

1 Instruction Set Summary

Type	Syntax	ISA			F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	
		1.1	1.2	2.0																	
DSI6	DS=IM6				1	1	1	1	1	1	1	0	0	0	IM6						
CALL	CALL A22				1	1	1	1	-	-	0	0	0	1	A22[21:16]						
					A22[15:0]																
JMPF	GOTO A22				1	1	1	1	1	1	1	0	1	0	A22[21:16]						
					A22[15:0]																
JMPR	GOTO MR				1	1	1	1	1	1	1	0	1	1	-	-	-	-	-	-	
FIR_MOV	FIR_MOV ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	0	1	0	FIR _(a)	
Fraction	FRACTION ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	0	1	1	FRA _(b)	
INT SET	INT FIQ/IRQ/OFF				1	1	1	1	-	-	-	1	0	1	0	0	0	0	F _(c)	I _(c)	
IRQ	IRQ ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	1	0	0	I _(d)	
SECBANK	SECBANK ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	1	0	1	S _(e)	
FIQ	FIQ ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	1	1	F _(f)	0	
IRQ Nest Mode	IRQNEST ON/OFF				1	1	1	1	-	-	-	1	0	1	0	0	1	1	N _(g)	1	
BREAK	BREAK				1	1	1	1	-	-	-	1	0	1	1	-	-	0	0	0	
CALLR	CALL MR				1	1	1	1	-	-	-	1	0	1	1	-	-	0	0	1	
DIVS	DIVS MR,R2				1	1	1	1	-	-	-	1	0	1	1	-	-	0	1	0	
DIVQ	DIVQ MR,R2				1	1	1	1	-	-	-	1	0	1	1	-	-	0	1	1	
EXP	R2 = EXP R4				1	1	1	1	-	-	-	1	0	1	1	-	-	1	0	0	
NOP	NOP				1	1	1	1	-	-	-	1	0	1	1	-	-	1	0	1	
DS Access	DS=Rs/ Rs=DS				1	1	1	1	-	-	-	0	0	0	1	0	W _(n)	Rs _(i)			
FR Access	FR=Rs/ Rs=FR				1	1	1	1	-	-	-	0	0	0	1	1	W _(i)	Rs _(i)			
MUL	MR = Rd* Rs, {ss/us/uu}				1	1	1	S _{Rd(k)}	Rd _(i)			S _{Rd(k)}	0	0	0	0	1	Rs _(i)			
MULS	MR = [Rd]*[Rs], {ss/us/uu},N				1	1	1	S _{Rd(k)}	Rd _(i)			S _{Rd(k)}	1	Size _(i)				Rs _(i)			
Register BITOP	BITOP Rd,Rs				1	1	1	0	Rd _(i)			0	0	0	Bitop _(m)		0	Rs _(i)			
Register BITOP	BITOP Rd,offset				1	1	1	0	Rd _(i)			0	0	1	Bitop _(m)		Offset				
Memory BITOP	BITOP {D:}[Rd],offset				1	1	1	0	Rd _(i)			1	1	D _(n)	Bitop _(m)		Offset				
Memory BITOP	BITOP {D:}[Rd],Rs				1	1	1	0	Rd _(i)			1	0	D _(n)	Bitop _(m)		0	Rs _(i)			
Memory BITOP	BITOP {D:}[A16],offset				1	1	1	1	-	D _(n)	1	0	0	1	Bitop _(m)		Offset				
					A16																
16 bits Shift	Rd=Rd LSFT Rs				1	1	1	0	Rd _(i)			1	0	LSFT _(o)			1	Rs _(i)			
RETI	RETI				1	0	0	1	1	0	1	0	1	0	0	1	1	0	0	0	
RETF	RETF				1	0	0	1	1	0	1	0	1	0	0	1	0	0	0	0	
Base+Disp6	Rd = Rd op [BP+IM6]				OP _(p)				Rd _(i)			0	0	0	IM6						
IMM6	Rd = Rd op IMM6				OP _(p)				Rd _(i)			0	0	1	IM6						
Branch	Jxx label				OP _(q)				1	1	1	0	0	D _(r)	IM6						

Type	Syntax	ISA			F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
		1.1	1.2	2.0																
Stack Operation	Push/Pop R_H, R_L to $[Rs]$																			
DS_Indirect	$Rd = Rd \text{ op } D:[Rs@]$																			
IMM16	$Rd = Rs \text{ op } IMM16$																			
Direct16	$Rd = Rs \text{ op } [A16] \text{ (W = 0)}$ $[A16] = Rd \text{ op } Rs \text{ (W = 1)}$																			
Direct6	$Rd = Rd \text{ op } [A6]$																			
Register	$Rd = Rd \text{ op } Rs \text{ SFT } N$																			
Ext Code(y)																				
Ext Register	$Ra = Ra \text{ op } Rb$																			
Ext Push/Pop	Push R_H, R_L to $[Rb]$ (W = 1) Pop R_H, R_L to $[Rb]$ (W = 0)																			
Ext IMM16	$Ra = Rb \text{ op } IMM16$																			
Ext A16	$Ra = Rb \text{ op } [A16] \text{ (W=0)}$ $[A16] = Ra \text{ op } Rb \text{ (W=1)}$																			
Ext DS_Indirect	$Rx = Rx \text{ op } D:[Ry@]$																			
Ext IM6	$Rx = Rx \text{ op } IM6$																			
Ext Base+Disp6	$Rx = Rx \text{ op } [BP+IM6]$																			
Ext A6	$Rx = Rx \text{ op } [A6]$																			

(a.)

Syntax	FIR
FIR_MOV ON	0
FIR_MOV OFF	1

(b.)

Syntax	FRA
FRACTION OFF	0
FRACTION ON	1

(c.)

Syntax	FIQ	IRQ	F	I
INT OFF	OFF	OFF	0	0
INT IRQ	OFF	ON	0	1
INT FIQ	ON	OFF	1	0
INT FIQ,IRQ	ON	ON	1	1

(d.)

Syntax	I
IRQ OFF	0
IRQ ON	1

(e.)

Syntax	S
SECBANK OFF	0
SECBANK ON	1

(f.)

Syntax	F
FIQ OFF	0
FIQ ON	1

(g.)

Syntax	N
IRQNEST OFF	0
IRQNEST ON	1

(h.)

Syntax	W
Rs = DS	0
DS = Rs	1

(i.)

Syntax	Rs/Rd	Ra/Rb	Rx/Ry
SP	000	0000	
R1	001	0001	
R2	010	0010	
R3	011	0011	
R4	100	0100	
BP	101	0101	
SR	110	0110	
PC	111	0111	
R8		1000	000
R9		1001	001
R10		1010	010
R11		1011	011
R12		1100	100

Syntax	Rs/Rd	Ra/Rb	Rx/Ry
R13		1101	101
R14		1110	110
R15		1111	111

(j.)

Syntax	W
Rs = FR	0
FR = Rs	1

(k.)

Syntax	{S _{Rs} , S _{Rd} }	Rd*Rs/[Rd]*[Rs]	ISA		
			1.1	1.2	2.0
uu	00	Unsigned*Unsigned	Not Support	Support	
us	10	Unsigned*Signed	Support		
ss	11	Signed*Signed			

(l.)

N	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Size	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	0000

(m.)

BITOP	00	01	10	11
Syntax	TSTB	SETB	CLRB	INVB

(n.)

Syntax	D	Description
D:[]	1	The MSB 6-bits of accessing address is data segment (DS).
[]	0	The MSB 6-bits of accessing address is zero.

(o.)

LSFT	000	001	010	011	100	101	110	111
Syntax	ASR	ASROR	LSL	LSLOR	LSR	LSROR	ROL	ROR

(p.)

A : Rd / Ra / Rx

B : [BP+IM6] / IM6 / {D:}[Rs@] / [A6] / Rs SFT N / Rb / {D:}[Ry@]

OP	Type	Syntax	N	Z	S	C
0000	ADD	A += B;	•	•	•	•
0001	ADC	A += B, Carry;	•	•	•	•
0010	SUB	A -= B;	•	•	•	•
0011	SBC	A -= B, Carry;	•	•	•	•
0100	CMP	CMP A, B;	•	•	•	•
0110	NEG	A == B;	•	•	-	-
1000	XOR	A ^= B;	•	•	-	-
1001	LOAD	A = B;	•	•	-	-
1010	OR	A = B;	•	•	-	-
1011	AND	A &= B;	•	•	-	-
1100	TEST	TEST A, B;	•	•	-	-
1101	STORE	B = A;	-	-	-	-

(q.)

OP	Syntax	Condition	Description
0000	JCC	C==0	carry clear
0000	JB	C==0	below (unsigned)
0000	JNAE	C==0	not above and equal (unsigned)
0001	JCS	C==1	carry set
0001	JNB	C==1	not below (unsigned)
0001	JAЕ	C==1	above and equal (unsigned)
0010	JSC	S==0	sign clear
0010	JGE	S==0	great and equal (signed)
0010	JNL	S==0	not less (signed)
0011	JSS	S==1	sign set
0011	JNGE	S==1	not great than (signed)
0011	JL	S==1	Less (signed)
0100	JNE	Z==0	not equal
0100	JNZ	Z==0	not zero
0101	JZ	Z==1	zero
0101	JE	Z==1	equal
0110	JPL	N==0	plus
0111	JMI	N==1	minus
1000	JBE	Not (Z==0 and C==1)	below and equal (unsigned)
1000	JNA	Not (Z==0 and C==1)	not above (unsigned)
1001	JNBE	Z==0 and C==1	not below and equal (unsigned)
1001	JA	Z==0 and C==1	above (unsigned)
1010	JLE	Not (Z==0 and S==0)	less and equal (signed)

OP	Syntax	Condition	Description
1010	JNG	Not (Z==0 and S==0)	not great (signed)
1011	JNLE	Z==0 and S==0	not less and equal (signed)
1011	JG	Z==0 and S==0	great (signed)
1100	JVC	N == S	not overflow (signed)
1101	JVS	N != S	overflow (signed)
1110	JMP	Always	unconditional branch
0000	JCC	C==0	carry clear
0000	JB	C==0	below (unsigned)

(r.)

D	Description
0	Forward jump
1	Backward jump

(s.)

OP	Syntax
1101	PUSH
1001	POP

(t.) Size : Register counts from RH to RL

(u.)

@	00	01	10	11
Syntax	Rs	Rs--	Rs++	++Rs

(v.)

Syntax	A	B	C
Rd = Rs op IMM16	Rd	Rs	IMM16
Rd = Rs op [A16]	Rd	Rs	[A16]
[A16] = Rd op Rs	[A16]	Rs	Rd
Ra = Rb op IMM16	Ra	Rb	IMM16
Ra = Rb op [A16]	Ra	Rb	[A16]
[A16] = Ra op Rb	[A16]	Rb	Ra

OP	Type	Syntax	N	Z	S	C
0000	ADD	$A = B + C;$	•	•	•	•
0001	ADC	$A = B + C, \text{Carry};$	•	•	•	•
0010	SUB	$A = B - C;$	•	•	•	•
0011	SBC	$A = B - C, \text{Carry};$	•	•	•	•
0100	CMP	CMP B, C;	•	•	•	•
0110	NEG	$A = -C;$	•	•	-	-
1000	XOR	$A = B \wedge C;$	•	•	-	-
1001	LOAD	$A = C;$	•	•	-	-
1010	OR	$A = B \vee C;$	•	•	-	-
1011	AND	$A = B \& C;$	•	•	-	-
1100	TEST	TEST B, C;	•	•	-	-
1101	STORE	$C = B;$	-	-	-	-

(w.)

Shift with Barrel shifter :

SFT	000	001	010	011	100	101
Syntax	NOP	ASR	LSL	LSR	ROL	ROR

(x.)

SFC	00	01	10	11
N	1	2	3	4

(y.)

Use 0xFF80 as extension prefix code and followed with extend instruction.