



Akademia Górniczo-Hutnicza
w Krakowie
Instytut Elektroniki
WIET



Laboratorium

Technika Mikroprocesorowa 2

Ćwiczenie 5

Przetwornik cyfrowo-analogowy C/A

Autor: Mariusz Sokołowski

wer. 28.09.2021

1. WSTĘP

1.1.CEL

Celem ćwiczenia jest:

- ✚ zapoznanie studenta z techniką poprawnej inicjalizacji, wyzwalania i obsługą danych przetwornika C/A,
- ✚ nabycie umiejętności łączenia funkcjonalnego układów peryferyjnych, na przykładzie:
 - współpracy przetwornika C/A z diodami LED,
 - współpracy przetwornika C/A z głośnikiem – generacja sygnałów o różnych kształtach:
 - sinus,
 - trójkąt,
 - piła.
 - wykorzystania przetwornika C/A do bezpośredniej, cyfrowej syntezy częstotliwości (DDS).

1.2.WYMAGANIA

Sprzętowe:

- komputer klasy PC, spełniający wymagania sprzętowe aplikacji KEIL v5,
- zestaw FRDMKL05Z

Programowe:

- system operacyjny Windows 7 lub wyższy (wszystkie instrukcje powstały w oparciu o Windows 7 Pro x64),
- środowisko Keil / uVision 5 MDK-ARM

Doświadczenie:

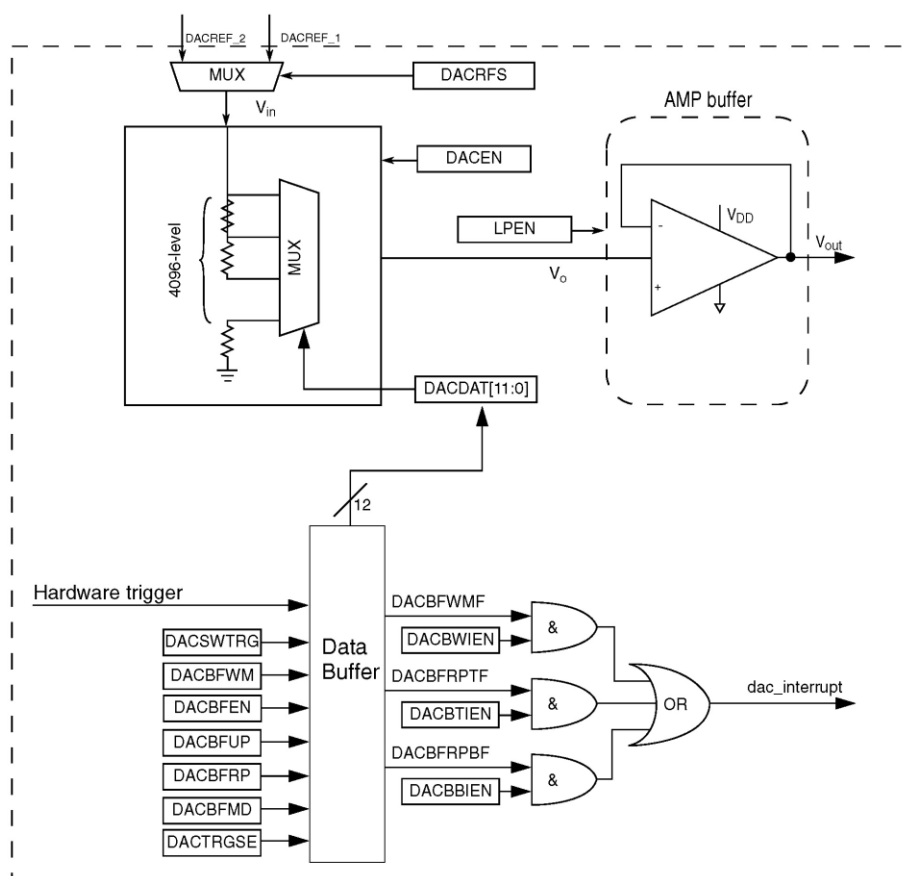
- podstawowa umiejętność obsługi komputera klasy PC,
- podstawowa znajomość systemów operacyjnych rodziny Windows,
- umiejętność programowania w języku C.

Literatura:

- KL05 Sub-Family Reference Manual, Freescale Semiconductor
- Kinetis L Peripheral Module Quick Reference, Freescale Semiconductor
- Joseph Yiu, The Definitive Guide to the ARM Cortex-M0, Elsevier, 2011
- ARM „Cortex-M0+ Devices Generic User Guide”
- schemat modułu głośnika ze wzmacniaczem WSR-04489

2. PRZETWORNIK C/A

Kolejnym układem peryferyjnym jest 12-bitowy przetwornik C/A. Jego schemat jest zamieszczony na rysunku Rys. 1.



Rys. 1 Schemat przetwornika C/A

Napięcie referencyjne przetwornika wynosi 2.91V (a nie 3.3V i może zależeć od egzemplarza). Czas ustalania się sygnału na wyjściu, to typowo 15 μ s, dla normalnego trybu pracy mikrokontrolera.

Podstawowa konfiguracja przetwornika C/A przebiega następująco:

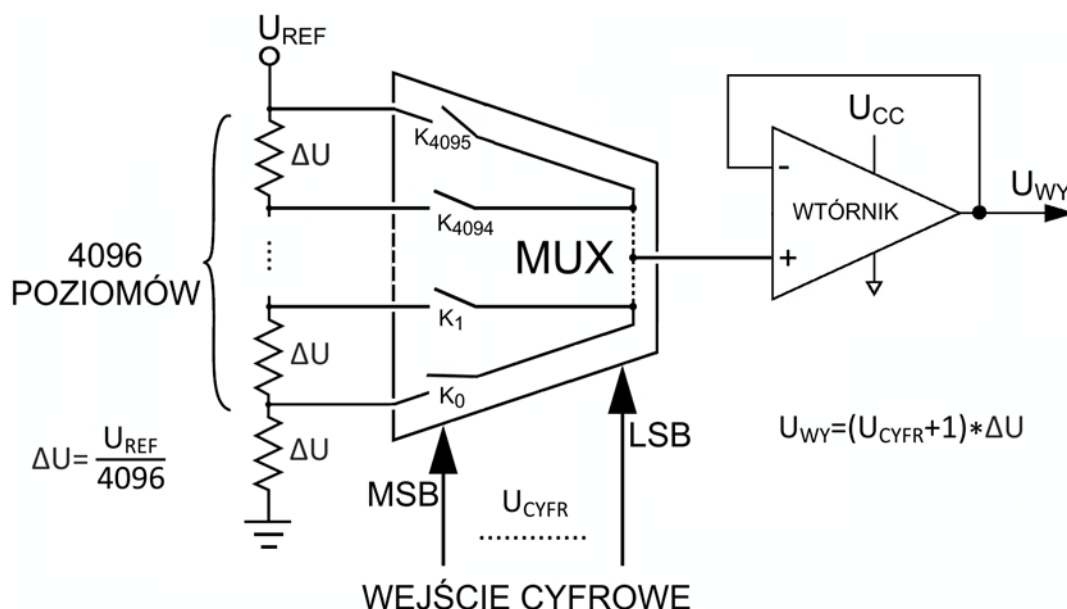
- ✓ dołączyć sygnał taktujący do modułu C/A, w rejestrze SIM_SCGC6[DAC0=1],
- ✓ dołączyć sygnał taktujący (w rejestrze SIM_SCGC5[PORTB=1] do odpowiedniego portu, którego pin będziemy wykorzystywać jako wyjście analogowe - w niniejszym ćwiczeniu jest to port B, pin PTB1 (jego domyślna funkcja, to wyjście analogowe przetwornika DAC0),
- ✓ ustawić tryb pracy z programowym wyzwaniem (DACTRGSEL=1) oraz włączyć przetwornik (DACEN=1), w rejestrze DAC0_C0.

Aby wymusić dany stan napięcia na wyjściu przetwornika, należy:

- wpisać młodszy bajt, 12-bitowej wartości, do rejestru DAC0_DAT[DATL],
- wpisać starszy bajt, 12-bitowej wartości, do rejestru DAC0_DAT[DATH],
- wyzwolić przetwarzanie, ustawiając bit DACSWTRG=1, w rejestrze DAC0_C0.

Warto pamiętać, że przetwornik jest 12-bitowy, więc zapisanie do jego rejestrów DATx, liczby większej niż 0xFFF, spowoduje obcięcie najstarszego półbajta. Dlatego dobrze jest stworzyć odpowiednią funkcję zapisującą, która najpierw sprawdzi zakres wpisywanej danej.

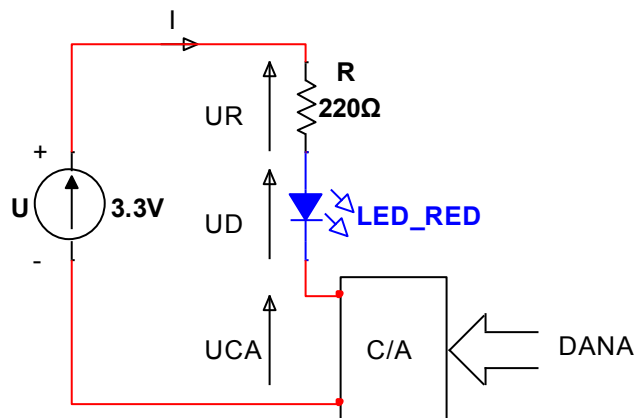
Tak jak w przypadku przetwornika A/C, istnieje również możliwość sprzętowego wyzwalania przetwarzania. Zaimplementowane jest tylko jedno źródło sygnału wyzwalającego, którym jest licznik PIT0, wybierany bitem DACTRGSEL=0, w rejestrze DAC0_CO.



Rys. 2 Bardziej „ludzki” schemat przetwornika C/A

3. STEROWANIE JASNOŚCIĄ ŚWIECENIA DIODY LED

Z poprzednich zajęć wiemy, że podanie napięcia o wartości 0V („0” logiczne), na katodę jednej z diod LED, pozwala tę diodę zaświecić i to z pełną jasnością. Natomiast podanie napięcia o wartości równej napięciu zasilania (3.3V), spowoduje jej zgaszenie. Co się stanie, jeśli zaczniemy podawać wartości z przedziału 0÷3.3V? Dioda LED potrzebuje konkretnej wartości napięcia, aby mogła świecić. W przypadku diody czerwonej jest to ok. 1.5V. Na rysunku Rys. 3 jest schemat podłączenia diody, z jednej strony do stałego napięcia zasilania, 3.3V, a z drugiej strony do regulowanego źródła napięcia, którym może być przetwornik C/A.



Rys. 3 Schemat podłączenia diody LED do przetwornika C/A

Z prawa Kirchhoff'a wiadomo, że:

$$U = U_R + U_D + U_{CA}$$

Jak powiedziano wcześniej, aby dioda zaczęła świecić (przewodzić prąd I), musi na niej zapanować napięcie 1.5V.

$$U_D = U - U_R - U_{CA}$$

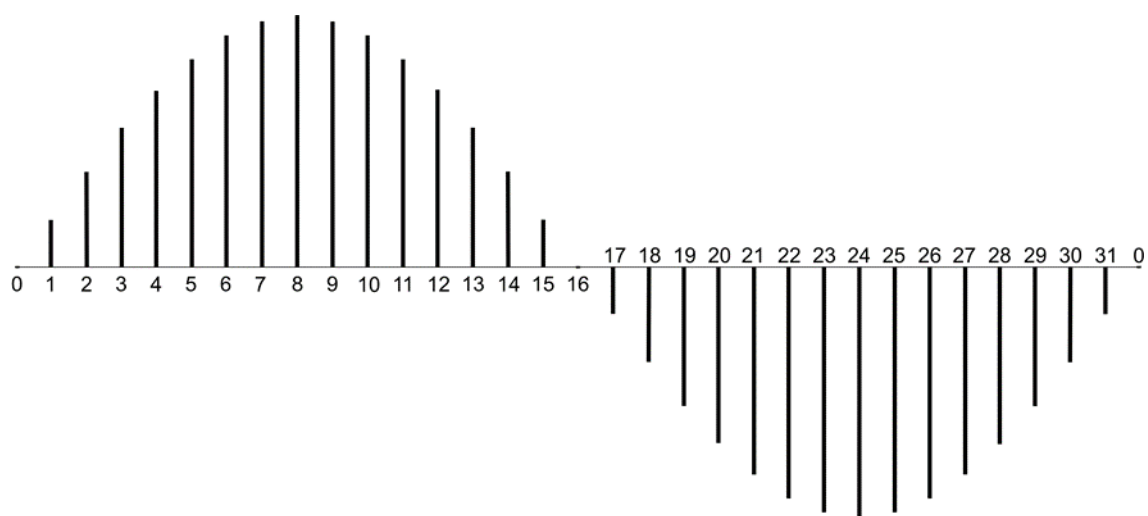
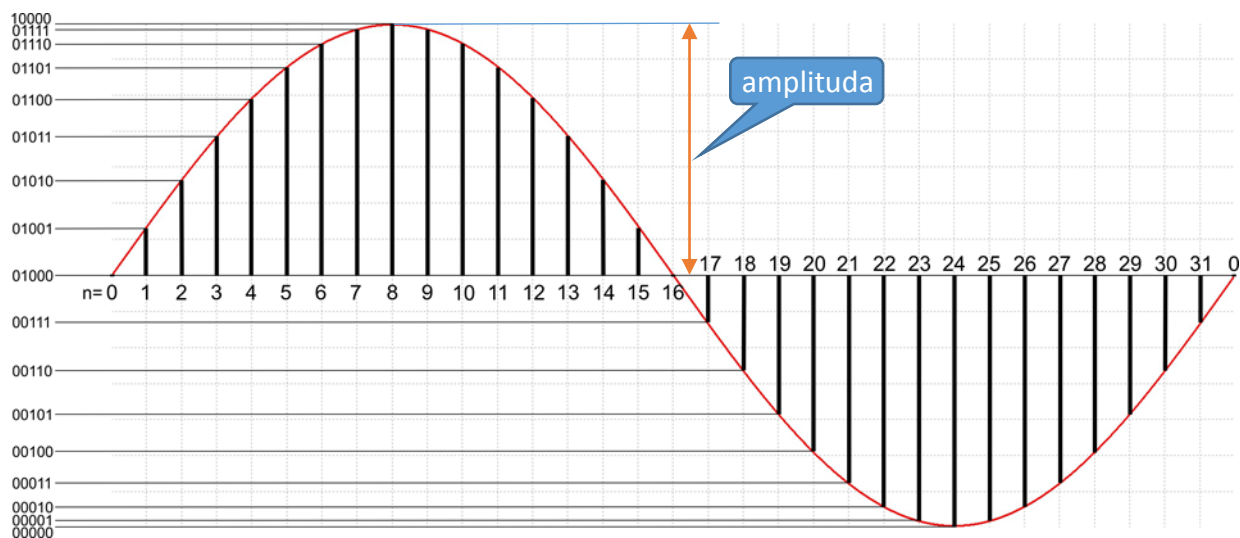
Dopóki na diodzie nie będzie napięcia 1.5V, prąd I nie płynie i napięcie $U_R = 0V$ na rezystorze R . Wyliczmy w takim razie, jakie napięcie powinien dać przetwornik C/A, aby dioda zaczęła świecić.

$$U_{CA} = U - U_D = 3.3V - 1.5V = 1.8V$$

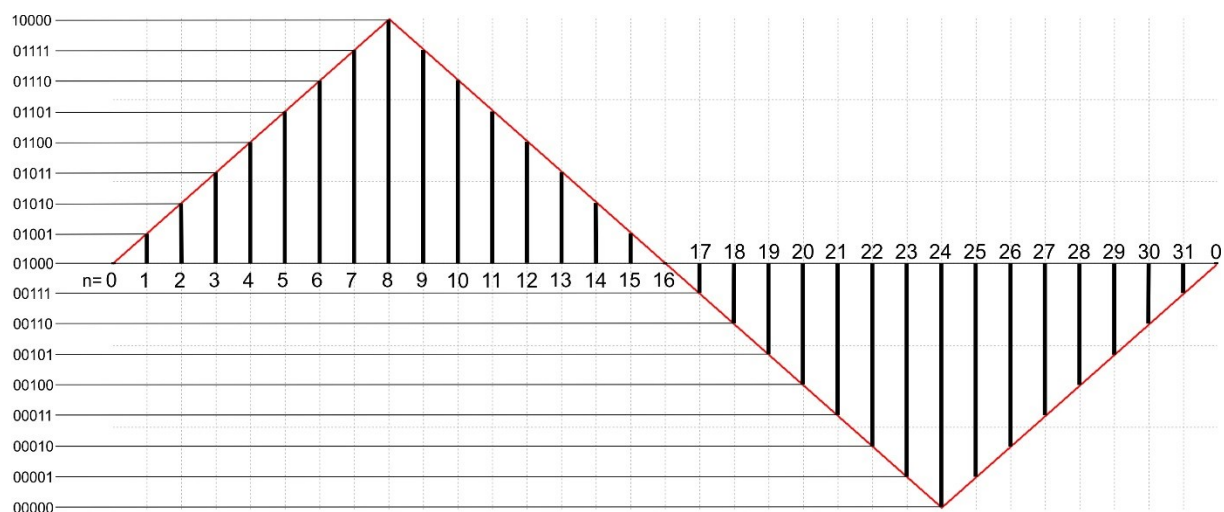
Im niższe napięcie na katodzie diody, czyli na przetworniku, tym dioda będzie świecić jaśniej, ponieważ rośnie wtedy wartość prądu I , a pozostała część napięcia odkłada się na rezystancji R , która ma za zadanie ograniczyć wartość tego prądu, aby nie był za duży i spalił diody. Modulując napięcie na przetworniku, w zakresie $0 \div 1.8V$ można regulować jasność świecenia diody. Podobny mechanizm występował w przypadku sterowania PWM, tylko wtedy dioda była sterowana impulsami, typu „świeci”-„nie świeci”, a napięcie na niej występujące było średnim napięciem za okres PWM. W przypadku przetwornika C/A napięcie to nie zmienia się (nie ma postaci impulsów) i jest stałe, do momentu zmiany jego wartości przez programistę.

4. GENERACJA SYGNAŁÓW O RÓŻNYCH KSZTAŁTACH

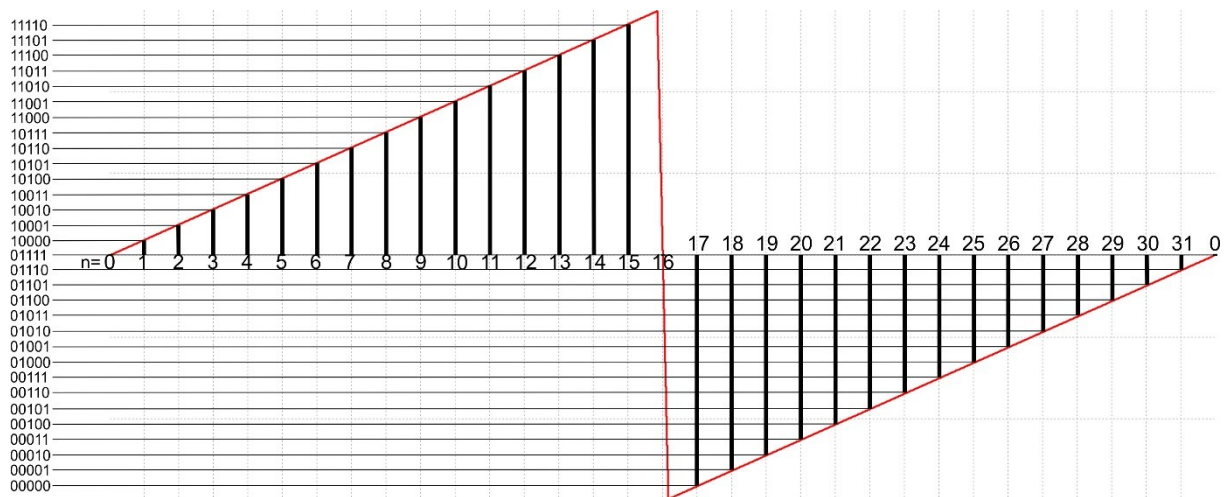
Jeśli teraz zaczniemy do przetwornika wpisywać, ze stałą prędkością (częstotliwością), kolejne liczby, które będą np. kolejnymi wartościami funkcji sinus, to na wyjściu przetwornika C/A pojawi się napięcie o kształcie sinusoidy oraz częstotliwości zależnej od prędkości wpisywania i od liczby próbek, tworzących okres sinusa (Rys. 4). Jeżeli zastosujemy próbki przebiegów o innym kształcie, np. trójkąta (Rys. 5) lub piły (Rys. 6), to takowy przebieg otrzymamy na wyjściu.



Rys. 4 Generacja funkcji sinus w oparciu o 32-ie próbki



Rys. 5 Generacja przebiegu trójkątnego w oparciu o 32-ie próbki



Rys. 6 Generacja piły w oparciu o 32-ie próbki

W przedstawionym przykładzie mamy 32 próbki funkcji sinus, składające się na cały okres 2π . Aby można było rozróżnić 17 różnych stanów wartości napięcia próbek, składających się na uformowanie okresu, potrzebny będzie przetwornik C/A minimum 5-cio bitowy: minimum $00000b=0d$ i maksimum $10000b=16d$.

Numer kolejnej próbki n , zmieniający się od 0 do 31, wyznacza dokładnie fazę funkcji sinus, a faza z kolei determinuje wartość napięcia (wartość cyfrowa od 0 do 16, dla przetwornika C/A) (Rys. 7). Krok zmiany fazy określa wzór:

$$\Delta\varphi = \frac{2\pi}{32}$$

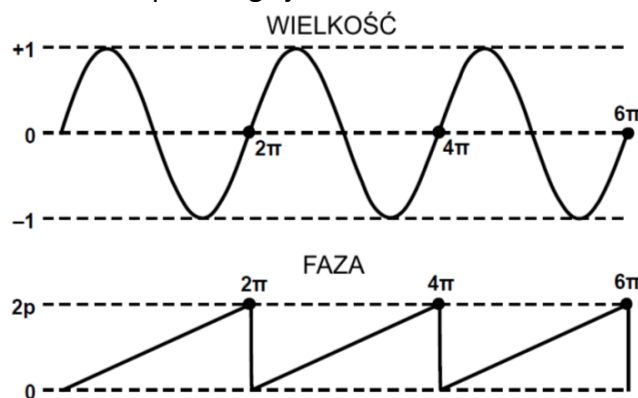
Pamiętajmy, że po próbce $n=31$ jest próbka $n=0$, która jest końcem jednego i zarazem początkiem następnego okresu.

Jeśli teraz odtworzymy wszystkie 32 próbki ze stałą częstotliwością f_{CLK} , to częstotliwość f_s wygenerowanej (zsyntezowanej) sinusoidy będzie wyrażać się wzorem:

$$f_s = \frac{f_{CLK}}{32}$$

W przypadku przebiegu trójkątnego jest podobna sytuacja, co dla sinusa.

W przypadku piły, pojawia się 31 stanów (od 0 do 30), ale również potrzeba 5 bitów, aby to przedstawić. Reszta rozumowania przebiega jak dla sinusa.



Rys. 7 Zależność wielkości i fazy funkcji sinus

5. BEZPOŚREDNIA, CYFROWA SYNTEZA CZĘSTOTLIWOŚCI DDS (ANG. DIRECT DIGITAL SYNTHESIS)

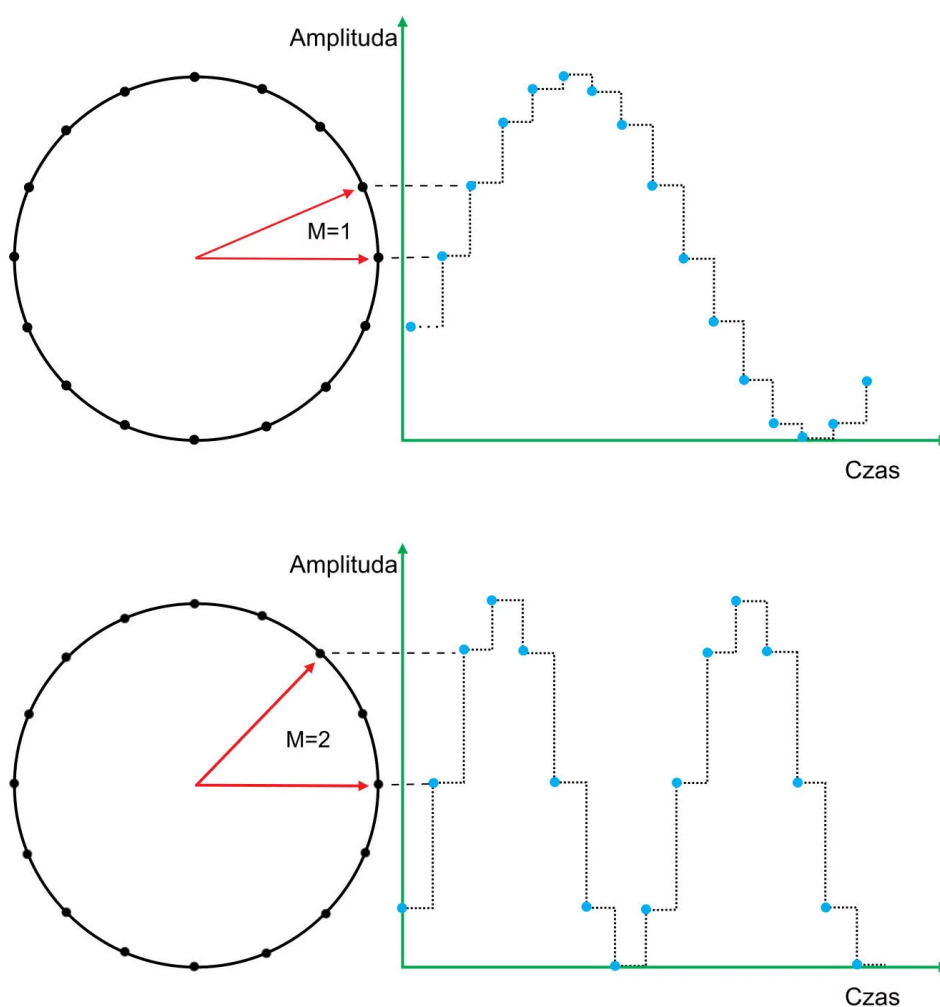
A co się stanie, jeśli będziemy odtwarzać np. co drugą próbkę, ale z tą samą częstotliwością f_{CLK} ? Otrzymamy:

$$f_s = 2 \frac{f_{CLK}}{32} = \frac{f_{CLK}}{16}$$

czyli dwa razy większą częstotliwość niż poprzednio. Gdyby to była co M-ta próbka (Rys. 8):

$$f_s = M \frac{f_{CLK}}{32} = M \Delta f_s$$

Wartość $\frac{f_{CLK}}{32}$ jest najmniejszą częstotliwością, którą możemy uzyskać, a każda następna jest jej M-tą wielokrotnością.



Rys. 8 Synteza sygnału w oparciu o M-tą próbkę

Gdyby zwiększyć rozdzielczość fazy, czyli wprowadzić większą liczbę próbek (zagęścić) do wartości 2^N (u nas $N=5$, bo $2^5=32$ próbki), to wzór na syntezywaną częstotliwość miałby postać:

$$f_s = M \frac{f_{CLK}}{2^N}$$

Jednak z wartością M nie można przesadzać, ponieważ sygnał wyjściowy nie może być stworzony z mniej niż dwóch próbek. Wynika stąd, że przy zadanym N , wartość M musi być mniejsza od 2^{N-1} , co oznacza, że maksymalna częstotliwość syntezywanego przebiegu nie może być większa od połowy częstotliwości taktowania f_{CLK} .

Przykład:

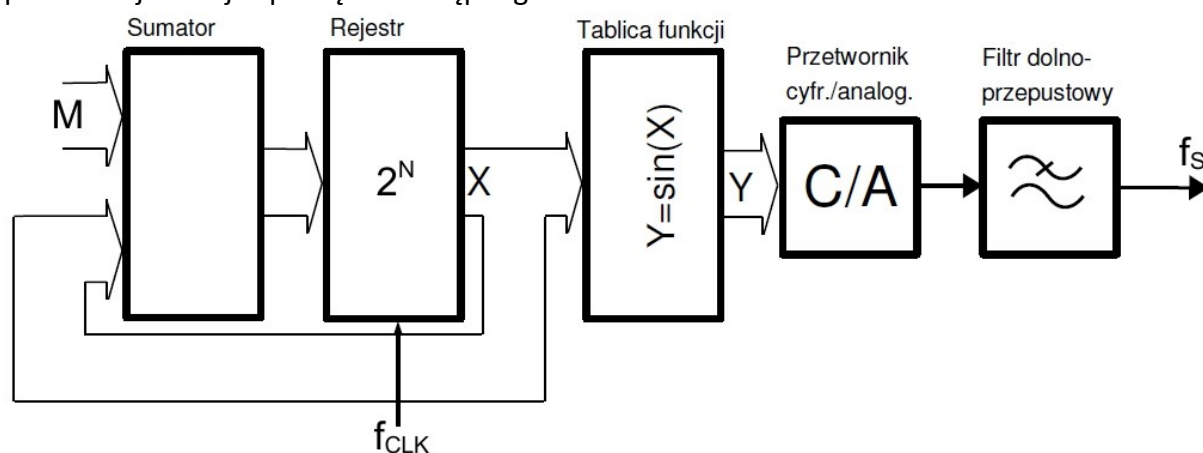
$f_{CLK}=100\text{MHz}$, $N=32 \rightarrow f_{s\min}=0.023\text{Hz}$, $f_{s\max}= 49.999999977\text{ MHz}$, rozdzielczość $\Delta f_s=0.023\text{Hz}$

Proszę zwrócić uwagę na niesamowitą rozdzielczość, nieosiągalną innymi metodami.

W/w technika pozwala na wygenerowanie praktycznie dowolnego sygnału, którego pasmo częstotliwości jest węższe niż połowa częstotliwości generacji próbek f_{CLK} , a którego próbki są przechowywane w pamięci generatora DDS (tablica funkcji).

Bezpośrednia synteza cyfrowa może być zrealizowana przy wykorzystaniu procesora sygnałowego lub wyspecjalizowanego układu, co umożliwia generację prostszego sygnału (np. sinusoidalnego), ale o większej częstotliwości. Układ realizujący syntezę częstotliwości składa się z generatora cyfrowej fazy, generatora funkcji oraz przetwornika cyfrowo-analogowego (Rys. 9).

Generator cyfrowej fazy składa się z rejestru oraz sumatora. Są one połączone tak, że w każdym takcie pracy układu liczba przechowywana w rejestrze jest zwiększana o stałą wielkość M (modulator fazy). Przeniesienie z najstarszej pozycji rejestru jest ignorowane i cykl zliczania jest powtarzany (modulo 2^N). Jeśli rejestr ma długość N bitów i w każdym cyklu dodawana jest liczba M , to rejestr przepełnia się średnio co $\frac{M}{2^N}$ taktów zegara f_{CLK} . Kolejne stany rejestru można interpretować jako kolejne próbki fazy generowanego przebiegu, przy czym kątom od 0 do 2π odpowiadają liczby od 0 do 2^{N-1} , przy czym dla 2π to znowu jest 0 , ponieważ jest to już początek następnego okresu.



Rys. 9 Schemat blokowy układu DDS

6. ĆWICZENIE 1

Pojedynczym kabelkiem połączyć wyjście przetwornika DAC0 (pin PTB1) z katodą diody czerwonej LED (pin PTB8). Rozpakować zbiór *1_DAC_LED.zip*. Uruchomić projekt *DAC_LED.uvprojx*. Na diodę czerwoną LED zostaje podany sygnał z przetwornika cyfrowo-analogowego DAC0. Na wyświetlaczu pokazywana jest aktualna wartość napięcia przetwornika, którą można regulować za pomocą pola dotykowego.

6.1.ZADANIE

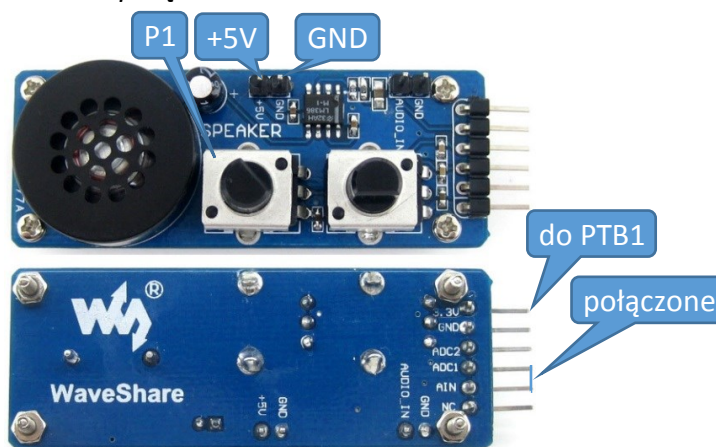
- ❖ Doprowadzić czerwoną diodę LED do stanu bardzo słabego świecenia i bardzo mocnego świecenia. Dlaczego, niezależnie od jasności, dioda nie mruga, tylko świeci światłem ciągłym?
- ❖ Doprowadzić czerwoną diodę LED do stanu, w którym jest na granicy świecenia. Odczytać wartość napięcia przetwornika DAC0 i na jego podstawie oraz wytycznych z punktu 3, wyliczyć napięcie przewodzenia (świecenia) diody czerwonej. Napięcie zasilania U przyjąć 2.91V.
- ❖ Wiedząc, że katoda zielonej diody LED jest podłączona do pinu PTB9, a niebieskiej do PTB10, wyznaczyć napięcie przewodzenia (świecenia) tych diod.

7. ĆWICZENIE 2

Podłączyć głośnik, wg tabeli Tab. 1 i rysunku Rys. 10. Potencjometr P1 – głośność.

FRDM-KL05	WSR-04489
PTB1	VCC (3.3V)
+5V	+5V
GND	GND

Tab. 1 Tabela połączeń modułu WSR-04489 z KL05Z



Rys. 10 Głośnik WSR-04489 i jego podłączenie do systemu

Rozpakować zbiór *2_DAC_Kształt.zip*. Uruchomić projekt *DAC_Kształt.uvprojx*. Układ generuje sygnał sinusoidalny, o częstotliwości 500Hz i amplitudzie 0.73V. Klawisz S2 włącza kształt sinus, klawisz S3 trójkąt, a klawisz S4 piłę. Polem dotykowym regulujemy głośność (amplitudę) przebiegu wyjściowego.

7.1.ZADANIE

- ❖ Na podstawie analizy programu oraz punktu 4, dokonać zmian, aby częstotliwość generowanych przebiegów wynosiła 1000Hz.
- ❖ Zauważyć, jak zmienia się barwa tonu, w zależności od kształtu przebiegu.

8. ĆWICZENIE 3

Rozpakować zbiór *3_DAC_DDS.zip*. Uruchomić projekt *DAC_DDS.uvprojx*. Układ generuje, na zasadzie DDS, przebieg sinusoidalny, o amplitudzie regulowanej za pomocą pola dotykowego. Klawisz S2 zwiększa częstotliwość o 8Hz, a klawisz S3 zmniejsza częstotliwość o 8Hz. Klawisz S4 służy do zatrzymywania i uruchamiania generatora.

Tablica „sinusów” posiada 1024 próbki ($2^{N=10}$), o rozdzielczości 12-u bitów. Częstotliwość taktująca f_{CLK} wynosi 8192Hz, a ustala ją licznik SysTick. Z powyższych danych wynika, że najmniejsza wartość f_s to 8Hz, a maksymalna 4088Hz, z rozdzielczością $\Delta f_s=8Hz$.

Rejestr cyfrowego generatora fazy ma długość 10-iu bitów ($N=10$). Modulator fazy M może przyjmować wartości od 1 do 510.

8.1.ZADANIE

- ❖ Przeanalizować pracę programu, zwracając szczególną uwagę na realizację cyfrowego generatora fazy.
- ❖ Jak można zwiększyć dwukrotnie maksymalny zakres generowanych częstotliwości, nie zmieniając długości rejestru generatora fazy? Jak wpłynie to na rozdzielczość częstotliwości Δf_s ?
- ❖ Pobawić się.