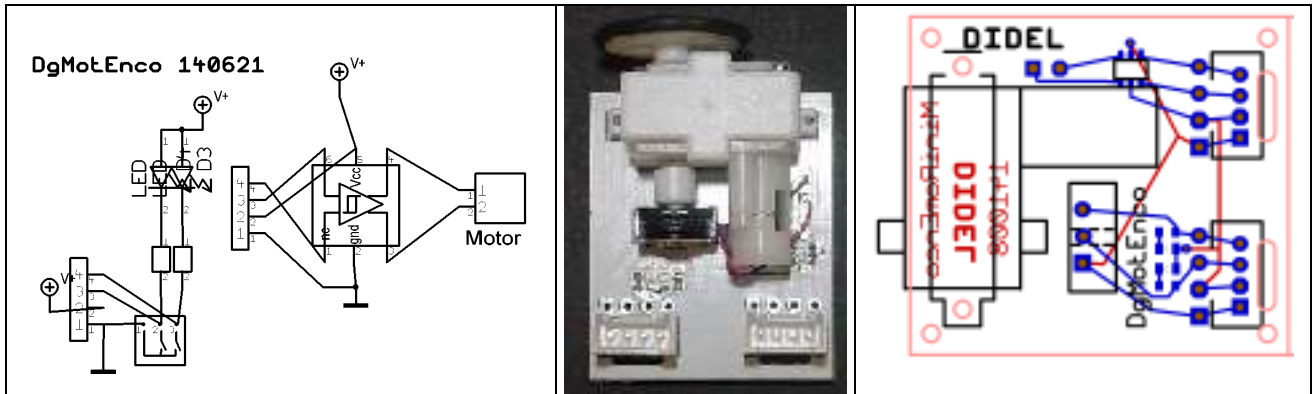


## Dg-MotEnco



Voir le texte plus général qui décrit la première version de MotEnco:

[www.didel.com/diduino/Encoder.pdf](http://www.didel.com/diduino/Encoder.pdf)



### Hardware

The motor is a Vigor Bo-30, gear factor 1:196

The encoder is Kaith part EN97511, 48 transitions per turn.

The shaft adapter is a specially made part developed by Didel.

The mechanical encoder has non constant friction that will disturb low speed movement if not servo-controlled. However, with a resolution of 48 transitions per turn, servo control theory is not applicable. Bouncing is rather important (plan for 20ms delay).

The MotEnco is primarily an educational tool. It is used on the Xplus for distance measurement and simple correction to the motor speed in order to get straight lines.

### Software

Making the motor move is trivial. Set the following combination on the two motor bits

- 0 0 stop
- 0 1 move in one direction
- 1 0 move in the other direction
- 1 1 stop

You can use PWM down to 20-30%. PFM can make the motor move as slowly as you wish, see

<http://www.didel.com/diduino/PwmPfm.pdf>

<pre>//TestEncoME.ino Enco wires on Arduino pins 8,9 #define IN0 digitalRead (8) #define IN1 digitalRead (9) byte olda, oldb, newab ; // portB 0b000000AB  void setup() {   DDRB = 0x00110000 ; // 11..8 in 13 12 out   DDRC = 0xFF ; // affichage 6 bits   PORTC = 0 ;   DDRD = 0xF0 ; //moteurs   PORTD = 0x0C0 ;   analogWrite (5,250) ; }</pre>	<pre>enum {e0,e1,e2,e3} next = e0; int pos=0 ; void loop() {   switch(next)   {   case e0:     if (IN0) {pos++; next = e1; break;}     if (IN1) {pos--; next = e3; break;}     break;   case e1:     if (IN1) {pos++; next = e2; break;}     if (!IN0) {pos--; next = e0; break;}     break;   case e2:     if (!IN0) {pos++; next = e3; break;}     if (!IN1) {pos--; next = e1; break;}     break;   case e3:     if (!IN1) {pos++; next = e0; break;}     break;   }   PORTC = pos ; // visible effect on port's   leds   delay (20); //debounce }</pre>