

# Lusorobótica Motor Board

## O porquê de utilizar um Multiplexer

O funcionamento básico de controlo de um motor através de uma ponte H, neste caso usando o L293D, é o esquematizado na seguinte figura:

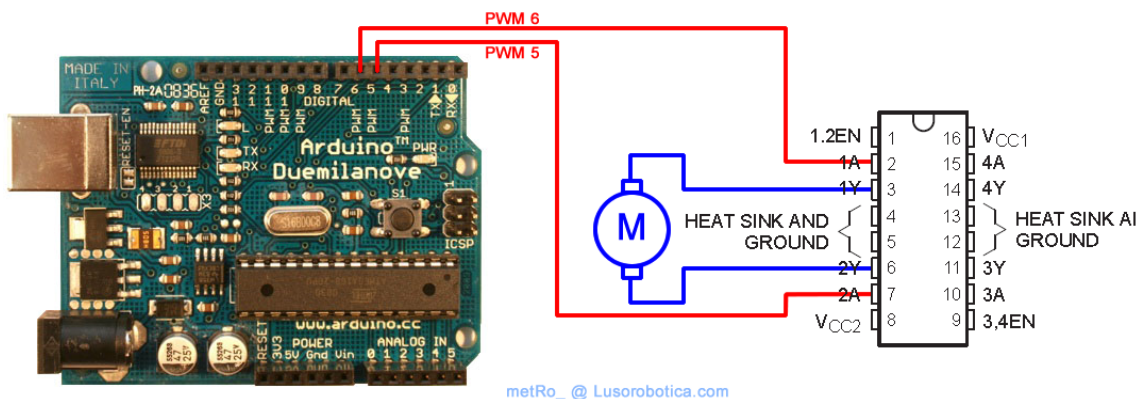


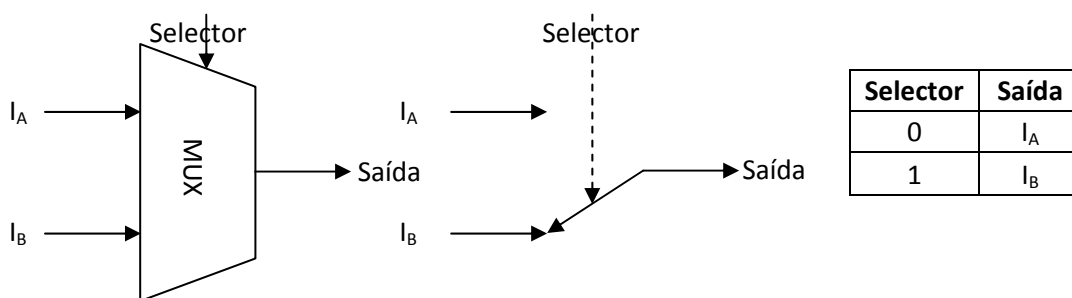
Figura 1 – Esquema básico do controlo de um motor.

Como se pode observar o controlo do motor utiliza duas saídas PWM's. Estas PWM's são utilizadas para definir a velocidade e o sentido de rotação, assim quando por exemplo a PWM 6 está a LOW(0) e a PWM 5 tem um valor maior que 0, isto é, quando escrevemos na saída um valor superior a 0 como por exemplo `analogWrite(velMotor, 255);` para a velocidade máxima, o motor vai rodar à velocidade máxima numa direcção. Se colocarmos a PWM 5 a LOW(0) e escrevermos na PWM 6 o valor 255, vamos passar a ter o motor a rodar à velocidade máxima mas no sentido contrário. Resumindo:

- PWM 6 : LOW(0)
  - PWM 5 : `analogWrite(velMotor, 255);`
  - PWM 6 : `analogWrite(velMotor, 255);`
  - PWM 5 : LOW(0)
  - PWM 6 : LOW(0)
  - PWM 5 : LOW(0)
  - PWM 6 : `analogWrite(velMotor, 255);`
  - PWM 5 : `analogWrite(velMotor, 255);`
- O motor roda à velocidade máxima, numa direcção.
- O motor roda à velocidade máxima, na direcção contrária à anterior.
- O motor para em ambos os casos.

Após se perceber o funcionamento da ponte percebemos onde são utilizadas as duas PWM's e se repararmos bem na verdade há sempre um PWM que está a LOW(0) e para isto uma saída digital chegava. Visto as PWM's serem mais importantes e mais utilizadas que as saídas digitais, decidiu-se encontrar uma solução para reduzir o número de PWM's utilizados e assim chegou-se à conclusão que seria possível utilizar somente uma saída PWM e uma digital ao invés de duas PWM's.

A solução adoptada foi a utilização de um Multiplexer, um multiplexer permite colocar na saída a entrada desejada entre as várias entradas. A figura seguinte demonstra o funcionamento básico de um mux(a partir deste ponto irei chamar a multiplexer simplesmente mux).



Na tabela de verdade podemos ver que a saída depende do estado da entrada Selector. Usando esta função e como entradas uma PWM e a outra ligada à massa (o mesmo que um pino a LOW) poderemos colocar na mesma saída ambos os valores dependendo da entrada Selector, repetindo este processo e invertendo as entradas obtemos a nossa solução que está representada na seguinte figura.

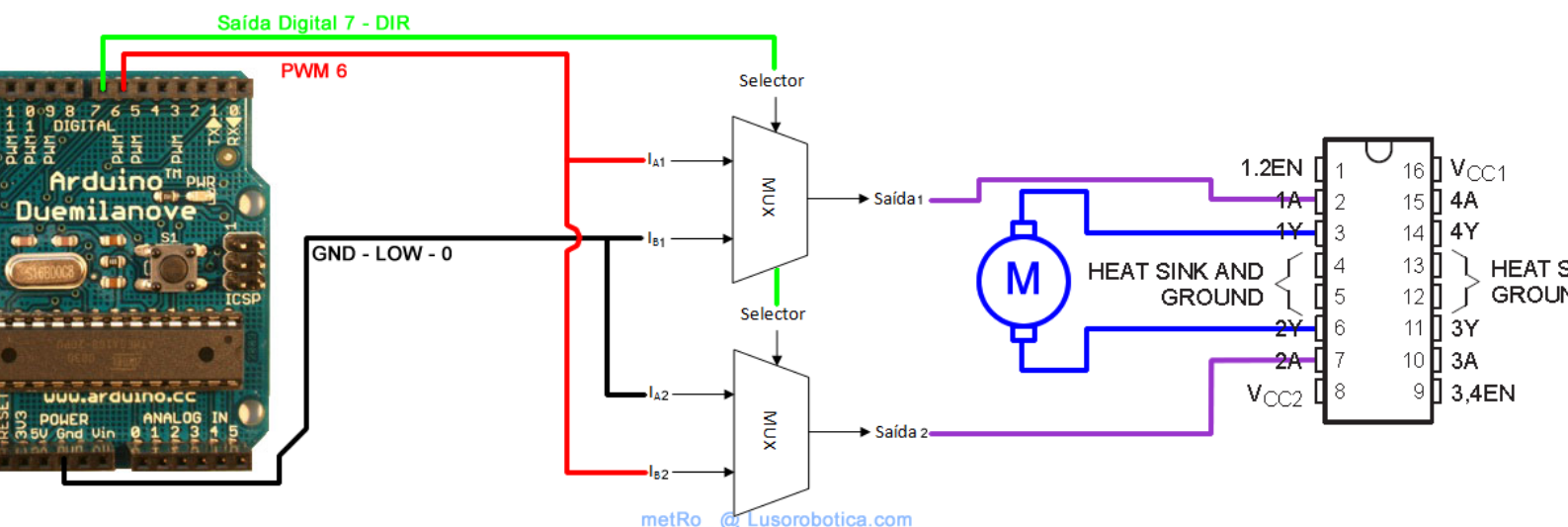


Figura 2 – Esquema das ligações do Arduino ao mux e ao motor.

Agora vamos analisar cada um dos estados. Vamos considerar que escrevemos na PWM 6 o seguinte valor: `analogWrite(velMotor, 128)`; (128 corresponde sensivelmente a metade da velocidade máxima do motor) e vamos alternar o valor da ligação DIR (saída digital 7) entre HIGH (1) e LOW (0).

DIR	Saída 1	Saída 2
HIGH	PWM 6 ( IA1 )	GND – LOW - 0 ( IA2 )
LOW	GND – LOW - 0 ( IB1 )	PWM 6 ( IB2 )

Alterando só a DIR podemos ver que obtemos na entrada da ponte uma PWM e um LOW ou um LOW e uma PWM.

Espera que a criação deste tutorial tenha sido útil.

Vários mux's disponíveis: 74LS151; 74LS153; 74LS157; 74LS257 (Usado na Lusorobótica Motor Board);