

Лабораторна робота №2

Тема:

Мета:

Завдання:

XYZrobot має 18 розумних сервоприводів, кожен з яких має 330 ° ефективного керування положенням, що дозволяє виконувати широкий спектр складних рухів, включаючи танці, ходьбу та можливість повертатися з майже будь-якого положення лежачи. Робот Bolide Y-01 попередньо запрограмований для виконання багатьох цих рухів у відповідь на команди з включеного контролера Bluetooth або смартфона або планшета під керуванням програми XYZrobot.



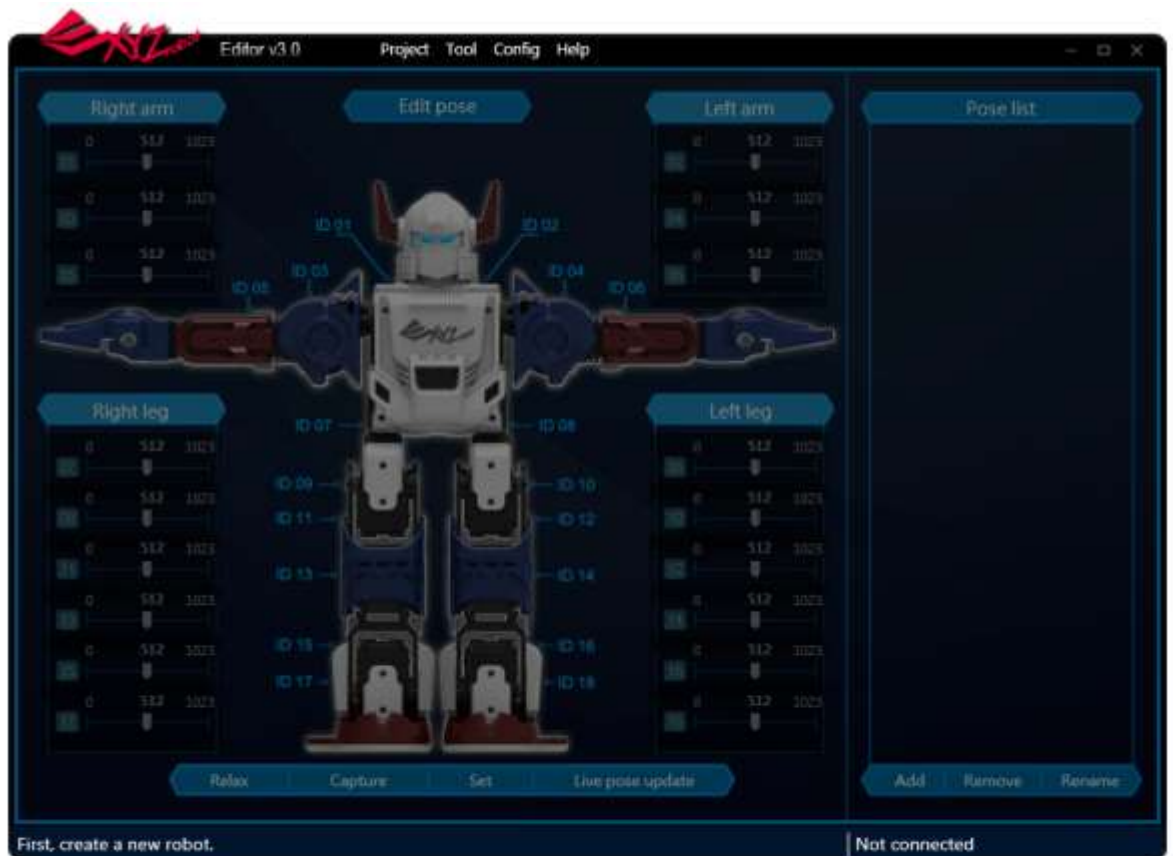
Bolide Y-01 оснащений мікроконтролером ATmega1280, сумісним з Arduino, який можна запрограмувати для виконання нових рухів і дій за допомогою Arduino IDE і XYZrobot Editor. Плата управління містить триходовий акселерометр для підтримки постуральної стійкості і виявлення падінь, а робот також має датчик відстані в грудях, який може виявляти об'єкти перед ним.

В основі робота Bolide лежать його сервоприводи 18 A1-16, які можуть виробляти до 25 кг-крутного моменту при 12 В. Ці висококонфігуровані сервоприводи можуть бути ланцюговими на одній шині і передавати температуру, швидкість, і повертають дані назад до контролера. Кожен сервопривод містить чотириколірний світлодіод для відображення статусу сервосистеми.

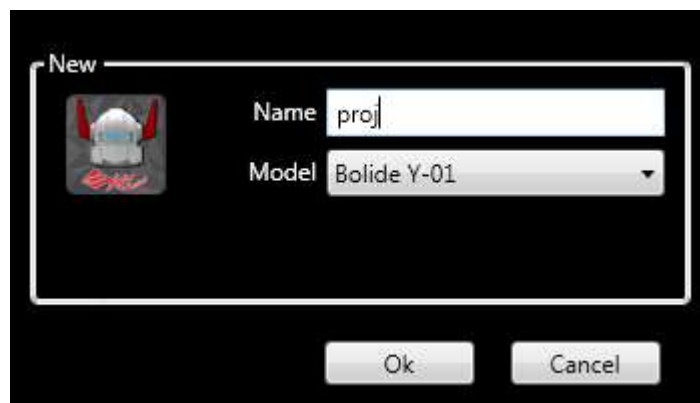


Цей розумний сервопривід від XYZrobot був розроблений для робота Bolide Y-01 Advanced Humanoid Robot і 6 DOF Robotic Arm, що робить його придатним для додатків, які вимагають багато сильних і складних дій. На відміну від звичайних сервоприводів RC, ці сервоприводи здатні постійно обертатися на 360 °, але при цьому пропонують контроль положення за ефективним діапазоном 330 °. Вони мають серійний інтерфейс TTL, який використовується для налаштування та керування сервоприводом і отримання зворотного зв'язку про положення, швидкість і температуру.

Запускаємо програму для керування рухами гуманоїда



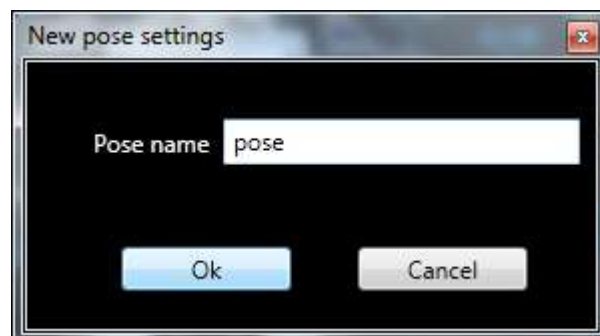
Для початку роботи з гуманоїдом необхідно створити новий проект



Після цього решта вікон програми стануть активними



Далі необхідно створити нову позу





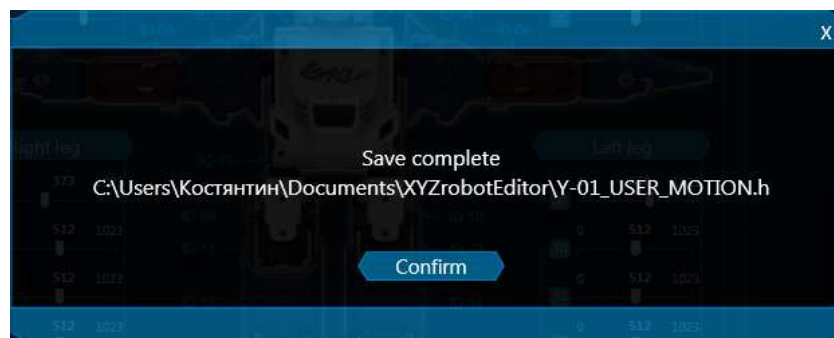
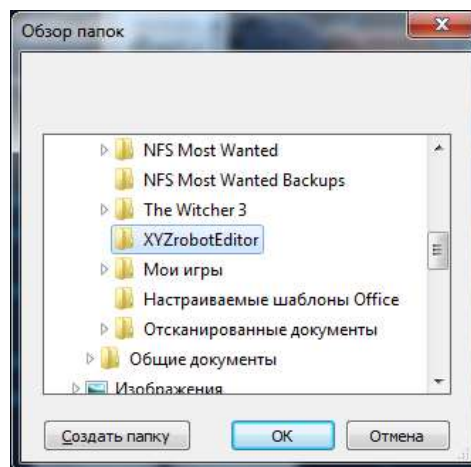
В даній програмі є можливість в реальному часі відобразити зміни позиції за допомогою повзунків. Для цього необхідно під'єднатися за допомогою USB до робота. У відповідному пункті меню обираємо пункт підключення до Serial port і з отриманого списку обираємо необхідний нам.



Для відображення змін в редакторі необхідно увімкнути функцію “Live pose update”

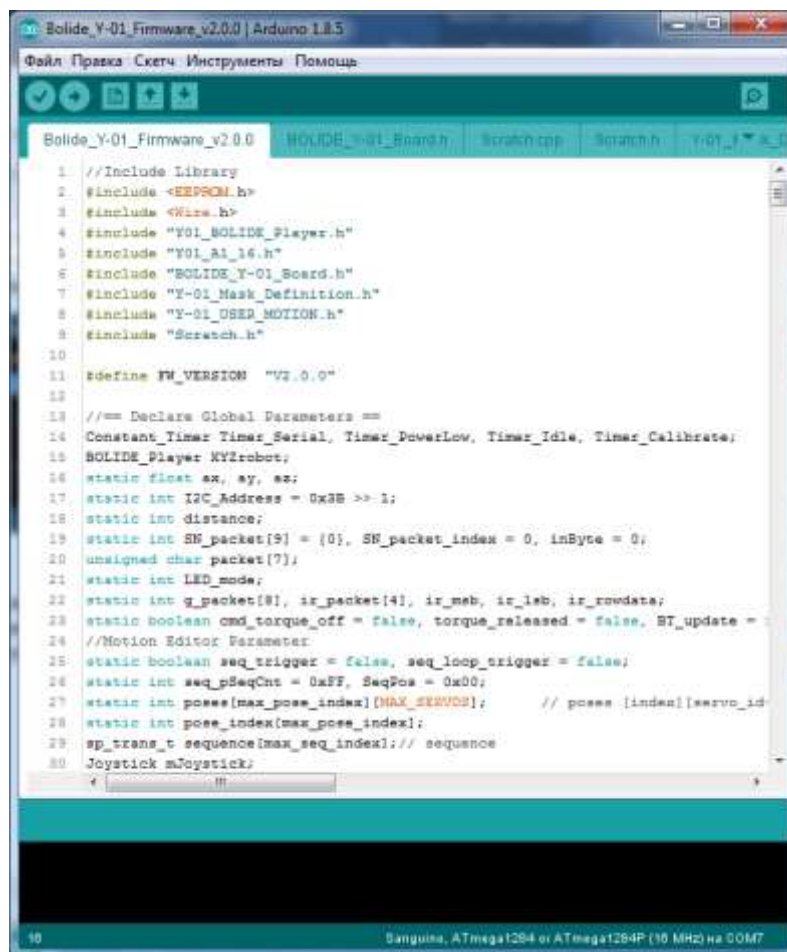


Далі для подальшого програмування гуманоїду нам необхідно отримати .h файл для того щоб замінити стандартний файл прошивки. Тому обираємо в меню пункт Tools->Export to AVR. Зберігаємо файл в папку з основними файлами.



Даний файл необхідно розмістити безпосередньо у папку з основною прошивкою та .ino файлом

Сама прошивка відбувається з програми Arduino IDE. Відкриваємо основну програму.



```
1 //Include Library
2 #include <EEPROM.h>
3 #include <Wire.h>
4 #include "Y01_BOLIDE_Player.h"
5 #include "Y01_A1_16.h"
6 #include "BOLIDE_Y-01_Board.h"
7 #include "Y-01_Mask_Definition.h"
8 #include "Y-01_USER_MOTION.h"
9 #include "Scratch.h"
10
11 #define FW_VERSION "V2.0.0"
12
13 //== Declare Global Parameters ==
14 Constant_Timer Timer_Serial, Timer_PowerLow, Timer_Idle, Timer_Calibrate;
15 BOLIDE_Player MYrobot;
16 static float ax, ay, az;
17 static int I2C_Address = 0x3B >> 1;
18 static int distance;
19 static int SN_packet[9] = {0}, SN_packet_index = 0, inByte = 0;
20 unsigned char packet[7];
21 static int LED_mode;
22 static int g_packet[8], ir_packet[4], ir_meb, ir_lsb, ir_rowdata;
23 static boolean cmd_torque_off = false, torque_released = false, BT_update =
24 //Motion Editor Parameter
25 static boolean seq_trigger = false, seq_loop_trigger = false;
26 static int seq_pSeqCnt = 0xFF, SeqPos = 0x00;
27 static int poses[max_pose_index][MAX_SERVOS]; // poses [index][servo_id]
28 static int pose_index(max_pose_index);
29 sp_trans_t sequence[max_seq_index]; // sequence
30 Joystick mJoystick;
```

В пункті інструменти обираємо необхідні плату та контролер, також вказуємо відповідний Serial Port та натискаємо кнопку Upload.





Оформлення звіту

Звіт повинен містити:

- назву та мету роботи;
- алгоритм роботи робота (стратегічний та тактичні рівні);
- опис технічних складових робота та їх характеристики (сприймаючі елементи, виконавчі механізми та ін.);
- програму завдання, написану в Arduino IDE або в XYZrobotEditor;
- висновки по роботі.