

# Pavlin.si -> same zanimive stvari

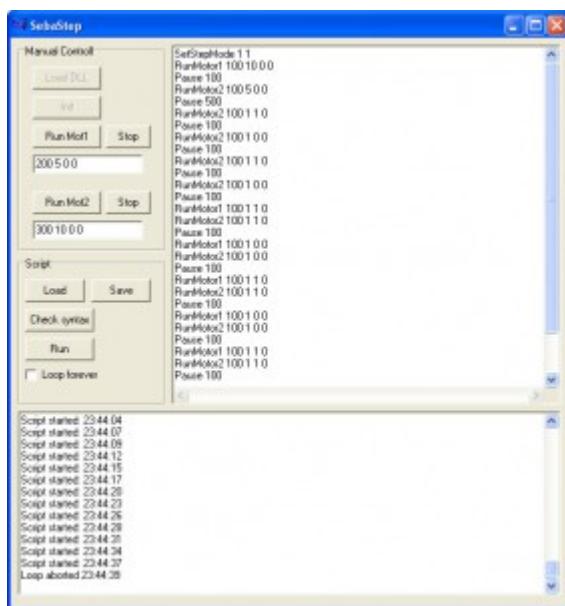
Šola, MTB, potepanja, elektronika, foto

Višinski diagram

Manjka GPX zapis.

## Stepper Bee+

Pred časom mi je pod roke prišel mali kontrolerček koračnih motorjev Stepper Bee+. Ker je priloženi software bolj čuden, sem naredil svojega. V bistvu je en enostaven interpreter, ki uporablja modulu priloženo dll knjižnico.



Za arhiv celotnega projekta za borland c++ builder pošlji mail.

Tule je zip arhiv, v katerem je EXE, DLL in help datoteka. (MD5 hash: e0fd54f766a4c0b8b64840ffa62b244d, velikost 412.59KB)

Ukazi so:

### Commands

#### [SetStepMode M1Mode M2Mode](#)

The StepMode can be set to one of three possible options... Full Step, Wave Step and Power Off. The SetStepMode command has therefore three possible values for its

---

parameters M1Mode and M2Mode:

M1Mode, M2Mode Step mode

0 Wavestep mode

1 Full step mode

2 Power Off mode

These mode settings are independent for motors 1 and 2 and can be changed at any time, even during a step interval.

### **StopMotor1 outputs**

**StopMotor2 outputs**  
To terminate the current task being performed by motor1 or motor2 immediately while updating the current state of the other switching outputs.

Outputs - integer in the range 0 to 7 corresponding to the bit pattern for on/off of the additional switching outputs associated with motor 1. e.g. a value of 5 (which is 00000101 in binary) would result in outputs 1 and 3 being on.

### **RunMotor1 steps interval direction outputs**

### **RunMotor2 steps interval direction outputs**

Steps - integer in the range 1 to 16000 corresponding to the number of steps to execute

Interval - integer in the range 1 to 16000 corresponding to the time interval in milliseconds between each step

Direction - integer in the range 0 to 1. Zero corresponds to forward and 1 to reverse.

Outputs - integer in the range 0 to 7 corresponding to the bit pattern for on/off of the additional switching outputs associated with motor 1. e.g. a value of 5 (which is 00000101 in binary) would result in outputs 1 and 3 being on.

### **Example:**

To run Motor1 forward for 200 steps with 50ms between steps and all additional switching outputs off would use the following command:

**RunMotor1** 200 50 0 0

### **Pause msec**

Command halts parsing for specified time in milliseconds. Time resolution is 10ms. As consequence, the interval msec is rounded to nearest 10ms and shortest interval is 10ms.

### **WaitMotor1**

---

**WaitMotor2**

Command waits for motor 1 or motor 2 to stop motion.

**# comment**

Comments are lines starting with # and one white space.