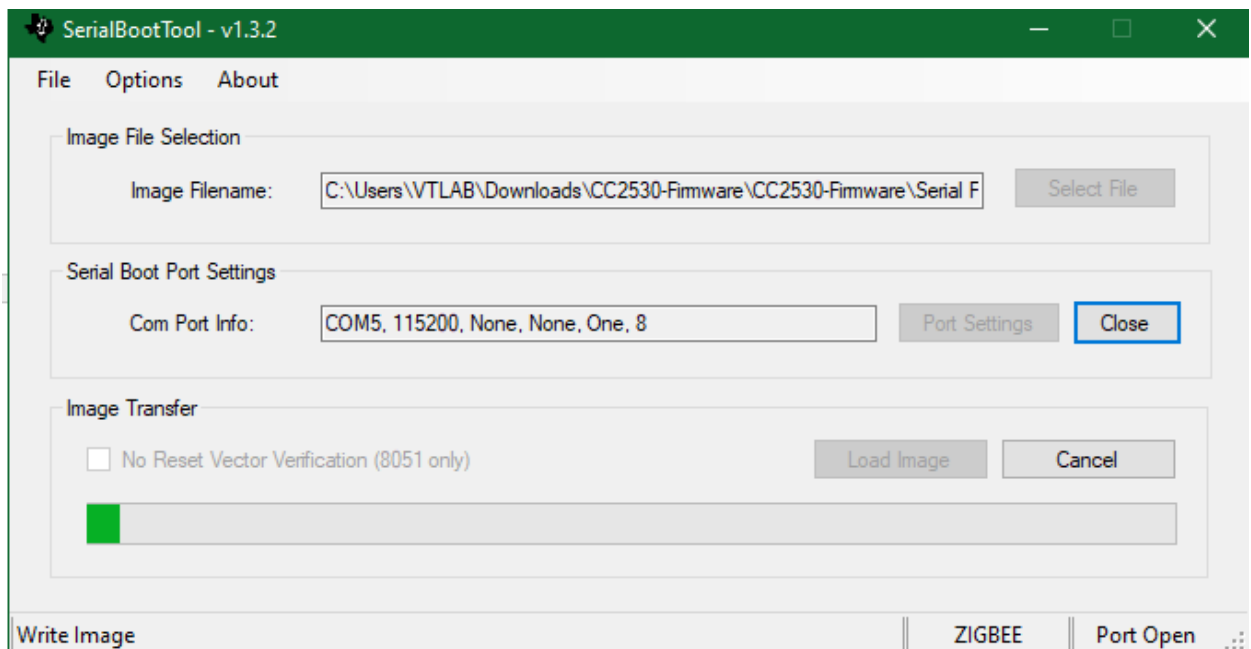


Cách cắm để giao tiếp với Zigbee rất bởi nó chỉ có Nguồn(VCC) và Đất (GND), 2 chân Tx, Rx. Cách cắm như sau.

Zigbee	USB TTL
VCC	3v3
GND	GND
Tx	Rx
Rx	Tx



Sau khi các bạn chọn cổng COM xong thì ấn Load Image để hoàn thành bước cuối



Nếu nó xuất hiện dòng chữ “Write Image” và có thanh màu xanh suốt hiện thì các bạn đang nạp đúng. Còn nếu không xuất hiện gì cả hoặc có dòng Connecting to device thì đang có vấn đề về Firmware các bạn và có thể sẽ phải làm lại bước 1.

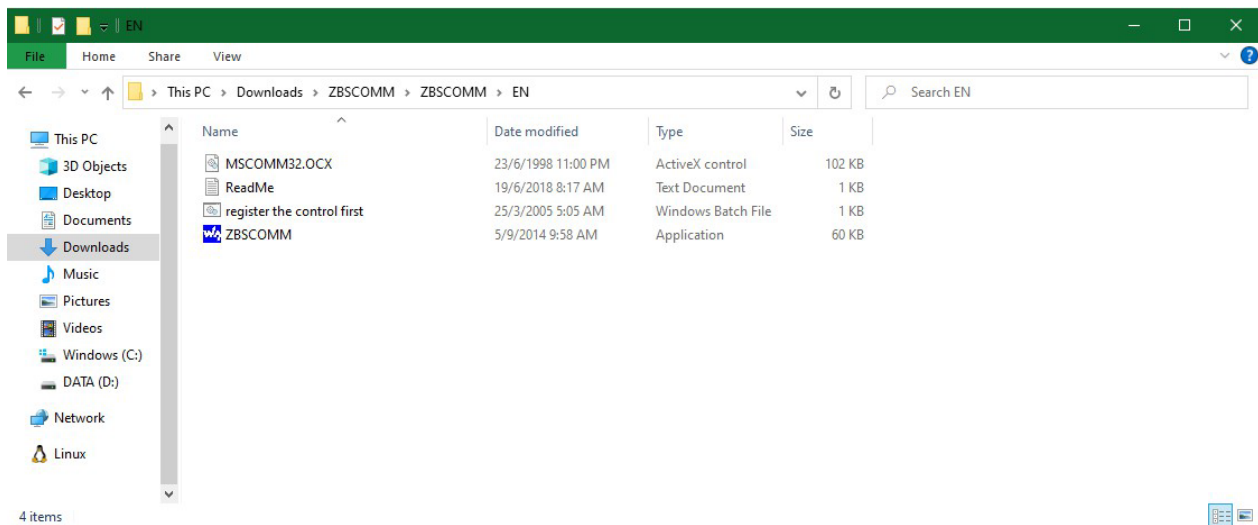
Sau khi các bạn đã Load Image xong thì các bạn nên kiểm tra xem Zigbee thật sự đã đúng chưa

Bước 3: Kiểm tra Zigbee đã có cài đặt được chưa

Để kiểm tra thì các bạn cài thêm phần mềm dưới đây cách cài cũng giống như cài SerialBootTool, link: <https://www.waveshare.com/wiki/File:ZBSCOMM.7z>

Giải nén file Zip trong đó có file “ReadMe” bạn có thể đọc và làm theo hoặc làm theo ở dưới đây

B1: Giống như hướng dẫn đầu tiên bạn vào file ZBSCOM phiên bản EN (ngôn ngữ Anh) và chạy file “register the control” bằng quyền admin



Nó sẽ ra file cửa sổ cmd và nếu nó lỗi thì sang bước tiếp theo còn nếu nó báo thành công thì coi như kết thúc và chỉ cần vào ZBSCOMM.

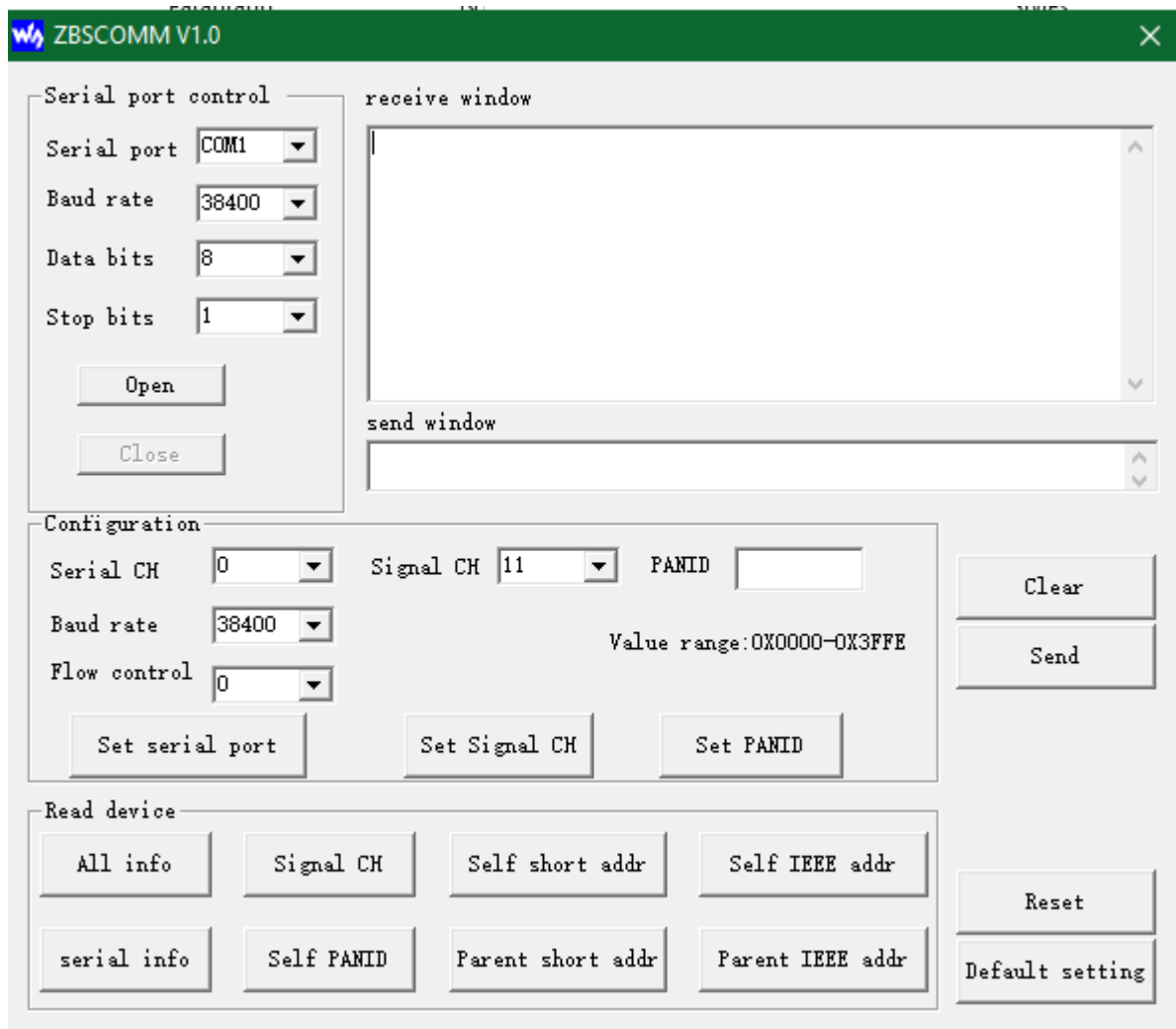
B2:

Copy file “MSCOM32.OCX” và Paste vào đường dẫn C:\Windows\System32. Nếu máy tính ở phiên bản 32-bit. Copy xong thì chạy lại “register the control” bằng quyền admin

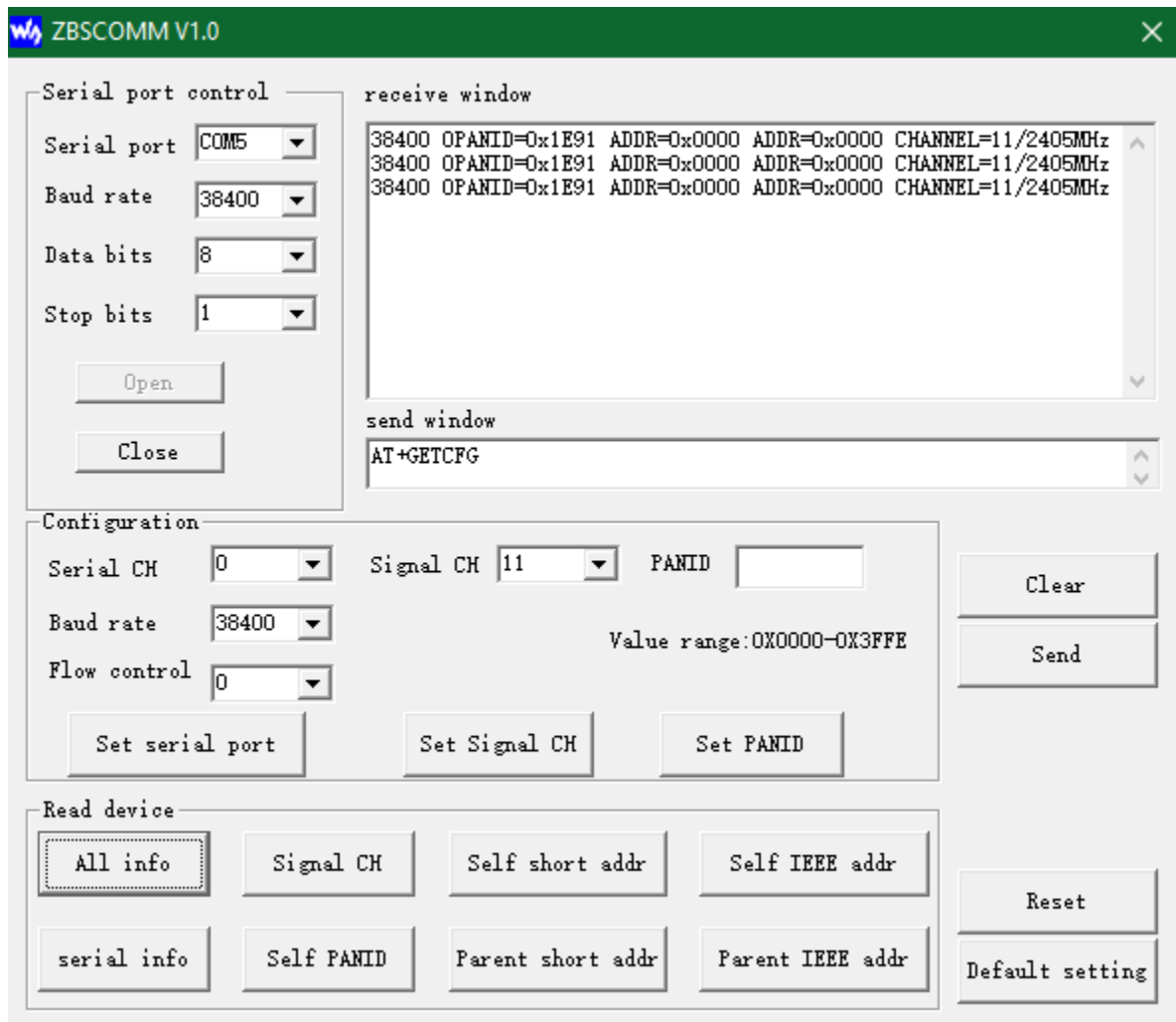
Copy file “MSCOM32.OCX” và Paste vào đường dẫn C:\Windows\SysWOW64. Nếu máy tính ở phiên bản 64-bit. Copy xong thì chạy lại “register the control” bằng quyền admin.

Trong trường hợp máy tính dùng phiên bản 64-bit mà copy và paste xong, chạy “register the control” bằng quyền admin mà vẫn lỗi. Thì copy file MSCOM32.OCX” và paste tiếp vào đường dẫn C:\Windows\System32. Xong chạy lại “register the control” bằng quyền admin.

Khi thông báo thành công thì bạn có thể vào ZBSCOMM được rồi và giao diện của nó trông thế này.



Chọn cổng COM mà bạn đã cắm USB TTL vào máy tính xong đó ấn Open. Rồi ấn vào chỗ “All info” nếu có thông tin phản hồi như địa chỉ (Address), tốc độ Baud, Kênh, PAN ID. Thì có nghĩa là Zigbee có thể được sử dụng và kết thúc nội dung 1.



Vì đang khai báo Zigbee là Coordinator nên nó có địa chỉ đặc chung và duy nhất là 0x0000

## 8. Thực hành truyền dữ liệu từ router về coordinator

**Mục tiêu:** Truyền dữ liệu từ router về coordinator trong mạng Zigbee, trong đó router lấy dữ liệu nhiệt độ độ ẩm từ cảm biến DHT, và truyền về coordinator để hiển thị lên LCD tại coordinator, báo hiệu bằng RGB.

**Lưu ý:** Mỗi nhóm sẽ có một Kit Zigbee router đánh số từ 1-3, và cùng truyền về một coordinator chung

Sơ đồ đấu nối cho Router:

Uno	DHT11
5V	Vcc
GND	GND
D8	Data

Uno	Zigbee
3v3	VCC
GND	GND
D10	Tx
D11	Rx

Sơ đồ đấu nối cho Coordinator:

Uno	LCD I2C
5V	Vcc
GND	GND
SCL	SCL
SDA	SDA

Uno	Zigbee
3v3	VCC
GND	GND
D10	Tx
D11	Rx

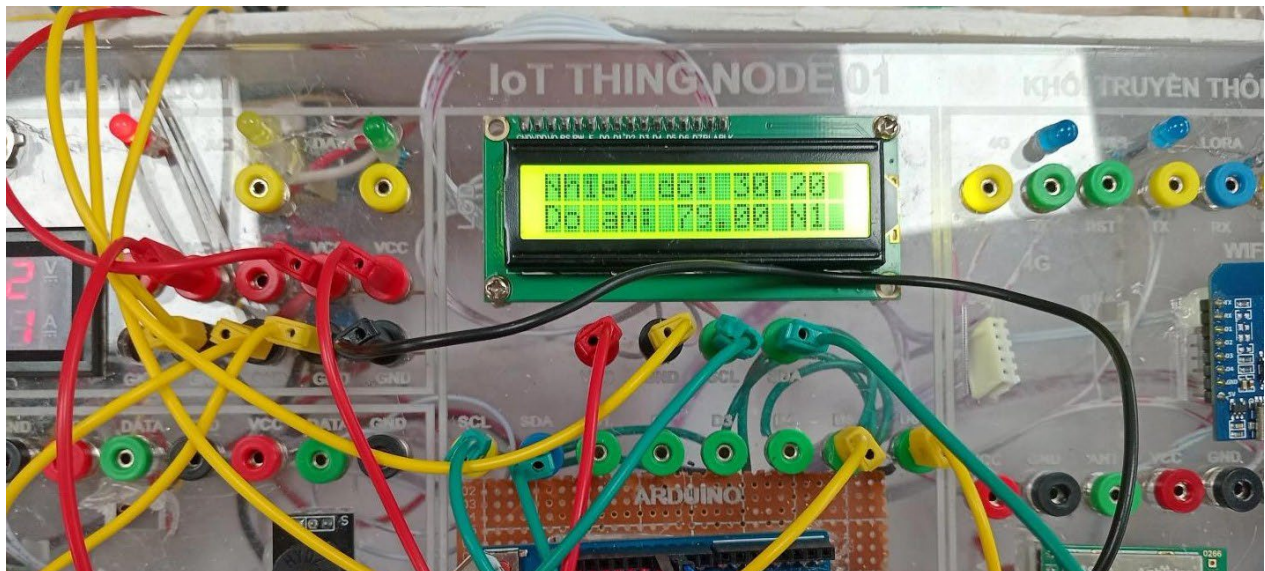
Code Cho Router chuyển tới Coordinator:

```
#include<SoftwareSerial.h>
#include<DHT.h>

#define DHTPIN 8
#define DHTTYPE 11

DHT dht(DHTPIN, DHTTYPE);

int group = 1;
SoftwareSerial mySerial(6, 7); //Tx, Rx void setup() {
    // put your setup code here, to run once:
    Serial.begin(38400); mySerial.begin(38400);
    dht.begin();
}
void loop() {
    float temp = dht.readTemperature(); float hum =
    dht.readHumidity();
    // doc[""]
    mySerial.println("Group: " + String(group));
    mySerial.println("Nhiệt độ: " + String(temp));
    mySerial.println("Độ ẩm: " + String(hum)); delay(5000);
}
```



Kết quả đạt được Khi Router nhận dữ liệu:

## 9. Thực hành truyền tín hiệu điều khiển từ coordinator xuống router

**Mục tiêu:** Truyền tín hiệu điều khiển từ coordinator xuống router trong mạng Zigbee, trong đó ấn nút ấn từ coordinator thì còi tại router sẽ kêu

**Lưu ý:** Mỗi nhóm sẽ có một Kit Zigbee router đánh số từ 1-3, và cùng nhận tín hiệu điều khiển từ một coordinator chung

Sơ đồ đấu nối cho Router:

Uno	Buzzer
D8	VCC
GND	GND

Uno	Zigbee
3v3	VCC
GND	GND
D10	Tx
D11	Rx

Sơ đồ đấu nối cho Coordinator:

Uno	Zigbee
5V	VCC
GND	GND
D10	Tx
D11	Rx

Uno	SW
D8	Data
GND	GND



Code nạp vào Router:

```
#include<SoftwareSerial.h>
```

```
#include<ArduinoJson.h>
```

```
StaticJsonDocument<100> doc;
```

```
SoftwareSerial mySerial(10, 11); //Rx, Tx
```

```
void setup() {
```

```
    // put your setup code here, to run once:
```

```
    Serial.begin(38400); mySerial.begin(38400);
```

```
    pinMode(8, OUTPUT);
```

```
}
```

```
void loop() { if(mySerial.available())
```

```
{
```

```
    String data = mySerial.readStringUntil('\n');
```

```
    data.remove(data.length()-1);
```

```
    if(data == "on")
```

```
{
```

```
        digitalWrite(8, HIGH); Serial.println("Led on");
```

```
}
```

```
    if(data == "off")
```

```
{
```

```
        digitalWrite(8, LOW); Serial.println("Led off");
```

```
}
```

```
}
```

```
}
```