

Generator sygnału 1 kHz

Raport Techniczny

Szymon Januszek, szymon_j@tutanota.com

Kwiecień 2023

Streszczenie

Głównym celem poniższej konstrukcji jest otrzymanie sygnału możliwie najwyższej jakości, zdefiniowanych w założeniach konkursowych, tzn. dokładność częstotliwości, stopień zniekształceń harmonicznych oraz dokładność amplitudy sygnału.

Spis treści

1 DDS	3
1.1 Mikrokontroler	3
1.2 Digital-to-Analog Converter - Przetwornik Cyfrowo-Analogowy (DAC)	3
1.3 Analiza sygnału wyjściowego	4
2 Kształtowanie	4
3 Kontrola Amplitudy	5
3.1 Pomiar amplitudy	5
3.2 Element sprzężający	5
3.3 Układ odniesienia	6
4 Implementacja	6
4.1 Kod źródłowy	6

1 Bezpośrednia synteza cyfrowa

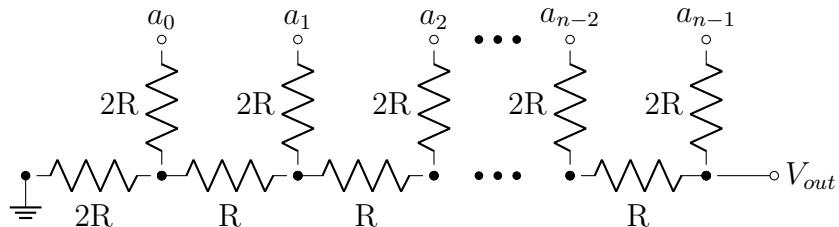
Aby wygenerować maksymalnie stabilny i dokładny częstotliwościowo sygnał, zdecydowałem się na wykorzystanie techniki *Direct Digital Synthesis - Bezpośrednia Synteza Cyfrowa (DDS)*, gdzie mikroprocesor wraz z układem *Digital-to-Analog Converter - Przetwornik Cyfrowo-Analogowy (DAC)* generuje sygnał zbliżony do sygnału oczekiwanej. Umożliwia to zastosowanie wysokiej jakości generatora kwarcowego, jako głównego źródła odniesienia, o精度 rzędu 50ppm¹, w całym zakresie temperatur. Oznacza to, iż częstotliwość uzyskanego sygnału nie będzie odbiegać o więcej niż ± 50 mHz od wartości oczekiwanej.

1.1 Mikrokontroler

Jako główny mikrokontroler wybrany został układ **AVR32DA28**[2] należący do rodziny 8-bitowych układów AVR. W porównaniu ze starszymi generacjami, producent dopuszcza pracę układu przy częstotliwościach przekraczających 8 MHz z napięciem 3 V. Jednocześnie dokonana została unifikacja modelu pamięci, co okaże być się przydatne przy pisaniu oprogramowania.

1.2 DAC

Do implementacji przetwornika zrealizowany został układ drabinki R-2R (Rys.1). Umożliwiając łatwą zamianę wartości liczbowej, przedstawionej jako liczba binarna na pinach mikrokontrolera, na ułamek napięcia zasilania.



Rysunek 1: Schemat drabiny R-2R [1]

Napięcie na węźle wyjściowym przetwornika można opisać w następujący sposób:

$$V_{out} = V_{ref} \times \frac{a_0 \times 2^0 + a_1 \times 2^1 + a_2 \times 2^2 + \dots + a_{N-1} \times 2^{N-1}}{2^N}$$

Jak widać, przy $N = 8$, przetwornik ten pozwala na uzyskanie 256 różnych wartości. Dla $V_{ref} = 3$ V, oznacza to, iż różnica pomiędzy dwoma dowolnymi stopniami wynosi ok. 11.72 mV.

Co ciekawe, okazuje się, iż impedancja wyjściowa takiego układu jest stała dla każdego poziomu ².

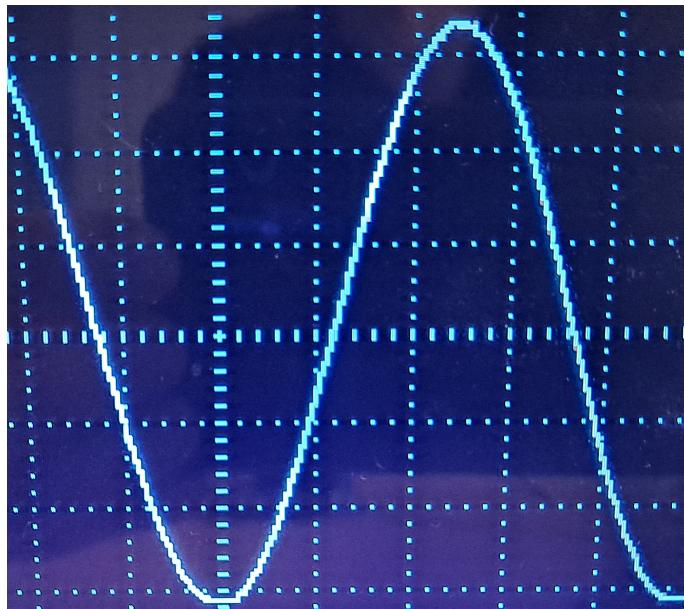
¹Części na milion

²Przy założeniu zerowej impedancji źródła oraz mikrokontrolera. W rzeczywistości ich wartości są znacznie mniejsze od wartości R . Wynika to z Twierdzenia Thévenina

Aby zachować użyteczność najniższych bitów, precyza wykorzystanych rezysto-rów musi być lepsza niż $\frac{1}{2^N} \times 100\% \approx 0.3\%$. Na szczęście, w sprzedaży dostępne są rezistory przeznaczone do montażu powierzchniowego o precyzyji 0.1%.

1.3 Analiza sygnału wyjściowego

Sygnal generowany w ten sposób nie opowiada jednak perfekcyjnej sinusoidzie (Rys. 2). Obecne jest tzw. zniekształcenie kwantyzacji, wynikające z faktu iż sygnał złożony jest z serii dyskretnych wartości. Jednak jeśli dokładnie przyjrzymy się różnicy pomiędzy naszym sygnałem a idealnej sinusoidzie, to okaże się, że amplituda takiego sygnału jest nie wielka ($\Delta V \leq \frac{V_{ref}}{2^N}$), a także składa się on z częstotliwości wielokrotnie większych niż częstotliwość fundamentalna generowanego sygnału.



Rysunek 2: Przebieg sinusoidy generowanej przy pomocy przetwornika R-2R.

Możemy uznać, iż dobrym przybliżeniem takiego sygnału będzie suma sinusoidy o amplitudzie równej połowie napięcia zasilania V_{ref} i o częstotliwości f_0 oraz sygnału prostokątnego o częstotliwości $f_0 \times N_{smp}$ i amplitudzie $0.5 \times \frac{V_{ref}}{2^N}$, gdzie N_{smp} oznacza ilość dyskretnych zmian napięcia w czasie jednego okresu.

Widzimy więc że częstotliwość już częstotliwości drugiej składowej jest N_{smp} -krotnie większa, a jej amplituda $2^{N-2}\pi$ -krotnie mniejsza w porównaniu ze składową fundamentalną. Oznacza to, iż przy zastosowaniu filtra dolnoprzepustowego niskiego stopnia, możliwe jest efektywne usunięcie składowych wyższej częstotliwości.

2 Kształtowanie

TBD

3 Kontrola Amplitudy

Ostatnią zmienną którą należy rozpatrzyć jest amplituda sygnału wyjściowego. Na tym etapie amplituda sygnału jest większa od oczekiwanej wartości 0.5 V. Potrzebna jest więc metoda dokładnego tłumienia, nie wprowadzająca dodatkowych zniekształceń i zakłuceń.

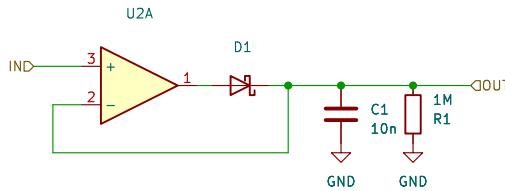
Rozwiążanie oparte o nawet precyzyjny potencjometr pozostaje jednak wrażliwe na takie zmienne jak napięcie zasilania, czy też dryft parametrów elementów związany ze zmianą temperatury.

Tak więc, niezbędny jest układ aktywnie korygujący amplitudę sygnału. Cyfrowa implementacja takiego rozwiązania nie wchodzi w grę, ponieważ przetwornik DAC nie posiada wystarczającej głębokości bitowej. Stąd też, proponuję rozwiązanie w pełni analogowe.

Przy użyciu wzmacniaczy operacyjnych możemy uzyskać pętle sprzężenia zwrotnego, korygującą amplitudę sygnału względem stałego napięcia odniesienia. Do realizacji tego rozwiązania potrzebujemy następujących elementów.

3.1 Pomiar amplitudy

Po pierwsze, konieczny jest pomiar amplitudy sygnału wyjściowego. To tego celu służy tzw. Detektor szczytowy (Rys. 3).



Rysunek 3: Przykładowy schemat połówkowego detektora szczytowego

3.2 Element sprzężający

Wymagany jest układ umożliwiający tłumienie sygnału zmiennego względem sygnału sterującego:

$$V_{wyj} \propto \frac{V_{wej}}{V_{ster}} \quad (1)$$

Fizyczna implementacja takiego układu okazuje się być niebanalna. Możliwy jest układ realizujący operacje mnożenia wartości sygnałów. Jego realizacja jednka jest wysoce złożona i wymaga relatywnie dużo części, ponieważ w jego skład wchodzą układy wyznaczające logarytm naturalny obu wartości, dodające te wartości do siebie, a na koniec wyznaczające funkcje wykładniczą e^x tej sumy.

Alternatywna metoda wykorzystuje element zmieniający swój opór elektryczny względem jakiegoś innego czynnika. Przykładem historycznej realizacji takiego układu jest zastosowanie niewielkiej lampy żarowej, której opór włukna jest prosty proporcjonalny do jego temperatury.

Istnieje jednak znacznie prostrze i wydajniejsze rozwiązanie.

Popularnym elementem o zmiennym oporze jest fotorezystor. W przeciwnieństwie do innych elementów światłoczułych, przy stałym natężeniu światła, zachowuje się on jak faktyczny rezistor, tzn. prąd przepływający jest wprost proporcjonalny do napięcia i odwrotnie do jego oporu $I \propto \frac{V}{R_\phi}$.

Tak więc, jeśli zamknie się fotorezystor w obudowie światłoszczelnej wraz z diodą LED, uzyskamy układ o zmiennym oporze elektrycznym, kontrolowany przez prąd diody.

$$R \propto \frac{1}{I_{Led}} \quad (2)$$

Umożliwia to łatwą implementację układu sprzężenia (Eq. 1).

Co prawda powyższy model ignoruje fakt, iż tak na prawdę opór fotorezystora zależny jest wykładniczo od natężenia padającego światła $R = \left(\frac{1}{\phi}\right)^A$. Nie stanowi jednak to dużego problemu ponieważ układ ten znajduje się w pętli sprzężenia zwrotnego.

$$R \propto \frac{1}{(I_{Led})^A} \quad (3)$$

3.3 Układ odniesienia

W tym momencie potrzebne jest jedynie stabilne oraz dostrajalne źródło napięcia odniesienia. Do tego celu z zupełności wystarczy układ LM4041 wraz z przeczyjnym potencjometrem.

4 Implementacja

4.1 Kod źródłowy

```

; (C) Szymon Januszek 2023
#include <avr/io.h>

.extern SINE_TAB
.global main

#define PORT_OUT VPORTD_OUT
#define PORT_DIR VPORTD_DIR

main:
    eor r1, r1

    ;select external clock as Main Clock source
    ldi r0, 0x3
    sts CLKCTRL_MCLKCTRLA, r0

    ;set port out and clear
    ldi r0, 0xFF
    out PORT_DIR, r0
    out PORT_OUT, r1

    ;load array address
    ldi r20, hi8(SINE_TAB)
    mov r30, r20
    mov r31, r1

    ldi r3, 3
    ldi r0, 1

loop:
    ;pull next byte out of memory and increment the pointer
    ld r2, Z+

    ; randomise lowest bit, g' = (g*3) % 256
    ;r0 * r3 = (r1, r0) unsigned
    mul r0, r3
    andi r1, 1
    eor r2, r1

    ;output data
    out PORT_OUT, r2

    ;cyclize data pointer
    andi r30, 0x03
    or r30, r20

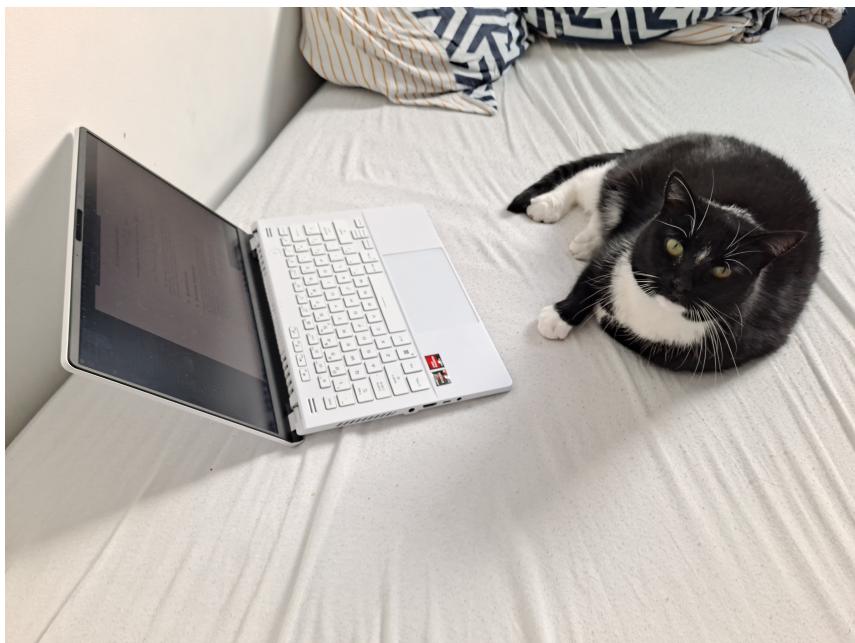
    ;pad for 12.288 MHz
    nop

    rjmp loop

```

References

- [1] CC BY-SA 2.0. URL: <https://commons.wikimedia.org/wiki/File:R2r-ladder.png>.
- [2] Microchip corp. *AVR32DA Final Data Sheet*. URL: <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/AVR32DA28-32-48-Data-Sheet-40002228B.pdf>. (7 kwiecień 2023).



Rysunek 4: Kot autora wspomagający proces twórczy