

**ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH**

### TRƯỜNG ĐẠI HỌC BÁCH KHOA KHOA ĐIỆN – ĐIỆN TỬ

**BỘ MÔN ĐIỆN TỬ**

---------------o0o---------------

**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**

**HỒ CÁ THÔNG MINH**

**GVHD: TRẦN HOÀNG QUÂN SVTH: LÊ QUỐC VŨ**

**MSSV: 41304882**

**TP. HỒ CHÍ MINH, THÁNG 01 NĂM 2018**

ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

TRƯỜNG ĐẠI HỌC BÁCH KHOA Độc lập – Tự do – Hạnh phúc.

-----✩----- -----✩-----

Số: /BKĐT Khoa: **Điện – Điện tử** Bộ Môn: **Điện Tử**

N**HIỆM VỤ LUẬN VĂN TỐT NGHIỆP**

1. HỌ VÀ TÊN: LÊ QUỐC VŨ MSSV: 41304882
2. NGÀNH: **ĐIỆN TỬ - VIỄN THÔNG** LỚP : DD13DV06
3. Đề tài: Hồ cá thông minh.
4. Nhiệm vụ (Yêu cầu về nội dung):

.............................................................................................................................................

.............................................................................................................................................

.............................................................................................................................................

.............................................................................................................................................

.............................................................................................................................................

.............................................................................................................................................

5. Ngày giao nhiệm vụ luận văn: ...............................

6. Ngày hoàn thành nhiệm vụ: ...................................

1. Họ và tên người hướng dẫn: Phần hướng dẫn

................................................................. .....................................

................................................................. .....................................

Nội dung và yêu cầu LVTN đã được thông qua Bộ Môn.

*Tp.HCM, ngày…... tháng….. năm 2018*

#### CHỦ NHIỆM BỘ MÔN NGƯỜI HƯỚNG DẪN CHÍNH

**PHẦN DÀNH CHO KHOA, BỘ MÔN:**

Người duyệt (chấm sơ bộ):....................... Đơn vị:......................................................

Ngày bảo vệ : ...........................................

Điểm tổng kết: .........................................

Nơi lưu trữ luận văn: ....................

***LàI CÃM ƠN***

Đe lu¾n văn này hoàn thành em chân thành gui lòi cám ơn sâu sac đen văn phòng b® môn đi¾n tu đã tao đieu ki¾n đe em đưoc tiep c¾n tìm hieu lu¾n văn.

Và thay Tran Hoàng Quân đã t¾n tình hưóng dan trong suot hqc kì đe em có the hoàn thành đưoc đe tài đã đe ra.

Đe đat đưoc đen ngày hôm nay, không the nhac đen sn day bao, hưóng dan t¾n tình cua các thay, cô cua Trưòng Đai hqc Bách khoa TP. HCM, đ¾c bi¾t là các thay cô trong khoa Đi¾n – Đi¾n tu đã giúp em có đưoc các kien thúc đe thnc hi¾n Lu¾n văn tot nghi¾p này.

Cuoi cùng, em xin đưoc cam ơn tat ca ban bè, nhung ngưòi đã bên canh đ®ng viên, giúp đõ em trong suot quá trình hqc t¾p tai trưòng và thòi gian thnc hi¾n Lu¾n văn tot nghi¾p.

Do kien thúc và kinh nghi¾m còn han che, không tránh khoi nhung thieu sót ve n®i dung lan hình thúc, mong quý thay cô góp ý đe em có the hoàn thi¾n san pham cũng như ban thân mình đe tiep tnc phát trien trong tương lai.

*Tp. Ho Chí Minh, ngày 01 tháng 01 năm 2018*

**Lê Quoc Vũ**

**TÓM TAT LU¾N VĂN**

Trong quá trình phát trien cua con ngưòi, nhung cu®c các mang ve công ngh¾ đóng m®t vai trò rat quan trqng, chúng làm thay đoi tùng ngày tùng giò cu®c song cua con ngưòi, theo hưóng hi¾n đai hơn. Đi đôi vói quá trình phát trien cua con ngưòi, nhung thay đoi do chính tác đ®ng cua con ngưòi trong tn nhiên, trong môi trưòng song cũng đang dien ra, tác đ®ng tro lai chúng ta, như ô nhiem môi trưòng, khí h¾u thay đoi, v.v... Dân so càng tăng, nhu cau cũng tăng theo, các d%ch vn, các ti¾n ích tù đó cũng đưoc hình thành và phát trien theo. Đ¾c bi¾t là áp dnng các công ngh¾ cua các ngành đi¾n tu, công ngh¾ thông tin và vien thông vào trong thnc tien cu®c song con ngưòi.

Lu¾n văn này se trình bày ve m®t úng dnng nho cua công ngh¾ đi¾n tu vào thnc tien cu®c song. Mà cn the o đây là su dnng giao thúc truyen nh¾n du li¾u MQTT. Mô hình đieu khien o lu¾n văn này se su dnng giao thúc MQTT gui và nh¾n du li¾u đieu khien thông qua internet đen thiet b% hoat đ®ng.

# Mnc lnc

* 1. GIOI THIfiU 1

[1.1 Tong quan . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1](#_TOC_250024)

* + 1. Tình hình nghiên cúu trong và ngoài nưóc . . . . . . . . . . . . . . . . . . . . 2

[1.3 Nhi¾m vn lu¾n văn . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2](#_TOC_250023)

* 1. LÝ THUYET 3

[2.1 Tìm hieu ve wifi . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3](#_TOC_250022)

* + 1. [Tong quan ve Internet Of Things (IoT) . . . . . . . . . . . . . . . . . . . . . . 5](#_TOC_250021)

[2.2.1 Giói thi¾u ve IoT . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6](#_TOC_250020)

[2.2.2 Cơ h®i và xu hưóng phát trien cua IoT . . . . . . . . . . . . . . . . . . 6](#_TOC_250019)

[2.3 Tìm hieu ve MQTT . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8](#_TOC_250018)

[2.3.1 Tong quan ve MQTT . . . . . . . . . . . . . . . . . . . . . . . . . . . 8](#_TOC_250017)

[2.3.2 Môi trưòng hoat đ®ng MQTT . . . . . . . . . . . . . . . . . . . . . . . 9](#_TOC_250016)

[2.3.3 Tang úng dnng QoS . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9](#_TOC_250015)

* + 1. [M®t so đ%nh nghĩa trong MQTT 10](#_TOC_250014)
    2. [Kí hi¾u su dnng trong giao thúc MQTT 11](#_TOC_250013)
    3. [Nh¾n xét ve MQTT 11](#_TOC_250012)
    4. [M®t so ưu điem cua MQTT so vói các giao thúc IoT khác 12](#_TOC_250011)
  1. [WebSocket 13](#_TOC_250010)
     1. [Tong quan 13](#_TOC_250009)
     2. [Ưu điem 13](#_TOC_250008)
     3. [Nhưoc điem 14](#_TOC_250007)
  2. Giao thúc CoAP 14
     1. [Giói thi¾u ve giao thúc CoAP 14](#_TOC_250006)
     2. [Môi trưòng hoat đ®ng CoAP 15](#_TOC_250005)
  3. Giao thúc TCP 16

iii

* + 1. [Tong quan TCP/IP 16](#_TOC_250004)
    2. [TCP 16](#_TOC_250003)
  1. Giao thúc UDP 18
     1. [M®t so thu¾t ngu UDP 18](#_TOC_250002)
     2. [Cau trúc gói 19](#_TOC_250001)
     3. [Phân bi¾t UDP và TCP 21](#_TOC_250000)
  2. Trang web [*www.cloudmqtt.com*](http://www.cloudmqtt.com/)22
     1. CloudMQTT là gì 22
     2. Tong quan ve trang web [*www.cloudmqtt.com*](http://www.cloudmqtt.com/)22
  3. Tìm hieu phan cúng 24
     1. Module wifi ESP8266 24
     2. Module Arduino Uno R3 27
     3. Vi đieu khien ATmega328 30
     4. Màn hình LCD 20x4 33
     5. Mach chuyen giao tiep LCD sang I2C 35
     6. Đ®ng cơ RC servo 9G 35
     7. Đ®ng cơ bơm P385 36
     8. Led dây 12V 37
     9. Cam bien ánh sáng Photodiod 38
     10. Cam bien lưu lưong nưóc 40
     11. Cam bien rung SW520D 42
     12. Module Relay 5VDC 43
     13. Mach thòi gian thnc RTC DS1307 44
     14. Mach giam áp AMS1117 46

1. THIET KE VÀ THUC HIfiN PHAN CÚNG 47
   1. Yêu cau thiet ke 47
   2. Sơ đo khoi tong quát 48
   3. Ket noi phan cúng 49
      1. Ket noi Arduino và ESP8266 49
      2. LCD vói Arduino 50
      3. Module DS1307 vói Arduino 50
      4. Các cam bien vói Arduino 51
      5. Các nút nhan vói Arduino 52
      6. Bang ket noi chân vi đieu đen các ngoai vi 53
      7. Mô hình 3D cho thiet b% cho các ăn 54
   4. Thnc hi¾n làm mach in 55
      1. Ve mach và layout trên protues 55
      2. Xuat mach in ra file PDF 56
2. THIET KE VÀ THUC HIfiN PHAN MEM 57
   1. Yêu cau đ¾t ra cho phan mem 57
   2. Nap lai firmware cho ESP8266 58
   3. Tao kênh trên cloudMQTT 59
   4. Thiet ke giao di¾n trên đi¾n thoai 61
   5. Lưu đo giai thu¾t 63
3. KET QUÂ THUC HIfiN 65
   1. Phan đieu khien tù xa 65
   2. Mô hình ho cá thông minh 66
4. KET LU¾N VÀ HƯONG PHÁT TRIEN 67
   1. Ket lu¾n 67
   2. Hưóng phát trien 68
5. TÀI LIfiU THAM KHÂO 69
6. PHU LUC 71

**Danh sách hình ve**

2.1 Mô hình Internet of Things . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

2.2 Giao thúc MQTT . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

* 1. Mô hình WebSocket 13
  2. H¾ thong kiem soát 15
  3. Cau trúc gói UDP 20
  4. CloudMQTT 22
  5. Thông tin ket noi tai CloudMQTT 23
  6. Trang quan lý truy nh¾p 23
  7. Ânh thnc te cua ESP8266 24
  8. Sơ đo ra chân cua ESP8266 25
  9. Sơ đo ra chân cua Arduino Uno R3 27
  10. Vi đieu khien ATmega328 30
  11. Sơ đo khoi ATmega328 (Block Diagram) 31
  12. Sơ đo cau trúc AVR CPU 32
  13. Màn hình LCD 20x4 33
  14. Mach chuyen giao tiep LCD sang I2C 35
  15. Đ®ng cơ RC servo 9G 35
  16. Đ®ng cơ bơm P385 36
  17. Led dây 12V 37
  18. Cam bien ánh sáng Photodiod 38
  19. Sơ đo nguyên lí cam bien ánh sáng 39
  20. Cam bien lưu lưong 40
  21. Hi¾u úng Hall 41
  22. Cam bien rung SW520D 42
  23. Nguyên lí hoat đ®ng SW520D 42
  24. Module Relay 43

vi

* 1. Sơ đo nguyên lí 1 relay 44

2.28 RTC DS1307 44

* 1. Sơ đo nguyên lí RTC DS1307 45
  2. Hình anh thnc te mach giam áp 46
  3. Sơ đo nguyên lí mach giam áp AMS 1117 46
  4. Sơ đo khoi phan cúng 48
  5. Ket noi giua ESP8266 vói Arduino Uno R3 49
  6. Ket noi giua DS1307 vói Arduino 50
  7. Mach cam bien rung 52
  8. Mach nút nhan 52
  9. Mô hình cho cá ăn 54
  10. Layout trên protues 55
  11. Xuat mach in ra file PDF 56
  12. Nap firmware MQTT cho ESP8266 58
  13. Thông tin cua tài khoan 59
  14. Tao user 60
  15. Tao topic 60
  16. Cài đ¾t publish trên điên thoai 61
  17. Cài đ¾t subscribe trên đi¾n thoai 62
  18. Lưu đo giai thu¾t phan mem 63
  19. Giao di¾n trên đi¾n thoai 65
  20. Mô hình ho cá thông minh 66

# Danh sách bang

* 1. Sn khác bi¾t giua UDP và TCP 21
  2. Các t¾p l¾nh AT 26
  3. Thông so cơ ban cua Arduino 27
  4. Hoat đ®ng các chân cua LCD 34
  5. Ket noi giua ESP8266 và Arduino Uno R3 49
  6. Ket noi giua module I2C vói Arduino 50
  7. Ket noi giua cam bien ánh sáng vói Arduino 51
  8. Ket noi giua cam bien lưu lưong vói Arduino 51
  9. Bang tóm tat các ket noi đen Arduino Uno R3 53

# Chương 1

## GIéI THIfiU

## Tong quan

IoT hi¾n nay đang là m®t xu hưóng manh me trên toàn the giói, mo ra nhung cơ h®i chưa tùng có cho các nen kinh te, doanh nghi¾p, to chúc và cá nhân đe canh tranh trong môi trưòng mói. Rat nhieu các công ty t¾p đoàn có giá tr% th% trưòng lón đang đau tư vào tý đô la vào nghiên cúu và phát trien lĩnh vnc này trên the giói.

Cha đe cua IoT - ông Kevin Ashton (nhà sáng l¾p Trung tâm Auto-ID tai Vi¾n Công ngh¾ Massachusetts, My) tùng giai thích, IoT đưoc hieu đơn gian là vien canh mà tat ca đo v¾t trên the giói, tù chiec máy bay khong lo cho đen nhung v¾t dnng bình thưòng nho bé trong cu®c song, đeu đưoc ket noi, truyen tai, tương tác và trao đoi du li¾u vói nhau qua mang internet.

Trong giai đoan gan đây giao thúc MQTT đang noi lên như là m®t giao thúc đưoc su dnng pho bien cho giao tiep IoT, rat nhieu hãng công ngh¾ lón đã su dnng, áp dnng và ho tro giao thúc MQTT cho các úng dnng cua mình như: Facebook cho úng dnng Facebook Messenger, IBM, Intel, Amazon, WSO2, Microsoft.

Sau khi nghe thay giói thi¾u và m®t thòi gian tìm hieu ve công ngh¾, ve các phương pháp giao tiep và đieu khien cũng như hưóng phát trien và úng dnng thnc vào thnc te, em quyet đ%nh chqn phương pháp đieu khien thông qua mang internet mà cn the o đây là su dnng module wifi ESP8266-01 truyen nh¾n du li¾u bang phương thúc MQTT.

Chúng ta biet rang khi đòi song con ngưòi càng nâng cao thì nhu cau lai càng đa dang và phong phú hơn. O đây là nhu cau can chăm sóc cho ho cá cua mình. Lu¾n văn này se giai quyet nhu cau đó bang cách thiet ke xây dnng m®t mô hình ho cá thông minh. Vói các van đe xung quanh ho cá như: cho ăn, đèn, bơm nưóc,... bang vi¾c thnc hi¾n đieu khien m®t thiet b% tù xa thông qua Internet bang úng dnng trên đi¾n thoai di đ®ng.

## Tình hình nghiên cNu trong và ngoài nưéc

Theo Gartner, đen năm 2020, the giói se có khoang 20 tý thiet b% su dnng IoT, doanh so dn kien trong năm là 437 tý USD. Các thiet b% này phan lón se chay các thu¾t toán thông minh (AI), ket noi tn đ®ng vói các h¾ thong IT/ERP, quan lý san xuat và mô hình kinh doanh mói. Theo đó, các hãng cũng thay đoi hình thúc kinh doanh tù bán san pham sang tính phí su dnng theo năm trên tùng thiet b%.

“IoT là xu the tat yeu cho quá trình đoi mói và ưóc đoán se mang lai giá tr% tương đương

19.000 tý USD trong vòng 10 năm tói. Theo xu the này, Vi¾t Nam can nhanh chóng nghiên cúu và phát trien các thiet b% IoT, tham gia vào cu®c chơi chung cua nen công ngh¾ toàn cau.” Trương Gia Bình, chu t%ch T¾p đoàn FPT.

Trong tương lai, IoT se úng dnng o tat ca các ngành nghe. Hi¾n tai, IoT t¾p trung vào các d%ch vn san pham trong nhà (Connected Home), tích hop vào h¾ thong công ngh¾ thông tin và công ngh¾ v¾n hành có san (IT/OT Integration) đe nâng cao chat lưong quan lý và năng suat lao đ®ng. Hqc máy thong kê, hqc du li¾u, thu¾t toán phân tích và dn báo thông minh đang đưoc IoT dna vào đe phnc vn các nhu cau khác nhau. Hai san pham đien hình là Google Nest và Amazon Echo. Các công ty lón trong ngành cũng đã và đang đay manh đau tư, xây dnng h¾ sinh thái cho riêng mình và nghiên cúu đe tao ra các san pham IoT mói.

## Nhi¾m vn lu¾n văn

Yêu cau lu¾n văn là thiet ke m®t mô hình ho cá thông minh có the đieu khien trnc tiep tai cho ho¾c đieu khien tù xa thông qua internet.

Nhung n®i dung chính trong lu¾n văn là:

* + - N®i dung 1: Tìm hieu đ%nh nghĩa các lí thuyet ve IOT, Wifi, giao thúc TCP,...
    - N®i dnng 2: Tìm hieu lí thuyet ve giao thúc su dnng IOT, lu¾n văn này lna chqn se su dnng giao thúc MQTT trong lu¾n văn.
    - N®i dung 3: Tìm hieu các module, phan cúng can su dnng trong lu¾n văn như là: ESP8266, Arduino,...
    - N®i dung 4: Tìm giai thu¾t cho phan mem, thú tn các múc đieu khien, hien th% chúc năng cua tùng khoi.
    - N®i dung 5: Tien hành thiet ke mach in và giao di¾n úng dnng trên đi¾n thoai.

# Chương 2

## LÝ THUYET

## Tìm hieu ve wifi

Wi-Fi viet tat tù Wireless Fidelity hay mang 802.11 là h¾ thong mang không dây su dnng sóng vô tuyen, giong như đi¾n thoai di đ®ng, truyen hình và radio.

H¾ thong này đã hoat đ®ng o m®t so sân bay, quán café, thư vi¾n ho¾c khách san. H¾ thong cho phép truy c¾p Internet tai nhung khu vnc có sóng cua h¾ thong này, hoàn toàn không can đen cáp noi. Ngoài các điem ket noi công c®ng (hotspots), WiFi có the đưoc thiet l¾p ngay tai nhà riêng.

Tên gqi 802.11 bat nguon tù vi¾n IEEE (Institute of Electrical and Electronics Engineers). Vi¾n này tao ra nhieu chuan cho nhieu giao thúc ky thu¾t khác nhau, và nó su dnng m®t h¾ thong so nham phân loai chúng; 6 chuan thông dnng cua WiFi hi¾n nay là 802.11a/b/g/n/ac/ad.

##### *Sóng Wifi:*

Các sóng vô tuyen su dnng cho WiFi gan giong vói các sóng vô tuyen su dnng cho thiet b% cam tay, đi¾n thoai di đ®ng và các thiet b% khác. Nó có the chuyen và nh¾n sóng vô tuyen, chuyen đoi các mã nh% phân 1 và 0 sang sóng vô tuyen và ngưoc lai.

Tuy nhiên, sóng WiFi có m®t so khác bi¾t so vói các sóng vô tuyen khác o cho:

Chúng truyen và phát tín hi¾u o tan so 2.4 GHz ho¾c 5 GHz. Tan so này cao hơn so vói các tan so su dnng cho đi¾n thoai di đ®ng, các thiet b% cam tay và truyen hình. Tan so cao hơn cho phép tín hi¾u mang theo nhieu du li¾u hơn.

Chúng dùng chuan 802.11:

* Chuan 802.11b là phiên ban đau tiên trên th% trưòng. Đây là chuan ch¾m nhat và re tien nhat, và nó tro nên ít pho bien hơn so vói các chuan khác. 802.11b phát tín hi¾u o tan so 2.4 GHz,

nó có the xu lý đen 11 megabit/giây, và nó su dnng mã CCK (complimentary code keying).

* Chuan 802.11g cũng phát o tan so 2.4 GHz, nhưng nhanh hơn so vói chuan 802.11b, toc đ® xu lý đat 54 megabit/giây. Chuan 802.11g nhanh hơn vì nó su dnng mã OFDM (orthogonal frequency-division multiplexing), m®t công ngh¾ mã hóa hi¾u qua hơn.
* Chuan 802.11a phát o tan so 5 GHz và có the đat đen 54 megabit/ giây. Nó cũng su dnng mã OFDM. Nhung chuan mói hơn sau này như 802.11n còn nhanh hơn chuan 802.11a, nhưng 802.11n van chưa phai là chuan cuoi cùng.
* Chuan 802.11n cũng phát o tan so 2.4 GHz, nhưng nhanh hơn so vói chuan 802.11a, toc đ® xu lý đat 300 megabit/giây.
* Chuan 802.11ac phát o tan so 5 GHz.
* Chuan 802.11ad phát o tan so 60 GHz.

WiFi có the hoat đ®ng trên ca ba tan so và có the nhay qua lai giua các tan so khác nhau m®t cách nhanh chóng. Vi¾c nhay qua lai giua các tan so giúp giam thieu sn nhieu sóng và cho phép nhieu thiet b% ket noi không dây cùng m®t lúc.

Tuy nhiên mang không dây luôn vap phai nhung van đe ton tai khien nó chưa the phát trien và thay the cho toàn b® cách ket noi hi¾n nay.

Nhưng van đe ti¾n loi cho ngưòi dùng se luôn đay công ngh¾ không dây ngày càng phát trien và chúng ta có the khang đ%nh rang mang không dây se ngày càng phát trien và thay the dan dan mang su dnng cáp hi¾n nay.

##### *Pham vi Wifi:*

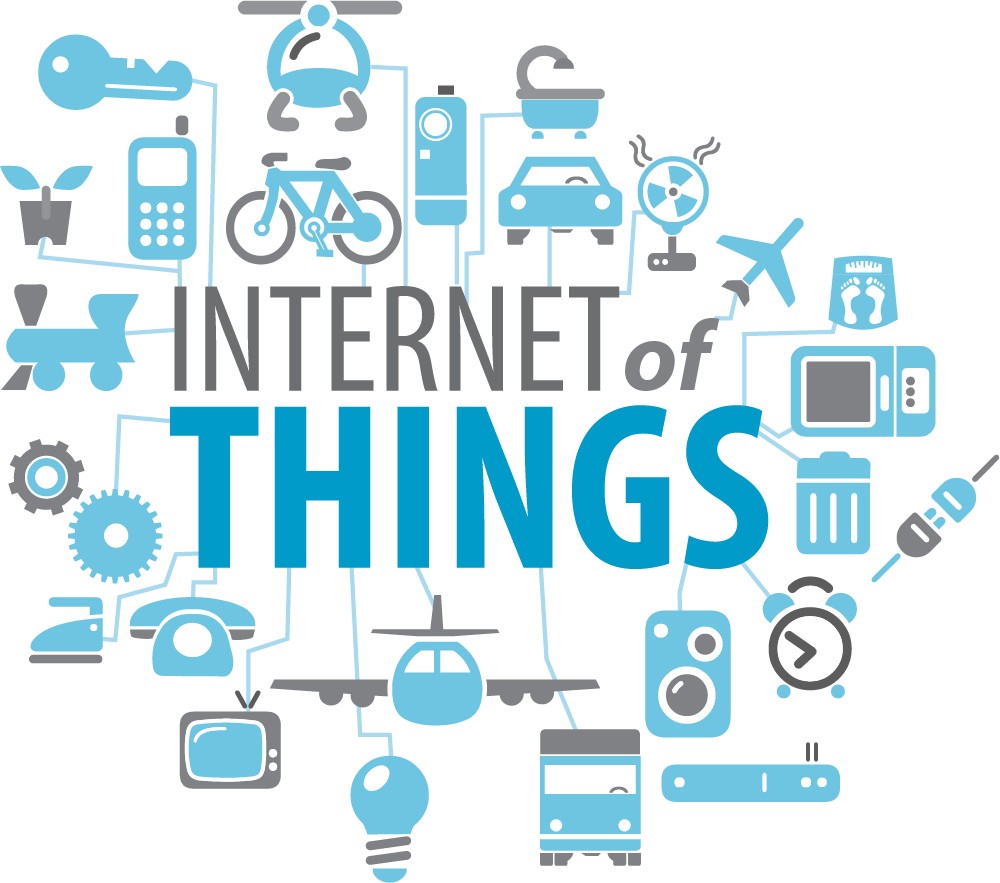
Pham vi tín hi¾u Wifi phn thu®c vào bang tan, công suat đau ra, đ® loi và loai anten cũng như các ky thu¾t đieu che. Line-of-sight là hưóng dan thu nho nhưng sn khúc xa và phan xa gây ra anh hưong đáng ke.

M®t điem truy c¾p phù hop vói m®t trong hai chuan 802.11b hay 802.11g , su dnng ăng-ten chúng khoán có the có m®t pham vi 100 m (330 ft). Các đài phát thanh cùng vói m®t ăng-ten parabol bán bên ngoài (15 dB đat đưoc) có the có m®t pham vi hơn 20 d¾m.

Tý l¾ đ® loi cao hơn (dBi) cho thay đ® l¾ch hơn nua (thưòng ve hưóng ngang) tù m®t lý thuyet, búc xa đang hưóng hoàn hao , và vì the hơn nua anten có the dn đoán m®t tín hi¾u thích hop, cũng như so sánh công suat đau ra tương tn trên m®t ăng-ten đang hưóng hơn. Ví dn, m®t ăng-ten 8 dBi su dnng vói m®t trình đieu khien 100 mW se có m®t pham vi ngang tương tn như m®t ăng-ten 6 dBi đưoc thúc đay tai 500 mW. Lưu ý rang đieu này gia đ%nh rang búc xa theo

chieu thang đúng b% mat; đieu này có the không phai là trưòng hop trong m®t so tình huong, đ¾c bi¾t là trong các tòa nhà lón ho¾c trong m®t ong dan sóng . Trong ví dn trên, m®t ong dan sóng đang hưóng có the gây ra cho anten công suat thap 6 DBi đưa ra nhieu hơn m®t đang hưóng so vói anten 8 dBi mà không phai là trong m®t ong dan sóng, ngay ca khi hq đưoc ca hai đang đưoc thúc đay o 100 mW.

## Tong quan ve Internet Of Things (IoT)



Hình 2.1: Mô hình Internet of Things

Trong khoang hai năm tro lai đây, chac han chúng ta đã không ít lan nghe thoáng qua ve khái ni¾m IoT, hay bat g¾p tin túc ve các san pham đưoc quang cáo là phnc vn cho nhu cau “smart home” nhà thông minh. Trong đó, nhung thiet b% gia dnng như lò nưóng hay tu lanh có the "nói chuy¾n" đưoc vói nhau. Nhưng chúng ket noi vói nhau như the nào, và li¾u xu hưóng này có thnc sn bùng no, đưa chúng ta đen m®t the giói tương lai như trong game hay phim anh? Trong boi canh mà hàng ngày càng nhieu chung loai thiet b% đưoc gán mác “thông minh - smart” và thi nhau “lên mây-cloud” như hi¾n nay, se là không thùa khi chúng ta tìm hieu các kien thúc căn ban ve h¾ sinh thái IoT này.

### Giói thi¾u ve IoT

IoT là khái ni¾m đe chi các thiet b% có kha năng ket noi vói nhau, ket noi vói Internet và tao nên m®t mang lưói các thiet b% thông minh phnc vn cho cu®c song ngưòi dùng.

IoT là thu¾t ngu dùng đe chi các đoi tưong có the đưoc nh¾n biet cũng như chi sn ton tai cua chúng trong m®t kien trúc mang tính ket noi. Cnm tù này đưoc đưa ra boi Kevin Ashton vào năm 1999. Ông là m®t nhà khoa hqc đã sáng l¾p ra Trung tâm Auto-ID o đai hqc MIT, nơi thiet l¾p các quy chuan toàn cau cho RFID cũng như m®t so loai cam bien khác.

IoT có anh hưong và úng dnng rat r®ng rãi trên nhieu lĩnh vnc như: quan lý ha tang, y te, xây dnng và tn đ®ng hóa, giao thông, quân sn, truyen thông, tài chính. . .

Cn the trong lĩnh vnc y te, các thiet b% IoT có the đưoc su dnng đe cho phép theo dõi súc khoe tù xa và h¾ thong thông báo khan cap. Các thiet b% theo dõi súc khoe như đo huyet áp, nh%p tim hay các thiet b% tiên tien có kha năng giám sát cay ghép đ¾c bi¾t như máy đieu hòa nh%p tim, máy tro thính. . . đang đưoc đưa vào nghiên cúu, san xuat đe phnc vn cho nhu cau cua con ngưòi m®t cách toi ưu nhat.

Nhò IoT, có nhieu tưong đ®c đáo đã tro thành hi¾n thnc. Không chi phát huy tot hơn cho công dnng von có cua thiet b%, IoT còn góp phan đem lai cu®c song tot đep hơn cho con ngưòi. Có the nói, loi ích cua IoT đoi vói cu®c song cua con ngưòi hi¾n đai là vô cùng phong phú.

Không nhung chi quan tâm chăm lo ve van đe súc khoe, mà còn hưóng đen nhung loi ích khác ve m¾t an ninh, giáo dnc.

Theo như dn báo cua các chuyên gia ve tiem năng cua IoT đen năm 2020, hq cho rang, IoT se càng ngày càng phát trien manh me hơn nua. IoT đã và đang tro thành chìa khóa thành công trong tương lai.

### Cơ h®i và xu hưóng phát trien cua IoT

IoT là xu hưóng phát trien manh me cua the giói và đã bat đau xuat hi¾n tai Vi¾t Nam như nhà thông minh, giao thông thông minh, đô th% thông minh. . . Trên nen tang cua bon trn c®t công ngh¾ là mang xã h®i, công ngh¾ di đ®ng, phân tích du li¾u lón, đi¾n toán đám mây, IoT đưoc dn báo se đem lai m®t ký nguyên mói bùng no ca ve so lưong ket noi vói khoang 80 tý ket noi vào năm 2020 cũng như các d%ch vn , úng dnng trên nen Internet, tao đ®ng lnc phát trien kinh te, đoi mói công ngh¾, tù đó tác đ®ng đen mqi m¾t đòi song kinh te xã h®i cua các quoc gia.

*Bùng no nhung trang chia se c®ng đong* vói chúc năng chia se anh, âm nhac hay video nham

kéo mqi ngưòi gan lai nhau hơn trong m®t “the giói phang” yêu cau m®t mang lưói truyen tai có dung lưong lón đe đam bao yêu cau ngày m®t cao hơn cua ngưòi dùng.

*Nhieu hơn nhung trang tìm kiem, chia sé và lưu tru thông tin*, thông tin ngày m®t nhieu lên yêu cau m®t h¾ thong lưu tru có the thoa mãn yêu cau cua ngưòi dùng.

*Nhung trang chia sé đa phương ti¾n* nhung trang úng dnng này yêu cau dung lưong lón ca ve van đe lưu tru cũng như truyen tai.

*Nguon loi mói trong lĩnh vnc kinh doanh* IoT mo ra cơ h®i đoi mói công ngh¾ và ưóc đoán se mang lai các giá tr% tương đương 19.000 tý USD trong vòng 10 năm tói. Hi¾n nay có khoang 8 tý các thiet b% ket noi, nhưng đen năm 2020 se lên tói 80 tý thiet b%. M®t so ý tưong và san pham ve IoT đã bat đau xuat hi¾n như nhà thông minh, các thiet b% đieu khien trong gia đình, giao thông.

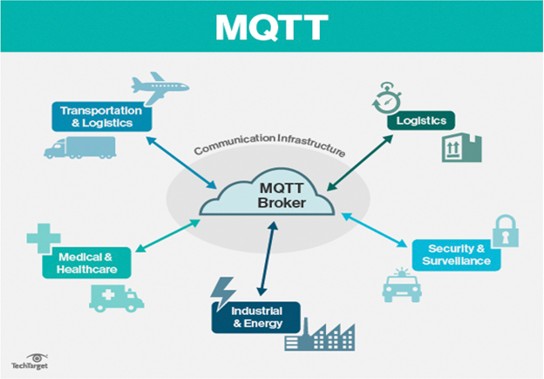
Sn phát trien cua IoT tao ra bon bưóc chuyen d%ch trong vai trò cua các nhà khai thác vien thông. Vai trò đau tiên là thu th¾p du li¾u đe nâng cao hi¾u qua n®i b® như h¾ thong báo cáo và roaming. Vai trò thú hai là phân tích thông tin tương tác cua khách hàng, đe cung cap nhung d%ch vn IoT mang tính cá nhân cho các thuê bao cua mình. Vai trò thú ba là su dnng cơ so du li¾u phân tích là giá tr%, ket noi vói các công ty cung cap d%ch vn o lĩnh vnc khác tao ra san pham hi¾u qua. Vai trò thú tư là cung cap d%ch vn quan lý du li¾u cho các ket noi IoT, làm cau noi giua các công ty cung cap úng dnng IoT vói chính các ket noi IoT có SIM và không có SIM đe các bên đeu mua đưoc d%ch vn mình can và bán đưoc d%ch vn mình có m®t cách hi¾u qua.

Đe có the trien khai IoT thành công và ben vung, can phai cân nhac đen bon yeu to là nen tang phan mem, h¾ sinh thái giua các ngành, quá trình chuan hóa ve công ngh¾ và giai quyet đưoc nhung lo lang cua khách hàng liên quan tói đam bao tính riêng tư và an toàn.

IoT đang tro thành xu hưóng toàn cau, đưoc dn đoán là se tao ra sn bùng no kinh te mói và là th% trưòng thiet b% lón nhat trên the giói. Ưóc tính đen năm 2019, th% trưòng này se gap đôi quy mô th% trưòng smartphone, PC, tablet, xe hơi ket noi và th% trưòng các thiet b% đeo trên ngưòi c®ng lai. IoT se mang lai 1,7 nghìn tý USD giá tr% gia tăng cho nen kinh te toàn cau trong 4 năm tói, bao gom phan cúng, phan mem, chi phí lap đ¾t, d%ch vn quan lý , và giá tr% kinh te tao ra tù hi¾u qua cua IoT.

**Các công ngh¾ nen tang cua IoT** là: RFID, Sensor, Smart, Nano. Sn phát trien cua nhung công ngh¾ nói trên cũng có the đưoc xem xét như là m®t xu hưóng phát trien cua IoT.

## Tìm hieu ve MQTT



Hình 2.2: Giao thúc MQTT

### Tong quan ve MQTT

MQTT đưoc tao ra boi Tien sĩ Andy Standford-Clark cua IBM và Arlen Nipper of Arcom (bây giò là Euro tech) năm 1999. Tính đen thòi điem hi¾n tai, MQTT phiên ban 3.1.1 đưoc công nh¾n chuan OASIS.

MQTT (Message Queuing Telemetry Transport) là m®t giao thúc gui dang publish/subscribe su dnng cho các thiet b% Internet of Things vói băng thông thap, đ® tin c¾y cao và kha năng đưoc su dnng trong mang lưói không on đ%nh.

M®t broker có the đăng ký su dnng mien phí tai trang https:[*//www.cloudmqtt.com/*](http://www.cloudmqtt.com/)

Boi vì giao thúc này su dnng băng thông thap trong môi trưòng có đ® tre cao nên nó là m®t giao thúc lý tưong cho các úng dnng M2M. MQTT cũng là m®t giao thúc đưoc úng dnng trong Facebook Messenger.

Trong m®t h¾ thong su dnng giao thúc MQTT, nhieu node tram (gqi là MQTT client – gqi tat là client) ket noi tói m®t MQTT server (gqi là broker). Moi client se đăng ký m®t vài

kênh (topic), ví dn như “/client1/channel1”, “/client1/channel2”. Quá trình đăng ký này gqi là

**“subscribe”**, giong như chúng ta đăng ký nh¾n tin trên m®t kênh Youtube v¾y.

Moi client se nh¾n đưoc du li¾u khi bat kỳ tram nào khác goi du li¾u và kênh đã đăng ký.

Khi m®t client goi du li¾u tói kênh đó, gqi là **“publish”**.

### Môi trưòng hoat đ®ng MQTT

Boi vì MQTT không đưoc thiet ke vói an ninh trong tâm trí, giao thúc có truyen thong đưoc su dnng trong các mang lưói an toàn, back-end cho các mnc đích úng dnng cn the. Cau trúc đe MQTT có the de dàng tao thành m®t cái cây khong lo và không có cách nào rõ ràng làm the nào đe phân chia m®t cây thành các mien logic nho hơn có the đưoc liên. Đieu này làm cho nó khó khăn đe tao ra m®t mang lưói MQTT kha năng mo r®ng trên toàn cau, boi vì như kích thưóc cua các chu đe cây trong, đ® phúc tap tăng lên.

### Tang úng dnng QoS

**QoS** : Qualities of service, có 3 tùy chqn khi publish và subscribe: QoS0, QoS1, QoS2.

* + - * QoS0 *“Fire and forget”* Broker/client se goi du li¾u đúng 1 lan, quá trình goi đưoc xác nh¾n boi chi giao thúc TCP/IP.
      * QoS1 *“Delivered at least once”* Broker/client se goi du li¾u vói ít nhat 1 lan xác nh¾n tù đau kia, nghĩa là có the có nhieu hơn 1 lan xác nh¾n đã nh¾n đưoc du li¾u.
      * QoS2 *“Delivered exactly once”* Broker/client đam bam khi goi du li¾u thì phía nh¾n chi nh¾n đưoc đúng 1 lan, quá trình này phai trai qua 4 bưóc bat tay.

M®t gói tin có the đưoc goi o bat kỳ QoS nào, và các client cũng có the subscribe vói bat kỳ yêu cau QoS nào. Có nghĩa là client se lna chqn QoS toi đa mà nó có đe nh¾n tin. Ví dn, neu 1 gói du li¾u đưoc publish vói QoS2, và client subscribe vói QoS0, thì gói du li¾u đưoc nh¾n ve client này se đưoc broker goi vói QoS0, và 1 client khác đăng ký cùng kênh này vói QoS2, thì nó se đưoc Broker goi du li¾u vói QoS2. M®t ví dn khác, neu 1 client subscribe vói QoS2 và gói du li¾u goi vào kênh đó publish vói QoS0 thì client đó se đưoc Broker goi du li¾u vói QoS0. QoS càng cao thì càng đáng tin c¾y, đong thòi đ® tre và băng thông đòi hoi cũng cao hơn.

### M®t so đ%nh nghĩa trong MQTT

**RETAIN** : Chi đưoc su dnng o PUBLISH. Mnc đích đe báo vói Server biet nên giu lai message sau khi chuyen đen các Subscriber hi¾n tai. Message giu lai se đưoc su dnng đe gui đen các Subsriber mói subsriber topic mà message này thu®c ve.

Trưòng này se có tác dnng trong trưòng hop khi mà Publisher can thông báo đen các Sub- sribers mói biet ve sn ton tai cua nó. Nó giong như là đe lai dau vet.

Tuy nhiên, đoi vói các Subsriber đang ket noi đen Server thì trưòng Retain này se không đưoc Set. Đieu này giúp Subscriber phân bi¾t đưoc đâu là message “mói toanh”, đâu mà message đưoc giu tù trưóc đó.

Các message phai đưoc lưu lai ngay ca khi khoi đ®ng lai server.

Đe xóa message đã yêu cau lưu lai, Publisher chi can gui m®t message cùng topic nhưng vói trưòng payload bang 0.

**LWT** : Gói tin LWT (last will and testament) không thnc sn biet đưoc Client có trnc tuyen hay không, cái này do gói tin KeepAlive đam nh¾n. Tuy nhiên gói tin LWT như là thông tin đieu gì se xay đen sau khi thiet b% ngoai tuyen.

**Kha năng duy trì ban tin** : MQTT ho tro lưu tru các ban tin trong Broker đe duy trì ban tin. Khi publish các ban tin, các Client có the yêu cau broker duy trì các ban tin. Chi có các ban tin mói nhat đưoc lưu lai. Khi m®t Client đăng ký đen m®t kênh, bat kì ban tin nào đã đưoc đăng ký lưu tru se đưoc gui đen Client.

**Message** : Trong giao thúc MQTT, message còn đưoc gqi là "message payload", có đ%nh dang m¾c đ%nh là plain-text (chu viet ngưòi đqc đưoc), tuy nhiên ngưòi su dnng có the cau hình thành các đ%nh dang khác.

**Topic** : Topic có the coi như m®t "đưòng truyen" logic giua 2 điem là publisher và subscriber. Ve cơ ban, khi message đưoc publish vào m®t topic thì tat ca nhung subscriber cua topic đó se nh¾n đưoc message này. Giao thúc MQTT cho phép khai báo các topic kieu phân cap.

**MQTT Bridge** : MQTT Bridge là m®t tính năng cua MQTT Broker cho phép các MQTT Broker có the ket noi và trao đoi du li¾u vói nhau. Đe su dnng tính năng này, ta can toi thieu 2 Broker, trong đó, m®t Broker bat kỳ se đưoc cau hình thành Bridge. Khi cau hình MQTT bridge, ta can lưu ý tói các thông so sau:

* + - * **address**: đ%a chi cua broker can ket noi.
      * **bridge protocol version**: phiên ban cua giao thúc MQTT đang su dnng chung cho 2 broker.
      * **topic**: phan này đ%nh nghĩa 3 thông so: tên topic đưoc trao đoi giua 2 broker, chieu trao đoi (1 chieu hay 2 chieu) và topic mapping giua 2 broker.

**Bao m¾t** : MQTT đưoc thiet ke m®t cách nhe và linh hoat nhat có the. Do đó nó chi có 1 lóp bao m¾t o tang úng dnng: bao m¾t bang xác thnc (xác thnc các client đưoc quyen truy c¾p tói broker). Tuy v¾y, MQTT vãn có the đưoc cài đ¾t ket hop vói các giai pháp bao m¾t đa tang khác như ket hop vói VPN o tang mang ho¾c SSL/TLS o tang transport.

### Kí hi¾u su dnng trong giao thúc MQTT

Trong MQTT, các kênh đưoc phân cap, giong như m®t h¾ thong t¾p tin.

Các kí hi¾u “+” và “#” đưoc dùng khi đăng ký cho phép Client thay đưoc toàn b® h¾ thong phân cap.

Các kí tn “+” đai di¾n cho tên cua bat kì phân cap nào ngang hàng vói nó.

Các kí tn “#” đai di¾n cho tat ca các cap dưói nó, đieu này có nghĩa nó phai đúng cuoi cùng trong đ%a chi đăng ký.

### Nh¾n xét ve MQTT

MQTT đưoc thiet ke nham phnc vn truyen thông machine-to-machine nhưng thnc te chúng minh nó lai linh hoat hơn mong đoi. Nó hoàn toàn có the áp dnng cho các k%ch ban truyen thông khác như: machine-to-cloud, cloud-to-machine, app-to-app. Chi can có m®t broker phù hop và MQTT client đưoc cài đ¾t đúng cách, các thiet b% xây dnng trên nhieu nen tang khác nhau có the giao tiep vói nhau m®t cách de dàng.

M¾c dù MQTT đưoc thiet ke gqn nhe, nó có hai nhưoc điem làm han che các thiet b%:

* Moi client MQTT phai ho tro TCP và thưòng se giu m®t ket noi mo đen broker o mqi thòi điem. Đoi vói m®t so môi trưòng mà xác suat mat ban tin cao hay không có san máy tính thì đây là ca m®t van đe.
* Tên kênh MQTT thưòng là các chuoi dài làm chúng không thoa đáng vói tiêu chuan 802.15.4

Ca hai thiet sót đưoc giai quyet bang giao thúc MQTT-SN, trong đó xác đ%nh m®t ánh xa UDP cua MQTT và thêm kha năng l¾p chi mnc (indexing) tên các kênh cho broker.

MQTT có m®t so đ¾c điem sau:

* Dang truyen message theo mô hình cung cap/thuê bao (publish/subscribe) cung cap vi¾c

truyen tin phân tán 1-nhieu, tách bi¾t vói phan úng dnng (trong mô hình này, thì ngưòi gui message là publisher, còn ngưòi nh¾n là subscriber).

* Vi¾c truyen message là luôn không quan tâm đen n®i dung đưoc truyen.
* Su dnng TCP/IP là giao thúc nen.
* Có 3 cap đ® cua QoS (Qualities of Service : chat lưong truyen tin) đưoc đưa ra.
* Phan bao bqc du li¾u truyen (overhead) nho (đ® dài co đ%nh luôn là 2 byte), và đưoc giam đen múc toi thieu đe giam tai cho đưòng truyen.
* M®t cơ che đe thông báo đen các thuê bao (subscriber) khi đưòng truyen b% đút bat thưòng, đó là su dnng kha năng thông báo Last Will (thông đi¾p cuoi) và Testament feature.

### M®t so ưu điem cua MQTT so vói các giao thúc IoT khác

MQTT là m®t giao thúc truyen thông nhieu - nhieu đe truyen các thông đi¾p giua nhieu khách hàng thông qua broker trung tâm. Nó tách riêng ngưòi san suat (producer) và ngưòi tiêu thn (consumer) bang cách cho phép các client gui ban tin đi và có các broker quyet đ%nh đưòng đi và sao chép ban tin. MQTT có m®t so ho tro đe duy trì ban tin, nhưng nó làm tot nhat nhi¾m vn là bus truyen du li¾u trnc tiep.

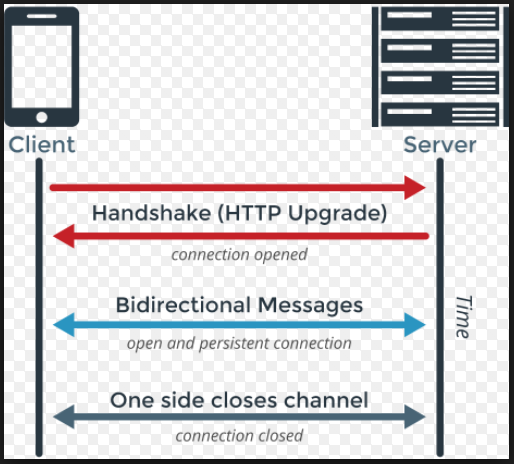
Giao thúc MQTT thì nhe nhàng và nhanh hơn. Nó mat rat ít bytes cho vi¾c ket noi vói server và ket noi có the giu trang thái mo xuyên suot.

MQTT hoat đ®ng tot trong môi trưòng băng thông thap, đ® tre cao, su dnng ít năng lưong.

Các đ%nh dang thông đi¾p cua giao thúc MQTT phong phú, su dnng cho đưoc nhieu mnc đích mien là phía client biet đưoc đ%nh dang ban tin đe giao tiep.

MQTT trong giai đoan gan đây đang noi lên như là m®t giao thúc đưoc su dnng pho bien đóng m®t vai trò quan trqng dùng cho giao tiep IoT, rat nhieu hãng công ngh¾ lón đã su dnng, áp dnng và ho tro giao thúc MQTT cho các úng dnng cua mình như: Facebook, IBM, Intel, Amazon, WSO2, Microsoft,. . .

## WebSocket



Hình 2.3: Mô hình WebSocket

### Tong quan

WebSoket là công ngh¾ ho tro giao tiep hai chieu giua client và server bang cách su dnng m®t TCP socket đe tao m®t ket noi hi¾u qua và ít ton kém. M¾c dù đưoc thiet ke đe chuyên su dnng cho các úng dnng web, l¾p trình viên van có the đưa chúng vào bat kì loai úng dnng nào.

WebSocket mói xuat hi¾n trong HTML5, là m®t ky thu¾t Reverse Ajax. WebSocket cho phép các kênh giao tiep song song hai chieu và hi¾n đã đưoc ho tro trong nhieu trình duy¾t (Firefox, Google Chrome và Safari). Ket noi đưoc mo thông qua m®t HTTP request (yêu cau HTTP), đưoc gqi là liên ket WebSocket vói nhung header đ¾c bi¾t. Ket noi đưoc duy trì đe ban có the viet và nh¾n du li¾u bang JavaScript như khi ban đang su dnng m®t TCP socket đơn thuan.

Du li¾u truyen tai thông qua giao thúc HTTP (thưòng dùng vói kĩ thu¾t Ajax) chúa nhieu du li¾u không can thiet trong phan header. M®t header request/response cua HTTP có kích thưóc khoang 871 byte, trong khi vói WebSocket, kích thưóc này chi là 2 byte (sau khi đã ket noi).

### Ưu điem

WebSocket cung cap kha năng giao tiep hai chieu manh me, có đ® tre thap và de xu lý loi. Không can phai có nhieu ket noi như phương pháp Comet long-polling và cũng không có nhung

nhưoc điem như Comet streaming. API cũng rat de su dnng trnc tiep mà không can bat kỳ các tang bo sung nào, so vói Comet, thưòng đòi hoi m®t thư vi¾n tot đe xu lý ket noi lai, thòi gian chò timeout, các Ajax request (yêu cau Ajax), các tin báo nh¾n và các dang truyen tai tùy chqn khác nhau (Ajax long-polling và jsonp polling).

### Nhưoc điem

Nó là m®t đ¾c ta mói cua HTML5, nên nó van chưa đưoc tat ca các trình duy¾t ho tro.

Không có pham vi yêu cau nào. Do WebSocket là m®t TCP socket chú không phai là HTTP request, nên không de su dnng các d%ch vn có pham vi-yêu cau, như SessionInViewFilter cua Hibernate. Hibernate là m®t framework kinh đien cung cap m®t b® lqc xung quanh m®t HTTP request. Khi bat đau m®t request, nó se thiet l¾p m®t contest (chúa các transaction và liên ket JDBC) đưoc ràng bu®c vói luong request. Khi request đó ket thúc, b® lqc huy bo contest này.

## Giao thNc CoAP

CoAP (Constrained Application Protocol) là giao thúc ràng bu®c úng dnng cua t¾p đoàn CoRE (Constrained Resource Environments) IETF.

### Giói thi¾u ve giao thúc CoAP

Các giao thúc CoAP là can thiet vì các giao thúc truyen thong đưoc coi là "quá n¾ng" cho các úng dnng liên quan đen các thiet b% IoT han che. Mang lưói các nút ket thúc IoT có xu hưóng "nén -lossy", nhưng các thiet b% năng lưong thap mà dna vào hq đưoc kỳ vqng se tiep tnc hoat đ®ng - đưoc trang b% pin ho¾c thu năng lưong - trong nhieu năm và can phai tiêu hao ít năng lưong càng tot.

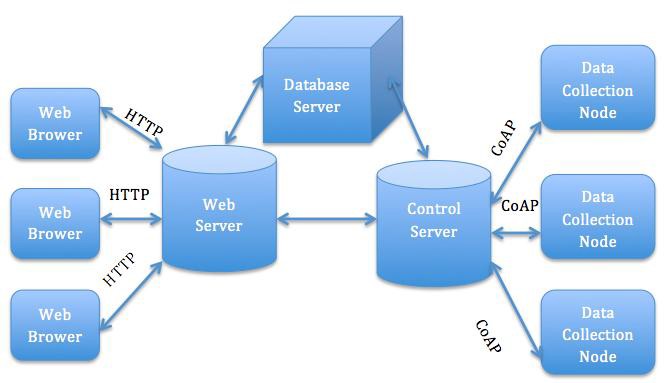
CoAP là m®t giao thúc phan mem cho phép đơn gian ràng bu®c nhieu thú chang han như cam bien năng lưong thap và cơ cau chap hành đe giao tiep tương tác thông qua internet. Nó chay trên các thiet b% có ho tro các UDP, và thnc hi¾n m®t "nhe" lóp úng dnng có tính năng kích thưóc nho nhan, quan lý tin nhan và trên không nhan nhe lý tưong cho các thiet b% đi¾n năng thap, b® nhó thap.

### Môi trưòng hoat đ®ng CoAP

CoAP là m®t giao thúc su dnng chu yeu như là m®t giao thúc m®t - m®t cho chuyen trang thái thông tin giua máy khách và máy chu. Trong khi nó có ho tro cho quan sát tài nguyên, CoAP là phù hop nhat vói m®t mô hình chuyen trang thái, không hoàn toàn là dna trên sn ki¾n.

Thông tin thiet b%, thiet b% đieu khien và thiet b% thông tin liên lac trong mang gia đình thông minh có các ký tn cua chi phí thap và trqng lưong nhe. Như v¾y, CoAP có the đưoc xem là sn lna chqn giao thúc tot nhat cho các mang truyen thông gia đình thông minh.

Mang nhà thông minh cung cap kiem soát và giám sát năng lưong cua các thiet b% trong nhà. H¾ thong kiem soát năng lưong su dnng quan lý o cam thông minh và thiet b% su dnng màn hình đi¾n đe cung cap đi¾n áp, thông tin năng lưong hi¾n tai và khác. Nó có the nh¾n ra canh báo tai nan, đieu khien tù xa và tiet ki¾m năng lưong năng đ®ng. Cau trúc h¾ thong đưoc the hi¾n trong Hình 2-3. Moi b® sưu t¾p nút du li¾u vói CoAP khách hàng có the trao đoi thông tin vói các nút khác. CoAP có the đưoc cài đ¾t đưoc trong mang LAN ho¾c Internet. Không giong như nhieu giao thúc không dây cho các thiet b%, CoAP đã thiet ke không b% han che trong m®t mang n®i b® nhưng cung cap nen tang cơ ban cua trang web. Trong h¾ thong này, các proxy CoAP - HTTP đưoc su dnng đe cung cap cho khách hàng ket noi HTTP đen tài nguyên CoAP và ngưoc lai.



Hình 2.4: H¾ thong kiem soát

## Giao thNc TCP

### Tong quan TCP/IP

TCP/IP là m®t hq các giao thúc đưoc gqi là hq giao thúc IP, bao gom bon tang. Can nhó rang TCP/IP không phai là m®t giao thúc mà thnc sn là m®t hq các giao thúc, và bao gom các giao thúc múc thap khác như IP, TCP, và UDP. UDP nam o tang giao v¾n, phía trên giao thúc IP. Tang giao v¾n cung cap kha năng truyen tin giua các mang thông qua các gateway. Nó su dnng các đ%a chi IP đe gui các gói tin trên Internet ho¾c trên mang thông qua các trình đieu khien thiet b% khác nhau. TCP và UDP là m®t phan cua hq giao thúc TCP/IP; moi giao thúc có nhung ưu và nhưoc điem riêng cua nó.

Giao thúc UDP là giao thúc đơn gian, phi liên ket và cung cap d%ch vn trên tang giao v¾n vói toc đ® nhanh. Nó ho tro liên ket m®t - nhieu và thưòng đưoc su dnng thưòng xuyên trong liên ket m®t - nhieu bang cách su dnng các datagram multicast và unicast.

Giao thúc IP là giao thúc cơ ban cua Internet. TCP và UDP đeu là hai giao thúc tang giao thúc v¾n trên cơ so cua giao thúc IP.

### TCP

TCP là viet tat cua Transmission Control Protocol. Đó là giao thúc pho bien nhat đưoc su dnng trên Internet.

Khi ban tai m®t trang web, máy tính cua ban se gui các gói tin TCP đen đ%a chi máy chu web, yêu cau nó gui các trang web cho ban. Các máy chu web phan úng bang cách gui m®t dòng cua các gói tin TCP, mà trình duy¾t web cua ban se ket noi vói nhau đe tao nên trang web và hien th% nó cho ban. Khi ban bam vào m®t liên ket, đăng nh¾p, bình lu¾n, ho¾c làm bat cú đieu gì khác, trình duy¾t web cua ban gui các gói tin TCP đen máy chu và máy chu se gui các gói tin TCP lai. TCP không phai là thông tin liên lac mà chi là m®t cách - h¾ thong tù xa gui gói tin tro lai đe thùa nh¾n nó nh¾n đưoc các gói du li¾u cua ban.

TCP đam bao ngưòi nh¾n se nh¾n đưoc các gói tin theo thú tn chúng đánh so. Bên nh¾n se gui tin nhan lai cho bên gui đe xác nh¾n nó đã nh¾n đưoc tin nhan. Neu bên gui không nh¾n đưoc m®t phan úng chính xác, nó se gui lai gói tin đe đam bao bên nh¾n nh¾n đưoc. Gói tin cũng đưoc kiem tra loi. Giao thúc TCP đưoc xem như là tat ca ve đ® tin c¾y này - các gói tin gui đi vói TCP đưoc theo dõi sát sao, vì v¾y không có du li¾u b% mat ho¾c b% hong trên đưòng v¾n chuyen. Đây là lý do tai sao t¾p tin tai ve không b% loi ngay ca khi có trnc tr¾c mang. Tat nhiên,

neu ngưòi nh¾n là hoàn toàn an, máy tính cua ban se tù bo và ban se thay m®t thông báo loi nói rang nó không the giao tiep vói máy chu tù xa.

TCP đòi hoi thiet l¾p ket noi trưóc khi bat đau gui du li¾u và ket thúc ket noi khi vi¾c gui du li¾u hoàn tat. Cn the, các ket noi TCP có ba pha:

#### Thiet l¾p ket noi

Có 3 bưóc bat tay:

* Client yêu cau mo cong d%ch vn bang cách gui gói tin SYN (gói tin TCP) tói server, trong gói tin này, tham so sequence number đưoc gán cho m®t giá tr% ngau nhiên X.
* Server hoi đáp bang cách gui lai phía client ban tin SYN-ACK, trong gói tin này, tham so acknowledgment number đưoc gán giá tr% bang X + 1, tham so sequence number đưoc gán ngau nhiên m®t giá tr% Y.
* Đe hoàn tat quá trình bat tay ba bưóc, client tiep tnc gui tói server ban tin ACK, trong ban tin này, tham so sequence number đưoc gán cho giá tr% bang X + 1 còn tham so acknowledgment number đưoc gán giá tr% bang Y + 1.

#### Truyen dÑ li¾u

Đ¾c điem cơ ban cua truyen du li¾u TCP:

* Truyen du li¾u không loi (do có cơ che sua loi/truyen lai).
* Truyen các gói du li¾u theo đúng thú tn.
* Truyen lai các gói du li¾u mat trên đưòng truyen.
* Loai bo các gói du li¾u trùng l¾p.
* Cơ che han che tac nghen đưòng truyen.

#### Ket thúc ket noi

Đe ket thúc ket noi hai bên su dnng quá trình bat tay 4 bưóc và chieu cua ket noi ket thúc đ®c l¾p vói nhau. Khi m®t bên muon ket thúc, nó gui đi m®t gói tin FIN và bên kia gui lai tin báo nh¾n ACK. Vì v¾y, m®t quá trình ket thúc tiêu bieu se có 2 c¾p gói tin trao đoi.

M®t ket noi có the ton tai o dang "nua mo": m®t bên đã ket thúc gui du li¾u nên chi nh¾n thông tin, bên kia van tiep tnc gui.

## Giao thNc UDP

UDP là viet tat cua User Datagram Protocol - m®t gói tương tn như m®t gói cua thông tin. Giao thúc UDP hoat đ®ng tương tn như TCP, nhưng nó bao gom tat ca nhung thú đã kiem tra và có loi.

Khi su dnng UDP, gói chi gui đen bên nh¾n. Bên gui se không chò đoi đe chac chan rang bên nh¾n đã nh¾n đưoc các gói tin - nó se tiep tnc gui các gói tiep theo. Neu ban là ngưòi nh¾n và ban bo lõ m®t so gói tin UDP vì quá xau - ban không the yêu cau nhung gói tin m®t lan nua. Không có gì đe đam bao ban đang nh¾n đưoc tat ca các gói và không có cách nào đe yêu cau m®t gói m®t lan nua neu ban bo lõ nó, nhưng bù vào đó, các máy tính có the giao tiep m®t cách nhanh chóng hơn.

UDP đưoc su dnng khi toc đ® là mong muon và sua loi là không can thiet. Ví dn, UDP thưòng đưoc su dnng cho chương trình phát sóng trnc tiep và trò chơi trnc tuyen.

Ví dn: Live Stream phát sóng thưòng su dnng UDP thay vì TCP. Các máy chu chi can gui m®t dòng cua các gói tin UDP đe máy tính xem. Neu ban b% mat ket noi trong vài giây, video se đóng băng cho m®t thòi điem và sau đó chuyen đen các bit hi¾n tai cua truyen hình, bo qua các bit ban đã b% bo qua. Video ho¾c âm thanh có the b% bóp méo m®t lúc và video tiep tnc chơi mà không có du li¾u b% mat.

Đieu này hoat đ®ng tương tn như trong các trò chơi trnc tuyen - neu ban bo lõ m®t so gói tin UDP, nhân v¾t ngưòi chơi có the xuat hi¾n d%ch chuyen trên ban đo khi ban nh¾n đưoc các gói tin UDP mói hơn. Không có điem trong vi¾c yêu cau các gói cũ neu ban b% mat hq, và các trò chơi đưoc tiep tnc mà không có ban. Tat ca nhung van đe là nhung gì đang xay ra ngay bây giò trên máy chu trò chơi - không phai nhung gì đã xay ra m®t vài giây.

### M®t so thu¾t ngu UDP

**Packet** Trong truyen so li¾u, m®t packet là m®t dãy các so nh% phân, bieu dien du li¾u và các tín hi¾u đieu khien, các gói tin này đưoc chuyen đi và chuyen tói host. Trong gói tin, thông tin đưoc sap xep theo m®t khuôn dang cn the.

**Datagram** M®t datagram là m®t gói tin đ®c l¾p, tn chúa, mang đay đu du li¾u đe đ%nh tuyen tù nguon tói đích mà không can thông tin thêm.

**MTU** MTU là viet tat cua Maximum Transmission Unit. MTU là m®t đ¾c trưng cua tang liên ket mô ta so byte du li¾u toi đa có the truyen trong m®t gói tin. M¾t khác, MTU là gói du

li¾u lón nhat mà môi trưòng mang cho trưóc có the truyen. Ví dn, Ethernet có MTU co đ%nh là 1500 byte. Trong UDP, neu kích thưóc cua m®t datagram lón hơn MTU, IP se thnc hi¾n phân đoan, chia datagram thành các phan nho hơn (các đoan), vì v¾y moi đoan nho có kích thưóc nho hơn MTU.

**Port** UDP su dnng các cong đe ánh xa du li¾u đen vào m®t tien trình cn the đang chay trên m®t máy tính. UDP đ%nh đưòng đi cho packet tai v% trí xác đ%nh bang cách su dnng so hi¾u cong đưoc xác đ%nh trong header cua datagram. Các cong đưoc bieu dien boi các so 16-bit, vì the các cong nam trong dai tù 0 đen 65535. Các cong cũng đưoc xem như là các điem cuoi cua các liên ket logic, và đưoc chia thành ba loai sau:

* Các cong pho bien: Tù 0 đen 1023.
* Các cong đã đăng ký: 1024 đen 49151.
* Các cong đ®ng/dành riêng 49152 đen 65535.

Các cong UDP có the nh¾n nhieu hơn m®t thông đi¾p o m®t thòi điem. Trong m®t so trưòng hop, các d%ch vn TCP và UDP có the su dnng cùng m®t so hi¾u cong, như 7 (Echo) ho¾c trên cong 23 (Telnet).

**TTL** Giá tr% TTL cho phép chúng ta thiet l¾p m®t giói han trên cua các router mà m®t datagram có the đi qua. Giá tr% TTL ngăn ngùa các gói tin khoi b% ket trong các vòng l¾p đ%nh tuyen vô han. TTL đưoc khoi tao boi phía gui và giá tr% đưoc giam đi boi moi router quan lý datagram. Khi TTL bang 0, datagram b% loai bo.

**Multicasting** Multicasting là phương pháp dna trên chuan có tính chat mo đe phân phoi các thông tin giong nhau đen nhieu ngưòi dùng. Multicasting là m®t đ¾c trưng chính cua giao thúc UDP. Multicasting cho phép chúng ta truyen tin theo kieu m®t nhieu, ví dn gui tin ho¾c thư đi¾n tu tói nhieu ngưòi nh¾n, đài phát thanh trên Internet, ho¾c các chương trình demo trnc tuyen.

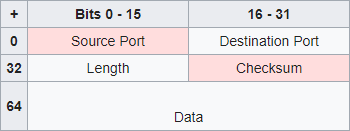
### Cau trúc gói

UDP là giao thúc hưóng thông đi¾p nho nhat cua tang giao v¾n hi¾n đưoc mô ta trong RFC 768 cua IETF.

Trong b® giao thúc TCP/IP, UDP cung cap m®t giao di¾n rat đơn gian giua tang mang bên dưói, và tang phiên làm vi¾c ho¾c tang úng dnng phía trên.

UDP không đam bao cho các tang phía trên thông đi¾p đã đưoc gui đi và ngưòi gui cũng không có trang thái thông đi¾p UDP m®t khi đã đưoc gui (Vì lý do này đôi khi UDP còn đưoc gqi là Unreliable Datagram Protocol).

UDP chi thêm các thông tin multiplexing và giao d%ch. Các loai thông tin tin c¾y cho vi¾c truyen du li¾u neu can phai đưoc xây dnng o các tang cao hơn.



Hình 2.5: Cau trúc gói UDP

Phan header cua UDP chi chúa 4 trưòng du li¾u, trong đó có 2 trưòng là tùy chqn.

**Source port** trưòng này xác đ%nh cong cua ngưòi gui thông tin và có ý nghĩa neu muon nh¾n thông tin phan hoi tù ngưòi nh¾n. Neu không dùng đen thì đ¾t nó bang 0.

**Destination port** trưòng xác đ%nh cong nh¾n thông tin, và trưòng này là can thiet.

**Length** trưòng có đ® dài 16 bit xác đ%nh chieu dài cua toàn b® datagram: phan header và du li¾u. Chieu dài toi thieu là 8 byte khi gói tin không có du li¾u, chi có header.

**Checksum** trưòng checksum 16 bit dùng cho vi¾c kiem tra loi cua phan header và du li¾u.

Phương pháp tính checksum đưoc đ%nh nghĩa trong RFC 768.

Do thieu tính tin c¾y, các úng dnng UDP nói chung phai chap nh¾n mat mát, loi ho¾c trùng du li¾u. M®t so úng dnng như TFTP có nhu cau phai thêm nhung ky thu¾t làm tin c¾y cơ ban vào tang úng dnng. Hau het các úng dnng UDP không can nhung ky thu¾t làm tin c¾y này và đôi khi nó b% bo đi. Streaming media, game trnc tuyen và voice over IP (VoIP) là nhung thí dn cho các úng dnng thưòng dùng UDP. Neu m®t úng dnng đòi hoi múc đ® cao hơn ve tính tin c¾y, nhung giao thúc như TCP ho¾c mã erasure có the dùng thay.

M¾c dù tong lưong lưu thông cua UDP trên mang thưòng chi vài phan trăm, nhưng có nhieu úng dnng quan trqng dùng UDP, bao gom DNS, SNMP, DHCP và RIP.

### Phân bi¾t UDP và TCP

#### NhÑng điem chung

Ca hai giao thúc TCP và UDP đeu là giao thúc đưoc su dnng đe gui các bit du li¾u - đưoc gqi là các gói tin - trên Internet. Ca hai đeu xây dnng trên đau trang cua giao thúc Internet. Nói cách khác, cho dù ban đang gui m®t gói du li¾u qua giao thúc TCP ho¾c UDP, gói đó đưoc gui đen m®t đ%a chi IP. Các gói du li¾u đưoc xu lý tương tn, chúng chuyen tù máy tính tói thiet b% đ%nh tuyen trung gian và đen đích.

TCP và UDP không phai là giao thúc duy nhat làm vi¾c trên IP. Tuy nhiên, chúng là nhung giao thúc đưoc su dnng r®ng rãi nhat. Thu¾t ngu su dnng r®ng rãi "TCP/IP" có nghĩa là TCP qua IP. UDP qua IP đưoc gqi là "UDP/IP” và đây không phai là m®t thu¾t ngu pho bien.

#### NhÑng điem khác nhau

|  |  |  |
| --- | --- | --- |
| **Các đ¾c trưng** | **UDP** | **TCP** |
| Hưóng liên ket | Không | Có |
| Su dnng phiên | Không | Có |
| Đ® tin c¾y | Không | Có |
| Xác thnc | Không | Có |
| Đánh thú tn | Không | Có |
| Đieu khien luong | Không | Có |
| Bao m¾t | Ít | Nhieu hơn |

Bang 2.1: Sn khác bi¾t giua UDP và TCP

* 1. **Trang web** [***www.cloudmqtt.com***](http://www.cloudmqtt.com/)

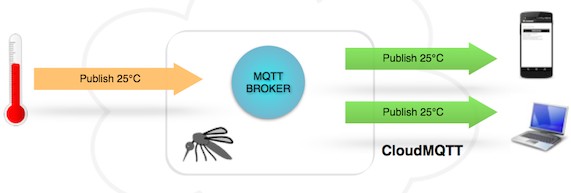
### CloudMQTT là gì

CloudMQTT đưoc quan lý các máy chu Mosquitto trong đám mây cung cap các phương pháp nhe đe thnc hi¾n vi¾c gui tin nhan bang cách publish (subscribe) tin nhan noi tiep nhau.

Các tin nhan noi tiep nhau cung cap m®t giao thúc không đong b®, ngưòi gui và nh¾n tin nhan không can thiet tương tác vói tin nhan cùng 1 lúc. Tin nhan đưoc lưu vào hàng đoi cho đen khi có ngưòi nh¾n lay ra ho¾c cho đen khi tin nhan het han. MQTT và Mosquitto đưoc su dnng tot vói các úng dnng nhay cam vói băng thông.

Trang có ho tro WebSocket vói cong cho WebSocket đưoc hien th% chi tiet trong trang quan lý quyen truy c¾p cua cá nhân.

Ket noi su dnng cho trang là ket noi TCP hoat đ®ng tù úng dnng (thiet b%) cá nhân tói Mosquitto broker. Neu 2 thiet b% ket noi tói broker, thì se tiêu thn 2 ket noi cua tài khoan.



Hình 2.6: CloudMQTT

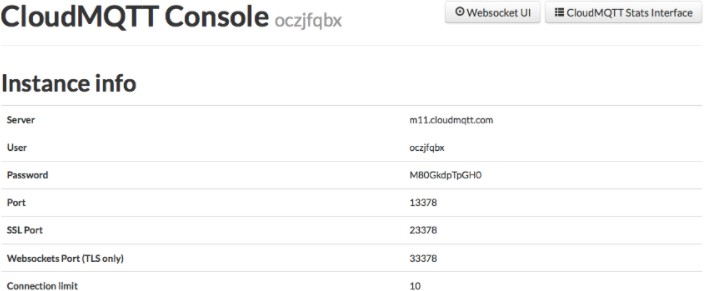
* + 1. Tong quan ve trang web [*www.cloudmqtt.com*](http://www.cloudmqtt.com/)

CloudMQTT cung cap m®t d%ch vn broker trai nghi¾m thu mien phí cho phép su dnng MQTT vói 10 ket noi vói toc đ® 10Kbit/s. Có the mo r®ng cong ket noi và toc đ® ket noi tăng lên theo các gói d%ch vn trang web cung cap.

Ho tro nhieu ngôn ngu: Heroku, Ruby, Python, Node.js, Java, Go, .NET, NodeMCU, PHP,...

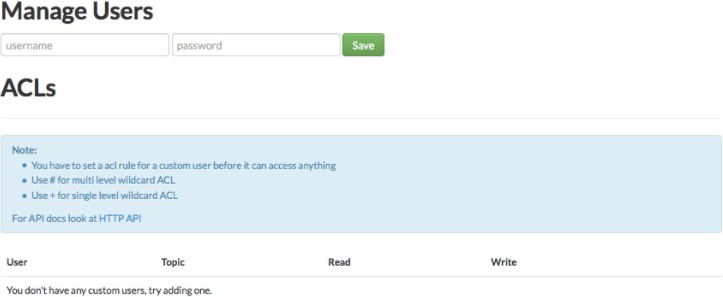
Sau khi đăng nh¾p, có the xem đưoc thông tin chi tiet ve thông tin ket noi tai trang chi tiet.

Có the truy c¾p quyen quan lý tù trang đó.



Hình 2.7: Thông tin ket noi tai CloudMQTT

Có the quan lý quyen truy c¾p và các lu¾t ACL cua host MQTT cua mình đưoc lưu tru thông qua HTTP API cua trang.



Hình 2.8: Trang quan lý truy nh¾p

## Tìm hieu phan cNng

### Module wifi ESP8266



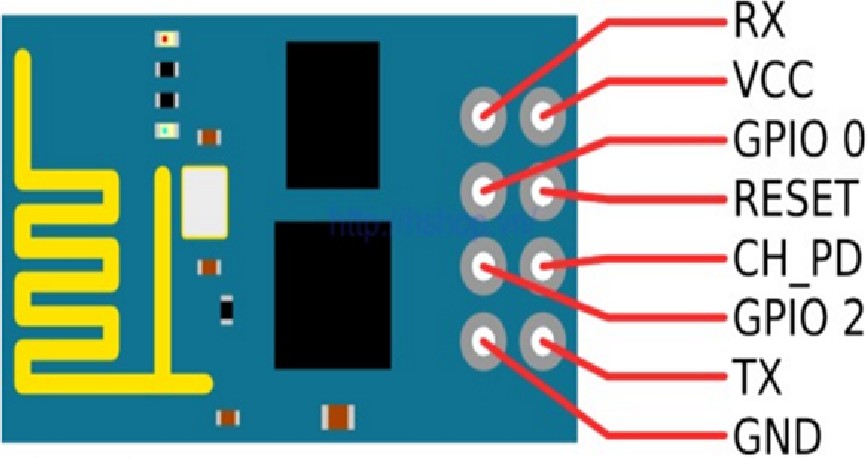
Hình 2.9: Ânh thnc te cua ESP8266

ESP8266 là module wifi giá re và đưoc đánh giá rat cao cho các úng dnng liên quan đen Internet và Wifi cũng như các úng dnng truyen nh¾n su dnng thay the cho các module RF khác. ESP8266 là m®t chip tích hop cao, đưoc thiet ke cho nhu cau cua m®t the giói ket noi mói, the giói Internet of thing (IoT). Nó cung cap m®t giai pháp ket noi mang Wi-Fi đay đu và khép kín, cho phép nó có the lưu tru các úng dnng ho¾c đe giam tai tat ca các chúc năng ket noi mang

Wi-Fi tù m®t b® xu lý úng dnng.

ESP8266 có xu lý và kha năng lưu tru manh me cho phép nó đưoc tích hop vói các b® cam bien, vi đieu khien và các thiet b% úng dnng cn the khác thông qua GPIOs vói m®t chi phí toi thieu và m®t PCB toi thieu. Su dnng giao thúc noi tiep vói toc đ® Baud là 9600(m¾c đ%nh). Toc đ® baud có the lên đen 115200.

ESP8266 là module su dnng các t¾p l¾nh AT. Tuy nhiên đã có thêm nhieu phiên ban firmware ho tro nhieu ngôn ngu, trong đó có LUA và Arduino. Chính vì the có the dùng bat kì board Arduino nào đe đieu khien thu phát thông qua phương thúc truyen noi tiep.



Hình 2.10: Sơ đo ra chân cua ESP8266

Chúc năng các chân cơ ban:

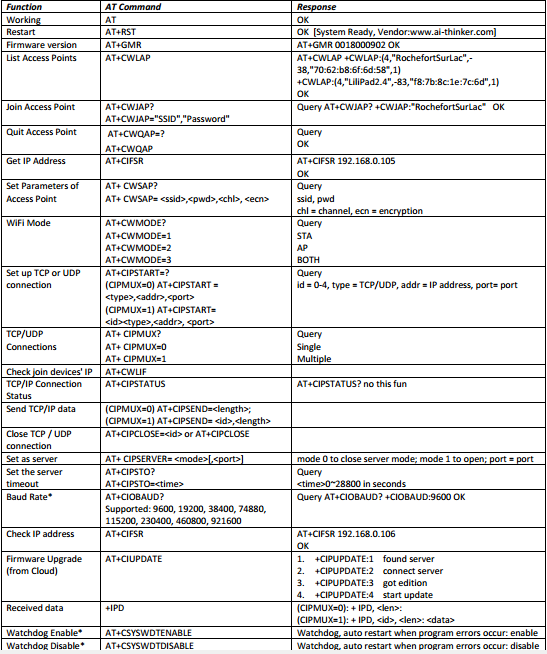
- Vcc: 3.3V.

* GND: 0V.
* TX: chân TX cua giao thúc UART, ket noi đen chân RX cua vi đieu khien.
* RX: chân RX cua giao thúc UART, ket noi đen chân TX cua vi đieu khien.
* RST: chân reset kéo xuong mass đe b¾t reset.
* CH-PD: chân này kéo lên múc cao module bat đau thu phát, kéo xuong thap module dùng. Vì ESP8266 khoi đ®ng hút dòng lón nên phai đe múc thap khi khoi đ®ng xong se kéo lên múc cao.
* GPIO0: kéo xuong thap cho che đ® upgrade firmwave.

#### ChNc năng:

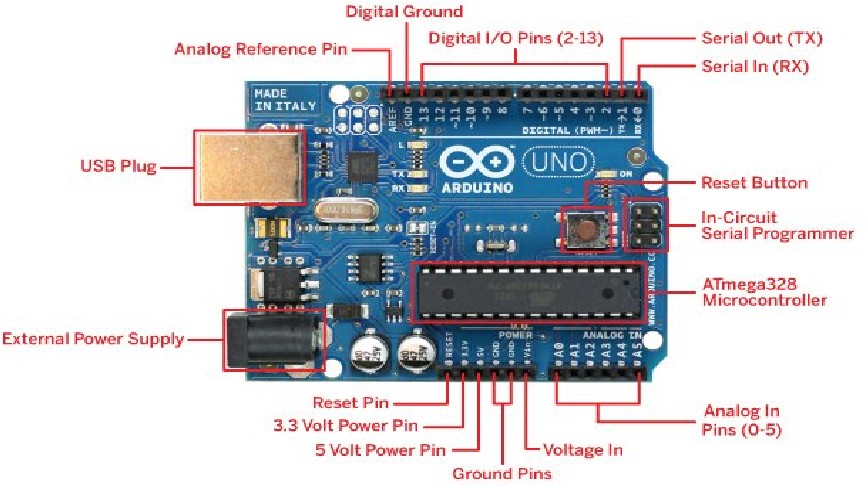
* Ho tro chuan 802.11 b/g/n.
* Wi-Fi 2.4 GHz, ho tro WPA/WPA2.
* Chuan đi¾n áp hoat đ®ng: 3.3V.
* Chuan giao tiep noi tiep UART vói toc đ® Baud lên đen 115200.
* Có 3 che đ® hoat đ®ng: Client, Access Point, Both Client and Access Point.
* Ho tro các chuan bao m¾t như: OPEN, WEP, WPA-PSK, WPA2-PSK, WPA-WPA2-PSK.
* Ho tro ca 2 giao tiep TCP và UDP.
* Làm vi¾c như các máy chu có the ket noi vói 5 máy con.

Dưói đây là các t¾p l¾nh AT cơ ban:



Bang 2.2: Các t¾p l¾nh AT

### Module Arduino Uno R3



Hình 2.11: Sơ đo ra chân cua Arduino Uno R3

|  |  |
| --- | --- |
| Vi đieu khien | ATmega328 hq 8bit |
| Đi¾n áp hoat đ®ng | 5V DC (chi đưoc cap qua cong USB) |
| Tan so hoat đ®ng | 16 MHz |
| Dòng tiêu thn | khoang 30mA |
| Đi¾n áp vào | khuyên dùng 7-12V DC |
| Đi¾n áp vào | giói han 6-20V DC |
| So chân Digital I/O | 14 (6 chân hardware PWM) |
| So chân Analog | 6 (đ® phân giai 10bit) |
| Dòng toi đa trên moi chân I/O | 30 mA |
| Dòng ra toi đa (5V) | 500 mA |
| Dòng ra toi đa (3.3V) | 50 mA |
| B® nhó flash | 32 KB (ATmega328) vói 0.5KB dùng boi bootloader |
| SRAM | 2 KB (ATmega328) |
| EEPROM | 1 KB (ATmega328) |

Bang 2.3: Thông so cơ ban cua Arduino

Arduino UNO có the đưoc cap nguon 5V thông qua cong USB ho¾c cap nguon ngoài vói đi¾n áp khuyên dùng là 7-12V DC và giói han là 6-20V. Thưòng thì cap nguon bang pin vuông 9V là hop lí nhat neu ban không có san nguon tù cong USB. Neu cap nguon vưot quá ngưõng giói han trên, ban se làm hong Arduino UNO.

M®t so chân digital có các chúc năng đ¾c bi¾t như sau:

* 2 chân Serial: 0 (RX) và 1 (TX): dùng đe gui (transmit – TX) và nh¾n (receive – RX) du li¾u TTL Serial. Arduino Uno có the giao tiep vói thiet b% khác thông qua 2 chân này. Ket noi bluetooth thưòng thay chính là ket noi Serial không dây. Neu không can giao tiep Serial, ban không nên su dnng 2 chân này neu không can thiet.
* Chân PWM ( ): 3, 5, 6, 9, 10, và 11: cho phép ban xuat ra xung PWM vói đ® phân giai 8bit (giá tr% tù 0 → 28-1 tương úng vói 0V → 5V) bang hàm analogWrite(). Nói m®t cách đơn gian, ban có the đieu chinh đưoc đi¾n áp ra o chân này tù múc 0V đen 5V thay vì chi co đ%nh o múc 0V và 5V như nhung chân khác.
* Chân giao tiep SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ngoài các chúc năng thông thưòng, 4 chân này còn dùng đe truyen phát du li¾u bang giao thúc SPI vói các thiet b% khác.
* LED 13: trên Arduino UNO có 1 đèn led màu cam (kí hi¾u chu L). Khi bam nút Reset, ban se thay đèn này nhap nháy đe báo hi¾u. Nó đưoc noi vói chân so 13. Khi chân này đưoc ngưòi dùng su dnng, LED se sáng.

Arduino UNO có 6 chân analog (A0 → A5) cung cap đ® phân giai tín hi¾u 10bit (0 → 210-1) đe đqc giá tr% đi¾n áp trong khoang 0V → 5V. Vói chân AREF trên board, ban có the đe đưa vào đi¾n áp tham chieu khi su dnng các chân analog. Túc là neu ban cap đi¾n áp 2.5V vào chân này thì ban có the dùng các chân analog đe đo đi¾n áp trong khoang tù 0V → 2.5V vói đ® phân giai van là 10bit.

Đ¾c bi¾t, Arduino UNO có 2 chân A4 (SDA) và A5 (SCL) ho tro giao tiep I2C/TWI vói các thiet b% khác.

Trên board còn có 1 nút reset, 1 ngõ ket noi vói máy tính qua cong USB và 1 ngõ cap nguon su dnng jack 2.1mm lay năng lưong trnc tiep tù AC-DC adapter hay thông qua ac-quy nguon.

Khi làm vi¾c vói Arduino board, m®t so thu¾t ngu sau can đưoc lưu ý: Flash Memory: b® nhó có the ghi đưoc, du li¾u không b% mat ngay ca khi tat đi¾n. Ve vai trò, ta có the hình dung b® nhó này như o cúng đe chúa du li¾u trên board. Chương trình đưoc viet cho Arduino se đưoc lưu o đây. Kích thưóc cua vùng nhó này thông thưòng dna vào vi đieu khien đưoc su dnng, ví

dn như ATmega8 có 8KB flash memory. Loai b® nhó này có the ch%u đưoc khoang 10,000 lan ghi / xoá.

RAM: tương tn như RAM cua máy tính, se b% mat du li¾u khi ngat đi¾n nhưng bù lai toc đ® đqc ghi xoá rat nhanh. Kích thưóc nho hơn Flash Memory nhieu lan.

EEPROM: m®t dang b® nhó tương tn như Flash Memory nhưng có chu kì ghi / xoá cao hơn

- khoang 100,000 lan và có kích thưóc rat nho. Đe đqc / ghi du li¾u ta có the dùng thư vi¾n EEPROM cua Arduino.

#### M®t so lưu ý khi sN dnng:

Arduino UNO không có bao v¾ cam ngưoc nguon vào. Do đó ta phai het súc can th¾n, kiem tra các cnc âm – dương cua nguon trưóc khi cap cho Arduino UNO. Ta nên dùng nguon tù cong USB neu có the.

Các chân 3.3V và 5V trên Arduino là các chân dùng đe cap nguon ra cho các thiet b% khác, không phai là các chân cap nguon vào. Vi¾c cap nguon sai v% trí có the làm hong board. Đieu này không đưoc nhà san xuat khuyen khích.

Cap nguon ngoài không qua cong USB cho Arduino UNO vói đi¾n áp dưói 6V có the làm hong board.

Cap đi¾n áp trên 13V vào chân RESET trên board có the làm hong vi đieu khien AT- mega328.

Cưòng đ® dòng đi¾n vào/ra o tat ca các chân Digital và Analog cua Arduino UNO neu vưot quá 200mA se làm hong vi đieu khien.

Cap đi¾p áp trên 5.5V vào các chân Digital ho¾c Analog cua Arduino UNO se làm hong vi đieu khien.

Cưòng đ® dòng đi¾n qua m®t chân Digital ho¾c Analog bat kì cua Arduino UNO vưot quá 40mA se làm hong vi đieu khien. Do đó neu không dùng đe truyen nh¾n du li¾u, ta phai mac m®t đi¾n tro han dòng.

**B® nhé:** Vi đieu khien Atmega328 tiêu chuan cung cap cho ngưòi dùng:

32KB b® nhó Flash: nhung đoan l¾nh ta l¾p trình se đưoc lưu tru trong b® nhó Flash cua vi đieu khien. Thưòng thì se có khoang vài KB trong so này se đưoc dùng cho bootloader nhưng đùng lo, ta hiem khi nào can quá 20KB b® nhó này đâu.

2KB cho SRAM (Static Random Access Memory): giá tr% các bien ta khai báo khi l¾p trình se lưu o đây. Ta khai báo càng nhieu bien thì càng can nhieu b® nhó RAM. Tuy v¾y, thnc sn thì cũng hiem khi nào b® nhó RAM lai tro thành thú mà ta phai b¾n tâm. Khi mat đi¾n, du li¾u trên

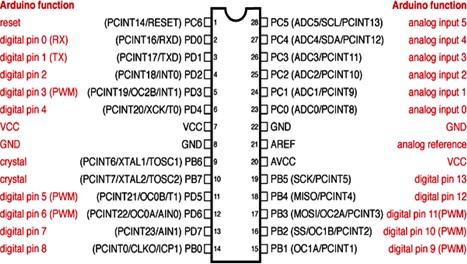
SRAM se b% mat.

1KB EEPROM (Electrically Eraseble Programmable Read Only Memory): đây giong như m®t chiec o cúng mini – nơi ta có the đqc và ghi du li¾u cua mình vào đây mà không phai lo b% mat khi cúp đi¾n giong như du li¾u trên SRAM.

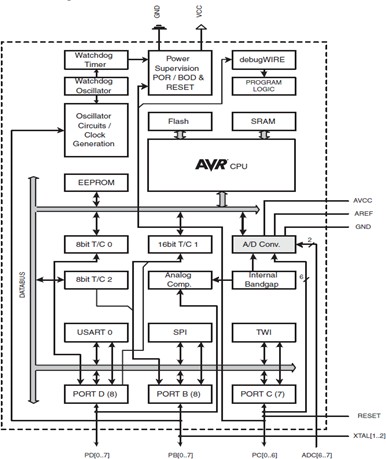
**Ngôn ngÑ l¾p trình trên Arduino:** Các thiet b% dna trên nen tang Arduino đưoc l¾p trình bang ngôn ngu riêng o đây là môi trưòng Arduino IDE. Ngôn ngu này dna trên ngôn ngu Wiring đưoc viet cho phan cúng nói chung. Và Wiring lai là m®t bien the cua C/C++.

Chương trình Arduino có the đưoc chia làm 3 phan: **cau trúc** (structure), **bien so** (variable) và **hang so** (constant), **hàm và thu tnc** (function).

### Vi đieu khien ATmega328

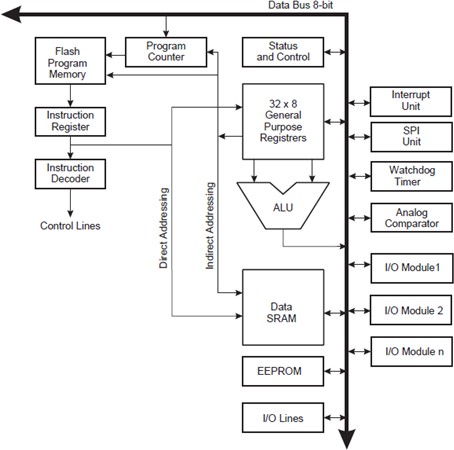
ATmega328 thu®c dòng vi đieu khien AVR, là m®t vi đieu khien 8-bit đưoc che tao bang công ngh¾ CMOS tiêu thn năng lưong thap, dna trên vi¾c cai tien cau trúc RISC. Bang cách tiêu ton ít năng lưong hơn trong 1 chu kỳ xung clock, ATmega328 đat đưoc thông lưong xap xi 1 MIPS trên 1MHz cho phép các nhà thiet ke h¾ thong toi ưu hóa đi¾n năng tiêu thn trong khi toc đ® xu lí.

Hình 2.12: Vi đieu khien ATmega328



Hình 2.13: Sơ đo khoi ATmega328 (Block Diagram)

Nhân AVR tích hop m®t cau trúc phong phú vói 32 thanh ghi đa dnng, noi trnc tiep vói Artithmectic Logic Unit (ALU). Vói cau trúc trên thì hi¾u suat mã hóa cao hơn trong khi thông lưong nhanh hơn gap 10 lan so vói cau trúc cua vi đieu khien con-ventional CISC.



Hình 2.14: Sơ đo cau trúc AVR CPU

Kien trúc lõi AVR tong quát. Các chúc năng chính cua lõi CPU là đe đam bao đúng chương trình thnc hi¾n. Do đó CPU phai có kha năng truy c¾p vào ký úc, thnc hi¾n tính toán, kiem soát thiet b% ngoai vi và xu lý ngat.

### Màn hình LCD 20x4



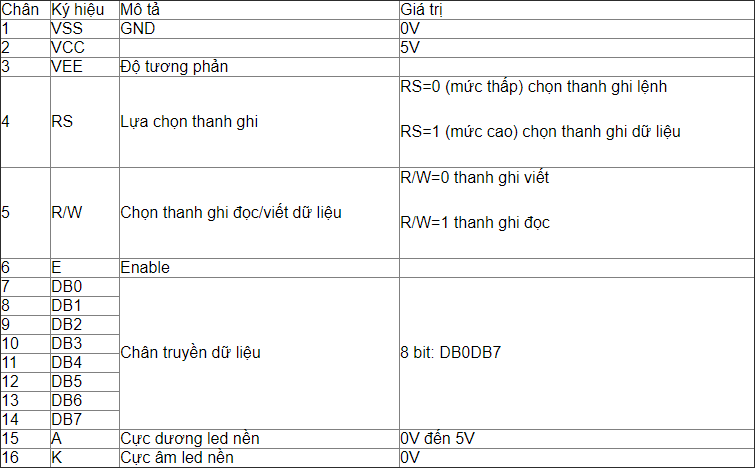
Hình 2.15: Màn hình LCD 20x4

Màn hình text LCD20x4 xanh dương su dnng driver HD44780, có kha năng hien th% 4 dòng vói moi dòng 20 ký tn, màn hình có đ® ben cao, rat pho bien, nhieu code mau có the su dnng, de su dnng phnc vn cho nhu cau hqc t¾p nghiên cúu.

#### Thông so kĩ thu¾t:

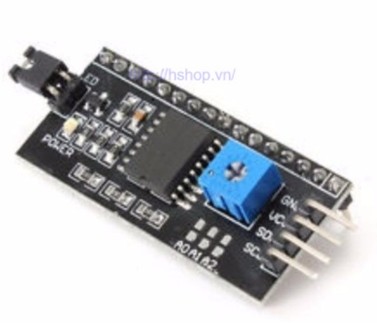
* Đi¾n áp hoat đ®ng: 5V.
* Kích thưóc: 98 x 60 x 13.5 mm.
* Chu trang, nen xanh dương.
* Khoang cách giua 2 chân ket noi là 0.1 inch.
* Đèn nen có the đieu chinh đ® sáng.
* Có che đ® 4bit ho¾c 8bit tùy nhu cau su dnng.
* B® ký tn có the hien th% tieng anh ho tro driver HD44780.

#### Bang cơ che hoat đ®ng cua các chân trên LCD20x4:



Bang 2.4: Hoat đ®ng các chân cua LCD

### Mach chuyen giao tiep LCD sang I2C



Hình 2.16: Mach chuyen giao tiep LCD sang I2C

Đe su dnng LCD20x4 can có ít nhat 6 chân cua MCU ket noi vói các chân RS, EN, D7, D6, D5, D4 đe có the giao tiep vói LCD. Nhưng vói mach chuyen giao tiep LCD sang I2C, chi can 2 chân (SDA và SCL) cua MCU ket noi vói 2 chân (SDA và SCL) cua mach là đã có the hien th% thông tin lên LCD. Ngoài ra trên module còn có san bien tro đe đieu chinh đ® tương phan cua LCD. Đ%a chi I2C cua mach o lu¾n văn này là 0x27.

### Đ®ng cơ RC servo 9G



Hình 2.17: Đ®ng cơ RC servo 9G

Đ®ng cơ RC Servo 9G có kích thưóc nho, là loai đưoc su dnng nhieu nhat đe làm các mô hình nho. Đ®ng cơ RC Servo 9G có toc đ® phan úng nhanh, các bánh răng làm bang nhna nên không the tai n¾ng đưoc.

#### Thông so ky thu¾t:

* Đi¾n áp hoat đ®ng: 4.8-5VDC
* Toc đ®: 0.12 sec/60 degrees(4.8VDC)
* Lnc kéo: 1.6KG.CM
* Kích thưóc: 21x12x22mm
* Trqng lưong: 9g

### Đ®ng cơ bơm P385



Hình 2.18: Đ®ng cơ bơm P385

Đ®ng cơ bơm P385 12VDC 3W có kích thưóc nho gqn, đưoc su dnng đe bơm nưóc, dung d%ch vói kha năng bơm toi đa lên đen 1.8L / 1 phút, đ®ng cơ su dnng đi¾n áp 12VDC, thích hop vói các thiet ke su dnng máy bơm nho: bơm ho cá, tưói nưóc cho cây,. . . lưu ý không cap ngưoc cnc vì có the làm hư cơ cau bơm cua đ®ng cơ.

#### Thông so kĩ thu¾t:

* Đ®ng cơ DC: 385.
* Đi¾n áp su dnng: 12VDC.
* Dòng đi¾n su dnng: 0.25A.
* Công suat: 3W.
* Lưu lưong bơm: 1,8 L / phút.
* Áp suat nưóc: 0.3Mpa.
* Thòi gian làm vi¾c liên tnc toi đa trong 1 ngày: không quá 8h.
* Kích thưóc: 86 x 46 x 46 mm.

### Led dây 12V



Hình 2.19: Led dây 12V

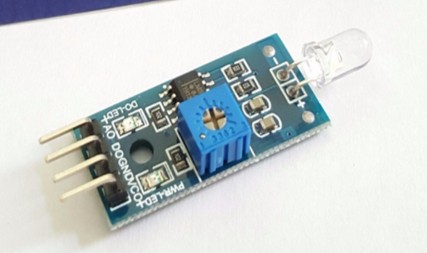
Đèn led dây 5050 su dnng chip led 5050 ưu vi¾c hơn các chip led khác 3528 và 3014. Cho ánh sáng tot hơn han, hình dáng đep, tuoi thq cao.

#### Thông so kĩ thu¾t:

* Đi¾n áp hoat đ®ng: 12 VDC.
* So lưong led: 60 led / m.
* Đi¾n áp hoat đông: 1.2 A / m.
* Công suat: 14.4 W / m.
* Đ® lux: 780 – 900 Lumens / m.
* Kích thưóc: 100cm x 1cm x 0.25cm.

**Lưu ý:** Có the su dnng adaptor chung vói 2 bơm vì dòng tiêu thn nho khoang 1.2A trên 1m nên có the dùng chung van đu dòng.

### Cam bien ánh sáng Photodiod



Hình 2.20: Cam bien ánh sáng Photodiod

Cam bien Photodiod su dnng Photodiod thay vì quang tro đe cam bien ánh sáng nên cho đ® chính xác và on đ%nh cao, cam bien có tích hop Op-Amp cho ca hai ngõ ra tín hi¾u là Digital và Analog rat de su dnng, thích hop cho các úng dnng nh¾n biet sáng toi, cam bien cưòng đ® sáng.

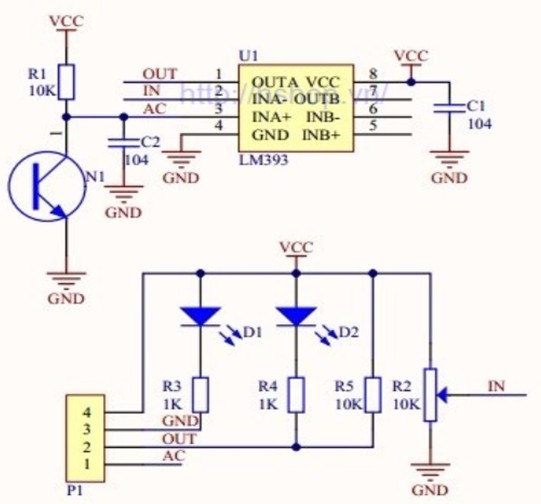
#### Thông so kĩ thu¾t:

* Nguon: 3.3 - 5 VDC.
* Su dnng Photodiod cho đ® chính xác cao.
* Xuat tín hi¾u Digital tinh chinh bang bien tro trên mach ho¾c Analog rat de su dnng. Lu¾n văn này se su dnng chân analog cua cam bien.
* Kích thưóc: 30 x 16 mm.

#### Cam bien ra 4 chân:

* A0: Ngõ ra tín hi¾u Analog.
* D0: Ngõ ra tín hi¾u Digital.
* Vcc : Nguon 3.3 - 5 VDC.
* GND : Mass.

#### Sơ đo nguyên lí cua cam bien Photodiod



Hình 2.21: Sơ đo nguyên lí cam bien ánh sáng

### Cam bien lưu lưong nưóc



Hình 2.22: Cam bien lưu lưong

Cam bien lưu lưong nưóc là cam bien thưòng dùng trong các máy bơm nưóc ho cá, máy bơm mini, máy nưóc nóng,. . . Cam bien hoat đ®ng dna trên quat nưóc và cam bien Hall bên trong, khi nưóc chay qua làm quat nưóc quay -> cam bien Hall -> xung vuông (tù NPN).

#### Công thNc tính lưu lưeng

- Q = F / 7.5.

* F: tan so (Hz).
* Q: lưu lưong (L / min).
* 7.5 là hang so.

#### Thông so kĩ thu¾t

* Nguon: 5 – 24 V.
* Dòng tiêu thn: < 10mA.
* Ch%u áp lnc đen: 1.75Mpa.
* Lưu lưong đo: 1 – 30 (L / min).
* Nhi¾t đ® hoat đ®ng: < 120 đ® C.

- Đ® am: 35% - 90% RH.

* Kích thưóc: 61 x 36 x 34 mm.

#### Hi¾u Nng Hall

Hi¾u úng Hall đưoc khám phá boi Edwin Herbert Hall vào năm 1879 và tro thành m®t trong các hi¾u úng cơ ban cua v¾t lý hqc. Hi¾u úng Hall là hi¾u úng xuat hi¾n trên m®t v¾t dan có dòng đi¾n chay qua khi đưoc đ¾t vào trong m®t tù trưòng.

M®t hi¾u úng v¾t lý đưoc thnc hi¾n khi áp dnng m®t tù trưòng vuông góc lên m®t ban làm bang kim loai hay chat bán dan hay chat dan đi¾n nói chung (thanh Hall) đang có dòng đi¾n chay qua. Lúc đó ngưòi ta nh¾n đưoc hi¾u đi¾n the (hi¾u the Hall) sinh ra tai hai m¾t đoi di¾n cua thanh Hall. Tý so giua hi¾u the Hall và dòng đi¾n chay qua thanh Hall gqi là đi¾n tro Hall, đ¾c trưng cho v¾t li¾u làm nên thanh Hall (m®t chat bán dan).

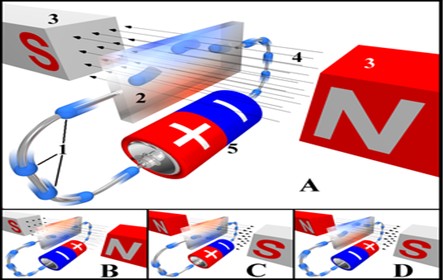
Công thúc liên h¾ giua hi¾u the Hall, dòng đi¾n và tù trưòng là:

VH = (IB) / (den)

vói:

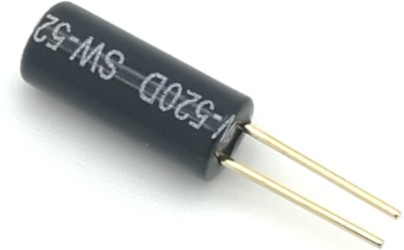
VH: hi¾u đi¾n the Hall. I: cưòng đ® dòng đi¾n. B: là cưòng đ® tù trưòng. d: đ® dày cua thanh Hall.

e: đi¾n tích cua hat mang đi¾n chuyen đ®ng trong thanh Hall. n: m¾t đ® các hat này trong Hall.



Hình 2.23: Hi¾u úng Hall

### Cam bien rung SW520D



Hình 2.24: Cam bien rung SW520D

Đây là loai cam bien rung dùng bi có đ® nhay cao, ít anh hưong boi âm thanh bên ngoài.

Thưòng dùng trong các h¾ thong báo đ®ng, chong tr®m, nh¾n biet rung đ®ng cua v¾t the.

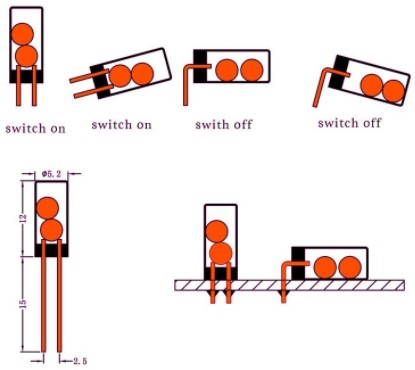
Nguyên lí hoat đ®ng: khi không có rung thì tro kháng gan bang 0. Khi có rung đ®ng ho¾c nghiêng tro kháng lón.

#### Thông so kĩ thu¾t:

- Đi¾n áp: 3 - 12 V.

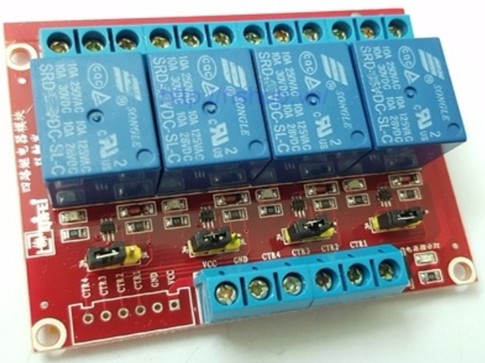
* Dòng hoat đ®ng: 10 mA.
* Tro kháng: 10 M.

#### Hình minh hqa nguyên lí hoat đ®ng cua cam bien rung SW520D:



Hình 2.25: Nguyên lí hoat đ®ng SW520D

### Module Relay 5VDC



Hình 2.26: Module Relay

Module 4 relay kích H/L su dnng nguon 5VDC đe nuôi mach, tín hi¾u kích có the tùy chqn kích múc cao (5VDC) ho¾c múc thap (0VDC) qua jumper trên moi relay. Thích hop cho các thiet b% su dnng tín hi¾u 5VDC như vi đieu khien,. . .

#### Thông so kĩ thu¾t:

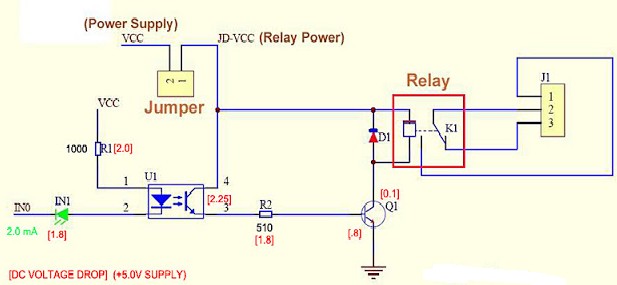
* Đi¾n áp nuôi mach: 5 VDC.
* Dòng tiêu thn: khoang 200 mA / 1 Relay.
* Tín hi¾u kích: có the tùy chqn cao ho¾c thap thông qua jumper.
* Relay trên mach:

+ Nguon nuôi: 5VDC.

+ Tiep điem đóng ngat max: 250VAC – 10A ho¾c 30VDC – 10A.

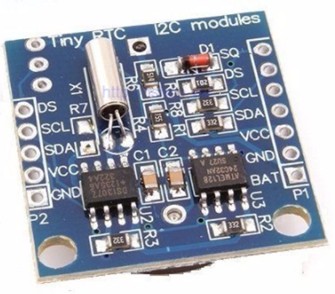
* Kích thưóc: 72 x 55 x 19 mm.

#### Sơ đo nguyên bên trong 1 relay:



Hình 2.27: Sơ đo nguyên lí 1 relay

### Mach thòi gian thnc RTC DS1307



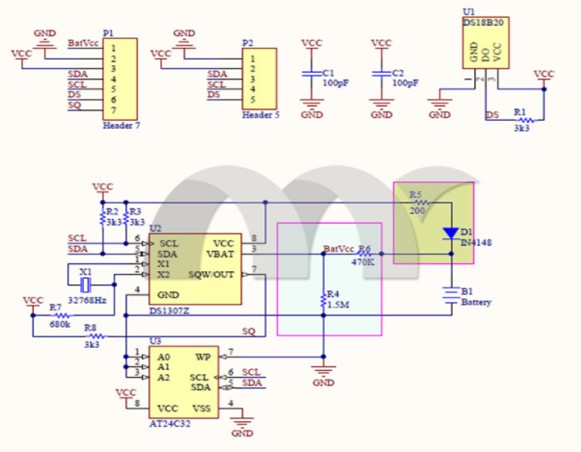
Hình 2.28: RTC DS1307

Module thòi gian thnc DS1307 có chúc năng lưu tru thòi gian, nó hoat đ®ng như 1 chiec đong ho và có the xuat du li¾u ra ngoài qua giao thnc I2C.

#### Thông so kĩ thu¾t:

* Nguon cung cap: 5VDC.
* Kha năng lưu tru 32K bit vói EEPROM AT24C32.
* Su dnng giao thúc I2C.
* Lưu tru thông tin giò phút giây.
* Lưu tru chính xác đen 2100.
* Có pin đong ho lưu tru thông tin.
* Có ngõ ra tan so 1Hz.
* Kích thưóc: 16 x 22 x23 mm.

#### Sơ đo nguyên lí RTC DS1307



Hình 2.29: Sơ đo nguyên lí RTC DS1307

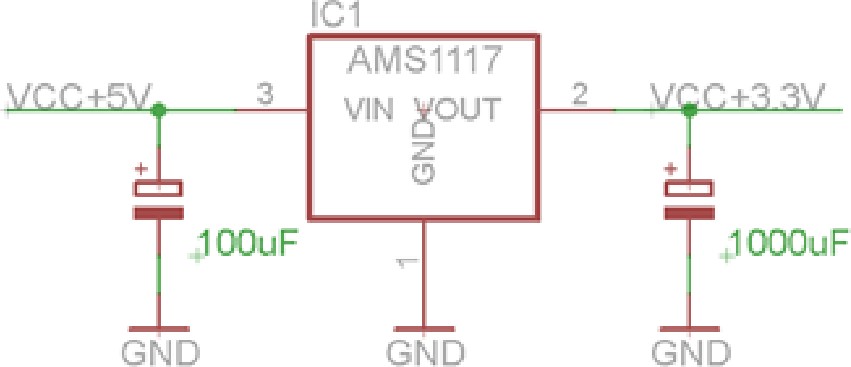
### Mach giam áp AMS1117



Hình 2.30: Hình anh thnc te mach giam áp

Dùng đe chuyen nguon tù (4.5V – 7V ) sang 3,3V đe cap nguon cho ESP8266. O đây là chuyen tù 5V sang 3.3V đe cap cho 2 chân Vcc và CH-PD cua ESP8266.

#### Sơ đo nguyên lí mach giam áp AMS 1117:



Hình 2.31: Sơ đo nguyên lí mach giam áp AMS 1117

# Chương 3

## THIET KE VÀ THUC HIfiN PHAN CÚNG

## Yêu cau thiet ke

Thiet ke mô hình ho cá thông minh có m®t so chúc năng:

* + - Đieu khien bơm thay nưóc: su dnng 2 bơm nưóc 12VDC, 1 bơm vào ho, 1 xa nưóc cũ tù ho ra.
    - Đieu khien b¾t tat đèn thap sáng ho cá. Cn the là đèn led dây xung quanh đe thap sáng ho khi can thiet.
    - Đieu khien đ®ng cơ cho cá ăn tn đ®ng. Su dnng mô hình servo xoay đe đay thúc ăn ra cho

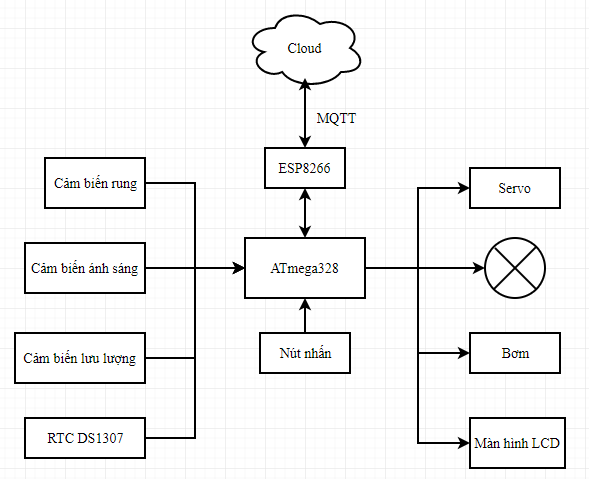
cá.

* + - Tat ca đeu có 2 chúc năng: đieu khien tù xa thông qua đi¾n thoai di đ®ng ho¾c có the đieu

khien trnc tiep tai cho bang nút nhan.

* + - Mô hình ho cá có h¾ thong hoi tiep đe đam bao rang các công vi¾c đã thnc hi¾n xong. Vi¾c hoi tiep đưoc su dnng bang các cam bien: cam bien lưu lưong (xác đ%nh hoat đ®ng cua bơm), cam bien ánh sáng (xác đ%nh trang thái đèn led), cam bien rung (xác đ%nh thúc ăn đã rơi ra chưa).
    - Có b® hien th% trang thái cua các mnc tiêu đieu khien tai cho và tù xa. Hien th% tai cho là hien th% thông qua màn hình LCD các trang thái ket noi b¾t tat cua ho cá. Hien th% trang thái gui và nh¾n du li¾u tù xa thông qua úng dnng trên đi¾n thoai di đ®ng.

## Sơ đo khoi tong quát



Hình 3.1: Sơ đo khoi phan cúng

* + - ESP8266 se là 1 cau noi giua du li¾u trên cloud vói du li¾u đieu khien cua ngưòi dùng.

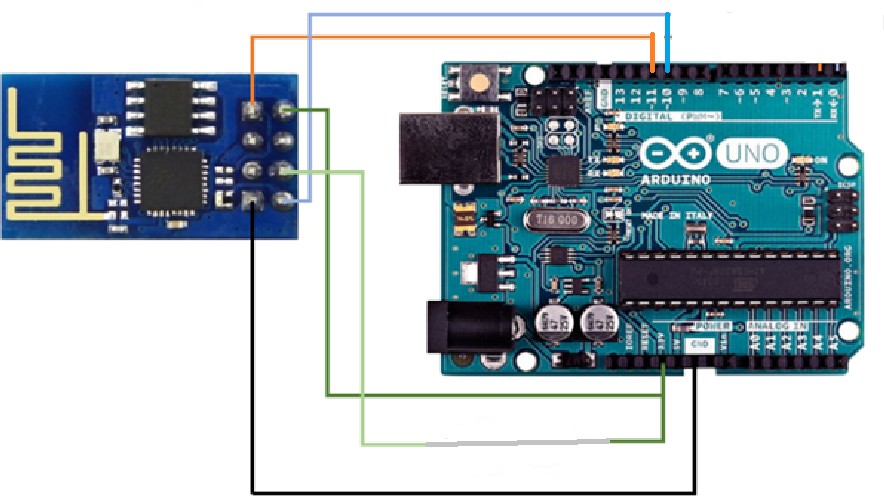
Giao thúc giua ESP8266 và cloud se là su dnng giao thúc MQTT.

* + - Cam bien rung SW520D se thnc hi¾n đqc tín hi¾u tù mô hình cho ăn đe gui ve cho vi đieu khien biet thông tin trang thái cho ăn.
    - Cam bien ánh sáng photodiod đqc đ® sáng cua đèn đe gui trang thái ve cho vi đieu khien.
    - Cam bien lưu lưong đqc lưu lưong nưóc đe gui trang thái bơm ve cho vi đieu khien biet.
    - RTC DS1307 đqc thòi gian thnc gui đen cho vi đieu khien hien th%.
    - Nút nhan nh¾n tín hi¾u đieu khien đe gui đen vi đieu khien đưa ra tín hi¾u đieu khien.
    - Servo, Đèn, Bơm nh¾n tín hi¾u đieu khien b¾t tat tù vi đieu khien.
    - Màn hình LCD hien th% các thông tin trang thái cua ho cá lên màn hình.
    - ATmega328 làm nhi¾m vn xu lí thông tin và đưa ra tín hi¾u đieu khien cho mô hình.

## Ket noi phan cNng

### Ket noi Arduino và ESP8266

Ket noi giua Arduino và ESP8266 là UART, nên ta se su dnng 2 chân cua ATmega328P ket noi vói RXD và TXD cua ESP8266.



Hình 3.2: Ket noi giua ESP8266 vói Arduino Uno R3

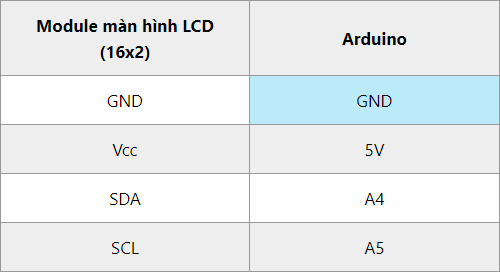
#### Ket noi giÑa ESP8266 và Arduino Uno R3 theo bang sau:

|  |  |
| --- | --- |
| **ESP8266** | **Arduino Uno R3** |
| TX | Pin 12 |
| RX | Pin 13 |
| Vcc | 3.3V |
| GND | GND |
| CH-PD | 3.3V |

Bang 3.1: Ket noi giua ESP8266 và Arduino Uno R3

### LCD vói Arduino

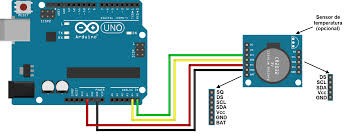
Su dnng module I2C o giua đe giam bót so lưong chân su dnng trên vi đieu khien. Ket noi LCD vói module I2C theo đúng thú tn chân. Sau đó ket noi module I2C vói vi đieu khien theo bang sau:



Bang 3.2: Ket noi giua module I2C vói Arduino

### Module DS1307 vói Arduino

Ket noi module Real time clock vói Arduino như sau:



Hình 3.3: Ket noi giua DS1307 vói Arduino

### Các cam bien vói Arduino

#### Cam bien ánh sáng

Cam bien ánh sáng Photodiod có ca 2 chân analog và digital o đây su dnng chân analog đe đqc du li¾u. Ket noi theo bang sau:

|  |  |
| --- | --- |
| **Cam bien ánh sáng Photodiod** | **Arduino Uno R3** |
| A0 | A1 |
| Vcc | 3.3V |
| GND | GND |

Bang 3.3: Ket noi giua cam bien ánh sáng vói Arduino

#### Cam bien lưu lưeng

O đây su dnng 2 cam bien lưu lưong nên su dnng 2 chân ngat cua vi đieu khien. Và su dnng nguon 5V cho cam bien.

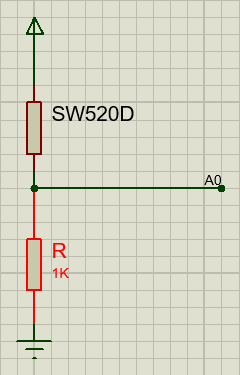
Ket noi giua cam bien lưu lưong vói vi đieu khien theo bang sau:

|  |  |
| --- | --- |
| **Cam bien lưu lưeng nưéc** | **Arduino Uno R3** |
| Dây đo | 5V |
| Dây đen | GND |
| Dây vàng | D2 và D3 |

Bang 3.4: Ket noi giua cam bien lưu lưong vói Arduino

#### Cam bien rung

Cam bien rung SW520D là 1 đi¾n tro bi nên thnc hi¾n thiet ke mach cho cam bien rung:

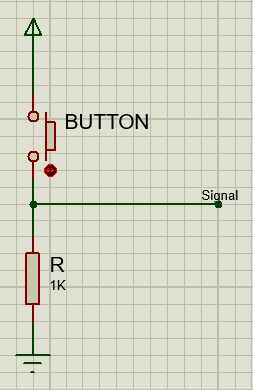


Hình 3.4: Mach cam bien rung

Cam bien se đưoc noi vói chân A0 cua vi đieu khien đe đqc du li¾u Analog.

### Các nút nhan vói Arduino

Su dnng 4 nút nhan trong mô hình. Và mach cho các nút nhan đưoc thiet ke như hình sau:



Hình 3.5: Mach nút nhan

Có 4 nút nhan su dnng 4 chân digital cua vi đieu khien lan lưot là 8, 9, 10 và 11.

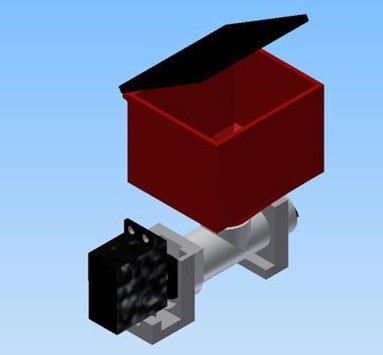
### Bang ket noi chân vi đieu đen các ngoai vi

|  |  |
| --- | --- |
| **Các ngoai vi** | **Arduino Uno R3** |
| Cam bien rung SW520D | A0 |
| Cam bien ánh sáng | A1 |
| Thòi gian thnc DS1307 và Module I2C (SDA và SCL) | A4 và A5 |
| Cam bien lưu lưong nưóc 1 | D2 |
| Cam bien lưu lưong nưóc 2 | D3 |
| Đ®ng cơ bơm P385 1 | D4 |
| Đ®ng cơ bơm P385 2 | D5 |
| Đèn led 12V | D6 |
| Đ®ng cơ cho cá ăn (servo) | D7 |
| Nút đieu khien servo | D8 |
| Nút đieu khien đèn led | D9 |
| Nút đieu khien Bơm | D10 |
| Nút đieu khien tat ca | D11 |
| ESP8266 TX | D12 |
| ESP8266 RX | D13 |

Bang 3.5: Bang tóm tat các ket noi đen Arduino Uno R3

### Mô hình 3D cho thiet b% cho các ăn

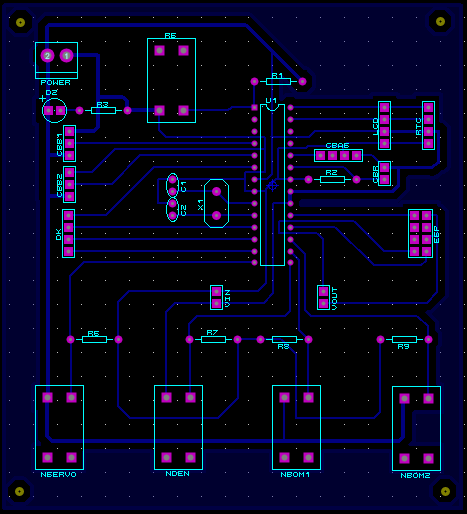
Mô hình nhna in 3D, đưoc tham khao tù trang web *thingiverse.com*. Su dnng đ®ng cơ Servo hoat đ®ng như đ®ng cơ DC thông thưòng đe quay trnc đay thúc ăn xuong.



Hình 3.6: Mô hình cho cá ăn

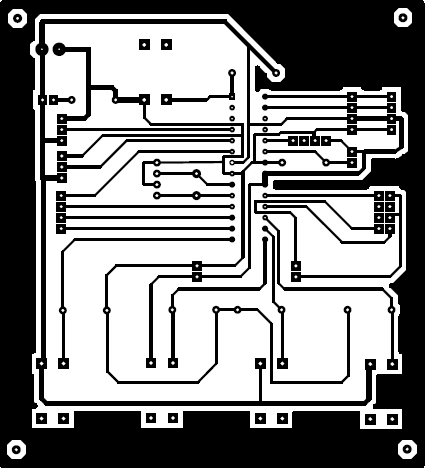
## ThNc hi¾n làm mach in

### Ve mach và layout trên protues



Hình 3.7: Layout trên protues

### Xuat mach in ra file PDF



Hình 3.8: Xuat mach in ra file PDF

# Chương 4

## THIET KE VÀ THUC HIfiN PHAN MEM

## Yêu cau đ¾t ra cho phan mem

Các yêu cau đ¾t ra khi thiet ke phan mem là:

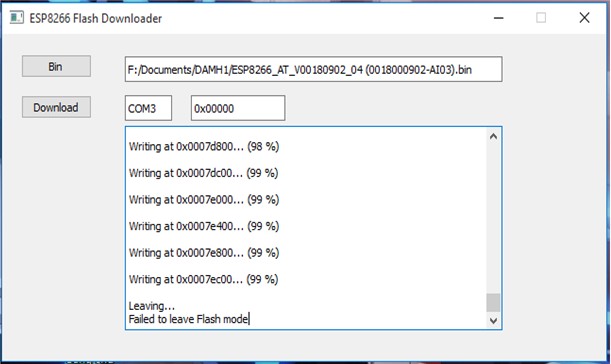
* + - Đieu khien trnc tiep tai cho. Thông qua vi¾c đqc du li¾u các nút nhan.
    - Đieu khien tù xa. Đieu khien tù xa thông qua giao thúc MQTT.
    - Hien th% các trang thái lên màn hình LCD.
    - Đqc thòi gian thnc tù module RTC DS1307.
    - Gui và nh¾n du li¾u thông qua giao thúc MQTT.
    - Đqc giá tr% các cam bien đe theo dõi trang thái hoat đ®ng.

## Nap lai firmware cho ESP8266

Firmware cho ESP8266 hoat đ®ng vói giao thúc MQTT đưoc tìm thay tai trang web:

*https://thingsboard.io/docs/samples/arduino/temperature/*

Quá trình nap firmware đưoc thnc hi¾n như hình:



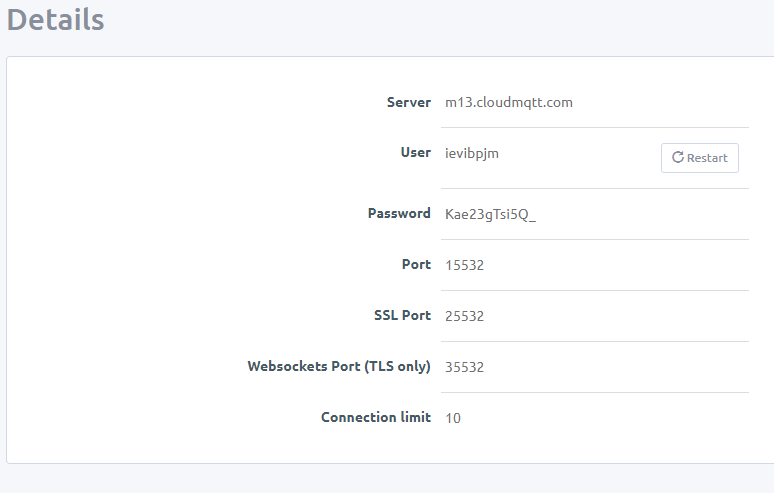
Hình 4.1: Nap firmware MQTT cho ESP8266

**Lưu ý:** Quá trình nap đen 99% nghĩa là đã thành công.

## Tao kênh trên cloudMQTT

Tien hành đăng ký tao 1 tài khoan trên trang: [*https://www.cloudmqtt.com/*](http://www.cloudmqtt.com/)

Đây là các chi tiet cua 1 tài khoan mà mình đã so huu sau khi đăng ký:



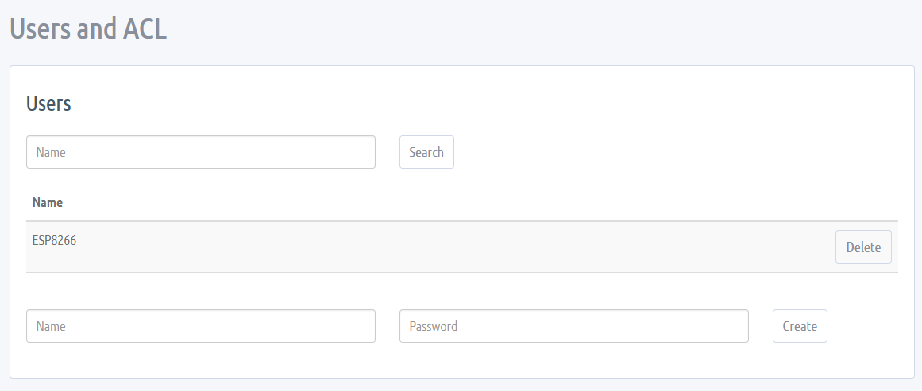
Hình 4.2: Thông tin cua tài khoan

Trên trang thông tin se hien th% m®t so chi tiet cua tài khoan: - MQTT server là: m13.cloudmqtt.com.

- Port su dnng là: 15532. - Và so lưong cong đưoc phép ket noi là 10. Do đang su dnng tài khoan mien phí nên có giói han. Neu su dnng và mua các tài khoan có tra phí thì se đưoc nhieu cong ket noi hơn.

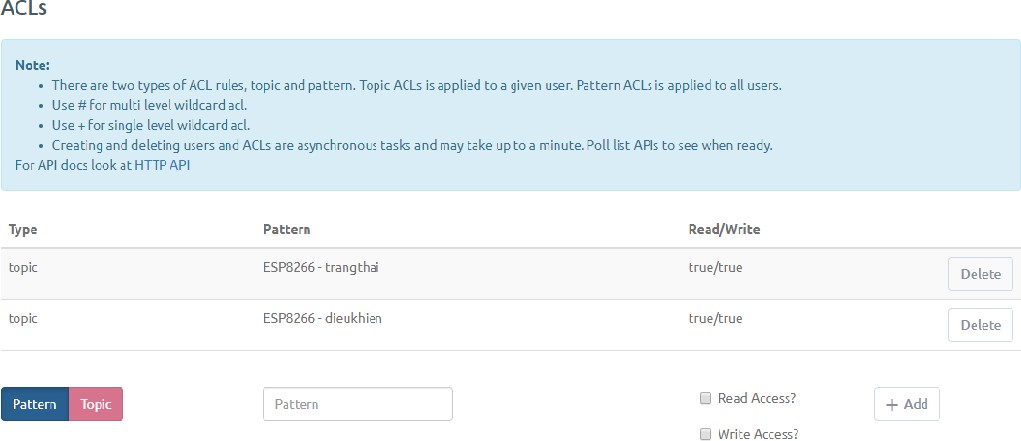
Sau khi đã có 1 tài khoan thì se tien hành tao 1 user cua mình đe chúa các topic mong muon.

Tao user mang tên ESP8266 Như trong hình:



Hình 4.3: Tao user

Tao 2 topic lan lưot mang tên trangthai và dieukhien đe pub và sub các tín hi¾u qua lai giua ngưòi dùng và vi đieu khien, như trong hình:



Hình 4.4: Tao topic

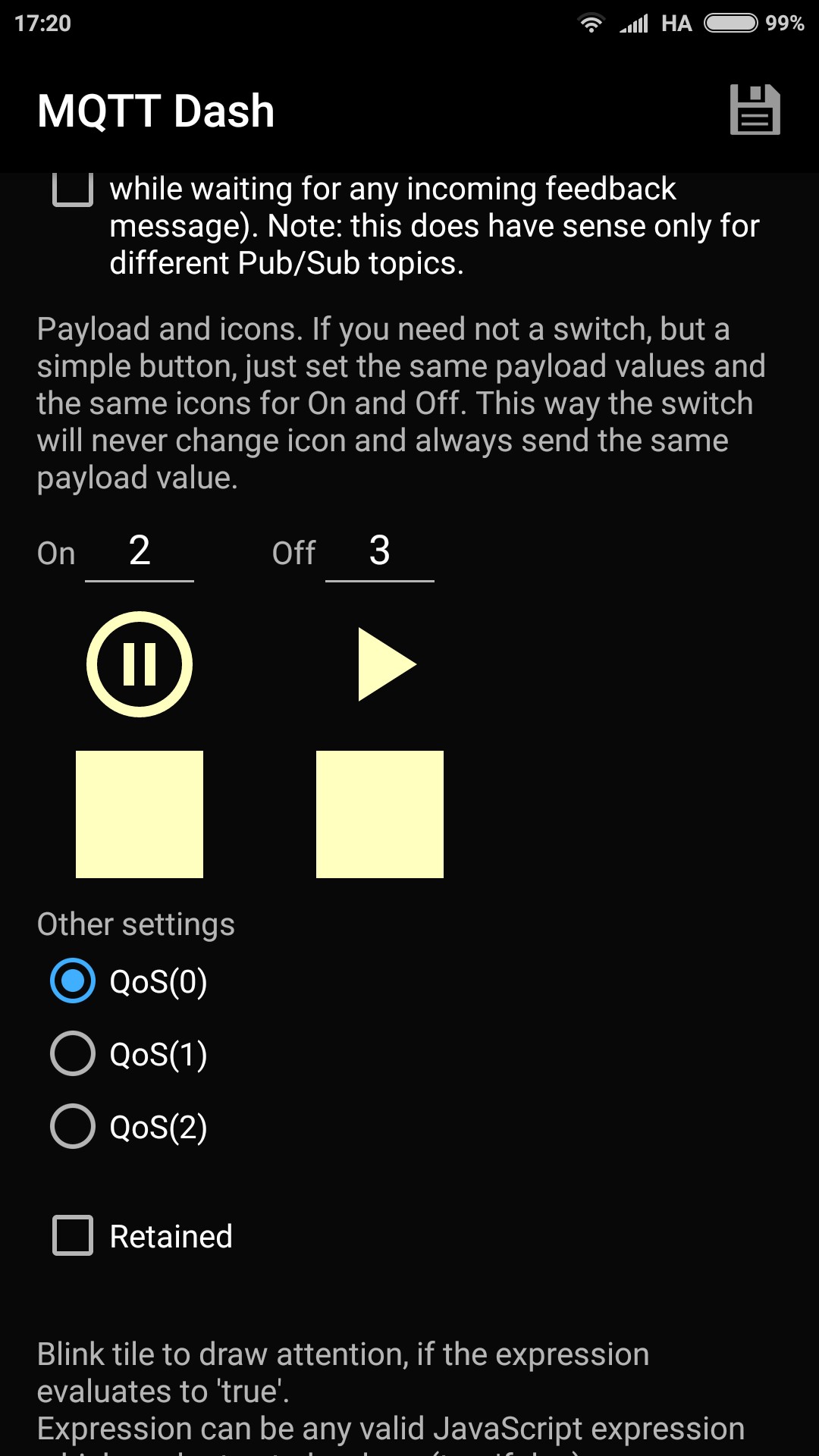
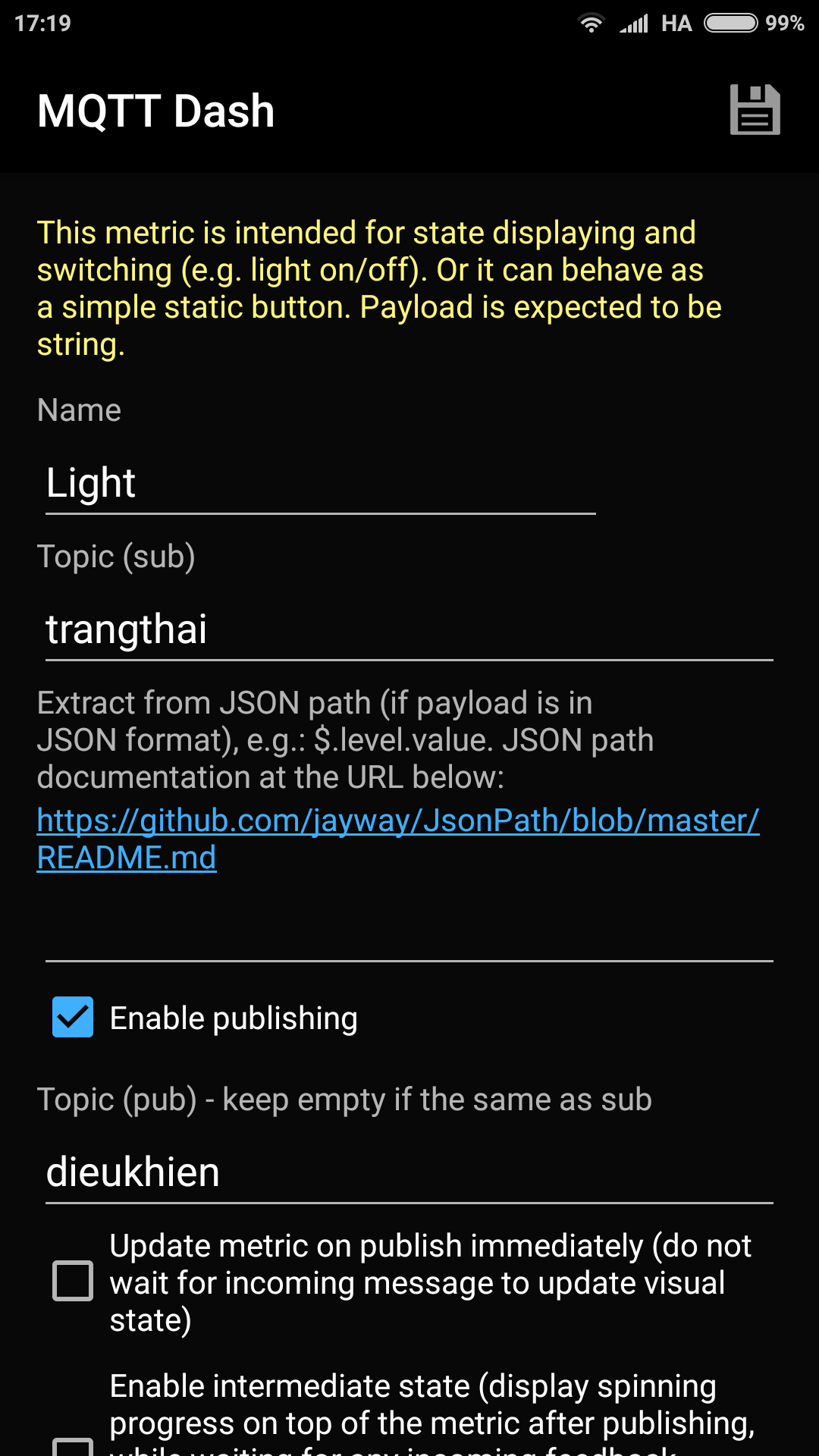
## Thiet ke giao di¾n trên đi¾n thoai

App su dnng cho đi¾n thoai đưoc su dnng tai:

*https://play.google.com/store/apps/details?id=net.routix.mqttdash&hl=vi*

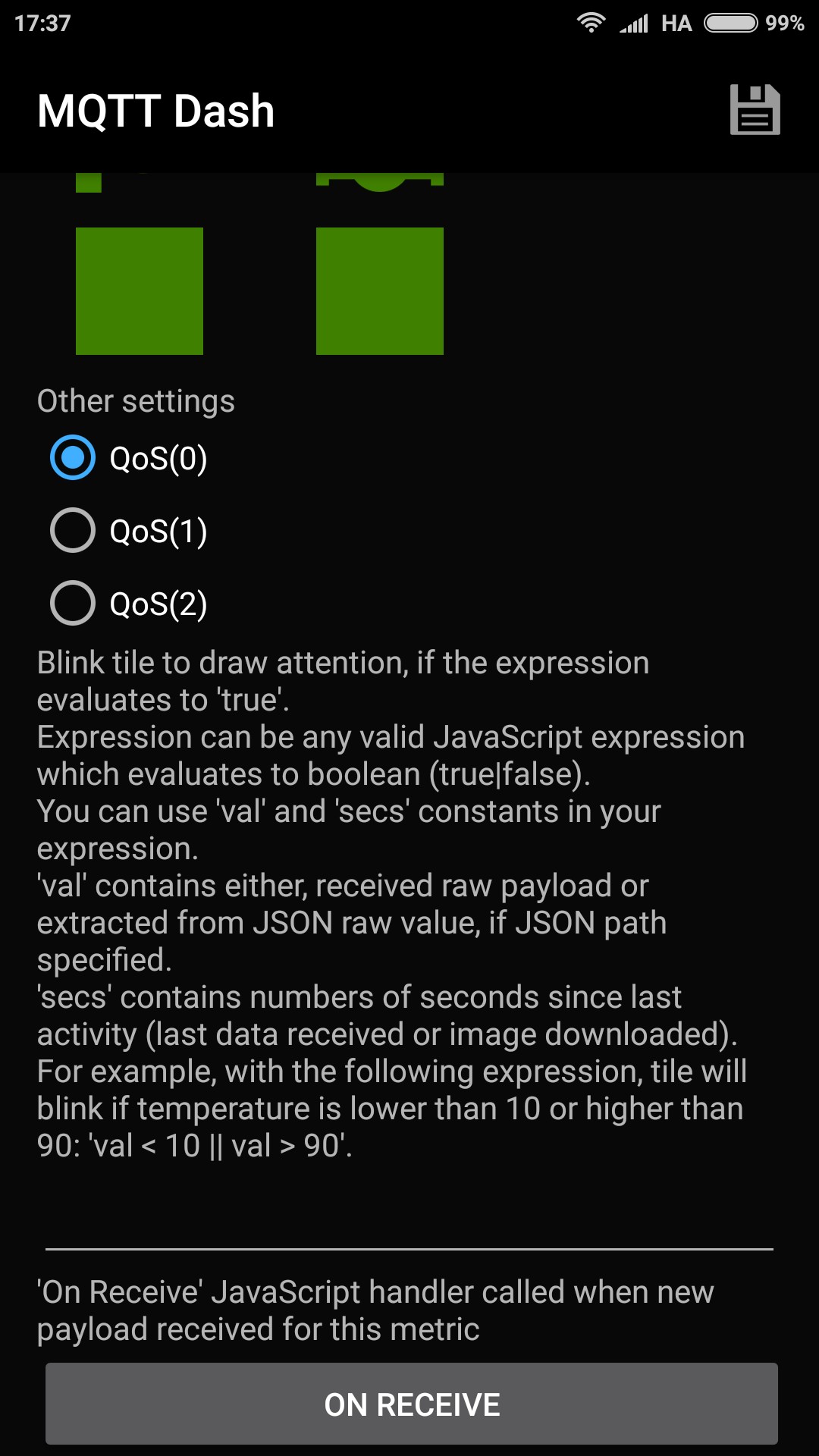
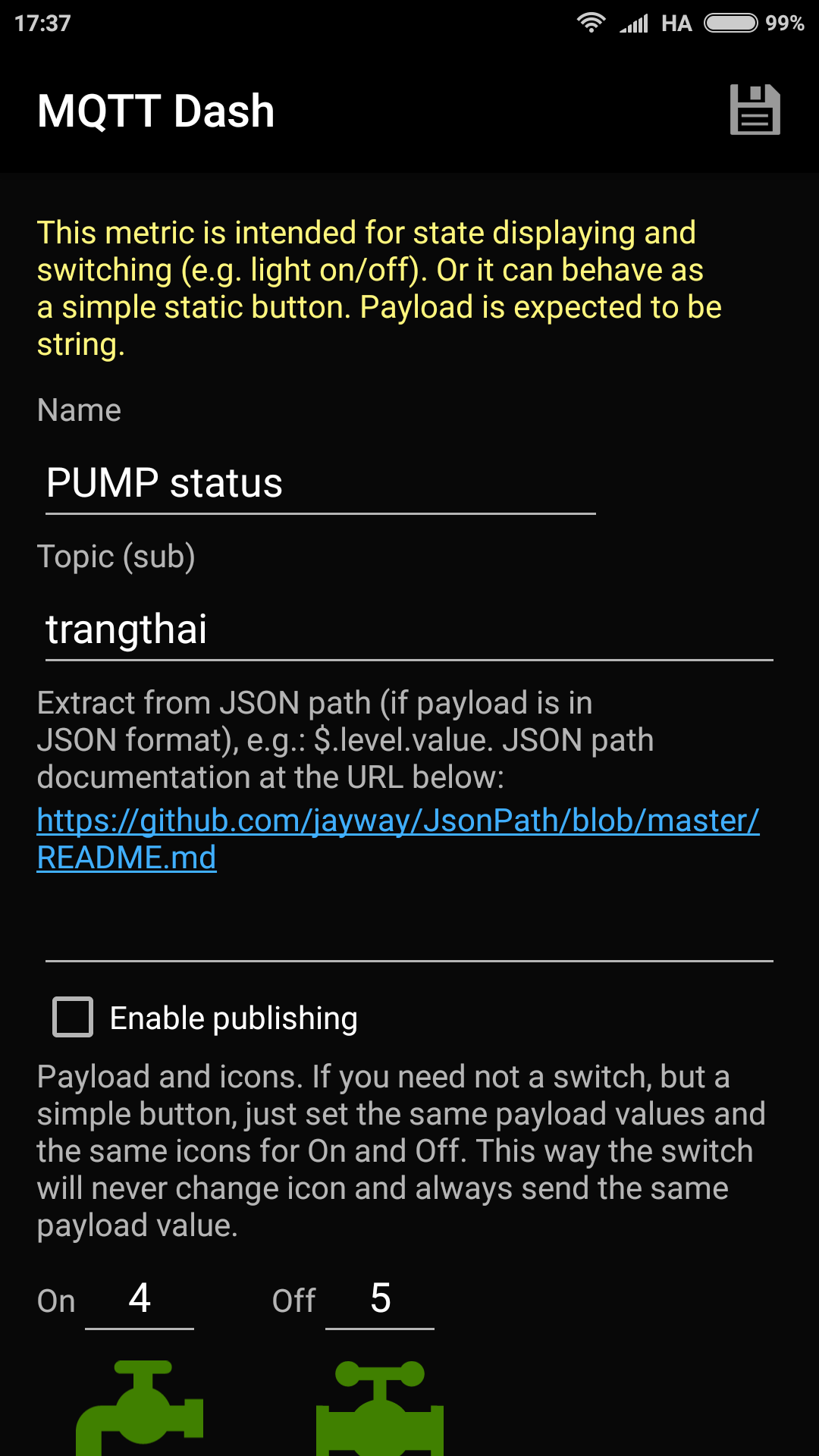
Giao di¾n trên đi¾n thoai bao gom:

* 3 nút nhan đieu khien servo, đèn, bơm. Đưoc thiet ke sau khi nhan thì se publish m®t tin nhan đen topic "dieukhien". Ví dn trong hình là cài đ¾t nút nhan đèn led.



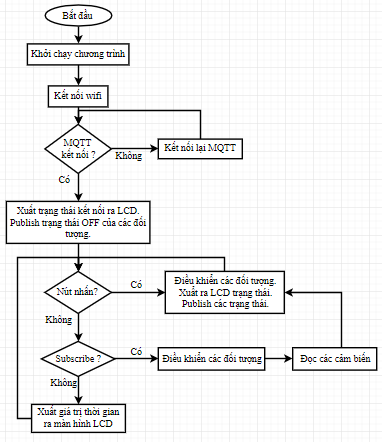
Hình 4.5: Cài đ¾t publish trên điên thoai

* 3 nút hien th% trang thái cua đoi tưong khi đưoc đieu khien tù xa. Đưoc thiet ke là subscribe topic trangthai đe nh¾n đưoc tín hi¾u gui đi tù ESP8266 neu có thay đoi trang thái cua thiet b% khi nó đưoc đieu khien. Ví dn trong hình là cài đ¾t nút trang thái cua bơm.



Hình 4.6: Cài đ¾t subscribe trên đi¾n thoai

## Lưu đo giai thu¾t



Hình 4.7: Lưu đo giai thu¾t phan mem

#### Chi tiet các khoi trong lưu đo giai thu¾t

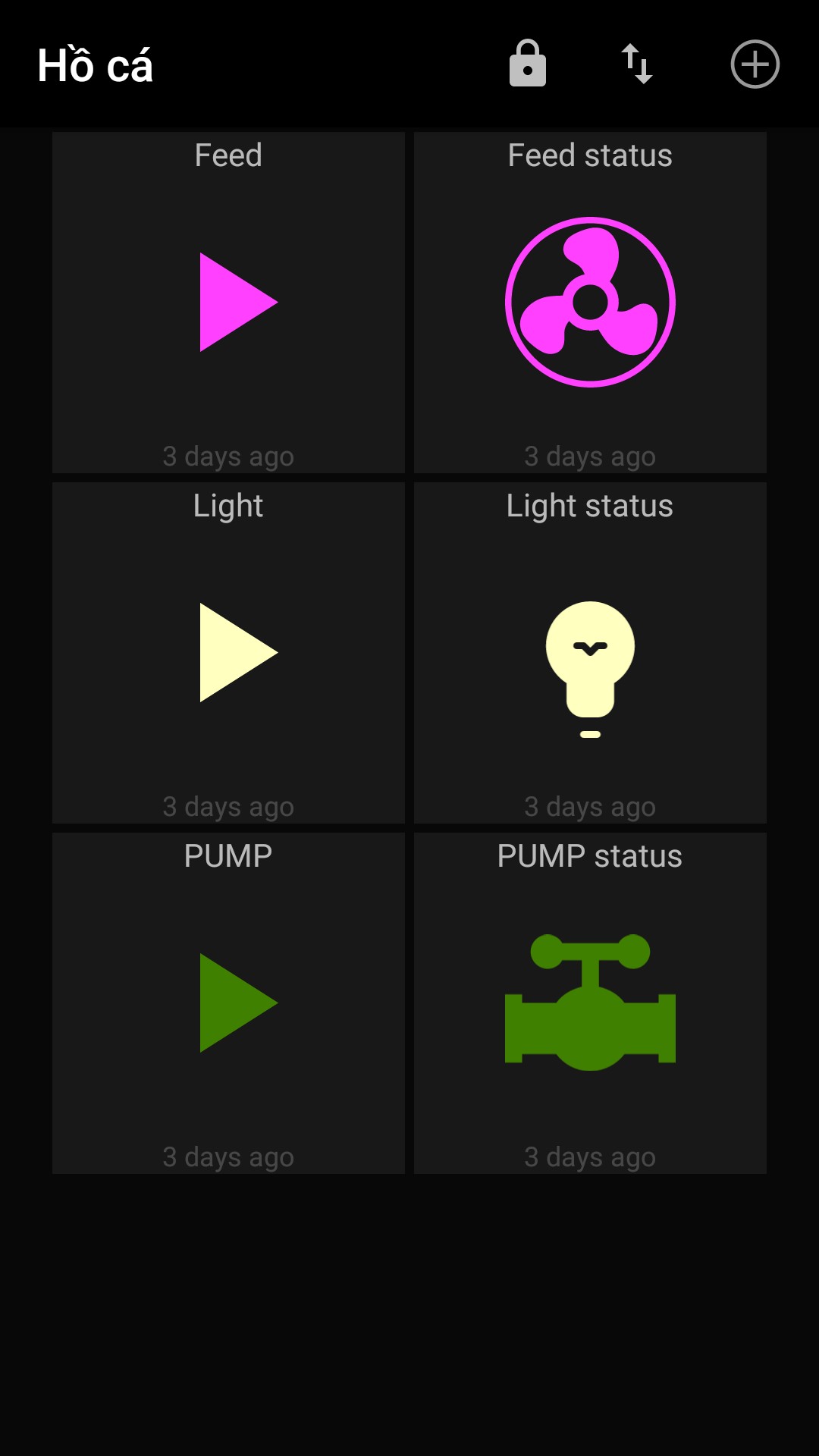
* + - Khoi chay chương trình: Thnc hi¾n tao các bien, khoi tao các cong,...
    - Ket noi wifi: Chay code ket noi vói wifi sau đó xuat ra màn hình LCD thông báo trang thái ket noi cho ngưòi dùng biet.
    - Kiem tra ket noi MQTT: Neu chưa thì se thnc hi¾n ket noi lai. Neu đã ket noi thì se publish cho client biet và sau đó se hien th% trang thái ra LCD.
    - Xuat trang thái ket noi cua các đoi tưong, publish trang thái OFF cua các đoi tưong lên server MQTT.
    - Kiem tra nút nhan: Neu đưoc nh¾n thì se đieu khien các đoi tưong, xuat ra màn hình LCD, publish trang thái các đoi tưong sau khi đieu khien.
    - Kiem tra trang thái topic đã subscribe, neu có tin nhan tói thì se nh¾n l¾nh và đieu khien đoi tưong.
    - Đqc cam bien: đqc giá tr% các cam bien sau đó se xu lí.
    - Xuat giá tr% thòi gian thnc ra màn hình LCD.

# Chương 5

## KET QUÂ THUC HIfiN

* 1. **Phan đieu khien tN xa**

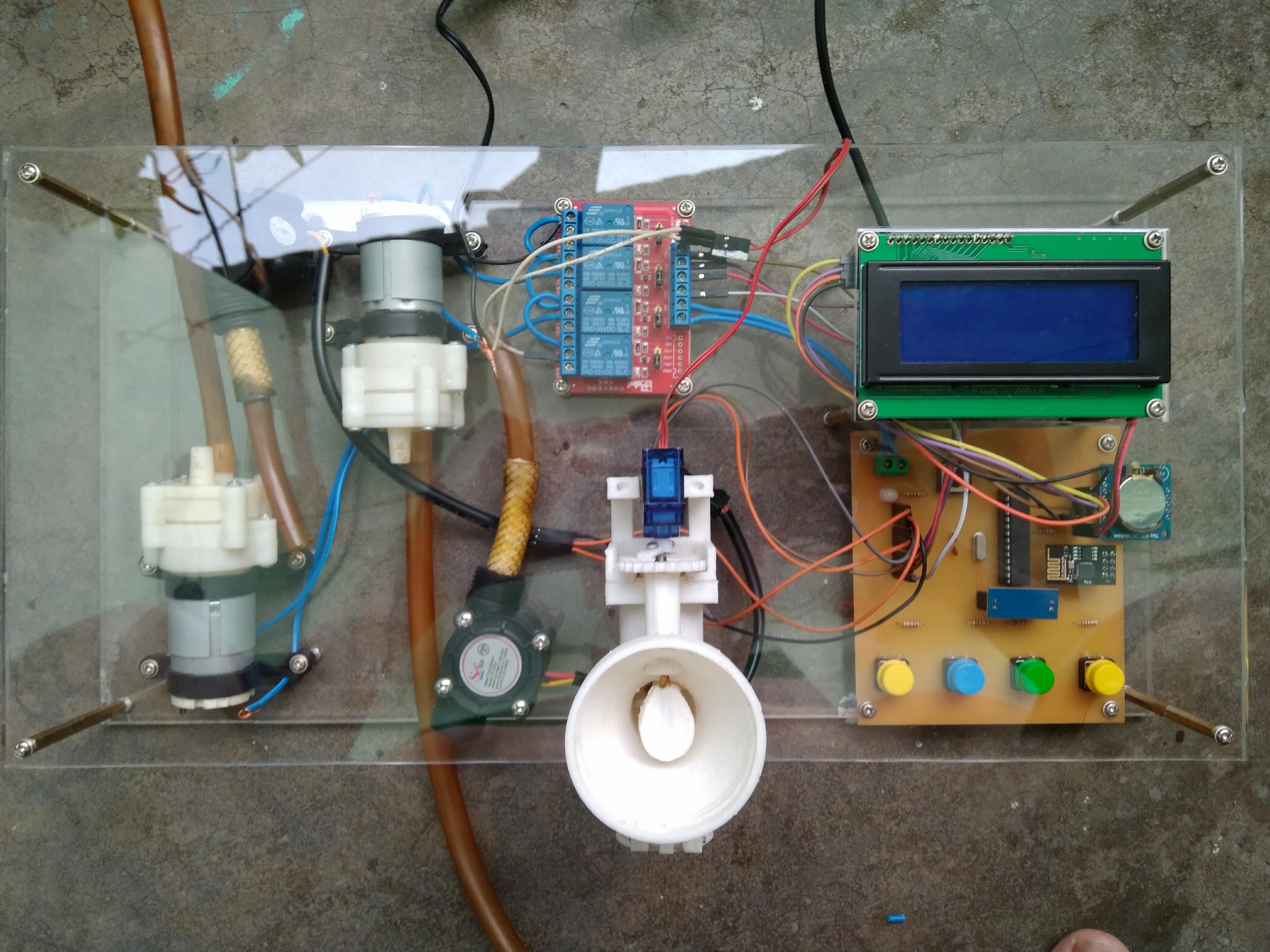
Giao di¾n chương trình sau khi đã đưoc thiet ke lai trên đi¾n thoai:



Hình 5.1: Giao di¾n trên đi¾n thoai

## Mô hình ho cá thông minh

Mô hình ho cá thông minh sau khi thiet ke phan cúng:



Hình 5.2: Mô hình ho cá thông minh

# Chương 6

## KET LU¾N VÀ HƯéNG PHÁT TRIEN

## Ket lu¾n

Thnc hi¾n đưoc mô hình ho cá thông minh vói các yêu cau đ¾t ra. Nhưng van còn chưa có kha năng hen giò tù đong ho thòi gian thnc.

#### Ưu điem:

* + - Là m®t đe tài có tính úng dnng cao, có tính kha thi, và góp phan nâng cao chat lưong đòi song cua con ngưòi.
    - San pham nho gqn, có the di chuyen m®t cách de dàng.
    - Đáp úng đưoc nhu cau ve m®t mô hình ho cá thông minh.

#### Nhưec điem:

* + - Module ESP8266 không on đ%nh, su dnng thòi gian dài dan đen ch¾p chòn, dan đen mat tin hi¾u không đieu khien đưoc.
    - Mô hình in 3D còn ton kém.
    - Du li¾u chưa đưoc lưu tru lai.
    - Truyen bang cloudMQTT mien phí nên b% giói han ve so lưong tin nhan subscribe và publish.

## Hưéng phát trien

* + - Phát trien thêm chúc năng hen giò thòi gian thnc cho mô hình.
    - Thnc hi¾n thêm b® lưu tru đe xác đ%nh so lan đieu khien trong ngày đe theo dõi kĩ lưõng hơn tình trang thnc te cua ho cá.
    - Cai thi¾n mô hình cho cá ăn đe tiet ki¾m chi phí mach in 3D.
    - Phía phan mem se phai phát trien phan úng dnng đi¾n thoai có nhieu option hơn nua đe thnc hi¾n đay đu các tính năng thoa mãn nhu cau su dnng cua ngưòi dùng.

# Chương 7

## TÀI LIfiU THAM KHÂO

"Arduino UNO R3 là gì?" [Online].

[*http://arduino.vn/bai-viet/42-arduino-uno-r3-la-gi*,](http://arduino.vn/bai-viet/42-arduino-uno-r3-la-gi) Oct.25, 2016

David A. Mellis. "Building an Arduino on a Breadboard."[Online].

[*https://www*](http://www.arduino.cc/en/Main/Standalone)*.ar*[*duino.cc/en/Main/Standalone*](http://www.arduino.cc/en/Main/Standalone), Oct.18, 2016

"ATmega328." [Online].

[*http://www.microchip.com/wwwproducts/en/ATmega328*,](http://www.microchip.com/wwwproducts/en/ATmega328) Oct.23, 2016

Pcunha. "Automatic Fish Feeder with micro servo control V2." [Online].

*https://www.thingiverse.com/thing:736693*, May.15, 2017

Tuan PM. "Tìm hieu ve giao thúc MQTT - IoT protocol" [Online].

*https://kipalog.com/posts/Tim-hieu-ve-giao-thuc-MQTT—IoT-protocol*, Sep.30, 2017

Techmaster Team. "[IoT] Nhung đieu can biet ve giao thúc MQTT" [Online].

*https://techmaster.vn/posts/34394/iot-giao-thuc-mqtt-va-ung-dung-trong-iot*, Oct.07, 2017

"Documentation MQTT." [Online].

[*https://www.cloudmqtt.com/*](http://www.cloudmqtt.com/), Oct.14, 2017

"Temperature upload over MQTT using Arduino UNO, ESP8266 and DHT22 sensor." [Online]. *https://thingsboard.io/docs/samples/arduino/temperature/*, Oct.21, 2017

NTP\_PRO. "Giao tiep I2C và su dnng module Realtime clock DS1307 (module RTC)"

[Online]. [*http://arduino.vn/bai-viet/369-giao-tiep-i2c-va-su-dung-module-realtime-clock-ds1307*,](http://arduino.vn/bai-viet/369-giao-tiep-i2c-va-su-dung-module-realtime-clock-ds1307) Nov.11, 2017

Knolleary. "Arduino Client for MQTT." [Online].

*https://github.com/knolleary/pubsubclient/*, Nov.18, 2017

# Chương 8

## PHU LUC

#### Phan mã nguon chương trình

# i n c l u d e < W i F i E s p C l i e n t . h> # i n c l u d e < WiFiEsp . h>

# i n c l u d e <WiFiEspUdp . h>

# i n c l u d e < P u b S u b C l i e n t . h>

# i n c l u d e < S o f t w a r e S e r i a l . h> # i n c l u d e < Wire . h>

# i n c l u d e < L i q u i d C r y s t a l \_ I 2 C . h>

/ / U p d a t e t h e s e w i t h v a l u e s s u i t a b l e f o r y o u r n e t w o r k . # d e f i n e s s i d " Redmi "

# d e f i n e p a s s w o r d " 1 2 3 4 5 6 7 8 "

/ / k h a i bao k e n h m q t t

# d e f i n e m q t t \_ s e r v e r " m13 . c l o u d m q t t . com " # d e f i n e m q t t \_ t o p i c \_ p u b " t r a n g t h a i "

# d e f i n e m q t t \_ t o p i c \_ s u b " d i e u k h i e n " # d e f i n e m q t t \_ u s e r " ESP8266 "

# d e f i n e mqtt\_pwd " 1 2 3 4 5 6 "

c o n s t u i n t 1 6 \_ t m q t t \_ p o r t = 1 5 5 3 2 ;

/ / k h o i dong s e r i a l

W i F i E s p C l i e n t e s p C l i e n t ;

P u b S u b C l i e n t c l i e n t ( e s p C l i e n t ) ; S o f t w a r e S e r i a l s o f t ( 1 2 , 1 3 ) ;

/ / k h o i dong LCD

L i q u i d C r y s t a l \_ I 2 C l c d ( 0 x27 , 2 0 , 4 ) ;

/ / k h a i bao b i e n

i n t c a m b i e n r u n g = A0 ; i n t t t r u n g ;

i n t c a m b i e n a s = A1 ; i n t t t a s ;

i n t n u t s e r v o = 8 ; i n t n u t d e n = 9 ;

i n t nutbom = 1 0 ; i n t n u t a l l = 1 1 ; i n t s e r v o = 7 ;

i n t den = 6 ;

i n t bom1 = 5 ; i n t bom2 = 4 ;

S t r i n g m e s s a g e ;

i n t t t s e r v o = 1 ; i n t t t d e n = 1 ;

i n t t t b o m 1 = 1 ; i n t t t b o m 2 = 1 ;

/ / d i a c h i I2C c u a DS1307 c o n s t b y t e DS1307 = 0 x68 ;

/ / s o b y t e du l i e u d u o c doc t u d s 1 3 0 7 c o n s t b y t e N u m b e r O f F i e l d s = 7 ;

i n t s e c o n d , m i n u t e , h o u r , day , wday , month , y e a r ;

/ / k h a i bao cam b i e n l u u l u o n g

v o l a t i l e i n t f l o w \_ f r e q u e n c y ; / / M e a s u r e s f l o w s e n s o r p u l s e s u n s i g n e d i n t l \_ h o u r ; / / C a l c u l a t e d l i t r e s / h o u r

u n s i g n e d c h a r f l o w s e n s o r = 2 ; / / S e n s o r I n p u t u n s i g n e d l o n g c u r r e n t T i m e ;

u n s i g n e d l o n g c l o o p T i m e ;

v o l a t i l e i n t f l o w \_ f r e q u e n c y 1 ; / / M e a s u r e s f l o w s e n s o r p u l s e s u n s i g n e d i n t l \_ h o u r 1 ; / / C a l c u l a t e d l i t r e s / h o u r

u n s i g n e d c h a r f l o w s e n s o r 1 = 3 ; / / S e n s o r I n p u t u n s i g n e d l o n g c u r r e n t T i m e 1 ;

u n s i g n e d l o n g c l o o p T i m e 1 ;

v o i d f l o w ( ) / / I n t e r r u p t f u n c t i o n

{

f l o w \_ f r e q u e n c y + + ;

}

v o i d f l o w 1 ( ) / / I n t e r r u p t f u n c t i o n

{

f l o w \_ f r e q u e n c y 1 + + ;

}

v o i d s e t u p ( ) {

pinMode ( c a m b i e n r u n g , INPUT ) ; pinMode ( c a m b i e n a s , INPUT ) ;

pinMode ( n u t s e r v o , INPUT ) ; pinMode ( n u t d e n , INPUT ) ;

pinMode ( s e r v o , OUTPUT ) ; pinMode ( den , OUTPUT ) ;

pinMode ( bom1 , OUTPUT ) ; pinMode ( bom2 , OUTPUT ) ;

Wire . b e g i n ( ) ;

/ ∗ c a i d a t t h o i g i a n cho module ∗ /

/ / s e t T i m e ( 9 , 1 5 , 0 0 , 7 , 3 0 , 1 2 , 1 7 ) ; / / 0 9 : 1 5 : 0 0 T7 30−12−2017

S e r i a l . b e g i n ( 9 6 0 0 ) ;

l c d . i n i t ( ) ; / / i n i t i a l i z e t h e l c d l c d . b a c k l i g h t ( ) ;

l c d . s e t C u r s o r ( 3 , 0 ) ;

l c d . p r i n t ( " S t a r t i n g " ) ;

s e t u p \_ w i f i ( ) ;

c l i e n t . s e t S e r v e r ( m q t t \_ s e r v e r , m q t t \_ p o r t ) ; c l i e n t . s e t C a l l b a c k ( c a l l b a c k ) ;

pinMode ( f l o w s e n s o r , INPUT ) ; pinMode ( f l o w s e n s o r 1 , INPUT ) ;

d i g i t a l W r i t e ( f l o w s e n s o r , HIGH ) ; / / O p t i o n a l I n t e r n a l P u l l −Up d i g i t a l W r i t e ( f l o w s e n s o r 1 , HIGH ) ; / / O p t i o n a l I n t e r n a l P u l l −Up a t t a c h I n t e r r u p t ( 0 , f l o w , RISING ) ; / / S e t u p I n t e r r u p t

a t t a c h I n t e r r u p t ( 1 , f l o w 1 , RISING ) ; / / S e t u p I n t e r r u p t s e i ( ) ; / / E n a b l e i n t e r r u p t s

c u r r e n t T i m e = m i l l i s ( ) ; c u r r e n t T i m e 1 = m i l l i s ( ) ; c l o o p T i m e = c u r r e n t T i m e ;

c l o o p T i m e 1 = c u r r e n t T i m e 1 ;

}

v o i d s e t u p \_ w i f i ( ) { d e l a y ( 1 0 ) ;

/ / We s t a r t by c o n n e c t i n g t o a WiFi n e t w o r k

s o f t . b e g i n ( 9 6 0 0 ) ; WiFi . i n i t (& s o f t ) ; S e r i a l . p r i n t l n ( ) ;

S e r i a l . p r i n t ( " C o n n e c t i n g t o " ) ;

l c d . c l e a r ( ) ;

l c d . s e t C u r s o r ( 3 , 0 ) ;

l c d . p r i n t ( " W i f i c o n n e c t i n g " ) ;

S e r i a l . p r i n t l n ( s s i d ) ;

WiFi . b e g i n ( s s i d , p a s s w o r d ) ;

w h i l e ( WiFi . s t a t u s ( ) ! = WL\_CONNECTED)

{

d e l a y ( 5 0 0 ) ;

S e r i a l . p r i n t ( " . " ) ;

}

S e r i a l . p r i n t l n ( " " ) ;

S e r i a l . p r i n t l n ( " WiFi c o n n e c t e d " ) ; S e r i a l . p r i n t l n ( " I P a d d r e s s : " ) ;

S e r i a l . p r i n t l n ( WiFi . l o c a l I P ( ) ) ;

l c d . c l e a r ( ) ;

l c d . s e t C u r s o r ( 3 , 0 ) ;

l c d . p r i n t ( " W i f i c o n n e c t e d " ) ;

}

v o i d c a l l b a c k ( c h a r ∗ t o p i c , b y t e ∗ p a y l o a d , u n s i g n e d i n t l e n g t h )

{

S e r i a l . p r i n t ( " Message a r r i v e d [ " ) ; S e r i a l . p r i n t ( t o p i c ) ;

S e r i a l . p r i n t ( " ] " ) ;

m e s s a g e . remove ( 0 ) ;

f o r ( i n t i = 0 ; i < l e n g t h ; i ++)

{

m e s s a g e += ( c h a r ) p a y l o a d [ i ] ;

}

S e r i a l . p r i n t ( m e s s a g e ) ; S e r i a l . p r i n t l n ( ) ;

i f ( m e s s a g e == " 0 " )

{

t t s e r v o = 0 ;

d i g i t a l W r i t e ( s e r v o , HIGH ) ; d e l a y ( 1 0 0 0 ) ;

t t r u n g = a n a l o g R e a d ( c a m b i e n r u n g ) ; S e r i a l . p r i n t l n ( t t r u n g ) ;

i f ( t t r u n g < 4 0 0 )

{

S e r i a l . p r i n t ( " FEED ON " ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 0 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 1 ) ;

l c d . p r i n t ( " FEED ON " ) ;

}

d e l a y ( 2 0 0 0 ) ; t t s e r v o = 1 ;

d i g i t a l W r i t e ( s e r v o ,LOW) ; d e l a y ( 1 0 0 0 ) ;

t t r u n g = a n a l o g R e a d ( c a m b i e n r u n g ) ; S e r i a l . p r i n t l n ( t t r u n g ) ;

i f ( t t r u n g > 8 0 0 )

{

S e r i a l . p r i n t ( " FEED OFF " ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 1 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 1 ) ;

l c d . p r i n t ( " FEED OFF " ) ;

}

}

i f ( m e s s a g e == " 1 " )

{

t t s e r v o = 1 ;

d i g i t a l W r i t e ( s e r v o ,LOW) ; d e l a y ( 1 0 0 0 ) ;

t t r u n g = a n a l o g R e a d ( c a m b i e n r u n g ) ; S e r i a l . p r i n t l n ( t t r u n g ) ;

i f ( t t r u n g > 8 0 0 )

{

S e r i a l . p r i n t ( " FEED OFF " ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 1 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 1 ) ;

l c d . p r i n t ( " FEED OFF " ) ;

}

}

i f ( m e s s a g e == " 2 " )

{

t t d e n = 0 ;

d i g i t a l W r i t e ( den , HIGH ) ;

d e l a y ( 1 0 0 0 ) ;

t t a s = a n a l o g R e a d ( c a m b i e n a s ) ; S e r i a l . p r i n t l n ( t t a s ) ;

i f ( t t a s < 2 0 0 )

{

S e r i a l . p r i n t ( " LIGHT ON " ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 2 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 1 1 , 1 ) ;

l c d . p r i n t ( " LIGHT ON " ) ;

}

}

i f ( m e s s a g e == " 3 " )

{

t t d e n = 1 ;

d i g i t a l W r i t e ( den ,LOW) ; d e l a y ( 1 0 0 0 ) ;

t t a s = a n a l o g R e a d ( c a m b i e n a s ) ; S e r i a l . p r i n t l n ( t t a s ) ;

i f ( t t a s > 5 0 0 )

{

S e r i a l . p r i n t ( " LIGHT OFF " ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 3 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 1 1 , 1 ) ;

l c d . p r i n t ( " LIGHT OFF " ) ;

}

}

i f ( m e s s a g e == " 4 " )

{

d i g i t a l W r i t e ( bom1 , HIGH ) ; d i g i t a l W r i t e ( bom2 , HIGH ) ; d e l a y ( 3 0 0 0 ) ;

c u r r e n t T i m e = m i l l i s ( ) ;

/ / E v e r y s e c o n d , c a l c u l a t e and p r i n t l i t r e s / h o u r i f ( c u r r e n t T i m e >= ( c l o o p T i m e + 1 0 0 0 ) )

{

c l o o p T i m e = c u r r e n t T i m e ; / / U p d a t e s c l o o p T i m e

/ / P u l s e f r e q u e n c y ( Hz ) = 7 . 5 Q , Q i s f l o w r a t e i n L / min l \_ h o u r = ( f l o w \_ f r e q u e n c y ∗ 60 / 7 . 5 ) ;

/ / ( P u l s e f r e q u e n c y x 60 min ) / 7 . 5Q = f l o w r a t e i n L / h o u r f l o w \_ f r e q u e n c y = 0 ;

/ / R e s e t C o u n t e r

S e r i a l . p r i n t ( l \_ h o u r , DEC ) ; / / P r i n t l i t r e s / h o u r S e r i a l . p r i n t l n ( " L / h o u r " ) ;

}

c u r r e n t T i m e 1 = m i l l i s ( ) ;

i f ( c u r r e n t T i m e 1 >= ( c l o o p T i m e 1 + 1 0 0 0 ) )

{

c l o o p T i m e 1 = c u r r e n t T i m e 1 ; / / U p d a t e s c l o o p T i m e

/ / P u l s e f r e q u e n c y ( Hz ) = 7 . 5 Q , Q i s f l o w r a t e i n L / min l \_ h o u r 1 = ( f l o w \_ f r e q u e n c y 1 ∗ 60 / 7 . 5 ) ;

/ / ( P u l s e f r e q u e n c y x 60 min ) / 7 . 5Q = f l o w r a t e i n L / h o u r f l o w \_ f r e q u e n c y 1 = 0 ;

/ / R e s e t C o u n t e r

S e r i a l . p r i n t ( l \_ h o u r 1 , DEC ) ; / / P r i n t l i t r e s / h o u r S e r i a l . p r i n t l n ( " L / h o u r " ) ;

}

i f ( l \_ h o u r > 100 and l \_ h o u r 1 > 1 0 0 )

{

t t b o m 1 = 0 ; t t b o m 2 = 0 ;

S e r i a l . p r i n t ( " PUMP ON " ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 4 " ) ;

l c d . s e t C u r s o r ( 0 , 2 ) ;

l c d . p r i n t ( " PUMP ON " ) ; S e r i a l . p r i n t l n ( ) ;

}

}

i f ( m e s s a g e == " 5 " )

{

d i g i t a l W r i t e ( bom1 ,LOW) ; d i g i t a l W r i t e ( bom2 ,LOW) ;

t t b o m 1 = 1 ; t t b o m 2 = 1 ;

S e r i a l . p r i n t ( " PUMP OFF " ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 5 " ) ;

l c d . s e t C u r s o r ( 0 , 2 ) ;

l c d . p r i n t ( " PUMP OFF " ) ; S e r i a l . p r i n t l n ( ) ;

}

S e r i a l . p r i n t l n ( ) ;

}

v o i d r e c o n n e c t ( )

{

/ / Loop u n t i l we ’ r e r e c o n n e c t e d w h i l e ( ! c l i e n t . c o n n e c t e d ( ) )

{

S e r i a l . p r i n t ( " A t t e m p t i n g MQTT c o n n e c t i o n . . . " ) ; l c d . c l e a r ( ) ;

l c d . s e t C u r s o r ( 3 , 0 ) ;

l c d . p r i n t ( "MQTT c o n n e c t i n g " ) ;

/ / A t t e m p t t o c o n n e c t

i f ( c l i e n t . c o n n e c t ( " ESP " , m q t t \_ u s e r , mqtt\_pwd ) )

{

S e r i a l . p r i n t l n ( " c o n n e c t e d " ) ;

/ / Once c o n n e c t e d , p u b l i s h an a n n o u n c e m e n t . . .

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " ESP c o n n e c t e d " ) ; c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 1 " ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 3 " ) ; c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 5 " ) ;

/ / . . . and r e s u b s c r i b e

c l i e n t . s u b s c r i b e ( m q t t \_ t o p i c \_ s u b ) ;

l c d . c l e a r ( ) ;

l c d . s e t C u r s o r ( 3 , 0 ) ;

l c d . p r i n t ( "MQTT c o n n e c t e d " ) ; l c d . s e t C u r s o r ( 0 , 1 ) ;

l c d . p r i n t ( " FEED OFF " ) ; l c d . s e t C u r s o r ( 1 1 , 1 ) ;

l c d . p r i n t ( " LIGHT OFF " ) ; l c d . s e t C u r s o r ( 0 , 2 ) ;

l c d . p r i n t ( " PUMP OFF " ) ;

}

e l s e

{

S e r i a l . p r i n t ( " f a i l e d , r c = " ) ; S e r i a l . p r i n t ( c l i e n t . s t a t e ( ) ) ;

S e r i a l . p r i n t l n ( " t r y a g a i n i n 5 s e c o n d s " ) ;

/ / W a i t 5 s e c o n d s b e f o r e r e t r y i n g d e l a y ( 5 0 0 0 ) ;

}

}

}

v o i d r e a d D S 1 3 0 7 ( )

{

Wire . b e g i n T r a n s m i s s i o n ( DS1307 ) ; Wire . w r i t e ( ( b y t e ) 0 x00 ) ;

Wire . e n d T r a n s m i s s i o n ( ) ;

Wire . r e q u e s t F r o m ( DS1307 , N u m b e r O f F i e l d s ) ;

|  |  |  |  |
| --- | --- | --- | --- |
| s e c o n d  m i n u t e | =  = | b c d 2 d e c ( Wire . r e a d ( )  b c d 2 d e c ( Wire . r e a d ( ) | & 0 x 7 f ) ;  ) ; |
| h o u r | = | b c d 2 d e c ( Wire . r e a d ( ) | & 0 x 3 f ) ; / / mode 24 h . |
| wday | = | b c d 2 d e c ( Wire . r e a d ( ) | ) ; |
| day | = | b c d 2 d e c ( Wire . r e a d ( ) | ) ; |
| month | = | b c d 2 d e c ( Wire . r e a d ( ) | ) ; |
| y e a r | = | b c d 2 d e c ( Wire . r e a d ( ) | ) ; |

y e a r += 2 0 0 0 ;

}

/ / c h u y e n t u f o r m a t BCD ( B i n a r y −Coded D e c i m a l ) s a n g D e c i m a l i n t b c d 2 d e c ( b y t e num )

{

r e t u r n ( ( num / 1 6 ∗ 1 0 ) + ( num % 1 6 ) ) ;

}

/ / c h u y e n t u D e c i m a l s a n g BCD i n t d e c 2 b c d ( b y t e num )

{

r e t u r n ( ( num / 1 0 ∗ 1 6 ) + ( num % 1 0 ) ) ;

}

v o i d d i g i t a l C l o c k D i s p l a y ( ) {

/ / d i g i t a l c l o c k d i s p l a y o f t h e t i m e S e r i a l . p r i n t ( h o u r ) ;

p r i n t D i g i t s ( h o u r , 5 ) ;

p r i n t D i g i t s ( m i n u t e , 8 ) ; p r i n t D i g i t s ( s e c o n d , 1 1 ) ; S e r i a l . p r i n t ( " " ) ;

S e r i a l . p r i n t ( day ) ; S e r i a l . p r i n t ( " " ) ;

S e r i a l . p r i n t ( month ) ; S e r i a l . p r i n t ( " " ) ;

S e r i a l . p r i n t ( y e a r ) ; S e r i a l . p r i n t l n ( ) ;

}

v o i d p r i n t D i g i t s ( i n t d i g i t s , i n t x ) {

/ / c a c t h a n h p h a n t h o i g i a n d u o c n g a n c a c h b a n g dau : i f ( x > 5 )

{

S e r i a l . p r i n t ( " : " ) ;

l c d . s e t C u r s o r ( x , 3 ) ; l c d . p r i n t ( " : " ) ;

}

i f ( d i g i t s < 1 0 )

{

S e r i a l . p r i n t ( ’ 0 ’ ) ;

l c d . s e t C u r s o r ( x + 1 , 3 ) ; l c d . p r i n t ( ’ 0 ’ ) ;

l c d . s e t C u r s o r ( x + 2 , 3 ) ; l c d . p r i n t ( d i g i t s ) ;

}

e l s e

{

S e r i a l . p r i n t ( d i g i t s ) ; l c d . s e t C u r s o r ( x + 1 , 3 ) ; l c d . p r i n t ( d i g i t s ) ;

}

}

v o i d s e t T i m e ( b y t e h r , b y t e min , b y t e s e c , b y t e wd , b y t e d , b y t e mth , b y t e y r )

{

Wire . b e g i n T r a n s m i s s i o n ( DS1307 ) ;

Wire . w r i t e ( b y t e ( 0 x00 ) ) ; / / d a t l a i p o i t e r Wire . w r i t e ( d e c 2 b c d ( s e c ) ) ;

Wire . w r i t e ( d e c 2 b c d ( min ) ) ; Wire . w r i t e ( d e c 2 b c d ( h r ) ) ;

Wire . w r i t e ( d e c 2 b c d ( wd ) ) ; / / day o f week : Sunday = 1 , S a t u r d a y = 7 Wire . w r i t e ( d e c 2 b c d ( d ) ) ;

Wire . w r i t e ( d e c 2 b c d ( mth ) ) ; Wire . w r i t e ( d e c 2 b c d ( y r ) ) ; Wire . e n d T r a n s m i s s i o n ( ) ;

}

v o i d l o o p ( )

{

i n t n u t s e r v o S t a t u s = d i g i t a l R e a d ( n u t s e r v o ) ; i f ( n u t s e r v o S t a t u s == HIGH )

{

i f ( t t s e r v o == 1 )

{

t t s e r v o = 0 ;

S e r i a l . p r i n t ( " FEED ON " ) ; d i g i t a l W r i t e ( s e r v o , HIGH ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 0 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 1 ) ;

l c d . p r i n t ( " FEED ON " ) ;

d e l a y ( 4 0 0 0 ) ; t t s e r v o = 1 ;

S e r i a l . p r i n t ( " FEED OFF " ) ; d i g i t a l W r i t e ( s e r v o ,LOW) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 1 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 1 ) ;

l c d . p r i n t ( " FEED OFF " ) ;

}

e l s e

{

t t s e r v o = 1 ;

S e r i a l . p r i n t ( " FEED OFF " ) ; d i g i t a l W r i t e ( s e r v o ,LOW) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 1 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 1 ) ;

l c d . p r i n t ( " FEED OFF " ) ;

}

}

i n t n u t d e n S t a t u s = d i g i t a l R e a d ( n u t d e n ) ; i f ( n u t d e n S t a t u s == HIGH )

{

i f ( t t d e n == 1 )

{

t t d e n = 0 ;

S e r i a l . p r i n t ( " LIGHT ON " ) ; d i g i t a l W r i t e ( den , HIGH ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 2 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 1 1 , 1 ) ;

l c d . p r i n t ( " LIGHT ON " ) ;

}

e l s e

{

t t d e n = 1 ;

S e r i a l . p r i n t ( " LIGHT OFF " ) ; d i g i t a l W r i t e ( den ,LOW) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 3 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 1 1 , 1 ) ;

l c d . p r i n t ( " LIGHT OFF " ) ;

}

}

i n t n u t b o m S t a t u s = d i g i t a l R e a d ( nutbom ) ; i f ( n u t b o m S t a t u s == HIGH )

{

i f ( t t b o m 1 == 1 and t t b o m 2 == 1)

{

t t b o m 1 = 0 ; t t b o m 2 = 0 ;

S e r i a l . p r i n t ( " PUMP ON " ) ; d i g i t a l W r i t e ( bom1 , HIGH ) ;

d i g i t a l W r i t e ( bom2 , HIGH ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 4 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 2 ) ;

l c d . p r i n t ( " PUMP ON " ) ;

}

e l s e

{

t t b o m 1 = 1 ; t t b o m 2 = 1 ;

S e r i a l . p r i n t ( " PUMP OFF " ) ; d i g i t a l W r i t e ( bom1 ,LOW) ;

d i g i t a l W r i t e ( bom2 ,LOW) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 5 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 2 ) ;

l c d . p r i n t ( " PUMP OFF " ) ;

}

}

i n t n u t a l l S t a t u s = d i g i t a l R e a d ( n u t a l l ) ; i f ( n u t a l l S t a t u s == HIGH )

{

i f ( t t b o m 1 == 1 and t t b o m 2 == 1 and t t s e r v o == 1 and t t d e n == 1 )

{

t t b o m 1 = 0 ; t t b o m 2 = 0 ;

t t s e r v o = 0 ; t t d e n = 0 ;

S e r i a l . p r i n t ( " ALL ON " ) ; d i g i t a l W r i t e ( bom1 , HIGH ) ; d i g i t a l W r i t e ( bom2 , HIGH ) ; d i g i t a l W r i t e ( s e r v o , HIGH ) ; d i g i t a l W r i t e ( den , HIGH ) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 4 " ) ; c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 0 " ) ; c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 2 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 2 ) ;

l c d . p r i n t ( " PUMP ON " ) ; l c d . s e t C u r s o r ( 1 1 , 1 ) ;

l c d . p r i n t ( " LIGHT ON " ) ; l c d . s e t C u r s o r ( 0 , 1 ) ;

l c d . p r i n t ( " FEED ON " ) ;

d e l a y ( 4 0 0 0 ) ; t t s e r v o = 1 ;

S e r i a l . p r i n t ( " FEED OFF " ) ; d i g i t a l W r i t e ( s e r v o ,LOW) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 1 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 1 ) ;

l c d . p r i n t ( " FEED OFF " ) ;

}

e l s e

{

t t b o m 1 = 1 ; t t b o m 2 = 1 ; t t d e n = 1 ;

t t s e r v o = 1 ;

S e r i a l . p r i n t ( " ALL OFF " ) ; d i g i t a l W r i t e ( bom1 ,LOW) ; d i g i t a l W r i t e ( bom2 ,LOW) ; d i g i t a l W r i t e ( s e r v o ,LOW) ; d i g i t a l W r i t e ( den ,LOW) ;

c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 5 " ) ; c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 1 " ) ; c l i e n t . p u b l i s h ( m q t t \_ t o p i c \_ p u b , " 3 " ) ; S e r i a l . p r i n t l n ( ) ;

l c d . s e t C u r s o r ( 0 , 1 ) ;

l c d . p r i n t ( " FEED OFF " ) ; l c d . s e t C u r s o r ( 0 , 2 ) ;

l c d . p r i n t ( " PUMP OFF " ) ; l c d . s e t C u r s o r ( 1 1 , 1 ) ;

l c d . p r i n t ( " LIGHT OFF " ) ;

}

}

/ / doc du l i e u c u a DS1307 r e a d D S 1 3 0 7 ( ) ;

/ / h i e n t h i t h o i g i a n r a S e r i a l m o n i t o r d i g i t a l C l o c k D i s p l a y ( ) ;

d e l a y ( 1 0 0 0 ) ;

i f ( ! c l i e n t . c o n n e c t e d ( ) )

{

r e c o n n e c t ( ) ;

}

c l i e n t . l o o p ( ) ;

}