

**WYDZIAŁ INFORMATYKI, ELEKTRONIKI I TELEKOMUNIKACJI**

KATEDRA ELEKTRONIKI

**PRACA DYPLOMOWA MAGISTERSKA**

Scalony system sterowania windą

ASIC for lift control

Autor: Mateusz Dyrdół

Kierunek studiów: Elektronika i Telekomunikacja

Opiekun pracy: prof. dr hab. inż. Andrzej Kos

Kraków, 2019

**OŚWIADCZENIE**

Uprzedzony o odpowiedzialności karnej na podstawie art. 115 ust. 1 i 2 ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (t.j. Dz.U. z 2006 r. Nr 90, poz.631 z późn. zm.):„ Kto przywłaszcza sobie autorstwo albo wprowadza w błąd co do autorstwa całości lub części cudzego utworu albo artystycznego wykonania, podlega grzywnie, karze ograniczenia wolności albo pozbawienia wolności do lat 3. Tej samej karze podlega, kto rozpowszechnia bez podania nazwiska lub pseudonimu twórcy cudzy utwór w wersji oryginalnej albo w postaci opracowania, artystyczne wykonanie albo publicznie zniekształca taki utwór, artystyczne wykonanie, fonogram, wideogram lub nadanie.”, a także uprzedzony o odpowiedzialności dyscyplinarnej na podstawie art. 211 ust. 1 ustawy z dnia 27 lipca 2005 r. Prawo o szkolnictwie wyższym (t.j. Dz. U. z 2012 r. poz. 572, z późn. zm.) „Za naruszenie przepisów obowiązujących w uczelni oraz za czyny uchybiające godności studenta student ponosi odpowiedzialność dyscyplinarną przed komisją dyscyplinarną albo przed sądem koleżeńskim samorządu studenckiego, zwanym dalej „sądem koleżeńskim”, oświadczam, że niniejszą pracę dyplomową wykonałem osobiście i samodzielnie i że nie korzystałem ze źródeł innych niż wymienione w pracy.

……………………………….*podpis dyplomanta*

**Spis treści**

[Wstęp 5](#_Toc10639435)

[Cel pracy 6](#_Toc10639436)

[Rozdział 1 Technologia dźwigów osobowych 7](#_Toc10639437)

[1.1 Pierwsze systemy dźwigowe 7](#_Toc10639438)

[1.2 Strategie sterowania windą 9](#_Toc10639439)

[1.2.1 Strategia zbiorowej kontroli 9](#_Toc10639440)

[1.2.2 Strategia strefowa 10](#_Toc10639441)

[1.2.3 Strategia oparta na wyszukiwaniu 11](#_Toc10639442)

[1.2.4 Strategia oparta na przepisach 12](#_Toc10639443)

[1.2.5 Algorytmy genetyczne 13](#_Toc10639444)

[1.3 Cechy standardowej windy 14](#_Toc10639445)

[1.4 PLC a ASIC 15](#_Toc10639446)

[1.4.1 Syntezatory analogowe 15](#_Toc10639447)

[1.4.2 Syntezatory cyfrowe 15](#_Toc10639448)

[1.4.3 Syntezatory w standardzie Virtual Studio Technology 16](#_Toc10639449)

[1.5 Przepisy prawne i normy 16](#_Toc10639450)

[1.6 System przyszłości 17](#_Toc10639451)

[Rozdział 2 Opis algorytmu w języku Verilog 21](#_Toc10639452)

[2.1 Tworzenie modułów z wykorzystaniem języka Verilog 21](#_Toc10639453)

[2.2 Feature-driven development 21](#_Toc10639454)

[2.3 Testy 21](#_Toc10639455)

[2.4 Cos ram 22](#_Toc10639456)

[Rozdział 3 Projektowanie układu scalonego 23](#_Toc10639457)

[3.1 Generacja 23](#_Toc10639458)

[3.2 Synteza 24](#_Toc10639459)

[3.3 Tworzenie layoutu 25](#_Toc10639460)

[3.4 Bezpośrednia synteza cyfrowa 25](#_Toc10639461)

[3.5 Generacja sygnałów 25](#_Toc10639462)

[3.5.1 Zegar referencyjny 25](#_Toc10639463)

[3.5.2 Akumulator fazy 25](#_Toc10639464)

[3.5.3 Tablica z próbkami 27](#_Toc10639465)

[3.5.4 Mikser 27](#_Toc10639466)

[3.6 Interfejs użytkownika 28](#_Toc10639467)

[3.7 Dźwięk w programie Audacity 28](#_Toc10639468)

[Podsumowanie i wnioski 30](#_Toc10639469)

[Streszczenie 31](#_Toc10639470)

[Summary 32](#_Toc10639471)

[Spis stosowanych skrótów 33](#_Toc10639472)

[Słowa kluczowe 34](#_Toc10639473)

[Keywords 35](#_Toc10639474)

[Bibliografia 36](#_Toc10639475)

[Dodatek A. Schemat połączenia MIDI z portem szeregowym UART 37](#_Toc10639476)

[Dodatek B. Spis zawartości dołączonej płyty CD 38](#_Toc10639477)

[Spis ilustracji 39](#_Toc10639478)

[Spis tabel 40](#_Toc10639479)

# Wstęp

W tej nowoczesnej erze windy stały się integralną częścią dowolny kompleks komercyjny lub publiczny. Ułatwia to szybciej ruch osób i bagażu między piętrami. Winda system sterowania jest jednym z najważniejszych aspektów moduł sterujący elektroniki, który jest używany w motoryzacji podanie. Zwykle windy są zaprojektowane dla określonych celów budynek z uwzględnieniem głównych czynników, takich jakwysokość budynku, liczba osób podróżujących do każdego piętro i przewidywane okresy wysokiego zużycia. Winda System został zaprojektowany z różnymi strategiami sterowania. To implementacja oparta jest na FPGA, który może być wykorzystany dobudynek o dowolnej liczbie pięter, z określonymi wejściamii wyjścia. Ten kontroler można zaimplementować dlawinda o wymaganej liczbie pięter po prostuzmiana zmiennej sterującej w kodzie HDL. Takie podejście jestw oparciu o algorytm, który zmniejsza ilośćwymagane obliczenia, koncentrując się tylko na odpowiednich zasadachpoprawia to wydajność grupy windysystem. Sterownik windy został opracowany przy użyciu VerilogHDL

# Cel pracy

Praca magisterska ma na celu przedstawienie algorytmu sterowania windą i jego realizację jako układ scalony. Opis behawioralny układu cyfrowego został w języku opisu sprzętu Verilog. Sterownik bazuje na automacie skończonym, gdzie poszczególne stany są instrukcjami wykonywanymi przez windę.

Podstawowym kwestią do rozważenia w pierwszym etapie pracy jest wybranie optymalnego rozwiązania ruchu dźwigu. Główną inspiracją do opisania algorytmu jest obserwacja istniejących rozwiązań oraz usprawnienie ich działania. System musi być przyjazny dla użytkownika, sterowanie będzie odbywać się przyciskami wewnątrz i na zewnątrz windy. Kolejnym ważnym aspektem przy projektowaniu systemu sterowania winda jest energooszczędność. Szacuje się ze w niskich budynkach mieszkalnych winda jeździ sporadycznie - poniżej **5% czasu dobowego**[[1]](#footnote-1).

# Technologia dźwigów osobowych

## Pierwsze systemy dźwigowe

Prymitywne windy były używane już w III wieku pne i były obsługiwane przez ludzi, zwierzęta lub koła wodne. W 1743 r. zbudowano dla króla Ludwika XV przeciwwagę, napędzaną przez człowieka, osobistą windę, łączącą jego mieszkanie w Wersalu z mieszkaniem jego kochanki, Madame de Chateauroux, której kwatera znajdowała się piętro wyżej od króla Ludwika. Pierwsze nowoczesne windy pasażerskie nie więcej niż 150 lat temu. Windy parowe i hydrauliczne zostały już wprowadzone przez 1852, kiedy Elisha Otis stworzył jedno z najważniejszych wynalazków wind, sprzęgło, które uniemożliwił upadek windy. Następnie w 1857 r. Zainstalowano pierwszą windę pasażerską w sklepie E. Haughwout & Company, Nowy Jork. Rozwój technologii windy był szybki.

Pierwsze windy były obsługiwane przez proste urządzenia mechaniczne, takie jak sterowanie „liną ręczną” (Strakosch 1967). Pasażer mógł zadzwonić do windy, naciskając linę po obu stronach samochodu. Ponieważ szyby nie były całkowicie zamknięte, eksploatacja wind była dość niebezpieczna. Prymitywna forma sterowanie windą w jednym samochodzie opierało się na sterowanym elektrycznie przełączniku samochodu (Barney et al.1985).

Nowoczesne windy zostały opracowane w XIX wieku. Te surowe dźwigi powoli ewoluowały od napędzanej parą do mocy hydraulicznej. Pierwsze hydrauliczne dźwigi zostały zaprojektowane z wykorzystaniem ciśnienia wody jako źródła energii. Technologia silnika i metody sterowania ewoluowały szybko, a elektryczność szybko stała się akceptowanym źródłem energii. Pierwsza elektryczna winda została zbudowana przez niemieckiego wynalazcę Wenera Von Siemensa w 1880 roku. W dzisiejszych czasach nowoczesne budynki komercyjne zwykle mają wiele wind z ujednoliconym systemem sterowania. Ze względu na przeznaczenie windy dzielimy na:

* dźwigi osobowe napędzane hydraulicznie bądź elektrycznie, które służą do przenoszenia osób na różne poziomy budynków. Zwykle przemieszczają się one dość szybko i sztywno w pozycji pionowej,
* dźwigi osobowo-towarowe to z kolei urządzenie, które nadaje się zarówno do przewozu osób, jak i towaru,
* dźwigi szpitalne, zwykle przeznaczony jest do przewozu noszy, a także aparatury szpitalnej oraz pacjentów danego oddziału, czy też personelu medycznego,
* dźwigi budowlane przeznaczone do przewozu grupy robotników oraz materiałów budowlanych, które zwykle przewożone są na wysokie elewacje.

Niezależnie od przeznaczenia, najważniejszą kwestią jest sposób sterowania dźwigiem. Od systemu korbowego sterowanego przez dźwigowego aż po systemy sterowane dyspozycjami, windy stały się szybsze, bardziej zaawansowane i niezawodne w poruszaniu się między piętrami.

Współczesne systemy sterowania windą są głównie w oparciu o logikę przekaźnikową, układy PLC i mikrokontrolery, ale główny wadą tych systemów jest ich zmniejszona liczba wejść i wyjść. Można rozważyć windę jako złożony system reaktywny, który wymaga równoległego zdarzenia przetwarzanie z wieloma wejściami i wyjściami//what??

.

## Strategie sterowania windą

Przez lata opracowano wiele strategii sterowania winda. Pierwsze z nich to bardzo proste, proste strategie, ale dziś to powszechne są algorytmy wykorzystujące sztuczną inteligencję i uczenie maszynowe poprawić ruchy wind. Zbadaliśmy niektóre z nich ciekawe strategie i algorytmy, aby zobaczyć, jakie strategie istnieją.

### Strategia zbiorowej kontroli

Do najstarszych strategii należy strategia zbiorowej kontroli. To jest podstawowe strategia windy. Winda jedzie w jednym kierunku, zabierając pasażerów przechodzi obok, który ma pasażera w tym samym kierunku i upada tym razem. Kiedy po drodze nie ma więcej żądań winda jedzie, w przeciwnym razie pójdzie w innym kierunku. W przeciwnym razie zatrzyma się i będzie bezczynny, gdy upuści ostatni pasażer [5]. Jeden wadą tej strategii jest zjawisko zwane grupowaniem, gdzie kilka samochody odbierają to samo połączenie z podłogi i przyjeżdżają w podobnych godzinach, zwiększając tym samym zarówno czas oczekiwania dla pozostałych pasażerów w systemie, jak i odległość podróży wózka windy.

To zachowanie przyjmuje się jako standardowe zachowanie windy. Popularność tej strategii spowodowała . Dostęp do danych na dysków twardym odbywa się w ruchu uporządkowanym. Nadejście nowego żądania, gdy napęd jest w stanie bezczynności, zapoczątkowuje ruch ramienia w kierunku cylindra, w którym przechowywane są dane, zarówno w, jak i na zewnątrz. Dodatkowe polecenia są obsługiwane tylko w bieżącym kierunku ruchu ramienia, aż ramię osiągnie krawędź dysku. kierunek ramienia odwraca się, a żądania pozostające w przeciwnym kierunku są obsługiwane. Cykl jest powtarzany, aż wszystkie aktywnych polecenia są wykonane. Jest to opis algorytmu SCAN nazywanego również *algorytmem windy.*

**

*Rysunek 1‑1 Ruch głowicy dysku twardego zgodny z algorytmem SCAN*

Dysk twardy znacząco różni się od dźwigu osobowego. Ważne jest by zestaw tych instrukcji odnieść do działania projektowanego algorytmu windy. Najważniejszą cechą jest utrzymanie kierunku. Winda powinna się zatrzymywać na piętrach, których żądania są zgodne z obecnym kierunkiem ruchu. Jeśli istnieją inne polecenia (kierunek przeciwny), kierunek będzie zmieniony po ustalonym czasie. Ostatecznie, jeśli istnieje brak jakichkolwiek żądań, winda będzie w stanie bezczynności.

### Strategia strefowa

Strakosch opisał strategię strefy w 1983 roku. Kontrola obszaru została opisana przez Sakai & Kurosawa w 1984 r. I dotyczy kontroli strefy [5]. Sterowanie strefą jest alternatywą dla sterowania zbiorowego i jest preferowane w budynkach wiele wind. W przypadku korzystania ze strategii strefowej budynek powinien zostać rozdzielony strefy o rozmiarach zależnych od ruchu każdej strefy. Każda winda ma swoje miejsce parkingowe i tylko odbiera pasażerów z pięter wewnątrz strefy. Można to również zrobić z finału podłogi docelowe. W przypadku strategii Strefy winda ignoruje wszystkie żądania nawet poza jej strefą podczas podróży. Niektóre windy mogą mieć te same piętra w strefach i zależy to od przepływy pasażerów i wykorzystywane do poprawy podstawowej strategii [4]. W strategii podejścia strefowego, która opiera się na wspólnej strategii kontroli, każda winda ma przypisaną strefę budynku. Samochód odpowiada tylko dzwoni z tego obszaru i zatrzymuje się w strefie, gdy jest bezczynna. Jednak tak jest pozwolono na to, aby samochód wysadził pasażerów poza swoją strefą. Ta strategia ma na celu utrzymanie samochodów w oddzieleniu i uniknięcie kilku samochodów odpowiadających tym samym połączenie. Nadaje się do dużego natężenia ruchu, gdy połączenia hali są rozłożone na wszystkie strony budynek, ale jednocześnie traci dużą elastyczność [6] od samochodów nie mogą się nawzajem pokrywać. Aby stworzyć optymalną strategię, należy wybrać rozkład stref ostrożnie. Podejmując decyzję o podziale budynku tam może być kilka zmiennych do rozważenia. Możesz go podzielić w zależności na populacji pięter lub jeśli jest ważne piętro, takie jak piętro executive. Ogólnym pojęciem jest posiadanie tylu stref, ile masz windy [7].

### Strategia oparta na wyszukiwaniu

Strategia oparta na wyszukiwaniu wykorzystywała możliwe zadania, aby zoptymalizować pracę windy. To używa niektórych kryteriów, takich jak czas oczekiwania lub piętra o najwyższym priorytecie itd. Są chciwi i nie chciwe strategie wyszukiwania. Różnica między tymi strategiami została właściwie opisana w[] „Chciwe strategie wyszukiwania wykonują natychmiastowe przydzielanie połączeń, czyli przypisują hale połączenia z samochodami po ich pierwszej rejestracji i nigdy nie rozważają ponownie tych zadań. Chciwy algorytmy rezygnują z pewnej miary wydajności ze względu na brak elastyczności, ale także wymagają mniej czasu obliczeniowego. Nie-zachłanne algorytmy odkładają swoje zadania lub rozważają je w świetle zaktualizowane informacje, które mogą otrzymać na temat dodatkowych połączeń telefonicznych lub miejsc dla pasażerów. ”[] Tobita (1991) opisała następną zasadę. System wybiera kabinę, która minimalizuje czas oczekiwania, czas podróży i liczba pasażerów. System wybiera współczynniki i szacowanie funkcji. Symulacje służą do weryfikacji ich skuteczności. Po każdym zdarzeniu kontroler szuka najlepszego przypisania połączeń do hali. The słabymi stronami tego podejścia są jego wymagania obliczeniowe [5] [4].

W przeciwieństwie do opisanych powyżej algorytmów, strategia oparta na wyszukiwaniu opiera się na a algorytm wyszukiwania, a nie chciwy wariant. Przeszuka możliwości przypisania, w oparciu o pewne kryterium, wezwania do samochodu. Poszukiwanie przestrzeń to wszystkie wywołania sali wykonane w budynku i kryterium, na którym jesteś wyszukiwanie może zminimalizować średni czas podróży lub średni czas oczekiwania czas. Te typy algorytmów są używane, gdy chcesz znaleźć minimalną wartość równania matematycznego lub znaleźć element o określonych cechach wśród konkretny zbiór przedmiotów. Używając zachłannego algorytmu, możesz skrócić czas przypisywania zadzwonić do samochodu, ponieważ zachłanny algorytm natychmiast przypisuje samochód na podstawie aktualnie dostępne dane. To jest dobre dla zminimalizowania średniego oczekiwania ale nie jest elastyczny, ponieważ chciwy algorytm nigdy nie ocenia go ponownie wybór przypisania połączenia. Przeciwieństwem chciwego algorytmu jest chciwość Algorytm, który jest elastyczny i może ponownie ocenić przydziały połączeń w świetle nowych informacji ciągłych z systemu windy. Nie chciwy algorytm zajmie więcej czasu, aby zdecydować, które połączenie zostanie przypisane, a więc średni czas oczekiwania może być dłuższy, ale ogólne wyniki mogą być lepiej [6].

### Strategia oparta na przepisach

W 1998 r. Ujihara i Tsuji opisali strategię opartą na przepisach. System sterowania korzysta technologia ekspercka i logika rozmyta i oparta na zasadzie IF-THEN. Wszystkie zasady zostały opracowane przez grupę ekspertów. Wykorzystuje doświadczenie i wiedzę niezbędną do tego, aby czas oczekiwania na odstraszanie był krótszy różne warunki ruchu. Eksperci tworzą reguły oparte na porównaniu decyzji podjętych przez konwencjonalny algorytm. Wyniki są analizowane przez ekspertów, do których wykorzystano wiedzę tworzyć reguły sterowania rozmytego. Ujihara i Amano opisali zmiany w systemie. Wcześniej stosowano formułę z obliczeniami na podstawie aktualnych pozycji samochodu i połączeń. Nowy algorytm wersji bierze pod uwagę przyszłe pozycje samochodów i prawdopodobne przyszłe połączenia telefoniczne. Natychmiastowa alokacja połączeń Algorytm ma wspólne funkcje dla zachłannych algorytmów opartych na wyszukiwaniu [5] [4]. Strategia oparta na regułach opiera się na logice „warunek IF - DZIAŁANIE”. The strategia jest powszechnie stosowana w sztucznej inteligencji, ale w pewnym sensie może być stosowane do wszystkich strategii kontroli. Baza wiedzy jest tworzona z wiedzy i badania, a stamtąd kontrola windy podejmuje decyzje. Kiedy to używany jest typ strategii, który można dopasować do logiki rozmytej. Logika rozmyta to a forma wielowartościowej logiki. W logice rozmytej chodzi o użycie zmiennych może być czymś więcej niż tylko wartościami boolowskimi, na przykład zmienną temperatury może być „ciepły”, „zimny” lub „nieco powyżej zera”. Wszystkie te różne wartości zmiennej temperatury mają różne wartości prawdy, w zakresie, dla przykład od 0 do 1 [8]. Wykorzystanie tego otwiera zupełnie nowy wymiar decyzje, które muszą zostać podjęte przez system. Najgorsza część IFTHEN jest częścią IF. To tam mogą być zmienne z różnymi wartości prawdy. Na przykład jedną zasadą może być JEŻELI (wywołanie z niższej) piętro) ORAZ (wszystkie samochody spadają) NASTĘPNIE (przypisz zstępujący samochód które będą miały najlepsze wyniki w oparciu o średni czas oczekiwania w system) [9]. Gdy spojrzysz na nasz przykład, zmienna „wywołanie” jest ustawiona na „niższa piętro ”i zmienna„ samochody ”jest ustawiona na malejącą. To przez ich ocenę dwie zmienne, które dostaniesz THEN. Będziesz inny TO działanie, jeśli dwie zmienne są różne

### Algorytmy genetyczne

Istnieją algorytmy, które rozwiązują problemy poprzez naśladowanie procesu ewolucja i oparta na doświadczeniu technika algorytmów genetycznych to jedna z nich te [10]. Geny składają się z chromosomów i dziedziczymy te geny z naszego rodzice. Kiedy chcesz sztucznie naśladować proces doboru naturalnego musisz stworzyć chromosomy, od których zaczynasz. Podczas programowania algorytmy genetyczne te chromosomy są generowane losowo i każdy z nich rozwiązać problem. Chromosomy są testowane decydować o wyniku fitness, który jest liczbą wskazującą na sukces chromosom to [11]. Przy wyborze chromosomów do „krycia”, fitness Wynik jest używany do decydowania, który chromosom powinien zostać wybrany. Ponieważ wybór chromosomów jest ważony, prawdopodobieństwo jest wysokie, że dwa wybiera się chromosomy z wysoką oceną sprawności. Tak więc ludność będzie ewoluować do lepszego w każdej iteracji, tak jak w przypadku życia stworzeń. Po zastosowaniu do problemu inżynierii skutecznej kontroli windy Chromosomy opisywałyby, w jaki sposób system działałby inaczej sytuacje i chromosomy z wynikiem najbliższym zapotrzebowaniu inżynierów otrzyma najwyższą ocenę sprawności. Przez „krycie” chromosomy w każdej iteracji będą coraz lepsze złe są przesiewane. Ta cecha prowadzi do korzyści w każdej chwili możesz go zatrzymać i nadal mieć lepsze rozwiązanie niż jeden, od którego zacząłeś. Wadą algorytmu genetycznego jest to, że jest iteracyjny i dlatego powolny w wykonaniu. Podsumowując, istnieje wiele interesujących strategii, z których możesz skorzystać chcesz stworzyć system sterowania windą, a wiele z nich jest używane dzisiaj. Który z nich jest najlepszy, zależy od twoich warunków ale algorytmy genetyczne stały się popularne w ostatnich latach [11] i wiele prowadzone są badania na ten temat, który jest stosowany we wszystkich rodzajach różne obszary

## Cechy standardowej windy

Dostęp do danych na dysków twardym odbywa się w ruchu uporządkowanym. Nadejście nowego żądania, gdy napęd jest w stanie bezczynności, zapoczątkowuje ruch ramienia w kierunku cylindra, w którym przechowywane są dane, zarówno w, jak i na zewnątrz. Dodatkowe polecenia są obsługiwane tylko w bieżącym kierunku ruchu ramienia, aż ramię osiągnie krawędź dysku. kierunek ramienia odwraca się, a żądania pozostające w przeciwnym kierunku są obsługiwane. Cykl jest powtarzany, aż wszystkie aktywnych polecenia są wykonane. Jest to opis algorytmu SCAN nazywanego również *algorytmem windy.*

**

*Rysunek 1‑1 Ruch głowicy dysku twardego zgodny z algorytmem SCAN*

Dysk twardy znacząco różni się od dźwigu osobowego. Ważne jest by zestaw tych instrukcji odnieść do działania projektowanego algorytmu windy. Najważniejszą cechą jest utrzymanie kierunku. Winda powinna się zatrzymywać na piętrach, których żądania są zgodne z obecnym kierunkiem ruchu. Jeśli istnieją inne polecenia (kierunek przeciwny), kierunek będzie zmieniony po ustalonym czasie. Ostatecznie, jeśli istnieje brak jakichkolwiek żądań, winda będzie w stanie bezczynności

## PLC a ASIC

### Syntezatory analogowe

Pierwszym typem muzycznych instrumentów wykorzystujących syntezę w układach elektronicznych jest syntezator analogowy. Synteza analogowa w swojej naturze jest bardzo prosta i każdy syntezator korzysta z tej samej metody. Stosowane są w niej dwa

(Rys

### Syntezatory cyfrowe

Syntezatory cyfrowe początkowo zostały przyjęte nieufnie przez artystów ze względu na ich nowatorskie rozwiązania, wkrótce jednak zdominowały rynek muzyczny. Okazały się równie elastyczne możliwościami generacji dźwięku, jak ich analogowe odpowiedniki. Oprócz tego instrumenty miały wbudowane biblioteki brzmień imitujących instrumenty akustyczne oraz dźwięki syntezatorowe, co spowodowało wzrost zainteresowania tymi instrumentami przez eksperymentujących artystów. Ze

Syntezatory cyfrowe w obecnych czasach często generowane są przez narzędzia: PureData, jMAX, MAX/MSP czy CPS, które mają świetną elastyczność w tworzeniu metod syntezy. W tych instrumentach obliczenia są wykonywane przez mikroprocesory i syntezatory mogą wytworzyć dosłownie każdy rodzaj dźwięku wybierając metodę przez nowoczesny i przejrzysty graficzny interfejs użytkownika. Producenci cyfrowych syntezatorów w celu zwiększenia nabywców dodają analogowe komponenty do instrumentów takie jak filtry czy pętle LFO.

### Syntezatory w standardzie Virtual Studio Technology

Virtual Studio Technology jest standardem wtyczek wprowadzanym w 1996 r. przez firmę Steinberg Media Technologies. Stworzony w ten sposób syntezator istnieje tylko, jako oprogramowanie. Do obsługi oprogramowania VST niezbędny jest program bazowy nazywanym Digital Audio Workstation, które symuluje tworzenie i edycję sygnałów Serum, LennarDigital Sylenth1, Native Instruments Massive, reFX Nexus 2. Wszystkie te wtyczki są duża częścią dzisiejszej muzyki elektronicznej. Ich prostota a także ogromna ilość możliwych kombinacji w połączniu z wygodą użytkownika sprawia, że te instrumenty powoli zastępują wszystkie klasyczne syntezatory.

## Przepisy prawne i normy

Technika syntezy metoda modulacji częstotliwości nadaje się do tworzenia dźwięków harmonicznych jak i nieharmonicznych. W tej metodzie korzysta się z zestawu kilku oscylatorów, często nazywany operatorami spełniają role funkcji nośnej i modulującej. W przeciwieństwie do modulacji FM, jaką znamy z zastosowania w radiach, w syntezie FM fala nośna i modulującą są podobnej wielkości. Dla częstotliwości modulującej niższej niż 20Hz efektem są periodyczne zmiany wysokości dźwięku, podobne jak przy użyciu LFO. Przy większych częstotliwością modulujących słyszalnym efektem jest

Przez niskie wymagania obliczeń metody FM, wielu dystrybutorów instrumentów elektronicznych zaczęło wydawać produkty o osobliwych tonach. Najbardziej popularnym był syntezator Yamaha DX-7 (Rys. 1-4), którego brzmienie opisywano, jako zimne i bezosobowe. Wraz z wygaśnięciem patentu w 1995r. synteza FM stała się

## System przyszłości

W dzisiejszych miastach, gdzie dominują wieżowce, windy stały się nieodzownym środkiem transportu w życiu codziennym. Jednak coraz więcej pięter i użytkowników sprawiło, że tradycyjny system windy nie jest w stanie skutecznie rozprowadzać i przenosić użytkowników do miejsca przeznaczenia, co powoduje problemy z wydajnością transportu w budynku, takie jak długi czas oczekiwania w holu windy czy wydłużony czas podróży.

Z tego powodu w większości budynków zastosowano metody mające na celu zwiększenie wydajności transportu budynków, takie jak wdrożenie strefowego systemu windy a nawet zastosowanie dwupoziomowych wind. Wraz z rozwojem technologii doprowadziło to do stworzenia bardziej optymalnego systemu dystrybucji wind, znanego jako „wysyłka docelowa”, który w ostatnich latach stał się bardziej atrakcyjną technologią dla wind, przerywając stosowanie tradycyjnych systemów wind i metod dystrybucji.

Tradycyjne systemy wind mają przyciski wezwania na zewnątrz windy do wywoływania windy. Po tym, jak pasażer wejdzie do windy, wciska przycisk żądanego piętra. System nie może przewidzieć pięter, do których wszyscy pasażerowie muszą się udać, więc pasażerowie są zmuszeni do jazdy na różne piętra. Zwiększa to zatem czas jazdy windy, a nawet powoduje, że wszystkie windy podnoszą się i wracają do głównego piętra w tym samym czasie, bezpośrednio wpływając na czas oczekiwania i wydajność windy. Dzięki docelowemu systemowi wysyłkowemu wyeliminowano tradycyjne przyciski wywołania. Zamiast tego pasażerowie wjeżdżają do miejsca docelowego przez wybranie docelowego piętra przed szybem windy, na urządzeniu takim jak klawiatura dziesiętna lub ekran dotykowy.



*Rysunek 1‑2 Wybór piętra w systemie z wysyłką docelową*

System wysyłania miejsc docelowych polega na grupowaniu pasażerów według tego samego miejsca docelowego, gdy tylko znajdą się w miejscu przeznaczenia i przypisuje ich do tych samych wagonów windowych. Kieruje poszczególnymi wagonami windy w grupie, aby obsługiwały tylko określone piętra. Dlatego liczba przystanków jest zmniejszona, a ponieważ winda zatrzymuje się na kilku przystankach, czas podróży jest krótszy niż w konwencjonalnym systemie wind. Zasadą tego systemu jest doprowadzenie pasażerów do miejsca docelowego w możliwie najkrótszym czasie z mniejszym skupiskiem i komfortem.



*Rysunek 1‑3 Przyciski żądań w windzie z wysyłką docelową i klasycznej windzie*

Metodę alokacji można zmienić w zależności od sytuacji. Na przykład, przejście do wyższych, niższych lub sąsiednich pięter zostanie przypisane do tej samej windy, aby uniknąć nieefektywności, takich jak zatrzymanie się na piętrze gdzie nie użytkownik nacisnął zły przycisk kierunku ruchu. Niektóre systemy wind dwupokładowych, w połączeniu z tym systemem, mogą sprawić, że winda będzie bardziej elastyczna, dopasowana do potrzeb budynku.

Windy wyposażone w docelowy system wysyłkowy mają co najmniej dwie klawiatury numeryczne lub ekrany dotykowe w każdym holu, które zastępują tradycyjne przyciski żądań. Jeśli windy mają ekrany dotykowe, zazwyczaj zawierają listę dostępnych pięter obsługiwanych przez windy. Istnieją również tabliczki identyfikacyjne windy zawierające literę do identyfikacji wind (np. A, B, C itd.).



*Rysunek 1‑4 Identyfikacja wind w systemie wysyłkowym*

Pasażerowie są przypisywani do danych szybów i nie mają potrzeby wybierania pięter w środku wagonów. Pozostawiając tylko przyciski otwierania i zamykania drzwi oraz przyciski alarmowe. Przyciski pięter są ukryte za panelem i zwykle nie są używane w normalnych warunkach. Niektóre windy mogą nadal posiadać przyciski podłogowe, ale nie można ich nacisnąć; wskazują tylko piętra na których winda się zatrzymuje. Istnieją również systemy z konfiguracją hybrydową gdzie przyciski można naciskać. W takim układzie panel sterowania z wysyłką docelową znajduje się tylko na niektórych określonych piętrach a na pozostałych piętrach obowiązuje układ konwencjonalnej windy.

# Opis algorytmu w języku Verilog

## Tworzenie modułów z wykorzystaniem języka Verilog

Język Verilog HDL zawiera funkcje opisujące charakter projektu, przepływ danych, skład strukturalny, opóźnienia i mechanizm generowania przebiegów, w tym aspekty monitorowania odpowiedzi i weryfikacji, wszystkie modelowane przy użyciu jednego języka. Ponadto język zapewnia interfejs języka programowania, dzięki któremu można uzyskać dostęp do elementów wewnętrznych projektu podczas symulacji, w tym do sterowania przebiegiem symulacji. Język nie tylko definiuje składnię, ale także definiuje bardzo jasną semantykę symulacji dla każdego konstruktu językowego. Dlatego modele napisane w tym języku można zweryfikować za pomocą symulatora Verilog. Język dziedziczy wiele symboli operatora i konstruktów z języka programowania C. Verilog HDL zapewnia szeroki zakres możliwości modelowania, z których niektóre są dość trudne do zrozumienia na początku. Jednak podstawowy podzbiór języka jest łatwy do opanowania i użycia. Jest to wystarczające do modelowania większości aplikacji.

Narzedia???

## Feature-driven development

## Testy

Testy strukturalne – white box

 *Statement Coverage* – pokrycie kodu, znane również jako pokrycie linii lub segmentu. Określa, które fragmenty programu pokrywają się ze sobą, czyli zostały wywołane przez zestaw testowy oraz te, których nie udało się wywołać. *Statement Coverage* pozwala w prosty sposób zidentyfikować obszar niewytestowanego kodu.

 *Branch Coverage* – w języku programowania jest zbliżony do instrukcji IF, która posiada dwa „odgałęzienia”: Prawdę i Fałsz (*True and False*). W zasięgu gałęzi (zwanej również „zasięgiem decyzji”) sprawdza się, czy każde z nich jest wykonywane co najmniej raz. W przypadku instrukcji IF będą występowały dwa warunki, jeden do sprawdzenia prawdziwego oddziału i pozostałe, by sprawdzić te fałszywe. Ta metoda zapewnia, że po uruchomieniu każda gałąź z decyzyjnego punktu widzenia jest wykonywana.

## Cos ram

# Projektowanie układu scalonego

## Generacja

Zestaw uruchomieniowy na płytce MAXimator (Rys. 3-1) jest świetnym rozwiązaniem nie tylko dla poczatkujących elektroników, ale również dla profesjonalistów i hobbystów. Cechuje się nowoczesnym układem firmy Altera, należącym do rodziny MAX10 o dużych zasobach logicznych. Najważniejsze cechy dostępne dla użytkownika to:

* ponad 8000 komórek LE,
* 378kb konfigurowalnej pamięci,
* 1376kb wewnętrznej pamięci Flash dla aplikacji,
* wbudowany 12-bitowy przetwornik A/C 1MSpS,
* wewnętrzny generator sygnału zegarowego,
* 2 wewnętrzne wielokanałowe syntezery PLL.

*Rysunek 3‑1 Wygląd płytki bazowej MAXimator*

Zalety układów MAX10 są bardzo dobrze wyeksponowane w zestawie MAXimator. Budowa płytki konstrukcyjnie przypomina Arduino Uno Rev.3 co pozwala na różnorodne zastosowanie, również w celach edukacyjnych. W skład zestawu MAXimator wchodzą:

* płytka MAXimator z układem 10M08,
* programator zgodny z USB Blaster,
* ekspander z 4-cyfrowym wyświetlaczem LED, analogowym sensorem temperatury, trzema przyciski i dwiema diodami LED RGB.

W odróżnieniu od zestawów Arduino, MAXimator nie ma wbudowanego programatora, lecz korzysta z dołączonego do zestawu USB Blastera (Rys. 3-2), który pozwala na zaprogramowanie dostarczonego układu naszym kodem przez środowisko Quartus Prime. Kolejną atrakcją zestawu jest ekspander (Rys. 3-3) mający szeroką gamę funkcjonalności dla projektantów. Wyposażony w cztero-pozycyjny, siedmio- segmentowy wyświetlacz LED sterowany multipleksowo, dwie diody RGB, czujnik temperatury oraz trzy przyciski: jeden przycisk *RESET* a pozostałe dwa do dowolnego wykorzystania przez użytkownika.

## Synteza

Cały projekt jest przystosowany do współczesnych standardów wraz z funkcją obsługi przez użytkownika. Syntezer składa się z:

* dekodera MIDI,
* sterownika oscylatorów,
* dwóch oscylatorów,
* miksera sygnałów,
* obsługi interfejsu użytkownika.

Schemat blokowy architektury projektu widnieje na rysunku 3.2.1. Wszystkie komponenty są sterowane narastającym zboczem zegara oraz negatywnym stanem resetu. Weryfikacja poprawnej działalności modułów następuje przez symulator ModelSIM Altera.

*Rysunek 3‑2 Architektura syntezera*

Rozwój architektury odbywał się na równoległym poziomie z testowaniem zaimplementowanych funkcji. W czasie tworzenia cyfrowego syntezatora opartego na DDS jedyną trudność sprawiła implementacja akumulatora fazy i zrozumienie jego działania.

## Tworzenie layoutu

Korzystanie z interfejsu MIDI jest praktycznym zabiegiem dla współczesnych syntezatorów. W implementacji, dekoder MIDI otrzymuje wejściową ramkę i przesyła dwie informacje: start dźwięku od danym tonie i stop dźwięku. Blok dekodera został przedstawiony na rysunku 3-5.

## Bezpośrednia synteza cyfrowa

Metoda bezpośredniej syntezy cyfrowej to technika generacji sygnałów, oparta w całości na cyfrowych metodach syntezy sygnału. Układy DDS stały się alternatywą dla analogowych syntezerów opartych na pętlach sprzężenia fazowego, zapewniając szybkie przestrajanie w szerokim zakresie częstotliwości oraz regulację amplitudy. Generatory korzystające z bezpośredniej syntezy cyfrowej charakteryzują się dokładnością w tworzeniu sygnałów dzięki dostępowi do pamięci, gdzie zawierają się próbki oraz specjalnego mechanizmowi zegarowemu, który różni się od tradycyjnej metody.

*Rysunek 3‑3 Blok bezpośredniej syntezy cyfrowej*

## Generacja sygnałów

*Rysunek 3‑4 Schemat blokowy oscylatora DDS*

### Zegar referencyjny

Zegar referencyjny ma za zadania wytworzenie słowa przestrajającego, aktualizowanie wartość akumulatora fazy oraz kieruje konwersją cyfrowo-analogową. Zegar referencyjny determinuje, kiedy próbka jest przesyłana w przetworniku, ale nie ma

### Akumulator fazy

Sercem każdego syntezatora korzystającego z bezpośredniej syntezy cyfrowej jest akumulator fazy. Ten komponent jest zbiorem słów przestrajających(ang. tuning word), które pozwalają na dokładne wytworzenie częstotliwości sygnału. Każde z tych słów oznacza skok między próbkami. Im większy skok tym więcej wartości jest pomijanych,co skutkuje szybszym pokonaniem koła fazy cyfrowej przedstawiony na rysunku 3-8. Jedno okrążenie wokół koła cyfrowego oznacza jeden okres dla sygnału. Największe

*Rysunek 3‑5 Koło fazy cyfrowej*

Moduł *NOTE\_TO\_PA* przedstawiony na rysunku 3-7 jest odpowiednikiem tablicy przekształcenia numeru danego dźwięku na skok akumulatora fazy. Wartości w tej tablicy są liczone według wzoru 3.5.1.

Do obliczenia częstotliwości generowanego sygnału służy wzór:

fOUT – częstotliwość wyjściowa,

M – słowo przestrajające,

fC - częstotliwość zegara referencyjnego,

n- rozdzielczość akumulatora fazy.

*Rysunek 3‑6 Analiza czterokrotnego pomnożenia kroku*

W celu przybliżenia generacji sygnału przy pomocy koła fazowego niżej przedstawiono graficznie jak wartości na kole fazowym odpowiadają wartościom napięć. Przed zapisywaniem sygnału sinusoidy, powstają próbki a po zakończeniu generacji sygnału cyfrowego, następuje konwertowanie do sygnału analogowego przez przetwornik.

*Rysunek 3‑7 Generacja sinusoidy*

Dodatkowym elementem w akumulatorze fazy jest rejestr fazy. Jest to operacja zmniejszania rozdzielczości akumulatora fazy. Duża rozdzielczość akumulatora fazy jest wymagana do otrzymania mały odstępów częstotliwość na każdy skok(około 0,006Hz na skok dla implementowanego syntezera). Rejestr fazy pozwala na użycie mniejszych tablic niż rozdzielczość akumulatora fazy. Ten zabieg oszczędzą miejsce w pamięci, jego początkiem był fakt nierozwiniętej w takim stopniu technologii do trzymania tak dużej ilości danych. W ten sposób możemy regulować dokładność generowanego przebiegu bez ingerencji w efektywność akumulatora fazy.

### Tablica z próbkami

Implementowany syntezator zawiera cztery różne pamięci tablicowe. Każda z nich zawiera jeden okres przebiegu o danym kształcie, co pozwala na dokładne odtworzenie sygnału przez akumulator fazy. Sygnał z wyjścia rejestru fazy determinuje wybór próbki z tablicy. Tablica składa się z 256 wartości, które zawierają kształt danego przebiegu. Wyjątkiem jest tablica przebiegu prostokątnego, która zawiera tylko dwie wartości – amplitudę stanu niskiego i wysokiego. Wartość wejściowa z rejestru fazy oznacza wybór danej wartości z tablicy próbek a następnie ta wartość próbki jest przesyłana do instancji miksera. Przykładowa generacja sygnału z wykorzystaniem tablicy została przedstawiona na rysunku 3-11.

*Rysunek 3‑8 Generacja sinusoidy z wykorzystaniem tablicy*

Wartości próbek w tablicach są zapisane w formacie 16-bitowych zmiennych ze znakiem. Przedział amplitud został ograniczony w przedziale od -32000 do 32000.

Użycie akumulatora fazy oraz metody tablicowej generuje dokładne przebiegi. Wady i zalety metody tablicowej zostały opisane w punkcie 1.4.1 *Synteza z użyciem tabeli przebiegu fali.*

### Mikser

Połączenie sygnałów z kilku oscylatorów wymaga użycia miksera. Mikser kontroluje wyjściową wartość amplitudy, aby sygnał nie uległ nasyceniu przez duże wartości amplitud kilku sygnałów.

*Rysunek 3‑9 Blok miksera*

Moduł miksera został zbudowany w oparciu o istniejące przykłady. Wszystkie sygnały z oscylatorów są przekazywane a użytkownik korzystając z przycisków zmienia wartość sygnału *WAV\_SELECT*. Dane wyjściowe są kierowane do przetwornika lub interfejsu audio.

## Interfejs użytkownika

Użytkownik ma umożliwiona kontrolę na generowanym sygnałem. Korzystając z przycisków oraz wyświetlacza na ekspanderze można wybrać kształt fali na wyjściu oraz liczbę użytych oscylatorów. Przyciski są wyposażone w debouncer, który zatrzaskuje się po naciśnięciu na jedną sekundę, dzięki czemu mikrokontroler nie ma problemów z odczytywaniem zamiarów użytkownika. Liczba użytych wyświetlaczy została zredukowana do jednego, ze względu na wybór mniej niż dziesięciu funkcji syntezatora. Ewentualna rozbudowa architektury projektu nie przeszkadza w użyciu pozostałych wyświetlaczy.

## Dźwięk w programie Audacity

Z powodu problemów generacji dźwięku interfejsem HDMI, osiągniecie rezultatów osiągnięto przez symulacje, zapis wygenerowanych próbek oraz odtworzenie pliku pcm w darmowym edytorze plików dźwiękowych Audacity.

*Rysunek 3‑10 Sygnały wygenerowane w Audacity*

Generacja sygnału w tym programie odbywa się poprzez importowanie pliku pcm. Końcowe rezultaty potwierdzają poprawne działanie syntezatora. Do wykonania poprawnej generacji należy użyć widoku listu, przeciągnąć wygenerowany sygnał do okna listy i zapisać do pliku.

Generowanie dźwięku odbywa się w trzech krokach:

* zapis danych wyjściowych z symulacji,
* konwersja z formatu hex do ASCII i zapis pliku z rozszerzeniem pcm,
* import danych z powyższymi ustawieniami.

Ustawienia do generowania dźwięku w Audacity:

* kodowanie – bez znaku 16-bit PCM,
* kolejność bajtów – big endian,
* kanały – 1 kanał (mono),
* offset – 0 bajtów,
* ilość importowanych bajtów – 100%,
* częstotliwość próbkowania – 44100 Hz.

Wygenerowane sygnały są eksportowane do formatu wav, który umożliwia odsłuchanie próbki. Otrzymane wyniki są zgodne z założeniami i potwierdzają prawidłowe działanie syntezera. Na tym etapie prace nad syntezatorem zostają zakończone.

# Podsumowanie i wnioski

Implementacja syntezatora została zakończona sukcesem. Niestety jedno z głównych założeń, czyli generacja dźwięku przy użyciu interfejsu HDMI nie zostało spełnione. Liczba prób realizacji tego założenia oraz czas włożony w rozwój projektu pod tym względem pozostawia możliwość ukończenia sterowania interfejsem HDMI w prawdopodobnie krótkim czasie. Potencjał płytki MAXimator nie został całkowicie wykorzystany, co pozostawia chęć do zbudowania syntezatora o większej funkcjonalności.

W projekcie wykorzystano układ FPGA jako generator sygnałów pracujący w trybie bezpośredniej syntezy cyfrowej. Istnieje duża możliwość rozbudowania projektu o większą ilość sygnałów, dodanie filtru lub efektów dźwiękowych, które ze zostały zaimplementowane z uwagi na. Udało się zaobserwować wygenerowane sygnały oraz przetworzyć do formatu .wav oraz je odsłuchać. Na chwilę obecną syntezator jest narzędziem czysto programowym ale już są planowane pracę nad uruchomieniem projektu wraz z interfejsem HDMI lub innym dostępnym kodekiem.

Projekt w układzie FPGA zajmuje:

* 1412 logicznych elementów (18% całości),
* 387 rejestrów,
* 11 z 178 dostępnych pinów,
* jeden moduł PLL z dwóch dostępnych.

# Streszczenie

Tematem pracy jest syntezator dźwięku w układzie FPGA. Projekt został zbudowany w środowisku Intel Quartus Prime z wykorzystaniem języku opisu sprzętu VHDL. Jednym z głównych założeń projektu była generacja dźwięku przez interfejs HDMI. Niestety po wielu próbach, prace nad uruchomieniem HMDI zakończyły się niepowodzeniem ale projekt będzie kontynuowany celem osiągniecia zamierzonych rezultatów.

Syntezator jest instrumentem muzycznym, generatorem sygnałów akustycznych. Dogłębna analiza tematu współczesnych syntezatorów wskazała na użycie interfejsu MIDI do komunikacji między urządzeniem a innymi instrumentami.

Użytkownik obsługuje instrument przy użyciu przycisków znajdujących się na płycie ekspandera dołączonej do zestawu MAXimator. Funkcja syntezatora zostaje wyświetlona na płytce, tak by osoba sterująca miała pełną kontrolę nad urządzeniem.

W pracy została zawarta rozprawa na temat różnic w budowie i funkcjonalności syntezatorów a także nacisk na funkcjonalność i dopasowanie metod syntezy do cyfrowego układu FPGA. Ostatecznie w projekcie wykorzystano metodę syntezy bezpośredniej z wykorzystanie tablic pamięci do generacji przebiegów. Satysfakcjonujące wyniki otrzymano przez symulacje i generacje zapisanych próbek do formatu wav.

# Summary

The subject of the diploma is a sound synthesizer in FPGA. The project was built in the Intel Quartus Prime environment using the VHDL hardware description language. One of the main assumptions of the project was the generation of sound via the HDMI interface. Unfortunately, after many attempts, work on starting the HMDI was unsuccessful. Despite the lack of results, the project will be continued in order to achieve the intended results.  
 Synthesizer is a musical instrument, a generator of acoustic signals. After depth analysis of the topic of modern synthesizers indicated the use of MIDI interface for communication between the device and other instruments.  
 The user operates the instrument using the buttons on the expander board attached to the MAXimator kit. The synthesizer function is displayed on the board so that the control person has full control over the device.  
 The work included a dissertation on the differences in the structure and functionality of synthesizers, as well as the emphasis on functionality and adaptation of synthesis methods to the digital FPGA system. Finally, the project used a direct synthesis method with the use of memory tables for generation of waveforms. Satisfactory results were obtained by simulations and generations of saved samples to the wav format.

# Spis stosowanych skrótów

ARMA – Autoregresja i ruchoma średnia

ADSR – Attack Decay Sustain Release

DDS – Direct Digital Synthesis

FPGA – Field Programmable Gate Array

HDMI – High Definition Multimedia Interface

LFO – Low Frequency Oscillator

MIDI – Musical Instrument Digital Interface

NCO - Numerically Controlled Oscillator

UART –Universal Asynchronous Receiver Transmitter

VCO – Voltage Controlled Oscillator

VST – Virtual Studio Technology

# Słowa kluczowe

* ASIC,
* syntezator,
* bezpośrednia synteza cyfrowa,
* synteza częstotliwości,
* generacja dźwięku,
* MAXimator,
* Altera MAX10.

# Keywords

* FPGA,
* synthesizer,
* direct digital synthesis,
* frequency synthesis,
* sound producing,
* MAXimator,
* Altera MAX10.

# Bibliografia

1. Antoszkiewicz K., *Generacja i synteza częstotliwości*, Warszawa: Oficyna Wydawnicza Politechniki Warszawskiej, 2015, str. 260-271.
2. Bhasker J., *A VHDL Primer,* Upper Saddle River: Prentice Hall PTR, 1999.
3. Naylor D., Jones S., *VHDL: a logic synthesis approach,* London: Chapman & Hall, 1997.
4. Pellegrino R., *The Electronic Arts of Sound and Light*, New York: Van Nostrand Reinhold Company, 1983.
5. Vankka J., Halonen K., *Direct digital synthesizers: theory, design and applications,* Boston: Kluwer Academic Publ., 2001.
6. Analog Devices, [Ask The Application Engineer – 33: All About Direct Digital Synthesis](http://www.analog.com/en/analog-dialogue/articles/all-about-direct-digital-synthesis.html,%20), [http://www.analog.com/en/analog-dialogue/articles/all-about-direct-digital-synthesis.html,](http://www.analog.com/en/analog-dialogue/articles/all-about-direct-digital-synthesis.html,%20) stan na dzień: 05.01.2018
7. All About Circuts, [Everything You Need to Know About Direct Digital Synthesis](https://www.allaboutcircuits.com/technical-articles/direct-digital-synthesis/)*,* <https://www.allaboutcircuits.com/technicalarticles/direct-digital-synthesis/>*,* stan na dzień: 05.01.2018

https://vignette.wikia.nocookie.net/elevation/images/a/aa/BigTransit.jpg/revision/latest?cb=20130115104717&format=original

http://www.neii.org/destdispatch.cfm

<https://www.tripadvisor.com.au/LocationPhotoDirectLink-g53449-d294369-i310683750-Cathedral_of_Learning-Pittsburgh_Pennsylvania.html>

<https://elevation.fandom.com/wiki/Destination_dispatch>

<https://www.retrofitmagazine.com/destination-dispatch-system-improves-buildings-elevator-performance-and-efficiency/>

<http://global.ctbuh.org/resources/papers/download/399-elevator-traffic-simulation-procedure.pdf>

https://link.springer.com/content/pdf/10.1023%2FA%3A1007518724497.pdf

# Dodatek A. Schemat połączenia MIDI z portem szeregowym UART



# Dodatek B. Spis zawartości dołączonej płyty CD

All About Direct Digital Synthesis \_ Analog Devices.htm – strona internetowa zawarta w bibliografii na pozycji 8,

Everything You Need to Know About Direct Digital Synthesis.htm – strona internetowa zawarta w bibliografii na pozycji 9,

Mateusz\_Dyrdół\_276528.doc – Tekst pracy zapisany w formacie MS Word,

Mateusz\_Dyrdół\_276528.pdf – Tekst pracy zapisany w formacie rozszerzonym,

synth.qar – Archiwum projektu wygenerowane w środowisku Intel Quartus Prime,

sine.pcm – plik w rozszerzeniu pcm do generacji sygnału sinusoidalnego,

saw.pcm – plik w rozszerzeniu pcm do generacji sygnału piłokształtnego,

triangle.pcm – plik w rozszerzeniu pcm do generacji sygnału trójkątnego,

sine.wav - plik dźwiękowy z sygnałem sinusoidalnym,

saw.wav - plik dźwiękowy z sygnałem piłokształtnym,

triangle.wav - plik dźwiękowy z sygnałem trójkątnym.

# Spis ilustracji

[Rysunek 1‑1 Ruch głowicy dysku twardego zgodny z algorytmem SCAN 9](#_Toc10206840)

[Rysunek 1‑2 Wybór piętra w systemie z wysyłką docelową 14](#_Toc10206841)

[Rysunek 1‑3 Przyciski żądań w windzie z wysyłką docelową i klasycznej windzie 15](#_Toc10206842)

[Rysunek 1‑4 Identyfikacja wind w systemie wysyłkowym 16](#_Toc10206843)

[Rysunek 3‑1 Wygląd płytki bazowej MAXimator 19](#_Toc10206844)

[Rysunek 3‑2 Architektura syntezera 20](#_Toc10206845)

[Rysunek 3‑3 Blok bezpośredniej syntezy cyfrowej 21](#_Toc10206846)

[Rysunek 3‑4 Schemat blokowy oscylatora DDS 21](#_Toc10206847)

[Rysunek 3‑5 Koło fazy cyfrowej 22](#_Toc10206848)

[Rysunek 3‑6 Analiza czterokrotnego pomnożenia kroku 22](#_Toc10206849)

[Rysunek 3‑7 Generacja sinusoidy 22](#_Toc10206850)

[Rysunek 3‑8 Generacja sinusoidy z wykorzystaniem tablicy 23](#_Toc10206851)

[Rysunek 3‑9 Blok miksera 23](#_Toc10206852)

[Rysunek 3‑10 Sygnały wygenerowane w Audacity 24](#_Toc10206853)

# Spis tabel

[***Tabela 1 Przyporządkowane piny do sygnałów* 25**](#_Toc502937456)

[***Tabela 2 Tablica częstotliwości dla piątej oktawy* 27**](#_Toc502937457)

1. Norma energtyczna [↑](#footnote-ref-1)