

## Biến tần 8051 V/f của động cơ cảm ứng

Trở lại với cái nhìn tổng quan

*Biến tần vô hướng (V/f) đơn giản của động cơ cảm ứng, sử dụng 8051 trong kỹ thuật PWM*



Ezequiel

Trở lại với cái nhìn tổng quan <sup>4</sup> Tập <sup>0</sup> Thành phần <sup>5</sup> Nhật Ký <sup>0</sup> hướng dẫn <sup>0</sup> thảo luận <sup>0</sup>

« Quay lại chi tiết dự án

### SẮP XẾP THEO:

Mới nhất



### Chiến lược V/f

01/05/2017 lúc 16:30 • 0 bình luận

Một số bài viết hay giải thích về điều khiển tốc độ động cơ cảm ứng thông qua chiến lược V / f có sẵn trong:

<http://www.electrical4u.com/variable-frequency-drive/>

<http://controlltrends.org/wp-content/uploads/2010/10/VFFundamentals.pdf>

Trong dự án này, tôi đã cố gắng ngăn vi điều khiển thực hiện các phép tính cho các biến điện áp và tần số. Nó sử dụng các giá trị được tính toán trước để tải các thanh ghi được sử dụng trong các quy trình, theo tần số đầu ra mong muốn. Tôi sẽ chỉ ra cách thu được những giá trị này (xem bảng tính với các tính toán trên tab "vf").

Đầu tiên là tần số. Để thay đổi tần số đầu ra, chỉ cần điều chỉnh tốc độ làm mới sóng hình sin, tức là thay đổi giá trị tải lại của bộ đếm ngắt timer 0 (các biến T\_SIN\_H và T\_SIN\_L, đại diện cho

từ 16 bit). Để biết thêm chi tiết, hãy tham khảo bảng tính và sách hướng dẫn 8051, hãy nhớ rằng bộ đếm thời gian đang ở chế độ 1. Các giá trị được tính toán được hiển thị bên dưới.

```
;GIÁ TRỊ ĐƯỢC TẢI LẠI TRONG ĐỒNG HỒ ĐỊA HÌNH Sine , THEO TẦN SỐ
T_60 THIẾT BỊ 65152 D
T_50 THIẾT BỊ 65075 D
T_40 THIẾT BỊ 64960 D
T_30 THIẾT BỊ 64768 D
T_20 THIẾT BỊ 64384 D
T_10 THIẾT BỊ 63232 D
```

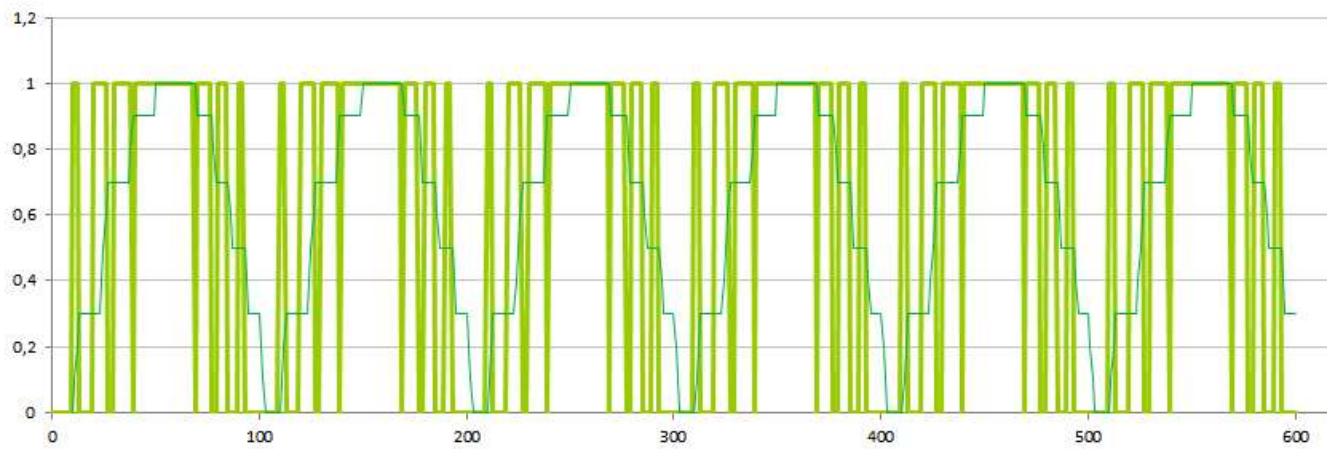
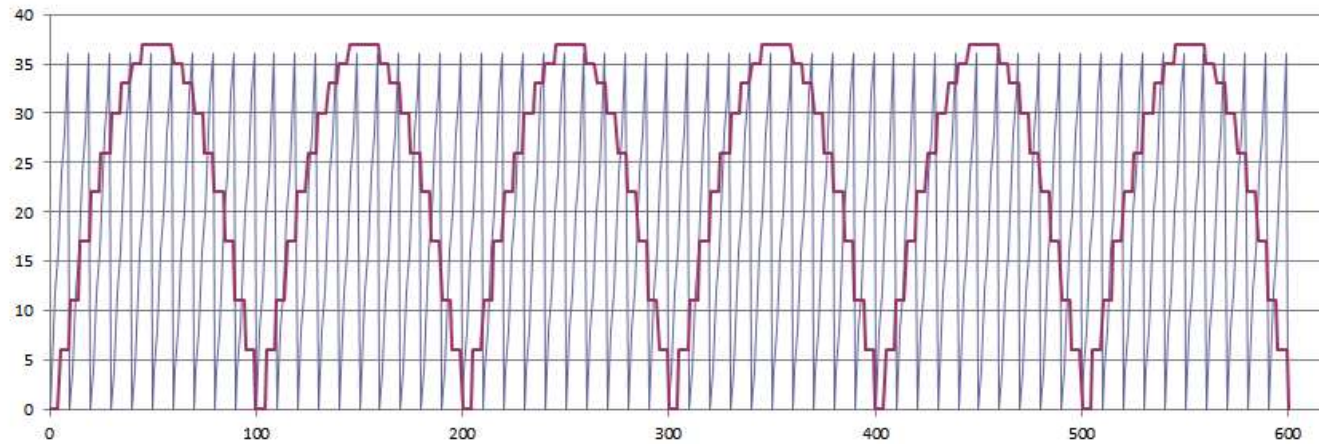
Để thay đổi điện áp, tùy chọn ít tính toán hơn trong vi điều khiển là thay đổi biên độ của sóng răng cưa. Điều này được thực hiện bằng cách thay đổi bước tăng ngắt hẹn giờ 1 (biến STEP\_SAW). Hành vi biến đổi là nghịch đảo, nghĩa là bằng cách tăng biên độ sóng, nó sẽ làm giảm điện áp đầu ra.

Trong trường hợp này, những thách thức là: tìm số gia số nguyên tạo ra độ phân giải tốt trong biến thể lực căng; Và đảm bảo rằng biên độ của cưa nhỏ hơn hoặc bằng 255 (để khớp với từ 8 bit) ở tất cả các tần số đầu ra. Bằng phương pháp thử và sai, tất cả các tham số phải được điều chỉnh để tuân theo các điều kiện này (bao gồm cả biên độ hình sin, 37, nhớ không?). Các giá trị tính toán được hiển thị dưới đây.

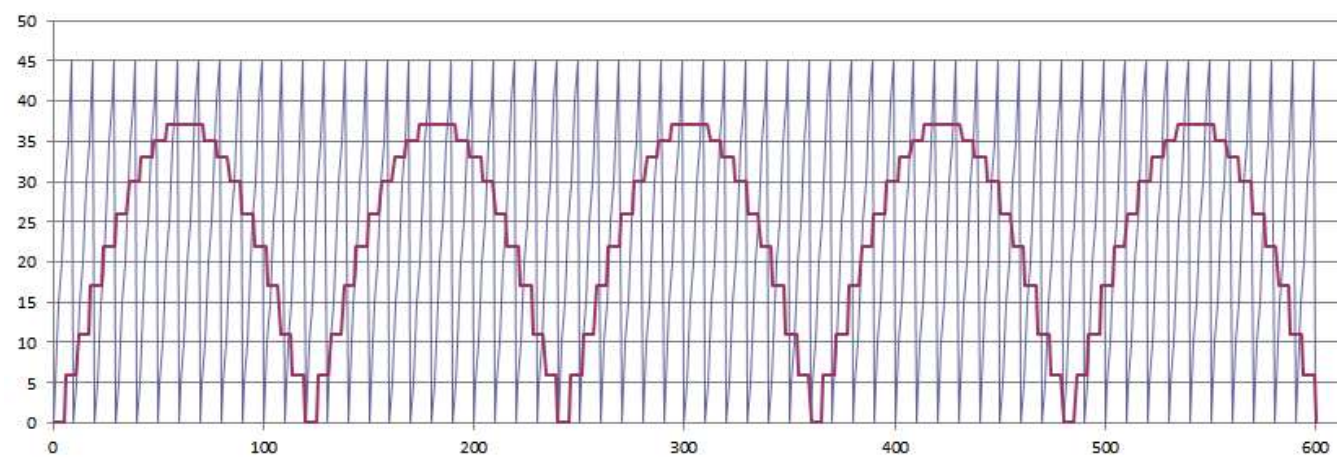
```
;GIÁ TRỊ BƯỚC KHUẾCH ĐẠI CỦA SÓNG RĂNG CƯA THEO TẦN SỐ
BƯỚC_60 THIẾT BỊ 4 D
BƯỚC_50 THIẾT BỊ 5 D
BƯỚC_40 THIẾT BỊ 6 D
BƯỚC_30 THIẾT BỊ 8 D
BƯỚC_20 THIẾT BỊ 11 D
BƯỚC_10 THIẾT BỊ 23 D
```

Bạn có thể xem mô phỏng toàn bộ đầu ra PWM trong tab "pwm" của trang tính. Các số liệu sau đây cho thấy kết quả (ở 10Hz là kém, nhưng có thể chấp nhận được)

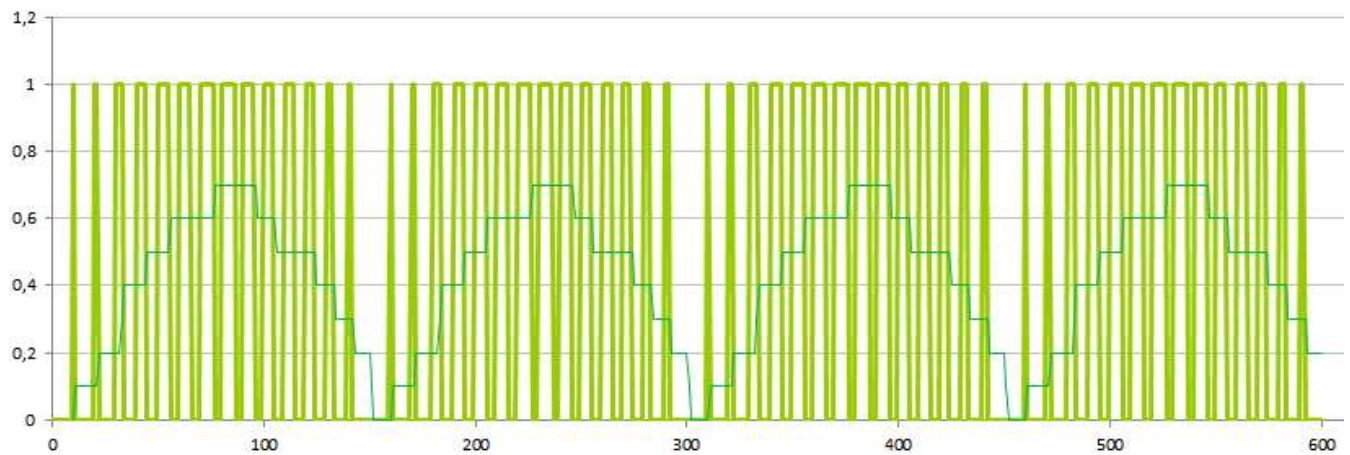
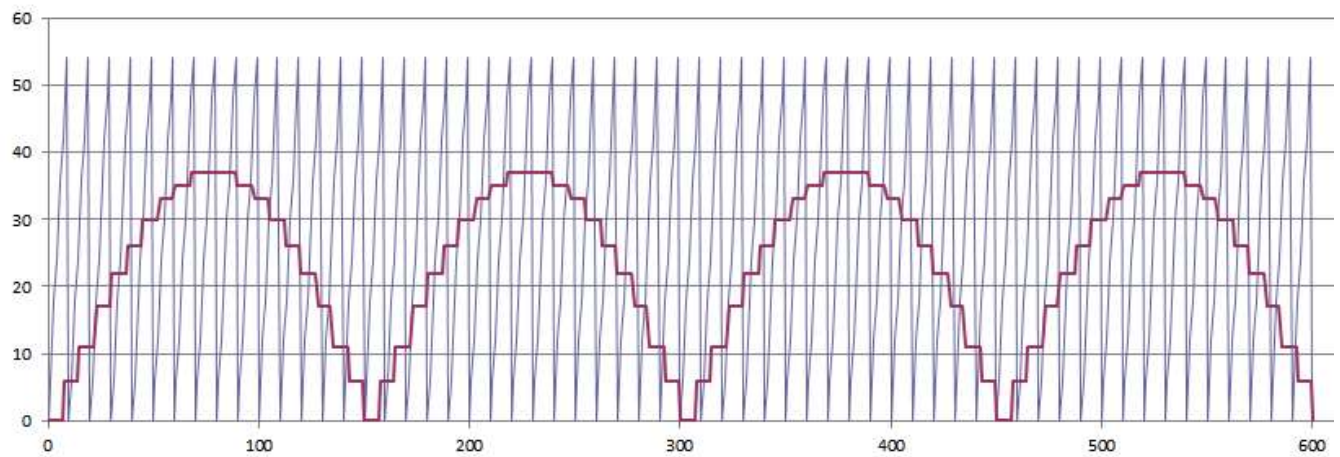
# Freq = 60 Hz



Freq = 50 Hz

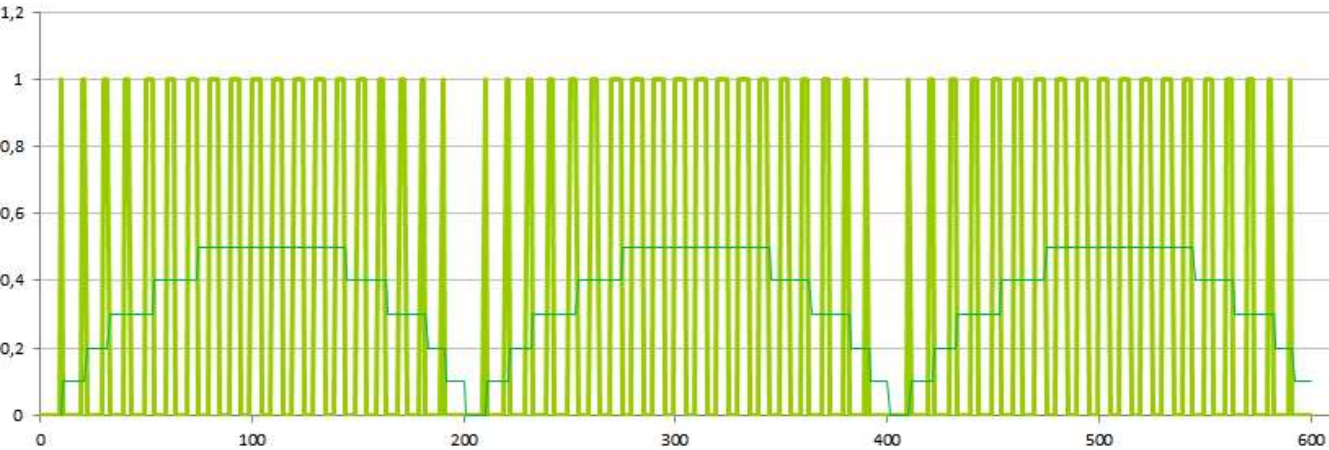
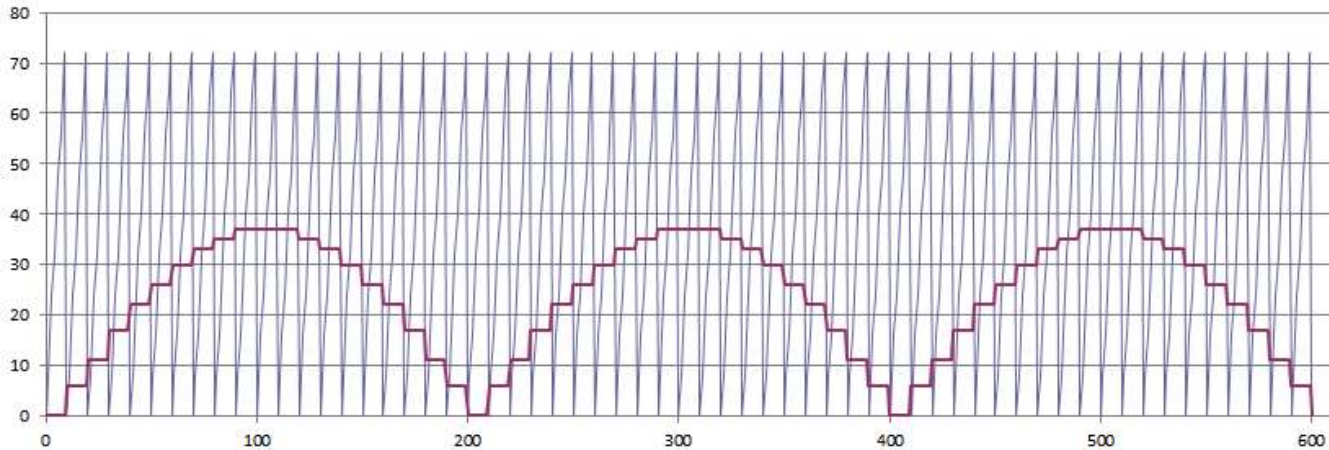


# Freq = 40 Hz

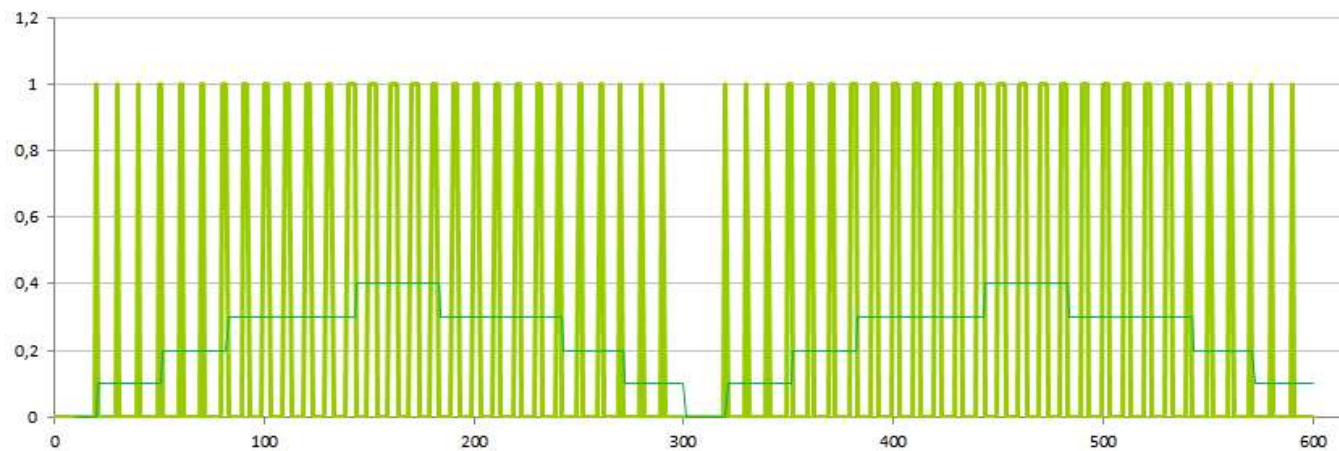
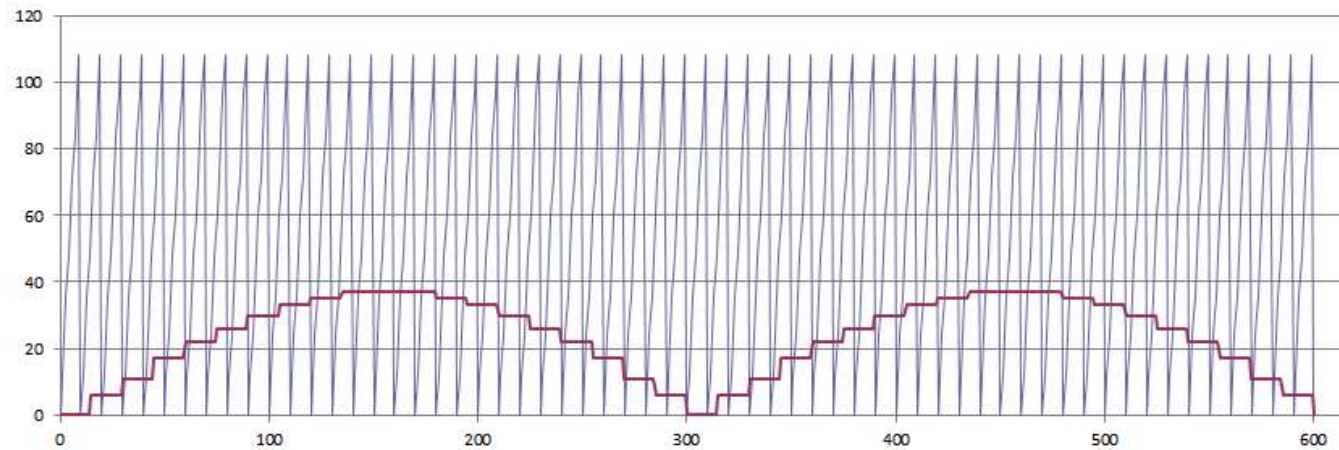




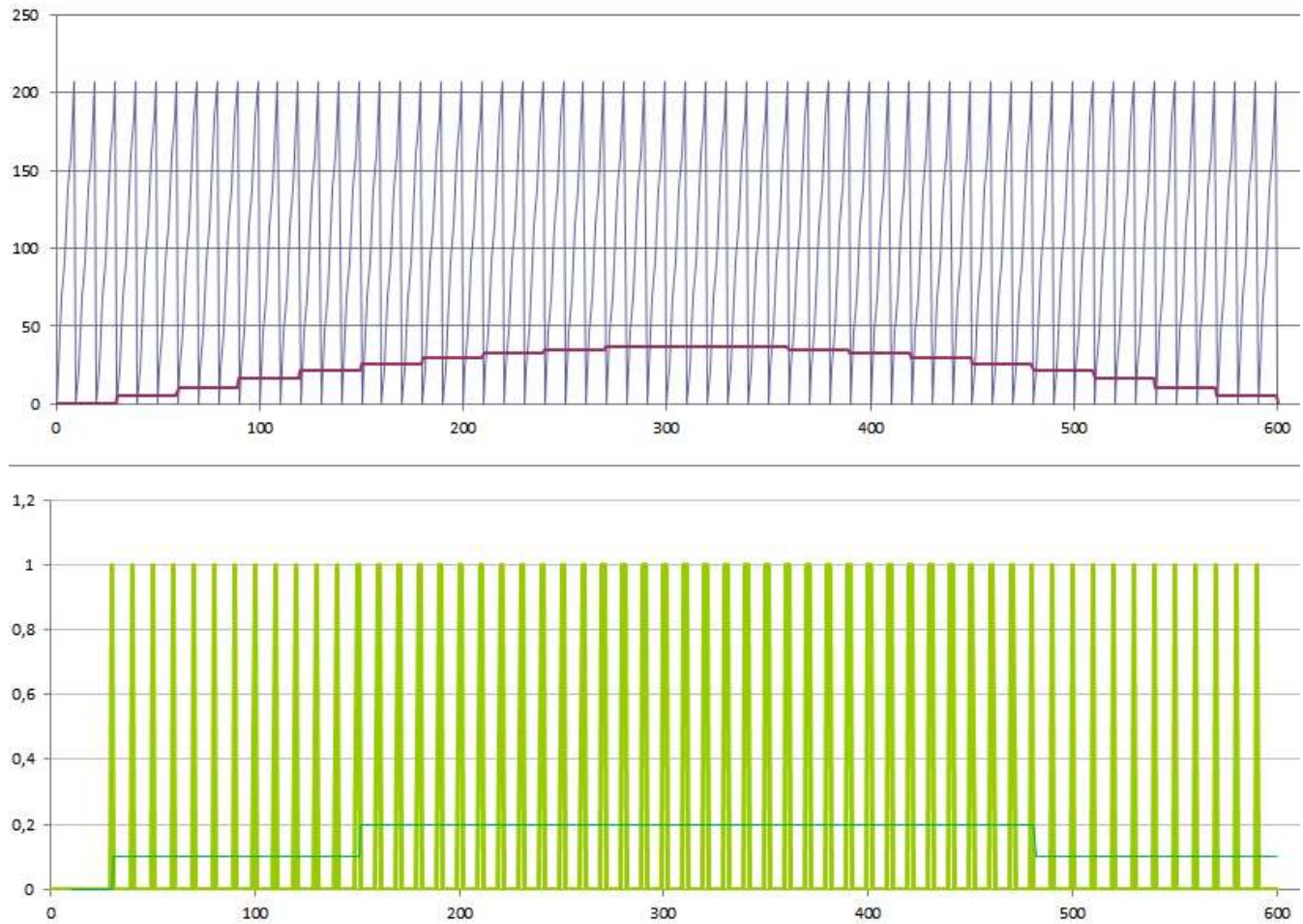
Freq = 30 Hz



Freq = 20 Hz



# Freq = 10 Hz



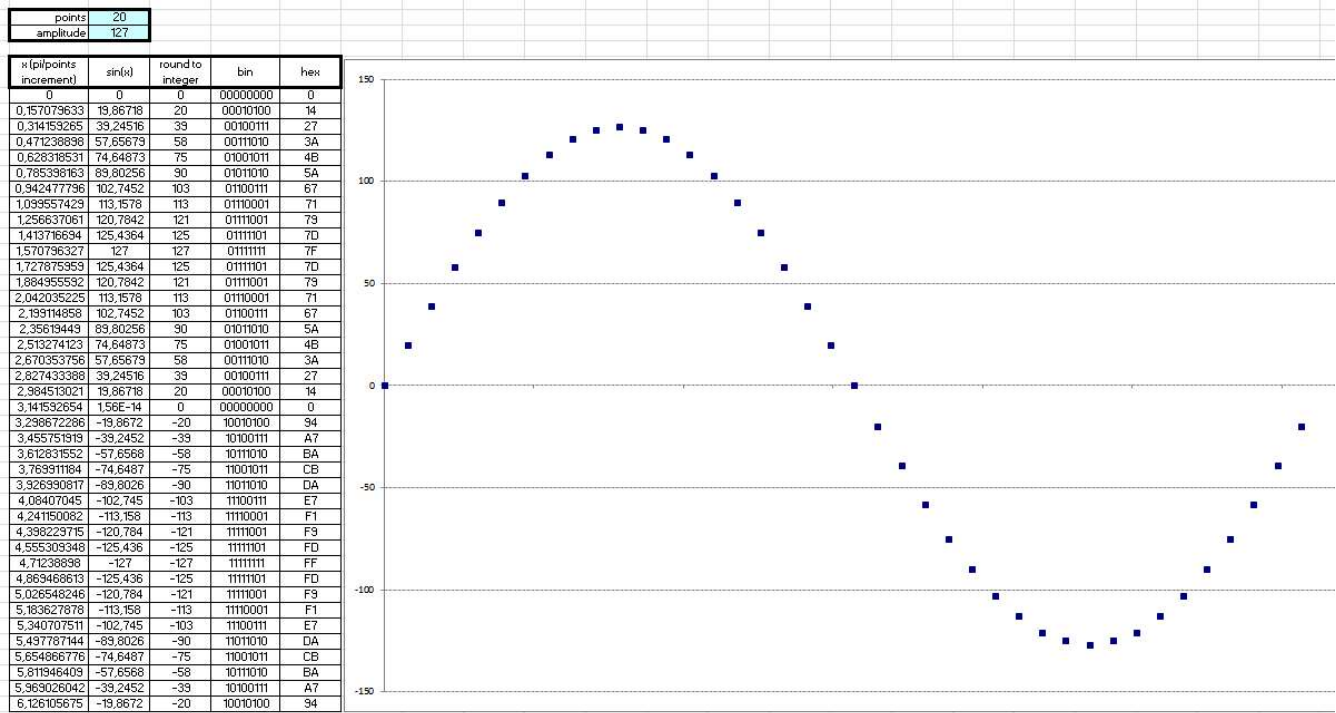
## Bảng sin

01/04/2017 lúc 10:54 • 0 bình luận

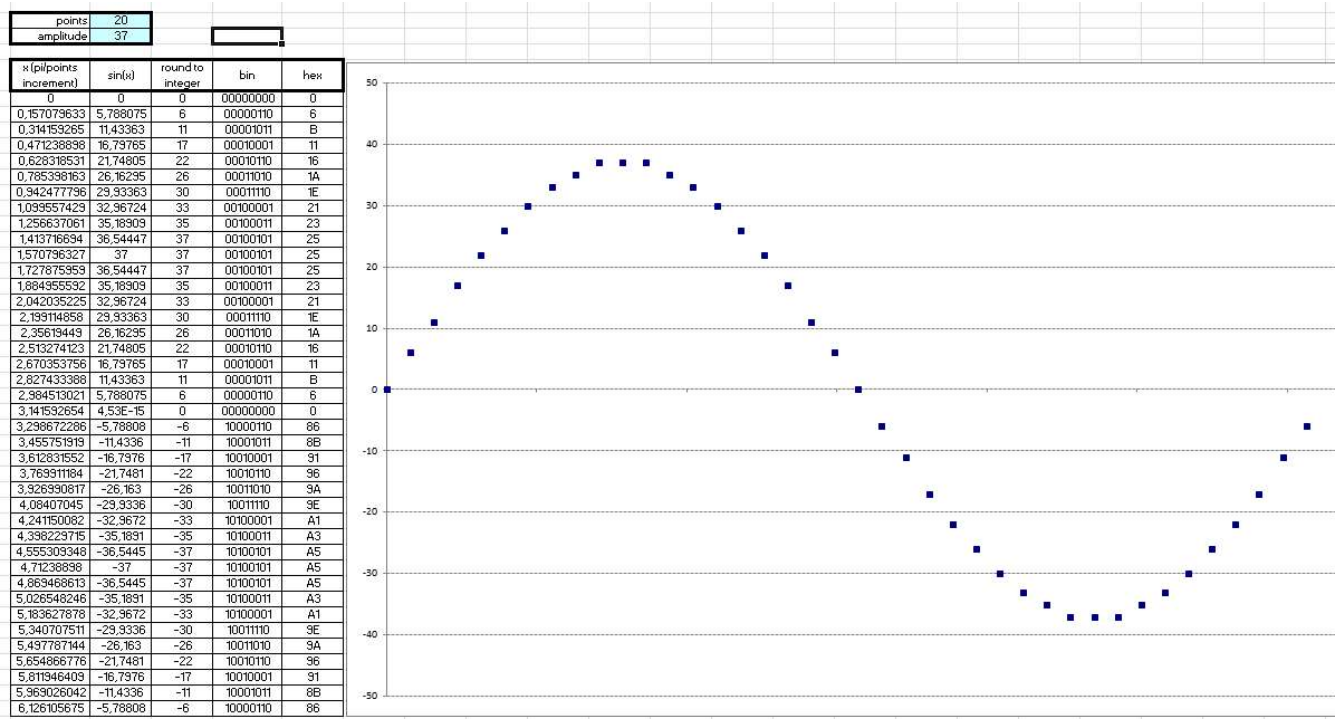
Để tạo bảng sóng hình sin được sử dụng trong thuật toán được hiển thị trong bài đăng trước, một bảng tính đã được sử dụng để giúp tính toán các giá trị (bảng tính này nằm trong vùng tệp, xem tab "sine" trong calc.xls).

Mỗi nửa chu kỳ của sóng được lấy mẫu tại 20 điểm và biên độ được giới hạn sao cho các giá trị khớp với các từ 8 bit. Bit cao hơn trong từ là cực tính và 7 còn lại là chính giá trị, xác định rằng biên độ là 127 cực đại. Điều này dẫn đến một dạng sóng tốt ngay cả với các giá trị số nguyên. Xem hình sau:





Để thực hiện chiến lược điều khiển V/f (chúng ta sẽ xem sau), biên độ sóng phải được đặt thành 37 cực đại (thử và sai). Nhưng điều này không làm giảm đáng kể dạng sóng. Xem hình sau:



Sau đó, bảng cuối cùng trong mã:

```

;===== BẢNG SÓNG HÌNH sin =====
;CHU KỲ ĐẦY ĐỦ CỦA SÓNG HÌNH Sine, ĐƯỢC LẤY MẪU TẠI 40 ĐIỂM
;BIT Ý NGHĨA NHẤT CHỈ ĐỊNH CỰC TIÊU CỰC

```

SÓNG HÌNH SIN:

DB 0,6,11,17,22,26,30,33,35,37,37,37,35,33,30,26,22,17,11,6,0,134,1

;=====

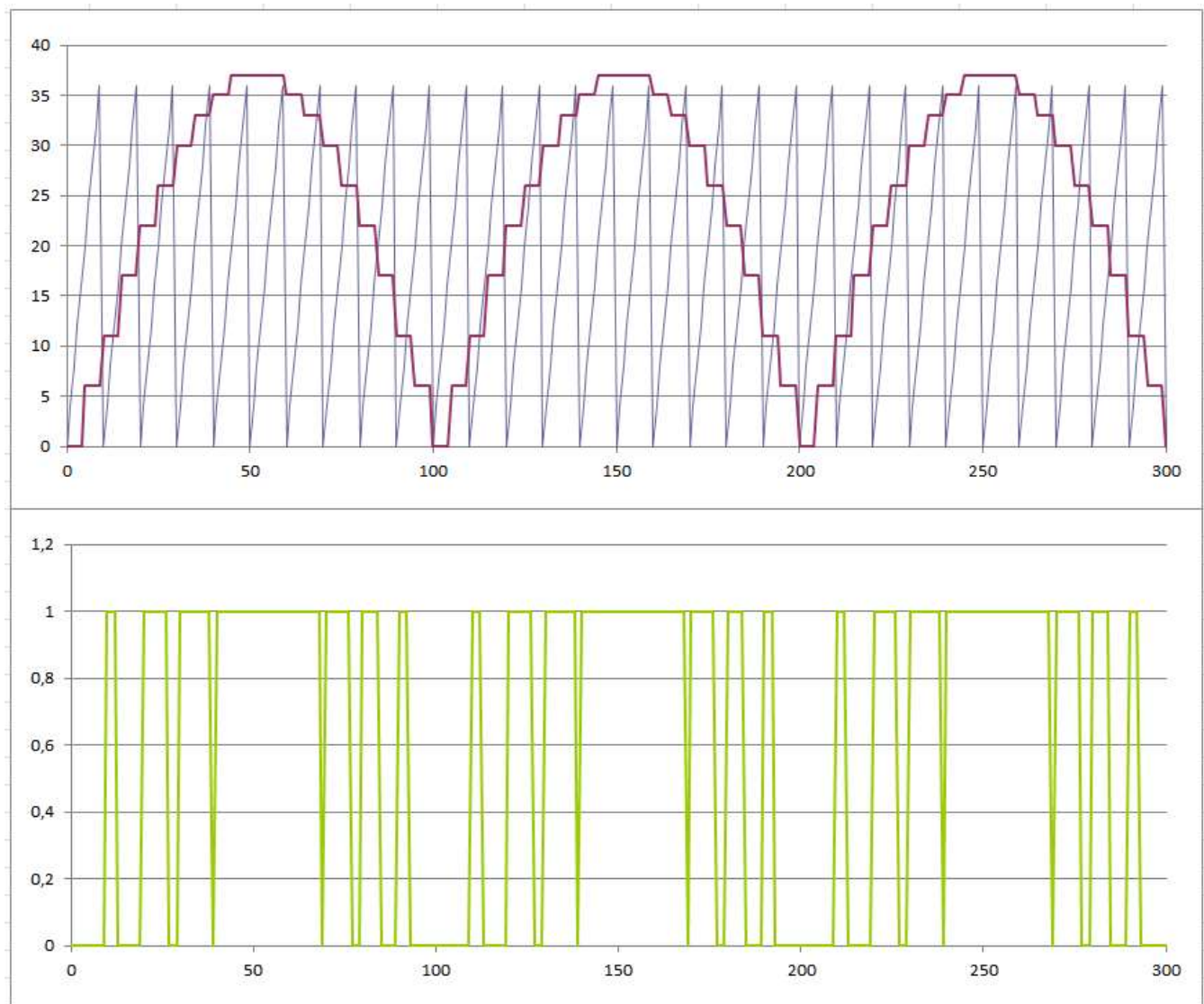


PWM

02/01/2017 lúc 16:07 • 0 bình luận

Một bài viết hay giải thích chi tiết về chiến lược PWM có sẵn trên Wikipedia tại [https://en.wikipedia.org/wiki/Pulse-width\\_modulation](https://en.wikipedia.org/wiki/Pulse-width_modulation) .

Hiện tại, chiến lược được áp dụng trong dự án này là so sánh tín hiệu đầu ra mong muốn (sóng hình sin) với tín hiệu tham chiếu, sóng răng cưa. Khi cái sau ở dưới tín hiệu máy phát, PWM ở trạng thái cao (1). Mặt khác, nó ở trạng thái thấp (0). Xem hình sau.



Không cần thiết phải làm điều này cho sóng hình sin hoàn chỉnh, chỉ nửa chu kỳ vì tính đối xứng của nó. Để đạt được nửa chu kỳ âm, một bit sẽ chuyển đổi trình tự chuyển mạch của cầu H.

8051 không có bất kỳ cơ sở nào để tạo đầu ra PWM, vì vậy chiến lược được mô tả phải được thực hiện hoàn toàn trong mã. Để tạo ra sóng hình sin và sóng răng cưa, hai bộ định thời 8051 và các ngắt của chúng sẽ được sử dụng.

Sóng hình sin sử dụng kỹ thuật bảng tra cứu. Hãy xem mã:

T0\_INT:

```
CLR TR0 ; DỪNG HẸN GIỜ
MOV DPTR, #SINE_WAVE
MOV A, COUNT_SIN
MOVC A, @A+DPTR ;XEM GIÁ TRỊ CỦA CHỈ SỐ COUNT_SIN TRONG BẢNG SINE_W
MOV SINE, A ; AND LƯU TRỮ TRONG BIẾN Sine

MOV C, SINE.7 ;BIT GIÁ TRỊ CAO LÀ PHÂN CỰC
```

```
MOV P1.1,C ; GỬI ĐẾN ĐẦU RA RIÊNG
CLR SINE.7; VÀ XÓA NÓ
```

```
INC COUNT_SIN ; ĐI ĐẾN GIÁ TRỊ TIẾP THEO
MOV A, COUNT_SIN
```

```
CJNE A, #40, QUIT_T0 ;KIỂM TRA NẾU BẢNG CÓ Ở CUỐI
MOV COUNT_SIN, #0 ; SAU ĐÓ, KHỞI ĐỘNG LẠI BẢNG ĐẾM
```

QUIT\_T0:

```
MOV TL0, T_SIN_L ; TẢI LẠI BỘ ĐẾM HẸN GIỜ
MOV TH0, T_SIN_H
ĐẶT TR0 ; KHỞI ĐỘNG LẠI BỘ HẸN GIỜ
RETI
```

Mỗi lần ngắt được gọi, giá trị hiện tại của wave được cập nhật bằng cách truy vấn một bảng được tính toán trước. Coi khoảng thời gian bằng nhau thì sóng sinh ra do việc sắp xếp các giá trị.

Đối với sóng răng cưa, quy trình đơn giản hơn: chỉ cần tăng giá trị hiện tại với bước không đổi cho đến khi đạt đến giá trị của chu kỳ (10 bước), bắt đầu lại trình tự. Khi tần số cao hơn, có thể sử dụng bộ định thời 8051 ở chế độ tải lại tự động 8 bit, đơn giản hóa mã. Xem mã:

T1\_INT:

```
;THỂ HỆ SÓNG RĂNG CỬA
MOV A, RĂNG CỬA
THÊM A, STEP_SAW ;TĂNG CỬA VỚI BƯỚC HIỆN TẠI
MOV RĂNG CỬA, A

DJNZ COUNT_SAW, QUIT_T1 ;KIỂM TRA XONG
MOV SAWTOOTH, #0 ; SAU ĐÓ, KHỞI ĐỘNG LẠI SÓNG
MOV COUNT_SAW, #10D ;VÀ ĐẶT LẠI BỘ ĐẾM CỦA NÓ
```

QUIT\_T1:

```
RETI
```

Cuối cùng, đầu ra PWM so sánh hai tín hiệu và kết quả sẽ là bit nhớ:

```
MOV A,RĂNG CỬA ; SO SÁNH SÓNG CỬA
SUBB A, HÌNH SIN ; SO VỚI SÓNG HÌNH SIN
MOV P1.0,C ;<==== ĐẦU RA PMW
```

Đặt mọi thứ lại với nhau (PWM chia sẻ bộ đếm thời gian răng cưa):

```
;===== TẠO SÓNG HÌNH SIN =====
```

```
;SÓNG HÌNH SIN ĐƯỢC SỬ DỤNG LÀM THAM KHẢO SO SÁNH TRONG ĐẦU RA PWM.
```

```
;GIÁ TRỊ SIN ĐƯỢC CẬP NHẬT ĐỊNH KỲ BỞI CUỘC GỌI GIÁN ĐOẠN T0,
```

```
;KỸ THUẬT BẢNG TÌM KIẾM QUA. THỜI GIAN GIÁN ĐOẠN THAY ĐỔI,
```

```
;PHỤC VÀO TẦN SỐ ĐẦU RA ĐÃ CHỌN.
```

```
T0_INT:
```

```
CLR TR0 ; DỪNG HẸN GIỜ
```

```
MOV DPTR, #SINE_WAVE
```

```
MOV A, COUNT_SIN
```

```
MOVC A, @A+DPTR ;XEM GIÁ TRỊ CỦA CHỈ SỐ COUNT_SIN TRONG BẢNG SINE_W
```

```
MOV SINE, A ; AND LƯU TRỮ TRONG BIẾN Sine
```

```
MOV C, SINE.7 ;BIT GIÁ TRỊ CAO LÀ PHÂN CỰC
```

```
MOV P1.1,C ; GỬI ĐẾN ĐẦU RA RIÊNG
```

```
CLR SINE.7; VÀ XÓA NÓ
```

```
INC COUNT_SIN ; ĐI ĐẾN GIÁ TRỊ TIẾP THEO
```

```
MOV A, COUNT_SIN
```

```
CJNE A, #40, QUIT_T0 ;KIỂM TRA NẾU BẢNG CÓ Ở CUỐI
```

```
MOV COUNT_SIN, #0 ; SAU ĐÓ, KHỞI ĐỘNG LẠI BẢNG ĐẾM
```

```
QUIT_T0:
```

```
MOV TL0, T_SIN_L ; TẢI LẠI BỘ ĐẾM HẸN GIỜ
```

```
MOV TH0, T_SIN_H
```

```
ĐẶT TR0 ; KHỞI ĐỘNG LẠI BỘ HẸN GIỜ
```

```
RETI
```

```
;===== TẠO SÓNG SAW VÀ ĐẦU RA PWM =====
```

```
;TẦN SỐ CỦA SÓNG CƯA ĐƯỢC CỐ ĐỊNH Ở 1200Hz,
```

```
;VÀ TÍN HIỆU NÀY ĐƯỢC TẠO BẰNG 10 BƯỚC.
```

```
T1_INT:
```

```
; ĐẦU TIÊN, PWM!
```

```
MOV A,RĂNG CƯA ; SO SÁNH SÓNG CƯA
```

```
SUBB A, HÌNH SIN ; SO VỚI SÓNG HÌNH SIN
```

```
MOV P1.0,C ;<===== ĐẦU RA PMW
```

```
;TIẾP TỤC TẠO SÓNG RĂNG CƯA
```

```
MOV A, RĂNG CƯA
```



```
THÊM A, STEP_SAW ;TĂNG CỬA VỚI BƯỚC HIỆN TẠI
MOV RĂNG_CỬA, A
```

```
DJNZ COUNT_SAW, QUIT_T1 ;KIỂM TRA XONG
MOV SAWTOOTH, #0 ; SAU ĐÓ, KHỞI ĐỘNG LẠI SÓNG
MOV COUNT_SAW, #10D ;VÀ ĐẶT LẠI BỘ ĐẾM CỦA NÓ
```

```
QUIT_T1:
```

```
DEC R0 ;R0 CHO QUY TRÌNH GỠ NÚT
RETI
```



## Mã và sơ đồ

31/12/2016 lúc 13:30 • 0 bình luận

Nguồn đầy đủ có sẵn trong khu vực tệp và cả trong GitHub và được nhận xét tốt (bản dịch có thể không tốt lắm). Mã có đầy đủ chức năng, nhưng có thể thay đổi trong tương lai.

Sơ đồ mạch cũng có sẵn. Mạch điều khiển dựa trên AT89C2051, đối với tôi khá dễ sử dụng và lắp ráp. Nhưng mã này hoàn toàn tương thích với kiến trúc 8051 và có thể được sử dụng trên các nền tảng và nhà sản xuất khác.

Mạch điều khiển dựa trên cầu H với MOSFET. Tôi chưa xây dựng nó, vì vậy tôi nghĩ rằng các thành phần có thể thay đổi một chút.

## bắt đầu

31/12/2016 lúc 13:10 • 0 bình luận

CHÀO,

Tôi sẽ cố gắng giải thích chi tiết về dự án trong các nhật ký tiếp theo này.

Đầu tiên, đối với dự án này, bạn không cần các công cụ phức tạp, chỉ cần một trình soạn thảo văn bản và trình biên dịch mã 8051 (có một số công cụ có sẵn trên Internet). Tôi đã chọn sử dụng một IDE hoàn chỉnh có các công cụ này và cũng là một trình giả lập hữu ích. Đó là MCU 8051 IDE, miễn phí, có sẵn tại <http://mcu8051ide.sourceforge.net>

[Về chúng tôi](#)

[Liên hệ Hackaday.io](#)

[Cung cấp thông tin phản hồi](#)

[Điều khoản sử dụng](#)

[Chính sách bảo mật](#)

[API Hackaday](#)

© 2023 Hackaday

