

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/281631566>

Speed Control of Dc Motor By PWM Method Using Microcontroller

Article · May 2012

DOI: 10.13140/RG.2.1.1724.7845

CITATIONS

5

READS

13,700

2 authors:



Trần Quốc Cường

Tien Giang University

5 PUBLICATIONS 5 CITATIONS

[SEE PROFILE](#)



Thanh Phong TRAN

University of Angers

26 PUBLICATIONS 25 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Article in Journal of science and technology of Tien Giang University [View project](#)

ĐIỀU KHIỂN TỐC ĐỘ ĐỘNG CƠ DC BẰNG PHƯƠNG PHÁP ĐIỀU CHẾ ĐỘ RỘNG XUNG SỬ DỤNG VI ĐIỀU KHIỂN

Speed Control of Dc Motor By PWM Method Using Microcontroller

TRẦN QUỐC CƯỜNG¹

TRẦN THANH PHONG²

TÓM TẮT

Bài báo này thực hiện phương pháp điều chế độ rộng xung (PWM) để điều khiển tốc độ động cơ một chiều (DC) trên cơ sở các họ vi điều khiển 8 bit. Nguyên tắc cơ bản của phương pháp PWM và các giải thuật lập trình tạo xung PWM cho các họ vi điều khiển khác nhau như: 8051, P89V51RD2, PIC 16F877A, ATMEGA16 sẽ được đề xuất. Việc điều khiển động cơ DC bằng phương pháp đề xuất sẽ được minh họa qua các kết quả mô phỏng bằng Proteus.

ABSTRACT

This paper presents an approach of DC motor PWM speed control based on 8 bit microcontrollers. The basic principle of the approach DC motor PWM speed control and the program algorithms to generate PWM wave for the different microcontrollers as: AT89C51, P89V51RD2, PIC 16F877A, ATMEGA 16 are proposed. Proteus simulation results of the proposed approach for Speed Control Of Dc Motor are provided for illustration.

I. GIỚI THIỆU

Động cơ một chiều được sử dụng phổ biến và rộng rãi trong tất cả các lĩnh vực từ quân sự đến công nghiệp và dân dụng. Những ứng dụng quan trọng của nó bao gồm: nhà máy cán, nhà máy giấy, nhà máy dệt, nhà máy in, máy công cụ, máy xúc, cần cẩu và đặc biệt là lĩnh vực robotic [2]...

Các mạch điều khiển động cơ yêu cầu thay đổi tốc độ quay của động cơ nhịp nhàng và điều khiển chính xác. Phương pháp truyền thống để điều khiển tốc độ động cơ một chiều là thay đổi giá trị điện áp cung cấp cho động cơ. Phương pháp đơn giản nhất là sử dụng biến trở. Phương pháp điều khiển này không chính xác như mong muốn do đặc tuyến của biến trở, tầm hoạt động bị giới hạn, điều khiển không hiệu quả và gây ra hiện tượng quá nhiệt của cuộn dây dẫn đến hư động cơ.

Phương pháp PWM được biết đến từ những năm 1970 cải thiện được hạn chế của các phương pháp truyền thống, tuy nhiên mạch điều khiển dùng linh kiện rời BJT hoặc vi mạch số nên mạch điện phức tạp, khó đạt được độ chính xác cao. Ngày

^{1,2} BỘ MÔN ĐIỆN ĐIỆN TỬ – KHOA KỸ THUẬT CÔNG NGHIỆP

nay việc sử dụng các vi mạch khả lập trình như vi xử lý trong các thiết bị điều khiển trở thành một xu thế quan trọng, mang lại hiệu quả cao, tốc độ xử lý nhanh, độ chính xác cao, mạch phần cứng tinh gọn, giảm giá thành sản phẩm, hạn chế rủi ro.

Trong bài báo này chúng tôi tập trung nghiên cứu phương pháp điều chế độ rộng xung PWM để điều khiển tốc độ động cơ DC sử dụng các dòng vi điều khiển phổ biến khác nhau như: AT89C51, P89V51RD2, PIC 16F877A và AVR ATMEGA 16... Các giải thuật tạo xung PWM và lập trình cho vi điều khiển bằng ngôn ngữ C sẽ được trình bày khá chi tiết.

Bộ cục của bài báo này gồm các phần như sau: Phần 2 trình bày lý thuyết cơ bản của phương pháp điều chế độ rộng xung. Các giải thuật lập trình tạo xung PWM cho từng họ vi điều khiển được thể hiện trong phần 3. Phần 4 kết quả mô phỏng để minh họa. Cuối cùng, phần 5 đưa ra kết luận.

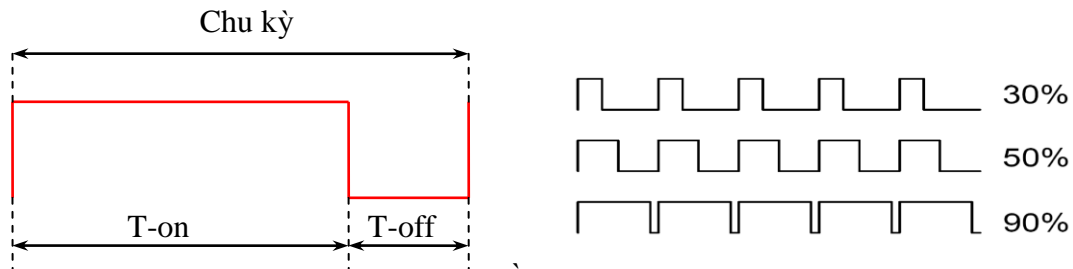
II. PHƯƠNG PHÁP ĐIỀU CHẾ ĐỘ RỘNG XUNG (PULSE WIDTH MODULATION PWM)

Tốc độ quay của động cơ DC tỷ lệ thuận với điện áp đặt vào nó. Do đó, cách đơn giản nhất để điều khiển tốc độ quay của rotor là thay đổi mức điện áp đặt vào động cơ. PWM là một phương pháp rất hiệu quả trong việc cung cấp ngay lập tức điện áp giữa mức cao và mức thấp của nguồn điện giúp động cơ thay đổi tốc độ mượt hơn so với phương pháp cổ điển. Với công tắc đơn giản và một bộ nguồn thông dụng, chúng ta chỉ có thể cung cấp điện áp lớn nhất của bộ nguồn khi đóng công tắc, nghĩa là động cơ sẽ chạy với vận tốc tối đa. Và ngược lại khi hở công tắc động cơ sẽ tắt hẳn.

Nguyên tắc cơ bản của phương pháp điều khiển PWM là giữ nguyên giá trị điện áp và thay đổi thời gian đặt điện áp vào động cơ. Điều này có nghĩa, với tần số đóng ngắt công tắc đủ lớn, thời gian cấp điện áp vào động cơ càng lâu thì điện áp trung bình càng cao, ngược lại thời gian cấp điện vào động cơ ngắn điện áp trung bình giảm. Như vậy, PWM là một kỹ thuật so sánh tỷ lệ phần trăm điện áp nguồn bằng cách đóng ngắt nhanh nguồn điện cấp vào động cơ tạo ra một tín hiệu xung, với độ rộng xung (thời gian cấp điện áp) xác định sẽ tạo ra một điện áp trung bình xác định (được minh họa như hình 1). Khi tần số đóng ngắt đủ lớn (thường từ 1 ÷ 20 kHz), động cơ sẽ chạy với một tốc độ ổn định nhờ moment quay.

Đại lượng mô tả mối quan hệ giữa khoảng thời gian T-on và T-off được gọi là độ rộng xung (duty cycle)

$$\text{duty_cycle} = \frac{T_{\text{on}}}{T_{\text{on}} + T_{\text{off}}} \times 100$$



Hình 1. Giải đồ thời gian xung PWM

III. PHƯƠNG PHÁP TẠO XUNG PWM SỬ DỤNG VI ĐIỀU KHIỂN

3.1 PWM BẰNG PHẦN MỀM

Một số họ vi xử lý và vi điều khiển (chẳng hạn AT89C51) không hỗ trợ điều chế độ rộng xung PWM bằng phần cứng, nên việc dùng phần mềm để tạo xung PWM sử dụng các timer là cần thiết. Việc này được thực hiện bằng cách xuất ra một chân nào đó bất kỳ của vi điều khiển tín hiệu xung có khoảng thời gian T-on và T-off khác nhau tùy thuộc vào độ rộng xung.

Ví dụ: Điều xung với độ rộng xung = 75% trên chân P1.0 của AT89C51

chọn $f = 10\text{kHz} \rightarrow$ chu kỳ $T = 1/f = 100\text{ms}$

độ rộng xung = 75% nên: $T\text{-on} = 75\text{ms}$ và $T\text{-off} = 25\text{ms}$

Chương trình:

// Chương trình khởi tạo ngắt timer

```
TMOD=0x10;      //timer 1 mode 1 (16 bits không tự nạp lại giá trị đầu)
EA = 1 ;        // cho phép ngắt
ET1 = 1 ;       // cho phép ngắt timer 1
TR1=1;
duty_cycle = 75; //0 ≤ duty_cycle ≤ 100
```

// Chương trình phục vụ ngắt (ISR)

```
void ISR_T1() interrupt 3
{
    TR1 = 0;      //Tắt timer
    if(P1^0 == 0) //Nếu T-off
    {
        //Nạp giá trị cho T-on = 75ms
        TH1 = -(duty_cycle*1000)/256;
        TL1 = -(duty_cycle*1000)%256;
        P1^0 = 1; //T-on
    }
}
```

```

    if(P1^0 == 1)          //Nếu T-off
    {
        //Nạp giá trị cho T-off = 25ms
        TH1 = -((100-duty_cycle)*1000)/256;
        TL1 = -((100-duty_cycle)*1000)%256;
        P1^0 = 0;
    }
    TR1 = 1;                //Chạy timer
}

```

3.2 PWM BẰNG PHẦN CỨNG

3.2.1 P89V51RD2

Vì điều khiển P89V51RD2 hỗ trợ đến 5 kênh điều rộng xung PWM (CEX0 – CEX4 trên chân P1.3 – P1.7) được sử dụng khá linh hoạt cho việc điều xung. Chúng ta chỉ cần thiết lập đúng các thông số thích hợp ở các thanh ghi để xuất ra tín hiệu xung PWM như mong muốn. Giá trị ở ngõ ra CEXn phụ thuộc vào giá trị của 2 thanh ghi: CCAPnL và CL.

Khi CL tăng lớn hơn CCAPnL thì CEXn sẽ chuyển sang mức 1, ngược lại, sẽ ở mức 0. Mỗi lần tràn cờ, CCAPnL sẽ được nạp lại giá trị từ CCAPnH, chính điều này cũng cho phép ta cập nhật độ rộng xung mới mà không gây ảnh hưởng tới quá trình PWM. Chức năng PWM của P89V51RD2 có độ phân giải 8 bit, tức là ta được 256 mức chia.

//Chương trình thiết lập thông số cho các kênh PWM:

```

void init_PWM()
{
    CCAPMn = 0x42;          //n là kênh 0 đến kênh 4, thiết lập chế độ PWM 8 bit
    CMOD = 0x00;            //xung nhịp cho PCA fosc/6
    CCAPnH = 0xFF;          //duty_cycle = 0%
    CCON |= (1<<CR);        //set bit điều khiển chạy counter PCA
}

```

Để thay đổi tốc độ động cơ, chúng ta chỉ việc chọn kênh cần xuất và nạp giá trị vào các thanh ghi CCAPnH như sau: (độ rộng xung = 100% ứng tốc độ cao nhất, độ rộng xung = 0% tốc độ thấp nhất).

// Chương trình tạo xung cho 5 kênh PWM

```

void PWM_out(char kênh, char dutyCircle)
{
    if(dutyCircle > 100) dutyCircle = 100;
    switch(kênh) {
        case 0:
            CCAP0H = ((100-dutyCircle)*256)/100;
            break;
        case 1:
            CCAP1H = ((100-dutyCircle)*256)/100;
            break;
        case 2:
            CCAP2H = ((100-dutyCircle)*256)/100;
            break;
        case 3:
            CCAP3H = ((100-dutyCircle)*256)/100;
            break;
        case 4:
            CCAP4H = ((100-dutyCircle)*256)/100;
            break;}}

```

3.2.2 PIC 16F877A

Vi điều khiển PIC16F877A có hỗ trợ 2 kênh điều xung bằng phần cứng ở 2 chân **C1 (CCP2)** và **C2(CCP1)** sử dụng **TIMER2**. Khi khai báo điều xung PWM ở một tần số và độ rộng xung cụ thể nào đó thì vi điều khiển sẽ thực hiện công việc xuất xung một cách liên tục và tự động cho đến khi có sự thay đổi các giá trị độ rộng xung mới.

Các hàm hỗ trợ tạo xung PWM trong CCS:

- **setup_timer_2 (mode, period, postscale)** // hàm thiết lập timer 2
 - mode: T2_DIV_BY_1, T2_DIV_BY_4, T2_DIV_BY_16
 - period: $0 \div 255$
 - postscale: 1

$$f_{PWM} = \frac{f_{osc}}{4 \cdot mode \cdot (period + 1)}$$

- **setup_ccp1(mode) và setup_ccp2(mode)** // hàm thiết lập chế độ PWM
 - mode:
 - CCP_PWM: chọn chế độ PWM.
 - CCP_OFF: tắt chế độ PWM.

- **set_pwm1_duty(value)** và **set_pwm2_duty(value)** // hàm thiết lập giá trị PWM

- Nếu value là giá trị kiểu int 8 bit:

$$duty_cycle = \frac{value}{period + 1}$$

- Nếu value là giá trị long int 16 bit:

$$duty_cycle = \frac{value \& 1023}{4 \cdot (period + 1)}$$

Ví dụ: Ta muốn điều xung PWM với tần số 20kHz với tần số thạch anh (f_{osc}) sử dụng là 20MHz (8 bit).

$$f = \frac{f_{osc}}{4 \cdot mode \cdot (period + 1)} \Leftrightarrow 20000 = \frac{20000000}{4 \cdot mode \cdot (period + 1)} \Leftrightarrow mode \cdot (period + 1) = 250$$

Với mode = [1, 4, 16] và period = [0, 255] ta có thể chọn

$$mode = 1 ; period = 249$$

$$mode = 4 ; period = 62$$

$$mode = 16 ; period = 30$$

Để cho việc điều xung được “mịn” (chọn được nhiều giá trị độ rộng xung) ta chọn mode = 1 và period = 249.

Như vậy, để duty_cycle từ 0% đến 100% ta cho value từ 0 đến 250.

$$duty_cycle = \frac{value}{period + 1} \Rightarrow value = \frac{duty_cycle}{250} \times 100$$

Ta viết mã nguồn như sau :

```
setup_timer_2(T2_DIV_BY_1, 249, 1);
setup_ccp1(CCP_PWM);
set_pwm1_duty(value);
```

3.2.3 ATMEGA 16

Vì điều khiển ATmega16 được trang bị 3 Timer, chúng ta có thể thiết lập từ 3 đến 4 kênh PWM tại các chân OC0, OC2, OC1A, OC1B. Trong đó có 2 kênh PWM 8 bit của Timer 0 và Timer 2, Timer 1 có thể cấu hình thành 1 kênh PWM 10 bit hoặc 2 kênh PWM 8bit. Để thay đổi độ rộng xung ngõ ra trên các chân của ATmega16 cần nạp giá trị vào các thanh ghi OCR0, OCR2, OCR1A, OCR2A. Cách thiết lập các Timer0 và Timer2 để sử dụng chế độ PWM khá giống nhau [3,4], riêng Timer 1 có 1 ít khác biệt. Giải thuật chương trình điều khiển tốc độ động cơ bằng PWM sử dụng Timer2 được thể hiện như sau:

//Chương trình khởi tạo PWM

```

{
    PORTD=0x00;    //reset port D
    DDRD=0xFF;     //thiết lập port D xuất
    ASSR=0x00;
    TCCR2=0x65;    //timer 2 chế độ PWM 8 bit
    TCNT2=0x00;    //xóa giá trị thanh ghi timer/counter
    OCR2=0x00;     //xóa giá trị thanh ghi so sánh
}

```

//Chương trình PWM

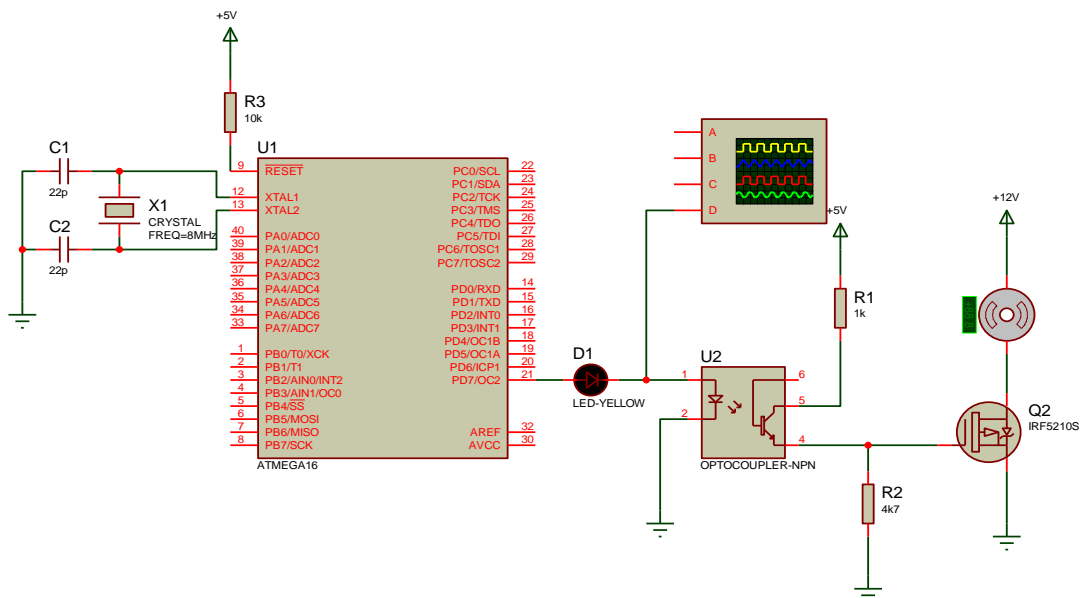
```

void PWM_OUT(unsigned char value)
{
    OCR2 = value*256/100; //value = 100, ngõ ra PWM là 1
}

```

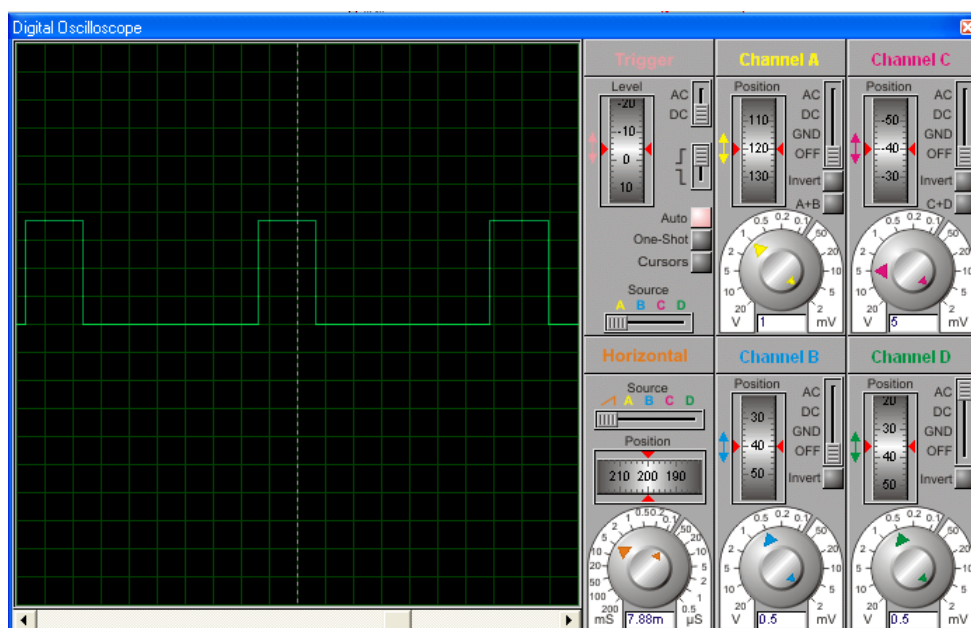
3.3 KẾT QUẢ THỰC NGHIỆM**3.3.1 Kết quả mô phỏng**

Sau đây là sơ đồ mạch điều khiển và kết quả mô phỏng kỹ thuật PWM cho vi điều khiển ATMEGA16

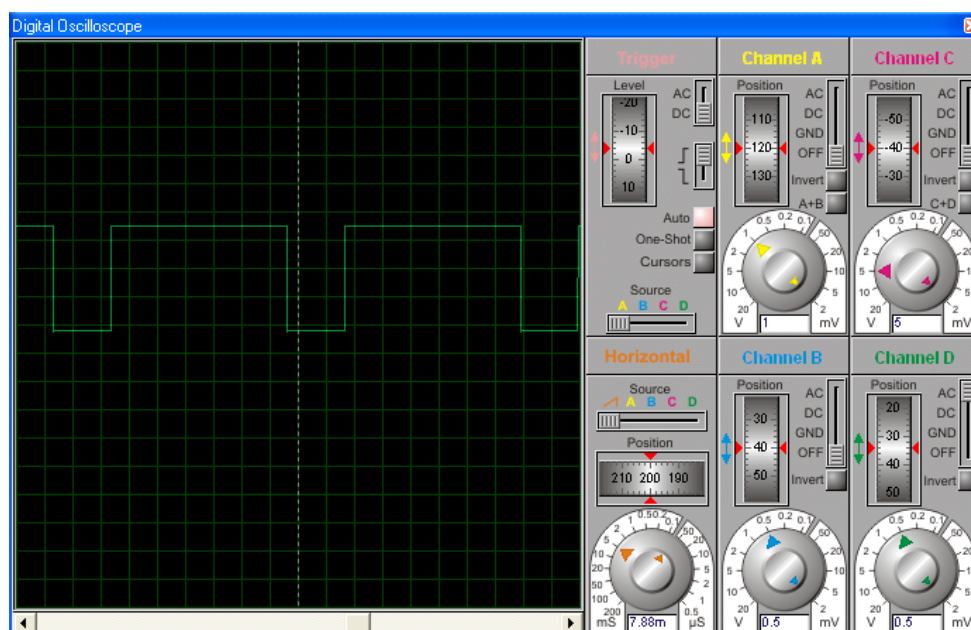


Hình 2. Sơ đồ mạch mô phỏng dùng vi điều khiển ATMEGA16

Kết quả mô phỏng:



Hình 3(a). Kết quả xung PWM trên dao động ký, độ rộng xung 25%



Hình 3(b). Kết quả xung PWM trên dao động ký, độ rộng xung 75%

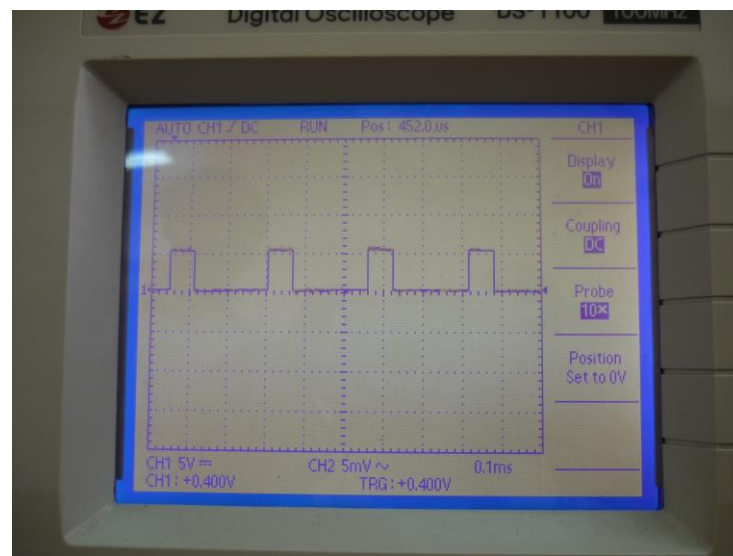
Từ kết quả mô phỏng hình 3 (a) và (b) cho thấy, xung PWM hiển thị trên dao động ký lần lượt có độ rộng (value) là 25% và 75% đúng như mong muốn.

3.3.2 Kết quả thực nghiệm

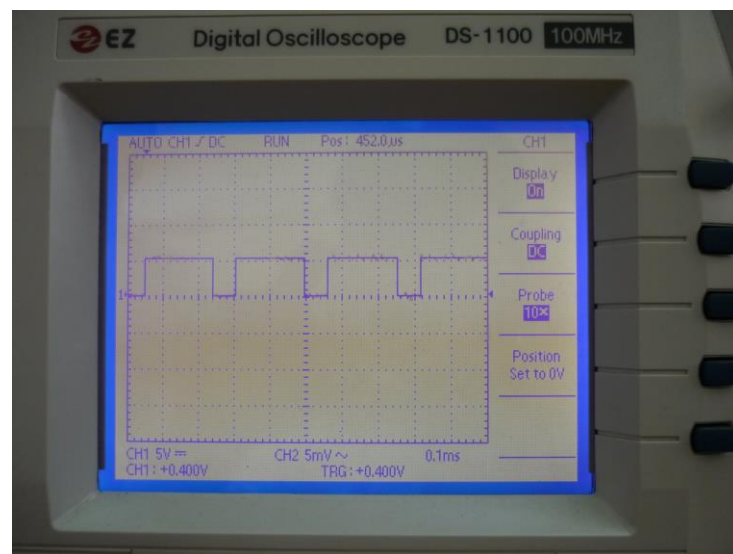
Mạch thực nghiệm sử dụng vi điều khiển P89V51RD2 để điều khiển động cơ DC 12V, hiển thị kết quả xung PWM trên dao động ký.



Hình 4. Mô hình mạch thực nghiệm P89V51RD2



Hình 5 (a). Kết quả xung PWM hiển thị trên dao động ký độ rộng xung 25%



Hình 5 (b). Kết quả xung PWM hiển thị trên dao động ký độ rộng xung 75%

Từ kết quả thực nghiệm hình 5 (a) và (b) cho thấy, xung PWM hiển thị trên dao động ký lần lượt có độ rộng (value) là 25% và 75% khá chính xác: ứng với độ rộng 25% ta thấy động cơ quay với tốc chậm và với độ rộng 75% ta thấy động cơ quay với tốc khá nhanh. Khi cấp nguồn 12V trực tiếp vào động cơ, chúng tôi nhận thấy rằng động cơ quay với độ nhanh nhất. Điều đó chứng tỏ rằng kỹ thuật PWM rất hiệu quả trong việc điều khiển tốc độ động cơ.

IV. KẾT LUẬN

Bài báo này giới thiệu phương pháp điều khiển tốc độ động cơ DC bằng kỹ thuật PWM sử dụng các họ vi điều khiển thông dụng. Chúng tôi đề xuất các giải thuật chương trình tạo xung PWM cho từng họ vi điều khiển cụ thể. Kết quả mô phỏng và thực nghiệm cho thấy rằng kỹ thuật PWM rất hiệu quả trong việc điều khiển tốc độ động cơ DC cũng như động cơ Servo và có khả năng ứng dụng cao trong thực tiễn sản xuất. Tuy nhiên, đối với các ứng dụng yêu cầu điều khiển vị trí một cách chính xác thì PWM vòng hở chưa đáp ứng tốt mà phải sử dụng kỹ thuật PWM vòng kín. Kỹ thuật này sẽ được giới thiệu trong các bài báo tới.

REFERENCE

- [1]. Cao Hoàng Long, EM-BOT Robot Team (2010), *Lập trình Robot tự động đơn giản với PIC16F877A*, Khoa Công nghệ, Đại học Cần Thơ.
- [2]. I.MOAZZEM, S.RAHMAN, M A MATIN, Microcontroller based closed-loop automatic speed control of DC motor using PWM, Dept. of Electrical Engineering and Computer Science, North South University Dhaka, Bangladesh, ISBN: 978-960-474-27-4.
- [3]. Steven F.Barrett and Daniel J.Park, (2008), *Atmel AVR Microcontroller Primer: Programming and Interfacing*, ISSN: 1932-3174.
- [4]. Kiều Xuân Thực, (2008), *Vi điều khiển cấu trúc - lập trình và ứng dụng*, NXB Giáo Dục.
- [5]. Zhijun Liu and Lianzhi Jiang, (12-14 Aug. 2011), PWM speed control system of DC motor based on AT89S51, Sch. of Electr. & Inf. Eng., Liaoning Inst. of Sci. & Technol., Benxi, China, IEEE ISBN: 978-1-61284-087-1.

Liên hệ nhóm tác giả

Ths. Trần Thanh Phong

Bộ môn Điện - Điện tử, Khoa Kỹ thuật Công Nghiệp

ĐT: 0908.536 689. Email: tranthanhphong@tgu.edu.vn

Ths. Trần Quốc Cường

Bộ môn Điện - Điện tử, Khoa Kỹ thuật Công Nghiệp

ĐT: 0975.257.258. Email: tranquoccuong@tgu.edu.vn