

УДК 004.415

Костючко С.М., Склянчук О.М., Ілюшик Р.С.
Луцький національний технічний університет

МАТЕМАТИЧНІ ОСНОВИ ТА ПРОГРАМУВАННЯ МІКРОКОНТРОЛЕРА АТМЕГА328 З ВИКОРИСТАННЯМ POLOLU 3PI ROBOT

Костючко С.М., Склянчук О.М., Ілюшик Р.С. Математичні основи та програмування мікроконтролера АТМЕГА328 з використанням POLOLU 3PI ROBOT. У даній статті досліджено математичні аспекти та можливості програмування мікроконтролера АТМЕГА328 для паралельного виконання ним поставлених задач. Наведено лістинг коду виконання команд роботом та його структурну схему.

Ключові слова: АТМЕГА328, 3pi robot, C++, мікроконтролер, модуль.

Костючко С.Н., Склянчук А.Н., Ілюшик Р.С. Математические основы и программирование микроконтроллера АТМЕГА328 с использованием POLOLU 3PI ROBOT. В данной статье исследованы математические аспекты и возможности программирования микроконтроллера АТМЕГА328 для паралельного выполнения им поставленных задач. Приведено листинг кода выполнения команд роботом и его структурную схему.

Ключевые слова: АТМЕГА328, 3pi robot, C ++, микроконтроллер, модуль.

Kostiuchko S.M., Sklyanchuk O.M., Pyushik R.S. Mathematical bases and programming of microcontroller АТМЕГА328 using POLOLU 3PI ROBOT. In this article the mathematical aspects and possibilities of programming the microcontroller АТМЕГА328 for parallel execution of the set tasks are given. Listing the command code execution robot and its block diagram.

Keywords: АТМЕГА328, 3pi robot, C ++, microcontroller, module.

Вступ. Людство з самого початку свого існування постійно розвивалось і тепер ми живемо в часи, коли техніка стала невід'ємною частиною нашого життя, і практично "влилась" в кожен сферу людської діяльності. Завдяки технологіям, найсміливіші фантазії людського розуму стають реальністю. Вони допомагають нам спілкуватись, пересуватись, а для деякого і взагалі стають частиною тіла. І в час прогресу технологій, коли машини покликані полегшити наше життя, на допомогу приходить робототехніка, дозволяючи нам досягнути тих висот, яких не можливо досягнути людськими силами. Тепер цілком зрозуміло, що подальший розвиток людства неможливий без розвитку технологій. Тому безперечно проблеми робототехніки є актуальними.

Робот Pololu 3pi розроблений, щоб досягти успіху в змаганнях по слідуванні лініями і проходженні лабіринту. Він має невеликий розмір (9,5 см / 3,7 дюйма, вагу 83 г / 2,9 унції без батарей) і чотири батарейки типу ААА, в той час як система живлення двигунів працює на постійній 9.25 В напрузі незалежно від заряду акумулятора. Регульована напруга дозволяє 3pi досягати швидкості до 1 м/с при виконанні точних поворотів, які не залежать від напруги акумулятора.

Pololu 3pi надає чудову платформу для людей з невеликим досвідом програмування, щоб вивчити робототехніку, і покращити свої навички програмування на мові С. В основі його лежить мікроконтролер Atmel АТmega328P, що працює на частоті 20 МГц і має 32 КБ програмної пам'яті, 2 КБ ОЗУ і 1 КБ постійної пам'яті EEPROM. Atmel Studio забезпечує зручне середовище розробки, а великий набір бібліотек, що надаються Pololu, робить його зрозумілим та простим для взаємодії з усім апаратним забезпеченням. Пристрій 3pi також сумісний з популярною платформою розробки Arduino. Архітектуру робота Pololu 3pi продемонстровано нижче.

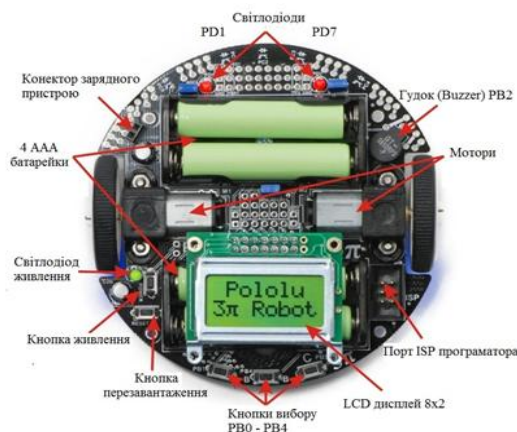


Рисунок 1 – Зовнішній вигляд та комплектація Pololu 3pi (верхня сторона)

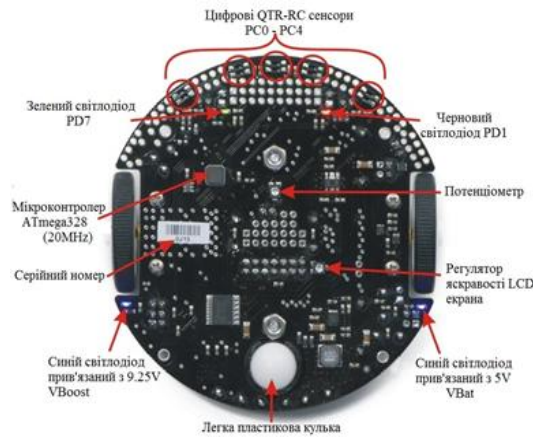


Рисунок 2 – Комплектація Pololu Зрі (нижня сторона)

Можливості робота Зрі можуть бути поліпшені за рахунок встановлення плати розширення Зрі Expansion Kit, що монтується на «другий поверх» робота та дає можливість користувачеві застосувати ряд апаратних модулів. Ключовою особливістю плати розширення Зрі є те, що вона дозволяє використовувати один з потужних 32-бітних контролерів ARM, як високорівневого контролера для робототехніки, пропонуючи значно більшу обчислювальну потужність і вільні лінії введення-виведення, ніж вбудований в робот Зрі 8-бітний AVR-мікроконтролер. Базові роботи Зрі, попередньо запрограмовані для роботи в якості послідовного підлеглого пристрою Mbed підключаються безпосередньо до роз'єму на платі розширення і послідовних ліній бази Зрі. Підтримка бібліотеки і приклади програм для Зрі доступні на сайті mbed.org, що дозволяє легко запускати і керувати вашим mbed-роботом Зрі.

Також є роз'єм для Wixel Programmable USB Wireless Module. Даний модуль дозволяє використовувати бездротові можливості для управління та обміну інформацією між вашим роботом Зрі, надаючи сокети для модулів XBee або Wixel. Плата розширення з'єднує ці модулі з одним з послідовних портів mbed, а перемички можуть використовуватися для безпосереднього підключення до послідовних ліній бази Зрі, тому робот може підтримувати бездротовий зв'язок навіть без mbed в якості контролера високого рівня.

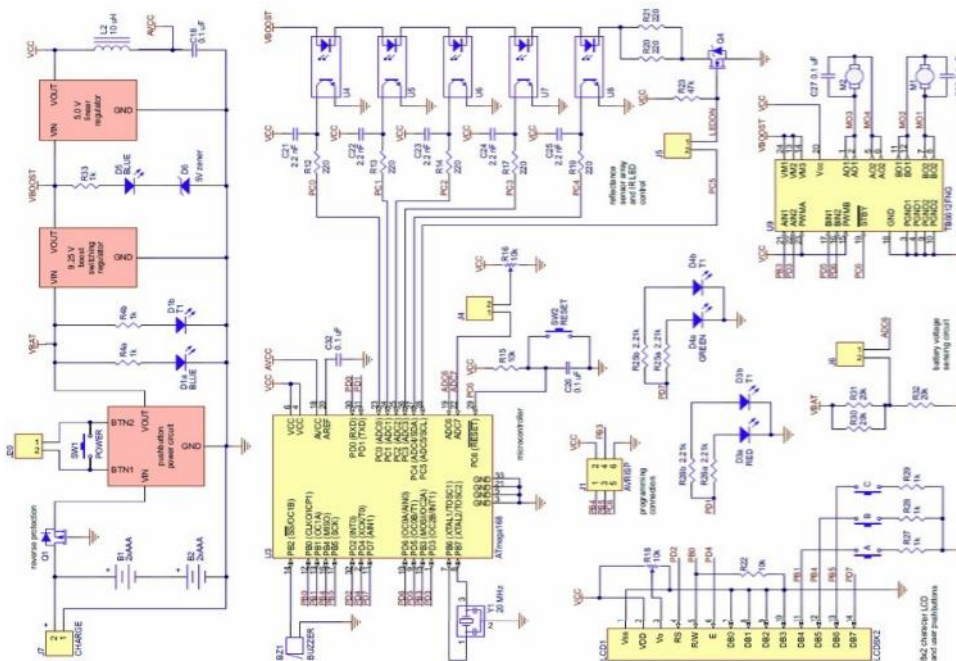


Рисунок 3 – Pololu Зрі схематична електронна схема

Математичні основи та програмування.

Для адекватної роботи робота та виконання ним поставлених задач необхідно було провести низку математичних обрахунків таких як, визначення швидкості руху, швидкості повороту, забезпечити правильну роботу сенсорів та світлодіодів, своєчасне програвання мелодії. Результати обрахунків продемонструємо в коментарях до програмного коду.

Планом передбачалося проведення завдання у 3 етапи:

1. Відтворення мелодії Buzzer-ом.

Насамперед слід оголосити масив нот, оголосити функцію відтворення мелодії та написати необхідну мелодію:

```
const char starwars[] PROGMEM =
"! V15 O5 L4 MS ggg L8 e-. L16 b- L4 g L8 e-. L16 b- L2 g O6 L4 ddd L8 e-. O5 L16 b- O5 L4
f+ L8 e-. L16 b- L2 g"
"O6 L4 g O5 L8 g-. L16 g O6 L4 g L8 f+. L16 f L16 ee- L8 e R O5 L8 a- O6 L4 d- L8 c. O5 L16
b"
"O5 L16 ba L8 b R L8 e L4 f+ L8 e. L16 f L4 b L8 g. L16 b O6 l2 d"
"O6 L4 g O5 L8 g-. L16 g O6 L4 g L8 f+. L16 f L16 ee- L8 e R O5 L8 a- O6 L4 d- L8 c. O5 L16
b"
"O5 L16 ba L8 b R L8 e L4 f+ L8 e. L16 b L4 g L8 e. L16 b L2 g R R"
play_from_program_space(starwars);
```

де «cdefgab» - ноти, відповідно «до, ре, мі, фа, соль, ля, сі», V15 – гучність Buzzer-а (Максимум 15), O(і цифра від 4 до 6) – це музичні октави, L(і цифри 4, 8, 16) – цілі, половинні і четвертинні ноти.

2. Написання коду програми для руху робота по чорній або білій лінії.

Першим кроком на даному етапі є оголошення масиву для зберігання значень сенсора та запуск ініціалізації. Після чого ми повинні забезпечити аргумент «sensors» для read_line(). Крім визначення положення лінії паралельно засвічуємо світлодіоди.

```
int main()
{
    unsigned int sensors[5];
    initialize();
    while(1)
    {
        unsigned int position = read_line(sensors,IR_EMITTERS_ON);
        if(position < 1000)
        {
            set_motors(0,100);
            left_led(1);
            right_led(0);
        }
        else if(position < 3000)
        {
            set_motors(100,100);
            left_led(1);
            right_led(1);
        }
        else
        {
            set_motors(100,0);
            left_led(0);
            right_led(1);
        }
    }
}
```

3. Злиття двох програм в одну і оптимізація коду для отримання бажаного результату.

В функції ініціалізації викликаємо написану нами мелодію, вводимо масив для зберігання значень сенсора, назначаємо автокалібрування датчиків, а також викликаємо функцію, що записує набір показів датчиків та стежить за їх граничними значеннями.

```

for(counter=0;counter<80;counter++)
{
    if(counter < 20 || counter >= 60)
        set_motors(40,-40);
    else
        set_motors(-40,40);
    calibrate_line_sensors(IR_EMITTERS_ON);
    delay_ms(20);
}

set_motors(0,0);
unsigned char button = wait_for_button(ANY_BUTTON);
if(button == BUTTON_B)
{
    unsigned int position = read_line(sensors,IR_EMITTERS_ON);
    clear();
    print_long(position);
    lcd_goto_xy(0,1);
    display_readings(sensors);
    delay_ms(100);
    play_from_program_space(go);
    while(is_playing());
    play_from_program_space(starwars);
}
    
```

Під час роботи Pololu 3pi реалізує дві фази: перша – ініціалізація і фаза калібрування за допомогою цифрових OTR-RC сенсорів, яка виконується один раз, на початку і її метою є відслідковування наступних даних:

- Зміна значень від 0 (абсолютно біла поверхня) до 2000 (абсолютно чорна поверхня) датчиків лінії.

- Показник напруги батареї, адже завжди необхідно знати скільки часу робот ще може пропрацювати.

- Калібрування датчиків. Це відбувається наступним чином: робот повертається направо і наліво, залишаючись на лінії (отримує мінімальні і максимальні значення від датчиків). Мінімальні і максимальні значення зберігаються в RAM-пам'яті. Це дозволяє повернутися до початкових значень, що можуть змінюватися від 0 до 1000 для кожного датчика.

- Значення відкаліброваних датчиків відображаються в гістограмі, щоб точно знати, чи працюють датчики лінії належним чином перед початком роботи робота.

У другій фазі, 3pi встановить частоту обертання двигуна (в залежності які дані надходять з датчиків). Ці кроки, контролюють рух робота, щоб той зміг постійно залишатися на лінії.

Висновки та перспективи подальшої роботи. Завдяки багатofункціональним можливостям Pololu 3pi, у середовищі Atmel Studio, було розроблено програмне забезпечення, яке реалізує одночасне відтворення унікальних мелодій Buzzer-ом, в залежності від натиснутої кнопки користувача, і рух робота по чорній або білій лінії. Отриманий результат функціонування 3pi може бути прототипом для розробки більш цікавих і складніших систем. Тобто, виконання мікроконтролерною технікою паралельно декількох завдань, як було показано на прикладі робота.

1. Korn G.A., Korn T.M. Mathematical handbook for scientists and engineers / McGRAW-HILL BOOK. – N.Y., Toronto, London. – 1961. – 720 pp.
2. Introduction to Robots, Robotics. 2003 [cited 2013 Jan 17]. Available from: <http://www.galileo.org/robotics/intro.html>.
3. Greenblatt D. 6 ways robots create jobs. 2013 [cited 2013 Apr 9] <http://www.inc.com/drew-greenblatt/6-ways-robots-create-jobs.html>.
4. Mayer, T.N., Jackson, & Lockey, M.E. (1995). A new role for microcontroller courses: integrating EE curricula. *Frontiers in Education*, 25th Annual Conference, November 1-4, Georgia Institute of Technology.
5. Hanson, A. (1981). A microcontroller laboratory for electrical engineering seniors, *IEEE Trans. Educ.* E24, pp. 8-14.
6. Schultz, A. (1991). New directions in microcomputer education. *1991 Frontiers in Education Conference*, 99. 642-645.

7. Hamrita, T.K., McLendon, R.W. (1997). A new approach for teaching microcontroller courses. *Int. J. Eng. Edu.*, Vol. 13, No. 4, pp. 269-274.
8. Tchaban V., Tchaban O., Kostiuchko S., Tchaban Z. About one two-point boundary value problem // *Technical news*, 2011 / 1(33), 2(34). – PP. 10-12.
9. Tchaban V., Tchaban O., Kostiuchko S., Tchaban Z. The theory of electromagnetic circuits. – In monograph: *Computing in Science and Technology*, Rzeszow: Wyd-wo Uniwersytetu Rzeszowskiego, ISBN 978-83-7338-895-6, 2012/13, 172 p. (pp. 34-55).