

NGHIÊN CỨU THIẾT KẾ ĐIỀU KHIỂN ROBOT HÌNH NGƯỜI PHỤC VỤ ĐÀO TẠO TẠI TRƯỜNG ĐHCN QUẢNG NINH

RESEARCH AND CONTROL DESIGN OF HUMAN ROBOT TRAINING SERVICES AT QUANG NINH INDUSTRIAL UNIVERSITY

Nguyễn Thị Trang*

Khoa Điện, Trường Đại học Công nghiệp Quảng Ninh

*Email: trang.edu84@gmail.com

Mobile: 0988353484

Tóm tắt

Từ khóa:

Cảm biến; Điều khiển; Động cơ servo; Robot hình người.

Bài báo nghiên cứu Robot mô phỏng một số chuyển động thực tế giống như con người. Trên cơ sở nghiên cứu đó thiết kế điều khiển Robot hình người phục vụ đào tạo tại trường Đại học Công nghiệp Quảng Ninh. Mô hình Robot hình người đã mô phỏng được các động tác như đi bộ, sang trái, sang phải, tiến, lùi, các động tác chào, chống đẩy, trượt... Kết quả nghiên cứu có tính ứng dụng cao trong phòng thực hành Robot của trường Đại học Công nghiệp Quảng Ninh.

Keywords:

Sensor; Control; Servo motor; Humanoid robot.

Abstract (*Tập san nên có*)

Research paper Robot simulates some real human-like movements. Based on that research, design and control humanoid robot for training at Quang Ninh University of Industry. The humanoid robot model has simulated movements such as walking, left, right, forward, backward, greeting movements, push-ups, sliding... The research results have high applicability in the practice room. Robot of Quang Ninh University of Industry.

1. ĐẶT VẤN ĐỀ

Trong một vài năm gần đây, cùng với sự phát triển mạnh mẽ của các công nghệ điện tử, vi mạch, điều khiển, xử lý ảnh/thị giác máy tính, ngành Robotics đã có nhiều bước phát triển đột phá và trở nên ngày càng phổ biến hơn. Các Robot hình người có thể sao chép được một số các động tác của con người như cử động bàn tay, cánh tay và thậm chí có thể chuyển động giống như con người [1].

Con người ngày càng tạo ra những sản phẩm robot với nhiều tính năng, khả năng làm thay con người rất nhiều công việc. Thậm chí các loại robot công nghiệp còn thực hiện những thao tác làm việc dễ dàng mà không phải ai cũng làm được. Nhờ những lợi ích của robot trong đời sống mà nó giúp mang đến một cuộc sống dễ dàng và chất lượng hơn rất nhiều. Nhiều trường đại học, cao đẳng và dạy nghề về công nghệ – kỹ thuật đã có chương trình đào tạo về robot. Nhưng hình thức đào tạo chủ yếu là lý thuyết và mô phỏng. Sinh viên ít có điều kiện về cơ sở vật chất để thực hành. Một số trường có trang bị cánh tay robot thực hành nhập từ nước ngoài. Tuy nhiên việc sửa chữa, bảo hành những thiết bị này cũng gặp nhiều khó khăn. Nguyên nhân là do phụ thuộc vào nhà cung cấp. Vì vậy, thiết kế điều khiển robot mô phỏng một số chuyển động thực tế giống như con người có tính ứng dụng cao trong phòng thực hành robot [2].

2. CƠ SỞ LÝ THUYẾT

2.1. Giải pháp kỹ thuật

2.1.1. Nhiệm vụ

Robot hình người cần thực hiện những nhiệm vụ sau:

- Mô phỏng một số chuyển động thực tế của con người.
- Kết nối robot hình người và bộ điều khiển từ xa qua mạng truyền thông không dây.

2.1.2. Giải pháp

Vấn đề mô phỏng chuyển động thực tế của một số loài vật như: Chó, nhện và bọ cạp: Sử dụng các động cơ servo, cảm biến khoảng cách, cảm biến hồng ngoại, cảm biến con quay hồi chuyển được bán sẵn trên thị trường, bộ điều khiển trung tâm nhận dữ liệu phản hồi và thực hiện các xử lý khác.

Vấn đề mô phỏng chuyển động thực tế của robot hình người: Sử dụng các động cơ servo, cảm biến con quay hồi chuyển, cảm biến khoảng cách được bán sẵn trên thị trường, bộ điều khiển trung tâm nhận dữ liệu phản hồi và thực hiện các xử lý khác.

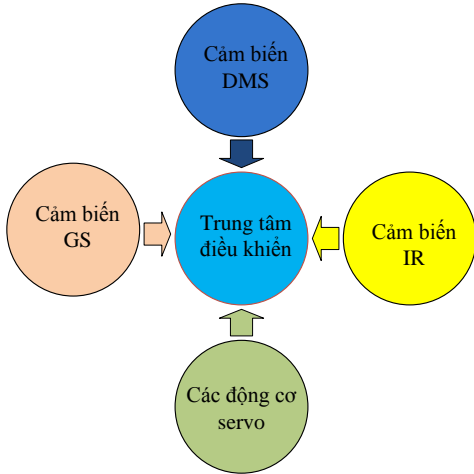
Vấn đề kết nối robot hình người và bộ điều khiển từ xa qua mạng truyền thông không dây cần có các đặc điểm sau:

- Phạm vi truyền thông ngắn
- Lượng thông tin cần truyền tải ít
- Độ tin cậy cao
- Tốc độ truyền tải thông tin nhanh
- Chi phí thấp
- Ưu tiên kết nối không dây
- Dựa vào các đặc điểm này nhóm nghiên cứu

chúng tôi lựa chọn xây dựng hệ thống truyền thông trên nền tảng mạng Zigbee và giao thức DYNAMIXEL.

2.1.3. Thiết kế sơ bộ

Sau khi lựa chọn giải pháp tôi đưa ra thiết kế sơ bộ như sau:



Hình 1. Mô hình tổng thể Robot nhiều hình dáng

Trung tâm điều khiển: tất cả các quá trình xử lý thông tin và đưa ra các lệnh điều khiển được thực hiện bởi khối này.

Cảm biến IR: khối này phát hiện các đối tượng hoặc màu sắc gần đó, sau đó gửi dữ liệu kết quả về trung tâm điều khiển.

Cảm biến DMS: khối này phát hiện và đo khoảng cách từ gần đến khá xa, sau đó gửi dữ liệu kết quả về trung tâm điều khiển.

Cảm biến GS: khối này phát hiện cân bằng thời điểm của robot, sau đó gửi dữ liệu kết quả về trung tâm điều khiển.

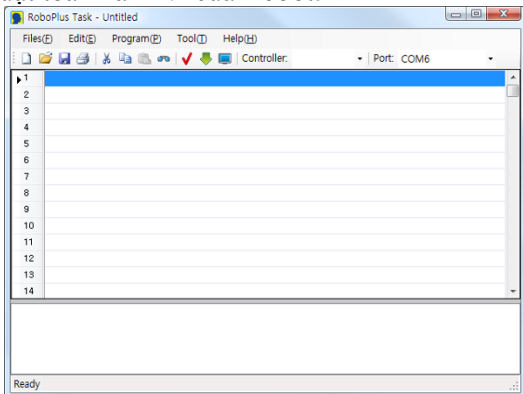
Các động cơ servo: thực hiện truyền động khi nhận được lệnh tương ứng từ trung tâm điều khiển.

2.2. Phần mềm RoboPlus

RoboPlus là một phần mềm Robot chuyên dụng dùng để lập trình Robot một cách dễ dàng. Ta có thể tận dụng các tính năng của RoboPlus để tạo ra các chuyển động robot độc đáo.

Phần mềm RoboPlus chia ra làm các RoboPlus khác nhau, mỗi loại có vai trò chức năng riêng [3].

RoboPlus Task: Là giao diện tác vụ dùng để tạo các thuật toán hành vi của Robot.



Hình 2. Giao diện của RoboPlus Task.

RoboPlus Manager: là giao diện quản lý dùng để quản lý các thành phần thiết bị robot.

Roboplus Motion: là giao diện để tạo ra các chuyển động phức tạp của Robot một cách dễ dàng. Các chuyển động được tạo trong Motion có thể nằm trong Task.

2.3. Giao thức DYNAMIXEL

Giao thức DYNAMIXEL dùng để truyền tải thông tin giữa các thiết bị

Để kiểm soát DYNAMIXEL, giao tiếp phải được thiết lập theo giao thức của DYNAMIXEL. DYNAMIXEL được điều khiển bằng cách nhận dữ liệu nhị phân.

Bộ điều khiển chính và DYNAMIXEL giao tiếp với nhau bằng cách gửi và nhận dữ liệu được gọi là Gói. Gói có hai loại: Gói hướng dẫn, mà Bộ điều khiển chính gửi để điều khiển DYNAMIXEL và Gói trạng thái, mà DYNAMIXEL phản hồi cho Bộ điều khiển chính.

ID là một số cụ thể để phân biệt từng DYNAMIXEL khi một số DYNAMIXEL được liên kết với một bus. Bằng cách cung cấp ID cho các gói lệnh và trạng thái, bộ điều khiển chính chỉ có thể điều khiển DYNAMIXEL mà ta muốn điều khiển.

DYNAMIXEL thực hiện Giao tiếp nối tiếp không đồng bộ với 8 bit, 1 bit dừng. Nếu DYNAMIXEL có cùng ID được kết nối, gói dữ liệu sẽ xung đột và sự cố mạng sẽ xảy ra. Do đó, hãy đặt ID sao cho không có DYNAMIXEL nào có cùng ID.

UART là một giao thức truyền thông nối tiếp trong đó cả TxD và RxD không thể được sử dụng cùng một lúc. Phương pháp này thường được sử dụng khi nhiều thiết bị cần được kết nối với một bus duy nhất. Vì nhiều thiết bị được kết nối với cùng một bus, nên tất cả các thiết bị khác cần ở chế độ đầu vào trong khi một thiết bị đang truyền. Bộ điều khiển Chính điều khiển các bộ truyền động DYNAMIXEL đặt hướng giao tiếp thành chế độ đầu vào và chỉ khi nó đang truyền một gói dữ liệu, nó mới thay đổi hướng sang chế độ đầu ra [4].

Gói ID cho biết ID của thiết bị sẽ nhận được Gói hướng dẫn và xử lý nó. Phạm vi: 0 ~ 253 (0x00 ~ 0xFD), là tổng số 254 số có thể được sử dụng. ID 254 (0xFE), làm cho tất cả các thiết bị được kết nối thực thi Gói lệnh.

Gói lệnh được sử dụng để kiểm tra xem gói có bị hỏng trong quá trình giao tiếp hay không. Checksum được tính theo công thức sau.

$$\text{Checksum} = \sim (\text{ID} + \text{Chiều dài} + \text{Chỉ dẫn} + \text{Tham số 1} + \dots + \text{Tham số N})$$

Trong đó “~” là toán tử bổ sung số lượng nhị phân. Khi kết quả tính toán của dấu ngoặc trong công thức trên lớn hơn 255 (0xFF), chỉ sử dụng các byte thấp hơn.

Ví dụ, khi muốn sử dụng Gói lệnh:

ID = 1 (0x01), Chiều dài = 5 (0x05), Hướng dẫn = 3 (0x03), Tham số 1 = 12 (0x0C), Tham số 2 = 100 (0x64), Tham số 3 = 170 (0xAA)
 Checksum = ~ (ID + Độ dài + Lệnh + Tham số 1 + ... Tham số 3) = ~ [0x01 + 0x05 + 0x03 + 0x0C + 0x64 + 0xAA] = ~ [0x123] // Chỉ byte dưới 0x23 thực hiện thao tác Không. = 0xDC

Do đó, Gói lệnh phải là 0xFF, 0xFF, 0x01, 0x05, 0x03, 0x0C, 0x64, 0xAA, 0xDC.

Gói trạng thái được sử dụng để kiểm tra xem gói có bị hỏng trong quá trình giao tiếp hay không. Kiểm tra trạng thái được tính theo công thức sau: Kiểm tra trạng thái = ~ (ID + Độ dài + Lỗi + Thông số 1 + ... Thông số N) (1)

3. THIẾT KẾ ĐIỀU KHIỂN ROBOT HÌNH NGƯỜI

3.1. Chức năng các khối của Robot hình người

Khối DYNAMIXEL: là đơn vị nắm giữ vai trò trung gian truyền tải thông tin giữa các thiết bị.

Khối Devices: gồm các thiết bị

Đơn vị xử lý trung tâm và truyền thông: thực hiện các xử lý chung và giao tiếp với các thiết bị khác.

- Cảm biến con quay hồi chuyển: phát hiện cân bằng thời điểm của robot sau đó gửi kết quả về đơn vị xử lý trung tâm.

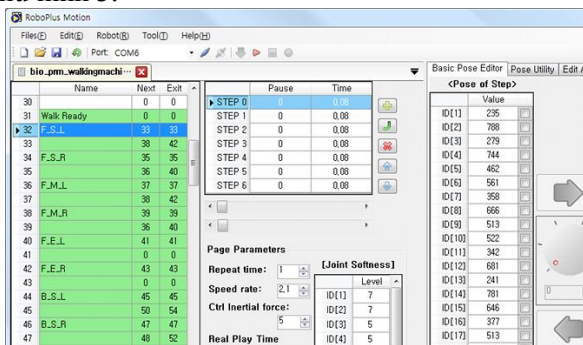
- Cảm biến khoảng cách: Phát hiện và đo khoảng cách sau đó gửi kết quả về đơn vị xử lý trung tâm.

- Các bộ truyền động Servo: Truyền động quay khi nhận lệnh tương ứng.

3.2. Chương trình điều khiển cơ bản của Robot hình người

3.2.1. Robot đi bộ

Tập chuyển động đi bộ của Robot được thể hiện như hình 3.



Hình 3. Tập chuyển động đi bộ của Robot

Các tập chuyển động đi bộ được sử dụng trong Robot đi bộ chứa 16 kiểu đi bộ khác nhau: Ở đằng trước, phía sau, rẽ trái, rẽ phải, đi bộ sang hai bên + trái, đi bộ trên vỉa hè + phải, rẽ trái+chuyển tiếp, rẽ phải+chuyển tiếp, lùi lại+trái, lùi lại+phải, tránh trái, tránh phải, chuyển tiếp+ theo đường chéo trái, chuyển tiếp+ theo đường chéo phải, lùi lại+theo đường chéo trái, lùi lại+theo đường chéo phải.

Mã tác vụ của Robot đi bộ bao gồm chức năng Initialization Walk và chức năng WalkExecute. Ta sử dụng 2 chức năng này để điều khiển robot từ xa.

- Chức năng Initialization Walk: Khởi tạo các biến và di chuyển robot đến vị trí mặc định của nó.

```

119 FUNCTION InitializationWalk
120 {
121     BalancePage = 224
122     WalkPageStart = 31
123     WalkCommand = 0
124     Motion Page = WalkPageStart
125     CALL WaitMotion
126     WalkState = 0
127 }
    
```

Hình 4. Khởi tạo các biến và di chuyển Robot

- Chức năng WalkExecute: Thực thi các mẫu đi bộ và tham gia các mô hình đi bộ.

```

129 FUNCTION WalkExecute
130 {
131     //Using WalkControl Variable
132     // 0 : Stop,      1 : Forward,      2 : Backward,      3 : Turn Left,
133     // 5 : Left Side, 6 : Right Side,  7 : 1 + 5,      8 : 1 + 6,
134     // 10 : 2 + 6,   11 : Avoid Left, 12 : Avoid Right, 13 : 1 + 3,   14 :
135     // 15 = 2+3,    16 = 2+4
136
137     //Return if the command is same.(Keep executing the present auto connecting motion.)
138     IF ( WalkCommand == WalkState )
139         RETURN
140
141
142     Temp1 = 196 + WalkPageStart
143     IF ( Motion Page < WalkPageStart || Motion Page > Temp1 )
144     {
145         CALL EXITPageWaitMotion
146         IF ( Motion Page != BalancePage )
    
```

Hình 5. Thực thi các mẫu và tham gia mô hình đi bộ

Bắt đầu chương trình: Viết một mẫu “Start Program” đơn giản bằng cách sử dụng chức năng Initialization Walk và chức năng WalkExecute để kết nối tron tru các mẫu đi bộ.Đầu tiên, hãy gọi hàm InitializationWalk để khởi tạo biến và đưa robot về vị trí mặc định của nó.

Chọn và nhập kiểu đi bộ từ 0 đến 16 cho biến WalkCommand, sau đó gọi hàm WalkExecute để chạy kiểu đi bộ đã chọn.

Khi nhập một số khác vào biến WalkCommand và gọi hàm WalkExecute, nó sẽ chuyển đổi sang kiểu đi bộ mới một cách tự nhiên nhất có thể.

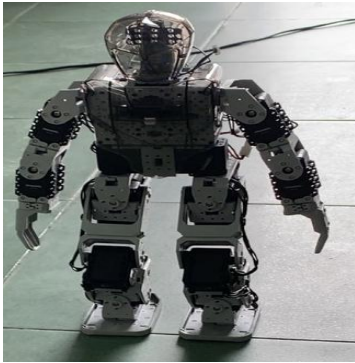
```

1 START PROGRAM
2 {
3     CALL InitializationWalk
4     ENDLESS LOOP
5
6     IF ( Remococon Arrived == TRUE )
7     {
8         ReceiveData = Remococon RXD
9         IF ( ReceiveData == U )
10        {
11            WalkCommand = 1
12            CALL WalkExecute
13        }
14        ELSE IF ( ReceiveData == D )
15        {
16            WalkCommand = 2
17            CALL WalkExecute
18        }
19    }
20 }
21 }
    
```

Hình 6. Bắt đầu viết chương trình cho Robot đi bộ

Có 16 mã mẫu kiểu đi bộ khác nhau để điều khiển thông qua bộ điều khiển từ xa trong tập mã nhiệm vụ của Người máy đi bộ. Thay đổi chức năng

trong các mã mẫu của Start Program để phù hợp với nhu cầu của mình muốn chọn.



Hình 7. Mô hình Robot hình người

3.2.2. Hiệu chỉnh cảm biến con quay hồi chuyển (Gyro)

Mục tiêu là ta tìm hiểu cách điều chỉnh tư thế của robot hình người bằng cảm biến con quay hồi chuyển.

Một cảm biến Gyro được sử dụng để xác định vận tốc góc (biến thiên góc trên giây). Khi rô bốt nghiêng và vận tốc góc tăng theo một hướng cụ thể, giá trị của động cơ servo có thể được điều chỉnh theo hướng ngược lại để làm thẳng rô bốt.

Gọi chức năng InitializationGyro được thể hiện như hình 8.

```

6  START PROGRAM
7  {
8    CALL InitializationWalk
9    CALL InitializationGyro
10
11  ENDLESS LOOP
12  {
13    IF ( Remocon Arrived == TRUE )
    
```

Hình 8. Gọi chức năng InitializationGyro

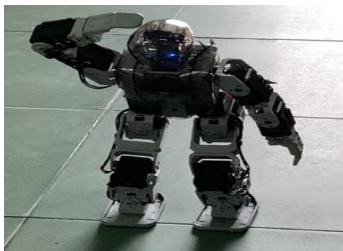
Thực thi BalancePage để áp dụng lệnh chung.

```

117  IF ( Motion Status == FALSE )
118  {
119    Motion Page = BalancePage
120  }
    
```

Hình 9. Thực thi BalancePage

Để thay đổi góc quay của khớp thì chuyển động phải được thực hiện. Khi không có chuyển động nào đang được phát, góc quay của khớp sẽ không được áp dụng ngay cả khi một giá trị được đưa vào bởi cảm biến con quay hồi chuyển. Các BalancePage được sử dụng để áp dụng chung khi không có chuyển động đã được thực hiện.



Hình 10. Robot mô phỏng động tác chào

Các chức năng InitializeGyro đọc giá trị cảm biến con quay hồi chuyển 10 lần trong một khoảng thời gian 0,128 giây và tiết kiệm giá trị trung bình

như một con quay hồi chuyển cảm biến giá trị tiêu chuẩn. (Giá trị tiêu chuẩn là Khoảng 250 độ). Nếu giá trị cảm biến con quay hồi chuyển nhỏ hơn 230 hoặc lớn hơn 270, nó giả định không có cảm biến con quay hồi chuyển và không sử dụng điều chỉnh (khi không có cảm biến con quay hồi chuyển hoặc rô bốt đã di chuyển trong quá trình khởi tạo).

```

345  FUNCTION InitializationGyro
346  {
347    FBBalCenter = 0
348    RLBalCenter = 0
349
350  LOOP FOR ( i = 1 ~ 10 )
351  {
352    FBBalData = PORT[3]
353    RLBalData = PORT[4]
354
355    FBBalCenter = FBBalCenter + FBBalData
356    RLBalCenter = RLBalCenter + RLBalData
357
358    Timer = 0.128sec
359    WAIT WHILE ( Timer > 0.000sec )
360  }
361
362  FBBalCenter = FBBalCenter / 10
363  RLBalCenter = RLBalCenter / 10
364
365  ExistGyro = TRUE
366  IF ( FBBalCenter > 270 || FBBalCenter < 230 )
367    ExistGyro = FALSE
368  IF ( RLBalCenter > 270 || RLBalCenter < 230 )
369    ExistGyro = FALSE
370
371  UseGyro = TRUE
372 }
    
```

Hình 11. Chức năng InitializationGyro

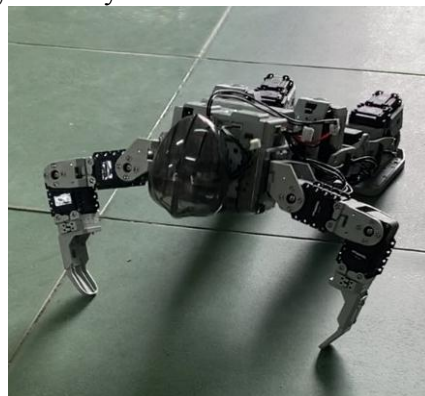
Tính toán giá trị điều chỉnh trong Callback Function được thể hiện như hình 12.

```

374  CALLBACK
375  {
376    IF ( UseGyro == TRUE && ExistGyro == TRUE )
377    {
378      FBBalData = PORT[3]
379      RLBalData = PORT[4]
380
381      FBBalError = FBBalData - FBBalCenter
382      RLBalError = RLBalData - RLBalCenter
383
384      FBBalErrorScaled = FBBalError * 4
385      RLBalErrorScaled = RLBalError * 4
386
387      FinalFBBal1 = FBBalErrorScaled / 54
388      FinalFBBal2 = FBBalErrorScaled / 18
389
390      FinalRLBal0 = RLBalErrorScaled / 20
391      FinalRLBal1 = RLBalErrorScaled / 40
    }
    
```

Hình 12. Tính toán giá trị điều chỉnh trong Callback Function

Trong các khoảng thời gian đều đặn, chức năng đọc giá trị cảm biến con quay hồi chuyển hiện tại và so sánh với giá trị tiêu chuẩn để tính toán giá trị điều chỉnh. Nếu biến UseGyro sai, rô bốt sẽ không tự điều chỉnh. Do đó, hãy đặt biến UseGyro thành false khi bạn không muốn sử dụng điều chỉnh con quay hồi chuyển.

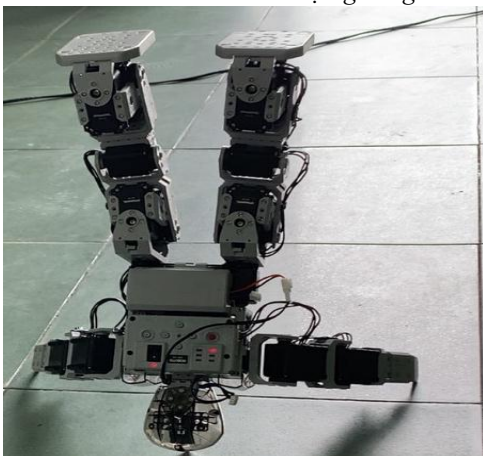


Hình 13. Robot mô phỏng động tác chống đẩy

Áp dụng giá trị điều chỉnh đã tính toán cho góc quay của khớp và điều chỉnh tư thế của rô bốt. Để điều chỉnh độ nghiêng trước / sau, ta phải sử dụng các khớp ở đầu gối và mắt cá chân, là cơ cấu truyền động 13-16. Để điều chỉnh độ nghiêng trái / phải, ta phải sử dụng các khớp ở mắt cá chân và thắt lưng, đó là các cơ cấu truyền động 9-10 và 17-18.

393	ID [13]:Joint offset = 0 + FinalFBBal1
394	ID [15]:Joint offset = 0 + FinalFBBal2
395	ID [14]:Joint offset = 0 - FinalFBBal1
396	ID [16]:Joint offset = 0 - FinalFBBal2
397	
398	ID [9]:Joint offset = 0 + FinalRLBal1
399	ID [10]:Joint offset = 0 + FinalRLBal1
400	ID [17]:Joint offset = 0 - FinalRLBal0
401	ID [18]:Joint offset = 0 - FinalRLBal0

Hình 14. Điều chỉnh độ nghiêng



Hình 15. Robot mô phỏng động tác chống đẩy lộn ngược

4. THẢO LUẬN

Nghiên cứu thiết kế điều khiển Robot hình người có tính ứng dụng cao trong phòng thực hành Robot của trường Đại học Công nghiệp Quảng Ninh. Những kết quả nghiên cứu của bài báo sẽ là bước

đầu mở ra nền tảng cho ngành nghiên cứu, thiết kế chế tạo robot của nhà trường, làm tài liệu tham khảo cho việc giảng dạy nghiên cứu của giảng viên, sinh viên. Ngoài ra, kết quả của bài báo cũng sẽ là bước đệm để tiếp tục phát triển cho các nghiên cứu chuyên sâu, tích hợp các công nghệ tiên tiến như IOT, AI vào ứng dụng, điều khiển robot, là cơ sở để giảng viên, sinh viên của nhà trường tiếp cận với những công nghệ tiên tiến.

5. KẾT LUẬN

Bài báo đã thiết kế điều khiển Robot hình người mô phỏng được một số động tác giống như con người như đi bộ, sang trái, sang phải, tiến, lùi, các động tác chào, chống đẩy, trượt....

Mô hình Robot hình người làm mô hình thí nghiệm cho phòng thực hành Robot, giúp cho sinh viên hiểu rõ hơn về Robot hình người.

TÀI LIỆU THAM KHẢO

- [1]. A. Schmitz, U. Pattacini, F. Nori, L. Natale, G. Metta, and G. Sandini. In *IEEE/RAS Int. Conf. on Humanoid Robots (HUMANOIDS)*, pages 186–191, 2010.
- [2]. Nestor Eduardo Nava Rodriguez. Design issue of a new iCub head sub-system. *Robotics and Computer-Integrated Manufacturing*, 26(2):119–129, 2010.
- [3]. U. Saranli, M. Buehler, and D. E. Koditschek, “Rhex-A simple and highly mobile hexapod robot,” *Int. J. Robot. Res.*, vol. 20, pp. 616–631, 2001.
- [4]. J.H. Su et. Al, An Intelligent Line-Following Robot Project for Introductory Robot Course, *World Transactions on Eng. And Technology Education*, Vol.8, No.4, pp.455-461, 2010.