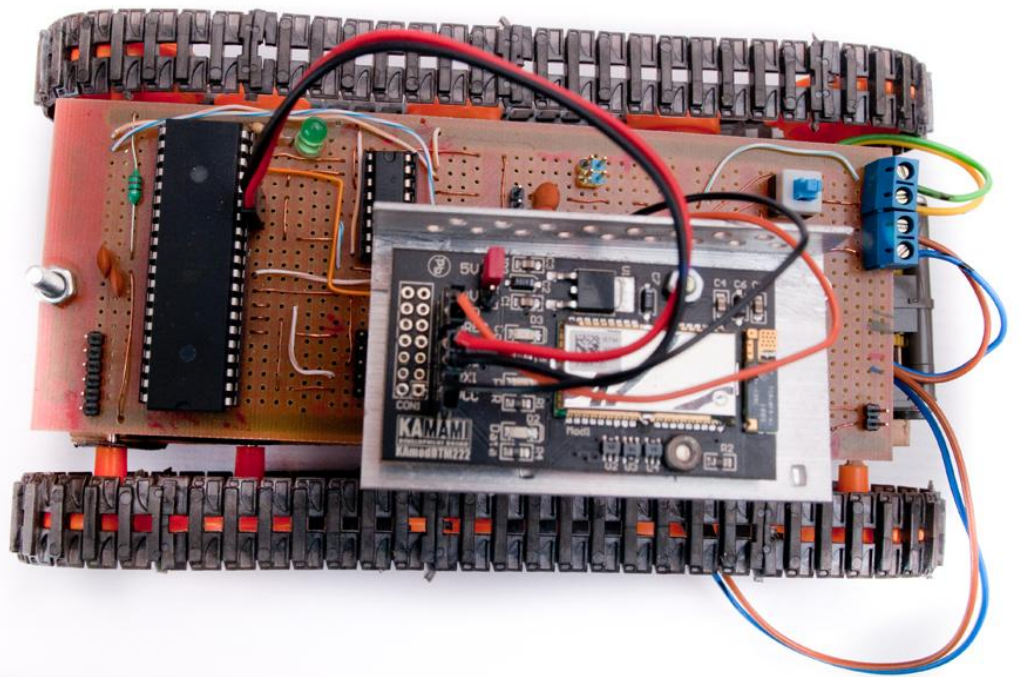


# LENy

## Zdalnie sterowany robot przy użyciu połączenia bluetooth



Marcin Kaciuba i Andrzej Kokoszka  
WFMII, II Informatyka

## Spis treści

<b><u>Założenia projektu</u></b> .....	<b>3</b>
<b><u>Budowa LENego - rozwiązania sprzętowe</u></b> .....	<b>3</b>
<u>Części</u> .....	3
<u>Schemat</u> .....	5
<u>Współpraca Atmegii z mostkiem</u> .....	6
<u>Zasilanie</u> .....	8
<u>Pulse-width modulation</u> .....	8
<b><u>Programowanie LENego - rozwiązania software'owe</u></b> .....	<b>9</b>
<b><u>Aplikacja sterująca LENym</u></b> .....	<b>10</b>
<b><u>Uwagi końcowe</u></b> .....	<b>11</b>
<b><u>Film pokazowy</u></b> .....	<b>11</b>

### 1. Założenia projektu

Celem, jaki sobie założyliśmy w ramach projektu z przedmiotu System Wbudowane było stworzenie robota-pojazdu poruszającego się na gąsienicach.

Robot miałby być sterowany za pomocą podłączonego panelu, bądź zdalnie za pomocą urządzenia na bazie systemu Android porozumiewając się poprzez moduł bluetooth.

### 2. Budowa LENego - rozwiązania sprzętowe.

#### 2.1 Części

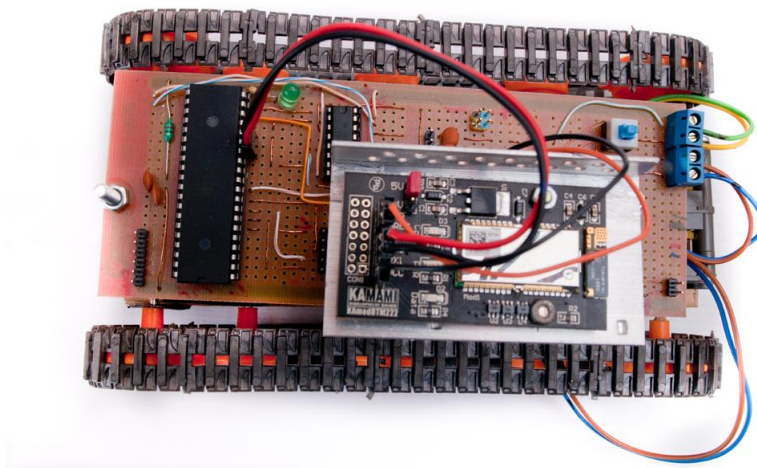
Do budowy LENgo wykorzystaliśmy następujące części:

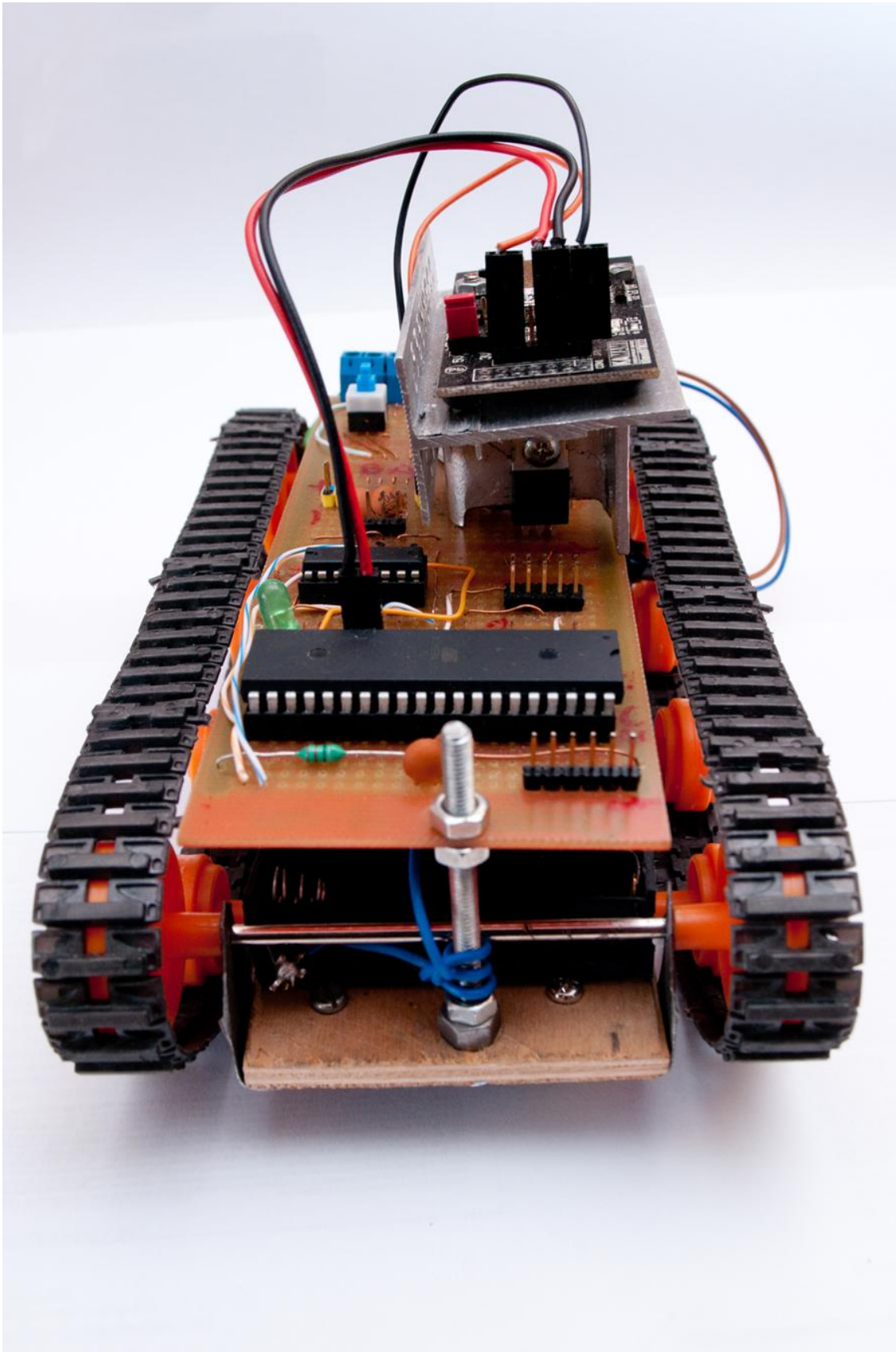
##### Elementy elektroniczne:

- ATmega16A-PU - DIP
- Dioda LED 5mm zielona
- Stabilizator 5V 7805
- Kondensator elektrolityczny 470uF/25V 105C
- Kondensator ceramiczny DIP 50V (5 szt.) - Wartość : 22pF
- Kondensator ceramiczny DIP 50V (5 szt.) - Wartość : 100nF
- Rezystor 330 Om - 1/4 W )
- Dławik przeciwzakłóceńowy 22uH
- Mostek H L293d

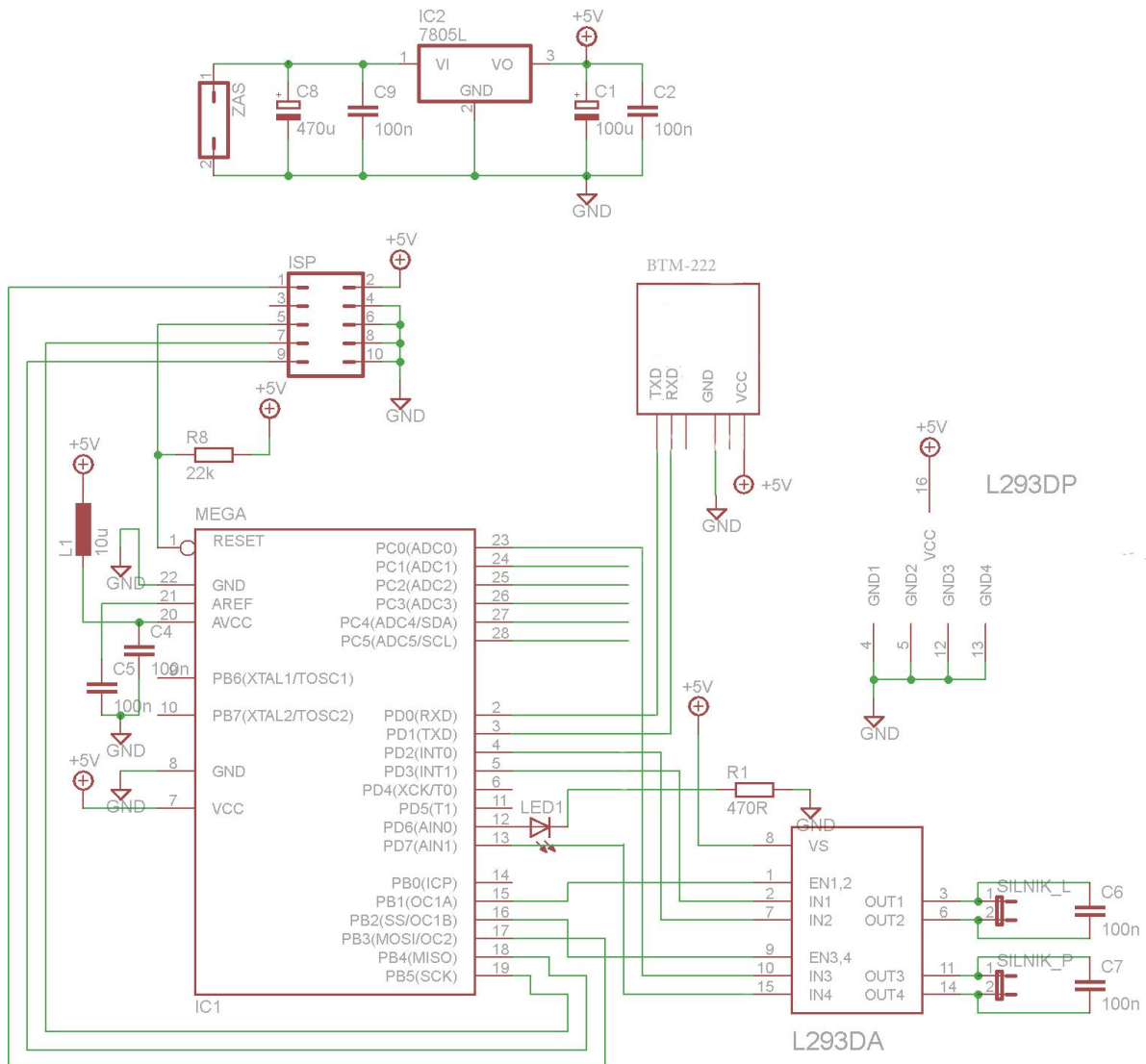
##### Elementy mechaniczne

- Zestaw kół gąsienicowych Tamiya 70100
- Przekładnia podwójna Tamiya 70097 -
- Kawałek sklejk





### 2.2 Schemat:



### 2.3 Współpraca Atmegii z mostkiem

EN1,2 – PWM dla silnika z wyjścia out1,2

Do tego pinu mostka podłączone jest OC1A(wyjście komparatora) atmegii

PD7 - IN4 OUT4 silnik P

PC0 - IN3 OUT3 silnik P

PD2 - IN2 OUT1 silnik L

PD3 - IN1 OUT2 silnik L

Z atmegii OC1B(wyjście komparatora) jest podłączone do EN 3,4. Analogicznie do EN1,2. Schemat jest pod atmega8, ale jeśli się pod odpowiadające piny może on działać dla każdej wyższej.

#### Działanie Mostka

EN1,2 IN1 IN2 Silnik

1 1 1 jedz

1 0 0 stop

1 1 0 Zmiana kierunku

1 0 1 jedz

0 X x Zabroniony,stop

X- bez znaczenia

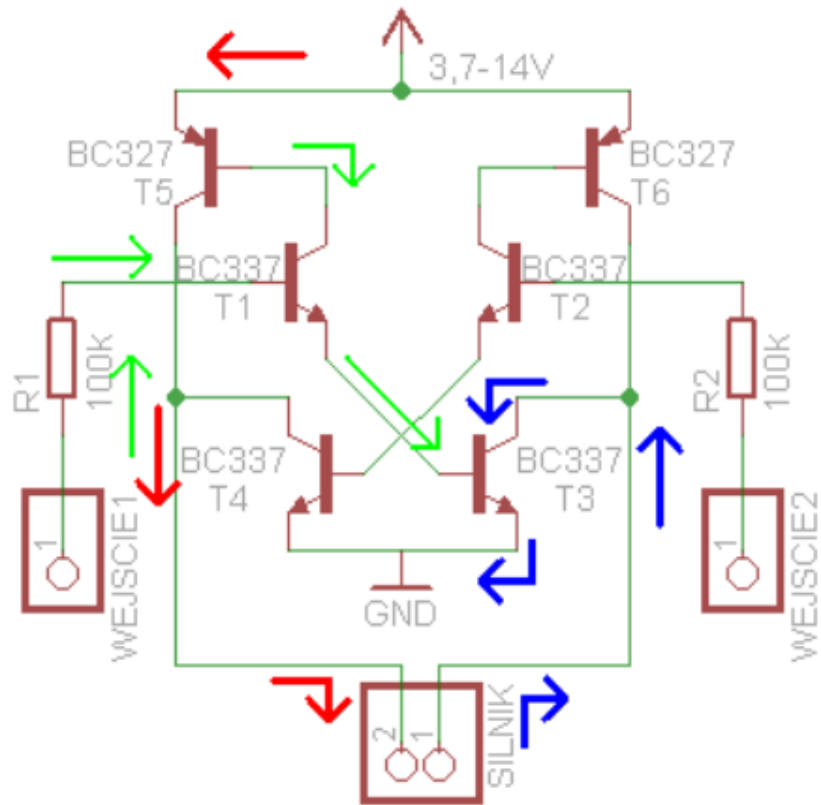
Analogicznie dla drugiego silnika.

Więcej informacji można znaleźć w dokumentacji mostka L293d.

#### Mostek H

Jest to układ pozwalający zmienianie kierunku obrotów silników. Złożony jest on z tranzystorów

połączony na kształt litery H. Dla tego nazywa się mostek H. Posiada on też możliwość płynnej regulacji napięcia na wyjściu dzięki PWM.



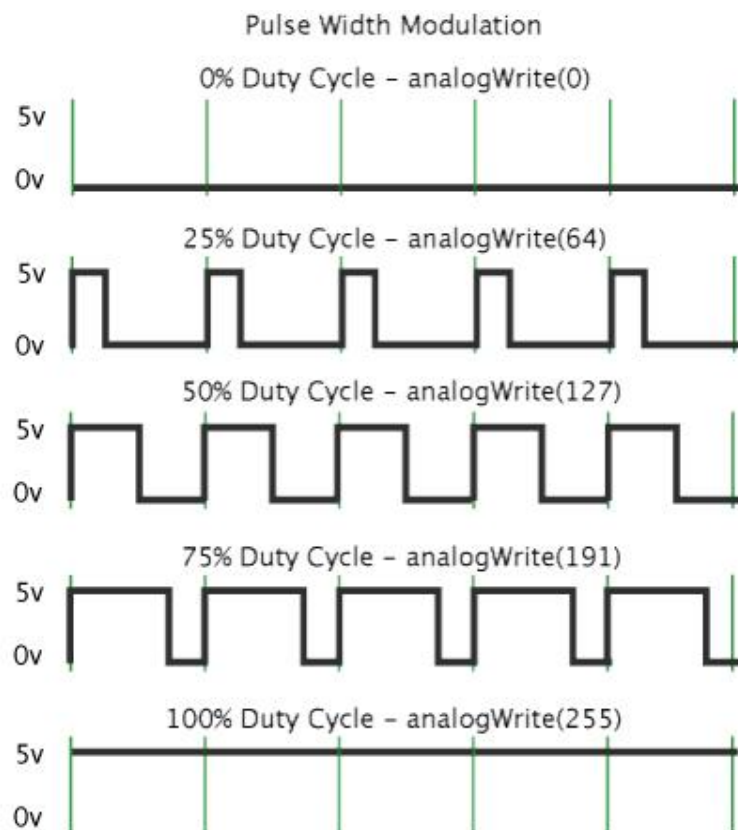


## 2.4 Zasilanie

Układ jest zasilany 6 akumulatorami 1.3 V 2400mAh co daje w sumie napięcie zasilania wynoszące 7.8V. Napięcie to jest stabilizowane na 5 V i dla atmegii i dla mostka H. Przed stabilizatorem został dodana LED, której zadaniem jest informowanie o podłączenie zasilania, a jej intensywność świecenia sygnalizuje naładowanie lub rozładowanie akumulatorów.

## 2.5 Pulse-width modulation

Pulse-width modulation - jest to modulacja wyjścia urządzenia. W tym przypadku napięcia na wyjściu mikrokontrolera. Istnieje kilka trybów modulacji na przykład dla atmegi16 jest ich 15. Do naszego projektu użyliśmy trybu 14 wraz z przerwaniami czyli fast pwm z top w rejestrze ICR1. Aby silnik nie dostawały zbyt wysokiego napięcia. Silniki dostają dzięki temu 1.5-2V.





### 3. Programowanie LENego - rozwiązania software'owe

Sterowanie robotem polega na ustawieniu odpowiednich bitów w portach wyjścia, czyli porty C i D.

Komunikacja Atmegii z modułem bluetooth odbywa się za pomocą portu UART. Inicjalizując ustawiamy odpowiednie flagi w rejestrach. Aby ustanowić połączenie z urządzeniem zewnętrznym ów urządzenie musi wywołać Len'ego który pełni rolę serwera i oczekiwać na połączenie. Urządzenie wysyła wiadomości a moduł je odczytuje, a Atmega ustawia porty D i C, podłączone do mostka.

Cały kod w załączniku „main2.c”. Poniżej fragment funkcji main.

```
int main()
{
    // . . .

    /* enable interrupt */
    sei();
    // Initialize communication
    if (KAmoBT_Initialize()!=BT_OK)
    {
        zapal1();
        // while(1);
        // Set_Name("Len");
    };
    while (1) {
        for (i=0;i<10;i++)
        {
            if (UCSRA&(1<<RXC)) {
                byte = USART_vReceiveByte();

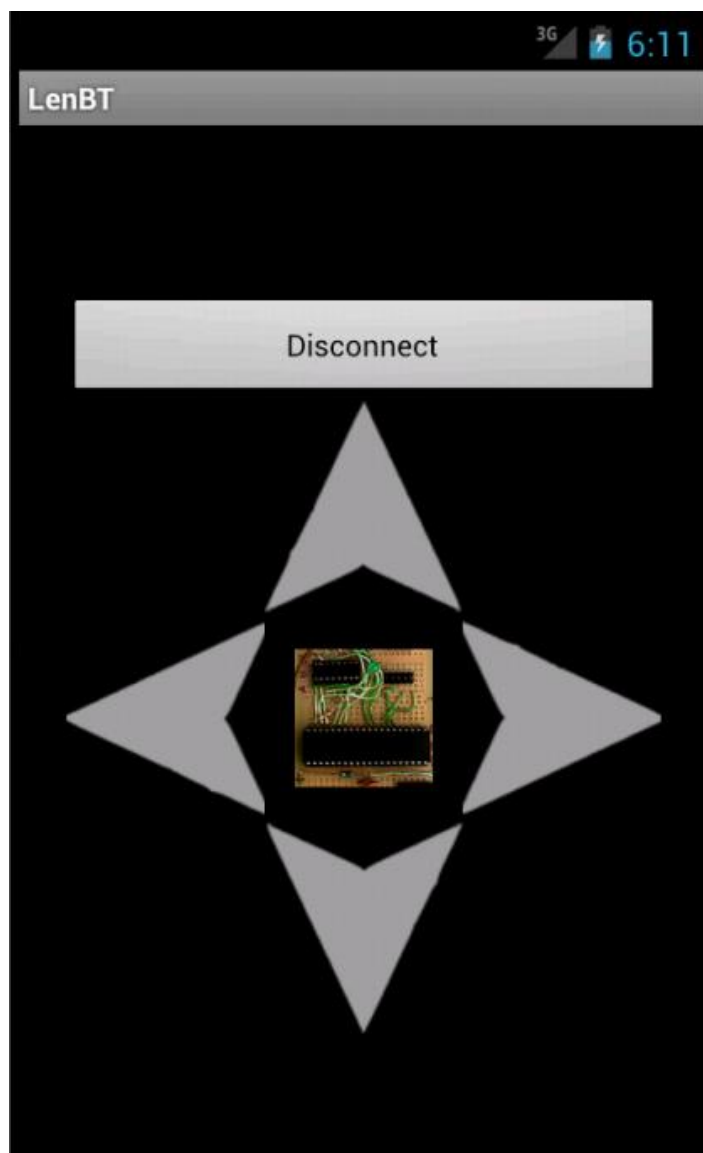
                switch (byte) {
                    case '0':
                        leds_clr();
                        break;
                    case '1':
                        leds_set();
                        break;
                    case '2':
                        kierunek(2);
                        break;
                    case '8':
                        kierunek(8);
                        break;
                    case '4':
                        kierunek(4);
                        break;
                    case '6':
                        kierunek(6);
                        break;
                    case '5':
                        kierunek(5);
                        break;
                    default:
                        break;
                }
            }
        }
        _delay_ms(250);
    }
    cli();
}
```

### 4. Aplikacja sterująca LENym

Aplikacja została napisana w języku Java pod system Android. Podstawowym i najważniejszym elementem programu jest ustanowienie poprawnego połączenia rfcomm z modulem bluetooth zamontowanym na LENym. Dokonaliśmy tego dzięki szeroko rozwiniętej i dobrze udokumentowanej obsłudze bluetooth w SDK. Urządzenie wybieramy z listy wcześniej sparowanych urządzeń, dzięki temu połączenie jest bezpieczne.

Po ustanowieniu połączenia nasz program przetwarza polecenia od użytkownika do LENego, który wykonuje rozkaz jazdy. W zależności od wydawanych komend – wciśniętych przycisków – interfejs graficzny przedstawiamy poniżej, do modułu wysyłane są odpowiednio instrukcje sterujące.

Kod w załączniku „LenBTActivity”.



### 5. Uwagi końcowe

Jedną z przeszkód, jako napotkaliśmy w trakcie realizacji projektu jest zawieszanie się modułu bluetooth, co powoduje niechciane zachowanie LENego. Pojazd wykonuje dwa razy tę samą instrukcję. Jest to spowodowane jednym źródłem energii zasilającym silniki i pozostałą elektronikę, co powoduje zakłócenia. Przewidywanym przez nas rozwiązaniem było by oddzielenie elementów, które mogą indukować prąd i powodować ów zakłócenia.

Mostek sterujący nagrzewa się do wysokich temperatur, co jest spowodowane wysokim poborem prądu przez silniki. W celu obniżenia temperatury do stabilizatora dołączony został radiator.

Sterowanie może się zmienić przy innym podłączeniu silników. Robot będzie wykonywał rozkazy odwrotnie, w związku z odwróconą polaryzacją.

### 6. Film pokazowy

Przedstawiamy film ilustrujący działanie naszego robota:

[LEN BT](#)