

# Wizualizacja danych pomiarowych z sensorów odległości umieszczonych na obrotowej platformie

Piotr Dąbek, Mateusz Pawelec

20 kwietnia 2017

# **Spis treści**

<b>1</b>	<b>Opis projektu</b>	<b>2</b>
<b>2</b>	<b>Sterownik robota</b>	<b>2</b>
<b>3</b>	<b>Komunikacja między modułami i komputerem</b>	<b>2</b>
<b>4</b>	<b>Moduł sensoryczny</b>	<b>3</b>
4.1	Schemat układu . . . . .	4
4.2	Projekt płytki drukowanej . . . . .	5
4.3	Wykonanie płytki . . . . .	6
<b>5</b>	<b>Aplikacja</b>	<b>6</b>
5.1	Funkcjonalności programu . . . . .	7
5.2	Diagram klas . . . . .	8
<b>6</b>	<b>Efekty działania</b>	<b>8</b>
<b>7</b>	<b>Połączenie radaru z Projektem Zespołowym</b>	<b>9</b>
<b>8</b>	<b>Projekt interfejsu</b>	<b>10</b>
<b>9</b>	<b>PCB</b>	<b>12</b>

# **1 Opis projektu**

Celem projektu było stworzenie oprogramowania graficznego służącego do wizualizacji a także przetwarzania danych odbieranych z zewnętrznego skanera otoczenia. Skaner jest rozbudowaniem projektu powstającego w ramach przedmiotu Projekt Zespołowy. Do oprogramowania została stworzona dokumentacja przy użyciu narzędzia doxygen.

Skaner składa się z trzech części:

- płytki PCB zawierającej sensory odległości i mikrokontroler
- silnika krokowego
- Sterownika robota komunikującego się z modułem sensorycznym, sterującego silnikiem krokowym oraz wysyłającego dane do komputera poprzez moduł bluetooth.

# **2 Sterownik robota**

Sterownik robota bazuje o moduł MLAGA zawierający 32-bitową jednostkę MK40DN512VLK10 firmy NXP(Freescale). Sterownik kieruje pracą silnika krokowego przy pomocy układu ULN2003. Jeden krok silnika odpowiada o obrót o 7,5 stopnia jednak możliwe jest sterowanie silnika półkrokowo. W ten sposób uzyskana dokładność zwiększa się dwukrotnie. Funkcja do obsługi kroków napisana jest w taki sposób, że kolejny krok następuje dopiero po odebraniu poprawnych danych. Jeśli dane nie zostaną odebrane poprawnie program czeka na kolejne. Uzależnienie obrotu od odebrania danych jest doskonale widoczne przy pracy z modułem HC-SR04 i jest to związane z tym jak odbierane są dane z tego czujnika. Można zaobserwować, że częstotliwość wykonywania kroków jest odwrotnie proporcjonalna do mierzonej odległości.

# **3 Komunikacja między modułami i komputerem**

Komunikacja z komputerem przebiega dwuetapowo. Pierwszym krokiem jest przesyłanie danych z modułu sensorycznego do sterownika robota. Kolejnym krokiem jest przesył danych ze sterownika do komputera. Wysyłane paczki

danych w obu przypadkach przyjmują następującą postać  
 $K : < 254 - \text{wychylenie} - \text{Sharp} - HC - SR04 - \text{sumakontrolna} - 255 >$

Liczby 254 i 255 kolejno oznaczają start i koniec paczki danych. Wychylenie to kąt o który obrócony jest silnik krokowy. Moduł sensoryczny jako tą daną wysyła kolejne liczby. Danej tej używa się w przypadku, gdy komunikacja modułu sensorycznego odbywa się bezpośrednio z komputerem i symuluje kąt wychylenia silnika. Kolejne dwie wartości to odczyt z dalmierzy podany w centymetrach. Suma kontrolna służy do sprawdzenia poprawności odebranej paczki danych i obliczana jest za pomocą algorytmu Luhna. Algorytm umożliwia wykrycie pomyłki pojedynczej cyfry lub większości zamian kolejności sąsiednich cyfr. Główną słabością jest niewykrywanie zamiany sekwencji 90 na 09 i odwrotnie. Do komunikacji w obu przypadkach używany jest protokoł USART. Przesył danych z modułu sensorycznego do sterownika robota odbywa się przez kabel natomiast sterownika z komputerem poprzez technologię Bluetooth i moduł HC-06. Warto zauważyć, że podczas pracy nad projektem największa ilość błędów sumy kontrolnej pojawiała się, gdy w pobliżu pracowało kilka urządzeń wykorzystujących technologię Bluetooth.

## 4 Moduł sensoryczny

Płytki PCB działającej jako radar zawiera dwa czujniki odległości

- optyczny Sharp GP2Y0A02YK0F
- ultradźwiękowy HC-SR 04

Użycie dwóch dalmierzy umożliwia lepsze wykrywanie przeszkód ze względu na różne specyfikacje czujników oraz to, że każdy z nich lepiej spełnia swoje zadanie w określonych sytuacjach. Zalety i wady tych dalmierzy wynikają bezpośrednio z rodzaju fal, które są wykorzystywane do pomiaru odległości. Czujnik ultradźwiękowy jest niewątpliwie wolniejszy biorąc pod uwagę prędkość rozchodzenia się dźwięku w powietrzu, która średnio wynosi 343m/s. Ponadto oba rodzaje fal ulegają podstawowym zjawiskom fizycznym takim jak odbicie czy załamanie. Optyczny czujnik Sharp lepiej sprawdza się przy obiektach, które nie są prostopadłe do źródła wysyłanej fali. Dalmierze ultradźwiękowe mają tendencję do odbierania fali odbitej od przedmiotu ustalonego pod kątem i w efekcie pomiar jest zakłamany a otrzymana odległość większa niż rzeczywista. Wadą czujnika Sharp jest natomiast jego nieliniowa charakterystyka napięcia wyjściowego od odebranej odległości. Funkcja

aproksymująca jest funkcją potęgową, która nie pokrywa się idealnie z charakterystyką podaną w dokumentacji. Ponadto efektem tego, że charakterystyka czujnika przypomina funkcję potęgową jest to, że dla dużych odległości zmiana napięcia wyjściowego jest stosunkowo niewielka przez co czujnik traci na dokładności.

Funkcja, którą przybliżony jest odczyt czujnika Sharp przyjmuje jedną z następujących postaci:

$$y = 65.734 * x^{-1,268}$$

lub

$$y = 60.459 * x^{-1,1904}$$

Charakterystyka czujnika opisana w dokumentacji nie jest dokładna i podane punkty nie są jednoznacznie określone dlatego mogą również powstawać błędy wynikające ze złego odczytania wartości. W programie użyta została funkcja druga, która dla dużych odległości jest bardziej łagodna. Dla odległości średnich i małych funkcje praktycznie pokrywają się ze sobą.

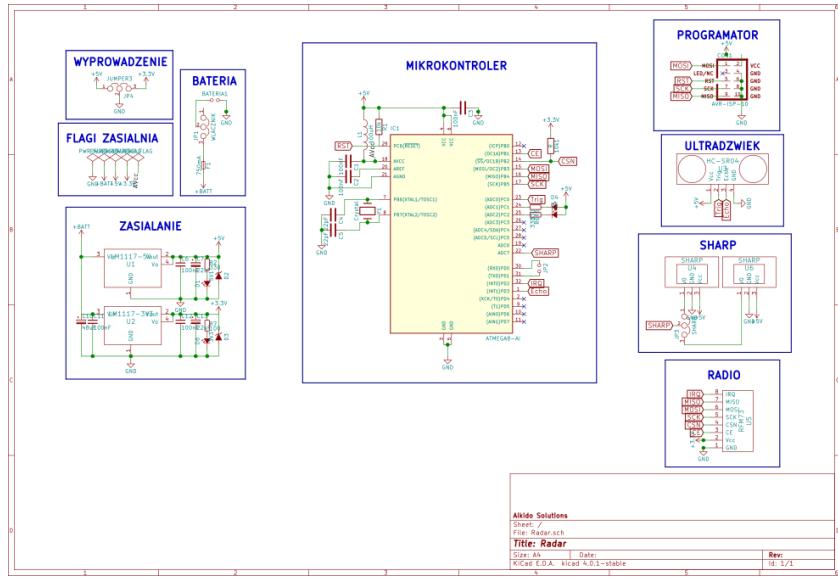
Czujnik HC-SR04 na wyjściu układu zwraca impuls o długości od 150 us do 25ms w zależności od zmierzonej odległości. Jest to w pewien sposób wada czujnika ponieważ ogranicza to ilość możliwych pomiarów na sekundę. Do przeliczenia długości impulsu na centometry użyto wzoru podanego przez producenta w dokumentacji czujnika:

$$cm = \text{pulsewidth}(us)/58$$

Podany wzór zakłada, że prędkość dźwięku jest równa 340m/s.

## 4.1 Schemat układu

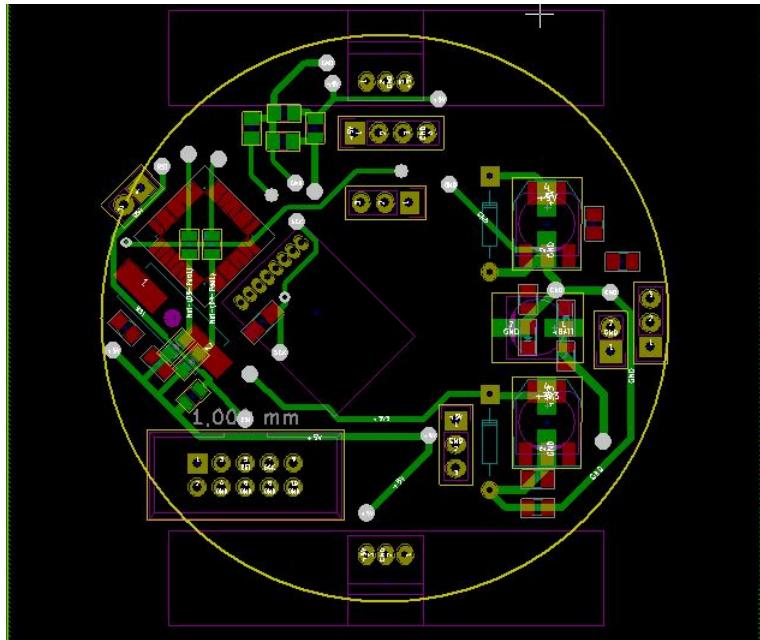
Układ radaru składa się z dwóch głównych części – sekcji zasilania i mikrokontrolera Atmega8 wraz z podpiętymi do niego zewnętrznymi peryferiami. Zasilanie składa się z dwóch stabilizatorów liniowych, które są odpowiednio filtrowane, zarówno na wejściu jak i na wyjściu. Na wyjściu stabilizatora ponadto umieszczona została dioda Zenera zapobiegająca przed pojawiением się w układzie napięcia wyższego niż planowane. Zasilanie sygnaлизowane jest poprzez zapalenie diod LED. Wejścia zasilania mikrokontrolera również są filtrowane a zasilanie części analogowej podłączone jest poprzez dławik o wartości 10uH. Biorąc pod uwagę, że do poprawnego działania dalekometra HC-SR06 wymaga dokładnego pomiaru czasu mikrokontroler taktowany jest zewnętrznym kwarcem 8MHz. Schemat układu jak również płytka PCB zostały zaprojektowane w programie KiCad firmy EDA Software.



Rysunek 1: Schemat radaru

## 4.2 Projekt płytki drukowanej

Główym założeniem przy projektowaniu płytki PCB były możliwe małe wymiary oraz pusta przestrzeń na środku płytki umożliwiająca zamocowanie radaru do silnika krokowego. Wykonana płytka PCB jest kołem o średnicy ok. 8cm. Wybrany kształt płytki oraz pozostawiona przestrzeń stanowiły duże utrudnienie przy projektowaniu. Większość elementów użytych w projekcie znajduje się w obudowach SMD. W celu ograniczenia ilości przelotek wszystkie elementy przewlekane umieszczone zostały przeciwnej stronie niż elementy do których prowadzony jest sygnał. Wolne przestrzenie wypełnione są polami masy. Niektóre używane obudowy nie wchodziły w skład bibliotek dołączonych do programu KiCad i konieczne było zaprojektowanie ich w jednym w podprogramów.



Rysunek 2: Projekt PCB

### 4.3 Wykonanie płytki

Płytnka została wykonana metodą fototransferu. Metoda ta pozwala na wykonanie PCB w lepszej jakości niż alternatywna metoda termotransferu i użycie jej było konieczne ze względu na małe rozmiary trawionych ścieżek. Obie warstwy projektu PCB wydrukowano na folii, a następnie sklejono tak, aby odpowiadające sobie przelotki pokrywały się. Kolejnym krokiem było naświetlenie płytka. W tym celu użyliśmy dostępnej naświetlarki domowej roboty. Po naświetleniu płytka została włożona od wywoływacza. Tak przygotowana płytka została włożona do roztworu wytrawiacza a po trawieniu przycięta do odpowiednich rozmiarów. Przy naświetleniu, kontakcie z chemią i mechanicznej obróbce płytka zachowane zostały konieczne środki ostrożności.

## 5 Aplikacja

Aplikacja komputerowa została wykonana na bazie biblioteki graficznej do języka C++ - Qt. Główna funkcjonalnością aplikacji jest interfejs do komunikacji użytkownik-skaner jak również graficzna reprezentacja otoczenia robota na podstawie odczytanych i przetworzonych pomiarów. Aplikacja umożliwia

wyświetlanie odczytów z czujników osobno jak również nakładając je na siebie. W ten sposób będzie możliwe dokładniejsze odzwierciedlenie rzeczywistości jak również zminimalizowanie błędów pomiarowych.

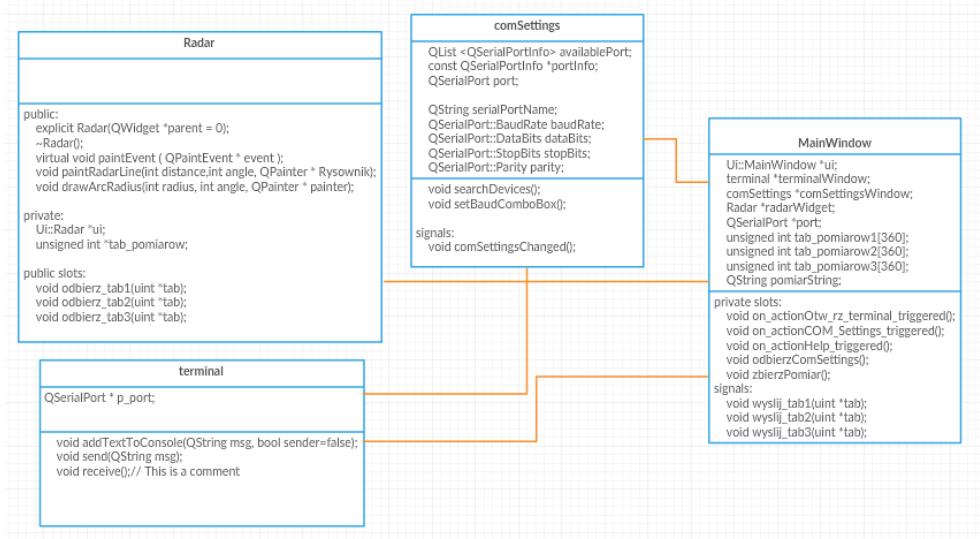
Główna klasą aplikacji jest klasa MainWindow. W niej zawierają się wskaźniki na obiekty klas podrzędnych jak również to ona odpowiedzialna jest za odbiór danych. Klasa comSettings przechowuje parametry z jakimi otwarty jest port szeregowy(Baud rate, przesyłane bity, bity parzystkości itp.) i sam wskaźnik na ten port. Informacje te przesyłane są do klasy MainWindow w momencie otwarcia portu. Jeśli otwarcie portu szeregowego nie jest możliwe wyświetlana jest stosowna informacja. Brak dostępu do portu może być spowodowany m.in tym, że port otwarty jest w innym programie lub to, że użytkownik nie posiada odpowiednich praw do używania portu. W drugim przypadku należy te prawa zmienić używając w terminalu poleceń pojawiającego się w oknie Help. W przypadku komunikacji przez moduł Bluetooth trzeba skonfigurować plik rfcomm.conf oraz nawiązać połączenie z modułem znajdującym się na płytce robota. W momencie otwarcia portu ustanawiane jest połączenie sygnału readyRead, który emitowany jest gdy na porcie pojawią się dane do odczytu i slotu wywołującego odebranie dostępnych danych. Po poprawnym odczytaniu danych i sprawdzeniu sumy kontrolnej dane zapisywane są do tablicy przechowującej zebrane do tej pory pomiary a także emitowany jest sygnał odświeżający narysowany obraz.

Klasa Radar służy do wizualizacji danych odebranych ze sterownika robota. Przechowuje ona wskaźniki na tablice do których zapisane są odczytane do tej pory pomiary i na ich podstawie rysuje obraz za pomocą klasy QPainter. Rysowanie odbywa się poprzez transformację i rotację układu współrzędnych. Każdorazowe odebranie nowych danych emituje sygnał, który wywołuje slot odświeżający obraz radaru klasą paintEvent. Aby zapisać aktualnie odczytane pomiary w formie graficznej aktualny obraz przechowywany jest w lkasie QImage, a następnie zapisywany na dysku.

## 5.1 Funkcjonalności programu

- połączenie z radarem
- wybór parametrów połączenia
- graficzna reprezentacja otoczenia robota
- zapis danych pochodzących z czujników w formie pliku graficznego
- zapis danych do pliku tekstowego

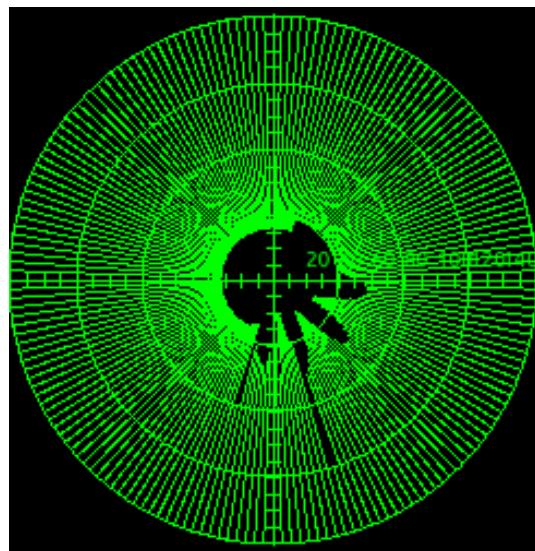
## 5.2 Diagram klas



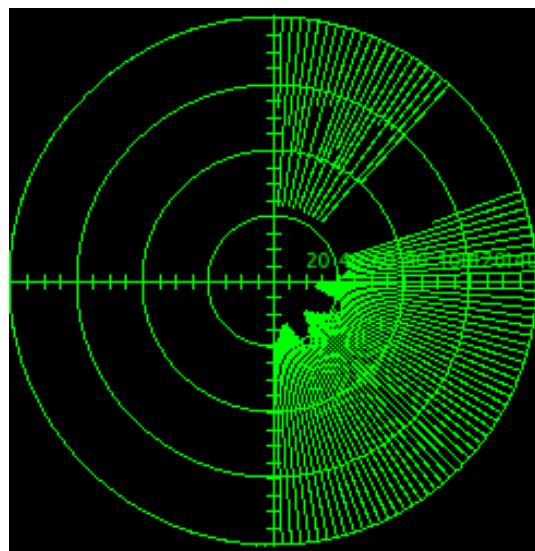
Rysunek 3: Diagram Klas

## 6 Efekty działania

W wyniku pracy nad radarem udało nam się stworzyć w pełni działające urządzenie mogące skanować otoczenie robota i wizualizować je w oknie aplikacji. Aplikacja umożliwia zapis uzyskanych pomiarów zarówno w formie graficznej jak i tekstowej. Efekty działania wodoczne są poniżej:



Rysunek 4: Zapis do pliku



Rysunek 5: Zapis do pliku

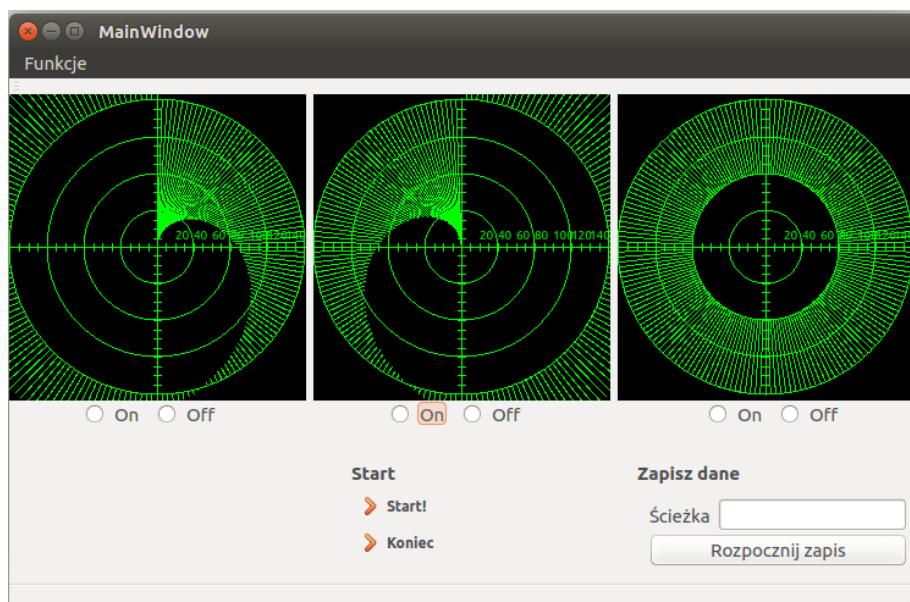
## 7 Połączenie radaru z Projektem Zespołowym

W projekcie zespołowym moduł sensoryczny umożliwia zatrzymanie robota przed przeszkodą. Silnik krokowy ustawiany jest tak, aby pokrywał się z

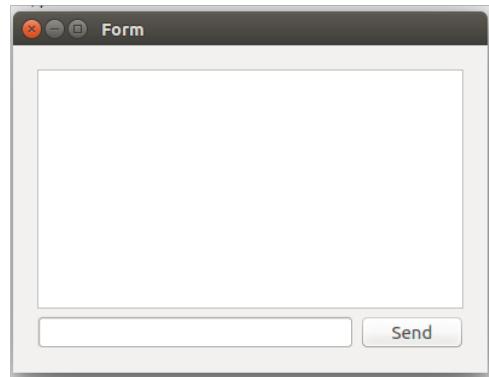
kieunkiem jazdy. Gdy przeszkoda zostanie wykryta w odległości mniejszej niż 50 cm robot nie może dalej poruszać się w tę stronę i musi zmienić kierunek.

Na potrzeby projektu na Wizualizację Danych Sensorycznych sterownik robota służy do obsługi silnika krokowego w trybie półkrokowym i pośredniczy w komunikacji z komputerem. Radar obraca się o 360 stopni zgodnie z ruchem wskazówek zegara a następnie przeciwnie.

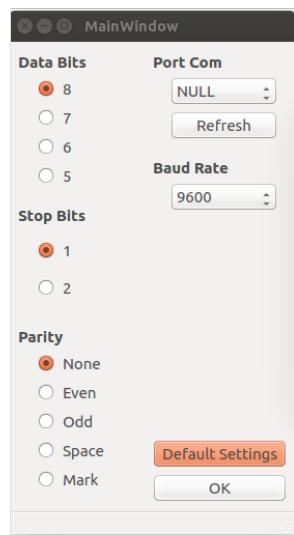
## 8 Projekt interfejsu



Rysunek 6: Okno główne



Rysunek 7: Terminal

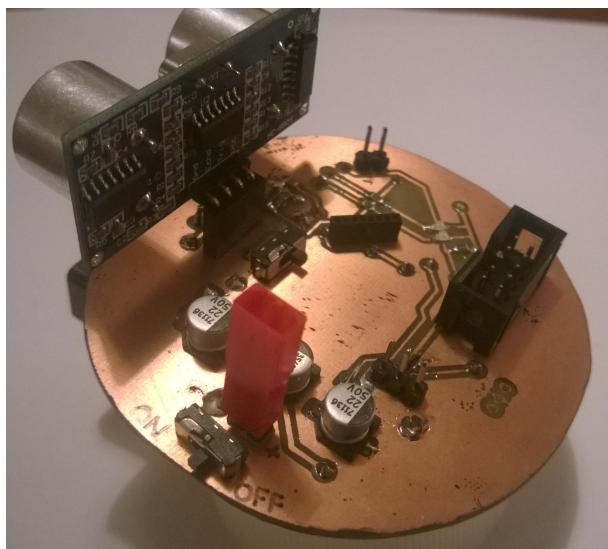


Rysunek 8: Ustawienia portu COM

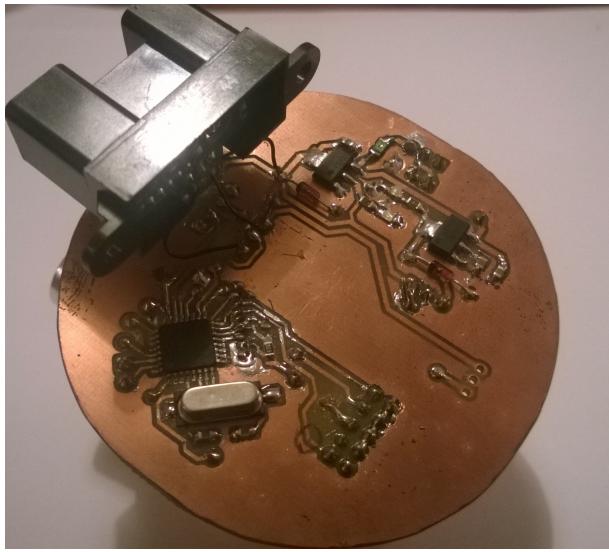
## 9 PCB



Rysunek 9: PCB, średnica ok. 8 cm



Rysunek 10: Strona górnna



Rysunek 11: Strona dolna