

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG**



ISO 9001:2015

**ĐIỀU KHIỂN VỊ TRÍ ĐỘNG CƠ DC QUA GIAO DIỆN  
LABVIEW**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY  
NGÀNH ĐIỆN TỰ ĐỘNG CÔNG NGHIỆP**

**HẢI PHÒNG - 2020**



Cộng hoà xã hội chủ nghĩa Việt Nam  
**Độc lập – Tự Do – Hạnh Phúc**  
-----o0o-----  
BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

## **NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP**

Sinh viên: Phạm Ngọc Giang - Mã SV: 1512102026

Lớp: DC1901 - Ngành: Điện Tự Động Công Nghiệp

Tên đề tài: Điều khiển vị trí động cơ DC qua giao diện Labview

## NHIỆM VỤ ĐỀ TÀI

1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp (về lý luận, thực tiễn, các số liệu cần tính toán và các bản vẽ).

.....

.....

.....

.....

.....

.....

.....

.....

2. Các số liệu cần thiết để thiết kế, tính toán.

.....

.....

.....

.....

.....

.....

.....

.....

3. Địa điểm thực tập tốt nghiệp.

.....

.....

## CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

### Người hướng dẫn thứ nhất:

Họ và tên : Ngô Quang Vĩ  
Học hàm, học vị : Thạc Sĩ  
Cơ quan công tác : Trường Đại Học Dân Lập Hải Phòng  
Nội dung hướng dẫn : Toàn bộ đề tài

### Người hướng dẫn thứ hai:

Họ và tên:.....  
Học hàm, học vị:.....  
Cơ quan công tác:.....  
Nội dung hướng dẫn:.....

Đề tài tốt nghiệp được giao ngày tháng năm  
Yêu cầu phải hoàn thành xong trước ngày tháng năm

Đã nhận nhiệm vụ ĐTTN

*Sinh viên*

Phạm Ngọc Giang

Đã giao nhiệm vụ ĐTTN

*Người hướng dẫn*

Th.S. Ngô Quang Vĩ

*Hải Phòng, ngày ..... tháng.....năm 2019*

**Hiệu trưởng**

**GS.TS.NGUYỄN Trần Hữu Nghị**

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  
**Độc lập - Tự do - Hạnh phúc**

**PHIẾU NHẬN XÉT CỦA GIÁNG VIÊN HƯỚNG DẪN TỐT NGHIỆP**

Họ và tên giảng viên: .....

Đơn vị công tác: .....

Họ và tên sinh viên: ..... Chuyên ngành: .....

Đề tài tốt nghiệp: .....

Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp

.....

.....

.....

.....

1. Đánh giá chất lượng của đề án/khóa luận (so với nội dung yêu cầu đã đề ra trong nhiệm vụ Đ.T. T.N trên các mặt lý luận, thực tiễn, tính toán số liệu...)

.....

.....

.....

.....

.....

2. Ý kiến của giảng viên hướng dẫn tốt nghiệp

Được bảo vệ  Không được bảo vệ  Điểm hướng dẫn

*Hải Phòng, ngày ... tháng ... năm .....*

*Giảng viên hướng dẫn*

*(Ký và ghi rõ họ tên)*

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**Độc lập - Tự do - Hạnh phúc**

**PHIẾU NHẬN XÉT CỦA GIÁO VIÊN CHẤM PHẢN BIỆN**

Họ và tên giảng viên: .....

Đơn vị công tác: .....

Họ và tên sinh viên: ..... Chuyên ngành: .....

Đề tài tốt nghiệp: .....

**1. Phần nhận xét của giáo viên chấm phản biện**

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

**2. Những mặt còn hạn chế**

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

**3. Ý kiến của giảng viên chấm phản biện**

*Được bảo vệ*  *Không được bảo vệ*  *Điểm hướng dẫn*

Hải Phòng, ngày ... tháng ... năm .....

Giảng viên chấm phản biện

(Ký và ghi rõ họ tên)

## LỜI CẢM ƠN

Khi hoàn thành đồ án tốt nghiệp này cũng là em kết thúc thời gian học tập tại trường Đại học Dân lập Hải Phòng. Khoảng thời gian học tập và nghiên cứu tại trường đã giúp em hiểu và yêu quý nơi đây nhiều hơn. Nhà trường và Thầy Cô không những truyền đạt cho em những kiến thức chuyên môn mà còn giáo dục cho em về lý tưởng, đạo đức trong cuộc sống. Đây là những hành trang không thể thiếu cho cuộc sống và sự nghiệp của em sau này. Em xin bày tỏ lòng biết ơn sâu sắc đến tất cả các Quý Thầy Cô đã tận tình chỉ bảo, dẫn dắt em đến ngày hôm nay để có thể vững bước trên con đường học tập và làm việc sau này.

Đồ án tốt nghiệp đã đánh dấu việc hoàn thành những năm tháng miệt mài học tập của em. Và đồ án này cũng đánh dấu sự trưởng thành trên con đường học tập của em. Qua đây em xin gửi lời cảm ơn đến gia đình và bạn bè đã luôn động viên và tạo mọi điều kiện để nhóm hoàn thành khóa học.

Cuối cùng, em xin bày tỏ lòng biết ơn sâu sắc nhất đến Thầy Ngô Quang Vĩ với sự nhiệt tình giúp đỡ, tạo điều kiện thuận lợi và sự định hướng đúng đắn và kịp thời của Thầy đã giúp em rất nhiều trong quá trình thực hiện đồ án.

**Sinh viên thực hiện**  
Phạm Ngọc Giang



## MỤC LỤC

Lời mở đầu .....	
CHƯƠNG 1 : .....	
TỔNG QUAN VỀ PHẦN MỀM LẬP TRÌNH LABVIEW .....	
1.1: TỔNG QUAN VỀ LABVIEW .....	
1.1.1. LabVIEW là gì? .....	
1.1.2. Vai trò của LabVIEW .....	
1.1.3. Các chức năng chính của LabVIEW .....	
1.1.4. Phần mềm nhúng vào LabVIEW .....	
1.1.5. Các giao thức kết nối.....	
1.1.6. Các Module và bộ công cụ LabVIEW .....	
1.1.6.1. Các module LabVIEW .....	
1.1.6.2. Các bộ công cụ LabVIEW .....	
1.1.7. LabVIEW làm việc như thế nào?.....	
1.2. CÁC THÀNH PHẦN CỦA LABVIEW .....	
1.2.1. Bảng giao diện (The Front panel) .....	
1.2.2. Sơ đồ khối (The Block Diagram).....	
1.2.3. Viết chương trình trên nền LabVIEW.....	
CHƯƠNG 2.....	
TÌM HIỂU ĐỘNG CƠ MỘT CHIỀU VÀ CÁC PHƯƠNG PHÁP ĐIỀU KHIỂN	
2.1 ĐỘNG CƠ ĐIỆN 1 CHIỀU THÔNG THƯỜNG.....	
2.2 ĐỘNG CƠ ĐIỆN MỘT CHIỀU KÍCH TỪ BẰNG NAM CHÂM VĨNH CỬU ( DCVC).....	
2.2.1 : Cấu tạo .....	
2.2.2 : Nguyên lý hoạt động của DCVC .....	
2.2.3 Nguyên lý làm việc và phân loại động cơ điện một chiều.....	
2.2.4. Phương trình đặc tính cơ động cơ điện một chiều .....	

2.3 ĐIỀU KHIỂN TỐC ĐỘ ĐỘNG CƠ MỘT CHIỀU.....	
2.3.1. Các phương pháp điều khiển động cơ điện một chiều.....	
.....	
CHƯƠNG 3 :.....	
ĐIỀU KHIỂN VỊ TRÍ ĐỘNG CƠ DC QUA GIAO DIỆN LABVIEW .....	
3.1.KHÁI NIỆM.....	
3.1.1: Bản chất toán học của thuật toán PID.....	
3.2.GIỚI THIỆU VỀ Drive L298.....	
3.2.1. Giới thiệu.....	
3.2.2.Sơ đồ khối.....	
3.2.3.Các giá trị chịu đựng tối đa.....	
3.2.4.Sơ đồ chân.....	
3.2.5.Thông số về nhiệt độ.....	
3.2.6.Chức năng các chân.....	
3.2.7.Thông tin ứng dụng.....	
3.2.8.Đặc tính điện.....	
3.3. GIỚI THIỆU VỀ CARD USB-9090.....	
3.4. ĐIỀU KHIỂN PID CHO ĐỘNG CƠ DC .....	
3.4.1. Sơ đồ khối.....	
3.4.2. Sơ đồ kết nối hoàn chỉnh.....	
3.4.3. Chạy chương trình.....	
KẾT LUẬN	

# *Lời mở đầu*

- Nghiên cứu về cách thức sử dụng phần mềm LabVIEW, tìm hiểu về các ứng dụng của LabVIEW như: cách tạo giao diện, lập trình sơ đồ khối, làm thế nào để tạo thiết bị ảo (VI) và thiết bị ảo con (Sub VI) ...

- Nghiên cứu về card USB- 9090, cách truyền và nhận tín hiệu từ phần mềm LabVIEW tới card USB- 9090.

- Điều khiển PID cho động cơ DC trên cơ sở lập trình labview

## **Đối tượng phạm vi nghiên cứu**

Với đề tài:”**Điều khiển vị trí động cơ DC qua giao diện LABVIEW**” đối tượng nghiên cứu của em bao gồm: nghiên cứu về phần mềm LabVIEW, card USB-9090 , động cơ một chiều , điều khiển PID, giao tiếp với pc thông qua card USB-9090 phần mềm LabVIEW xử lý tín hiệu đưa vào, sau đó xuất tín hiệu ra để điều khiển động cơ DC

Ý nghĩa thực tiễn của đề án:

Nghiên cứu phần mềm LabVIEW với những ứng dụng rất rộng rãi, bởi vì bằng phần mềm chúng ta có thể thiết kế, điều khiển và kiểm tra như các phần cứng điều khiển và đo đạc. LabVIEW có khả năng kết nối tới rất nhiều thiết bị giúp tập hợp dữ liệu dễ dàng, đồng thời cung cấp tính kết nối tới hầu hết mọi thiết bị đo, vì vậy có thể dễ dàng kết hợp những ứng dụng LabVIEW mới vào các hệ thống hiện đại

## **Nội dung thực hiện đề tài gồm 3 chương**

Chương 1 : Tổng quan về phần mềm lập trình labVIEW

Chương 2 :Tìm hiểu động cơ một chiều các phương pháp điều khiển

Chương 3 :Điều khiển vị trí động cơ DC qua giao diện labview

Do thời gian có hạn và kiến thức chuyên môn cũng như thực tế còn hạn chế nên bản đề án này không tránh khỏi những thiếu sót. Kính mong nhận được sự chỉ bảo, góp ý của các thầy cô và các bạn để đề án này được hoàn thiện hơn.

Em xin chân thành cảm ơn sự giúp đỡ chỉ bảo tận tình của các thầy, cô, đặc biệt là thầy Ngô Quang Vĩ đã tạo điều kiện tốt nhất để em hoàn thành đồ án này. Em xin kính chúc các thầy cô luôn luôn mạnh khỏe để có thể tiếp tục dìu dắt nhiều thế hệ sinh viên.

Sinh viên thực hiện đề tài:

Phạm Ngọc Giang

## **CHƯƠNG 1 :**

# **TỔNG QUAN VỀ PHẦN MỀM LẬP TRÌNH LABVIEW**

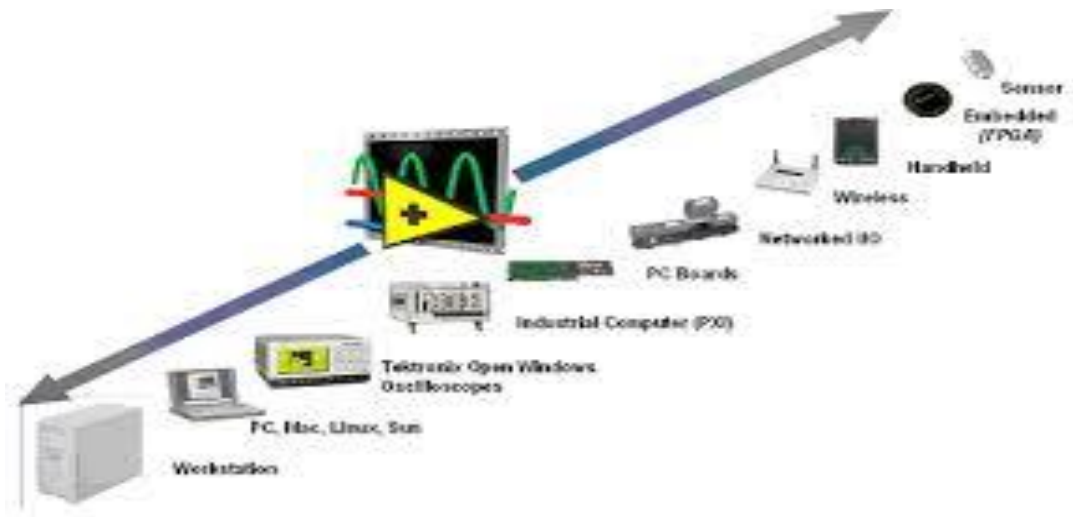
## **1.1: TỔNG QUAN VỀ LABVIEW**

### **1.1.1. LabVIEW là gì?**

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) là ngôn ngữ lập trình đồ họa sử dụng các biểu tượng (Icon) thay cho những dòng lệnh để tạo ứng dụng.

### **1.1.2. Vai trò của LabVIEW**

- Kiểm tra, đo kiểm và phân tích tín hiệu trong kỹ thuật (đo nhiệt độ, phân tích nhiệt độ trong ngày)
- Thu thập dữ liệu (Data Acquisition ), (thu thập các giá trị áp suất, cường độ, dòng điện,...)
- Điều khiển các thiết bị ( điều khiển động cơ DC, điều khiển nhiệt độ trong lò ...)
- Phân loại sản phẩm (dùng chương trình xử lý ảnh để phân biệt sản phẩm bị lỗi, phế phẩm)
- Báo cáo trong công nghiệp (thu thập, phân tích dữ liệu và báo cáo cho người quản lý ở rất xa thông qua giao thức truyền TCP/IP trong môi trường mạng Ethernet)
- Giao tiếp máy tính và truyền dẫn dữ liệu qua các cổng giao tiếp ( hỗ trợ hầu hết các chuẩn giao tiếp như USB, PCI, COM, RS-232, RS-485)



**Hình 1.1 Phạm vi ứng dụng của LabVIEW**

### **1.1.3. Các chức năng chính của LabVIEW**

- Thu thập tín hiệu từ các thiết bị bên ngoài như cảm biến nhiệt độ, hình ảnh từ webcam, vận tốc của động cơ...
- Giao tiếp với các thiết bị ngoại vi thông qua nhiều chuẩn giao tiếp như: RS232, RS485, USB, PCI, TCP/IP, Ethernet.
- Mô phỏng và xử lý các tín hiệu thu nhận được để phục vụ các mục đích nghiên cứu hay mục đích của hệ thống mà người lập trình mong muốn.
- Xây dựng các giao diện người dùng một cách nhanh chóng và thẩm mỹ hơn nhiều lần so với các ngôn ngữ như VB, Matlab, Visual C...
- Cho phép kết hợp với nhiều ngôn ngữ truyền thống như C, C++...
- Cho phép thực hiện các thuật toán điều khiển như PID, Logic mờ (Fuzzy).

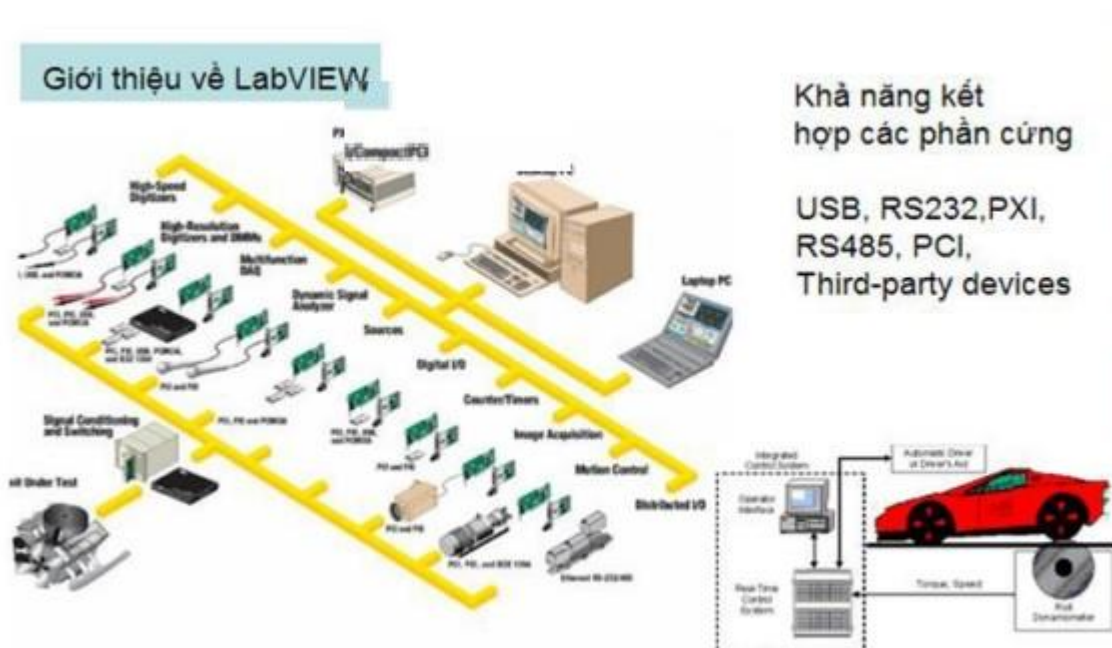
### **1.1.4. Phần mềm nhúng vào LabVIEW**

- Wolfram Research Mathematica.
- Microsoft Excel.
- The MathWorks MATLAB and Simulink.
- MathSoft MathCAD.
- Electronic Workbench MultiSim.
- Texas Instruments Code Composer Studio.

- Ansoft RF circuit design software
- Microsoft Access.
- Microsoft SQL Server.
- Oracle.

### 1.1.5. Các giao thức kết nối

- Ethernet
- CAN
- DeviceNet
- USB
- IEEE 1394;
- RS-232
- GPIB
- RS-485



Hình 1.2 Các giao thức kết nối của LabVIEW

## **1.1.6. Các Module và bộ công cụ LabVIEW**

### **1.1.6.1. Các module LabVIEW**

Để tăng cường sức mạnh và mở rộng khả năng của bộ phần mềm phát triển LabVIEW, NI cung cấp thêm các module hỗ trợ đến nhiều loại phần cứng nhúng khác nhau:

- Module thời gian thực (LabVIEW Real-Time Module).
- Module FPGA.
- Module điều khiển giám sát và ghi dữ liệu (LabVIEW Datalogging and Supervisory Control Module).
- Module biểu đồ trạng thái (LabVIEW Statechart Module).
- Module mô phỏng và thiết kế bộ điều khiển (LabVIEW Control Design and Simulation Module).
- Module phát triển thị giác (NI Vision Development Module).
- Module cho màn hình cảm ứng và PDA (LabVIEW PDA and LabVIEW Touch Panel Module).
- LabVIEW DSP Module ( xử lý tín hiệu số )

### **1.1.6.2. Các bộ công cụ LabVIEW**

NI cũng thêm vào LabVIEW các bộ công cụ để đem lại các tiện ích khác nhau như: tạo báo cáo, phân tích nâng cao, thông tin liên lạc cơ sở dữ liệu, phân tích âm thanh và rung động.

- Bộ công cụ kết nối cơ sở dữ liệu (LabVIEW Database Connectivity Toolkit).
- Bộ công cụ xử lý tín hiệu nâng cao (LabVIEW Advanced Signal Processing Toolkit).
- Bộ đo lường âm thanh và rung động (LabVIEW Sound and Vibration Measurement Suite).



- Bộ công cụ nhận dạng hệ thống (LabVIEW System Identification Toolkit).
- Bộ công cụ tương tác mô phỏng (LabVIEW Simulation Interface Toolkit).
- Bộ công cụ theo dõi thực thi thời gian thực (LabVIEW Real-Time Execution Trace Toolkit).
- Bộ công cụ kết nối Internet (LabVIEW Internet Toolkit).
- Bộ công cụ điều biến (LabVIEW Modulation Toolkit).
- Bộ công cụ điều khiển PID (LabVIEW PID Control Toolkit).
- Bộ công cụ thiết kế bộ lọc số (LabVIEW Digital Filter Design Toolkit).

### **1.1.7. LabVIEW làm việc như thế nào?**

LabVIEW được biết đến như là một ngôn ngữ lập trình với khái niệm hoàn toàn khác so với các ngôn ngữ lập trình truyền thống như ngôn ngữ C,

Pascal... Bằng cách diễn đạt cú pháp thông qua các hình ảnh trực quan trong môi trường soạn thảo có sẵn hàng ngàn thư viện, hàm và cấu trúc lập trình, LabVIEW đã được gọi với tên khác là lập trình G (viết tắt của Graphical). Những chương trình LabVIEW được gọi là những thiết bị ảo (Virtual Instruments – VIs), bởi vì hình dạng và cách hoạt động giống với những thiết bị vật lý, chẳng hạn như máy nghiệm dao động, máy hiện sóng...

Trong LabVIEW, bạn xây dựng giao diện người dùng bằng cách sử dụng một bộ các công cụ và đối tượng, và cửa sổ Front panel được xem như là giao diện người dùng. Còn cửa sổ Block diagram chứa các hàm thao tác là các biểu tượng đồ họa, nơi mà dòng dữ liệu thực thi.

## **1.2. CÁC THÀNH PHẦN CỦA LABVIEW**

LabVIEW bao gồm các thư viện của các hàm chức năng và các công cụ phát triển được thiết kế đặc biệt dành cho thiết bị điều khiển. Các chương trình LabVIEW được gọi là những dụng cụ ảo bởi vì sự xuất hiện và hoạt động của chúng mô phỏng các dụng cụ thực tế. Các VI có cả 2 tương tác đó là: một tương tác

giao diện người dùng và một mã nguồn tương đương, và truy nhập các tham số từ các VI tầng cao.

LabVIEW gồm có 3 thành phần chính đó là: bảng giao diện ( The Front Panel), sơ đồ khối (The Block Diagram) và biểu tượng & đầu nối (The Icon - Connect).

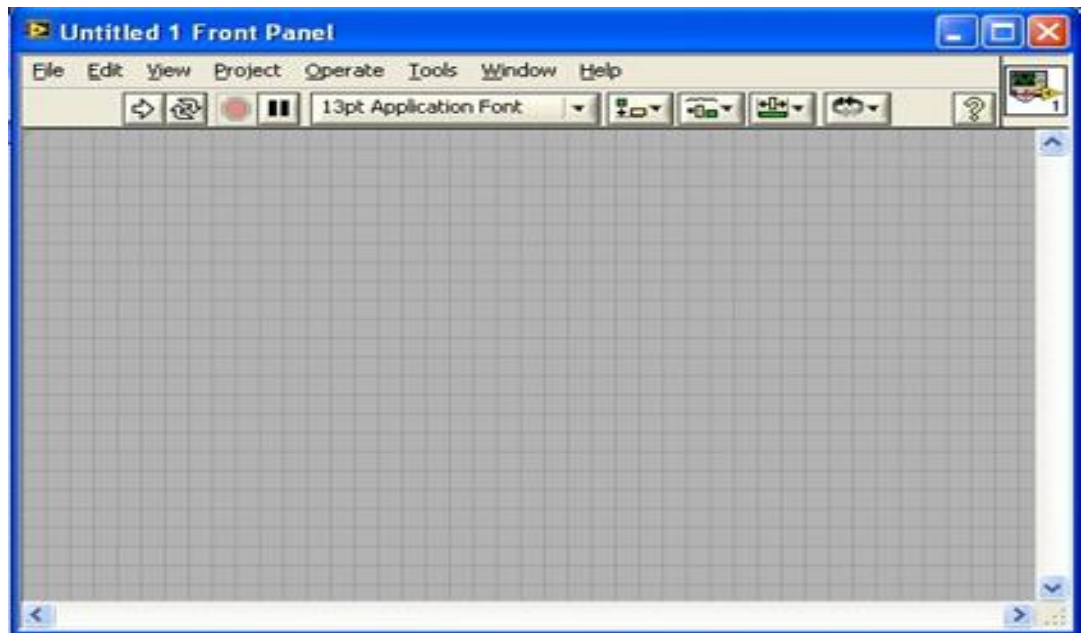
### **1.2.1. Bảng giao diện (The Front panel)**

Front Panel là giao diện mà người sử dụng hệ thống nhìn thấy. Các VI bao gồm một giao diện người dùng có tính tương tác mà được gọi là bảng giao diện, vì nó mô phỏng mặt trước của một dụng cụ vật lý. Bảng giao diện có thể bao gồm các nút, các nút đẩy, các đồ thị và các dụng cụ chỉ thị và điều khiển khác.

Bạn nhập vào dữ liệu sử dụng bàn phím và chuột rồi sau đó quan sát các kết quả trên màn hình máy tính.

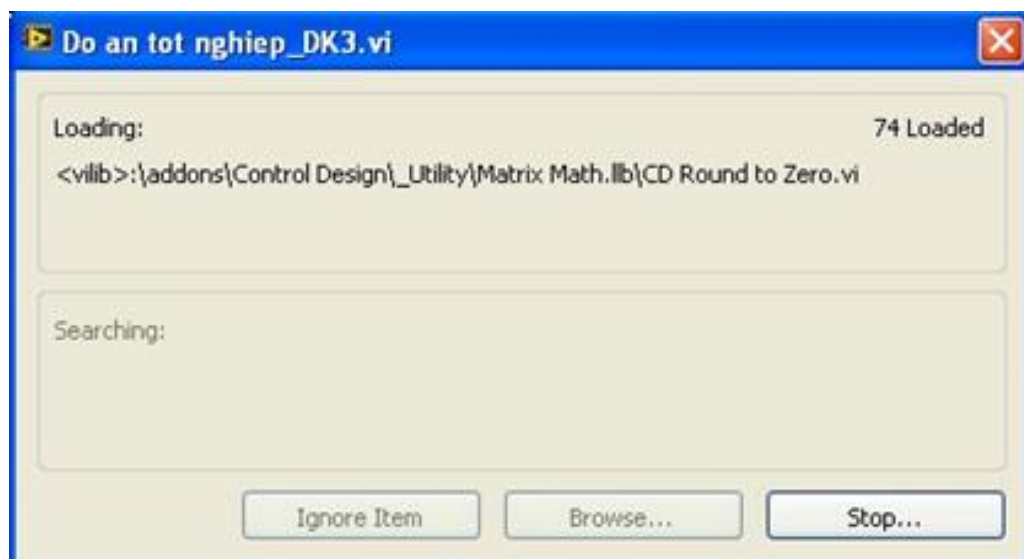
Vào **Start>>All Programs>> National Instruments LabVIEW** một cửa sổ LabVIEW xuất hiện. Bạn tiếp tục chọn Evaluate và cửa sổ Getting Started sẽ xuất hiện ngay sau đó. Bạn chọn Blank VI để hiển thị bảng giao diện hoặc bạn có thể chọn New và sau đó hộp thoại New xuất hiện và trong hộp thoại đó mặc định con trỏ ở danh mục Blank VI. Để hiển thị bảng giao diện bạn chỉ cần kích vào nút OK ở phía góc phải dưới. Cả hai cách trên đều để mở bảng giao diện mới để bạn có thể xây dựng một VI mới hoàn toàn.

Ngoài ra bạn có thể mở một bảng giao diện có sẵn trong LabVIEW bằng cách trong hộp thoại **New**, từ mục **Create New**, lựa chọn **VI>>From template>>Tutorial (Getting Started)>>Generate and Display**. Và sau đó kích nút OK để hiển thị bảng giao diện. Bảng giao diện sẽ xuất hiện như hình 1.3



**Hình 1.3. Bảng giao diện**

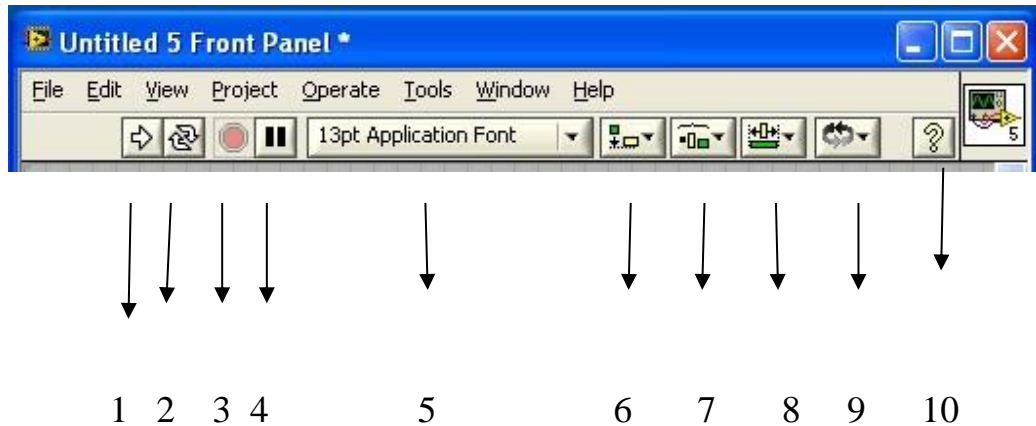
Ta cũng có thể mở bảng giao diện của một VI có sẵn trong thư viện LabVIEW bằng cách trong hộp thoại bảng giao diện vào **File>>Open** sau đó kích đúp vào các ví dụ có sẵn. Trong khi VI đang tải, một hộp thoại xuất hiện, cái mà mô tả tên của VI hiện thời đang tải, tên của điều khiển cứng mà VI được định vị trên đó, các thư mục và các đường dẫn đang được tìm kiếm, và số lượng VI trong quá trình tải. Hộp thoại xuất hiện như hình 1.4 bên dưới:



## Hình 1.4 Mô tả tên của VI hiện thời đang tải

Trong bảng giao diện bao gồm một thanh công cụ của các nút lệnh và các dụng cụ chỉ báo trạng thái mà bạn sử dụng cho quá trình chạy và xử lý các VI.

Nó cũng bao gồm những tùy chọn phong và các tùy chọn phân phối và sắp thành hàng cho việc soạn thảo các VI




Hình 1.5 Thanh công cụ giao diện

Trong đó:

1. Nút chạy chương trình (thanh không sáng – bị vỡ: lỗi, phải sửa lại chương trình)
2. Nút chạy lặp
3. Nút dừng cưỡng ép chương trình
4. Nút tạm dừng
5. Text setting (màu sắc, định dạng, kích thước- phong)
6. Gióng đều đối tượng theo hàng dọc và ngang
7. Phân bố các đối tượng
8. Thay đổi kích thước các đối tượng
9. Lệnh bổ sung
10. Cửa sổ trợ giúp

### Các lưu ý trong hoạt động VI

1. Trong bảng giao diện, chạy VI bằng cách kích vào nút chạy 

trên thanh công cụ

Nút chạy thay đổi để chỉ báo rằng VI đang chạy



2. Sử dụng công cụ Operating để thay đổi các giá trị giới hạn cao và thấp. Đầu tiên chiếu sáng giá trị cũ, sau đó bằng việc tiếp tục nhấn đúp giá trị bạn muốn thay đổi, hoặc kích và kéo ngang qua giá trị với công cụ Labeling. Khi nào giá trị ban đầu được chiếu sáng, nhập một giá trị mới và nhấn <Enter>. Bạn cũng có thể kích trên nút nhập vào trong thanh công cụ, hoặc kích chuột trong một vùng mở của cửa sổ để nhập vào giá trị mới.

3. Thay đổi điều khiển trượt Update Period, bằng cách đặt công cụ Operating trên thanh trượt và kéo của nó tới một vị trí mới.

4. Thực hành điều chỉnh những điều khiển khác.

5. Dừng VI bằng cách kích vào công tắc chuyển đổi thu nhận. VI không thể dừng ngay lập tức bởi vì VI còn phải đợi cho phương trình hay sự phân tích cuối cùng đặt tới hoàn thành thao tác.

Lưu ý: Ta nên đợi cho một VI thực thi hoàn toàn hoặc nên thiết kế một cách thức để dừng nó, chẳng hạn như đặt một công tắc trên giao diện.

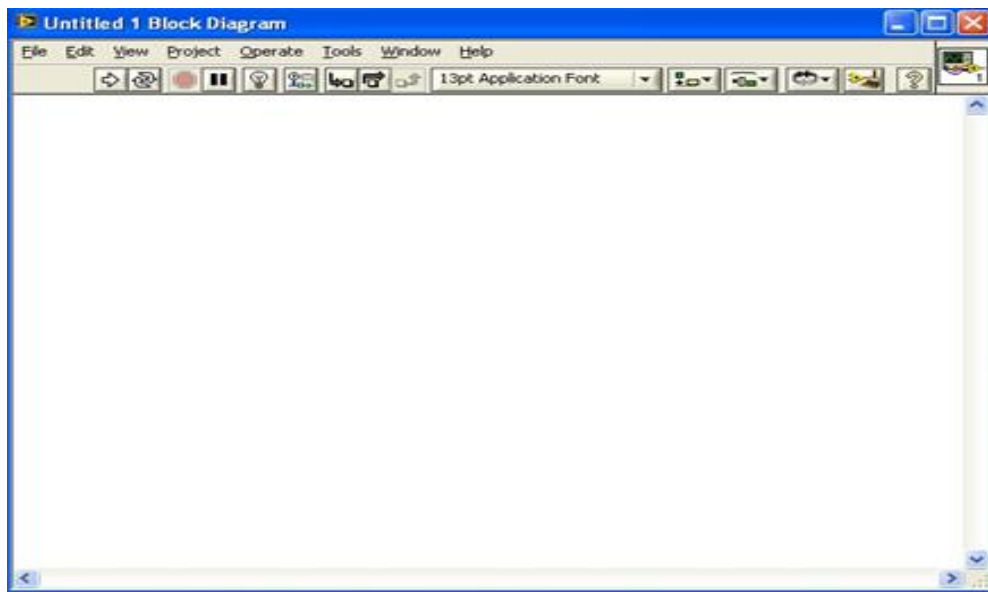
Mặc dù VI dừng nếu ta kích vào nút dừng trên thanh công cụ, đây không phải là cách tốt nhất để dừng các VI lại bởi vì nút dừng dừng chương trình ngay lập tức. Điều này có thể làm gián đoạn các hàm chức năng I/O, và vì thế nó có thể dẫn đến tình trạng không mong muốn.

### **1.2.2. Sơ đồ khối (The Block Diagram)**

Sơ đồ khối chứa đựng mã nguồn đồ thị, thường biết như là mã G hoặc mã sơ đồ khối, cho đến VI chạy như thế nào. Mã sơ đồ khối sử dụng đồ thị biểu diễn các chức năng để điều khiển các đối tượng trên giao diện. Các đối tượng trên giao diện xuất hiện như biểu tượng các thiết bị trên sơ đồ khối. Kết nối điều khiển và các đầu của dụng cụ chỉ thị tới Express VIs, VIs, và các chức năng. Dữ liệu chuyển thông qua dây dẫn từ các điều khiển đến các VI và các hàm chức năng, từ các VI và các

hàm chức năng đến các VI và các hàm chức năng khác, và từ các VI và các hàm chức năng đến các dụng cụ chỉ thị. Sự di chuyển của dữ liệu thông qua các nút trên sơ đồ khối xác định mệnh lệnh thực hiện của các VI và các hàm chức năng. Sự di chuyển dữ liệu này được biết như lưu đồ lập trình.

1. Mở sơ đồ khối của một hệ thống nào đó bằng cách chọn **Window>>Show Block Diagram**. Hoặc cũng có thể gọi tới sơ đồ khối bằng cách trên bảng giao diện nhấn <Ctrl E>. Sơ đồ khối có nền màu trắng như hình 1.6 dưới đây:



### 1.2.3. Viết chương trình trên nền LabVIEW

Chương trình của LabVIEW được chia làm 2 phần: Viết trên Front Panel và Block Diagram

Phần Front Panel là phần giao diện thực thi chương trình điều khiển, tạo cách lệnh điều khiển đối tượng của mình, là phần Control để điều khiển cánh tay.

– Control là các đối tượng đặt trên Front Panel để cung cấp dữ liệu cho chương trình. Nó tương tự như đầu vào cung cấp dữ liệu.

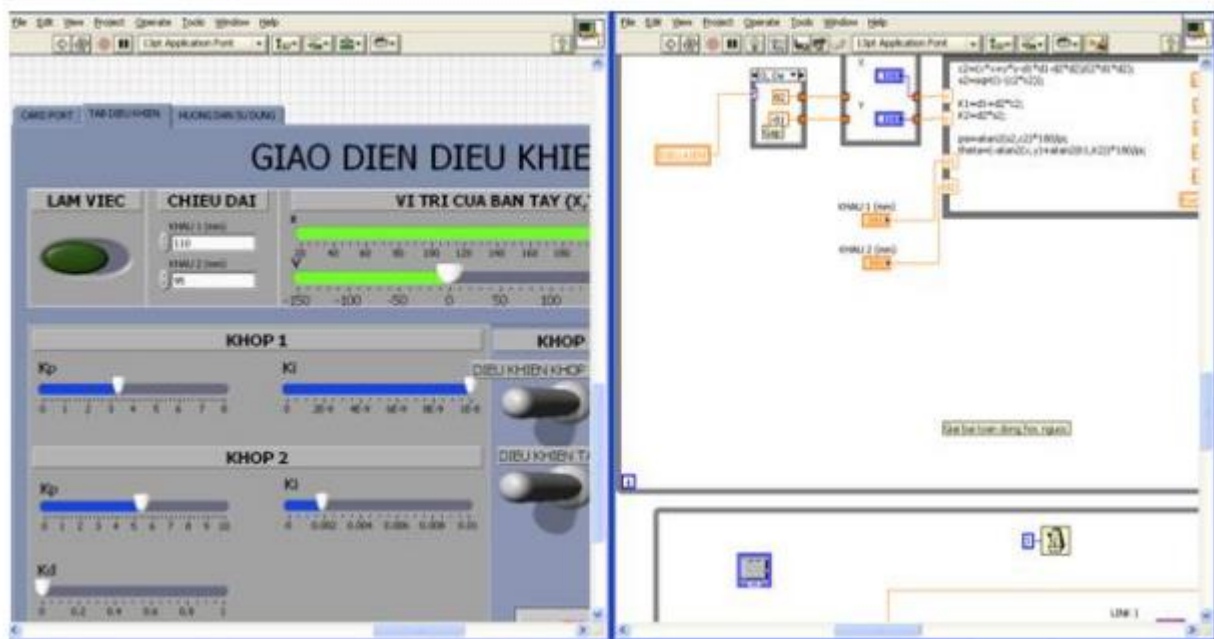
– Indicator là đối tượng được đặt trên Front Panel dùng để hiển thị kết quả nó tương tự như một đầu vào của chương trình.

Phần Block Diagram là phần sơ đồ khối, tạo các khối chương trình lỗi để điều khiển (Hình 1.7). Block Diagram là phần để viết thuật giải cho robot, các phương trình động học ngược, thuật giải PID cho các động cơ và cách mô phỏng 3D cho cánh tay robot.

– Terminal: là các cổng dữ liệu truyền qua giữa Block Diagram và Front Panel và giữa các Node trong Block Diagram. Các Terminal nằm ở dưới dạng các Icon của các Function.

– Node: là các phần tử thực thi chương trình, chúng tương tự các mệnh đề, toán tử hàm và các chương trình con trong các ngôn ngữ lập trình thông thường.

– Wires: là các dây nối dữ liệu giữ các node

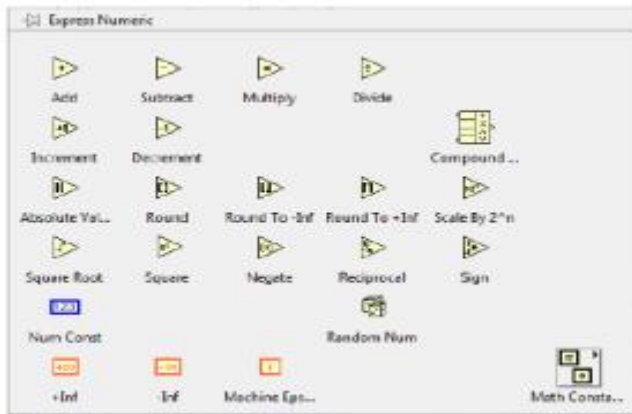


**Hình 1.7: Chương trình điều khiển Front Panel và Block Diagram**

+ Giao diện điều khiển bằng máy tính

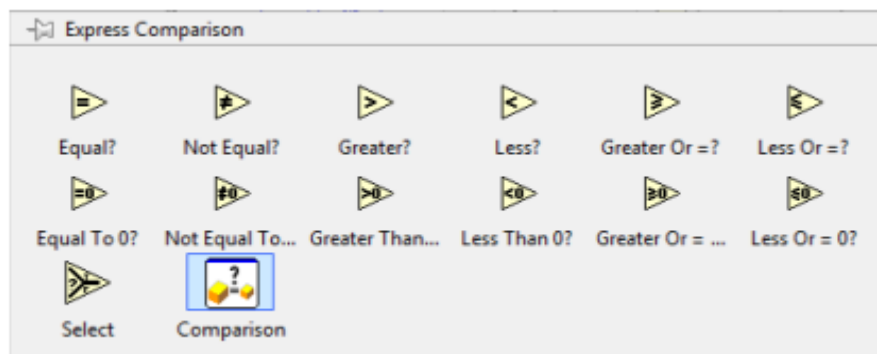
Các khối được sử dụng trong chương trình

– Các kí hiệu toán học: Functions/Express/Arithmetic & Comparsion/ Numeric (Hình 1.8).



**Hình 1.8: Các kí hiệu toán học**

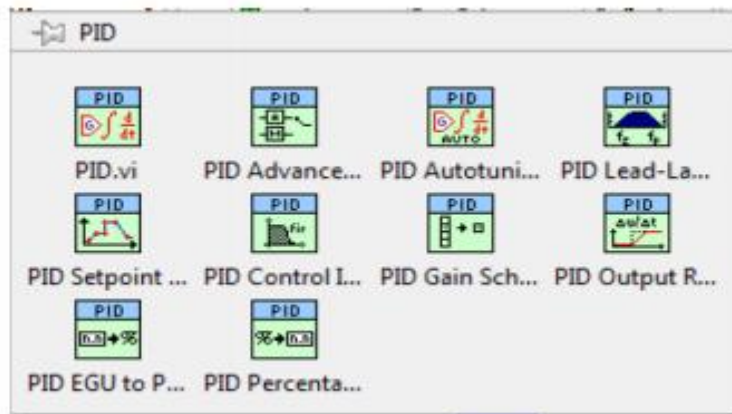
- Các toán tử so sánh ( Hình 1.9).



**Hình 1.9: Khối toán tử so sánh**

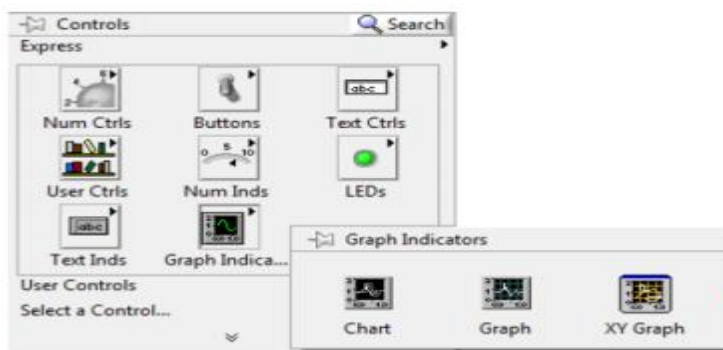
- Khối PID: control design & Simulation / PID (Hình 1.10).





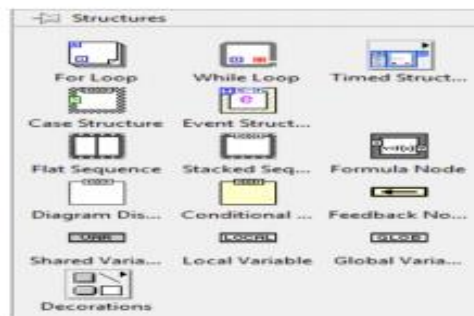
**Hình 1.10: Khối PID**

– Khối biểu diễn đồ thị: Controls /Grap Indicators (Hình 1.11).



**Hình 1.11: Khối biểu diễn đồ thị**

-Các hàm: Functions/Programming/Structures (Hình 1.12).



**Hình 1.12: Khối các hàm**

- Các biến: Controls/Numeric controls (Hình 1.13).



**Hình 1.13: Khối các biến**

- Định dạng kiểu dữ liệu cho các biến: Nhấp phải chuột vào biến numeric control/Representation/chọn kiểu dữ liệu (Hình 1.14)



**Hình 1.14: Khối các kiểu dữ liệu**

## CHƯƠNG 2

# TÌM HIỂU ĐỘNG CƠ MỘT CHIỀU VÀ CÁC PHƯƠNG PHÁP ĐIỀU KHIỂN

### 2.1 ĐỘNG CƠ ĐIỆN 1 CHIỀU THÔNG THƯỜNG

\*Cấu tạo

Cấu tạo bên ngoài:



**Hình 2.1** cấu tạo bên ngoài của động cơ điện một chiều

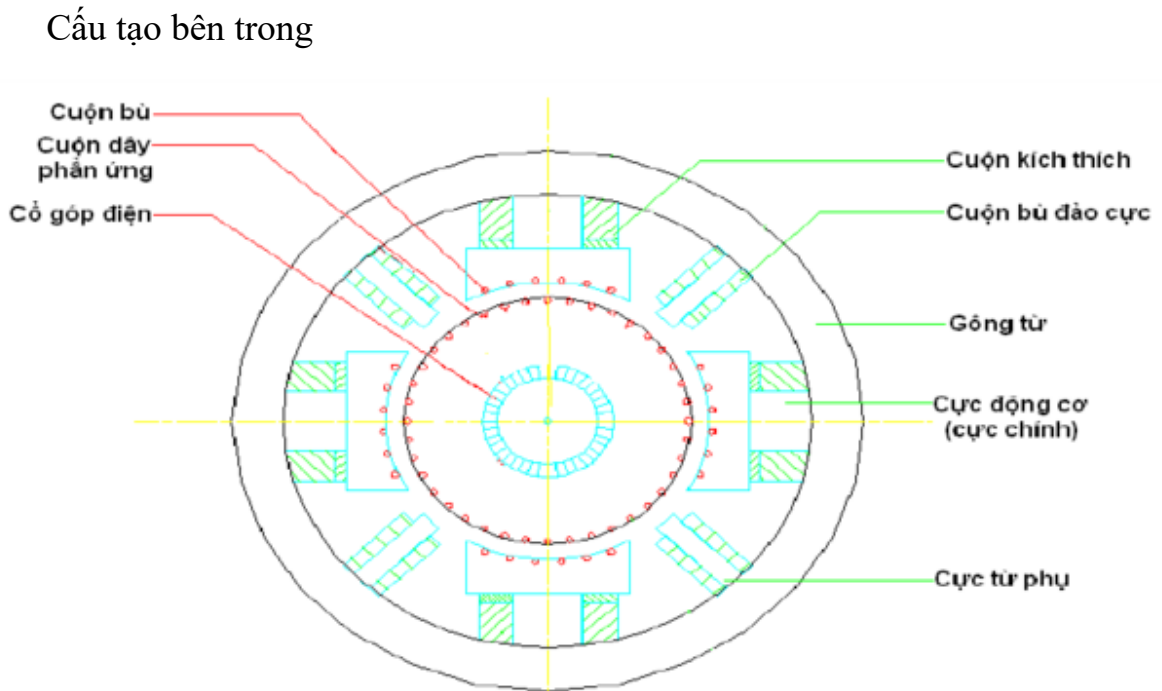
Chú thích

**Cáp nguồn** : là 2 dây dẫn điện ( một chiều) từ nguồn phát tới nuôi động cơ, nguồn điện phát một chiều thương có điện áp từ 6Vol, 12 Vol đến hàng ngàn Vol

**Vỏ kim loại** : Là vỏ bọc động cơ, tác dụng bảo vệ động cơ tránh những tác động của môi trường .Đồng thời cũng để định vị động cơ vào vị trí làm việc

**Trục động cơ** : Để truyền mô men quay của động cơ tới cơ cấu công tắc

**Nắp máy** : để bảo vệ máy khỏi những vật ngoài rơi vào làm hư hỏng dây quấn và an toàn cho người khỏi chạm vào điện. Trong máy điện nhỏ vừa nắp máy còn có tác dụng làm giá đỡ ổ bi. Nắp máy thường làm bằng gang.



**Hình 2.2 Cấu tạo bên trong của động cơ điện một chiều**

**Gông từ** : Gông từ dùng làm mạch từ nối liền các cực từ, đồng thời làm vỏ máy .

Trong động cơ điện nhỏ vừa thường dùng thép dày uổng và hàn lại. Trong máy điện lớn thường dùng thép đúc. Có khi trong động cơ điện nhỏ dùng gang làm vỏ máy

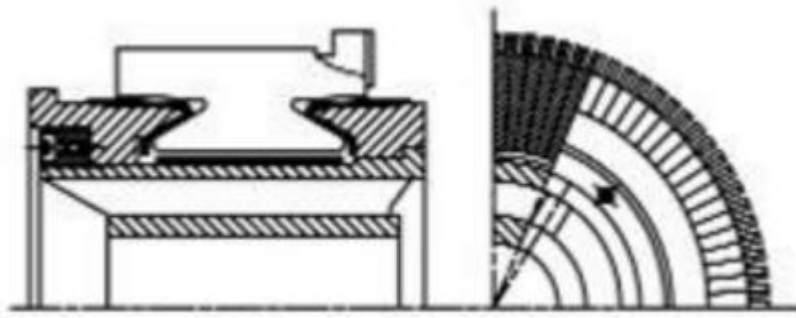
**Cực động cơ (cực chính)** : Là bô phân sinh ra từ trường gồm có lõi sắt cực từ và dây quấn kích từ lồng ngoài lõi sắt cực từ. Lõi sắt cực từ làm bằng những lá thép kỹ thuật điện hay thép các bon dày 0.5 đến 1mm ép lại và tán chặt. Trong động cơ điện nhỏ có thể dùng thép khối . Cực từ được gắn chặt vào vỏ máy nhờ các bulông

Dây quấn kích từ được quấn bằng dây đồng bọc cách điện

**Cực từ phụ** : được đặt bên các cực từ chính và dùng để cải thiện đổi chiều. Lõi thép cực từ phụ thường làm bằng thép khối và trên thân cực từ phụ có đặt dây quấn mà cấu tạo giống như dây quấn cực từ chính. Cực từ phụ được gắn vào vỏ máy nhờ những bulông

**Cuộn bù** : có tác dụng khử méo dạng từ thông phân bố trên bề mặt Rôto do ảnh hưởng của cuộn dây phản ứng

**Cổ góp điện** :



Hình 2.3 cấu tạo của cổ góp điện

Là cụm chi tiết phức tạp nhất của máy điện một chiều vì trong kết cấu của nó có rất nhiều là đồng ( được gọi là lam đồng) xếp xen kẽ với các tấm mi ca cứng tạo thành vành tròn( được gọi là vành góp). Các chi tiết của cổ góp có hình dạng rất phức tạp, ghép lại với nhau bằng mặt côn được chế tạo với các yêu cầu nghiêm ngặt về bề mặt gia công cùng các kích thước có cấp chính xác cao.

Cổ góp điện có nhiều kết cấu khác nhau. ở máy điện một chiều từ vài chục kW trở lên, cổ góp có kết cấu bạc ép ( xem hình 2.3) . cổ góp được tạo thành từ việc ghép nhiều chi tiết ( có thể lên đến trên 2000 chi tiết), bằng nhiều loại vật liệu khác nhau như đồng đỏ.... mi ca cứng, mi ca mềm , thép 45, thép CT, sơn cánh kiến, bằng sợi tổng hợp ... Khi làm việc, cổ góp chịu tác động rất lớn của lực ly tâm , nhiệt độ cao có thể lên đến  $1300^{\circ}\text{C}$  phát sinh từ lực ma sát của viên than lên vành

góp từ tia lửa do tiếp xúc của viên than không ổn định, do dòng dao chiều và do độ lệch đường trung tính hình học của cụm chổi than, một chi tiết chế tạo không đạt yêu cầu kỹ thuật, thực hiện không chuẩn xác một bước công nghệ, cổ góp sẽ bị phá hủy khi làm việc, dẫn đến sự cố của máy điện một chiều

Phân loại : theo kiểu kích từ thì động cơ một chiều được phân ra những loại như sau :

Động cơ một chiều kích từ độc lập

Động cơ một chiều kích từ nối tiếp

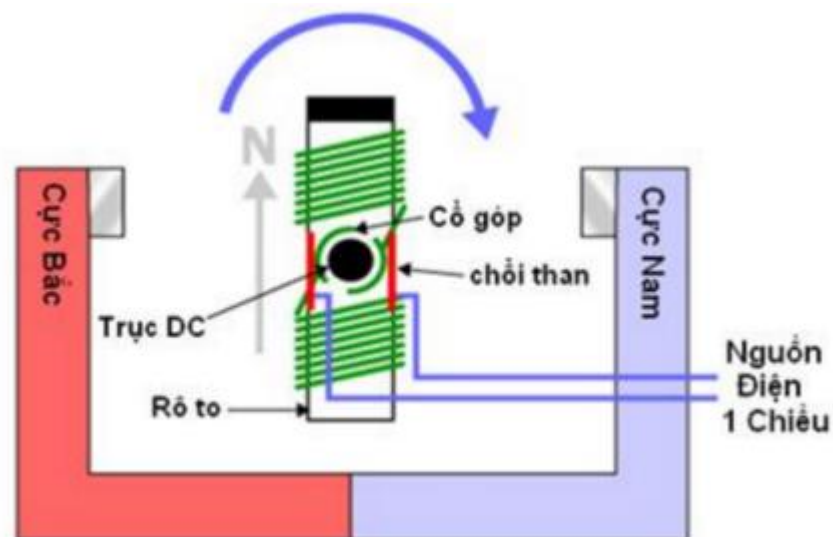
Động cơ một chiều kích từ song song

Động cơ một chiều kích từ bằng nam châm vĩnh cửu.

## 2.2 ĐỘNG CƠ ĐIỆN MỘT CHIỀU KÍCH TỪ BẰNG NAM CHÂM VĨNH CỬU (DCVC)

### 2.2.1 : Cấu tạo

Động cơ một chiều kích từ bằng nam châm vĩnh cửu (DCVC) – là trường hợp đặc biệt của động cơ một chiều kích từ độc lập. Cuộn dây kích từ trên Stator được loại bỏ và thay bằng một cặp nam châm vĩnh cửu



Hình 2.4 Cấu tạo động cơ điện một chiều kích từ bằng nam châm vĩnh cửu

Một DCVC thì bao gồm 6 phần như sau

Phần ứng hay rô to ;

Cổ góp ;

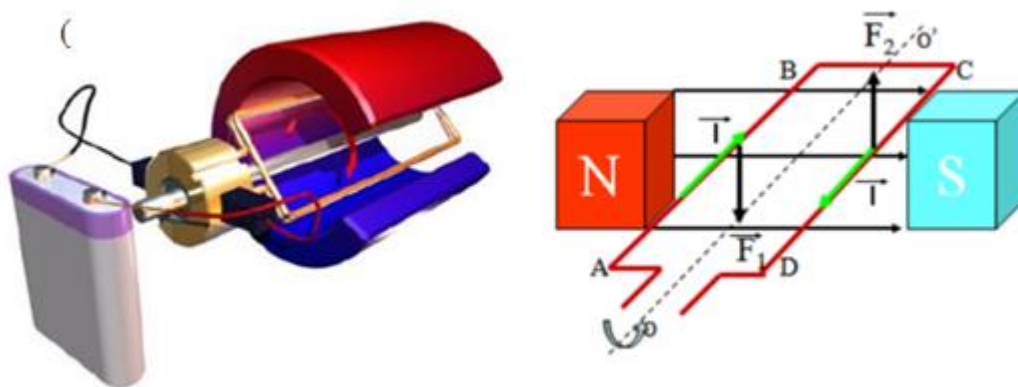
Chổi than ;

Trục động cơ ;

Miếng nam châm tạo từ trường ;

Bộ phận cung cấp dòng điện một chiều ;

### 2.2.2 : Nguyên lý hoạt động của DCVC

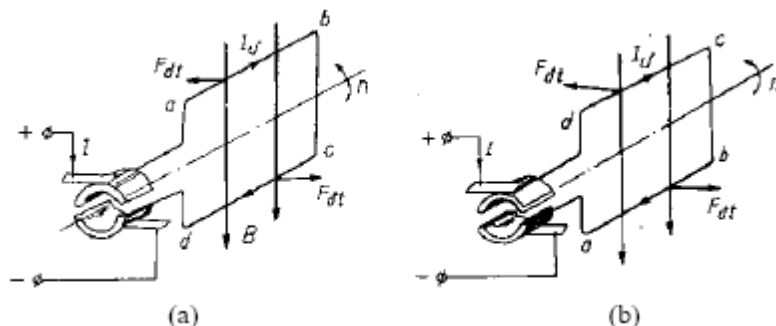


Khi động cơ được cấp điện, dòng điện đi vào cuộn dây( như 1 khung dây) của Rotor thông qua cơ cấu chổi than – cổ góp. Theo nguyên tắc bàn tay trái sẽ có ngẫu lực điện từ đặt lên khung dây và làm cho khung dây quay tức rotor quay, động cơ hoạt động. Theo hình bên, cuộn dây quấn trên rotor được mô tả như khung dây ABCD. Đặt bàn tay trái sao cho các đường sức từ hướng vào lòng bàn tay, chiều từ cổ tay đến ngón tay giữa hướng theo chiều dòng điện thì ngón tay cái choãi ra  $90^0$  chỉ chiều của lực điện từ tác dụng lên dây dẫn. Trục của rotor theo hình trên sẽ quay theo chiều ngược chiều kim đồng hồ( nhìn từ ngoài vào trong)

### 2.2.3 Nguyên lý làm việc và phân loại động cơ điện một chiều

*Nguyên lý làm việc động cơ điện một chiều:* khi cho điện áp một chiều U vào hai chổi điện A và B, trong dây quấn phần ứng có dòng điện. Các thanh dẫn ab và cd mang dòng điện nằm trong từ trường sẽ chịu lực tác dụng tương hỗ lên nhau tạo

nên mômen tác dụng lên rôto, làm quay rôto. Chiều lực tác dụng được xác định theo quy tắc bàn tay trái



Hình 2.5. Mô tả nguyên lý làm việc của động cơ điện một chiều

Khi phần ứng quay được nửa vòng, vị trí thanh dẫn ab, cd đổi chỗ nhau (hình 2.5 b), nhờ có phiến góp đổi chiều dòng điện, nên dòng điện một chiều biến đổi thành dòng điện xoay chiều đưa vào dây quấn phần ứng, giữ cho chiều lực tác dụng không đổi, do đó lực tác dụng lên rôto cũng theo một chiều nhất định, đảm bảo động cơ có chiều quay không đổi.

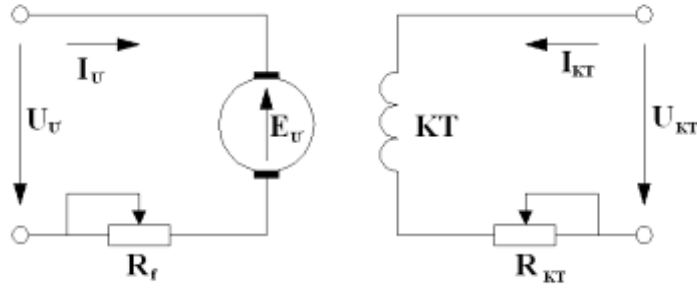
#### 2.2.4. Phương trình đặc tính cơ động cơ điện một chiều

Do đối tượng sử dụng trong luận văn là động cơ điện một chiều kích từ độc lập nên dưới đây tác giả tập trung trình bày cách xây dựng đặc tính cơ của động cơ một chiều kích từ độc lập.

##### Xây dựng phương trình đặc tính cơ

Để điều khiển được tốc độ động cơ điện một chiều kích từ độc lập thì ta phải phân tích, tìm các mối quan hệ giữa tốc độ với các thông số khác của động cơ để từ đó đưa ra phương pháp điều khiển. Động cơ điện một chiều kích từ độc lập thì dòng kích từ độc lập với dòng phần ứng và được cấp bởi hai nguồn một chiều độc lập với nhau [2].





Hình 2.6. Sơ đồ động cơ điện một chiều kích từ độc lập

Theo hình 2.6 ta viết phương trình cân bằng điện áp của mạch phản ứng [2]

$$U_v = E_v + (R_r + R_f) \cdot I_v \quad (2.1)$$

Trong đó:  $U_v$  : Điện áp đặt lên phần ứng động cơ (V).

$E_v$  : Sức điện động phần ứng (V)

$R_r$  : Điện trở của mạch phần ứng ( $\Omega$ )

$R_f$  : Điện trở phụ trong mạch phần ứng ( $\Omega$ )

$I_v$  : Dòng điện mạch phần ứng (A)

Với:  $R_r = r_r + r_{cf} + r_b + r_{ct}$

Trong đó:  $r_r$  : điện trở cuộn dây phần ứng ( $\Omega$ )

$r_{cf}$  : điện trở cuộn cực từ phụ ( $\Omega$ )

$r_b$  : điện trở cuộn bù ( $\Omega$ )

$r_{ct}$  : điện trở tiếp xúc chổi điện ( $\Omega$ )

Sức điện động  $E_v$  của phần ứng động cơ được xác định theo biểu thức (2.2) [2]

$$E_{o,,} = K \cdot \omega = \frac{P \cdot N}{2 \cdot \pi \cdot a} \cdot \Phi \cdot \omega \quad (2.2)$$

Trong đó:  $K = \frac{P \cdot N}{2 \cdot \pi \cdot a}$  hệ số cấu tạo của động cơ

$P$  : Số đôi cực

$N$  : Số thanh dẫn tác dụng

$a$  : Số đôi mạch nhánh song song

$\Phi$  : Từ thông kích từ dưới mỗi cực từ (wb)

$\omega$  : Tốc độ góc (rad/s)

Nếu biểu diễn sức điện động theo tốc độ quay  $n$  (vòng/ phút) [2]:

$$E_u = K_e \cdot \Phi \cdot n \quad (2.3)$$

Mà 
$$\omega = \frac{2 \cdot \pi \cdot n}{60}$$

Thay (2.4) vào (2.2) ta có

$$E_u = \frac{P \cdot N}{60 \cdot a} \cdot \Phi \cdot n$$

$$\Rightarrow K_e = \frac{P \cdot N}{60 \cdot a} = \frac{K}{9,55} \approx 0,105$$

$K$ : Hệ số sức điện động của động cơ

Thay (2.1) vào (2.2) và biến đổi ta được [2]:

$$\omega = \frac{U}{k \cdot \phi} - \frac{R_u + R_f}{k \cdot \phi} \cdot I_u \quad (2.4)$$

Biểu thức (2.6) là phương trình đặc tính cơ điện của động cơ.

Mặt khác, mômen điện từ:  $M_{dt} = K \cdot \Phi \cdot I_u \quad (2.6)$

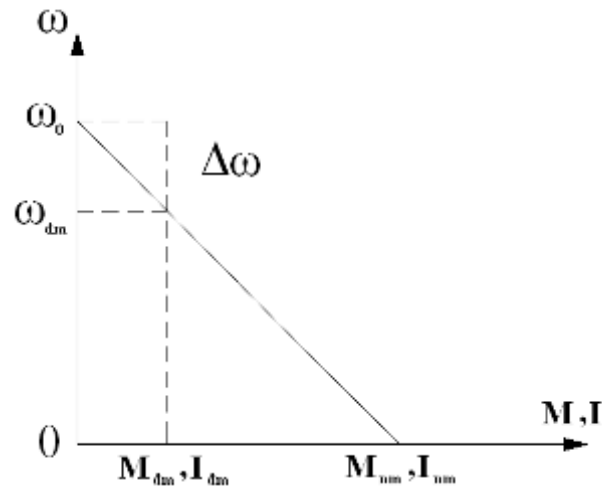
Nếu bỏ qua mọi tổn hao thì mômen cơ trên trục của động cơ chính bằng mômen điện từ, ta ký hiệu là  $M$ , tức là

$$M_{dt} = M_{cv} = M$$

Thay  $M_{dt} = M_{cv} = M$  vào phương trình (2.4) ta thu được phương trình đặc tính cơ của động cơ [2]:

$$\omega = \frac{U}{k \cdot \phi} - \frac{R_u + R_f}{(k \cdot \phi)^2} \cdot M \quad (2.7)$$

Biểu thức (2.7) là phương trình đặc tính cơ của động cơ điện một chiều kích từ độc lập. Nếu ta xem  $\Phi = \text{const}$  thì quan hệ  $\omega = f(M, I)$  là tuyến tính.



Hình 2.7. Đồ thị đặc tính cơ của động cơ điện một chiều kích từ độc lập

Từ hình 2.7 ta thấy : khi  $I_r = 0$  hoặc  $M = 0$  ta có:

$$\omega = \frac{U}{k.\phi} = \omega_0 \quad (2.8)$$

với  $\omega_0$  : tốc độ không tải lý tưởng của động cơ điện một chiều kích từ độc lập

Khi  $\omega = \omega_0$  ta có:

$$I_r = \frac{U}{R_u + R_f} = I_{nm} \quad (2.9)$$

và  $M = k. \Phi. I_{nm} = M_{nm} \quad (2.10)$

$I_{nm}$ ,  $M_{nm}$  : là dòng điện ngắn mạch và mômen ngắn mạch.

Mặt khác ta có:  $I_r = \frac{M}{k.\phi}$ ,  $R = R_u + R_f$ ,  $\omega_0 = \frac{U}{k.\phi}$  nên phương trình (2.7) có

thể viết dưới dạng:

$$\omega = \frac{U}{k.\phi} - \frac{R_u + R_f}{(k.\phi)^2}.M = \omega_0 - \Delta\omega \quad (2.11)$$

## 2.3 ĐIỀU KHIỂN TỐC ĐỘ ĐỘNG CƠ MỘT CHIỀU

### 2.3.1. Các phương pháp điều khiển động cơ điện một chiều

Điều chỉnh tốc độ động cơ một chiều ta có thể thực hiện bằng ba phương pháp điều khiển sau

Điều khiển tốc độ bằng cách thay đổi điện trở phụ phản ứng

Điều khiển tốc độ bằng cách thay đổi từ thông kích từ

Điều khiển tốc độ bằng cách thay đổi điện áp phần ứng

### **Phương pháp điều khiển tốc độ bằng cách thay đổi điện trở phụ phần ứng**

Nguyên lý điều chỉnh: Nối thêm điện trở phụ  $R_f$  vào mạch phần ứng.

ta có mối quan hệ:  $\omega = f(R_f, \Phi, kt, U)$  với giả thiết rằng : Nếu giữ  $\Phi = \Phi_{dm} = const$ ;  $U = U_{dm} = const$ ;  $R_r = const$  thì tốc độ của động cơ chỉ phụ thuộc vào điện trở phần ứng  $\omega = f(R_f)$

Để thực hiện thay đổi giá trị  $R_f$  của mạch phần ứng ta thực hiện bằng cách nối tiếp một điện trở phụ ( $R_f$ ) có thể thay đổi giá trị vào mạch phần ứng.

Khi thêm điện trở  $R_f$  vào mạch phần ứng ta có :  $R = R_r + R_f$

Theo phương trình đặc tính cơ :  $\omega = \frac{U}{k.\phi} - \frac{R_u + R_f}{(k.\phi)^2} . M$  (2.1)

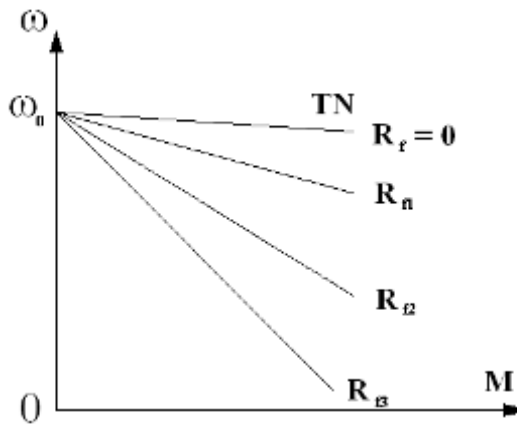
Từ (2.1) ta thấy: khi ta điều chỉnh tăng giá trị của  $R_f$  thì tốc độ của động cơ giảm và ngược lại

Lúc này ta có tốc độ không tải lý tưởng:  $\omega_0 = \frac{U_{dm}}{k.\phi_{dm}} = const$  (2.2)

Và độ cứng của đặc tính cơ:  $\beta = \frac{\Delta M}{\Delta \omega} = -\frac{(k.\phi_{nm})^2}{R_u + R_f} = var$  (2.3)

Khi  $R_f$  càng lớn,  $\beta$  càng nhỏ nghĩa là đặc tính cơ càng dốc. Ứng với  $R_f = 0$  ta có đặc tính cơ tự nhiên.  $\beta$  tự nhiên có giá trị lớn nhất nên có độ cứng hơn tất cả các đường đặc tính có điện trở phụ.

Như vậy khi thay đổi  $R_f$  cho ta một họ đặc tính như sau:



Hình 2.8. Đặc tính cơ của động cơ điện một chiều kích từ độc lập khi thay đổi điện trở phụ mạch phản ứng

Nếu giá trị điện trở  $R_f$  càng lớn thì tốc độ động cơ càng giảm, đồng thời dòng điện ngắn mạch ( $I_{nm}$ ) và mô men ngắn mạch ( $M_{nm}$ ) cũng giảm. Phương pháp này được dùng để hạn chế dòng khởi động của động cơ.

**Phương pháp điều khiển tốc độ động cơ bằng cách thay đổi từ thông kích từ**

Nguyên lý điều chỉnh : Phương pháp điều khiển tốc độ động cơ bằng cách thay đổi từ thông kích từ của động cơ điện một chiều là thực hiện điều chỉnh mô men điện từ của động cơ  $M = k \cdot \Phi \cdot I_u$  và sức điện động quay của động cơ  $E_u = k \cdot \Phi \cdot \omega$

Theo phương trình đặc tính cơ  $\omega = f(U, \varphi_{kt}, R_f)$ , thực hiện giữ  $U = U_{đm} = \text{const}$ ,  $R_u = \text{const}$  ( $R_f = 0$ ) thì lúc này tốc độ của động cơ chỉ phụ thuộc vào từ thông kích từ  $\omega = f(\Phi_{kt})$ .

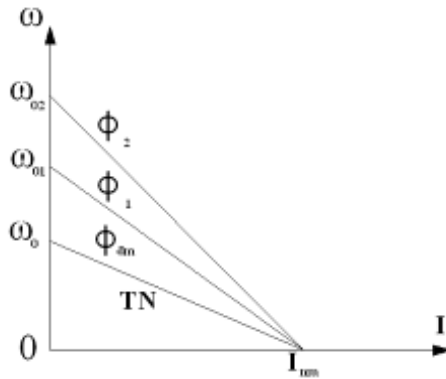
Vậy để điều chỉnh từ thông kích từ  $\Phi_{kt}$  ta thực hiện mắc thêm biến trở  $R_v$  vào mạch kích từ của động cơ, khi điều chỉnh từ thông kích từ  $\Phi_{kt}$  ta phải tuân theo điều kiện sau: không được tăng dòng kích từ  $I_{kt}$  lớn hơn dòng định mức của cuộn kích từ.

Khi thay đổi tốc độ bằng cách thay đổi từ thông kích từ ta có:

$$\text{Tốc độ không tải lý tưởng của động cơ: } \omega_0 = \frac{U_{dm}}{k \cdot \phi_x} = \text{var} \quad (2.4)$$

$$\text{Độ cứng của đặc tính cơ: } \beta = \frac{(k \cdot \phi_x)^2}{R_u} = \text{var} \quad (2.5)$$

Do cấu tạo của động cơ điện, khi từ thông kích từ:  $\Phi_{kt} = \Phi_{dm}$  đã bão hòa, nếu muốn thực hiện tăng dòng kích từ  $I_{kt}$  thì  $\Phi_{kt}$  tăng cũng không tăng đáng kể (do từ thông đã bão hòa) nên thực tế thường ta điều chỉnh bằng cách giảm từ thông kích từ  $\Phi_{kt}$ . Khi từ thông kích từ giảm thì tốc độ của động cơ ( $\omega_x$ ) tăng, còn độ cứng đặc tính cơ ( $\beta$ ) sẽ giảm. Ta có đồ thị đặc tính cơ (hình 2.9) với  $\omega_0$  tăng dần và độ cứng của đặc tính giảm dần khi giảm từ thông.



Hình 2.9. Đặc tính cơ điện của động cơ điện một chiều kích từ độc lập khi giảm từ thông.

Từ hình 2.4 ta thấy khi từ thông kích từ thay đổi với  $\Phi_{dm} > \Phi_1 > \Phi_2$  ta có:

$$\text{Dòng điện ngắn mạch: } I_{nm} = \frac{U_{dm}}{R_u} = \text{const}$$

$$\text{Mô men ngắn mạch: } M_{nm} = K \cdot \Phi_x \cdot I_{nm} = \text{var} \quad (M_{nm} > M_{nm1} > M_{nm2})$$

Từ đồ thị đặc tính 2.4 ta thấy  $\omega_0 < \omega_{01} < \omega_{02}$ .

**Phương pháp điều khiển tốc độ động cơ bằng cách thay đổi điện áp phản ứng**

$$\text{Từ phương trình đặc tính cơ: } \omega = \frac{U}{k \cdot \phi} - \frac{R_u + R_f}{(k \cdot \phi)^2} \cdot M$$

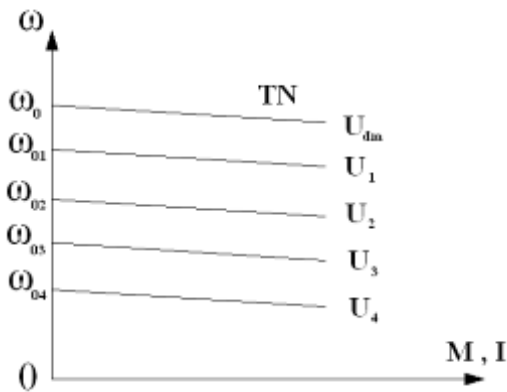
Thực hiện điều chỉnh để giữ:  $\Phi = \Phi_{dm} = const, R_u = const, (R_f = 0), M = const$ . Lúc này tốc độ của động cơ chỉ phụ thuộc vào điện áp phần ứng:  $\omega = f(U_u)$

Khi thực hiện thay đổi điện áp phần ứng của động cơ thì tốc độ không tải lý tưởng sẽ thay đổi, độ cứng đặc tính cơ là không đổi.

$$\text{Tốc độ không tải lý tưởng của động cơ: } \omega_0 = \frac{U_u}{k.\phi} = \text{var} \quad (2.6)$$

$$\text{Độ cứng của đặc tính cơ của động cơ: } \beta = -\frac{(k.\phi)^2}{R_u} = \text{const} \quad (2.7)$$

Như vậy khi thay đổi điện áp phần ứng của động cơ ta thu được một họ các đặc tính cơ như hình 2.10

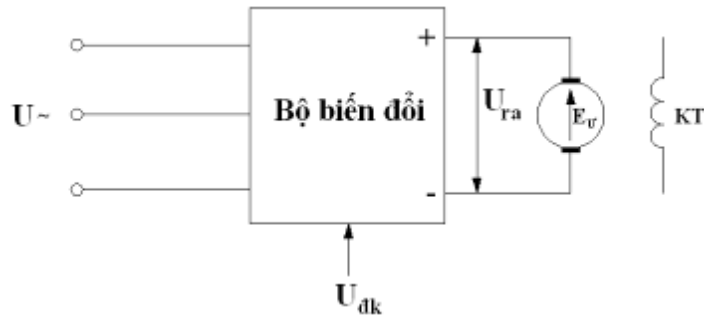


Hình 2.10. Đồ thị đặc tính cơ của động cơ điện một chiều kích từ độc lập khi điện áp phần ứng thay đổi

Từ hình 2.5 ta thấy khi thay đổi điện áp phần ứng của động cơ ta được một họ đặc tính cơ nhân tạo song song với đặc tính tự nhiên.

Cũng từ hình 2.10 ta thấy rằng khi thay đổi điện áp phần ứng của động cơ thì mômen ngắn mạch ( $M_{nm}$ ), dòng điện ngắn mạch ( $I_{nm}$ ) của động cơ giảm và tốc độ động cơ cũng giảm. Do đó phương pháp điều chỉnh điện áp phần ứng được dùng để thực hiện điều chỉnh tốc độ động cơ và hạn chế dòng khởi động.

Để thực hiện điều chỉnh điện áp phần ứng, ta phải sử dụng một bộ biến đổi để thực hiện điều chỉnh điện áp đầu ra cấp cho mạch phần ứng của động cơ



Hình 2.11. Sơ đồ nguyên lý điều chỉnh động cơ

Bộ biến đổi dùng để biến đổi điện áp xoay chiều của lưới điện thành điện áp một chiều với giá trị điện áp đầu ra có thể điều chỉnh được theo yêu cầu của người điều khiển.

## CHƯƠNG 3: ĐIỀU KHIỂN VỊ TRÍ ĐỘNG CƠ DC QUA GIAO DIỆN

### LABVIEW

#### 3.1. Giới thiệu

Có thể nói bộ Điều khiển PID (viết tắt của: Proportional–Integral–Derivative Controller) là một trong những bộ điều khiển phổ biến và quan trọng nhất trong các thiết bị và hệ thống công nghiệp từ ở đĩa CD tới vận tốc xe ô tô đều được thực hiện bởi các thuật toán PID. Tài liệu này đó dùng điều khiển các hệ thống vật lý như động cơ DC, hệ thống lái tự động trên robot, ô tô, lò nhiệt, vv.

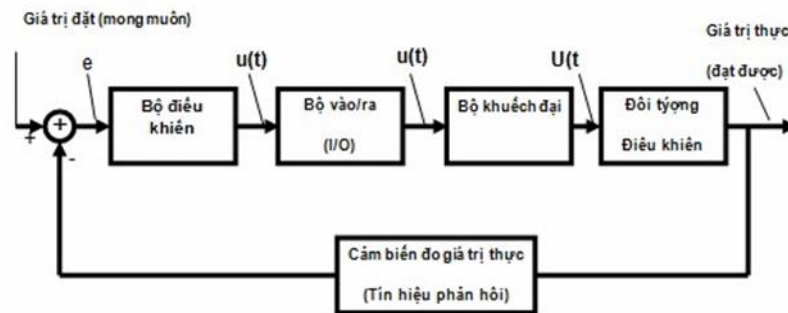
Điều khiển pid là gì ? bộ điều khiển PID (Proportional–Integral– Derivative Controller) là một bộ hiệu chỉnh có phản hồi nhằm làm giá trị sai lệch của một tín hiệu khiển bằng không. Bộ PID có ba thành phần: proportional - tỷ lệ, integral - tích phân, và derivative - đạo hàm), ba thành phần này đều có vai trò đưa sai lệch về không. Tính chất tác động của mỗi thành phần có đặc điểm riêng được khảo sát chi tiết trong phần sau. Tín hiệu phản hồi (feedback signal) thường là tín hiệu thực



được đo bằng cảm biến. Giá trị sai lệch là hiệu của tín hiệu đặt (setpoint) trừ cho tín hiệu phản hồi.

- PID là bộ điều khiển thông dụng nhất trong công nghiệp vì tính dễ áp dụng, và mang lại chất lượng điều khiển ổn định cho hệ thống. Cụ thể, bộ điều khiển PID thường sử dụng trong điều khiển động cơ DC, robot, các hệ thống trong ô tô, điều khiển áp xuất, băng truyền, vv.

### 3.1.1: Bản chất toán học của thuật toán PID



Hình 3.1: bộ PID điều khiển vị trí

Một bộ điều khiển PID có sơ đồ như hình trên. Trong bộ điều khiển PID, sai lệch  $e$  được tính bằng hiệu giá trị đặt hoặc điểm đặt (Set point  $\theta_s$ ) trừ cho giá trị thực tế đo được (measured value của hệ thống  $\theta_m$ ).

$$e = \theta_s - \theta_m$$

Hoạt động của hệ thống điều khiển vị trí. Bộ PID này sẽ đọc và hiểu giá trị mà người điều khiển mong muốn (gọi là giá trị đặt, ở đây là vị trí của B có tọa độ  $x_B=20\text{cm}$ ), thường người điều khiển đưa giá trị đặt vào bộ điều khiển PID thông qua GUI (Graphical user interface - giao diện người dùng đồ họa). Bộ điều khiển PID sẽ tính sai lệch  $e$ , và qua bộ PID thành tín hiệu điều khiển  $u(t)$ 1 tính theo công thức 10.2, sai lệch sau khi tính toán được truyền ra ngoài hệ thống thực thông qua card vào/ra (I/O) như card USB-9001 hoặc NI 6009 ở đây tín hiệu lúc này là tín hiệu điện áp và được gọi là  $u(t)$ 2. Sau đó, tín hiệu này được khuếch đại nhờ một bộ

Driver (ví dụ Motor driver) để tăng tín hiệu đủ công suất điều khiển cơ cấu chấp hành (động cơ DC), gọi là tín hiệu  $U(t)$ . Tín hiệu điều khiển động cơ sẽ điều khiển cơ cấu 5, khi động cơ quay thì thanh kim loại trượt theo phương X và đầu di chuyển dần từ A tới B. Hoàn thành một vòng điều khiển. Sau đó bộ điều khiển PID sẽ liên tục thực hiện lại việc tính toán sai lệch của vị trí đặt (vị trí B) so với giá trị vị trí thực tế (measured signal) của đầu 1 (nhờ vào bộ đo vị trí gắn với động cơ), Nếu giá trị sai lệch vẫn còn thì bộ điều khiển PID tiếp tục phát ra tín hiệu để quay động cơ cho tới khi giá trị thực tế của động cơ trùng khớp với giá trị đặt. Tức khi đó sai lệch sẽ bằng 0. Chừng nào còn sai lệch thì bộ điều khiển PID còn hoạt động để hiệu chỉnh tín hiệu điều khiển. Bản chất toán học của bộ PID sẽ được giải thích trong công thức

$$u(t) = K_p e(t) + K_i \int e(\tau) d(\tau) + K_d \frac{d}{dt} e(t)$$

–  $u(t)$ : tín hiệu điều khiển, là tín hiệu do bộ PID sinh ra, (thường tín hiệu này đi qua một module công suất) và đi vào hệ thống (động cơ một chiều). Có đơn vị phụ thuộc phân cứng, như đối với điều khiển động cơ một chiều 24V thông qua module công suất là một Motor driver 24V thì  $u(t)$  có đơn vị là Volt.

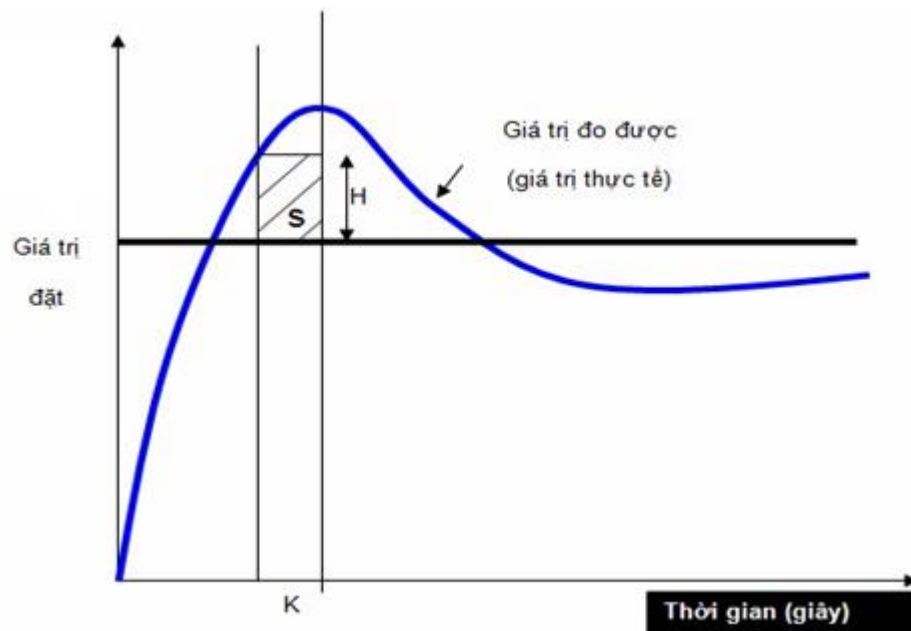
–  $K_p$  : Khâu tỷ lệ.

–  $K_p, K_i, K_d$  : các hệ số tỷ lệ, tích phân và vi phân của bộ PID ,

–  $e(t)$  sai lệch tại thời điểm hiện tại.

–  $e(\tau)d(\tau)$  : Diện tích S tạo bởi đường cong giá trị thực tế, đường thẳng giá trị đặt, và các cận là thời điểm trước và sau

một vòng điều khiển (cũng chính là thời gian thực hiện hết một vòng lặp toàn bộ chương trình điều khiển K hay nói cách khác K chính là đơn vị thời gian trích mẫu  $d(\tau)$  (thường thời gian trích )  $d(\tau)$  (mẫu bé khoảng ms thậm chí nano giây). Vậy tích phân từ  $0 \rightarrow t$  của sai lệch nhân với thời gian lấy mẫu chính là tổng hợp các sai lệch từ khi hệ thống bắt đầu được điều khiển tới thời điểm hệ thống năng hoạt động hiện tại.



Hình 3.3 Mô tả giá trị đặt và đo được và diện tích sai lệch

- Các hệ số luôn không âm: tức  $K_p, K_i, K_d \geq 0$
- $d[e(t)]/dt$ : Tốc độ thay đổi của sai lệch tại thời điểm hiện tại

### **Ý nghĩa các hệ số gain trong bộ PID khi tăng các hệ số $K_p, K_i, K_d$ độc lập nhau**

– Với cùng một giá trị  $e(t)$ , nếu tăng  $K_p$  thì tín hiệu điều khiển  $u(t)$  tăng.  $u(t)$  tăng đồng nghĩa điện áp tác động vào động cơ một chiều tăng lên, điện áp tăng sẽ làm giảm  $e(t)$  nhanh hơn có nghĩa hiệu giá trị đặt và giá trị đo được sẽ giảm nhanh hơn. Khi đó ta gọi khả năng đáp ứng của thống nhanh. đáp ứng của hệ thống được xem là càng nhanh khi thời gian cần thiết để tín hiệu đầu ra của hệ thống đạt tới giá trị đặt càng nhỏ. Tuy nhiên khi  $K_p$  quá lớn, thì hệ thống không ổn định vì có hiện tượng giá đo được của hệ thống vượt quá giá trị đặt, gọi nó là vọt lố

–  $K_i$  có tác dụng làm tăng tốc độ đáp ứng vì nó cũng làm tăng điện áp ( $U(t)$ ) đặt vào động cơ. đồng thời khâu này làm sai lệch tĩnh (steady-state error) trở về 0 nhờ vào đặc tính cộng dồn sai lệch của phép toán tích phân. Sai lệch tĩnh là sai lệch sau khi tín hiệu đầu ra của hệ thống đã ổn định.

–  $K_d$  khâu D có tác dụng làm ổn định hệ thống. Khi  $e(t)$  đổi dấu tức giá trị phản hồi lớn hơn giá trị setpoint, thì khâu D nó có tác dụng làm giảm tín hiệu điều khiển

Phương pháp định bộ thông số  $K_p K_i K_d$  (Các hệ số gain) thủ công thường dựa vào việc thử nghiệm các thông số. Ta xác lập các hệ số  $K_p, K_i, K_d$  bằng không. Sau đó tăng dần  $K_p$  cho tới khi vọt lố đạt bằng gần 1,5 lần giá trị đặt. Hiệu chỉnh  $K_i$  cho sai lệch tĩnh bằng không. Và hiệu chỉnh  $K_d$  cho hệ thống giảm rung lắc.

**Bảng 3.1 : phương pháp chỉnh pid**

Dạng điều khiển	$k_p$	$k_i$	$k_d$
<b>P</b>	$0.50K_u$		
<b>PI</b>	$0.45K$	$1.2K_p/P_u$	
<b>PID</b>	$0.60K_u$	$2K_p/P_u$	$K_p P_u/8$

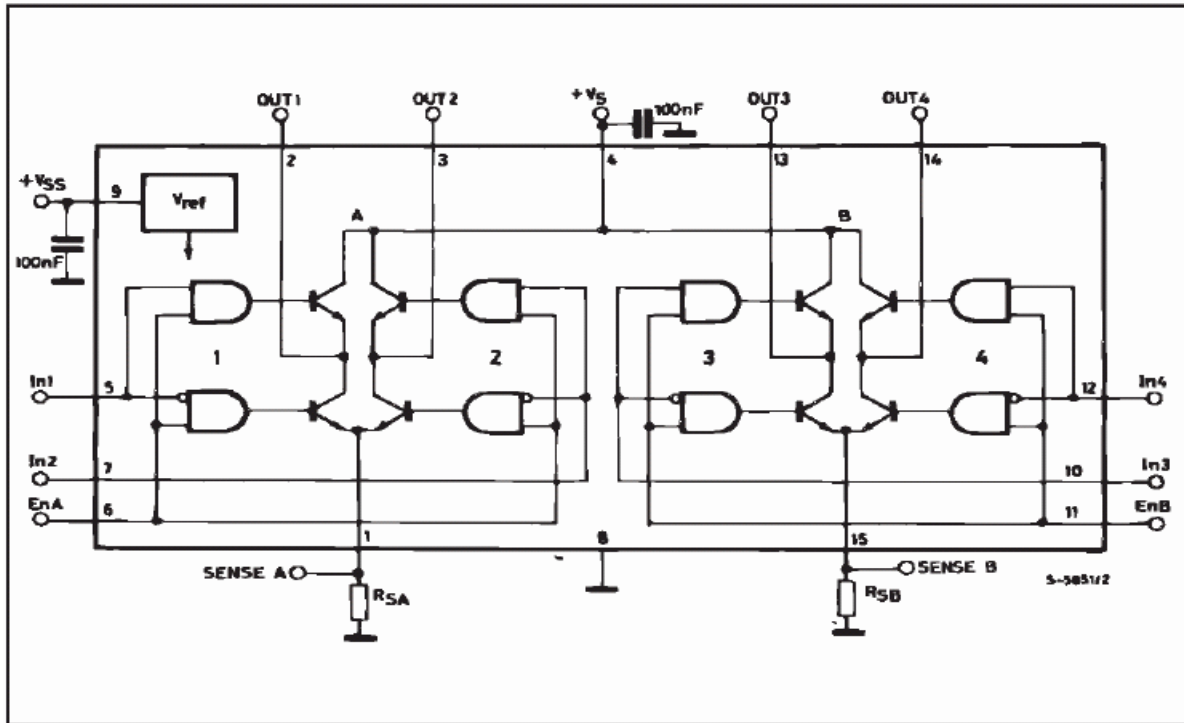
## 3.2. Giới thiệu về drive L298

### 3.2.1. Giới thiệu

IC L298 là mạch tích hợp đơn chip có kiểu vỏ công suất 15 chân (multiwatt 15) và PowerSO20 (linh kiện dán công suất). Là IC mạch cầu đôi (dual full-bridge) có khả năng hoạt động ở điện thế cao, dòng cao. Nó được thiết kế tương thích chuẩn TTL và lái tải cảm kháng như relay, cuộn solenoid, động cơ DC và động cơ bước. Nó có 2 chân enable (cho phép) để cho phép/không cho phép IC hoạt động, độc lập với các chân tín hiệu vào. Cực phát (emitter) của transistor dưới của mỗi mạch cầu được nối với nhau và nối ra chân ngoài để nối với điện trở cảm ứng dòng khi cần. Nó có thêm một chân cấp nguồn giúp mạch logic có thể hoạt động ở điện thế thấp hơn.

### 3.2.2. Sơ đồ khối

### BLOCK DIAGRAM

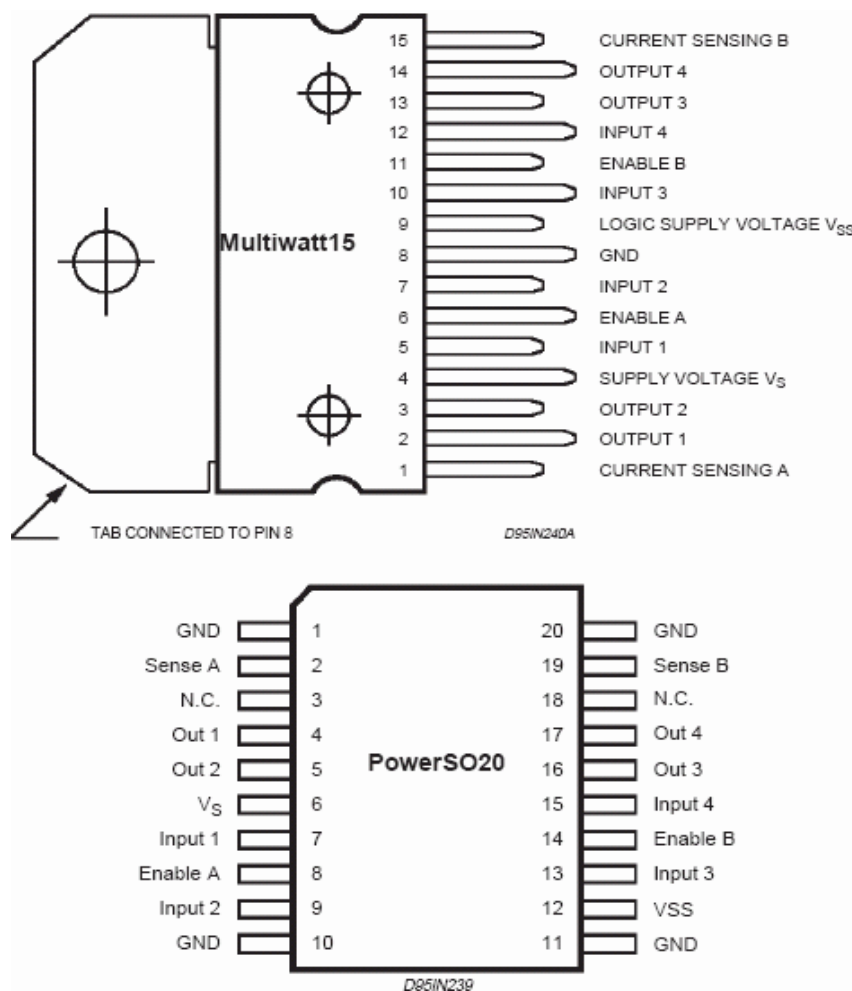


### 3.2.3. Các giá trị chịu đựng tối đa

#### ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
$V_S$	Power Supply	50	V
$V_{SS}$	Logic Supply Voltage	7	V
$V_i, V_{en}$	Input and Enable Voltage	-0.3 to 7	V
$I_o$	Peak Output Current (each Channel)		
	- Non Repetitive ( $t = 100\mu s$ )	3	A
	- Repetitive (80% on -20% off, $t_{on} = 10ms$ )	2.5	A
	-DC Operation	2	A
$V_{sens}$	Sensing Voltage	-1 to 2.3	V
$P_{tot}$	Total Power Dissipation ( $T_{case} = 75^\circ C$ )	25	W
$T_{op}$	Junction Operating Temperature	-25 to 130	$^\circ C$
$T_{stg}, T_j$	Storage and Junction Temperature	-40 to 150	$^\circ C$

### 3.2.4. Sơ đồ chân



### 3.2.5. Thông số về nhiệt độ:

Kí hiệu	Thông số	PowerSO 20	Multiwatt 15	Đơn vị
$R_{th-j-case}$	Độ bền nhiệt của mối nối PN – vỏ(MAX)	-	3	$^{\circ}C/W$
$R_{th-j-amp}$	Độ bền nhiệt của mối nối PN – môi trường(MAX)	13(*)	35	$^{\circ}C/W$

(\*) không tính cho kiểu vỏ nhôm

### 3.2.6. Chức năng các chân:

MW.1 5	PowerS O	Tên	Chức năng
1;15	2;19	Sense A; Sense B	Nối chân này qua điện trở cảm ứng dòng xuống GND để điều khiển dòng tải
2;3	4 ; 5	Out 1; Out 2	Ngõ ra của cầu A. Dòng của tải mắc giữa hai chân này được qui định bởi chân 1.
4	6	VS	Chân cấp nguồn cho tầng công suất. Cần có một tụ điện không cảm kháng 100nF nối giữa chân này và chân GND
5;7	7 ; 9	Input 1; Input 2	Chân ngõ vào của cầu A, tương thích chuẩn TTL
6;11	8;14	Enable A; EnableB	Chân ngõ vào enable (cho phép) tương thích chuẩn TTL. Mức thấp ở chân này sẽ cấm (disable) ngõ ra cầu A (đối với chân EnableA) và/hoặc cầu B (đối với chân EnableB)
8	1,10,11, 20	GND	Chân đất (Ground)
9	1 2	VSS	Chân cấp nguồn cho khối logic. Cần có tụ điện 100nF nối giữa chân này với GND
10; 12	13;15	Input 3; Input 4	Các chân logic ngõ vào của cầu B
13; 14	16;17	Out 3; Out 4	Ngõ ra của cầu B. Dòng của tải mắc giữa hai chân này được qui định bởi chân 15.
-	3;18	N.C.	Không kết nối (bỏ trống)

### 3.2.7. Thông tin ứng dụng

Mạch cầu H kép L298 của [SGS-Thompson](#) (và các hãng khác) có thể điều khiển dòng điện lên tới 2A trên một kênh và được đóng

vỏ như một linh kiện công suất, ta có thể nối hai cầu H bên trong một vỏ LS298 thành một cầu H 4A.

IC L298 tích hợp 2 tầng công suất (A, B). Tần công suất chính là mạch cầu và ngõ ra của nó có thể lái các loại tải cảm thông dụng ở nhiều chế độ hoạt động khác nhau (tùy thuộc vào sự điều khiển ở ngõ vào) Dòng điện từ chân ngõ ra chảy qua tải đến chân cảm ứng dòng : điện trở ngoài RSA, RSB cho phép việc cảm ứng cường độ dòng điện này.

Mỗi cầu được điều khiển bởi 4 cổng ngõ vào In1, In2, EnA, và In3, In4, EnB. Các chân In có tác dụng khi chân En ở mức cao, khi chân En ở mức thấp, các chân ngõ vào In ở trạng thái cấm. Tất cả các chân đều tương thích với chuẩn TTL.

### 3.2.8. Đặc tính điện:

**ELECTRICAL CHARACTERISTICS** ( $V_s = 42V$ ;  $V_{SS} = 5V$ ,  $T_j = 25^\circ C$ ; unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
$V_s$	Supply Voltage (pin 4)	Operative Condition	$V_{IH} + 2.5$		46	V
$V_{SS}$	Logic Supply Voltage (pin 9)		4.5	5	7	V
$I_s$	Quiescent Supply Current (pin 4)	$V_{en} = H$ ; $I_L = 0$ $V_i = L$ $V_i = H$		13 50	22 70	mA mA
		$V_{en} = L$ $V_i = X$			4	mA
$I_{SS}$	Quiescent Current from $V_{SS}$ (pin 9)	$V_{en} = H$ ; $I_L = 0$ $V_i = L$ $V_i = H$		24 7	36 12	mA mA
		$V_{en} = L$ $V_i = X$			6	mA
$V_{iL}$	Input Low Voltage (pins 5, 7, 10, 12)		-0.3		1.5	V
$V_{iH}$	Input High Voltage (pins 5, 7, 10, 12)		2.3		$V_{SS}$	V
$I_{iL}$	Low Voltage Input Current (pins 5, 7, 10, 12)	$V_i = L$			-10	$\mu A$
$I_{iH}$	High Voltage Input Current (pins 5, 7, 10, 12)	$V_i = H \leq V_{SS} - 0.6V$		30	100	$\mu A$
$V_{en} = L$	Enable Low Voltage (pins 6, 11)		-0.3		1.5	V
$V_{en} = H$	Enable High Voltage (pins 6, 11)		2.3		$V_{SS}$	V
$I_{en} = L$	Low Voltage Enable Current (pins 6, 11)	$V_{en} = L$			-10	$\mu A$
$I_{en} = H$	High Voltage Enable Current (pins 6, 11)	$V_{en} = H \leq V_{SS} - 0.6V$		30	100	$\mu A$
$V_{CEsat(H)}$	Source Saturation Voltage	$I_L = 1A$ $I_L = 2A$	0.95	1.35 2	1.7 2.7	V V
$V_{CEsat(L)}$	Sink Saturation Voltage	$I_L = 1A$ (5) $I_L = 2A$ (5)	0.85	1.2 1.7	1.6 2.3	V V
$V_{CEsat}$	Total Drop	$I_L = 1A$ (5) $I_L = 2A$ (5)	1.80		3.2 4.9	V V
$V_{sens}$	Sensing Voltage (pins 1, 15)		-1 (1)		2	V



## ELECTRICAL CHARACTERISTICS (continued)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
T <sub>1</sub> (V <sub>i</sub> )	Source Current Turn-off Delay	0.5 V <sub>i</sub> to 0.9 I <sub>L</sub> (2); (4)		1.5		μs
T <sub>2</sub> (V <sub>i</sub> )	Source Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (2); (4)		0.2		μs
T <sub>3</sub> (V <sub>i</sub> )	Source Current Turn-on Delay	0.5 V <sub>i</sub> to 0.1 I <sub>L</sub> (2); (4)		2		μs
T <sub>4</sub> (V <sub>i</sub> )	Source Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (2); (4)		0.7		μs
T <sub>5</sub> (V <sub>i</sub> )	Sink Current Turn-off Delay	0.5 V <sub>i</sub> to 0.9 I <sub>L</sub> (3); (4)		0.7		μs
T <sub>6</sub> (V <sub>i</sub> )	Sink Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (3); (4)		0.25		μs
T <sub>7</sub> (V <sub>i</sub> )	Sink Current Turn-on Delay	0.5 V <sub>i</sub> to 0.9 I <sub>L</sub> (3); (4)		1.6		μs
T <sub>8</sub> (V <sub>i</sub> )	Sink Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (3); (4)		0.2		μs
f <sub>c</sub> (V <sub>i</sub> )	Commutation Frequency	I <sub>L</sub> = 2A		25	40	KHz
T <sub>1</sub> (V <sub>en</sub> )	Source Current Turn-off Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (2); (4)		3		μs
T <sub>2</sub> (V <sub>en</sub> )	Source Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (2); (4)		1		μs
T <sub>3</sub> (V <sub>en</sub> )	Source Current Turn-on Delay	0.5 V <sub>en</sub> to 0.1 I <sub>L</sub> (2); (4)		0.3		μs
T <sub>4</sub> (V <sub>en</sub> )	Source Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (2); (4)		0.4		μs
T <sub>5</sub> (V <sub>en</sub> )	Sink Current Turn-off Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (3); (4)		2.2		μs
T <sub>6</sub> (V <sub>en</sub> )	Sink Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (3); (4)		0.35		μs
T <sub>7</sub> (V <sub>en</sub> )	Sink Current Turn-on Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (3); (4)		0.25		μs
T <sub>8</sub> (V <sub>en</sub> )	Sink Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (3); (4)		0.1		μs

1) Sensing voltage can be -1 V for t ≤ 50 μsec; in steady state V<sub>sens</sub> min ≥ -0.5 V.

2) See fig. 2.

3) See fig. 4.

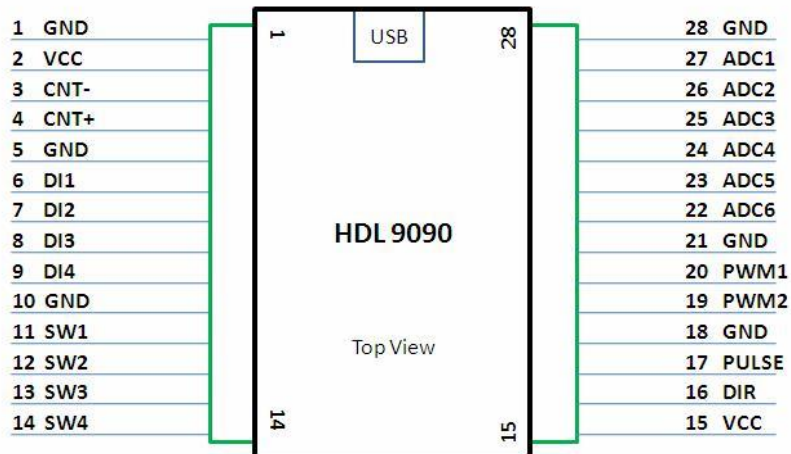
4) The load must be a pure resistor.

## 3.3. Giới thiệu về Card USB HDL 9090

### 3.3.1. GIỚI THIỆU SẢN PHẨM

Card USB HDL 9090 là card thu thập dữ liệu và xuất tín hiệu điều khiển đa năng thế hệ tiếp theo của HDL 9001. Ngoài các chức năng đã có trên HDL 9001 như thu thập dữ liệu từ các cảm biến, điều khiển ON/OFF, điều chế xung PWM v.v. HDL 9090 còn có chức năng Digital Input giúp bạn ứng dụng linh hoạt hơn. Với tính năng vượt trội như tốc độ cao ADC cao hơn 5 lần và chính xác hơn 4 lần so với HDL 9001, card USB 9090 là lựa chọn tối ưu cho ứng dụng thu thập dữ liệu và điều khiển của bạn.

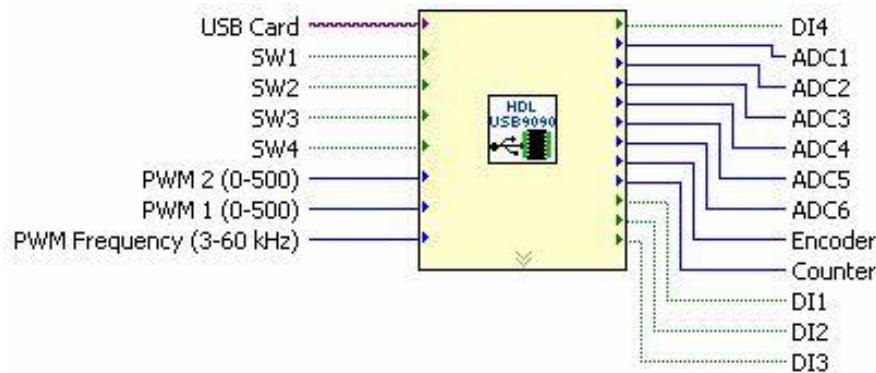
- Mô tả cụ thể các chân tín hiệu của card HOCDELAM USB 9090 như bảng sau:



Ký hiệu	Mô tả	Loại
GND	Ground – chân mát	Nguồn
VCC	Nguồn 5V lấy từ USB	Nguồn
CNT+/CNT	Chân đếm Counter	Input
DI1-DI4	Tín hiệu vào dạng số 0-5V	Input
SW1-SW4	Tín hiệu ra dạng số 0-5V	Output
DIR	Set bộ đếm xung đếm lên(5V) hay đếm xuống (0V)	Input
PULSE	Đếm xung tín hiệu 0-5V	Input
PWM1- PWM2	Xuất tín hiệu PWM để điều khiển	Output
ADC1- ADC6	Nhận tín hiệu vào tương tự (Analog) 0-5V	Input

- Phần mềm đi kèm
  - o NI VISA phiên bản 3.1 hoặc cao hơn: đây là phần mềm của công ty NI, bạn phải cài phần mềm này trước khi muốn giao tiếp LabVIEW (PC) với bất kì thiết bị ngoại vi nào.

- o Thực hiện các bước kết nối thiết bị HDL USB 9090 vào máy tính (Xem mục 2. *Kết nối Card HDL USB 9090 với máy tính*).
- o Hàm HDL USB 9090: để truyền nhận tín hiệu từ LabVIEW với card HDL USB 9090 (Việc sử dụng hàm được hướng dẫn chi tiết tại mục 3. *Lập trình giao tiếp và điều khiển thiết bị với Card HDL USB 9090*), sơ đồ chân (I/O) của hàm như sau:



- Mô tả các chân của hàm HDL USB 9090 để lập trình như bảng sau:

Chân	Loại	Mô tả
USB Card	Control	Tạo control tại chân này để chọn thiết bị USB HDL 9090
SW1-SW4	Control	Nội giá trị Boolean (TRUE-FALSE) vào các chân này để phân cứng HDL 9090 xuất ra tín hiệu số tương ứng (TRUE là 5V, FALSE 0V)
PWM1-PWM2	Control	Nội giá trị số nguyên 0-500 vào để phân cứng xuất xung PWM có duty cycle tương ứng là 0-100%
PWM Frequency (3-60kHz)	Control	Nội giá trị từ 3000-60000 vào để chọn tần số tín hiệu PWM card xuất ra.
DI1-DI4	Indicator	Giá trị tín hiệu số đọc được từ chân DI1-DI4 trên card. Giá trị mặc định là TRUE (5V)

ADC1-ADC6	Indicator	Giá trị ADC (0-1023) đọc được từ các chân ADC trong ứng trên card
Encoder	Indicator	Giá trị encoder 0-100000

### 3.3.2. KẾT NỐI CARD HDL USB 9090 VỚI MÁY TÍNH

Máy tính cần cài NI VISA phiên bản 3.1 hay cao hơn. Sau đó thực hiện các bước sau:

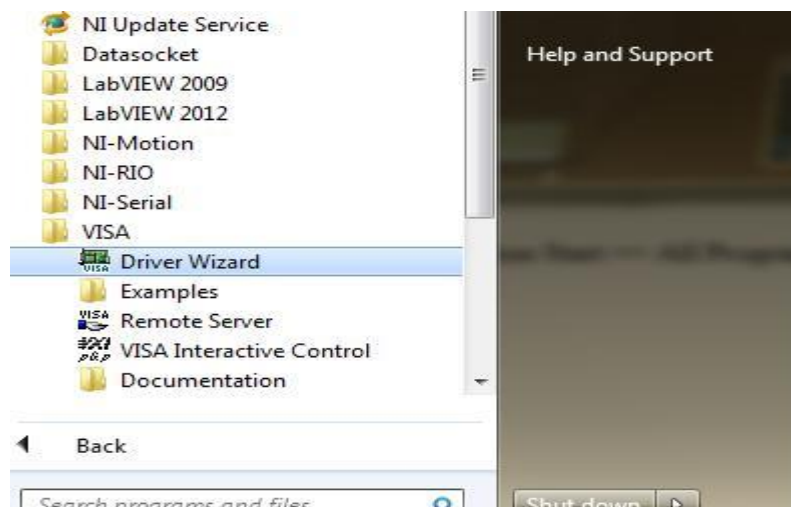
#### Bước 1:

Kết nối HDL USB 9090 vào máy tính bằng cable USB

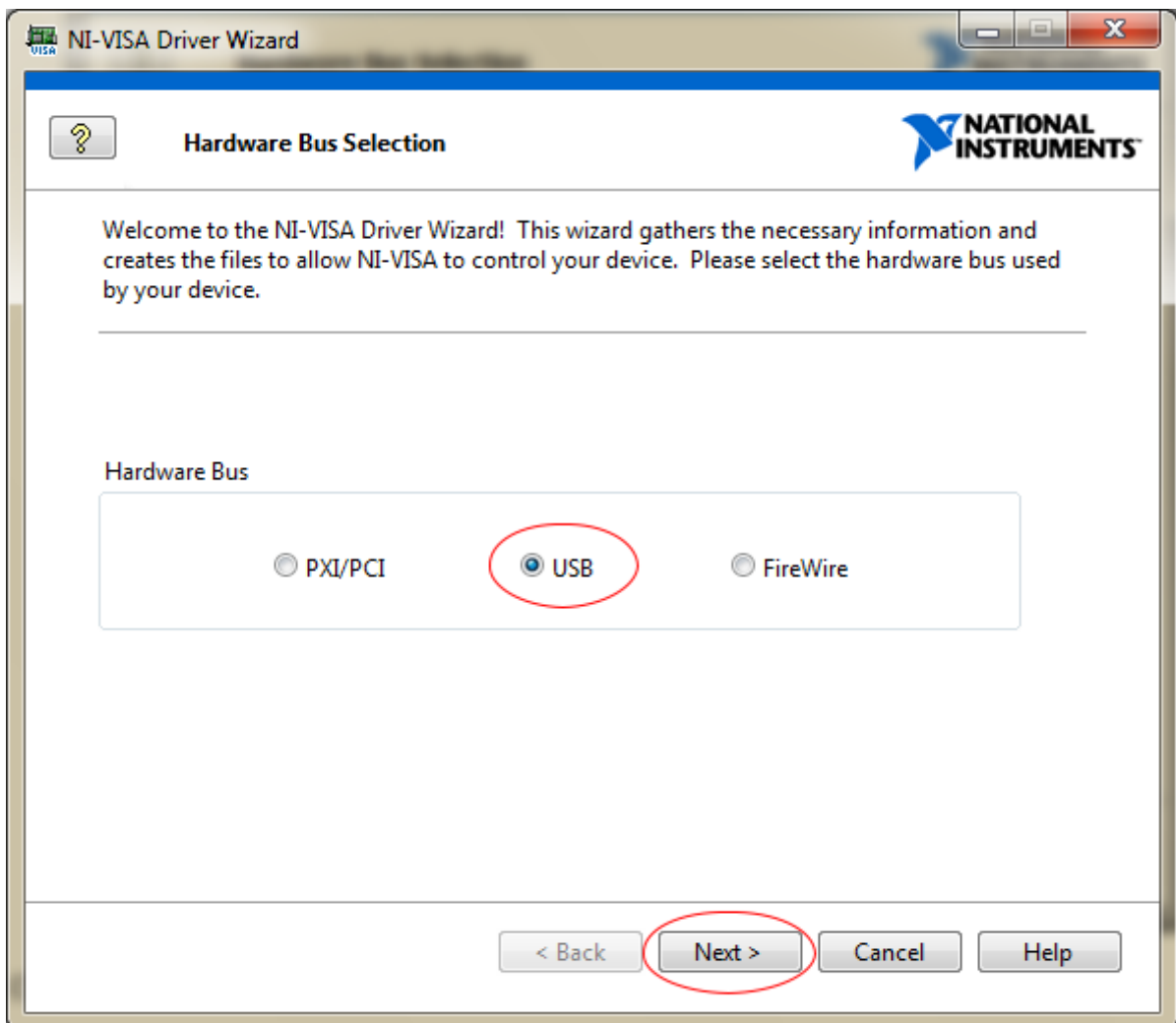


#### Bước 2:

Chọn Start >> All Programs >> National Instruments >> VISA >> Driver Wizard



**Bước 3:**  
Chọn USB



Chọn Next

**Bước 4:**

Nhập 9090 vào khung USB Manufacturer ID (Vendor ID)

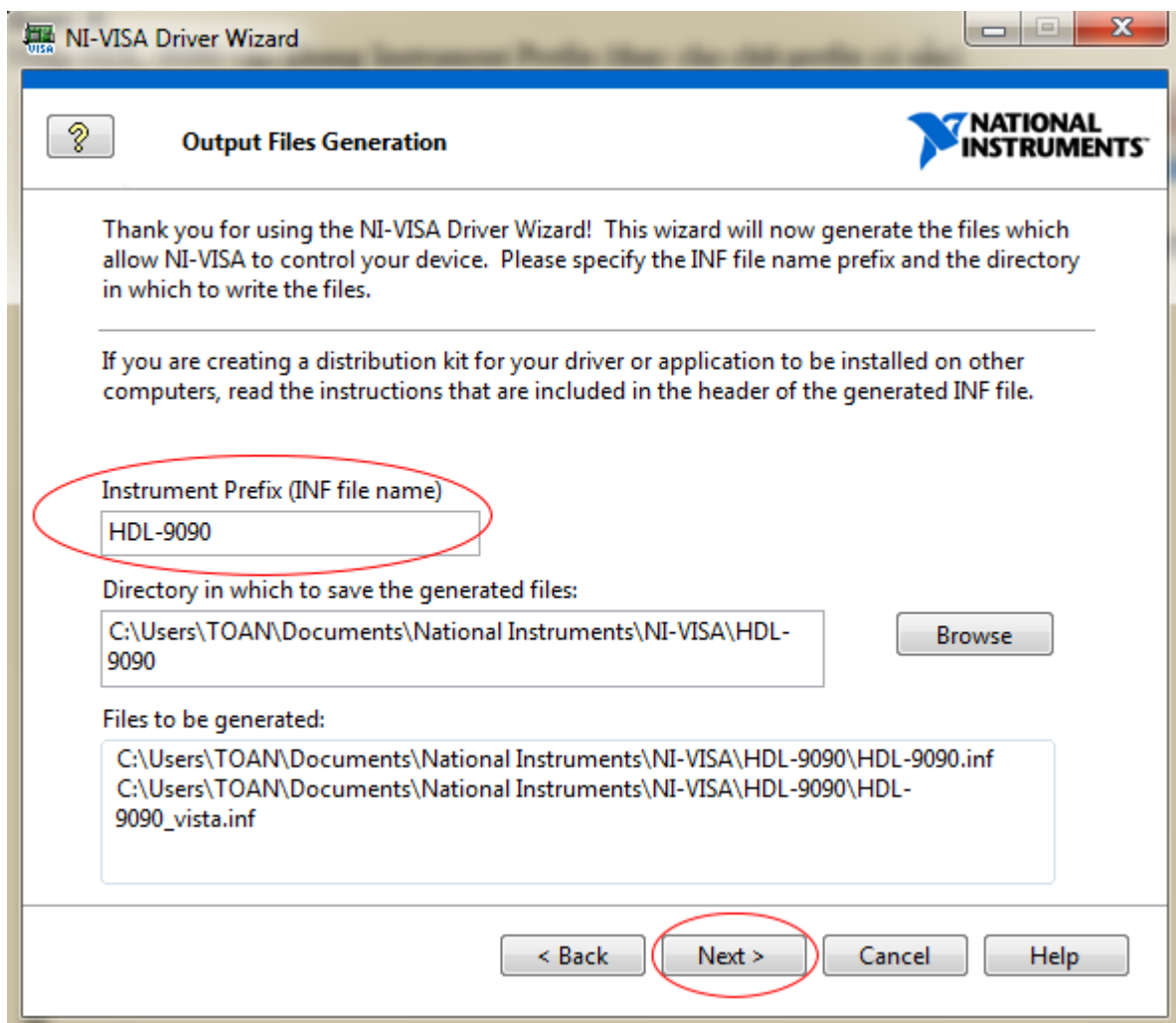
Nhập 0001 vào khung USB Model Code (Product ID)

The screenshot shows the 'NI-VISA Driver Wizard' window with the 'USB - Device Information' tab selected. The window title bar includes the NI-VISA logo and the text 'NI-VISA Driver Wizard'. The main content area contains a welcome message and instructions. Below the text, there are several input fields: 'USB Manufacturer ID (Vendor ID)' with the value '0x 9090', 'USB Model Code (Product ID)' with the value '0x 0001', 'Manufacturer Name' with 'Device Manufacturer Name', 'Model Name' with 'Device Model Name', and 'Number of Interfaces' with the value '2'. A checkbox for 'Compound Device?' is present and unchecked. A 'NOTE' section at the bottom explains that the wizard may not be necessary if the device conforms to the USB Test & Measurement Class (USBTMC) protocol. At the bottom of the window, there are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'. The 'Next >' button is circled in red.

Chọn Next

### Bước 5:

Nhập HDL-9090 vào khung Instrument Prefix (thay cho chữ prefix có sẵn).



The screenshot shows the 'NI-VISA Driver Wizard' window at the 'Output Files Generation' step. The window title is 'NI-VISA Driver Wizard' and it features the National Instruments logo. The main text reads: 'Thank you for using the NI-VISA Driver Wizard! This wizard will now generate the files which allow NI-VISA to control your device. Please specify the INF file name prefix and the directory in which to write the files.' Below this, it says: 'If you are creating a distribution kit for your driver or application to be installed on other computers, read the instructions that are included in the header of the generated INF file.'

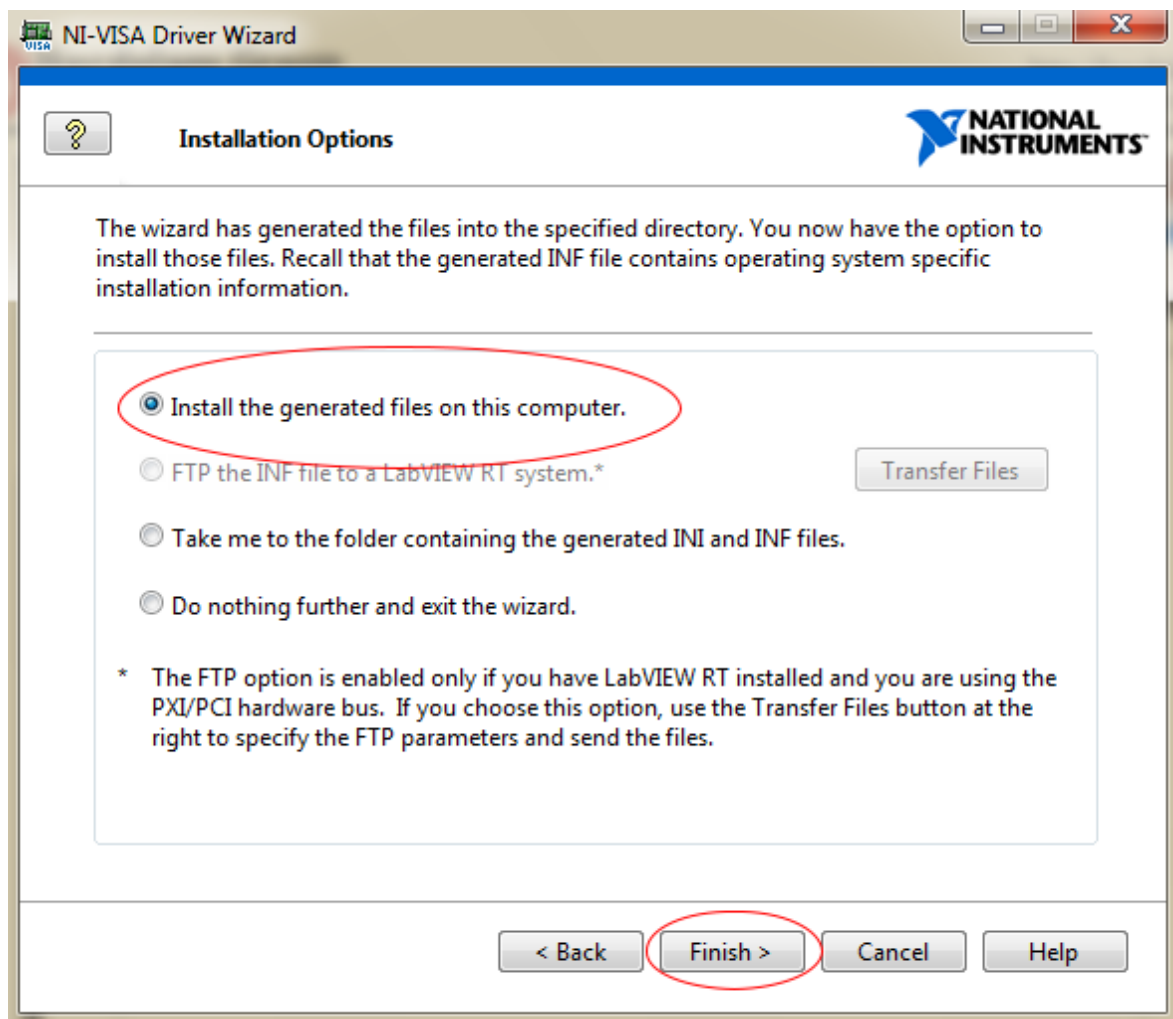
The 'Instrument Prefix (INF file name)' field is highlighted with a red circle and contains the text 'HDL-9090'. Below this field, the 'Directory in which to save the generated files:' is shown as 'C:\Users\TOAN\Documents\National Instruments\NI-VISA\HDL-9090', with a 'Browse' button to its right. The 'Files to be generated:' section lists two files: 'C:\Users\TOAN\Documents\National Instruments\NI-VISA\HDL-9090\HDL-9090.inf' and 'C:\Users\TOAN\Documents\National Instruments\NI-VISA\HDL-9090\HDL-9090\_vista.inf'.

At the bottom of the window, there are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'. The 'Next >' button is highlighted with a red circle.

Chọn Next

## Bước 6:

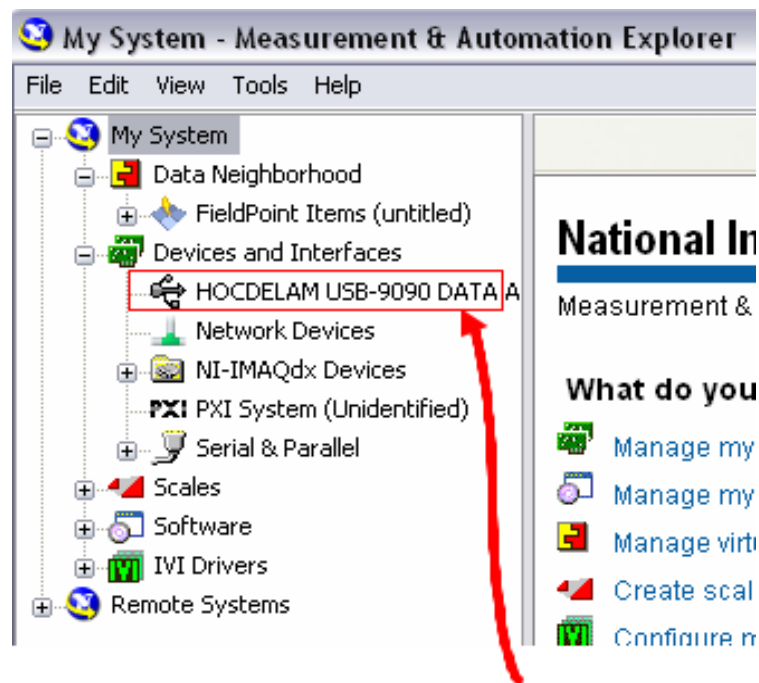
Chọn mục đầu tiên hình dưới (Install the generated files on this computer)





Chọn Finish

Chọn mục Device & Interface:



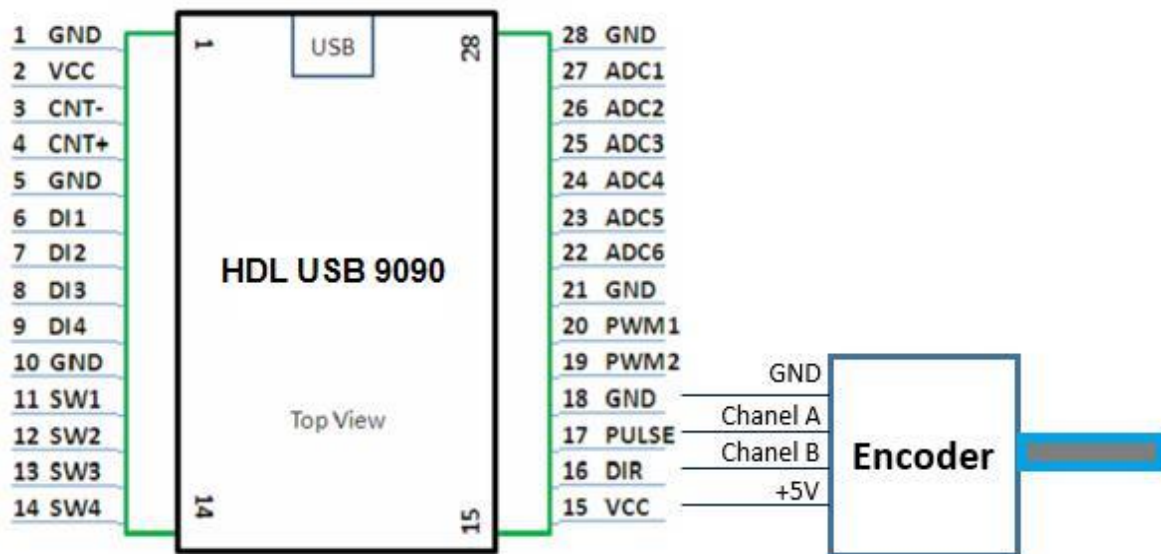
## LẬP TRÌNH GIAO TIẾP VÀ ĐIỀU KHIỂN THIẾT BỊ VỚI HDL USB 9090

### 3.4.1. Đọc tín hiệu từ Encoder

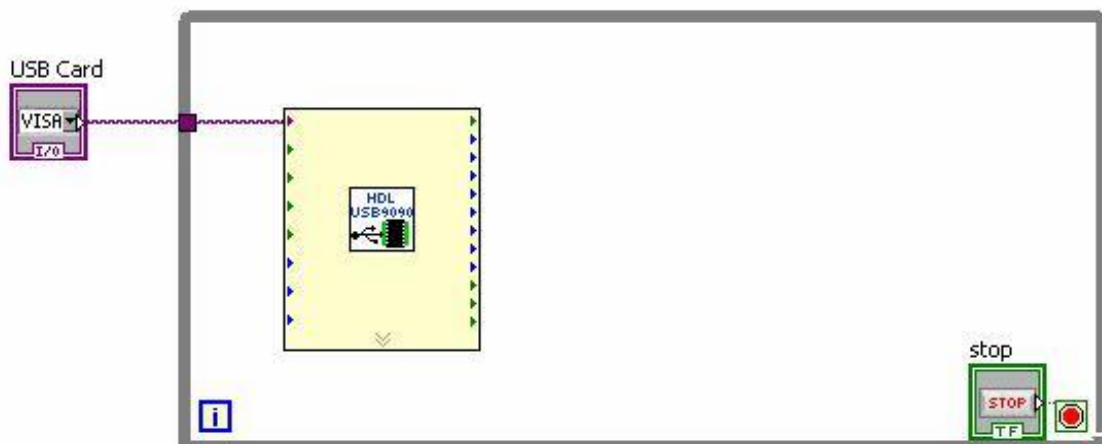
- Trong động cơ thường có gắn encoder để đọc được góc quay, vòng quay của động cơ. Encoder xuất ra tín hiệu xung và card HDL sẽ đếm số xung này. Từ số tính được số vòng và góc quay tùy theo thông số của encoder ( ví dụ 1 vòng = 100

xung)

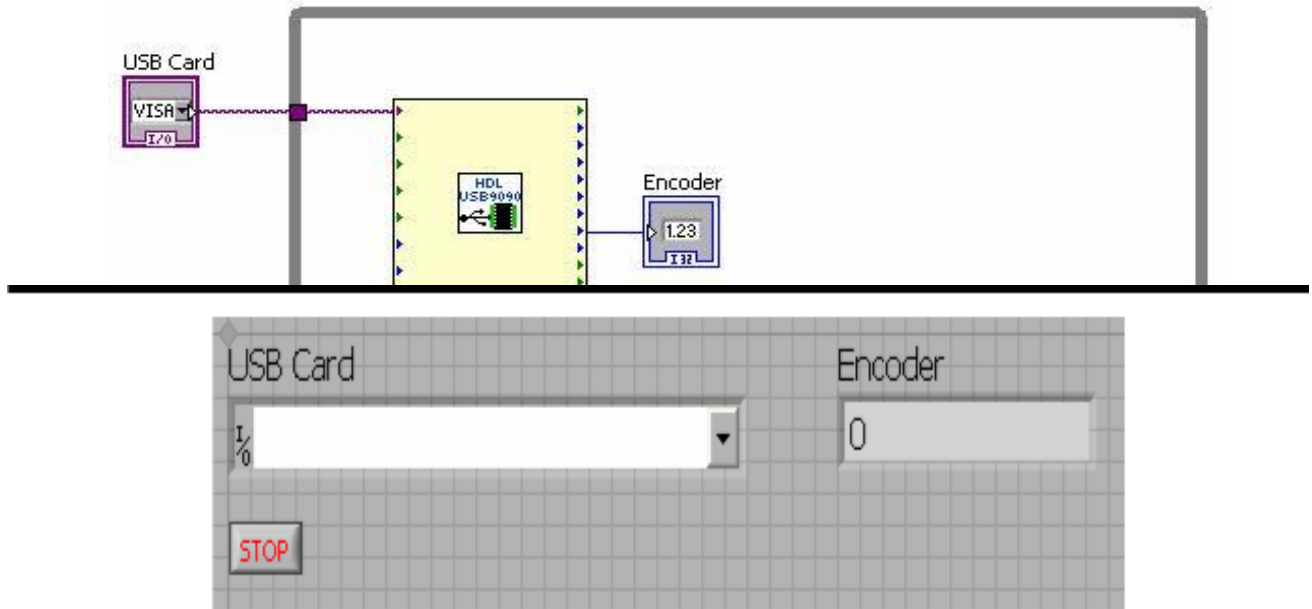
Encoder thông thường có 4 dây nối: VCC, GND, Channel A & Channel B. Nối dây VCC và GND tương ứng vào card HDL 9090. Dây Channel A nối vào chân PULSE (chân 17) và dây channel B nối vào chân DIR (chân 16)



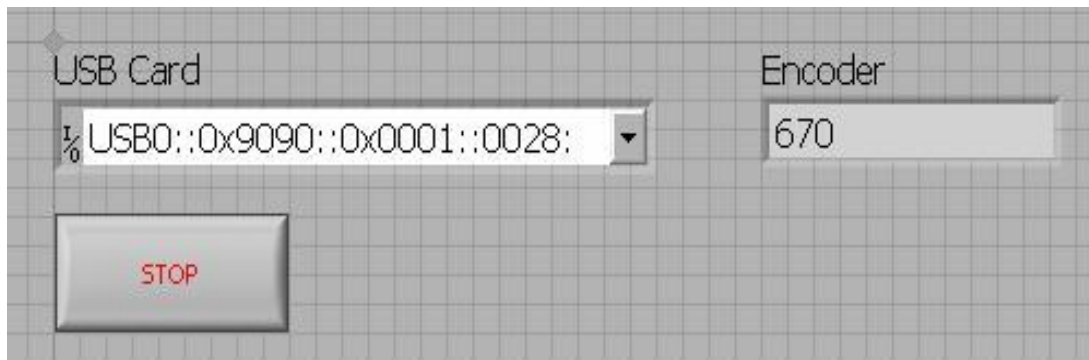
Lúc này bạn sẽ lập trình chương trình đọc Encoder rất đơn giản. Trước tiên là làm theo các bước như bài 1 để được block diagram sau:



- Bây giờ, right click lên chân Encoder của hàm và chọn Create >> Indicator



- Chương trình đã hoàn thành. Chọn thiết bị USB HDL 9090 trong ô USB Card và chạy chương trình. Bạn dùng tay để xoay trục của Encoder thì sẽ thấy giá trị của Encoder thay đổi tương ứng.



- Bạn đã hoàn thành bài 3 là đọc giá trị từ Encoder

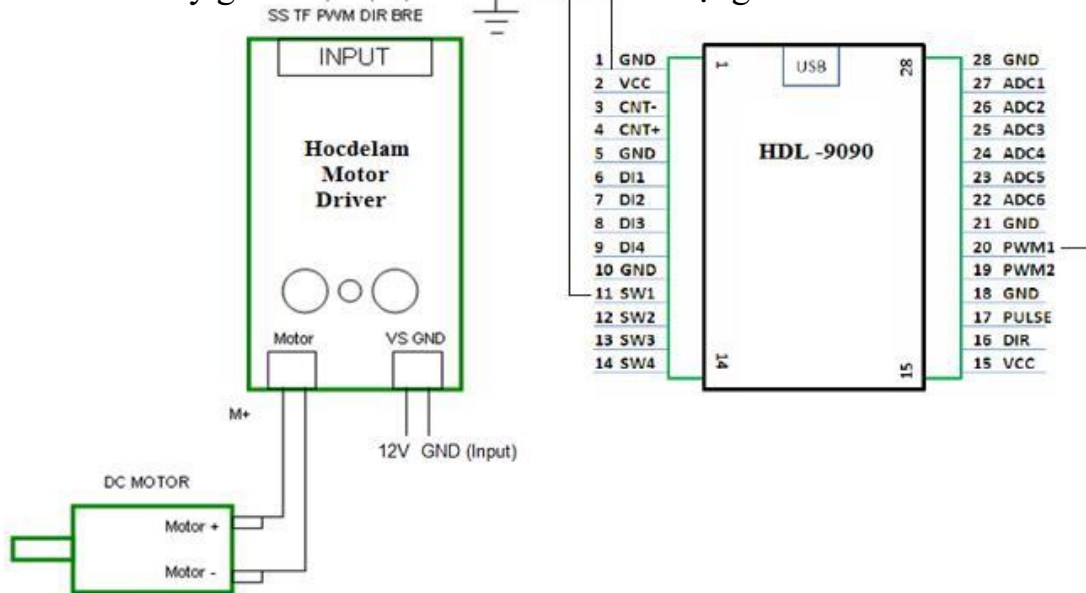
### 3.4.2. Lập trình xuất xung PWM điều khiển động cơ DC

Để có thể điều khiển 1 động cơ DC với card HDL USB 9090 bạn cần chuẩn bị thêm:

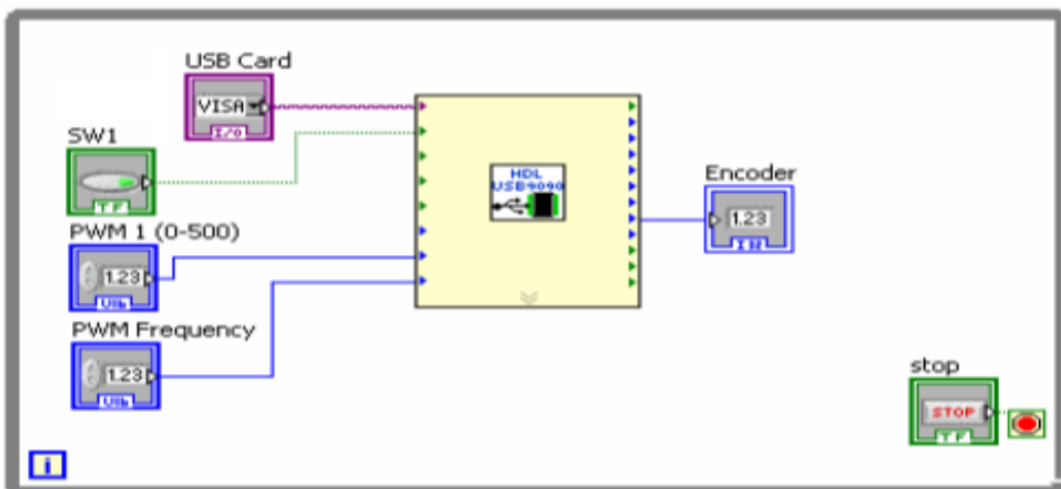
- o 1 động cơ DC
- o 1 mạch công suất (Motor driver)
- o 1 bộ nguồn 1 chiều phù hợp với công suất động cơ.

Ở bài hướng dẫn này sử dụng động cơ DC 12V, mạch công suất HDL Driver Hippo (xem thêm tại <http://hocdelam.org>), bộ nguồn Adapter 12V.

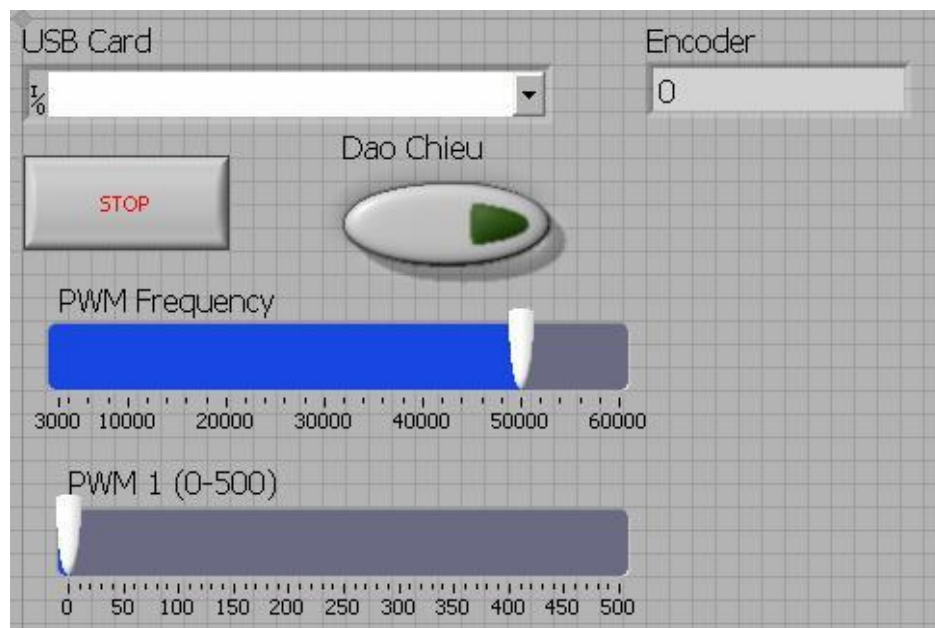
- Cách nối dây giữa card HDL USB 9090 với động cơ DC và motor driver:



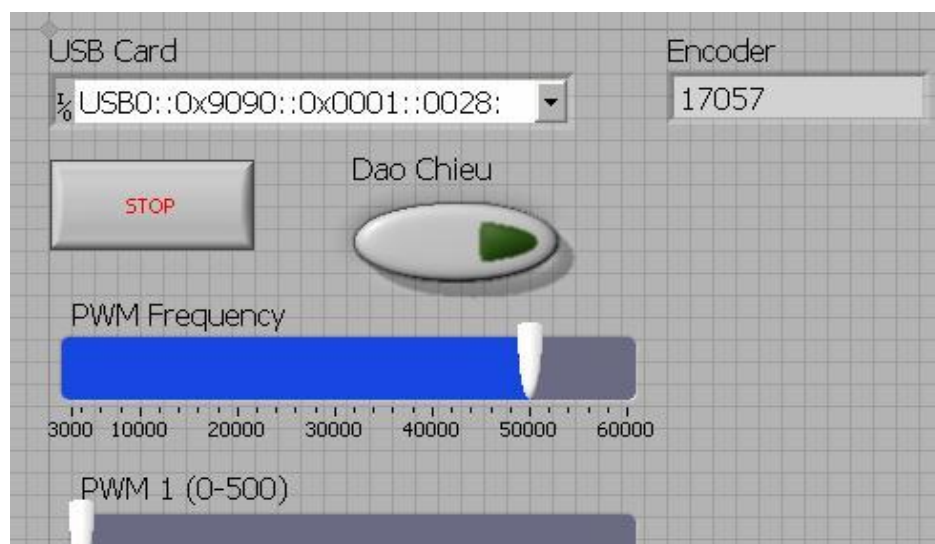
Để phát xung chuột phải vào điều khiển động cơ thì cần thêm vào chương trình đọc encoder đã viết ở trên: thư viện HDL USB9090, chân PWM Frequency, chọn Create >> Control kênh PWM1, chọn Create >> Control kênh SW1, chọn Create >> Control



Chương trình cơ bản đã hoàn thành. Front Panel của chương trình điều khiển này có thể sửa lại cho đẹp hơn bằng cách dùng control thanh trượt:



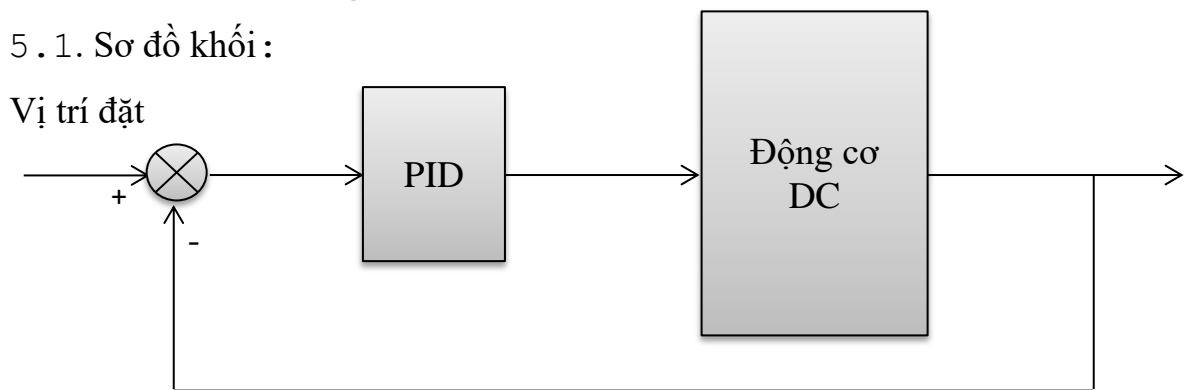
- Chọn thiết bị HDL 9090 trong ô USB Card, và chạy chương trình. Bạn nhập giá trị PWM Frequency từ 3000-60000 và nhập giá trị PWM1 từ 0-500:



- Động cơ sẽ chạy nhanh hay chậm tùy vào giá trị PWM1. Khi muốn đổi chiều động cơ, bạn click trái chuột vào nút SW1 và quan sát đảo chiều.
- Phát xung và điều khiển động cơ.

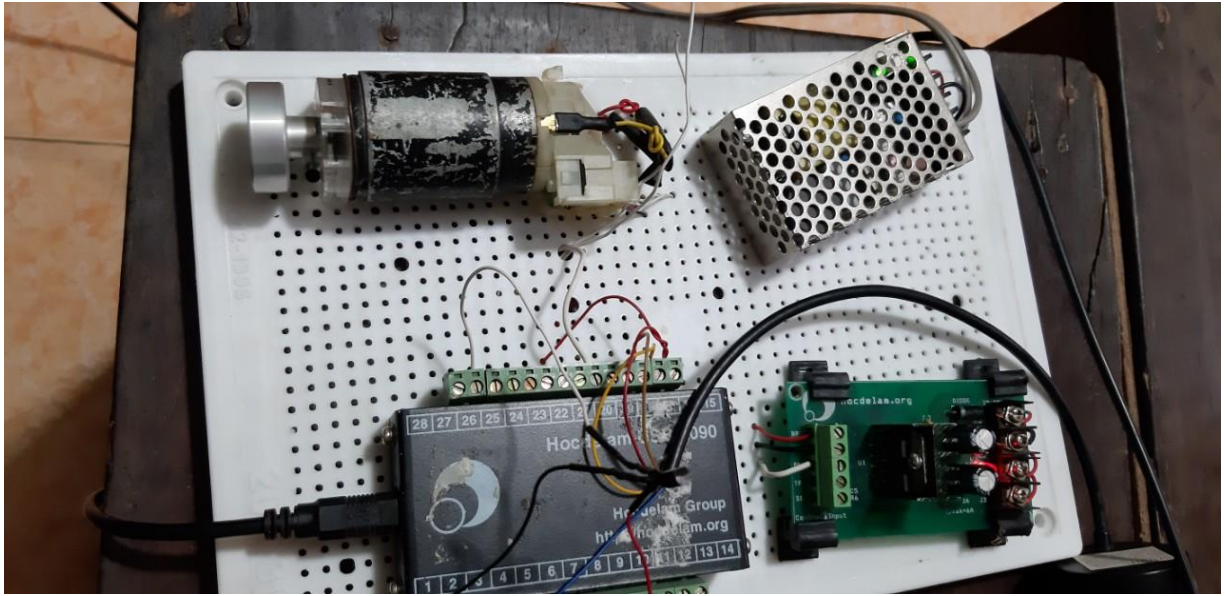
### Điều khiển PID cho động cơ DC

#### 3.5.1. Sơ đồ khối:



Hình 3.5: Sơ đồ khối mạch điện của điều khiển PID

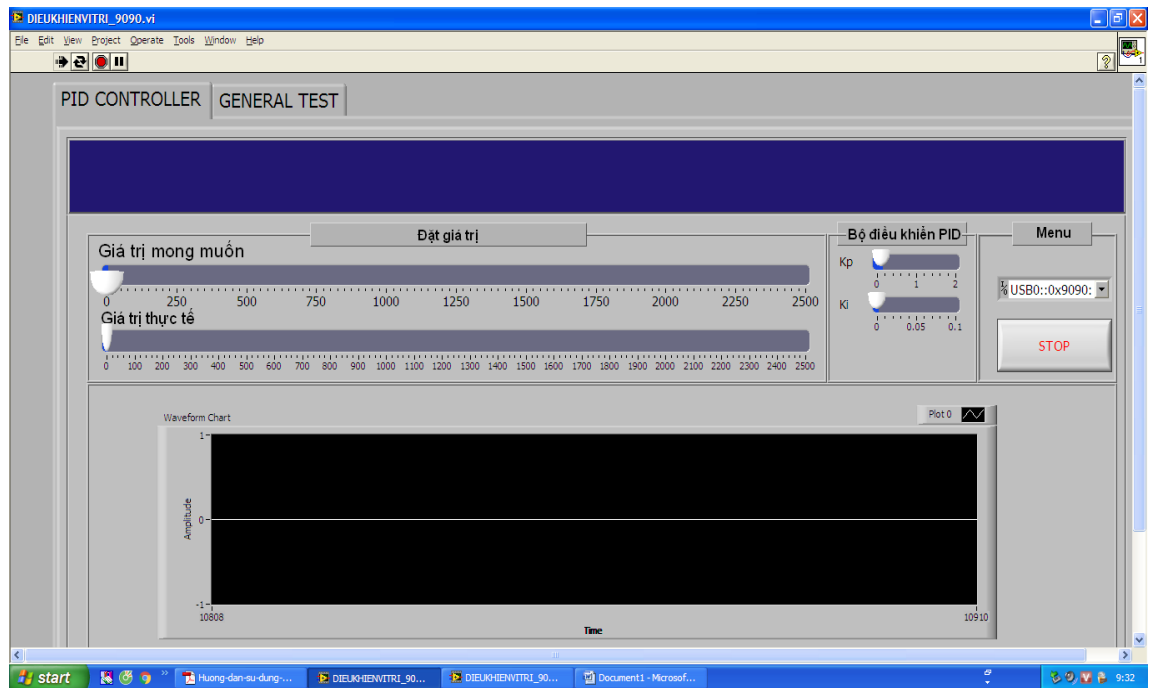
### 3.5.2. Sơ đồ kết nối hoàn chỉnh:



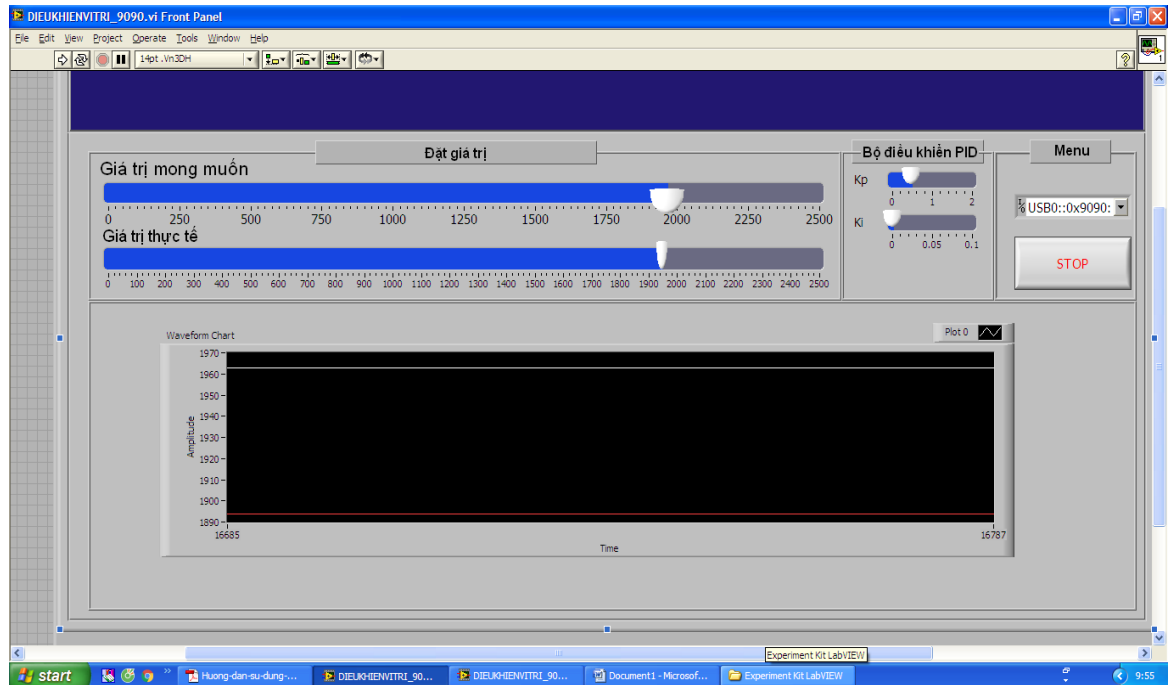
Hình 3.5: Sơ đồ kết nối phần cứng điều khiển PID

### 3.6. Chạy chương trình:

- Khi giá trị bằng 0:



- Khi giá trị đặt 2000:



Khi giá trị mong muốn đặt 2000 thì động cơ sẽ quay đồng thời encoder sẽ quay theo khi nào giá trị thực tế đạt bằng giá trị mong muốn thì sẽ dừng lại



## KẾT LUẬN

### Kết luận

Sau một thời gian nỗ lực tìm hiểu, nghiên cứu cùng với sự chỉ bảo tận tình và hỗ trợ về nhiều mặt của thầy giáo **Ths. Ngô Quang Vĩ**, đề tài: “**Điều khiển vị trí động cơ DC qua giao diện labVIEW**”. Đã giải quyết được các vấn đề sau:

- Tìm hiểu về LabVIEW cơ bản, các giao thức kết nối, phần mềm nhúng vào LabVIEW, và các bộ công cụ đi kèm.
- Xây dựng bộ điều khiển vị trí động cơ qua giao diện LabVIEW

Trong quá trình làm đồ án, mặc dù đã rất cố gắng nhưng do kiến thức và kinh nghiệm còn hạn chế nên đồ án này không thể tránh khỏi những thiếu sót. Em rất mong nhận được sự chỉ bảo đóng góp của các thầy, cô giáo và các bạn để đồ án này được hoàn thiện hơn.

Em xin chân thành cảm ơn thầy giáo **Ths. Ngô Quang Vĩ**, người đã trực tiếp tận tình hướng dẫn, chỉ bảo và tạo điều kiện cho em nghiên cứu, xây dựng thành công mô hình và hoàn thành đồ án này. Em xin cảm ơn thầy cô giáo trong bộ môn điện công nghiệp trường ĐHDL Hải Phòng, các bạn sinh viên lớp DC1901 đã đưa ra nhiều góp ý để hoàn thiện đồ án.

Em xin chân thành cảm ơn!

Hải Phòng, ngày tháng năm 2020

Sinh viên thực hiện

Giang

PHẠM NGỌC GIANG