

벡터 부동 소수점 명령어 세트

빠른 참조 카드

표기 규칙 풀이							
{C}	조건 필드 표 참조 F32(단정밀도) 또는 F64(배정밀도) 단정밀도, 배정밀도 또는 반정밀도(F16) 단정밀도 또는 배정밀도 부동 소수점 부호 있는 정수 또는 부호 없는 정수 FPSCR 또는 FPSID 2: VFPv2 이상, 3: VFPv3 이상, 3H: VFPv3 이상(반정밀도 확장)	<P>	<fpconst> Fd, Fn, Fm {E} {R} <VFPregs> <fbits> <type>	+/- m * 2 <sup>-n</sup> 여기서 m 및 n은 정수, 16 ≤ m ≤ 31, 0 ≤ n ≤ 7 Sd, Sn, Sm(단정밀도) 또는 Dd, Dn, Dm(배정밀도) E가 있는 경우: 모든 NaN에 대해 예외 발생. E가 없는 경우: NaN 신호에 대해서만 예외 발생 FPSCR 반올림 모드 사용하고 그렇지 않으면 0으로 반올림 콤마로 구분되고 중괄호 { }로 묶인 consecutive VFP 레지스터 목록 고정 소수점 숫자에 있는 소수 비트 수, 0-16 또는 1-32 부호가 있거나 부호가 없는 16비트 또는 32비트의 경우 s16, s32, u16 또는 u32			
<P>							
S, D, H							
F							
SI, UI <VFPsysreg> \$							
연산		\$	어셈블리	예외	동작	메모	
벡터 산술	곱하기 및 부정 및 누산 부정 및 누산 및 빼기 부정 및 빼기 더하기 빼기 나누기 절대값 음수 제곱근		VMUL{C}.<P> Fd, Fn, Fm VNMUL{C}.<P> Fd, Fn, Fm VMLA{C}.<P> Fd, Fn, Fm VMLS{C}.<P> Fd, Fn, Fm VNMLS{C}.<P> Fd, Fn, Fm VNMLA{C}.<P> Fd, Fn, Fm VADD{C}.<P> Fd, Fn, Fm VSUB{C}.<P> Fd, Fn, Fm VDIV{C}.<P> Fd, Fn, Fm VABS{C}.<P> Fd, Fm VNEG{C}.<P> Fd, Fm VSQRT{C}.<P> Fd, Fm	IO, OF, UF, IX IO, OF, UF, IX IO, OF, UF, IX IO, OF, UF, IX IO, OF, UF, IX IO, OF, UF, IX IO, OF, IX IO, OF, IX IO, DZ, OF, UF, IX  IO, IX	Fd := Fn * Fm Fd := - (Fn * Fm) Fd := Fd + (Fn * Fm) Fd := Fd - (Fn * Fm) Fd := - Fd + (Fn * Fm) Fd := - Fd - (Fn * Fm) Fd := Fn + Fm Fd := Fn - Fm Fd := Fn / Fm Fd := abs(Fm) Fd := - Fm Fd := sqrt(Fm)		
	스칼라 비교		VCMP{E}{C}.<P> Fd, Fm VCMP{E}{C}.<P> Fd, #0.0	IO IO	Fd - Fm에서 FPSCR 플래그 설정 Fd - 0에서 FPSCR 플래그 설정	VMRS APSR_nzcv, FPSCR을 사용하여 플래그 전송	
	스칼라 변환	단정밀도에서 배정밀도로 변환 배정밀도에서 단정밀도로 변환 부호 없는 정수에서 부동 소수점 숫자로 변환 부호 있는 정수에서 부동 소수점 숫자로 변환 부동 소수점 숫자에서 부호 없는 정수로 변환 부동 소수점 숫자에서 부호 있는 정수로 변환 고정 소수점 숫자에서 부동 소수점 숫자로 변환 부동 소수점 숫자에서 고정 소수점 숫자로 변환		VCVT{C}.F64.F32 Dd, Sm VCVT{C}.F32.F64 Sd, Dm VCVT{C}.<P>.U32 Fd, Sm VCVT{C}.<P>.S32 Fd, Sm VCVT{R}{C}.U32.<P> Sd, Fm VCVT{R}{C}.S32.<P> Sd, Fm	IO IO, OF, UF, IX IX IX IO, IX IO, IX	Dd := convertStoD(Sm) Sd := convertDtoS(Dm) Fd := convertUtoF(Sm) Fd := convertStoF(Sm) Sd := convertFtoUI(Fm) Sd := convertFtoSI(Fm)	
		3	VCVT{C}.<P>.<type> Fd, Fd, #<fbits>	IO, IX	Fd := convert<type>toF(Fd)	소스는 Fd의 아래쪽 16비트나 32비트에 있음 대상은 Fd의 아래쪽 16비트나 32비트에 있음 대상은 Sd의 맨 위 16비트에 있음 대상은 Sd의 맨 아래 16비트에 있음 소스는 Sm의 맨 위 16비트에 있음 소스는 Sm의 맨 아래 16비트에 있음	
		3	VCVT{C}.<type>.<P> Fd, Fd, #<fbits>	IO, IX	Fd := convertFto<type>(Fd)		
		3H	VCVTT{C}.F16.F32 Sd, Sm	ID, IO, OF, UF, IX	Sd:=convertStoH(Sm)		
		3H	VCVTB{C}.F16.F32 Sd, Sm	ID, IO, OF, UF, IX	Sd:=convertStoH(Sm)		
		3H	VCVTT{C}.F32.F16 Sd, Sm	ID, IO, OF, UF, IX	Sd:=convertHtoS(Sm)		
		3H	VCVTB{C}.F32.F16 Sd, Sm	ID, IO, OF, UF, IX	Sd:=convertHtoS(Sm)		
		상수 삽입	레지스터에 상수 삽입	3	VMOV{C}.<P> Fd, #<fpconst>		Fd := <fpconst>
레지스터 전송	VFP 레지스터 복사		VMOV{C}.<P> Fd, Fm		Fd := Fm		
	ARM 레지스터를 단정밀도 레지스터로		VMOV{C} Sn, Rd		Sn := Rd		
	단정밀도 레지스터를 ARM 레지스터로		VMOV{C} Rd, Sn		Rd := Sn		
	두 개의 ARM 레지스터를 두 개의 단정밀도 레지스터로	2	VMOV{C} Sn, Sm, Rd, Rn		Sn := Rd, Sm := Rn	Sm은 S(n+1)이어야 함	
	두 개의 ARM 레지스터를 두 개의 단정밀도 레지스터로	2	VMOV{C} Rd, Rn, Sn, Sm		Rd := Sn, Rn := Sm	Sm은 S(n+1)이어야 함	
	두 개의 ARM 레지스터를 배정밀도 레지스터로	2	VMOV{C} Dm, Rd, Rn				
	배정밀도 레지스터를 두 개의 ARM 레지스터로	2	VMOV{C} Rd, Rn, Dm		Dm[31:0] := Rd, Dm[63:32] := Rn Rd := Dm[31:0], Rn := Dm[63:32]		
	ARM 레지스터를 배정밀도 레지스터의 하위 반으로		VMOV{C} Dn[0], Rd		Dn[31:0] := Rd		
배정밀도 레지스터의 하위 반을 ARM 레지스터로		VMOV{C} Rd, Dn[0]		Rd := Dn[31:0]			

벡터 부동 소수점 명령어 세트

빠른 참조 카드

연산		§	어셈블러	예외	동작	메모
레지스터 전송 (계속)	ARM 레지스터를 배정밀도 레지스터의 상위 반으로 배정밀도 레지스터의 상위 반을 ARM 레지스터로 ARM 레지스터를 VFP 시스템 레지스터로 VFP 시스템 레지스터를 ARM으로 FPSCR 플래그를 APSR로		VMOV{C} Dn[1], Rd VMOV{C} Rd, Dn[1] VMSR{C} <VFPsysreg>, Rd VMRS{C} Rd, <VFPsysreg> VMRS{C} APSR_nzcv, FPSCR		Dn[63:32] := Rd Rd := Dn[63:32] VFPsysreg := Rd Rd := VFPsysreg APSR flags := FPSCR flags	

연산		§	어셈블러	동의어	동작
VFP 레지스터 저장	단일 레지스터 단일, PC 기준 다중 레지스터, 인덱싱되지 않음 / 이후 증가 이전 감소 스택에 푸시		VSTR{C} Fd, [Rn{, #<immed>}] VSTR{C} Fd, <label> VSTM{C} Rn{!}, <VFPregs> VSTMDB{C} Rn!, <VFPregs> VPUSH{C} <VFPregs>	VSTMIA, VSTMEA VSTMFD (full descending) VSTMFD SP!	[address] := Fd. 즉치값 범위 0 ~ 1020, 4의 배수  Rn의 주소에서 시작하여 VFP 레지스터 목록 저장
VFP 레지스터 로드	단일 레지스터 단일, PC 기준 다중 레지스터, 인덱싱되지 않음 / 이후 증가 이전 감소 스택에서 팝		VLDR{C} Fd, [Rn{, #<immed>}] VLDR{C} Fd, <label> VLDM{C} Rn{!}, <VFPregs> VLDMDB{C} Rn!, <VFPregs> VPOP{C} <VFPregs>	VLDMIA, VLDMFD VLDMEA (빈 오름차순) VLDM SP!	Fd := [address]. 즉치값 범위 0 ~ 1020, 4의 배수  Rn의 주소에서 시작하여 VFP 레지스터 목록 로드

FPSCR 형식								반올림		(스트라이드-1)*3				백터 길이 - 1					예외 트랩에서 비트 활성화									누적 예외 비트				
31	30	29	28	27	26	25	24	23	22	21	20		18	17	16	15			12	11	10	9	8	7				4	3	2	1	0
N	Z	C	V	QC	AHP	DB	FZ	RMODE		STRIDE			LEN			IDE			IXE	UFE	OFE	DZE	IOE	IDC				IXC	UFC	OFC	DZC	IOC
FZ: 1=0으로 플러시 모드								반올림: 0=근사값으로 반올림, 1= +∞로 반올림, 2=-∞로 반올림, 3=0으로 반올림										(백터 길이 * 스트라이드)는 배정밀도 피연산자의 경우 4를 초과하지 않아야 함 (제공되지 않음)														

조건 플드										예외	
니모닉	설명(VFP)	설명(ARM 또는 Thumb)		니모닉	설명(VFP)	설명(ARM 또는 Thumb)					
EQ	같음	같음		HI	보다 큼 또는 순서가 지정되지 않음	부호 없는 높음	ID 비정규 입력				
NE	같지 않음 또는 순서가 지정되지 않음	같지 않음		LS	작거나 같음	부호 없는 낮거나 같음		IO 잘못된 연산			
CS / HS	크거나 같음 또는 순서가 지정되지 않음	carry 설정/부호 없는 높거나 같음		GE	크거나 같음	부호 있으면서 크거나 같음	OF 오버플로				
CC / LO	보다 작음	carry 지우기/부호 없는 낮음		LT	보다 작음 또는 순서가 지정되지 않음	부호 있으면서 보다 작음	UF 언더플로				
MI	보다 작음	음수		GT	보다 큼	부호 있으면서 보다 큼	IX 정확하지 않은 결과				
PL	크거나 같음 또는 순서가 지정되지 않음	양수 또는 0		LE	작거나 같음 또는 순서가 지정되지 않음	부호 있으면서 작거나 같음	DZ 0으로 나누기				
VS	순서가 지정되지 않음(하나 이상의 NaN 피연산자)	오버플로		AL	항상(대개 생략됨)	항상(대개 생략됨)					
VC	순서가 지정됨	오버플로 없음									

소유권 고지 사항

이 소유권 고지 사항의 아랫부분에서 달리 명시되지 않는 한<sup>®</sup> 또는 <sup>™</sup> 표시가 있는 단어와 로고는 EU, 대한민국 및 기타 국가에서 ARM Limited의 등록 상표 또는 상표입니다. 이 설명서에 언급된 기타 브랜드와 이름은 해당 소유자의 상표일 수 있습니다.

이 설명서에 포함된 전체 또는 일부 정보나 설명된 제품은 해당 저작권 소유자의 사전 서면 승인 없이는 어떤 형태로도 개조되거나 복제될 수 없습니다.

이 설명서에 설명된 제품은 지속적으로 개발 및 개선될 수 있습니다. 이 설명서에 포함된 모든 제품 명세와 해당 사용법은 ARM의 신뢰하에 제공됩니다. 그러나 ARM에서는 상품성 또는 특정 목적에의 적합성을 비롯하여 그 밖의 묵시적이거나 명시적인 모든 보증을 부인합니다.

이 참조 카드는 제품 사용자를 지원하는 용도로만 만들어졌습니다. ARM Ltd는 이 설명서 정보의 사용, 정보의 오류나 누락 또는 제품의 잘못된 사용에 따른 어떠한 손실이나 손상도 책임지지 않습니다.

설명서 번호

ARM QRC 0007E

변경 내역

발행판	날짜	변경된 내용
A	2004년 11월	첫 번째 릴리스
B	2005년 5월	RVCT 2.2 SP1용 릴리스
C	2006년 3월	RVCT 3.0용 릴리스
D	2007년 3월	RVCT 3.1용 릴리스
E	2008 9월	RVCT 4.0용 릴리스