ECE/CSC 506: Architecture Of Parallel Computers

Homework 6 – Fall 2015

Q1. Following the example in the class, assume a 3-processor multiprocessor system with directory-based coherence protocol. Assume that the cost of a network transaction is solely determined by the number of sequential hops involved in the transaction. Each hop takes 50 cycles to complete, while a cache hit costs 1 cycle. Furthermore, ignore NACK traffic and speculative replies.

Display the state transition of all the 3 caches, the directory content and its state, and the network messages generated for the reference stream shown in the tables.

a. Assume a full-bit vector directory scheme. Each cache has MESI states, and the directory has 3 states: U for unowned, EM for exclusive or modified, and S for shared.

MemRef	P1	P2	P3	Directory content	All network messages	Cost
r1	E	-	-	EM,100	Read(P1->H), ReplyD(H->P1)	100
w1	M	-	-	EM,100	-	1
r2	S	S	-	S,110	Read(P2->H), WB+Int(H->P1), Flush(P1->H,P2)	150

w2	I	M	-	EM,010	Upgr(P2->H),
					Reply(H->P2)//Inv(H->P1),
					InvAck(P1->P2)
						150

r3	I	S	S	S,011	Read(P3->H), WB+Int(H->P2), Flush(P2->H,P3)	150
r1	S	S	S	S,111	Read(P1->H), ReplyD(H->P1)	100

w3	I	I	M	EM,001	Upgr(P3->H),
					Reply(H->P3)//Inv(H->P1,P2),
					InvAck(P1->P3)//InvAck(P2->P3)
						150

r3	I	I	M	EM,001	-	1
w1	M	I	I	EM,100	ReadX(P1->H), Reply(H->P1)//Inv(H->P3), Flush+InvAck(P3->P1)	150
					Total	952

ECE/CSC 506: Architecture Of Parallel Computers

Homework 6 – Fall 2015

Q1. Following the example in the class, assume a 3-processor multiprocessor system with directory-based coherence protocol. Assume that the cost of a network transaction is solely determined by the number of sequential hops involved in the transaction. Each hop takes 50 cycles to complete, while a cache hit costs 1 cycle. Furthermore, ignore NACK traffic and speculative replies.

Display the state transition of all the 3 caches, the directory content and its state, and the network messages generated for the reference stream shown in the tables.

a. Assume a full-bit vector directory scheme. Each cache has MESI states, and the directory has 3 states: U for unowned, EM for exclusive or modified, and S for shared.

MemRef	P1	P2	P3	Directory content	All network messages	Cost
r1	E	-	-	EM,100	Read(P1->H), ReplyD(H->P1)	100
w1	M	-	-	EM,100	-	1
r2	S	S	-	S,110	Read(P2->H), WB+Int(H->P1), Flush(P1->H,P2)	150

w2	I	M	-	EM,010	Upgr(P2->H),
					Reply(H->P2)//Inv(H->P1),
					InvAck(P1->P2)
						150

r3	I	S	S	S,011	Read(P3->H), WB+Int(H->P2), Flush(P2->H,P3)	150
r1	S	S	S	S,111	Read(P1->H), ReplyD(H->P1)	100

w3	I	I	M	EM,001	Upgr(P3->H),
					Reply(H->P3)//Inv(H->P1,P2),
					InvAck(P1->P3)//InvAck(P2->P3)
						150

r3	I	I	M	EM,001	-	1
w1	M	I	I	EM,100	ReadX(P1->H), Reply(H->P1)//Inv(H->P3), Flush+InvAck(P3->P1)	150
					Total	952

b. Assume a full-bit vector directory scheme. Each cache has MESI states, and the directory has four states: E for exclusive, M for modified, S for shared, U for unowned. You can assume a new message type “UpgrAck" that is used by the home to reply to an upgrade request from the owner of a cache line.

Mem Ref	P1	P2	P3	Directory content	All network messages	Cost
r1	E	-	-	E,100	Read(P1->H), ReplyD(H->P1)	100
w1	M	-	-	M,100	Upgr(P1->H), UpgrAck(H->P1)	100
r2	S	S	-	S,110	Read(P2->H), WB+Int(H->P1), Flush(P1->H,P2)	150

w2	I	M	-	M,010	Upgr(P2->H),
					Reply(H->P2)//Inv(H->P1),
					InvAck(P1->P2)
						150

r3	I	S	S	S,011	Read(P3->H), WB+Int(H->P2), Flush(P2->H,P3)	150
r1	S	S	S	S,111	Read(P1->H), ReplyD(H->P1)	100

w3	I	I	M	M,001	Upgr(P3->H),
					Reply(H->P3)//Inv(H->P1,P2),
					InvAck(P1->P3)//InvAck(P2->P3)
						150

r3	I	I	M	M,001	-	1
w1	M	I	I	EM,100	ReadX(P1->H), Reply(H->P1)//Inv(H->P3), Flush+InvAck(P3->P1)	150
					Total	1051

Q2. Suppose a 4-processor multiprocessor system uses a directory-based coherence protocol with full bit vector. The directory keeps U, EM, and S states, while the cache maintains MESI states. Assume that cost of a network transaction is solely determined by the number of protocol hops involved in the transaction, and each hop has a latency of 60 cycles.

Suppose that a parallel program incurs the following accesses to a single block address: r1, r3, w2, and r4, where r indicates a read request, w indicates a write request, and the number indicates the processor issuing the request. Suppose that the requests are issued almost simultaneously to the directory, but the directory receives them in the following order: r1, r3, w2, r4. Assume that the occupancy of the directory (i.e., the length of time the directory looks up and updates the directory state) is 15 cycles.

b. Assume a full-bit vector directory scheme. Each cache has MESI states, and the directory has four states: E for exclusive, M for modified, S for shared, U for unowned. You can assume a new message type “UpgrAck" that is used by the home to reply to an upgrade request from the owner of a cache line.

Mem Ref	P1	P2	P3	Directory content	All network messages	Cost
r1	E	-	-	E,100	Read(P1->H), ReplyD(H->P1)	100
w1	M	-	-	M,100	Upgr(P1->H), UpgrAck(H->P1)	100
r2	S	S	-	S,110	Read(P2->H), WB+Int(H->P1), Flush(P1->H,P2)	150

w2	I	M	-	M,010	Upgr(P2->H),
					Reply(H->P2)//Inv(H->P1),
					InvAck(P1->P2)
						150

r3	I	S	S	S,011	Read(P3->H), WB+Int(H->P2), Flush(P2->H,P3)	150
r1	S	S	S	S,111	Read(P1->H), ReplyD(H->P1)	100

w3	I	I	M	M,001	Upgr(P3->H),
					Reply(H->P3)//Inv(H->P1,P2),
					InvAck(P1->P3)//InvAck(P2->P3)
						150

r3	I	I	M	M,001	-	1
w1	M	I	I	EM,100	ReadX(P1->H), Reply(H->P1)//Inv(H->P3), Flush+InvAck(P3->P1)	150
					Total	1051

Q2. Suppose a 4-processor multiprocessor system uses a directory-based coherence protocol with full bit vector. The directory keeps U, EM, and S states, while the cache maintains MESI states. Assume that cost of a network transaction is solely determined by the number of protocol hops involved in the transaction, and each hop has a latency of 60 cycles.

Suppose that a parallel program incurs the following accesses to a single block address: r1, r3, w2, and r4, where r indicates a read request, w indicates a write request, and the number indicates the processor issuing the request. Suppose that the requests are issued almost simultaneously to the directory, but the directory receives them in the following order: r1, r3, w2, r4. Assume that the occupancy of the directory (i.e., the length of time the directory looks up and updates the directory state) is 15 cycles.

a) What is the latency to complete the processing of all the requests using a home-centric approach?
b) What is the latency to complete the processing of all the requests using a requestor-assisted approach, which tries to overlap the request processing as much as possible? Assume that the occupancy of the directory (i.e., the length of time the directory is unable to serve a request) is 10 cycles.

Answer:

Suppose that r1, r3, w3, and r4 were issued at time 0.

Part (a): Home-centric Approach

Request	Time	Messages
r1	(0,60) (60,75) (75,135) (135,195)	Read(P1->Home) Directory Occupancy ReplyD(Home->P1) Done(P1->Home)
r3	(0,60) (60,195) (195,210) (210,270) (270,330) (330,390)	Read(P3->Home) Waiting Directory Occupancy Int(Home->P1) Flush(P1->Home, P3) Done(P3->Home)
w2	(0,60) (60,390) (390,405) (405,465) (465,525) (525,585)	ReadX(P2->Home) Waiting Directory Occupancy Inv(Home->P1, P3) InvAck(P1, P3->P2) Done(P2->Home)
r4	(0,60) (60,585) (585,600) (600,660) (660,720) (720,780)	Read (P4->Home) Waiting Directory Occupancy Int(Home->P2) Flush(P2->P4, Home) Done(P4->Home)

Part (b): Requestor-Assisted Approach

Request	Time	Messages
r1	(0,60) (60,70) (70,130)	Read(P1->Home) Directory Occupancy ReplyD(Home->P1)
r3	(0,60) (60,70) (70,80) (80,140) (140,200)	Read(P3->Home) Waiting Directory Occupancy Int(Home->P1) Flush(P1->Home, P3)
w2	(0,60) (60,200)	ReadX(P2->Home) Waiting

a) What is the latency to complete the processing of all the requests using a home-centric approach?
b) What is the latency to complete the processing of all the requests using a requestor-assisted approach, which tries to overlap the request processing as much as possible? Assume that the occupancy of the directory (i.e., the length of time the directory is unable to serve a request) is 10 cycles.

Answer:

Suppose that r1, r3, w3, and r4 were issued at time 0.

Part (a): Home-centric Approach

Request	Time	Messages
r1	(0,60) (60,75) (75,135) (135,195)	Read(P1->Home) Directory Occupancy ReplyD(Home->P1) Done(P1->Home)
r3	(0,60) (60,195) (195,210) (210,270) (270,330) (330,390)	Read(P3->Home) Waiting Directory Occupancy Int(Home->P1) Flush(P1->Home, P3) Done(P3->Home)
w2	(0,60) (60,390) (390,405) (405,465) (465,525) (525,585)	ReadX(P2->Home) Waiting Directory Occupancy Inv(Home->P1, P3) InvAck(P1, P3->P2) Done(P2->Home)
r4	(0,60) (60,585) (585,600) (600,660) (660,720) (720,780)	Read (P4->Home) Waiting Directory Occupancy Int(Home->P2) Flush(P2->P4, Home) Done(P4->Home)

Part (b): Requestor-Assisted Approach

Request	Time	Messages
r1	(0,60) (60,70) (70,130)	Read(P1->Home) Directory Occupancy ReplyD(Home->P1)
r3	(0,60) (60,70) (70,80) (80,140) (140,200)	Read(P3->Home) Waiting Directory Occupancy Int(Home->P1) Flush(P1->Home, P3)
w2	(0,60) (60,200)	ReadX(P2->Home) Waiting

	(200,210) (210,270) (270,330)	Directory Occupancy Inv(Home->P1, P3) InvAck(P1, P3->P2)
r4	(0,60) (60,210) (210,220) (220,280) (280,330) (330,390)	Read (P4->Home) Waiting Directory Occupancy Int(Home->P2) Waiting Flush(P2->P4, Home)

	(200,210) (210,270) (270,330)	Directory Occupancy Inv(Home->P1, P3) InvAck(P1, P3->P2)
r4	(0,60) (60,210) (210,220) (220,280) (280,330) (330,390)	Read (P4->Home) Waiting Directory Occupancy Int(Home->P2) Waiting Flush(P2->P4, Home)