Arduino Mega PWM API description

The code for Arduino Mega is aimed to control I/O pins and provide Serial port interaction with the PC.

According to the initial decision, PWM pin #2,3,5-12 can be used to control coils, PWM pins #44-46 are used to control the light, or other analog devices, and analog pins #A0-A3 are used to receive data from thermistors.

The code for Arduino includes software watch dog timer, which turns off PWM pins if the PC command was not received during some period of time (3 sec for coils pins, 10 min for LED pins). Also the inbuild Arduino’s watchdog timer is activated to restart the board in case when the system freezes for more than 8 sec.

Frequency of PWM pins is raised up to 3921.16 Hz (instead of default 490Hz or 980 Hz). The serial connection baud rate is 19200.

The typical command from PC to Arduino is a byte-array, sent via Serial connection. The first byte of the array is the command code, other bytes include channels numbers and PWM states.

|  |  |
| --- | --- |
| Command codes (decimal) | |
| 250 | All coils off |
| 251 | Single coil-pin control when other pins are turned off |
| 252 | Periodic switching between up to 4 coil-pins with a period of 7 msec. The command array is extended: [252,channel1,value1,channel2,value2,channel3,value3,channel4,value4] |
| 253 | Individual coil-pin control (set the state for this pin only) |
| 230 | All LEDs off |
| 231 | Individual LED-pin control (set the state for this pin only) |
| 233 | Measure analog input and send result to Serial |
| 234 | Measure voltage on single analog channel [234,channel,0] |
| 235 | Measure voltage on all 15 channels [235,0,0] |
| 220 | All coils off (Pololu driver) |
| 221 | Single coil-pin control when other coils are turned off (Pololu driver) [221, pwm\_pin, direction\_pin, pwm\_value, direction\_value] |
| 223 | Individual coil control (set the state for this coil only, Pololu driver) [223, pwm\_pin, direction\_pin,pwm\_value, direction\_value] |

A simple Python example to send the command to Arduino to run the inbuild Arduino LED at port #13:

import array

import serial

ser = serial.Serial(self.usbport, 19200, timeout=1)

time.sleep(2)

# individual control (253) of PIN 13, PWM power – 50 (max is 255)

ar = array.array('B', [253, 13, 50]).tobytes()  
ser.write(ar)

ser.close()