

SPRAWOZDANIE					PROSZĘ Podać NR GRUPY:						
					ZIISS1	3	5	1	2	IO	
IMIĘ	NAZWISKO	Temat ćwiczenia zgodny z wykazem tematów:			PONIŻEJ PROSZĘ Podać TERMIN ZAJĘĆ:			ROK:			
Amelia	Lis	Podstawy programowania PWM Oraz silniki krokowe			PN	WT	SR	CZ	PT	SB	ND
					GODZINA ROZPOCZĘCIA ZAJĘĆ:						11 : 30
UWAGA !!! Wypełniamy tylko białe pola. W punkcie 1, proszę zakreślić odpowiednie pola i podać godzinę w której odbywają się zajęcia, zgodnie z planem zajęć.											

Wprowadzenie teoretyczne:

Rozwiń skrót PWM i przedstaw przykłady zastosowania

PWM, czyli Pulse Width Modulation, to technika szeroko stosowana w elektronice, umożliwiająca kontrolę mocy dostarczanej do urządzeń elektrycznych poprzez regulację szerokości impulsów sygnału. W skrócie, PWM polega na cyklicznym zmienianiu stosunku czasu trwania sygnału w stanie wysokim do czasu trwania sygnału w stanie niskim. Ta elastyczność sprawia, że PWM znalazło liczne zastosowania w różnych dziedzinach.

Jednym z kluczowych obszarów zastosowań PWM jest sterowanie prędkością silników elektrycznych. Dzięki regulacji szerokości impulsów, można precyzyjnie kontrolować ilość dostarczanej energii do silnika, co umożliwia płynną zmianę prędkości obrotowej. Przykładowo, w elektronice samochodowej PWM jest wykorzystywane do kontroli prędkości wentylatorów chłodzących silnik czy też do sterowania prędkością silników w elektrycznych pojazdach.

Kolejnym obszarem zastosowań PWM jest oświetlenie LED. Poprzez regulację szerokości impulsów sygnału zasilającego diody LED, można dostosować jasność oświetlenia. To rozwiązanie jest efektywne energetycznie i umożliwia tworzenie dynamicznych efektów świetlnych. PWM jest także wykorzystywane w oświetleniu ulicznym, gdzie pozwala na oszczędność energii poprzez regulację intensywności światła w zależności od warunków atmosferycznych czy pory dnia.

W dziedzinie elektroniki audio PWM znalazło zastosowanie w układach klasy D wzmacniaczy. Dzięki tej technice można uzyskać wysoką efektywność energetyczną, co jest istotne w urządzeniach przenośnych, takich jak głośniki Bluetooth. Ponadto, PWM jest wykorzystywane w regulacji głośności dźwięku w różnych urządzeniach audio, zapewniając dokładną kontrolę nad poziomem sygnału dźwiękowego.

W automatyce PWM jest używane do sterowania serwomechanizmami, które znajdują zastosowanie w sterowaniu ruchem w modelarstwie, robotyce czy nawet w lotnictwie. Poprzez precyzyjną kontrolę impulsów, można osiągnąć precyzyjne pozycjonowanie i stabilność ruchu. Sygnał PWM, generowany przez kontroler mikroprocesorowy, pełni rolę sterującą dla serwomechanizmów. Okres sygnału to czas cyklu, a stosunek czasu ON do czasu OFF określa szerokość impulsu, istotną dla serwomechanizmu. Serwa, zaprojektowane do określonych kątów obrotu, reagują proporcjonalnie do stosunku wypełnienia sygnału PWM.

Servomechanizmy posiadają pętlę sprzężenia zwrotnego, która monitoruje położenie. W trakcie działania serwo porównuje rzeczywiste położenie z żądanym (ustawionym przez PWM). Regulacja prędkości i kierunku następuje poprzez zmianę szerokości impulsów: krótszy impuls oznacza większą prędkość, a dłuższy spowalnia ruch. Manipulacja stosunkiem wypełnienia umożliwia także zmianę kierunku obrotu.

Dzięki zdolności serwomechanizmów do reagowania na zmiany w sygnale PWM, osiąga się wysoką stabilność i dokładność pozycjonowania. Pętla sprzężenia zwrotnego pozwala serwu dostosować położenie do zmian w warunkach obciążenia lub oporu, minimalizując błędy.

W energetyce odnawialnej PWM wykorzystuje się w regulatorach ładowania baterii w systemach fotowoltaicznych. Dzięki tej technologii można skutecznie kontrolować proces ładowania baterii, dostosowując go do zmiennych warunków pogodowych i zapewniając optymalne wykorzystanie energii słonecznej.

Podsumowując, PWM, czyli Pulse Width Modulation, to wszechstronna technika, która znalazła zastosowanie w wielu dziedzinach, umożliwiając precyzyjną kontrolę mocy i efektywne zarządzanie energią elektryczną. Jego elastyczność sprawia, że jest to niezastąpione narzędzie w projektowaniu nowoczesnych układów elektronicznych.

* rysunki proszę zamieszczać na drugiej stronie a w tekście podać odnośniki

UWAGA!

W sprawozdaniu nie trzeba wklejać listingów programu i nie trzeba robić zrzutów ekranu z zaprojektowanych schematów.

Proszę jednak pamiętać o dołączeniu plików w postaci INO I SIMU (w sumie 6 plików 3 INO i 3 SIMU)

UWAGA!!!!

W ZADANIU NIE WOLNO UŻYWAĆ GOTOWYCH BIBLIOTEK DO STEROWANIA SERWOMECHANIZMAMI.

Zadanie 1

Zestaw schemat jak na rysunku. Proszę zbudować zegar analogowy, który będzie odmierzał czas z dokładnością 1 sekundy.

Minutnik, sekundnik i zegar godzinowy zbuduj z użyciem serwomechanizmów (Rys.1)

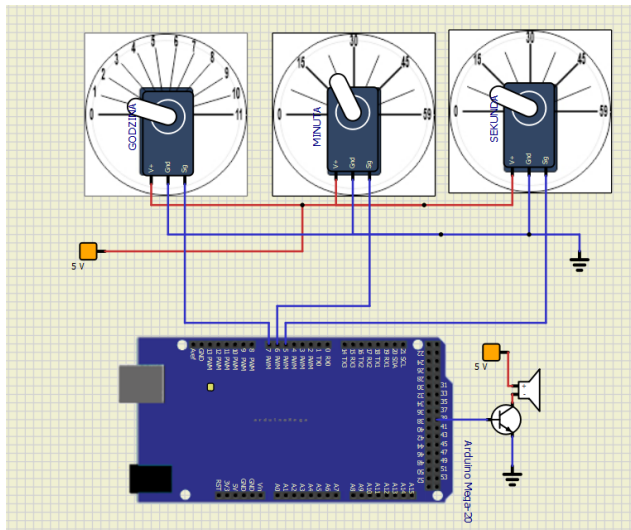
Tarcze zegarów pasujące do serwomechanizmów znajdziesz w postaci załączonych do zadania plików.

Będą to tarcze sekundowe i minutowe od 0 do 59 i tarcza godzinowa od 0 do 11

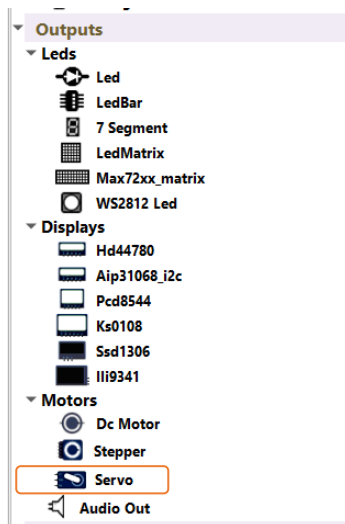
Ustal położenie wskazówki serwomechanizmu z wykorzystaniem sygnału PWM.

Po przejściu każdej minuty powinien pojawiać się sygnał dźwiękowy.

Proszę przyspieszyć działanie zegara dziesięciokrotnie



Rys. 1 Schemat połączeń serwomechanizmów w zadaniu 1



Rys. 2 Menu „Outputs” z zaznaczonym serwomechanizmem

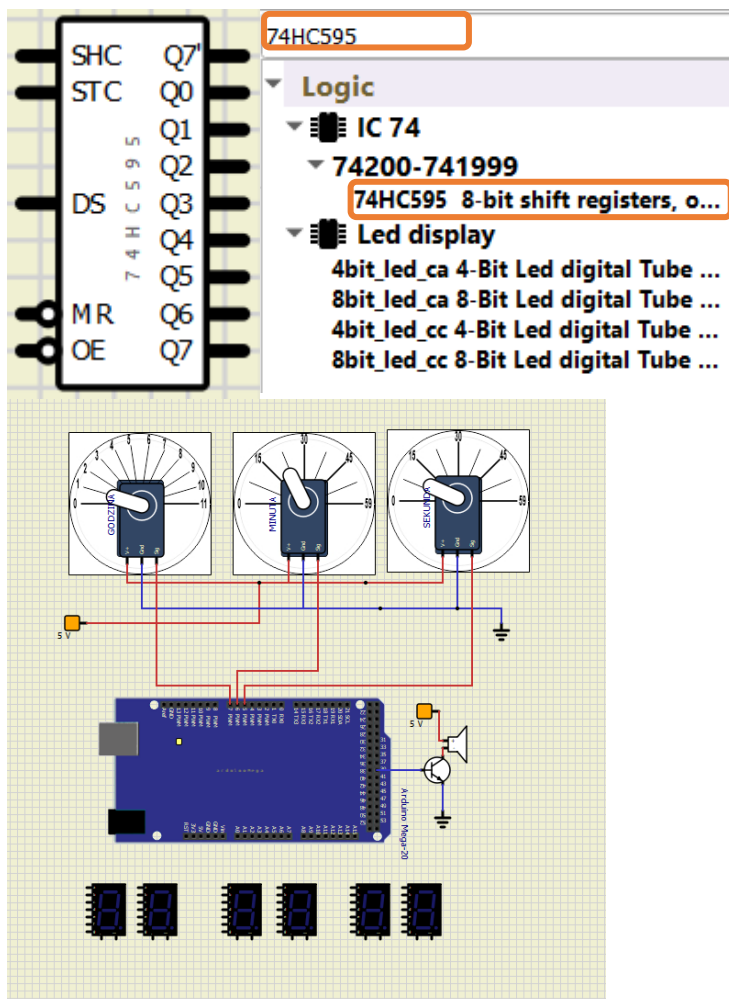
Zadanie 2

Po wykonaniu zadania pierwszego dołącz i oprogramuj sześć wyświetlaczy led, które będą spełniały identyczną rolę jak serwomechanizmy.

Proszę spróbować samodzielnie skonstruować schemat tak aby wyświetlaczami można było sterować za pomocą 3 portów.

Wskazówka:

Użyj układów 74 HC 595 aby zwiększyć liczbę portów. Wyszukaj scalaka wpisując jego symbol na pasku wyszukiwania.

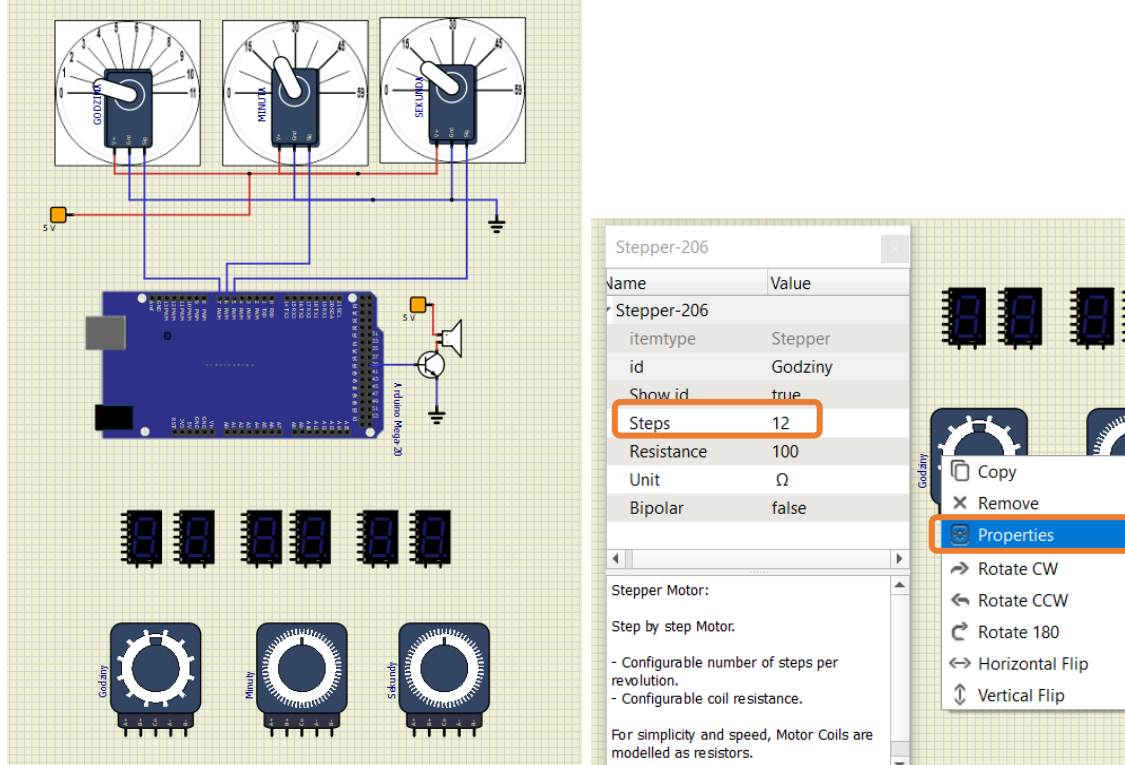


Rys. 3 Schemat przedstawiający wyświetlacze LED

Zadanie 3

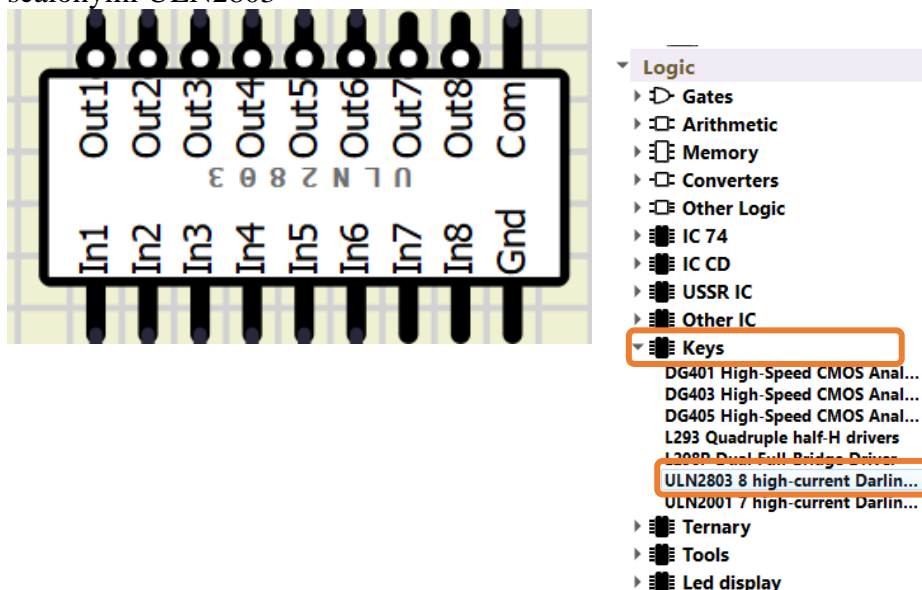
Wykorzystaj 3 silniki krokowe i podłącz je tak aby odmierzaly czas podobnie jak serwomechanizmy

Wskazówka: Aby zmienić ilość kroków dla silnika krokowego należy wybrać go prawym przyciskiem myszy i we właściwościach ustalić liczbę kroków. Na rysunku 4B przedstawiono sposób konfiguracji silnika krokowego.



Rys. 4(A) Schemat przedstawiający rozmieszczenie elementów w zadaniu 3 oraz (B) sposób konfiguracji silnika krokowego.

Wskazówka. Aby prawidłowoysterować silniki krokowe należy posłużyć się układami scalonymi ULN2803



Uwagi i wnioski:

Kod skupia się na obsłudze zegara mechanicznego z serwomechanizmami, generowaniu sygnałów dźwiękowych i wyświetlaniu czasu na wyświetlaczu siedmiosegmentowym.

Zegar jest inkrementowany w funkcji `calculateTime()` co sekundę. Serwomechanizmy dla sekund, minut i godzin są odpowiednio aktualizowane. Dźwięk jest generowany co minutę. Kąty obrotu serwomechanizmów są obliczane na podstawie aktualnego czasu, co pozwala na ich poprawne ustawienie.

Istnieje funkcja `displayTime()`, która obsługuje wyświetlanie czasu na siedmiosegmentowym wyświetlaczu. Każda cyfra jest przekazywana jako osobny argument. Dane są przekazywane przy użyciu rejestrów przesuwanych.

Funkcja `moveStepMotor()` jest odpowiedzialna za sterowanie krokomierzami silnikowymi.

Stan krokomierza jest określany przez fazę, a nie przez konkretny stan (np. HIGH/LOW). Jest to standardowe podejście przy sterowaniu krokowymi silnikami. Krokomierze silnikowe nie mają stanu zerowego, ponieważ nie ma jednoznacznej pozycji zerowej dla silnika krokowego. Silnik krokowy porusza się o określoną liczbę kroków, a jego pozycja zależy od liczby wykonanych kroków od momentu zresetowania. Dlatego nie jest możliwe ustalenie absolutnej pozycji zerowej.

W pętli głównej jest umieszczone opóźnienie 100 ms. Ma to na celu przyspieszenie działania zegara dziesięciokrotnie.