

POLITECHNIKA WROCŁAWSKA
WYDZIAŁ INFORMATYKI I TELEKOMUNIKACJI

Urządzenia peryferyjne

Silnik Krokowy

ĆWICZENIE NR 15

PROWADZĄCY:

dr inż. Jan Nikodem

GRUPA:

Piątek 13:15 TP

AUTORZY:

Daniel Glazer, 252743

Paweł Helisz, 252779

Wrocław 14.12.2021

1 Cel ćwiczenia

Celem ćwiczenia było napisanie programu sterującego silnikiem krokowym M42SP-7.

2 Wstęp teoretyczny

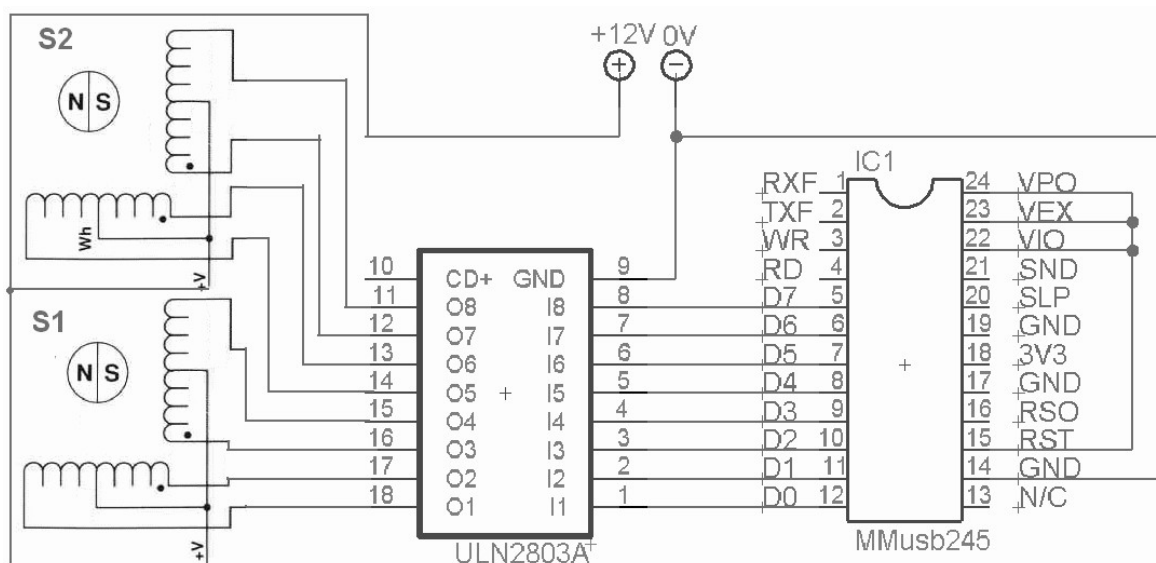
Silnik krokowy jest silnikiem elektrycznym, który charakteryzuje się impulsowym zasilaniem prądem elektrycznym. Dzięki temu wirnik nie obraca się ruchem ciągłym, tylko za każdym razem wykonuje ruch o ustalony wcześniej kąt obrotu. Z tego powodu nadaje się do ustalania precyzyjnej pozycji przy sterowaniu w otwartej pętli sprzężenia zwrotnego.

2.1 Budowa silnika krokowego

Silnik krokowy składa się z rotora i stojana, które umożliwiają jego rotację. Wyróżnia się wiele rodzajów silników krokowych. Najczęstsze różnice występujące między nimi to: sposób sterowania, liczba uzwojeń, a także rodzaj materiału, z którego zostały wykonane. Wyróżniamy kilka rodzajów napędów krokowych prądu stałego:

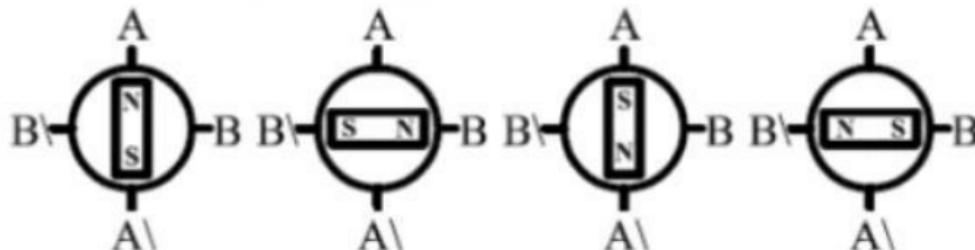
1. o zmiennej reluktancji (VR)
2. z magnesem stałym (PB) oraz hybrydowe (HB)
3. bipolarne i unipolarne - silniki z dzielonym uzwojeniem

Na zajęciach rozważanym silnikiem był silnik unipolarny, czyli taki silnik, który posiada uzwojenie z odczepem. Sterowanie odbywa się za pomocą portu USB komputera PC. Za konwersję standardu USB na 8 bitową linię odpowiedzialny jest układ firmy FTDI o symbolu FT245BM. Do wyjścia tego układu, które ma postać 8 bitowej szyny dołączony jest układ ULN2803A. Jest to 8 układów Darlington'a z zanegowanym wyjściem, pozwalających sterować uzwojeniami 2 silników unipolarnych. Dla silnika M42SP-7 wartość obrotu dla jednego kroku wynosi 7.5° .

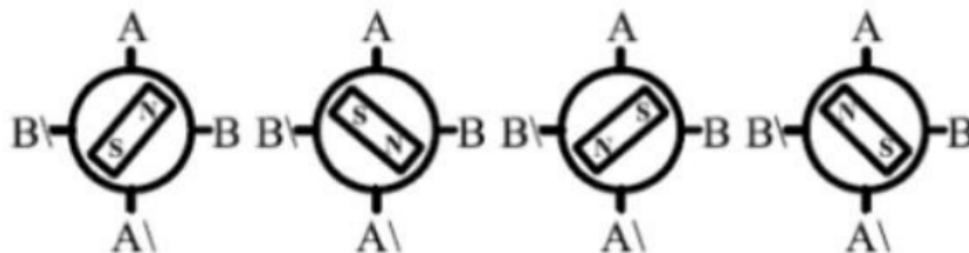


3 Rodzaje sterowania silnikiem krokowym

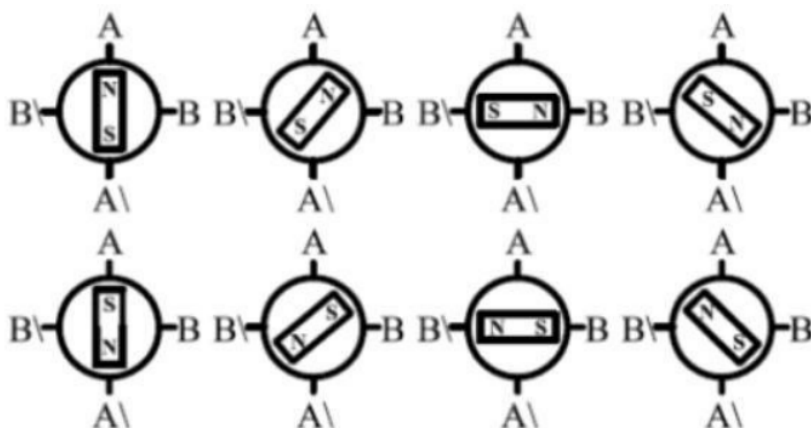
1. Sterowanie falowe - inaczej zwane jednofazowym. Kolejność ich pracy jest następująca: A-B-A'-B'. Wynikiem tego rodzaju sterowania jest wykonanie pełnego kroku. Silnik unipolarny wykorzystuje przy sterowaniu falowym 25% całkowitego uzwojenia silnika w danej chwili czasu.



2. Sterowanie pełnokrokowe - inaczej zwane dwufazowym. W każdej chwili czasu zasilane są dwie fazy. Kolejność pracy opisuje się jako AB-A'B'-AB'. Wynikiem tego rodzaju sterowania są takie same ruchy jak przy sterowaniu jednofazowym z tym, że pozycja wirnika jest przesunięta o pół kroku. Silnik używający sterowania pełnokrokowego uzyskuje dwukrotnie większy moment wyjściowy w porównaniu do sterowania jednofazowego, będący odpowiednikiem 50% całkowitego uzwojenia silnika.



3. Sterowanie półkrokowe - jest kombinacją sterowania dwufazowego i jednofazowego. Stojan zasilany jest naprzemiennie na jedną, a następnie dwie fazy. Daje to w efekcie dwukrotnie krótszy krok. Schemat pracy w trybie półkrokowym opisuje się kolejno AB-B-A'B'-A'-A'B'-B'-AB'-A.



4 Program

4.1 Załączone biblioteki

Za obsługę układu od firmy FTDI odpowiedzialna jest biblioteka *FTD2XX_NET*. Która została załączoną do kodu programu:

```
using FTD2XX_NET;
```

4.2 Tablice rodzajów sterowania / kroków

Aby móc obsługiwać ruch za pomocą jednego z trzech trybów sterowania, zdefiniowano tablice zawierające szesnastkowo zapisane bity sterowania dla silnika poziomego i pionowego. W zależności od liczby zasilanych faz po przekonwertowaniu na system dwójkowy w zapisie występują jedna albo dwie jedyńki.

```
// ruch falowy
byte[] leftWave = { 0x08, 0x02, 0x04, 0x01 }; //7.5
byte[] rightWave = { 0x01, 0x04, 0x02, 0x08 };

// ruch pełnokrokowy
byte[] leftFullStep = { 0x09, 0x0A, 0x06, 0x05 }; //7.5
byte[] rightFullStep = { 0x05, 0x06, 0x0A, 0x09 };

// ruch polkrokowy
byte[] leftHalfStepping = { 0x09, 0x08, 0x0A, 0x02, 0x06, 0x04, 0x05, 0x01 }; //3.75
byte[] rightHalfStepping = { 0x01, 0x05, 0x04, 0x06, 0x02, 0x0A, 0x08, 0x09 };

// ruch falowy dla drugiego silnika
byte[] leftWave2 = { 0x10, 0x80, 0x20, 0x40 }; //7.5
byte[] rightWave2 = { 0x40, 0x20, 0x80, 0x10 };

// ruch pełnokrokowy dla drugiego silnika
byte[] leftFullStep2 = { 0x50, 0x90, 0xA0, 0x60, }; //7.5
byte[] rightFullStep2 = { 0x60, 0xA0, 0x90, 0x50 };

// ruch polkrokowy dla drugiego silnika
byte[] leftHalfStepping2 = { 0x50, 0x10, 0x90, 0x80, 0xA0, 0x20, 0x60, 0x40 }; //3.75
byte[] rightHalfStepping2 = { 0x40, 0x60, 0x20, 0xA0, 0x80, 0x90, 0x10, 0x50 };
```

Zadeklarowano również tablicę kończąca przesyłanie bitów do silnika:

```
byte[] stop = { 0x00 };
```

4.3 Obrót silnika

Obrót silnika odbywa się za pomocą podawania kolejnych komórek z tablic rodzajów sterowania, które zostały zadeklarowane wyżej. Po wykonaniu podanej przez użytkownika liczby kroków wysyłany jest sygnał kończący przesyłania bitów:

```
try
{
    for (int i = 0; i < stepNumber; i++)
    {
        int j;

        // wybieramy typ silnika
        if (engineType)
        {
            // wybieramy indeksowanie tablicy w zależności od sterownia
            if (rotationType)
                j = index % 4;
            else
                j = index % 8;

            // podawanie bitów sterownia
            byte[] x = { bytes[j] };
            ftstatus = device.Write(x, bytesToWrite, ref bytesWritten);

            System.Threading.Thread.Sleep(1000 / speed);
            index++;
        }
        ...
    }
}
finally
{
    // wysyłany sygnał kończący
    ftstatus = device.Write(stop, 1, ref bytesWritten);
}
```