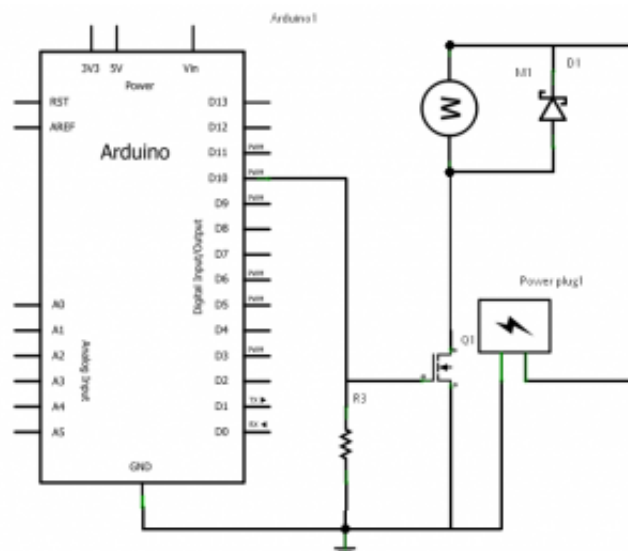


Arduino + motor DC: circuito PWM

Jorge García Tíscar | July 21, 2011

Un Arduino, como cualquier microcontrolador, es incapaz de proporcionar la potencia (el amperaje) que necesita un motor para funcionar, por lo que debemos emplear una fuente externa y una fase de potencia: la más simple posible se basa en un transistor MOSFET de nivel lógico.

Su funcionamiento es el siguiente: una señal lógica alta (5V) abre la puerta y permite el paso de la corriente, una señal de nivel bajo (0V) cierra la puerta e impide el paso de corriente. Esta señal que se debe transmitir tiene la ventaja de emplear un amperaje mínimo, del orden de mili o nanoamperios, asumible por el Arduino.

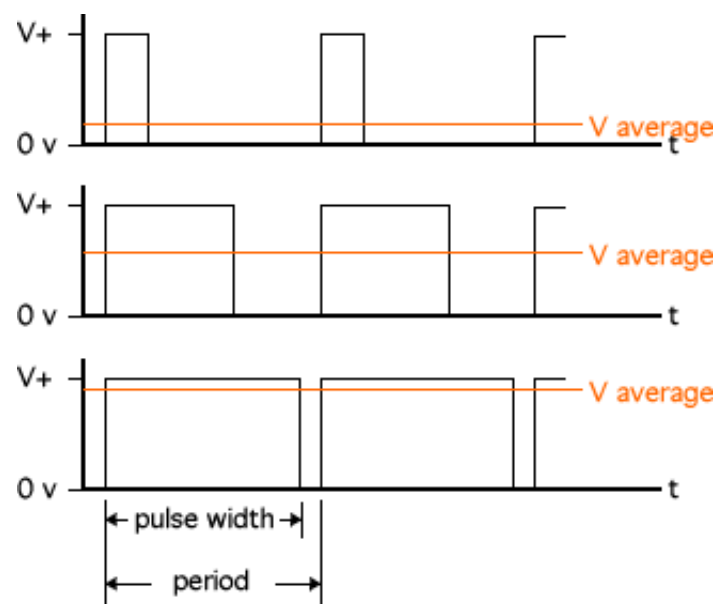


A la derecha un esquema del montaje donde se puede ver el Arduino, el

MOSFET y el motor. También se incluye una resistencia en la puerta y un diodo de protección.

El tren de pulsos PWM

El lector avisado, no obstante, señalará que esto sólo permite al Arduino mantener el motor apagado o a máxima potencia (señal binaria), mientras que nosotros pretendíamos obtener una señal analógica, esto es, variable a voluntad entre 0V y 5V; y es más, tendrá toda la razón! Pero, afortunadamente, hay una manera de obtener lo segundo a partir de lo primero: el tren de pulsos PWN (*pulse width modulation*):

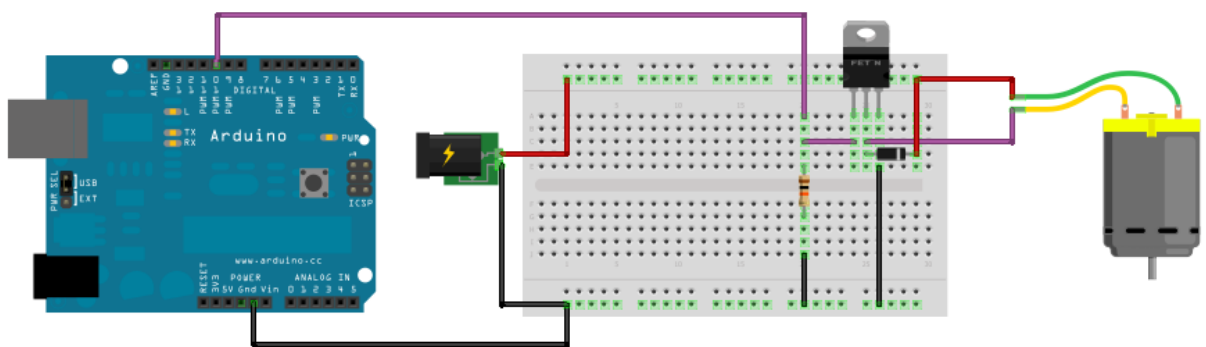


Se emite una señal de pulsos como la representada arriba, donde sea posible variar la relación D entre tiempo de señal alta y tiempo de señal baja ($D = \frac{\text{pulse width}}{\text{period}} \in [0, 1]$ en la figura). Los dispositivos analógicos que se encuentren esta señal la percibirán como una señal constante, cuyo voltaje será el valor medio de la función de pulsos:

$$V_{\text{medio}} = \frac{1}{T} \int_0^T f(t) dt = D \cdot V^+ + (1 - D) \cdot 0 = D \cdot V^+$$

Montaje

Lo importante de esta solución es que el MOSFET es lo suficientemente rápido como para “transmitir” la señal PWM, es decir, se abre y se cierra según se lo indica el Arduino, con lo que la señal PWM se mantiene intacta (pero ya a mayor voltaje e intensidad) hasta llegar al motor, donde actúa como un valor analógico constante. El montaje sería pues el siguiente:



Esquema del montaje real

Nótese el diodo de protección contra corrientes de retorno del motor, importante! A modo de ejemplo, los componentes utilizados en nuestro montaje son (se incluyen sus *datasheets* en los enlaces):

- Diodo MOSFET: modelo [BUZ11](#)
- Diodo protección: diodo Schottky [SB320](#)
- Resistencia: entre 10K ~ 1M
- Motor: motor DC de 12V

Utilización

Una vez conectado correctamente y alimentado el montaje, ya

podemos controlar el motor mediante el Arduino, en concreto, haciendo uso de la función `analogWrite(pin,value)`, que manda al pin designado una señal PWM con una D controlada por el argumento value, que admite valores entre 0 ($D = 0$) y 255 ($D = 1$). Si por ejemplo este valor entre 0 y 255 lo recibiéramos como un byte a través del puerto serie, podríamos hacer:

```
1  int in = 0;
2  byte pinOut = 10;
3
4  void setup() {
5    // inicializar puerto serie
6    Serial.begin(9600);
7    // preparar output
8    pinMode(pinOut, OUTPUT);
9  }
10
11 void loop() {
12   // leer dato si disponible
13   if(Serial.available()){
14     in = Serial.read();
15     // escribir en el pin
16     analogWrite(pinOut, in);
17   }
18 }
```

Con este código se recibe un valor ente 0 y 255 a través del puerto serie, cuando está disponible, y lo pasa como argumento a la función `analogWrite()` de tal manera que se genera la señal PWM correspondiente en el pin 10, tal y como corresponde en el esquema de montaje propuesto. *Et voilà!*



Thanks for reading! To share this post, use this [permalink](https://wechoosethemoon.es/2011/07/21/arduino-motor-dc-circuito-pwm-mosfet/)

« ARDUINO + MATLAB/SIM

ARDUINO + LABVIEW: INSTRUMENTOS VIRTUALES »

Comments

Comments Community  Privacy Policy  1 Login ▾

 Recommend 5

Sort by Best ▾

Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS 

Name

dahmane hamid • 2 years ago

my friend is very good works

^ | ▾ • Reply • Share ›



Alejandro Jaumandreu • 4 years ago

Tengo un problema con este tipo de circuito y es que el motor pierde potencia y no consigue girar una serie de poleas al conectar el circuito de potencia al arduino. No se si puede ser debido al transistor o es habitual esta pérdida de fuerza (utilizo un BD139). Qué me recomendarías?

^ | ▾ • Reply • Share ›

fran • 5 years ago

puedo controlar un motor de 6 Amperes con este circuito. usando el mosfet y el diodo. y que resistencia utilizaria.

^ | ▾ • Reply • Share ›

Rodolfo Escobar • 7 years ago

Tengo el mismo problema con el programa. En general he estado teniendo problemas con la función `analogWrite(pin,value)` cuando `value` es una lectura serial. El pin ignora cualquier valor que envío desde el monitor seria o desde Matlab con `fwrite(s,'n','uint8')`. ¿Hay algo que no estoy tomando en cuenta?

^ | v • Reply • Share ›

El Rey Del Mundo • 8 years ago

Parece que no funciona.... le doy cualquier valor y solo prende, no cambia de intensidad

^ | v • Reply • Share ›

Jose • 3 years ago

Hola.

Como puedo controlar una salida PWM en funcion de una entrada analogica?

Me explico: quiero fijar una analogica en un valor determinado y en funcion de eso la salida debera variar

CC-BY-SA JorgeGT 2018 | Powered by procrastination™

