

THIẾT KẾ XÂY DỰNG RƠ LE KỸ THUẬT SỐ BẢO VỆ QUÁ ĐIỆN ÁP CÓ ĐẶC TÍNH BẢO VỆ ĐỘC LẬP PHỤC VỤ CÔNG TÁC GIẢNG DẠY TẠI KHOA ĐIỆN – TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP – ĐẠI HỌC THÁI NGUYÊN

Cao Xuân Tuyền*, Nguyễn Thị Hương

Trường Đại học Kỹ thuật Công nghiệp - ĐH Thái Nguyên

TÓM TẮT

Mục tiêu của bài báo này là trình bày việc thiết kế xây dựng rơ le bảo vệ quá áp kỹ thuật số với mục đích phục vụ công tác giảng dạy tại khoa Điện trường đại học Kỹ thuật Công nghiệp Thái Nguyên. Bài báo lần lượt trình bày thiết kế phần cứng trên cơ sở sử dụng vi điều khiển Atmega16, là loại vi điều khiển phổ biến trên thị trường, thiết kế phần mềm và các thí nghiệm đối với rơ le kỹ thuật số bảo vệ quá điện áp đã thiết kế xây dựng nhằm đánh giá, khẳng định chất lượng của rơ le. Những nội dung trên cũng là những thông tin hữu ích cho các sinh viên học tập, nghiên cứu các rơ le kỹ thuật số nói chung.

Từ khóa: Rơ le kỹ thuật số; bảo vệ quá áp; đặc tính thời gian độc lập; Vi điều khiển Atmega16; trình biên dịch CodeVisionAvr.

Ngày nhận bài: 13/5/2020; Ngày hoàn thiện: 31/5/2020; Ngày đăng: 31/5/2020

DESIGN, CONSTRUCTION OF DIGITAL OVER VOTAGE PROTECTION RELAY WITH INDEPENDENT TIME CHARACTERISTICS FOR TEACHING AT THE DEPARTMENT OF ELECTRICAL ENGINEERING AT UNIVERSITY OF TECHNOLOGY - THAI NGUYEN UNIVERSITY

Cao Xuan Tuyen*, Nguyen Thi Huong

TNU - University of Technology

ABSTRACT

The objective of this paper is to present the design and construction of digital overvoltage protection relays with independent protection characteristics for the purpose of teaching at the Department of Electrical Engineering at University of Technology, Thai Nguyen University. The article in turn presents hardware design based on Atmega16 microcontroller, which is a popular microcontroller in the market, software design and experiments for digital over-voltage protection relays. The purpose of these experiments is to assess and confirm the quality of the relay. The above content is also useful information for students to study digital relays in general.

Keywords: digital relay; overvoltage protection; independent time characteristics; Atmega16 microcontroller; CodeVisionAvr Compiler.

Received: 13/5/2020; Revised: 31/5/2020; Published: 31/5/2020

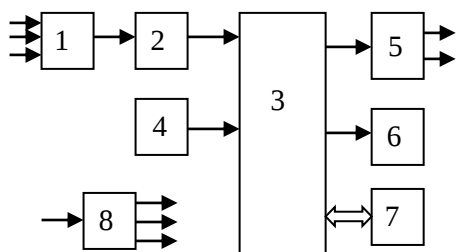
* Corresponding author. Email: tuyenkdmd@gmail.com

1. Đặt vấn đề

Hiện nay rơ le kỹ thuật số đã có mặt ở hầu hết trong công nghiệp cũng như trong đời sống sinh hoạt nhờ những ưu điểm nổi trội của nó gắn với cuộc cách mạng 4.0 [1]-[3]. Trong công tác giảng dạy tại Khoa Điện, Trường Đại học Kỹ thuật Công nghiệp - ĐH Thái Nguyên, rơ le kỹ thuật số là một nội dung quan trọng. Tuy nhiên các rơ le kỹ thuật số do các hãng nổi tiếng trên thế giới chế tạo chỉ phục vụ cho mục đích sử dụng, còn về kết cấu chi tiết phần cứng cũng như phần mềm và kỹ thuật công nghệ để tạo ra phần cứng và phần mềm cho các rơ le kỹ thuật số thì các hãng sản xuất giữ bí mật. Điều này gây khó khăn cho công tác giảng dạy sinh viên để sinh viên nắm bắt được cốt lõi bên trong cả phần cứng cũng như phần mềm của rơ le kỹ thuật số nói chung. Với mục đích khắc phục tồn tại trên trong công tác giảng dạy, tác giả đã thực hiện thiết kế, xây dựng rơ le kỹ thuật số bảo vệ quá điện áp có đặc tính bảo vệ độc lập cho phép sinh viên nắm bắt cụ thể về cấu tạo phần cứng và đặc biệt là kỹ năng lập trình phần mềm và kỹ thuật nạp phần mềm vào phần cứng rơ le, qua đó trang bị cho sinh viên khả năng tự chế tạo rơ le kỹ thuật số nói chung và rơ le kỹ thuật số bảo vệ quá áp nói riêng.

2. Thiết kế phần cứng

2.1. Sơ đồ nguyên lý mạch phần cứng



Hình 1. Sơ đồ nguyên lý mạch phần cứng

Theo [3], sơ đồ nguyên lý mạch phần cứng được chỉ ra ở hình 1, trong đó:

- 1- Là bộ biến đổi đầu vào tương tự; 2 - Là bộ lọc tương tự; 3 - Là vi điều khiển atmega16;
- 4 - Bàn phím; 5 - Khối khuếch đại đầu ra; 6 - mạch hiển thị màn hình tinh thể lỏng LCD; 7 - Máy tính; 8 - Khối nguồn.

2.2. Lựa chọn bộ vi điều khiển

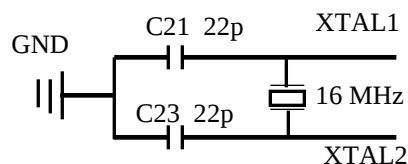
Với tính phổ biến và giá thành rẻ, đồng thời về đáp ứng được yêu cầu về chức năng đặt ra của rơ le thiết kế, đề tài lựa chọn vi điều khiển ATMEGA16 do hãng Atmel (Mỹ) sản xuất [4].

ATmega16 là vi điều khiển 8 bit dựa trên kiến trúc RISC. Với khả năng thực hiện mỗi lệnh trong vòng một chu kỳ xung clock, ATmega16 có thể đạt được tốc độ 1MIPS trên mỗi MHz (một triệu lệnh/s/MHz), tốc độ làm việc 16MIPS với thạch anh 16 MHz.

Ngoài ra ATmega16 có các đặc điểm sau: 16KB bộ nhớ Flash với khả năng đọc trong khi ghi, 512 byte bộ nhớ EEPROM, 1KB bộ nhớ SRAM, 32 thanh ghi chức năng chung, 32 đường vào ra chung, 3 bộ định thời/bộ đếm, ngắt nội và ngắt ngoại, USART, giao tiếp nối tiếp 2 dây, 8 kênh ADC 10 bit, ... ATmega 16 hỗ trợ đầy đủ các chương trình và công cụ phát triển hệ thống như: trình dịch C, macro assemblers, chương trình mô phỏng/sửa lỗi.

2.3. Thiết kế mạch dao động thạch anh

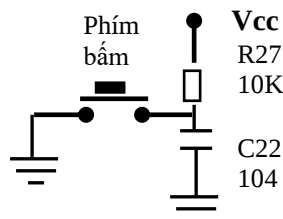
Mạch dao động thạch anh được chỉ ra ở hình 2. Trong đó 2 chân XTAL1, XTAL2 của vi điều khiển được nối với thạch anh 16 MHz và 2 tụ gốm C21, C22 có trị số 22pF và nối đất.



Hình 2. Mạch dao động thạch anh

2.4. Thiết kế mạch RESET

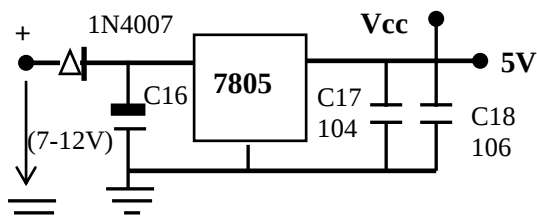
Mạch Reset nối vào chân RST của vi điều khiển để khởi động cứng lại mọi hoạt động của hệ thống được chỉ ra ở hình 3. Trong đó: R27 là điện trở 10 kΩ, C22 là tụ gốm 104, GND là mức 0 V.



Hình 3. Mạch RESET

2.5. Thiết kế khối nguồn

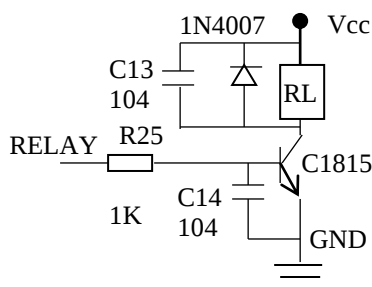
Vì Atmega16 và các phần tử khác trong rơ le yêu cầu nguồn cung cấp là 5 V có độ ổn định cao, nên bài báo lựa IC ổn áp 7805. Sơ đồ mạch nguồn như hình 4, trong đó 1N4007 là diode chỉnh lưu phụ, C16 là tụ phân cực (100 μ F, 16 V), C17 và C18 là các tụ gốm.



Hình 4. Mạch nguồn

2.6. Thiết kế khối khuếch đại đầu ra

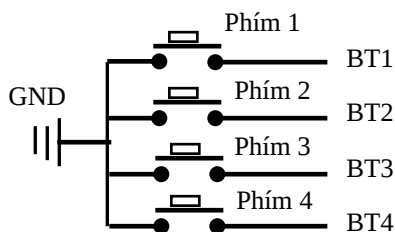
Sơ đồ khối khuếch đại đầu ra được chỉ ra ở hình 5. Trong đó, RL là rơ le điện từ trung gian loại HUIKE HK4100F – DC5V (3A-6 chân); C1815 là transistor khuếch đại.



Hình 5. Khối khuếch đại đầu ra

2.7. Thiết kế mạch phím đơn

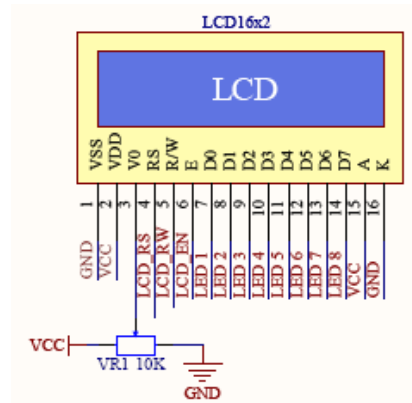
Mạch phím đơn có nhiệm vụ thay đổi điện áp đặt, thay đổi thời gian đặt cho rơ le. Sơ đồ nguyên lý mạch phím đơn được chỉ ra ở hình 6. Trong đó GND – là nối đất; BT1, BT2, BT3, BT4 là các đầu nối với cổng PB.0, PB.1, PB.2, PB.3 của vi điều khiển.



Hình 6. Mạch phím đơn

2.8. Thiết kế khối hiển thị LCD

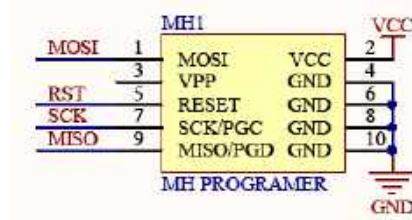
Chọn thiết bị hiển thị LCD loại 16x02 như hình 7.



Hình 7. Khối hiển thị LCD

2.9. Lựa chọn mạch nạp chương trình, đồng thời cũng là mạch ghép nối giữa máy tính và rơ le

Để ghép nối với máy tính nhằm nạp mã chương trình cho rơ le, bài báo lựa chọn mạch nạp (MN) USB ISP của Công ty Cổ phần Công nghệ và sản xuất Minh Hà. Sơ đồ chân kết nối MN và vi điều khiển (VĐK) được chỉ ra ở hình 8. Trong hình 8: chân 1 của MN nối với chân MOSI của VĐK; chân 5 của MN nối với chân RST của VĐK; chân số 7 của MN nối với chân SCK của VĐK; chân số 9 của MN nối với chân MISO của VĐK; chân số 3 của MN không sử dụng; chân số 2 của MN nối với dương nguồn (5 V); các chân số 4, 6, 8, 10 của MN nối với chân nối đất của nguồn (0 V).



Hình 8. Sơ đồ chân kết nối mạch nạp và vi điều khiển

3. Thiết kế phần mềm

Để viết chương trình phần mềm cho rơ le, bài báo sử dụng trình biên dịch trên cơ sở sử dụng ngôn ngữ lập trình C, CodeVisionAVR, là môi trường phát triển tích hợp và bộ tạo

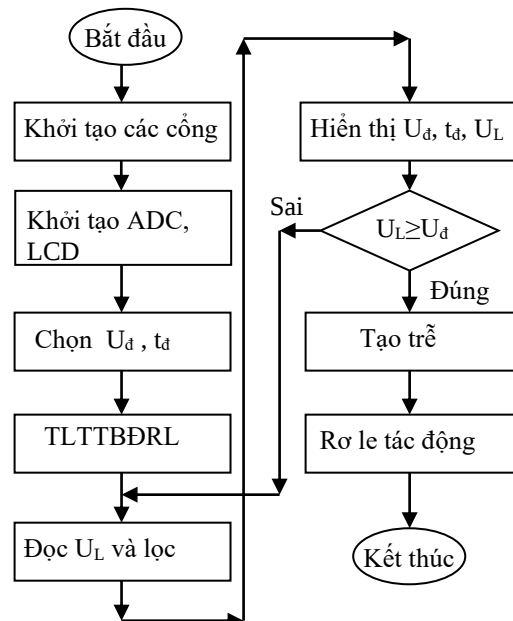
chương trình tự động được thiết kế cho họ các vi điều khiển AVR của Atmel [5]. Bên cạnh đó, tác giả viết thêm các module chương trình đặc thù căn cứ vào chức năng của rơ le.

3.1. Lưu đồ thuật toán

Lưu đồ thuật toán của rơ le được chỉ ra ở hình 9. Trong đó: U_d là trị hiệu dụng điện áp đặt; t_d – là thời gian đặt; U_L – là điện áp lưới; TLTTBĐRL – là khối thiết lập trạng thái ban đầu cho rơ le.

Hoạt động của lưu đồ thuật toán như sau:

Khi cấp nguồn cho rơ le, phần mềm trong rơ le sẽ khởi tạo các cổng vào/ra cho VĐK, cụ thể các cổng của khối ADC (PORTA) của VĐK là cổng vào, các cổng nối với bàn phím (PORTB.0 đến PORTB.3 là cổng vào, các cổng nối với màn hình hiển thị tinh thể lỏng LCD là các cổng ra (các cổng PORTC.3, PORTC.4, PORTC.5, PORTB.4, PORTB.5, PORTB.6, PORTB.7)). Tiếp theo phần mềm khởi tạo khối chuyển đổi tương tự số để kích hoạt khối ADC trong vi điều khiển hoạt động nhằm đọc điện áp lưới tương tự lấy từ đầu ra của máy biến áp cách ly. Đồng thời với khởi tạo ADC, phần mềm cũng thực hiện khởi tạo khối hiển thị màn hình tinh thể lỏng LCD, mục đích là kích hoạt thiết bị LCD sẵn sàng làm việc, hiển thị các tham số của rơ le như điện áp đặt, điện áp lưới, thời gian đặt. Tiếp theo, phần mềm cho phép người sử dụng chọn, thay đổi điện áp đặt U_d , thời gian đặt t_d , thông qua bàn phím, đồng thời đặt rơ le ở trạng thái ban đầu (chưa tác động). Tiếp theo phần mềm thực hiện đọc dữ liệu U_L và lọc tín hiệu đọc vào, đồng thời hiển thị các tham số rơ le trên LCD và thực hiện so sánh điện áp lưới đọc được với điện áp đặt, nếu $U_L \geq U_d$ thì kích hoạt thủ tục tạo trễ thời gian, thời gian trễ này càng nhỏ nếu điện áp càng lớn. Vì nội dung bài báo là xây dựng đặc tính bảo vệ độc lập, nên thời gian trễ được giữ là hằng số ứng với mỗi khoảng điện áp xác định, các khoảng điện áp khác nhau có thời gian trễ khác nhau. Ngược lại, nếu $U_L < U_d$ thì rơ le tiếp tục đọc U_L và tiếp tục hiển thị các tham số rơ le và lại so sánh với điện áp đặt. Quá trình này được thực hiện trong vòng lặp vô hạn.



Hình 9. Lưu đồ thuật toán

Đánh giá lưu đồ thuật toán trong hình 9: lưu đồ thuật toán đã thực hiện được các chức năng cơ bản của một rơ le số đơn chức năng là bảo vệ quá áp với đặc tính thời gian độc lập theo tiêu chuẩn châu Âu, tuy nhiên vẫn còn một số hạn chế sau: chưa lưu giữ được thông tin sự cố và chưa thực hiện được việc truyền dữ liệu tới trạm điều khiển trung tâm. Đây cũng là hướng nghiên cứu tiếp theo của các tác giả.

3.2. Viết chương trình sử dụng công cụ tạo chương trình tự động của phần mềm CodeVisionAVR

3.2.1. Tạo project

Để tạo mã chương trình cho rơ le, bài báo sử dụng phần mềm CodeVisionAvr và thực hiện: Chọn File/New→chọn project→OK

Sau đó chọn sử dụng trợ giúp của phần mềm CodeVisionAvr.

3.2.2. Chọn vi điều khiển

Thông qua phần trợ giúp viết chương trình tự động của CodeVision Avr, chọn vi điều khiển Atmega16, chọn tần số xung nhịp là 16 MHz và chọn loại loại chương trình là chương trình ứng dụng.

3.2.3. Thiết lập cổng vào/ra

Các cổng vào là các cổng nối với bàn phím, nối với khối xử lý trung gian để nhận tín hiệu điện áp lưới đưa vào vi điều khiển, các cổng ra là các cổng nối với mạch hiển thị. Các cổng vào thì chọn là IN, các cổng ra thì chọn là OUT. Các thao tác này được thực hiện thông qua phần trợ giúp viết chương trình tự động của CodeVision Avr.

3.2.4. Khởi tạo khối chuyển tương tự- số ADC

Để chuyển tín hiệu áp tương tự thành tín hiệu số, ta phải sử dụng khối ADC trong vi điều khiển.

Để kích hoạt khối ADC trong vi điều khiển hoạt động, ta phải thực hiện khởi tạo ADC sử dụng phần trợ giúp viết chương trình tự động của CodeVision Avr.

3.2.5. Khởi tạo màn hình hiển thị tinh thể lỏng LCD

Để sử dụng được các thủ tục hiển thị số, ký tự trên LCD có trong thư viện của CodeVisionAVR, ta phải thực hiện khởi tạo LCD sử dụng phần trợ giúp viết chương trình tự động của CodeVision Avr.

Ở bước này, ta phải khai báo việc nối các chân điều khiển và dữ liệu của LCD với vi điều khiển cũng như loại LCD mà ta đã sử dụng trong rơ le.

3.2.6. Tạo code chương trình

Phần mềm CodeVisionAVR sẽ tự động tạo code chương trình cho những phần đã khởi tạo ở trên thông qua thực hiện việc chọn **Program/Generate, Save and Exit** nhờ phần trợ giúp viết chương trình tự động của CodeVision Avr.

3.3. Các module chương trình chức năng đặc thù

3.3.1. Module hiển thị hệ cơ số 10 trên LCD

Chức năng của module này là hiển thị giá trị điện áp lưới đọc được từ lưới nhờ khối ADC trong bộ vi điều khiển và các giá trị điện áp đặt, thời gian đặt (thiết lập qua bàn phím) ở dạng hệ cơ số 10 ở trên LCD.

```
void lcd_putnum(signed int so){
    unsigned char a,b,c,d;
    a=so/1000; b=(so-1000*a)/100;
```

```
c=(so-1000*a-100*b)/10;
d=(so-1000*a-100*b-10*c);
if(a>0){
    lcd_putchar(48+a); lcd_putchar(48+b);
    lcd_putchar(48+c); lcd_putchar(48+d);
    .....
}
```

3.3.2. Module chống rung bàn phím

Bàn phím của chúng ta là bàn phím cơ học, bề mặt tiếp xúc của cơ cấu bên trong phím không phải là phẳng lí tưởng, do vậy, mỗi khi bấm phím hay nhả phím, xung vào vi điều khiển sẽ không phải là 1 xung thẳng đứng, mà là rất nhiều xung kim. Vì thời gian quét của vi điều khiển rất nhanh, nên tất cả các giá trị tại thời điểm rung đó đều được ghi lại. Chúng ta phải tìm cách sao cho vi điều khiển không lấy giá trị tại thời điểm rung. Có 2 phương pháp chống rung là chống rung bằng phần cứng và chống rung bằng phần mềm. Ở đây bài báo sử dụng chống rung bằng phần mềm.

Mỗi khi phát hiện có tín hiệu bấm phím, chúng ta cho vi điều khiển không đọc liên tục giá trị của phím nữa bằng cách cho delay một khoảng thời gian, khoảng trên 10 ms, sau khoảng thời gian đó, chúng ta lại đọc phím như bình thường. Ví dụ code như sau:

```
If(phát hiện bấm phím){
    Delay_ms(10);
    //Tiếp tục làm các công việc khác
    .....
}
```

3.3.3. Module thay đổi điện áp đặt

Đoạn code dưới đây sẽ cho phép tăng điện áp đặt với bước thay đổi là 0,4 V (tương ứng với 1 đơn vị của biến dienapd).

```
if (!PINB.0 {
    delay_ms(50);
    if (!PINB.0){
        if (dienapd <1020)
        {
            dienapd+= 1;
```

}

.....

Đoạn code giảm điện áp đặt được viết tương tự.

3.3.4. Đoạn code thay đổi thời gian đặt

Đoạn code tăng giảm thời gian đặt được thực hiện tương tự như đoạn code tăng giảm điện áp đặt, trong bài báo này, thời gian đặt được thay đổi với bước thay đổi là 50 ms.

3.3.5. Code tạo thời gian trễ

Để tạo thời gian trễ, ta sử dụng thủ tục:

```
delay_ms(thời gian trễ);
```

3.3.6. Code tạo rơ le trung gian tác động

Vì cuộn dây rơ le qua transistor C1815, chân điều khiển của C1815 được nối với bit 0 của PORTD, nên để rơ le tác động, ta gửi mức logic 1(5v) ra bit 0 của PORTD, ngược lại muốn rơ le trung gian trở về trạng thái ban đầu (chưa tác động), ta gửi mức logic 0(0V) ra bit 0 của PORTD với các lệnh sau:

```
PORTD.0=1; // khi cho rơ le tác động
```

```
PORTD.0=0; // khi cho rơ le ngừng tác động
```

3.3.7. Code hiển thị điện áp đặt, điện áp lưới, thời gian đặt trên LCD

Sau khi khởi tạo LCD bằng phần mềm CodeVisionAVR, phần mềm sẽ cho phép sử dụng các thủ tục hiển thị trên LCD sau:

```
lcd_clear(); // Xóa màn hình.
```

```
lcd_gotoxy(x,y); // Di chuyển con trỏ màn hình đến vị trí cột x, hàng y.
```

```
lcd_putsf("chuỗi ký tự"); // Hiển thị chuỗi ký tự.
```

3.3.8. Lệnh đọc điện áp lưới vào vi điều khiển

Sau khi khởi tạo ADC, cho phép chúng ta sử dụng hàm đọc dữ liệu vào từ khối ADC thông qua hàm read_adc():

```
Biến chứa điện áp lưới = read_adc(0);
```

3.3.9. Module theo dõi giám sát điện áp lưới cần bảo vệ

Đoạn chương trình này thực hiện theo dõi giám sát điện áp lưới cần bảo vệ, khi điện áp lưới lớn hơn hoặc bằng điện áp đặt thì sẽ gửi tín hiệu tác động tới rơ le trung gian đầu ra.

```
if (data>=dienapdat) {
```

```
    delay_ms(thời gian trễ);
```

```
    PORTD.0=1; // Rơ le tác động
```

```
    break; // thoát khỏi vòng lặp
```

```
}
```

3.4. Thực hiện biên dịch chương trình thành mã hexa và nạp mã chương trình dưới dạng file Hex vào bộ nhớ vi điều khiển

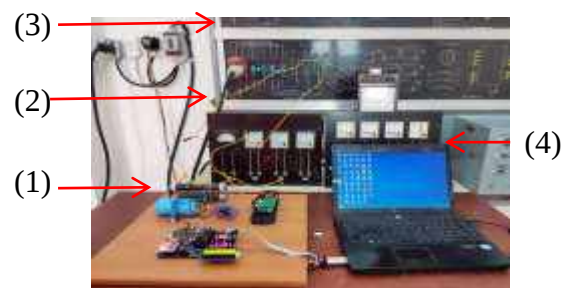
Để nạp chương trình vào bộ nhớ của vi điều khiển, ta phải thực hiện biên dịch chương trình đã viết dưới dạng ngôn ngữ C thành mã hexa sử dụng phần mềm CodeVisionAVR bằng thao tác: Project→Build All

Sau đó sử dụng phần mềm PROGISP nạp file HEX của chương trình vào bộ nhớ Flash của vi điều khiển.

4. Kiểm tra thử nghiệm, đánh giá rơ le

4.1. Hệ thống kiểm tra, thử nghiệm

Hệ thống kiểm tra, thử nghiệm rơ le như hình 10.



Hình 10. Hệ thống kiểm tra, thử nghiệm rơ le

Trong hình 10: (1) là rơ le bảo vệ quá dòng có đặc tính thời gian độc lập; (2) là khối đồng hồ đo điện áp lưới, là các đồng hồ Voltmet không thuộc rơ le; (3) là khối nguồn bên ngoài có thể điều chỉnh được điện áp thông qua máy biến áp tự ngẫu; (4) là máy tính ghép nối với rơ le.

Việc kiểm tra, thử nghiệm rơ le được thực hiện trong phòng thí nghiệm của bộ môn Thiết bị điện, Trường Đại học Kỹ thuật Công nghiệp – Đại học Thái Nguyên, Việt Nam.

4.2. Nội dung kiểm tra và thử nghiệm

4.2.1. Cài đặt điện áp đặt, thời gian đặt cho rơ le

Cài đặt điện áp đặt, thời gian đặt cho rơ le được thực hiện thông qua bàn phím và quan sát hiển thị trên màn hình LCD.

4.2.2. Xác định điện áp tác động U_{td} và điện áp trở về U_{tv} và hệ số k_{tv} của rơ le ứng với $U_{đặt} = 250, 300, 350, 400$ V

Kết quả thí nghiệm được ghi ở bảng 1.

Bảng 1. Kết quả thí nghiệm xác định điện áp tác động U_{td} , điện áp trở về U_{tv} và hệ số k_{tv} của rơ le ứng với $U_{đặt} = 250, 300, 350, 400$ V

$U_{đặt}(V)$	250	300	350	400
$U_{td}(V)$	250	300	350	400
$U_{tv}(V)$	248,8	298,8	348,8	398,8
$K_{tv}=U_{tv}/U_{td}$	0,995	0,996	0,996	0,997

Nhận xét: Từ bảng 1, cho thấy rơ le đã thay đổi được điện áp đặt, giá trị điện áp tác động bằng giá trị điện áp đặt, hệ số trở về cao (từ 0,995 đến 0,999), điều này khẳng định độ nhạy rơ le cao, đáp ứng được tiêu chuẩn của châu Âu [3].

4.2.3. Xây dựng đặc tính bảo vệ có thời gian tác động độc lập thay đổi được thời gian tác động của rơ le

Theo thiết kế, khoảng giá trị của các mức điện áp tác động và thời gian trễ tương ứng được ghi ở bảng 2.

Bảng 2. Khoảng giá trị của các mức điện áp tác động và thời gian trễ tương ứng

$U_{td}(V)$	250÷299	300÷349	350÷400
$t_d(ms)$	$3 \cdot t_d$	$2 \cdot t_d$	t_d

Bảng 3. Kết quả thí nghiệm với $t_d = 50$ ms

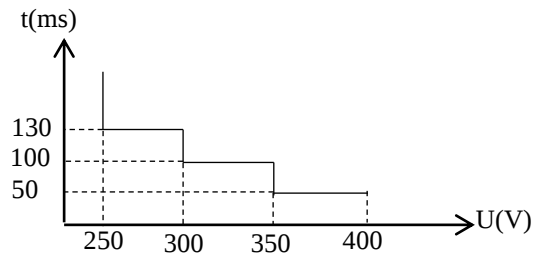
$U_{td}(V)$	250÷299	300÷349	350÷400
$t_d(ms)$	150	100	50

Nhận xét bảng 3: Kết quả thí nghiệm ở bảng 3 cho thấy thời gian tác động đúng bằng thời gian đặt thiết kế ứng với các khoảng điện áp khác nhau như thiết kế ở bảng 2.

Từ kết quả thí nghiệm bảng 3, ta vẽ được đặc tính bảo vệ thời gian độc lập như hình 11.

Nhận xét hình 11: Hình 11 có dạng là đặc tính bảo vệ thời gian độc lập theo thiết kế với tiêu chuẩn châu Âu [3].

Để chứng minh khả năng thay đổi thời gian tác động của rơ le, bài báo tiến hành thí nghiệm với thời gian đặt khác, $t_d = 100$ ms, kết quả thí nghiệm ở bảng 4.



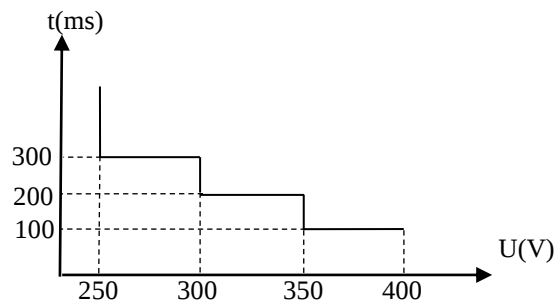
Hình 11. Đặc tính bảo vệ với $t_d = 50$ ms

Bảng 4. Kết quả thí nghiệm với $t_d = 100$ ms

$U_{td}(V)$	250÷299	300÷349	350÷400
$t_d(ms)$	300	200	100

Nhận xét bảng 4: Kết quả thí nghiệm ở bảng 4 cho thấy thời gian tác động đúng bằng thời gian đặt thiết kế ứng với các khoảng điện áp khác nhau như thiết kế ở bảng 2, đồng thời cũng cho thấy tính năng thay đổi được thời gian tác động của rơ le.

Từ kết quả thí nghiệm bảng 4, ta vẽ được đặc tính vào vệ thời gian độc lập như hình 12.



Hình 12. Đặc tính bảo vệ với $t_d = 100$ ms

Nhận xét hình 12: Hình 12 có dạng là đặc tính bảo vệ thời gian độc lập theo thiết kế với tiêu chuẩn châu Âu [3].

4.3. Đánh giá khả năng làm việc của rơ le

Qua các kết quả thí nghiệm thu được, ta thấy rơ le đã làm việc tốt trong phạm vi điện áp làm việc dưới 400 V, hệ số k_{tv} lớn phản ánh độ nhạy của rơ le cao, đặc tính bảo vệ thu được có dạng như yêu cầu thiết kế theo tiêu chuẩn châu Âu [3], thời gian đặt cho phép thay đổi từ 0 đến 60 s.

5. Kết luận

Qua các kết quả thí nghiệm và đánh giá ở trên, bài báo khẳng định đã thiết kế và xây

dùng được rơ le bảo vệ quá áp kỹ thuật số có đặc tính bảo vệ độc lập, thay đổi và hiển thị được giá trị điện áp đặt (đến 400 V) và thời gian đặt (đến 60 s), hiển thị được điện áp lưới, kết nối được với máy tính. Việc thay đổi các tham số và dạng đặc tính bảo vệ cũng có thể được thực hiện thông qua máy tính PC và chương trình trên máy tính. Điều này có ý nghĩa lớn với công tác giảng dạy và nghiên cứu tiếp theo đối với các loại thiết bị bảo vệ kỹ thuật số trong thực tế tại Trường Đại học Kỹ thuật Công nghiệp – Đại học Thái Nguyên.

TÀI LIỆU THAM KHẢO/ REFERENCES

- [1]. H. Q. Nguyen, “Inverse time overcurrent protection numerical relay,” (in Vietnamese), *TNU - Journal of Science and Technology*, vol. 137, no. 07, pp. 112-120, 2015.
- [2]. T. V. Nguyen, *Research, design, manufacture of intelligent digital relays in the electricity system*, (in Vietnamese), Periodical report of implementation of KC03.19, term 1, Nov., 2009.
- [3]. Siprotech team, “Numerical Votage, Frequency and OverFlux Protection Relay SIPROTECT 7RW600 v3.0,” Siemens AG, 2001. [Online]. Available: <http://www.electricalmanuals.net/index.php/manuals1/manuals/protective-relays/siemens/7rw600>. [Accessed April 12, 2020].
- [4]. Atmel team, “Atmega 16A Datasheet,” Microchip Technology Inc, 2014. [Online], Available: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-8154-8-bit-AVR-ATmega16A_Datasheet.pdf. [Accessed April 12, 2020].
- [5]. The HP Info Tech team, “CodeVisionAvr User Manual,” Pavel HaiDuc and HP InfoTech S.R.L., 2016. [Online]. Available: <http://www.arctan.ca/files/cvavrman326.pdf>. [Accessed April 12, 2020].