

LEGv8

Reference Data



①

CORE INSTRUCTION SET in Alphabetical Order by Mnemonic

NAME, MNEMONIC	FOR- FORMAT	OPCODE (9) (Hex)	OPERATION (in Verilog)	Notes
ADD ADD	R	458	$R[Rd] = R[Rn] + R[Rm]$	
ADD Immediate ADDI	I	488-489	$R[Rd] = R[Rn] + ALUImm$	(2,9)
ADD Immediate & Set flags ADDIS	I	588-589	$R[Rd], FLAGS = R[Rn] + ALUImm$	(1,2,9)
ADD & Set flags ADDS	R	558	$R[Rd], FLAGS = R[Rn] + R[Rm]$	(1)
AND AND	R	450	$R[Rd] = R[Rn] \& R[Rm]$	
AND Immediate ANDI	I	490-491	$R[Rd] = R[Rn] \& ALUImm$	(2,9)
AND Immediate & Set flags ANDIS	I	790-791	$R[Rd], FLAGS = R[Rn] \& ALUImm$	(1,2,9)
AND & Set flags ANDS	R	750	$R[Rd], FLAGS = R[Rn] \& R[Rm]$	(1)
Branch B	B	0A0-0BF	$PC = PC + BranchAddr$	(3,9)
Branch conditionally B.cond	CB	2A0-2A7	$if(FLAGs == cond)$ $PC = PC + CondBranchAddr$ $R[30] = PC + 4;$ $PC = PC + BranchAddr$	(4,9)
Branch with Link BL	B	4A0-4BF	$PC = R[Rt]$ $if(R[Rt] != 0)$	(3,9)
Branch to Register BR	R	6B0	$PC = PC + CondBranchAddr$	(4,9)
Compare & Branch if Not Zero CBNZ	CB	5A8-5AF	$PC = R[Rt] \& CondBranchAddr$	(4,9)
Compare & Branch if Zero CBZ	CB	5A0-5A7	$PC = R[Rt] \& CondBranchAddr$	(4,9)
Exclusive OR EOR	R	650	$R[Rd] = R[Rn] ^ R[Rm]$	
Exclusive OR Immediate EORI	I	690-691	$R[Rd] = R[Rn] ^ ALUImm$	(2,9)
Load Register LDUR	D	7C2	$M[Rt] = M[R[Rn]] + DTAddr]$	(5)
Unscaled offset				
Load Signed Word Unscaled offset	LDURSW	D	5C4	$R[Rt] = \{ 32\{ M[R[Rn]] + DTAddr] [31:1], M[R[Rn]] - DTAddr] [31:0] \}$
Load eXclusive Register LDXR	D	642	$R[Rd] = M[R[Rn]] + DTAddr]$	(5,7)
Logical Shift Left LSL	R	69B	$R[Rd] = R[Rn] << shamt$	
Logical Shift Right LSR	R	69A	$R[Rd] = R[Rn] >> shamt$	
				$R[Rd] = Instruction[22:21]*16;$
Inclusive OR ORR	R	550	$R[Rd] = R[Rn] R[Rm]$	(6,9)
Inclusive OR Immediate ORRI	I	590-591	$R[Rd] = R[Rn] ALUImm$	(2,9)
STore Register STUR	D	7C0	$M[R[Rn]] + DTAddr = R[Rt]$	(5)
Unscaled offset				
STore Word Unscaled offset	STURW	D	5C0	$M[R[Rn]] + DTAddr[31:0] = R[Rt][31:0]$
STore eXclusive Register STXR	D	640	$M[R[Rn]] + DTAddr = R[Rt]; R[Rn] = (atomic) ? 0 : 1$	(5,7)
SUBtract SUB	R	658	$R[Rd] = R[Rn] - R[Rm]$	
SUBtract Immediate SUBI	I	688-689	$R[Rd] = R[Rn] - ALUImm$	(2,9)
SUBtract Immediate & Set flags SUBIS	I	788-789	$R[Rd], FLAGS = R[Rn] - ALUImm$	(1,2,9)
SUBtract & Set flags SUBS	R	758	$R[Rd], FLAGS = R[Rn] - R[Rm]$	(1)

- (1) FLAGS are 4 condition codes set by the ALU operation: Negative, Zero, oVerflow, Carry
- (2) ALUImm = { 52'b0, ALU_immediate }
- (3) BranchAddr = { 36[BR_address[25]], BR_address, 2'b0 }
- (4) CondBranchAddr = { 43[COND_BR_address[25]], COND_BR_address, 2'b0 }
- (5) DT.Addr = { 55[DT_address[8]], DT_address }
- (6) MOVImm = { 48'b0, MOV_immediate }
- (7) Atomic test&set pair; R[Rm] = 0 if pair atomic, 1 if not atomic
- (8) Operands considered unsigned numbers (vs. 2's complement)
- (9) Since I, B, and CB instruction formats have opcodes narrower than 11 bits, they occupy a range of 11-bit opcodes

(10) If neither is operand a NaN and Value1 == Value2, FLAGS = 4'b0110;
 If neither is operand a NaN and Value1 < Value2, FLAGS = 4'b1000;
 If neither is operand a NaN and Value1 > Value2, FLAGS = 4'b0010;
 If an operand is a NaN, operands are unordered

ARITHMETIC CORE INSTRUCTION SET

NAME, MNEMONIC	FOR- FORMAT	OPCODE/ SHAMT (Hex)	OPERATION (in Verilog)	Notes
MUL	R	4D8 / 1F	$R[Rd] = (R[Rn] * R[Rm]) (63:0)$	(1,10)
Signed DIVide SDIV	R	4D6 / 02	$R[Rd] = R[Rn] / R[Rm]$	(1,10)
Signed MultiPLY High SMULH	R	4DA	$R[Rd] = (R[Rn] * R[Rm]) (127:64)$	
Unsigned DIVide UDIV	R	4D6 / 03	$R[Rd] = R[Rn] / R[Rm]$	(5)
Unsigned MultiPLY High UMULH	R	4DE	$R[Rd] = (R[Rn] * R[Rm]) (127:64)$	(5)

CORE INSTRUCTION FORMATS

R	opcode	Rm	shamt	Rn	Rd
31	21 20	16 15	10 9	5 4	0
I	opcode	ALU immediate			Rd
31	22 21				0
D	opcode	DT_address	op	Rn	Rt
31	21 20	12 11 10 9			5 4
B	opcode	BR address			
31	26 25	0			
CB	Opcode	COND BR address			Rt
31	24 23				0
IW	opcode	MOV immediate			Rd
31	21 20				5 4

PSEUDOINSTRUCTION SET

NAME	MNEMONIC	OPERATION
CoMPare	CMP	$FLAGS = R[Rn] - R[Rm]$
CoMPare Immediate	CMPI	$FLAGS = R[Rn] - ALUImm$
Load Address	LDA	$R[Rd] = R[Rn] + DTAddr$
MOVE	MOV	$R[Rd] = R[Rn]$

REGISTER NAME, NUMBER, USE, CALL CONVENTION

NAME	NUMBER	USE	PRESERVED ACROSS A CALL?
X0 – X7	0-7	Arguments / Results	No
X8	8	Indirect result location register	No
X9 – X15	9-15	Temporaries	No
X16 (IP0)	16	May be used by linker as a scratch register; other times used as temporary register	No
X17 (IP1)	17	May be used by linker as a scratch register; other times used as temporary register	No
X18	18	Platform register for platform independent code; otherwise a temporary register	No
X19-X27	19-27	Saved	Yes
X28 (SP)	28	Stack Pointer	Yes
X29 (FP)	29	Frame Pointer	Yes
X30 (LR)	30	Return Address	Yes
XZR	31	The Constant Value 0	N.A.

OPCODES IN NUMERICAL ORDER BY OPCODE

Instruction Mnemonic	Format	Opcode Width (bits)	Binary	Shamt Binary	11-bit Opcode Range (1) Start (Hex) End (Hex)
B	B	6	000101		0A0 0BF
B.cond	CB	8	01010100		2A0 2A7
AND	R	11	10001010000		450
ADD	R	11	10001011000		458
ADDI	I	10	1001000100		488 489
ANDI	I	10	1001001000		490 491
ORR	R	11	10101010000		550
ORRI	I	10	1011001000		590 591
CBZ	CB	8	10110100		5A0 5A7
CBNZ	CB	8	10110101		5A8 5AF
STURS	R	11	10111100000		5E0
LDURS	R	11	10111100010		5E2
EOR	R	11	11001010000		650
SUB	R	11	11001011000		658
SUBI	I	10	1101000100		688 689
EORI	I	10	11010001000		690 691
MOVZ	IM	9	110100101		694 697
LSR	R	11	11010011010		69A
LSL	R	11	11010011011		69B
ANDS	R	11	11101010000		750
SUBS	R	11	11101011000		758
SUBIS	I	10	1111000100		788 789
ANDIS	I	10	1111001000		790 791
MOVK	IM	9	111100101		794 797

- (1) Since I, B, and CB instruction formats have opcodes narrower than 11 bits, they occupy a range of 11-bit opcodes, e.g., the 6-bit B format occupies $32 (2^5)$ 11-bit opcodes.

IEEE 754 FLOATING-POINT STANDARD

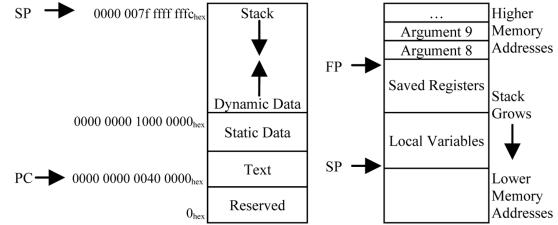
$(-1)^s \times (1 + Fraction) \times 2^{(Exponent - Bias)}$
where Single Precision Bias = 127,
Double Precision Bias = 1023

IEEE Single Precision and Double Precision Formats:

IEEE 754 Symbols		
Exponent	Fraction	Object
0	0	± 0
0	$\neq 0$	\pm Denorm
1 to MAX - 1	anything	$\pm F1.Pt.Nm$
MAX	0	$\pm \infty$
MAX	$\neq 0$	NaN

S.P. MAX = 255, D.P. MAX = 2047

S	Exponent	Fraction
31	30 23 22	0
63	62 52 51	0

MEMORY ALLOCATION

DATA ALIGNMENT

Double Word							
Word				Word			
Halfword		Halfword		Halfword		Halfword	
Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte
0	1	2	3	4	5	6	7

Value of three least significant bits of byte address (Byte 0 to Byte 7)

EXCEPTION SYNDROME REGISTER (ESR)

Exception Class (EC)	Instruction Length (IL)	Instruction Specific Syndrome field (ISS)
31	26	25
		24

EXCEPTION CLASS

EC	Class	Cause of Exception	Number	Name	Cause of Exception
0	Unknown	Unknown	34	PC	Misaligned PC exception
7	SIMD	SIMD/FP registers disabled	36	Data	Data Abort
14	FPE	Illegal Execution State	40	FPE	Floating-point exception
17	Sys	Supervisor Call Exception	52	WPT	Data Breakpoint exception
32	Instr	Instruction Abort	56	BKPT	SW Breakpoint Exception

SIZE PREFIXES AND SYMBOLS

SIZE	PREFIX	SYMBOL	SIZE	PREFIX	SYMBOL
10^3	Kilo-	K	2^{10}	Kibi-	Ki
10^6	Mega-	M	2^{20}	Mebi-	Mi
10^9	Giga-	G	2^{30}	Gibi-	Gi
10^{12}	Tera-	T	2^{40}	Tebi-	Ti
10^{15}	Peta-	P	2^{50}	Pebi-	Pi
10^{18}	Exa-	E	2^{60}	Exbi-	Ei
10^{21}	Zetta-	Z	2^{70}	Zebi-	Zi
10^{24}	Yotta-	Y	2^{80}	Yobi-	Yi
10^{-3}	milli-	m	10^{-15}	femto-	f
10^{-6}	micro-	μ	10^{-18}	atto-	a
10^{-9}	nano-	n	10^{-21}	zepto-	z
10^{-12}	pico-	p	10^{-24}	yocto-	y