Instruction	Cycle																			
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
L.D F0, 0(R1)	IF	IS	EX	M	WB	C														
MUL.D F4, F0, F2		IF	IS	S	E1	E2	E3	E4	E5	E6	WB	C								
S.D F4, 0(R1)			IF	IS	EX	S	S	S	S	S			C	Store has no memory stage because it writes memory during commit						
DADDIU R1, R1, #-8				IF	IS	E1	E2	WB						C						
BNE R1, R2, Loop					IF	IS	S	*							C					
L.D F0, 0(R1)						IF	IS	EX	M	WB						C				
MUL.D F4, F0, F2							IF	IS	S	E1	E2	E3	E4	E5	E6	WB	C			
S.D F4, 0(R1)								IF	IS	EX	S	S	S	S	S			C		
DADDIU R1, R1, #-8									IF	IS	E1	E2	WB						C	
BNE R1, R2, Loop										IF	IS	S	*							C

* Branch outcome can be determined once both operands are available

Example assumes following latencies:
2 cycles for memory operations (1 EX, 1 MEM)
2 cycles for addition (DADDIU)
6 cycles for multiply (MUL.D)