

Izučavanje Avalon MM interfejsa

Uvod

Avalon interfejsi pojednostavljaju dizajn sistema omogućavajući lakše povezivanje komponenata u Intel FPGA sistemu. Porodica interfejsa Avalon definiše interfejse koji su prikladni za strimovanje podataka velike brzine, čitanje i pisanje registara i memorije i kontrolu uređaja izvan čipa.

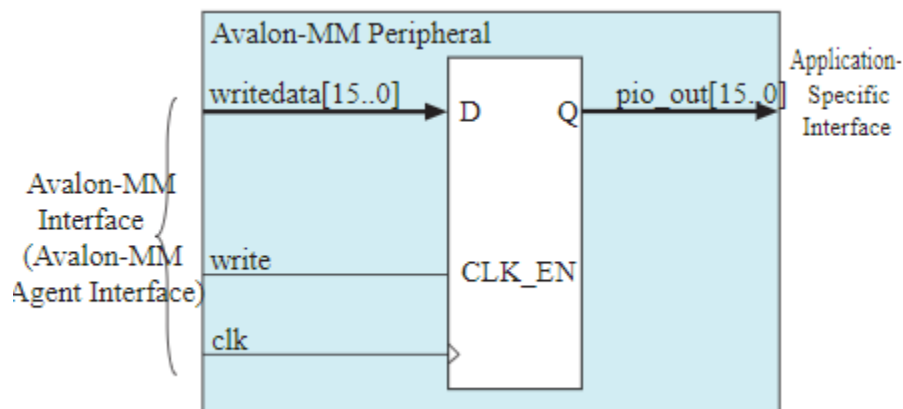
Pored navedenog, možemo da ugradimo Avalon interfejse u prilagođene komponente, poboljšavajući interoperabilnost dizajna.

U kontekstu našeg projektnog zadatka, od najvećeg značaja nam je Avalon MM interfejs (memorjski mapiran interfejs) koji služi za čitanje/pisanje i zasnovan je na adresi tipičnoj za Host-Agent veze.

Neke od najznačajnijih komponenata koje uključuju interfejse mapirane u memoriji su:

- Mikroprocesori
- Memorije
- UART
- DMA
- Tajmeri

Avalon MM interfejsi se kreću od jednostavnijih do složenijih.



Slika 1.0 *Primjer Agent komponente*

16-bitna I/O periferij opšte namjene prikazana na prethodnoj slici odgovaraju samo na zahtjeve za pisanje.

Uloge signala u Avalon MM interfejsu

Uloga signala jeste da se definišu tipovi signala koje Avalon memorijski mapirani portovi hosta i agenta dozvoljavaju.

Minimalni zahtjev, koji mora biti ispunjen za Avalon memorijski mapirani interfejs, su **readdata** za read-only interfejs ili **writedata** za write-only interfejs

Navešćemo neke od najznačajnijih signala potrebnih za izradu našeg projektnog zadatka:

Signal role	Width	Direction	Description
address	1-64	Host->Agent	Predstavlja bajt adresu, sa strane hosta
byteenable byteenable_n	2,4,8,16,32,64,128	Host->Agent	Omogućava 1 ili više traka tokom prenosa na interfejsima širine veće od 8 bita.
read read_n	1	Host->Agent	Zahtjev za čitanje.
readdata	8,16,32,64,128,256	Host->Agent	Služi za čitanje podataka.
write write_n	1	Host->Agent	Zahtjev za upisivanje podataka.
writedata	8,16,32,64,128,256	Host->Agent	Služi za upisivanje podataka.
waitrequest waitrequest_n	1	Agent->Host	Agent potvrđuje zahtjev za čekanjem kada ne može odgovoriti na zahtjev za čitanje ili pisanje. Host čeka dok ne bude omogućen nastavak prenosa.

Neki Avalon MM signali mogu biti aktivni na visok ili nizak logički nivo. Kada je signal aktivan na nizak logički nivo, naziv signala se završava sa **_n**.

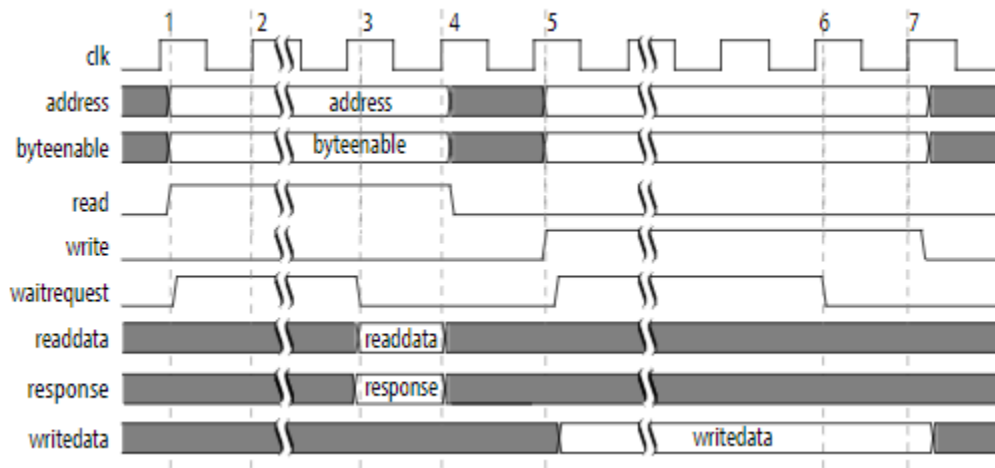
Avalon MM interfejs je sinhroni. Svaki Avalon MM interfejs je sinhronizovan sa povezanim interfejsom clk. Signali mogu biti kombinovani ako se pokreću sa izlaza registra koji su sinhroni sa signalom takta.

Prenos podataka preko Avalon MM interfejsa

Transfer podataka ili prenos se odvija između Avalon MM interfejsa i interkonekcije. I Host i Agent su dio prenosa podataka. Avalon MM Host pokreće prenos, a Avalon MM Agent odgovara.

Tipično, Avalon MM podržava read and write transfer sa agent-controlled waitrequest signalom. Agent može zaustaviti međusobno povezivanje, zahtjevan broj ciklusa, postavljajući waitrequest signal.

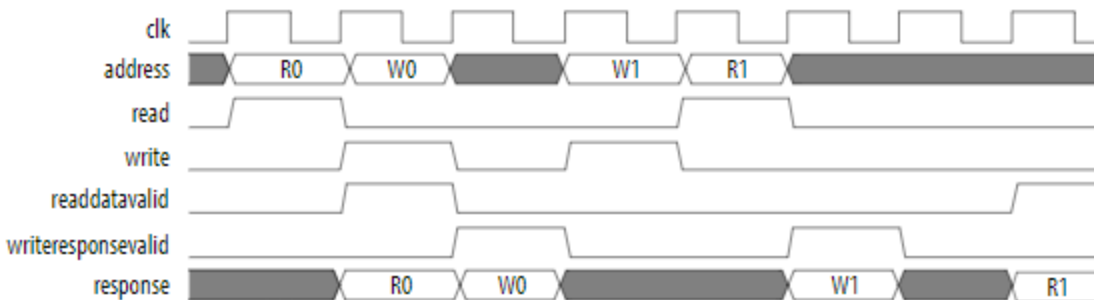
Agent tipično prima adress, byteenable, read ili write i writedata, nakon rastuće ivice signala.



Slika 1.1 *Read and write transfer sa waitrequest-om*

Kada Agent potvrdi waitrequest, prenos se odgađa. U toku potvrde waitrequest signala, address i drugi kontrolni signali se ne mijenjaju. Ne postoji ograničenje koliko dugo Agent može da pauzira prenos podataka, stoga je neophodno osigurati da Agent ne potvrđuje waitrequest na neodređeno vrijeme.

Specifikacije Avalon interfejsa garantuju da komande istom Agentu stižu do tog Agentu po redoslijedu izdavanja komande. Odgovori se vraćaju u redoslijedu izdavanja naredbne, bez obzira da li su naredbe čitanja ili pisanja za iste ili različite Agente.



Slika 1.2 *Read and write responses diagram*

Na prethodnoj slici prikazan je vremenski dijagram odgovora za Avalon MM interfejs. Budući da interfejs za čitanje i pisanje dijeli signal odgovora, Avalon interfejs ne može da prihvati read odgovor i write odgovor u istom vremenskom ciklusu.

Da bismo riješili ovaj problem potrebno je da prvo pročitamo jedan odgovor, pošaljemo jedan odgovor za svaki pročitani podatak, a zatim napišemo odgovor i posaljemo odgovor za svaku naredbu za pisanjem. Agent šalje odgovor nakon primanja posljednje naredbe za pisanjem.

Avalon MM Agent adresiranje

Dinamičko određivanje veličine magistrale upravlja podacima tokom prenosa Host-Agent. Ako je širina podataka Hosta veća od podataka Agent, riječi u adresnom prostoru Hosta se mapiraju na više lokacija u adresnom prostoru Agent.

Agenti moraju imati podatke od 8, 16, 32, 64, 128, 256, 512 ili 1024 bita.

U narednoj tabeli prikazano je poravnanje za podatke Agent, različitih širina, unutar 32-bitnog Hosta.

U ovoj tabeli, OFFSET se odnosi na pomak veličine riječi Agent u njegovom adresnom prostoru.

Host Byte Address (1)	Access	32-Bit Host Data		
		When Accessing an 8-Bit Agent Interface	When Accessing a 16-Bit Agent Interface	When Accessing a 64-Bit Agent Interface
0x00	1	OFFSET[0] 7..0	OFFSET[0] 15..0 (2)	OFFSET[0] 31..0
	2	OFFSET[1] 7..0	OFFSET[1] 15..0	—
	3	OFFSET[2] 7..0	—	—
	4	OFFSET[3] 7..0	—	—
0x04	1	OFFSET[4] 7..0	OFFSET[2] 15..0	OFFSET[0] 63..32
	2	OFFSET[5] 7..0	OFFSET[3] 15..0	—
	3	OFFSET[6] 7..0	—	—
	4	OFFSET[7] 7..0	—	—
0x08	1	OFFSET[8] 7..0	OFFSET[4] 15..0	OFFSET[1] 31..0
	2	OFFSET[9] 7..0	OFFSET[5] 15..0	—
	3	OFFSET[10] 7..0	—	—
	4	OFFSET[11] 7..0	—	—
0x0C	1	OFFSET[12] 7..0	OFFSET[6] 15..0	OFFSET[1] 63..32
	2	OFFSET[13] 7..0	OFFSET[7] 15..0	—
	3	OFFSET[14] 7..0	—	—
	4	OFFSET[15] 7..0	—	—
And so on		And so on	And so on	And so on