

NGHIÊN CỨU, THIẾT KẾ BÀN PHÍM CẢM ỨNG ĐIỆN DUNG ĐA ĐIỂM TUYẾN TÍNH TRÊN CƠ SỞ NGUYÊN LÝ PHÓNG NẠP TỤ CƠ SỞ ỨNG DỤNG TRONG NHÀ THÔNG MINH

Research, Designing Linear capacitive Multi- touch Sensors Keypad that rely on charge transfer acquisition principle implicated in smart home

Đinh Văn Nam
Trường Đại Học Vinh
e-Mail: quangnam.auto.tech@gmail.com

Hồ Sỹ Phương
Trường Đại Học Vinh
e-Mail: hosyphuong@gmail.com

Đặng Thái Sơn
Trường Đại Học Vinh
e-Mail: sondtbk@gmail.com

Tóm tắt

Bài báo trình bày nguyên lý và giải pháp thiết kế thiết bị cảm ứng điện dung đa điểm dạng tuyến tính ứng dụng làm công tắc điều khiển tuyến tính các đại lượng trong nhà thông minh như: thiết bị chiếu sáng, quạt, độ mở rèm cửa.... Thiết bị này được chế tạo trên cơ sở ứng dụng bộ vi điều khiển ARM Cortex M3 của hãng ST Microelectronic trên nền tảng hệ điều hành thời gian thực RTX. Bộ vi điều khiển điều khiển quá trình phóng nạp tụ và thu nhận thông tin về vị trí chạm của ngón tay trên thiết bị cảm ứng. Sau đó thực hiện việc điều khiển thiết bị điện một cách trực tiếp hoặc thông qua Wi-Fi từ bàn phím cảm ứng.

Từ khóa: Cảm ứng điện dung, phóng nạp tụ, nhà thông minh

Abstract: Linear capacitive multi touch sensors can be used to produce linear control relays for some parameters that are controlled by devices in a smart home such as light devices, fans and autonomous window blind cords. In this paper, the authors present the fundamental principles of these sensors and proposed a solution to design them. According to this proposed approach, ARM M3 micro controllers of STmicroelectronics with RTX real-time operating system are used as the central module. The authors use the micro-controller to monitor and calculate the charge transfer process of the capacitor. Eventually, the system performs light intensity control via wi-fi from keypads.

Keywords: capacitive touch sensor, charge transfer capacitor, smart home

Ký hiệu

Ký hiệu	Đơn vị	Ý nghĩa
$t_{TRANSFER}$	s	Chu kỳ xung phóng nạp tụ
$t_{SAMPLING}$		Thời gian lấy mẫu
t_{BURST}		Thời gian nạp tụ C_X
V_{REG}	V	Điện áp ngưỡng trên
V_{TRIP}		Điện áp ngưỡng
V_{OUT}		Điện áp ra
V_{IN}		Điện áp vào
C	F	Tụ điện
C_X		Tụ điện tích điện
C_s		Tụ điện cơ sở (mẫu)

Chữ viết tắt

AT	Attention Command
FSM	Finite State Machine
IOT	Internet of Things
RTOS	Real Time Operating System
LCD	Liquid Crystal Display
UART	Universal Synchronous Receiver/Transmitter
PWM	Pulse-Width Modulation

1. Phần mở đầu

Ngày nay, với chất lượng đời sống ngày càng cao thì nhu cầu có một ngôi nhà thông minh với đầy đủ các tiện nghi với xu hướng “internet of thing” là điều tất yếu. Ở Việt Nam, nhà thông minh đã xuất hiện và phát triển mạnh trong vài năm trở lại đây. Với yêu cầu thay thế các công tắc cơ khí trước đây bằng các thiết bị cảm ứng có sự thông minh, bền bỉ, tiện dụng và có tính thẩm mỹ cao. Hiện tại ở Việt Nam các loại bàn phím kiểu cảm ứng này có giá thành cao và không tích hợp các ứng dụng truyền thông thông minh. Với ý tưởng làm chủ công nghệ, nội địa hóa sản phẩm này

và đưa vào ứng dụng trong nhà thông minh. Trong báo cáo này, chúng tôi sẽ trình bày những vấn đề liên quan tới việc nghiên cứu, thiết kế bàn phím cảm ứng đa điểm dạng tuyến tính điều khiển cường độ sáng của đèn, đồng thời truyền thông dữ liệu vị trí chạm của ngón tay thông qua mạng Wi-Fi.

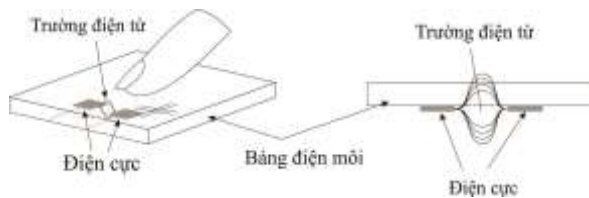
2. Nội dung chính

2.1 Các Nguyên lý cơ bản về cảm ứng điện dung [2][3][4]

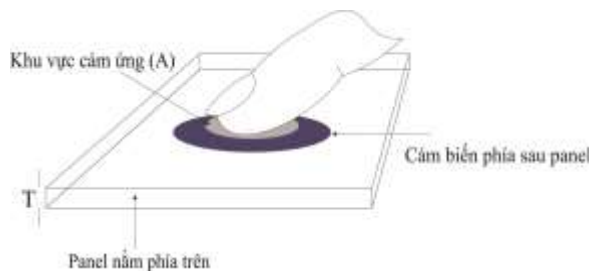
2.1.1 Nguyên lý cơ bản của cảm biến điện dung

Điện dung cảm ứng sẽ thay đổi khi ngón tay đến gần vùng nhạy cảm với điện trường được thể hiện như trong hình H.1. Sơ dĩ xảy ra điều này vì có sự xuất hiện của:

- Tụ sinh ra giữa tay người và thiết bị
- Tụ sinh ra giữa cơ thể người và board đồng qua lớp không khí
- Tụ giữa chân người và mặt đất



H.1 Minh họa bàn phím cảm ứng



H.2 Minh họa quá trình chạm vào bàn phím

Trong H.2 khi ngón tay được đưa đến gần bản mạch thì sẽ có một tụ điện được sinh ra do hiện tượng cảm ứng điện từ. Giá trị của tụ là:

$$C = \frac{\epsilon_0 \times \epsilon_r \times A}{T} \quad (1)$$

Trong đó:

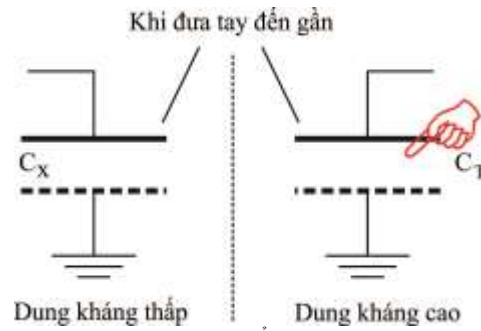
ϵ_0 và ϵ_r lần lượt là hằng số điện môi trong môi trường không khí và chất liệu cách ly.

A: diện tích chạm; T : độ dày của chất liệu cách ly.

Rõ ràng trong công thức (1), khi tiến gần đến bàn phím, thì có sự xuất hiện của không gian A và nó thay đổi trong quá trình chạm. Điều này dẫn đến sự biến đổi của giá trị điện dung C, đây chính là nguyên lý cơ bản dùng để nhận dạng vị trí điểm chạm thông qua giá trị điện dung C.

2.1.2 Nguyên lý đo thời gian phóng nạp của tụ

Như ta đã trình bày trong mục 2.1.1 khi có ngón tay lại gần và chạm vào mặt kính thì sẽ có sự thay đổi của điện dung, cụ thể là giá trị điện dung sẽ cao hơn khi chưa chạm vào kính như trong hình H.3.



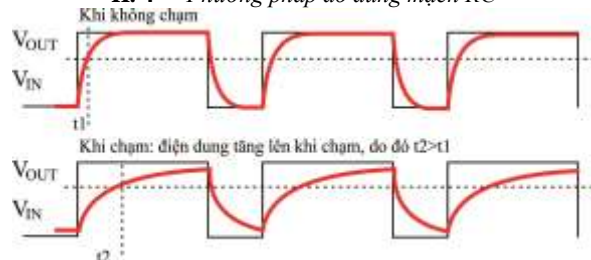
H.3 Minh họa sự thay đổi của điện dung khi chạm

Có hai phương pháp để đo sự thay đổi điện dung khi chạm:

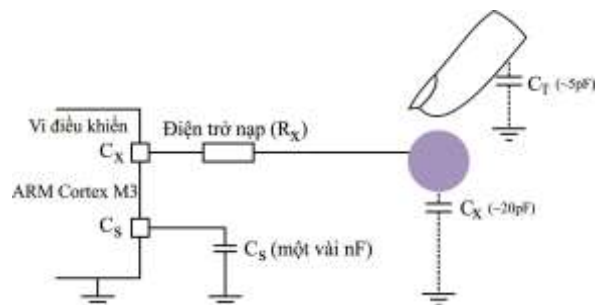
- Phương pháp đo thời gian phóng nạp dùng mạch RC [2]
- Phương pháp đo thời gian phóng nạp tụ cơ sở Cs [3]



H.4 Phương pháp đo dùng mạch RC



H.5 Quá trình đo RC khi chạm và không chạm

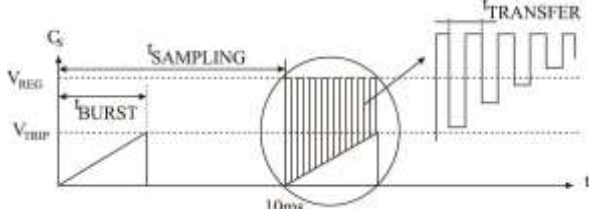


H.6 Phương pháp đo thời gian phóng nạp[3]

Với phương pháp đo RC (H.4, H.5) khi tạo một xung vuông đầu vào V_{IN} thì dựa vào thời gian quá độ của V_{OUT} sẽ suy ra được giá trị của tụ C, từ đó xác định được khả năng chạm hay không chạm của ngón tay, phương pháp này thường dùng để xác định khả năng chạm hay không của nút bấm cảm ứng do thời gian phóng nạp không đo được chính xác. Một phương pháp đo khác cho độ chính xác cao hơn gọi là phương pháp là đo thời gian phóng nạp tụ (H.6), cụ thể ở đây là đo thời gian nạp đầy tụ cơ sở C_s với giá trị cho trước, nguyên lý của quá trình này như sau:

- B1: Cách ly C_x và C_s , xả hết cả hai tụ.

- B2: Nạp đầy tụ C_x sau thời gian t_{BURST} , sau đó làm rỗng một khoảng thời gian và nạp vào tụ cơ sở C_s .
- B3: Lập lại bước 2 cho đến khi điện áp trên tụ C_s bằng V_{TRIP} .
- B4: Kết thúc quá trình đọc giá trị trong thời gian $t_{TRANSFER}$ như trong hình H.7.



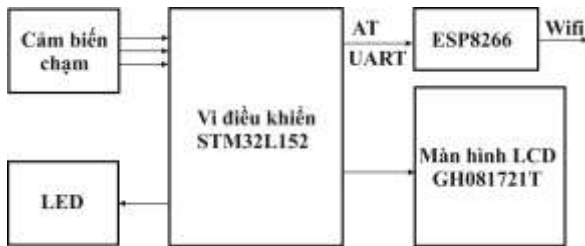
H. 7 Quá trình đo thời gian phóng nạp [3]

Phương pháp này cho độ chính xác cao đặc biệt là trong bài toán thiết kế kiểu cảm ứng điện dung đa điểm dạng tuyến tính.

Trong bài báo này chúng tôi tập trung vào thực hiện phương pháp đo thời gian phóng nạp tụ cơ sở được trình bày ở trên.

2.2 Thiết kế phần cứng

Với thử nghiệm thiết kế được xây dựng như trong H.8



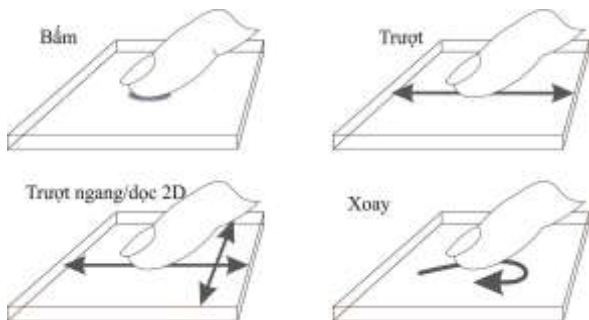
H. 8 Sơ đồ khối tổng quan của hệ thống

Hệ thống thử nghiệm như trong H.8 gồm có:

- Vi điều khiển trung tâm [1]
- Bàn phím cảm ứng điện dung tuyến tính
- Module Wi-Fi ESP8266 giao tiếp với vi điều khiển qua tập lệnh AT [7]
- Màn hình LCD hiển thị một số thông tin

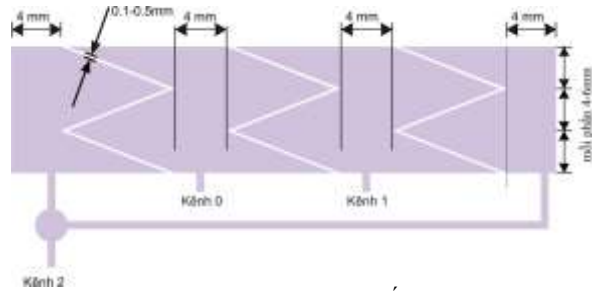
H.9 chỉ ra các dạng bàn phím cảm ứng [4]:

- Button: cảm ứng kiểu công tắc bấm
- Slider: cảm ứng kiểu trượt tuyến tính
- Touchscreen/Touchpad: cảm ứng kiểu trượt 2D
- Wheel: cảm ứng kiểu tròn xoay tuyến tính



H. 9 Các dạng bàn phím cảm ứng

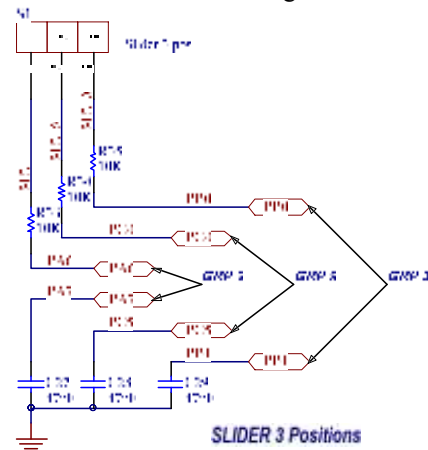
Trong bài báo này chúng tôi thử nghiệm bàn phím kiểu slider tuyến tính dạng thẳng trượt (H.9). Trong H.10 là hình ảnh bố đồng được thiết kế với kích thước dưới, độ dài tầm khoảng 60mm được bao quanh bởi đất để chống nhiễu ESD tốt hơn.



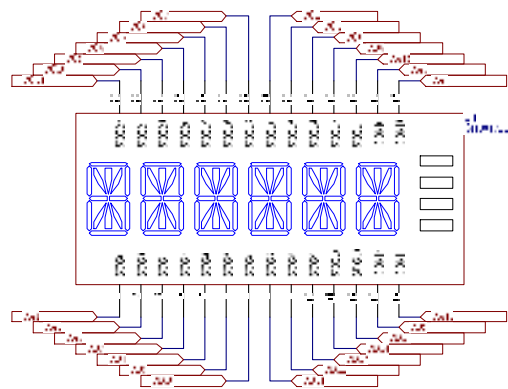
H. 10 Bàn phím Slider tuyến tính [4]

Chú ý rằng: khi ngón tay di chuyển trượt từ trái sang phải thì giá trị điện dung sẽ tuyến tính với từng vùng của bo mạch.

Với nguyên lý đo thời gian nạp tụ cơ sở C_s , thì trong báo cáo này chúng tôi sử dụng 3 Groups như H.11 và kết nối tới vi điều khiển như trong H.14.



H. 11 Nguyên lý kết nối ba nhóm với vi điều khiển



H. 12 Sơ đồ nguyên lý ghép nối LCD -GH081721T



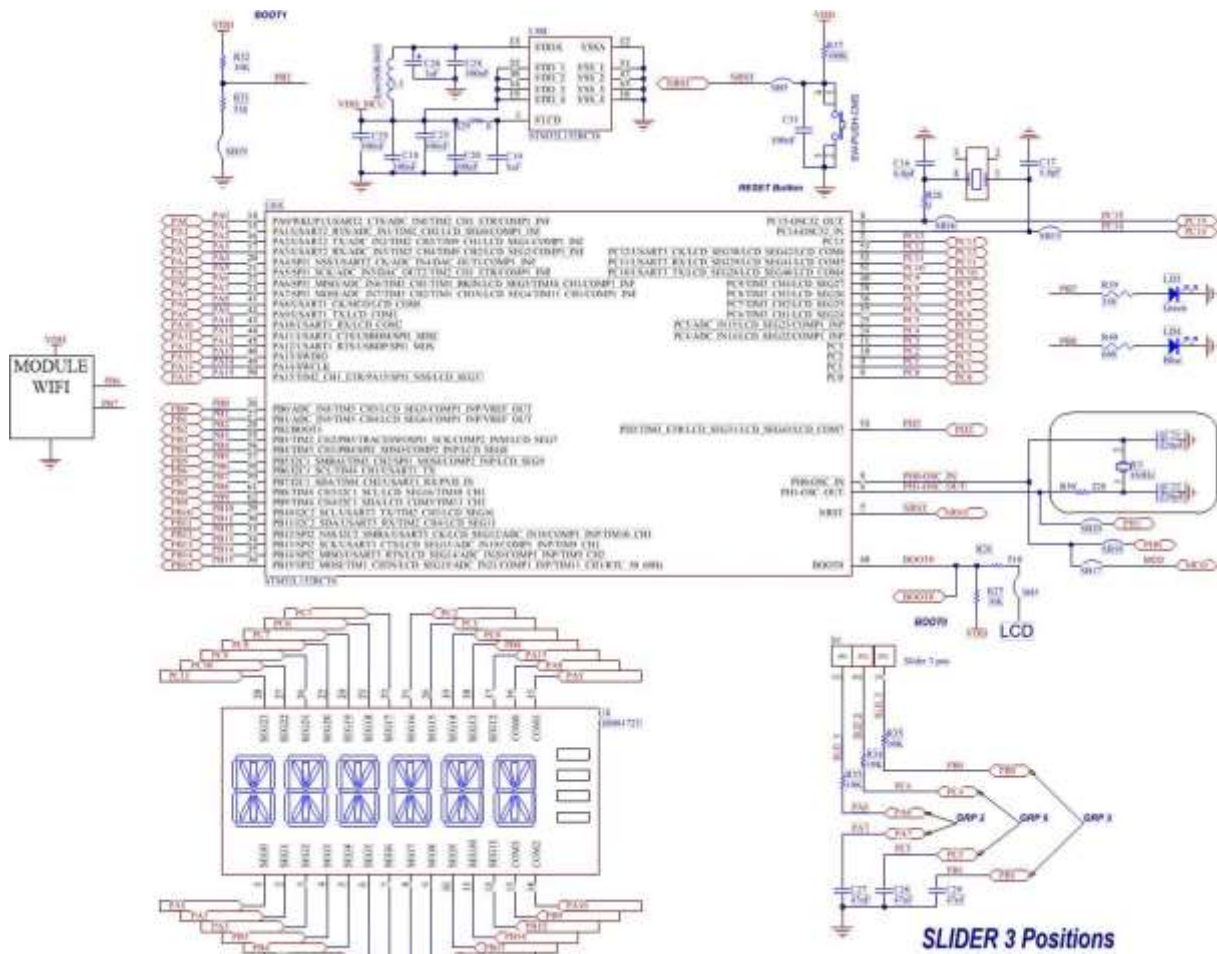
H.13 Module Wi-Fi ESP8266 [7]

H.13 là module truyền nhận Wi-Fi sử dụng chip ESP8266 SoC (System on Chip), được sử dụng làm nhiệm vụ truyền nhận thông tin điều khiển từ bộ điều khiển trung tâm. ESP8266 với lõi vi xử lý 32 bit hoạt

động với tần số 80MHz, 64KB RAM và 512KB Flash. Kết nối mạng không dây, giống như một máy chủ hoặc một cầu nối trung gian.

Chức năng:

- Hỗ trợ chuẩn 802.11 b/g/n Wi-Fi 2.4 GHz, hỗ trợ WPA/WPA2.
- Chuẩn điện áp hoạt động: 3.3V
- Chuẩn giao tiếp nối tiếp UART với tốc độ Baud lên đến 115200
- Có 3 chế độ hoạt động: Client, Access Point, Both Client and Access Point
- Hỗ trợ các chuẩn bảo mật như: OPEN, WEP, WPA_PSK, WPA2_PSK, WPA_WPA2_PSK.



H.14 Sơ đồ nguyên lý các khối cơ bản của mạch

2.3 Các thuật toán và giải thuật trong chương trình

Trong hệ thống này chúng tôi viết 2 module chương trình:

- Chương trình nhúng cho vi điều khiển
- Chương trình giao tiếp TCP/IP dùng phần mềm Labview

Với chương trình nhúng thực hiện rất nhiều công việc cùng một lúc như: tạo xung, đo số xung của cảm biến,

điều khiển module Wi-Fi, LCD,...chúng tôi sử dụng hệ điều hành thời gian thực [5] RTX.

2.3.1 Hệ điều hành thời gian thực RTX [5][6]

Cơ chế thực hiện đa nhiệm được thực hiện trên cơ sở hệ điều hành thời gian thực RTX. Khác với nhân trong các hệ điều hành như iOS, Android,... thì nhân của RTX RTOS có kiến trúc nhỏ gọn hơn và thích hợp trong các ứng dụng điều khiển tự động hóa trên nhiều nền tảng vi điều khiển có các kiến trúc phần cứng khác nhau. Trong RTX RTOS cung cấp các mức

xử lý ưu tiên và ngắt. Các công việc ưu tiên cao hơn sẽ được thực hiện trước. Ngoài các chức năng của hệ điều hành như trên, hệ điều hành thời gian thực có thể hỗ trợ thêm các chức năng sau:

- Lập lịch phân chia thời gian sử dụng tài nguyên, đặt mức ưu tiên các tác vụ.
- Truyền thông và đồng bộ giữa các tác vụ hoặc giữa tác vụ và ngắt.
- Phân phối bộ nhớ động.
- Quản lý các thiết bị vào ra.

Hệ điều hành thời gian thực sẽ phân chia các khe thời gian dựa vào systick timer (H.16), các tick thời gian này sẽ làm nhiệm vụ luân chuyển các tác vụ liên quan, sự giao tiếp giữa các tác vụ này theo cơ chế Semaphore (H.17), nhằm đảm bảo việc thực hiện các tác vụ không bị xung đột lẫn nhau.

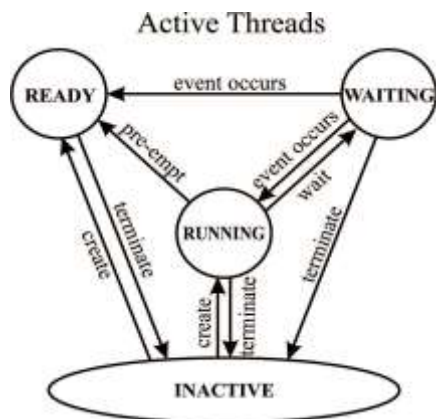
Trong H.15 với:

- Running: bộ xử lý đang thực hiện tác vụ. Với một bộ xử lý thì chúng ta chỉ chạy một tác vụ tại một thời điểm nhất định.

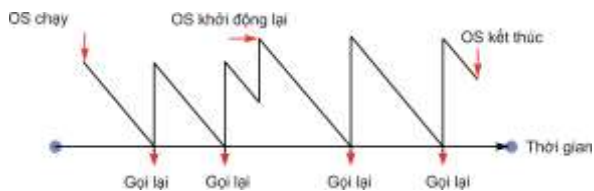
- Ready: một số tác vụ khác sẵn sàng chạy nếu bộ xử lý rỗi.

- Inactive: tác vụ không sẵn sàng chạy kể cả khi bộ xử lý trong trạng thái nghỉ- Idle. Tác vụ ở trong trạng thái này vì chúng đợi một sự kiện bên ngoài tác động để kích hoạt nó trở lại trạng thái sẵn sàng.

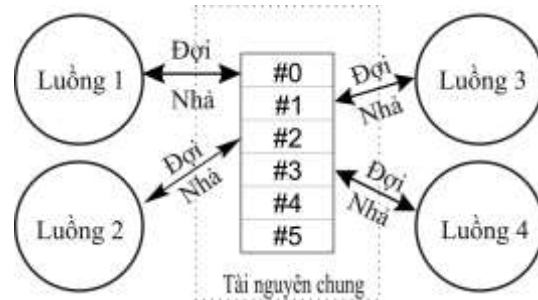
- Waiting: đợi tác vụ tiếp



H. 15 Cơ chế thực hiện chuyển các trạng thái[4]



H. 16 Cơ chế phân chia khe thời gian[4]



H. 17 Cơ chế tránh xung đột sử dụng Semaphore[4]

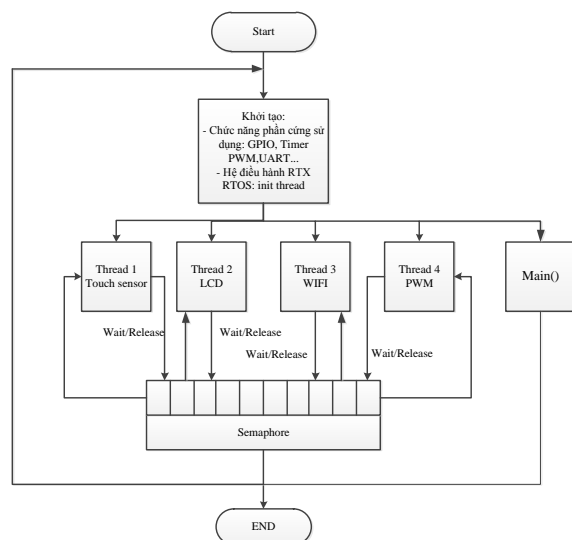
2.3.2 Chương trình nhúng cho vi điều khiển[1]

Vi điều khiển ARM Cortex M3 STM32L152 sẽ có nhiệm vụ làm bộ điều khiển trung tâm cho hệ thống client, vi điều khiển sẽ thực hiện đồng thời các nhiệm vụ sau:

- Đo và tính toán giá trị của cảm biến điện dung
- Điều khiển PWM đèn LED
- Hiển thị giá trị lên LCD
- Điều khiển module Wi-Fi

H.17 là thuật toán nhúng cho vi điều khiển, chương trình sẽ bắt đầu bằng việc khởi tạo hệ thống gồm có các cấu hình GPIO, Timer, UART, PWM...sau đó là cấu hình hệ điều hành RTX RTOS và khởi tạo 4 thread:

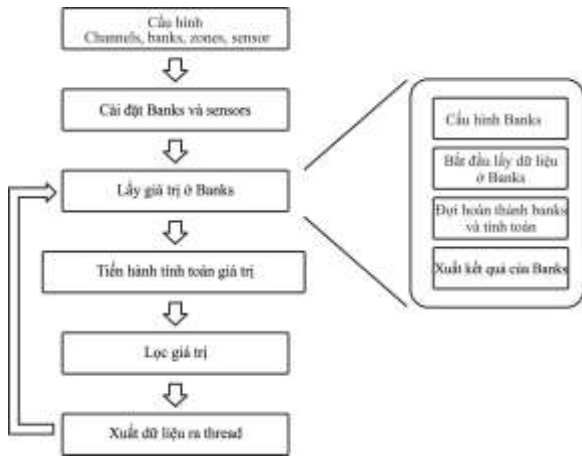
- Thread 1: Touch sensor là luồng làm việc với cảm biến điện dung, nhiệm vụ là phát xung và đọc thời gian nạp đầy tụ Cs.
- Thread 2: LCD luồng này làm nhiệm vụ điều khiển LCD
- Thread 3: Wi-Fi nhiệm vụ là điều khiển module Wi-Fi bằng lệnh AT qua UART.
- Thread 4: PWM làm nhiệm vụ điều khiển độ sáng của LED
- Main(): nhiệm vụ trung tâm



H. 18 Thuật toán nhúng của hệ thống

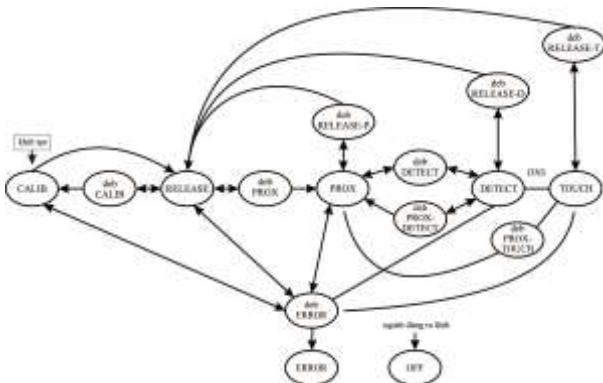
Sau đây chúng tôi sẽ đi sâu chi tiết vào các thread:

- Thread 1: luồng làm việc với cảm biến điện dung với lưu đồ thuật toán như dưới được viết theo kiểu máy trạng thái FSM:



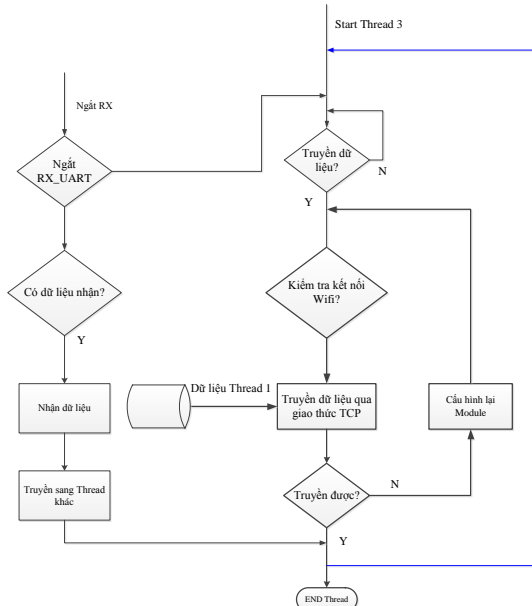
H.19 Thuật toán nhúng của hệ thống

Mô tả máy trạng thái gồm 20 trạng thái như H.20 được viết trong code.



H.20 Mô tả trạng thái FSM của hệ thống

- Thread 2: Điều khiển LCD 24 đoạn, 4 dòng sơ đồ như H.12, hiển thị các ký tự, các chữ số lên LCD
- Thread 3: Điều khiển quá trình truyền nhận dữ liệu qua lệnh AT với module Wi-Fi qua giao thức TCP/IP chuyển dữ liệu qua socket. Thuật toán như H.21:



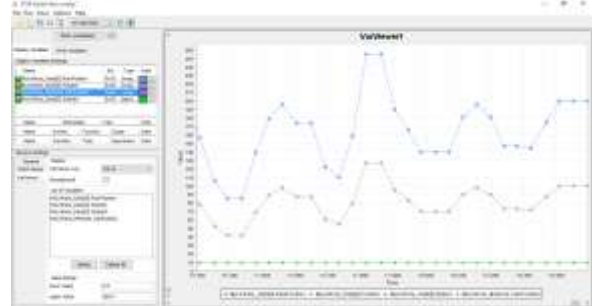
H.21 Thuật toán thread Wi-Fi

- Thread 4: PWM là thread tạo ra phục vụ cho việc băm xung, để điều khiển độ sáng LED thử nghiệm. sử dụng timer 4 làm nhiệm vụ tạo ngắt để băm xung.

Hình H.22 đo các biến trong chương trình với phần mềm STM studio. Các giá trị vị trí 8 bit (chỉ có CalcPosition 32 bit) được lưu trong biến:

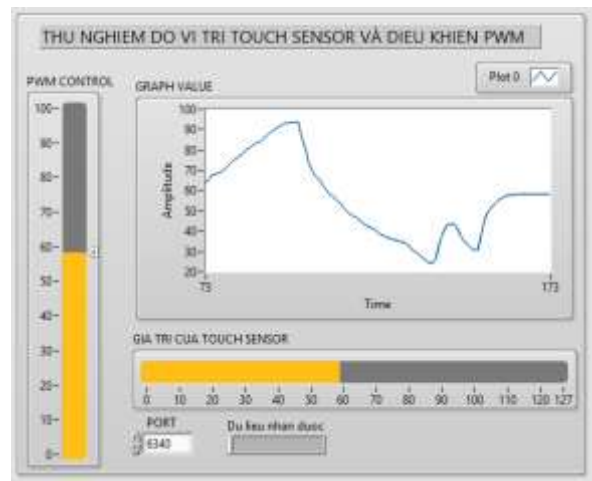
- MyLinRots_Data[0].RawPosition
- MyLinRots_Data[0].Position
- MyLinRots_Methods.CalcPosition
- MyLinRots_Data[0].StateId

Kết quả hình H. 22 thể hiện sự biến động khi chạm của các biến và giá trị MyLinRots_Data[0].StateId luôn bằng 10.



H. 22 Đo giá trị các biến giá trị và vị trí của touch sensor với phần mềm STM Studio

Trên H.23 là chương trình chúng tôi thử nghiệm để đọc giá trị touch sensor và vị trí chạm của ngón tay. Với các chức năng: hiển thị giá trị cảm biến touch sensor qua giao thức TCP dùng socket, điều khiển độ sáng Led trên phần mềm.



H. 23 Phần mềm đo lường điều khiển trên máy tính

3. Kết luận

Kết quả của nghiên cứu là đã tạo ra được một thiết bị bàn phím thông minh theo xu hướng IOT. Bước đầu đã thử nghiệm thành công dùng công nghệ bàn phím cảm ứng đa điểm dạng tuyến tính. Bài báo trình bày một số kiến thức về cảm ứng điện dung và công nghệ đo điện dung trên cơ sở đo thời gian nạp cho tụ cơ sở và ứng dụng trong chế tạo bàn phím cảm ứng. Trên cơ sở đó đã trình bày một thiết kế hoàn chỉnh cho bàn phím

mại. Các thuật toán điều khiển, giám sát đã được mô tả ở mục 2.3. Công trình nghiên cứu có ý nghĩa khoa học tốt và ứng dụng cao, phù hợp với chủ trương của đảng và nhà nước về chính sách phát triển KHCN.

Tài liệu tham khảo

- [1] STMicroelectronics Inc, 2015 STM32L151x6/8/B, STM32L152x6/8/B
- [2] STMicroelectronics Inc, 2015 AN2927 Application note, RC acquisition principle for touch sensing applications
- [3] STMicroelectronics Inc, 2015, STM8T141, Single-channel capacitive sensor for touch or proximity detection with shielded sensing electrode
- [4] STMicroelectronics Inc, 2015 AN2869 Guidelines for designing touch sensing applications
- [5] CMSIS-RTOS guide, 2015, Keil Tools by ARM
- [6] Jean J. Labrosse, uC/Os III The realtime kernel, 2009
- [7] ESP8266EX Datasheet v4.3, 2015 Espressif System IOT Team

động hóa của Học viện kỹ thuật Quân sự năm 2010. Hiện tại anh là NCS chuyên ngành kỹ thuật điện tử tại Đại Học Bách Khoa Hà Nội từ 2012. Từ năm 2010 anh là trưởng bộ môn ĐKTD- khoa Điện tử Viễn Thông, trường Đại học Vinh. Hướng nghiên cứu chính là xử lý và nhận dạng tín hiệu phi tuyến, thiết kế các hệ thống điện tử, điều khiển tự động hóa thông minh.



Đinh Văn Nam sinh năm 1989. Tốt nghiệp đại học ngành Kỹ sư Tự Động hóa, đại học Bách Khoa Hà Nội năm 2012; Nhận bằng thạc sỹ về Điều khiển và Tự động hóa của Học viện kỹ thuật Quân sự năm 2015. Từ năm 2013 anh là cán bộ giảng dạy tại bộ môn ĐKTD, trường Đại học Vinh.

Hướng nghiên cứu chính là thiết kế các hệ thống điều khiển nhúng, các hệ thống điều khiển thời gian thực, ứng dụng Machine Learning trong xử lý ảnh và mô hình hóa - mô phỏng hệ thống điều khiển.



Hồ Sỹ Phương sinh năm 1986. Tốt nghiệp đại học ngành Kỹ sư Điện tử Viễn thông năm 2009, Đại Học Vinh; Nhận bằng thạc sỹ về Tự động hóa của Học viện kỹ thuật Quân sự năm 2012. Từ năm 2009 anh là cán bộ giảng dạy khoa Điện tử Viễn thông, trường Đại học Vinh. Hướng

nghiên cứu chính là điều khiển hệ nhúng, FPGA trong xử lý tín hiệu, mô hình hóa mô phỏng và điều khiển thiết bị bay UAV.



Đặng Thái Sơn sinh năm 1981. Tốt nghiệp đại học ngành Kỹ sư tài năng Điện tử Viễn thông năm 2005, Đại học Bách Khoa Hà Nội; Nhận bằng thạc sỹ về Tự