

# opb2opb\_dwidth

Autor: Ernest Jamro, <http://galaxy.uci.agh.edu.pl/~jamro>

Moduł mostka opb2opb\_dwidth służy do łączenia dwóch różnych magistrali OPB. Moduł posiada dwie magistrale: jedną po stronie slave oraz drugą po stronie master. Magistrala po stronie slave wymusza odpowiednie transfery na magistrali po stronie master.

Moduł opb\_opb\_dwidth umożliwia:

- łączenie dwóch różnych magistral OPB
- konwersję szerokości danych poprzez odpowiednie ustawienie parametrów c\_mdwidth i c\_sdwidth.
- separację dwóch magistral poprzez dodanie bufora FIFO (first-in first-out).
- wykonywanie tylko operacji zapisu lub tylko operacji odczytu poprzez odpowiednie ustawienie parametrów c\_fifo\_wr\_awidth i c\_fifo\_rd\_awidth
- zmianę adresów poprzez odpowiednią zmianę parametru c\_msbadr

## Parametry modułu:

**c\_opb\_dwidth** – szerokość magistrali danych OPB (obu magistral). Szerokość ta jest definiowana przez program EDK i wynosi 32-bity. Aby jednak możliwa była konwersja szerokości magistrali zdefiniowano dodatkowe parametry c\_mdwidth oraz c\_sdwidth, które określają rzeczywistą szerokość magistrali OPB. Dozwolone wartości to 8, 16, 32, przy czym  $c\_opb\_dwidth \geq c\_mdwidth$  oraz  $c\_opb\_dwidth \geq c\_sdwidth$ .

**c\_mdwidth** – rzeczywista szerokość magistrali danych po stronie nadrzędnej (master). Dla  $c\_mdwidth < c\_opb\_dwidth$  bity najmniej znaczące (o wyższych indeksach) magistrali danych są ignorowane w przypadku zapisu oraz ustawione na zero w przypadku odczytu. Jeśli chodzi o magistralę adresową i Byte Enable to zachowują się one tak jakby magistrala danych miała szerokość równą c\_mdwidth. W związku z powyższym nie jest możliwe poprawne przyłączenie do tej samej magistrali urządzeń o różnych rzeczywistych szerokościach magistrali. Dozwolone wartości: 8, 16, 32.  $c\_mdwidth \leq c\_opb\_dwidth$

**c\_sdwidth** – podobnie jak c\_mdwidth ale dotyczy rzeczywistej szerokości magistrali danych po stronie podrzędnej (slave).

**c\_opb\_awidth** – szerokość magistrali adresowej (musi być taka sama dla obu magistral OPB po stronie slave i master)

**c\_fifo\_wr\_awidth** – szerokość magistrali adresowej pamięci FIFO dla zapisu. Pamięć FIFO jest typu distributed RAM dlatego zalecana szerokość magistrali jest 4. Dozwolone wartości:

<0 ścieżka zapisu nie zostanie w ogóle zaimplementowana (urządzenie jest wykorzystywane tylko do odczytu).

= 0 bezpośrednie połączenie (brak pamięci FIFO). Ta opcja jest zalecana w momencie kiedy nie jest używane adresowanie sekwencyjne. Przy braku adresowania sekwencyjnego pamięć FIFO tylko opóźnia działanie układu i zajmuje dodatkową logikę.

> 0 – powoduje dodanie pamięci FIFO dla zapisu. Dane są magazynowane w pamięci FIFO tylko w momencie kiedy sygnał SOPB\_seqAddr jest aktywny. Zalecana wartość to 4 (optymalne wykorzystanie pamięci distributed RAM). Dodanie FIFO powoduje separację magistral (zob. również c\_speedrate) co wpływa pozytywnie na minimalny okres zegara.

**c\_fifo\_rd\_awidth** – podobnie jak c\_fifo\_wr\_awidth ale dotyczy ścieżki odczytu. Zaleca się nie stosowanie zbyt dużych wartości tego parametru w przypadku kiedy występują krótkie transfery blokowe ponieważ układ po stronie master będzie wykonywał dużo bezsensownych odczytów, które będą później anulowane po stronie slave.

**c\_use\_seqAddr** – Powoduje zaimplementowanie logiki adresowania sekwencyjnego. W przypadku kiedy  $\leq 0$  sygnał SOPB\_seqAddr jest ignorowany, w przeciwnym przypadku sygnał SOPB\_seqAddr jest odczytywany oraz generowany jest sygnał M\_seqAddr.

**c\_use\_toutSup** – Powoduje, że urządzenie po stronie Slave będzie generowało sygnał Sl\_toutSup. Parametr ten jest przydatny w szczególności wtedy gdy czas oczekiwania na odpowiedź jest często dłuższy niż 16 cykli zegara oraz gdy zbyt długie przetrzymanie magistrali po stronie Slave nie powoduje skutków ubocznych (np. jest tylko jeden Slave). Użycie wartości domyślnej 0 powoduje brak generacji sygnału Sl\_toutSup,  $>0$  sygnał jest generowany.

**c\_transfer\_size** - w przypadku transferu sekwencyjnego (po stronie master) generowany jest sygnał M\_busLock, który powoduje, że magistrala ta jest zarezerwowana tylko dla tego urządzenia. Dlatego aby uniemożliwić przywłaszczenie sobie magistrali na zbyt długi okres czasu po przetransferowaniu c\_transfer\_size danych oraz w przypadku gdy inne urządzenie żąda dostępu do magistrali, magistrala jest zwalniana.

**c\_speedrate** – określa szybkość pracy. Czym mniejsza wartość (0) układ pracuje szybciej. Dlatego zaleca się używanie początkowo wartości 0. W przypadku kiedy układ nie spełnia wymagań czasowych tzn. minimalny okres zegara jest większy od wymaganego (zob. Static Timing Analyser) i maksymalne opóźnienie jest związane z tym modulem (a dokładniej ze stroną slave – ten parametr nie wpływa na stronę master dla  $\text{fifo\_size} > 0$ ) to należy zwiększyć wartość tego współczynnika. Dla wartości 1 generowany jest dodatkowy cykl zegara oczekiwania w przypadku transferu nie sekwencyjnego. W przypadku wartości 2 generowany jest dodatkowy cykl zegara oczekiwana również dla transferów sekwencyjnych. Czym większa wartość tego współczynnika tym większa powinna być częstotliwość pracy układu ale więcej taktów zegara potrzebnych na pojedynczy transfer, ponadto ze wzrostem tego współczynnika wzrasta powierzchnia zajmowana przez ten moduł. Dozwolone wartości: 0, 1, 2.

**C\_BASEADDR – C\_HIGHADDR** - adres po stronie slave, na który urządzenie slave odpowie i prześle go dalej na stronę master (zob. C\_MSBADDR).

**C\_MSBADDR** – określa najbardziej znaczące linie adresu po stronie master. W przypadku kiedy C\_MSBADDR= „FFFF\_FFFF” (czyli MSBADDR pozostaje niezmieniona), urządzenie master odczytuje lub zapisuje dane na stronie master bez zmiany adresu czyli zgodnie z adresem po stronie slave (adres mieści się w zakresie: C\_BASEADDR – C\_HIGHADDR). W przypadku kiedy wartość C\_MSBADDR zostanie zmieniona na inną niż „FFFF\_FFFF” to dane po stronie master są transferowane na zmieniony adres. Zmianie ulegają tylko najbardziej znaczące bity magistrali adresowej. W konsekwencji po stronie master dane transferowane są pod lokacje adresowe w zakresie od C\_MSBADDR do C\_MSBADDR + C\_HIGHADDR - C\_BASEADDR. Jeżeli strona slave wymusza transfer pod adres *sadr* to po stronie master zostanie wymuszony adres:  $\text{madr} = \text{C\_MSBADDR} + \text{sadr} - \text{C\_BASEADDR}$ . Ilustruje to poniższy przykład: dla C\_BASEADDR= 0x100, C\_HIGHADDR= 0x1FF oraz C\_MSBADDR= 0x400 to zakres adresowy wystawiany przez stronę master jest: 0x400 – 0x4FF. Dana wysłana pod adres 0x123 po stronie slave zostanie zapisana pod adres 0x423.