**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**NGÀNH: CÔNG NGHỆ THÔNG TIN**

***Đề tài: GIÁM SÁT GIẤC NGỦ EM BÉ THÔNG QUA THIẾT BỊ IOT***

Sinh viên thực hiện: NGUYỄN THÀNH NHÂN

MSSV: 15066141

ĐẶNG VÕ ĐỨC TRỌNG

MSSV: 15055721

Giảng viên hướng dẫn: TS. Trần Thị Minh Khoa

**Tp Hồ Chí Minh, 3/2019**

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*(phần này GVHD ghi rõ ý thức trách nhiệm của sinh viên trong thời gian thực hiện đồ án, kết quả đạt được của đề tài: ưu điểm, tồn tại. GVHD ghi rõ ý kiến của mình về sinh viên có đủ điều kiện được bảo vệ hay không).*

……………………………………………………………………………………….….……………………………………………………………………………………….……………………………………………………………………………………….……………………………………………………………………………………….……………………………………………………………………………………….……………………………………………………………………………………….……………………………………………………………………………………….……………………………………………………………………………………….……………………………………………………………………………………….……………………………………………………………………………………….……………………………………………………………………………………

**Cán bộ chấm 1** *(ghi rõ họ và tên):*

**Cán bộ chấm 2** *(ghi rõ họ và tên):*

**Thư ký:**

**Kết quả điểm** *(Báo cáo TTTN):*

………………………………………………………………………….…………….….………………………………………………………...……………………………….……………………………………………………………………………………….……………………………………………………………………………………….……………………………………………………………………………………….…………………………………………………………………………….………….………………………………………………………………

*Tp HCM, ngày ...... tháng ...... năm 20......* **CHỦ TỊCH HỘI ĐỒNG**

**MỤC LỤC**

**DANH SÁCH HÌNH VẼ**

**DANH SÁCH BẢNG BIỂU**

**LỜI MỞ ĐẦU**

**LỜI CẢM ƠN**

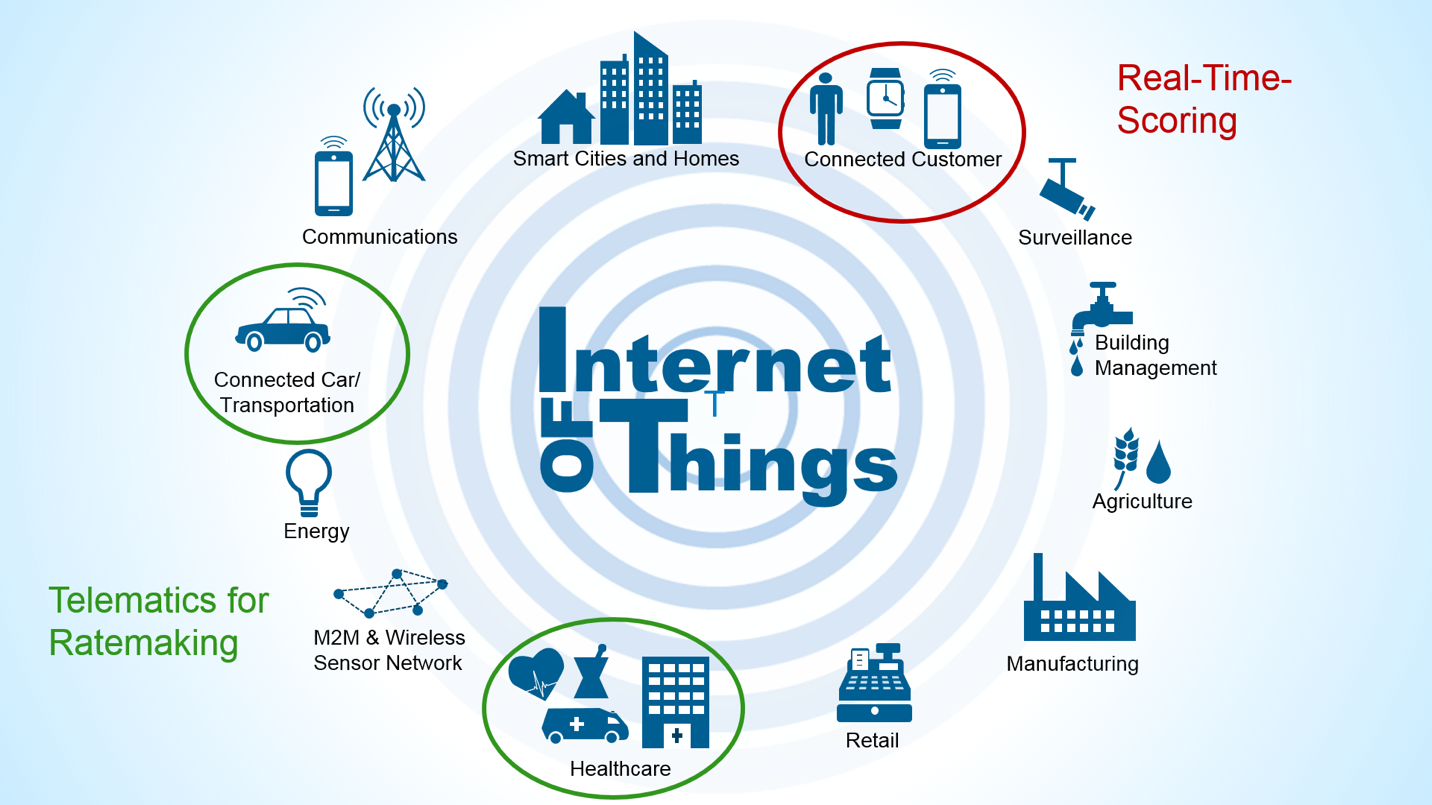
Trên thực tế không có sự thành công nào mà không gắn liền với những sự hỗ trợ, giúp đỡ dù ít hay nhiều, dù trực tiếp hay gián tiếp từ người khác. Trong suốt thời gian từ khi bắt đầu học tập ở giảng đường đại học đến nay, em đã nhận được rất nhiều sự quan tâm , giúp đỡ của quý Thầy cô, gia đình và bạn bè.

Với lòng biết ơn sâu sắc nhất, em xin gửi đến quý Thầy cô ở Khoa Công nghệ thông tin – Đại học Công Nghiệp TPHCM đã cùng với tri thức và tâm huyết của mình để truyền đạt vôn kiến thức quý báu cho chúng em trong suốt thời gian học tập tại trường. Và đặc biệt, trong học kỳ vừa qua, Khoa đã tổ chức cho chúng em được tiếp cận với môn học mà theo em là rất hữu ích đối với cá nhân em. Đó là môn học “Công nghệ mới”.

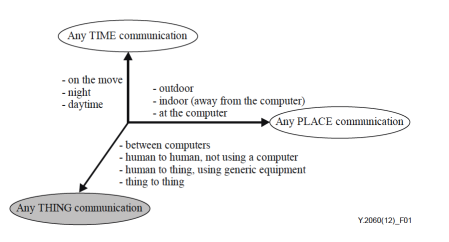
Em xin chân thành cảm ơn TS. đã tận tâm hướng dẫn chúng em qua từng buổi học trên lớp cũng như những buổi nói chuyện, thảo luận về lĩnh vực IoT. Nếu không có những lời hướng dẫn, dạy bảo của thầy thì em nghĩ bản báo cáo này của em rất khó có thể hoàn thiện được. Một lần nữa, em xin chân thành cảm ơn thầy.

Bước đầu đi vào thực tế, tìm hiếu về lĩnh vực còn khá mới, kiến thức của em còn hạn chế và còn nhiều bỡ ngỡ. Do vậy, không tránh khỏi những thiếu sót là điều chắc chắn, em rất mong nhận được những ý kiến đóng góp quý báu của quý Thầy Cô và các bạn học cùng lớp để kiến thức của em trong lĩnh vực này được hoàn thiện hơn.

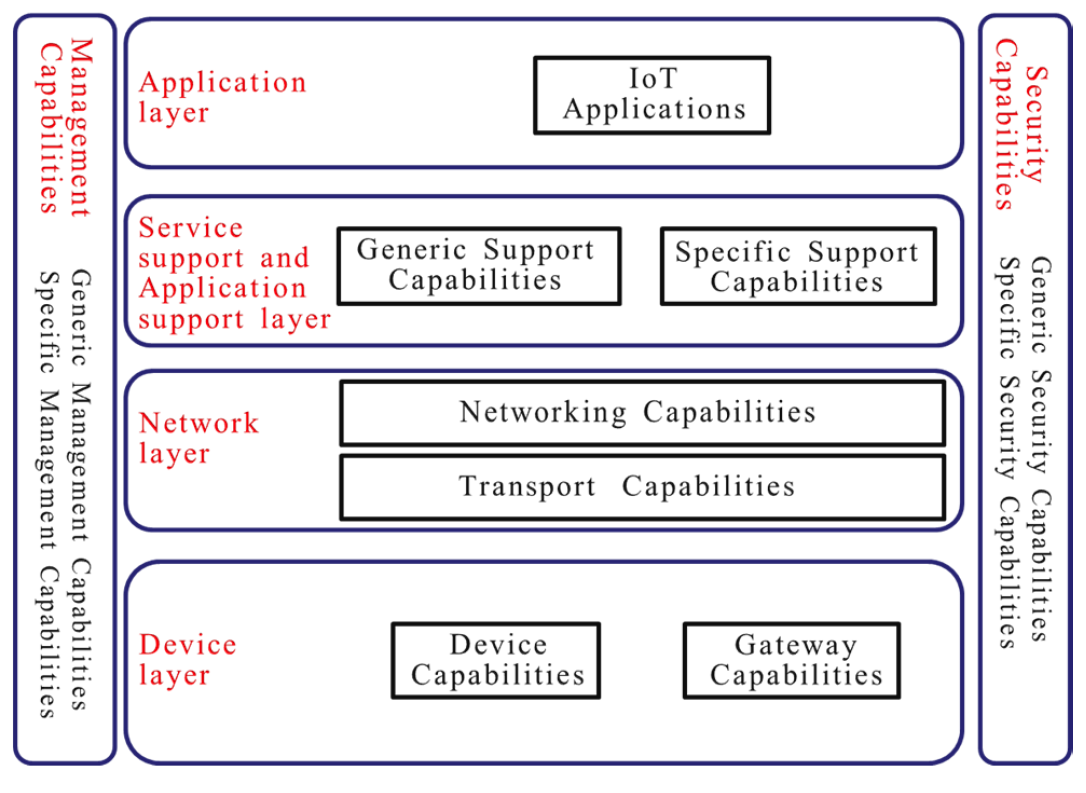
CHƯƠNG 1: TỔNG QUAN VỀ IOT1.1. Định nghĩa- Thiết bị(devices): Đối với Internet Of Things, đây là một phần của cả hệ thống với chức năng bắt buộc là communication và chức năng không bắt buộc là: cảm biến, thực thi,thu thập dữ liệu, lưu trữ và xử lý dữ liệu.  
- Internet Of Things: Là một cơ sở hạ tầng mang tính toàn cầu cho xã hội thông tin, mang đến những dịch vụ tiên tiến bằng cách kết nối các “Things” (cả physical lẫn virtual) dựa trên sự tồn tại của thông tin, dựa trên khả năng tương tác của các thông tin đó, và dựa trên các công nghệ truyền thông.  
- Things: Đối với Internet Of Things, “Thing” là một đối tượng của thế giới vật chất (physical things) hay thế giới thông tin ảo(virtual things). “Things” có khả năng được nhận diện, và “Things” có thể được tích hợp vào trong mạng lưới thông tin liên lạc.



1.2. Khái niệm của IoTIoT có thể được coi là một tầm nhìn sâu rộng của công nghệ và cuộc sống. Từ  
quan điểm của tiêu chuẩn kỹ thuật, IoT có thể được xem như là một cơ sở hạ tầng  
mang tính toàn cầu cho xã hội thông tin, tạo điều kiện cho các dịch vụ tiên tiến thông qua sự liên kết các “Things”. IoT dự kiến sẽ tích hợp rất nhiều công nghệ mới, chẳng hạn như các công nghệ thông tin machine-to-machine, mạng tự trị, khai thác dữ liệu và ra quyết định, bảo vệ sự an ninh và sự riêng tư, điện toán đám mây. Như hình dưới, một hệ thống thông tin trước đây đã mang đến 2 chiều – “Any TIME” và “Any PLACE” communication. Giờ IoT đã tạo thêm một chiều mới trong hệ thống thông tin đó là “Any THING” Communication.



1.3 IOT từ góc nhìn kỹ thuậtNhư đề cập ở mục 1, “Things” trong IoT có thể là đối tượng vật lý (Physical)  
hoặc là đối tượng thông tin (hay còn gọi là đối tượng ảo – Virtual). Hai loại đối tượng này có thể ánh xạ (mapping) qua lại lẫn nhau. Một đối tượng vật lý có thể được trình bày hay đại diện bởi một đối tượng thông tin, tuy nhiên một đối tượng thông tin có thể tồn tại mà không nhất thiết phải được ánh xạ từ một đối tượng vật lý nào.  
Yêu cầu tối thiểu của các “device” trong IOT là khả năng giao tiếp. Devices sẽ  
được phân loại vào các dạng như device mang thông tin, device thu thập dữ liệu,  
device cảm nhận(sensor), device thực thi, hay general device:  
1.4. Đặc điểm cơ bản và yêu cầu ở mức high-level của một hệ thống IOT1.4.1 Đặc tính cơ bản– Tính kết nối liên thông(interconnectivity).  
– Những dịch vụ liên quan đến “Things”.  
– Tính không đồng nhất.  
– Thay đổi linh hoạt.  
– Quy mô lớn.  
1.4.2 Yêu cầu ở mức high-level đối với một hệ thống IOTMột hệ thống IOT phải thoả mãn các yêu cầu sau:  
– Kết nối dựa trên sự nhận diện.  
– Khả năng cộng tác.  
– Khả năng tự quản của network.  
– Dịch vụ thoả thuận.  
– Các Khả năng dựa vào vị trí(location-based capabilities).  
– Bảo mật.  
– Bảo vệ tính riêng tư.  
– Plug and play.  
– Khả năng quản lý.  
1.5 Mô hình của một hệ thống IOTBất kỳ một hệ thống IOT nào cũng được xây dựng lên từ sự kết hợp của 4  
layer sau:  
– Lớp ứng dụng (Application Layer)  
– Lớp Hỗ trợ dịch vụ và hỗ trợ ứng dụng (Service support and application  
support layer)  
– Lớp mạng (Network Layer)  
– Lớp thiết bị (Device Layer)



### 2.1. Lớp ứng dụng

Lớp ứng dụng (Application Layer) cũng tương tự như trong mô hình OSI 7 lớp, lớp này tương tác trực tiếp với người dùng để cung cấp một chức năng hay một dịch vụ cụ thể của một hệ thống IOT.

### 2.2. Lớp hỗ trợ dịch vụ và hỗ trợ ứng dụng

Các dịch vụ hỗ trợ (Service support and application support layer) gồm hai nhóm khả khả năng sau:

        Nhóm dịch vụ chung: Các dịch vụ hỗ trợ chung, phổ biến mà hầu hết các ứng dụng IOT đều cần, ví dụ như xử lý dữ liệu hoặc lưu trữ dữ liệu.

        Nhóm dịch vụ cụ thể, riêng biệt: Những ứng dụng IOT khác nhau sẽ có nhóm dịch phụ hỗ trợ khác nhau và đặc thù. Trong thực tế, ví dụ trong SmartFarming, nhóm dịch vụ cụ thể riêng biệt là tính toán độ tăng trưởng của cây mà đưa ra quyết định tưới nước hoặc bón phân.

### 2.3. Lớp mạng

Lớp mạng (Network Layer) có 2 chức năng:

        Chức năng Networking: cung cấp chức năng điều khiển các kết nối kết nối mạng, chẳng hạn như tiếp cận được nguồn tài nguyên thông tin và chuyển tài nguyên đó đến nơi cần thiết, hay chứng thực, uỷ quyền…

        Chức năng Transporting: tập trung vào việc cung cấp kết nối cho việc truyền thông tin của dịch vụ/ứng dụng IOT.

### 2.4. Lớp thiết bị

Lớp thiết bị (Device Layer) chính là các phần cứng vật lý trong hệ thống IOT. Device có thể phân thành hai loại như sau:

        Thiết bị thông thường: Device này sẽ tương tác trực tiếp với network: Các thiết bị có khả năng thu thập và tải lên thông tin trực tiếp (nghĩa là không phải sử dụng gateway) và có thể trực tiếp nhận thông tin (ví dụ, lệnh) từ các network. Device này cũng có thể tương tác gián tiếp với network: Các thiết bị có thể thu thập và tải network gián tiếp thông qua khả năng gateway. Ngược lại, các thiết bị có thể gián tiếp nhận thông tin (ví dụ, lệnh) từ network. Trong thực tế, các thiết bị thông thường bao gồm các cảm biến, các phần cứng điều khiển motor, đèn,…

        Thiết bị Gateway: Gateway là cổng liên lạc giữa device và network. Một Gateway hỗ trợ 2 chức năng sau:

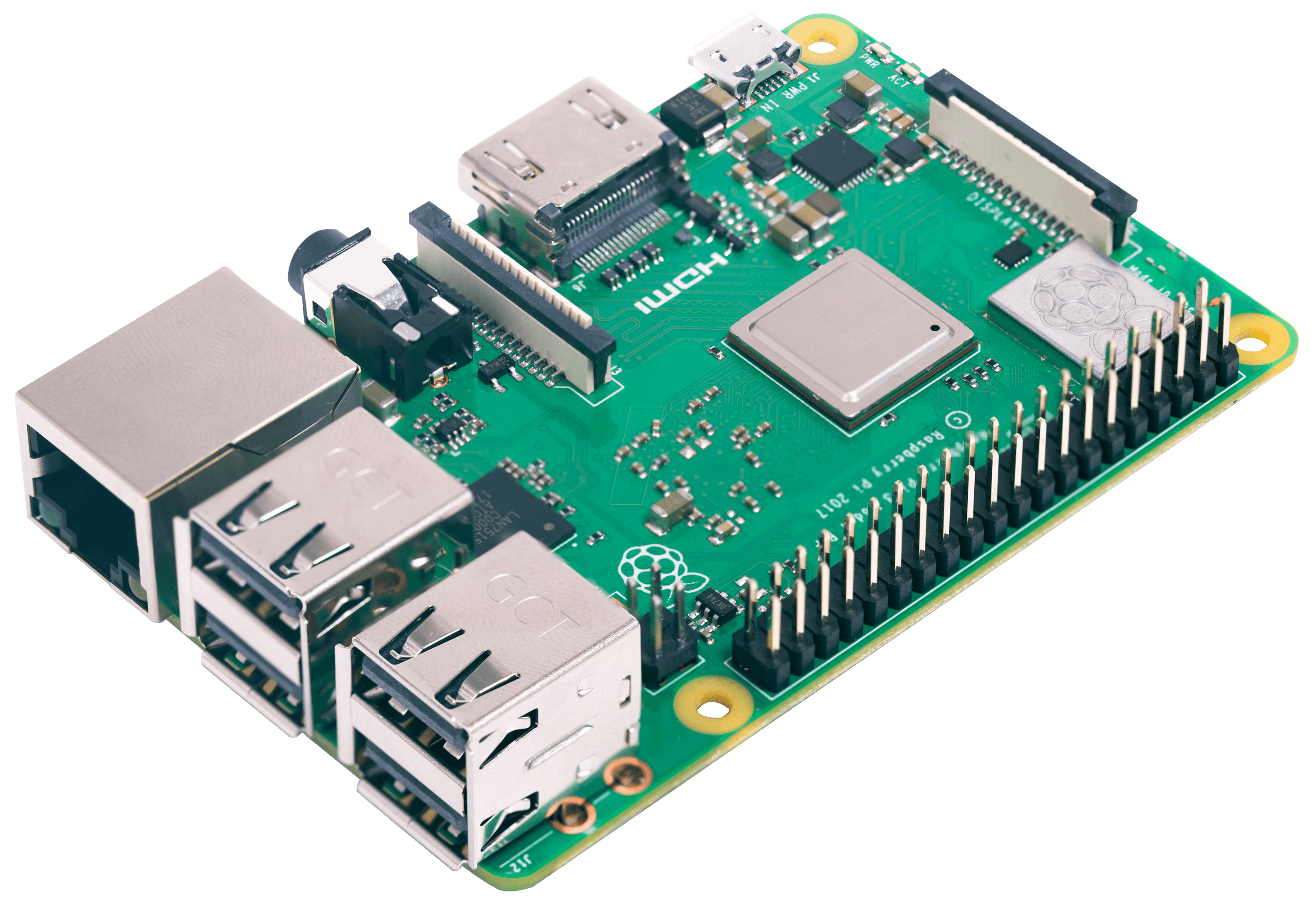
-         Có nhiều chuẩn giao tiếp: Vì các Things khác nhau có kiểu kết nối khác nhau, nên Gateway phải hỗ trợ đa dạng từ có dây đến không dây, chẳng hạn CAN bus, ZigBee, Bluetooth hoặc Wi-Fi. Tại Network layer, gateway có thể giao tiếp thông qua các công nghệ khác nhau như PSTN, mạng 2G và 3G, LTE, Ethernet hay DSL.

-         Chức năng chuyển đổi giao thức: Chức năng này cần thiết trong hai tình huống là khi truyền thông ở lớp Device, nhiều device khác nhau sử dụng giao thức khác nhau, ví dụ, ZigBee với Bluetooth, và là khi truyền thông giữa các Device và Network, device dùng giao thức khác, network dùng giao thức khác, ví dụ, device dùng ZigBee còn tầng network thì lại dùng công nghệ 3G. Trong thực tế, Gateway có thể được build từ các board như Raspberry Pi hay Arduino, hoặc Gateway được sản xuất công nghiệp bởi các tập đoàn lớn như Intel hay Texas Instrument.

CHƯƠNG 2: MÔ HÌNH ĐỒ ÁN

CHƯƠNG 4: XÂY DỰNG ỨNG DỤNG GIÁM SÁT GIẤC NGỦ ĐỨA TRẺ

4.1 Tìm hiểu về thiết bị Raspberry Pi4.1.1 Giới thiệu chungRaspberry Pi là một chiếc máy tính tí hon chạy hệ điều hành Linux ra mắt  
vào tháng 2 năm 2012 với giá chỉ $25. Ban đầu Raspberry Pi được phát triển  
dựa trên ý tưởng tiến sĩ Eben Upton tại đại học Cambridge muốn tạo ra một  
chiếc máy tính giá rẻ để học sinh có thể dễ dàng tiếp cận và khám phá thế giới  
tin học. Dự định khiêm tốn của ông đến cuối đời là có thể bán được tổng cộng  
1000 bo mạch cho các trường học. Tuy nhiên với những ưu điểm nổi bật, hơn  
một triệu board Raspberry Pi đã được bán ra chỉ trong vòng chưa đầy một năm.



Chỉ cần 1 bàn phím, 1 tivi hoặc 1 màn hình có cổng HDMI/DVI, 1 nguồn USB 5V và 1 dây micro USB là đã có thể sử dụng Raspberry Pi như 1 máy tính bình thường. Với Raspberry Pi, ta có thể sử dụng các ứng dụng văn phòng, nghe nhạc, xem phim độ nét cao... Một điều quan trọng là nó rất tiết kiệm điện và khả năng chạy liên tục 24/24.

Phần CứngThông số kỹ thuật chi tiết:

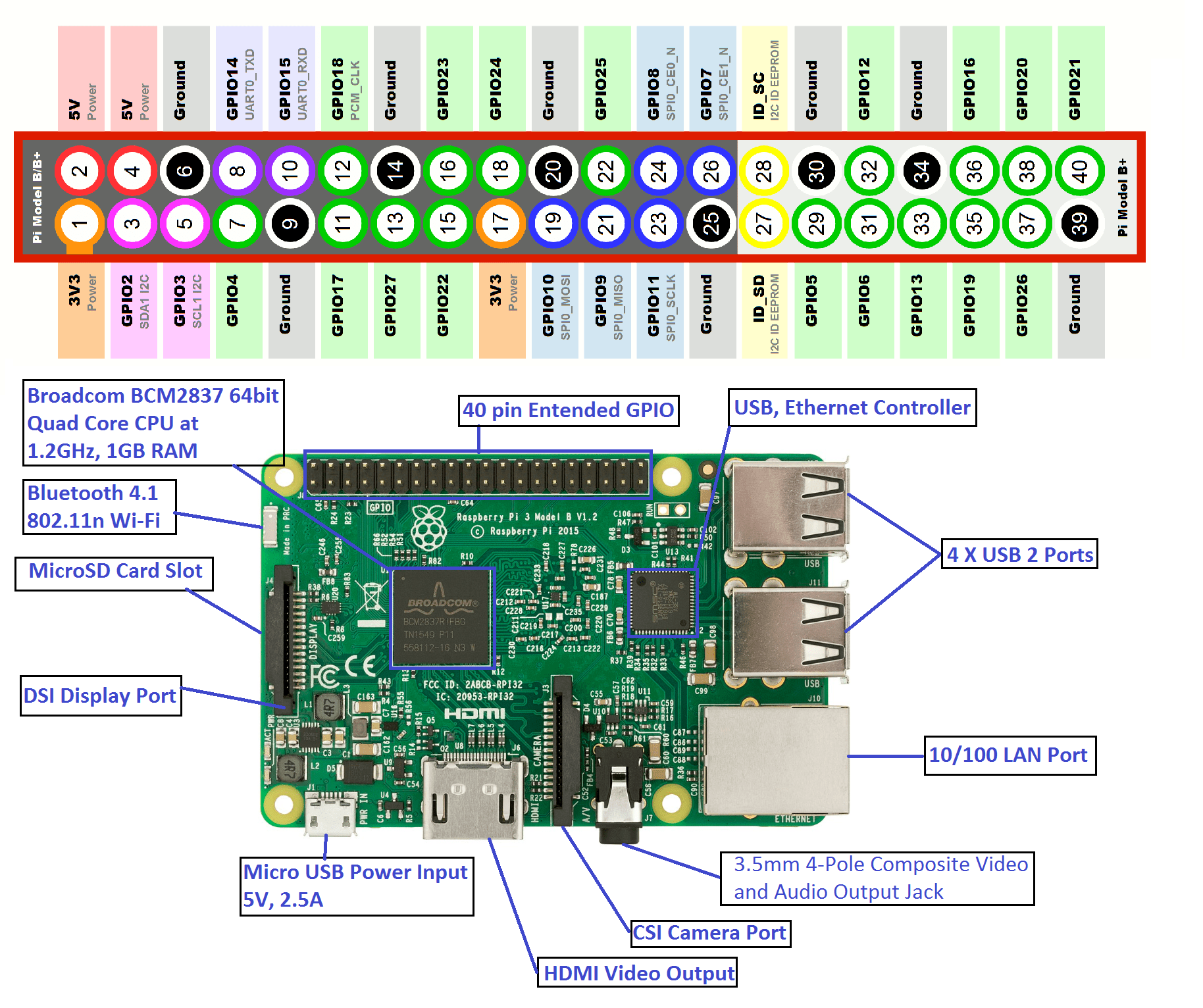
|  |  |
| --- | --- |
| Vi xử lý | Broadcom BCM2837B0, quad-core A53 (ARMv8) 64-bit SoC @1.4GHz |
| RAM | 1GB LPDDR2 SDRAM |
| Kết nối | 2.4GHz and 5GHz IEEE 802.11 b/g/n/ac wireless LAN, Bluetooth 4.2, BLE, Gigabit Ethernet over USB 2.0 (Tối đa 300Mbps) |
| Cổng USB | 4 x 2.0 |
| Mở rộng | 40-pin GPIO |
| Video và âm thanh | 1 cổng full-sized HDMI, Cổng MIPI DSI Display, cổng MIPI CSI Camera, cổng stereo output và composite video 4 chân |
| Multimedia | H.264, MPEG-4 decode (1080p30), H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics |
| Lưu trữ | MicroSD |
| Nguồn điện sử dụng | 5V/2.5A DC cổng microUSB, 5V DC trên chân GPIO, Power over Ethernet (PoE)  (yêu cầu thêm PoE HAT) |

Raspberry Pi 3 Model B+ là sản phẩm mới nhất trong gia đình Raspberry Pi, nổi bật với chip 4 nhân 64-bit có tốc độ 1.4GHz – nhanh nhất từ trước đến nay! Phiên bản mới còn hỗ trợ Wifi Dual-band 2.4GHz và 5GHz, Bluetooth 4.2/Bluetooth Low Energy, cổng Ethernet tốc độ cao (300Mbps) và Power over Ethernet (PoE) thông qua PoE HAT.

Trái tim của Raspberry Pi là chip SoC (System-On-Chip) Broadcom  
BCM2835 chạy ở tốc độ 700mHz. Chip này tương đương với nhiều loại được sử  
dụng trong smartphone phổ thông hiện nay, và có thể chạy được hệ điều hành Linux.  
Tích hợp trên chip này là nhân đồ họa (GPU) Broadcom VideoCore IV. GPU này đủ mạnh để có thể chơi 1 số game phổ thông và phát video chuẩn full HD.  
Hệ thống GPIO (General Purpose Input Output): gồm 26 chân chia làm hai hang. Đúng như tên gọi của nó, từ đây ta có thể kết nối và điều khiển rất nhiều thiết bị điện tử/cơ khí khác.  
Ngõ HDMI: dùng để kết nối Pi với màn hình máy tính hay tivi có hỗ trợ cổng HDMI.  
Ngõ RCA Video (analog): khi thiết kế Pi người ta cũng tính đến trường hợp người sử dụng ở các nước đang phát triển không có điều kiện sắm một chiếc tivi đời mới tích hợp cổng HDMI. Vì vậy cổng video analog này được thêm vào, giúp Raspberry Pi có thể kết nối với chiếc tivi đời cũ.  
Ngõ audio 3.5mm: kết nối dễ dàng với loa ngoài hay headphone. Đối với  
tivi có cổng HDMI, ngõ âm thanh được tích hợp theo đường tín hiệu HDMI nên  
không cần sử dụng ngõ audio này.

Cổng CSI: khe cắm này là để cắm modul camera vào Raspberry Pi. Khi sản  
xuất Raspberry Pi thì nhà sản xuất còn sản xuất thêm một modul camera 5MP  
nhưng người mua không được hỗ trợ mà phải mua thêm. Chúng ta có thể chụp  
hình, quay phim, ... làm việc tất cả các tác vụ như trên một camera bình thường.  
Cổng DSI: nơi đây sẽ giúp ta có thể kết nối Raspberry Pi với màn hình  
cảm ứng để hiển thị và sử dụng Raspberry một cách trực quan nhất. Chúng ta có  
thể thực hiện các tác vụ tương đương như khi sử dụng chuột và bàn phím.  
Cổng USB: một điểm mạnh nữa của Raspberry Pi là tích hợp 2 cổng USB  
2.0. Ta có thể kết nối với bàn phím, chuột hay webcam, bộ thu GPS… qua đó  
có thể mở rộng phạm vi ứng dụng. Vì Raspberry Pi chạy Linux nên hầu hết thiết bị chỉ cần cắm-và-chạy (Plug-and-Play) mà không cần cài driver phức tạp.  
Cổng Ethernet: cho phép kết nối Internet dễ dàng. Cắm dây mạng vào Pi, kết nối với màn hình máy tính hay tivi và bàn phím, chuột là có thể lướt web dễ dàng.  
Khe cắm thẻ SD: Raspberry Pi không tích hợp ổ cứng. Thay vào đó nó dùng thẻ SD để lưu trữ dữ liệu. Toàn bộ hệ điều hành Linux sẽ hoạt động trên thẻ SD này vì vậy nó cần kích thước thẻ nhớ tối thiểu 4 GB và dung lượng hỗ trợ tối đa là 32 GB.

Sơ đồ chân của thiết bị Raspberry Pi 3:



*Cấu trúc phần mềm*Các Raspberry Pi sử dụng hệ điều hành dựa trên nền tảng Linux. Phần  
cứng GPU được truy cập thông qua Image Firmware được nạp vào GPU vào  
lúc khởi động từ thẻ SD. Image Firmware được gọi là đốm màu nhị phân  
(Binary Blob), trong khi ARM liên kết với mã trình điều khiển Linux ban đầu  
được dựa vào nguồn đóng. Một phần của mã điều khiển đã được giải phóng, tuy  
nhiên nhiều chương trình điều khiển thực tế được thực hiện bằng cách sử dụng  
mã nguồn đóng GPU. Phần mềm ứng dụng sử dụng các cuộc gọi đến thư viện  
thời gian chạy nguồn đóng (OpenMax, OpenGL ES hay OpenVG). Nó sẽ gọi  
một trình điều khiển nguồn mở bên trong lõi Linux, sau đó gọi mã điều khiển  
nguồn đóng GPU VideoCore IV. Các API của trình điều khiển lõi là cụ thể cho  
những thư viện đóng. Các ứng dụng Video sử dụng OpenMax, ứng dụng 3D sử dụng OpenGL ES và ứng dụng 2D sử dụng OpenVG và cả hai lần lượt sử dụng EGL. OpenMax và EGL sử dụng trình điều khiển nền tảng mã nguồn mở.



Nhà sản xuất Raspberry sẽ cung cấp một tập hợp các thư viện mã nguồn  
đóng cho phép chúng ta truy cập vào các tính năng tăng tốc GPU. Các thư viện  
sẽ có sẵn là:  
⮚ OpenGL ES 2.0 (opengl) là một thư viện 3D, rất thường được sử dụng trên  
máy tính để bàn và các hệ thống nhúng. Nó được định nghĩa bởi Khronos Group.  
⮚ OpenVG là một thư viện bản vẽ véc tơ 2D, cũng thường được sử dụng  
trên máy tính để bàn và các hệ thống nhúng. Một lần nữa, được định nghĩa bởi  
Khronos Group.  
⮚ EGL là một giao diện lập trình ứng dụng giữa Khronos và API như  
OpenGL ES hay OpenVG và hệ thống cửa sổ nền tảng nguồn gốc cơ bản.  
⮚ Openmax cung cấp một tập hợp các API với khái niệm trừu tượng của  
người dùng cho những thói quen sử dụng trong âm thanh, video, vàxử lý hình  
ảnh tĩnh. OpenMax định nghĩa ba lớp, đây là lớp IL, cung cấp một giao diện giữa các khuôn khổ đa phương tiện như Gstreamer và một tập hợp các thành  
phần đa phương tiện (như bảng mã).  
⮚ Openmax IL không có một API chuẩn ở giai đoạn này, vì vậy đó là  
một cài đặt tùy chỉnh. Tất cả các thư viện này được cung cấp bởi chip SoC  
Broadcom.  
4.1.2 Tìm hiểu hệ điều hành  
Raspberry Pi là một máy tính, để máy tính này hoạt động cần cài đặt hệ  
điều hành. Trong thế giới nguồn mở linux, có rất nhiều phiên bản hệ điều hành  
tùy biến (distro) khác nhau. Tùy theo nhu cầu và mục đích, cũng như khả năng  
học hỏi mà ta sẽ sử dụng distro phù hợp với mình.  
Có 5 phiên bản hệ điều hành được cung cấp chính thức cho Raspberry Pi:  
⮚ Raspian "wheezy": đây là distro dựa trên Debian wheezy, sử dụng  
hard-float ABI (tính toán dấu chấm động bằng phần cứng) cho thời gian chạy  
các ứng dụng nhanh hơn. Có sẵn giao diện đồ họa. Phù hợp với người mới bắt  
đầu tiếp cận Linux vì tính dễ sử dụng và trực quan.  
⮚ Soft-float "wheezy": vẫn được xây dựng dựa trên Debian wheezy  
nhưng việc xử lý dấu chấm động được thực hiện bằng phần mềm. Việc này giúp  
có thể sử dụng máy ảo Java (Oracle JVM) trên Raspberry.  
⮚ Arch Linux: phiên bản giành cho ARM. Đảm bảo thời gian khởi động trong vòng 10 giây. Chỉ khởi động và load các gói cần thiết. Để sử dụng được  
Arch Linux cần có kiến thức cơ bản về Linux.  
⮚ Pidora: là phiên bản của Fedora được tối ưu cho Raspberry Pi, có sẵn  
giao diện đồ họa. Giành cho những ai đã quen xài Fedora.  
⮚ RISC OS: là hệ điều hành do nhóm phát triển ARM thiết kế riêng. Đây  
không phải là một phiên bản Linux, do vậy cần làm quen với cấu trúc và câu  
lệnh đặc trưng cho hệ điều hành này.  
4.2 Xây dựng mạng cảm biến giám sát các thông số môi trƣờng qua  
việc sử dụng thiết bị Raspberry Pi và các Sensor  
Chúng ta sẽ xây dựng một ứng dụng để giám sát các thông số môi trường  
tiêu biểu đối với việc bảo quản thực phẩm, đặc biệt là thực phẩm tươi sống

4.2. Cảm biến

4.2.1. Cảm biến chuyển động:

Ứng dụng

- Dùng trong cho an ninh: phát hiện người trong khoảng cách cho phép.

- Dùng để điều khiển tự động các thiết bị trong nhà khi có người: đèn điện, cửa…

Giới thiệu về HC – SR501

Điện áp hoạt động : DC  4.5V – 20V.

Đầu ra : High 3.3V/ Low 0V.

Có 2 chế độ hoạt động:

+ L không lặp lại kích hoạt.

+ H lặp lại kích hoạt.

Thời gian trễ: 5 – 200s có thể điều chỉnh từ 0,xx đến hàng chục giây.

Thời gian khóa: 2.5s (mặc định).

Góc quét  <  100 độ.

Kích thước ống cảm biến: 23mm (mặc định).

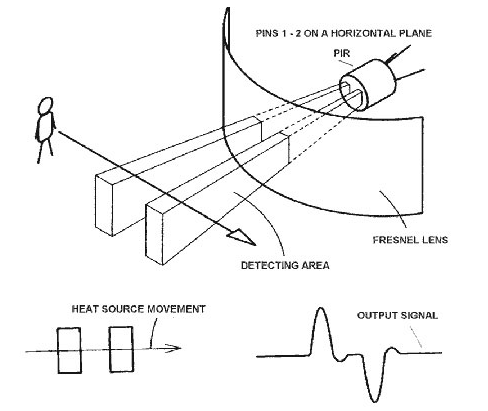
Nhiệt độ hoạt động: -15oC -  70oC.

Kích thước board:  32mm\*24mm.

Khối lượng: 6g.

Nguyên tắc hoạt động

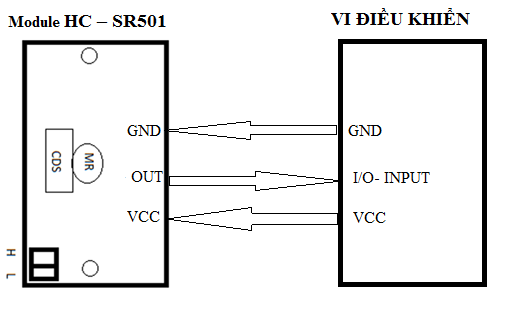
Cơ chế hoạt động của cảm biến hồng ngoại PIR :  là cảm biến thu tia hồng ngoại được phát ra từ các vật thể phát ra tia hồng ngoại như thân thể con người(hay nguồn nhiệt bất kì).



Các cảm biến PIR luôn có sensor (mắt cảm biến) với 2 đơn vị (element). Chắn trước mắt sensor là một lăng kính (thường làm bằng plastic), chế tạo theo kiểu lăng kính fresnel. Lăng kính fresnel này có tác dụng chặn lại và phân thành nhiều vùng (zone) cho phép tia hồng ngoại đi vào mắt sensor. 2 đơn vị của mắt sensor có tác dụng phân thành 2 điện cực. Một cái là điện cực dương (+) và cái kia là âm (-). Khi 2 đơn vị này được tuần tự kích hoạt (cái này xong rồi mới đến cái kia) thì sẽ sinh ra một xung điện, xung điện này kích hoạt sensor.

Kiểm soát ánh sáng (tùy chọn): Bạn có thể lắp thêm quang trở, khi có quang trở, sẽ thiết lặp module hoạt động ban ngày hoặc ban đêm.

Sơ đồ kết nối



Chân VCC : nguồn hoạt động của cảm biến cấpvào từ  4.5V đến  20V.

Chân OUT :  Output kết nối với chân I/O của vi điều khiển hoặc relay. Khi cho tín hiệu:

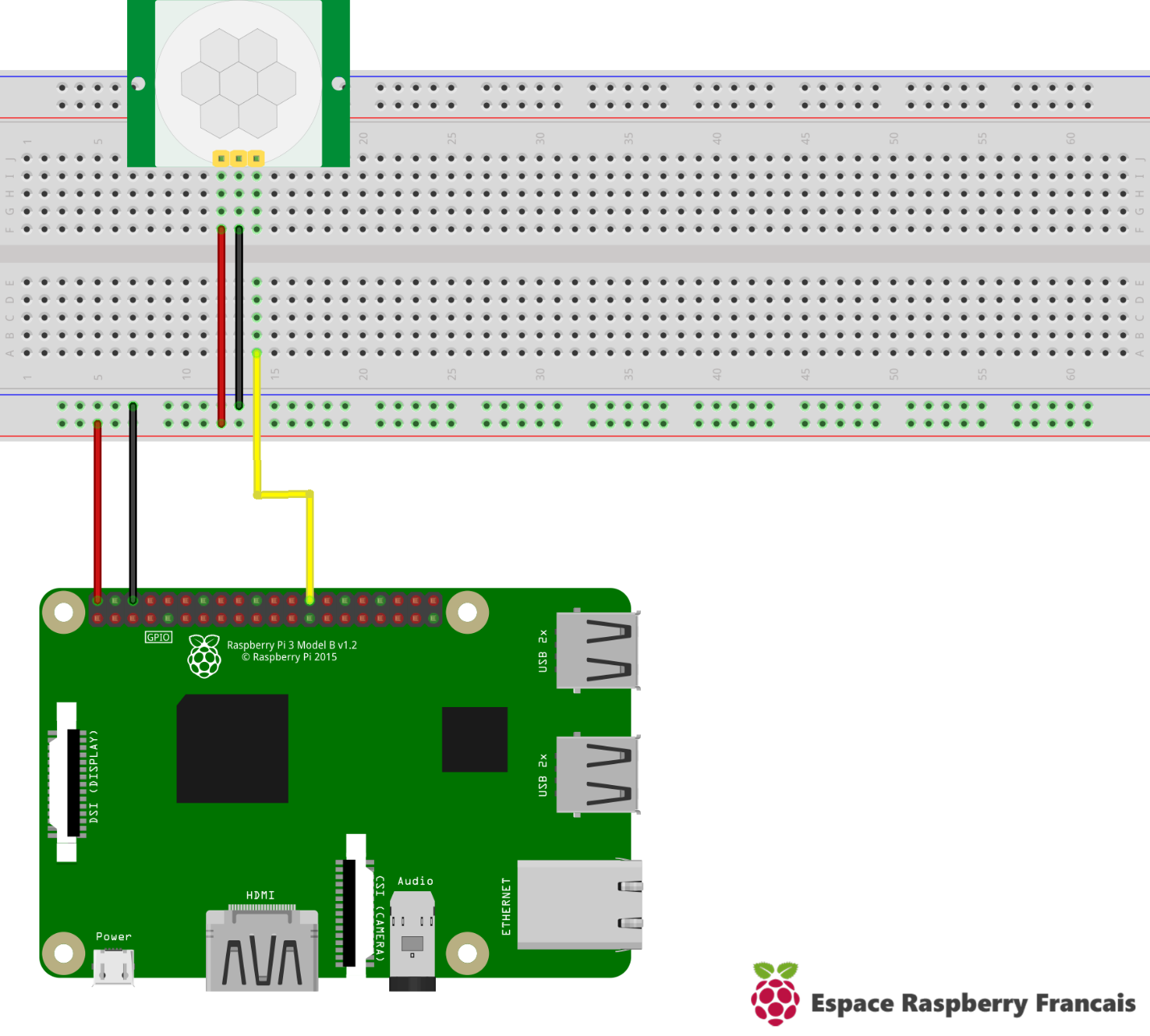
+ 3,3V có vật thể chuyển động qua.

+ 0V không có vật thể qua.

Chân GND : chân đất nối GND.

Chế độ H: Điện áp ra V\_out tự động giữ nguyên 3.3V cho đến khi không còn chuyển động.

Chế độ L: Điện áp ra V\_out tự động chuyển về 0 khi hết thời gian trễ.



Lưu ý sử dụng

Cài đặt: Khi khởi tạo, module cần thời gian khởi tạo khoảng 1 phút. Trong thời gian này, moudle tạo ra điện áp cao từ 1-3 lần sau đó vào chế độ chờ.

Điện áp ra 1.5-3.3V, nếu sử dụng I/O 4.5-5.5V bạn nên lắp thêm tran.

Nên cố gắng tránh ánh sáng trực tiếp và nguồn nhiễu gần với bề mặt lăng kính của các module, để tránh đưa ra tín hiệu nhiễu, tránh sử dụng môi trường nhiều gió.

4.2.2. Cảm biến âm thanh:

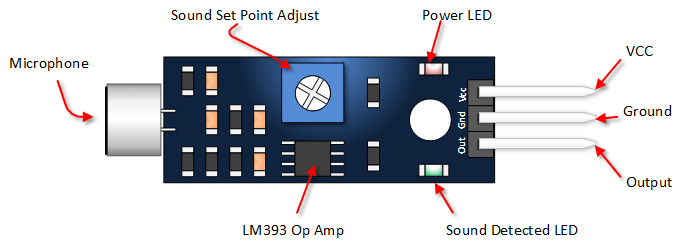
Cảm biến âm thanh giúp nhận biết và phát hiện cường độ âm thanh của môi trường xung quanh. Thích hợp để làm các ứng dụng cơ bản về nhận biết, giám sát các tiếng động của môi trường xung quanh, và nhiều ứng dụng thú vị khác. Độ nhạy có thể được điều chỉnh bằng chiết áp trên module để phù hợp với từng nhu cầu ứng dụng.

Nó sử dụng một microphone, mạch khuếch đại tín hiệu, máy dò biên độ và bộ đệm. Khi phát hiện âm thanh nó sẽ qua mạch khuếch đại để khuếch đại tín hiệu rồi gửi tín hiệu digital qua chân “OUT”.

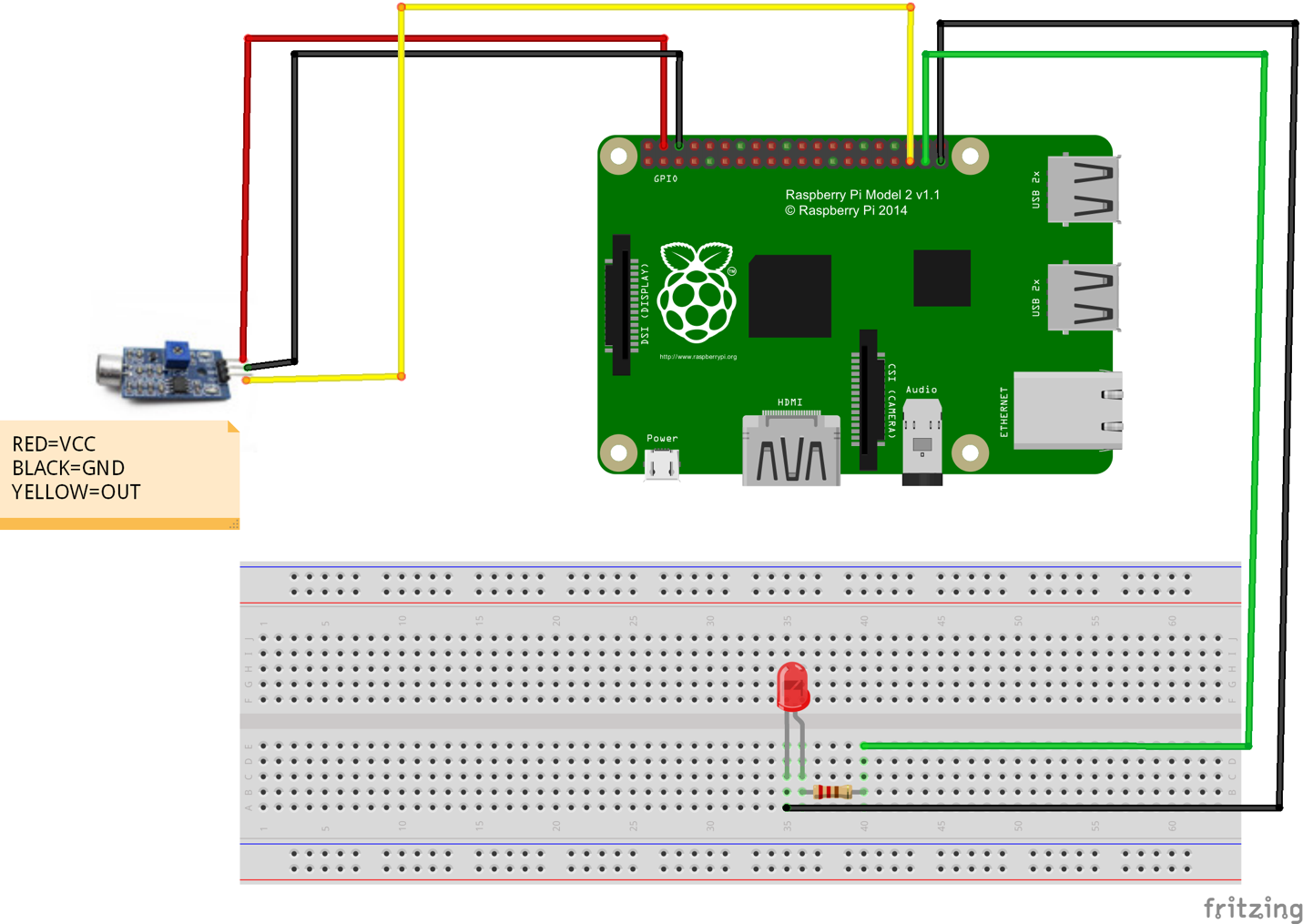
**Thông số kỹ thuật**

Điện áp hoạt động: DC 4-6V

* Xuất ra kiểu tín hiệu: digital (0 hoặc 1)
* Kích thước: 38mm \* 16mm \* 9mm



|  |  |
| --- | --- |
| VCC | Nguồn 4V đến 6V |
| GND | Mass |
| OUT | Đầu ra tín hiệu số (mức cao hoặc mức thấp) |



4.3. Các thiết bị khác:

4.3.1. Thiết bị chuyển động điều khiển đồ chơi trẻ em:

Động cơ RC Servo 9G có kích thước nhỏ, là loại được sử dụng nhiều nhất để làm các mô hình nhỏ hoặc các cơ cấu kéo không cần đến lực nặng. Động cơ RC Servo 9G có tốc độ phản ứng nhanh, các bánh răng được làm bằng nhựa nên cần lưu ý khi nâng tải nặng vì có thể làm hư bánh răng, động cơ RC Servo 9G có tích hợp sẵn Driver điều khiển động cơ bên trong nên có thể dễ dàng điều khiển góc quay bằng phương pháp điều độ rộng xung PWM.

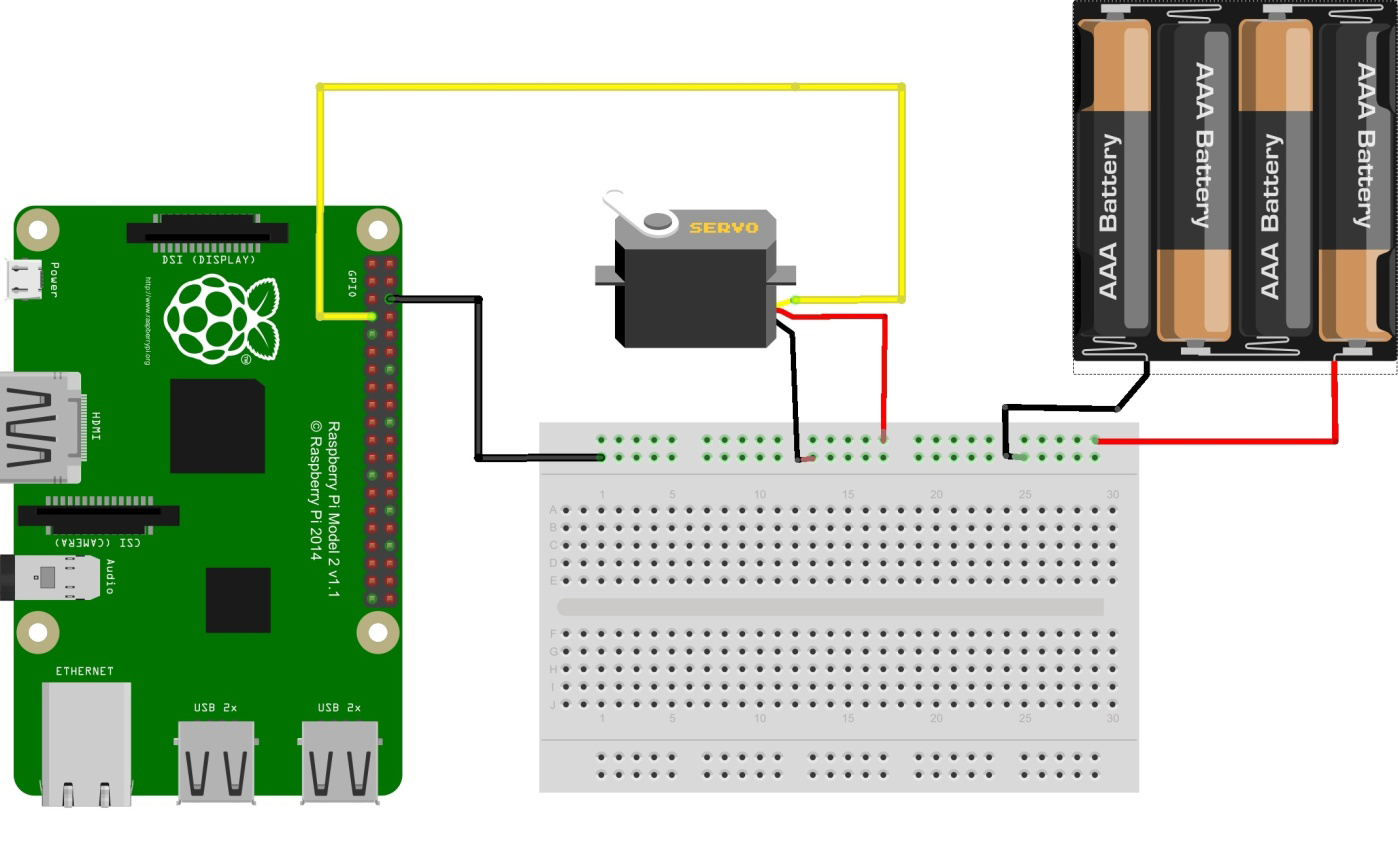
Servo là một dạng động cơ điện đặc biệt. Không giống như động cơ thông thường cứ cắm điện vào là quay liên tục, servo chỉ quay khi được điều khiển (bằng [xung PPM](http://arduino.vn/reference/xung-ppm)) với góc quay nằm trong khoảng bất kì từ 0o -180o. Mỗi loại servo có kích thước, khối lượng và cấu tạo khác nhau. Có loại thì nặng chỉ 9g (chủ yếu dùng trên máy bay mô mình), có loại thì sở hữu một momen lực bá đạo (vài chục Newton/m), hoặc có loại thì khỏe và nhông sắc chắc chắn,...

Động cơ servo được thiết kế những hệ thống hồi tiếp vòng kín. Tín hiệu ra của động cơ được nối với một mạch điều khiển. Khi động cơ quay, vận tốc và vị trí sẽ được hồi tiếp về mạch điều khiển này. Nếu có bầt kỳ lý do nào ngăn cản chuyển động quay của động cơ, cơ cấu hồi tiếp sẽ nhận thấy tín hiệu ra chưa đạt được vị trí mong muốn. Mạch điều khiển tiếp tục chỉnh sai lệch cho động cơ đạt được điểm chính xác. Các động cơ servo điều khiển bằng liên lạc vô tuyến được gọi là động cơ servo RC (radio-controlled). Trong thực tế, bản thân động cơ servo không phải được điều khiển bằng vô tuyến, nó chỉ nối với máy thu vô tuyến trên máy bay hay xe hơi. Động cơ servo nhận tín hiệu từ máy thu này.

**Thông số kỹ thuật:**

* Điện áp hoạt động: 4.8-5VDC
* Tốc độ: 0.12 sec/ 60 degrees (4.8VDC)
* Lực kéo: 1.6KG.CM
* Kích thước: 21x12x22mm
* Trọng lượng: 9g.





4.3.2. Camera Raspberry:

Camera Raspberry Pi V1 5MP là Version đầu tiên của module camera cho Raspberry Pi với cảm biến OV5647 độ phân giải 5MP, sử dụng tương thích với tất cả các dòng Raspberry Pi từ trước đến nay, chất lượng hình ảnh tốt, độ phân giải cao và có khả năng quay phim ở chất lượng HD.  
  
Để sử dụng Module Camera cho Raspberry Pi tốt và bền Hshop.vn có tặng cho Quý Khách khi mua hàng vỏ Case bằng Mica để chống tĩnh điện từ tay người làm hư cảm biến và bảo vệ mạch.  
  
**Thông số kỹ thuật:**

* Module Camera V1 cho Raspberry Pi.
* Cảm biến: OV5647
* Độ phân giải: 5MP
* Độ phân giải hình: 2592x1944 pixel.
* Quay phim HD 1080P 30, 720P 60, VGA 640x480P 60.
* Lens: Fixed Focus.
* Conector: Ribon conector.
* Kích thước: 25x24x9mm



4.3.3. Loa phát nhạc:

Lựa chọn loa có jack cắm chuẩn 3.5mm có thể tương thích với các loại desktop, laptop, smartphone, máy tính bảng, máy nghe nhạc...

Loa được tích hợp sẵn volume thuận tiện sử dụng, lấy nguồn điện trực tiếp từ cổng usb máy tính hoặc các bộ adapter sạc usb. Cách sử dụng vô cùng đơn giản, chỉ cần cắm cáp vào là sử dụng ngay mà không cần cài đặt gì.

- Tích hợp với máy tính để bàn, laptop, điện thoại, tai nghe, usb

- Giao tiếp qua cổng USB phổ biến và jack cắm tiêu chuẩn 3.5mm giúp bạn dễ dàng kết nối với nhiều dòng máy nghe nhạc có cổng tương thích.



## Tổng quan về ASP.NET MVC

### Giới thiệu về ASP.NET

Ta phải công nhận một điều là .NET Framework và các ứng dụng của nó đã và đang tạo một cuộc cách mạng kỹ thuật trong các công nghệ tin học (Information Technology), thay đổi tận gốc rễ các kiểu mẫu lập trình hay phét triển và triển khai mạng trên thế giới và do đó tạo một vận hội mới đáp ứng mọi yêu cầu khẩn thiết cho các ngành nghề kỹ thuật và thương mại hiện nay cũng như vạch một hướng đi vững chắc và lâu dài cho tương lai tin học. ASP.NET chính là một trong những ứng dụng quan trọng nhất để phát triển và triển khai mạng một cách dễ dàng chưa từng thấy … từ xưa đến nay. Hãy lắng nghe chính Microsoft nói về ASP.NET như thế nào:

“ASP.NET là một framework lập trình mang tính cách mạng cho phép phát triển nhanh chóng các ứng dụng và dịch vụ web mạnh mẽ. Là một phần của Microsoft .NET Platform, nó cung cấp cách dễ nhất và dễ mở rộng nhất để phát triển, triển khai và chạy các ứng dụng web phân tán có thể nhắm mục tiêu bất kỳ trình duyệt hoặc ứng dụng nào.”

Chúng ta cùng tìm hiểu những đặc tính chung của ASP.NET:

* Easy Programming Model: ASP.NET giúp chúng ta phát triển và triển khai các ứng dụng về mạng trong một thời gian kỉ lục vì nó cung cấp cho ta một kiểu mẫu lập trình dễ dàng và gọn gàng nhất. Ngoài ra, các trang ASP.NET còn làm việc với mọi trình duyệt hiện nay như Chrome, CocCoc, Firefox, Internet Explorer (IE), Netscape, Opera, AOL…mà không cần đổi tới đổi lui các nguồn mã vật vã như trước.
* Flexible Language Options: Không như ASP kiểu cổ điển chỉ giới hạn với VBScripts and Java Scripts, ASP.NET yểm trợ trên 25 ngôn ngữ lập trình (dĩ nhiên ngoài các ngôn ngữ mới thiết lập đã cài sẵn yểm trợ .NET framework như là VB.NET, C# và JScript.NET, ngoài ra còn có Perl.NET, Component Pascal.NET, Mercury.NET, Oberon.NET, Python.NET…).
* Great Tool Support: Mặc dù ta có thể chỉ dùng đến Notepad để triển khai các trang ASP.NET nhưng Visual Studio.NET giúp năng suất triển khai mạng thêm phần hiệu quả vì ta có thể quan sát các kế hoạch của ta dễ dàng hơn khi phác họa các thành phần của ASP.NET bằng hình ảnh với ASP.NET Web Form Shay Services theo phương pháp “drag-drop-double-click” quen thuộc của nền Windows. Hơn nữa, còn yểm trợ ta trong việc phát hiện và loại bỏ các lỗi sai một cách rất thuận lợi trong khi phát triển các ứng dụng về mạng (support for debugging and deploying ASP.NET Web applications).
* Rich Class Framework: Nhờ nền tảng vững vàng và tài nguyên phong phú .NET Framework với hơn 5000 classes bao gồm đủ thứ như XML, data access, file upload, regular expressions, transactions, message queuing, SMTP mail, … nên việc thiết kế các đặc tính trong một ứng dụng trở nên nhẹ nhàng và thoải mái hơn xưa rất nhiều.
* Compile execution: ASP.NET không những chạy nhanh hơn ASP cổ điển gấp 5 lần mà còn có thể duy trì kiểu mẫu cập nhật gọi là kiểu mẫu “just hit save”, nghĩa là ASP.NET tự động dò tìm mọi sự thay đổi và biên dịch khi cần thiết cũng như lưu trữ kết quả compile đó để cung ứng dịch vụ cho những yêu cầu tiếp theo sau, nhờ vậy ứng dụng của bạn luôn được cập nhật hóa và chạy với tốc độ nhanh.
* Rich output caching: ASP.NET có khả năng lưu trữ một kết quả chung trong bộ phận memory của trang để gửi giải đáp cho cùng một yêu cầu từ nhiều khách hàng khác nhau và nhờ đó không những tiết kiệm được sự lặp đi lặp lại công tác thi hành của một trang web mà còn gia tăng hiệu suất một cách ngoạn mục do giới hạn tối đa việc chất vấn các cơ sở dữ liệu (eliminating the need to query the database on every request) rất tốn nhiều thời gian.
* .NET Outperforms J2EE: Trong việc đối đầu nhau về hiệu suất (performance) và scalability với cùng một ứng dụng phát triển giữa Sun's Java Pet Store J2EE và ASP.NET thì ASP.NET không những nhanh vượt trội hơn J2EE đến 28 lần.
* Memory Leak, Dead Lock và Crash protection: ASP.NET cũng có khả năng dò tìm và phục hồi (detects and recovers) những trở ngại nghiêm trọng như deadlocks hay bộ nhớ bị rỉ để đảm bảo ứng dụng của bạn luôn sẵn sàng khi dùng mà không làm cản trở việc cung ứng dịch vụ cần thiết thường lệ.
* Simple application deployment: ASP.NET đơn giản hóa việc triển khai ứng dụng mạng, do đó biến việc triển khai toàn bộ ứng dụng trở nên dễ dàng và thuận tiện hơn hẳn trước kia vì bây giờ ta chỉ cần sao (với XCOPY) và lưu trữ ở server chứ không cần phải chạy chương trình “regsrv32” để đăng ký bất cứ thành phần nào cả, và thêm nữa khi cần lưu trữ những yếu tố phụ cần thiết cho việc thiết lập hay bố trí các ứng dụng, ta chỉ cần lưu trữ nó vào trong một tệp dưới dạng XML là đủ.
* Dynamic update of running application: ASP.NET cho phép ta tự động cập nhật hóa các thành phần đã compiled (compiled components) mà không cần phải khởi động lại các Web server.
* Easy Migration Path: Ta không cần phải du nhập những ứng dụng được phát triển và triển khai bằng ASP cổ điển hiện có vào ASP.NET vì ASP.NET có thể chạy song song với ASP ở cùng một Internet Information Server (IIS) trong nền windows 2000 hay windows XP. Các ứng dụng cũ vẫn được chạy hết sức thoải mái với ASP.DLL trong khi ASP.NET sẽ xử lý các ứng dụng mới. Ngoài ra, ASP.NET còn cho phép bạn dùng lại những thành phần thương mại hiện nay kiểu COM cổ điển trong các ứng dụng của nó.
* XML Web Services: Dịch vụ về mạng với XML cũng cho phép bạn truyền đạt (communicate) và chia sẻ các dữ kiện xuyên qua mạng Internet dễ dàng hơn các SOAP client mà không hề phân biệt đối xử các hệ điều hành hay ngôn ngữ lập trình khác nhau (regardless of OS or programming language). Nhờ đó, ta không cần phải học thêm hay đào sâu các kiến thức về Networking, XML hay SOAP, ...
* Mobile Web Service Support: Thêm nữa, ASP.NET Mobile Controls còn giúp ta phát triển và triển khai mạng nhắm vào thị trường những cell phone hay PDA với gần hơn 80 Mobile Web Services được cung cấp trong .NET framework. Bạn chỉ cần lập trình cho ứng dụng của bạn rồi phó mặc cho Mobile Controls đó tự động phát sinh ra những nguồn mã WAP/WML, HTML hay iMode thích hợp với từng loại thiết bị riêng biệt.

### ASP là gì?

Trước hết, họ tên của ASP.NET là Active Server Pages .NET (.NET dây là .NET framework). Nói đơn giản và ngắn gọn thì ASP.NET là một công nghệ có tính cách mạng dùng để phát triển các ứng dụng về mạng hiện nay cũng như trong tương lai (ASP.NET is a revolutionary technology for developing web applications). Lưu ý ở chỗ ASP.NET là một phương pháp tổ chức hay khung tổ chức (framework) để thiết lập các ứng dụng hết sức hùng mạnh cho mạng dựa trên CLR (Common Language Runtime) chứ không phải là một ngôn ngữ lập trình. Ngôn ngữ lập trình được dùng để diễn đạt ASP.NET là VB.NET (Visual Basic .NET) và VB.NET chỉ là một trong 25 ngôn ngữ .NET hiện nay được dùng để phát triển các trang ASP.NET mà thôi.

Tuy mang họ tên gần giống như ASP cổ điển nhưng ASP.NET không phải là ASP. Ta sơ lược ở đây một vài khác biệt giữa ASP.NET và ASP để ta có một khái niệm tổng quát.

Sự khác nhau giữa ASP.NET và ASP

ASP.NET được phác thảo lại từ số không, nó được thay đổi tận gốc rễ và phát triển phù hợp với yêu cầu hiện nay cũng như vạch một hướng đi vững chắc cho tương lai tin học. Lý do chính là vì Microsoft đã quá chán nản trong việc thêm thắt và kết hợp những công cụ mới vào các kiểu mẫu lập trình hay thiết kế mạng theo kiểu cổ điển nên Microsoft nghĩ rằng tốt nhất là nên làm một kiểu mẫu hoàn toàn mới thay vì vá víu chỗ này chỗ nọ vào ASP. Đó là chưa kể tới nhiều phát minh mới ra đời sau này dựa trên những khái niệm mới mẻ theo xu hướng phát triển hiện nay của công nghệ tin học (Information Technology) cần được đưa vào kiểu mẫu phát triển mới đó. ASP.NET cung cấp một phương pháp hoàn toàn khác biệt với ASP.

1. Restful API sử dụng python flask:

Thiết kế REST hay RESTful API (State Representational State Transfer) được thiết kế để tận dụng các giao thức hiện có. Mặc dù REST có thể được sử dụng trên hầu hết các giao thức, nhưng thường sử dụng HTTP khi được sử dụng cho các API Web. Điều này có nghĩa là các nhà phát triển không cần cài đặt thư viện hoặc phần mềm bổ sung để tận dụng thiết kế REST API. Thiết kế API REST được định nghĩa bởi Tiến sĩ Roy Fielding trong luận án tiến sĩ năm 2000 của ông. Điều đáng chú ý là lớp linh hoạt không thể tin được của nó. Vì dữ liệu không gắn với các phương pháp và tài nguyên, REST có khả năng xử lý nhiều loại cuộc gọi, trả lại các định dạng dữ liệu khác nhau và thậm chí thay đổi cấu trúc với việc thực hiện đúng của hypermedia.  
Sự tự do và sự linh hoạt của nó trong thiết kế API REST cho phép bạn xây dựng một API đáp ứng nhu cầu của bạn đồng thời đáp ứng được nhu cầu của rất nhiều khách hàng khác nhau. Không giống như SOAP, REST không bị giới hạn bởi XML, mà có thể trả về XML, JSON, YAML hoặc bất kỳ định dạng nào tùy thuộc vào yêu cầu của khách hàng. Và không giống như RPC, người dùng không bắt buộc phải biết tên thủ tục hoặc các thông số cụ thể theo thứ tự cụ thể.

Flask là một web frameworks, nó thuộc loại micro-framework được xây dựng bằng ngôn ngữ lập trình Python. Flask cho phép bạn xây dựng các ứng dụng web từ đơn giản tới phức tạp. Nó có thể xây dựng các api nhỏ, ứng dụng web chẳng hạn như các trang web, blog, trang wiki hoặc một website dựa theo thời gian hay thậm chí là một trang web thương mại. Flask cung cấp cho bạn công cụ, các thư viện và các công nghệ hỗ trợ bạn làm những công việc trên.

Flask là một micro-framework. Điều này có nghĩa Flask là một môi trường độc lập, ít sử dụng các thư viện khác bên ngoài. Do vậy, Flask có ưu điểm là nhẹ, có rất ít lỗi do ít bị phụ thuộc cũng như dễ dàng phát hiện và xử lý các lỗi bảo mật.

 Giờ để tạo một REST API đơn giản sử dụng python, chúng ta cần những thứ sau:

Python

Flask

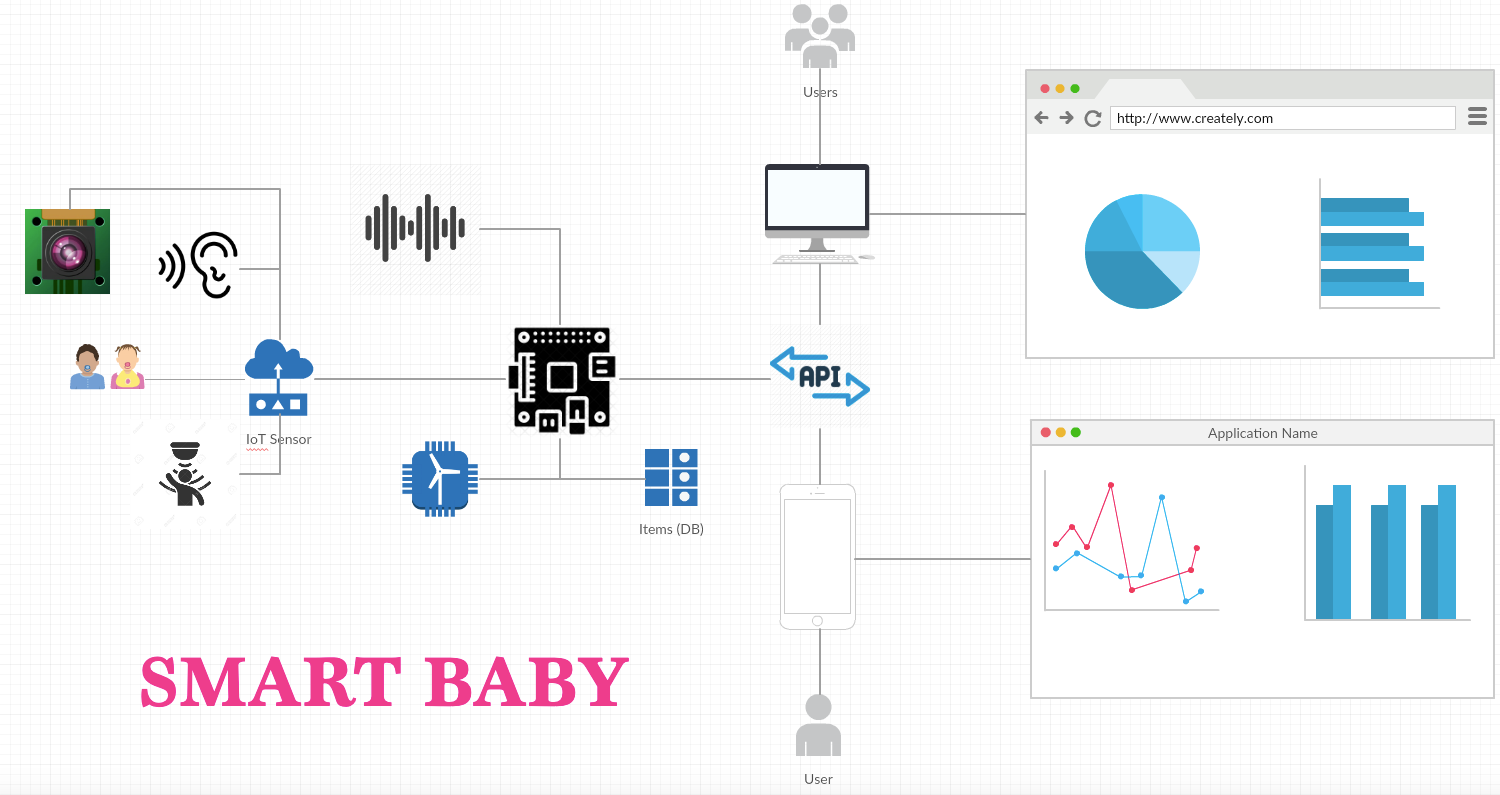
Flask-SQLAlchemy

Flask-Restful

SQlite3

Jsonify

4.4. Mô hình sử dụng trong đồ án:



KẾT LUẬNCông nghệ IoT nói chung và mạng cảm biến không dây nói riêng hứa hẹn  
tạo ra những ứng dụng đầy tiềm năng, có thể áp dụng trong nhiều lĩnh vực khác  
nhau, mà đối với các công nghệ khác còn nhiều hạn chế. Tuy nhiên để triển  
khai mạng người thiết kế hệ thống yêu cầu phải nắm bắt được những nhân tố  
tác động đến mạng, những nhược điểm của mạng cần phải được khắc phục. Tức  
là, người thiết kế cần phải quan tâm đến các tham số mạng, ví dụ như tập các  
chất lượng dịch vụ QoS. Nhờ quá trình mô phỏng người thiết kế hệ thống có thể  
đánh giá được chất lượng dịch vụ mạng cung cấp, để từ đó có thể thiết kế hệ  
thống theo cách tối ưu nhất.  
Trên cơ sở nghiên cứu tổng quan về IoT, công nghệ cảm biến không dây  
và các ứng dụng trong thực tiễn, em đã xây dựng thành công một chương  
trình thực nghiệm có tính khả thi cao (như đã trình bày trong phần đánh giá kết  
quả chạy thử nghiệm). Tuy nhiên, do thời gian và số lượng các cảm biến hạn  
chế, chương trình chưa chạy thử nghiệm với một số lượng lớn các thiết bị cảm  
biến, vì vậy chưa đánh giá hết được một số vấn đề như: việc truyền nhận dữ  
liệu từ thiết bị cảm biến đến các node mạng cảm biến, vấn đề xung đột dữ  
liệu… Đây cũng là một trong những hướng nghiên cứu, phát triển tiếp theo của  
luận văn.

TÀI LIỆU THAM KHẢO1. Luigi Atzori, Antonio Iera, Giacomo Morabito, Internet of Things: Asurvey, Computer Networks 54 (2010) 2787–2805.  
2. Dr. Ovidiu Vermesan, Dr. Peter Friess, Patrick Guillemin, Internet ofThings Strategic Research Roadmap, 2009 Strategic Research Agenda, The IoT  
European Research Cluster - European Research Cluster on the  
Internet of Things (IERC).  
3. Everton Cavalcante, Marcelo Pitanga Alves, An Analysis of ReferenceArchitectures for the Internet of Things, Corba 2015.  
4. Anna Ha’c, Wireless Sensor Network Designs, University of Hawaii at  
Manoa, Honolulu, USA, John Wiley & Sons Ltd, Copyright 2003.  
5. Edgar H.Callaway, Jr. Wireless Sensor Networks: Architectures andProtocols, A CRC Press Company, Copyright © 2004 CRC Press LLC.  
6. Ovidiu Vermesan, Peter Friess, Internet of Things – ConvergingTechnologies For Smart Environments and Integrated Ecosystems, River  
Publishers Series in Communications.  
7. Kiran Maraiya, Kamal Kant, Nitin Gupta, Application based Study onWireless Sensor Network, International Journal of Computer Application (0975-  
8887), Volume 21, No.8, May 2011.  
8. I.F. Akyildiz, W. Su\*, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor  
networks: A survey, Broadband and Wireless Networking Laboratory, School  
of Electrical and Computer Engineering, Georgia Institute of Technology,  
Atlanta, GA 30332, Received 12 December 2001; accepted 20 December  
2001.<http://www.ece.gatech.edu/research/labs/bwn/sensornets.pdf>.9. Mainwaring, Polastre, et al. Wireless Sensor Networks For HabitatMonitoring, online posting. 2002 ACM International Workshop on Wireless  
Sensor Networks and Applications September 28, 2002. Atlanta, GA. (also Intel Research, IRB-TR-02-006, June 2002.) 12 Dec 2002.  
<http://www.cs.berkeley.edu/~polastre/papers/wsna02.pdf>.  
10. Matt Richardson & Shawn Wallace, Getting Started with Raspberry Pi.11. Maik Schmidt, Raspberry Pi, A Quick-Start Guide.