
Wtyczki audio i wirtualne instrumenty

ZARZYCKI GRACJAN 19433

4 IV 2023

Spis treści

I.	Wprowadzenie	3
1.	Inżynieria audio	3
2.	Z czego składa się dźwięk?	3
3.	Rodzaje sygnałów	4
4.	Przetwornik A/C w skrócie	5
II.	Wtyczki audio	6
1.	Czym są wtyczki audio?	6
2.	Rodzaje wtyczek audio	6
3.	Projektowanie, technologie i standardy	6
4.	Informatyk w branży muzycznej.....	8
III.	Wirtualne instrumenty	8
1.	Czym są wirtualne instrumenty?	8
2.	Rodzaje wirtualnych instrumentów	9
IV.	Zastosowania	9
1.	Jak było kiedyś?	9
2.	Jak jest dzisiaj?	9
3.	Jak może być?	9
V.	Źródła.....	10

I. Wprowadzenie

1. Inżynieria audio

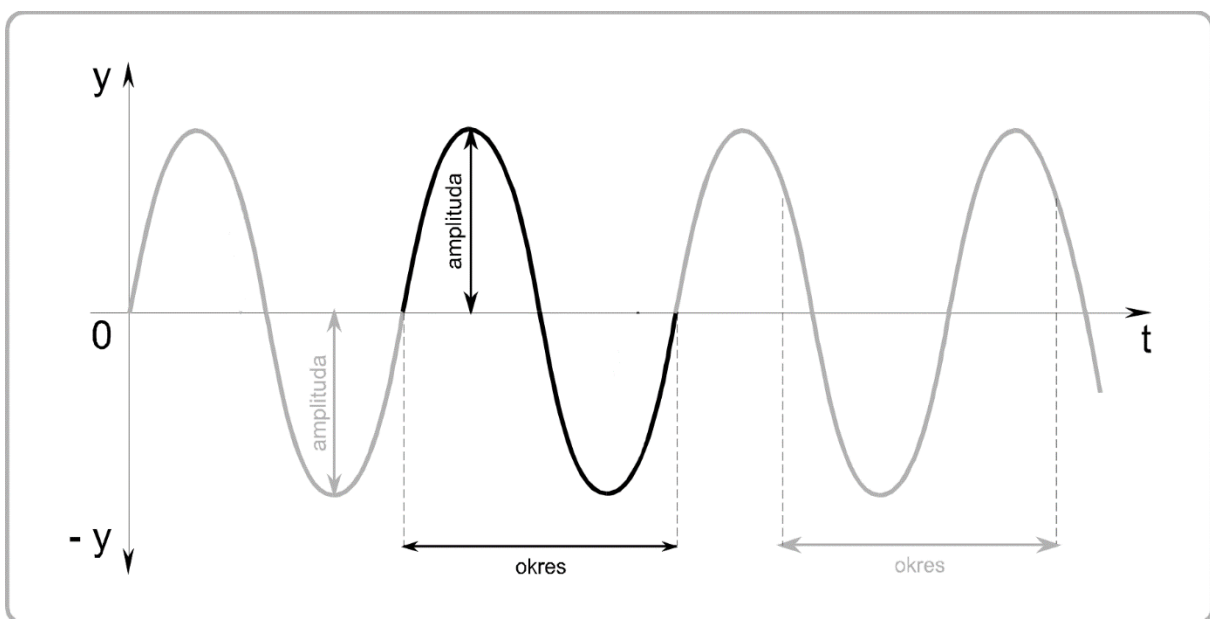
Inżynieria audio to dziedzina zajmująca się projektowaniem i ogólnym poprawianiem jakości dźwięku. Jest to jedna z branż, która ma wpływ na nasze życie codzienne, oraz z łatwością znajduje zastosowanie w wielu dziedzinach m. in. rozrywka, komunikacja, medycyna lub edukacja. Dzięki rozwijającej się technologii przetwarzania sygnałów (o czym będzie dalej), inżynierowie audio wpływają na komfort życia od prostego słuchania muzyki, aż do specjalistycznych badań i diagnoz medycznych.

2. Z czego składa się dźwięk?

Można by zacytować wiele rozmaitych definicji z dziedziny fizyki, aby dokładnie wyjaśnić pojęcie dźwięku. Jednak chciałbym podejść do tego bardziej „po ludzku i informatycznie”. Za przykład obierzmy dobrze wszystkim znaną falę sinusoidalną, która niestety (w kontekście fali dźwiękowej) nie występuje w naturze, ale to z takich, nałożonych na siebie fal, składa się dźwięk. Podstawowymi parametrami tej fali są:

Częstotliwość – jest określana przez ilość cykli występujących w jednej sekundzie drgania fali, gdzie cykl to jedna pełna zmiana wartości fali (Rys. 1 *pogrubiona linia*). Na przykładzie, jeśli w fali występuje 450 cykli w sekundzie uzyskamy falę o częstotliwości 450Hz.

Okres fali - to czas potrzebny do wykonania jednego cyklu fali sinusoidalnej (Rys. 1).



Rys. 1 Okres, cykl i amplituda fali sinusoidalnej

Amplituda – określa głośność dźwięku (Rys. 1)

A więc w końcu nadszedł czas, aby odpowiedzieć na pytanie „Z czego składa się dźwięk?”. Wspomniałem, że czysta fala sinusoidalna nie występuje w naturze... i to prawda. Jednak z większej ilości takich fal nałożonych na siebie składa się muzyka, dźwięk i nawet nasz głos. Dokładnie złożony jest z różnie drgających fal o różnej częstotliwości i różnej amplitudzie. Co powoduje nakładanie się fal?

Harmonia - w muzyce to zjawisko polegające na brzmieniu kilku dźwięków jednocześnie, których częstotliwości są ze sobą w stałych proporcjach. W kontekście fali dźwiękowej informatycznej harmonia może być generowana poprzez dodanie do siebie kilku sinusów o różnych częstotliwościach, które tworzą akord.

Brzmienie - to charakter dźwięku, który pozwala odróżnić dwa dźwięki o takiej samej częstotliwości i amplitudzie. Brzmienie zależy od wielu czynników, takich jak kształt fali dźwiękowej, obecność harmoniczných częstotliwości składowych, charakterystyka filtra (np. w przypadku syntezatora), a także od sposobu, w jaki dźwięk został nagrany lub zarejestrowany. W fali dźwiękowej, informatycznej, brzmienie może być generowane przez odpowiednie manipulacje sygnałem, takie jak dodanie efektów przetwarzających (efekty dźwiękowe).

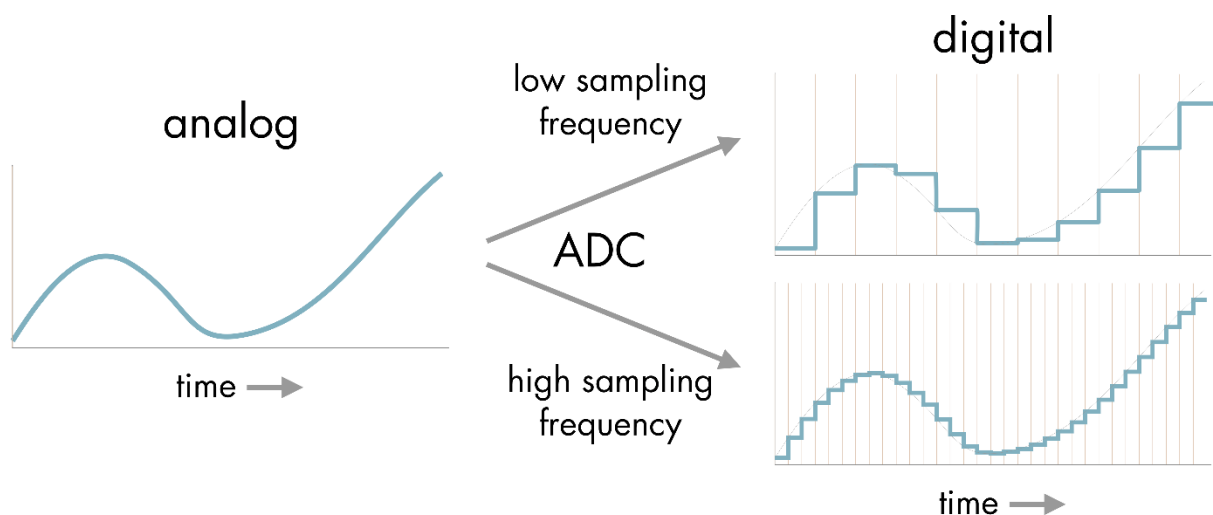
3. Rodzaje sygnałów

Myślę, że nie zagłębiając się w tematy z zakresu fizyki, psychoakustyki i biologii, powyższe proste wyjaśnienie zupełnie wystarczy nam do zrozumienia informatycznych aspektów dźwięku i tego jak jest przetwarzany. Aktualnie żyjemy w świecie dźwięku cyfrowego. Oczywiście nie możemy zapominać o ciągłym, nieprzerwanym sygnale (stosowanym m. in. na winylu). Bez zbędnych definicji, wyjaśnimy różnicę sygnałami (Rys.1.1):

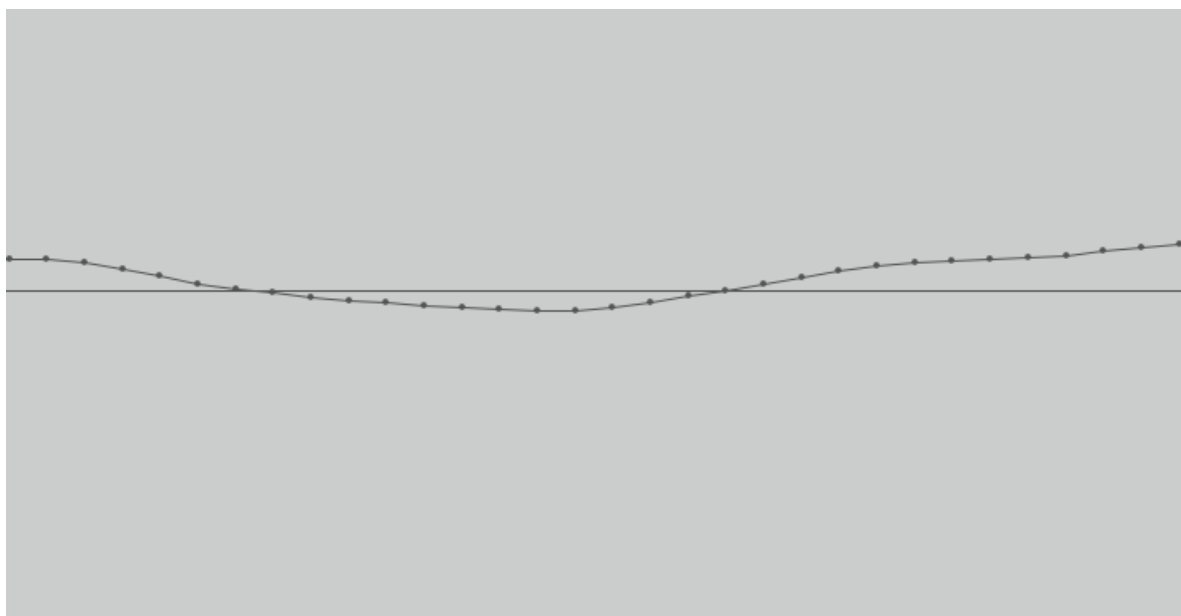
Analogowy – jest ciągłą falą. To znaczy, nie można w nim wyodrębnić pojedynczych momentów dźwięku np. zmian wysokości. W muzyce wszystko co nagramy, za pomocą metody analogowej, zostanie zarejestrowane.

Cyfrowy – to reprezentacja cyfrowa dźwięku, czyli fali akustycznej, która jest przenoszona przez powietrze lub inne ośrodki. Dźwiękowy sygnał cyfrowy jest reprezentowany przez sekwencję wartości numerycznych. Za pomocą prostego języka maszynowego, intensywność i wysokość dźwięku jest reprezentowana przez 0 i 1 w precyzyjnych odstępach czasu. Kod informuje komputer, jak ma odwzorować dany dźwięk. Sygnał cyfrowy nie jest ciągłą, pojedynczą falą. Jest to zlepek punktów (próbek)(Rys.1.2.) w jednej sekundzie nagrania, z określoną częstotliwością próbkowania.

Dobłą analogią do zrozumienia różnicy między sygnałem analogowym, a cyfrowym jest porównanie ich do światła żarówek. Sygnał analogowy jest zwykłą żarówką, która świeci nieprzerwanym światłem. Sygnał cyfrowy jest żarówką stroboskopową. Z małą częstotliwością z łatwością zauważymy kiedy światło zapala się, a kiedy gaśnie. Jednak zwiększając częstotliwość, w którymś momencie stroboskop (pozornie) zacznie świecić nieprzerwanym strumieniem światła. Tak samo sygnał cyfrowy, odwzorowuje analogowy bazując na częstotliwości próbkowania (sample rate / sampling frequency).



Rys. 1.1. Konwersja analogowo-cyfrowa (różnica między sygnałami)



Rys. 1.2. Wykres próbek sygnału cyfrowego

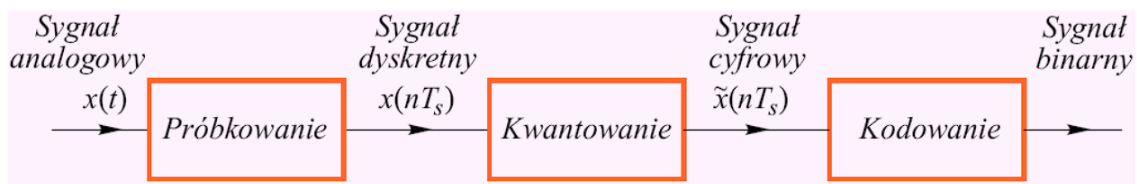
4. Przetwornik A/C w skrócie

Przetwornik analogowo-cyfrowy, jak można wywnioskować, służy do przetwarzania sygnału analogowego na cyfrowy. Oczywiście przetworniki takie znajdują zastosowanie w wielu dziedzinach takich jak medycyna, przemysł, czy automatyka. Jednak, aby przedstawić w jaki sposób sygnał jest przetwarzany, za przykład obiorę nagrywanie dźwięku za pomocą mikrofonu.

W celu rejestrowania dźwięku, mikrofon zamienia falę dźwiękową na sygnał elektryczny, który początkowo jest w postaci sygnału analogowego. Po tym sygnał trafia do przetwornika A/C, w którym zachodzą kolejno procesy (Rys.1.3.):

1. *Próbkowanie*, czyli pobieranie próbki sygnału (jednego pojedynczego zadrżania membrany mikrofonu) w określonych odstępach czasu,
2. *Kwantowanie*, czyli przekształcenie próbki na wartość cyfrową (nadanie próbce jakiejś wartości),

3. *Kodowanie*, czyli zapisanie uprzednich wartości na pamięci urządzenia (np. karty dźwiękowej lub interfejsu audio).



Rys. 1.3. Procesy powstawania sygnału cyfrowego

Oczywiście jakość uzyskanego sygnału cyfrowego zależy od wielu czynników. Skupiając się tylko na przetworniku mogą nimi być: częstotliwość próbkowania, liczba bitów, zakres dynamiczny itd. (są to parametry określające jakość uzyskanego sygnału).

II. Wtyczki audio

1. Czym są wtyczki audio?

Wtyczki audio, a dokładnie wtyczki VST, powstały, działają i ciągle rozwijają się dzięki coraz to nowszej technologii i „przejściu” na ww. sygnał cyfrowy. Są to programy komputerowe, których zadaniem jest przetwarzanie i/lub generowanie dźwięku w środowisku cyfrowym. Jest to pewnego rodzaju dodatek do oprogramowania do produkcji muzyki (obierając ją za przykład – więcej branż korzysta z tego typu wtyczek i cyfrowego przetwarzania sygnału).

2. Rodzaje wtyczek audio

Aby bardziej sprecyzować ich działanie, należy wyróżnić najpopularniejsze grupy wtyczek audio:

1. Efekty dźwiękowe – działają one na sygnale po wejściu do programu, na którym pracujemy. Zadaniem ich jest przetwarzanie sygnału audio np. wzmacnianie, pogłos. W skrócie powodują zmianę brzmienia dźwięku.
2. Narzędzia edycyjne – działają na tej samej zasadzie co efekty dźwiękowe, jednak służą one do „oczyszczania” sygnału. Pozwalają na np. wyrównywanie partii lub manipulacje poszczególnymi częstotliwościami.
3. Analizatory – pozwalają na dokładną analizę charakterystyk dźwięku/sygnału np. spektrum, amplitudy lub wykrywanie częstotliwości.

3. Projektowanie, technologie i standardy

Projektowanie wtyczek jest procesem tworzenia programów komputerowych, które pozwalają na przetwarzanie sygnałów audio w czasie rzeczywistym. Technologie wykorzystywane we wtyczkach obejmują: przetwarzanie cyfrowe sygnału, matematyczną analizę sygnału audio oraz algorytmy przetwarzania sygnału.

Wtyczki audio mogą być bardzo złożone i składać się z tysięcy linii kodu, w zależności od funkcjonalności. Jednak chciałbym pokazać, że każdy początkujący programista jest w stanie stworzyć proste efekty na sygnale. Przykładem będzie prosta, darmowa wtyczka implementująca opóźnienie napisana w języku C++ z użyciem dwóch, darmowych bibliotek skupiających się na cyfrowym przetwarzaniu audio.

```

#include "PluginProcessor.h"
#include "PluginEditor.h"

void DelayAudioProcessor::processBlock (AudioBuffer<float>& buffer, MidiBuffer&)
{
    const int numSamples = buffer.getNumSamples();

    for (int channel = 0; channel < buffer.getNumChannels(); ++channel)
    {
        float* const channelData = buffer.getWritePointer (channel);
        for (int sample = 0; sample < numSamples; ++sample)
        {
            const float input = channelData[sample];
            const int delaySamples = mDelayTime * getSampleRate();
            const float delayedSample = mDelayBuffer.read(delaySamples);
            const float output = input + mFeedback * delayedSample;

            mDelayBuffer.write(input + delayedSample * mFeedback);
            channelData[sample] = output;
        }
    }
}

```

Działanie kodu polega na zapisaniu wcześniejszych próbek dźwięku do bufora opóźnienia i następnie dodawaniu ich do obecnej próbki z określonym czasem opóźnienia.

A dokładnie:

Definiujemy funkcję *processBlock*, która jest wywoływana przez hosta wtyczki dla każdego bloku próbek audio.

*W kontekście przetwarzania sygnałów audio, **blok** oznacza fragment strumienia sygnału, który jest przetwarzany w jednym kroku czasowym. W tym przypadku, linia kodu "void DelayAudioProcessor::processBlock (AudioBuffer<float>& buffer, MidiBuffer&)" oznacza funkcję przetwarzającą bloki danych sygnału audio. Każdy blok składa się z pewnej liczby próbek (zwykle jest to wartość stała), a funkcja przetwarza każdy kanał sygnału w danym bloku osobno.*

Następnie zapisujemy liczbę próbek w aktualnym bloku przetwarzania zmiennej *numSamples*. Pierwsza pętla „for” iteruje przez każdy kanał dźwiękowy bloku. Kolejnym krokiem jest uzyskanie wskaźnika na bufor danych dla bieżącego kanału *channel* w celu odczytu lub zapisu próbek audio. Zagnieżdżona pętla „for” iteruje przez każdą próbkę w bloku. W niej odczytujemy wartości wejściowej próbki dla bieżącej iteracji pętli z bufora danych *channelData*. Obliczamy liczbę próbek opóźnienia na podstawie wartości parametru opóźnienia *mDelayTime* i aktualnej częstotliwości próbkowania. Odczytujemy próbkę z bufora o indeksie opóźnienia obliczonego na podstawie zmiennej *delaySamples*. Wyliczamy wartość wyjściową próbki poprzez dodanie wartości wejściowej do złożenia opóźnionej próbki *delayedSample*, pomnożonej przez wartość sprzężenia zwrotnego *mFeedback*. Na końcu aktualizujemy zawartość bufora, poprzez dodanie do niego nowych wartości próbek.

Efektem takiego zapisu jest "przesunięcie" zawartości bufora o wartość opóźnienia *mDelayTime*, co powoduje, że na wyjściu procesora efektu pojawiają się echa. Wartość sprzężenia zwrotnego *mFeedback* kontroluje ilość sprzężenia zwrotnego w efekcie, czyli ilość opóźnień, które będą zwracane z bufora, co wpływa na charakter dźwięku generowanego przez efekt.

Innymi interesującymi efektami specjalnymi, które często stosowane są przez inżynierów audio, przykładowo są:

- *Phaser* - efekt, który manipuluje sygnałem dźwiękowym przez dodanie opóźnienia i zmianę fazy, co tworzy efekt przestrzennego brzmienia;
- *Pitch Shifter* - efekt przesuwały wysokość dźwięku, co umożliwia uzyskanie niestandardowych brzmień;
- *Chorus* - efekt przetwarzania dźwięku, który dodaje pogłosu i powtórzeń, co daje wrażenie, że jest więcej niż jedno źródło dźwięku.

Są to jedynie przykłady wtyczek oraz nakreślenie w jaki sposób działają, jednak każda z nich musi wpisywać się w standardy (jak to bywa w całej informatyce). Wyróżniamy najpopularniejsze 3 standardy:

1. *VST (Virtual Studio Technology)* – standard opracowany przez firmę Steinberg. Standard ten cechuje kompatybilność z większością programów/systemów.
2. *AU (Audio Units)* – standard firmy Apple. Kompatybilność tylko z systemem macOS.
3. *DirectX* – standard firmy Microsoft. Dostępne tylko dla systemu Windows. Często używane w grach komputerowych i programach multimedialnych.

4. Informatyk w branży muzycznej

Branża muzyczna dzięki technologii staje się coraz bardziej dostępna dla przeciętnego użytkownika, jak i początkujących informatyków próbujących swoich sił, by zrewolucjonizować rynek. Jednak pod kątem technicznym jest szereg podstaw, na które trzeba zwracać uwagę.

Przykładowo:

Informatyk, jako programista, w branży muzycznej powinien mieć wiedzę z zakresu programowania, cyfrowej obróbki dźwięku, algorytmów syntezy dźwięku oraz zagadnień związanych z akustyką. Powinien także posiadać znajomość różnych standardów i formatów dźwiękowych, a także interfejsów audio i MIDI.

Lecz patrząc pod innym kątem niż programowanie:

Informatyk starający się o pracę jako np. akustyk, projektant studyjny lub producent, powinien posiadać podstawową wiedzę na temat funkcjonowania sprzętu i oprogramowania używanego w produkcji muzycznej. W szczególności powinien znać się na zastosowaniu interfejsów audio, mikrofonów, monitorów studyjnych, kontrolerów MIDI, oprogramowania DAW (Digital Audio Workstation), wtyczek efektów i instrumentów wirtualnych. Powinien także znać zasady nagrywania, miksowania i masteringowania muzyki oraz umieć wykorzystać narzędzia np. efekty specjalne do tworzenia różnych brzmień i efektów.

III. Wirtualne instrumenty

1. Czym są wirtualne instrumenty?

Jeśli zapytalibyśmy kogoś „czym są wirtualne instrumenty?” na pewno odpowiedziałby dobrze nawet własnymi zdaniem, ponieważ już sama nazwa doskonale je definiuje. Z teorii można uznać je także za jeden z rodzajów wtyczek VST. Działają na zasadzie syntezy i umożliwiają tworzenie różnego rodzaju brzmień i instrumentów za pomocą algorytmów i próbek dźwiękowych.

2. Rodzaje wirtualnych instrumentów

Wirtualne instrumenty mogą być dzielone, w zależności od rodzaju brzmienia i sposobu tworzenia dźwięków. Najbardziej popularnymi rodzajami są:

1. Sampler'y – pozwalają na manipulowanie i miksowanie próbek dźwiękowych. W skrócie bazują na uprzednio załadowanych próbkach dźwiękowych (lub dostępnych w bibliotece) i na ich podstawie tworzony jest nowy dźwięk
2. Syntezatory – jest to rodzaj wirtualnego instrumentu generującego zupełnie nowy dźwięk za pomocą algorytmów matematycznych. Działa na zasadzie generowania fal dźwiękowych o różnych kształtach i parametrach.

IV. Zastosowania

1. Jak było kiedyś?

Przed wprowadzeniem przetwarzania cyfrowego i rozwinięciu się inżynierii audio wiele branż działało na zasadzie technologii analogowej. Branże takie jak: muzyczna, telekomunikacyjna, czy nawet medyczna musiały bazować na tej technologii, co wiązało się z czasochłonnym procesem przetwarzania, co dodatkowo wymagało specjalistycznych umiejętności dalszego rozwoju przetwarzania sygnału w technologii analogowej.

2. Jak jest dzisiaj?

Obecnie inżynieria audio i przetwarzanie cyfrowe odgrywa kluczową rolę w wielu branżach. Wprowadziła liczne ułatwienia, modyfikacje starszych rozwiązań, jak i stworzyła wiele nowych. W powyższym tekście skupiłem się na przetwarzaniu cyfrowym i wtyczkach VST wykorzystywanych w branży muzycznej. Oczywiście dzięki rozwojowi tej technologii produkcja muzyki stała się łatwiejsza i szybsza, to samo dotyczy się dystrybucji. Jednak warto wspomnieć, że inżynierowie audio bazując na technologii przetwarzania cyfrowego, pracy z dźwiękiem/falami i wykorzystując narzędzia oparte na wtyczkach zmieniają wiele branż jak np.:

Medycyna – inżynieria audio i przetwarzanie cyfrowe umożliwia wykonywanie precyzyjnych pomiarów, diagnoz i badań, np. wiele z tej technologii wykorzystywane jest do tworzenia aparatów słuchowych, czy badań logopedycznych.

3. Jak może być?

Jest to już pytanie, aby można było podywagować. Wiadomym jest, że rozwój technologii, a przy tym przetwarzania cyfrowego i inżynierii, mają ogromny potencjał do zmiany świata w najbliższych latach. Najbardziej realnymi przykładami mogą być:

W dziedzinie muzyki i sztuki dźwiękowej, przetwarzanie cyfrowe i inżynieria audio mogą umożliwić jeszcze bardziej zaawansowane narzędzia do tworzenia, nagrywania i miksowania muzyki oraz dźwięków. Wirtualne instrumenty i efekty dźwiękowe są już teraz bardzo zaawansowane, a dalszy rozwój w tej dziedzinie może przynieść jeszcze bardziej realistyczne i wyrafinowane dźwięki.

W dziedzinie zdrowia, przetwarzanie cyfrowe i inżynieria audio mogą pomóc w diagnozowaniu i leczeniu różnych chorób. Na przykład, badania wykazały, że dźwięki o określonej częstotliwości mogą pomóc w leczeniu chorób neurologicznych. W przyszłości możliwe jest wykorzystanie zaawansowanych technologii dźwiękowych w celu diagnozowania chorób, monitorowania stanu pacjentów i leczenia różnych schorzeń.

Oczywiście nie możemy zapomnieć, że zaczynamy powoli wkraczać w erę coraz to bardziej rozwijającej się sztucznej inteligencji. Co w połączeniu z wieloma technologiami, w tym przypadku z wyżej wymienionymi, może znacznie wzbogacić, przyspieszyć i poprawić zamierzony efekt.

V. Źródła

<https://www.shure.com/pl-PL/wystepy-produkcja/louder/microphone-basics-transducers-polar-patterns-frequency-response>

Teoria przetwarzania A/C i C/A. Autor: Bartłomiej Gorczyński

<https://www.tophifi.pl/blog/post/analogowe-a-cyfrowe-brzmienie-audio-czego-moze-oczekiwac-audiofil.html>

<https://www.hifi.pl/slownik/>

<https://www.adobe.com/pl/creativecloud/video/discover/audio-sampling.html>

https://home.agh.edu.pl/~zbrudnic/Automatyka/Cw1/Przetw_AC.html

<http://www.marekczeszczek.com/sitemap/adrem/adrem-10/adrem-nnnn-wykl-text.htm>

https://neurotext.library.stonybrook.edu/C7/C7_1/C7_1.html