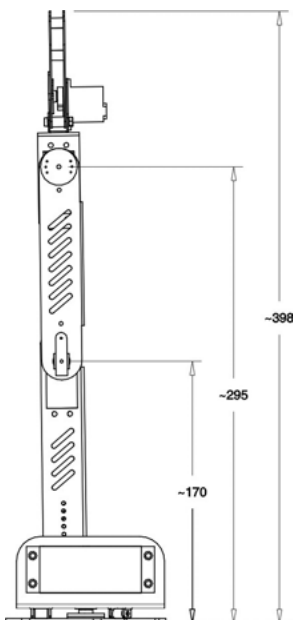


Manipulator Mobot-ARM1

Manipulator to ramię mechaniczne używane głównie na automatycznych liniach produkcyjnych. Na przestrzeni ostatnich lat manipulatory uległy miniaturyzacji i znalazły także wiele innych zastosowań. Przykładowo, coraz częściej montowane są w robotach używanych przez wojsko czy policję jako narzędzia do zdalnego unieszkodliwiania ładunków wybuchowych lub do interweniowania w sytuacjach, które mogłyby zagrozić zdrowiu lub życiu.

Do napędu manipulatora o skromniejszych niż omawiane we wstępie możliwościach, można zastosować popularne serwomechanizmy modelarskie. Opisujący w artykule manipulator **Mobot-ARM1** zbudowano z myślą o pasjonatach robotyki, którzy mogą go zastosować wprost lub wyposażać w dodatkowe czujniki. Na przykład można zainstalować czujnik nacisku umożliwiający określenie siły chwytu ramion, czy czujniki odległości pozwalające na zorientowanie ramienia czy też wyposażonego w nie robota w przestrzeni jego pracy. Można również zamontować manipulator na pojeździe, co pozwoli wykonać robota podobnego konstrukcyjnie do tych używanych przez policję i zamontować na ramieniu kamerę, która pozwoli na dokładne „przyjrzenie się” unoszonemu przedmiotom.

Zaawansowani programiści wykorzystując obraz dostarczany przez kamerę mogą wykonać system wizyjny służący do sterowania ruchem manipulatora i np. przenoszenia obiektów wykrytych przez kamerę.



Rysunek 1. Wymiary mechaniczne manipulatora

Elastyczność konstrukcji manipulatora daje wiele możliwości indywidualnego zastosowania go w celach rozrywkowych, edukacyjnych, a nawet praktycznych. Ograniczeniem są jedynie wyobrażenia i umiejętności konstruktora.

Manipulator Mobot-ARM1

Manipulator Mobot-ARM1 ma cztery stopnie swobody i chwytak umożliwiające wygodne manipulowanie przedmiotami znajdującymi się w przestrzeni roboczej. Pieczę nad pracą poszczególnych serwomechanizmów sprawuje sterownik z mikrokontrolerem STM32. Układ ten ma wbudowane timery z wyjściami typu *Output Compare*, które po załączeniu odpowiedniego trybu pracy idealnie nadają się do sterowania serwomechanizmami modelarskimi.

Mechanika manipulatora

Manipulator wykonano w całości z elementów blaszanych, wycinanych laserowo i odpowiednio wyginanych. Kompletna konstrukcja mechaniczna przeznaczona do samodzielnego montażu jest dostępna poprzez stronę internetową www.mobot.pl. Wymiary mechaniczne ramienia manipulatora pokazano na **rysunku 1**.

Podstawę zbudowano z elementów o grubości 3 mm oraz 1,5 mm. Jej konstrukcję wykonano w taki

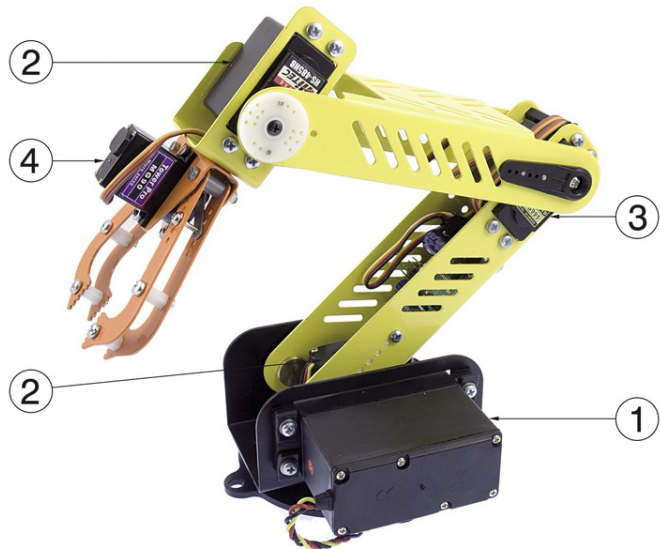


sposób, aby zapewnić sztywne podparcie członu obrotowego oraz ramienia.

Sterownik serwomechanizmów

Od strony sterownika, kontrolowanie serwonapędów modelarskich polega na generowaniu co około 20 ms impulsów o szerokości zmiennej w zakresie 1...2 ms. Najdłuższemu i najkrótszemu czasowi trwania impulsów odpowiadają skrajne wychylenia serwomechanizmu.

Jak wspomniano wcześniej, sterownik serwomechanizmów wykonano w oparciu o mikrokontroler **STM32F103CBT6**. Układ ten ma wbudowane 7 timerów, z których do sterowania serwomechanizmami użyto dwóch. Mają one 4 niezależne kanały wyjściowe, które skonfigurowano jako wyjścia typu *Output Compare* i użyto do sterowania serwami. Dzięki temu przy wykorzystaniu



Rysunek 2. Rozmieszczenie serwomechanizmów napędzających manipulator. Pierwsza część ramienia manipulatora, z racji przenieszonego największego obciążenia (zasada dźwigni), jest napędzana przez serwomechanizm o największym momencie obrotowym [1] (np. Hitec HS-815BB). Pozostałe elementy ramienia manipulatora oraz jego obrót napędzane są przez serwomechanizmy standardowe [2, 3, 4] (np. Hitec HS-485HB). Chwytak, ze względu na wymagania odnośnie do ciężaru, jest napędzany przez serwomechanizm miniaturowy [5] (np. Tower Pro MG90).

Listing 1 – konfiguracja wyjść OC do generowania impulsów sterujących serwami

```
// Włączenie zegarów timerów
RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM2 | RCC_APB1Periph_TIM3, ENABLE);
// Ustalenie okresu pracy timera (36000000/59999/11 = 54)
TIM_TimeBaseStructure.TIM_Period = 59999;
TIM_TimeBaseStructure.TIM_Prescaler = 11;
TIM_TimeBaseStructure.TIM_ClockDivision = 0;
TIM_TimeBaseStructure.TIM_CounterMode = TIM_CounterMode_Up;
TIM_TimeBaseInit(TIM2, &TIM_TimeBaseStructure);
TIM_TimeBaseInit(TIM3, &TIM_TimeBaseStructure);
// Konfiguracja wyjść OC jako PWM w trybie 1
TIM_OCInitStructure.TIM_OCMode = TIM_OCMode_PWM1;
TIM_OCInitStructure.TIM_OutputState = TIM_OutputState_Enable;
TIM_OCInitStructure.TIM_OCPolarity = TIM_OCPolarity_High;
// Ustawienie początkowej wartości wypełnienia rejestrów porównań
// gdzie: START_POS = 1000 (1ms), US_TO_IMP = 6
TIM_OCInitStructure.TIM_Pulse = START_POS * US_TO_IMP;
// Inicjalizacja wyjść OC
TIM_OC1Init(TIM2, &TIM_OCInitStructure);
TIM_OC1Init(TIM3, &TIM_OCInitStructure);
TIM_OC2Init(TIM2, &TIM_OCInitStructure);
TIM_OC2Init(TIM3, &TIM_OCInitStructure);
TIM_OC3Init(TIM2, &TIM_OCInitStructure);
TIM_OC3Init(TIM3, &TIM_OCInitStructure);
TIM_OC4Init(TIM2, &TIM_OCInitStructure);
TIM_OC4Init(TIM3, &TIM_OCInitStructure);
TIM_ARRPreloadConfig(TIM2, ENABLE);
TIM_ARRPreloadConfig(TIM3, ENABLE);

// Timer 3 - slave, Timer 2 - master
TIM_SelectMasterSlaveMode(TIM2, TIM_MasterSlaveMode_Enable);
TIM_SelectOutputTrigger(TIM2, TIM_TRGOSource_Update);
TIM_SelectSlaveMode(TIM3, TIM_SlaveMode_Reset);
TIM_SelectInputTrigger(TIM3, TIM_TS_ITR1);
TIM_Cmd(TIM2, ENABLE);
TIM_Cmd(TIM3, ENABLE);
```

tylko dwóch timerów istnieje możliwość sterowania aż 8 serwomechanizmami.

Przebiegi wyjściowe z mikrokontrolera są buforowane w celu dopasowania poziomów napięcia oraz zabezpieczenia mikrokontrolera przed ewentualnymi przepięciami i zwarciami.

Kontroler wyposażono w dwa interfejsy komunikacyjne: UART oraz USB. Służą one do sterowania poszczególnymi serwonapiędaniami z urządzenia zewnętrznego, na przykład komputera PC.

Na płycie PCB przewidziano złącza umożliwiające bezpośrednie dołączenie do 8 serwomechanizmów modelarskich. Do zasilania całości należy użyć zasilacza lub pakietu akumulatorów dostarczających napięcie 4,8...6 V, o wydajności prądowej co najmniej 2 A.



Rysunek 3. Miejsce zamocowania płytki kontrolera serwomechanizmów

Oprogramowanie mikrokontrolera

Na **listingu 1** pokazano fragment programu sterującego pracą mikrokontrolera odpowiedzialny za konfigurowanie timerów. Należy zauważyć, że Timer 3 został skonfigurowany do pracy jako *slave* i jest wyzwalany przez Timer 2. Ma to na celu synchroniczną pracę obu timerów i regulację serw w tym samym czasie.

Zmiana szerokości impulsu sterującego serwomechanizmem powoduje natychmiastową zmianę jego położenia. Czasami jednak zależy nam na powolnym, a co za tym idzie – precyzyjnym, „dojściu” do jakiejś pozycji. W tym celu użyto Timera 4, który generuje przerwania w stałych odstępach czasu. W obsłudze jego przerwania jest stopniowo zmniejszana bądź zwiększana pozycja danego serwomechanizmu, aż do osiągnięcia zadanej.

Komunikacja poprzez UART oraz USB

Do komunikacji ze sterownikiem zastosowano interfejs UART. Prosta ramka sterująca pozwala na zwiększanie bądź zmniejszenie kąta wychylenia poszczególnego serw. Interfejs USB umożliwia bezpośrednie połączenie z komputerem PC. Ramię jest widziane jako urządzenie klasy HID i dzięki temu nie jest konieczna instalacja żadnych dodatkowych sterowników. Dla komputera PC napisano oprogramowanie komunikujące się z urządzeniem HID i pozwalające na niezależne sterowanie prędkością oraz pozycją każdego z serw.

Źródła oprogramowania (wersja nieobsługująca USB), plik wynikowy HEX dla wersji obsługującej USB, oprogramowanie na PC oraz rysunki mechaniczne są dostępne na stronie www.mobot.pl.

Adam Sarzyński
Koordynator Działu Badań i Rozwoju
P.P.H. WObit E.K.J. Ober s.c.

Kinco

Panele HMI

już od 382 zł netto



Sterowniki PLC

już od 389 zł netto



Moduły rozszerzeń I/O

już od 147 zł netto



Kinco®

www.kinco.com.pl

Wylączny dystrybutor w Polsce:
P.P.H. WObit E.K.J. Ober S.C.
tel. +48 61 2912 225
wobit@wobit.com.pl