Instruction Specifier	Mnemonic	Instruction	Addressing Modes	Status Bits
0000 0000	STOP	Stop execution	U	
0000 0001	RETTR	Return from trap	U	
0000 0010	MOVSPA	Move SP to A	U	
0000 0011	MOVFLGA	Move NZVC flags to A	U	
0000 010a	BR	Branch unconditional	i, x	
0000 011a	BRLE	Branch if less than or equal to	i, x	
0000 100a	BRLT	Branch if less than	i, x	
0000 101a	BREQ	Branch if equal to	i, x	
0000 110a	BRNE	Branch if not equal to	i, x	
0000 111a	BRGE	Branch if greater than or equal to	i, x	
0001 000a	BRGT	Branch if greater than	i, x	
0001 001a	BRV	Branch if V	i, x	
0001 010a	BRC	Branch if C	i, x	
0001 011a	CALL	Call subroutine	i, x	
0001 100r	NOTr	Bitwise invert r	U	NZ
0001 101r	NEGr	Negate r	U	NZV
0001 110r	ASLr	Arithmetic shift left r	U	NZVC
0001 111r	ASRr	Arithmetic shift right r	U	NZC
0010 000r	ROLr	Rotate left r	U	C
0010 001r	RORr	Rotate right r	U	C
0010 01nn	NOPn	Unary no operation trap	U	
0010 1aaa	NOP	Nonunary no operation trap	i	
0011 0aaa	DECI	Decimal input trap	d, n, s, sf, x, sx, sxf	NZV
0011 1aaa	DECO	Decimal output trap	i, d, n, s, sf, x, sx, sxf	
0100 0aaa	STRO	String output trap	d, n, sf	
0100 1aaa	CHARI	Character input	d, n, s, sf, x, sx, sxf	
0101 0aaa	CHARO	Character output	i, d, n, s, sf, x, sx, sxf	
0101 1nnn	RETn	Return from call with n local bytes	U	
0110 0aaa	ADDSP	Add to stack pointer (SP)	i, d, n, s, sf, x, sx, sxf	NZVC
0110 1aaa	SUBSP	Subtract from stack pointer (SP)	i, d, n, s, sf, x, sx, sxf	NZVC
0111 raaa	ADDr	Add to r	i, d, n, s, sf, x, sx, sxf	NZVC
1000 raaa	SUBr	Subtract from r	i, d, n, s, sf, x, sx, sxf	NZVC
1001 raaa	ANDr	Bitwise AND to r	i, d, n, s, sf, x, sx, sxf	NZ
1010 raaa	ORr	Bitwise OR to r	i, d, n, s, sf, x, sx, sxf	NZ
1011 raaa	CPr	Compare r	i, d, n, s, sf, x, sx, sxf	NZVC
1100 raaa	LDr	Load r from memory	i, d, n, s, sf, x, sx, sxf	NZ
1101 raaa	LDBYTEr	Load byte from memory	i, d, n, s, sf, x, sx, sxf	NZ
1110 raaa	STr	Store r to memory	d, n, s, sf, x, sx, sxf	
1111 raaa	STBYTEr	Store byte r to memory	d, n, s, sf, x, sx, sxf	