



Conditional Branch Summary

Test	Boolean	Mnemonic	Complementary	Boolean	Mnemonic	Comment
$Rd > Rr$	$Z \cdot (N \oplus V) = 0$	BRLT ⁽¹⁾	$Rd \leq Rr$	$Z + (N \oplus V) = 1$	BRGE*	Signed
$Rd \geq Rr$	$(N \oplus V) = 0$	BRGE	$Rd < Rr$	$(N \oplus V) = 1$	BRLT	Signed
$Rd = Rr$	$Z = 1$	BREQ	$Rd \neq Rr$	$Z = 0$	BRNE	Signed
$Rd \leq Rr$	$Z + (N \oplus V) = 1$	BRGE ⁽¹⁾	$Rd > Rr$	$Z \cdot (N \oplus V) = 0$	BRLT*	Signed
$Rd < Rr$	$(N \oplus V) = 1$	BRLT	$Rd \geq Rr$	$(N \oplus V) = 0$	BRGE	Signed
$Rd > Rr$	$C + Z = 0$	BRLO ⁽¹⁾	$Rd \leq Rr$	$C + Z = 1$	BRSH*	Unsigned
$Rd \geq Rr$	$C = 0$	BRSH/BRCC	$Rd < Rr$	$C = 1$	BRLO/BRCS	Unsigned
$Rd = Rr$	$Z = 1$	BREQ	$Rd \neq Rr$	$Z = 0$	BRNE	Unsigned
$Rd \leq Rr$	$C + Z = 1$	BRSH ⁽¹⁾	$Rd > Rr$	$C + Z = 0$	BRLO*	Unsigned
$Rd < Rr$	$C = 1$	BRLO/BRCS	$Rd \geq Rr$	$C = 0$	BRSH/BRCC	Unsigned
Carry	$C = 1$	BRCS	No carry	$C = 0$	BRCC	Simple
Negative	$N = 1$	BRMI	Positive	$N = 0$	BRPL	Simple
Overflow	$V = 1$	BRVS	No overflow	$V = 0$	BRVC	Simple
Zero	$Z = 1$	BREQ	Not zero	$Z = 0$	BRNE	Simple

Note: 1. Interchange Rd and Rr in the operation before the test, i.e., CP Rd,Rr → CP Rr,Rd

Mnemonics	Operands	Description	Operation	Flags	#Clock Note
Arithmetic and Logic Instructions					
ADD	Rd, Rr	Add without Carry	$Rd \leftarrow Rd + Rr$	Z,C,N,V,S,H	1
ADC	Rd, Rr	Add with Carry	$Rd \leftarrow Rd + Rr + C$	Z,C,N,V,S,H	1
ADIW	Rd, K	Add Immediate to Word	$Rd+1:Rd \leftarrow Rd+1:Rd + K$	Z,C,N,V,S	2 ⁽¹⁾
SUB	Rd, Rr	Subtract without Carry	$Rd \leftarrow Rd - Rr$	Z,C,N,V,S,H	1
SUBI	Rd, K	Subtract Immediate	$Rd \leftarrow Rd - K$	Z,C,N,V,S,H	1
SBC	Rd, Rr	Subtract with Carry	$Rd \leftarrow Rd - Rr - C$	Z,C,N,V,S,H	1
SBCI	Rd, K	Subtract Immediate with Carry	$Rd \leftarrow Rd - K - C$	Z,C,N,V,S,H	1
SBIW	Rd, K	Subtract Immediate from Word	$Rd+1:Rd \leftarrow Rd+1:Rd - K$	Z,C,N,V,S	2 ⁽¹⁾
AND	Rd, Rr	Logical AND	$Rd \leftarrow Rd \bullet Rr$	Z,N,V,S	1
ANDI	Rd, K	Logical AND with Immediate	$Rd \leftarrow Rd \bullet K$	Z,N,V,S	1
OR	Rd, Rr	Logical OR	$Rd \leftarrow Rd \vee Rr$	Z,N,V,S	1
ORI	Rd, K	Logical OR with Immediate	$Rd \leftarrow Rd \vee K$	Z,N,V,S	1
EOR	Rd, Rr	Exclusive OR	$Rd \leftarrow Rd \oplus Rr$	Z,N,V,S	1
COM	Rd	One's Complement	$Rd \leftarrow \$FF - Rd$	Z,C,N,V,S	1
NEG	Rd	Two's Complement	$Rd \leftarrow \$00 - Rd$	Z,C,N,V,S,H	1
SBR	Rd,K	Set Bit(s) in Register	$Rd \leftarrow Rd \vee K$	Z,N,V,S	1
CBR	Rd,K	Clear Bit(s) in Register	$Rd \leftarrow Rd \bullet (\$FFh - K)$	Z,N,V,S	1
INC	Rd	Increment	$Rd \leftarrow Rd + 1$	Z,N,V,S	1
DEC	Rd	Decrement	$Rd \leftarrow Rd - 1$	Z,N,V,S	1
TST	Rd	Test for Zero or Minus	$Rd \leftarrow Rd \bullet Rd$	Z,N,V,S	1
CLR	Rd	Clear Register	$Rd \leftarrow Rd \oplus Rd$	Z,N,V,S	1
SER	Rd	Set Register	$Rd \leftarrow \$FF$	None	1
MUL	Rd,Rr	Multiply Unsigned	$R1:R0 \leftarrow Rd \times Rr$ (UU)	Z,C	2 ⁽¹⁾
MULS	Rd,Rr	Multiply Signed	$R1:R0 \leftarrow Rd \times Rr$ (SS)	Z,C	2 ⁽¹⁾
MULSU	Rd,Rr	Multiply Signed with Unsigned	$R1:R0 \leftarrow Rd \times Rr$ (SU)	Z,C	2 ⁽¹⁾
FMUL	Rd,Rr	Fractional Multiply Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$ (UU)	Z,C	2 ⁽¹⁾
FMULS	Rd,Rr	Fractional Multiply Signed	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$ (SS)	Z,C	2 ⁽¹⁾
FMULSU	Rd,Rr	Fractional Multiply Signed with Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$ (SU)	Z,C	2 ⁽¹⁾
Branch Instructions					
RJMP	k	Relative Jump	$PC \leftarrow PC + k + 1$	None	2
IJMP		Indirect Jump to (Z)	$PC(15:0) \leftarrow Z, PC(21:16) \leftarrow 0$	None	2 ⁽¹⁾

Instruction Set Summary (Continued)

Mnemonics	Operands	Description	Operation	Flags	#Clock Note
EIJMP		Extended Indirect Jump to (Z)	$PC(15:0) \leftarrow Z, PC(21:16) \leftarrow EIND$	None	2 ⁽¹⁾
JMP	k	Jump	$PC \leftarrow k$	None	3 ⁽¹⁾
RCALL	k	Relative Call Subroutine	$PC \leftarrow PC + k + 1$	None	3 / 4 ⁽⁴⁾
ICALL		Indirect Call to (Z)	$PC(15:0) \leftarrow Z, PC(21:16) \leftarrow 0$	None	3 / 4 ⁽¹⁾⁽⁴⁾
EICALL		Extended Indirect Call to (Z)	$PC(15:0) \leftarrow Z, PC(21:16) \leftarrow EIND$	None	4 ⁽¹⁾⁽⁴⁾
CALL	k	Call Subroutine	$PC \leftarrow k$	None	4 / 5 ⁽¹⁾⁽⁴⁾
RET		Subroutine Return	$PC \leftarrow STACK$	None	4 / 5 ⁽⁴⁾
RETI		Interrupt Return	$PC \leftarrow STACK$	I	4 / 5 ⁽⁴⁾
CPSE	Rd,Rr	Compare, Skip if Equal	if (Rd = Rr) $PC \leftarrow PC + 2$ or 3	None	1 / 2 / 3
CP	Rd,Rr	Compare	Rd - Rr	Z,C,N,V,S,H	1
CPC	Rd,Rr	Compare with Carry	Rd - Rr - C	Z,C,N,V,S,H	1
CPI	Rd,K	Compare with Immediate	Rd - K	Z,C,N,V,S,H	1
SBRC	Rr, b	Skip if Bit in Register Cleared	if (Rr(b)=0) $PC \leftarrow PC + 2$ or 3	None	1 / 2 / 3
SBRB	Rr, b	Skip if Bit in Register Set	if (Rr(b)=1) $PC \leftarrow PC + 2$ or 3	None	1 / 2 / 3
SBIC	A, b	Skip if Bit in I/O Register Cleared	if(I/O(A,b)=0) $PC \leftarrow PC + 2$ or 3	None	1 / 2 / 3
SBIS	A, b	Skip if Bit in I/O Register Set	if(I/O(A,b)=1) $PC \leftarrow PC + 2$ or 3	None	1 / 2 / 3
BRBS	s, k	Branch if Status Flag Set	if (SREG(s) = 1) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRBC	s, k	Branch if Status Flag Cleared	if (SREG(s) = 0) then $PC \leftarrow PC + k + 1$	None	1 / 2
BREQ	k	Branch if Equal	if (Z = 1) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRNE	k	Branch if Not Equal	if (Z = 0) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRCS	k	Branch if Carry Set	if (C = 1) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRCC	k	Branch if Carry Cleared	if (C = 0) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRSH	k	Branch if Same or Higher	if (C = 0) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRLO	k	Branch if Lower	if (C = 1) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRMI	k	Branch if Minus	if (N = 1) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRPL	k	Branch if Plus	if (N = 0) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRGE	k	Branch if Greater or Equal, Signed	if (N ⊕ V = 0) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRLT	k	Branch if Less Than, Signed	if (N ⊕ V = 1) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRHS	k	Branch if Half Carry Flag Set	if (H = 1) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRHC	k	Branch if Half Carry Flag Cleared	if (H = 0) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRTS	k	Branch if T Flag Set	if (T = 1) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRTC	k	Branch if T Flag Cleared	if (T = 0) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRVS	k	Branch if Overflow Flag is Set	if (V = 1) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRVC	k	Branch if Overflow Flag is Cleared	if (V = 0) then $PC \leftarrow PC + k + 1$	None	1 / 2
BRIE	k	Branch if Interrupt Enabled	if (I = 1) then $PC \leftarrow PC + k + 1$	None	1 / 2

Instruction Set Summary (Continued)

Mnemonics	Operands	Description	Operation	Flags	#Clock Note
BRID	k	Branch if Interrupt Disabled	if (I = 0) then PC ← PC + k + 1	None	1 / 2
Data Transfer Instructions					
MOV	Rd, Rr	Copy Register	Rd ← Rr	None	1
MOVW	Rd, Rr	Copy Register Pair	Rd+1:Rd ← Rr+1:Rr	None	1 ⁽¹⁾
LDI	Rd, K	Load Immediate	Rd ← K	None	1
LDS	Rd, k	Load Direct from data space	Rd ← (k)	None	2 ⁽¹⁾⁽⁴⁾
LD	Rd, X	Load Indirect	Rd ← (X)	None	2 ⁽²⁾⁽⁴⁾
LD	Rd, X+	Load Indirect and Post-Increment	Rd ← (X), X ← X + 1	None	2 ⁽²⁾⁽⁴⁾
LD	Rd, -X	Load Indirect and Pre-Decrement	X ← X - 1, Rd ← (X)	None	2 ⁽²⁾⁽⁴⁾
LD	Rd, Y	Load Indirect	Rd ← (Y)	None	2 ⁽²⁾⁽⁴⁾
LD	Rd, Y+	Load Indirect and Post-Increment	Rd ← (Y), Y ← Y + 1	None	2 ⁽²⁾⁽⁴⁾
LD	Rd, -Y	Load Indirect and Pre-Decrement	Y ← Y - 1, Rd ← (Y)	None	2 ⁽²⁾⁽⁴⁾
LDD	Rd, Y+q	Load Indirect with Displacement	Rd ← (Y + q)	None	2 ⁽¹⁾⁽⁴⁾
LD	Rd, Z	Load Indirect	Rd ← (Z)	None	2 ⁽²⁾⁽⁴⁾
LD	Rd, Z+	Load Indirect and Post-Increment	Rd ← (Z), Z ← Z+1	None	2 ⁽²⁾⁽⁴⁾
LD	Rd, -Z	Load Indirect and Pre-Decrement	Z ← Z - 1, Rd ← (Z)	None	2 ⁽²⁾⁽⁴⁾
LDD	Rd, Z+q	Load Indirect with Displacement	Rd ← (Z + q)	None	2 ⁽¹⁾⁽⁴⁾
STS	k, Rr	Store Direct to data space	(k) ← Rr	None	2 ⁽¹⁾⁽⁴⁾
ST	X, Rr	Store Indirect	(X) ← Rr	None	2 ⁽²⁾⁽⁴⁾
ST	X+, Rr	Store Indirect and Post-Increment	(X) ← Rr, X ← X + 1	None	2 ⁽²⁾⁽⁴⁾
ST	-X, Rr	Store Indirect and Pre-Decrement	X ← X - 1, (X) ← Rr	None	2 ⁽²⁾⁽⁴⁾
ST	Y, Rr	Store Indirect	(Y) ← Rr	None	2 ⁽²⁾⁽⁴⁾
ST	Y+, Rr	Store Indirect and Post-Increment	(Y) ← Rr, Y ← Y + 1	None	2 ⁽²⁾⁽⁴⁾
ST	-Y, Rr	Store Indirect and Pre-Decrement	Y ← Y - 1, (Y) ← Rr	None	2 ⁽²⁾⁽⁴⁾
STD	Y+q, Rr	Store Indirect with Displacement	(Y + q) ← Rr	None	2 ⁽¹⁾⁽⁴⁾
ST	Z, Rr	Store Indirect	(Z) ← Rr	None	2 ⁽²⁾⁽⁴⁾
ST	Z+, Rr	Store Indirect and Post-Increment	(Z) ← Rr, Z ← Z + 1	None	2 ⁽²⁾⁽⁴⁾
ST	-Z, Rr	Store Indirect and Pre-Decrement	Z ← Z - 1, (Z) ← Rr	None	2 ⁽²⁾⁽⁴⁾
STD	Z+q, Rr	Store Indirect with Displacement	(Z + q) ← Rr	None	2 ⁽¹⁾⁽⁴⁾
LPM		Load Program Memory	R0 ← (Z)	None	3 ⁽³⁾
LPM	Rd, Z	Load Program Memory	Rd ← (Z)	None	3 ⁽³⁾
LPM	Rd, Z+	Load Program Memory and Post-Increment	Rd ← (Z), Z ← Z + 1	None	3 ⁽³⁾
ELPM		Extended Load Program Memory	R0 ← (RAMPZ:Z)	None	3 ⁽¹⁾
ELPM	Rd, Z	Extended Load Program Memory	Rd ← (RAMPZ:Z)	None	3 ⁽¹⁾

Mnemonics	Operands	Description	Operation	Flags	#Clock Note
ELPM	Rd, Z+	Extended Load Program Memory and Post-Increment	$Rd \leftarrow (RAMPZ:Z), Z \leftarrow Z + 1$	None	3 ⁽¹⁾
SPM		Store Program Memory	$(Z) \leftarrow R1:R0$	None	- ⁽¹⁾
IN	Rd, A	In From I/O Location	$Rd \leftarrow I/O(A)$	None	1
OUT	A, Rr	Out To I/O Location	$I/O(A) \leftarrow Rr$	None	1
PUSH	Rr	Push Register on Stack	$STACK \leftarrow Rr$	None	2 ⁽¹⁾
POP	Rd	Pop Register from Stack	$Rd \leftarrow STACK$	None	2 ⁽¹⁾
Bit and Bit-test Instructions					
LSL	Rd	Logical Shift Left	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0, C \leftarrow Rd(7)$	Z,C,N,V,H	1
LSR	Rd	Logical Shift Right	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0, C \leftarrow Rd(0)$	Z,C,N,V	1
ROL	Rd	Rotate Left Through Carry	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	Z,C,N,V,H	1
ROR	Rd	Rotate Right Through Carry	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	Z,C,N,V	1
ASR	Rd	Arithmetic Shift Right	$Rd(n) \leftarrow Rd(n+1), n=0..6$	Z,C,N,V	1
SWAP	Rd	Swap Nibbles	$Rd(3..0) \leftrightarrow Rd(7..4)$	None	1
BSET	s	Flag Set	$SREG(s) \leftarrow 1$	SREG(s)	1
BCLR	s	Flag Clear	$SREG(s) \leftarrow 0$	SREG(s)	1
SBI	A, b	Set Bit in I/O Register	$I/O(A, b) \leftarrow 1$	None	2
CBI	A, b	Clear Bit in I/O Register	$I/O(A, b) \leftarrow 0$	None	2
BST	Rr, b	Bit Store from Register to T	$T \leftarrow Rr(b)$	T	1
BLD	Rd, b	Bit load from T to Register	$Rd(b) \leftarrow T$	None	1
SEC		Set Carry	$C \leftarrow 1$	C	1
CLC		Clear Carry	$C \leftarrow 0$	C	1
SEN		Set Negative Flag	$N \leftarrow 1$	N	1
CLN		Clear Negative Flag	$N \leftarrow 0$	N	1
SEZ		Set Zero Flag	$Z \leftarrow 1$	Z	1
CLZ		Clear Zero Flag	$Z \leftarrow 0$	Z	1
SEI		Global Interrupt Enable	$I \leftarrow 1$	I	1
CLI		Global Interrupt Disable	$I \leftarrow 0$	I	1
SES		Set Signed Test Flag	$S \leftarrow 1$	S	1
CLS		Clear Signed Test Flag	$S \leftarrow 0$	S	1
SEV		Set Two's Complement Overflow	$V \leftarrow 1$	V	1
CLV		Clear Two's Complement Overflow	$V \leftarrow 0$	V	1
SET		Set T in SREG	$T \leftarrow 1$	T	1
CLT		Clear T in SREG	$T \leftarrow 0$	T	1
SEH		Set Half Carry Flag in SREG	$H \leftarrow 1$	H	1
CLH		Clear Half Carry Flag in SREG	$H \leftarrow 0$	H	1

AVR Instruction Set

Instruction Set Summary (Continued)

Mnemonics	Operands	Description	Operation	Flags	#Clock Note
MCU Control Instructions					
BREAK		Break	(See specific descr. for BREAK)	None	1 ⁽¹⁾
NOP		No Operation		None	1
SLEEP		Sleep	(see specific descr. for Sleep)	None	1
WDR		Watchdog Reset	(see specific descr. for WDR)	None	1

- Notes:
1. This instruction is not available in all devices. Refer to the device specific instruction set summary.
 2. Not all variants of this instruction are available in all devices. Refer to the device specific instruction set summary.
 3. Not all variants of the LPM instruction are available in all devices. Refer to the device specific instruction set summary. The LPM instruction is not implemented at all in the AT90S1200 device.
 4. Cycle times for Data memory accesses assume internal memory accesses, and are not valid for accesses via the external RAM interface. For LD, ST, LDS, STS, PUSH, POP, add one cycle plus one cycle for each wait state. For CALL, ICALL, EICALL, RCALL, RET, RETI in devices with 16-bit PC, add three cycles plus two cycles for each wait state. For CALL, ICALL, EICALL, RCALL, RET, RETI in devices with 22-bit PC, add five cycles plus three cycles for each wait state.