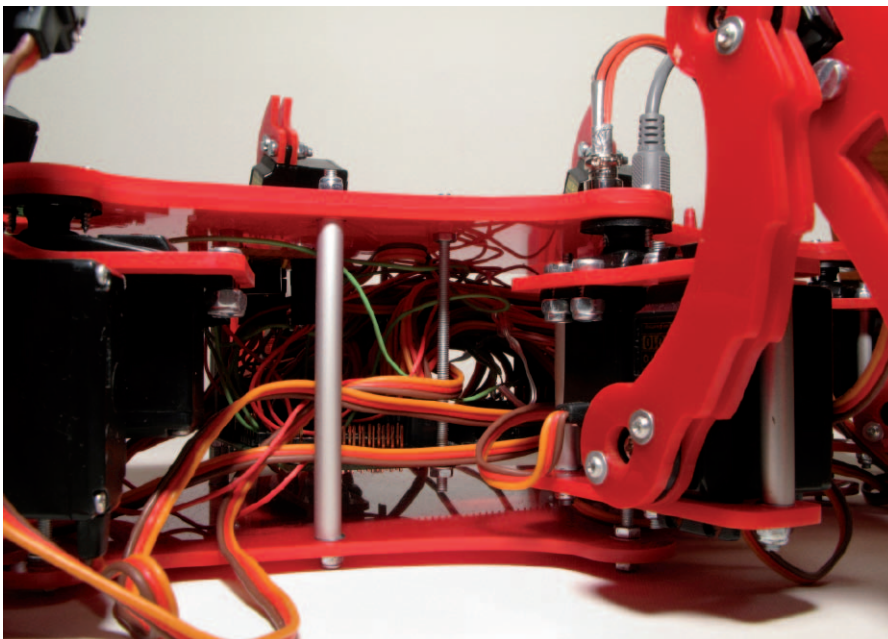


Maniek

Robot kroczący

Jak powiedział Mark Twain, „ludzie nie dlatego przestają się bawić, że się starzeją, lecz starzeją się, bo przestają się bawić”.

W artykule prezentujemy efekty takiej „zabawy”, które zdumiały nawet fachowców. Bo sterowanie 19 serwomechanizmami i budowa algorytmów poruszania się robota, to nie lada sztuka. A przecież robot został zbudowany przez człowieka, który jak sam o sobie pisze, nie miał pojęcia o elektronice i programowaniu.



Fot. 1. Widok płytek z pleksi tworzących korpus robota. Na górze widać wzmocnioną płytkę o podwójnej grubości

Po przeprowadzeniu analizy możliwości zdobycia poszczególnych elementów do budowy robota oraz analizie rozwiązań dostępnych w Internecie przyjąłem następujące założenia projektowe:

- robot kroczący na 6 nogach, o prostej, zwartej i estetycznej konstrukcji,
- inteligencja robota – płytki mikrokomputera Arduino,
- sterownik serwomechanizmów – pracujący niezależnie od płytki Arduino,
- osobne, przewodowe zasilanie serwomechanizmów i elektroniki. Zasilanie z kabla,
- maksymalne obniżenie ceny konstrukcji.

Powyższe założenia koncepcyjne wynikały z moich możliwości i naprawdę nie finanse były tu najważniejsze. Moje wykształcenie w żaden sposób nie jest związane z robotyką, elektroniką czy programowaniem. A w związku z tym, robot musiał być prosty do wykonania oraz zaprogramowania. Ponadto, miał stanowić platformę łatwą do rozbudowy, gdyż to dopiero pierwsze eksperymenty miały wykazać, w którym kierunku rozwijana będzie konstrukcja.

Konstrukcja robota

Wzorem dla mojej konstrukcji były roboty *Phoenix* firmy *Lynxmotion* oraz *MSR-H01* firmy *Micromagic System*.

Robot 6-cio nożny, potocznie zwany „hexapodem”, wykonano z pleksi o grubości 3



Fot. 2. Płytki utrzymywane są w stałej odległości przez tuleje dystansowe

mm. Całość wycięto laserem zgodnie z moim projektem. Nie używałem przy tym programu do modelowania przestrzennego, a jedynie przed wycięciem konstrukcji robota wykonałem kartonowy model w skali 1:1. Dla usztywnienia korpusu robota dodatkowo wzmocniłem go płytą pleksi w tej górnej części, do której zamocowane są serwomechanizmy. Taką podwójną warstwę pleksi widać np. na fot. 1.

Całość elektroniki umieściłem wewnątrz korpusu robota pomiędzy płytkami pleksi. Płytki tworzące korpus są zamocowane przy użyciu tulei dystansowych i śrub (fot. 2).

Każda z nóg ma 3 punkty swobody (fot. 3). Dla zwiększenia wytrzymałości i sztywności nóg zaprojektowano podwójne kształtki z pleksi. Na końcu każdej z nich zamontowano plastikowe nakładki zabezpieczające na śruby M10 dla zapewnienia lepszej przyczepności do terenu i zabezpieczenia pleksi przed zniszczeniem. Nogi są napędzane przez serwomechanizmy.

Nogami robota porusza 18 serwomechanizmów typu *Tower Pro SG-5010*, natomiast ruchomy sonar jest obsługiwany przez jeszcze jeden, dodatkowy serwomechanizm. Dobrze widać je na fot. 4 ukazującej robota w widoku od góry.

Robot ma całkowite wymiary wynoszące $35 \times 25 \times 15$ cm i bez akumulatora waży ok. 1,5 kg. Poruszanie się przy wadze własnej ok. 2,5 kg (czyli obciążonego np. kilogramowym akumulatorem) raczej też nie powinno sprawiać mu większych problemów.

Inteligencja, czyli płyta główna

Postanowiłem nie budować własnego sterownika, lecz posłużyć się gotowym mikrokomputerem. Po namyśle zastosowałem produkt o nazwie *Arduino Mega*. Jest to płyta mikrokomputera z zamontowanym całym niezbędnym wyposażeniem. Oprogramowanie dla niej tworzy się z użyciem darmowego środowiska *Arduino*.

Zależało mi na zastosowaniu mikrokomputera, który będzie umożliwiał rozwój i rozbudowę robota oraz pomieścił spory kod

programu. Dlatego wybrałem płytę mającą najwięcej możliwości. *Arduino Mega* jest wyposażona w mikrokontroler ATmega1280, który można programować przez port USB, bez potrzeby dołączania dodatkowego programatora. Podstawowe parametry techniczne zastosowanego „mózgu” są następujące:

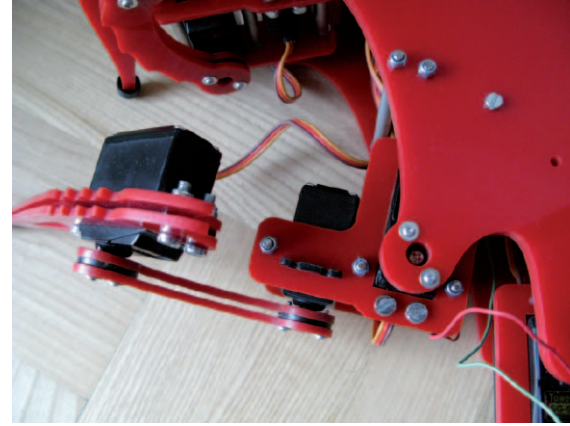
- mikrokontroler ATmega1280 zasilany napięciem 5 V;
- napięcie zasilania 7...16 V;
- 54 doprowadzenia I/O, z których 14 dostarcza sygnał PWM;
- 16 wejść analogowych;
- pamięć Flash o pojemności 128 kB, z której 4 kB są zajmowane przez bootloader;
- SRAM 8 kB, EEPROM 4 kB, zegar 16 MHz.

Płyta może być zasilana bezpośrednio z portu USB lub zewnętrznego źródła zasilania. Ja zamocowałem złącza (fot. 5) umożliwiające doprowadzenie zasilania z zewnętrznego zasilacza, co jest bardzo przydatne podczas pisania programu dla robota.

Sterownik serwomechanizmów

Mimo iż płyta *Arduino Mega* jest w stanie obsłużyć do 48 serwomechanizmów, to zastosowanie osobnego sterownika serwomechanizmów ma sens. Pozwala bowiem odciążyć płytę główną i przeznaczyć moc obliczeniową mikrokontrolera na realizację innych zadań. Gdyby mikrokontroler bezpośrednio sterował napędami, to musiałby odświeżać stan serwomechanizmów co 20 ms co negatywnie wpłynęłoby na szybkość pracy robota.

Z opisanych wyżej powodów wyposażyłem robota w osobny sterownik serwomechanizmów typu *SD-21* angielskiej firmy *Devantech Ltd*. Sterownik ten jest w stanie obsłużyć do 21 serwomechanizmów z okresem odświeżania 20 ms, co oznacza, że jest



Fot. 3. Każda noga ma trzy punkty swobody. Na zdjęciu widać serwomechanizmy zastosowane do poruszania pojedynczą nogą

w stanie obsłużyć większość serwomechanizmów dostępnych na polskim rynku. Komunikacja z płytą główną odbywa się poprzez interfejs I²C. Dodatkową zaletą tego sterownika jest możliwość zapewnienia rozdziału zasilania serwomechanizmów i płyty mikrokomputera. Jest to o tyle ważne, że podczas ich pracy skokowo pobierany jest duży prąd. Rozdzielenie zasilania sterownika i płytki mikrokomputera powoduje, że praca tej drugiej nie jest zakłócana.

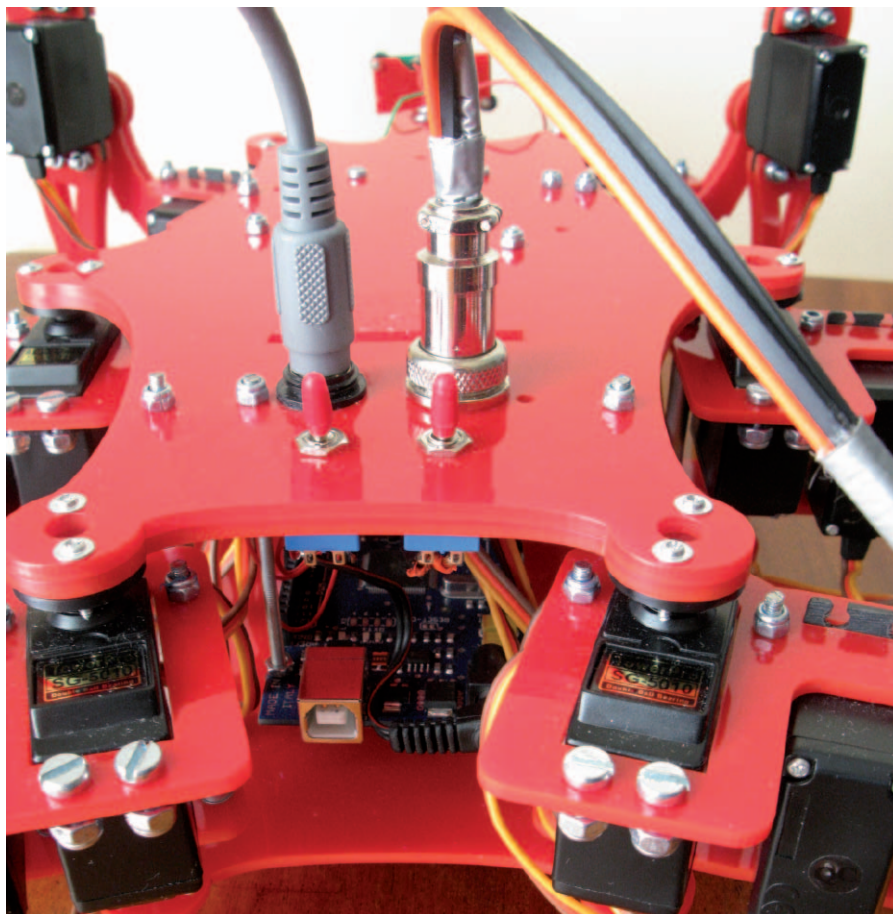
Podstawowe parametry techniczne sterownika serwomechanizmów *DS21* są następujące:

- 21 kanałowy sterownik serw z mikrokontrolerem PIC18F2220;
- okres odświeżania 20 ms;
- komunikacja przez interfejs I²C;

Istotną cechą, która wpłynęła na decyzję o zakupie, jest to, że sterownik doskonale współpracuje z płytą mikrokomputera *Arduino Mega*.



Fot. 4. Robot Maniek w widoku od góry. Widać wszystkie 19 serwomechanizmów



Fot. 5. Złącza przeznaczone do zasilania i programowania płytki *Arduino Mega* oraz układów wykonawczych

Dźwięk

Robot może wydawać dźwięki, co pozwala np. na stwierdzenie statusu jego programu. Sygnały dźwiękowe wydawane są poprzez buzzer piezoceramiczny z generatorem.

Zasilanie

Zdecydowałem się na zewnętrzne źródło zasilania. W prototypie zastosowałem zasilacz komputerowy, z którego osobno wyprowadziłem napięcia 5 V i 12 V. Napięcie z zasilacza jest przekazywane do robota z użyciem kabla o długości ok. 5 m. Z napięcia 5 V zasilane są serwomechanizmy (szacunkowy, maksymalny pobór prądu to ok. 6,5 A), natomiast napięcie 12 V jest używane do zasilania płytki mikrokomputera. Robota skonstruowałem w taki sposób, że bez żadnych problemów można zainstalować akumulatory i zastosować je do zasilania całości. Przewidziano dla nich miejsce na górze korpusu robota.

Czujnik odległości

Z przodu robota zamontowałem ultradźwiękowy czujnik odległości typu *SRF05* firmy *Devantech Ltd.* Czujnik ma pojedyncze wyprowadzenie umożliwiające nadawanie i nasłuch sygnałów. Pomimo dużego zasięgu (do 4 m) sensowne wyniki są uzyskiwane przy maksymalnej odległo-

ści około 1,5 m. Prawdopodobnie jest to związane tym, że sonar znajduje się kilkanaście cm nad ziemią. Nie powoduje to jednak jakichkolwiek problemów w funkcjonowaniu robota, gdyż zwykle analizuje się obiekty znajdujące się w mniejszej odległości.

Klawiatura PS/2

Robotem można sterować z użyciem klawiatury PS/2. Do kontroli prototypu jest używana klawiatura numeryczna PS/2. Aktualnie pracuję również nad sterowaniem z użyciem nadajnika podczerwieni (sterowanie z użyciem kodów RC5).

Programowanie

Jak wspominałem, programowanie robota odbywa się z użyciem środowiska *Arduino*. Środowisko to zaprojektowano w taki sposób, aby było przyjazne dla hobbystów. Programowanie odbywa się w języku C, jednak nie jest trudne. A wszystko dlatego, że *Arduino* zawiera wiele gotowych bibliotek, które pozwalają na bardzo szybkie i proste napisanie programu. Po napisaniu programu cały proces przesłania kodu wynikowego do płytki mikrokomputera przez interfejs USB jest wykonywany automatycznie. Ponadto, na stronie internetowej *arduino.cc* są bardzo dokładnie opisane różne przykłady. Bardzo pomocne jest też forum użytkowników ma-

jące kilkadziesiąt tysięcy wpisów i przykładów.

Aktualnie oprogramowanie robota umożliwia mu pracę w czterech trybach:

- Tryb pracy ręcznej: sterowanie robotem z użyciem klawiatury numerycznej PS/2. Operator robota ma możliwość wykonania następujących ruchów: krok w przód/w tył; zwrot w lewo/w prawo; robot przechodzi w tryb „uśpienia” (podwija kończyny i osiada na podłożu); robot wstaje; robot podnosi się na nogach/opada do pozycji podstawowej; „rozcłąda się” w lewo i w prawo poprzez obrót sonaru i częściowo ciała; robot kroczy w miejscu, losowo wybierając nogę.
- Tryb pracy automatycznej: robot idzie przed siebie, rozglądając się po każdym kroku w lewo i w prawo, a następnie podejmuje decyzję co do dalszego ruchu. Jeśli odległość do przeszkody jest mniejsza niż graniczna, robot skręca w kierunku, gdzie odczyt był większy i następnie idzie dalej.
- Tryb pracy demonstracyjnej: robot wykonuje losowe ruchy. Tryb ten został zaprogramowany na potrzeby zawodów, aby móc zaprezentować ciągłą pracę robota, a zarazem nie szukać go po okolicy. Robot w zasadzie po dwóch sekwencjach ruchu stoi w tym samym miejscu.
- Tryb pracy demonstracyjnej sonaru: w tym trybie robot wykonuje unik przed zbliżającą się przeszkodą. W chwili osiągnięcia granicznej dla robota odległości przeszkody podążającej w jego kierunku ten robot cofa się o jeden krok.

Odpowiednie ustawienia serwomechanizmów, a co za tym idzie – poszczególnych kroków robota, poza niektórymi ruchami (jak tryb demonstracyjny sonaru), odbywa się na zasadzie sczytywania odpowiednich wartości wychyleń z tablic zawartych w programie. Wychylenia te dobrałem empirycznie. Program zajmuje około 29 kB pamięci, a więc może być jeszcze znacznie rozbudowany.

Podsumowanie

Robot całkowicie spełnił moje oczekiwania, jak również dostarczył potężnej dawki wiedzy z zakresu robotyki, elektroniki oraz programowania. A przy okazji jest ciekawym przykładem funkcjonalności, którą można zrealizować już za pierwszym razem, tzn. budując swój pierwszy projekt. Wcześniej nigdy nie miałem styczności z robotyką. Tylko własne obserwacje i przemyślenia doprowadziły mnie do konstrukcji platformy, praktycznie nie mającej ograniczeń co do dalszej rozbudowy o kolejne czujniki czy układy wykonawcze.

Marcin Sowa
marcin_sowa@o2.pl