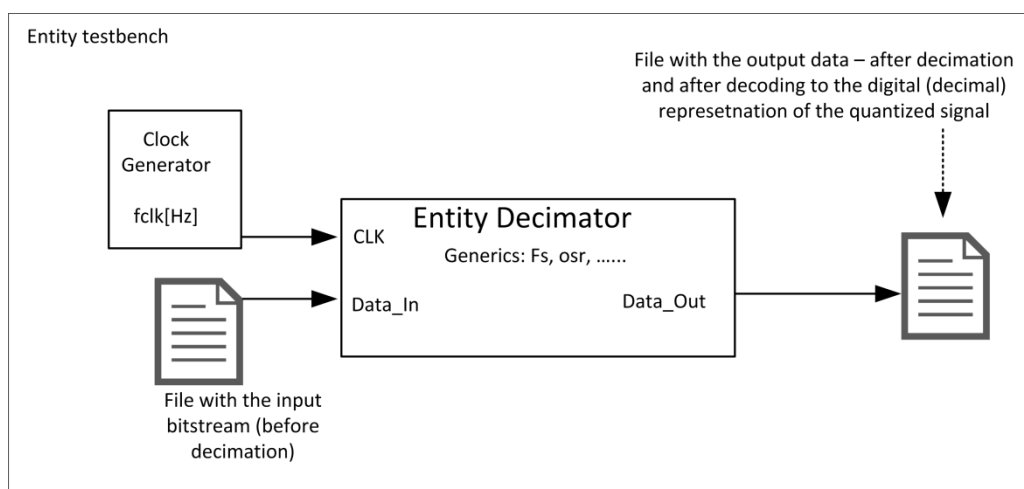


LAB: Testbench układu decymacyjnego**Zakres ćwiczenia:**

Przedmiotem laboratorium jest zaprojektowanie i uruchomienie układu testującego (testbench), który posłuży do testowania układu decymacyjnego. Opis układu należy wykonać w języku VHDL. Środowisko uruchomieniowe MODELSIM (zalecany) lub SYSTEM VISION.

Projektant ma wypełnić podstawowe wymogi projektu, ale jest zobowiązany do opracowania własnej koncepcji realizacji układu. Należy dokładnie opisać przyjęte założenia i wprowadzone ograniczenia.

Rys. 1 przedstawia ogólną strukturę koncepcję układu.



Rys. 1. Struktura testbenchu układu decymacyjnego.

Autor powinien zaprojektować układ według swoich założeń, zdefiniować porty wejściowe i wyjściowe decymatora. Należy odpowiednio wygenerować potrzebne sygnały testujące. Minimalnie wymagane jest jednak, aby układ wczytywał dane z pliku, w którym znajduje się ciąg zer i jedynek (bitstream). W tym ćwiczeniu plik wejściowy, który posłuży do uruchomienia/testowania układu, Autor ćwiczenia wygeneruje samodzielnie (technika dowolna). Należy jednak opisać/wyjaśnić jak skonstruowane są dane wejściowe i jakie są oczekiwane wyniki w pliku wyjściowym. W przyszłości będzie to plik/bitstream generowany przez modulator sigma-delta.

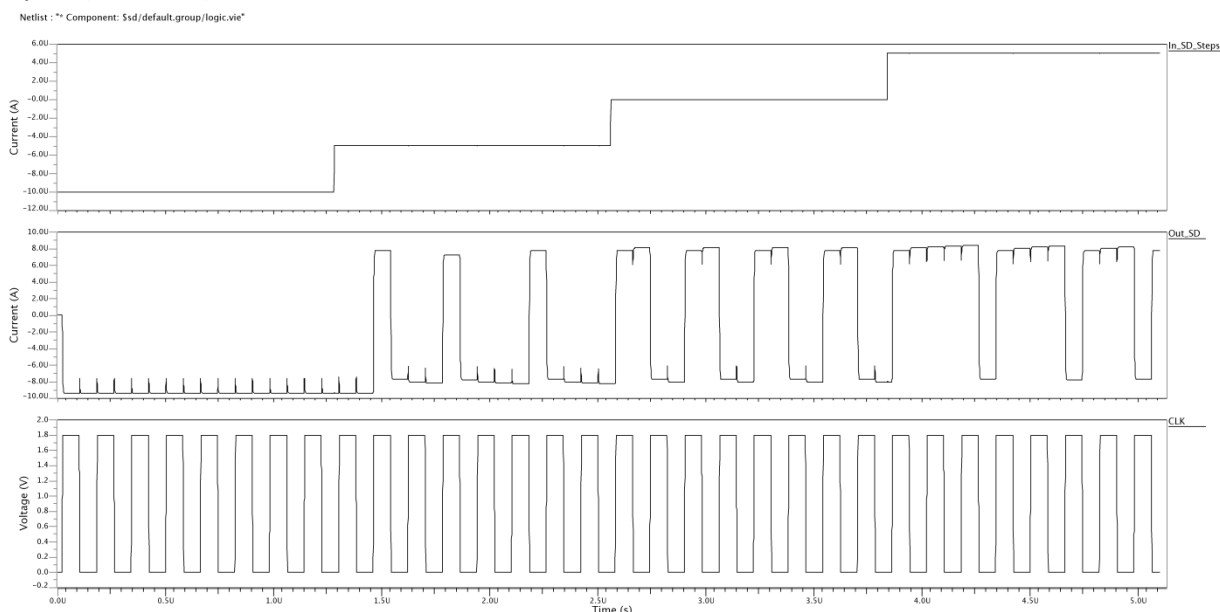
Działanie układu decymacyjnego (w najprostszej postaci) zostało omówione na wykładzie. W tym ćwiczeniu, funkcję decymatora będzie spełniał prosty licznik liczący od wartości środkowej w górę lub w dół w zależności od wartości bitów wejściowych. Należy odpowiednio zaprojektować sposób resetowania (ustawienie w połowie) licznika przy nowym cyklu przetwarzania. Należy również rozwiązać problem ewentualnego „przekręcenia się licznika”.

Układ powinien zapewniać możliwość zadawania częstotliwości taktowania zegara podstawowego oraz współczynnika nadpróbkowania (OSR – OverSampling Ratio) – zdefiniować jako „generic” w VHDL. Należy wyjaśnić przyjętą konwencję taktowania i zależność zegara podstawowego w stosunku do szerokości bitu.

Dokumentacja powinna zawierać:

1. Opis koncepcji układu decymacyjnego i wyjaśnienie jego działania.
2. Opis koncepcji działania całego układu testbenchu – zależności czasowe taktowania.
3. Wyjaśnienie znaczenia/funkcji wszystkich portów (we/wy) oraz parametrów (generic). W kodzie należy nadać wartości domyślne (default) wszystkim parametrom.
4. Kody źródłowe VHDL – Testbenchu – bogato udokumentowany komentarzami.
5. Kod źródłowy układu decymacyjnego - bogato udokumentowany komentarzami.
6. Plik wejściowy (z opisem sposobu jego generacji i wyjaśnienie co jest w nim zakodowane i z jakimi parametrami –OSR)
7. Plik wyjściowy z wyjaśnieniem zawartości.
8. Zrzut ekranu przedstawiający działania układu, który zawiera, co najmniej następujące przebiegi (sygnały):
 - Zegar bazowy (CLK)
 - Bitstream wejściowy (Data_In) – przed decymacją
 - Ciąg wyjściowy liczb całkowitych reprezentujący sygnał wejściowy (Data_Out)

Przykładowe przebiegi wyjaśniające omawiany problem przedstawiono na Rys.2 (sygnał schodkowy) i na Rys. 3 (sinusoida).



Rys. 2. Przykładowe przebiegi. Od dołu - zegar, bitstream wejściowy, przebieg wyjściowy (w naszym ćwiczeniu będą to liczby jak przedstawiono na Rys. 3).

