



**WYDZIAŁ
ELEKTROTECHNIKI
I INFORMATYKI**
POLITECHNIKI RZESZOWSKIEJ

Kacper Rychel

Robot mobilny z nawigacją autonomiczną oparty na Arduino

PRACA INŻYNIERSKA

Opiekun pracy:

Dr inż. Mariusz Mączka

Rzeszów, 2025

Spis treści

Wykaz symboli, oznaczeń i skrótów (opcjonalny)	5
1. Wprowadzenie	6
2. Tekst zasadniczy – I	9
2.1. Formatowanie rozdziałów i podrozdziałów	9
3. Tekst zasadniczy – II	10
3.1. Formatowanie tekstu	10
3.1.1. Marginesy i akapity	10
3.1.2. Zalecenia co do sposobu pisania jednostek i symboli wielkości fizycznych	11
3.1.3. Rysunki i tabele	13
3.1.4. Wzory matematyczne	14
3.1.5. Listingi programów	15
3.1.6. Numerowanie i punktowanie	16
3.1.7. Uwagi końcowe dla użytkowników L ^A T _E X	17
3.2. Wykaz literatury	17
3.3. Wydruk pracy	17
4. Podsumowanie i wnioski końcowe	19
Załączniki	20
Literatura	21

Wykaz symboli, oznaczeń i skrótów (opcjonalny)

1 ÷ 2 stron wykaz ważniejszych symboli i oznaczeń (jeśli jest potrzebny).

1. Wprowadzenie

Roboty mobilne stanowią obecnie jeden z najintensywniej rozwijających się obszarów robotyki oraz inżynierii mechatronicznej. Ich rosnące znaczenie wynika z szerokiego spektrum zastosowań obejmujących przemysł, logistykę, eksplorację środowisk niebezpiecznych, systemy autonomiczne oraz edukację techniczną. Zgodnie z definicją Międzynarodowej Federacji Robotyki robot mobilny jest systemem mechatronicznym zdolnym do samodzielnego przemieszczania się w przestrzeni oraz realizacji określonych zadań bez stałego połączenia z infrastrukturą stacjonarną [1]. Współczesny rozwój technologii mikroprocesorowych oraz dostępność platform open-source znacząco ułatwiły projektowanie i implementację tego typu systemów. Jak zauważa Monk, popularyzacja mikrokontrolerów, takich jak Arduino, umożliwiła szybkie prototypowanie rozwiązań robotycznych nawet w warunkach dydaktycznych i amatorskich [2].

Kluczowym zagadnieniem w robotyce mobilnej jest autonomia, rozumiana jako zdolność robota do samodzielnego podejmowania decyzji na podstawie danych pochodzących z czujników. Autonomiczne systemy mobilne muszą integrować warstwy percepcji, planowania oraz sterowania ruchem. Według Siegwart’a, Nourbakhsha i Scaramuzzy autonomia robota opiera się na ścisłej współpracy tych trzech elementów, nawet w przypadku prostych algorytmów reaktywnych [3]. W konstrukcjach edukacyjnych i prototypowych stosuje się zazwyczaj uproszczone algorytmy decyzyjne, których celem jest zapewnienie podstawowych funkcji, takich jak unikanie przeszkód czy poruszanie się w nieznanym środowisku.

Istotnym elementem systemów autonomicznych są czujniki umożliwiające detekcję otoczenia. W robotach niskokosztowych powszechnie stosowane są czujniki ultradźwiękowe, które wykorzystują zjawisko odbicia fali akustycznej do pomiaru odległości od przeszkody. Czujnik HC-SR04, często wykorzystywany w projektach edukacyjnych, umożliwia pomiar odległości w zakresie od kilku do kilkuset centymetrów przy stosunkowo niewielkim koszcie oraz prostej integracji z mikrokontrolerem [4]. Zgodnie z dokumentacją techniczną producenta, dokładność pomiaru jest wystarczająca dla podstawowych algorytmów nawigacyjnych, choć należy uwzględnić podatność czujnika na zakłócenia wynikające z charakterystyki powierzchni odbijających falę ultradźwiękową [5].

Ruch robota mobilnego realizowany jest najczęściej przy wykorzystaniu napędu różnicowego, składającego się z dwóch niezależnie sterowanych silników prądu stałego. Taki układ umożliwia zmianę kierunku jazdy oraz skręcanie poprzez regulację prędkości obrotowej poszczególnych kół. Jak wskazują Jones, Flynn i Seiger, napęd różnicowy jest jednym z najprostszych i najbardziej rozpowszechnionych rozwiązań w robotach mobilnych ze względu na niewielką złożoność mechaniczną i sterowniczą [6]. Do sterowania silnikami prądu stałego stosuje się układy typu mostek H, takie jak L293D, które pozwalają na zmianę kierunku obrotów oraz regulację prędkości za pomocą sygnału modulowanego szerokością impulsu PWM. Horowitz i Hill podkreślają, że mostki H stanowią podstawowy element układów wykonawczych w systemach sterowania napędem o małej mocy [7].

Centralnym elementem sterującym robota mobilnego jest mikrokontroler, który odpowiada za przetwarzanie danych sensorycznych oraz generowanie sygnałów sterujących elementami wykonawczymi. Platforma Arduino UNO, oparta na mikrokontrolerze ATmega328P, jest jedną z najczęściej wykorzystywanych w projektach edukacyjnych i badawczych ze względu na dostępność dokumentacji, bibliotek programistycznych oraz wsparcie społeczności open-source [2][8]. Zastosowanie gotowych bibliotek umożliwia szybkie wdrażanie algorytmów sterowania ruchem oraz obsługi czujników bez konieczności implementowania niskopoziomowych mechanizmów sprzętowych.

Projekty polegające na budowie autonomicznych robotów mobilnych mają istotne znaczenie dydaktyczne, gdyż umożliwiają integrację wiedzy z zakresu elektroniki, programowania, automatyki oraz mechaniki w jednym, spójnym systemie. Jak zauważa Corke, robotyka stanowi naturalne środowisko do kształcenia kompetencji inżynierskich, łącząc teorię z praktycznym rozwiązywaniem rzeczywistych problemów technicznych [9]. Z tego względu realizacja projektu autonomicznego robota mobilnego stanowi wartościowe zagadnienie pracy dyplomowej o charakterze inżynierskim.

Rozwój autonomicznych robotów mobilnych oraz ich rosnące znaczenie w edukacji i przemyśle stanowi bezpośrednie uzasadnienie wyboru tematyki pracy. Celem niniejszej pracy jest zaprojektowanie i wykonanie autonomicznego robota mobilnego opartego na mikrokontrolerze Arduino UNO, zdolnego do samodzielnego poruszania się oraz unikania przeszkód na podstawie pomiarów odległości. Zakres pracy obejmuje analizę teoretyczną zagadnień związanych z robotyką mobilną, projekt i realizację

układu sprzętowego, implementację algorytmu sterowania oraz przeprowadzenie testów funkcjonalnych opracowanego rozwiązania.

2. Tekst zasadniczy – I

Do 20% objętości pracy. W zależności od charakteru pracy ten rozdział powinien zawierać:

- a) opis tematyki zagadnienia – aktualny stan zagadnienia,
- b) metody i rozwiązania,
- c) dyskusja i krytyczna ocena stanu aktualnego,
- d) podsumowanie stanu wiedzy, techniki literaturowe itp.

2.1. Formatowanie rozdziałów i podrozdziałów

Rozdziały zaczynają się u góry nowej strony (parzystej lub nieparzystej). Podrozdziały i zakresy mogą zaczynać się w dowolnym miejscu strony. Przy końcu pracy zamieszcza się podsumowanie i wnioski. Ostatni akapit podsumowania musi zawierać wyszczególnienie własnej pracy autora i zaczynać się od sformułowania: „Autor za własny wkład pracy uważa:”. W tym miejscu kończy się numeracja rozdziałów.

Ewentualne listingi programów, instrukcje obsługi stanowisk lub inne tego rodzaju materiały zaleca się zamieścić w formie dodatków. Kolejno zamieszcza się: wykaz literatury, spis rysunków/tabel oraz streszczenie (zgodne ze „Wzorem streszczenia”). Wykaz literatury rozpoczyna od strony nieparzystej.

Opisując własne dokonania, stosuje się formę bezosobową w czasie przeszłym np. celem pracy było zaprojektowanie..., zakres pracy obejmował wyznaczenie..., w ramach pracy wykonano model... itp.

3. Tekst zasadniczy – II

Ponad 50% objętości pracy – część autorska:

- a) założenia – dane,
- b) opis zastosowanej metody rozwiązania lub analizy,
- c) opis proponowanego rozwiązania, wyniki analizy teoretycznej, obliczenia, projekt konstrukcyjny, procesowy, technologiczny,
- d) wyniki badań analitycznych, symulacyjnych lub eksperymentalnych itp.

Przy stosowaniu podziału na rozdziały i podrozdziały zaleca się unikać podziału więcej niż trzystopniowego. Podział tekstu, szczególnie na rozdziały główne, wynikać powinien z zakresu i charakterystyki realizowanej pracy.

3.1. Formatowanie tekstu

Należy pamiętać, że na końcu tytułu rozdziału, podrozdziału i zakresu nie umieszcza się kropki. Podobnie pierwszy wiersz po tytule rozdziału, podrozdziału czy zakresu nie ma wcięcia poziomego, ponieważ już został dodany odstęp pionowy.

3.1.1. Marginesy i akapity

Marginesy deklaruje się jako „lustrzane” i ustawia na 2 cm plus na oprawę 1,5 cm. Nagłówek i stopka 1,25 cm. Tekst podstawowy akapitu: czcionka szeryfowa, styl Times (Times New Roman, Liberation Serif itp.), rozmiar 12 punktów, interlinia 1,5 wiersza. Akapity wyjustowane, wcięcie pierwszego wiersza w kolejnych akapitach to 1,25 cm.

Na końcu każdego akapitu, którego tekst zaczerpnięto z literatury, musi znajdować się odnośnik do właściwej pozycji w wykazie literatury. W pracy nie stosuje się odnośników w formie przypisów. Liczby w nawiasie kwadratowym oznaczają kolejny numer pozycji w wykazie, np. [1] lub [1, 4, 7] lub [1, 6-8] itp. Efekt ten można uzyskać stosując polecenie `\cite` z parametrem odwołującym się do pozycji bibliografii oraz stylem bibliografii ustawionym na `plain`.

Cytaty (dosłowne przytoczenie obcego tekstu w pracy) pisze się czcionką pochylą (kursywą) albo ujmuje w cudzysłów (nigdy oba naraz). Przykład: *Współpracując z jednostkami gospodarczymi działającymi w kraju, kształci wysokokwalifikowaną kadrę inżynierów*, albo „Współpracując z jednostkami gospodarczymi działającymi w kraju, kształci wysokokwalifikowaną kadrę inżynierów”.

Fragmenty kodów programów pisze się czcionką o stałej szerokości, styl *Courier* (Courier New, Liberation Mono itp.) o rozmiarze 10 punktów.

3.1.2. Zalecenia co do sposobu pisania jednostek i symboli wielkości fizycznych

Poniższy podrozdział opracowano na podstawie [5]. W trakcie pisania pracy należy zwracać uwagę na sposób oznaczania jednostek i symboli wielkości fizycznych. Przy zapisywaniu jednostek i symboli wielkości fizycznych można wyróżnić zapis w postaci kursywy (pismo pochyłe) oraz antykwy¹ (pismo proste).

1) Kursywę należy stosować w następujących przypadkach:

- symboli wielkości fizycznych niezależnie od tego czy jest to litera alfabetu greckiego (np. przenikalność magnetyczna μ) czy też łacińskiego (np. rezystancja R). Należy przestrzegać tej zasady niezależnie od miejsca, w którym pojawia się symbol tj. tekst, wzory matematyczne, rysunki, tabele,
- ogólny symbol zapisu funkcji czyli np. f , a nie f . Nie dotyczy to jednak zapisu konkretnych funkcji np. $\cos\omega t$ a nie $\cos\omega t$,
- macierze, wektory, których elementami są wielkości fizyczne należy zapisywać dodatkowo czcionką półgrubą (bold) np. $\mathbf{R} = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}$, $\mathbf{U} = \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}$,
- wskaźnik dolny, górny, prawo- i lewostronny, ale tylko gdy odnosi się do konkretnej wielkości fizycznej, czyli np. składowa x -owa indukcji magnetycznej B_x , a nie B_x ,
- wskaźniki górne i dolne oznaczające dowolną liczbę np. R_j , I^k , ale nie R_1 , I^2 .

2) Czcionkę prostą należy stosować w następujących sytuacjach:

- wszystkie cyfry,
- symbole konkretnych funkcji np. $\tan \omega t$, a nie $\tan \omega t$,
- operatory operacji matematycznych np. pochodne zwyczajne $\frac{dx}{dt}$, a nie $\frac{dx}{dt}$,
- symbole liczb o konkretnej wartości np. przenikalność elektryczna próżni $\varepsilon_0 = 8,8542 \cdot 10^{-12} \text{ F} \cdot \text{m}^{-1}$, a nie $\varepsilon_0 = 8,8542 \cdot 10^{-12} \text{ F} \cdot \text{m}^{-1}$,
- indeksy, jeżeli odnoszą się do: obiektów (fizycznych, geometrycznych), czyli, np. natężenie pola elektrycznego w punkcie A to E_A , a nie E_A , zjawisk lub

¹Potoczna nazwa pisma prostego.

stanów fizycznych, np. moment obciążenia to T_L , a nie T_L , do nazwisk czy też oznaczeń pierwiastków, np. straty w miedzi to P_{Cu} a nie P_{Cu} , do charakteru wielkości symbolizowanej przez literę źródłową, np. wartość maksymalna siły to F_{\max} , a nie F_{max} , oznaczeń jednostek miary np. $M\Omega$, a nie $M\Omega$.

3) W przypadku jednostek miar (które zawsze należy pisać krojem prostym) zapisując konkretną wartość liczbą należy podać jej wartość i jednostkę z zachowaniem następujących zasad:

- zapisując wartość liczbową wielkości fizycznej po spacji należy podać jej jednostkę, ale nie nazwę jednostki np. 10 A, ale nie 10 amper czy też 10 amperów,
- zapisując wartość liczbową słownie należy w tej konwencji podać też jednostkę np. dziesięć omów, ale nie dziesięć Ω
- do oznaczeń jednostek nie wolno dopisywać indeksów, np. moc wyjściowa silnika wynosi $P = 100 \text{ kW}_{\text{out}}$. W takim przypadku należy zapisać $P_{\text{out}} = 100 \text{ kW}$,
- jednostek nie należy umieszczać w nawiasach kwadratowych, np. $I = 1 \text{ [A]}$. Odstępstwem od tej zasady mogą być tabele, nagłówki kolumn, opisy osi na wykresach oraz w sporadycznych sytuacjach we wzorach matematycznych (ale tylko wówczas, gdy zależność matematyczna nie wskazuje w jakiej jednostce wystąpi wartość liczbowa). Przykłady odstępstw zamieszczono w podrozdziale 3.1.3.

4) W trakcie zapisu symboli wielkości matematycznych można stosować również szereg znaków diakrytycznych, jak również należy przestrzegać następujących zaleceń:

- wartości chwilowe podstawowych wielkości fizycznych używanych np. w elektrotechnice należy zapisać małymi literami, np. u , i , lub stosować zapis np. $u(t)$, lub stosować indeks „ t ” przy wielkości, np. U_t ,
- wartości skuteczne wielkości okresowych należy zapisać dużą literą np. U , I ,
- wartości szczytowe funkcji zmiennej, amplitudę funkcji sinusoidalnej czasu należy zapisać jako np. U_m ,

- podkreślenie symboli reprezentujących wielkości fizyczne, których wartość liczbową jest liczbą zespoloną, przy czym podkreślenie dotyczy tylko literki źródłowej np. \underline{Z}_1 , a nie \underline{Z}_1 ,
- kreska nad literą źródłową oznacza wartość średnią, np. \bar{I} co jest równoważne I_{av} .

3.1.3. Rysunki i tabele

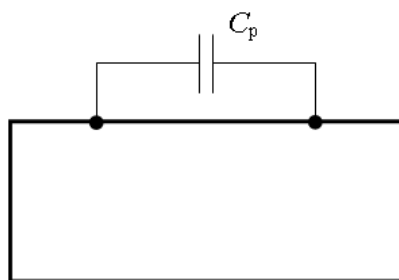
Tekst podstawowy w tabeli pisze się czcionką o rozmiarze 10 punktów, pojedyncza interlinia. Dane liczbowe – wyśrodkowane, dane tekstowe – wyrównane do lewej. Rysunki i tabele zamieszcza się wyśrodkowane na stronie, bez wcięcia pierwszego wiersza.

W akapicie poprzedzającym rysunek lub tabelę musi znajdować się krótki opis, czego dotyczy dany rysunek/tabela (odniesienie do rysunku/tabeli). Tytuły numeruje się zgodnie z kolejnością w danym rozdziale: numer_rozdziału.numer_tabeli/rysunku (np. rys. 2.1, tabela 3.5). W tytule rysunku/tabeli, zaczerpniętych z literatury, podaje się odnośnik do właściwej pozycji. Należy zadbać o to, aby opisy na rysunkach były czytelne (czcionka 8 punktów lub większa). Staraj się nie wymuszać numeracji, pozwól aby robił to za ciebie L^AT_EX. Stosuj `\label` do znakowania obiektów, do których być może w tekście się będziesz odwoływał (rozdziały, rysunki, tabele, wzory, listingi ...). Odwołuj się do nich w tekście za pomocą funkcji `\ref{NazwaObiektu}`. Pamiętaj, że L^AT_EX, korzystając z polecenia `latex` nie odczytuje z plików .jpg, .png ich wielkości. Polecenie `latex` generuje plik DVI. Jeżeli chcesz go używać zgłosi stosowny błąd. Aby się go pozbyć zdefiniuj wielkość natywną pliku grafiki. Polecamy jednak używanie zamiast polecenia `latex`, polecenie `pdflatex`, wówczas problem nie wystąpi.

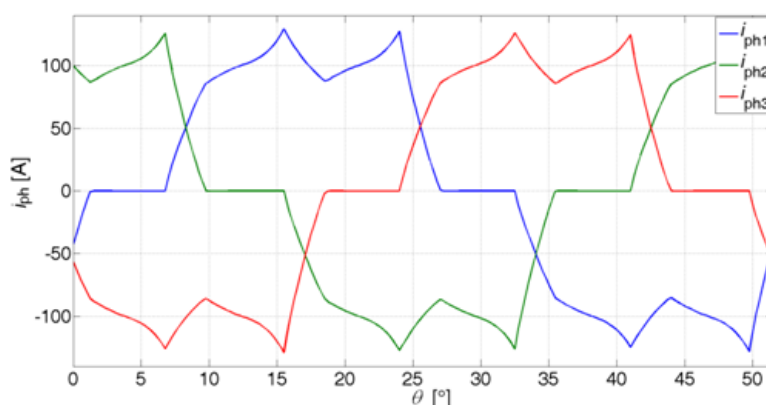
Przykład: [...] co umożliwi wyznaczenie wartości napięcia. Na rys. 3.1 przedstawiono schemat obwodu z równolegle dołączoną pojemnością C_p .

Przykład: [...] Na rysunku 3.2 pokazano przykładową zależność prądów pasmowych i_{ph} bezszczotkowego silnika prądu stałego z magnesami trwałymi w funkcji położenia wirnika θ .

Przykład: [...] oraz indukcyjności wzajemnej. W tabeli 3.1 przedstawiono podstawowe parametry obwodu nieliniowego, zasilanego napięciem trójfazowym.



Rys. 3.1. Tytuł rysunku, rozmiar 11 pkt, pojedyncza interlinia, akapit wyśrodkowany, bez wcięcia pierwszego wiersza. Na końcu tytułu rysunku/tabeli nie stawia się kropki [8]



Rys. 3.2. Tytuł rysunku, rozmiar 11 pkt, pojedyncza interlinia, akapit wyśrodkowany, bez wcięcia pierwszego wiersza. Na końcu tytułu rysunku/tabeli nie stawia się kropki [8]

3.1.4. Wzory matematyczne

Zmienne we wzorach pisze się czcionką pochyłą (styl edytora równań „Matematyka”) natomiast symbole, niebędące zmiennymi, czcionką prostą (styl „Tekst”). Rozmiary czcionek: normalny 12 punktów, indeks dolny/górny 9 pkt, indeks podrzędny 7 pkt, symbol 24 pkt, podsymbol 12 pkt Separatorem dziesiętnym w liczbach jest przecinek, a nie kropka (dotyczy to również liczb pisanych w tekście akapitu). Poddawaj się w tym zakresie \LaTeX owi – pisz wzór, a poprawnie się utworzy.

Pod wzorem należy zamieścić objaśnienia użytych symboli (chyba, że znajdują się w wykazie na początku pracy). Wzory umieszcza się wyśrodkowane i numeruje zgodnie z kolejnością w danym rozdziale: (numer_rozdziału.numer_wzoru). Numery wzorów wyrównuje się do prawego marginesu. W akapicie poprzedzającym wzór musi znajdować się krótki opis, czego dotyczy dany wzór i – jeżeli potrzeba – odwołanie do literatury.

Tabela 3.1. Tytuł tabeli, rozmiar 11 pkt, pojedyncza interlinia, akapit wyrównany do lewej

U [V]	I [mA]	R , [k Ω]	L [mH]	R/R_{20}
13,6	7,29	3,94	100	1,25

Przykład: [...] wyznacza się, na podstawie wyrażenia (3.1). W nawiasach podano rozmiary czcionek używanych we wzorach

$$A(12) = \sum (24)m_{s(9)}N^{k_p(7)} \quad (3.1)$$

gdzie: m_s – masa próbki, N – natężenie oświetlenia, k_p – wykładnik potęgi ($k_p = 1,3 - 2,1$).

3.1.5. Listingi programów

W pracy dyplomowej możesz umieszczać fragmenty programów. Pamiętaj, aby umieszczać krótkie, tylko najważniejsze fragmenty kodów źródłowych. Zawsze je komentuj w treści pracy dyplomowej. Typowo w L^AT_EX kody źródłowe umieszczane są w środowisku verbatim (`\begin{verbatim}... \end{verbatim}`). Obecnie istnieje jednak bardziej nowoczesne i bardziej funkcjonalne środowisko `lstlisting` (wymaga zainstalowanego w systemie pakietu `listings`). Zwróć uwagę, że możesz kolorować składnię automatycznie za pomocą parametru `language`. W niniejszym dokumencie przedstawiono trzy przykłady listingów – listing 3.1 to przykład kodu źródłowego Matlaba, poniżej listing 3.2 dla Perla i dołączony z zewnętrznego pliku listing języka C (3.3).

Listing 3.1. Listing programu Matlab

```

1 i = 1
2 p = 3
3 for i = 1:10
4     if i > 3
5         i=i+p
6     else
7         i=i+1
8     end
9 end

```

Listing 3.2. Listing programu Perl

```

1 my $url = 'http://pei.prz.edu.pl';
2 use LWP::Simple;
3 my $content = get $url;
4 die "Couldn't get $url" unless defined $content;
5 print $content;
6 print "\n";
7 print "Length " + length($content)

```

Z pewnością przeglądając źródło tego dokumentu zobaczysz, że kody źródłowe powinny mieć zdefiniowane parametry `label`, aby łatwo w tekście do nich się odwoływać. Numeracja linii jest w stylu domyślnie włączona (to przydatne, bo w treści pracy łatwo odwołać się dzięki temu do konkretnego wiersza w kodzie źródłowym), możesz je wyłączyć podając jako parametr `numbers=none`. Więcej szczegółów możesz odnaleźć w sekcji `\lstset` pliku arkusza stylów.

Można również dołączać listingi z zewnętrznych plików², **zwłaszcza jeżeli zawierają polskie znaki diakrytyczne**. Można w tym celu wykorzystać polecenie `lstinputlisting`, które przekoduje znaki na te z dostępnej czcionki:

```
\lstinputlisting[language={ANSI}C,inputencoding=utf8/cp1250,
firstline=2,lastline=6,breaklines=true,numbers=left,frame=none,
caption={Polskie znaki diakrytyczne w kodzie},label={lst:plchr}]
{auxiliary/start.c}
```

Listing 3.3. Polskie znaki diakrytyczne w kodzie

```
1 #include <stdio.h>
2
3 int main()
4 {
5     printf("Witaj świecie listingów!\n");
```

Można zauważyć, że oryginalny plik `start.c` z katalogu `auxiliary` zawiera więcej linii, a tutaj zamieszczono tylko wybrane (od 2 do 6). Można też zmienić im numerację linii parametrem `firstnumber`, np. `firstnumber=5`.

3.1.6. Numerowanie i punktowanie

- 1) Pierwszy poziom (stosuje się numerowanie lub punktowanie). Formatowanie: akapit wyjustowany, wcięcie od lewej 0,75 cm, wysunięcie co 0,5 cm.
- 2) Znakiem numerowania jest liczba (z kropką lub nawiasem).
 - drugi poziom (stosuje się wyłącznie punktowanie). Formatowanie: akapit wyjustowany, wcięcie od lewej 1,25 cm, wysunięcie co 0,5 cm,
 - znakiem punktowania jest łącznik lub mała litera alfabetu (z nawiasem). Nie zaleca się stosowania kropek, strzałek itp.,
 - punktowane akapity rozpoczyna się minuskulą (małą literą), na końcu akapitu stawia się przecinek, ostatni punktowany akapit kończy się kropką.

²i nadać im taki sam (albo inny) wygląd jak ze środowiska `lstlisting`.

- 3) Numerowane akapity rozpoczyna się majuskułą (wielką literą) i kończy kropką.
- 4) Należy zwrócić uwagę, aby nie rozdzielać numerowania/punktowania pomiędzy kolejnymi stronami tekstu.

3.1.7. Uwagi końcowe dla użytkowników L^AT_EX

Kompilację dokumentu można przyspieszyć dodając opcjonalny parametr `draft` do polecenia `\documentclass`. Wówczas rysunki nie będą zamieszczane, a w ich miejscu pojawi się prostokąt z obrysem. Ponadto tryb ten zaznacza również fragmenty tekstu wychodzące na marginesy sygnalizowane przez komunikaty podczas kompilacji np. `Overfull \hbox (14.2264pt too wide) in paragraph at lines` Takie naruszenia marginesów oznaczane są czarnym prostokątem na końcu wystającej linii. Gdy linia ewidentnie nie mieści się w obszarze strony, to również ten prostokąt nie zostanie zaznaczony. Dlatego warto przejrzeć finalną postać tekstu, aby wykryć potencjalne niewielkie naruszenia marginesu.

3.2. Wykaz literatury

W wykazie literatury zamieszcza się wyłącznie pozycje, na które powołano się w pracy. Kolejność numerów w wykazie – zgodna z kolejnością pojawiania się danej pozycji w tekście.

Format akapitu: akapit wyjustowany, wysunięcie 0,75 cm. Prawidłowo opracowany wykaz został zaprezentowany w niniejszym dokumencie w odpowiednim rozdziale, oznaczonym jako „Literatura” (pozycja nr [1] to zasoby internetowe, [2] – książka, [3] – artykuł w czasopiśmie, [4] – karta katalogowa).

3.3. Wydruk pracy

Przed wydrukiem należy usunąć ewentualne błędy literowe i sprawdzić prawidłową interpunkcję. Przykładowo, łącznik zapisuje się za pomocą krótkiego minusa (np. badawczo-rozwojowy) natomiast myślnik – stosowany w zdaniach wtrąconych – zapisuje się za pomocą długiej pauzy. Dzielenie wyrazów według uznania autora (można podzielić długie wyrazy, powodujące duże „rozstrzelenie” tekstu w poprzedzającym wierszu. Zaleca się usunięcie pojedynczych znaków na końcu wiersza przez zastosowanie w L^AT_EX znaku tyldy w tekście. Dla przedrostka „mikro” należy unikać stosowania litery „u” czy „μ” zamiast „μ”. Znak stopień otrzymuje poleceniem `\circ`. W celu uniknię-

cia „rozstrzelenia” liczb i ich jednostek zaleca się używanie „twardej” spacji pomiędzy liczbą i jednostką (znak tyldy w L^AT_EX „~”). Należy sprawdzić, czy tytuły podrozdziałów/zakresów nie zostały jako pojedyncze wiersze na poprzedniej stronie oraz czy rysunki/tabele i ich tytuły nie zostały rozdzielone pomiędzy kolejnymi stronami.

Pracę drukuje się dwustronnie. Zaleca się wydruk w kolorze. Przed wydrukiem należy ponumerować strony (czcionka 10 pkt, dół strony, akapit wyśrodkowany). Strony tytułowej oraz strony z podziękowaniem pozostawia się bez widocznego numeru strony. Spis treści rozpoczyna się od strony numer 3 (lub 5, jeżeli zamieszczono podziękowania).

4. Podsumowanie i wnioski końcowe

1 ÷ 3 stron merytorycznie podsumowanie najważniejszych elementów pracy oraz wnioski wynikające z osiągniętego celu pracy. Proponowane zalecenia i modyfikacje oraz rozwiązania będące wynikiem realizowanej pracy.

Ostatni akapit podsumowania musi zawierać wykaz własnej pracy dyplomanta i zaczynać się od sformułowania: „Autor za własny wkład pracy uważa: ...”.

Załączniki

Według potrzeb zawarte i uporządkowane uzupełnienie pracy o dowolny materiał źródłowy (wydruk programu komputerowego, dokumentacja konstrukcyjno-technologiczna, konstrukcja modelu – makiety – urządzenia, instrukcja obsługi urządzenia lub stanowiska laboratoryjnego, zestawienie wyników pomiarów i obliczeń, informacyjne materiały katalogowe itp.).

Literatura

- [1] <http://weii.portal.prz.edu.pl/pl/materialy-do-pobrania>. Dostęp 5.01.2015.
- [2] Jakubczyk T., Klette A.: Pomiary w akustyce. WNT, Warszawa 1997.
- [3] Barski S.: Modele transmitancji. Elektronika praktyczna, nr 7/2011, str. 15-18.
- [4] Czujnik S200. Dokumentacja techniczno-ruchowa. Lumel, Zielona Góra, 2001.
- [5] Pawluk K.: Jak pisać teksty techniczne poprawnie, Wiadomości Elektrotechniczne, Nr 12, 2001, str. 513-515.

STRESZCZENIE PRACY DYPLOMOWEJ INŻYNIERSKIEJ

ROBOT MOBILNY Z NAWIGACJĄ AUTONOMICZNĄ OPARTY NA ARDUINO

Autor: Kacper Rychel, nr albumu: EF-173701

Opiekun: Dr inż. Mariusz Mączka

Słowa kluczowe: Arduino Uno, Robot mobilny, Czujnik HC-SR04, Czujnik Ultradźwiękowy Odległości

Cel pracy: Stworzenie robota mobilnego, który potrafi omijać przeszkody i poruszać się autonomicznie w środowisku. Zakres pracy: zaprojektowanie platformy robota z silnikami i czujnikami ultradźwiękowymi do detekcji przeszkód; zaprogramowanie algorytmu omijania przeszkód i nawigacji w środowisku; możliwość rozbudowy o funkcje śledzenia wyznaczonego celu lub powrotu do bazy. Testy robota w różnych scenariuszach, takich jak omijanie przeszkód czy poruszanie się po określonej trasie.

BSC THESIS ABSTRACT

TEMAT PRACY PO ANGIELSKU

Author: Kacper Rychel, Student's ID: EF-173701

Supervisor: Prof. Imię i nazwisko opiekuna, (academic degree)

Key words: max. five keywords in English

Treść streszczenia po angielsku