



AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

WYDZIAŁ INFORMATYKI, ELEKTRONIKI I TELEKOMUNIKACJI

KATEDRA ELEKTRONIKI

Systemy Wbudowane

Sprawozdanie z projektu

Generacja przebiegów sinusoidalnych przy użyciu algorytmu CORDIC

Systemy dedykowane w układach programowalnych, 2018

Mateusz Dyrdół

Damian Świerk

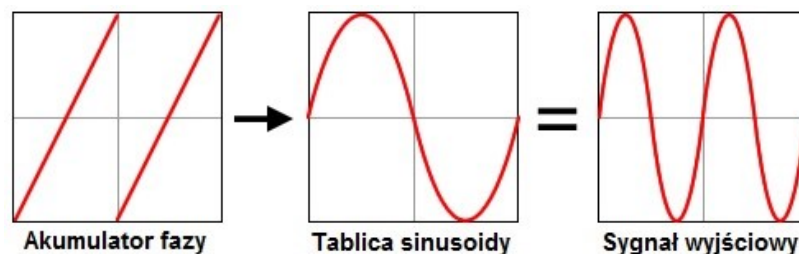
1. Cel projektu

Celem projektu jest implementacja algorytmu CORDIC do generacji fali sinusoidalnej o zadanej przez użytkownika częstotliwości sygnału generowanego.

2. Opis projektu

W celu dokonania implementacji układu generującego sygnał sinusoidalny przez generację kolejnych próbek sygnału za pomocą układu FPGA pierwszym krokiem jest implementacja bloku funkcjonalnego algorytmu CORDIC. Algorytm ten pozwala na wyznaczenie wartości funkcji sinus i cosinus z podanego na wejście kąta zawierającego się w zakresie od 0 do $\Pi/2$ wyrażonego w radianach i podanie tych wartości na wyjście bloku. W trakcie projektu została wykorzystana zmodyfikowana przez autorów wersja potokowego algorytmu CORDIC przedstawiona na ćwiczeniach laboratoryjnych.

Kolejnym krokiem projektu jest implementacja bloku funkcjonalnego, który będzie realizował generację kolejnych próbek kąta podawanego na wejście bloku z algorytmem CORDIC. Blok ten, czyli inkrementator fazy, będzie miał za zadanie podawanie na wejście algorytmu CORDIC próbek kąta generowanych z każdym taktom zegara, wraz z inkrementacją kolejnych wartości kąta, która jest zależna od podanej na wejście bloku częstotliwości żądanego sygnału. Ideowy schemat takiego sposobu generacji wartości sygnału sinusoidalnego przedstawia rysunek:



Rysunek 1 Schemat ideowy generacji sygnału sinusoidalnego z wykorzystaniem inkrementatora fazy

Z racji tego, że algorytm CORDIC przyjmuje jedynie wartości kąta zawierające się w przedziale $0 \div \Pi/2$, proces generacji sygnału należy jeszcze uzupełnić o blok, który wyznaczane przez CORDIC wartości funkcji sinus i cosinus będzie przekształcał za pomocą trygonometrycznych wzorów redukcyjnych na wartości zawierające się w pełnym zakresie okresu sygnału. Takie działanie wykonywane jest przez wyjściowy blok funkcjonalny, który przyjmuje wartości przekazywane z modułu CORDIC'a oraz wektor wyjściowy bloku inkrementatora „mode”, który określa jaka „ćwiartka” sygnału jest właśnie generowana.

3. Realizacja projektu

Jednym z założeń realizacji projektu było zachowanie dokładności wyliczonych wartości sygnału na poziomie rozdzielczości sygnału wyjściowego. Założono, że sygnał wyjściowy jest sygnałem szesnasto-bitowym. W celu zapewnienia dokładności obliczeń, dokonano porównania jakie błędy wyznaczenia wartości funkcji sinus w algorytmie CORDIC są popełniane dla różnych wielkości wektora kąta wejściowego i wektorów obliczeń. Na podstawie sprawdzonych błędów zdecydowano, że wektory danych kąta, funkcji sinus i funkcji cosinus w bloku algorytmu CORDIC będą miały format stałoprzecinkowy 18-to bitowy (2 bity części całkowitej i 16 bitów części ułamkowej) zapisane w kodzie U2.

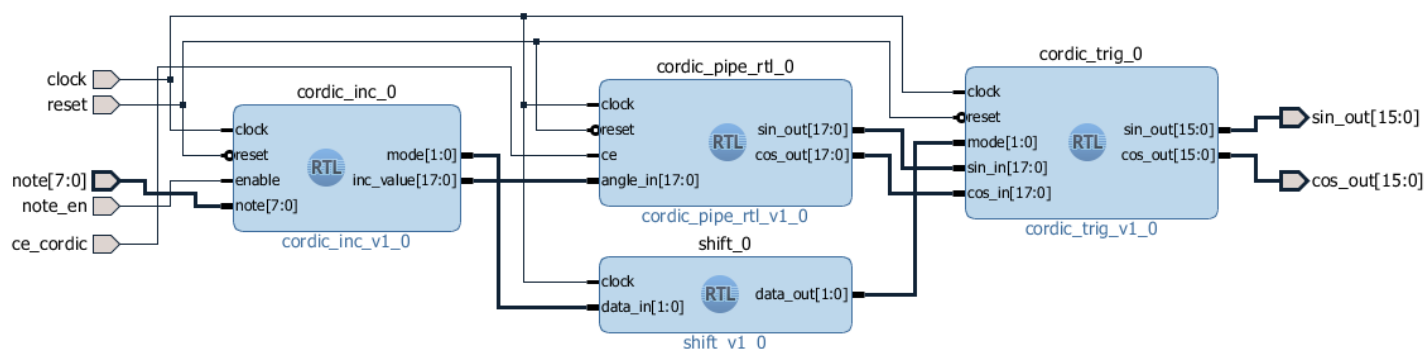
Wartość częstotliwości generowanego sygnału podawana jest na wejście układu jako 8-io bitowy wektor.

Dodatkowo w celu zapewnienia generacji sygnału dla całego okresu (a nie jak ma to miejsce dla algorytmu CORDIC gdzie generowane są wartości dla kąta w zakresie od 0 do $\Pi/2$) jest przekazywanie do ostatniego blozka dwu bitowej zmiennej „mode” zawierającej informację dla jakiej ćwiartki okresu są generowane wartości sygnału.

Wyjściowymi sygnałami układu są 16-to bitowe wektory w formacie stałoprzecinkowym (2 bity części całkowitej i 14 bitów części ułamkowej) zapisane w kodzie U2 reprezentujące kolejne wartości generowanego sygnału sinusoidalnego o zadanej przez nas częstotliwości.

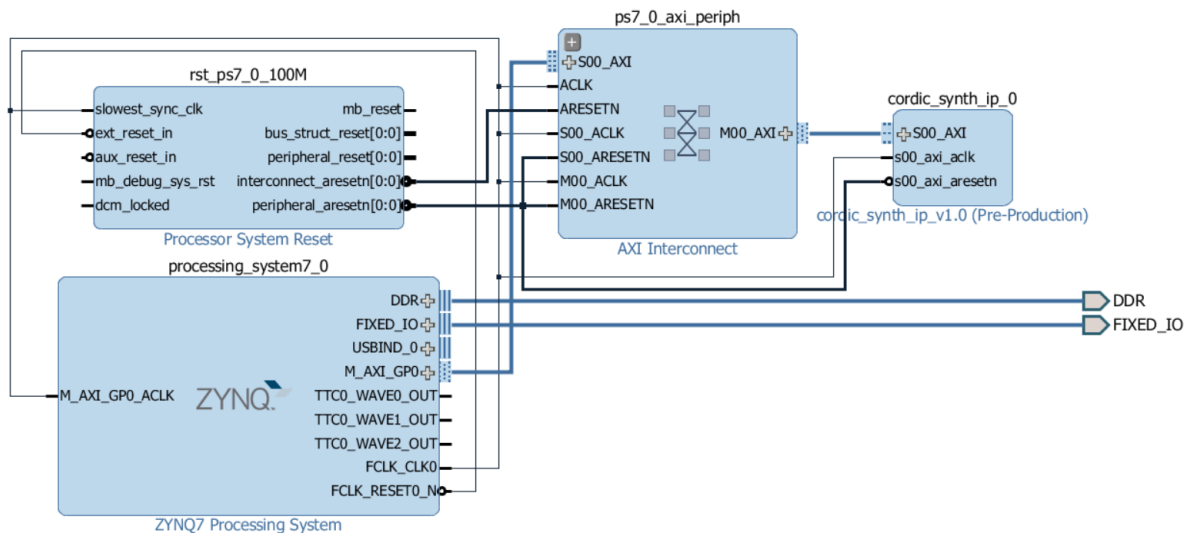
Do przeprowadzenia rzutowania wartości obliczonych funkcji trygonometrycznych przez blok algorytmu CORDIC z przedziału kątów od 0 do $\Pi/2$ na zakres kątów od 0 do 2Π przeznaczony jest ostatni z bloków funkcjonalnych układu o nazwie „cordic_trig”. Moduł ten za pomocą wzorów redukcyjnych i informacji o ćwiartce dla jakiej generowane są wartości dokonuje zamiany wartości funkcji sinus i cosinus i ewentualnie odwraca te wartości względem liczby 0.

Schemat blokowy implementacji algorytmu wraz z sygnałami przedstawia poniższy rysunek:



Rysunek 2 Schemat blokowy implementacji algorytmu generacji sygnału sinusoidalnego za pomocą algorytmu CORDIC

Moduł po implementacji i syntezie został umieszczony w module IPCore AXI4 w celu łatwiejszego połączenia z modułem procesora ZYNQ. Po zbudowaniu projektu z podłączeniem do modułu ZYNQ7 Processing system oraz podłączeniu magistrali AXI, projekt jest gotowy do generacji bitstreamu.



Rysunek 3 Schemat blokowy projektu gotowego do generacji bitstream'u

4. Wnioski i wyniki

Realizacja projektu pozwoliła nam na zgłębieniu metod generacji sygnału sinusoidalnego przy użyciu układu FPGA, które mogą posłużyć jako akceleracja obliczania funkcji trygonometrycznych bądź jako generator sygnału cyfrowego o zadanej częstotliwości.

Wyzwania jakie stawiał projekt umożliwiły poznanie nowych aspektów programowania układów FPGA za pomocą języka Verilog w środowisku Vivado.

Implementacja projektu spełniła wymagane założenia, a sam układ działa poprawnie i zawiera również syntezywalną implementację.

Niestety nie udało się spełnić założenia o zachowaniu błędów wyznaczania wartości funkcji trygonometrycznych na poziomie rozdzielczości wyjściowych wartości. Rozdzielczość wyjściowych wektorów wynosi $6,10352e-5$ natomiast maksymalny otrzymany błąd wyniósł $8.51586e-5$.

Link do repozytorium z kodem źródłowym i sprawozdaniem:

https://github.com/Fysek/Cordic_project